

本书由多年从事 MATLAB 软件教学、研究和开发的专家主编

系统实时仿真 开发环境与应用

MATLAB/RTW and It's Application



杨 涤 李立涛
杨 旭 朱承元 编著



清华大学出版社
<http://www.tup.tsinghua.edu.cn>

系统实时仿真开发环境与应用

杨 涤 李立涛
杨 旭 朱承元 编著

清 华 大 学 出 版 社

(京) 新登字 158 号

内 容 简 介

RTW(Real-Time Workshop, 实时工作间)是 MATLAB 的重要组成部分,也是完成系统的技术实现、部件与系统性能测试的有效开发环境,可应用于实时系统的快速原型化、硬件在回路中的仿真、系统半实物仿真或全物理仿真等方面。RTW 与 MATLAB/Simulink 的无缝连接为实现 CAD 与实时仿真一体化提供了最好的技术途径。

本书系统、全面地介绍了 RTW 的组成、功能和操作方法,及其在系统实时仿真等方面的应用。同时,本书还重点针对两种基于 RTW 环境的实时仿真和开发平台——xPC 和 dSPACE,介绍了这两种软件环境的使用方法,及其在航天控制中的实际应用。

本书重点突出 MATLAB 在系统技术实现、系统性能测试等实时操作方面的应用。同时,本书注重实例和应用,图文并茂、易学易用。

本书既可以作为大专院校本科生、研究生系统学习 MATLAB/RTW 的教材,也可以作为广大科研人员、工程技术人员掌握与应用系统 CAD 与实时仿真一体化技术的自学教材。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

系统实时仿真开发环境与应用/杨滌等编著. —北京:清华大学出版社, 2002
ISBN 7-302-05879-2

I.系... II.杨... III.计算机辅助计算—软件包, MATLAB IV.TP391.75
中国版本图书馆 CIP 数据核字(2002) 第 069250 号

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 张秋香

封面设计: 王伟

版式设计: 康博

印 刷 者: 北京密云胶印厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 29 字数: 688 千字

版 次: 2002 年 10 月第 1 版 2002 年 10 月第 1 次印刷

书 号: ISBN 7-302-05879-2/TP·3486

印 数: 0001~4000

定 价: 46.00 元

前 言

MATLAB 是美国 MathWorks 公司开发的软件产品, 是一种使用简便的工程计算语言、完成系统从概念设计到技术实现全过程设计的 CAD 工具集, 是目前世界各国科学研究与工程设计领域普遍采用的标准设计软件。近年来, 国内各高校和研究部门越来越重视对 MATLAB 的学习与应用, 有关 MATLAB 软件的教材或书籍已普遍用于国内大学, 对工科大学本科生和研究生培养起到了重要作用。

目前, 国内有关介绍 MATLAB 及其工具箱的书籍和教材, 多数是重点介绍 MATLAB 软件在系统概念设计或方案设计阶段的功能与应用, 如软件的工程计算、系统设计与分析、系统建模与数学仿真等功能。很少有反映 MATLAB 软件在系统设计技术实现阶段的功能与作用, 如实现系统的快速原型化、硬件在回路中的仿真、系统半实物仿真及系统或部件的性能测试等方面的功用。

RTW(Real-Time Workshop)是 MathWork 系列软件的重要组成部分。RTW 与 MATLAB 其他组成软件的无缝连接, 既满足了设计者在系统概念与方案设计等方面的需求, 也为系统的技术实现或完成不同功能的系统实时操作试验提供了方便。

本书力求在作者多年应用 MATLAB 软件教学与科研的基础上, 较全面地介绍 RTW 及其在系统实时仿真与开发方面的功用, 可使读者方便、快速地完成不同功用的系统实时操作实验和系统开发工作, 为提高读者的科研与设计水平, 缩短设计与研制周期、降低系统的研制成本, 提供了一个最佳的设计与开发环境。本书注重工程应用, 结合作者在航天、航空领域的科研与教学, 加强了具体工程应用方面的内容, 使读者能够借鉴本书的实例, 联系自己的工程实际问题, 尽快掌握所学的技术和方法。

本书以 MATLAB 较新的版本 6.0(Release 12)为进行介绍, 直接面向 MATLAB 的高级用户。在阅读和学习本书之前, 读者应基本掌握 MATLAB 的 Simulink 仿真环境的基础知识和使用方法, 以及基本的 C 语言编程经验。

本书共分 4 部分, 第 1 部分主要介绍 RTW 的基本知识、使用方法及其在系统仿真和实时系统方面的应用; 第 2 部分和第 3 部分主要介绍两种基于 RTW 的实时系统开发、仿真平台的使用方法和具体应用实例。前者定位于低造价、易实现的实时系统(用普通 PC 机即可实现实时系统的快速原型化), 后者则定位于高端用户, 满足用户对高性能的实时系统仿真平台的要求。第 4 部分为附录部分, 主要介绍 RTW 术语表和 xPC 目标环境。全书的主要章节安排如下。

- 第 I 部分 RTW 代码自动生成和实时系统开发环境(1~8 章)

第 1 章主要介绍 RTW 的基本情况、功能和特点, 及其安装方法。

第 2 章介绍与 RTW 有关的基本概念和术语, 通过一个实例练习使用户初步了解 RTW 用

户界面, 以及代码生成和程序创建过程, 并简要介绍 RTW 的开放式体系结构和自动的程序创建过程。

第 3 章详细介绍 RTW 用户界面及其使用方法, 以及如何根据 RTW 设置仿真参数, 如何配置 RTW 来创建用户需要的应用程序。最后将介绍如何从子系统中生成代码。

第 4 章通过一个练习介绍外部模式的基本使用方法, 对外部模式用户界面进行详细介绍, 以及如何通过基于 TCP/IP 的通信协议来实现外部模式通信。

第 5 章对 RTW 生成的两种代码类型的程序体系结构进行介绍, 包括快速原型化和嵌入式代码类型。

第 6 章对 RTW 的某些高级应用, 例如如何设置代码格式、代码的优化、多采样频率模型处理和如何加入自定义代码等进行较详细的介绍。

第 7 章对 RTW 提供的几种具有实用价值的目标环境的使用方法进行介绍。

第 8 章介绍如何根据用户的要求生成自定义的目标配置, 生成自定义的设备驱动程序以及如何生成外部模式的通信协议。

• 第 II 部分 xPC 目标环境的应用(9~14 章)

第 9 章对 xPC 目标环境及其特征, 安装和使用 xPC 目标所需的软件和硬件环境进行简要的介绍, 最后介绍 xPC 目标提供的几种用户交互方式。

第 10 章介绍如何安装 xPC 目标所需的软件和硬件环境, 和实现 xPC 目标的通信连接。

第 11 章通过一个实例介绍 xPC 目标的基本使用方法。

第 12 章主要介绍 xPC 目标的高级应用部分, 包括如何将设备驱动模块和 xPC 目标的专用示波器模块加入到模型中, 以及一些目标程序的控制方法。

第 13 章对 xPC 目标嵌入式选项模块进行介绍, 以及如何创建 DOS 系统启动盘和生成 DOS Loader 型及单机型目标应用程序。

第 14 章通过一个具体的应用实例, 对 xPC 目标的使用方法和具体应用做一个演示。

• 第 III 部分 dSPACE 实时系统平台的应用(15~18 章)

第 15 章简要介绍 dSPACE 实时仿真系统的主要功能、软件与硬件产品以及产品的主要功能。

第 16 章介绍如何使用 RTI 或 RTI-MP 软件环境实现实时系统模型的建立以及生成实时代码。

第 17 章较详细地介绍 ControlDesk 的用户界面和基本操作方法, 再以示例的形式介绍使用 ControlDesk 完成实验和测试的一般步骤, 以及如何使用 ControlDesk 进行 Simulink 的数学仿真。

第 18 章通过一个卫星敏感器姿态确定仿真实验系统的具体应用实例, 对 dSPACE 实时仿真系统的使用方法和具体应用做一个演示。

在本书编写过程中, 程杨博士、王淑一硕士也参与了部分组织和编写工作, 在此表示衷心的感谢。

由于作者水平有限、经验不足, 错误和不当之处在所难免, 恳请读者和专家批评指正。

目 录

第 I 部分 RTW 代码自动生成和实时系统开发环境

第 1 章	RTW 基础	1
1.1	RTW 简介	1
1.1.1	RTW 的概念	1
1.1.2	使用 RTW 的原因	2
1.1.3	RTW 的主要功能和特征	5
1.1.4	RTW 的应用	7
1.1.5	RTW 支持的代码格式	9
1.1.6	RTW 支持的目标环境	10
1.1.7	开放的、可扩展的建模环境	14
1.2	RTW 的安装	15
1.2.1	安装 RTW	15
1.2.2	相关的 MathWorks 工具箱或产品	16
1.3	RTW 支持的编译器及安装	18
1.3.1	RTW 支持的编译器	18
1.3.2	在 Windows 下安装第三方编译器	18
第 2 章	RTW 入门	20
2.1	RTW 的基本概念	20
2.2	创建一个简单的目标程序	21
2.2.1	创建一个通用实时目标程序	23
2.2.2	数据记录	27
2.2.3	代码校验	30
2.3	RTW 自动程序创建过程简介	32
2.3.1	程序创建过程	33
2.3.2	程序创建过程中生成文件	35
2.4	RTW 的开放式体系结构	36
第 3 章	RTW 的代码生成和程序创建过程	38
3.1	使用 RTW 的用户界面	38

3.1.1	Real-Time Workshop 选项卡的使用	38
3.1.2	设置目标配置选项	39
3.1.3	设置通用代码生成选项	41
3.1.4	设置目标专用代码生成选项	44
3.1.5	使用 TLC 调试选项	45
3.1.6	使用 RTW 子菜单	46
3.2	仿真参数配置	46
3.2.1	设置 Solver 选项卡	46
3.2.2	设置 Workspace I/O 选项卡和数据记录	47
3.2.3	设置 Diagnostics 选项卡	48
3.2.4	设置 Advanced 选项卡	49
3.2.5	辨识原模型中的模块	53
3.2.6	Simulink 和 RTW 之间的关系	54
3.3	RTW 程序创建过程及配置	56
3.3.1	选择和配置编译器	56
3.3.2	选择目标配置文件	57
3.3.3	通过 TLC 对所生成的代码进行配置	60
3.3.4	配置模板联编文件和 Make 选项	62
3.3.5	创建可执行程序	66
3.4	使用 RTW 的高级功能	67
3.4.1	从子系统中生成代码	67
3.4.2	从非虚拟型子系统生成代码	68
第 4 章	外部模式	71
4.1	外部模式简介	71
4.2	外部模式快速入门	72
4.2.1	模型的设置	72
4.2.2	建立目标可执行程序	73
4.2.3	在外部模式下运行目标程序	76
4.2.4	在外部模式下进行参数调整	78
4.3	外部模式用户界面的使用	78
4.3.1	与外部模式相关的菜单和工具栏	78
4.3.2	使用 External Mode Control Panel 对话框	81
4.3.3	建立通信连接和仿真功能控制	82
4.3.4	使用 External Target Interface 对话框	82
4.3.5	使用 External Signal & Triggering 对话框	83
4.3.6	数据存档设置	86

4.3.7 使用参数下载选项	87
4.4 外部模式的通信及其 TCP/IP 实现	88
4.4.1 外部模式下载机制	88
4.4.2 RTW 的 TCP/IP 技术实现	89
4.5 与外部模式兼容的模块和子系统	93
4.5.1 与外部模式兼容的模块	93
4.5.2 信号查看子系统	94
4.6 外部模式的限制	95
第 5 章 RTW 程序的框架结构	97
5.1 模型的执行	97
5.1.1 模型代码的执行过程	98
5.1.2 程序的定时	104
5.1.3 程序的执行	105
5.1.4 外部模式通信	105
5.1.5 执行单/多任务模型时的数据记录	105
5.1.6 快速原型化程序与嵌入式程序的执行过程	106
5.1.7 快速原型化程序相关函数	106
5.1.8 嵌入式程序相关函数	110
5.2 快速原型化程序框架	110
5.2.1 快速原型化程序体系结构	111
5.2.2 快速原型化程序系统相关组件	112
5.2.3 快速原型化程序系统独立组件	113
5.2.4 快速原型化程序应用程序组件	114
5.3 嵌入式程序框架	117
第 6 章 RTW 的高级应用	119
6.1 配置代码格式	119
6.1.1 选择代码格式	119
6.1.2 实时代码格式	121
6.1.3 实时 malloc 代码格式	122
6.1.4 S 函数代码格式	122
6.1.5 嵌入式 C 代码格式	122
6.2 对 RTW 生成代码进行优化	123
6.2.1 常用的建模技术	123
6.2.2 模型图的性能调整	124
6.2.3 Stateflow 的优化设置	135

6.2.4	仿真参数的优化设置	135
6.2.5	编译器选项的优化设置	136
6.3	多采样频率模型	136
6.3.1	单任务和多任务环境	137
6.3.2	采样速率的过渡	141
6.4	使用自定义代码模块	146
6.4.1	使用模型代码模块库	147
6.4.2	使用子系统代码模块库	149
6.5	使用异步中断	150
6.5.1	中断处理	151
6.5.2	生成自定义的异步模块库	161
第 7 章	RTW 目标环境及使用	163
7.1	S 函数目标	163
7.1.1	从子系统中生成 S 函数模块	164
7.1.2	对生成的 S 函数进行参数调整	167
7.1.3	S 函数的自动生成	168
7.1.4	S 函数目标的限制	170
7.2	RTW 快速仿真目标	171
7.2.1	创建快速仿真目标程序	171
7.2.2	运行快速仿真目标程序	172
7.2.3	仿真性能	177
7.2.4	批参数仿真和蒙特·卡洛型仿真	177
7.3	Tornado 目标	177
7.3.1	Tornado 目标运行时结构简介	178
7.3.2	实现过程	181
7.4	嵌入式代码生成器	188
7.4.1	嵌入式代码的数据结构和代码模块	188
7.4.2	嵌入式程序的执行	190
7.4.3	S 函数封装器的自动生成	196
7.4.4	代码优化	197
7.4.5	使用高级代码生成选项	201
7.4.6	嵌入式代码生成器的要求和限制	201
第 8 章	定制自己的实时系统	203
8.1	自定义目标配置的组成	203
8.1.1	代码组成部分	203

8.1.2	需要用户编写的运行时界面代码	204
8.1.3	用于快速原型化的运行时界面	204
8.1.4	用于嵌入式目标的运行时界面	205
8.1.5	控制文件	205
8.2	生成一个自定义的目标配置	206
8.3	自定义程序创建过程	209
8.3.1	系统目标文件的结构	210
8.3.2	将自定义目标配置添加到系统目标文件浏览器中	215
8.3.3	模板联编文件	216
8.4	生成自定义设备驱动程序	222
8.4.1	内嵌和非内嵌型设备驱动程序	222
8.4.2	设备驱动程序的要求和限制	223
8.4.3	设备驱动模块的参数化	224
8.4.4	编写非内嵌的 S 函数设备驱动模块	225
8.4.5	编写内嵌的 S 函数设备驱动程序	232
8.4.6	创建 MEX 文件和设备驱动模块	236
8.5	访问参数和信号	236
8.5.1	通过模块的输出进行信号监视	237
8.5.2	通过 model_pt.c 对参数进行调整	241
8.5.3	与信号和参数有关的目标语言编译器 API	242
8.6	生成一个外部模式通信协议	242
8.6.1	外部模式设计	243
8.6.2	外部模式通信概述	243
8.6.3	外部模式源代码文件	245
8.6.4	实现通信传输协议层	246

第 II 部分 xPC 目标环境的应用

第 9 章	xPC 目标环境简介	249
9.1	什么是 xPC 目标	249
9.2	xPC 目标的特征	250
9.2.1	实时内核	250
9.2.2	实时应用程序	251
9.2.3	信号采集和分析功能	252
9.2.4	参数调节功能	252

9.3	硬件环境	253
9.3.1	宿主机	253
9.3.2	目标 PC 机	253
9.3.3	宿主机与目标机的通信连接	253
9.3.4	输入/输出设备驱动程序支持	254
9.4	软件环境	255
9.4.1	宿主机与目标机的通信	256
9.4.2	快速原型化过程	256
9.4.3	嵌入式过程	257
9.5	用户交互方式	258
9.5.1	xPC 目标图形用户界面	258
9.5.2	MATLAB 命令行界面	259
9.5.3	目标机命令行界面	259
9.5.4	Simulink 外部模式界面	260
9.5.5	Simulink 图形仪器仪表界面	260
9.5.6	Web 浏览器界面	260
第 10 章	xPC 目标的安装和测试	261
10.1	系统要求	261
10.1.1	宿主机	261
10.1.2	目标机	262
10.2	xPC 目标软件的安装	263
10.2.1	获取或更新许可协议	263
10.2.2	从光盘上安装	263
10.2.3	从网络下载安装程序并进行安装	264
10.2.4	宿主机上的文件介绍	264
10.2.5	设置初始工作路径	265
10.3	串口通信	265
10.3.1	串口通信的硬件	266
10.3.2	串口通信的环境属性	266
10.4	网络通信	267
10.4.1	网络通信的硬件	267
10.4.2	安装 PCI 总线类型网卡	268
10.4.3	安装 ISA 总线类型网卡	268
10.4.4	网络通信的环境属性	269
10.5	制作目标启动盘	270
10.6	安装测试及相关的问题	272

10.6.1	对安装过程进行测试	272
10.6.2	对目标系统的网络通信进行标准测试	273
10.6.3	对目标系统进行 xPC 目标的网络工作情况检测	274
10.6.4	采用直接的命令调用重启目标机	275
10.6.5	创建和下载目标应用程序	275
第 11 章	xPC 目标的基本应用	276
11.1	Simulink 模型的仿真	276
11.1.1	载入仿真程序	276
11.1.2	用 Simulink 图形化界面运行仿真	276
11.1.3	通过 MATLAB 命令行进行仿真	277
11.2	创建 xPC 目标应用程序	278
11.2.1	启动目标 PC 机	278
11.2.2	设置仿真参数	279
11.2.3	创建和下载目标应用程序	282
11.3	对目标程序进行控制	283
11.4	对 xPC 目标程序进行信号监视	284
11.5	对 xPC 目标程序进行信号记录	284
11.5.1	使用 xPC 目标图形界面进行信号记录	285
11.5.2	使用 MATLAB 命令进行信号记录	286
11.6	对 xPC 目标程序进行信号跟踪	288
11.6.1	使用 xPC 目标的图形用户界面进行信号跟踪	288
11.6.2	使用 xPC 目标的目标管理器进行信号跟踪	291
11.6.3	使用 MATLAB 命令进行信号跟踪	294
11.7	对 xPC 目标程序进行参数调整	297
11.7.1	使用 MATLAB 命令进行参数调节	297
11.7.2	使用 Simulink 外部模式调节参数	298
第 12 章	xPC 目标的高级应用	301
12.1	使用 xPC 目标的 I/O 设备驱动模块	301
12.1.1	xPC 目标 I/O 设备驱动模块	301
12.1.2	将 I/O 设备驱动模块添加到模型中	301
12.1.3	定义 I/O 设备驱动模块的参数	305
12.2	使用 xPC Target Scope 模块	306
12.2.1	xPC Target Scope 模块	306
12.2.2	将 xPC Target Scope 模块添加模型中	306
12.2.3	定义 xPC Target Scope 模块参数	308

12.3	目标机命令行界面.....	309
12.3.1	使用 xPC 目标对象的方法和属性.....	309
12.3.2	xPC 目标对象的方法和属性.....	310
12.3.3	示波器对象方法和属性.....	311
12.3.4	在目标机上使用变量及相应的命令.....	312
12.4	使用 xPC 目标的 Web 交互界面.....	313
12.4.1	与 Web 界面进行连接.....	313
12.4.2	Web 界面主页的使用.....	313
12.4.3	改变 WWW 属性.....	314
12.4.4	使用 Web 浏览器观察信号.....	315
12.4.5	在 Web 浏览器中使用 Scope 模块.....	315
12.4.6	使用 Web 界面察看和改变参数.....	316
12.4.7	改变 Web 浏览器的访问级别.....	316
第 13 章	xPC 嵌入式代码选项模块.....	317
13.1	xPC 目标嵌入式选项.....	317
13.2	更新 xPC 目标环境.....	319
13.3	创建 DOS 系统启动盘.....	321
13.4	使用 DOS 载入器目标应用程序.....	321
13.4.1	生成 DOS 载入器的目标启动盘.....	321
13.4.2	创建 DOS 载入器的目标应用程序.....	322
13.5	创建单机目标应用程序.....	323
13.5.1	创建单机模式的目标应用程序.....	323
13.5.2	创建单机模式的目标启动盘.....	323
13.5.3	在单机模式下使用 xPC Target Scope 模块.....	324
第 14 章	xPC 目标在卫星姿态控制系统中的应用.....	326
14.1	卫星姿态控制系统的基本情况.....	326
14.1.1	卫星姿态控制系统基本构成及相关的数学模型.....	326
14.1.2	卫星姿态控制系统特性参数.....	329
14.1.3	姿态控制系统控制器设计.....	330
14.2	卫星姿态控制系统的数学仿真及分析.....	330
14.3	姿态控制半物理仿真系统的方案设计和技术实现.....	331
14.3.1	半实物仿真系统总体方案设计.....	331
14.3.2	卫星姿态控制实时仿真系统设计.....	333
14.3.3	半物理实时仿真系统.....	336

第III部分 dSPACE 实时系统平台的应用

第 15 章 dSPACE 实时仿真系统介绍	339
15.1 dSPACE 仿真系统简介	339
15.1.1 dSPACE 简介	339
15.1.2 dSPACE 实时仿真功能	340
15.1.3 基于 dSPACE 的控制系统开发步骤	342
15.2 dSPACE 软件产品介绍	343
15.2.1 代码生成及下载软件	343
15.2.2 测试软件	343
15.3 dSPACE 硬件产品介绍	345
15.3.1 智能化的单板系统	345
15.3.2 标准组件系统	346
15.4 dSPACE 的简单应用范例	347
第 16 章 RTI/RTI-MP 及其应用	351
16.1 RTI 和 RTI-MP 的使用	351
16.1.1 怎样调用 RTI 提供的模块	351
16.1.2 为程序创建过程选项设置默认值	352
16.2 接入 I/O 设备模块	354
16.2.1 dSPACE 系统 I/O 模块命名规范	354
16.2.2 PHS 总线地址和板卡号	355
16.2.3 加入 I/O 模块	355
16.2.4 数据类型及其选用	357
16.3 任务处理	357
16.3.1 任务处理规则	358
16.3.2 由定时器驱动的任务	361
16.3.3 由事件驱动的任务	365
16.3.4 改变任务的特性	369
16.3.5 溢出情况的处理	370
16.3.6 定义后台任务	371
16.4 模型的建立和下载	371
16.4.1 模型代码生成和下载的基本知识	371
16.4.2 规定程序创建过程的选项	372
16.4.3 生成实时代码及下载	373

16.4.4	重新生成用户的 C 代码	374
16.5	外部仿真	375
16.5.1	开始外部模式仿真	375
16.5.2	通过外部仿真下载参数	376
16.5.3	停止外部的仿真	376
第 17 章	dSPACE 综合实验和测试环境——ControlDesk 软件工具	377
17.1	ControlDesk 快速入门	377
17.1.1	相关的术语和文件类型	377
17.1.2	ControlDesk 窗口	378
17.1.3	实时应用程序的处理	381
17.1.4	实验文件(.CDX 文件)的操作	382
17.1.5	ControlDesk 应用入门	383
17.2	创建仪表面板	389
17.2.1	创建仪表面板工具	389
17.2.2	布置仪表	391
17.2.3	连接变量	396
17.2.4	保存/下载仪表面板	397
17.3	使用仪表面板	398
17.3.1	捕获数据	398
17.3.2	打印数据曲线	401
17.3.3	参考数据管理器	402
17.4	批量修改参数	403
17.4.1	参数编辑器基本构成	403
17.4.2	处理参数文件	405
17.4.3	访问平台参数	405
17.5	ControlDesk 中 Simulink 平台上的仿真	407
17.5.1	建立 Simulink 模型	408
17.5.2	设置 Simulink 属性	408
17.5.3	系统描述文件的建立	409
17.5.4	仪表面板的复用	409
17.5.5	分配参数文件至另一仿真	410
第 18 章	dSPACE 系统在卫星姿态确定系统中的应用	411
18.1	星敏感器的姿态确定系统数学仿真	411
18.1.1	星敏感器姿态确定系统概述	411
18.1.2	姿态确定系统设计	412

18.2 星敏感器姿态确定系统半物理仿真	416
18.2.1 半物理仿真系统	416
18.2.2 半物理仿真实验结果	418

第IV部分 附 录

附录 A 依赖于绝对时间的 Simulink 模块	421
附录 B RTW 术语表	423
附录 C xPC 目标环境	426
C.1 xPC 目标环境	426
C.1.1 xPC 目标环境属性	426
C.1.2 xPC 目标环境函数	428
C.2 使用 xPC 目标环境属性和函数	429
C.2.1 获得环境属性列表	429
C.2.2 存储和载入环境属性	429
C.2.3 通过图形界面改变环境属性	430
C.2.4 通过命令行改变环境属性	431
C.2.5 通过图形界面生成目标启动盘	431
C.2.6 通过命令行生成目标启动盘	431
C.3 系统函数	432
C.3.1 图形用户界面函数	432
C.3.2 测试函数	432
C.3.3 xPC 目标演示函数	432
C.4 环境和系统函数	433
附录 D 设备驱动程序源代码	439

第 I 部分 RTW代码自动生成和实时系统开发环境

第1章 RTW 基 础

作为一种功能强大的数学计算软件，MATLAB 具有数值计算、数据可视化功能和易于使用的编程环境，典型的应用包括工程计算、算法开发、建模和仿真、数据分析和可视化应用程序开发等。除此以外，MATLAB 还提供了一个实时开发环境，可用于实时系统仿真和产品的快速原型化，这一点是通过特殊应用工具箱——Real-Time Workshop(实时工作间，RTW)实现的。

RTW 是 MathWorks 公司提供的 MATLAB 工具箱之一，是 Simulink、Stateflow、DSP Blockset 和通信工具箱的一个补充功能模块，可用于各种类型的实时应用。通过本章的学习，用户可以了解到 RTW 的功能和特点、RTW 的安装，以及 RTW 支持的编译器。

1.1 RTW 简 介

1.1.1 RTW 的概念

RTW 是 MATLAB 图形建模和仿真环境 Simulink 的一个重要的补充功能模块，简而言之，它是一个基于 Simulink 的代码自动生成环境。它能直接从 Simulink 的模型中产生优化的、可移植的和个性化的代码，并根据目标配置自动生成多种环境下的程序。利用它可加速仿真过程，提供知识产权保护，或生成可在不同的快速原型化实时环境或产品目标下运行的程序。其特点如下：

- RTW 支持连续时间、离散时间和混合时间系统，包括条件执行型系统和非虚拟型系统。
- RTW 将 Simulink 外部模式的运行时监视器(Run-Time Monitor)与实时目标无缝集成在一起，提供了极好的信号监视和参数调整界面。
- RTW 支持 Ada 代码的自动生成。
- RTW 支持 Stateflow 代码生成器，用来生成事件驱动型系统的有限状态机代码。