ARM Cortex M4 嵌入式系统开发实践

-基于飞思卡尔 K60 系列微控制器

王宜怀 王林 编著

内容简介

本书以飞思卡尔半导体公司（原摩托罗拉半导体部）的 32 位 K60 系列微控制器中 MK60N512VMD100 为蓝本阐述嵌入式系统的软件与硬件设计。全书共 17 章，其中第 1 章为概述，阐述嵌入式系统的知识体系、学习误区、学习建议及基于硬件构件的嵌入式系统开发方法。第 2 章给出 MK60N512VMD100 硬件最小系统。第 3 章给出第一个样例程序及 CodeWarrior、IAR 工程组织，完成第一个 MK60N512VMD100 工程的入门。第 4 章阐述串行通信接口 UART，并给出第一个带中断的实例。1-4 章完成了学习一个新 MCU 完整要素（知识点）的入门。6-16 章分别给出 GPIO 的应用（键盘、LED 及 LCD）、定时器、A/D 转换、SPI、I2C、I2S、Flash、CAN、USB、SDHC、TSI、以太网及 MK60N512VMD100 其他模块等。第 17 章讲述了嵌入式操作系统有关的知识。

本书提供了所有实例源程序、辅助资料、相关芯片资料及常用软件工具。

本书可供大学有关专业的高年级学生和研究生用作教材或参考读物，也可供嵌入式系统开发与研究人员用作参考和进修资料。

前言

嵌入式计算机系统简称为嵌入式系统，其概念最初源于传统测控系统对计算机的需求。随着以微处理器（MPU）为内核的微控制器（MCU）制造技术的不断进步，计算机领域在通用计算机系统与嵌入式计算机系统这两大分支分别得以发展。通用计算机已经在科学计算、事物管理、通信、日常生活等各个领域产生重要的影响。在后 PC 时代，嵌入式系统的广阔应用将是计算机发展的重要特征。一般来说，嵌入式系统的应用范围可以粗略分为两大类：一类是电子系统的智能化（如工业控制、现代农业、家用电器、汽车电子、测控系统、数据采集、传感网应用等）；另一类是计算机应用的延伸（如手机、电子图书、通信、网络、计算机外围设备等）。不论如何分类，嵌入式系统的技术基础是不变的，即要完成一个以 MCU 为核心的嵌入式系统应用产品设计，需要有硬件、软件及行业领域相关知识。但是，随着嵌入式系统中软件规模日益增大，对嵌入式底层驱动软件的封装提出了更高的要求，可复用性与可移植性受到特别的关注，嵌入式软硬件构件化开发方法逐步被业界所重视。

本书基本思想

本书以嵌入式硬件构件与底层软件构件设计为主线，按照嵌入式软件工程的要求，以飞思卡尔半导体公司（原摩托罗拉半导体部）的 32 位 K60 系列中 MK60N512VMD100 微控制器为蓝本阐述嵌入式系统的软件与硬件设计。并阐述嵌入式操作系统相关知识。

我从事单片机与嵌入式系统科研与教学工作是从 1991 年开始的。1991-1999 年间，使用 MCS-51 系列 MCU。2000 年至现在，一直使用飞思卡尔（2004 年以前是摩托若拉半导体部）的 MCU。十多年来，陆续以飞思卡尔的 HC08/S08（8 位）、S12/S12X（16 位）、ColdFire（32 位）、M*Core（32 位，该内核转给中国后称为 C*Core）进行科研开发与教学工作，并以这些MCU为蓝本先后写了一些嵌入式应用技术入门方面的书，得到了大多数读者的肯定，深受感动。2010-2011 年，苏州大学嵌入式团队的工作重点是进行 ARM Cortex-M4 核 Kinetis 系列 MCU（K60）、新型 Zigbee 芯片 MC1323x、DSC 芯片 MC56F825x 等方面的工作，这些工作成果也将会逐步与读者分享。在写书方面，多年来一直在探索如何能够使读者不误入歧途，如何能够快速入门，如何能够规范编程，如何能够由浅入深、循序渐进，如何能够使读者打好嵌入式硬件与软件基础。为此从以下几点把握写作：（1）把与芯片无关的通用知识分离出来，从涉及底层编程角度对基本原理进行简明扼要的阐述，分别放入相应章节的前面或网上光盘中。这些知识主要包括通用 I/O、串行通信、键盘编码原理、LED 扫描原理、SPI、PWM、USB、I^2C、CAN、A/D、D/A、嵌入式以太网等。并在各书中基本保持不变。这一点是接受了飞思卡尔全球大学计划负责人 Andy Mastronardi 先生的建议，经过几年不断修改完善，可把通用部分斟酌得更好一些。也使得 8 位、16 位、32 位的书风格保持一致。新的芯片出来后，书的修改只要更新与芯片的相关部分。（2）硬件相关的部分，采用了硬件构件思想，制定了一些基本规范，对底层驱动进行构件化封装，提高了可复用性与可移植性。使程序结构更加清晰，初学者可以“先使用、后理解”。（3）不论是 8 位、16 位、32 位，也不论是哪个芯片，从编程角度，把与硬件相关的共性和与硬件无关的共性分别抽象出来，力求做到，硬件相关部分风格一致，硬件无关部分程序一致。这样便于融会贯通，不再纠结芯片位数、操作系统等问题。

关于飞思卡尔微控制器

飞思卡尔半导体是全球最大半导体公司之一，在微控制器领域长期居全球市场领先地位，以高可靠性获得业界的一致赞誉。该公司的微控制器产品系列齐全，由不同位数（如 8 位、16 位、32 位等）、不同封装形式（如 DIP、SOIC、QFP、LQFP、BGA 等）、不同温度范围（0～70℃、-40～85℃、-40～105℃、-40～125℃等）、所含模块不同等构成了庞大的产品系列。飞思卡尔的 S08（8 位）、S12/S12X（16 位）、ColdFire（32 位）、ARM Cortex（32 位）等系列 MCU，广泛地应用于汽车电子、消费电子、工业控制、网络和无线市场等嵌入式系统各个领域。系列齐全的微控制器产品，为嵌入式系统各种应用提供了选择与解决方案，使得用户可以各取所需。不论是电子系统智能化还是计算机应用延伸的嵌入式应用设计，无论需要怎样的系统功能和集成度，总能从飞思卡尔庞大产品系列中选取一款合适的芯片进行应用开发。这正是嵌入式系统产品设计者所期望的，也节省了嵌入式学习者的时间，可以加快开发进度，提高开发质量。

本书特点

2009 年，我撰写了《基于 32 位 ColdFire 构建嵌入式系统》一书，2010 年又撰写了《嵌入式技术基础与实践（第 2 版）》。两书中系统阐述和应用了嵌入式构件开发思想，本书秉承这些工作，按照“通用知识—芯片编程结构概要—基本编程方法—底层驱动构件封装—应用方法与举例”的线条，逐步阐述电子系统智能化嵌入式应用的软件与硬件设计。

（1）把握通用知识与芯片相关知识之间的平衡。书中对于嵌入式“通用知识”的基本原理，以应用为立足点，进行语言简洁、逻辑清晰的阐述，同时注意与芯片相关知识之间的衔接，使读者在更好地理解基本原理的基础上，理解芯片应用的设计，同时反过来，加深对通用知识的理解。

（2）把握硬件与软件的关系。嵌入式系统是软件与硬件的综合体，嵌入式系统设计是一个软件、硬件协同设计的工程，不能像通用计算机那样，软件、硬件完全分开来看。特别是对电子系统智能化嵌入式应用来说，没有对硬件的理解就不可能写好嵌入式软件，同样没有对软件的理解也不可能设计好嵌入式硬件。因此，本书注重把握硬件知识与软件知识之间的关系。

（3）对底层驱动进行构件化封装。书中对每个模块均给出根据嵌入式软件工程基本原则并按照构件化封装要求编制底层驱动程序，同时给出详细、规范的注释及对外接口，为实际应用提供底层构件，方便移植与复用，可以为读者进行实际项目开发节省大量时间。

（4）设计合理的测试用例。书中所有源程序均经测试通过，并保留测试用例在本书的随书光盘中，避免了因例程的书写或固有错误给读者带来烦恼。这些测试用例，也为读者验证与理解带来方便。

（5）随书光盘提供了所有模块完整的底层驱动构件化封装程序、文档与测试用例，同时随书光盘中还包含芯片参考手册、写入器安装与使用方法、工具软件（如开发环境、程序写入与读出软件、串口调试工具等）、有关硬件原理图及其他技术资料。

（6）提供硬件评估版、写入调试器，并给出单独进行程序写入与读出的软件工具，方便读者进行实践与应用。

本书主要内容

本书以飞思卡尔半导体的 32 位 K60 系列微控制器中 MK60N512VMD100 为蓝本阐述嵌入式系统的软件与硬件设计。全书共 17 章，其中第 1 章为概述，阐述嵌入式系统的知识体系、学习误区、学习建议及基于硬件构件的嵌入式系统开发方法。第 2 章给出 MK60N512VMD100 硬件最小系统。第 3 章给出第一个样例程序及 CodeWarrior、IAR 工程

组织，完成第一个 MK60N512VMD100 工程的入门。第 4 章阐述串行通信接口 UART，并给出第一个带中断的实例。1-4 章完成了学习一个新 MCU 完整要素（知识点）的入门。6-16 章分别给出 GPIO 的应用（键盘、LED 及 LCD）、定时器、A/D 转换、SPI、I2C、I2S、Flash、CAN、USB、SDHC、TSI、以太网及 MK60N512VMD100 其他模块等。第 17 章讲述了嵌入式操作系统有关的知识。

致谢

本书除封面署名作者外，还有苏州大学计算机科学与技术学院嵌入式应用方向研究生姚丹丹、李翠霞、朱乐乐、冯上栋、石晶、苏勇等协助书稿整理及程序调试工作，他们卓有成效的工作，使本书更加实用。飞思卡尔半导体有限公司的 Andy Mastronardi 先生、马莉女士一直关心支持苏州大学飞思卡尔嵌入式系统研发中心的建设，为本书的撰写提供了硬件及软件资料，并提出了许多宝贵建议。飞思卡尔半导体有限公司的许多技术人员提供了技术支持。北京航空航天大学出版社的董立娟编辑为本书的出版付出了大量细致的工作。在此一并表示诚挚的谢意。

鉴于作者水平有限，书中难免存在不足和错误之处，恳望读者提出宝贵意见和建议，以便再版时改进。

王宜怀

2011 年 4 月于苏州大学

第1章 概述

作为全书导引，本章主要知识点有：①简要给出嵌入式系统定义、由来及特点；②讨论嵌入式开发中的硬件、软件的可复用与可移植性问题；③提出嵌入式硬件构件的基本思想，阐述基于硬件构件的嵌入式系统开发方法；④给出基于硬件构件的嵌入式系统硬件电路设计方法与嵌入式底层软件构件的编程方法。

1.1 嵌入式系统定义、由来及特点

1.1.1 嵌入式系统的定义

嵌入式系统（Embedded system）有多种多样的定义，但本质是相同的。本书关于嵌入式系统的定义取自美国 CMP Books 出版的 Jack Ganssle 和 Michael Barr 著作《Embedded System Dictionary》[1]。

嵌入式系统的定义：一种计算机硬件和软件的组合，也许还有机械装置，用于实现一个特定功能。在某些特定情况下，嵌入式系统是一个大系统或产品的一部分。世界上第一个嵌入式系统是 1971 年 Busicom 公司用 Intel 单芯片 4004 微处理器完成的商用计算器系列。该词典还给出了嵌入式系统的一些示例：微波炉、手持电话、计算器、数字手表、录像机、巡航导弹、GPS 接收机、数码相机、传真机、跑步机、遥控器和谷物分析仪等，难以尽数。通过与通用计算机的对比可以更形象地理解嵌入式系统的定义。**该词典给出的通用计算机定义是：计算机硬件和软件的组合，用作通用计算平台。**PC、MAC 和 Unix 工作站是最流行的现代计算机。

我国《国家标准 GB/T 5271 信息技术词汇－嵌入式系统与单片机》部分，给出的嵌入式系统定义是：置入应用对象内部起操作控制作用的专用计算机系统。

国内对嵌入式系统定义曾进行过广泛讨论，有许多不同说法。其中嵌入式系统定义的涵盖面问题是主要争论焦点之一。例如，有的学者认为不能把手持电话叫嵌入式系统，而只能把其中起控制作用的部分叫嵌入式系统，而手持电话可以称为嵌入式系统的应用产品。其实，这些并不妨碍人们对嵌入式系统的理解，所以不必对定义感到困惑。有些国内学者特别指出，在理解嵌入式系统定义时，不要把嵌入式系统与嵌入式系统产品相混淆。实际上，从口语或书面语言角度，不区分“嵌入式系统”与“嵌入式系统产品”，只要不妨碍对嵌入式系统的理解就没有关系。

为了更清楚阐述嵌入式系统特点，首先介绍大多数嵌入式系统的核心部件－MCU（微控制器）的基本概念。

1.1.2 嵌入式系统的由来及其与微控制器的关系

1．MCU（微控制器）的基本含义

MCU 是单片微型计算机（单片机）的简称，早期的英文名是 Single-chip Microcomputer，后来大多数称之为微控制器（Microcontroller）或嵌入式计算机（Embedded computer）。现

[1] 中译本：Jack Ganssle 等著，马广云等译，《英汉双解嵌入式系统词典》，北京航空航天大学出版社，2006 年。

在 Microcontroller 已经是计算机中一个常用术语，但在 1990 年代之前，大部分英文词典并没有这个词。我国学者一般使用中文“单片机”一词，而缩写使用“MCU”[2]。所以本书后面的简写一律以 MCU 为准。**MCU 的基本含义是：在一块芯片上集成了中央处理单元（CPU）、存储器（RAM/ROM 等）、定时器/计数器及多种输入输出（I/O）接口的比较完整的数字处理系统**。图 1-1 给出了典型的 MCU 组成框图。

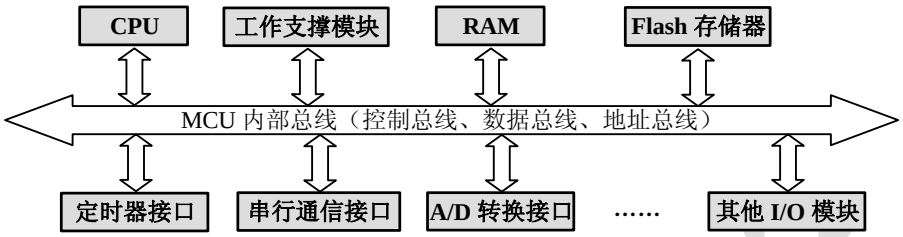

图 1-1 一个典型的 MCU 内部框图

MCU 是在计算机制造技术发展到一定阶段的背景下出现的，它使计算机技术从科学计算领域进入到智能化控制领域。从此，计算机技术在两个重要领域——通用计算机领域和嵌入式（Embedded）计算机领域都获得了极其重要的发展，为计算机的应用开辟了更广阔的空间。

就 MCU 组成而言，虽然它只是一块芯片，但包含了计算机的基本组成单元，仍由运算器、控制器、存储器、输入设备、输出设备五部分组成，只不过这些都集成在一块芯片上，这种结构使得 MCU 成为具有独特功能的计算机。

2．嵌入式系统的由来

通俗地说，计算机是因科学家需要一个高速的计算工具而产生的。直到二十世纪七十年代，电子计算机在数字计算、逻辑推理及信息处理等方面表现出非凡的能力。在通信、测控与数据传输等领域，人们对计算机技术给予了更大的期待。这些领域的应用与单纯的高速计算要求不同，主要表现在：直接面向控制对象；嵌入到具体的应用体中，而非计算机的面貌出现；能在现场连续可靠地运行；体积小，应用灵活；突出控制功能，特别是对外部信息的捕捉与丰富的输入输出功能等。由此可以看出，满足这些要求的计算机与满足高速数值计算的计算机是不同的。因此，**一种称之为 MCU 或微控制器的技术得以产生并发展**。为了区分这两种计算机类型，通常把满足海量高速数值计算的计算机称为**通用计算机系统**，而把**嵌入到实际应用系统中，实现嵌入式应用的计算机称之为嵌入式计算机系统，简称嵌入式系统**。

3．嵌入式系统与 MCU 的关系

何立民先生说：“有些人搞了十多年的 MCU 应用，不知道 MCU 就是一个最典型的嵌入式系统”[3]。实际上，MCU 是在通用 CPU 基础上发展起来的，MCU 具有体积小、价格低、稳定可靠等优点，它的出现和迅猛发展，是控制系统领域的一场技术革命。MCU 以其较高的性能价格比、灵活性等特点，在现代控制系统中具有十分重要的地位。**大部分嵌入式系统以 MCU 为核心进行设计**。MCU 从体系结构到指令系统都是按照嵌入式系统的应用特点专门设计的，它能很好地满足应用系统的嵌入、面向测控对象、现场可靠运行等方面的要求。因此**以 MCU 为核心的系统是应用最广的嵌入式系统**。在实际应用时，开发者可以根据具体要求与应用场合，选用最佳型号的 MCU 嵌入到实际应用系统中。

在 MCU 出现之前，人们必须用模拟电路、数字电路实现大部分计算与控制功能，这样

[2] MCU 的英文全称是 Microcontroller Unit。

[3] 详见《单片机与嵌入式系统应用》，2004 年第 1 期。

使得控制系统体积庞大，易出故障。MCU 出现以后，情况发生了变化，系统中的大部分计算与控制功能由 MCU 的软件实现。其它电子线路成为 MCU 的外围接口电路，承担着输入、输出与执行动作等功能，而计算、比较与判断等原来必须用电路实现的功能，可以用软件取代，大大地提高了系统的性能与稳定性，这种控制技术称之为嵌入式控制技术。在嵌入式控制技术中，核心是 MCU，其它部分依此而展开。

1.1.3 嵌入式系统的特点

要谈嵌入式系统特点，不同学者也许有不同说法。这里从与通用计算机对比的角度谈嵌入式系统的特点。

1．嵌入式系统属于计算机系统，但不单独以通用计算机的面目出现

嵌入式系统的本名叫嵌入式计算机系统（Embedded computer system），它不仅具有通用计算机的主要特点，又具有自身特点。嵌入式系统也必须要有软件才能运行，但其隐含在种类众多的具体产品中。同时，通用计算机种类屈指可数，而嵌入式系统不仅芯片种类繁多，而且由于应用对象大小各异，嵌入式系统作为控制核心，已经融入到各个行业的产品之中。

2．嵌入式系统开发需要专用工具和特殊方法

嵌入式系统不像通用计算机那样有了计算机系统就可以进行应用开发。一般情况下，MCU 芯片本身不具备开发功能，必须要有一套与相应芯片配套的开发工具和开发环境。这些工具和环境一般基于通用计算机上的软硬件设备以及各种逻辑分析仪、混合信号示波器等。开发时往往有主机和目标机的概念，主机用于程序的开发，目标机作为程序的执行机，开发时需要交替结合进行。

3．使用 MCU 设计嵌入式系统，数据与程序空间采用不同存储介质

在通用计算机系统中，程序存储在硬盘上。实际运行时，通过操作系统将要运行的程序从硬盘调入内存（RAM），运行中的程序、常数、变量均在 RAM 中。而以 MCU 为核心的嵌入式系统，其程序被固化到非易失性存储器中[4]。变量及堆栈使用 RAM 存储器。

4．开发嵌入式系统涉及软件、硬件及应用领域的知识

嵌入式系统与硬件紧密相关，嵌入式系统的开发需要硬件、软件协同设计、协同测试。同时，由于嵌入式系统专用性很强，通常是用在特定应用领域，如嵌入在手机、冰箱、空调、各种机械设备、智能仪器仪表中起核心控制作用，功能专用。因此，进行嵌入式系统的开发，还需要对领域知识有一定的理解。当然，一个团队协作开发一个嵌入式产品，其中各个成员可以扮演不同角色，但对系统的整体理解与把握并相互协作，有助于一个稳定可靠嵌入式产品的诞生。

[4] 目前，非易失性存储器通常为 Flash 存储器，特点见有关“Flash 存储器在线编程”章节。

5．嵌入式系统的其他特点

除了以上特点之外，嵌入式系统还具有其他方面的特点。

在资源方面：嵌入式系统通常专用于某一特定应用领域，其硬件资源不会像通用计算机那样丰富；**在可靠性方面**：嵌入式系统一般要求更高可靠性和稳定性；**在实时性方面**：相当多嵌入式系统有实时性要求；**在成本方面**：嵌入式系统通常极其关注成本；**在功耗要求方面**：一些嵌入式系统要求低功耗；**在生命周期方面**：嵌入式系统通常比通用计算机系统生命周期长，升级换代比通用计算机慢。**在知识综合方面**：嵌入式系统是将先进的计算机技术、半导体技术及电子技术与各个行业的具体应用相结合的产物，是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。它的构成既有硬件又有软件，不仅包括应用软件，也可能包括系统软件。它既有数字电路又有模拟电路。其产品技术含量高，涉及多种学科，不容易开发，因此也不容易形成技术垄断。

这些特点决定了嵌入式系统的开发方法、开发难度、开发手段等，均不同于通用计算机，也不同于常规的电子产品。

1.2 嵌入式系统开发所遇到的若干问题

自从1974年第一款微处理器芯片问世以来，嵌入式系统应用已深入到军事、航空航天、通信、家电等各个领域。近年来，随着微控制器（MCU）内部Flash存储器可靠性提高及擦写方式的变化，内部RAM及Flash存储器容量的增大，以及外部模块内置化程度的提高，设计复杂性、设计规模及开发手段已经发生了根本变化。

在嵌入式系统发展的最初阶段，嵌入式系统开发（包括硬件和软件设计）通常是由一个工程师来承担，软件在整个工作中的比例很小。随着时间的推移，硬件设计变得越来越复杂，软件的份量也急剧增长，嵌入式开发人员也由一人发展为由若干人组成的开发团队。

目前，嵌入式系统开发主要存在以下两大问题：

问题 1：硬件设计缺乏重用支持

导致硬件设计缺乏重用支持的主要原因是：目前缺少可供硬件设计工程师们共同遵守的设计规范。设计人员往往是凭借个人工作经验和习惯的积累进行系统硬件电路的设计。在开发完一个嵌入式应用系统再进行下一个应用开发时，硬件电路原理图往往需要从零开始，重新绘制；或者在一个类似的原理图上修改，但往往又很麻烦，容易出错。

问题 2：驱动程序可移植性差

驱动程序的开发在嵌入式系统的开发中具有举足轻重的地位。驱动程序的好坏直接关系着整个嵌入式系统的稳定性和可靠性。然而，开发出完备、稳定的驱动程序并非易事。长期以来，开发人员在编写驱动程序时缺少软件工程思想的支撑，软、硬件设计过程孤立，造成与硬件密切相关的底层软件缺乏通用性，可移植性和可复用性较差，开发过程中缺少标准化、文档化的管理，给开发人员之间的交流以及日后系统的维护带来很大的困难。

上述两个问题导致的结果是系统开发周期长，效率低。下面提出的基于硬件构件的软硬件设计思想可在一定程度上解决这些问题。

1.3 嵌入式硬件构件的基本思想与应用方法

什么是嵌入式硬件构件？它与我们常说的硬件模块有什么不同？

众所周知，嵌入式硬件是任何嵌入式产品不可分割的重要组成部分，是整个嵌入式系统

的构建基础，嵌入式应用程序和操作系统都运行在特定的硬件体系上。一个以 MCU 为核心的嵌入式系统通常包括以下硬件模块：电源、写入器接口电路、硬件支撑电路、UART、USB、Flash、A/D、D/A、LCD、键盘、传感器输入电路、通信电路、信号放大电路、驱动电路等模块。其中有些模块集成在 MCU 内部，有的位于 MCU 之外。

与硬件模块的概念不同，**嵌入式硬件构件是指将一个或多个硬件功能模块、支撑电路及其功能描述封装成一个可重用的硬件实体，并提供一系列规范的输入/输出接口**。由定义可知，传统概念中的硬件模块是硬件构件的组成部分，一个硬件构件可能包含一个硬件功能模块，也有可能包含多个。

根据接口之间的生产消费关系，接口可分为提供接口和需求接口两类。根据所拥有接口类型的不同，硬件构件分为核心构件、中间构件和终端构件三种类型。核心构件只有提供接口，没有需求接口。也就是说，它只为其它硬件构件提供服务，而不接受服务。在以单 MCU 为核心的嵌入式系统中，MCU 的最小系统就是典型的核心构件。中间构件既有需求接口又有提供接口，即它不仅能够接受其它构件提供的服务，而且也能够为其它构件提供服务。而终端构件只有需求接口，它只接受其它构件提供的服务。这三种类型构件的区别如表 4-1 所示。

表 1-1 核心构件、中间构件和终端构件的区别

类型	需求接口	提供接口	举例
核心构件	无	有	芯片的硬件最小系统
中间构件	有	有	电源控制构件、232电平转换构件
终端构件	有	无	LCD构件、LED构件、键盘构件

利用硬件构件进行嵌入式系统硬件设计之前，应该进行硬件构件的合理划分，按照一定规则，设计与系统目标功能无关的构件个体，然后进行“组装”，完成具体系统的硬件设计。这样，这些构件个体也可以被组装到其他嵌入式系统中。在硬件构件被应用到具体系统中后，设计人员需要做的仅仅是为需求接口添加接口网标。

1.4 基于硬件构件的嵌入式系统硬件电路设计

1.4.1 设计时需要考虑的基本问题

设计以 MCU 为核心的嵌入式系统硬件电路需要根据需求分析进行综合考虑，需要考虑的问题较多，这里给出几个特别要注意的问题。

1．MCU 的选择

选择 MCU 时要考虑 MCU 所能够完成的功能、MCU 的价格、功耗、供电电压、I/O 口电平、管脚数目以及 MCU 的封装等因素。MCU 的功耗可以从其电气性能参数中查到。供电电压有 5V、3.3V 以及 1.8V 超低电压供电模式。为了能合理分配 MCU 的 I/O 资源，在 MCU 选型时可绘制一张引脚分配表，供以后的设计使用。

2．电源

（1）考虑系统对电源的需求，例如系统需要几种电源，如 24V、12V、5V 或者 3.3V 等，估计各需要多少功率或最大电流（mA）。在计算电源总功率时要考虑一定的余量，可按公式“电源总功率=2×器件总功率”来计算。

（2）考虑芯片与器件对电源波动性的需求。一般允许电源波动幅度在±5%以内。对于 A/D 转换芯片的参考电压一般要求±1%以内。

（3）考虑工作电源是使用电源模块还是使用外接电源。

3．普通 I/O 口

（1）上拉、下拉电阻：考虑用内部或者外部上/下拉电阻，内部上/下拉阻值一般在 700Ω左右，低功耗模式不宜使用。外部上/下拉电阻根据需要可选 10KΩ～1MΩ之间。

（2）开关量输入：一定要保证高低电压分明。理想情况下高电平就是电源电压，低电平就是地的电平。如果外部电路无法正确区分高低电平，但高低仍有较大压差，可考虑用 A/D 采集的方式设计处理。对分压方式中的采样点，要考虑分压电阻的选择，使该点通过采样端口的电流不小于采样最小输入电流，否则无法进行采样。

（3）开关量输出：基本原则是保证输出高电平接近电源电压，低电平接近地电平。I/O 口的吸纳电流一般大于放出电流。对小功率元器件控制最好是采用低电平控制的方式。一般情况下，若负载要求小于 10mA，则可用芯片引脚直接控制；电流在 10~100mA 时可用三极管控制，在 100mA～1A 时用 IC 控制；更大的电流则适合用继电器控制，同时建议使用光电隔离芯片。

4．A/D 电路与 D/A 电路

（1）A/D 电路：要清楚前端采样基本原理，对电阻型、电流型和电压型传感器采用不同的采集电路。如果采集的信号微弱，还要考虑如何进行信号放大。

（2）D/A 电路：考虑 MCU 的引脚通过何种输出电路控制实际对象。

5．控制电路

对外控制电路要注意设计的冗余与反测，要有合适的信号隔离措施等。在评估设计的布板时，一定要在构件的输入输出端引出检测孔，以方便排查错误时测量。

6. 考虑低功耗

低功耗设计并不仅仅是为了省电，更多的好处在于降低了电源模块及散热系统的成本。由于电流的减小也减少了电磁辐射和热噪声的干扰。随着设备温度的降低，器件寿命则相应延长，要做到低功耗一般需要注意以下几点：

（1）并不是所有的总线信号都要上拉。上下拉电阻也有功耗问题需要考虑。上下拉电阻拉一个单纯的输入信号，电流也就几十微安以下。但拉一个被驱动了的信号，其电流将达毫安级。所以需要考虑上下拉电阻对系统总功耗的影响。

（2）不用的 I/O 口不要悬空，如果悬空的话，受外界的一点点干扰就可能成为反复振荡的输入信号，而 MOS 器件的功耗基本取决于门电路的翻转次数。

（3）对一些外围小芯片的功耗也需要考虑。对于内部不太复杂的芯片功耗是很难确定的，它主要由引脚上的电流确定。例如有的芯片引脚在没有负载时，耗电大概不到 1 毫安，但负载增大以后，可能功耗很大。

7.考虑低成本

（1）正确选择电阻值与电容值。比如一个上拉电阻，可以使用 4.5K-5.3K 的电阻，你觉得就选个整数 5K，事实上市场上不存在 5K 的阻值，最接近的是 4.99K（精度 1%），其次是 5.1K（精度 5%），其成本分别比精度为 20%的 4.7K 高 4 倍和 2 倍。20%精度的电阻阻值

只有 1、1.5、2.2、3.3、4.7、6.8 几个类别（含 10 的整数倍）；类似地，20%精度的电容也只有以上几种值，如果选了其它的值就必须使用更高的精度，成本就翻了几倍，却不能带来任何好处。

（2）指示灯的选择。面板上的指示灯选什么颜色呢？有些人按颜色选，比如自己喜欢蓝色就选蓝色。但是其它红绿黄橙等颜色的不管大小（5mm 以下）封装如何，都已成熟了几十年，价格一般都在 5 毛钱以下，而蓝色却是近三四年才发明的，技术成熟度和供货稳定度都较差，价格却要贵四五倍。

（3）不要什么都选最好的。在一个高速系统中并不是每一部分都工作在高速状态，而器件速度每提高一个等级，价格差不多要翻倍，另外还给信号完整性问题带来极大的负面影响。

1.4.2 硬件构件化电路原理图绘制的简明规则

1．硬件构件设计的通用规则

在设计硬件构件的电路原理图时，需遵循以下基本原则：

（1）元器件命名格式：对于核心构件，其元器件直接编号命名，同种类型的元件命名时冠以相同的字母前缀。如电阻名称为 R1、R2 等，电容名称为 C1、C2 等，电感名称为 L1、L2 等，指示灯名称为 E1、E2 等，二极管名称为 D1、D2 等，三极管名称为 Q1、Q2 等，开关名称为 K1、K2 等。对于中间构件和终端构件，其元器件命名格式采用“构件名-标志字符?”。例如，LCD 构件中所有的电阻名称统一为“LCD-R?”，电容名称统一为“LCD-C?”。当构件原理图应用到具体系统中时，可借助原理图编辑软件为其自动编号。

（2）为硬件构件添加详细的文字描述，包括中文名称、英文名称、功能描述、接口描述、注意事项等，以增强原理图的可读性。中英文名称应简洁明了。

（3）将前两步产生的内容封装在一个虚线框内，组成硬件构件的内部实体。

（4）为该硬件构件添加与其它构件交互的输入/输出接口标识。接口标识有两种：接口注释和接口网标。它们的区别是：接口注释标于虚线框以内，是为构件接口所作的解释性文字，目的是帮助设计人员在使用该构件时理解该接口的含义和功能；而接口网标位于虚线框之外，且具有电气特性。为使原理图阅读者便于区分，接口注释采用斜体字。

在进行核心构件、中间构件和终端构件的设计时，除了要遵循上述的通用规则外，还要兼顾各自的接口特性、地位和作用。

2．核心构件设计规则

设计核心构件时，需考虑的问题是：“核心构件能为其他构件提供哪些信号？”。核心构件其实就是某型号 MCU 的最小系统。核心构件设计的目标是：凡是使用该 MCU 进行硬件系统设计时，核心构件可以直接“组装”到系统中，无须任何改动。为了实现这一目标，在设计核心构件的实体时必须考虑细致、周全，包括稳定性、扩展性等，封装要完整。核心构件的接口都是为其它构件提供服务的，因此接口标识均为接口网标。在进行接口设计时，需将所有可能使用到的引脚都标注上接口网标(不要考虑：核心构件将会用到怎样的系统中去)。若同一引脚具有不同功能，则接口网标依据第一功能选项命名。遵循上述规则设计核心构件的好处是：当使用核心构件和其它构件一起组装系统时，只要考虑其它构件将要连接到核心构件的哪个接口（不是考虑：核心构件将要连接到其它构件的哪个接口），这也符合设计人员的思维习惯。

3．中间构件设计规则

设计中间构件时，需考虑的问题是：“中间构件需要接受哪些信号，以及提供哪些信号？”。中间构件是核心构件与终端构件之间通信的桥梁。在进行中间构件的实体封装时，实体的涉及范围应从构件功能和编程接口两方面考虑。一个中间构件应具有明确的且相对独立的功能，它既要有接受其它构件提供的服务的接口，即需求接口，又要有为其他构件提供服务的接口，即提供接口。**描述需求接口采用接口注释，描述提供接口采用接口网标**。当中间构件被作为一个“零件”组装到具体系统中时，设计人员只要考虑为构件提供服务的来源，为接口注释添加对应的应用网标即可，其它内容无须关心或改动。

中间构件的接口数目没有核心构件那样丰富。为直观起见，设计中间构件时，将构件的需求接口放置在构件实体的左侧，提供接口放置在右侧。接口网标的命名规则是：构件名称-引脚信号/功能名称。而接口注释名称前的构件名称可有可无，它的命名隐含了相应的引脚功能。

电源控制构件（如图 1-2 所示）、可变频率产生构件（如图 1-3 所示）是常用的中间构件。图 1-2 中的 *Power-IN* 和图 1-3 中的 *SDI*、*SCK* 和 *SEN* 均为接口注释，Power-OUT 和 LTC6903-OUT 为接口网标。

4．终端构件设计规则

设计终端构件时，需考虑的问题是：“终端构件需要什么信号才能工作？”。终端构件是嵌入式系统中最常见的构件。终端构件没有提供接口，它仅有与上一级构件交互的需求接口，因而接口标识均为斜体标注的接口注释。LCD（YM1602C）构件（如图 1-4 所示）、LED 构件、指示灯构件以及键盘构件（如图 1-5）等都是典型的终端构件。

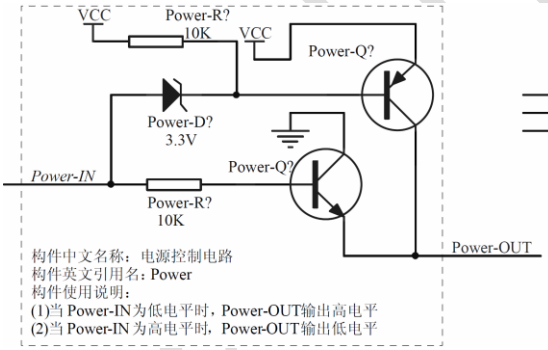

图 1-2 电源控制构件

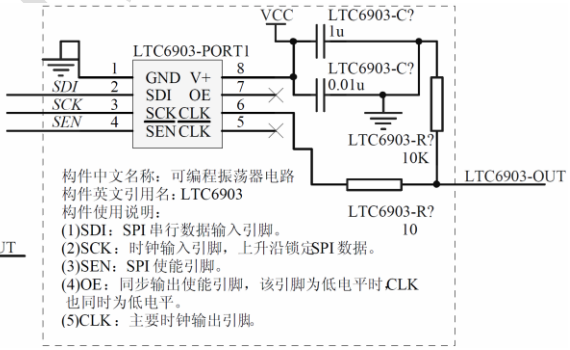

图 1-3 可变频率产生构件

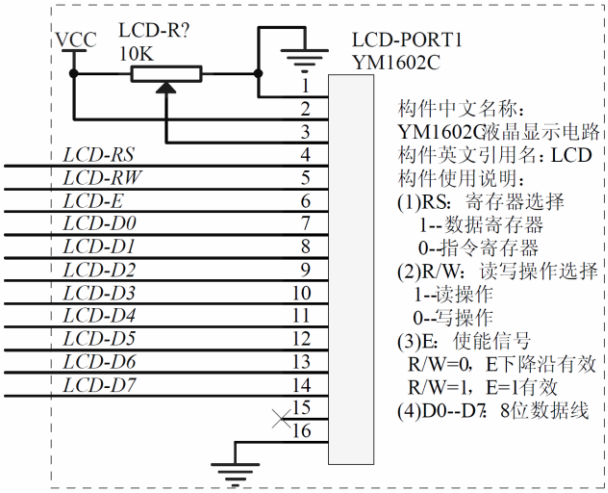

图 1-4 LCD 构件

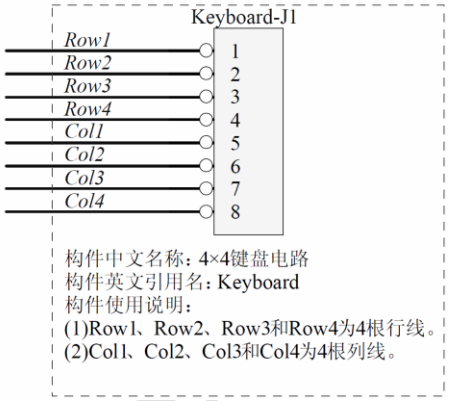

图 1-5 键盘构件

5．使用硬件构件组装系统的方法

对于核心构件，在应用到具体的系统中时，不必作任何改动。具有相同 MCU 的应用系统，其核心构件完全相同。对于中间构件和终端构件，在应用到具体的系统中时，仅需为需求接口添加接口网标，在不同的系统中，接口网标名称不同，但构件实体内部完全相同。

使用硬件构件化思想设计嵌入式硬件系统的过程与步骤是：

（1）根据系统的功能划分出若干个硬件构件。

（2）将所有硬件构件原理图“组装”在一起。

（3）为中间构件和终端构件添加接口网标。

1.4.3 实验 PCB 板设计的简明规则

印刷电路板（Printed-Circuit Board，PCB）是由环氧树脂粘合而成的镀了铜的玻璃纤维板。蚀刻掉部分镀铜，只留下以构成电路互连通路的铜层走线。PCB 可以是单层、双层或者四层、六层、八层、十二层甚至更多。层数越多，连接的布线就越容易，但成本却越高，调试也越难。如果 PCB 有专用供电层和接地层，系统将会有更强的燥声抗扰度。

在使用电子设计自动化软件（如 DXP）设计 PCB 时需注意以下几个方面的问题。

1．PCB 板布局

在正式走线之前要对 PCB 的大体格局进行规划。布局规划应遵循下列基本原则。

（1）在 PCB 布板之前首先要打印出相应的原理图，然后根据原理图确定整个 PCB 板的大体布局，即各个硬件构件的位置安排。

（2）PCB 板的形状如无其他要求，一般为矩形，长宽比为 4:3 或 3:2。

（3）考虑面板上元件的放置要求。

（4）考虑边缘接口。

2．元件放置

（1）元件放置要求整齐，尽可能正放，属于同一硬件构件内的元件尽可能排放在一起。排列方位尽可能与原理图一致，布线方向最好与电路图走线方向一致。元器件在 PCB 上的

排列可采用不规则、规则和网格等三种排列方式中的一种，也可同时采用多种。

不规则排列：元件轴线方向彼此不一致，这对印制导线布设是方便的，且平面利用率高，分布参数小，特别对高频电路有利。

规则排列：元器件轴线方向排列一致，布局美观整齐，但走线较长且复杂，适于低频电路。

网格排列：网格排列中的每一个安装孔均设计在正方形网格交点上。

布局的元器件应有利于发热元器件散热，高频时要考虑元器件之间的分布参数，高、低压之间要隔离，隔离距离与承受的耐压有关。

（2）电容的位置要特别注意，其中电源模块的滤波电容要求靠近电源，而 IC 的滤波电容要靠近 IC 的引脚。

（3）考虑元件间的距离，防止元件之间出现重叠。还要考虑元器件的引脚间距，元器件不同，其引脚间距也不相同，在 PCB 设计中必须弄清楚元器件的引脚间距，因为它决定着焊盘放置间距。对于非标准器件的引脚间距的确定最直接的方法就是：使用游标卡尺进行测量。

（4）PCB 四周留有 5-10mm 空隙不布器件。

（5）先放置占用面积较大的元器件，先集成后分立，先主后次，多块集成电路时先放置主电路。

（6）可调元件应放置在便于调节的地方，质量超过 15g 的元器件应当用支架，热敏元件应远离发热元件。晶体应平放，而不要竖直放置。

（7）PLL 滤波电路应尽量靠近 MCU。

3．有关设定

在正式布线之前，需要对软件环境的以下参数进行设置：

（1）线宽。导线的最小宽度由导线与绝缘基板间的粘附强度和流过它们的电流值决定。走线时导线尽可能宽，最好不要小于 0.5mm，手工制板应不小于 0.8mm，这样既可以减小阻抗，又可以防止由于制造工艺的原因导致导线断路。电源线和接地线因电流量较大，宽度要大于普通信号线，一般不要小于 1mm。三者关系为：地线宽度>电源线宽度>信号线宽度。

（2）间距。导线间的间距由它们之间的安全工作电压决定，相邻导线之间的峰值电压、基板的质量、表面涂层、电容耦合参数等都影响导线的安全工作电压，为满足电气安全要求，导线间距离以及导线与元件间距离要尽可能地大，一般不要小于 1mm，这样可以有效解决焊接时短路的问题。

（3）过孔大小。过孔大小设定要适中。

4．布线

布线时，应该首先对时钟和高速信号进行布线，以确保它们的走线尽可能直接。石英晶振和对噪声特别敏感的器件下面不要走线。对总线进行布线时，尽可能地保持信号线平行走线，而时钟信号线则要避免平行，且在这些导线之间最好加接地线。布板完成后一定要进行自动与人工检查。过孔数尽可能少。最小系统中未使用的 I/O 口，可通过电阻接地。走线尽量少拐弯，线宽不要突变，导线拐角应≥90°，力求线条简单明了。信号线的拐角应设计成钝角走向，为圆形或圆弧形，切忌画成 90°或更小角度形状。

5．测量点

考虑到硬件测试的方便，在 PCB 布板时要留下一些测量点，以便调试之用。测量点要

根据原理图确定。以下几处需要留测量点：

（1）原理图中模块的输入输出引脚。

（2）最小系统模块中 MCU 的引脚。

（3）各硬件功能模块单元的输入、输出口；

画测量点的步骤是：引出、打孔、标字。孔的大小以万用表头方便测量为原则，一般不要在线上直接打孔。

6．模块标示

由于在整体布局时，已经将各个硬件构件的组成元件放在一起，因而可在 PCB 板上用矩形框将各个硬件构件区分开，并用汉字标出构件名（与原理图一致），并注意字体字号。

7．铺地

在布板的最后都要铺地，目的是减小干扰，提高 PCB 板的稳定性。铺地需注意：

（1）在铺地前，要设定地与导线、地与引脚之间的距离，并要求该距离尽可能大。

（2）铺地本应该双面铺，作为实验用板，为了方便检查，可只铺反面地。

（3）如果电路板中有数字地和模拟地，应将它们隔离开，两者间使用磁珠相连。

8．空余位置的利用并标注相关信息

PCB 板的空余位置可适度作如下用途：

（1）电源、地。空白处多留几排电源和地。

（2）双排孔。留出几排两孔相连的排孔，以用来扩展或试验时焊接其他元件。

（3）固定孔。在 PCB 上画固定板的固定孔，一般在板的四个角落。

在完成 PCB 板的铺地之后，要在板的正面适当位置标出以下信息：单位、日期、责任人、PCB 板的名称、编号等。

9．抗干扰问题的特别考虑

PCB 设计中除了考虑以上问题外，还要考虑一些隐藏的问题，这些问题设计时不起眼，但是解决的时候，却非常麻烦，这就是电路的干扰问题，为此，在 PCB 设计时还应解决如下问题：

1）热干扰及抑制

元器件在工作中都有一定程度的发热，尤其是功率较大的器件所发出的热量会对周边比较敏感的器件产生干扰，若热干扰得不到很好的抑制，整个电路的电性就会发生变化，最后造成短路．为了对热干扰进行抑制，可采取以下措施：

（1）发热元件的放置

不要贴板放置，也可以单独设计为一个功能单元，放在靠近边缘容易散热的地方。另外，发热量大的器件与小热量的器件应分开放置。

（2）大功率器件的放置

应尽量靠近边缘布置，在垂直方向时应尽量布置在板上方。

（3）温度敏感器件的放置

对温度比较敏感的器件应安置在温度最低的区域，千万不要将它放在发热器件的正上方。

（4）器件的排列与气流

非特定要求，一般设备内部均以空气自由对流进行散热，故元器件应以纵式排列；若强制散热，元器件可横式排列。另外，为了改善散热效果，可添加与电路原理无关的零部件以引导热量对流。

2）共阻抗及抑制

共阻干扰是由 PCB 上大量的地线造成，当两个或两个以上的回路共用一段地线时，不同的回路电流在共用地线上产生一定压降，此压降经放大就会影响电路性能，当电流频率很高时，会产生很大的感抗而使电路受到干扰。为了抑制共阻抗，可采取以下措施：

（1）一点接地

使同级单元电路的几个接地点尽量集中，适用于信号的工作频率小于 1MHZ 的低频电路，如果工作频率在 1 一 1OMHz 而采用一点接地时，其地线长度应不超过波长的 1／20。

（2）就近多点接地

PCB 上有大量公共地线分布在板的边缘，且呈现半封闭回路(防磁场干扰)，各级电路采取就近接地，以防地线太长。适用于信号的工作频率大于 lOMHz 的高频电路。

（3）大面积接地

在高频电路中将 PCB 上所有不用面积均布设为地线，以减少地线中的感抗，从而削弱在地线上产生的高频信号，并对电场干扰起到屏蔽作用。

（4）加粗接地线

若接地线很细，接地电位则随电流的变化而变化，致使电子设备的定时信号电平不稳，抗噪声性能变坏，其宽度至少应大于 3mm。

（5）D／A(数／模)电路的地线分开

两种电路的地线各自独立，然后分别与电源端地线相连，以抑制它们相互干扰。

3）电磁干扰及抑制

电磁干扰是由电磁效应而造成的干扰，由于 PCB 上的元器件及布线越来越密集，如果设计不当就会产生电磁干扰。针对由电源布线、信号布线产生的电磁干扰，可采取不同的措施。

（1）电源布线引起的电磁干扰

电源布线可采用以下预防措施：

布线要宽。

加去耦电容。这种电容起到旁路滤波的作用。要在电源的输入端并联较大的和较小的滤波电容。

地线环绕。地线要靠近供电电源母线和信号线，因电流沿路径传输会产生回路电感，地线靠近，回路面积减小，电感量减小，回路阻抗减小，从而减小电磁干扰耦合。

（2）信号布线引起的电磁干扰

信号布线可采用以下预防措施：

不同功能的单元电路(如数字电路与模拟电路，高频与低频)分开设置，布线图形应易于信号流通且使信号流向尽可能保持一致。

合理使用屏蔽和滤波技术，注意高低压之间的隔离。

尽量不选用比实际需要的速度更快的元件，在元件的位置安排上，易受电磁干扰的元器件不能相距太近，应大于信号波长的四分之一，输入器件与输出器件尽量远离。

做到安全接地。