معماری کامپیوتر (۱۱۳۵-۱۱-۱۳)

جلسه‌ی هجدهم

دانشگاه شهید بهشتی

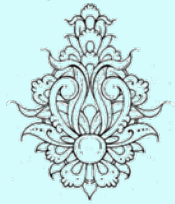
دانشکده‌ی مهندسی برق و کامپیوتر

بهار ۱۳۹۱

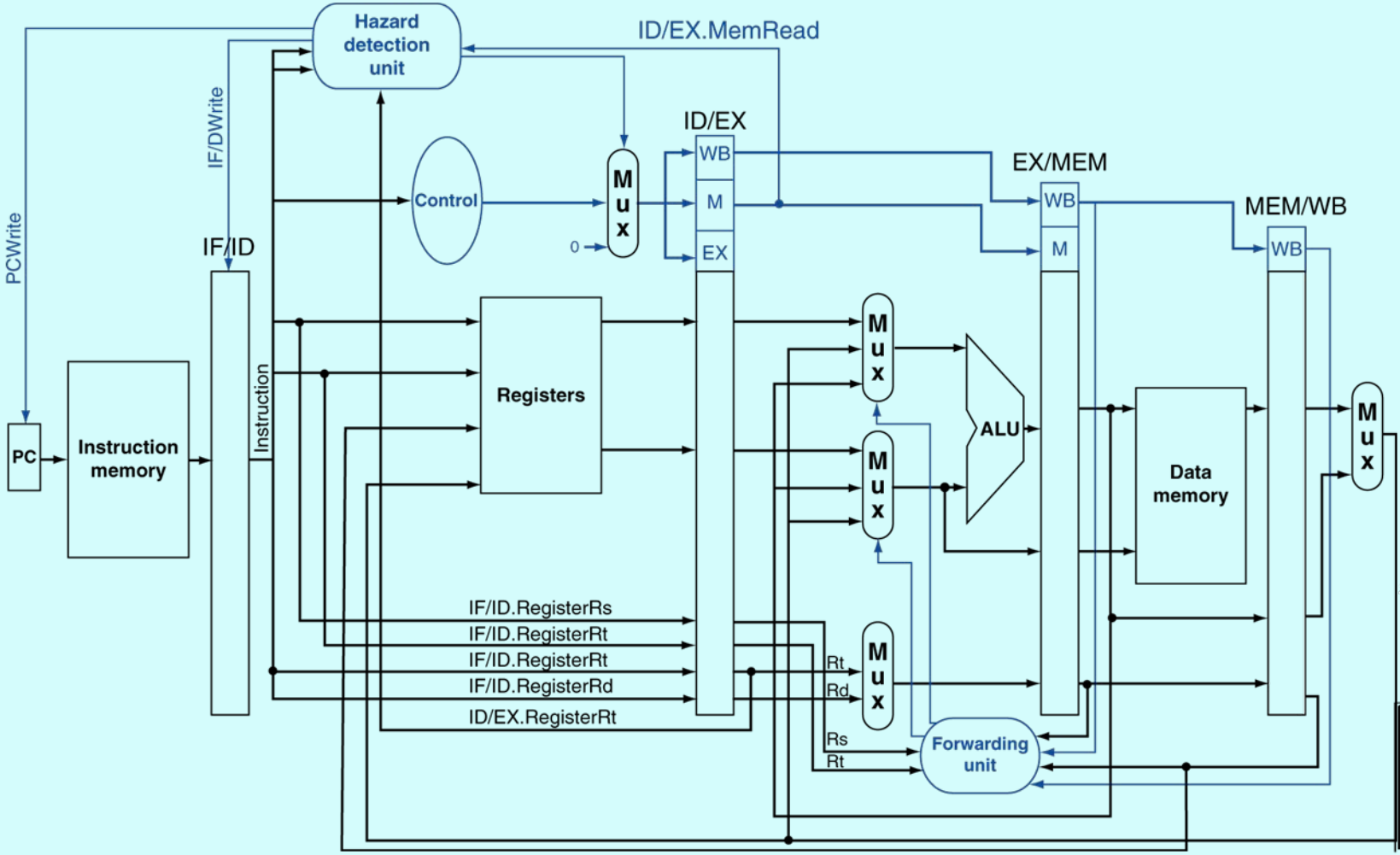
احمد محمودی ازناوه

فهرست مطالب

- مخاطرات کنترلی
  - پیش‌بینی نتیجه‌ی پرش
- استثنائات

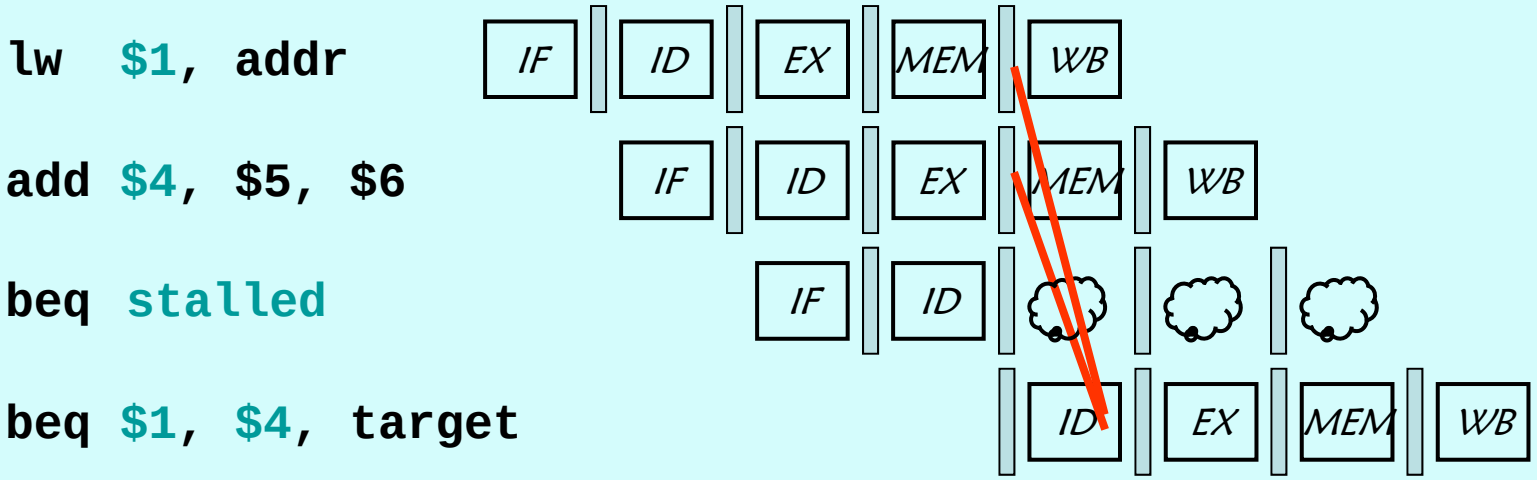


داده‌گذر همراه با مدار تشخیص مخاطره



مخاطره‌ی داده در پرش شرطی (ادامه...)

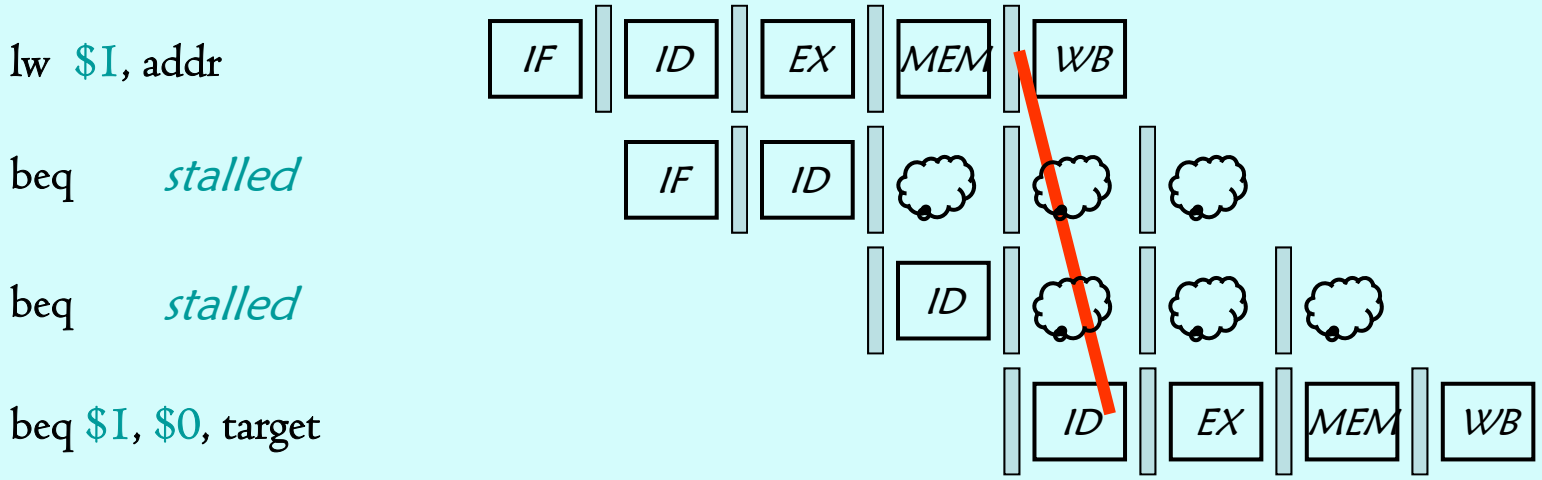
- اگر یکی از ثبات‌های مقایسه در حال بارگذاری از حافظه باشد:



به یک حباب احتیاج خواهیم داشت

مخاطره‌ی داده در پرش شرطی (ادامه...)

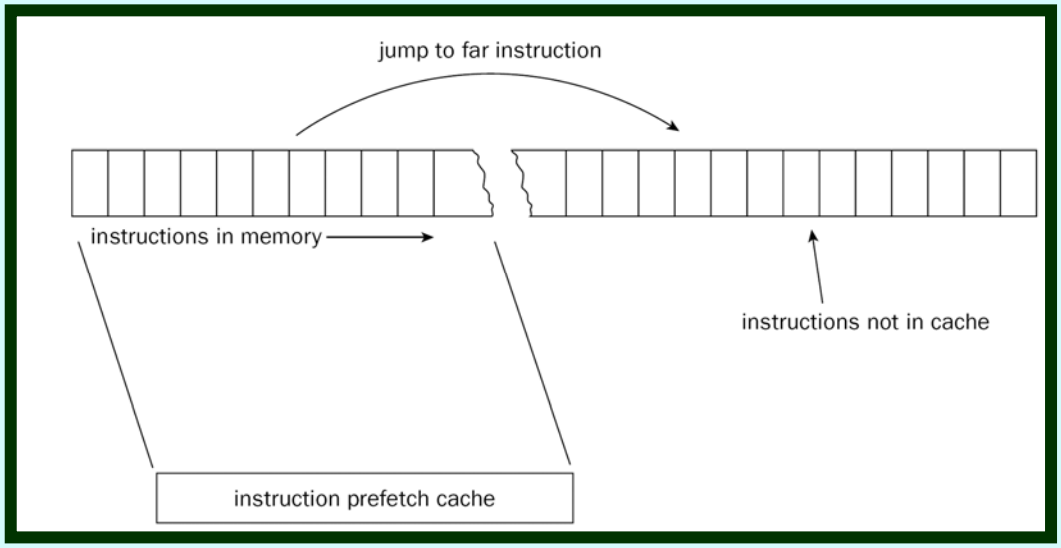
- اگر یکی از ثبات‌های مقایسه در حال بارگذاری از حافظه و دقیقا پیش از دستور پرش باشد:



به دو حباب احتیاج خواهیم داشت

بهینه‌سازی دستورات انشعاب

- دستورات انشعاب، به شدت در کارایی سیستم مؤثر هستند.
- در اکثر پردازنده‌های امروزی برای افزایش کارایی دستورات پیش‌واکشی می‌شود.
- پرش‌های غیرشرطی
  - باعث کاهش کارایی برنامه می‌شوند.



بهینه‌سازی دستورات انشعاب (ادامه...)

- **پرش‌های شرطی**
  - برای تشخیص توالی دستوراتی که هنوز نتیجه‌ی شرط مشخص نیست، از الگوریتم‌های پیش‌بینی استفاده می‌شود.
- **پیش‌بینی نتیجه‌ی شرط به صورت‌های زیر انجام می‌شود:**
  - در پرش رو به عقب، فرض بر آن است که شرط برقرار است.
  - در پرش رو به جلو، فرض بر آنست که شرط برقرار نیست.
  - دستورات پرشی که در اجرای قبلی، انجام شده‌اند، باز هم اجرا خواهند شد.
    - این فرض بر دو فرض قبل غلبه دارد.
    - BTB، برای ره‌گیری آخرین نتیجه مورد استفاده قرار می‌گیرد.

```
movl $100, $ecx
loop1:
addl %cx, %eax
decl %ecx
jns loop1
```

Branch Target Buffer

پیش‌بینی پویا

- در خط لوله‌های عمیق‌تر و در «سوپراسکالرها» زیان ناشی از پرش قابل تحمل نیست.
- در چنین حالاتی از پیش‌بینی پویا استفاده می‌شود.

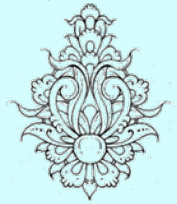
Branch prediction buffer

branch history table

– میان‌گیر پیش‌بینی خطا (جدول تاریخچه)

– نتیجه‌ی آخرین پرش را ذخیره می‌کند.

– در دفعات بعدی مطابق با حالت پیش عمل خواهد کرد.

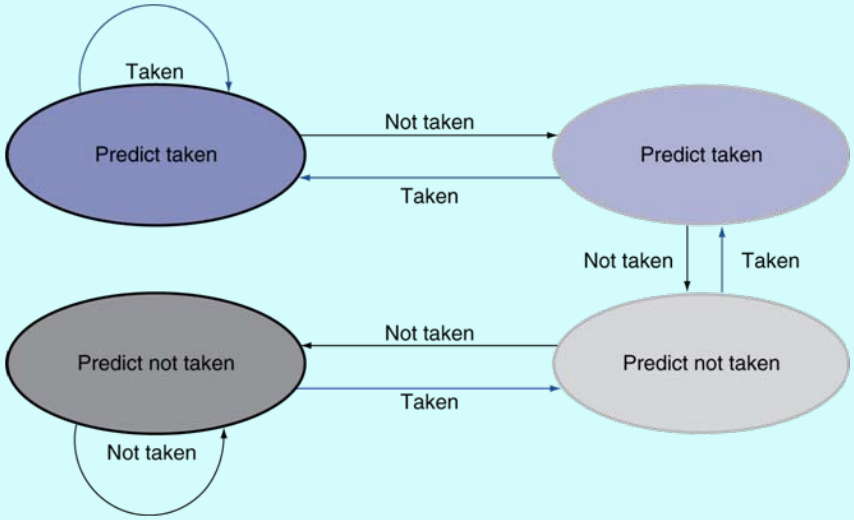


کاستی‌های پیش‌بینی‌کننده‌ی یک بیتی

- **حلقه‌های تودرتو: دو بار نتیجه‌ی پیش‌بینی اشتباه خواهد بود.**

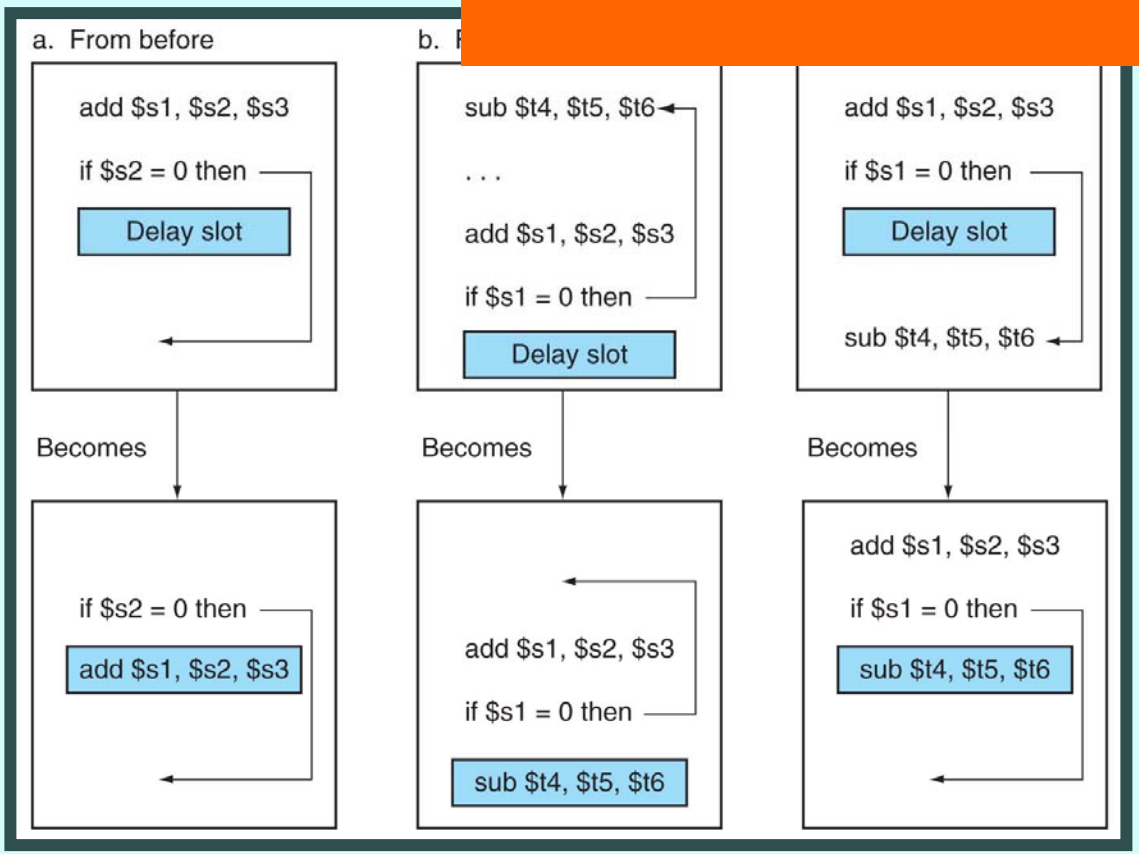
```
outer: …
       …
inner: …
       …
       beq …, …, inner
       …
       beq …, …, outer
```

یکی از راه‌های متداول اختصای دو بیت، برای نگهداری آخرین وضعیت است



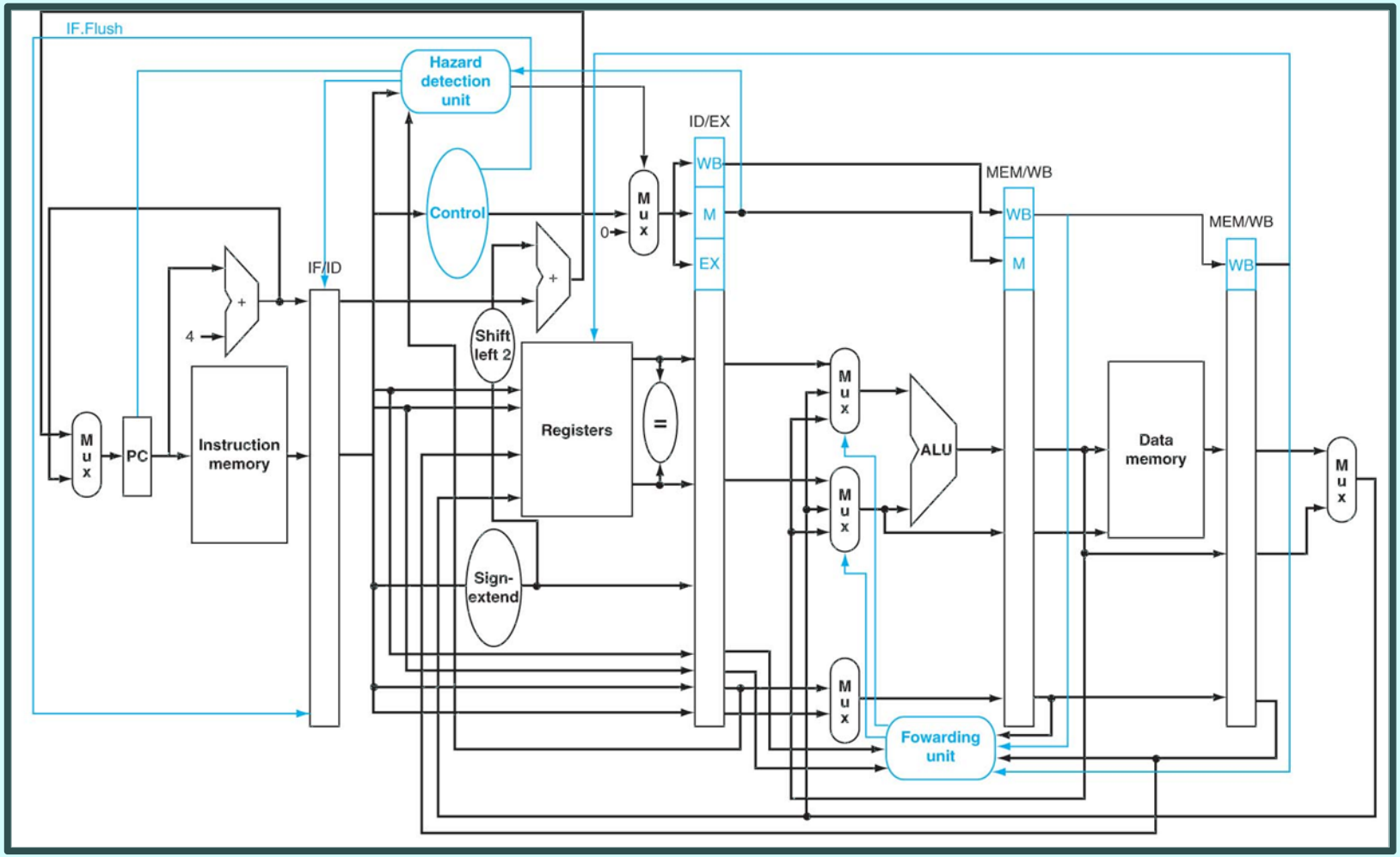
انشعاب تأخیر یافته

راه‌های دیگری نیز برای افزایش کارایی پیشنهاد شده است
یکی از آنها تاخیر پس از پرش شرطی است

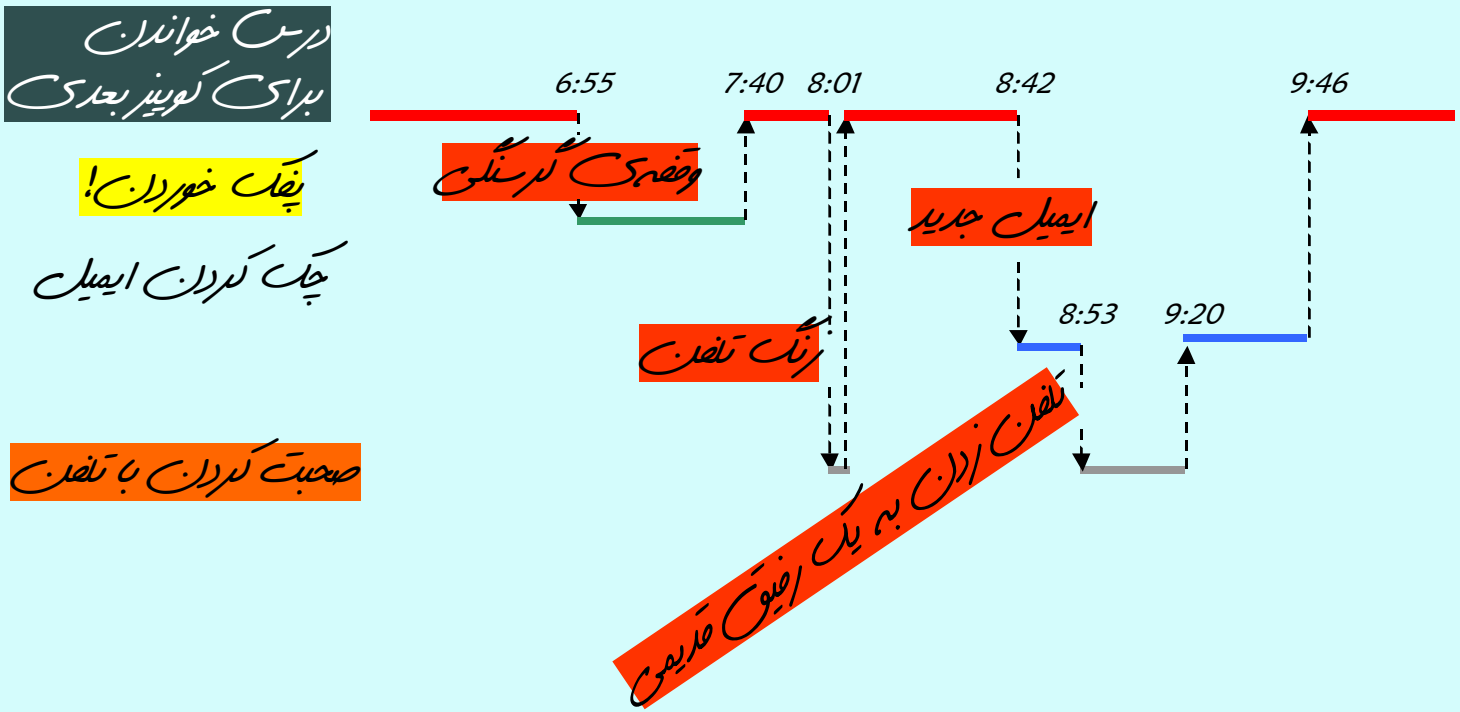


دیگری ذخیره کردن آدرس دستور پرش شرطی است

داده‌گذر نهایی



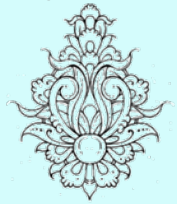
استثنائات


درس خواندن برای کوییز بعدی
پُف خوردن!
چک کردن ایمیل
صحبت کردن با تلفن
6:55
7:40
8:01
8:42
9:46
8:53
9:20
وضعیت گرسنگی
ایمیل جدید
زنگ تلفن
تلفن زدن به یک رفیق قدیمی

استثنائات (ادامه...)

"Unexpected" events

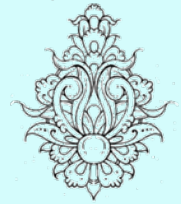
- حوادث پیش‌بینی نشده، می‌تواند روند عادی اجرای برنامه را تغییر دهد.
- در بسیاری از مراجع تمایزی بین وقفه و استثنا قائل نمی‌شوند، در برخی منابع وقفه را حالت کلی‌تر می‌دانند و استثنا را مربوط به عملکرد نادرست .
  - در x86 از واژه‌ی وقفه استفاده شده است.
- در MIPS، از واژه‌ی استثنا (برای هر نوع حادثه با منشأ داخلی و خارجی) استفاده می‌شود. وقفه شامل حوادث با منشأ خارجی می‌باشند.



نمونه‌ای از استثنائات

نوع حادثه	منشأ	واژه‌ی رایج در MIPS
درخواست واحد ورودی/خروجی	خارجی	وقفه
درخواست از سیستم‌عامل از طرف برنامه‌ی کاربر	داخلی	استثنا
رخداد سرریز	داخلی	استثنا
استفاده از دستورالعمل نامشخص	داخلی	استثنا
خرابی سخت‌افزار	هردو	استثنا یا وقفه

برخورد با استثنائات، بدون قربانی کردن کارایی کاری بسیار دشوار است.



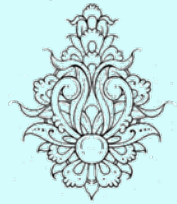
استثنائات (ادامه...)

- عدم توجه کافی به استثنائات در هنگام طراحی واحد کنترل می‌تواند موجب افت کارایی سیستم شود.
- در ادامه به طراحی دو نوع استثناء می‌پردازیم:
  - دستور ناشناخته
  - سرریز

EPC (exception programmer counter)

هنگام بروز استثناء، پردازنده آدرس دستور جاری را در ثبات وقفه ذخیره نموده و کنترل را به بخش خاص از سیستم عامل می‌سپارد

سیستم عامل، در این مواقع واکنشی از پیش تعیین شده انجام خواهد داد. سپس یا اجرای برنامه را خاتمه می‌دهد و یا ادامه‌ی برنامه را اجرا می‌کند.



استثنائات (ادامه...)

- سیستم عامل افزون بر دستوری که موجب رخداد استثنا شده است، می‌باید دلیل آن را نیز بداند.
  - در MIPS از یک ثبات وضعیت (status register) با نام ثبات سبب استفاده می‌شود.

Cause register

0 for undefined opcode, 1 for overflow

- *در صورت بروز وقفه، مقدار PC به 8000 00180 تغییر خواهد کرد. در واقع رویه‌ی رسیدگی‌کننده به وقفه در آن‌جا قرار دارد.*

Exception Handler

- *با بررسی ثبات سبب نوع استثنا و در نتیجه واکنش مناسب تشخیص داده می‌شود.*

استثنائات (ادامه...)

- راه دیگری نیز وجود دارد: **بردار وقفه**
  - که برداری از آدرس‌های رویه‌های رسیدگی‌کننده به وقفه (interrupt handler) می‌باشد. در واقع در این شیوه مشخصا رویه‌ی مورد نظر فراخوانی می‌شود.

Exception type	Exception vector address (in hex)
Undefined instruction	$8000\ 0000_{hex}$
Arithmetic overflow	$8000\ 0180_{hex}$

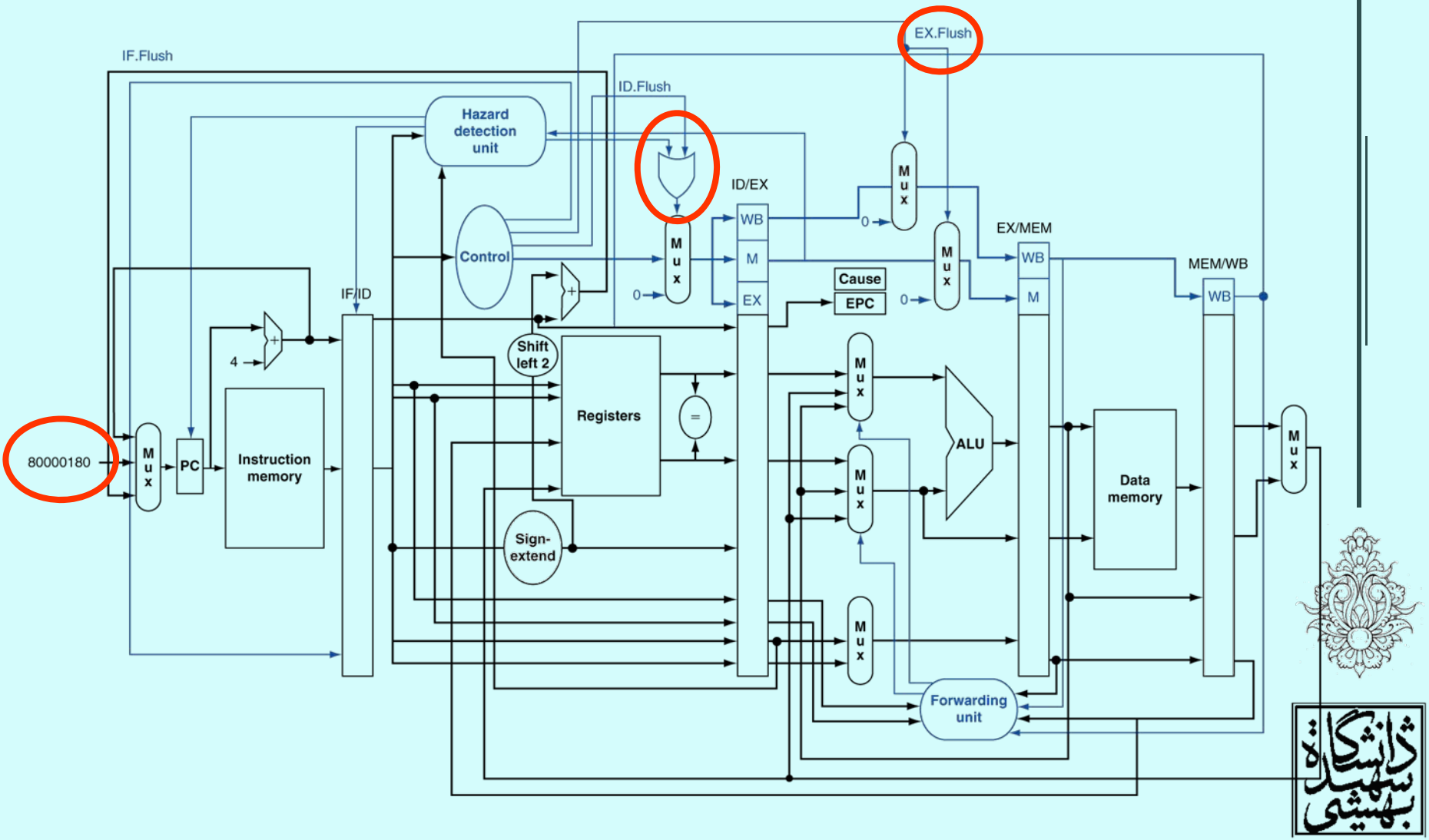
بروز استثنا در پردازنده‌ی مجهز به خط لوله

- در یک سیستم خط لوله یک استثنا، نوعی مخاطره‌ی کنترلی است.

```
add  $1,  $2,  $1
```

- اجرای دستورات پیش از دستور add می‌باید کامل شود.
- دستور add و دستورهای بعدی از خط لوله تخلیه شوند.
- ثبات‌های سبب و EPC مقدار دهی شوند.
  - کنترل به رسیدگی‌کننده به وقفه سپرده شود.

بروز استثنا در پردازنده‌ی مجهز به خط لوله (ادامه...)



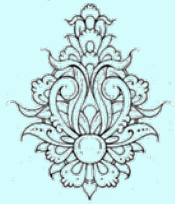
مثال

Exception on add

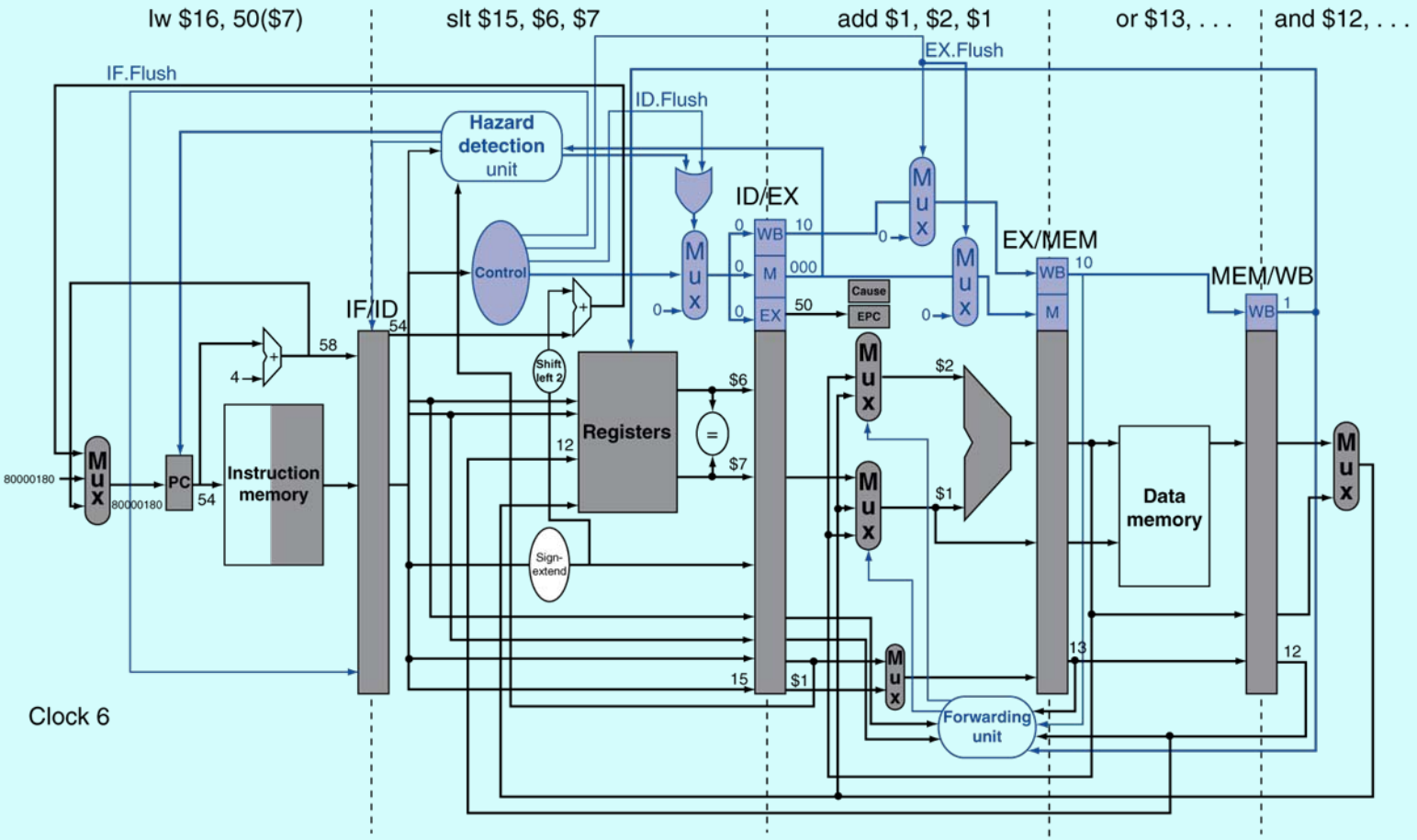
```
40 sub   $11, $2, $4
44 and   $12, $2, $5
48 or    $13, $2, $6
4C add   $1,  $2, $1
50 slt   $15, $6, $7
54 lw    $16, 50($7)
…
```

```
80000180      sw    $25, 1000($0)
80000184      sw    $26, 1004($0)
…
```

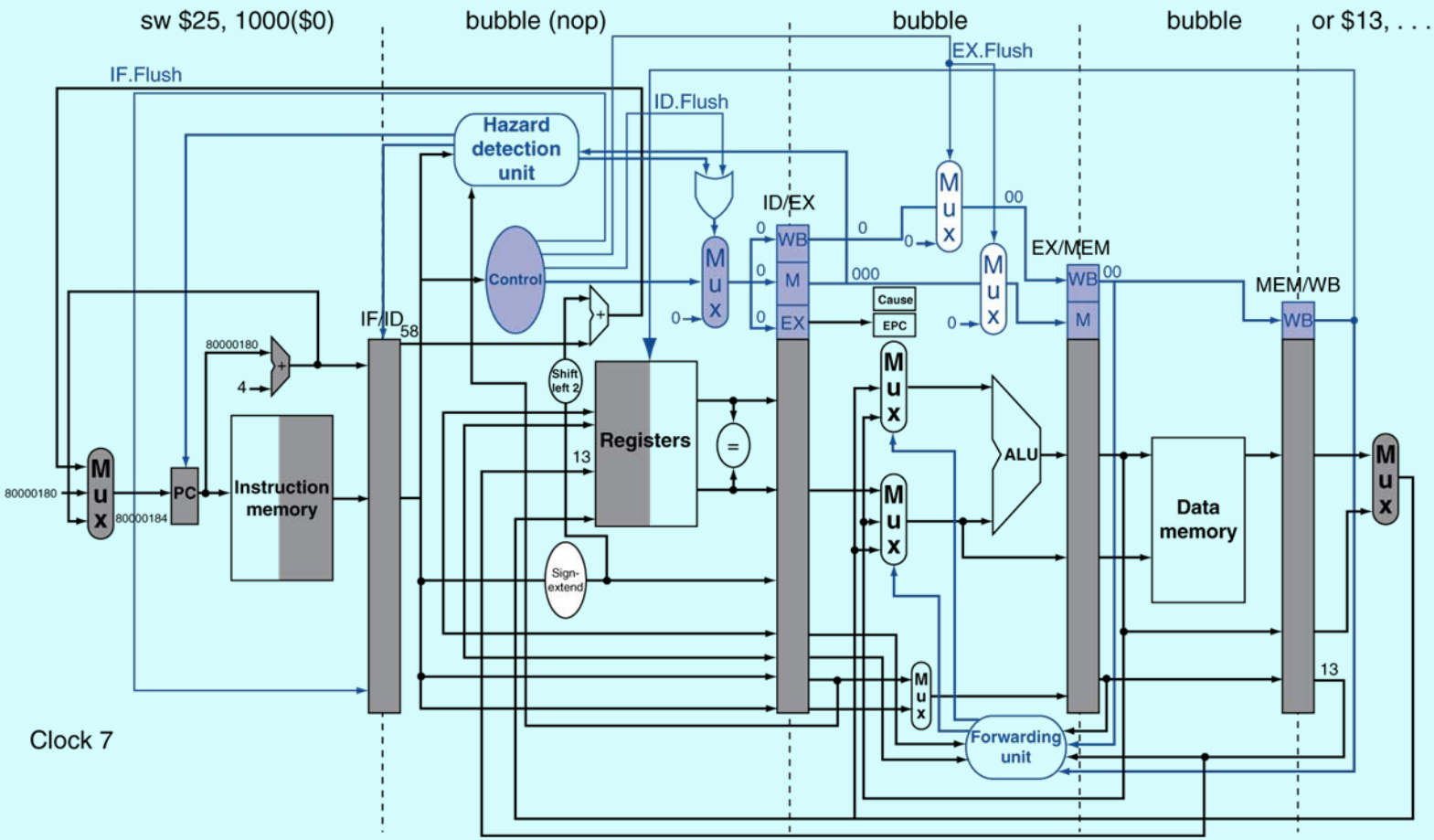
Handler



مثال


lw $16, 50($7)
slt $15, $6, $7
add $1, $2, $1
or $13, . . .
and $12, . . .
IF.Flush
ID.Flush
EX.Flush
Hazard detection unit
Control
ID/EX
EX/MEM
MEM/WB
IF/ID
WB
M
EX
Cause
EPC
Mux
PC
Instruction memory
Registers
Shift left 2
Sign-extend
Data memory
Forwarding unit
80000180
Clock 6

مثال


sw $25, 1000($0)
bubble (nop)
bubble
bubble
or $13, . . .
IF.Flush
ID.Flush
EX.Flush
Hazard detection unit
Control
ID/EX
EX/MEM
MEM/WB
IF/ID
WB
M
EX
Cause
EPC
Mux
PC
80000180
80000184
4
Instruction memory
Shift left 2
Registers
Sign-extend
ALU
Data memory
Forwarding unit
13
Clock 7

بروز همزمان چندین استثنا

- **در خط لوله چند دستورالعمل همزمان اجرا می‌شوند.**
  - بنابراین امکان بروز چند استثنا به صورت همزمان وجود دارد.
    - می‌توان به استثنایی که مربوط به دستورالعمل‌های جلوتر هستند، زودتر رسیدگی نمود.