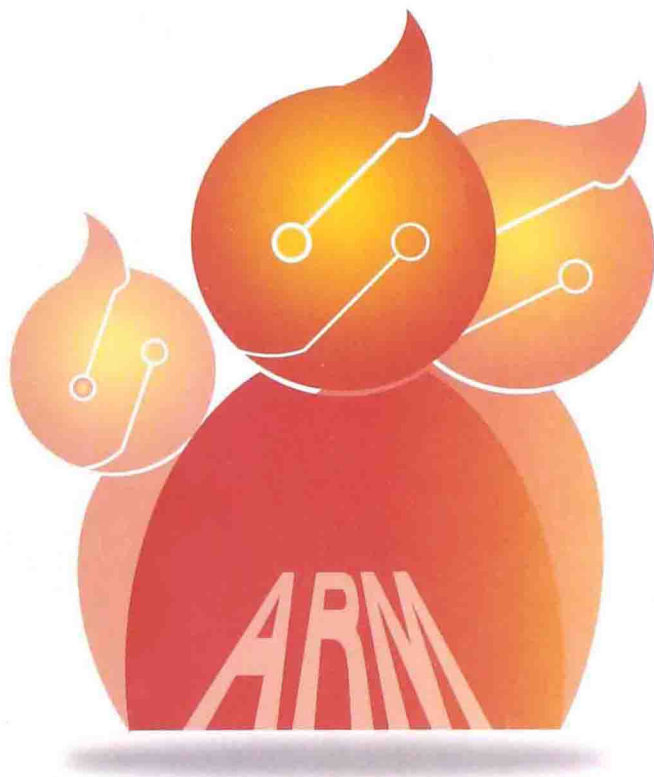




电子设计与嵌入式开发实践丛书

国内首本系统论述ARM Cortex-M0+内核LPC800系列的著作！

按照“认识—应用—改造”的思路，循序渐进地带您进入ARM Cortex-M0+的开发之路！



ARM Cortex-M0+

嵌入式开发与实践

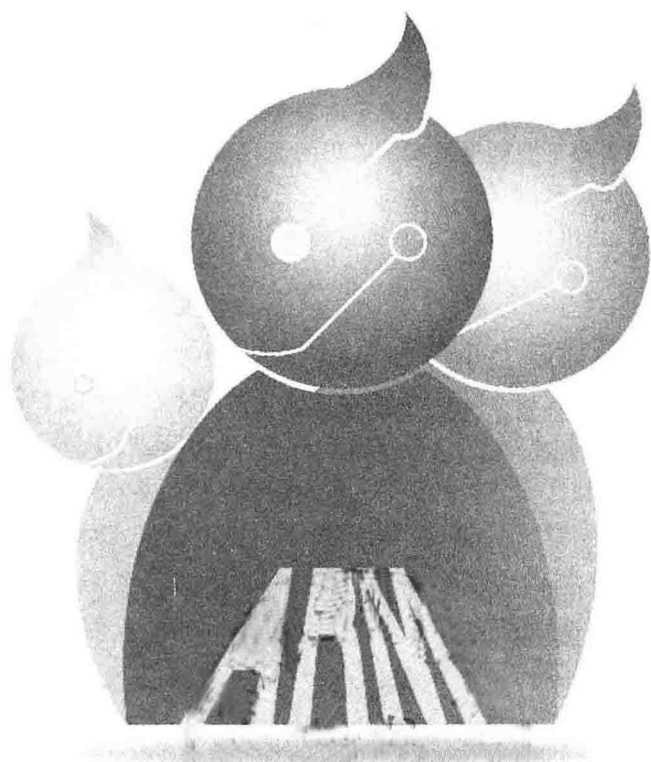
——基于LPC800

张 勇 编著

清华大学出版社



 电子设计与嵌入式开发实践丛书



ARM Cortex-M0+

嵌入式开发与实践

——基于LPC800

张 勇 编著

清华大学出版社
北 京

内 容 简 介

基于 ARM Cortex-M0+内核的 NXP LPC8XX 系列微控制器是替换传统单片机的最佳选择之一,它具有丰富的片上外设和强大的处理核心。本书基于 NXP LPC812 微控制器和 Keil MDK 集成开发环境,讲述 LPC812 硬件设计系统以及芯片级别与 $\mu\text{C}/\text{OS-II}$ 系统级别的软件设计方法。全书共分为 14 章,包括 LPC812 芯片架构、LPC812 典型开发电路系统、LPC812 芯片级程序框架、定时器、通用 I/O 口、串口、外部中断、按键与数码管驱动、温度传感器 DS18B20、点阵式 LCD 屏、 $\mu\text{C}/\text{OS-II}$ 用户任务管理与程序框架、 $\mu\text{C}/\text{OS-II}$ 信号量与消息邮箱以及 $\mu\text{C}/\text{OS-II}$ 系统定时器等。本书理论与应用结合紧密,实例丰富,读者可在清华大学出版社网站本书页面下载到全部实例工程的源代码。

本书可作为高等院校电子信息、通信、计算机、软件工程、自动控制、智能仪器和物联网等相关专业的高年级本科生或研究生教材,也可作为嵌入式系统爱好者和开发人员的参考用书。

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

ARM Cortex-M0+嵌入式开发与实践:基于 LPC800/张勇编著.--北京:清华大学出版社,2014
(电子设计与嵌入式开发实践丛书)

ISBN 978-7-302-36880-9

I. ①A… II. ①张… III. ①微处理器—系统设计 IV. ①TP332

中国版本图书馆 CIP 数据核字(2014)第 131366 号

责任编辑:刘 星

封面设计:迷底书装

责任校对:李建庄

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:18 字 数:449 千字

版 次:2014 年 9 月第 1 版 印 次:2014 年 9 月第 1 次印刷

印 数:1~3000

定 价:55.00 元

产品编号:060459-01

序言

在过去的几十年中,8 位微控制器擅长处理简单任务,且具备可靠、确定的性能。本书介绍的 LPC800 是首款恩智浦从一开始就针对应用单纯、具备实时性能的 8 位应用市场而设计的 32 位微控制器,它集 8 位的简单性、32 位的多功能性以及精简、高效、灵活、创新的外设于一身。

LPC800 基于超低功耗的 30MHz ARM Cortex-M0+ 处理器,提供优于 8/16 位架构的代码密度,允许对 GPIO 进行单周期访问,具有两级流水线,可在改善性能的同时降低功耗。

恩智浦还重新设计了 LPC800 系列外设,使其尽可能精简,响应更灵敏、高效。例如, SPI 作为从机工作时其频率与处理器时钟无关; I²C 可以基于地址匹配进行唤醒; 提供 I²C 和 UART 的 ROM 驱动程序,仅需简单的 API 调用即可操作等。

同时 LPC800 系列还有灵活、创新的外设,例如,灵活的开关矩阵外设(switch matrix)使得设计人员可以为几乎任一引脚分配片内外设 I/O,帮助缓解 PCB 走线拥挤的情况; 另一重要外设是状态可配置定时器(State Configuration Timer, SCT),简单的 2 个 16 位 PWM,具有 4 个捕获输入和 4 个匹配输出,其中每个匹配寄存器均有影子寄存器。同时融合了状态和事件的理念,用户可以创建复杂的计数、输出、输入和控制功能,实现照明、电源和其他定制应用,而不受固定功能的制约。开关矩阵外设和 SCT 还提供了配套的 GUI 的配置工具,以降低设计人员的使用难度,完成更好的控制。

LPC800 提供一系列深受 8 位系统开发人员喜爱的低引脚数封装,包括 SO20、TSSOP20、TSSOP16 和 DIP8 等。为了也满足可穿戴设备等领域的需求, LPC800 同时还提供了 XSON16 (3.2mm×2.5mm) 等小封装产品。更多的信息和详细资料可以从恩智浦 MCU 官方技术支持网站(www.lpcware.com)获取。

因此,很高兴看到张勇老师编著的这样一本 LPC800 图书问世。从最初的 8051、ARM7 到今天的 Cortex-M 处理器,张勇老师已经在嵌入式系统领域深耕多年,所以本书的内容涵盖了多个方面,相信无论您是新手还是熟练的开发人员,都可以从本书中找到有用的信息。希望通过本书的出版,使更多的读者掌握 LPC800 的开发技能,更好地促进 MCU 开发技术的普及和推广。

恩智浦半导体(上海)有限公司

MCU 市场经理 辛华峰

2014 年 6 月于北京

Preface

前言

传统的 8051 系列单片机由于具有硬件结构简单、编程操作方便以及芯片价格低廉等特点,长期以来被广泛地应用于各种控制和显示等嵌入式系统中。此外,单片机易于教学与推广,目前在普通高校中,几乎所有的电子工程和控制工程相关专业都设有单片机课程。随着科技的进步和人们对高智能性设备的喜爱和需求,传统单片机因其控制逻辑简单而在很多领域显得应用乏力。因此,近些年来,很多半导体公司推出了兼容 8051 系列传统单片机的新型增强性单片机,例如 TI 公司的 MSP430 系列、Renesas 公司的 RL78 系列、Atmel 公司的 megaAVR 单片机和 Silicon Labs 公司的 C8051F 系列混合型单片机等。新型的单片机具有存储空间大、代码效率高和执行速度快等优点,一定程度上缓解了单片机的应用衰退,但仍然无法从根本上改变单片机正在慢慢退出嵌入式应用系统的趋势。

ARM 公司出品了众多微处理器内核,包括目前市场上流行的 ARM7、ARM9 和 ARM11 内核,当前 ARM 公司主推的内核为 Cortex 系列内核。这个系列又分为 M 系、R 系和 A 系,其中 A 系是高性能系列,支持 ARM、Thumb 和 Thumb-2 指令集,主要针对带有操作系统的智能平板电脑;R 系为普通嵌入式内核,支持 ARM、Thumb 和 Thumb-2 指令集,用于嵌入式系统;M 系为低功耗系列,仅支持 Thumb-2 指令集,目前有 Cortex-M0、M0+、M1、M3 和 M4,用于需要快速中断的嵌入式实时应用系统中。

Cortex-M3 是最早推出的 Cortex 系列处理器内核,于 2004 年诞生,5 年以后,ARM 公司推出了可商用的 Cortex-M0 内核,之后,在 Cortex-M0 基础上推出了能耗进一步优化的 Cortex-M0+ 内核(ARM 公司公布的功耗数据为 $11.2\mu\text{W}/\text{MHz}$),被誉为全球功耗最低微处理器。

Cortex-M 系列内核中,M0 和 M0+ 系列主要针对控制领域,涵盖了传统 8051 系列单片机的应用领域,比传统 8051 系列单片机在处理速度、功耗、中断反应能力、编程与调试等诸多方面都有很大优势;M3 系列针对控制领域中的高端实时应用领域,具有控制和数字信号处理能力,除了可用于传统 8051 系列单片机的应用领域外,还可用于 DSP 处理器应用的领域;M4 系列主要针对高速控制、语音信号处理和数字信号处理领域,涵盖了传统网络控制芯片和 DSP 处理器的应用领域。

NXP(恩智浦)公司是全球最早推出 Cortex-M 系列内核微处理器产品的厂商之一,目前主要产品有以 LPC1115 微控制器为代表的 LPC11XX 和 LPC12XX 系列(Cortex-M0 内核)、以 LPC812 微控制器为代表的 LPC8XX 系列(Cortex-M0+ 内核)、以 LPC1788 微控制

Foreword

器为代表的 LPC13XX、LPC17XX 和 LPC18XX 系列(Cortex-M3 内核)、以 LPC4088 微控制器为代表的 LPC40XX 系列(Cortex-M4 内核)和以 LPC4357 微控制器为代表的 LPC43XX 系列(Cortex-M4 和 M0 双核心)。目前,NXP 公司是 Cortex-M 系列微处理器出品最多、型号最全和应用最广的公司之一。NXP 公司的 Cortex-M 系列芯片都体现了低功耗、易使用和高性能的特点。

本书将阐述基于 Cortex-M0+ 的 LPC812 微控制器的系统应用和程序设计方法。由于 LPC812 片内集成了端口配置矩阵单元,使得 LPC812 在电路设计上特别灵活,在产品升级换代时,只需要通过软件编程方式修改端口配置矩阵,而不需要重新设计电路板(类似于 FPGA 芯片)。并且,LPC812 还具有编程方便、处理速度快和控制能力强等特点,有些专家称 LPC812 是具有划时代标志特征的微控制器芯片。

希望上述内容能够回答很多读者关于“为什么学习 ARM Cortex-M0+ 内核微控制器”和“为什么要学习 LPC812 微控制器”等问题。

一、本书的简介与结构

本书包括 LPC812 芯片架构以及芯片级与操作系统级的程序设计等内容,概括如下:

- 第 1 章介绍 LPC812 芯片特点与内部结构;
- 第 2 章介绍基于 LPC812 的典型硬件开发系统;
- 第 3~7 章重点介绍基于 LPC812 片上外设的软件设计;
- 第 8~10 章介绍 LPC812 开发板的典型应用实例,如 ZLG7289B 键盘与数码管控制实例、温度传感器 DS18B20 应用实例和 LCD 屏显示技术实例等;
- 第 11~14 章介绍基于 $\mu\text{C}/\text{OS-II}$ 的工程程序设计方法,重点介绍用户任务管理、信号量与消息邮箱典型应用方法和 $\mu\text{C}/\text{OS-II}$ 系统定时器用法等。

第 2 章所给出的 LPC812 开发板典型应用电路是第 3~14 章程序设计的硬件基础。

二、本书的教学思路

本书根据作者的讲义整理扩充而成,理论课时为 48 学时,实验课时为 32 学时,开放实验课时为 16 学时。如果用作大学本科教材,则理论课时宜为 32~48 学时,建议讲述第 1~10 章内容,按书中章节顺序讲述;实验课时建议为 24~32 学时。书中第 8~10 章可用于课程设计,第 11~14 章内容用于面向有嵌入式操作系统基础的学生开展教学与讨论。针对教师教学研究活动,作者提供更多的交流和技术支持,读者可以通过微信(微信号: ZhangYongLPC)直接联系作者。

建议理论教学与实验教学同步进行。理论教学过程中,可设置 2~4 学时讨论课,或安排学生分组作学习交流主题报告。实验教学可设置 3~4 个基础性实验和 1~2 个设计性实验,可结合全国大学生电子设计大赛的题目开展拓展性实验工作,并且应以学生自己动手为主。

对于自学本书的嵌入式爱好者而言,要求至少具有数字电路、模拟电路、C 语言程序设计等课程的基础知识,并建议在学习过程中设计一套 LPC812 开发板配套学习。

本书的每个实例都是完整的,读者可以自行输入实例代码,或登录清华大学出版社网站
本书页面下载全部工程实例代码,也可通过 Email: zhangyong@jxufe. edu. cn 或 QQ:
850998740 向作者索取源代码。

三、本书的特色

本书具有以下四个方面的特色。

其一,详细讲解了基于 Cortex-M0+ 核心的 LPC812 微控制器的存储配置、异常(或中断)向量表以及片上各种外设,讲述了 LPC812 微控制器片上外设的访问方法。

其二,详细描述了基于 LPC812 的典型应用电路,这些电路涉及 LED 灯、串口、按键、蜂鸣器、数码管、JTAG(SWD)和 ISP 电路、复位电路、测温电路和 LCD 屏电路等。

其三,实例丰富,通过完整的实例详细阐述了芯片级和系统级的程序设计方法,对基于 LPC812 微控制器的嵌入式系统软件开发具有较强的指导作用。

其四,基于 LPC812 开发板结合工程实例对嵌入式实时操作系统 $\mu\text{C}/\text{OS-II}$ 的用户任务管理、信号量、消息邮箱和系统定时器等进行了详细的讲解,对学习和应用 $\mu\text{C}/\text{OS-II}$ 具有较好的可借鉴性。

四、致谢

感谢 NXP(恩智浦)公司为本书编写提供了开发板和集成开发环境。在本书写作过程中,NXP 的辛华峰、王朋朋和张宇等专家通过电话、E-mail 和微信等提供了大量技术支持,帮助解决了一系列技术难题,并审阅了本书初稿,提出了很多建设性意见,使得本书理论与工程实践能紧密结合,成为一本兼有教学、科研和工程应用价值的针对性很强的参考书。

作者要特别感谢张宇工程师,在作者进行从 LPC812 到 LPC4357 的长期教研过程中,张宇工程师给予了无私的帮助和鼓励,使作者得以将这方面的教学材料和技术资料出版成册。

作者还要感谢北京博创兴盛陆海军总经理对本书出版的关心和支持。

感谢同事夏家莉、陈滨、蔡鹏、黄坚、张志兵、吴文华和贾晓天等老师在繁忙的工作之余阅读了本书部分初稿,并提出了宝贵建议。

还要特别感谢阅读了作者已经出版的图书并反馈宝贵意见的读者,使得本书的写作能按照“认识—应用—改造”的思路进行,从而使自学门槛较以往的书大大降低。

由于作者水平有限,书中难免会有纰漏之处,敬请专家和读者批评指正。有兴趣的读者可发送邮件到 zhangyong@jxufe. edu. cn,与本书作者联系;也可以发送邮件到 liux@tup. tsinghua. edu. cn,与本书策划编辑联系。

五、免责声明

知识的发展和科技的进步是多元的。本书内容上广泛引用的知识点均罗列于参考文献中,主要为 LPC812 用户手册、LPC812 芯片手册、Cortex-M0+ 技术手册、嵌入式实时操作

系统 μ C/OS-II、Keil MDK 集成开发环境等内容,所有这些引用内容的知识产权归相关公司所有,作者保留其余内容的所有权利。本书内容仅用于教学目的,旨在推广 Cortex-M0+ 核心 LPC812 微控制器、 μ C/OS-II 和 Keil MDK 集成开发环境等,禁止任何单位或个人摘抄或扩充本书内容用于出版发行,严禁将本书内容用于商业场合。

张 勇

2014 年 5 月

目 录

第 1 章 认识 LPC812 芯片	1
1.1 LPC812 芯片	1
1.2 LPC812 引脚	2
1.3 在系统编程	4
1.4 I/O 口配置矩阵	7
1.5 LPC812 特点与内部结构	11
1.6 LPC812 存储配置	13
1.7 LPC812 中断	14
1.8 系统配置模块 SYSCON	18
1.9 本章小结	28
第 2 章 LPC812 开发平台	29
2.1 LPC812 核心电路	30
2.2 电源电路	30
2.3 LED 驱动电路	31
2.4 串口通信电路	31
2.5 用户按键与蜂鸣器电路	32
2.6 DS18B20 电路	33
2.7 ZLG7289B 电路	33
2.8 SWD、ISP 和复位电路	36
2.9 LCD 屏接口电路	37
2.10 本章小结	37
第 3 章 LPC812 时钟系统与程序框架	39
3.1 LPC812 开发环境	39
3.2 LPC812 启动代码文件	48
3.3 初始化 LPC812 系统时钟	54

Contents

3.4	LPC812 最小工程实例	57
3.5	LPC812 程序框架	61
3.6	本章小结	67
第4章	定时器	68
4.1	看门狗定时器	68
4.1.1	看门狗定时器寄存器	69
4.1.2	看门狗定时器工作过程	71
4.1.3	看门狗工程实例	73
4.2	系统节拍定时器	78
4.2.1	SysTick 定时器工作原理	78
4.2.2	SysTick 定时器工程实例	80
4.3	多速率定时器	84
4.3.1	MRT 工作原理	84
4.3.2	MRT 定时器工作实例	87
4.4	自唤醒定时器	91
4.5	状态可配置计数器	95
4.5.1	SCT 寄存器	95
4.5.2	SCT 工程实例一	109
4.5.3	SCT 工程实例二	112
4.6	本章小结	115
第5章	通用输入/输出口	116
5.1	I/O 口配置	116
5.2	通用目的输入/输出口	119
5.3	GPIO 应用实例	121
5.4	本章小结	123
第6章	串口	124
6.1	LPC812 串口工作原理	124
6.2	串口工作程序实例	132
6.3	串口中断程序实例	139
6.4	本章小结	142
第7章	外部中断与模式匹配	143
7.1	外部中断与模式匹配工作原理	143
7.2	LPC812 外部中断实例	153
7.2.1	下降沿触发外部中断实例	153
7.2.2	上升沿触发外部中断实例	158

7.2.3 双边沿触发外部中断实例	159
7.2.4 电平触发外部中断实例	161
7.3 LPC812 模式匹配实例	163
7.4 本章小结	167
第 8 章 ZLG7289B 模块按键与数码管	168
8.1 ZLG7289B 工作原理	168
8.2 ZLG7289B 程序实例	171
8.3 本章小结	183
第 9 章 温度传感器 DS18B20	184
9.1 DS18B20 工作原理	184
9.2 温度显示实例	186
9.3 本章小结	196
第 10 章 LCD 屏	197
10.1 SGX12864 点阵 LCD 显示屏驱动原理	198
10.2 字符、汉字与图形显示技术	205
10.3 SGX12864 显示实例	207
10.4 LPC812 显示缓存技术	211
10.5 本章小结	220
第 11 章 μC/OS- II 任务	221
11.1 μ C/OS- II 系统配置	221
11.2 μ C/OS- II 用户任务	228
11.3 μ C/OS- II 应用程序框架	230
11.4 本章小结	237
第 12 章 μC/OS- II 信号量	238
12.1 μ C/OS- II 信号量工作原理	238
12.2 μ C/OS- II 信号量程序实例	239
12.3 本章小结	246
第 13 章 μC/OS- II 消息邮箱	247
13.1 μ C/OS- II 消息邮箱工作原理	247
13.2 μ C/OS- II 消息邮箱程序实例	249
13.3 本章小结	258

第 14 章 $\mu\text{C}/\text{OS-II}$ 软定时器	259
14.1 $\mu\text{C}/\text{OS-II}$ 软定时器工作原理	259
14.2 $\mu\text{C}/\text{OS-II}$ 软定时器程序实例	260
14.3 本章小结	266
附录 A Keil MDK v5 集成开发环境	267
附录 B ASCII 码表	274
参考文献	276

认识 LPC812 芯片

本章介绍 LPC812 芯片的外形封装、内部结构和片上资源,这些内容是后续硬件电路和程序设计的基础。

1.1 LPC812 芯片

LPC812 微控制器芯片是 LPC800 微控制器家族的一员。LPC800 微控制器家族是 NXP(恩智浦)公司设计的基于 ARM Cortex-M0+内核的低功耗 32 位微控制器,工作频率可达 30MHz。目前,LPC800 家族成员情况如表 1-1 所示。

表 1-1 LPC800 家族成员情况表

型 号	Flash /KB	SRAM /KB	USART	I ² C	SPI	比较器	GPIO	封装
LPC810M021FN8	4	1	2	1	1	1	6	DIP8
LPC811M001JDH16	8	2	2	1	1	1	14	TSSOP16
LPC812M101JDH16	16	4	3	1	2	1	14	TSSOP16
LPC812M101JD20	16	4	2	1	1	1	18	SO20
LPC812M101JDH20	16	4	3	1	2	1	18	TSSOP20

表 1-1 中,Flash 即通常所说的闪存,是用于存储代码或常数的不易失型存储器;SRAM 表示静态随机可读/可写存储器;USART(Universal Synchronous/Asynchronous Receiver/Transmitter)即通用同步或异步串行收发器,USART 可实现 RS-232 标准的串口通信;I²C 总线是由串行数据(SDA)和串行时钟(SCL)两根总线组成的串行通信总线;SPI(Serial Peripheral Interface)总线指串行外设接口,一般由片选信号(CS)、串行位时钟(SCLK)、主器件数据输出或从器件数据输入(MOSI)和主器件数据输入或从器件数据输出(MISO)四根总线组成;GPIO 指通用目的输入/输出口,类似于单片机的 I/O 口。表 1-1 的封装一栏中,DIP8 是指双列直插式 8 脚,TSSOP16 指薄片缩小外形贴片 16 脚,SO20 指小外形贴片 20 脚。

LPC800 家族各个成员芯片内核相同，都是 ARM Cortex-M0+；存在差异的是片上外设和封装，如表 1-1 所示。本书的主角是表 1-1 中的 LPC812M101JDH20 芯片，它具有 16KB Flash、4KB SRAM、3 个 USART、1 个 I²C、2 个 SPI、1 个比较器和 18 个通用输入/输出，下文简称 LPC812，其外形如图 1-1 所示。本书内容也适用于 LPC800 家族的其他成员。

在图 1-1 中，左下角的凹形圆圈对应着芯片的第 1 号引脚，于是，下面一排引脚的编号从左到右依次为 1~10；上面一排引脚的编号从右到左依次为 11~20。

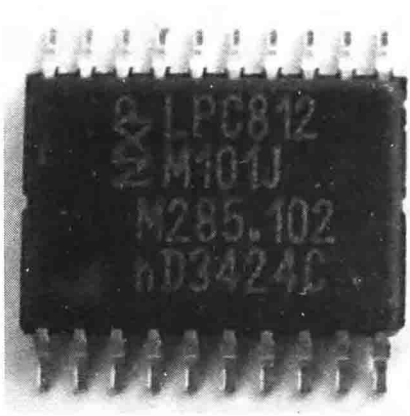


图 1-1 LPC812M101JDH20 芯片

1.2 LPC812 引脚

对应于图 1-1 的 LPC812(TSSOP20 封装)的引脚分布如图 1-2 所示，摘自 LPC81XM 芯片手册。

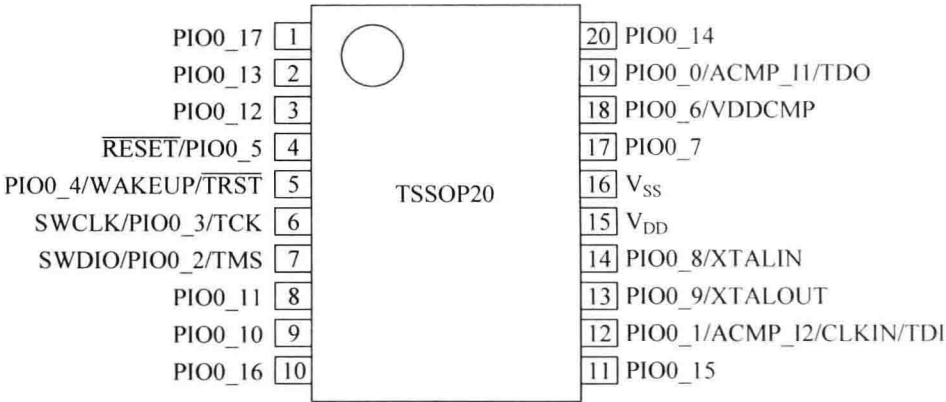


图 1-2 LPC812(TSSOP20 封装)芯片引脚分布图

图 1-2 与图 1-1 的引脚位置互相对应。不过习惯上，在借助 Altium Designer 制作原理图时会使用如图 1-3 所示的引脚分布图。

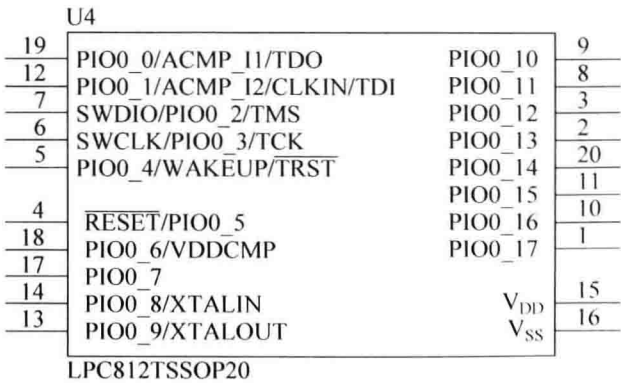


图 1-3 LPC812(TSSOP20 封装)芯片引脚分布图

在图 1-3 中,两侧的数字表示引脚号,例如,左上角的“19”表示该引脚是芯片的第 19 号引脚,引脚的名称对应于引脚号并且被标注在方框内。图 1-3 中的“U4”表示原理图中 LPC812 的符号。在图 1-3 中可以看到,LPC812 共有 18 个通用输入/输出口,即 PIO0_0~PIO0_17,并且 PIO0_0~PIO0_6、PIO0_8 和 PIO0_9 都复用了其他的功能(实际上,全部的 PIO0_0~PIO0_17 通过接口配置矩阵均可作为其他的功能引脚)。下面在表 1-2 中对图 1-3 中所有引脚的功能进行了详细说明,供编程时参考。

表 1-2 LPC812(TSSOP20 封装)芯片引脚功能表

引脚号	引脚名	类型	复位值	描 述
19	PIO0_0	I/O	I, PU	通用目的数字输入/输出口;在 ISP 模式下用作 U0_RXD(串口 0 的串行数据接收口);在仿真时用作 TDO(测试数据输出)
	ACMP_I1	AI	—	模拟比较器信号输入 1
12	PIO0_1	I/O	I, PU	通用目的数字输入/输出口;在仿真时用作 TDI(测试数据输入)
	ACMP_I2	AI	—	模拟比较器信号输入 2
	CLKIN	I	—	外部时钟输入
7	SWDIO	I/O	—	默认功能,串行调试数据输入/输出口;在仿真时,用作 TMS(测试模式选择)
	PIO0_2	I/O	I, PU	通用目的数字输入/输出口
6	SWCLK	I/O	—	默认功能,串行调试时钟;在仿真时,用作 TCK(测试时钟)
	PIO0_3	I/O	I, PU	通用目的数字输入/输出口
5	PIO0_4	I/O	I, PU	通用目的数字输入/输出口;在 ISP 模式下用作 U0_TXD(串口 0 的串行数据发送口);在仿真时用作 TRST(测试复位);若该引脚为高电平时 LPC812 进入深度掉电模式,则该引脚上 50ns 的低电平可将 LPC812 从深掉电中唤醒
4	RESET	I	—	该引脚上 50ns 的低电平将复位 LPC812,PC(程序计数器)从地址 0 开始执行
	PIO0_5	I/O	I, PU	通用目的数字输入/输出口
18	PIO0_6	I/O	I, PU	通用目的数字输入/输出口
	VDDCMP	AI	—	模拟比较器备用参考电压
17	PIO0_7	I/O	I, PU	通用目的数字输入/输出口
14	PIO0_8	I/O	I, PU	通用目的数字输入/输出口
	XTALIN	I	—	外部晶体振荡信号输入,电压不能超过 1.95V
13	PIO0_9	I/O	I, PU	通用目的数字输入/输出口
	XTALOUT	O	—	外部晶体振荡信号输出
9	PIO0_10	I	IA	通用目的数字输入/输出口;还用作 I ² C 时钟 SCL 总线(开路)
8	PIO0_11	I	IA	通用目的数字输入/输出口;还用作 I ² C 串行数据 SDA 总线(开路)
3	PIO0_12	I/O	I, PU	通用目的数字输入/输出口;还用作 ISP(在系统编程)控制口
2	PIO0_13	I/O	I, PU	通用目的数字输入/输出口
20	PIO0_14	I/O	I, PU	通用目的数字输入/输出口

续表

引脚号	引脚名	类型	复位值	描 述
11	PIO0_15	I/O	I, PU	通用目的数字输入/输出口
10	PIO0_16	I/O	I, PU	通用目的数字输入/输出口
1	PIO0_17	I/O	I, PU	通用目的数字输入/输出口
15	VDD	—	—	3.3V 电源输入
16	VSS	—	—	地

表 1-2 中,“类型”一栏中的 I/O 表示数字输入/输出,I 表示数字输入,O 表示数字输出,AI 表示模拟输入;“复位值”一栏中的 I 表示输入,IA 表示不活跃且无上拉电阻,PU 表示内部上拉电阻使能。在表 1-2 中各个 PIO 口均描述为通用目的数字输入/输出口,更详细的描述方法则需要添加上口号和位号,例如,PIO0_13 表示通用输入/输出口 0 号的第 13 位。在 LPC812 中,只有 0 口,在以后的芯片扩展中,会添加上 1 口、2 口等。由于 LPC812 为 32 位微控制器,每个口的位数应为 32 位,对于 0 口而言,应该为 PIO0_0~PIO0_31,在 LPC812 中只有 PIO0_0~PIO0_17,在以后的芯片扩展中将会出现其他的位。

1.3 在系统编程

表 1-2 中提到了 ISP,即在系统编程(in-system programming),即 LPC812 可以通过串口线连接到计算机,并通过计算机(通常称为上位机)的 Flash Magic 软件向 LPC812 写入程序。这里用到的引脚有:PIO0_12、PIO0_0 和 PIO0_4,如果在 LPC812 复位时 PIO0_12 外接低电平,则启动 LPC812 片内 Bootloader 代码。这部分代码位于 LPC812 片内 8KB ROM 空间中,地址为 0x1FFF 0000~0x1FFF 2000,此时,Bootloader 代码将 PIO0_0 和 PIO0_4 配置为串口 0 的 RXD 和 TXD,用于与上位机通信,从而实现上位机向 LPC812 片内 Flash 存储器写入程序的功能。下面通过具体的电路连接图说明 ISP 工作电路,如图 1-4~图 1-6 所示。

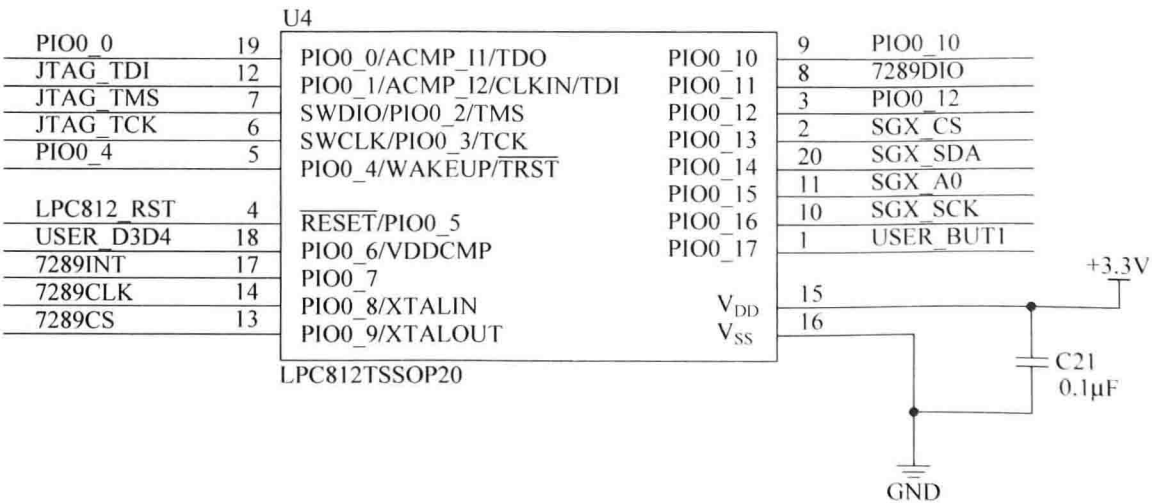
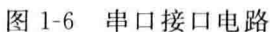


图 1-4 LPC812 核心电路

在图 1-5 中,将 P1 的 2、3 脚、P2 的 1、2 脚和 P4 的 1、2 脚分别用 3 个跳线帽短接(即使它们分别相连通),将 J2 的 1、2 脚用跳线帽短接。由图 1-4~图 1-6 可知,LPC812 的 PIO0_



用于 ISP 的上位机软件称为 Flash Magic, 可以从 <http://www.flashmagictool.com/> 上下载。Flash Magic 是专门为 NXP 的微控制器设计的通过串口向芯片下载程序的软件, 其工作界面如图 1-7 所示。