



普通高等教育“十一五”国家级规划教材
软件工程专业核心课程系列教材

嵌入式技术基础与实践

(第2版)

王宜怀 张书奎 王林 吴瑾 著

- ◆ 教育部高等学校软件工程专业教学指导分委员
- ◆ 教育部高等学校软件工程专业教学指导分委员会推荐教材
- ◆ 汇集示范性软件工程专业教学成果

清华大学出版社



普通高等教育“十一五”国家级规划教材
软件工程专业核心课程系列教材

嵌入式技术基础与实践

(第2版)

王宜怀 张书奎 王林 吴瑾 著

清华大学出版社
北京

内 容 简 介

本书以飞思卡尔半导体公司(原摩托罗拉半导体部)的8位S08系列微控制器中通用型号MC9S08AW60、带USB 2.0接口MC9S08JM60、带CAN总线接口MC9S08DZ60三个型号为蓝本阐述嵌入式系统的软件与硬件设计。全书共14章,其中第1章为概述,阐述嵌入式系统的知识体系、学习误区与学习建议。第2章给出AW60硬件最小系统,并简要介绍了S08CPU。第3章给出第一个样例程序及CodeWarrior工程组织,完成第一个S08工程的入门。第4章给出基于硬件构件的嵌入式系统开发方法。第5章阐述串行通信接口SCI,并给出第一个带中断的编程实例。第1~5章完成了学习一个新MCU完整要素的入门。第6~12章分别给出键盘、LED、LCD、定时器、SPI、Flash存储器在线编程、I²C、A/D及AW60芯片的其他模块等。第13、14章分别给出JM60的USB 2.0编程及DZ60的CAN总线编程。附录给出了相关资料。

本书提供的所有实例源程序、辅助资料、相关芯片资料及常用软件工具,可在清华大学出版社网站(www.tup.com.cn)或苏州大学飞思卡尔嵌入式系统研究中心网站(sumcu.suda.edu.cn)下载。

本书可供大学有关专业的高年级学生和研究生用作教材或参考读物,也可供嵌入式系统开发与研究人员用作研发参考资料。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

嵌入式技术基础与实践/王宜怀等著.--2版.--北京:清华大学出版社,2011

(软件工程专业核心课程系列教材)

ISBN 978-7-302-24913-9

I. ①嵌… II. ①王… III. ①微处理器—系统设计—教材 IV. ①TP332

中国版本图书馆CIP数据核字(2011)第036932号

责任编辑:魏江江

责任校对:梁毅

责任印制:何芊

出版发行:清华大学出版社

<http://www.tup.com.cn>

社总机:010-62770175

投稿与读者服务:010-62795954, jsjic@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

地 址:北京清华大学学研大厦A座

邮 编:100084

邮 购:010-62786544

印刷者:北京富博印刷有限公司

装订者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185×260 印 张:24.75 字 数:617千字

版 次:2011年3月第2版 印 次:2011年3月第1次印刷

印 数:1~3000

定 价:39.50元

产品编号:041216-01

前 言

嵌入式计算机系统简称为嵌入式系统,其概念最初源于传统测控系统对计算机的需求。随着以微处理器(MPU)为内核的微控制器(MCU)制造技术的不断进步,计算机领域在通用计算机系统与嵌入式计算机系统这两大分支分别得以发展。通用计算机已经在科学计算、事务管理、通信、日常生活等各个领域产生重要的影响。在后 PC 时代,嵌入式系统的广泛应用将是计算机发展的重要特征。一般来说,嵌入式系统的应用范围可以粗略地分为两大类:一类是电子系统的智能化(如工业控制、现代农业、家用电器、汽车电子、测控系统、数据采集等);另一类是计算机应用的延伸(如 MP4、手机、通信、网络、计算机外围设备等)。不论如何分类,嵌入式系统的技术基础是不变的,即要完成一个以 MCU 为核心的嵌入式系统应用产品设计,需要有硬件、软件及行业领域相关知识。但是,随着嵌入式系统中软件规模日益增大,对嵌入式底层驱动软件的封装提出了更高的要求,可复用性与可移植性受到特别的关注,嵌入式软件构件化开发方法逐步被业界所重视。

本书第一版以飞思卡尔 HC08 系列中 GP32(通用型)、JB8(USB 1.1)、GZ60(CAN)三个型号 MCU 为蓝本阐述嵌入式应用,已经印刷了 1 万多册。随着飞思卡尔 8 位 MCU 的升级及我们多年教学与开发的经验积累,本书以嵌入式硬件构件与底层软件构件设计为主线,按照嵌入式软件工程的要求,以 S08 系列微控制器中通用型号 MC9S08AW60、带 USB 2.0 接口 MC9S08JM60、带 CAN 总线接口 MC9S08DZ60 三个型号为蓝本阐述嵌入式系统的软件与硬件设计。2009 年,笔者撰写了《基于 32 位 ColdFire 构建嵌入式系统》一书,系统阐述和应用了嵌入式构件开发思想,本书秉承这些工作,按照“通用知识→芯片编程结构概要→基本编程方法→底层驱动构件封装→应用方法与举例”的思路,逐步阐述电子系统智能化嵌入式应用的软件与硬件设计。

本书特点

(1) 把握通用知识与芯片相关知识之间的平衡。书中对于嵌入式“通用知识”的基本原理,以应用为立足点,进行语言简洁、逻辑清晰的阐述,同时注意与芯片相关知识之间的衔接,使读者在更好地理解基本原理的基础上,理解芯片应用的设计,同时反过来,加深对通用知识的理解。

(2) 把握硬件与软件的关系。嵌入式系统是软件与硬件的综合体,嵌入式系统设计是一个软件、硬件协同设计的工程,不能像通用计算机那样,软件、硬件完全分开来看。特别是对电子系统智能化嵌入式应用来说,没有对硬件的理解就不可能写好嵌入式软件;同样,没有对软件的理解,也不可能设计好嵌入式硬件。因此,本书注重把握硬件知识与软件知识之间的关系。

(3) 对底层驱动进行构件化封装。书中对每个模块均给出根据嵌入式软件工程基本原则并按照构件化封装要求编制底层驱动程序,同时给出详细、规范的注释及对外接口,为实际应用提供底层构件,方便移植与复用,可以为读者进行实际项目开发节省大量时间。

(4) 设计合理的测试用例。书中所有源程序均经测试通过,并提供测试用例下载,避免了因例程的书写或固有错误给读者带来烦恼。这些测试用例,也为读者验证与理解带来方便。

(5) 提供了所有模块完整的底层驱动构件化封装程序、文档与测试用例,同时还包含芯片参考手册、写入器安装与使用方法、工具软件(如开发环境、程序写入与读出软件、串口调试工具、USB工具、以太网工具)、有关硬件原理图及其他技术资料。

(6) 提供硬件评估版、写入调试器,并给出单独进行程序写入与读出的软件工具,方便读者进行实践与应用。

主要内容

全书共14章,其中第1章为概述,阐述嵌入式系统的知识体系、学习误区、学习建议,以及嵌入式系统常用术语、嵌入式系统常用的C语言基本语法概要。第2章给出AW60硬件最小系统,并简要介绍S08CPU。第3章给出第一个样例程序及CodeWarrior工程组织,完成第一个S08工程的入门。第4章介绍基于硬件构件的嵌入式系统开发方法。第5章阐述串行通信接口SCI,并给出第一个带中断的编程实例。第1~5章完成了学习一个新MCU完整要素的入门。第6~12章分别给出键盘、LED、LCD、定时器、SPI、Flash存储器在线编程、I²C、A/D及AW60芯片的其他模块等。第13、14章分别给出JM60的USB 2.0编程及DZ60的CAN总线编程。本书提供了所有实例源程序、辅助资料、相关芯片资料及常用软件工具,可在清华大学出版社网站(www.tup.com.cn)或苏州大学飞思卡尔嵌入式系统研发中心网站(sumcu.suda.edu.cn)下载。附录A介绍了Freescale MCU常用封装形式与规格;附录B与C给出了AW60映像寄存器列表;附录D介绍了S08/S12/ColdFire BDM简明使用方法;附录E给出了S08的C语言函数库;附录F给出了常见实践问题解答。

致谢

本书除封面署名作者外,还有苏州大学计算机科学与技术学院嵌入式应用方向研究生陈爱兵、秦保波、周杰、姚丹丹、李翠霞、程玉娟、顾霞萍等协助书稿编写、整理及程序调试工作,他们卓有成效的工作,使本书更加实用。飞思卡尔半导体有限公司的Andy Mastronardi先生、马莉女士一直关心支持苏州大学飞思卡尔嵌入式系统研发中心的建设,为本书的撰写提供了硬件及软件资料,并提出了许多宝贵建议。飞思卡尔半导体有限公司的许多技术人员提供了技术支持。清华大学出版社的魏江江先生为本书的出版做出了大量细致的工作。在此一并表示诚挚的谢意。

鉴于作者水平有限,书中难免存在不足和错误之处,恳望读者提出宝贵意见和建议,以便再版时改进。

王宜怀

2010年11月于苏州大学

目 录

第 1 章 概述	1
1.1 嵌入式系统定义、由来及特点.....	1
1.1.1 嵌入式系统的定义.....	1
1.1.2 嵌入式系统的由来及其与微控制器的关系.....	2
1.1.3 嵌入式系统的特点.....	3
1.2 嵌入式系统的知识体系、学习误区及学习建议.....	4
1.2.1 嵌入式系统的知识体系.....	4
1.2.2 嵌入式系统的学习误区.....	5
1.2.3 基础阶段的学习建议.....	8
1.3 嵌入式系统常用术语.....	9
1.3.1 与硬件相关的术语.....	9
1.3.2 与通信相关的术语.....	10
1.3.3 与功能模块及软件相关的术语.....	11
1.4 嵌入式系统常用的 C 语言基本语法概要.....	12
习题 1.....	23
第 2 章 AW60 硬件最小系统与 S08 CPU	24
2.1 S08 系列 MCU 命名规则与资源简介.....	24
2.2 AW60 微控制器功能概述及存储器映像.....	25
2.2.1 AW60 功能概述.....	26
2.2.2 AW60 存储器映像.....	27
2.3 AW60 的引脚功能与硬件最小系统.....	29
2.3.1 AW60 的引脚功能.....	29
2.3.2 AW60 的硬件最小系统.....	31
2.3.3 硬件最小系统测试方法.....	33
2.4 S08 CPU 的特点与 CPU 内部寄存器.....	34
2.5 S08 CPU 的寻址方式.....	36
2.6 S08 CPU 的指令系统.....	40
2.6.1 数据传送类指令.....	41
2.6.2 算术运算类指令.....	44
2.6.3 逻辑运算类指令.....	48
2.6.4 位操作类指令.....	49
2.6.5 移位类指令.....	49

2.6.6	程序控制类指令	51
2.6.7	其他指令	53
2.7	S08 CPU 汇编语言基础	54
2.7.1	S08 汇编源程序格式	55
2.7.2	S08 的伪指令	57
习题 2	60
第 3 章	第一个样例程序及 CodeWarrior 工程组织	61
3.1	通用 I/O 接口基本概念及连接方法	61
3.2	AW60 的 GPIO	62
3.2.1	AW60 GPIO 编程的基本原理	62
3.2.2	GPIO 模块寄存器与 GPIO 编程的基本方法	63
3.3	开发套件 CodeWarrior 开发环境与 S08/S12/ColdFire 写入器	64
3.3.1	CodeWarrior 开发环境简介与基本使用方法	65
3.3.2	S08/S12/ColdFire 三合一写入器	66
3.3.3	SD-AW60EVB 硬件评估板	67
3.4	CW 环境 C 语言工程文件的组织	67
3.4.1	工程文件的逻辑组织结构	68
3.4.2	工程文件的物理组织结构	69
3.4.3	系统启动及初始化相关文件	70
3.4.4	芯片初始化、主程序、中断程序及其他文件	76
3.4.5	机器码文件(.s19 文件)的简明解释	79
3.4.6	.lst 文件与.map 文件	81
3.4.7	如何在 CW 环境下新建一个 S08 工程	82
3.5	第一个 C 语言工程: 控制小灯闪烁	83
3.5.1	GPIO 构件	83
3.5.2	Light 构件	86
3.5.3	Light 测试工程主程序	89
3.5.4	理解第一个 C 工程的执行过程	90
3.6	第一个汇编语言工程: 控制小灯闪烁	91
3.6.1	汇编工程文件的组织	91
3.6.2	Light 构件汇编程序	95
3.6.3	Light 测试工程主程序	96
3.6.4	理解第一个汇编工程的执行过程	97
习题 3	97
第 4 章	基于硬件构件的嵌入式系统开发方法	99
4.1	嵌入式系统开发所遇到的若干问题	99
4.2	嵌入式硬件构件的基本思想与应用方法	100

4.3	基于硬件构件的嵌入式系统硬件电路设计	100
4.3.1	设计时需要考虑的基本问题	100
4.3.2	硬件构件化电路原理图绘制的简明规则	101
4.3.3	实验 PCB 板设计的简明规则	103
4.4	基于硬件构件的嵌入式底层软件构件的编程方法	106
4.4.1	嵌入式硬件构件和软件构件的层次模型	106
4.4.2	底层构件的实现方法与编程思想	106
4.4.3	硬件构件及底层软件构件的重用与移植方法	108
	习题 4	110
第 5 章	串行通信接口 SCI	111
5.1	异步串行通信的通用基础知识	111
5.1.1	串行通信的基本概念	111
5.1.2	RS-232 总线标准	113
5.1.3	TTL 电平到 RS-232 电平转换电路	114
5.1.4	串行通信编程模型	115
5.2	AW60 的 SCI 模块的编程结构	116
5.3	AW60 的 SCI 构件设计与测试	121
5.3.1	SCI 构件设计概述	121
5.3.2	SCI 构件的头文件 SCI.h	122
5.3.3	SCI 构件的 C 语言源程序文件 SCI.c	124
5.3.4	SCI 构件的测试工程	128
5.4	AW60 的中断源与第一个带有中断的编程实例	131
5.4.1	中断处理的相关基本概念	131
5.4.2	S08 CPU 的中断实现过程	132
5.4.3	AW60 的中断源与中断向量表	133
5.4.4	AW60 的中断编程方法	134
5.4.5	AW60 的中断编程示例	136
	习题 5	138
第 6 章	GPIO 的应用实例——键盘、LED 与 LCD	140
6.1	键盘编程实例	140
6.1.1	未编码键盘的接线方式	140
6.1.2	键盘编程的基本问题	141
6.1.3	AW60 键盘中断模块	141
6.1.4	键盘编程实例	143
6.2	数码管 LED 编程实例	151
6.2.1	扫描法编程原理	151
6.2.2	数码管编程实例	152

6.3	液晶 LCD 编程实例	156
6.3.1	点阵字符型 LCD 的基本特点	156
6.3.2	HD44780 概述	157
6.3.3	点阵字符型 LCD 编程实例	160
	习题 6	166
第 7 章	定时器模块	167
7.1	计数器/定时器的基本工作原理	167
7.1.1	实现计数与定时的基本方法	167
7.1.2	AW60 的定时接口的功能框图	168
7.2	定时器模块的编程结构与编程实例	169
7.2.1	定时器模块的编程寄存器	169
7.2.2	定时溢出中断构件与编程实例	172
7.3	定时器模块的输入捕捉功能	178
7.3.1	输入捕捉的基本含义	178
7.3.2	输入捕捉的寄存器	179
7.3.3	输入捕捉中断编程实例	180
7.4	定时器模块的输出比较功能	184
7.4.1	输出比较的基本知识	185
7.4.2	用于输出比较的寄存器	186
7.4.3	输出比较编程实例	186
7.5	定时器模块的脉宽调制(PWM)输出功能	191
7.5.1	脉宽调制器 PWM 工作原理	192
7.5.2	AW60 定时器的两种 PWM 模式	192
7.5.3	PWM 编程实例	194
	习题 7	198
第 8 章	串行外设接口 SPI	199
8.1	SPI 的基本工作原理	199
8.1.1	SPI 基本概念	199
8.1.2	SPI 的数据传输	200
8.1.3	SPI 模块的时序	201
8.1.4	模拟 SPI	202
8.2	AW60 SPI 模块编程结构	203
8.2.1	AW60 的 SPI 模块概述	203
8.2.2	SPI 模块寄存器	203
8.3	AW60 的 SPI 构件设计	207
8.3.1	SPI 构件设计概述	207
8.3.2	SPI 构件的源程序文件	207

8.4 利用 SPI 扩展外接 Flash 芯片	209
8.4.1 Flash 芯片 AT45DB041D 概述	209
8.4.2 AT45DB041D 操作说明	210
8.4.3 扩展 Flash 的程序实现	211
习题 8	214
第 9 章 Flash 存储器在线编程	216
9.1 Flash 存储器概述	216
9.1.1 Flash 存储器的基本特点与编程模式	217
9.1.2 S08 系列 MCU 的 Flash 存储器	217
9.2 AW60 的 Flash 存储器编程方法	218
9.2.1 Flash 存储器编程的基本概念	218
9.2.2 Flash 存储器的编程寄存器	219
9.2.3 Flash 存储器的编程过程	222
9.3 AW60 的 Flash 在线编程 C 语言实例	223
9.3.1 Flash 存储器的擦除及写入 C 语言子程序	223
9.3.2 Flash 存储器在线编程擦写测试工程	226
9.4 S08 Flash 存储器的保护特性和安全性	227
9.4.1 S08 Flash 存储器的保护特性	227
9.4.2 S08 Flash 存储器的安全性	229
习题 9	233
第 10 章 集成电路互连总线 I²C	234
10.1 I ² C 总线通用知识	234
10.1.1 I ² C 总线的历史概况与特点	234
10.1.2 I ² C 总线的硬件相关术语与典型电路	235
10.1.3 I ² C 总线数据通信协议概要	236
10.1.4 主机向从机读/写 1 个字节数据的过程	238
10.2 AW60 的 I ² C 模块编程结构	239
10.2.1 AW60 的 I ² C 模块概述	239
10.2.2 AW60 的 I ² C 模块寄存器	240
10.3 AW60 的 I ² C 模块底层驱动构件设计	244
10.4 AW60 的 I ² C 构件应用实例——与 PCF8563 时钟芯片通信	248
10.4.1 PCF8563 芯片简介	248
10.4.2 AW60 的 I ² C 与 PCF8563 时钟芯片通信	251
10.5 关于 I ² C 的进一步讨论	254
10.5.1 仲裁程序	254
10.5.2 数据传输同步交换	255
10.5.3 SPI 与 I ² C 的比较	255

习题 10	256
第 11 章 模数转换模块	257
11.1 A/D 转换通用知识	257
11.1.1 A/D 转换的基本问题	257
11.1.2 A/D 转换常用传感器简介	258
11.1.3 电阻型传感器采样电路设计	259
11.2 AW60 的 A/D 转换模块编程结构	260
11.2.1 AW60 的 A/D 转换结构与特性	260
11.2.2 AW60 的 A/D 模块编程寄存器	261
11.2.3 功能描述	265
11.2.4 AW60 的内置温度传感器	269
11.3 AW60 的 A/D 模块底层驱动构件设计	269
11.3.1 AD 构件的函数原型设计	269
11.3.2 AD 构件的头文件	269
11.3.3 AD 构件的源程序文件	271
11.4 AW60 A/D 模块编程实例	274
11.4.1 低端软件设计	274
11.4.2 高端软件设计	275
习题 11	276
第 12 章 AW60 芯片的其他模块	277
12.1 AW60 的工作模式与低功耗	277
12.2 内部时钟发生器模块 ICG	279
12.2.1 AW60 的 ICG 结构	279
12.2.2 ICG 操作模式	281
12.2.3 ICG 的寄存器与运行机制	282
12.2.4 AW60 的初始化方法与示例	283
12.3 AW60 的复位	287
12.3.1 复位	287
12.3.2 低电压检测(LVD)复位	288
12.3.3 看门狗复位	289
12.3.4 其他寄存器	291
12.4 AW60 的 $\overline{\text{IRQ}}$ 引脚、RTI、BRK 及 SWI 中断	292
12.4.1 $\overline{\text{IRQ}}$ 引脚中断	292
12.4.2 实时中断(RTI)模块	293
12.4.3 断点模块 BRK 与软件中断 SWI 指令	294
习题 12	294

第 13 章 JM60 的 USB 2.0 开发方法	295
13.1 USB 基本概念及硬件特性	295
13.1.1 USB 概述	295
13.1.2 USB 相关基本概念	297
13.1.3 USB 的物理特性	302
13.2 USB 的通信协议	303
13.2.1 USB 基本通信包	304
13.2.2 USB 通信中的事务处理	306
13.2.3 从设备的枚举看 USB 数据传输	307
13.3 JM60 硬件最小系统及 USB 模块功能简介	311
13.3.1 JM60 硬件最小系统	311
13.3.2 JM60 的 USB 模块功能简介	312
13.3.3 JM60 的 USB 模块主要寄存器介绍	314
13.4 JM60 作为 USB 从机的开发方法	316
13.4.1 PC 方 USB 设备驱动程序的选择及基本原理	316
13.4.2 PC 作为 USB 主机的程序设计	320
13.4.3 JM60 作为 USB 从机的程序设计	321
习题 13	325
第 14 章 DZ60 的 CAN 总线开发方法	327
14.1 CAN 总线通用知识	327
14.1.1 CAN 总线协议的历史概况	327
14.1.2 CAN 硬件系统的典型电路	327
14.1.3 CAN 总线的有关基本概念	329
14.1.4 帧结构	332
14.1.5 位时间	336
14.2 DZ60 硬件最小系统	337
14.3 MSCAN 模块概述与编程结构	340
14.3.1 MSCAN 的特性	340
14.3.2 运行模式	341
14.3.3 低功耗选项	341
14.3.4 中断	344
14.3.5 MSCAN 模块寄存器定义	345
14.4 MSCAN 功能描述	353
14.4.1 报文发送/接收功能描述	353
14.4.2 报文发送基础	353
14.4.3 发送结构	354
14.4.4 接收结构	355

14.4.5 时钟系统	355
14.5 MSCAN 初始化、报文发送与接收的函数设计	357
14.5.1 MSCAN 初始化过程	357
14.5.2 MSCAN 报文发送/接收过程	359
14.5.3 MSCAN 测试实例	362
习题 14	364
附录 A Freescale MCU 常用封装形式与规格	365
附录 B AW60 直接页寄存器列表	366
附录 C AW60 高页寄存器列表	369
附录 D S08/S12/ColdFire BDM 简明使用方法	371
附录 E S08 的 C 语言函数库	375
附录 F 常见实践问题解答	379
参考文献	381

第1章 概述

作为全书导引,本章主要内容有:①简要给出嵌入式系统定义、由来及特点;②简要阐述嵌入式系统的知识体系,分析如何避免进入嵌入式系统的学习误区,根据嵌入式系统的特点,就如何学习嵌入式系统提出几点建议;③归纳嵌入式系统的常用术语;④给出嵌入式系统常用的C语言基本语法概要。素材的【第1章(概述)阅读资料】中还补充给出了嵌入式C语言工程简明规范与嵌入式C++语言的基本语法概要。

1.1 嵌入式系统定义、由来及特点

1.1.1 嵌入式系统的定义

嵌入式系统(Embedded System)有多种多样的定义,但本质是相同的。本书关于嵌入式系统的定义取自美国CMP Books出版的Jack Ganssle和Michael Barr著作“Embedded System Dictionary”^①。

嵌入式系统的定义:一种计算机硬件和软件的组合,也许还有机械装置,用于实现一个特定功能。在某些特定情况下,嵌入式系统是一个大系统或产品的一部分。世界上第一个嵌入式系统是1971年Busicom公司用Intel单芯片4004微处理器完成的商用计算器系列。该词典还给出了嵌入式系统的一些示例:微波炉、手持电话、计算器、数字手表、录像机、巡航导弹、GPS接收机、数码相机、传真机、跑步机、遥控器和谷物分析仪等,难以尽数。通过与通用计算机的对比可以更形象地理解嵌入式系统的定义。该词典给出的通用计算机定义是:计算机硬件和软件的组合,用作通用计算平台。PC、MAC和UNIX工作站是最流行的现代计算机。

我国《国家标准GB/T 5271 信息技术词汇——嵌入式系统与单片机》部分,给出的嵌入式系统定义是:置入应用对象内部起操作控制作用的专用计算机系统。

国内对嵌入式系统定义曾进行过广泛讨论,有许多不同说法。其中嵌入式系统定义的涵盖面问题是主要争论焦点之一。例如,有的学者认为不能把手持电话叫嵌入式系统,而只能把其中起控制作用的部分叫嵌入式系统,而手持电话可以称为嵌入式系统的应用产品。其实,这些并不妨碍人们对嵌入式系统的理解,所以不必对定义感到困惑。有些国内学者特别指出,在理解嵌入式系统定义时,不要把嵌入式系统与嵌入式系统产品相混淆。实际上,从口语或书面语言角度,不区分“嵌入式系统”与“嵌入式系统产品”,只要不妨碍对嵌入式系统的理解就没有关系。

为了更清楚地阐述嵌入式系统特点,首先介绍大多数嵌入式系统的核心部件——MCU

^① 中译本: Jack Ganssle等著,马广云等译,《英汉双解嵌入式系统词典》,北京航空航天大学出版社,2006年。

(微控制器)的基本概念。

1.1.2 嵌入式系统的由来及其与微控制器的关系

1. MCU(微控制器)的基本含义

MCU 是单片微型计算机(单片机)的简称,早期的英文名是 Single-chip Microcomputer,后来大多数人称之为微控制器(Microcontroller)或嵌入式计算机(Embedded Computer)。现在 Microcontroller 已经是计算机中一个常用术语,但在 1990 年之前,大部分英文词典并没有这个词。我国学者一般使用中文“单片机”一词,而缩写使用“MCU”^①。所以本书后面的简写一律以 MCU 为准。MCU 的基本含义是:在一块芯片上集成了中央处理单元(CPU)、存储器(RAM/ROM 等)、定时器/计数器及多种输入输出(I/O)接口的比较完整的数字处理系统。图 1-1 给出了典型的 MCU 组成框图。

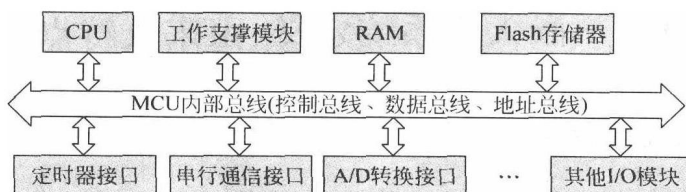


图 1-1 一个典型的 MCU 内部框图

MCU 是在计算机制造技术发展 to 一定阶段的背景下出现的,它使计算机技术从科学计算领域进入到智能化控制领域。从此,计算机技术在两个重要领域——通用计算机领域和嵌入式(Embedded)计算机领域都获得了极其重要的发展,为计算机的应用开辟了更广阔的空间。

就 MCU 组成而言,虽然它只是一块芯片,但包含了计算机的基本组成单元,仍由运算器、控制器、存储器、输入设备、输出设备五部分组成,只不过这些都集成在一块芯片上,这种结构使 MCU 成为具有独特功能的计算机。

2. 嵌入式系统的由来

通俗地说,计算机是因科学家需要一个高速的计算工具而产生的。直到 20 世纪 70 年代,电子计算机在数字计算、逻辑推理及信息处理等方面表现出非凡的能力。在通信、测控与数据传输等领域,人们对计算机技术给予了更大的期待。这些领域的应用与单纯的高速计算要求不同,主要表现在:直接面向控制对象;嵌入到具体的应用体中,而不是以计算机的面貌出现;能在现场连续可靠地运行;体积小,应用灵活;突出控制功能,特别是对外部信息的捕捉与丰富的输入输出功能等。由此可以看出,满足这些要求的计算机与满足高速数值计算的计算机是不同的。因此,一种称为 MCU 或微控制器的技术得以产生并发展。为了区分这两种类型的计算机,通常把满足海量高速数值计算的计算机称为通用计算机系统,而把嵌入到实际应用系统中,实现嵌入式应用的计算机称为嵌入式计算机系统,简称嵌入式系统。

^① MCU 的英文全称是 Microcontroller Unit。

3. 嵌入式系统与 MCU 的关系

何立民先生说：“有些人搞了十多年的 MCU 应用，不知道 MCU 就是一个最典型的嵌入式系统”^①。实际上，MCU 是在通用 CPU 基础上发展起来的，MCU 具有体积小、价格低、稳定可靠等优点，它的出现和迅猛发展，是控制系统领域的一场技术革命。MCU 以其较高的性能价格比、灵活性等特点，在现代控制系统中具有十分重要的地位。大部分嵌入式系统以 MCU 为核心进行设计。MCU 从体系结构到指令系统都是按照嵌入式系统的应用特点专门设计的，它能很好地满足应用系统的嵌入、面向测控对象、现场可靠运行等方面的要求。因此以 MCU 为核心的系统是应用最广的嵌入式系统。在实际应用时，开发者可以根据具体要求与应用场合，选用最佳型号的 MCU 嵌入到实际应用系统中。

在 MCU 出现之前，人们必须用模拟电路、数字电路实现大部分计算与控制功能，这样使得控制系统体积庞大，易出故障。MCU 出现以后，情况发生了变化，系统中的大部分计算与控制功能由 MCU 的软件实现。其他电子线路成为 MCU 的外围接口电路，承担着输入、输出与执行动作等功能，而计算、比较与判断等原来必须用电路实现的功能，可以用软件取代，大大地提高了系统的性能与稳定性，这种控制技术称为嵌入式控制技术。在嵌入式控制技术中，核心是 MCU，其他部分依此而展开。

1.1.3 嵌入式系统的特点

嵌入式系统的特点，不同学者也许有不同的说法。这里从与通用计算机对比的角度谈嵌入式系统的特点。

1. 嵌入式系统属于计算机系统，但不单独以通用计算机的面目出现

嵌入式系统的本名叫嵌入式计算机系统(Embedded Computer System)，它不仅具有通用计算机的主要特点，又具有自身特点。嵌入式系统也必须有软件才能运行，但其隐含在种类众多的具体产品中。同时，通用计算机种类屈指可数，而嵌入式系统不仅芯片种类繁多，而且由于应用对象大小各异，嵌入式系统作为控制核心，已经融入到各个行业的产品之中。

2. 嵌入式系统开发需要专用工具和特殊方法

嵌入式系统不像通用计算机那样有了计算机系统就可以进行应用开发。一般情况下，MCU 芯片本身不具备开发功能，必须要有一套与相应芯片配套的开发工具和开发环境。这些工具和环境一般基于通用计算机上的软硬件设备以及各种逻辑分析仪、混合信号示波器等。开发时往往有主机和目标机的概念，主机用于程序的开发，目标机作为程序的执行机，开发时需要交替结合进行。

3. 使用 MCU 设计嵌入式系统，数据与程序空间采用不同的存储介质

在通用计算机系统中，程序存储在硬盘上。实际运行时，通过操作系统将要运行的程序从硬盘调入内存(RAM)，运行中的程序、常数、变量均在 RAM 中。而以 MCU 为核心的嵌入式系统，其程序被固化到非易失性存储器中^②。变量及堆栈使用 RAM 存储器。

^① 详见《单片机与嵌入式系统应用》，2004 年第 1 期。

^② 目前，非易失性存储器通常为 Flash 存储器，特点见有关“Flash 存储器在线编程”章节。

4. 开发嵌入式系统涉及软件、硬件及应用领域的知识

嵌入式系统与硬件紧密相关,嵌入式系统的开发需要硬件、软件协同设计、协同测试。同时,由于嵌入式系统专用性很强,通常是用在特定应用领域,如嵌入在手机、冰箱、空调、各种机械设备、智能仪器仪表中,起核心控制作用,功能专用。因此,进行嵌入式系统的开发,还需要对专业领域知识有一定的理解。当然,一个团队协作开发一个嵌入式产品,虽然各个成员可以扮演不同角色,但对系统的整体理解与把握及相互协作,有助于一个稳定可靠嵌入式产品的诞生。

5. 嵌入式系统的其他特点

除了以上特点之外,嵌入式系统还具有其他方面的特点。

(1) **在资源方面:** 嵌入式系统通常专用于某一特定应用领域,其硬件资源不像通用计算机那样丰富;

(2) **在可靠性方面:** 嵌入式系统一般要求更高的可靠性和稳定性;

(3) **在实时性方面:** 相当多的嵌入式系统有实时性要求;

(4) **在成本方面:** 嵌入式系统通常极其关注成本;

(5) **在功耗要求方面:** 一些嵌入式系统要求低功耗;

(6) **在生命周期方面:** 嵌入式系统通常比通用计算机系统生命周期长,升级换代比通用计算机慢;

(7) **在知识综合方面:** 嵌入式系统是将先进的计算机技术、半导体技术及电子技术与各个行业的具体应用相结合的产物,是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。它的构成既有硬件又有软件,不仅包括应用软件,也可能包括系统软件。它有数字电路又有模拟电路。其产品技术含量高,涉及多种学科,不容易开发,因此也不容易形成技术垄断。

这些特点决定了嵌入式系统的开发方法、开发难度、开发手段等均不同于通用计算机,也不同于常规的电子产品。

1.2 嵌入式系统的知识体系、学习误区及学习建议

1.2.1 嵌入式系统的知识体系

嵌入式系统的应用范围可以粗略地分为两大类:

(1) 电子系统的智能化(工业控制、现代农业、家用电器、汽车电子、测控系统、数据采集等);

(2) 计算机应用的延伸(MP3、手机、通信、网络、计算机外围设备等)。

从这些应用可以看出,要完成一个以MCU为核心的嵌入式系统应用产品设计,需要有硬件、软件及行业领域相关知识。硬件主要有MCU的硬件最小系统、输入输出外围电路、人机接口设计。软件设计有固化软件的设计,也可能含PC软件的设计。行业知识需要通过协作、交流与总结获得。

概括地说,学习以MCU为核心的嵌入式系统,需要以下软、硬件基础知识与实践训练: