



ĐỀ ÔN TẬP KIẾN TRÚC MÁY TÍNH 1

Kiến trúc máy tính (Trường Đại học Cần Thơ)



Scan to open on Studocu

ĐỀ ÔN TẬP KIẾN TRÚC MÁY TÍNH – THẦY TRÍ

CÂU 1: Hãy mô tả các thành phần cơ bản của kiến trúc máy tính theo mô hình John von Neumann. Giải thích ngắn gọn về vận hành của máy tính.

*Kiến trúc của máy Von NeuMann gồm 4 thành phần cơ bản của máy tính ngày nay:

- Bộ nhớ chính lưu chương trình và dữ liệu
- Hoạt động của khối ALU trên dữ liệu nhị phân
- CU thông dịch các lệnh từ bộ nhớ và thực thi
- Các thiết bị nhập xuất I/O khác

*Cách vận hành: chương trình được lưu trong bộ nhớ, bộ điều khiển sẽ lấy lệnh và biến đổi giá trị dữ liệu trong phần bộ nhớ, bộ làm toán và logic (ALU) được điều khiển để tính toán dữ liệu nhị phân, điều khiển hoạt động của các thiết bị vào ra.

CÂU 2: Cho một màn hình sử dụng 32 bit để hiển thị một màu cơ bản (đỏ - Red, xanh lá - Green, xanh dương - Blue) trong mỗi pixel có độ phân giải 1920x1080 pixel. Hãy cho biết ung lượng tối thiểu cần có của bộ đệm màn hình để có thể chứa một khung ảnh?

Chú ý: ghi rõ đơn vị gì (bit/byte...)

*Tổng số Pixel của màn hình là $1920 \times 1080 = 2.073.600$ pixel

Mà cần 32 bit để biểu diễn 1 màu cơ bản = 4 byte

➔ Bộ nhớ cần để lưu là: $2.073.600 \times 4 = 8.294.400$ byte

Vậy bộ nhớ cần tối thiểu 8.2MB

CÂU 3: Cho 3 bộ xử lý P1, P2, P3 cùng thực thi một tập lệnh với các tần số xung clock và CPI được cho như bảng bên dưới:

Bộ xử lý	Clock Rate	CPI
P1	2 Ghz	1.5
P2	1.5 Ghz	1.0
P3	3 Ghz	2.5

- Bộ xử lý nào có hiệu suất cao nhất dựa trên tiêu chí số lệnh thực thi trong 1 giây (IPS) và số triệu lệnh thực thi trong một giây (MIPS)
- Nếu các bộ xử lý thực thi 1 chương trình nào đó hết 10 giây, tính tổng số chu kỳ và tổng số lượng lệnh tương ứng
- Nếu chúng ta mong muốn giảm 30% thời gian thực thi sẽ dẫn tới việc tăng 20% CPI thì tần số xung clock mới của từng bộ xử lý tương ứng phải là bao nhiêu?

Chú ý: Phải tính MIPS và IPS xong mới kết luận! Ghi rõ công thức ra...

Đáp án:

a/ IPS: **CPU Execution Time P1= (IC x CPI₁)/Clock Rate=IC x 7,5x10⁻⁹**
 CPU Execution Time P2= (IC x CPI₂)/Clock Rate=IC x 6.6667x10⁻¹⁰
 CPU Execution Time P3= (IC x CPI₃)/Clock Rate=IC x 8,3333x10⁻¹⁰

MIPS: **MIPS_{P1}=Clock rate/CPIx10⁶ = 1333,3333**
 MIPS_{P2}=Clock rate/CPIx10⁶ = 1500
 MIPS_{P3}=Clock rate/CPIx10⁶ = 1200

Vậy theo kiến trúc IPS và MIPS thì P2 có hiệu suất cao nhất

b/ Time P1= IC x CPI/ Clock rate → IC P1 = 1,333 x 10¹⁰
→ CPU cycles= 2 x 10¹⁰

Time P2=IC x CPI/ Clock rate → IC P2 = 1,5 x 10¹⁰
→CPU cycles = 1,5x 10¹⁰

Time P3=IC x CPI/Clock rate → IC P3 = 1,2 x 10¹⁰
→CPU cycles = 3 x 10¹⁰

c/

CÂU 4: Xét 2 cách thiết kế khác nhau của cùng kiến trúc bộ lệnh cho hai bộ xử lý P1 và P2. Có 4 lớp lệnh: A, B, C, D. Tần số xung và CPI của mỗi cách thiết kế được cho như bảng bên dưới:

Bộ xử lý	Clock Rate	CPI
P1	2 Ghz	1.5
P2	1.5 Ghz	1.0
P3	3 Ghz	2.5

- a) Cho một chương trình với 10^6 lệnh được chia thành các lớp sau: 10% lớp A, 20% lớp B, 50% lớp C và 20% lớp D. Cách thiết kế cho bộ xử lý nào thực thi chương trình này nhanh hơn
- b) Tính CPI trung bình của mỗi bộ xử lý với chương trình trên
- c) Tính tổng số chu kì xung clock của chương trình trên P1 và P2

Chú ý: Ghi rõ công thức, đơn vị...

CÂU 5: Hãy tìm mã hợp ngữ MIPS tương đương với chương trình C như sau:

a) Cho các biến x,y,a,b được lưu ở \$s0,\$s1,\$s2,\$s3; \$s4 lưu cho z

$x = 10$ // addi \$s0,\$zero,10;

$y = x - 8$ //addi \$t0,\$0,8; sub \$s1,\$s0,\$t0;

$a = x * 2$ // sll \$s2,\$s0,1

$b = y / 4$ // srl \$s3,\$s1,2

$z = (x - a) - (y + b)$ // sub \$t1,\$s0,\$s2; add \$t2,\$s1,\$s3; sub \$s4,\$t1,\$t2;

Biết rằng các biến x, y, a, b là các số nguyên 32 bit

b) $x = y - a[10]$

lw \$t3,40(\$s2);

sub \$s0,\$s1,\$t3;

CÂU 6: Hãy phân biệt sự khác nhau giữa Big End (Big Endian) và Little End (Little Endian). MIPS sử dụng Big End hay Little End?

Big End: lưu từ byte cao đến thấp ở địa chỉ từ thấp đến cao

Little End: Lưu từ byte thấp đến cao ở địa chỉ từ thấp đến cao

MIPS sử dụng Big End

CÂU 7: Cho 2 số x, y viết theo như sau

x	y
4	-6
12	-9
19	-12
60	88

Hãy biểu diễn x, y theo nhị phân 8 bit có dấu bù 2 và thực hiện phép tính $(x+y)$ theo hệ nhị phân

Chú ý: Tràn số! Đề ghi rõ 8 bit

CÂU 8: Chuyển đổi số 20 ở dạng thập phân sang định dạng IEEE 754 chính xác đơn và kép.

Chú ý: Phải ghi rõ từng bước làm: đổi nhị phân, chuẩn hóa, tính mũ...

CÂU 9: Cho $x = 0100\ 0110\ 1101\ 1000\ 0000\ 0000\ 0000\ 0000$

$y = 1011\ 1110\ 1110\ 0000\ 0000\ 0000\ 0000\ 0000$

Ở dạng dấu chấm động chính xác đơn IEEE 754. Hãy tính:

- a) $x + y$
- b) $x * y$

Chú ý: Phải tính từng bước theo công thức trong slide Thầy đã giảng (chuẩn hóa mũ, phép tính, chuẩn hóa, kiểm tra tràn nếu có...)

CÂU 10: Hãy biểu diễn nhị phân các lệnh sau:

- a) `Addu $t2, $zero, $t3`
- b) `slt $t0, $t3, $zero`
- c) `beq $t0, $zero, L1` //PC: 0x00400008
- d) `j L2` //PC: 0x0040000c

L1 có địa chỉ 0x00400000

L1 có địa chỉ 0x00400014

Chú ý: Các bạn phải tra cứu bảng MIPS Reference! Lệnh `addu` có opcode gì... `beq` và `jump` thì phải xem lại các ví dụ Thầy đã giảng trên lớp

CÂU 11: Mô tả các bức chuyển 1 chương trình C thành mã máy thực thi trên máy tính

CÂU 12: Hãy viết chuyển đổi tổng quát từ ngôn ngữ C => assembly MIPS

Chú ý: Phải ghi giả thuyết $<$, $<=$, $>$, $>=$...

- a) `if (condi) { stats 1 } else { stats 2 }`
- b) `while (condi) {stats; }`

CÂU 13: Cho đoạn mã sau:

`If (x < 0) h = 0 else h = x`

Với x, h được lưu trong \$s0, \$s1

Hãy dịch sang assembly MIPS đoạn mã trên

CÂU 14: Dùng 4 bit nhị phân để biểu diễn số nguyên hệ thập phân. Thực hiện các phép tính:

- a) $7 + 8$
- b) $7 - 8$
- c) $7 * 8$
- d) $13 : 4$

Chú ý: *Phải trình bày từng bước của giải thuật*

CÂU 15: Thực hiện các phép tính trên số thực chấm động:

- a) $2.125 + 1.25$
- b) 1.25×2.5
- c) $2.5 : 1.25$

Chú ý: *Phải đổi về số thực dấu chấm động rồi thực hiện phép tính!*

CÂU 16: Hãy giải thích lệnh quy trình thực thi (datapath) lệnh có mã nhị phân là:

101011 01000 01001 00000 00000 010000

CÂU 17: Pipeline

- a) Ý tưởng tăng tốc thực thi chương trình với kỹ thuật Pipeline
- b) Trường hợp lý tưởng thì Pipeline có thể tăng tốc được bao nhiêu so với không sử dụng Pipeline
- c) Các vấn đề gặp phải khi sử dụng kỹ thuật Pipeline

CÂU 18: Trình bày các đường tín hiệu và các khối chức năng được sử dụng khi thực thi lệnh **addi**? (*Tra cứu bảng MIPS REFERENCE DATA*)

CÂU 19: Trình bày các đường tín hiệu và các khối chức năng được sử dụng khi thực thi lệnh **beq**? (*Tra cứu bảng MIPS REFERENCE DATA*)

CÂU 20: Trong trường hợp Pipeline 5 tầng sử dụng kỹ thuật forwarding, sử dụng lệnh ‘nop’ để giải quyết xung đột xảy ra (nếu có) trong chuỗi lệnh bên dưới, sau đó tính thời gian thực thi chuỗi lệnh, biết rằng chu kỳ CPU khi sử dụng forwarding là 400ps.

lw \$1, 40(\$6)

add \$6, \$2, \$2

sw \$6, 32(\$1)