配套课件，精美绝伦，完全公开

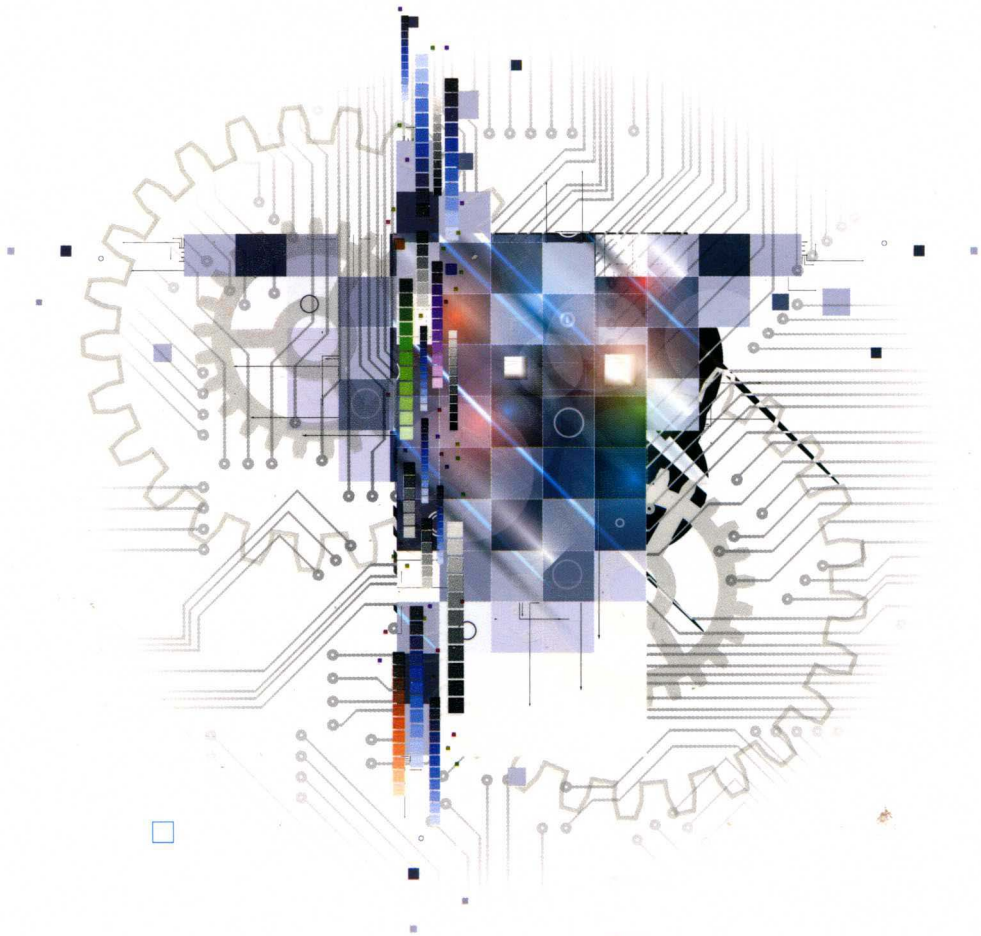
绍兴市重点教材

嵌入式技术与应用 丛书

STM32单片机

应用与全案例实践

沈红卫 任沙浦 朱敏杰 杨亦红 卢雪萍 著



中国工信出版集团

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

嵌入式技术与应用丛书

STM32单片机应用与全案例实践

沈红卫　任沙浦　朱敏杰　杨亦红　卢雪萍　著

電子工業出版社
Publishing House of Electronics Industry
北京 • BEIJING

内 容 简 介

本书为市级重点教材。以基于 ARM 的 STM32 单片机的基本概念、基本原理为主线，详细阐述 STM32 的学习方法与应用系统开发的一般技术。本书在内容组织和框架设计上具有两个鲜明特点：全案例、基于学习者学习。从学习者的角度，精心组织每个章节的内容体系，对 STM32 常用的典型外设模块原理及其应用设计均以完整案例方式呈现。配套课件逻辑严密，思路清晰，制作精良，与教材相得益彰。

本书可作为计算机、电子、通信、机电、自动化及其他相关专业的本、专科学生及研究生的教材，也可作为从事检测、自动控制等领域的嵌入式系统开发工程技术人员的参考用书。

图书在版编目（CIP）数据

STM32 单片机应用与全案例实践 / 沈红卫等著. —北京：电子工业出版社，2017.6
（嵌入式技术与应用丛书）
ISBN 978-7-121-31620-3

Ⅰ. ①S… Ⅱ. ①沈… Ⅲ. ①单片微型计算机 Ⅳ.①TP368.1

中国版本图书馆 CIP 数据核字（2017）第 107707 号

策划编辑：牛平月
责任编辑：桑 昀
印 刷：北京京科印刷有限公司
装 订：北京京科印刷有限公司
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：787×1 092 1/16 印张：21 字数：537.6 千字
版 次：2017 年 6 月第 1 版
印 次：2017 年 6 月第 1 次印刷
定 价：59.00 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）88254888，88258888。

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

本书咨询联系方式：（010）88254454，niupy@phei.com.cn。

前　言

嵌入式系统的发展确实超乎我们的想象。从早期的 8 位单片机，到目前主流的 32 位单片机，其应用已深深渗透于生产、生活的各个方面。作为 ARM 的一个典型系列，STM32 以其较高的性能和优越的性价比，毫无疑问地成为 32 位单片机市场的主流。把 STM32 引入大学的培养体系，已经成为高校广大师生的普遍共识和共同实践。

目前普遍地认为，基于 ARM 的嵌入式系统不仅难教而且难学。究其原因，无外乎三个方面：

（1）功能多导致的问题。功能多导致系统复杂，这样给理解带来诸多困难，使得从传统的 8 位单片机系统转过来的学习者感觉难以适应，因为传统的 8 位单片机，例如 51 单片机，功能相对单一，结构原理也相对简单。

（2）芯片系列多导致的问题。由于基于 ARM 的单片机系列较多，功能、性能差异较大，开发环境又往往不一样，尤其是与 8 位单片机学习者已熟悉的 KEIL C 差异较大，再加上 ARM 单片机出现晚，发展历史没有 8 位单片机长，资料积累远不如 8 位单片机丰富，这些都对学习者提出了挑战、形成了恐慌。

（3）开发模式不一样导致的问题。每个 ARM 系列单片机，往往其开发环境、开发模式是不一样的。以 STM32 为例，开发环境就有好几种，开发模式又有寄存器模式、固件库函数模式两种，再加上各自又可对应基于操作系统和无操作系统的形式，在工程模板的配置方面很多初学者往往不得其要领，所有这些使得传统的单片机学习者在从 8 位单片机向 ARM 迁移的过程中，感觉信心不足，不敢轻易涉足。

作者本身是传统 8 位单片机的一个资深学习者和应用开发者，对上述这些感同身受、体会深厚。正因为如此，由于一个偶然的契机，让作者在 2014 年暑假下定决心一定要破破这个邪，从而开始了长达近三年的学习和教材撰写过程，走上了一个人的朝圣之旅。

正是这不折不扣的三年学习和思考，引领作者在本教材的框架构思和具体撰写中，毫无觉察地将自己设置在学习者的立场和视野上。本教材的撰写自始至终都坚定地遵循了“二二”思想。第一个“二”，即教材撰写的两个原则；第二个“二”，即教材撰写的两个特色。

教材撰写的两个原则：一是基于学习者学习的原则，而不是方便教授者教的原则；二是删繁就简、容易上手的原则，对传统学习者感觉恐慌的部分，围绕应用和实践，采取够用、适用的方式，将其简化，减少学习恐惧，对诸如工程模板配置等问题从根本上、本质上阐述到位，使学习者真正理解为什么要这样做，从而避免云里雾里、一知半解，达到得心应手、运用自如。

教材撰写的两个特色：一是围绕学习者学习；二是全案例驱动。具体地说，就是完全站在学习者学习的角度，设计整个教材的逻辑关系，组织每个章节的内容体系，在简明扼要地阐述 STM32 常用的每一个典型外设模块的原理的基础上，围绕其应用，均以一个以上完整案例的形式讨论其设计精髓，并在教材的最后给出了一个完整的工程案例，所有这些案例的硬件和软件完全公开、毫无保留，因此十分有利于学习者学习和模仿。大量的教学实践表

明，模仿是学习单片机最为成功的方式之一，它不仅可以让学习者产生成就感，而且可以较快地激发学习者的兴趣和动力。

本教材的第一部分讨论了怎么学 STM32 的问题，分别从学习 STM32 的基础要求、STM32 的基本架构和大致原理、学习 STM32 的基本方法、学习 STM32 需要哪些工具或平台等几个方面加以阐述。

教材的第二部分围绕一个 GPIO 输出的简单案例，讨论了 STM32 应用开发所必需的开发环境配置（包括模板的建立）、程序的下载与调试、STM32 程序开发的三种模式等问题。

教材的其余部分，分别通过一个及以上的完整案例，讨论了 STM32 中最为典型的外设与功能模块，即 GPIO 输入/输出、延时的 3 种实现（延时函数、SysTick、定时器中断）、TIMER 与 PWM、USART、基于液晶和按键的人机界面、I^2C 与 SPI、A/D、D/A、DMA、中断等的工作原理、应用设计、程序实现。

教材的最后部分，讨论了一个基于线性 CCD 路径识别的综合性工程案例。这个案例可帮助学习者进一步建立模块化思想，提高设计与开发 STM32 的综合应用系统的能力与信心。

本教材的所有案例均经过作者精心设计并一一经过实验验证。所有案例的功能要求完整、注释完整、代码完整，真正做到了全公开、全透明、无保留。

一点建议：从学习入门和一般应用的角度，作者倾向于不要先花很多精力去学嵌入式实时操作系统（如 μC/OS），原因有二：一方面，因其体系和概念过于复杂、抽象，初学者难以驾驭，学习起来会非常困难，学习效率会异常低下，这样反而影响最重要的、最根本的内容的学习，可能会使原本不足的学习兴趣和动力出现“断崖式”下降；另一方面，对于一般的应用，多数是不需要基于操作系统的，况且，多数观点认为 STM32 并不十分适合嵌入操作系统。

教材的编写是一个艰难和孤独的过程，一本好的教材的出版更是需要作者做到心无旁骛、摒弃杂念。在整整三年的撰写和实验验证过程中，作者对此深信不疑。本教材绝大多数的内容均为作者原创，编写风格也不同于大多数教材的模式（因为将显得复杂的原理分解至各个功能模块去阐述和讨论，因此较好地迎合了学习者的学习规律）。可以不自谦地说，这是一本真正意义上以学习产出（OBE）为导向的教材。

本教材得到了绍兴文理学院浙江省新兴特色专业自动化专业建设项目经费的资助，是 2015 年绍兴市重点建设教材。

本教材由绍兴文理学院沈红卫教授、任沙浦副教授、朱敏杰讲师、卢雪萍讲师，浙江工业职业技术学院杨亦红讲师等共同完成，沈翊、涂强、章英、章清、金梦琪、傅飞娜、葛琼、赵伟强也为本书贡献了心智，绍兴文理学院自动化专业 13 级项烨雯、12 级陈剑泓等同学参与了部分图表的绘制。在本教材的编写过程中，参阅了许多资料，在此对本教材参考资料的作者表示诚挚感谢，对不能一一标明来源的资料的作者表示真诚的歉意和敬意。对直接、间接为本教材的出版倾注智慧、付出心力、提供帮助的所有人，作者都心怀满满的感谢！

由于水平所限，书中难免存在错误和不周之处，恳请同行专家和学习者不吝指正。

本教材配套课件逻辑严密、思路清晰、制作精良，可登录华信教育资源网（http://www.hxedu.com.cn）免费注册后下载。

沈红卫

于绍兴风则江边

2017 年 1 月 20 日

目　录

第 1 章 如何学习 STM32

本章导览

从 8 位单片机转到基于 ARM 的 32 位单片机 STM32，这个过程需要通过合适的方法跨越。由于 STM32 的功能多，其原理又与传统的 8 位单片机（主要是 MCS-51 系列）完全不同，所以本章重点讨论学习 STM32 的方法和策略问题，主要内容如下：

- STM32 的基本架构和基本原理。
- 学习 STM32 的基本方法。
- 学习 STM32 需要的工具或平台。
- STM32 程序开发的几种模式。

1.1 学习 STM32 必须具备的知识基础

为了学习 STM32，必要的知识基础是需要掌握的。它们主要包括以下部分内容。

1. 电路原理

作为理解和学习硬件最基础的知识，必须了解电流、电源、电阻、电容等的概念和基本属性、基本关系，能正确选择和使用电阻、电容等基本元器件。

2. 数字电路、模拟电路

掌握二极管、三极管的基本工作原理，掌握二极管的导通和截止的条件，掌握三极管的饱和导通和截止（开与关）的条件，基本掌握 A/D、D/A 转换的基本原理与性能指标，初步掌握直流稳压电源的原理与工作要求等。

3. 单片机

如果有 8 位单片机原理的学习经历和应用开发的实践经验，例如 MCS-51 单片机，则肯定对学习和理解 STM32 是极为有利的。但这个不是必备条件，可以作为选项，只要方法得当，也是可以从零起点学习 STM32 单片机的。

4. 计算机语言

必须有比较扎实的 C 语言基础，能使用 C 语言开发一定复杂度的应用系统，因为 STM32 开发基本上都是基于 C 语言的。

5. 实践能力

能比较熟练地使用数字式万用表对电阻、电压等物理量进行检测、对电路的通断进行判断，能熟练使用电烙铁进行焊接，若能使用数字示波器则当然更好。

1.2 STM32 的基本架构和基本原理

如果你是学过 8 位单片机的，例如 51 单片机，那么对于理解 STM32 的系统架构和功能模块是十分有利的，毕竟它们都属于单片机范畴（英文为 Microcontroller）。只不过前者是 8 位单片机（即数据总线是 8 位的），而后者是 32 位单片机（数据总线是 32 位的）。但是，如果你没有学过任何单片机，那么从零基础学 STM32 也不是没有可能，只是在理解时会困难些。学了 C 语言，大家都知道，开发 PC 程序，压根不用了解 PC 的硬件结构和功能部件的特点。但是开发单片机程序，必须知道单片机的内部结构和功能部件的特点和属性，从学习入门的角度而言，初学者往往被 STM32 复杂而多样的内部结构和功能部件所吓到，从而望而生畏、放弃学习。其实，依作者之见，学习者可以不必完全弄清楚硬件结构和原理后才开始 STM32 单片机的学习、应用系统的设计与开发。

下面是对 STM32 单片机的内部结构和功能部件的一个大致描述，在了解这些特点的基础上，就可以开始尝试 STM32 的应用设计与实践。

1.2.1 什么是 ARM

ARM 这个缩写包含两个意思：一是指 ARM 公司；二是指 ARM 公司设计的低功耗 CPU 及其架构，包括 ARM1～ARM11 与 Cortex，其中，被广泛应用的是 ARM7、ARM9、ARM11 以及 Cortex 系列。

1. ARM 公司及其 ARM 架构

ARM 是全球领先的 32 位嵌入式 RISC 芯片内核设计公司。RISC 的英文全称是 Reduced Instruction Set Computer，对应的中文是精简指令集计算机。特点是所有指令的格式都是一致的，所有指令的指令周期也是相同的，并且采用流水线技术。

ARM 公司本身并不生产和销售芯片，它以出售 ARM 内核的知识产权为主要模式。全球顶尖的半导体公司，例如 Actel、TI、ST、Fujitsu、NXP 等均通过购买 ARM 的内核，结合各自的技术优势进行生产和销售，共同推动基于 ARM 内核包括 Cortex 内核的嵌入式单片机的发展。

ARM 的设计具有典型的精简指令系统（RISC）风格。ARM 的体系架构已经经历了 6 个版本，版本号分别是 V1～V6。每个版本各有特色，定位也各有不同，彼此之间不能简单

地相互替代。其中，ARM9、ARM10 对应的是 V5 架构，ARM11 对应的是发表于 2001 年的 V6 架构，时钟频率为 350～500MHz，最高可达 1GHz。

2. Cortex 内核

Cortex 是 ARM 的全新一代处理器内核，它在本质上是 ARM V7 架构的实现，它完全有别于 ARM 的其他内核，是全新开发的。按照 3 类典型的嵌入式系统应用，即高性能、微控制器、实时类，它又分成 3 个系列，即 Cortex-A、Cortex-M、Cortex-R。而 STM32 就属于 Cortex-M 系列。

Cortex-M 旨在提供一种高性能、低成本的微处理器平台，以满足最小存储器、小引脚数和低功耗的需求，同时兼顾卓越的计算性能和出色的中断管理能力。目前典型的、使用最为广泛的是 Cortex-M0、Cortex-M3、Cortex-M4。

与 MCS-51 单片机采用的冯·诺依曼结构不同，Cortex-M 采用的是哈佛结构，即程序存储器和数据存储器不分开、统一编址。

1.2.2　什么是 STM32

STM32 是意法半导体（STMicroelectronics）较早推向市场的基于 Cortex-M 内核的微处理器系列产品，该系列产品具有成本低、功耗优、性能高、功能多等优势，并且以系列化方式推出，方便用户选型，在市场上获得了广泛好评。

STM32 目前常用的有 STM32F103～107 系列，简称“1 系列”，最近又推出了高端系列 STM32F4xx 系列，简称“4 系列”。前者基于 Cortex-M3 内核，后者基于 Cortex-M4 内核。STM32F4xx 系列在以下诸多方面做了优化：

（1）增加了浮点运算；

（2）DSP 处理；

（3）存储空间更大，高达 1M 字节以上；

（4）运算速度更高，以 168MHz 高速运行时可达到 210DMIPS 的处理能力；

（5）更高级的外设，新增外设，例如，照相机接口、加密处理器、USB 高速 OTG 接口等，提高性能，更快的通信接口，更高的采样率，带 FIFO 的 DMA 控制器。

本书侧重于“1 系列”中的 STM32F103 系列，所讨论的内容大部分可用于“4 系列”。

1.2.3　STM32 的内部结构

STM32 跟其他单片机一样，是一个单片计算机或单片微控制器，所谓单片就是在一个芯片上集成了计算机或微控制器该有的基本功能部件。这些功能部件通过总线连在一起。就 STM32 而言，这些功能部件主要包括：Cortex-M 内核、总线、系统时钟发生器、复位电路、程序存储器、数据存储器、中断控制、调试接口以及各种功能部件（外设）。不同的芯片系列和型号，外设的数量和种类也不一样，常有的基本功能部件（外设）是：输入/输出接口 GPIO、定时/计数器 TIMER/COUNTER、串行通信接口 USART、串行总线 I^2C 和 SPI 或 I^2S、SD 卡接口 SDIO、USB 接口等。

根据 ST 的官方手册，STM32F10X 的系统结构图如图 1.1 所示。

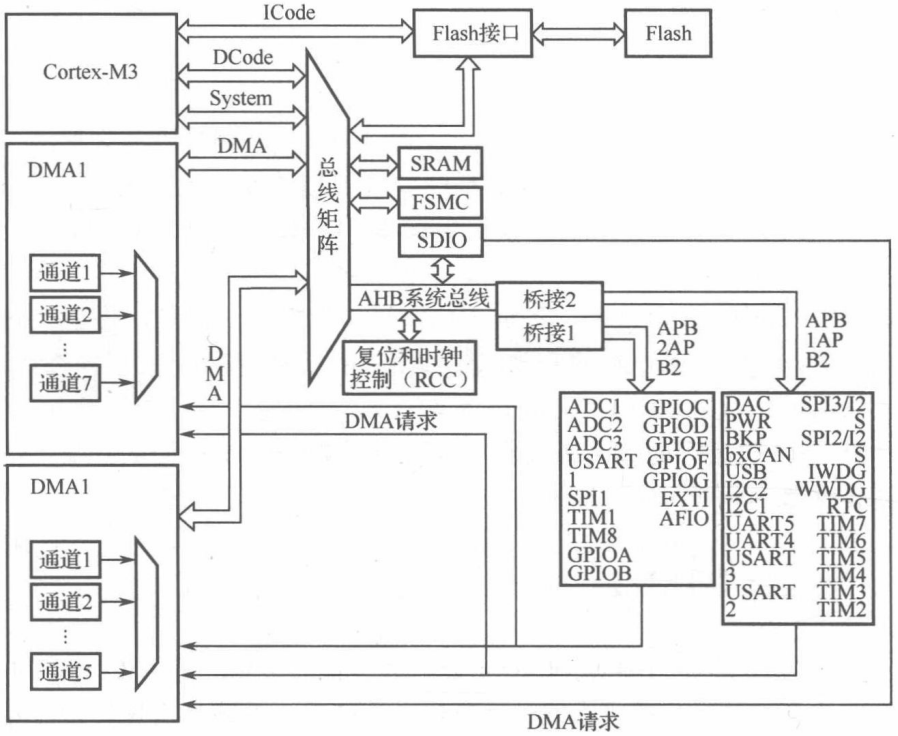


图 1.1　STM32F10X 系统结构图

为更加简明地理解 STM32 单片机的内部结构，对图 1.1 进行抽象简化后得到图 1.2，这样对初学者的学习理解会更加方便些。

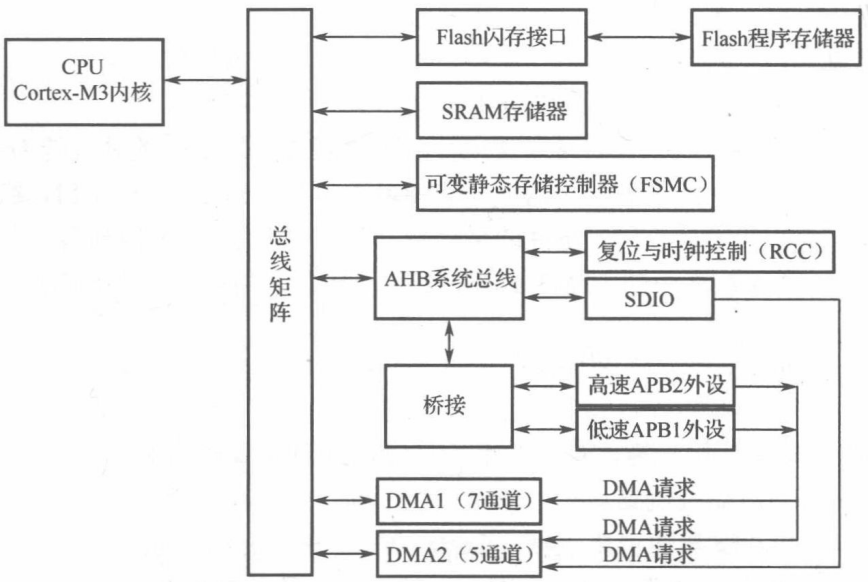


图 1.2　STM32F10X 系统结构简化图

现结合图 1.2 对 STM32 的基本原理做一简单分析，主要包括以下内容。

（1）程序存储器、静态数据存储器、所有的外设都统一编址，地址空间为 4GB。但各

自都有固定的存储空间区域，使用不同的总线进行访问。这一点跟 51 单片机完全不一样。具体的地址空间请参阅 ST 官方手册。如果采用固件库开发程序，则可以不必关注具体的地址问题。

（2）可将 Cortex-M3 内核视为 STM32 的“CPU”，程序存储器、静态数据存储器、所有的外设均通过相应的总线再经总线矩阵与之相接。Cortex-M3 内核控制程序存储器、静态数据存储器、所有外设的读写访问。

（3）STM32 的功能外设较多，分为高速外设、低速外设两类，各自通过桥接再通过 AHB 系统总线连接至总线矩阵，从而实现与 Cortex-M3 内核的接口。两类外设的时钟可各自配置，速度不一样。具体某个外设属于高速还是低速，已经被 ST 明确规定，可参阅图 1.1 标示的信息。所有外设均有两种访问操作方式：一是传统的方式，通过相应总线由 CPU 发出读写指令进行访问，这种方式适用于读写数据较小、速度相对较低的场合；二是 DMA 方式，即直接存储器存取，在这种方式下，外设可发出 DMA 请求，不再通过 CPU 而直接与指定的存储区发生数据交换，因此可大大提高数据访问操作的速度。

（4）STM32 的系统时钟均由复位与时钟控制器 RCC 产生，它有一整套的时钟管理设备，由它为系统和各种外设提供所需的时钟以确定各自的工作速度。

1.2.4　典型型号——STM32F103ZET6

根据程序存储容量，ST 芯片分为三大类：LD（小于 64KB），MD（小于 256KB），HD（大于 256KB），而 STM32F103ZET6 类型属于第三类，它是 STM32 系列中的一个典型型号。以下是它的性能简介：

（1）基于 ARM Cortex-M3 核心的 32 位微控制器，LQFP-144 封装。

（2）512KB 片内 Flash（相当于硬盘，程序存储器），64KB 片内 RAM（相当于内存，数据存储器），片内 Flash 支持在线编程（IAP）。

（3）高达 72MHz 的系统频率，数据、指令分别走不同的流水线，以确保 CPU 运行速度达到最大化 。

（4）通过片内 BOOT 区，可实现串口的在线程序烧写（ISP）。

（5）片内双 RC 晶振，提供 8MHz 和 40kHz 的频率。

（6）支持片外高速晶振（8MHz）和片外低速晶振（32kHz）。其中片外低速晶振可用于 CPU 的实时时钟，带后备电源引脚，用于掉电后的时钟行走。

（7）42 个 16 位的后备寄存器（可以理解为电池保存的 RAM），利用外置的纽扣电池，实现掉电数据保存功能。

（8）支持 JTAG、SWD 调试。可在廉价的 J-LINK 的配合下，实现高速、低成本的开发调试方案。

（9）多达 80 个 GPIO（大部分兼容 5V 逻辑）；4 个通用定时器，2 个高级定时器，2 个基本定时器；3 路 SPI 接口；2 路 I^2S 接口；2 路 I^2C 接口；5 路 USART；1 个 USB 从设备接口；1 个 CAN 接口；1 个 SDIO 接口；可兼容 SRAM、NOR 和 NAND Flash 接口的 16 位总线的可变静态存储控制器（FSMC）。

（10）3 个共 16 通道的 12 位 ADC，2 个共 2 通道的 12 位 DAC，支持片外独立电压基

准。ADC 转换速率最高可达 1μs。

（11）CPU 的工作电压范围：2.0～3.6V。

1.2.5 STM32 的时钟树

STM32 的时钟系统比较复杂，但又十分重要。理解 STM32 的时钟树对理解 STM32 十分重要。下面分五个部分择要对其进行阐述。

1. 内部 RC 振荡器与外部晶振的选择

STM32 可以选择内部时钟（内部 RC 振荡器），也可以选择外部时钟（外部晶振）。但如果使用内部 RC 振荡器而不使用外部晶振，必须清楚以下几点：

（1）对于 100 脚或 144 脚的产品，OSC_IN 应接地，OSC_OUT 应悬空。

（2）对于少于 100 脚的产品，有两种接法：

方法 1：OSC_IN 和 OSC_OUT 分别通过 10kΩ电阻接地。此方法可提高 EMC 性能。

方法 2：分别重映射 OSC_IN 和 OSC_OUT 至 PD0 和 PD1，再配置 PD0 和 PD1 为推挽输出并输出 0。此方法相对于方法 1，可以减小功耗并节省两个外部电阻。

（3）内部 8MHz 的 RC 振荡器的误差在 1%左右，内部 RC 振荡器的精度通常比用 HSE（外部晶振）要低十倍以上。STM32 的 ISP 就是利用了 HSI（内部 RC 振荡器）。

2. STM32 时钟源

在 STM32 中，有 5 个时钟源，分别为 HSI、HSE、LSI、LSE、PLL。

（1）HSI 是高速内部时钟，RC 振荡器，频率为 8MHz。

（2）HSE 是高速外部时钟，可接石英谐振器、陶瓷谐振器，或者接外部时钟源，它的频率范围为 4MHz～16MHz。

（3）LSI 是低速内部时钟，RC 振荡器，频率为 40kHz。

（4）LSE 是低速外部时钟，接频率为 32.768kHz 的石英晶体。

（5）PLL 为锁相环倍频输出，其时钟输入源可选择为 HSI/2、HSE 或者 HSE/2。倍频可选择 2～16 倍，但是其输出频率最大不得超过 72MHz。

3. STM32 时钟树的输入与输出

对于初次接触 STM32 的学习者来说，在熟悉了开发环境的使用之后，往往“栽倒”在同一个问题上，这个问题就是如何理解和掌握时钟树。

众所周知，微控制器（处理器）的运行必须要依赖周期性的时钟脉冲，它往往由一个外部晶体振荡器提供时钟输入为始，最终转换为多个外部设备的周期性运作为末，这种时钟“能量”扩散流动的路径，犹如大树的养分通过主干流向各个分支，因此常称之为“时钟树”。在一些传统的低端 8 位单片机，诸如 51、AVR 等单片机，它们也具备自身的一个时钟树系统，但它们中的绝大部分是不受用户控制的，亦即在单片机上电后，时钟树就固定在某种不可更改的状态。例如，51 单片机使用典型的 12MHz 晶振作为时钟源，则其诸如 I/O 口、定时器、串口等外设的驱动时钟速率便被系统固定，用户将无法更改此时钟的速率，除非更换晶振。

而 STM32 微控制器的时钟树则是可配置的，其时钟输入源与最终达到外设处的时钟速率不再有固定的关系。图 1.3 是 STM32 微控制器的时钟树。要学会 STM32，必须理解时钟树的输入和输出关系。现以图 1.3 中的圆框数字序号标示的部分所示，说明时钟输入与时钟输出之间的关系，输入至输出之间的路径一可表示为①-②-③-④-⑤-⑥-⑦，当然也可以选择路径二：①-⑤-⑥-⑦。此处以路径一为例，做以下具体分析。

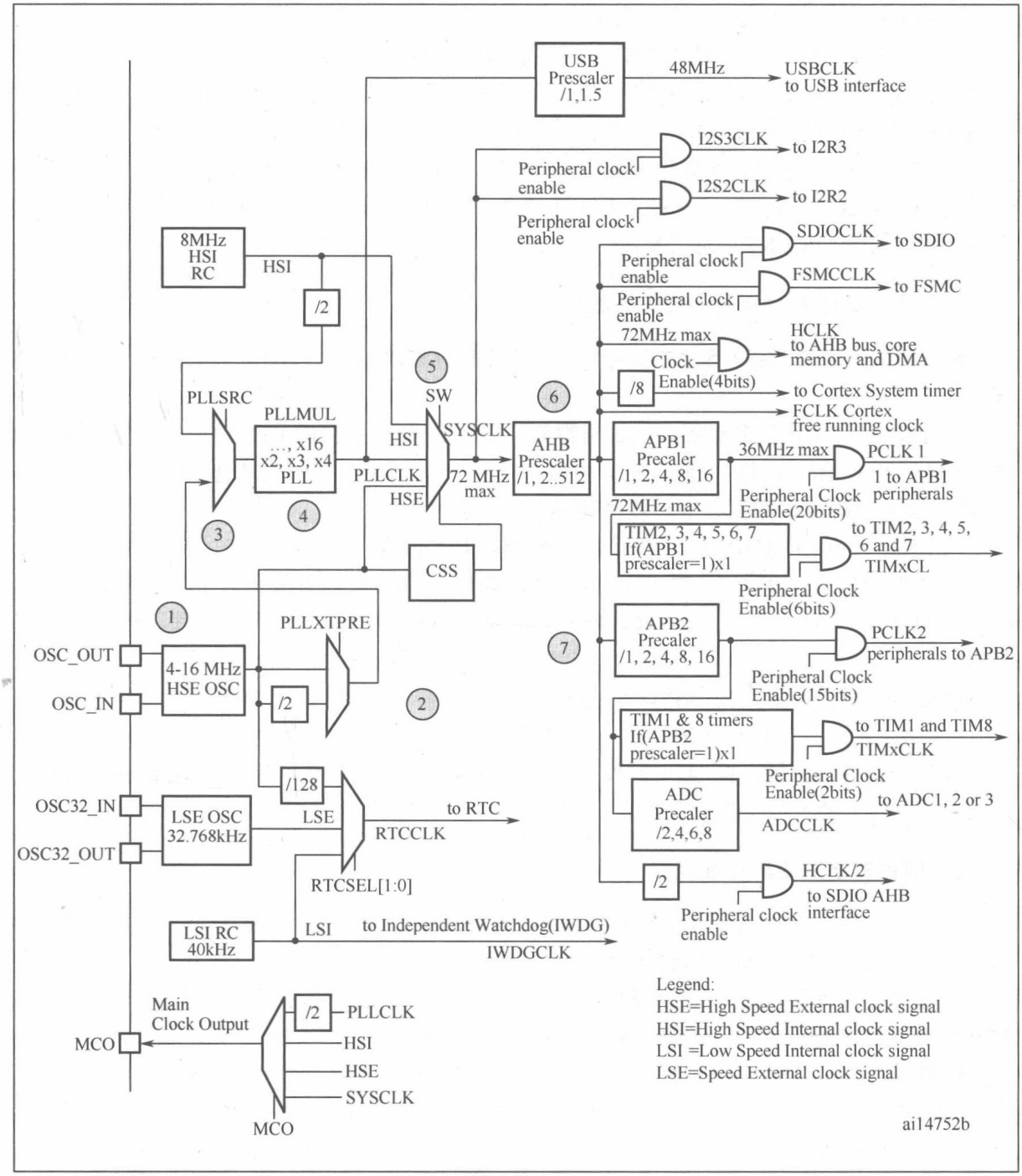


图 1.3　STM32 微控制器的时钟树

①——输入，外部晶振（HSE），可选为 2～16MHz。

②——第一个分频器 PLLXTPRE，可以选择 1 分频或 2 分频。

③——时钟源选择，开关 PLLSRC（PLL entry clock source），我们可以选择其输出，输出为外部高速时钟（HSE）或是内部高速时钟（HSI）。这里选择输出为 HSE。

④——PLL（锁相环），具有倍频功能（输入倍频因子 PLLMUL，2～16 倍），经过 PLL 的时钟称为 PLLCLK。倍频因子设定为 9 倍频，也就是说，经过 PLL 之后，时钟从原来 8MHz 的 HSE 变为 72MHz 的 PLLCLK。

⑤——开关 SW，经过这个开关之后就是 STM32 的系统时钟（SYSCLK）了。通过这个开关，可以切换 SYSCLK 的时钟源，可以选择为 HSI、PLLCLK、HSE。我们选择为 PLLCLK 时钟，所以 SYSCLK 就为 72MHz 了。

⑥——AHB 预分频器（分频系数为 1～512）。如果选为 1，分频系数为 1。

⑦——APB2 预分频器（分频系数为 1, 2, 4, 8, 16）。如果选为 1，则分频系数为 1，所以高速外设 APB2（PCLK2）为 72MHz。

4. STM32 中几个与时钟相关的概念

SYSCLK：系统时钟，STM32 大部分器件的时钟来源。它由 AHB 预分频器分配到各个部件。

HCLK：由 AHB 预分频器直接输出得到，它是高速总线 AHB 的时钟信号，提供给存储器、DMA 及 Cortex 内核，是 Cortex 内核运行的时钟，CPU 主频就是这个信号，它的大小与 STM32 运算速度、数据存取速度密切相关。

FCLK：同样由 AHB 预分频器输出得到，是内核的“自由运行时钟”（free running clock）。“自由”表现在它不来自时钟 HCLK，因此在 HCLK 时钟停止时 FCLK 也会继续运行。它的存在，可以保证在处理器休眠时，也能够采样中断和跟踪休眠事件，它与 HCLK 互相同步。

PCLK1：外设时钟，由 APB1 预分频器输出得到，最大频率为 36MHz，提供给挂载在 APB1 总线上的外设（低速外设）。

PCLK2：外设时钟，由 APB2 预分频器输出得到，最大频率可为 72MHz，提供给挂载在 APB2 总线上的外设（高速外设）。

5. 时钟输出的使能及其流程

在以上的时钟输出中有很多是带使能控制的，如 AHB 总线时钟、内核时钟、各种 APB1 外设时钟、APB2 外设时钟等。

当需要使用某模块时，必须先使能对应的时钟。需要注意的是定时器的倍频器，当 APB 的分频为 1 时，它的倍频值为 1，否则它的倍频值就为 2。

连接在 APB1 上的设备（低速外设）有：电源接口、备份接口、CAN、USB、I^2C1、I^2C2、UART2、UART3、SPI2、窗口看门狗、Timer2、Timer3、Timer4。注意：USB 模块虽然需要一个单独的 48MHz 时钟信号，但它不是供 USB 模块工作的时钟，而只是提供给串行接口引擎（SIE）使用的时钟。USB 模块工作的时钟应该是由 APB1 提供的。

连接在 APB2 上的设备（高速外设）有：GPIO_A-E、USART1、ADC1、ADC2、ADC3、TIM1、TIM8、SPI1、AFIO。