操作系統概述

国营二六二厂情报室

一九七九年四月

目　录

说　　明

二机部自动化情报网举办的第三期电子计算机培训班的主要目的，是学习小型DJS-140系列机上、实时操作系统。为了搞好这一学习，商定在实时操作系统讲述之前，用不长的时间，简要的学习一下一般操作系统的基本概念，以便为后面的学习打下一定的基础。考虑到讲述时间不能很长，另一方面又要有主要内容，为此，编写这份简要讲义，以方便大家学习。但是，由于本人在这方面的知识肤浅，水平实在有限。又加之编写时间仓促，定有错误与不当之处，请同志们热情的提出宝贵意见，以便今后改正，更好的完成这方面的工作。

第一章　概述

第一节　操作系统的由来

1. 计算机软件发展概况

在50年代末期以前的以电子管为主要特性的第一代计算机中，软件处于初期的发展阶段。这期间，从机器语言发展到符号语言，汇编语言已完成。从50年代末期到60年代中期的以晶体管为主要特性的第二代计算机中，软件得到了很快的发展。先后出现了FORTRAN语言和ALGOL语言，从而完成了编译程序。此外，相继出现了批号处理、执行系统以及多道、分时系统。多道和分时的出现标志着操作系统真正形成。从50年代末期开始，集成电路研制成功，60年代中期计算机进入第三代，软件得到更大的发展，出现了许多通用和专用高级语言，多道程序和分时也发展到比较成熟的阶段。操作系统在使用上开始普及，通用的操作系统已完成，并且出现了实时系统，远程批号处理和计算机网。60年代末期，由于操作系统进一步发展的复杂性，促使操作系统开始总结提高，进入理论化阶段。提出了操作系统的一般设计原则，并且对各种算法作理论分析。也出现了设计操作系统使用的语言：PASCAL语言。以上阶段可列表如下：

阶　段	硬件特性	软件情况
第一代（50年代末期以前）	电子管	从机四语言到符号语言、汇编语言完成
第二代（50年代末到60年代中期	晶体管	编译程序完成 采用批易处理和执行系统、多道、分时开始出现
第三代（60年代中期到68年）	集成电路	操作系统普及化 通用操作系统完成 计算机网
第三代后期（68年至今）	集成电路	操作系统理论化阶段

2. 早期电子计算机的运行情况

早期的电子计算机并没有配置多少软件，更谈不上有操作系统了。操作系统是随着计算机的不断发展而产生的。我们现在来

看一下操作系统是针对解决计算机发展中的什么矛盾而提出来的，便可以帮助我们明了操作系统的目的、功能，以及操作系统是一个什么东西了。

首先，让我们来了解一下早期计算机的运行和使用情况。

早期的计算机，主要由中央处理机（CPU），内存贮器和外设备组成。中央处理机主要由运算器和控制器组成。国内有时把中央处理机与内存贮器合起来称为主机。外设备主要指输入设备（如纸带输入机）和输出设备（如行式打印机）。早期计算机的运行情况是：中央处理机通过输入设备把程序和数据输入到内存贮器，然后进行运算，接着又通过输出设备把计算结果从内存贮器输送出来。图1.1画出了这种计算机的工作流程。图1.2描述了这种流程的时间关系。假如输入所需的时间为t_1，运算所需的时间为t_2，输出所需的时间为t_3。输入结束后，才能进行运算，运算结束后才能进行输出。因此，总共所需的时间为$T = t_1 + t_2 + t_3$。这种运行方式的特点是输入——运算——输出是串行的，也就是这三个步骤在时间上是不重迭的。

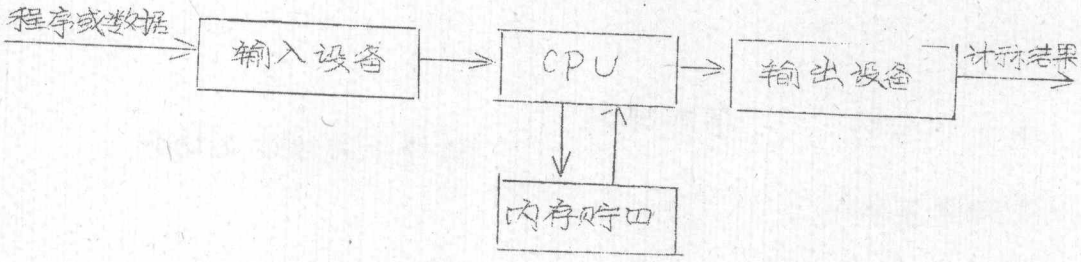

图1.1 早期计算机的工作流程图

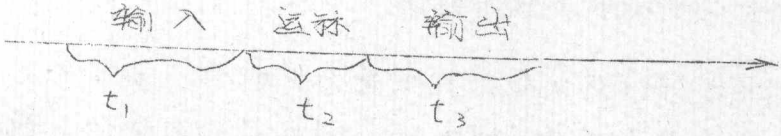

图1.2 早期计算机工作流程时间图

早期的计算机基本上是人工操作的。用户上机计题时，用手工拨弄控制台上的各种按钮来指挥计算机执行程序，各种外设备的动作也是直接通过控制台开关手动操作的。

2、电子计算机使用中的矛盾

上述计算机的串行方式及手工操作，在运算速度不是很高的初期，没有表现出什么明显的矛盾。但是，随着计算机应用的要求与发展，主机速度得到迅速的提高。此时，在计算机的使用过程中，早期的串行方式与手工操作方式越来越明显的暴露出两个矛盾。

第一个矛盾是外设备与主机速度不相匹配的矛盾。在早期，尽管外设备传输数据的速度比起主机的运算速度来要慢许多，但是，由于主机运算速度本身就不是太高，所以这一矛盾在当时并不突出。当主机的速度从原来的每秒几千次提高到每秒几百万次，几千万次，虽然外设备的速度也在不断提高，但比起主机提高速度的数级相差很大。因此，主机和外设备速度之间的矛盾就越来越突出了。

这样一来，在等待外设备传输数据时，高速的主机白白的空闲着，其结果计算机的实际利用率却大大下降了。

举一个例子来说明。一个计题程序在主机速度为每秒5千次的计算机上运算时间需要30分钟，假设输入数据需3分钟，则主机空闲的时间占全个时间约90%。现在，同一计题程序改在主机速度为每秒50万次的计算机上做，运算时间仅需0.3分钟就够了，输入数据的时间还是3分钟。此时，主机空闲的时间则占全个时间约90%。随着主机速度再提高，这个百分比还要增大，这就是因为外设备与主机速度不相匹配而引起了主机利用率的大大下降，使昂贵的高速主机造成了极大的浪费。

第二个矛盾是手工操作与主机运算速度不断提高的矛盾。随

着主机运算速度的迅速提高，仍然采用手工操作时，主机的大号宝贵时间由于等待人工操作而白白浪费了。例如，假定一个计算程序在每秒1千次的机口上运算30分钟，手工操作花费1分钟，则手工操作时间占总时间的比例约$\frac{1}{30}$。若同一计算题在每秒1百万次的机口上，则仅需0.03分钟，而手工操作仍花费1分钟，则手工操作占总时间的比例是100:1。可见在高速计算机中，手工操作已严重的影响计算机的利用率，对于大型计算机而言，手工操作实在是不能允许了。

3. 如何解决矛盾

解决矛盾的思路，首先考虑，在很多情况下，外设的输入，输出工作和CPU的运算在时间上完全可以允许重迭，即外设与中央处理机在时间上可以并行工作。我们来看一个例子：假如纸带上穿了4组数，每组2个，共8个。第一组输入第一组数，然后求它们的和并打印出来。接着输入第2组数，重复同样的过程，直到4组数全下输完并打印完全下结果。我们把第i组数的步工作记作S_{ij}，X，y是两个缓冲单元则进行步骤如下：

S_{i1}：输入两个数给X、y

S_{i2}：A=X，B=y，

S_{i3}：C=A+B

S_{i4}：输出C

事实上，当S_{13}开始时（即S_{12}已结束）S_{21}也可以同时开始。因为，此时X，y两个缓冲单元已可提供新数据使用。类似地，当S_{23}开始时，S_{31}也可以同时开始。我们把这种时间上重迭的关系用图1.3加以描述：

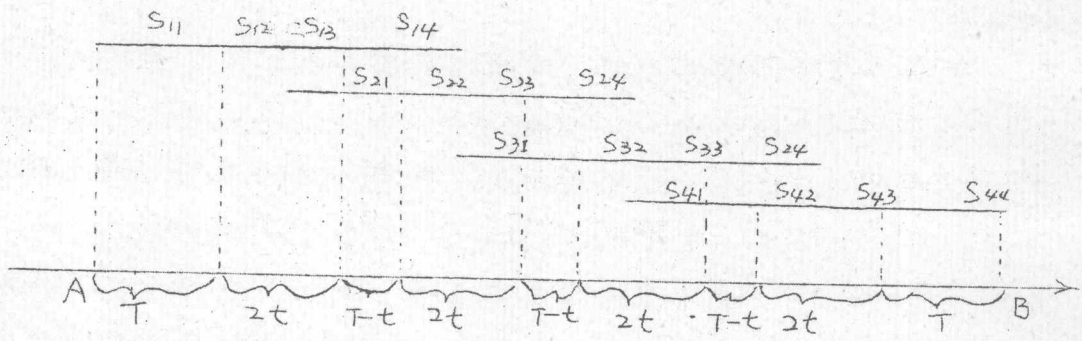

图1.3 输入输出和运算在时间上重迭的关系

假设第1步与第4步所需的时间都是T，第2步和第3步所需的时间都是t，全卩工作所需的时间就是5T+5t，CPU的效率是 $\frac{8t}{5T+5t}$。如果时间不重迭，则全卩工作所需的时间为8T+8t。CPU的使用效率只有 $\frac{8t}{8T+8t}$，可见时间重迭后，CPU的效率提高了 $\frac{3t}{5T+5t}$。

为了实现外卩设备与主机并行工作，除了要增加一些能够完成予期功能的硬件设备之外，还必需事先编制好一些程序，有这些程序参与执行，才可以完成并行工作的全过程。编这些程序的目的，不是解决某一个用户的标题，而是为了完成外卩设备与主机并行工作所需要的，是属於计算机系统本身的。这些程序便是操作系统的一卩分。

采取输入输出与运算在时间上重迭的办法可以提高CPU的效率，但是，还不能全卩解决CPU等待外卩设备的问题。也就是说，往往不可能做到完全的重迭。例如，当上一批中间计算结果还未打印完，行印机还正外在忙碌状态时，如果新的一批中间结果又要求打印，这时，CPU就必需等待，直到行印机打印完第一批结果、开始接受第二批结果时，CPU才能接着往下运算。也就是仅仅采取外卩设备与主机并行的办法，问题解决的不够彻底，仍然

存在着CPU的等待。

要进一步解决这个问题，采取多道程序的办法。其思路是不仅仅使CPU的运算与输入输出在时间上重迭，而且CPU本身的运算也能在时间上重迭。这当然不是说在某一瞬间让CPU同时做几个运算，而是说在一定的时间间隔内，几个运算可以交替进行，从这个意义上说，这几个运算是重迭的。例如，当CPU等待行印机打印第一批中间结果时，就可以让CPU转去计算另一道题目，以提高CPU的效率。为了使计算机有条不紊的完成多道程序的运行，除了增设必要的硬件设备外，也还必需编制为了完成多道程序进行的系统程序，这又是属于操作系统的一个重要组成部分。

如何考虑解决手工操作的问题呢？解决的思路是用计算机来代替手工操作。为此，必需事先交给计算机一份用专用语言书写的操作说明书。为了使计算机能够"懂得"这份操作说明书，必需给计算机系统配置对于这种专用语言的解决程序，以便计算机按着操作说明书的要求进行操作。这就是操作系统代替人工操作完成自动操作的功能。

综上所述，操作系统就是在计算机发展过程中，为了提高计算机的效率而逐步产生形成的一个程序系统。

第二节　通道与中断系统

1. 什么是通道？

为了提高CPU的效率，可以采取CPU与外设备并行的办法。但是，早期的计算机是不能做到CPU与外设备并行工作的。这是因为一个中央处理机不可能在执行运算的同时又在控制着外设备去进行数据传输。为了达到在时间上能够重迭，首先就必需增加相应的硬件设备，一般采取一个专门用于输入，输出的处理

机。这个处理机与中央处理机公用一个内存，并可与中央处理机并行工作。这样，中央处理机就可以专门从事运算，而把大量的输入输出工作交给这种处理机去完成。我们把这种处理机就叫做通道。通道这个名称由来于这种处理机提供了输入输出设备与内存好口之间流通数据的通路。现代的计算机系统一般由一台中央处理机及若干通道所组成。每个通道可与中央处理机并行工作，可控制一台或几台外下设备传输数据，这样就提供了输入输出与运算在时间上可以重迭的物质基础。

2. 通道的情况

通道的结构简单，用途较专门，速度较低，因而它的造价比通常的CPU要低得多。通道的指令系统很简单，一般只有几条，因都是与外下设备传输数据有关的指令。通道的指令一般称为通道命令。执行通道命令的过程就是输入输出的过程。一般一条通道命令给出了本次传输是什么性质的传输。例如是从内存读出还是输入到内存，以及传送数据的个数和被传输数据在内存的起始地址。

通道本身除了有为了控制按一定时间节拍工作的逻辑电路外，一般还设计有缓冲寄存口与控制寄存口。在启动外下设备进行数据传输之前，就要把相应的通道命令送到控制寄存口内。缓冲寄存口用来存放外下设备与内存传送的一个数据。

3. 通道与中央处理机的关系

通道与中央处理机可以并行工作，並不等於说它们彼此完全独立、互不相干，而是有着密切的配合关系的。这就是由中央处理机通过执行专用指令来启动通道进行工作，而当通道执行完通道命令后，便向中央处理机发出回答，由中央处理机再做处理。至於中央处理机何时启动通道以及本次的通道命令是怎样的，则是由中央处理机届来运行的称题程序中决定的，並且是通过操作

系统的参与运行才能完成的。通道命令执行后向中央处理机发出回答，是通过计算机的中断系统来实现的。关于中断系统的问题，下面再做讨论。

还要说明一点，通道与中央处理机既然公用一个内存吟口，那么在访问内存时，就不能互相冲突。一般地，是在硬件上采取措施，例如在控制口的逻辑电路中可以做到：CPU访问内存与通道访问内存互相封锁。

4. 中断系统

"中断"的总思就是打断中央处理机当前正在做的工作，让它去处理其它更为紧迫的事，等处理完后，再继续做被中断的工作。所谓中断系统就是为了完成上述功能而产生发展起来的硬设备。为了完成中央处理机与通道的并行工作，中断系统是不可缺少的。当然，整个功能的实现是通过属於操作系统的中断处理程序的执行才能完成的。

中断系统具体要完成哪些工作，具有什么功能呢？为了使中央处理机作完中断响应后能够准确无误地从原来被打断的地方接着往下运行，中断系统在某个通道发出中断请求后，至少应该做以下几件事：

①记住被打断的位置（称为断点）。

②保留必要的现场（例如某些触发口的状态）。

③记下中断请求来自哪个通道，並把控制转到操作系统里的中断处理程序的一个入口。

当中央处理机做完中断响应后，通过专门的指令将保留的断点、现场加以恢复，并把控制转到断点处继续往下执行。

必须指出，中断系统的功能远远不只是实现中央处理机与通道之间的通讯。在计算机的发展过程中，中断系统的功能逐步增

强。现代计算机系统除了来自通道的中断请求外，通常还有以下几种类型的中断：

(1) 错误中断：例如在运算过程中发生溢出后，早期的计算机就导致行机，由程序员在行机状态下查找自己的错误，查到后改正错误重新运算。这样使用计算机效率是很低的。多道系统出现后，由于多个用户同时使用一台计算机，因此不允许计算机随便行机。当某一道程序发生溢出后，中断系统便发出错误中断请求，中央处理机收到中断请求后，便暂时停止该道程序的运行，通过操作系统自动另一道程序投入运行。此外，为了便于程序员查找错误，中央处理机在停止出错程序的运行后还应该先转到一个出错处理程序去打印必要的现场信息。

(2) 故障中断：这是指计算机的某个部件发生了故障，而不是程序本身有错误。这种部件发生的故障，可能是来自某个通道，也可能来自内存或电源系统等。只要不是全局性的故障，就不应该造成整个计算机系统的瘫痪。故障中断就是避免全局瘫痪的一种措施，它通知中央处理机暂时中断当前的运算，转到故障处理程序去进行必要的处理。

(3) 时钟中断：这是根据计算机中的时钟部件，每隔一定的时间，固定发出的一个中断请求。时钟中断可以使计算机用来自动计时，这对于分时系统和实时系统是不可缺少的。

(4) 程序中断："程序中断"的含义，有不同的解释。我们

这里讲的是这样的一种中断：它的具体功能不是由硬件规定死的，而是由软件根据需要加以定义的。有时也称为软中断。这种中断又可分为两种类型。一种是在一定条件下执行某种特殊指令所产生的（例如转管指令）这种中断与前面讲过的那些中断不同，那些中断的产生一般是程序员主观上无法控制的，而这种中断却可以由程序员根据需要在自己的程序的适当位置安排这种特殊指令而产生。另一种类型是在控制台上按下某一个按钮所产生的。这两种类型的中断都可以导致中央处理机中断当前的运行，把控制转到某一段程序的入口，去完成某种特定的软件功能。

不论是哪种性质的中断，产生中断的原因称为中断源。各种中断源本身的紧迫程度是不同的。为了根据各种中断源的紧迫程度加以区别对待，有时对各种中断源分成几种优先等级，称为多级中断。中央处理机先响应优先级别高的中断，后响应优先级别低的中断。

硬件的通道和中断系统为软件的操作系统提供了基本的物质条件。在讲解操作系统之前对计算机和通道及中断系统建立一个比较清晰的概念是十分重要的。

第三节　多道程序

一、多道程序的运行方式

在第一节中讨论了为了提高中央处理机的效率，一方面採取中央处理机与通道并行工作，另一方面採取多道程序运行方式。

所谓多道程序，就是在内存贮器中同时存放几个用户程序，每一个用户程序也可称它为一个作业。当某一个作业由于等待通道的输入输出而不能继续运行时，这种情况就叫做I/O等待。通过操作系统的选择，中央处理机就利用这个I/O等待时间去运行另一道作业。当然，致使中央处理机从运行一道作业而转向去运行另一道作业的原因，决不只是I/O等待，例如，运行中程序出错发出中断等，都可以通过操作系统停止正在运行的一道作业而转向另一道。

举例说明：现在在内存中存放有三个作业A、B、C。起初，CPU运行A作业，当有I/O等待时，CPU转去运行B作业。假设运行了一段时间，B作业也进入了I/O等待，这时CPU转入C作业，运行一段时间后，A作业的I/O等待结束，此时，CPU转回来向下继续运行A作业。此期间B作业的I/O等待也结束，待CPU将A作业运行结束后，就转入继续向下运行B作业，直到结束，cpu再转去继续向下运行C作业直到结束。CPU在作业之间转来转去是通过执行操作系统的指挥程序才能完成的。

二、多道程序中CPU的效率

下面简要的讨论一下多道程序系统能够在多大程度上提高CPU的效率，进而探索一下多道程序的道数当多少比较合适。

下面用概率论来简要地讨论一下在单道与多道两种情况下

CPU的效率。为了使问题简化，假定有n道相同的作业，它们各占用一个I/O通道，并且各道平等地占用CPU的时间。

我们规定：状态Si表示有i道作业正在等待I/O。因此，状态S0表示所有的作业都没有等待I/O，状态Sn表示所有的作业都在等待I/O，在多道程序系统中，只有在状态Sn时，CPU才是空闲的

设每道作业平均运行$\frac{1}{\lambda}$个时间单位后就要发出一次I/O请求，而每个通道平均需要$\frac{1}{\mu}$个时间单位才能完成一次I/O请求。则λ就是正在运行的作业在单位时间内发出I/O请求的概率，而μ则是单位时间内正在工作的一个通道完成一次I/O等待的概率。由于作业的运行时间可以和通道的I/O时间重迭，因此，只有当作业的前一个I/O请求尚未完成而下一个I/O请求已经到来时，该作业才需要等待。参看图1.5。

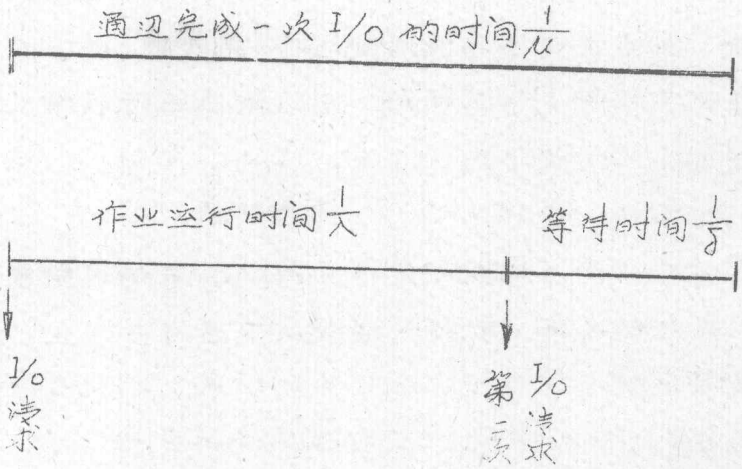

图1.5 CPU与通道并行工作时作业的等待时间。

由图1.5可知，等待时间$\frac{1}{\gamma}=\frac{1}{\mu}-\frac{1}{\lambda}=\frac{\lambda-\mu}{\lambda\mu}$。因此，$\gamma$是单位时间内一个正在等待的作业结束等待状态的概率。显然，$\gamma=\frac{\lambda\mu}{\lambda-\mu}$。