

一种应用 LONWORKS 控制模块的 现场智能节点的设计

杨 侃 ,赵维琴 ,郎文鹏 ,马利民

(上海大学自动化系 ,上海 200072)

[摘 要] 本文首先简要地介绍了 LonWorks 现场总线技术以及 LonWorks 控制模块 ,在此基础上提出了一种 LonWorks 控制模块与 MCS - 51 单片机并行通信节点的组成原理及其设计与实现。

[关键词] 现场总线 ;LonWorks ;并行通信 ;神经元芯片

[中图分类号] TP336 [文献标识码] B [文章编号] 1000-068X(2001)06-0037-03

The design of a field intelligent node using the LonWorks control module

YANG Kan ,ZHAO Wei-qin ,LANG Wen-peng ,MA Li-min

(Automation Department of Shanghai University ,Shanghai 200072 ,China)

Abstract : This paper gives a brief account of the control technology of the LonWorks field bus and the LonWorks control module ,then presents the principle and design of the parallel communication between the LonWorks control module and the MCS - 51 single-chip microcomputer including the practical application of the LonWorks node.

Key words : Field bus ;LonWorks ;Parallel communication ;Neuron chip

1 引言

现场总线是在自动化领域中出现的一种崭新的控制技术 ,它以其全数字化、全分散式、全开放性、互操作性以及开放式的互连网络等特点为传统的自动控制系统带来了革命性的变革 ,已成为未来控制系统的发展方向。在众多流动的现场总线中 ,由美国 Echelon 公司研制的 LonWorks 现场总线是唯一遵循了国际标准化组织 ISO 定义的开放系统互连 OSI 全部 7 层模型的现场总线标准 ,因其能充分满足未来发展对测控网络的要求 ,具有广阔的应用前景。LonWorks 现场总线技术的优势在于其高性能低成本

的网络接口产品、含 3 个 CPU 的超大规模 Neuron 芯片、固化的 LonTalk 通信协议以及方便的开发调试服务工具。实现 LonWorks 技术的关键是 LonWorks 智能设备 ,即控制网络节点的应用开发。本文将着重讨论并提出了一种 LonWorks 控制模块与 MCS - 51 单片机并行通信节点的组成原理及其设计与实现。

2 原理

2.1 神经元芯片

神经元芯片是 LonWorks 技术的核心 ,它有 3 个 CPU ,每个 CPU 各自分工不同 ,如图 2-1 所示。

CPU - 1 是介质访问控制处理器 ,处理 LonTalk



图 2-1 神经元芯片结构示意图

收稿日期 2000-10-09

作者简介 :杨侃(1968-),男,电气工程师,硕士生;研究方向:自动化装置及系统。

协议的第一和第二层,包括驱动通信子系统硬件和执行 MAC 算法。CPU - 2 是网络处理器,它实现 LonTalk 协议的第三到第六层,包括处理网络变量、寻址、事件处理、软件计时器、网络管理和路由等,同时还控制网络通信端口,发送和接收数据包。CPU - 3 是应用处理器,它执行用户的代码以及用户代码调用的操作系统命令。CPU - 1 和 CPU - 2 用共享存储区中的网络缓存区进行通信。CPU - 2 和 CPU - 3 用共享存储区中的应用缓存区进行通信。在多数应用中,编程采用 Neuron C 语言。

2.2 LonWorks 节点与控制模块

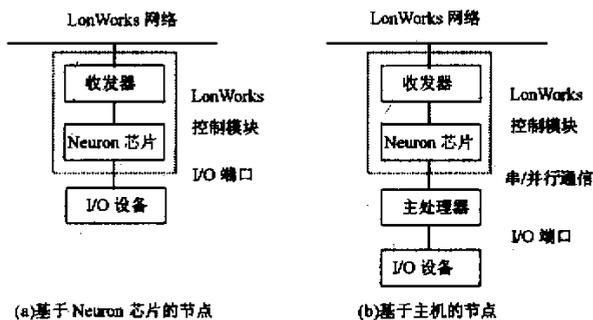


图 2—2 LonWorks 网络控制节点

LonWorks 网上的每个控制点,即 LonWorks 节点是和物理上与之相连的现场 I/O 设备交互作用并在控制网络中使用 LonTalk 协议与其他节点互相通信的一类对象。

LonWorks 网络控制节点有两种类型。如图 2—2 所示:

在图 2—2a 中,Neuron 芯片是唯一的处理器,此类 LonWorks 节点适合于 I/O 设备简单、处理任务不复杂的系统,称为基于 Neuron 芯片的节点;而在图 2—2b 中,Neuron 芯片只作为通信处理器,充当 LonWorks 网络的网络接口,节点应用程序由主处理器来执行,此类 MIP 结构的 LonWorks 节点适合于对 I/O 设备及处理任务要求较高的系统,称为基于主机的节点,其中的主处理器可以是任何微控制器或 PC 等。

不论哪种类型的 LonWorks 节点都有一个电源、一片 Neuron 芯片用于通信和控制、一个 I/O 接口以及一个收发器负责将节点连接到 LonWorks 网络。为了便于 LonWorks 现场总线的应用, Echelon 公司开发出了多种型号的 LonWorks 控制模块,将 Neuron 芯片、收发器以及存储器集成在一起,如图中虚线框所示,极大地方便了 LonWorks 总线的应用设计开发人员,使开发人员将注意力更集中于控制系统的选择、应用和实现。

3 节点硬件的设计

LonWorks 控制模块与 MCS - 51 单片机并行通信节点的硬件电路原理示意图如图 3—1 所示,它属于图 2—2b 类基于主机的 LonWorks 节点。

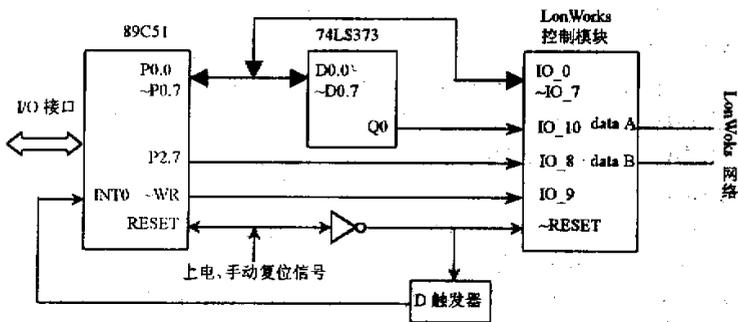


图 3—1 硬件原理图

图 3—1 是其硬件电路原理图。需要说明的是: Neuron 芯片提供有 11 个可编程的 I/O 引脚(IO_0 至 IO_10),它们可以配置为多达 34 种不同的应用对象,从而借助于最小的外接电路实现灵活的输入输出功能。Neuron 芯片的并行 I/O 对象需要使用全部 11 个引脚,其中 IO_0 ~ IO_7 用于双向数据线, IO_8 ~ IO_10 用于控制信号线,它有 3 种工作方式:即主方式(Master)、从 A 方式(Slave A)和从 B 方式

(SlaveB)。对于要求建立 Neuron 芯片与微处理器或微控制器之间的连接,即建立基于主机的 LonWorks 节点来说,Neuron 芯片的并行接口工作在从 B 方式是最佳的选择。工作在从 B 方式的 Neuron 芯片,在主机的地址空间,就像是两个寄存器,一个是读写数据寄存器(偶地址),另一个是只读状态寄存器(奇地址)。主机正是通过对这两个寄存器的访问实现主机与 Neuron 芯片之间的数据并行传输的。在从 B

方式下,IO_0 除作为数据低位外,还兼作握手 HS 位,用于主机与 Neuron 芯片的握手应答;IO_8 则作为片选信号位;IO_9 作为读写信号线;IO_10 作为寄存器寻址输入位。另外,由于并行通信要求双方设备必须同步,无论 MCS-51 处理器和 Neuron 芯片哪一方复位,双方都必须重新进行同步,因此在 MCS-51 处理器和 Neuron 芯片复位电路的设计上应遵循以下原则:要求双方都能意识到对方的复位,MCS-51 处理器的复位要触发 Neuron 芯片的复位,这是通过直接控制(硬件连接)实现的,而 Neuron 芯片在由于某种原因引起复位时,会在其 RESET 端自动产生低电平输出,利用这一复位输出信号,通过一个 D 触发器以中断方式来通知 MCS-51 处理器,以使 MCS-51 处理器作出相应处理,这是通过中断服务子程序实现的。

4 节点软件的设计

由于 Neuron 芯片内装有 LonTalk 通信协议固件,因此无论 Neuron 芯片并行 I/O 对象工作于何种方式下,对于 Neuron 芯片来说,其实现软件相对容易。故对于 LonWorks 控制模块与 MCS-51 单片机的并行通信节点的实现,关键在于 MCS-51 单片机的软件设计,这是因为 MCS-51 处理器内不含 LonTalk 通信协议固件,因此要实现与 Neuron 芯片的并行数据传输,MCS-51 处理器这一方必须复制 Neuron 芯片的行为,即能执行 Neuron 芯片的握手/令牌传递算法。Neuron 芯片并行 I/O 对象的令牌命令字节格式描述如图 4-1 所示。

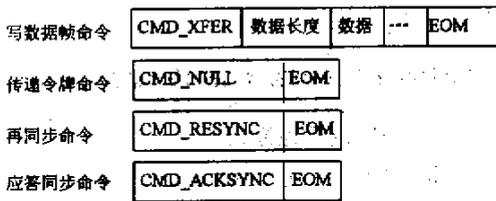


图 4-1 协议命令格式

上述每一命令的第一字节代表的是该命令类型,其中,CMD_XFER = 0x01,CMD_NULL = 0x00,CMD_RESYNC = 0x5A,CMD_ACKSYNC = 0x07,数据长度是实际发送的数据长度,不包括 EOM,EOM 是命令结束字节,可以为任意字节,它只被发送,接收方不读该字节,主要是通过写该字节保持 HS 位为

读方可写的状态以便传递令牌。

令牌传递过程如图 4-2 所示。

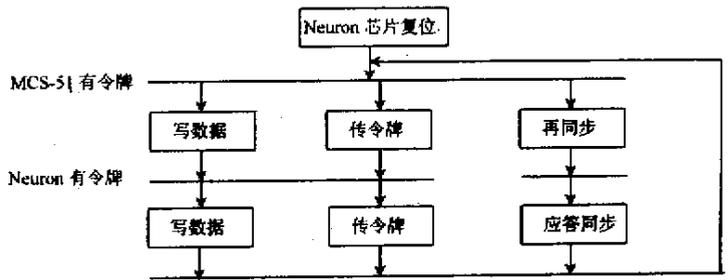


图 4-2 令牌传递过程

为实现与并行 I/O 设备的通信并保证安全可靠,Neuron 芯片由固件自动执行令牌传递协议,以防止总线冲突。在任何给定的时间内,仅有一个设备拥有令牌。该令牌是一虚拟令牌,它决定着哪一设备拥有写总线权。若主机拥有令牌,它将有将准备好的数据发送给或将令牌交给 Neuron 芯片,若 Neuron 芯片拥有令牌,它可将准备好的数据发送给主机或交出令牌。因而令牌在主机与 Neuron 芯片之间以乒乓方式来回传递。

当 Neuron 芯片复位后,MCS-51 处理器将自动获得令牌并负责令牌的传递。Neuron 芯片在任何复位的情况下都通过并行总线要求同步,其目的是为了防止 Neuron 芯片作出错误的假定,比如错误的传送开始或错误的传送数据,这将由 Neuron 芯片通过同步作用自动结束。持有令牌的 MCS-51 处理器在任何时候应用 RESYNC 命令可以初始化重新同步操作。无论 Neuron 芯片和 MCS-51 处理器哪一方设备复位,都要求双方必须重新同步,因此双方都应能意识到对方的复位,MCS-51 处理器要监视 Neuron 芯片的复位,且 MCS-51 处理器的复位要触发 Neuron 芯片的复位。

HS 位(即 IO_0)是主要的握手控制信号,用于控制实际数据的传输。在 Neuron 芯片并行 I/O 对象从 B 方式下,HS 由 Neuron 芯片驱动,它通知 MCS-51 处理器目前 Neuron 芯片是忙还是闲,当 Neuron 芯片执行一次读写数据操作后,HS 状态位被置“0”,当 MCS-51 处理器执行一次读写操作后,HS 状态位被置“1”,由于这是由硬件而非由固件控制的,因此 MCS-51 处理器必须在程序中轮询该状态位,从而正确地启动自身的读写操作。

为了实现 LonWorks 控制模块与 MCS-51 处理器

(下转第 62 页)

(上接第 39 页)

的并行通信, MCS-51 处理器这一方需要人为地完成 Neuron 芯片的握手/令牌传递协议, 在编制程序时, 还要注意的是要保证 Neuron 芯片一方保持令牌的时间不能超过 Neuron 芯片看门狗定时器的定时时间, 以防止 Neuron 芯片意外复位而导致通信失败。

5 结束语

基于 Neuron 芯片的 LonWorks 控制网络节点, 只能用于 I/O 及控制较简单的系统。对于工业应用中 I/O 及控制较复杂的控制系统, 往往要采用基于主要处理器的 LonWorks 网络节点。本文所设计的控制网络节点, 其主处理器采用的是 MCS-51 系列单片机, 主处理器与 Neuron 芯片之间采用的是并行双向数据通信, 其传输速率最高可达 9600bps, 目前该节点已经应用于智能小区安保控制系统。

[参考文献]

- [1] 张友德, 等. 单片微型机原理、应用与实验[M]. 复旦大学出版社, 1995.
- [2] 杨育红. LON 网络控制技术与应用[M]. 西安电子科技大学出版社, 1999.
- [3] Echelon 公司. Parallel I/O Interface to the Neuron Chip[M].

