

基于 LON 与 CAN 现场总线的智能协议转换器

王忠锋 王毅峰 李令奇 中国科学院沈阳自动化研究所五室(110003)

Abstract

This paper introduces an intelligent protocol converter based on LON and CAN FieldBus, including its functional principle, hardware circuit and software design. Solutions for problems should be considered in design are provided in this paper. This intelligent protocol converter has resolved successfully the problem of communication between the two kinds of control network.

Keywords: FieldBus, intelligent protocol converter, LonWorks, CAN

摘要

本文介绍了基于 LON 与 CAN 总线的智能协议转换器的工作原理、硬件电路设计及其软件设计,并在文章的最后给出了设计中应注意的问题及解决方法。该协议转换器成功地解决了两种不同控制网络之间的通信问题。

关键词:现场总线,智能协议转换器, LonWorks, CAN

自 80 年代以来,现场总线技术取得了迅猛地发展,在控制系统中的应用越来越广泛,必将逐渐取代传统的集中式控制系统。在众多的现场总线当中,由美国 Echelon 公司推出的 LON 总线和由德国的 BOSCH 公司在推出的 CAN 总线,各因其所独有的技术特点而越来越受到自控界的重视。本文介绍了基于 LON 总线和 CAN 总线的智能协议转换器,以便实现两个基于不同协议的控制总线之间的数据通信,充分结合两种控制总线的优点,同时大大方便了控制系统的建设。

1 LonWorks 技术及 CAN 总线概述

LON (Local Operating Networks) 总线是美国 Echelon 公司推出的局部操作网络,它是一种全分布式智能控制网络技术。提供了一套完整的开发平台 LonWorks 技术,其核心是神经元芯片 (Neuron Chip)。该芯片同时具备了通信和测控功能,并将通信协议固化到芯片当中,只有协议的应用层需用户编写。LonWorks 技术采用的通信协议称为 LonTalk,是一种直接面向对象的通信协议。该协议使用了网络变量这一形式,大大简化了控制系统的构建工作。LonWorks 技术提供了基于整个控制系统的 LonBuilder 和直接针对节点的 NodeBuilder 开发工具,保证整个网络具有互操作性的网络服务体系架构 LNS。应用程序采用专用的 Neuron C 语言编写。

支持不通的通信介质,并且可以在一个网络中混合使用。LonWorks 技术能方便灵活地构建低成本的控制系统。

有关 CAN 总线的技术特点,限于篇幅,在此不在赘述,相关内容请参考文献 [1]。基于 LON 总线和 CAN 总线的特点,我们研制了在两个控制网络之间进行实时通信的智能协议转换器。

2 智能协议转换器的整体设计思想

智能协议转换器的结构原理框图如图 1 所示:

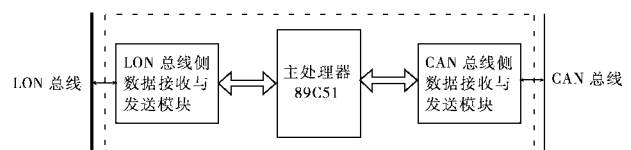


图 1 智能协议转换器原理图

该协议转换器的主处理器采用微型单片机 89C51,负责将来自 LON 总线的数据经过处理后发送到 CAN 总线上,或者将来自 CAN 总线的数据经过处理后发送到 LON 总线上。为适应控制网络中大量的数据信息,89C51 外扩了 2K 的 EPROM,以便存储 LON 总线和 CAN 总线的大量的配置信息。来自 LON 总线和来自 CAN 总线的数据分别通过符合 LonTalk 协议的神经元芯片和符合 CAN 总线协议标准的独立的控制器进行解码和重新编码,以中断的

方式发送到微处理器 89C51, 89C51 根据配置信息实现两种总线间的通信。

3 智能协议转换器的硬件配置

在本协议转换器的硬件设计当中, 采用的主要芯片为 89C51, 神经元芯片 TMPN3150, 自由拓扑收发器 FTT-10A, CAN 通信控制器 SJA100 和 CAN 总线收发接口电路 PCA82C250。

神经元芯片共有两个系列 3150 和 3120。本设计采用 TMPN3150B1AF, 可外接程序存储器。芯片内部有 512 字节的 EEPROM, 用来存储网络参数如网络配置、编址信息和用户应用程序代码。另外, 还有 6 个字节用来放置神经元芯片的唯一 48 位标识符 Neuron ID。2k 字节的静态 RAM 用来存储系统数据和应用程序。芯片内部含有 3 个 CPU, 分别为介质访问控制处理器、网络处理器和应用处理器, 前两个处理器负责执行网络协议, 应用处理器负责处理用户应用程序。介质访问控制处理器和网络处理器可根据固化在芯片内的通信协议自动地完成控制模块与网络的通信过程。该芯片有 11 个双向、可编程 I/O 口, 共可配置成 34 种 I/O 对象, 可根据实际需要进行接口方式的选择。5 个通信端口 CP0~CP4 可配置成三种不同的接口模式: 单端、差分和专用模式, 以适合不同的编码方案和波特率, 方便地外接不同的通信介质。3150 芯片内置低电压复位电路, 以防止供电电压低于规定值时引起 EEPROM 错误的操作, 使芯片及时地复位。

在智能协议转换器的 LON 总线侧采用了 Echelon 公司的自由拓扑收发器 FTT-10A。神经元芯片与 FTT-10A 的连接如图 2 所示:

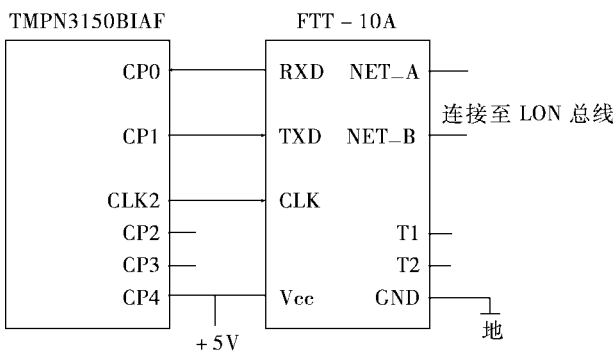


图 2 神经元芯片与收发器 FTT-10A 的连接

3150 的 CP0 引脚连接至 FTT-10A 的引脚 RXD, CP1 引脚与 FTT-10A 的 TXD 引脚相连, 以单端模式实现与 FTT-10A 的通信。CP2 和 CP3 引脚

悬空。CP4 和 FTT-10A 的 Vcc 引脚接 +5V 的直流电压源。CLK 为 FTT-10A 的时钟输入引脚, 输入信号由神经元芯片的 CLK2 引脚提供。FTT-10A 收发器可自动检测时钟频率, 并对内部电路进行相应的配置。

智能协议转换器采用的 CAN 控制器为 PHILIP 公司生产的 SJA1000, 它符合 CAN2.0B 协议, 支持 11 位和 29 位标识符模式, 芯片内扩展了 64 字节的接收缓冲区 (64-byte FIFO), 通过验收屏蔽码寄存器和验收码寄存器对标准帧或扩展帧进行单向/双向接收滤波, 完全兼容 PCA82C200, 含多种微处理器接口。其功能有了很大增强, 通信性能也大大提高。在 CAN 总线的接口电路中, 主要芯片为 PCA82C250, 它是 CAN 控制器与物理总线之间的接口, 实现 SJA1000 与 CAN 总线间的电平转换, 它可以提供对总线的差动发送能力和对 CAN 控制器的差动接收能力, 抗干扰能力强。82C250 驱动电路内部有限流电路, 可防止发送输出级对电源、地或负载短路。

4 协议转换器的硬件实现

智能协议转换器的 89C51 与神经元芯片 TMPN3150 和 CAN 控制器 SJA1000 的接口电路如图 3 所示:

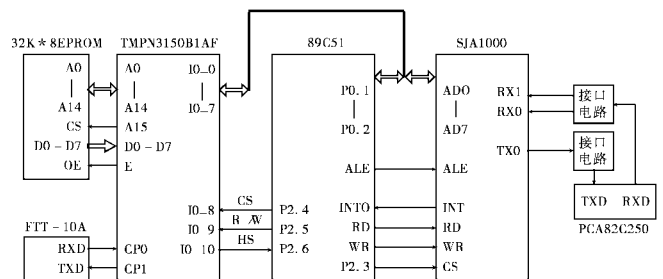


图 3 智能协议转换器的硬件接口

智能协议转换器采用 89C51 来完成 LON 总线与 CAN 总线间的数据通信的协调工作。P0 口作为并行双向数据口, P2.3 和 P2.4 分别作为 SJA1000 和 3150 芯片的选通控制位, P2.5 作为 3150 芯片的读写控制位, P2.6 作为握手信号位。神经元芯片 TMPN3150 扩展了 32K*8 的 EPROM, 连接方式见图 3, 其中 3150 的地址线 A15 与外部存储器的片选端 CS 相连, 当 A15 为低电平时, 3150 芯片可读取外存的地址为 H0000 到 H7FFF, 使能信号 E 作为启动信号, 3150 的读写控制端 R/W 在本协议转换器的设计中没有用到。在外部存储器的 H0000 到 H3FFF

的 16k 的字节中存有 LonTalk 协议的内容,包括介质访问控制和网络处理器执行的系统软件以及 I/O 驱动和事件驱动工作软件。

3150 通过收发器 FTT-10A 与 LON 总线相连接。收发器 FTT-10A 从 LON 总线接收数据,并通过通信引脚 RXD 将数据经由神经元芯片的通信端口 CP0 进行发送。3150 将接收来的数据按照 LonTalk 协议标准重新进行解码,将经过处理后的数据向 89C51 发送。3150 与外部处理器接口有异步串行方式,同步串行方式,并行 I/O 方式,多总线 I/O 方式等,在本设计中,将 3150 的工作方式设定在并行 I/O 方式的从 A 模式下,其 IO_0 ~ IO_7 作为双向数据口,负责 3150 与 89C51 在并行工作方式下传送数据。IO_8 作为 3150 的片选端 CS,接收来自主处理器 89C51 的片选信号,启动 3150 向主处理器发送或从主处理器接收数据。CS 线总是由主处理器的 P2.4 引脚驱动,当被激活时,进行字节传送操作,这条线上的下降沿对主处理器和 3150 的数据进行滤波。CS 线是异步方式的,应尽量保持无噪声。当 CS 线上的电平处于低状态时,数据传送的方向是由 R/W 线上的电平来决定的。IO_9 作为读写控制信号端 R/W 接收来自主处理器的读写信号,由主处理器 P2.5 引脚来驱动,电平为高时(此时 CS 应当为低),3150 从主处理器读取数据,否则向主处理器发送数据。IO_10 设置为握手信号端 HS,由神经元芯片驱动,当 3150 将要向主处理器 89C51 发送数据时,通过 HS 线向 89C51 发送握手信号,89C51 产生中断来接收数据,根据存储在其 EPROM 中的配置信息对数据进行处理,并将处理后的数据通过 SJA1000 按照 CAN 协议标准进行编码,然后发送到 CAN 总线控制网络。

微控制器对 SJA1000 的操作相对比较简单,由于 SJA1000 内可存储数据的地址较少,所以只需用一个 8 位的地址端口 AD0 ~ AD7 就可以对实现地址的选择,且地址和数据总线是共用的,其具体操作过程是:当 ALE 信号从低变高时,开始了读/写周期,CPU 把 8 位地址送到总线上,ALE 的下降沿用来把地址信息锁存到 SJA1000 的片内地址锁存器;对读周期,在读控制信号 RD 变为有效后,它使得被寻址的数据存储器把有效的数据送上总线;而对写周期 CPU 发送完地址后就直接送数据,当写控制信号 WR 有效时,数据就被写入选通的存储器内。SJA1000 由 89C51 的 P2.3 引脚发送的片选信号 CS 来驱动,当 CS 为低时,允许微控制器对其进行访问。

同样原理,SJA1000 通过 CAN 总线收发接口电路 PCA82C250 从 CAN 总线接收数据,二者的连接如上图所示,SJA1000 的 RX0、RX1 引脚通过接口电路与 82C250 的 RXD 引脚相连,接收来自 CAN 总线的的数据, TX0 引脚与 82C250 的 TXD 引脚相连,向 CAN 总线发送数据。SJA1000 通过引脚 INT 以中断的方式将解码处理后的数据发送到 89C51,89C51 根据配置信息将数据发送给 3150,3150 按照 LonTalk 协议标准对接收来的数据的进行重新编码后发送到 LON 总线上。

为了实现单片机芯片和神经元芯片以及 SJA1000 之间的同步,在设计中采用在单片机复位的同时,也触发 3150 和 SJA1000 复位。另外,单片机需查询 HS 的状态以保证数据传输的同步,但是存在 3150 芯片还没有设置好 HS 的正确状态,而此时单片机已经开始查询 HS 的情况,为了避免这种情况的出现,以防止单片机读取 HS 的无效状态,可在 HS 引脚上加上 10KΩ 上拉电阻。

5 软件及硬件设计中应注意的问题

5.1 软件设计要点

对于神经元芯片 3150,采用 Echelon 公司提供 NodeBuilder 节点开发工具进行开发。NodeBuilder 节点开发工具集成了 NodeBuilder 节点定义和控制工具,NodeBuilder 编辑器,Neuron C 编译器,调试器等工具软件。NodeBuilder 编辑器用来创建和修改 Neuron C 源程序。网络变量浏览器可在程序运行过程中对网络变量进行监测,并可修改节点的行为。该开发工具还包括 LON 模块应用接口和 LonMaker DDE Server。NodeBuilder 节点开发工具的硬件部分包括:PCNSS PC 接口卡,LTM-10 协议节点,LTM-10 协议模块,LON 网络模块接口, Motorola Gizmo 3 模块。神经元芯片的应用程序用 Neuron C 语言编写,它是在 ANSI C 语言的基础上做了扩展,包括一个内部多任务调度程序,网络变量类型,软件定时器,I/O 读写函数和事件驱动编程结构等。在编写应用程序时应注意,事件处理程序不易过长,否则的话会影响其他事件的触发,还有可能引起芯片的复位。

Neuron 芯片固件允许在同一条总线上挂接多个设备,但必须保证在同一时刻只有一个设备向总线写数据。89C51 通过令牌传递握手协议与 3150 进行通信,神经元芯片的令牌传递和握手信号可由 Neu

(下转第 62 页)

(2) 设计移位寄存器循环扫描输入程序,使移位寄存器数与选定的输入点数相等,并使各位与各输出点一一对应,在移位寄存器循环移位时,各输出点在相对应的移位寄存器各位状态输出驱动下会依次循环导通。当选用 100ms 时钟脉冲作为移位输入脉冲时,每个输出点会依次各导通 100ms。

(3) 用 n 个输入点与 m 个输出点的状态两两相“与”来指定一组 nm 个内部继电器。以这 nm 个内部继电器的编号分别代替对应的 nm 个输入信号开关的编号,就可以得到 nm 个输入信号。

(4) 编制应用程序时即以 nm 个内部继电器的编号作为相应的 nm 个输入信号开关的地址号编入程序。

3 输出端点不足的处理方法

(1) 某些控制逻辑简单,而又不参与工作循环,或在工作循环开始之前必须预先起动的电动机、电器,可以不通过 PLC 控制。如:液压设备的液压泵,电动机的启动、停止控制等就可以不由 PLC 承担。

(2) 启动、停止时间完全相同的负载,在 PLC 的输入点功率允许的情况下,可并联于同一输出端点,即一个输出点带多个负载。

(3) 利用接触器的辅助触点

(上接第 9 页)

ron C 语言提供的函数和事件驱动自动完成,而对于 89C51,则必须通过编程来实现令牌传递和握手信号过程,即在单片机要采用 3150 的主工作方式,以控制 3150 工作在从 A 方式。在设计中,我们采用了 C51 语言对 89C51 进行编程来实现该算法以及数据的传输。

5.2 电路设计中应注意的问题

智能协议转换器工作在复杂的电磁环境当中,与其他电子设备之间不可避免地存在各种形式的电磁干扰(EMI),另外,静态放电(ESD)也会对电路产生冲击。这两点在电路设计当中必须慎重考虑。EMI 引入的主要途径是对地公共阻抗,可采用星形接地使金属构件地的连接点置于驱动器电路和快速电路之间,同时增大地线宽度,减小对地阻抗,从而将电路间通过公共地的传导干扰减至最小。在印刷板 PCB 电路中,电源连接端,网络连接端和所有的金属构件地连接点都尽量靠近星形中心,这样,可在对内部电路扰动最小的情况下将 ESD 的瞬态电流导出节点。在收发器 FTT-10A 与网络的连接端设计可在 1-2kV 的静电电压下击穿而放电的火花隙,削弱到达收发器和

许多控制系统,通常都含有接触器,必要时可考虑用接触器的辅助触点进行电气联锁或控制指示灯,这样也可减少输出点。

(4) 用数字显示器代替指示灯

如果工作状态的指示灯或程序步比较多,最好采用数字显示器代替指示灯。这样也可有效地减少输出点的个数。

(5) 多种故障显示或报警并联连接

有些系统可能有多种故障报警显示或报警,例如设有过压、过载、超速、越位、失磁、断相等显示或报警,只要条件允许,可把部分或全部显示或报警电路连接,用一个或少用几个输出继电器驱动,这样也可以减少输出点的个数。

参考文献

1. 田瑞庭,可编程控制器应用技术,机械工业出版社,1994.5
2. 齐从谦、王士兰,PLC 技术及应用,机械工业出版社,2000.8
3. 熊葵客等,电器逻辑控制技术,科学出版社,1996
4. W. Raymond. 8th Annual Programmable Controller Update. Control Engineering. 1991

后继缓冲电路的 ESD 能量。另外,在电路设计中还应注意通过电容对电源进行解藕,以减小对地电感。

在 CAN 控制器 SJA1000 与收发器 PCA82C250 的接口电路中,采用了光电隔离元件 6N137,以实现总线上节点间的电气隔离。

6 结束语

在复杂的控制系统中,采用组合控制网络有时是最好的选择。LON 总线和 CAN 总线各具特点,其组合基本上能满足大多数控制系统的要求。本文所论述的智能协议转换器,设计简单,实用性强,成本低廉,可用于大型采用 LON 总线和 CAN 总线的控制系统,经试验表明,该转换器能以良好的性能完成 LON 总线和 CAN 总线之间的通信。

参考文献

1. 王毅峰,李令奇,基于 CAN 总线的分布式数据采集与控制系统,工业控制计算机,2000.5
2. TAEC Neuron Chip Databook,TOSHIBA,2000
3. LonWorks Microprocessor Interface Program (MIP) User's Guide,Echelon,Revision 3
4. 阳宪惠,现场总线技术及其应用,清华大学出版社,1999