



# معماری و سازمان کامپیوتر

دانشگاه صنعتی اصفهان

دانشکده مهندسی برق و کامپیوتر

امیر خورسندی

پاییز ۱۴۰۲

# سیستم های چند پردازنده ای

## مقدمه

- یک سیستم چندپردازنده ای از دو یا تعداد بیشتری هسته پردازی به همراه حافظه و تجهیزات ورودی/خروجی تشکیل شده است.
- امکان گنجاندن چندین هسته بر روی تراشه بر اساس پیشرفت تکنولوژی و بر مبنای تحقق قانون مور حاصل شده است.
- این سیستم ها در دسته سیستم های چند دستورالعمل و چند داده از طبقه بندی فلین قرار می گیرند.

# مقایسه با سیستم های چند کامپیوتری

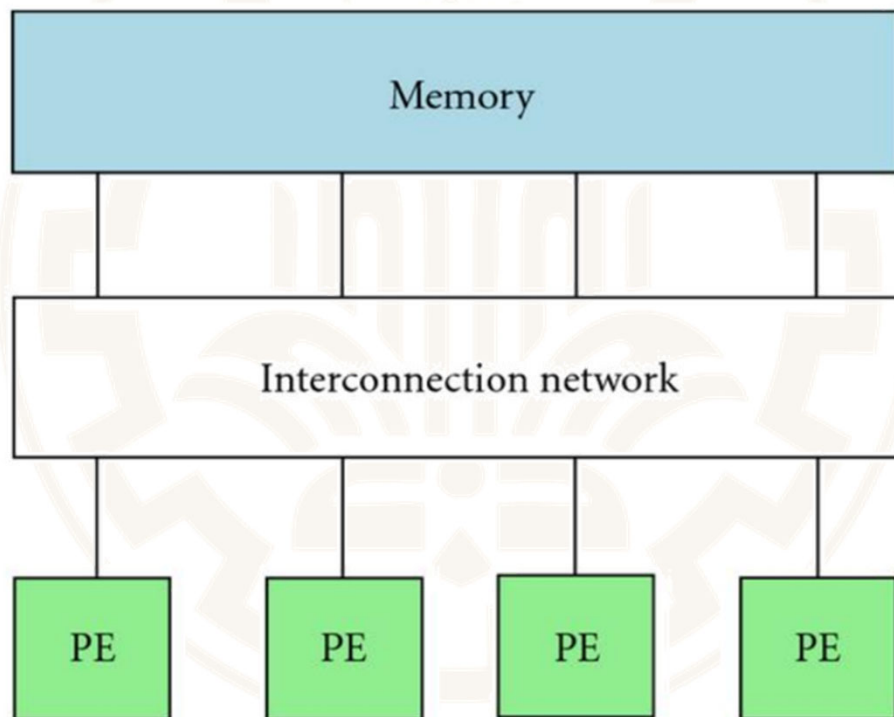
- ارتباط بین کامپیوترها از طریق شبکه کامپیوتری برقرار می شود.
- هر کامپیوتر سیستم عامل مجزای خود را دارد.
- هر کامپیوتر وظیفه مجزایی را بر عهده دارد.
- سیستم های چند کامپیوتری از نوع سیستم های با اتصال سست محسوب می شوند.

# مزایا

- مهم ترین مزیت سیستم چند پردازنده افزایش کارایی سیستم به دلیل امکان اجرای همزمان بخش های مجزای برنامه نرم افزاری است.
- کاربر می تواند برنامه را به بخش های غیرمرتبط تقسیم کند و هر بخش را به صورت همزمان با سایر بخش ها بر روی یک هسته مجزا اجرا نماید.
- برنامه نویس و کامپایلرها به تقسیم وظایف و سیستم عامل ها به توزیع آن بر روی هسته های مختلف جهت افزایش کارایی می پردازند.
- قابلیت تحمل خطای سیستم افزایش می یابد.

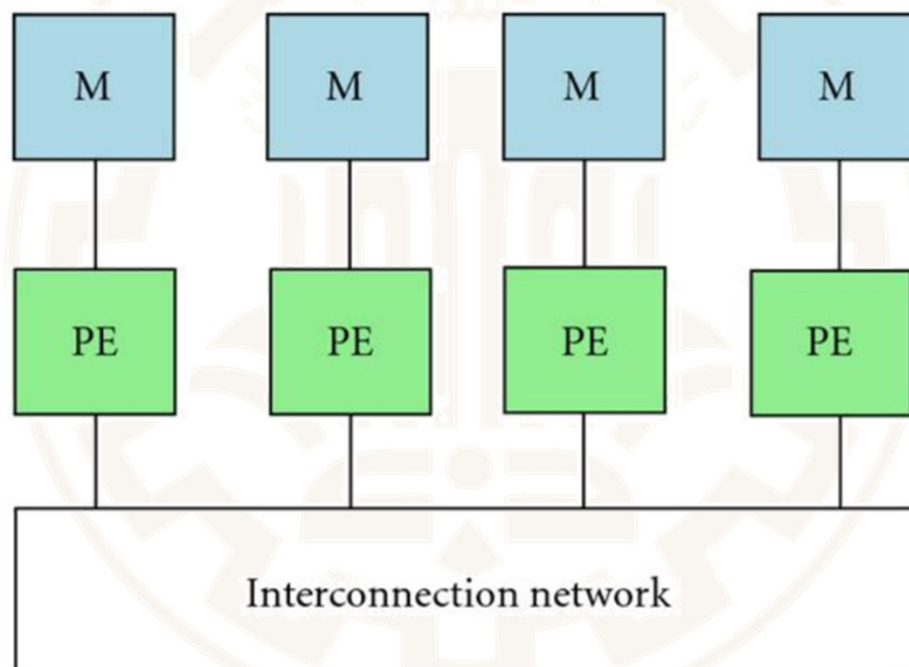
# انواع سیستم چند پردازنده ای

- سیستم با حافظه مشترک:



# انواع سیستم چند پردازنده ای (ادامه)

• سیستم با حافظه توزیع شده:

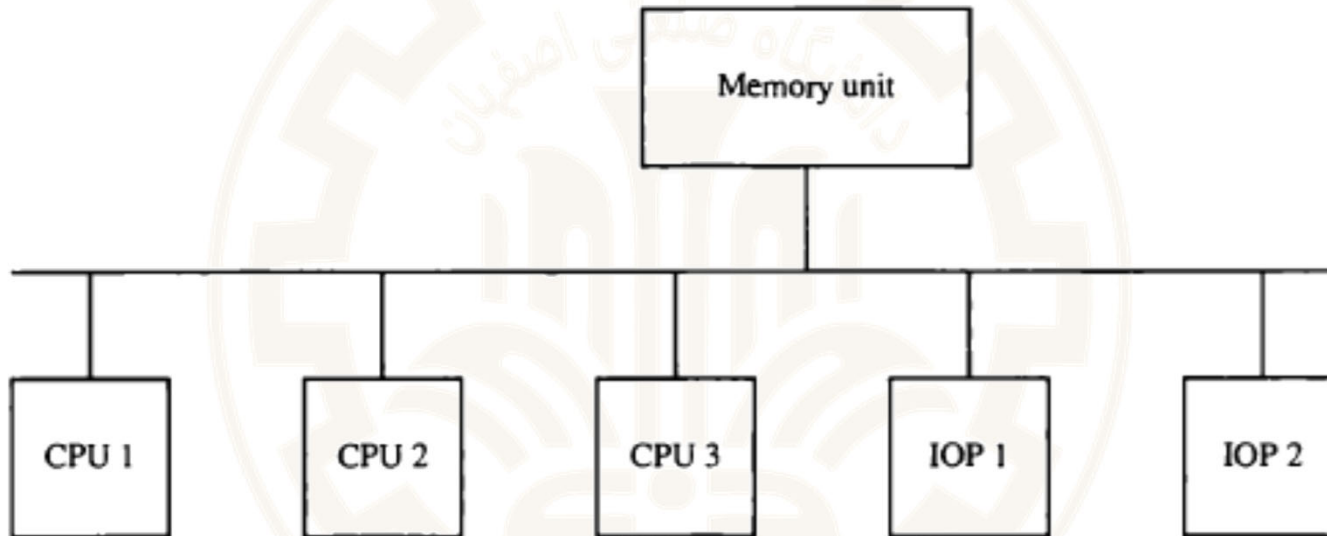


# انواع ساختارهای ارتباطی

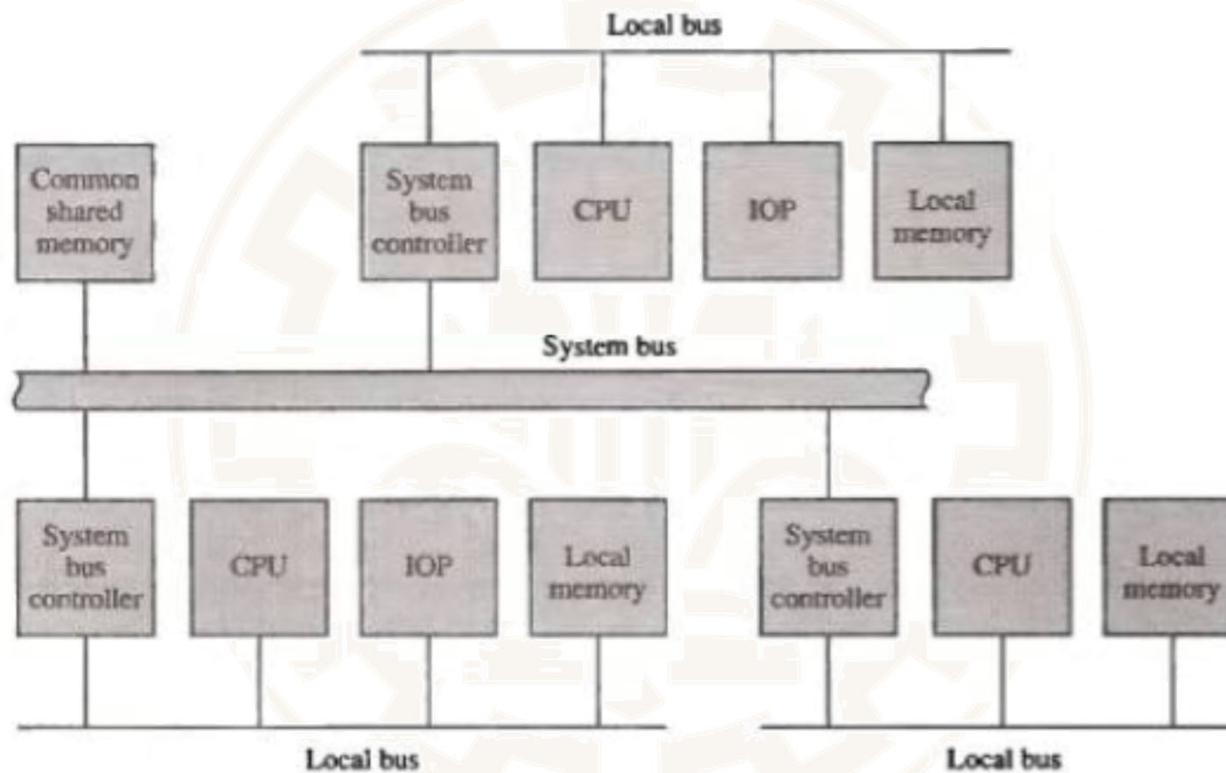
- گذرگاه مشترک با اشتراک زمانی
- سیستم حافظه با درگاه چندگانه
- سویچ های Crossbar
- سویچ چند طبقه ای
- اتصال Hypercube



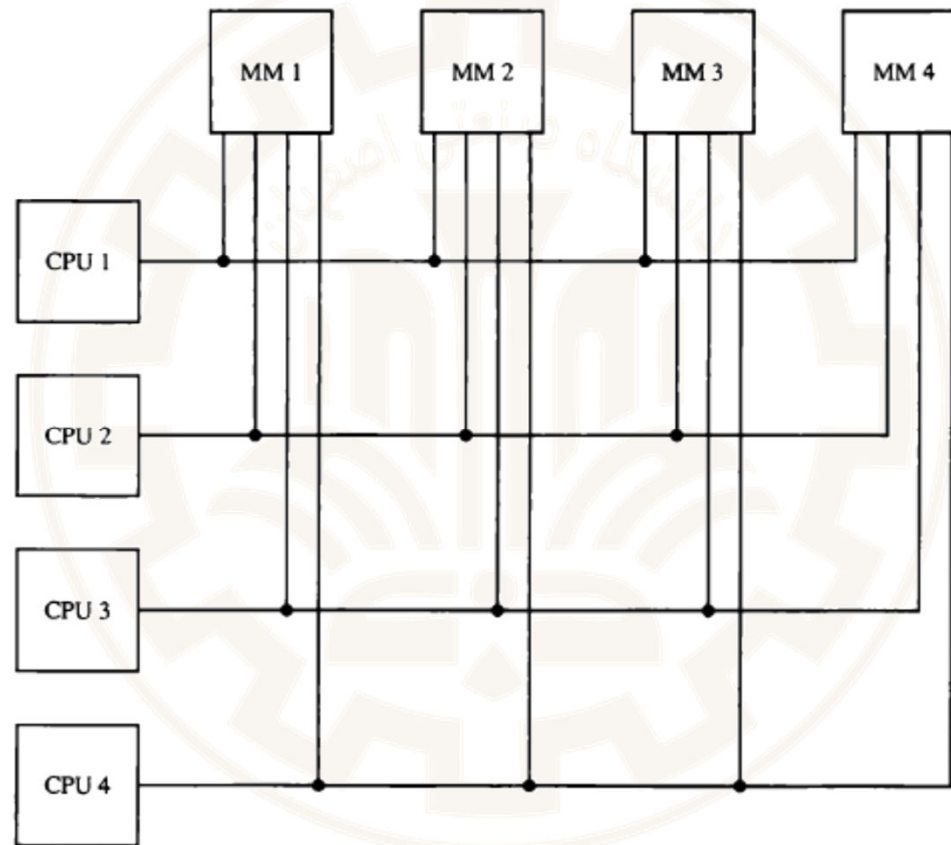
# گذرگاه مشترک با اشتراک زمانی



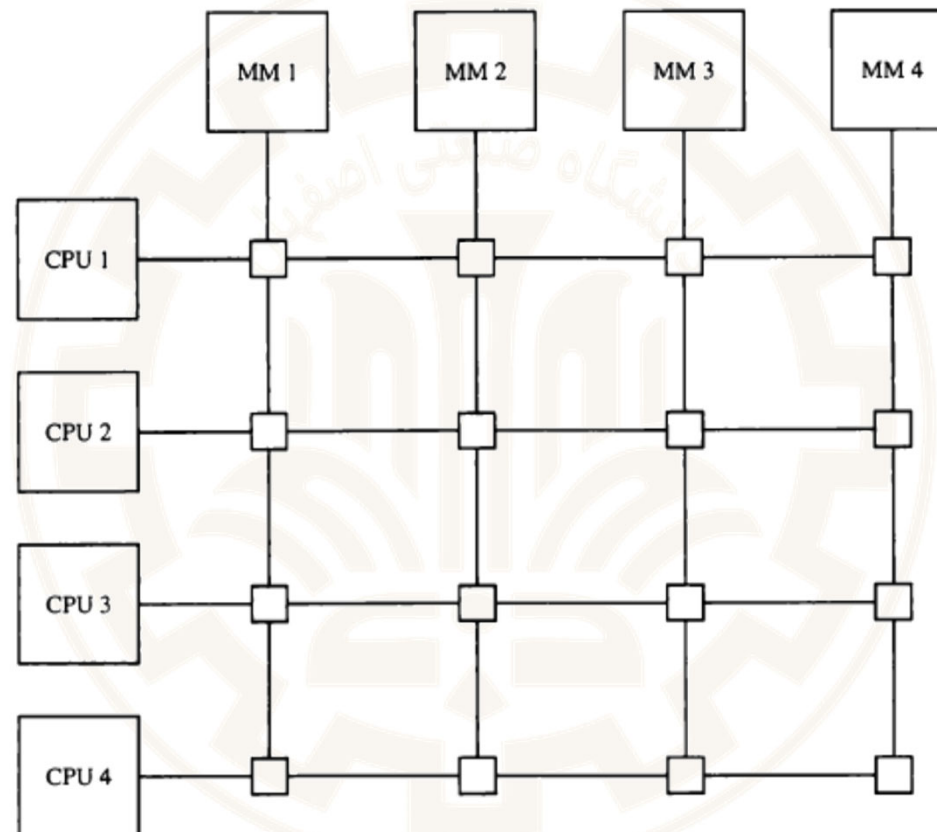
# گذرگاه مشترک سلسله مراتبی



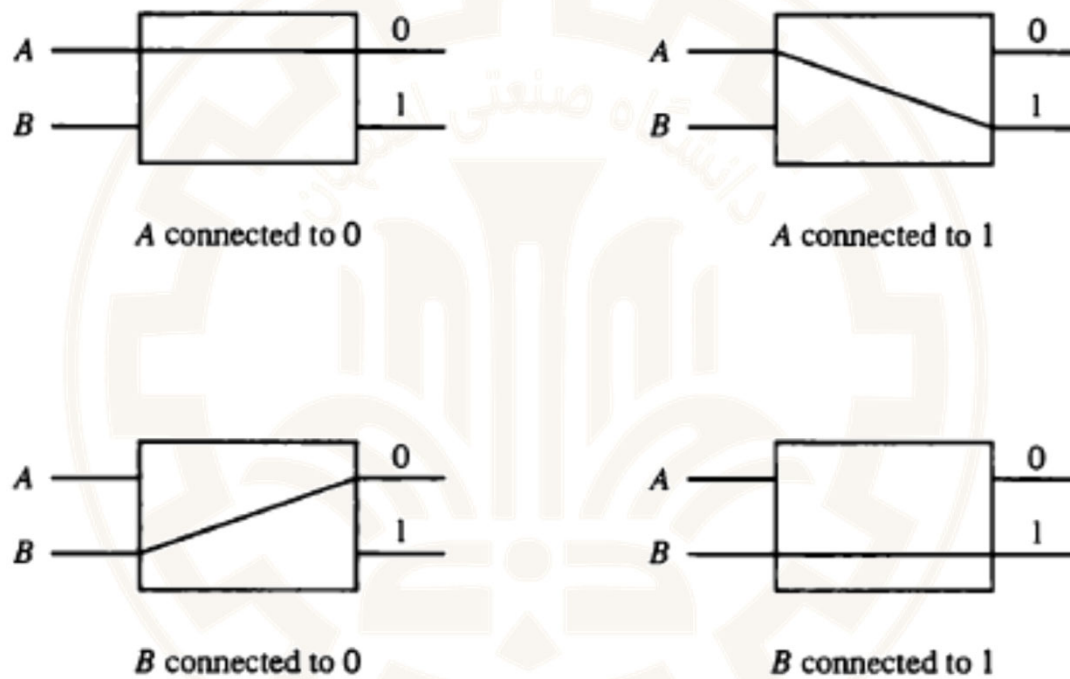
# سیستم حافظه با درگاه چندگانه



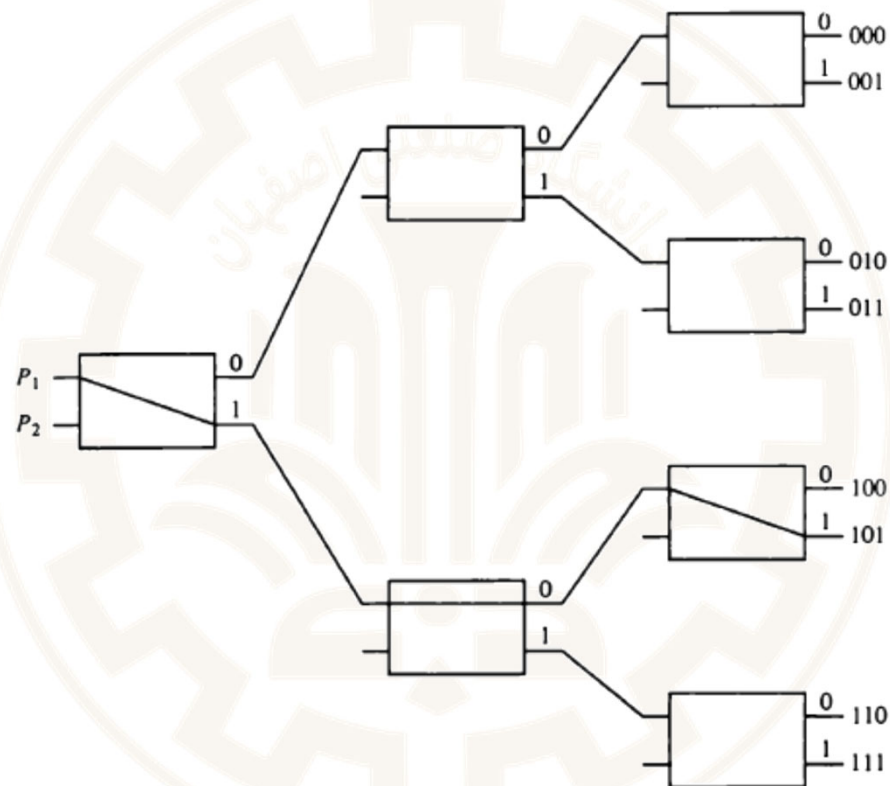
# سویچ Crossbar



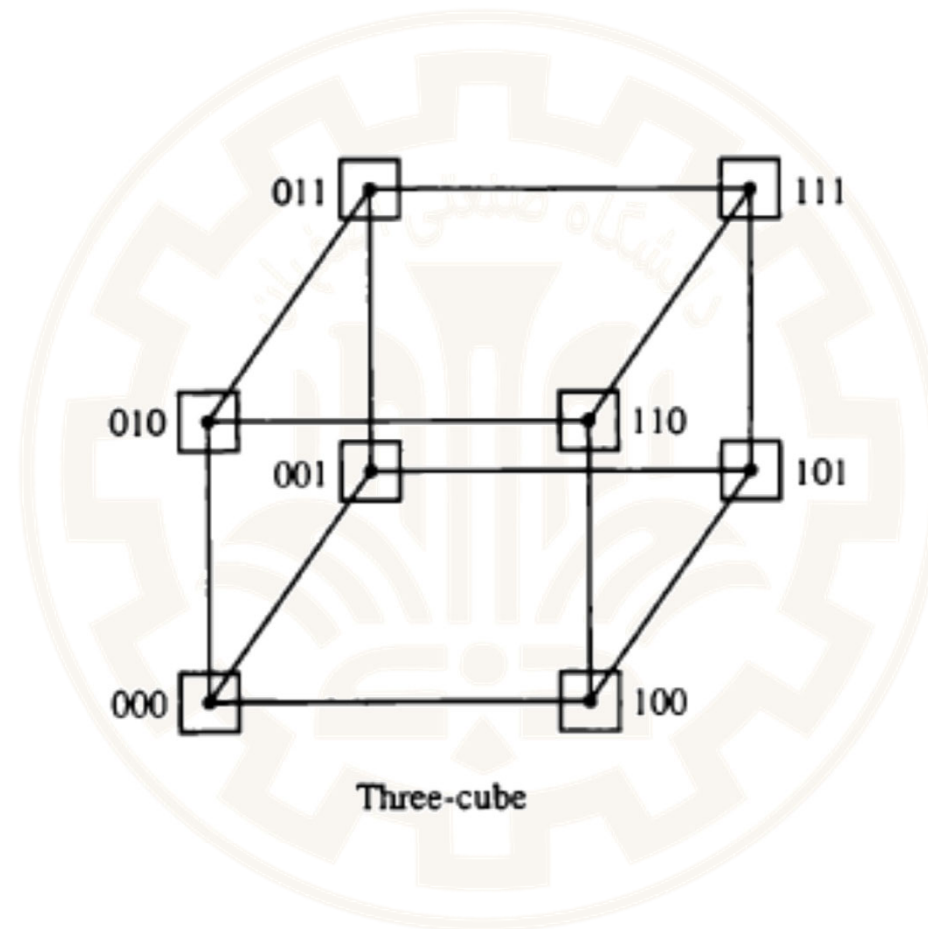
## سویچ ۲ در ۲



# سویچ چند طبقه



# اتصال Hypercube



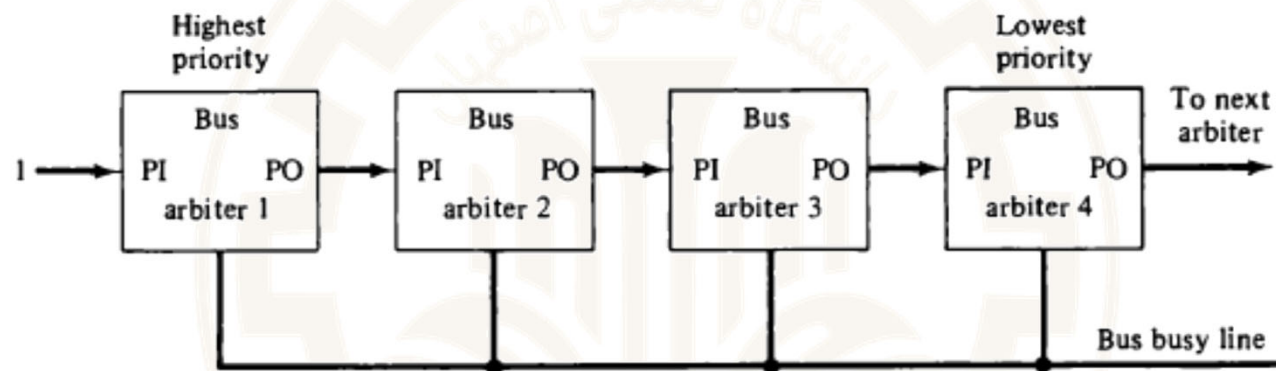
# کنترل دسترسی به ساختار ارتباطی مشترک

- ساختار زنجیره ای
- ساختار موازی

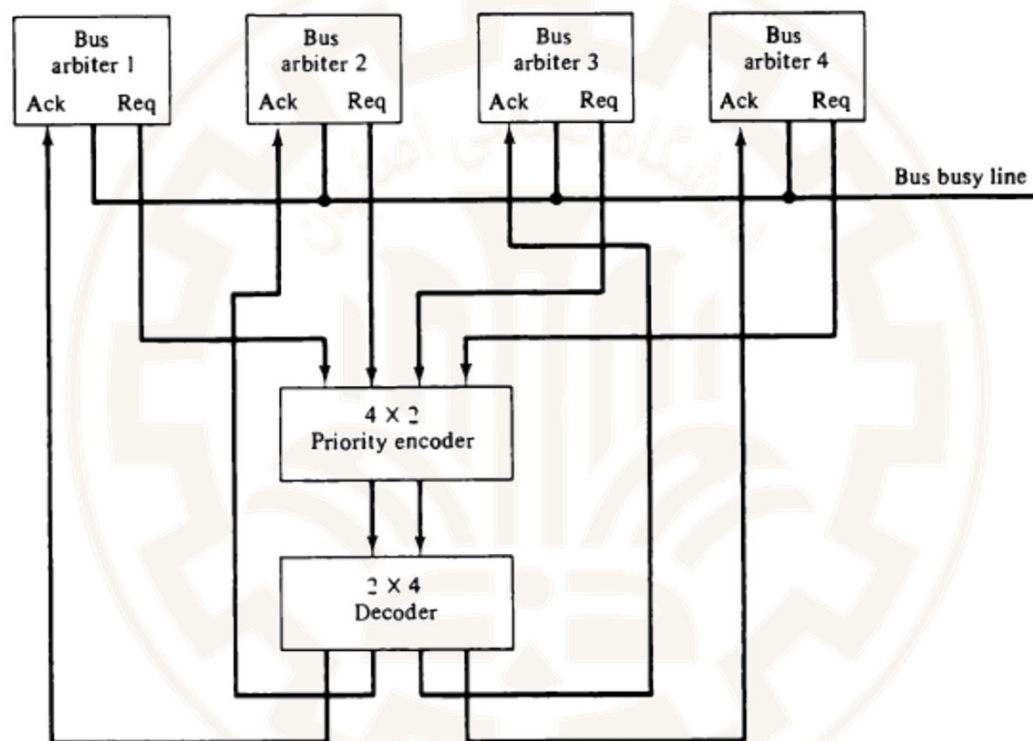




# ساختار زنجیره ای



# ساختار موازی



# چالش ها

- هماهنگ سازی عملکرد هسته های مختلف
- کنترل دسترسی به منابع مشترک
- حفظ سازگاری بین نسخه های مختلف یک داده

# هماهنگ سازی عملکرد هسته های مختلف

• حافظه مشترک

• تبادل پیام



# کنترل دسترسی به منابع مشترک

- ساختار پایه/پیرو

- قفل نمودن داده در ناحیه بحرانی (سمافور)

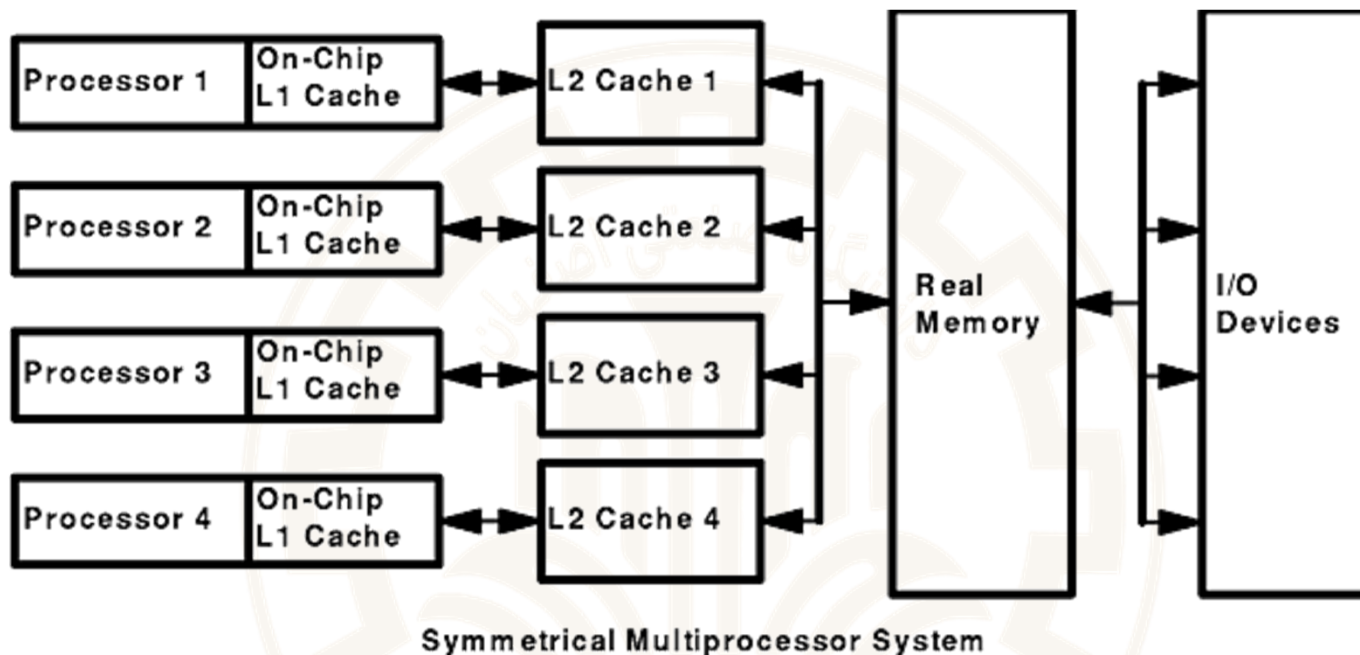
```
while (Lock);
```

```
Lock = true;
```

```
....
```

```
Lock = false;
```

# حفظ سازگاری بین نسخه های مختلف یک داده



سیاست های به روزرسانی در حافظه نهان

- Write Through
- Write Back