

快速响应实时操作系统中的进程调度与通信*

吴新垣

〔提要〕本文提出了“进程状态矩阵选优算法”与“不抢占处理机的优先调度策略”及相应的通信方式，来满足一类快速响应控制系统的特殊要求。事实表明，这种选优算法与调度策略对提高处理机的快速响应性能有显著效果。

〔Abstract〕 In this paper, 'Optimization algorithm of process state matrices' and 'priority scheduling and no seizure strategy' and the corresponding communicating modes are presented, so that special requirements from a class of rapid response control systems are satisfied. It has been shown that this kind of optimization algorithm and scheduling strategy have resulted in significant improvement in the processor speed response.

一 引言

在现代战斗指挥控制系统中，从雷达发现目标到武备发射弹丸，通常只有十几秒钟的时间可供利用。这段时间内，系统要进行信息检测、数据处理、指挥控制等一系列工作。作为指控系统的调度中心，首先要求计算机实时操作系统具有快速响应的性能，要在毫秒级时间内做完所需的工作。为了取得这种性能，除提高处理机本身的速度外，更重要的是减少系统本身的开销，提高资源利用率。

我们这里所说的系统开销，是指进程调度所占用的时空。一个实时中断信号，从开始响应到真正进入处理，必须经过进程调度，进程调度占用的时间越长，实时处理的效率就越低。因此，应当尽量压缩进程调度程序的数量和占用的时间，采用合理的结构来提高实时处理效率。

实时控制系统的共同特点是带有大量的终端装置，系统必须保证在规定的响应时间内对每个终端装置作出反应。操作系统的任务就是根据系统控制的要求，统一组织和安排系统工作流程，支持和协调并发执行的程序，达到提高系统的实时性、可靠性及使用方便的目的。

二 系统结构概述

为了使操作系统结构清晰可靠，便于调试和扩充，我们采用了层次结构，把系统分成内、外两层，外层是在调用内层功能的基础上构造起来的。模块之间只能按层次关系单向调用，以便防止死锁发生。

本系统结构层次关系如下：

内核 (管态)	一级	进程调度
	二级	时钟管理、修改系统表、进程建立和启动 中断管理
	三级	进程通信
外层 (目态)	系统进程（命令处理、设备管理） 用户进程	

内核是机器硬件功能的延伸，它为进程的运行创造条件，是建立系统的基础。内核的功能、结构和效率对整个系统的功能、结构和效率都有重大影响。

模块之间只许由外向内或同级调用。调用方式有两种：一是通过访管，统一由进管模块和退管模块实现调用与返回；一是直接转入和返回（如中断处理模块）。

内核模块受中断或别的进程的激发，并且通常是在禁中状态下运行的。为了提高系统的并行性与实时性，一部分内核模块可以开中运行。当在这种模块中发生中断时，为了保证系统功能的正确，仍应返回该内核模块而不能进入调度。

外层的所有模块都是作为进程运行的。它们需要经过内核的中断处理模块与进程调度模块才能投入运行。系统的主要功能是以内核为基础通过外层的扩充而最后实现。

需要指出的是，在进行系统的结构设计时，我们力图避免系统层次过多，或把尽可能多的系统程序建

* 收稿时间1981年1月28日。

立成进程。因为每进入一层，要有一层的保护区，每建立一个进程，要有一个进程控制块。这样既费时间，又费内存，使系统开销太大，是快速响应实时操作系统难以容纳的。

三 进程与进程状态

进程是程序的一种动态概念，是指运行于某个数据集上的程序。为了描述进程的动态变化，通常把一个进程分为程序、数据集与进程控制块三部分。操作系统只要通过有关程序访问和改变相应进程的进程控制块的内容，就能控制进程的状态变化，协调进程之间的相互制约关系，维持系统的工作。

本系统将进程分为系统进程和用户进程两大类。系统进程又分为命令处理进程与设备管理进程。用户进程则是指用户的实时处理程序。

进程的每个阶段称为一个状态。在本系统中，我们将进程分为建立、就绪、运行、停止、挂起、潜伏六个状态。它们之间的相互关系如图 1 所示。

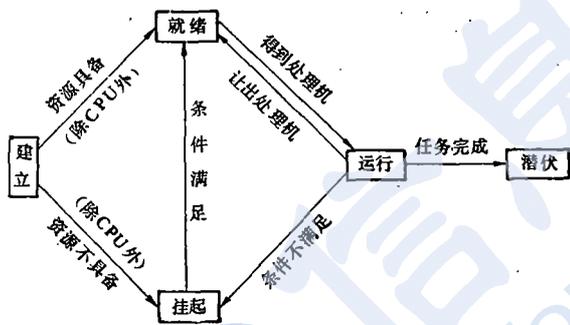


图 1 进程状态转换图

建立状态：由中断处理进程或别的进程建立了进程控制块的状态。

就绪状态：进程已经具备运行的条件，但未被分配到处理机。

运行状态：正在被处理机执行的进程。

挂起状态：由于各种原因，使进程不能继续下去而暂时停止。如果上述原因消失，该进程便被解挂，变成就绪状态。

停止状态：已被建立，但处于非运行状态的进程。

潜伏状态：进程控制块被取消后的状态。

四 进程状态矩阵选优算法

前述进程的六种状态可归结为下面三种互斥关系，即：建立/潜伏，就绪/挂起，运行/停止。正好可用布尔变量来描述这种关系。为了实现进程调度，我们把各个进程的状态及其优先级排列成下面的矩阵形式：

进程状态矩阵

进程号 状态	1	2	3	4	5	6	·	n
运行/停止								
建立/潜伏								
就绪/挂起								
优先级 1								
优先级 2								
优先级 3								

这里规定：若进程处于建立、就绪或运行状态时，该进程的相应位取值为 1；若进程处于潜伏、挂起或停止状态时，相应位取值为 0。

状态矩阵由系统初启或建立新进程时赋值。撤销某个进程时，只要将建立/潜伏位置零即可。

进程调度的任务是把处理机分配给系统进程或用户进程。由于调度程序使用频繁，而且是在禁中状态下完成的，因此力求减少它的时、空开销，以便保证快速响应的性能。

按照我们建立的进程状态矩阵选优算法，实现进程调度就十分简便。该算法是：对由各进程状态及优先级所赋值的布尔矩阵进行逻辑运算，先求出各进程全部状态变量的逻辑积，再对该逻辑积施以规格化运算，然后取已规格化的逻辑积的最高位所对应的进程投入运行。这样就完成了从当前全部就绪进程中选取优先级最高的进程的工作。

如果处理机内没有可运行的进程时，进程选优模块就重复执行自己转自己的空操作，同时开放全部中断源，以便一旦有中断产生，就立即进入中断处理模块，使系统继续工作。

进程选优模块框图如下：

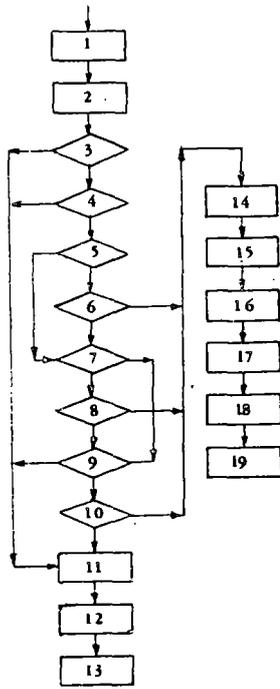


图2 进程选优模块框图

- 1 禁中
- 2 清进调度标志
- 3 建立了进程吗?
- 4 有就绪进程吗?
- 5 有一级优先进程吗?
- 6 能选上运行进程吗?
- 7 有二级优先进程吗?
- 8 能选上运行进程吗?
- 9 有三级优先进程吗?
- 10 能选上运行进程吗?
- 11 清除运行标志
- 12 开中
- 13 空转程序
- 14 选出运行进程号
- 15 填运行标志
- 16 找出新选进程的进控块首址
- 17 恢复处理机状态信息
- 18 开中
- 19 转入新进程运行

五 进程调度策略

为了节省进程在处理机上反复调度的开销，便要求处理机上进程交换的次数最少，也就是说，希望某个进程一旦获得处理机，就尽可能运行到它结束后再转入别的进程去运行。但是，为了保证实时系统的响应时间，当发生中断请求时，又不得不打断正在运行的进程。因为与处理机相连的每种设备都有它自己的工作周期，前一周期的信息若未及时处理，后一周期的信息便要取而代之。所以，减少进程交换次数与保证响应时间是互相矛盾的。

我们的目标是制订一种进程调度策略，既要使处理机上进程的交换尽量少；又能在每个进程所要求的响应时间内做完应做的工作。

在实时控制系统中，每个进程都是周期性地工作的，也就是说，在每种设备的工作周期内，与之相应的进程必须运行一次。如果按某个固定时间间隔来划

分时间片，则每种设备的工作周期 X_i 便可用时间片来衡量；同样，每个进程在处理机上运行的时间 Y_i 也可用时间片来衡量。后者与前者之比称为该进程在处理机上运行的效率 η_i 。要提高处理机的效率，就应通过进程调度来提高进程的并发性。如能安排多个进程在处理机上同时运行，则处理机的总效率 η 为：

$$\eta = \sum_{i=1}^n P_i \cdot \eta_i = \sum_{i=1}^n P_i \cdot \frac{Y_i}{X_i}$$

其中 P_i 为各进程在处理机上被调度到的概率。

为了合理安排各个进程在处理机上的运行时间，可以求出需要在处理机上运行的全部进程的工作周期的最小公倍数 Z 。于是，处理机的总效率 η 亦可表示为：

$$\eta = \sum_{i=1}^n P_i \cdot Y_i \cdot K_i / Z$$

式中 $K_i = Z / X_i$

显然，处理机的总效率不可能大于 1。

下面论述三种进程调度策略。为了说明问题方便，在图3中姑且假设有三个进程需在处理机上运行。它们的优先顺序可按其工作周期由小到大的次序来排列，工作周期越短优先级越高。例如，图中第一个进程的工作周期为 4 个时间片，它在处理机上需运行 2 个时间片；第二个进程的工作周期为 5 个时间片，它在处理机上只需运行一个时间片，而第三个进程的工作周期为 10 个时间片，它在处理机上要运行 3 个时间片。这三个进程工作周期的最小公倍数为 20 个时间片。也就是说，每隔 20 个时间片，整个进程的状态变化就重复一次。显然，仅当分给各个进程的时间片总数不超过 20 时，系统才能正常地工作。如何给每个进程分配时间片呢？可以用下面三种方法来达到目的：

第一种调度策略(A)称为“抢占处理机的优先调度策略。”它的原则是：1)按优先顺序选取已就绪的进程；2)如果没有新的就绪进程产生，则让已占有处理机的进程运行完毕(除非它因故被挂起)；然后再由选优模块选择下个运行进程；3)若有新的就绪进程产生，则停止正在运行的进程，并由选优模块选出当时优先级最高的进程投入运行；4)为了保证在每个进程的工作周期结束前做完所需的工作，应在每个进程进入就绪状态的同时建立一个调度时钟。该时钟

对它所监视的进程起定时作用，定时中断一出现，便立即抢占处理机，使它监视的进程投入运行。调度时钟的定时参数 τ_i 可由下式确定：

$$\tau_i = T_i - t_i^0 + t_i^1,$$

式中 T_i ——第 i 个进程的工作周期，

t_i^0 ——第 i 个进程所需的运行时间，

t_i^1 ——第 i 个进程已经运行的时间。

第二种调度策略(B)称为“不抢占处理机的优先调度策略”。它与前一种策略的不同之处在于：不论有无新的就绪进程产生，都让已占有处理机的进程继续运行下去，直到该进程运行完毕或因故挂起时，才转入选优模块，选择下个运行进程。唯一的例外是：正在运行着的进程可以被调度时钟所唤醒的进程取代，这是保证及时响应所必需的。

第三种调度策略(C)是“按时间片轮转的优先调度策略”。其原则是：1)按已就绪的进程的优先级每次运行一个时间片，按高——低——高的顺序循环下去；2)若有新的就绪进程产生，则由选优模块重新选

择优先级高的进程投入运行；3)关于调度时钟的规定与第一种策略的第4项完全相同。

人们通常认为：抢占处理机的优先调度策略似乎比不抢占处理机的优先调度策略具有更好的响应性能。但从图3可以看出：采用策略(A)时，进程交换14次；采用策略(B)时，进程只交换10次；而采用策略(C)时，进程竟交换17次之多！因此我们得出结论：

1)抢占处理机的做法使进程交换的次数明显增加，其响应特性并未因此得到改善。

2)伴随每个进程的定时调度时钟是保证实时响应的关键措施，如果没有它，当处理机比较繁忙时，不论采用何种调度策略，优先级低的进程都很难得到及时响应。

3)按时间片轮转的优先调度策略对优先级低的进程可能会有好处，但进程交换的次数太多，系统开销太大，会影响实时处理的效率。

根据上述分析比较，我们认为不抢占处理机的优先调度策略(B)最适合快速响应的要求。

图3 进程调度策略之比较

时间片 策略 进程 状态	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
策略 A	1	运行	停止	运行																
	2	就绪	运行	停止	就绪															
	3	就绪	就绪	运行	停止	运行	停止	运行	停止	运行	停止	就绪	就绪	运行	停止	就绪	就绪	运行	停止	就绪
策略 B	1	运行	停止	就绪	运行															
	2	就绪	运行	停止	就绪															
	3	就绪	就绪	运行	停止															
策略 C	1	运行	停止	运行	运行	停止	运行	运行	停止	运行										
	2	就绪	运行	停止	就绪															
	3	就绪	就绪	运行	停止															

六 中断管理

中断管理的基本功能是把中断信号和输入信息转化成消息，传递给有关进程或设备去执行。

本系统的中断分三级：一级中断为电源故障，二级中断为处理机故障，三级中断为外部设备请求，包括各种专用与通用外设。每级中断各有自己的处理入口，并各有一个屏蔽触发器来控制它是否允许中断，

触发器的状态可由操作系统指定。

中断处理时，首先保存被中断进程的现场（指令计数器、状态寄存器、中断屏蔽触发器的内容等）。接着分析中断源，分析时从左到右扫描，按优先顺序依次响应，先把信息送到相应进程的通信区，再置信号灯V操作和修改该进程的状态，如果中断处理的工作量过大，则需对这个中断请求建立一个进程及调度时钟，然后恢复被中断进程的现场，退出中断并返回

被中断的进程。(若按前述进程调度策略(A)与(C)调度时。退出中断后便进入选优模块)。

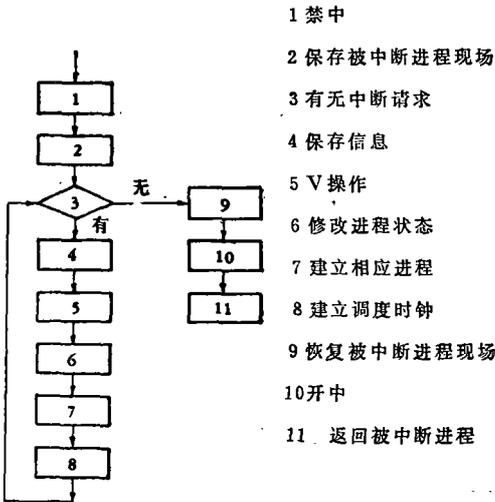


图4 中断处理模块

为了节省中断响应及保存和恢复现场所占的时间,每进行完一种中断请求的处理之后,应回到中断状态寄存器继续查找,看是否还有请求中断的信息存在?如有,则继续进行图4中4~8款的处理;如无,才可返回被中断的进程。

各种调用命令产生的程序中断与设备中断的处理相似,进入中断后根据不同的调用命令转入不同的处理模块,完成任务后同样返回被中断的进程。

中断处理是其它程序的驱动程序。一个中断信号,通常需要一个对应的进程来为它服务。对于简单的中断处理(例如仅需传送少量信息),为了减少系统的开销,则无需另设进程,就在中断期间完成。

中断管理包括中断引导模块和若干处理模块,中断模块之间的调用是直接转入和返回的,向内调用则通过访管。

七 进程通信

一个进程可能有两种因素改变它的状态:一是外来事件(中断引起);一是运行进程本身发出的广义指令(改变自己或改变其它进程的状态)。因此,进程通信应包括中断模块与进程之间,进程与进程之间这两种通信方式。前者是单向的,即中断模块可发消息给进程,而进程不可向中断模块发消息。进程与进程之间则使用高级通信原语通信。

本系统采用公共通信区来解决进程之间的通信问

题。公共通信区的格式与进程控制块完全一样,由控制信息与通信区两部分组成。它被所有的进程共享,但只有获得处理机的进程才有权使用它。

当一个进程获得处理机后,便将它的进程控制块的内容复制到公共通信区上;当进程被挂起时,又把公共通信区的内容重新送回该进程的进控块上。公共通信区的内容根据事先约定的格式填写。

例如,正在运行的进程A要求进程B输出一批信息。这时,进程A便在公共通信区填写:进程B的名字,进程B撤消时解挂进程A的标志,进程A的名字,输出信息首址,输出信息长度等内容。然后转入创建进程模块。

创建进程模块从公共通信区取出进程B的名字,建立进程B的进控块(填写B的控制信息和把公共通信区的内容复制到自己的通信区上),并使它处于就绪状态。然后创建进程模块按公共通信区有无返回标志而返回到进程A(继续运行或挂起等待进程B发回管),或者转入进程选优模块。

当进程B被选优模块选中后,就从公共通信区取出输出信息的首址与长度,输出进程A所需的全部信息,撤销进程模块再按公共通信区中解挂进程A的标志将进程A解挂,通信至此结束。

中断模块与进程之间的通信采用P、V操作的方式来实现。例如,当用户进程A需要申请打印机时,由于本系统只有一台打印机,故信号量S的初值为1。做一次P操作后 $P(S)=0$,表示打印机空闲,可供进程A使用,若进程A正在使用打印机时,进程B又申请打印机,这时 $P(S)=-1$,表示打印机无空,于是将进程B挂起,并在打印机等待站中挂号,等待站只能接纳一定数量的,申请同一设备的进程,超出此数则发警报。

当打印机空闲,并且选优模块选上了打印进程时,则在打印机等待站中清除被选中的进程号,并进行整队,然后由打印进程按公共通信区所指定的起始地址与信息长度打出全部信息。在打印机关机中断模块内应做一次释放打印机的V操作,然后转入撤销模块。做V操作后, $V(S)=1$ 表示打印机已经空闲; $V(S)\leq 0$ 表示还有别的进程需用打印机。这时,按照我们的状态矩阵选优算法,应该解挂所有的等待者,让选优模块来决定这些申请者中谁该获得打印机,获得打印机的进程与打印进程则按前面规定的公共通信区的方式进行通信。

八 结 语

为了加强实时操作系统的响应性能,我们致力于提高系统内核的工作效率,减少系统的开销,为此,我们采取了下述措施:

1. 简化系统的层次结构。整个操作系统只分内外两层,节省了多层结构所需的保护区与运行时间。
2. 采用进程状态矩阵选优算法来分配处理机,精简了许多队列,节省了排队与整队的开销。
3. 采用不抢占处理机的优先调度策略来管理进程,在保证快速响应的前提下,使进程在处理机上交

换的次数最少,节省了进程调度的大量开销,

4. 为了实现上述调度策略,本系统规定所有的中断处理模块一律返回被中断的进程,简化了系统的转换关系。

5. 中断模块与系统通信时,做V操作后便解挂全部等待外设之进程,让选优模块给它们分配处理机,保证了快速响应的性能。

这些措施对我们设计中、小型计算机的实时操作系统可能会有所补益。

作者在工作中得到翟贤同志的启发和帮助,特此致谢。

(上接65页)

信号,经过适当处理就有可能形成自动测试系统。根据这一设想构成了图3所示的误差值线性测量法方框图。

码源产生顺序的数字信号,同时输送给被测和标准数模转换器。为便于起始零值箝位,各位数字信号均为“0”的时间应稍长。码源还产生箝位脉冲,取样脉冲和控制脉冲,分别供给相应部分。

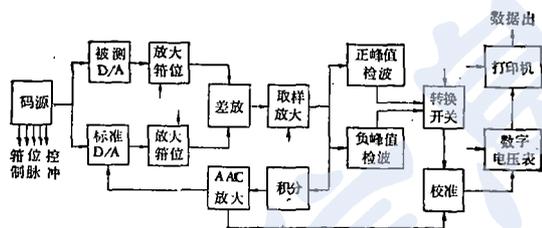


图3 误差值线性测量法方框图

两路数模转换器的模拟输出,经箝位后将起始零值箝位在相同的电位上,并在差分放大器中进行比较输出误差电压。

为了使测量准确,应对误差电压进行取样,取样点应选择误差电压稳定后的时刻。取样后的误差电压,经适当放大后送至正负峰值检波器。检波后的直流电压即反映了最大的正或负误差电压。

平均误差最小是数模转换器运用的最佳状态。为了实现自动调整到这种状态,将取样放大后的误差信号进行积分而得到一个控制电压,经过一个增益较高

的自动幅度控制(AAC)放大后去控制标准数模转换器,使其模拟输出幅度相应变化,从而使平均误差电压减小。当这个环节的增益足够高时,平均误差将很小,这时正或负峰值检波器的直流输出电压就代表了最佳状态下的最大正或负误差电压。

由于不同的被测数模转换器的模拟输出幅度可能不同,在自动控制平均误差为最小的同时也使误差电压幅度相应变化。为了能直接指示相对线性误差,需将峰值检波器的直流输出电压再经过一个校准器加以校准,并由控制电压控制,使之对于具有不同模拟输出幅度的被测数模转换器,只要相对线性误差相同,校准器就具有同样的输出电压值。

测试数据的指示可有两种,一种用表头指示,并直接按相对线性误差刻度,另一种用数字电压表并配合打印机由码源产生的控制脉冲控制,直接打印出测试结果。

以上两种测量数模转换器线性的方法,各有特点。阶梯值线性测量法不需要一个标准数模转换器作为参考就能准确地进行线性测量,组成也比较简单,从图2可以看出有较好的测量效果。误差值线性测量法则有可能快速准确地自动进行线性测量并直接打印出结果,但组成比较复杂,需要一个标准数模转换器作为参考。