

嵌入式软 PLC 系统的研究和实现

吴玉香, 周东霞, 林锦赞

(华南理工大学自动化学院, 广州 540641)

摘要:介绍嵌入式软 PLC 硬件设计平台和软件设计平台, 利用梯形图、硬件接口映射及模块化设计技术开发面向工业控制嵌入式软 PLC 系统。软 PLC 系统由开发系统和运行系统组成, 其中开发系统将程序翻译成目标代码, 运行系统加载目标代码, 根据目标代码执行指令, 实现控制功能。

关键词:嵌入式软 PLC; 梯形图; 映射技术

Research and Implementation of Embedded Soft PLC System

WU Yu-xiang, ZHOU Dong-xia, LIN Jin-yun

(College of Automation, South China University of Technology, Guangzhou 540641)

【Abstract】The hardware and software design platform of embedded soft PLC is presented. Embedded soft PLC system for industrial control is developed by the techniques of ladder diagram, hardware interface mapping and modular design. This system is composed of development system and operating system. The development system translates the programs based on graphics language into target code, and the operating system loads the target code and executes command based on the code to control the system.

【Key words】embedded soft PLC; ladder diagram; mapping technology

1 概述

传统 PLC 由于技术封闭, 导致各产品互不兼容, 硬件和软件的开放性差^[1]。软 PLC 就是针对传统 PLC 的缺陷提出的。软 PLC 是一种基于 PC 机用软件实现传统硬件 PLC 控制功能的控制装置。或者说将 PLC 要实现的功能封装在软件里, 然后运行于硬件系统平台上。软 PLC 综合了计算机和 PLC 的开关量控制、模拟量控制、逻辑运算、数值处理、网络通信等功能, 通过一个多任务内核提供强大的指令集、快速准确的扫描、可靠的操作、可连接的各种 I/O 系统及网络接口。嵌入式软 PLC 由软 PLC 开发系统和软 PLC 运行系统组成, 开发系统负责梯形图的编辑编译, 运行系统负责 PLC 指令的解释执行。

2 嵌入式软 PLC 设计平台

2.1 硬件平台

嵌入式软 PLC 系统理想的硬件设备应具有运算能力强、程序容量大和接口丰富等特点, 因此, 一般软 PLC 系统选择的硬件平台为工业 PC 或嵌入式 PC。嵌入式 ARM 处理器具有高性能、低功耗、低成本等特点, 能为普通嵌入式系统提供完整的解决方案, 因此, 本硬件平台采用一款 ARM 内核的 32 bit RISC 芯片——Samsung 公司基于 ARM7TDMI 内核的 S3C44B0 芯片。它支持片上的 ICE 断点调试, 内含 32 bit 的硬件乘法器, 并通过提供丰富的、通用的片上外设, 减少嵌入式系统中的元器件配置, 从而使系统成本最小化。

2.2 软件平台

uClinux 是一个功能强大、高度优化的嵌入式操作系统, 具有强大的网络和任务管理功能, 便于硬件资源的分配、管理, 因此, 本文设计选 uClinux 为软 PLC 运行系统的运行环境, 采用 Red Hat Linux 操作系统作为软 PLC 的编辑编译系统的上层环境。

嵌入式系统是一个资源受限的系统, 在其硬件平台上编写程序很困难, 甚至不可能。目前一般采用交叉编译方法, 在 PC 上生成目标平台能运行的二进制代码格式。本系统的运行环境是 uClinux, 因此, 要在 Linux 下对运行系统的程序进行交叉编译, 将生成的可执行文件加入到 uClinux 的文件系统中生成目标文件, 然后运行于 ARM 上。

3 嵌入式软 PLC 系统设计

3.1 嵌入式软 PLC 系统开发的总体框架

嵌入式软 PLC 系统主要由开发系统和运行系统组成。开发系统进行梯形图语言的编辑和指令的解析, 生成目标代码; 运行系统执行目标代码, 进行逻辑运算, 实现控制目的。系统总体结构如图 1 所示^[2]。

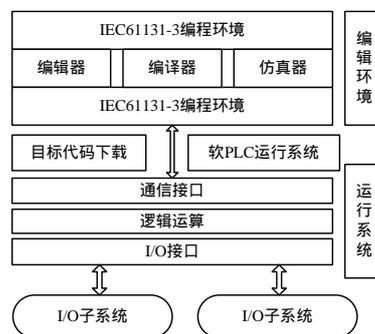


图 1 嵌入式软 PLC 系统总体框图

3.2 软件开发工具

IEC61131-3 是 PLC 编程语言的国际标准^[3], 它定义了 5 种 PLC 编程语言规范, 其中结构化文本(ST)和指令表(IL)

作者简介:吴玉香(1968 -), 女, 副教授, 主研方向: 智能控制及应用; 周东霞、林锦赞, 硕士研究生

收稿日期: 2008-11-17 E-mail: 564090759@qq.com

为文本语言,顺序功能图(SFC)、梯形图(LD)、功能模块(FBD)为图形语言。图形语言简单、形象、易于理解,因此,本系统采用梯形图语言作为用户程序的开发语言。

本系统中嵌入式软 PLC 人机界面模块采用 Gtk/Gtk+设计语言实现。Linux 系统下最主要的图形库是 QT 和 GTK+。QT 系统库源代码不开放,如果使用这个库,需要支付昂贵的授权费;而 Gtk+是基于 LGPL 授权的,使用 Gtk+开放源代码软件,不需付费,而且 Gtk+图形库使用一系列的构件对象来创建图形用户接口,易于学习和使用,可以解决复杂的控制和计算问题,也可根据需要自定义功能强大的数据结构。其他模块均采用 C 语言实现,C 语言代码具有良好的移植性。

3.3 梯形图数据结构与存储

3.3.1 梯形图主要的数据结构

梯形图的编辑元器件主要包括常开接点、常闭接点、线圈等简单元件和定时、比较等功能块元器件。为了灵活应用和扩展编辑元器件,方便编程,采用统一的数据结构标准^[4]。

```
typedef struct StrElement
{
    short int Type;           //元件类型
    char ConnectedWithTop;   //元件逻辑连接关系
    int SaveType;           //元件存储类型
    int SaveNum;           //元件存储位置
    char DynamicInput;      //模拟仿真输入
    char DynamicState;      //模拟仿真状态
    char DynamicVarBak;
    //复杂元件模拟仿真的触发类型
}StrElement;
```

对于简单元件,逻辑运算只是进行接点状态的逻辑转移,参数类型为布尔型,其值为真(TRUE)或是假(FALSE)。但功能块元器件的参数可以是任意类型,除输入输出是布尔类型外,还有其他的类型,如定时器的基数(分、秒)、预置值等采用的是整形。因此,对功能元器件还要定义相关的参数数据结构,如定时器。

```
typedef struct StrTimer
{
    int Value;
    int Base;
    //基准值: MINS、SECS、100SECS
    int Preset;           //预置值
    char DisplayFormat[10];
    //仿真时在梯级图上定时显示格式
    char InputEnable;
    char OutputDone;
    char OutputRunning;
}StrTimer;
```

梯形图整个编辑过程都是动态的,避免了繁杂的数据出现,采用梯形图梯级链表结构,简化了开发难度。在数据加载、逻辑运算和数据处理时通过链表结构对梯级数据进行遍历及相关操作。

```
typedef struct StrRung
{
    Int Used;
    int PrevRung; //标记前一梯级
    int NextRung; //标记后一梯级
```

```
char Label[LGT_LABEL];
//梯级标号 跳转标识
char Comment[LGT_COMMENT];
StrElement
Element[RUNG_W][RUNG_H];
}StrRung;
```

3.3.2 梯形图的数据存储格式

软件设计中采用模块化设计,每个模块实现一定的功能。全局变量是所有模块的公共数据,各模块通过全局变量进行通信^[5]。

```
Char* VarArray;
//存储物理 I/O、中间位变量指针
StrTimer * TimerArray;
//存储定时器数据指针
StrArithmExpr * ArithmExpr;
//存储比较表达式数据指针
...
```

梯形图中简单元件有效的信息包括元件类型、元件的链接逻辑关系、元件保存类型和元件保存相对位置,对于功能元器件还有预置值等。其存储结构和梯形图数据存储格式如表 1 所示。

表 1 梯形图数据存储格式

| ; Rung : | | | |
|----------|----------|---------------|---------|
| #LABEL= | | | |
| | | #COMMENT=Test | |
| 3-0-0-19 | 1-0-0-39 | 9-0-0-0 | 9-0-0-0 |
| 99-0-0-0 | 99-0-0-0 | 20-0-0-1 | 9-0-0-0 |
| ;Timers: | | | |
| ;Base | | Present | |
| 0 | | 2 | |
| 1 | | 5 | |

在表 1 中,

Rung: Type-ConnectedWithTop-SaveType- SaveNum
//元件存储结构

Timers: Base- Preset
//定时器预置值存储结构

3.3.3 梯形图扫描算法

梯形图语言 LD 源自机电一体化的继电器系统,它描述一个 POU 的网络自左向右的能量流,主要用于处理布尔信号。元件设计时采用左连接,即将元件左边的状态连接到元件右边。

梯形图扫描和编译时以梯级为单位,梯级是由相互影响的行组成的最小单元,图 2 所示即为一个梯级。采用经典的深度扫描算法,按照梯形图网络的自左向右、自上到下地逐个梯级进行扫描。此算法容易实现,可以迅速深入到梯形图的逻辑关系中,图 2 中标出了扫描的详细顺序。

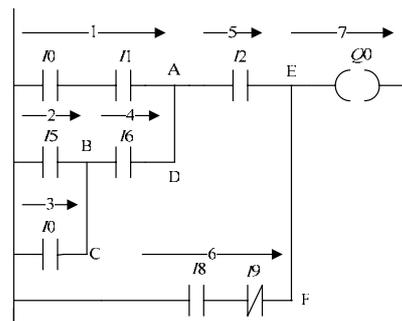


图 2 梯形图扫描过程

4 嵌入式软 PLC 开发系统的实现

整个软件设计采用模块化的设计方法,模块化设计易于裁剪,代码可利用率。开发系统分为软 PLC 界面模块、逻辑运算模块、文件操作模块、通信模块和硬件接口模块。

4.1 软 PLC 界面模块

软 PLC 开发系统的界面如图 3 所示,主要由元件库模块、编辑区模块和仿真模块 3 部分组成。

(1)元件库模块定义了常用的元件并对梯形图编辑进行管理,设置了梯级加入、确定或取消等功能键。

(2)编辑区模块采用栅格存储机制,每一栅格只能放置一个元器件,整个编辑区设定为一个二维空间,与梯级结构体中的 StrElement Element[RUNG_W][RUNG_H]结构体数据相一致。编辑区中定义了一些隐含的语法检错机制,如梯级的最后一列只能放 0 置输出元件。

(3)仿真模块在界面的右边,包含输入输出端口变量和定时器变量的状态仿真,可形象地检测程序运行性。仿真变量有物理输入变量、中间模拟输入输出变量和物理输出变量。在仿真时,点击相应的物理输入变量 I_n 、中间模拟输入输出变量 B_n 和物理输出变量 Q_n 的会作出相应的响应。

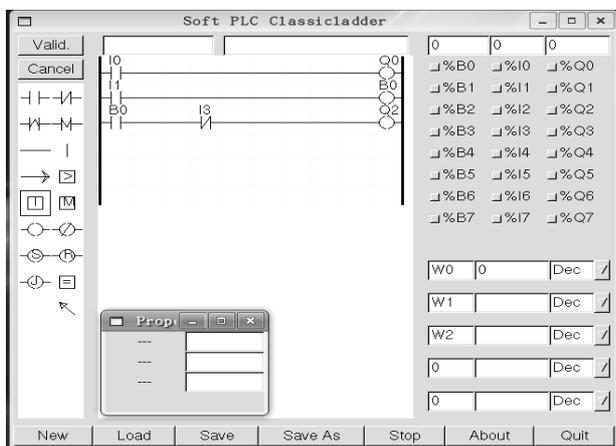


图 3 人机界面图

4.2 逻辑运算模块

逻辑运算模块是软 PLC 开发系统的核心模块,负责软 PLC 数据处理功能的实现。包含梯形图扫描、逻辑状态的运算转移、元件存储结构的识别等。软 PLC 调用逻辑运算模块来实现逻辑状态转移,如常开、常闭元件,其实现如下:

```

/* Elements : -|- and -|/- */
char CalcTypeInput(int x, int y, StrRung * TheRung, char IsNot,
char OnlyFronts)
{
...
if (IsNot)
//常闭 右连接 := 左连接 AND NOT
StateElement = !StateElement;
StateVar = StateElement;
...
StateVar = StateElement;
//常开右连接:=左连接 }

```

4.3 文件操作模块

文件操作模块的功能是:在开发系统下实现梯形图数据的保存;在运行系统下实现梯形图数据的加载。

4.4 硬件接口模块

硬件模块主要实现 I/O 接口和外部物理输入输出子系统

的映射关系,使软 PLC 的状态点和硬件的 I/O 点一一对应。在硬件接口模块中定义了 2 个函数接口:

```

void ReadPhysicalInputs(void);
void WritePhysicalOutputs(void);

```

在软 PLC 运行系统中,通过对上述 2 个函数的调用,使编译系统中定义的 I/O 变量 I_n 和 Q_n 与实际 I/O 子系统的 I/O 端口相关联,从而实现对 I/O 状态点的读取和输出控制。

在传统 PLC 中,I/O 端口的类型是预先定义的,其扩展只能通过 I/O 扩展模块进行,这样就需要增加硬件,而软 PLC 实现方法没有定义具体的硬件类型,可以根据具体选用的硬件类型来实现这 2 个驱动函数,获得灵活的 I/O 接口。

5 嵌入式软 PLC 运行系统的实现

运行系统由主程序模块、文件模块、逻辑运算模块和硬件接口模块构成。主程序模块调用文件模块加载文件数据,通过逻辑运算模块来刷新硬件 I/O。

嵌入式软 PLC 系统的运行系统的操作平台——uClinux,采用实存储器管理策略(real memory management),对内存进行直接访问(它对地址的访问不需要经过 MMU,而是直接送到地址线上输出),所有程序中访问的地址都是实际物理地址。操作系统对内存空间没有保护,各个进程共享一个运行空间(没有独立的地址转换表),因此,对按键和 LED 数据寄存器进行直接访问。

本设计的硬件实验平台是恒通科技的 44B0X 开发板,外部有 4 个独立按键和 3 个独立的 LED。通过对 void ReadPhysicalInputs(void)和 void WritePhysicalOutputs(void)这 2 个函数的实现,使 44B0X 的按键输入端口 GPC1~GPC3 和 LED 输出端口 GPG4~GPG7 映射到开发系统中定义的 $I_0\sim I_2$ 变量和 $Q_0\sim Q_3$ 变量,实现对 I/O 状态点的读取和输出。运行系统程序流程如图 4 所示。

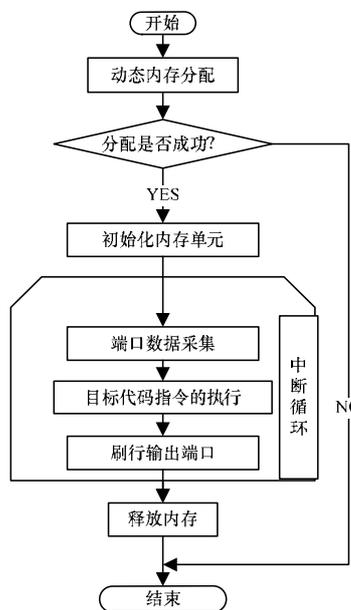


图 4 运行系统程序流程

6 结束语

本文设计的嵌入式 PLC 系统的初步测试已经完成,基本达到设计要求。由于运行系统环境为 uClinux,是一个非实时操作平台,因此控制有一定延时,但可以对 uClinux 使用 Rt-Linux 的 patch 进行改进,增强 uClinux 的实时性,使 uClinux 满足工业控制的实时要求。(下转第 242 页)