单片机应用系统的开发工具

本项目要达到的目标包括 3 个方面: 一是让读者理解单片机与单片机应用系统的基本概念; 二是了解单片机应用系统的开发流程, 学会用 Keil C 集成开发环境输入、编辑、编译与调试用户程序; 三是学会用 STC-ISP 在线编程软件进行在线编程与在线仿真。此外, Proteus 仿真软件在单片机应用系统开发中很方便、很实用, 但非必需, 限于篇幅, 在此不作介绍。

知识点

- ◇ 微型计算机的基本结构与工作过程。
- ⇒ 单片机与单片机应用系统的基本概念。
- ◇ 单片机应用系统的开发流程。
- ♦ Keil C 集成开发环境的基本功能。
- ♦ STC-ISP 在线编程软件的基本功能。

技能点

- ◆ 应用 Keil C 集成开发环境输入、编辑、编译与调试单片机应用程序。
- ♦ 应用 STC-ISP 在线编程软件下载用户程序到单片机中。
- ◇ 应用 STC-ISP 在线编程软件进行在线仿真。

任务 1.1 单片机与单片机应用系统



从微型计算机的基本组成、工作原理与工作过程等相关知识,引出单片机的基本定义,建立起单片机应用系统的概念。通过单片机应用系统的演示让同学们体会单片机在电子系统中的控制作用,理解单片机在自动化、智能化电子产品中的核心地位,理解单片机在现代电子产品设计中的重要性与必要性。



1. 微型计算机的基本组成

图 1.1.1 所示为微型计算机的组成框图,由中央处理单元(CPU)、存储器(ROM、

RAM)和输入/输出接口(I/O 接口)和连接它们的总线组成。微型计算机配上相应的输入/输出设备(如 I/O 设备)就构成了微型计算机系统。

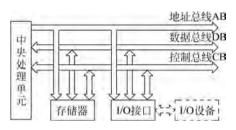


图 1.1.1 微型计算机组成框图

1) 中央处理单元

中央处理单元(CPU)由运算器和控制器两部分组成,是计算机的控制核心。

- (1)运算器。运算器由算术逻辑单元(ALU)、累加器和寄存器等几部分组成,主要负责数据的算术运算和逻辑运算。
- (2) 控制器。控制器由程序计数器、指令寄存器、指令译码器、时序发生器和操作控制器等组成,是计算机发布命令的"决策机构",即协调和指挥整个计算机系统的操作。

2) 存储器

通俗地讲,存储器是微型计算机的仓库,包括程序存储器和数据存储器两部分。程序存储器用于存储程序和一些固定不变的常数和表格数据,一般由只读存储器(ROM)组成;数据存储器用于存储运算中的输入/输出数据或中间变量数据,一般由随机存取存储器(RAM)组成。

3) I/O 接口

微型计算机的 I/O 设备(简称外设,如键盘、显示器等)有高速的也有低速的,有机电结构的也有全电子式的,由于种类繁多且速度各异,因而它们不能直接同高速工作的 CPU 相连。I/O 接口是 CPU 与 I/O 设备连接的桥梁,I/O 接口的作用相当于一个转换器,保证 CPU 与外设间协调地工作。不同的外设需要不同的 I/O 接口。

4) 总线

CPU 与存储器、I/O 接口是通过总线相连的,包括地址总线、数据总线与控制总线。

- (1) 地址总线。地址总线用作 CPU 寻址,地址总线的多少标志着 CPU 的最大寻址能力。若地址总线的根数为 16,即 CPU 的最大寻址能力为 2¹⁶B=64KB。
- (2) 数据总线。数据总线用于 CPU 与外围器件(存储器、I/O 接口)交换数据,数据总线的多少标志着 CPU 一次交换数据的能力,决定 CPU 的运算速度。通常所说的 CPU 位数就是指数据总线的位数。如 8 位机,就是该计算机的数据总线为 8 位。
- (3) 控制总线。控制总线用于确定 CPU 与外围器件交换数据的类型,从广义上讲就是"读"和"写"两种类型。

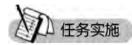
2. 微型计算机的工作过程

一个完整的计算机是由硬件和软件两部分组成的,缺一不可。上面所述为计算机的硬件部分,是看得到、摸得着的实体部分,但计算机硬件只有在软件的指挥下,才能发挥其效能。计算机采取"存储程序"的工作方式,即事先把程序加载到计算机的存储器中,当启动运

行后,计算机便自动地进行工作。

计算机执行程序是一条指令一条指令执行的。执行一条指令的过程分为3个阶段,即取指、指令译码与执行指令。每执行完一条指令,自动转向下一条指令的执行。

- (1) 取指。根据程序计数器中的地址,到程序存储器中取出指令代码,并送到指令寄存器中。
- (2) 指令译码。指令译码器对指令寄存器中的指令代码进行译码,判断出当前指令代码的工作任务。
- (3) 执行指令。判断出当前指令代码任务后,控制器自动发出一系列微指令,指挥计算机协调地动作,完成当前指令指定的工作任务。



1. 单片机的概念

将微型计算机的基本组成部分(CPU、存储器、I/O接口以及连接它们的总线)集成在一块芯片中而构成的计算机,称为单片机。

由于单片机是完全作嵌入式应用,故又称为嵌入式微控制器。根据单片机数据总线的宽度不同,单片机主要可分为 4 位机、8 位机、16 位机和 32 位机。在高端应用(图形图像处理与通信等)中,32 位机应用已越来越普及;但在中、低端控制应用中,在将来较长一段时间内,8 位单片机仍是单片机的主流机种,近期推出的增强型 8051 单片机产品内部集成有高速 I/O 接口、ADC、DAC、PWM、WDT等接口部件,并在低电压、低功耗、串行扩展总线、程序存储器类型、存储器容量和开发方式(在线系统编程 ISP)等方面都有较大的发展。

单片机自身仅是一个只能处理数字信号的装置,必须配置好相应的外围接口器件或执行器件,才是一个能完成具体任务的工作系统,称为单片机应用系统。

2. 单片机应用系统的演示与体验

演示与体验: 计算机时钟(在 STC 大学计划实验箱(9.3)上,采用项目 9 中的任务 9.3 程序进行演示与体验)。

3. 单片机的应用与发展趋势

1) 单片机的应用领域

由于单片机具有较高的性能价格比、良好的控制性能和灵活的嵌入特性,单片机在各个领域里都获得了极为广泛的应用。

- (1)智能仪器仪表。单片机用于各种仪器仪表,一方面提高了仪器仪表的使用功能和精度,使仪器仪表智能化,同时简化了仪器仪表的硬件结构,从而可以方便地完成仪器仪表产品的升级换代,如各种智能电气测量仪表、智能传感器等。
- (2) 机电一体化产品。机电一体化产品是集机械技术、微电子技术、自动化技术和计算机技术于一体,具有智能化特征的各种机电产品。单片机在机电一体化产品的开发中可以发挥巨大的作用,典型产品有机器人、数控机床、自动包装机、点钞机、医疗设备、打印机、传真机、复印机等。
 - (3) 实时工业控制。单片机还可用于各种物理量的现场采集与控制。电流、电压、温

度、液位、流量等物理参数的采集和控制均可以用单片机方便地实现。在这类系统中,采用单片机作为系统控制器,可以根据被控对象的不同特征采用不同的智能算法,实现期望的控制指标,从而提高生产效率和产品质量,如电动机转速控制、温变控制与自动生产线等。

- (4)分布系统的前端模块。在较复杂的工业系统中,经常要采用分布式测控系统完成 大量的分布参数的采集。在这类系统中,采用单片机作为分布式系统的前端采集模块,系统 具有运行可靠、数据采集方便灵活、成本低廉等一系列优点。
- (5) 家用电器。家用电器是单片机的又一重要应用领域,前景十分广阔,如空调器、电冰箱、洗衣机、电饭煲、高档洗浴设备、高档玩具等。
- (6) 交通工具。在交通领域中,汽车、火车、飞机、航天器等均有单片机的广泛应用,如 汽车自动驾驶系统、航天测控系统、黑匣子等。

2) 单片机的发展趋势

1970 年微型计算机研制成功之后,随着大规模集成电路的发展又出现了单片机,并且按照不同的发展要求,形成了系统机与单片机两个独立发展的分支。美国 Intel 公司 1971 年生产的 4 位单片机 4004 和 1972 年生产的雏形 8 位单片机 8008,特别是 1976 年 MCS-48 单片机自问世以来,其发展速度为每 2~3 年要更新一代,集成度增加 1 倍,功能翻一番。发展速度之快、应用范围之广,已达到惊人的地步。它已渗透到生产和生活的诸多领域,可谓"无孔不人"。

综观 50 多年的发展过程,单片机正朝着多功能、多选择、高速度、低功耗、低价格、扩大存储容量和加强 I/O 功能及结构兼容方向发展。预计今后的发展趋势会体现在以下几个方面。

- (1) 多功能。在单片机中尽可能多地把应用系统中所需要的存储器、各种功能的 I/O 接口都集成在一块芯片内,即外围器件内装化,如把 LED、LCD 或 VFD 显示驱动器集成在单片机中。
- (2) 高性能。为了提高速度和执行效率,在单片机中开始使用 RISC 体系结构、并行流 水线操作和 DSP 等的设计技术,使单片机的指令运行速度得到大大提高,其电磁兼容等性 能明显优于同类型的微处理器。
- (3) 产品系列化。对单片机的应用情况进行评价,根据应用系统对 I/O 接口要求分层次配置,形成单片机产品系列化,单片机应用者在进行单片机应用系统开发时总能选择到既可满足系统功能要求又不浪费的单片机,以提高开发产品的性能价格比。
- (4)推行串行扩展总线。推行串行扩展总线可以显著减少引脚数量,简化系统结构。随着外围器件串行接口的发展,单片机的串行接口的普遍化、高速化使并行扩展接口技术日渐衰退。从而许多公司都推出去掉了并行总线的非总线单片机,需要外扩器件(存储器、I/O接口等)时,采用串行扩展总线,甚至用软件模拟串行总线来实现。

4. 单片机市场情况

单片机市场主要以8位机和32位机(ARM)为主,通常所说的单片机是指8位机,而32位机一般称为ARM。

1) MCS-51 系列单片机与 51 兼容机

MCS-51 系列单片机是美国 Intel 公司研发的,但 Intel 公司后期的重点并不在单片机上,因此市场上很难见到 Intel 公司生产的单片机。市场上的 51 单片机更多的是以 MCS-51 系列单片机为核心和框架的兼容 51 单片机,主要生产厂家有美国 Atmel 公司、荷兰菲利

普公司和我国 STC。本书以本土卓越的高端 8051 单片机——STC8H8K64U 系列单片机 为学习机型。

2) PIC 系列单片机

Microchip 单片机是市场份额增长较快的单片机。它的主要产品是 16C 系列 8 位单片机,CPU 采用 RISC 结构,仅 33 条指令,运行速度快,且以低价位著称,单片机价格都在 1 美元以下。Microchip 单片机没有掩膜产品,全部是 OTP 器件,Microchip 强调节约成本的最优化设计,适于用量大、档次低、价格敏感的产品。

目前, Microchip 为全球超过 65 个国家或地区的 5 万多客户提供服务。大部分芯片有 其兼容的 Flash 程序存储器芯片, 支持低电压擦写, 擦写速度快, 而且允许多次擦写, 程序修 改方便。

3) AVR 单片机

1997 年由 Atmel 公司挪威设计中心的 A 先生与 V 先生利用 Atmel 公司的 Flash 新技术,共同研发出 RISC 精简指令集的高速 8 位单片机,简称 AVR。AVR 单片机的推出彻底打破了这种旧设计格局,废除了机器周期,抛弃复杂指令计算机(CISC)追求指令完备的做法,采用精简指令集,以字作为指令长度单位,将内容丰富的操作数与操作码安排在一字之中,取指周期短,又可预取指令,实现流水作业,故可高速执行指令。

它具有增强性的高速同步/异步串口,具有硬件产生校验码、硬件检测和校验侦错、两级接收缓冲、波特率自动调整定位(接收时)、屏蔽数据帧等功能,提高了通信的可靠性,方便程序编写,更便于组成分布式网络和实现多机通信系统的复杂应用。AVR单片机博采众长,又具独特技术,成为8位机中的佼佼者。

任务 1.2 单片机应用程序的输入、编辑、编译与调试



单片机应用系统由硬件和软件两部分组成,单片机应用系统的开发包括硬件设计与软件设计。单片机自身只能识别机器代码,但人们为了便于记忆、识别和编写应用程序,一般采用汇编语言或 C 语言编程,为此就需要一个工具能将汇编语言源程序或 C 语言源程序转换成机器代码程序,Keil C 集成开发环境就是一个融汇编语言和 C 语言编辑、编译与调试于一体的开发工具,目前流行的 Keil C 集成开发环境版本主要有 Keil μ Vision4 和 Keil Vision5。

本任务以程序实例为媒介系统地学习与实践 Keil μ Vision4 完成用户程序的输入、编辑、编译与模拟仿真调试。



1. 单片机应用程序的编辑、编译与调试流程

单片机应用程序的编辑、编译一般都采用 Keil C 集成开发环境实现,但程序的调试有

多种方法,如 Keil C 集成开发环境的软件仿真调试与硬件仿真调试、硬件的在线调试与专用仿真软件(Proteus)的仿真调试,如图 1.2.1 所示。

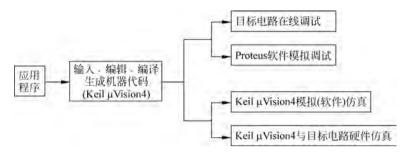


图 1,2.1 应用程序的编辑、编译与调试流程

2. Keil C 集成开发环境

1) Keil uVision4 的编辑、编译界面

Keil μ Vision4 集成开发环境从工作特性来分,可分为编辑、编译界面和调试界面,启动 Keil μ Vision4 后,进入编辑、编译界面,如图 1.2.2 所示。在此用户环境下可创建、打开用户项目文件,以及进行汇编源程序或 C51 源程序的输入、编辑与编译。

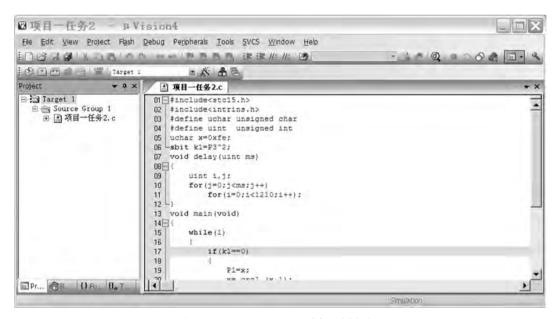


图 1.2.2 Keil μVision4 编辑、编译界面

- (1) 菜单栏。 $Keil \mu Vision4$ 在编辑、编译界面和调试界面的菜单栏是不一样,呈灰色显示的为当前界面无效菜单项。
- ① File 菜单。File(文件)菜单命令主要用于对文件的常规(新建文件、打开文件、关闭文件与文件存盘等)操作,其功能、使用方法与一般的 Word、Excel 等应用程序一致。但文件菜单的 Device Database 命令是特有的,Device Database 用于修改 Keil μVision4 支持的8051 芯片型号以及 ARM 芯片的设定。Device Database 对话框如图 1.2.3 所示,用户可在

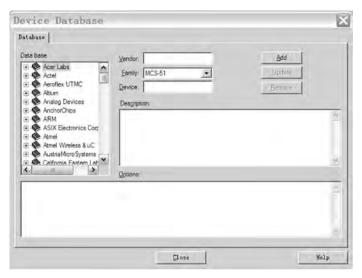


图 1.2.3 Device Database 对话框

该对话框中添加或修改 Keil μVision4 支持的单片机型号及 ARM 芯片。

Device Database 对话框各个选项功能如下。

Data base 列表框:浏览 Keil μVision4 支持的单片机型号及 ARM 芯片。

Vendor 文本框:用于设定单片机的类别。

Family 下拉列表框: 用于选择 MCS-51 单片机家族以及其他微控制器家族,有 MCS-51、MCS-251、80C166/167、ARM。

Device 文本框:用于设定单片机的型号。

Description 列表框:用于设定型号的功能描述。

Options 列表框:用于输入支持型号对应的 DLL 文件等信息。

Add 按钮:单击 Add 按钮添加新的支持型号。

Update 按钮:单击 Update 按钮确认当前修改。

- ② Edit 菜单。Edit(编辑)菜单主要包括剪切、复制、粘贴、查找、替换等通用编辑操作。此外,本软件还有 Bookmark(书签管理命令)、Find(查找)及 Configuration(配置)等操作功能。其中,Configuration(配置)命令用于设置软件的工作界面参数,如编辑文件的字体大小及颜色等参数。Configuration(配置)操作对话框如图 1.2.4 所示,包括 Editor(编辑)、Colors & Fonts(颜色与字体)、User Keywords(设置用户关键词)、Shortcut Keys(快捷关键词)、Templates(模板)、Other(其他)等配置选项卡。
- ③ View 菜单。View(视图)菜单用于控制 Keil μ Vision4 界面显示,使用 View 菜单中的命令可以显示或隐藏 Keil μ Vision4 的各个窗口和工具栏等。在编辑/编译工作界面、调试界面有不同的工具栏和显示窗口。
- ④ Project 菜单。Project(项目)菜单命令包括项目的建立、打开、关闭、维护、目标环境设定、编译等。Project 菜单各个命令功能介绍如下。

New Project: 建立一个新项目。

New Multi-Project Workspace: 新建多项目工作区域。



图 1.2.4 Configuration(配置)操作对话框

Open Project: 打开一个已存在的项目。

Close Project: 关闭当前项目。

Export: 导出为 μVision3 格式。

Manage: 工具链、头文件和库文件的路径管理。

Select Device for Target: 为目标选择器件。

Remove Item: 从项目中移除文件或文件组。

Options: 修改目标、组或文件的选项设置。

Build Target: 编译修改过的文件并生成应用程序。

Rebuild Target: 重新编译所有文件并生成应用程序。

Translate: 传输当前文件。

Stop Build: 停止编译。

- ⑤ Flash 菜单。Flash(下载)菜单主要用于程序下载到 EEPROM 的控制。
- ⑥ Debug 菜单。Debug(调试)菜单中命令用于软件仿真环境下的调试,提供断点、单步、跟踪与全速运行等操作命令。
- ⑦ Peripherals 菜单。Peripherals(外设)菜单是外围模块菜单命令,用于芯片的复位和片内功能模块的控制。
- ⑧ Tools 菜单。Tools (工具)菜单主要用于支持第三方调试系统,包括 Gimpel Software 公司的 PC-Lint 和西门子公司的 Easy-Case。
- ⑨ SVCS 菜单。SVCS(软件版本控制系统)菜单命令用于设置和运行软件版本控制系统(software version control system, SVCS)。
- ⑩ Window 菜单。Window(窗口)菜单命令用于设置窗口的排列方式,与 Window 的窗口管理兼容。
 - ① Help菜单。Help(帮助)菜单命令用于提供软件帮助信息和版本说明。
 - (2) 工具栏。Keil μVision4 在编辑、编译界面和调试界面有不同的工具栏,在此介绍编

辑、编译界面的工具栏。

① 常用工具栏。图 1. 2. 5 所示为 Keil µVision4 的常用工具栏,从左至右依次为 New (新建文件)、Open(打开文件)、Save(保存当前文件)、Save All(保存全部文件)、Cut(剪切)、Copy(复制)、Paste(粘贴)、Undo(取消上一步操作)、Redo(恢复上一步操作)、Navigate Backwards(回到先前的位置)、Navigate Forwards(前进到下一个位置)、Insert/Remove Bookmark(插入或删除书签)、Go to Previous Bookmark(转到前一个已定义书签处)、Go to the Next Bookmark(转到下一个已定义书签处)、Clear All Bookmarks(取消所有已定义的书签)、Indent Selection(右移一个制表符)、Unindent Selection(左移一个制表符)、Comment Selection(选定文本行内容)、Uncomment Selection(取消选定文本行内容)、Find in Files...(查找文件)、Find...(查找内容)、Incremental Find(增量查找)、Start/Stop Debug Session(启动或停止调试)、Insert/Remove Breakpoint(插入或删除断点)、Enable/Disable Breakpoint(允许或禁止断点)、Disable All Breakpoint(禁止所有断点)、Kill All Breakpoint(删除所有断点)、Project Windows(窗口切换)、Configuration(参数配置)等工具图标。单击工具图标,执行图标对应的功能。

图 1.2.5 常用工具栏

② 编译工具栏。图 1. 2. 6 所示为 Keil μVision4 的编译工具栏,从左至右依次为 Translate(传输当前文件)、Build(编译目标文件)、Rebuild(编译所有目标文件)、Batch Build(批编译)、Stop Build(停止编译)、Down Load(下载文件到 Flash ROM)、Select Target (选择目标)、Target Option(目标环境设置)、File Extensions、Books and Environment(文件的组成、记录与环境)、Manage Multi-Project Workspace(管理多项目工作区域)等工具图标。单击某图标,可执行图标对应的功能。



- (3) 窗口。Keil μ Vision4的窗口在编辑、编译界面和调试界面有不同的窗口,在此介绍编辑、编译界面的窗口。
- ① 编辑窗口。在编辑窗口中用户可以输入或修改源程序,Keil μ Vision4 的编辑器支持程序行自动对齐和语法高亮显示。
- ② 项目窗口。选择菜单命令 View → Project Window 或单击工具栏中的相应图标可以显示或隐藏项目窗口。该窗口主要用于显示当前项目的文件结构和寄存器状态等信息。项目窗口中共有 4 个选项页,即 Project、Books、Functions、Templates。 Files 选项页显示当前项目的组织结构,可以在该窗口中直接单击文件名打开文件,如图 1.2.7 所示。



图 1.2.7 项目窗口中的 Project 选项页

10

③ 输出窗口。Keil μ Vision4 的编译信息输出窗口用于显示编译时的输出信息,如图 1.2.8 所示。在窗口中,双击输出的 Warning 或 Error 信息,可以直接跳转至源程序的警告或错误所在行。

```
Build target 'Simulator'
compiling HELLO.C...
linking...
Program Size: data=30.1 xdata=0 code=1096
"HELLO" - 0 Error(s), 0 Warning(s).
```

图 1.2.8 Keil μVision4 的编译信息输出窗口

2) Keil μVision4 的调试界面

Keil μ Vision4 集成开发环境除了可以编辑 C 语言源程序和汇编语言源程序外,还可以软件模拟调试和硬件仿真调试用户程序,以验证用户程序的正确性。在模拟调试中主要学习两个方面的内容: 一是程序的运行方式; 二是如何查看与设置单片机内部资源的状态。

选择菜单命令 Debug→Start/Stop Debug Session 或单击工具栏中的调试按钮 ②,系统进入调试界面,如图 1.2.9 所示。若再次单击调试按钮 ②,则退出调试界面。

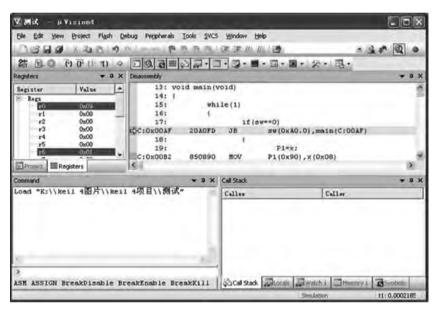


图 1.2.9 Keil μVision4 的调试界面

(1) 程序的运行方式。

图 1. 2. 10 所示为 Keil μVision4 的运行工具栏,从左至右依 次为 Reset(程序复位)、Run(程序全速运行)、Stop(程序停止运行)、Step(跟踪运行)、Step Over(单步运行)、Step Out(执行跟踪 图 1. 2. 10 程序运行工具栏 并跳出当前函数)、Run to Cursor Line(运行至光标处)等工具图标。单击工具图标,执行图标对应的功能。

☎(程序复位):使单片机的状态恢复到初始状态。