

Hardware

A-Level Computer Science 9618 · Topic 3 revision notes

Everything from the 5 lessons of this topic on a few pages: the key idea of each lesson, the diagram to remember, the points to learn and the traps examiners set. Read it, cover it, test yourself.

- | | |
|---|--|
| 1. <u>Input, output and storage devices</u> | 3. <u>Embedded systems, monitoring and control</u> |
| 2. <u>RAM and ROM: SRAM, DRAM, PROM, EPROM and EEPROM</u> | 4. <u>Logic gates</u> |
| | 5. <u>Logic circuits, truth tables and expressions</u> |

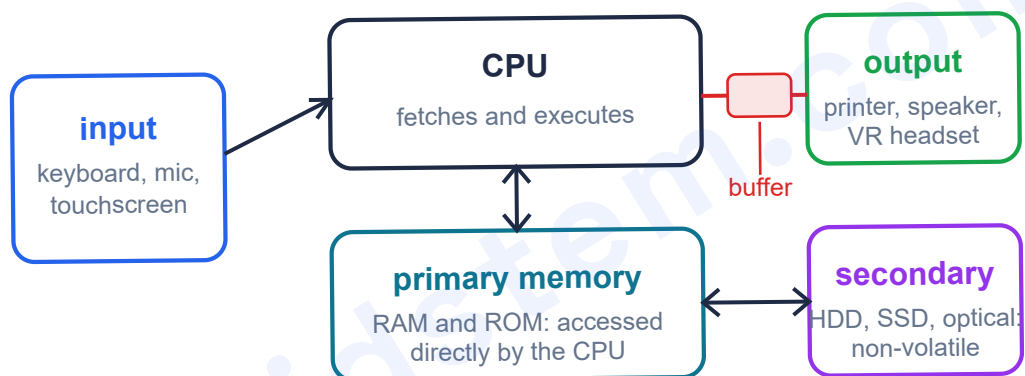
thelucidstem.com

1 Input, output and storage devices

Syllabus 3.1

A computer needs **input devices** to capture data, **output devices** to present results, **primary memory** (RAM and ROM) that the CPU can access **directly** to hold the instructions and data in use, and **secondary storage** that is **non-volatile** to keep programs and files when the power is off; **removable** storage lets data be moved between computers or kept off-site as a backup. At AS you must be able to describe the **principal operation** of each named device as a sequence of steps. A **buffer** is a temporary area of memory that holds data being transferred between two devices or processes that work at different speeds.

why a computer needs all four kinds of hardware



a buffer evens out the speed difference between a fast CPU and a slow device

Fig. 1: input, output, primary memory and secondary storage around the CPU, with a buffer on the way to a slow output device

Worked example: A four-mark description

Describe how a magnetic hard disk drive reads a file. [4]

1. The platters spin at high speed.
2. The actuator arm moves the read head to the correct track.
3. The drive waits until the required sector rotates under the head.
4. The head detects the direction of magnetisation of each area and turns it into 1s and 0s.

Answer: Four separate steps, in order, each worth a mark

One step, one mark.

- ✓ **Operations:** write each device's operation as separate steps in the order they happen.
- ✓ **Use the key words:** drum, charge, toner, fuser; platter, track, sector, head; floating gate; pits and lands.
- ✓ **Justify, do not just name:** secondary storage is needed because it is non-volatile and larger.
- ✓ **Buffers:** different speeds, temporary memory, CPU freed for other tasks, interrupt when empty.

Watch out!

Do not say secondary storage is needed "to store data": RAM stores data too. The mark is for **non-volatile** (kept when the power is off) or for a much **larger capacity**.

A touchscreen is both an input and an output device, and so is a VR headset (screens out, motion sensors in). Say which part does which job.

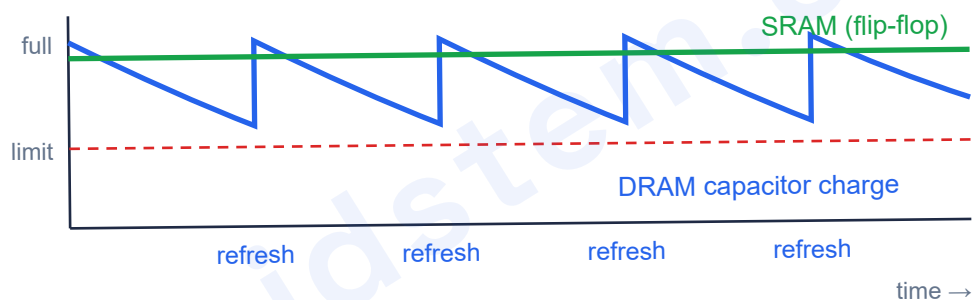
A buffer does not make the printer itself faster. It frees the **processor** so its time is not wasted.

2 RAM and ROM: SRAM, DRAM, PROM, EPROM and EEPROM

Syllabus 3.1

RAM (random access memory) is **volatile**, can be read and written, and holds the operating system, programs and data currently in use. **ROM** (read only memory) is **non-volatile** and holds data that must survive power-off, such as the bootstrap program and firmware. **DRAM** stores each bit as charge on a **capacitor**, which leaks, so it must be **refreshed** constantly; it is cheaper and denser, so it is used for **main memory**. **SRAM** stores each bit in a **flip-flop**, needs no refresh and has a faster access time, so it is used for **cache**. **PROM** is programmed once; **EPROM** is erased with **ultraviolet light**; **EEPROM** is erased **electrically** without removing it from the circuit.

DRAM leaks, so every cell is refreshed; SRAM holds its bit



PROM

programmed once:
fuses are blown

EPROM

erased by UV light,
chip removed

EEPROM

erased electrically,
left in the circuit

Fig. 1: a DRAM cell's charge leaks away and is topped up by a refresh before it falls below the limit; the three programmable ROMs differ in how they are erased

Worked example: Choosing the right RAM

Explain why a processor's cache is made from SRAM but the computer's 16 GiB main memory is made from DRAM.

1. Cache must keep up with the processor, so the deciding factor is access time: SRAM is faster and needs no refresh cycles.
2. Cache is small, so the high cost per bit of SRAM is affordable.
3. Main memory needs a large capacity at a sensible price: DRAM is cheaper per bit and fits more bits on each chip.

Answer: SRAM for speed where little is needed; DRAM for capacity at low cost

One difference explains the rest.

- ✓ **RAM v ROM:** volatile and read-write against non-volatile and read-only.
- ✓ **DRAM:** capacitor, leaks, refreshed, cheaper, denser, main memory.
- ✓ **SRAM:** flip-flop, no refresh, faster, more expensive, cache.
- ✓ **Programmable ROM:** PROM once (fuses); EPROM UV light, whole chip, removed; EEPROM electrically, in the circuit.

Watch out!

"Random access" does not mean the data is random. It means any location can be accessed directly, in the same time, without reading through the others.

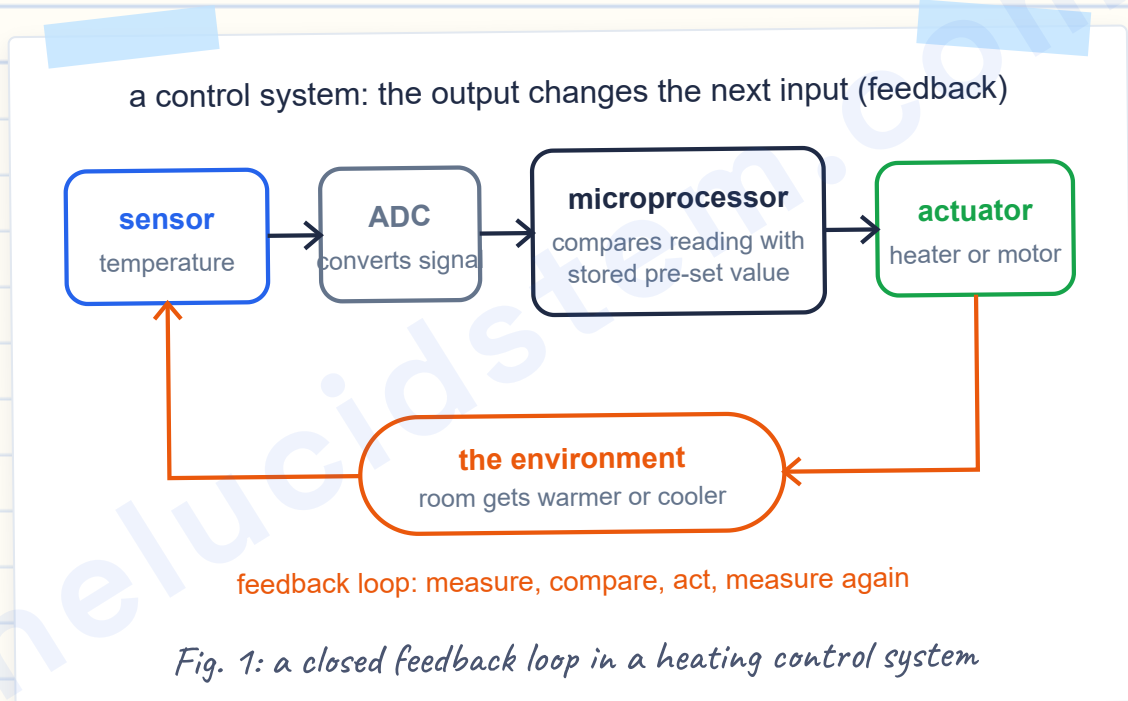
Both SRAM and DRAM are **volatile**. "Static" means no refresh is needed while powered, not that data is kept when the power is off.

EPROM and EEPROM are both reprogrammable. The mark is for how they are erased: **UV light, whole chip, removed** against **electrically, in place, byte by byte**.

3 Embedded systems, monitoring and control

Syllabus 3.1

An **embedded system** is a computer system, with its own microprocessor or microcontroller, memory and input/output, built into a larger device to perform one **dedicated function**. A **monitoring system** uses **sensors** to measure physical quantities and records, displays or warns, but does **not** change the conditions. A **control system** also uses **actuators** to change the conditions. Analogue sensor readings pass through an **ADC**; the processor compares each reading with a **pre-set value**. In a control system the output affects the next input: this **feedback** lets the system keep conditions within limits automatically.



Worked example: Describe a control system

Describe how a microprocessor keeps the soil in a greenhouse moist. [5]

1. A moisture sensor continuously measures the moisture level in the soil.
2. The analogue reading is converted to digital by an ADC and sent to the microprocessor.
3. The microprocessor compares the reading with a pre-set value stored in memory.
4. If the reading is below the pre-set value, it sends a signal to an actuator to open the water valve; if it is at or above the value, the valve is closed.
5. This repeats continuously, so the effect of watering is fed back as new readings (feedback).

Answer: Sensor, ADC, compare with stored value, actuator, continuous loop

Name every link.

- ✓ **Embedded:** built into a larger device, one dedicated function, firmware in ROM or flash.
- ✓ **Monitoring** measures and warns; **control** also changes the conditions with actuators.
- ✓ **The chain:** sensor, ADC, processor compares with a stored pre-set value, signal to actuator, repeat.
- ✓ **Feedback:** output affects the next input, keeping conditions within limits automatically.

Watch out!

A laptop is **not** an embedded system: it is general purpose. The test is whether the computer is part of a larger device and does one dedicated job.

A sensor does not decide anything. Sensors only send readings; it is the **processor** that compares the reading with the stored value and decides whether an actuator is switched.

"The sensor turns the heater on" earns nothing. Name the chain: sensor sends reading, ADC, processor compares with pre-set value, processor signals actuator.

4 Logic gates

Syllabus 3.2

A **logic gate** takes binary inputs (0 or 1) and produces one binary **output**. **NOT** has one input and inverts it; in 9618 **every other gate has exactly two inputs**. **AND** outputs 1 only if both inputs are 1; **OR** outputs 1 if at least one input is 1; **XOR** (EOR) outputs 1 if the inputs are different. **NAND** and **NOR** are AND and OR with the output inverted, shown by a small circle on the symbol. A **truth table** lists the output for every combination of inputs: 2 rows for one input, 4 for two, 8 for three.

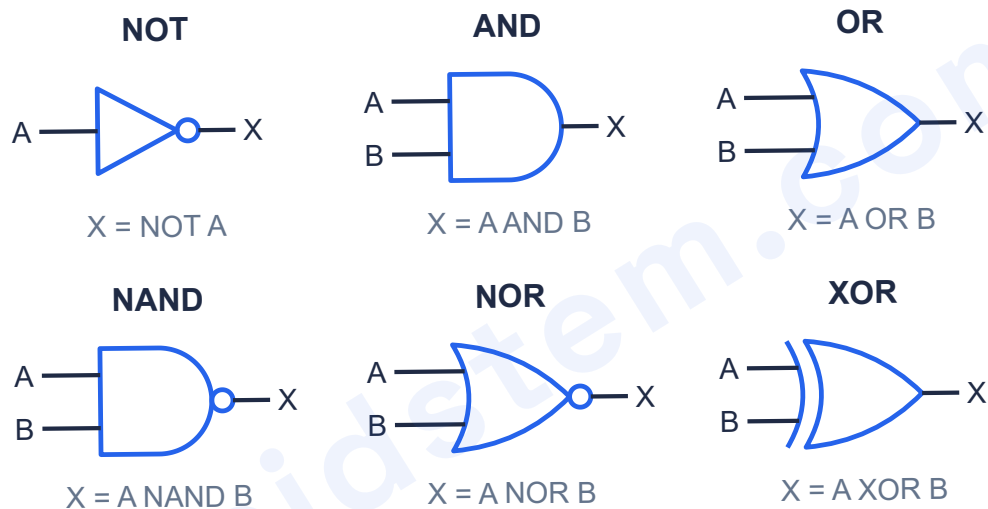


Fig. 1: the standard symbols for the six logic gates used in 9618

Worked example: Reading a gate

A NOR gate has inputs $A = 0$ and $B = 1$. What is the output?

1. Work out OR first: $0 \text{ OR } 1 = 1$.
2. NOR inverts this: $\text{NOT } 1 = 0$.

Answer: 0

One rule per gate.

- ✓ **AND** needs both inputs 1; **OR** needs at least one; **XOR** needs them to be different.
- ✓ **NAND** and **NOR** are AND and OR with every output inverted.
- ✓ **Symbols:** flat back = AND family; curved back = OR family; extra curve = XOR; circle = inverted.
- ✓ **Rows:** list inputs in binary counting order (00, 01, 10, 11) so none is missed.

Watch out!

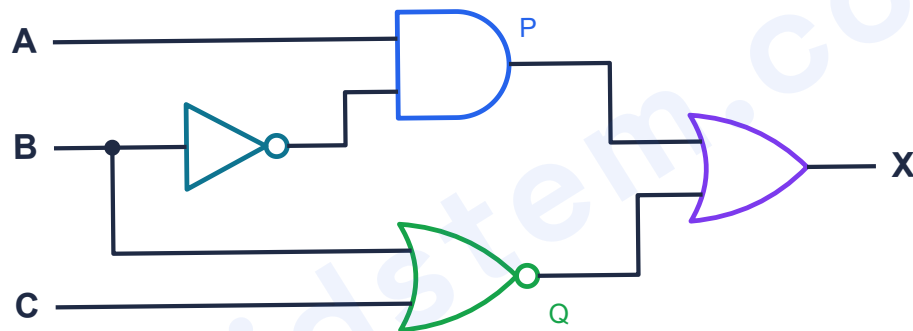
Draw the shapes carefully: AND has a straight back, OR a curved back. A NOT gate must have its circle; a triangle alone is a buffer, not a NOT gate.

Learn the output columns in the order 00, 01, 10, 11: AND 0001, OR 0111, NAND 1110, NOR 1000, XOR 0110. Writing rows in binary counting order means you never miss a combination.

5 Logic circuits, truth tables and expressions

Syllabus 3.2

A **logic circuit** connects gates so that the output of one is an input to the next, from the **inputs on the left** to the **output on the right**. A **logic expression** such as $X = (A \text{ AND } \text{NOT } B) \text{ OR } (B \text{ NOR } C)$ describes the same circuit: each bracket is one gate. To build a **truth table**, list every input combination in binary counting order and add a **working column** for each intermediate gate output. To get an expression from a truth table, take each row where $X = 1$, AND the inputs together (using NOT for each 0), then OR these terms: the **sum of products**. In 9618, every gate except NOT has exactly **two inputs**.



$$X = P \text{ OR } Q = (A \text{ AND } \text{NOT } B) \text{ OR } (B \text{ NOR } C)$$

Fig. 1: a three-input circuit; input B is used twice, shown by the junction dot, and the intermediate outputs are labelled P and Q

Worked example: One row by hand

For the circuit in Fig. 1, find X when $A = 1$, $B = 0$, $C = 1$.

1. $P = A \text{ AND } \text{NOT } B = 1 \text{ AND } \text{NOT } 0 = 1 \text{ AND } 1 = 1$.
2. $Q = B \text{ NOR } C = \text{NOT } (0 \text{ OR } 1) = \text{NOT } 1 = 0$.
3. $X = P \text{ OR } Q = 1 \text{ OR } 0 = 1$.

Answer: $X = 1$ (row 101 of the table)

Bracket by bracket.

- ✓ **Expression to circuit:** one gate per bracket, two inputs per gate (except NOT), label inputs and output.
- ✓ **Truth table:** rows 000 to 111 in order, a working column for each intermediate gate.
- ✓ **Statement to expression:** one letter per condition, brackets around conditions that belong together.
- ✓ **Table to expression:** one AND term per row with $X = 1$, joined with OR; check one row of each output.

Watch out!

Working columns earn marks and catch errors. Examiners give credit for a correct output column; a missing or out-of-order row loses it, so always list the inputs 000 to 111.

Do not draw a three-input gate. To combine three conditions with AND, use two two-input gates, for example $(A \text{ AND } B) \text{ AND } C$.

A sum-of-products answer is accepted even if it is not the simplest. A shorter expression is only right if it gives the **same output for all 8 rows**: check it with a truth table.