

# 绞线式列车总线原型节点的设计

徐晓松 谢维达 同济大学电气工程系(200331)

## Abstract

Wired Train Bus is a specification for train-level communication proposed by IEC. A framework of full function WTB node is proposed in this paper, and a prototype of WTB node is also implemented according to the framework. The link-layer of WTB specification, which is the core of WTB, is emphasized in the prototype, and essential function of WTB is implemented.

Keywords: IEC standard, Train Communication Network, Wired Train Bus, HDLC, fieldbus

## 摘要

绞线式列车总线(WTB)是 IEC 制订的用于列车级车载数据通信的总线标准。本文提出了一种 WTB 节点设计的框架方案,并设计实现了具备 WTB 基本功能的原型节点。该原型节点的设计重点放在 WTB 的核心部分——WTB 链路层的软硬件实现,实现了 WTB 的基本功能,并为进一步开发实用的 WTB 技术和产品作准备。

关键词: IEC 标准, 列车通信网络标准, TCN, 绞线式列车总线, WTB, HDLC, 现场总线

列车通信网络标准 (Train Communication Network, TCN), 即 IEC 61375-1, 已经在铁路行业受到了广泛的重视, 并正被逐步应用到各种新型的机车车辆上。TCN 标准定义了两种用于列车车载数据通信的现场总线: 绞线式列车总线 (Wire Train Bus, WTB) 和多功能车辆总线 (Multifunction Vehicle Bus, MVB)。WTB 被设计用于实现频繁改变编组的列车组中的各车辆间的互连, 具有自动适应列车编组的能力, 它可对经常改变编组的铁路列车、地铁列车和城市轨道交通车辆的控制、安全、运营和维护提供关键的数据通信支持。

## 1 WTB 简介

WTB 传输速率为 1Mbps, 传输介质为屏蔽双绞线, 通过手插式跨接电缆或自动连接器来实现车辆之间的互连, 最大直接通信距离 860m (相当于 22 节车辆的长度), 最多可连接 62 个节点。通常每车辆安装 1 个节点, 也可以没有或多于 1 个节点。为提高可靠性, WTB 采用了介质冗余。数据在两条介质上同时发送, 一个仲裁逻辑根据收发器提供的载波检测信号在两路接收信号间做出选择。

WTB 的介质访问控制方式为主-从控制方式, 即总线上唯一的主节点控制介质的访问。所有其它节点为从节点, 只在受到主节点请求时才能发送数

据。主节点将总线带宽以 25ms 为基本周期进行分配, 基本周期又被分为周期相和偶发相。周期相用于传输实时的周期性的过程数据, 每帧过程数据中可包含多个相同传输周期的变量。每个基本周期开始是周期相。在此期间, 主节点按预先生成的周期列表依次请求各发送过程数据的节点向总线广播其过程数据, 想要发送消息数据的节点还在其数据中发送一个请求标志, 主节点将记录此请求。基本周期中周期相之后的部分是偶发相, 在此期间主节点向请求发送消息数据的节点依次发送传送请求, 收到请求的节点将要发送的消息数据发向目的节点。这种介质访问控制方式可保证实时变量传输的确定性。实时变量的最大传输延时只与总线的基本周期有关, 而与总线负载情况无关。在过程数据和消息数据之间, 主节点还发送一些监控数据请求以监控总线状态。

WTB 最为显著的特色是它对列车编组结构的自适应能力<sup>[1]</sup>。每当列车编组改变时 (例如连挂或摘除车辆), WTB 各节点执行初运行过程, 该过程在电气上将各节点连接起来, 并给每个节点分配连续地址。为了实现初运行, 每个节点包含两个 HDLC 通道, 每个通道对应 1 个方向。初始时, 连接在各节点上两个方向的电缆是被节点内部的总线开关断开

的。节点在两个方向上利用独立的通道检测是否有邻节点存在。由应用任命的主节点运行命名过程,将两个方向上的邻节点(如果有的话),以及邻节点的邻节点,依次赋予其相对主节点的连续地址,最终将所有相邻的节点连接起来构成列车总线。初运行后,所有节点均获得列车的结构信息,包括它们各自相对于主节点的地址、方向(左/右)和位置(前/后)列车中其他车辆的数量和位置等。

在正常运行中,总线两端的节点(即端节点)在线路上接入终端电阻以保证线路阻抗匹配,与此同时各中间节点闭合其总线开关以在端节点间实现总线的连续性。当列车总线构成后,端节点的外向通道探测附加节点的存在,并准备将它们集合到列车组成中。在中间节点上,仅一个通道是使能的。

图 1 显示了有 5 个节点的 WTB 初运行后的情况。

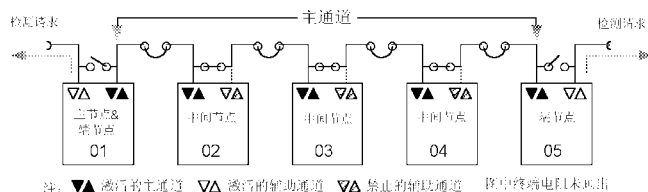


图 1 WTB 初运行

初运行过程是复杂的,因为它不仅关注各节点的正确编号和标识,而且关注低功率休眠模式(用于在蓄电池供电时保护蓄电池不被过放电)与正常模式间的转换。为了在总线中断或主节点失效的事件中可以快速恢复,每个节点都可以变成主节点。在这样的过程中,主节点的身份通过初运行自动地转移到邻近节点。

有关 WTB 的详细内容可参考文献[1]。

目前,WTB 在国内尚无应用,相关的技术和产品研发基本上是空白。本文提出一种 WTB 节点的原型设计构架,并在此基础上对其核心部分的软硬件技术做了研究开发和实验室实现,希望能为开发实用的 WTB 技术和产品作准备。

## 2 WTB 原型节点设计目标

### 2.1 基本设计原则

①) WTB 原型节点的设计依据是 IEC 61375-1 标准 part 4(WTB),应符合 WTB 的基本功能要求。图 2 显示了 WTB 标准分层的参考模型。

②) WTB 节点只实现 WTB 物理层和链路层,高层协议的实现需要宿主机的支持。

③) 作为原型设计,应以实现 WTB 链路层功能为重点,暂不考虑可选功能的具体实现,如介质冗余功能、节点睡眠功能、介质连接器触点去氧化功能等。由于 WTB 标准是分层良好的,这些功能将来可以方便地加入。

根据以上基本设计原则,WTB 原型节点将重点放在 WTB 参考模型中的 3~7 层的实现上。这 5 个层次都属于 WTB 的链路层,是 WTB 节点功能的核心部分,而且这 5 个层次的功能主要是由软件来实现的。

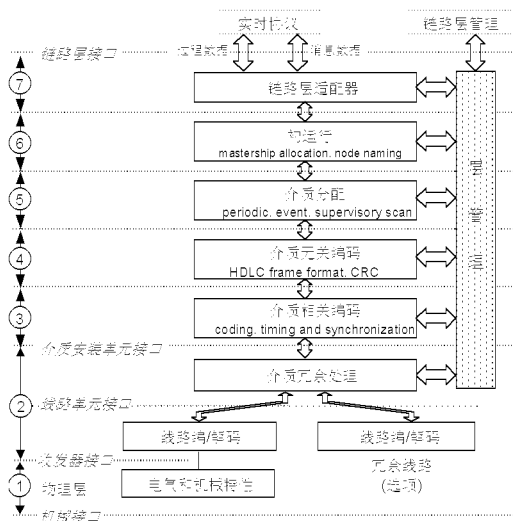


图 2 WTB 参考模型

### 2.2 软硬件设计原则

①) WTB 原型节点软硬件应采用模块化的设计,节点功能可扩充或裁剪。模块化设计使节点能够被灵活的配置以适应不同的应用场合。

②) WTB 原型节点应具有监控和服务接口,方便节点的调试、配置和程序装载。

③) WTB 原型节点应可脱离宿主机独立工作。

## 3 WTB 原型节点硬件

### 3.1 WTB 节点硬件框架结构

依据 WTB 标准要求,按照硬件模块化设计的原则,可将 WTB 节点分成处理器、线路单元和睡眠功能 3 个模块,如图 3 所示。

①) 处理器模块:是 WTB 节点的核心模块,用于实现 WTB 的链路层功能。该模块包括 CPU 及相应的 RAM、ROM 和其它外设、HDL C 通信控制器、宿主机接口、监控和服务端口、Watchdog、线路单元接口逻辑、睡眠状态管理逻辑等。WTB 功能由两个状态机实现:睡眠状态管理逻辑中运行的硬件状态机负责节点

\* 实时性——WTB 通信的实时性是由 WTB 节点软件的实时性保证的。

\* 可靠性——WTB 节点软件的可靠性必须通过严格的设计、编码和测试来保证。应选择软件质量相对容易控制的高级语言来开发。

\* 高效率——WTB 节点软件应有较高的代码执行效率和较小的代码尺寸。

此外, WTB 节点软件的设计还应考虑以下的性能要求:

\* 可移植——WTB 节点软件应可移植到更为紧凑的硬件环境中,甚至宿主机中。

\* 可扩充——WTB 节点软件应具有扩充诸如高层协议栈、网关功能和应用功能的可能性。

基于对 WTB 节点软件的功能要求和性能要求, WTB 原型节点软件选择稳定可靠的嵌入式实时操作系统作为平台, ANSI-C 作为主要的程序设计语言, 按照模块化、分层的方式设计。图 4 显示了 WTB 原型节点软件的总体结构。

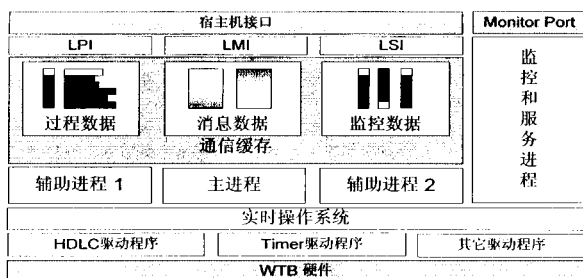


图 4 WTB 节点软件总体结构

位于 WTB 节点软件结构最底层的是驱动程序, 包括驱动 SAB 82525 的 HDLC 驱动程序, 驱动系统时钟的 Timer 驱动程序等。驱动程序直接和硬件通信, 为高层软件提供更为高级的接口。驱动程序层上面是嵌入式实时操作系统。实时操作系统提供进程调度、内存管理和进程间同步和通信的服务。驱动程序利用实时操作系统提供的通信和同步服务来同应用进程通信。主进程和两个一样的辅助进程(各负责一个方向)相互配合来完成 WTB 链路层的核心功能, 它们是 WTB 的核心进程。这 3 个进程作为任务运行在实时操作系统之上。

位于 WTB 节点软件结构最高层的是 WTB 的链路层接口, 包括链路层过程数据接口 LPI、链路层消息数据接口 LMI 和链路层监控数据接口 LSI。链路层接口通过宿主机接口为高层协议或应用提供服务。

在 WTB 链路层接口和 WTB 核心进程之间是通信缓存(Traffic Store)。这是一共享存储结构。其中包括用于过程数据通信的端口映射, 用于消息数据通信的接收和发送队列, 以及用于监控和网络管理服务的监控数据结构。通信缓存由链路层接口和 WTB 核心进程共同维护。

监控和服务进程作为独立的任务运行, 提供 WTB 节点的人机界面。通过它可对 WTB 节点进行监控、配置和调试。

#### 4.2 主进程和辅助进程

主进程和辅助进程完成 WTB 的核心功能。图 5 显示了主进程和辅助进程同线路的两个方向的连接关系。主进程和辅助进程之间通过共享的本地变量进行通信和同步。

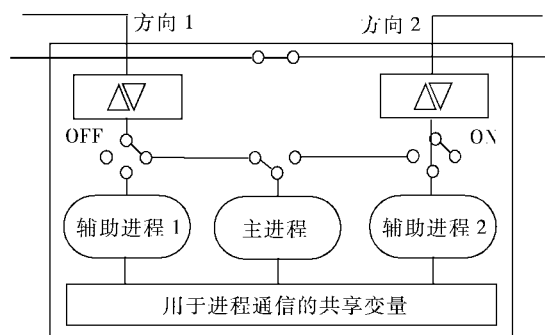


图 5 主进程和辅助进程

辅助进程是一个简单的状态机, 用于检测是否有邻节点存在和响应命名请求。连接在两个方向上的辅助进程功能是一样的, 并只在总线开关为断开时才在节点上运行。孤立的 WTB 节点上运行两个辅助进程。辅助进程在一个方向上收到命名请求后将该方向切换到主进程。

主进程是一个复杂的状态机, 完成 WTB 链路层功能的主要部分。每个命名的 WTB 节点上都有一个主进程运行。在从节点上, 主进程总是连接在指向主节点的方向; 在主节点上, 主进程连接在指向第一个被命名的从节点的方向。主进程有两个基本的状态: 主节点状态和从节点状态。主节点状态下有两个基本子状态。在命名主节点状态下的主进程控制 WTB 的初运行, 为从节点命名。初运行结束后主进程进入常规主节点状态, 此时主进程控制 WTB 总线上的通信进入正常运行。从节点状态下也有两个基本子状态。未命名从节点状态下的主进程在初运行期间接

(下转第 50 页)

在掉电状态、正常状态和睡眠状态之间的转换;CPU上运行WTB软件状态机完成参考模型中3~7层的功能。处理器模块通过宿主机接口同宿主机通信,由运行在宿主机上的软件完成TCN协议的变量通信、消息数据通信和网络监控与管理等高层功能。

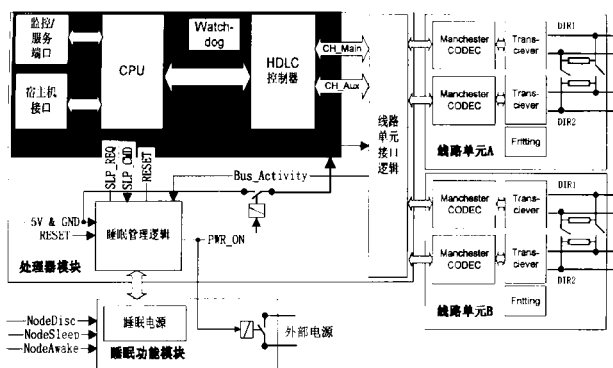


图3 WTB节点硬件框架结构

②) 线路单元模块:实现参考模型中1~2层的功能,即WTB物理层功能。其中包括曼彻斯特编码/解码器、线路收发器、线路切换开关、线路终端器、可选的去氧化功能模块等。线路单元模块通过线路单元接口同处理器模块直接连接,通过安装两个线路单元模块可实现WTB介质冗余功能。

③) 睡眠功能模块:接收控制节点睡眠状态的外部信号,配合处理器模块中的睡眠状态管理逻辑实现WTB节点的睡眠功能,并在节点处于睡眠状态时为线路单元和睡眠状态管理逻辑供电。该模块还具有可用于在节点睡眠时切断外部其它子系统电源的开关电路。

### 3.2 WTB原型节点的硬件结构

按照原型节点着重实现链路层功能的设计原则,通过对图3中的功能模块进行裁剪,只留下满足基本功能要求的部分来构成WTB节点的硬件原型。它包括处理器模块和1个线路单元。

#### 3.2.1 处理器模块

①) 采用PC-104模块作为CPU。这样可以获得良好的开发环境,缩短开发周期,并且在PC-104上开发的软件可以方便地移植到x86系列的嵌入式CPU上。

②) 采用SIEMENS的SAB 82525作为HDL C控制器。SAB 82525的特性完全满足WTB的要求,包括:集成了两个独立的HDL C通道;有多种工作模式可选择,包括透明模式,提供了全面的用户控制能

力;支持Modem接口(RTS/CTS信号);可支持4Mbit/s的通信速率;每通道的发送和接收方向各有64字节深度的FIFO,并支持DMA界面。

SAB 82525的两个HDL C通道分别用作主通道(CH\_Main)和辅助通道(CH\_Aux)。主通道采用DMA方式同CPU模块接口,以保证长帧的顺利收发。辅助通道数据量少,采用中断方式同CPU模块接口。主通道和辅助通道都采用Modem界面同线路单元接口。

③) 采用双端口RAM作为同宿主机通信的界面。WTB节点上的CPU可向宿主机发中断信号,同时宿主机也可以向WTB节点CPU发中断信号。

④) 线路单元接口逻辑由PLD器件实现。由于不实现介质冗余和节点睡眠功能,线路单元接口逻辑只需要完成线路交换开关功能,使主通道和辅助通道可在线路单元的方向1(DIR1)和方向2(DIR2)之间交换连接。

⑤) CPU模块上的一个RS-232端口用作监控和服务端口,用于节点的监控、配置和程序数据下载。

#### 3.2.2 线路单元

由于不实现介质冗余功能,WTB原型节点中只配置了1个线路单元。线路单元通过Modem界面同处理器模块接口。该接口包括接收/发送数据信号和时钟信号、请求发送/清除发送(RTS/CTS)握手信号、载波检测信号和信号质量信号(用于冗余介质的仲裁)。

线路单元中包含两套相同的通信驱动电路,分别对应方向1和方向2。方向1和方向2之间有总线开关和终端器电路,用于连接或断开两个方向的线路并保证线路阻抗匹配。

通信驱动电路由曼彻斯特编码器/解码器、线路驱动器、隔离变压器、隔离继电器等组成,实现WTB物理层的功能。

### 4 WTB原型节点软件

WTB功能的核心部分是由链路层功能软件完成的。这部分软件处理WTB参考模型中第37层的功能,包括:帧格式,报文的编码、定时与同步,介质访问控制,初运行过程,以及向高层协议提供服务的链路层接口。

#### 4.1 WTB原型节点软件结构

由于软件是实现WTB节点功能的关键,所以对WTB节点软件提出了较高的性能要求:

针打印。

4) 打印机可靠性的设计: 为了保证打印机不会出现死机现象, 在软件设计时设计了定时复位程序, 当打印机处于空闲状态达到 2 秒时, 引起定时中断, 打印机自动复位, 从复位入口点开始重新运行程序, 当打印机处于走纸或打印状态时, 不断复位定时器, 禁止打印机复位。微型打印机程序流程图见图 4。

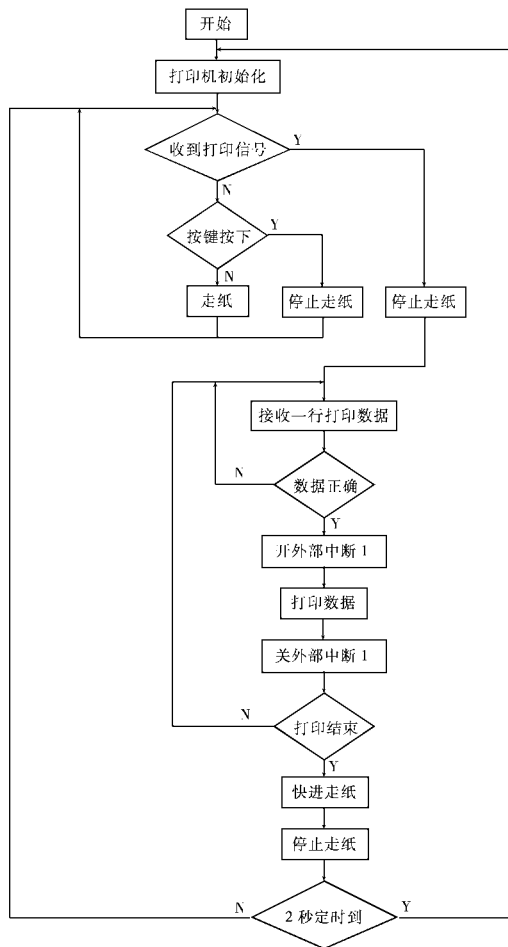


图4 微型打印机程序流程图

## 5 结束语

本文所设计的出租车计价器微型打印机完全按照香港运输署的要求进行设计, 具有较高可靠性, 已在香港市场大量使用。

### 参考文献

1. www.atmel.com ; ATMEL 公司 2051 CPU 数据
2. EPSON 公司: P180 系列微型点阵打印头产品说明书
3. 何立民: 单片机应用系统设计, 北京航空航天大学出版社, 1990. 1

(上接第 4 页)

受主节点的命名, 同主节点协商节点的过程数据周期, 并从主节点接收总线拓扑信息。总线进入正常运行后进入常规从节点状态, 主进程响应主节点的命令, 完成数据通信。主进程可在应用的命令、接收的数据和各种超时事件的驱动下在各种状态间转换。

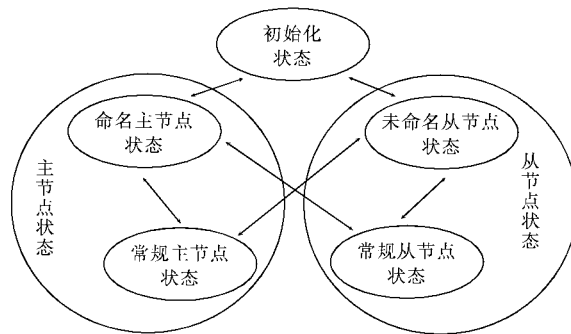


图6 主进程的几个主要状态

主进程和辅助进程通过实时操作系统提供的通信和同步机制同 HDLC 驱动程序通信, 这样既充分利用了操作系统的多任务能力, 又能实现对总线通信的快速响应, 保证 WTB 通信的实时性。

## 5 结束语

WTB 是 TCN 标准中的重要组成部分, 是专为列车级的数据通信设计的。当前, 国内对各种准高速 / 高速列车和动车、地铁和轻轨列车的需求不断扩大, 各种新型机车车辆对数据通信和互联的要求日益提高。在这种形势下, 研制国际标准的 WTB 技术和产品具有重要的意义。

本文提出了一种 WTB 节点设计的框架方案, 并在此框架下设计实现了简化的具备基本功能的 WTB 原型节点。在该原型节点中, 我们围绕 WTB 的核心部分——WTB 链路层设计开发了基本的硬件模块, 并按 WTB 标准开发了 WTB 链路层软件, 实现了 WTB 的基本功能。我们将以这个原型节点为基础, 进一步完善软硬件设计, 争取早日开发出实用的 WTB 技术。

### 参考文献

1. IEC 61375 - 1 part 4, Wire Train Bus. IEC, Geneva, 1999. 9
2. Hubert Kirmann, Ulrich Claessen, 汤林书译, IEC 列车通信网, 机车电传动, 1999. 3
3. 曾祝林, 谢维达, 智能列车总线 Arcnet 通信网卡的研究, 机车电传动, 1997. 6