



## آزمایش سوم: پیاده سازی واحد تشخیص هازارد در پردازنده MIPS

گرد آورنده: علیرضا یزدان پناه- ادریس نصیحت کن

### پیاده سازی واحد تشخیص هازرد

#### اهداف

- ۱- یادگیری روش های مقابله با هازارد خواندن پس از نوشتن (RAW)
- ۲- استخراج موارد وقوع هازارد RAW
- ۳- نحوه پیاده سازی مازول تشخیص هازارد
- ۴- تکمیل پردازنده Mips
- ۵- نحوه کدنویسی Verilog با قابلیت سنتز
- ۶- نحوه عیب یابی و تست مازول های جدید

#### توضیحات کلی

- ۱- در این قسمت واحد تشخیص هازارد به پردازنده اضافه می گردد.
- ۲- پس از انجام این آزمایش گزارش این بخش با ذکر تغییرات اعمال شده در پردازنده به همراه نتایج ارائه شود.

#### دستور کار

در جلسات گذشته پردازنده ای کامل بدون واحد تشخیص هازارد پیاده سازی شد. در معماری پردازنده پیاده سازی شده هازردهای ساختاری و کنترلی رفع گردیده است، بنابراین تنها هازارد داده نیاز به تشخیص و رفع آن است. هازارد داده ای به صورت زیر دسته بندی می گردد:

- ۱- **خواندن پس از نوشتن (RAW):** هازارد خواندن بعد از نوشتن در حالتی رخ می دهد که یک دستور، رجیستری که هنوز محاسبه یا ذخیره نشده است را فراخوانی می نماید. در این حالت می بایست دستورات جدید تا محاسبه یا ذخیره شدن آن رجیستر متوقف گردند. در مثال زیر دستور ۲ نیاز به رجیستر R2 دارد که دستور ۱ آنرا محاسبه می نماید، بنابراین تا ذخیره یا محاسبه شدن مقدار R2 دستور ۲ باید متوقف گردد.

1. SUB R2,R0,R1
2. AND R3,R2,R1

در این آزمایش سعی در برطرف نمودن این نوع هازارد را داریم.



## آزمایش سوم: پیاده سازی واحد تشخیص هازارد در پردازنده MIPS

گرد آورنده: علیرضا یزدان‌پناه - ادریس نصیحت‌کن

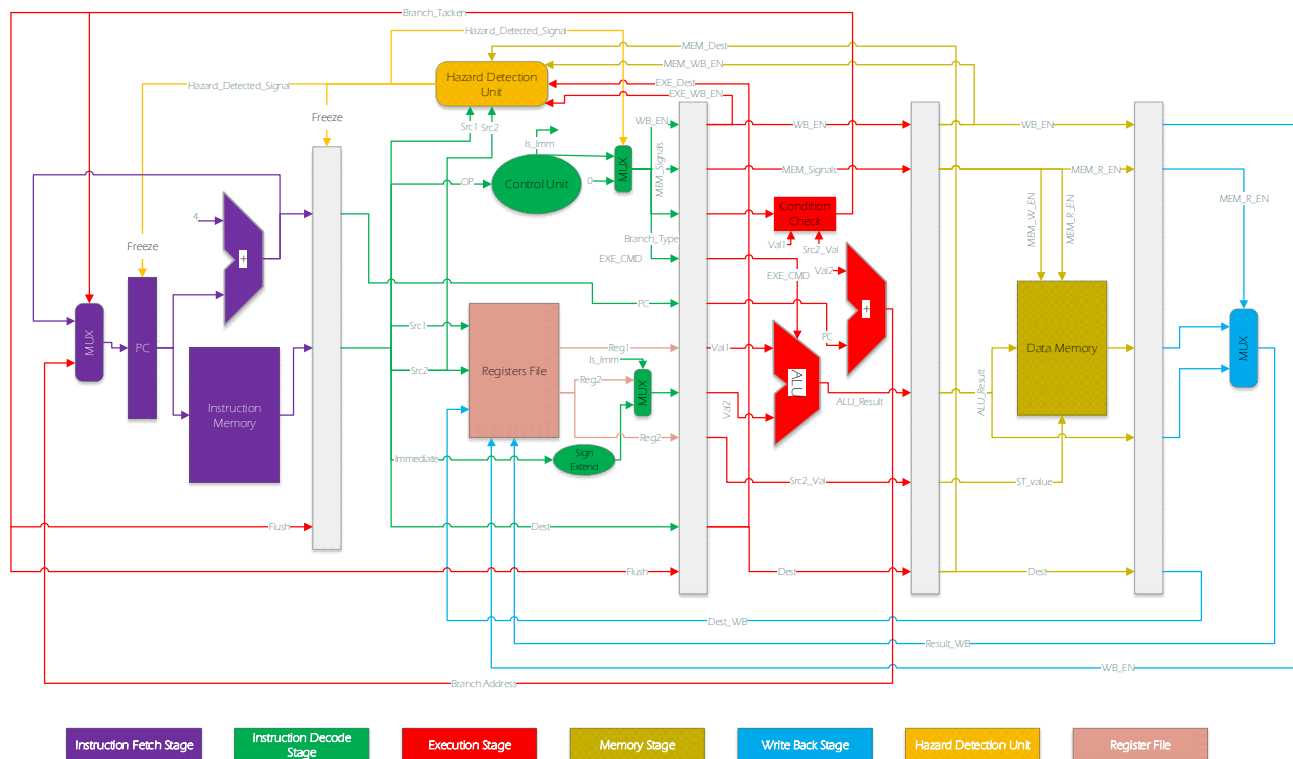
۲- نوشتن پس از خواندن (WAR): در مثال زیر رجیستر R1 در حالتی ممکن است قبل از خوانده شدن توسط دستور ۱ مقدار آن به وسیله دستور ۲ تغییر کند. به این رخداد هازارد داده‌ای از نوع WAR گفته می‌شود.

1. SUB R2,R0,R1
2. AND R1,R3,R4

این نوع هازارد در پردازنده‌های In-Order مانند پردازنده مورد بحث در این دستور کار رخ هرگز رخ نخواهد داد.

۳- نوشتن پس از نوشتن (WAW): هازارد داده‌ای نوشتن پس از نوشتن نیز همچون هازارد WAR در پردازنده‌های In-Order رخ نمی‌دهد.

برای رفع هازارد RAW ماژول Hazard Detection Unit همانند شکل زیر به پردازنده MIPS اضافه نمایید. در این واحد، منابع Src1 و Src2 در مرحله ID با مقصدهای مراحل EXE و MEM مقایسه می‌شود و در صورت برابر بودن سیگنال کنترلی Hazard\_Detected\_Signal را برابر ۱ قرار می‌دهد. این سیگنال باید دستورات درون IF و رجیسترهای پس از آن را متوقف نماید و حبابی را به خط لوله تزریق نماید.



شکل ۱: پردازنده MIPS همراه با واحد تشخیص هازارد

برای پیاده‌سازی این آزمایش به روش زیر عمل نمایید:

۱- واحد تشخیص هازارد با حداقل ورودی‌های زیر ایجاد نمایید.



## آزمایش سوم: پیاده سازی واحد تشخیص هازارد در پردازنده MIPS

گرد آورنده: علیرضا یزدان پناه- ادریس نصیحت کن

```
1 module hazard_Detection_Unit(  
2     input [4:0]src1,  
3     input [4:0]src2,  
4     input [4:0] Exe_Dest,  
5     input Exe_WB_EN,  
6     input [4:0]Mem_Dest,  
7     input Mem_WB_EN,  
8     output hazard_Detected  
9 );
```

- ۲- تمامی حالاتی که هازارد RAW رخ می دهد را در نظر بگیرید و خروجی واحد تشخیص هازارد را یک کنید.
- ۳- به دستورات Immediate که دارای تنها یک منبع هستند توجه نمایید. در این دستورات تنها یک منبع چک می گردد.  
نکته: دستورات ST و BNE دستوراتی از نوع Immediate هستند که دارای دو منبع هستند.
- ۴- برای تشخیص نوع دستورات به هیچ وجه از ارسال OP-Code به واحد تشخیص هازارد استفاده ننمایید، در صورت نیاز سیگنال جدیدی برای این منظور تعریف کنید.
- ۵- ورودی های مورد نیاز واحد هازارد را به خروجی دیگر مراحل متصل نمایید.
- ۶- قابلیت Freeze را به رجیستر PC در مرحله IF و رجیستری های بعد از IF اضافه نمایید و خروجی واحد تشخیص هازارد را به آن متصل نمایید.
- ۷- برای ایجاد حباب در خط لوله سیگنال های کنترلی خروجی از واحد کنترل را صفر نمایید.
- ۸- دستورات برنامه محک را بدون تغییر اجرا نمایید.
- ۹- نتایج را به دستیار آموزشی ارائه دهید.

موفق باشید

نصیحت کن