

文章编号:1001-9081(2006)02-0453-03

## 多总线控制系统通用化通信程序框架设计

兰京川, 陈光禄

(电子科技大学 自动化工程学院, 四川 成都 610054)

(lanjc@uestc.edu.cn)

**摘 要:** 对应用于控制系统的通用化通信软件的设计方法进行了研究, 通过监听任务下移及监听信息上传的技术方法实现了信息监听功能的通用化设计; 通过设计通信路径表的方法实现了数据交互的通用性。此处的通信是指控制系统中各软硬件模块的数据交互; 通用化特性体现在, 这种通信程序可以应用在任何多总线控制系统中, 达到一次设计、多个系统使用的目的, 便于控制系统的升级。

**关键词:** 总线; 计算机控制系统; 通用化; 通信程序

**中图分类号:** TP311 **文献标识码:** A

## Universal communication software architecture design for multi-bus control system

LAN Jing-chuan, CHEN Guang-ju

(School of Automation, University of Electronic Science & Technology of China, Chengdu Sichuan 610054, China)

**Abstract:** A design method for universal communication software applying to multi-bus control systems was studied in this paper. Implements a universal communication software by moving downwards the monitoring duty and transferring upwards the monitored information, and by designing communication paths table. Here, communication means transferring data in control systems. This communication software can used in any multi-bus control systems, and need not be modified when upgrading control systems, and this reflects its general characteristic.

**Key words:** bus; computer control systems; general purpose; communication software;

### 0 引言

目前, 有很多总线可以应用在各种控制系统中, 比如 CAN 总线、VME 总线、VXI 总线、CPCI 总线等等, 而且随着总线的发展, 控制系统可以不断的升级换代。复杂的控制系统, 除了需要有先进的控制算法以外, 还必须要有个通信程序来负责完成系统中各个软硬件模块的数据交互。然而随着总线的发展和控制系统的升级换代, 通信程序必须不断的修改以适应新系统中的总线和软硬件模块的变化。

通信程序必须修改的原因是, 通信程序在控制系统中是各软硬件模块信息交互的桥梁, 它负责捕获控制系统中模块的数据发送请求并首先获取数据, 然后根据数据用途转发到相应的目的模块。当系统升级换代, 由高性能总线替换旧的总线, 由软件模块替换硬件模块时, 捕获数据发送请求的机制将发生变化, 所以通信程序必须做相应的修改; 当系统功能扩展, 需要更多的软硬件模块来完成特定任务时, 系统中的数据交互将发生变化, 所以, 通信控制程序也必须做相应的改进。

设计通用化通信程序就是为了建立一种通用的, 可扩展的软件框架, 使得通信程序可以适应系统的上述变化, 避免软件的重复开发。设计通用化通信程序要解决的主要问题有两个: 第一, 如何建立一种通用的数据监听机制; 第二, 如何适应数据通信路径的改变, 快速准确地转发数据。本文将给出解决以上两个问题的实现方法。

### 1 监听机制的通用化设计

可以抽象出一个多总线控制系统的模型如图 1 所示。从图中可以看到:

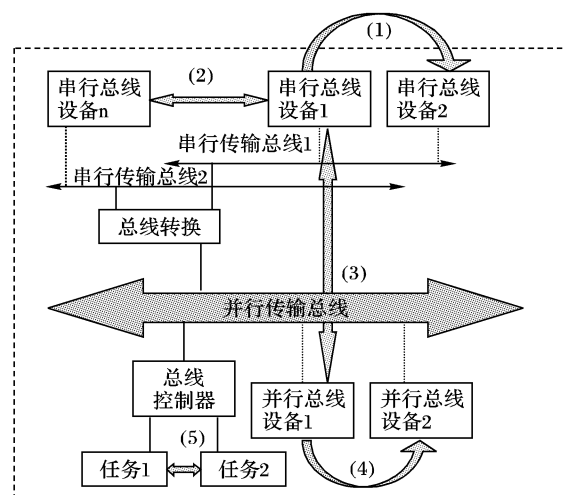


图1 系统通讯控制模型

1) 控制系统中可能既存在串行总线, 如 CAN 总线, 也有并行总线, 如 VME 总线、CPCI 总线;

2) 系统中可能存在多个硬件设备 (硬件模块), 也有软件模块 (图 1 中的“任务 1”和“任务 2”);

3) 图 1 中的“总线控制器”通常是一个带并行总线控制

收稿日期: 2005-08-03; 修订日期: 2005-10-26

作者简介: 兰京川 (1977-), 男, 讲师, 博士研究生, 主要研究方向: 自动测试系统集成、测试、控制系统建模; 陈光禄 (1939-), 男, 博士生导师, 主要研究方向: 测试理论、大规模集成电路测试、系统集成。

功能的计算机,通信程序驻留在此处。所有软件模块也可以驻留在此计算机或其他计算机中,需要具体情况而定;

4) 图 1 中的“总线转换”单元可以是挂接在并行总线上的一个智能硬件模块,或是上述计算机中的一块带有控制程序的智能转换卡。

图 1 中标注的(1)、(2)、(3)、(4)、(5)是系统中各模块的数据交互需求。在以上的数据交互过程中,多数情况下通信程序是数据交互的桥梁,如图 1 中的数据交互(3),通信程序必须首先监听到“串行总线设备 1”的数据传送请求及待传送的数据,然后转发给“并行总线设备 1”;又如图 1 中的数据交互(5),通信程序必须获得软件模块“任务 1”的数据传送请求及待传送的数据,然后转发给软件模块“任务 2”。

由以上分析可以看出,系统中存在多种数据请求监听机制;当系统的软硬件模块或总线结构发生变化时,数据请求的监听机制也必须发生变化。如果将监听程序集成在通信程序中,那么,当系统发生变化时,必须修改通信程序。用以下介绍的监听层次结构及软件实现方法可解决监听机制的通用化问题。

总的来说,可以用一句话概括:监听任务下移,监听信息上传。

### 1.1 监听任务下移

首先,为需要接入控制系统的软硬件模块提出以下接入规范:每个模块必须带有一个用于和其通信的驱动库。驱动库至少带有 3 个函数,分别是:

- 1) xxx\_ReadData (DATASTRUCT data),用于读取模块数据;
- 2) xxx\_WriteData (DATASTRUCT data),用于向模块写数据;
- 3) xxx\_Init(),用于初始化模块,主要是安装数据监听功能,函数参数见后续内容。

所谓监听任务下移就是指:把安装监听功能的任务下移到模块的通信驱动库中去完成。因为模块的设计厂商知道此模块将用于何种总线系统以及模块的软硬件特性,因此,设计厂商可以在 xxx\_Init() 函数中完成监听功能的程序设计。通用化的通信程序只需要调用模块的 xxx\_Init() 函数即可对模块安装监听功能。图 2 表示其监听层次结构。

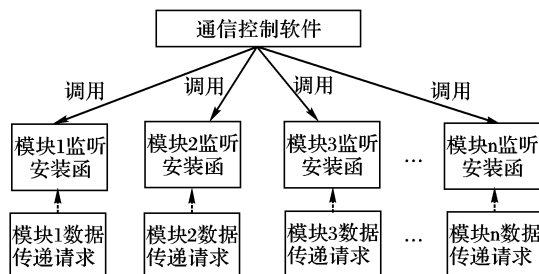


图2 监听层次

### 1.2 监听信息上传

当模块通信驱动库中的 Init() 函数捕获到模块的数据传递请求时,请求仅存在于驱动库(动态库)中,通信控制软件这时还未得到此请求。只有当通信请求到达通信控制软件后,通信控制软件才能调用 xxx\_ReadData() 函数获取数据,然后分析和转发数据。通过设计监听信息上传机制可解决此问题。具体方法如下:

1) 首先,在通信控制软件中实现监听任务处理函数 Monitorfunc(int 模块识别号)。

```

Monitorfunc(int moduleid)
{
    switch( moduleid)
    {
        case module1:
            module1_ReadData( pdata);
            break;
        case module2:
            module2_ReadData( pdata);
            break;
        ...
    }
}
  
```

并在通信控制软件中定义函数指针类型 FUNC; typedef int ( \* FUNC) ( int);

2) 然后,在通信控制软件初始化阶段,以 Monitorfunc 函数的地址作为实际参数,调用每个模块的初始化函数 Init(), 此函数具有以下函数原型:int Init ( FUNC FuncAddr)。

3) 最后,当模块的通信驱动库监听到数据传递请求后,在模块通信驱动库的监听函数中调用通信控制函数的监听任务处理函数 Monitorfunc( int moduleid), 并把模块自身的 ID 作为 Monitorfunc 函数的实参。

这样,监听信息就从模块通信驱动库中传递到了通信控制软件,完成了监听信息上传。通过监听任务下移和监听信息上传,实现了监听功能的通用化设计。

## 2 数据交互的通用化设计

通信控制软件需要不断修改的第二个原因是:当系统功能扩展,需要由更多的软硬件模块来完成系统功能时,先前的数据交互路径会发生改变;或则,当系统中某个软硬件模块出现问题,系统需要暂时降级使用时,数据交互的路径也会发生改变。如果通信控制软件以一种固定的通信路径来进行设计,那么面对上述情况,软件必须修改。通过为通信控制软件提供通信路径编辑界面并设计相应的数据结构来保存通信路径,解决了数据交互的通用化问题。具体实现方法如下:

1) 设计一通信路径编辑界面,如图 3 所示,供用户编辑路径信息,并通过界面信息填充后续介绍的两张表;



图3 通信路径编辑界面

2) 设计统一的数据结构体 DATASTRUCT,存放交互的数据;

```

typedef struct DATASTRUCT
{
    unsigned char SourceAddr;
    //源 ID, 发送数据时,即为模块自身代号
  
```

```

unsigned char DataType;
//数据类型,即相互间约定好的功能代号,代表一个特定的操作
short Data[8];           // 传送的数据
};

```

3) 在通信控制软件中设计两张表,来索引数据的转发目的地及其写数据函数指针;

表1 特定数据的目的地址存放表

源地址	源数据类型	目的个数	目的地址串
-----	-------	------	-------

相应的数据结构为:

```

#define SOURCE_NUM 256           //源 ID 最大值
#define TYPE_NUM 100            //数据类型最大值
#define DEST_WRITE_NUM 4
//同一源地址、数据类型的数据需要发送到目的个数
struct TABLE_1
{
    unsigned char DestNum;           //目的个数
    unsigned char DestAddr[ DEST_WRITE_NUM]; //目的 ID 串
} Table1[ SOURCE_NUM][ TYPE_NUM];

```

表2 特定目的地址的写函数指针存放表

目的地址	写数据函数指针
------	---------

相应的数据结构为:

```

#define DEST_NUM 256
//目的地 ID 最大值
typedef int ( * PFUN_WRITE)( DATASTRUCT trans_data);
//统一的写函数指针类型
struct WRITE_TABLE_2
{
    PFUN_WRITE DeviceWrite;           //目的写数据函数指针
} WriteTable2[ DEST_NUM];

```

4) 数据交互过程如图4所示。

5) 通用化体现在:(1)用户可利用通信路径编辑界面更改通信路径,通信程序自动更新 Table1 及 WriteTable2;(2)当系统需要降级使用时,通信程序也只需要自动更新 Table1 及 WriteTable2 即可。

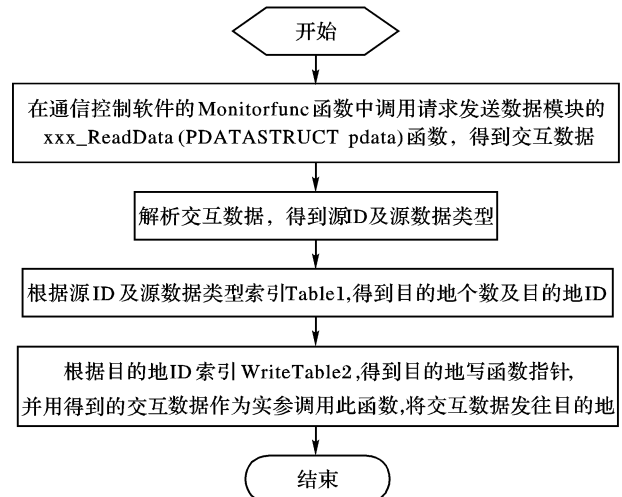


图4 数据交互过程

### 3 结语

控制系统中的通信程序是系统正常运行时数据交互的桥梁,是控制系统的关键部分之一,设计一个通用化的控制系统通信程序可以更加方便的维护和升级系统。本文提出了一种控制系统通信程序的通用化设计框架结构,此软件结构解决了控制系统因升级换代,模块更换必须修改通信程序的问题,为控制系统的构建提供了一种很好的软件设计方案。

#### 参考文献:

- [1] JURGEN J, PETER N. Switched ethernet for factory communication [J]. IEEE Symposium on Emerging Technologies and Factory Automation, 2001, 1: 205 - 212.
- [2] 梅格, 沈璞. 关于总线国际标准 IEC61158 的研究报告[J]. 仪器仪表标准化与计量, 2003, (2): 30 - 34.
- [3] 张海藩. 软件工程导论 [M]. 北京: 清华大学出版社, 1998.
- [4] RICHTER J. Windows 核心编程 [M]. 杨建华, 张焕生, 侯丽坤, 译. 北京: 机械工业出版社, 2000.
- [5] 朱友芹. 新编 Windows API 参考大全 [M]. 北京: 电子工业出版社, 2000.
- [6] 阳宪惠. 现场总线技术及其应用 [M]. 北京: 清华大学出版社, 1999.

(上接第452页)

判断,若剩余带宽不足 20%,则拒绝分配请求。此时广域 DNS 对请求的准入概率为:

$$RA = 1 - \bigcup_{i=1}^K P(R_{q_i} \geq 0.8b_i) \quad (7)$$

其中  $R_{q_i}$  和  $b_i$  分别是局域集群  $i$  占用的带宽和其基本带宽。图 2(d) 是接入概率的统计情况。

### 4 结语

由于广域 Web 集群系统的复杂性及其应用网络环境的多样性,通过仿真模拟进行对其系统性能分析是一种切实可行的方法。本文根据广域集群系统的要求建立功能模型,设计实现了广域集群的仿真平台软件。仿真实例表明,该仿真平台软件能够有效地分析 Web 集群系统的性能,为其设计和评估提供可靠的依据。下一步的工作我们还将对该广域 Web 集群仿真平台进行深入加工,比如改进 Web 服务器结点的排队论模型。

#### 参考文献:

- [1] SCHROEDER T, GODDARD S, et al. Scalable Web server cluste-

ring technologies[J]. IEEE Network, 2000, 14(3): 38 - 45.

- [2] 林闯, 单志广, 任丰原. 计算机网络的服务质量 [M]. 北京: 清华大学出版社, 2004.
- [3] 单志广, 戴琼海, 林闯, 等. Web 请求分配和选择的综合方案与性能分析[J]. 软件学报. 2001, 12(3): 355 - 366.
- [4] ARLITT ME, FRIEDRICH R, JIN T. Workload characterization of a Web proxy in a cable modem environment[J]. ACM Performance Evaluation Review, 1999, 27(2): 25 - 36.
- [5] PITKOW JE. Summary of WWW characterizations[J]. World Wide Web, 1999, (2)2: 3 - 13.
- [6] THOMPSON K, MILLER GL, WILDER R. Wide-area Internet traffic patterns and characteristics[J]. IEEE Network, 1997, 6(11): 10 - 23.
- [7] HEIDEMANN J, OBRACZKA K, TOUCH J. Modeling the performance of HTTP over several transport protocols[J]. IEEE Trans. on Networking, 1997, 5(5): 616 - 630.
- [8] FIELDING R, GETTYS J, MOGUAL J, et al. RFC 2068: Hypertext transfer protocol - HTTP/1.1[S]. 1997.