

Freelance 2000 系统冗余控制站的实现

刘长生 济南钢铁集团总公司发展规划部(250101)

侯丽丽 济南钢铁集团总公司自动化部 (250101)

Abstract

This article has narrated how to achieve a redundant control by Freelance 2000 controller in detail. Hardware connection and software configuration must be done in correct step. It has also told you the solution of problems in the process of debugging.

Keywords: redundant control, CAN bus, diagnose, Freelance 2000

摘 要

本文详细阐述了实现 Freelance 2000 集散控制系统冗余控制的硬件连接及软件组态的正确步骤,以及调试过程中出现问题的解决方法。

关键词: 冗余控制, CAN 总线, 诊断, Freelance 2000

济钢焦化厂新建煤气净化回收系统的过程检测与控制采用 ABB 公司的 Freelance 2000 集散控制系统, 系统硬件结构为: 3 台上位机(即 2 个操作员站, 1 个工程师站), 一个过程站。由于工艺复杂, 需要检测与控制的量多, 为了保证生产的连续性与控制的可靠性, 过程站采用双冗余方式, 即连接模块为冗余供电方式。其配置如图 1 所示。

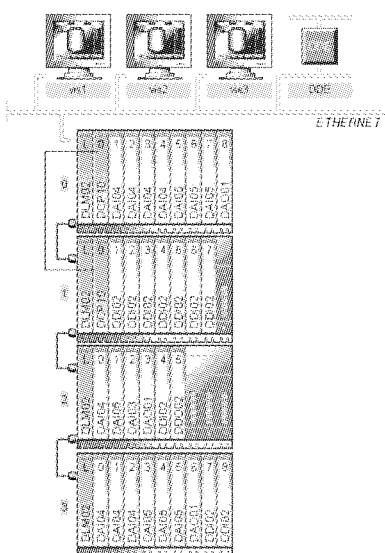


图 1 系统硬件配置图

1 硬件连接

CPU 模块为冗余配置, 且各占一个机架, 两 CPU 之间通过细缆自动将实时数据由主 CPU 传到备用 CPU, 确保在主 CPU 发生故障时及时地切换到从 CPU。Diginet S(Ethernet)系统总线连接各个站, 并进行数据传输, 本系统采用 10Base5 粗缆作为总线传输介质, 传输速率为 10Mbps, 各个站通过收发器接入 Ethernet; 过程站总线为 Diginet P(又称 CAN 总线), 通过它连接中央 I/O 单元与 I/O 扩展单元并在 CPU 与 I/O 模块之间传输 I/O 数据。

1.1 诊断电缆制作

CPU 模块的检测接口 Diag 用于建立“点对点”连接, 即 CPU 模块与上位机直接电缆连接。由于默认的网络接口设置为细缆连接口(COAX1), 而我们实际采用 AUI 口进行连接, 为此需将设置改为 AUI 连接口, 故需通过 Diag 口对 CPU 作端口设置上的修改。软件设置之前需先制作一根诊断电缆, 用于连接 PC 机的

9 针串口与 CPU 模块的 Diag 口, 做法为通用 D 型 9 针头串口连接的方法, 如图 2 所示。

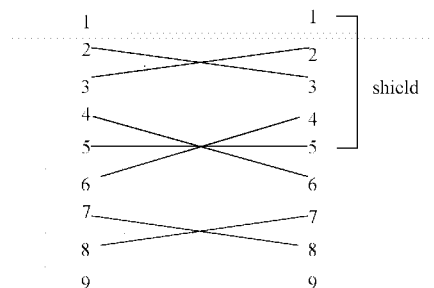


图 2 诊断电缆连接图

1.2 冗余 CPU 模块之间电缆连接

冗余 CPU 模块之间使用细同轴电缆 10Base2 进行连接。

除了同轴电缆外, 还需要两个“T”型头, 两个 BNC 接头 (Bayonet Neill-Concelman, Connector used with coaxial cable invented by Mr. Neill Concelman), 两个 50Ω 端结器。

“T”型头有三个头, 其中两个分别连接做好的同轴电缆线或 50Ω 端结器, 另一个连接 CPU 模块上的 COAX2 口。

BNC 接头安装在一段同轴电缆线两端, 使电缆线能连接上“T”型头, 50Ω 端结器安装在 10Base2 的两端, “T”型头的一头连接做好的同轴电缆线, 另一头连接 50Ω 端结器。

1.3 以太网电缆连接

本系统网络拓扑结构为总线型, 用 10Base5 粗同轴电缆作总线传输介质, 各计算机通过收发器与总线连接, 两个收发器之间距离不小于 2.5 米, 计算机与收发器之间为 AUI 电缆, 粗缆两端安装 50Ω 终端电阻。

2 软件组态

2.1 系统环境

工程师站所用 PC 机操作系统为 Windows NT

4.0(Service Pack 4),开发软件为 Freelance 2000 V6.1(包括 Digitooll,Digivis,Digilock,DDE),安装有 Digitooll 硬 Key。

2.2 诊断与设置 CPU 模块参数

连接好诊断电缆(两端各接 CPU 模块的 Diag 口和计算机的 COM2 口)之后,运用 NT 提供的“超级终端”,对 CPU 模块作诊断与设置:第一网络接口为 AUI(收发器),第二网络接口为 Cheapnet(细缆为冗余)。

2.3 互为冗余的 CPU 设为不同的 IP 地址

在系统组态软件 Digitooll 中,在“组态项目树”下,组态一个冗余过程站;在组态“硬件结构”时,此过程站的 IP 地址为两个且不同。

3 遇到问题及解决方法

在联机调试时,我们遇到了以下问题:

问题一:所有的 I/O 模块在系统自诊时,有时认

为组态的模块与实际插在机架上的模块一致,有时认为不一致。经检查发现,LINK 模块地址冲突,修改后,问题解决了。

问题二:中央 I/O 单元的所有模块均正常,而系统不识别带冗余 CPU 模块的扩展 I/O 单元。经检查发现,网络电缆连接错误。正确的连接为:每个 CPU 模块的 AUI1 口均连接收发器的电缆,用作冗余连接的细缆均接在 CPU 模块的 COAX2 口上。

4 结束语

由于连接模块采用双路电源供电,即使在其中一路电源断电时,另一路仍能给 I/O 单元提供电源,保证了系统运行的不间断性;CPU 模块互为热备冗余,一块为主,一块为从,在一块 CPU 出现故障时,另一块自动切换为主状态,保证了数据的实时性。

[收稿日期:2002.12.23]

(上接第 40 页)

3.3 整个服务器程序的流程

下面就对整个程序做一个总体的分析。

1)main 函数部分。程序的一开始通过宏定义对服务器的 IP 地址、网关以及子网掩码等局域网参数做了设置,接着初始化 Rabbit 各管脚的工作状态、接口电路芯片的状态和一些相关的参数。在 main 函数的主体部分是一个一直在侦听 80 端口和 3000 端口的循环,循环内部采用了 costate 语句以多任务的方式执行任务 1 和任务 2。80 端口是服务器把含有 Java Applet 的 HTML 页面发送给浏览器的通道,3000 端口是服务器与客户端进行数据通信的通道。

2)任务 1 部分。任务 1 的流程如图 5 所示,它的主要功能是每隔一段固定的时间对被控对象进行测量、控制并且把要发送给客户端的数据保存在一个数组中。这个任务中我们用 waitfor(DelayMs())作为延时函数,它是一个非阻塞函数,如果没有达到延时要求的时间,它将会跳出任务 1,程序直到任务 1 或者任务 2 中有一个满足要求,才会执行下去。这样保证了整个系统的实时性。

3)任务 2 部分。任务 2 的流程如图 6 所示,它的主要功能是接收由客户端发送过来的请求,判断是何种请求并做出相应的操作。在这个任务的开始调用了函数 http_handler(),它是 Dynamic C 自带的函数,用来运行 http 后台程序。这个任务还会判断套接字 3000 端口的状态,如果 3000 端口的状态由“Listen”变为“Established”,则说明客户

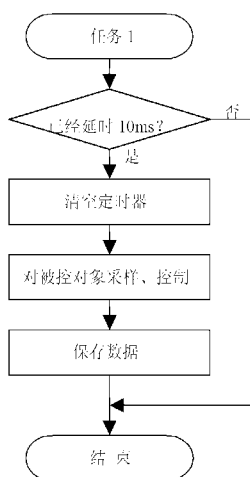


图 5 测控程序流程图

端已经建立了与服务器的连接,当缓冲区收到客户端传来的完整的字符串后,程序会把缓冲区中的字符串读出来,再对其进行解析和操作。客户端收到满意的数据后,便断开与服务器的连接,服务器会检测到此时 3000 端口的状态为“Closed”,这时服务器会重新把 3000 端口的状态设为“Listen”,等待客户端建立下一次的连接。

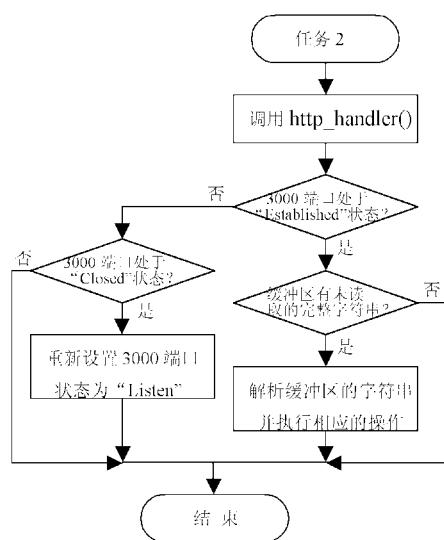


图 6 通信程序流程图

参考文献

- 1 Z-World,Rabbit 2000 Microprocessor User's Manual, 2001
- 2 Z-World, Dynamic C TCP_IP User's Manual,2001
- 3 张根源.嵌入式系统与 Internet 技术.微计算机信息,2000(3)
- 4 张晶,等.嵌入式系统的应用.电气时代,2002(5)
- 5 (美)Morgan Stern,Tom Rasmosen 著.邸瑞华,冯国臻,等译.Intranet 组网方式.电子工业出版社,1998(8)

[收稿日期:2002.10.14]