

基于模型的设计 ——MCU篇

刘 杰 翁公羽 周宇博/著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

基于模型的设计 ——MCU 篇

刘 杰 翁公羽 周宇博 著

北京航空航天大学出版社

前 言

20 世纪 90 年代初,航空航天、汽车等行业中开始大量使用微控制器单元,开发人员首先发现通过建模与仿真,可以大大提高 MCU 系统的开发效率。到了 20 世纪 90 年代中期,出现了代码自动生成技术,使基于模型设计的雏形随之显现。

近十几年来,信息技术飞速发展,产品中软件控制代码呈爆炸性增长的趋势。一台中高档汽车的控制代码超过 500 万行,第四代战机 F-35 的软件代码则高达 1500 万行,它们都超过了人类第一次登月——阿波罗计划中软件代码的规模。随着代码量的迅速膨胀,传统的手工编程模式面临着产品开发周期被迫拉长,开发成本成倍增加,同时产品可靠性也难以保障等诸多难以克服的困难,已很难适应当今科技发展的需要,严重制约了我国现代化的进展。目前,世界上已有数家具有前瞻性的软件公司推出了自己的基于模型设计软件,如 SCADE、dSpace、MATLAB 等,它们是世界科技创新的技术源泉。空客 A380、美国通用公司的混合动力汽车、我国二汽新能源汽车的电池管理系统、喷气发动机、洛克-马丁公司的联合攻击机 F-35 等重大创新项目,都采用了基于模型的设计。

作者自 2000 年开始关注自动代码生成技术,以及后来的基于模型设计,尤其是最近三年来,夜以继日地钻研基于模型设计的思想,阅读浏览了超过 10 万页的外文资料与技术文档,做了大量基于模型设计的实验,撰写了国内第一本基于模型设计的专著——《基于模型的设计及其嵌入式实现》。也许是外国政府或国际知名企业对这项新技术有所保留,网上很难找到有价值的开发实例,美国 CRC 出版社 2009 年 11 月出版的《Model-Based Design for Embedded Systems》一书,也仅是作了知识性的介绍。

通过这几年作者及实验室全体同仁的共同努力,基本掌握了基于模型的设计方法,并将此技术推向了实用阶段。现将近年的研究成果整理成册,以期能为我国的科技创新,建立创新型国家贡献微薄之力。

考虑到国内有众多的 MATLAB 用户,本书以 MATLAB R2010b 为软件平台介绍基于模型设计的方法。书中涉及 8051 单片机、英飞凌 C166 单片机、Microchip dsPIC 数字信号控制器、ARM 处理器的基于模型设计的开发,其优势在于:在统一的可视化开发测试平台上,使用户轻松地设计概念到实现一气呵成,减少不必要的重复劳动,这将大大缩短项目的开发周期、减少资金投入、提高代码的稳健性与一致性。一些对安全性要求不高的产品几周就可以完成,有效地规避了潜在的市场风险,使企业在激烈的市场竞争中占得先机;满足 DO-178b 航空电子规范的工作流程,使基于模型设计能提供解决那些对于安全性、可靠性要求极高甚至近乎苛刻的设计要求的绝好方案。

全书可分三个部分:1~4 章为第一部分,介绍模型的创建与调试,是基于模型设计的起点与核心;5~8 章为第二部分,介绍基于模型的 8051 单片机、英飞凌 C166 单片机、Microchip dsPIC3x 数字信号控制器、ARM 处理器的快速开发;第 9 章为第三部分,以直流电动机的 PID

控制为例,介绍基于模型设计在 ARM 处理器上实现的全过程。

本书的核心内容由作者实验室自主创作,这些内容多是首次在国内或国外的公开出版物中出现。除三位作者外,还有以下人员也参加了本书个别章节的撰写、资料整理和全书的策划工作,他们是:刘高阳、杨元廷、史进、陈天丁、李晗、郑明奎、万方、吴义炳、李昌、孙瑶瑶、郑仁富、郭丹、陈丽红、曹国忠、冯新凯、田利平、李泳毅、吴玉林、刘绍丽、姜慧、施文龙、万文青、张颖、周勋、于福海、李阳、朱宁、王敬卿、魏莲花、朱伟龙、胡泽东、许家胜、翟朝亮、胡育涛、李燕。由于涉及的 MCU 器件较多,无法在 500 页的图书中进行全面阐述,加上时间紧且作者水平有限,书中的错误或遗漏在所难免,欢迎读者批评指正,提出宝贵意见,作者的 email:1158867891@qq.com。

刘杰

2010 年 10 月于怡园

目 录

第 1 章 MATLAB 编程基础	1
1.1 MATLAB R2010a 与 2010b 的若干更新	1
1.1.1 压缩文件	2
1.1.2 目录浏览器	3
1.1.3 文件夹及文件比较	5
1.1.4 登录 MATLAB 文件交换服务器	7
1.2 M 文件的编写	8
1.2.1 M 文件结构	8
1.2.2 M 脚本文件	10
1.2.3 快捷方式	12
1.2.4 M 函数	13
1.2.5 匿名函数	17
1.2.6 函数提示	17
1.3 M 文件的调试	18
1.3.1 M-Lint	18
1.3.2 使用 cells 加快调试	19
1.4 M 文件的发布	21
1.5 Embedded MATLAB	24
1.5.1 Embedded MATLAB 的主要功能特点	24
1.5.2 Embedded MATLAB 的编程规范	25
1.5.3 C 编译器的设置	25
1.5.4 Embedded MATLAB 编程实例	27
第 2 章 Simulink 建模与调试	36
2.1 Simulink 基本操作	37
2.1.1 模块库和编辑窗口	37
2.1.2 Simulink 模块库	38
2.1.3 模块的基本操作	43
2.2 搭建直流电动机模型	46
2.2.1 数学模型分析	46
2.2.2 模型搭建与参数设置	48
2.2.3 子系统与库	56
2.2.4 添加模块到库浏览器及知识产权保护	61

2.2.5	数据格式与输入/输出	63
2.2.6	PID 控制	67
2.3	Simulink 模型调试	77
2.3.1	图形界面调试	77
2.3.2	命令行调试	79
2.3.3	运行调试器	80
2.3.4	断点设置	84
2.3.5	显示模型和仿真信息	86
第 3 章	Stateflow 建模与应用	92
3.1	Stateflow 基本概念	92
3.1.1	状态图编辑器	94
3.1.2	状 态	95
3.1.3	迁 移	99
3.1.4	数据与事件	101
3.1.5	对象的命名规则	101
3.2	Stateflow 状态图	102
3.2.1	状 态	102
3.2.2	迁 移	103
3.2.3	计时器状态图	106
3.2.4	数据与事件	107
3.2.5	动 作	109
3.2.6	自动创建对象	111
3.3	Stateflow 流程图	112
3.3.1	流程图与节点	112
3.3.2	建立流程图	113
3.4	层次结构	116
3.4.1	层次的概念	116
3.4.2	迁移的层次	117
3.4.3	历史节点	118
3.4.4	子状态图	119
3.4.5	层次状态图中的流程图	120
3.5	并行机制	120
3.5.1	设置状态关系	120
3.5.2	并行状态活动顺序配置	121
3.5.3	本地事件广播	122
3.5.4	直接事件广播	123
3.5.5	隐含事件和条件	124
3.6	Stateflow 其他对象	125
3.6.1	真值表(Truth table)	125

3.6.2	图形函数(Graphical function)	127
3.6.3	Embedded MATLAB	129
3.6.4	图形盒(Box)	131
3.6.5	Simulink 函数调用	132
3.6.6	目 标	134
3.7	综合应用	137
3.7.1	计时器	137
3.7.2	交通灯	144
第4章	设备驱动模块的编写	152
4.1	创建 S 函数模块的示例	153
4.1.1	手工编写 Wrapper S 函数	153
4.1.2	代码继承工具(Legacy Code Tool)	157
4.1.3	S-Function Builder	160
4.1.4	三种方法的比较	162
4.2	S 函数	164
4.2.1	S 函数工作机制	164
4.2.2	C MEX S 函数模板	166
4.2.3	其他回调方法	173
4.2.4	宏函数	178
4.2.5	数据访问	179
4.2.6	目标语言编译器	181
4.3	S-Function Builder	184
4.3.1	S-Function Builder 简介	184
4.3.2	初始化界面(initialization)	186
4.3.4	数据属性界面(Data Properties)	187
4.3.5	库文件界面(Libraries)	188
4.3.6	输出界面(Outputs)	189
4.3.7	连续状态求导(Continuous Derivatives)	191
4.3.8	离散状态更新(Discrete Update)	192
4.3.9	编译信息(Build Info)	193
4.4	创建设备驱动实例	194
4.4.1	HC12 模数转换模块	194
4.4.2	DAS1600 数据输入模块	210
4.4.3	S-Function Builder	218
第5章	8051 单片机代码的快速生成	223
5.1	仿真软件 Proteus 快速入门	223
5.1.1	Proteus 简介	223
5.1.2	快速绘制原理图	225
5.1.3	PCB 制板	232

5.2	Keil C51 集成开发环境(IDE)	235
5.2.1	预备知识	235
5.2.2	RTW-EC 快速代码生成	245
5.2.3	脉宽调制	250
5.2.4	流水灯	259
5.3	TASKING 嵌入式开发环境(EDE)	269
5.3.1	预备知识	269
5.3.2	直流电机控制	276
5.3.3	算术乘法	286
5.3.4	流水灯	293
第 6 章	C166 代码的快速生成	296
6.1	英飞凌 C166 模块库简介	297
6.2	TASKING EDE for C166	299
6.2.1	电动机控制模型	299
6.2.2	设置 IDE 与模型参数	300
6.2.3	处理器在环测试(PIL)	302
6.2.4	代码的自动生成	305
第 7 章	基于 Simulink 模块的 dsPIC 单片机开发	309
7.1	MPLAB 嵌入式开发环境及工具	310
7.1.1	软件的下载和安装	310
7.1.2	利用 MPLAB IDE 及 Proteus VSM 进行虚拟硬件调试	315
7.1.3	dsPIC 外围驱动模块简介	323
7.2	dsPIC 外围驱动模块应用	324
7.2.1	数模转换实验	324
7.2.2	闪烁灯	333
7.2.3	调用现有 C 函数	342
7.3	无对应模块时的应用	355
7.3.1	创建功能验证模型	355
7.3.2	自动代码生成	355
7.3.3	虚拟硬件测试	360
第 8 章	ARM 代码的快速生成	361
8.1	ARM 简介	361
8.2	蜂鸣器	363
8.2.1	蜂鸣器发声模型	363
8.2.2	蜂鸣器功能验证模型	364
8.2.3	软件在环测试	365
8.2.4	自动代码生成	366
8.2.5	虚拟硬件测试	370
8.3	交通灯控制	371

8.3.1	软件在环测试	371
8.3.2	自动代码生成及编译	375
8.3.3	虚拟硬件测试	379
8.4	步进电动机控制	382
8.4.1	步进电动机原理简介	382
8.4.2	步进电动机控制模型	382
8.4.3	步进电动机的功能验证模型	383
8.4.4	软件在环测试	385
8.4.5	自动代码生成	387
8.4.6	虚拟硬件测试	391
8.5	无刷电动机的控制	393
8.5.1	无刷电动机原理简介	393
8.5.2	TASKING IDE FOR ARM	394
8.5.3	无刷电动机控制模型	395
8.5.4	无刷电动机功能验证模型	397
8.5.5	软件在环测试	398
8.5.6	编写驱动代码	401
8.5.7	自动代码生成	401
8.5.8	代码效率比较	407
8.5.9	虚拟硬件测试	417
第9章	基于模型的设计	421
9.1	传统设计的弊端	422
9.2	基于模型设计的优势	423
9.3	基于模型设计的流程	425
9.3.1	建立需求文档	425
9.3.2	建立可执行的技术规范	425
9.3.3	浮点模型	426
9.3.4	需求与模型间的双向跟踪	426
9.3.5	Model Advisor 检查	426
9.3.6	模型验证	426
9.3.7	定点模型	427
9.3.8	软件在环测试(SIL)	427
9.3.9	处理器在环测试(PIL)	427
9.3.10	代码与模型间的双向跟踪	427
9.3.11	代码优化	428
9.3.12	生成产品级代码	428
9.4	需求分析及跟踪	428
9.4.1	系统模型	428
9.4.2	需求关联	431

9.4.3 一致性检查	433
9.5 模型检查及验证	435
9.5.1 System Test	435
9.5.2 Design Verifier	443
9.5.3 Model Advisor 检查	454
9.6 定点模型	458
9.6.1 Fixed Point Advisor	459
9.6.2 Fixed Point Tools	465
9.7 软件在环测试	469
9.8 代码跟踪	470
9.9 代码优化及代码生成	473
9.9.1 子系统原子化	473
9.9.2 确定芯片类型	474
9.9.3 代码检查	475
9.9.4 代码生成	478
9.10 虚拟硬件测试	479
附录 Embedded MATLAB 支持的各函数	486
参考文献	500

第 1 章

MATLAB 编程基础

MATLAB 是一种高度集成化的交互式编程环境,大到航空航天,小到计算二元矩阵的逆矩阵,都可以看到 MATLAB 的身影,其优势在于算法研究、数据分析与可视化、并行计算等方面。随着信息技术和计算机硬件的不断发展, MATLAB 的应用领域得到了广泛的拓展,主要包括电机控制、飞行建模、音视频处理、通讯、测试与测量、财务建模与分析等。

MATLAB 是基于模型设计的起点,本章仅对新版 MATLAB 软件的一些特色功能、MATLAB 程序的编写及调试方法、Embedded MATLAB 的编程规范作简单介绍,更详细的内容请读者参考 MathWorks 公司的相关技术手册及各种著作。

本章的主要内容如下:

- MATLAB 开发环境新功能。
- M 文件编写、调试与发布。
- Embedded MATLAB。

1.1 MATLAB R2010a 与 2010b 的若干更新

MATLAB R2010a 与 R2010b 针对 MATLAB 和 Simulink 新增加了若干功能,并对其他多款产品进行了更新和缺陷修复,本章首先列出 MATLAB 新增加及加强的功能。

1. 开发环境 (MATLAB 7.10)

- 新增了解压缩功能,能够自动压缩和解压当前文件夹的文件和文件夹。
- 提示“当前文件夹”是否包含在 MATLAB 搜索路径列表。
- MATLAB 变量编辑器中的表格填写功能,可支持局部变量、子函数和嵌套函数。
- 扩展了曲线拟合、滤波器设计、图像处理与信号处理工具箱绘图选择界面的图形访问能力。
- 使用比较工具比较文件时,可高亮显示各行的变化;比较文件夹时,可按名称、类型、大小或时间排列比较结果。

2. 开发环境(MATLAB 7.11)

- 编辑器可高亮显示变量或子函数的所有使用情况,并识别共享变量。
- 可以将 ZIP 文件看作文件夹进行操作。
- 预览当前文件夹的图像文件内容,并以星号提示 MATLAB 文件是否保存。
- 在绘图选择界面中可以访问系统标识、制图和生物信息学工具箱的图形。
- 使用比较工具能够比较 ZIP 文件、文件夹和 Simulink 表单,并改善了 MAT 文件比较性能。

3. 语言和编程(MATLAB 7.11)

- 可使用若干组已命名的值,自行定义枚举数据类型。

4. 数学(MATLAB 7.11)

- 支持 64 位整型数据的算法。

5. 文件 I/O 和外部接口连接(MATLAB 7.11)

- 新增了 VideoWriter 对象,可创建大于 2GB 的 MPEG 和非压缩 AVI 文件。
- 支持 netCDF0.1,可将 HDF5 用作 netCDF API 的数据存储层。
- 加强了与 Microsoft .NET framework 的接口,支持与 Microsoft Office 产品的授权和交互。

根据本书的定位,以下选取几个开发环境典型的新功能,作简要说明。

1.1.1 压缩文件

为了备份文件、节省存储空间、与他人共享,通常是使用第三方压缩程序将文件打包压缩,而 MATLAB 7.10 集成了压缩功能。在目录浏览器可以直接建立或解压 ZIP 格式的压缩文件。MATLAB 7.11 进而又可将 ZIP 文件看作文件夹进行操作。

1. 创建压缩文件

选中需要打包的文件,选择右键菜单项 Creat Zip File,如图 1.1.1 所示。在当前目录建立以 Untitled1 为名的 ZIP 压缩文件,如图 1.1.2 所示。

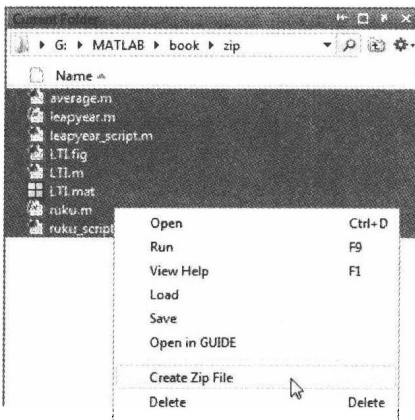


图 1.1.1 创建压缩文件

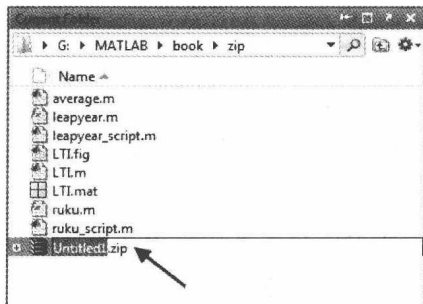


图 1.1.2 命名压缩文件

2. 压缩文件管理

单击压缩包左侧的十号,将其看作一个文件夹展开显示,如图 1.1.3 所示。双击压缩包内的各文件,可执行对应的操作,例如打开 M 文件,加载 MAT 数据等。不过这些文件是只读的,用户修改后需要将其另存。

3. 解压缩文件

双击压缩文件、选择压缩文件右键菜单项 Extract、或选中文件后按 Enter 键,在当前目录解压得到同名文件夹,如图 1.1.4 所示。当然,如果压缩包里包含与当前目录下同名的文件或文件夹,在解压时系统会提示用户是否替换当前文件。

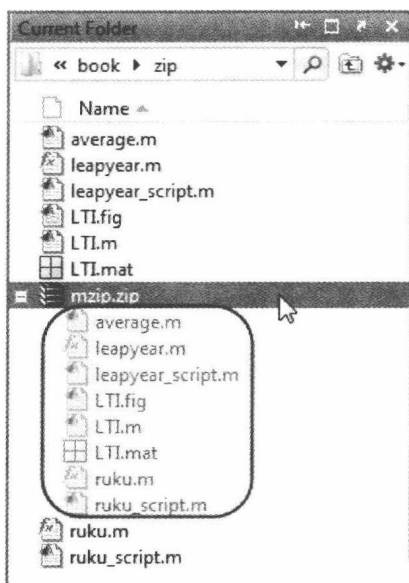


图 1.1.3 查看压缩文件

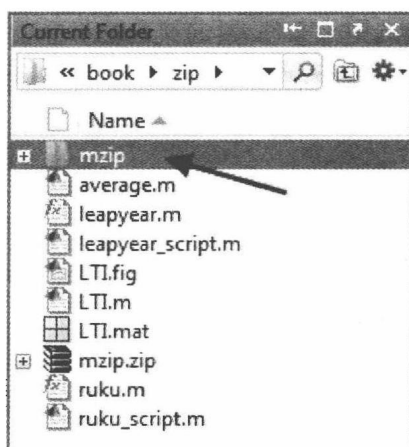


图 1.1.4 解压缩文件

1.1.2 目录浏览器

对于大型的工程,脚本、函数、数据等文件可能处于不同的文件夹,如果 a 目录下的脚本文件需要调用 b 目录下的函数与 c 目录下的数据,用户可以将 b、c 目录添加到 MATLAB 搜索路径,这类类似于 C 语言的头文件。

单击 MATLAB 主窗口菜单项 File→Set Path...,打开搜索路径对话框,用户可使用左侧的按钮增加或删除搜索路径并调整路径顺序,如图 1.1.5 所示。

使用早期版本的用户可能感觉到,访问位于不同文件夹下的文件是个很麻烦的过程,需要频繁地切换目录。自 MATLAB 7.9 起,目录浏览器改进了显示方式,使用类似于 Windows 资源管理器的树形结构,用户可同时查看多个目录下的文件。MATLAB 7.10 将搜索路径体现在目录浏览器,淡化显示未加入搜索路径的文件夹及下属文件。

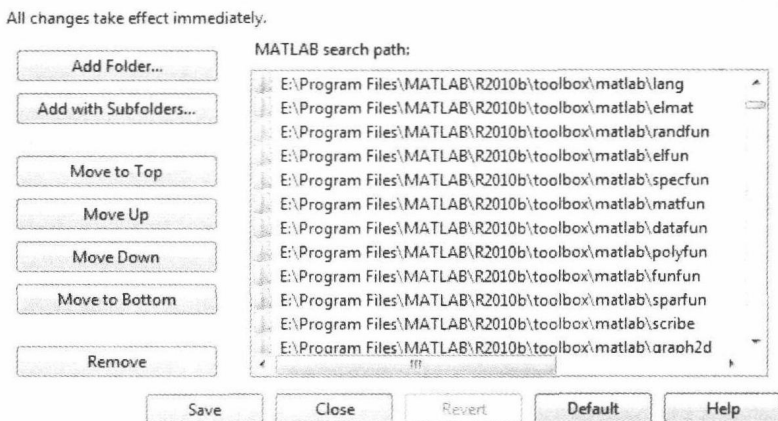


图 1.1.5 MATLAB 搜索路径

用户选中某文件夹,选择右键菜单项 Add to Path(或 Remove from Path),将当前文件夹(或及其子文件夹)加入搜索路径(或删除),如图 1.1.6 所示。

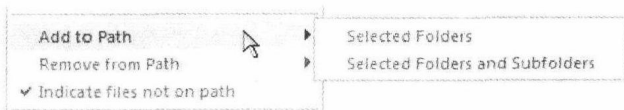


图 1.1.6 添加搜索路径

默认情况下,加入搜索路径的文件夹以正常的亮度显示,未加入搜索路径的文件夹则淡化显示,如图 1.1.7 所示。这样用户可明确地分辨当前目录下哪些文件夹及文件是可访问的,从而简化了搜索路径的设置。



图 1.1.7 文件夹显示

单击 MATLAB 主窗口菜单项 File→Preferences..., 选择 Current Folder 面板, 其中复选项 Indicate inaccessible files 的功能与图 1.1.6 所示的右键菜单项 Indicate files not on path 一致, 可开启或关闭淡化显示功能。拖动 Text and icon transparency 滑块, 调整淡化显示的透明程度, 如图 1.1.8 所示。

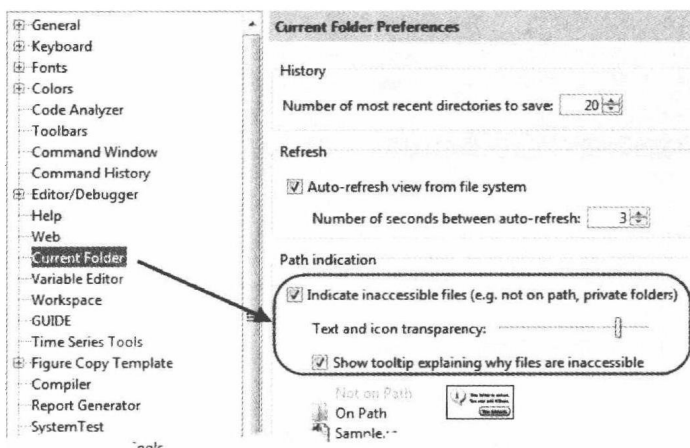



图 1.1.8 调整透明度

1.1.3 文件夹及文件比较

用户可以使用 MATLAB 文件比较工具, 比较两个文件夹下属文件的类型、文件名、大小、修改时间, 以及两个具体的文本文件、MAT 文件、二进制文件之间的差异。

绿色、红色、无色、蓝色分别表示右侧文件夹与左侧文件夹的比较结果: 增加、修改、一致、删除。单击工具栏的按钮 , 可以调换两个文件夹位置, 如图 1.1.9 所示。

		In left list (folder mprogram)		In right list (folder mscrip)		Change Summary
Type	File Name	Size (bytes)	Last Modified Date	Size (bytes)	Last Modified Date	
FIG file	raku.fig (open)		that in this list!	9487	2010-04-07 23:36:37	added
M file	LTI.m (open: left right)	590	2010-04-15 01:27:51	603	2010-04-12 02:22:21	contents changed (compare)
M file	leapyear_script.m (open: left right)	43	2010-04-14 21:29:41	61	2010-04-07 23:52:51	contents changed (compare)
M file	raku_script.m (open: left right)	50	2010-04-01 01:37:03	50	2010-04-01 01:37:03	identical
M file	leapyear.m (open)	720	2010-04-14 21:24:41	that in this list!		removed
M file	average.m (open)	193	2010-04-14 21:21:44	that in this list!		removed
M file	raku.m (open)	852	2010-04-12 03:21:49	that in this list!		removed
MAT file	LTI.mat (open)	42608	2010-04-12 00:27:52	that in this list!		removed
FIG file	LTI.fig (open)	6595	2010-03-29 01:10:17	that in this list!		removed

图 1.1.9 比较结果

单击 Type、File Name、Size、Last Modified Date、Change Summary 等栏目名,可调整表格的排序方式(如图 1.1.9 按比较结果排序)。

单击链接 left 或 right,可以打开左侧或右侧文件夹的对应文件,若同时打开了两个文件,用户还可以将 MATLAB 编辑器窗口调整为纵向并列,具体分析文件的差异如图 1.1.10 所示。

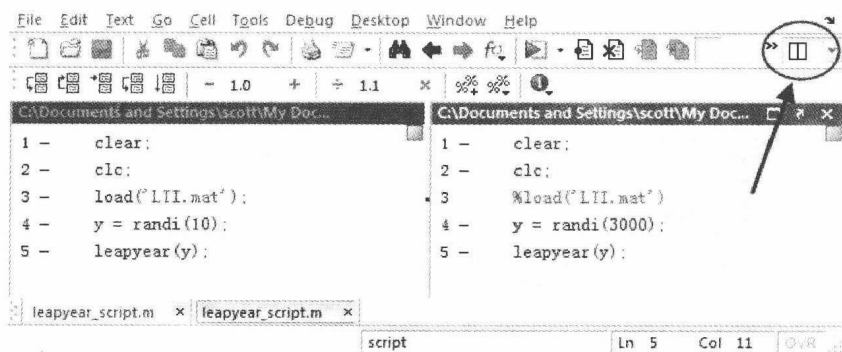


图 1.1.10 并列 MATLAB 编辑器窗口

显然人工比较容易出错,用户可继续单击图 1.1.9 表格最后一列的链接 compare,自动比较文件差异,如图 1.1.11 所示。

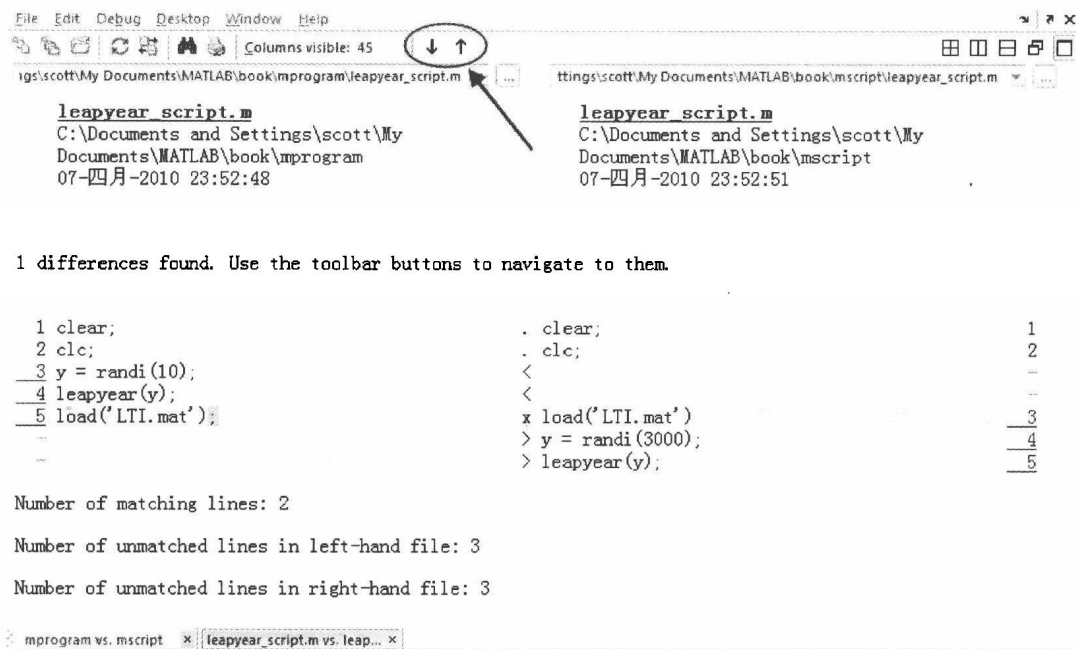




图 1.1.11 自动比较文件差异

MATLAB 7.10 的文件比较器新增了高亮功能,两个文件之间存在差异的行将被高亮显示,同时在工具栏新增了按钮  与 ,便于逐一比较各个差异。

1.1.4 登录 MATLAB 文件交换服务器

在 MATLAB 环境中,用户可轻松地登录 MATLAB File Exchange 服务器,享受 MATLAB 文件交换服务。即用户可以使用由 MathWorks 公司或世界各地用户上传至 MathWorks 服务器的各种 MATLAB 程序、Simulink 模型、演示视频等。合理有效地利用这些文件,可节省用户的学习与开发时间、启发思路、扩展视野。

使用该项服务是免费的,不过用户需要拥有 MathWorks 账户并能够访问互联网,当然建立账户也是免费的。

单击 MATLAB 主窗口菜单项 Desktop → File Exchange,打开登录界面,如图 1.1.12 所示。

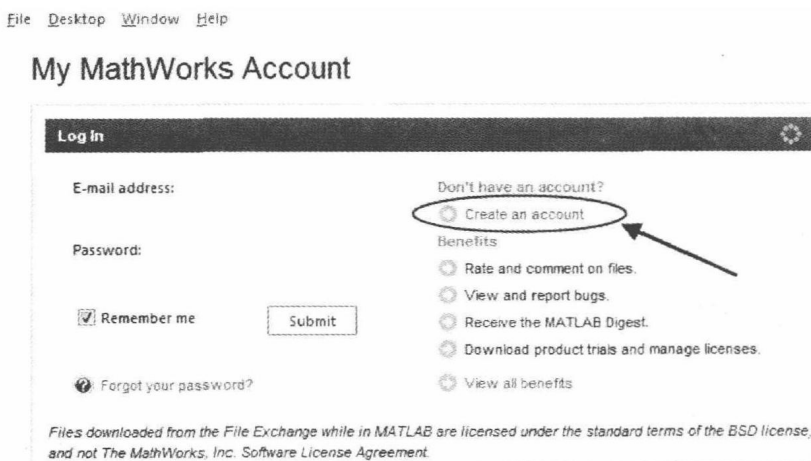


图 1.1.12 用户登录

用户可以单击链接 [Creat an account](#) 创建一个 Mathworks 账户,或使用已创建的账户登录 MATLAB File Exchange 服务器,如图 1.1.13 所示。

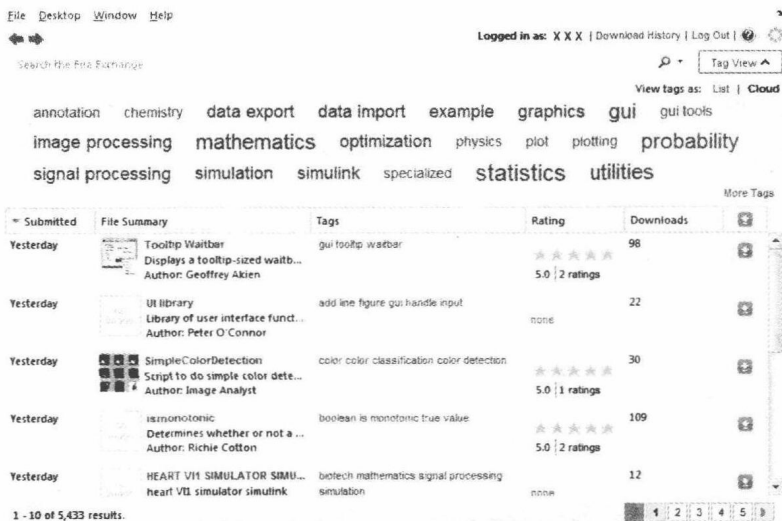


图 1.1.13 MATLAB File Exchange 主页