源程序和相关资料下载
http://bbs.cepark.com

STM32

自学笔记

蒙博宇　编著

北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

内 容 简 介

本书以新颖的思路、简单的逻辑、简洁的语言来阐述作者初遇 STM32 以来的种种认识，书中多处内容都是由作者从 STM32 初学时的实践中总结而来。本书主要介绍 ARM Cortex－M3 系列 STM32 的原理及应用，全书共 7 章。第 1 章主要对 STM32 做基本介绍；第 2 章介绍 ARM Cortex－M3 内核架构的大致概况；第 3 章从外设特性、功耗特性、安全特性等方面对 STM32 进行全面的剖析；第 4 章主要介绍开发工具；第 5 章则引导读者针对 STM32 的外设进行一系列的基础实验设计；第 6 章通过 10 篇高级应用文章介绍 STM32 的一些高级知识；第 7 章则通过一个综合实例讲述一个 STM32 完整应用方案的实现过程。本书共享源代码和相关资料，下载地址为 http://bbs.cepark.com 和北京航空航天大学出版社“下载中心”。

本书条理清楚，通俗易懂，贴近读者，主要面向 STM32 的初学者，以及所有对 ARM Cortex－M3 系列微控制器感兴趣的朋友们。

图书在版编目(CIP)数据

STM32 自学笔记 / 蒙博宇编著. －北京 ：北京航空航天大学出版社，2012.2

ISBN 978－7－5124－0698－8

Ⅰ. ①S… Ⅱ. ①蒙… Ⅲ. ①微控制器 Ⅳ. ①TP332.3

中国版本图书馆 CIP 数据核字(2012)第 001050 号

STM32 自学笔记

蒙博宇 编著

责任编辑 刘 星

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) http://www.buaapress.com.cn

发行部电话：(010)82317024 传真：(010)82328026

读者信箱：emsbook@gmail.com 邮购电话：(010)82316936

北京时代华都印刷有限公司印装 各地书店经销

*

开本：787×960 1/16 印张：27 字数：605 千字

2012 年 2 月第 1 版 2012 年 2 月第 1 次印刷 印数：4 000 册

ISBN 978－7－5124－0698－8 定价：49.50 元

前 言

STM32 微控制器是近年来迅速兴起的基于 ARM Cortex－M3 内核的高端 32 位微控制器的代表。STM32 微控制器依托意法半导体公司(ST Microelectronics,简称 ST)本身雄厚的研发、生产实力,在正确的市场推广策略引导下,迅速占据了国内高端微控制器的大部分应用领域,优秀的性能、丰富的外设、稳定的供货以及低廉的价格等优点,使其长期保持优势。目前,STM32 微控制器在工业控制、消费电子、手持设备、汽车电子、安防监控等众多领域得到了广泛的应用;正因为其高性价比、适合手工 DIY 的优点,在高校学生群体中也有非常高的人气。

(1) 笔者与 STM32 的点点滴滴

2006 年,ST 公司开始在中国推广 STM32 微控制器,至 2008 年时,STM32 在国内已经有相当的地位了;但此时在高校内很多学生仍然热衷于使用传统的 8 位单片机来进行电子设计。最明显的一个证据就是,笔者当初想在淘宝上购买一个 STM32 开发板,但发现销售此类开发板的店家不过数十家,与今时今日相比可谓差距甚大。经过反复比较,最终选定了一个比较简单的开发板,就此踏上了 STM32 的学习之路。当时,笔者是第一次接触 ARM 体系结构的处理器,虽说之前也有一些 8 位单片机的开发经历,但毕竟还是差异不小,困难也就接踵而来了。

首先开发环境的搭建就耗费了一周的时间。当时 STM32 的资料很零散,而且以英文居多;开发环境功能选项复杂,难以上手;而 STM32 的工程复杂度更是之前的 8 位单片机所不能比的;最要命的是,当时没有任何一份详实的入门教程或入门手册……相信时至今日,有相当多刚刚接触 STM32 的朋友也有这样的感觉。但无论如何,开发环境总算搭建好了,当时想终于可以来点个灯啥的。

此时第二个问题来了,STM32 微控制器的开发主要依托于固件函数库进行,这使得开发者不再面对底层寄存器进行操作,笔者对这种开发方式相当陌生,只得找到库函数说明手册(找了很长时间才找到个英文的)逐个函数地查看其作用、参数定义,费了一番周章后,才把一个发光二极管点亮。

此后,学习 STM32 的道路也逐渐变得平坦起来:慢慢地认识了 STM32 的时钟树、普通外设、通信接口等外设单元的应用;开始尝试实现 STM32 的一些高级应用,如 Bootloader、IAP、

USB、DFU、脚本控制等；同时也开始深入了解 ARM Cortex－M3 内核的体系结构。从此之后，参与开发的项目也一直使用 STM32 微控制器作为主控核心，越发地能深切体会到这个“小东西”的超高性价比，也越发地喜爱这个具有划时代意义的片子。而现在回想起当初的“青葱”岁月，不得不说其实是一段相当令人愉悦和欣慰的时光。

(2) 如何入门 STM32 微控制器

对于一个初学者而言，特别是只有少数 8 位单片机开发经验的人来说，跨入 STM32 这扇大门的门槛在于开发方式的改变，这里的“改变”包括：开发环境的改变、开发工具的改变、工程结构的改变和调试手段的改变。详述起来其实有如下几点：

- 开发环境从常见的 Keil C51 或 WinAVR 转移至 Keil MDK 或 IAR EWARM，这其中变化较大的是开发环境的配置，如 IAR EWARM 要求用户自行配置文件路径、优化选项、脚本文件等。
- 常见的 8 位单片机的开发主要通过串口或 ISP 方式进行下载，同时几乎无法实时跟踪调试（虽然各种单片机都有实时仿真器，但当一个仿真器的价格和开发板的价格相当时，多数人选择望而却步），而 STM32 的开发则几乎必须要借助硬件仿真器才能完成。
- STM32 微控制器基于 ARM 体系结构，同时其开发主要基于固件库函数，这样使得 STM32 的工程结构必然和传统的 51、AVR 单片机有不小的差别。
- 进行 STM32 微控制器开发，很大一部分精力其实是耗费在调试这一环节上，这要求开发人员要经常查看寄存器的内容、内存的内容，跟踪变量的变化，甚至实时地修改内存的值。这都要依托具体的仿真调试器和开发环境的特性来实现，而跨入电子设计大门不久的初学者们则较少涉及这些操作。

那么，究竟应该如何去着手入门 STM32 微控制器呢？笔者用一句简练的话来概括，就是：入门 STM32，就是要适应上述“改变”，同时克服适应过程中所遇到的困难。笔者想在此稍微强调的是，若仅是入门 STM32 微控制器，则必需的所有硬件成本不会超过 200 元。笔者编写这本书所使用的开发板，单板成本在 100 元以内。

(3) 本书导读

本书实际上是笔者在入门 STM32 微控制器后，回顾这段过程所得到的点点滴滴的想法和灵感而作，是面向广大 STM32 初学者们而编写的一本“STM3 入门书籍”，引导读者走进 STM32 这扇大门。

本书从内容上可分为理论部分和实践部分，理论部分大概占据 30％的篇幅，实践部分则占据了大部分篇幅。理论部分主要围绕“STM32 是什么”和“STM32 可以用来干什么”这两个主题来对 STM32 做深入浅出的介绍。读者通过阅读理论部分的内容，对 STM32 有感性的认识即可。实践部分主要通过 STM32 多个外设应用实例，来引导读者有针对性地进行 STM32

外设实验。实践部分编写的核心思路在于：以实验设计为核心，阐述实现每个实验所需的全部要点。这种编写思路的好处在于，可以把本书的内容精练化，读者通过阅读本书可以掌握STM32微控制器60%的特性；但笔者最希望看到的是，在这60%的引导下，读者能自主地去学习余下那40%的特性。

全书共7章。第1章主要对STM32做基本介绍；第2章介绍ARM Cortex－M3内核架构的大致概况；第3章从外设特性、功耗特性、安全特性等方面对STM32进行全面的剖析；第4章主要介绍开发工具；第5章则引导读者针对STM32的外设进行一系列的基础实验设计；第6章通过10篇高级应用文章介绍STM32的一些高级知识；第7章则通过一个综合实例讲述一个STM32完整应用方案的实现过程。

本书主要使用ST公司发布的STM32微控制器固件函数库v2.0版本进行实验设计，也许有读者会提出疑问，为何不使用当前最新的v3.x版本固件库呢？笔者的观点是：v3.x版本的固件库，引入了ARM公司的CMSIS标准，这使得v3.x版本的固件库对初学者而言更难接受，因为v3.x版本固件库相比于v2.0版本的固件库，掩盖了更多的实现细节。笔者认为从学习的角度而言，v2.0固件库更为适合。

笔者建议有少许单片机开发经历的读者，可以直接进行实践部分的阅读，而在完成这些实验的过程中，穿插地阅读理解理论部分，这样可以得到最好的阅读效果。

(4) 相关资源

本书共享相关资料，包含：书中所有出现的代码，它们都位于完整的应用工程中；笔者所使用的CEPARK STM32学习板的原理图、PCB和实物照片，读者若有兴趣可以自行参考制作开发板。在自制过程中若遇到问题，笔者乐于提供支持。

共享资料内容索引如下：

基础实验	该文件夹下包含了本书第5章“STM32基础实验”的所有源程序。
进阶应用	该文件夹下包含了本书第6章“STM32进阶应用”的所有源程序。
综合性实验	该文件夹下是本书第7章“综合性实例：STM32的IAP方案”的源程序。
硬件描述	该文件夹下包含本书所用CEPARK STM32学习板的实物图、原理图和PCB图。

读者可参考书中讲述内容和共享资料中的内容，自行进行学习和实践，并可根据自己的实际应用开发属于自己的STM32应用方案。

共享资料下载和有关开发板的问题可访问以下网址：

http://bbs.cepark.com

(5) 致 谢

感谢朋友崔翠茹、郭志芳、连兰双，他们完成了本书资料的收集、整理、翻译等工作；感谢工作上的指导人苏杰，他在本书的创作期间给予不少技术方面的帮助；感谢CEPARK电子园网

站的站长李艳强、管理员曹佃生、马伟力、版主匡伟给笔者提供了一个硬件平台；感谢同事周葵华承担了书中部分程序的调试工作；感谢女友梁琼之承担了书中部分文字的校对工作；尤其要感谢北京航空航天大学出版社为笔者提供了这次出版机会；最后要感谢父母、亲人和所有支持笔者的朋友们。

限于笔者的水平和经验，加之时间比较仓促，疏漏或者错误之处在所难免，敬请读者批评指正。有兴趣的朋友可发送邮件到：losingamong@qq.com，与作者进行交流；也可发送邮件到：bhcbslx@sina.com，与本书策划编辑进行交流。

蒙博宇

2011 年 10 月

目 录

第 1 章

什么是 STM32

在过去的数年里，微控制器设计领域里一个主流的趋势是：基于 ARM7 和 ARM9 内核设计通用控制器的 CPU。而如今，已经有超过 240 种基于 ARM 核心的微控制器从众多芯片制造商手中诞生。意法半导体(ST Microelectronics，简称 ST)推出了 STM32 微控制器，这是 ST 第一个基于 ARM Cortex－M3 内核的微控制器。STM32 的出现将当前微控制器的性价比水平提升到了新的高度，同时它在低功耗场合和硬实时控制场合中亦能游刃有余。

1.1 从 Cortex－M3 说起

Cortex 是 ARM 公司最新系列的处理器内核名称，其推出的目的旨在为当前对技术要求日渐广泛的市场提供一个标准的处理器架构。和其他 ARM 处理器内核不一样的是，Cortex 系列处理器内核作为一个完整的处理器核心，除了向用户提供标准 CPU 处理核心之外，还提供了标准的硬件系统架构。Cortex 系列分为 3 个分支：专为高端应用场合而设的“A”(Application)分支，为实时应用场合而设的“R”(Real-time)分支，还有专为对成本敏感的微控制器应用场合而设的“M”(Microcontroller)分支。STM32 微控制器基于“M”分支的 Cortex－M3 内核，是专为实现系统高性能与低功率消耗并存而设计的，同时它足够低廉的价格也向传统的 8 位和 16 位微控制器发起了有力的挑战。

ARM7 和 ARM9 处理器被成功地整合进标准微控制器里的结果就是出现了各自独特的 SoC(System on Chip，也即片上系统)。特别从对异常和中断的响应处理方式上，用户会更容易看到这些 SoC 之间的区别，因为每家芯片制造商都有属于自己的一套解决方案。Cortex－M3 提出标准化的微控制器核心，在 CPU 的基础上又提供了整个微控制器的核心部分，包括中断系统、系统节拍时钟、调试系统以及存储区映射。Cortex－M3 内部的 4 GB 线性地址空间被分为 Code 区、SRAM 区、外部设备区以及系统设备区。和 ARM7 不同，Cortex－M3 处理器基于哈佛体系，拥有多重总线，可以进行并行处理，因而提升了整体性能。同时也和早期的 ARM 架构不同，Cortex－M3 处理器允许数据非对齐存取，以确保内部的 SRAM 得到充分地利用。Cortex－M3 处理器还可以使用一种称为 Bit－banding(译为“位带”)的技术，利用两个 32 MB 大小的“虚拟”内存空间实现对两个 1 MB 大小的物理内存空间进行“位”的置位和

清除操作。这样就可以有效地对设备寄存器和位于 SRAM 中的数据变量进行位操作，而不再需要冗长的布尔逻辑运算过程。

如图 1.1.1 所示，STM32 的核心 Cortex－M3 处理器，是一个标准的微控制器结构，拥有 32 位 CPU、并行总线结构、嵌套中断向量控制单元、调试系统以及标准的存储映射。

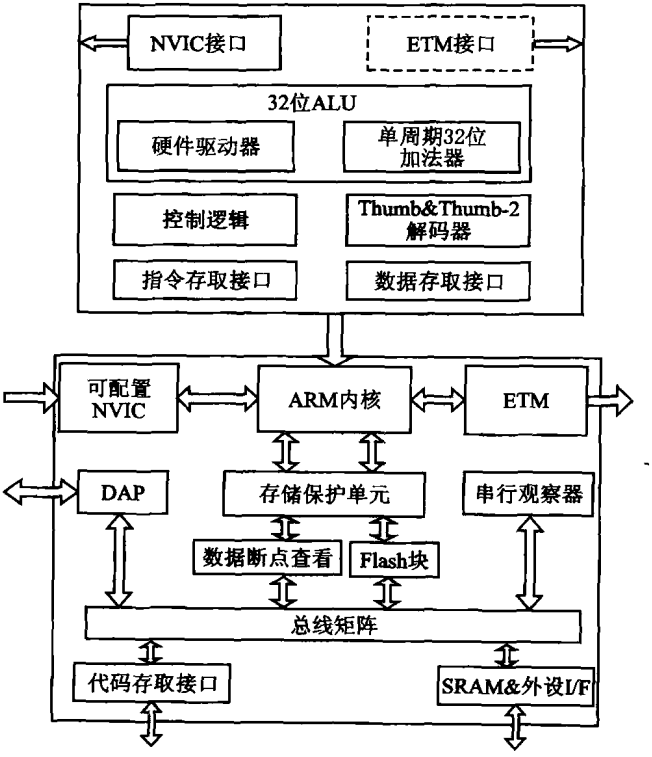

图 1.1.1　Cortex－M3 处理器内部架构一览

嵌套中断向量控制器(Nested Vector Interrupt Controller，简称 NVIC)是 Cortex－M3 处理器中一个比较关键的组件。NVIC 为基于 Cortex－M3 核心的微控制器提供了标准的中断架构和优秀的中断响应能力，为超过 240 个中断源提供专门的中断入口，而且可以赋予每个中断源单独的优先级。利用 NVIC 可以达到极快的中断响应速度，从收到中断请求到执行中断服务程序的第 1 条指令所要花费的时间仅仅为 12 个时钟周期。之所以能实现这种响应速度，一方面得益于 Cortex－M3 内核对堆栈的自动处理机制，这种机制是通过固化在 CPU 内部的微代码实现。另一方面，在中断请求连续出现的情况下，NVIC 使用一种称为“尾链”的技术使连续而来的中断在 6 个时钟周期之内得到服务。在中断压栈阶段，更高优先级的中断可以不耗费任何额外的 CPU 周期就能完成嵌入低优先级中断的动作。Cortex－M3 的中断结构和 CPU 的低功耗实现也有紧密的联系。用户可以设置 CPU 自动进入低功耗状态，而使用中断

来将其唤醒，CPU 在中断事件来临之前会一直保持睡眠状态。

Cortex－M3 的 CPU 支持两种运行模式：线程模式(Thread Mode)与处理模式(Handler Mode)，并且此两种模式都拥有各自独立的堆栈。这种设计使开发人员可以进行更为精密的程序设计，对实时操作系统的支持也更好。Cortex－M3 处理器还包含一个 24 位的可自动重装载定时器，可以为实时内核(RTOS)提供一个周期性的中断。ARM7 和 ARM9 处理器都有两种指令集(32 位指令集和 16 位指令集)，而 Cortex－M3 系列处理器支持新型的 ARM Thumb－2 指令集。由于 Thumb－2 指令集融合了 Thumb 指令集和 ARM 指令集，使 32 位指令集的性能和 16 位指令集的代码密度之间取得了平衡。ARM Thumb－2 专为 C/C＋＋编译器设计，这意味着 Cortex－M3 系列处理器的开发应用可以全部在 C 语言环境中完成。

1.2 STM32 面面观

尽管 ST 公司已经拥有 4 个基于 ARM7 和 ARM9 处理器的微控制器产品系列，但是 STM32 微控制器的推出仍然标志着 ST 在其两条产品主线(低价位主线和高性能主线)上都迈出了重大的一步。“单片最低价格低于 1 欧元!”——STM32 的出现伴随着如此凌厉的口号，对于市场上现有的 8 位单片机而言是一个严峻的挑战。STM32 最初发布的时候共推出 14 个不同的型号，它们被分为两个版本：最高 CPU 时钟为 72 MHz 的“增强型”和最高 CPU 时钟为 36 MHz 的“基本型”。不同版本的 STM32 器件之间在引脚功能和应用软件上是兼容的。这些不同的 STM32 型号里内置 Flash 最大可达 128 KB，SRAM 最大为 20 KB。而且在 STM32 最初发布之时，配备更大 Flash、RAM 容量和更多复杂外设的版本就已经在规划之中了。图 1.2.1 和图 1.2.2 显示了“增强型”和“基本型”的 STM32 在结构组成上的区别。

1. 精密性

乍一看 STM32 的设备配备，就像一个典型的单片机，配备常见的外设诸如多通道 ADC、通用定时器、I2C 总线接口、SPI 总线接口、CAN 总线接口、USB 控制器、实时时钟 RTC 等。然而，它的每一个设备都是非常有特点的，如 12 位精度的 ADC 具备多种转换模式，并带有一个内部温度传感器，带有双 ADC 的 STM32 器件，还可以使两个 ADC 同时工作，从而衍生出更为高级的 9 种转换模式。如 STM32 的每一个定时器都具备 4 个捕获比较单元，而且每个定时器都可以和另外的定时器联合工作以生成更为精密的时序；如 STM32 有专门为电机控制而设的高级定时器，带有 6 个死区时间可编程的 PWM 输出通道，同时其带有的紧急制动通道可以在异常情况出现时，强迫 PWM 信号输出保持在一个预定好的安全状态；如 SPI 接口设备含有一个硬件 CRC 单元，支持 8 位字节和 16 位半字数据的 CRC 计算，在对 SD 或 MMC 等存储介质进行数据存取时相当有用。

令人称奇的是，STM32 还包含了 7 个 DMA 通道。每个通道都可以用来在设备与内存之

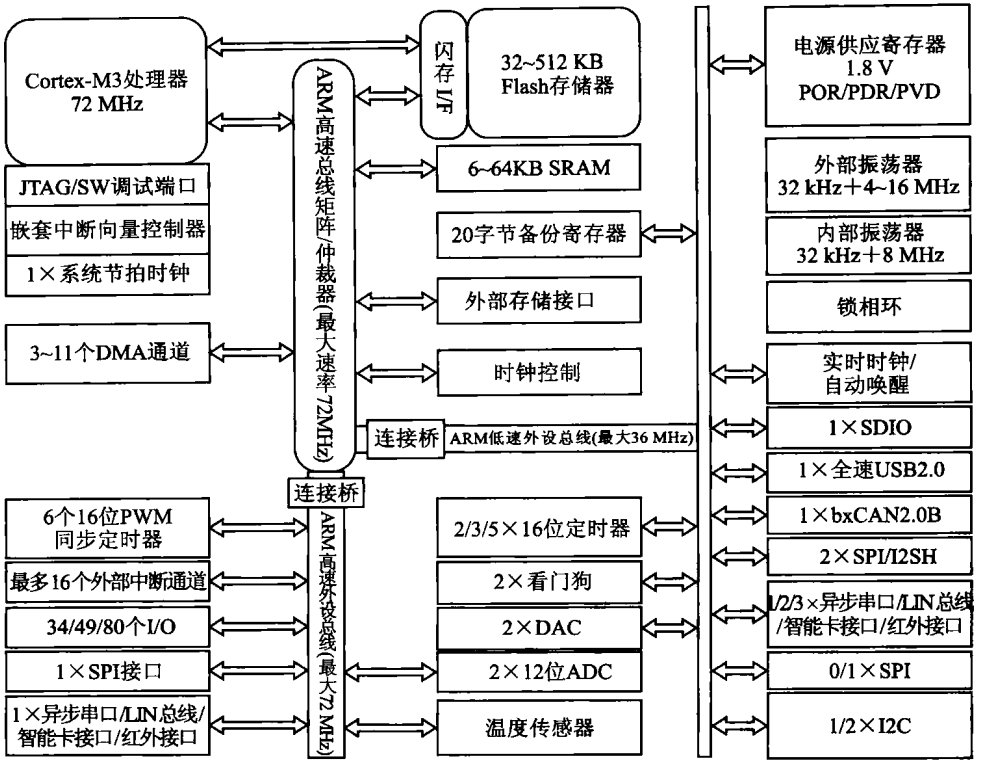

图 1.2.1　增强型 STM32 结构一览

间进行 8 位、16 位或 32 位数据的传输。每个设备都可以向 DMA 控制器请求发送或接收数据。STM32 内部总线仲裁器和总线矩阵将 CPU 数据接口和 DMA 通道之间的连接大大地简化了，这意味着 DMA 单元是很灵活的，其使用方法简单，足以应付微控制器应用中常见的数据传输需求。

为了在具备高性能表现的同时保持低功耗特性，STM32 微控制器在低功耗方面也做了不少努力。它可以在 2 V 供电的情况下运行，同时在所有设备打开且运行在满速 72 MHz 主频的情况下，也仅消耗 36 mA 的电流；在与 Cortex－M3 内核的低功耗模式结合之后只有最低达 2 μA 的电流消耗。即便外部振荡器处在待启动状态，STM32 使用内部 8 MHz 的 RC 振荡器也可以迅速地退出低功耗模式。这种快速进出低功耗模式的特性，更进一步降低了 STM32 微控制器的整体功率消耗，同时能仍然保持器件整体的高性能。

2. 可靠性

当代的电子应用领域，对处理器处理能力的要求越来越高，需要越来越多的精密外设，同时对处理器的运行可靠性要求也越来越高。出于对可靠性的考虑，STM32 配备了一系列的硬