



## آزمایش پنجم: استفاده از حافظه نهان (Cache) در پردازنده ARM

گردآورندگان:

دکتر علیرضا یزدان‌پناه

مهندس ادریس نصیحت‌کن

مهندس مرضیه رستگار

### اهداف

- ۱- آشنایی با حافظه نهان و کاربرد آن
- ۲- استفاده نمودن از حافظه‌ی ON-CHIP به عنوان حافظه‌ی نهان در پردازنده
- ۳- تغییرات افزودن حافظه نهان در معماری پردازنده
- ۴- تاثیر حافظه نهان در کارایی پردازنده

### توضیحات کلی

- ۱- در آزمایش‌های قبل یک پردازنده MIPS پیاده سازی گردید که از حافظه خارجی SRAM به عنوان حافظه داده استفاده شد. زمان خواندن و نوشتن در حافظه خارجی چندین کلاک است. هنگامی که نیاز به خواندن از حافظه یا نوشتن در آن است، پردازنده باید متوقف (Freeze) شود، تا عملیات حافظه به اتمام رسد. بنابراین دسترسی به حافظه به عنوان یک گلوگاه در کارایی پردازنده مطرح می شود.
- ۲- بهترین روشی که برای رفع مشکل فوق و بهبود کارایی پردازنده ارائه شده است، استفاده از حافظه نهان است.
- ۳- حافظه نهان یک حافظه بسیار سریع است که از نوع پردازنده ساخته می شود. بنابراین این نوع حافظه قابلیت دسترسی در یک کلاک را داراست. از این رو اگر به جای دسترسی به حافظه اصلی به حافظه نهان مراجعه شود، آنگاه نیاز به متوقف کردن پردازنده برای انجام عملیات حافظه نیست و کارایی پردازنده افزایش چشمگیری خواهد داشت.
- ۴- حافظه نهان قیمت بالایی دارد و مساحت زیادی را اشغال می کند، بنابراین نمی توان حافظه زیادی از این نوع در پردازنده قرار داد. حافظه نهان درصد بسیار کمی از فضای آدرس دهی حافظه اصلی را تحت پوشش خود قرار می دهد. معماری حافظه نهان، اینکه چه داده‌هایی از حافظه اصلی را داخل حافظه نهان قرار داد و نحوه جایگزینی داده ها به چه شکل باشد، مسائلی مهمی هستند.
- ۵- در این آزمایش از حافظه روی تراشه FPGA به عنوان حافظه نهان استفاده می شود و کارایی پردازنده را پس از اضافه کردن آن بررسی می گردد.



## دستور کار

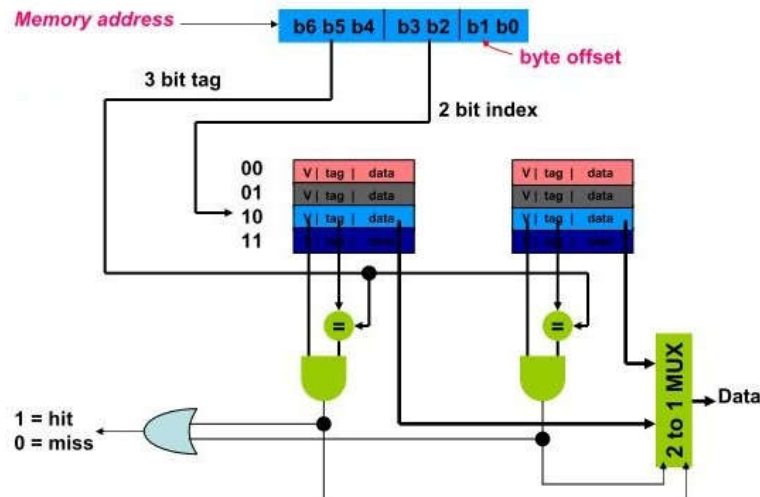
ابتدا از عملکرد درست پردازنده‌ای که در جلسات گذشته پیاده سازی کرده اید، مطمئن گردید. پردازنده شما باید شامل یک حافظه خارجی دارای تاخیر دسترسی ۶ کلاک باشد که با استفاده از SRAM روی برد DE2 طراحی شده است. پس از آن یک حافظه نهان با مشخصات زیر را پیاده سازی و آن را به پردازنده‌ی خود اضافه نمایید.

### مشخصات ساختاری حافظه نهان:

- معماری 2-Way Set Associates
- اندازه هر کلمه: 32 بیت
- اندازه هر بلاک: 64 بیت (2 کلمه)
- تعداد مجموعه‌ها (set): 64
- اندازه حافظه نهان: 1 کیلو بایت برای داده (همچنین حدود 1 کیلو بایت هم برای نشانه‌ها و غیره مورد نیاز است).
- گذرگاه آدرس: 19 بیت
- تعداد بیت مورد نیاز برای نشانه (tag): 10 بیت
- تعداد بیت شاخص (index): 6 بیت
- دارای بیت اعتبار (valid) برای هر بلاک
- هر مجموعه نیاز به یک بیت (used) برای سیاست جایگزینی LRU دارد.

\* الگوریتم جایگزینی حافظه نهان LRU است، یعنی قدیمی‌ترین داده‌ای که از آن استفاده شده است، باید جایگزین شود.  
\* سیاست بازنویسی داخل حافظه نهان رویه نویسی کامل (Write Through) است، یعنی به محض تغییر داده در حافظه نهان داده در حافظه اصلی نوشته می‌شود.

در شکل 1 نمونه‌ای از حافظه نهان way-2 نشان داده شده است. (توجه کنید که اندازه‌ی این حافظه نهان مطابق اطلاعات گفته شده در بالا نمی‌باشد. همچنین حافظه در اینجا Byte addressable نمی‌باشد).



شکل 1- نمونه ای از یک حافظه نهان way-2

## روش پیاده سازی:

حافظه‌های مورد نیاز می‌تواند به صورت زیر در نظر گرفته شود:

- 4 بلاک حافظه با طول کلمه‌ی 32 بیت و گذرگاه آدرس 6 بیت برای داده‌های اصلی (حافظه نهان way-2 است که هر مجموعه شامل دو بلاک می‌باشد و هر بلاک هم شامل دو کلمه است، بنابراین نیاز به 4 بلاک داده است).
- 2 بلاک حافظه با طول کلمه 10 بیت و گذرگاه آدرس 6 بیت برای بیت‌های نشانه
- 2 بلاک حافظه با طول کلمه‌ی 1 بیت و گذرگاه آدرس 6 بیت برای بیت اعتبار
- 1 بلاک حافظه با طول کلمه‌ی 1 بیت و گذرگاه آدرس 6 بیت برای پیاده سازی الگوریتم جایگزینی LRU

همانطور که در شکل زیر مشاهده می‌شود، ماژول کنترل حافظه نهان بین حافظه اصلی و پردازنده قرار می‌گیرد. پردازنده ابتدا به حافظه نهان مراجعه می‌کند، در صورتی که داده در آنجا موجود باشد، به درخواست حافظه پاسخ داده می‌شود. در صورتی که داده در حافظه نهان موجود نباشد، به حافظه اصلی ارجاع داده می‌شود. اگر درخواست خواندن به حافظه باشد، ابتدا یک بلاک از حافظه اصلی به حافظه نهان منتقل می‌گردد و سپس به درخواست حافظه پردازنده پاسخ داده می‌شود. اما در صورتی که درخواست نوشتن در حافظه باشد، تنها در حافظه نوشته می‌شود به داده به حافظه نهان وارد نمی‌شود (No-write allocate).



## آزمایش پنجم: استفاده از حافظه نهان (Cache) در پردازنده ARM



شکل 2- نحوه ارتباط حافظه نهان به پردازنده و حافظه اصلی

\*\*\* نکته: برای سادگی بهتر است که هیچ سیگنالی (حتی سیگنال کنترلی) بین پردازنده و حافظه اصلی وجود نداشته باشد.

یعنی تمامی ارتباطات و کنترل از طریق خود حافظه نهان صورت گیرد. \*\*\*

ماژول کنترل حافظه نهان واسطی مطابق شکل 3 دارد. در صورت نیاز می توانید آن را تغییر دهید.

```
1 module cache_controller(  
2     input clk,  
3     input rst,  
4  
5     //memory stage unit  
6     input [31:0] address,  
7     input [31:0] wdata,  
8     input      MEM_R_EN,  
9     input      MEM_W_EN,  
10    output [31:0] rdata,  
11    output      ready,  
12  
13    //SRAM controller  
14    output [31:0] sram_address,  
15    output [31:0] sram_wdata,  
16    output      write,  
17    input  [63:0] sram_rdata,  
18    input      sram_ready  
19 );
```

شکل 3- پورت های ماژول کنترل حافظه نهان

پس از پیاده سازی حافظه نهان ابتدا آن را با برنامه ساده‌ای که قبلاً داشتید، تست نمایید. سپس یک برنامه بزرگ که نیاز به داده‌های زیادی روی حافظه اصلی دارد، بنویسید و آن را در دو حالت با حافظه نهان و بدون آن اجرا نمایید. نتایج حاصل را در گزارش کار آورده و مقایسه نمایید.



تاثیر میزان دسترسی به حافظه (هم از نظر تعداد دفعات و هم از نظر تعداد خانه‌هایی که به آنها ارجاع می‌شود) روی کارایی پردازنده را بررسی نمایید.

\*\*\* برای انجام این آزمایش دو جلسه فرصت دارید. \*\*\*

### پیش گزارش:

- مطالب مربوط به حافظه نهان را مطالعه نمایید.
- ساختار حافظه نهان، شامل مجموعه‌ها، بلاک‌ها و کلامت را به طور کلی رسم نمایید.
- RTL ماژول کنترل کننده حافظه نهان را رسم نمایید.

### گزارش کار:

- نتایج اجرای برنامه‌های گفته شده روی برد را توضیح دهید و میزان کارایی پردازنده را در سه حالت زیر مقایسه کنید:
  - حالت اول: از حافظه داخلی برای حافظه داده استفاده شد.
  - حالت دوم: از حافظه SRAM موجود روی برد برای حافظه داده استفاده شد.
  - حالت سوم: از حافظه SRAM موجود روی برد برای حافظه داده استفاده شد و حافظه نهان به پردازنده اضافه گردید.
- در قسمت بعد نتایج سنتز آورده شود، و هزینه سخت افزاری را در سه حالت گفته شده در بالا، مقایسه کنید.
- نمره اضافی:  
در این قسمت پیشنهاداتی در زمینه معماری حافظه نهان، نحوه پیاده سازی آن، الگوریتم جایگزینی و سایر نکاتی که می‌تواند کارایی پردازنده را بهتر کند، ارائه کنید. به پیشنهادات خوب نمره اضافی تعلق می‌گیرد.