Learning Aims

- Describe the factors affecting the performance of the CPU:
  - clock speed
  - number of cores
  - cache
- Understand the use of **pipelining** in a processor to improve **efficiency**
- Understand how address and data bus size relates to assembly language programs

Key words

Key term	Definition
Clock	The CPU's internal timer that sends out regular pulses to synchronise operations. Its speed (clock speed, measured in GHz) determines how many instructions can be processed per second.
Cache	A very fast type of memory inside or close to the CPU that stores frequently used instructions and data, reducing the time needed to fetch them from main memory.
Cores	Independent processing units inside a CPU. Each core can fetch, decode, and execute its own instructions, allowing true parallel processing.
Pipelining	A technique where the CPU overlaps the steps of different instructions (fetch, decode, execute), so several instructions are in progress at once.
Increases throughput	means that the processor is able to execute more instructions per second.
Idle components	Pipelining means all major CPU components (e.g., fetch unit, decode unit, ALU) are working at the same time on different instructions. It reduces wasted time and prevents components from being idle.
Word size	The number of bits the CPU can process or transfer at once (e.g., 32-bit or 64-bit). A larger word size means more data can be handled in a single operation.
Buffer	a temporary storage area used to hold data while it is being transferred from the CPU to other components.

Factors affecting processor performance

The main factors affecting processor performance are:

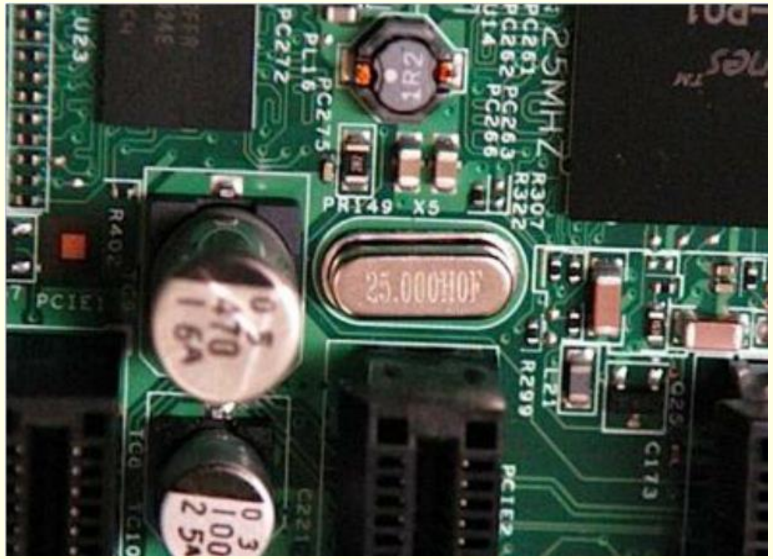
- **Clock speed**
- The number of **cores**, or duplicate processors, linked together on a single chip
- The amount and type of **cache memory**

System Clock

- Using a quartz crystal, the clock in a computer breathes life into the microprocessor by feeding it a constant flow of pulses.

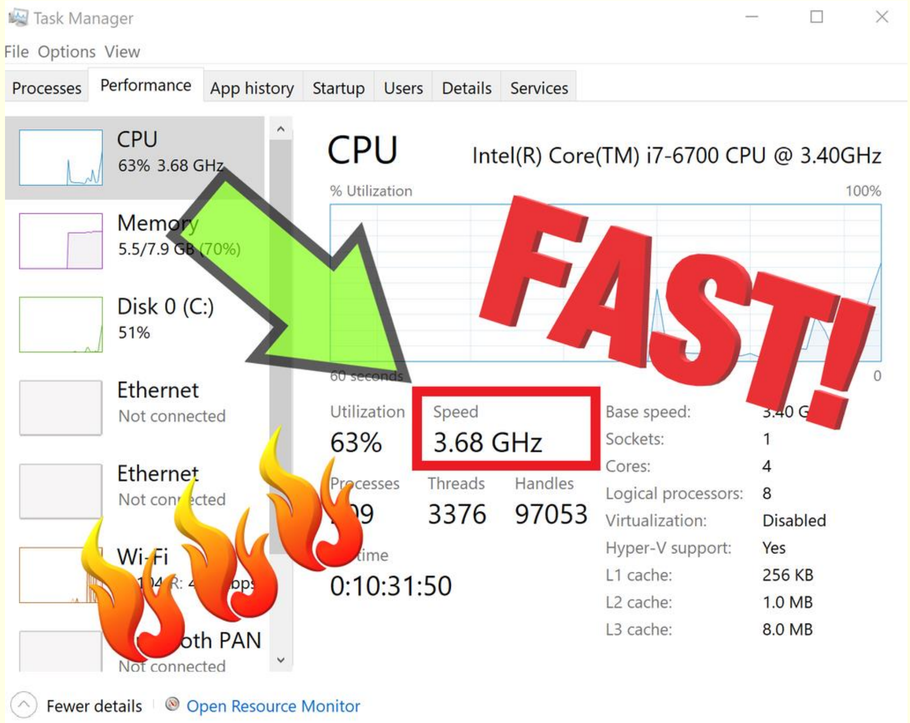
- The clock rate of a CPU is normally determined by the frequency of an oscillator crystal.

- In other words, the **frequency of electronic signals per second**



System Clock

- A clock determines the speed of the CPU.
- Regular electrical pulse which synchronises all the components.
- **The faster the clock the more instructions can be executed per second.**
- The speed of the clock is measured in **Hertz** (Hz), which is the **amount of cycles per second**.
- **A clock speed of 500Hz means 500 cycles per second.**
- Current computers have clock speeds of **3GHz**, which means **3-billion cycles per second.**
- Each ‘tick’ means that one part of the fetch-decode-execute cycle can be carried out.



CPU 63% 3.68 GHz

Memory 5.5/7.9 GB (70%)

Disk 0 (C:) 51%

Ethernet Not connected

Ethernet Not connected

Wi-Fi

oth PAN Not connected

CPU Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz

% Utilization 100%

60 seconds 0

Utilization 63%

Speed 3.68 GHz

Processes 9 Threads 3376 Handles 97053

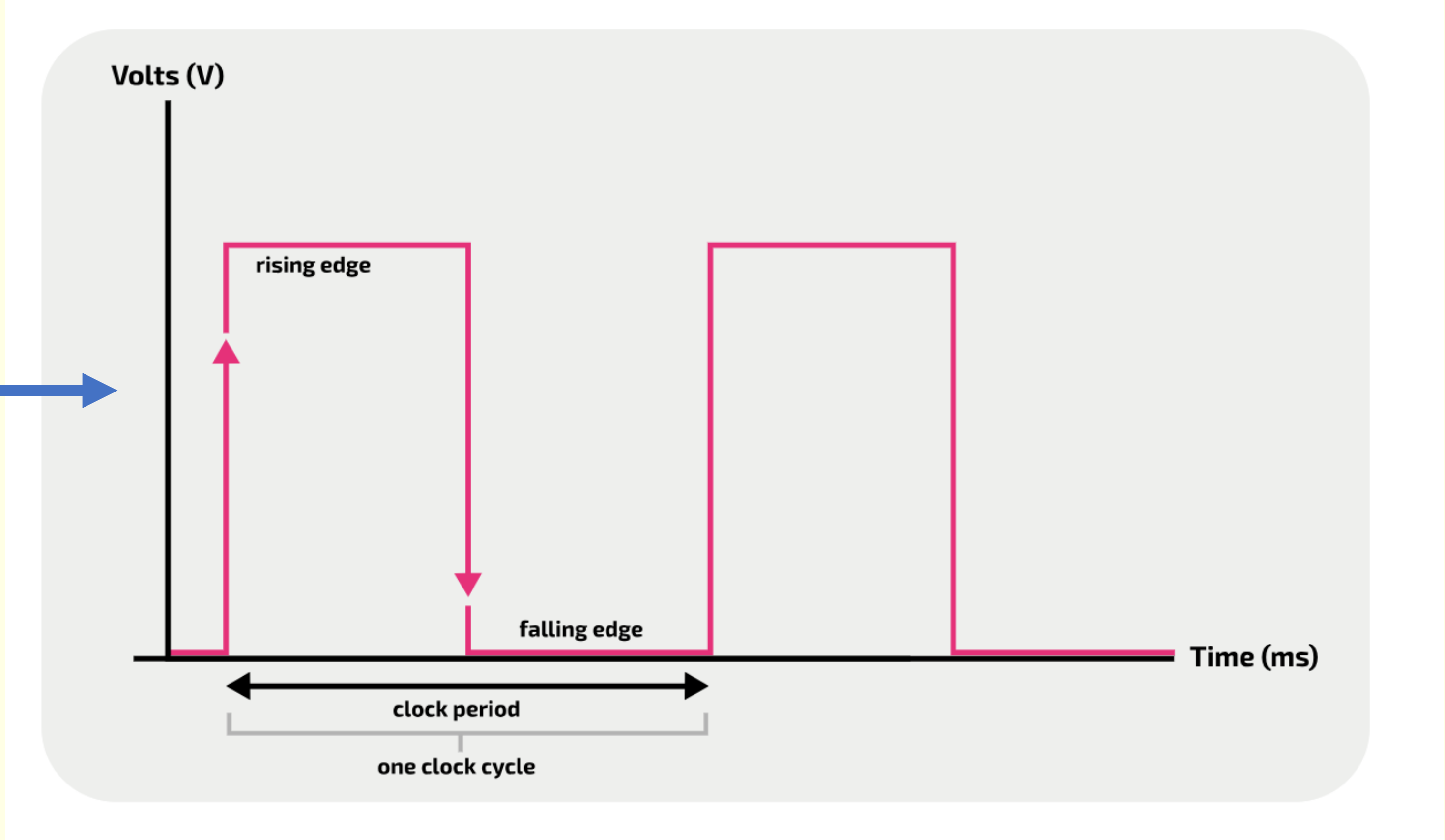
time 0:10:31:50

Base speed:	3.40 G
Sockets:	1
Cores:	4
Logical processors:	8
Virtualization:	Disabled
Hyper-V support:	Yes
L1 cache:	256 KB
L2 cache:	1.0 MB
L3 cache:	8.0 MB

System Clock

- The **system clock** — also simply referred to as the **clock** — generates regular clock pulses by emitting a signal that continuously switches between a low (or '0') and a high (or '1') state.

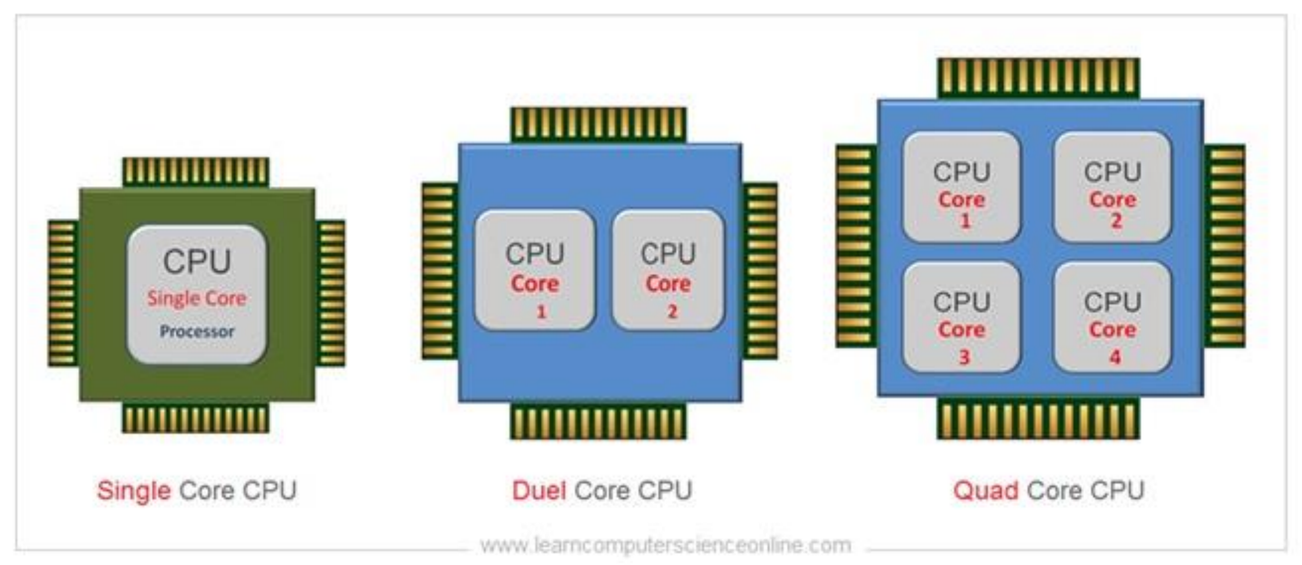
On the rising edge a FDE task is carried out



The time taken between two sequential rising edges is called a **clock cycle** or a **clock period**. The **clock speed** is measured by the number of clock cycles in one second — 1 clock cycle per second is 1Hz.

Cores

- A CPU traditionally had one 'core' but processors these days might be dual-core or quad-core
- For example, a core is actually a processor with its own cache.
- So a dual-core CPU has not one but two processors.
- A quad core CPU has four processors.
- Each core is theoretically able to process a different instruction at the same time with its own fetch-execute cycle, making the processor two or even four times faster with a quad-core chip.
- However, although a dual-core processor has twice the power, it does not always perform twice as fast, because the **software may not always be able to take full advantage of both processors.**

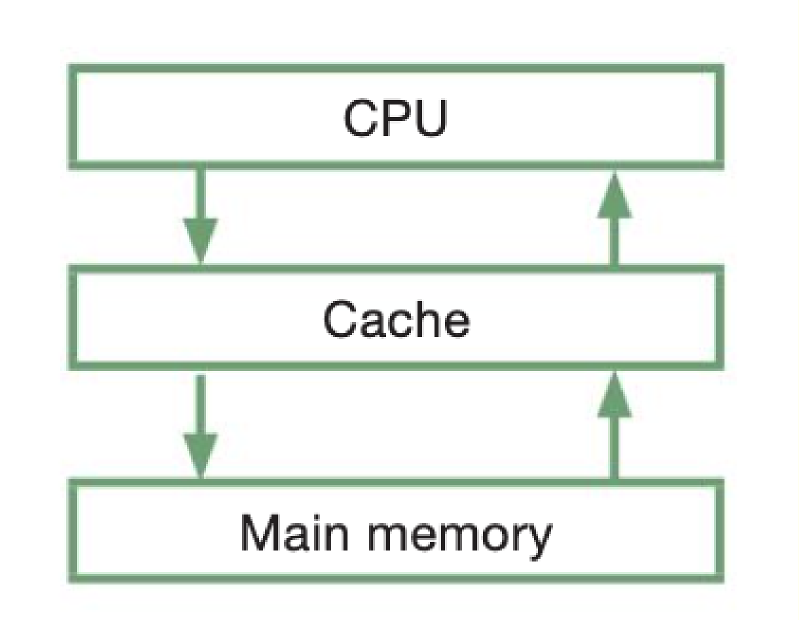


Amount and type of cache memory

Cache is a small amount of **expensive, very fast memory inside the CPU**.

When an instruction is fetched from main memory it is copied into the cache so if it is needed again soon after, it can be fetched from cache, which is much quicker than going back to main memory.

As cache fills up, unused instructions or data still being held are replaced with more recent ones.



There are different “levels” of cache:

- Level 1 cache is extremely fast but small (between 2-64KB)
- Level 2 cache is fairly fast and medium-sized (256KB-2MB)
- Some CPUs also have Level 3 cache

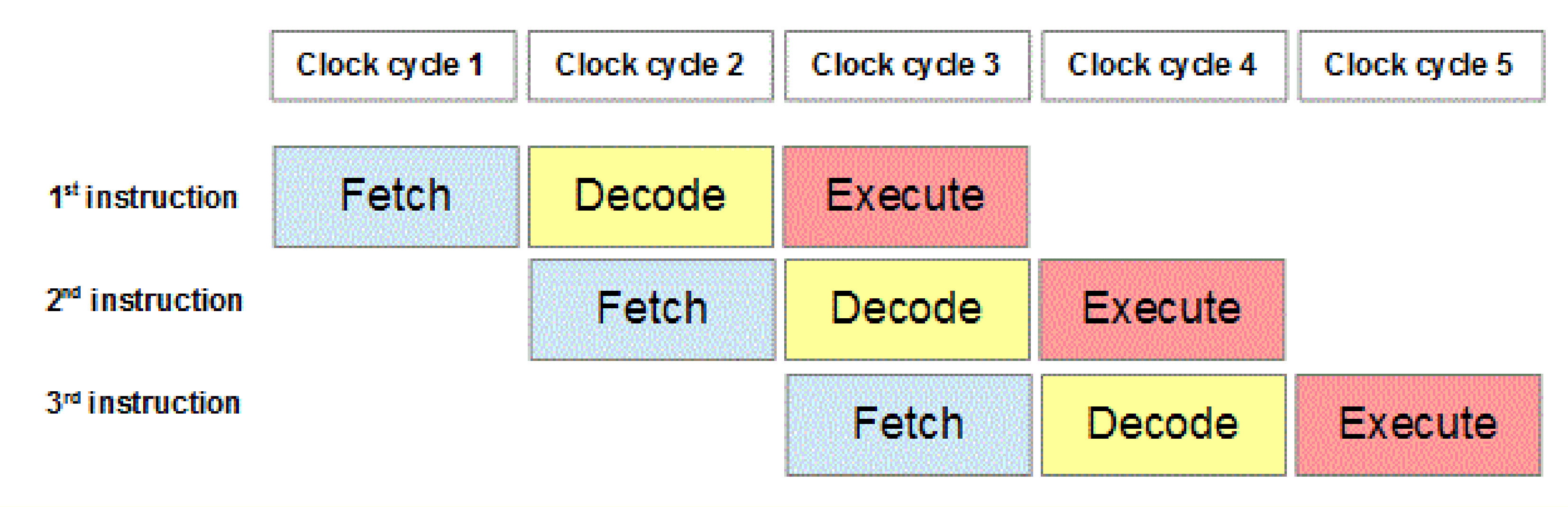
Pipelining

Pipelining is a technique used by some processors to improve performance.

Without pipelining, the steps in the Fetch-Execute cycle take place one after the other.

- Pipelining allows the next instruction to be fetched **whilst** the previous one is being decoded/executed
- It allows the overlapping of different parts of the FDE
- It increases **throughput** // increases the number of instructions processed in a set period of time
- It prevents the CPU having to wait and prevents idle components

Pipelining is now common in microprocessors used in personal computers. Intel's Pentium chip uses pipelining to execute as many as six instructions simultaneously.



	Clock cycle 1	Clock cycle 2	Clock cycle 3	Clock cycle 4	Clock cycle 5
1st instruction	Fetch	Decode	Execute		
2nd instruction		Fetch	Decode	Execute	
3rd instruction			Fetch	Decode	Execute

Words and word size

Address bus word size

- Each word, or group of bytes, in memory has its own specific address.
- When the processor wishes to read a word of data from memory, it first puts the address of the desired word on the address bus.
- The width of the address bus determines the maximum possible memory capacity of the system.
- For example, if the address bus consisted of only 8 lines, then the maximum address it could transmit would be (in binary) 11111111 or 255, giving a maximum memory capacity of 256 (including address 0).
- A system with a 32-bit address bus can address 2^{32} (4,294,967,296) memory locations giving an addressable memory space of 4GiB.

Words and word size

Data bus word size

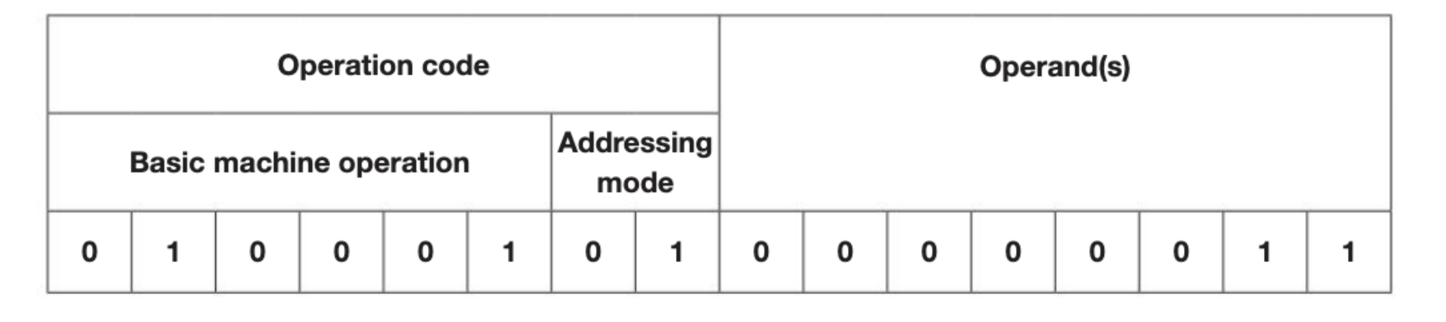
- The data bus transmits the data held in a word of memory, between processor components and memory.

- The largest operand (which is either an address or an actual value) that can be held in a word is therefore related to the size of the data bus.

- If the data bus is 16 bits wide, a word cannot hold an integer greater than 2^{16} at a time, or allow more bits per instruction.

How this relates to assembly language

The basic structure of a machine code instruction in a computer with a 16-bit word may take the format shown below:



Operation code								Operand(s)							
Basic machine operation						Addressing mode									
0	1	0	0	0	1	0	1	0	0	0	0	0	0	1	1

In assembly language, the operation code (opcode) will be expressed as a mnemonic such as ADD, SUB, LDA (load into the accumulator) etc.

With only six bits for the opcode, there cannot be more than 2^6 Operand(s) -1, or more than two characters.

A wider data bus can transmit larger values, or more characters different instructions.

The operand has to be held in only 8 bits.

This would clearly not be practical in a **general purpose computer** which is more likely to have a word size of **32, 64 or 128 bits.**

Summary

Clock Speed:	The frequency at which a CPU executes instructions, measured in cycles per second (Hertz). Higher clock speeds generally result in better performance, but other factors also play a significant role.
Instructions per Cycle (IPC):	The number of instructions a CPU can execute per clock cycle. Increasing IPC improves performance without requiring an increase in clock speed.
Cache Size and Efficiency:	The CPU cache stores frequently accessed data and instructions, reducing the need to fetch them from main memory. Larger and more efficient caches enhance performance.
Number of Cores:	Multi-core CPUs contain multiple processing units, enabling them to execute multiple tasks simultaneously. Software must be optimized to take advantage of multiple cores to see performance benefits.
Instruction Set Architecture (ISA):	The design of the CPU's instruction set affects its performance. Complex instruction set computers (CISC) have fewer instructions that perform more complex operations, while reduced instruction set computers (RISC) have a larger number of simpler instructions.
Pipelining:	Pipelining breaks down the execution of instructions into multiple stages, allowing different stages of different instructions to be executed concurrently. A typical pipeline includes stages such as instruction fetch, instruction decode, execute, memory access, and write back. Advantages: Pipelining increases CPU throughput by overlapping the execution of instructions. It can improve performance by reducing the time taken to complete a single instruction. Hazards: Data hazards, control hazards, and structural hazards can occur in pipelined architectures, potentially reducing performance. Techniques like forwarding and branch prediction are used to mitigate these hazards.