



操作系统原理



第2章 进程的描述与控制

授课教师：刘丹

计算机科学技术学院



第二章 进程的描述与控制

2.1 前趋图和程序执行

2.2 进程的描述

2.3 进程控制

2.4 进程同步

2.5 经典进程的同步问题

2.6 进程通信

2.7 线程 (Threads) 的基本概念

2.8 线程的实现





第二章 进程的描述与控制

今日学习任务整理单：

- 1.理解前趋图和程序执行的概念；
- 2.掌握程序顺序执行和并发执行的特征；
- 3.理解进程的概念；
- 4.掌握进程的状态转换； 5.理解PCB的作用。

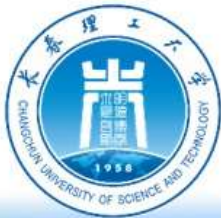
今日学习资料：

1. 雨课堂中的MOOC资源-第二章进程管理（1）
进程基本概念；
2. 雨课堂中的MOOC资源-第二章进程管理（2）
进程状态；
3. 雨课堂中的MOOC资源-第二章进程管理（3）
进程状态；
4. 教材2.1。

今日作业：学习资源1中的主观题。

07:44

3



第二章 进程的描述与控制

操作系统的复杂源于多道程序设计。

操作系统的四个特征：

并发、共享、虚拟、异步。

程序进入内存就称之为进程。

程序与进程的差别是什么？

由单道—>多道，程序执行有了什么变化？



2.1 前趋图和程序执行

2.1.1 前趋图

前趋关系 (Precedence Relation) “ \rightarrow ”

前趋图 (Precedence Graph) 是一个有向无循环图 记为

$\rightarrow = \{(P_i, P_j) \mid P_i \text{ must complete before } P_j \text{ may start}\}$,

如果 $(P_i, P_j) \in \rightarrow$, 可写成 $P_i \rightarrow P_j$, 称 P_i 是 P_j 的直接前趋, 而称 P_j 是 P_i 的直接后继。

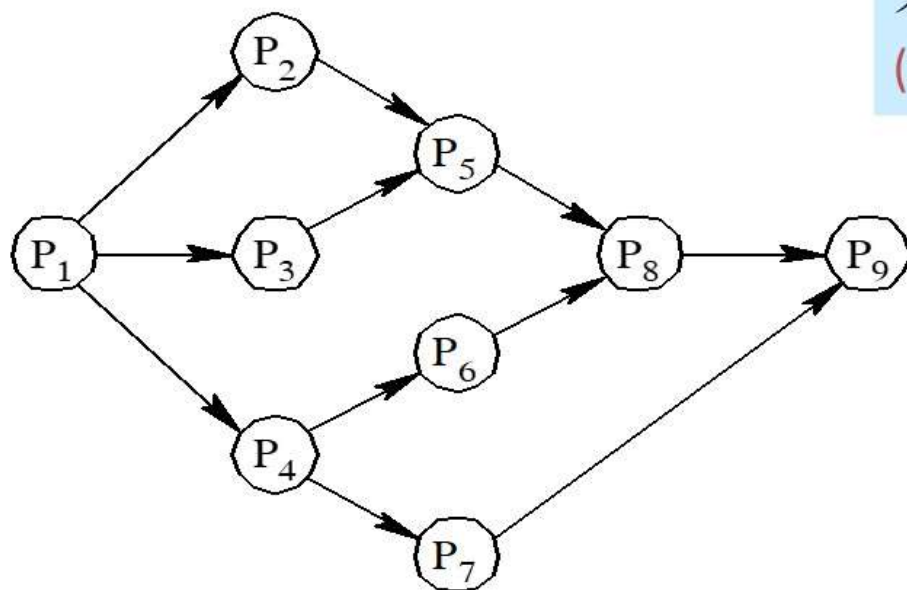
在前趋图中, 把没有前趋的结点称为初始结点 (Initial Node), 把没有后继的结点称为终止结点 (Final Node)。

每个结点还具有一个重量 (Weight), 用于表示该结点所含有的程序量或结点的执行时间。



2.1 前趋图和程序执行

2.1.1 前趋图



(a) 具有九个结点的前趋图

应当注意，前趋图中必须不存在循环，但在图2-1(b)中却有着下述的前趋关系：

$S_2 \rightarrow S_3, S_3 \rightarrow S_2$

(这种前趋关系是不可能满足的)

$P_1 \rightarrow P_2, P_1 \rightarrow P_3, P_1 \rightarrow P_4, P_2 \rightarrow P_5,$
 $P_3 \rightarrow P_5, P_4 \rightarrow P_6, P_4 \rightarrow P_7, P_5 \rightarrow P_8,$
 $P_6 \rightarrow P_8, P_7 \rightarrow P_9, P_8 \rightarrow P_9$

或表示为：

$P = \{P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9\}$
 $\rightarrow = \{(P_1, P_2), (P_1, P_3), (P_1, P_4),$
 $(P_2, P_5), (P_3, P_5), (P_4, P_6), (P_4, P_7),$
 $(P_5, P_8), (P_6, P_8), (P_7, P_9), (P_8, P_9)\}$

图 2-1 前趋图



2.1 前趋图和程序执行

2.1.2 程序顺序执行

1. 程序的顺序执行

前趋关系:

$I_i \rightarrow C_i \rightarrow P_i$ 和 $S_1 \rightarrow S_2 \rightarrow S_3$

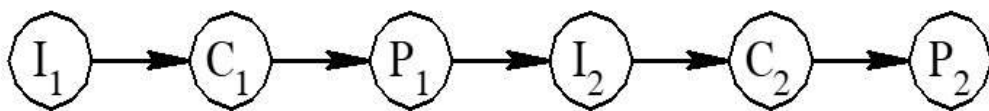
仅当前一操作(程序段)执行完后,才能执行后继操作。例如,在进行计算时,总须先输入(I)用户的程序和数据,然后进行计算(C),最后才能打印(P)计算结果。

例b:

S_1 : $a:=x+y$;

S_2 : $b:=a-5$;

S_3 : $c:=b+1$;

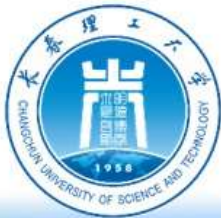


(a) 程序的顺序执行



(b) 三条语句的顺序执行

图 2-1 程序的顺序执行



2.1 前趋图和程序执行

2.1.2 程序顺序执行

1. 程序的顺序执行
2. 程序顺序执行时的特征

(1) 顺序性:

(2) 封闭性:

(3) 可再现性:



2.1 前趋图和程序执行

2.1.3 程序并发执行

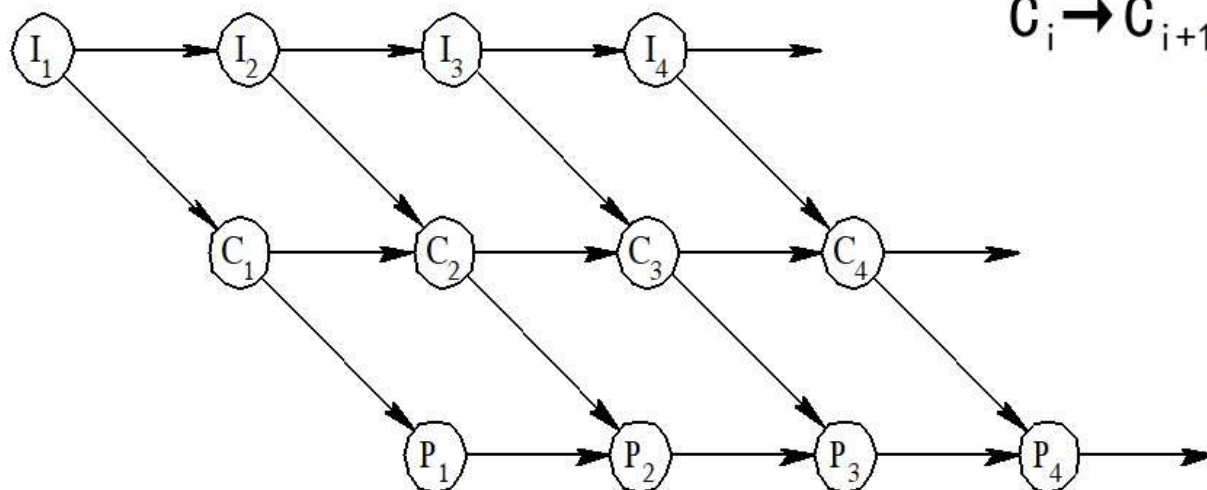
1. 程序的并发执行

批作业处理时描述为：

在该例中存在下述前趋关系：

$I_i \rightarrow C_i$, $I_i \rightarrow I_{i+1}$, $C_i \rightarrow P_i$,

$C_i \rightarrow C_{i+1}$, $P_i \rightarrow P_{i+1}$



而 I_{i+1} 和 C_i 及 P_{i-1} 是重迭的，亦即在 P_{i-1} 和 C_i 以及 I_{i+1} 之间，可以并发执行。

图 2-3 并发执行时的前趋图



2.1 前趋图和程序执行

2. 程序并发执行时的特征

例如，有两个循环程序A和B，它们共享一个变量N。程序A和B以不同的速度运行。

A程序:
Repeat
N:=N+1;
Untile false

B程序:
Repeat
Print(N);
N:=0;
Untile false

- 1) 间断性
- 2) 失去封闭性
- 3) 不可再现性

(1)执行过程:

N:=N+1

Print(N)

N:=0

输出**N+1,N=0**

(2)执行过程

Print(N)

N:=0

N:=N+1

输出**N,N=1**

(3)执行过程

Print(N)

N:=N+1

N:=0

输出**N, N=0**



2.2 进程的描述

2.2.1 进程的定义和特征

1. 进程的定义

进程控制块 (Process Control Block, **PCB**)

操作系统用于管理进程的专门的数据结构

进程实体（进程映像）就是由**代码、数据、进程控制块PCB**这三部分构成。



2.2 进程的描述

2.2.1 进程的定义和特征

1. 进程的定义

- (1) 进程是程序的一次执行。
- (2) 进程是一个程序及其数据在处理机上顺序执行时所发生的活动。
- (3) 进程是程序在一个数据集合上运行的过程，它是系统进行资源分配和调度的一个独立单位。

在引入了进程实体的概念后，我们可以把传统OS中的进程定义为：“**进程是进程实体的运行过程，是系统进行资源分配和调度的一个独立单位**”。



2.2 进程的描述

2.2.1 进程的定义和特征

1. 进程的定义

2. 进程的特征

1) 动态性

由创建而产生，由调度而执行，由撤销而消亡。

2) 并发性

多个进程实体同存于内存之中，且能在一段时间内同时执行。

3) 独立性

独立执行、独立获得资源、独立接受调度的基本单位。

4) 异步性

按各自独立的、不可预知的速度向前推进。

单选题 1分



进程是一个_____概念，而程序是一个_____的概念。

- ☒ A 动态，静态
- ☐ B 静态，动态
- ☐ C 组合态，非组合态
- ☐ D 非组合态，组合态



2.2 进程的描述

2.2.2 进程的基本状态及转换

1. 进程的三种基本状态定义

- 1) 就绪 (Ready) 状态
- 2) 执行 (Running) 状态
- 3) 阻塞 (Block) 状态



2.2 进程的描述

2.2.2 进程的基本状态及转换

1. 进程的三种基本状态定义
2. 三种基本状态的转换

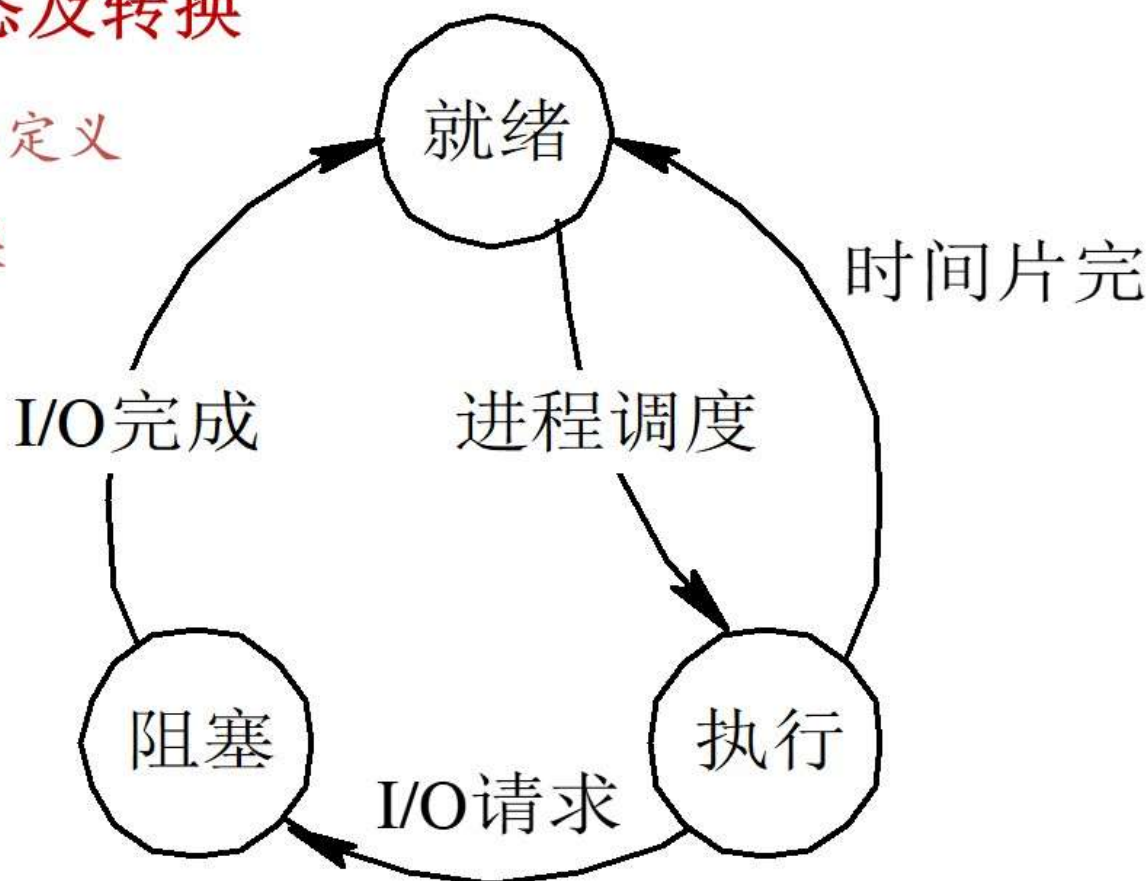


图 2-5 进程的三种基本状态及其转换



2.2 进程的描述

2.2.2 进程的基本状态及转换

1. 进程的三种基本状态定义
2. 三种基本状态的转换
3. 创建状态和终止状态

- 1) 创建状态
 - 创建工作：分配PCB，并填写必要信息；其次将其转入就绪队列。
- 2) 终止状态
 - 终止工作：OS做善后处理；PCB清空并归还给系统。



2.2 进程的描述

2.2.2 进程的基本状态及转换

3. 创建状态和终止状态

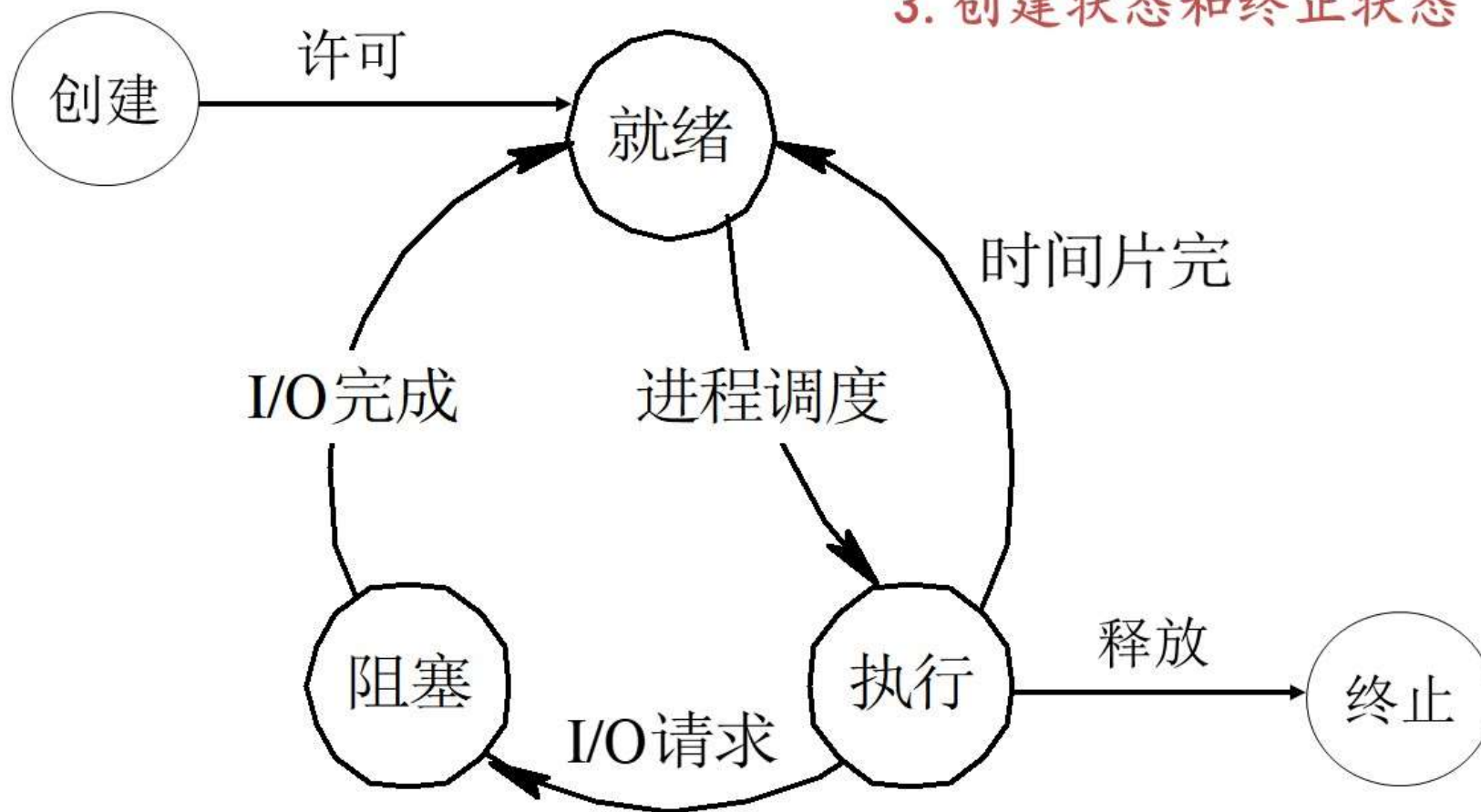


图 2-6 进程的五种基本状态及其转换

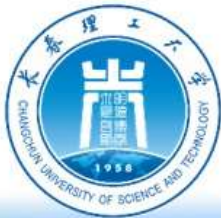


2.2 进程的描述

2.2.3 挂起操作和进程状态的转换

1. 挂起操作的引入

- (1) 终端用户的请求。
- (2) 父进程请求。
- (3) 负荷调节的需要。
- (4) 操作系统的需要。



2.2 进程的描述

2.2.3 挂起操作和进程状态的转换

1. 挂起操作的引入
2. 引入挂起原语操作后，基本进程状态的改变
 - (1) 活动就绪→静止就绪。
 - (2) 活动阻塞→静止阻塞。
 - (3) 静止就绪→活动就绪。
 - (4) 静止阻塞→活动阻塞。



2.2 进程的描述

2.2.3 挂起操作和进程状态的转换

3. 引入挂起操作后，进程状态的转换

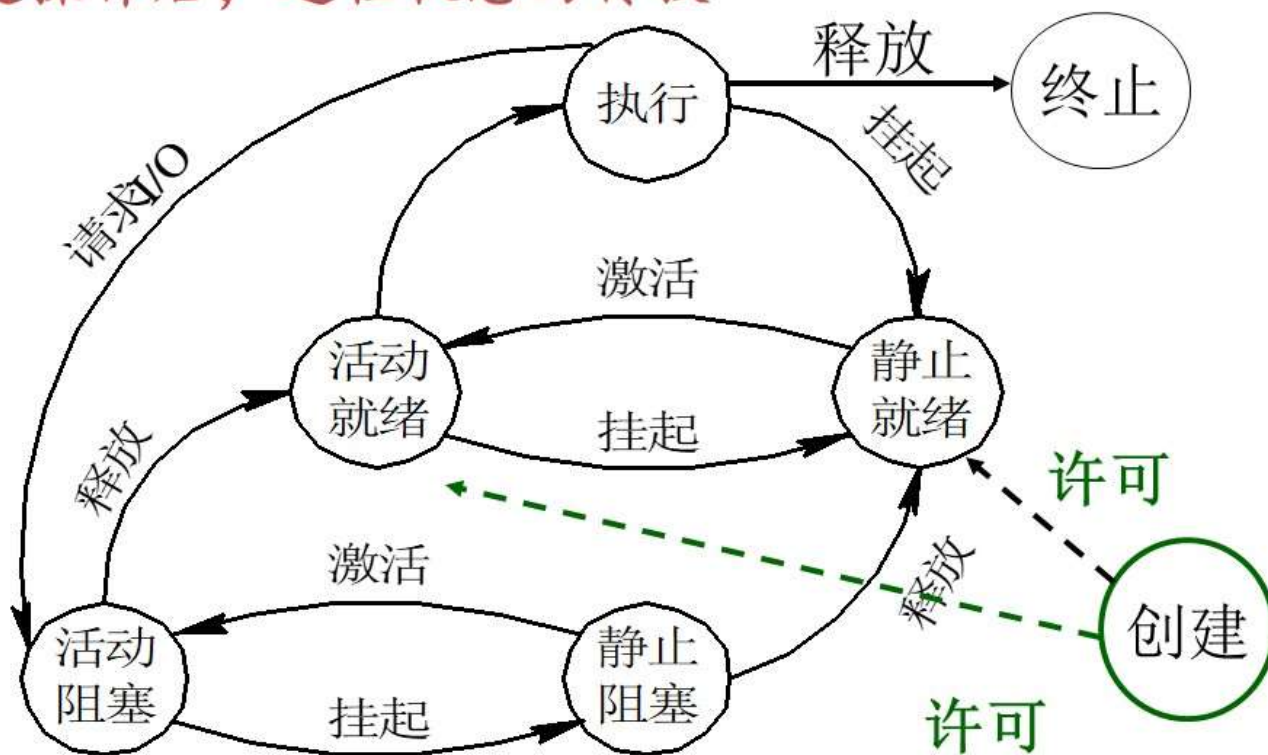


图 2-8 具有7种状态的进程状态图



2.2 进程的描述

2.2.4 进程管理中的数据结构

1. 操作系统中用于管理控制的数据结构

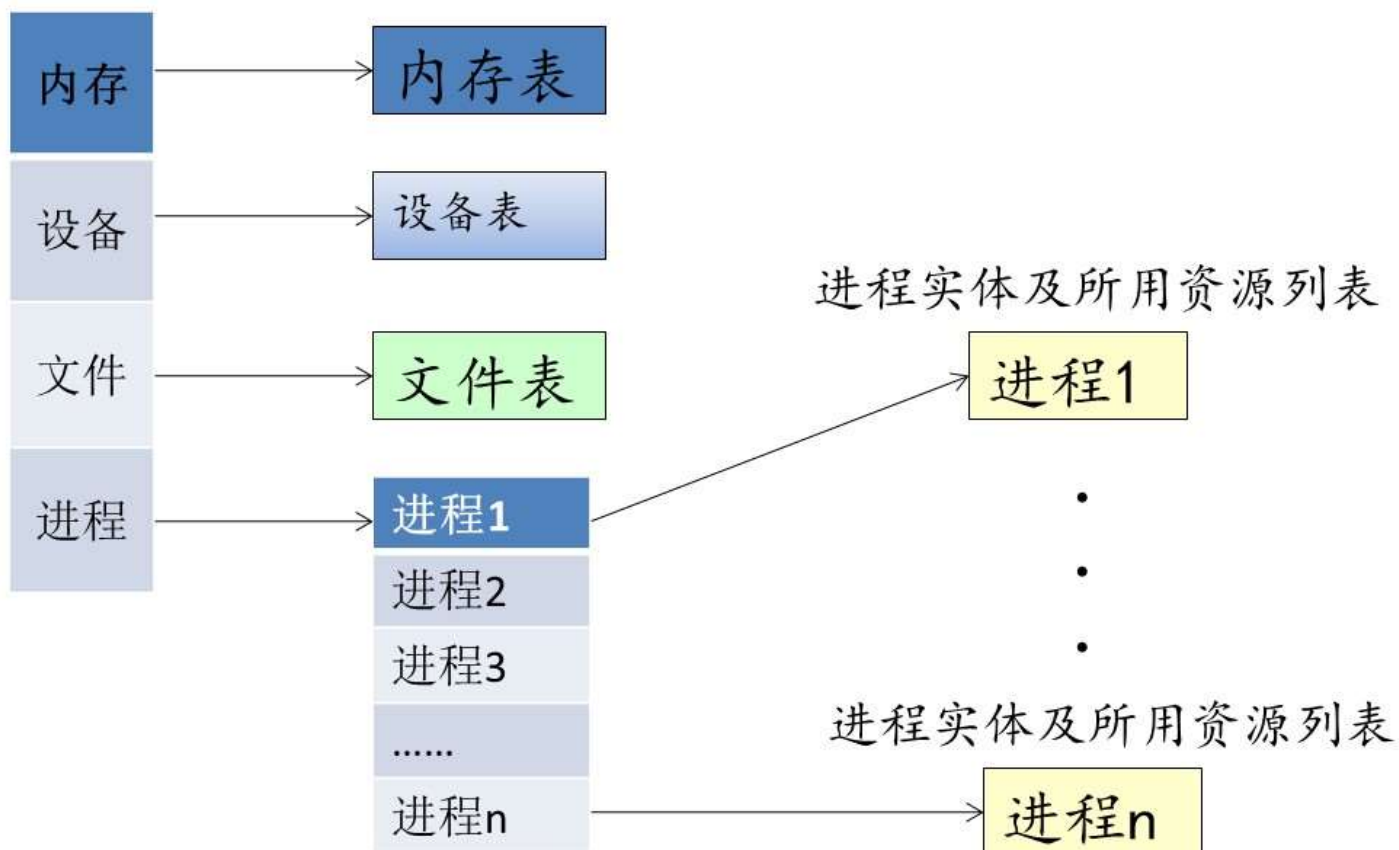
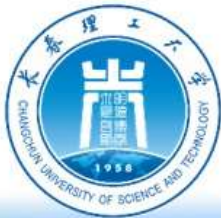


图 2.9 操作系统控制表的一般结构



2.2 进程的描述

2.2.4 进程管理中的数据结构

1. 操作系统中用于管理控制的数据结构
2. 进程控制块PCB的作用

进程控制块的作用是使一个在多道程序环境下不能独立运行的程序(含数据)，成为一个能独立运行的基本单位，一个能与其它进程并发执行的进程。或者说，**OS**是根据**PCB**来对并发执行的进程进行控制和管理。



2.2 进程的描述

2.2.4 进程管理中的数据结构

1. 操作系统中用于管理控制的数据结构

2. 进程控制块PCB的作用

- (1) 作为独立运行基本单位的标志。
- (2) 能实现间断性运行方式。
- (3) 提供进程管理所需要的信息。
- (4) 提供进程调度需要的信息。
- (5) 实现与其它进程的同步与通信。



2.2 进程的描述

2.2.4 进程管理中的数据结构

1. 操作系统中用于管理控制的数据结构
2. 进程控制块PCB的作用
3. 进程控制块中的信息

1) 进程标识符

进程标识符用于唯一地标识一个进程。一个进程通常有两种标识符：

(1) **内部标识符**。在所有的操作系统中，都为每一个进程赋予一个唯一的数字标识符，它通常是一个进程的序号。设置内部标识符主要是为了方便系统使用。

(2) **外部标识符**。它由创建者提供，通常是由字母、数字组成，往往是由用户(进程)在访问该进程时使用。为了描述进程的家族关系，还应设置父进程标识及子进程标识。此外，还可设置用户标识，以指示拥有该进程的用户。



2.2 进程的描述

2.2.4 进程管理中的数据结构

3. 进程控制块中的信息

2) 处理机状态

处理机状态信息主要是由处理机的各种寄存器中的内容组成的

- ① **通用寄存器**，又称为用户可视寄存器，它们是用用户程序可以访问的，用于暂存信息，在大多数处理机中，有 $8 \sim 32$ 个通用寄存器，在RISC结构的计算机中可超过 100 个；
- ② **指令计数器**，其中存放了要访问的下一条指令的地址；
- ③ **程序状态字PSW**，其中含有状态信息，如条件码、执行方式、中断屏蔽标志等；
- ④ **用户栈指针**，指每个用户进程都有一个或若干个与之相关的系统栈，用于存放过程和系统调用参数及调用地址。栈指针指向该栈的栈顶。



2.2 进程的描述

2.2.4 进程管理中的数据结构

3. 进程控制块中的信息

3) 进程调度信息

在PCB中还存放一些与进程调度和进程对换有关的信息，包括：

- ① **进程状态**，指明进程的当前状态，作为进程调度和对换时的依据；
- ② **进程优先级**，用于描述进程使用处理机的优先级别的一个整数，优先级高的进程应优先获得处理机；
- ③ **进程调度所需的其它信息**，它们与所采用的进程调度算法有关，比如，进程已等待CPU的时间总和、进程已执行的时间总和等；
- ④ **事件**，是指进程由执行状态转变为阻塞状态所等待发生的事件，即阻塞原因。



2.2 进程的描述

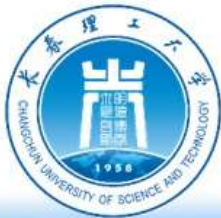
2.2.4 进程管理中的数据结构

3. 进程控制块中的信息

4) 进程控制信息

进程控制信息包括：

- ① **程序和数据的地址**，是指进程的程序和数据所在的内存或外存地(首)址，以便再调度到该进程执行时，能从PCB中找到其程序和数据；
- ② **进程同步和通信机制**，指实现进程同步和进程通信时必需的机制，如消息队列指针、信号量等，它们可能全部或部分地放在PCB中
- ③ **资源清单**，是一张列出了除CPU以外的、进程所需的全部资源及已经分配到该进程的资源清单；
- ④ **链接指针**，它给出了本进程(PCB)所在队列中的下一个进程的PCB的首地址。



2.2 进程的描述

2.2.4 进程管理中的数据结构

1. 操作系统中用于管理控制的数据结构
2. 进程控制块PCB的作用
3. 进程控制块中的信息
4. 进程控制块的组织方式

(1) 线性方式



(2) 链接方式

(3) 索引方式

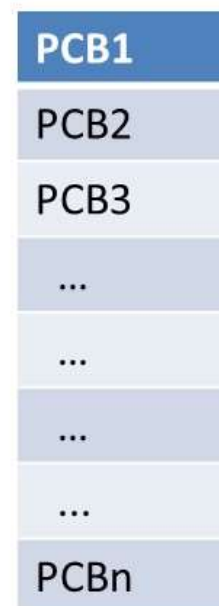


图2-10 PCB线性表示意图



2.2 进程的描述

2.2.4 进程管理中的数据结构

4. 进程控制块的组织方式

(2) 链接方式

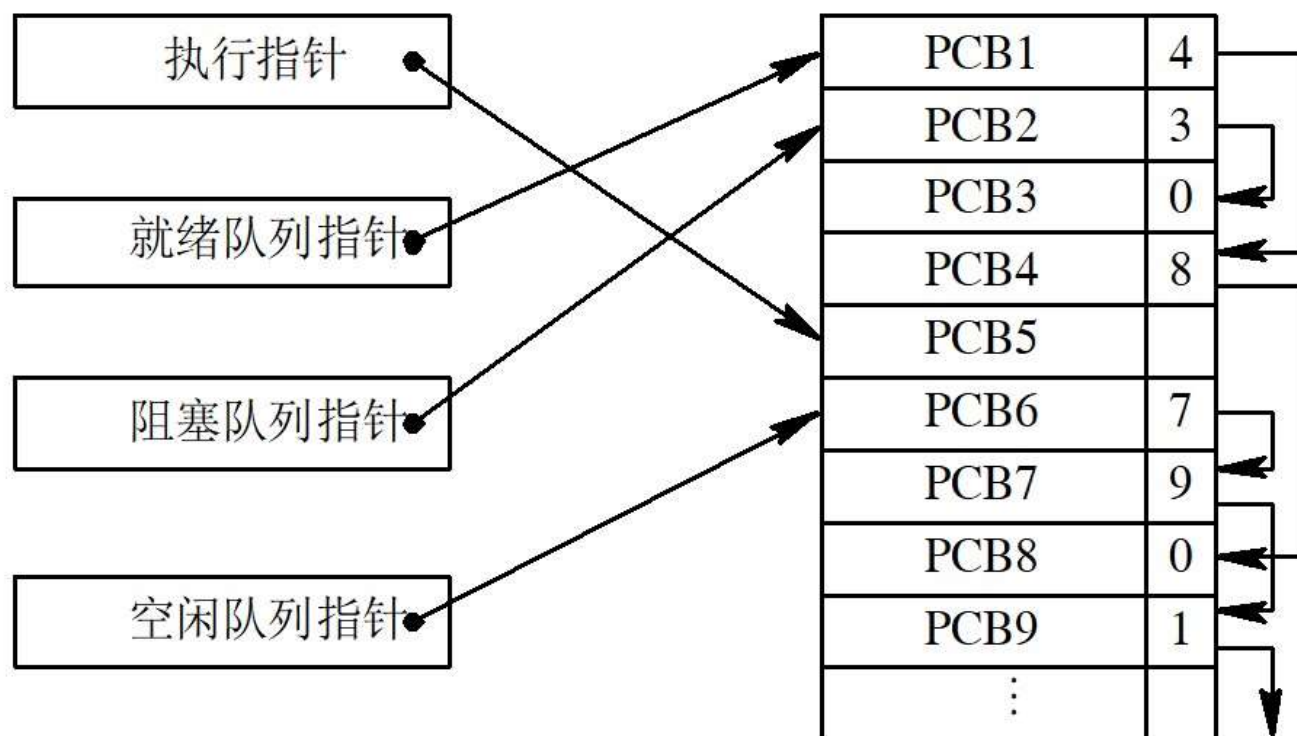


图 2-11 PCB链接队列示意图



2.2 进程的描述

2.2.4 进程管理中的数据结构

4. 进程控制块的组织方式

(3) 索引方式

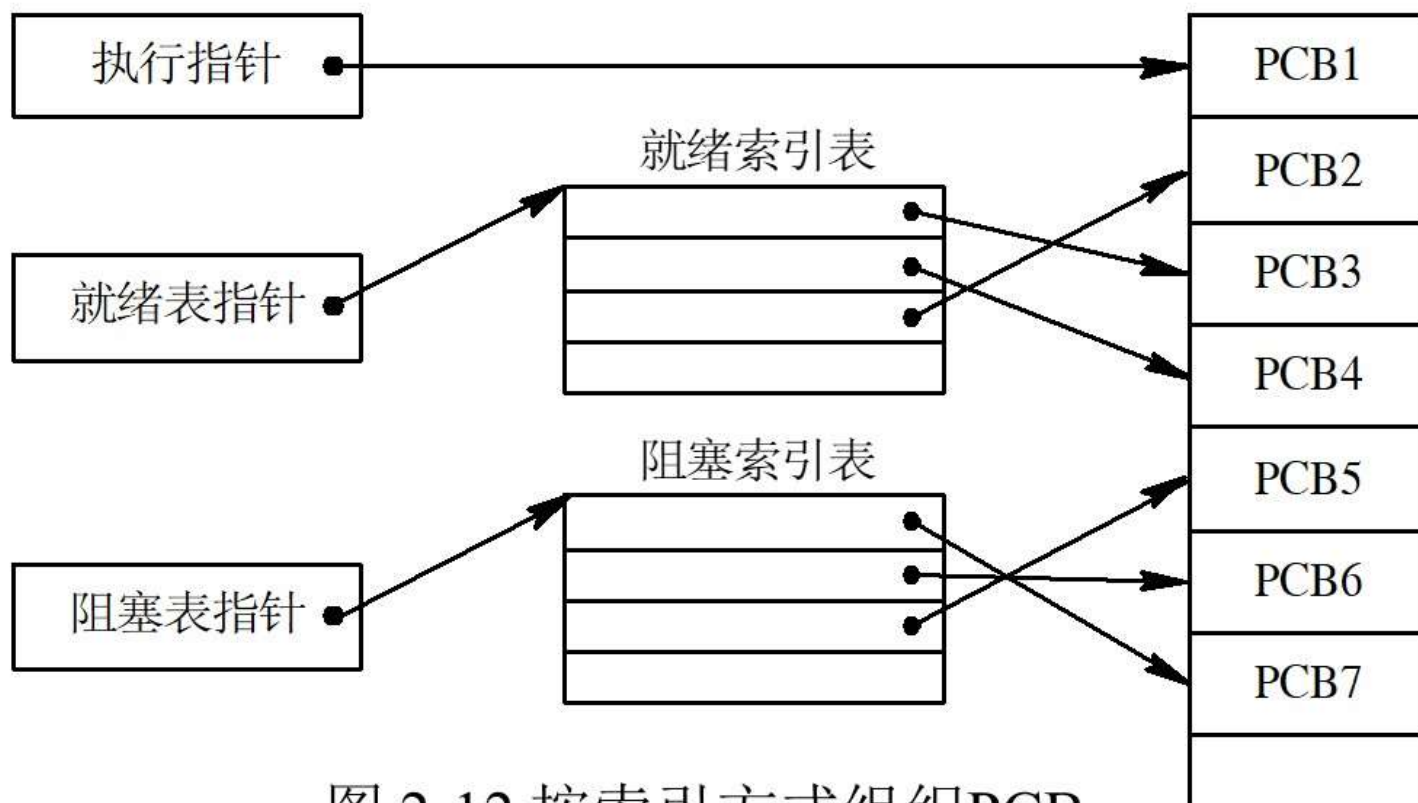


图 2-12 按索引方式组织PCB



第二章 进程的描述与控制

今日学习任务整理单：

- 1.理解前趋图和程序执行的概念；
- 2.掌握程序顺序执行和并发执行的特征；
- 3.理解进程的概念；
- 4.掌握进程的状态转换； 5.理解PCB的作用。

今日学习资料：

1. 雨课堂中的MOOC资源-第二章进程管理（1）
进程基本概念；
2. 雨课堂中的MOOC资源-第二章进程管理（2）
进程状态；
3. 雨课堂中的MOOC资源-第二章进程管理（3）
进程状态；
4. 教材2.1。

今日作业：学习资源1中的主观题。

07:44

36