

# سیستم عامل

## Operating Systems

---

جلسه اول

**مدرس: اسماعیل طغراعی**

[www.Teach.Toghraee.ir](http://www.Teach.Toghraee.ir)



نگاهی کلی به

---

سخت افزار کامپیوتر



## سیستم عامل و مدیریت سخت افزار:

- سیستم عامل از منابع سخت افزاری شامل یک یا چند پردازنده برای ارائه خدمات به کاربران استفاده می کند. همچنین حافظه ثانویه و I/O را از طرف کاربران مدیریت می کند.
- بنابراین برای بررسی سیستم عامل داشتن درک مناسبی از سخت افزار کامپیوتر ضروری است.



## اجزای اصلی هر کامپیوتر عبارتند از:

- **پردازنده**: کنترل عملیات و نیز اعمال پردازش داده ها را به عهده دارد.
- **حافظه اصلی**: ذخیره و نگهداری داده ها و برنامه ها را به عهده دارد. همچنین فرار و ناپایدار است.
- **مؤلفه I/O**: انتقال داده ها میان کامپیوتر و دنیای خارج.
- **گذرگاه سیستم**: ساختار هایی برای ارتباط بین پردازنده و حافظه اصلی و I/O

# نگاهی کلی به اجزاء کامپیوتر:

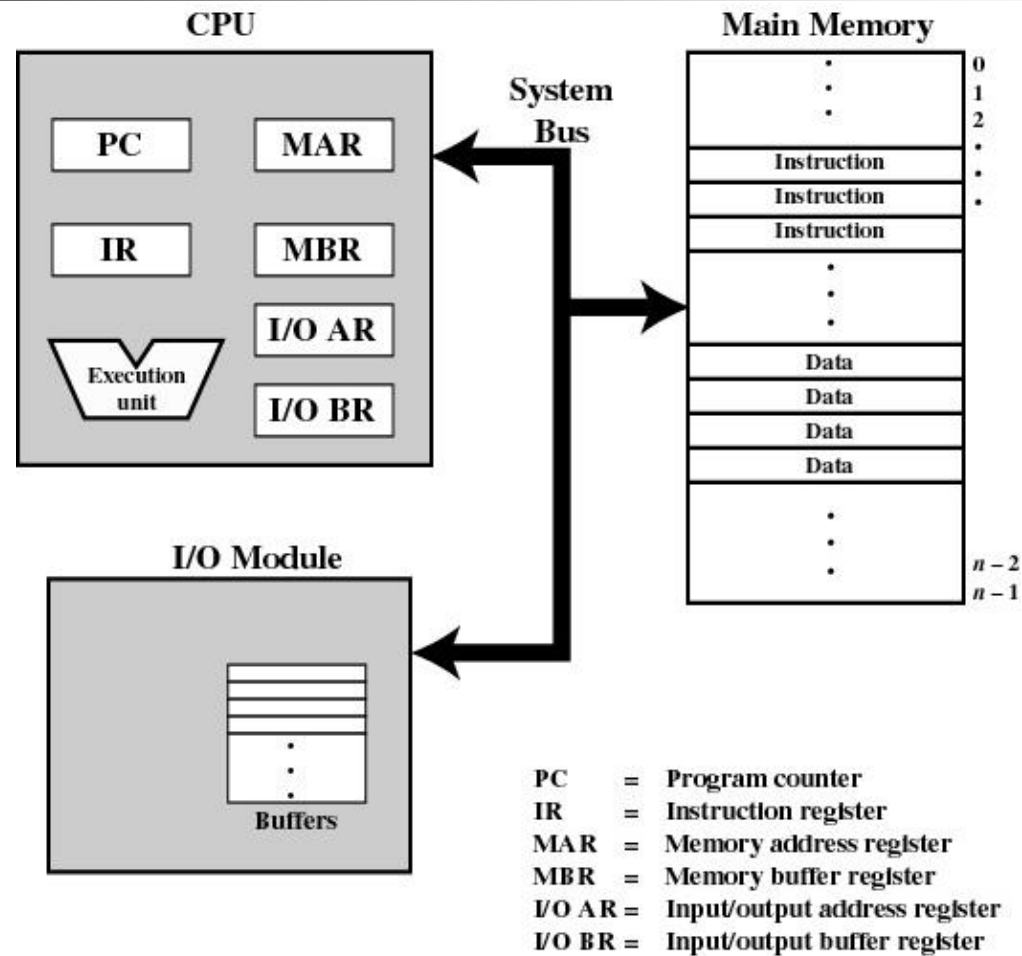


Figure 1.1 Computer Components: Top-Level View



## ارتباط با حافظه و ثبات های کمکی

- یکی از اعمال اصلی پردازنده ارتباط با حافظه اصلی است. برای تبادل اطلاعات بین CPU و MEMORY از دو ثبات به نام MAR و MBR استفاده میشود.
- MAR آدرس محل بعدی برای عملیات خواندن یا نوشتن را مشخص میکند.
- MBR حاوی داده هایی است که قرار است در حافظه خوانده یا نوشته شوند.
- متشابهاً برای تبادل داده ها بین CPU و I/O از دو ثبات I/OBR و I/OAR استفاده می شود.



## انواع ثبات های پردازنده:

- **ثبات های قابل رویت برای کامپیوتر :** ثبات هایی که برای کاربران قابل استفاده اند و این امکان را به برنامه نویس زبان ماشین یا اسمبلی میدهند تا با استفاده بهینه از این ثبات ها میزان مراجعه به حافظه اصلی را به حداقل برساند.
- **ثبات های کنترل وضعیت:** برای کنترل عملیات پردازنده



## ثبات های قابل رؤیت:

■ **ثبات های داده :** برای نگهداری داده ها و برای انتقال داده ها بین توابع (این ثبات ها همه منظوره اند)

■ **ثبات های آدرس:** حاوی آدرس داده ها و دستور العمل ها در حافظه (همه منظوره یا تک منظوره)

■ **ثبات شاخص :** در آدرس دهی شاخص استفاده میشود.

(آدرس شاخص + آدرس پایه = آدرس موثر)

■ **اشاره گر قطعه :** در آدرس دهی قطعه بندی حافظه به قطعه هایی تقسیم میشود. از این ثبات برای نگهداری آدرس پایه (آدرس شروع) استفاده میشود.

■ **اشاره گر پشته:** برای اشاره به بالای پشته به کار میرود.





## ثبات های کنترل وضعیت:

■ **PC** : شمارنده برنامه که برای نگهداری آدرس دستور العمل بعدی به کار میرود.

■ **IR** : ثبات دستورالعمل که برای ذخیره دستورالعمل فعلی که در حال اجراست به کار میرود.

■ **PSW(PROGRAM STATUS WORD)** : یک یا چند ثبات که حاوی اطلاعات وضعیت میباشد.



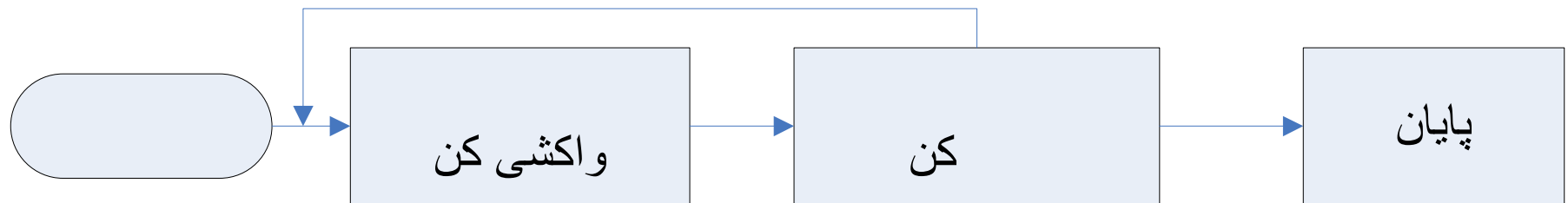
## چرخه دستور العمل:

---

- هر برنامه که اجرا میشود مجموعه ای از دستور العمل هاست که در حافظه ذخیره میشود.
- پردازش دستور العمل دو گام دارد:
  - واکشی دستور العمل از حافظه (چرخه واکشی)
  - اجرای دستور العمل واکشی شده (چرخه اجرا)

## نمودار چرخه دستورالعمل:

- پایان چرخه تنها در موارد زیر رخ میدهد:
  - خاموش شدن کامپیوتر
  - رخ دادن خطای غیر قابل جبران
  - رسیدن به فرمان توقف





## چگونگی واکنشی و اجرای دستورالعمل:

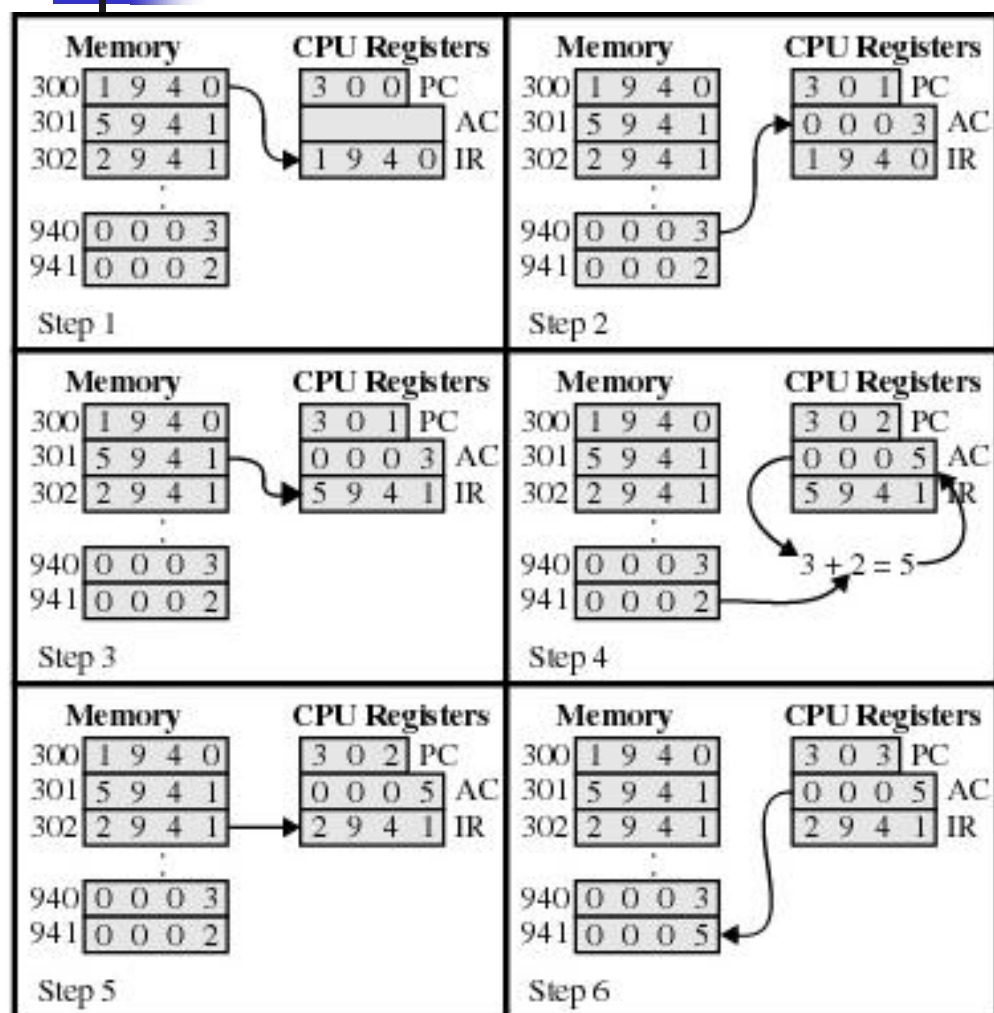
- در ابتدای هر چرخه، پردازنده دستورالعمل را از حافظه واکنشی میکند.
- شمارنده برنامه (ثبات PC)، آدرس دستورالعمل بعدی را که میخواید واکنشی شود در خود نگه می دارد.
- پس از واکنشی یک واحد به مقدار PC می افزاید تا دستور بعدی به ترتیب واکنشی شود.



## ثبات دستورالعمل (IR):

- دستورالعمل واکش شده در IR قرار میگیرد. این دستورالعمل به صورت دودویی میباشد و به صورت زیر است:
  - انتقال داده ها بین پردازنده و حافظه
  - انتقال داده ها بین پردازنده و ورودی خروجی
  - پردازش داده ها شامل عملیات حسابی یا منطقی روی داده ها
  - کنترل : ممکن است که یک دستورالعمل ترتیب اجرای برنامه را تغییر دهد این کار با تغییر در محتوای PC انجام میشود.

# مثالی برای فهم دقیق چرخه دستور العمل



■ فرضیات:

■ ۱: بار کردن به AC

■ ۲: ذخیره کردن مقدار AC در حافظه

■ ۵: اضافه کردن به AC



## دسترسی مستقیم به حافظه (DMA):

- همانطور که پردازنده میتواند عمل خواندن یا نوشتن را روی حافظه انجام دهد میتواند این عمل را روی یک مولفه I/O انجام دهد.
- گاهی مطلوبست داده ها بین I/O و حافظه مستقیما مبادله شوند تا از بار CPU کاسته شود و CPU بتواند برای انجام کارهای دیگر آزاد باشد. این فرایند دسترسی مستقیم به حافظه بدون دخالت پردازنده را DMA میگویند.



## وقفه ها (Interrupt)

---

- وقفه علامتی است که از طرف یک منبع خارجی به پردازنده داده میشود و موجب توقف برنامه فعلی میشود.
- وقفه ها برای افزایش کارایی پردازنده استفاده میشوند.
- وقفه ها به پردازنده اجازه میدهند تا در حین اجرای عملیات I/O به اجرای دستورالعمل دیگری بپردازد.

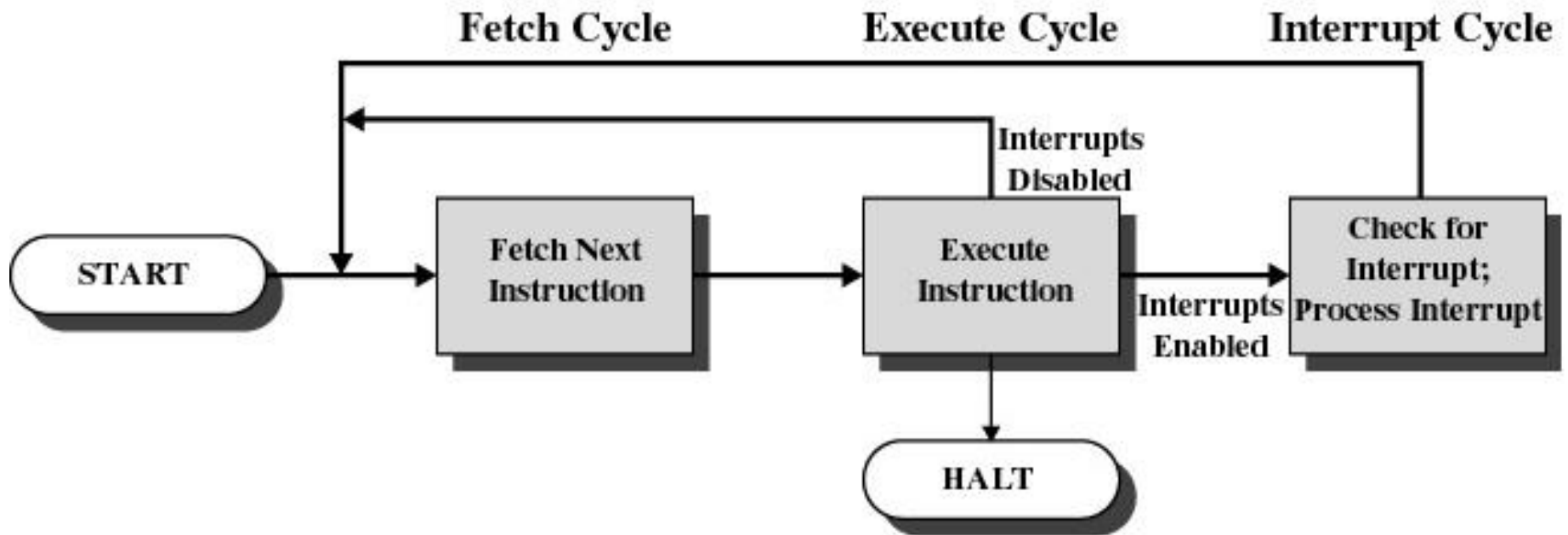


## دسته های وقفه

- **برنامه** : وقفه هایی که به دلیل بعضی شرایط حاصل از اجرای یک دستورالعمل برنامه بروز می کنند. مانند : سرریز شدن محاسباتی، تقسیم بر صفر، تلاش برای اجرای یک عمل غیر مجاز، مراجعه به آدرس خارج از فضای مجاز
- **زمان سنج**: وقفه هایی که توسط زمان سنج داخلی پردازنده تولید میشوند و به سیستم عامل اجازه میدهد بعضی عملیات را به صورت مرتب و دوره ای انجام دهد.
- **ورودی خروجی**: وقفه ای که توسط کنترل کننده I/O تولید میشوند.
- **نقص سخت افزار**: با نقص سخت افزار تولید میشود. مانند قطع برق، خطای توازن حافظه

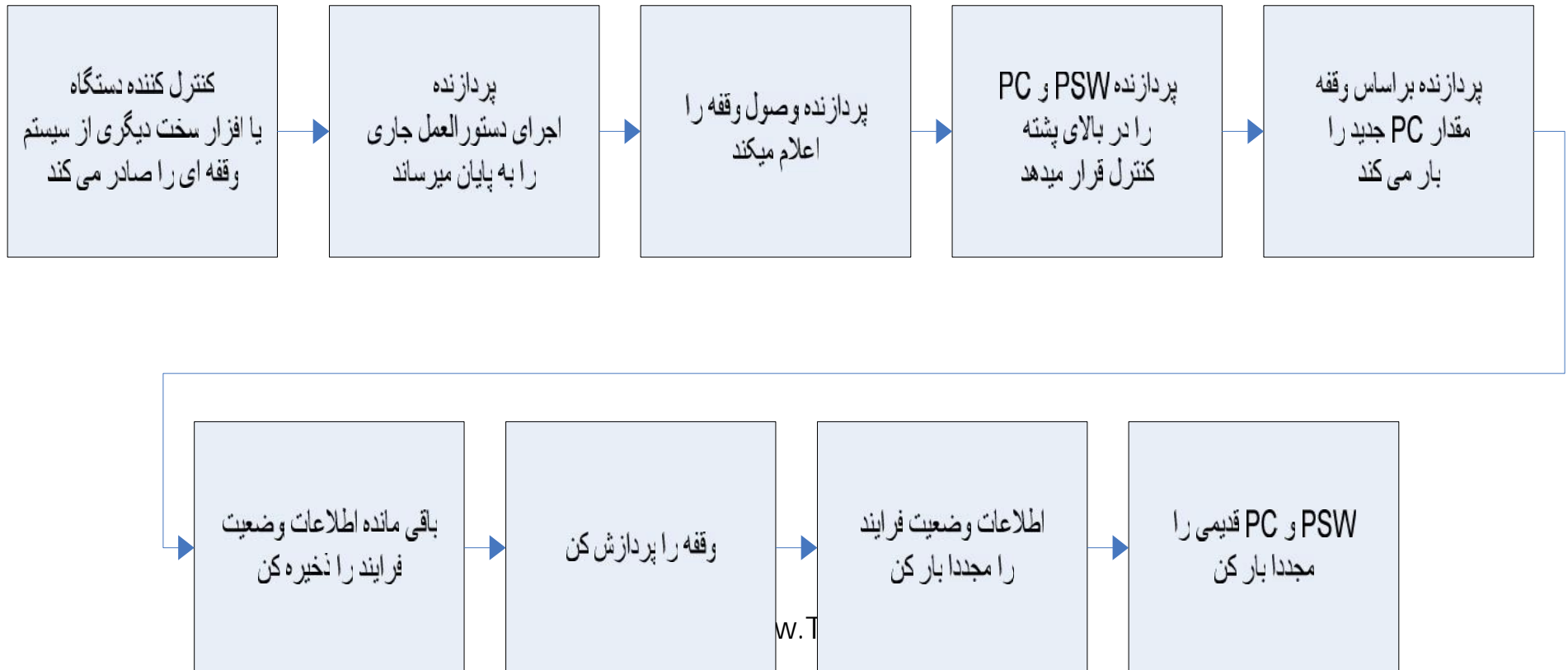
# گرداننده وقفه:

- چرخه دستورالعمل با وقفه: در پایان چرخه اجرا پردازنده وجود وقفه را بررسی می کند. در صورتیکه وقفه مطرح باشد پردازنده اجرای برنامه را مسکوت گذاشته و روال خدماتی وقفه یا برنامه گرداننده وقفه مربوطه را اجرا می کند.

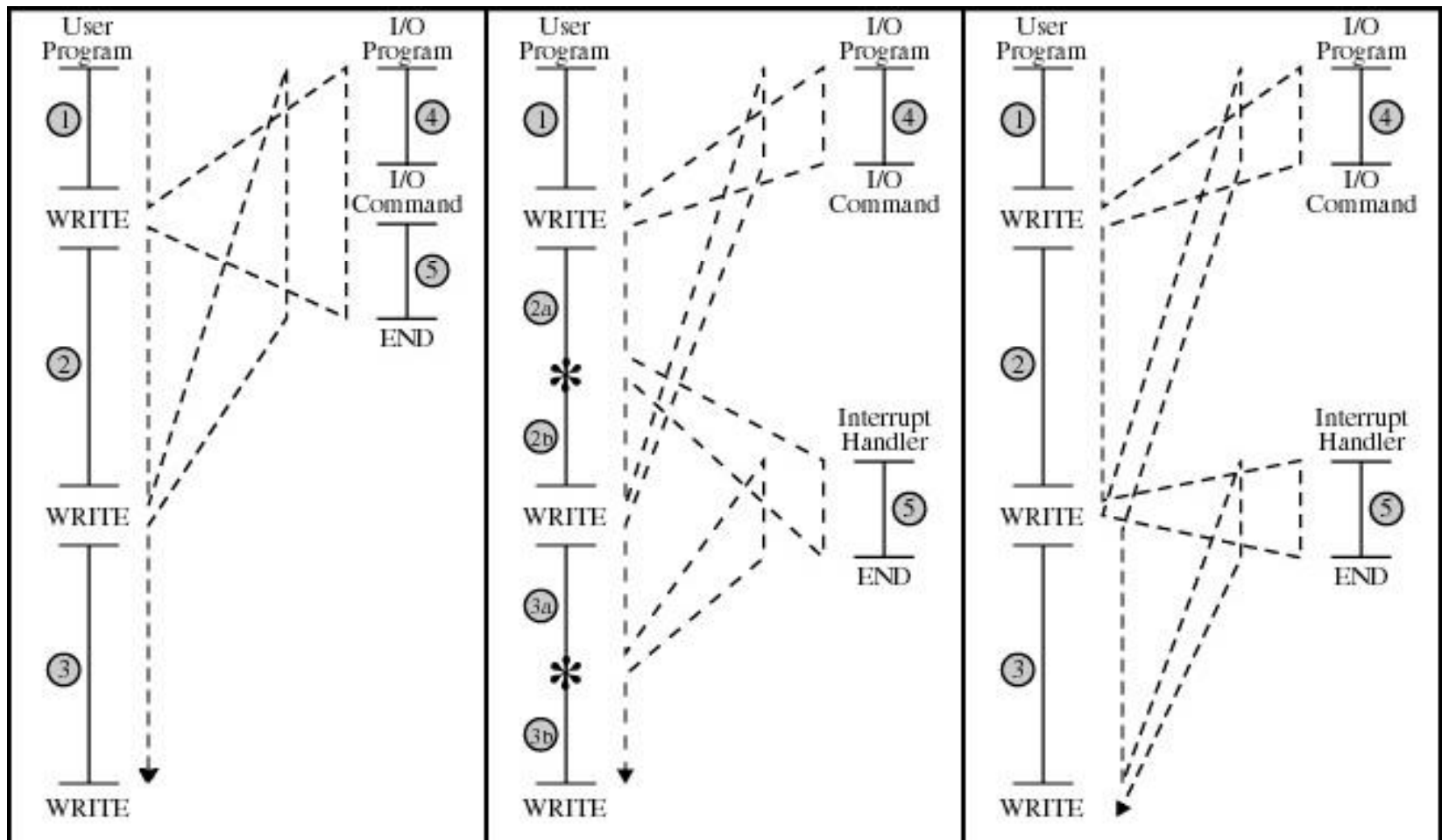


## پردازش وقفه ها :

- بروز وقفه موجب حوادث متعددی در سخت افزار و نرم افزار میشود. این حوادث در شکل نشان داده شده اند:



# جریان کنترل برنامه با و بدون وقفه:



(a) No interrupts

(b) Interrupts; short I/O wait

(c) Interrupts; long I/O wait

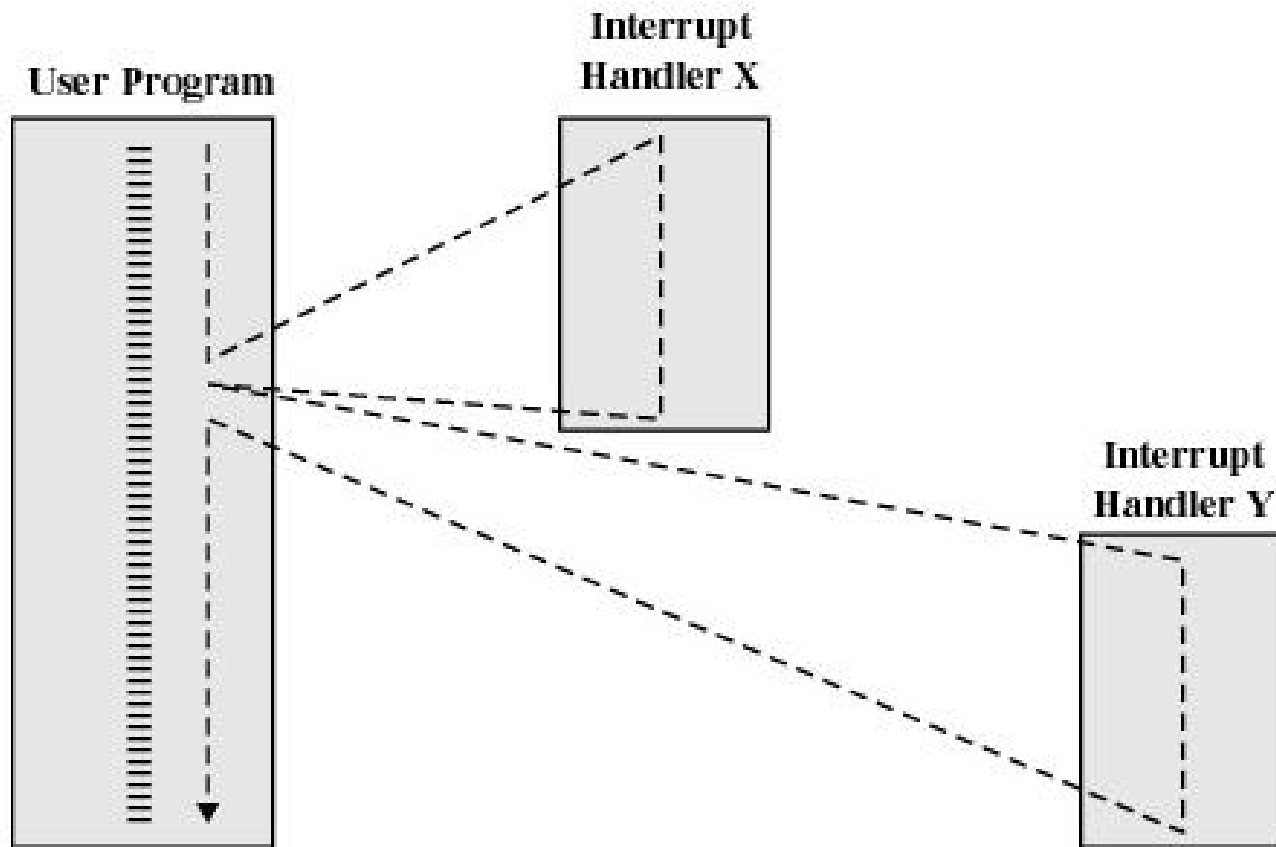


## وقفه های چند گانه :

- بروز چندین وقفه با هم را وقفه چندگانه می گویند.
- در برخورد با وقفه های چند گانه دو رویکرد می تواند اتخاذ شود.
- **غیر فعال کردن سایر وقفه ها در هنگام پردازش یک وقفه :** یعنی پردازنده میخواهد و میتواند سیگنال وقفه را نادیده بگیرد. وقفه های از کار انداخته شده زمانی پردازش می شوند که به کار انداخته شوند. همه وقفه ها به طور ترتیبی قبل از اجرای برنامه کاربر توسط ریزپردازنده پردازش می شوند.
- **نکته منفی این رویکرد :** اولویت نسبی یا محدودیت های زمانی برای وقفه ها نادیده گرفته می شوند.
- **وقفه های تو در تو با تعریف اولویت نسبی برای وقفه ها :** پردازنده مجاز است برای پردازش وقفه با اولویت بالاتر وقفه با اولویت پایین تر را نادیده بگیرد.

## انتقال کنترل با وقفه های چند گانه:

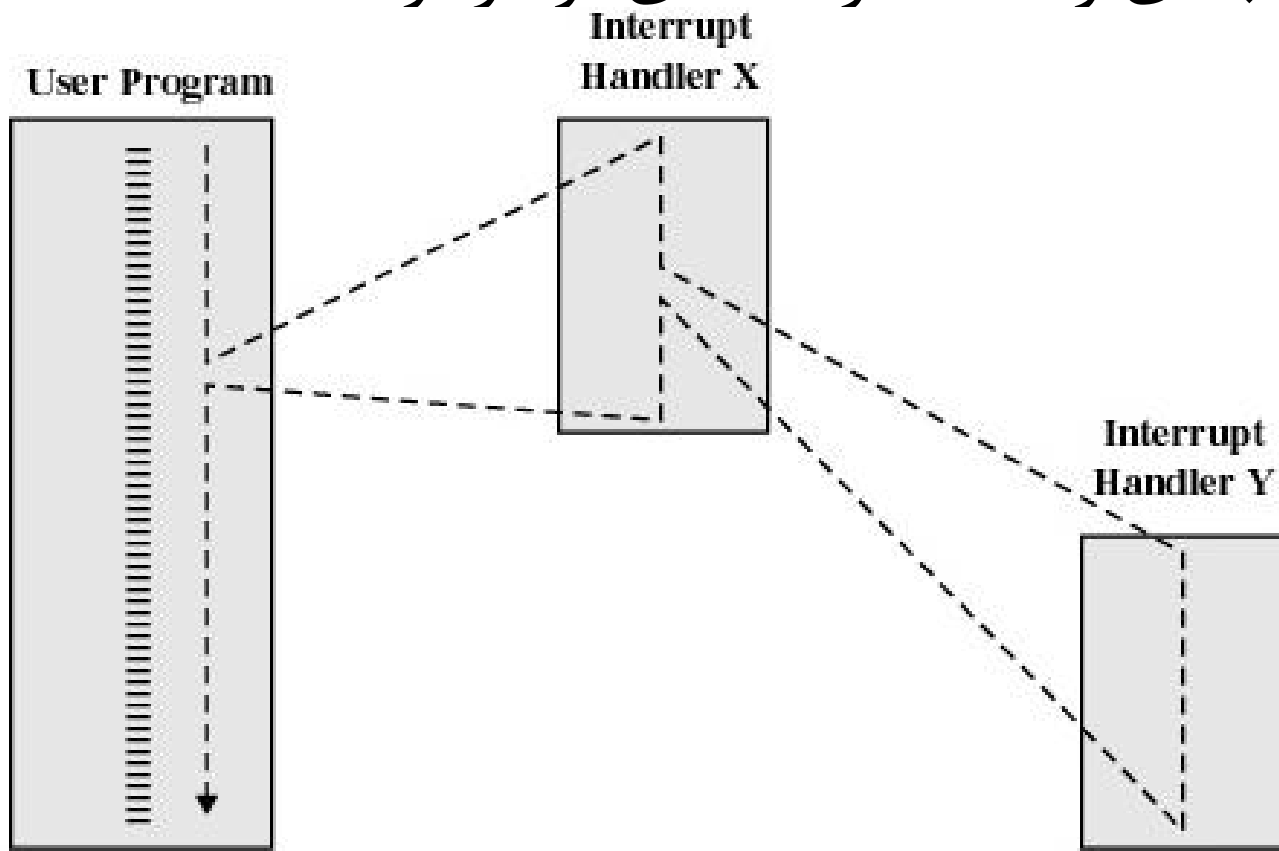
- رویکرد اول : غیر فعال کردن سایر وقفه ها



(a) Sequential Interrupt processing

## انتقال کنترل با وقفه های چند گانه:

- رویکرد دوم : اولویت بندی وقفه ها (وقفه های تو در تو)



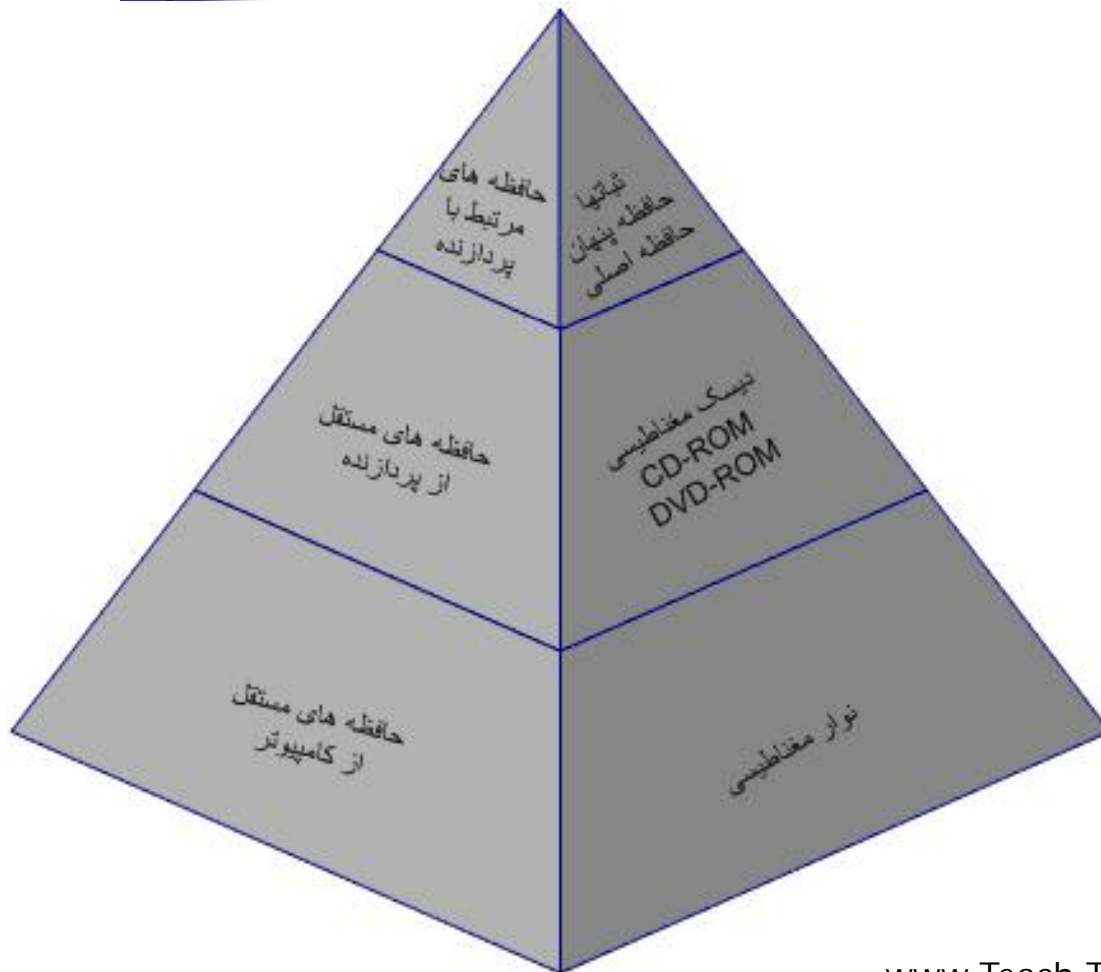
(b) Nested Interrupt processing

## چندبرنامگی:

- پردازنده بیش از یک برنامه برای اجرا دارد.
- ترتیب اجرای برنامه به اولویت نسبی آنها و میزان انتظار آنها برای I/O است.
- هنگامی که برنامه با وقفه مواجه می شود، پردازنده کنترل را به برنامه گرداننده وقفه انتقال میدهد. پس از تکمیل برنامه گرداننده وقفه ممکن است کنترل بلافاصله به برنامه قبلی باز نگردد.



## سلسله مراتب حافظه:



■ برای طراحی حافظه به 3

مورد باید توجه شود:

■ (ظرفیت)

■ (زمان دسترسی)

■ قیمت (هزینه)



## پایین رفتن در سلسله مراتب حافظه:

- کاهش هزینه
- افزایش ظرفیت
- افزایش زمان دسترسی
- کاهش تعداد دفعات دسترسی پردازنده به حافظه
- کلید توفیق این طراحی اصل چهارم می باشد.



## حافظه اصلی (Main memory)

- هر گاه پردازنده به داده‌ها نیاز دارد ابتدا آنها را در حافظه پنهان جستجو می‌کند، اگر آنها را یافت (اصابت) آنها را بازیابی میکند. در غیر این صورت (عدم اصابت) آنها را از **حافظه اصلی** می‌خواند.
- **ثباتها** : کوچکترین ، گران ترین ، سریعترین نوع حافظه اند.
- **حافظه پنهان**: حافظه‌ای قابل رؤیت برای پردازنده، برای تبادل داده‌ها بین حافظه اصلی و ثباتها
- راه‌های توسعه حافظه اصلی :
- از نظر سرعت : استفاده از حافظه پنهان
- از نظر ظرفیت : استفاده از دیسک‌ها

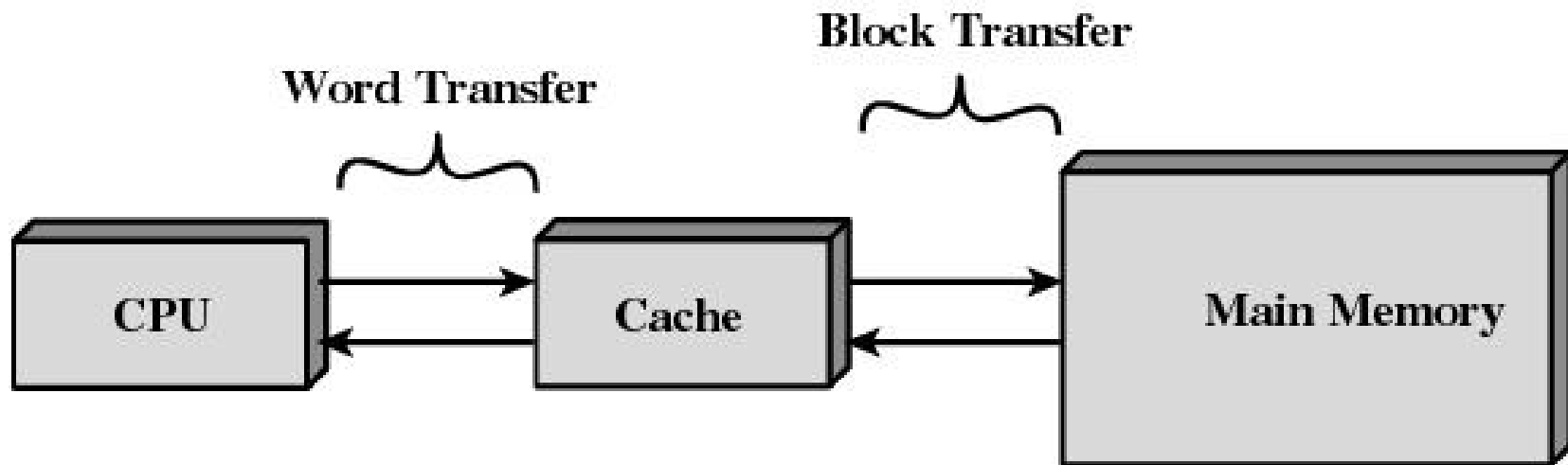


## حافظه پنهان دیسک :

- بخشی از حافظه اصلی را میتوان به عنوان بافر برای داده هایی که قرار است با دیسک مبادله شوند به کار برد که به آن حافظه پنهان دیسک گفته میشود.
- مزایای حافظه پنهان دیسک
  - نوشتن داده ها روی دیسک به صورت بلوکی (خوشه ای) انجام می شود . (نه بصورت جداگانه) ← افزایش کارایی دیسک
  - بعضی از داده ها که قرار است روی دیسک ذخیره شوند ممکن است توسط برنامه دوباره نیاز باشند بنابر این سریعا از حافظه اصلی خوانده میشوند ← مراجعه حافظه کمتر می شود.

## حافظه نهان (Cache):

- برای سیستم عامل قابل مشاهده نیست.
- حافظه نهان به دلیل عدم تطابق سرعت حافظه اصلی با سرعت پردازنده طراحی شد.
- سرعت حافظه اصلی را افزایش میدهد.

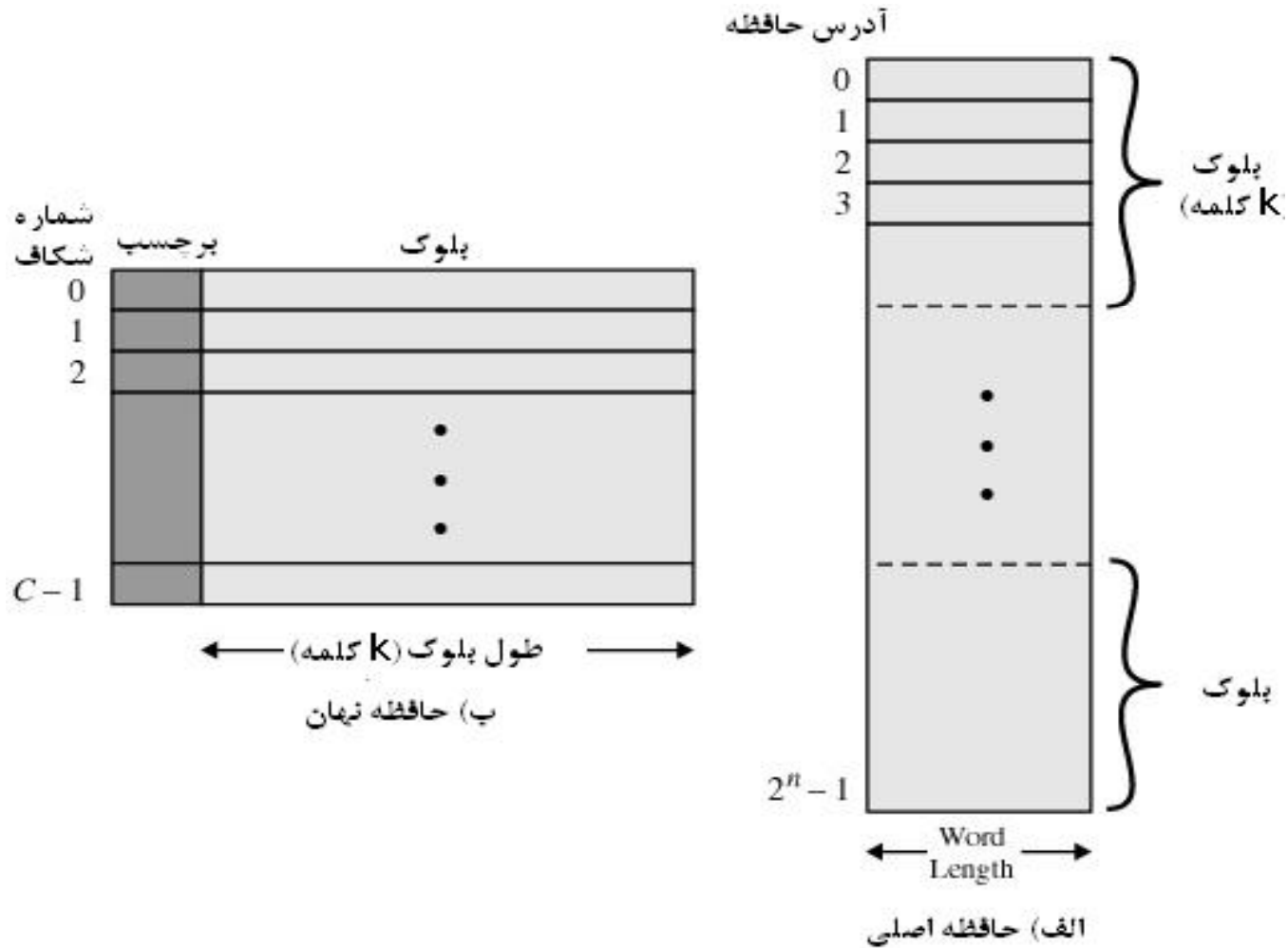




## حافظه نهان (Cache):

- قسمتی از حافظه اصلی را نگه میدارد.
- پردازنده ابتدا حافظه نهان را بررسی میکند.
- اگر در حافظه نهان پیدا نشد بلوکی از حافظه اصلی که حاوی داده های مورد نیاز است به حافظه نهان آورده میشود.
- حافظه اصلی به  $M$  بلوک  $K$  کلمه ای تقسیم میشود و حافظه نهان به  $C$  شکاف  $K$  کلمه ای. اما چون  $M \gg C$  بنابراین هیچ بلوکی نمیتواند برای مدت طولانی در حافظه نهان بماند.

# ساختار حافظه اصلی و حافظه نهان:





## طراحی حافظه نهان:

### ■ اندازه حافظه نهان :

■ یک حافظه کوچک و سریع بسیار کارایی دارد

### ■ اندازه بلوک :

■ واحد ها بین حافظه نهان و حافظه اصلی مبادله میشوند.

■ اصابت به معنای پیدا کردن اطلاعات در حافظه نهان است

■ بلوک های با اندازه بزرگتر تا زمانی میزان اصابت را افزایش میدهند که احتمال وجود داده ها در کاشه بیشتر از احتمال واکشی داده های جدید از حافظه اصلی باشد.





## طراحی حافظه نهان:

### ■ تابع نگاشت:

■ تعیین کننده محل بلوک جدید وارد شده به حافظه نهان

### ■ الگوریتم تعویض:

■ : تعیین کننده بلوکی که باید از حافظه نهان حذف شود تا بلوک جدید جایگزین شود.

■ معمولا از الگوریتم LRU(Least Recently Used) استفاده میشود.

## طراحی حافظه نهان:

### ■ سیاست نوشتن:

- تعیین می کند چه زمانی بلوک موجود در کاشه در حافظه اصلی نوشته شود.
- زمانی که محتوای بلوک موجود در حافظه نهان تغییر کند
- هنگام جایگزینی آن را روی حافظه اصلی
  - حداقل میزان نوشتن
  - متروک ماندن حافظه

## حافظه:

### ■ انواع مدل های حافظه:

- ثباتها، حافظه نهان، حافظه اصلی، دیسکها، CD، ...
- حافظه ها از نظر قیمت، سرعت و ظرفیت متفاوتند.
- حافظه نهان یک حافظه کوچک ولی بسیار سریع است که سرعت حافظه اصلی را تقویت میکند.



## روش های انتقال ورودی خروجی:

---

- ورودی / خروجی برنامه سازی شده
- ورودی / خروجی مبنی بر وقفه
- دسترسی مستقیم به حافظه (DMA)



## ورودی / خروجی برنامه سازی شده :

- هنگام مواجهه با یک عمل I/O در حین اجرای برنامه، CPU فرمان لازم را به مؤلفه I/O می فرستد.
- مؤلفه I/O فرمان را اجرا میکند.
- بیت های وضعیت را تغییر میدهد.
- هیچ عمل دیگری انجام نمی دهد (بویژه وقفه نمی دهد)
- وظیفه CPU است که متناوباً وضعیت مؤلفه I/O را چک کند.
- در حین انجام فرامین توسط I/O، پردازنده مدام باید در چرخه تست وضعیت دستگاه باقی بماند بنابراین کارایی سیستم بشدت پایین میاید.



## ورودی / خروجی مبنی بر وقفه :

- در این روش CPU پس از صدور فرمان به I/O به کار دیگری مشغول میشود.
- وقتی I/O برای مبادله داده ها آماده شد وقفه ای به پردازنده صادر میکند.
- CPU اجرای دستور العمل فعلی را به پایان میرساند.
- CPU درخواست وقفه را شناسایی میکند.
- CPU مقدار PC و PSW را ذخیره میکند.
- CPU آدرس اولین دستور روال گرداننده وقفه را در PC کپی میکند
- پس از اجرای روال گرداننده وقفه مقدار PC و PSW بازیابی میشود.
- CPU به اجرای برنامه قبلی خود باز میگردد.



## دسترسی مستقیم به حافظه (DMA) :

- در این روش از مولفه ای به نام DMA استفاده میشود. هنگام خواندن یا نوشتن پردازنده فرامین زیر را به DMA صادر می کند و خودش کنار میرود.
  - عمل خواندن مورد نیاز است یا نوشتن
  - آدرس دستگاه I/O درگیر
  - محلی از حافظه که از آن محل خواندن یا نوشتن شروع میشود.
  - تعداد کلماتی که باید خوانده یا نوشته شوند.
- پس از اتمام عملیات DMA به پردازنده وقفه ای مبنی بر اتمام عملیت میدهد.
- در این روی CPU تنها در ابتدا و انتهای عملیات I/O مشغول است.