

حل تست های معماری کامپیوتر

کارشناسی ارشد ۱۴۰۳

استاد: مسعود کبیری

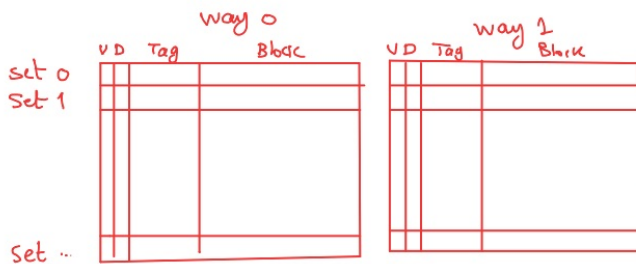
۸۳- یک سیستم حافظه اصلی با ظرفیت ۲G بایت و حافظه نهان با ظرفیت ۲۵۶ KW (w: یک کلمه معادل ۴ بایت) است. حافظه نهان از روش ۲-way set associative استفاده می‌کند و اندازه هر بلاک در آن، ۱۶ بایت است. تعداد بیت‌های index و TAG به ترتیب کدام‌اند؟

۱۲ و ۱۵ (۲✓)

۱۱ و ۱۶ (۱)

۱۲ و ۱۶ (۴)

۱۱ و ۱۵ (۳)



۱ Byle : واحد آدرس‌نویسی

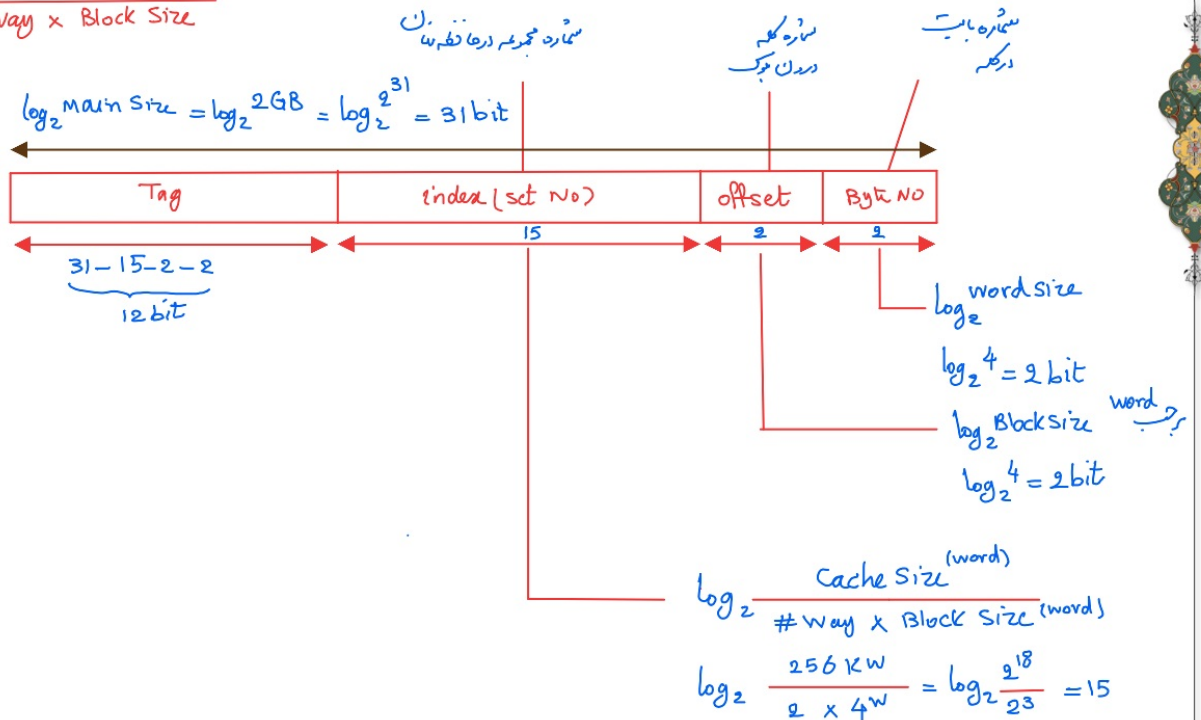
4 Byle : word کلمه

$4w = 16 \text{ Byle}$: Block

256 KW : Cache Size

2 G Byle : Main Size

$$\# \text{Set} = \frac{\text{Cache Size}}{\# \text{Way} \times \text{Block Size}}$$



میان

۸۴- یک سیستم نمایش اعداد ممیز شناور را در نظر بگیرید که بخش‌های مانتیس ۴ بیت، توان ۳ بیت و علامت یک بیت باشند (مطابق جدول زیر). نمایش توان $biased-4$ و نمایش مانتیس $implicit\ one\ representation$ فرض شده است. نمایش عدد دهدهی 0.75 در این صورت کدام است؟ (گزینه‌ها در مبنای ۱۶ هستند). ^{۱ صفتی}

علامت	توان	مانتیس
۱ bit	۳ bit	۴ bit

$$Bias = 4$$

$$E_{biased} = E_{ واقعی } + Bias$$

$$M = 1.F$$

نمایره کردن

گام ۱: عدد مبنای ۲ برده می‌شود.

$$0.75 = 0.5 + 0.25 = \frac{1}{2} + \frac{1}{4} = (0.11)_2$$

$$0.75 \times 2 = 1.5$$

$$0.5 \times 2 = 1.0$$

گام ۲: نمایره کردن عدد

$$(0.11)_2 = 1.1 \times 2^{-1}$$

$$\begin{array}{r} M \\ 1.1000 \\ \hline F \end{array}$$

$$E = -1$$

$$E_{biased} = E_{ واقعی } + Bias = -1 + 4 = 3 = 011$$

3			8			
0	0	1	1	0	0	0
S			F			
			E_{biased}			

ISA

۸۵- دو پیاده‌سازی مختلف از یک معماری مجموعه دستورات را در نظر بگیرید. این دستورات براساس میزان متوسط تعداد پالس‌های ساعت برای اجرای هر دستور (CPI)، به چهار کلاس مختلف با نام‌های A، B، C و D تقسیم شده‌اند. مقادیر CPI در هر کلاس و درصد استفاده از کلاس‌های دستورات در دو پیاده‌سازی مختلف P1 و P2 مطابق با جدول زیر است. در پیاده‌سازی اول (P1)، نرخ پالس ساعت ۲/۴ GHz و در پیاده‌سازی دوم (P2) ۳ GHz است. در اجرای یک برنامه با 10^6 دستور، کدام پیاده‌سازی و چند برابر سریع‌تر است؟

	درصد استفاده در برنامه				مقدار CPI در هر کلاس			
	A	B	C	D	A	B	C	D
P1	۳۰	۲۰	۴۰	۱۰	۱	۳	۲	۱
P2	۲۰	۵۰	۲۰	۱۰	۱	۱	۳	۲

۱) P2 و ۱/۵ ✓

۲) P2 و ۲

۳) P1 و ۱/۵

۴) P1 و ۲

$$f_1 = 2.4 \text{ GHz}$$

$$f_2 = 3 \text{ GHz}$$

$$T_{EX1} = P1 \text{ به } T_{CK1} \times \text{مقدار کل کلاس} \rightarrow \frac{1}{2.4 \times 10^9}$$

$$T_{CK} = \frac{1}{f_{CK}}$$

$$T_{EX2} = P2 \text{ به } T_{CK2} \times \text{مقدار کل کلاس} \rightarrow \frac{1}{3 \times 10^9}$$

$$P1 \text{ به } T_{EX1} = \underbrace{\frac{30}{100} \times N \times 1}_{\text{کلاس A}} + \underbrace{\frac{20}{100} \times N \times 3}_{\text{کلاس B}} + \underbrace{\frac{40}{100} \times N \times 2}_{\text{کلاس C}} + \underbrace{\frac{10}{100} \times N \times 1}_{\text{کلاس D}} = 1.8 N$$

$$P2 \text{ به } T_{EX2} = \underbrace{\frac{20}{100} \times N \times 1}_{\text{کلاس A}} + \underbrace{\frac{50}{100} \times N \times 1}_{\text{کلاس B}} + \underbrace{\frac{20}{100} \times N \times 3}_{\text{کلاس C}} + \underbrace{\frac{10}{100} \times N \times 2}_{\text{کلاس D}} = 1.5 N$$

$$T_{EX1} = \frac{1.8 N}{2.4 \times 10^9}$$

$$T_{EX2} = \frac{1.5 N}{3 \times 10^9}$$

$$T_{EX2} < T_{EX1}$$

$$S = \frac{T_{EX1}}{T_{EX2}} = \frac{\frac{1.8}{2.4 \times 10^9}}{\frac{1.5}{3 \times 10^9}} = \frac{1.8}{1.2} = \frac{3}{2} = 1.5$$

۸۶- برای افزایش کارایی یک پردازنده، در بخش جمع کننده از واحد محاسبات و منطق (ALU) آن، به اندازه ۲ برابر تسريع ایجاد شده است. در صورتی که احتمال استفاده از جمع کننده در اجرای برنامه‌ها برابر ۲۰ درصد و احتمال استفاده از واحد محاسبات و منطق برابر ۷۰ درصد باشد، میزان تسريع پردازنده در اجرای برنامه‌ها، کدام است؟

۱/۸ (۲)

۱/۰۷ (۱)

۱/۵۳ (۴)

۱/۱۲ (۳) ✓

Amdahl's Law

نظریه آمل

$$f_{Add} = 20\% \quad K_{Add} = 2$$

$$f_{ALU} = 70\% \quad K_{ALU} = 1$$

$$f_{Others} = 10\% \quad K_{Others} = 1$$

$$S = \frac{1}{\frac{f_{Add}}{K_{Add}} + \frac{f_{ALU}}{K_{ALU}} + \frac{f_{Others}}{K_{Others}}} = \frac{1}{\frac{0.2}{2} + \frac{0.7}{1} + \frac{0.1}{1}} = \frac{1}{0.9} = 1.12$$

۸۷- کدامیک از موارد زیر، مشخص کننده Subroutine مربوط به عملیات INDRCT (دسترسی به آدرس غیرمستقیم) در یک واحد کنترل مبتنی بر ریزبرنامه است؟

NOP	I	CALL	INDRCT	NOP	S	JMP	OVER
ACTDR	U	JMP	NEXT	NOP	U	JMP	FETCH
WRITE	U	JMP	FETCH				

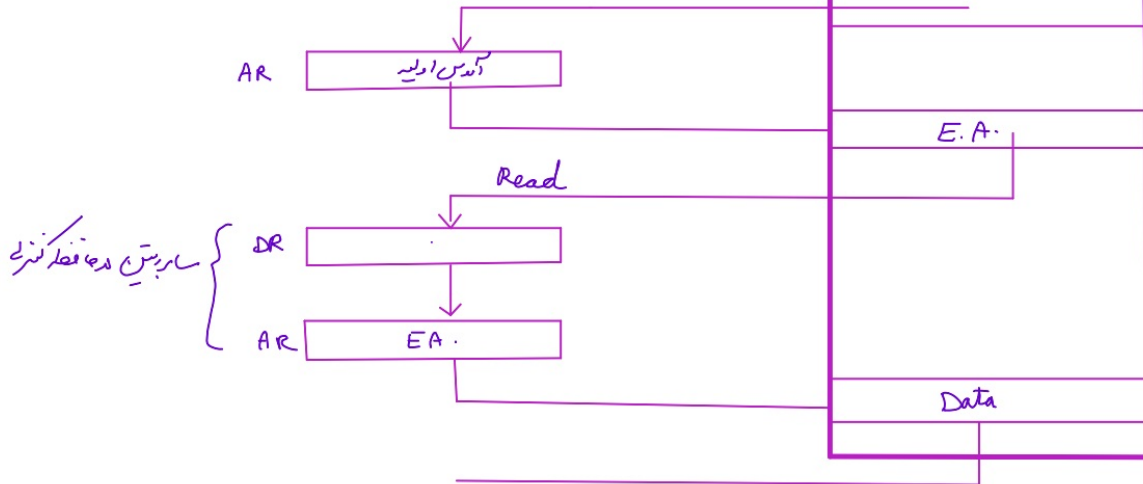
^{EW}
 READ U JMP NEXT
 DRTAR U RET
 A ← DR

Unconditional

PCTAR U JMP NEXT
 READ, INCPC U JMP NEXT
 DRTAR

در کدس و در زیر مستقیم

Main Mem.



cw

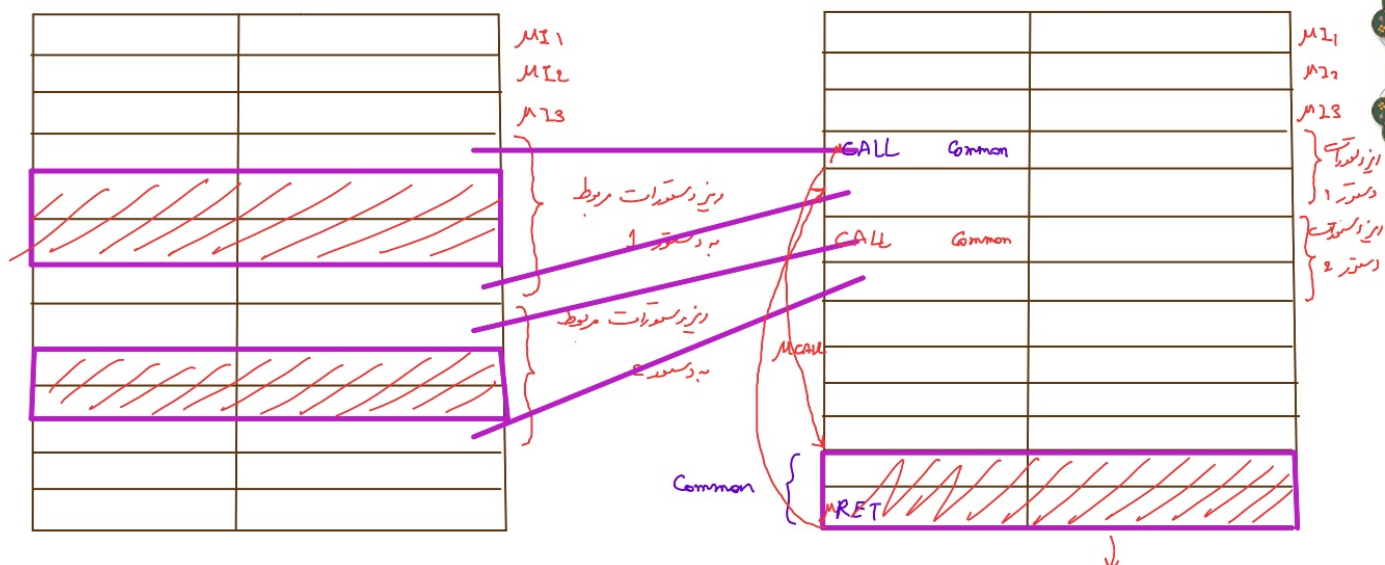
NEXT	۱. کد نرسل بعدی
MP	۲. محاسبه
MAP	۳. نگاشت
CALL	۴. فراخوانی
RET	۵. بازگشت

در محل کربن

MCALL

MRET

opcode $\xrightarrow{\text{Map}}$ آپس کے کنٹرول سے طریقہ دستور
opcode



Fully Associative

۸۸- فرض کنید یک سامانه کامپیوتری از یک سطح حافظه نهان داده تمام انجمنی با گنجایش ۴ بلوک و یک حافظه اصلی با گنجایش ۲۵۶ بلوک در کنار پردازنده تشکیل شده است. چه تعداد کلاک برای اجرای دنباله دسترسی های بلوکی زیر، مورد نیاز است؟ (دسترسی از چپ به راست اتفاق می افتد)

$\overline{A}, B, C, D, E, A$

توضیح: از زمان اجرای تمامی دستورات غیر از دستورات دسترسی به حافظه صرف نظر شده است. همچنین زمان جستجو در حافظه نهان یک کلاک، زمان انتقال داده از حافظه نهان به پردازنده یک کلاک و زمان انتقال داده از حافظه اصلی به حافظه نهان، ۱۰ کلاک است. سیاست جایگزینی استفاده شده در حافظه نهان، سیاست MRU در نظر گرفته شده است. توجه نمایید که ارتباط پردازنده با سلسله مراتب حافظه تنها از طریق حافظه نهان صورت می گیرد.

۶۲ (۲ ✓)

۳۳ (۱)

۸۴ (۴)

۷۵ (۳)

MRU : Most Recently Used

-miss

A

1 + 10 + 1
خوانش Cache
انتقال از Main Cache
جستجو Cache

-miss

B
A

1 + 10 + 1
خوانش Cache
انتقال از Main Cache
جستجو Cache

-miss

C
B
A

1 + 10 + 1
خوانش Cache
انتقال از Main Cache
جستجو Cache

-miss

D
C
B
A

1 + 10 + 1
خوانش Cache
انتقال از Main Cache
جستجو Cache

-miss

E
C
B
A

1 + 10 + 1
خوانش Cache
انتقال از Main Cache
جستجو Cache

hit

E
C
B
A

1 + 1
خوانش Cache
جستجو Cache

$$5 \times 12 + 2 = 62 \text{ clock}$$

۸۹- چه تعداد عملیات شیفت به راست در الگوریتم ضرب Booth برای انجام عملیات ضرب $(-9) \times (-13)$ ، مورد

نیاز است؟

۷ (۱)

۵ (۳) ✓

۶ (۲)

۴ (۴)

Multiplicand
ضرب‌کننده

Multiplier

Booth's Algorithm برای اعداد علامت دار
Add & shift برای اعداد بدون علامت

Multiplier تعداد شیفت = تعداد بیت‌های ۱ در Multiplier

$$\begin{array}{r} -13 \rightarrow \\ +13 \quad \underline{01101} \xrightarrow{2's \text{ Comp}} \underline{10011} \end{array}$$

$$\begin{array}{c} - \quad + \quad - \\ \boxed{10011} \cdot 0 \end{array}$$