

خط لوله ۲

معماری کامپیوتر ...

۱۳۰۱-۱۱-۱۳۰۱

جلسه شانزدهم



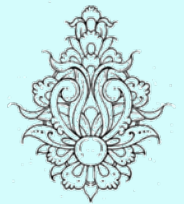
دانشگاه شهید بهشتی
دانشکده مهندسی برق و کامپیوتر

بهار ۱۳۹۲

احمد محمودی ازناوه

فهرست مطالب

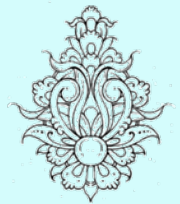
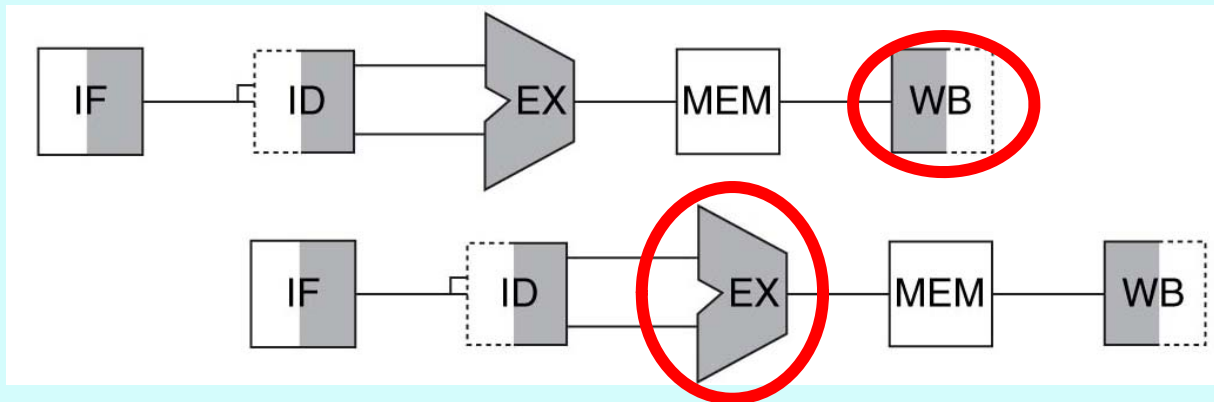
- مخاطرات خط لوله
- گذر داده‌ی مجهز به خط لوله



مفاهیم داده‌ای

- مفاهیم داده‌ای و موانع ناشی از داده موقعی رخ خواهد داد که خط لوله باید در انتظار تکمیل یکی از مراحل قبلی از حرکت باز داشته شود

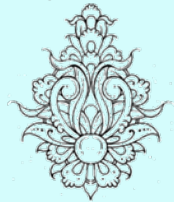
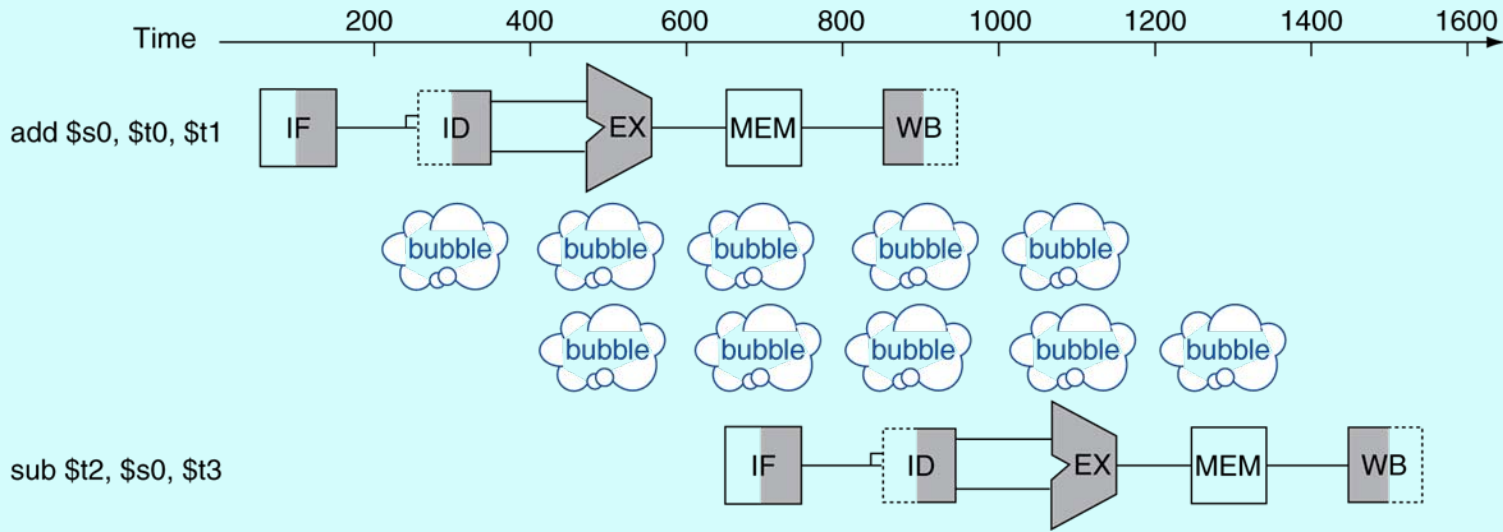
```
add $s0, $t0, $t1
sub $t2, $s0, $t3
```



مفاهیم داده‌ای

- یک دستورالعمل به داده‌ای نیاز دارد، که در دستور قبلی مشغول آماده کردن آن است

```
add $s0, $t0, $t1
sub $t2, $s0, $t3
```

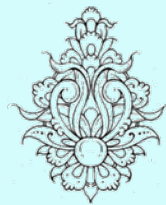
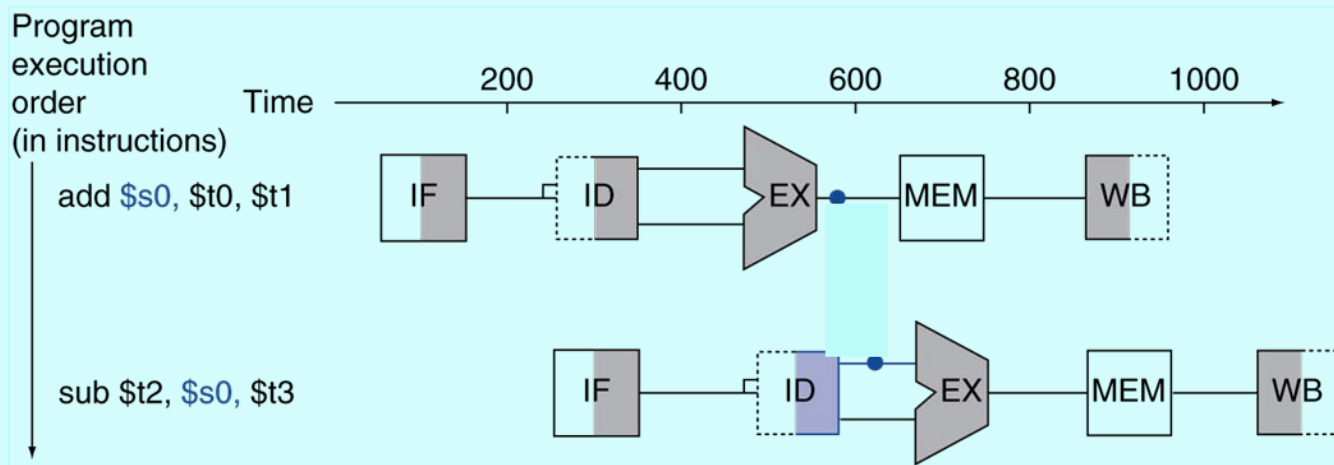


Forwarding or Bypassing

هدایت روبه جلو

شگرد پیش فرستادن

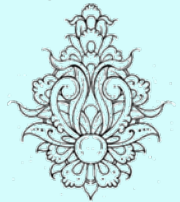
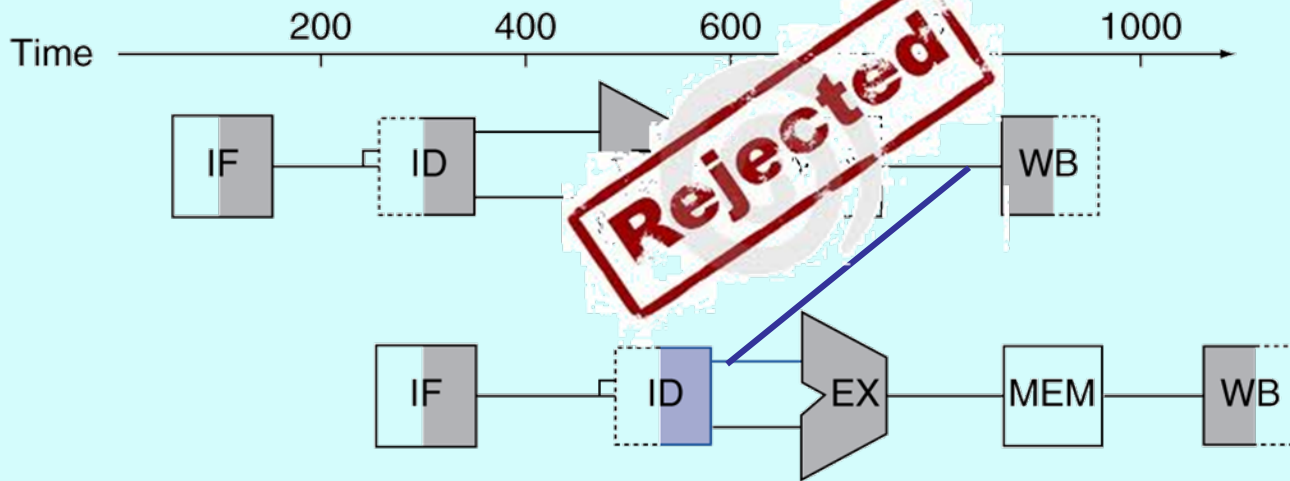
- در حالتی که **مخاطره‌ی داده** رخ می‌دهد، پس از انجام دستورالعمل و آمده شدن داده، نتیجه به دست آمده پیش از ذخیره در ثبات، در دستور بعدی استفاده می‌شود.
- نیاز به اتصالات بیشتری در داده‌گذر (datapath) دارد.



مخاطره‌ی load-use data

- نوع خاصی از مخاطره‌ی داده‌ای است که در آن داده‌ای در خواندن از حافظه است، هنوز برای استفاده در دستور بعدی آماده نیست.

IW *\$s0*, *20(\$t1)*
sub *\$t2*, *\$s0*, *\$t3*

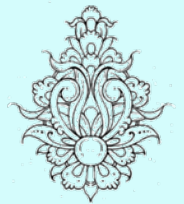
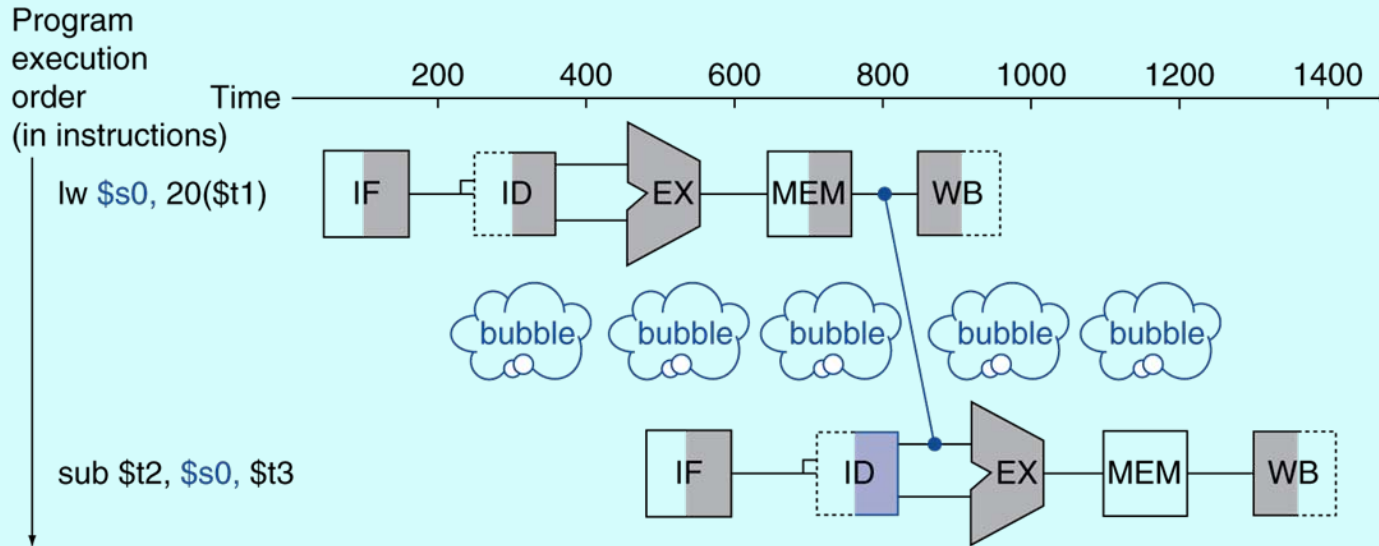


دانشگاه
تهران
پیشین

مفاهیمی load-use data (ادامه...)

– با پیش‌فرستادن نمی‌توان از وقوع تعلیق اجتناب کرد.

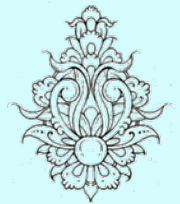
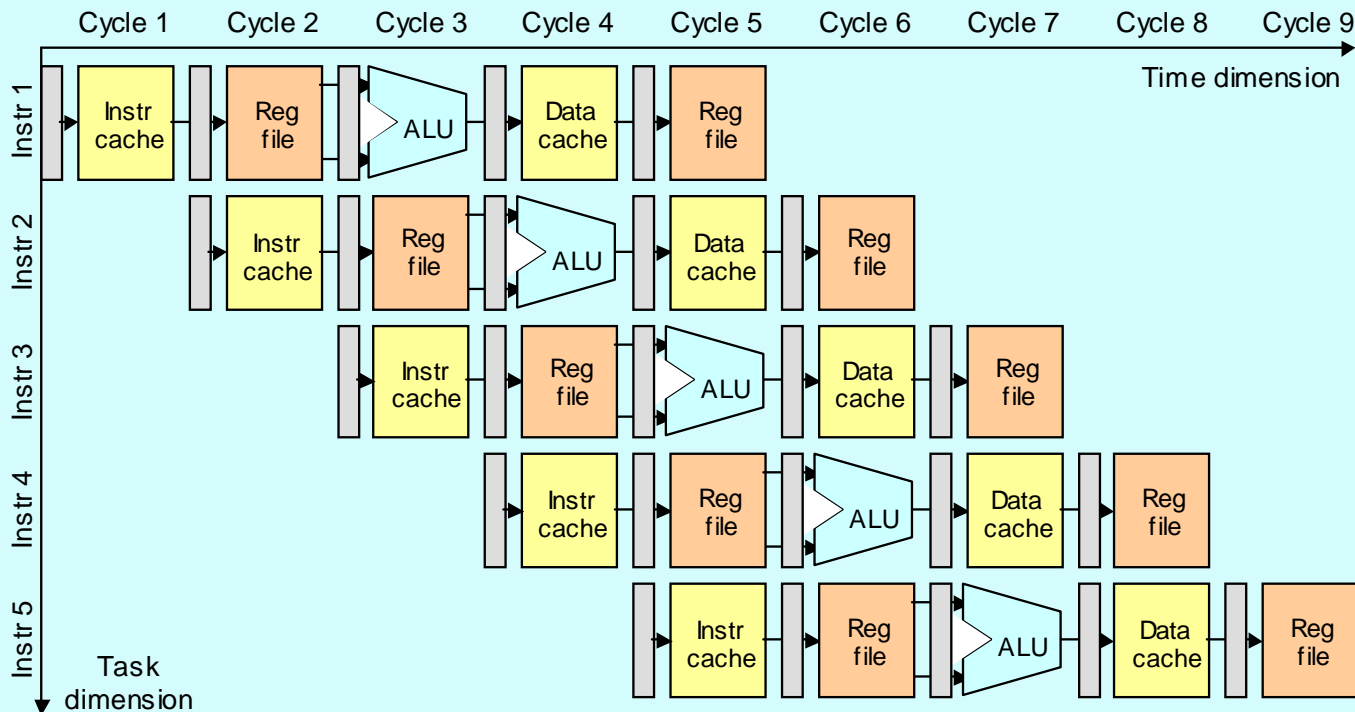
– نمی‌توان در زمان سفر کرد !!



مفاهیمی load-use data (ادامه...)

- در صورت عدم استفاده از پیش‌فرستادن چند مباب وارد خط لوله می‌شود؟

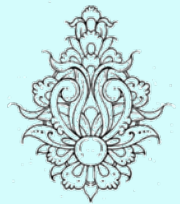
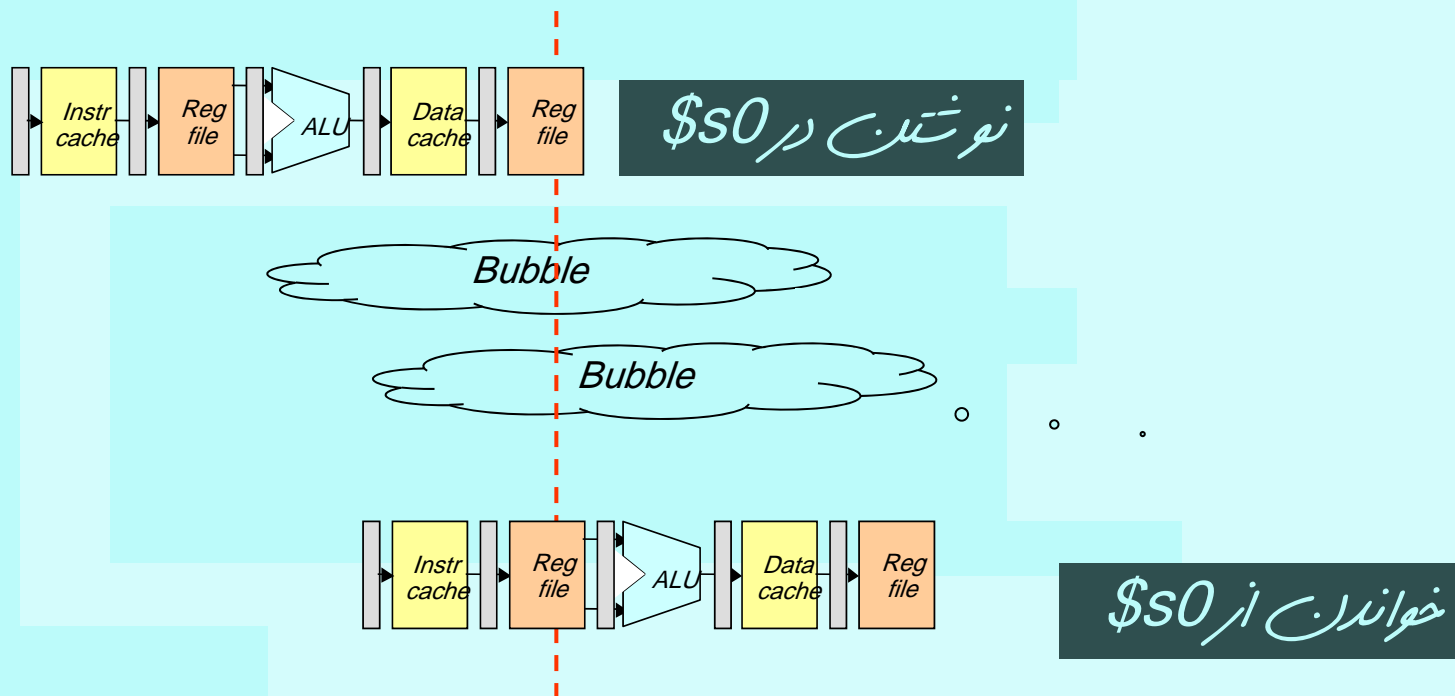
```
lw    $s0, 20($t1)
sub   $t2, $s0, $t3
```



مفاهیمی load-use data (ادامه...)

- در صورت عدم استفاده از پیش‌فرستادن چند مباب وارد خط لوله می‌شود؟

```
lw    $s0, 20($t1)
sub   $t2, $s0, $t3
```



تغییر ترتیب برنامه

- با جابجا کردن کد می‌توان طول اجرای برنامه را کاهش داد. (از پیش‌فرستادن استفاده می‌شود.)

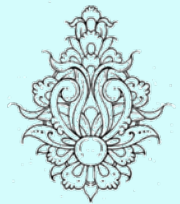
$$A = B + E;$$
$$C = B + F;$$

	<i>lw</i>	<i>\$t1</i> ,	<i>0(\$t0)</i>
	<i>lw</i>	<i>\$t2</i> ,	<i>4(\$t0)</i>
stall	<i>add</i>	<i>\$t3</i> ,	<i>\$t1</i> , <i>\$t2</i>
	<i>sw</i>	<i>\$t3</i> ,	<i>12(\$t0)</i>
	<i>lw</i>	<i>\$t4</i> ,	<i>8(\$t0)</i>
stall	<i>add</i>	<i>\$t5</i> ,	<i>\$t1</i> , <i>\$t4</i>
	<i>sw</i>	<i>\$t5</i> ,	<i>16(\$t0)</i>

13 cycles

	<i>lw</i>	<i>\$t1</i> ,	<i>0(\$t0)</i>
	<i>lw</i>	<i>\$t2</i> ,	<i>4(\$t0)</i>
	<i>lw</i>	<i>\$t4</i> ,	<i>8(\$t0)</i>
	<i>add</i>	<i>\$t3</i> ,	<i>\$t1</i> , <i>\$t2</i>
	<i>sw</i>	<i>\$t3</i> ,	<i>12(\$t0)</i>
	<i>add</i>	<i>\$t5</i> ,	<i>\$t1</i> , <i>\$t4</i>
	<i>sw</i>	<i>\$t5</i> ,	<i>16(\$t0)</i>

11 cycles

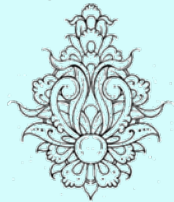
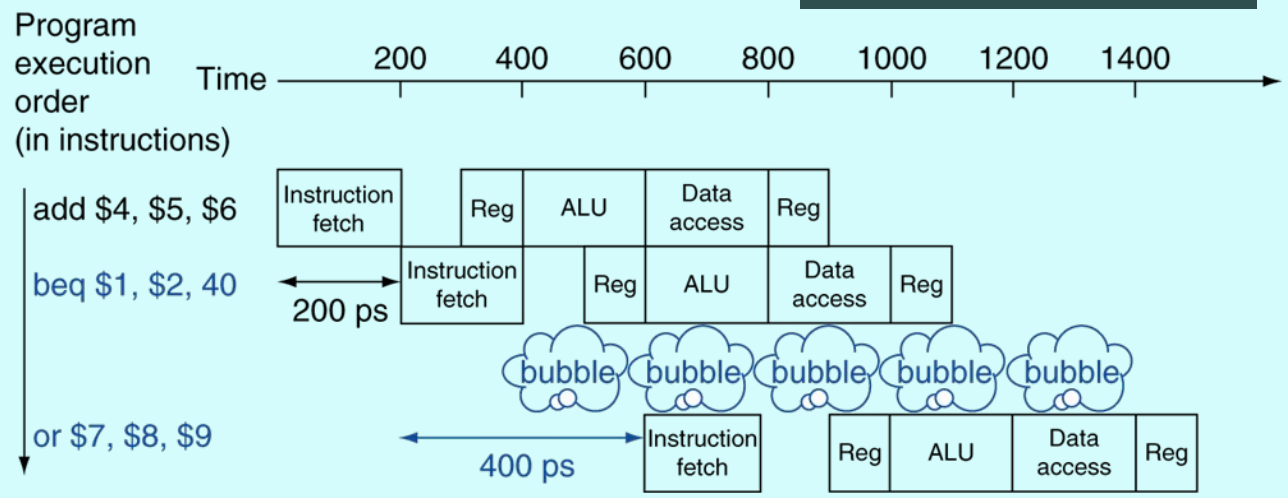


- در مواقعی که یک دستور پرش پرش شرطی وجود دارد، ادامه‌ی دستورهایی که باید توسط پردازنده اجرا شوند وابسته به تصمیمی است که در این دستور گرفته می‌شود.

چه راه حلی پیشنهاد می‌دهید؟

Stall on Branch

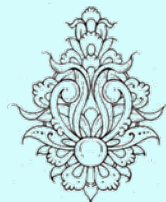
- تحلیل؟؟؟



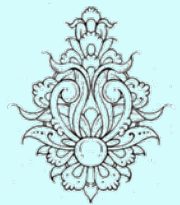
مخاطره‌ی کنترلی (ادامه...)

- در MIPS می‌توان با افزودن سخت‌افزار اضافی در گام ID، نتیجه‌ی شرط را زودتر به دست آورد.
 - برای خط لوله‌های که دارای تعداد گام بالا هستند، دستیابی سریع‌تر به نتیجه‌ی به سادگی امکان‌پذیر نیست.
- در چنین حالاتی زیان ناشی تحلیق، پذیرفتنی نیست.

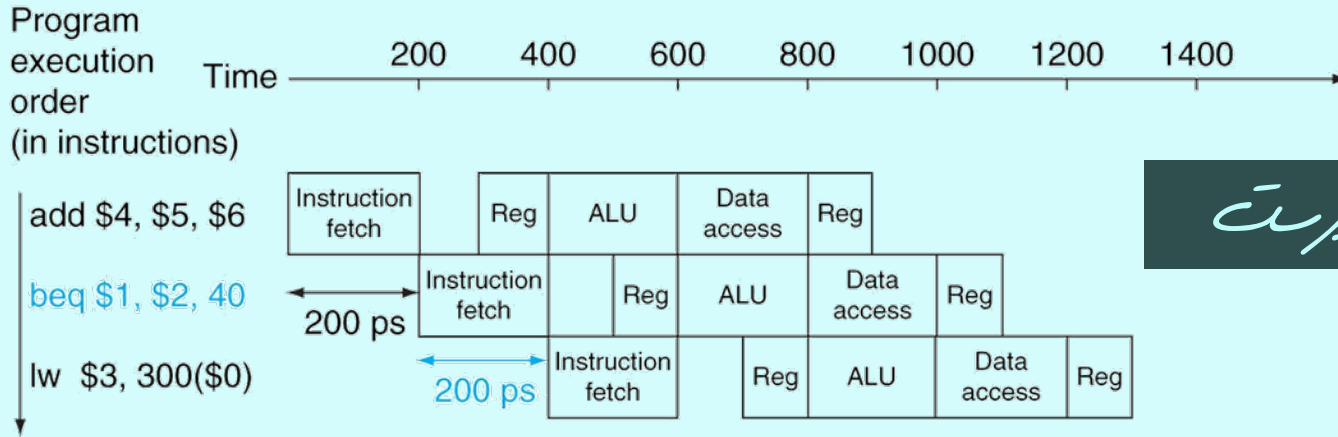
در صورتی که از تحلیق برای رفع مشکل مخاطره‌ی کنترلی استفاده شود، تأثیر دستورات پرش شرطی بر CPI چه خواهد بود؟



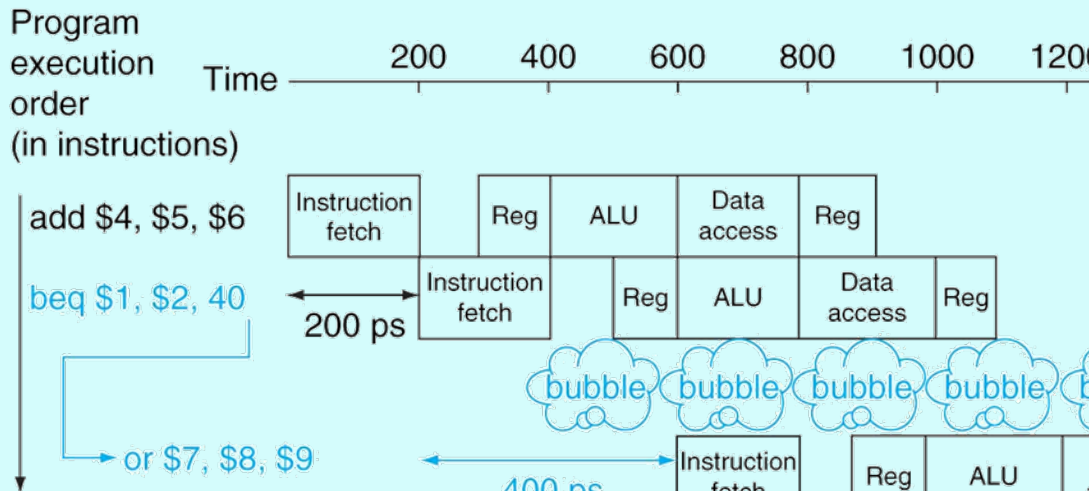
- استفاده از پیش‌بینی: یک راه ساده این است که همیشه فرض کنیم نتیجه شرط درست (نادرست) خواهد بود.
- در صورتی که پیش‌بینی ما درست باشد، خط لوله کار خود را به درستی انجام داده است.
- در غیر این صورت خط لوله دچار تحلیق می‌شود. ضمناً در چنین حالاتی باید مطمئن شویم که دستوراتی که به اشتباه وارد خط لوله شده‌اند، تأثیری از خود به جای نخواهند گذاشت



مفاهمی کنترلی (ادامه...)



پیش بینی درست



پیش بینی نادرست



- یک راه پیچیده‌تر، استفاده از پیش‌بینی واقع‌گرایانه‌تر است.

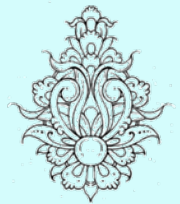
Predict backward branches taken

- مثلاً در حلقه‌ها پیش‌بینی می‌شود، که شرط برقرار است.
- در پیش‌بینی پویا، سخت‌افزاری برای پیش‌بینی عملکرد شرط در نظر گرفته می‌شود.

Dynamic branch prediction

- به عنوان مثال، می‌توان تاریخچه‌ای از رفتار شرط ذخیره نمود.
- راه سومی هم هست که در راه سومی هم هست که در MIPS مورد استفاده قرار می‌گیرد و آن این است که اسمبلر دستوراتی را که در شرط مؤثر نیستند، به بعد از دستور پرش منتقل می‌کند.

Delayed decision



گذرداده‌ی مجهز به نطلوله

جرمان داده از راست به چپ منبر
به بروز مفاخره می شود

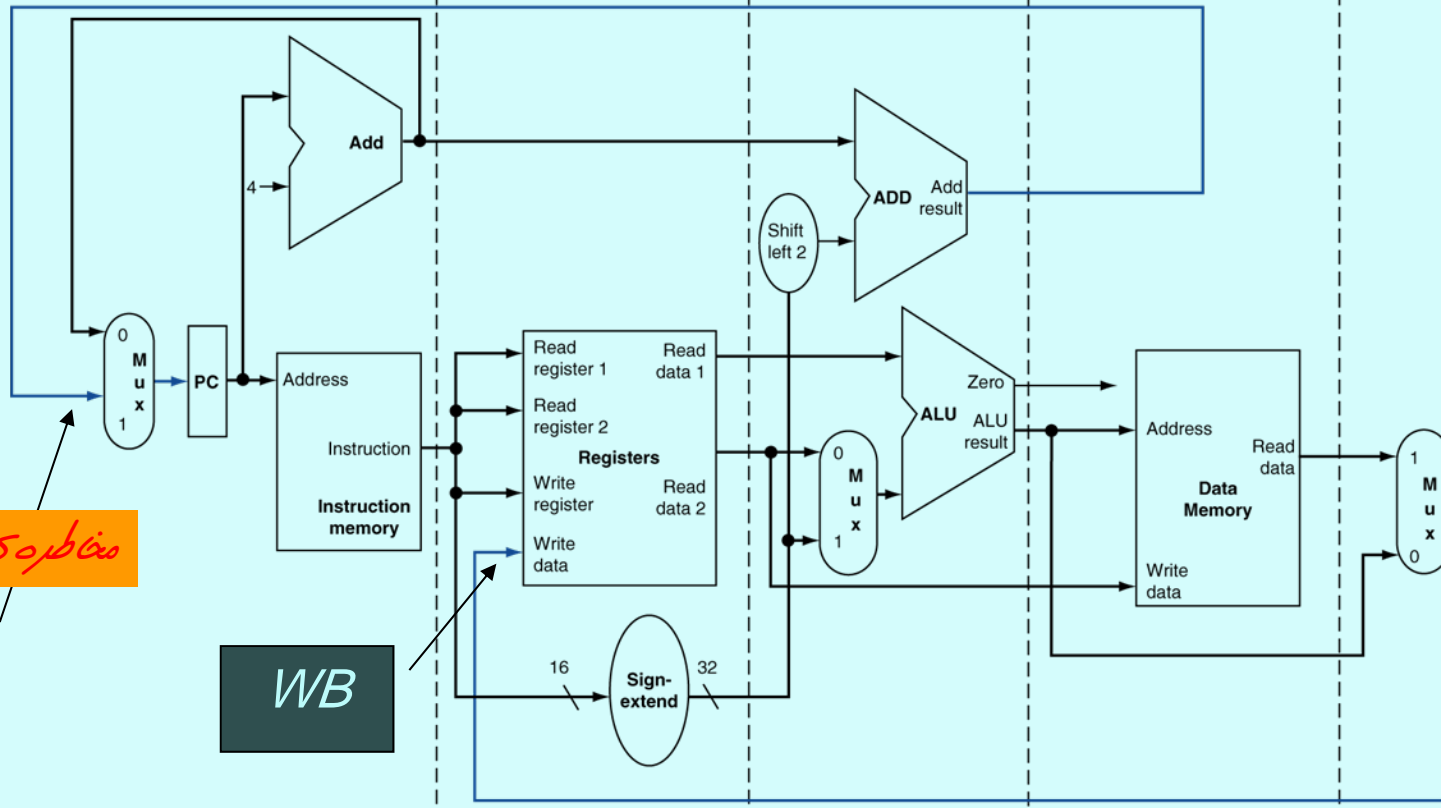
IF: Instruction fetch

ID: Instruction decode/
register file read

EX: Execute/
address calculation

MEM: Memory access

WB: Write back

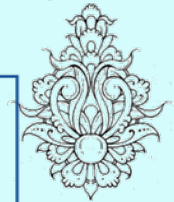


مفاخره‌ی کشوری

MEM

WB

مفاخره‌ی داده‌ای



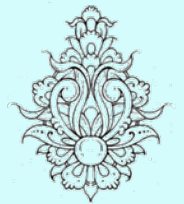
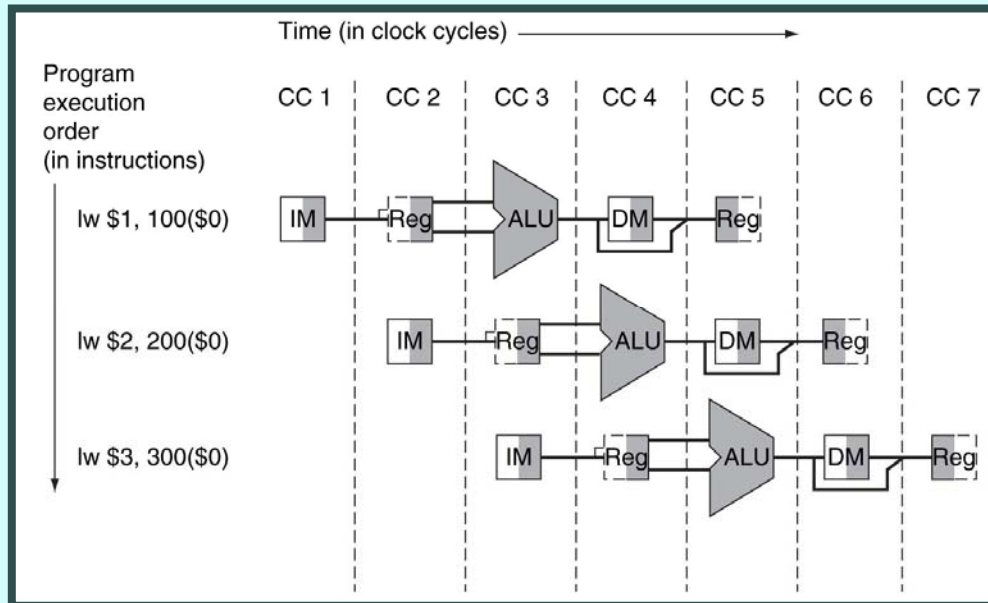
تراشگاه
سپهر
بهشتی

گذرداده‌ی مجهز به فلوله (ادامه...)

- چنانچه ملاحظه شد، مخاطره‌ی خط لوله دو سرچشمه دارد، که در هر دو جریان داده از سمت راست به چپ می‌باشد:

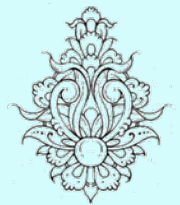
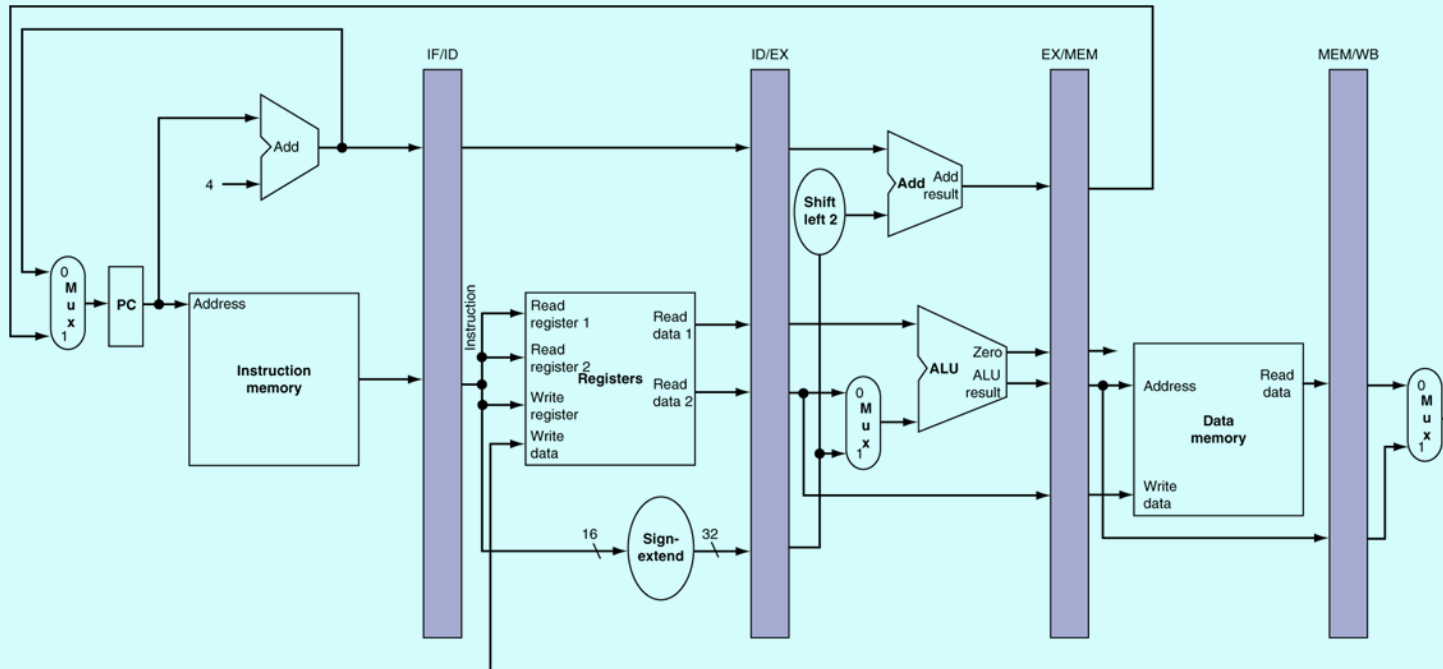
– مرحله‌ی WB (مخاطره‌ی داده‌ای)

– انتخاب آدرس بعدی (مخاطره‌ی کنترلی)

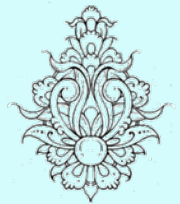
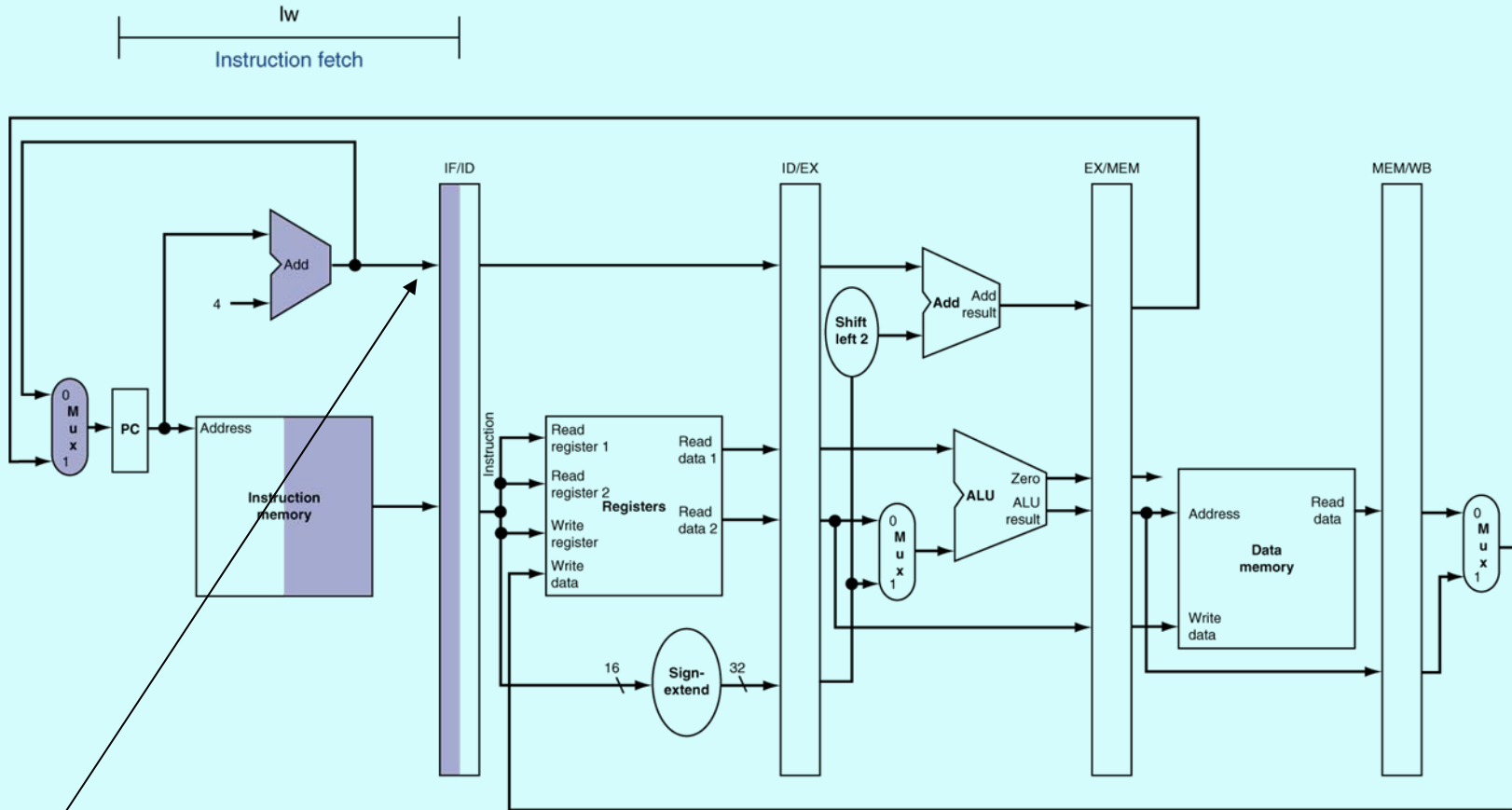


ثبات‌های خط لوله

- برای این که داده‌ی تولید شده در هر گام حفظ شود، می‌باید بین گام‌های مختلف خط لوله از ثبات استفاده شود.

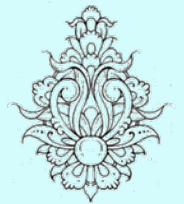
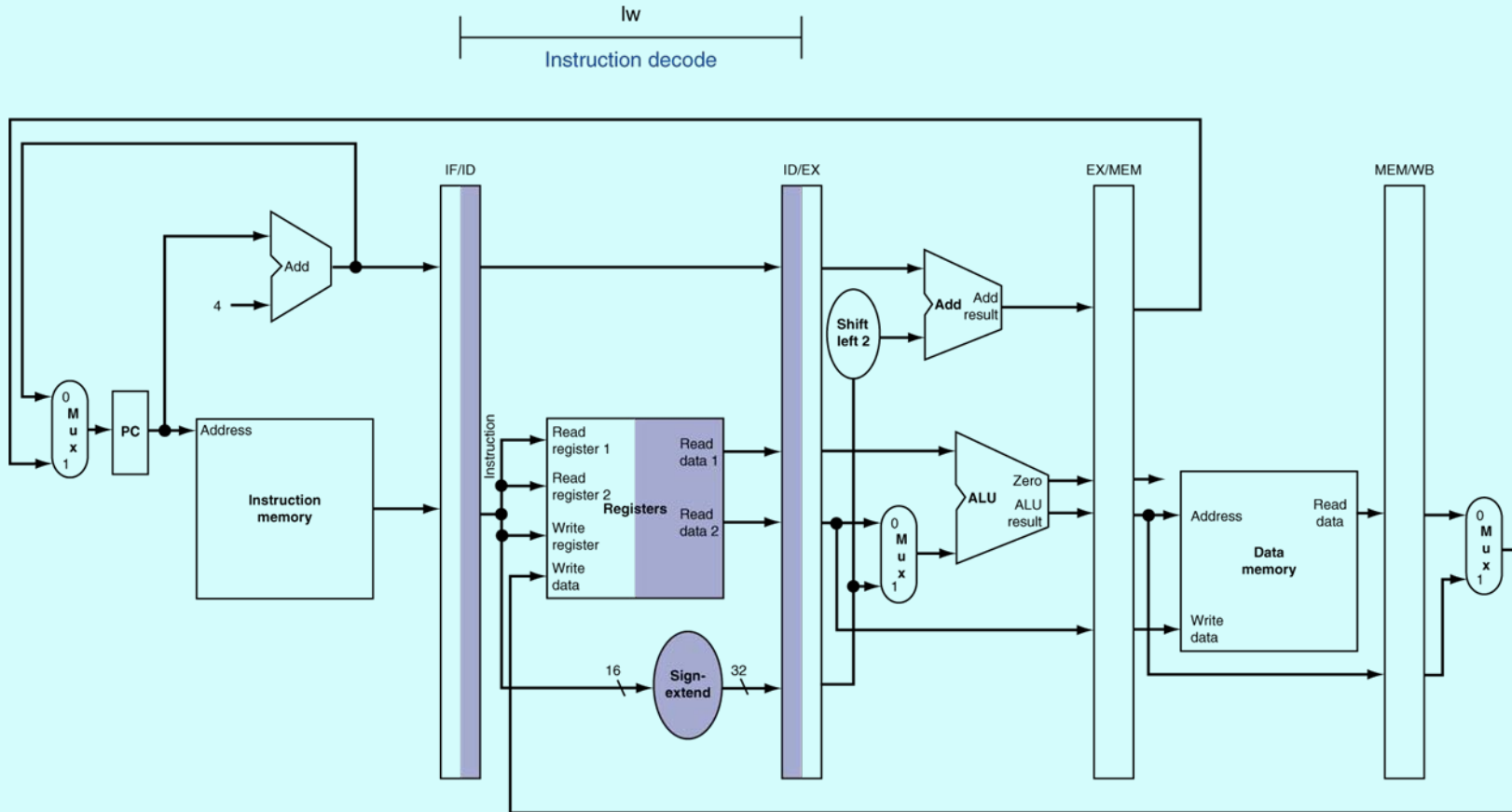


دستورات نوشتن و خواندن حافظه IF

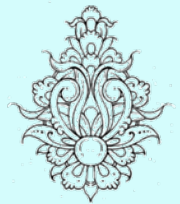
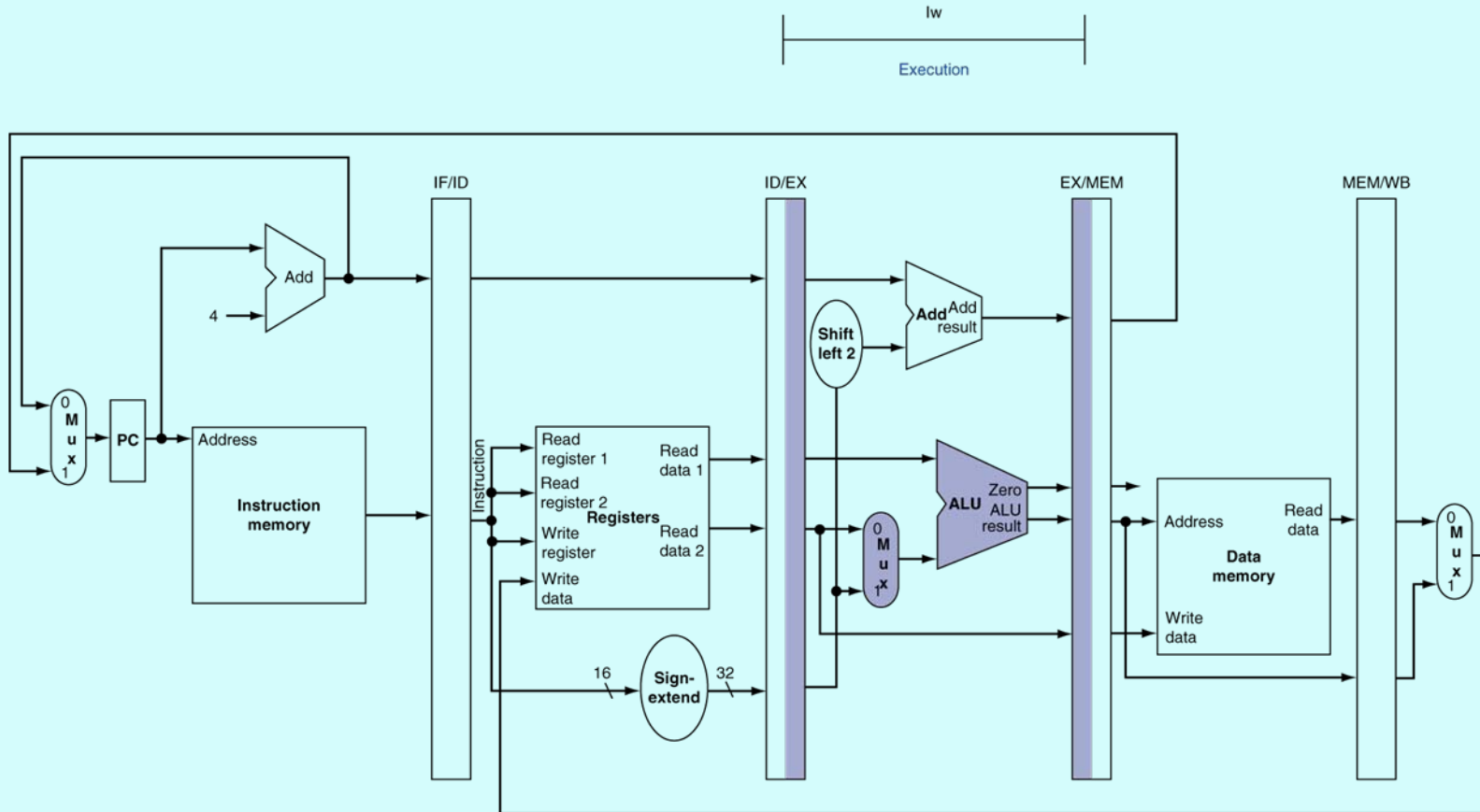


در صورتی که دستور بعدی پرش شرطی باشد، به این آدرس نیز خواهیم داشت

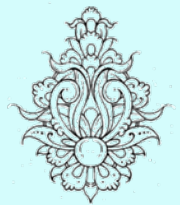
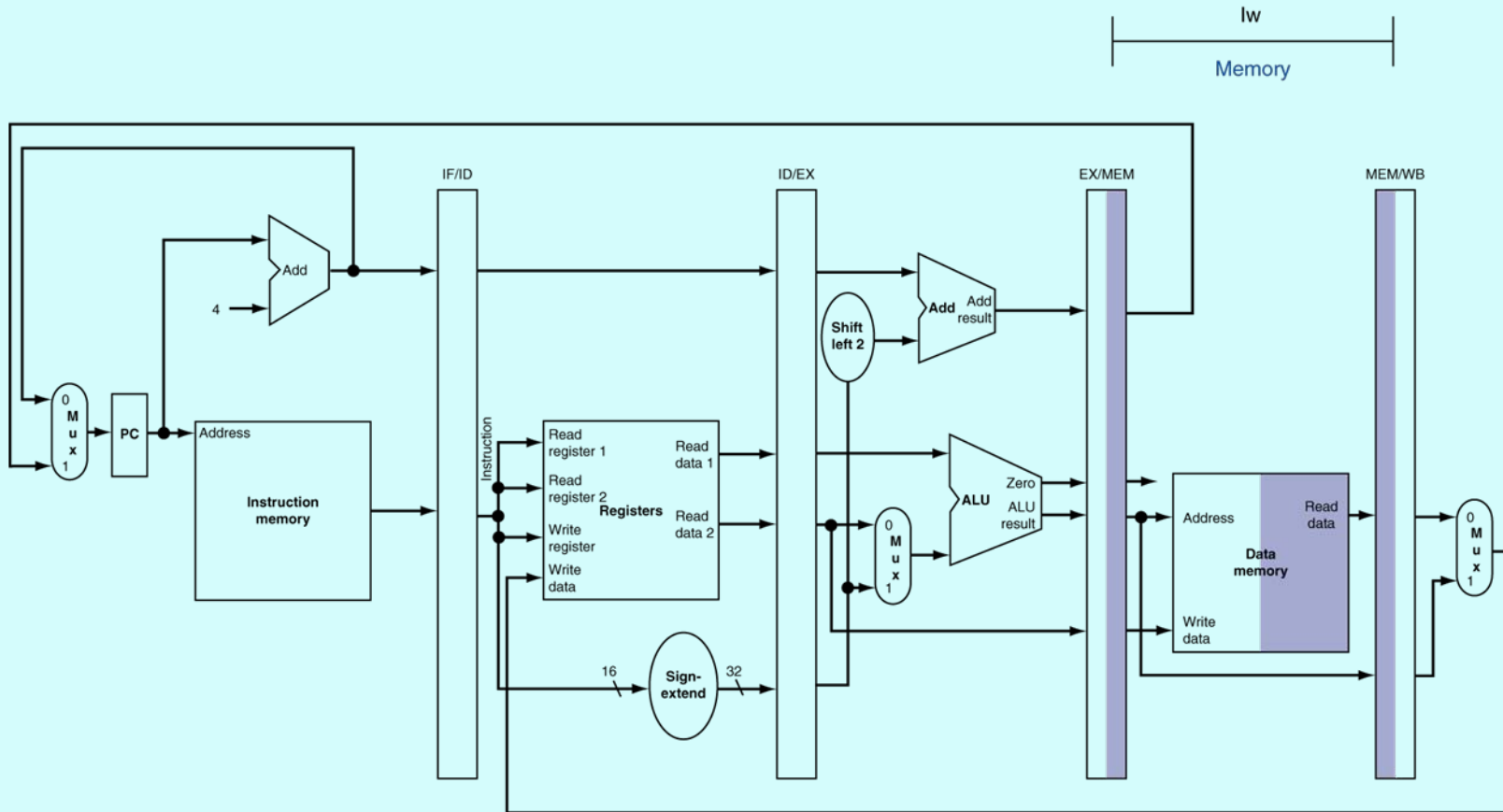
دستورات نوشتن و خواندن حافظه ID



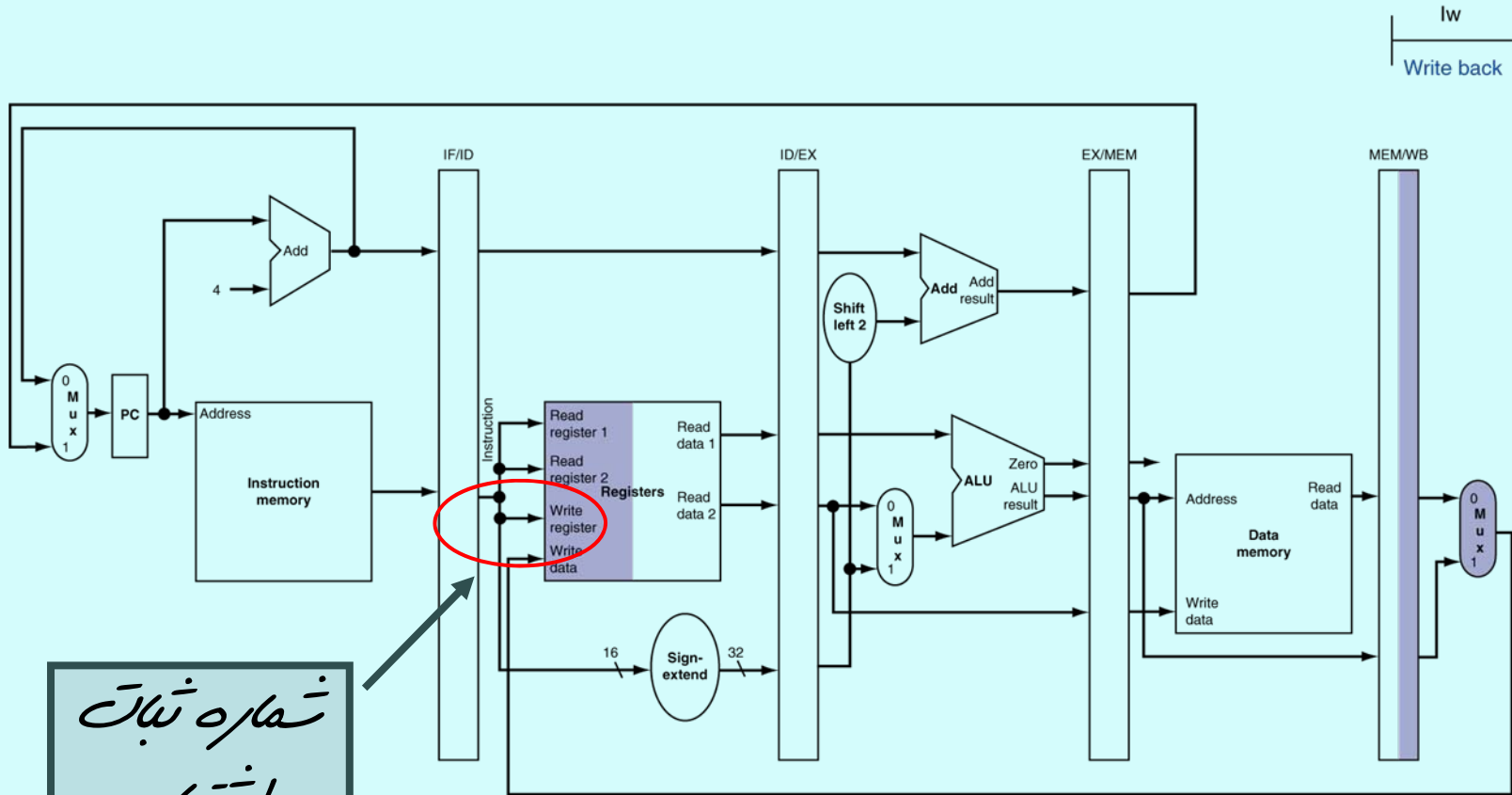
دستور خواندن حافظه EX



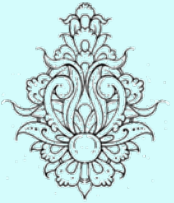
MEM دستور خواندن حافظه



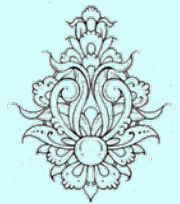
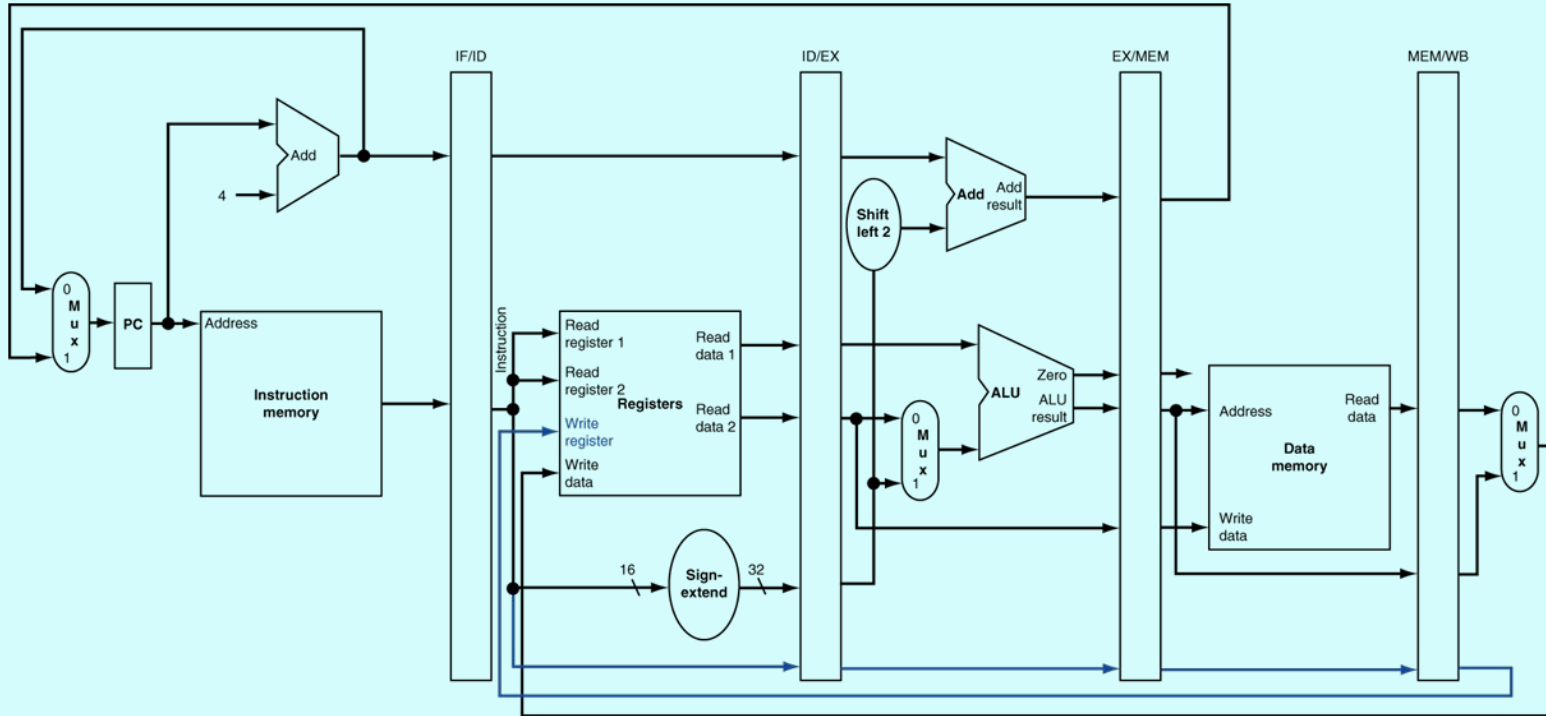
دستور خواندن حافظه WB



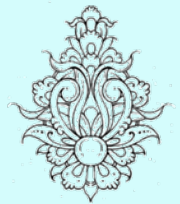
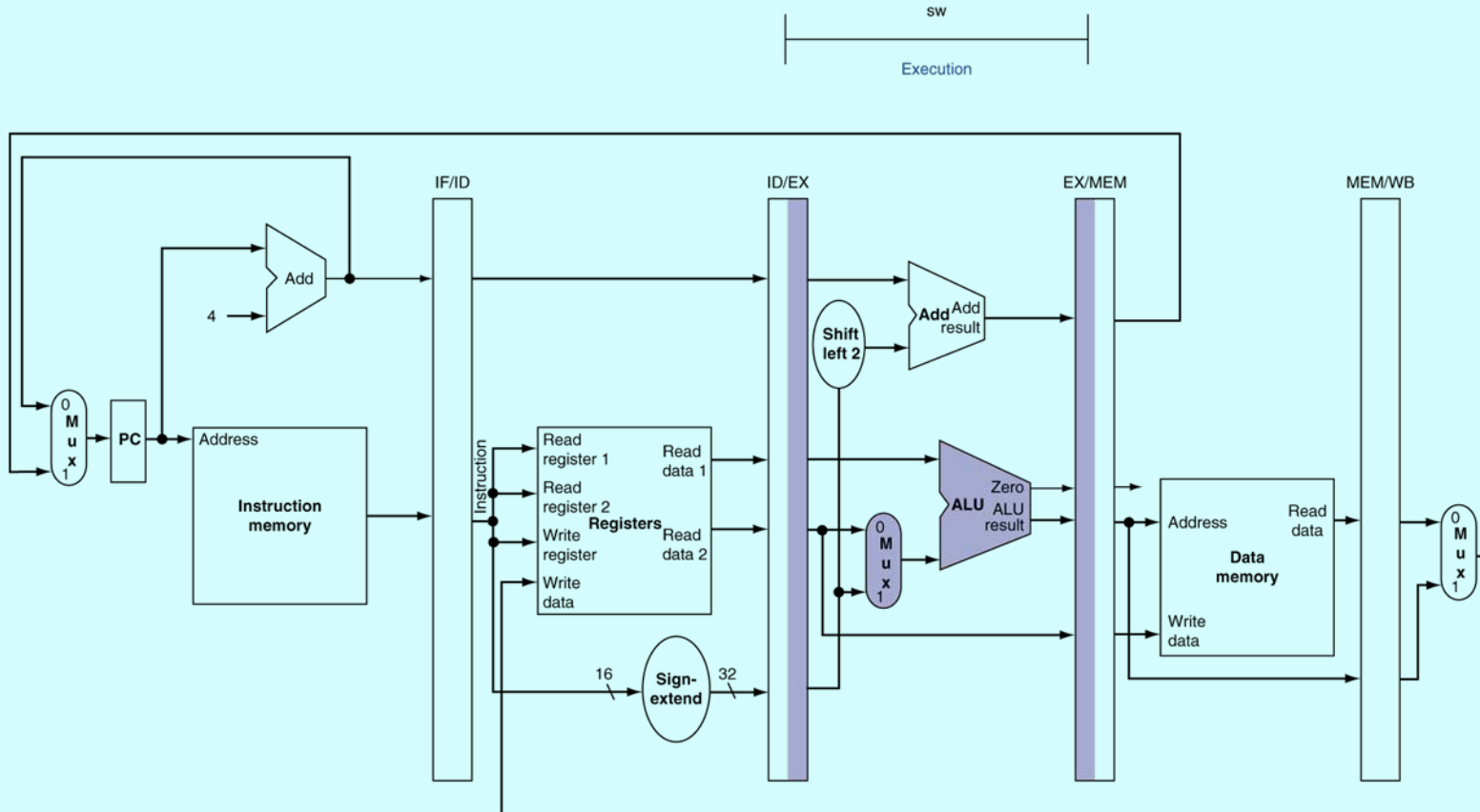
شماره ثبت
استبانه



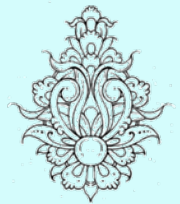
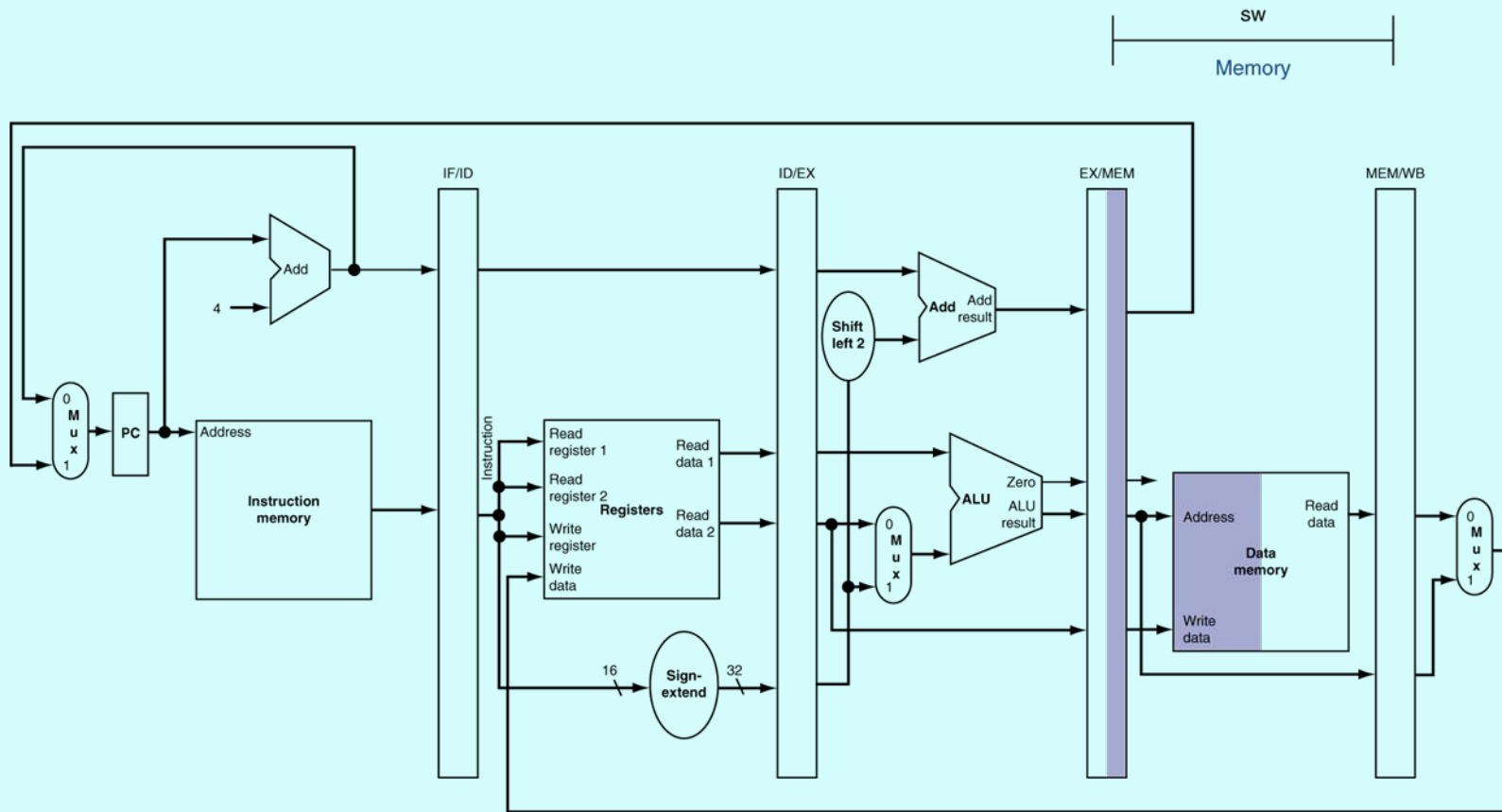
داده‌گذر اصلاح شده برای دستور خواندن



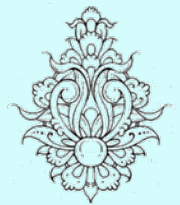
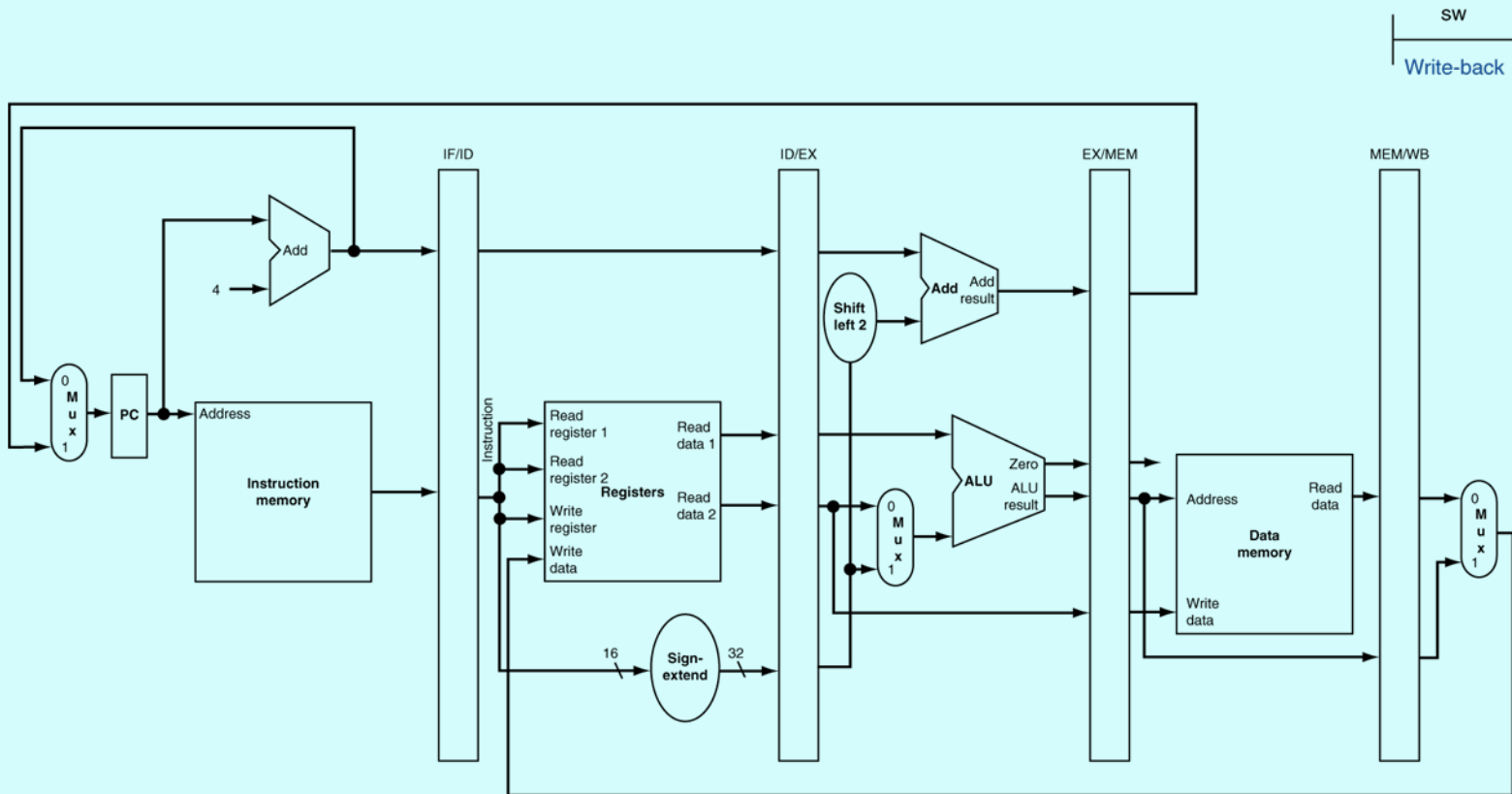
دستور نوشتن در حافظه EX



MEM دستور نوشتن در حافظه

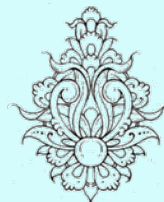
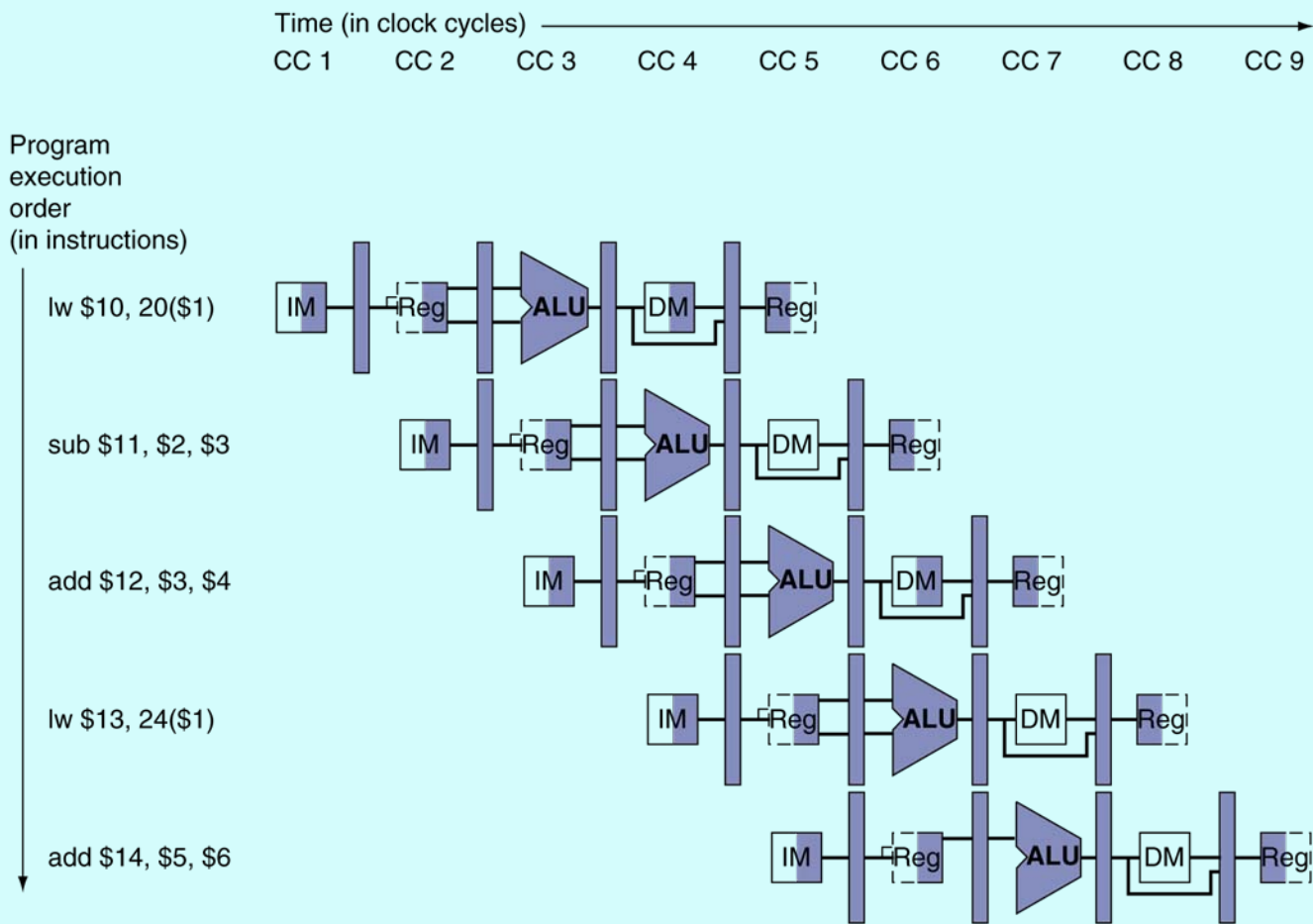


WB دستور نوشتن در حافظه WB



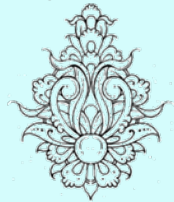
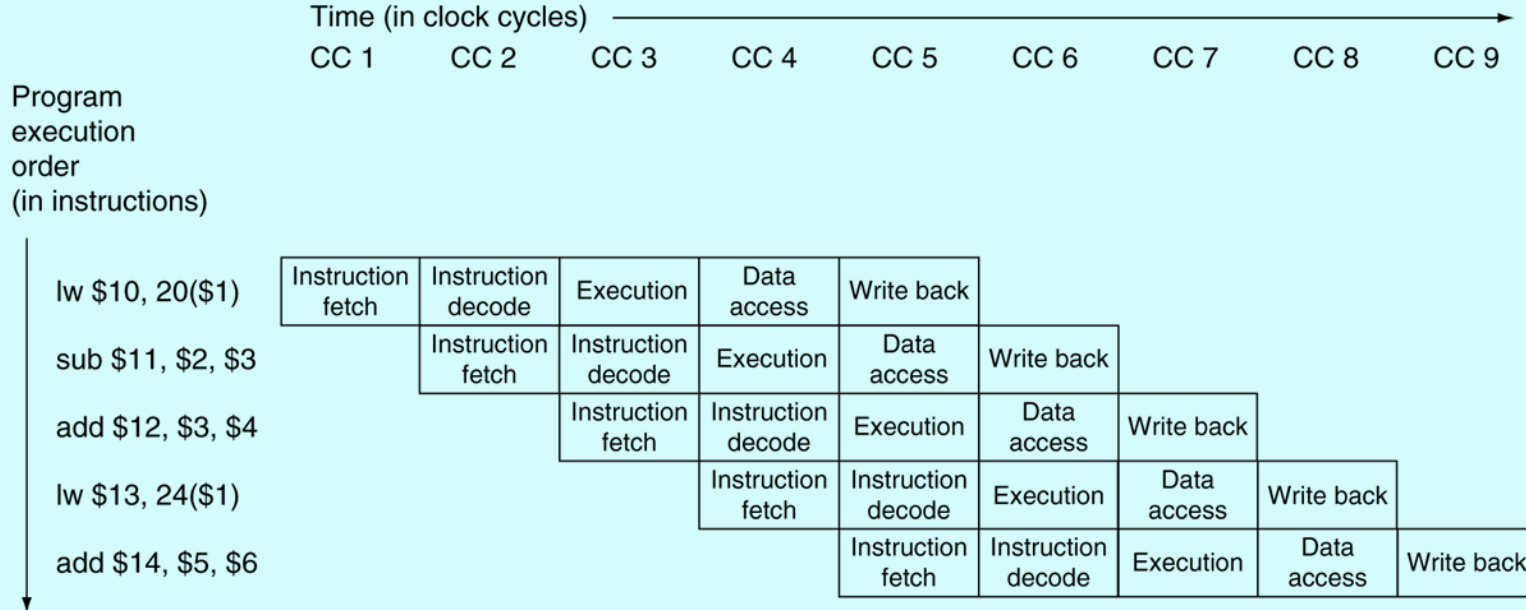
نمودار خط لوله به صورت چندسیکلی

- در این شیوه به کارگیری منابع نشان داده شده است



نمودار خط لوله به صورت چندسیکلی (ادامه...)

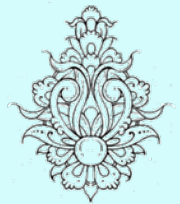
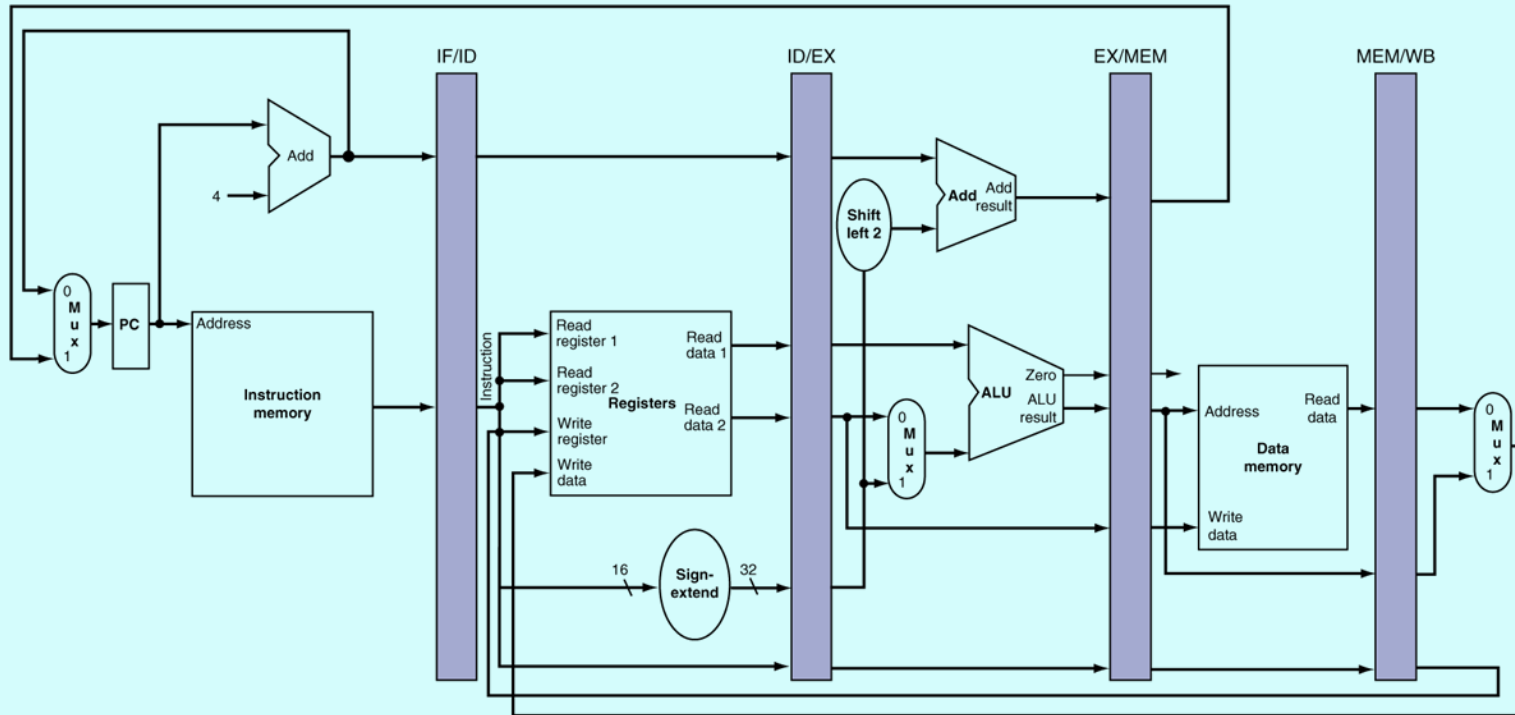
• شیوهی متعارف



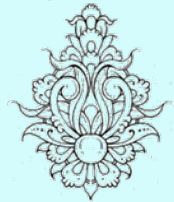
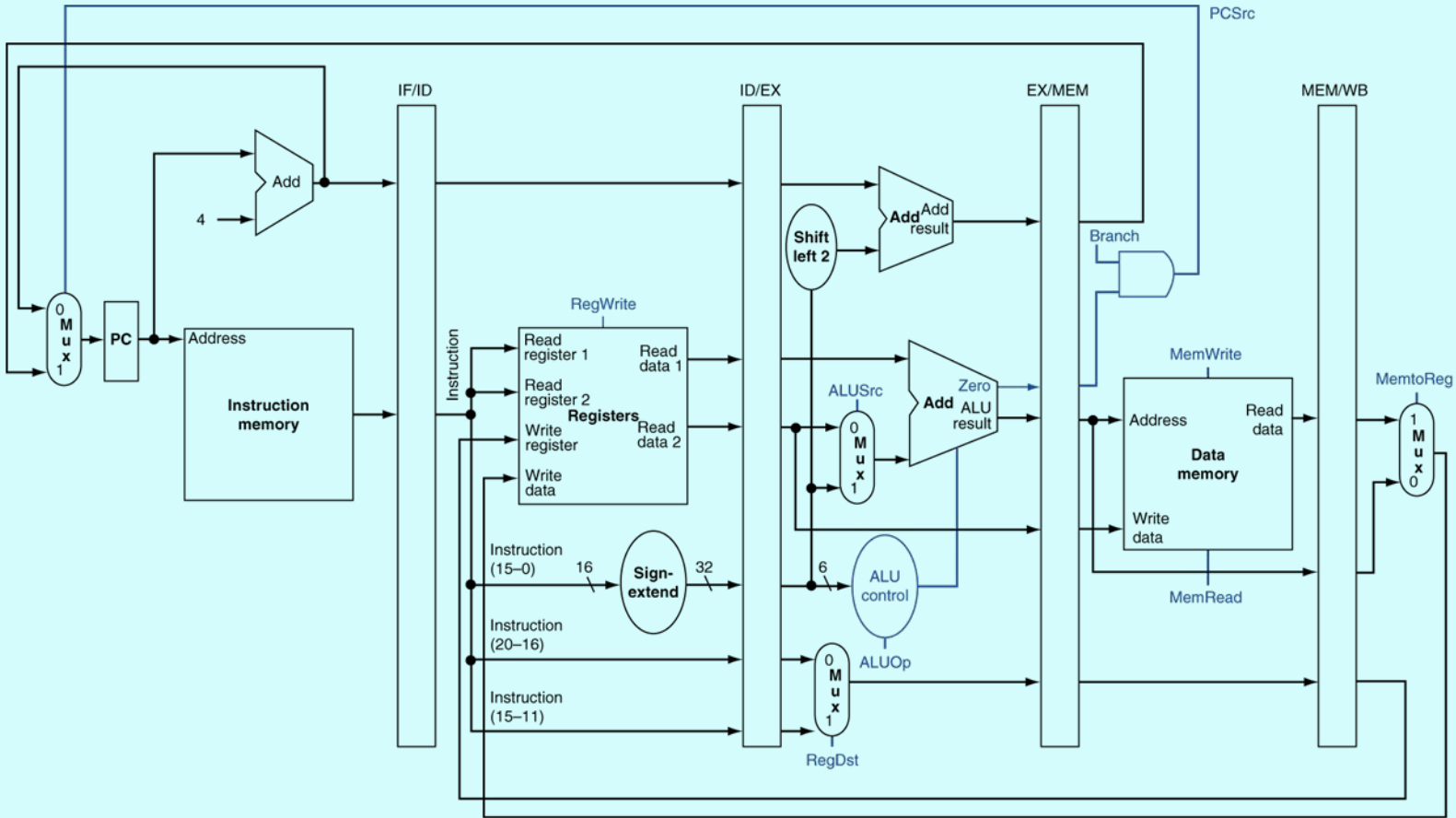
نمودار خط لوله به صورت تک سیگلی

• حالت خط لوله در یک سیگل

add \$14, \$5, \$6	lw \$13, 24 (\$1)	add \$12, \$3, \$4	sub \$11, \$2, \$3	lw \$10, 20(\$1)
Instruction fetch	Instruction decode	Execution	Memory	Write-back



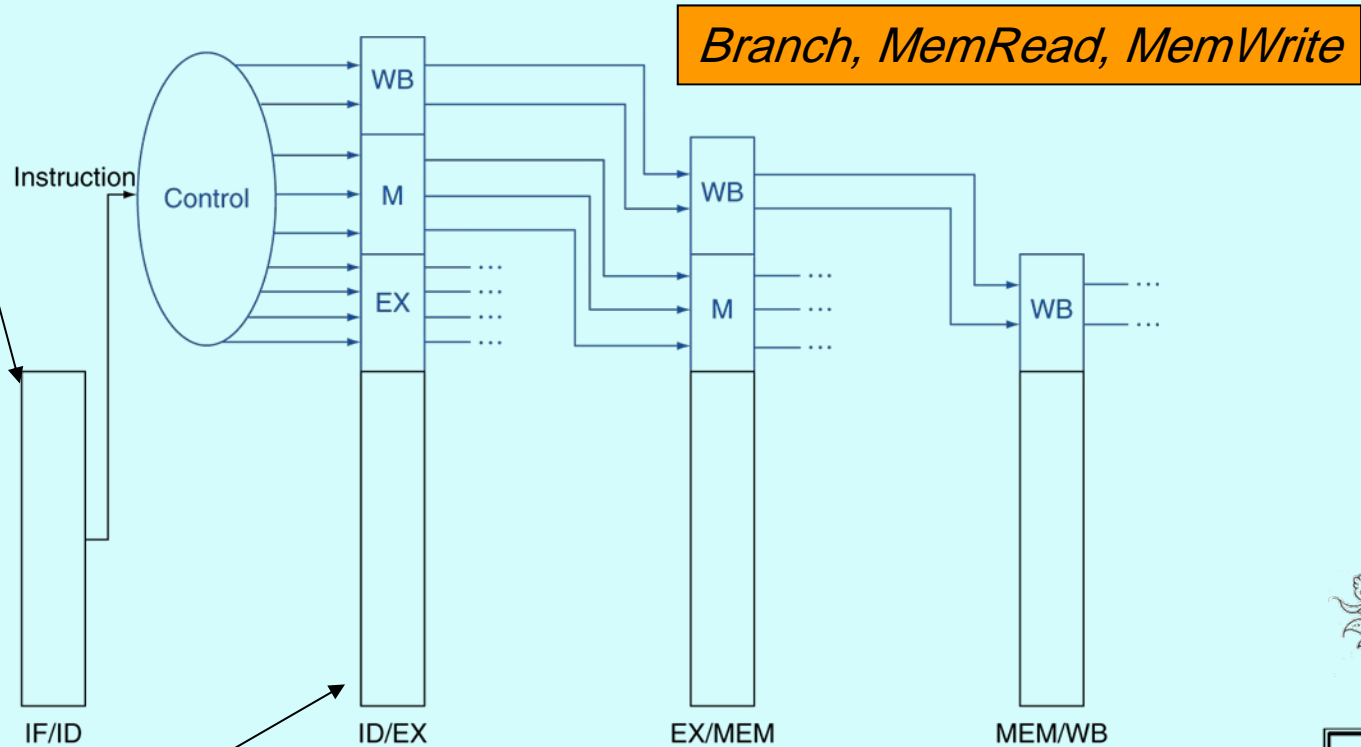
واحد کنترل خط لوله



تراشگاه
سپید
بهشتی

واحد کنترل خط لوله (ادامه...)

در این بخش من باید دستور العمل بعدی و آنتی شود،
کاری که برای همگی دستورها به یک شیوه خواهد بود



همانند تمام پیشین، نیاز
به سیگنال کنترلی خاصی
نیست

