

CBTC 系统车载人机界面的设计与实现

郑 伟 王 超 陈宁宁

摘 要: 从 CBTC 系统车载 DMI 接口和功能需求入手, 采用跨平台的 Qt 开发环境, 完成了 Linux 系统下的车载人机界面的设计与开发, 并对软件流程和界面设计进行了详细描述。经过现场和实验室大量测试和不断修改, 界面内容丰富、运行稳定。

关键词: CBTC 系统; 车载人机界面; 界面设计

Abstract: Starting from the interface and function of vehicle DMI in CBTC system, a cross-platform Qt development environment is used for the design and development of vehicle human-machine interface in Linux system and the software flow and interface design are described in detail. Through lots of tests both in field and in laboratory and constant modifications, the contents of the interface become much richer and the program also can run more stably.

Key words: CBTC system; Vehicle human-machine interface; Interface design

DOI: 10.13879/j.issn1000-7458.2016-01.15448

近年来, 随着我国城市化进程的加快, 各大中型城市交通的压力越来越大, 地面交通越来越拥挤, 乘坐地铁、轻轨等城市轨道交通工具已成为市民出行的首选方式。同时, 面对密度、速度以及大客流快速增长的压力, 各大中型城市新建的地铁项目纷纷选择基于通信的列车控制系统 (CBTC)。

MTC-I 型 CBTC 系统是完全立足国产技术研发的功能、架构完整的 CBTC 系统。作为 CBTC 系统中车载控制子系统的一部分, 车载人机界面 (Driver-Machine Interface, 简称 DMI) 是实现司机与车载信号系统设备沟通的重要通道。它主要通过图像、声音、文字等方式, 向司机告知列车各种信息、状态, 提示司机执行相应的操作, 从而提高列车驾驶的可操作性。

DMI 系统采用基于 X86 架构的硬件平台, 以

及嵌入式 Linux 系统的软件平台, 并采用开源的图形界面库 Qt 开发车载人机界面应用软件。

1 DMI 功能设计

车载 DMI 主要向司机提供如下信息: ①到站及终点站信息; ②速度、目标速度、距离信息; ③列车运行模式、控制等级信息; ④车门、屏蔽门信息; ⑤列车牵引/制动信息; ⑥ATP/ATO 等设备状态信息; ⑦停站精度、跳停/扣车、折返信息; ⑧列车完整性、车辆段信息; ⑨司机号、目的地号、车次号。其他报警信号、文字消息或者时间信息, 也需要提供给司机。

同时, 司机还可以通过车载 DMI 输入司机号、目的地号、车次号, 以及完成显示屏工作参数调节功能, 如显示亮度和触摸屏校准等。此外, 维护人员还可以通过车载 DMI 进行软件参数配置、软件信息查看、在线日志查看等功能。DMI 功能流程如图 1 所示。

郑 伟: 中国铁道科学研究院通信信号研究所 助理研究员
100081 北京

王 超: 中国铁道科学研究院通信信号研究所 助理研究员
100081 北京

陈宁宁: 中国铁道科学研究院通信信号研究所 工程师 100081
北京

收稿日期: 2015-08-18



图1 功能流程图

2 DMI 接口设计

2.1 硬件接口

基于车载信号系统架构设计, 车载 DMI 只与同端信号设备进行数据交换。在列车的头端或尾端, 采用双网冗余设计, 2 个网络 A、B 设置成不同网段, 相互独立。车载 DMI 通过 2 个不同网段的以太网口, 分别连接到 2 个车载交换机上。正常工作时, 2 块以太网卡同时收发数据, 分别与 A 网、B 网上的其他信号设备进行通信, 从而实现双网冗余。其网络连接如图 2 所示。

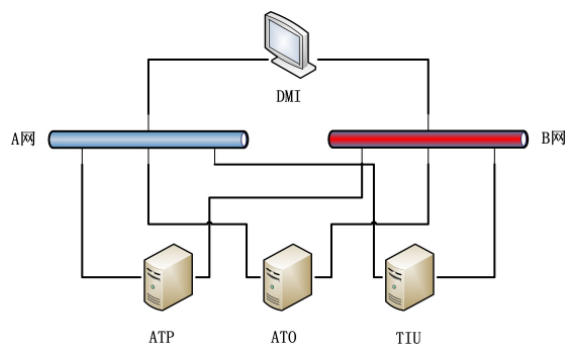


图2 DMI 网络冗余连接图

2.2 业务信息流

车载 DMI 主要与 ATP/ATO/TIU 进行数据通信。首先, 进行数据类型、数据长度等判断; 然后, 对数据进行循环冗余检验和帧序号判断。如果数据类型、长度等不符, 循环冗余检验错误, 无效帧序号等, 则该数据无效。下面分别介绍各接口的数据信息流。

1. 与 ATP 接口。车载 ATP 将状态报文周期性地发送到 DMI 上, 实现车载 ATP 状态数据的实时显示。

2. 与 ATO 接口。车载 ATO 将状态报文周期性地发送到 DMI 上, 实现车载 ATO 状态数据的实时显示; 断电时, DMI 还会接收 ATO 发出的断电保护帧。此外, 车载 ATO 将接收并处理司机通过 DMI 完成的设置数据 (如司机号等)。

3. 与 TIU 接口。车载 TIU 将对时报文周期性地发送到 DMI 上, 实现车载 DMI 时间与系统时间保持一致。

3 DMI 软件设计与界面实现

3.1 软件设计

1. 安装与配置设计。在现场首次安装 DMI 时, 必须由现场维护人员检查确认 DMI 的单、双端位置是否正确。软件启动后, 程序会初始化数据, 并读取存储 DMI 位置信息的文件, 来判断 DMI 处于单端还是双端。如果未读取正确的位置信息, 程序无法正确配置网卡信息, 需要维护人员重新配置位置信息。然后重启 DMI, 再次检查 DMI 位置信息。DMI 位置信息检查流程如图 3 所示。同时还需检查程序版本等信息。在操作过程中, 为防止司机误碰按键, 需要对命令进行二次确认; 维护人员和司机进行操作时, 需要输入相应口令进入操作界面。

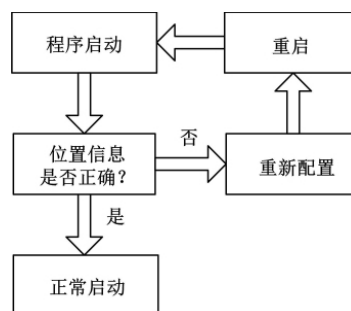


图3 DMI 位置检查流程图

2. 软件流程。DMI 软件是一个独立进程, 为了加快程序处理, 释放较长时间的资源和内存占用, 采用了多线程技术。按线程划分, DMI 软件可分为主线程、数据处理线程、对时线程。主线程主要完成程序初始化 (包括数据初始化、数据处理线程、对时线程的创建等), 通信数据接收, 信息周期显示, 系统断电保护, 司机/维护人员操作响应等功能。数据处理线程主要进行 ATP 及 ATO 周期性显示数据的有效性检查、通信判断; 对时线程负责完成系统对时功能, 由于只有满足对时条件才进行系统对时, 所以平时主线程将对时线程挂起, 在满足条件时执行该线程。DMI 软件流程如

图4所示。

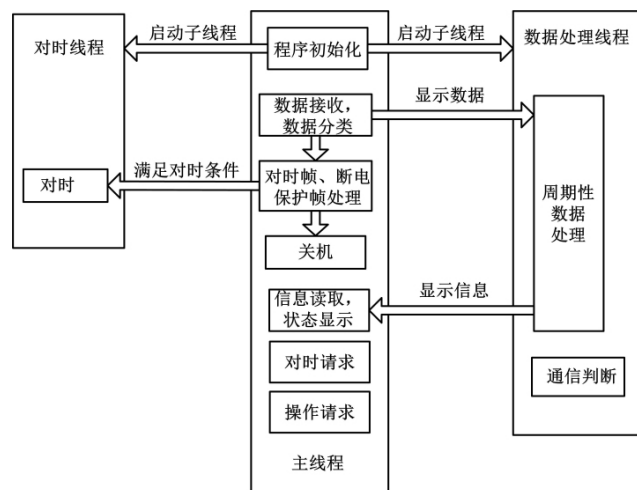


图4 DMI软件流程图

同时，在多线程间必须考虑数据的共享问题，需要对两线程间的接口数据进行加锁同步，当一方需要访问数据时，必须先加锁，访问完成后解锁。

3.2 界面设计与实现

车载DMI界面采用三级界面设计。主界面主要负责车载信息的实时显示，二级界面的进入；二级界面主要进行功能选择，司机号输入、目的地号输入、车次号输入、在线日志查询、亮度/触摸屏调节、单双端选择/软件信息；三级界面完成二级界面选择功能的具体操作。

1. 主界面布置。根据界面图形化、人性化和一致性的设计原则，并根据功能划分车载主界面的区域。其区域分布图如图5所示。

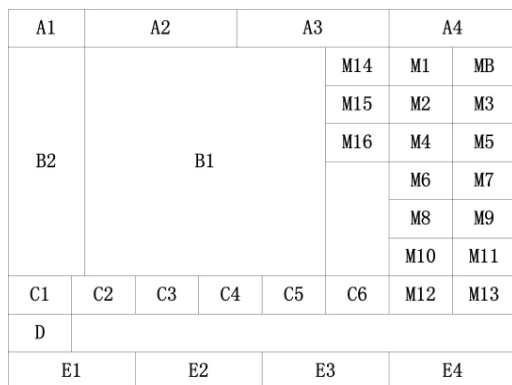


图5 车载界面分布图

A区：制动请求、到站信息、终点站信息、logo显示。

B区：速度、目标速度、目标距离等信息显示。

C区：列车驾驶辅助信息显示。

D区：司机确认信息提示。

E区：司机号、目的地号、车次号、系统时间显示。

M区：列车运行状态和驾驶信息显示。

2. 界面实现。主界面实际运行效果如图6所示。



图6 主界面

4 总结

结合MTC-I型CBTC车载系统自身特点和车载DMI功能，利用跨平台的Qt软件，完成了Linux系统下的车载人机界面开发。通过现场和试验室的大量测试，软件运行稳定，界面显示清晰、操作简洁。同时，还需要根据工程应用和系统需求，不断完善DMI功能，并不断提高界面的稳定性、友好性。

参考文献

- [1] 任亚飞. 基于嵌入式Linux的列控系统车载人机界面的实现[J]. 铁路计算机应用, 2005, 14(12).
- [2] 谢红晨. CTCS-300T列控车载人机界面的仿真研究[J]. 机电工程, 2014, 31(5).
- [3] 邓晓燕. CTCS-3级列控车载计算机操作系统的设计与实现[J]. 河北省科学院学报, 2011, 28(4).
- [4] 曾志伦. 车载设备人机界面DMI的设计与实现[J]. 中国水运, 2010, 10(12).
- [5] 刘超. 基于Linux Qt的列车人机界面设计与实现[J]. 计算机与现代化, 2013(4).

(责任编辑：温志红)