

电气与控制实验科学技术

王科俊 主编

嵌入式系统开发实验技术

QIAN RU SHI XI TONG KAI FA SHI YAN JI SHU

管凤旭 张爱筠 编著

黑龙江人民出版社

· 电气与控制实验科学技术 · 王科俊 主编

江南大学图书馆



91142444

嵌入式系统开发实验技术

管凤旭 张爱筠 编著



黑龙江人民出版社

图书在版编目(CIP)数据

电气与控制实验科学技术/王科俊 主编. —哈尔滨:黑龙江人民出版社,2008.7

ISBN 978-7-207-07885-8

I. 电… II. 王… III. 电气控制—实验—研究 IV. TM921.5

中国版本图书馆 CIP 数据字(2008)第 110906 号

责任编辑:李荣焕

装帧设计:李若萌

嵌入式系统开发实验技术

管凤旭 张爱筠 编著

出版发行 黑龙江人民出版社

通讯地址 哈尔滨市南岗区宣庆小区 1 号楼

邮 编 150008

网 址 www.longpress.com

电子邮箱 hljrmcbs@yeah.net

印 刷 哈尔滨太平洋彩印有限公司

开 本 16 开

印 张 135 印张

字 数 355 万字

印 数 2000 套

版 次 2008 年 8 月第 1 版 2008 年 8 月第 1 次印刷

书 号 ISBN 978-7-207-07885-8/TP·14

定 价 350.00 元(全套 14 本)

(如发现本书有印制质量问题,印刷厂负责调换)

本社常年法律顾问:北京市大成律师事务所哈尔滨分所律师赵学利、赵景波

前言

在日益信息化的现代社会中,嵌入式系统在电子、通信、交通、金融、网络、监控、自动化等领域应用广泛,各种各样的新型嵌入式系统设备不断出现。任何一个普通人可能拥有从小到大的各种使用嵌入式技术的电子产品,小到 MP3、手机等数字化产品,大到智能家电、车载组合电子设备等。在工业和服务领域中,使用嵌入式技术的数字机床、智能工具、工业机器人、服务机器人也将逐渐改变传统的工业和服务方式。为此,各高校的电气自动化、机电一体化、计算机等相关专业相继开设了各种关于嵌入式系统原理及应用的课程。嵌入式系统课程是一门实践性很强的课程,不仅要熟悉和理解基础知识,还要通过实践逐步掌握开发嵌入式系统的硬件和软件设计方法,进而巩固和深化嵌入式系统的基础知识,以及硬件和软件设计知识。

本书以 ARM7 S3C44B0X 和 DSP F2812 的各种实验为主,按照深入浅出,循序渐进的方式介绍给读者。第 1 章介绍嵌入式系统的定义、组成、主要特征等;第 2 章分别介绍两种实验教学系统和基本使用说明;第 3 章介绍嵌入式系统软件基础实验;第 4 章介绍嵌入式系统资源实验;第 5 章介绍嵌入式系统综合与创新实验;第 6 章介绍嵌入式操作系统实验。在后面两章中给出具有工程背景的温度控制系统等实例,不仅提供参考方案,而且还给出了参考方案的硬件设计原理图、控制算法以及软件源代码。本书对嵌入式系统的学习和实际工程设计都有一定的参考价值。

本书由哈尔滨工程大学两位教师合作编写。书中第 1 章、第 2.1~2.2 节、第 3.1~3.3 节、第 4.1~4.14 节、第 5.1~5.2 节、第 6 章由管凤旭编写;第 2.3~2.4 节、第 3.4~3.7 节、第 4.15~4.27 节、第 5.3~5.4 节由张爱筠编写。本书的编写还得到哈尔滨工程大学教师王科俊与姜倩、学生段旭良与张国宇、哈尔滨理工大学软件学院教师董怀国与高俊锋、哈尔滨中云科技发展有限公司何云斌、深圳市英蓓特信息技术有限公司和北京瑞泰创新科技有限责任公司等的大力帮助,在此,笔者表示深切的感谢。

书中可能会存在疏漏、不妥和错误之处,恳请广大读者指正。

作者

2008 年 2 月

内 容 简 介

本书以 ARM7 S3C44B0X 和 DSP F2812 的各种实验为主,按照深入浅出,循序渐进的方式介绍给读者。本书分为实验教学系统和使用说明部分,嵌入式系统软件基础实验部分,嵌入式系统资源实验部分,嵌入式系统综合与创新实验部分,嵌入式操作系统实验部分。在后两部分中给出具有工程背景的温度控制系统等实例,不仅提供参考方案,而且还给出参考方案的硬件设计原理图、控制算法以及软件源代码,对嵌入式系统的学习和实际工程设计都有一定的参考价值。

本书适用于高等院校电气自动化与控制、机电一体化、计算机、电子信息等相关专业的高年级本科生和研究生使用,也可供从事嵌入式系统应用开发的工程技术人员参考。

目 录

第1章 嵌入式系统综述	(1)
第2章 嵌入式开发实验教学系统	(4)
2.1 ARM 实验教学系统简介	(4)
2.2 ARM 实验教学系统使用说明	(5)
2.3 DSP 实验教学系统简介	(14)
2.4 DSP 实验教学系统使用说明	(17)
第3章 软件基础实验	(22)
3.1 ARM 汇编语言	(22)
3.2 ARM C 语言程序	(24)
3.3 ARM C 语言和汇编语言混合程序	(26)
3.4 DSP 软件入门	(28)
3.5 DSP 汇编语言	(34)
3.6 DSP C 语言程序	(38)
3.7 DSP C 语言和汇编语言混合程序	(43)
第4章 系统资源实验	(47)
4.1 ARM 存储器	(47)
4.2 ARM I/O 接口	(49)
4.3 ARM 中断	(50)
4.4 ARM 串口通信	(54)
4.5 ARM 实时时钟	(57)
4.6 ARM 键盘与数码管显示	(59)
4.7 ARM 看门狗控制	(61)
4.8 ARM 液晶显示控制	(64)
4.9 ARM 触摸屏控制	(67)
4.10 ARM IIC 串行通信	(70)
4.11 ARM A/D 转换	(74)
4.12 ARM PWM 电机控制	(77)
4.13 ARM IIS 音频接口	(83)
4.14 ARM USB 接口	(87)
4.15 DSP 存储器	(91)
4.16 DSP 指示灯及灯阵列	(93)
4.17 DSP 外中断	(95)
4.18 DSP 定时器	(97)
4.19 DSP 单路及多路模数转换(A/D)	(99)
4.20 DSP 异步串口通信	(102)

4.21	DSP PWM 输出	(103)
4.22	DSP 液晶显示器控制显示	(105)
4.23	DSP 键盘输入	(107)
4.24	DSP 直流电机控制	(109)
4.25	DSP 步进电机控制	(112)
4.26	DSP 有限冲击响应滤波器(FIR)	(114)
4.27	DSP 快速傅立叶变换(FFT)算法	(117)
第5章	综合与创新实验	(120)
5.1	ARM 温度控制系统	(120)
5.2	ARM 万年历	(125)
5.3	DSP 交通灯综合控制	(126)
5.4	DSP 温度控制系统	(129)
第6章	μC/OS-II 操作系统实验	(132)
6.1	ARM μ C/OS-II 移植	(132)
6.2	ARM μ C/OS-II 操作系统的温度控制系统	(135)
	参考文献	(137)

第1章 嵌入式系统综述

随着大规模集成电路的电子技术和软件技术的迅速发展,逐渐形成一个与通用计算机系统并列的嵌入式系统(Embedded Systems)。目前嵌入式系统已经渗透到日常生活的各个方面,包括消费电子、通信、汽车、国防、航空航天、工业控制、仪表、办公自动化等领域。例如:手机、电子字典、高清电视、数码相机、可视电话、机顶盒、智能玩具、路由器、雷达、坦克、战舰、飞机、航天器等,一辆高档轿车拥有几十个、甚至上百个嵌入式微处理器。可以说“嵌入式系统无处不在”。

1. 嵌入式系统定义

根据 IEEE(国际电机工程师协会)的定义,嵌入式系统是“控制、监视或者辅助装置、机器和设备运行的装置”。从中可以看出嵌入式系统是软件和硬件的综合体,还可以涵盖机械等附属装置。

目前国内普遍认同的一个定义是:以应用为中心、以计算机技术为基础、软件硬件可裁剪、适应应用系统,对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。

2. 嵌入式系统组成

嵌入式系统一般由嵌入式硬件和软件组成(图 1-1)。硬件以微处理器为核心集成存储器和系统专用的输入/输出设备。软件包括初始化代码及驱动、嵌入式操作系统和应用程序等,这些软件有机地结合在一起,形成系统特定的一体化软件。

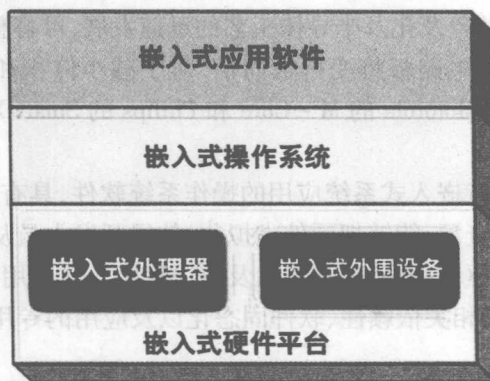


图 1-1 嵌入式系统组成

(1) 嵌入式处理器

嵌入式系统的核心部件是各种类型的嵌入式处理器,目前据不完全统计,全世界嵌入式处理器的品种总量已经超过 1000 多种,流行体系结构有 30 几个系列。根据其现状,嵌入式计算机可以分成下面几类。

① 微处理器

微处理器(MPU: Microprocessor Unit)的基础是通用计算机中的 CPU。在应用中,将微处理器装配在专门设计的电路板上,只保留和嵌入式应用有关的母板功能,这样可以大幅度减小系统体积和功耗。为了满足嵌入式应用的特殊要求,嵌入式微处理器虽然在功能上和标准微处理器基本是一样的,但在工作温度、抗电磁干扰、可靠性等方面一般都做了各种增强。

和工业控制计算机相比,嵌入式微处理器具有体积小、重量轻、成本低、可靠性高的优点,但是在电路板上必须包括 ROM、RAM、总线接口、各种外设等器件,从而在一定程度上降低了系统的可靠性,技术保密性也较差。嵌入式微处理器及其存储器、总线、外设等安装在一块电路板上,称为单板计算机。

嵌入式处理器目前主要有 Am186/88、386EX、SC-400、Power PC、68000、MIPS、ARM 系列等。

②微控制器

微控制器(MCU: Microcontroller Unit)又称单片机。早期比较流行的处理器,将整个计算机系统集成为一个芯片中,内部以某种微处理器为核心,根据需求有选择的集成 ROM、RAM、总线、总线逻辑、定时器/计数器、看门狗、I/O、串行口、A/D 转换、D/A 转换等功能和外设。和嵌入式微处理器相比,微控制器的最大特点是单片化,体积大大减小,从而使功耗和成本下降、可靠性提高。微控制器是目前嵌入式系统工业的主流。微控制器的片上外设资源一般比较丰富,适合于控制,因此称微控制器。

嵌入式微控制器目前的品种和数量最多,比较有代表性的通用系列包括 8051、P51XA、MCS-251、MCS-96/196/296、C166/167、MC68HC05/11/12/16、C540 等。

③ 数字信号处理器

数字信号处理器(DSP: Digital Signal Processor)对系统结构和指令进行特殊设计,使其适合于执行数字滤波、FFT、谱分析等 DSP 算法,编译效率较高,指令执行速度也快。

嵌入式 DSP 处理器比较有代表性的产品是 Texas Instruments 的 TMS320 系列和 Motorola 的 DSP 56000 系列。TMS320 系列处理器包括用于控制的 C2000 系列,移动通信的 C5000 系列,以及性能更高的 C6000 和 C8000 系列。DSP 56000 目前已经发展成为 DSP56000、DSP56100 等几个不同系列的处理器。

④ 片上系统

片上系统(SoC: System on Chip)将重要处理器的内核和各种外围的芯片器件整合在一起,进一步降低功耗。随着电子数据交换(EDI: Electric Data Interchange)的推广和超大规模集成电路(VLSI: Very Large Scale Integration)设计的普及化及半导体工艺的迅速发展,可将整个嵌入式系统大部分集成到一块或几块芯片中去,应用系统电路板将变得很简洁,对于减小体积和功耗、提高可靠性非常有利。代表产品有 Infineon 的 TriCore、Motorola 的 M-Core 和 Philips 的 Smart XA 等。

(2)嵌入式操作系统

嵌入式操作系统是一种支持嵌入式系统应用的操作系统软件,具有通用操作系统的基本特点,能够有效管理越来越复杂的系统资源;能够把硬件虚拟化,使得开发人员从繁忙的驱动程序移植和维护中解脱出来;能够提供库函数、驱动程序、工具集以及应用程序。与通用操作系统相比较,嵌入式操作系统在系统实时高效性、硬件的相关依赖性、软件固态化以及应用的专用性等方面具有较为突出的特点。

从应用角度可分为通用型嵌入式操作系统和专用型嵌入式操作系统。常见的通用型嵌入式操作系统有 Linux、VxWorks、Windows CE、 μ C/OS-II 等。常用的专用型嵌入式操作系统有 Smart Phone、Pocket PC、Symbian 等。

按实时性可分为两类。实时嵌入式操作系统主要面向控制、通信等领域。如 WindRiver 公司的 VxWorks、ISI 的 pSOS、QNX 系统软件公司的 QNX、ATI 的 Nucleus 等。非实时嵌入式操作系统主要面向消费类电子产品。这类产品包括 PDA、移动电话、机顶盒、电子书、WebPhone 等,如微软面向手机应用的 Smart Phone 操作系统。

3. 嵌入式系统主要特征

嵌入式系统与通用型计算机系统相比,具有以下特点:

(1)嵌入式系统通常是形式多样且面向特定应用。嵌入式系统是将先进的计算机技术、半导体技术以及电子技术与各个行业的具体应用相结合的产物,这一点就决定了它必然是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。没有哪个公司的操作系统和处理器能够垄断市场。

(2)嵌入式系统都具有功耗低、体积小、集成度高特点。嵌入式系统把通用计算机许多由板卡完成的任务集成在芯片内部,从而有利于嵌入式系统设计趋于小型化与低功耗,使其移动能力大大增强,和网络的耦合也越来越紧密。

(3) 嵌入式系统通常极其关注成本。根据具体功能尽可能量体裁衣、去除冗余,力争在同样的硅片面积上实现更高的性能,并降低成本,这样才能在具体应用中更具有竞争力。

(4) 嵌入式系统有实时性和可靠性的要求。由于大部分嵌入式系统必须具有较高的实时性,因此对程序的质量,特别是可靠性,有着较高的要求。为了提高执行速度和系统可靠性,嵌入式系统中的软件一般都固化在存储器芯片或单片机本身中,而不是存贮于磁盘等载体中。而且操作系统一般是适应多种处理器、可剪裁、轻量型、实时可靠、可固化的嵌入式操作系统。

(5) 嵌入式系统开发需要专门工具和特殊方法。嵌入式系统本身不具备自举开发的能力,即使设计完成以后用户通常也是不能对其中的程序功能进行修改的,必须有一套开发工具和环境才能进行开发。

4. 嵌入式系统开发学习

对于电气工程与控制等相关专业的学生,在完成各种电气与控制专业基本原理,微型计算机与单片机等基础控制器学习后,需要学习更高级、功能更强的控制器系统——高端嵌入式系统,以适应现代日新月异科技发展的需求,才能在电气与控制等相关专业领域的研发中立于不败之地。

学习嵌入式系统要有一定基础,熟悉微机原理和接口知识,熟悉一门汇编语言,尤其要弄清楚处理器执行的过程和常用指令的执行操作;至少会 C 语言编程。从嵌入式基础学起时,最好要先从简单的 51 单片机开发开始,因为 51 单片机便宜而且资料非常多。运用简单汇编语言和 C51 进行编程,在这个过程要熟悉开发流程和掌握访问硬件的方法。然后针对不带操作系统的高端嵌入式处理器进行学习,在这个过程要根据学习对象,研究其体系结构,学会查看硬件手册。最后,在积累了较多的开发经验后再学习嵌入式操作系统,一旦加入操作系统,编程难度大大增加,思维和编程方法转变比较艰难。通过学习操作系统的原理掌握进程、调度、同步和互斥、优先级翻转等知识后,才能使嵌入式系统的学习与开发进入一个全新的层次。在这个过程要多做几个有具体目标的项目,在学习和研发中摸索,要知道编写程序容易,但要编写稳定可靠的程序绝非易事。以上说得很简单,实际操作起来则需要付出很多努力和辛苦,需要学习很多知识,积累大量经验。这里只是给出学习的大概步骤,给初学者一个整体概念。



图 2-1-2 ARM 嵌入式系统硬件平台示意图

第2章 嵌入式开发实验教学系统

2.1 ARM 实验教学系统简介

ARM 实验教学系统(图 2-1-1)采用深圳英蓓特公司实验系统,包括 Embest IDE 集成开发环境,Embest EduKit-III 实验台,Embest JTAG 仿真器和 Flash 编程器。

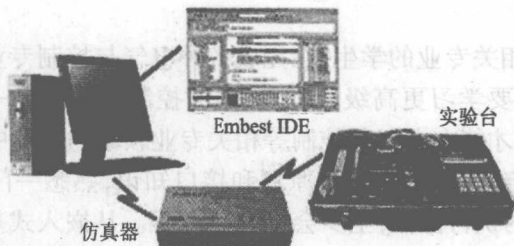


图 2-1-1 ARM 实验教学系统组成

1. Embest IDE 集成开发环境

Embest IDE 英文全称是 Embest Integrated Development Environment,是深圳市英蓓特信息技术有限公司推出的一套应用于嵌入式软件开发的新一代集成开发环境。Embest IDE 是一个高度集成的图形界面操作环境,包含编辑器、编译汇编链接器、调试器、工程管理、Flash 编程等工具。其 Embest IDE for ARM 版本支持所有基于 ARM7 和 ARM9 核的处理器,支持标准 C 和汇编语言开发。

2. Embest EduKit-III 实验台

实验台提供两套 CPU 子板,分别是 ARM7 的 S3C44BOX 芯片和 ARM9 的 S3C2410 芯片,两套 CPU 子板可以自由插拔。实验台提供多操作系统,ARM7 实验系统除可实现基础指令与接口实验, μ C/OS-II 操作系统和 μ CLinux 操作系统实验;ARM9 实验系统可以实现高端的 ARM 嵌入式教学,主要包括扩展接口实验,Linux、WinCE 和 VxWorks 操作系统的实验。

本书主要针对 ARM7 实验系统,包括 LED 灯、数码管、音频、麦克风、串口、并口、USB 接口、以太网接口、直流电机、步进电机、SD 卡、液晶、触摸屏、AD、DA 等实验。实验台硬件分布示意图如图 2-1-2。

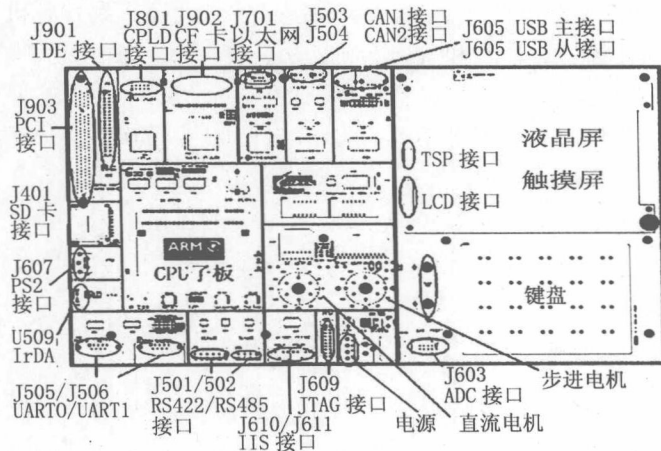


图 2-1-2 ARM 实验台硬件示意图

3. Embest JTAG 仿真器

JTAG 仿真器也称为调试器,使用集成开发环境配合 JTAG 仿真器进行开发是目前采用最多的一种调试方式。Embest JTAG 仿真器目前分为三种类型,包括标准型(Embest Emulator for ARM),增强型(Embest PowerICE for ARM)和通用型(Embest UnetICE for ARM),主要区别在于下载调试的速度和连接线不同。Embest JTAG 仿真器面板上面都有三只指示灯,对当前的工作状态做出指示。

4. Flash 编程器

当程序编完后,需把程序生成的二进制执行文件烧写到开发板的 Flash 存储器中观察实际运行结果。实验系统提供了一套完善的 Flash 编程器,配合 Embest JTAG 仿真器,对开发板上的 Flash 进行读写。图 2-1-3 是 Flash Programmer 软件界面。

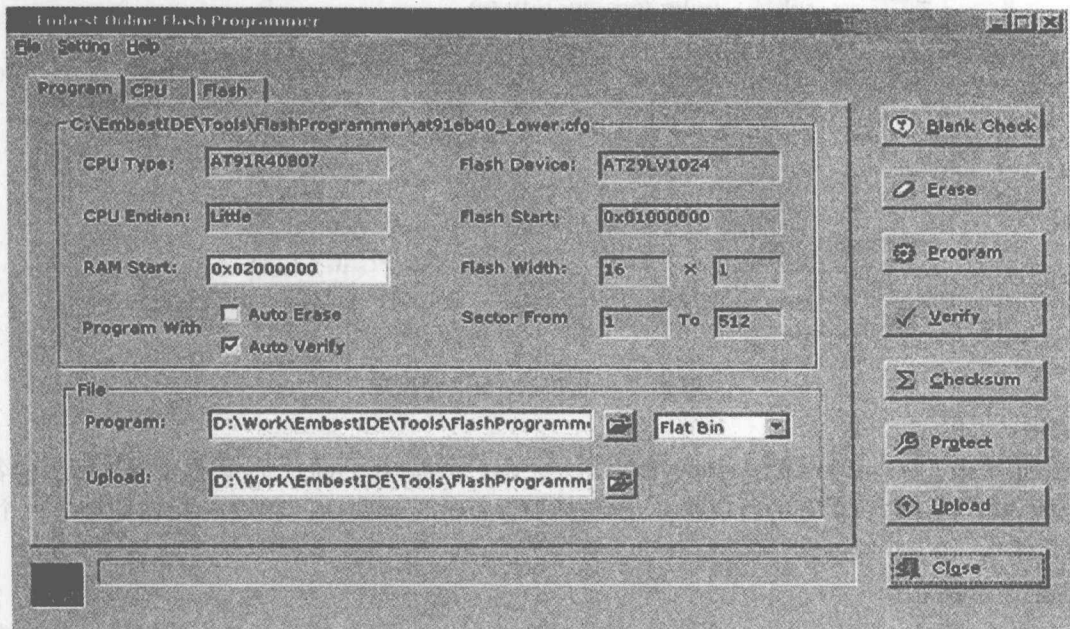


图 2-1-3 Flash Programmer 软件界面

2.2 ARM 实验教学系统使用说明

1. ARM 实验教学系统安装步骤

首先,在计算机和实验台都断电情况下,将 DB25 接口线分别连接到计算机和仿真器的并行口上;将 20 针的 IDC 插头,分别连接到仿真器和实验台的 JTAG 接口上(注:连接 JTAG 线只需要关闭实验台的电源即可)。

然后,打开计算机,分别安装 Embest IDE for ARM 软件和 Flash 软件。第一次使用这两个软件时,都需要按照实验教学系统供应商提供的注册方式对软件进行注册,才能使用。将光盘提供例程目录 EduKit44b0 拷贝到 Embest IDE 安装目录 \Examples\Samsung 的目录下。

最后,连接好实验台的 220V 电源线,插上 ARM7 CPU 核心板,并打开实验台开关,实验台自动运行自检程序。

2. Embest IDE for ARM 使用简介

运行 EmbestIDE.exe 后,进入 Embest IDE 集成开发环境(如图 2-2-1)。

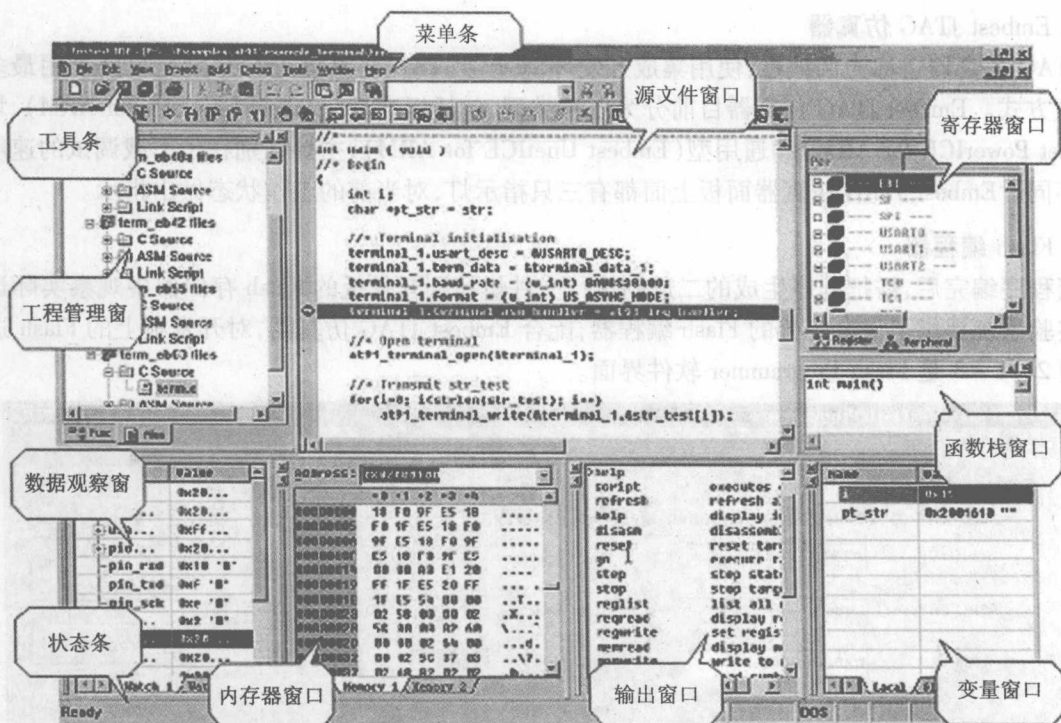


图 2-2-1 Embest IDE 主界面

(1) 建立工程

点击菜单项 File > New Workspace, 弹出建立工程对话框; 在 Project name 框中输入新工程名, Location 框中输入该工程要保存的目录路径, 或选择要保存的目录路径; 点击 OK 按钮即创建一个新工程(pjf 文件)。

(2) 建立新源文件

点击菜单项 File > New, 弹出一个新的、没有标题的文本编辑窗, 按照 C 或汇编格式保存文件到工程文件所在目录下; 输入源文件代码, 编辑器按语法来进行颜色加亮显示。

(3) 加入工程

点击菜单项 Project > Add To Project > Files 命令, 或单击工程管理窗口中相应右键菜单命令, 弹出文件选择对话框; 在工程文件所在目录下选择要加入的源文件, 可按 CTRL、或按 Shift 选择多个要加入的源文件。

(4) 设置活动工程

在一个工作区中可以存在几个同时打开的工程, 但只有一个工程处于活动状态, 其图标显示为加亮彩色, 处于活动状态的工程才可以作为调试工程。选中要设置为活动状态的工程, 点击鼠标右键, 在右键菜单中选择 Save as Active Project 项, 或点击菜单项 Project > Set Active Project, 可将该工程设置为活动状态。

(5) 工程基本配置

由于工程配置比较复杂, 其详细配置方法请看《Embest IDE for ARM 用户手册》。本文针对 ARM7 实验系统中的常规调试方式进行通用性设置。

① 处理器配置

选择菜单项 Project > Settings... 或快捷键 Alt + F7, 弹出工程配置对话框。在工程配置对话框中, 选择 Processor 配置对话框, 具体配置参数如图 2-2-2。

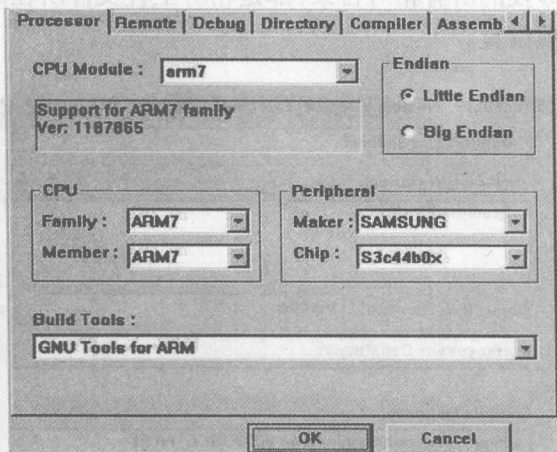


图 2-2-2 处理器配置

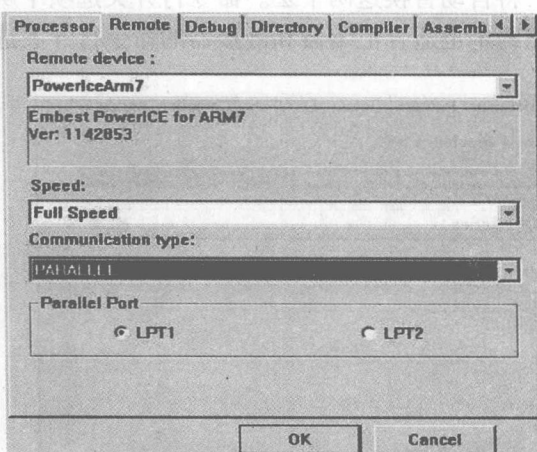


图 2-2-3 仿真器配置

② 仿真器配置

选择 Remote 配置对话框。具体配置参数如图 2-2-3。

③ 调试配置

选择 Debug 配置对话框。调试设置有三个类别选择页,分别为常规调试(General)配置页、下载(Download)配置页和内存映像(Memory Map)配置页。其中常规和下载方式具体配置参数如图 2-2-4 和 2-2-5。

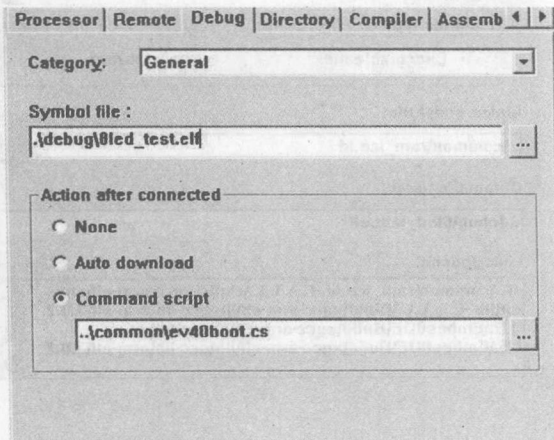


图 2-2-4 Debug 常规信息配置

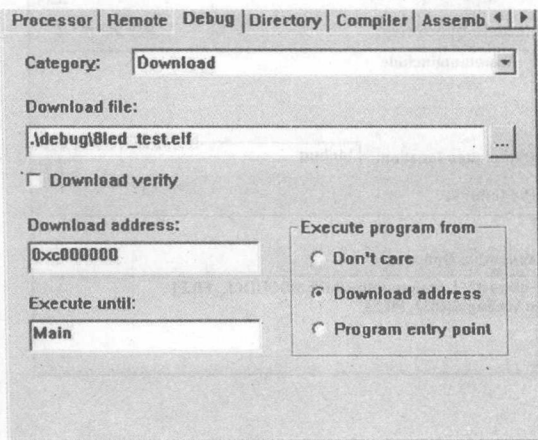


图 2-2-5 Debug 下载信息配置

④ 目录配置

选择 Directory 配置对话框。如果希望跟踪驱动函数库和部件函数库中的程序,可以在此添加其源文件目录,如没有则不添加,如图 2-2-6。

⑤ 编译器配置

选择 Compiler 配置对话框。该页中的命令行开关选项用于 C 和 C++ 编译器,用户的所有设置显示在 Compile Options 的编辑框中(图 2-2-7),工程在选定相应的 Build Tools(图 2-2-2)时,系统提供缺省设置。编译器命令开关选项格式如下:

[Opt-1] [Opt-2]... \$(SOURCE_FILE)... [Opt-n]... -o[Path] \$(OBJ_FILE)

用户可手工输入、修改开关选项,但必须注意保留每个 Option 之间的空格,并且不得改变和删除 \$(SOURCE_FILE) 和 \$(OBJ_FILE) 宏,它们分别代表源文件和目标文件名称。系统在执行编译命令

行时,将自动替换这两个宏。命令行开关选项中所涉及的所有相对目录,都是相对工程文件所在的目录,该约定也适合汇编器和链接器的命令行开关选项设置。

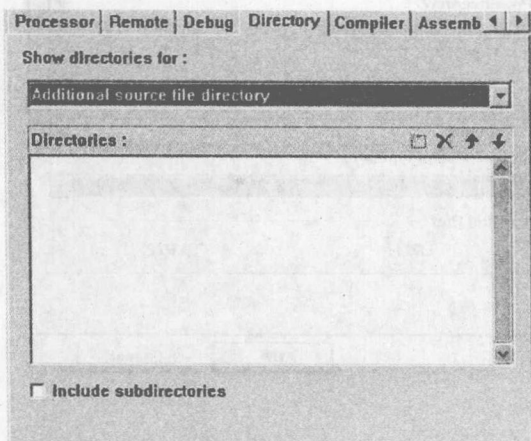


图 2-2-6 工程目录配置

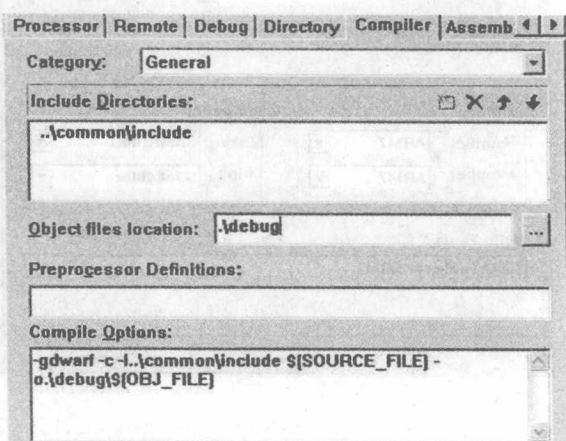


图 2-2-7 编译 General 选项配置

⑥ 汇编器配置

选择 Assembler 配置对话框(图 2-2-8)。汇编器命令开关选项格式如下:

[Opt-1] [Opt-2]... \$(SOURCE_FILE)... [Opt-n]... -o[Path] \$(OBJ_FILE)

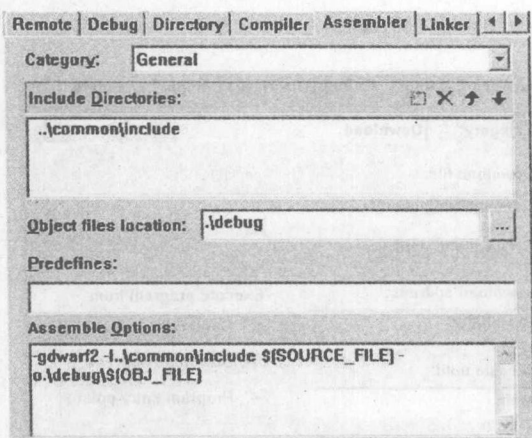


图 2-2-8 汇编 General 选项配置

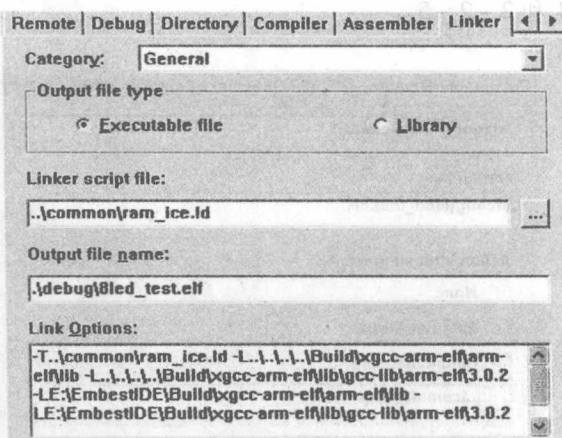


图 2-2-9 链接 General 选项配置

⑦ 链接器配置

选择 Linker 配置对话框。工程缺省链接输出的目标文件为执行文件,其链接器命令开关选项格式如下:

[Opt-1]... -o[Path] \$(TARGET_NAME) \$(OBJ_FILES) [Lib-1]...

其中 \$(TARGET_NAME) 为链接器输出的执行文件名称, \$(OBJ_FILES) 是链接的目标文件集合。

当工程链接输出的目标文件为库文件时,命令开关选项的格式如下:

[Opt-1]... \$(TARGET_NAME) \$(OBJ_FILES) [Lib-1]...

链接器配置有五个类别选择页,分别为常规信息配置页(General)、程序入口点配置页(Image Entry Options)、代码生成选项配置页(Code Generation Options)、链接库配置页(Include Object and Library Modules)和库搜索目录配置页(Add Library Searching Path)。具体配置参数如图 2-2-9 ~ 图 2-2-13。

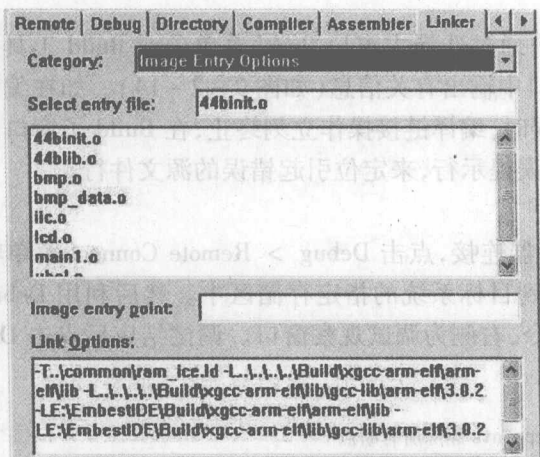


图 2-2-10 链接程序入口点配置

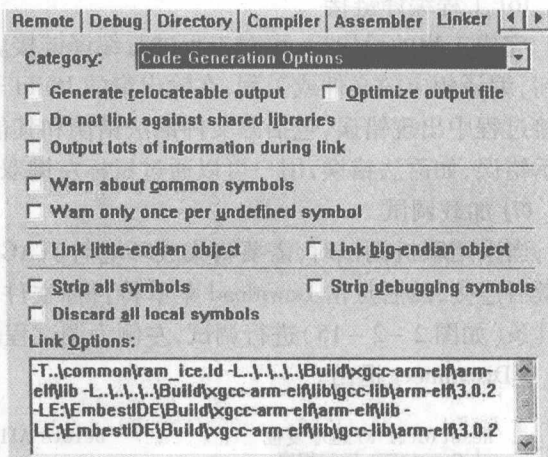


图 2-2-11 链接代码生成选项配置

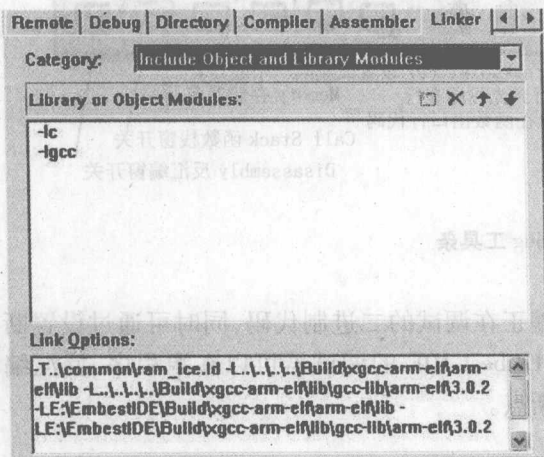


图 2-2-12 链接库配置

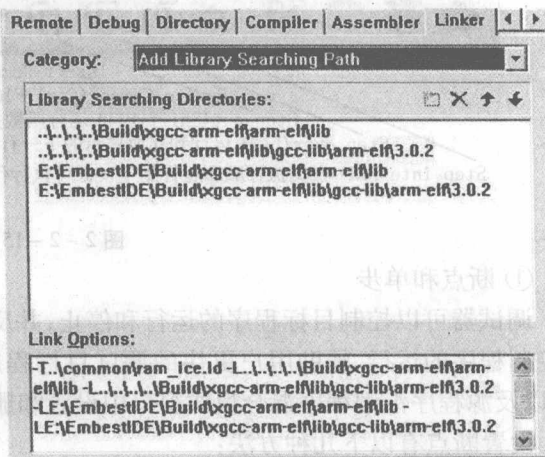


图 2-2-13 链接库搜索目录配置

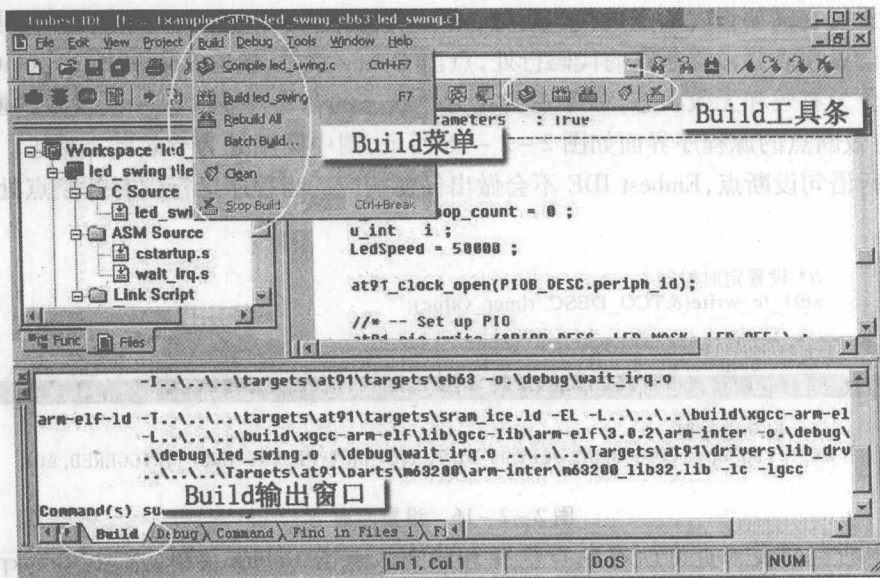


图 2-2-14 编译相关显示界面

(6) 工程编译链接

完成工程的设置后,可对工程进行编译链接。用户通过选择主窗口 Build 菜单项或 Build 工具条按钮,编译相应的文件或工程,在输出窗的 Build 子窗口中输出有关信息(如图 2-2-14)。如在编译链接过程中出现错误,包括源文件语法错误和其他错误时,编译链接操作立刻终止,在 Build 子窗口中提示错误,如语法错误,用户可以通过鼠标左键双击错误提示行,来定位引起错误的源文件行。

(7) 加载调试

当采用联机调试时,需要将集成环境与 JTAG 仿真器连接,点击 Debug > Remote Connect 菜单项可激活连接,然后点击 Download 菜单将目标文件下载到目标系统的指定存储区中。然后利用 Debug 工具条(如图 2-2-15)进行调试,左侧为调试程序命令,右侧为调试观察窗口。调试结束后点击 Debug > Disconnect 退出。

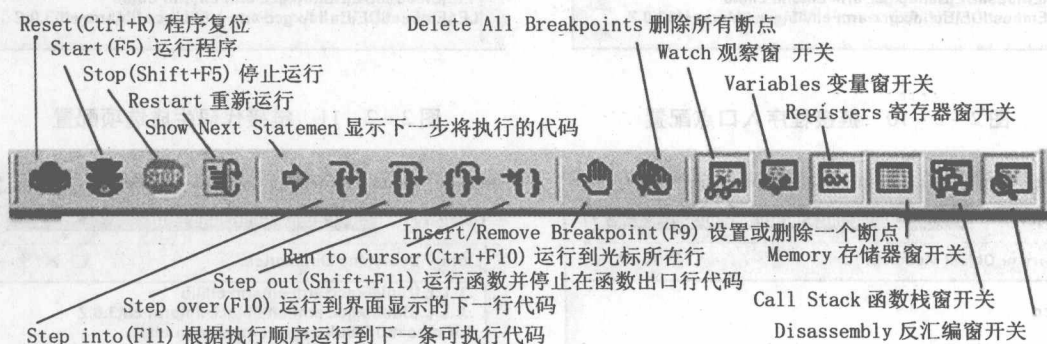


图 2-2-15 Debug 工具条

① 断点和单步

调试器可以控制目标程序的运行和停止,并反汇编正在调试的二进制代码,同时可通过设置断点来控制程序的运行,辅助用户更快的调试目标程序。Embest IDE 的调试器可以在源程序、反汇编程序、以及源程序汇编程序混合模式窗口中设置和删除断点。

设置断点有以下几种方法:

- 将鼠标指向源程序行左边需设定断点处,鼠标形状会变成手形指针,然后单击鼠标左键,这时,程序行左边会出现红色圆形含黄色条的断点标志;
- 将光标设置在需要设置断点的代码行处,按 F9 快捷键或 Debug 工具条设置断点图标;
- 将光标设置在需要设置断点的代码行处,点击 Debug 菜单,选取 Toggle Breakpoint 菜单项。

断点状态分为有效和无效状态,程序不会在无效状态的断点停止。第一次设置的断点均为有效断点,设置了有效断点的源程序界面如图 2-2-16 所示,图中圆圈处为有效断点标志。在源程序窗,若在一非执行语句设断点,Embest IDE 不会做出任何响应。当程序运行到有效断点处时,会停止在有效断点处。

```

/* 设置定时时间
at91_tc_write(&TCO_DESC.timer_value);

/* --Software Trigger on Timer
/* --Generates a software trigger simultaneously for each of the chan
at91_tc_trig_cmd(&TCO_DESC.TC_TRIG_CHANNEL);

/* 启动定时器
at91_irq_open(TCO_DESC.periph_id, 7, AIC_ARCTYPE_INT_EDGE_TRIGGERED, &0)

```

图 2-2-16 设置断点

用户可使用断点列表来查询所有断点信息和状态。点击 Debug 菜单,选取“Breakpoint...”菜单项,即可弹出断点列表对话框。在断点列表对话框点击 Delete 按钮将会删除当前选择的断点,点击 Delete ALL 按钮将删除当前列表中所有断点,点击 View Code 按钮将显示当前所选择断点对应的源文