



VIETNAM NATIONAL UNIVERSITY HANOI (VNU)
INFORMATION TECHNOLOGY INSTITUTE

Computer Architecture

Lecture 7: Input/Output Control

Duy-Hieu Bui, PhD

VNU Information Technology Institute
Laboratory for Smart Integrated System (SISLAB)

Email: hieubd@vnu.edu.vn

<https://duyhieubui.github.io>



1. Giới thiệu

2. Khối kiểm soát vào/ra

- Chức năng
- Cấu trúc

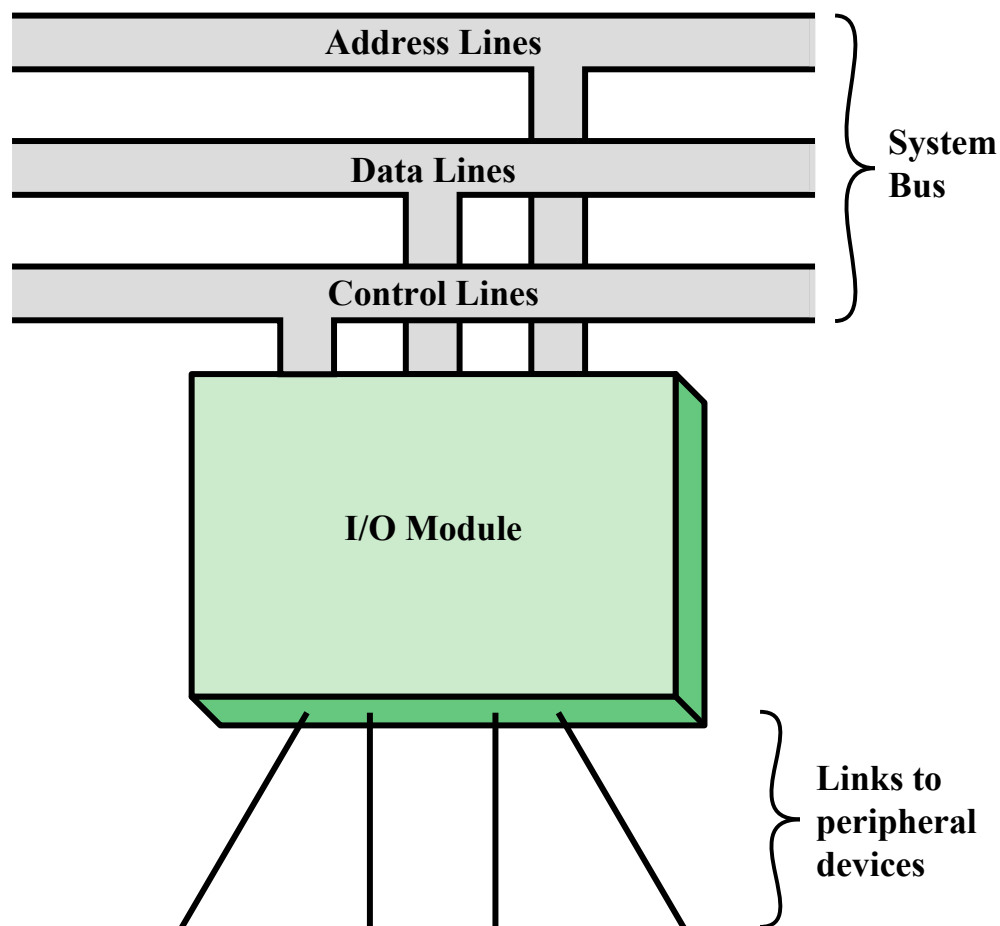
3. Kỹ thuật kiểm soát vào/ra

- Programmed I/O
- Interrupt driven I/O
- Direct Memory Access – DMA
- Kênh vào/ra & CPU

4. Interfacing

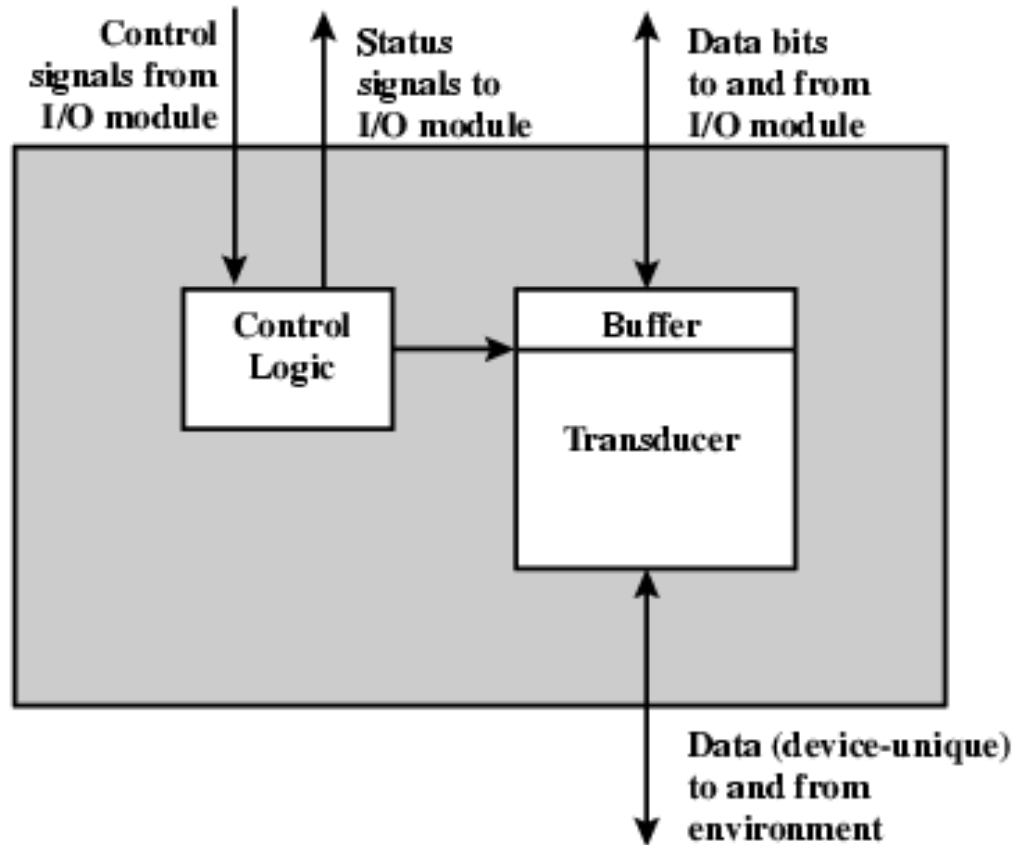
1. Quản lý vào/ra (I/O)

- Rất nhiều thiết bị ngoại vi – peripherals, dữ liệu thường có các đặc điểm:
 - Dung lượng khác nhau
 - Tốc độ khác nhau
 - Định dạng khác nhau
- Tốc độ xử lý chậm hơn nhiều so với CPU và MM
- Cần phải có các khối vào/ra



Thiết bị ngoại vi

- Human readable
 - Screen, printer, keyboard
- Machine readable
 - Monitoring and control
- Communication
 - Modem
 - Network Interface Card (NIC)





2. Chức năng của I/O Module

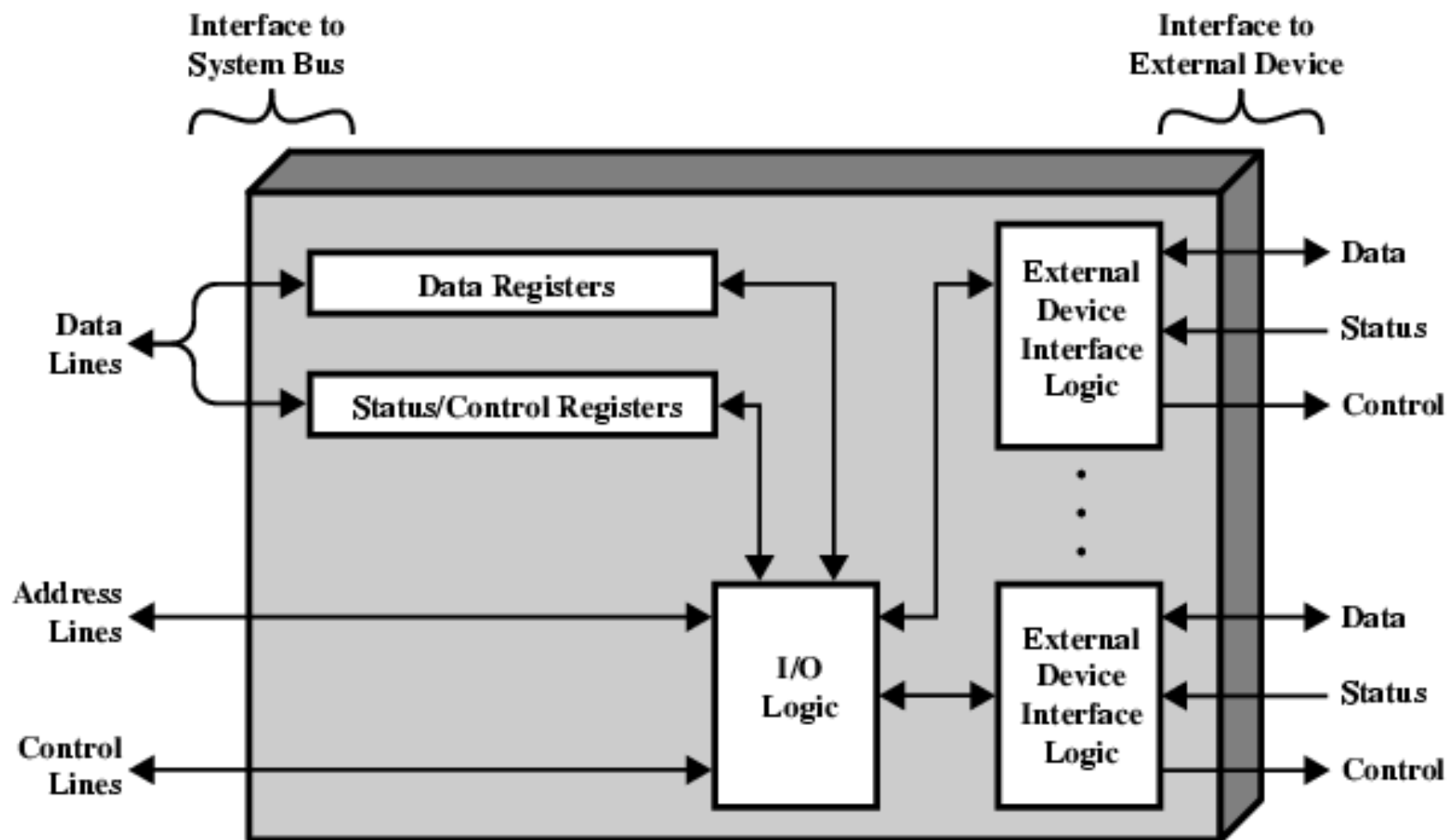
- Kiểm soát và điều phối (timing)
- Truyền thông với CPU / Main Memory
- Truyền thông với thiết bị ngoại vi
- Tạo cơ chế đệm dữ liệu (Data Buffering)
- Phát hiện và kiểm soát lỗi



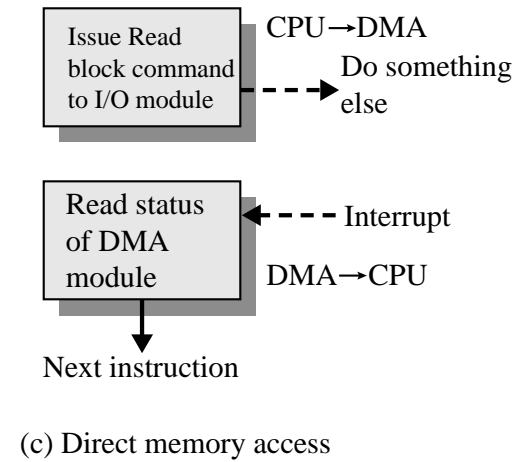
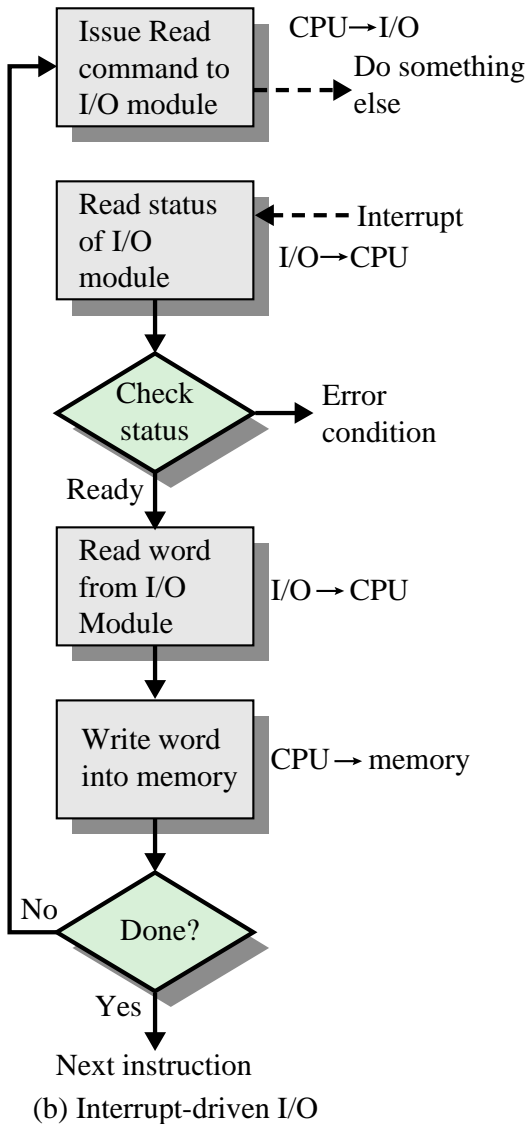
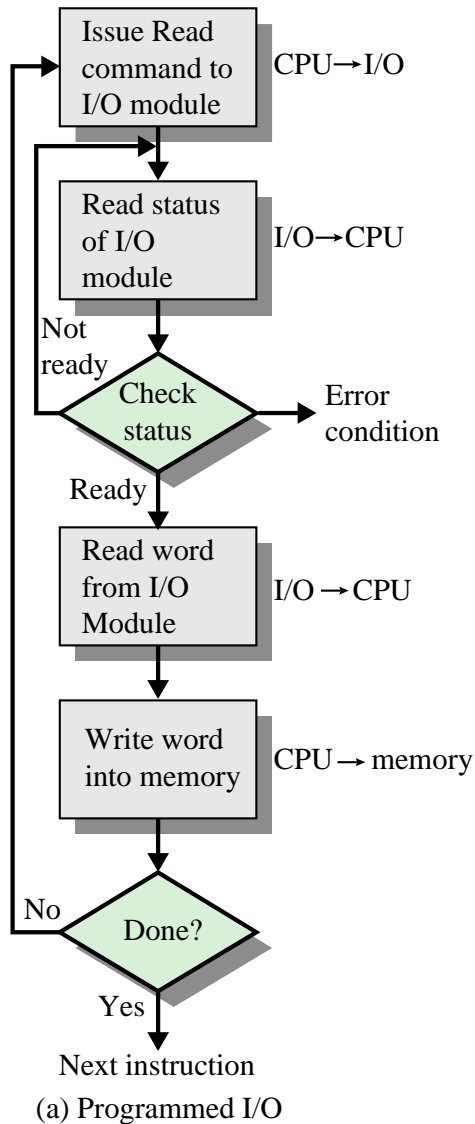
Quy trình hoạt động mô đun I/O

1. CPU kiểm tra trạng thái của thiết bị được kiểm soát bởi mô đun I/O
2. Mô đun I/O trả kết quả trạng thái của thiết bị I/O đó
3. Nếu sẵn sàng, CPU tiến hành truyền/nhận dữ liệu
4. Mô đun I/O lấy dữ liệu từ thiết bị
5. Mô đun I/O truyền dữ liệu đến CPU

Ngoài ra còn có thêm những kỹ thuật truyền dữ liệu khác nữa như DMA, etc.



3. Kỹ thuật kiểm soát vào/ra





i. Programmed I/O

- Idea: CPU kiểm soát trực tiếp các thiết bị ngoại vi
 - Phát hiện thay đổi (sensing status)
 - Gửi các lệnh read/write
 - Truyền dữ liệu
- ➔ CPU phải đợi các I/O module hoàn tất các thao tác
- ➔ lãng phí tài nguyên CPU

Các bước thực hiện

- CPU requests I/O operation
- I/O module performs operation
- I/O module sets status bits
- CPU checks status bits periodically
- I/O module does not inform CPU directly
- I/O module does not interrupt CPU
- CPU may wait or come back later

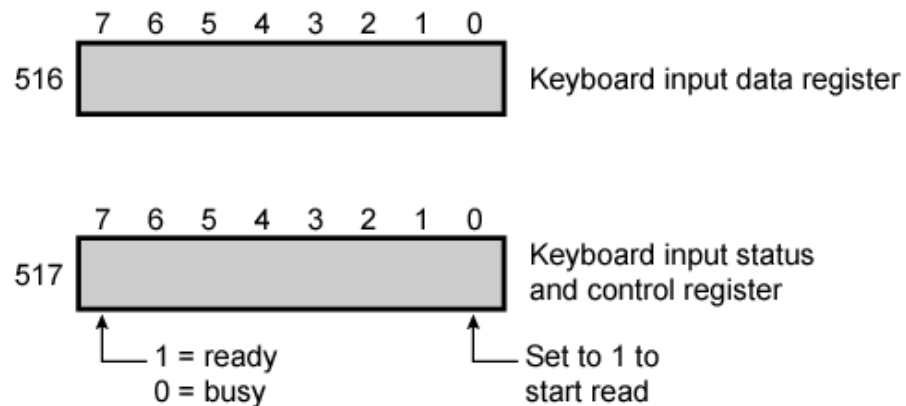


I/O Commands

- CPU gửi địa chỉ
 - Xác định bộ vào/ra (& thiết bị nếu >1 mỗi bộ)
- CPU gửi command
 - Control : yêu cầu module thực hiện thao tác
 - Ví dụ: quay đĩa, di chuyển đầu từ, ...
 - Test – kiểm tra trạng thái thiết bị
 - Ví dụ: đã được cung cấp nguồn? Có lỗi hay không?...
 - Read/Write
 - Module truyền dữ liệu sử dụng buffer từ/đến thiết bị

Đánh địa chỉ với các thiết bị I/O

- Với kỹ thuật *programmed I/O*, được được truyền giống như việc truy cập bộ nhớ chính (CPU viewpoint)
- Mỗi thiết bị có một định danh duy nhất (port ID)
- Các lệnh từ CPU sẽ chứa định danh này
- Phân loại
 - I/O được ánh xạ vào MM: các thiết bị có địa chỉ nằm trong không gian địa chỉ của MM
 - Các thao tác với I/O tương tự như đọc/ghi bộ nhớ,
 - Không cần lệnh đặc biệt
 - I/O độc lập so với MM: các thiết bị có địa chỉ độc lập so với không gian địa chỉ MM
 - Cần cơ chế liên kết riêng cho I/O
 - Cần có lệnh riêng thao tác với I/O



ADDRESS	INSTRUCTION	OPERAND	COMMENT
200	Load AC	"1"	Load accumulator
	Store AC	517	Initiate keyboard read
202	Load AC	517	Get status byte
	Branch if Sign = 0	202	Loop until ready
	Load AC	516	Load data byte

(a) Memory-mapped I/O

ADDRESS	INSTRUCTION	OPERAND	COMMENT
200	Load I/O	5	Initiate keyboard read
201	Test I/O	5	Check for completion
	Branch Not Ready	201	Loop until complete
	In	5	Load data byte

(b) Isolated I/O



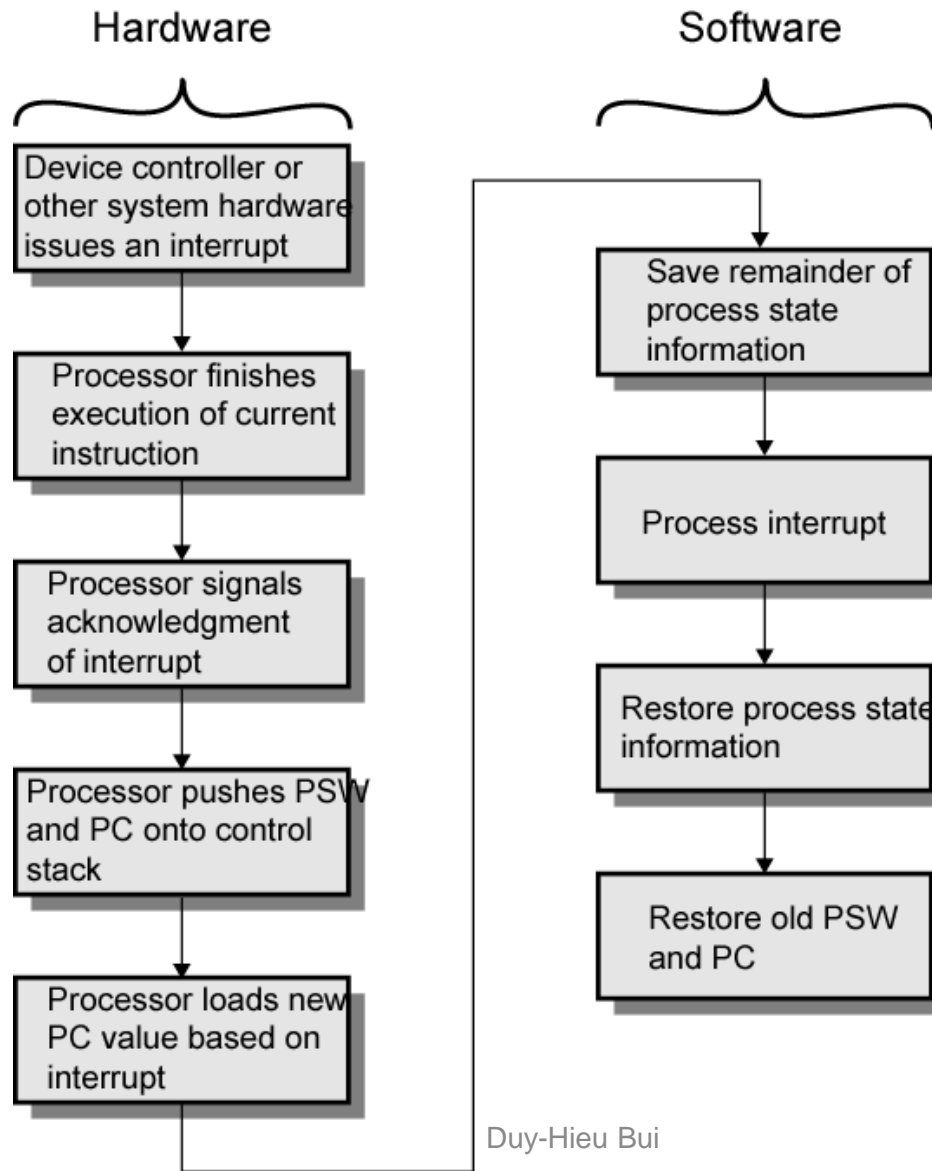
ii. Interrupt Driven I/O

- Tránh được sự lãng phí CPU
- CPU không cần phải kiểm tra định kỳ trạng thái thiết bị
- Bộ vào/ra sinh ngắt khi đã sẵn sàng

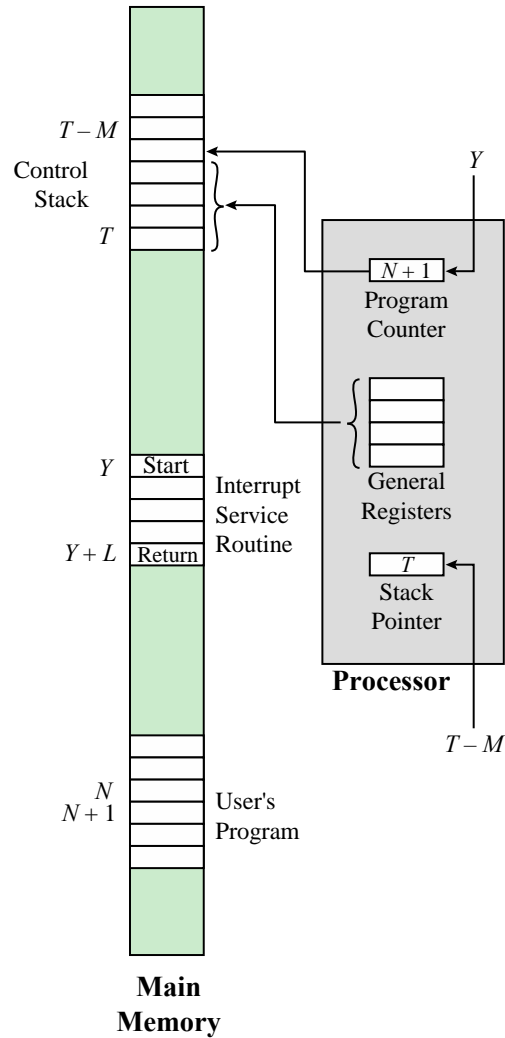
Các bước thực hiện

- CPU issues read command
- I/O module gets data from peripheral whilst CPU does other work
- I/O module interrupts CPU
- CPU requests data
- I/O module transfers data

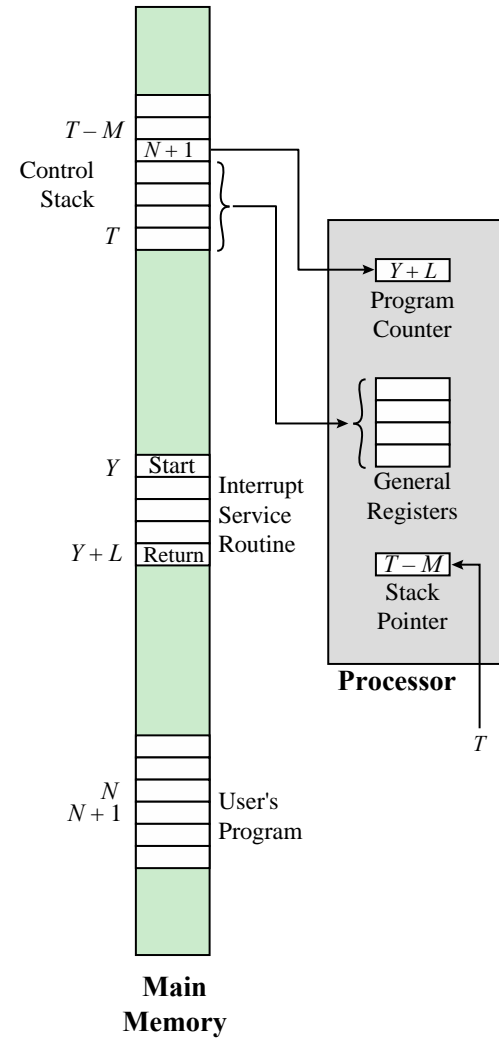
Sơ đồ xử lý ngắt



Thay đổi trong MM và Registers khi xử lý ngắt



(a) Interrupt occurs after instruction at location N



(b) Return from interrupt



Vấn đề cần quan tâm

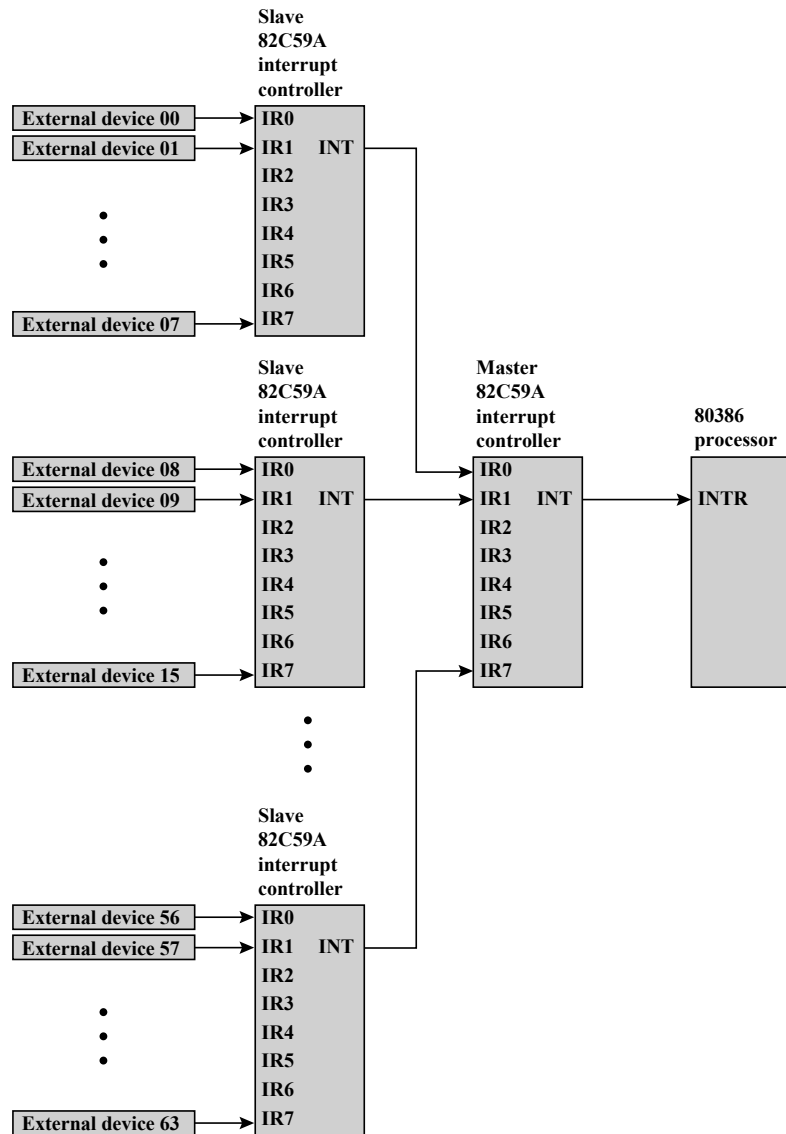
- Làm thế nào để xác định được module nào yêu cầu ngắt?
- Việc xử lý đa ngắt được thực hiện như thế nào?

Xác định module ngắt

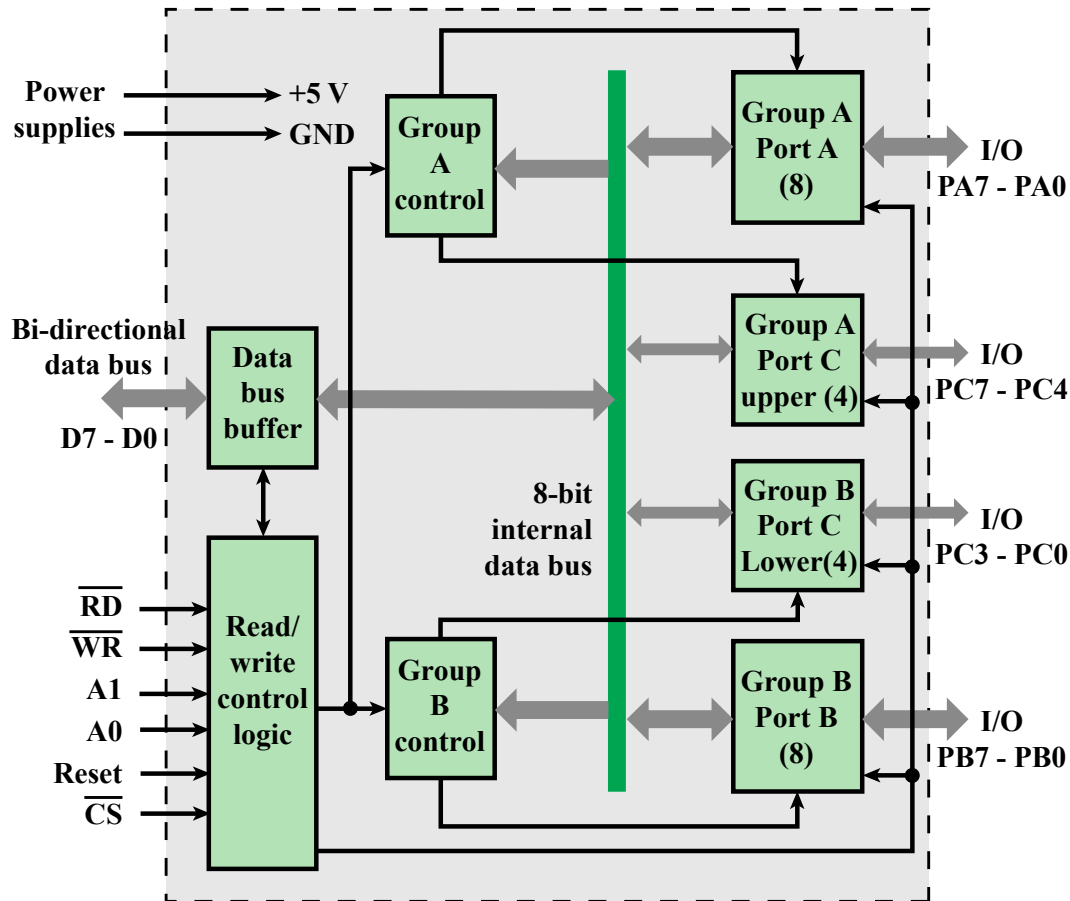
- Sử dụng nhiều đường ngắt (multiple interrupt lines)
 - Số pins của CPU là giới hạn → số lượng thiết bị cũng bị giới hạn
 - Không được sử dụng trong thực tế
- Software poll
 - Sử dụng chương trình con (routine) làm nhiệm vụ xác định thiết bị yêu cầu ngắt mỗi khi CPU nhận được yêu cầu ngắt
 - → mất nhiều chu kỳ cho việc xác định thiết bị yêu cầu ngắt
- Daisy Chain (hardware poll): dùng chung đường interrupt
 - CPU gửi lại tín hiệu *Interrupt Acknowledge* khi nhận được yêu cầu ngắt và sẽ được truyền lần lượt các thiết bị cho đến khi đến được thiết bị yêu cầu
 - Thiết bị yêu cầu ngắt sẽ gửi dữ liệu trên bus dưới dạng vector chứa địa chỉ
 - CPU sử dụng vector để định danh thiết bị
- Bus Master
 - I/O module phải đăng ký sử dụng bus trước khi gửi ngắt
 - Được sử dụng trong PCI & SCSI

- Multiple lines: Mỗi đường ngắt được gán một độ ưu tiên (priority), đường ngắt có độ ưu tiên cao hơn có thể ngắt thấp hơn
- Software polling/daisy chain: đánh độ ưu tiên thứ tự poll cho các modules
- Bus mastering: chỉ duy nhất thiết bị có quyền master được xử lý ngắt

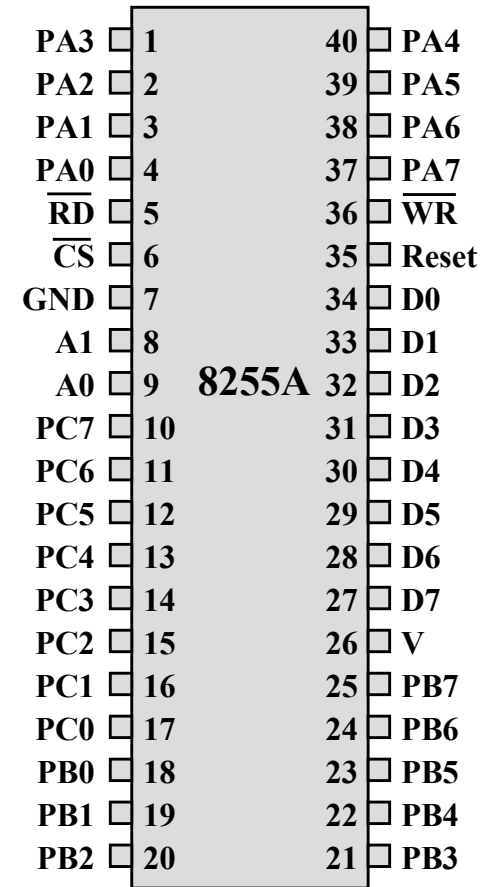
82C59A Interrupt controller



8255A programmable peripheral interface

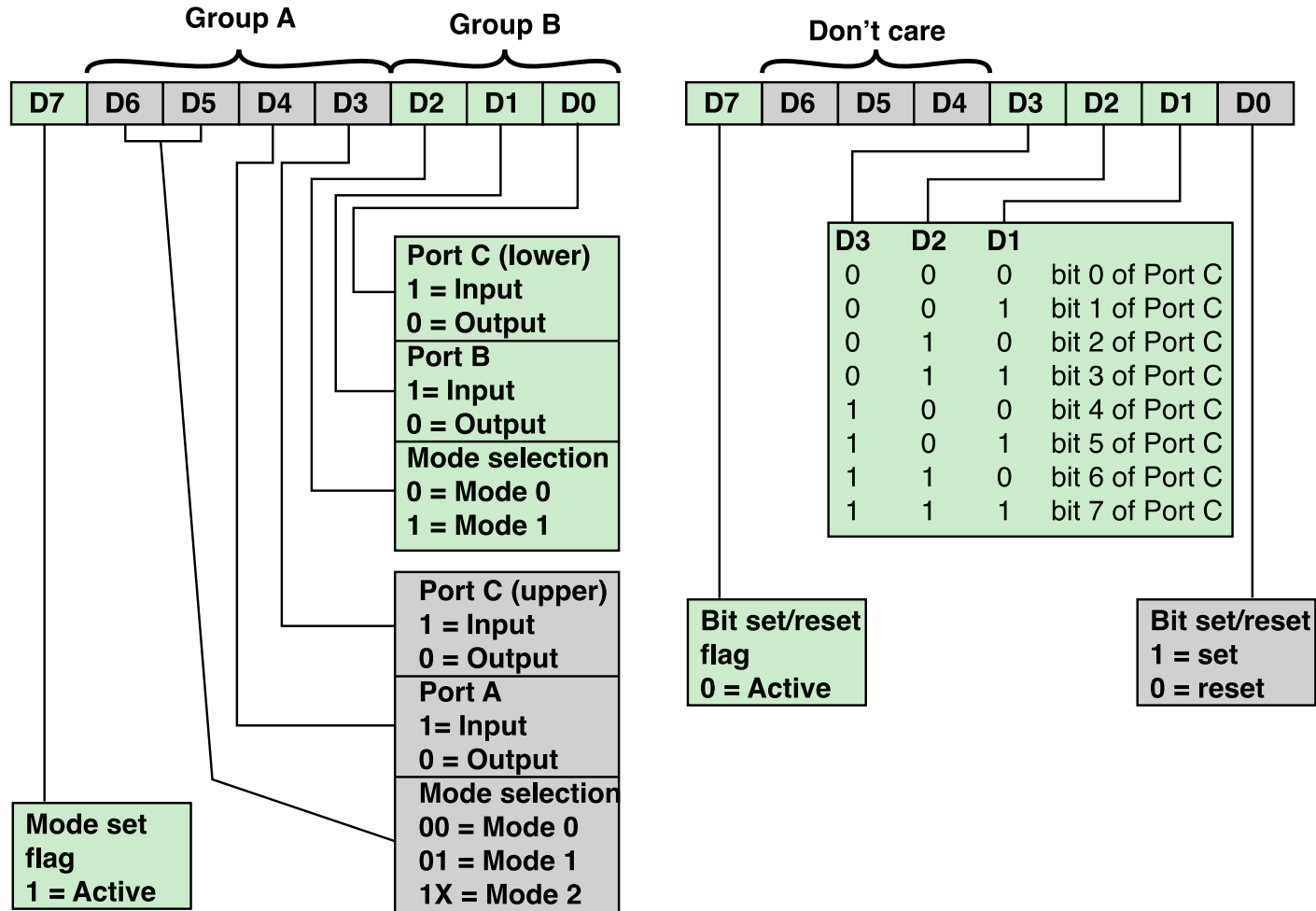


(a) Block diagram



(b) Pin layout

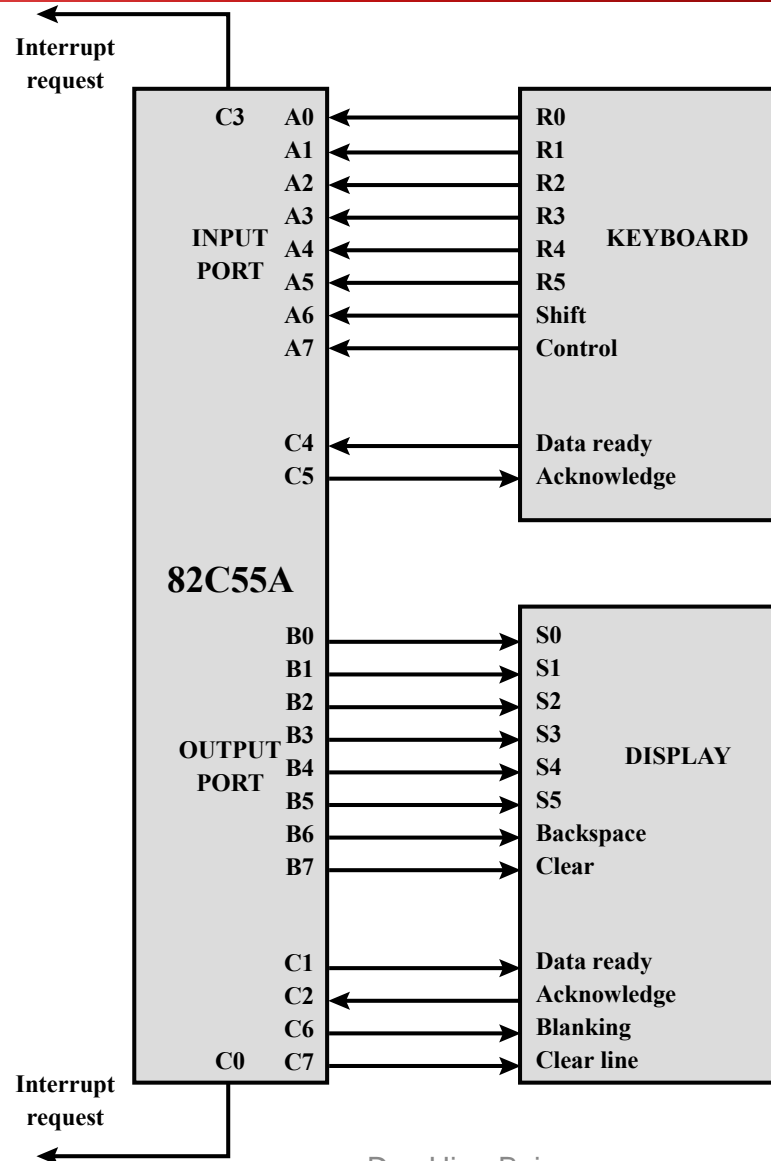
Intel 8255A control word



(a) Mode definition of the 8255 control register to configure the 8255

(b) Bit definitions of the 8255 control register to modify single bits of port C

Keyboard/Display interfacing with 82C55A





Drawbacks of Programmed and Interrupt-Driven I/O

- Both forms of I/O suffer from two inherent drawbacks:
 - 1) The I/O transfer rate is limited by the speed with which the processor can test and service a device
 - 2) The processor is tied up in managing an I/O transfer; a number of instructions must be executed for each I/O transfer

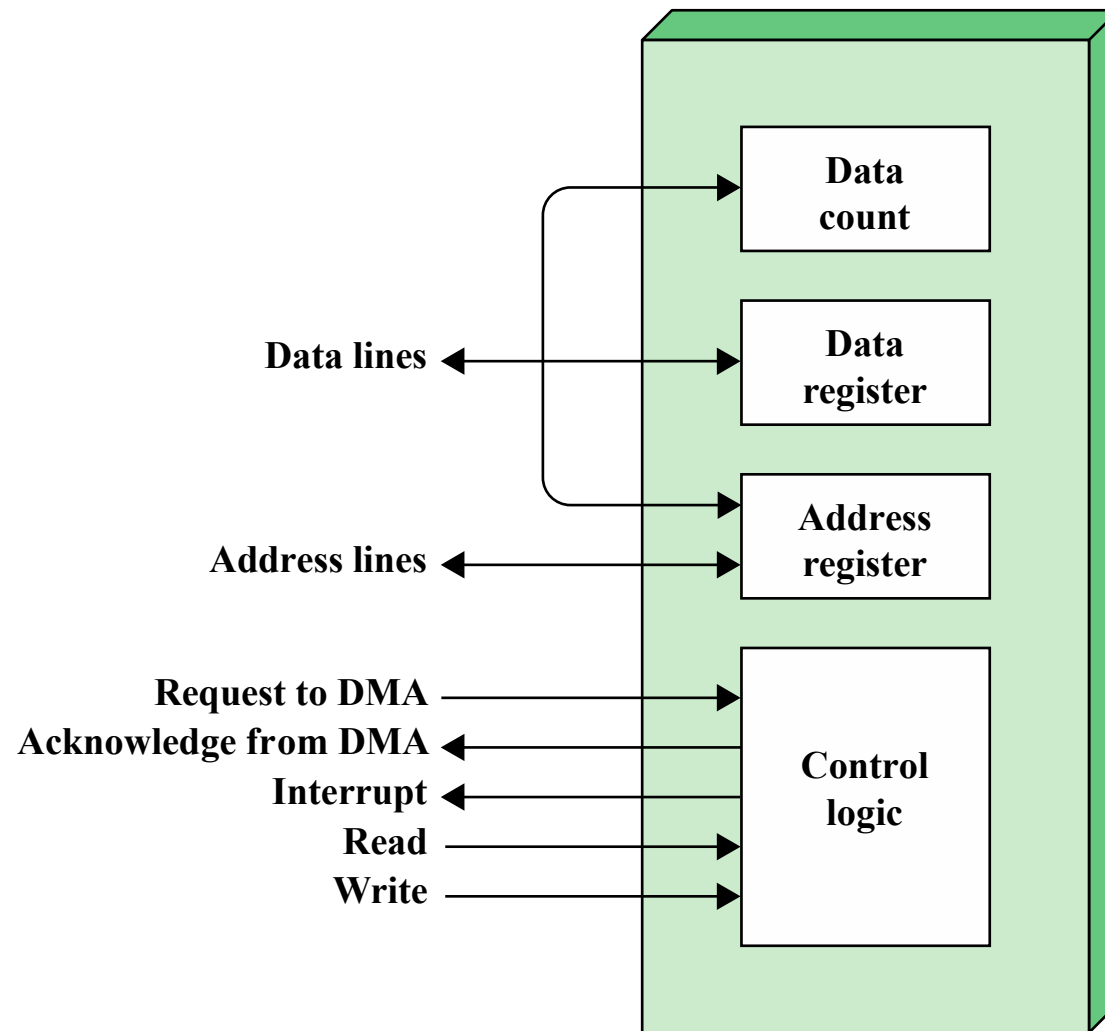
- When large volumes of data are to be moved a more efficient technique is *direct memory access* (DMA)



iii. Direct Memory Access

- Cả Interrupt-driven và programmed I/O đều cần sự “can thiệp” của CPU
 - ➔ Tốc độ truyền giữa các thiết bị sẽ bị giới hạn
 - ➔ CPU có thể bị quá tải
- DMA là kỹ thuật sẽ cho phép giải quyết được những nhược điểm nêu trên khi truyền khối lượng dữ liệu lớn
 - Sử dụng thêm module (hardware) DMA Controller
 - DMA controller giữ vai trò điều phối vào/ra

Sơ đồ bộ DMA điển hình



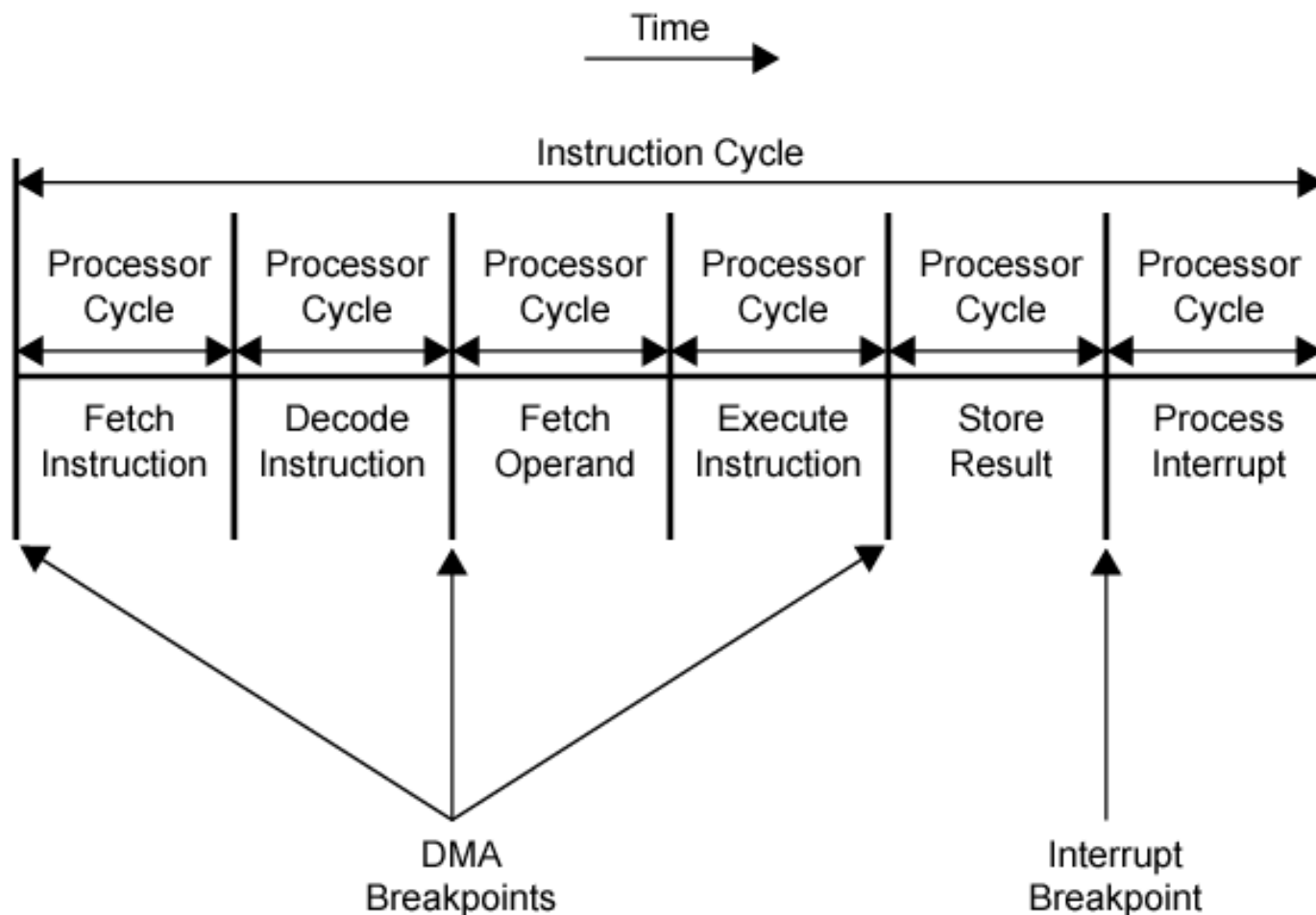


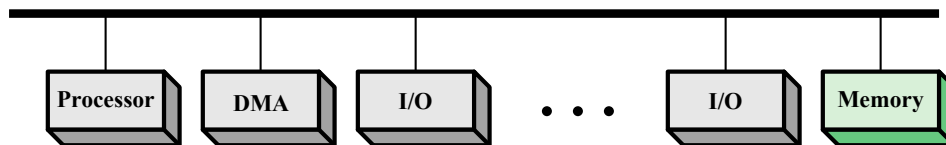
Quy trình hoạt động

- CPU gửi lệnh đến DMA controller:
 - Read/Write
 - Device address
 - Starting address of memory block for data
 - Amount of data to be transferred
- CPU tiến hành các lệnh khác
- DMA controller đảm nhiệm điều phối truyền dữ liệu
- DMA controller gửi interrupt khi kết thúc truyền

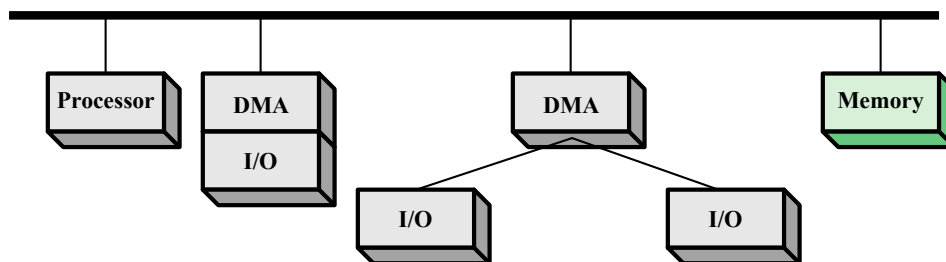
DMA truyền với kỹ thuật Cycle Stealing

- DMA controller tận dụng các cycles mà CPU không sử dụng data bus để truyền dữ liệu (one word/cycle)

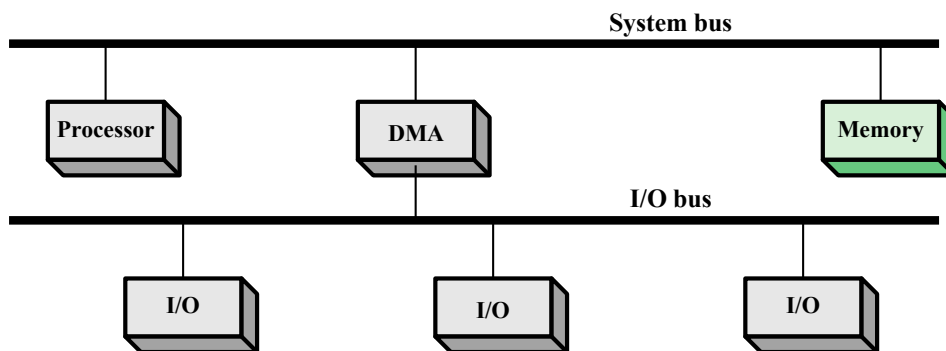




(a) Single-bus, detached DMA

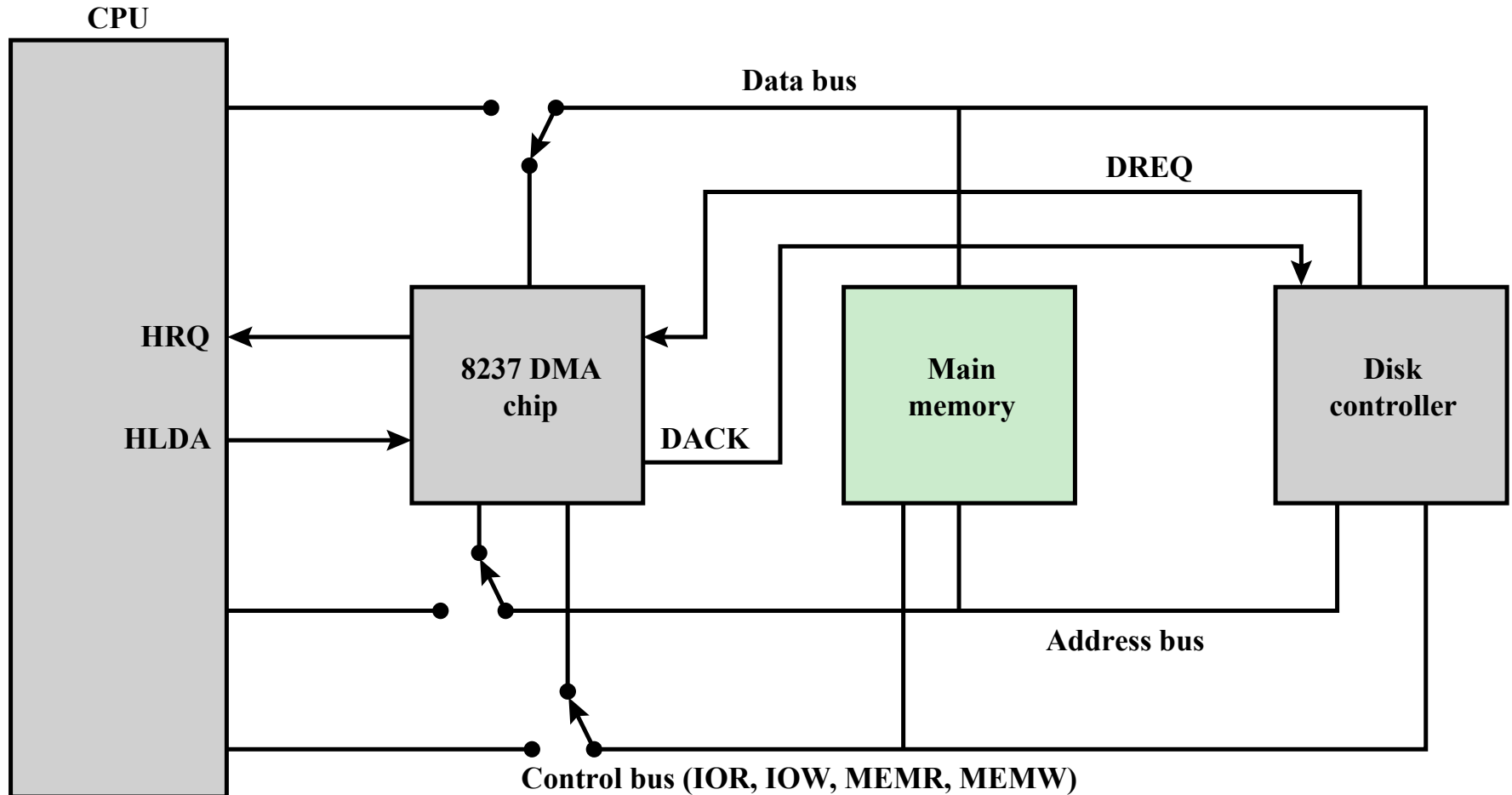


(b) Single-bus, Integrated DMA-I/O



(c) I/O bus

Ví dụ: Intel 8237A DMA Controller



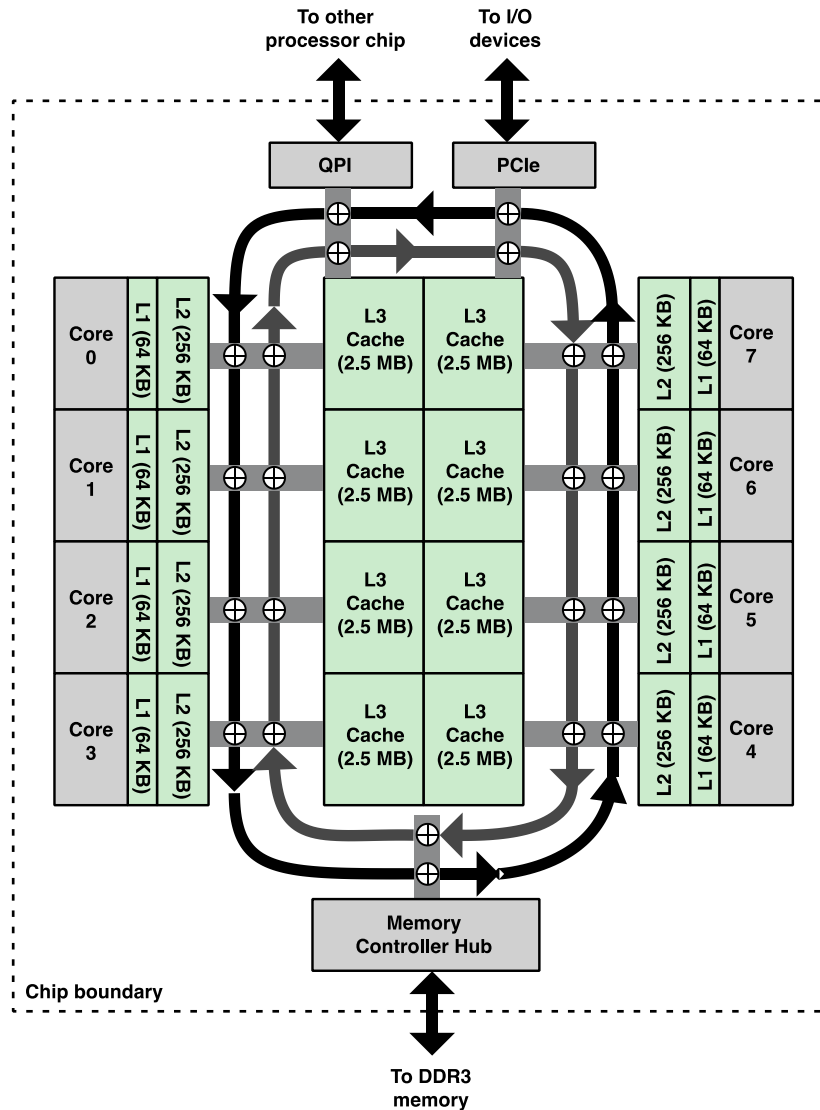
DACK = DMA acknowledge
DREQ = DMA request
HLDA = HOLD acknowledge
HRQ = HOLD request



Direct Cache Access (DCA)

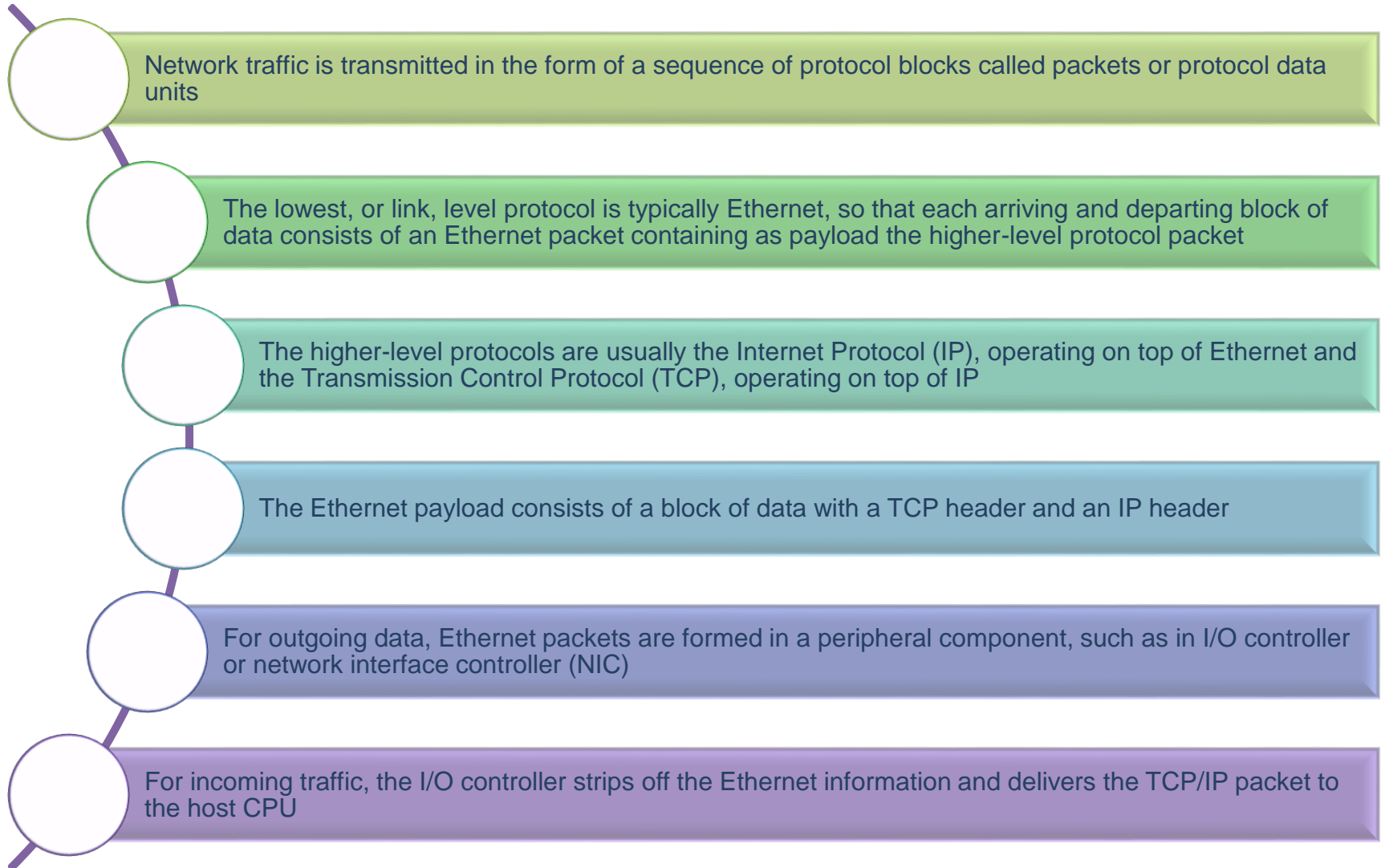
- DMA is not able to scale to meet the increased demand due to dramatic increases in data rates for network I/O
- Demand is coming primarily from the widespread deployment of 10-Gbps and 100-Gbps Ethernet switches to handle massive amounts of data transfer to and from database servers and other high-performance systems
- Another source of traffic comes from Wi-Fi in the gigabit range
- Network Wi-Fi devices that handle 3.2 Gbps and 6.76 Gbps are becoming widely available and producing demand on enterprise systems

Xeon E5-2600/4600 chip architecture





Cache-Related Performance Issues





Cache-Related Performance Issues

For both outgoing and incoming traffic the core, main memory, and cache are all involved

In a DMA scheme, when an application wishes to transmit data, it places that data in an application-assigned buffer in main memory

- The core transfers this to a system buffer in main memory and creates the necessary TCP and IP headers, which are also buffered in system memory
- The packet is then picked up via DMA for transfer via the NIC
- This activity engages not only main memory but also the cache
- Similar transfers between system and application buffers are required for incoming traffic



Packet Traffic Steps:

Incoming

- Packet arrives
- DMA
- NIC interrupts host
- Retrieve descriptors and headers
- Cache miss occurs
- Header is processed
- Payload transferred

Outgoing

- Packet transfer requested
- Packet created
- Output operation invoked
- DMA transfer
- NIC signals completion
- Driver frees buffer

Direct Cache Access Strategies

Simplest strategy was implemented as a prototype on a number of Intel Xeon processors between 2006 and 2010

This form of DCA applies only to incoming network traffic

The DCA function in the memory controller sends a prefetch hint to the core as soon as the data is available in system memory

This enables the core to prefetch the data packet from the system buffer



Much more substantial gains can be realized by avoiding the system buffer in main memory altogether

The packet and packet descriptor information are accessed only once in the system buffer by the core

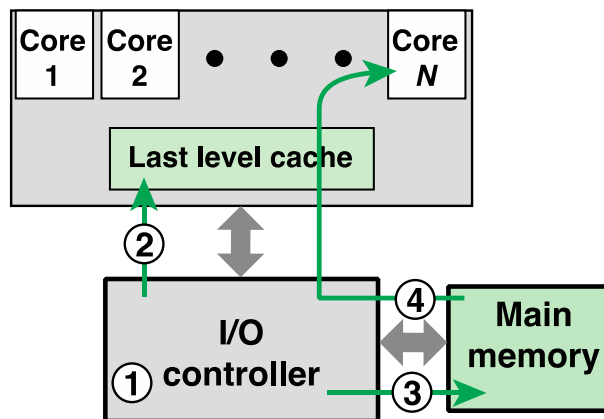
For incoming packets, the core reads the data from the buffer and transfers the packet payload to an application buffer

It has no need to access that data in the system buffer again

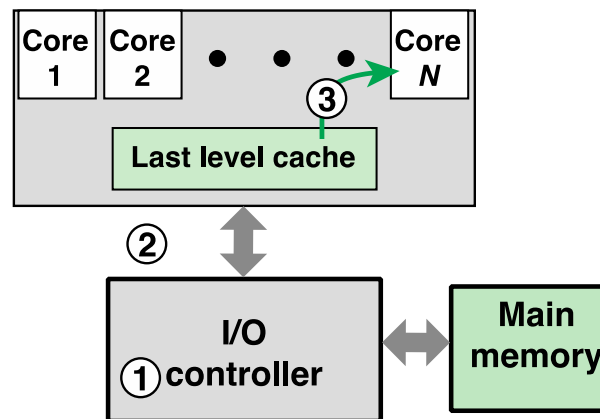
Cache injection

Implemented in Intel's Xeon processor line, referred to as Direct Data I/O

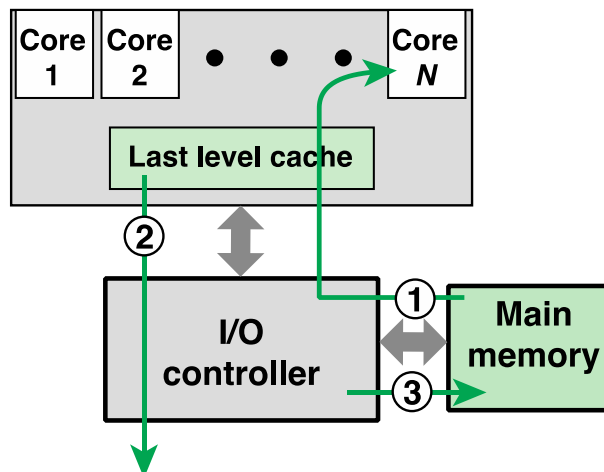
Direct memory access for Multicore



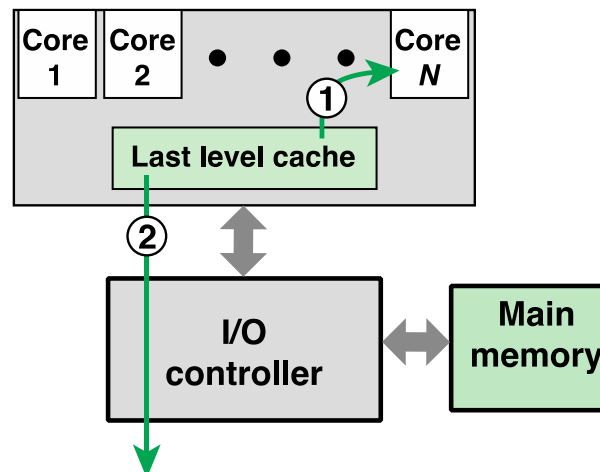
(a) Normal DMA transfer to memory



(b) DDIO transfer to cache



(c) Normal DMA transfer to I/O



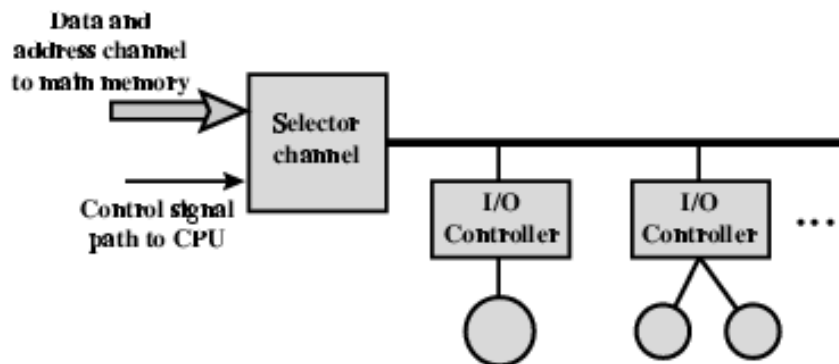
(d) DDIO transfer to I/O



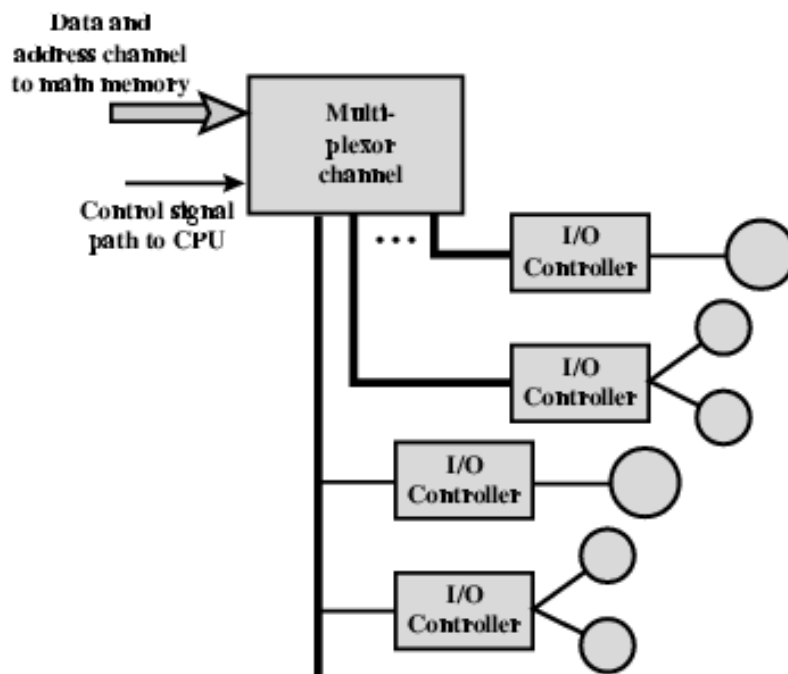
iv. I/O Channels

- Thiết bị I/O càng ngày càng phức tạp, có xu hướng tích hợp luôn bộ vi xử lý riêng
 - VD: Card đồ họa (GPU), ...
 - CPU gửi lệnh đến I/O controller và giao toàn quyền việc truyền dữ liệu
- ➔ Cải thiện được tốc độ truyền dữ liệu I/O
- I/O channels: thể hiện mở rộng của kỹ thuật DMA

Kiến trúc kênh vào/ra



(a) Selector



(b) Multiplexor



4. Interfacing

- Giao tiếp giữa các thiết bị:
 - Serial <> Parallel
 - Dedicated processor/memory/buses?
- Một số chuẩn giao tiếp ngoại vi thông dụng:
 - USB
 - FireWire
 - InfiniBand
 - ...



Universal Serial Bus (USB)

- Được sử dụng rộng rãi để kết nối với thiết bị ngoại vi
- Đã có nhiều thế hệ được ứng dụng:
 - USB 1.0
 - Defined a Low Speed data rate of 1.5 Mbps and a Full Speed rate of 12 Mbps
 - USB 2.0
 - Provides a data rate of 480 Mbps
 - USB 3.0
 - Higher speed bus called SuperSpeed in parallel with the USB 2.0 bus
 - Signaling speed of SuperSpeed is 5 Gbps, but due to signaling overhead the usable data rate is up to 4 Gbps
 - USB 3.1
 - Includes a faster transfer mode called SuperSpeed+
 - This transfer mode achieves a signaling rate of 10 Gbps and a theoretical usable data rate of 9.7 Gbps
- Được kiểm soát bởi bộ điều khiển gốc (root host controller), thể hiện theo mô hình cây phân cấp các thiết bị kết nối

Small Computer System Interface

- Chuẩn được sử dụng để kết nối với các thiết bị ngoại vi đối với các máy tính cỡ nhỏ và vừa
- Hiện thường được thay thế bởi chuẩn USB hoặc FireWire đối với các máy tính cỡ nhỏ;
- Bản hiệu năng cao vẫn còn được sử dụng trong các hệ thống tính toán phục vụ doanh nghiệp cỡ vừa/lớn
- Về tổ chức, SCSI sử dụng hệ thống bus dùng chung, có thể hỗ trợ từ 16-32 thiết bị
 - The bus provides for parallel transmission rather than serial, with a bus width of 16 bits on earlier generations and 32 bits on later generations
 - Speeds range from 5 Mbps on the original SCSI-1 specification to 160 Mbps on SCSI-3 U3





FireWire Serial Bus

- Was developed as an alternative to small computer system interface (SCSI) to be used on smaller systems, such as personal computers, workstations, and servers
- Objective was to meet the increasing demands for high I/O rates while avoiding the bulky and expensive I/O channel technologies developed for mainframe and supercomputer systems
- IEEE standard 1394, for a High Performance Serial Bus
- Uses a daisy chain configuration, with up to 63 devices connected off a single port
- 1022 FireWire buses can be interconnected using bridges
- Provides for hot plugging which makes it possible to connect and disconnect peripherals without having to power the computer system down or reconfigure the system
- Provides for automatic configuration
- No terminations and the system automatically performs a configuration function to assign addresses

- Provides up to 10 Gbps throughput in each direction and up to 10 Watts of power to connected peripherals
- Most recent and fastest peripheral connection technology to become available for general-purpose use
- Developed by Intel with collaboration from Apple
- The technology combines data, video, audio, and power into a single high-speed connection for peripherals such as hard drives, RAID arrays, video-capture boxes, and network interfaces



Thunderbolt™ 3

The USB-C that does it all



More Speed



40 Gbps

More Pixels



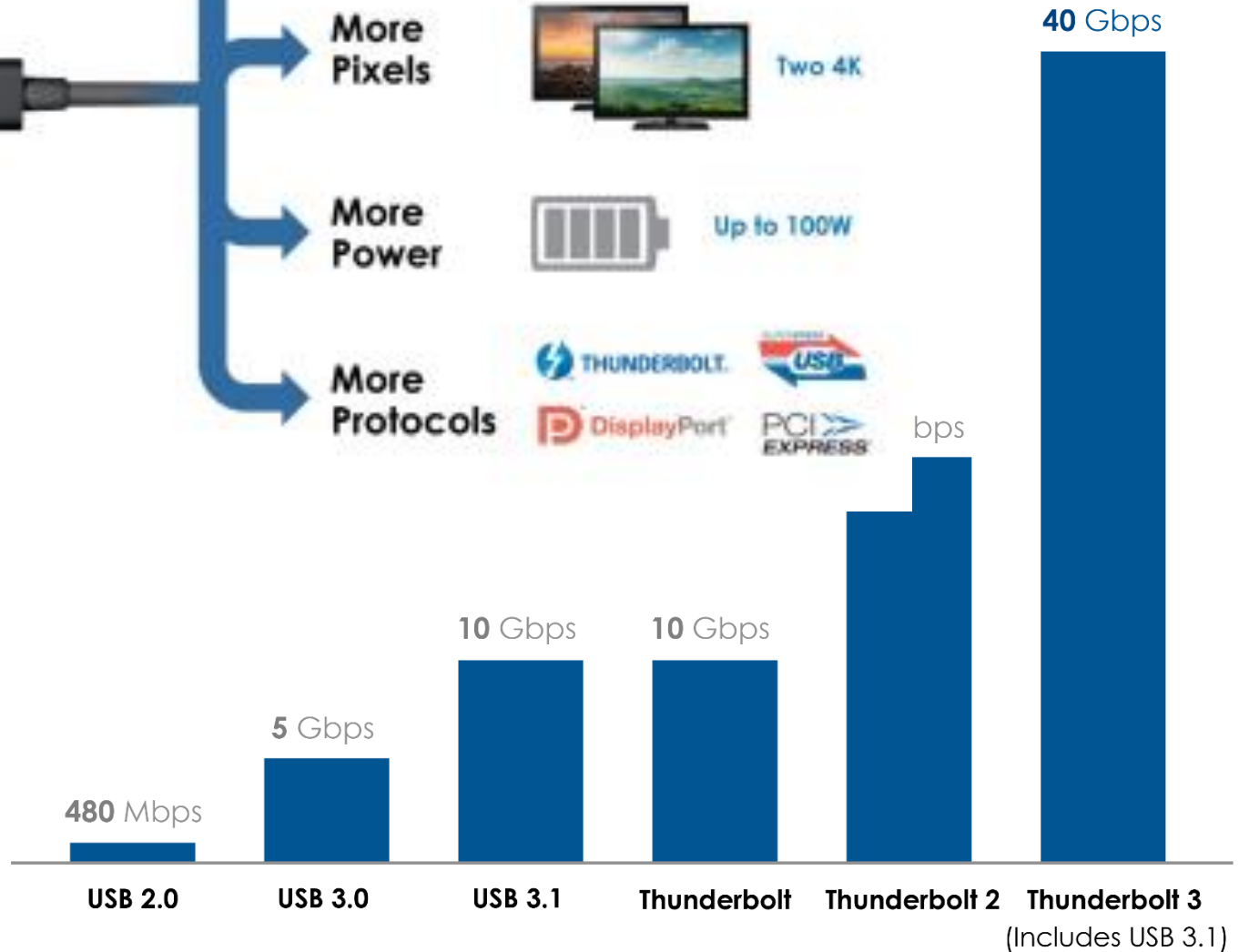
Two 4K

More Power



Up to 100W

More Protocols





InfiniBand

- I/O specification aimed at the high-end server market
- First version was released in early 2001
- Heavily relied on by IBM zEnterprise series of mainframes
- Standard describes an architecture and specifications for data flow among processors and intelligent I/O devices
- Has become a popular interface for storage area networking and other large storage configurations
- Enables servers, remote storage, and other network devices to be attached in a central fabric of switches and links
- The switch-based architecture can connect up to 64,000 servers, storage systems, and networking devices
- Speeds up to 120 Gbit/s





PCI EXPRESS®



- High-speed bus system for connecting peripherals of a wide variety of types and speeds
- 2017: PCI Express 5.0, Bandwidth is expected to increase to 32 GT/s, yielding 128 GB/s in full duplex networking configurations



- Serial Advanced Technology Attachment; interface for disk storage systems
- SATA3: data rates of up to 6 Gbps (600MB/s), max per device of 300 Mbps
- 2013: SATA3.2 – SATA Express, 16 Gbit/s - 1969 MB/s
- Widely used in desktop computers and in industrial and embedded applications



Ethernet

- Predominant wired networking technology
 - Has evolved to support data rates up to 100 Gbps and distances from a few meters to tens of km
 - Has become essential for supporting personal computers, workstations, servers, and massive data storage devices in organizations large and small
 - Began as an experimental bus-based 3-Mbps system
- Has moved from bus-based to switch-based
 - Data rate has periodically increased by an order of magnitude
 - There is a central switch with all of the devices connected directly to the switch
 - Ethernet systems are currently available at speeds up to 100 Gbps



- Is the predominant wireless Internet access technology
- Now connects computers, tablets, smart phones, and other electronic devices such as video cameras TVs and thermostats
- In the enterprise has become an essential means of enhancing worker productivity and network effectiveness
- Public hotspots have expanded dramatically to provide free Internet access in most public places
- As the technology of antennas, wireless transmission techniques, and wireless protocol design has evolved, the IEEE 802.11 committee has been able to introduce standards for new versions of Wi-Fi at higher speeds
 - Current version is 802.11ac (2014) with a maximum data rate of 3.2 Gbps



- Kiến trúc vào/ra cho phép máy tính “giao tiếp” được với thế giới bên ngoài
- Có 3 kỹ thuật vào/ra chính: programmed I/O, interrupt driven I/O và DMA
- Giao tiếp với các thiết bị ngoài vi được thực hiện theo các chuẩn: USB, Firewire, SATA, PCIe, ...