Basic Operational Concepts

Basic Operational Concepts

- Program
  - Set of instruction
  - Used to perform a specific task
  - Stored in memory
  - Individual instructions are brought from the memory into the processor
  - specific operations are performed by executing instruction
  - Data to be used as instruction operands are also stored in the memory

Basic Operational Concepts

- A typical instruction:
  - **Load R2, LOC // R2 ←LOC**
  - reads the contents of a memory location ‘LOC’ and store in register ‘R2’
  - Content of
    - LOC are preserved
    - R2 are overwritten

Basic Operational Concepts

- Execution of instruction: (requires several steps)
- Load Operation:
  - Fetch the instruction- from memory into processor
    - Load R2, LOC
  - Determine the operation - by the control unit
    - Load (read from memory)
  - Fetch the operand - from the memory into processor
    - Read LOC
  - Perform the operation - Operand is stored in register
    - Store LOC content into R2

Basic Operational Concepts

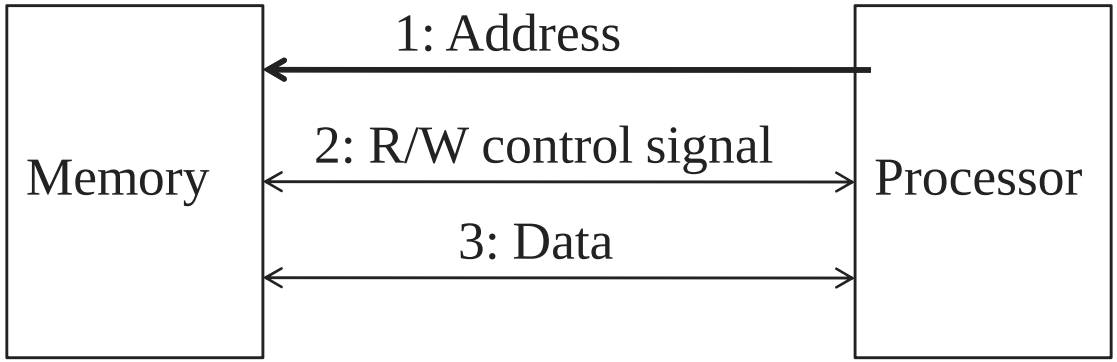
- ALU Operation:
  - After operands have been loaded from memory into processor registers, arithmetic or logic operations can be performed on them
  - For example, the instruction
    - **Add R4, R2, R3**
      - R4 ← R2+R3
      - operands in R2 and R3 are not altered
      - R4 is overwritten by the sum

Basic Operational Concepts

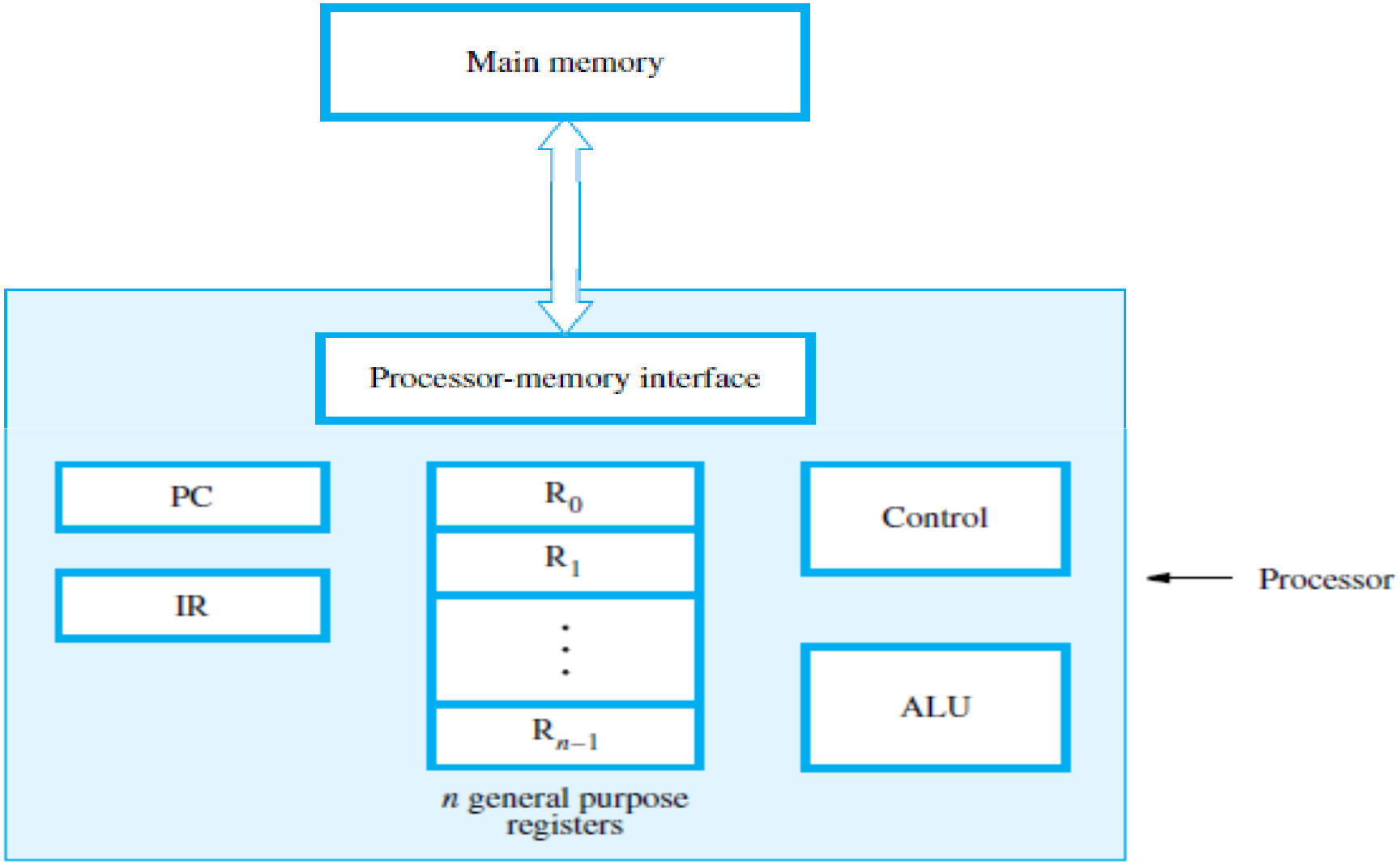
- Store Operation:
  - After completing the desired operations, the results are in processor registers.
  - They can be transferred to the memory using instructions such as
    - **Store R4, LOC**
      - LOC←R4
      - LOC are overwritten
      - R4 are preserved

Basic Operational Concepts

- Memory Transfer:
  - **Load, Store**
    - Transfers between the memory and the processor
    - Initiated by sending the address of the desired memory location to the memory unit
    - Then asserting the appropriate control signals
    - The data are then transferred to or from the memory



Basic Operational Concepts



Connection between the processor and the main memory.

Basic Operational Concepts

- Processor
  - *ALU*
  - *Control circuitry*
  - *Registers*
    - *Specialized register*
      - *Instruction register (IR)*
      - *Program counter (PC)*
    - *General-purpose registers :*
      - *R0 through Rn−1*
  - Processor-memory interface

Basic Operational Concepts

- **Instruction Register (IR)**
  - *Holds the* instruction that is currently being executed
  - Its output is available to the control circuits
  - Control circuit generate the timing signals that control the various processing elements involved in executing the instruction

Basic Operational Concepts

- *Program counter (PC)*
  - Contains the memory address of the next instruction to be fetched and executed
  - PC *points to the next* instruction that is to be fetched from the memory
  - During the execution of an instruction, the contents of the PC are updated to correspond to the address of the next instruction to be executed.

Basic Operational Concepts

- *General-purpose registers*
  - *R0 through Rn−1,*
  - *Called processor registers*
  - Hold operands

Basic Operational Concepts

- Processor-memory interface
  - Manages the transfer of data between the main memory and the processor : R/W
  - Read:
    - Interface sends the address of that word to the memory along with a Read control signal
    - The interface waits for the word to be retrieved, then transfers it to the appropriate processor register
  - Write:
    - If a word is to be written into memory, the interface transfers both the address and the word to the memory along with a Write control signal

Basic Operational Concepts

- Operating steps
    - Instruction Fetch
    - Operand Fetch
    - Instruction Execution
    - Store Operand

Basic Operational Concepts

- Operating steps
  - A program must be in the main memory in order for it to be executed
  - It is often transferred there from secondary storage through the input unit
  - Execution of the program begins when the PC is set to point to the first instruction of the program
  - The contents of the PC are transferred to the memory along with a Read control signal
  - When the addressed word (in this case, the first instruction of the program) has been fetched from the memory it is loaded into register IR
  - At this point, the instruction is ready to be interpreted and executed.

Basic Operational Concepts

- Operating steps
  - If an operand that resides in the memory is required for an instruction, it is fetched by sending its address to the memory and initiating a Read operation
  - When the operand has been fetched from the memory, it is transferred to a processor register
  - ALU can perform a desired arithmetic operation, such as Add, on the values in processor registers
  - The result is sent to a processor register
  - If the result is to be written into the memory with a Store instruction, it is transferred from the processor register to the memory, along with the address of the location where the result is to be stored, then a Write operation is initiated

Basic Operational Concepts

- Operating steps
  - At some point during the execution of each instruction, the contents of the PC are incremented so that the PC points to the next instruction to be executed
  - Thus, as soon as the execution of the current instruction is completed, the processor is ready to fetch a new instruction.
  - In addition to transferring data between the memory and the processor, the computer accepts data from input devices and sends data to output devices.
  - Thus, some machine instructions are provided for the purpose of handling **I/O transfers.**

Basic Operational Concepts

- Operating steps
  - Normal execution of a program may be preempted if some device requires urgent service.
  - For example, a monitoring device in a computer-controlled industrial process may detect a dangerous condition.
  - In order to respond immediately, execution of the current program must be suspended.
  - To cause this, the device raises an *interrupt signal, which* is a request for service by the processor. The processor provides the requested service by executing a program called an ***interrupt-service routine.***

Basic Operational Concepts

- ***interrupt-service routine***
- *Alter* the internal state of the processor
- Its state must be saved in the memory before servicing the interrupt request
- Normally, the information that is saved includes the contents of the PC, the contents of the general-purpose registers, and some control information
- When the interrupt-service routine is completed, the state of the processor is restored from the memory so that the interrupted program may continue.