

基于 ERTEC200P-2 芯片的 PROFINET IO 设备开发

贾鹏飞^{1,2}, 贺永鹏^{1,2}, 杨建新^{1,2}, 张青^{1,2}, 伍闻达^{1,2}

(1. 天津电气科学研究院有限公司, 天津 300180;

2. 电气传动国家工程中心, 天津 300180)

摘要:工业通信技术是实现工业设备组网控制的有效方式,随着近些年以太网技术的发展,一种基于以太网的上层应用技术 PROFINET 快速发展起来,但是由于下层设备不具备高速的网络接口,需要将高速的通信协议中控制字和状态字转换为能被下层设备轻松解析的协议,进而将设备接入高速以太网。方案采用了 ERTEC200P-2 通信协议芯片进行外部以太网高速的数据交互,下层通过 UART 串行接口接入到变频器,该方案通过了试验验证,并且进行了现场验证,取得了良好的效果。

关键词:PROFINET 技术;ERTEC200P-2 芯片;工业通信

中图分类号:TM461 **文献标识码:**A **DOI:**10.19457/j.1001-2095.dqed25240

Development of PROFINET IO Device Based on ERTEC200P-2 Chip

JIA Pengfei^{1,2}, HE Yongpeng^{1,2}, YANG Jianxin^{1,2}, ZHANG Qing^{1,2}, WU Wenda^{1,2}

(1. Tianjin Research Institute of Electric Science Co., Ltd., Tianjin 300180, China;

2. National Engineering Research Center of Electric Drive, Tianjin 300180, China)

Abstract: Industrial communication technology is an effective way to achieve network control of industrial equipment. With the development of Ethernet technology in recent years, a top layer application technology based on Ethernet, PROFINET has rapidly developed. However, due to the lack of high-speed network interfaces in the lower layer devices, it is necessary to convert the control and status words in the high-speed communication protocol into a protocol that can be easily parsed by the lower layer devices, and then connect the devices to high-speed Ethernet. This scheme adopted the ERTEC200P-2 communication protocol chip for high-speed data exchange with external Ethernet, and the lower layer was connected to the frequency converter through the UART serial interface. The scheme has passed experimental verification and on-site verification, achieved good results

Key words: PROFINET technology; ERTEC200P-2 chip; industrial communication

在中国 2030 年前实现碳达峰的背景下,如何推动工业技术绿色转型并以更加高效姿态开展工业制造成为了当今话题。自动化、智能化成为解决上述问题的关键,但是这些问题本质是如何高效地将各个设备集成在一起,实现各种信息的统一。目前主要分为有线通信和无线通信,其中有线通信占据了工业通信半壁江山,主要有 CANOPEN, ENETHERCAT, PROFIBUS, PROFINET, MODBUS 等,这些技术在各自领域都发挥着重要作用。

高速度、高稳定性、复杂网络拓扑使 EN-

THERCAT, PROFINET 脱颖而出,以上两种通信技术都以工业以太网硬件为基础,具备以太网技术优点,又有各自应用层特点。

ENETHERCAT, PROFINET 依靠以太网协议进行开发。它们利用了以太网的高带宽、灵活性和普及性,同时通过实时性和可靠性的增强,满足了工业自动化领域对数据传输的严格要求。这两种协议都支持多种设备之间的通信。

本文主要研究 PROFINET IO 设备开发。PROFINET 是一种基于开放以太网自动化通信网络,借助以太网高速、稳定硬件基础,使 PROFI-

基金项目:中国机械工业集团有限公司重大科技专项(ZDZX2021-6)

作者简介:贾鹏飞(1991—),男,硕士,工程师,Email:jiapengfei@me.com

NET具有更强的数据吞吐量^[1]。PROFINET是一个完整的通信接口解决方案,在自动化领域中覆盖了很多应用,包括变频器、传感器、人机接口等,PROFINET将现场多种设备通过网络总线系统无缝集成,成为了当今自动化领域的主要通信网络协议,并且随着科技发展,占有率持续攀升^[2]。

目前,PROFINET开发有多种方式,如基于FPGA开发、Linux开发等。基于ERTEC200P-2的PROFINET IO设备接口技术进行开发,主要包括硬件设计、软件设计、联机测试等技术方案。

ERTEC200P-2芯片为ARM9内核,具备高速数据处理能力,具备双网口管理能力,通过eCos操作系统和PNIO协议栈协作,可快速实现PROFINET IO设备开发并应用。

1 PROFINET和ERTEC200P-2

1.1 PROFINET协议简介

PROFINET基于标准以太网技术进行开发,是工业实时通信的一种。PROFINET中国标准为GB/T 25105.1~3—2014。

PROFINET硬件传输线为全双工双绞线,采用自协商方式进行,通信速率为100 Mb/s,一根总线即可实现多种设备连接,提高了设备的可靠性^[3]。由于PROFINET硬件支持硬件协议自动转发,所以不需要任何交换机即可实现网络拓扑的自由扩展。由于PROFINET具备设备自动诊断和网络自动诊断功能,提高了多种设备间的通信效率,降低了设备的总体故障率,提高了总体的效率,为企业带来了更好的经济效益。

PROFINET为了达到实时响应设备之间指令的目的,采用了独立网络链路进行数据传输,无需借用TCP/IP通道即可完成数据实时交互,为多设备间同步提供了可能。PROFINET数据通路模型如图1所示。

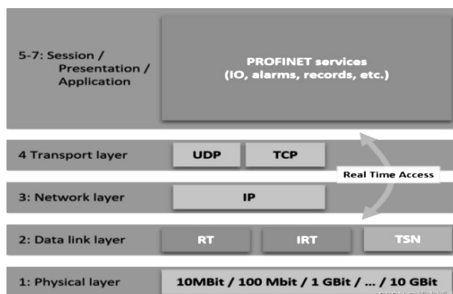


图1 PROFINET数据通路模型

Fig.1 Diagram of PROFINET data channel

根据数据通路模型,PROFINET数据通道主

要分为以下几类:

1)标准通道数据通信^[3]。标准通道数据为设备的参数化配置、诊断数据、设备信息等。此类通信基于TCP/IP,称为非实时(non real-time, NRT)数据通信。

2)实时(real-time, RT)数据通信^[4]。RT数据通信是基于固定循环周期进行的数据传输,通信周期长短受制于网络速度,一般为数百 μ s级别。这种周期通信在自动化领域应用较为广泛。RT通信基于TCP/IP第二层协议开发并进行部分优化,提供了更好的数据处理时间。在设备应用中,报警状态一般通过此协议进行周期传输。

3)同步实时(isochronous real-time, IRT)通信。IRT通信技术的存在是为了保证关键数据能按照约定时间到达,响应时间一般控制在50 μ s以内,并且数据会根据设备距离主站的远近进行时钟动态补偿,保证多台设备间控制同步。常见应用为伺服驱动器、多轴并联驱动器等。

1.2 ERTEC200P-2芯片基本介绍

ERTEC200P-2芯片是ERTEC200P的升级版,解决了部分硬件问题,同时对部分参数进行了优化。

ERTEC200P-2是西门子研发的一种PROFINET协议栈处理芯片,该芯片可以支持双百兆网络通信,内部自带硬件转发IP核,双网口数据传输完全通过硬件进行,无需软件干预,保证了数据流的实时性。

ERTEC200P-2内核主频250 MHz,型号为ARM926EJ,具备裸机运行能力,也可以增加操作系统进行任务调度。芯片外设具备并口通信能力,支持16位数据单周期访问,方便用户扩展芯片上内存,并且芯片外部总线可以直接访问内存通信数据,方便数据扩展,保证了数据流的带宽。

ERTEC200P-2内嵌了PROFINET硬件IP核,用于实现超短延时数据转发。PROFINET通信为了降低软件负载,硬件实现了自动数据帧封装,为开发IRT等实时通信提供了可能。并且在数据通信时,ERTEC200P-2还可以随时进行TCP/IP数据服务,方便了整个设备的其他功能扩展。

1.3 PROFINET IO协议栈

PROFINET IO协议栈为西门子推出的完整PROFINET IO协议栈,支持完整的PROFINET协议栈规范,方便了用户的开发。开发用户无需关心内部的通信协议以及流程如何在芯片内部流

动和封装,即可轻松实现PROFINET IO设备开发,并且根据自己的外部接口开发内部协议,加快产品的开发与上市时间。

PROFINET IO设备主要实现了主站与从站之间的周期与非周期数据交互;处理诊断和协议栈中断;通信模块的动态增加与删除;分配以太网地址设备名称;处理设备之间的冲突,屏蔽了复杂打包拆包算法。

2 PROFINET IO 硬件设计

本文基于ERTEC200P-2构建PROFINET IO设备开发方案。系统主要组成为ERTEC200P-2 PROFINET IO设备和280039C DSP 控制板,硬件结构图如图2所示。

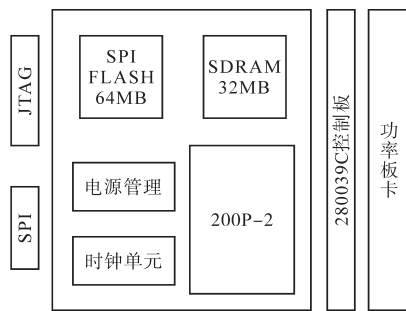


图2 硬件结构图

Fig.2 Diagram of hardware structure

图2中,主要硬件单元为ERTEC200P-2控制芯片,考虑到ERTEC200P-2芯片内存容量较低,外扩SDRAM动态存储器进行数据的存储,保证协议栈和用户的应用程序足够运行。

JTAG接口为调试接口,方便在线调试协议栈,增加程序开发的便利性。串口为外部通信口,方便ERTEC200P-2芯片与280039C芯片进行控制字和协议字的通信。网口为RJ45网络接口,为2路PN网络线接入口。电源管理部分主要提供了SDRAM电源、ERTEC200P-2电源。由于ERTEC200P-2内部没有FLASH,增设了外部FLASH单元为ERTEC200P-2协议栈程序提供存储。

3 PROFINET IO 软件设计

3.1 PROFINET 软件结构和任务简介

ERTEC200P-2基于西门子PN协议栈进行开发,协议栈主要组成部分为操作系统、任务调度单元、串口通信解析单元、网络协议栈处理单元、通信监测单元、硬件IO单元、网络协议处理单元。

系统软件结构通过eCos操作系统进行了整体调度。eCos操作系统是一种微内核实时嵌入式操作系统,占用内存空间小,具有多种优先级抢占模式,提供了完整的内存管理机制,简化了内存管理,提高了内存的利用效率。PN协议栈主要通过此操作系统来完成,通过在芯片初始化时创建不同的其他任务,让eCos操作系统完成各个任务之间的抢占,保证各个任务按约定的周期执行。软件结构图如图3所示。

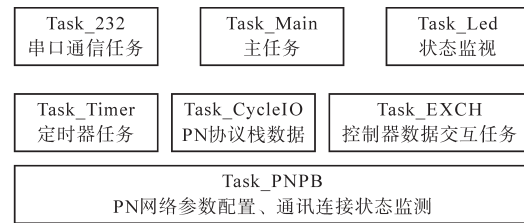


图3 软件结构图

Fig.3 Diagram of software structure

图3中,主要功能由以下几部分组成:

1)Task_Main任务。这是芯片初始化后第一次调用的任务,在此任务中,主要进行各个任务的创建,初始化PN协议栈,进行协议栈的设备检查和协议栈初始工作状态确认。

2)Task_Led任务^[5]。此任务主要是通过检查协议栈的状态进行相关外部LED的指示,帮助用户快速判断通信状态。此任务通过BSP板级支持包实现外部IO控制任务。

3)Task_232任务。此任务是负责与280039C通信,将PN网络控制字和状态字相关信息进行打包打包,发送到DSP进行电机控制,并实时返回电机控制侧状态信息供PLC调用。

4)Task_CycleIO任务。此任务通过200P-2的硬件网口中断触发,每次受到触发,要及时地响应来自PN网络的数据,将网络数据解压,送到上层供串口任务获取。

5)Task_PNPB任务。此任务为协议栈内部任务,用于事件处理。主要是监测通信连接或断开时将内存数据临时更新或删除。

6)Task_EXCH任务。此任务主要功能是根据设备的组态模型进行数据的交换,主要负责和Task_CycleIO任务交互。

3.2 数据流动与存取模型

数据流动模型显示了各层数据流之间的关系,如图4所示。

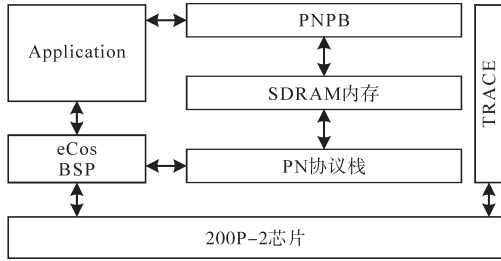


图4 数据流动模型

Fig.4 Diagram of data flow model

数据存取模型主要用来表示上位机配置和PN协议栈内部对应关系,如图5所示。

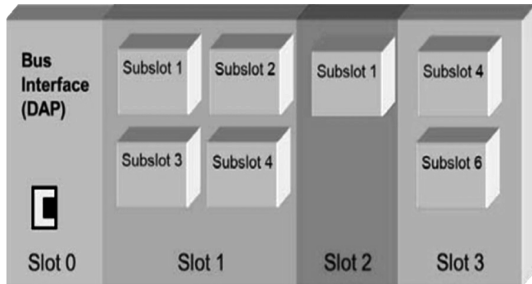


图5 数据存取模型

Fig.5 Diagram of data access model

PROFINET设备的基本数据模型是槽(Slot),所有的设备均是由一个或多个插槽组成,每个插槽均由若干子插槽(Sub-slot)组成,其中插槽0固定作为总线接口,也叫做设备访问点。插槽1及之后的插槽作为输入输出,提供周期数据访问或非周期数据访问以及报警、诊断功能。在GSD-ML文件中,描述了每个槽上可用的模块,以及每个模块包含的可用子模块。在组态软件(TIA 博途)中根据实际的设备配置,将模块、子模块分配到槽和子槽。同时,PLC 会为每个模块、子模块的IO数据在过程映像区内分配对应的地址。输入数据对应I过程映像区,输出数据对应Q映像区。基于PROFINET通信指定的OB块将从PN设备接收的输入数据更新到I区,将Q区的数据发送至PN设备建立输出。

PNIO用PNIO_initiate_data_read接收PLC控制器下发数据,将根据PNIO缓冲区数据对应关系放到槽和子槽中。同时,调用PNIO_initiate_data_write将槽和子槽中数据更新到PNIO缓冲区,上传给PLC控制器。

3.3 串口通信协议

PNIO设备通过串口与280039C进行通信,通信速率选择460 800波特率,加快内部传输速度,提高整个控制系统实时性。

通信数据帧主要由帧头、功能码、数据长度、

数据(PZD+PKW)及CRC校验组成,通信数据格式如图6所示。

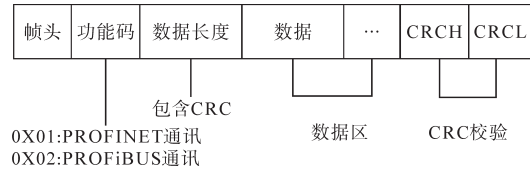


图6 通信数据格式

Fig.6 Communication data format

4 PROFINET通信测试

为了验证PROFINET IO设备与底层的通信能力,通过西门子PLC S7-1500搭建程序进行底层控制,验证整个控制器的可靠性^[6-7],原理如图7所示。

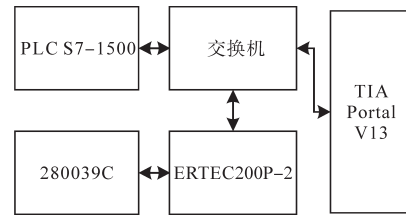


图7 测试环境结构图

Fig.7 Diagram of test environment

如图7所示,测试由PLC S7-1500、交换机、TIA PortalV13及PROFINET IO接口设备(ERTEC 200P-2)和280039C控制板组成。

首先,需要将PLC S7-1500与交换机相连,然后通过交换机网口分别连接PC机和ERTEC200P-2控制板,通过TIA Portal设置设备组态,将PLC S7-1500与ERTEC200P-2组态。将PLC S7-1500的IP地址设置为192.168.0.1,PRFINET IO设备设置为192.168.0.2,设备通信模块设置成12B输入,12B输出,地址为IW0-11,QW0-11,如图8所示。

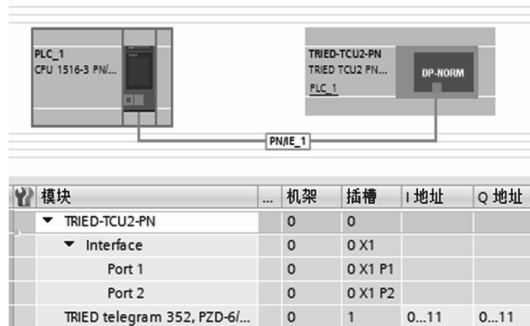


图8 设备组态图

Fig.8 Diagram of device configuration

搭建PLC S7-1500通信程序,程序主要功能为通过定时器进行周期计数,计数器数值通过QW0进行外部传输,实现QW0地址数据逐渐增

加。IW0数据通过寄存器进行临时保存,并与上一周期发出数据QW0比较,验证整个数据是否正确。然后,继续发送下一次计数值,验证整个PROFINET环路是否能正确收发。为了方便观察数据准确度,通过DSP 280039C将接收到的PN数据通过串口打印到串口调试助手,观察效果。如图9所示,PROFINET每周数据相比上一个周期每次增加1,说明数据发送正常。计数器增加之后PLC S7-1500还能继续发送下一帧数据,说明PLC S7-1500收到了上次的数据,并且校验正确。通过数据帧格式查看每个位置数据,均为正确的。

F2 01 14 37	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 8E A2
F2 01 14 38	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C1 A6
F2 01 14 39	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 A6
F2 01 14 3A	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 43 A7
F2 01 14 3B	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 82 A7
F2 01 14 3C	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C5 A5
F2 01 14 3D	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04 A5
F2 01 14 3E	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 47 A4
F2 01 14 3F	44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 86 A4

图9 数据包分析
Fig.9 Packet analysis

5 改进与提高

以上虽然实现了通信功能,但是考虑到现场应用中并不是每次所有通信字长都能被使用,但如果程序始终按照固定字长去解析,会导致DSP负载加重,响应速度降低。

为了解决上述问题,需要在PN协议栈中判定用户组态字长,将通信格式传输到DSP。西门子在PN协议栈定义IO模型如图10所示。

```
#define IO_MODULE_TEL1 0x41// Standard telegram 1, PZD2/2
#define IO_MODULE_TEL20 0x42// Standard telegram 20, PZD6/2
#define IO_MODULE_TEL351 0x43// Tried telegram 351, PZD4/4
#define IO_MODULE_TEL352 0x44// Tried telegram 352, PZD6/6
#define IO_MODULE_TEL353 0x45// Tried telegram 353, PKW+PZD4/4
#define IO_MODULE_TEL354 0x46// Tried telegram 354, PKW+PZD6/6
#define IO_MODULE_TEL355 0x47// Tried telegram 355, PKW+PZD28/28
#define IO_MODULE_TEL997 0x48// Free telegram, PZD 8/8
```

图10 协议栈IO模型
Fig.10 Protocol stack IO model

在通信协议中新增IO模型控制位,通信数据格式如图11所示。

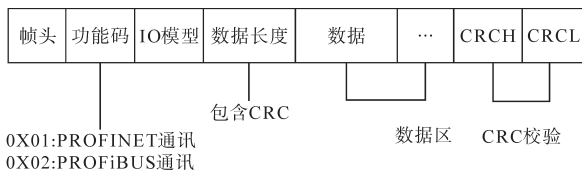


图11 通信数据格式
Fig.11 Communication data format

还需在PN协议栈中新增组态完成标志位,通过PN协议栈中的组态完成标志位将组态信息赋值给IO模型控制位。在DSP端新增动态内存申请与动态字长解析程序,根据IO模型进行相应

解析,协议解析程序如图12所示。

```
if(Pn_Data.pn_Buf_R[Pn_Data.recieve_pn_out] == 0xf2) //查找帧头
{
    crc_pos_1 = Pn_Data.recieve_pn_out;//拿到帧头
    Pn_Data.recieve_pn_out++;//移位,检查板卡类型
    Pn_Data.recieve_pn_out = Max_Z(Pn_Data.recieve_pn_out,PN_BUF_LENGTH);
    Board_type = Pn_Data.pn_Buf_R[Pn_Data.recieve_pn_out];//查看板卡类型

    Pn_Data.recieve_pn_out++;//移位,帧长
    Pn_Data.recieve_pn_out = Max_Z(Pn_Data.recieve_pn_out,PN_BUF_LENGTH);
    legth_t = Pn_Data.pn_Buf_R[Pn_Data.recieve_pn_out];//查看帧长
    recieve_pos_t = Pn_Data.recieve_pn_out+1;
    if(recieve_pos_t>=PN_BUF_LENGTH)
    {
        recieve_pos_t = recieve_pos_t- PN_BUF_LENGTH;
    }
    bag = Pn_Data.pn_Buf_R[recieve_pos_t];//获取包号
    recieve_pos_t = Pn_Data.recieve_pn_out+2;
    if(recieve_pos_t>=PN_BUF_LENGTH)
    {
        recieve_pos_t = recieve_pos_t- PN_BUF_LENGTH;
    }
    if(Pn_Data.pn_Buf_R[recieve_pos_t]==0x41)//2
    {
        Frame_length = 12;
    }
    else if(Pn_Data.pn_Buf_R[recieve_pos_t]==0x43)//4
    {
        Frame_length = 16;
    }
}
```

图12 协议解析程序
Fig.12 Protocol parser

通过上述改进,提高了DSP内存利用率,通过CCS编译软件观察改进前、后内存使用情况,结果为内存使用率降低38%。通过改进,也提高了代码执行效率,降低了DSP负载,为其他算法程序执行提供了时间。改进前、后CPU负载降低5%左右。

上述改进已经在水刺法同步机项目实现了应用,并取得良好效果。

6 结论

本文介绍了基于ERTEC200P-2的PROFINET IO设备接口的开发方案,包括硬件设计、软件实现和相应的通信功能测试。本设计方案已经通过现场应用,并委托中国PROFIBUS & PROFINET协会(PI-China)进行检测并顺利通过认证测试,充分验证了方案设计的合理性,对工业通信的设计具有很强的指导意义。

参考文献

- [1] RIBEIRO Fabio Mallet, COSTA Tulio Sousa, BARATELLA Alexandre, et al. Comparative analysis of industrial Ethernet networks: Profinet, Ethernet/IP and HSE[J]. International Journal of Innovative Computing, Information and Control, 2014, 10 (5):1931-1948.
- [2] 陈雷. 基于PROFINET通信协议的IO设备研究[D]. 合肥:合肥工业大学, 2016.
CHEN Lei. Research on IO based on PROFINET communication protocol[D]. Hefei: Hefei Polytechnic University, 2016.
- [3] 张国栋,王有春. PROFINET的实时性及其协议分析[J]. 计算机测量与控制, 2017, 25(3):187-190.

(下转第40页)