

微型电子计算机

系统设计初步

(培训班用讲稿)

(中 册)

1

北京工业大学微型机中心

1982.5

第二章 操作系统的原理

§2-1 : 结 论	2-1
2-1-1 : 操作系统的任务	2-1-1
2-1-2 : 当前操作系统的分类	2-2-5
§2-2 : 从资源管理观点去分析操作系统	2-2-5
2-2-1 : 处理机管理	2-6-18
2-2-2 : 存贮器管理	2-18-28
2-2-3 : I/O设备管理	2-28-33
2-2-4 : 信息管理 (文件系统)	2-33-65

第三章 CP/M 操作系统

§3-1 : CP/M 简介	3-1
3-1-1 : CP/M 的起源及特点	3-1
3-1-2 : CP/M 操作系统结构	3-1-6
3-1-3 : 目前已配 CP/M 操作系统的 微型计算机	3-6-8
§3-2 : DYNABYTE-DOS 2.12 (由 CP/4 1.4 扩充) 的系统结构 (简称 CP/M)	3-8
3-2-1 : DYNABYTE A, B 档机的硬件环境	3-8
3-2-2 : CP/M 操作系统的引导方法	3-8-11
3-2-3 : CP/M 操作系统的数据结构	3-11-20
3-2-4 : CP/M 操作系统的内存分配	3-21-22
3-2-5 : CP/M 操作系统的系统调用	3-22-26
3-2-6 : CP/M 操作系统的程序结构	3-26-29

第四章 CP/M的BIOS模块分析

§4-1 : BIOS模块的整体结构	4-1-4
4-1-1 : BIOS模块的数据结构	4-4-10
4-1-2 : BIOS模块的程序结构	4-10-16
§4-2 : I/O设备驱动程序	4-16
4-2-1 : I/O端口地址分配 (5501)	4-16-19
4-2-2 : I/O与CP/M接口逻辑框图	4-20-23
4-2-3 : I/O子程序	4-24-26
§4-3 : 磁盘驱动程序	4-27
4-3-1 : DYNABYTE磁盘机的特点	4-27-28
4-3-2 : 磁盘控制端口分配	4-29-30
4-3-3 : 磁盘与CP/M接口逻辑框图	4-31-85
4-3-4 : 磁盘驱动程序	

第二章 操作系统的基本原理

§2-1 绪论:

2-1-1: 操作系统的任务:

操作系统是一种软件,其任务为:统管计算机系统的一切硬件和软件资源,使系统能自动地、协调地、高效率地工作。从资源管理角度来看,操作系统要进行以下几方面的管理:

- 1 存贮管理——合理使用存贮器,有多道作业时,合理地分配存贮器;
- 2 外部设备管理——合理使用外部设备;
- 3 CPU管理——合理地将CPU分配给各个作业;
- 4 信息管理——自动管理文件。

如图2-1所示。

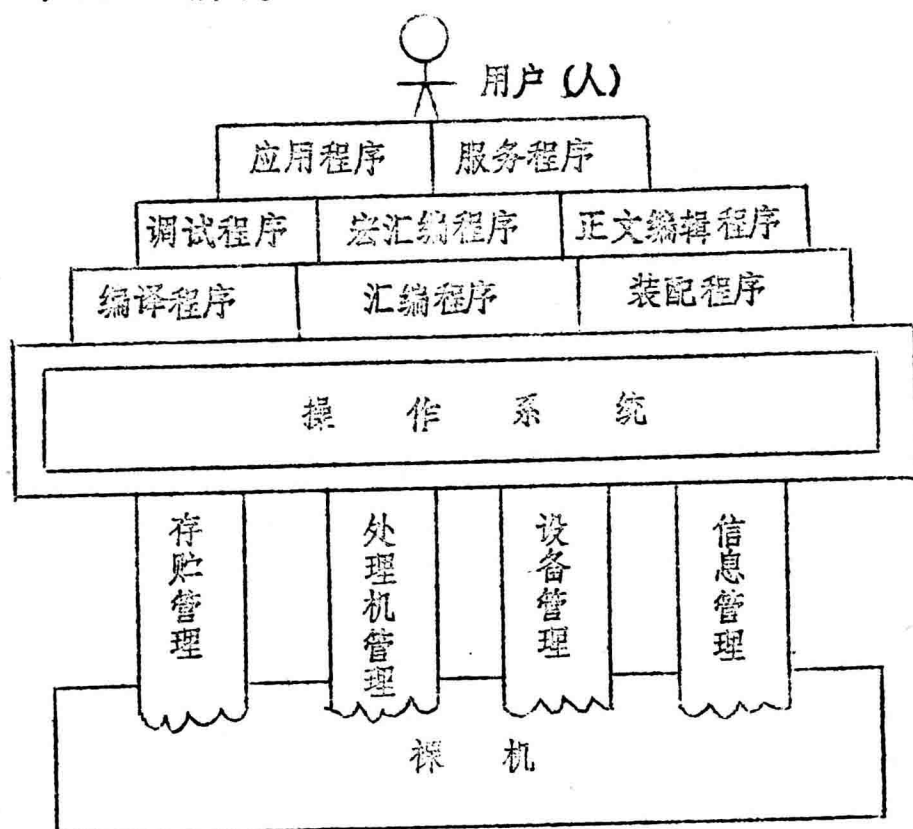


图2-1 操作系统与基本计算机软硬件软件之间的关系

有了操作系统，机器的全部资源被操作系统管起来。用户不必具体地过问各部分资源的使用情况，例如，不必为多道作业划分内存，也不必编制程序控制输入和输出。只要发出简单的命令，整个机器就会在操作系统的指挥下自动地、协调地和高效率地工作起来，而这就大大方便了用户。这时，用户所面向的不再是一部裸机（没有装入软件的硬件系统），而是面向操作系统。因此操作系统又可理解为用户与裸机之间的接口，见图2-2所示。

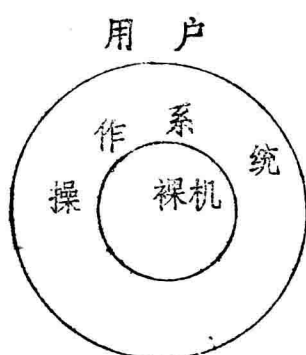


图2-2 操作系统与用户及裸机的关系

2-1-2 当前操作系统的分类：

1. 单用户的磁盘操作系统：

它实际上主要是一个磁盘文件系统，也是最简单的一种操作系统，CP/M、CDOS等均属于这一种。

2. 多道成批操作系统：

其基本思想就是主存中同时放入几道作业，且这些作业是应当互相合理搭配的。例如：可以把运算量大的科技计算作业和输入输出量大的数据处理作业搭配在一起。只有搭配得当，才能保证主机和外设高度并行地工作，充分发挥其效率。

多道成批操作系统的大致工作流程如下：

操作员可把一批批的作业随时提交给操作系统；假脱机输入程序及时地把这些作业预先输入到辅助存储器中等待运行；以后，当系统需要调入新的作业时，作业调度程序就根据系统的当时情况和各后备作业的特点，按照一定的调度原则，选择一个或多个搭配得当的作

业，调入内存准备运行；内存中的作业交替执行，当某个作业因某种原因（如输入输出完成）工作不下去时，系统就转去执行另一个作业，如此重复，直至某个作业工作完毕为止；而当某个作业完成时，系统把该作业的计算结果交给假脱机输出程序准备输出，并令资源管理程序收回该作业的全部资源。重复上述步骤，使得各作业一个接一个地流入系统，经过处理后又挨个地退出系统，从而构成了一个源源不断的作业流。

多道成批操作系统有两个特点：其一是“多道”，其二是“成批”。所谓“多道”是指内存中有多个作业同时存在，并且在辅助存储器（磁盘等）中还有大量的后备作业。因此，这种系统有相当灵活的调度原则，易于选择搭配合理的作业，从而能充分发挥系统中各种资源的利用率。“成批”的特点是在系统运行过程中不允许用户和机器之间发生交互作用。由于系统工作时不允许用户插手干预他的作业，就要求用户事先考虑到各种情况，事先规定好相应措施写在操作说明书中，所以这种系统便于实现整个计算机的工作流程自动化。

3. 分时操作系统

成批处理是一种脱机操作方式，而分时系统是高级的联机操作方式。

分时系统有三个特点，即多路调制性、交互性和独占性。分时操作系统又是多用户操作系统，它的基本思想是：一台计算机上可以挂上几个控制台和上百台终端（如电传打字机，CRT等），每个用户都可在一台终端或控制台上操纵或控制他的作业运行，这样就使很多联机用户可以同时使用一台计算机（多路调制性），并与计算机进行“对话”（交互性），而用户彼此之间都感觉不到别人也在使用计算机（独占性）。换言之，这种系统为每个用户提供的工作环境是一台可以交互会话的通用虚拟计算机。分时操作系统是把处理机时间划分成短的时间片，轮流地分配给各个联机作业使用。如果某个作业在时间片结束之前计算还未完成，那末该作业就被暂时中断并等待下一轮再继续计算，此时处理机让给另一作业使用。这样，每个用户的各次要求都能得到快速响应，因而给每个用户的印象是，好象只有他自

已独占计算机一样。

分时操作系统的优点是：第一，使用户能在较短的时间内采用交互会话工作方式，能及时地编写、调试、修改和运行自己的程序，因而加快了解题周期；第二，无论是本地用户或远地用户，只要与计算机连上一台终端设备，就可以随时随地使用计算机；分时用户之间可以通过计算机的文件系统，彼此交流程序、信息和计算结果，便于多个用户协作完成共同关心的任务。

4. 实时操作系统：

计算机硬件发展到用组件代替分立元件，而使其应用范围迅速扩大，从传统的科学计算发展到商业数据处理，深入到各行各业，例如工厂的生产控制、医疗诊断、教学及飞机、火车订票等等。这样，就出现了实时操作系统。

实时系统分成两类：一类是实时控制，如炼钢控制或医疗诊断。在实时控制系统中，计算机通过特定的外围设备与被控制的对象（炼钢炉、操作工人等）发生联系。被控制对象的信息通过外围设备传送到计算机，计算机加工之后通过显示屏幕向控制人员显示出来，或者通过外围设备向控制对象发出指示。另一类是实时信息处理，如飞机订票与情报检索就属于这一类。在实时信息处理系统中，用户通过终端设备向系统提出服务请求（预定飞机票、查阅情报资料等）、系统完成服务工作后通过终端回答用户。

实时系统对响应时间的要求比分时系统高，一旦向实时系统提出服务请求后，要求系统立即处理（要求秒级、毫秒级甚至微秒级的响应时间）。实时系统没有分时系统那样强的交互会话能力，但是它对系统的可靠性和安全性要求很高。

5. 通用操作系统：

实时系统出现后，使操作系统具备了形成通用系统的条件。为了使系统应用于广泛的范围并具有强的处理能力，通常使操作系统兼有成批、分时和实时处理中两种或三种处理能力。意即设计成通用操作系统。

例如：在实时成批处理系统中，把实时任务作为“前台”作业，

而把成批处理作业作为“后台”作业，而当系统中没有前台作业时，便立即将控制转向后台作业。

6. 网络操作系统（网络环境中的操作系统）：

没有接入网中去的计算机为单计算机。单机操作系统的最大特点是其密闭性，也就是它们有自己的用户，自己的资源以及自己独自制定的一套体制和规定。用户只能利用特定的语言和操作命令，并按系统的规定去控制作业的运行和使用各种资源。然而，计算机一旦加入计算机网络后，就要适当地改变这种特性，以适应同网中更多的系统和用户进行交往，从而变为开放式的面向网络的计算机操作系统。入网后的计算机系统，由于通过通讯子网和网中各种资源相连，所以，它不但大大地扩大了本机用户的可用资源范围，而且还使自己的用户范围从本机用户扩大到网际用户。这就要求网络环境中的操作系统既要为本机用户提供简便有效的使用网络资源的手段，又要为网络用户提供使用本机资源的服务。为了实现这一要求，网络中的操作系统除了配置普通操作系统所具备的模块（如：核心、文件管理、作业监督、操作管理等）外，还要配置一个网络通信模块。该模块是操作系统和网络之间的接口模块。相接的两个介面分别为网络接口介面和操作系统接口介面。

网络通信模块的主要功能是支持终端与计算机、计算机与计算机以及计算机与通信网络之间的通信，并实现用户命令／响应与协议命令／响应之间的翻译和转换工作。该模块是由通信接口中断处理程序、通信控制程序以及各级网络协议软件组成的。

§ 2-2：从资源管理观点去分析操作系统：

操作系统要管理很多的硬设备和软设备，这些硬软设备统称为“资源”。按其性质来分，可把资源分为四大类，即处理机、存储器、外部设备以及信息（程序和数据）。这四类资源构成了操作系统本身和用户作业赖以活动的物质基础及工作环境。它们的使用方法和策略决定了整个操作系统的规模、类型、功能以及实现。所以，引入资源管理观点的重要作用之一，就在于我们能够应用这一观点去组织

操作系统的有关内容。基于这一观点，我们可以把整个操作系统看成是由一组资源管理程序所组成的。对应于上述四类资源，可以把操作系统划分成处理机管理、存贮管理、设备管理和信息管理（即文件系统）这四大部分而去加以分析和研究。

2-2-1: 处理机管理：

1. 进程：在早期的计算机中，对处理机不需要进行管理，那时采用单道的使用方法，即每一次只有一个用户使用计算机；而在使用机器的整个过程中，计算机的一切资源（包括中央处理机在内）都为该用户所独占。CPU大部分时间在执行用户程序，但当用户程序请求输入、输出时，CPU就执行管理程序，以便完成输入、输出工作。输入、输出结束后，仍返回用户程序执行。用户程序进入或退出管理程序，只要经过一个简单的接口程序，无须设置专门的处理机管理程序。CP/M, CDOS 这类的单用户操作系统就没有处理机管理程序而只有简单的 I/O 接口程序。

为了充分发挥计算机的能力，于硬件中引入了 I/O 通道中断系统。为了使 CPU 和 I/O 通道并行工作并充分发挥计算机各部件的能力而引入了多道程序设计，即在一个时刻有几个作业同时在计算机内运行。在这种情况下，各个作业都需要有 CPU 使其运行。但一般说来，在一个计算机系统中，处理机的数目总是少于同时运行的用户作业数，并且往往只有一台处理机；这样，就产生了 CPU 的分配问题，而最简单的办法是，把 CPU 的时间轮流分给各个作业，但在实际应用中往往需要根据不同的目标采用不同的 CPU 分配办法。

例如：在会话或分时系统中，当用户提出一次服务请求时，系统就得快速地响应用户要求，以使得用户感觉到好象整个计算机只有他一人在使用，又如，对于实时系统来说，响应的速度的要求就更高了，这更需要合理地调配 CPU 了。

结论是：系统中有多多个作业同时运行时，都必须有一个处理机管理机构，以便对处理机作妥善管理，满足各类用户的特殊要求和实现系统的设计目标。

那么操作系统中的处理机管理部分如何管理 CPU 呢？我们通过

一个例子来说明CPU管理的机构及其基本工作原理，参见图2—3。

图2—3上半部份给出了成批处理作业的计算流程。一个成批作业从进入系统到退出系统大致分成四个阶段：

(1) 程序员把他作业的全部穿孔信息【包括程序数据和操作说明书】交给机房；

(2) 由机房操作员通过输入机把作业信息送入机内，并由系统将其安置在磁盘的某些盘区（这种盘区常称为“输入井”）中等待运行。在“输入井”中可以堆积许多等待运行的作业；

(3) 系统从这些等待运行的各作业中挑出若干个作业分配给它们所需的资源，让它们同时投入运行；

(4) 作业运行结束时，系统收回该作业所占用的资源，并令其退出系统。

为了刻划作业在上述流程中所处的不同阶段，相应地引入下述四种状态（图2—3的下半部给出）：

(1) 从程序员把作业提交给机房到该作业的全部信息输入到“输入井”为止，该作业处于“提交状态。”

(2) 从作业信息输入到“输入井”开始，到该作业被选中因而进入内存运行为止，该作业处于“后备状态”。

(3) 从作业进入内存运行起，到作业计算完成为止，该作业处于“运行状态”。

(4) 从作业计算完成开始，到善后处理完毕并退出系统为止，该作业处于“完成状态”。

作业的各种状态之间的改变是靠操作系统的专门机构实现的。从“提交状态”到“后备状态”的转变乃是由假脱机系统（SPOOLING系统）完成的，它属于设备管理的范围，这里不说明了。

从“后备状态”到“运行状态”的转变以及从“运行状态”到“完成状态”的转变都是由处理机管理中的作业调度程序完成的。

作业调度程序为了完成作业从“后备状态”到“运行状态”的转变，它首先从后备作业中挑选出一个或几个作业，然后为每个被选中的作业做好下述两项准备工作：第一，为每个作业建立一组“进程”，

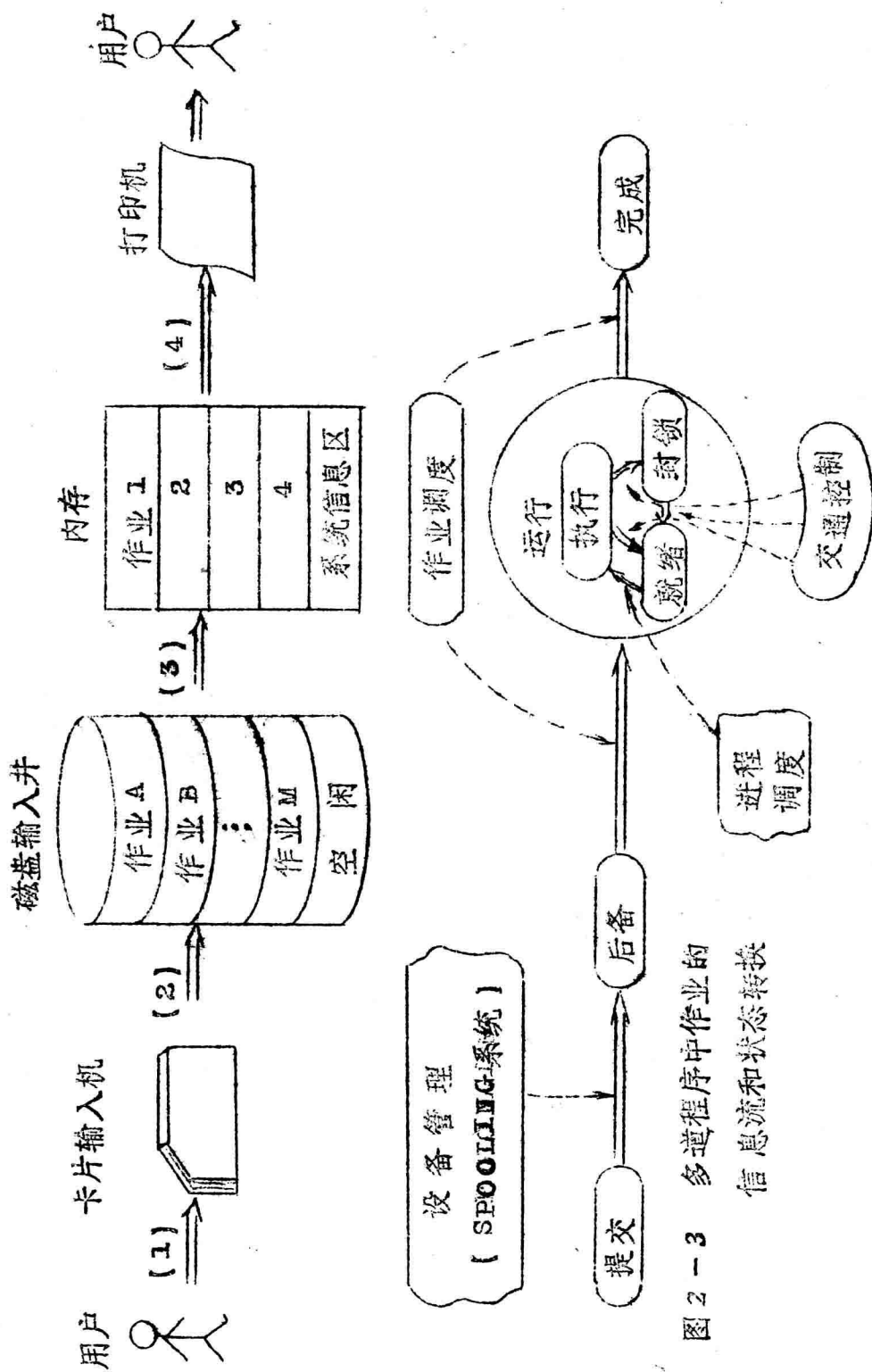


图 2-3 多道程序中作业的信息流和状态转换

即是说，为每个作业建立一个用户进程，同时建立若干个为这个用户进程服务的系统进程（例如：替用户完成输入、输出工作的 I/O 进程）。这组进程在处理机管理程序的控制下协同运行，完成相应作业的计算任务。第二，为被选中的作业分配资源。

作业调度程序为了保证每个被选中的作业能够运行，除了应给它分配内存和外设等资源外，还应给它分配 CPU。但是，由于个数少于运行的作业数，甚至只有一个 CPU，则作业调度程序不可能给每个运行作业分配一台实际的 CPU；而为了解决这一矛盾，可把处理机管理程序分成作业调度程序和进程管理程序两部分，使处理机的管理工作分两级进行。作业调度程序在选中一个“后备作业”并让它转为运行时，只保证该作业就得实际 CPU 的权力，即使该作业具备获得 CPU 的资格，而真正分配 CPU 的工作是交给比它低一级的程序（进程管理程序）去完成的。

进程管理程序的管理对象是“进程”，而为了弄清是如何管理进程的，先要对进程的概念加以简述。

由于多道作业的运行，而仅用“程序”这个概念就不能说明问题了，还要求引入“进程”（PROCESS）的概念。现通过两个例子来说明：

例 1：有两个作业 A 和 B 都要求用 BASIC 的解释程序解释并执行。操作系统使每个程序运行一定的时间（比如 100 毫秒）。

“A 作业”解释并运行 100 毫秒（完成一“进程”）

“B 作业”解释并运行 100 毫秒（完成一“进程”）

“A 作业”解释并运行 100 毫秒（完成一“进程”）

“B 作业”解释并运行 100 毫秒（完成一“进程”）

·
·
·
·
·
·

对 A 和 B 作业解释并运行 100 毫秒，这不能说是完成一个程序，而只能说是一个“进程”。

例2：有两个作业C和D都要求调用宽行打印机程序。执行的办法是先打印C作业，再打印B作业。这称为完成了两个“进程”。

通过以上例子，可看出“进程”有以下特点：

- “进程”与程序相对应。没有程序的执行，就没有“进程”。所对应的程序执行了一次，不管程序是否完成，称为完成一个“进程”；
- “进程”与多道程序有关。没有多道程序就无所谓“进程”；
- 程序是完成某一任务而设计的指令的集合，它是静态的；“进程”则是程序的一次执行过程，它是动态的；
- 一个作业，根据情况，可由一个或多个“进程”来完成，通常往往要多个“进程”来完成。

当代操作系统的一个重要特点就是并发性，而所谓并发性，是指操作系统控制很多能并发执行的程序段。这些并发程序段是一些“松散”联系的个体，在大部分时间中，它们彼此独立并发地执行自己的工作，但有时彼此之间都要以直接或间接方式发生着相互依赖、相互制约的关系。直接方式的制约关系通常是在彼此有逻辑关系的程序段之间发生的。例如：一个程序段等待另一个程序段的计算结果或信息。只有当另一程序段送来相应的计算结果和信息时，这个程序段才能继续工作下去。

间接方式的制约关系通常是由于并发程序段竞争相同的资源而产生的。得到资源的程序段可以继续执行，而得不到资源的程序段就暂时挂起，等到有可用的资源时再继续工作。

正是因为这些可以并发执行的程序段之间存在着这样或那样的相互制约关系，所以每个程序段不能与外界隔绝，也不能随心所欲地在处理机上运行。它不仅要受其它程序段的活动所制约，而且还要动态地依赖系统资源的分配情况。因此，每个可以并发执行的程序段就会因外界条件的不同而处于不同的状态，有时处于可运行状态；有时就因为等待某种资源或其它程序段的信息而不能运行，即处于封锁（挂起）状态。这样，对于这些可以并发执行的程序段来说，只用“程序”这一概念就不够用了，而使用程序这个概念，只能对可并发

执行的程序进行简单、孤立、静止地研究分析，因而不能深刻揭示它们的活动联系及其动态变化。因此，我们必须从变化的角度，动态地分析研究这些可以并发执行的程序段，为此，在操作系统中就引入了“进程”的概念。

每个进程在其运行过程中可能经历几种不同的状态。如

- 正获得CPU控制权（即真正在执行）的进程称为处于“执行状态”。若只有一台处理机，那么在任一时刻处于执行状态的进程只有一个，其它进程则在等待执行。

- 在等待的进程中，又有两种情况：

- 其一是：一些进程的一切执行条件皆具备，只是因为尚未获得CPU控制权而暂时等待，这种进程称为处于“就绪状态”。

- 另一是：一些进程因为某种原因（如用户进程等待输入、输出工作的完成）暂时无法执行，这种进程称为“封锁（挂起）状态”。进程管理程序就是完成进程上述状态的转变的。

为了在计算机中刻画这种动态变化的进程，通常把进程分成三部分，一部分是程序，一部分是数据空间，另一部分称做进程控制块（PCB）。见图 2-4(a)(b)。程序部分描述了进程本身所要完成的功能；数据空间是程序进行加工的场所；而进程控制块则包含了进程的有关信息，它们表达了进程在当前时刻的状态以及它与其它进程和资源的关系。进程控制块所包括的信息有：进程的名字、当前状态、所需的资源和已分配的资源、调度信息、通讯信息、指向其它进程控制块的连接字以及会计信息等等。

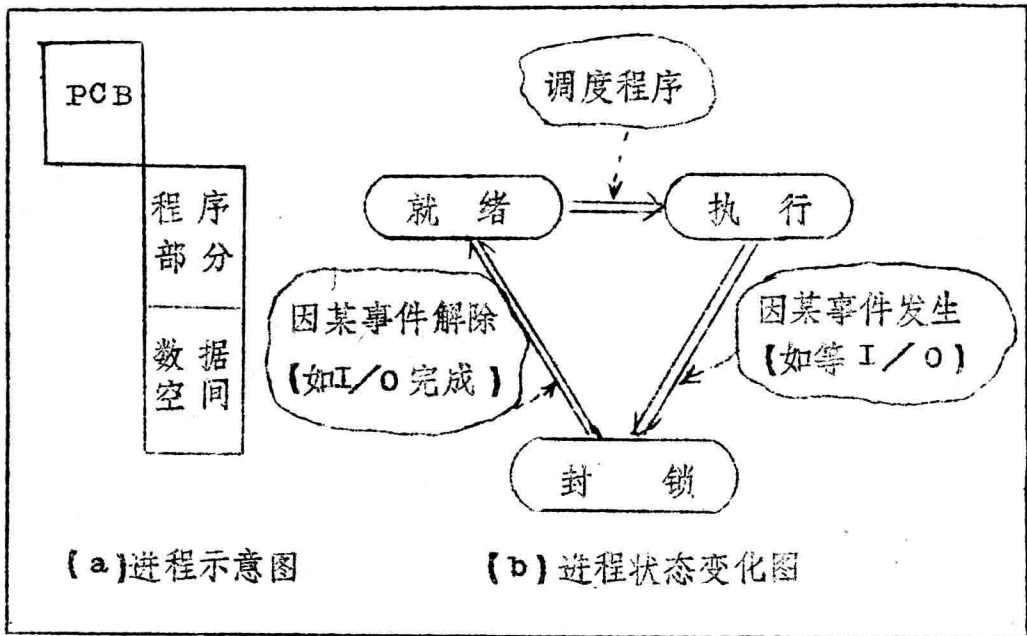


图 2-4 进程示意及状态变化

2. 处理机管理的三个主要机构——作业调度、进程调度和交通控制。

- 作业调度：这是粗的调度，用来确定哪一个作业将进入运转状态；
- 进程调度：这是细的调度，用来确定哪些作业中的哪些进程将进入运行状态；
- 交通控制：即运行→就绪，执行→封锁（挂起），封锁（挂起）→就绪三种状态变化的控制。

下面分别对这三种机构进行说明。

1) 作业调度：

① 作业调度的职能是完成作业从后备状态到运行状态的转变和从运行状态到完成状态的转变，见图 2-3。为完成这一任务，作业调度程序应包括以下四个职能：

a) 记录已进入系统的各作业的情况，为此系统的每一个已进入系统的作业分配一个作业控制块 JCB，每个作业的 JCB 在该作业进入后备状态时由系统建立，在该作业退出系统时撤消。每个作业在各阶段的情况【包括分配的资源 and 状态等】都记录在它的 JCB 中，作业

调度程序就是凭据各个作业的JCB提供的信息对作业进行调度和管理
的。

b. 从后备作业中挑出一些作业来投入运行：系统中处于后备状态的作业比较多，有几十个甚至上百个；个数取决于存放后备作业的输入井空间的大小，但是处于运行状态的作业一般只是有限的几个，最多不超过四个或八个。要求按一定的调度原则从后备作业中挑出若干个作业来投入运行，因此要综合平衡各种因素，确定出能实现设计目标的合理的调度算法。

c. 为被选中的作业做好运行前的准备工作：

作业调度程序在让一个作业从后备状态进入运行状态之前，必须为该作业建立相应的进程，并为这些进程提供资源（内存和外设）。只保证此作业有获得使用CPU的资格，而CPU时间的真正分配工作乃是由进程调度程序完成的。

d. 在作业结束时做善后处理工作：

在一个作业运行结束时，作业调度程序要把相应作业的一些必要信息（如运行时间、作业执行情况等）进行输出，然后再收回该作业所占用的全部资源，撤消与该作业有关的全部进程和该作业的作业控制块。

②作业控制块JCB：

JCB是一个作业存在的标志。每个JCB记录与该作业有关的信息，而具体的内容有所不同，表2-1列出了JCB的主要内容。

表 2-1 作业控制块 JCB 之例

作业名字	
资源 要求	要求运行时间
	最迟完成时间
	要求的内存量
	要求外设的类型及台数
	要求的文件量和输出量
资源 使用 情况	进入系统时间
	开始运行时间
	已运行时间
	内存地址
	外设台号
类型、 级别	控制方式
	作业类型
	优先数
状 态	

③作业调度算法：

作业调度算法规定了从已接纳的后备作业中选择运行作业的原则。目前有几种不同的作业调度算法如下：

a. “先到先服务”调度算法：

其优点是处理简单，缺点是由于没有考虑各个作业运行特性和资源要求的差异，因而影响了系统效率的发挥。

b. “优先数”调度算法：