Cambridge IGCSE · Computer Science 0478 · Notes

Types and methods of data transmission

Read online at **pastopic.com/cambridge/computer-science-0478/notes/transmission-methods**

Last checked 30 September 2026

What you need to know

By the end of this topic you should be able to:

1. **Explain that data is split into packets** before it is sent across a network, rather than being sent as one big block.
2. **Describe the structure of a packet.** Every packet has three parts: a **packet header**, a **payload** and a **trailer**. The header holds the **destination address**, the **packet number** and the **originator's address** (the sender's address).
3. **Describe packet switching.** Data is broken into packets; each packet can take a different route; **routers** control the route each packet takes; packets may arrive out of order; once the last packet has arrived, they are put back in order.
4. **Describe how data travels from one device to another** using each method of transmission: **serial**, **parallel**, **simplex**, **half-duplex** and **full-duplex**, and combinations of them such as "serial half-duplex".
5. **Explain which method suits a given situation**, using the advantages and disadvantages of each method.
6. **Explain how a USB (universal serial bus) connection sends data**, and give its benefits and drawbacks.

Error detection (parity, checksums, echo checks) and encryption are separate topics, even though exam questions often ask about them in the same scenario.

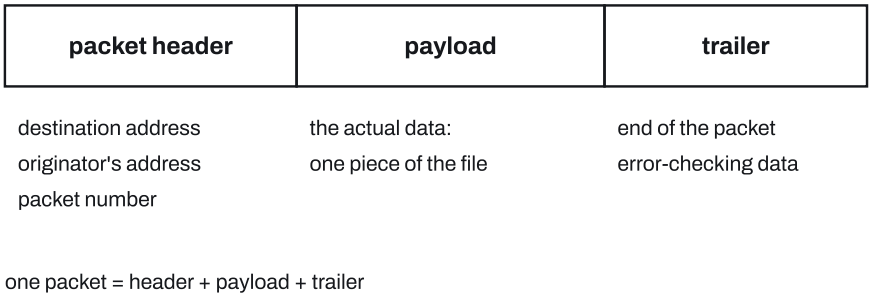
Understand it

Why data is sent in packets

A file sent across a network, such as a photo or an email, is not sent in one piece. It is broken down into small, equal-sized units called **packets**, and each packet is sent separately. If one packet is lost or damaged, only that small packet has to be sent again, not the whole file, and many devices can share the same network because their packets take turns on the links.

For example, the message "SEE YOU AT NOON" could be split into four packets, each carrying four characters: packet 1 "SEE ", packet 2 "YOU ", packet 3 "AT N" and packet 4 "OON".

The structure of a packet

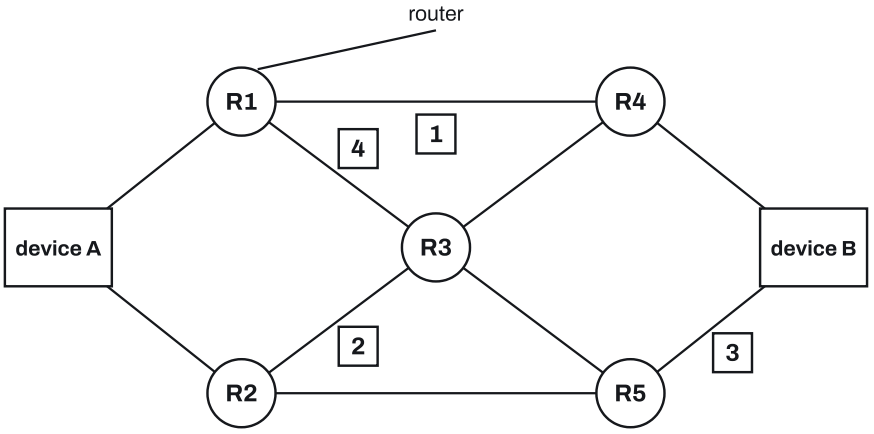


- The **packet header** is the "envelope" information that gets the packet to the right place and back into the right order:
  - the **destination address**: the IP address of the device the packet is going to;
  - the **originator's address**: the IP address of the device that sent it;
  - the **packet number**: this packet's position in the sequence (for example, packet 3 of 4).
- The **payload** is the actual data being sent: one piece of the file or message.
- The **trailer** marks the **end of the packet** and holds **error-checking data** (such as a checksum value), so the receiver can tell whether the packet was damaged on the way.

The payload is **not** part of the header. The header describes the payload; it doesn't contain it.

Packet switching

On a large network such as the internet, there are many possible paths between two devices, joined by **routers**. A router reads the **destination address** in each packet's header and forwards the packet on towards that destination, choosing the most efficient route it can at that moment.



packet 1: A → R1 → R4 → B
packet 2: A → R2 → R3 → R4 → B
packet 3: A → R2 → R5 → B
packet 4: A → R1 → R3 → R5 → B

sent: 1, 2, 3, 4
arrive: 3, 1, 4, 2
reordered: 1, 2, 3, 4

Each packet can take a different route, so they can arrive out of order.
When the last packet has arrived, they are put back in order
using the packet numbers in their headers.

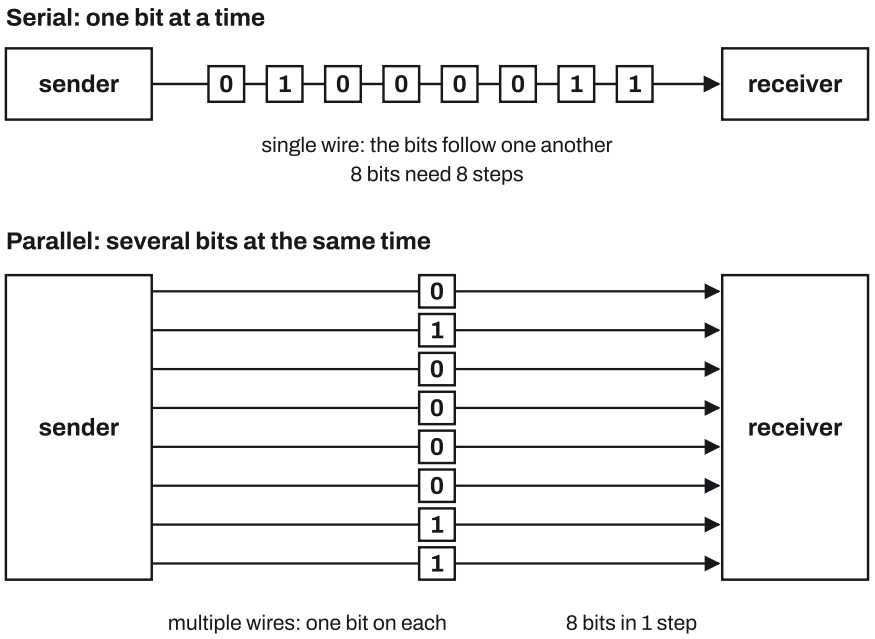
This is **packet switching**:

1. The data is broken down into packets, and each is given a header (with its packet number) and a trailer.
2. Each packet is sent separately, and **each packet can take a different route**.
3. **Routers** decide the route: at each router, the packet is sent on towards its destination address.
4. Because the routes differ in length and in how busy they are, the packets **may arrive out of order** (3, 1, 4, 2 in the diagram). A packet that is lost or corrupted is requested again, so it arrives even later.
5. The receiver waits until **the last packet has arrived**, then **reorders** the packets using their packet numbers and rebuilds the file. It can't finish reordering while any packet is still missing.

Read in arrival order, the example message would be "AT NSEE OONYOU ", which is nonsense; sorted by packet number it is "SEE YOU AT NOON" again.

Serial and parallel: how many bits at a time

The first word of a method's name says **how many wires** are used and so **how many bits travel at once**.



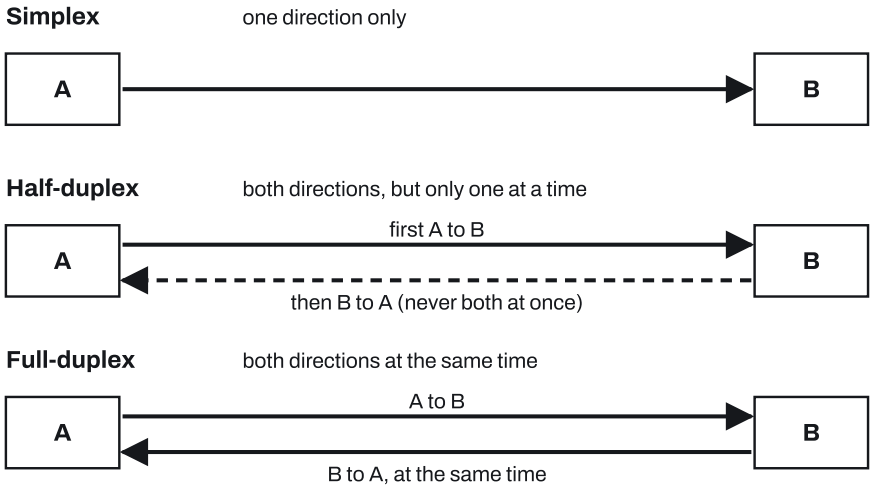
- **Serial**: the data is sent **one bit at a time**, down a **single wire**. The bits arrive one after another, **in the order they were sent**.
- **Parallel**: **several bits are sent at the same time**, down **several wires** (often 8, one byte at a time). For the byte in the diagram, parallel needs 1 step where serial needs 8, so parallel is **faster**.

Parallel has a catch. The wires are never exactly identical, so over a long cable the bits sent together start to arrive at slightly different times and can get out of step. This is called **skewing**, and it corrupts the data. The wires lying side by side also interfere with each other's signals (**crosstalk**). Both problems grow with distance, so parallel is only reliable over **short distances** (up to about 5 metres, such as inside a computer or to a printer on the same desk). Serial has one wire, so its bits can't skew and there's less interference: it is **more reliable over long distances**, and its speed is **adequate** for most jobs. That is why most connections between devices (USB, network cables) are serial.

Be careful with the wording: in serial, **many** bits are sent too, just not **at the same time**. "Parallel sends multiple bits" earns nothing unless you add **at the same time**.

Simplex, half-duplex and full-duplex: which direction

The second word says **which way data can travel**.



- **Simplex**: data is sent in **one direction only**. The receiver can never reply, for example a sensor sending readings to a computer that never needs to answer.
- **Half-duplex**: data can be sent in **both directions, but only one direction at a time**. Each device must wait for the other to finish, like a walkie-talkie. The receiver can reply (for example, a printer telling the computer it has run out of paper), but not while it is receiving.
- **Full-duplex**: data can be sent in **both directions at the same time**, like a phone call. Nobody waits, so there is no delay in either direction.

Two-way methods let the receiver send back confirmations, requests to resend and error messages. Simplex is simpler, and fine when nothing needs to come back.

Putting the two words together

A full method name has one word from each group: serial or parallel (how many bits at once), then simplex, half-duplex or full-duplex (which way). So there are six methods, and a description always has three parts:

Method	Wires	Bits at a time	Direction
Serial simplex	one	one	one direction only
Serial half-duplex	one	one	both, but one at a time
Serial full-duplex	one	one	both at the same time
Parallel simplex	several	several at the same time	one direction only
Parallel half-duplex	several	several at the same time	both, but one at a time
Parallel full-duplex	several	several at the same time	both at the same time

"Serial duplex" in older papers means serial full-duplex.

The USB interface

USB stands for **universal serial bus**. It is the standard connection for plugging devices such as mice, keyboards, printers, phones and flash drives into a computer. As its name says, it is **serial**: data is sent **one bit at a time down a single wire**.

- **Benefits**:
  - it is a **universal** (industry standard) connection, so almost every computer has a USB port and one type of port suits many devices;
  - it can **power** (or charge) the device through the same cable;
  - the device is **detected automatically** when it is plugged in, and its **driver is installed automatically**;
  - it is **backward compatible**: newer ports still work with older USB devices;
  - the connector only fits **one way round**, so it can't be connected wrongly;
  - it supports **different transmission speeds**, and as it is serial the bits don't skew, so there are **fewer errors**.
- **Drawbacks**:
  - it is **slower than a parallel** connection could be;
  - the cable has a **limited length** (a few metres), and a wire can **get in the way** or limit movement (for example, of a mouse).

A USB connection is not the same as a USB flash drive. Benefits of the connection are about how it connects and sends data, not about the storage device being small and portable.

Key facts and formulas

There is no formula list for this subject, so you must learn all of these.

Formula	Status
Packet = packet header + payload + trailer	Learn it
Header: destination address, originator's (sender's) address, packet number	Learn it
Payload: the actual data; trailer: end-of-packet marker and error-checking data	Learn it
Packet switching: split into packets → each can take a different route → routers control the route → may arrive out of order → reordered once the last packet arrives	Learn it
Serial: one bit at a time, one wire, bits arrive in order, reliable over long distances	Learn it
Parallel: several bits at the same time, several wires, faster, short distances only (skew, crosstalk)	Learn it

Formula	Status
Simplex: one direction only	Learn it
Half-duplex: both directions, but only one direction at a time	Learn it
Full-duplex: both directions at the same time	Learn it
USB (universal serial bus): serial, one bit at a time down a single wire	Learn it

How to do it

In the Paper 1 questions we have tagged for this topic (29 questions, 2021–2025), the types came up this often. 20 of the 29 questions mix types, so the counts add up to more than 29.

Question type	Questions
Describe (or draw) how a named method sends data	11
Describe the structure of a packet, or name header items	11
Explain packet switching and the router's role	10
Justify the most suitable method for a scenario	9
Give benefits or drawbacks of a method	6
Identify the method from a description	4
USB: how it sends data, benefits and drawbacks	4

1. Describe (or draw) how a named method sends data (11 questions)

1. **Split the name into its two words**, for example "parallel half-duplex" → parallel + half-duplex.
2. **First word** (one or two marks): serial → "data is sent one bit at a time" and "down a single wire"; parallel → "multiple bits are sent at the same time" and "down multiple wires".
3. **Second word** (one or two marks): simplex → "in one direction only"; half-duplex → "in both directions" and "but not at the same time"; full-duplex → "in both directions" and "at the same time".
4. Write each fact as its own point. A 4-mark answer is exactly those four facts.

For example, serial half-duplex: data is sent one bit at a time; down a single wire; data can be sent in both directions; but only one direction at a time.

Variations you will meet:

- **Draw and annotate a diagram.** Draw two labelled devices joined by the right number of lines (one line for serial, several for parallel). Draw arrowheads for the direction (one arrow for simplex, arrows both ways for duplex). Then annotate in words: "one bit at a time", "single wire", "both directions, not at the same time". Make the drawing and the words agree: don't draw several wires for serial, or arrows both ways for simplex.
- **Contrast two methods on one diagram**, for example serial full-duplex against parallel simplex: one wire with arrows both ways, labelled "at the same time", against several wires with arrows one way only.
- **Define only the direction word**, for example "Define full-duplex": both directions at the same time (both parts are needed).

2. Describe the structure of a packet, or name header items (11 questions)

1. Say a packet has **three sections**: header, payload and trailer.
2. Give what **each** holds: header → destination address, originator's address, packet number; payload → the actual data; trailer → the end of the packet and error-checking data.
3. For "describe the structure", name the part and then what it contains, as separate points.

Variations you will meet:

- **Label a diagram of a packet** (first, second, third section) with each name and an example of what it stores: header, then payload, then trailer, in that order.
- **Name two (or three) items in the header**: any of destination address, originator's address, packet number. Write the whole item ("packet number", not "the sequence").
- **Tick or circle** the item that is (or is not) in the header. The payload, the trailer and anything unrelated (such as "interrupt" or "antivirus") are not header items.
- **Which part holds the destination IP address?** The header.
- **"Name the unit data is broken into"**: packet.
- **Fill in a paragraph** from a word list: packet **header**, **destination address**, **routers**, **last** packet.

3. Explain packet switching and the router's role (10 questions)

1. The data is **broken down into packets** (each with a header, payload and trailer).
2. **Each packet can take a different route** to the destination.
3. A **router** controls the route each packet takes: it reads the destination address and sends the packet on towards it, choosing the most efficient path.
4. Packets **may arrive out of order**.
5. **Once the last packet has arrived**, the packets are **reordered** using their packet numbers.

For a 5-mark "explain how a file is broken into packets and transmitted", mix points from the structure (split into packets, the data is the payload, a header with an example, a trailer with an example) and from the steps above.

Variations you will meet:

- **Name the device** that reads the destination address or chooses the path: **router** (not "IP address", not "server").
- **Name the method** in which each packet takes its own path: packet switching.
- **Give the purpose of a router**: to control the route a packet takes / send each packet towards its destination / choose the most efficient route.
- **Explain why packets need reordering**: each packet can take a different route; some routes take longer than others; so they arrive in the wrong order; a lost or corrupted packet has to be sent again, so it arrives late.
- **Draw a diagram** from device A to device B: draw several routers between them, show packets taking **different routes** through the routers, show them **arriving out of order**, and show them **reordered** at B once all have arrived. Each of those four ideas is a mark; the reordering is the one most often left out.

4. Justify the most suitable method for a scenario (9 questions)

1. **Choose serial or parallel from the distance and the speed needed.**
   - Long distance (another room, another building, more than a few metres, a network) → **serial**: the bits arrive in order and won't be skewed, there's less crosstalk so fewer errors, and its speed is adequate for the amount of data.
   - Very short distance (on the same desk, inside the computer) and speed matters → **parallel**: several bits at a time, so faster, and skew is unlikely over a short cable.
2. **Choose the direction from whether anything needs to come back.**
   - Data only goes one way (a sensor to a display, a microphone to speakers) → **simplex**.
   - Replies, acknowledgements or error messages are needed, but not at the same moment → **half-duplex**.
   - Both sides must send at the same time without waiting (a two-way conversation, orders and confirmations flowing both ways) → **full-duplex**.
3. **Explain each choice using the scenario's details.** Every reason must say why it matters *here*: "the speakers are around the theatre, so the cables are long; serial will not skew over that distance".

Variations you will meet:

- **"Explain why this method is no longer suitable"**, for example parallel after a server is moved 1 km away: over that distance the bits would skew, arrive out of sequence and suffer interference, so the data would be corrupted.
- **"Explain why half-duplex rather than simplex"**: the receiver must send data back (a printer reporting a paper jam or asking for more data), which simplex can't do.
- **Open choice** (any method with a sensible reason): any defensible choice earns marks if the reasons match it. Choose the method whose reasons you can explain most clearly.
- **"Give one other method the company could have chosen"**: any of the other five combinations.

5. Give benefits or drawbacks of a method (6 questions)

Give short, separate points, each about the method named:

- **Serial benefits**: the bits arrive in order (not skewed); less crosstalk/interference, so less chance of error; works over long distances; transmission speed is adequate.
- **Serial drawback**: slower than parallel, because only one bit is sent at a time.
- **Parallel benefit**: faster, because several bits are sent at the same time.
- **Parallel drawbacks**: bits can be skewed or arrive out of sequence; more crosstalk/interference between the wires, so more errors; only suitable for short distances.
- **Half-duplex benefit**: data can go both ways (uploads and downloads). **Drawback**: data can't be sent and received at the same time, so it can be slow, as each side waits.
- **Full-duplex drawback**: sending both ways at once can lead to more data collisions.

When the question says "instead of serial", give what parallel improves: the data is transmitted faster.

6. Identify the method from a description (4 questions)

1. Look for three clues: **how many wires** (one → serial, multiple → parallel), **how many bits at a time**, and **which directions** (one, both one at a time, both at once).
2. Put the two words together: "a single wire, one bit at a time, both directions at the same time" is **serial full-duplex**.

Variations you will meet:

- **A table of statements to tick** against several methods. Tick every method a statement fits: "bits along a single wire" fits every serial method; "may arrive out of sequence" and "only for distances under 5 metres" fit every parallel method; "both directions" fits half- and full-duplex. Serial is about wires; simplex is about direction. Don't mix them up.
- **Circle two methods** that fit a description of one wire used both ways: serial half-duplex **and** serial full-duplex.
- **Spot the odd one out**: "parity" is an error-detection method, not a transmission method.

7. USB: how it sends data, benefits and drawbacks (4 questions)

1. **How it sends data**: USB is serial, so data is sent one bit at a time, down a single wire. If asked for the type of transmission, answer "serial".
2. **Benefits** (one mark each, don't repeat any benefit the question has already given): universal connection; powers or charges the device; device detected and driver installed automatically; backward compatible; can only be inserted one way; supports different speeds; less chance of skew or errors.
3. **Drawback**: slower than parallel; the cable's length is limited and the wire may get in the way.

Worked examples

The bold codes show where marks are earned, as in Cambridge mark schemes. The number is how many marks.

- **B1** mark for a correct result on its own, with no method needed.

Example 1

A photographer uploads a large photo to an online gallery. The photo is broken down into packets and sent across the internet using packet switching.

(a) Describe the structure of a packet. **[4]**

(b) Explain how the packets are transmitted from the photographer's computer to the gallery's server. **[4]**

(c) Identify the device that controls the path each packet takes. **[1]**

(a) A packet has a **packet header B1**, which holds the destination address, the originator's address and the packet number **B1**. It has a **payload B1**, which is part of the photo's data itself. It has a **trailer**, which holds error-checking data and marks the end of the packet **B1**. (Any four of these points.)

(b) Each packet can take a different route to the server **B1**. At each router the packet is sent towards its destination address along the most efficient path **B1**. The packets may arrive at the server out of order **B1**. When the last packet has arrived, they are put back in order using the packet numbers **B1**.

(c) Router **B1**.

Example 2

A design studio has a 3D printer on the same desk as its computer. They are connected using parallel half-duplex data transmission.

(a) Describe how data is sent using parallel half-duplex data transmission. **[4]**

(b) Give one reason the 3D printer must be able to send data to the computer. **[1]**

(c) The studio adds a second computer in an office downstairs, 40 metres away, which will also send print jobs to the printer. Compare serial and parallel transmission for this new link, and recommend one. **[4]**

(a) Multiple bits are sent at the same time **B1**, down multiple wires **B1**. Data can be sent in both directions **B1**, but only in one direction at a time **B1**.

(b) Any one of: to report a fault or that it has run out of filament; to say it is ready for the next part of the job; to confirm the job has finished **B1**.

(c) Parallel would send the data faster, as several bits travel at once **B1**, but over 40 metres the bits sent together would be skewed and arrive out of step **B1**. Serial sends the bits one after another down a single wire, so they arrive in order, with less interference and so fewer errors **B1**. Recommend **serial**: print jobs don't need parallel's extra speed, and serial's speed is adequate **B1**.

Example 3

A school buys document cameras for its classrooms. Each camera will be connected to the teacher's computer, about 2 metres away, either with a USB connection or with a parallel connection.

(a) Recommend which connection the school should use. Justify your recommendation. **[4]**

(b) Identify the method of data transmission that each statement describes. Choose from: serial, parallel, simplex, half-duplex, full-duplex. **[4]**

Statement	Method
The receiver can never reply to the sender.	
The eight bits of a byte leave the sender together, each on its own wire.	
Each device waits for the other to finish before it transmits.	
A reply can come back while data is still being sent.	

(a) USB **B1** (the mark comes with the reasons). Any three of: USB is serial, so the bits arrive in order and won't be skewed **B1**; the camera can be powered through the USB cable, so it needs no separate power lead **B1**; the camera is detected and its driver installed automatically when it is plugged in, so teachers don't have to set it up **B1**; USB is a universal connection, so any classroom computer has a port for it; its speed is adequate for sending the camera's images.

(b) Simplex **B1**; parallel **B1**; half-duplex **B1**; full-duplex **B1**.

Common mistakes

- **Leaving out "at the same time".** "Parallel sends multiple bits" and "full-duplex sends both ways" are incomplete. Serial also sends many bits, just not at the same time; half-duplex also goes both ways, just not at once. Examiners report this almost every session (June 2021, March 2025).
- **Mixing up serial and simplex.** Serial is about the number of wires and bits; simplex is about direction. Candidates ticking "parallel simplex" for "bits along a single wire" confused the two (November 2021 report).
- **Defining the methods instead of justifying them.** When asked why serial suits a scenario, a definition of serial and parallel earns little. Link each property to the scenario, for example the distance between the rooms (March 2021 and March 2023 reports).
- **Saying serial has no errors.** Serial has **less chance** of error, not none (March 2021 report).

- **Saying "it is faster" with no object.** Say what is faster: "the data is transmitted faster" (March 2023 report).
- **Reordering too early.** Packets are reordered **once the last packet has arrived**, because the receiver can't put them in order while some are still missing (March 2025 report). On a packet-switching diagram, the reordering at the end is the step most often left out (June 2023 report).
- **Vague header items.** "The sequence" is not enough; write **packet number**. "Address" alone is not enough; say **destination** or **originator's** address (March 2023 report).
- **Naming the IP address as the device.** The device that forwards packets is a **router**; "IP address" was the most common wrong answer (June 2023 report).
- **Diagrams that contradict their labels**, for example several wires drawn for serial, or arrows both ways for simplex (March 2025 report).
- **Benefits of a USB flash drive instead of a USB connection.** "It is portable" is about the storage device. And don't repeat a benefit the question already gave you, such as "universal" (June and November 2021 reports).

How to get the marks

- **"Describe how data is sent using …"**: one mark per fact. Cover **both** words of the method name, and give each as a separate point: bits at a time, number of wires, direction, and at the same time or not.
- **"Explain why … is suitable" / "Explain your choice"**: each mark is a property **linked to the scenario**: "the kitchen is a long way from the tables, so serial, as bits will not be skewed". Give a reason for **each** word of the method if both are asked about.
- **"Identify" / "State" / "Give"**: a short, exact answer: "router", "packet", "header", "serial". No explanation needed, but use the exact term.
- **"Describe the structure"**: name each part **and** what it holds; three parts named usually earn at least three marks.
- **"Draw and annotate"**: marks come from the annotations as much as the drawing. Label the devices, show the wires and arrows correctly, and write the key facts next to them.
- **"Complete the statements"** from a word list: each term is used once, and some are not used at all.
- **Tick one box**: ticking two boxes scores nothing.
- **Use the context and don't repeat it.** If the question already says "it is a universal connection", give different benefits.

Quick check

1. Name the three parts of a packet, and give one thing each part contains.
2. Data is sent down a single wire, one bit at a time. Both devices can send, but only one at a time. Identify the method.
3. A message is split into packets numbered 1 to 5. They arrive in the order 2, 5, 1, 4, 3. Explain why they arrived out of order, and state when and how the receiver puts them back in order.

4. Give two drawbacks of using parallel data transmission to connect two computers 10 metres apart.
5. Two friends make a video call between their laptops over the internet. Identify the most suitable direction of transmission (simplex, half-duplex or full-duplex) and explain your choice.

Answers

1. Header: the destination address (or the originator's address, or the packet number). Payload: the actual data. Trailer: error-checking data (or the end-of-packet marker).
2. Serial half-duplex.
3. Each packet can take a different route, chosen by the routers, and some routes take longer, so they arrived in a different order. The receiver waits until the last packet (here packet 3) has arrived, then sorts them into the order 1, 2, 3, 4, 5 using the packet numbers in their headers.
4. Any two of: the bits may be skewed (arrive out of sequence) over 10 metres; there is more crosstalk/interference between the wires; so there are more errors in the data.
5. Full-duplex: both friends need to send sound and video to each other at the same time, without waiting for the other to stop.

Past questions to try

- **0478/13/M/J/24 Q7**: draw and annotate serial half-duplex, then its benefits and a drawback.
- **0478/13/M/J/23 Q5**: an annotated diagram of packet switching through routers.
- **0478/12/O/N/23 Q6**: name the method from four descriptions.
- **0478/13/O/N/22 Q4**: how a USB connection sends data, and three of its benefits.