Peripheral devices

- In addition to the processor and a set of memory modules, the third key element of a computer system is a set of input-output subsystem referred to as I/O, provides an efficient mode of communication between the central system and the outside environment.
- Programs and data must be entered into computer memory for processing and results obtained from computations must be recorded or displayed for the user.
- Devices that are under the direct control of the computer are said to be connected on-line. These devices are designed to read information into or out of the memory unit upon command from CPU.
- Input or output devices attached to the computer are also called peripherals.
- Among the most common peripherals are keyboards, display units, and printers.
- Perhaps those provide auxiliary storage for the systems are magnetic disks and tapes.
- Peripherals are electromechanical and electromagnetic devices of some complexity.
- We can broadly classify peripheral devices into three categories:
    - **Human Readable**: Communicating with the computer users, e.g. video display terminal, printers etc.
    - **Machine Readable**: Communicating with equipments, e.g. magnetic disk, magnetic tape, sensor, actuators used in robotics etc.
    - **Communication**: Communicating with remote devices means exchanging data with that, e.g. modem, NIC (network interface Card) etc.

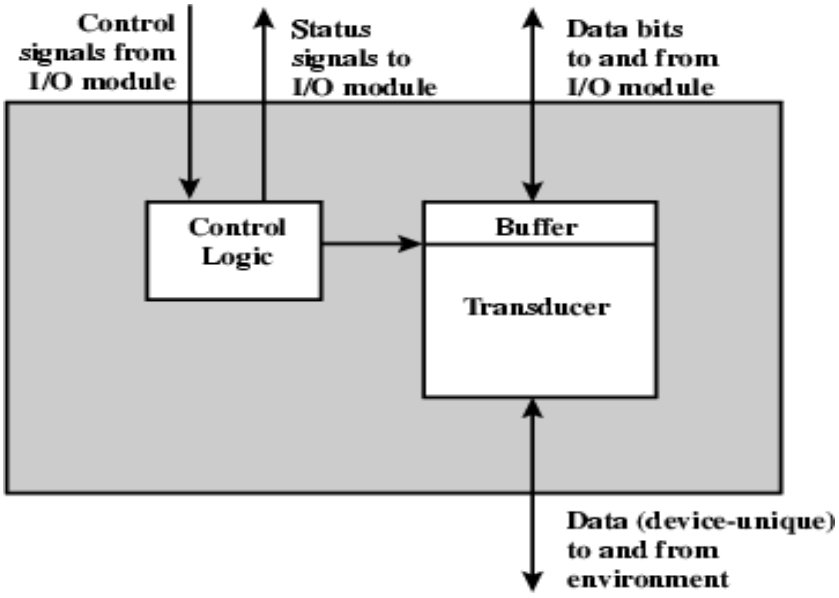


Fig: Block diagram of Peripheral device

- Control signals determine the function that the device will perform such as senddata to I/O module, accept data from I/O module.
- Status signals indicate the state of the device i.e. device is ready or not.
- Data bits are actual data transformation.

- Control logic associated with the device controls the device's operation in response to direction from the I/O module.
- The transducer converts data from electrical to other forms of energy during output and from other forms to electrical during input.
- Buffer is associated with the transducer to temporarily hold data being transferredbetween the I/O module and external devices i.e. peripheral environment.

Input Device

- Keyboard
- Optical input devices
  - Card Reader
  - Paper Tape Reader
  - Optical Character Recognition (OCR)
  - Optical Bar code reader (OBR)
  - Digitizer
  - Optical Mark Reader
- Magnetic Input Devices
  - Magnetic Stripe Reader
  - Magnetic Ink Character Recognition (MICR)
- Screen Input Devices
  - Touch Screen
  - Light Pen
  - Mouse
- Analog Input Devices

Output Device

- Card Puncher, Paper Tape Puncher
- Monitor (CRT, LCD, LED)
- Printer (Impact, Ink Jet, Laser, Dot Matrix)
- Plotter
- Analog
- Voice

I/O modules

- I/O modules interface to the system bus or central switch (CPU and Memory), interfaces and controls to one or more peripheral devices. I/O operations are accomplished througha wide assortment of external devices that provide a means of exchanging data between external environment and computer by a link to an I/O module. The link is used to exchange control status and data between I/O module and the external devices.

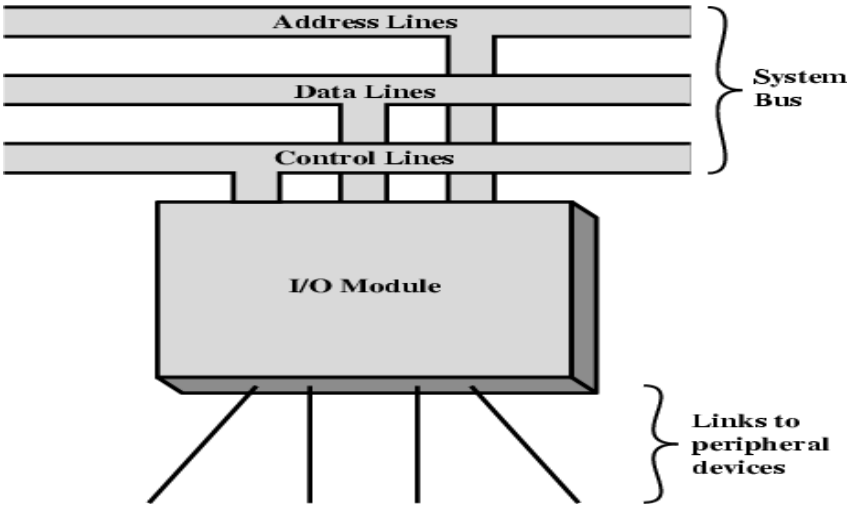


Fig: Model of I/O module

- Peripherals are not directly connected to the system bus instead an I/O module is used which contains logic for performing a communication between the peripherals and the system bus. The reasons due to which peripherals do not directly connected to the system bus are:
  - There are a wide variety of peripherals with various methods of operation. It would be impractical to incorporate the necessary logic within the processor to control a range of devices.
  - The data transfer rate of peripherals is often much slower than that of the memory or processor. Thus, it is impractical to use high speed system bus to communicate directly with a peripheral and vice versa.
  - Peripherals often use different data format and word length than the computer to which they are connected.
- Thus an I/O module is required which performs two major functions.
  - Interface to the processor and memory via the system bus
  - Interface to one or more peripherals by tailored data links

I/O Module Functions

- The I/O module is a special hardware component interface between the CPU and peripherals to supervise and synchronize all I/O transformation The detailed functions of I/O modules are;

  Control & Timing: I/O module includes control and timing to coordinate the flow of traffic between internal resources and external devices. The control of the transfer of data from external devices to processor consists following steps:
  - The processor interrogates the I/O module to check status of the attached device.
  - The I/O module returns the device status.
  - If the device is operational and ready to transmit, the processor requests

thetransfer of data by means of a command to I/O module.

- The I/O module obtains the unit of data from the external device.
- The data are transferred from the I/O module to the processor.

- **Processor Communication**: I/O module communicates with the processor which involves: Command decoding: I/O module accepts commands from the processor.
- Data: Data are exchanged between the processor and I/O module over the bus.
- Status reporting: Peripherals are too slow and it is important to know the status of I/O module.
- Address recognition: I/O module must recognize one unique address for each peripheral it controls.

Device Communication: It involves commands, status information and data.

Data Buffering: I/O module must be able to operate at both device and memory speeds. If the I/O device operates at a rate higher than the memory access rate, then the I/O module performs data buffering. If I/O devices rate slower than memory, it buffers data so as not to tie up the memory in slower transfer operation.

Error Detection: I/O module is responsible for error detection such as mechanical and electrical malfunction reported by device e.g. paper jam, bad ink track & unintentional changes to the bit pattern and transmission error.

I/O Module Structure

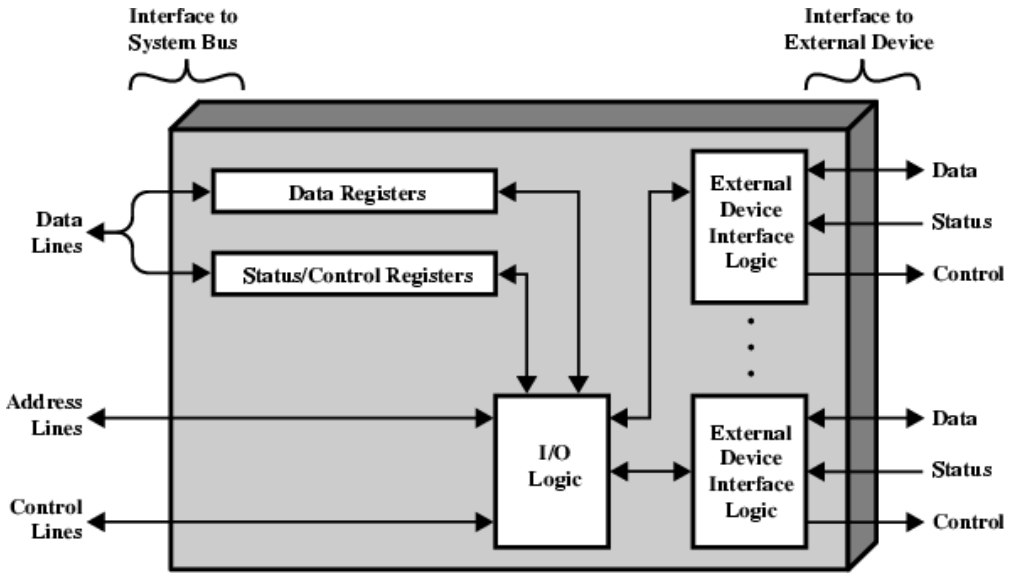


Fig: Block diagram of I/O Module

- The I/O bus from the processor is attached to all peripheral interfaces
- To communicate with the particular devices, the processor places a device address on the address bus.
- Each interface contains an address decoder that monitors the address line. When the interface detects the particular device address, it activates the path between the data line and devices that it controls.
- At the same time that the address is made available in the address line, the processor

provides a function code in the control way includes control command, output data and input data.

I/O Module Decisions

- Hide or reveal device properties to CPU
- Support multiple or single device

- Control device functions or leave for CPU
- Also O/S decisions
  - e.g. Unix treats everything it can as a file

Input-Output interface

- Input-Output interface provides a method for transferring information between internalstorage (such as memory and CPU registers) and external I/O devices.
- Peripherals connected to a computer need special communication links for interfacingthem with the central processing unit.
- The communication link resolves the following *differences* between the computer andperipheral devices.
  - Devices and signals
    Peripherals - Electromechanical DevicesCPU or
    Memory - Electronic Device
  - Data Transfer Rate
    Peripherals - Usually slower
    CPU or Memory - Usually faster than peripherals
    Some kinds of Synchronization mechanism may be needed
  - Unit of Information
    Peripherals - Byte
    CPU or Memory - Word
  - Operating Modes
    Peripherals - Autonomous, AsynchronousCPU or
    Memory – Synchronous
- To resolve these differences, computer systems include special hardware components (Interfaces) between the CPU and peripherals to supervise and synchronize all input and output interfaces.

I/O Bus and Interface Modules

- The I/O bus consists of data lines, address lines and control lines.

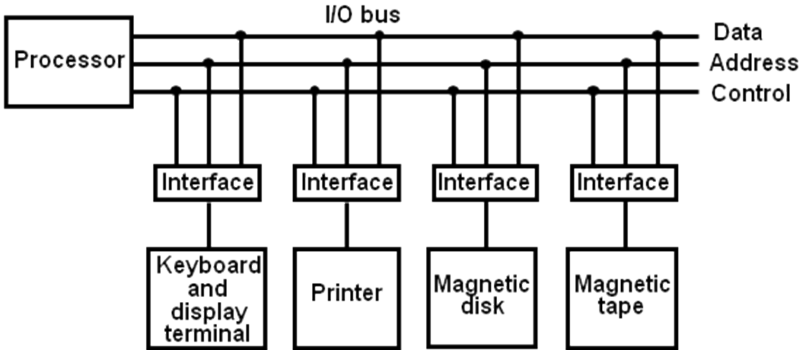


Fig: Connection of I/O bus to input-output devices

- Interface performs the following:
  - Decodes the device address (device code)
  - Decodes the commands (operation)
  - Provides signals for the peripheral controller
  - Synchronizes the data flow and supervises the transfer rate between peripheraland CPU or Memory
- I/O commands that the interface may receive:
  - Control command: issued to activate the peripheral and to inform it what to do.
  - Status command: used to test various status conditions in the interface and theperipheral.
  - Output data: causes the interface to respond by transferring data from the bus intoone of its registers.
  - Input data: is the opposite of the data output.

I/O versus Memory Bus

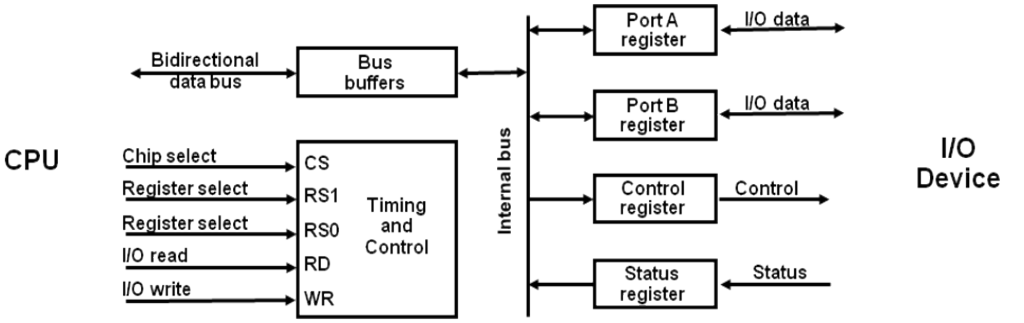
- Computer buses can be used to communicate with memory and I/O in three ways:
  - Use two separate buses, one for memory and other for I/O. In this method, all data, address and control lines would be separate for memory and I/O.
  - Use one common bus for both memory and I/O but have separate control lines. There is a separate read and write lines; I/O read and I/O write for I/O and memory read and memory write for memory.
  - Use a common bus for memory and I/O with common control line. This I/O configuration is called memory mapped.

Isolated I/O versus Memory Mapped I/O

- **Isolated I/O**
  - Separate I/O read/write control lines in addition to memory read/write control lines
  - Separate (isolated) memory and I/O address spaces
  - Distinct input and output instructions

- Memory-mapped I/O
  - A single set of read/write control lines (no distinction between memory and I/Otransfer)
  - Memory and I/O addresses share the common address space which reduces memoryaddress range available
  - No specific input or output instruction so the same memory reference instructions canbe used for I/O transfers
  - Considerable flexibility in handling I/O operation

Example of I/O Interface



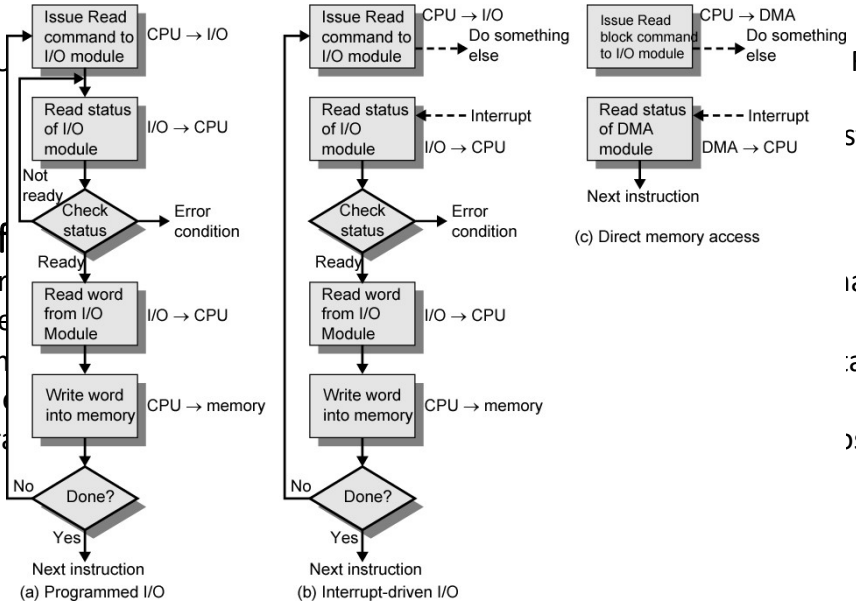
CS	RS1	RS0	Register selected
0	x	x	None - data bus in high-impedence
1	0	0	Port A register
1	0	1	Port B register
1	1	0	Control register
1	1	1	Status register

- Information in each port can be assigned a meaning depending on the mode of operationof the I/O device
- CPU Register

ster, it is possible to

Modes of

- Data Tr handled in a variety ofmode
- Some m a directly to and fromth
- Data tra ossible modes.
  - o
  - o
  - o
  - o



Programmed I/O

- Programmed I/O operations are the result of I/O instructions written in the computer program.
- In programmed I/O, each data transfer in initiated by the instructions in the CPU and hence the CPU is in the continuous monitoring of the interface.
- Input instruction is used to transfer data from I/O device to CPU, store instruction is used to transfer data from CPU to memory and output instruction is used to transfer data from CPU to I/O device.
- This technique is generally used in very slow speed computer and is not a efficient method if the speed of the CPU and I/O is different.

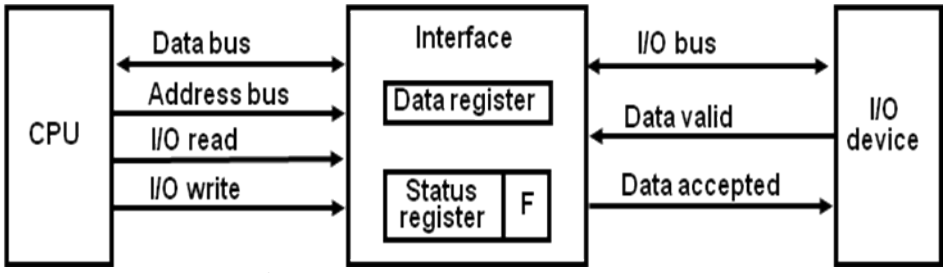


Fig: Data transfer from I/O device to CPU

- I/O device places the data on the I/O bus and enables its data valid signal
- The interface accepts the data in the data register and sets the F bit of statusregister and also enables the data accepted signal.
- Data valid line is disables by I/O device.
- CPU is in a continuous monitoring of the interface in which it checks the F bit of the status register.
    - If it is set i.e. 1, then the CPU reads the data from data register and sets Fbit to zero
    - If it is reset i.e. 0, then the CPU remains monitoring the interface.
- Interface disables the data accepted signal and the system goes to initial statewhere next item of data is placed on the data bus.

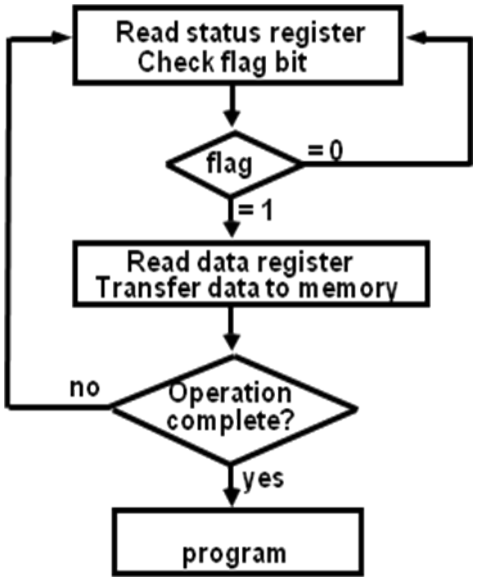


Fig: Flowchart for CPU program to input data

Characteristics:

- Continuous CPU involvement

- CPU slowed down to I/O speed
- Simple
- Least hardware

Polling, or polled operation, in computer science, refers to actively sampling the status of an external device by a client program as a synchronous activity. Polling is most often used in terms of input/output (I/O), and is also referred to as **polled I/O or software driven I/O.**