



دانشکده مهندسی کامپیوتر
دانشگاه صنعتی امیرکبیر



آشنایی با حافظه‌ها



مقدمه



K در حافظه ها 1024 است نه 1000

- حافظه یکی از ملاحظات اصلی در طراحی سیستم های مبتنی بر ریزپردازنده است
- حافظه به وسیله ای می گویند که بتواند اطلاعات باینری را در خود ذخیره کند.
- انواع کاملاً متفاوتی از حافظه وجود دارد.

• چند پارامتر اساسی و مطرح برای حافظه ها عبارتند از:

• فرآر یا غیر فرآر بودن حافظه و مانند آن

• میزان تراکم حافظه (حجم حافظه)

• قابلیت برنامه ریزی مجدد

• سرعت خواندن یا نوشتن داده در حافظه

• هزینه



مفاهیم پایه در حافظه‌ها



• دستورات برنامه معمولاً در یک نوع حافظه (حافظه برنامه) نگهداری می‌شوند

• داده‌هایی که در حین انجام محاسبات به وجود می‌آیند در نوع دیگری از حافظه (حافظه داده) ذخیره می‌شوند.

• داده‌ها در حافظه ذخیره می‌شوند و همه بخش‌های دیگر از این داده‌ها استفاده می‌کنند.

• برای انتخاب نوع مناسب حافظه برای داده‌های مختلفی که در سیستم با آن سرو کار داریم، لازم است انواع حافظه‌های موجود را بشناسیم.



حافظه‌های فرار و غیرفرار



- حافظه فرار: حافظه‌ای است که بعد از قطع برق داده خود را از دست می‌دهد.
- حافظه غیرفرار: حافظه‌ای است که در نبود برق هم اطلاعات خود را حفظ می‌کند.



حافظه‌های فرار و غیرفرار



- حجم حافظه غیرفرار می‌تواند بسیار کم باشد مثلاً در کامپیوترها، اگر سیستم عامل و برنامه‌ها در یک حافظه انبوه ذخیره شده باشند، حافظه غیرفرار حاوی برنامه کوچکی است که سیستم عامل را از حافظه انبوه جانبی می‌خواند و در حافظه اصلی خود قرار می‌دهد.



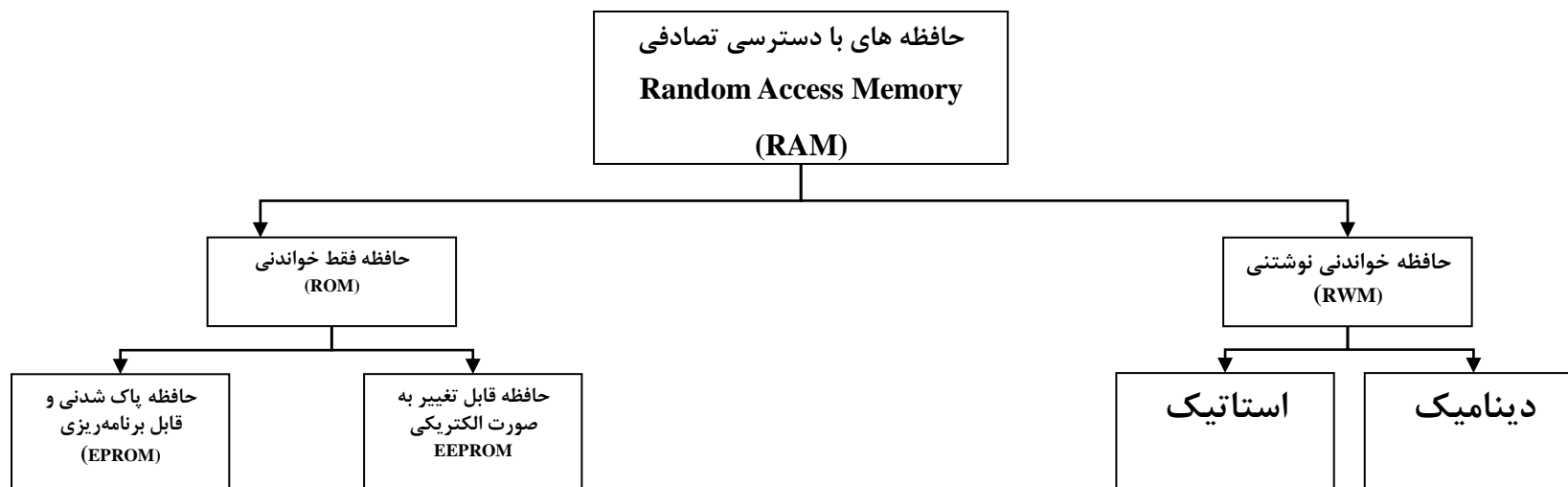


طبقه‌بندی حافظه‌ها



Sequential Access Memory
Random Access Memory

Memories



RAM ای که در کامپیوترها استفاده می‌شود در واقع RWM است (RAM خیل اصطلاح کلرای هست ROM هم یک نوع حافظه RAM است)



حافظه‌ها با دسترسی تصادفی



- حافظه‌های با دسترسی تصادفی به دو دسته زیر تقسیم می‌شوند:

- حافظه‌های فقط خواندنی ROM

- حافظه‌های نوشتنی-خواندنی RWM

- بخش مهمی از حافظه هر سیستم مبتنی بر ریزپردازنده، حافظه ROM است.

- داده‌های موجود در این حافظه در حین اجرای برنامه فقط خوانده می‌شوند و تغییری نمی‌یابند.

- این نوع حافظه عموماً از نوع غیرفرار است.

- دستوراتی که ریزپردازنده باید انجام دهد در این حافظه قرار می‌گیرد.

- ROM: Read Only Memory
- RWM: Read Write Memory



حافظه‌های ROM



• حافظه‌های ROM را می‌توان برنامه‌ریزی و سپس خواند. این حافظه‌ها توسط ریزپردازنده قابل نوشتن نیستند.

• این گروه خود به دو زیر گروه تقسیم می‌شود:

• حافظه فقط خواندنی قابل برنامه‌ریزی EPROM

• حافظه فقط خواندنی قابل تغییر برنامه‌ریزی و پاک شدن به صورت الکتریکی EEPROM

EPROM: Erasable Programmable ROM

EEPROM : Electrically Erasable Programmable ROM



حافظه EPROM



- این حافظه غیرفرار است.

- برنامه‌ریزی EPROM به صورت الکتریکی انجام می‌شود.

- این کار با ارسال یک پالس کوتاه با دامنه ولتاژ مناسب بر روی gate ترانزیستور FET موجود در محل سلول حافظه صورت می‌گیرد که با محبوس شدن شارژ الکتریکی در کانال بین source و drain این ترانزیستور، آن سلول حافظه در منطق 0 قرار می‌گیرد.

- کار برنامه‌ریزی سلول‌های حافظه به روش فوق توسط دستگاهی بنام EPROM Programmer انجام می‌شود.

- در EPROM برای پاک کردن محتوای یکی از خانه‌های حافظه لازم است که کل محتوای آن حافظه پاک شود. ✓

- برای پاک کردن لازم است که حافظه از سیستم جدا شود و در معرض اشعه فرابنفش قرار گیرد تا محتوایش پاک شود.



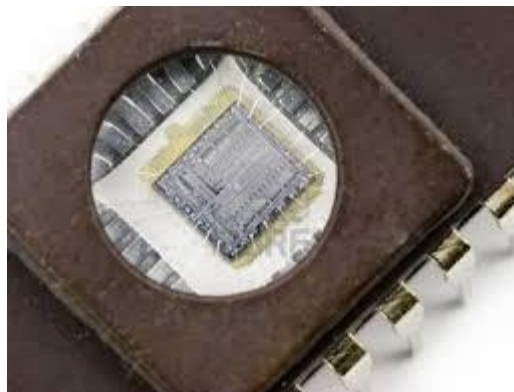
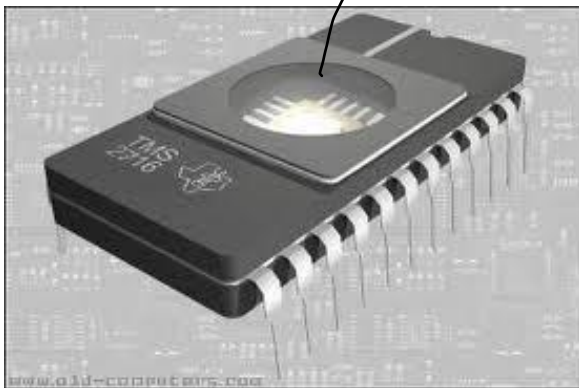
حافظه EPROM



• نمونه‌هایی از حافظه‌های EPROM

برای پاک کردن حافظه EPROM از اسلحه مادر برای بنفش استفاده می‌کنند

از اینجای نور را می‌تابانند



حافظه پاک تمام بیت هایش یک است

عمر حافظه EPROM محدود است



حافظه EPROM

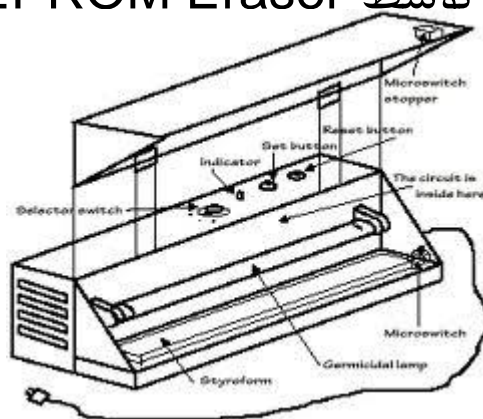


• فرآیند برنامه‌ریزی و پاک کردن در این حافظه، به لایه عایقی که در سلول‌های آن به کار رفته آسیب می‌رساند و این امر نهایتاً موجب تخریب سلول می‌شود و دفعات پاک کردن و برنامه‌ریزی مجدد را محدود می‌سازد.

• پاک کردن حافظه EPROM توسط EPROM Eraser انجام می‌شود.



EPROM Eraser



لامپ ماوراء بنفش برای پاک کردن EPROM



حافظه EEPROM

EEPROM : Electrically Erasable Programmable ROM



• این حافظه غیرفرار است.

• حافظه فقط خواندنی قابل پاک شدن به صورت الکتریکی و قابل برنامه ریزی مجدد به صورت الکتریکی، عنوانی است که به این نوع حافظه اختصاص داده شده است.

• این حافظه غیر فرار است و در کامپیوترها و دیگر وسایل دیجیتال برای ذخیره مقادیر محدودی داده به کار می رود.

• این حافظه ها در ظرفیت های گوناگونی از چند بایت تا چند صد کیلوبایت ارائه می شوند.

• EEPROM ها در تعداد دفعات نوشتن محدود هستند و این تعداد در انواع مدرن آن به حدود ۱۰۰۰۰۰۰ بار می رسد.

عمر EEPROM هم محدود است اما خیلی بیشتر از EPROM هست



حافظه‌های RWM



خرد CPU می‌تواند حافظه‌های RWM بنویسد و بخواند

حافظه‌های RWM: حافظه با دسترسی تصادفی خواندنی-نوشتنی

• اصطلاح "دسترسی تصادفی" یا Random Access Memory (RAM) که در مورد حافظه به کار می‌رود، بدین معنی است که هر مکان حافظه به صورت کاملاً مستقل از دیگر مکان‌های حافظه قابل دسترسی است.

• اصطلاح RAM به تنهایی نمی‌تواند تعیین کند که حافظه از نوع فقط خواندنی یا خواندنی-نوشتنی است.

• لیکن معمولاً اصطلاح RAM برای حافظه‌های خواندنی-نوشتنی RWM به کار می‌رود.

• حافظه‌های خواندنی-نوشتنی RWM محدودیتی در نوشتن مجدد در حافظه ندارند.

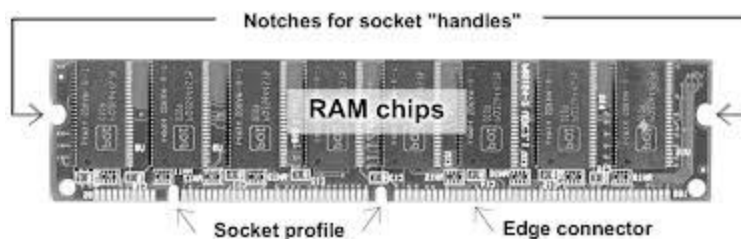
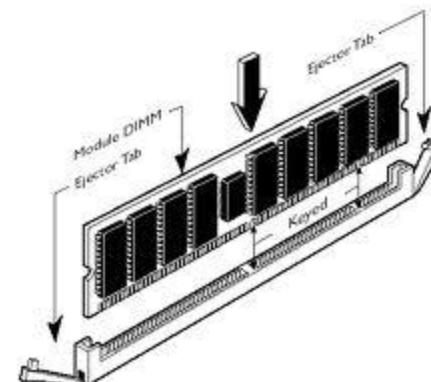


مثال‌هایی از حافظه‌های RWM

دانشکده مهندسی کامپیوتر
دانشگاه صنعتی امیرکبیر



حافظه با دسترسی تصادفی خواندنی-نوشتنی (حافظه‌های RWM)





زمان دسترسی و زمان سیکل حافظه



زمان دسترسی به حافظه و زمان سیکل حافظه:

• زمان "دسترس‌ی به حافظه" و زمان "سیکل حافظه" هر دو بیانگر سرعت یک حافظه است.

• فاصله زمانی بین لحظه‌ای که واحد حافظه دستور خواندن را دریافت می‌کند تا لحظه‌ای که داده مکان مورد نظر از حافظه در خروجی حافظه قرار می‌گیرد را **زمان دسترسی به حافظه** می‌گویند.

• زمان **سیکل حافظه** حداقل زمان ممکن بین دو عملیاتی است که با حافظه سر و کار دارند.



حافظه‌های استاتیک و دینامیک



حافظه‌های خواندنی-نوشتنی ممکن است استاتیک یا دینامیک باشند.

- حافظه‌های استاتیک نیاز به رفرش کردن ندارند. RWM $\left\{ \begin{array}{l} \text{استاتیک: تا زمانیکه تغذیه وصله به داخل می‌ماند} \\ \text{دینامیک: هر سلول نیاز دارد Refresh به بازماندن مقدار} \end{array} \right.$
لحظه‌ای: سرعت بیشتر هزینه کمتر (تعداد ترانزیستور کمتر)
- حافظه‌های دینامیک نیاز به رفرش (refresh) دارند.
- حافظه‌های استاتیک برای حجم‌های کوچک حافظه مناسب هستند.
- در حافظه‌های دینامیک مدارات مجتمع ارزان قیمت‌تری را به کار می‌برند.
- به دلیل نیاز به رفرش، حافظه‌های دینامیک به مدارات حمایتی برای عمل رفرش نیاز دارند.





حافظه‌های دینامیک



در هر سلول حافظه دینامیک یک خازن قرار دارد. باری که در خازن ذخیره شده است سطح منطقی داده ذخیره شده در سلول را تعیین می‌کند.

• به خاطر دشارژ شدن، خازن داده را فقط به مدت چند میلی ثانیه در خود نگه می‌دارد.

• بعد از این مدت لازم است که داده مجدداً در سلول نوشته شود. این عملیات را تازه کردن (Refresh) سلول گویند.

• سیستم‌های حافظه‌ای که از RAM دینامیک استفاده می‌کنند مدار خاصی برای این عملیات دارند که Refresh Logic نامیده می‌شود.

• این مدار به داده‌های حافظه به صورت سطری دسترسی می‌یابد و تمامی سلول‌های یک سطر را با هم Refresh می‌کند.

• این عملیات نباید با عملیات خواندن-نوشتن در حافظه که CPU انجام می‌دهد تداخل کند.

• مدار Refresh باید بداند که چه بخش‌هایی از حافظه Refresh نشده‌اند تا قبل از تخریب شدن داده‌های آن سلول‌ها، آنها را Refresh کند. ۱۷



حافظه‌ها



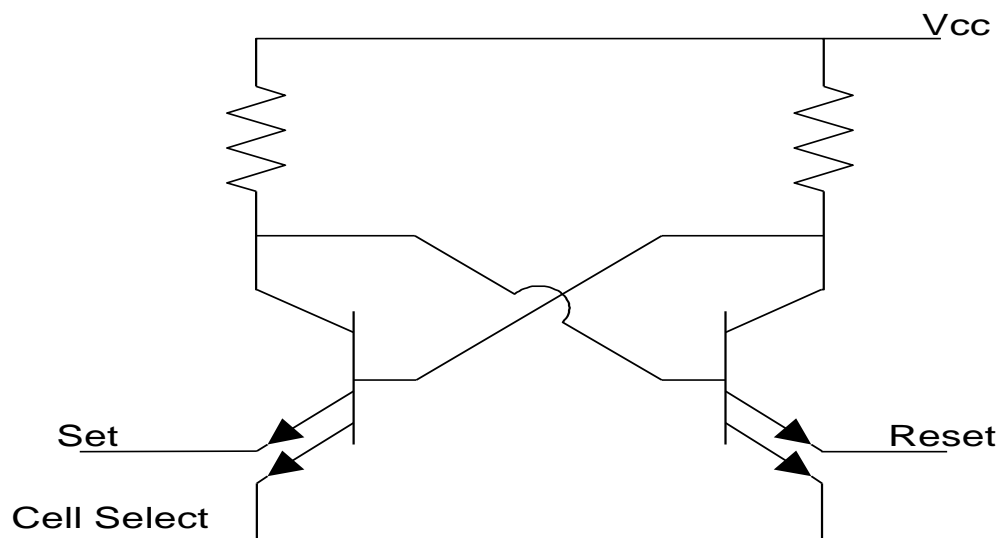
BJT : معرف بالا ، سرعت بالا

CMOS : معرف پایین ، سرعت کم

Memories based on technology

• شکل زیر شماتیک ساده‌ای از یک سلول حافظه bipolar را نشان می‌دهد.

• این سلول چیزی جز یک فلیپ فلاپ نیست که ورودی های Set و Reset دارد.





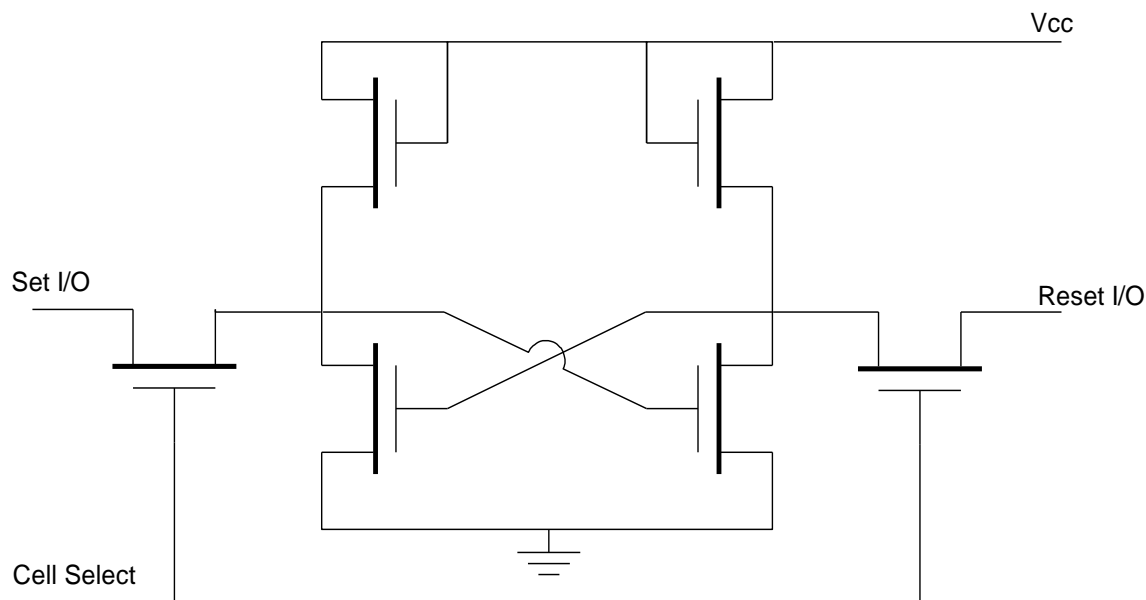
حافظه‌ها



- شکل زیر شمای ساده یک سلول حافظه استاتیک NMOS را نشان می‌دهد.

- این سلول نیز یک فلیپ فلاپ است.

- در این مدار ترانزیستور MOS با بایاس ثابت به عنوان بار در درین ترانزیستورهای دیگر به کار رفته است.





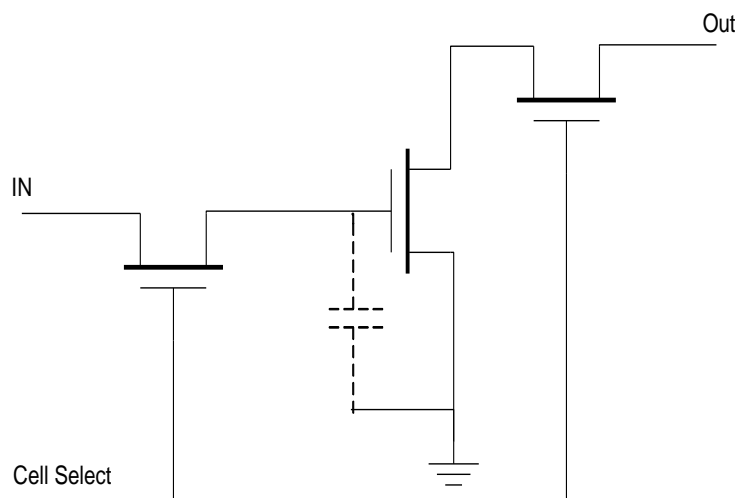
حافظه ها



حافظه های دینامیک

• سلول های حافظه دینامیک ساده تر از استاتیک هستند.

• این سلول کار حافظه استاتیک را با استفاده از تعداد ترانزیستورهای کمتری انجام می دهد و لذا بر روی یک سطح یکسان از تراشه، حجم حافظه بیشتری را فراهم می کند.

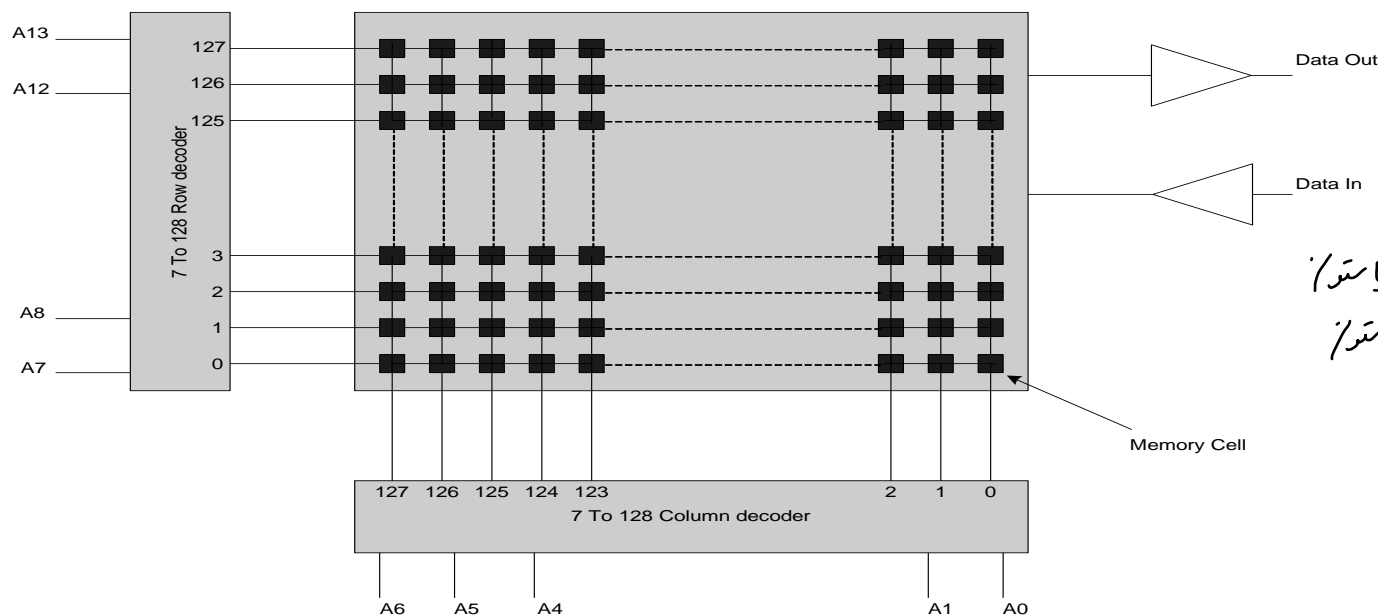




حافظه‌ها



- تعداد زیادی از این سلول‌های حافظه در ساختارهای ماتریسی قرار می‌گیرند و حافظه‌های بزرگ را ایجاد می‌کنند.
- شکل زیر یک تراشه ۱۶۳۸۴ بیتی را نشان می‌دهد.
- این بلوک دیاگرام برای هر دو تکنولوژی bipolar و MOS، استاتیک یا دینامیک یکسان است.
- این تراشه ۱۴ خط آدرس دارد که این خطوط به دیکودرهای سطر و ستون وصل می‌شوند. خطوط A0 تا A6 به دیکودر ستون و خطوط A7 تا A13 به دیکودر سطر وصل می‌شوند.



برای اینده $2^{14} \times 1 \text{ bit}$ حافظه داشته باشیم
اگر بخواهیم بجای 1 bit از 8 byte استفاده کنیم
باید ۸ تا از این ماتریس‌ها را در کنار هم
قرار دهیم
در حافظه‌های *dynamic* وقتی هر سطر یا ستون
انتخاب می‌شود تمام سطرهای دیگر سطر یا ستون
refresh می‌شوند



مثالی از یک حافظه استاتیک



دانشکده مهندسی کامپیوتر
دانشگاه صنعتی امیرکبیر

شکل زیر یک نمونه حافظه استاتیک از نوع MOS را نشان می‌دهد. در این تصویر یک حافظه SRAM با حجم 2Kbyte یا 16Kbit نشان داده شده است.

سلول‌های این حافظه به صورت کلمات هشت بیتی هستند که متناسب با سیستم‌های مبتنی بر ریزپردازنده انتخاب شده‌اند.

در این IC علاوه بر ۱۲ خط آدرس، هشت خط داده نیز موجود است که در حین عملیات خواندن یا نوشتن، ورودی یا خروجی داده هستند. ورودی کنترلی عملیات خواندن یا نوشتن را تعیین می‌کند.

- پایه‌های حافظه:
- خطوط (پایه‌های) آدرس (خطوط آدرس برای یک حافظه با ۲^{۱۱} کلمات)
 - خطوط داده (D_۰ - D_۷)
 - پایه فعال ساز حافظه (\overline{CE} = chip enable)
 - پایه فعال‌ساز خروجی (\overline{OE} = output enable)
 - مراحل خواندن از حافظه ROM:
 - $\overline{CE} = 0$ - گناختن آدرس در خطوط آدرس
 - $\overline{OE} = 0$ - نتیجه رگسترهای خوانده شده در پایه‌های D_۷ - D_۰ آماده هست

1	N/C	28	VCC
2	N/C	27	R/W
3	A7	26	N/C
4	A6	25	A8
5	A5	24	A9
6	A4	23	N/C
7	A3	22	OE
8	A2	21	A10
9	A1	20	CE
10	A0	19	D7
11	D0	18	D6
12	D1	17	D5
13	D2	16	D4
14	GND	15	D3

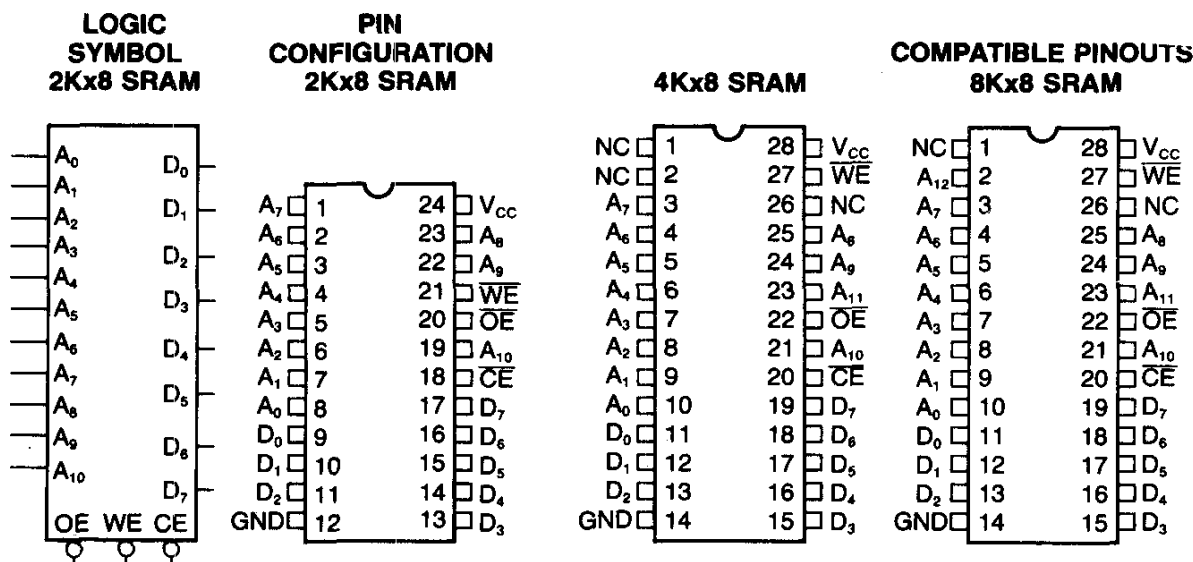
R/W

فرانز از حافظه R/W به خواندن از RAM است

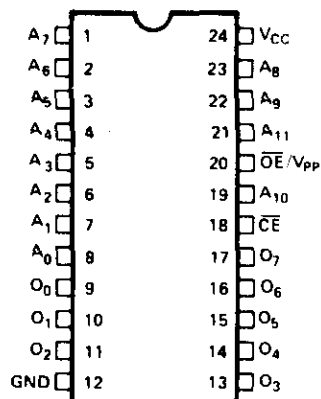


مثال‌هایی از انواع حافظه EPROM و SRAM

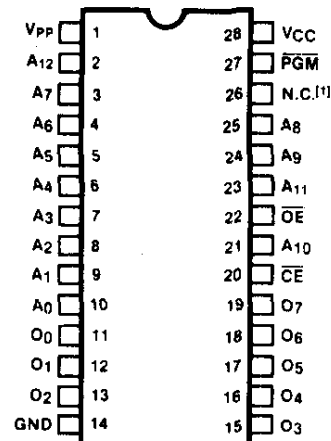
دانشکده مهندسی کامپیوتر
دانشگاه صنعتی امیرکبیر



2K byte ← **2732A**
PIN CONFIGURATION



2764
PIN CONFIGURATION → 4K byte





پایه Chip Enable



ورودی Chip Enable برای انتخاب تراشه است که اگر در سطح منطقی 1 قرار بگیرد، خروجی تراشه در حالت امپدانس بالا قرار می گیرد.

این پایه کنترلی برای اتصال چند تراشه به هم و ایجاد حافظه های با حجم بالاتر مناسب است. چون با به کار بردن آن، خروجی تراشه ها را می توان به هم وصل کرد.

به عنوان مثال برای ایجاد یک حافظه 16KB با استفاده از دو تراشه حافظه 8KB، کفایت ۱۳ خط آدرس A0 تا A12 را به هریک از دو تراشه حافظه 8KB وصل کنیم و خط A13 آدرس را به ورودی مدار انتخاب تراشه (دیکودر) وصل کنیم.

با این کار به ازای نیمی از آدرس ها یکی از تراشه ها و به ازای نیمه دیگر آدرس ها تراشه دوم عمل می کند.



حافظه FLASH



- این تکنولوژی در کارت‌های حافظه و فلش درایوهای USB برای ذخیره کردن اطلاعات و انتقال آن بین کامپیوترها و دیگر وسایل دیجیتال به کار گرفته می‌شود.
- امروزه در بسیاری از تراشه‌ها از جمله ریزپردازنده‌ها و میکروکنترلرها از این نوع حافظه استفاده می‌شود.
- حافظه فلش اطلاعات را در آرایه‌ای از ترانزیستورها با گیت شناور (Floating Gate) ذخیره می‌کند.
- دسترسی سریع به اطلاعات موجود در حافظه فلش امکان‌پذیر است (البته نه به سرعت حافظه‌های RWM).
- استقامت مکانیکی بیشتری نسبت به دیسک‌های سخت دارند، مقاومت نسبت به فشار زیاد هوا، دمای بالا و حتی غرق شدن در آب و ... همگی توجیهی بر محبوبیت این حافظه در دستگاه‌هایی است که نیاز به حافظه دارند و از باتری برای تامین انرژی استفاده می‌کنند.

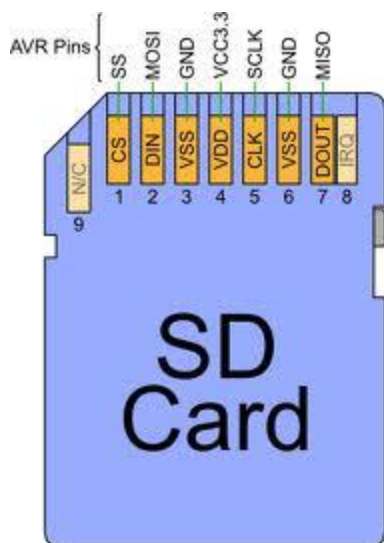


دانشکده مهندسی کامپیوتر
دانشگاه صنعتی امیرکبیر

انواع حافظه FLASH



wiseGEEK





حافظه FLASH



- دو تکنولوژی اصلی در ساخت حافظه‌های فلش NOR و NAND هستند.

- حافظه NAND برای ذخیره حجم زیاد داده مناسب هستند.

- پاک کردن و خواندن حافظه NAND بصورت بلوکی انجام می‌شود.

- حافظه NOR به دلیل سرعت زیاد دسترسی به اطلاعات، برای ذخیره کدهای اجرایی برنامه‌ها و فراخوانی آنها برای اجرا مناسب هستند.

- پاک کردن حافظه NOR بصورت بلوکی و خواندن آنها بصورت بایتی انجام می‌شود.

- محدودیت دیگر برای حافظه فلش، تعداد دفعات پاک کردن حافظه است که در مورد فلش‌های تجاری که امروزه به کار گرفته می‌شوند این محدودیت تا یک میلیون بار است.

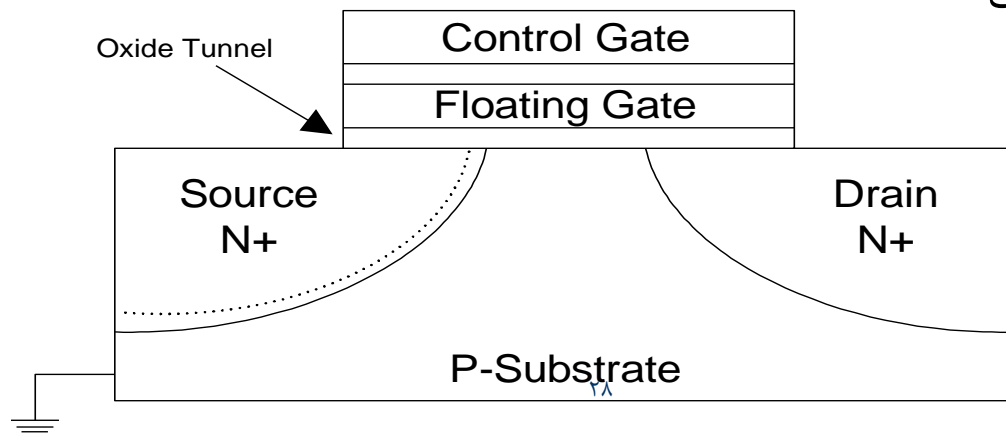


• در فلش‌های نوع NOR هر سلول شبیه یک MOSFET استاندارد است، به جز اینکه دو گیت دارد (شکل زیر).

• عبور یا عدم عبور جریان بین Source و Drain در ترانزیستور یک سلول حافظه را می‌توان به صفر یا یک بودن آن سلول ترجمه کرد.

• برنامه‌ریزی یک سلول بدین صورت است که جریانی از الکترون‌ها از پایه سورس به درین ترانزیستور راه‌اندازی می‌شود، سپس ولتاژ زیادی بر پایه گیت کنترلی قرار می‌گیرد تا یک میدان الکتریکی به اندازه کافی بزرگ فراهم شود.

• این میدان موجب مکش الکترون‌ها به سمت بالا، درون گیت شناور می‌شود. این فرآیند را **تزریق الکترون‌های داغ** می‌نامند.





حافظه FLASH



• برای پاک کردن یک سلول NOR، ولتاژ تفاضلی بزرگی بین سورس و گیت کنترلی ایجاد می شود.

• یعنی برای پاک کردن یک بایت داده از یک بلوک، تمام داده های آن بلوک باید پاک شوند.

• بعد از پاک شدن یک بلوک، محتوای تمامی سلول ها 1 خواهد بود.

• برای مقدار صفر باید آن را درون سلول ایجاد کرد.

• زمانی که یک سلول مقدار صفر را دریافت کرد، دیگر قابل بازگشت به مقدار یک نیست، مگر اینکه محتوای کل آن بلوک پاک شود.