

第四章 单片机系统设计与开发工具

在学习和掌握如何应用单片机来设计和开发嵌入式系统时,除了首先要对所使用的单片机有全面和深入的了解外,配备和使用一套好的开发环境和开发平台也是必不可少的。在嵌入式系统的设计开发中,选用了好的开发工具和开发平台,往往能加速嵌入式应用系统的研制开发、调试、生产和维修,起到事半功倍的效果。

国内外许多公司根据不同单片机的性能和特点,研制推出了各种类型的用于开发单片机嵌入式系统的单片机开发装置和软件开发平台。不同类型的单片机使用的开发系统是不同的。对同一类型的单片机来讲,也有多种类型和功能的开发装置和开发平台。价格便宜、性能适中的系统在几百元,高性能的开发系统则要数千元到上万元,甚至仅仅一套软件开发平台就要上万元。虽然设计开发一个嵌入式系统,可以选用多家公司、多种类型的单片机,但在决定学习和使用哪种单片机时,应对单片机的性能价格,开发装置和开发平台的性能价格,以及是否方便使用等,几方面做一个综合的评估。

由于 AVR 单片机的程序存储器采用的是可多次下载的 Flash 存储器,具有可在线下载 (ISP) 等的优良特性,给学习和使用都带来极大的方便。

本章将在介绍单片机嵌入式系统设计开发基础知识后,重点讲述本书推荐和使用的一套采用 C 高级语言的软件开发平台—CodeVisionAVR (简称 CVAVR),和一套简易并且开放的,集下载编程、实验开发一体的实验板。

4.1 单片机嵌入式应用系统设计

4.1.1 单片机嵌入式系统开发所需的基础知识和技能

在 IT 行业,应用系统设计可以分成两大类,一类用于科学计算、数据处理、企业管理、Internet 网站建立等;另一类用于工业过程检测控制、智能仪表仪器和自动化设备、小型电子系统、通信设备、家用电器等。

对于前一类的应用系统设计,通常都是基于通用计算机系统和网络的系统开发,硬件设备也是通用的,可以从市场购买,而其主要的工作是软件开发,使用的开发平台以 C++、VB、数据库系统、网站建立开发平台等。

而后一类应用系统的设计则同前一类有很大的不同。它涉及的应用系统是一个专用的系统,往往要从零开始。既必须根据实际的需求,从系统硬件的构成设计与实现,到相应的软件设计与实现,两者并重,相辅相成,缺一不可。

第二类应用系统的特点是:

- 系统功能、要求、性能的多样性和专用性。
- 硬件电路和软件设计的不可分割和专一性。
- 可靠性高,抗干扰能力强。
- 体积小、重量轻、功耗省、投资少。
- 开发周期短,见效快。

单片机嵌入式应用系统设计归属于第二类应用系统的范畴。因此,对于从事单片机嵌入式系统设计、开发、学习的电子工程师和专业人员来讲,不仅要熟悉各种电子器件和 IC 芯片的使用和特性,具备模拟电路、数字电路等各类硬件电路和硬件系统的设计能力,还必须具有很强的计算机综合应用和软件编程设计能力。

在今天,单片机嵌入式系统的硬件设计、软件编程、系统仿真调试和程序的编程下载,

大都是在个人电脑 PC 的支撑下实现的。因此，单片机嵌入式系统设计开发人员所具备的另一个基本重要的技能就是要熟练掌握和使用个人电脑 PC，应具备熟练使用个人电脑 PC 的基础，具备相应的软件设计编程能力，熟悉相关软件(如 Protel、VHDL)的使用，同时对 PC 机的硬件接口 (RS-232 串行通信口、LPT 并行打印机接口、USB 接口等)也要有一定的了解。

早期的单片机系统开发平台是以 PC 的 DOS 操作系统为支撑的，但随着 PC 机的发展，现在的单片机系统开发平台都已经转到以 Windows 平台支撑的 PC 上。Windows 平台具有的多任务、多窗口性能给单片机嵌入式系统的设计开发带来极大的方便。

当你设计研制的单片机嵌入式系统是一个大型管理控制系统的下位机，或要与 Internet 或局网中的数据库联网，那么你除了要熟练掌握与单片机有关的硬件(模拟电路、数字电路、单片机等)和软件开发技术外，你还要具备与整个大的系统有关的基础和技术(如，数据库、Internet 协议、VB、VC 等)。因此，对一个高级电子工程师来讲，他对个人电脑 PC 的熟练掌握程度，以及软件设计和编程的能力，决不亚于计算机专业的人员，在某些方面比计算机专业的人员要求还高，还要全面。

要具备较高的硬件系统设计开发能力和水平，不是在短期内通过理论和书本的学习就能得以实现的。它需要经过一定时间的学习，并且特别注重理论与实际相结合，要亲自独立的动手去做，去实践，才能打下良好的基础。所以说，不亲自动手实践，你是不可能真正掌握设计开发单片机嵌入式系统技术的。有了良好的基础，有了长期的实践经验，加上紧跟世界半导体器件的最新发展，你才能成为一个真正的电子工程师

4.1.2 单片机嵌入式系统开发过程

对于单片机嵌入式系统的设计与开发来讲，由于涉及对象和要求的多样性和专用性，其硬件和软件结构有很大差异，但系统设计开发的基本内容和主要步骤是基本相同的。图 4-1 是单片机嵌入式系统开发过程示意图。

在一个具体的单片机嵌入式系统的设计时，一般需要作以下几个方面的考虑：

1. 确定系统设计的任务

在进行系统设计之前，首先必须进行设计方案的调研，包括查找资料、进行调查、分析研究。要充分了解对系统的技术要求、使用的环境状况以及使用人员的技术水平。明确任务，确定系统的技术指标，包括系统必须具有那些功能等。这是系统设计的出发点，它将贯串于整个系统设计的全过程，也是产品设计开发工作成败、好坏的关键，因此必须认真做好这项工作。

2. 系统方案设计

在系统设计任务和技术指标确定以后，即可进行系统的总体方案设计，一般包括：

(1) 单片机芯片的选择。单片机芯片的选择应适合于应用系统的要求。不仅要考虑单片机芯片本身的性能是否能够满足系统的需要，如：执行速度、中断功能、I/O 驱动能力与数量、系统功耗以及抗干扰性能等，同时还要考虑开发和使用时是否方便、市场供应情况与价格、封装形式等其它因素。

(2) 外围电路芯片和器件的选择。仅仅一片单片机芯片是不能构成一个完整的嵌入式系统的。一个典型的系统往往由输入部分(按键、A/D、各种类型的传感器与输入接口转换电路)，输出部分(指示灯、LED 显示、LCD 显示、各种类型的传动控制部件)，存储器(用于系统数据记录与保存)，通信接口(用于向上位机交换数据、构成联网应用)，电源供电等多个单元组成。这些不同的单元涉及到模拟、数字、弱电、强电以及它们相互之间的协调配合、转换、驱动、抗干扰等。因此，对于外围芯片和器件的选择，整个电路的设计，系统硬件机械结构的设计，接插件的选择，甚至产品结构、生产工艺等，都要进行全面和细致的考虑。任何一个忽视和不完善，都会给整个系统带来隐患，甚至造成系统设计和开发的失败。

(3) 综合考虑软、硬件的分工与配合。单片机嵌入式系统中的硬件和软件具有一定的互换性，有些功能可以用硬件实现，也可以用软件来实现，因此，在方案设计阶段要认真考虑软、硬件的分工和配合。采用软件实现功能可以简化硬件结构，降低成本，但软件系统则相应的复杂化，增加了软件设计的工作量。而用硬件实现功能则可以缩短系统的开周期，使软件设计简单，相对提高了系统的可靠性，但可能提高了成本。在设计过程中，软、硬件的分工与配合需要取得协调，才能设计出好的应用系统。

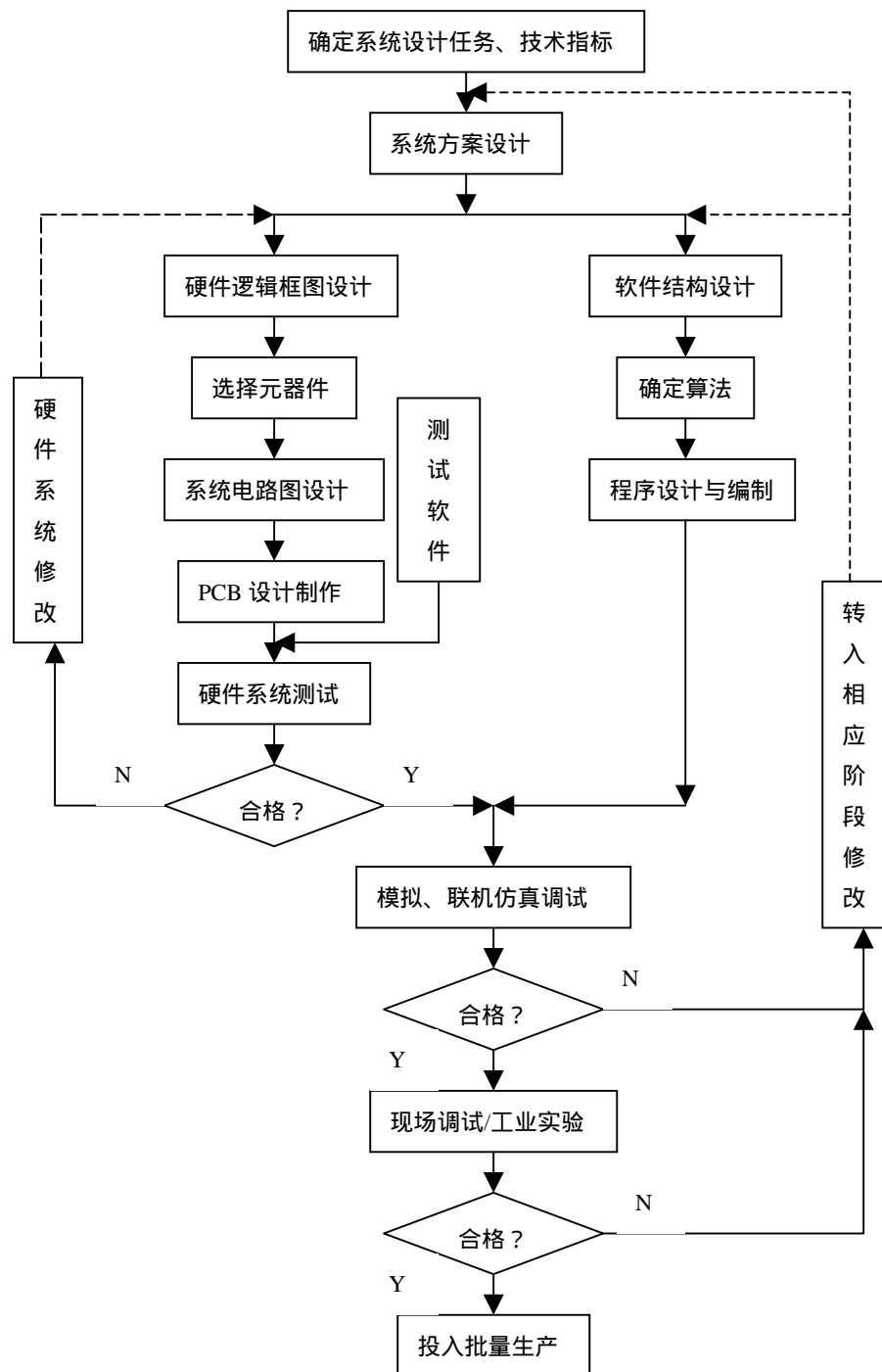


图 4-1 单片机嵌入式系统开发过程

3. 硬件系统设计

开发人员在全面了解要设计开发系统所具备的功能和要求，制定出整体的系统设计方案后，接下来就是根据具体的需求和设计方案，选择能可靠实现全部功能的单片机芯片和相应的外围电路器件，设计整个系统的电原理图。原理图设计完成后，还要根据实际需要设计相应的印刷板（PCB）图。这个阶段常使用的软件平台是电子电路 CAD 软件，如 PROTEL 等。

单片机嵌入式系统的硬件系统设计是一个综合能力的表现，它全面反映和体现了设计开发人员的所具有的技术水平和创新设计能力。比如说，设计一个具备相同功能的单片机嵌入式系统，如采用传统并行总线扩展外围设备的设计思路，设计出的硬件系统就相对庞大和复杂，因为仅地址线 and 数据线就有 $16+8=24$ 根，还需要相应的锁存器和地址译码器等器件，稳定性、抗干扰性都相对差一些。如采用新型的单片机，CMOS 器件，选用串行接口的大容量存储器、AD/DA 等器件，就可减少硬件开发的工作量，大大缩短系统设计开发的周期，同时也提高了系统的可靠性。

4. 系统软件设计编写

在硬件系统设计的基础上，则要根据系统的功能要求和硬件电路的结构设计和编写系统软件。作为单片机系统软件设计人员，应该具备扎实的硬件功底，不仅是对系统的功能和要求有深入的了解，而且对实现的硬件系统、使用的芯片和外围电路的性能也要很好的掌握。这样才能设计出可靠的系统程序。

一个嵌入式系统的系统软件实际上就是该系统的监控程序。对于一些小型嵌入式系统的应用程序一般采用汇编语言编写。对于中、大型的嵌入式系统，常采用高级语言（如：C 语言、Basic 语言）来编写。软件设计和编辑是开发嵌入式系统过程中最重要也是最困难的任务之一，因为它直接关系到实现系统的功能和系统的性能。

通常在编制程序前应对系统要实现的功能、硬件系统的结构和电路、系统中使用的单片机和外围器件进行全面仔细和深入的了解，对系统软件的结构进行全面和完整的设计，编制程序流程图。系统程序的设计应实现结构化、模块化、子程序化，这不仅便于调试，还便于修改。

要特别注意的是，设计编写嵌入式系统的软件同编写其它类型的软件程序有较大的区别。由于嵌入式系统是直接面对硬件、控制对象的，因此，设计编写嵌入式系统的程序需要考虑更多的硬件细节，要掌握和使用很多软件技巧，要多学习、多实践。如，嵌入式系统程序的设计要仔细地考虑和划分程序存储器、数据存储器；合理定义、安排和使用各种变量；尽量使用位变量和位标志变量，优化程序，节省内存容量；估算子程序调用和嵌套的最大级数，预留出足够的堆栈的空间等等。

5. 系统调试

当硬件和软件设计好后，就可以进行系统调试了。硬件电路系统调试检查分为静态检查和动态检查。硬件的静态检查主要检查电路制作的正确性，如路线、焊接等。动态检查一般首先要使用仿真系统（对于采用 ISP 技术的系统可直接）输入各种单元部分的系统调试和诊断程序，检查系统的各个部分的功能是否能正常工作。硬件电路调试完成后可进行系统的软硬件联调。先将各功能模块程序分别调试完毕，然后组合，进行完整的系统运行程序调试。最后还要进行各种工业测试和现场测试，考验系统在实际应用环境中是否能正常可靠的工作，是否达到设计的性能和指标。

系统的调试往往要经过多次的反复。硬件系统设计的不足、软件程序中的漏洞，都可能是造成系统调试出现问题。系统调试要具备相当水平和实践经验，它全面反映了嵌入式系统设计开发者的水平和能力，但又不是能够通过书本的理论学习就能够掌握的。因此，学习单

片机嵌入式系统的设计与开发，要非常注重实际的动手练习，加强这方面的实践和积累。

4.2 单片机嵌入式系统的开发工具与环境

4.2.1 单片机嵌入式系统的程序设计语言

在掌握单片机结构和系统设计基础上，根据系统的设计和系统的功能要求就可以编写系统应用程序。掌握程序设计的方法和技术对于嵌入式系统的学习和应用开发具有十分重要的地位。

开发单片机嵌入式系统所用的程序设计语言可分为三类：机器语言、汇编语言和高级语言。

1. 机器语言

机器语言是完全面向芯片的语言，由二进制码“0”和“1”组成。在单片机的程序存储器中存放就是以“0”和“1”构成的二进制序列指令字，它是单片机 CPU 直接识别和执行的语言。用机器语言表示的程序称为机器语言程序或目标程序。如下面一条 AVR 机器语言代码：

```
0000110000000001
```

就是将 AVR 单片机内部的寄存器单元 R0 和 R1 的内容相加，结果保存在 R0 中。

采用机器语言编程不仅难学、难记，而且也不易理解和调试，因此人们不直接使用机器语言来编写系统程序，往往使用汇编语言或高级语言编写程序。不过，无论使用汇编语言还是高级语言来编写系统程序，最终都需要使用相应的开发软件系统（一般在软件开发平台中的都提供编译软件系统）将其编译成机器语言，生成目标程序的二进制代码文件（.bin 或 .hex），然后再把目标代码写入单片机的程序存储器中，最后由单片机的 CPU 执行。

2. 汇编语言

汇编语言是一种符号化的语言，它使用一些方便记忆特定的助记符（特定的英文字符）来代替机器指令。如“ADD”表示加，“MOV”表示传送等。上面的 AVR 机器指令用汇编语言表示为：ADD R0, R1

用汇编语言编写的程序称为汇编语言程序，显然，它比机器语言易学、易记。但是，汇编语言也是面向机器的，也属于低级语言。由于各种单片机的机器指令不同，每一类单片机的汇编语言也是不同的，如 8051 的汇编语言同 AVR 的汇编语言是完全不一样的。

传统开发单片机嵌入式系统主要是用汇编语言编写系统程序。学习和采用汇编语言开发系统程序的优点是：能够全面和深入的理解单片机硬件的功能，充分发挥单片机的硬件特性。但由于汇编语言编写的程序可读性、可移植性（各种单片机的机器指令不同）和结构性都较差，因此采用汇编语言来开发单片机应用系统程序的比较麻烦，调试和排错也比较困难，产品开发周期长，同时要求软件设计人员要具备相当高的能力和经验。

3. 高级语言

高级语言是一种基本不依赖硬件的程序设计语言（当然，对于开发直接面向单片机的嵌入式系统，系统程序的设计总是要同硬件有关的），它面向问题或过程，其形式类似自然语言和数学公式。为了提高编写系统应用程序的效率，改善程序的可读性和可移植性，缩短产品的开发周期，采用高级语言来开发单片机系统已成为当前的发展趋势。

作为一个有经验的单片机嵌入式系统开发人员，应能同时掌握和使用汇编语言和高级语言设计系统程序。概括起来说，基于高级语言开发单片机系统具有语言简洁，使用方便灵活，可移植性好，表达能力强，可进行结构化程序设计等优点。对于开发大型和复杂的嵌入式系统来讲，采用高级程序设计语言进行系统开发的效率比使用汇编语言高几倍甚至

几十倍。但对于小型、简易的系统，或有定时精确，高测量精度要求的系统，使用汇编语言则具有优势。在许多情况下，采用高级语言嵌入汇编程序的软件设计技术往往是最有效的方法。

如果你对单片机的内部结构和汇编语言根本不了解，请先不要用 C 语言编程。

如果你对单片机的内部结构和汇编语言根本不了解，也写不出好的单片机的 C 程序。

不管是使用汇编语言还是高级语言来开发单片机系统程序，都需要一个专用的软件平台把软件设计人员编写的源程序“翻译”成二进制的机器指令代码，这个“翻译”过程对汇编语言来讲称为汇编，对高级语言来讲，它包括编译和连接两个过程。因此，一个性能优良的，专门用于开发单片机的软件平台和环境也是必不可少的开发工具。

4.2.2 单片机嵌入式系统的开发软件平台

单片机嵌入式系统的设计和开发需要一个好的软件开发平台的支持。一个好的单片机嵌入式系统的开发软件通常具备以下几个重要的功能：

- 单片机系统程序编写和运行代码的生成。（编辑、编译功能）
嵌入式系统开发平台支持用户采用专用汇编程序设计语言或高级程序设计语言（C、Basic 等）编写嵌入式系统控制程序的源代码，并将源代码编译连接生成可在单片机中执行的二进制代码（Hex、Bin）。
- 软件模拟仿真。
提供一个纯软件的仿真环境，在此环境的支持下，单片机的系统程序可以进行模拟的运行，以实现第一步的软件调试和排错功能。
- 在线仿真功能。
与专用的仿真器配合，提供一个硬件在线的实时仿真调试环境（见下节在线仿真器）。用户将编写好的目标系统运行代码下载到仿真器中，通过开发系统软件控制仿真器中程序的运行，同时观察硬件系统的运行结果，分析、调试和排除系统中存在的问题。
- 程序下载烧入功能。
与专用的编程器配合或使用 ISP 技术，将二进制运行代码写入到单片机的程序存储器中。

要熟练掌握和应用单片机来设计开发嵌入式系统，除了对所使用的单片机要有全面和深入的了解外，配备和使用一套好的开发环境和开发平台也是必不可缺的。在嵌入式系统的设计开发中，选用了好的开发工具和开发平台，往往能加速嵌入式应用系统的研制开发、调试、生产和维修，起到事半功倍的效果。

国内外许多公司根据不同单片机的性能和特点，研制推出了各种类型的用于开发单片机嵌入式系统的单片机开发装置和软件开发平台。不同类型的单片机使用的开发系统是不同的。对同一类型的单片机来讲，也有多种类型和功能的开发装置和开发平台。价格便宜、性能适中的系统在几百元，高性能的开发系统则要数千元到上万元，甚至仅仅一套软件开发平台就要上万元。虽然设计开发一个嵌入式系统，你可以选用多家公司、多种类型的单片机，但你在决定学习和使用哪种单片机时，应对单片机的性能价格，开发装置和开发平台的性能价格，以及是否方便使用等，几方面做一个综合的评估。

4.2.3 单片机嵌入式系统的硬件开发工具

在学习和应用单片机来设计开发嵌入式系统的过程中，一般应配备两种硬件设备：仿真

器和编程烧入器。

1. 仿真器

仿真器也称为实时在线仿真开发机，它实际是一个特殊的单片机嵌入式系统，一般是由专业公司研制和生产。仿真器的内部含有一个具有“透明性”和“可控性”的 MCU，可以代替被开发系统（目标系统）中的 MCU 工作，既用仿真器的资源来仿真目标机。仿真器是用来开发和调试嵌入式系统的重要工具之一，是使用软件和硬件技术相结合的嵌入式系统调试及排除故障的一种手段。

仿真器一般使用串行口（COM 口或 USB 接口）或并行口（打印机口）同 PC 机通信，并提供一个与目标机系统上的 MCU 芯片引脚相同的插接口（仿真口）。使用时，将目标机上的 MCU 取下，插上仿真器的仿真口，仿真器的通信口与 PC 连接（图 4-2）。

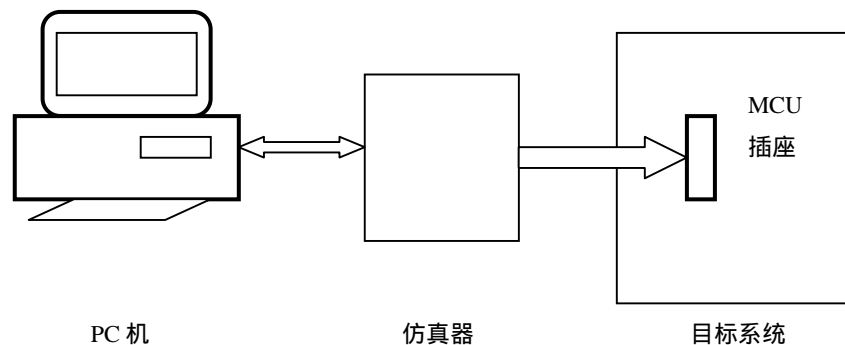


图 4-2 仿真器的连接与使用示意图

仿真器所提供的 MCU 称为仿真 MCU，它与目标系统上使用的 MCU 是相同系列的，其控制作用和工作过程几乎完全一样。使用者将编写好的目标系统的软件下载到仿真器中，并通过 PC 机来控制仿真器中程序的运行，同时观察系统外围器件和设备的运行结果，分析、调试和排除系统中存在的问题。这种运行调试方法称为在线仿真。

为了能够实现 MCU 的仿真功能，仿真开发系统通常具有的一些基本功能为：

- 可控性。可以根据调试的需要，控制目标程序的运行方式，如单步、连续、带断点等多种运行方式。
- 透明性。能对 MCU 的各个部分进行监控，如查看和设置内存单元、寄存器、I/O 的数据。

仿真开发系统都必须配备一套在 PC 机上运行的专用仿真开发软件系统，用以配合和实现仿真器的在线仿真调试工作。因此嵌入式系统的开发人员，除了要掌握单片机和嵌入式系统的应用和设计能力，还应熟练地掌握和使用仿真器和仿真系统软件。

2. 编程烧入器

编程烧入器也称为程序烧入器或编程器，它的作用是将开发人员编写生成的嵌入式系统的二进制运行代码下载（写入）到单片机的程序存储器中。高档的编程器一般称作万用编程器，它不仅可以下载运行代码到多种类型和型号的单片机中，还可以对 EPROM、PAL、GAL 等多种器件进行编程。

目前，性能较好的仿真器也都具备了对其可仿真的 MCU 的编程功能，这样就可以不用专门购置编程器设备。当单片机芯片具备 ISP 功能时，程序的下载更加简单了，一般通过 PC 的串行口或并行口，使用简单的软件就可将编译生成的嵌入式系统的运行代码直接下载到 MCU 中。

现在一些新型的单片机内部集成了一种标准的串行接口 JTAG，专门用于在线仿真调试

和程序下载。使用 JTAG，可以简化仿真器（无需使用专用的仿真 MCU）和编程器的结构，甚至可以淘汰专用仿真器和编程器，而将 PC 直接同系统板连接（一般经过简单的隔离），利用系统板上的 MCU 直接实现在线的仿真调试，这为嵌入式系统的设计提供了更为有效和方便的开发手段和方法。当系统使用贴片封装或 BGA 封装的小体积芯片和器件时，它的优点尤为突出。

4.2.4 AVR 单片机嵌入式系统的软件开发平台

ATMEL 公司为开发使用 AVR 单片机提供了一套免费的集成开发平台：AVR Studio (<http://www.atmel.com>)。该软件平台支持 AVR 汇编程序的编辑、编译、连接以及生成目标代码。同时该软件平台可以配合 ATMEL 公司设计推出的多种类型的仿真器，如 ICE40、ICE50、JTAG ICE、JTAGICE mkII 等，以实现系统的在线硬件仿真调试功能和目标代码的下载功能。

此外，一些第三方公司也推出了许多采用高级语言编程的开发平台，用于 AVR 单片机系统的开发。

采用高级程序语言 C 的开发平台有：

- ICCAVR (<http://www.imagecraft.com/software>)
- CodeVision AVR (<http://www.hpinfotech.ro>)
- IAR Systems (<http://www.iar.com>)
- AVR GCC (<http://www.avrfreaks.net>)

采用高级程序语言 BASIC 的开发平台有：

- BASIC AVR (<http://www.digimok.com>)
- FastAVR Basic (<http://www.fastavr.com>)
- BASCOM-AVR (<http://www.mcselec.com>)

其中 AVR Studio 和 AVR GCC 是完全免费的软件，而 ICCAVR、CodeVisionAVR、IAR System、BASCOM-AVR 等均为商业软件，但它们都有提供给用户试用的 DEMO 版软件（在功能上或时间上有限制），可以从网上免费下载。在学习单片机嵌入式开发的起步阶段，完全可以使用这些 DEMO 版的开发平台。

本书将介绍 ATMEL 公司提供的 AVR Studio 4.11 的使用。但主要还是以高级程序设计语言 C 为设计手段的 CodeVisionAVR(简称 CVAVR)作为本书使用的开发软件。因为采用高级程序设计语言开发嵌入式系统已成为当前的发展趋势。

由于 AVR 单片机具有 ISP 性能，其程序存储器可多次编程、在线下载的优点；加上采用高级程序设计语言来开发单片机系统具有语言简洁，使用方便灵活，表达能力强，可进行结构化程序设计等优点；再配合软件模拟仿真调试；使得我们可以不必购买价格在几千元的仿真器和编程器，就能够很好的学习和掌握 AVR 单片机嵌入式系统的设计和开发。

为配合本书的学习，我们专门设计了一套半开放式的，性能良好，方便学习，制作简便的“AVR-51 多功能单片机系统学习、实践开发板”。它不仅便于处学者的学习和实践的使用，同时也适用工程师作为产品设计和开发的前期使用。建议有条件的学习者，按本书提供的设计和指南，自己动手 DIY 制作，配合本书用于 AVR 单片机的学习和实践使用，这是一个真正的起步。

一． 汇编语言开发平台

ATMEL 公司提供免费的 AVR 汇编语言编译器。在 AVR STUDIO 中已经将 AVR 汇编语言编译器集成在一起，你可以在 AVR STUDIO 中完成 AVR 汇编代码的编辑，编译和连接，生成可下载的运行代码。

由于 AVR 的指令与 C 语言有很强的对应性，再加上 AVR 汇编语言编译器有强大的预编

译能力，如宏，表达式计算能力等，所以使用 AVR 汇编语言写出的代码可读性也是很强的。如果你不想花很多的钱在您的编译工具上的话，AVR STUDIO 是一个不错的选择。

另外在 AVR STUDIO 中还提供一个纯软件的软仿真环境，在此软件环境的支持下，单片机的系统程序可以在 PC 上进行模拟的运行（完全脱离硬件环境），以实现第一步的软件调试和排错功能。

大部分的高级语言开发平台不具备纯软件的软仿真环境和在线实时仿真的功能，但他们都能够生成在 AVR STUDIO 中可以使用的，用于仿真的文件，这样高级语言的开发平台与 AVR STUDIO 配合使用，就能构成和实现一个基于高级语言的软仿真和在线实时仿真调试的开发环境。

二．高级语言开发台

由于 AVR 单片机自身的优势，吸引了大量的第三方厂商为 AVR 单片机编写开发出各种各样的 AVR 高级语言编译器和开发软件平台。很少有一个 8 位单片机能有这么众多的编译器以及开发平台可供选择。根据高级语言的种类，AVR 有 C、BASIC、PASCAL，ADA 等多种语言的开发平台。如果您对其中的一种语言比较熟悉的话，那您就不必重新学习另一种语言，而直接选择您熟悉的进行开发。而且这些编译器的厂商在其网站上都提供了免费试用版本的下载，用户可以在试用了一段时间，在比较其之间的优缺点之后，选择购买。

下面就对其中的几种高级语言编译器和开发软件平台进行一个简略的介绍。

(1) IAR Systems 的 Embedded Workbench 编译器

IAR Systems 是非常著名的嵌入式系统的编译工具的提供商。如果您访问其网站，您就会发现它几乎为所有的 8 位、16 位、32 位的单片机和微处理器提供 C 编译器，由此可见也其在业界的地位。正因为如此，当初 ATMEL 在开发和设计 AVR 时，决定咨询 IAR Systems 的编译器设计工程师，看如何设计 AVR 能够使高级语言的编译效率更高。此后，IAR Systems 与 ATMEL 一直保持着良好的而又紧密的合作关系，这使其设计出来的编译器的编译效率也是同类中最高的。但是价格较高。

IAR Systems 的 Embedded Workbench 集成了一个集成环境包括编译器和图形化的调试工具，能够完成系统的设计，测试和文档工作。您可以在其中完全无缝的完成新建项目，编辑源文件，编译，链接和调试等工作。可以同时打开多个项目。很容易扩展集成诸如代码分析等外部工具。

其 C 编译器和汇编编译器支持几乎所有 AVR 芯片，具备以下特点：

- C 编译器支持 ISO/ANSI C 的标准 C 和可选的 Embedded C++ 编译器。
- 所有代码都可重入。
- 有多种存储器模型和指针类型，以充分利用存储器。
- 内建针对 AVR 优化的选项，多重的代码大小和执行速度的优化控制。
- 针对 AVR 的语言扩展以适应嵌入式编程。
- 新增的强大全局优化器。
- 可以直接在 C/C++ 中写快速易用的中断处理函数。
- 高效的 32 位和 64 位的 IEEE 兼容的浮点运算。
- 扩展的 C 和 EC++ 的函数库，并对数学和浮点运算。

IAR Systems 的网站地址为 <http://www.iar.com>。

(2) IMAGE CRAFT 的 ICCAVR 编译器

这是 IMAGE CRAFT 提供一款低成本高性能的 C 语言编译器，其包括了 C 编译器和 IDE 集成编译环境，简称 ICCAVR。

其支持除AT90S1200外的所有AT90系列和ATmega系列，Tiny26和AT94KFPSLIC器件，自动生成对I/O寄存器操作的I/O指令。其编译器是对LCC通用C编译器的移植，完全支持标准的ANSI C，支持32位的长整数和32位的单精度浮点数运算，支持在线汇编，同时也能和单独的汇编模块进行接口。拥有包括printf，存储器分配，字符串和数学函数的ANSI C库函数的子集库函数和针对特定目标访问片上E²PROM和各种片上外设的库函数。可以生成用于AVR STUDIO源码级调试的目标文件。在其IDE中包含了对项目的管理，源文件的编辑，编译和链接源选的设置，还有内嵌的ISP编程界面。

但是由于其源自通用C编译器，其几乎完全不支持位寻址。

ImageCraft的网站地址为<http://www.imagecraft.com>，提供30天的试用版下载。国内广州双龙公司是ICCAVR的代理商。

(3) HP Info Tech 的 CodeVision AVR 编译器

CodeVision AVR 是 HP Info Tech 专门为 AVR 设计的一款低成本的 C 语言编译器，它产生的代码非常严密，效率很高。它不仅包括了 AVR C 编译器，同时也是一个集成 IDE 的 AVR 开发平台，简称 CVAVR。

CVAVR 支持所有片内含有 RAM 的 AVR 芯片，具备以下特点：

- 支持 bit、char、short、int、long、float 以及指针等多种数据类型，充分利用存储器。
- 内建针对 AVR 优化的多种选项。
- 支持内嵌汇编。
- 扩展的一些标准的外部器件支持和接口函数，如：标准字符 LCD 显示器、I2C 接口、SPI 接口、延时、BCD 码与格雷码转换等。
- 可以直接在 C/C++ 中写快速易用的中断处理函数。
- 高效的 32 位和 64 位的 IEEE 兼容的浮点运算。
- 扩展的 C 和 EC++ 的函数库，并对数学和浮点运算。

HP Info Tech 的网站地址为<http://www.hpinfotech.ro>，提供试用板（2K代码限制）的下载。清华大学出版社出版的《嵌入式C编程与Atmel AVR》一书中，对CVAVR的使用和程序设计给出了全面和详细的介绍。本书也采用CVAVR作为主要开发语言平台。

(4) GNU GCC AVR

GNU GCC AVR 是著名的自由软件编译器的 GNU GCC 的 AVR 平台的移植。其包括两部分，编译和链接的命令行程序包和针对 AVR Libc 函数库。如同其他 GNU 协议下的软件一样，所有这些都是以源程序的形式发布，用户可以根据其自身的计算机平台进行配置编译，生成适合用户自身计算机平台的可执行版本的 GNU GCC AVR。对于 WINDOWS 用户，也有已经预先编译好的二进制版本可供下载。

GCCAVR 的特点为：

- 所有源代码都是向用户开放，完全免费。
- GCCAVR 本身支持 ANSI C/C++/EMBEDDED C++。
- GCCAVR 本身的编译效率和稳定性，编译后代码执行效率仅次于 IAR Systems 的 Embedded Workbench。
- 支持几乎所有的 AVR 器件。
- 包括兼容 ANSI C 的部分标准函数库和针对 AVR 的各个外设的函数库。
- 缺乏专业的技术支持，缺乏图形的集成编辑环境（IDE），所有程序都是命令行执行的。

用户可以在<http://www.avrfreaks.net>上获得最新的 GNU GCC AVR 软件包。

(5) 几种 C 语言开发平台的对比

表 4.1 给出上述 4 种 C 语言开发平台的性能价格对比。

表 4.1 AVR 四种 C 语言开发平台的比较

	IAR	Imagecraft	CodeVision	GNU GCC
代码效率	+++	++	++	++
价格	\$\$	\$	\$	Free
易用性	++	+++	+++	+
与 AVR Studio 集成度	++	+++	+++	++
技术支持	+	+++	+++	-

(5) BASCOM-AVR

BASCOM-AVR 是荷兰 MCS Electronics 公司设计的一款针对 AVR 系列单片机的 BASIC 编译器，其软件包由 BACIS 编译器和 IDE 集成编辑环境组成。IDE 集成编辑环境支持对源代码的高亮显示，提供上下文提示，以提高编码效率。IDE 集成编辑环境还包含了一系列工具，图形化的模拟仿真环境，无需连结硬件，你就可以通过它对 LCD，LED，UART，和 PIO 端口进行仿真。此外，你还可以在 IDE 集成环境中对目标板进行 ISP 编程。其主要特点有：

- 采用可带语句标示符的结构型 BASIC 高级程序设计语言编程，程序语句和 Microsoft VB/QB 高度兼容。
- 结构化的 IF-THEN-ELSE-ENDIF、DO-LOOP、WHILE-WEND、SELECT-CASE 程序设计。
- 变量名和语句标示符长达 32 个字符。
- 支持位(Bit)、字节(Byte)、整型(Integer)、字(Word)、长型(Long)、字符串(String) 多种类型的变量。
- 编译产生的运行代码可在所有带内部存储器的 AVR 微控制器微中运行。
- 为标准LCD显示器，I²C 芯片和单总线协议芯片等扩充了大量的专用语句。
- 内置模拟终端和程序下载功能。
- 自带内置的图形软件仿真平台，并同时支持和采用 AVR STUDIO 作为其软件模拟仿真器。
- 完善的联机帮助功能和大量的例程。

BASCOM-AVR是采用结构型BASIC作为程序设计语言，简单易学，尤其适合中学生、大中专学生学习使用，以及开发一些相对简单的小系统使用。用户可以到MCS Electronics的网站<http://www.mcselec.com>下载试用版（2KB代码限制）。清华大学出版社出版的《AVR单片机BASIC语言编程及开发》一书中，对BASCOM-AVR的使用和程序设计给出了全面和详细的介绍，读者可以参考学习。

4.2.5 AVR 实验开发板

1. STK500 系列开发板

STK500 系列开发板是 ATMEL 公司为 AVR 的 90 系列和 mega 系列的新用户提供了一些的开发板和相应的适配板，以使用户快速入门。当然，如果在开发中用户也可以在这些开发板上完成初步的验证，而免去了自己制板的成本与风险。STK500 系列开发板包括 STK500 主板和 STK501、STK502、STK503、STK504 子板。

STK500 是 ATMEL 推出的主要针对 40 脚及 40 脚以下的 DIP 封装的 90 系列和 mega 系列单片机的开发板。其具有并行和 ISP 编程功能，JTAG 仿真接口。其上还有一些 LED 和按键，通过扁平线可以和单片机的端口连接，观察端口的电平变化或者手动触发端口电平的变化，

这在没有仿真的情况下是非常有用的。在板上除了一个用于和下载程序的 RS232 接口外，还有一个 RS232 接口，通过跳线可以和单片机的 UART 连接，完成与 PC 机进行通信的任务。另外，板上还有一个振荡电路，您可以根据自己的需求选择不同的时钟源驱动您的单片机。图 4-3 所示为 STK500 开发板。

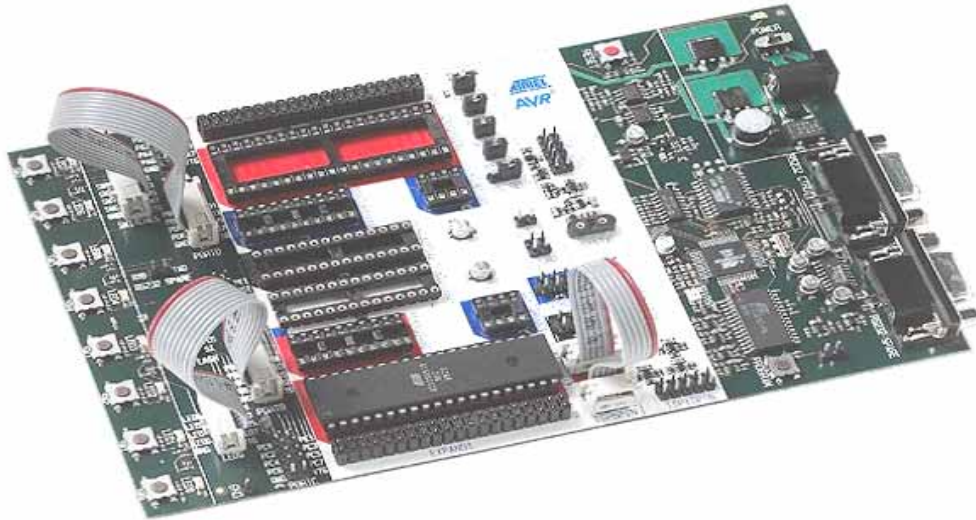


图 4-3 STK500 开发板

由于 ATmega128 只有 TQFP 的封装，所以无法在 STK500 完成开发，于是 ATMEL 开发了便于使用 TQFP 封装的 ATmega103/ATmega64/ATmega128 而设计的 STK500 顶置模块 STK501(图 4-4)。

STK501 作为 STK500 的子板，配有安装 ATmega103/ATmega128 的 ZIF 插座和 PCB 封装，需要安装在 STK500 上才能完成对 ATmega128 开发功能（如图 4-5）。此外，由于 ATmega128 有两个 USART 口，所以在 STK501 上还扩展了一个额外的 RS232 口，以及 32kHz 的 RTC 振荡器。



图 4-4 STK501 子板

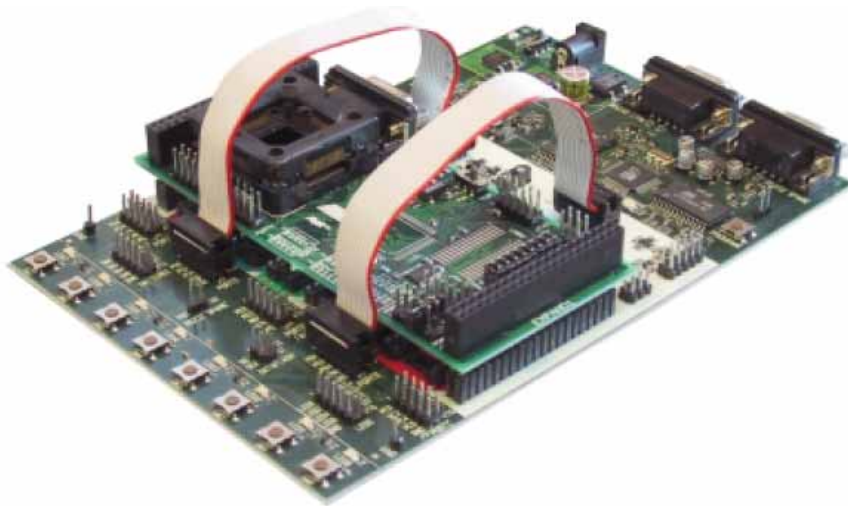


图 4-5 使用 STK500+STK501 开发 ATmega128

2. AVR-51 多功能实验开发板

AVR-51 多功能实验开发板是华东师范大学电子科学技术系 AVR 实验室采用“模块独立、开放、灵活”的设计思想设计的适用于初学者使用的多功能实验开发板。传统的单片机实验系统透明度不高，实验板上的芯片、接口都是固定连接的，硬件只需很少连接，甚至不需要做硬件的连接，就直接进行程序设计，这就把单片机以硬件为主，软硬结合的训练变成了纯软件的训练。

单片机系统具有“硬件决定软件”的特点，硬件设计、接口一旦改变，口地址就会变，程序也就要跟着变。如果把硬件部分简化或是省略，往往不能深刻理解程序的含义，也就谈不上软件的训练。

我们设计的这款“AVR-51 多功能实验开发板”，本着能够使学习者在硬件设计和软件设计全面得到真正的训练和学习的目的，一改以往的固定线路、固定接口、固定芯片的模式，对开发板上的硬件资源采用了全部开放的结构，同时也提供比较丰富的接口，以及构成单片机系统最常使用的同时也是最基本的外围功能模块。实验板上单片机的引脚全部开放，同时将在单片机系统中最常使用的显示、按键、键盘、等都作为独立的开放模块单元，其连接信号接口和电源接口也是开放的。这样用户就可以非常灵活的，根据需要构建自己的系统。

“AVR-51 多功能实验开发板”不仅可以用于配合本书教学实验，而且也适合单片机系统的设计开发人员作为产品开发前期使用的开发板。该板由国内“我们的 AVR”网站生产 (<http://www.ouravr.com>)，读者可以从该网站邮购。

实验板分成三大部分：系统电源、MCU 座和外围功能模块单元。其中外围功能模块单元分成 A—O 共 15 个区域。图 4-6 为“AVR-51 多功能实验开发板”的实物图，图 4-7、4-8 为该板的电原理图。下面对该板的功能和特点进行一些介绍，以方便今后的使用。

1) 系统电源。在 8-12V 输入电压范围内提供高稳定的 5V/1A 的系统电源，配有电源指示灯、极性保护电路及开关。一般工作时，输入电压为 9V，系统电源部分提供实验板工作需要的 5V/1A 电源。同时板上有多个高频和低频的电源滤波电容。

2) MCU 座。实验板的中间部分有两个 40PIN 的锁紧插座，供插入 MCU 使用。

- ✓ 2 个 40PIN 锁紧插座引脚全部是开放的，与外围没有任何的连接。
- ✓ 左边 40PIN 锁紧插座与 JU1 短路排 (6 组)、X2 和 X1 短路排 (2 组)、JU2-GND 和 JU3-Vcc (2 组) 配合，构成与通用的 51 系列 40PIN 单片机引脚兼容的方式，适用

- 于 AT89S5x 系列的单片机以及 AVR 的 ATmega8515 单片机。
- ✓ 右边 40PIN 锁紧插座与 JU4 短路排 (10 组) JU7 短路排 (3 组) JU8 (4 组) 配合, 适用于 AVR 的 ATmega16、ATmega8535、ATmega32 单片机。
- ✓ 左边 40PIN 锁紧插座顶部的 2*5 针的插座为 ISP 下载接口, 用于配合与 STK200、STK300 兼容的 ISP 下载线实现对 AT89S5x 和 AVR 单片机进行程序下载和熔丝配置。

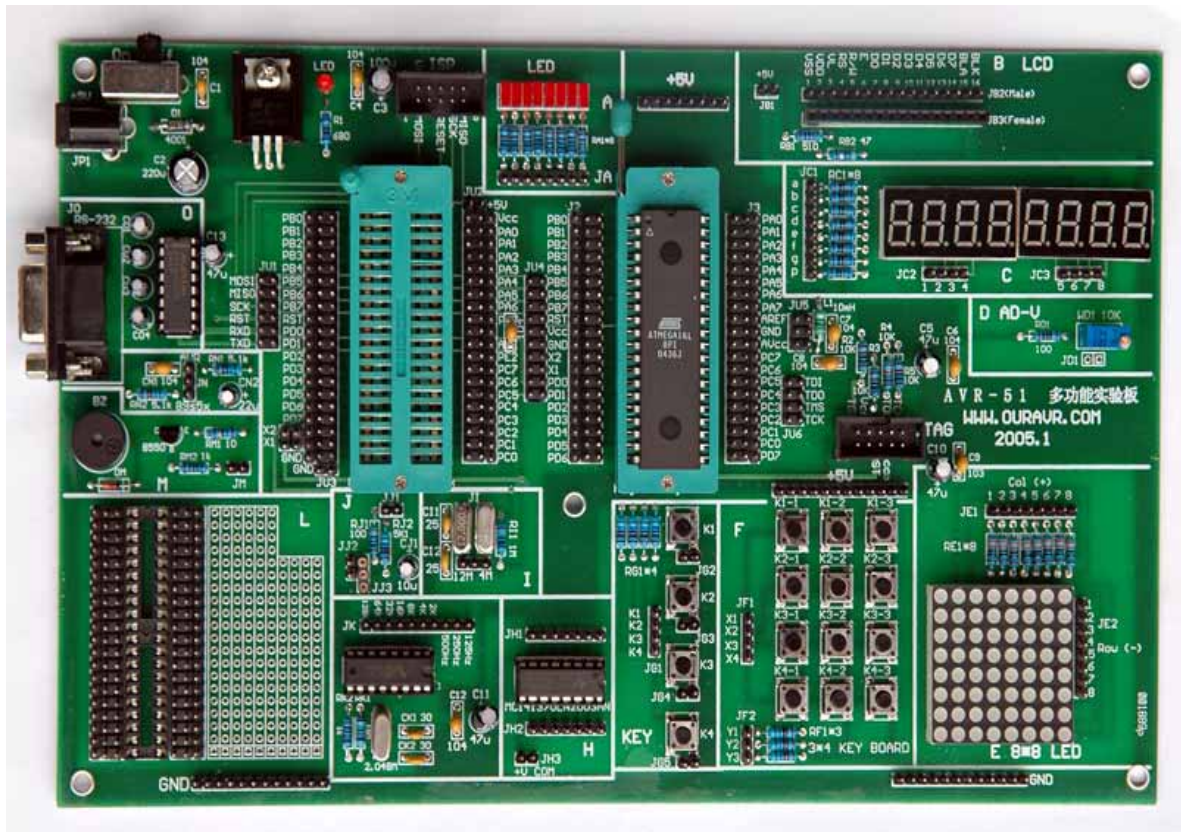


图 4-6 AVR-51 多功能实验开发板

- ✓ 右边 40PIN 锁紧插座的右下角处的 2*5 针的插座为 JTAG 接口, 用于配合使用专用的 JTAG 仿真器实现对含有 JTAG 接口的 AVR 单片机进行在线的实时仿真调试、程序下载以及熔丝位的配置。
- ✓ 由于 2 个 40PIN 锁紧插座引脚全部是开放的, 因此当 2 个锁紧插座中的任何一个插入单片机后, 另一个就可作为扩展插座使用, 如插入 DIP 封装的 EEPROM 芯片 24C02 做 I2C 通信实验。
- ✓ 原则上该板可以适合任何 DIP 封装的, 引脚在 40PIN 以内的单片机的使用。但此时需要用户使用更多的连接线进行必要的连接, 同时还要考虑匹配适当的编程器, 以及该单片机 I/O 口的驱动能力问题 (>10mA)。
- ✓ 该板最适合 40PIN 以内 DIP 封装的 AVR 单片机和 AT89S5x 系列的 51 单片机使用。因为它们使用相同的 ISP 编程技术, 配合与 STK200、STK300 兼容的 ISP 下载线都能实现程序的下载, 不必另外配置专用的编程器。同时他们的 I/O 驱动能力都大于 20mA, 可以直接驱动 LED 显示。

3) 使用 ATmega16 的缺省连接。在“AVR-51 多功能实验开发板”上对使用 ATmega16 采用缺省的短路片连接, 构成最小系统。

将 ATmega16 插入右边的 40PIN 插座中锁紧。将 JU4 短路排 (10 组) JU7 短路排 (3 组) JU8 (4 组) 共 17 组短路排用短路片短路。17 组短路排的作用见表 4.2。

表 4.2 使用 ATmega16 时短路片的缺省连接

	连 接		说 明
JU4	ISP MOSI	PB5	与 ISP 接口连通, 用于 ISP 方式的下载程序和配置熔丝位。 如在实验中 PB5、PB6、PB7 还作为 I/O 使用时, 注意不要直接与 5V 电源或地连接, 防止与 ISP 发生冲突, 造成无法进行程序下载
	ISP MISO	PB6	
	ISP SCK	PB7	
	ISP RESET	RST	
	+5V	Vcc	提供芯片的工作电源
	GND	GND	芯片接地引脚
	外部晶体	X2	当使用芯片内部 RC 振荡源时, X2、X1 的短路片可以不用。
	外部晶体	X1	
	MAX202 R2out	PD0	当不使用芯片的 USART 功能时, 这 2 个短路片可以不用, PD0、PD1 可作为普通 I/O 使用
MAX202 T2in	PD1		
JU7	C7	AREF	ADC 参考电压输入端。确省接 C7 滤波电容, 使用内部参考
	GND	GND	芯片接地引脚
	+5V	AVcc	提供芯片 A 口 (ADC) 的工作电源, 通过 L1、C8 滤波。
JU8	JTAG TD1	PC5	与 JTAG 接口连通, 用于 JTAG 方式的在线仿真调试, 以及下载程序和配置熔丝位。如不使用 JTAG 方式, 这 4 个短路片可以不用, 此时 PC5、PC4、PC3、PC2 作为普通 I/O 使用
	JTAG TDO	PC4	
	JTAG TMS	PC3	
	JTAG TCK	PC2	
JN	JN 的中心针与 AVR 针短接		AVR 采用低电平复位, 正常工作 RESET 接高电平

- 4) 外围功能模块单元。外围功能模块单元分成 A—O 共 15 个区域。
- A 区, 8 路 LED 发光二极管, 用于输出显示。
- B 区, 标准 2*16 字符的 LCD 液晶显示器接口, 同时还兼容 3310 图形液晶显示器接口。3310 图形液晶显示器可以显示 84*48 点阵, 能显示 3 行中文, 每行 7 个汉字。
- C 区, 8 位 8 段 LED 数码管显示器, 采用共阴接法, 动态扫描方式点亮。
- D 区, 这是一个由精密电位器与电源组成的 0~5V 的可调直流电压, 可提供 0~5V 可调直流电压信号, 作为 ADC 的输入电压源。用于实现 AD 转换、直流电压表的实验。
- E 区, 8*8LED 点阵显示模块, 用于做点阵字符、小型广告屏, 电梯运行指示器等实验。
- F 区, 4*3 矩阵键盘。
- G 区, 4 个独立按键。用于按键输入、外部中断输入等。使用 JG1-4 短路片可以将一位 I/O 口对地短路。连接 RESET 线, 可以实现人工复位。
- H 区, 7 路 300mA 功率驱动。用于驱动小型步进电机、继电器等。
- I 区, 系统时钟选择。当单片机使用外部晶体振荡时, 可选择 4M 或 11.0592M 的晶体。使用 11.0592M 晶体可以产生高精度的 RS-232 通信波特率。
- J 区, 该区电路可变化为 2 种应用。其一, 作为 RC 滤波电路用于对 PWM 输出的平滑。其二用于使用 DS1802 数字温度传感器做单总线实验。
- K 区, 该区模块使用一个 2.048MHz 的晶体振荡器, 经过 CD4060 的分频, 提供 125Hz~128KHz 之间, 占空比 50% 的 10 种频率方波脉冲信号, 可作为频率、周期测量实验的输入信号, 外部计数的输入信号等。
- L 区, 40PIN 窄型扩展座和标准 PCB 板。用于扩展和插入其它外围芯片。

M 区, 该区提供一个无源蜂鸣器, 由 MCU 的某一引脚输出一定频率的方波, 就可以发出声响, 可以作为一个简单的外设。

N 区, 单片机外部复位电路选择。由于 AVR 系列单片机采用低电平复位, 而 51 系列单片机为高电平复位, 因此需要根据使用单片机的类型正确的选择复位方式。选择通过 JN 短路片确定。

O 区, RS-232 串行接口单元。通过 MAX202 电平转换, 连接单片机的 USART (UART) 口, 实现与 PC 的异步通信。

综上所述, 整个实验开发系统板提供了一下功能:

✓ 系统的资源与能源

电源供电系统单元、可调直流电压单元 (D) 和脉冲信号发生器单元 (K) 为实验提供了必要的条件和手段。电源供电系统为实验板提供了高精度的电源; D 单元为 A/D 转换实验提供信号; K 单元则为计数、频率测量等提供了信号源。

✓ 基本的输入输出设备或接口

这部分主要为基本实验提供必要的外围设备: 4 键按键单元 (G)、4×3 键盘单元 (F)、无源蜂鸣器单元 (M)、8×8 LED 点阵式显示单元 (E)、8 位 LED 数码管显示单元 (C)、2×16 字符型液晶显示单元 (B)、LED×8 显示单元 (A)、功率驱动单元 (H) 以及 RS-232 (O) 等。

✓ 系统的扩展和多功能

单片机引脚的全部开放, 采用 2 个 40PIN 的锁紧插座, 扩展插座 (L)、系统时钟选择 (I)、外部复位选择 (N) 等使得实验板可以非常灵活的扩展、组合, 同时也适合多种类型以及不同引脚数的单片机使用。用户能够根据需要, 采用不同的连接方式, 构成新的系统电路或新的 MCU 系统, 以兼容更多的实验和应用的需要。

由于 I2C 总线、SPI 总线、单总线 (1 WIRE) 通信接口均是串行通信方式, 使用连接线少, 因此在做这些实验时, 只要将相关的外围芯片插入扩展座中, 再使用几根连接线联上电源、地, 和接口引脚就可以了。

对于 AVR 单片机所有的基本功能和单元实验, 如: I/O 使用、ADC、时钟、中断、PWM, 键盘、按键、LED、LED 数码、LCD 显示、测频率、测周期 (利用板上的 125Hz-128K 的方波源)、功率驱动、蜂鸣器、RS-232 等, 都可以在板上实现。

如果将这些单元有机的组合, 就可以完成和实现一些实际电子产品的设计, 如带音乐报时的实时时钟、秒表、频率计、速度表、电话拨号器、电压表、LED 广告屏, 计算器……。

4.3 自制 ISP 下载电缆

在本节中, 我们将介绍一个与 STK200/STK300 兼容的 ISP 下载电缆。该下载电缆支持所有使用 ISP 技术的 AVR 芯片, 同时也支持 ATMEL 公司 51 系列兼容芯片 AT89S51、AT89S52、AT89S53、89S8252。图 4-7 为 AVR ISP 下载电缆电原理图。

ISP 编程的原理是使用 PC 机的并行口来驱动 ISP 编程所需的信号波形, 实现对 AVR 的程序下载和熔丝位的配置编程。出于安全的考虑, 为了防止使用中误操作而损坏 PC 机的并行口, 图中使用 74HC244 作为缓冲, 以保护计算机的并行口。74HC244 由目标板供电, V_{TC} 经过 D1 (极性保护) 和 D2 (5.1 V 限压保护) 使 74HC244 工作在 +3~+5V。R1 为 MISO 信号线

的上拉电阻。

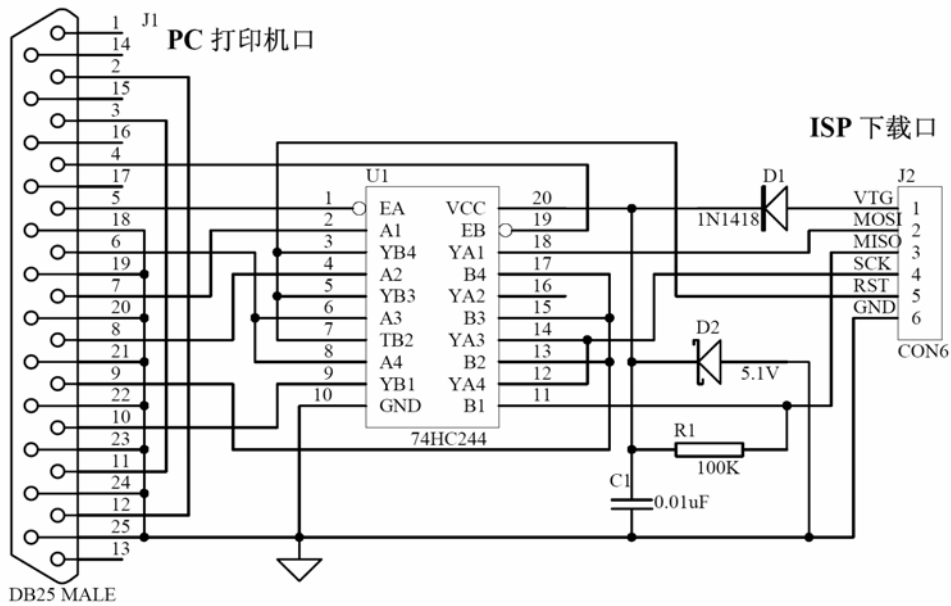


图 4-7 AVR ISP 下载电缆原理图

制作 AVR ISP 的成本非常低,采用贴片封装器件时整个电路板可以按装在一个普通 DB25 的接口盒中(图 4-8)。配备这样一根下载电缆,一台配备了相应开发软件的 PC 机,再加上开发目标系统板,一套基本的 AVR 软硬件开发环境就建立起来了。接下来需要的,就是发挥你的聪明才智和实际经验了!



图 4-8 AVR IS 下载电缆实物图

要使用 AVR ISP 编程电缆下载程序还需要相应的编程软件支持。本节介绍的 AVR ISP 下载电缆与 STK200/STK300 完全兼容,ICCAVR、CVAVR、BASC0M-AVR 等开发平台的程序下载单元中都直接支持使用 STK200/STK300 的下载编程(可惜,ATMEL 的 AVR STUDIO 不支持使用 STK200/STK300 下载电缆)。此外,也可以在网上下下载一些专用的免费编程软件使用。

(1) 在 ICCAVR、CVAVR、BASC0M-AVR 中的使用

在 ICCAVR、CVAVR、BASC0M-AVR 等 AVR 高级语言开发平台中,都有内置的程序编程下载

功能模块，支持使用多种不同形式的 ISP 下载电缆（编程器）实现对 AVR 的编程操作。STK200/STK300 是他们支持的下载电缆之一。在下载程序前，你只要在相应的 Programmer 的选项栏中选定使用 STK200/STK300，就可在程序正确编译后直接将运行代码下载到 AVR 芯片中了，非常方便。

ICCAVR、CVACR 的编程速度比 BASCOM-AVR 快一些。但在 BASCOM-AVR 中对 AVR 芯片熔丝位的编程配置界面更加体现了以人为本的原则（图 4-9），它提供简洁的说明选择，而其它编程软件对 AVR 芯片熔丝位的编程配置界面使用不方便，需要用户仔细查阅手册后作出选择。因此，建议用户对 AVR 芯片熔丝位进行编程配置时，使用 BASCOM-AVR 进行选择和操作。

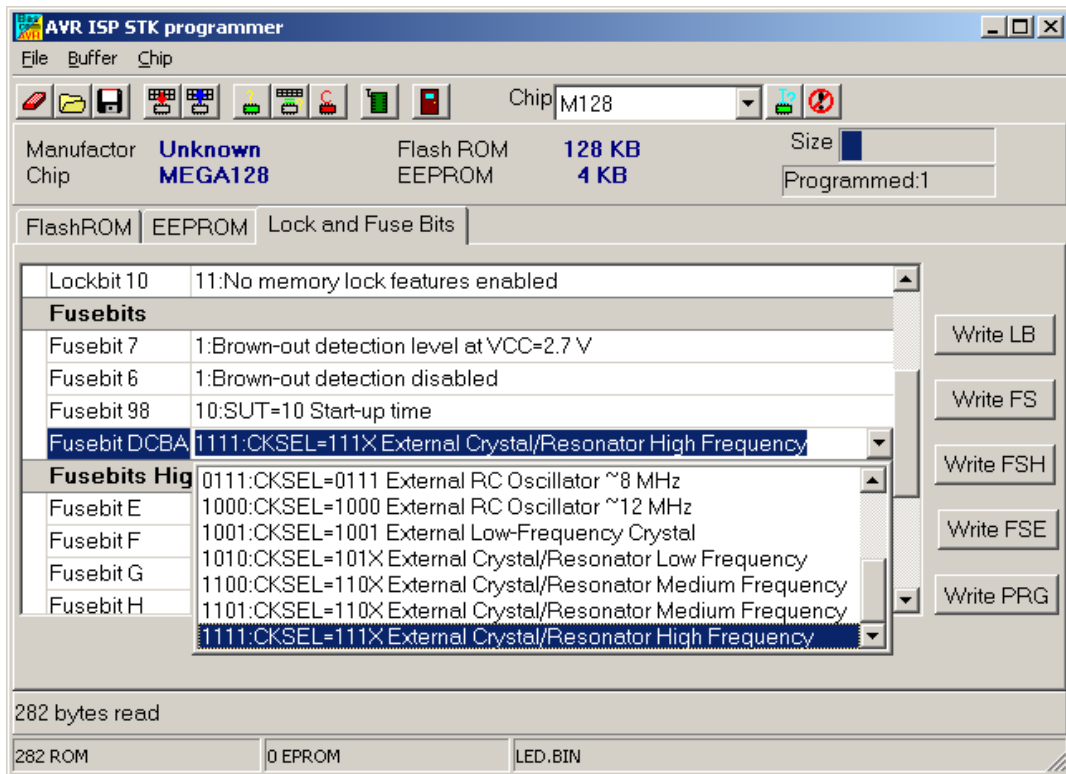


图 4-9 BASCOM-AVR 熔丝位编程配置窗口

(2) 免费通用编程软件 PonyProg。

PonyProg 是一个通用串行器件编程软件，配合的特定的编程硬件（电缆），它支持对 I²C 总线（24 系列），Microwire 总线（93 系列），SPI 总线（25 系列）的 EEPROM 存储器，ATMEL 的 AVR 单片机（支持使用 STK200/STK300 下载电缆），MICROCHIP 的 PIC 单片机的进行读写编程。

你可以在 <http://www.lancos.com/prog.html> 下载最新的 PonyProg。同时该网站上还提供其支持的编程器的硬件电路图，你可以参考，并制作一个多功能的编程器。

(3) 免费编程软件 SLISP

SLISP 是国内广州双龙公司推出的免费 ISP 编程软件，它也可以配合 STK200/STK300 实现对 AVR 的下载编程，同时也能对 AT89S5x 编程。该公司网站地址：<http://www.sl.com.cn>。

4.4 AVR 开发环境的建立

单片机嵌入式系统应用和开发的学习，不能仅仅依赖于从书本上学习原理和理论，更主要的学习环节是动手实践。单片机嵌入式系统的应用开发技术是一门实战（践）性很强的学

科，也是一门综合性的学科，最好的学习方法是边学习、边实践、边总结和归纳。

在学习了 AVR 单片机的基本结构和汇编指令以后，我们对 AVR 单片机已经有了一个理论上的认识和初步的了解。进一步的学习和使用 AVR 单片机就应该与实践相结合，动手实践是真正学好和掌握单片机与嵌入式系统设计和开发的必要途径。

4.3.1 AVR 单片机实验开发环境

一个 AVR 单片机的基本学习和开发环境由以下几部分组成：

1. PC 机一台，运行 Windows 操作系统

个人电脑 PC 是 AVR 嵌入式系统设计开发的主要工具之一。一般的讲，一台 586 以上，运行 Windows98/Windows2000/Windows XP 操作系统的个人电脑就可以了。

2. AVR 软件开发平台

一般需要选择一个或两个 AVR 软件开发平台。如采用汇编语言来开发 AVR 的系统程序，则首选 ATMEL 公司免费提供的 AVR Studio。喜欢和习惯采用高级语言开发系统程序，可以选取 C 或 BASIC 的开发平台。

3. AVR 实验开发板 (Developing Kit)

AVR 实验开发板是系统实现的硬件环境。在开发板上，除了具有可供使用的 AVR 芯片，还提供电源，基本的外围器件 (LED 发光管、LED 数码管、LCD 显示器、按键.....)，通信接口器件，通信连接线，用于执行代码程序下载的连接线等等。它方便用户实际动手学习、调试和检验自己的设计。因此，一块开发板对于学习是不可缺少的。

4. 其它辅助工具、设备和软件

在 AVR 嵌入式系统的开发过程中，一些必要的辅助工具和设备有：万用表，示波器，信号源，频率计等。工具软件有：串口调试软件，执行代码程序的下载编程软件等。

以上我们没有提到 AVR 的在线实时仿真器。由于 AVR 单片机具备 ISP 功能，以及大多数的 AVR 软件开发平台都与 AVR Studio 配合，使用 AVR Studio 中的软件模拟仿真功能，因此对于学习以及开发一些普通的系统，基本可以不必购买价格比较昂贵的 AVR 在线仿真器。当然，在开发一个比较复杂的系统时，手头配备一台 AVR 在线仿真器也是有必要的。

4.3.2 本书中采用的 AVR 单片机实验开发环境的建立

在本书中，我们推荐和使用以下的软硬件环境作为开始学习和使用 AVR 单片机的手段和工具，开发环境有以下部分构成：

- PC 机一台，运行 Windows 操作系统；
- AVR Studio 4.11(<http://www.atmel.com> , Free)。汇编开发，软件模拟调试。
- AVR高级C语言开发平台CVAVR (DEMO版) (<http://www.hpinfotech.ro> , Free)。C 语言开发。
- BASCOM-AVR (DEMO版) (<http://www.mcselec.com> , Free)。用于熔丝位的配置编程和程序下载。
- 串口调试精灵。用于 PC 机与单片机之间的 RS-232 通信，软件调试等。
- AVR-51 多功能实验开发板一套。目标实验板。

以上软件都是免费软件，用户可以网上下载，或从本书附带的光盘中获取。软件的安装

都比较简单，用户可以根据安装提示进行安装。

思考与练习

1. 学习单片机嵌入式系统的原理与应用开发，应具备和掌握哪些方面的基础知识和技能，为什么？
2. 为什么仅通过书本和课堂是不能学好和掌握单片机嵌入式系统的原理与应用开发的？
3. 简单讲述单片机嵌入式系统的开发过程和步骤，并说明在开发过程中要使用的主要硬件和软件工具是什么。
4. 硬件仿真器和程序烧入器的作用是什么？
5. 一个好的单片机软件开发平台应具备那些必要的功能？
6. 使用汇编语言和高级程序设计语言编写系统程序各有何优点和不足？
7. 通过网络、杂志与广告了解国内外主要的单片机生产商，它们的单片机产品型号，以及相应的开发系统和工具的名称和价格。
8. 本书推荐的学习 AVR 嵌入式系统开发的实验开发环境包含哪些硬件与软件？有何特点？
9. 建立一个学习 AVR 嵌入式系统开发的实验开发环境，熟悉 AVR-51 多功能实验版的硬件电路图与实际的连接，下载和安装软件开发系统和工具软件。
10. 仔细阅读各个软件的使用说明(Online-Help)，熟悉软件的使用环境和主要功能。
11. 在本书附带的光盘中有 AVR Studio、CVAVR 等软件使用参考的中文翻译电子文档，可以作为辅助参考资料。但最好阅读英文原文，可以得到更详细和准确的帮助。