

Chương 3

Tổ chức hoạt động của VXL

3.1 Những đặc điểm cấu trúc của bộ VXL

3.1.1 Công suất của bộ vi xử lý

Công suất của bộ vi xử lý (VXL) là khả năng xử lý dữ liệu, bao gồm 3 đặc điểm chính sau:

- Độ dài từ của bộ VXL (data word length), tính bằng byte
- Dung lượng nhớ vật lý có thể đánh địa chỉ (addressing capacity)
- Tốc độ xử lý lệnh của bộ VXL (instruction execute speed).

Công suất của máy tính, hay tốc độ xử lý thông tin, khả năng lưu trữ thông tin, khả năng kết nối nhiều loại thiết bị ngoại vi... phụ thuộc vào công suất của bộ VXL trong CPU.

02/03/2016

1

Tổ chức hoạt động của VXL (tt)

b) Khả năng đánh địa chỉ.

- Các từ dữ liệu và lệnh máy được lưu trong bộ nhớ tại các ô nhớ khác nhau. Mỗi ô nhớ phải có 1 địa chỉ riêng biệt. Dài đánh địa chỉ càng lớn thì dung lượng bộ nhớ càng nhiều. Độ đánh địa chỉ, bộ VXL thường sử dụng thanh ghi địa chỉ (address register).
- Với số mũ ở hệ cơ số 2 (nhị phân) ta có thể đánh giá được độ rộng của thanh ghi địa chỉ hay bus địa chỉ, ví dụ, để đánh địa chỉ đến 4GB cần phải có 32 đường dây địa chỉ (A0 - A31).
- Khả năng đánh địa chỉ càng lớn thì càng cho phép tạo ra một hệ thống máy tính có cấu hình mạnh với nhiều loại thiết bị ngoại vi, bộ nhớ chính có dung lượng lớn (đến vài trăm MB) và khả năng xử lý nhanh.

02/03/2016

3

Tổ chức hoạt động của VXL (tt)

a) Độ dài từ.

- Mỗi một bộ VXL chỉ có thể xử lý dữ liệu với độ dài từ cố định. Phụ thuộc vào từng thế hệ VXL và mức độ phát triển của công nghệ VXL, độ dài từ có thể là 4-bit, 8-bit, 16-bit, 32-bit, 64-bit. Tập lệnh của bộ VXL thường có các lệnh thực hiện theo từ và theo byte.
- Độ rộng của từ có độ dài bao nhiêu bit thì cũng có bấy nhiêu bit đối với các thanh ghi, ALU và bus dữ liệu bên trong của bộ VXL. Bus dữ liệu bên ngoài có thể bằng hoặc nhỏ hơn.
- Độ dài từ càng lớn càng tạo ra nhiều khả năng tính toán của bộ VXL: khoảng biểu diễn số rộng hơn, tốc độ tính toán nhanh hơn.

02/03/2016

2

Tổ chức hoạt động của VXL (tt)

c) Tốc độ thực hiện lệnh.

- Tốc độ thực hiện lệnh của bộ VXL có thể đo bằng tốc độ thực hiện các lệnh dấu phẩy động FLOPS (Floating Point Operations per Second) hoặc tính bằng triệu lệnh/giây (MIPS - Millions of instructions per second). Công thức tính MIPS là:

$$MIPS = \frac{f * N}{M + T}$$

trong đó:

f - tần số làm việc của bộ VXL

N - số lượng các đơn vị xử lý số học và logic (ALU) không phụ thuộc vào nhau bên trong bộ VXL.

4

Tổ chức hoạt động của VXL (tt)

c) Tốc độ thực hiện lệnh.

$$MIPS = \frac{f * N}{M + T}$$

M - số lượng vi lệnh (microinstruction) trung bình của 1 lệnh trong bộ VXL (thường để thực hiện 1 lệnh VXL cần từ 4-7 vi lệnh)

T- hệ số thời gian truy nhập bộ nhớ

Theo công thức này, tốc độ thực hiện lệnh của bộ VXL có thể thay đổi nhờ 4 yếu tố. Để nâng cao tốc độ VXL, kiến trúc song song, đường ống, đồng xử lý, bộ nhớ đệm trữ (cache memory) và bus rộng (wide bus) đã được áp dụng cho các chip VXL công nghệ cao hiện nay.

02/03/2016

5

Tổ chức hoạt động của VXL (tt)

a) Kỹ thuật đường ống (pipelining technique).

- Ý tưởng của kỹ thuật này là dựa trên công nghệ ché tao các sản phẩm như ô tô, điện tử, ... Theo đó, các bộ VXL hiện nay có chức năng thực hiện các lệnh máy lén tục thành 1 dây chuyền với các công đoạn:
 - nhận lệnh từ bộ nhớ,
 - giải mã lệnh,
 - thực hiện lệnh,
 - ghi kết quả thực hiện lệnh vào bộ nhớ.
- Khi lệnh thứ nhất bước vào thực hiện ở giai đoạn thứ 2 thì mã lệnh của lệnh tiếp theo được đọc từ bộ nhớ, quá trình tiếp tục như vậy, ...

02/03/2016

7

Tổ chức hoạt động của VXL (tt)

3.1.2 Những đặc tính nâng cao tốc độ của bộ VXL

- Kỹ thuật đường ống (pipelining technique)
- Xử lý song song (parallel processing) và kiến trúc siêu hướng (superscalar)
- Đồng xử lý
- Kỹ thuật nhớ dự trữ (Cache memory technique)
- Bus rộng (wider bus)

02/03/2016

6

Tổ chức hoạt động của VXL (tt)

a) Kỹ thuật đường ống (pipelining technique). (tt)

In a typical early RISC processor (eg MIPS R3000), there would be four pipeline stages

IF - Instruction Fetch

Fetch the instruction from memory

DEC - Decode and Operand Fetch

Decode it and fetch operands from the register file

EX - Execute

Execute the instruction in the ALU

WB - WriteBack

Write the result back in to a register

Time	IF	DEC	EX	WB
1	i_1			
2	i_2	i_1		
3	i_3	i_2	i_1	
4	i_4	i_3	i_2	i_1
5	i_5	i_4	i_3	i_2

Population of the pipeline at each clock cycle: i_1, i_2, \dots are successive instructions in the instruction stream.

02/03/2016

8

Tổ chức hoạt động của VXL (tt)

b) Xử lý song song.

- Xử lý song có nghĩa là 2 quá trình tính toán cùng xảy ra đồng thời. Trong kiến trúc máy tính, sự kết hợp 2 bộ VXL trong khối xử lý trung tâm (CPU) tạo ra khả năng xử lý song song cùng một thời gian.
- Khi một nhiệm vụ được xử lý tuần tự như kiểu Von Neumann thì bộ xử lý đó (hay máy tính) được gọi là bộ xử lý tuần tự. Nếu một nhiệm vụ T được chia thành n nhiệm vụ con T_1, T_2, \dots, T_n có kích thước xấp xỉ nhau được xử lý đồng thời (thực hiện song song) thì tốc độ thực hiện nhiệm vụ T sẽ nhanh gấp n lần và bộ xử lý (hay máy tính) đó gọi là bộ xử lý song song (máy tính song song).

02/03/2016

9

Tổ chức hoạt động của VXL (tt)

b) Xử lý song song.

- Nhu vậy xử lý song song có thể hình thành 1 chuỗi các lệnh liên tiếp nhau (instruction stream) đọc từ bộ nhớ vào bộ xử lý, trong khi các toán hạng cũng hợp thành một chuỗi dữ liệu (data stream) theo sau đó và từ bộ xử lý.
- Trên cơ sở đó, Flynn chia máy tính song song thành 4 nhóm dựa trên số lượng các chuỗi lệnh và chuỗi dữ liệu (luồng lệnh, luồng dữ liệu), như ở bảng 3.1 sau:

02/03/2016

10

Tổ chức hoạt động của VXL (tt)

b) Xử lý song song. (tt)

- SISD** - Single Instruction, Single Data (Một chuỗi lệnh, một chuỗi dữ liệu)

"Conventional" single processors: các máy tính có một VXL được thiết kế theo *cơ chế đường ống*. Đó là một kiểu xử lý song song khi bộ xử lý tìm-nạp lệnh kế tiếp ngay trong lúc đang thực thi lệnh hiện tại.

- SIMD** - Single Instruction, Multiple Data (Một chuỗi lệnh, nhiều chuỗi dữ liệu).

Đây là hệ thống dạng "Massively Parallel". Máy này thao tác trên nhiều tập dữ liệu song song. Nó thực hiện 1 lệnh giống nhau nhưng với nhiều dữ liệu khác nhau. Một vài hệ thống máy tính này như: CM-2, Systolic arrays và Vector processors

02/03/2016

11

Tổ chức hoạt động của VXL (tt)

b) Xử lý song song. (tt)

- MISD** - Multiple Instruction, Single Data (Nhiều chuỗi lệnh, một chuỗi dữ liệu)

Tại đây có nhiều bộ xử lý làm việc song song với những lệnh riêng nhưng cùng sử dụng chung 1 bộ dữ liệu. Loại máy này ít thông dụng trừ 1 vài máy chuyên dụng (specialised systems) có tính năng xử lý đường ống.

- MIMD** - Multiple Instruction, Multiple Data (Nhiều chuỗi lệnh, nhiều chuỗi dữ liệu)

(Multiple threads of control operating on multiple thread of data. Most current parallel systems belong here). Thuộc nhóm này là các máy tính được liệt vào kiến trúc đa xử lý trong đó các bộ xử lý làm việc song song độc lập với nhau, thực thi các lệnh khác nhau với các dữ liệu khác nhau.

02/03/2016

12

Tổ chức hoạt động của VXL (tt)

b) Xử lý song song. (tt)

- Máy tính Von Neumann truyền thống có thể được xem là thuộc loại SISD, có một chuỗi lệnh (1 chương trình) được thực hiện bởi 1 bộ VXL và 1 bộ nhớ chứa dữ liệu.
- Có thể thực hiện xử lý song song ngay bên trong cấu trúc của bộ VXL, bằng cách thiết kế sao cho quá trình xử lý dữ liệu bên trong chip VXL chia thành các phiên khác nhau và thực hiện song song nhờ sự phân chia khối logic điều khiển (CU) bên trong thành 2 phần riêng.

02/03/2016

13

Tổ chức hoạt động của VXL (tt)

b) Xử lý song song. (tt)

- Kiến trúc của bộ VXL có các khối chức năng xử lý song song bên trong gọi là kiến trúc siêu hướng (superscalar architecture), nghĩa là cùng 1 lúc có nhiều hướng xử lý khác nhau bên trong VXL.
- Kiến trúc siêu hướng là sự phát triển tiếp theo của kiến trúc RISC, nó không những nâng cao tốc độ xử lý mà còn nâng cao độ tin cậy của CPU, bởi vì khi có sự cố ở 1 chip VXL (hay 1 đơn vị xử lý bên trong chip) thì chip VXL còn lại (hay đơn vị xử lý còn lại bên trong 1 chip) vẫn đảm nhận chức năng được bình thường.
- Hiện nay, các máy tính sử dụng các bộ VXL kiến trúc RISC và các máy tính sử dụng các chip Intel Pentium II, III đều có khả năng xử lý song song bằng cấu trúc kết nối song song 2-4 chip VXL.

02/03/2016

14

Tổ chức hoạt động của VXL (tt)

c) Đồng xử lý(coprocessing).

- Tương tự như xử lý song song, bộ đồng xử lý là 1 bộ VXL riêng biệt kết nối với bộ VXL chính thông qua bus hệ thống. Ví dụ, trong họ VXL Intel 80X86 có bộ đồng xử lý 80387 làm việc với 80386 hay 80486.
- Bộ đồng xử lý chỉ thực hiện 1 số chức năng đặc biệt, như các phép toán đòi hỏi sự chính xác sử dụng dấu phẩy động. Tốc độ xử lý của bộ đồng xử lý những phép tính này sẽ nhanh hơn rất nhiều so với bộ xử lý chính.
- Các bộ VXL công nghệ cao hiện nay đã cấy vào bên trong khối xử lý dấu phẩy động FPU càng làm tăng tốc độ tính toán các phép tính nhanh và chính xác hơn nhiều

02/03/2016

15

Tổ chức hoạt động của VXL (tt)

d) Bộ nhớ đệm trứ (Cache memory).

- Từ công thức (1) ta thấy: tốc độ của bộ VXL sẽ tăng đáng kể nếu tổng M+T tiến tới 1. Thời gian truy nhập bộ nhớ T có thể giảm tối thiểu nhờ giảm tối thiểu số lượng lệnh đặt trong bộ VXL và từng giai đoạn trong khi thực hiện lệnh và truy nhập bộ nhớ gài chồng lên nhau về thời gian.
- Biện pháp này được áp dụng trong các bộ VXL kiến trúc RISC (Reduced Instruction Set Computing). Bộ VXL kiến trúc RISC chỉ làm việc với thanh ghi và chỉ có 2 lệnh truy nhập bộ nhớ là LOAD và STORE.

02/03/2016

16

Tổ chức hoạt động của VXL (tt)

d) Bộ nhớ dự trữ (Cache memory).

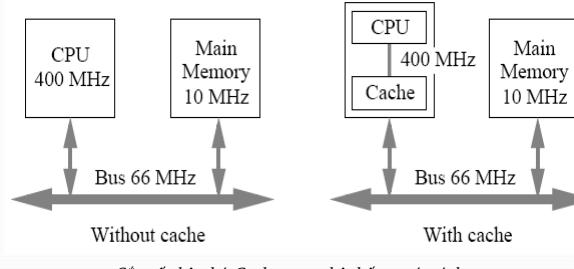
- Một biện pháp giảm T nữa là áp dụng kỹ thuật bộ nhớ dự trữ. Bộ nhớ Cache là bộ nhớ có tốc độ rất cao, nó có thể nằm ngay bên trong bộ VXL với dung lượng hạn chế (internal cache), hoặc nằm kè ngay bên cạnh bộ VXL và kết nối trực tiếp với chip VXL với dung lượng đủ lớn (external cache), trong khi bộ nhớ chính kết nối với VXL thông qua bus hệ thống.
- Sự trao đổi dữ liệu giữa bộ nhớ chính và bộ VXL bị hạn chế về tốc độ, vì vậy để tăng tốc độ xử lý, khi thực hiện chương trình, bộ VXL sẽ tìm kiếm lệnh ở bộ nhớ Cache trước, nếu không có thì mới thực hiện tìm trong bộ nhớ chính. Vì vậy, thông thường các lệnh có tần số xuất hiện trong chương trình cao thường được lưu trong bộ nhớ dự trữ.

02/03/2016

17

Tổ chức hoạt động của VXL (tt)

d) Bộ nhớ dự trữ (Cache memory).(tt)



02/03/2016

18

Tổ chức hoạt động của VXL (tt)

3.1.3 Tập lệnh của bộ VXL.

- Tập lệnh là một đặc tính then chốt của kiến trúc bộ VXL. Các bộ VXL hoạt động theo sự chỉ dẫn của tập lệnh riêng của chúng. Người lập trình hay nhà thiết kế hệ thống dựa vào tập lệnh để soạn thảo các chương trình điều khiển hoạt động của bộ VXL theo 1 trình tự nào đó.
- Mỗi bộ VXL có 1 tập lệnh riêng và thường có tính kề thừa. Tập lệnh thường được chia thành nhiều nhóm và thường được biểu diễn ở dạng các từ gọi nhớ (Hợp ngữ). Ví dụ:
 - + Các lệnh chuyển dữ liệu: MOV, IN, OUT, ...
 - + Các lệnh số học và logic: AND, OR, ...
 - + Các lệnh dịch và quay: SHR, SHL, ...
 - + Các lệnh chuyển điều khiển: Jmp, Call, Ret, ...

02/03/2016

19

Tổ chức hoạt động của VXL (tt)

3.1.3 Tập lệnh của bộ VXL. (tt)

- Tập lệnh máy mà ta xét ở trên thực ra chỉ là các lệnh vĩ mô. Trong cấp máy vi chương trình, để thực hiện các lệnh này, thường các đơn vị trong bộ xử lý phải dịch chúng ra thành các lệnh vi mô, đó là một số hạn chế các thao tác đơn giản nhất có thể được thực hiện trực tiếp bởi phần cứng logic số.
- Cùng với các thanh ghi, đơn vị điều khiển CU là thành phần quan trọng và phức tạp nhất trong bộ xử lý. Việc khảo sát quá trình thực hiện lệnh máy bên trong bộ xử lý bởi đơn vị điều khiển, bộ nhớ và các thanh ghi được thực hiện qua ngôn ngữ truyền thanh ghi RTL (Register Transmission Language).

02/03/2016

20

Tổ chức hoạt động của VXL (tt)

3.1.3 Tập lệnh của bộ VXL. (tt)

- Đơn vị CU có thể được thiết kế theo 2 cách:

+ Đơn vị điều khiển được vi chương trình hóa, trong đó các lệnh máy phức tạp được dịch thành các vi chương trình. Vì chương trình bao gồm 1 tập hạn chế các vi lệnh là những thao tác đơn giản nhất thực hiện được bởi mạch điện logic số. Rõ ràng quá trình thực hiện 1 lệnh máy loại này phải mất 1 khoảng thời gian giải mã vi lệnh.

+ Đơn vị điều khiển logic ngẫu nhiên, trong đó bộ xử lý được thiết kế sao cho các mạch logic có thể “hiểu” và thực hiện được ngay 1 số lệnh máy mà không qua giải mã vi lệnh.

02/03/2016

21

Tổ chức hoạt động của VXL (tt)

3.1.3 Tập lệnh của bộ VXL. (tt)

- Xuất phát từ đó cùng với các thông kê thực tiễn trên các lệnh máy trong các chương trình đã dẫn tới 2 loại bộ xử lý:

- Loại có tập lệnh phức tạp CISC, trong đó gồm tập lệnh có cấu trúc phức tạp và muốn thực hiện phải qua giải mã vi lệnh. Loại này rất được thịnh hành khi tốc độ truy nhập bộ nhớ còn chậm.

- Loại có tập lệnh được rút gọn RISC, trong đó bộ xử lý được thiết kế để có thể thực hiện được ngay 1 số lệnh có cấu trúc đơn giản hơn, hay được dùng nhất, không phải qua giải mã vi lệnh (thường khoảng 20%).

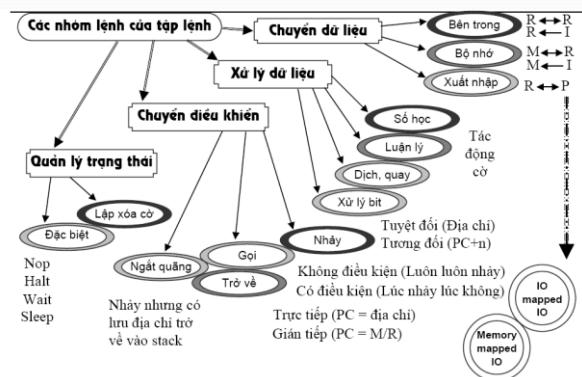
Các VXL như 80486 hay Pentium đều sử dụng 1 cách linh hoạt cả 2 loại kiến trúc bộ xử lý này, trong đó có tăng cường các kỹ thuật tiên tiến như kỹ thuật đường ống, siêu đường ống, tăng cường số thanh ghi và bộ nhớ cache nội.

02/03/2016

22

Tổ chức hoạt động của VXL (tt)

3.1.3 Tập lệnh của bộ VXL. (tt)



02/03/2016

23

Tổ chức hoạt động của VXL (tt)

3.1.4 Các vi mạch hỗ trợ cho các bộ VXL

- Các bộ VXL phải có những vi mạch hỗ trợ (đa phần là các vi mạch lập trình) thì mới có thể thiết kế được 1 đơn vị xử lý trung tâm (CPU) hoàn chỉnh trong một máy vi tính hay 1 hệ thống MVT nào đó. Sự liên kết giữa bộ VXL với các khối bên ngoài thông qua Bus hệ thống cần phải có các vi mạch giao tiếp, đệm và chốt, điều khiển bus, điều khiển trao đổi thông tin với thiết bị ngoại vi (I/O devices), điều khiển bộ nhớ Cache, các vi mạch điều khiển ngắn, điều khiển truy nhập trực tiếp bộ nhớ chính DMA.

Tuỳ theo công nghệ tích hợp và nhà sản xuất, các bộ VXL có những vi mạch hỗ trợ cho từng chức năng giao tiếp hay tổng hợp 1 số chức năng đi kèm. Các chip vi mạch hỗ trợ đi kèm với chip VXL tạo thành 1 họ (family). Các bộ VXL công nghệ cao hiện nay (kể từ 80386) đã có những vi mạch VLSI (Very Large – Scale Intergration: chip vi mạch mật độ tích hợp rất cao) hỗ trợ gồm nhiều chức năng điều khiển khác nhau được đưa vào để có thể thành 1 CPU của 1 máy vi tính (1 Mainboard).

02/03/2016

24

Tổ chức hoạt động của VXL (tt)

3.2 Tổ chức tổng quát của bộ xử lý trung tâm CPU

- Đơn vị xử lý trung tâm CPU (Central processing Unit), gọi tắt là bộ xử lý (processor), là bộ phận quan trọng nhất của máy tính. Khi được thiết kế chế tạo trên 1 chip, nó được gọi là bộ vi xử lý (microprocessor). Nhiệm vụ của bộ vi xử lý (VXL) là thực thi các chương trình chứa sẵn trong bộ nhớ chính bằng cách tìm-nạp (fetch) các lệnh của chương trình, giải mã chúng và sau đó thực hiện từng bước các thao tác trong lệnh.
- Mô hình lập trình của bộ VXL là tổ chức các thanh ghi mà người sử dụng có thể lập trình và truy nhập tới được bên trong bộ VXL. Mô hình lập trình rất thuận tiện để hiệu họ VXL khi ta xây dựng các chương trình ứng dụng.

02/03/2016

25

Tổ chức hoạt động của VXL (tt)

3.2 Tổ chức tổng quát của bộ xử lý trung tâm CPU (tt)

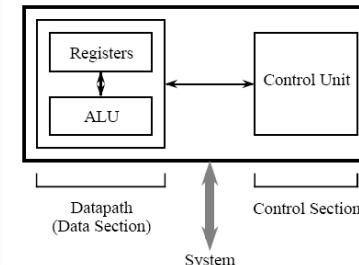
- Cấu trúc cơ bản của bộ VXL gồm 3 phần:
 - Đơn vị điều khiển
 - Đơn vị số học và logic
 - Các thanh ghi.
- Các khối này liên kết dữ liệu với nhau thông qua bus dữ liệu bên trong. Bộ VXL trao đổi thông tin với bên ngoài thông qua các đường dây dữ liệu, địa chỉ và điều khiển nới ra ngoài. Các đường dây tín hiệu này gọi là bus hệ thống.

02/03/2016

27

Tổ chức hoạt động của VXL (tt)

3.2 Tổ chức tổng quát của bộ xử lý trung tâm CPU (tt)



02/03/2016

26

Tổ chức hoạt động của VXL (tt)

a) Đơn vị điều khiển (Control unit).

- Khối logic điều khiển có liên hệ thông tin với tất cả các đơn vị trong bộ VXL, bởi vì nó điều khiển toàn bộ hoạt động xử lý thông tin bên trong bộ VXL. Kết quả giải mã lệnh được đưa đến khối logic điều khiển tạo ra chuỗi các tín hiệu để điều khiển quá trình ghi, đọc với các thanh ghi bên trong, tính toán trong ALU.
- Từ đơn vị điều khiển, thông qua bus điều khiển của hệ thống, các xung tín hiệu điều khiển tác động đến bộ nhớ, hoặc đơn vị I/O để thực hiện trao đổi dữ liệu, hoặc CU nhận các tín hiệu điều khiển từ bên ngoài (qua bus điều khiển của hệ thống) để tiếp tục quá trình xử lý bên trong bộ VXL.
- Đơn vị điều khiển thường là một khối được vi chương trình hóa (microprogrammed logic unit), tức là kiến trúc của CU giống như kiến trúc của VXL thu nhỏ.

02/03/2016

28

Tổ chức hoạt động của VXL (tt)

a) Đơn vị điều khiển (Control unit). (tt)

- Một tín hiệu quan trọng quyết định trình tự sản sinh các chuỗi tín hiệu của CU là nhịp đồng hồ của bộ VXL. Nhịp đồng hồ từ bộ tạo nhịp bên ngoài đi vào bộ VXL (đối khi bản thân bộ VXL cũng có đơn vị tạo nhịp đồng hồ), tác động đến CU, và CU từ đây tạo ra các xung tín hiệu điều khiển có chu kỳ khác nhau. CU nhận các tín hiệu điều khiển từ bên ngoài, như các tín hiệu ngắn (INT, NMI), RESET... để xử lý bên trong trước khi đưa ra các tín hiệu trả lời như: chấp nhận ngắn INTA, dừng HALT, chấp nhận yêu cầu chiếm bus BUSACK...
- Đơn vị điều khiển quyết định thứ tự làm việc của từng đơn vị trong bộ VXL và sự trao đổi thông tin với bên ngoài VXL. CU là trung tâm điều khiển của bộ VXL.

02/03/2016

29

Tổ chức hoạt động của VXL (tt)

a) Đơn vị điều khiển (Control unit). (tt)

Các tín hiệu đưa đến đơn vị điều khiển

- Clock: tín hiệu nhịp từ mạch tạo dao động bên ngoài.
- Mã lệnh từ thanh ghi lệnh đưa đến để giải mã.
- Các cờ từ thanh ghi cờ cho biết trạng thái của CPU.
- Các tín hiệu yêu cầu từ bus điều khiển

02/03/2016

31

Tổ chức hoạt động của VXL (tt)

a) Đơn vị điều khiển (Control unit). (tt)

Chức năng:

- Điều khiển nhận lệnh từ bộ nhớ đưa vào thanh ghi lệnh
- Tăng nội dung của PC (Program Count) để trỏ sang lệnh kế tiếp
- Giải mã lệnh đã được nhận để xác định thao tác mà lệnh yêu cầu
- Phát ra các tín hiệu điều khiển thực hiện lệnh
- Nhận các tín hiệu yêu cầu từ bus hệ thống và đáp ứng với các yêu cầu đó.

02/03/2016

30

Tổ chức hoạt động của VXL (tt)

a) Đơn vị điều khiển (Control unit). (tt)

Các tín hiệu phát ra từ đơn vị điều khiển

- Các tín hiệu điều khiển bên trong CPU:
 - Điều khiển các thanh ghi
 - Điều khiển ALU
- Các tín hiệu điều khiển bên ngoài CPU:
 - Điều khiển bộ nhớ
 - Điều khiển các môđun vào-ra

02/03/2016

32

Tổ chức hoạt động của VXL (tt)

b) Đơn vị số học và logic (ALU).

- Đơn vị ALU là một trong những thành phần chính bên trong bộ VXL. ALU chứa khối logic thực hiện xử lý dữ liệu. ALU có 2 cổng vào (ký hiệu In) và một cổng ra (ký hiệu là Out). Cổng vào để nhận dữ liệu từ ALU. Cổng ra để lấy kết quả xử lý dữ liệu của ALU ra ngoài.
- Dữ liệu được nhận từ các nơi khác nhau bên trong bộ VXL thông qua bus dữ liệu bên trong và được lưu ở 2 thanh ghi tạm thời (ký hiệu là Temp1 và Temp2) trong quá trình xử lý dữ liệu bên trong ALU. Trường tự, công ra kết nối với bus dữ liệu bên trong, do đó kết quả phép toán có thể được đưa tới các nơi khác nhau.
- Dữ liệu thường được gọi tới bộ cộng (accumulator). Ví dụ, khi ALU cộng 2 toán hạng, một toán hạng có trong bộ cộng, toán hạng thứ 2 nằm ở thanh ghi nào đó hay được chuyển từ bộ nhớ vào ALU thông qua bus bên trong.

02/03/2016

33

Tổ chức hoạt động của VXL (tt)

b) Đơn vị số học và logic (ALU). (tt)

- Các lệnh máy được ALU xử lý có thể là một hay hai từ (hai toán hạng). Có một số lệnh chỉ có một toán hạng, ví dụ: lệnh lấy mã bù, nghĩa là chỉ cần một cổng vào của ALU. Lệnh máy được đọc từ bộ nhớ vào bộ VXL được giải mã nhờ bộ giải mã lệnh (instruction decoder) để tạo ra chuỗi các tín hiệu đi tới ALU điều khiển quá trình xử lý dữ liệu trong ALU.
 - ALU của đa số các loại VXL có thể thực hiện các chức năng sau:
- | | |
|------------|--------------------------|
| Add (cộng) | Complement (bù nhị phân) |
| Sub | Shift Right |
| AND | Shift Left |
| OR | Increment (tăng) |
| XOR | Decrement (giảm) |

02/03/2016

34

Tổ chức hoạt động của VXL (tt)

c) Tập các thanh ghi

Chức năng và đặc điểm:

- Tập hợp các thanh ghi nằm trong CPU
- Chứa các thông tin tạm thời phục vụ cho hoạt động ở thời điểm hiện tại của CPU
- Được coi là mức đầu tiên của hệ thống nhớ
- Tuỳ thuộc vào bộ xử lý cụ thể
- Số lượng thanh ghi nhiều → tăng hiệu năng của CPU
- Có hai loại thanh ghi
 - Các thanh ghi lập trình được
 - Các thanh ghi không lập trình được

02/03/2016

35

Tổ chức hoạt động của VXL (tt)

c) Tập các thanh ghi

Phân loại thanh ghi theo chức năng:

- Thanh ghi địa chỉ: quản lý địa chỉ của ngăn nhớ hay cổng vào-ra.
- Thanh ghi dữ liệu: chứa tạm thời các dữ liệu
- Thanh ghi đa năng: có thể chứa địa chỉ hoặc dữ liệu.
- Thanh ghi điều khiển/trạng thái: chứa các thông tin điều khiển và trạng thái của CPU.
- Thanh ghi lệnh: chứa lệnh đang được thực hiện

02/03/2016

36

Tổ chức hoạt động của VXL (tt)

d) Bus dữ liệu bên trong.

Bus dữ liệu bên trong của bộ VXL đóng vai trò như là đường giao thông hai chiều chính để vận chuyển dữ liệu trao đổi giữa các đơn vị bên trong bộ VXL và với bus dữ liệu của hệ thống (system data bus).

- Người ta phân biệt 3 loại BUS :
- **BUS dữ liệu** cho phép truyền dữ liệu (các lệnh hay dữ liệu cần xử lý) giữa các thành phần của hệ thống. Tuỳ theo số dây trong mỗi BUS, người ta có thể truyền 8, 16, 32 hay 64 bit. Số các bit này có thể truyền trong cùng một thời gian (song song) gọi là độ rộng của BUS. Các thông tin có thể truyền trên 2 chiều trong một BUS như vậy, (từ bộ nhớ về đơn vị trung tâm hay từ đơn vị trung tâm về bộ nhớ), do đó BUS dữ liệu là BUS hai chiều.

02/03/2016

37

Tổ chức hoạt động của VXL (tt)

3.3 Thực hiện lệnh máy bên trong bộ VXL

- Trong chương trước ta đã biết bộ VXL thực hiện 1 lệnh theo chuỗi các bước sau :

 - (1). Lấy lệnh kế tiếp từ bộ nhớ vào thanh ghi lệnh IR.
 - (2). Chuyển nội dung bộ đếm chương trình (trỏ) tới địa chỉ lệnh tiếp theo.
 - (3). Xác định loại lệnh vừa lấy.
 - (4). Nếu lệnh sử dụng dữ liệu trong bộ nhớ, xác định nơi chứa dữ liệu
 - (5). Tìm-nạp dữ liệu vào các thanh ghi trong CPU (nếu có)
 - (6). Thực thi lệnh
 - (7). Lưu các kết quả vào nơi thích hợp.
 - (8). Trở lại bước 1 để thực thi lệnh kế tiếp.

02/03/2016

39

Tổ chức hoạt động của VXL (tt)

d) Bus dữ liệu bên trong. (tt)

- **BUS địa chỉ** được dành để truyền địa chỉ, hoặc là địa chỉ lệnh được nạp trong thanh ghi lệnh, hoặc là dữ liệu được nạp trong một thanh ghi đặt biệt hay dữ liệu được gởi đến lối vào của ALU. Độ rộng của BUS địa chỉ xác định kích thước bộ nhớ có thể địa chỉ hoá trực tiếp được. Thực tế, với BUS có độ rộng 16 bit, người ta có thể thu được 2^{16} tổ hợp các ô bộ nhớ để lưu trữ các lệnh hay dữ liệu. Trong kiểu BUS này, các địa chỉ chỉ được truyền từ đơn vị trung tâm về bộ nhớ, nên BUS này được gọi là một hướng.

- **BUS lệnh** cho phép các vi lệnh được sinh ra bởi bộ tuần tự truyền từ bộ tuần tự về các thành phần khác nhau của hệ thống.

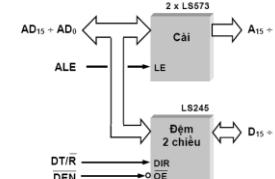
02/03/2016

38

Bộ VXL Intel 8088/8086

CPU 8086/88 :

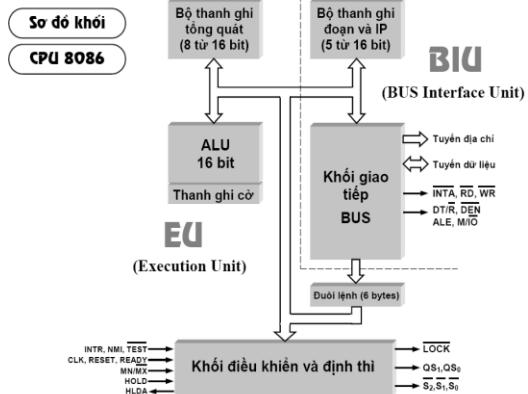
- Vì xử lý 16 bit.
- . Tuyến địa chỉ : $A_{19} + A_0 \Rightarrow$ Quản lý 1MB bộ nhớ (100 000H)
địa chỉ bộ nhớ 0000 + FFFF
- . Tuyến dữ liệu : $D_{15} + D_0$ (8086) / $D_7 + D_0$ (8088)
- . Thực tế sử dụng tuyến địa chỉ dữ liệu chung : $AD_7 + AD_0 \Rightarrow$ Cần mạch tách tuyến



02/03/2016

40

Bộ VXL 8088/8086



02/03/2016

41

Bộ VXL 8088/8086

Sơ đồ khối 8086 phân chia thành 2 nhóm cơ bản:

- **Đơn vị thực hiện EU** (EU - Execution Unit), bao gồm ALU, khối các thanh ghi 16-bit, thanh ghi cờ (flag) và đơn vị điều khiển CU. EU duy trì trạng thái CPU, kiểm soát các thanh ghi đa năng và toán hạng lệnh. Tất cả các thanh ghi và đường truyền dữ liệu trong EU dài 16-bit.
- **Đơn vị giao tiếp bus** (BIU - Bus Interface Unit), gồm có bộ cộng đê tính địa chỉ, 4 thanh ghi segment 16-bit (CS, DS, SS, ES), hàng lệnh độ dài 6 bytes được làm từ 6 byte bộ nhớ trong của bộ vi xử lý và logic điều khiển bus.

02/03/2016

42

Bộ VXL 8088/8086

- Việc chia thành 2 đơn vị làm việc đồng thời có liên hệ với nhau qua bộ đệm lệnh làm tăng đáng kể tốc độ xử lý của VXL. Các bus bên trong VXL có nhiệm vụ chuyển tải các tín hiệu của các đơn vị khác. Trong số các bus đó có bus dữ liệu 16 bit của ALU, bus các tín hiệu điều khiển ở EU và bus trong của hệ thống ở BIU.
- Trước khi đưa ra bus ngoài hoặc đi vào bus trong của bộ VXL, các tín hiệu truyền trên bus thường được cho đi qua các bộ đệm để nâng cao tính tương thích cho nối ghép hoặc nâng cao khả năng phối ghép.
- Đơn vị BIU đưa ra địa chỉ, đọc mã lệnh từ bộ nhớ, đọc/ghi dữ liệu từ/vào công hoặc bộ nhớ. Nói cách khác BIU chịu trách nhiệm đưa địa chỉ ra bus và trao đổi dữ liệu với bus.

02/03/2016

43

Bộ VXL 8088/8086

- Trong BIU còn có một bộ nhớ đệm lệnh với dung lượng 4 byte (6 byte đối với 8086) dùng để chứa các mã lệnh đọc được nằm sẵn để chờ EU xử lý (còn được gọi là *hàng đợi lệnh trước*). Đây là cấu trúc mới được cấy vào bộ VXL 8088/8086 do việc Intel đưa cơ chế *xử lý xen kẽ liên tục dòng mã lệnh* (*instruction pipelining*) vào ứng dụng trong các bộ VXL thế hệ mới.
- Trong EU có đơn vị điều khiển CU (Control Unit), bên trong nó có mạch giải mã lệnh. Mã lệnh đọc vào từ bộ nhớ được đưa đến đầu vào của bộ giải mã, các thông tin thu được từ đầu ra của nó sẽ được đưa đến mạch tạo xung điều khiển, kết quả ta thu được các xung khác nhau (tùy theo mã lệnh) để điều khiển hoạt động của các bộ phận bên trong và bên ngoài VXL.
- Trong EU còn có đơn vị tính toán số học và logic ALU (Arithmetic and Logic Unit) dùng để thực hiện các thao tác khác nhau với các toán hạng của lệnh.

02/03/2016

44

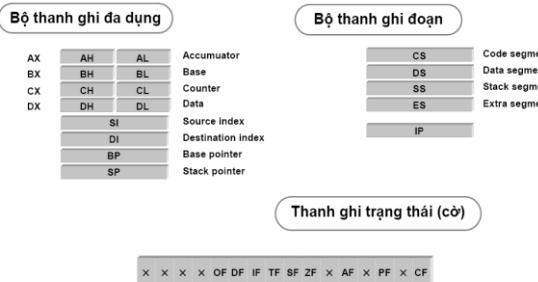
Bộ VXL 8088/8086

- Tóm lại, khi CPU hoạt động EU sẽ cung cấp thông tin về địa chỉ cho BIU để đơn vị này đọc lệnh và dữ liệu, còn bản thân nó thì giải mã lệnh và thực hiện lệnh.
- Như vậy một chu kỳ lệnh của CPU có thể được xem 1 cách đơn giản là bao gồm 2 thời khoảng: lấy lệnh từ bộ nhớ và thực hiện lệnh. Việc đọc mã lệnh (opcode fetch) là do đơn vị BIU thực hiện, việc giải mã lệnh và thực hiện lệnh do EU đảm nhiệm.
- Các khối chức năng này có khả năng làm việc đồng thời, trong hầu hết trường hợp có sự trùng lặp giữa thời gian thực hiện lệnh trước và lấy lệnh sau và các bus sẽ liên tục được sử dụng; trong khi EU lấy mã lệnh từ bộ đệm để giải mã hoặc thực hiện các thao tác nội bộ thì BIU vẫn có thể đọc mã lệnh từ bộ nhớ chính rồi đặt chúng vào bộ nhớ đệm lệnh.

02/03/2016

45

Bộ VXL 8088/8086



02/03/2016

47

Bộ VXL 8088/8086

- Bộ đệm lệnh này làm việc theo kiểu FIFO (first in – first out). Kết quả có thể coi thời gian lấy lệnh là bằng 0 vì EU chỉ việc thực hiện lệnh đã được lấy sẵn bởi BIU. Chế độ hoạt động này, còn được gọi là **pipe line** (đường ống), cho phép tăng tốc độ xử lý và được sử dụng rất nhiều trong tin học. 8086 là bộ vi xử lý đầu tiên sử dụng chế độ hoạt động này.
- Nếu có sự vào/ra liên tục của dòng mã lệnh trong bộ đệm này thì có nghĩa là có sự phối hợp hoạt động hiệu quả giữa 2 khối EU và BIU theo cơ chế xử lý xen kẽ liên tục dòng mã lệnh để làm tăng tốc độ xử lý tổng thể.
- Trong bộ VXL 8088/8086 còn có các thanh ghi 16 bit nằm trong cả 2 khối EU và BIU, ngoài ra cũng có một số thanh ghi 8 hoặc 16 bit tại EU.

02/03/2016

46

Bộ VXL 8088/8086

Các thanh ghi dữ liệu

- Các thanh ghi dữ liệu AX, BX, CX và DX được dùng như các thanh chia và các thanh ghi toán hạng 16 bit. Mỗi một thanh ghi này có thể được tách ra làm hai thanh ghi 8 bit. Ví dụ thanh ghi 16 bit AX được tách thành hai thanh ghi 8 bit AH và AL.
- Thông thường, mỗi thanh ghi có một chức năng đặc trưng sau :
 - AX được sử dụng đối với các phép toán vào ra, nhân chia ...
 - BX dùng như là thanh ghi cơ sở khi sự địa chỉ hóa gián tiếp bởi thanh ghi cơ sở.
 - CX dùng để đếm dữ liệu trong các phép toán trên những dãy ký tự
 - DX dùng với AX để nhân và chia hay thanh ghi địa chỉ gián tiếp

02/03/2016

48

Bộ VXL 8088/8086

- Các thanh ghi đoạn

Các thanh ghi đoạn CS, DS, SS và ES là một phần của đơn vị quản lý bộ nhớ được tích hợp trong 8086. Mỗi thanh ghi này chứa chia địa chỉ cơ sở vật lý của một trang 64Kb

- Các thanh ghi con trả

- Các thanh ghi SP (Stack Pointer), BP, SI và DI là những thanh ghi được sử dụng đối với các phép toán số học và logic, hai thanh ghi đầu tiên cũng được sử dụng để chỉ ra một sự dịch chuyển trong đoạn ngắn xếp trong khi hai thanh ghi cuối được sử dụng để định vị trí sự dịch chuyển trong đoạn dữ liệu.
- Thanh ghi IP (Instruction Pointer) tương ứng với bộ đếm thứ tự và chứa đựng sự dịch chuyển, bên trong đoạn chương trình được liên quan, đối với địa chỉ tham chiếu.

02/03/2016

49

Bộ VXL 8088/8086

Thanh ghi trạng thái (thanh ghi cờ)

- IF (interrupt enable flag – Cờ cho phép ngắt) : Ngắt là một chương trình con được cài sẵn trong bộ nhớ. Các chương trình này có thể được kích hoạt bởi CPU để phối hợp với các chương trình khác thực hiện. Khi IF=1, cho phép ghi nhận lời gọi ngắt và IF=0, không cho phép thực hiện lời gọi ngắt. (CLI & STI)
- DF (Direction flag – Cờ hướng) : DF sẽ quyết định việc truy xuất theo chiều tăng hay chiều giảm trong các phép truy xuất dữ liệu từ bộ nhớ có sử dụng các thanh ghi SI và DI. Nếu DF=1 thì SI và DI sẽ tự động giảm đi 1. Nếu DF=0, SI và DI sẽ tự động tăng lên 1. (STD & CLD)
- TF (Trap flag) : Cờ này dùng để đặt (TF=1) hoặc xoá (TF=0) chế độ thực hiện chương trình từng lệnh một. Cờ này được sử dụng trong các chương trình gỡ lỗi (debug).

02/03/2016

51

Bộ VXL 8088/8086

Thanh ghi trạng thái và điều khiển (thanh ghi cờ)

- CF (Carry flag – Cờ nhô) : CF được đặt (CF=1) sau khi thực hiện các phép tính số học mà trong các phép tính này có hiện tượng nhô từ bit cao nhất của toán hạng
- PF (Parity flag – Cờ kiểm tra chẵn lẻ) : PF được đặt (PF=1) khi trong 8 bit thấp của kết quả có số bit=1 là chẵn. Trong trường hợp ngược lại, PF=0.
- AF (Auxiliary flag – Cờ nhô phụ) : AF được sử dụng trong các phép toán số học với mã BCD (Binary Coded Decimal). Chẳng hạn AF=1 khi có nhô từ bit 3 sang bit 4.
- ZF (Zero flag – Cờ không) : ZF=1 khi kết quả của một phép tính bằng 0.
- SF (Sign flag – Cờ dấu) : SF=1 khi bit cao nhất của kết quả là 1 (kết quả âm), SF=0 trong trường hợp ngược lại (kết quả dương)
- OF (Overflow flag – Cờ tràn) : OF=1 khi mà trong các phép tính số học, kết quả lớn hơn hoặc bé, nó không thể chứa trong toán hạng được (sai dấu)

02/03/2016

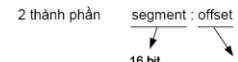
50

Bộ VXL 8088/8086

Quản lý bộ nhớ :

*Địa chỉ vật lý : dùng để thiết kế mạch
số 20 bit $A_{19}A_{18}A_{17}\dots A_1\bar{A}_0$ (00000 + FFFFF)

*Địa chỉ luân lý : dùng trong chương trình

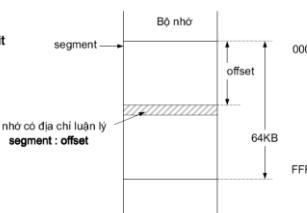


*Dùng thanh ghi đoạn chứa segment
*Dùng thanh ghi đa dụng chứa offset

FFFFh:10h

02/03/2016

52



Ô nhớ có địa chỉ luân lý

segment : offset

segment

offset

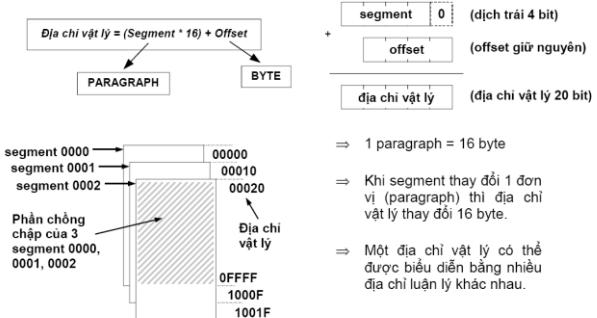
0000

64KB

FFFF

Bộ VXL 8088/8086

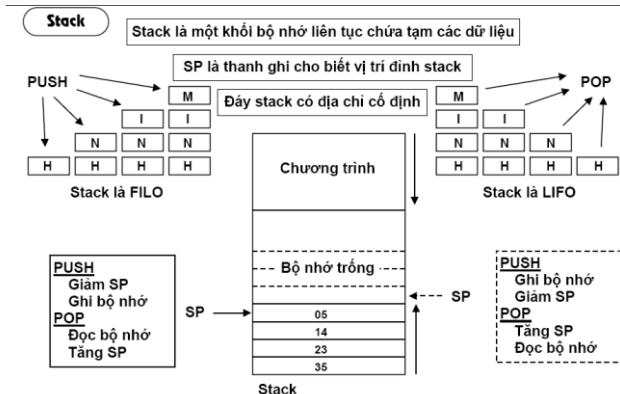
***Đổi địa chỉ luân lý ra địa chỉ vật lý :**



02/03/2016

53

Bộ VXL 8088/8086



03/03/2016

55

Bộ VXL 8088/8086

Các cặp thanh ghi mặc định :

CS : IP địa chỉ lệnh kế.
SS : SP địa chỉ định stack.
DS : SI địa chỉ đầu chuỗi nguồn (chỉ trong lệnh xử lý chuỗi).
ES : DI địa chỉ đầu chuỗi đích (chỉ trong lệnh xử lý chuỗi).

Truy xuất biến bộ nhớ : địa chỉ hiệu dụng (Effective Address)

Offset: là tổ hợp của 3 thành phần trong ngoặc vuông |

<i>Nhóm nền</i> (BX, BP)	<i>Nhóm chỉ số</i> (SI, DI)	<i>Địa chỉ trực tiếp</i> (số 16 bit)
-----------------------------	--------------------------------	---

Segment :

Không có BP	DS	[BX]	[BX+SI]	[BX+1000]	[BX+SI+1000]
Có BP	SS	[BP]	[BX+DI]	[BP+1000]	[BX+DI+1000]
		[SI]	[BP+SI]	[SI+1000]	[BP+SI+1000]
		[DI]	[BP+DI]	[DI+1000]	[BP+DI+1000]
		[1000]			

Qui tắc : không dùng các thanh ghi trong cùng nhóm với nhau

02/03/2016

54

Bộ VXL 8088/8086

To obtain other results

Các phương pháp định giá sản xuất hàng

Dựa vào một trong các đặc tính sau

- làm ngắn câu lệnh
 - đơn giản hóa câu lệnh
 - truy xuất toán hạng nhanh
 - làm câu lệnh có tính linh động

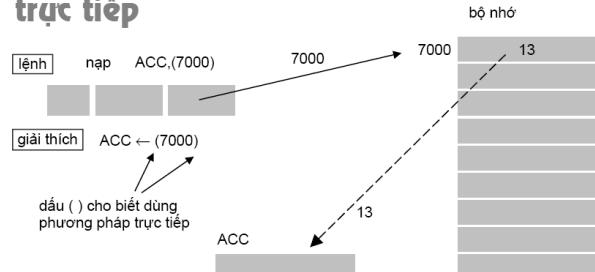
8 phương pháp

03/03/2016

56

Bộ VXL 8088/8086

trực tiếp



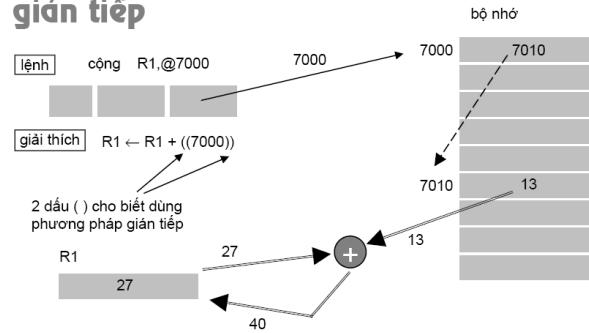
Địa chỉ trong lệnh là địa chỉ ô nhớ chứa toán hạng

02/03/2016

57

Bộ VXL 8088/8086

gián tiếp



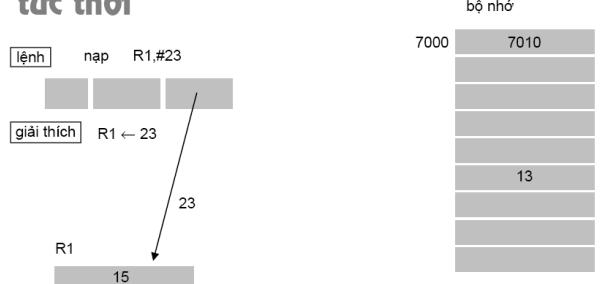
Địa chỉ trong lệnh là địa chỉ ô nhớ chứa địa chỉ toán hạng

02/03/2016

58

Bộ VXL 8088/8086

tức thời



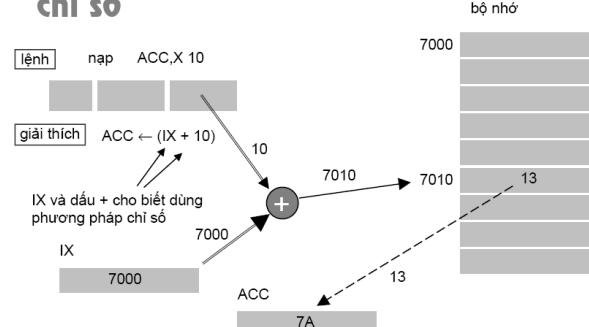
Toán hạng nằm trong lệnh

02/03/2016

59

Bộ VXL 8088/8086

chỉ số



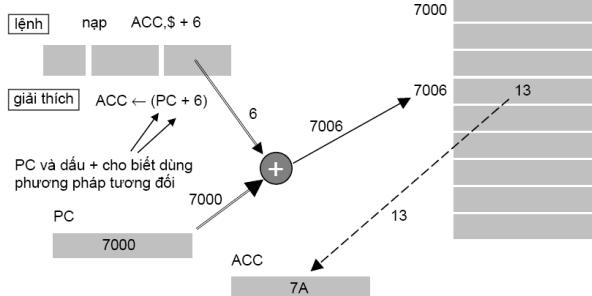
Địa chỉ toán hạng là tổng của nội dung thanh ghi chỉ số với độ dời trong lệnh

02/03/2016

60

Bộ VXL 8088/8086

tương đối



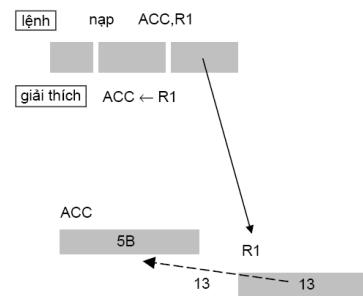
Địa chỉ toán hạng là tổng của nội dung thanh ghi PC với độ dời trong lệnh

02/03/2016

61

Bộ VXL 8088/8086

thanh ghi



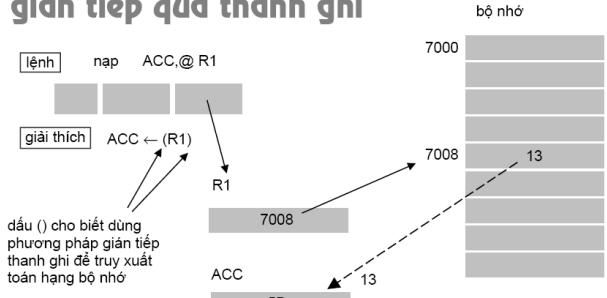
Toán hạng nằm trong thanh ghi

02/03/2016

62

Bộ VXL 8088/8086

gián tiếp qua thanh ghi



Thanh ghi chứa địa chỉ ô nhớ chứa toán hạng

02/03/2016

63

Sơ đồ chân và các tín hiệu của 8088/8086

- 8086 có thể hoạt động ở 2 chế độ (mode): MIN hoặc MAX.
 - + Chế độ MIN: CPU tự phát ra tín hiệu điều khiển cho các bus.
 - + Chế độ MAX: CPU chỉ ra phát ra các tín hiệu trạng thái tới 1 chip gọi là bộ điều khiển bus (bus controller), tiếp đó chip này sẽ dịch các tín hiệu trạng thái và phát ra các tín hiệu điều khiển tới các bus. Cách này đảm bảo ít trực trắc hơn trong quá trình đọc số liệu.

02/03/2016

64

Sơ đồ chân và các tín hiệu của 8088/8086

		Chế độ MIN	Chế độ MAX
[A / D14]	GND	1	40 V _{CC}
[A / D13]	A14	2	39 A15
[A / D12]	A13	3	38 A16 / S3
[A / D11]	A12	4	37 A17 / S4
[A / D10]	A11	5	36 A18 / S5
[A / D9]	A10	6 μP	35 A19 / S6
[A / D8]	A9	7 8088	34 SSO (BHE / S7)
	A8	8 [8086]	33 MN / MX
A / D7	A7	9	32 RD
A / D6	A6	10	31 HOLD (RQ / GT0)
A / D5	A5	11	30 HLDA (RQ / GT1)
A / D4	A4	12	29 WR (LOCK)
A / D3	A3	13	28 IO / M (S2)
A / D2	A2	14	27 DT / R (SI)
A / D1	A1	15	26 DEN (S0)
A / D0	A0	16	25 ALE (QS0)
NMI		17	24 INTA (QS1)
INTR		18	23 TEST
CLK		19	22 READY
GND		20	21 RESET

02/03/2016

65

Sơ đồ chân và các tín hiệu của 8088/8086

- Các tín hiệu của 8086/8088 phân thành các nhóm tín hiệu sau:
 - + Đường dây ghép bus dữ liệu và địa chỉ thấp (AD0 – AD7)
 - + 8 đường dây địa chỉ cao (A8 – A15).
 - + 4 đường dây địa chỉ phần cao và trạng thái (A16 / S3 – A19 / S6)
 - + Các đường dây điều khiển hệ thống (system control).
 - + Các đường dây của bus điều khiển (control bus)
 - + Các đường dây điều khiển CPU (CPU control)
 - + Các đường dây cung cấp nguồn và nhịp đồng hồ.
 - + Đồng hồ.

02/03/2016

66

Sơ đồ chân và các tín hiệu của 8088/8086

- Ta xét các tín hiệu tại các chân cụ thể như sau:
 - + AD0 – AD7 [I,O]: tín hiệu vào và ra]: Các chân dồn kênh cho các tín hiệu phân thấp của bus dữ liệu và bus địa chỉ. Xung ALE sẽ báo cho mạch ngoài biết khi nào trên các đường đó có tín hiệu dữ liệu (ALE = 0) hoặc địa chỉ (ALE = 1). Các chân này ở trạng thái trở kháng cao khi **μP** chấp nhận treo.
 - + A8 – A15 [O]: Các bit phần cao của bus địa chỉ. Các chân này ở trạng thái trở kháng cao khi **μP** chấp nhận treo.
 - + A16 / S3, A17 / S4, A18 / S5, A19 / S6 [O]: Các chân dồn kênh của địa chỉ phần cao và trạng thái. Địa chỉ A16 – A19 sẽ có mặt tại các chân đó khi ALE = 1, còn khi ALE = 0 thì trên các chân đó có các tín hiệu trạng thái S3 – S6. Các chân này ở trạng thái trở kháng cao khi **μP** chấp nhận treo.

02/03/2016

67

Sơ đồ chân và các tín hiệu của 8088/8086

Các bit trạng thái và việc truy nhập các thanh ghi đoạn

S4	S3	Truy nhập đến
0	0	Đoạn dữ liệu phụ
0	1	Đoạn ngăn xếp
1	0	Đoạn mã hoặc không đoạn nào
1	1	Đoạn dữ liệu

02/03/2016

68

Sơ đồ chân và các tín hiệu của 8088/8086

- Bit S6 = 0 liên tục, bit S5 phản ánh giá trị bit IF của thanh ghi cờ. Hai bit S3 và S4 phối hợp với nhau để chỉ ra việc truy nhập các thanh ghi đoạn (bang 3.3)
- + RD [O]: Xung cho phép đọc. Khi RD = 0 thì bus dữ liệu sẵn sàng nhận số liệu từ bộ nhớ hoặc thiết bị ngoại vi. Chân RD ở trạng thái trở kháng cao khi μP chấp nhận treo.
- + READY [I]: Tín hiệu báo cho CPU biết trình trạng sẵn sàng của thiết bị ngoại vi hay bộ nhớ.
 - Khi READY = 1 thì CPU thực hiện ghi/đọc mà không cần chèn thêm các chu kỳ đợi.
 - Ngược lại khi thiết bị ngoại vi hay bộ nhớ có tốc độ hoạt động chậm, chúng có thể đưa tín hiệu READY = 0 để báo cho CPU biết mà chờ chúng. Lúc này, CPU tự kéo dài thời gian thực hiện ghi/đọc bằng cách chèn thêm các chu kỳ đợi.

02/03/2016

69

Sơ đồ chân và các tín hiệu của 8088/8086

- + RESET [I]: Tín hiệu khởi động lại 8086. Khi RESET = 1 kéo dài ít nhất trong thời gian 4 chu kỳ đồng hồ thì 8086 bị buộc phải khởi động lại: nó xóa các thanh ghi DS, ES, SS, IP và FR về 0 và bắt đầu thực hiện chương trình tại địa chỉ CS:IP=FFFF:0000h (chú ý cờ IF ← 0 để cảm các yêu cầu ngắt khác tác động vào CPU và cờ TF ← 0 để bộ VXL không bị đặt trong chế độ chạy từng lệnh).
 - + CLK [I]: Tín hiệu đồng hồ (xung nhịp), cung cấp nhịp làm việc cho CPU.
 - + Vcc [I]: Chân nguồn. Tại đây CPU được cung cấp $+5V \pm 10\%$, 340mA.
 - + GND [O]: 2 chân nguồn để nối với điểm 0V của nguồn nuôi.
 - + MN / MX [I]: Chân điều khiển hoạt động của CPU theo chế độ MIN/MAX.
- Do 8088/8086 có thể làm việc ở 2 chế độ khác nhau nên có một số chân tín hiệu phụ thuộc vào các chế độ đó.

02/03/2016

71

Sơ đồ chân và các tín hiệu của 8088/8086

- + INTR [I]: Tín hiệu yêu cầu ngắt che được. Khi có yêu cầu ngắt mà cờ cho phép ngắt IF = 1 thì CPU dừng lệnh đang làm, sau đó nó đi vào chu kỳ chấp nhận ngắt và đưa ra bên ngoài tín hiệu INTA = 0. (STI & CLI)
- + TEST [I]: Tín hiệu tại chân này được kiểm tra bởi lệnh WAIT. Khi CPU thực hiện lệnh WAIT mà lúc đó tín hiệu TEST = 1, nó mới chờ cho đến khi tín hiệu TEST = 0 thì mới thực hiện lệnh tiếp theo.
- + NMI [I]: Tín hiệu yêu cầu ngắt không che được. Tín hiệu này không bị khống chế bởi cờ IF. Nhận được yêu cầu này, CPU dừng việc đang thực hiện, chuyển sang thực hiện chương trình phục vụ ngắt kiểu INT 2.

02/03/2016

70

Sơ đồ chân và các tín hiệu của 8088/8086

- Chế độ MIN
 - + IO/M [O]: Tín hiệu này phân biệt trong thời điểm đã định phần tử nào trong các thiết bị vào/ra (I/O) hoặc bộ nhớ (M) được chọn làm việc với CPU. Trên bus địa chỉ lùi đó sẽ có các địa chỉ tương ứng của các thiết bị đó. Chân này ở trạng thái trở kháng cao khi μP chấp nhận treo.
 - + WR [O]: Xung cho phép ghi.
 - + INTA [O]: Tín hiệu báo cho các mạch bên ngoài biết CPU chấp nhận yêu cầu ngắt INTR. Lúc này CPU đưa ra INTA = 0 để báo là nó đang chờ mạch ngoài đưa vào số hiệu ngắt (kiểu ngắt) trên bus dữ liệu.
 - + ALE [O]: Xung cho phép chốt địa chỉ. Khi ALE = 1 có nghĩa là trên bus dồn kênh AD có các địa chỉ của thiết bị vào/ra hay của ô nhớ. ALE không bao giờ bị thả nỗi (trong trạng thái trở kháng cao), khi CPU bị treo ALE = 0.
 - + DT/R [O]: Tín hiệu điều khiển các đệm 2 chiều của bus dữ liệu để chọn chiều chuyên vận của dữ liệu trên bus D. Chân này ở trạng thái trở kháng cao khi μP chấp nhận treo.

02/03/2016

72

Sơ đồ chân và các tín hiệu của 8088/8086

• Chế độ MIN

- + DEN [O]: Tín hiệu báo cho bên ngoài biết là lúc này trên bus dồn kênh AD có dữ liệu ổn định. Chân này ở trạng thái trờ kháng cao khi μP chấp nhận treo.
- + HOLD [I]: Tín hiệu yêu cầu treo CPU để mạch ngoài thực hiện việc trao đổi dữ liệu với bộ nhớ bằng phương pháp truy nhập trực tiếp bộ nhớ (direct memory access, DMA). Khi HOLD = 1, CPU 8088 sẽ tự tách ra khỏi hệ thống bằng cách treo tất cả các bus A, bus D, bus C của nó (các bus ở trạng thái trờ kháng cao) để bộ điều khiển DMA (DMA controller, DMAC) có thể lấy được quyền điều khiển hệ thống thực hiện các công việc trao đổi dữ liệu.
- + HLDA [O]: Tín hiệu báo cho bên ngoài biết yêu cầu treo CPU để dùng các bus đã được chấp nhận, và CPU 8088 đã treo các bus A, bus D và một số tín hiệu của bus C.

02/03/2016

73

Sơ đồ chân và các tín hiệu của 8088/8086

• Chế độ MIN

- + SS0 [O]: Tín hiệu trạng thái. Tín hiệu này giống như S0 trong chế độ MAX và được kết hợp với IO/M và DT/R để giải mã các chu kỳ hoạt động của bus (bảng 3.4)

Các chu kỳ của bus qua các tín hiệu SS0, IO/M và DT/R

IO/M	DT/R	SS0	kỳ điều khiển của bus
0	0	0	Đọc mã lệnh
0	0	1	Đọc bộ nhớ
0	1	0	Ghi bộ nhớ
0	1	1	Bus rỗi (nghi)
1	0	0	Chấp nhận yêu cầu ngắt
1	0	1	Đọc thiết bị ngoại vi
1	1	0	Ghi thiết bị ngoại vi
1	1	1	Dừng (Halt)

02/03/2016

74

Sơ đồ chân và các tín hiệu của 8088/8086

• Chế độ MAX (chân MN/MX nối đất)

Trong chế độ MAX một số tín hiệu điều khiển cần thiết được tạo ra trên cơ sở các tín hiệu trạng thái nhờ dùng thêm ở bên ngoài một mạch điều khiển bus 8288. Chế độ MAX được sử dụng khi trong hệ thống có mặt bộ đồng xử lý toán học 8087.

+ S2, S1 và S0 [O]: Các chân trạng thái dùng trong chế độ MAX để ghép với mạch điều khiển bus 8288. Các tín hiệu này được 8288 dùng để tạo ra các tín hiệu điều khiển trong các chu kỳ hoạt động của bus. Các tín hiệu điều khiển đó được chỉ ra trong bảng sau:

02/03/2016

75

Sơ đồ chân và các tín hiệu của 8088/8086

• Chế độ MAX (chân MN/MX nối đất)

S2	S1	S0	kỳ điều khiển của bus	Tín hiệu
0	0	0	Chấp nhận yêu cầu ngắt	INTA
0	0	1	Đọc thiết bị ngoại vi	IORC
0	1	0	Ghi thiết bị ngoại vi	IOWC, AIOWC
0	1	1	Dừng (Halt)	Không
1	0	0	Đọc mã lệnh	MRDC
1	0	1	Đọc bộ nhớ	MRDC
1	1	0	Ghi bộ nhớ	MWTC, AMWC
1	1	1	Bus rỗi (nghi)	Không

Trong hệ VXL với sự có mặt của bộ đồng xử lý toán học 8087, các tín hiệu này được mạch 8087 dùng để đóng bộ quá trình hoạt động của nó với bộ VXL 8088

02/03/2016

76

Tổ chức của bộ VXL 80286

- Bộ VXL 80286 là sự nâng cấp hoàn chỉnh của 8086.
- Nó là VXL kiến trúc 16-bit bên trong (các thanh ghi) và một bus dữ liệu bên ngoài cũng 16 bit.
- Bus địa chỉ là 24 bit cho phép đánh địa chỉ (địa chỉ hóa) tới 16MB bộ nhớ vật lý, và 1GB nhớ logic (bộ nhớ ảo). 80286 có đi kèm đồng xử lý 80287 và chúng là trung tâm của hệ máy vi tính thế hệ IBM/AT rất thông dụng trong những năm 1980.

02/03/2016

77

Tổ chức của bộ VXL 80286

- CPU 80286, sau khi được bật điện hoặc sau khi có xung reset, sẽ làm việc ở chế độ địa chỉ thực, trong chế độ địa chỉ này địa chỉ vật lý nằm trong giới hạn 1MB và được tính bằng cách cộng địa chỉ đoạn đã được dịch trái với địa chỉ offset như đã được làm với 8086.
- Địa chỉ khởi động của 80286 là FFFFFFF0h, từ địa chỉ này CPU nhảy về địa chỉ khởi động giống như ở 8086 (FFFFFF0h) để bắt đầu làm việc.
- Trong chế độ này 80286 hoạt động giống hệt 8086, nhưng nhờ những cải tiến về mặt điện tử mạch 80286 sẽ chạy nhanh hơn 8086 với cùng một tần số đồng hồ.

02/03/2016

79

Tổ chức của bộ VXL 80286

- Cấu trúc bên trong của 80286 tương tự như 8086, nhưng đặc biệt nó có đơn vị quản lý bộ nhớ bên trong MMU (Memory Management Unit) để làm việc với bộ nhớ ảo với cơ chế bảo vệ bộ nhớ và khả năng địa chỉ hóa bộ nhớ vật lý với dung lượng đạt tới 16MB, hỗ trợ cho cơ chế đa sử dụng và đa nhiệm.
- Các hệ điều hành đa nhiệm đã có thể cài đặt trên các máy tính dùng 80286.
- 80286 có thể hoạt động ở 2 chế độ: *chế độ thực* (real mode) giống như trong 8086 và *chế độ bảo vệ* (protected mode).

02/03/2016

78

Tổ chức của bộ VXL 80286

- Khi đã ở chế độ địa chỉ thực, nếu 80286 chạy dưới một hệ điều hành đa nhiệm, nó có thể được chuyển sang chế độ bảo vệ bằng cách thay đổi bit cho phép làm việc ở chế độ bảo vệ có trong thanh ghi từ trạng thái máy (MSW) của 80286.
- Lúc này bộ MMU của 80286 bắt đầu làm việc và cho phép các cơ chế bộ nhớ ảo và cơ chế bảo vệ hoạt động.

02/03/2016

80

Tổ chức của bộ VXL 80286

- Trong chế độ bảo vệ, phần cứng của CPU sẽ kiểm tra sự thâm nhập của chương trình vào dữ liệu và mã lệnh, gán cho các thâm nhập này một trong các *mức đặc quyền* (privilege levels) khi thực hiện nhiệm vụ.
- Điều này bảo vệ PC tránh khỏi va chạm giữa các dữ liệu và mã lệnh khác nhau. 80286 ngoài các lệnh thực hiện trong chế độ thực như 8086 còn được tăng cường thêm một số lệnh dùng cho chế độ bảo vệ.
- Do đó, các bản đồ địa chỉ cho các chương trình ứng dụng vẫn như cũ. Đặc điểm trên có thể hỗ trợ cho một vài chương trình (nhiệm vụ) cùng được chạy trong môi trường hệ điều hành đa nhiệm (ví dụ như OS/2).

02/03/2016

81

Tổ chức của bộ VXL 80286

- Một khi 80286 đã làm việc ở chế độ bảo vệ, để đưa nó trở lại chế độ địa chỉ thực ta chỉ có 1 cách duy nhất là tác động vào chân reset, hệ điều hành không có khả năng để chuyển qua lại giữa 2 chế độ.
- Hạn chế này sẽ được gỡ bỏ trong các bộ VXL từ 80386 trở đi.
- Chu kỳ bus có khả năng đọc và viết số liệu nhanh hơn bằng phương pháp hoạt động theo cơ chế đường ống (pipelined addressing).

02/03/2016

82

Tổ chức của bộ VXL 80286

- Các thanh ghi của 80286.**

80286 cũng có các thanh ghi thông thường với chức năng giống như các thanh ghi trong 8086, tuy nhiên để hoạt động trong chế độ bảo vệ 80286 được thêm vào các thanh ghi mới có tên như sau:

- Thanh ghi nhiệm vụ TR (Task Register)
- Thanh ghi từ trạng thái máy MSW (Machine Status Word)
- Thanh ghi bảng bộ mô tả cục bộ LDT (Local Descriptor Table)
- Thanh ghi bảng bộ mô tả ngắt IDT (Interrupt Descriptor Table)
- Thanh ghi bảng bộ mô tả toàn cục GDT (Global Descriptor Table)

02/03/2016

83

Đặc điểm của CPU 80386/80486/Pentium

- VXL 80386 và 80486-Pentium có nhiều điểm khá giống nhau, ngoại trừ việc trong 80486 đã có cáy săn bộ xử lý dấu phẩy động FPU hay trong Pentium có cả FPU lẫn 2 bộ IU (2 bộ ALU cho số nguyên 64 bit). Về mặt lập trình có thể coi các bộ VXL 80386 và 80486, Pentium là như nhau.

02/03/2016

84

Đặc điểm của VXL Intel 80386

- Bộ VXL 386 ra đời nhằm khắc phục trực tiếp các nhược điểm của 80286: phải chuyển đổi được nhanh chóng giữa chế độ thực và chế độ bảo vệ, và phải có khả năng hoạt động với bộ nhớ RAM tối đa 4 GB.
- Chip 386 còn có một bộ cache nội bộ đồng thời có thể sử dụng thêm cache ngoài để tăng tốc độ hoạt động.
- 80386 là một bộ VXL mạnh và tối ưu với các hệ điều hành đa nhiệm. So với VXL Intel 8086, nó có thêm những đặc điểm chính như sau:
 - Bus dữ liệu 32 bit
 - Bus địa chỉ 32 bit, do vậy có thể địa chỉ hóa được 4GB nhớ vật lý và 64Terabyte bộ nhớ ảo (=16404 Gb).

02/03/2016

85

Đặc điểm của VXL Intel 80386

- Hơn nữa, để cải tiến hiệu quả, 386 nạp trước 16 byte lệnh trong quá trình thực hiện.
 - Tốc độ thực hiện lệnh lên tới 25 triệu lệnh/giây (MIPS).
 - Thêm bộ quản lý trang của khối nhớ. Có chế độ quản lý và bảo vệ bộ nhớ dùng cho các hệ điều hành đa nhiệm (chế độ thực, chế độ bảo vệ, chế độ 8086 ảo).

02/03/2016

86

Đặc điểm của VXL Intel 80386

- Các thanh ghi của 80386:
 - Các thanh ghi địa chỉ và số liệu thông dụng 32 bit: EAX, EBX, ECX, EDX, ESI, EDI, EBP, ESP.
 - Các thanh ghi cờ và con trỏ lệnh 32 bit: EIP, EFLAGS.
 - Các thanh ghi đoạn 16 bit: đoạn dữ liệu (DS, ES, FS, GS), CS, SS.
 - Các thanh ghi điều khiển 32 bit: CR₀, CR₁, CR₂, CR₃; ghi số liệu để điều khiển VXL

02/03/2016

87

Đặc điểm của VXL Intel 80386

- Các thanh ghi của 80386:
 - Các thanh ghi mô tả mảng: GDTR (mô tả toàn cục), LDTR (mô tả cục bộ), TR (ghi nhiệm vụ).
 - Các thanh ghi gỡ rối 32 bit ($DR_0 \div DR_7$): ghi địa chỉ và trạng thái của các điểm dừng chương trình để kiểm tra lỗi.
 - Các thanh ghi kiểm tra (thử) 32 bit: TR₆ (Test Control: kiểm tra điều khiển), TR₇ (Test Status)
 - 80386 không có FPU (đơn vị dấu phẩy động) bên trong, mà phải sử dụng bộ đồng xử lý 80387 bên ngoài để thực hiện các phép tính dấu phẩy động.

02/03/2016

88

Đặc điểm của VXL Intel 80386

- Hộ 80386 có 2 phiên bản:

- 80386DX (10/1985): 40MHz; 6MIPS, 0,275 triệu tranzito, có bus dữ liệu là 32 bit.
- 80386SX (6/1988): 33MHz; 2,5MIPS, 0,275 triệu tranzito, có bus dữ liệu là 16 bit.

02/03/2016

89

Đặc điểm của VXL Intel 80386

- VXL 80386 có thể làm việc ở một trong 3 chế độ:

- **Chế độ địa chỉ thực**

Trong chế độ địa chỉ thực, 80386 hoạt động ở chế độ phân đoạn với khả năng phân biệt địa chỉ bộ nhớ 1MB và với các đoạn 64KB như 8086.

Các thanh ghi đa năng có thể có kích thước **8-32 bit** tùy theo toán hạng. Ta có thể làm việc với 4 thanh ghi đoạn dữ liệu. Thanh ghi cờ cho chế độ địa chỉ thực chỉ là 16 bit.

Khi bật điện hoặc sau khi có xung reset 80386 làm việc ở chế độ địa chỉ thực. Mục đích chính của chế độ địa chỉ thực là để khởi đầu cho 80386 chuẩn bị cho chế độ bảo vệ.

02/03/2016

90

Đặc điểm của VXL Intel 80386

➤ Chế độ 8086 ảo

Một tính năng mới của 80386 là có thể mô phỏng một hoặc nhiều bộ VXL 8086 cùng một lúc nên cho phép chạy nhiều chương trình DOS đồng thời.

Chế độ đó được gọi là chế độ **8086 ảo**, tức là cho phép các chương trình ứng dụng của 8086 chạy được trên 80386 ở chế độ bảo vệ.

Bộ VXL 80386 chuyển đổi rất dễ dàng từ chế độ bảo vệ sang chế độ **8086 ảo** hoặc ngược lại bằng cách lập/xóa bit VM trong thanh ghi EFLAGS.

02/03/2016

91

Đặc điểm của VXL Intel 80386

➤ Chế độ 8086 ảo

❖ Trong chế độ này, VXL 80386 xem mỗi chương trình là 1 tác vụ rồi dành cho mỗi tác vụ một không gian địa chỉ 1MB và tính địa chỉ vật lý theo **địa chỉ segment:offset** như 8086.

❖ Các không gian 1MB này được VXL 80386 quản lý theo kiểu phân trang nên có thể ném rác trong không gian tổng thể 4GB mà 80386 có khả năng phân biệt được.

❖ Chính vì vậy có thể dùng các bit U/S và R/W để tạo ra cơ chế bảo vệ cho bộ nhớ trong chế độ 8086 ảo này.

02/03/2016

92

Đặc điểm của VXL Intel 80386

➤ Chế độ bảo vệ

Chế độ bảo vệ cho phép ta khai thác hết sức mạnh được thiết kế trong cho VXL 80386.

Mạch 80386 chuyển chế độ làm việc từ chế độ địa chỉ thực sang chế độ bảo vệ hoặc ngược lại nhờ việc lập/xóa bit D₀ của thanh ghi điều khiển CR₀.

Trong chế độ bảo vệ 80386 có thể làm việc theo phương pháp phân đoạn hoặc phân trang (sẽ trình bày ở chương sau). Mỗi đoạn có thể có độ dài tới 4GB, mỗi trang là 4KB và bộ nhớ ảo có thể có độ dài lớn nhất là 64TB.

02/03/2016

93

Đặc điểm của VXL Intel 80486

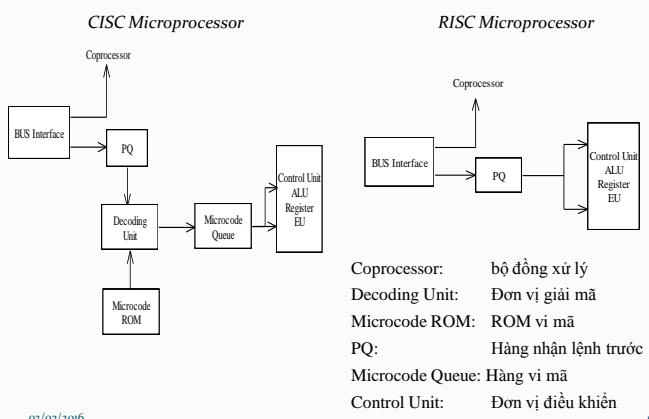
➤ Bộ VXL 80486 về cơ bản có quy mô như 80386, nhưng nó được tăng cường thêm ngay ở bên trong một bộ nhớ cache (Cache nội) với dung lượng 8KB dùng cho lệnh và dữ liệu.

➤ VXL 80486 còn được cấy thêm một bộ đồng xử lý toán học dấu phẩy động (*floating point unit, FPU*). Khả năng thao tác của 80486 vì thế cao hơn, tuy khả năng bảo vệ và quản lý bộ nhớ, khả năng phân biệt địa chỉ bộ nhớ thì vẫn giống như 80386.

02/03/2016

94

Bộ xử lý CISC và RISC



02/03/2016

95

Bộ xử lý CISC và RISC

➤ Họ VXL của Intel thường hoạt động theo nguyên tắc dùng VXL có *tập lệnh phức hợp CISC (Complex Instruction Set Computer)*.

➤ Đặc điểm là nó có số lượng lệnh lớn (thường hơn 300 lệnh), có khả năng định địa chỉ phức tạp, các lệnh được vi lệnh hoá có nghĩa là các CPU hoạt động bằng vi chương trình.

02/03/2016

96

Bộ xử lý CISC và RISC

- Trong đó, bộ xử lý nhận lệnh qua ghép nối bus đưa vào hàng nhận lệnh trước PQ. Sau đó PQ sẽ phát lệnh tới bộ giải mã.
- Bộ giải mã sẽ giải mã 1 lệnh máy thành các vi lệnh và truyền chúng tới hàng vi mã, từ đây các vi lệnh được truyền tới các đơn vị điều khiển và chạy lệnh sao cho bộ ALU và các thanh ghi được điều khiển 1 cách thích hợp.
- Do vậy, trong bộ VXL kiểu CISC, thời gian thi hành 1 lệnh mã máy bao gồm thời gian giải mã và khoảng thời gian thực hiện chương trình vi lệnh tương ứng.

02/03/2016

97

Bộ xử lý CISC và RISC

- Ngày nay, các bộ nhớ chính có thời gian truy cập ngắn hơn 80 ns làm cho các ưu điểm kể trên của vi chương trình không những biến mất mà còn gây cản trở nữa cho hệ thống.
- Trong nhiều thập kỷ, các nhà khoa học máy tính đã nghiên cứu cách thức mà các máy tính hiện nay chạy chương trình. Điều tra thống kê đã chỉ ra những kết luận sau:

02/03/2016

99

Bộ xử lý CISC và RISC

- Vì chương trình được phát triển mạnh trong những năm 70 khi bộ nhớ chính có tốc độ truy cập tương đối chậm.
- Lúc đó các nhà thiết kế máy tính sử dụng bộ nhớ chính tốc độ chậm để lưu trữ các lệnh phức tạp tạo nên chương trình ở mức máy.
- Các lệnh đó sẽ được dịch bằng vi mã trong bộ nhớ điều khiển vi chương trình có tốc độ truy nhập nhanh hơn.

02/03/2016

98

Bộ xử lý CISC và RISC

- Trong nhiều chương trình điển hình được xử lý bằng bộ VXL CISC thì đến 80% thời gian chạy chương trình được thực hiện chỉ bởi 20% số lệnh trong tập lệnh.
- Ví dụ, tác giả Fairclough đã chia các lệnh ra thành 8 nhóm và đã thống kê được các kết quả trung bình tần suất sử dụng (xét về cả chương trình và kiến trúc máy tính) của chúng như sau:

Nhóm	1	2	3	4	5	6	7	8
Tần suất %	45,28	28,73	10,75	5,92	3,91	2,93	2,05	0,44

02/03/2016

100

Bộ xử lý CISC và RISC

▪ Trong đó:

- 1 là nhóm các lệnh chuyển dữ liệu
- 2 là nhóm các lệnh chuyển điều khiển chương trình (call, jmp, ret...)
- 3 là nhóm các lệnh số học
- 4 là nhóm các lệnh so sánh
- 5 là nhóm các lệnh logic
- 6 là nhóm các lệnh dịch
- 7 là nhóm các lệnh xử lý bit
- 8 là nhóm các lệnh vào/ra

02/03/2016

101

Bộ xử lý CISC và RISC

Rõ ràng trong tổng số 8 nhóm lệnh trên thì 2 nhóm lệnh chuyển dữ liệu và chuyển điều khiển chương trình có tần suất sử dụng chiếm đến 74%.

+ Có những trường hợp, để đạt được cùng 1 kết quả thì việc chạy một chuỗi lệnh đơn giản sẽ nhanh hơn là chạy một lệnh phức tạp.

02/03/2016

102

Các đặc điểm của kiến trúc RISC

Kiến trúc RISC cần có những đặc điểm sau:

- Cần có đủ số thanh ghi (thực chất là các bộ nhớ ngay trong chip, có thời gian truy cập nhanh hơn so với bộ nhớ ngoài) để khắc phục được hiệu ứng ngẽn cổ chai (bottleneck).
- Vì các lệnh trong nhóm 2 (call, ret, ...) thường xảy ra nên một kiến trúc hiệu dụng cần tạo sự dự phòng cho việc chuyển hiệu quả các tham số giữa các thủ tục.

02/03/2016

103

Bộ xử lý CISC và RISC

→ Những kết luận trên là tiền đề để một loại VXL khác ra đời; nó hoạt động theo phương pháp sử dụng *tập lệnh rút gọn RISC (Reduced Instruction Set Computer)*.

→ Tập lệnh ở đây được hạn chế vào các lệnh chính yếu cần thiết trong số 20% lệnh nói trên, và phần cứng được thiết kế sao cho các phần tử RISC hiểu ngay được lệnh mã máy do chương trình cung cấp.

→ Do đó, các lệnh này không cần chuyển cho bộ giải mã vì lệnh nữa mà được thực hiện ngay. Rõ ràng thời gian để thực hiện 1 lệnh ở đây được rút ngắn đi nhiều so với trường hợp CISC (thực hiện 1 mã thao tác chỉ trong 1 chu kỳ máy).

02/03/2016

104

Các đặc điểm của kiến trúc RISC

Kiến trúc RISC cần có những đặc điểm sau:

- Nếu một bộ xử lý RISC đủ nhanh thì nó phải chạy được một lệnh trong một chu kỳ máy. Điều này cũng có nghĩa là sự phức tạp của các lệnh phải có giới hạn của chúng.
- Kết quả tất yếu suy ra từ điều trên là một kiến trúc có hiệu quả không nên được vi chương trình hóa vì nó phải dịch một lệnh bằng việc chạy các vi lệnh. Một bộ xử lý RISC gần với một kiến trúc vi chương trình trong đó sự phân biệt giữa chu trình máy và chu trình vi lệnh bị biến mất.

02/03/2016

105

Các đặc điểm của kiến trúc RISC

Kiến trúc RISC cần có những đặc điểm sau:

- Bộ xử lý hiệu quả phải có định dạng lệnh đơn. Khi đó việc giải mã lệnh có thể được thực hiện bằng một mạch logic giải mã tối thiểu.
- Để giảm thời gian chạy lệnh, các lệnh dài có liên quan tới các vi xử lý thế hệ 1 và 2 cần phải bỏ đi.

02/03/2016

107

Các đặc điểm của kiến trúc RISC

Kiến trúc RISC cần có những đặc điểm sau:

- Bộ xử lý hiệu quả phải có định dạng lệnh đơn. Khi đó việc giải mã lệnh có thể được thực hiện bằng một mạch logic giải mã tối thiểu.
- Để giảm thời gian chạy lệnh, các lệnh dài có liên quan tới các vi xử lý thế hệ 1 và 2 cần phải bỏ đi.

02/03/2016

106

Các đặc điểm của kiến trúc RISC

→ 3 khía cạnh cơ bản của kiến trúc RISC:

- Dùng một số lớn các thanh ghi được chế tạo trên chip (on-chip),
- Dùng các cửa sổ thanh ghi gói lên nhau để giảm một cách đáng kể sự truyền tham số giữa các thủ tục,
- Cơ chế đường ống (pipelining) trong đó cho phép sự gói lên nhau trong việc thực hiện lệnh.

02/03/2016

108

Các đặc điểm của kiến trúc RISC

Kiến trúc RISC cần có những đặc điểm sau:

- Trong kiến trúc RISC do nhóm Berkeley đề xướng sử dụng định dạng lệnh 3 thanh ghi (*hình 3.7*). Mỗi trường toán hạng cho phép một trong số 32 thanh ghi bên trong được truy cập.

OpCode	SCC	Đích	Nguồn 1	IM+8 zeros	Nguồn 2
7 bit	1 bit	5 bit	5 bit	9 bit	5 bit

32 bit

02/03/2016

109

Các đặc điểm của kiến trúc RISC

- Hai trường SCC và IM+8 zeros liên quan đến việc cập nhật thanh ghi mã điều kiện và toán hạng tức thời.
- Định dạng RISC cho phép các lệnh dạng ADD X,Y,Z trong đó X, Y, Z là các thanh ghi bên trong. Vì mỗi trường toán hạng dài 5 bit nên RISC có $2^5=32$ thanh ghi bên trong.
- Nhưng trong thực tế, RISC có đến 138 thanh ghi đa năng bên trong dùng cho người sử dụng. Nguyên nhân khác nhau này là do một cơ chế gọi là cửa sổ, cơ chế đó cho phép người lập trình chỉ nhìn thấy một tập nhỏ trong số toàn bộ các thanh ghi trong một thời điểm nào đó.

02/03/2016

110

Các đặc điểm của kiến trúc RISC

- Kiến trúc RISC và xử lý đường ống.

Instruction Fetch	Instruction Decode	Operand Fetch	Execute	Operand Store
IF	ID	OF	E	OS

- 1) *Nhận lệnh* (Instruction Fetch): lấy lệnh từ bộ nhớ và tăng giá trị thanh ghi đếm chương trình PC lên 1.
- 2) *Giải mã lệnh* (Instruction Decode): Giải mã lệnh được đọc từ bộ nhớ trong giai đoạn trước. Bản chất của giai đoạn này rất phụ thuộc vào tính phức tạp của việc lập mã lệnh. Một lệnh thông thường có thể được giải mã trong một vài ns trong khi 1 lệnh phức tạp có thể đòi hỏi phải qua một bảng tra cứu nạp sẵn trong ROM để giải mã.

02/03/2016

111

Các đặc điểm của kiến trúc RISC

- Kiến trúc RISC và xử lý đường ống.

Instruction Fetch	Instruction Decode	Operand Fetch	Execute	Operand Store
IF	ID	OF	E	OS

- 3) *Nhận toán hạng* (Operand Fetch): Toán hạng của lệnh được đọc từ bộ nhớ hệ thống hoặc từ 1 thanh ghi trên chip rồi nạp vào bộ xử lý.
- 4) *Thực hiện lệnh* (Execute): phép toán lệnh được thực hiện
- 5) *Lưu trữ toán hạng* (Operand Store): kết quả của giai đoạn thực hiện lệnh được viết vào toán hạng đích. Đó có thể là 1 thanh ghi trên chip hoặc 1 vùng nhớ ngoài.

02/03/2016

112

Các đặc điểm của kiến trúc RISC

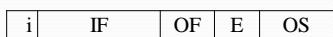
- Kiến trúc RISC và xử lý đường ống.
- Mỗi giai đoạn kể trên được thực hiện trong 1 khoảng thời gian nhất định (thường là 1 bội số nguyên lần của chu kỳ nhịp đồng hồ hệ thống).
- Một vài lệnh có thể đòi hỏi ít hơn 5 giai đoạn trên; ví dụ lệnh CMP R1,R2 thực hiện so sánh R1 và R2 để đặt các mã điều kiện (các cờ) nên không cần qua giai đoạn lưu trữ toán hạng.

02/03/2016

113

Các đặc điểm của kiến trúc RISC

- Kiến trúc RISC và xử lý đường ống.
- Nói 1 cách đơn giản, một xử lý đường ống có thể cho phép thực hiện lệnh thứ i trong khi lấy lệnh thứ i+1 trong cùng 1 thời gian.
- Hình sau mô tả cách mà bộ xử lý RISC thực hiện xử lý đường ống:



02/03/2016

115

Các đặc điểm của kiến trúc RISC

- Kiến trúc RISC và xử lý đường ống.
- Tuy nhiên, với cách sắp trên có thể dẫn đến không hiệu quả.
- Ta xét giai đoạn thực hiện lệnh. Giai đoạn này được diễn ra trong *đơn vị thực hiện lệnh*, nó chỉ chiếm mất một phần năm chu trình lệnh; nói cách khác *đơn vị thực hiện lệnh* trong bộ xử lý sẽ không làm gì trong 80% thời gian còn lại của chu trình lệnh.
- Vì lý do này, kỹ thuật đường ống (pipelining) đã được áp dụng nhằm nâng cao tốc độ xử lý hiệu dụng lên bằng cách gộp lên nhau về mặt thời gian các giai đoạn khác nhau trong việc thực hiện lệnh.

02/03/2016

114

Các đặc điểm của kiến trúc RISC

- Kiến trúc RISC và xử lý đường ống.
- | | | | | |
|---|----|----|---|----|
| i | IF | OF | E | OS |
|---|----|----|---|----|

i + 1	IF	OF	E	OS
-------	----	----	---	----
- RISC gộp giai đoạn lấy lệnh từ bộ nhớ ngoài với 3 giai đoạn xảy ra bên trong chip: lấy toán hạng, thực hiện lệnh và lưu trữ toán hạng.
- Lưu ý rằng 3 giai đoạn này chỉ chiếm mất 1 thời gian xấp xỉ như giai đoạn lấy lệnh vì các hoạt động xảy ra ngay bên trong bộ xử lý và các toán hạng được nhận hoặc lưu trữ vào ngay các thanh ghi của bộ xử lý nên có tốc độ xử lý rất nhanh.
- Hơn nữa vì RISC có 1 định dạng lệnh thông thường rất đơn giản nên giai đoạn giải mã lệnh là không cần thiết.

02/03/2016

116

VXL Intel 80486

80486 bao gồm các thành phần chính sau đây:

- Giao tiếp bus (bus interface), nó kết hợp với bus hệ thống bên ngoài, cache bên trong chip và các đơn vị tiền đọc lệnh (prefetcher units).
- Đơn vị tiền đọc lệnh (prefetcher) với hàng 32-byte cho các lệnh và nó được nối với giao tiếp bus, cache, giải mã lệnh và đơn vị phân đoạn (segmentation unit).

02/03/2016

117

VXL Intel 80486

80486 bao gồm các thành phần chính sau đây:

- Đơn vị cache(cache unit), gồm có 8KB cache (chứa dữ liệu và mã lệnh) và logic quản lý cache. Nó được kết nối với đơn vị segmentation, ALU và FPU thông qua bus 64-bit truyền dữ liệu giữa các khối (interunit data transfer bus). Đơn vị cache cũng kết nối trực tiếp với đơn vị định trang (paging unit), giao tiếp bus, bộ tiền đọc lệnh.
- Đơn vị giải mã lệnh nhận 3 bytes của lệnh đọc từ hàng lệnh và chuyển lệnh được giải mã tới đơn vị điều khiển và kiểm tra bảo vệ CPTU (control and protection test unit).

02/03/2016

118

VXL Intel 80486

80486 bao gồm các thành phần chính sau đây:

- CPTU tạo ra các vi lệnh và gửi chúng tới các đơn vị khác để thực hiện kiểm tra bảo vệ.
- Một đơn vị tính toán số nguyên tốt ưu, tương thích với 386DX
- Một đơn vị tính toán số thực (FPU-Floating Processor Unit) tương thích với coprocessor toán học 80387.

02/03/2016

119

VXL Intel 80486

80486 bao gồm các thành phần chính sau đây:

- Đơn vị phân đoạn (segmentation unit), trong đó có logic quản lý phân đoạn, logic điểm ngắt.
- Đơn vị định trang (paging unit), trong đó có logic quản lý định trang và bộ đệm chuyển đổi 32 điểm vào TLB.

02/03/2016

120

VXL Intel 80486

- Kiến trúc của i486 là dạng mở rộng của i386, cả i386 và i486 đều có cache và MMU bên trong, nhưng i486 có cache bên trong dung lượng lớn hơn.
- Kiến trúc của VXL Intel 80486 được thiết kế dựa trên kết quả của việc tích hợp 2 loại phần tử xử lý CISC và RISC trong cùng 1 chip.
- Trong 486, các lệnh mã máy thường xuyên được dùng sẽ được cài đặt sẵn và có thể thực hiện ngay, trong khi các lệnh phức tạp nhất và ít được sử dụng sẽ thông qua bộ giải mã vi lệnh.
- Theo phương thức này, hầu như tất cả các lệnh có thể được thực hiện gọn trong 1 chu kỳ xung nhịp đồng hồ.

02/03/2016

121

VXL Intel 80486

- Mạch cache liên hiệp 4 đường trong 486 làm việc giống như cache ngoại nhưng đường cache ở đây là 128 bit chứ không phải 32 bit.
- Trong 486 có đường ống lệnh 5 tầng cho phép chạy các lệnh nhanh hơn ở 386 nhiều, cơ chế này thường dùng trong VXL RISC cho phép 1 vài lệnh có thể “chảy” trong đường ống cùng 1 lúc.
- 486 sẽ nhận 1 vài lệnh trước và trong khi đang xử lý 1 lệnh nó sẽ giải mã và khởi phát chạy lệnh tiếp theo sớm đến mức có thể.

02/03/2016

122

VXL Intel 80486

- Thực sự thì 486 có thể thực thi các phần của 1 vài lệnh trong cùng 1 thời gian.
- Ví dụ giả sử 486 thực hiện 1 chuỗi lệnh có ý nghĩa như sau:

```

MOV    AX, <nội dung vùng nhớ>
ADD    CX,BX
SHR    AX,1
MOV    <nội dung vùng nhớ>, CX

```

02/03/2016

123

VXL Intel 80486

- Một vài nhịp đồng hồ trước khi thực hiện lệnh thứ nhất, 486 đã lấy tất cả các lệnh đơn giản và khởi phát giải mã chúng.
 - Khi lệnh đầu tiên được thực hiện thì việc giải mã lệnh thứ 2 đã hoàn thành.
 - Vì lệnh ADD không dùng các bus hoặc số liệu đọc từ bộ nhớ bởi lệnh MOV nên nó có thể được thực hiện trước khi lệnh MOV hoàn thành.

02/03/2016

124

VXL Intel 80486

- Tương tự như vậy, sự giải mã lệnh SHR sẽ được hoàn thành trong khi lệnh ADD CX,BX đang thực hiện; và trong chu kỳ nhịp tiếp theo lệnh SHR sẽ được thực hiện.
- Trong chu kỳ nhịp mà lệnh SHR được thực hiện, sự giải mã lệnh cuối cùng được hoàn thành và địa chỉ của vùng nhớ sẽ được đưa ra bus địa chỉ.
- Trong chu kỳ đồng hồ tiếp theo, từ trong thanh ghi CX sẽ được đưa ra trên bus số liệu.

02/03/2016

125

VXL Intel 80486

- Việc tăng cường sự xen phủ như vậy cho phép 486 chạy nhiều lệnh trong một chu kỳ đồng hồ một cách hiệu dung.
- Việc nhận lệnh, giải mã lệnh và thực hiện mỗi lệnh thực tế xảy ra vài nhịp đồng hồ nhưng vì các hoạt động này xen phủ lên nhau với các quá trình giải mã và thực hiện lệnh nên thời gian thực cho mỗi lệnh chỉ mất có 1 nhịp đồng hồ.
- Ví dụ, 1 hoạt động viết bộ nhớ 16 bit mất chừng 22 nhịp đồng hồ trong VXL 8088 và 4 nhịp trong 386 thì chỉ diễn ra 1 nhịp trong 80486.

02/03/2016

126

VXL Intel 80486

- Cơ chế bảo vệ của i386 và i486: Mỗi loại vi xử lý có chức năng hỗ trợ cho đa xử lý và đa nhiệm có cơ chế bảo vệ bộ nhớ khác nhau, tuy nhiên chúng có 1 số điểm chung:
 - Cơ chế bảo vệ bộ nhớ chủ định cho các hệ điều hành đa nhiệm (multitasking OS) để cách ly và bảo vệ các chương trình của người sử dụng và hệ điều hành khỏi bị xâm nhập không được phép (không có thẩm quyền).
 - Cơ chế bảo vệ phân chia thành các cấp bảo vệ, thường gọi là cấp thẩm quyền PL (privilege levels).

02/03/2016

127

VXL Intel 80486

- Cơ chế bảo vệ của i386 và i486: Có 4 cấp thẩm quyền, theo thứ tự từ cao đến thấp:
 - Cấp 0: cấp thẩm quyền cao nhất, giành cho nhân của hệ điều hành (OS kernel)
 - Cấp 1: giành cho các dịch vụ của hệ thống
 - Cấp 2: giành cho các mở rộng hệ thống
 - Cấp 3: cấp thẩm quyền thấp nhất, giành cho các ứng dụng của người sử dụng.

02/03/2016

128

VXL Pentium

- Năm 1993, Intel cho ra đời một bộ vi xử lý mới, không phải mang tên i586 mà là Pentium.
- Đây là VXL CISC được tích hợp thêm một số tính chất của VXL RISC.
- VXL Pentium không chỉ nhanh hơn 80486 mà còn có 1 số đặc điểm mới được thiết kế đặc biệt cho các ứng dụng phục vụ file.

02/03/2016

129

VXL Pentium

- Nó cũng có hai bộ nhớ cache riêng biệt 8Kb, một cho các lệnh và một cho dữ liệu, một đơn vị tính toán số thực tốt ưu và 2 pipeline 5 tầng.
- Nhịp đồng hồ là 60, 66 và có thể đạt đến 100MHz. Dưới một vài điều kiện, công suất của nó có thể nhanh gấp 4 hay 5 lần i486.

02/03/2016

131

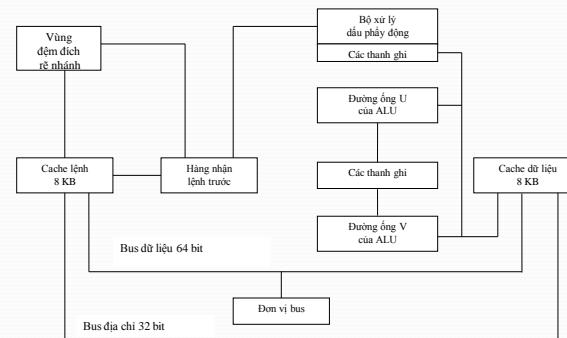
VXL Pentium

- Bus dữ liệu rộng 64 bit tuy vẫn chỉ có các thanh ghi 32 bit bên trong.
- Chip có khả năng thực hiện 2 lệnh song song (nó có hai đơn vị thực hiện độc lập 32 bit, có khả năng thực hiện đồng thời), theo công nghệ siêu vô hướng (superscalar) với 2 đường ống U và V. Công nghệ này có khả năng làm việc với các cấu trúc RISC.

02/03/2016

130

VXL Pentium



02/03/2016

132

VXL Pentium

- Hai bộ nhớ Cache 8KB riêng biệt, một dành cho lệnh và 1 dành cho số liệu.
- Hai hàng nhận lệnh trước 32 bit và 2 bộ tính toán số học và logic ALU được cung cấp lệnh và dữ liệu trên 2 đường ống song song siêu vô hướng U và V cho phép thi hành 2 lệnh máy trong 1 chu kỳ.
- Bộ xử lý vẫn làm việc với các thanh ghi 32 bit và bus địa chỉ 32 bit nhưng có thể kết nối với bus dữ liệu ngoài là 64 bit.

02/03/2016

133

VXL Pentium

- Bus nội bộ là 64 bit và 128 bit, bus từ bộ nhớ cache nội nối với các vùng đệm của các ALU rộng 256 bit cho phép tăng tốc độ trao đổi dữ liệu trong bộ nhớ VXL rất cao.
- Có một vùng nhớ gọi là vùng đệm đích rẽ nhánh BTB (Branch Target Buffer) của lệnh đối với 256 lệnh rẽ nhánh sử dụng gần đây nhất.

02/03/2016

134

VXL Pentium

- Một vài đặc trưng khác làm cho Pentium phân biệt được với các VXL trước là:
 - Kích thước trang của bộ nhớ trong chế độ bảo vệ và ảo không chỉ hạn chế ở 4KB mà có thể lên đến 2 MB hay 4 MB.
 - Khả năng tự tìm kiếm các lỗi phức tạp và gỡ rối (debugging) bằng các mạch phần cứng lắp thêm.

02/03/2016

135

VXL Pentium

- Một vài đặc trưng khác làm cho Pentium phân biệt được với các VXL trước là:
 - Có thể tự đánh giá được hiệu suất thi hành chương trình.
 - Nhờ việc sử dụng các cơ chế đường ống, cấu trúc superscalar và dự đoán rẽ nhánh.
 - Các lệnh chạy trong Pentium được thực hiện trên 1 cấu trúc superscalar 2 đường ống song song U và V, các cache nội bộ và vùng đệm đích rẽ nhánh BTB.

02/03/2016

136

VXL Pentium

- *Cấu trúc superscalar với 2 đường ống song song:* Việc thi hành lệnh trong Pentium diễn ra qua 5 công đoạn:
 - 1) Lấy lệnh trước (Prefetch): lệnh máy được lấy từ bộ nhớ chuyên vào bộ xử lý.
 - 2) Giải mã lệnh 1 (Decode 1): giải mã, phân tích để xác định loại lệnh.
 - 3) Giải mã lệnh 2 (Decode 2): xác định địa chỉ ô nhớ cần truy nhập
 - 4) Thực hiện lệnh (Execute).
 - 5) Viết các kết quả dữ liệu vào các thanh ghi và bộ nhớ

02/03/2016

137

VXL Pentium

- *Cấu trúc superscalar với 2 đường ống song song:*
 - Các công đoạn trên cũng được Pentium thực hiện theo cơ chế đường ống, nhờ đó với 5 lệnh sẽ cùng được xử lý đồng thời trên 1 đường ống và có thể coi mỗi lệnh chỉ được xử lý trong 1 chu kỳ nhịp đồng hồ.
 - Hơn nữa Pentium có 2 đường ống U và V được chế tạo hoạt động song song với nhau nên về mặt lý thuyết nó sẽ có khả năng thi hành 1 lúc 2 lệnh.

02/03/2016

138

VXL Pentium

- *Cấu trúc superscalar với 2 đường ống song song:*
 - Tuy nhiên, trên thực tế, không phải lúc nào cũng thực hiện được điều đó vì 2 lệnh kế tiếp nhau thường phụ thuộc vào nhau và lúc này chỉ có thể xử lý chúng 1 cách tuân tự.
 - Một mạch đặc biệt sẽ kiểm tra tính phụ thuộc dữ liệu đó xem chúng có yêu cầu phải liên kết trong quá trình xử lý hay không.

02/03/2016

139

VXL Pentium

- *Cấu trúc superscalar với 2 đường ống song song:*
 - Nếu không phụ thuộc, 2 lệnh sẽ được gởi đến 2 đường ống theo đúng thứ tự. Quá trình có thể dẫn giải như sau:
 - Trong công đoạn 1, lệnh hiện tại sẽ được đưa vào đơn vị giải mã của đường ống U, lệnh kế tiếp vào đơn vị giải mã của đường ống V.
 - Nếu trong công đoạn 2, mạch đặc biệt nói trên xác định có thể thực hiện đồng thời được 2 lệnh thì mỗi lệnh sẽ trải qua các công đoạn còn lại trong mỗi đường ống của nó, ngược lại thì lệnh trong ống U sẽ trải qua công đoạn kế tiếp như bình thường, còn lệnh kế tiếp trong ống V sẽ được chuyển qua thi hành trong ống U như là lệnh theo sau lệnh hiện tại.

02/03/2016

140

Một số câu hỏi

1. Vi lệnh là gì?
2. Trình bày sự khác nhau cơ bản giữa đơn vị điều khiển vi chương trình hoá và đơn vị điều khiển logic ngẫu nhiên
3. Kiến trúc RISC có những đặc điểm gì để phân biệt nó với kiến trúc CISC.

02/03/2016

141

Một số câu hỏi

5. Hãy tìm xem những đặc điểm của kiến trúc RISC nào được thể hiện trong VXL Pentium.
6. Tại sao người ta thường nói: VXL RISC là VXL có kiến trúc “LOAD-STORE”?

02/03/2016

143

Một số câu hỏi

4. Xử lý đường ống là gì? Vấn đề này sinh khi thực hiện cơ chế này? Nếu một vài cách khắc phục những vấn đề đó.

Gợi ý: Hiện tượng “bọt nước” xuất hiện trên đường ống do các lệnh rẽ nhánh sinh ra và tính phụ thuộc của các dữ liệu (RISC).

02/03/2016

142