

掌上设备和PC机之间的主要差别是,前者没有若干GB的、不断变化的硬盘。在掌上设备上最主要的两个操作系统是Symbian OS和Plam OS。

1.4.6 嵌入式操作系统

嵌入式系统在用来控制设备的计算机中运行,这种设备不是一般意义上的计算机,并且不允许用户安装软件。典型的例子有微波炉、电视机、汽车、DVD刻录机、移动电话以及MP3播放器一类的设备。区别嵌入式系统与掌上设备的主要特征是,不可信的软件肯定不能在嵌入式系统上运行。用户不能给自己的微波炉下载新的应用程序——所有的软件都保存在ROM中。这意味着在应用程序之间不存在保护,这样系统就获得了某种简化。在这个领域中,主要的嵌入式操作系统有QNX和VxWorks等。

1.4.7 传感器节点操作系统

有许多用途需要配置微小传感器节点网络。这些节点是一种可以彼此通信并且使用无线通信基站的微型计算机。这类传感器网络可以用于建筑物周边保护、国土边界保卫、森林火灾探测、气象预测用的温度和降水测量、战场上敌方运动的信息收集等。

传感器是一种内建有无线电的电池驱动的小型计算机。它们能源有限,必须长时间工作在无人的户外环境中,通常是恶劣的环境条件下。其网络必须足够健壮,以允许个别节点失效。随着电池开始耗尽,这种失效节点会不断增加。

每个传感器节点是一个配有CPU、RAM、ROM以及一个或多个环境传感器的实实在在的计算机。节点上运行一个小型但是真实的操作系统,通常这个操作系统是事件驱动的,可以响应外部事件,或者基于内部时钟进行周期性的测量。该操作系统必须小且简单,因为这些节点的RAM很小,而且电池寿命是一个重要问题。另外,和嵌入式系统一样,所有的程序是预先装载的,用户不会突然启动从Internet上下载的程序,这样就使得设计大为简化。TinyOS是一个用于传感器节点的知名操作系统。

1.4.8 实时操作系统

另一类操作系统是实时操作系统。这些系统的特征是将时间作为关键参数。例如,在工业过程控制系统中,工厂中的实时计算机必须收集生产过程的数据并用有关数据控制机器。通常,系统还必须满足严格的最终时限。例如,汽车在装配线上移动时,必须在限定的时间内进行规定的操作。如果焊接机器人焊接得太早或太迟,都会毁坏汽车。如果某个动作必须绝对地在规定的时刻(或规定的时间范围)发生,这就是硬实时系统。可以在工业过程控制、民用航空、军事以及类似应用中看到很多这样的系统。这些系统必须提供绝对保证,让某个特定的动作在给定的时间内完成。

另一类实时系统是软实时系统,在这种系统中,偶尔违反最终时限是不希望的,但可以接受,并且不会引起任何永久性的损害。数字音频或多媒体系统就是这类系统。数字电话也是软实时系统。

由于在(硬)实时系统中满足严格的时限是关键,所以操作系统就是一个简单的与应用程序链接的库,各个部分必须紧密耦合并且彼此之间没有保护。这种类型的实时系统的例子有e-Cos。

掌上、嵌入式以及实时系统的分类之间有不少是彼此重叠的。几乎所有这些系统至少存在某种软实时情景。嵌入式和实时系统只运行系统设计师安装的软件用户不能添加自己的软件,这样就使得保护工作很容易。掌上和嵌入式系统是为普通消费者使用的,而实时系统则更多用于工业领域。无论怎样,这些系统确实存在一些共同点。

1.4.9 智能卡操作系统

最小的操作系统运行在智能卡上。智能卡是一种包含有一块CPU芯片的信用卡。它有非常严格的运行能耗和存储空间限制。其中,有些智能卡只具有单项功能,诸如电子支付,但是其他的智能卡则可在同一块卡中拥有多项功能。它们是专用的操作系统。

有些智能卡是面向Java的。其含义是在智能卡的ROM中有一个Java虚拟机(Java Virtual Machine, JVM)解释器。Java小程序被下载到卡中并由JVM解释器解释。有些卡可以同时处理多个Java小程序,这就是多道程序,并且需要对它们进行调度。在两个或多个小程序同时运行时,资源管理和保护就成为突出的问题。这些问题必须由卡上的操作系统(通常是非常原始的)处理。