

The latter technique is more common and is referred to as cycle stealing, because the DMA module in effect steals a bus cycle

Figure 7.7 Typical DMA Block Diagram When the processor wishes to read or write a block of data, it issues a command to the DMA module, by sending to the DMA module the following information: Whether a read or write is requested, using the read or write control line between the processor and the DMA module. I/O controllers are commonly seen on microcomputers, whereas I/O channels are used on mainframe

Figure 7.2 Block Diagram of an I/O Module 7.3 Programmed I/O With programmed I/O, data are exchanged between the processor and the I/O module. Similarly, if the I/O device operates at a rate higher than the memory access rate, then the I/O module performs the needed buffering operation Finally, an I/O module is often responsible for error detection and for subsequently reporting errors to the processor. There are four types of I/O commands that an I/O module may receive when it is addressed by a processor:

- o Control: Used to activate a peripheral and tell it what to do.
- o Test: Used to test various status conditions associated with an I/O module and its peripherals. Typically, a buffer is associated with the transducer to temporarily hold data being transferred between the I/O module and the external environment

KEYBOARD/MONITOR The basic unit of exchange here is the character : ? When the processor is executing a program and encounters an instruction relating to I/O, it executes that instruction by issuing a command to the appropriate I/O module.

- o Read: Causes the I/O module to obtain an item of data from the peripheral and place it in an internal buffer
- o Write: Causes the I/O module to take an item of data (byte or word) from the data bus and subsequently transmit that data item to the peripheral

Figure 7.3a gives an example of the use of programmed I/O to read in a block of data from a peripheral device into memory. The minimum information required is (a) the status of the processor, which is contained in a register called the program status word (PSW); and (b) the location of the next instruction to be executed, processor now loads the program counter with the entry location of the interrupt-handling program that will respond to this interrupt. As a result, the next instruction to be executed

Figure 7.5 **Figure 7.6** Changes in Memory and Registers for an Interrupt

7.5 Direct Memory Access When large volumes of data are to be moved, a more efficient technique is required: direct memory access (DMA). The number of words to be read or written, again communicated via the data lines and stored in the data count register

Figure 7.8 shows where in the instruction cycle the processor may be suspended.

INPUT/OUTPUT 7.1 External Devices I/O operations are accomplished through a wide assortment of external devices that provide a means of exchanging data between the external environment and the computer. In a fixed-head disk, the transducer is capable of converting between the magnetic patterns on the moving disk surface and bits in the device's buffer (**Figure 7.1**). This system includes two basic components (which not only connects an I/O device with the system bus, but plays a very crucial role in between) :

- i- the I/O devices and I/O module,
- ii- Command decoding: The I/O module accepts commands from the processor, typically sent as signals on the control bus. For example, an I/O module for a disk drive might accept the following commands: READ SECTOR, WRITE SECTOR, SEEK track number, and SCAN record ID. The latter two commands each include a parameter that is sent on the data bus. This communication involves commands, status information, and data (An essential task of an I/O module is data buffering. An I/O module that takes on most of the detailed processing burden, presenting a high-level interface to the processor, is usually referred to as an I/O

channel or I/O processor. To explain the programmed I/O technique, we view it first from the point of view of the I/O commands issued by the processor to the I/O module, and then from the point of view of the I/O instructions executed by the processor.

I/O COMMANDS To execute an I/O-related instruction, the processor issues an address, specifying the particular I/O module and external device, and an I/O command.

I/O INSTRUCTIONS With programmed I/O, there is a close correspondence between the I/O-related instructions that the processor fetches from memory and the I/O commands that the processor issues to an I/O module to execute the instructions. Because the address space for I/O is isolated from that for memory, this is referred to as isolated I/O memory-mapped I/O.

7.4 Memory-Mapped and Isolated I/O

7.4 Interrupt-Driven I/O

The problem with programmed I/O is that the processor has to wait a long time for the I/O module of concern to be ready for either reception or transmission of data. When the interrupt from the I/O module occurs, the processor saves the context (e.g., program counter and processor registers) of the current program and processes the interrupt. The processor finishes execution of the current instruction before responding to the interrupt. The processor tests for an interrupt, and sends an acknowledgment signal to the device that issued the interrupt. Control signals determine the function that the device will perform, such as send data to the I/O module (INPUT or READ), accept data from the I/O module (OUTPUT or WRITE), report status, or perform some control function particular to the device (e.g., position a disk head).

Printable characters are the alphabetic, numeric, and special characters that can be printed on paper or displayed on a screen. On output, IRA code characters are transmitted to an external device from the I/O module. A moving-head disk must also be able to cause the disk arm to move radially in and out across the disk's surface.

7.2 I/O MODULES

The computer consists of an I/O (input-output) system. Functions of I/O Module: The major functions of an I/O module are:

- Processor communication -- this involves the following tasks:
 - When a byte is received, the I/O module checks the parity to determine whether an error has occurred.

7.2.1 I/O module structure

Figure 7.2 provides a general block diagram of an I/O module. The module must also be able to recognize and generate addresses associated with the devices it controls. An I/O module that is quite primitive and requires detailed control is usually referred to as an I/O controller or device controller. With programmed I/O, the I/O module will perform the requested action and then set the appropriate bits in the I/O status register. The I/O module takes no further action to alert the processor. The processor treats the status and data registers of I/O modules as memory locations and uses the same machine instructions to access both memory and I/O devices. In this case, a user program is interrupted after the instruction at location N. The contents of all of the registers plus the address of the next instruction are pushed onto the stack. When interrupt processing is complete, the saved register values are retrieved from the stack and restored to the registers (e.g., see Figure 7.6b). Figure 7.5 shows a typical sequence.