

# BÀI 6

## THỰC THI CHƯƠNG TRÌNH CÓ NGẮT

---



# Khái niệm Ngắt (Interrupt)

---

p ngắt quá

trình xử lý bình thường của CPU.

Có một số loại ngắt phổ biến:

- Program: phát sinh bởi điều kiện, là kết quả thực thi một chỉ thị, như tràn số học, chia 0...
- Timer: phát sinh bởi một bộ định thời trong CPU, cho phép OS thực hiện một chức năng thông thường nào đó.
- I/O: phát sinh bởi I/O controller, báo hoàn tất một công việc hay thông báo một điều kiện lỗi nào đó.
- Lỗi phần cứng



# Khái niệm Ngắt (Interrupt)...

---

nh

- 
- Các thiết bị ngoài hoạt động chậm hơn rất nhiều so với CPU → lãng phí CPU
- Ví dụ hoạt động in
- Ngắt là một phương thức nhằm cải thiện hiệu suất, CPU sẽ chuyển đến thực thi chỉ thị khác giữa lúc hoạt động I/O đang diễn ra.



# Chương trình I/O tiêu biểu

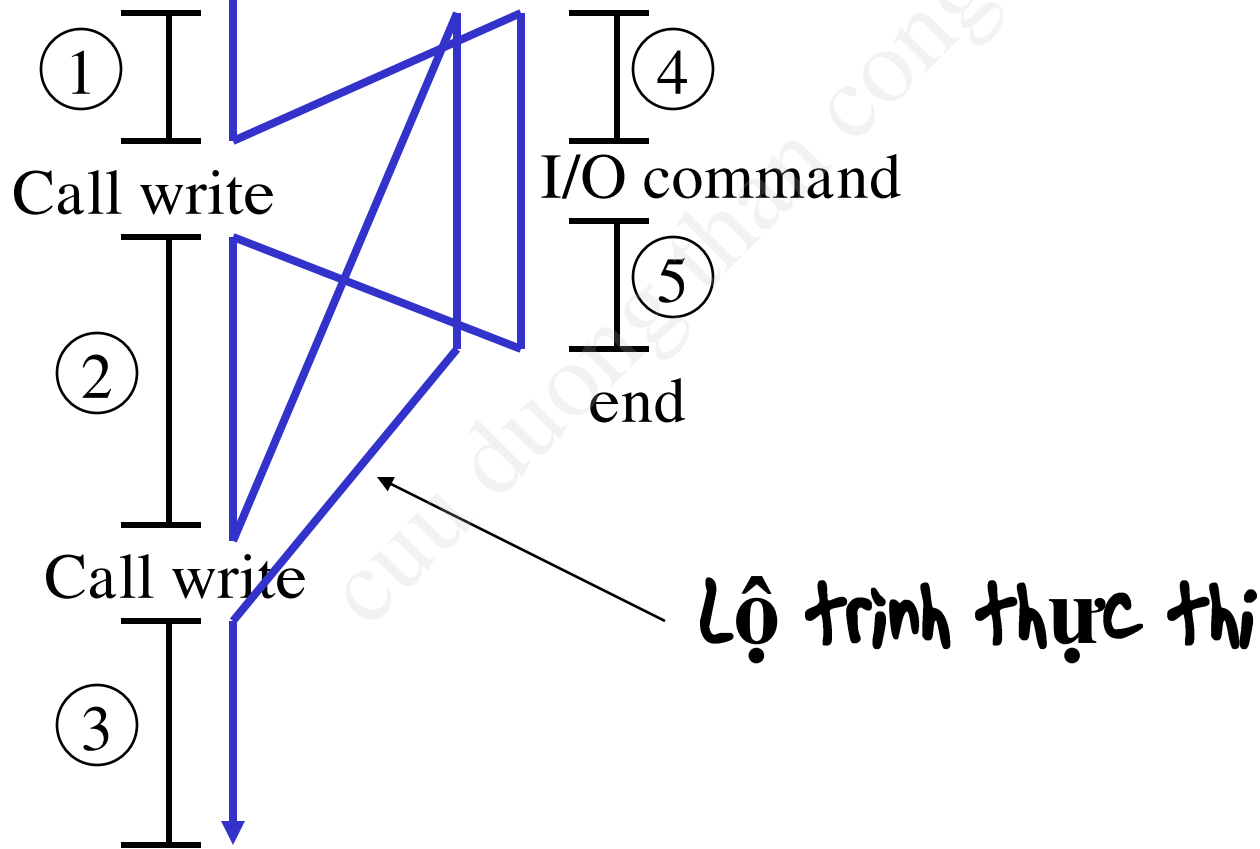
---

- Một chương trình I/O gồm có ba phần cơ bản:
  - Tuần tự các chỉ thị nhằm chuẩn bị cho hoạt động I/O thực sự
  - Lệnh I/O, khi gọi lệnh này chương trình phải đợi thiết bị I/O thực hiện chức năng được yêu cầu. Chương trình có thể đợi theo cách lặp lại thao tác kiểm tra để xác định hoạt động I/O đã được thực hiện hay chưa.
  - Tuần tự chỉ thị làm thủ tục hoàn tất hoạt động I/O, ví dụ dựng các cờ, thiết lập thông báo...

# Hoạt động của chương trình không ngắt

**Main program**

**I/O program**



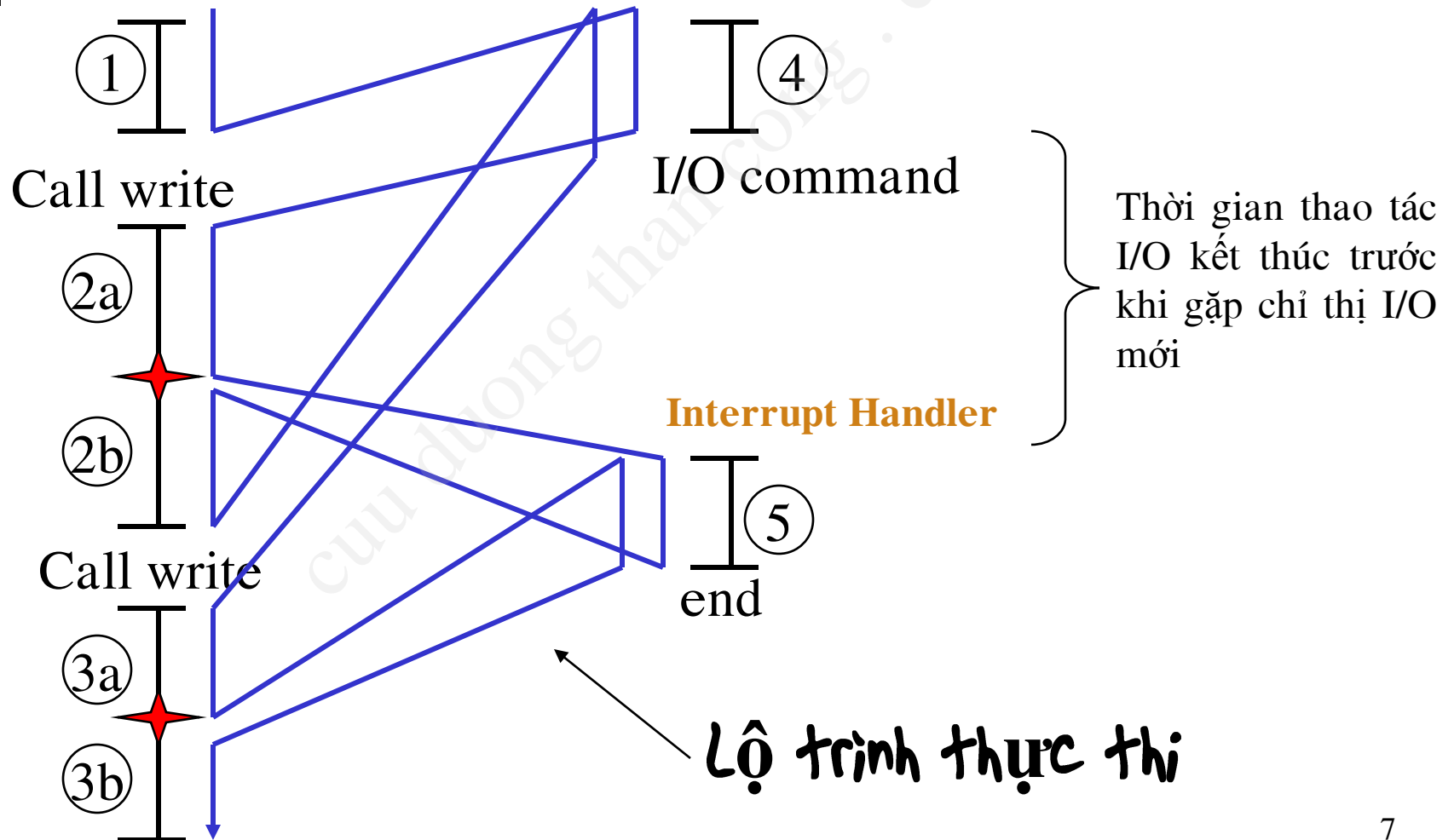


# Hoạt động của chương trình không ngắt...

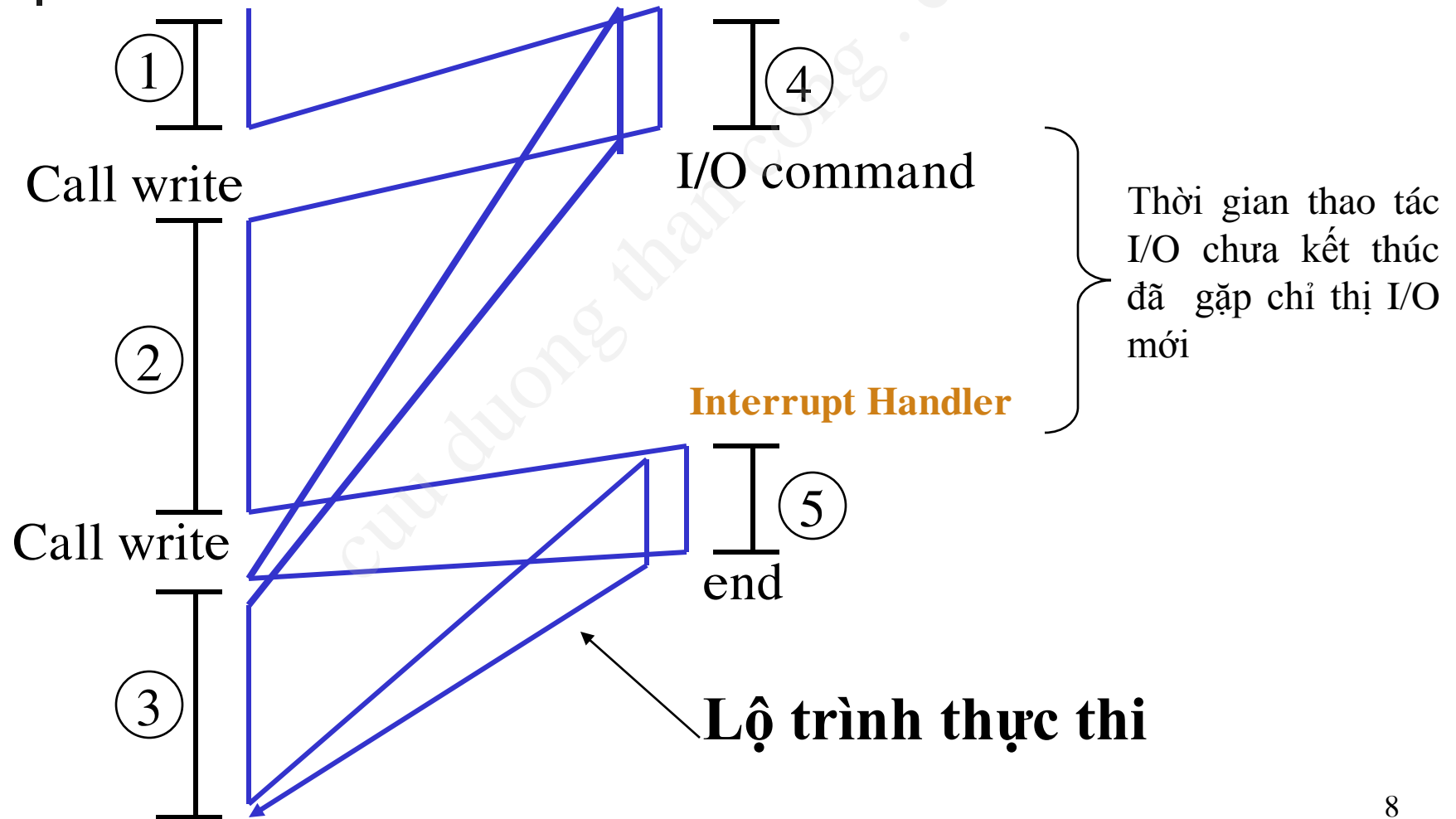
---

- Hoạt động I/O liên quan đến các thiết bị ngoài với các thao tác cơ-điện mất nhiều thời gian.
  - Chương trình I/O phải đợi cho các thao tác I/O hoàn tất.
- => Chương trình chính sẽ phải dừng lại mỗi khi gặp một chỉ thị I/O (ví dụ write) trong một thời gian khá dài.

# Hoạt động của chương trình có ngắt



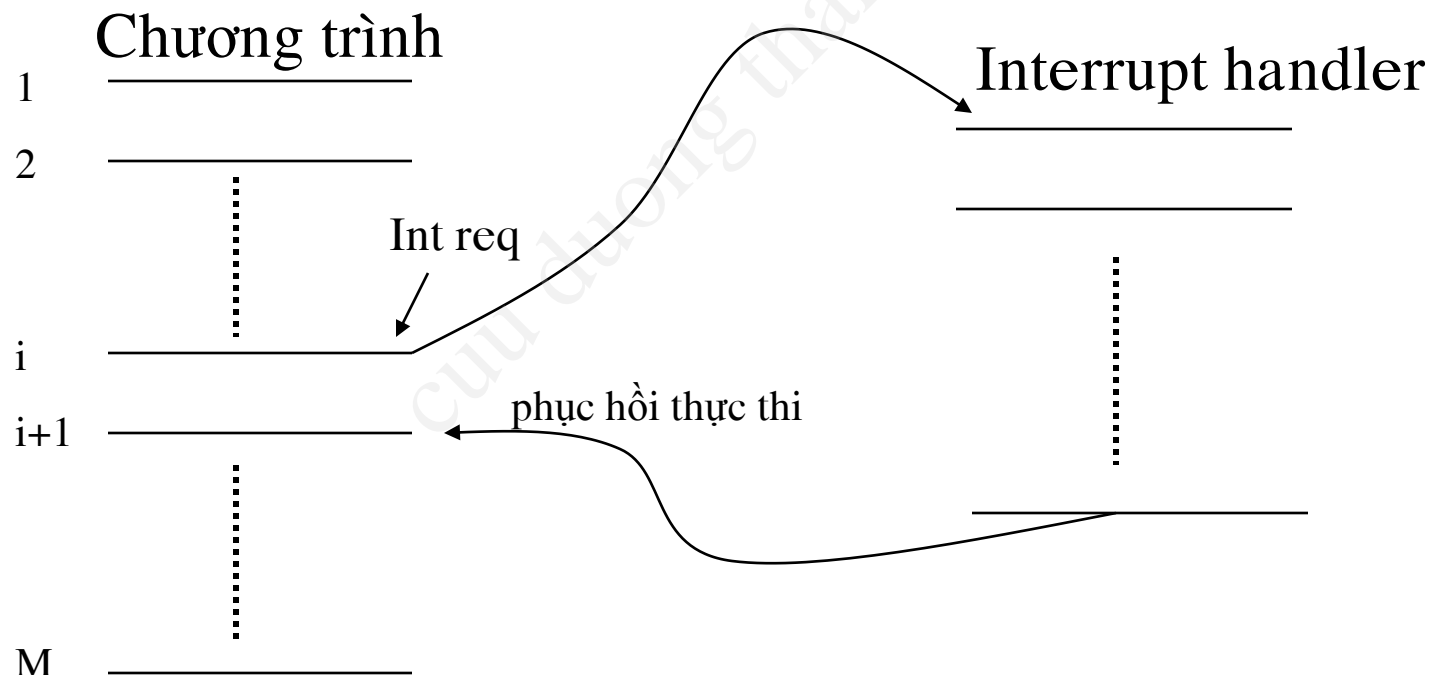
# Hoạt động của chương trình có ngắt...





# Thủ tục ngắt tổng quát

- Phát sinh tín hiệu I/O module
- Đáp ứng ngắt từ CPU: treo, chuyển

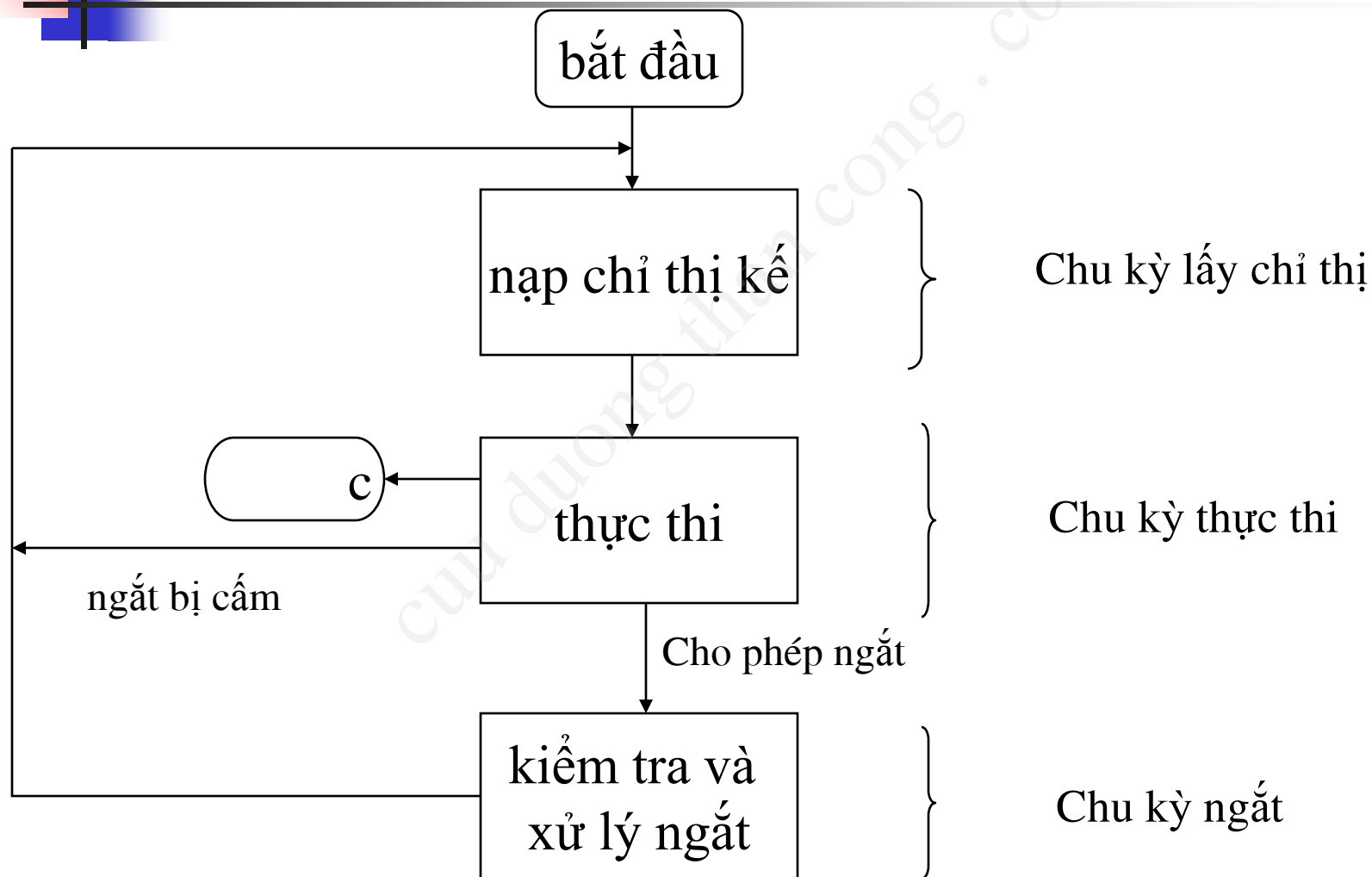
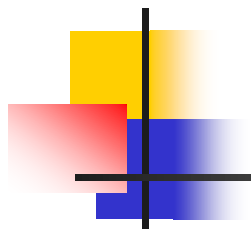




# Chu kỳ chỉ thị có ngắt

---

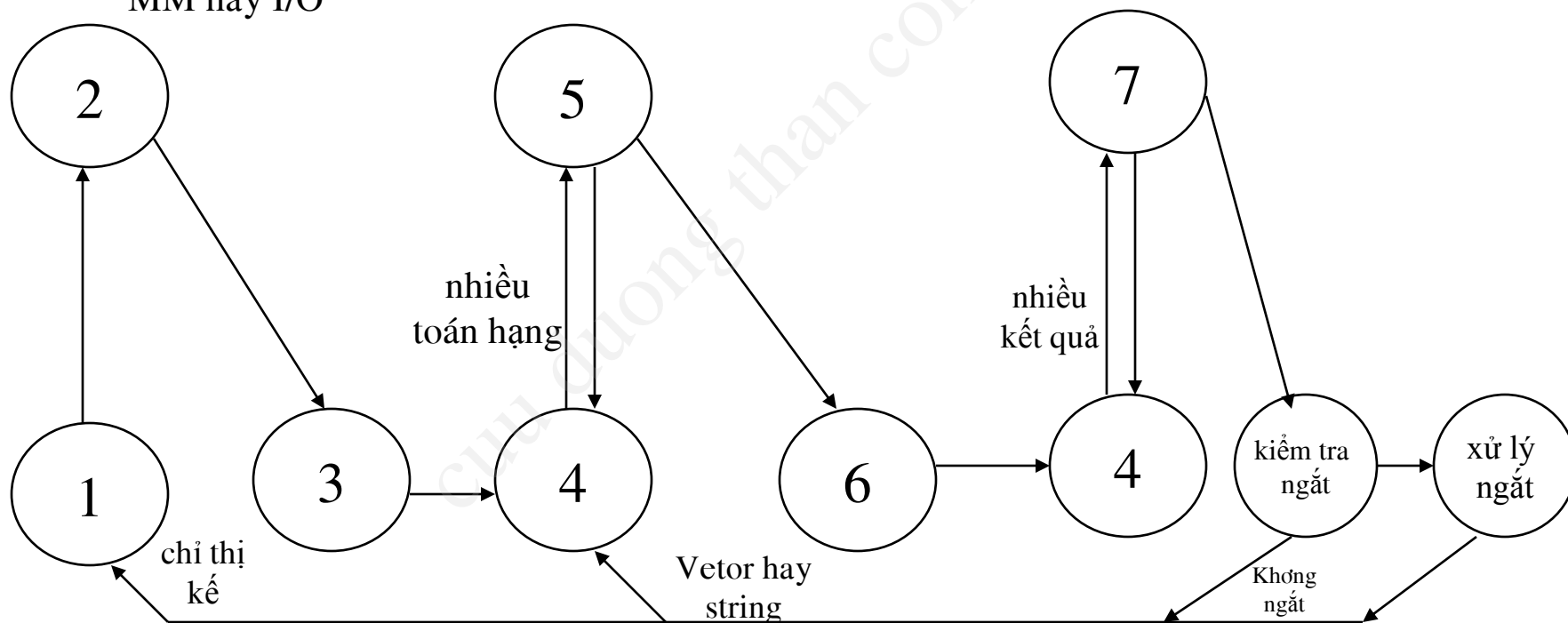
- Chương trình chính không chứa mã lệnh chuyên trách về ngắt
- CPU & OS chịu trách nhiệm
- Chu kỳ ngắt (interrupt cycle): CPU kiểm tra phát hiện int req.



# Sơ đồ trạng thái trong chu kỳ chỉ thị

CPU truy xuất

MM hay I/O



Hoạt động

bên trong CPU

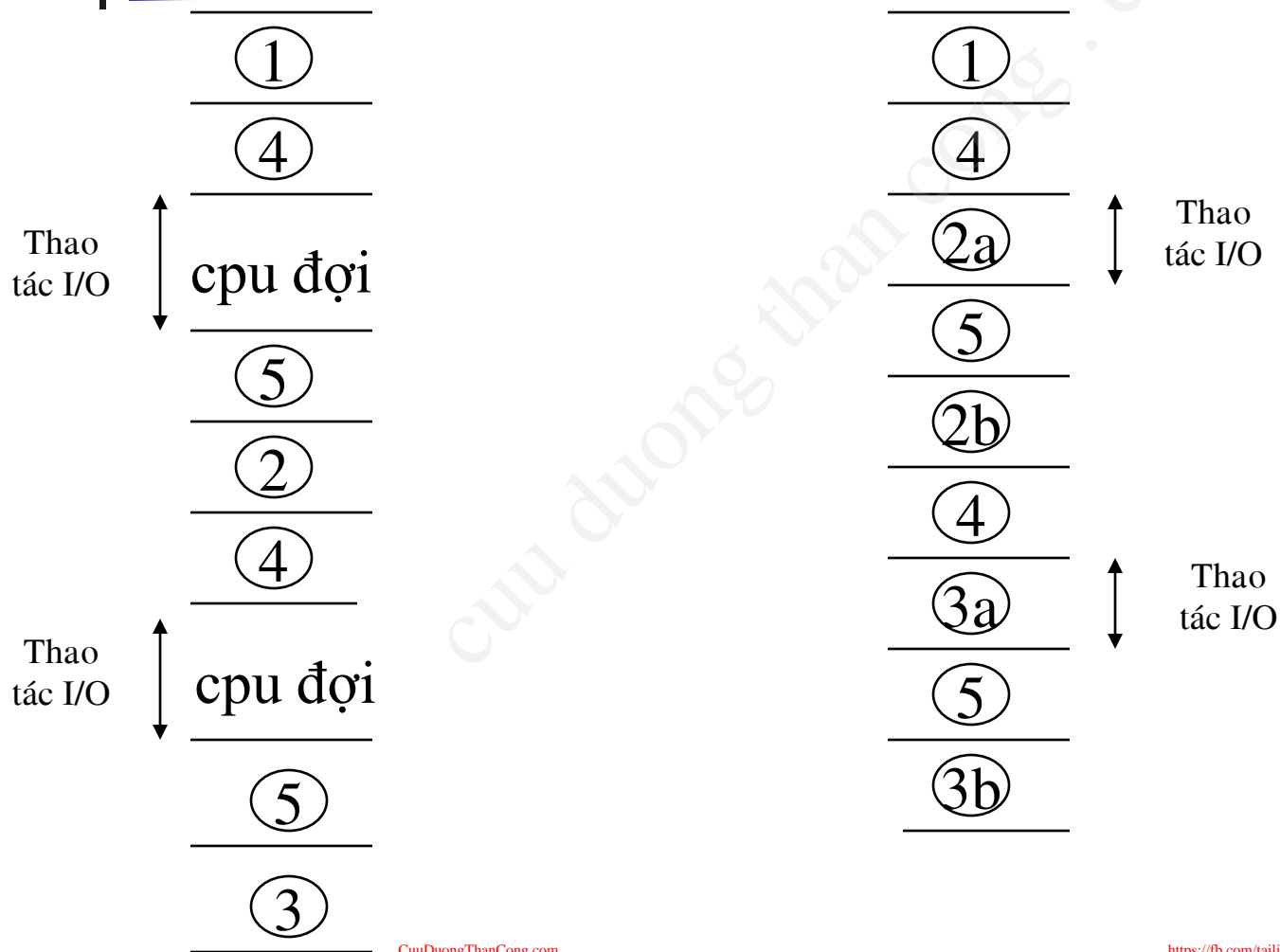


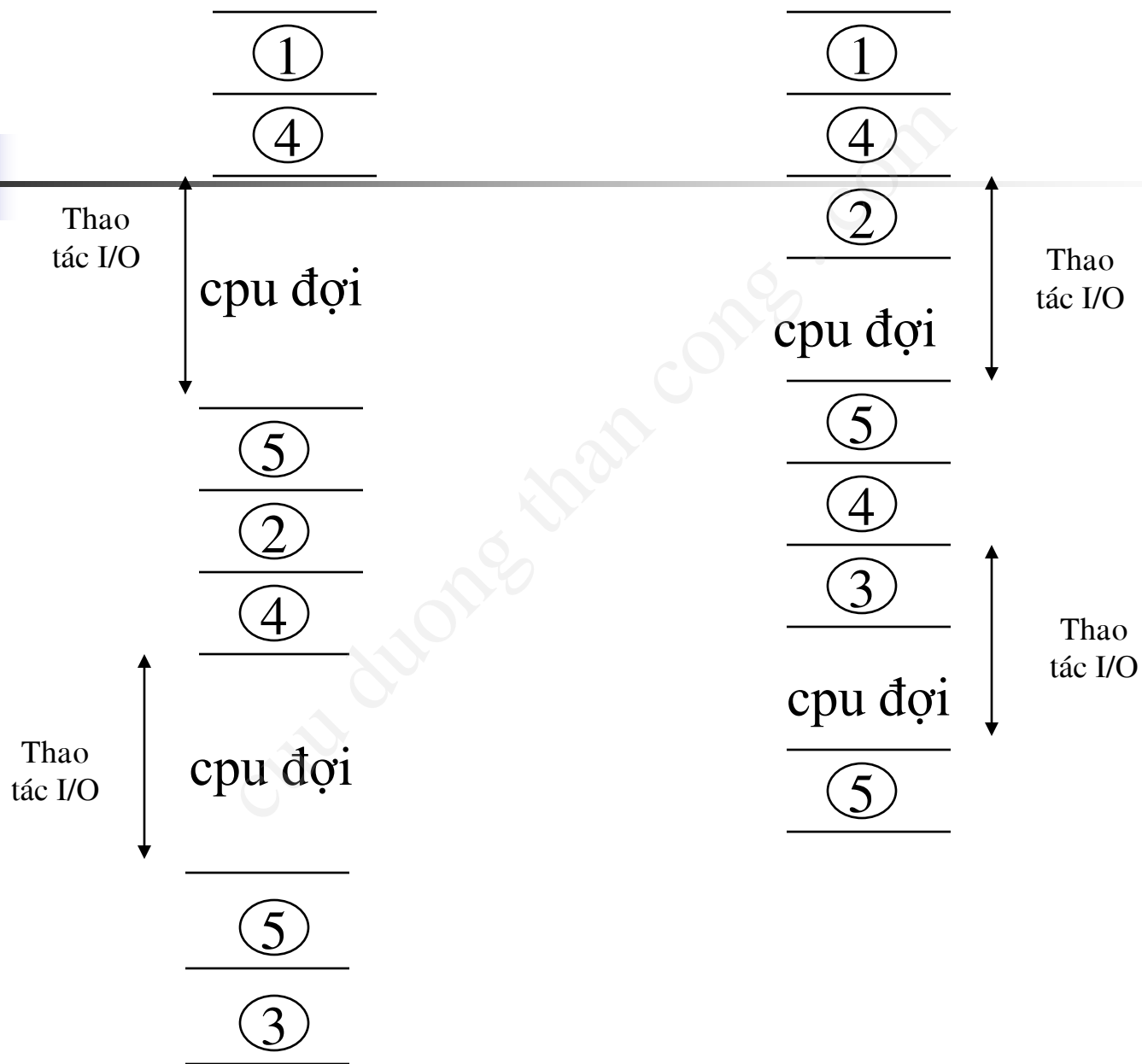
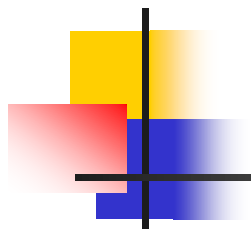
# Hoạt động ngắt và Interrupt Handler

---

- Nếu một ngắt xảy ra, bộ xử lý tiến hành:
  - Dừng hoạt động thực thi hiện hành và lưu ngữ cảnh.
  - Gán địa chỉ của Int handler vào thanh ghi PC
- Thực thi diễn ra trên chương trình phục vụ ngắt
- Int Handler  $\in$  OS
  - Xác định bản chất ngắt
  - Thực hiện các thao tác cần thiết

# Đánh giá





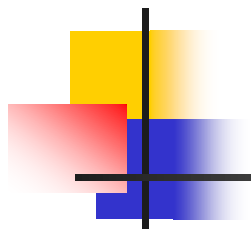


# Đa ngắt

---

- Có thể có nhiều ngắt xảy ra đồng thời
- Có hai giải pháp kiểm soát:
  - cấm các ngắt khác khi đang xử lý một ngắt nào đó
  - xử lý ưu tiên

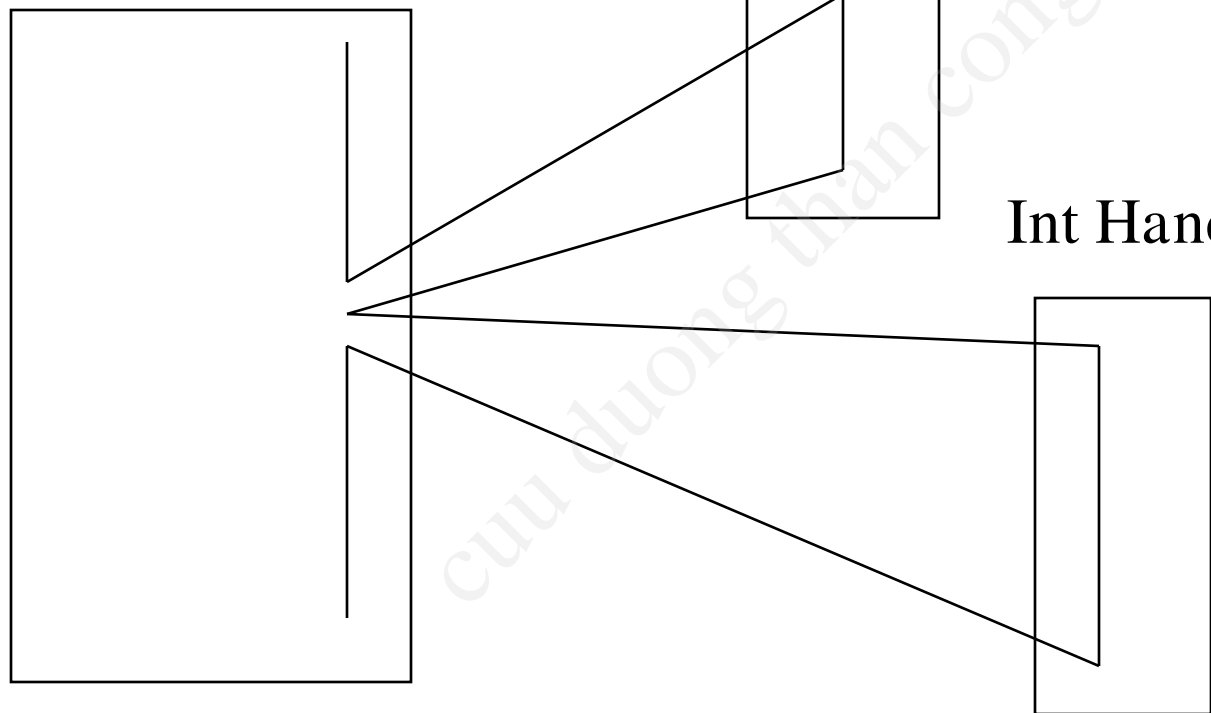


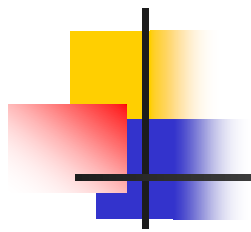


Chương trình chính

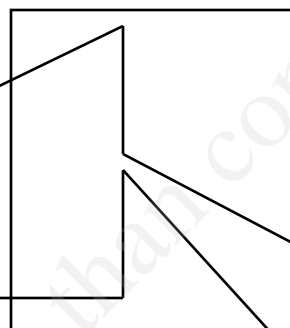
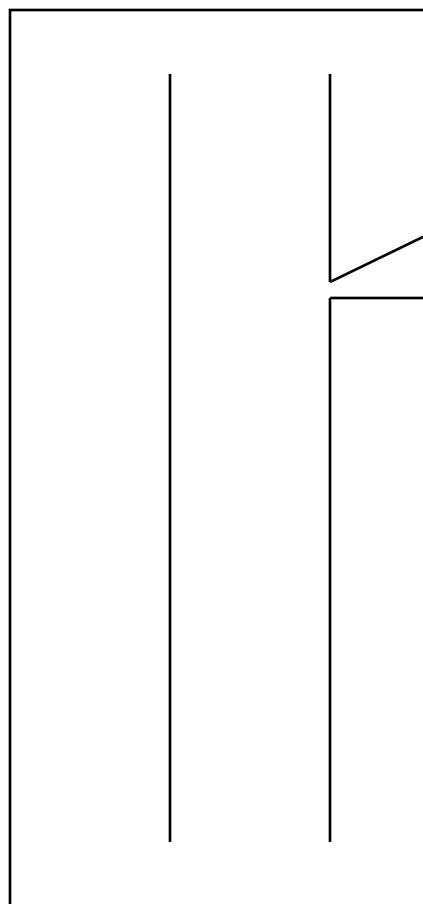
Int Hand

Int Hand

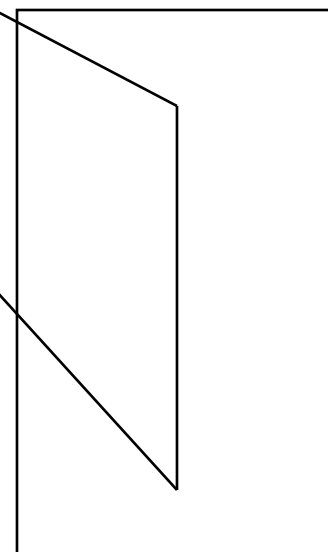




Int Hand



Int Hand

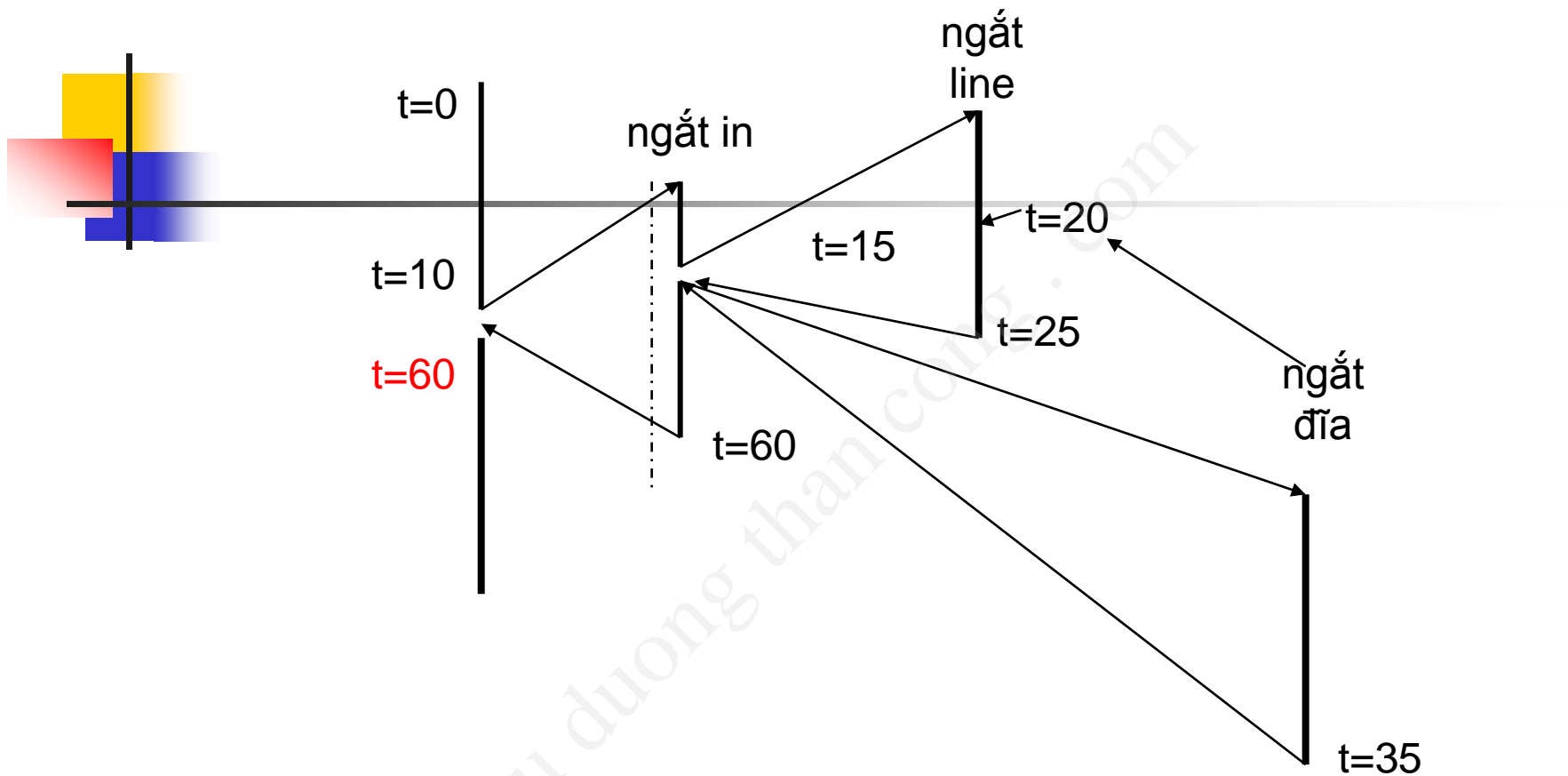




# Ví dụ

---

- Ba thiết bị ngoại vi:
  - Line (ưu tiên 1) thời gian cần 10
  - Đĩa (ưu tiên 2) thời gian cần 10
  - In (ưu tiên 3) thời gian cần 30
  - bắt đầu lúc  $t=0$ , ngắt in lúc  $t=10$ , ngắt thông tin lúc  $t=15$ , ngắt đĩa lúc  $t=20$
  - Trình bày diễn biến của các ngắt?



- Giải thích:
- Sau khi thực hiện xong chương trình ngắt đường dây, ngữ cảnh dừng trước đó sẽ được phục hồi, PC được nạp lại giá trị dừng trước đó, quay về chương trình ngắt in.
  - Tuy nhiên tại thời điểm này tồn tại hai ngắt: ngắt in và ngắt đĩa, ngắt đĩa có ưu tiên cao hơn nên sẽ được phục vụ trước, do đó giá trị PC hiện hành tại ch/tr ngắt in sẽ được lưu lại
  - PC được nạp địa chỉ của ch/tr ngắt đĩa để chuyển sang thực hiện ch/tr này