

文章编号：1005-8451(2011)07-0050-03

用于列车网络的司机显示单元的系统设计与实现

肖阳俊¹, 吴旋¹, 王爱武¹, 李金波²

(1. 南车南京浦镇车辆有限公司, 南京 210031;
2. 中国铁道科学研究院 电子计算技术研究所, 北京 100081)

摘要：地铁列车司机显示单元(DDU: Dirver Display Unit)作为列车网络的重要部分, 是完成人机对话和系统信息显示的重要系统。为驾驶员的操作及地铁车辆安全运行提供信息保证。本设计选择X86架构计算机作为DDU单元的硬件平台, 嵌入Linux操作系统, 构建了基于CANopen通信协议的地铁列车司机显示单元。本文详细介绍了DDU的系统原理和软件架构设计。

关键词：司机显示单元; 列车网络; 嵌入式Linux; CANopen

中图分类号：U28-TP39 **文献标识码：**A

Design and implementation of Driver Display Unit for train network

XIAO Yang-jun¹, WU Xuan¹, WANG Ai-wu¹, LI Jin-bo²

(1.CSR NanJing PUZHEN Rolling Stock CO.,LTD, Nanjing 210031, China;

2. Institute of Computing Technology, China Academy of Railway Sciences, Beijing 100081, China)

Abstract: The Driver Display Unit(DDU) was as an important part of the train network to implement the human-machine interface and information display. The DDU provided the information for the operation of driver to assure the safety of the subway train. This paper chose the X86-CPU base hardware as the hardware platform of DDU and Embedded Linux Operation System to build the DDU based on CANopen protocol, introduced the system principle and the software architecture design of DDU in detail.

Key words: Driver Display Unit(DDU); train network; Embedded Linux; CANopen

地铁列车司机显示单元DDU是地铁上完成人机对话和系统信息显示的重要单元, 为驾驶员的操作及地铁安全运行提供信息保证。DDU是列车网络TIMS(列车综合管理系统)的一部分^[1], 它负责监测列车所有子系统的状态并进行部分非安全性的控制, 采用CANopen协议进行网络通讯, 整个列车网络采用两路CAN总线(CAN总线1, CAN总线2), 将各子系统的控制单元合理地分配到这两路总线上, 每路CAN总线在单元间设一个网桥或中继器(Repeater)进行连接, 沟通两个单元之间的通讯和数据交换。TIMS中的主控设备(VTCU)负责收集所有子系统的状态信息、过程数据和故障信息并进行分析。同时VTCU将重要的信息发送至DDU系统, 显示列车各种信息。这些信息包括: 列车状态信息、各种故障信息等。列车司机还可以通过DDU对列车的某些子系统进行设置和控制。如空调系统、PIS(Passenger Information System, 乘客信息系统)等。

收稿日期: 2010-09-19

作者简介: 肖阳俊, 工程师; 吴旋, 工程师。

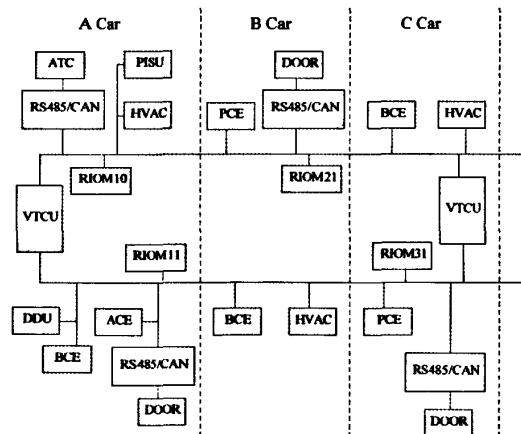


图1 TIMS系统网络拓扑图

1 系统原理

(1) VTCU的消息通过CAN网络传送到DDU系统硬件CAN适配卡。CAN数据经过CAN硬件设备的物理层和数据链路层的接受/发送处理过程。这个过程包括数据分析、校验过程, 保证CAN数据的可靠性。

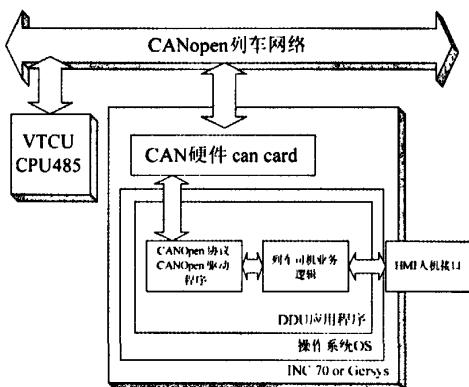


图2 TIMS网络与DDU子系统间通讯基本原理图

(2) DDU 应用程序将处理后的 CAN 数据, 根据 CANopen 协议标准, 封包为 CANopen 数据包。即标准的 CANopen 数据(包括 SDO, PDO 等等)。

(3) DDU 应用程序、根据 VTCU 与 DDU 之间的接口文件, 处理 PDO 数据包。并将数据包的含义还原为 DDU 的业务逻辑, 显示在 HMI 显示屏上。

(4) 司机通过 DDU 操作其他子系统的数据流程与该过程类似。最终, 发送至 VTCU 的 CAN 数据被解析为 CANopen 数据。然后根据数据包的特定含义, 执行特定的操作。

表1 目前几种列车网络的比较

列车网络	标准	特点
TCN	IEC 61375-1	列车级 WTB、车辆级 MVB、设备级控制冗余总线结构
WorldFIP	IEEE61158 EN50170	三层结构协议、曼切斯特编码
LonWorks	IEEE1473 ISO11898	七层结构协议、多种网络拓扑
CAN	CiA DS 301 v 4.02 open IEC 61375-3	串行总线、三层结构(应用层为 CAN 协议)、短帧结构、多主竞争
Ethernet	IEC 61375-3	高速、开放式、未来列车网络发展方向

2 系统架构

DDU 系统的实现主要是通过软件实现的方法。DDU 应用程序运行在特定的嵌入式操作系统 (EOS: Embedded Operation System) 平台上。嵌入式操作系统主要负责与硬件的底层打交道, 包括与计算机硬件的各种子设备通讯, 如磁盘、网卡等。统一调配系统的资源。CAN 网络适配卡的资源管理也由 EOS 负责, 因此, 发送至 DDU 的 CAN 数据经由 EOS 处理, 实现了物理层和数据链路层

通讯。处理后的数据将发送至特定的应用程序。

图 3 的第 2, 3 部分是 CANopen 通讯协议的实现部分。其中, 通讯子协议实现了标准的 CANopen 协议, DS301 V4。设备子协议确定了 DDU 设备的重要特性。这些特性决定了 DDU 在网络通讯的作用。DDU 应用程序根据通讯子协议和设备子协议, 将标准的 CAN 数据转化为 CANopen 数据包。

层次关系的第 4 部分是 DDU 特定的功能逻辑。功能逻辑一般在 DDU 系统与网络的接口文件中定义。如: DDU 与 VTCU 的接口文件 (The Interface between VTCU and DDU)。

在第 5 部分, 应用程序将功能逻辑转化为图形、图标、等可视化符号展现在物理的显示屏上。

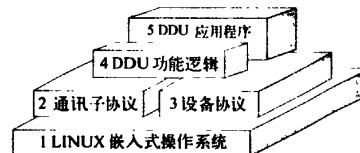


图3 系统软件设计架构示意图

3 通讯协议层 CANopen 设计实现

图 4 显示了 CANopen 通讯协议和设备协议之间的关系^[2]。

CAN 数据转变为 CANopen 数据后, 被解析为 PDO 或 SDO 数据包。

通过定义设备字典 (object dictionary), 将 DDU 所有的网络行为和状态信息都定义好。应用程序通过对对象字典来查询 PDO 和 SDO 的含义。

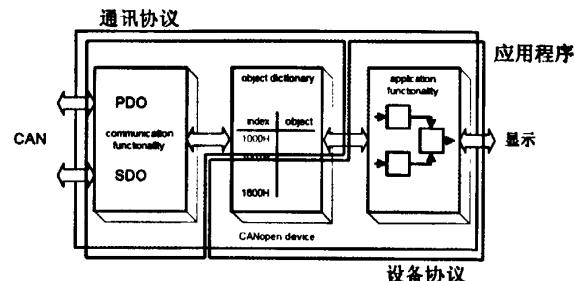


图4 DDU 通讯协议与设备 CANopen 协议之间的关系图

4 硬件环境的构建

硬件采用 Pixy 公司的 INC70 显示屏。为适应轨道交通特殊的要求, 设备具备下列特点:

(1) 工作温度: $-30^{\circ}\text{C} \sim 70^{\circ}\text{C}$, (2) 超强的防震动, 抗冲击能力; (3) 适应铁路交通的宽工作电压范围 $24\text{ vDC} \sim 110\text{ vDC}$; (4) 2*Compact Flash存储卡, 适应密集存储; (5) 标准扩展接口; (6) 按键或触摸屏; (7) 宽视角, 高亮度的TFT-LCD; (8) 适应夜间操作的背光按键。

INC70的CPU为intel CELERON M 600 MHz, 主板集成了RS232, RS422/RS485, Ethernet, MVB, CAN, digital I/O机车常用接口, INC满足EN50155 / EN50121-3-2 / EN50126 / NF F16-101/102等常用铁路标准。

5 软件环境的构建

目前, 嵌入式操作系统的种类很多, 如VxWorks, Windows CE和Linux等^[3]。表2列出了这几种嵌入式操作系统的优缺点, 在这些操作系统中, Linux是发展最快, 应用最广泛的。

表2 目前常用嵌入式操作系统比较

操作系统	优点	缺点
Windows XP embedd	兼容性好, 功能多	内核太大, 收费, 实时性可靠性不高, 对硬件要求高
Windows CE	开发难度低, 图形界面优秀, 应用集成度高	收费, 实时性可靠性不高
Linux	开源代码, 内核小, 效率高, 可裁剪, 多任务	开发难度高
Vxwork	可靠性、实时性高, 可裁剪, 多任务	费用高

嵌入式Linux是按照嵌入式操作系统的需要而设计的一种小型操作系统, 由一个Kernel内核及一些根据需要进行定制的系统模块组成。Kernel一般只有几百kb左右即使加上其他必须的模块和应用程序, 所需的存储空间也很小。具备多任务、多进程的系统特征, 有些还具备实时性。一个小型的嵌入式Linux系统只需要引导程序、Linux微内核、初始化进程3个基本元素。

由于嵌入式系统资源有限, 所以很难在嵌入式系统的硬件平台上进行软件开发。通常解决的方法是在通用计算机上编好程序, 通过交叉编译生成可在目标板运行的二进制代码, 然后再下载到目标板上的特定位置运行。需要交叉编译环境是嵌入式应用软件开发时的一个显著特点。所谓交叉编译环境是指编译、链接和调试嵌入式应用

软件的环境, 它与运行嵌入式应用软件的环境有所不同, 通常采用宿主机/目标机模式。

在宿主机中安装RedHat9.0操作系统, 并装有GCC编译器和Linux内核: linux-2.4.20版本, 同时建立交叉编译环境。目标机使用intel CELERON M 600 MHz芯片, 基于X86体系, 以CF(compact flash)卡为存储设备。这样在宿主机上开发的程序, 通过交叉编译以后, 就可以下载到目标机上运行。

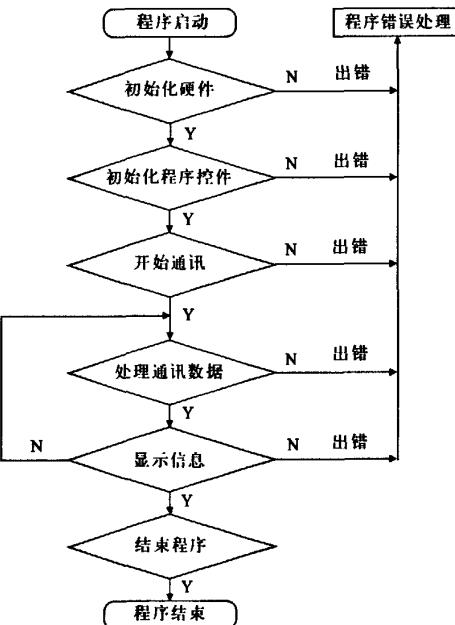


图5 DDU软件的主流程图

6 结束语

列车司机显示单元是列车网络的重要组成部分, 采用模块化、多层框架系统设计对于研制多用途、多功能、多接口的系统产品是十分有效。

参考文献:

- [1] 倪文波, 王雪梅.高速列车网络与控制技术[M].成都:西南交通大学出版社, 2008, 5.
- [2] 谷丰.基于Linux的机车显示系统平台的设计[J].机车电传动, 2005, 1.
- [3] 陈伟文, 凌玉华, 顾毅斌.CANopen总线协议在地铁通信网络中的应用[J].可编程控制器与工厂自动化, 2010, 3.