

1.互斥信号量:

互斥互斥,意思就是我用你就不能用,你用了我就不能用。永远都只有一个人独占这个东西~!举个例子:比如说打印机。

我任务 1 现在让他打印《静夜思》,那么在我还没打印完之前,别的任务就不能命令打印机去打印别的东西。否则如果任务 2 让他打印《春晓》,那最后打印出来的会是什么~???反正肯定不是任务 1 想要的,肯定也不是任务 2 想要的。

上面讲的比较通俗。打印机就是共享资源,谁都可以访问他~!但是同一时间,肯定要保证只有 1 个任务再操作打印机。那样才能得到大家想要的结果。也就是要独占共享资源的访问权~!

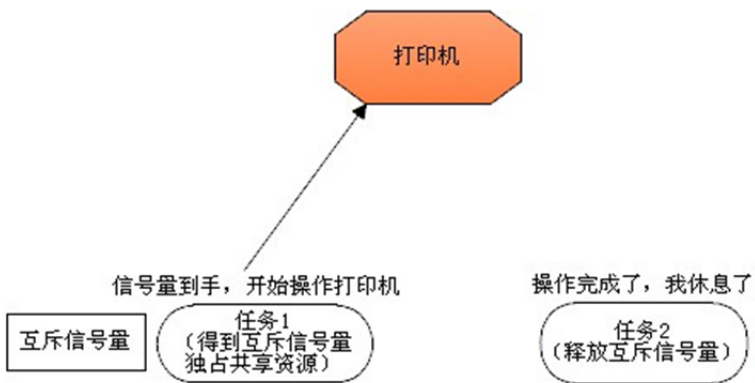
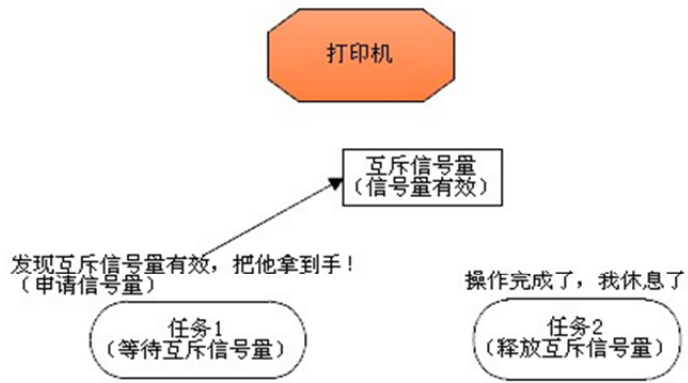
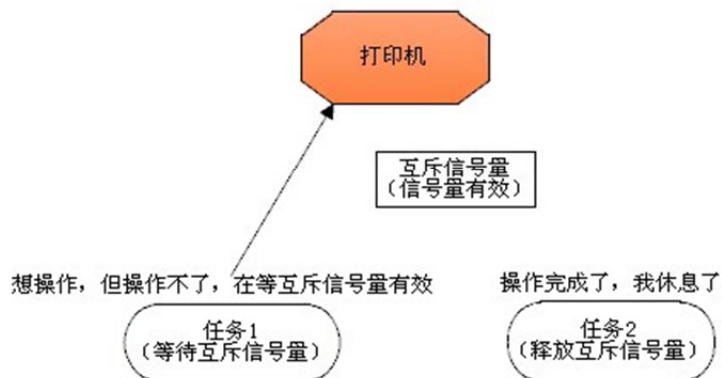
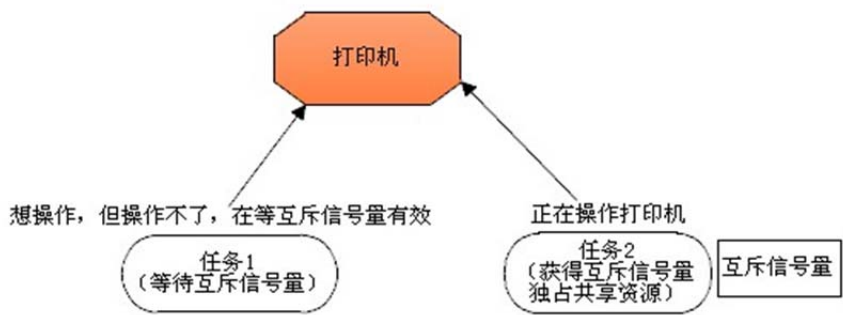
ucos2 中通过互斥信号量来解决这个问题。简单说就是任务 1 开始访问打印机的时候,先去查询这个互斥信号量是否有效,有效,说明没人在访问打印机,这时任务 1 就把这个互斥信号量置无效,然后开始操作打印机。这样,每个任务再操作打印机前都要去查询这个互斥信号量时候有效。无效就等,等到有效才可以访问,或者等到不耐烦了(术语叫等待超时)就不等了~!任务一直到用完了打印机后才把信号量置有效,这时其他任务才有可能去访问,操作打印机。

这里又有一个问题:再任务 1 操作打印机器件,可能有多个任务申请打印机的所有权。那么再任务 1 结束后,我应该给谁用呢~~??也许我们马上就反应过来了~废话~!!当然是排队了~~谁先到的谁用啊~~~。没错,这是一种机制,谁最先等待共享资源,就给谁用。但是~!再 uclos 里面 2.52 版本还不支持这种方式。他用的另外一种方法!如果你和你 BOSS 都再等着用打印机,你先到的,这个时候任务 1 结束了对打印机的操作。你说你敢先用么???

(除非你第二天不想干了~~)你肯定先让老板先用,这就是 uclos 的实现方式,基于优先级,任务 1 结束对打印机的操作后,uclos 再等待队列中看那个等待任务优先级最高,就先给他用~!即使他是最晚才等待的~!!(这就是 BOSS 的威力~!)

关于事件等待列表,有兴趣的去看看事件控制块 ECB 的内容,不在本文讨论。当然,uclos 中的互斥信号量还有许多要素,比如说他的继承优先级之类的。本文旨在说明它是干嘛用的,至于其他请参考相关书籍。

下面的图解释了互斥信号量的基本用法:(简单的两个任务,没有包含多任务等待的情况)



2.信号量:

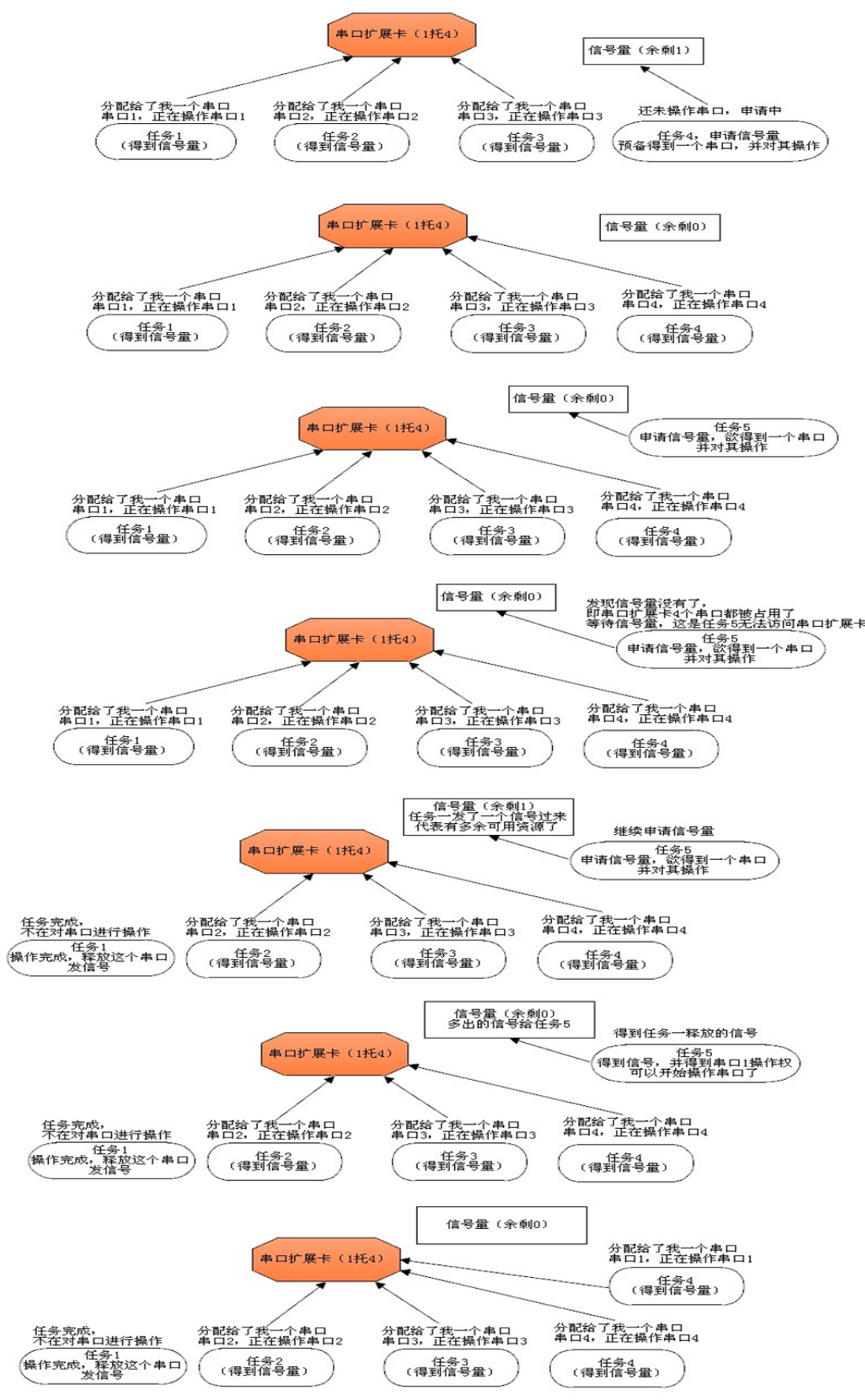
至于信号量，和互斥信号量是用区别的，简单来说（个人理解，欢迎纠正）就是互斥信号量再同一时刻，任务得到互斥信号量后是独占共享资源的，在他没有释放信号量之前，任何其他任务都是不能访问共享资源的。而信号量的不同在于。信号量可以设定一个值，允许最多又几个任务同时去访问共享资源。比如我给他设定一个 5，那么对多就有 5 个任务能同时访问共享资源。每个任务获得信号量的时候就把信号量计数器减去 1，这样，再第五个任务获取后，计数器是 0.当第六个任务要去访问的时候申请信号量就只能等待了，等到之前的任务发一个信号出来，这样第六个任务才能去访问共享资源。

互斥信号量可以看成特殊情况下的信号量，他的计数器就是 0 或者 1,只在这两个之间徘徊。举个例子（不一定恰当，欢迎纠正）:

现在有很多串口扩展卡，一张卡能扩展出好几个串口，比如说 4 个，这个扩展卡就是一个共享资源。现在定义一个信号量 `semcom`，初始给他 4，那么可以有 4 个任务去访问这个资源，他每次就给这 4 个任务分配不同的串口。每个任务要访问这个扩展卡就要去测试 `semcom` 看看他时候有信号。这样，前 4 个任务申请信号后，信号量计数器就等于 0 了，这样，在第五个任务要去访问扩展卡的时候，他也去测试这个 `semcom`，发现信号量无效，他只能等了~！等到之前的任务释放一个串口为止，如果不用信号量，那么任务五可能就会去访问扩展卡上的串口 1，而串口 1 之前已经分配给了任务 1 了，~造成什么后果就自己想想吧~~~~如果用互斥信号量，那么无疑浪费了资源，~~~那你就买个扩展 1 个串口的卡就行了~~你买个扩展 4 个的然后你用互斥信号量~~~不是摆明再说你是富二代么~~~

等待信号的任务在有信号以后也是按照等待列表中优先级最高的任务先得到信号处理。有关信号量的具体数据结构参考事件控制块 ECB 的内容，具体操作参考信号量函数等。在此不做介绍

下面这个图说明了以上的例子：（



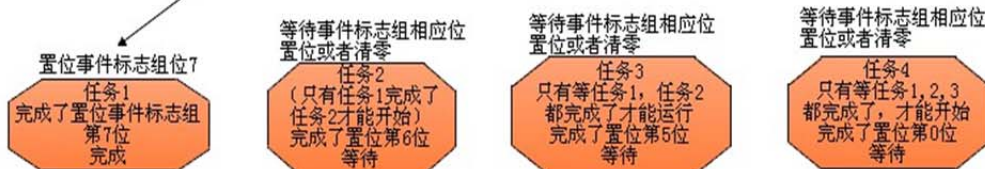
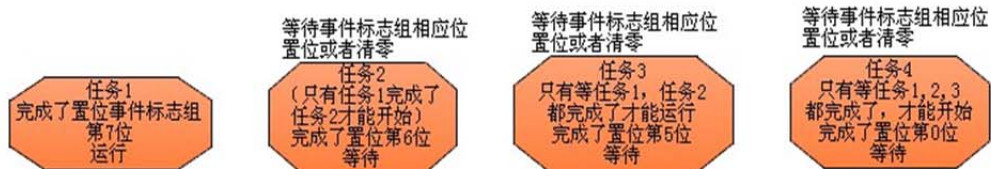
3.事件标志组:

在理解信号量和互斥信号量的时候都可以类比,因为他们在 ucos2 里面都通过相同的时间控制块即 ECB 这个数据结构来实现,理解了一个就很好能看懂另外一个,设置更后面的邮箱和消息队列,也能和信号量之类的类比来学习,他们都有通过 ECB 来维护。但是事件标志组比较特别,他是 ucos2 所有这些内核事件里面没有用到 ECB 的。他有自己的做法。不太合群。什么是事件标志组?

上面说的信号量,互斥信号量。都是用来同步任务对共享资源的访问,防止冲突而设立的。事件标志组----他是用来同步几个任务,协调几个任务工作而设立的。打个比方你现在要打个电话,打电话这个任务要执行,你必须有手机吧!那你要先执行买手机这个任务,你手机有了,没话费~你也大不了吧~,也就是说打电话这个任务要等买手机这个任务和充话费这个任务都完成了以后你才能去开始打电话这个任务。事件标志组就是用来标志买手机或者充话费这两个任务完成了没有。完成了的话他们会相应的置位事件标志组里面的某些标志位。那么打电话这个任务。发现事件标志组里面买手机对应的位和充话费对应的位都置位了以后就明白,现在可以开始打电话了~!实际中比如你想要读数据,那你肯定要等数据采集更新好了以后你去读才有意义吧~所以数据采集和读取数据这两个任务也可以用事件标志组来实现。当然,事件标志组不一定只用于两个任务之间,通过对头文件的修改,可以让事件标志组达到 32 位,你可以用事件标志组来协调多个任务的合理运行。达到你预期想达到的目的!事件标志组就是专门干这个活的。

事件标志组的结构比其他的会复杂一点。没一个事件标志组都维护这自己的一个等待队列的双向链表。每个事件标志组的节点里面都有一个指针和相应的任务控制块 TCB 一一对应。至于事件标志组的具体实现方法,可以自己去看看源代码。只要懂得一些浅显的双向链表的知识,大概理解他的运作机制不会很难。

下面这幅图大概反应出了事件标志组是如何协调任务工作的:



.....

任务3, 任务4也如上的步骤进行, 通过事件标志组同步多个任务, 达到多个任务的协调, 完成系统所需功能

终于要完了~~总的来说，到目前为止，都还能大概理解 uclos 的每个模块的实现机理。但是是不是真的懂了自己不敢说。毕竟要等能真的自己动手一直，再 uclos2 系统上开发出应用那样才能真的算懂了。也知道到那个时候，才有可能能对其有更深入的理解。个人认为初看 uclos2 的时候只要能有一些基本的数据结构基础如线性表，堆栈，队列，单链表，双链表之类的就差不多。在看的过程中慢慢体会作者的学习和学习别人的编程思路才是最重要的。