

实时操作系统应用开发

指南

唐寅 编著 博嘉科技 审



中国电力出版社

www.infopower.com.cn

实时操作系统应用开发指南

唐寅 编著 博嘉科技 审

中国电力出版社

内 容 提 要

实时操作系统(RTOS)是计算机中的一种重要系统软件。与通用平台操作系统不同,RTOS 往往嵌入到机器或设备内部运行,一般用户无法看到它的运行界面,因而又被称为嵌入式操作系统。本书是根据作者长期从事基于 RTOS 及其有关产品的研究开发经验精心编写而成,其内容包括:RTOS 的基本概念、RTOS 的任务管理、定时器管理、内存管理、I/O 处理、中断处理、队列管理、线程管理、RTOS 应用开发技术和常见 RTOS 及其开发工具等。

本书是为对 RTOS 应用开发感兴趣的工程技术人员编写的,也可作为有关专业高年级本科生或研究生学习 RTOS 应用开发的教材或参考资料。

图书在版编目(CIP)数据

实时操作系统应用开发指南/唐寅编著. —北京:中国电力出版社, 2002. 6
ISBN 7-5083-1069-1

I. 实... II. 唐... III. ①实时操作系统-应用-指南 ②实时操作系统-软件开发-指南 IV. TP316. 2-62

中国版本图书馆 CIP 数据核字(2002)第 032894 号

中国电力出版社出版、发行

(北京三里河路 6 号 100044 <http://www.infopower.com.cn>)

汇鑫印务有限公司印刷

各地新华书店经售

*

2002 年 7 月第一版 2002 年 7 月北京第一次印刷

787 毫米×1092 毫米 16 开本 20 印张 479 千字

定价 32.00 元

版 权 所 有 翻 印 必 究

(本书如有印装质量问题,我社发行部负责退换)

前 言

随着信息技术的发展，信息时代的来临，各类基于实时操作系统（RTOS）的产品应运而生，PDA 个人数字助理、HPC 掌上电脑、机顶盒和手机上网等新名词仿佛突然间从天而降。今天，嵌入式 RTOS 已经广泛应用于工业、交通、能源、科研和国防等各个领域，嵌入式系统带来的工业年产值已超过了 1 万亿美元，它将成为 IT 产业争夺的重点之一。这就迫使我们不得不将目光投入到操作系统中最轻巧、最具活性的 RTOS 中去。

RTOS 发展到今天，功能越来越强大，除了实时性、稳定性和可靠性越来越高外，还具有网络功能、能支持 TCP/IP、GUI 图形用户界面以及多用户多任务等。在众多的 RTOS 系统中，VxWorks 是当前 RTOS 市场上最具实力的代表，除了其强大的系统功能外，与之配套的开发环境——Tornado 为用户提供了良好的调试跟踪工具以及软件管理工具，在 RTOS 业界具有良好的口碑；由 Accelerated Technology 公司开发的 Nucleus PLUS，提供源代码、价位较低，其配套开发工具 UDB 界面友好，功能齐全，是进行 RTOS 产品开发的一个不错的选择；Windows CE 是 Microsoft 公司为开发各类功能强大的信息设备而开发出来的一个嵌入式实时操作系统，它的核心全部是由 C 语言开发的，操作系统本身还包含了许多由第三方厂家用 C 语言和汇编语言开发的驱动程序，它的内核提供了内存管理、抢先多任务和中断处理等功能；相比之下嵌入式 Linux 有着得天独厚的优势：源码开放、内核小、功能强大、运行稳定、系统健壮和效率高，不仅支持 x86 CPU，还可以支持其他数十种 CPU 芯片，有大量的且不断增加的开发工具，遵循国际标准，可以方便地获得众多第三方软硬件厂商的支持等，是今后 RTOS 的一个重要发展方向。

本书特点

长期以来，讲述 RTOS 系统知识及其应用开发的书籍一直不多，而已有的有关书籍又都是针对某一 RTOS 系统进行专门的介绍，或者是采用高深的理论方法来论述 RTOS 系统，因此尚缺少能够覆盖 RTOS 共性的基础知识的书籍，本书就是基于这种背景而编写的。

本书系统阐述了嵌入式实时操作系统的基本概念、历史与发展、研究现状，RTOS 的系统原理，当前常见的 RTOS 功能及其开发工具，以及 RTOS 应用开发范例。是一本较为全面地讲述 RTOS 理论与应用的专业书籍，既可作为工程指导资料，也可作为大专院校学生学习 RTOS 的辅导教程。本书兼顾了理论性与工程实用性，通过本书的学习能对现阶段 RTOS 的基本技术和应用开发方向有一个较为全面的认识。

适用对象

本书是为有意从事 RTOS 系统开发的工程技术人员而编写的。在阅读本书之前，您最好已具备有关计算机和操作系统的基本知识，以及一定的软件开发经验，比如用 C 语言开发软件，有网络编程知识等。

内容安排

全书分为5篇，共26章：

第1篇包括第1、2章，分别介绍了RTOS的基本概念、历史与发展、体系结构、技术标准、技术现状等基础知识。

第2篇包括第3~14章，分别介绍了RTOS内核实现的关键技术，包括系统初始化、任务管理、时钟管理、内存管理、队列管理、线程管理、I/O处理、中断处理、系统检测和通用操作系统平台的联系与区别等核心内容。

第3篇包括第15、16章，分别介绍了开发RTOS软件常用的开发方法，以及在实际工作中怎样选择一款适合自己需求的RTOS。

第4篇包括第17~23章，分别介绍了在当前的IT市场上常见的RTOS及其开发工具。

第5篇包括第24~26章，分别介绍了如何利用RTOS进行嵌入式Internet设备开发，包括当前嵌入式Internet设备的基本技术情况、嵌入式TCP/IP，嵌入式Telnet设备和防火墙设备的开发技术、范例和部分源代码。

在最后的附录中，列出了RTOS系统常用词汇表、缩略语表，供读者在阅读相关文档和进行相关技术开发时查阅。

致 谢

本书的编撰过程是一个艰辛的劳动过程，其间幸而得到了周围热心朋友的大力支持，此书最终得以奉献给广大读者，谨在此对他们表示深深的感谢。

本书由唐寅编著，另外参与编写工作的还有王松、杨睿、陈文、王素中、巩伟强、刘青松、田茂敏、巫文斌、付欣、黄巍、杨思忠、郭乐深、李鹏、黄兰、田达、陈宇、刘郁林、李修华、吴向阳、马卫东、潘俊峰、樊江、张键、周朝、许勇、李欣、戴刚、魏风、江涛、刘杰等，在此对他们表示感谢。

由于时间仓促，再加上实时操作系统的开发和设计是一个飞速发展着的全新领域，本书中的不足在所难免。虽然我们力图做到精益求精，但难免有所疏忽，敬请读者批评指正。您中肯的意见将是对我们最大的信任和帮助。

若读者、网友发现有网站未经授权，而转载本书内容或提供各种形式的下载服务，请予举报。

延伸服务：如果读者愿意参加“实时操作系统应用开发指南”的学习培训，或是在学习过程中发现问题，或有更好的建议，欢迎致电。联系电话：(028) 5404228 或 5460593；E-mail:bojiakeji@163.net。通讯地址：成都四川大学（西区）建筑学院成都博嘉科技资讯有限公司；邮编：610065。

在本书编写过程中，我们得到了很多专家的热情帮助和指导，并提供了宝贵的学习资料，在此深表谢意。但是，我们惟恐还有未联系上的人员，请与我们联系。谢谢！

作 者

目 录

前 言

第 1 篇 实时操作系统基础知识

第 1 章 实时系统概述	3
1.1 操作系统概述	3
1.2 实时系统基本概念	5
1.3 实时操作系统发展史	11
1.4 实时操作系统现状	12
1.5 实时系统典型应用	14
1.6 实时系统研究发展趋势	15
第 2 章 实时操作系统基本原理和技术	22
2.1 实时操作系统基本特征	22
2.2 嵌入式实时系统分类	24
2.3 实时操作系统体系结构	27
2.4 实时操作系统关键技术指标	31
2.5 实时操作系统基本术语	32

第 2 篇 实时操作系统核心技术

第 3 章 系统初始化	37
3.1 实时操作系统核心技术概述	37
3.2 系统初始化概述	38
3.3 基本功能	38
3.4 实例	39
第 4 章 任务管理	42
4.1 基本概念	42
4.2 常见术语	43
4.3 任务通信	45
4.4 任务同步	47

第 5 章 定时器管理	51
5.1 概述	51
5.2 常量和数据结构	51
5.3 定时器基本服务	55
第 6 章 内存管理	57
6.1 静态内存池	57
6.2 动态内存池	61
第 7 章 I/O 处理	66
7.1 I/O 概述	66
7.2 I/O 驱动管理数据结构与服务	67
7.3 I/O 驱动管理主要例程	71
第 8 章 事件组管理	73
8.1 事件组管理模块概述	73
8.2 数据结构定义或参量	73
8.3 事件组管理关键例程	75
第 9 章 邮箱	77
9.1 邮箱概述	77
9.2 邮箱管理模块数据结构或参量	78
9.3 邮箱管理模块关键例程	80
第 10 章 管道	82
10.1 管道概述	82
10.2 管道管理数据结构和参量	82
10.3 管道管理模块关键例程	84
第 11 章 队列	87
11.1 队列概述	87
11.2 队列管理数据结构和参量	87
11.3 队列管理关键例程	89
第 12 章 信号	92
12.1 信号概述	92
12.2 信号管理数据结构的定义和参量	93
12.3 信号管理关键例程	94
第 13 章 线程	96

13.1 线程概述	96
13.2 线程控制数据结构和参量	97
13.3 线程控制关键例程	104
第 14 章 其他技术	109
14.1 中断处理	109
14.2 系统检测	110
14.3 历史管理	111
14.4 公共服务	112
14.5 错误处理	113

第 3 篇 实时操作系统应用开发技术

第 15 章 实时操作系统应用开发方法	117
15.1 应用开发概貌	117
15.2 开发步骤	118
15.3 生命周期法	122
15.4 增量开发法	123
15.5 变换型开发法	124
15.6 交叉开发模式	124
第 16 章 怎样选择实时操作系统	126
16.1 概述	126
16.2 选择实时操作系统的依据	127
16.3 综述	131

第 4 篇 常见实时操作系统与开发工具

第 17 章 Vxworks 与 Tornado	135
17.1 常见的实时操作系统开发工具	135
17.2 VxWorks	137
17.3 Tornado	140
第 18 章 Nucleus Plus/UDB	145
18.1 Nucleus Plus	145
18.2 Nucleus UDB	146
第 19 章 pSOSystem	153

19.1	pSOSystem 概述	153
19.2	pSOSystem 系统结构	153
19.3	pSOSystem 的集成开发环境	155
19.4	pSOS 系统的输入/输出	156
19.5	文件和磁盘文件	157
第 20 章 QNX		159
20.1	QNX 简介	159
20.2	体系结构	160
20.3	资源管理和路径名空间管理	161
20.4	Fsys (文件系统管理)	162
20.5	Dev (设备管理)	163
20.6	可扩展性	164
20.7	网络服务——FLEET™ 网络技术	164
20.8	可维护性	166
20.9	小结	167
第 21 章 Pencil		168
21.1	Pencil 概述	168
21.2	系统特征	169
21.3	开发和调试	174
21.4	Pencil 的实时性能	176
21.5	中断处理	176
21.6	Pencil 文件清单	177
21.7	小结	178
第 22 章 Windows CE		179
22.1	Windows CE 概述	179
22.2	Windows CE 的特征	181
22.3	Windows CE 编程知识	182
22.4	Windows CE 的结构化存储	183
22.5	Windows CE 展望	184
第 23 章 嵌入式 (embedded) Linux		186
23.1	嵌入式 Linux 概述	186
23.2	嵌入式 Linux 的特点	187
23.3	嵌入式 Linux 系统的开发	187
23.4	uClinux 介绍	192
23.5	其他几款嵌入式 Linux	198

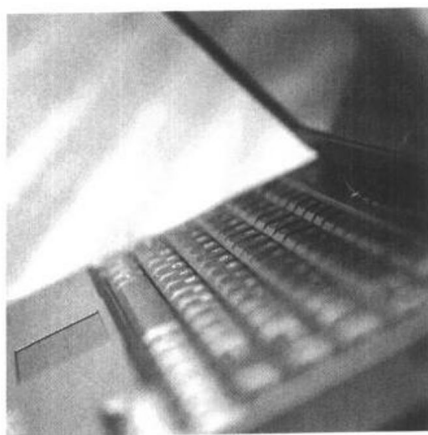
23.6 嵌入式 Linux 展望	200
-------------------------	-----

第 5 篇 嵌入式 Internet 设备开发

第 24 章 嵌入式 Internet	203
24.1 嵌入式 Internet 概述	203
24.2 嵌入式 Internet 的应用	204
24.3 嵌入式 Internet 的原理	205
24.4 嵌入式 TCP/IP	206
24.5 嵌入式 Internet 的开发	210
24.6 嵌入式 Internet 接入	211
第 25 章 嵌入式 TELNET 设备的实现	217
25.1 pSOS 系统 I/O 简介	217
25.2 I/O Switch 表	218
25.3 pSOS 核心对应用程序的接口	218
25.4 驱动程序对 pSOS 核心的接口	219
25.5 pSOS 系统对设备的操作	219
25.6 TELNET 设备挂入系统	220
25.7 TELNET 设备驱动程序的实现举例	221
第 26 章 嵌入式防火墙开发	225
26.1 防火墙技术现状	225
26.2 硬件防火墙的组成	229
26.3 防火墙设计与实现	230
附录 实时操作系统常用词汇与缩略语表	232

第 1 篇

实时操作系统基础知识



原书空白页

第1章 实时系统概述

本章导读：

本章主要讲述实时系统与实时操作系统的基本概念，实时系统的研究领域、技术现状以及未来发展趋势。通过本章的学习可以了解到有关实时系统、实时操作系统、分时操作系统的联系与区别，以及实时系统与实时操作系统的起源、发展和今后的研究发展方向等相关知识，为后续章节的学习做必要准备。

本章主要内容：

- ❏ 操作系统基本概念与分类
- ❏ 实时系统与嵌入式系统的定义、属性和分类
- ❏ 实时操作系统的定义、属性和分类
- ❏ 实时系统与实时操作系统的起源与发展
- ❏ 实时系统与实时操作系统技术现状
- ❏ 实时系统典型应用
- ❏ 实时系统未来发展趋势与研究领域

随着计算机与信息技术的发展，实时系统在日常工作、生活中所占的比重越来越大，在人们的身边随处可见基于实时系统的产品。例如，个人数字助理(PDA)、掌上电脑(HPC)、家电机顶盒(TOP BOX)等等。另外，在国防、科研、交通、能源等领域中实时系统也有着广泛的应用。在电子制导武器中，利用实时系统对搜集的信息进行实时的处理，然后控制导弹的飞行方向，从而有效地摧毁目标。在通信网络系统中，专用交换机、路由器、硬件防火墙等设备，通常也是通过实时系统来实现数据信息的实时处理。

今天，实时系统已经获得了长足的发展，无论是产品的种类还是技术性能指标都得到了质的飞跃。作为实时系统领域的一个重要内容，实时操作系统在整个实时系统领域占有举足轻重的地位。

本章将首先概述操作系统的分类。然后，介绍实时系统的概念与定义，实时操作系统与通用操作系统的联系与区别，实时系统的发展及应用领域等。最后，介绍实时系统的技术现状和发展趋势。

1.1 操作系统概述

1.1.1 操作系统基本概念

操作系统是配置在计算机上的最基本的系统软件，是对硬件功能的首次扩充，其他所有的软件，如编译程序、数据库管理系统等系统软件，以及任何应用程序，都是依靠操作

系统的支持，获得它的服务。操作系统主要实现以下功能：

- 为计算机用户和计算机硬件系统之间提供接口，使计算机系统更容易使用；
- 控制和管理计算机系统的硬件和软件资源，使之得以更有效地利用；
- 合理组织系统中作业的工作流程，以改善系统性能，如响应时间、系统吞吐量等。

1.1.2 操作系统分类

按照操作系统的功能特征可将其分为批处理操作系统、分时操作系统和实时操作系统三种类型。

1. 批处理操作系统

批处理操作系统又可分为两种类型：单道批处理操作系统和多道批处理操作系统。

单道批处理操作系统是一种早期的批处理系统形式，它只在内存中驻留当前正在运行的程序。仅当该程序运行完毕或发生异常情况时，才换入后继程序进入内存运行，程序完成的顺序与其直接进入内存的顺序密切相关，类似于 FIFO（先进先出）的执行顺序。

多道批处理操作系统是基于多道程序设计而形成的一种批处理系统，它允许内存中同时驻留多道程序，不同程序之间交替执行，程序完成的顺序与其进入内存的顺序间无严格的对应关系。从作业进入提交给系统直至完成，需经过作业调度和进程调度几个阶段。

2. 分时操作系统

分时操作系统是指在系统的一台主机上，连接了多个带有显示器和键盘的终端，允许多个用户以分时方式共享一台主机的操作系统或计算机系统。即多个用户可以通过自己的终端以交互方式使用计算机，共享系统中的硬、软件资源。

实现分时操作系统的键是使用户能与自己的作业交互作用，即在自己的终端上键入命令以请求系统服务，系统应能及时接收和及时处理该命令，并将处理结果立即返回给用户。此后，用户又可键入下一条命令，这便实现了人机交互。

实现分时系统的方式有三种：

（1）单道分时系统 即内存中只驻留一道作业，其余作业都在系统外存中。

（2）具有前、后台的分时系统 将系统的内存划分为“前台”和“后台”两部分。“前台”存放调入和调出的作业流；“后台”存放批处理作业。当“前台”无作业运行时，才运行“后台”中的作业。

（3）多道分时系统 可在系统中同时存放多道作业。系统把已具备运行条件的所有作业排成一个队列，使它们依次地轮流获取一个时隙来运行。

分式系统具有多路性、独立性、及时性和交互性的特征。

3. 实时操作系统

实时操作系统是指系统能及时响应外部事件的请求，在规定时间内对该事件进行处理，并控制所有实时任务协调一致地运行的操作系统。

虽然实时操作系统与分时操作系统都是以人、设备或系统所能接受的等待时间来确定任务完成的及时性。但是，实时操作系统的及时性是以控制对象所要求的开始、截止时间或完成时间来确定的，一般为毫秒（ms）级、10 毫秒级或 100 毫秒级，甚至有的要求低于

100 微秒 (μs)。

实时操作系统虽然也具有交互性,但这是系统与人的交互,仅限于系统中特定的应用程序,而分时操作系统则能向用户提供用户数据处理、资源共享等服务。

相对分时操作系统而言实时操作系统要求有更高的可靠性,尤其在具有硬实时要求的系统中,一点点误差都有可能带来严重的危害。

1.2 实时系统基本概念

1.2.1 嵌入式系统

嵌入式系统指的是同时将操作系统和功能软件集成于计算机硬件系统之中的一种系统。简单地说就是应用软件、操作系统与硬件系统一体化,类似于 BIOS (基本输入输出系统)的工作方式。具有软件代码小、高度自动化、响应速度快等特点。特别适合于要求实时的和多任务的体系。

在通信领域中,嵌入式系统早在 20 世纪 60 年代后期就被用于控制电话的电子式机械交换并被称为“存储程序控制系统”。所谓的存储程序是指那些放有程序和路由信息的内存。用软件来存储这些控制逻辑而不是用硬件来实现,这是在观念上的一种突破。

为了适应各个不同的应用,这些嵌入式系统被按照特定需求定做出来。简而言之,这些嵌入式系统是面向应用的。按今天的标准来看,他们有着奇怪的专用指令以及与主要计算引擎集成在一起的 I/O 设备。

CPU 的出现,使得嵌入式系统可以通过 CPU 在大型系统中像搭积木那样,将系统的各个功能部件集成在一起。它利用总线的方式将不同外设所构建的严格的硬件体系结构挂接在一起,并提供一个可以简化编程的通用目的编程模型。

用于嵌入式系统的操作系统在 20 世纪 70 年代后期开始出现,它们中的许多是用汇编语言写成的,并且仅能用于为其编写的微处理器上。当这些微处理器过时的时候,它们使用的操作系统同时面临了一个问题——只能在新的处理器上重新编写一遍才能运行。今天,许多这种早期的系统已渐渐成为了人们模糊的记忆。当 C 语言出现后,操作系统采用一种高效的、稳定的和可移植的方式来编写。这种方式对使用和经营软件的个人或企业有直接的吸引力,因为当微处理器更新时,它能保护个人或企业的软件投资。今天,用 C 来编写操作系统已基本成为了一种标准,即便其中极少数部分由于考虑与硬件打交道的效率问题而仍然使用了汇编语言。但用 C 来编写软件的可复用性已经被广泛接受,并且正在很好地发挥作用。

在 20 世纪 80 年代早期,有一种原始的嵌入式操作系统——Wendon 操作系统大概只需要 150 美元就可以获得它的 C 源代码库。并且 Wendon 提供一个开发套件,用户可以通过选择一些组件来构建自己的操作系统,整个过程就像是从中餐菜单中订餐一样。比如,可以从库中的多个可行的选项列表中选出一种任务调度算法和内存管理方案。

许多用于嵌入式系统的商业操作系统在 20 世纪 80 年代获得了蓬勃发展。Wendon 这一原始的操作系统已经发展成为了商业操作系统这一现代操作系统。今天已经有几大商业性操作系统可供选择,同时也出现了许多相互竞争的产品,如 VxWorks、pSOS、Neculeus 和

Windows CE 等等。

对于嵌入式系统的定义，一种观点认为如果一个应用没有用户界面，它必须是嵌入式的，因为用户不能直接与之交互。当然，这是简单化的概括。例如，一个控制电梯的电脑被认为是嵌入式的系统，通过按键选择楼层，楼层指示灯显示电梯将要到达的楼层。对于联网的嵌入式系统，如果系统包含监视和控制的网络浏览器，这种界限就更加模糊了。更好的定义应该是依据系统的集中功能和主要目的来定义嵌入式系统。

实际应用中的许多嵌入式系统根本就没有操作系统，只不过有一个控制环而已。对于简单的嵌入式系统来说，这可能已经足够。但是，随着嵌入式系统复杂程度的提高，对操作系统的需求显得重要起来，因为没有操作系统的复杂嵌入式系统将会使（控制）软件的复杂度变得极不合理。遗憾的是，现实中确实有一些复杂得令人生畏的嵌入式系统。它们之所以变得复杂，就因为它们的设计者坚持认为它们的系统不需要操作系统。

今天，更多的嵌入式系统需要被连接到某些网络上，因而需要在嵌入式系统中有网络协议栈（支持），例如很多宾馆中的门把手都有一个连接到网络的微处理器。类似这种把网络协议栈添加到一个仅用控制环来实现的嵌入式系统所带来的软件复杂性可能会唤起人们对一个操作系统的渴望。

1.2.2 实时系统

在实际应用中，嵌入式系统常常被错误地称为实时系统，尽管多数嵌入式系统一般并不要求实时功能。实时是一个相对的词，纯理论者常常严格地定义实时为对一事件以预定的方式在极短的时间（如微秒）内做出响应。后来，在如此短暂的时间间隔内，严格实时功能在专用 DSP 芯片或 ASIC 上得到了实现。但这只有在设计底层硬件 FIFO、分散/聚集 DMA 引擎和定制硬件时才会有这样的要求。

许多设计人员因为对具体应用的真实需求没有清晰的理解，因而对实时的要求含糊不清。对于大多数的系统，在 $1\sim 5\mu\text{s}$ 内的近似实时响应就已经足够了。同样，软实时也是可以接受的。如 Windows 98 已经崩溃的中断必须在 4ms 内（ $\pm 98\%$ ）内、或 20ms （ ± 0 ）内进行处理。

软实时是比较容易实现的，包括进程调度时间、中断等待时间、任务优先级和排序等。进程调度时间曾是操作系统的一个热门话题。总之，多数 CPU 对这些要求都处理得很好，而且 CPU 的速度现在已经快了很多，这个问题也就不重要了。

严格的实时要求通常由中断例程或其他内核环境驱动程序处理完成，以确保可靠稳定的等待时间。一旦请求出现，要求服务的时间很大程度上取决于中断的优先级以及其他能暂时取代中断的例程。

如同许多其他的操作系统一样，实时系统的中断必须进行统一管理，以确保时间要求能满足。在 IntelX86 处理器中，这项工作很容易由操作系统进行实时扩展处理。它提供了一个以后台任务方式运行操作系统中断处理的调度。紧急的中断响应不必通知操作系统，它可以通过紧急时钟来实现控制。中断在实时控制级和时间限制要求不高的基本操作系统之间提供接口，它提供与其他嵌入式操作系统相似的实时框架。因此，实时处理的关键代码是单独实现的，并根据特定的需求进行设计。关键代码处理的结果一般都能满足实时任

务的需要。

1. 定义

所谓实时，顾名思义表示立即、及时的意思。如人们常将联机系统视作实时系统，或是将人-机交互性的系统称为实时系统。当然，这都是计算机发展到一定阶段的产物。

实时系统与非实时系统的本质区别就在于实时系统中的任务都有时间限制。时间限制有两种：一种是任务的周期性，按一定周期启动执行的任务称为周期性任务，而随机启动执行的任务称为非周期性任务。另一种时间限制为时限（deadline，最后期限），即要求任务在规定的时间内完成。

实时系统的基本特征是它的实时操作模式，但实际应用中人们对实时系统的理解往往不尽相同，从而导致实时系统这一术语经常被误用。为了准确地解释这一概念，在德国工业标准 DIN44300[2]（1972 年 3 月版本 161 号以及 1985 年 10 月版本的 9.2.11 号）中将其定义如下：

“当外界数据到来时，计算机系统能够立即进行数据处理，以使处理的结果能够满足预订的时间要求；数据到来的时间随机分布并且能依据不同的应用需求进行有效的数据处理。”

在这种操作模式下，对计算机的性能要求与具体的外部应用有关。计算机的处理时间必须与外部事件同步，所以实时系统通常被作为嵌入式系统应用在一个大的环境中，因而又被称为嵌入式实时系统。

实际应用中把对外部事件能在限定时间内作出反应的系统称为实时系统。请注意，这里“限定时间”是一个相对的概念，它视具体应用而定。具体来说，一个实时系统对响应时间和数据吞吐量的要求取决于特定的实时应用。它所指的范围可以从微秒级（如：信号处理）到分级（如：联机查询系统）。

例如：在一个能源管理系统中，其主要的功能是监视和控制环境参数（如温度和空气流量）。这对计算的实时要求不高，是因为它只需要相对较低速的抽样速率和设备响应。然而，在一个发电站监测系统中对数据计算的要求则比较高，必须随时对电站的设备和系统进行监视和控制，以保障安全操作和阻止非人为的停电和拉闸。

2. 实时系统属性要求

实时系统因其实时性要求而与一般的系统有所不同。实时性要求一般有两层含义：时间上同时发生和过程上同时响应。后面将会看到，这涉及实时系统的两类基本属性：可预测性和可靠性。

对实时系统的第一个要求是：数据识别、数据建立和正确响应必须在规定时间内完成。单纯的处理速度并不是决定因素，必须充分考虑到预计的响应时间界限。所以，实时系统的本质不仅依赖于处理结果，也依赖于事件结果的有效性。合理的事件处理应当取决于环境，而不应该强迫它去适应计算机的处理速度。类似地，批处理系统和时间共享系统也是这样。第二个要求是对外部事件处理的同时性，即实时系统必须能进行分布式或并行处理。

实时系统工作模式的定义对实时系统的可靠性非常重要，这是因为这种模式下的实时系统要求计算机能随时处于就绪状态。这些可靠性取决于硬件和软件，这对于那些可能会导致数据丢失甚至危及人身安全以及导致巨大损失的应用尤为重要。