IMAN ASHLY
Faculty of Computing
NSBM Green University

CS104.3 – Computer Architecture

Computer Systems Organization

22691
R B WICKRAMARATHNE

Computer Basics

Computer system = Hardware + Software

- Hardware: The physical parts of a computer system.
- Software: Instructions that tell the hardware what to do

Computer Systems Organization

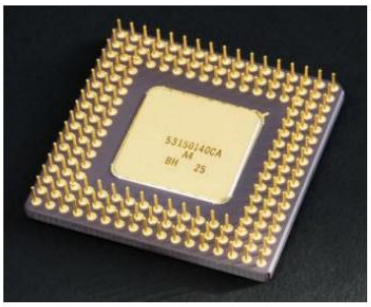
A digital computer consists of the following hardware components:

- Central Processing Unit (CPU)
- Main memory
- Auxiliary Storage
- I/O devices

Central Processing Unit

- The CPU is the “brain” of any computer system.
- Its main function is to execute instructions stored in the main memory by fetching the instructions, examining them, and executing them one after another.
- In addition, the CPU directs the flow of data in a computer and basically controls all operations that are performed on computers.

CPU



- Inside a processor, we can store zeros and ones using transistors.
- These are microscopic switches that control the flow of electricity depending on whether the switch is on or off.
- So, the transistor contains binary information: a one if a current passes through and a zero if a current does not pass through.
- Transistors are located on a very thin slice of silicon.
- A single silicon chip can contain thousands of transistors.
- A single CPU contains many chips.
- Combined, these only cover about a square inch or so.
- In a modern CPU, however, that square inch can hold several hundred million transistors - the very latest high-end CPUs have over one billion! Calculations are performed by signals turning on or off different combinations of transistors.
- And more transistors mean more calculations.

Memory

Computer memory (or storage) can be classified into two main divisions:

- Main memory
- Auxiliary storage (backing store)

Main memory (RAM)

Main memory is used to temporarily store:

- data for processing
- Instructions to process data
- Information (processed data) to be sent to an O/P device or to a secondary storage device.
- When the computer is turned off, any information stored will be erased from the main memory. Therefore, the main memory is called volatile memory.

Auxiliary Store (Backing store/Secondary storage)

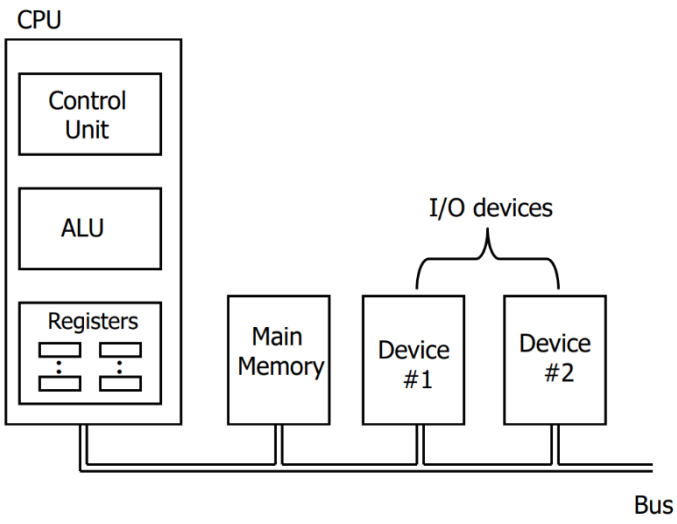
- Programs and data not immediately needed by the computer are placed in mass storage devices. They are much but also much than main memory.
- Secondary storage devices are used for permanent data storage and are usually found in the form of magnetic disks.

Note:

It is much faster for the CPU to access data stored in memory than data stored in auxiliary storage devices. Therefore, in order to work with data, the information on secondary storage is accessed indirectly by transferring the required information to the main memory. Once the CPU processes this data, the data and the results are filed away on auxiliary storage for more permanent safekeeping.

I/O Devices

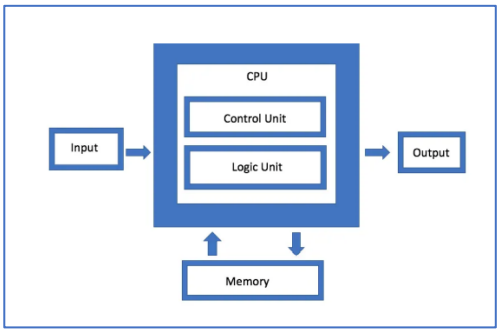
These devices provide the means of communication between the computer system and the outside world



CPU Organization

Von Neumann Architecture

- ✓ The von Neumann architecture is the basis of almost all computing done today.
- ✓ It assumes that every computation pulls data from memory, processes it, and then sends it back to memory.
- ✓ This has created what is known as the von Neumann bottleneck.

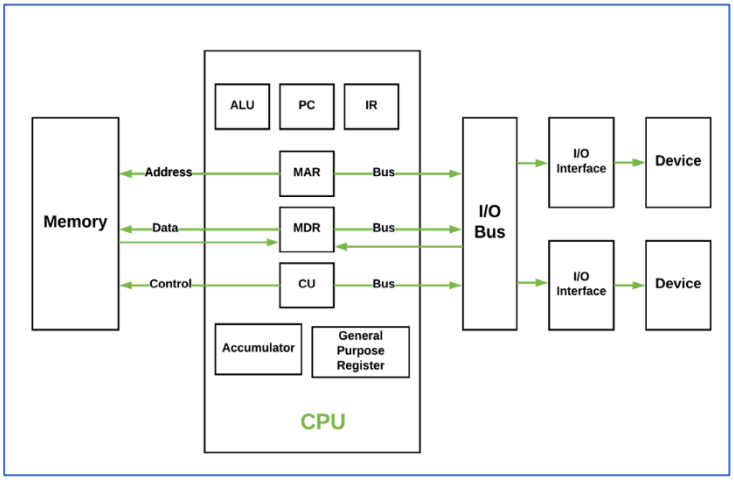


Neumann bottleneck

- In the von Neumann architecture, programs and data are held in memory; the processor and memory are separate, and data moves between the two. In that configuration, latency is unavoidable.
- In recent years, processor speeds have increased significantly.
- Memory improvements have mostly been in density (the ability to store more data in less space) rather than transfer rates.
- As speeds have increased, the processor has spent an increasing amount of time idle, waiting for data to be fetched from memory.
- No matter how fast a given processor can work, in effect it is limited to the rate of transfer allowed by the bottleneck. Often, a faster processor just means that it will spend more time idle.
- The von Neumann bottleneck, which the processor being idle for a certain amount of time while memory is accessed, has often been considered a problem that can only be overcome through significant changes to computer or processor architectures.

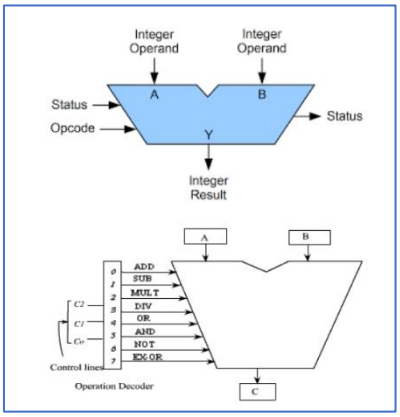
Approaches to overcoming the von Neumann bottleneck:

- Caching - the storage of frequently used data in a special area (usually RAM), so that it is more readily accessible than if it were stored in main memory.
- Prefetching - moving some data into cache before it is requested to speed access in the event of a request.
- Multithreading - managing multiple requests simultaneously in separate threads.
- New types of RAM (random access memory) - for example, DDR SDRAM, which activates output on both the rising and falling edge of the system clock rather than on just the rising edge, to potentially double output.
- RAMBUS - a memory subsystem consisting of the RAM, the RAM controller, and the bus (path) connecting RAM to the microprocessor and devices in the computer that use it.
- Processing in memory (PIM), which integrates a processor and memory in a single microchip.



The CPU is composed of several parts:

- Arithmetic Logic Unit
- Control Unit
- Registers



Arithmetic Logic Unit (ALU)

- The ALU performs arithmetic and logical operations (comparison) in a computer.
- Arithmetic operations include addition, subtraction, multiplication and division.
- Logic operations make a comparison and act based on the result.
- Example: Two numbers can be compared to determine if they are equal.

Control Unit

- This is the part of the CPU that controls & coordinates the activities taking place in the CPU, memory and the peripherals by sending control signals to the various devices.

Registers

- These are special fast storage areas available within the CPU for temporary storage of instructions and data during processing.
- The number of bits that can be stored in a (typical) register is usually referred to as the word size of that computer.
- The word size is one of the factors that determines the speed of a computer because, larger the word size, the more information a CPU can process at a time. Generally the word size can be 8, 16, 32 or 64 bits (or more).

Registers inside a CPU

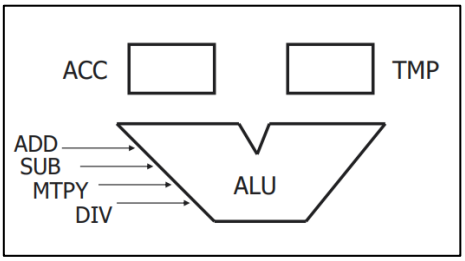
Some of the special purpose registers of a CPU can be identified as follows:

- ✓ Program Counter (PC) --It is also called the Instruction Pointer (IP) and contains the memory address of the next instruction to be executed.
- ✓ Instruction Register (IR) -- This register contains the current instruction (both the operator and the operand).
- ✓ Memory Address Register (MAR) -- Holds the address of the memory location from which information will be read from or to which data will be written to.
- ✓ Memory Data register (MDR) -- Used to temporary store information read from or written to memory.

- In addition, there are several general-purpose registers used for performing arithmetic functions.
- One of these registers called the accumulator which is a type of register included in a CPU.
- It acts as a temporary storage location which holds an intermediate value in mathematical and logical calculations.
- (In modern CPUs, accumulators are replaced by general-purpose registers because they offer more flexibility. However, accumulators may still be in some special-purpose processors.)

Example 1: You have written a program to multiply two numbers A and B. If A =5 and B =10 explain how the ALU will perform this operation.

Example 2: You have written a program to add three numbers A, B and C. If A =10, B =5 and C=100 explain how the ALU will perform this operation.



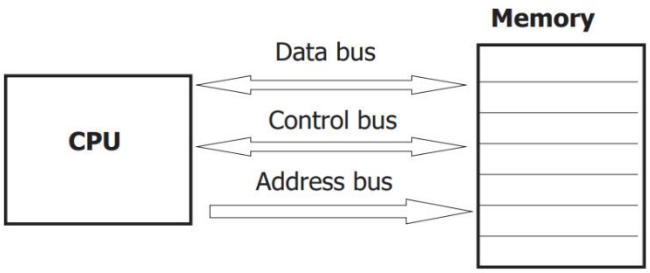
Bus organization

The main components of a computer system (CPU, main memory, I/O devices) are connected electronically using a set of wires which facilitate the communication between these units. These wires are referred to as a bus.

A bus may be unidirectional (capable of transmitting data in one direction) or it may be bidirectional (transmitting data in both directions)

Buses can be categorized as:

- Data bus
- Address bus
- Control bus



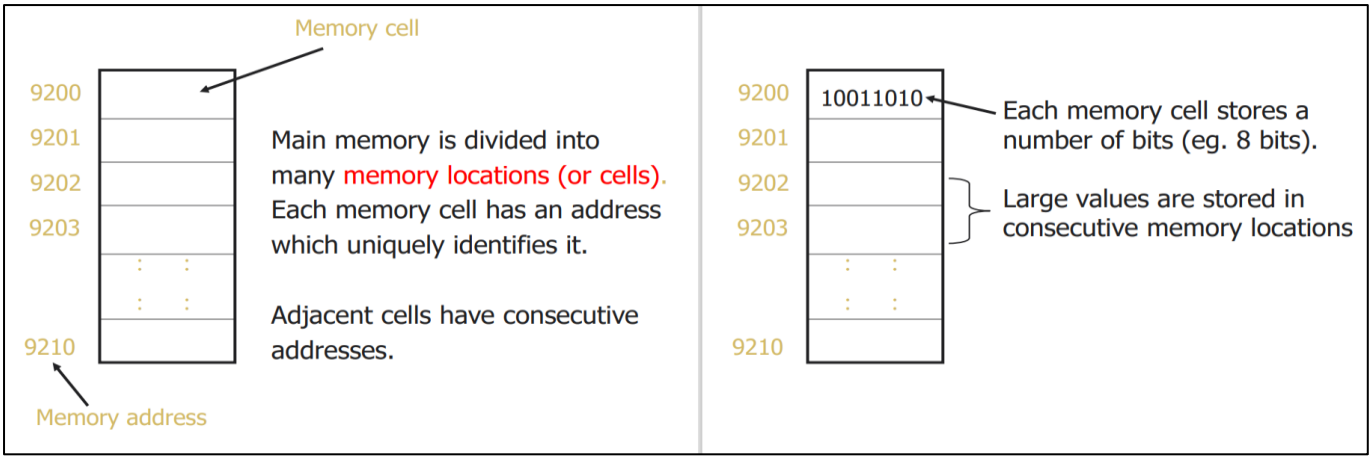
The data bus transfers data whereas the address bus transfers information about where the data should go.

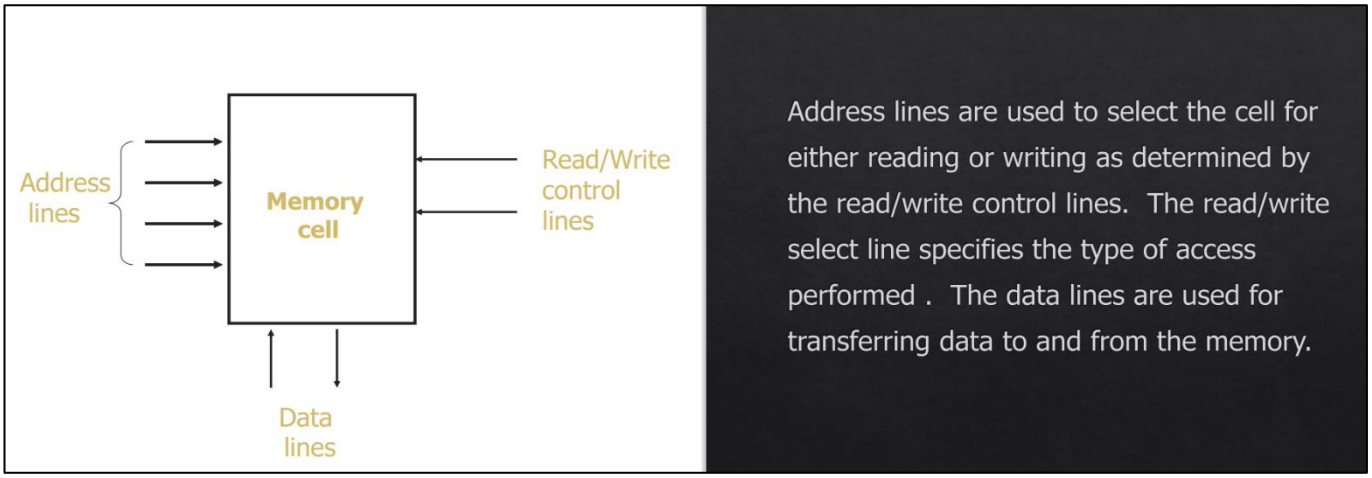
The control bus carries commands from the CPU (control information) and returns status signals from the devices.

The size of the data bus (bus width) is important because it determines how much data can be transmitted at a time. A 16-bit bus for example has 16 lines and can transmit 16 bits of data simultaneously. The larger the number of bits that can be transferred by the bus , the faster the computer can transfer data.

Processor	Data bus width
8088	8
80286	16
80386	32
Pentium IV	64

Memory addresses





The number of lines in the address bus determines how many locations can be accessed. With a 16-bit address, the highest location that can be addressed is 65,535 (2^{16})

Processor	Address bus width
8088	20
80286	24
80386	32
Pentium IV	32

Executing instructions

The CPU executes each instruction in a series of small steps. This sequence of steps is referred to as the **fetch-decode-execute** cycle or **machine cycle**.

Four operations of the machine cycle

Fetch- fetch the next program instruction from memory.

Decode- decode the instruction stored in the IR.

Execute- process the command.

Store – write the results of the instruction into main memory.

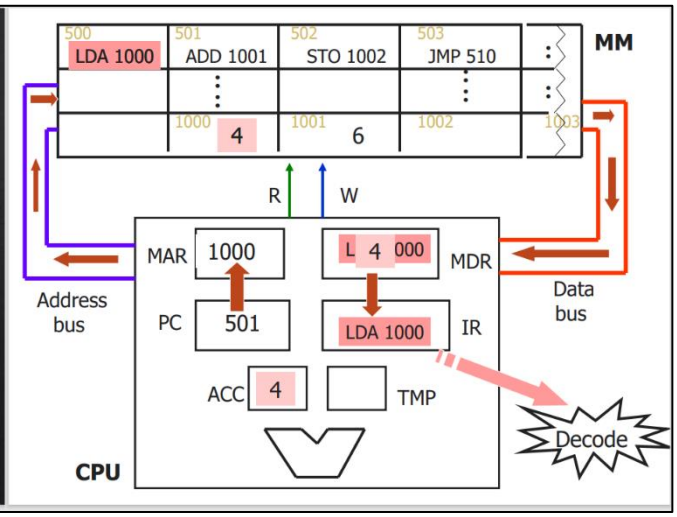
◈ Example 1:

Describe the sequence of events carried out during the machine cycle when executing the following instructions.

Address	Contents
500	LDA 1000
501	ADD 1001
502	STO 1002
503	JMP 510
1000	4
1001	6
1003	

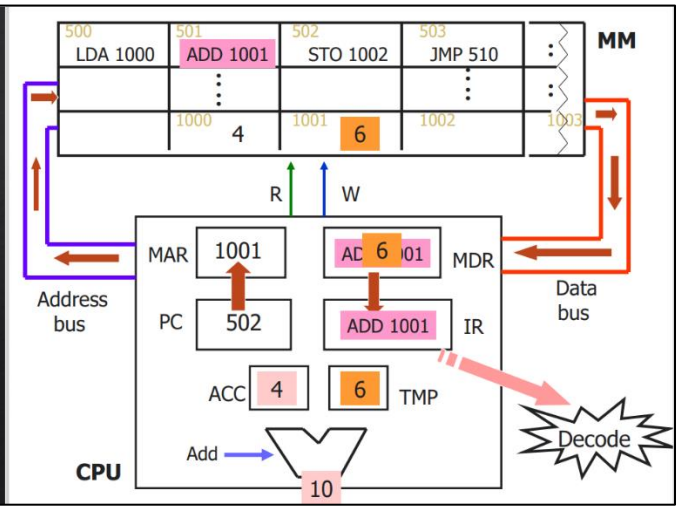
◈ Instruction #1

LDA 1000 – Load to the accumulator the contents of the memory location 1000.



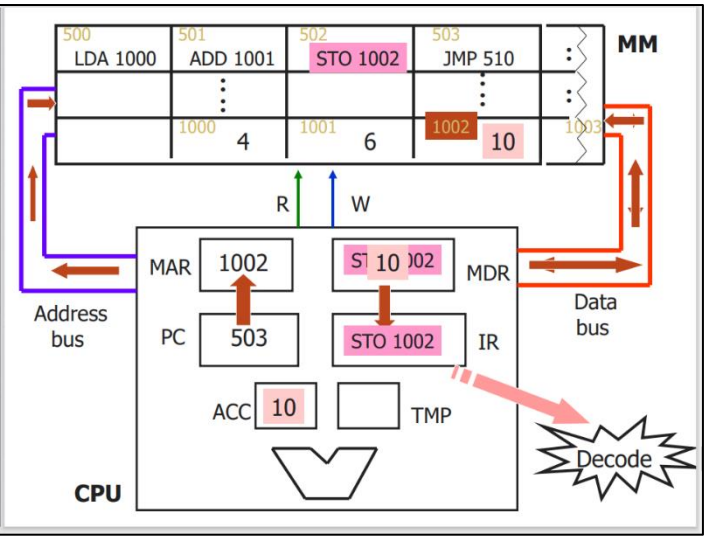
◈ Instruction #2

ADD 1001 – add the contents of location 1001 and the contents of the accumulator and store the result back in the accumulator.



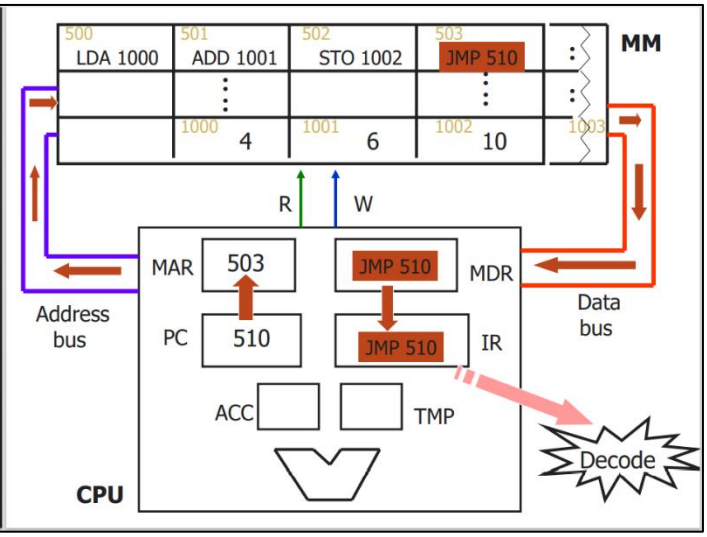
◈ Instruction #3

STO 1002 – store the contents of the accumulator to the memory location 1002.



- Instruction #4

JMP 510 – Jump to memory location 510



◈ Example 2:

Describe the sequence of events carried out during the machine cycle when executing the following instructions.

Address	Contents	
100	JMP 200	
200	MOV R1 R2	Move the contents of register R2 to R1.
201	STO 800 R1	Store the contents of R1 in memory location 800.

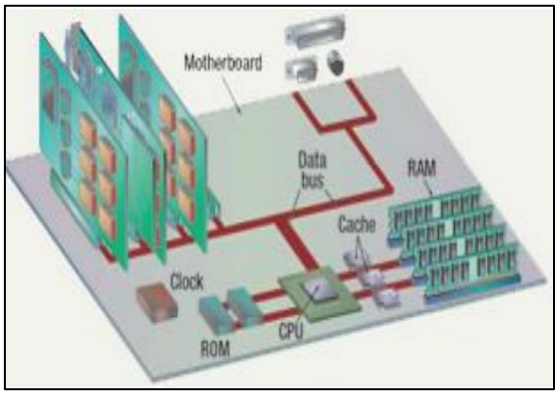
Example 3:

Address	Contents	Comments
100	JMP 200	
200	LDA 1000	
201	MPY 1001	Multiply the contents of the Accumulator with the contents of the memory location 1001 and store the result back in the Accumulator.
202	STO 1002	
1000	5	
1001	10	

Address	Contents	Comments
299	LDA 1000	
300	JMP 310	
310	MPY 1001	Multiply the contents of the Accumulator with the contents of the memory location 1001 and store the result back in the Accumulator.
311	STO 1002	
1000	15	
1001	20	

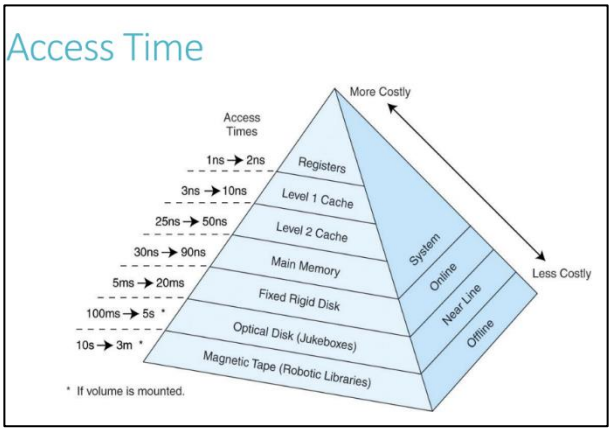
Components affecting Speed

- CPU
- Memory Registers
- Clock speed
- Cache memory
- Data bus



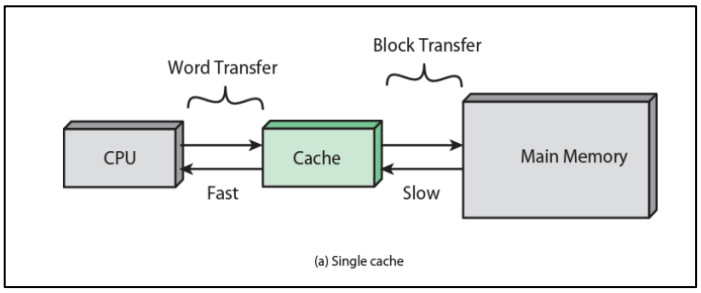
Achieving Increased Processor Speed

- Increase the hardware speed of the processor- shrinking the size of the logic gates on the processor chip, so that more gates can be packed together more tightly
- Increasing the clock rate- individual operations are executed more rapidly.
- Increase the size and speed of caches - By dedicating a portion of the processor chip, itself to the cache, cache access times drop significantly.
- Make changes to the processor organization and architecture - that increase the effective speed of instruction execution. Typically, this involves using parallelism in one form or another.

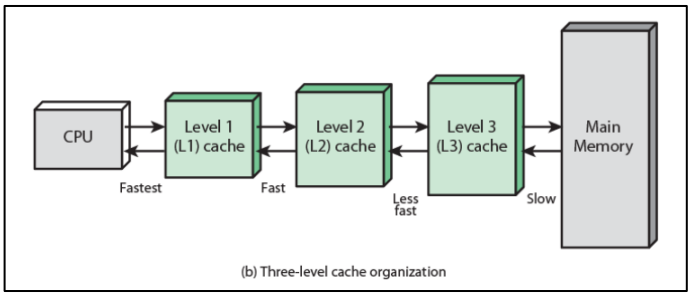


Intel Cache Evolution

Problem	Solution	Processor on which feature first appears
External memory slower than the system bus.	Add external cache using faster memory technology.	386
Increased processor speed results in external bus becoming a bottleneck for cache access.	Move external cache on-chip, operating at the same speed as the processor.	486
Internal cache is rather small, due to limited space on chip	Add external L2 cache using faster technology than main memory	486
Contention occurs when both the Instruction Prefetcher and the Execution Unit simultaneously require access to the cache. In that case, Prefetcher is stalled while the Execution Unit's data access takes place.	Create separate data and instruction caches.	Pentium
Increased processor speed results in external bus becoming a bottleneck for L2 cache access.	Create separate back-side bus that runs at higher speed than the main (front-side) external bus. The BSB is dedicated to the L2 cache.	Pentium Pro
	Move L2 cache on to the processor chip.	Pentium II
Some applications deal with massive databases and must have rapid access to large amounts of data. The on-chip caches are too small.	Add external L3 cache.	Pentium III
	Move L3 cache on-chip.	Pentium 4



(a) Single cache



(b) Three-level cache organization