



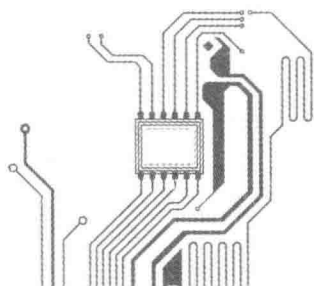
嵌入式软件设计 入门与进阶

—— 基于Kinetis K60/K64

◎ 杨 熙 苏 勇 刘屹霄 编著

- ◎ 一线工程师亲手打造的NXP Kinetis系列微控制器入门与进阶图书
- ◎ 以主题来组织材料，以一个个“技术小品”来阐述遇到的问题和解决方法
- ◎ 分享实际项目的开发笔记，分享嵌入式开发的乐趣

清华大学出版社



嵌入式软件设计 入门与进阶

—— 基于Kinetis K60/K64

◎ 杨 熙 苏 勇 刘屹霄 编著

清华大学出版社
北京

内 容 简 介

本书以 ARM Cortex-M4 为内核的恩智浦(NXP)Kinetis 系列微控制器作为硬件平台,从嵌入式软件工程师的实践角度出发,阐述了基于 MCU 的嵌入式软件开发中的若干细节问题。与目前的入门类书籍中对片上外设及固件库软件的说明指导相比,书中对实际开发中的具体问题进行了更为详细的分析;从零开始搭建软件开发平台,不仅描述了工程模板的创建,还具体分析了芯片的软硬件启动过程,尤其对链接过程及典型应用进行了详述;介绍了几种典型的嵌入式系统开发框架,根据具体的应用场景选择程序的组织方式;在低功耗和 DMA 的相关章节中介绍了降低产品功耗与优化性能的典型做法及思路;而作为运行在单片机上的命令行系统、CMSIS-DSP 运算库及 git 版本管理系统,都是软件工程人员常用的功能调试与代码维护的常用工具,对这些内容也进行了介绍。

本书可作为通信工程、电子信息工程、计算机、微电子等专业相关课程的教材,也可供具有专业背景并有一定 C 语言及嵌入式开发基础的在校大学生、研究生以及对嵌入式技术感兴趣的读者参考。

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

嵌入式软件设计入门与进阶:基于 Kinetis K60/K64/杨熙,苏勇,刘屹霄编著. —北京:清华大学出版社,2017

(电子设计与嵌入式开发实践丛书)

ISBN 978-7-302-44207-3

I. ①嵌… II. ①杨… ②苏… ③刘… III. ①微处理器—系统设计 IV. ①TP332

中国版本图书馆 CIP 数据核字(2016)第 152389 号

责任编辑:刘 星 梅荣芳

封面设计:刘 键

责任校对:时翠兰

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 装 者:北京鑫丰华彩印有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:13.25 字 数:321 千字

版 次:2017 年 1 月第 1 版 印 次:2017 年 1 月第 1 次印刷

印 数:1~2000

定 价:39.00 元

产品编号:067928-01

前言

在经典的 51 单片机时代诞生了许多实用的技术书籍,覆盖了从硬件架构到软件应用的方方面面,这些作品培养了大量单片机工程师,对单片机技术的发展提供了非常大的助力。现如今,随着 ARM 单片机的兴起和繁荣,相关的书籍不断涌现。然而,从作者的学习经历来看,目前关于 ARM 单片机书籍的描述,重点还是放在参考手册和固件库上,也就是说,书中的内容直接或间接地来源于单片机原厂的参考手册或是固件库。诚然,这些知识对于单片机开发非常重要,它们是使用 ARM 单片机进行开发的基础。但是,随着 ARM 单片机的普及和开发门槛的降低,单片机工程师们对 ARM 技术书籍的需求远不止于此,在实践中,还有更广阔的空间需要探索。本书作者曾经也是单片机开发的初学者,经过了多年对 ARM 单片机的学习和实践,在不断解决问题的过程中,积累了一些心得和体会。这些源自于实践的经验,不同于手册中已有的内容,更多的是通过实验和具体的应用而进行的总结,具有比较高的实用价值,对具体的项目应用有直接的参考意义。

本书由来

提及作者学习 ARM 单片机的过程,就不得不聊一点飞思卡尔杯智能汽车比赛的经历。2011 年,作者(其中之一)带队参加第六届“飞思卡尔”杯智能车比赛(电磁组),开始在使用 K60 单片机,这是由飞思卡尔半导体公司推出的全球首款基于 ARM Cortex-M4 内核的单片机,在之后的几年里广为流行。作者在参加比赛的过程中受益颇多,在明确的项目目标(更快、更稳)的驱动下,将所有相关的知识整合起来,全面地解决小车系统中的每一个具体问题,这刚好是一个完整的单片机工程师从入门到成熟的学习过程。在制作和调试小车的过程中,仅仅依赖芯片手册的信息不足以解决所有问题,更多的问题还需要搜集信息、思考、测试、再思考、再测试的反复过程才能取得最终的解决方案。这些通过实践积累下来的经验总结是最宝贵的财富。早年对开发问题的总结,也催生了撰写一本关于开发单片项目经验书籍的念头。

参加工作后,作者有幸继续使用 Kinetis 系列单片机的开发平台进行单片机系统软件的设计工作。作为常年工作在 Kinetis 微控制器底层驱动和操作系统接口层的应用工程师,本着进一步对芯片功能的使用和理解的目的,将近几年在工作岗位和项目中的技术细节整理出来。为了方便使用 Kinetis 产品的广大读者和应用工程师,作者对各类实际工作中看似模棱两可的功能性配置过程做了较为详细的解释,从系统上电启动的最初步骤到进入

Foreword

应用系统之后的环节都进行了较为详细的演示,这样做的目的是为了更方便读者在遇到一些未知的故障时能够通过本书中讲解的内容快速将故障定位,从而缩短产品开发周期,为应用开发奠定基础。同时本书还介绍了嵌入式软件开发过程的一些辅助工具,通过对这些工具的了解可以帮助读者在单片机开发工作中锦上添花。

本书特色

本书从 NXP Kinetis 系列产品的综合介绍开始,对 Kinetis 芯片的一些功能细节做了较为详细的介绍。书中引用了作者团队开发和使用多年的 CHK 开源 Kinetis 产品驱动库,读者可以自行在网络上下载并使用,这些完全免费。CHK 开源驱动软件库开发至今多年,在业界有着较好的口碑,可以帮助有 MCU 开发经验的用户几乎零时间进入 Kinetis 产品的应用开发工作,到目前为止,CHK 软件库已经可以支持多个系列 Kinetis 的衍生型号开发,其他一些子型号只要稍加修改便可使用。

由于目前市面上已经有较多关于恩智浦 Kinetis 系列单片机的书籍,但大多是针对入门学习者或是刚接触 Kinetis 单片机的工程师,着重偏向外设模块介绍,如简单的 GPIO 使用、点亮小灯等。再例如串口通信,只是介绍收发数据的相关寄存器,然后以打印或接收一些串口数据作为示例,强调快速见效。而本书主要面向具有一定经验的嵌入式开发者,根据作者实际的开发经验,在书中总结了一些使用 Kinetis 控制器进行嵌入式软件开发过程中用到的实用技术,以“第一视角”讲述一线工程师的应用心得,实用价值高。另外,本书内容绝不只限于 Kinetis 系列控制器,其中很多是所有嵌入式开发工程师经常碰到的问题。本书对技术点的描述深入细致,内容丰富,是一本可读性强的读物。

在内容的组织形式上,本书并未采用传统教科书式的顺序结构,而是以主题组织材料,将每个章节整理成一个个“技术小品”。每个“技术小品”相当于是一个解决问题的小故事,将相关信息放在一起说明中心问题,这种内容组织的方式直接来源于作者在项目开发过程中整理的开发笔记。这样做,对于读者而言,也可以减轻阅读压力,每阅读完一个章节,就可以了解到一个主题。对于闲暇时间有限的单片机工程师来说,这样组织内容的方式非常便于利用零星的时间进行阅读。

本书读者对象

本书主要面向有一定编程基础的嵌入式 MCU 工程师、高校师生、创客以及所有热爱嵌入式技术的朋友们。本书特别介绍了大量关于 Kinetis 系列产品的入门及应用知识,包括常用外设的编程方法、寄存器,以及飞思卡尔系列单片机一些特有的功能等,这些知识对于想了解该系列产品的朋友来说具有很好的参考价值。

特别提醒读者的是,鉴于本书完稿之时正处于飞思卡尔半导体公司和恩智浦半导体合并过程中,所以以前的“飞思卡尔 Kinetis 系列”在本书出版时已经更替为“恩智浦(NXP) Kinetis”系列。

CHK-LIB Kinetis 驱动固件库是由本书作者自行开发的针对 Kinetis 的底层驱动库,支持数十种 Kinetis 微控制器型号,使用简单并配套详细的使用例程,可以帮助用户以极短的时间入手 Kinetis 系列微控制器的应用编程,大大缩短开发周期,CHK-LIB 驱动库已经开发多年,历经时间检验,稳定可靠。它与目前 NXP 官方的 SDK 2.0 互为补充,SDK 2.0 侧重与以芯片为中心,大而全的设计驱动 API,而 CHKinetis 固件库则以应用为中心来设计驱动 API,讲究实用简单,砍掉了一些不常用的外设功能。API 也比官方的 SDK 2.0 少很多。目

前,CHKinetis 依然在持续更新中。读者可以前往开源中国(OS China)进行下载,地址链接如下:<http://git.oschina.net/yandld/CH-K-Lib>。

本书的作者团队来自于单片机开发领域,包括面向产品的应用软件工程师以及致力于提供固件库服务的系统软件工程师,由于常年与程序代码和开发板为伍,但是作为技术书籍的作者尚显文笔生疏,书中所用描述之辞,如有不当之处恳请见谅。

仅以此书献给曾经的飞思卡尔和一代藉由“飞思卡尔”智能车比赛成长起来的年轻工程师们!

作 者

2016 年 5 月

目 录

第 1 章 Kinetis 平台概述	1
1.1 ARM-Cortex 系列简介	1
1.1.1 ARM Cortex-A	1
1.1.2 ARM Cortex-R	2
1.1.3 ARM Cortex-M	2
1.2 恩智浦(NXP)Kinetis 系列 MCU 简介	2
1.2.1 Kinetis 产品家族介绍	2
1.2.2 Kinetis 系列芯片的命名规则及选型	4
1.3 开发平台硬件电路分析	6
1.3.1 MCU	6
1.3.2 复位	7
1.3.3 时钟	7
1.3.4 调试接口	8
1.3.5 LED	8
1.3.6 串口	9
1.3.7 SPI Flash	9
1.4 软件环境搭建	10
1.4.1 MDK 安装下载	10
1.4.2 串口驱动	12
1.4.3 Jlink 驱动安装	13
1.4.4 下载本书的示例代码	13
1.5 编译实例 HelloWorld 工程	14
1.6 新建工程	21
1.7 本章小结	27
第 2 章 从零开始	28
2.1 样例程序	28

Contents

2.2	芯片上电	29
2.3	启动软件系统	33
2.3.1	分配堆和栈的空间	33
2.3.2	中断向量表	36
2.3.3	Flash 配置选项	40
2.3.4	复位中断入口到用户 main 函数	44
2.4	分散加载文件	48
2.5	本章小结	49
第 3 章	链接命令文件	50
3.1	命令方式存在的链接配置参数	50
3.2	创建 sct 文件模板	51
3.3	SCF 应用——直接在 RAM 中调试程序	55
3.4	SCF 应用——将程序复制到 RAM 中执行	58
3.5	本章小结	60
第 4 章	嵌入式系统软件的典型开发框架	61
4.1	概述	61
4.2	裸机嵌入式系统的组成	62
4.3	典型的裸机软件系统实现框架	63
4.3.1	基本的超循环结构	63
4.3.2	使用中断的前后台结构	71
4.3.3	完全依赖中断的事件驱动系统	76
4.3.4	状态机驱动系统	79
4.4	本章小结	88
第 5 章	低功耗应用设计	89
5.1	概述	89
5.2	K64 中的低功耗模式介绍	89
5.3	功耗状态机模型的运作	94
5.3.1	进入 STOP 模式操作序列	94
5.3.2	退出 STOP 模式操作序列	95
5.3.3	中断进入 STOP 模式	96
5.4	低功耗所涉及的模块详解	96
5.4.1	ARM 核的系统控制寄存器	96
5.4.2	系统模式控制器	98
5.4.3	低功耗模式下的频率限制	100
5.4.4	低漏唤醒单元	101
5.5	低功耗应用开发典型模式	101

5.6	K64 低功耗编程心得	108
5.7	本章小结	109
第 6 章	DMA 的基本概念	110
6.1	DMA 的基本概念	110
6.2	DMA 的工作原理	111
6.3	K64 的 DMA 介绍	113
6.3.1	TCD	113
6.3.2	主循环和子循环	114
6.3.3	eDMA 通道请求使能开关	115
6.3.4	DMA 的触发源	115
6.3.5	DMA 传输完成的判断	117
6.3.6	关于 SMOD 和 DMOD 周期性地址传输	118
6.3.7	关于 DREQ 位	118
6.4	使用 DMA 实现内存间搬运的实例	119
6.4.1	DMA TCD 的配置	119
6.4.2	等待 DMA 传输完成	120
6.4.3	实验效果	121
6.5	本章小结	121
第 7 章	DMA 在串行通信总线中的应用	123
7.1	通过 DMA 来进行串口数据收发	123
7.1.1	DMA 触发条件的确定	123
7.1.2	DMA TCD 的配置	124
7.2	使用 DMA 进行 SPI 通信	125
7.2.1	SPI 通信的一般模型	126
7.2.2	DMA 触发条件的确定	127
7.2.3	DMA TCD 的配置	128
7.3	本章小结	133
第 8 章	DMA 实现 ADC 扫描触发链	134
8.1	从应用案例开始	134
8.1.1	明确需求	134
8.1.2	围绕 ADC 的软触发特性设计触发链	135
8.2	触发链的设计与实现	136
8.2.1	DMA0->ADC0: 实现 DMA0 自动搬运命令字触发 ADC 转换	136
8.2.2	ADC0->DMA1: 实现 ADC 转换完成后触发 DMA1 自动搬运 转换结果	141
8.2.3	PIT0->DMA0: 实现 PIT0 定期触发 DMA0 控制转换的节奏	145

8.2.4 见缝插针,自定义更加丰富的操作接口	150
8.3 本章小结	155
第9章 使用命令行来帮助开发	156
9.1 概述	156
9.2 Shell 的引入	156
9.2.1 单片机中的 Shell	158
9.2.2 单片机 Shell 的输入和输出	159
9.3 简单的 Shell 软件设计	159
9.3.1 构建 Shell 命令结构体	159
9.3.2 获取输入命令	160
9.3.3 解析命令	162
9.3.4 调用对应的函数并返回结果	164
9.4 应用实例	166
9.5 本章小结	168
第10章 使用 CMSIS-DSP 数学运算库	169
10.1 什么是数字信号处理	169
10.2 浮点运算与定点运算	170
10.3 CMSIS-DSP 简介	172
10.4 CMSIS-DSP 初探	173
10.4.1 DSP 库的分类	173
10.4.2 使用 CMSIS-DSP 库	174
10.5 基础数学函数	176
10.5.1 向量绝对值运算	176
10.5.2 向量加法运算	177
10.5.3 向量点积	178
10.5.4 向量(元素)乘积	178
10.5.5 逆向量(反向量)	179
10.5.6 向量平移	179
10.5.7 向量扩围	180
10.6 快速功能函数	180
10.6.1 快速余弦函数(Cosine)	180
10.6.2 快速正弦函数(Sine)	181
10.6.3 快速平方根(Square Root)	181
10.7 本章小结	182
第11章 专业的软件版本控制系统——git	183
11.1 git 的诞生	183

11.1.1	git 的诞生.....	184
11.1.2	git 的核心——数据库.....	184
11.1.3	远程数据库和本地数据库	185
11.2	git 入门笔记.....	185
11.2.1	下载安装 git	185
11.2.2	初始化代码仓库	186
11.2.3	添加文件并提交修改	187
11.2.4	增删改查 git 仓库中的文件	189
11.2.5	退回到某一个 commit	192
11.2.6	使用远程仓库	194
11.2.7	克隆远程库	197
11.3	本章小结	197
后记		198
参考文献		199

Kinetis 平台概述

1.1 ARM-Cortex 系列简介

从现代意义上的计算机诞生至今,无论规模如何,计算机的五大组成部分从未改变:①控制器(Controller);②运算器(ALU);③存储器(Memory);④输入设备(Input);⑤输出设备(Output)。

无论多么小型的计算机设备(嵌入式设备)都是如此,嵌入式微控制器(MCU)也不例外。虽然在计算机科学中,通常把控制器和运算器结合在一起形成中央处理器(CPU),但从嵌入式的视角来看,PC的每一个组成部分几乎无处不是嵌入式系统,即便是PC所使用的CPU,本身也是一个嵌入式系统,内部也包含了计算机系统的五大组成部分。为了避免术语上的冲突,一般将MCU中的控制器和运算器合称为内核(Core),它主要负责指令执行和算术运算,ARM就是一家设计这样的内核的公司。ARM公司的产品从早期的ARM7、ARM9到后来的Cortex-M系列和Cortex-A系列,都是RISC(精简指令集计算机)类型。

ARM公司在经典处理器ARM11以后的产品停止使用“ARM+数字”的命名方式,而全部改用Cortex命名,并分成A、R和M三大类,旨在为各种不同的市场提供服务。

Cortex-M系列属于ARMv7架构,这是到2010年为止ARM公司的指令集架构。ARMv7架构定义了三大分工明确的系列:

(1) A系列,面向高端的基于虚拟内存的操作系统和用户应用,主要应用在手机、平板等多媒体处理领域;

(2) R系列,针对安全及实时操作系统、工业机器人等;

(3) M系列,针对低功耗系统或者称之为微控制器(MCU)系统。

1.1.1 ARM Cortex-A

ARM Cortex-A系列应用型处理器可向托管丰富OS(Operating System)平台和用户应用程序的设备提供全方位的解决方案,从超低成本手机、智能手机、移动计算平台、数字电

视和机顶盒到企业网络、打印机和服务器的解决方案。高性能的 Cortex-A15、可伸缩的 Cortex-A9、经过市场验证的 Cortex-A8、高效的 Cortex-A7 以及 Cortex-A5 处理器,均共享同一架构,因此具有完全的应用兼容性,支持传统的 ARM、Thumb 指令集和高性能紧凑型 Thumb-2 指令集。以上处理器内核版本均基于 ARMv7-A 子架构,其中 ARMv7-A 子架构的最后一代产品是 Cortex-A17。2013 年后,Cortex-A 系列架构升级为 64 位,新版本的内核架构被命名为 ARM-v8-A,主要代表产品有 Cortex-A53、Cortex-A57 以及 2014 年年底推出的 Cortex-A72,这类内核的处理器目前主要针对移动终端,主要服务对象为手机或者高端的平板电脑。

1.1.2 ARM Cortex-R

ARM Cortex-R 实时处理器为要求可靠性、高可用性、容错功能、可维护性和实时响应的嵌入式系统提供高性能计算解决方案。

Cortex-R 系列处理器的优越性已经在数以亿计的产品中得到验证,其成熟的技术使得产品能够快速上市。Cortex-R 利用广泛的 ARM 生态系统、全球和本地语言以及全天候的支持服务,保证快速、低风险的产品开发。

1.1.3 ARM Cortex-M

ARM Cortex-M 处理器系列是一系列可向上兼容的高能效、易使用的处理器,这些处理器旨在帮助开发人员满足未来嵌入式应用的需要。这些需要包括以更低的成本提供更多功能、不断增加连接、改善代码重用性和提高能效等。Cortex-M 系列主要适用于对成本和功耗敏感的 MCU 和终端应用。

下面给出不同 Cortex-M 系列的一些典型应用。

- (1) Cortex-M0: 极低成本,可用于触摸屏控制器和电源管理模块。
- (2) Cortex-M0+: M0 的改良版,可用于物联网传感器节点和蓝牙智能收发器。
- (3) Cortex-M3: 可用于智能可穿戴设备和 WiFi 收发器。
- (4) Cortex-M4: 可用于智能手表、高端电机控制、工业仪表、工业显示面板。
- (5) Cortex-M7: 可用于高端耳机和汽车电子应用。

1.2 恩智浦(NXP)Kinetis 系列 MCU 简介

1.2.1 Kinetis 产品家族介绍

Kinetis 系列微控制器是飞思卡尔公司于 2010 年下半年推出的基于 ARM Cortex-M4 内核的微控制器,是业内首款 Cortex-M4 内核芯片。Cortex-M4 内核是 ARM 公司 2009 年下半年发布的嵌入式内核。

Kinetis 微控制器(MCU)包含多个硬件和软件相互兼容的 ARM Cortex-M0+、Cortex-

M4 和 Cortex-M7 MCU 系列,具备出色的低功耗性能、可扩展性和特性集成特性。

目前,Kinetis 系列仍然在以飞快的速度扩张,图 1-1 展示了 Kinetis 家族中的八大阵营。

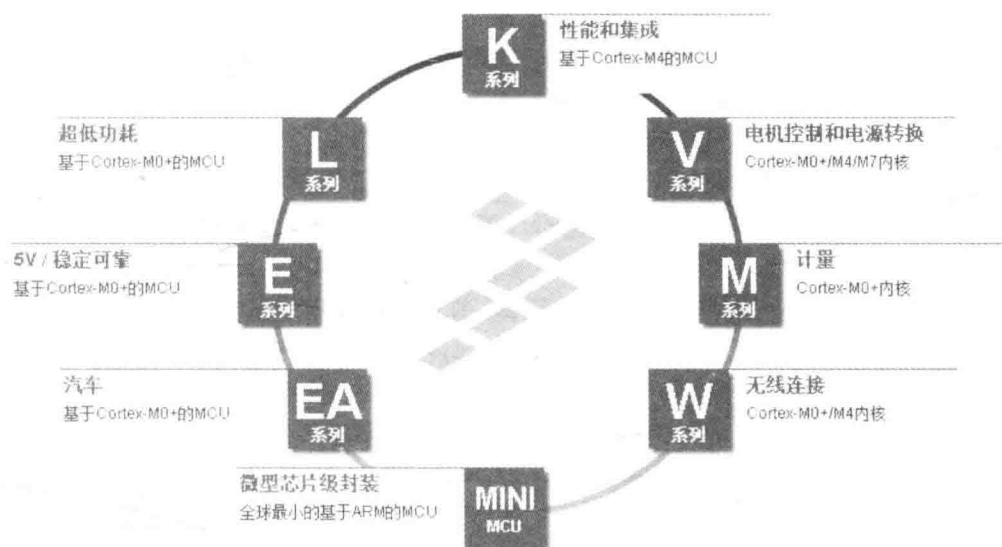


图 1-1 Kinetis 产品家族

下面简要介绍这几种 Kinetis 系列的特点。

(1) K 系列: 最早出现的系列,是 Kinetis 系列中的主力军。其中 K60 是第一款 Kinetis 型号,到目前为止仍然是最为广泛应用的 Kinetis 型号之一。该系列的通信性能和集成性能都很强,能够提供可扩展的数据连接、通信、HMI(人机交互)和安全功能。

(2) L 系列: 是 Low Power 和 Low Cost 的代名词,它的特点是低功耗和低成本。第一款 L 系列的 Kinetis 型号是 KL25,采用 Cortex-M0+ 内核,设计主频最高为 48MHz,价格相对较低,可以算作 Kinetis 系列中低端低成本的一支。目前比较热门的型号是 KL16/26/27 等,相对于主要竞争对手 8051 系列 MCU,其最大的优势在于先进的处理器结构和较小的 MCU 封装技术。

(3) V 系列: 主要面向电机控制领域,具有增强的电机 PWM 控制模块以及较高的主频,可以广泛地应用在电机和马达控制领域、四轴飞行器领域等。目前比较热门的型号有 KV1x、KV3x、KV4x、KV5x 等。

(4) M 系列: 主要应用在电表及仪器测量领域。所有的 M 系列产品都包含一个模拟前端(AFE)和 4 个 24 位 $\Sigma\text{-}\delta$ ADC 以及两个低噪音可编程增益放大器(PGA)。该系列产品的应用市场是电表以及各种精密测量仪器。

(5) W 系列: 基于 ARM Cortex-M0+/M4 处理器内核,片上集成有 RF 射频收发器,因此 W 系列产品的应用市场是无限传感器网络、无线数据采集等领域。其中应用最多的 KW2x 系列,片上集成 IEEE 802.15 协议物理层与数据链路层硬件协议栈,可实现 ZigBee、蓝牙等无线射频功能。

(6) E 系列和 EA 系列: 也是基于 ARM-Cortex-M0+ 处理器内核,可在复杂的电气噪声环境和要求高可靠性的应用中保持稳定。E 系列几乎所有 I/O 都可以支持 5V 电平,并且具有丰富的内存、外设和产品可供选择。EA 系列则是 E 系列的汽车级版本,所有产品

全部经过汽车级产品认证,并且与 E 系列 100%兼容。

以上只是关于 Kinetis 家族的简略介绍,如果读者对选型感兴趣,可以查看恩智浦官方网站的 Kinetis 页面: <http://www.nxp.com/products/microcontrollers-and-processors/arm-processors/kinetis-cortex-m-mcus>; KINETIS 来获取更多信息。

1.2.2 Kinetis 系列芯片的命名规则及选型

Kinetis 芯片的型号也叫作 Part Number,用一串字符串表示,如 MK64FN1MVLQ12。Kinetis 的型号看起来复杂,实际上则是有一定规则的,通常所说的 K60、KL25、KV10 等,每个名字背后其实都代表一大类不同封装的产品。首先需要认识一下前几个字母及数字的意义:首字母是产品的家族归类,如 K、L、V 等;后面的第一位数字代表大类,第二个数字一般代表升级版本号,例如 KV10 的 1 代表大类,0 代表升级版本号。不同的大类在芯片外设集成上有着较大的区别,表 1-1 总结了产品大类中不同的数字所代表的不同意义。

表 1-1 Kinetis 产品分类

产品型号中的大类编号	意 义
0	入门型号,包含 MCU 开发必备的模块,如 UART I ² C SPI 定时器等,一般片内 Flash 和 RAM 都比较小,价格也是最低廉的
1	基础型号,包含一般 MCU 最常用的模块,如 UART I ² C SPI 定时器等
2	在基础型号的基础上增加 USB 模块
3	在基础型号的基础上增加 SLCD 模块
4	在基础型号的基础上增加 USB + SLCD 模块
5	在基础型号的基础上增加高精度 ADC 模块
6	在基础型号的基础上增加 USB 和以太网模块
7	在基础型号的基础上增加 LCD 驱动器、USB 和以太网模块
8	在基础型号的基础上增加一些面向特定领域的应用,例如加密、外扩存储器总线等

例如 KL26 和 K22 有一个共同的特性,就是都含有 USB、K40 和 KL46,都具有 USB 和段式液晶。

至于大类后面的数字,即升级版本号,则表示更复杂的含义。一般升级版本号数字越大,代表这类产品越新,比如 K64 就是 K60 的新一代替代产品,在工艺、性能、性价比上更有优势。通常情况下,越大的升级版本号数字还代表着这类产品的 Flash 和 RAM 越大。

关于兼容性:不同的 Kinetis 系列产品之间有着很好的兼容性,大部分程序都可以通用。对于操作外设的那部分程序,只要外设模块相同,则程序必然兼容。如果外设换掉了,则只需要修改对应的外设操作代码即可。对于上层的 App 应用程序,则基本不需要改动。另外需要注意的是,一些高级外设,例如 USB、以太网等,在 Kinetis 的基础型号(K1x)上是没有的,例如用户想移植一个跑在 K64 上的以太网应用程序到 K10 上,显然是不可能的。

下面结合图 1-2 所示的分段释义来说明 Kinetis 系列芯片完整的命名规则,如表 1-2 所示。

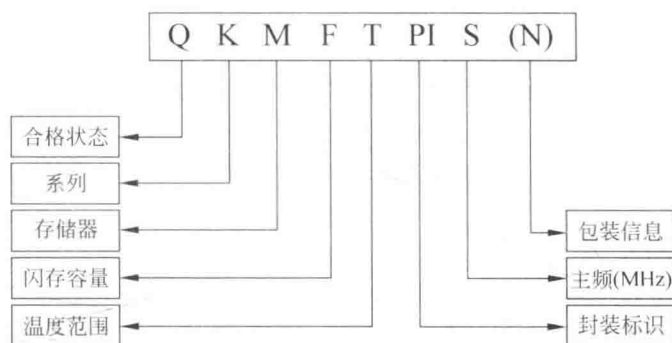


图 1-2 Kinetis 产品命名规则

表 1-2 Kinetis 型号包含字母代表的意义

域	描 述	取 值
Q	合格状态	M=正式量产版,P=工程样片
K	系列	K、L、V、W、M 等
M	是否带 FlexMemory 存储器	N=不带 Flex 存储器,X=带 Flex 存储器
F	存储容量	32 代表 32KB 64 代表 64KB 128 代表 128KB 512 代表 512KB 1M 代表 1MB 2M 代表 2MB
T	温度范围	-40~125℃
PI	封装	列举几个常用封装信息: FM 代表 32QFN (QFN 封装 32 引脚) LF 代表 48LQFP (LQFP 48 引脚) LH 代表 64LQFP LL 代表 100LQFP LQ 代表 144LQFP MJ 代表 256MAPBGA
S	最高主频	4 代表 48MHz 7 代表 72MHz 10 代表 100MHz 12 代表 120MHz 15 代表 150MHz 18 代表 180MHz

表 1-3 是几个常用产品型号的举例说明。

表 1-3 Kinetis 型号含义举例

型号(Part Number)	说 明
MK64FN1MVLQ12	MK64,带 FPU, 1MBFlash,不带 FlexMem, LQFP144,最高主频 120MHz
MK60DN512VLL10	MK60,不带 FPU, 512KBFlash,不带 FlexMem, LQFP100,最高主频 100MHz
MKL26Z128VLH4	MKL26,不带 FPU,128KBFlash, LQFP64,最高主频 48MHz

1.3 开发平台硬件电路分析

本节介绍本书所使用的硬件开发环境,可帮助读者快速上手开发板并且为后面的学习做铺垫。

另外需要说明的是,本书所使用的硬件资源其实只占开发板硬件资源的一部分,所以读者无须逐个分析开发板上所有的硬件电路模块,下面逐一分析本书中所有用到的硬件资源。

目前市场上的 Kinetis 开发平台还很有限,本书所使用的开发平台是超核电子推出的 K60/K64 渡鸦系列,该开发板价格适中,板载的 USB-UART 电路可以直接将 MCU 的串口信号转换成 USB 信号并打印在 PC 上,非常适合 MCU 开发中经常需要用到的打印调试信息操作等。另外,板载资源丰富,包含了 UART/CAN/I²C/SPI 等最常用的外设接口模块,纵观整个开发板,几乎所有 K64 的外设都可在开发板上做实验。开发板整体照片如图 1-3 所示。

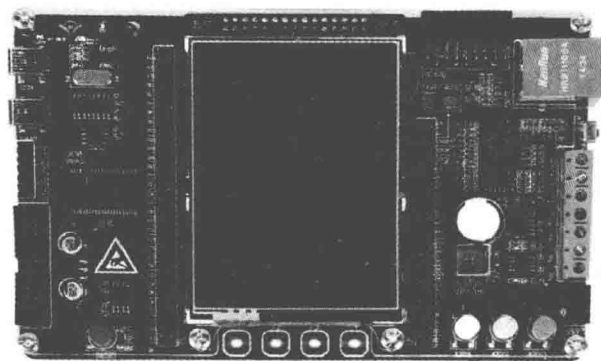


图 1-3 渡鸦开发板照片

下面将详细介绍开发板的硬件资源和原理图。

1.3.1 MCU

本书所选择的 MCU 为 MK64FN1MVLQ12,该芯片是 Kinetis K 系列产品,具备超大容量的 Flash 和 RAM,正是由于其强大的配置,该款芯片成为物联网网关产品中的理想选择。它集成 FPU 和 DSP 指令,拥有 256KB SRAM、1024KB Flash,并且最多可拥有 16 个 16 位定时器,可实现 16 路 PWM 波形或用来做编码器解码。该芯片还包含 4 个内部定时器模块、1 个 DMA 控制器(16 个通道)、3 个 SPI、1 个全双工 I²S、3 个 I²C、6 个串口、1 个 USB OTG 模块、1 个 CAN 总线、2 个 16 位 ADC 模块、2 个 12 位 DAC 模块、以太网、RTC、以及 112 个通用 I/O 口。关于 K64 芯片的引脚分配可以参考 MK64 的参考手册(RM)中的第 10 章: Signal Multiplexing and Signal Descriptions。

MK64 是 Kinetis K 系列产品,与 MK60 系列相比,最主要的区别有以下几点:

(1) MK64 为新一代产品,功耗和成本更加有优势;