**Instruction Pipelining – Pipelining Strategy**

**( Câu lệnh và kỹ thuật ống dẫn )**

* + Khái niệm :
    - Đẩy các câu lệnh lên tiếp vào quá trình thực hiện , thay vì chờ cho 1 câu lệnh chạy hoàn thành.
    - Thay vì chờ chờ 1 câu lệnh hoàn thành rồi mới chấp nhận câu lệnh khác, Pipelining sẽ chấp nhận liên tục các câu lệnh và thực hiện chúng song song
  + Ưu điểm :
    - tăng hiệu suất (vì giảm thời gian xử lý chuỗi câu lệnh )là thực hiện các công việc theo cách song song

**Additional Stages**

1. Fetch instruction (FI): nạp câu lệnh vào buffer
2. Decode instruction (DI): giải mã câu lệnh ( Xác định opcode và các chỉ định toán hạng)
3. Calculate operands (CO): tính địa chỉ toán hạng ( Tính toán địa chỉ hiệu quả của từng toán hạng nguồn. Điều này có thể liên quan đến việc dịch chuyển, đăng ký gián tiếp, gián tiếp hoặc các hình thức tính toán địa chỉ khác)
4. Fetch operands (FO): Lấy từng toán hạng từ bộ nhớ ( Các toán tử trong thanh ghi không cần phải tìm nạp )
5. Execute instruction (EI): thực hiện câu lệnh
6. Write operand (WO): lưu kết quả vào bộ nhớ

**Timing Diagram for Instruction Pipeline Operation**

**( Sơ đồ thời gian hoạt động của kỹ thuật ống dẫn)**

Nếu không sử dụng kỹ thuật ống dẫn

sau khi câu lệnh 1 hoàn thành ở đv thời gian thứ 6,

mới tiếp tục nhận câu lệnh thứ 2, câu lệnh thứ 2 hoàn thành mất thêm 6 đv thời gian nữa là đến đv thời gian thứ 12 hoàn thành, lúc đó bộ xử lý mới nhận tiếp câu lệnh thứ 3

Nhưng với kỹ thuật ống dẫn

Sau khi câu lệnh 1 được xác nhận ở đv thời gian 1

Qua đv thời gian 2 bộ xử xử lý sẽ tiếp tục thực hiện câu lệnh 1 và xác nhận thêm câu lệnh 2

Cứ như v các câu lệnh sẽ được thực hiện song song

**The Effect of a Conditional Branch on Instruction Pipeline Operation**

**(Ảnh hưởng của một nhánh có điều kiện đối với hoạt động của đường ống chỉ dẫn)**

Giả xử câu lệnh 3 là điều kiện của câu lệnh 15, thì chỉ khi câu lệnh 3 thực hiện xong => câu lệnh 15 mới đủ điều kiện để tiếp tục.

Lãng phí

**Six Stage   
Instruction Pipeline**

**(logic xác định điều kiện rẽ nhánh)**

Sau khi đọc, giải mã, tinh các toán hạng

**Pipeline Hazards( rủi ro)**

Xảy ra khi đường ống, hoặc một phần của đường ống, phải ngừng hoạt động (giữ lại) vì điều kiện không cho phép tiếp tục thực hiện

**Resource Hazards**

* Một nguy cơ tài nguyên xảy ra khi hai hoặc nhiều hướng dẫn đã có trong đường ống cần cùng một tài nguyên. Kết quả là các hướng dẫn phải được thực hiện nối tiếp chứ không phải song song cho một phần của đường ống

**Data Hazards**

*Nguy cơ dữ liệu* xảy ra khi có xung đột trong việc truy cập vị trí toán hạng

**Types of Data Hazard**

• Đọc sau khi viết (RAW) hoặc phụ thuộc thực sự

o Một lệnh sửa đổi một thanh ghi hoặc vị trí bộ nhớ

o Lệnh thành công đọc dữ liệu trong bộ nhớ hoặc vị trí đăng ký

o Rủi ro xảy ra nếu quá trình đọc diễn ra trước khi thao tác ghi hoàn tất

• Viết sau khi đọc (WAR) hoặc chống phụ thuộc

o Một lệnh đọc một thanh ghi hoặc vị trí bộ nhớ

o Hướng dẫn thành công ghi vào vị trí

o Nguy hiểm xảy ra nếu thao tác ghi hoàn thành trước khi hoạt động đọc diễn ra

• Viết sau khi ghi (WAW) hoặc phụ thuộc đầu ra

o Hai hướng dẫn cả hai ghi vào cùng một vị trí

o Nguy hiểm xảy ra nếu các thao tác ghi diễn ra theo thứ tự ngược lại của chuỗi dự định