ARM Cortex-M4

自学笔记

——基于Kinetis K60

杨东轩　王　嵩　编著

北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

ARM Cortex－M4 自学笔记

——基于 Kinetis K60

杨东轩　王　嵩　编著

北京航空航天大学出版社

内 容 简 介

本书介绍了飞思卡尔公司推出的 Kinetis 系列微处理器 K60 的原理与开发方法，分为 3 篇，共 20 章：第 1 篇 初识 M4，介绍了 K60 的相关基础，并用一个“Hello World!”实例告诉读者如何完成一个简单的 K60 工程。第 2 篇 新手上路，具体介绍 K60 基本模块的使用，包括 GPIO 模块、串口模块、PIT 模块以及模拟模块等。每个模块都通过实例来讲解，简单易懂，非常适合读者掌握。第 3 篇 高手晋级，不仅包含 K60 复杂模块的应用，例如 DMA、PDB、FTM、RTC、LPTMR、I^2C、SPI、CAN、USB、ENET 和 SDHC，还包含 $\mu C/OS-II$ 的移植等内容。

本书配套资料包含了所有实例的代码，读者可以到 http://www.buaapress.com.cn/的“下载专区”或者 www.lpld.cn 免费下载。

本书面向高等院校电子、电气、计算机、机械电子和仪器仪表等相关专业的本科生和研究生，也适合相关工程师参考阅读。

图书在版编目(CIP)数据

ARM Cortex-M4 自学笔记 ：基于 Kinetis K60 /杨东轩，王嵩编著. -- 北京 ：北京航空航天大学出版社，2013.4

ISBN 978-7-5124-1102-9

Ⅰ. ①A… Ⅱ. ①杨… ②王… Ⅲ. ①微控制器 Ⅳ. ①TP332.3

中国版本图书馆 CIP 数据核字(2013)第 068160 号

ARM Cortex-M4 自学笔记——基于 Kinetis K60

杨东轩 王 嵩 编著

责任编辑 董立娟

*

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

涿州市新华印刷有限公司印装 各地书店经销

*

开本：710×1 000 1/16 印张：28 字数：597 千字

2013 年 4 月第 1 版 2013 年 4 月第 1 次印刷 印数：3 000 册

ISBN 978-7-5124-1102-9 定价：64.00 元

前言

笔者2011年初次接触飞思卡尔公司推出的ARM Cortex－M4内核的微控制器，当时正在为一个机器人项目的主控芯片选型。该项目的上一代产品采用Cortex－M3，虽然其各方面控制性能优越，但是在运算一些算法上显得尤为吃力，于是最终选用了Kinetis系列的K60微控制器，其优秀的数据处理能力以及丰富的外设资源，使得我们的硬件设计具有极高的集成性和稳定性。

如果你是Cortex－M3的开发者，那么就可以像笔者一样很快地投入到Cortex－M4的开发中。如果之前使用过飞思卡尔的其他系列微控制器，那么你也会觉得Kinetis系列非常容易上手。

拉普兰德K60底层库

我们在开发产品的过程中逐步将用到的模块归纳整理，并分别编写了各个模块的驱动函数，最终形成了现在的“拉普兰德K60底层库”。该底层库并不是一开始就包含了很多模块，而是随着学习和使用逐步完善的，尤其在网络上开源该底层库后受到了众多开发者的支持，于是根据使用者反馈的意见和建议，进行了多次更新。网址：www.lpld.cn。

虽然已经针对该底层库编写了详细的函数手册，但是对于初学者来说，真正了解某一模块的工作原理远比仅会使用该模块的驱动函数重要得多。这便是“授人以鱼，不如授人以渔”，因此，作为驱动编写者同样是学习者的笔者来说，一直希望能为初学者总结这样一份学习笔记。但是迫于紧张的工作时间一直未能兑现，直至笔者遇到了北京航空航天大学出版社的编辑才得以实现。

本书主要内容

本书介绍了飞思卡尔公司推出的Kinetis系列微处理器K60的原理与开发方法，分为3篇，共20章：第1篇 初识M4，介绍了K60的相关基础，并用一个“Hello World!”实例告诉读者如何完成一个简单的K60工程。第2篇 新手上路，具体介绍K60基本模块的使用，包括GPIO模块、串口模块、PIT模块以及模拟模块等。每个

模块都通过实例来讲解，简单易懂，非常适合读者掌握。第 3 篇 高手晋级，不仅包含 K60 复杂模块的应用，例如 DMA、PDB、FTM、RTC、LPTMR、I^2C、SPI、CAN、USB、ENET 和 SDHC，还包含 μC/OS－II 的移植等内容。

在写作过程中，笔者以通俗易懂的语言和较少的篇幅介绍模块的工作原理，以较大的篇幅详细介绍 K60 的模块功能和寄存器的使用方法，然后结合底层库的实际代码讲解如何开发 K60 各模块的底层驱动。各章的最后还根据不同的应用实例具体介绍 K60 底层库的使用方法，使读者能够在利用本书学习的过程中真正做到理论与实践相结合。

本书所有的示例工程均基于拉普兰德 K60 底层库开发，运行于拉普兰德开发的 Card 系列 K60 核心板上（采用插卡式设计，便于学习开发），并使用 IAR 6.3 开发环境编译和调试（使用的过程中必须安装不低于该版本的开发环境，试用版可至官网获取），在学习的过程中有任何问题可以到拉普兰德的官方网站进行反馈：http://www.lpld.cn/。

笔者还推荐读者多利用网络资源进行学习，也许困惑自己已久的问题正是别人已经解决的问题。

飞思卡尔官方论坛（英文）：https://community.freescale.com/

飞思卡尔技术社区（中文）：http://www.freescaleic.org/

飞思卡尔智能车制作论坛（中文）：http://www.znczz.com/

致　谢

完成本书并非两位署名作者的全部功劳，其中，参与 K60 底层库开发的人员还包括：陈豪、陈建涛、杨晓玲、谢妤婵和赵瑾；参与内容整理的研究生有叶晓剑和赵小龙。感谢他们对嵌入式开发以及对本书的写作所做出的卓越贡献。

本书配套资源

本书配备了书中所有实例的代码，读者可以到 www.lpld.cn 免费下载，也可以到：www.buaapress.com.cn 的“下载专区”免费下载，也可以通过以下方式与作者索要、交流：

邮箱：laplenden@126.com

QQ 群：184156168（入群验证消息：K60）

与非网论坛专区：www.freescaleic.org/bbs/forum_1278.html

杨东轩

2013 年 2 月

目 录

第 1 篇 初识 M4

第 2 篇 新手上路

第 3 篇 高手晋级

第 1 篇　初识 M4

本书的第 1～3 章为第 1 篇，带领读者初步认识 ARM Cortex－M4，通过对比深入了解其特点及优势所在。接下来具体介绍 Kinetis K60 系列微控制器的特点及功能，并介绍本书所使用的 K60 硬件平台；最后带领初学者完成 M4 学习的第一步，建立并运行一个“Hello Word!”示例工程。

第1章 概 述

在学习飞思卡尔的 Cortex - M4 系列处理器之前有必要对其体系有一个整体的认识。作为本书的第 1 章，读者可以在这里了解 ARM Cortex - M4 内核的特性及其与 Cortex - M3 内核的区别、飞思卡尔 Kinetis 系列处理器的族谱以及它们之间的不同。

1.1 ARM Cortex - M4 简介

ARM Cortex - M4 处理器是由 ARM 公司专门开发的用于信号控制市场的嵌入式处理器，满足了对于效能、易用以及控制和信号混合特性的应用需求。作为一款 32 位的高性能处理器，Cortex - M4 为开发者提供以下诸多好处：

- 结合快速中断处理，提供了出色的信号处理能力；
- 丰富的断点以及跟踪能力，增强了系统的调试性能；
- 具有高效的处理器内核、系统和内存；
- 具有超低功耗的集成睡眠模式和一个可选的深度睡眠模式；
- 可集成存储器保护单元(MPU)以提高系统的鲁棒性。

Cortex - M4 处理器的内核结构如图 1 - 1 所示。

Cortex - M4 处理器构建于一个高性能的处理器核心，具有 3 级流水线的哈佛架构，为一些要求苛刻的嵌入式应用提供了理想的选择。该处理器还具有卓越的能耗效率，通过高效的指令集和广泛的优化设计使得该内核还可以提供高端的处理硬件，比如可选的适应 IEEE754 标准的单精度浮点运算、单指令周期内德 SIMD 乘法和乘法累加能力、饱和算数和专用的硬件除法。

为了促进产品设计成本的进一步降低，Cortex - M4 处理器实现了系统组件的紧密耦合，降低了处理器的面积，同时显著改善了中断处理和系统调试的能力。Cortex - M4 处理器实现了一个基于 Thumb - 2 技术的 Thumb 指令集版本，确保了代码的高密度特性并降低了对程序存储器的要求。因此，Cortex - M4 的指令集在为 32 位处

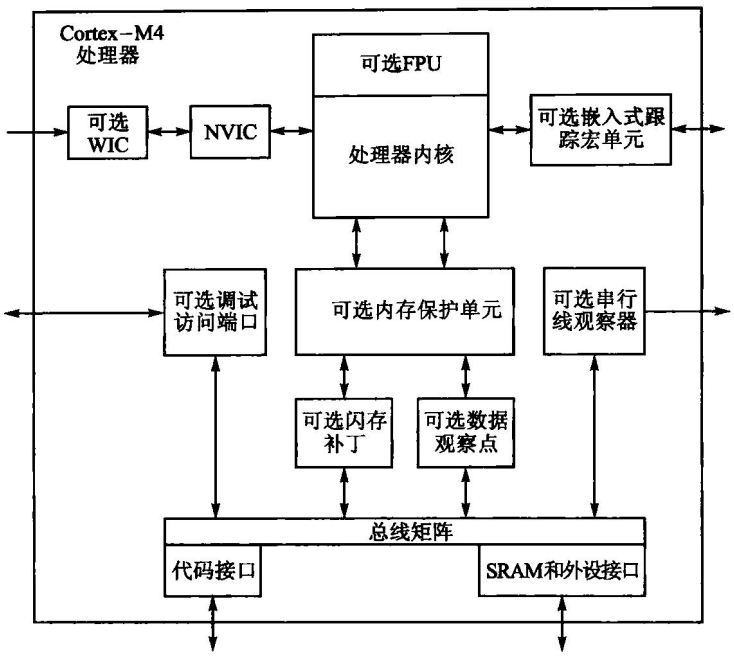

图 1-1 Cortex-M4 结构

理器提供了卓越性能的同时，还保持与 16 位、8 位处理器一样的高代码密度。

Cortex-M4 处理器高度集成了一个可配置的嵌套向量中断控制器(NVIC)，可以提供业界领先的中断性能。NVIC 包含了一个不可屏蔽中断(NMI)，可以提供高达 256 个中断优先级。紧密集成在一起的处理器内核和 NVIC 可以快速地执行中断服务程序(ISR)，从而极大地减少中断延迟。该中断过程是通过硬件堆栈和寄存器来实现的，并具有暂停多个加载和多个储存操作的能力。中断处理程序不需要包装汇编代码和删除任何 ISR 的代码开销。尾链优化也可以显著地减少从中断服务程序切换到其他地方时的开销。为了优化低功耗设计，NVIC 还集成了睡眠模式，并包括一个可选的深度睡眠功能，可以使得整个器件在保持程序状态的情况下快速地关断电源。

1. 系统级接口

Cortex-M4 处理器使用 AMBA 技术提供了众多接口，方便提供高速、低延时的内存访问。同时，支持对未对齐的数据进行访问从而实现原子位操作，使得处理器更快地控制外设、实现系统的自旋锁和线程的安全布尔数据处理。

Cortex-M4 处理器具有可选的内存保护单元(MPU)接口，允许单独地对内存的某个区域进行访问，使应用程序能够利用多个权限级别；还可实现代码的分离和保护、数据和堆栈以逐个任务为基础的功能。在许多嵌入式应用中，如汽车、医疗电子，对以上这些特性的需求变得越来越重要。

2. 可选的集成配置调试

Cortex－M4 处理器可以实现完整的硬件调试方案。该方案通过传统的 JTAG 接口或 2 针串行线调试(SWD)接口可提供系统处理器和内存的高可见度特性，这是微控制器和其他小封装器件的理想调试方案。

对于系统跟踪，Cortex－M4 处理器依靠数据观察点和分析单元集成了一个仪表跟踪宏单元(ITM)。为了使系统事件简单而有效地生成，串行观察器(SWV)可以导出软件生成的信息流、数据跟踪以及单一引脚的分析信息。

可选的嵌入式跟踪宏单元(ETM)的面积远远小于传统的跟踪单元，该单元可以提供优越的指令跟踪捕获功能，使许多低成本微控制器可以轻松实现完整的指令跟踪。

可选的 Flash 补丁和断点单元(FPB)提供 8 个硬件断点比较器，以供调试器使用。FPB 的比较器可在 CODE 区域中支持高达 8 个字的函数重映射功能。该单元使得储存在非可擦除的只读内存中的应用程序可以被修补，例如 Flash 内存。在初始化过程中，只读内存中的应用程序检测可编程存储器中是否有一个需要的补丁。如果有一个需要的补丁，则应用程序通过编程 FPB 单元来重映射多个地址。当这些地址被访问时，访问将被重定向到 FPB 配置中指定的重映射表，这就意味着只读内存中的程序可以通过此方法被修补。

3. Cortex－M4 处理器特性

- 紧密集成的系统外设，减少面积及开发成本；
- Thumb 指令集相结合高代码密度的 32 位性能；
- 可选的适应 IEEE754 标准的单精度 FPU；
- ROM 系统具有可更新代码补丁的能力；
- 系统组件的电源控制优化；
- 集成低功耗的睡眠模式；
- 快速代码执行，允许处理器低速时钟和增加睡眠模式的时间；
- 硬件除法和快速数字信号处理为导向的乘法累加；
- 饱和信号处理算法；
- 为时序严格的应用提供确定性、高性能的中断处理；
- 为安全严格的应用提供可选的内存保护单元(MPU)；
- 串行调试和串行跟踪接口可减少代码分析、调试、跟踪所需的引脚数量。

4. Cortex－M4 内核外设

嵌套向量中断控制器(NVIC)是一个嵌入式的中断控制器，支持低延时中断处理。

系统控制模块(SCB)是处理器的编程模型接口，提供了系统执行信息和系统控