# Cache

## سوال اول (لکچر ۲ و۳)

در یک حافظه نهان با سیاست جایدهی نگاشت مستقیم و با چهار بلوک، محتوای آدرس های زیر به ترتیب از چپ به راست درخواست شده است. وضعیت موفقیت هریک از درخواست ها و نرخ موفقیت را مشخص کنید. ([برای تبدیل اعداد هگز به دسیمال](https://www.rapidtables.com/convert/number/how-hex-to-decimal.html))

6A, 6B, 6C, 2C, 2D, 6A, 6B, 6C, 2C, 2D, 11, D, 41, 6B, 11, 3D, 13, 2B, 3D, 2D

پاسخ

| 2D | 2C | 6C | 6B | 6A | 2D | 2C | 6C | 6B | 6A | Hex |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 45 | 44 | 108 | 107 | 106 | 45 | 44 | 108 | 107 | 106 | Dec |
| 1 | 0 | 0 | 3 | 2 | 1 | 0 | 0 | 3 | 2 | Block |
| h | m | m | h | h | m | m | m | m | m | h/m |

| 2D | 3D | 2B | 13 | 3D | 11 | 6B | 41 | D | 11 | Hex |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 45 | 61 | 43 | 19 | 61 | 17 | 107 | 65 | 13 | 17 | Dec |
| 1 | 1 | 3 | 3 | 1 | 1 | 3 | 1 | 1 | 1 | Block |
| m | h | m | m | m | m | h | m | m | m | h/m |

Hits : 5, Total : 20

Hit Rate = 0.25

## سوال دوم (لکچر ۲ و۳)

فرمول محاسبه CPU time از سه پارامتر تشکیل شده است این فرمول را بنویسید موارد موثر بر هر پارامتر را نام ببرید.

سوال از متن اسلاید و کلاس، جواب:

CPU time = IC \* CPI \* T = IC \* CPI / F

* Instruction Count
  + Program (algorithm, language)
  + ISA
  + Compiler
* CPI
  + uArch: the way processor is implemented
  + Code CPI also depends on the program
* Clock Cycle Time (or CPU Frequency)
  + Technology and uArch

## سوال سوم (لکچر ۲ و۳)

باتوجه به جدول زیر به سوالات پاسخ دهید. (زمان اجرا را برابر CPU time درنظر بگیرید )

| Processor C | Processor B | Processor A |  |
| --- | --- | --- | --- |
| 2.5 | 3 | 4 | Frequency (Ghz) |
| 1.5 | 2 | 1.5 | CPI |

الف) پردازنده ها را براساس معیار MIPS مرتب کنید.

ب) برنامه p توسط هر یک از این پردازنده ها در مدت زمان 30 ثانیه اجرا میشود. آیا میتوانید تعداد کلاک و تعداد instructionهای اجرا شده در برای هر پردازنده را محاسبه کنید؟ (دقت کنید که هر P برای هر پردازنده یه طور متفاوت کامپایل شده و در هر پردازنده میتواند از منظر تعداد دستورات متفاوت باشد)

ج) قصد داریم که زمان اجرای برنامه را برای این پردازنده‌ َA را بدون اینکه CPI را برای آن تغییر دهیم به اندازه 20% کاهش دهیم. فرکانس کاری این پردازنده با به چه عددی برسد؟ ( برای IC میتوانید از مقدار بدست آمده در قسمت ب استفاده کنید)

نمونه سوال دکتر فربه همراه با تغییرات

اگر انتخاب شد جوابشو مینویسم ولی کلا فرمول اصلی حل سوال اینه:

Cpu time = (IC \* CPI)/f

#of clock = IC \* CPI

## 

## سوال چهارم (لکچر ۵)

الف - سیاست نگاشت مستقیم (direct mapped) را توضیح دهید.

ب - نقطه ضعف اصلی این سیاست جایابی چیست ؟

ج - پیشنهاد شما برای حل این نقطه ضعف چیست ؟ (امتیازی)

پاسخ  
الف - در این سیاست یک نگاشت یکتا از main memory به حافظه نهان (cache) وجود دارد. یعنی اطلاعات داخل هر خانه حافظه اصلی، پس از انتقال به حافظه نهان، تنها در یک خانه مشخص از حافظه نهان می تواند قرار گیرد.

ب - در این سیاست با توجه به اینکه هر خانه حافظه اصلی دقیقا به یکی از خانه های حافظه نهان نگاشت می شود، در صورتی که در آن خانه حافظه نهان از پیش مقدار دیگری قرار داشته باشد، حتی با وجود خالی بودن خانه های دیگر، داده جدید جایگزین داده قبلی در خانه مورد نظر می شود.

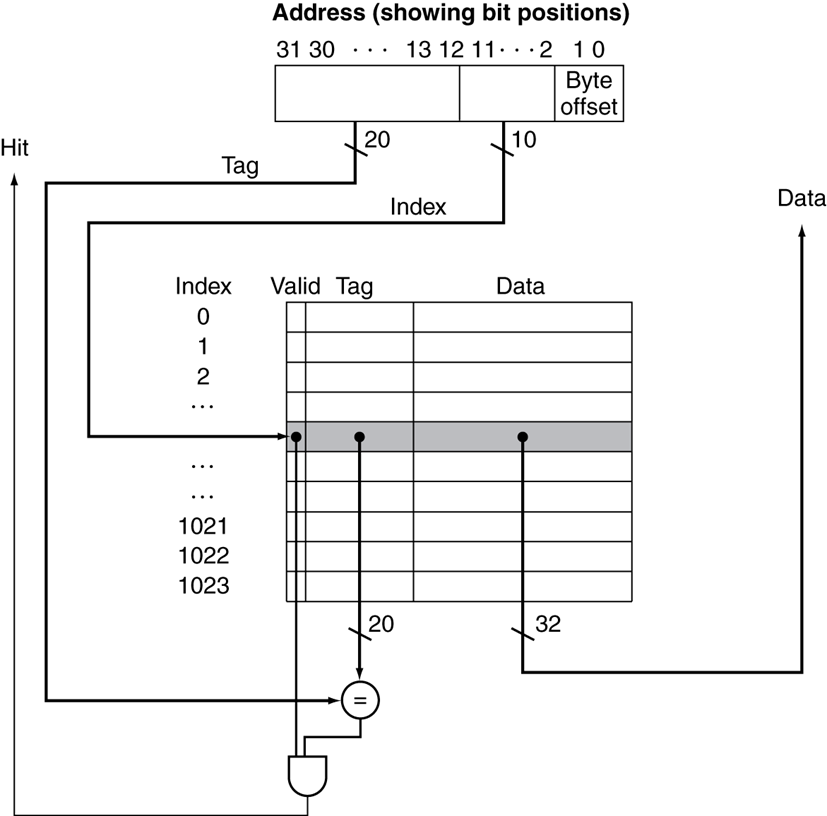
ج - به جای اینکه برای هر آدرس فقط یک جای مشخص در حافظه نهان موجود باشد، چند جا داشته باشد و هر وقت تمام آن جاها پر شد با یک سیاست خاص تصمیم گرفته شود که کدام یک از آنها جای خود را به بلوک جدید دهد. در واقع به این روش associative گفته می شود و از سیاست های جایگزینی گوناگونی همچون LRU، FIFO و غیره استفاده می شود که جلوتر با آنها آشنا خواهید شد.

## سوال پنجم (لکچر ۵)

فرایند بازیابی یک بلوک داده از طریق کش در سیاست نگاشت مستقیم را توضیح دهید.

پاسخ

مطابق شکل زیر، ابتدا از بیت های آدرس دهی، بیت های index که ردیف های حافظه نهان را آدرس دهی می کنند، خانه مورد نظر در cache را تعیین می کنند. قسمت tag آدرس با tag ردیف تعیین شده در حافظه نهان مقایسه می شود و نتیجه با بیت valid ردیف حافظه and می شود و در صورتی که نتیجه برابر با یک باشد hit رخ می دهد و داده از حافظه نهان خوانده می شود و در غیر این صورت miss رخ داده و داده از حافظه اصلی گرفته می شود و در حافظه نهان نیز نوشته می شود.



## سوال ششم (لکچر ۵)

\*\*(سوال در تمرین بوده ولی برای میانترم با عددای جدید )  
گنجایش یک حافظه اصلی 512 کیلوبایت است و اندازه هر بلوک آن دو word می باشد. در صورتی که گنجایش حافظه نهان 64 بایت بوده و سیاست جایدهی به صورت نگاشت مستقیم باشد قالب آدرس دهی خانه های حافظه را مشخص کنید. (هر word یک بایت است)

پاسخ  
ابتدا تعداد بایت های کل حافظه را حساب می کنیم تا تعداد کل بیت های آدرس دهی مشخص شوند.

در نتیجه تعداد کل بیت های آدرس دهی 19 می باشد. با توجه به اینکه اندازه هر بلوک 16 بیت (2 بایت یا دو کلمه در این سوال) است یک بیت () برای بخش offset نیاز داریم. همین طور با توجه به اینکه هر بلوک 2 بایت است و گنجایش حافظه نهان نیز 64 بایت است، در حافظه نهان 64/2 = 32 بلوک جا می گیرد در نتیجه برای آدرس دهی حافظه نهان نیز به 5 بیت () نیاز داریم (قسمت index). در نهایت نیز 13 بیت می آدرس دهی می ماند که مربوط می شود به بخش tag.

| Tag (13bits) | Index (5bits) | Offset (1bit) |
| --- | --- | --- |

## سوال هفتم (لکچر ۶)

برای کاهش Miss Penalty بهتر است اندازۀ بلوکهای حافظۀ نهان کوچکتر باشند یا بزرگتر؟ پاسخ خود را توضیح داده و دو روش موجود برای کاهش اثر این پدیده را مختصر بیان کنید.

پاسخ

ابتدا بررسی میکنیم که miss penalty چیست تا به این سوال پاسخ دهیم. زمانی که میخواهیم دادهای را از حافظۀ اصلی بخوانیم و به حافظۀ نهان درخواست میدهیم، اگر داده در حافظۀ نهان یافت نشد علاوه بر هزینۀ جستجو در آن باید هزینۀ خواندن داده از حافظۀ اصلی را نیز بدهیم. داده ها word به word از حافظۀ اصلی به حافظۀ نهان منتقل می شوند، بنابراین زمانی که این انتقال داده طول میکشد رابطۀ مستقیم با تعداد word هایی که میخواهیم منتقل کنیم دارد. همانطور که میدانیم در هر انتقال از حافظۀ اصلی به حافظۀ نهان یک بلوک از کلمات را جابجا میکنیم، بنابراین می توان گفت میزان miss penalty رابطۀ مستقیم با اندازۀ بلوکهای حافظۀ نهان دارد. لذا برای کاهش آن بهتر است اندازۀ بلوکها "کوچکتر" باشد.

## سوال هشتم (لکچر ۶)

\*\*(سوال در کوییز بوده ولی برای میانترم با عددای جدید )

فرض کنید در یک پردازنده از سیاست write-back استفاده میکنیم و مقدار base CPI برابر با 3 میباشد. به علاوه در حدود 15% از دستورات پردازنده مربوط به نوشتن در حافظه بوده و هر بار نوشتن در حافظه به طور متوسط 120 کلاک طول میکشد. وجود دستورات نوشتن بر حافظۀ اصلی پردازنده را چند برابر کندتر کرده است؟ برای کاهش این مشکل چه ترفندی را میتوان به کار برد؟

پاسخ

برای مقایسه عملکرد سیستم در دو حالت، CPI واقعی را با base CPI مقایسه میکنیم.

Effective CPI = 3 + 0.15 × 120 = 3 + 18 = 21

بنابراین:

Effective CPI / Base CPI = 21 / 3 = 7

در نتیجه میتوان گفت پردازنده 7 برابر کندتر شده است.

برای کاهش این مشکل باید تا جای ممکن ارجاعات مستقیم پردازنده به حافظۀ اصلی را کاهش دهیم. برای این کار از یک write buffer که یک بافر سریع میان پردازنده و حافظۀ اصلی است استفاده میکنیم تا CPU مجبور نباشد تا نوشتن داده بر حافظه منتظر بماند. هر زمان که قرار بود دادهای بر حافظۀ اصلی نوشته شود ابتدا بر روی write buffer نوشته میشود و پردازنده میتواند به کارهای خود ادامه دهد تا write buffer در زمان مناسب داده را به حافظۀ اصلی منتقل کند. البته اگر تعداد داده هایی که میخواهند بر حافظه نوشته شوند زیاد باشد ممکن است این بافر پر شود و پردازنده مجبور شود منتظر بماند تا بافر خالی شود، پس همچنان احتمال معطلی پردازنده وجود دارد.

## سوال نهم (لکچر ۶)

پردازندهای با فرکانس 3GHz کار میکند. همچنین هر کلاک گذرگاه حافظه معادل 3 کلاک پردازنده بوده و پهنای این گذرگاه به اندازۀ یک کلمۀ 2 بایتی است. فرض کنید هنگام خواندن دادهای از حافظۀ نهان miss رخ داده و می خواهیم یک بلوک 8 کلمه ای از داده را از حافظۀ اصلی بخوانیم. در این روند 8 کلاک گذرگاه[[1]](#footnote-0) برای دسترسی به دادۀ درون حافظه، 1 کلاک برای انتقال آدرس داده از حافظۀ نهان به حافظۀ اصلی و 2 کلاک برای انتقال داده از حافظۀ اصلی به حافظۀ نهان زمان صرف می شود. در این شرایط نرخ ارسال داده میان حافظۀ نهان و حافظۀ اصلی چند بایت بر ثانیه است؟

پاسخ

داده های موجود در مساله به شرح زیر هستند:

* CPU frequency: 3GHz 🡪 3 × 109 clocks/second
* 1 CPU clock = 3 bus cycle
* 1 word = 2 bytes
* 1 block = 8 words
* 1 bus cycle needed for address transfer
* 8 bus cycles needed per DRAM access
* 2 bus cycle needed per data transfer

در نتیجه می توان نوشت:

Miss penalty = 1 + 8 × 4 + 2 × 4 = 41 bus cycles

بنابراین

Band width = (4 × 2 bytes) / 41 cycles ≈ 0.2 bytes/cycle

هر کلاک پردازنده معادل 3 کلاک گذرگاه میباشد، بنابراین گذرگاه در یک ثانیه 109 سیکل را طی خواهد کرد. پس در نهایت می توان نوشت:

Band width = 0.2 bytes/cycle × 109 cycles/second = 2 × 108 bytes/second

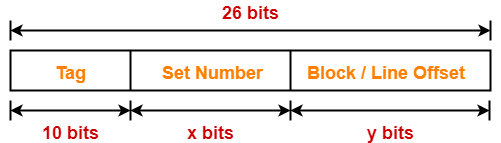
## سوال دهم (لکچر ۶)

[link](https://www.gatevidyalay.com/set-associative-mapping-practice-problems/)  
فرض کنید در یک سیستم حجم حافظه اصلی 64MByte می باشد و در آدرس دهی 10 بیت برای بخش tag درنظر گرفته شده. اگر سیاست جای دهی حافظه نهان آن 4way set associative باشد، اندازه این حافظه نهان چند بایت است ؟

پاسخ

برای محاسبه حجم حافظه نهان ابتدا باید ببینیم تعداد کل بیت های آدرس دهی چند تاست:

بعد از تعداد بیت های tag کم می کنیم. که میشه جمع بیت های block و set :



با توجه به اینکه نمیدونیم داخل هر بلوک چند word وجود داره نمیتونیم جدا جدا x و y رو حساب کنیم اما دونستن جمعشون برای ما کفایت می کنه. حجم حافظه نهان برابره با (تعداد set ها داخل حافظه نهان \* تعداد بلوک های هر set \* تعداد word هر بلوک). تعداد بلوک های هر set که با توجه به 4way set associative بودن حافظه میدونیم 4 هست، جمع x و y هم میدونیم برابره با 16. در نتیجه :

cache memory size = =

# 

# ALU & FP

## سوال اول (FP)

عدد 18.72 را به یک عدد ممیز شناور با 5 رقم exponent و3 رقم fraction تبدیل کنید، و سپس به عدد مبنای ده برگردانید. میزان خطای این تبدیل چقدر است؟

## سوال دوم (ALU)

\*\*جدید و احتمالا سخت

در یک ضرب­کننده ترتیبی به روش Add & Shift به فرض اینکه عمل جمع به 15 نانوثانیه و عمل شیفت به 5 نانوثانیه نیاز داشته باشد و با فرض اینکه عملیات شیفت و جمع در دو کلاک متوالی انجام می­شوند، انجام ضرب 1100 : مضروب و 1011 : مضروب فیه بر حسب نانوثانیه چقدر زمان نیار دارد؟

## سوال سوم (ALU)

\*\*جدید و احتمالا سخت

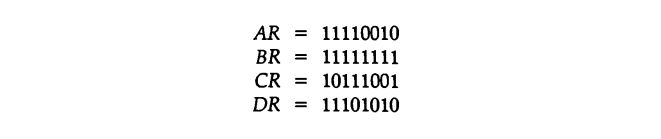
اگر یک عدد صحیح که 2n بیت دارد را بر عدد n بیتی X تقسیم کنیم به چه شرطی سرریز نداریم و خارج قسمت در n بیت می گنجد؟

# 

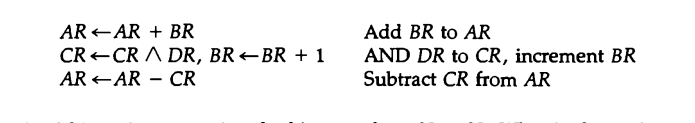
# Computer Design

## سوال اول (لکچر ۱۳)

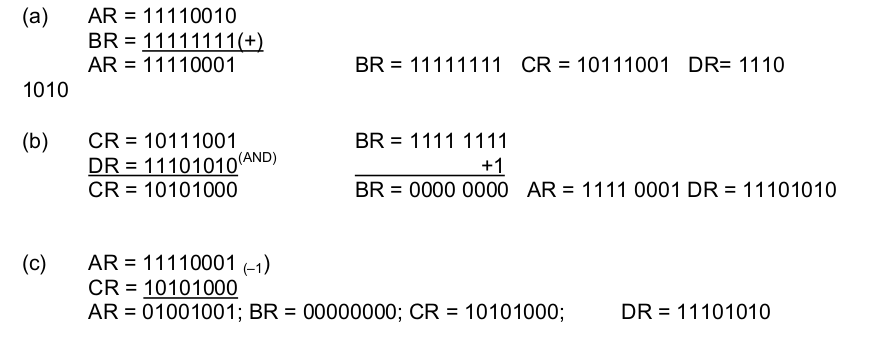
رجیسترهای ۸ بیتی AR, BR, CR و DR دارای مقادیر زیر میباشند:



مقادیر این رجیسترها را پس از انجام هریک از عملیات زیر بر روی آنها مشخص کنید:



پاسخ:



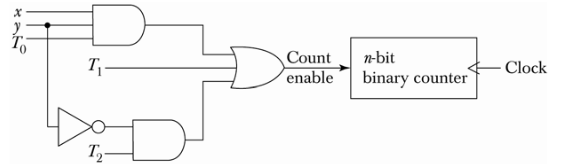
## 

## سوال دوم (لکچر ۱۳)

پیاده سازی سخت افزاری عبارت زیر را نشان دهید. دیاگرام بلوکی برای بخش شمارنده باینری و گیت های منطقی برای بخش تابع کنترلی را همراه با اتصالات میان آنها نمایش دهید.



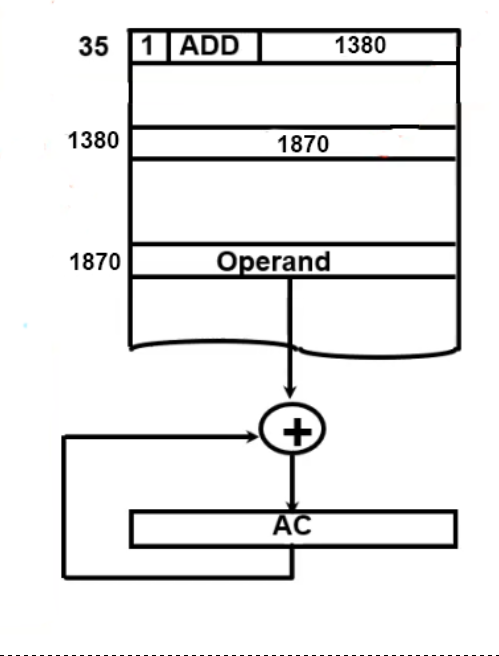
پاسخ:



## 

## سوال سوم (لکچر ۱۴)

آدرس موثر چیست؟‌ در شکل زیر آدرس موثر دستوری که در خانه ۳۵ حافظه وجود دارد چند است؟



پاسخ: آدرس موثر آدرسی است که می‌توان از ان بدون اصلاح و تغییر به عملوند یک دستور محاسباتی، یا مقصد یک دستور پرشی دست یافت. در شکل فوق آدرس موثر ۱۸۷۰ است زیرا طبق نوع آدرس‌دهی غیر مستقیم آدرسی است که در آن عملوند وجود دارد.

## 

## سوال چهارم (لکچر ۱۵)

یک ISA برای کامل بودن چند نوع دستور دارد؟

پاسخ:

چهار نوع: Functional Instructions, Transfer instructions, Control Instructions, IO Instructions

1. Bus cycle [↑](#footnote-ref-0)