

بنام خدا

امین رشیدبیگی

شماره دانشجویی: 9431009

درس : معماری کامپیوتر

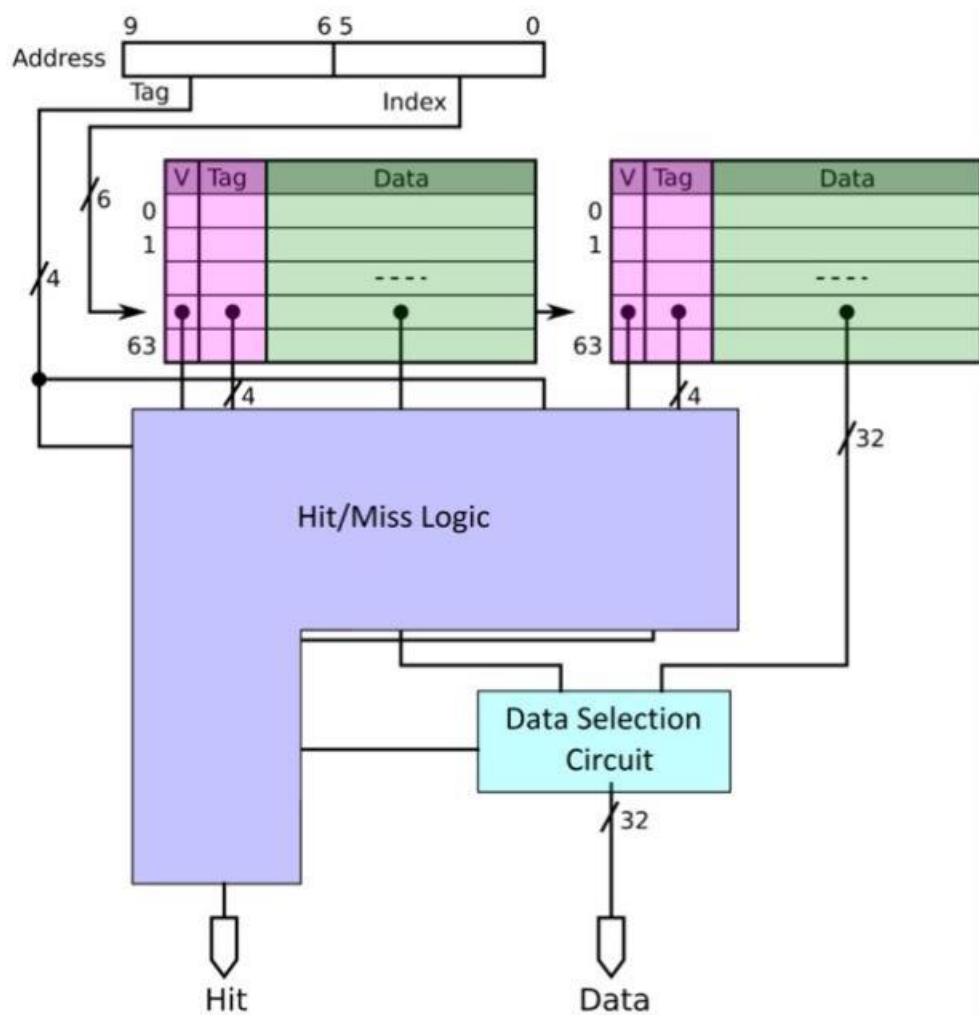
استاد : دکتر شیری

SAYEH Cache Ram

توضیحات کلی

در کامپیوتر پایه سایه در بخش قبلی پروژه پیاده سازی گردید، تنها حافظه موجود برای ذخیره سازی اطلاعات، Ram بود. برای بهینه سازی و افزایش کارایی و سرعت کامپیوتر پایه سایه، یک 2-way associative cache طراحی و پیاده سازی شد که با ram قبلی نیز ارتباط برقرار می کند. به این صورت که اگر داده ای از حافظه درخواست شود، ابتدا cache چک می شود و اگر در کش قرار داشت، داده با سرعت بسیار بیشتر از ram پس داده می شود ولی اگر در کش وجود نداشته باشد علاوه بر این که در اختیار کاربر قرار داده می شود، برای استفاده های احتمالی بعدی در cache هم ذخیره می شود.

ساختار کلی کش :



در ادامه، به بخش های مختلف SAYEH Cache Ram می پردازیم.

آرایه داده (Data Array)



همانطور که از اسم ماژول مشخص است، این ماژول وظیفه نگه داری داده هایی را دارد که از حافظه اصلی آمده است. این ماژول دارای یک آرایه 64 تایی شامل داده های 32 بیتی است.

- سیگنال ورودی address: این سیگنال خط آرایه 64 تایی را مشخص می کند.
- سیگنال ورودی wren: این سیگنال وظیفه چک کردن فعال بودن یا نبودن برای نوشتن در Data Array را بر عهده دارد.
- سیگنال ورودی wrdata: این سیگنال داده ای که برای نوشتن در Data Array را در خود دارد
- سیگنال خروجی data: این سیگنال که 32 بیتی است، داده ای را که در آدرس address قرار دارد را به بیرون می دهد.

توجه داشته باشید که چون از تکنولوژی 2-way associative استفاده شده است، به دو ماژول Data Array نیاز داریم.

آرایه تگ-ولید (Tag-Valid Array)



این ماژول دارای یک آرایه 64 تایی شامل داده های 5 بیتی است. همانند Data Array سیگنال address خطی از آرایه را نشان می دهد که باید روی آن کار کنیم. در این ماژول محتوای آرایه خط address به خروجی منتقل می شود. همچنین هنگامی که wren فعال باشد، دیتای wrdata بر روی 4 بیت کم ارزش خانه address آرایه سوال می شود. همچنین اگر invalidate 1 باشد، بیت پرارزش خانه address آرایه، برابر با متمم invalidate می شود و معتبر بودن آن خانه از cache را نشان می دهد.

- سیگنال ورودی address: این سیگنال خط آرایه 64 تایی را مشخص می کند.
- سیگنال ورودی wren: این سیگنال وظیفه چک کردن فعال بودن یا نبودن برای نوشتن در Data Array را بر عهده دارد.
- سیگنال ورودی wrdata: این سیگنال داده ای که برای نوشتن در Data Array را در خود دارد.
- سیگنال ورودی invalidate: این سیگنال معتبر بودن یا نبودن داده داخل در خانه address آرایه را مشخص می کند.
- سیگنال reset_n: این سیگنال، تمامی خانه های داخل Tag Valid Array را صفر میکند.
- سیگنال خروجی output: این سیگنال که 5 بیتی است، داده ای را که در آدرس address قرار دارد را به بیرون می دهد.

توجه داشته باشید که چون از تکنولوژی 2-way associative استفاده شده است، به دو ماژول Tag-Valid Array نیاز داریم.

واحد میس-هیت (Miss-Hit Logic)



این ماژول وظیفه تعیین شدن Miss یا Hit شدن داده خواسته شده در Cache و همچنین مشخص کردن valid بودن way ها است.

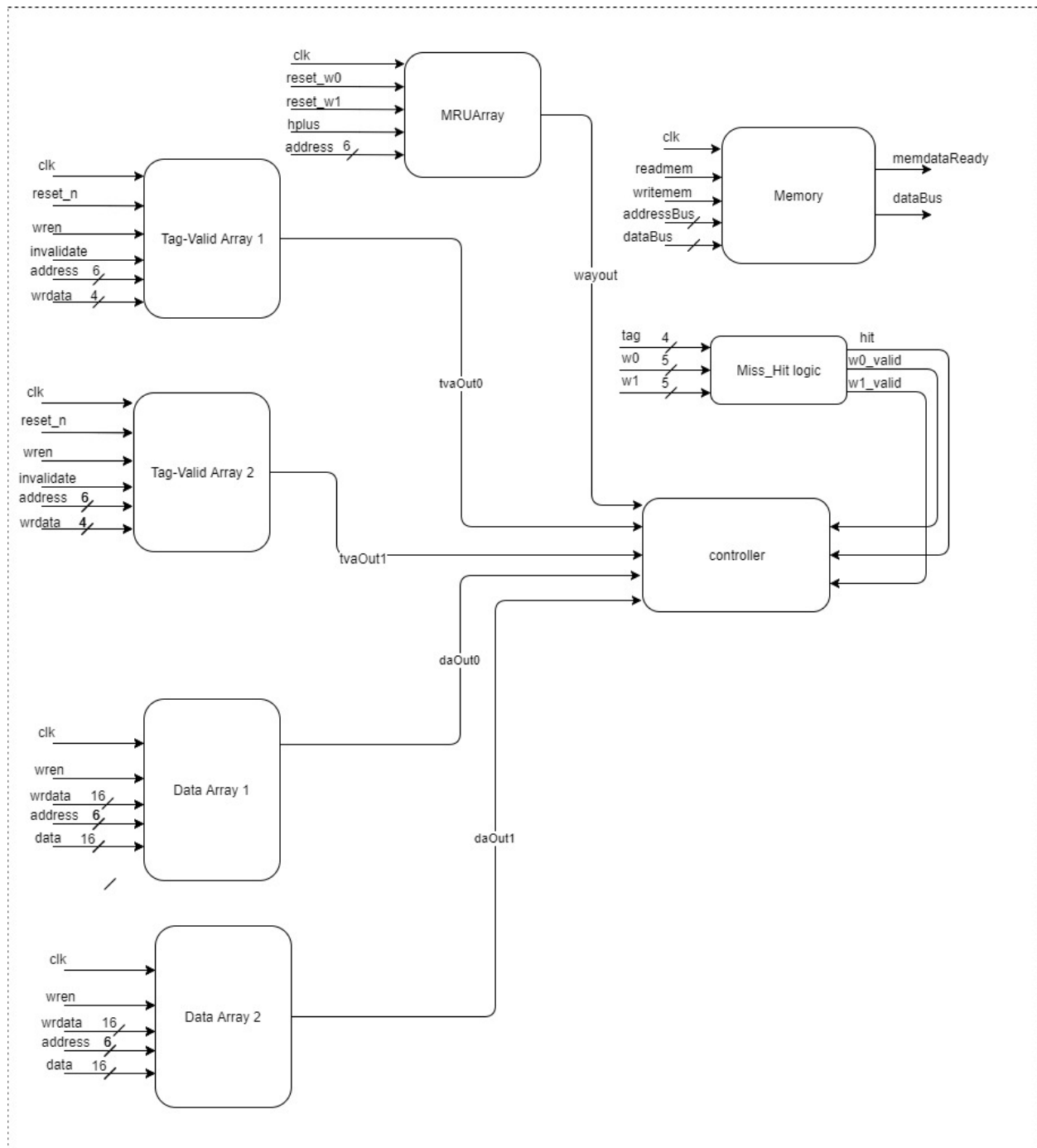
- سیگنال ورودی tag: چهار بیت پر ارزش آدرس
- سیگنال های ورودی w0 و w1: جهت مقایسه با tag
- سیگنال خروجی hit: اگر tag با یکی از چهار بیت کم ارزش w0 و w1 برابر باشد hit یک می شود.

MRU

این ماژول مشخص می کند که داده بعدی ای که قرار است در کش نوشته شود، در کجا نوشته شود. با توجه به قراردادی که تعیین شده است ، باید داده ی جدید در قسمتی نوشته شود که بیشترین استفاده از آن شده باشد . به همین دلیل به ازای w_0 و w_1 یک آرایه شمارنده در نظر می گیریم و تعداد دفعاتی که داده مربوط به آن خانه از آرایه ، خوانده می شود، شمارنده اضافه می شود.

دیتاپث (Datapath)

وظیفه Datapath، به هم وصل کردن component هایی است که در بالا وظایفشان تعریف شد. ورودی های این component تمامی سیم هایی است که در شکل زیر مشخص شده است:



کنترلر (Controller)

واحد کنترل وظیفه کنترل سیم هایی که در Datapath تعریف شد را دارد. برای پیاده سازی واحد کنترل از دو حالت `current_state` و `next_state` استفاده شده است. به این صورت که `current_state` حالت فعلی و `next_state` حالت بعدی را مشخص می کند و در هر حالت، `next_state` مقدارش تغییر میکند.

ابتدا همه سیگنال ها صفر هستند. اولین حالت `reset` نام دارد. سپس به حالت `init` می رود. اگر سیگنال `is_read` فعال باشد وارد استیت `read` می شود همچنین اگر سیگنال `is_write` فعال باشد وارد استیت `write` می شود. در استیت `read` اگر `hit` فعال باشد، حالت بعدی `reset` است. اگر `hit` فعال نباشد وارد استیت `read_from_memory` می شود. در این استیت سیگنال ها برای خواندن داده از مموری فعال می شود. سپس به یک کلاک تاخیر نیاز است. وارد استیت `wait` می شود. در مرحله بعدی وارد `write_to_cache` می شود. در این مرحله سیگنال های نوشتن رو روی کش فعال می شود و وارد مرحله `reset` می شود.

