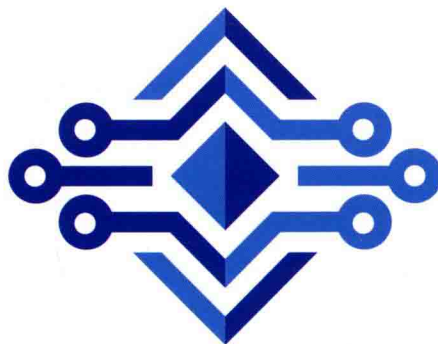
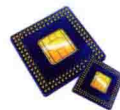




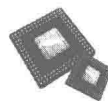
**DEVELOPMENT PRACTICE OF
MSP432 MCU BASED
ON ARM CORTEX-M4F**

**基于ARM Cortex-M4F内核的
MSP432 MCU开发实践**

叶国阳 刘铮 徐科军 编著



机械工业出版社
China Machine Press



电子电气工程师技术丛书

**DEVELOPMENT PRACTICE OF
MSP432 MCU BASED
ON ARM CORTEX-M4F**

**基于ARM Cortex-M4F内核的
MSP432 MCU开发实践**

叶国阳 刘争 徐科军 编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

基于 ARM Cortex-M4F 内核的 MSP432 MCU 开发实践 / 叶国阳, 刘铮, 徐科军编著.
—北京: 机械工业出版社, 2017.12
(电子电气工程师技术丛书)

ISBN 978-7-111-58779-8

I. 基… II. ①叶… ②刘… ③徐… III. 微控制器—研究 IV. TP332.3

中国版本图书馆 CIP 数据核字 (2017) 第 320123 号

本书以基于 ARM Cortex-M4F 内核的 MSP432P401r 微控制器为例, 全面介绍了 MSP432 微控制器的结构原理、外设模块、示例程序及应用。全书共分为 10 章。第 1 章概述 MSP432 微控制器; 第 2 章介绍 MSP432 微控制器软件工程开发基础; 第 3 章介绍 MSP432 微控制器 CPU 与存储器; 第 4 章介绍 MSP432 微控制器中断系统; 第 5 章介绍 MSP432 微控制器时钟系统与低功耗结构; 第 6 章介绍 MSP432 微控制器输入输出模块; 第 7 章介绍 MSP432 微控制器片内通信模块; 第 8 章介绍 MSP432 微控制器片内控制模块; 第 9 章讲述了 MSP432 微控制器应用设计实例——口袋实验套件; 第 10 章介绍基于 MSP432 微控制器的参考设计。

本书可以作为 MSP432 微控制器初学者快速入门教材, 也可作为从事自动控制、信号检测及仪器仪表等专业的科研工作者提供学习和参考。

基于 ARM Cortex-M4F 内核的 MSP432 MCU 开发实践

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 陈佳媛

责任校对: 殷 虹

印 刷: 三河市宏图印务有限公司

版 次: 2018 年 1 月第 1 版第 1 次印刷

开 本: 186mm×240mm 1/16

印 张: 20

书 号: ISBN 978-7-111-58779-8

定 价: 69.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

微控制器（俗称单片机）的应用日趋广泛，这对处理器的综合性能提出了更高的要求。美国德州仪器（TI）公司推出的 MSP432 微控制器是具有极低功耗的基于 ARM Cortex-M4F 内核的 32 位微控制器（MCU），其主频为 48MHz，有效功耗只有 95 μ A/MHz，实现了功耗与性能的完美兼得。MSP432 微控制器性能优良，将在过程控制、便携仪表、无线通信、能源收集、消费类电子产品和公共事业计量等方面得到广泛的应用。MSP432P401r 微控制器是 MSP432 系列中的第一款。本书以此微控制器为代表，全面介绍 MSP432 微控制器的原理及应用。全书共分 10 章，具体内容包括：

第 1 章介绍 MSP432 微控制器的发展历史、应用领域和技术特点。

第 2 章介绍 MSP432 微控制器软件工程的开发基础，主要讲解 MSP432 微控制器 C 语言编程基础、MSP432 微控制器的软件编程方法，以及软件开发集成环境 CCS 的基本操作。

第 3 章以 MSP432P401r 微控制器为例，简单介绍 MSP432 微控制器的结构和特性；同时，介绍 Cortex-M4 架构、内核及其主要功能，重点介绍 MSP432 微控制器的 CPU 和存储器。

第 4 章介绍中断的一些基本概念，讲解 MSP432 微控制器具有的中断源及中断处理过程，叙述 MSP432 微控制器的中断嵌套，并以两个例程介绍 MSP432 微控制器中断的应用。

第 5 章重点讲述 MSP432 微控制器的时钟系统及其低功耗结构。

第 6 章重点讲述各典型输入输出模块的结构、原理及功能，并给出各个模块的简单应用例程。

第 7 章详细讲述片内通信模块的结构、原理及功能，包括 URAT、SPI 和 I²C，并给出了简单的数据通信例程。

第 8 章重点介绍 Flash 控制器和 DMA 控制器的结构、原理及功能。

第 9 章介绍编者实验室自行研制的基于 MSP432P401r 微控制器的口袋实验套件。实验套件由 MSP432P401r LaunchPad（最小系统）和口袋实验板组成，可完成检测、综合和互动三大类实验。

第 10 章介绍基于 MSP432 微控制器的参考设计。

本书由徐科军统筹，由叶国阳、刘铮和徐科军编写。其中，叶国阳编写了前言、第1~8章和第10章，刘铮编写第9章，徐科军审阅了全书。美国德州仪器（TI）公司大学计划的王沁工程师和蒋荣慰工程师对于本书的编写给予了极大的支持，就本书框架的确定和目录的编写提出了许多宝贵的意见。在此，表示衷心的感谢。

本书所附的程序代码来源于TI官网或由编者编写。对于TI官网的程序，读者可登录TI官方网站进行下载，编者编写的程序可登录华章图书（www.hzbook.com）官网下载。

由于水平有限，书中可能存在不妥之处，敬请广大读者批评指正。

编者

2017年10月

前言

第1章 MSP432 微控制器概述 ... 1

- 1.1 MSP432 微控制器的诞生及应用 ... 1
- 1.2 MSP432 微控制器的特点 4
- 1.3 MSP432 微控制器应用选型 6
- 1.4 本章小结 6
- 1.5 思考题与习题 7

第2章 MSP432 微控制器软件

工程开发基础 8

- 2.1 MSP432 微控制器 C 语言基础 8
- 2.2 MSP432 微控制器软件工程
基础 23
- 2.3 MSP432 微控制器软件开发
集成环境 CCSv6.1 26
- 2.4 本章小结 41
- 2.5 思考题与习题 42

第3章 MSP432 微控制器 CPU
与存储器 43

- 3.1 MSP432P4xx 系列微控制器
结构概述 43

- 3.2 MSP432P401r 微控制器特性、
结构和外部引脚 46

- 3.3 MSP432P401r 微控制器 CPU
的寄存器资源 53

- 3.4 MSP432 微控制器的存储器 57

- 3.5 本章小结 59

- 3.6 思考题与习题 60

第4章 MSP432 微控制器中断
系统 61

- 4.1 中断的基本概念 61

- 4.2 可嵌套的向量中断控制器
NVIC 62

- 4.3 MSP432 微控制器中断源 62

- 4.4 中断响应过程 67

- 4.5 中断返回过程 67

- 4.6 中断应用 68

- 4.7 本章小结 69

- 4.8 思考题与习题 70

第5章 MSP432 微控制器时钟
系统与低功耗结构 71

- 5.1 时钟系统结构与原理 71

5.2 低功耗结构及应用	88	8.2 DMA 控制器	261
5.3 本章小结	96	8.3 本章小结	271
5.4 思考题与习题	96	8.4 思考题与习题	272
第6章 MSP432 微控制器输入 输出模块	97	第9章 MSP432 微控制器应用 设计实例——口袋实验 套件	273
6.1 通用 I/O 端口	97	9.1 口袋实验套件概述	274
6.2 模/数转换模块 ADC14	103	9.2 测量类实验	275
6.3 比较器 E	135	9.3 综合类实验	292
6.4 定时器	147	9.4 本章小结	303
6.5 本章小结	181	9.5 思考题与习题	303
6.6 思考题与习题	182		
第7章 MSP432 微控制器片内 通信模块	184	第10章 基于MSP432 微控制器 的参考设计.....	304
7.1 eUSCI 的异步模式—— UART	184	10.1 IWR1443 77GHz 级发射机的 功率优化参考设计	304
7.2 eUSCI 的同步模式	212	10.2 电容式触摸 MCU 和 LCD 的 参考设计	307
7.3 eUSCI 的 I ² C 模式	225	10.3 近场通信 (NFC) 读/写器 参考设计	309
7.4 本章小结	246	10.4 本章小结	311
7.5 思考题与习题	246	10.5 思考题与习题	311
第8章 MSP432 微控制器片内 控制模块	247	参考文献	312
8.1 Flash 存储器.....	247		

MSP432 微控制器概述

为了实现单片机更低的功耗和更优的性能，2015 年 3 月美国德州仪器公司（简称 TI 公司）在原有超低功耗 MSP430 单片机的基础上，推出了 32 位 ARM Cortex-M4F 的 MCU（微控制器单元，俗称单片机）。ARM Cortex-M4F 处理器是由 ARM 公司专门开发的最新嵌入式处理器，TI 公司将其作为 MSP432 单片机的内核，让用户可以更容易、更方便地设计 IoT（物联网）产品，提供面向电动机控制、汽车、电源管理、嵌入式音频和工业自动化市场的解决方案。为了让读者对 MSP432 单片机有一个初步的认识和了解，本章首先介绍 MSP430 单片机的发展历史和 MSP432 单片机的诞生过程，然后叙述 MSP432 单片机具有的特点及优势，最后简要讲解 MSP432 单片机的应用选型。

1.1 MSP432 微控制器的诞生及应用

1.1.1 MSP430 微控制器的发展和 MSP432 微控制器的诞生

TI 公司于 1996 年推出 MSP430 单片机。经过 20 年的发展，MSP430 单片机家族的成员不断壮大，特别是 MSP430F5xx、MSP430F6xx 系列产品，充分体现了 MSP430 的优点。在 MSP430 的基础上，TI 公司又推出了极低功耗的 MSP432 单片机。

从 MSP430 发展到 MSP432 诞生，经历了 4 个阶段。

(1) 开始阶段

这一阶段是从 1996 年 TI 公司推出 MSP430 单片机开始到 2000 年年初。在这个阶段，TI 公司首先推出了 MSP430 单片机中的 33x、32x 和 31x 等几个系列，而后于 2000 年年初又推出了 11x 和 11x1 系列。

MSP430 单片机的 33x、32x 和 31x 等系列具有 LCD（液晶显示器）驱动模块，有利于提高系统的集成度。每一系列有 ROM（只读存储器）型（C）、OTP（一次性可编程存储器）型（P）、EPROM（可擦除可编程只读存储器）型（E）等芯片。因为 EPROM 型的价格昂贵，运行环境温度范围窄，所以主要用于样机开发。这也反映了 TI 公司的开发模式：用 EPROM 型开发样机；用 OTP 型进行小批量生产；用 ROM 型进行大批量生产。2000 年 TI 公司推出的 11x/11x1 系列采用 20 引脚封装，虽然内存容量、片上功能和 I/O 引脚数比较少，但是价格比较低廉。

这个时期的 MSP430 单片机已经显露出其特低功耗等技术特点,但是,也有不尽如人意之处。例如:只有 32x 系列才有片内高精度 A/D 转换器;只有 33x 系列才具备片内串行通信接口、硬件乘法器、足够的 I/O 引脚等。33x 系列价格较高,比较适合用于较为复杂的应用系统。当用户在设计时,若需要更多地考虑成本,则 33x 并不一定是最适合的选择。

(2) 寻找突破,引入 Flash 技术

随着 Flash (闪存) 技术的迅速发展, TI 公司也将这一技术引入 MSP430 单片机中,于 2000 年 7 月推出 F13x/F14x 系列,2001 年 7 月~2002 年又相继推出 F41x、F43x、F44x 系列,这些全部是 Flash 型单片机。

F41x 系列单片机具有 48 个 I/O 口和 96 段 LCD 驱动。F43x 和 F44x 系列在 13x 和 14x 的基础上,增加了 LCD 驱动器,并将驱动 LCD 的段数由 3xx 系列的最多 120 段增加到 160 段,还相应地调整了显示存储器在存储区内的地址,为以后的发展拓展了空间。

MSP430 单片机由于具有 Flash 存储器,在系统设计、开发调试及实际应用上都表现出较为明显的优点。这时 TI 公司推出了具有 Flash 型存储器及 JTAG (联合测试行为组织,常指一种边界扫描技术) 的廉价开发工具 MSP-FET430x110,将国际上先进的 JTAG 技术和 Flash 在线编程技术引入 MSP430 单片机。这种以 Flash 技术与 FET (场效应晶体管) 开发工具组合的开发方式,具有方便、廉价和实用的特点,给用户提供了一个较为理想的样机开发方式。

另外,2001 年 TI 公司又公布了 BOOTSTRAP LOADER (BSL,引导装入程序) 技术,它可在烧断熔丝以后,通过口令更改并运行内部的程序,这为系统软件的升级提供了又一个方便的手段。BSL 具有很高的保密性,口令长度可达到 32 字节。

(3) 蓬勃发展阶段

TI 公司在 2003 年年底至 2004 年推出了 F15x 和 F16x 系列产品。这些产品大大增加了 RAM (随机存取存储器) 容量,如 F1611 的 RAM 容量增加到了 10kB,以便引入实时操作系统 (RTOS) 或简单文件系统等。同时,还增加了 I²C、DMA、DAC12 和 SVS 等外设模块。

TI 公司在 2004 年下半年推出了 MSP430x2xx 系列。该系列进一步精简了 MSP430x1xx 的片内外设,使得单片机具有小型、快速和灵活的特点,且价格低廉,可以用于开发超低功耗医疗、工业与消费类嵌入式系统。与 MSP430x1xx 系列相比,MSP430x2xx 的 CPU 时钟提高到 16MHz (MSP430x1xx 系列是 8Hz),待机电流从 2 μ A 降到 1 μ A,具有最小 14 引脚的封装产品。

2003 年以来, TI 公司针对热门的应用领域,利用 MSP430 的超低功耗特性,还推出了一系列专用单片机,如专门用于电能计量的 MSP430FE42x、用于水表计量的 MSP430FW42x 和用于医疗仪器的 MSP430FG4xx 等。

2007 年, TI 公司推出了具有 120kB 闪存、8kB RAM 的 MSP430FG461x 系列超低功耗单片机。该系列产品可满足设计大型应用系统时对内存的要求;同时,高集成度与超低功耗的特性,使它可以应用于便携式医疗设备和无线射频等嵌入式系统。

2008 年, TI 公司推出了具有革命性突破的超低功耗 MSP430F5xx 系列产品。该系列单片机能够针对主频高达 25MHz 的产品实现最低的功耗,并拥有更大的闪存与 RAM 存储器,以及诸如射频 (RF)、USB (通用串行总线)、加密和 LCD 接口等片上外设。与 1xx、2xx 以及 4xx 等前几代产品相比,F5xx 器件的处理性能提高了 50% 以上,闪存与 RAM 存储器容量也实现了双倍增长,从而使系统能以极小的功耗执行复杂度极高的任务。

2011 年年底, TI 公司推出了具有 LCD 控制器的 MSP430F6xx 系列产品。该系列产品支持高

达 25MHz 的 CPU 时钟，且能够提供不同的内存，如 256kB 闪存和 18kB RAM，可在电能计量和能源监测应用中为开发人员提供更大的发挥空间。

(4) MSP432 诞生

为了将低功耗 MCU 的理念与优势从 16 位领域延伸至 32 位领域，同时实现低功耗与高性能的完美结合，TI 公司于 2015 年 3 月推出了内核为 Cortex-M4F 的 MCU——MSP432 单片机。MSP432 中的 32 代表该 MCU 是 32 位的。相较于 16 位 RISC（精简指令集）的 MSP430 单片机，MSP432 单片机采用的是 32 位 RISC，性能有很大的提升。这一全新的 MCU 充分利用 TI 公司在超低功耗 MCU 方面的独特技术，在实现优化性能的同时降低了功率的损耗，使其有效功耗和待机功耗分别只有 95 μ A/MHz 和 850nA。MSP430 单片机与 MSP432 单片机的比较如表 1-1 所示。

表 1-1 MSP430 单片机与 MSP432 单片机的比较

MSP430	MSP432
16 位	32 位
1996 年面世	2015 年面世
至今 500 余款，且在不断增加	至今只有一款，后续将增加
超低功耗、丰富外设和模拟集成度方面的行业领导者	目前业内最低功耗的 ARM-Cortex-M4F MCU
适用于低功耗要求、长时间工作在睡眠模式的应用	适用于高效数据处理和增强的低功耗运行的至关重要的应用

MSP432 单片机的内核使用 32 位的 Cortex-M4F 内核，具有 32 位的数据总线、32 位的寄存器组和 32 位的存储器接口。Cortex 系列属于 ARMv7 架构，这是到 2010 年为止 ARM 公司最新的指令集架构。ARMv7 架构定义了三大分工明确的系列：“A”系列面向尖端的、基于虚拟内存的操作系统和用户应用；“R”系列针对实时系统；“M”系列针对微控制器。Cortex-M 系列是一系列针对成本敏感的应用程序进行优化的深层嵌入式处理器。Cortex-M4F 是由 ARM 专门开发的最新嵌入式处理器，在 M3 的基础上强化了运算能力，新加了浮点、DSP、并行计算等功能，用以满足需要有效且易于使用的控制和信号处理功能混合的数字信号控制市场。Cortex-M4F 中的 F 表示具有浮点运算功能（FPU，浮点运算单元）。Cortex-M4F 采用哈佛结构，这意味着它拥有独立的指令总线和数据总线，对指令和数据的访问可以同时进行，数据访问的过程不会影响或干扰指令的流水线，因此，提升了处理器的性能。此特性使得整个 Cortex-M4F 内核中有多个总线和接口，每个总线和接口均可同时使用，以实现最佳的利用率。数据总线和指令总线共享同一存储空间，此空间称为统一的存储系统。

1.1.2 MSP432 微控制器的应用领域

由于 MSP432 单片机与 MSP430 单片机有相同的片上外设，在实际应用中，MSP432 单片机将凭借其超低功耗和高性能特性，受到越来越多设计者的青睐，在 MSP430 的应用领域中得到广泛的使用。

作为一个极低功耗、高性能的 MCU，MSP432 单片机的应用前景十分广阔。MSP432 主要瞄准对于功耗和性能要求都比较严苛的应用领域，包括电动机控制、电源管理、嵌入式音频、楼宇控制、可穿戴设备和传感器检测等。一个典型的应用是智能手表，其组成框图如图 1-1 所示。与 Cortex-M0 + 内核的 MCU 相比，MSP432 具有巨大的性能优势，可迅速处理距离、心率和卡路里变化等数据；同时，允许电池的充电周期长达数天，甚至一周。在物联网设备和可穿戴设备

风靡的今天，相信 MSP432 单片机将大有用武之地。

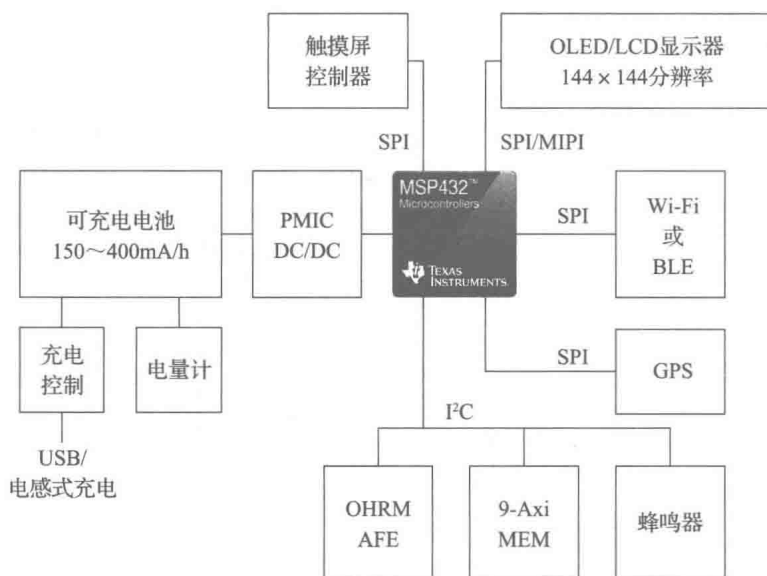


图 1-1 基于 MSP432 的智能手表框图

1.2 MSP432 微控制器的特点

MSP432 单片机凭借 32 位的 48MHz Cortex-M4F 内核及周围设备，实现低功耗与高性能的完美结合。此单片机可提供更高性能，它的性能是 M3 内核的两倍；极低功耗是 MSP432 单片机的另一个重要特点，在工作模式下功耗仅为 $95\mu\text{A}/\text{MHz}$ ，而待机功耗仅为 850nA ，其中包括了 RTC（实时时钟）的功耗。同时，用户能够充分利用 MSP430 的工具链（工具链是将用户写的代码转换为可执行程序，一般由编译器、连接器、解释器和调试器组成），以及 ARM 的工具链，以获得最佳的高性能和低功耗。现在，由于 MSP430 平台的延伸，MSP430 和 MSP432 之间的代码可以无缝移植，即用户可以将现有的基于 MSP430 平台的项目移植到 MSP432 平台。

MSP432 单片机具有以下主要特点。

（1）强大的处理能力

由于性能是设计 MSP432 单片机中的一个关键目标，因此，选择了性能最高的 Cortex M 内核。Cortex-M4F 内核包含对完整 ARM 指令集的访问权限，此外还包含了 DSP 扩展指令和一个浮点 FPU 模块，可以更高效地执行运算。MSP432 单片机的主频可以高达 48MHz，而 MSP430 单片机中最新的 MSP430F6xx 系列只有 25MHz。同时，该单片机内置了高性能的外设且独具特色，例如，把 MCU 驱动部分放到只读存储器里而不是闪存中运行，由于只读存储器中的驱动程序执行速度比闪存中高 200%，所以可以更快地运行程序，并且，可以把节省下来的 Flash 存储空间直接用于存储用户数据或者进行数据计算。另外，单片机内置的模拟模块是目前为止速度最快的 1MSPS 的 14 位 ADC，这可以让用户以更快的速度进行数据采样，MSP432 单片机中 14

位的 ADC 的采样速度比原来的 MSP430 单片机中 ADC 快了 5 倍。

(2) 极低功耗

MSP432 单片机在硅片的级别上就进行了低功耗优化。它加入了宽工作电压范围等功能，可在 1.62V 电压下工作，这包括全速代码运行以及闪存访问。MSP432 单片机还集成了 DC/DC 稳压器（将一个固定的直流电压变换为可变的直流电压），可以在频率超过 24MHz 时提高工作效率，与以前的 LDO（低压差线性稳压器）相比，可把整个功耗再降 40%。而闪存的缓冲器 NTMA 可以最大限度地缩短 CPU 的执行周期。MSP432 单片机包含一种独特的可选 RAM 保持特性，此特性能够为运行所需的 8 个 RAM 段中的每一段提供专用电源，由此每段的功耗可以减少 30nA，从而降低了总体系统功率。

在器件具备低功耗性能的同时，MSP432 单片机也提供帮助实现低功耗的工具和软件。位于 ROM 中的驱动程序库所需要的功耗也低于在闪存中运行驱动程序的功耗。而 TI 提供的 ULP Advisor 和 Energy Trace + 等工具可帮助用户优化代码，从而避免在不必要的情况下产生额外的功耗。

通过硅片级的低功耗优化和针对低功耗的软件优化，MSP432 单片机的有效功耗和待机功耗分别只有 95 μ A/MHz 和 850nA，如图 1-2 所示。



图 1-2 MSP432 单片机的低功耗优化

很多传感器输出的都是模拟信号，模拟信号只有通过 ADC 转化为数字信号后才能用单片机进行处理。MSP432 单片机中 14 位的 ADC 采样速度比 MSP430 中的 ADC 快了 5 倍。当使用 14 位 ADC 时，以 1MSPS 的速度运行采样传感器数据时，能耗仅有 375 μ A，非常省电。同时，ADC 的采样速度可调，最低功耗可以低至 200 μ A。

(3) 高性能模拟技术及丰富的片上外设

MSP432 单片机结合 TI 的高性能模拟技术，具有非常丰富的片上外设：时钟模块（UCS）、Flash 控制器、RAM 控制器、DMA（直接存储器存取）控制器、通用 I/O 口（GPIO）、CRC（循环冗余校验）模块、定时器（Timer）、实时时钟模块（RTC）、14 位模数转换器（ADC14）、12 位数模转换器（DAC12）、比较器（COMP）、UART（通用异步收发器）、SPI（串行外设接口）、I²C（内部串行总线）、USB 模块等。丰富的片上外设不仅给应

用系统设计带来了极大的方便，同时也降低了应用系统的成本。

(4) 高效灵活的开发环境

MSP432 单片机具有十分方便的开发调试环境，这是由于其内部集成了 JTAG 调试接口和 Flash 存储器，可以在线实现程序的下载和调试，开发人员只需一台计算机、一个 JTAG 调试器和一个软件开发集成环境即可完成系统的开发。目前针对 MSP432 单片机，推荐使用 CCSv6.1 软件开发集成环境。CCSv6.1 为 CCS 软件的最新版本，功能更强大，性能更稳定，可用性更高，是开发 MSP432 单片机软件的理想工具。

1.3 MSP432 微控制器应用选型

目前 TI 公司仅发布了一款 MSP432 单片机——MSP432P401，有 6 种不同的型号，如表 1-2 所示。带 R 的器件具有 256kB 闪存和 64kB SRAM，而带 M 的器件则有 128kB 闪存和 32kB SRAM。TI 提供 3 种不同的封装类型，可以根据具体应用来选择。最小的是 5 × 5mm 的 BGA（球栅阵列）封装，此外还有 64QFN（方形扁平无引脚）和 100LQFP（薄型方形扁平）封装。

表 1-2 MSP432 单片机部分选型表

单片机型号	Flash (kB)	SRAM (kB)	I/O	定时器	ADC14	eUSCI		比较器		封装
						ChA	ChB	E0	E1	
MSP432P401 RIPZ	256	64	84	5	24	4	4	8	8	100 PZ
MSP432P401 MIPZ	128	32	84	5	24	4	4	8	8	100 PZ
MSP432P401 RIZXH	256	64	64	5	16	3	4	6	8	80 ZXH
MSP432P401 MIZXH	128	32	64	5	16	3	4	6	8	80 ZXH
MSP432P401 RIRGC	256	64	48	5	12	3	3	2	4	64 RGC
MSP432P401 MIRGC	128	32	48	5	12	3	3	2	4	64 RGC

一般来说，在进行 MSP432 单片机选型时，可以考虑以下几个原则：

- 1) 选择内部功能模块最接近应用系统需求的型号；
- 2) 若应用系统开发任务重，且时间比较紧迫，可以首先考虑比较熟悉的型号；
- 3) 所选型号的 Flash 和 RAM 空间是否能够满足应用系统设计的要求；
- 4) 在满足系统设计的前提下，尽量选用价格最低 MSP432 单片机型号。

1.4 本章小结

TI 公司最新推出的 MSP432 是具有极低功耗的 32 位 ARM Cortex-M4F 的微控制器，兼具低功耗与高性能的特点。MSP432 单片机具有强大的处理能力、极低的功耗、高性能模拟技术，以及丰富的片上外设、高效灵活的开发环境等特点，应用前景十分广阔，包括电源管理、楼宇控制、可穿戴设备等。目前推出的 MSP432P401 有 6 种不同的型号，可根据系统设计需要选择

合适的型号。通过本章的学习，读者对 MSP432 单片机具有初步的了解和认识，可为后面章节的学习打下良好的基础。

1.5 思考题与习题

1. 简述 MSP430 单片机的发展和 MSP432 单片机的诞生历史。
2. MSP432 单片机具有哪些特点？可以应用到哪些场合？
3. 为什么 MSP432 单片机具有极低功耗的特性？
4. 相比 MSP430 单片机，MSP432 单片机在性能上具有哪些优点？
5. 列举 MSP432 单片机所具有的片内外设，简述它们的功能。
6. 了解 MSP432 单片机的命名规则。

MSP432 微控制器软件工程开发基础

MSP432 单片机的内核是 ARM Cortex-M4F，使用了 32 位 RISC（精简指令集计算机）处理器。RISC 的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。RISC 是为高级语言所设计的，因为它在很大程度上降低了编译器的设计难度，有利于产生高效紧凑的代码。初学者完全可以在不深入了解汇编指令系统的情况下，直接开始 C 语言的学习。本章介绍 MSP432 单片机软件工程的开发基础，主要讲解 MSP432 单片机 C 语言编程基础、MSP432 单片机的软件编程方法以及软件集成开发环境的基本操作，使读者对 MSP432 单片机的编程思想有一定的了解。

2.1 MSP432 微控制器 C 语言基础

程序设计语言的发展经历了从机器语言、汇编语言到高级语言的过程。C 语言是一门高级语言，其特征为：语句简洁紧凑、运算符灵活、数据类型丰富、控制语句结构化和可移植性好。使用 C 语言进行程序设计是当前单片机系统开发和应用的必然趋势，这主要有两个方面的因素：一方面，随着芯片行业的快速发展，单片机能够以较低的成本提供较快的运算速度和更大的存储空间，所以，在单片机系统开发过程中，单片机的计算能力和存储空间已经不是考虑的主要因素；另一方面，现在单片机系统处理的任务越来越复杂，产品更新的周期也越来越短，这对开发的进度提出了更高的要求。

现在使用汇编语言进行程序设计已经不能满足要求，并且目前 MSP432 单片机的 C 编译器的性能非常优秀，因此，初学者完全可以在不深入了解汇编指令系统的情况下，直接学习 C 语言的编程。本书的代码均是用 C 语言编写的。

MSP432 单片机使用的 C 语言集成开发环境（CCS）是由 TI 公司提供的。为了叙述方便，以下将 MSP432 的 C 语言简称为 C432。为了让读者更好地理解以后章节中的例程，本节重点介绍 MSP432 单片机的 C 语言基础。C432 语法与标准 C 基本一致，但是也有一些很重要的差异。为此，本节在介绍标准 C 语法的同时，穿插介绍其与 C432 的不同之处。

2.1.1 标识符与关键字

1. 标识符

标识符用来标识程序中某个对象的名字，这些对象可以是语句、数据类型、函数、变量、常量、数组等。标识符的第一个字符必须是字母或下划线，随后的字符必须是字母、数字或下划线。例如：_Data、Hao123、test1 是正确形式，而 2016Y、#123、.COM 是错误形式。标识符长度是由机器的编译系统决定的，一般限制为 8 字符（注：8 字符长度限制是 C89 标准，C99 标准已经扩充长度，其实大部分工业标准的长度都更长）。

C 语言区分大小写字符，所以，在编写程序时，要注意大小写字符的区别。例如：对于 book 和 BOOK 这两个标识符来说，C 语言会认为这是两个完全不同的标识符。

注意：C432 中，标识符的命名应该简洁明了和含义清晰，这样便于程序的阅读和维护。例如，在比较最大值时，最好使用 max 来定义该标识符；在片内模块初始化函数部分，函数命名后面尽量加上_init，如 ADC14_init()表示 ADC14 模块初始化函数。

2. 关键字

关键字又称为保留字，是 C 语言中预定义的符号。它们有固定的含义，用户定义的标识符不得与它们冲突。C 语言中的关键字是由小写字母构成的字符序列，标准 C 提供了 32 个关键字，如表 2-1 所示。

表 2-1 C 语言中的 32 个关键字

关键字	语义	关键字	语义	关键字	语义	关键字	语义
auto	自动	double	双精度	int	整数	struct	结构
break	中断	else	否则	long	长整型	switch	开关
char	字符	enum	枚举	register	寄存器	typedef	类型定义
case	情况	extern	外部	return	返回	union	共用
const	常量	float	浮点	short	短整型	unsigned	无符号
continue	继续	for	对于	signed	带符号	void	空
default	默认	goto	转向	sizeof	字节数	volatile	可变的
do	做	if	如果	static	静态	while	当

32 个关键字可以分为 4 类：

(1) 数据类型

int、char、float、double、short、long、void、signed、unsigned、enum、struct、union、const、typedef、volatile

(2) 存储类别

auto、static、register、extern

(3) 语句命令字

break、case、continue、default、do、else、for、goto、if、return、switch、while

(4) 运算符

sizeof

除关键字外，还有一些准关键字，它们也有固定的含义，主要作为库函数名和预处理命

令。C 语言允许这些符号另做他用，使这些符号失去原本含义。但是，为了避免出现不必要的麻烦，建议不将这些准关键字另做他用。

编译预处理的准关键字在使用时前应加“#”，常用的有：define、elif、else、endif、error、ifdef、ifndef、include、line、pragma、undef。2.1.8 节将对这些准关键字做详细的说明。系统标准库函数的准关键字有：scanf、printf、putchar、getchar、strcpy、strcmp 和 sqrt 等。

2.1.2 变量

在程序中，其值可以改变的量称为变量，它是内存单元的符号地址。变量有两个基本要素：一是变量名，其命名规则要符合标识符的规定；二是变量类型，其类型决定了变量在内存中占据的存储单元的长度。C432 中变量类型及值域如表 2-2 所示。在 C 语言中，变量必须先定义，后使用。程序运行过程中，变量的值存储在内存中。从变量中取值，实际上是根据变量名找到相应的内存地址，从该存储单元中读取数据。在定义变量时，变量的类型必须与其存储的数据类型相匹配，以保证程序中变量能够正确地使用。当指定了变量的数据类型时，系统将为它分配若干字节的内存空间。

表 2-2 C432 中变量类型

变量类型	所占字节数	值 域
char	1	-128 ~ 127
unsigned char		0 ~ 255
int	2	-32768 ~ 32767
unsigned int		0 ~ 65535
long	4	$-2^{31} \sim 2^{31} - 1$
unsigned long		$0 \sim 2^{32} - 1$
long long	8	$-2^{63} \sim 2^{63} - 1$
unsigned long long		$0 \sim 2^{64} - 1$
float	4	$-3.40282e^{38} \sim 3.40282e^{38}$
double	8	$-1.79769e^{308} \sim 1.79769e^{308}$

在定义变量表达式中增加某些关键字，可以给变量赋予某些特殊性质。

1) **const**：定义常量。在 C432 语言中，const 关键字定义的常量实际上放在 Flash 中，可以用 const 关键字定义常量数组。

2) **static**：相当于本地全局变量，只能在函数内使用，以避免全局变量混乱。

3) **volatile**：定义“易失性”变量。编译器将认定该变量的值会随时改变，对该变量的任何操作都不会被优化过程删除。比如，在实际编程的过程中发现，利用变量 i 递减或递加产生的软件延时函数，会被编译器优化而不会执行。因此，若读者遇到这种情况且希望延时函数工作，只需在变量 i 前加关键字 volatile 即可。

2.1.3 C 语言运算符

C 语言内部运算符很丰富，用运算符可以将常量、变量、函数连接成 C 语言表达式。因此，