

TẬP CHỈ THỊ

- Tập hợp các chỉ thị khác nhau mà bộ xử lý có thể thực thi.

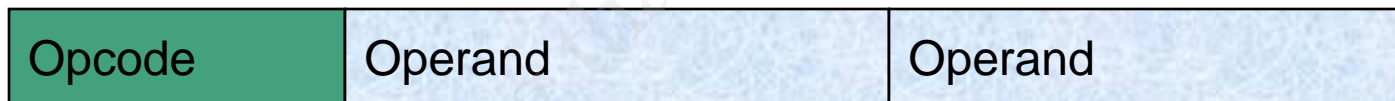
CÁC ĐẶC TRƯNG

- Các thành phần của một chỉ thị
- Biểu diễn chỉ thị
- Loại chỉ thị
- Số lượng địa chỉ
- Đặc trưng thiết kế

CÁC THÀNH PHẦN

- Mã lệnh (Operation code)
- Toán hạng nguồn (Source operand)
- Toán hạng đích (Result operand)
- Tham chiếu chỉ thị kế (Next instruction reference)

BIỂU DIỄN CHỈ THỊ



LOẠI CHỈ THỊ

- Toán học
 - Số học
 - Luận lý
- Lưu trữ dữ liệu
- Di chuyển dữ liệu
- Điều khiển

SỐ LƯỢNG ĐỊA CHỈ

- Bao nhiêu địa chỉ được chứa trong một chỉ thị
- Địa chỉ được biểu diễn tường minh hay không tường minh
 - ☐ Địa chỉ bộ nhớ
 - ☐ Thanh ghi, bộ tích lũy (accumulator)
 - ☐ Chỉ thị không địa chỉ

CÁC ĐẶC TRƯNG THIẾT KẾ

- Tác vụ của lệnh
- Kiểu toán hạng
- Các thanh ghi
- Chế độ địa chỉ

MỘT SỐ ĐỘ ĐO TẬP CHỈ THỊ

- Kích thước lưu trữ mà chương trình cần
- Độ phức tạp của tập chỉ thị cũng như độ phức tạp của các tác vụ
- Chiều dài của chỉ thị
- Tổng số chỉ thị
- Bao nhiêu thanh ghi và tổ chức các thanh ghi như thế nào

CÂN NHẮC KHI THIẾT KẾ

- Chỉ thị ngắn hay dài
- Chiều dài cố định hay thay đổi (cố định dễ giải mã nhưng lãng phí)
- Tổ chức bộ nhớ (địa chỉ hóa theo byte hay không)
- Chỉ thị có chiều dài cố định không nhất thiết phải cố định số toán hạng (expanding opcode)
- Chế độ địa chỉ hóa
- Thứ tự lưu giữ các byte của các từ có nhiều byte như thế nào (little endian và big endian)

HỖ TRỢ LƯU TRỮ BÊN TRONG CPU

- Kiến trúc stack
- Kiến trúc accumulator
- Kiến trúc thanh ghi mục đích tổng quát (GPR)

CHẾ ĐỘ ĐỊA CHỈ (ADDRESSING MODE)

- Immediate addressing
- Direct addressing
- Indirect addressing
- Register addressing
- Register indirect addressing
- Displacement addressing
- Based addressing
- Indexed addressing
- Stack addressing
- Các chế độ cải tiến

ĐỊA CHỈ TỨC THỜI (IMMEDIATE ADDRESSING)

- Giá trị tham chiếu nằm ngay trong chỉ thị
- Không có tham chiếu bộ nhớ để lấy dữ liệu
- Nhanh
- Ví dụ

ADD #5

Cộng 5 vào nội dung của thanh ghi AC
5 là toán hạng

ĐỊA CHỈ TRỰC TIẾP (DIRECT ADDRESSING)

- Toán hạng là địa chỉ của giá trị tham chiếu
- Tham chiếu một vị trí bộ nhớ để truy xuất dữ liệu
- Ví dụ

ADD 3BF

Cộng nội dung của ô nhớ 3BF với nội dung trong AC

3BF là địa chỉ hiệu quả (effective address)

ĐỊA CHỈ GIÁN TIẾP (INDERECT ADDRESSING)

- Toán hạng là địa chỉ của con trỏ chỉ đến dữ liệu
- Địa chỉ hiệu quả chính là con trỏ
- Ví dụ

ADD 38F

Cộng nội dung tại ô nhớ có địa chỉ được chứa trong ô nhớ 38F với nội dung trong AC
38F không phải là địa chỉ hiệu quả.

ĐỊA CHỈ THANH GHI (REGISTER ADDRESSING)

- Toán hạng là tên thanh ghi (địa chỉ thanh ghi)

-

u.

- Truy xuất nhanh
- Ví dụ

ADD R1 cộng nội dung thanh ghi R1 với nội dung của AC

ĐỊA CHỈ GIÁN TIẾP THANH GHI (REGISTER INDIRECT ADDRESSING)



- Kết hợp địa chỉ thanh ghi và địa chỉ gián tiếp
- Dùng thanh ghi để chứa con trỏ chỉ đến vị trí chứa giá trị tham chiếu

cuu duong than cong .com

ĐỊA CHỈ DÙNG ĐỘ DỜI (DISPLACEMENT ADDRESSING)

- Phần toán hạng chứa hai giá trị:
 - Địa chỉ
 - Thanh ghi giữ độ dời (offset)
- Địa chỉ hiệu quả = địa chỉ + độ dời

ĐỊA CHỈ QUAN HỆ (RELATIVE ADDRESSING)

- Từ địa chỉ dùng độ dời, nếu thanh ghi là PC thì gọi là địa chỉ quan hệ.
- Lấy nội dung từ ô nhớ tại vị trí "địa chỉ" tính từ vị trí hiện hành được chỉ ra trong thanh ghi PC.
- Ví dụ

LD A

nạp nội dung từ ô nhớ $A+(PC)$ vào thanh ghi AC

ĐỊA CHỈ DÙNG THANH GHI NỀN (BASE-REGISTER ADDRESSING)



- Giá trị thứ nhất là độ dời
- Thanh ghi chứa con trỏ chỉ đến địa chỉ nền
- Thanh ghi có thể là tường minh hay không tường minh (ngầm)

INDEXED ADDRESSING

- Giá trị địa chỉ trong chỉ thị chứa địa chỉ nền
- Thanh ghi chứa độ dời (offset)
- Địa chỉ hiệu quả = địa chỉ nền + độ dời
- Thích hợp cho truy xuất mảng

Địa chỉ truy xuất = địa chỉ nền + độ dời trong
thanh ghi R

R++

ĐỊA CHỈ NGẪN XẾP (STACK ADDRESSING)

- Các toán hạng được ngầm định trên đỉnh stack
- Ví dụ

ADD

Lấy hai giá trị trên đỉnh stack và thực hiện cộng hai giá trị với nhau

CÁC CHẾ ĐỘ ĐỊA CHỈ CẢI TIẾN

- Có thể kết hợp các chế độ địa chỉ với nhau.
- Ví dụ indirect indexed addressing, indirect based register addressing ...