

## 一种面向温室应用的嵌入式采集终端设计方案

善挺壁 汪懋华

(中国农业大学 信息与电气工程学院, 北京 100083)

**摘要** 针对我国温室生产中信息采集装备产业化相对落后,采集终端采用 PC 机不利于安装且成本较高的情况,利用运算速度较快的 32 位嵌入式处理器,移植实时性较强的嵌入式 Linux 操作系统,扩展 USB 接口、CAN 总线接口和以太网接口,设计开发了一个基于现场总线网络和以太网,能够对温室信息进行实时采集,实现了多种数据融合、存储、传输的嵌入式采集终端。用户可以通过驱动程序和应用程序操作终端的 CAN 总线接口进行数据的采集和命令的发送,可将相关信息存储在 USB 海量信息存储设备中或通过以太网实现数据的远程传输。测试结果表明:终端对于错误的数数据帧具有识别能力,应用协议软件运行稳定;节点对正确的远程帧和数据帧的控制命令均能正确响应;终端能实时显示温室状态的变化并保存数据,采集器和监控中心的数据通讯稳定性较高。系统实现了温室环境信息实时、远程监测的功能,同时,系统的设计降低了温室生产和管理成本。

**关键词** 温室; CAN 总线; 嵌入式终端; 数据采集

**中图分类号** TP 399; S 625.5

**文章编号** 1007-4333(2007)02-0080-04

**文献标识码** A

### Applied research on embedded terminator for greenhouse based on CAN bus

Shan Tingbi, Wang Maohua

(College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China)

**Abstract** An embedded terminator based on CAN bus-mastering system was developed for the control system in a greenhouse. A S3C2410 microprocessor with an operating system of ARM-Linux was employed. The terminator could collect data from sensors and send commands manipulated by applications through the CAN bus port. The CAN-bus nodes include the I/O modules and the sensor modules to transmit data as well as to execute the control functions. The stabilization and the control functions of this system performed well in the experiment. The system was developed at a low cost and has a great potential application and extension after being industrialized.

**Key words** greenhouse; CAN bus; embedded terminator; Linux data collection

现代传感器技术、通信技术、自动化技术和计算机技术的发展为现代温室测控系统的架构提供了多种可选方案。集散控制系统(distributed control system, DCS)与现场总线控制系统(field control system, FCS)在温室测控领域的应用已有较多报道。集散控制系统得益于集中管理、分散控制的核心思想,现场总线控制系统宣称实现了全分散、全开放的要义。随着设施农业朝集约化、规模化方向发展,对测控系统远程通信的要求愈加强烈,而且控制网络的发展趋势是与 Internet 集成<sup>[1]</sup>。

传统温室监控系统一般由基于 51 单片机的采

集控制仪构成。为了便于集中监控,采集控制仪通过现场总线与负责监控的 PC 机连接并进行数据通讯,一定程度上满足了中小型温室自动化生产的需要;但 PC 机体积较大,不利于安装,布线复杂且成本较高,不利于产业化应用推广<sup>[2]</sup>。

基于 CAN 总线的嵌入式终端是对传统温室监控系统的改进与扩充。用基于 ARM 技术的微处理器取代 PC 机,系统整体性能不会改变,而且便于引入 Internet 技术,成本大幅度下降。现场总线技术和嵌入式技术应用于植物环境生产系统,可实时获取植物长势、土壤、基质等信息,支持温室作物生产

收稿日期: 2006-08-11

基金项目: 中国-希腊 2005—2007 年科技合作项目

作者简介: 善挺壁,硕士研究生;汪懋华,教授,博士生导师,通讯作者,主要从事精细农业系统集成技术研究, E-mail: mhw@public.bta.net.cn

管理辅助决策,稳定性高而且成本低廉,十分利于产业化推广应用<sup>[3]</sup>。

本文中提出一种温室采集系统中应用的嵌入式总线终端设计方案,终端基于 ARM9 处理器设计,可以支持嵌入式 Linux 操作系统,既作为一个节点挂接在 CAN 总线上,又可以作为嵌入式 Web 服务器,旨在实现温室环境数据的实时采集与远程传输,降低成本。

## 1 硬件设计

系统硬件结构见图 1,整个硬件系统由 ARM9 微处理单元节点、智能模块节点和 CAN 总线组成。

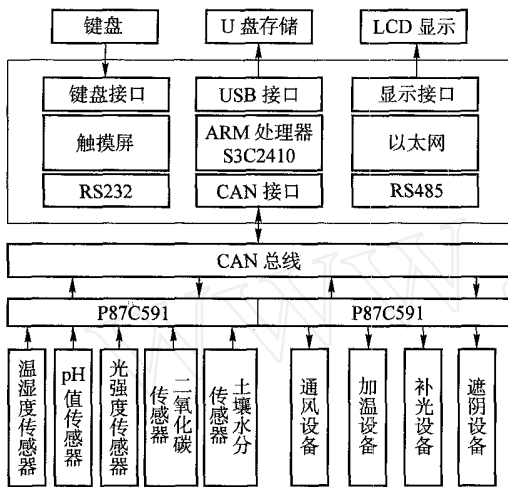


图 1 系统硬件结构框图

Fig. 1 Structural chart of system hardware

ARM9 微处理单元以 S3C2410 微处理器为核心,采用 203 MHz 的 ARM920T 内核,集成支持 STN 屏和 TFT 屏的 LCD 控制器、2 路 SPI 总线接口、117 个通用 I/O 口、24 通道外部中断源等,同时外接 16 MB 的 Nor Flash 和 64 MB 的 SDRAM,可为研究提供足够的空间。

通过 SPI 接口扩展 CAN 控制器 MCP2510,使 ARM9 微处理单元作为一个节点挂接在总线上,接收总线数据并发送控制命令。

人机接口由液晶显示屏和 4 × 4 键盘构成,可以显示温室状态信息和执行机构状态信息,设置系统参数。S3C2410 微处理器通过它的 LCD 控制器控制液晶显示模块。液晶显示模块采用夏普 LQ035Q7DB02,屏宽 3.5 英寸,分辨率 240 × 320 像素,TFT 真彩显示,主控芯片为 LZ9FC22。4 × 4 键盘按键分别定义为:数字键(0~9)、退出键(Esc)、上移键(↑)、下移键(↓)、确定键(Enter)、返回键(Back)、小数点(.)、关机键(Close)。键盘采用 4 行

4 列的矩阵形式,8 条引出线与 S3C2410 微处理器相应的 8 个 GPIO 口相接,采用查询方式进行判断。通过键盘和液晶显示屏,温室的状态和输出命令能够实时显示。人机交互主要通过小键盘和液晶屏进行,在此基础上设计开发了友好易操作的人机交互界面。

ARM9 微处理单元定时接收总线数据,并由显示屏分 3 屏显示当前室内外传感器节点的返回值和执行机构的状态值,显示屏数由 4 × 4 键盘控制。

由于系统要求处理大量数据,S3C2410 微处理器扩展的 Flash 和 RAM 存储器容量有限,而 USB 存储设备具有容量大、携带方便、支持热插拔等特点,非常适合温室大容量数据存储。S3C2410 微处理器具有 2 个 USB 接口,1 个主机接口和 1 个主从复用接口。本设计中使用 1 个主机接口连接 USB 存储设备。

CAN 总线上其他节点分别用于采集传感器状态信息,接收总线终端控制命令,进而控制相应执行机构。传感器模块可将采集到的物理信号转换为电信号,并以 CAN 数据帧的形式发到总线上,作为控制执行机构动作的依据。

节点模块采用 Philips 半导体公司生产的集成有 CAN 控制器的增强型 51 单片机 P87C591。它是一个 8 位高性能微处理器,除了具有 51 系列微处理器的通用功能和片内 CAN 控制器外,还集成有 6 路模拟输入的 10 位模数转换通道,并可选择快速 8 位模数转换,可将传感器输出的模拟信号转换为数字信号,发送到总线上<sup>[4]</sup>。

## 2 软件设计

### 2.1 操作系统的移植

S3C2410 微处理器主频最高可达 266 MHz,分 5 级流水线工作,支持 Linux 操作系统。移植过程主要分为 4 部分:

- 烧写 ppcboot. bin;
- 烧写内核 zImage;
- 烧写文件系统 ramdisk. image. gz;
- 烧写 cramfs 文件系统和 JFFS2 文件系统。

内核运行之前需要系统引导程序 ppcboot 完成加载内核和一些辅助性工作,然后跳转到内核代码的起始地址。ArmLinux 内核的启动分为 3 个阶段:第 1 阶段进行 CPU 和体系结构的检查,CPU 本身的初始化,页表的建立等;第 2 阶段对系统中一些基础设施进行初始化;第 3 阶段则是更高层次的初始化,如根设备和外部设备的初始化。Ppcboot 的主要

作用是初始化必要设备,然后调用内核,同时传递参数给内核;主要完成的工作有:建立和初始化 RAM,初始化 1 个串口,检测机器的系统结构,建立内核的 tagged list 和调用内核镜像。

## 2.2 CAN 控制器 MCP2510 驱动程序的编写

在 Linux 系统中,对用户程序而言,设备驱动程序隐藏了设备的具体细节,向各种不同设备提供了一致的接口,一般来说是把设备映射为一个特殊的设备文件,用户程序可以像对其他文件一样对此设备进行操作。Linux 对硬件设备支持 2 个标准接口:块特别设备文件和字符特别设备文件。通过块(字符)特别设备文件存取的设备称为块(字符)设备或块(字符)设备接口。块设备接口只支持面向块的 I/O 操作,所有 I/O 操作都通过内核地址空间的 I/O 缓冲区进行,它可以支持几乎任意长度和任意位置上的 I/O 请求,即提供随机存取的功能<sup>[5]</sup>。

设备驱动程序主要组成部分:

1) 自动配置和初始化子程序,检测所需驱动的硬件设备是否存在和能否正常工作;

2) 服务于 I/O 请求的子程序,又称驱动程序的上半部分;

3) 中断服务子程序,又称驱动程序的下半部分。

驱动的编写过程,其实就是根据需求“填写”Linux 系统 file\_operations 结构体中定义的全部或部分函数的过程。file\_operations 结构体中定义了设备驱动程序所提供的入口点位置和常用操作函数,如 read、write、ioctl、mmap、open、release/close 等,通过定义各种函数,实现对设备的操作<sup>[6]</sup>。

## 2.3 应用程序的编写

应用程序通过 Linux 下的 C 编程实现,主要功能是遵循 CAN 总线协议,按照预定的时间间隔采集温室信息,显示在液晶屏上,保存到 U 盘中并控制现场执行机构动作。根据实现的功能将系统分为总线数据采集显示和 U 盘保存 2 个主要功能模块。

总线数据采集和显示可直观展示温室环境信息,将用户关心的属性包括温室温湿度、光照强度、二氧化碳浓度等保存至不同的全局变量中分别处理。

### 2.3.1 数据采集

MCP2510 芯片加载驱动之后就可以实现数据采集工作,实现步骤如下。

1) 打开 MCP2510 设备: `int CAN_fd = open("/dev/CAN/MCP2510", O_RDWR);` CAN\_fd 为返回的设备句柄。

2) 调用 read 函数读取总线数据。由于从驱动中获得的只是 CAN 数据帧,需要对其数字字节进

行转换。

3) 转换后得到温室的温湿度值、光照强度值、二氧化碳浓度值等,将其显示在液晶屏上,并对执行机构发出控制命令。

数据采集流程见图 2。

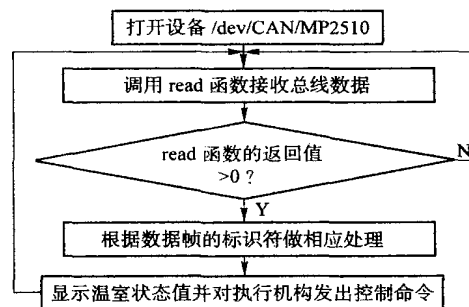


图 2 数据采集流程图

Fig. 2 Flow chart of data collection

### 2.3.2 数据存储

1) 存为文本文件。终端获得的数据是多样的,包括反映温室状态信息的多种传感器的数据、反映执行机构状态的数据等,但是目前这些信息都是独立的。只有将这些独立的数据融合起来,即只有在某个特定时间、特定地点采集到的特定的温室状态数据,才有意义。

系统使用结构体存放整合数据,结构体中定义了多种数据,将前面应用程序采集的全局变量依次填充此结构体,包括模拟传感器数据、数字传感器数据、执行机构状态数据、日期和时间等;然后将数据依次写入打开的文本文件,每条语句的格式为:日期、时间、传感器 1、传感器 2、传感器 3、传感器 4、执行机构 1、执行机构 2、执行机构 3、执行机构 4。有着固定顺序的数据流将有助于系统用户对温室状态进行测控。

2) U 盘存储。文本文件存在终端的内存中,如果要对数据做进一步分析,就需将其存入 U 盘实现长久保存。Linux 系统对 USB 设备的存储提供了良好的支持。执行方式是首先挂载 U 盘,然后将写有数据的文本文件存入其中。注意在存入 U 盘前需要把文本文件关闭,否则会出现数据丢失的情况。

### 2.3.3 以太网传输

Linux 中的网络编程通过 socket 接口进行。socket 接口是一种特殊的 I/O,也是一种文件描述符。每一个 socket 接口都用一个半相关描述(协议、本地地址、本地端口)表示;一个完整的套结字则用一个相关描述(协议、本地地址、本地端口、远程地址、远程端口)表示。socket 也有一个类似于打开文

件的函数调用,该函数返回一个整形的 socket 描述符,随后的连接建立、数据传输等操作都通过 socket 来实现。

进行 socket 编程的基本函数如下:

socket 函数用于建立一个 socket 连接,可指定 socket 类型信息。建立 socket 连接后,对 socketadd 或 socket . in 初始化,以保存建立的 socket 信息。

bind 函数用于将本地 IP 地址与端口号绑定,若绑定其他地址则不能成功;另外,它主要用于传输控制协议 TCP 的连接,在用户数据报协议 UDP 的连接中则无必要。

connect 函数在 TCP 中用于 bind 之后的 client 端,用于与服务器端建立连接。由于 UDP 中没有 bind 函数,connect 类似于 bind 函数的作用。

send 和 recv 函数用于接收和发送数据,既可用在 TCP 中也可用在 UDP 中。用在 UDP 中时,应在 connect 函数建立连接之后再行。

sendto 和 recvfrom 函数的作用与 send 和 recv 函数类似,也可以用在 TCP 和 UDP 中。用在 TCP 中时,函数作用等同于 send 和 recv;用在 UDP 中时,可以用于没有使用 connect 的情况,这 2 个函数可以自动寻找地址并进行连接。

实际应用中服务器端首先建立 socket,然后调用客户端的绑定与客户端建立联系,并接受客户端发送的消息;客户端则在建立 socket 连接之后调用 connect 函数来建立连接。

最后编写 Makefile 文件,利用 GUN 下的 gcc 和 gdb 工具编译和调试应用程序,通过以太网口与串口更新文件系统,加载设备驱动程序并移植应用程序。

### 3 终端性能测试

在实验室环境下对系统的软、硬件进行了一系列试验。

1) 应用协议稳定性测试。通过 CAN 总线向终端发送随机数据帧,观察终端动作和回送数据。结果表明:终端对不符合应用协议的数据帧没有响应,不回送数据,输出通道没有错误响应,程序正常运行。表明终端对于错误的的数据帧具有识别能力,应用协议软件运行稳定。

2) 应用协议正确性试验。人为升高环境温度,测试终端收到数据帧后的响应。若环境温度升高,应发送输出控制命令,终端通过输出板做输出响应:开启或关闭相应的发光二极管。测试结果表明,节点对正确的远程帧和数据帧的控制命令均能正确响应。

3) 温室状态显示与数据处理正确性试验。使环境温度变化,检验终端对温室状态的显示和数据处理情况。结果表明:终端能实时显示温室状态的变化并保存数据,采集器和监控中心的数据通讯稳定性较高,一般文本文件都能接收到,没有出现错误。

## 4 结 论

嵌入式终端基于 ARM9 处理器设计,支持嵌入式 Linux 操作系统,既作为一个节点挂接在 CAN 总线上,又可以作为嵌入式 Web 服务器,通过友好的人机交互界面实现温室环境数据的实时采集与远程传输,且实现容易。其特点如下:

1) 作为 CAN 总线上的节点功能齐全,利用主流嵌入式操作系统 ArmLinux,可实现多任务、多用户操作,实时性良好;通过键盘和显示屏结合,可准确设置系统参数和实时显示各工作节点的状态与数据。

2) 可以实现温室控制系统的数据采集、存储、传输,采集到的数据符合预期结果,符合工程指标,保存的文本文件为标准格式,可以利用 Excel 或 Access 进行处理。

3) 温室嵌入式采集终端实现了温室环境信息监控的网络化、实时化。管理人员通过访问嵌入式采集终端的 IP 地址就可以实时监测温室内部环境信息。管理人员可以对各种突发事件迅速做出决定,提高了工作效率,克服了农业设施自身的地域性缺陷。

运行测试结果证明,终端工作安全可靠、数据收发正确,系统设计更加紧凑,资源利用率和数据传输速度都有了很大的提高,对建设工厂化、智能化的温室极具参考价值。

## 参 考 文 献

- [1] 胡涛. 基于以太网的温室测控系统架构[J]. 安徽农业科学, 2005, 33(5): 870-871
- [2] 齐文新,王明爽,陈凤. 基于 ARM 和 Internet 的远程温室监控系统[J]. 计算机与数字工程, 2006, 34(3): 125-127
- [3] 王伟,阎勤劳,李磊,等. 嵌入式网络通信系统在温室中的应用[J]. 农机化研究, 2006(9): 158-160
- [4] 杨明,马祖长,赵广发. 基于 CAN 总线的智能温室控制系统[J]. 自动化技术与应用, 2006, 25(4): 50-51
- [5] 孙天泽. 嵌入式设计及 Linux 驱动开发指南—基于 ARM9 处理器[M]. 北京:电子工业出版社, 2005
- [6] Corbet J, Ruibini A. Linux 设备驱动程序[M]. 北京:中国电力出版社, 2000