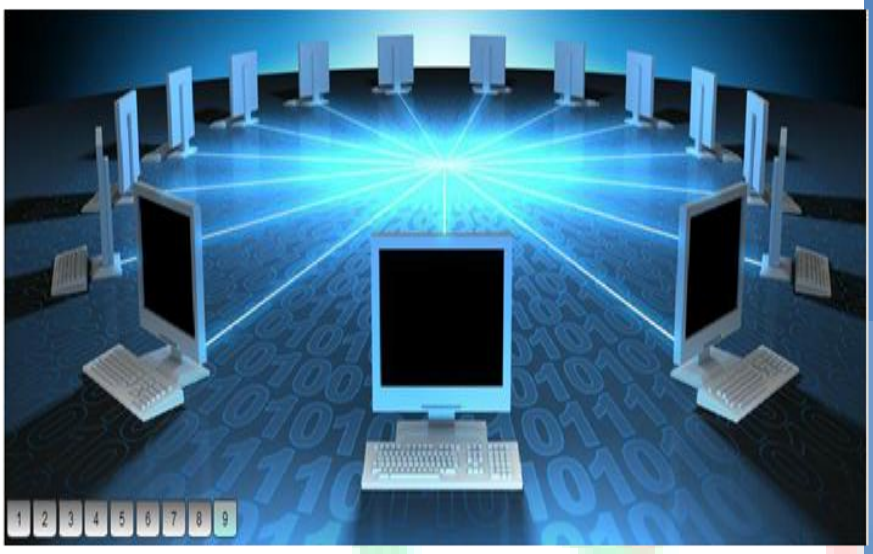
International Journal of Technology, Management & Humanities



www.ijtmh.com
ISSN (e) : 2454—566X

Volume - 1, Issue - 4
March - 2016

International Journal of Technology, Management and Humanities (IJTMH) refereed e-journal form in English.
International Journal of Technology, Management and Humanity is published on Quarterly basis with the aim to provide an appropriate platform presenting well considered, meaningful, constructively thought provoking and non-controversial but critically analyzing and synthesizing present and future aspects of Technical & scientific Education System with particular reference to the world.
The following types of article will be considered types of article will be considered

1. Research Articles: Original research in different fields of Science, Engineering and Management, Humanities will be evaluated as research articles.
2. Research Notes: These include articles such as manuscripts.
3. Reviews: Reviews of recent improvements, discoveries, developments, and thoughts in various fields of Science, management, and Engineering will be considered.
4. Frequency: FOUR issues in a year.

Indexing

JadounScience
http://www.jadounscience.com

International Society of Universal Research in Sciences

INDEX COPERNICUS INTERNATIONAL

DOAJ DIRECTORY OF OPEN ACCESS JOURNALS

Google Scholar

CABELL PUBLISHING, INC. Directories of Academic Journals

EBSCO HOST Electronic Journals Service

IJTMH

Contact Us
E-mail
submission@ijtmh.co
submissionijtmh@gmail.com
editor@ijtmh.com
editorijtmh@gmail.com

A Special Mode of I/O Operation that Overcome the Problem Programmed I/O, Interrupt I/O (Direct Memory Access)

Author

[1] Dr. Rakesh Kumar Gangwar, [2]Akash Singh

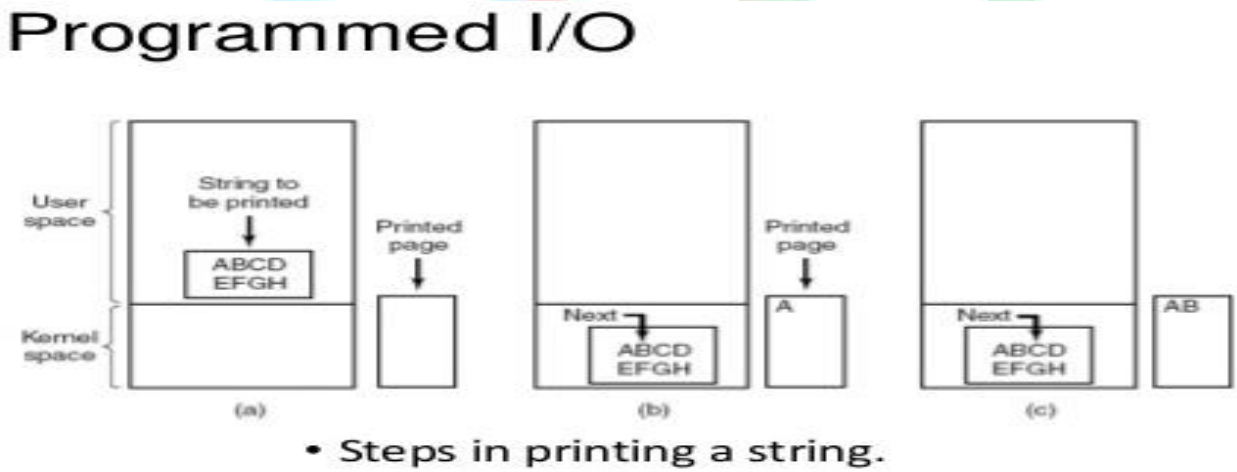
[1](Associate Professor / Department of Computer Science/Manipal University, Karnataka/India)
[2] (Research Scholar/Department of Computer Science/ Manipal University, Karnataka/India)

Abstract : *A general purpose computer should have the ability to exchange information with a wide range of devices in varying environments. Computers can communicate with other computers over the Internet and access information around the globe. They are an integral part of home appliances, manufacturing equipment, transportation systems, banking and point-of-sale terminals. In this chapter, we study the various ways in which I/O operations are performed.*

Keywords : *I/O Operations, CPU, PIO, Memory, DMA*

Programmed I/O

Programmed I/O (PIO) refers to data transfers initiated by a CPU under driver software control to access registers or memory on a device. The CPU issues a command then waits for I/O operations to be complete. As the CPU is faster than the I/O module, the problem with programmed I/O is that the CPU has to wait a long time for the I/O module of concern to be ready for either reception or transmission of data.
The CPU, while waiting, must repeatedly check the status of the I/O module, and this process is known as Polling. As a result, the level of the performance of the entire system is severely degraded.



Programmed I/O basically works in these ways

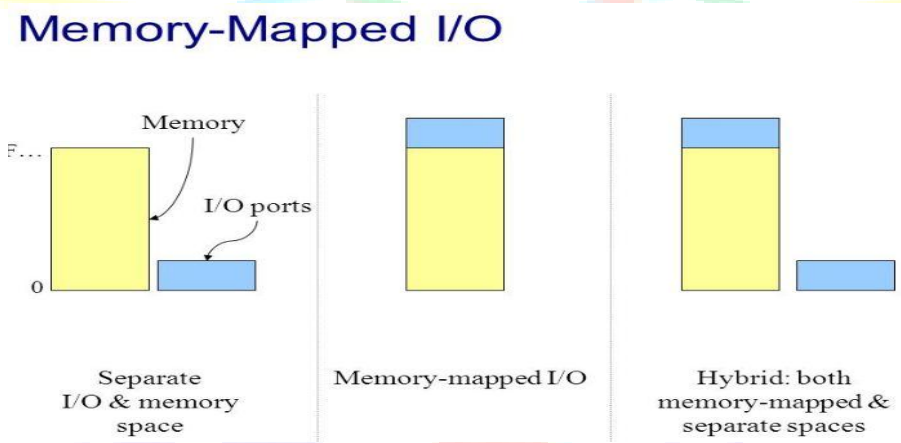
1. CPU requests I/O operation
2. I/O module performs operation
3. I/O module sets status bits
4. CPU checks status bits periodically
5. I/O module does not inform CPU directly
6. I/O module does not interrupt CPU
7. CPU may wait or come back later

Limitation of Programmed I/O

1. It used only in some low-end microcomputers.
2. It has single input and single output instruction.
3. Each instructions selects one I/O device (by number) and transfers a single character (byte)
4. Example: microprocessor controlled video terminal.
5. Four registers: input status and character, output status and character.

Memory Mapped I/O

There is a single address space for memory location and I/O devices. (the address space is shared) With memory mapped I/O a single read line a single write line are needed on the bus. The bus may be equipped with memory read and write plus Input and output command lines. Now the command lines specifies whether the address refers to memory location or an I/O device.

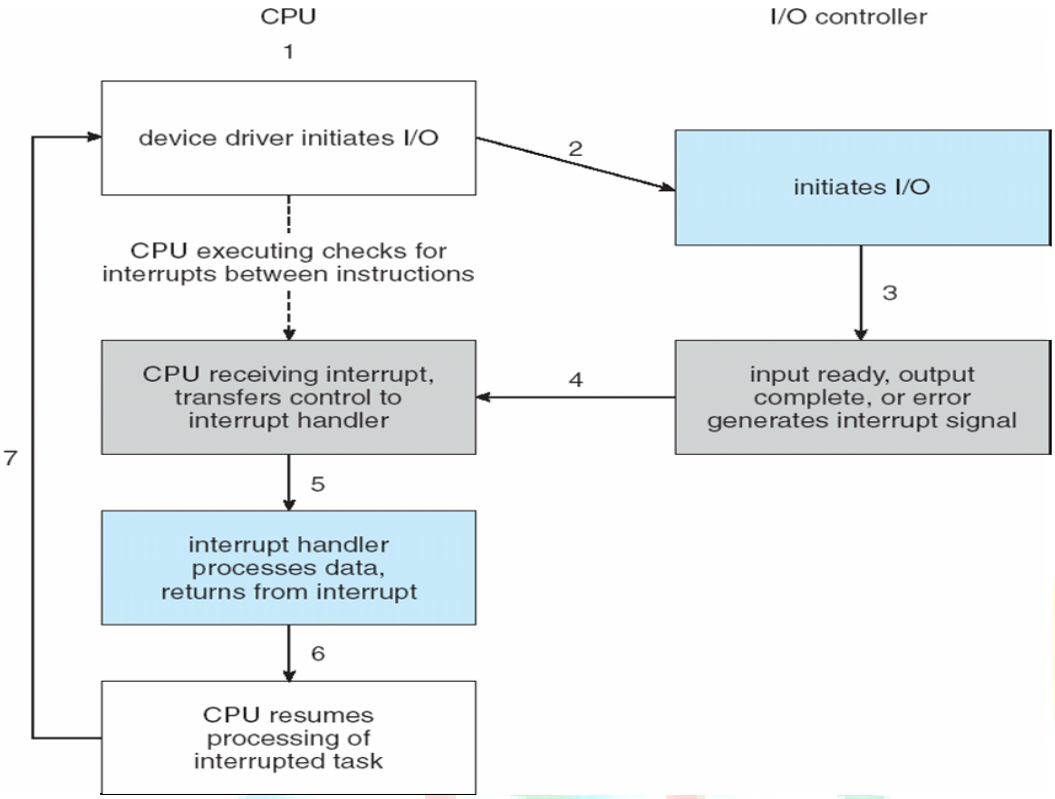


1. Most CPU uses memory mapped I/O.
2. Always CPU assigns address to memory some of memory space is stolen and assigned to I/O device.
3. It deals with fewer address lines.

Interrupt

The CPU issues commands to the I/O module then proceeds with its normal work until interrupted by I/O device on completion of its work. For input, the device interrupts the CPU when new data has arrived and is ready to be retrieved by the system processor. The actual actions to perform depend on whether the device uses I/O ports, memory mapping.

For output, the device delivers an interrupt either when it is ready to accept new data or to acknowledge a successful data transfer. Memory-mapped and DMA-capable devices usually generate interrupts to tell the system they are done with the buffer.
Although Interrupt relieves the CPU of having to wait for the devices, but it is still inefficient in data transfer of large amount because the CPU has to transfer the data word by word between I/O module and memory.



Below are the basic operations of Interrupt:

1. CPU issues read command
2. I/O module gets data from peripheral whilst CPU does other work
3. I/O module interrupts CPU
4. CPU requests data
5. I/O module transfers data

Limitation of Interrupt I/O

1. Primary disadvantage of programmed I/O is that CPU spends most of its time in a tight loop waiting for the device to become ready. This is called busy waiting.
2. With interrupt-driven I/O, the CPU starts the device and tells it to generate an interrupt when it is finished.
3. Done by setting interrupt-enable bit in status register.
4. Still requires an interrupt for every character read or written.
5. Interrupting a running process is an expensive business (requires saving context).
6. Requires extra hardware (DMA controller chip).

Direct Memory Access (DMA)

Direct Memory Access (DMA) means CPU grants I/O module authority to read from or write to memory without involvement. DMA module controls exchange of data between main memory and the I/O device. Because of DMA device can transfer data directly to and from memory, rather than using the CPU as an intermediary, and can thus relieve congestion on the bus. CPU is only involved at the beginning and end of the transfer and interrupted only after entire block has been transferred.

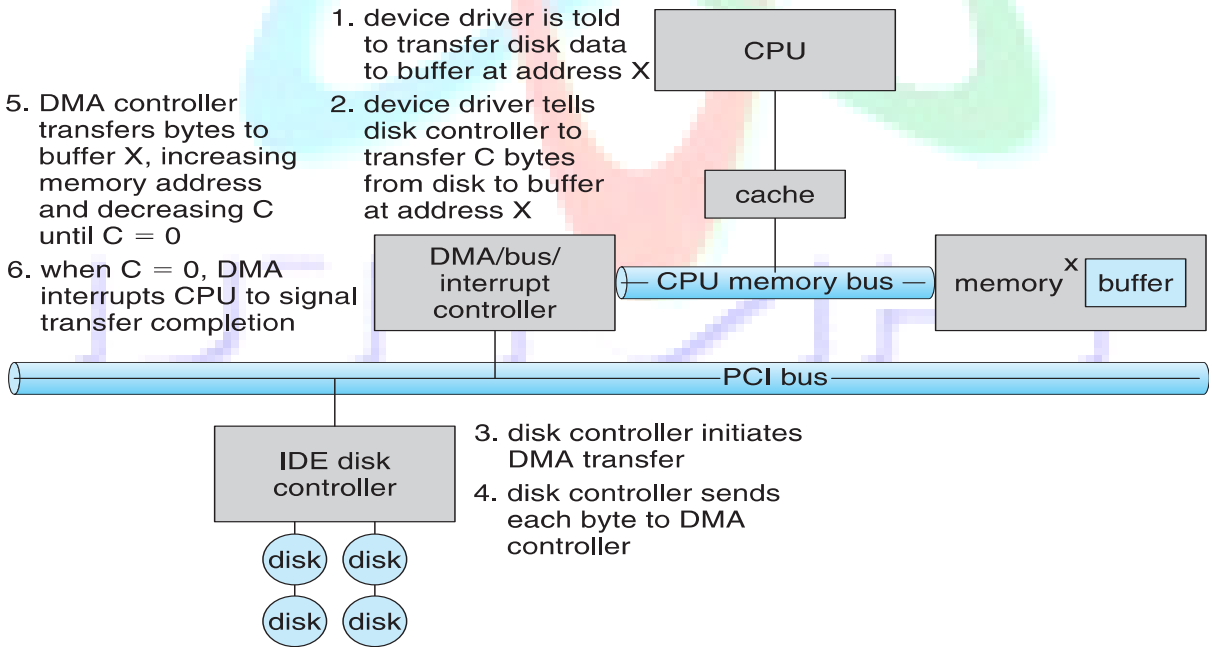
Direct Memory Access needs a special hardware called DMA controller (DMAC) that manages the data transfers and arbitrates access to the system bus. The controllers are programmed with source and destination pointers (where to read/write the data), counters to track the number of transferred bytes, and settings, which includes I/O and memory types, interrupts and states for the CPU cycles.

Direct Memory Access (DMA) means CPU grants I/O module authority to read from or write to memory without involvement. DMA module controls exchange of data between main memory and the I/O device. Because of DMA device can transfer data directly to and from memory, rather than using the CPU as an intermediary, and can thus relieve congestion on the bus. CPU is only involved at the beginning and end of the transfer and interrupted only after entire block has been transferred.

Direct Memory Access needs a special hardware called DMA controller (DMAC) that manages the data transfers and arbitrates access to the system bus. The controllers are programmed with source and destination pointers (where to read/write the data), counters to track the number of transferred bytes, and settings, which includes I/O and memory types, interrupts and states for the CPU cycles.

DMA increases system concurrency by allowing the CPU to perform tasks while the DMA system transfers data via the system and memory busses. Hardware design is complicated because the DMA controller must be integrated into the system, and the system must allow the DMA controller to be a bus master. Cycle stealing may also be necessary to allow the CPU and DMA controller to share use of the memory bus.

..



Types of DMA Transfer

1. Cycle stealing
2. Brust mode

1. Cycle Stealing

DMA controller 'steals' memory cycles from the processor though processor originates most memory access**.**

2. Brust mode

The DMA controller may be given exclusive access to the main memory to trnasfer a block of data without interruption.

Conflicts of DMA

1. Conflict between processor and DMA
2. Two DMA Controller try to access the BUS at same time to access the main memory.

Conclusion

DMA increases system concurrency by allowing the CPU to perform tasks while the DMA system transfers data via the system and memory busses
Hardware design is complicated because the DMA controller must be integrated into the system, and the system must allow the DMA controller to be a bus master.

References

[1]. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, fifth edition,
[2]. Mc-graw Hill higher education.
[3]. John P. Hayes, Computer architecture and organization; (2nd ed.), McGraw-Hill, Inc., New York, NY, 1988
[4]. John Daintith. "interrupt I/O." A Dictionary of Computing. 2004. Retrieved March 05, 2016 from Encyclopedia.com: http://www.encyclopedia.com/doc/1O11-interruptIO.html