

بسم الله الرحمن الرحيم

ساختار کامپیوتر

امیر حسین رستمی

96101635

گزارش کار فاز **چهارم** پروژه + فاز اضافی خودم

دکتر موحیدیان عطار

دانشگاه صنعتی شریف بهار 98

پاسخ خواسته اول:

تست بنچی که تحت عنوان hitMissTestBench ضمیمه کرده ام در صورت ران شدن خروجی ای می دهد که به ترتیب هیت یا میس شدن داده در کش را نشان می دهد و در نهایت داریم که 28 عدد هیت داریم و الباقی میس می باشند.

Number of hits = 28 out of 100 miss/hits so HitRate = $28/100 = 0.28$

MissRate = $1 - \text{HitRate} = 0.72$.

پاسخ خواسته دوم:

پیاده سازی ساختار کش:

طبق درس و داده های مساله داریم که از آنجا که اندازه ی بلوک سایز یک است و نوع پیاده سازی way - 2 است لذا تعداد ست ها برابر است با 8 و لذا سه بیت برای مشخص کردن ست ها لازم است و از آنجا که تعداد بیت های آدرس 6 بیت می باشد لذا داریم که سه بیت پایینی آن برابر setNumber می باشد . حواسمان باشد که با توجه به بررسی دیتای آدرس، طراح پروژه دو بیت Byte offset را در نظر نگرفته است و لذا من برای setNumber همان 3 بیت LSB را در نظر میگیرم و سه بیت MSB را جهت تگ در نظر میگیرم.

- اما برای پیاده سازی سیاست LRU در هر Read یا Write داریم که اگر Hit رخ بدهد بیت LRU متناظر با way ای قرار میگیرد که هیت به آن مربوط نبوده است (کمتر اخیرا استفاده شده است) و اگر میس رخ بدهد LRU مخالف مقدار قبلی اش میگردد. به بیان دیگر به way ای اشاره می کند که قبلا به آن اشاره نمی کرد.

```
//                               Way0                               way1
// My syntax = LRU // FIRST BLOCK : DIRTY_BIT VALID_BIT TAG MAIN_DATA // SECOND BLOCK : DIRTY_BIT VALID_BIT TAG MAIN_DATA;
```

توجه کنید که من main Data را کل 8 بیت تولیدی پروسسور را گرفته ام.

لذا تعداد بیت هر واحد کش به صورت زیر محاسبه می شود:

total number = 1 bit LRU + 2 bit dirtyBits for two blocks + 2 bit for valid bits + 3 + 3 bits for to tags
for two blocks + 8 + 8 two main Datas = 27 bits.

توجه کنید که با 1 شدن سیگنال reset بیت های Dirty , Valid , LRU همگی رسیت می شود.

توجه کنید که با توجه به آفست یدونه ای که خروجی تست بنچ داشت من یدونه سینکر گذاشتم و این سینکر این یک عدد کلاک رو بعد از ریست می خورد! و فقط جهت سینک شدن خروجی کد ام با تست بنچ و از بین رفتن یدونه آفست موجود این کار را کرده ام.

با مشخص شدن سیگنال های ورودی ابتدا وضع هیت شدن با نشدن با توجه به آدرس ورودی مشخص می شود و اگر ورودی هیت شود دیتا در خروجی قرار میگیرد.

با توجه به نوع درخواست و یا هیت شدن یا نشدن نوع عملیات مشخص می شود

- اگر دیتایی که هیت شده قرار است نوشته شود علاوه بر تغییر LRU لازم است دیرتی بیت یک شود و دیتا در کش نوشته شود.
- اگر دیتا که هیت نشده باشد و قرار است خوانده شود memsys دیتای مورد نظر را از Ram می خواند و این دیتا در خروجی قرار میگیرد و LRU عوض می شود و بیت valid یک و بیت Dirty صفر می شود و مقدار tag نیز جایگزین می شود.
- اگر دیتایی که هیت نشده قرار است نوشته شود مقدار دیتا در کش نوشته می شود و LRU عوض می شود و بیت valid یک و بیت dirty یک می شود و مقدار tag جایگزین می شود.
- اگر در دو حالت آخر دیتایی که قرار است پاک شود dirtybit یک داشته باشد باید در Ram نوشته شود و برای اینکار یک بافر در ماژول memsys قرار دارد که اگر Ram در حال خواندن نباشد دیتا را در مموری می نویسد.

ریزه حالت بندی ها و جزییات در کد ضمیمه شده است به همراه کلی کامنت خوش مزه!

جزییات پیاده سازی را در کد ببینید!

پاسخ خواسته ی 3:

با توجه به بالا بودن MissRate چه باید کرد؟ باید به طریقی بتوان تعداد word داده های بیشتری را در کنار هم قرار بدهیم
راه حل:

- افزایش سایز بلوک ها

• افزایش تعداد راه های Ways of associativity

در اصل این کار کمک می کند تا نقش spatial locality هر چه بیشتر خودش رو نشان بدهد و لذا در نتیجه خواهیم داشت که missRate کاهش پیدا کرده و عملکرد کش بهتر خواهد شد.

در صورت ران کردن تست بنچ ها هم خواهید دید که خروجی کد منطبق بر خروجی های تست بنچ می باشد.

اما برویم سراغ کارهای مازاد انجام شده!!!

من خروجی فاز های قبل رو به کمک کش گرفتم و خب این کار شگرفی است!!!

و کد کامل این پیاده سازی ضمیمه شده است.

In order to see the result, go to BOMB folder then copy all the folder's files into Your modelSim project and the run the biggest! test Bench.

Run the simulation → test bench with name multiCycleTestBench.

نکته : توجه کنید که برای اینکه من از صحت عملکرد تک تک بخش ها خبردار بشوم لازم بود که بخش بخش تست بنچ بنویسم و ران کنم (چون این فاز خود ساخته پیچیدگی به نسبت زیادی داشت می خواستم از صحت عملکرد همه ی بخش ها مطلع بشوم) لذا لازم بود بخش بخش تست بنچ بنویسم که همه ی تست بنچ ها ضمیمه شده است و در نهایت هم خروجی فاز های قبل را گرفتم. 😊😊😊😊

The End.