معماری کامپیوتر (۱۳۹۱-۱۱-۱۳)
جلسه‌ی بیستم

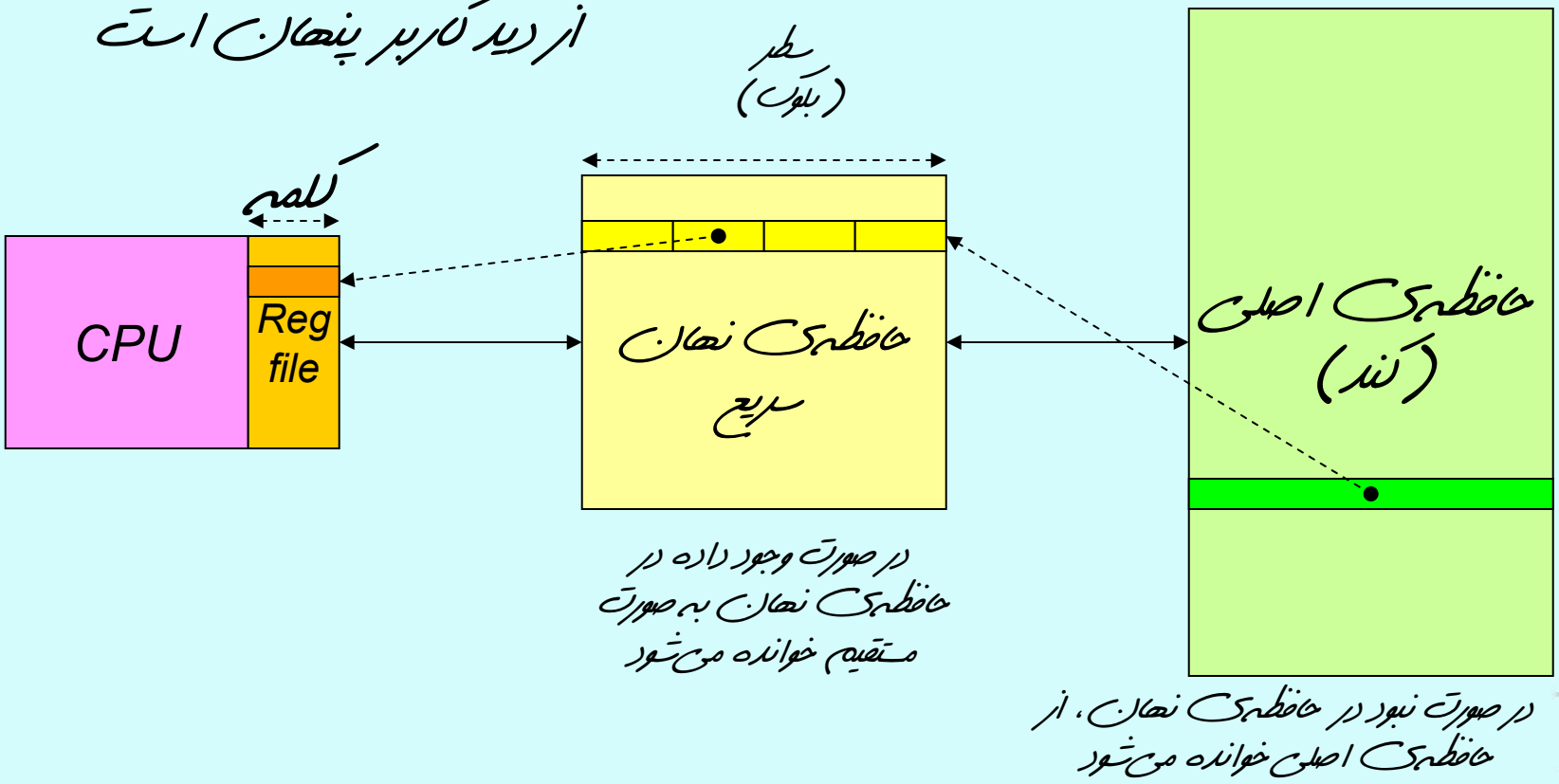
دانشگاه شهید بهشتی
دانشکده‌ی مهندسی برق و کامپیوتر
بهار ۱۳۹۱
احمد محمودی ازناوه

فهرست مطالب

- حافظه‌ی نهان
  - نگاشت مستقیم

حافظه‌ی نهان (ادامه...)

دسترسی به حافظه نهان
از دید کاربر پنهان است



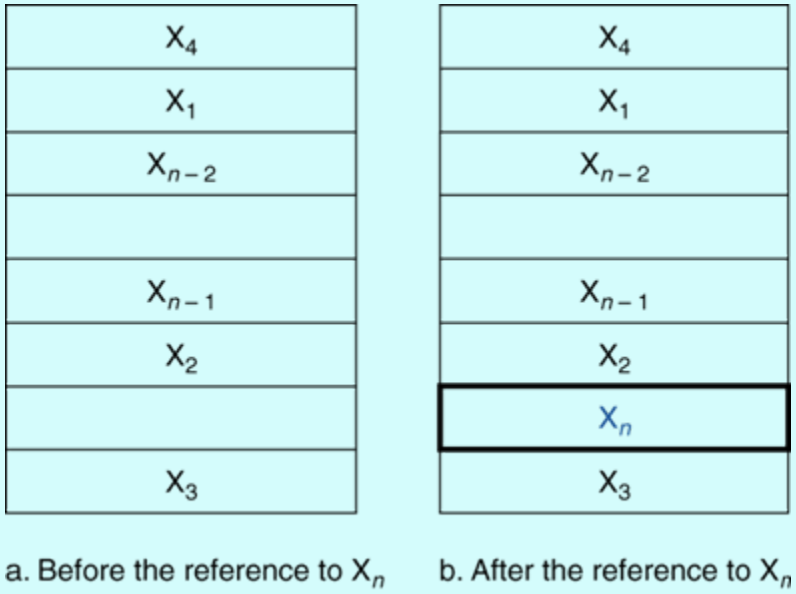
در صورت در اختیار داشتن یک سطح حافظه‌ی نهان با hit ratio برابر با h

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

حافظه‌ی نهان

- **حافظه‌ی نهان:**
  - در سلسله مراتب حافظه، نزدیک‌ترین واحد به پردازشگر مرکزی است.
  - فرض می‌کنیم به خانه‌های حافظه با آدرس‌های زیر مراجعه می‌شود:

X1, …, Xn–1, Xn

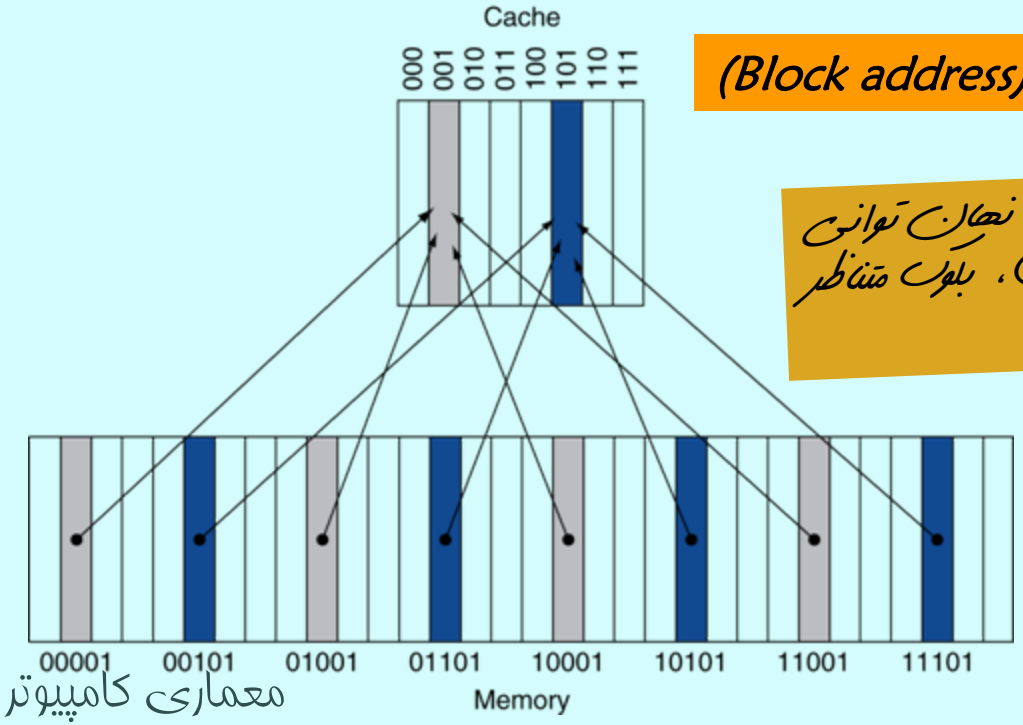


a. Before the reference to X_n b. After the reference to X_n

- از کجا بدانیم که داده در حافظه‌ی نهان وجود دارد؟
- در صورت موجود بودن، چگونه جای آن را بیابیم؟

نگاشت مستقیم

- ساده‌ترین راه، اختصاص یک بلوک از حافظه‌ی نهان برای هر بلوک در حافظه‌ی اصلی است.
- برای محاسبه‌ی آدرس چنین بلوکی از رابطه‌ی زیر استفاده می‌شود.



(Block address) modulo (#Blocks in cache)

در صورتی که تعداد بلوک‌های حافظه‌ی نهان توانی از دو باشد، بیت‌های کم ارزش آدرس، بلوک متناظر با حافظه را نشان خواهد داد

نگاشت مستقیم

- از کجا بدانیم کدام بلوک حافظه در حافظه‌ی نهان قرار گرفته است؟
  - بیت‌های پرارزش بلوک نیز باید ذخیره شوند.
  - این بیت‌ها برچسب (tag) نامیده می‌شود. در واقع با کمک tag مشخص می‌شود کدامین بلوک از بین بلوک‌هایی که مجاز به حضور در یک بلوک حافظه‌ی نهان هستند، در آن‌جا قرار دارند؟
- چگونه می‌توان فهمید که بلوکی از حافظه‌ی نهان خالی است، یا داده‌ی موجود در آن معتبر نمی‌باشد؟
  - به ازای هر بلوک یک بیت به نام بیت اعتبار (valid bit) اختصاص داده می‌شود.

مثالی از حافظه‌ی نهان

- 8-blocks, 1 word/block, direct mapped

Index	V	Tag	Data
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	N		
111	N		

حالت اولیه

Word addr	Binary addr	Cache block
22	10 110	110

miss

مثالی از حافظه‌ی نهان (ادامه...)

Word addr	Binary addr	Hit/miss	Cache block
22	10 110	Miss	110

Index	V	Tag	Data
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	**Y**	**10**	**Mem[10110]**
111	N		

Word addr	Binary addr	Cache block
26	11 010	010

miss

مثالی از حافظه‌ی نهان (ادامه...)

Word addr	Binary addr	Hit/miss	Cache block
26	11 010	Miss	010

Index	V	Tag	Data
000	N		
001	N		
010	**Y**	**11**	**Mem[11010]**
011	N		
100	N		
101	N		
110	Y	10	Mem[10110]
111	N		

Word addr	Binary addr	Cache block	
22	10 110	110	hit
26	11 010	010	hit

مثالی از حافظه‌ی نهان (ادامه...)

Word addr	Binary addr	Hit/miss	Cache block
22	10 110	Hit	110
26	11 010	Hit	010

Index	V	Tag	Data
000	N		
001	N		
010	Y	11	Mem[11010]
011	N		
100	N		
101	N		
110	Y	10	Mem[10110]
111	N		

Word addr	Binary addr	Cache block
16	10 000	000
3	00 011	011
16	10 000	000

miss

miss

hit

مثالی از حافظه‌ی نهان (ادامه...)

Word addr	Binary addr	Hit/miss	Cache block
16	10 000	Miss	000
3	00 011	Miss	011
16	10 000	Hit	000

Index	V	Tag	Data
000	**Y**	**10**	**Mem[10000]**
001	N		
010	Y	11	Mem[11010]
011	**Y**	**00**	**Mem[00011]**
100	N		
101	N		
110	Y	10	Mem[10110]
111	N		

Word addr	Binary addr	Cache block
18	10 010	010

miss

مثالی از حافظه‌ی نهان (ادامه...)

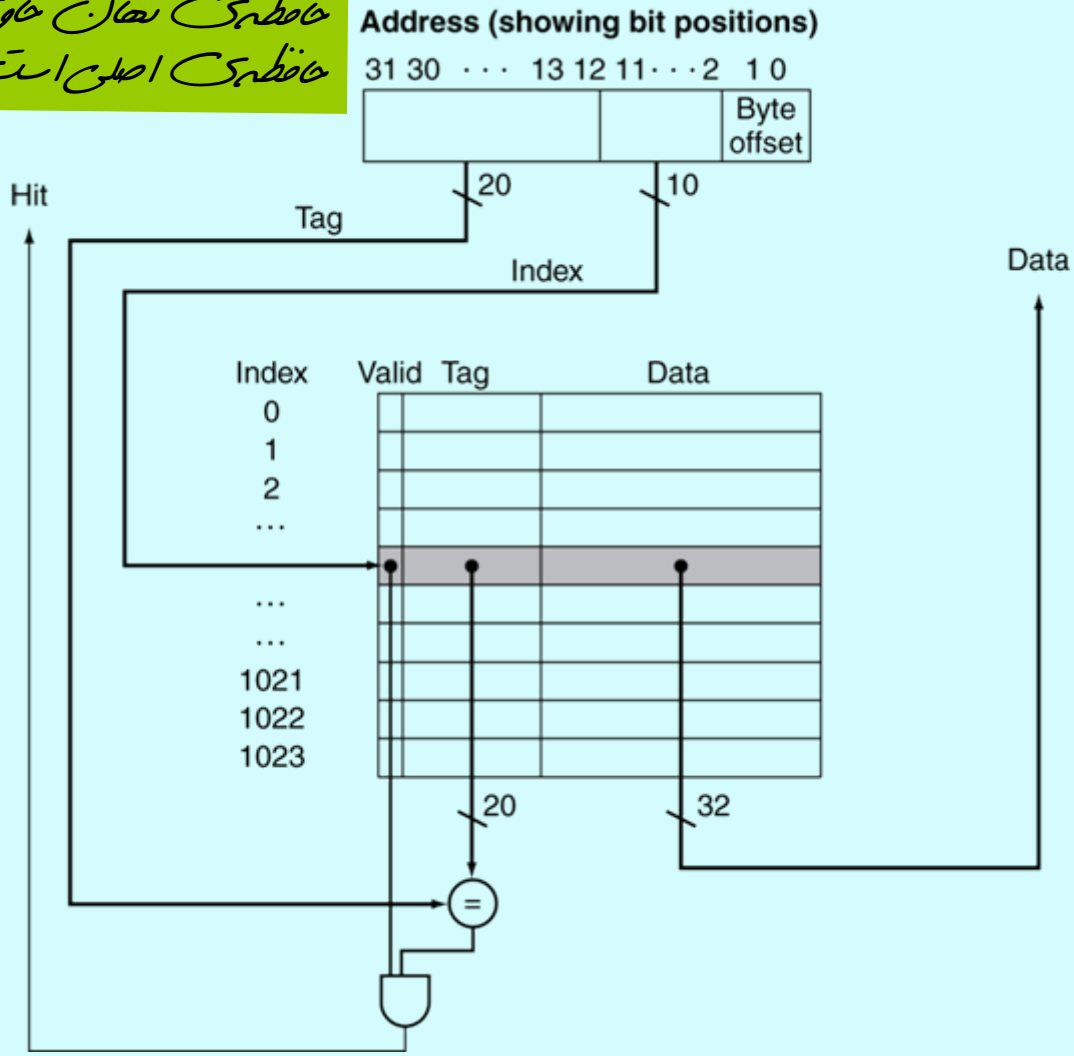
Word addr	Binary addr	Hit/miss	Cache block
18	10 010	Miss	010

Index	V	Tag	Data
000	Y	10	Mem[10000]
001	N		
010	**Y**	**10**	**Mem[10010]**
011	Y	00	Mem[00011]
100	N		
101	N		
110	Y	10	Mem[10110]
111	N		

زیر بخش مربوط به آدرس

شاخص (index) : بخش کم ارزش آدرس بلوک که بین حافظه‌ی اصلی و حافظه‌ی نهان مشترک است.

برچسب (tag) : مشخص می‌کند هر بلوک حافظه‌ی نهان حاوی کدامین بلوک از حافظه‌ی اصلی است



مثال

- یک سیستم با مشخصات زیر را در نظر بگیرید:
  - سی‌ودو بیت آدرس
  - حافظه‌ی نهان با نگاشت مستقیم
  - حافظه‌ی نهان با 2^n بلوک؛ شاخص با n بیت
  - اندازه‌ی هر بلوک 2^m کلمه

پرسش: اندازه‌ی بخش برچسب چند بیت خواهد بود؟

پاسخ: $32-(n+m+2)$

پرسش: تعداد کل بیت‌های حافظه‌ی مورد نیاز برای حافظه‌ی نهان را حساب کنید؟

پاسخ:

$$2^n \times (\text{block size} + \text{tag size} + \text{valid field size})$$

$$= 2^n \times (2^m \times 32 + (32-n-m-2) + 1) = 2^n \times (2^m \times 32 + 31 - n - m)$$

البته باید توجه داشت، معمولا حجم داده قابل ذخیره سازی به عنوان گنجایش ذکر می‌شود