معماری کامپیوتر (۱۳۹۱-۱۱-۱۳)

جلسه‌ی نوردهم

دانشگاه شهید بهشتی

دانشکده‌ی مهندسی برق و کامپیوتر

بهار ۱۳۹۱

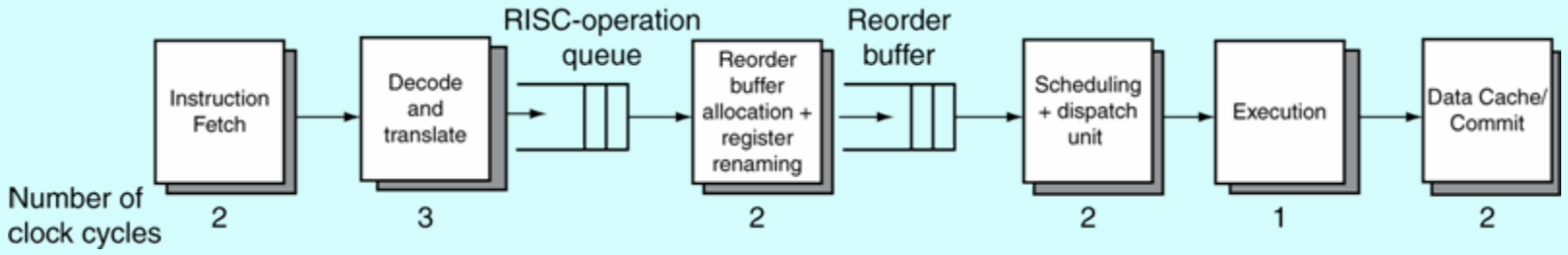
احمد محمودی ازناوه

فهرست مطالب

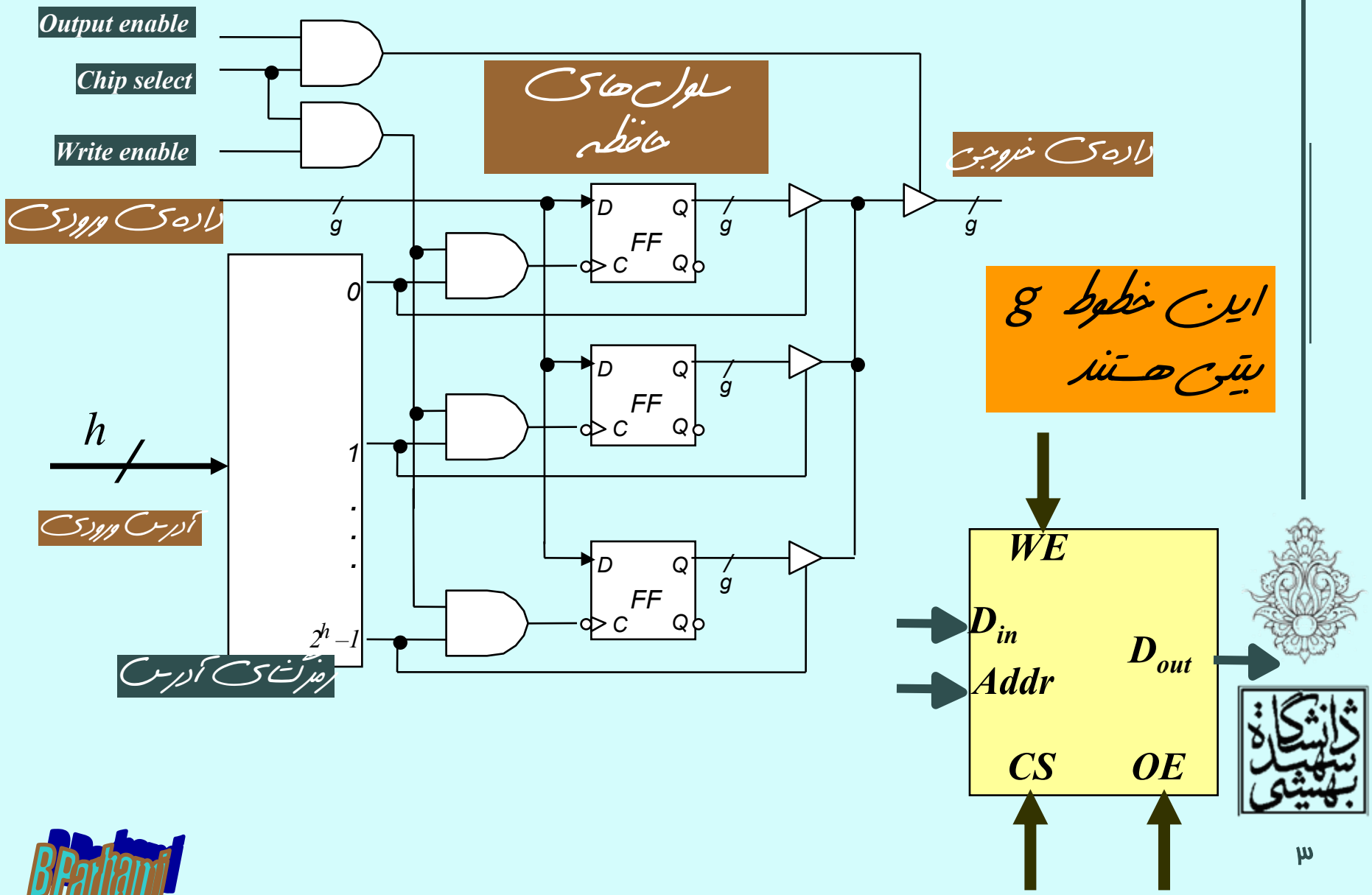
- نگاهی به خط لوله‌ی پردازنده‌های امروزی
- حافظه
  - حافظه‌ی نهان

Opeteron X4

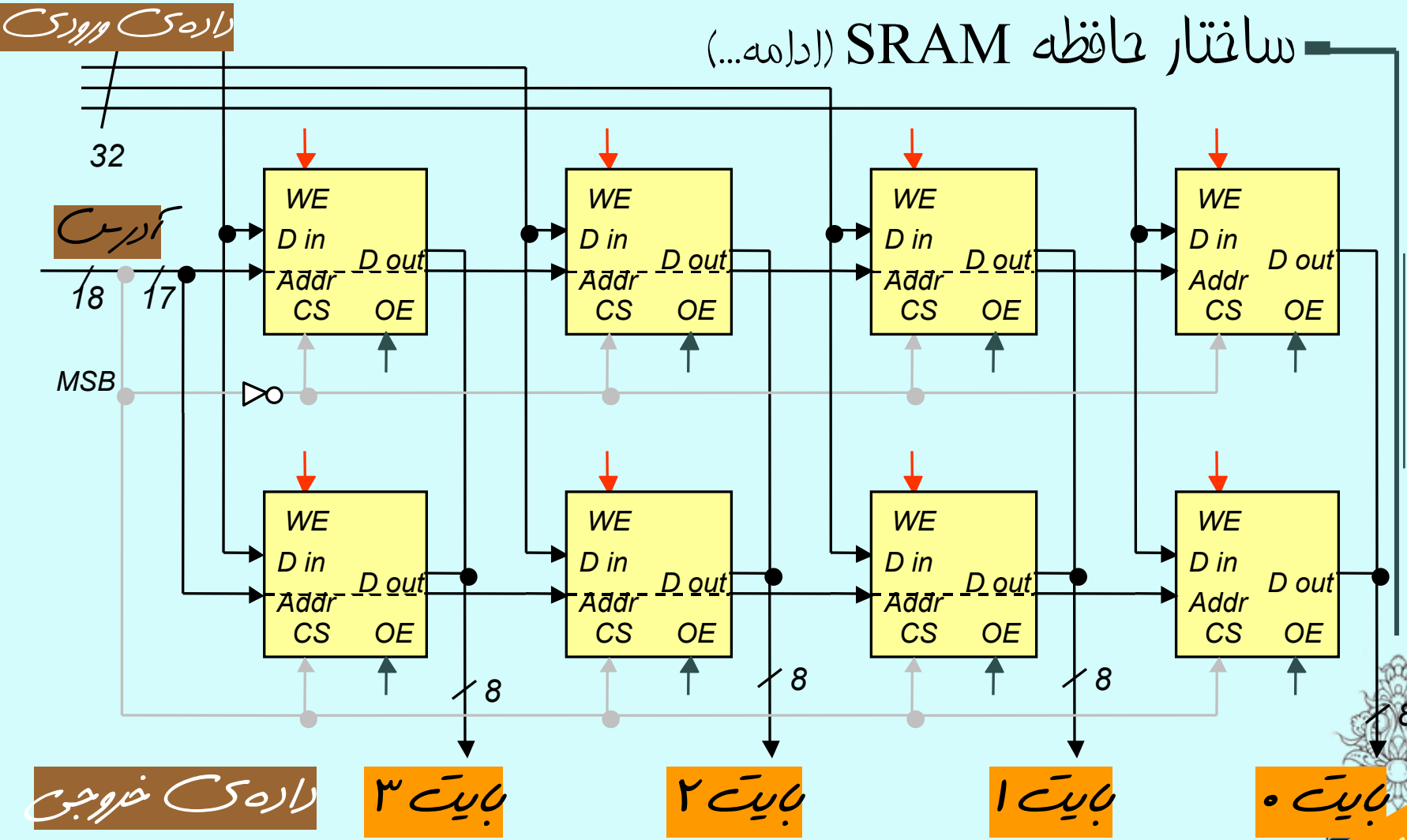
با وجود بحث‌هایی که در مورد خط لوله داشتیم، هنوز مباحث مهمی مطرح نشده است، به همین اندازه بسنده کرده و در پایان نگاهی به ساختار خط لوله در پردازنده‌ی Opeteron X4 خواهیم داشت:



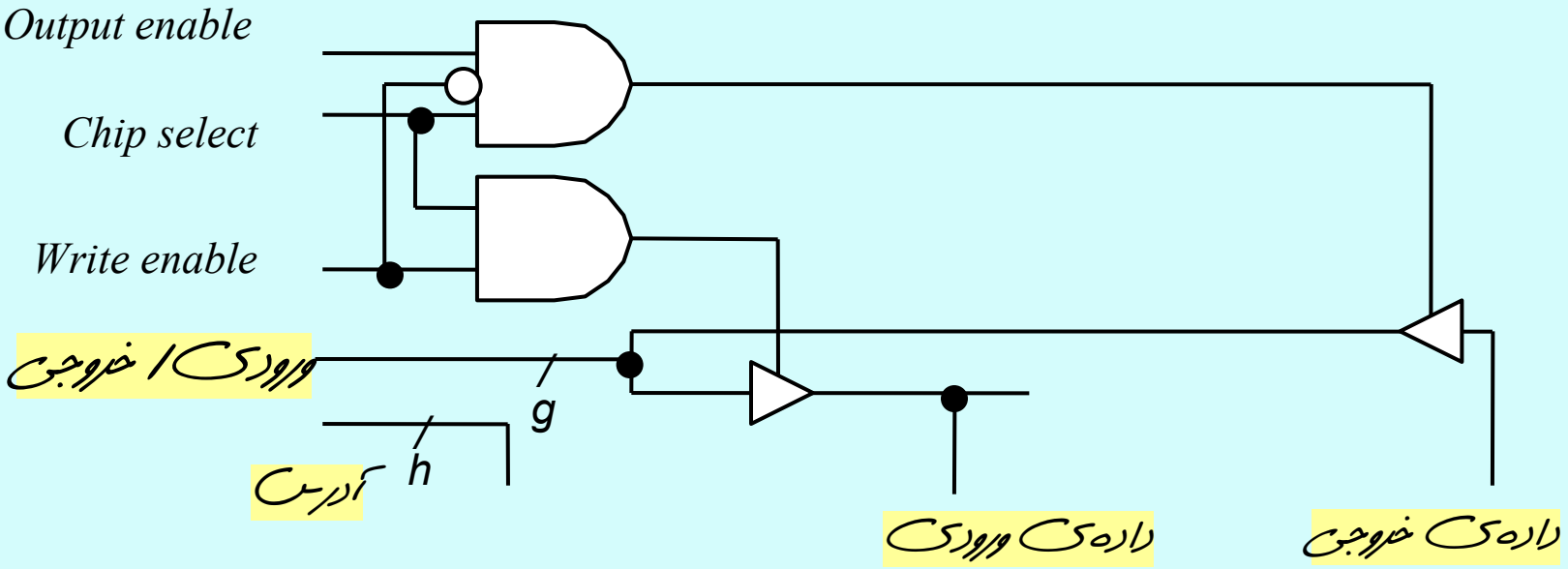
ساختار حافظه SRAM



ساختار حافظه SRAM (ادامه...)

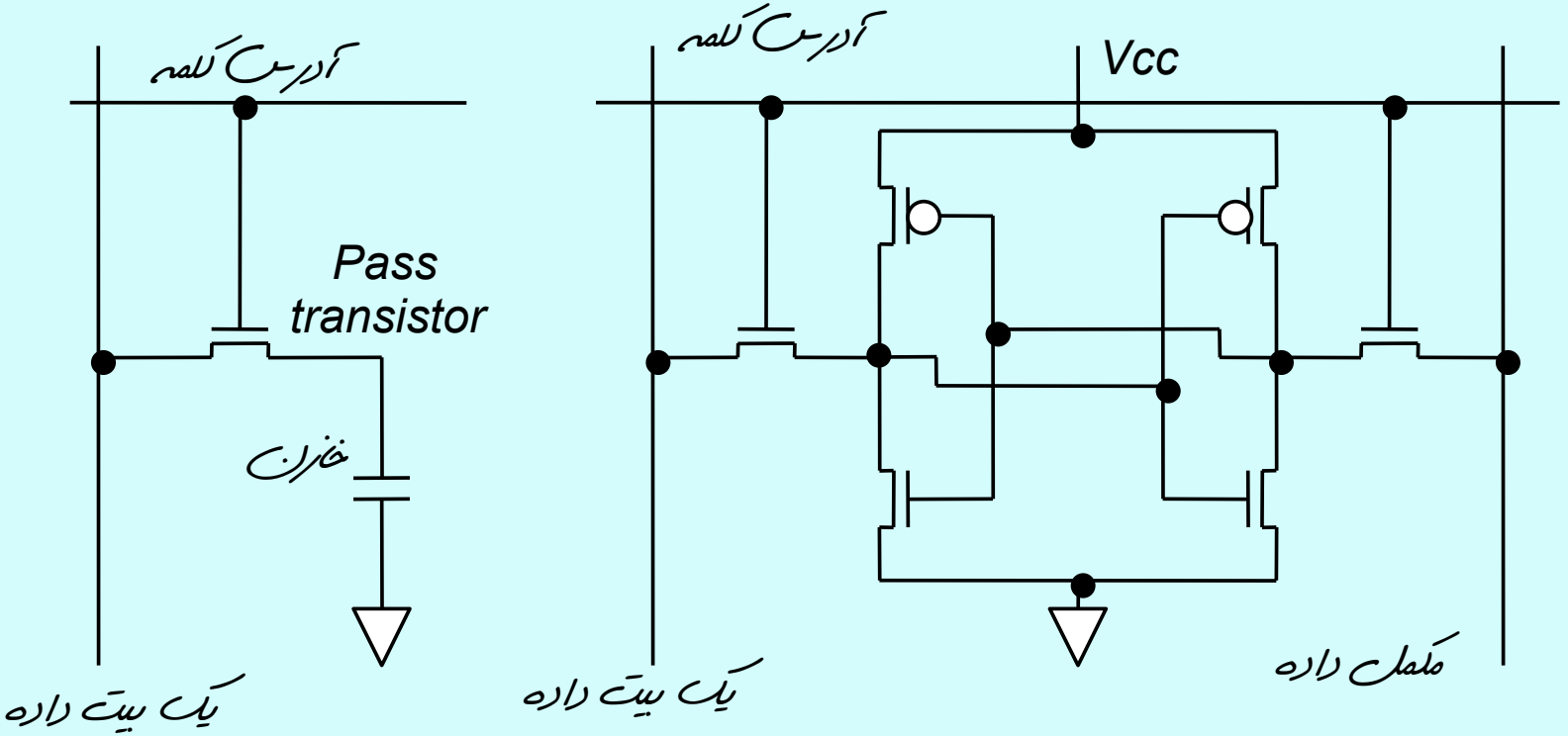


گذرگاه داده دو جهته



هنگامی که داده‌ی ورودی و خروجی به یک گذرگاه مشترک متصل شده باشند،
هنگامی که داده‌ای برای نوشتن در گذرگاه داده قرار می‌گیرد، خروجی سلول‌های حافظه باید غیرفعال شوند

DRAM در برابر SRAM

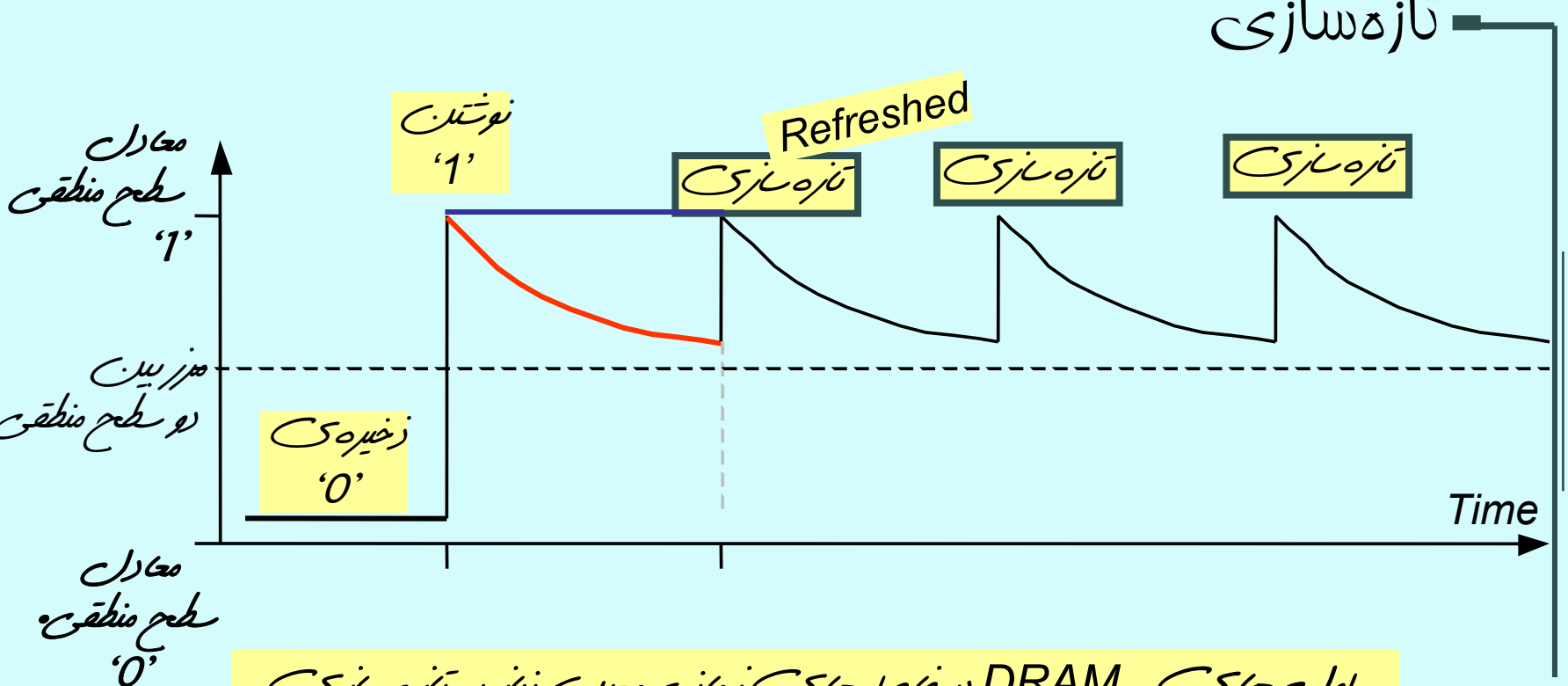


سلول DRAM

سلول SRAM

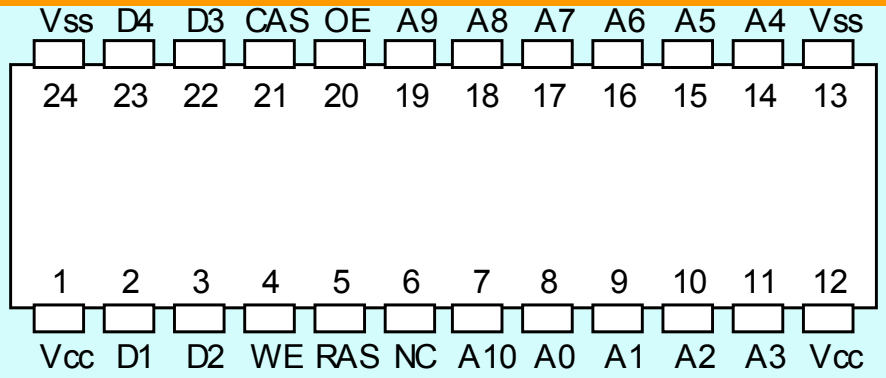
سلول‌های DRAM از سلول‌های SRAM ساده‌تر هستند، در نتیجه با این سلول می‌توان حافظه‌هایی با گنجایش بالاتر و در عین حال ارزان‌تر ساخت

تازه‌سازی



سلول‌های DRAM در فاصله‌های زمانی معین نیاز به تازه‌سازی محتوای خود دارند

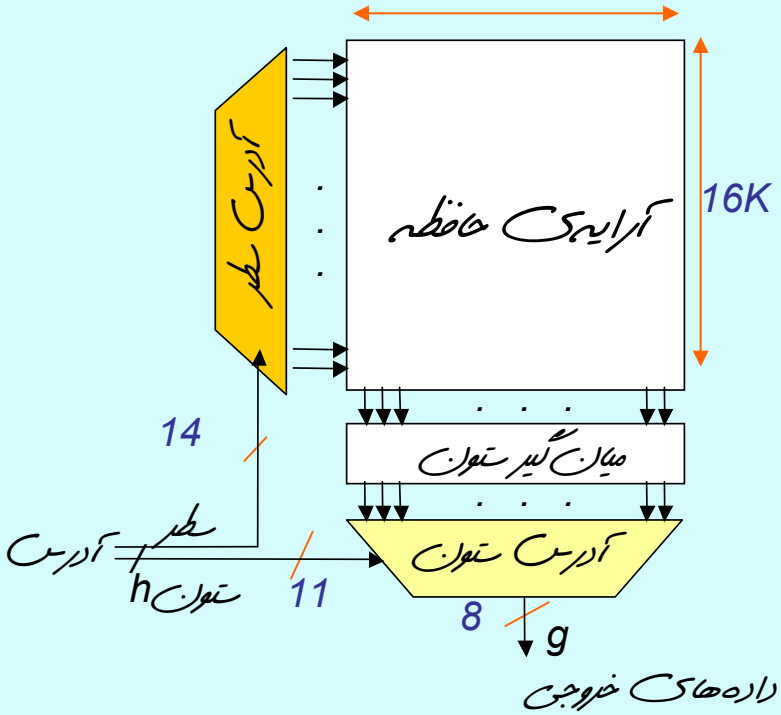
به نظر شما نیاز به تازه‌سازی چه مشکلی ایجاد می‌کند؟



Legend:

A*i*	Address bit *i*
CAS	Column address strobe
D*j*	Data bit *j*
NC	No connection
OE	Output enable
RAS	Row address strobe
WE	Write enable

یک DRAM با گنجایش 256 Mb مفروض است. این حافظه که به صورت $32M \times 8$ قابل دسترسی است، به صورت آرایه‌ای $16K \times 16K$ می‌باشد. هر سطر دست‌کم هر 50 ms **باید** تازه‌سازی شود؛ این کار 100ns زمان می‌برد، چند درصد از پهنای باند هدر می‌رود؟

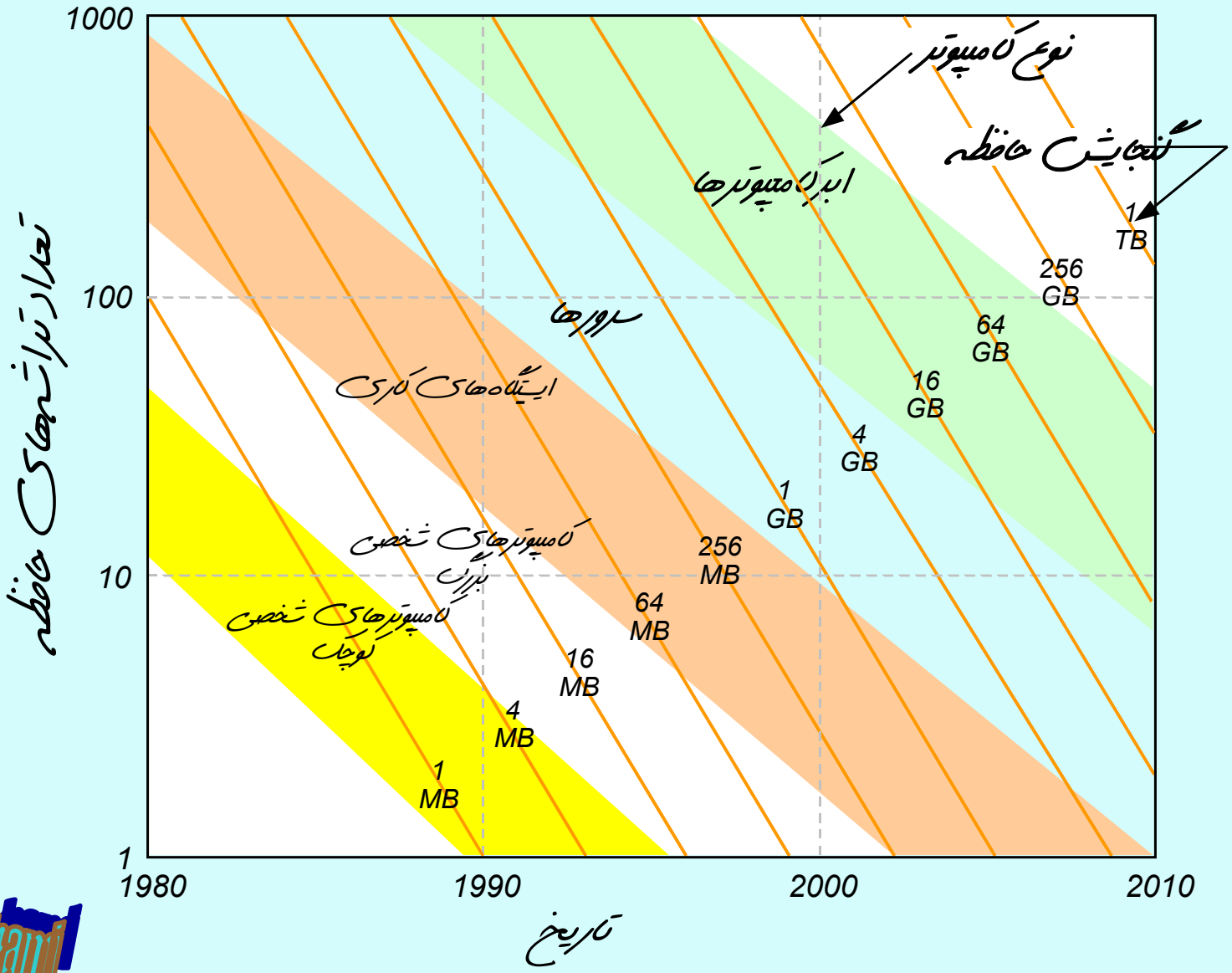


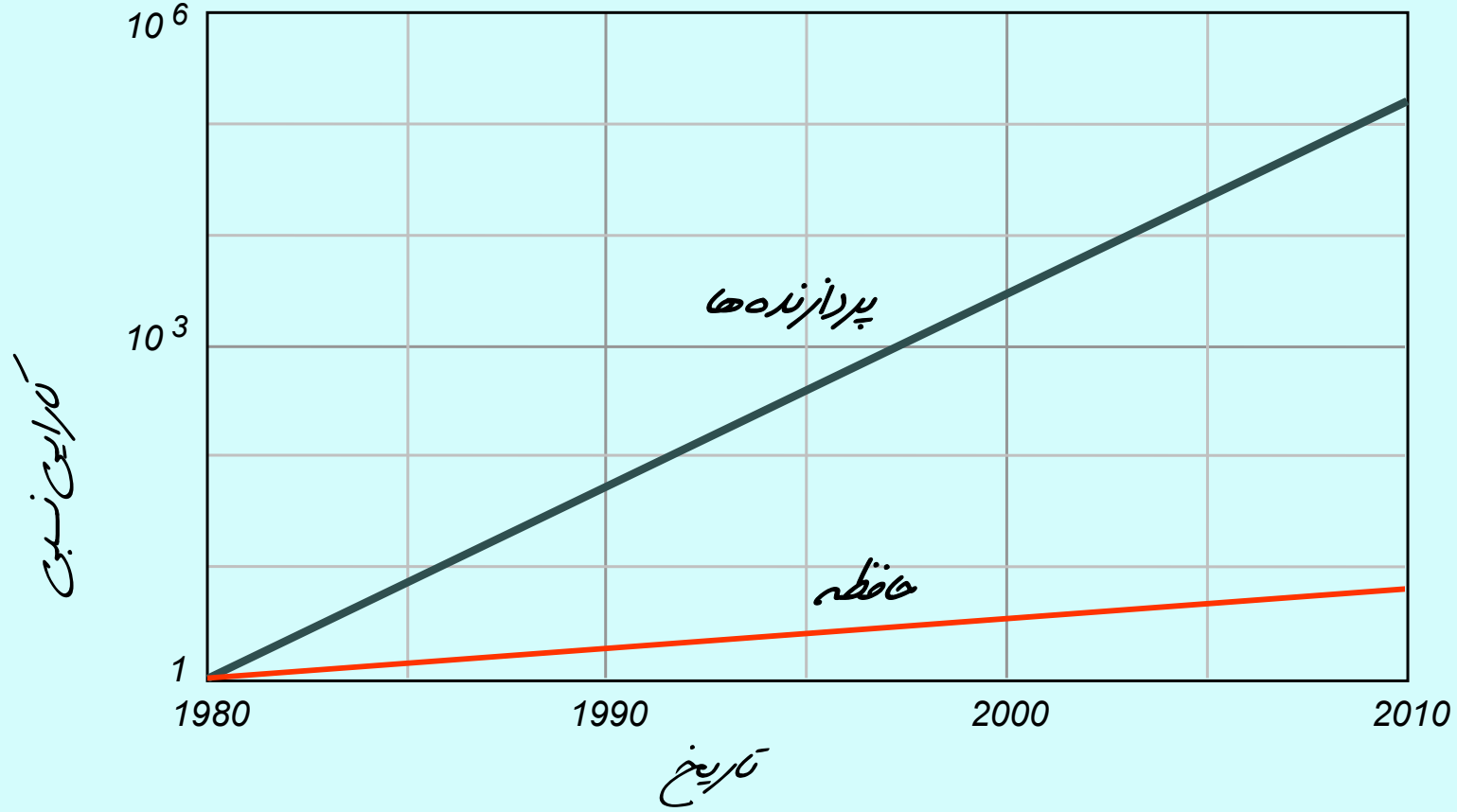
راه حل:

16 $\times 1024 \times 100$ ns = 1.64 ms.

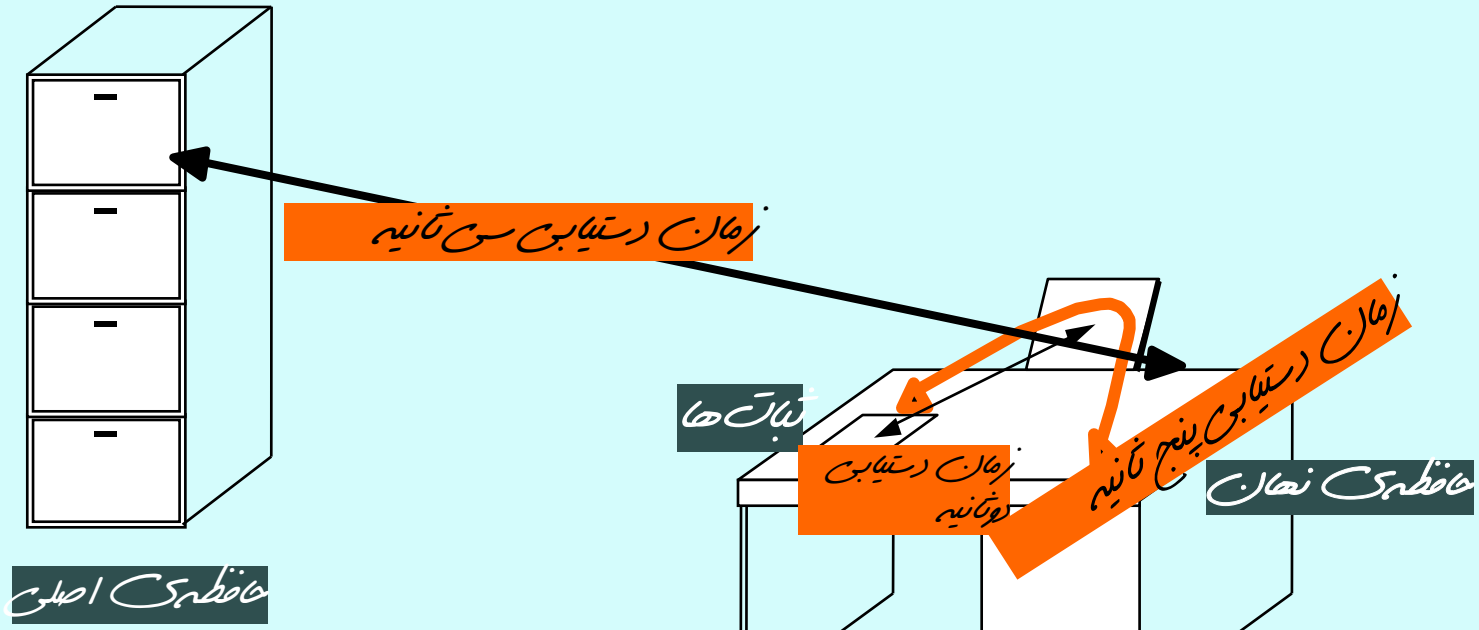
1.64/50 = 3.3%.

از کل پهنای باند بابت تازه‌سازی هدر می‌رود


1000
100
10
1
1980
1990
2000
2010
تعداد ترانزیستورهای حفظه
تاریخ
نوع کامپیوتر
گنجایش حافظه
ابر کامپیوترها
سرورها
ایستگاه‌های کاری
کامپیوترهای شخصی بزرگ
کامپیوترهای شخصی کوچک
1 TB
256 GB
64 GB
16 GB
4 GB
1 GB
256 MB
64 MB
16 MB
4 MB
1 MB


10^6
10^3
1
پردازنده‌ها
حافظه
تراکم نسبی
1980
1990
2000
2010
تاریخ

سلسله‌مراتب حافظه



سرعت، گنجایش و قیمت حافظه

Memory technology	Typical access time	$ per GB in 2008
SRAM	0.5-2.5 ns	$2000-$5000
DRAM	50-70 ns	$20-$75
Magnetic Disk	5,000,000-20,000,000 ns	$0.20-$2

یک حافظه‌ی ایده‌آل، زمان دسترسی پایین (مانند SRAM) و گنجایش بالا (مانند دیسک سخت)

locality

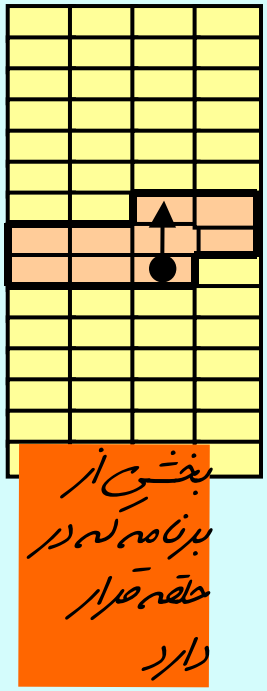
- برای اجرای برنامه‌های ذخیره شده در حافظه، مراجعه به حافظه به بخش کوچکی محدود می‌شود.

همجواری

Temporal locality

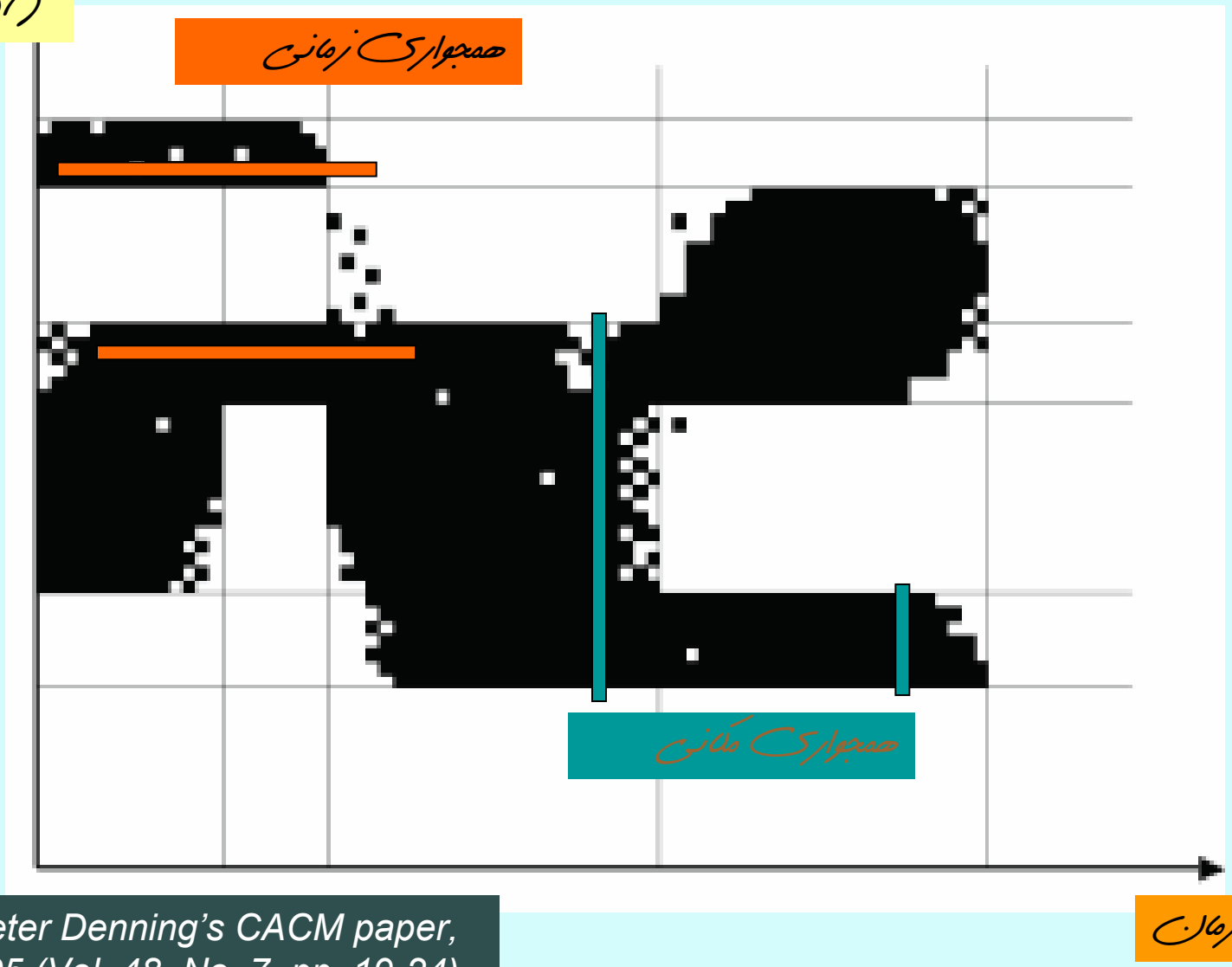
- **همجواری زمانی**: بخشی از حافظه که مورد مراجعه قرار گرفته است، با احتمال بالایی مجدداً مورد استفاده قرار می‌گیرد.
  - متغیرها، و یا دستورهایی که در حلقه قرار گرفته‌اند.

Spatial locality



- **همجواری مکانی**: نواحی مجاور بخشی از حافظه که مورد مراجعه قرار گرفته است، با احتمال بالایی مجدداً مورد استفاده قرار می‌گیرد.
  - آرایه‌ها، و یا توالی دستورالعمل‌های یک برنامه

همجواری (ادامه...)



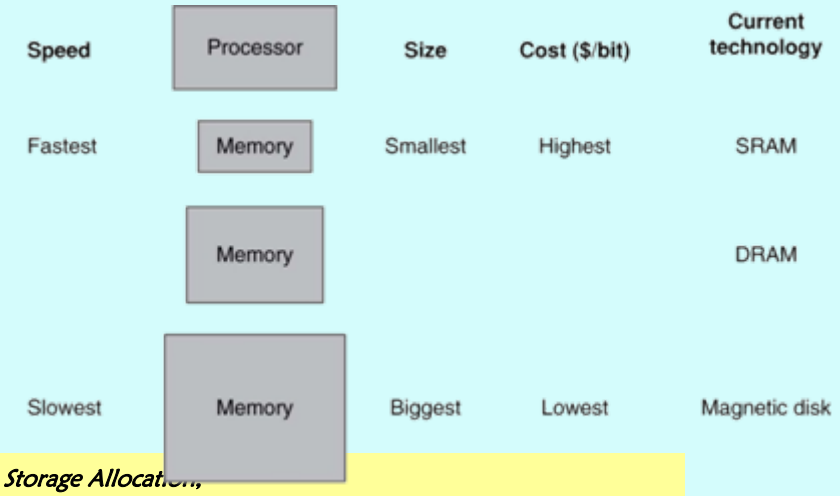
From Peter Denning's CACM paper, July 2005 (Vol. 48, No. 7, pp. 19-24)

همجواری (ادامه...)

- سلسله مراتب در در حافظه:
  - همه چیز را در دیسک سخت ذخیره کن
  - یک نسخه از موارد نیاز را در DRAM بنویس
  - یک نسخه از موارد مورد نیاز که اخیرا به کار گرفته شده‌اند را در SRAM ذخیره کن.

Main memory

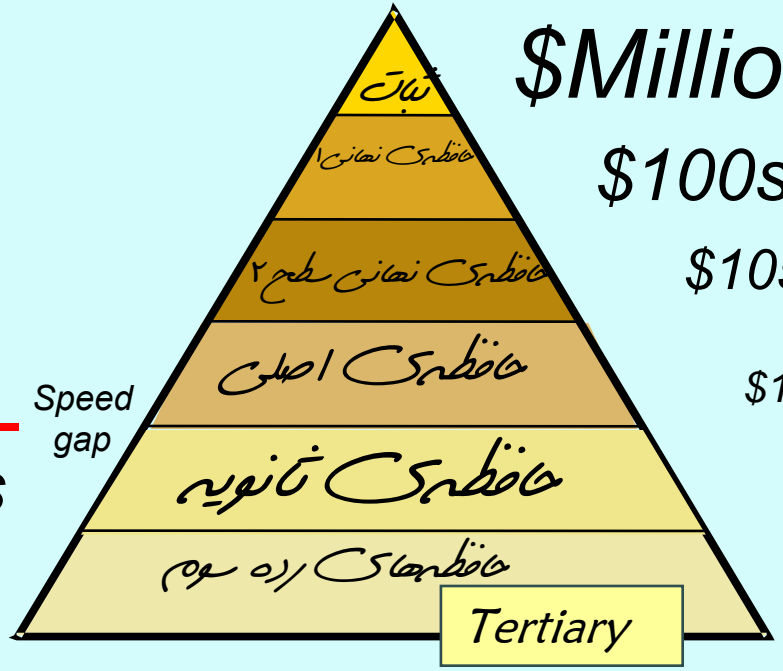
Cache memory attached to CPU



M. V. Wilkes, "Slave Memories and Dynamic Storage Allocation," IEEE Transactions on Electronic Computers, vol. EC-14, no. 2, pp. 270-271, April 1965.

سطوح مختلف حافظه

گنجایش	زمان دستیابی	قیمت هر GB
100s B	ns	$Millions
10s KB	a few ns	$100s Ks
MBs	10s ns	$10s Ks
100s MB	100s ns	$1000s
10s GB	10s ms	$10s
TBs	min+	$1s



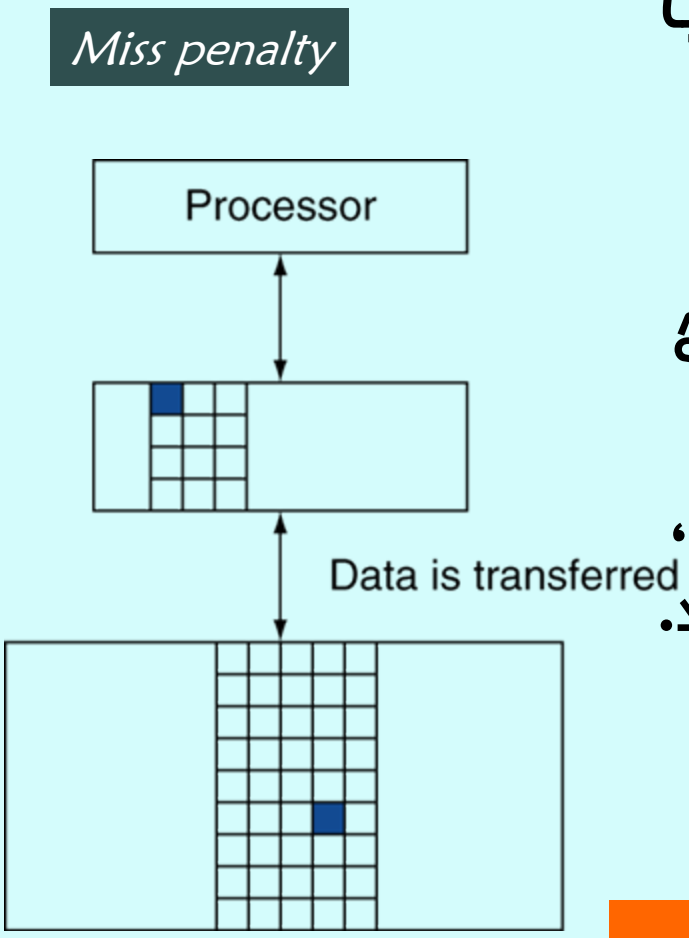
سلسله‌مراتب در حافظه

- کوچک‌ترین واحد حافظه در حافظه‌ی نهان که وجود یا عدم وجود آن قابل بررسی است، سطر یا بلوک نامیده می‌شود.
- در صورتی که داده در سطوح بالا وجود داشت، دستیابی موفق بوده است و hit شده است!

$$\text{Hit ratio} = \frac{\#\text{Hits}}{\#\text{Accesses}}$$

سلسله‌مراتب در حافظه (ادامه...)

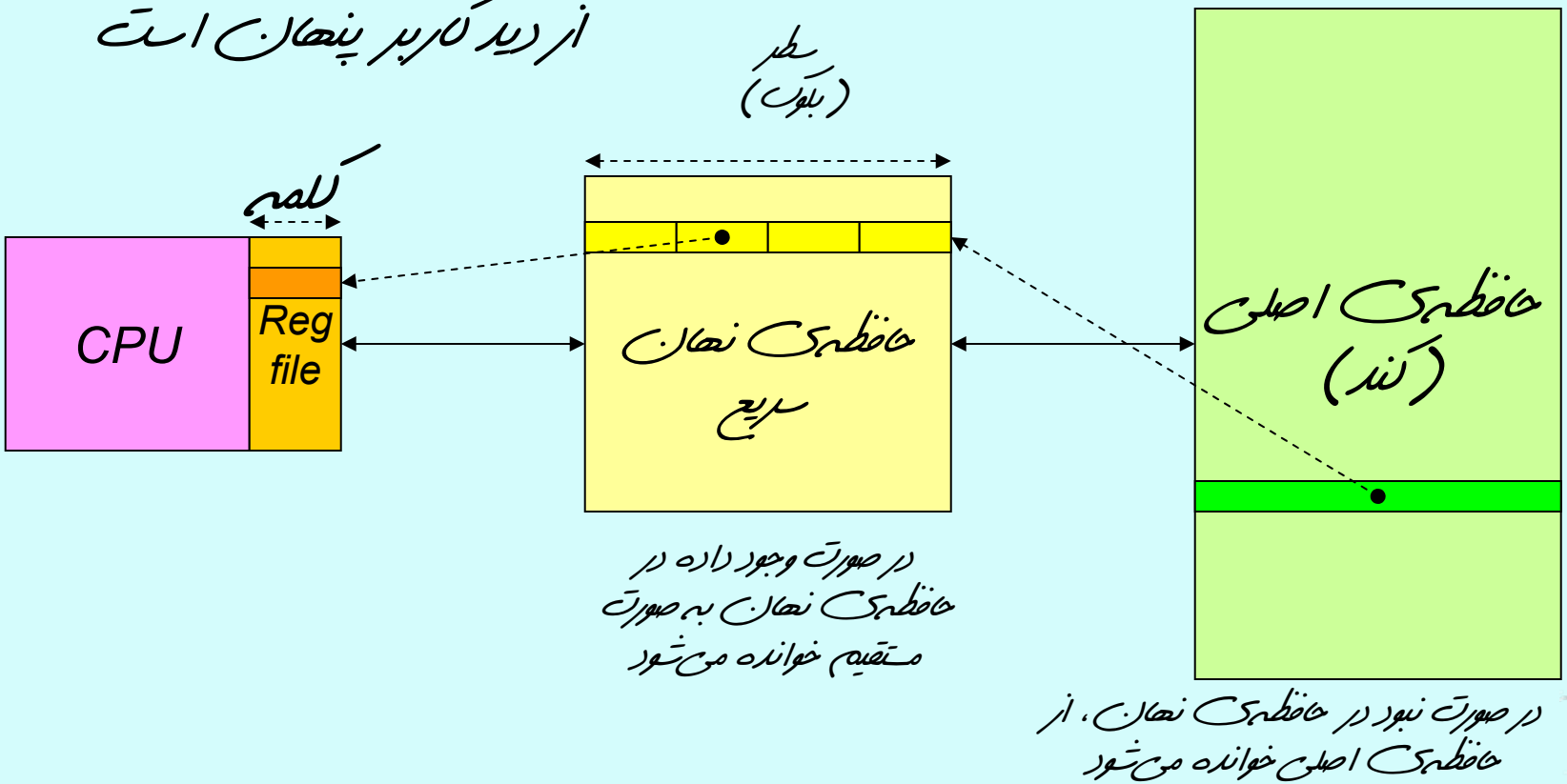
- در غیر این صورت درخواست، با پاسخی موفقیت‌آمیز روبرو نمی‌شود و miss شده است!
  - در این حالت باید داده از سطوح پایین‌تر منتقل شود.
  - زمانی که صرف انتقال می‌شود، جریمه‌ی فقدان نامیده می‌شود.



$$\text{Miss ratio} = \frac{\#\text{Miss}}{\#\text{Accesses}} = 1 - \text{Hit ratio}$$

حافظه‌ی نهان (ادامه...)

دسترسی به حافظه نهان
از دید کاربر پنهان است



در صورت وجود داده در
حافظه‌ی نهان به صورت
مستقیم خوانده می‌شود

در صورت نبود در حافظه‌ی نهان، از
حافظه‌ی اصلی خوانده می‌شود

در صورت در اختیار داشتن یک سطح حافظه‌ی نهان با hit ratio برابر با h

$$Ceff = hCfast + (1 - h)(Cslow + Cfast) = Cfast + (1 - h)Cslow$$