

بسمه تعالی  
گزارش پروژه اول درس معماری کامپیوتر  
استاد: دکتر زرندی

اعضای گروه:

زهرایوسفی 9531086

آذین اله خانی 9531005

● توضیح روش ایجاد همجواری مکانی و زمانی:

برای ایجاد همجواری مکانی ابتدا تعدادی آدرس تولید میکنیم به این صورت که به ازای هر آدرس تولید شده تعداد رندومی از آدرس های پس از آن را نیز تولید کرده و به آدرس های اولیه اضافه میکنیم. این تعداد رندوم از ۰ تا ۵ متغیر است. سپس برای ایجاد همجواری زمانی ۵۰ بار و به صورت رندوم دسته های ۱۰ تایی از این آدرس های را انتخاب کرده و به لیست آدرس های نهایی می افزاییم. با این روش در نهایت ۵۰۰ آدرس با همجواری مکانی و زمانی خواهیم داشت. بازه عدد آدرس های از ۰ تا ۳۰۰ است.

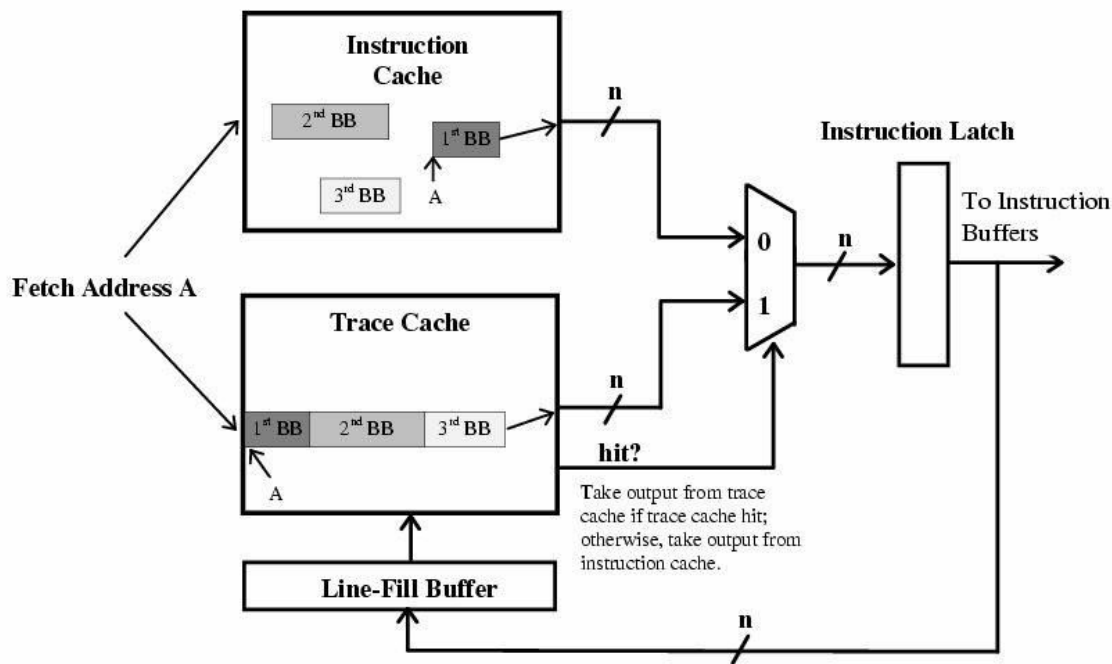
● شرح انواع حافظه نهان:

1. Trace Cache:

پردازشگرها با کارایی بالا نیازمند دریافت دستورات متعدد در هر cycle هستند. همچنین تاخیر در دریافت دستورات هم باید کاهش یابد. یکی از راه های دست یافتن به چنین پردازشگری استفاده از Trace Cache است. ایده اصلی این حافظه نهان این است که دستورات غیر مجاور را در کنار هم قرار دهد.

این حافظه نهان برای افزایش پهنای باند از یک حافظه اضافی برای دستورات استفاده میکند و دستورات را به صورت trace هایی از دستورات دریافتی و دیکود شده ذخیره می کند به طوری که خواندن از آن این امکان را به ما میدهد که دستورات با ترتیب دلخواه (در مجاورت هم) دریافت کنیم.

از معایب این روش میتوان به پیچیدگی سخت افزاری، ناکارآمد بودن توان مصرفی و وجود یک حافظه اضافی بین Instruction Cache و Trace Cache اشاره کرد.

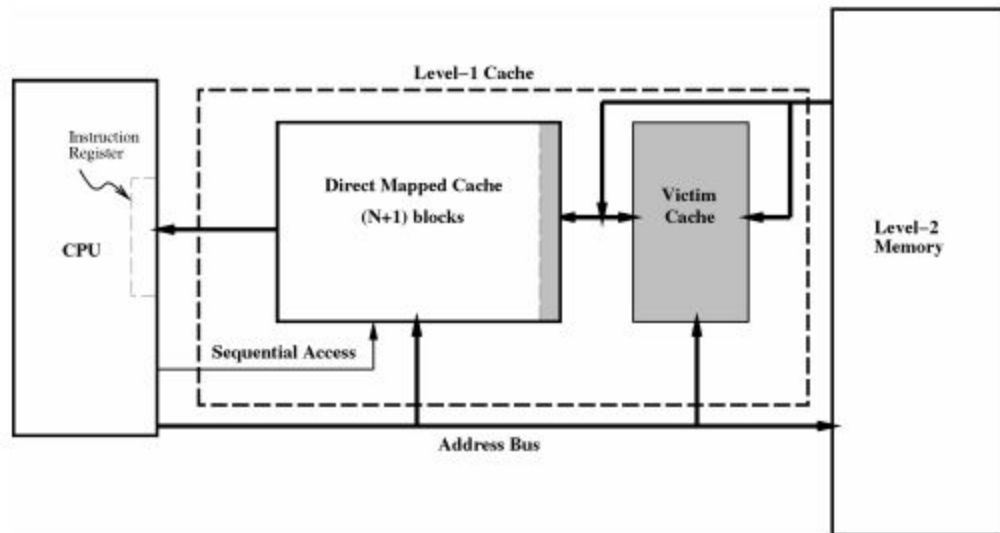


## 2. Victim Cache:

یکی از روش های کاهش تعداد miss ها در حافظه این است که بخاطر داشته باشیم کدام داده ها دور ریخته شده اند. چون داده دور ریخته شده قبلا از حافظه اصلی استخراج شده میتواند دوباره با هزینه کمی مورد استفاده قرار گیرد. چنین بازیافت داده ها با استفاده از Victim Cache ممکن است.

این حافظه نهان که ابتدا برای کاهش مناطق Hot در حافظه نهان نگاشت مستقیم بدون تاثیر گذاشتن برا زمان دسترسی بالای آن معرفی شد، Fully Associative بوده و اندازه آن به طور متوسط ۴ تا ۱۶ Cache Line است و بین حافظه نهان نگاشت مستقیم L1 و سطح بعدی بعدی حافظه قرار میگیرد.

وقتی در حافظه نهان حافظه miss داریم پیش از آنکه به سراغ سطح بعدی حافظه برویم Victim Cache را چک میکنیم. اگر داده موردنظر در این حافظه نهان باشد داده مورد نظر به CPU و حافظه نهان منتقل میشود و داده ای که این داده به جای آن در حافظه نهان قرار گرفته به Victim Cache انتقال می یابد و در واقع جای این دو داده با هم عوض میشود. اگر داده موردنظر در Victim Cache نیز پیدا نشد به بررسی سطح بعدی حافظه می پردازیم و داده به دست آمده از این جستجو به حافظه نهان انتقال یافته و داده ای که این داده در حافظه نهان جایگزین آن شده به Victim Cache انتقال میابد. داده ای که از Victim Cache خارج میشود در صورت dirty بودن به سطح بعدی حافظه انتقال می یابد و در غیر این صورت دور ریخته میشود. برخی سیستم های اخیر AMD از این نوع حافظه نهان در سطح سوم L3 استفاده میکنند.



● بررسی نتایج:

	File1.text	File2.text	File3.text	File4.text	File5.text	Average
With L2 Cache	90	91.4	88	90.6	88.8	89.76
Without L2 Cache	61.2	65	52.2	59.4	52.2	58

در نتیجه با استفاده از Victim Cache در سطح اول حافظه شاهد افزایش Hit Rate و بهبود عملکرد حافظه هستیم.