

文章编号: 1000-5013(2010)04-400-04

# 利用 ISP 技术的控制系统网络实验室的设计

徐园园, 郑力新

( 华侨大学 信息科学与工程学院, 福建 泉州 362021 )

摘要: 将在系统编程( ISP) 技术和虚拟仪器有机结合, 开发网络化的单片机实验系统, 实现虚拟实验室的远程实时功能. 用户可在远程的客户端在线实时地调试自己编写的单片机程序, 通过网络对服务器端( 实验室) 的单片机进行远程在系统编程和实验开发, 完善网络控制实验室的功能. 经测试, 系统操作灵活, 运行可靠.

关键词: 网络实验室; 虚拟仪器; 单片机; 在系统编程

中图分类号: TP 391. 9; TP 273<sup>+</sup>. 5

文献标识码: A

在早期设计的网络实验室系统中, 由设计者编好单片机控制程序, 用户只能在客户端通过调节实验参数进行控制系统的实验, 却无法设计单片机程序. 为此, 本文将单片机的在系统编程( ISP) 技术引进控制系统网络实验室, 使用户可在客户端通过网络对服务器端( 实验室) 的单片机进行远程在系统编程和实验开发.

## 1 控制系统架构

单片机网络虚拟实验系统的结构框图, 如图 1 所示. 它采用 C/ S 模式, 但不是一般意义的“胖客户端/ 瘦服务器”, 而是针对高校实验室的特点设计的一种服务器( 实验服务器) 与客户端一一对应的实验系统<sup>[1-2]</sup>, 由客户端、服务器及单片机硬件电路 3 部分构成<sup>[3]</sup>.

用户程序运行过程中的状态和数据由服务器端负责采集, 服务器端把采集到的电机速度、误差数据、控制数据等通过网络传送到客户端, 客户端再把这些数据显示给用户.

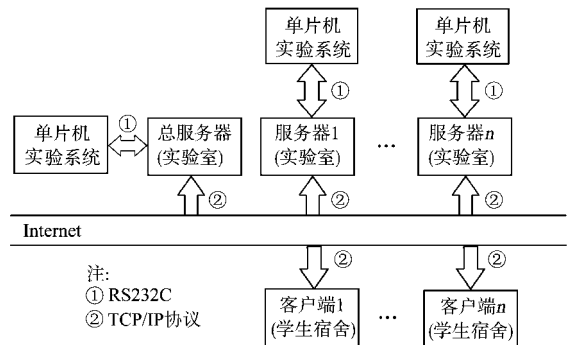


图 1 控制系统的结构图

Fig. 1 Control system diagram

## 2 上位机实现

系统的上位机功能要求客户端通过网络远程控制服务器, 并通过服务器端与单片机的 RS232 接口实现单片机程序的自动下载. 其设计包括客户端和服务器两部分.

### 2.1 客户端实现

首先, 用户在客户端使用 WAVE 等常用编译器进行源代码的编辑、编译及调试, 将单片机实验程序编译成 HEX 或 BIN 文件; 然后, 客户端通过基于 Lab Windows / CVI 平台的控制系统网络实验室, 应用 TCP/ IP 协议将 HEX 或 BIN 文件发送到服务器端, 以便单片机实现对其下载运行. 最后, 将单片机的运行结果从服务器端通过网络返回客户端的面板显示.

收稿日期: 2009-02-19

通信作者: 徐园园( 1980- ), 女, 助教, 主要从事虚拟仪器及智能检测控制技术的研究. E-mail: hixyy@126.com.

基金项目: 教育部科技研究重点项目( 207145 )

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

2.2 服务器实现

在服务器端利用计算机串口进行 ISP 编程方式下的程序下载. 服务器接收到客户端的 ISP 更新请求后, 接收用户的目标程序即 HEX 或 BIN 文件形成的字节流到一个数组. 然后向单片机传送 ISP 更新命令, 并通过 RS232 串口向单片机发送该字节流. 用户程序下载结束后, 服务端自动向单片机发送复位信号, 用户程序开始运行.

要实现自动下载功能必须实现 Lab Windows/CVI 环境下对 HEX 文件的解析. 其具体实现过程有如下 3 个步骤.

(1) 读取 HEX 文件. 可用 CVI 函数 IntFileSelectPopup(), GetFileInfo( path, & filesize) 等来实现<sup>[4]</sup>.

(2) 对照 ASCII 码转换表, 将 HEX 文件转化成十六进制的 ASCII 码. 由于读取的是 HEX 文件的十六进制字符的 ASCII 码的值, 要把其转化成十六进制码的真正表达的数值大小, 应参照 ASCII 码表做出 1~ 16 的码制转换. 需要注意的是, 读取的 HEX 文件中的换行和行头的“:” 也代表一个的 ASCII 的值, 要把其剔出.

(3) 剔除标识符之类的字符. 得到真正十六进制码后, 还需再对每行的 ASCII 码进行分析, 把其中的标识符等无效字符进一步剔除, 就可得到单片机程序的有效数码.

其主要源代码如下:

```
switch (event)
{ case EV ENT_COMMIT:
{ if( FileSelectPopup (“”, “* .c; * .h; * HEX”, “”, “”, VAL_SELECT_BUTTON, 0, 0,
1, 0, path))
GetFileInfo(path, & filesize); // 获取字节大小
n= sizeof(char);
fYdata= (char* ) malloc( filesize);
SetCtrlVal (panelHandle, PANEL_NUM, filesize/n);
FileToArray(path, fYdata, VAL_CHAR, filesize/n, 1, VAL_GROUPS_TOGETHER,
VAL_GROUPS_AS_COLUMNS, VAL_BINARY);
for(i= 1; i< filesize/n; i+ + )
{ if(fYdata[i] == 10) kongzhi= kongzhi+ 1; }
for(i= 1; i< filesize/n; i+ + )
{ if(fYdata[i] == 48) a[i] = 0X0;
if(fYdata[i] == 49) a[i] = 1; }
```

3 单片机的 ISP 功能实现

C8051F310( 美国 Cygnal 公司) 一般都固化有 ISP 引导程序, 但该引导程序是针对与它配套的 IDE 软件来设计的, 这显然不能达到设计要求. 因此, 需要重新编写 ISP 引导程序.

3.1 硬件电路设计

单片机硬件电路即下位机部分, 其主要任务是接收并响应上位机传来的信息, 对各种实验对象进行控制, 对运行所得的有用数据进行显示, 并通过串口回送给上位机. 图 2 为单片机硬件电路的原理框图.

3.2 ISP 编程

对 C8051F310 的 16 KB 的 FLASH 进行了 3

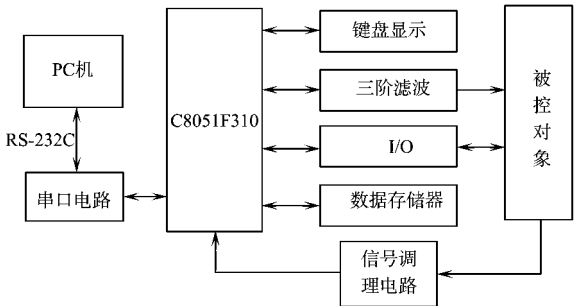


图 2 硬件电路的原理框图

Fig. 2 Block diagram of the hardware circuit

段分区, 分别为引导程序区(4 KB)、用户程序区(8 KB)和固化程序区(4 KB). 引导程序区存放系统初始化、程序更新及如何跳转的代码; 用户程序区存放用户自己编写的程序代码; 固化程序区存放设计者已编好的默认程序代码. 从服务器下载的程序存放在一个 8 KB 的外部数据存储器, 同时置位程序更新位, 以使引导程序能正确查询.

引导程序的流程图, 如图 3 所示. 单片机开始运行后, 先进行初始化设置, 包括初始化定时器/计数器、设定串行口波特率及其工作方式、设定中断优先级、启动定时器等. 然后, 读取更新位并进行判断. 若更新位为 1, 调用更新用户程序区的子程序, 执行用户程序的更新并跳转到用户程序区; 若更新位为 0, 跳转到固化程序区. 如果单片机已经进入固化程序区, 此时若接收到服务器的 ISP 命令, 则在完成接收代码后, 跳转到引导程序的更新部分, 执行程序更新并跳转到用户程序区.

FLASH 存储区要求在写入新值之前, 必须先擦除待编程地址的数据, FLASH 存储器是以 512 B 的扇区为单位组织的, 一次擦除操作将擦除整个扇区. 因此, 子程序中对 FLASH 的擦除和写操作都是以扇区为单位的. 首先, 先擦除一个扇区; 其次, 读取外部数据存储器(XRAM), 写入已擦除的扇区; 然后扇区数加 1 并判断是否为 16(8 KB=512 B×16), 直到等于 16, 退出子程序. 在对 FLASH 存储器编程的过程中, 要注意 FLASH 锁定和关键码功能, FLASH 的擦除及写操作<sup>[5]</sup>.

从用户软件写和擦除 FLASH 受 FLASH 锁定和关键码功能的保护. 在进行 FLASH 操作之前, 必须按顺序向 FLASH 锁定和关键码寄存器(FLKEY)写入正确的关键码 0xA5, 0xF1. 在使用 MOVX 指令对 FLASH 存储器写入之前, 必须先允许 FLASH 写操作.

擦除一个扇区有如下 6 个步骤: (1) 禁止中断; (2) 置“1”程序存储器擦除允许位(PSCTL 中的 PSEE), 以允许 FLASH 扇区擦除; (3) 置“1”程序存储器写允许位(PSCTL 中的 PSWE), 以允许 FLASH 写入; (4) 向 FLKEY 写第 1 个关键码(0xA5); (5) 向 FLKEY 写第 2 个关键码(0xF1); (6) 用 MOVX 指令向待擦除页内的任何一个地址写入一个数据字节.

// FLASH 擦除

```
FLASH_PageErase:      ; DPTR 传地址
MOV C, EA
CLR EA                ; 禁止中断
PUSH ACC
MOV VDM0CN, # 80h    ; VDD 监视器使能位
MOV RSTSRC, # 02H    ; 使能 VDD 监视器为复位源
XRL PSCTL, # 03h     ; 置位擦除允许位和写允许位
MOV FLKEY, # 0A5H    ; 关键码 1
MOV FLKEY, # 0F1H    ; 关键码 2
MOV A, # 0
MOVB @DPTR, A        ; 启动擦除
ANL PSCTL, # 0FCh    ; 置零擦除允许位和写允许位
MOV EA, C             ; 恢复中断
POP ACC
RET
```

允许 FLASH 写操作的过程: (1) 将程序存储写允许位 PSWE(PSCTL.0) 设置为逻辑“1”; (2) 按顺序向 FLASH 锁定寄存器(FLKEY)写入 FLASH 关键码. PSWE 位将保持置位状态, 直到被软件清除. 写 FLASH 存储器可以清除数据位, 但不能使数据位置“1”. 只有擦除操作能将 FLASH 中的数据位

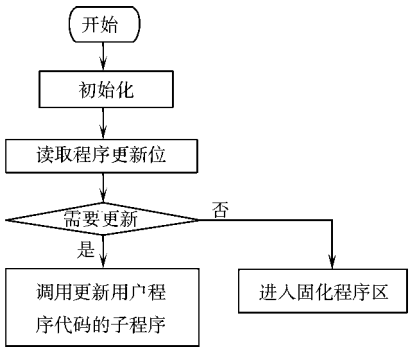


图 3 引导程序的流程图  
Fig. 3 Guide the process flow chart

置“1”. 因此, 在写入新值之前, 必须先擦除待编程的地址.

4 运行实例

以直流电机转速实验为例. 通过基于 ISP 技术的单片机网络实验系统, 用户可在客户端编写电机控制实验的单片机程序, 并将其下载到远程服务器端的单片机系统中. 经过实验运行, 可达到实时控制直流电机转速的系统要求. 客户端的运行结果, 如图 4 所示.

5 结束语

将 ISP 技术和虚拟仪器有机结合, 开发了网络化的单片机实验系统, 从而实现了虚拟实验室的远程实时功能. 该系统操作灵活, 运行可靠, 为组建基于网络的单片机实验开发系统提供一种新思路, 可在高校和科研机构等推广应用.

参考文献:

[ 1 ] INGVAR G. Remote laboratory experiments in electrical engineering education[ C] // Proceedings of Fourth IEEE International Caracas Conference on Devices, Circuits and Systems. Aruba:IEEE, 2002: 1-5.

[ 2 ] DEEPAK S, BABU J. An internet-mediated-process control laboratory[ J] . IEEE Control Systems Magazine, 2003, 23(1): 11-18.

[ 3 ] 徐园园, 郑力新. TCP/ IP 和 DataSocket 技术在网络虚拟实验室中的应用 [ J] . 电气电子教学学报, 2007, 29( 4) : 85-87.

[ 4 ] 张毅刚, 乔立岩. 虚拟仪器软件开发环境 Lab Windows/ CVI 编程指南[ M] . 北京: 机械工业出版社, 2002.

[ 5 ] 张迎新, 雷文, 姚静波. C8051F 系列 SOC 单片机原理及应用[ M] . 北京: 国防工业出版社, 2005.

Design of the Network Control System Laboratory  
Based on ISP Technology

XU Yuan-yuan, ZHENG Li-xin

( College of Information Science and Engineering, Huaqiao University, Quanzhou 362021, China)

**Abstract:** This article designs the network single-chip experimental system by the combination of in-system programming ( ISP) technology and virtual instruments. Through internet, users on the client can remotely debug their single-chip program, and program in the system to the single-chip on the server side ( Lab) . ISP technology makes functions of the network control laboratory more perfect. After testing, the system is flexible and reliable.

**Keywords:** network laboratory; virtual instrument; single-chip; in-system programming

( 责任编辑: 鲁 斌 英文审校: 吴逢铁)



图 4 客户端运行结果  
Fig.4 The results of clients running