Operating Systems

Masoud Baeimani

November 2019

توصیف و کنترل فرآیند
Process Description and Control

فرآیند (Process)

- فرآیند:
  - یک برنامه که برای اجرا در حافظه بار شده است و اصطلاحا به آن حافظه اختصاص یافته است.
  - یک موجودیت که قابل اختصاص به یک پردازنده برای اجراست.
  - یک برنامه مجزای در حال اجرا.

- وظیفه اصلی سیستم عامل اجرای صحیح فرآیندهاست.

اجزای هر فرآیند

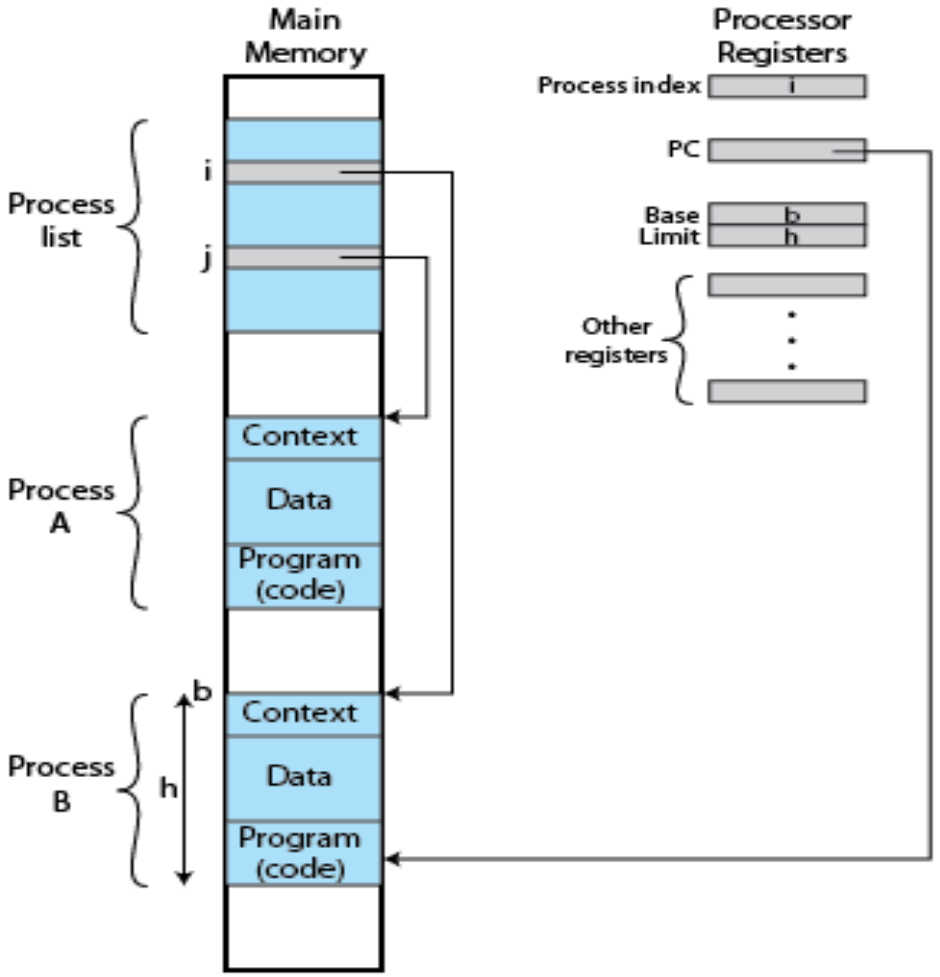


Figure 2.8 Typical Process Implementation

- یک **برنامه** قابل اجرا (کد برنامه)
- **داده‌های** برنامه (متغیرها، فضای کاری، بافرها و ...)
- **متن** یا وضعیت اجرای برنامه (Execution context)
  - اطلاعات لازم برای اجرای برنامه توسط سیستم عامل
    - محتوای ثباتهای مختلف پردازنده
    - اولویت فرآیند
    - حالت فرآیند (اجرا یا مسدود)

حالات فرآیند

- اساسی‌ترین عمل پردازنده اجرای دستورالعمل های موجود در حافظه است.
- رفتار یک فرآیند به خصوص را می توان با فهرست کردن دنباله دستورالعمل‌هایی که برای آن فرآیند اجرا می‌شود مشخص نمود که به آن رد (Trace) آن فرآیند گویند.
- با نمایش چگونگی تداخل ردهای فرآیندهای مختلف، می‌توان رفتار پردازنده را مشخص کرد.

مثال

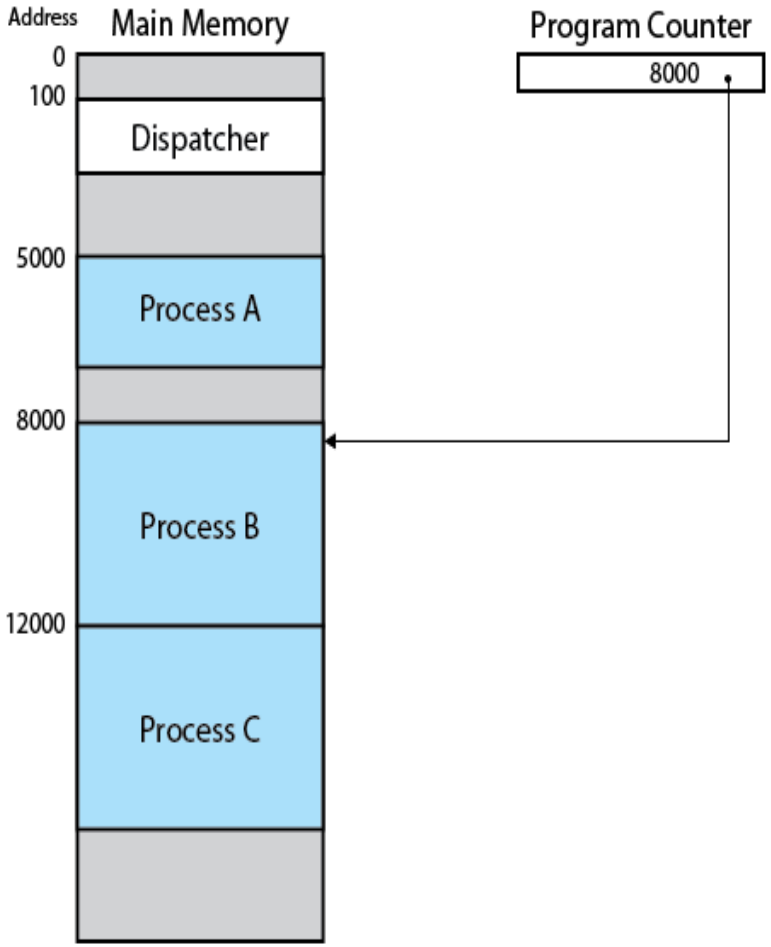


Figure 3.2 Snapshot of Example Execution (Figure 3.4) at Instruction Cycle 13

1	5000	27	12004
2	5001	28	12005
3	5002	----------	Time out
4	5003	29	100
5	5004	30	101
6	5005	31	102
----------	Time out	32	103
7	100	33	104
8	101	34	105
9	102	35	5006
10	103	36	5007
11	104	37	5008
12	105	38	5009
13	8000	39	5010
14	8001	40	5011
15	8002	----------	Time out
16	8003	41	100
----------	I/O request	42	101
17	100	43	102
18	101	44	103
19	102	45	104
20	103	46	105
21	104	47	12006
22	105	48	12007
23	12000	49	12008
24	12001	50	12009
25	12002	51	12010
26	12003	52	12011
		----------	Time out

100 = Starting address of dispatcher program

shaded areas indicate execution of dispatcher process;
first and third columns count instruction cycles;
second and fourth columns show address of instruction being executed

Figure 3.4 Combined Trace of Processes of Figure 3.2

علت بروز خطاها در سیستم نرم افزاری

- ۱- همگام سازی نا مناسب (Improper synchronization)
  - برنامه‌ها برای هماهنگی با یکدیگر نیازمند علامت‌دهی هستند، طراحی نامناسب راهکار علامت‌دهی، می‌تواند موجب از دست رفتن علامتها و یا علامتهای تکراری گردد.

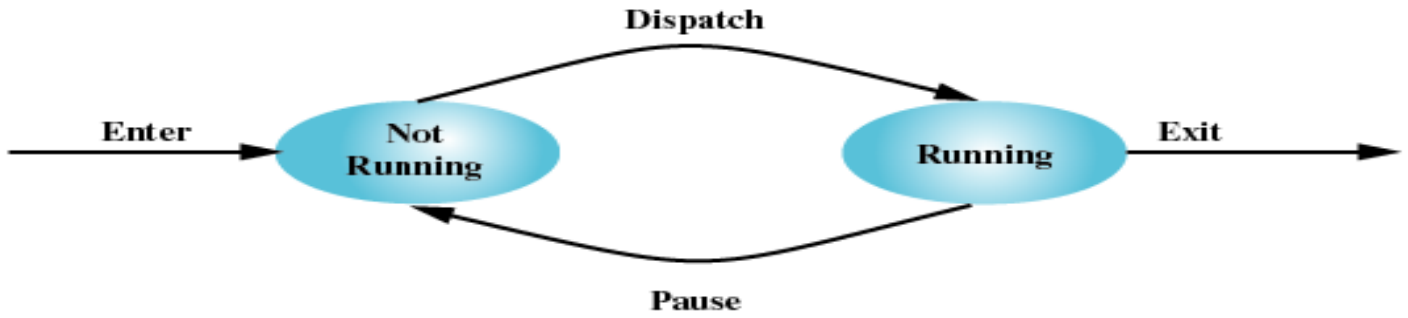
- ۲- شکست در انحصار متقابل (Failed mutual exclusion)
  - اغلب بیش از یک برنامه یا کاربر در تلاش برای استفاده همزمان از یک منبع مشترک هستند. نوعی راهکار انحصار متقابل باید وجود داشته باشد تا در یک زمان تنها یک روال با آن بخش از دادها مشترک تراکنش داشته باشد.

- ۳- عملکرد غیر قطعی برنامه
  - در حالت عادی، نتایج برنامه باید تنها به ورودی آن (و نه به فعالیتهای برنامه‌های دیگر در محیط اشتراکی) بستگی داشته باشد.

- ۴- بن بست (Deadlock)
  - به حالتی که دو یا چند برنامه به خاطر یکدیگر معلق بمانند. هر کدام منتظرند تا برنامه دیگر منبع مورد نظرش را رها کند.

مدل عملکرد فرآیند (مدل دو حالته)

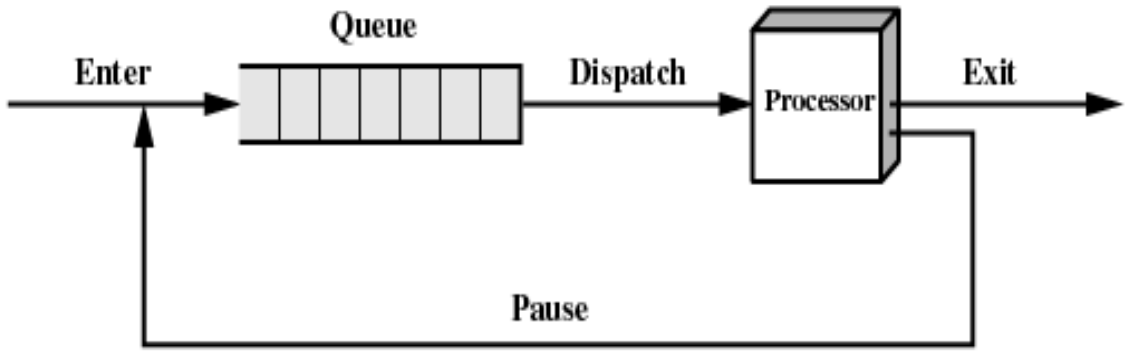
- در این مدل یک فرآیند دو حالت ممکن دارد :
  - اجرا (Running)
  - عدم اجرا
- وقتی یک سیستم عامل فرآیند جدیدی را ایجاد می‌کند آن را در حالت غیر اجرا قرار می‌دهد.
- پس از رسیدن نوبت، توزیع کننده (Dispatcher) فرآیند فعلی را به حالت غیر اجرا برده و فرآیند جدید را به حالت اجرا می‌برد.
- این مدل در مورد حالاتی که فرآیند منتظر I/O هست چیزی بیان نمی‌کند.



(a) State transition diagram

نمودار صف بندی (مدل دوحالته)

- فرآیندی که در حال اجرا نیست باید در صفی به انتظار نوبت قرار گیرد.
- وقتی یک فرآیند در معرض وقفه قرار می گیرد به صف انتظار می رود.
- اگر یک فرآیند کارش تمام شود کنار گذاشته می شود.



(b) Queuing diagram

ایجاد فرآیند

- دستور اجرا از سوی کاربر
- ورود یک کاربر جدید
- شروع یک سرویس داخلی سیستم عامل مثل چاپ
- یک فرآیند (والد)، فرآیند (فرزند) جدیدی را ایجاد نماید.

ختم فرآیند

- تکمیل اجرای فرآیند
- خروج کاربر از سیستم
- اتمام زمان مجاز کاربر
- ایجاد خطا در سیستم
  - عدم وجود حافظه کافی
  - عدم رعایت محدوده مجاز
- خطاهای محاسباتی
  - Overflow، DBZ
- خرابی در I/O
- ختم فرآیند والد (Parent)

مدل پنج حالته

- **جدید**(New): فرآیندی که هم اکنون ایجاد شده اما هنوز در حافظه اصلی بار نشده.
  - اطلاعات موردنیاز برای اجرای فرآیند در حافظه اصلی است ولی خود فرآیند (کد برنامه) در حافظه جانبی است.
- **آماده** (Ready): با گرفتن فرصت به اجرا در می‌آید.
- **اجرا** (Running): فرآیندی که هم اکنون پردازنده را در اختیار دارد.
- **مسدود** (Blocked): فرآیندی که تا وقوع رویدادی مثل کامل شدن I/O نمی‌تواند اجرا شود.
- **خروج**: به علت تکمیل شدن یا خطا، سیستم عامل آن را اخراج کرده است.

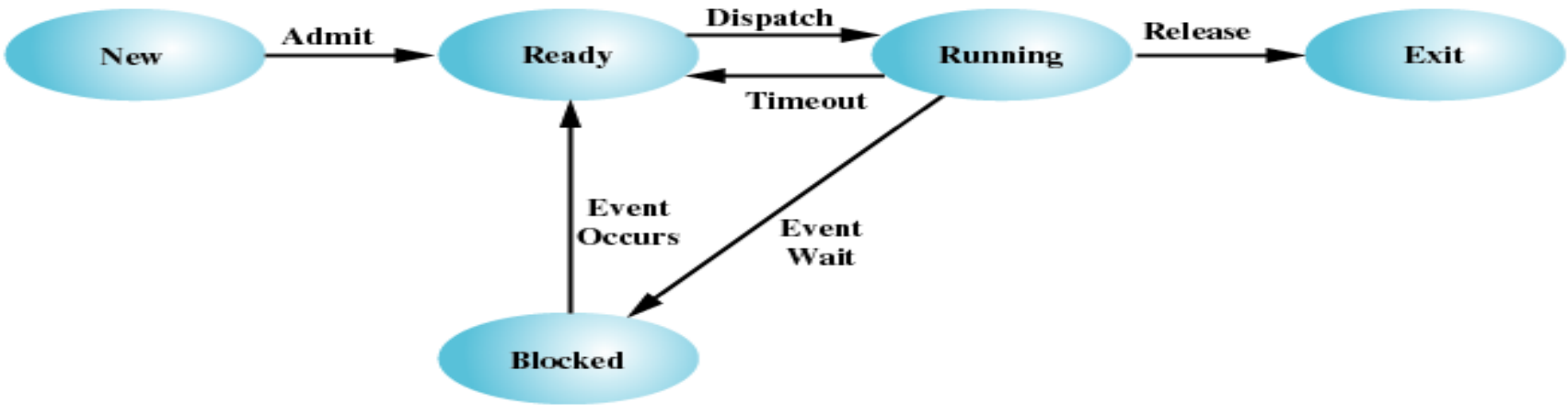
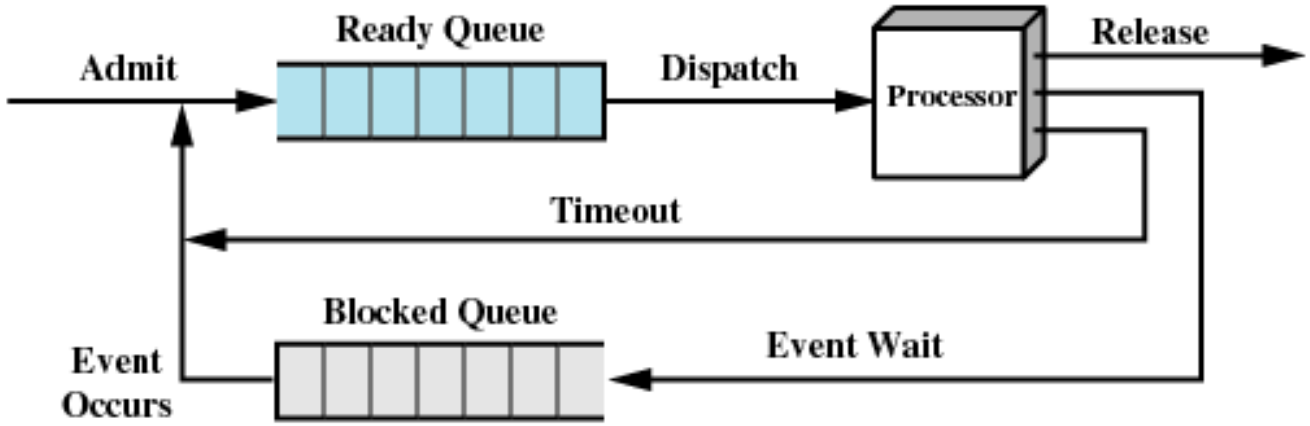


Figure 3.6 Five-State Process Model

مدل پنج حالته ...

- **اجرا مسدود**: اگر فرآیند در حین اجرا چیزی درخواست کند که برای بدست آوردن آن باید منتظر بماند، به حالت مسدود می‌رود.
- **مسدود آماده**: اگر رویداد مورد انتظار فرآیندی که در حالت مسدود بوده است رخ دهد، به حالت آماده می‌رود.



(a) Single blocked queue

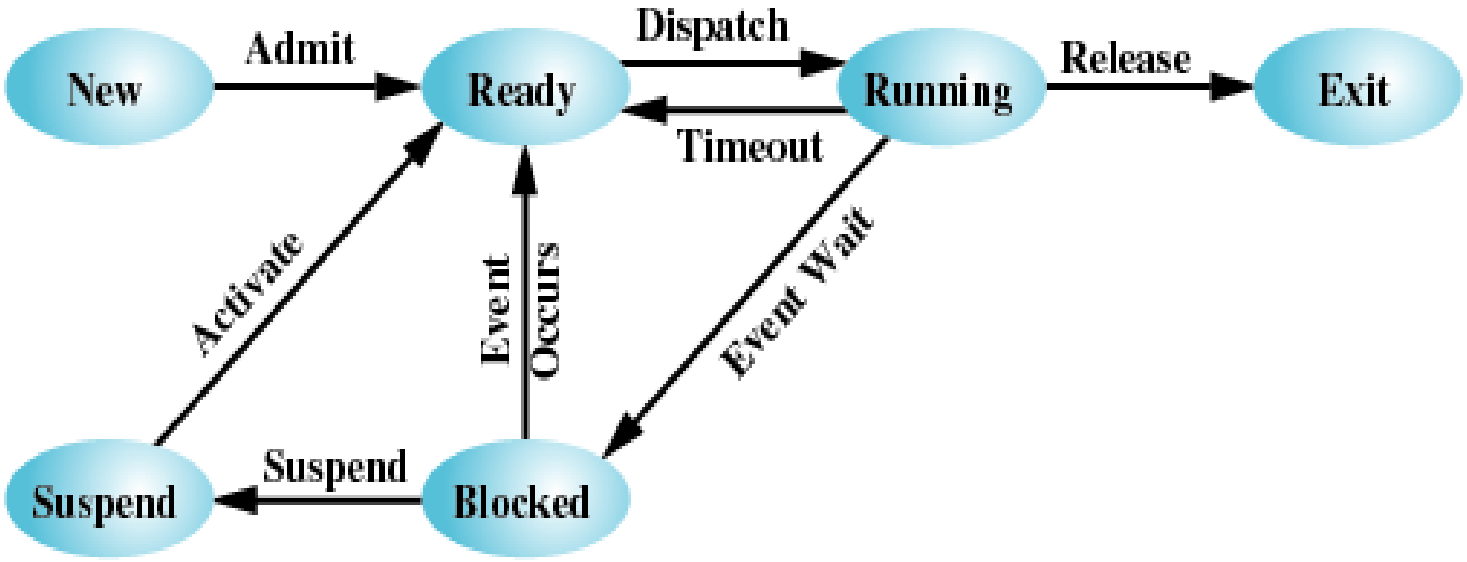
فرآیند معلق(Suspend)

- در مدل پنج حالته، بدلیل اینکه عملیات I/O خیلی کندتر از محاسبات است، بعد از مدتی همه فرآیندها در حالت مسدود قرار می‌گیرند پس در اکثر اوقات پردازنده بیکار می ماند. دو راه حل زیر پیشنهاد می شود:

- ۱- حافظه اصلی را گسترش داد
  - با افرایش هزینه همراه است.
  - درخواست برای حافظه افزایش پیدا میکند. بنابراین، حافظه بزرگتر، فرآیندهای بزرگتر را نتیجه می‌دهد، نه اینکه تعداد فرآیندها بیشتر شود.

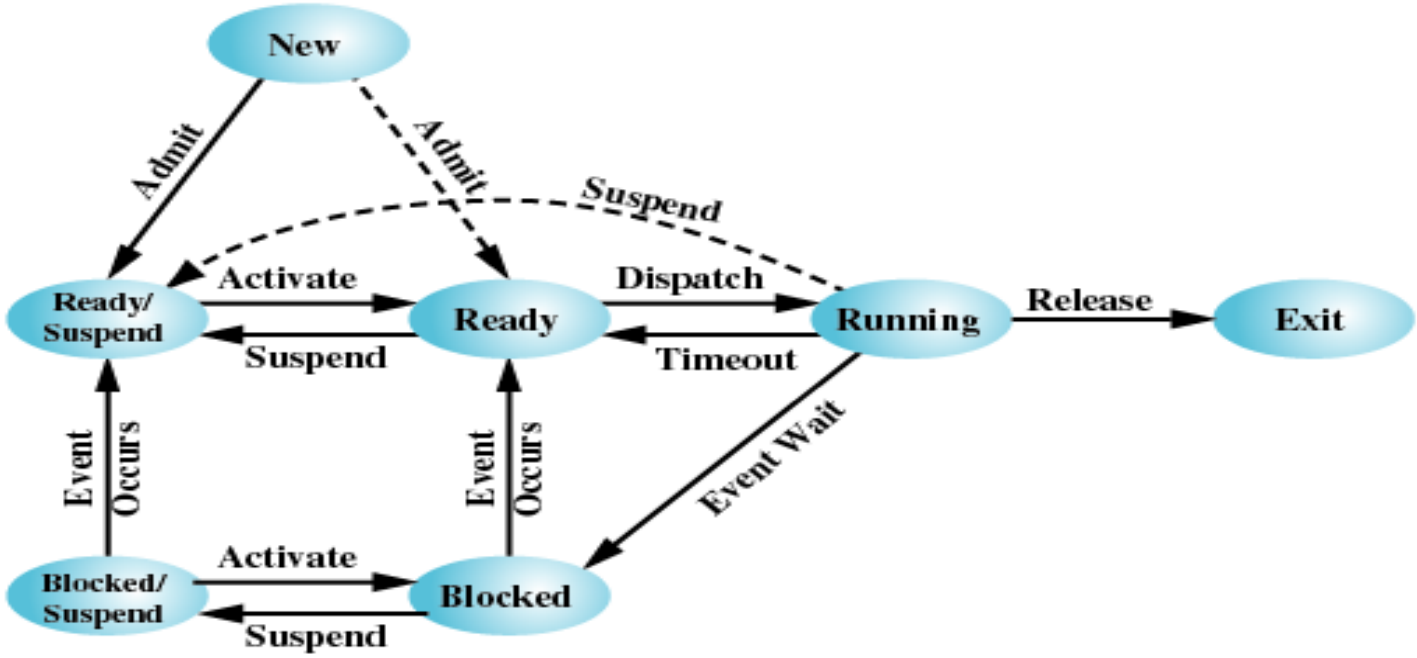
- ۲- **مبادله**: انتقال بخشی یا تمام یک فرآیند از حافظه اصلی به حافظه جانبی (دیسک).
  - هنگامی که هیچ یک از فرآیند های موجود در حافظه اصلی در حالت آماده نیستند، در این صورت سیستم عامل یکی از فرآیندهای مسدود را از حافظه اصلی خارج و به صف فرآیندهای **معلق** در روی حافظه جانبی می‌برد تا فضای حافظه اصلی آزادتر شود. (مبادله به خارج)
  - سپس سیستم عامل فرآیند دیگری را از صف معلق به حافظه اصلی می‌آورد، یا درخواست فرآیند جدیدی را می‌پذیرد. (مبادله به داخل)

مدل با یک حالت تعلیق



(a) With One Suspend State

مدل با دو حالت تعلیق



(b) With Two Suspend States

Figure 3.9 Process State Transition Diagram with Suspend States

مدل با دو حالت تعلیق ...

- **آماده**: فرآیند در حافظه اصلی و آماده برای اجراست
- **مسدود**: فرآیند موردنظر در حافظه اصلی و منتظر حادثه‌ای است.
- **مسدود و معلق**: فرآیند موردنظر در حافظه جانبی و منتظر حادثه‌ای است.
- **آماده و معلق**: فرآیند موردنظر در حافظه جانبی است و به محض بار شدن در حافظه اصلی آماده اجراست.

بلوک کنترل فرآیند
(Process Control Block)

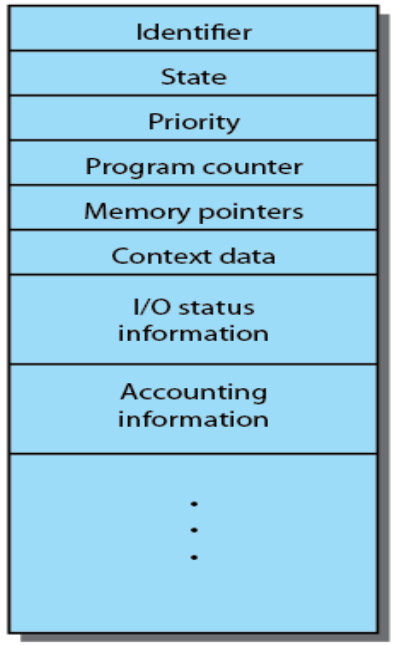


Figure 3.1 Simplified Process Control Block

- یک ساختمان داده که تمام اطلاعات مورد نیاز سیستم‌عامل در مورد یک فرآیند را دربر دارد.
- معمولا این بلوک در حافظه جانبی نگهداری می‌شود و برای اجرای فرآیند تمام یا قسمتی از آن باید در حافظه اصلی بار شود. شامل:

- شناسه‌های فرآیند
  - شناسه فرآیند
  - شناسه والد
  - شناسه کاربر
- اطلاعات وضعیت فرآیند
  - ثبات ها
  - اشاره‌گرهای حافظه
- اطلاعات کنترل فرآیند
  - حالت
  - اولویت
  - اطلاعات وضعیت I/O
  - اطلاعات حسابداری

حالت‌های اجرای پردازنده

- لازم است سیستم عامل و جداول مهم آن مثل بلوک‌های کنترل فرآیند از دخالت برنامه‌های کاربران محافظت شوند. بنابراین، دو حالت داریم:

- حالت کاربر
- حالت سیستم یا حالت کنترل یا حالت هسته
  - حالت ممتاز
  - دستورالعمل‌های خواندن و تغییر ثبات‌های کنترلی، دستورالعمل‌های اولیه ورودی/خروجی و دستورالعمل‌های مربوط به مدیریت حافظه.

- در ثبات PSW یک بیت برای تعیین حالت اجرا وجود دارد.

تعویض فرآیند
(Process Switching)

- وقفه‌های خارجی (یک عامل خارجی غیر از خود فرآیند)
  - وقفه ساعت
  - وقفه تکمیل یک ورودی/خروجی

- وقفه‌های داخلی یا تله (Trap)
  - خطاهای سخت افزاری
    - نقص حافظه
  - خطاهای برنامه
    - تقسیم به صفر

تعویض حالت
(State Switching)

- در اکثر سیستم‌های عامل، بروز وقفه الزاما منجر به تعویض فرآیند نمی‌شود. تنها با یک تعویض حالت (از حالت کاربر به هسته) جهت اجرای روتین وقفه، پردازش ادامه می‌یابد.

- بعد از اجرای گرداننده وقفه، اجرای فرآیندی که در حال اجرا بوده از سر گرفته می‌شود.

- در این حالت تنها کاری که لازم است، ذخیره‌سازی اطلاعات پردازنده در موقع وقفه و بارگذاری مجدد آن در موقع بازگشت از وقفه است.

تعویض فرآیند

- ذخیره سازی وضعیت پردازنده شامل ثبات‌ها و پرچم‌ها (متن پردازنده)
- بهنگام سازی PCB فرآیند در حال اجرا
- انتقال فرآیند به یکی از صف‌های آماده یا مسدود
- انتخاب فرآیند دیگری از صف آماده برای اجرا
- بهنگام سازی PCB فرآیند انتخاب شده
- بازیابی وضعیت پردازنده برای فرآیند جدید

تفاوت فرآیند و نخ (Thread)

- در بسیاری از سیستم عامل‌های جدید مفهوم فرآیند به دو قسمت شکسته شده است:
  - ۱- **فرآیند**
    - تملک منابع
  - ۲- **نخ** یا فرآیند سبک وزن
    - جریان اجرای دستورالعمل‌ها

- هر فرآیند دارای یکسری منابعی اختصاصی (مثل فایل‌ها و منابع I/O) است که این منابع بین نخ‌های یک فرآیند به اشتراک گذاشته می‌شوند.
- تمام نخ‌های یک فرآیند، در حالت و منابع آن فرآیند شریک هستند.
- معلق کردن یا ختم فرآیند منجر به ختم تمام نخ‌های آن فرآیند خواهد شد.

نخ‌ها و فرآیندها

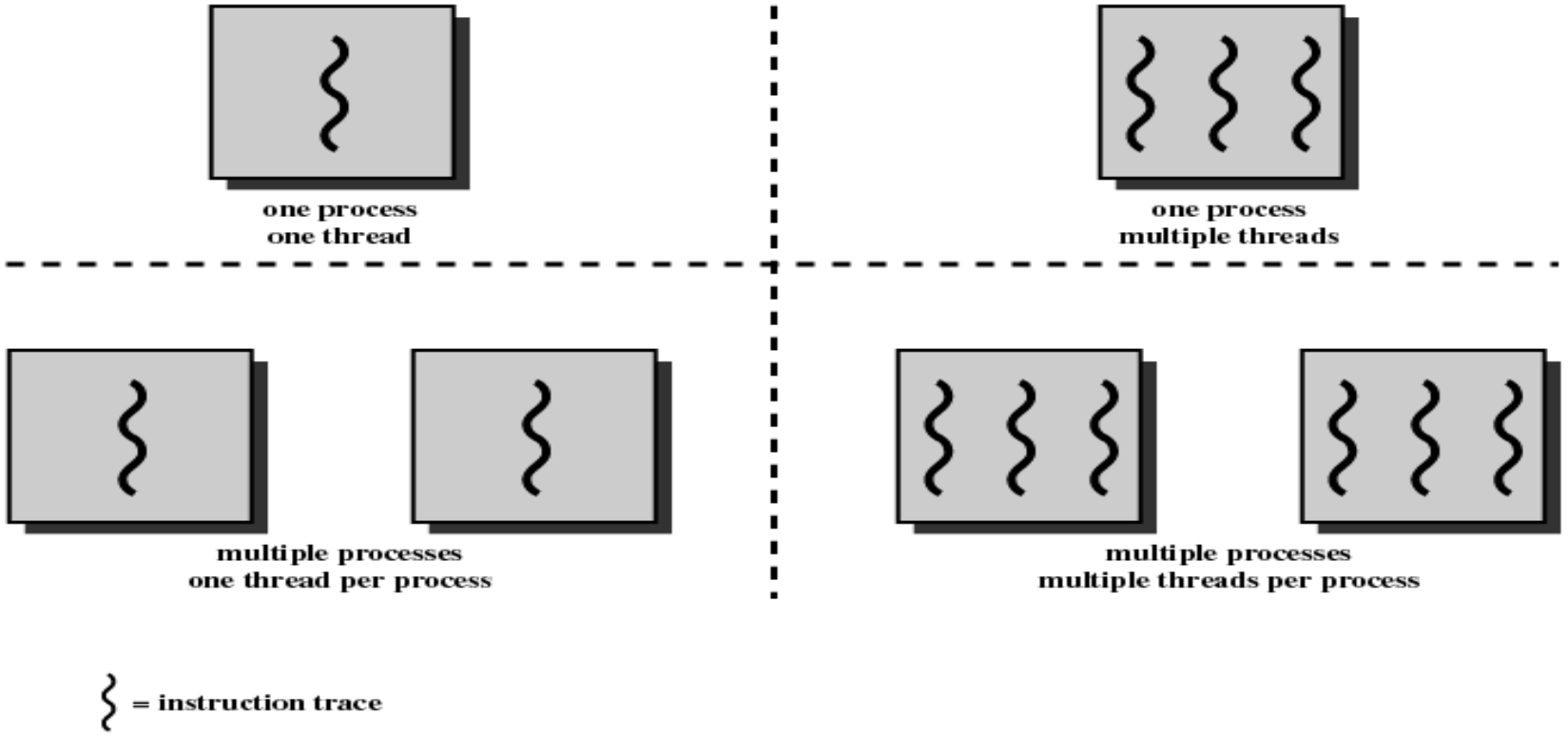


Figure 4.1 Threads and Processes [ANDE97]

مدل‌های فرآیند تک‌نخی و چندنخی

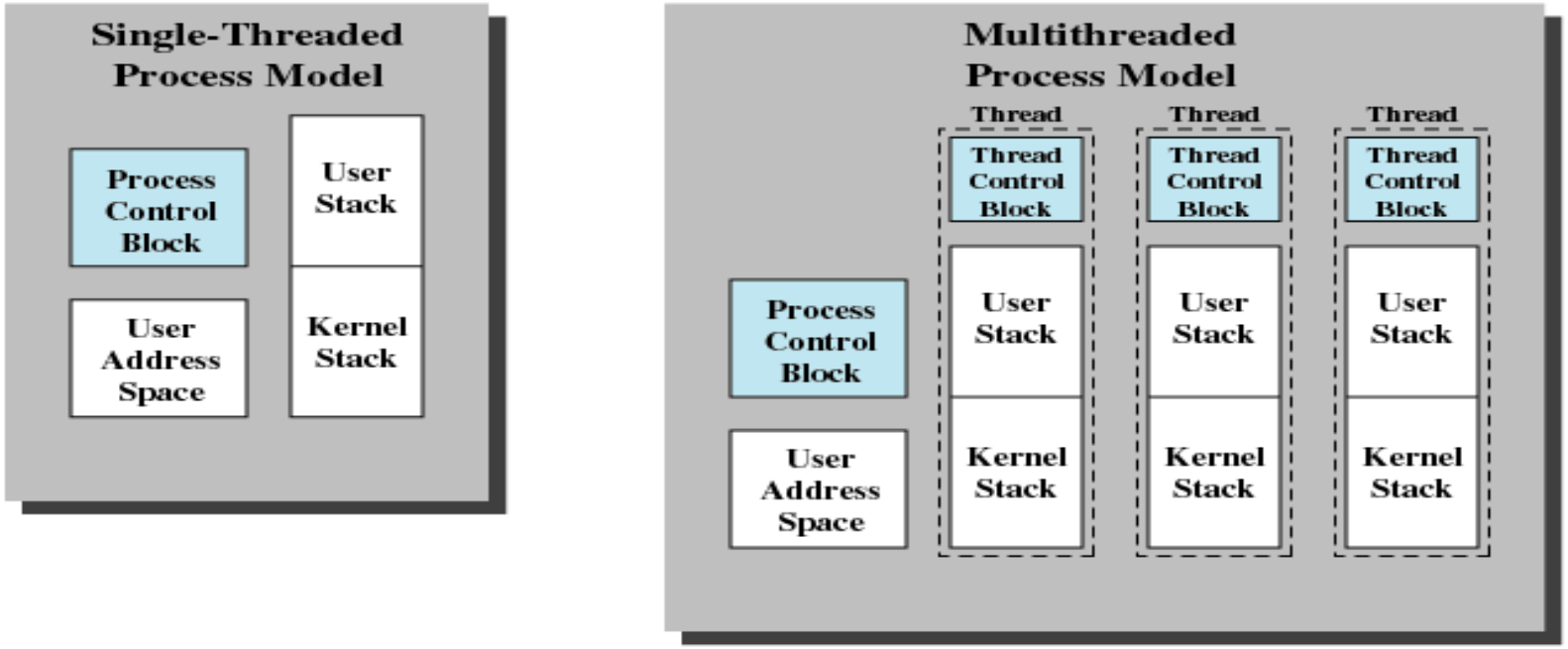


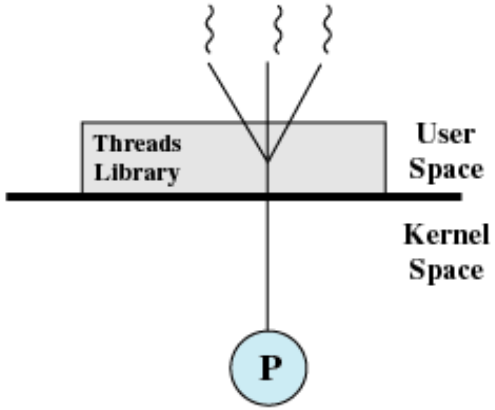
Figure 4.2 Single Threaded and Multithreaded Process Models

مزایای سیستم چند نخی

- نخ‌ها باعث افزایش کارایی سیستم می‌شوند
  - زمان ایجاد یک نخ کوتاهتر از فرآیند است.
  - زمان تعویض نخ کوتاهتر از فرآیند است.
  - بدلیل وجود منابع مشترک، ایجاد ارتباط بین نخ‌ها، بدون استفاده از امکانات داخلی هسته سیستم عامل ممکن است.

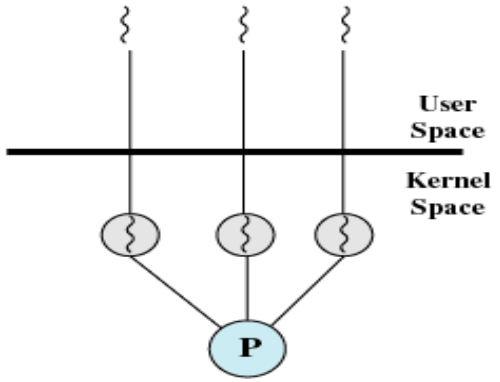
روشهای پیاده‌سازی نخ‌ها

- نخ‌های سطح کاربر
  - سیستم عامل هیچ خبری از نخ‌ها ندارد و کاربر (با کمک کامپایلر یا بدون کمک آن) نخ‌ها را ایجاد می‌کند.
  - مزیت اصلی این روش سادگی آن است.
    - تعویض نخ به حالت ممتاز (تعویض حالت) نیازی ندارد.
  - اشکال این روش این است که با مسدود شدن یک نخ کل نخ‌های آن فرآیند مسدود می‌شوند.
  - سیستم عامل Windows 98



روشهای پیاده‌سازی نخ‌ها ...

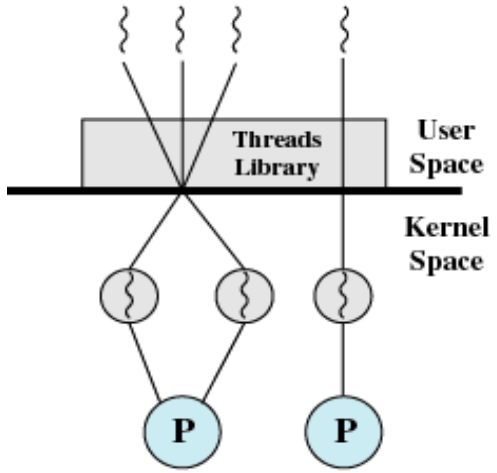
- نخ‌های سطح هسته
  - نخ‌ها در هسته سیستم عامل پیاده‌سازی می‌شوند و سیستم عامل خود آنها را مدیریت می‌کند.
  - زمانبندی بین فرآیندها انجام نمی‌شود بلکه بین نخ‌ها انجام می‌گیرد.
  - اشکال اصلی این روش اینست که انتقال کنترل از یک نخ به نخ دیگر در داخل فرآیند واحد نیازمند تغییر حالت پردازنده (از کاربر به هسته) است.
  - سیستم عامل Linux



(b) Pure kernel-level

روشهای پیاده‌سازی نخ‌ها ...

- رویکرد ترکیبی
  - ترکیبی از امکانات نخ سطح کاربر و سطح هسته
  - ایجاد نخ نهایی در فضای کاربر، مقید به یک کاربرد (Application) در فضای هسته
  - نخ‌های چندگانه سطح کاربر یک کاربرد، به تعداد کمتر یا مساوی از نخ‌های سطح هسته نگاشته می‌شوند.
  - سیستم عامل Solaris



(c) Combined