



VIETNAM NATIONAL UNIVERSITY HANOI (VNU)
INFORMATION TECHNOLOGY INSTITUTE

Computer Architecture

Lecture 2: Basic Components

Duy-Hieu Bui, PhD

AIoT Research Laboratory

Email: hieubd@vnu.edu.vn

<https://duyhieubui.github.io>



Nội dung

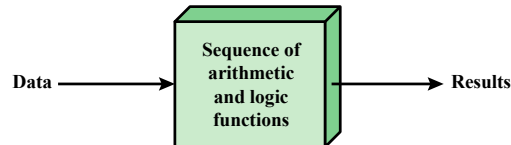
1. Nguyên tắc chung
2. Các thành phần cơ bản
3. Chức năng chính
4. Cơ chế ngắt và chu trình lệnh đầy đủ

Tham khảo chương 3 của “Computer Organization and Architecture: Designing for Performance”, William Stallings, 10th edition



1. Khái niệm chương trình

- Chương trình cứng hoá không cho phép thay đổi



Programming in hardware

- Phần cứng đa dụng: có thể làm nhiều tác vụ khác nhau, kiểm soát bằng các tín hiệu điều khiển
- Chương trình mềm: Thay vì thiết kế lại mạch - re-wiring, cung cấp một tập mới các tín hiệu điều khiển

9/11/2024

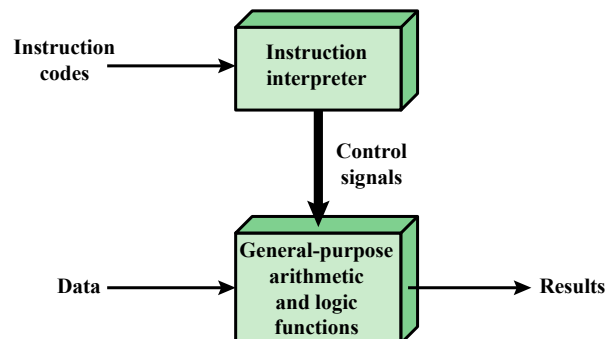
KTMT/INT2212

3



Khái niệm chương trình...

- Chuỗi các bước
- Với mỗi bước, một phép tính logic/số học được thực hiện
- Mỗi phép tính khác nhau được xác định bởi 1 tập các tín hiệu điều khiển khác nhau



Programming in software

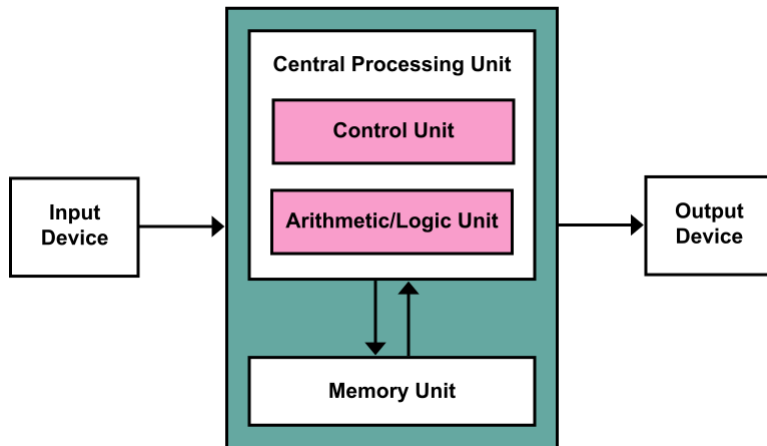
9/11/2024

KTMT/INT2212

4



Mô hình máy tính vận năng



9/11/2024

KTMT/INT2212

5



Nguyên tắc chung

1. Dữ liệu và chương trình (lệnh) được lưu giữ trên cùng bộ nhớ
2. Nội dung bộ nhớ xác định thông qua địa chỉ vị trí, không phụ thuộc vào kiểu dữ liệu đã lưu
3. Chương trình được thi hành tuần tự từ lệnh này đến lệnh kế tiếp (ngoại trừ những thay đổi cụ thể)

9/11/2024

KTMT/INT2212

6



Minh họa chương trình với MARS

The screenshot displays the MARS application window. The main window shows assembly code for a Fibonacci program. The 'Text Segment' pane lists instructions with their addresses and sources. The 'Data Segment' pane shows memory addresses and their corresponding values. The 'Registers' pane on the right lists MIPS registers and their current values. The 'Mars Messages' pane at the bottom shows the assembly process completion message.

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0x00000000
\$s2	18	0x00000000
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$t8	24	0x00000000
\$t9	25	0x00000000
\$d0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x7ffffffc
\$fp	30	0x00000000
\$ra	31	0x00000000
\$pc		0x00400000
\$hi		0x00000000
\$lo		0x00000000

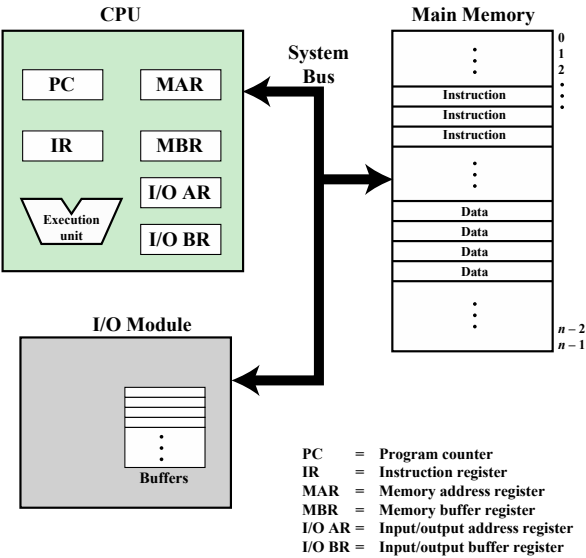


2. Thành phần máy tính

- Central Processing Unit: Control Unit & Arithmetic and Logic Unit
- Input/Output: dữ liệu và các lệnh cần phải được nạp vào hệ thống cũng như kết xuất kết quả
- Main memory: lưu trữ lệnh và dữ liệu tạm thời trong quá trình tính toán
- Hệ thống liên kết: kết nối, truyền thông các thành phần chính nêu trên



Thành phần: mức tổng quan



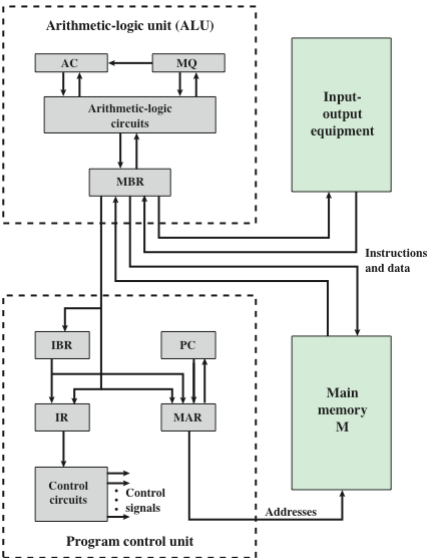
9/11/2024

KTMT/INT2212

9



Kiến trúc của máy tính IAS



9/11/2024

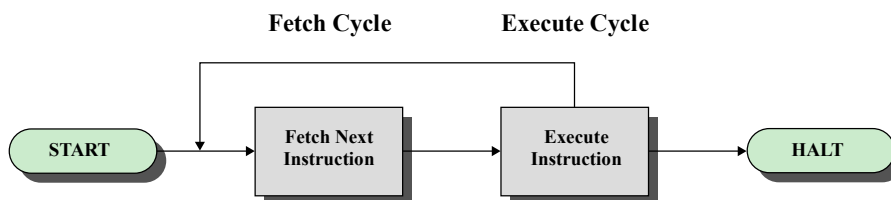
KTMT/INT2212

10



3. Chức năng máy tính

- Thực thi chương trình, đã được xây dựng thông qua tập các lệnh của CPU, lưu trong bộ nhớ
- Các bước chính khi thực thi chương trình trong CPU
 - Đọc lệnh từ bộ nhớ (fetch)
 - Thực thi lệnh (execute)
 - Lưu kết quả trong bộ nhớ (store)
- Chu trình lệnh đơn giản



9/11/2024

KTMT/INT2212

11



Chu trình fetch

- Program Counter (PC) lưu địa chỉ lệnh kế tiếp sẽ được tải lên
- Processor tải lệnh từ bộ nhớ xác định bởi địa chỉ lưu trong PC
- Tăng giá trị PC (ngoại trừ những trường hợp tự xác định)
- Lệnh được tải lên thành ghi IR - Instruction Register

9/11/2024

KTMT/INT2212

12



Chu trình thực thi

- Processor dịch lệnh và thực thi các phép toán tương ứng

Chu trình thực thi bao gồm những kiểu sau:

- Chuyển dữ liệu
 - Processor-memory (data transfer between CPU and main memory)
 - Processor - I/O (Data transfer between CPU and I/O module)
- Xử lý dữ liệu - Data processing
 - Some arithmetic or logical operation on data
- Điều khiển - Control
 - Thi hành các lệnh của CPU thông qua việc biến đổi thành chuỗi các thao tác logic cơ bản

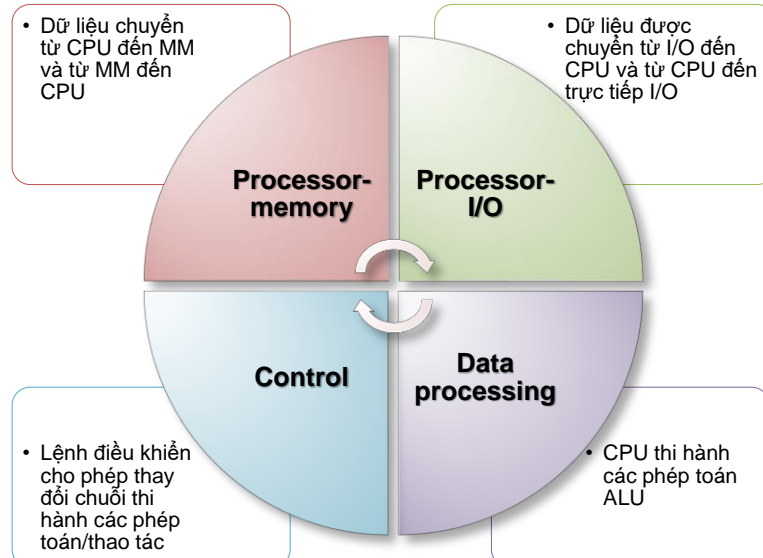
9/11/2024

KTMT/INT2212

13



Các loại thao tác



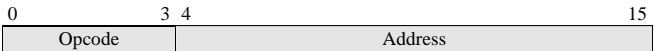
9/11/2024

KTMT/INT2212

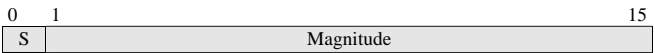
14



Máy tính đơn giản



(a) Instruction format



(b) Integer format

Program Counter (PC) = Address of instruction
Instruction Register (IR) = Instruction being executed
Accumulator (AC) = Temporary storage

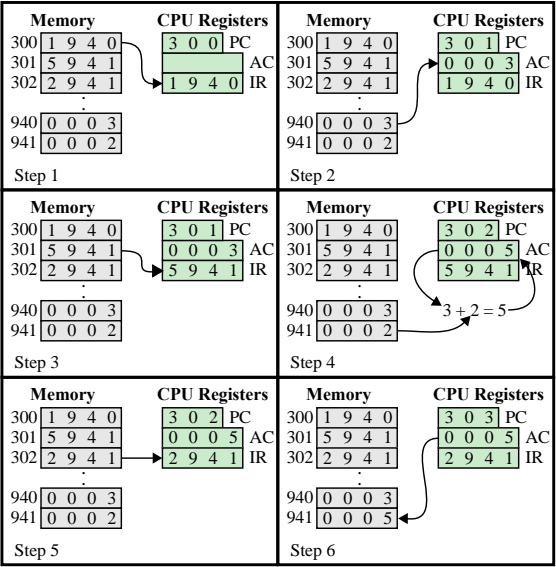
(c) Internal CPU registers

- 0001 = Load AC from Memory
- 0010 = Store AC to Memory
- 0101 = Add to AC from Memory

(d) Partial list of opcodes

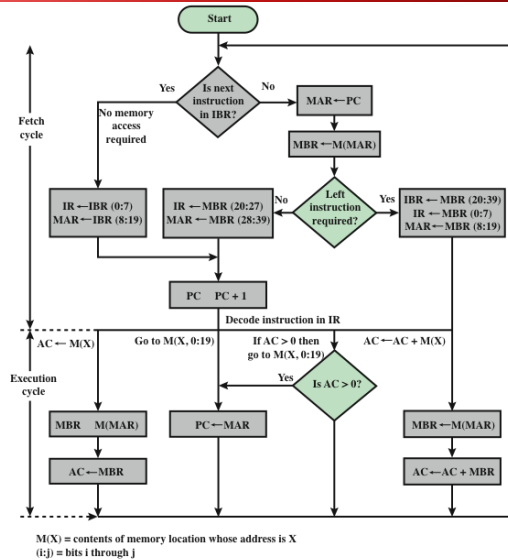


Ví dụ





Luồng điều khiển của máy tính IAS



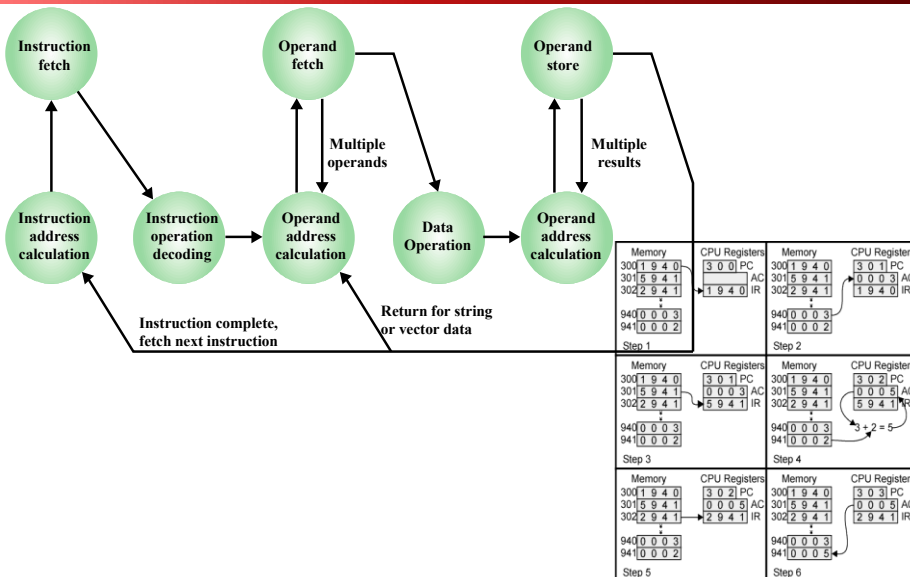
9/11/2024

KTMT/INT2212

17



4. Sơ đồ trạng thái chu trình lệnh



9/11/2024

KTMT/INT2212



Interrupts

- Cơ chế cho phép các modules khác (I/O) có thể tạm dừng chuỗi xử lý hiện hành (normal sequence of processing) để thực thi các lệnh của các modules đó
- Các kiểu ngắt thông dụng
 - Program
 - e.g. overflow, division by zero
 - Timer
 - Generated by internal processor timer
 - Used in pre-emptive multi-tasking
 - I/O
 - from I/O controller
 - Hardware failure
 - e.g. memory parity error

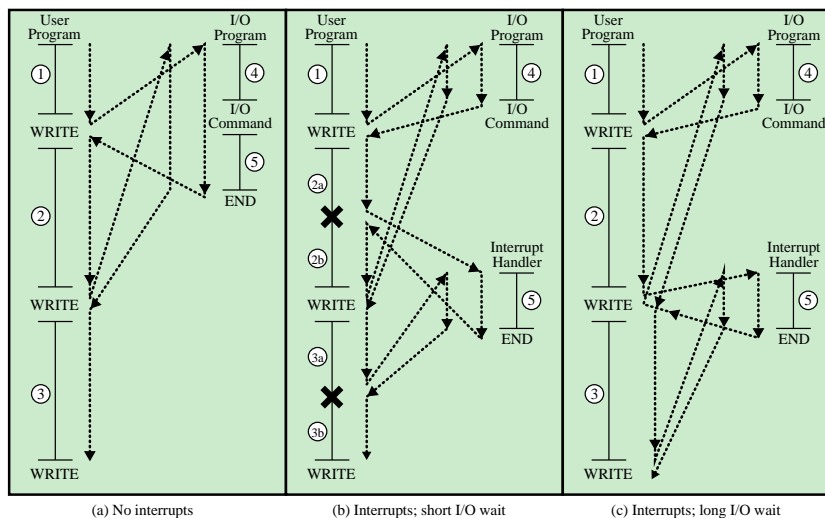
9/11/2024

KTMT/INT2212

19



Ví dụ



✗ = interrupt occurs during course of execution of user program

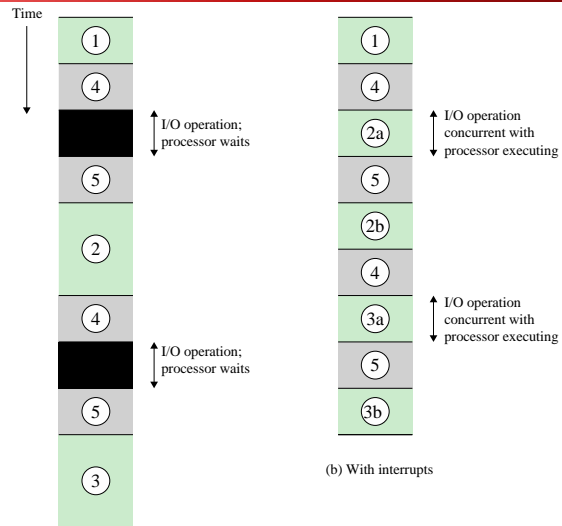
9/11/2024

KTMT/INT2212

20



Ví dụ 1



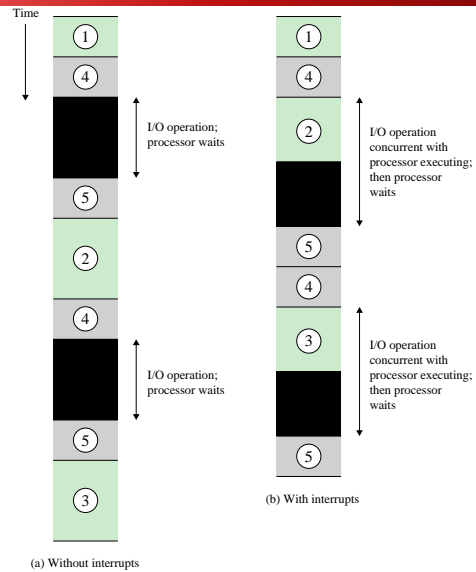
9/11/2024

KTMT/INT2212

21



Ví dụ 2



9/11/2024

KTMT/INT2212

22



Chu trình ngắt

- Được thêm vào sơ đồ chu trình lệnh để xử lý các yêu cầu ngắt
- Quy trình
 - Processor checks for interrupt
 - Indicated by an interrupt signal
 - If no interrupt, fetch next instruction
 - If interrupt pending:
 - Suspend execution of current program
 - Save context
 - Set PC to start address of interrupt handler routine
 - Process interrupt
 - Restore context and continue interrupted program

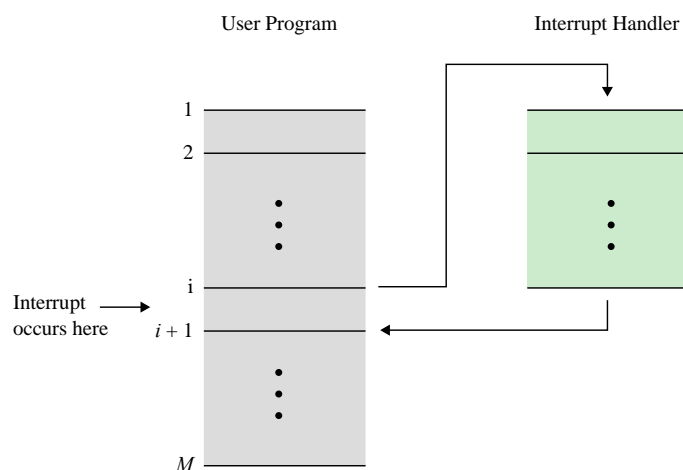
9/11/2024

KTMT/INT2212

23



Ví dụ



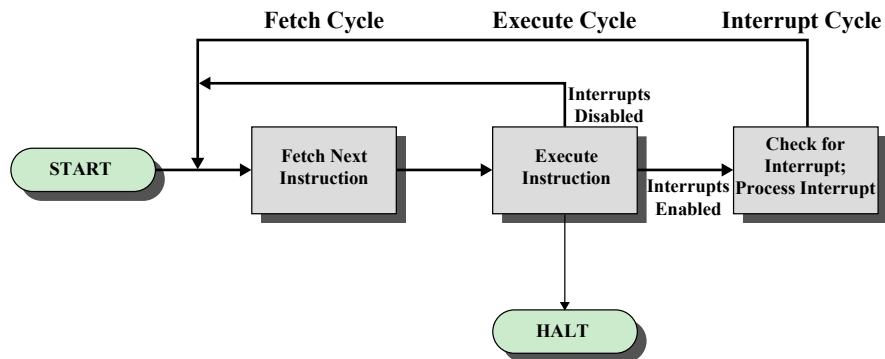
9/11/2024

KTMT/INT2212

24



Chu trình lệnh có ngắt



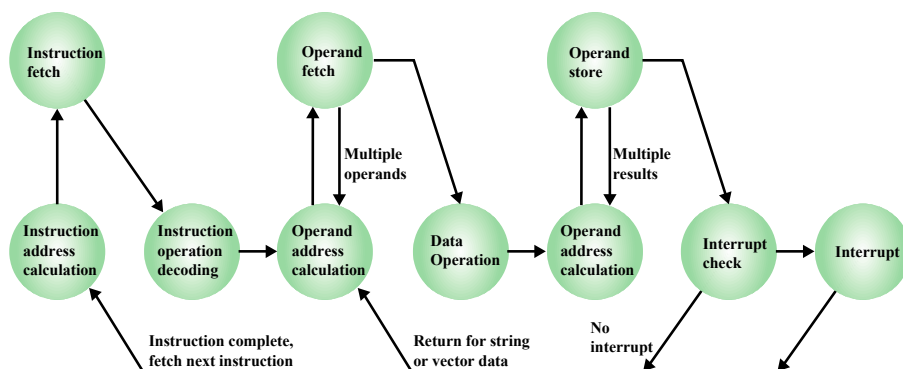
9/11/2024

KTMT/INT2212

25



Sơ đồ trạng thái chu trình lệnh



9/11/2024

KTMT/INT2212

26



Đa ngắt - Multiple Interrupts

- **Disable interrupts**
 - Trong khi xử lý 1 ngắt, processor sẽ tạm thời bỏ tất qua các ngắt còn lại
 - Các ngắt còn lại sẽ ở trạng thái treo và được xử lý khi ngắt hiện thời đã được xử lý xong
 - Các ngắt được xử lý theo thứ tự yêu cầu
- **Define priorities**
 - Các ngắt có độ ưu tiên thấp có thể bị ngắt bởi các ngắt có độ ưu tiên cao hơn
 - Khi ngắt có độ ưu tiên cao hơn được xử lý xong, processor sẽ quay lại xử lý tiếp ngắt trước đó

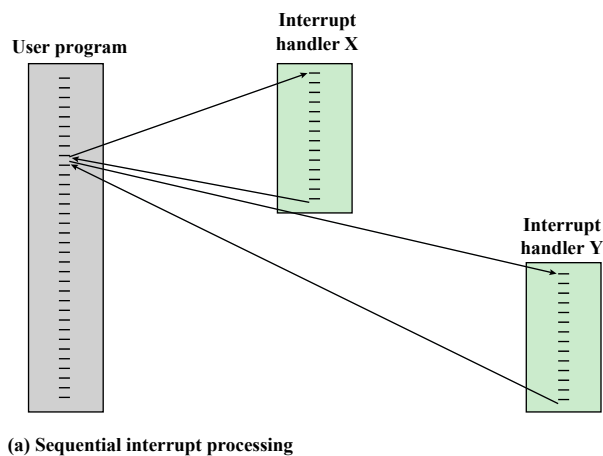
9/11/2024

KTMT/INT2212

27



Đa ngắt tuần tự



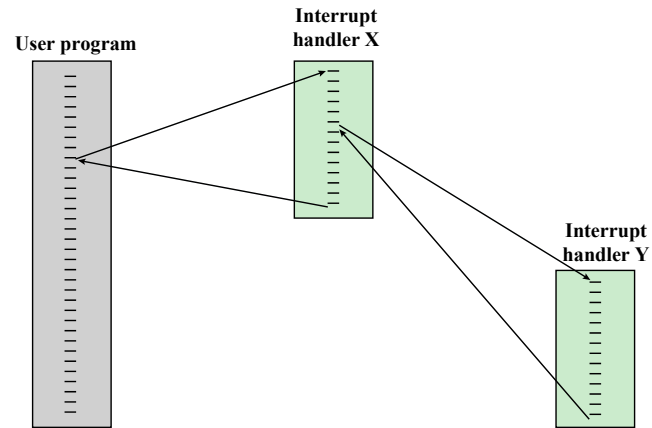
9/11/2024

KTMT/INT2212

28



Đa ngắt lồng nhau



(b) Nested interrupt processing

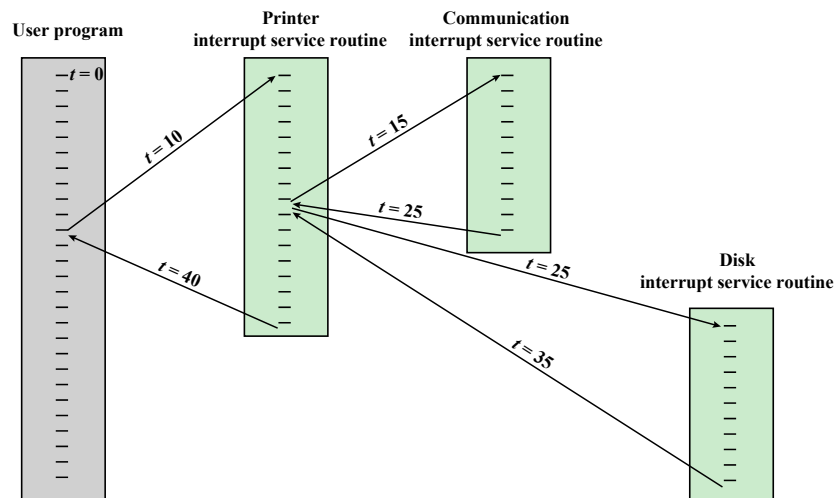
9/11/2024

KTMT/INT2212

29



Mô hình thời gian thi hành đa ngắt



9/11/2024

KTMT/INT2212

30



Tổng kết

- Hiểu rõ khái niệm chương trình cứng, chương trình mềm
- Ba nguyên tắc chính xây dựng mô hình kiến trúc Von Neuman
- Các chức năng và thành phần chính của máy tính
- Chu trình lệnh đầy đủ (có kèm cơ chế cho phép xử lý ngắt)