

iOS核心开发手册（原书第5版）



[iOS核心开发手册（原书第5版） 下载链接1](#)

著者:Erica Sadun

出版者:机械工业出版社

出版时间:2015-4

装帧:平装

isbn:9787111491859

世界知名iOS开发专家详细讲解创建优秀iOS移动应用程序的技术细节，系统阐释如何高效开发iOS程序。

用直观的代码演示当前流行的编程技巧，为iOS开发者提供之有效的解决方案。

作者介绍:

目录: 译者序

前言

致谢

第1章 手势与触摸 1

1.1 触摸 1

1.1.1 触摸操作所处的阶段 2

1.1.2 UIResponder类中的触摸事件响应方法 3

1.1.3 对视图的触摸 4

1.1.4 多点触摸 4

1.1.5 手势识别器 5

1.2 解决方案: 添加简单的直接操纵界面 5

1.3 解决方案: 添加拖动手势识别器 7

1.4 解决方案: 同时使用多个手势识别器 9

1.5 解决方案: 限制移动 14

1.6 解决方案: 测试触摸 15

1.7 解决方案: 针对位图的触摸测试 17

1.8 解决方案: 根据触摸情况在屏幕上绘制内容 19

1.9 解决方案: 令绘制效果变得平滑 21

1.10 解决方案: 启用多点触摸 24

1.11 解决方案: 检测圆圈手势 27

1.12 解决方案: 创建自定义手势识别器 32

1.13 解决方案: 把滚动视图中的内容拖曳到外面 34

1.14 解决方案: 实时的触摸反馈 37

1.14.1 启用触摸反馈效果 38

1.14.2 拦截并转发触摸事件 38

1.14.3 实现TOUCHkit的TOUCHkitView类 40

1.15 解决方案: 向视图添加菜单 42

1.16 小结 43

第2章 构建并使用控件 45

2.1 UIControl类 45

2.1.1 目标-动作模式 46

2.1.2 控件的种类 46

2.1.3 控件事件 46

2.2 按钮 48

2.3 Interface Builder中的按钮 50

2.4 解决方案: 构建按钮 51

2.4.1 多行按钮文本 54

2.4.2 为按钮添加动画元件 54

2.4.3 为按钮添加额外状态 55

2.5 解决方案: 使按钮以动画效果来响应用户 55

2.6 解决方案: 为滑杆控件添加自定义的滑块 57

2.6.1 定制UISlider控件 57

2.6.2 添加优化代码 58

2.7 解决方案: 创建可以连续点击两次的分段选择控件 61

2.7.1 实现第二次点击时的反馈效果 62

2.7.2 控件及带属性的字符串 63

2.8 开关控件与步进控件 64

2.9 解决方案: 编写UIControl的子类 65

- 2.9.1 创建控件 68
- 2.9.2 追踪触摸事件 68
- 2.9.3 派发控件事件 69
- 2.10 解决方案：构建评分所用的Star Slider控件 69
- 2.11 解决方案：构建触摸转盘控件 72
- 2.12 解决方案：创建拉曳控件 75
 - 2.12.1 为控件添加提示效果 75
 - 2.12.2 测试触摸 77
- 2.13 解决方案：构建自定义的锁定控件 80
- 2.14 解决方案：图片库查看器 83
- 2.15 构建工具栏 85
- 2.16 小结 88
- 第3章 提醒用户 89
 - 3.1 直接向用户弹出警告视图 89
 - 3.1.1 构建简单的警告视图 89
 - 3.1.2 设置UIAlertView的委托 91
 - 3.1.3 显示UIAlertView 92
 - 3.1.4 各种UIAlertView 92
 - 3.2 解决方案：构建支持块的警告视图 93
 - 3.2.1 块简介 93
 - 3.2.2 使用块时避免保留循环 95
 - 3.3 解决方案：将变长参数列表与UIAlertView结合起来使用 98
 - 3.4 展示选项列表 99
 - 3.4.1 滚动菜单 101
 - 3.4.2 在动作表中显示文本 101
 - 3.5 将操作进度告知用户并提示其稍等片刻 102
 - 3.5.1 使用UIActivity-IndicatorView 103
 - 3.5.2 使用UIProgressView 103
 - 3.6 解决方案：在屏幕上绘制模态的进度指示器 104
 - 3.7 解决方案：自制的模态警告视图 106
 - 3.8 解决方案：基本的popover 110
 - 3.9 解决方案：本机通知 111
 - 3.10 用网络活动指示器提醒用户 113
 - 3.11 解决方案：播放简单的提示音 114
 - 3.11.1 System Sound 114
 - 3.11.2 为使用系统框架而引入模块 115
 - 3.11.3 震动 115
 - 3.11.4 警示音 116
 - 3.11.5 延迟 117
 - 3.11.6 释放系统音 117
 - 3.12 小结 117
- 第4章 编排视图及其动画效果 119
 - 4.1 视图层级 119
 - 4.2 解决方案：用树状图来描述视图层级 121
 - 4.3 解决方案：查询子视图 123
 - 4.4 管理子视图 125
 - 4.4.1 添加子视图 125
 - 4.4.2 重排及删除子视图 125
 - 4.4.3 UIView的回调方法 125
 - 4.5 为视图设定标签并查找视图 126
 - 4.6 解决方案：通过对象关联机制为视图设定名称 127
 - 4.7 视图的几何特征 129
 - 4.7.1 框架 130
 - 4.7.2 与CGRect有关的工具函数 130

- 4.7.3 CGPoint与CGSize 131
- 4.7.4 CGAffineTransform 132
- 4.7.5 坐标系统 133
- 4.8 解决方案：操控视图的框架 133
 - 4.8.1 调整视图的尺寸 134
 - 4.8.2 CGRect与中心点 136
 - 4.8.3 视图的其他几何特征 137
- 4.9 解决方案：获取与坐标变换有关的信息 141
 - 4.9.1 获取与变换有关的属性 141
 - 4.9.2 判断两个视图是否相交 142
- 4.10 与显示和交互有关的特征 147
- 4.11 UIView的动画效果 148
- 4.12 解决方案：视图的淡入与淡出 150
- 4.13 解决方案：交换两个视图的前后顺序 151
- 4.14 解决方案：翻转视图 151
- 4.15 解决方案：采用Core Animation API来制作切换效果 153
- 4.16 解决方案：使视图在出现之后回弹 155
- 4.17 解决方案：关键帧动画 156
- 4.18 解决方案：UIImageView的动画效果 157
- 4.19 小结 158

第5章 视图的约束系统 160

- 5.1 什么是约束 161
- 5.2 约束系统所用的属性 161
- 5.3 约束系统的运作规律 163
- 5.4 约束规则与框架属性 165
 - 5.4.1 固有内容的尺寸 165
 - 5.4.2 对齐矩形 166
- 5.5 创建约束规则 167
 - 5.5.1 基本约束规则声明 167
 - 5.5.2 用可视化格式字符串声明约束规则 168
 - 5.5.3 变量绑定 169
- 5.6 格式字符串 169
 - 5.6.1 方向 169
 - 5.6.2 连接 171
- 5.7 谓词 173
 - 5.7.1 指标 173
 - 5.7.2 描述两个视图关系的谓词 174
 - 5.7.3 优先级 174
- 5.8 格式字符串总结 174
- 5.9 用格式字符串将视图对齐并灵活调整其尺寸 176
- 5.10 处理约束规则的流程 176
- 5.11 管理约束规则 177
- 5.12 解决方案：实现约束规则之间的对比 178
- 5.13 解决方案：创建尺寸固定且受规则约束的视图 181
 - 5.13.1 禁用translatesAutoresizing-MaskIntoConstraints 181
 - 5.13.2 令视图出现在上级视图范围内 182
 - 5.13.3 限定视图的尺寸 183
 - 5.13.4 把前面各节内容拼装起来 183
- 5.14 解决方案：将两个视图居中对齐 185
- 5.15 解决方案：设定宽高比 186
- 5.16 解决方案：响应屏幕方向的变更 188
- 5.17 调试约束规则 190
- 5.18 解决方案：描述约束规则 191
- 5.19 用宏来创建约束规则 194

- 5.20 小结 197
- 第6章 文本输入 198
 - 6.1 解决方案：隐藏UITextField的键盘 199
 - 6.1.1 阻止系统把键盘隐藏起来 200
 - 6.1.2 UITextFieldInputTraits协议中的属性 200
 - 6.1.3 文本框的其他属性 201
 - 6.2 解决方案：把带有自定义辅助视图的键盘隐藏起来 203
 - 6.3 解决方案：根据键盘来调整文本视图 205
 - 6.4 解决方案：创建自定义的输入视图 209
 - 6.5 解决方案：使视图具备文本输入功能 213
 - 6.6 解决方案：为非文本视图添加自定义的输入视图 216
 - 6.7 解决方案：创建更好的文本编辑器（第一部分） 218
 - 6.8 解决方案：创建更好的文本编辑器（第二部分） 221
 - 6.8.1 启用Attributed Text 221
 - 6.8.2 控制文本的样式 221
 - 6.8.3 可供UIResponder使用的其他功能 223
 - 6.9 解决方案：过滤用户所输入的文本 224
 - 6.10 解决方案：检测文本模式 226
 - 6.10.1 构建自己的正则表达式 227
 - 6.10.2 枚举正则表达式 227
 - 6.10.3 数据探测器 228
 - 6.10.4 使用内置类型的探测器 229
 - 6.10.5 有用的网站 229
 - 6.11 解决方案：检测UITextView中的拼写错误 231
 - 6.12 搜寻文本中的字符串 232
 - 6.13 小结 233
- 第7章 使用视图控制器 234
 - 7.1 视图控制器 234
 - 7.1.1 UINavigationController类 235
 - 7.1.2 导航控制器 235
 - 7.1.3 标签栏控制器 235
 - 7.1.4 分栏视图控制器 236
 - 7.1.5 页面视图控制器 236
 - 7.1.6 popover控制器 236
 - 7.2 使用导航控制器与分栏视图控制器来开发程序 237
 - 7.2.1 使用导航控制器与导航栈 238
 - 7.2.2 推入与弹出视图控制器 239
 - 7.2.3 导航栏上的按钮 239
 - 7.2.4 延伸至屏幕边缘的布局形式 240
 - 7.3 解决方案：UINavigationController类 241
 - 7.3.1 标题与后退按钮 242
 - 7.3.2 宏 242
 - 7.4 解决方案：模态界面 244
 - 7.5 解决方案：构建分栏视图控制器 248
 - 7.6 解决方案：用分栏视图及导航控制器创建通用的程序 253
 - 7.7 解决方案：标签栏 255
 - 7.8 记住标签的状态 259
 - 7.9 解决方案：页面视图控制器 262
 - 7.9.1 与书籍展示风格有关的属性 262
 - 7.9.2 封装实现细节 263
 - 7.9.3 范例代码详解 269
 - 7.9.4 构建界面索引 270
 - 7.10 解决方案：自定义的容器 271
 - 7.10.1 添加与移除子视图控制器 274

- 7.10.2 视图控制器之间的切换效果 275
- 7.11 解决方案：segue 276
- 7.12 小结 282
- 第8章 常用的控制器 284
- 8.1 图像选取器控制器 284
 - 8.1.1 图像来源 284
 - 8.1.2 在iPhone和iPad中显示选取器 285
- 8.2 解决方案：选取图像 286
 - 8.2.1 向模拟器中添加图片 286
 - 8.2.2 AssetsLibrary模块 286
 - 8.2.3 展示选取器 287
 - 8.2.4 处理delegate的回调 288
- 8.3 解决方案：拍摄照片 293
 - 8.3.1 配置选取器 293
 - 8.3.2 显示图像 295
 - 8.3.3 把图像保存到相册 295
- 8.4 解决方案：录制视频 297
 - 8.4.1 创建录制视频用的选取器 298
 - 8.4.2 保存视频 299
- 8.5 解决方案：用媒体播放器播放视频 299
- 8.6 解决方案：编辑视频 302
- 8.7 解决方案：选取并编辑视频 304
- 8.8 解决方案：通过电子邮件发送图片 306
- 8.9 解决方案：发送文本消息 309
- 8.10 解决方案：在社交网站发布消息 311
- 8.11 小结 313
- 第9章 创建并管理表格视图 314
- 9.1 iOS的表格 314
- 9.2 委托 315
- 9.3 创建表格 316
 - 9.3.1 表格的样式 316
 - 9.3.2 排布表格视图 316
 - 9.3.3 设置数据源 317
 - 9.3.4 提供单元格 317
 - 9.3.5 注册单元格类 317
 - 9.3.6 从队列中取出单元格 318
 - 9.3.7 设置delegate 318
- 9.4 解决方案：实现简单的表格 319
 - 9.4.1 数据源方法 319
 - 9.4.2 响应用户的触摸 322
- 9.5 UITableViewCell类 322
 - 9.5.1 单元格的selectionStyle属性 323
 - 9.5.2 添加自定义的单元格受选效果 323
- 9.6 解决方案：创建带有选取标记的单元格 323
- 9.7 给单元格添加详情展示控件 325
- 9.8 解决方案：编辑表格 327
 - 9.8.1 添加撤销功能 331
 - 9.8.2 实现撤销功能 332
 - 9.8.3 显示移除单元格所用的控件 332
 - 9.8.4 处理删除请求 332
 - 9.8.5 通过滑动手势删除单元格 333
 - 9.8.6 调整单元格的顺序 333
 - 9.8.7 添加单元格 333
- 9.9 解决方案：操控表格的区段 334

- 9.9.1 构建区段 334
- 9.9.2 区段数量与区段内的行数 335
- 9.9.3 返回单元格 335
- 9.9.4 创建每个区段的头部标题 337
- 9.9.5 定制表格与区段的头部及尾部 338
- 9.9.6 创建区段索引 338
- 9.9.7 处理索引与区段不匹配的问题 339
- 9.9.8 为分区表格实现委托方法 339
- 9.10 解决方案：在表格中搜索 339
 - 9.10.1 创建搜索显示控制器 341
 - 9.10.2 为搜索显示控制器注册单元格 341
 - 9.10.3 构建支持搜索功能的数据源方法 342
 - 9.10.4 委托方法 343
 - 9.10.5 使用与搜索功能相配套的索引 344
- 9.11 解决方案：给表格添加下拉刷新功能 345
- 9.12 解决方案：添加指令行 348
- 9.13 制作自定义的分组表格 351
- 9.14 解决方案：构建含有多个滚轮的表格 352
 - 9.14.1 创建UIPickerView 353
 - 9.14.2 数据源方法与委托方法 353
 - 9.14.3 使用带有选取器的视图 354
- 9.15 使用UIDatePicker 356
- 9.16 小结 357
- 第10章 集合视图 358
 - 10.1 集合视图与表格的异同 358
 - 10.2 建立集合视图 360
 - 10.2.1 通过控制器使用集合视图 361
 - 10.2.2 直接使用集合视图 361
 - 10.2.3 数据源与委托 362
 - 10.3 流式布局 362
 - 10.3.1 滚动方向 362
 - 10.3.2 条目的尺寸以及行间距 362
 - 10.3.3 头部与尾部的尺寸 364
 - 10.3.4 内边距 365
 - 10.4 解决方案：采用流式布局的简单集合视图 366
 - 10.5 解决方案：自定义单元格 370
 - 10.6 解决方案：水平滚动的列表 372
 - 10.7 解决方案：创建交互式的布局效果 375
 - 10.8 解决方案：滚动之后自动调整位置 377
 - 10.9 解决方案：创建圆形布局 378
 - 10.9.1 实现创建条目与删除条目时的动画效果 381
 - 10.9.2 增强圆形布局的实用性 382
 - 10.9.3 布局对象 383
 - 10.10 解决方案：用手势调整布局 383
 - 10.11 解决方案：创建真正的网格状布局 385
 - 10.12 解决方案：为集合视图中的条目添加自定义菜单 391
 - 10.13 小结 393
- 第11章 分享文档与数据 394
 - 11.1 解决方案：使用统一类型标识符 394
 - 11.1.1 根据文件扩展名来决定UTI 395
 - 11.1.2 把UTI转换成扩展名或MIME类型 396
 - 11.1.3 判断两个UTI之间是否有依从关系 397
 - 11.1.4 获取依从关系列表 398
 - 11.2 解决方案：访问系统剪贴板 400

- 11.2.1 存储数据 401
- 11.2.2 存储常见类型的数据 401
- 11.2.3 获取数据 402
- 11.2.4 自动更新剪贴板 402
- 11.3 解决方案：监控Documents文件夹 403
 - 11.3.1 启用文件分享功能 403
 - 11.3.2 用户对Documents文件夹的控制能力 403
 - 11.3.3 在Xcode里访问应用程序沙盒 405
 - 11.3.4 扫描新的文档 405
- 11.4 解决方案：活动视图控制器 408
 - 11.4.1 展示活动视图控制器 409
 - 11.4.2 UIActivityItemSource协议 409
 - 11.4.3 UIActivityItemProvider类 411
 - 11.4.4 实现UIActivityItemSource协议中的回调方法 411
 - 11.4.5 添加分享服务 412
 - 11.4.6 与各种数据类型相对应的操作 416
 - 11.4.7 排除某些操作 417
- 11.5 解决方案：Quick Look预览控制器 417
- 11.6 解决方案：使用文档交互控制器 420
 - 11.6.1 创建UIDocumentInteraction-Controller实例 420
 - 11.6.2 UIDocumentInteraction-Controller的属性 424
 - 11.6.3 提供快速查看文档的功能 424
 - 11.6.4 判断是否应启用“Open in...”操作 425
- 11.7 解决方案：声明程序所支持的文档类型 426
 - 11.7.1 创建自定义的文档类型 427
 - 11.7.2 实现对文档的支持 428
- 11.8 解决方案：创建基于URL的服务 431
 - 11.8.1 声明模式 432
 - 11.8.2 测试URL 433
 - 11.8.3 添加处理程序方法 433
- 11.9 小结 434
- 第12章 浅谈Core Data 435
 - 12.1 Core Data简介 435
 - 12.2 实体与模型 436
 - 12.2.1 构建模型文件 436
 - 12.2.2 属性与关系 437
 - 12.2.3 构建NSManagedObject的子类 437
 - 12.3 创建上下文 438
 - 12.4 添加数据 439
 - 12.5 查询数据库 441
 - 12.5.1 配置NSFetchRequest 442
 - 12.5.2 执行数据获取操作 443
 - 12.6 移除对象 444
 - 12.7 解决方案：用Core Data来充当表格的数据源 445
 - 12.7.1 访问索引路径 445
 - 12.7.2 sectionNameKeyPath属性 445
 - 12.7.3 获取每个区段内的对象 445
 - 12.7.4 sectionIndexTitles属性 446
 - 12.7.5 Core Data与表格之间的紧密结合 446
 - 12.8 解决方案：用Core Data实现表格的搜索功能 448
 - 12.9 解决方案：为Core Data表格视图添加编辑功能 450
 - 12.9.1 添加撤销/重做功能 451
 - 12.9.2 创建撤销事务 452
 - 12.9.3 重新思考编辑功能 452

- 12.10 解决方案：由Core Data所驱动的集合视图 456
- 12.11 小结 461
- 第13章 网络编程基础 462
 - 13.1 解决方案：判断网络状态 462
 - 13.2 监测联网状况是否发生变化 465
 - 13.3 URL加载系统 467
 - 13.3.1 配置 467
 - 13.3.2 任务 468
 - 13.3.3 NSURLSession 468
 - 13.4 解决方案：简单的下载 469
 - 13.5 解决方案：在下载过程中提供反馈 473
 - 13.6 解决方案：后台传输 482
 - 13.6.1 测试后台传输 484
 - 13.6.2 Web服务 484
 - 13.7 解决方案：使用NSJSONSerialization类 485
 - 13.8 解决方案：将XML转换为树状结构 487
 - 13.8.1 树 489
 - 13.8.2 构建解析树 489
 - 13.9 小结 492
- 第14章 针对特定设备的开发 493
 - 14.1 访问基本的设备信息 493
 - 14.2 添加设备能力限制 494
 - 14.2.1 提供描述信息以征求用户同意 496
 - 14.2.2 Info.plist文件中其他常用的键 496
 - 14.3 解决方案：检查设备距离与电池状态 496
 - 14.3.1 启用与禁用距离感应器 497
 - 14.3.2 监控电池状态 497
 - 14.3.3 判断设备是否具有Retina显示屏 499
 - 14.4 解决方案：获取设备的其他信息 500
 - 14.5 Core Motion基础知识 502
 - 14.5.1 判断设备是否支持某种感应器 502
 - 14.5.2 获取感应器数据 503
 - 14.6 解决方案：通过加速度来判断“上”方向 503
 - 14.7 使用基本的方向值 504
 - 14.7.1 根据加速计来判断方向 505
 - 14.7.2 计算相对角度 506
 - 14.8 解决方案：使用加速计来移动屏幕上的物体 507
 - 14.9 解决方案：基于加速计的滚动视图 511
 - 14.10 解决方案：获取并使用设备的姿态 513
 - 14.11 用Motion Event来检测晃动 514
 - 14.12 使用外接屏幕 515
 - 14.12.1 检测屏幕 516
 - 14.12.2 获取屏幕分辨率 517
 - 14.12.3 配置视频输出 517
 - 14.12.4 添加CADisplayLink 518
 - 14.12.5 对过扫描进行补偿 518
 - 14.12.6 VIDEOkit 518
 - 14.13 追踪用户 521
 - 14.14 查询可用的磁盘空间 522
 - 14.15 小结 523
- 第15章 辅助功能 524
 - 15.1 辅助功能基础知识 524
 - 15.2 启用辅助功能 526
 - 15.3 特征 526

- 15.4 标签 528
- 15.5 提示语 528
- 15.6 用模拟器测试辅助功能 529
- 15.7 把变化情况传播出去 531
- 15.8 在iOS上面测试辅助功能 531
- 15.9 语音合成 533
- 15.10 动态字体 534
- 15.11 小结 535
- 附录A Objective-C字面量 536
- • • • • [\(收起\)](#)

[iOS核心开发手册（原书第5版）](#) [下载链接1](#)

标签

iOS

手機軟件開發

ios

iOS, 移动开发

iOS開發

编程

技术

iOS进阶

评论

这本书写的很棒，很奇怪的是卖的却一般。。

有点乱

学习iOS

[iOS核心开发手册（原书第5版）_下载链接1](#)

书评

[iOS核心开发手册（原书第5版）_下载链接1](#)