Al-Azhar University
Faculty of Eng.,
Systems and Computer Eng., Dept.,

SCE 409: Computer Architecture

Grade 4, 1st Semister

Ass. Prof. Khaled Elshafey

Course Objectives

OBJECTIVES:

This course aims to provides a strong foundation to understand modern computer system architecture.

Text Books

- "Computer Architecture, A Quantitative Approach", by: **John L. Hennessy, and David A. Patterson, 5th Edition.**

- **"Logic and Computer Design Fundamentals", by:** M. Morris Mano Charles Kime, Fourth Edition

Syllabus

- Introduction
- Quantitative principles of computer design
- Memory and PLDs (FPGAs)
- Register Transfers and Datapaths
- Single-cycle hardwired and multiple-cycle microprogramming control
- I/O organization
- Parallel computer Architecture: Pipeline and Superscalar techniques.
- High-Performance Computing (HPC)

Course Plan

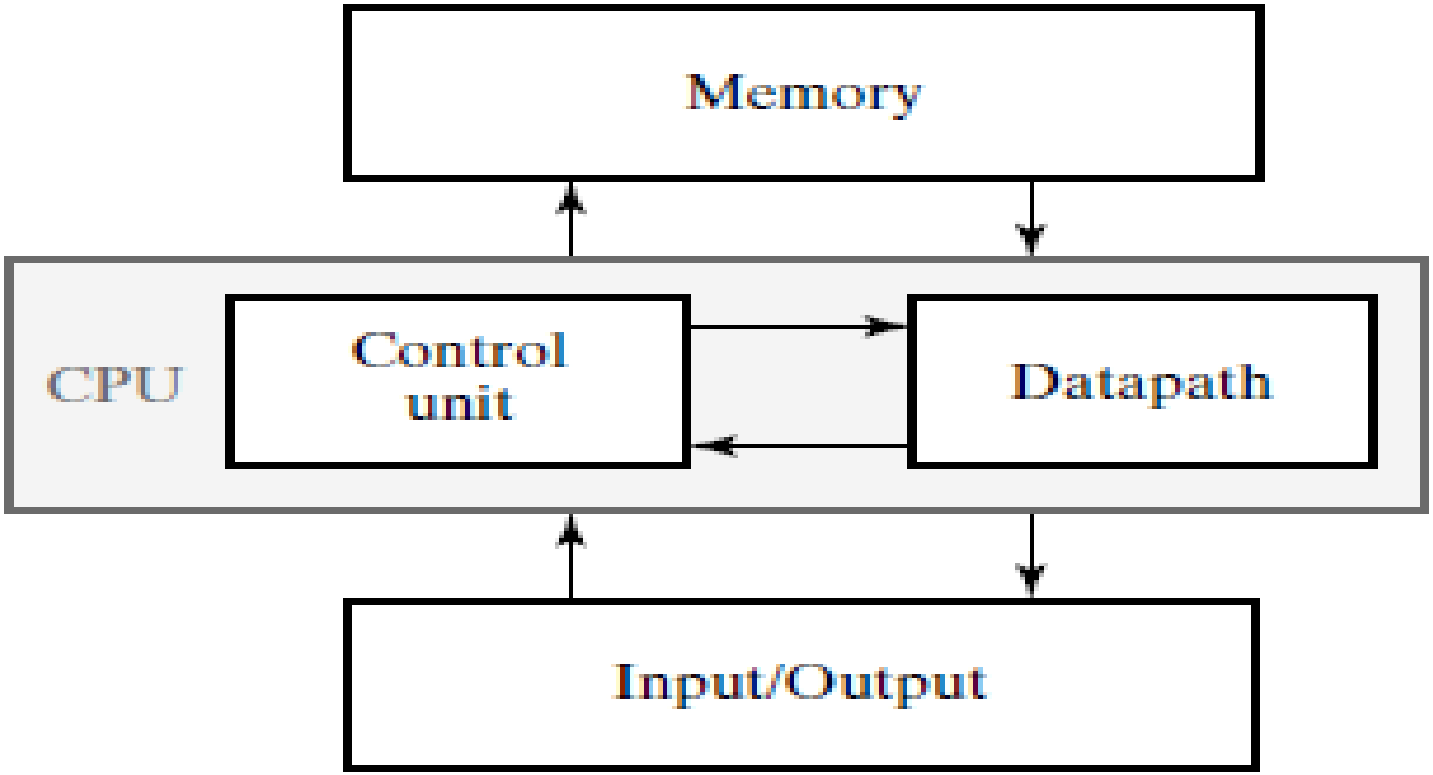
- Final Exam: 100
- Mid Term : 20
- Course Project: 20
- Quizzes& Assignments: 10

- Total 150

Basic Terms

- **Computer Architecture** includes the study of functional blocks that make up the computer system and the way they are connected.
- **Computer Organization**: concerned with the implementation of computer architecture, and involves # processors, memory size, time to execute an instruction,….etc.
- **Computer engineering**: referred to the actual construction of a computer.

Von-Neumann Architecture

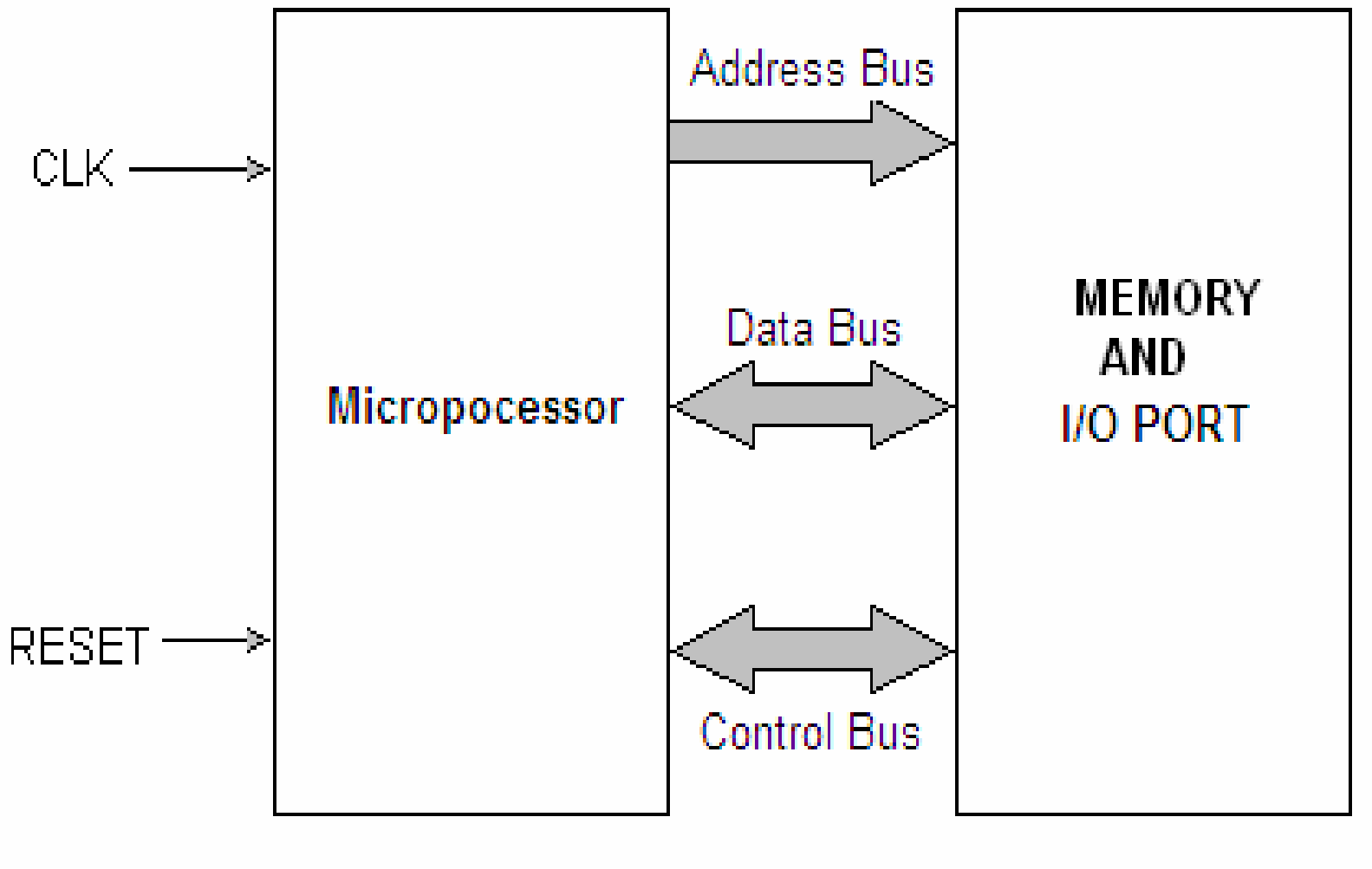


The main Features of Von-Neumann Arch.,

- The computer consists of memory, I/O, and CPU
- The computer structure is independent of the problem
- Binary signals are used for representing data and instructions
- The memory is divided into cells of equal size
- Instructions and Data are stored in the same memory
- The program is a sequence of instructions (control flow)

BASIC MICROPROCESSOR SYSTEM

- The Microprocessor alone does not serve any useful purpose unless it is supported by memory and I/O ports.
- The combination of memory and I/O ports with microprocessor is known as microprocessor-based system.
- The microprocessor executes the program stored in the memory and transfer data to and from the outside world through I/O ports.
- The microprocessor is interconnected with memory and I/O ports by busses called: the data bus, the Address bus and the control bus.
- A bus is basically a communication link between the processing unit and the peripheral devices.


CLK
RESET
Micropocessor
Address Bus
Data Bus
Control Bus
MEMORY
AND
I/O PORT

Control Unit

- The control unit performs the most important function in a computer.
- It controls all other units and controls the flow of data from one unit to another for performing computations.
- It also sequences the operations.
- It instructs all the units to perform the task in a particular sequence with the help of clock pulses.

ALU

- Microprocessors (Datapath) are defined by their registers and the operations performed on binary data stored in the registers.
- This operation unit (ALU) is used for performing arithmetic operations such as Addition, Subtraction, Multiplications, division and other logical operations on the data.
- The control unit guides ALU which of the operations are to be performed.
- The sequence of the instructions is controlled by the control unit.

Address Bus

- The address bus is unidirectional and is to be used by the CPU to send out address of the memory location to be accessed.
- It is also used by the CPU to select a particular input or output port.
- It may consist of 8, 12, 16, 20 or even more number of parallel lines.
- Number of bits in the address bus determines the minimum number of bytes of data in the memory that can be accessed.
- A 16-bit address bus for instance can access 2^{16} bytes of data.

Data Bus

- Data bus is bidirectional, that is, data flow occurs both to and from CPU and peripherals.
- A microprocessor is characterized by the width of its data bus.
- The size of the internal data bus determines the largest number that can be processed by a microprocessor, for instance, having a 16-bit internal data bus is 65536 (64K).
- A microprocessor is specified by its ‘Word Size’, e.g. 4-bit, 8-bit, 16-bit etc.
- By the term ‘word size” means the number of bits of data that is processed by the microprocessor as a unit.
- It also specifies the width of the data bus.

Control Bus

- Control bus contains a number of individual lines carrying synchronizing signals.
- The control bus sends out control signal to memory, I/O ports and other peripheral devices to ensure proper operation.
- For instance, if it is desired to read the contents of a particular memory location, the CPU first sends out address of that very location on the address bus and a ‘Memory Read’ control signal on the control bus.
- The memory responds by outputting data stored in the addressed memory location on the data bus.

Summary

What are the three main units of a digital computer?

- **Ans.** The three main units of a digital computer are: the central processing unit (CPU), the
- memory unit and the input/output devices.

How does the microprocessor communicate with the memory and input/output devices?

- **Ans.** The microprocessor communicates with the memory and the Input/Output devices via
- the three buses, data bus, address bus and control bus.

What are the different jobs that the CPU is expected to do at any given point of time?

- **Ans.** The CPU may perform a memory read or write operation, ALU operations, an I/O read or write operation or an internal activity.

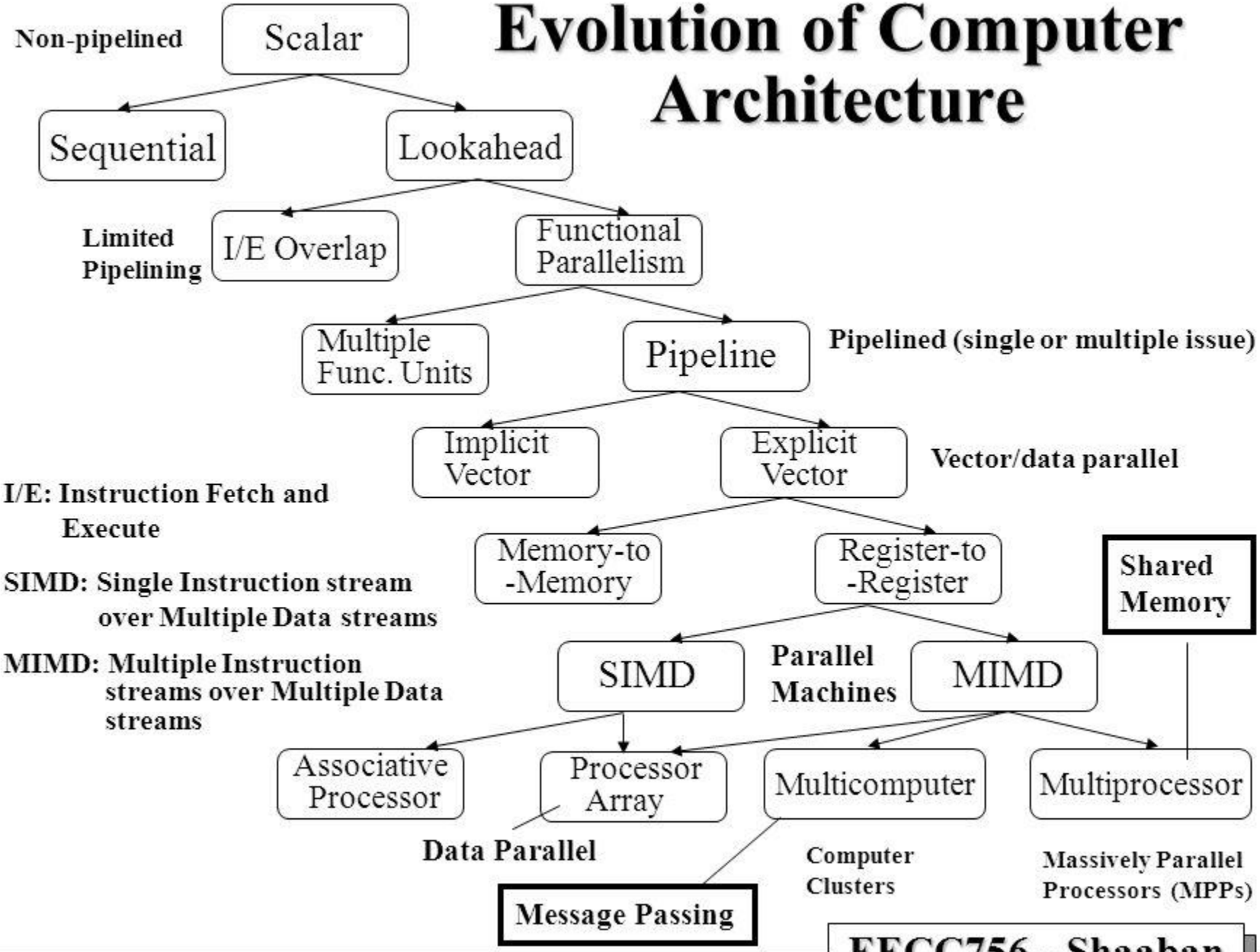
Assessment #1

Give a brief about :

A- “HARVARD Computer Architecture”.

B- Data Flow computers

Evolution of Computer Architecture



The Task of computer Designer

- Determine what attributes are important for a new computer, then design a computer to maximize performance and energy efficiency while staying within cost, power, and availability constraints.
- This task has many aspects, including instruction set design, functional organization, logic design, and implementation.

Cont.,

- The implementation may encompass integrated circuit design, packaging, power, and cooling.
- Optimizing the design requires familiarity with a very wide range of technologies, from compilers and operating systems to logic design and packaging.

Beyond the Computer

- In terms of world impact, computers, such as the PC, are not the end of the story.
- Smaller, often less powerful, single-chip computers called *microcomputers* or *microcontrollers*, or special-purpose computers called *digital signal processors* (DSPs) actually are more prevalent in our lives.
- These computers are parts of everyday products (Embedded systems).

Classes of Parallelism and Parallel Architectures

- Parallelism at multiple levels is now the driving force of computer design across all four classes of computers, with energy and cost being the primary constraints.
- Parallelism can appear in different forms:

  Lookahead, Pipelining, Multitasking, Multiprogramming, Concurrency, multithreading, ….etc

Cont.,

- There are basically two kinds of parallelism in applications:

  1. *Data-Level Parallelism (DLP),* which arises because there are many data items that can be operated on at the same time.

  2. *Task-level parallelism (TLP)*, which arises because tasks of work are created that can operate independently and largely in parallel.

- We then explain the four architectural styles that exploit DLP and TLP

Michael Flynn's Classifications

He looked at the parallelism in the instruction and data streams, and placed all computers into one of four categories:

- *Single instruction stream, single data stream* (SISD)
- *Single instruction stream, multiple data streams* (SIMD)
- *Multiple instruction streams, single data stream* (MISD)
- *Multiple instruction streams, multiple data streams* (MIMD)

Quantitative Principles of Computer Design

- **Take Advantage of Parallelism** is one of the most important methods for improving performance.
- **Principle of Locality:** Programs tend to reuse data and instructions they have used recently. A widely held rule of thumb is that a program spends 90% of its execution time in only 10% of the code. An implication of locality is that we can predict with reasonable accuracy what instructions and data a program will use in the near future based on its accesses in the recent past.

Cont.,

- Two different types of locality have been observed.

a) Temporal locality states that recently accessed items are likely to be accessed in the near future.

b) Spatial locality says that items whose addresses are near one another tend to be referenced close together in time.

Cont.,

- **Focus on the Common (frequent) Case**
  Perhaps the most important and pervasive principle of computer design is to focus on the common case: In making a design trade-off, favor the frequent case over the infrequent case.
- How much performance can be improved by making that frequent case faster.

Amdahl's Law

- The performance gain that can be obtained by improving some portion of a computer can be calculated using Amdahl's law.
- Amdahl's law states that the performance improvement to be gained from using some faster mode of execution is limited by the fraction of the time the faster mode can be used.

What is speedup?

- Suppose that we can make an enhancement to a computer that will improve performance when it is used.
- Speedup is the ratio:
- Speedup = (Execution time for entire task without using enhancement) / (Execution time for the task with using enhancement).

Cont.,

Amdahl's law gives us a quick way to find the speedup from some enhancement, which depends on two factors:

1.The fraction enhancement:

is the fraction of the computation time in the original computer that can be converted to take advantage of the enhancement—

For example,

if 20 seconds of the execution time of a program that takes 60 seconds in total can use an enhancement, the fraction is 20/60.

Cont.,

2- Speedup enhancement:

The improvement gained by the enhanced execution mode, that is, how much faster the task would run if the enhanced mode were used for the entire program.

For example:

- If the enhanced mode takes, say, 2 seconds for a portion of the program, while it is 5 seconds in the original mode, the improvement is 5/2.

Speedup Overall

- The overall speedup is the ratio of the execution times:
- Speedup overall = Execution time old / Execution time new

The overall speedup is the ratio of the execution times:

$$\text{Speedup}_{\text{overall}} = \frac{\text{Execution time}_{\text{old}}}{\text{Execution time}_{\text{new}}} = \frac{1}{(1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}}$$

Example Suppose that we want to enhance the processor used for Web serving. The new processor is 10 times faster on computation in the Web serving application than the original processor. Assuming that the original processor is busy with computation 40% of the time and is waiting for I/O 60% of the time, what is the overall speedup gained by incorporating the enhancement?

Answer $\text{Fraction}_{\text{enhanced}} = 0.4$; $\text{Speedup}_{\text{enhanced}} = 10$; $\text{Speedup}_{\text{overall}} = \dfrac{1}{0.6 + \dfrac{0.4}{10}} = \dfrac{1}{0.64} \approx 1.56$

Processor Performance Equation

- CPU time = CPU clock cycles for a program × Clock cycle time
- CPU time = Instruction count × Cycles per instruction × Clock cycle time.

Where:

Instruction count (IC): the number of instructions executed

Cycles per instruction (CPI): the average number of *clock cycles per instruction.*

Clock cycle time: Hardware technology and organization

Cont.,

- As this formula demonstrates, processor performance is dependent upon three characteristics:

  clock cycle (or rate), clock cycles per instruction, and instruction count.

- Furthermore, CPU time is *equally* dependent on these three characteristics; for example, a 10% improvement in any one of them leads to a 10% improvement in CPU time.

Cont.,

- Unfortunately, it is difficult to change one parameter in complete isolation from others because the basic technologies involved in changing each characteristic are interdependent:
- ■ *Clock cycle time*—Hardware technology and organization
- ■ *CPI*—Organization and instruction set architecture
- ■ *Instruction count*—Instruction set architecture and compiler technology

Benchmarks

- **Natural**: real programs used to solve a real task

- **Synthetic**: Sequence of instructions constructed for the purpose of measuring performance.

Dependability

- Historically, integrated circuits were one of the most reliable components of a computer.
- Although their pins may be vulnerable, and faults may occur over communication channels, the error rate inside the chip was very low (such as stuck at zero , stuck at one, or bridging faults).
- Systems alternate between two states of service:

    1. *Service accomplishment*, where the service is delivered as specified

    2. *Service interruption*, where the delivered service has error.

- Quantifying these transitions leads to the two main measures of dependability:

1- Module Reliability

- The mean time to failure (MTTF) is a reliability measure.
- The reciprocal of MTTF is a rate of failures λ
- Service interruption is measured as mean time to repair (MTTR).
- Mean time between failures (MTBF) is simply the sum of MTTF + MTTR.
- Although MTBF is widely used, MTTF is often the more appropriate term.

2- Module Availability

- For nonredundant systems with repair, module availability is:

$$\text{Module availability} = \frac{\text{MTTF}}{(\text{MTTF} + \text{MTTR})}$$

Sheet 1

- Examples at pages: 34 , 47, and 50 from Hennessy ‘s book.