



卓越工程师培养计划

▪ 单片机 ▪

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

张 涵 王海堂 沈孝芹 于复生 编著



ARM Cortex-M0

嵌入式系统 设计与应用



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY



卓越工程师培养计划

■ 单片机 ■

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

张 涵 王海堂 沈孝芹 于复生 编著



ARM Cortex-M0

嵌入式系统 设计与应用

电子工业出版社

Publishing House of Electronics Industry

内 容 简 介

本书系统介绍了基于 ARM Cortex-M0 内核的 LPC1114/48 ARM 处理器的使用方法。本书内容包括三个部分：第一部分为入门篇，主要介绍 ARM Cortex-M0 处理器，LPC11××最小系统设计，MDK 编译环境的建立与使用等内容；第二部分为基础篇，对 LPC1114/48 ARM 处理器的内核及各个外设进行了讲解，并且针对每一部分都有详尽的实例分析；第三部分为综合篇，从应用开发的角度出发，详尽介绍了利用 LPC1114/48 ARM 处理器对当今许多热门技术予以实现的方法。

本书面向的是工业控制领域的工程技术人员和在校学生，包括一些单片机技术的初级用户、具有一定基础知识的初入职场的单片机编程爱好者，以及高等学校相关专业的在校学生等。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目 (CIP) 数据

ARM Cortex-M0 嵌入式系统设计与应用 / 张涵等编著. —北京：电子工业出版社，2013.7

(卓越工程师培养计划)

ISBN 978-7-121-21022-8

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 (2013) 第 162795 号

策划编辑：张 剑 (zhang@phei.com.cn)

责任编辑：桑 昀

印 刷：三河市鑫金马印装有限公司

装 订：三河市鑫金马印装有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1092 1/16 印张：16.5 字数：436.3 千字

印 次：2013 年 7 月第 1 次印刷

印 数：4 000 册 定价：45.00 元



凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：(010) 88258888。

前 言

ARM Cortex-M0 处理器是由 ARM 公司开发的,是现有的体积最小、能耗最低和能效最高的 ARM 处理器。该处理器硅面积积极小、能耗极低并且所需的代码量极少,这使得开发人员能够以 8 位的设备实现 32 位设备的性能,从而省略了 16 位设备的研发步骤,降低了研发成本,提高了研发效率。

本书对基于 ARM Cortex-M0 内核的 LPC1114/48 ARM 处理器的使用方法进行了详细介绍,与一般单片机教程的明显区别在于,本书将从最简单的部分出发,讲一步用一步,坚决避免长篇累牍,坚决摒弃大知识量的灌输式讲解方法。使得初学者能够轻轻松松踏入 ARM Cortex-M0 之门。

本书详细介绍基于 ARM Cortex-M0 内核的 LPC1114/48 ARM 处理器的具体使用方法,包括入门篇、基础篇和综合篇三个部分。入门篇讲述 ARM Cortex-M0 处理器、LPC11×× 最小系统设计、MDK 编译环境的建立与使用等内容。基础篇对 LPC1114/48 ARM 处理器的内核及各个外设进行了讲解,并且针对每一部分都有详尽的实例分析。综合篇从应用开发的角度出发,详尽介绍如何利用 LPC1114/48 ARM 处理器对当今许多热门技术予以实现。

本书面向的是工业控制领域的学生和工程技术人员,包括一些单片机技术的初级用户、具有一定基础知识初入职场的单片机编程爱好者以及高等院校相关专业的在校学生等。

本书共分 12 章。第 1 章介绍 Cortex-M0 处理器及其最小系统设计;第 2 章为 MDK 编译环境的建立与使用;第 3 章为 I/O 口的配置与使用;第 4 章为外部中断与 NVIC;第 5 章为 LPC1100 时钟系统分析;第 6 章为 SysTick 模块与定时器/计数器;第 7 章为通用异步收发器;第 8 章为 I²C 总线接口;第 9 章为 SSP 控制器;第 10 章为 A/D 转换器(ADC);第 11 章为基于 nRF24L01 的无线通信;第 12 章为电子阅读器制作。本书从实用角度出发,采用理论讲解与实例相结合的讲述方法,简明清晰,重点突出。在叙述上力求深入浅出、通俗易懂。相信会为读者的学习和工作带来一定的帮助。

本书第 1、2、3、10 和 11 章由张涵编写,第 4 章和第 8 章由沈孝芹编写,第 5 章和第 12 章由于复生编写,第 6、7 和 9 章由王海堂编写。全书由张涵统稿,参加本书编写的还有任秀华、管殿柱、陈继文、范文利、赵志超、耿宗亮、原政军、金毓海、罗映和张超。在本书编写过程中参阅了相关的书籍和文献资料,在此一并表示衷心的感谢。

由于编者水平有限,书中错误和疏漏在所难免,恳请广大读者批评指正。

编 著 者

目 录

入 门 篇

第 1 章 Cortex-M0 处理器及其最小系统设计	2
1.1 Cortex-M0 处理器概述	2
1.2 器件特性	3
1.3 器件选型	5
1.4 硬件结构	5
1.5 存储器寻址	6
1.6 异常向量表及其重映射	9
1.7 引导块	10
1.8 最小系统设计	11
1.9 电源系统设计	14
1.10 思考与练习	15
第 2 章 MDK 编译环境的建立与使用	16
2.1 MDK 的安装	16
2.2 μ Vision 4.0 IDE	18
2.3 建立工程	25
2.4 Flash Magic 的安装与使用	28
2.5 思考与练习	29

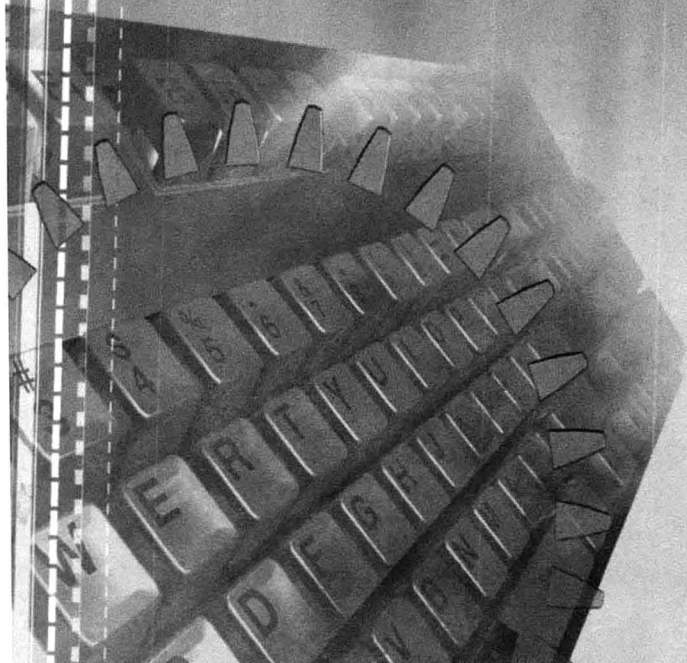
基 础 篇

第 3 章 I/O 口的配置与使用	32
3.1 概述	32
3.2 相关寄存器的功能与配置	35
3.3 编程方法	39
3.4 实例讲解	40
3.5 思考与练习	41
第 4 章 外部中断与 NVIC	42
4.1 相关寄存器的功能与配置	43
4.2 中断处理函数的编写	46
4.3 实例讲解	47
4.4 NVIC	49
4.5 思考与练习	52
第 5 章 LPC1100 时钟系统分析	53
5.1 时钟系统概述	53
5.2 LPC1114 时钟系统配置	53
5.3 思考与练习	62

第 6 章 SysTick 模块与定时器/计数器	63
6.1 系统节拍定时器概述	63
6.2 相关寄存器的描述与配置	63
6.3 系统节拍定时器配置	65
6.4 实例讲解	66
6.5 定时器/计数器概述	68
6.6 引脚描述	69
6.7 寄存器描述	69
6.8 定时器中断设置	76
6.9 应用示例	78
6.10 思考与练习	81
第 7 章 通用异步收发器	82
7.1 引脚描述	82
7.2 典型应用电路	84
7.3 寄存器描述	84
7.4 UART 中断	104
7.5 应用示例	108
7.6 思考与练习	114
第 8 章 I²C 总线接口	115
8.1 概述	115
8.2 I ² C 总线规范	116
8.3 I ² C 接口描述	120
8.4 I ² C 操作模式	121
8.5 寄存器描述	131
8.6 I ² C 接口中断	137
8.7 I ² C 总线的一个例子	138
8.8 应用软件包	139
8.9 思考与练习	150
第 9 章 SSP 控制器	151
9.1 概述	151
9.2 引脚描述	151
9.3 总线规范	152
9.4 寄存器描述	157
9.5 SSP 控制器接口中断设置	162
9.6 应用示例	164
9.7 思考与练习	168
第 10 章 A/D 转换器	169
10.1 概述	169
10.2 寄存器描述	170
10.3 基本操作	174

10.4	A/D 转换器中断设置	175
10.5	A/D 转换器应用举例	176
10.6	思考与练习	179
综 合 篇		
第 11 章	基于 nRF24L01 的无线通信	182
11.1	nRF24L01 概述	182
11.2	nRF24L01 的工作方式	182
11.3	nRF24L01 引脚描述	185
11.4	nRF24L01 的寄存器配置	186
11.5	nRF24L01 电路设计	191
11.6	代码分析	191
11.7	思考与练习	197
第 12 章	电子阅读器制作	198
12.1	TFT 液晶显示器的驱动方法	198
12.2	液晶屏 GUI 功能的实现	207
12.3	电阻式触摸屏原理与应用	217
12.4	SD 卡的读/写操作	221
12.5	FATFS 文件系统移植	232
12.6	电子阅读器程序设计	239
12.7	思考与练习	241
附录 A	ARM 汇编伪指令介绍	242
参考文献	254

入门篇



第 1 章 Cortex-M0 处理器 及其最小系统设计

ARM 公司于 2009 年 2 月推出了 Cortex-M0 处理器，该处理器一出现就得到来自业界的大量关注。目前，市场上已经出现了多种基于 ARM Cortex-M0 内核的微处理器，如 NXP（恩智浦半导体公司）的 LPC1100 系列、Nuvoton 公司的 NuMicro 系列等。



1.1 Cortex-M0 处理器概述

ARM Cortex-M0 处理器，是目前市场上体积最小、能耗最低的 ARM 处理器。该处理器能耗非常低、门数量少、代码占用空间小，使得 MCU 开发人员能够以 8 位处理器的价位，获得 32 位处理器的性能。超低门数还使其能够用于模拟信号设备和混合信号设备及 MCU 应用中，明显节约系统成本。ARM 凭借其作为低能耗技术的领导者和创建超低能耗设备的主要推动者的丰富专业技术，使得 Cortex-M0 处理器有着令人惊叹的超低功耗。该处理器把 ARM 的 MCU 路线图扩展到超低能耗 MCU 和 SoC 应用中，如医疗器械、电子测量、照明、智能控制、游戏装置、紧凑型电源、电源和电动机控制、精密模拟系统和 IEEE 802.15.4 (ZigBee) 及 Z-Wave 系统。Cortex-M0 处理器还适合拥有诸如智能传感器和调节器的可编程混合信号市场，这些应用在传统上一直要求使用独立的模拟设备和数字设备。

LPC1100 系列是恩智浦公司于 2009 年 11 月推出的基于 ARM Cortex-M0 内核的处理器，是市场上定价最低的 32 位 MCU 解决方案，它的价值和易用性比现有的 8 位/16 位微控制器更胜一筹。LPC1100 系列 ARM 处理器性能卓越、简单易用、功耗低，更重要的是，它能显著降低所有 8 位/16 位应用的代码长度。LPC1100 系列为那些寻求用可扩展 ARM 架构来执行整个产品开发过程的 8 位/16 位用户提供无缝的需求整合。

LPC1100 系列 Cortex-M0 微控制器的主频时钟高达 50MHz，拥有高性能的运算控制能力，每秒可执行 4500 多万条指令，支持睡眠、深度睡眠和深度掉电 3 种低功耗模式。同时，它还拥有丰富的外设组件：高达 32KB 的片内 Flash 程序存储器、8KB 片内 SRAM、一路 I2C(FM+)、一路 RS-485/EIA-485 UART、两路 SSP、4 个通用定时器以及多达 42 个通用 I/O 口。

对于很多初学者而言，“ARM Cortex-M0”与“LPC1100”这两个词的区别在他们心里可能有些模糊，下面明确一下两者的区别。ARM Cortex-M0 只是一个 ARM 内核，该内核是 ARM 公司开发的，而且它仅仅是庞大的 ARM 内核家族的一分子，因为还有 Cortex-A 系列、Cortex-M3、ARM7，ARM9，ARM11 等很多内核。ARM 公司自己并不生产芯片，他

们只把设计好的内核技术以授权的方式卖给其他芯片生产厂商，比如恩智浦、三星等。这些厂商购得授权之后，将会以该内核为基础，根据自己的需要添加外设，最终就形成了各具特色的 ARM 芯片（大家是否应该认识到，“ARM 芯片”只是一系列以 ARM 内核为基础的芯片的总称）。现在再来概括一下“ARM Cortex-M0”与“LPC1100”之间的区别：LPC1100 只是恩智浦公司以 ARM Cortex-M0 内核为基础，添加了多种外设；最终设计出来的一款 ARM 芯片。Cortex-M0 处理器有着如下特点。

【32 位的性能和效率】Cortex-M0 是 32 位的 RISC 处理器，基于 ARMv6 M 架构，为 3 段流水线的冯·诺依曼结构。该处理器使用 ARMv6M Thumb 指令集，包含 Thumb 2 技术，因此该处理器拥有 32 位处理器的高性能，又有着比 8 位、16 位处理器更好的代码密度。

【小尺寸、低功耗】Cortex-M0 处理器的门数不到 12000 门，如果使用 180nm 超低漏电（ULL）的工艺，其功耗仅为 $85\mu\text{W}/\text{MHz}$ ，而其性能可达到 $0.9\text{DMIPS}/\text{MHz}$ 。另外，Cortex-M0 处理器具有的睡眠模式和深度睡眠模式可以进一步节省功耗。因此，Cortex-M0 处理器将更多地应用于电池供电的应用之中。

【高效而且可确定的中断处理】Cortex-M0 处理器内置一个紧密连接的可配置的嵌套向量中断控制器（Nested Vectored Interrupt Controller, NVIC），以实现中断服务程序（ISR）的高速执行。同时，还采用硬件实现寄存器堆栈、尾链（Tail-chaining）、抢占（Preemption）、迟到（Late-arriving）等技术，大大降低了中断处理延迟。而且中断处理程序不需要任何汇编封装代码，去掉了 ISR 的所有多余代码开销。

【硬件乘法器】Cortex-M0 处理器提供两种硬件乘法器可选，以实现性能或尺寸优化：

- ☉ 单周期乘法器，以实现高性能的优化；
- ☉ 32 周期乘法器，以实现面积的优化。

【简单易用】编程模型简单，没有 ARM7、ARM9 那么多工作模式，指令集简单。更重要的是可以 100% 使用 C 语言编程。

【良好的兼容性】Cortex-M0 处理器的工具及二进制代码可以向上兼容其他 Cortex M 系列处理器，因此非常方便用户将其应用进行移植和扩展。

【方便的调试接口】IC 厂商在实现 Cortex-M0 处理器时，可选 0~4 个硬件断点、0~2 个观测点。另外，厂商可以选择 JTAG 或 SWD 接口作为调试端口。由于 Cortex-M0 具有以上特性，因此它特别适用于医疗器械、电子测量、照明、智能控制、游戏装置、电源控制、电动机控制、精密模拟系统、ZigBee 及 Z Wave 系统等领域。

另外，Cortex-M0 处理器还适用于可编程混合信号应用，比如以往需要使用单独模拟和数字设备的智能传感器和传动装置。



1.2 器件特性

【Cortex-M0 内核特性】

- ☉ Cortex-M0 内核集成了嵌套向量中断控制器（NVIC），支持多达 32 个中断；集成了系统节拍定时器。

- ☺ 集成 JTAG 调试接口, LPC1100 系列微控制器未将内核的 JTAG 调试接口引出, 因此只支持 SWD 串行调试。
- ☺ 集成串行调试接口, 支持两个观测点和 4 个中断点。
- ☺ ARM Cortex-M0 处理器最高工作频率为 50MHz。

【LPC1100 系列处理器的主要特性】

- ☺ 32KB (LPC1114)、24KB (LPC1113)、16KB (LPC1112) 或 8KB (LPC1111) 的片内 Flash 程序存储器。
- ☺ 8KB 的静态 RAM。
- ☺ 通过片内 Bootloader 软件来实现在系统编程 (ISP) 和在应用中编程 (IAP)。
- ☺ 串行接口。
 - ☞ UART: 可产生小数波特率, 带有内部 FIFO, 支持 RS-485/EIA-485, 具有 MODEM 控制。
 - ☞ 2 个 SSP 控制器, 具有 FIFO 和多协议功能 (LQFP48 和 PLCC44 封装只有第二个 SSP 有该功能)。
 - ☞ I²C 总线接口支持全部 I²C 总线规范和 Fast-mode Plus 模式, 数据速率高达 1Mbit/s, 具有多地址识别和监控模式。
- ☺ 其他外设。
 - ☞ 多达 42 个通用 I/O (GPIO) 引脚, 上拉/下拉电阻可配置。
 - ☞ 一个引脚具有 20mA 的高电流驱动能力。
 - ☞ 2 个 I²C 总线引脚在 Fast-mode Plus 模式下具有 20mA 的高电流驱动能力。
 - ☞ 4 个通用定时器/计数器, 共有 4 个捕获输入和 13 个比较输出。
 - ☞ 看门狗定时器 (WDT)。
 - ☞ 系统节拍定时器。
- ☺ 串行调试, 支持 SWD 模式调试。
- ☺ 集成的 PMU (Power Management Unit), 在睡眠、深度睡眠和深度掉电模式下自动调节; 内部稳压器, 将功耗降至最低。
- ☺ 3 种节能模式: 睡眠、深度睡眠和深度掉电。
- ☺ 单个 3.3V 电源 (2.0~3.6V)。
- ☺ 10 位 ADC, 在 8 个引脚之间实现输入多路复用。
- ☺ GPIO 引脚可以用作边沿和电平触发的中断源。
- ☺ 带分频器的时钟输出功能可以输出主振荡器时钟、IRC 时钟、CPU 时钟或看门狗时钟。
- ☺ 掉电检测有 4 个中断阈值和 1 个强制复位阈值。
- ☺ 上电复位 (POR)。
- ☺ 晶体振荡器的工作范围为 1~25MHz。
- ☺ 12MHz 内部 RC 振荡器可调节到 1% 的精度, 可以选择作为一个系统时钟。
- ☺ PLL 允许 CPU 无须使用高频晶体而工作在最大 CPU 速率下。时钟可以由主振荡器、内部 RC 振荡器或看门狗振荡器提供。
- ☺ 提供 LQFP48、PLCC44 和 HVQFN33 几种封装形式。



1.3 器件选型

LPC1100 系列处理器目前主要有：LPC1100 系列、LPC11C1×系列、LPC11U1×系列、LPC11E1×系列等。本书中介绍的处理器为其中的 LPC1100 系列，具体选型参见表 1-1。

表 1-1 器件选型表

型 号		Flash (KB)	SRAM (KB)	UART RS-485	I ² C	SSP	ADC
LPC1111	LPC1111FHN33/101	8	2	1	1	1	8ch/b
	LPC1111FHN33/201	8	4	1	1	1	8ch/b
LPC1112	LPC1112FHN33/101	16	2	1	1	1	8ch/b
	LPC1112FHN33/201	16	4	1	1	1	8ch/b
LPC1113	LPC1113FHN33/201	24	4	1	1	1	8ch/b
	LPC1113FHN33/301	24	8	1	1	1	8ch/b
	LPC1113FBD48/301	24	8	1	1	1	8ch/b
LPC1114	LPC1114FHN33/201	32	4	1	1	1	8ch/b
	LPC1114FHN33/301	32	8	1	1	1	8ch/b
	LPC1114FBD48/301	32	8	1	1	1	8ch/b
	LPC1114FA44301	32	8	1	1	1	8ch/b



1.4 硬件结构

下面分析一下 LPC1114 的基本结构图，如图 1-1 所示。LPC1114 内部分为四部分：ARM Cortex-M0 内核、时钟系统、高速 GPIO 和片上外设。

- ☉ ARM Cortex-M0 内核。该部分同时集成 SWD 调试接口。关于 Cortex-M0 内核的工作方式、内部寄存器、指令集等将在后面的学习中详细讲解。
- ☉ 时钟系统。通俗地讲，时钟系统就好比心脏，一个人能够健康地活着依靠的正是心脏规律性的跳动。对于 LPC1114 而言，时钟系统就好比是它的核心，只有时钟系统正常工作，才能保证 LPC1114 顺利完成任务。关于时钟系统的配置将在后面的学习中详细讲解。
- ☉ 高速 GPIO。GPIO 即为“普通输入输出”，LPC1114 共有 4 组 42 个输入输出，0、1、2 组各有 12 个，第 3 组有 6 个，共计 42 个；42 个通用 I/O 均可配置上拉/下拉电阻；单个 I/O 具有 20mA 的电流驱动能力。
- ☉ 片上外设。LPC1100 系列 ARM 芯片有着丰富的片上外设。这种片上集成的方法有效地降低了电路设计的复杂性，很好地降低了开发成本，提高了开发效率。片上外

设有：1 个通用异步串行收发器 (UART)、2 个 SSP 控制器、1 个 I²C 总线接口、看门狗定时器 (WDT)、10 位 ADC 等。

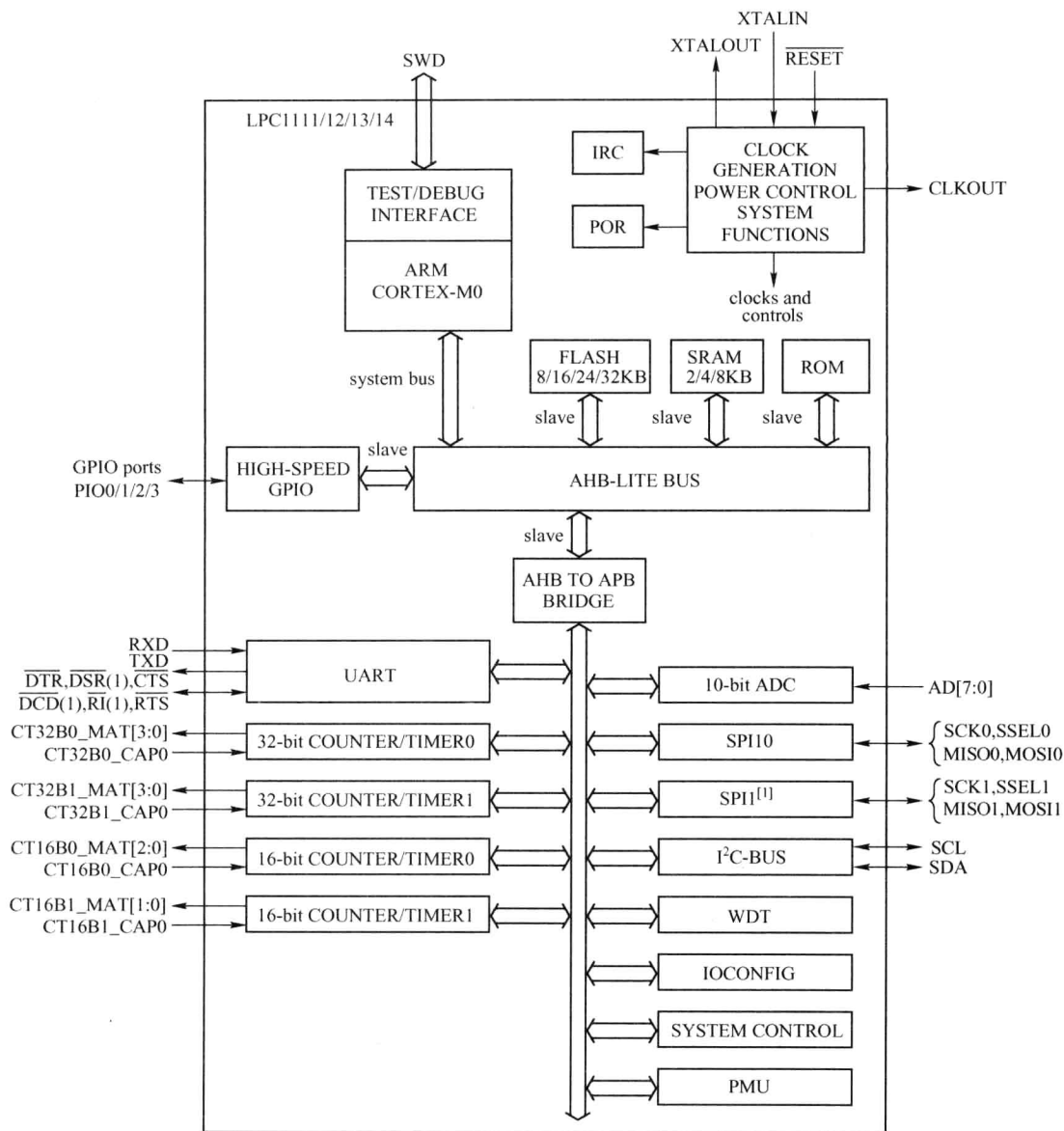


图 1-1 处理器基本结构



1.5 存储器寻址

LPC1100 系列 Cortex-M0 处理器内部 Flash 容量参见表 1-2，片内 Flash 可用于代码和数据存储。对片内 Flash 的编程有三种方法：

- ☺ 使用 SWD 仿真/调试器，通过芯片的 SWD 接口下载程序；
- ☺ 使用在系统编程技术（即 ISP），通过 UART 接口下载程序；
- ☺ 使用在应用编程技术（即 IAP），通过 IAP，可实现用户程序运行时对 Flash 进行擦除/编程，这样就为数据存储和现场固件的升级都带来了极大的灵活性。

LPC1100 系列 Cortex-M0 处理器的片内 RAM 为静态 RAM（SRAM），芯片容量参见表 1-2，它可用作代码或数据的存储。

表 1-2 LPC1100 系列 Cortex-M0 存储器容量

器 件	片内 Flash	SRAM		
		/101	/201	/301
LPC1111	8KB	2KB	4KB	—
LPC1112	16KB	2KB	4KB	—
LPC1113	24KB	—	4KB	8KB
LPC1114	32KB	—	4KB	8KB

LPC1100 系列 Cortex-M0 含有 4GB 的地址空间，地址分布参见表 1-3。

表 1-3 LPC1100 系列 Cortex-M0 存储器的地址分布

寻 址 范 围	用 途	描 述
0xE000 0000~0xE00F FFFF	Cortex-M0 内核相关寄存器	包括 NVIC 和系统节拍定时器
0x5000 0000~0x501F FFFF	AHB 外设	高速 GPIO
0x4000 0000~0x4007 FFFF	APB 外设	32 个外设模块，每个 16KB
0x1FFF 0000~0x1FFF 3FFF	片上 ROM	16KB 引导 ROM
0x1000 0000~0x1000 1FFF	片上 SRAM	静态 RAM（8KB）
0x0000 0000~0x0000 7FFF	片上非易失性存储器	Flash 存储器（32KB）

存储器本身不具有地址的信息，它们在芯片中的地址是由芯片厂商或用户分配的，给存储器分配地址的过程称为存储器映射。LPC1100 系列 Cortex-M0 存储器空间由几个不同的存储区域组成，如图 1-2 所示为复位后从用户编程角度所看到的地址空间映射。AHB 外设区域为 2MB，可分配多达 128 个外设。在 LPC111×系列 Cortex-M0 处理器上，GPIO 端口是唯一的 AHB 外设。APB 外设区的大小为 512KB，可分配多达 32 个外设，每个外设空间大小都为 16KB，这样可简化外设的地址译码。

所有外设寄存器不管规格大小，地址都是字对齐的（32 位边界）。这样不管是字节、半字还是字长度的寄存器都是一次性访问。例如，不可能对一个字寄存器的最高字节执行单独的读或写操作。

AHB 外设是挂接在芯片内部的 AHB 总线上的外设部件，具有较高的速度。LPC1100 系列 Cortex-M0 处理器中，AHB 外设地址映射参见表 1-4。APB 外设是挂接在芯片内部 VPB 总线上的外设部件，速度通常比 AHB 外设要低，LPC1100 系列 Cortex-M0 处理器中 APB 外设地址映射参见表 1-5。



图 1-2 LPC1111/12/13/14 系统存储器地址空间映射

表 1-4 AHB 地址映射表

AHB 外设	基地址	外设名称
4-127	0x5004 0000~0x501F FFFF	保留
3	0x5003 0000	GPIO PIO3
2	0x5002 0000	GPIO PIO2
1	0x5001 0000	GPIO PIO1
0	0x5000 0000	GPIO PIO0

表 1-5 APB 地址映射表

APB 外设	基 地 址	外 设 名 称
23-31	0x4005 C000~0x4007 FFFF	保留
22	0x4005 8000	SSP
19-21	0x4004 C000~0x4005 7FFF	保留
18	0x4004 8000	系统控制
17	0x4004 4000	I/O 配置
16	0x4004 0000	SSP0
15	0x4003 C000	Flash 控制器
14	0x4003 8000	电源管理
10-13	0x4002 8000~0x4003 7FFF	保留
9	0x4002 4000	保留
8	0x4002 0000	保留
7	0x4001 C000	ADC
6	0x4001 8000	32 位定时器/计数器 1
5	0x4001 4000	32 位定时器/计数器 0
4	0x4001 0000	16 位定时器/计数器 1
3	0x4000 C000	16 位定时器/计数器 0
2	0x4000 8000	UART
1	0x4000 4000	看门狗定时器
0	0x4000 0000	I ² C



1.6 异常向量表及其重映射

由 Cortex-M0 体系结构可知, Cortex-M0 的异常向量表位于 Flash 的起始地址 0x0000 0000 处, 如图 1-3 所示。当发生异常事件时, 硬件将自动从向量表中取出对应中断服务程序的入口地址, 异常向量参见表 1-6。

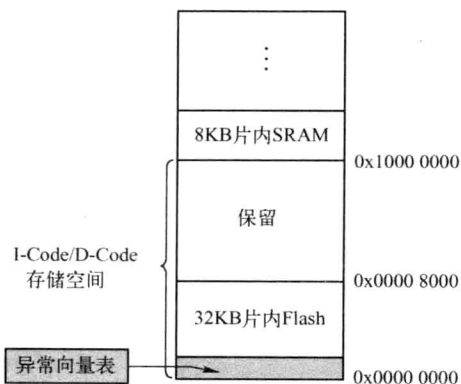


图 1-3 异常向量表

表 1-6 复位后异常向量表

地 址	异常编号	描 述
...	...	其他异常处理函数的入口地址
0x0000 000C	3	硬 Fault 处理函数入口地址
0x0000 0008	2	复位向量 (PC 初始值)
0x0000 0004	1	复位向量 (PC 初始值)
0x0000 0000	—	MSP 的初始值

LPC1100 系列 Cortex-M0 处理器含有片内 Flash、片内 SRAM、BootROM 等，通过存储器的重映射机制对异常向量表进行重映射，可以实现在不同的存储器中处理异常事件。异常向量表的重映射是通过“系统中断向量重映射寄存器 (SYSMEMREMAP)”实现的，SYSMEMREMAP 寄存器地址为 0x4004 8000，描述参见表 1-7。

表 1-7 SYSMEMREMAP 寄存器

位	符 号	值	描 述	复 位 值
31:2	—	—	保留	—
1:0	MAP	—	保留	0x10
		10	用户 Flash 模式。中断向量不会被映射，一直位于 Flash	
		01	用户 RAM 模式。中断向量被重新映射到静态 RAM	
		00	BootLoader 模式。中断向量被重新映射到 BootROM	



1.7 引导块

Boot Block 是芯片设计厂家在 LPC1100 系列 Cortex-M0 处理器内部固化的一段代码，用户无法对其修改或删除。这段代码在芯片复位后首先运行，它提供对 Flash 存储器编程的方法。

Boot Block 的功能包括：判断用户代码是否有效、芯片是否加密、在应用编程 (IAP)。下面分别说明 Boot Block 的功能。

【用户代码是否有效】 Boot Block 在把芯片的控制权交给用户程序之前，要先判断用户程序是否有效，否则将不运行用户程序，这样可以避免在现场设备中的芯片因为代码损坏而导致程序跑飞引起事故。异常向量表中第 7 个单元是一个保留字 (位于 0x001C)，它保存的数据是向量表前 7 个字校验和的补数，Boot Block 就是利用这个保留字来判断用户程序是否有效。当异常向量表前 8 个字校验和为 0 时，Boot Block 认为用户代码有效，否则为无效。当代码无效时，Boot Block 将令 CPU 进入 ISP 状态。

【芯片是否加密】 芯片可加密是 LPC1100 Cortex-M0 系列处理器的一个重要特性，该功能可以保护芯片用户的知识产权不受侵害。加密后的芯片是无法使用 JTAG 接口进行调试的，也无法使用 ISP 工具对存储器进行代码下载和读取，而只有对芯片整片擦除后才能做进一步的操作。