

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

# طراحی سیستمهای ریزپردازنده

دانشکده مهندسی برق

دانشگاه صنعتی شاهرود

حسین خسروی

۱۴۰۰

# ارزشیابی

## ◀ کتاب درسی

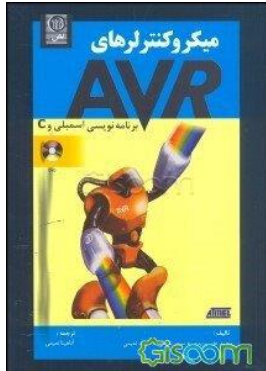
◀ میکروکنترلرهای AVR برنامه نویسی اسمبلی و C

◀ محمدعلی مزیدی

◀ تمرینها: حدود ۶ نمره

◀ میان ترم: حدود ۶ نمره

◀ پایان ترم: حدود ۸ نمره



# میکروکنترلر چیست؟

◀ یک مدار مجتمع که برای انجام یک کار خاص، برنامه ریزی می شود

◀ میکروکنترلر در واقع یک مینی کامپیوتر است



Atmel AVR



AVR



ATX Mega



ATmega 328P



PIC 16F877A



8051



Arduino



ARM

# میکروکنترلرها در کجا استفاده می شوند

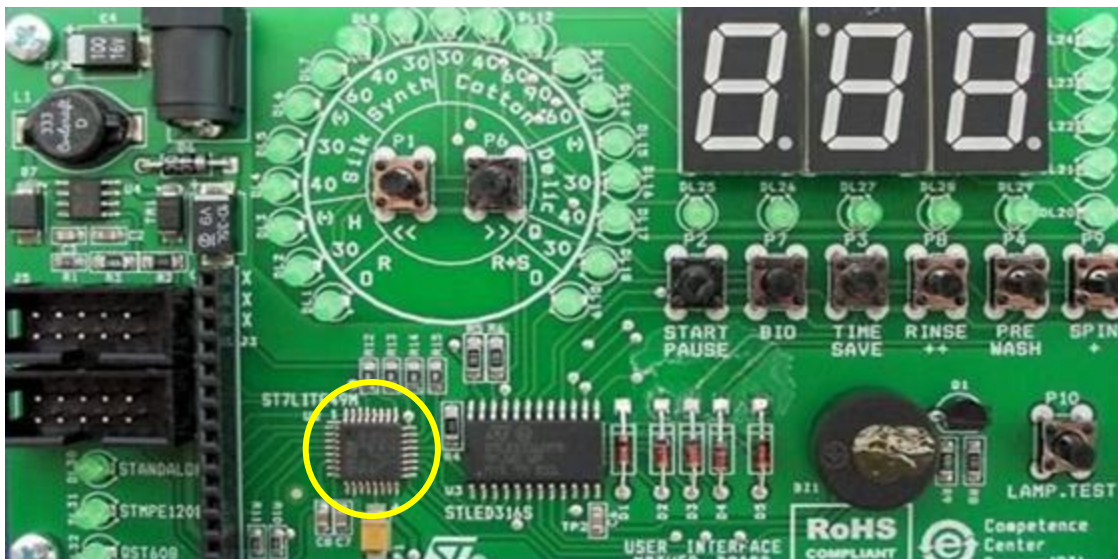
میکروکنترلرها در بسیاری از وسایل خانگی، گجت‌های هوشمند و سایر وسایل الکترونیکی وجود دارند



# نمونه ای از کاربردها

◀ برد ماشین لباسشویی

◀ میکروی هشت بیتی ST7LITE49M با تنها ۴ کیلوبایت حافظه کد





# نمونه ای از کاربردها - برد کنترلر سمند

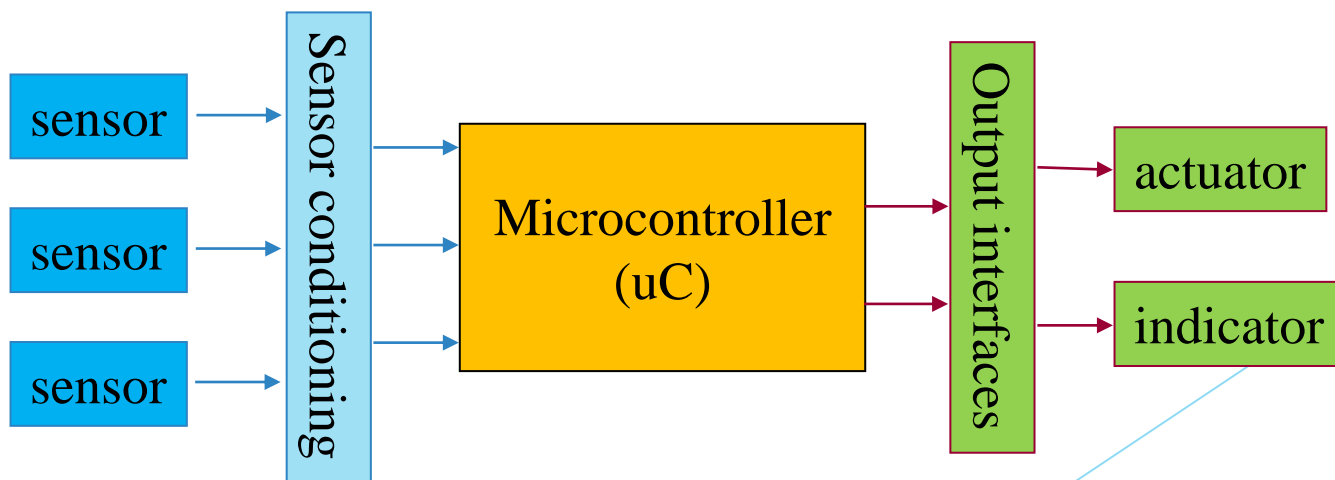
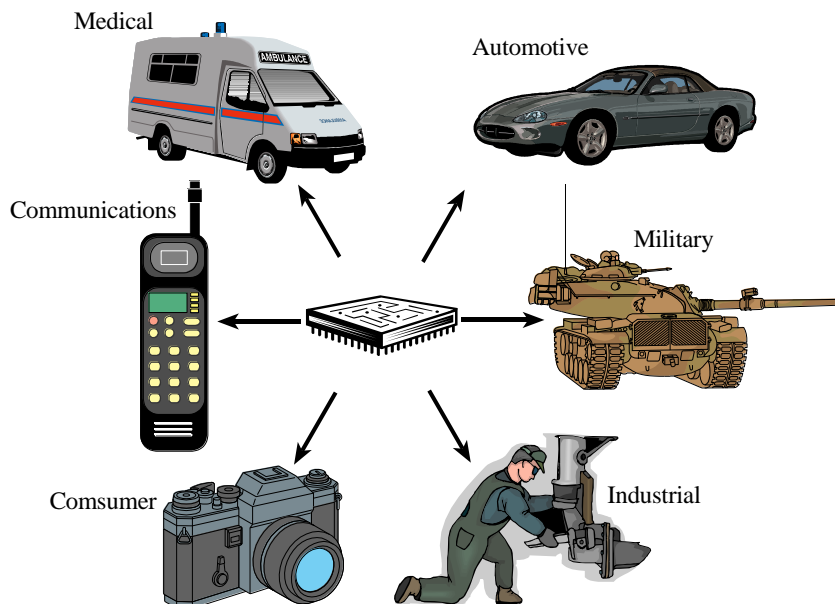


- |  |  |
|--|--|
| 1 - ST10F275-CGE - میکروکنترلر ۱۶ بیتی با حافظه فلش داخلی ۷۶۸ کیلوبایت     | 10 - 30536 - آیسی مولتی راه انداز انژکتورهای گاز و گیج گاز |
| 2 - کریستال ساعت داخلی   | 11 - خازن صافی برق ۱۲ ولت                                  |
| 3 - نوسان ساز کریستالی اصلی میکروکنترلر                                    | 12 - 30424 - آیسی پردازنده سیگنال سنسور ناک                |
| 4 - 74HC08D - دو گیت اند دو ورودی  | 13 - LM2903 - تقویت کننده سیگنال تفاضلی دور موتور          |
| 5 - خازن صافی برای ولتاژ تغذیه ۵ ولت                                       | 14 - TLE7209 - آیسی راه انداز دریچه گاز برقی               |
| 6 - 30522 - واسط سنسورهای اکسیژن   | 15 - DALE - نوسان گیر پرازیت کوئل ها                       |
| 7 - B82793 - بخشی از مدار محافظتی خطوط شبکه کن                             | 16 - 07096 - راه انداز کوئل ۳                              |
| 8 - 30639 - رگولاتور ولتی و محرک منفی رله های پمپ بنزین و اصلی و گیج بنزین | 17 - 07096 - راه انداز کوئل ۲                              |
| 9 - 30536 - گرم کن سنسور اکسیژن و منفی شیربرقی                             | 18 - 07096 - راه انداز کوئل ۴                              |
| CVVT   | 19 - 07096 - راه انداز کوئل ۱                              |

راهنمای تعمیرکار روی برد بوش ۷۴۹ سمند و دنا با  
موتور EF7

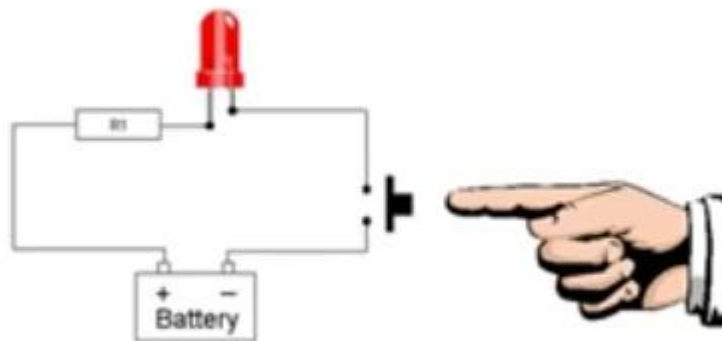
# سیستمهای نهفته - Embedded Systems

سیستمهایی که نیاز به  
رایانه جداگانه ندارند.  
به جای آن یک میکرو  
در درون خود دارند



# کجا باید از میکروکنترلر استفاده کنیم

◀ هر کجا که بخواهیم هوشمندی به سیستم اضافه کنیم







Vacuum Tube  
1939



Transistor  
1947



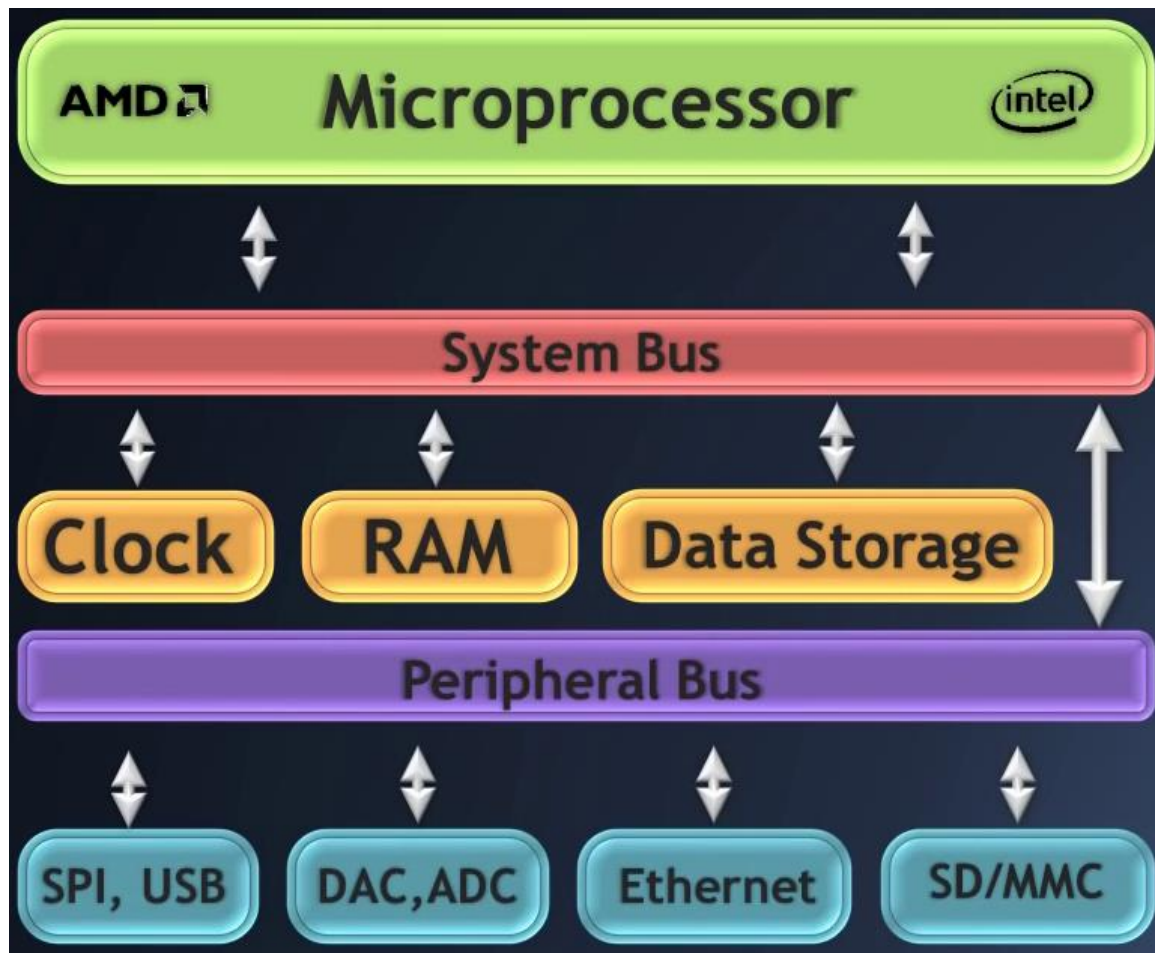
Logic Gate  
1960



Microcontroller  
1971

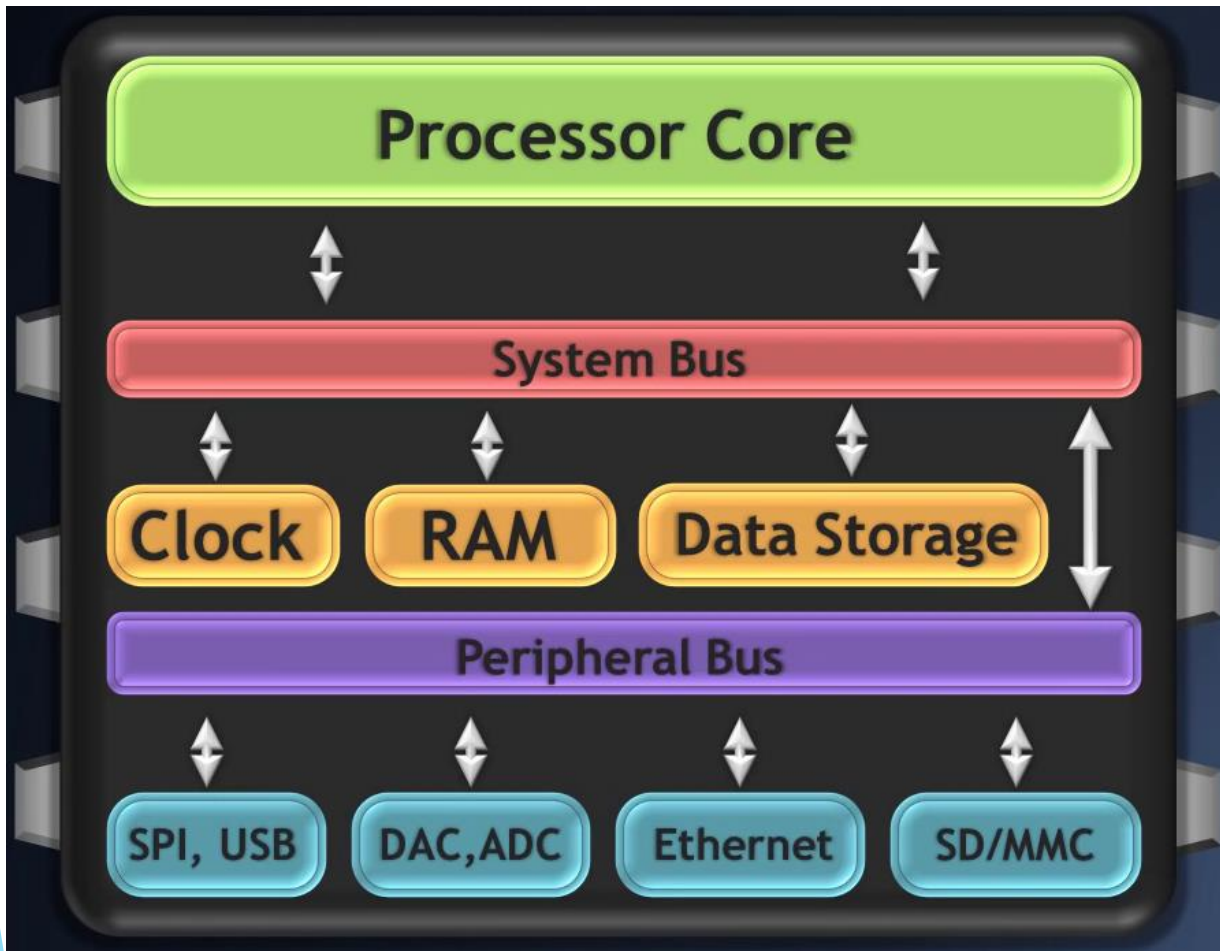
# ریزپردازنده

◀ ریزپردازنده بدون مادربرد و قطعات جانبی، قابل استفاده نیست



# میکروکنترلر

◀ میکروکنترلر، بسیاری از تراشه های مورد نیاز مثل RAM, ROM, Clock, ADC, I2C و مانند اینها را درون خود داشته و نیازی به مادربرد ندارد.

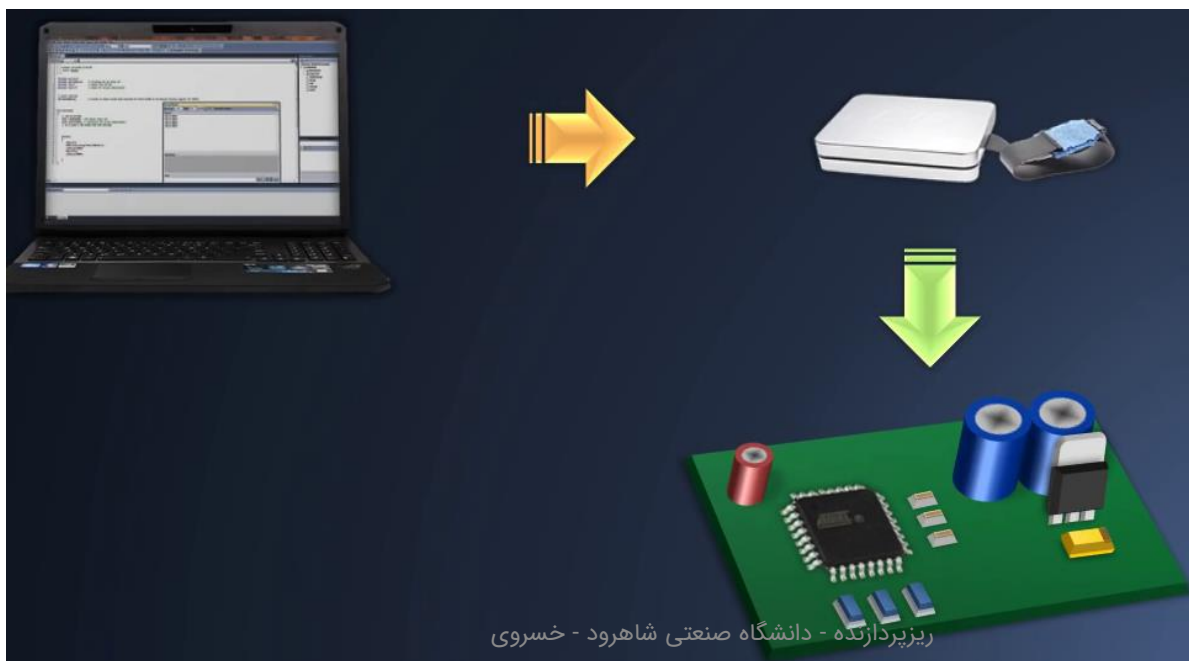


# اساس عملکرد میکروکنترلر

- ◀ میکروکنترلر برای کاربردهای خاص استفاده می شود
- ◀ نیاز نیست خیلی مشخصات قوی داشته باشند چرا که بسیاری از کاربردها سرعت و حافظه کمی نیاز دارند
- ◀ هوشمندی میکرو به کدی است که شما می نویسید
- ◀ شاید بتوان گفت ۸۰ درصد دانش میکروکنترلر کدنویسی است
- ◀ میکرو همانقدر مفید است که کد شما به آن دیکته می کند. اگر شما می خواهید با عبور دما از یک حد آستانه، چراغ قرمزی روشن شود، این کار باید از طریق کد بیان شود.

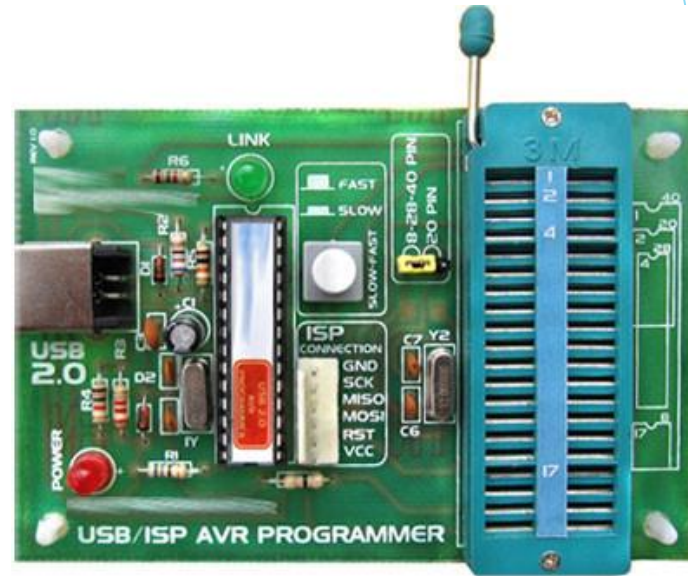
# برنامه نویسی میکرو

- ▶ کد لازم در یک محیط توسعه یکپارچه (IDE) به زبانهایی مثل C یا اسمبلی نوشته می شود
- ▶ بعد از خطایابی توسط کامپایلر و اصلاح خطاهای احتمالی، کد به فرمت باینری (کد ماشین یا کد هگز) مناسب برای میکرو تبدیل می شود
- ▶ در نهایت توسط دستگاه برنامه ریز (Programmer) کد هگز روی میکروکنترلر ریخته می شود. برخی بردها مثل آردینو برنامه ریز سرخود دارند و نیاز به دستگاه جانبی نیست





# دو نمونه برنامه ریز (AVR و آردینو)



# بسته بندی میکروکنترلر



## DIP

(Dual Inline Package)

Through hole

8 pins

9mm x 6mm

0.15pins/mm<sup>2</sup>



## SOIC

(Small Outline IC)

Surface Mount

18 pins

11mm x 7mm

0.23pins/mm<sup>2</sup>



## QFP

(Quad Flat Package)

Surface Mount

32 pins

7mm x 7mm

0.65pins/mm<sup>2</sup>



## BGA

(Ball Grid Array)

Surface Mount

100 pins

6mm x 6mm

2.78pins/mm<sup>2</sup>

# بسته بندی میکروکنترلر



## DIP

(Dual Inline Package)

Through hole

8 pins

9mm x 6mm

0.15pins/mm<sup>2</sup>



## SOIC

(Small Outline IC)

Surface Mount

18 pins

11mm x 7mm

0.23pins/mm<sup>2</sup>



## QFP

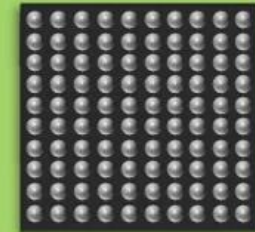
(Quad Flat Package)

Surface Mount

32 pins

7mm x 7mm

0.65pins/mm<sup>2</sup>



## BGA

(Ball Grid Array)

Surface Mount

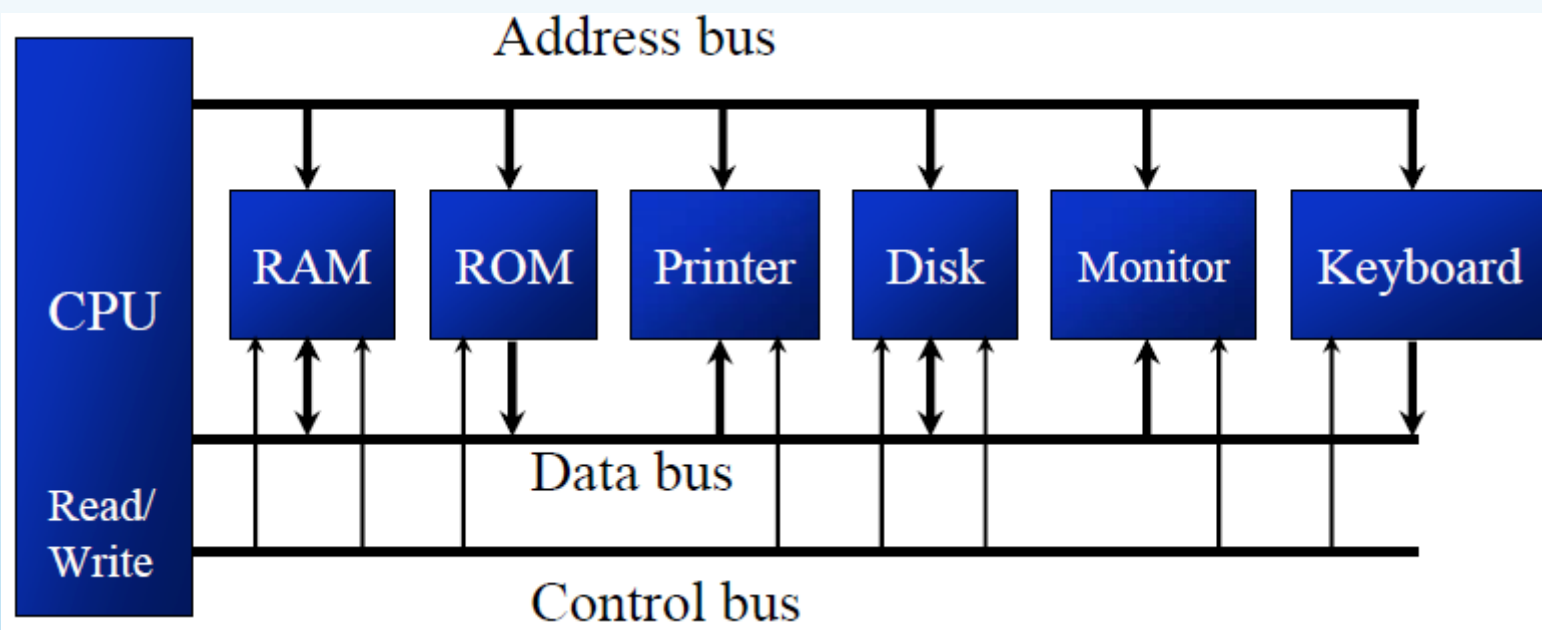
100 pins

6mm x 6mm

2.78pins/mm<sup>2</sup>

## □ ساختار ساده یک کامپیوتر

■ پردازنده مرکزی، حافظه‌های RAM و ROM، ابزارهای ورودی خروجی، گذرگاههای آدرس و داده



# انواع حافظه کد (ROM)

□ PROM یا OTP : تنها یک بار قابلیت نوشته شدن دارد.

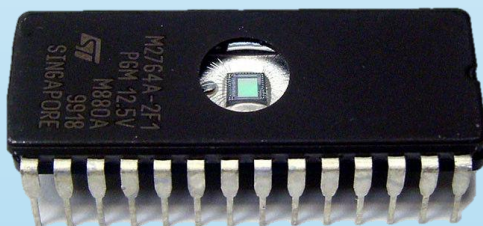
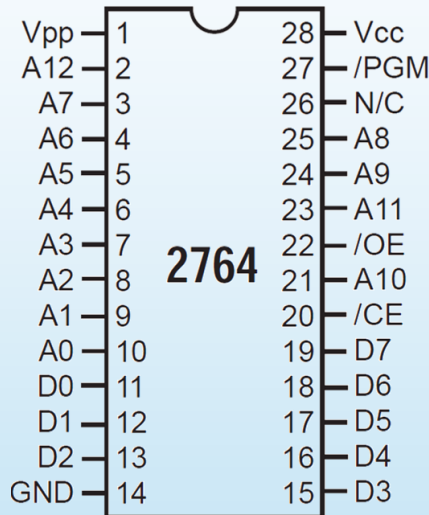
□ EPROM یا UV-EPROM

■ قابلیت پاک شدن دارد؛ ۲۰ دقیقه در مجاورت UV

■ نیاز به دو دستگاه programmer و پاک کننده

■ تعداد دفعات نوشتن و پاک کردن: هزار بار

■ مثال: ۲۷۶۴ (ظرفیت ۶۴ کیلوبیت)



□ EEPROM

■ قابلیت پاک شدن الکتریکی

■ پاک کردن بایتهای دلخواه

■ قابلیت برنامه ریزی روی برد (نیاز به مدار برنامه ریز)

■ تعداد دفعات نوشتن و پاک کردن: ۱۰۰ هزار بار



# انواع حافظه کد (ROM) - ادامه

## Flash

- EEPROM سریع
- پاک شدن کل حافظه در کمتر از یک ثانیه
- تعداد دفعات نوشتن و پاک کردن: ۱۰۰ هزار بار
- مورد استفاده در بایاس کامپیوتر و حافظه برنامه میکروهای AVR
- در آینده جایگزین دیسکهای سخت خواهد شد

## Mask ROM

- توسط تولید کننده IC برنامه ریزی می شود
- مناسب برای تولید انبوه
- پس از نهایی شدن کد، آن را به سازنده IC داده تا تولید کند.

 نکته: تمام حافظه های ROM، ۸ پایه برای داده دارند.

# انواع حافظه داده (RAM)

## □ حافظه ایستا یا SRAM

- از فلیپ فلاپ ساخته شده و نیازی به تازه سازی ندارد.
- حجم بیشتری روی تراشه اشغال می کند ( هر سلول ۴ تا ۶ ترانزیستور)

## □ حافظه پویا یا DRAM

- از خازن برای سلولهای حافظه استفاده می کند. **نیاز به تازه سازی دارد.**
- مزایا: ظرفیت بالا، هزینه کمتر و مصرف برق کمتر

## □ حافظه غیرفرار NV-RAM

- مزایای RAM و ROM را شامل می شود
- سلولهای SRAM با تکنولوژی CMOS (کم مصرف)
- باتری لیتیوم داخلی (قابلیت نگهداری اطلاعات تا ۱۰ سال)
- کنترل هوشمند جریان پین VCC جهت سوئیچ کردن بین منبع داخلی و خارجی

□ نکته: تعداد دفعات نوشتن و خواندن در RAM **بینهایت** است.

# نمونه ای از SRAM

## □ مثالی از حافظه ایستا: ۶۱۱۶

■ پینهای A0-A10 امکان آدرس دهی 2k خانه را فراهم می کنند.

■ پینهای D0-D7 بیانگر ۸ بیتی بودن خانه های حافظه است.

■ ساختار حافظه: 2kx8

■ (Write Enable) WE

□ برای نوشتن روی حافظه

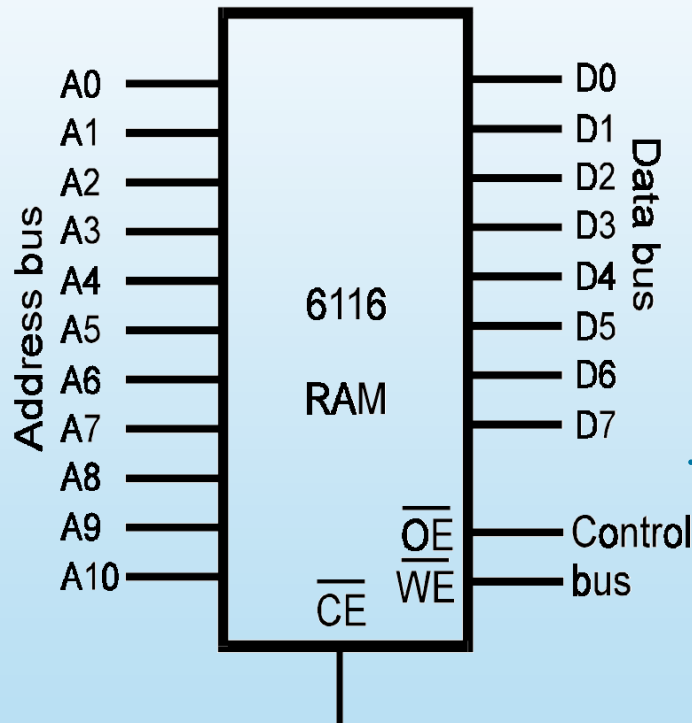
■ (Output Enable) OE

□ برای خواندن از حافظه

■ پینهای کنترلی WE و OE صفر-فعال هستند.

■ (Chip Select) CS

□ برای انتخاب تراشه



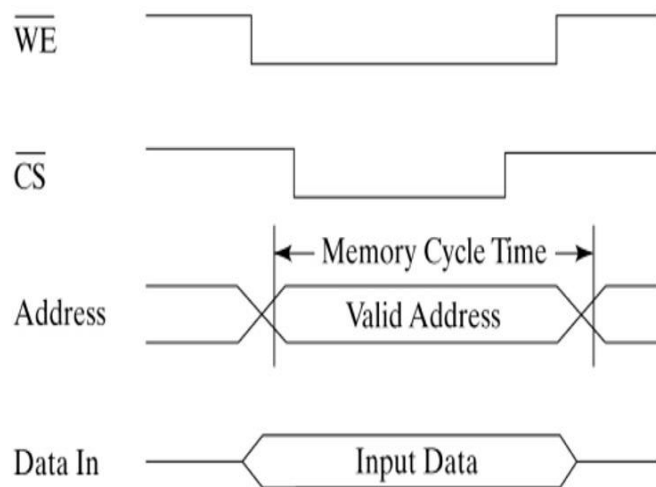
# نمودار زمانبندی نوشتن در SRAM - حالت ساده

▶ پایه های A0-A10 را فراهم کنید

▶ پایه CS را فعال کنید

▶ WE را فعال کنید

▶ داده ها را در D0-D7 قرار دهید

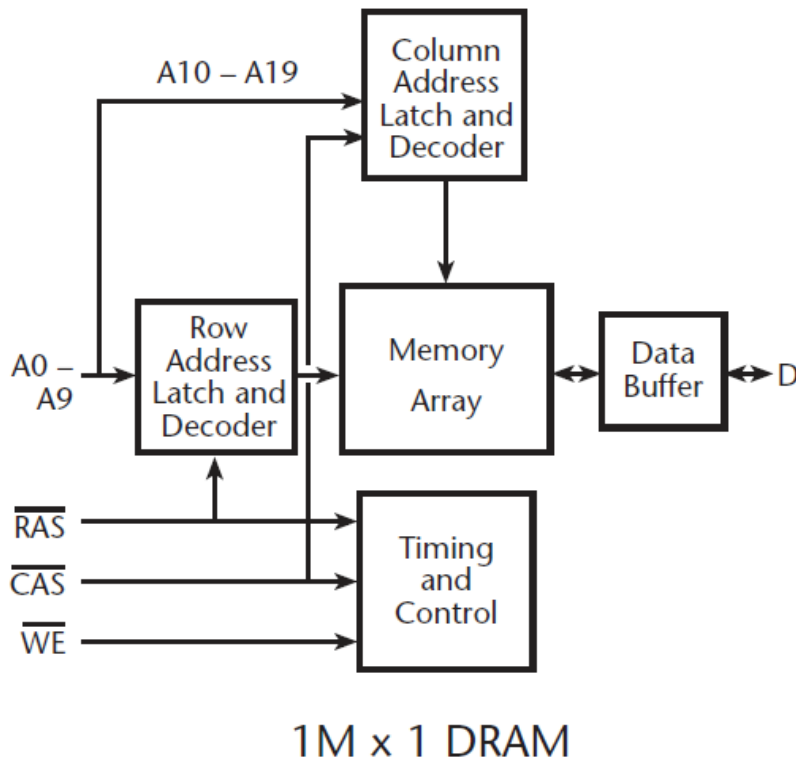


## تعداد زیاد پینهای آدرس

■ استفاده از تکنیک چیدمان ماتریسی

■ دسترسی به خانه های حافظه با استفاده از RAS و CAS

■ تعداد پینهای لازم برابر است با نصف + ۲ پین برای RAS و CAS





- ❑  $975 = 9 * 10^2 + 7 * 10^1 + 5 * 10^0$
- ❑  $1011 = 1 * 2^3 + 0 + 1 * 2^1 + 1 * 2^0 = 11$
- ❑  $1111 = 1 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * 2^0 = 15$
- ❑  $0x2F = 2 * 16 + 15 = 47$
- ❑  $0xFF = 15 * 16 + 15 = 255$
- ❑  $0x2F = 0b00101111$
- ❑  $0xF5A9 = 0b1111010110101001$

# کد گشایی آدرس حافظه

- هر ابزار جانبی مانند حافظه‌های خارجی باید محدوده‌ی آدرس خاص خود را داشته باشند.
- مثلا اگر دو حافظه ROM خارجی با ساختار 2kx8 داریم می‌توانیم یکی را در محدوده‌ی آدرس 0X0000-0X07FF و دیگری را در محدوده‌ی آدرس 0X5000-0X57FF جاگذاری کنیم.
- به این عمل نگاشت حافظه (memory mapping) گویند.
- برای این کار پینهای کم ارزش گذرگاه آدرس را به خطوط آدرس حافظه ROM متصل کرده و پینهای باقیمانده را با استفاده از گیت‌های منطقی یا مالتی پلکسر به آدرس مورد علاقه نگاشت می‌دهیم.
- پینهای باقیمانده باید به CS یا CE حافظه وصل شوند.

□ پایه های کنترلی پردازنده، (RD و WR) به پایه های متناظر از حافظه متصل می شوند (OE و WE).

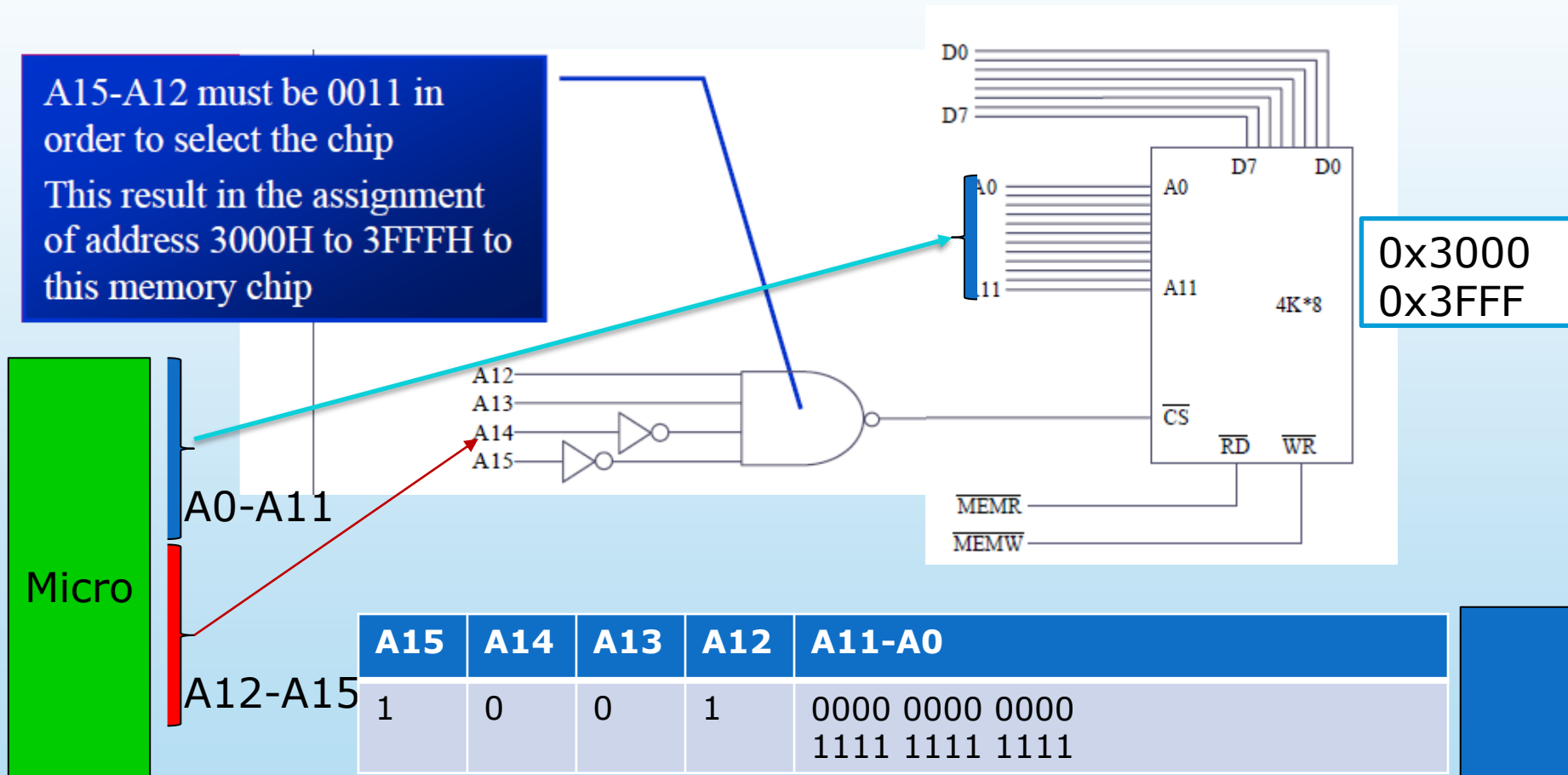
- RD: Read
- WR: Write
- OE: Output Enable
- WE: Write Enable

□ دو روش متداول برای کدگشایی:

- استفاده از گیت NAND
- استفاده از دیکدر (مثل 74LS138)

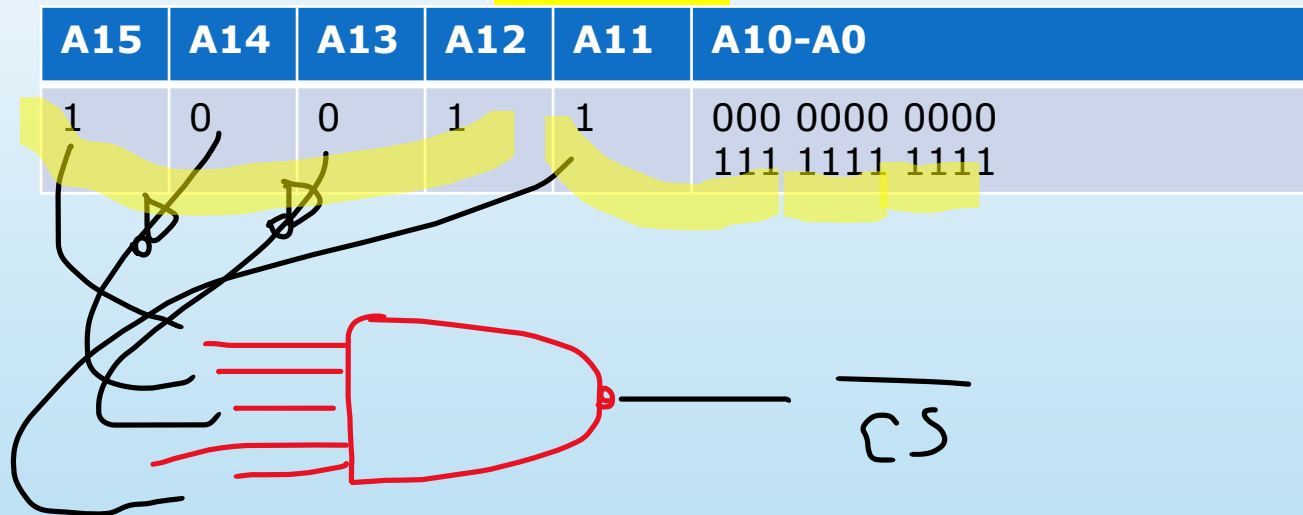
# کدگشایی با استفاده از NAND

خروجی NAND صفر فعال است و پایه های CS یا CE هم غالباً صفر فعال هستند، لذا استفاده از NAND بسیار ساده است:



□ آی سی حافظه با ۱۱ خط آدرس را در نظر بگیرید. اگر بخواهیم نگاشت حافظه بدهیم که در محدوده با شروع از 0xA800 قرار گیرد. چه باید بکنیم؟

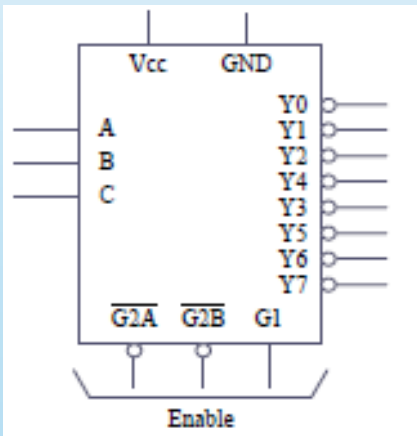
□  $0xA800 = 0b\ 1010\ 1000\ 0000\ 0000$





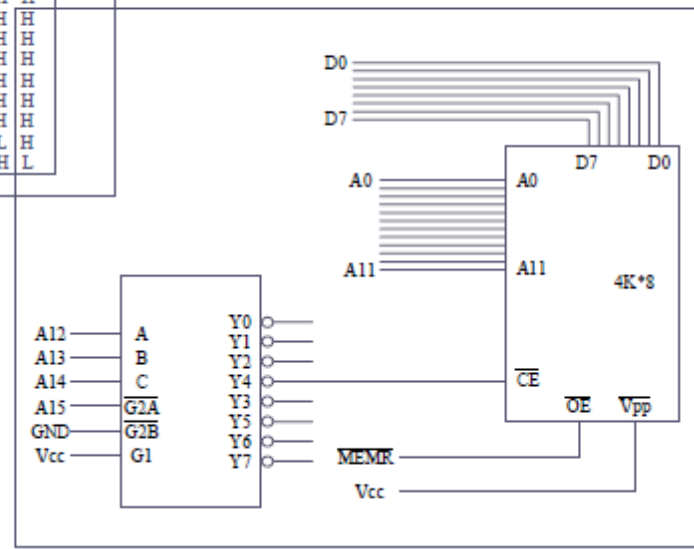
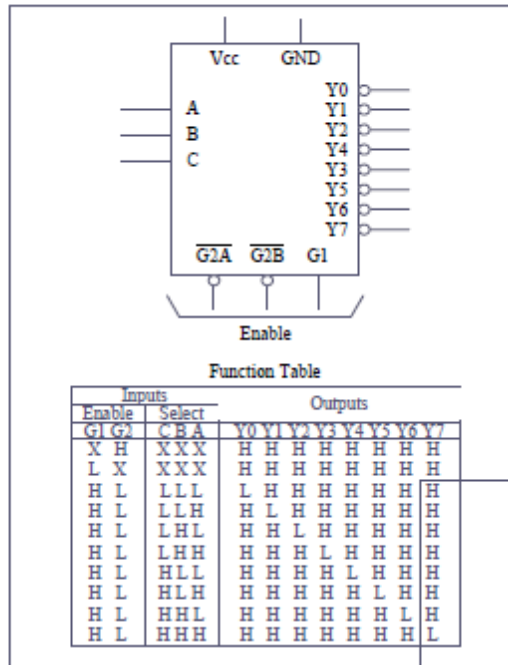
# کدگشایی با استفاده از 74LS138

- ☐ یک دیکدر ۳ به ۸
- ☐ ورودیها:  $A, B, C$  و خروجی  $Y_0-Y_7$  صفر فعال
- ☐ هر خروجی را می توان به یک حافظه جداگانه متصل کرد و به این ترتیب به ۸ حافظه تخصیص آدرس داد.
- ☐ سه ورودی اضافی  $G_2A, G_2B$  و  $G_1$  هم، خود دیکدر را فعال یا غیرفعال می کنند.
- ☐ این سه می توانند همواره فعال باشند (با اتصال به زمین یا  $V_{CC}$ ) یا در صورت نیاز به خطوط آدرس وصل شوند.



# کدگشایی با 74LS138

## 74LS138 Decoder



□ در شکل قبل، فضای آدرس مربوط به  $Y_4, Y_2$  و  $Y_7$  را بیابید

(a) The address range for  $Y_4$  is calculated as follows.

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1

The above shows that the range for  $Y_4$  is 4000H to 4FFFH. In Figure 14-6, notice that  $A_{15}$  must be 0 for the decoder to be activated.  $Y_4$  will be selected when  $A_{14} A_{13} A_{12} = 100$  (4 in binary). The remaining  $A_{11}-A_0$  will be 0 for the lowest address and 1 for the highest address.

(b) The address range for  $Y_2$  is 2000H to 2FFFH.

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1

(c) The address range for  $Y_7$  is 7000H to 7FFFH.

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

# طرز کار پردازنده

# ساختار داخلی کامپیوتر

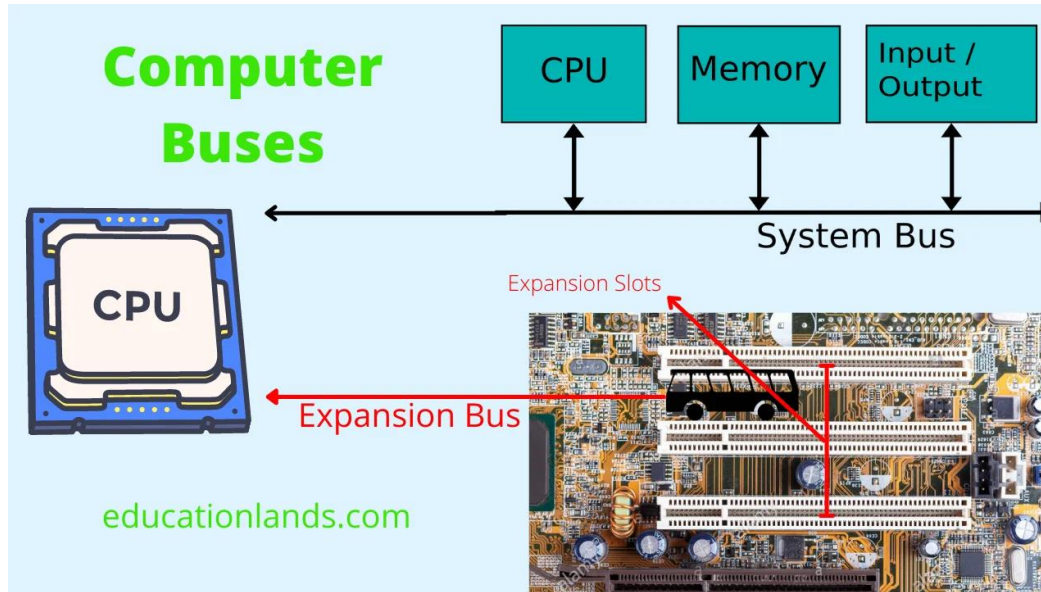
پردازنده از طریق نوارهای سیمی به حافظه و I/O ها متصل است که به این نوارها گذرگاه می گوییم.

اطلاعات توسط این گذرگاهها از جایی به جای دیگر منتقل می شود

گذرگاه آدرس

گذرگاه داده

گذرگاه کنترل



# گذرگاهها

## گذرگاه آدرس

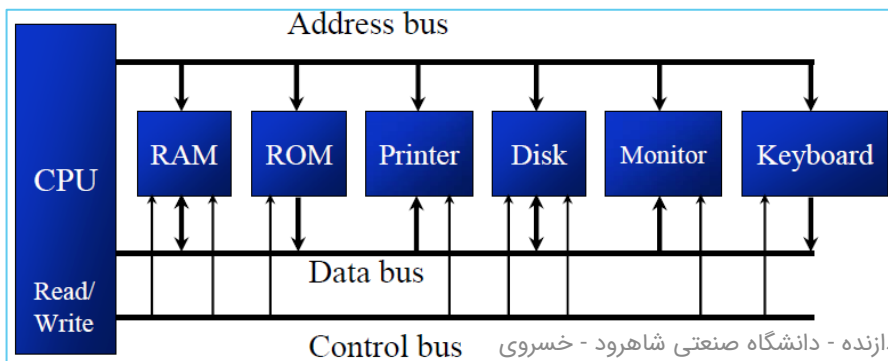
- برای تشخیص یک وسیله توسط پردازنده، باید آدرسی به آن تخصیص داده شود.
- آدرس باید یکتا باشد.
- پردازنده آدرس را روی گذرگاه آدرس قرار می دهد و مدار کدگشا قطعه مورد نظر را می یابد.

## گذرگاه داده

- برای خواندن داده از قطعات I/O یا نوشتن داده در آنها

## گذرگاه کنترل

- سیگنال های خواندن و نوشتن توسط پردازنده تهیه شده و روی این گذرگاه قرار می گیرد تا قطعات از تصمیم CPU برای دریافت یا ارسال اطلاعات مطلع شوند





# گذرگاه داده

- ◀ هرچه تعداد گذرگاه های داده بیشتر باشد، پردازنده نیز بهتر خواهد بود.
- ◀ گذرگاه های داده را مانند خطوط بزرگراه تصور کنید.
- ◀ گذرگاه داده بزرگتر به معنای کامپیوتر و پردازنده گرانتر است.
- ◀ نوعا اندازه گذرگاه داده بین ۸ تا ۶۴ است.
- ◀ گذرگاه های داده ۲ طرفه هستند.
- ◀ برای خواندن و نوشتن
- ◀ توانایی پردازش یک کامپیوتر به اندازه این گذرگاه ها ارتباط دارد.

# گذرگاه آدرس

- هرچه **گذرگاه آدرس** بزرگتر باشد، تعداد قطعاتی که می توان آدرس دهی کرد بیشتر می شود.
- تعداد مکان‌هایی که پردازنده می تواند آدرس دهی کند، همیشه برابر  $2^x$  است که  $x$  تعداد خطوط آدرس بوده و ارتباطی با گذرگاه داده ندارد.
- مثال:
- پردازنده با **۲۴** خط آدرس و **۸** خط داده می تواند چند مگابایت را آدرس دهی کند؟

# گذرگاه آدرس

◀ مثال:

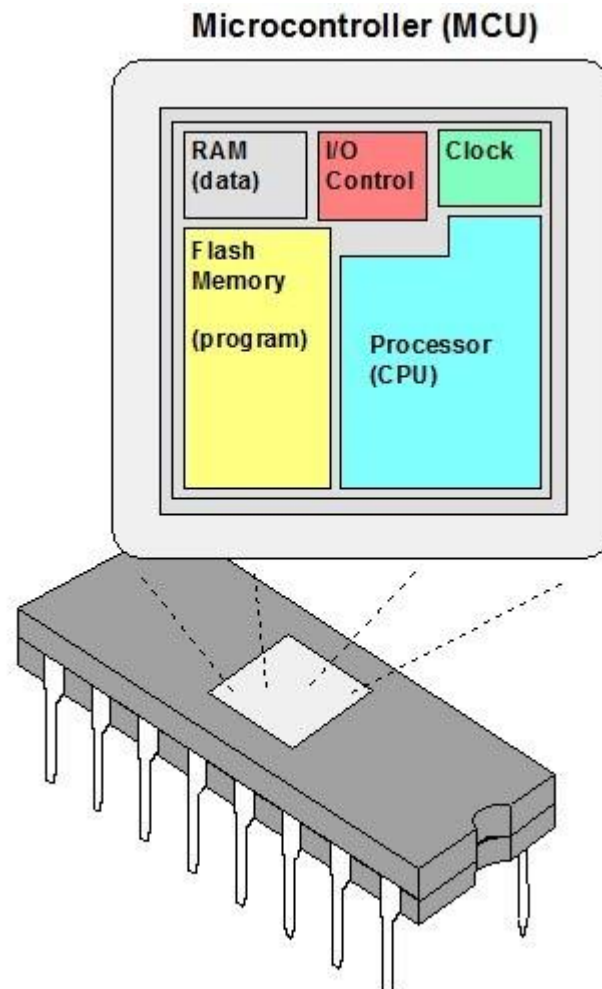
◀ پردازنده با ۲۴ خط آدرس و ۸ خط داده می تواند چند مگابایت را آدرس دهی کند؟

◀  $2^{24}$  یا ۱۶ مگابایت

◀ هرخانه حافظه نوعا یک بایت است لیکن بسته به گذرگاه داده و معماری حافظه، این عدد می تواند متفاوت باشد

◀ گذرگاه آدرس یک طرفه است. (چرا؟)

# ارتباط پردازنده با RAM و ROM



# ارتباط پردازنده با RAM و ROM

- ◀ برای پردازش اطلاعات توسط پردازنده، باید در حافظه RAM یا ROM ذخیره شوند.
- ◀ به این حافظه ها، حافظه اولیه (Primary) می گوئیم
- ◀ وظیفه ROM فراهم کردن اطلاعات ثابت و دائمی است
  - ◀ برنامه راه انداز میکرو
  - ◀ جداول ثابت مثل مراکز استانها یا مختصات جغرافیایی شهرها
- ◀ حافظه RAM حاوی اطلاعاتی است که ثابت نبوده و می توانند تغییر کنند
  - ◀ مثل متغیرها و آرایه های موقت که در برنامه استفاده می شوند
  - ◀ در مورد کامپیوتر، خود سیستم عامل و برنامه های کاربردی هم در RAM قرار می گیرند و ROM تنها نقش راه انداز کامپیوتر را دارد
- ◀ پردازنده ابتدا اطلاعات مورد پردازش را از RAM درخواست می کند، در صورتیکه در آن جا پیدا نشد، شروع به جستجو در وسایل ذخیره سازی انبوه مانند دیسک می کند که حافظه ثانویه نام دارد و سپس اطلاعات را به RAM منتقل می کند.

# ثباتها

◀ ثبات (Register) به حافظه های کوچکی گفته می شود که برای کاربردهای خاص استفاده می شوند

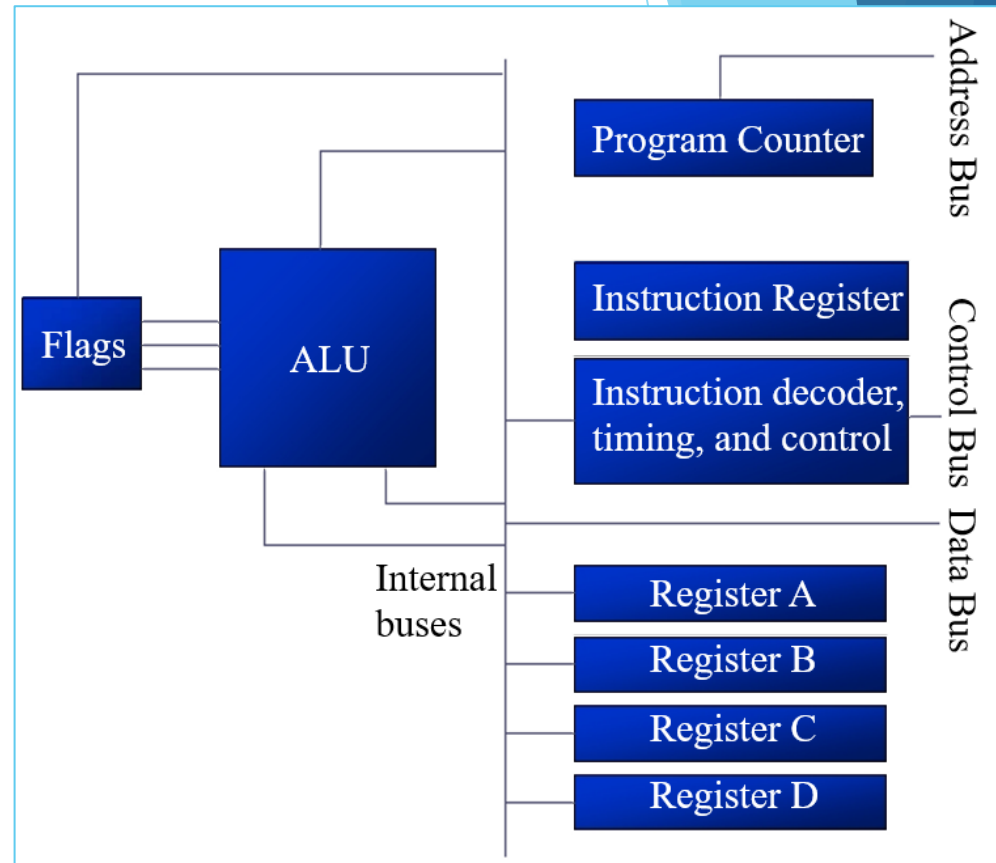
◀ پردازنده از ثبات ها برای ذخیره اطلاعات به صورت موقت استفاده می کند.

◀ این اطلاعات می تواند داده ای برای پردازش باشد

◀ یا آدرس داده ای که لازم است واکنشی شود.

◀ ثبات بیشتر و بزرگتر ← پردازنده بهتر و گرانتر!

◀ ثبات ها می توانند ۸، ۱۶، ۳۲ یا ۶۴ بیتی باشند.





# داخل پردازنده

## ◀ واحد محاسبه و منطق (ALU)

◀ توابع محاسباتی مثل جمع، تفریق، ضرب، تقسیم و توابع منطقی مانند AND، OR و NOT را اجرا میکند.

## ◀ ثبات شمارنده برنامه

◀ به دستور بعدی که باید اجرا شود، اشاره می کند

◀ هر بار که دستوری اجرا می شود، شمارنده برنامه افزایش می یابد و به آدرس دستورالعمل بعدی اشاره می کند.

## ◀ کدگشای دستورالعمل

◀ دستورالعمل وارد شده به CPU را تفسیر می کند

◀ پردازنده برای فهم دستورات بیشتر به ترانزیستورهای بیشتر نیاز دارد

# مثالی از عملکرد ریزپردازنده

- ◀ پردازنده ای را فرض کنید که:
  - ◀ شامل ثباتهای A، B، C و D باشد
  - ◀ گذرگاه داده ۸ بیتی دارد
  - ◀ گذرگاه آدرس ۱۶ بیتی دارد
  - ◀ یعنی آدرسهای 0x0000 تا 0xFFFF را پوشش می دهد
  - ◀ کد لازم برای انتقال یک مقدار به ثبات A عبارت 0xB0 باشد
  - ◀ و کد لازم برای جمع یک مقدار با ثبات A عبارت 0x04 باشد
- ◀ کاری که پردازنده می خواهد انجام دهد:
  - ◀ عدد 0x21 را در ثبات A بریزد
  - ◀ سپس مقادیر 0x42 و 0x12 را با A جمع کند

A = 0x21

A = A + 0x42

A = A + 0x12

# ادامه مثال

Action	Code	Data
Move value 0x21 into reg. A	0xB0	0x21
Add value 0x42 to reg. A	0x04	0x42
Add value 0x12 to reg. A	0x04	0x12

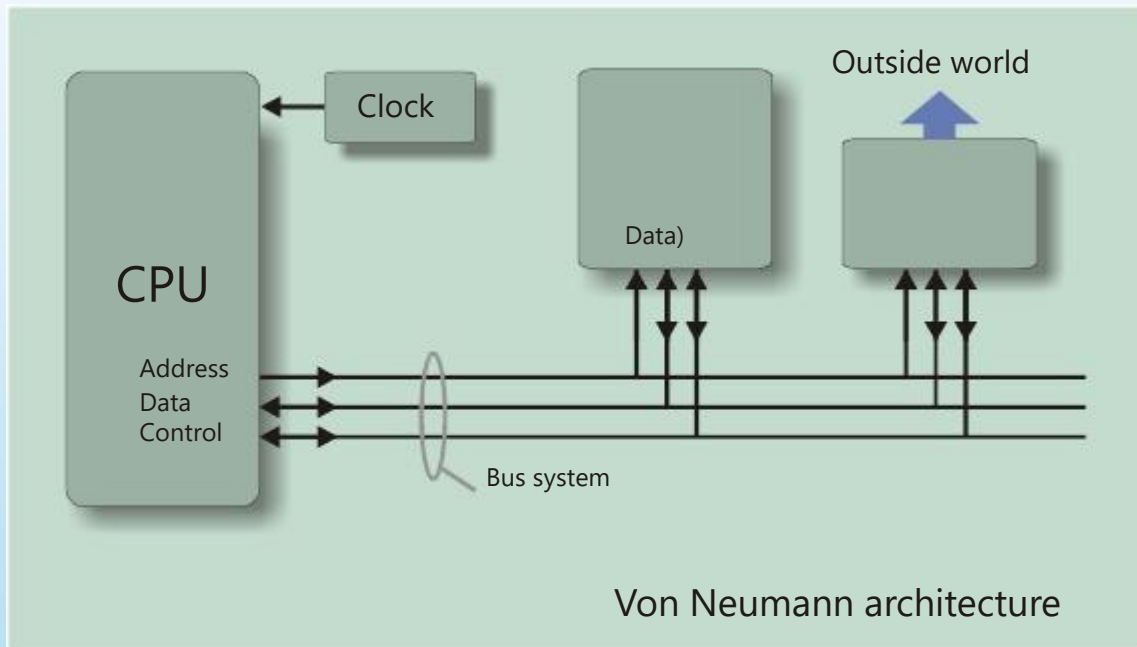
A = 0x21  
A = A + 0x42  
A = A + 0x12

## Mem. Addr. Contents of memory address

1400	(B0) code for moving a value to register A
1401	(21) value to be moved
1402	(04) code for adding a value to register A
1403	(42) value to be added
1404	(04) code for adding a value to register A
1405	(12) value to be added
1406	(F4) code for halt

## معماری ون نیومن □

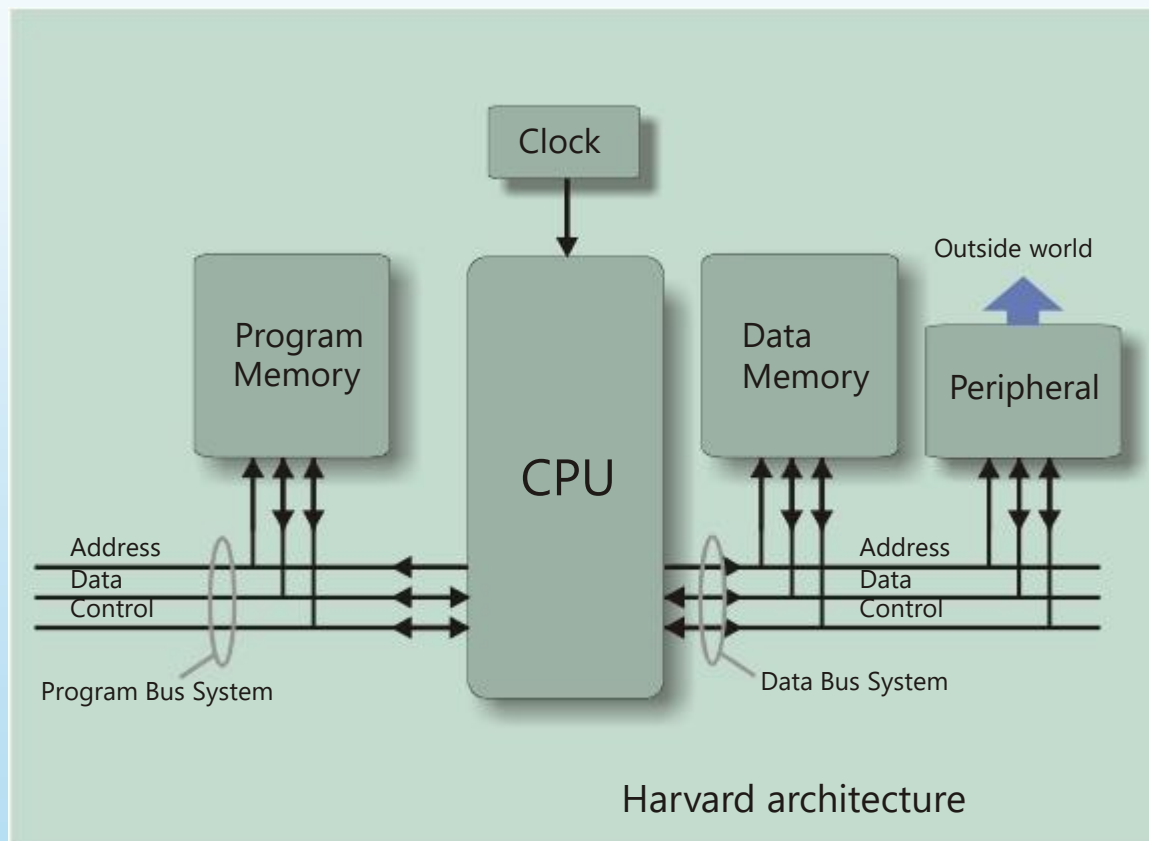
- گذرگاه داده و آدرس مشترک برای داده‌ها و کد
- مورد استفاده در اکثر کامپیوترهای امروزی



## □ دو گذرگاه داده و دو گذرگاه آدرس

■ یکی برای داده ها و ابزارهای جانبی

■ دیگری برای کد



# تمرین سری اول کاغذی

◀ فصل صفرم

◀ ۱، ۲، ۳، ۴، ۷، ۱۰، ۱۱، ۱۴، ۲۱، ۲۲، ۲۳، ۲۶، ۳۱، ۳۵، ۳۷، ۴۲، ۴۴، ۴۶، ۵۰

◀ فصل اول

◀ ۱، ۹، ۱۱، ۱۲، ۱۵، ۱۶، ۱۷، ۱۸، ۲۶، ۲۹

◀ تحویل ۱۳ مهر

