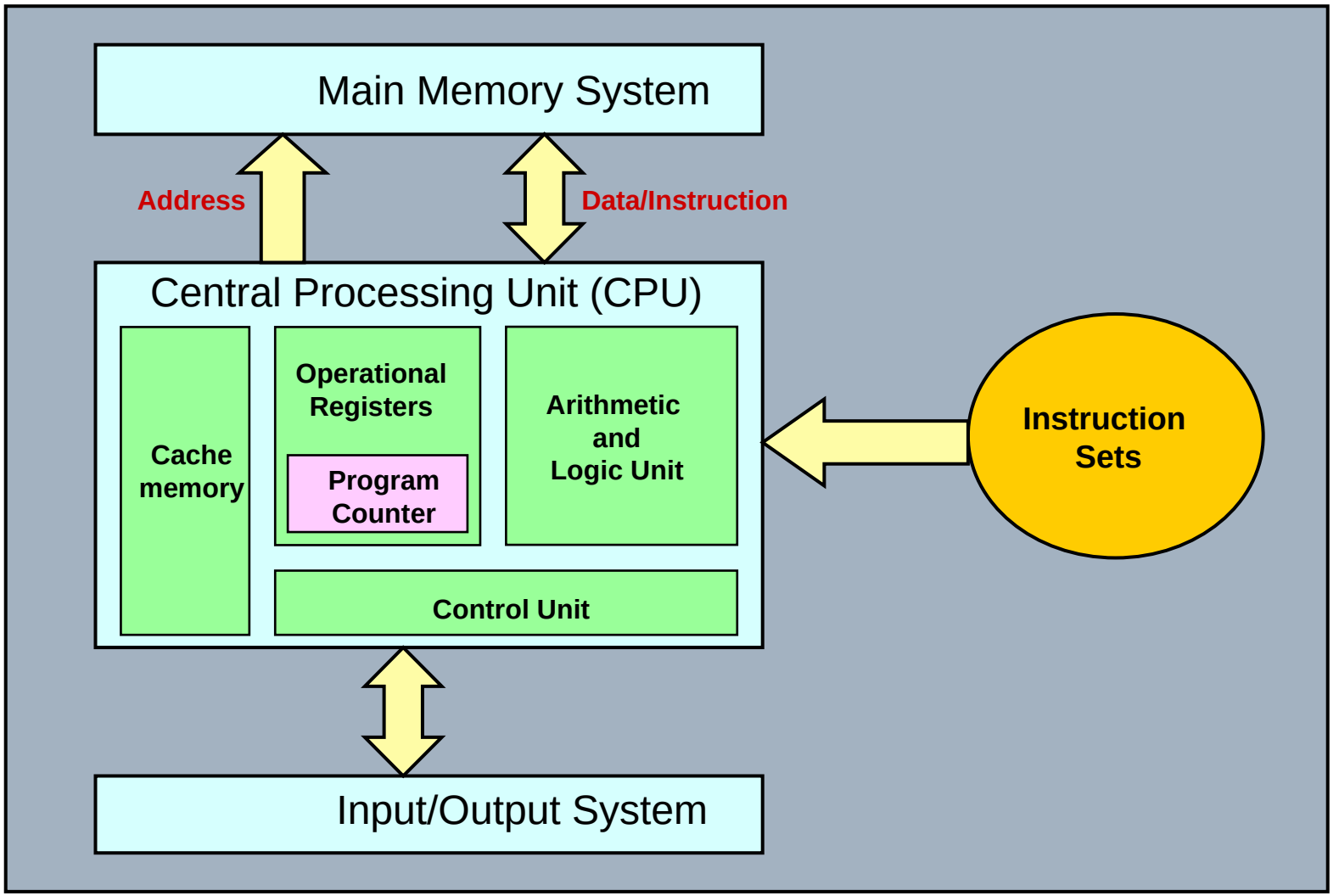
Chapter 2
Basic Structure of Computers

Jin-Fu Li
Department of Electrical Engineering
National Central University
Jungli, Taiwan

Outline

- Functional Units
- Basic Operational Concepts
- Bus Structures
- Software
- Performance

Content Coverage



Functional Units

- A computer consists of three main parts:
  - A processor (CPU)
  - A main-memory system
  - An I/O system
- The CPU consists of a control unit, registers, the arithmetic and logic unit, the instruction execution unit, and the interconnections among these components
- The information handled by a computer
  - Instruction
    - Govern the transfer information within a computer as well as between the computer and its I/O devices
    - Specify the arithmetic and logic operations to be performed
  - Data
    - Numbers and encoded characters that are used as operands by the instructions

Program

- A list of instructions that performs a task is called a program
- The program usually is stored in a memory called program memory
- The computer is completely controlled by the stored program, except for possible external interruption by an operator or by I/O devices connected to the machine
- Information handled by a computer must be encoded in a suitable format. Most present-day hardware employs digital circuits that have only two stable states, 0 (OFF) and 1 (ON)

Memory Unit

- Memory
  - The storage area in which programs are kept when they are running and that contains the data needed by the running programs
- Types of memory
  - Volatile memory: storage that retains data only if it is receiving power, such as dynamic random access memory (DRAM)
  - Nonvolatile memory: a form of memory that retains data even in the absence of a power source and that is used to store programs between runs, such as flash memory
- Usually, a computer has two classes of storage
  - Primary memory and secondary memory
- Primary memory
  - Also called main memory. Volatile memory used to hold programs while they are running; typically consists of DRAM in today's computers

Memory Unit

- Secondary memory
  - Nonvolatile memory used to store programs and data between runs; typically consists of magnetic disks in today's computers
- The memory consists of storage cells, each capable of storing one bit of information
  - The storage cells are processed in groups of fixed size called words
  - To provide easy access to any word in the memory, a distinct address is associated with each word location
- The number of bits in each word is often referred to as the word length of the computer
  - Typical word length from 16 to 64 bits
- The capacity of the memory is one factor that characterizes the size of a computer

Memory Unit

- Instruction and data can be written into the memory or read out under the control of the processor
  - It is essential to be able to access any word location in the memory as quickly as possible
  - Memory in which any location can be reached in a short and fixed amount of time after specifying its address called random-access memory (RAM)
- The time required to access one word is called the memory access time
  - This time is fixed, independent of the location of the word being accessed
- The memory of a computer is normally implemented as a memory hierarchy of three or four levels
  - The small, fast, RAM units are called caches
  - The largest and slowest unit is referred to as the main memory

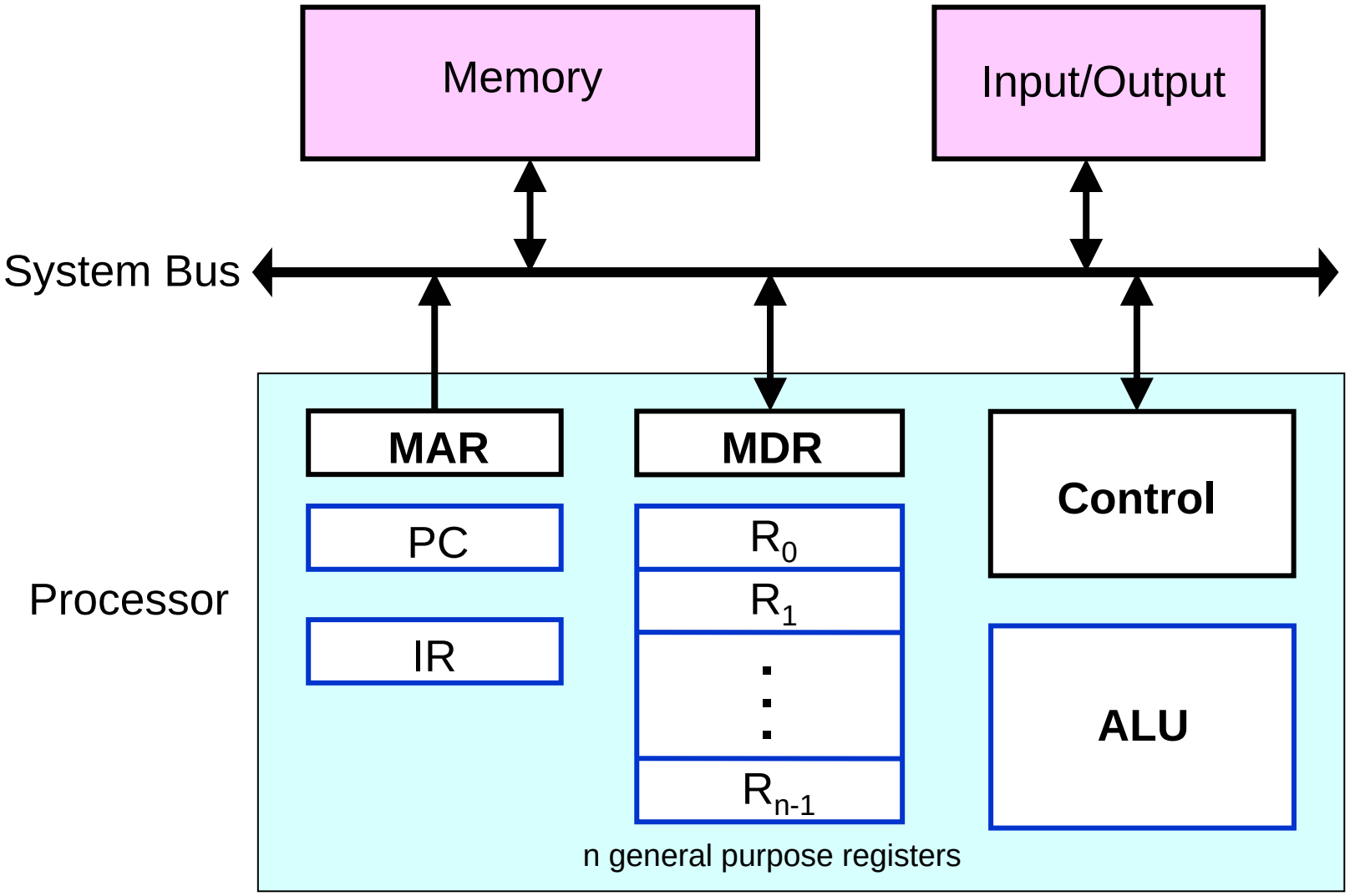
Arithmetic and Logic Unit

- Most computer operations are performed in the arithmetic and logic unit (ALU) of the processor
- For example, consider two numbers stored in the memory are to be added
  - They are brought into the processor, and the actual addition is carried out by the ALU. Then sum may be stored in the memory or retained in the processor for immediate use
- Typical arithmetic and logic operation
  - Addition, subtraction, multiplication, division, comparison, complement, etc.
- When operands are brought into the processor, they are stored in high-speed storage elements called registers.
  - Each register can store one word of data

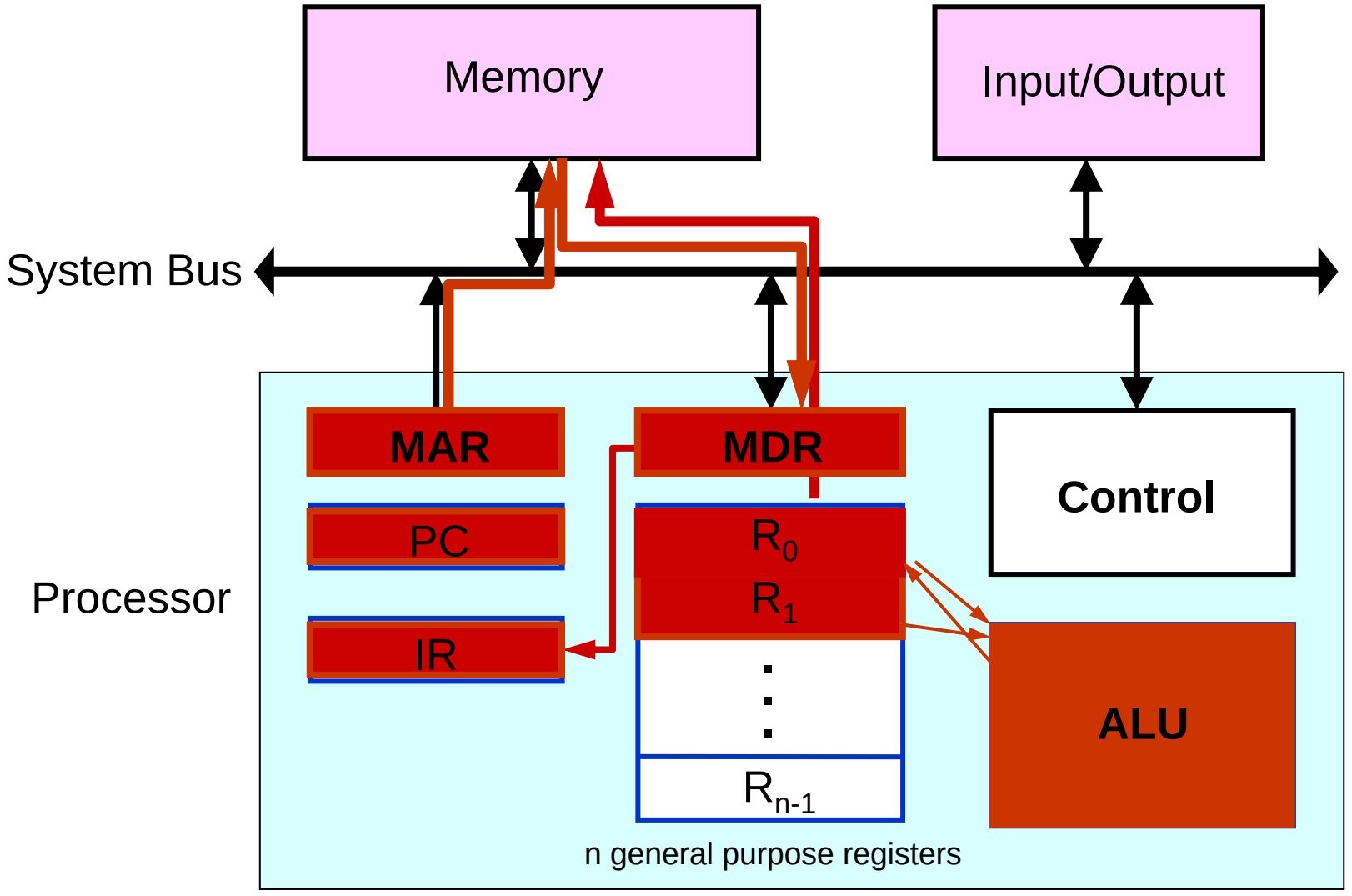
Control Unit

- The control unit is the nerve center that sends control signals to other units and senses their states
  - Thus the control unit serves as a coordinator of the memory, arithmetic and logic, and input/output units
- The operation of a computer can be summarized as follows:
  - The computer accepts information in the form of programs and data through an input unit and stores it in the memory
  - Information stored in the memory is fetched, under program control, into an ALU, where it is processed
  - Processed information leaves the computer through an output unit
  - All activities inside the machine are directed by the control unit

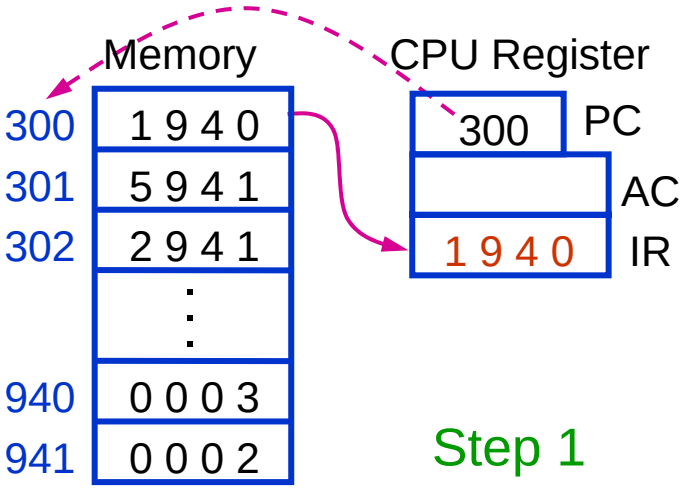
Computer Components: Top-Level View

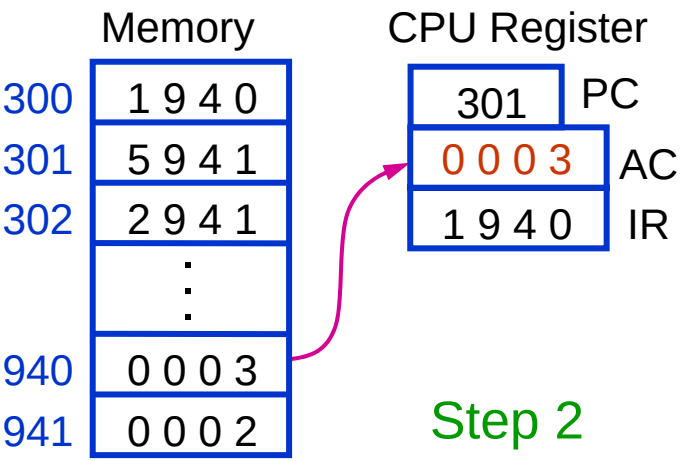


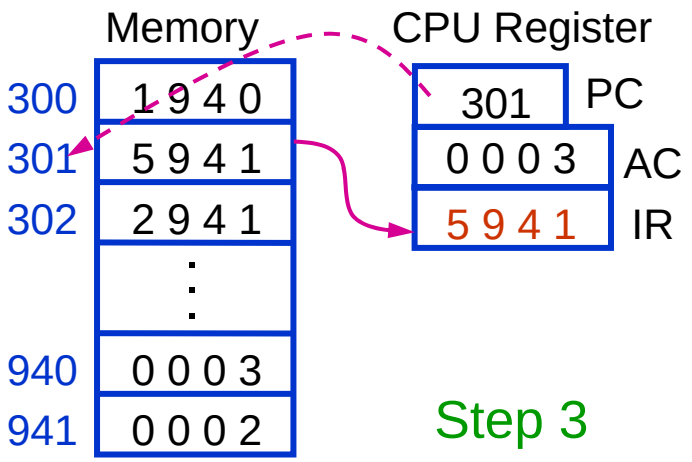
Basic Operational Concepts

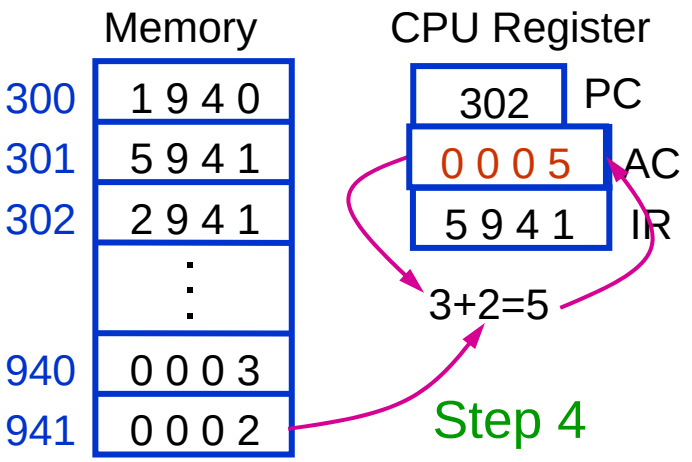


A Partial Program Execution Example

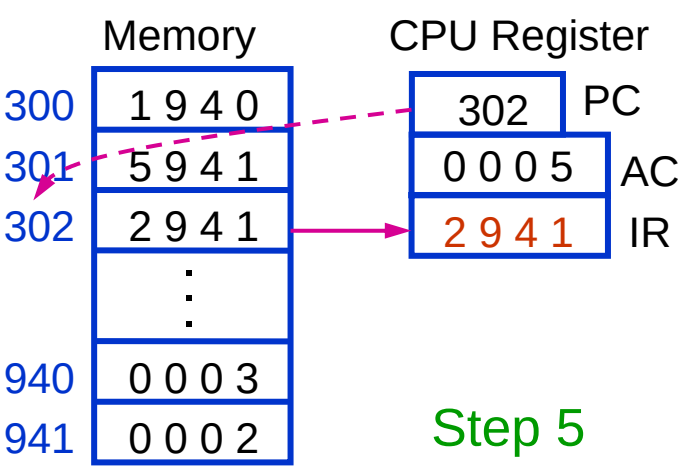


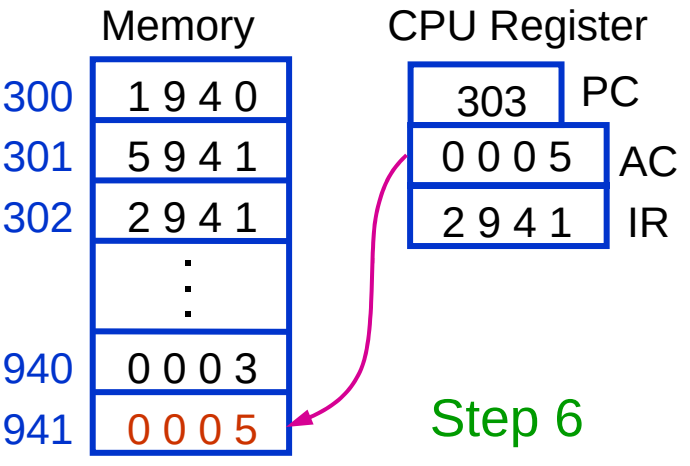






A Partial Program Execution Example





Interrupt

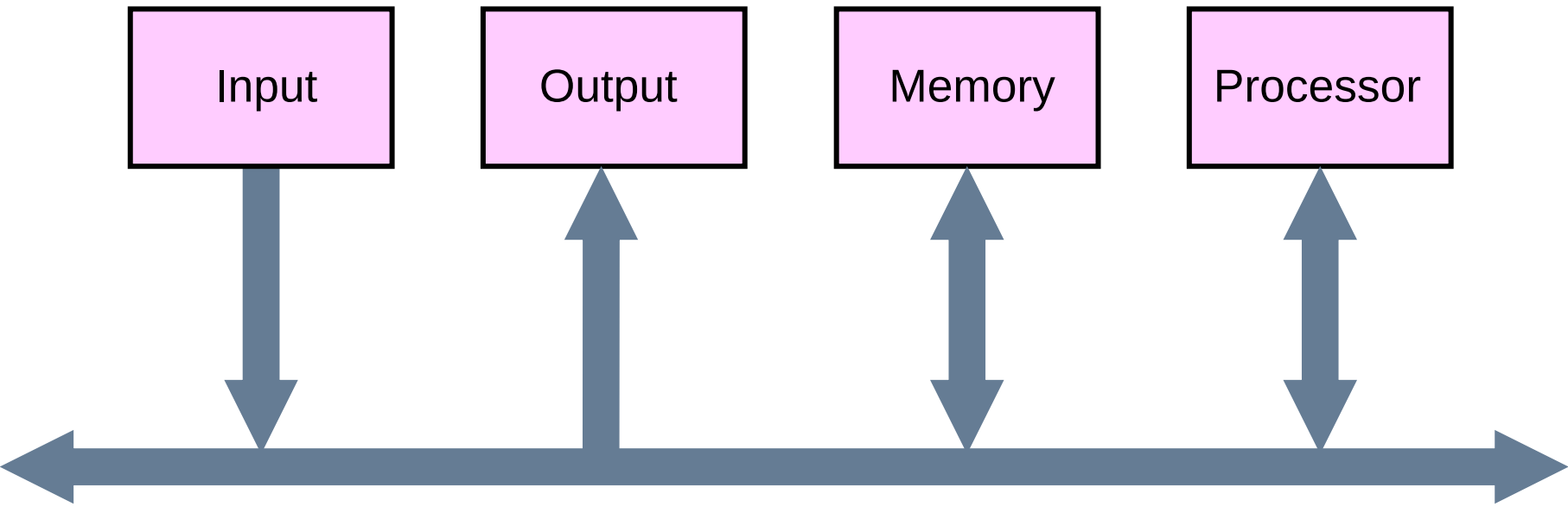
- Normal execution of programs may be preempted if some device requires urgent servicing
- To deal with the situation immediately, the normal execution of the current program must be interrupted
- Procedure of interrupt operation
  - The device raises an interrupt signal
  - The processor provides the requested service by executing an appropriate interrupt-service routine
  - The state of the processor is first saved before servicing the interrupt
    - Normally, the contents of the PC, the general registers, and some control information are stored in memory
  - When the interrupt-service routine is completed, the state of the processor is restored so that the interrupted program may continue

Classes of Interrupts

- Program
  - Generated by some condition that occurs as a result of an instruction execution such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, or reference outside a user's allowed memory space
- Timer
  - Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis
- I/O
  - Generated by an I/O controller, to signal normal completion of an operation or to signal a variety of error conditions
- Hardware failure
  - Generated by a failure such as power failure or memory parity error

Bus Structures

- A group of lines that serves a connecting path for several devices is called a bus
  - In addition to the lines that carry the data, the bus must have lines for address and control purposes
  - The simplest way to interconnect functional units is to use a single bus, as shown below



Drawbacks of the Single Bus Structure

- The devices connected to a bus vary widely in their speed of operation
  - Some devices are relatively slow, such as printer and keyboard
  - Some devices are considerably fast, such as optical disks
  - Memory and processor units operate are the fastest parts of a computer
- Efficient transfer mechanism thus is needed to cope with this problem
  - A common approach is to include buffer registers with the devices to hold the information during transfers
  - An another approach is to use two-bus structure and an additional transfer mechanism
    - A high-performance bus, a low-performance, and a bridge for transferring the data between the two buses. ARMA Bus belongs to this structure

Software

- In order for a user to enter and run an application program, the computer must already contain some system software in its memory
- System software is a collection of programs that are executed as needed to perform functions such as
  - Receiving and interpreting user commands
  - Running standard application programs such as word processors, or games
  - Managing the storage and retrieval of files in secondary storage devices
  - Running standard application programs such as word processors, etc
  - Controlling I/O units to receive input information and produce output results

Software

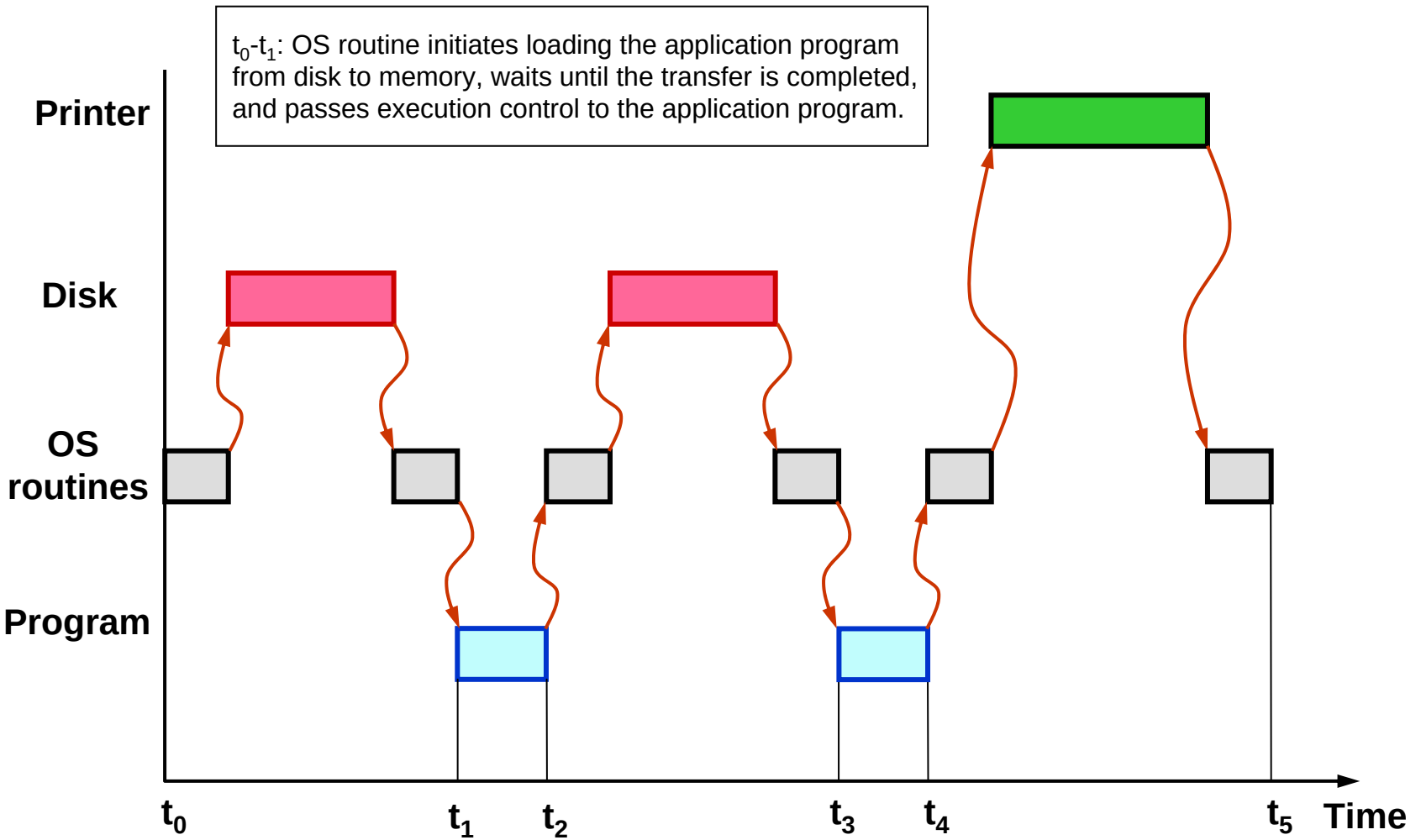
- Translating programs from source form prepared by the user into object form consisting of machine instructions
- Linking and running user-written application programs with existing standard library routines, such as numerical computation packages

➢ System software is thus responsible for the coordination of all activities in a computing system

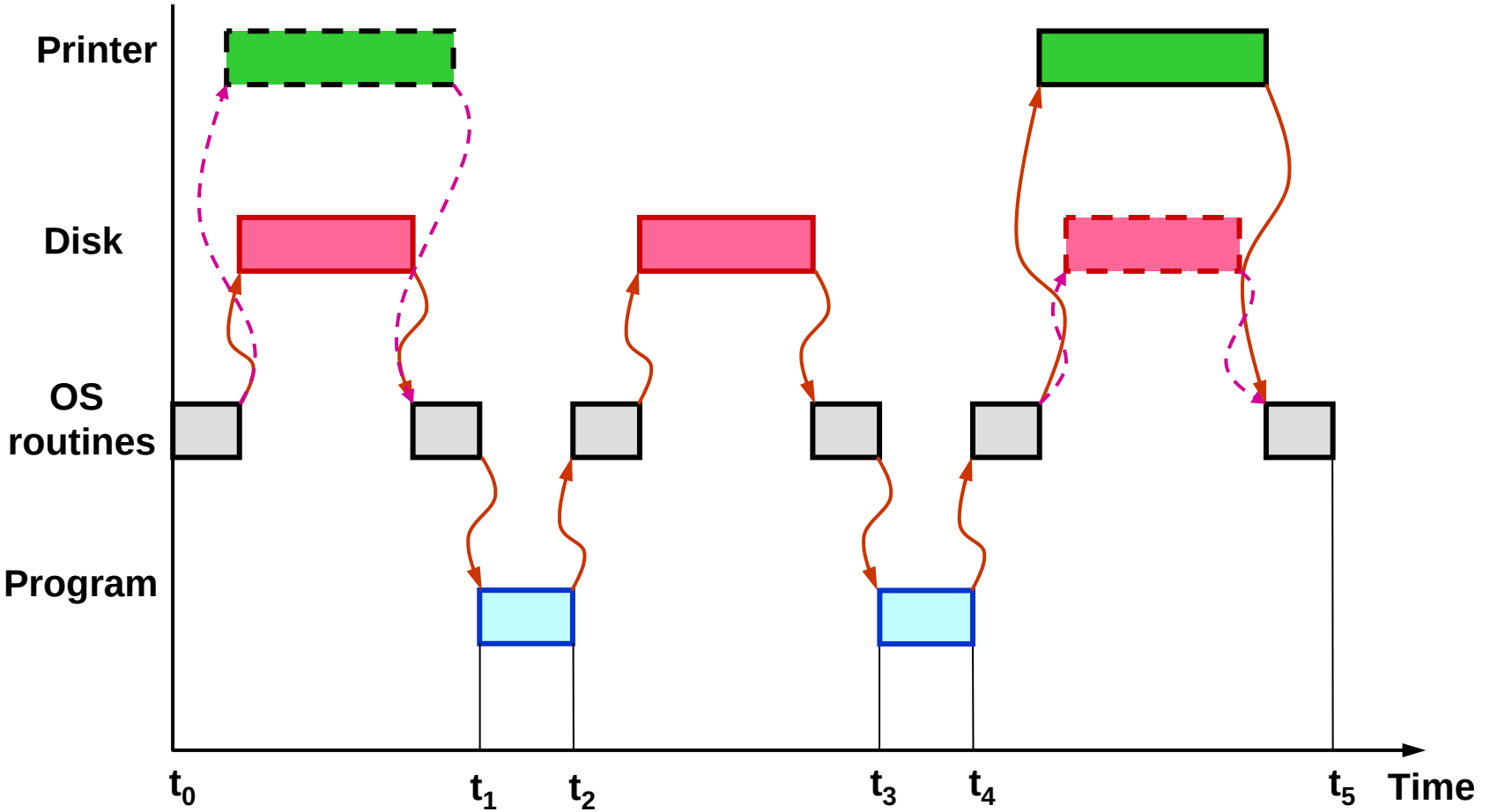
Operating System

- Operating system (OS)
  - This is a large program, or actually a collection of routines, that is used to control the sharing of and interaction among various computer units as they perform application programs
- The OS routines perform the tasks required to assign computer resource to individual application programs
  - These tasks include assigning memory and magnetic disk space to program and data files, moving data between memory and disk units, and handling I/O operations
- In the following, a system with one processor, one disk, and one printer is given to explain the basics of OS
  - Assume that part of the program's task involves reading a data file from the disk into the memory, performing some computation on the data, and printing the results

User Program and OS Routine Sharing



Multiprogramming or Multitasking



Performance

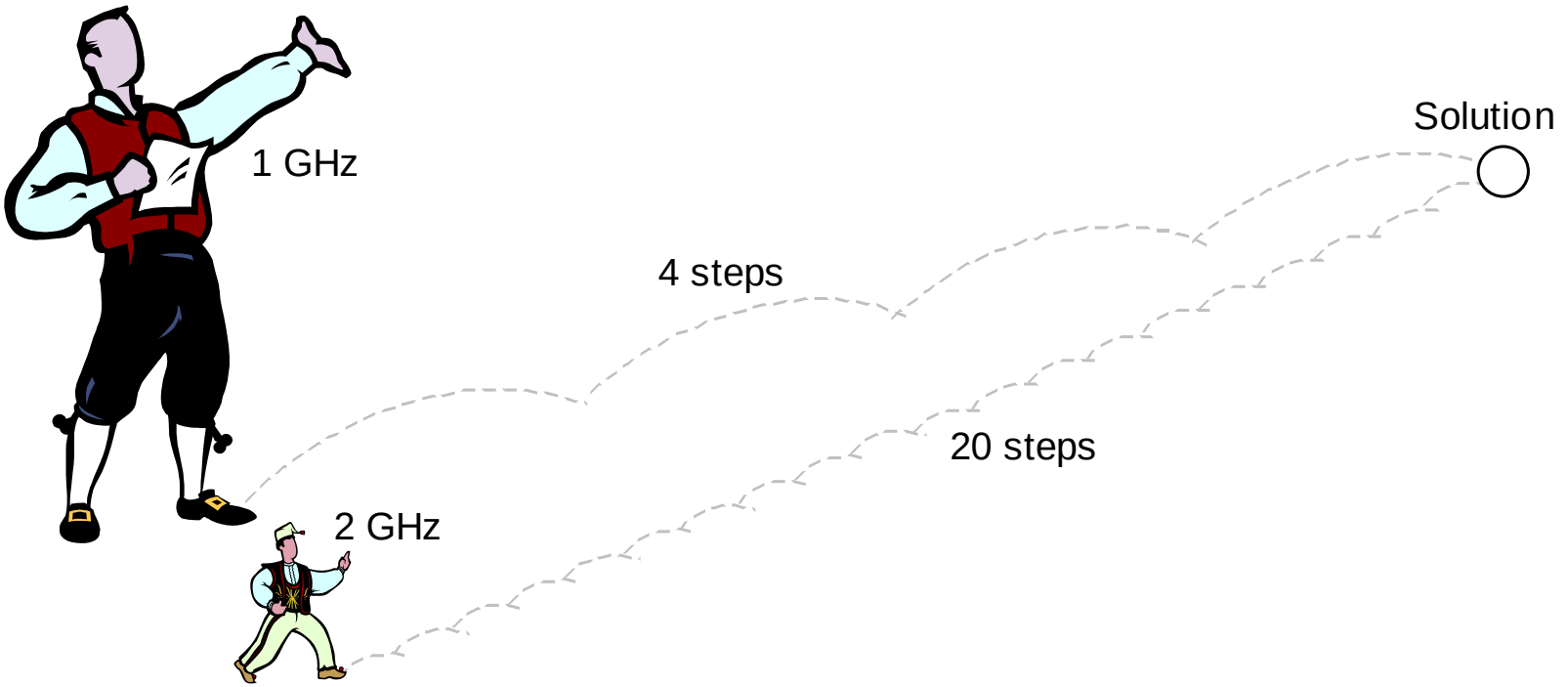
- The speed with which a computer executes programs is affected by the design of its hardware and its machine language instructions
- Because programs are usually written in a high-level language, performance is also affected by the compiler that translates programs into machine languages
- For best performance, the following factors must be considered
  - Compiler
  - Instruction set
  - Hardware design

Performance

- Processor circuits are controlled by a timing signal called a clock
  - The clock defines regular time intervals, called clock cycles
- To execute a machine instruction, the processor divides the action to be performed into a sequence of basic steps, such that each step can be completed in one clock cycle
- Let the length P of one clock cycle, its inverse is the clock rate, R=1 / P
- Basic performance equation
  - T=(NxS) / R, where T is the processor time required to execute a program, N is the number of instruction executions, and S is the average number of basic steps needed to execute one machine instruction

Faster Clock=Shorter Running Time?

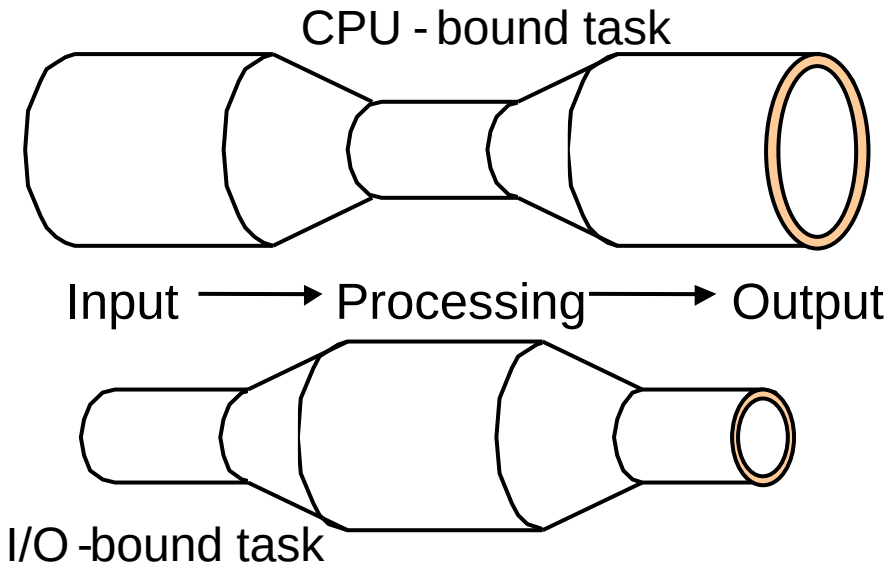
- Faster steps do not necessarily mean shorter travel time



[Source: B. Parhami, UCSB]

System Balance is Essential

- Note that system balance is absolutely essential for improving performance
- If one replaces a machine's processor with a model having twice the performance, this will not double the overall system performance unless corresponding improvements are made to other parts of the system



[Source: B. Parhami, UCSB]

Performance Improvement

- Pipelining and superscalar operation
  - Pipelining: by overlapping the execution of successive instructions
  - Superscalar: different instructions are concurrently executed with multiple instruction pipelines. This means that multiple functional units are needed
- Clock rate improvement
  - Improving the integrated-circuit technology makes logic circuits faster, which reduces the time needed to complete a basic step
  - Reducing amount of processing done in one basic step also makes it possible to reduce the clock period, P. However, if the actions that have to be performed by an instruction remain the same, the number of basic steps needed may increase
- Reduce the number of basic steps to execute
  - Reduced instruction set computers (RISC) and complex instruction set computers (CISC)

Reporting Computer Performance

- Measured or estimated execution times for three programs

	Time on machine X	Time on machine Y	Speedup of Y over X
Program A	20	200	0.1
Program B	1000	100	10.0
Program C	1500	150	10.0
All 3 prog's	2520	450	5.6

- Analogy
  - If a car is driven to a city 100 km away at 100 km / hr and returns at 50 km / hr, the average speed is not (100 + 50) / 2 but is obtained from the fact that it travels 200 km in 3 hours

[Source: B. Parhami, UCSB]