HZ BOOKS
华章教育

计 算 机 科 学 丛 书

Springer

原书第2版

分布式实时系统原理与设计方法

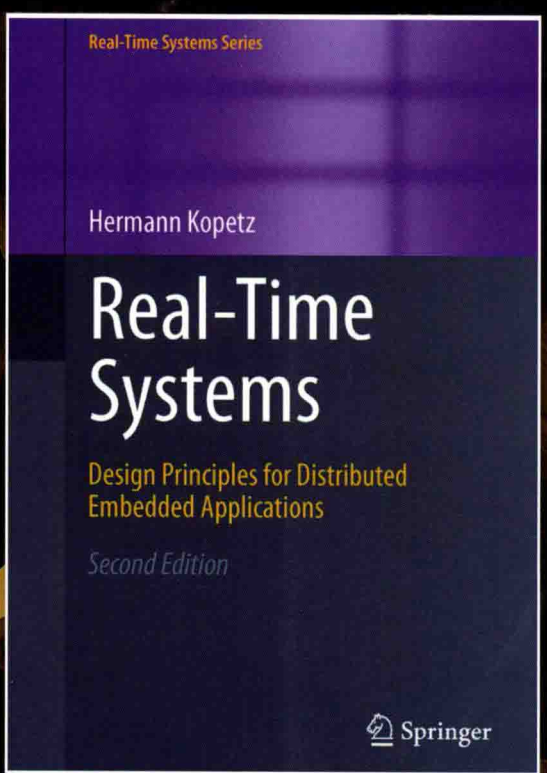
[奥地利] 赫尔曼·科佩茨（Hermann Kopetz） 著

吴际 龙翔 尚利宏 等译

北京航空航天大学

Real-Time Systems

Design Principles for Distributed Embedded Applications, Second Edition



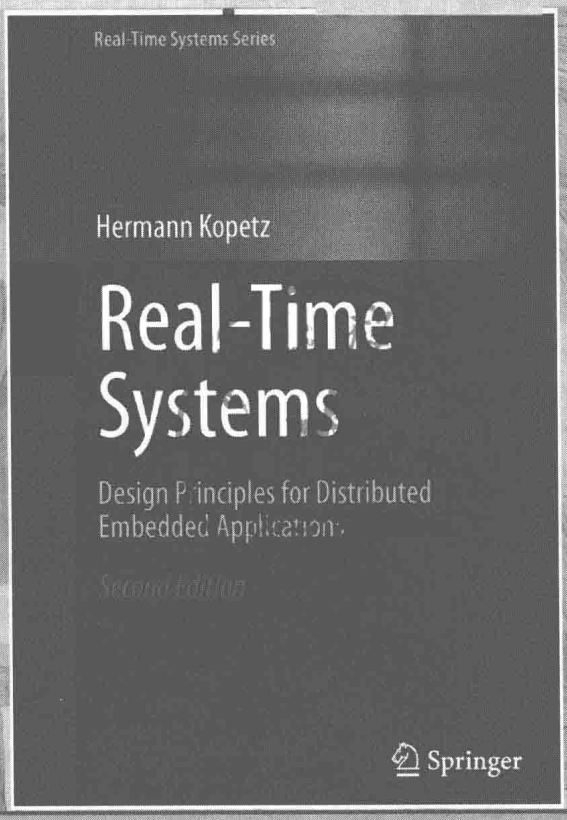
机械工业出版社
China Machine Press

计 算 机 科 学 丛 书

分布式实时系统
原理与设计方法

[奥地利] 赫尔曼·科佩茨（Hermann Kopetz） 著
吴际 龙翔 尚利宏 等译
北京航空航天大学

Real-Time Systems
Design Principles for Distributed Embedded Applications, Second Edition



机械工业出版社
China Machine Press

图书在版编目（CIP）数据

分布式实时系统原理与设计方法（原书第 2 版）/（奥）赫尔曼·科佩茨（Hermann Kopetz）著；吴际等译 . —北京：机械工业出版社，2019.1
（计算机科学丛书）
书名原文：Real-Time Systems: Design Principles for Distributed Embedded Applications, Second Edition

ISBN 978-7-111-61377-0

I. 分… II. ① 赫… ② 吴… III. 分布式操作系统 IV. TP316.4

中国版本图书馆 CIP 数据核字（2018）第 262283 号

本书版权登记号：图字 01-2015-1281

本书主要介绍分布式实时系统的技术原理和设计方法，重点围绕安全关键实时系统的行为确定性、可组合性和容错能力等难题提出了时间触发机制和相应的设计原则。本书整体内容分为四大部分，包括关于时间的基础理论、围绕实时性的平台技术介绍、围绕实时性和可信性的系统设计与确认，以及关于物联网和时间触发体系结构的最新进展。为了阐述相关概念和方法，本书结合三个样例系统提供了大量的案例解析，并贯穿始终。

本书适合作为计算机科学、计算机工程和电子工程相关学科的高年级本科生或研究生的教材。

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号　邮政编码 100037）
责任编辑：朱秀英　　责任校对：殷　虹
印　　刷：三河市宏图印务有限公司　　版　　次：2019 年 1 月第 1 版第 1 次印刷
开　　本：185mm × 260mm　1/16　　印　　张：17.5
书　　号：ISBN 978-7-111-61377-0　　定　　价：89.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88378991　88361066　　投稿热线：（010）88379604
购书热线：（010）68326294　88379649　68995259　　读者信箱：hzjsj@hzbook.com

文艺复兴以来，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的优势，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭示了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自 1998 年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与 Pearson、McGraw-Hill、Elsevier、MIT、John Wiley & Sons、Cengage 等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出 Andrew S. Tanenbaum、Bjarne Stroustrup、Brian W. Kernighan、Dennis Ritchie、Jim Gray、Afred V. Aho、John E. Hopcroft、Jeffrey D. Ullman、Abraham Silberschatz、William Stallings、Donald E. Knuth、John L. Hennessy、Larry L. Peterson 等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力相助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专门为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近 500 个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段，我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下：

华章网站：www.hzbook.com
电子邮件：hzjsj@hzbook.com
联系电话：（010）88379604
联系地址：北京市西城区百万庄南街 1 号
邮政编码：100037

华章科技图书出版中心

译 者 序

Real-Time Systems: Design Principles for Distributed Embedded Applications, Second Edition

知悉本书第 1 版大概是 2010 年，那时我因科研项目的需要开始接触嵌入式实时系统，虽然还称不上硬实时系统。做领域调研分析时，我从网络上找相关资源时看到了本书的介绍。阅读的第一印象是该书提供了非常坚实的理论分析，同时概念也非常简洁清晰。当时绝对没有想到我会在四年后接下本书第 2 版的翻译工作。翻译此书的提议是在拜访本书作者（维也纳，2014 年 7 月）之前提出的，并在拜访中经过交流讨论最终形成决定。与本书作者 Hermann Kopetz 的认识则可追溯到 2012 年，那时我与 Simula 研究所的 Tao Yue 和 Bran Selic 共同策划 SafeMOVE 2013 国际研讨会，并邀请 Hermann Kopetz 来北京做主题报告和关于时间触发技术的专题培训。

不得不说，本书的翻译具有相当大的难度，甚至可以说具有挑战性。实时系统的行为确定性、设计的可组合性和故障管理等都是艰深的主题，需要对系统全局特性有深入的理解。准确和完整理解这特性的主要困难在于系统的时域行为，需要在各个抽象层次上开展分析和设计，而这一直是经典的实时系统著作有所欠缺的地方。

时间触发技术恰恰就是为了这个目标所提出的，在对时间进行精确度量和控制的基础上，逐步构造组件行为和组件间的交互行为，并以可组合的方式实现整个系统行为的确定性控制目标。时间触发技术涉及全局时间、时间推进的任务控制、实时调度、冗余和容错、通信控制、设计原则和体系结构风格等多方面的方法和技术。经过多年的发展和应用，时间触发技术已形成了相应的国际标准，并在航空、航天、汽车等领域的核心系统中得到了广泛应用。

随着网络通信技术的不断发展，实时性正在得到越来越多的关注，目前时间敏感网络（TSN，可近似理解为松弛的时间触发技术）正呈现井喷式的发展态势。目前国内对实时系统的设计技术与验证技术的关注度越来越高，有不少科研单位听闻我们在翻译本书后，多次表达了希望尽快看到中文版出版的意愿。

在整个翻译过程中得到了本书作者的大量无私帮助。我们经常就一些关键概念和论述与作者进行沟通。了解越多，越是由衷佩服作者的睿智和前瞻性。非常感谢作者专门为本书的中文版作序。在组织译文的过程中，我们意识到准确理解相关核心概念的重要性，而很多概念涉及多方面的知识，甚至是跨领域的知识。为了增强中文版的可读性，我们在每一章都增加了一定数量的必要译者注，为相关概念补充一定的解读分析。

本书的翻译历时四年，经过了四个阶段：初步翻译阶段、整理阶段、分章节校对阶段和整体校对阶段。有不少研究生参与了初步翻译阶段的工作，按照章节独立进行了文字翻译。在整理阶段，对相关概念的把握分析和统一翻译是主要工作。在分章节校对阶段，主要解决论述表达方式和中文字句组织的差异化问题，尽可能使各章统一。最后的整体校对阶段，通读全书并统一术语。每一遍校对都能发现一定数量的逻辑问题和文字表达问题，我们意识到“英文味”的译文一定不能让读者满意，因此在确保遵从原作的论述逻辑的前提下，尽量按照中文表达方式来组织译文。

龙翔老师承担了本书第 3、4、8、10 章的翻译工作，赵永望老师完成了第 5、6、9、13

章的初步翻译，尚立宏老师对第 5、6、9 章进行了修订翻译，李云春老师则对第 13 章进行了修订翻译。吴际老师承担了第 1、2、7、11、12、14 章的翻译，并负责全书的校对、翻译修订和译者注梳理工作。在翻译过程中得到了高小鹏老师、刘超老师和马殿富老师的大力支持和帮助，也得到了 TTTech 中国分公司的欧阳杨经理的很多关心和支持。陈逊、胡京徽、吕佳辉、姜徐、孙思杰、谭宇、杨经纬、燕保跃、张峰等同学参与了本书翻译的相关工作，在此一并致谢。

限于译者的业务水平，对原作的理解和译文的组织都不可避免地存在错误和不足，希望读者在阅读中指正并告知我们。

吴际

于北京航空航天大学

2018 年 7 月

中文版序

Real-Time Systems: Design Principles for Distributed Embedded Applications, Second Edition

In the past twenty years, the domain of *real-time embedded systems*—nowadays they are often called *cyber-physical systems*—has grown in importance in industry and academia.In these systems, the *time-less cyber world* has to interact with *physical processes* that are governed by natural laws that are based on the progression of physical time. While many computer scientists abstract from physical time and focus on symbol manipulation in cyberspace, this book considers physical time a first order citizen that cannot be neglected in cyber-physical systems. The explicit consideration of physical time—as demonstrated in the *time-triggered architecture*–can contribute to a significant simplification of many industrial control problems. Since the first edition of this book has been published twenty years ago, the *time-triggered architecture* has been successfully deployed in a number of aerospace, industrial and automotive applications.

The fundamental design principles for embedded real-time systems that were published twenty years ago in the first edition of the book are still valid today. In the second edition of the book, some additional chapters on *cognitive complexity, energy awareness*, and the *internet of things* have been added, with only minor changes to the central parts of the first edition.

The translation of a technical book from one language—let us call it the *source language* (in our case *English*)—to another language—let us call it the *target language* (in our case *Chinese*)—is a challenging endeavor. In a first phase, the translator must gain an understanding of a *thought* represented by words in the source language. In a second phase the translator must express this thought in words that can be comprehended by persons that are familiar with the target language. Although the representation of the thought is radically changed, the content of the thought must remain the same. As an author of the book who does not speak the target language, I can—and did—support the translator in the first phase of the translation. By using email we could clarify some topics to ensure that the translator had a full understanding of the *thoughts* expressed in the source language. I am therefore confident to assume that the second phase of the translation—the finding of proper words that express the meaning of the thoughts in the target language—is well done. I would like to thank the translators for their immense efforts in doing the translation.

I hope that the reader of this book will enjoy the study of the material and will gain a deep insight in the design principles of embedded real-time systems and the rational for the time-triggered architecture.

H. Kopetz

July 10, 2018

本书适合用作高年级本科生或一年级研究生的实时嵌入式系统（也称为信息物理融合系统）课程的教材，首要目标是系统地介绍相关知识。本书内容划分为 14 章，正好对应一个学期的 14 周教学。本书也可作为技术参考书，向工业界的实践者提供实时嵌入式系统设计的现状，以及该领域涉及的基础性概念。从本书第 1 版出版至今的 14 年间，维也纳技术大学有超过 1000 名学生使用该书作为教材来学习实时系统课程。这些学生的反馈和嵌入式实时系统这个动态变化领域的许多新进展，都融入了第 2 版中。本书关注体系结构层次的分布式实时系统设计。然而我们发现，相当大一部分计算机科学文献都忽略了实时时间的推进，这使得实时系统设计者不掌握这个关键知识的抽象层次就无法开展系统设计工作。因此，物理时间推进是本书中最重要的概念，在此基础上定义很多相关的概念。本书使用大量来自工业界的案例来洞察解释与时间推进相关的基础性概念。本书扩展了分布式实时系统的概念模型，并精确定义了与时间相关的重要概念，如稀疏时间、状态、实时数据的时域精确性和确定性等。

大规模计算机系统的认知复杂性演化是个极为受关注的主题，第 2 版专门增加了一章来论述简约设计（第 2 章）。本章采纳了认知领域的一些最新研究发现，包括概念形成、理解、人类的简化策略、模型构建，并形成了有助于简约系统设计的 7 个原则。在后续的 12 章中，都围绕这些原则展开论述。另外还新增了两章，分别是第 8 章和第 13 章，论述移动设备这一巨大市场中越来越重要的主题。关于第 6、7、11、12 章都进行了系统性修订，并特别关注基于组件的设计和基于模型的设计。在第 6 章中，新增了关于信息安全和功能安全的多个小节。第 14 章介绍了时间触发体系结构，把本书论述的概念整合成连贯一致的框架，用来开发可信嵌入式实时系统。自本书第 1 版出版以来，在许多应用领域都可以清楚地看到，已经从采用事件触发设计方法学转向采用时间触发设计方法学来设计可信分布式实时系统。

本书假设读者拥有计算机科学或计算机工程方面的背景知识，或者在嵌入式系统设计、实现方面有一些实践经验。

作为不可分割的组成部分，本书最后对贯穿全书的技术术语给出了相应定义。如果读者在阅读过程中不确定某些术语的确切内涵，建议参考术语定义部分。

致谢

无法在这里一一列举所有对本书第 2 版有贡献的学生、工业界和科学界同行的姓名，他们在过去十几年为本书提出了诸多富有启发的问题或给出了建设性的评论。在完成本书第 2 版的最后阶段——2010 年 10 月，我在范德堡大学讲授一门由 Janos Sztipanovits 组织的课程，从听众那里得到了宝贵的意见。在这里要特别感谢 Christian Tessarek，他承担了本书的插图设计工作。感谢阅读了部分或全部手稿并提出了许多宝贵修改建议的 Sven Bünte、Christian El-Salloum、Bernhard Frömel、Oliver Höftberger, Herbert Grünbacher、Benedikt Huber、Albrecht Kadlec、Roland Kammerer、Susanne Kandl、Vaclav Mikolasek、Stefan Poledna、Peter Puschner、Brian Randell、Andreas Steininger、Ekarin Suethanuwong、Armin Wasicek、Michael Zolda，以及来自范德堡大学的学生 Kyoungho An、Joshua D. Carl、Spencer Crosswy、Fred Eisele、Fan Qui 和 Adam C. Trewyn。

目　录

Real-Time Systems: Design Principles for Distributed Embedded Applications, Second Edition

实时环境

概述 作为本书的引言，本章从多个视角来介绍实时计算机系统的运行环境。深入理解实时应用的特征性技术因素和经济因素，有助于解释对系统设计人员所提出的设计要求。本章首先介绍实时系统的定义，讨论其功能需求和非功能需求，并特别关注那些从熟知的控制应用特性推导而来的时域需求。控制算法的目标是控制一个过程使得其性能满足相应的准则。系统运行环境中出现的随机扰动会降低系统运行性能，这是控制算法设计必须要考虑的因素。任何由控制系统本身在控制回路中引入的额外不确定性（如不可预测的控制回路时间抖动）都会降低控制品质。

本章 1.2 ～ 1.5 节从多个视角介绍实时应用的分类，特别关注硬实时系统和软实时系统的根本性差异。因为软实时系统不会出现严重的失效模式，允许使用欠严格的方法来进行设计，所以有时为了经济考虑，会采用资源受限的设计方案，不对极少出现的峰值负载场景做专门处理。但是对于硬实时系统，由于要保证在所有已知场景下的安全性，就不能使用这样的设计方法，即使一些特殊场景的出现概率极小，也必须向认证机构如实[⊖]论证系统的安全性。1.6 节简要分析实时系统市场，重点论述嵌入式实时系统。嵌入式实时系统是自包含系统的组成部分，如电视机或汽车。一般来说，又称为 CPS（Cyber-Physical System，信息物理融合系统）的嵌入式实时系统是实时技术和计算机行业最重要的用武之地。

1.1 实时计算机系统

实时计算机系统的行为正确性不仅取决于它的逻辑计算结果，也取决于计算结果的输出物理时间。本书使用系统行为来表示系统按时间产生的输出序列。

本书使用从过去到未来的有向时间线来描述时间的流动。时间线上的切点称为时刻。恰好[⊜]在一个时刻发生的任何出现都称为事件。用来描述一个事件（见 5.2.4 节的事件观测[⊜]）的信息称为事件信息。当前时间点，称为现在，是一个非常特殊的事件，可以区分过去和将来（这里采用基于牛顿物理学的时间模型，但不考虑时间的相对效果）。时间线上的区间称为时间间隔，由该区间的起始事件和终止事件来定义。数字时钟把时间线划分为等间距的时间间隔序列，称为时钟的颗粒度，它由特别的周期性事件即时钟节拍来界定。

*实时计算机系统总是一个更大系统的组成部分。这个更大的系统称为实时系统或信息物理融合系统。*实时系统遵循物理时间的某个函数来改变自己的状态，如化学反应系统持续改

⊖ 原文是“vis-a-vis”，表示面对面，这里翻译为“如实”。——译者注

⊜ 原文为 ideal occurrence，表示理想情况下在时间轴切线时刻点发生的出现，这里译为“恰好”。因为按照本书对时间的定义，“时刻”是时间的最小单位，在时刻之外发生的事情其实都是不可知的，系统无法感应。——译者注

⊜ 原文为 observation，中文翻译有“观察”和“观测”两种方案。从中文角度来看，“观测”与“观察”有细微区别，前者更多指需要使用相关设施获得观察，并进行必要的量化处理后才能有最终结果。故本书统一翻译为“观测”。

变自己的状态，即便控制它的计算机系统已停止运行。一个实时系统可以被合理地分解成一组自包含的子系统，称为集群（cluster）。典型的集群（如图 1-1 所示）包括受控的物理设备或机器（受控集群）、实时计算机系统（计算集群）以及操作员（操作集群）。本书把受控集群和操作集群作为整体，统称为计算集群的运行环境。

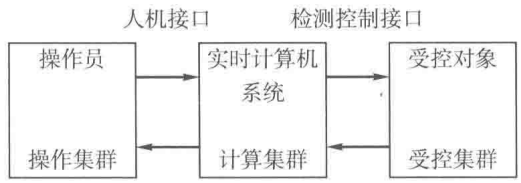


图 1-1　实时系统

分布式（事实上大部分都是）实时计算机系统由一组通过实时通信网络进行连接的节点（即计算机）组成。操作员和实时计算机系统之间的接口称为人机接口，实时计算机系统和受控对象之间的接口则称为检测控制接口。人机接口由输入设备（如键盘）和输出设备（如显示器）组成，为操作员提供操作界面。检测控制接口由传感器和作动器组成，用来把受控集群中的物理信号（如电压、电流）转换为数字信号，以及把数字信号转换为物理信号。

实时计算机系统必须在其环境所要求的时间间隔内对来自环境（受控集群或操作集群）的激励做出响应。必须产生处理结果的时刻称为截止时间。如果系统在截止时间之后产生的结果还能发挥作用，则称为软截止时间，否则称为严格截止时间。如果系统错过了严格截止时间会导致严重后果，则称为硬截止时间。

例：考虑铁路交叉路口的信号灯，如果在火车经过路口前未能把信号灯变为红色，就可能会导致事故。

必须满足至少一个硬截止时间的实时计算机系统称为硬实时计算机系统，或安全关键实时计算机系统。如果一个系统没有硬截止时间，则称为软实时计算机系统。

硬实时系统的设计和软实时系统的设计有根本的不同。软实时计算机系统偶尔错过截止时间是可接受的，但是硬实时计算机系统必须要确保在任何负载和故障条件下的时域行为都能满足要求。这两类系统的差异将在本书后续章节详细阐述。本书关注硬实时系统的设计。

1.2　功能需求

实时系统的功能需求关注实时计算机系统必须执行的功能，包括数据采集需求、直接数字控制需求和人机交互需求。

1.2.1　数据采集

受控对象，如汽车或工业设备，会按照时间的某个函数来改变其状态（本书中如果在时间前未加修饰词，则指 3.1 节所阐述的物理时间）。假如把时间冻结，则可以通过记录在冻结时刻的状态变量取值来描述什么是当前状态。对于汽车这个受控对象而言，状态变量可以包括汽车的位置、速度、仪表盘上各个开关的位置以及气缸里活塞的位置等。通常我们不需要关注所有可能的状态变量，而只关注其中与设计目标显著相关的状态变量子集。与目标显著相关的状态变量又称为实时实体（简称 RT 实体）。

每个 RT 实体都处于某个子系统的控制范围（Sphere of Control，SOC）内，即它属于授

权对该 RT 实体取值进行改变的某个子系统（参见 5.1.1 节）。RT 实体可以在控制范围之外被观测到，但是其语义内容（参见 2.2.4 节）不能被修改。例如，尽管在汽车引擎之外可以观测到气缸中活塞的当前位置，但不允许对这个观测的语义内容进行修改（语义内容的表示则可以被改变）。

实时计算机系统的首要功能需求是观测受控集群中的 RT 实体并收集观测数据。计算机系统通过实时（RT）镜像来表示对 RT 实体的一次观测。因为受控集群中受控对象的状态是关于时间的函数，一个给定 RT 镜像只在有限时间段内可以准确表示 RT 实体的当前状态。这个时间段长度取决于受控对象的动态特性。如果受控对象的状态改变非常快，相应 RT 镜像的时域精确范围就会很短。

例：如图 1-2 所示，一辆汽车进入由交通信号灯控制的交叉区域。“绿灯”这个观测在多长时间内能精确表示交通灯实际状态？如果不在其时域精确范围内使用“绿灯”这个观测，即车在绿灯转换为红灯后进入交叉区域，就可能导致事故。在该例子中，时域精确范围的上限由交通灯处于黄色状态的时间间隔给出。

图 1-2 交通灯状态观测的时域精确性

受控集群的所有时域精确的 RT 镜像集合称为实时数据库，一旦有 RT 实体的值发生改变，该数据库就必须更新。可以周期性地实施更新，由实时时钟按照固定周期来触发（称为时间触发（TT）观测），或者在状态改变时立刻发生（称为事件触发（ET）观测）。关于时间触发观测和事件触发观测会分别在第 4 章和第 5 章详细分析。

信号调节。物理传感器，如热电偶，会产生原始数据（如电压）。通常，在收集到原始数据序列后需使用均值算法来减少测量误差。接下来，必须对原始数据进行校准并转换到标准的测量单位。信号调节这个术语专指所有从原始传感器数据获得关于 RT 实体有意义测量数据的必要处理步骤。在信号调节后，必须要对测量数据的合理性进行检查，并与其他测量数据进行关联分析以检测可能的传感器故障。RT 实体的一个数据如果被判定为正确的 RT 镜像，则该数据称为议定数据。

告警监视。持续监视 RT 实体以检测处理行为出现的异常是实时计算机系统的一个重要功能。

例：化工厂工控系统的管道爆裂是个需要关注的首要事件，它会导致诸多 RT 实体（如多处的压力、温度和液面）取值与它们常规运行下的范围不符，并会突破预设的告警门限，从而产生一系列关联的警报，称为爆发式警报。

实时计算机系统必须要检测和显示这些警报，并辅助操作人员识别导致这些警报的初始首要事件。为了达到这个目标，在观测到警报后，系统必须在一个特殊的警报日志中记录警报出现的确切时刻。准确记录警报发生时序有助于识别次级警报，即由首要事件引起的所有