

HZ BOOKS
华章科技

ELSEVIER
爱思唯尔

mbed是由嵌入式领域的领导厂商ARM在全球大力推广的开源硬件项目。

本书是首本针对mbed开源硬件项目的书籍。

理论与实践相结合，对嵌入式系统设计每个环节的讲解都入木三分。

本书是开源硬件、积木式中间件、快速系统原型设计三大核心理念的载体。



电子与嵌入式系统
设计译丛



Fast and Effective Embedded Systems Design

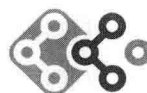
ARM快速嵌入式系统 原型设计

基于开源硬件mbed

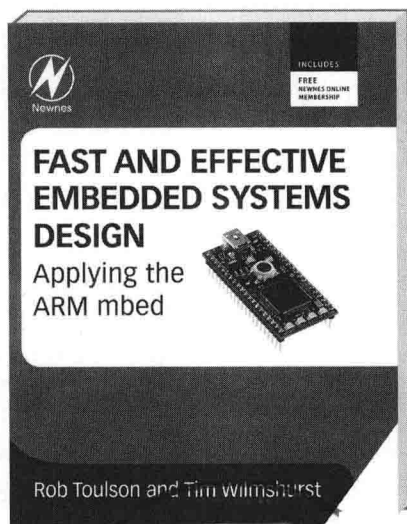
[英] Rob Toulson Tim Wilmshurst 著 韩德强 鲁鹏程 等译



机械工业出版社
China Machine Press



电子与嵌入式系统
设计译丛



Fast and Effective Embedded Systems Design
Applying the ARM mbed

ARM快速嵌入式系统 原型设计

基于开源硬件mbed

[英] Rob Toulson Tim Wilmsheurst 著 韩德强 鲁鹏程 等译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

ARM 快速嵌入式系统原型设计: 基于开源硬件 mbed / (英) 托尔森 (Toulson, R.), (英) 威尔姆斯特 (Wilmshurst, T.) 著; 韩德强等译. —北京: 机械工业出版社, 2014.3
(电子与嵌入式系统设计译丛)

书名原文: Fast and Effective Embedded Systems Design: Applying the ARM mbed

ISBN 978-7-111-46019-0

I. A… II. ①托… ②威… ③韩… III. 微处理器-系统设计 IV. TP332

中国版本图书馆 CIP 数据核字 (2014) 第 035605 号

本书版权登记号: 图字: 01-2013-1634

Fast and Effective Embedded Systems Design: Applying the ARM mbed

Rob Toulson and Tim Wilmshurst

ISBN 978-0-08-097768-3

Copyright © 2012 Rob Toulson and Tim Wilmshurst. Published by Elsevier Ltd. All rights reserved.

Authorized Simplified Chinese translation edition published by the Proprietor.

Copyright © 2014 by Elsevier (Singapore) Pte Ltd. All rights reserved.

Printed in China by China Machine Press under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in China only, excluding Hong Kong SAR, Macau SAR and Taiwan. Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书简体中文版由 Elsevier (Singapore) Pte Ltd. 授权机械工业出版社在中国大陆境内独家出版和发行。本版仅限在中国境内 (不包括香港特别行政区、澳门特别行政区及台湾地区) 出版及标价销售。未经许可之出口, 视为违反著作权法, 将受法律之制裁。

本书封底贴有 Elsevier 防伪标签, 无标签者不得销售。

本书旨在通过 mbed 介绍嵌入式系统设计的所有主要议题, 便于读者快速掌握嵌入式系统的设计方法。本书共 15 章。第 1~10 章从基本的原理和简单的项目入手, 使用 mbed 项目示例提供一套完整的嵌入式系统设计入门课程, 旨在揭示如何使用 mbed 快速地设计嵌入式系统。第 11~15 章逐渐深入到更专业的领域, 阐述嵌入式系统的设计精髓, 为读者进一步阅读或学习更高级的课程打基础。

ARM 快速嵌入式系统原型设计: 基于开源硬件 mbed

(英) Rob Toulson Tim Wilmshurst 著

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: 谢晓芳 秦健 刘涛

印刷: 北京市荣盛彩色印刷有限公司

版次: 2014 年 3 月第 1 版第 1 次印刷

开本: 186mm × 240mm 1/16

印张: 19.25

书号: ISBN 978-7-111-46019-0

定价: 69.00 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

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

译者序

近年来，基于 ARM 处理器的嵌入式系统飞速发展，遍及各种移动设备及工业和家庭控制系统。

介绍基于 ARM 处理器的嵌入式系统的书籍琳琅满目，这些书籍或以某个或某类处理器为基础，又或者以某个嵌入式系统为基础介绍嵌入式系统的概念及应用开发。而本书的两位作者均是大学教师，具有丰富的嵌入式系统教学和工程实践经验，他们倡导“在实践中学习”的先进理念，在书中给出了大量基于 ARM mbed 的动手实验教程，在突出实践能力培养的同时，又在每章中针对嵌入式系统的各个功能模块给出了相应基础知识的介绍，免除了读者再去翻阅其他书籍的麻烦。本书的另一个亮点就是不需读者安装软件和配置繁琐的软件开发环境，只需一台能够上网的 PC 即可，通过特有的 Web 软件开发环境，即可完成应用程序的开发过程。以译者 20 多年的嵌入式系统教学与开发经验来看，本书非常适合作为本科、高职高专各专业的嵌入式系统基础课程教材。

本书由北京工业大学计算机学院的部分教师翻译，其中韩德强翻译了前言和第 1~2 章，鲁鹏程翻译了第 3~6 章和附录，张丽艳翻译了第 7~9 章，杨淇善翻译了第 10 章，王宗侠翻译了第 11~12 章，邵温翻译了第 13~15 章，全书的审校由韩德强完成。

在本书的翻译过程中得到了 ARM 公司中国大学计划经理时昕博士的大力支持与关注，并提供了 ARM mbed 开发板，在此对时博士表示由衷的感谢！

限于译者的水平，翻译中难免有错误或不妥之处，真诚希望各位读者批评指正。

韩德强

2014 年 1 月于北京工业大学

前 言

微处理器无处不在，它为汽车、手机、家居和办公设备、电视和娱乐系统、药品、飞机等无穷无尽的产品提供“智力”。这些司空见惯的产品内部都隐藏着一个微处理器使其智能化，我们称这些产品为嵌入式系统。

不久前，嵌入式系统的设计者必须是电子或软件方面的专家，或者两者都是。现在，无论是专业人员还是初学者，使用界面友好的、复杂的积木式构件，便可以很快设计出嵌入式系统。这种积木式构件便是最近由著名的计算机巨头 ARM 推出的 mbed。本书将围绕 mbed 介绍嵌入式系统设计的所有主要议题，目的在于通过 mbed 的使用教会读者嵌入式系统设计的要素。

本书分为两部分。第 1 ~ 10 章广泛介绍嵌入式系统，使用 mbed 展示如何应用它快速地进行成功的嵌入式系统设计。这几章旨在全力帮助读者掌握一系列精心构建的概念和练习，从基本的原理和简单的项目入手，逐步完成更高级的系统设计。第 11 ~ 15 章在此基础上进入到了许多更高级的系统设计领域。这里讲的速度可能会有点快，你会发现需要进行更多的背景研究。

本书仅要求读者具备基本的电工电子理论知识。本书采用“在实践中学习”的方法，为了更好地使用本书，你需要一块 mbed 开发板，一台连接到因特网的计算机，还需要书中指定的各种额外的电子元件。如果不做某一个实验或者实践书中某一部分的内容，就不需要准备其中所需的东西。你还会用到数字电压表，不过最好使用示波器，这样能看到更多细节。

全书每章围绕一个嵌入式系统主题展开。每一章或多或少都有一些理论的介绍，其中很多章需要更多的理论基础，然后才能进行一系列实际的实验。准备好将你的 mbed 连接到下一个电路，下载并编译下一个示例程序，然后运行该程序以理解到底是怎么回事。随着你对 mbed 信心的增长，你的创造力和原创性会随之增加，开始将你的想法变成可行的项目吧！

本书会快速地帮助你：

- 理解和应用嵌入式系统的关键环节。
- 理解和应用 ARM mbed 的关键环节。
- 从头学起或提高嵌入式 C/C++ 编程技巧。

- 加深对电子元器件及配置的理解。
- 了解 mbed 如何被应用到一些新兴的、最令人兴奋的、创新的智能产品中。
- 完成你从未想过自己会有能力完成的设计和创新!

如果你遇到问题或者有任何疑问，可以通过本书的网站和 mbed 的网站寻求技术支持，你也可以通过电子邮件与作者讨论。

如果你是一位大学教师，本书为你的嵌入式系统课程提供“完整的解决方案”。两位作者都是经验丰富的大学讲师，他们在写书时考虑到了你的学生。本书包含一个实践和理论学习活动的组织顺序。理想状态下，你需要为每个或每对学生配备一套 mbed（一块原型面包板和一个套件），因其高度便携，开发工作不必局限在学校的实验室中进行。在后面的课程中，学生们将会用他们的 mbed 开发板联网。教师可以在本书配套网站（www.embeddedacademic.com）下载每一章的完整 PPT 演示文稿，也可以下载测验题的答案以及练习题和小项目的示例代码。

本书第一部分（第 1 ~ 10 章）使用 mbed 项目示例提供了一个完整的嵌入式系统设计入门课程。第二部分（第 11 ~ 15 章）用于提高课程，为进一步阅读或学习更高级的课程打基础。

本书采用了一种亲身实践的有趣学习方式，适用于任何想要引入嵌入式系统概念的课程。因为仅需要一点电子学理论，所以本书更适用于那些不以使用嵌入式系统为目的的学科。本书原本供工程学、物理学或计算机科学等学科大学一年级本科生使用，但我们希望学生们能更多地高年级使用它。实践类的专业人员和业余爱好者也会对本书感兴趣。

本书由位于英国剑桥的安格利亚鲁斯金大学的研究员 Rob Toulson 和英国德比大学电子专业负责人 Tim Wilmshurst 编写。博士毕业后的几年里，Rob 主要在音频和汽车领域从事数字信号处理与控制系统工程项目。之后，他成为了一名研究员，当前主要致力于推进技术和创意产业的合作研究。到德比大学之前，Tim 负责剑桥大学工程系的电子开发小组多年，他的设计生涯见证了很多微控制器和嵌入式系统的发展过程。除了分享对嵌入式系统的兴趣之外，我们还分享对音乐和音乐技术的兴趣。这本书汇集了我们广泛的经验。规划好书的整体布局，进行了一些最初的准备工作后，我们对章节进行了划分。Tim 负责前几章大部分内容的撰写，并负责电子和计算机硬件相关的章节。Rob 主要负责后面几章的撰写以及 mbed 高级应用。这种工作分工主要是为了方便，而在出版物中我们彼此都对所有章节承担责任。由于 Tim 之前写过一些嵌入式系统方面的书籍，本书中的一些背景知识和图表就取自这些书，这主要是因为需要在需要解释背景的地方“从零开始”似乎是没有意义的。

目 录

译者序

前 言

第一部分 嵌入式系统概述与 玩转 mbed

第 1 章 嵌入式系统、微控制器 与 ARM

1.1 嵌入式系统简介	2
1.1.1 什么是嵌入式系统	2
1.1.2 嵌入式系统示例	3
1.2 微处理器与微控制器	4
1.2.1 计算机主要组件	5
1.2.2 微控制器	6
1.3 嵌入式系统的开发流程	7
1.3.1 程序语言：C/C++ 有什么 特别之处	7
1.3.2 开发周期	7
1.4 进入 ARM 世界	8
1.4.1 关于 ARM 的历史	8
1.4.2 技术细节：RISC 的意义	9
1.4.3 Cortex 内核	10
本章回顾	11
习题	11
参考文献	11

第 2 章 mbed 开发板

2.1 mbed 简介	12
2.1.1 mbed 体系结构	14
2.1.2 LPC1768 微控制器	15
2.2 mbed 入门教程	16
2.2.1 步骤 1：连接 mbed 到 PC	17
2.2.2 步骤 2：创建 mbed 账户	17
2.2.3 步骤 3：运行程序	17
2.2.4 步骤 4：编译程序	18
2.2.5 步骤 5：下载程序二进制 代码	19
2.2.6 步骤 6：修改程序代码	19
2.3 开发环境	19
2.3.1 mbed 编译器和 API	19
2.3.2 C/C++ 的使用	20
本章回顾	20
习题	20
参考文献	21

第 3 章 数字输入和输出

3.1 开始编写程序	22
3.1.1 思考第一个程序	22
3.1.2 了解 mbed 的 API 函数	25
3.1.3 分析 while 循环	25
3.2 用电压表示逻辑值	27

3.3	mbed 数字输出	27	4.4.1	使用 mbed 的 PWM 信号源	52
3.3.1	发光二极管的使用	28	4.4.2	一些 PWM 输出实验	53
3.3.2	mbed 外部引脚的使用	29	4.4.3	控制小电机的速度	55
3.4	mbed 数字输入	30	4.4.4	用软件方式产生 PWM	55
3.4.1	开关与数字系统的连接	30	4.4.5	伺服控制	56
3.4.2	DigitalIn API	31	4.4.6	输出到一个压电转换器	57
3.4.3	用 if 语句响应开关输入	31	本章回顾		59
3.5	简单的光电设备接口	33	习题		60
3.5.1	光敏反射和透射传感器	33	参考文献		60
3.5.2	光敏传感器与 mbed 开发板的连接	34	第 5 章 模拟输入		61
3.5.3	七段数码管显示	35	5.1	数模转换	61
3.5.4	七段数码管与 mbed 开发板的连接	36	5.1.1	模 - 数转换器	61
3.6	驱动大型直流负载	39	5.1.2	范围、分辨率和量化	62
3.6.1	使用晶体管驱动	39	5.1.3	采样频率	64
3.6.2	用 mbed 进行电机驱动控制	40	5.1.4	mbed 开发板上的模拟输入	64
3.6.3	驱动多个七段数码管	41	5.2	模拟输入和输出混合应用	65
3.7	小项目: 字母计数器	42	5.2.1	用可变电压控制 LED 亮度	65
本章回顾		42	5.2.2	用 PWM 控制 LED 亮度	66
习题		43	5.2.3	PWM 频率控制	67
参考文献		44	5.3	模拟输入数据的处理	68
第 4 章 模拟输出		45	5.3.1	在计算机屏幕上显示数值	68
4.1	数据转换简介	45	5.3.2	将 ADC 输出调整到识别范围内	69
4.2	mbed 开发板上的模拟输出	46	5.3.3	采用平均值降低噪声	69
4.2.1	产生恒定的输出电压	47	5.4	一些简单的模拟传感器	70
4.2.2	锯齿波	47	5.4.1	光敏电阻	70
4.2.3	测试 DAC 分辨率	50	5.4.2	集成电路温度传感器	71
4.2.4	产生正弦波	50	5.5	分析数据转换时间	71
4.3	另一种形式的模拟量输出: 脉冲宽度调制	51	5.6	小项目: 二维光跟踪	73
4.4	mbed 开发板上的脉冲宽度调制	52	本章回顾		73
			习题		74

参考文献	74	7.3.1 ADXL345 加速器简介	99
第 6 章 高级编程技术	75	7.3.2 简单 ADXL345 程序开发	100
6.1 思考程序设计和程序结构带来 的好处	75	7.4 SPI 评估	102
6.2 函数	75	7.5 I ² C 总线	103
6.3 程序设计	76	7.5.1 I ² C 总线简介	103
6.3.1 使用流程图定义代码结构	76	7.5.2 mbed 开发板上的 I ² C 总线	105
6.3.2 伪代码	77	7.5.3 设置 I ² C 数据链路	105
6.4 在 mbed 开发板上使用函数	78	7.6 用 I ² C 总线标准的温度传感器 通信	108
6.4.1 实现七段数码管计数器	79	7.7 SRF08 超声波测距仪的使用	110
6.4.2 函数重用	80	7.8 I ² C 总线评估	112
6.4.3 一个使用函数且更复杂的 程序	81	7.9 异步串行数据通信	112
6.5 在 C/C++ 中使用多个文件	83	7.9.1 异步串行通信简介	113
6.5.1 C/C++ 程序编译过程 概述	83	7.9.2 mbed 开发板上的异步串行 通信应用	113
6.5.2 C/C++ 预处理器和预 处理器指令	84	7.9.3 同宿主计算机的同步串行 通信应用	116
6.5.3 #ifndef 伪指令	85	7.10 小项目: 多节点 I ² C 总线	116
6.5.4 全局地使用 mbed 对象	86	本章回顾	116
6.6 模块化程序示例	86	习题	116
本章回顾	89	参考文献	117
习题	90	第 8 章 液晶显示器	118
第 7 章 串行通信	91	8.1 显示技术	118
7.1 同步串行通信简介	91	8.1.1 液晶技术简介	118
7.2 串行外围接口	92	8.1.2 液晶字符显示	119
7.2.1 SPI 简介	93	8.2 使用 PC1602F LCD	120
7.2.2 mbed 开发板上的 SPI	94	8.2.1 PC1602F 显示器简介	121
7.2.3 设置 mbed SPI 主设备	94	8.2.2 连接 PC1602F 到 mbed 开发板	121
7.2.4 创建 SPI 数据链路	95	8.2.3 LCD 接口的模块化编程	122
7.3 智能仪表和 SPI 加速器	99	8.2.4 初始化显示	123
		8.2.5 向 LCD 发送显示数据	124

8.2.6	完整的 LCP.cpp 定义	125	9.5.2	使用计数器作为定时器	146
8.2.7	使用 LCD 函数	126	9.5.3	mbed 上的定时器	146
8.2.8	向指定位置添加数据	127	9.6	使用 mbed 定时器	146
8.3	使用 mbed 开发板的 TextLCD 库	128	9.6.1	使用多个 mbed 定时器	147
8.4	在 LCD 上显示模拟输入数据	130	9.6.2	测试定时器延迟	148
8.5	更先进的 LCD	131	9.7	使用 mbed 超时	150
8.5.1	彩色 LCD	131	9.7.1	超时应用简单示例	150
8.5.2	控制 SPI 标准的 LCD 手机 显示屏	132	9.7.2	超时进阶应用	151
8.6	小项目：数字水平仪	134	9.7.3	用超时测试反应时间	152
	本章回顾	134	9.8	使用 mbed 断续装置	153
	习题	135	9.8.1	节拍器中使用断续装置	154
	参考文献	135	9.8.2	思考多任务节拍器程序	156
			9.9	实时时钟	157
第 9 章	中断、定时器和任务	136	9.10	开关去除抖动	157
9.1	嵌入式系统中的定时和任务	136	9.11	小项目	159
9.1.1	定时器和中断	136	9.11.1	独立节拍器	159
9.1.2	任务	136	9.11.2	加速度计阈值中断	159
9.1.3	事件触发任务和时间触发 任务	137		本章回顾	160
9.2	响应事件触发的事件	137		习题	160
9.2.1	轮询	137	第 10 章	存储器与数据管理	161
9.2.2	中断简介	138	10.1	存储器综述	161
9.3	简单的 mbed 中断	139	10.1.1	存储器功能类型	161
9.4	深入理解中断	140	10.1.2	基本电子存储器类型	161
9.4.1	LPC1768 中断	142	10.2	使用 mbed 的数据文件	163
9.4.2	测试中断延迟	142	10.2.1	回顾部分所需的 C/C++ 库函数	164
9.4.3	禁用中断	143	10.2.2	定义 mbed 的本地文件 系统	164
9.4.4	模拟输入中断	144	10.2.3	打开和关闭文件	164
9.4.5	中断总结	145	10.2.4	写入和读取文件数据	165
9.5	定时器	145	10.3	mbed 数据文件存取示例	165
9.5.1	数字计数器	145	10.3.1	文件存取	165

10.3.2	字符串文件存取	166
10.3.3	使用格式化数据	167
10.4	使用 mbed 的外部存储器	168
10.5	指针简介	170
10.6	小项目：加速度计阈值的记录	172
	本章回顾	173
	习题	173
	参考文献	173

第二部分 高级和专家级应用

第 11 章 数字信号处理 176

11.1	数字信号处理器简介	176
11.2	数字滤波示例	176
11.3	mbed DSP 示例	178
11.3.1	数字数据的输入和输出	178
11.3.2	信号重构	180
11.3.3	添加一个数字低通滤波器	182
11.3.4	添加一个激活按钮	183
11.3.5	数字高通滤波器	184
11.4	延迟 / 回声效果	184
11.5	使用 wave 音频文件	187
11.5.1	波形信息的头部	187
11.5.2	用 mbed 读取 wave 文件的头部	189
11.5.3	读取、输出单声道 wave 数据	191
11.6	DSP 小结	194
11.7	小项目：立体声播放器	194
11.7.1	基本功能的立体声播放器	194
11.7.2	拥有 PC 接口的立体声播放器	194

11.7.3	拥有手机显示接口的便携式立体声播放器	194
	本章回顾	194
	习题	195
	参考文献	195

第 12 章 高级串行通信 196

12.1	高级串行通信协议简介	196
12.2	蓝牙串行通信	196
12.2.1	蓝牙简介	196
12.2.2	蓝牙模块 RN-41 和 RN-42 的接口	197
12.2.3	通过蓝牙发送 mbed 数据	197
12.2.4	从主机终端应用程序接收的蓝牙数据	199
12.2.5	两个 mbed 之间通过蓝牙通信	199
12.3	USB 简介	202
12.3.1	使用 mbed 模拟 USB 鼠标	203
12.3.2	从 mbed 端发送 USB MIDI 数据	203
12.4	以太网简介	206
12.4.1	以太网概述	206
12.4.2	实现简单的 mbed 以太网通信	207
12.4.3	mbed 之间的以太网通信	209
12.5	用 mbed 进行本地网络和 Internet 通信	211
12.5.1	用 mbed 作为 HTTP 客户端	211
12.5.2	用 mbed 作为 HTTP 文件服务器	213
12.5.3	用远程过程调用修改 mbed 输出	214

12.5.4 用远程 JavaScript 接口 控制 mbed	216	14.4 深入了解控制寄存器	244
本章回顾	218	14.4.1 引脚功能选择寄存器和 引脚模式寄存器	245
习题	219	14.4.2 功率控制寄存器和时钟 选择寄存器	246
参考文献	219	14.5 使用 DAC	248
第 13 章 控制系统	220	14.5.1 mbed DAC 控制寄存器 ..	248
13.1 控制系统简介	220	14.5.2 DAC 的应用	249
13.1.1 闭环和开环控制系统	220	14.6 使用 ADC	250
13.1.2 闭环巡航控制示例	221	14.6.1 mbed ADC 控制寄存器 ..	250
13.1.3 比例控制	223	14.6.2 ADC 应用	251
13.1.4 PID 控制	224	14.6.3 改变 ADC 转换速度	253
13.2 闭环数字罗盘示例	225	14.7 控制寄存器使用小结	255
13.2.1 HMC6352 数字罗盘的 使用	225	本章回顾	255
13.2.2 360° 旋转伺服系统的 实现	227	习题	256
13.2.3 闭环控制算法的实现	229	参考文献	256
13.3 基于控制器局域网控制数据 通信	231	第 15 章 项目扩展	257
13.3.1 控制器局域网	231	15.1 去往何方	257
13.3.2 mbed 上的 CAN 总线	232	15.2 mbed Pololu 机器人	257
本章回顾	237	15.3 高级音频项目	258
习题	237	15.4 物联网	258
参考文献	237	15.5 mbed LPC11U24 简介	259
第 14 章 mbed 库函数入门	238	15.6 从 mbed 到实际生产	260
14.1 简介	238	15.7 结束语	262
14.2 控制寄存器概念	238	参考文献	263
14.3 数字输入 / 输出	240	附录 A 数制系统	264
14.3.1 mbed 数字输入 / 输出 控制寄存器	240	附录 B C 语言基础	269
14.3.2 数字输出的应用	241	附录 C mbed 技术资料	286
14.3.3 添加第二个数字输出	242	附录 D 配件清单	290
14.3.4 数字输入	243	附录 E Tera Term 终端模拟器 ..	292

第一部分

嵌入式系统概述与玩转 mbed

第 1 章

嵌入式系统、微控制器与 ARM

1.1 嵌入式系统简介

1.1.1 什么是嵌入式系统

我们都对台式机与笔记本电脑非常熟悉，并且惊异于它们的强大功能。这些计算机都是通用计算机，我们可以利用它们在不同的时间做不同的事情，不过这取决于我们在计算机上运行的应用程序。在这些计算机中重要的部分就是微处理器。微处理器是一块微小的、非常复杂的、包含计算机核心细节的电路。其中所有的器件都焊接在一小块称为集成电路（IC）的硅片上。一些非专业的人员也常常把集成电路称为微电路，或是仅仅称为芯片。

为很多人所不熟悉的是，微处理器不仅仅存在于通用计算机中，也可以安置到一些不需要计算的设备内部，比如，洗衣机、烤面包机或者摄像机中。微处理器常常用于控制以上这些产品。微处理器放置在产品内部，在计算机中往往是不可见的，用户可能根本就不知道它在哪儿。此外，计算机往往还关联着如键盘、显示器和鼠标这些附加组件。因为控制这些产品的微控制器镶嵌在产品内部，所以这类产品称作嵌入式系统。在许多情况下，由于部分微处理器更倾向于关注控制，在嵌入式系统中所使用的微处理器相对于通用计算机的中的处理器又逐渐发展出不同的特性，这些称为微控制器。虽然它们在微处理器大家庭中并不起眼，但是它们的影响是巨大的。对于一些电气与系统设计人员来说，这也提供了巨大的商机。

嵌入式系统有很多形式。它们在家庭、电机与工作场合中随处可见。绝大多数家用电器，比如洗衣机、洗碗机、烤箱、中央空调和报警器，都是嵌入式系统。汽车中充满了嵌入式系统，比如引擎管理模块与安全模块（如汽车锁与防盗设备）、空调、刹车、收音机等等。嵌入式系统广泛应用于工业、商业、机械控制、工厂自动化、电子商务与办公设备中。当然，其应用领域远远超过以上范围，并且应用范围正不断扩大。

图 1.1 展示了一个嵌入式系统的简单框图。其中有一组转入来自于被控制系统。嵌入式计算机通常是指一个微控制器，它可以运行一个永久地存储在存储器中的程序。与那些可以

运行许多程序的通用台式机不同，“嵌入式计算机”只能运行一个程序。微控制器根据输入所提供的信息，计算出输出。输出信息往往与系统内部的执行器相连。实际的电子电路与其他与之相连的机电组件通常一同称为硬件。在硬件上运行的程序通常称为软件。除了这些之外，嵌入式系统仍然有很多可以与用户交互的方法，比如通过键盘与显示器。也有很多可以与其他子系统的交互，虽然这些是必不可少的一般性概念。在嵌入式系统中，时间因素是另一个可以影响我们所作所为的变量，在下面的图中表示为一个贯穿的箭头。我们需要测量时间，使得其准确地在我们预先安排的时间内运行，并产生对时间依赖很强的数据流，同时可以及时处理发生的异常。

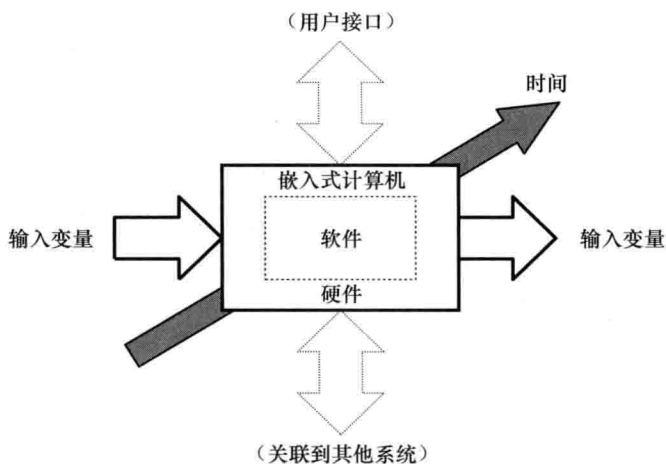


图 1.1 嵌入式系统

这一章阐述并回顾了许多关于计算机、微处理器、微控制器以及嵌入式系统的概念。通过这种回顾的形式，为我们以后学习更深层次的内容提供了一个平台。后面章节描述的内容最终都将回归这些概念，以它们为基础，再加以详细的介绍。如果需要了解更多信息，请参见参考文献 1.1。

1.1.2 嵌入式系统示例

零食自动售货机是一个很好的嵌入式系统例子。在图 1.2 中以框图形式展示了这个例子。图 1.2 的中间是一个微控制器。如图 1.2 所示，微控制器从用户键盘、硬币计数模块以及物品分配模块接受一系列输入信号，并且产生依赖于这些输入的输出。

饥饿的客户们会靠近自动售货机并且猛击按钮或者投入硬币。在第一种按下按钮的情况下，键盘模块会向微控制器发送回馈信号，使得微控制器可以识别出不同的按键，并把这些按键的键值汇总到解码器中以产生更多、更复杂的消息。硬币计数模块也会根据所投入的硬币数量发送信息。微控制器会试图筛选接收到的信息，并输出自身的状态信息至液晶显示屏上。用户是否正确选择了一个产品？是否支付了足够的钱？如果选择了正确的产品并支付

了，微控制器会控制执行器模块分发产品。如果没有，微控制器会在显示屏上显示消息，提示用户投入更多的硬币或者重新输入产品代码。好的自动售货机还应该提供找零功能。同时自动售货机也需要一些感应机制来确保产品使用的可靠性以及分发动作最终完成。以上这些都在图 1.2 中以门位置传感器的形式显示了出来。不好的产品（为什么我们没有遇到呢？）会在显示屏上显示一些没有用的或者让人恼怒的消息，在用户明明给了正确的钱款后却仍然要求付款，或发放给用户错误的产品，甚至不把用户选择的产品送出来。

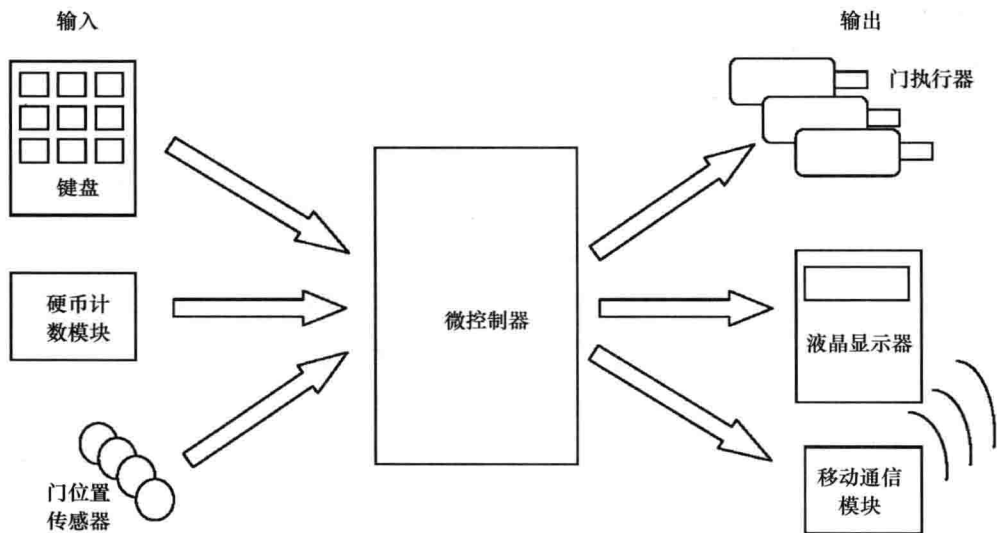


图 1.2 嵌入式自动售货机

以上所描述的都是传统机器的特点，但是我们可以走得更远。在现代的自动售货系统中，可能会植入一些移动通信功能，允许自动售货机直接向维修团队报告产生的故障，这样维修团队很快就能解决这些问题。与此相似，机器也可以通过移动通信或者物联网通信报告库存水平，并提醒服务团队补充库存的不足。

这个简单的例子在图 1.1 中得到准确的反映。微控制器接受输入变量，计算、处理并产生输出。（微处理器的）这些行为其实都包含对时序的正确使用。在上述过程中，一般系统都会提供用户接口。一些现代的机器还提供网络接口。尽管图 1.2 似乎在描述一个硬件系统，但是事实上图 1.2 所述的一切都被设计者编写的软件所控制。那些运行在微控制器上的软件决定了系统实际的功能。

1.2 微处理器与微控制器

回顾之前所述，微控制器是任何嵌入式系统的核心。由于微控制器在本质上是一类计算机，这对我们掌握基本的计算机信息非常重要。为此这里仅作概述，具体功能会在后面的章节中详细介绍。

1.2.1 计算机主要组件

图 1.3 显示了计算机系统的基本元素。作为其存在的目的，一台计算机可以执行算术或者逻辑运算。这些操作在一个称为算术逻辑单元（ALU）的数字电路中完成。算术逻辑单元被放置在一个称为中央处理单元（CPU）的大型电路中，这个电路能为 ALU 提供细节支持。ALU 可以承担一些简单的算术和逻辑运算工作。代码的反馈称为指令。现在我们更加深入地了解什么是计算机。如果我们能够通过给 ALU 提供一系列指令以及指令需要的数据来保持 ALU 的忙碌，那么我们实际上就拥有了一个非常有用的机器。

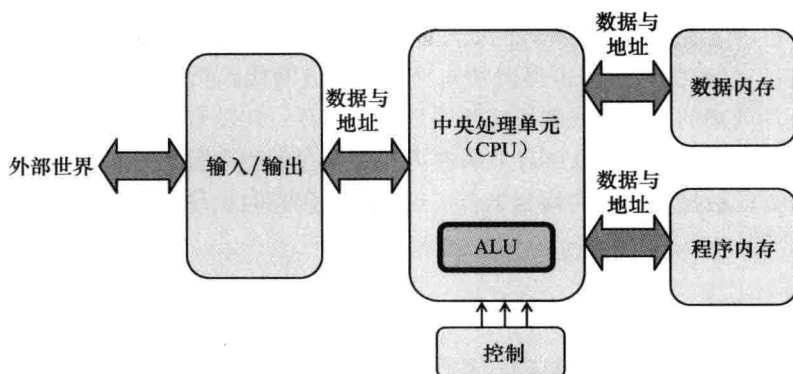


图 1.3 计算机的基本元素

ALU 所需要的指令与数据是由它周边的电路提供的。值得注意的是，这些指令中的任何一条仅仅能完成一个非常简单的功能。但是，由于计算机的运行速度非常快，使得其整体拥有巨大的计算能力。一系列指令集合称为一个程序。程序通常存储在一段内存中，这段内存称为程序存储器。如果这段内存可以永久存放，不论是否供电，程序都会永久地保留在内存中。一旦电源打开，程序就会立刻运行。像这样可以在电源关闭时一直保持内容的内存称为非易失性存储，老式的术语称为只读存储器（ROM）。这个术语一直在使用，尽管对于新的内存技术来说，这一说法已经不再准确。控制电路需要不断访问程序内存以查找下一条指令。ALU 所处理的数据可能来自数据存储器，计算的结果也会存储在那儿。由于通常计算的结果都是一些临时数据，因此存放这些数据的内存不必是永久性的，尽管使用永久性内存也没有什么坏处。这种类型的内存有个过时的定义叫做随机存取存储器（RAM）。尽管我们在这儿使用了过时的术语，但是这也传达了很有用的信息。

无论是什么用途的电脑都必须可以通过它的输入/输出（I/O）与外界沟通。在个人计算机上，这意味着通过键盘、视觉显示单元（Visual Display Unit, VDU）与打印机这类的外设进行人机交互。在最简单的嵌入式系统中，通信更主要是通过传感器和执行器与它周围的物理环境进行的。从外界进入的数据可能会被迅速转移到 ALU 中进行处理，或者存储在数据存储器中。发送到外界的数据也很有可能是 ALU 最近所计算得到的数据。

最后，如图 1.2 中空心箭头所示，在几个主要的框图之间应该有数据通路。数据通路是