## Learning hook

In order to pique curiosity, put the question “What makes a computer a computer?” on the board and ask learners to discuss. This discussion should stimulate thinking around software, hardware, form, function and purpose.

Explain to students that a single computer system comprises both hardware and software working together.

**Software**

Computer software can be application software, system software or utility software.  
An operating system is just one type of software, but it is the central one. Its role in a computer system is the management of resources and hardware operations. Users and software applications need the services of operating systems in order to make system calls and to use application programming interfaces (APIs). Users interact with a computer operating system through the computer interface, which on most computers is a graphic user interface (GUI).   
In summary, an operating system allows user interaction with computer systems by acting as an interface between users or application programs and the computer hardware.

**Hardware**

Computer hardware may be an input, output, storage or processing component, such as a central processing unit (CPU). Have students give an example of each of these components.  
The CPU takes data and instructions from the storage unit and makes calculations based on the instructions given and the type of data provided. It is then sent back to the storage unit. The CPU includes the arithmetic logic unit (ALU) and the control unit.

*Fetch–execute cycle*

Provide a brief overview of the fetch–execute cycle, described below.

In the fetch–execute cycle, three major sections of a CPU work together. These are the registers (address register, program counter, instruction register, data registers and accumulator), the ALU and the control unit.

All these sections work to the drumbeat of the system clock at billions of times per second. The speed is measured by the clock speed of the computer, now typically around 3 Gigahertz or 3 billion cycles per second.

A single machine cycle proceeds as follows:

1. Fetch or read instruction and address: The **program counter** points to the address in memory of the current program instruction – an operation and an operand. All the information contained at this address is written to the **instruction register**.
2. Decode instruction and send data to address register: The operand is sent to the **address register** and the operator is decoded by the **instruction register interpreter**.
3. Execute the instruction: This will often be a single step, as in the case of a simple LOAD or STORE instruction. However, in our drama example, the ADD will involve two steps:
   * In the first of these the number held at the location indicated by the address register from Stage 2 is added in the**ALU** to the number stored (at some earlier cycle) in the **accumulator**.
   * In the second stage the sum is sent back to the accumulator and wipes out the previous value stored there while the CPU increments the program counter.

*Role-playing a machine cycle*

1. Give students the machine cycle script: [Script: Machine cycle drama (PDF)](https://www.digitaltechnologieshub.edu.au/docs/default-source/getting-started-years-9-10/interacting-with-hardware-and-software/machine_cycle_drama.pdf?sfvrsn=2).
2. Select nine students for the roles. Each student highlights their parts on their script. Students may be seated out the front of a whiteboard and their roles can be written on the board above them to assist the class to follow. It is helpful to have the script projected so the class can follow as it is read.
3. Students play roles and read aloud a drama based on the cycle. Explain that the purpose is to understand the roles of the various parts of the computer as a program is executed. Tell students they will be asked to summarise the roles at the end of hearing the drama.
4. Afterwards, students complete: 'Worksheet: Review questions and solutions A'. (Note: the solutions are included with this file.) Students have access to the script as they do this. This could be set as homework.

## Learning map and outcomes

This lesson covers the roles of hardware and software in managing, controlling and securing the movement of and access to data in networked digital systems. It first examines the operation of the CPU in the fetch–execute cycle. It then examines the functions of devices that enable networked communications between computers.   
Students gain an appreciation of the ‘small picture’ examination of the operation of a computer and later the ‘big picture’ examination of large-scale networking.

You could also focus on the skillset and mindsets that learners mind need to adopt and use during this project, this ties in with the [Creative and Critical Thinking Capabilities](http://www.australiancurriculum.edu.au/generalcapabilities/critical-and-creative-thinking/introduction/key-ideas).

## Learning input

1. Introduce and explain very briefly the functions of the main **physical components** of a networked computer system:
   * modem
   * router
   * hub
   * switch
   * bridge
   * physical media: (optic fibre/twisted pair/satellite).
2. Introduce and explain very briefly the function of the main **protocols**used by two networked computers:

* TCP
* IP
* Internet protocols: HTTP/SMTP/FTP.

Prepare a lucky dip of these terms. Duplicate sufficient for each student to receive one. Students with identical terms form groups to research that term together. Students research the main function of the item and when it was invented.

Students present to the class the results of their research for each of the items. Ask students with the **physical components** to present first, followed by those researching **protocols**.

The following indicates the level of detail expected from the student lucky-dip research.

Explain: First introduced in 1973, TCP is what applications use to communicate with one another. It was created to unify the many networking protocols. The web browser communicates with network software using TCP. TCP breaks down the data communicated between applications into packets so that IP can send them to another computer. IP is the communication that takes place between computers. It is IP that actually sends packets between computers and routes them to their correct destination. TCP reassembles the packets once delivered by IP.

## Learning construction

1. Select one student from the existing groups to role-play a **receiving computer**, which will **receive** an email communication from the class’s simulated human network.
2. Prepare a message and place it **inside**an envelope.
3. Make clear to the class that you (the teacher) represent the **sending**computer. Write '**EMAIL**' on the front of the envelope.
4. Groups make a label description of their function. Groups each sit in separate parts of the room.
5. Students remain in their original group, although it is preferable now to split the TCP/IP group into two separate groups: TCP and IP.
6. Form an additional group by selecting one member from the larger sized groups. They form a new Internet service provider (ISP) group.
7. Groups elect a spokesperson.
8. Groups spend time preparing and clarifying the precise functions they will perform when or if they receive data (the email message). They will be told from whom they will receive it and to whom they will be sending it.
9. Write the following for all to see, as a suggested order in which the message will travel:

transmitting computer (teacher), Internet protocol, TCP, IP, modem, ISP, router, ISP, switch, bridge, router, hub, modem, IP, TCP, receiving computer.

(Note: [How Does the Internet Work](https://web.stanford.edu/class/msande91si/www-spr04/readings/week1/InternetWhitepaper.htm) is a useful diagram.)

1. Hand the message to the Internet protocol group.
2. The message now begins its journey. Each group’s spokesperson in turn describes the function they perform on the message and who they will next hand it to until it reaches the final destination computer.

## Learning demo

Students view the episode: [The Internet: Packets, Routing and Reliability](https://www.youtube.com/watch?v=ZhEf7e4kopM&feature=youtu.be&list=PLzdnOPI1iJNfMRZm5DDxco3UdsFegvuB7), part of the excellent expanding video series produced by code.org. The role-play is repeated now that the functions have been clarified and better understood.

## Curriculum links

| Links with Digital Technologies Curriculum Area | |
| --- | --- |
| **Strand** | **Content Description** |
| **Knowledge and Understanding** | Investigate how hardware and software manage, control and secure access to data in networked digital systems [(AC9TDI10K01)](https://v9.australiancurriculum.edu.au/f-10-curriculum.html/learning-areas/digital-technologies/year-9_year-10/content-description?subject-identifier=TECTDIY910&content-description-code=AC9TDI10K01&detailed-content-descriptions=0&hide-ccp=0&hide-gc=0&side-by-side=1&strands-start-index=0&subjects-start-index=0&view=quick). |

## Assessment

Note: Criteria are cumulative.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Quantity of knowledge |  |  | Quality of understanding |  |
| Fetch-execute cycle | No evidence of understanding | Student is able to describe the purpose of the CPU | Student is able to describe the function of the CPU | Student is able to describe the functions of each part of the CPU | Student is able to apply the fetch-execute cycle to the addition of two integers |
| Computer networking | No evidence of understanding | Student can identify a number of devices | Student can describe the functions of a number of devices | Student can describe the functions of all devices and demonstrates and understanding of the purpose of each | Student can describe the functions of all devices and demonstrates understanding of the interactions between them |
| Optional score | 0 | 1 | 2 | 3 | 4 |