

Abstract

LonWorks technical platform has been developed by Echelon corporation since the 1990s. Now it becomes the general standard of the control network. LonWorks network can monitor the sensors, carry out the reliable communication management and network operation by using LonTalk protocol. LonWorks network mode can help the manufacture cut cost, reduce developing period as well as increase mutual-operational capability.

In this system we make some modifications on the LonWorks network mode to suit to the actuality of the buildings intelligentized control system in China. The system uses three hierarchical network to connect all the equipments instead of the simplex LonWorks network. We use the RS-485 serial communication mechanism at the bottom of the system, while the CAN bus in the middle and the LonWorks network mode was only used on the top of the system.

By using the new network structure the system can carry out the functions as below: The system monitor and maintenance the fire protection equipment regularly or irregularly in the usual time, whereas as the fire alarming, the system would unfold the scaling ladder, turn on the lamp and help the people in danger evacuate.

Keywords: LonWorks LonTalk LNS CAN RS-485 TCP/IP

1 绪论

1.1 系统开发背景及必要性介绍^{〔17〕〔19〕}

消防监控与智能逃生系统时刻关系到人们的生命财产安全，它是衡量楼宇智能化程度的重要标准之一。在目前绝大部分的智能楼宇中，门禁、抄表、安防报警等家庭模块的监控管理系统已经可以集成为一套整体的控制系统，使用共同的网络资源进行集中监控报警，然而消防逃生系统仍独立于现有的监控体系单独存在，往往每栋大楼都要分派专门人员进行 24 小时的轮班值守，这给小区的管理人员带来了很多的麻烦，不但浪费了人力物力，也使消防信息的管理水平的提升受到很大限制。

随着网络技术的发展，利用网络实现各大楼消防分系统的联网，把需要监控的一系列楼宇连成一个消防网络，同时将其嵌入到已有的家庭智能模块监控网络中，就可以在中央控制室（总控中心）监控各家庭模块信息的同时监控消防信息变化的情况，这样就只需在总控中心安排值守人员，既节约了人力物力，又保证了消防信息传递和响应的快速性、准确性，还提高了消防信息管理的水平，顺应了时代对网络化便捷化的要求。^{〔17〕〔19〕}

1.2 消防应急与逃生联网系统介绍^{〔19〕〔24〕}

消防应急与逃生联网系统通过在每户家庭安装温度、烟道红外探测装置和消防逃生梯可以定期启动远程监视和巡检功能实时监测各个防范对象的火警信息。如有突发性事件发生，系统可以及时采集处理事故现场的数据和反馈的信息，生成智能救生信息，启动救生联动设施（逃生梯、应急灯等），引导被困人员安全逃生，同时通过短信及大屏显示警情，指挥搜救人员快速救援，从而最大限度地抢救现场被困人员的生命。此外，系统还可以扩展为训练模式，定期对相关人员进行训练和实施现场演习。通过实施早期的督察、监视、巡检确保防范对象的安全措施到位以及安防设施运行正常，而在突发事件发生时则着重于现场被困人员的安全疏散、逃生引导和救生。它实际上是一个具有远程监视、现场探测、数据

分析、引导逃生、现场救助等功能的智能化消防应急与逃生联网系统。^[19]

系统的主要功能如下：

1. 远程监控：系统安装有较完善的现场监测子系统，具备消防逃生通道实时监视和救生设施巡检功能。在平时，系统可以定期或不定期地检测楼宇内家庭火警信息，楼层安全疏散走道和火警探测设备的情况，防止有通道堵塞和火警探测设备损坏的情况发生，保证各通道畅通。同时系统也能对相关消防逃生梯、应急灯等救生设备进行自我诊断校验，保证整个系统的正常运行。
2. 现场探测：在突发事件发生后，系统能及时监测事故现场各安全疏散走道和消防逃生设备的完好情况、烟雾和有害气体的浓度、环境温度等数据，同时也能测定事故现场被困人员的人数、位置和状态等情况。
3. 数据分析：系统可将事故现场采集的数据实时传送给总控中心进行分析、处理，并加以保存备案。今后也可在此系统基础上扩展开发现场模拟和指挥决策软件，调用人员安全疏散模型，生成救生、引导、联动、指挥等相关信号和信息。
4. 现场救助：系统经对事故现场采集的数据和用户反馈的信息进行处理后，一方面启动救生联动设施，如打开逃生梯、应急灯等，另一方面则向专业救援人员传递准确的救援指挥信息，如采用电话、广播、短信等方式，协助其及时对现场被困人员实施救助。
5. 引导逃生：智能救生系统能通过信息发布和反馈子系统向事故现场定向发布安全逃生引导信息。事故现场的每个被困人员可以借助各类信息接收终端（例如：消防广播、个人电脑、PDA、手机、公告牌、指示牌、能接收信息的 ID 卡等）接收到准确的安全逃生引导信息，以便及时安全逃生。

消防应急与逃生联网系统的特点：

1. 技术先进：系统采用分布式结构。整个系统采用了 3 级总线：RS-485，CAN，LonWorks。三种总线技术各不相同，家庭模块级采用 RS-485 主要考虑降低成本和设计复杂度；单元级采用 CAN 主要考虑成本、响应速度和传输可靠性；小区级采用 LonWorks 主要考虑可扩展性、通用性、兼容

性（与前一系列产品的兼容）和可维护性。同时，在管理员主机中加载远程监控管理系统，支持 TCP/IP 和 UDP 协议。

2. 可靠性高：通信链路采用有线、无线等多种通信方式，并互为备份，确保数据传输不会因偶发事故而中断。系统控制中心的信息处理主机采用现场和远程双重配置方案，从而实现了远程和现场的双余度控制。
3. 开放性结构：系统采用模块化设计和开放性结构，可与其它家庭模块监控管理系统无缝连接并进行数据交换。因而消防应急与逃生联网系统既能独立作为一个专业的安防系统，也可作为现有智能楼宇监控管理系统中的一个子系统存在。
4. 操作性好：系统的应用程序采用视窗应用界面，人机对话友好，使用人员易学易用。^[24]

1.3 现场总线技术介绍^{[6] [7] [34] [35]}

上世纪 80 年代提出了现场总线概念，原始的现场总线思想只是针对传统的 DCS 问题试图用一个开放的、独立于产品的多回路通信系统来取代 mA 的电流标准。经历十几年的发展，由于这期间飞速发展的电子技术和计算机技术，现场总线技术的内涵被不断延伸。目前现场总线的重要性已经被越来越多的人所重视。

现场总线的协议是依据国际标准化组织 ISO 的开放系统互连协议 OSI。OSI 协议为计算机制定了 7 层参考模型，只要网络中所有要处理的要素都是通过共同的路径进行通信就可与使用该协议。^{[34] [35]}

	OSI 层	目的	服务
1	应用	应用程序	标准对象和类型、配置属性、文档转移、网络服务
2	显示	数据解释	网络变量、应用、报文、外来帧
3	对话	远程行动	对话、远程程序调用、连接恢复
4	传输	端到端可靠性	端到端确认、业务类型、包排序、双重检测
5	网络	目的地寻址	单路传输和多路传输、目的地寻址、包路由选择
6	数据链路	介质访问和组帧	组帧、数据编码、CRC、介质访问、冲突检测
7	物理	电互联	介质特定细节、收发器类型、物理连接

图 1.1 ISO/OSI 七层网络协议模型^[6]

从物理结构来看，现场总线系统有两个主要的组成部分：一是现场设备；二是形成系统的传输介质。现场设备由现场微处理芯片以及外围电路构成。现场总线系统使用最多的传输介质是双绞线。现场总线系统的拓扑结构有很多种：总线型、环型、树型、星型以及混合型等。

现场总线目前也没有统一的标准，许多厂家出于各自利益的考虑，都在研究各种各样的现场总线产品，先后有多种现场总线的企业标准、集团标准以及国家标准。当然，各厂家自己制定的通信协议并非都实现了 OSI 的 7 层协议，而是根据实际情况侧重点的不同，实现其中的几层。这就使现场总线产品失去了统一的标准，其中比较有影响力的有以下几个：

1. ISA/SP50

它是美国仪表学会 ISA 下属的标准实施第 50 组所制定的现场总线标准。它对现场总线的定义是这样的：“现场总线是一种串行的数字数据通信链路，它完成了过程控制领域的基本控制设备如传感器、控制器、智能阀门、微处理器、存储器等仪表产品之间的沟通以及与更高层次自动控制领域的自动化控制设备之间的联系。”

SP50 的最大特点是物理层上使用双绞线，用于传输数字信号的同时也用于为现场供电。这是利用“两相曼彻斯特编码”技术实现的。SP50 的数据链路层遵循 LAS 协议，是一种集中调度管理机制。其工作是按令牌的循环传递

进行的, 最高传输速率可达 2.5mb/s, 最大传输距离是 2km, 每条双绞线可连接节点 32 个。

2. PROFIBUS

PROFIBUS 也就是过程现场总线, 它是德国制定的总线标准。PROFIBUS 是一种多主从的令牌网络。物理层使用 EIA485, 数据链路层为分布式令牌协议, 每个系统最多为 4 个网段, 每个网段可接 32 个节点。其通信速率最高可达到 500kb/s, 最远通信距离是 1.2km。

PROFIBUS 引入了功能模块的概念, 不同的应用需要使用不同的模块, 在一个确定的应用中, 按照规范来定义功能模块, 写明其硬件、软件性能, 设备功能与通信性能的一致性。

3. ISP 和 ISPE

它是由 Siemens、Foxboro、Rosemount、Fisher、Yokogawa、ABB 等公司成立的 ISP 组织以 PROFIBUS 为基础制定的。

4. WorldFIP

它是由 Honeywell、Bailey 等公司以法国标准 FIP 为基础制定的现场总线标准。WorldFIP 只是一个高层协议, 不包括物理层以及链路层的规范, 而高层协议称为 FIP。

5. HART 和 HCF

HART 是由 Rosemount 提出的可寻址远程传感器数据链路通信协议, 它是一种将模拟信号调制成数字调频信号, 利用双绞线实现数字调频信号的传输协议。HART 协议参照 ISO/OSI 模型中的第 1、2、7 层, 即物理层、数据链路层和应用层。其中物理层采用基于 Bell202 通信标准的 FSK 技术; 数据链路层的数据帧长度不固定, 寻址范围为 0~15, 通信模式为问答式或广播式; 应用层规定了三类命令: 第一类是通用命令, 适用于遵守 HART 协议的所有产品; 第二类是普通命令, 适用于遵守 HART 协议的大部分产品; 第三类是特殊命令, 适用于遵守 HART 协议的特殊产品。另外还为用户提供了设备描述语言 DDL。

6. FF

FF 即现场总线基金会, 它是 1994 年 ISPF 和 WorldFIP 共同成立的, 是

国际公认的唯一一个不隶属于任何企业的公正的非商业化的国际标准化组织，它的宗旨是制定一个单一的国际现场总线标准，无专利许可要求，可供任何人使用。

FF 对现场总线的定义是这样的：现场总线是连接智能现场设备和自动化系统的数字式、双向传输、多分支结构的通信网络，其本质表现在 6 个方面：

- 现场通信网络：用于过程以及制造自动化的现场设备或现场仪表互连的通信网络。
- 现场设备互连：现场设备或现场仪表通过一对传输线互连，传输线可以是双绞线、同轴电缆、光纤和电源线等，并可根据需要因地制宜的选择不同类型的传输介质。
- 互操作性：现场设备或现场仪表种类繁多，没有任何一家制造商可以提供全部现场设备，所以，互相连接不同厂商的产品是不可避免的。用户希望选用各制造商性价比最优的产品，并将其集成在一起，对不同品牌的现场设备统一组态、构成他所需要的控制回路。
- 分散功能块：FCS 废弃了 DCS 的输入/输出单元和控制站，把 DCS 控制站的功能分散的分配给现场仪表，从而构成虚拟控制站。
- 通信线供电：通信线供电方式允许现场仪表直接从通信线上摄取能量，对于要求本征安全的低功耗现场仪表，可采用这种供电方式。
- 开放式互连网络：现场总线为开放式互连网络，它既可与同层网络互连，也可与不同层网络互连，还可与实现网络数据库的共享。通过网络对现场设备和功能统一组态，把不同厂商的网络以及设备融为一体，构成统一的 FCS。

对 FF 标准的分析可以从以下几方面入手：

1) FF 体系结构

除参照 ISO/OSI 模型的物理层、数据链路层和应用层外，还增加了用户层。

2) 物理层 PL

H1 为用于过程自动化的低速总线，波特率为 31.25kb/s，传输距离 (200~1900) m。由总线供电、提供本质安全型。

H2 为用于制造自动化的高速总线，波特率为 1.0Mb/s 或 2.5Mb/s，传输介质可以是双绞线、同轴电缆、光纤和无线。

H1 每段节点数最多为 32 个，H2 每段节点数最多为 124 个，H1 与 H2 之间通过网桥互连。

H1 的拓扑结构为总线型或树型；H2 为总线型。IEC 已批准物理层标准，目前已有多家公司生产 H1 低速总线的专用芯片。

3) 数据链路层 DLL

DLL 低层的基本设备 BD 功能不能主动发起通信，只能接收查询；链路主设备 LMD 在得到令牌时可以发起一次通信；每个网段的 LMD 中有一个链路活动调度器 LAS 发起周期和非周期性的通信。

DLL 高层数据传输功能可连接数据传输、发行数据定向连接传输，请求/响应数据定向连接传输。

4) 现场总线访问子层 FAS

FAS 提供 3 类服务：发布/索取、客户机/服务器和报告分发服务。这三类服务被称为虚拟通信管辖 VCR。

5) 现场总线报文规范 FMS

FMS 规定了访问应用进程 AP 的报文格式及服务，FMS 与对象字典 OD 配合，为现场设备规定了功能接口。FMS 通过调用 VCR 在现场设备之间传递报文。

上述 FAS 和 FMS 构成应用层 AL。

6) 用户层 UL

用户层规定了 29 个标准的功能模块 FB，其中基本功能模块 10 个，先进功能模块 7 个，计算功能模块 7 个，辅助功能模块 5 个。功能模块由输入、输出、算法、参数四大要素组成。功能模块参数分为三个层次：第一层由 FF 规定；第二层由用户基团定义；第三层由制造厂商定义。

为了支持功能模块模型的标准化和互操作性，FF 还定义了两个工具：设备描述语言 DDL 和对象字典 OD，用来定义和描述应用层的网络可见的对象如功能块及其参数，另外还有网络管理和系统管理。⁽⁷⁾

1.4 基于 Internet 的远程控制介绍^{[14][28][30]}

系统中的远程监控子系统采用 Internet 的远程控制, 支持 TCP/IP 和 UDP 协议。使用 Visual C++ 中的 CSocket 类, 将系统构架设计为 C-S 结构。

TCP/IP 协议是一组包括 TCP 协议和 IP 协议, UDP (User Datagram Protocol) 协议、ICMP (Internet Control Message Protocol) 协议和其他一些协议的协议组。

TCP/IP 协议并不完全符合标准的 OSI 七层参考模型。传统的开放式系统互连参考模型, 是一种通信协议的 7 层抽象的参考模型, 其中每一层执行某一特定的任务。该模型的目的是使各种硬件在相同的层次上相互通信。这 7 层分别是: 物理层、数据链路层、网路层、传输层、会话层、表示层和应用层。而 TCP/IP 通讯协议采用了 OSI 中的 4 层层级结构, 每一层都呼叫它的下一层所提供的网络来完成自己的需求。这 4 层分别为: 应用层: 应用程序间沟通的层, 如简单电子邮件传输 (SMTP)、文件传输协议 (FTP)、网络远程访问协议 (Telnet) 等; 传输层: 它提供了节点间的数据传送服务, 如传输控制协议 (TCP)、用户数据报协议 (UDP) 等, TCP 和 UDP 向数据包中加入传输数据并把它传输到下一层中, 传输层负责传送数据, 并且确定数据已被送达并接收; 网络层: 负责提供基本的数据封包传送功能, 每一块数据包都能够被传递到目的主机, 但网络层并不检查数据包是否被正确接收, 如网际协议 (IP); 网络接口层: 对实际的网络媒体的管理, 定义如何使用实际网络 (如 Ethernet、Serial Line 等) 来传送数据。

网际协议 IP 是 TCP/IP 和网络层中最重要的协议。IP 层接收由更低层 (网络接口层例如以太网设备驱动程序) 发来的数据包, 并把该数据包发送到更高层如 TCP 或 UDP 层; 同样的, IP 层也把从 TCP 或 UDP 层接收来的数据包向下传送到更低层中去。IP 数据包是不可靠的, 因为 IP 并没有做任何事情来确认数据包是否是按顺序发送的或者有没有被破坏。IP 数据包中含有发送它的主机的地址 (源地址) 和接收它的主机的地址 (目的地址)。

高层的 TCP 和 UDP 服务在接收数据包时, 通常假设包中的源地址是有效的。也可以这样说, IP 地址形成了许多服务的认证基础, 这些服务相信数据包是从一个有效的主机发送来的。IP 确认包含一个叫做 IP source routing 的选项,, 可以用来指定一条源地址和目的地址之间的直接路径。对于一些 TCP 和 UDP 的服

务来说，使用了该选项的 IP 包好像是从路径上的最后一个系统传递过来的，而不是来自于它的真实地点。这个选项是为了测试而存在的，说明了它可以被用来欺骗系统进行平常被禁止的连接。这样许多依靠 IP 源地址做确认的服务将产生问题并且会被非法入侵。

如果 IP 数据包中含有已经封好的 TCP 数据包，那么 IP 将把它们向‘上’传送到 TCP 层。TCP 首先将包排序并进行错误检查，同时实现虚电路间的连接。TCP 数据包中包括序号和确认，所以可以把未按照顺序接收的包排序，并重传损坏的包。

TCP 可以将它的信息送到更高层的应用程序，例如 Telnet 的服务程序和客户程序。应用程序轮流将信息送回 TCP 层，TCP 层便将它们向下传送到 IP 层，设备驱动程序和物理介质，最后到接收方。

面向连接的服务（例如 Telnet、FTP、rlogin、X Windows 和 SMTP）需要高度的可靠性，所以它们使用了 TCP。DNS 在某些情况下使用 TCP（发送和接收域名数据库），但使用 UDP 传送有关单个主机的信息。

UDP 与 TCP 位于同一层，但没有对于数据包的顺序错误或重发。因此，那些使用虚电路的面向连接的服务不使用 UDP，UDP 主要用于那些面向查询一应答的服务，例如 NFS。相对于 FTP 或 Telnet，这些服务需要交换的信息量较小。使用 UDP 的服务包括 NTP（网落时间协议）和 DNS（DNS 也使用 TCP）。

欺骗 UDP 包比欺骗 TCP 包更容易，因为 UDP 没有建立初始化连接（也可以称为握手）（因为在两个系统间没有虚电路），也就是说，与 UDP 相关的服务面临着更大的危险。

TCP 和 UDP 服务通常存在一个客户/服务器的关系，例如，一个 Telnet 服务进程开始在系统上处于空闲状态，等待着连接。用户使用 Telnet 客户程序与服务进程来建立一个连接。客户程序向服务进程写入信息，服务进程读出信息并发出响应，客户程序读出响应并向用户报告。因而，这个连接是双工的，可以用来进行读写。

两个系统间的多重 Telnet 连接是如何相互确认并协调一致呢？TCP 或 UDP 连接唯一地使用每个信息中的如下四项进行确认：

表 1.1 TCP/UDP 确认信息项列表^[28]

源 IP 地址	发送包的 IP 地址
目的 IP 地址	接收包的 IP 地址
源端口	源系统上的连接的端口
目的端口	目的系统上的连接的端口

端口是一个软件结构，客户程序或服务进程用它来接收和发送信息。一个端口对应一个 16bit 的数。服务进程通常使用一个已分配好的固定端口，例如，SMTP 使用 25、Xwindows 使用 6000。这些端口号是‘广为人知’的，因为在建立与特定的主机或服务的连接时，需要这些地址和目的地址进行通讯。^[14]

1.5 CSocket 类的远程访问技术介绍^{(29) (30) (32)}

客户端与服务端的通信简单来讲是服务端 socket 负责监听，应答，接收和发送消息，而客户端 socket 只是连接，应答，接收，发送消息。

1. 网络传输服务提供者(网络传输服务进程)，Socket 事件,Socket Window

网络传输服务提供者 (transport service provider) 是以 DLL 的形式存在的，在 windows 操作系统启动时由服务进程 svchost.exe 加载。当创建 socket 时，调用 API 函数 Socket (在 ws2_32.dll 中)，Socket 函数将传递三个参数：套接字类型，地址族和协议，由哪一个类型的网络传输服务提供者来启动网络传输服务功能是由这三个参数决定。所有的网络通信正是由网络传输服务提供者完成，网络传输服务提供者是由 svchost.exe 服务进程所加载的。

下图描述了网络应用程序、CSocket (WSock32.dll)、Socket API(ws2_32.dll)和网络传输服务进程之间的接口层次关系：

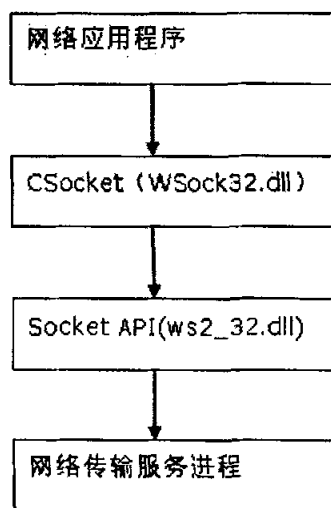


图 1.2 网络应用程序、CSocket、Socket API 和网络传输进程间的接口层次关系^[30]

当客户端 socket 与服务器端 socket 相互通信时，两端均会触发 socket 事件：这里仅简要说明两个 socket 事件：

- 1) FD_CONNECT：连接事件。通常客户端 socket 调用 socket API 函数的 Connect 时所触发，这个事件发生在客户端；
- 2) FD_ACCEPT：正在引入的连接事件。通常服务器端 socket 正在接收来自客户端 socket 连接时触发，这个事件发生在服务器端

socket 事件被网络传输服务进程保存至 socket 的事件队列中。此外，网络传输服务进程还会向 socket window 发送消息 WM_SOCKET_NOTIFY，通知有 socket 事件产生：调用 CSocket::Create 函数后，创建 socket。socket 创建过程中调用 CAsyncSocket::AttachHandle(SOCKET hSocket, CAsyncSocket* pSocket, BOOL bDead)。这一函数的作用是：a. 将 socket 实例句柄和 socket 指针添加至当前模块状态的一个映射表变量 m_pmapSocketHandle 中；b. 在 AttachHandle 过程中，会新建一个 CSocketWnd 实例（基于 CWnd 派生），这里将这个实例称之为 socket window，进一步理解为它是存放所有 sockets 的消息池（window 消息），这里 socket 后多加了一个 s，表示创建的多个 socket 将共享一个消息池；c. 当客户端 socket 与服务器端相互通信时，网络传输服务进程会向 socket window 发送消息 WM_SOCKET_NOTIFY，需要说明的是 CSocketWnd 窗口句柄保存在当前模块状

态的 `m_hSocketWindow` 变量中。

2. 阻塞模式

阻塞模式下服务器端与客户端之间的通信处于同步状态下。在服务器端直接实例化 `CSocket` 类,调用 `Create` 方法创建 socket,然后调用方法 `Listen` 开始侦听,最后用一个 `while` 循环阻塞调用 `Accept` 函数用于等待来自客户端的连接,如果这个 socket 在主线程(主程序)中运行,这将导致主线程的阻塞。因此,需要创建一个新的线程以运行 socket 服务。调试跟踪至 `CSocket::Accept` 函数源码:

```
while(!Accept(...))
{
    if (GetLastError() ==WSAEWOULDBLOCK)
        PumpMessage(FD_ACCEPT);
    else
        return FALSE;
}
```

它不断调用 `CAsyncSocket::Accept` (`CSocket` 派生自 `CAsyncSocket` 类)判断服务器端 socket 的事件队列中是否存在正在引入的连接事件—`FD_ACCEPT`,换句话说,就是判断是否有来自客户端 socket 的连接请求。如果当前服务器端 socket 的事件队列中存在正在引入的连接事件, `Accept` 返回一个非 0 值。否则, `Accept` 返回 0,调用 `GetLastError` 将返回错误代码 `WSAEWOULDBLOCK`,表示队列中无任何连接请求。在循环体内有一句代码: `PumpMessage(FD_ACCEPT)`; `PumpMessage` 作为一个消息泵使得 socket window 中的消息能够维持在活动状态。实际跟踪进入 `PumpMessage` 中,发现这个消息泵与 `Accept` 函数的调用并不相关,它只是使很少的 socket window 息(典型的是 `WM_PAINT` 窗口重绘消息)处于活动状态,而绝大部分的 socket window 消息被阻塞,被阻塞的消息中含有 `WM_SOCKET_NOTIFY`。

很显然,如果没有来自客户端 socket 的连接请求, `CSocket` 就会不断调用 `Accept` 产生循环阻塞,直到有来自客户端 socket 的连接请求而解除阻塞。阻塞解除后,表示服务器端 socket 和客户端 socket 已成功连接,服务器端

与客户端彼此相互调用 Send 和 Receive 方法开始通信。

3. 非阻塞模式

在非阻塞模式下利用 socket 事件的消息机制。此时服务器端与客户端之间的通信处于异步状态下。通常需要从 CSocket 类派生一个新类，派生新类的目的是重载 socket 事件的消息函数，然后在 socket 事件的消息函数中添加合适的代码以完成服务器端与客户端之间的通信，与阻塞模式相比，非阻塞模式无需创建一个新的线程。

这里将讨论当触发服务器端 socket 事件—FD_ACCEPT 后，该事件的处理函数 OnAccept 是如何进一步被触发的。其它事件的处理函数如 OnConnect, OnReceive 等的触发方式与此类似。

在 1 中已提到服务器/客户端通信时，服务器端 socket 正在接收来自客户端 socket 连接请求，这将会触发 FD_ACCEPT 事件，同时服务器端的网络传输服务进程向服务器端的 socket window(CsocketWnd) 发送事件通知消息 WM_SOCKET_NOTIFY，通知有 FD_ACCEPT 事件产生，CsocketWnd 在收到事件通知消息后，调用消息处理函数 OnSocketNotify:

```
LRESULT CSocketWnd::OnSocketNotify(WPARAM wParam, LPARAM lParam)
{
    CSocket::AuxQueueAdd(WM_SOCKET_NOTIFY, wParam, lParam);
    CSocket::ProcessAuxQueue();
    return 0L ;
}
```

消息参数 wParam 是 socket 的句柄，lParam 是 socket 事件。CSocketWnd 类是作为 CSocket 类的友元类，这意味着它可以访问 CSocket 类中的类和私有类成员函数和变量，AuxQueueAdd 和 ProcessAuxQueue 是 CSocket 类的静态成员函数。ProcessAuxQueue 是实质处理 socket 事件的函数，在该函数中有这样一句代码：

```
CAsyncSocket* pSocket = CAsyncSocket::LookupHandle((SOCKET)wParam,
TRUE);
```

其实也就是由 socket 句柄得到发送事件通知消息的 socket 指针

pSocket:从 m_pmapSocketHandle 中查找。最后,WSAGETSELECTEVENT(lParam) 会取出事件类型,在一个简单的 switch 语句中判断事件类型并调用事件处理函数。在这里,事件类型是 FD_ACCEPT,当然就调用 pSocket->OnAccept!

服务器端 socket 处于阻塞调用模式下,它必须在一个新创建的线程中工作,防止主线程被阻塞。当有多个客户端 socket 与服务器端 socket 连接及通信时,服务器端采用阻塞模式就显得不适合了,应该采用非阻塞模式,利用 socket 事件的消息机制来接受多个客户端 socket 的连接请求并进行通信。在非阻塞模式下,利用 CSocketWnd 作为所有 sockets 的消息池,是实现 socket 事件的消息机制的关键技术。

其中:“当前模块状态”用于保存当前线程和模块状态的一个结构,可以通过 AfxGetThreadModule() 获得。AFX_MODULE_THREAD_STATE 在 CSocket 重新定义为_AFX SOCK_THREAD_STATE;“socket 类型”在 TCP/IP 协议中,服务器/客户端网络程序采用 TCP 协议:即 socket 类型为 SOCK_STREAM,它是可靠的连接方式。在这里不采用 UDP 协议:即 socket 类型为 SOCK_DGRAM,它是不可靠的连接方式。^[32]

2 系统原理介绍

2.1 性能要求

系统各设备主要技术指标:

1. 智能消防砖:

供电电源: DC24V;

接口数量: 4 个;

警铃声级: >85dB(A);

软梯承重: >200Kg。

2. 智能应急灯

供电电源: DC24V;

接口数量: 4 个;

灯光功率: >40W;

持续时间：>20min。

3. 单元控制器：

网络接口形式：LonWorks；

驱动节点数量：24 个；

数据交换速率：<1mS/帧；

输出电源电压：DC24V；

输出电源电流：8A。

4. 区域控制器：

网络网接口形式：LonWorks；

控制网节点容量：5000；

控制网传输速率：78KBPS；。

5. 远程监控主机：

网络媒介：Internet；

传输速率：>56KBPS。

2.2 系统工作原理

2.2.1 系统构架

整个消防逃生系统分为两部分：消防应急与逃生联网控制系统和远程监控子系统。其中消防应急与逃生联网控制系统与原有的家庭智能模块控制系统集成为一套完整的小区智能楼宇控制系统，共享网络资源，在无火警时，它可作为小区智能楼宇控制系统的一部分与其它模块共同接受定期巡检和维护，可有效节省人力物力以及网络资源；当火警发生时，它可作为单独控制系统独占网络资源响应火警，这样大大提高了系统效率。

消防逃生的远程监控子系统可作为原有的家庭模块远程监控系统的一部分，管理员可在查询、设置家庭模块信息的同时对消防逃生模块信息进行查询和更新。

2.2.2 LonWorks 总线原理及组网模式 ^{(1) (2) (3) (4) (5) (6) (9)}

LonWorks 技术平台是在 80 年代后期由 Echelon 公司开始开发并逐渐成为控制网的通用标准。该平台是一个开放的控制网络平台，它提供一个平坦的、对等式的控制网络架构，给各种控制网络应用提供端到端的解决方案。它把系统中的一次元件，如传感器、执行元件、断路器等，通过基于 LonTalk 协议采用自由拓扑结构方式把控制单元连接成开放的测控网络，这个网络符合国际标准化组织（ISO）制定的开放系统互联（OSI）模式。LonWorks 技术平台追求网络的全面控制，包含了传感器总线、设备总线和现场总线三层意义，实用性很强。目前，这一技术被广泛应用于各种设备和系统产品中，遍及全世界的楼宇、工厂、家庭、火车和飞机等领域，并在家庭、火车、半导体制造业、智能楼宇、加油站和货运列车制动系统领域，是理想的现场实时控制网络。

该技术有一些主要特点：首先，它是完全开放式的技术。开放系统就是使用开放系统相互连接（OSI）基准模型（块）以发展计算机通信标准。OSI 模型是建立在七层协议基础上，每层都有一定等级功能，具有规定的高层或低层的接口；为提供一定功能通信标准，并不一定所有的层都需要。当与定义好的程序模块连接时，该模块定义了数据意义及格式，OSI 模块将提供一个多卖主互操作的工具。一个典型的开放系统结构可以应用在工业和商业的控制系统上，按照布置，所有的开放系统元件，使用标准协议作为本系统语言，无需翻译，可以相互通信。^[6]

国际标准化组织（ISO）所定义的开放系统互连模型 OSI 作为通信网络国际化的参考模型，它详细描述了软件功能的七个层次。如下表所示：

表 2.1 ISO/OSI 七层网络协议^[6]

A 实体					B 实体	
	OSI 层次	标准服务	中继系统	处理器	OSI 层次	
7	应用层	网络应用		应用处理器	应用层	7
6	表示层	数据表示		网络处理器	表示层	6

5	会话层	远程遥控动作		网络处理器	会话层	5
4	传送层	端对端的可靠传输		网络处理器	传送层	4
3	网络层	传输分组	网络层	网络处理器	网络层	3
2	链路层	介质访问, 帧结构	数据链路层	MAC 处理器	链路层	2
1	物理层	电路连接	物理层	MAC 处理器, XCVR	物理层	1
物理介质						

LonWorks 技术提倡运用分布式的智能设备组建控制网络,同时也支持主从式网络结构。控制网络的核心部分—LonTalk 通讯协议,即 ANSI/EIA709.1-A-1999 已经固化在了神经元芯片之中。该技术包括一个称之为 LNS 网络操作系统的管理平台,该平台对 LonWorks 控制网提供全面的管理和服务。LonWorks 网又可通过各种连接设备接入 IP 数据网络和互联网,与信息技术应用实现无缝的结合。该技术的另一个重要特点是它的互操作性。国际 LonMark 互操作性协会推广、制定基于 LonWorks 技术的具有互操作性的标准(LonMark 标准)。符合该标准的设备,无论来之什么生产厂商都可集成在一起,形成多厂商、多产品的开放系统。系统的整体和周期成本较传统的封闭系统低很多。

LonTalk 通信协议是 LonWorks 技术的核心。该协议提供一套通信服务,使装置中的应用程序能在网上对其他装置发送和接收报文而无需知道网络拓扑、名称、地址或其他装置的功能。LonTalk 协议能有选择的提供端到端的报文确认、报文证实、和优先级发送以便设定有界事务处理时间。对网络管理业务的支持使远程网络管理工具能通过网络和其他装置互相作用,包括网络地址和参数的重新配置、下载应用程序、报告网络问题和节点应用程序的起始/复位。

LonTalk 协议是一个分层的以数据包为基础的对等的通信协议。如同有关的以太网和因特网协议一样,它是一个公布的标准并遵守国际标准化组织 ISO 的分层体系结构要求。可是, LonTalk 协议设计用于控制系统而不是数据处理系统的特定的要求。每个包由可变数目的字节构成,长度必定,并且包含应用层的信息以及寻址和其他信息。信道上的每个装置监视在信道上传输的每个包以确定自己是否是收信人。为了处理网上报文冲突, LonTalk 使用类似以太网所用的“载波

监听多路访问”(CSMA)算法。LonTalk 协议建立在 CSMA 基础上,提供介质访问协议,使得可以根据预测网络业务量发送优先级报文和动态调整时间槽的数目。通过动态调整网络带宽,称为预测性 P-persistent CSMA 协议的算法使网络能在极高网络业务量出现时继续运行,而在业务量较少时期不降低网络速度。

LonWorks 技术的最高目标是方便和经济地建立开放控制系统。在控制设备市场上创造互可操作产品有 3 个基本问题必须解决。首先,必须开发一个协议,它针对控制网优化,同时又具有一定程度的通用性来和各种类型的控制设备一起工作。其次,把协议结合和布置在装置中的成本必须有竞争力。第三,协议的推行方式应使其实施不会因销售商而异,否则互可操作性会受到破坏。为了解决所有这些问题,Echelon 公司开始建立一个完整平台来设计、建造和安装智能控制设备。

为了经济、标准化布置,Echelon 设计了神经元芯片。就开发者和集成员来说,神经元芯片之优越性在于它的完整性。内装协议和处理器免除了在这些方面的任何开发和编程。它提供了通信协议 ISO/OSI 模型的前面六层,只需要提供应用层编程和配置,这就使协议的实施标准化,并使开发和配置较为容易。神经元芯片是一种半导体装置,基本上是一个“芯片上的系统”,由多个微处理器,读写存储器和只读存储器(RAM 和 ROM)、通信和 I/O 接口组成。只读存储器包含操作系统、LonTalk 通信协议和 I/O 功能数据库系统。芯片有用于装置数据和应用程序的非易失性 RAM,两者都可以从通信网络上下载。

一个全面实施 LonTalk 协议的 LonTalk 固件的程序包含在每个神经元芯片的 ROM 中。大部分 LonWorks 装置包括一个具有完全相同的嵌入式 LonTalk 协议实施工具的神经元芯片。这个方法解决了“99%兼容性”问题,并保证 LonWorks 装置在同一网络上的连接只需要很少甚至不需要很少甚至不需要额外的硬件。神经元芯片实际上是结合成一体的 3 个 8 位的联机处理器。其中两个优化以执行协议;第三个供节点应用。所以,芯片既是网络通信处理器又是应用处理器。这保证了不论控制装置/网络来自哪个制造商,这些装置能相互通信的内在协议是相同的。

每个神经元芯片,或任何其他实施已公布的 LonTalk 协议的,允许复制件的处理器都有一个被保证是唯一的 48 位 ID。这样,每个 LonWorks 装置就有唯一

的可由 LonTalk 协议使用的物理地址。但是，为了简化正常网络运行，使用了逻辑寻址方法。

收发器在神经元芯片和 LonWorks 网络间提供物理通信接口。这些装置简化了互可操作的 LonWorks 节点的开发，可用于各种通信介质和拓扑。结合在网络上的每个 LonWorks 装置或节点通常含有位于适当的机械封装中的神经元芯片和收发器。

对多种介质的透明支持是 LonWorks 技术的独特能力，它使开发者能选择最适合他们需要的介质和通信方法。对多种介质的支持通过路由器才可能。路由器也能用于控制网络业务量，将网络分段，抑制从其他部分来的数据流量，从而增加了网络总通过量和容量。网络工具以网络拓扑为基础自动配置路由器，使安装者便于安装并对节点透明。路由器装置使单一的对等网络能跨接许多类型传输介质，支持成千上万的装置。路由器通常有 2 个互联的神经元，每个神经元有一个适用于 2 个信道的收发器，路由器就连接在这 2 个信道上。路由器对网络的逻辑操作是完全透明的，但是它们并不一定传输所有的包。智能路由器充分了解系统配置，能将没有远地地址的包闭塞。使用了另一类型叫做穿越路由器的路由器，LonWorks 系统能在因特网这样的广域网上跨接巨大的距离。

网络接口装置拥有对外部主机的物理接口，它的应用程序提供通信协议和 API，让基于主机的程序能访问 LonWorks 网络。网关装置使专用的传统控制系统能接到 LonWorks 系统。它拥有适合外系统装置或通信总线的物理接口。它的应用程序对接外系统的专用通信协议。网关根据需要在这两个协议间翻译，以便让报文在两个系统间传输。在某些情况下，网关能把专用的以指令为基础的外系统报文转换成以信息为基础的 LonWorks 网络使用的 SNVT。网关是一个外部对象，是对不相似系统的一个连系。即使选定的报文能在两个系统间传输，这个连系也远不是无缝的，它构成一个瓶颈并把分离的操作系统和网络工具引进到集成工作中。

LNS 网络操作系统提供在 LonWorks 网上支持互可操作应用的标准平台。LNS 让多个应用和用户同时管理网络并和其相互作用。这个特点使备有软件工具的多个安装者能同时在网上启动装置。

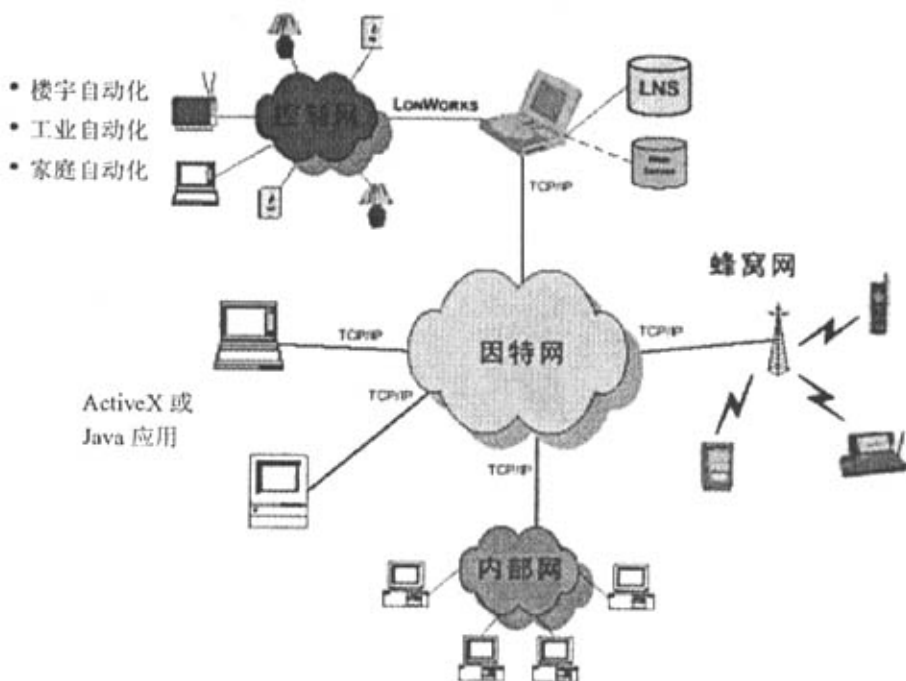
LNS 是用户/服务器体系结构，为新一代的互可操作的 LonWorks 网络工具设立

基础。它是一个为控制网服务提供标准平台的软件。LNS体系结构由两个主要部分组成,网络服务服务器NSS(network services, server)和网络服务接口 NSI(network services interface)。服务定义为一个能被客户调用、在服务器上执行的操作。网络服务服务器负责网络服务处理,维护网络数据库、协调对其服务和数据的多点访问,它同时维护所有网络 and 应用程序的服务提供者和事件发生器的目录。客户通过一个称为网络服务接口的硬件元件与服务器对话。网络服务接口提供到网设备的网络联接和到网络服务服务器的信息联接,提供对网络服务服务器和应用服务器的透明访问。

LNS 体系结构以对象集台的方式提供网络服务。每一个对象都带有方式(methods)、特性(properties)和事件(event)。方式用来调用对象上的服务,例如用以启动设备或进行网络变量链接;特性用以读写对象上的参数。

实现一个无缝地继承控制系统,需要系统中的所有组件都具有互操作性,并且能与其他相关系统传递信息。互操作性的实现,需要来自不同厂家之间以及网络管理和网络维护采用标准的通信方式,这需要在系统级采取一些方法,包括一个通用的通信协议、不同类型的通信收发器、路由器、对象模型以及管理和故障检测工具。

互可操作性带来以下许多好处:由于一个传感器或者控制设备能够被不同的子系统共享,所以需要使用的传感器和控制设备的数量比以前少了,从而可以减少整个系统的成本;系统间共享信息的功能使许多重要的应用成为可能;互操作性产品能使设备拥有者在不同的制造商间选择产品,而不会象以前那样被束缚在一家制造商的封闭技术上;互操作性也是因特网能够快速成长的促进因素之一:因特网通信的连接标准 TCP/IP 协议为连接分散在各地的政府和社会公共机构的计算机提供了一个通用的传输机制。超文本连接标识语言和 Web 浏览器的出现帮助创造了链接界面,使任何人只要有最基本的工具就能使用因特网,这大大促进了因特网的普及性和新应用的发展。^{[1][2][3][4][5]}

图 2.1 LNS 网络模型拓扑图^[9]

2.2.3 CAN 总线原理^{(34) (35)}

CAN 即控制局域网，主要用于各种过程检测及控制的一种网络，它是由德国 Bosch 公司设计的，目前逐步应用到工业控制中并成为 ISO 国际标准。

CAN 的技术有以下特点：

1. CAN 总线接口芯片支持 8 位、16 位 CPU，可做成 IA 与 PCI 总线的插卡任意插在 PC、XT、AT 兼容机上，也可置于物理量的变送器中，构成智能仪表；
2. CAN 可以是对等结构，采用多主从工作方式，网络上任意一个节点可以在任意时刻主动的向网络上其它节点发送信息而不分主从。这种通信方式很灵活，可以很方便的构成多机备份系统；
3. CAN 网络上的节点，可分成不同的优先级以满足不同的实时要求；
4. CAN 采用非破坏性总线仲裁技术，当两个节点同时向网络上传送信息时，优先级低的节点主动停止数据发送，而优先级高的节点可以不受影响的继续传输数据，有效避免了总线冲突；
5. CAN 可以点对点、点对多点以及广播式发送及接收数据。通信距离最远为

10km, 通信速率最高可达到 1mb/s, 节点数可达 110 个;

6. CAN 采用短帧结构, 每一帧的有效字节数为 8 个。这样短的帧传输时间短, 受干扰概率小, 重发时间短。每帧信息都有 CRC 校验以及其它纠错措施, 保证了数据的正确率;

7. CAN 节点在错误严重的情况下, 还具有自动关闭总线的功能, 切断自己与总线的联系, 避免影响总线上的其它节点操作;

8. 通信介质采用性价比高的双绞线, 无特殊要求。开放系统成本低, 用户接口简单、编程方便, 容易构成用户系统。^{[34][36]}

2.2.4 改进的系统总线结构^{[23][24]}

出于对成本、性能等多方面因素的平衡考虑, 系统联网方式改进为 3 级总线方式, 即分级使用 RS-485, CAN 总线和 LonWorks 总线模式。将这三种总线技术有效结合在一起。

1. RS485 通讯: RS485 通讯采用主从方式, 每个模块通讯仅需一个收发器, 在小模块上使用完全符合系统要求, 将其用于安装于每户家庭中的智能模块上, 可以大大节省成本。
2. CAN 通讯: 在单元级采用 CAN 总线, 实际上本系统的该通讯方式仅在物理层采用 CAN 收发器, 在数据链路层并未按照 CAN 标准。系统特别设计了一套数据链路层协议, 固化到一块单片机作为通信控制器, 通过主 MCU 控制实现通信功能。其优点为: 采用对等通信, 在处于最大通信负载(需发送的数据包)情况下, 吞吐量(传输的有效数据包总量)不会降低。单元级通讯的原理: 在数据链路层之上设计了一个数据映像层, 使网关内存与各个数字智控模块的内存一一映像, 通讯的目的是达到两者之间数据的同步, 网关根据内存的数据独立决定操作方式。^[23]
3. LonWorks 通讯: 小区级采用 LonWorks 总线方式, 主要有以下优点:
 - 1) 在远距离传输方面 LonWorks 有完善成熟的解决方案。
 - 2) 可实现与前一系列产品的兼容。

3) LonWorks 有灵活的传输组网方式和程序在线下载功能,遇到特殊情况时修改余地更大。

4) 本系统在一个单元或一幢楼采用一个控制网关(即一个 LonWorks 节点),一个 LonWorks 通信节点的成本很高,但摊到每户成本就低了。节点数的减少,不仅降低成本,也减少了组网的复杂度。^[24]

三级总线分层传输机制是对标准 LonWorks 组网系统的优化和改进。LonWorks 系统的标准设计理念是:检测机构(输入)和执行机构(输出)分别由不同的节点控制,节点与节点之间通过 LonWorks 总线相连,数据传输方式以及检测机构和执行机构之间的逻辑关系可以由用户根据现场情况灵活决定。

LonWorks 系统是一种理想的全分布式控制网络。但是, LonWorks 应用于智能小区遇到许多特殊问题:

1. 一个完整的智能小区,需要的控制节点数目很多,大量的控制节点不仅使整体费用偏高,而且在设计的过程中,必须考虑主干通道的带宽问题,如果不采用特殊的流量控制机制,普通的网络变量通信方式在一定条件下会引起网络阻塞。

2. 一个很大的 LonWorks 网络,不仅工程现场组网需要花费大量的时间,而且维护困难。

3. 对于智能小区系统内部的数据,其数据流向都是很规整的:如用户的报警信息,都是传到同一管理中心;用户的抄表数据也是传到物业管理中心。如果每户都需要组网,那么现场维护的很多时间实际上是在重复工作。

在本系统中,控制网关模块实际上是对其所属的家庭消防逃生模块、单元控制器、区域控制器的 LonWorks 映像。对于控制中心而言,它认为每一户都是一个 LonWorks 节点,其操作方式与标准 LonWorks 系统(如原有 B 型系统)完全一致。但是在实际硬件联接中,一幢或一个单元只有一个 LonWorks 节点,而且在设计时考虑了流量控制机制,这样在现场应用时不仅大大减少维护工作量,而且

也增加了系统的可靠性。

3 硬件设计

3.1 硬件系统简介

本项目研制一种基于 LonWorks 控制网络的智能消防应急与逃生联网系统，系统由智能逃生砖、智能应急灯、智能控制网三部分构成，其中智能控制网又包括单元控制器、区域控制器和远程监控主机。

1. 智能逃生砖：包含电子报警与控制接口电路、警铃、逃生软梯等几个部分，智能逃生砖的外型与建筑外墙协调，统一嵌装在楼房各窗户外侧墙面，垂直成为一列，构成分布式智能逃生网络系统。当打开智能逃生砖时，逃生软梯被自动打开，同时系统发出声光报警信号并向管理中心通报该逃生设备已被启用，困在家中的人可以沿着逃生软梯逐阶而下，直到下一层的智能逃生砖被打开继续下行，最终安全到达地面为止。智能逃生砖还可以外接火灾探测器、紧急求救按钮的报警接口，实现自动火灾报警与紧急求救等系统功能。
2. 智能应急灯：由电子电路与应急灯组成，它被安装在楼房房间或通道内，系统可以通过控制网络自动定时检测应急灯故障情况。在管理中心接到紧急求救或火灾报警信号后，联动进行网络应急灯控制。当消防通道与消防设备的方向性符号被印制在应急灯处时，系统就具有了火灾应急指示功能，可以满足建筑设计防火规范的要求。
3. 单元控制器：含有智能电源模块与智能网关节点，可作为一个标准的 LonWorks 网络节点。单元控制器可为智能逃生砖与智能应急灯系统提供不间断电源以及多达 24 路设备接入与控制通道。
4. 区域控制器：由电源电路、节点电路、控制电路以及显示界面等部分构成，为标准的 LonWorks 节点，遵循 LonWorks 控制网络的各项技术要求，与单元控制器之间采用 LONTALK 协议通讯，每 64 个节点要增加一台 LonWorks 路由器。
5. 配置网络节点：内部安装 PCLTA-LON 网卡以及系统配置、调试与监控软

件，能够在需要时对整个系统进行网络变量捆绑、系统调试、系统组网以及系统通讯与工作状态控制与监测。

6. 远程监控主机：远程监控主机通过 Internet 宽带网络与设置在区域的 IP 服务器进行远程通讯，实现在大社区对异地安装的智能逃生系统进行系统设置、状态查询、故障诊断、远程修复及软件升级换代功能。同时也可以实现在杭州本部对全国各地的用户设备进行远程售后服务。

本项目产品在平时可以自动对消防应急与逃生设备进行远程检查，如果出现异常打开或损坏状况能够马上现场报警并及时向管理中心报警，解决了消防应急与逃生设备的非正常使用或损坏问题。在有火警发生时又能给人们提供快速有效的报警、应急指示与紧急逃生通道。本项目具有自主知识产权，是现代高科技网络技术与消防应急逃生设备的理想结合模式，可广泛应用于各种民用建筑群与商用建筑与建筑群系统。

3.2 上位机通信设计

1. 主 CPU 内部 SRAM 分配

表 3.1 主 CPU 内部 SRAM 配置表

内容	起始地址	终止地址
模块 1 配置	100H	106H
模块 2 配置	108	10FH
...
模块 20 配置	198H	19FH
定时器 1	1A0H	1A2H
定时器 2	1A4H	1A6H
定时器 3	1A8H	1AAH
定时器 4	1ACH	1AEH
时钟	1B0H	1B4H
系统设置状态	1B5H	1B6H
振铃次数	1B7H	1B7H
密码(4 位)	1B8H	1BBH

楼号, 房号(new)	1BCH	1BDH
电话号码 1	1C0H	1CFH
电话号码 2	1D0H	1DFH
电话号码 3	1E0H	1EFH
电话号码 4	1F0H	1FFH
...
模块 1-8 向中心报告区	(MFLAG1) 0E0H	
模块 9-16 向中心报告区	(MFLAG2) 0E1H	
模块 17-20 向中心报告区	(MFLAG3) 0E2H	
KFLAG	0E3H	

DFLAG (107H) 定义:

表 3.2 DFLAG 位信息列表

位	BIT0	BIT1	BIT2	BIT3	BIT4
定义	0B0H-修改	0C0H-修改	0D0H-修改	0E0H-修改	0F0H-修改

MFLAG3 (0E2H) 定义:

表 3.3 MFLAG 位信息列表

位	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
定义	模块 17	模块 18	模块 19	模块 20	1: 防拆告警			

KFLAG (0E3H) 定义:

表 3.4 KFLAG 位信息列表

位	BIT0	BIT1	BIT2	BIT3
定义	1: 开锁	1: 请求服务	发类型登录	1: 请求中心通话

位	BIT4	BIT5	BIT6	BIT7
定义	1: 发设防状态和密码	对讲呼叫应答 (正常)	对讲呼叫应答 (忙)	1:

0E4H, 0E5H: 00, 00——开锁 其余——服务

系统设置 (1B5H)

表 3.5 系统设置位信息列表

Bit7	Bit6	Bit5	Bit4
设防	撤防	模式 bit2	模式 bit1
Bit3	Bit2	Bit1	Bit0
模式 bit0		0: 拨号 CPU 正常	0: 有模式切换操作

系统状态 (1B6H) (0——告警, 1——正常)

表 3.6 系统状态位信息列表

Bit7	Bit6	Bit5	Bit4
0: 防拆告警		电话断线告警	电话断线已告警
Bit3	Bit2	Bit1	Bit0
电话告警允许	1: 双口 RAM 正常	0: 市电停	0: 设防提示

2. 主 CPU 485 总线设备存储协议

长度: 7 字节

格式:

字节一:

表 3.7 字节一位信息列表

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
设备类型				物理地址			

字节二~字节四:

其余设备设置量

字节五~字节七:

主 CPU 暂存变量

设备类型定义:

- 0001——键盘模块 (无联动)
- 0010——DDC 智控模块 (有联动)
- 0011——DDC 报警模块 (有联动)
- 0100——DDC 三表模块 (有联动)
- 0101——A/V 模块 (有联动)
- 0110——门铃模块
- 0111——红外模块
- 1000——拨号模块
- 1001——门口机模块
- 1010——智能逃生砖 (有联动)
- 1011——智能应急灯 (有联动)
- 其余——空

智能逃生设备详细列表:

1) 智能逃生砖

字节一:

表 3.8 智能逃生砖字节一位信息列表

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
设备类型 (1010)				物理地址			

字节二:

表 3.9 智能逃生砖字节二位信息列表

Bit7——Bit4	Bit3——Bit0
联动源	

注: Bit7 =0 时联动源屏蔽

字节三:

表 3.10 智能逃生砖字节三位信息列表

Bit7——Bit0
联动允许 7——0

字节四:

表 3.11 智能逃生砖字节四位信息列表

Bit7——Bit0
联动对应 DO 状态 7——0

字节五:

表 3.12 智能逃生砖字节五位信息列表

Bit7	Bit6	Bit5	Bit4
D0 给定状态	DI 反馈状态	输入状态	清按键标志
Bit3	Bit2	Bit1	bit0
已拨号告警	休眠	防拆报警	通信失效

字节六:

表 3.13 智能逃生砖字节六位信息列表

Bit7	Bit6	Bit5	Bit4
Bit3	Bit2——bit0		
	设置后稳定次数		

表 3-14 探测器类型列表

序号	探测器类型	24 小时开	启动延时	告警延时	LED 锁定
0	紧急	√	X	X	√
1	红外 1	X	√	√	√
2	红外 2	X	X	X	√
3	红外 3	X	√	X	√
4	红外 4	X	X	√	√
5	门磁 1	X	√	√	√
6	门磁 2	X	X	X	√

7	煤气	√	X	X	√
8	烟雾	√	X	X	√
9	火警	X	X	X	
10	窗磁	X	X	X	

2) 智能应急灯

字节一:

表 3.15 智能应急灯字节一位信息列表

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
设备类型 (1011)				物理地址			

字节二:

表 3.16 智能应急灯字节二位信息列表

Bit7——Bit4	Bit3——Bit0
无	类型

字节三:

表 3.17 智能应急灯字节三位信息列表

Bit7——Bit0
无

字节四:

表 3.18 智能应急灯字节四位信息列表

Bit7——Bit0 (LSB)
无

字节五:

表 3.19 智能应急灯字节五位信息列表

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
						防拆报警	1: 通信失效

字节六:

表 3.20 智能应急灯字节六位信息列表

Bit7	Bit6	Bit5	Bit4
------	------	------	------

1: 要求送数据			
----------	--	--	--

3. 系统设置 (1B5H)

表 3.21 系统设置位信息列表

Bit7	Bit6	Bit5	Bit4
设防	撤防	模式 bit2	模式 bit1
Bit3	Bit2	Bit1	Bit0
模式 bit0		0: 拨号 CPU 正常	0: 有模式切换操作

模块类 1:智能逃生软梯

字节一:

表 3.22 智能逃生软梯系统设置字节一位信息列表

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
设备类型 (1010)							

字节二(五):

表 3.23 智能逃生软梯系统设置字节二位信息列表

Bit7	Bit6	Bit5	Bit4
	D1 反馈状态	输入状态	
Bit3	Bit2	Bit1	bit0
		防拆报警	

表 3.14 探测器类型列表

序号	探头类型	24 小时开	启动延时	告警延时	LED 锁定
0	紧急	√	X	X	√
1	红外 1	X	√	√	√
2	红外 2	X	X	X	√
3	红外 3	X	√	X	√
4	红外 4	X	X	√	√
5	门磁 1	X	√	√	√
6	门磁 2	X	X	X	√

7	煤气	√	X	X	√
8	烟雾	√	X	X	√
9	火警	X	X	X	
10	窗磁	X	X	X	

模块类 2: 智能应急灯

字节一:

表 3.23 智能应急灯系统设置字节一位信息列表

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
设备类型 (1011)				物理地址			

字节五:

表 3.24 智能应急灯系统设置字节二位信息列表

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
						防拆报警	1: 通信失效

4 软件设计

本系统软件部分由消防应急与逃生联网控制软件和远程监控子系统控制软件两部分组成:

1. 消防应急与逃生联网控制程序

通过三级总线组网方式实现对智能逃生砖、智能应急灯、单元控制器、区域控制器、远程监控主机的联网, 内部安装 PCLTA-LON 网卡以及系统配置、调试与监控软件, 能够在需要时对整个系统进行网络变量捆绑、系统组网、系统调试以及系统通讯与工作状态检测与控制。

2. 远程监控模块

通过 Internet 宽带网络与设置在区域的 IP 服务器进行远程通讯, 实现在小区控制中心对各楼宇安装的智能逃生模块进行参数设置、状态查询、故障诊断、远程修复及软件下载升级等功能。同时也可以实现杭州本部对全国各地用户设备进行远程售后技术支持。

4.1 消防应急与逃生联网控制程序设计

4.1.1 设计思想

消防应急与逃生联网系统网络结构采用三层总线组网模式，即上层的 LonWorks 总线、单元级的 CAN 总线和底层设备的 RS-485 通讯。整个系统的网络拓扑图如下图所示：

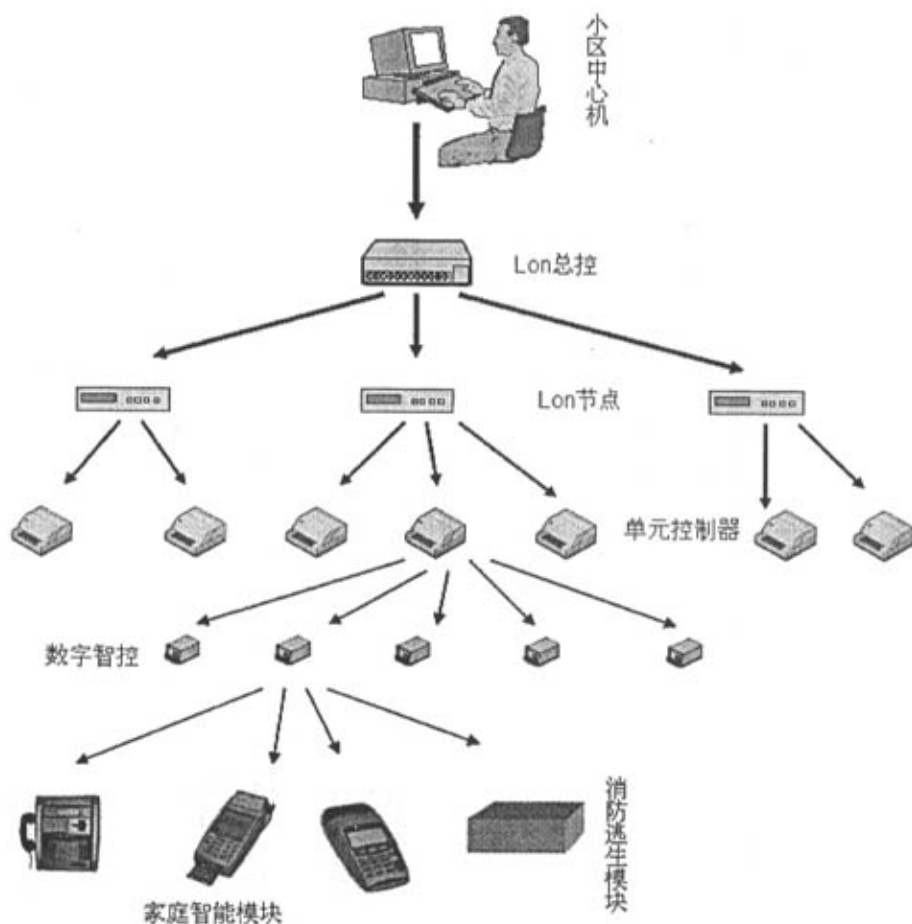


图 4.1 消防逃生系统结构图

从软件设计方面，系统也分为三层，上层为安装在小区中心控制机上的控制软件、中间层为智能控制网、底层为智能模块层。系统首先将各个模块按照各层的通信协议，把模块信息写入控制软件的数据库，当需要执行查询、设置等命令时，只需读数据库来找出*栋*单元**室**模块，即可对其进行相应的操作，同时

用操作后修改的书架更新备份数据库。

4.1.2 功能设计

消防应急与逃生联网控制软件主要设计为以下功能：

1. 用户管理

给每个管理员分配账号、密码，并保存在数据库中。当管理员登录时，查询数据库中的预存密码进行身份校验。同时给每个管理员发放含管理员信息的 IC 卡，用作进行设备维护时的身份鉴别卡。

程序流程图如下图所示：

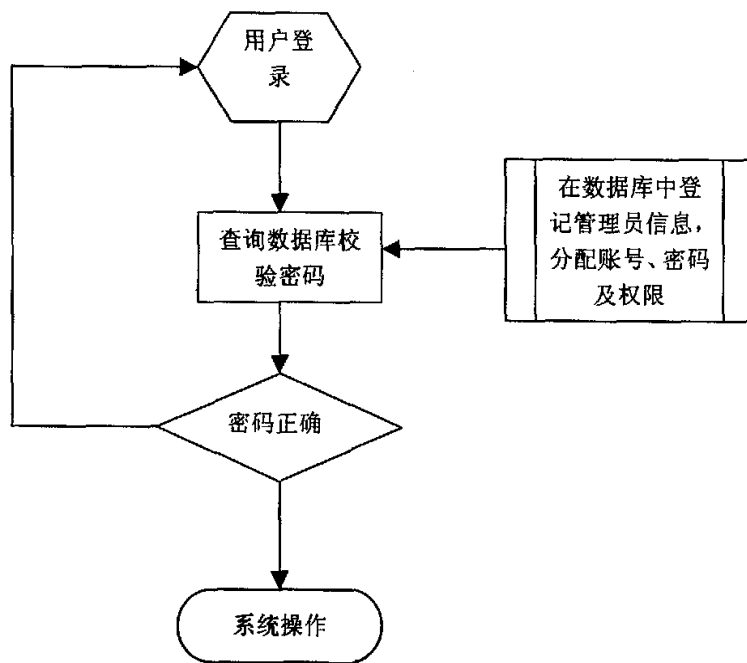


图 4.2 用户管理流程图

2. 系统管理

系统管理是针对各级网络设备的查询及设置，主要功能有：

- 串口表分配：在数据库中写入所用各串口的名称、对应的苑号、并为其分配网络地址初始值及网络地址终止值。
- 通讯口分配：读取串口列表，设置主通讯口和制卡串口。

- 住户登记：登记住户的户名、详细住址（苑名、楼号、单元号、层号、房号等）、物理地址、联系方式、户主照片等信息，并将其写入数据库中保存，以供今后查对。
- 电子定位：输入节点标识即可在小区地形图中定位节点的详细地址，便于故障发生时，工作人员可及时上门维修。
- 服务类型设置：将服务类型如消防逃生砖维护、应急灯维修、检测等服务类型加以编号写入数据库，以便生成维修记录报表。
- 节点测试：输入网络节点地址，对节点进行功能及状态测试，以便及时发现并排除故障。
- 读取设备物理地址：输入网关所在的苑号、网络地址即可读取其相应的物理地址。
- 权限设置：设置、修改用户的操作权限。具体权限有：报警查询、警情历史记录查询、IC卡维护、住户短信发送、大屏显示、电梯控制、应急灯控制、逃生砖控制、设备维护记录查询、服务请求查询、门禁查询、住户登记、串口表配置、通讯口配置、电子定位、修改密码、修改权限、更新数据库等。
- 校正时间：定期校正智能控制网络中各设备的系统时钟，并在控制软件用户界面中显示。
- 数据库备份：在硬盘、磁盘等存储容器中对当前数据库进行备份，以便数据库丢失时及时进行恢复。
- 数据库整理：定期对数据库进行整理、维护，保证其正常运行。

程序流程图如下图所示：

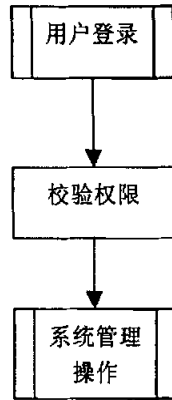


图 4.3 系统管理流程图

3. 设备管理

包括设备参数初始化、当前状态查询、故障报警、设备参数修改、读取物理地址、数据库备份及整理等。

其程序流程图如下图所示：

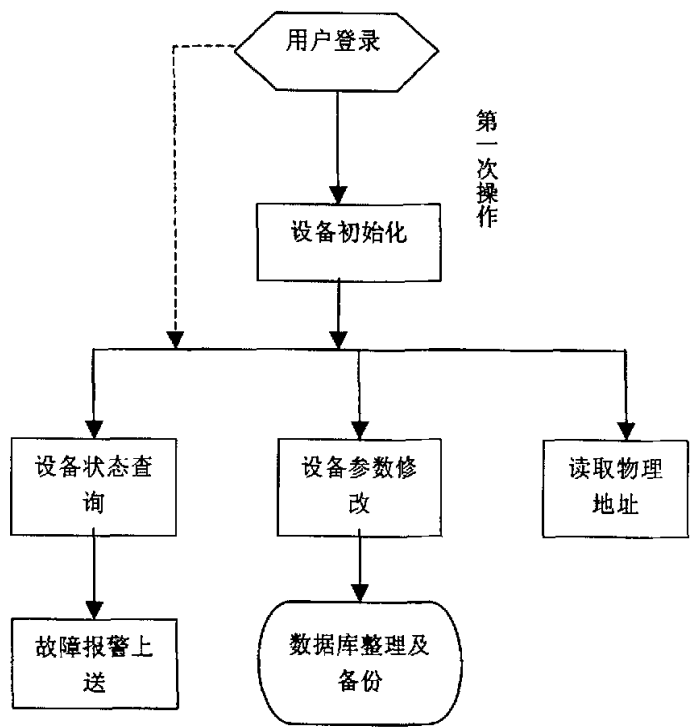


图 4.4 设备管理流程图

4. 短信发布

定期将设备查询的明细用短信方式发送给管理维护人员，同时支持大屏滚动播放。

其程序流程图如下图所示：

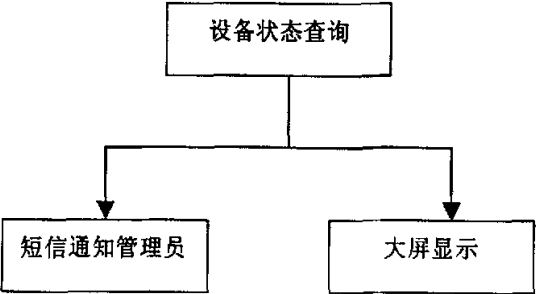


图 4.5 短信发布流程图

5. 报警管理

当有火警发生时，根据探测器判定并上送的火警地点，指挥打开逃生梯并逐层放下逃生梯及打开应急灯，同时将火警信息用短信及广播的方式传达给相关住户和管理员以便及时报警。今后可在此系统的基础上扩展与消防部门网络直接联网报警。

其程序流程图如下图所示：

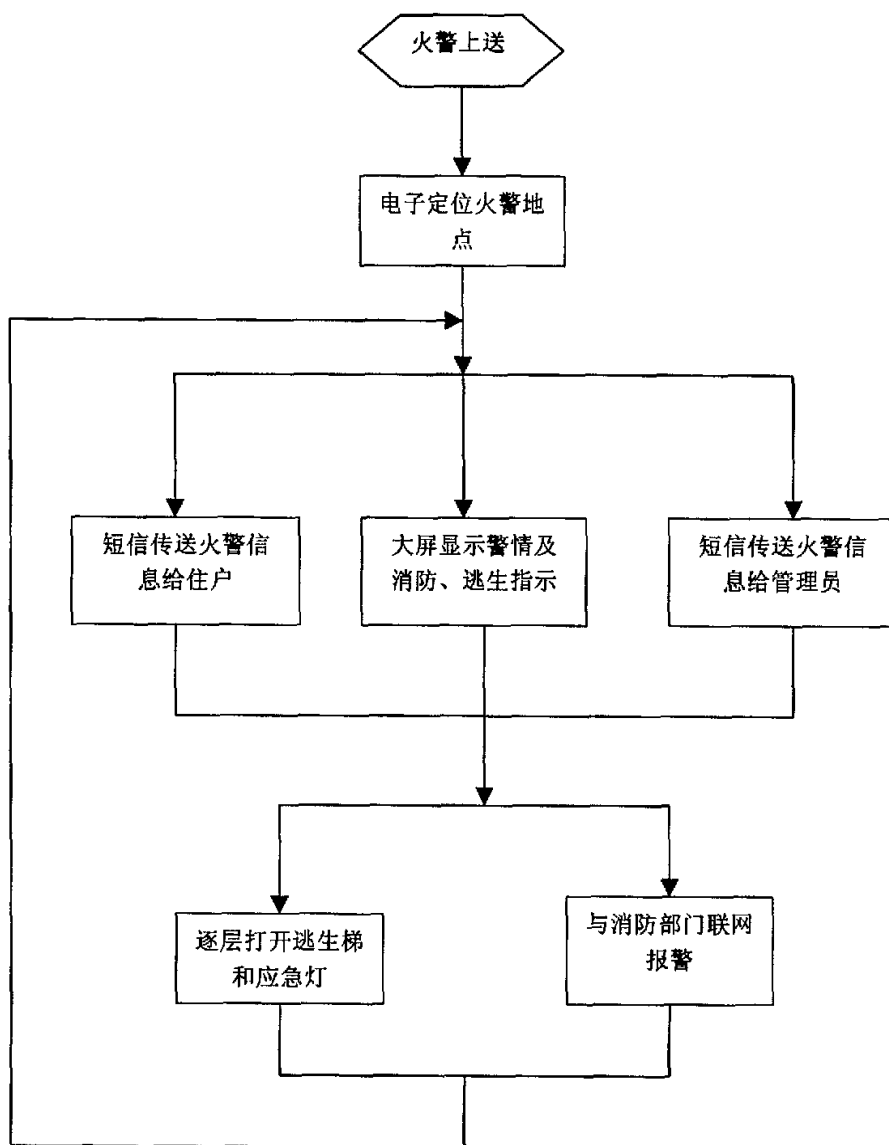


图 4.6 报警管理流程图

6. 报表查询

可对已发送短信、警情信息、故障信息、历史记录等进行报表呈现及查询，同时支持报表的 Excel 生成与保存。

4.2 监控程序设计

4.2.1 设计思想

远程监控模块是在原有的智能小区控制系统硬件网络层的基础上,采用 TCP/IP 协议和 UDP 协议,通过网络传输数据指令,实现对消防报警智能模块状态的远程查询和设置。它的作用分为两部分:

1. 小区管理员对每栋楼内安装的消防报警模块进行增加、删除、查询、设置、恢复。并将每次修改的数据保存到服务器数据库中。
2. 允许多个住户同时使用监控程序对所在楼层消防报警模块进行查询和设置。并将更新的数据保存到服务器数据库中。

远程监控模块的实现大大提高了模块操作的方便性,减少了现场维护的工作量。

远程监控模块设计为三层网络结构:

1. 模块层

模块层分为硬件软件两部分,硬件即为安装在每栋楼内的各个模块,包括智能逃生砖、智能应急灯等模块。软件部分即模块控制程序,负责向模块发送操作指令。

2. 服务器端

服务器的功能是接收客户端的指令数据包,然后将其发送到模块控制程序,并把模块响应的数据包回送给客户端;

3. 多客户端

多客户端分管理员客户端和住户客户端。允许多个客户同时访问网络,对网络上的模块进行设置和查询。

远程监控模块的网络拓扑图如下:

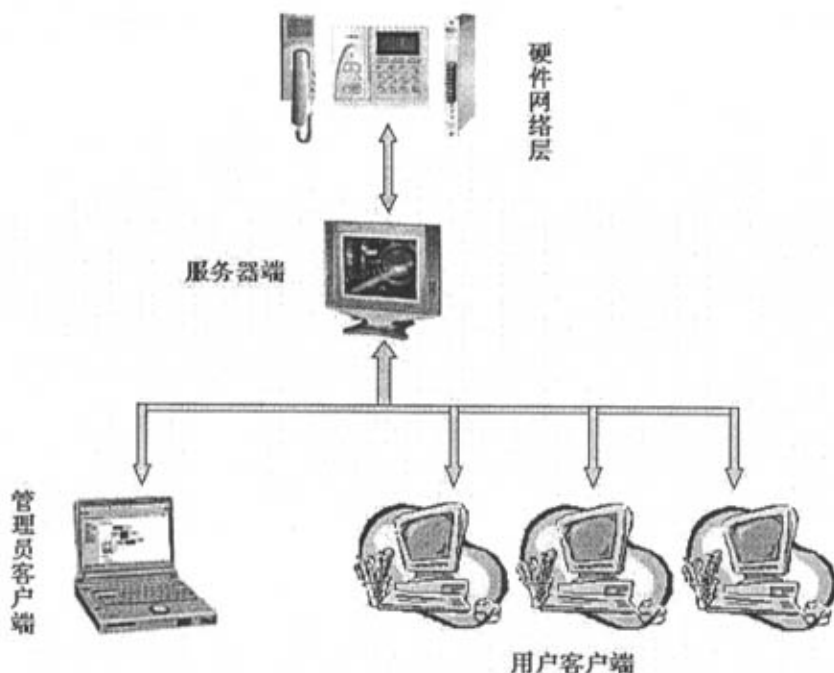


图 4.7 远程监控系统结构图

4.2.2 实现方式

1. 服务器端程序设计

服务器程序主要实现数据的交互，区分两种协议，采用多线程方式监听客户端的数据请求。此程序为原有 21CN 远程监控服务器程序的扩展。

2. 客户端程序设计

为了便于程序的维护和升级，这部分程序采用分层设计。

第一层为模块层，采用 MFC ActiveX 控件的方式，将要显示的模块按其类型和硬件协议打包成八个独立控件，每个控件留出接口用于输出和输入模块信息、显示和改变模块状态，支持动态调整模块的显示位置。

第二层为通讯层，继承 MFC 的 CSocket 类，根据上述通讯协议对接收和发送的函数进行重载，实现与服务器之间的通讯。

第三层为应用层，采用 SDI 分隔视图方式，调用多个无模式对话框分别实现登陆服务器、提交住户信息、查询模块状态、设置模块状态、恢复模块状态、修改密码等功能，并在后台执行数据库修改。

程序主流程图如下图所示：

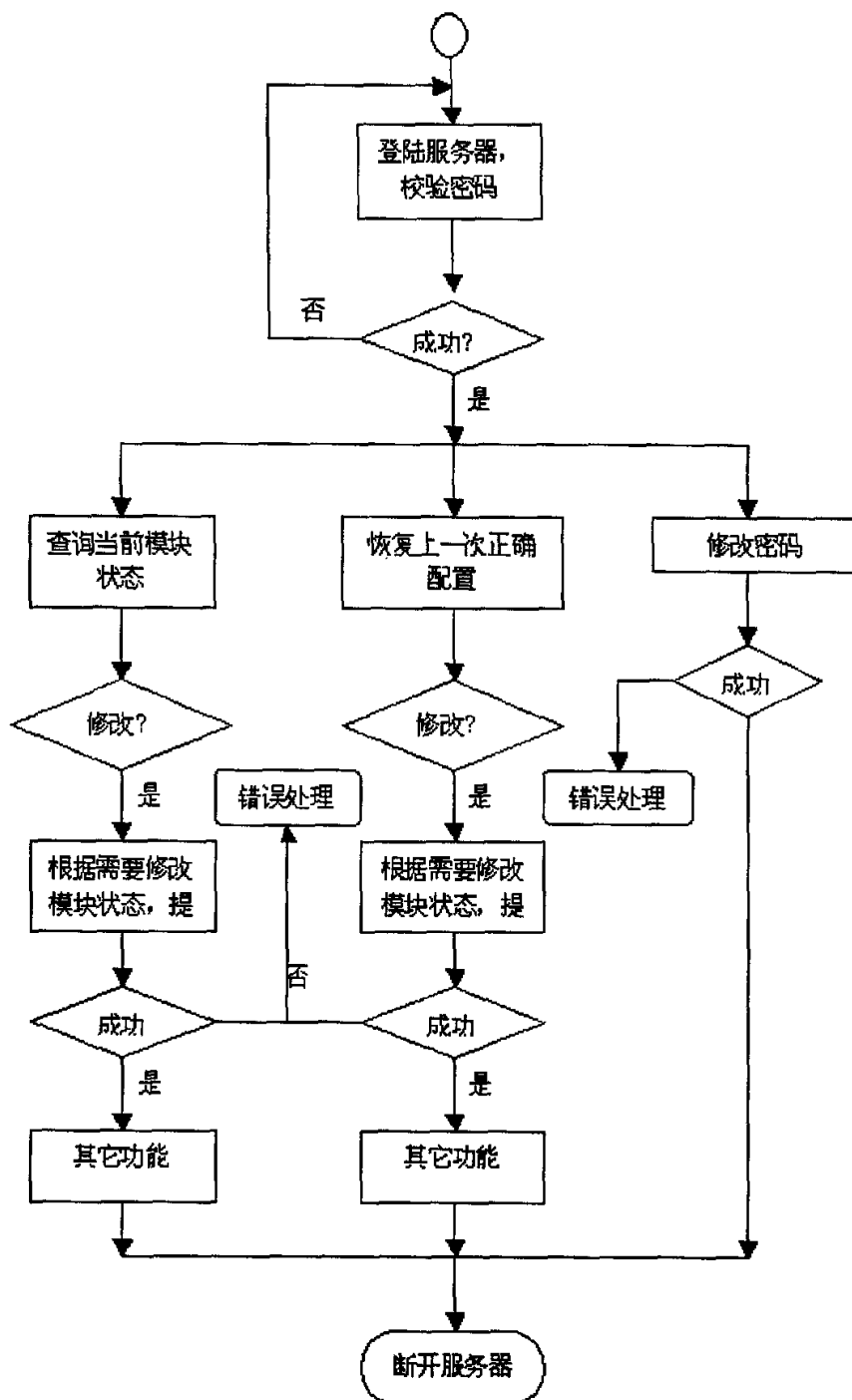


图 4.8 远程监控系统主流程图

4.3 系统通信结构与通信协议

4.3.1 总线结构与通信协议

1. LON 总线结构

系统的上层设备通信采用 LON 总线方式，其中上层设备包括远程监控主机、总控模块、控制网关。远程监控主机可以直接连到 LON 网络上，作为一个 LON 节点，通过 LON 总控和 LON 网络上的其他节点通信，这种连接方法要使用 LON 网络的接口，会提高产品成本，并且网络接口的使用有时间限制，到期要重新注册。为了削减成本，可先采用 RS-232 协议用串口和 LON 网络通信，再用一个 LON 转换节点把串口的 RS-232 协议转换成 LonTalk 协议，然后把远程监控主机的监控命令转发给其他的网络节点。同样，发送到远程监控主机的 LON 网络的消息，也要通过转换节点转换成 RS-232 协议包，发送给控制中心机。其总线结构图如下所示：

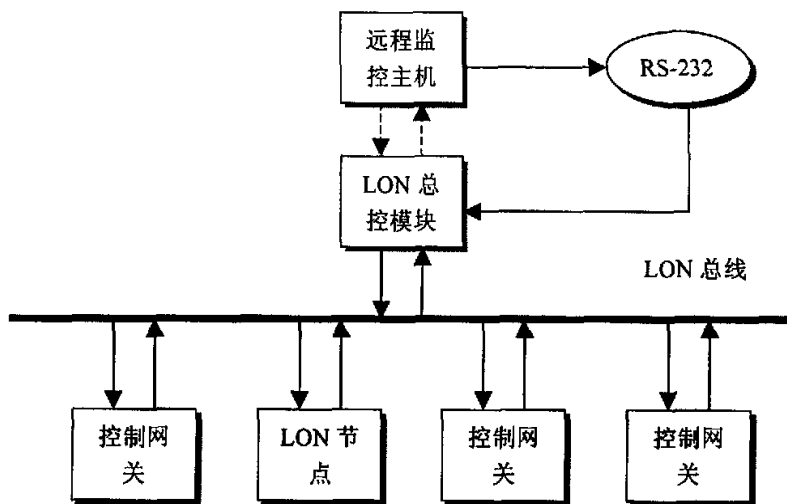


图 4.9 LON 总线组网拓扑图

其通信协议遵循 LonTalk 协议规范。

2. CAN 总线结构

在单元级采用 CAN 总线，使网关内存与各个数字智控模块的内存一一映像，达到两者之间数据的同步，网关根据内存的数据独立决定操作方式。其

总线结构图如下所示：

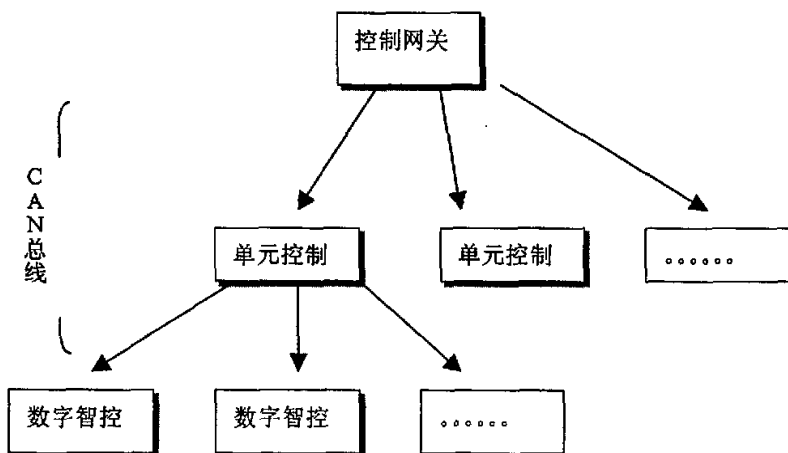


图 4.10 单元级 CAN 总线组网拓扑图

3. 485 总线设备通信协议

消防逃生模块与数字智控之间采用 RS485 通讯，通讯采用主从方式，每个模块通讯仅需一个收发器，用多串口卡将其连接在一起，其总线结构如下图所示：

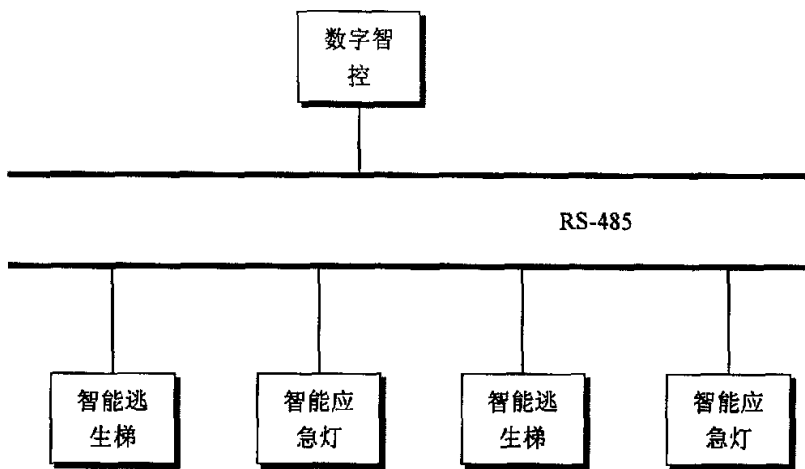


图 4.11 模块级串口拓扑图

通信协议格式如下：

命令帧

表 4.1 RS-485 协议命令帧格式

byte1		byte2		N*byte	
Bit7:5	bit4:0	Bit7:4	Bit3:0		
设备类型	地址 (saddr)	命令(scom)	长度(slen)	数据	校验(CRC)

应答帧

表 4.2 RS-485 协议应答帧格式

byte1		byte2		N*byte	
Bit7:5	bit4:0	Bit7:4	Bit3:0		
设备类型	地址 (saddr)	应答(ackn)	长度(slen)	数据	校验(CRC)

注： 485 总线子模块地址仅为模块标志，与模块功能无关。

1) 智能逃生砖模块

主机命令：

表 4.3 逃生砖主机命令格式

Byte1(7:4)	byte1(3:0)	byte2(7:5)	byte2(4:0)	data	crc
1010	地址	command	1		

从机应答：

表 4.4 逃生砖从机命令格式

Byte1(7:4)	byte1(3:0)	byte2(7:5)	byte(4:0)	data	crc
1010	地址	ackn	1		

读 DDC 模块状态

主机命令：

表 4.5 逃生砖 DDC 模块主机命令格式

命令码	数据长度	数据(data)	说明
111	1	bit3:0—模块类型 bit4=0—清按键操作 bit7=0—D0	读每个子模块告警状态

从机应答：

表 4.6 逃生砖 DDC 模块从机命令格式

应答 bit7	应答 bit6	应答 bit5	数据长度
CF: 0 防拆报警	0—有按键	DI: 告警状态	0

表 4.7 DI、CF 组合含义

DI	0	0	1	1
CF	0	1	0	1
意义	防拆告警	探头告警	防拆屏蔽	正常

2) 智能应急灯模块

主机命令:

表 4.8 应急灯主机命令格式

byte1(7:5)	byte1(4:0)	byte2(7:5)	byte2(4:0)	data	crc
1011	地址	command	1	X	

从机应答:

表 4.9 应急灯从机命令格式

byte1(7:5)	byte1(4:0)	byte2(7:5)	byte(4:0)	data	crc
1011	地址	ackn	0		

读 DDC 模块状态

主机命令:

表 4.10 应急灯 DDC 模块主机命令格式

命令码	数据长度	数据(data)	说明
111	1	bit3:0-模块类型 bit4=0-清按键操作 bit7=0-D0	读每个子模块告警状态

从机应答:

表 4.11 应急灯 DDC 模块从机命令格式

应答 bit7	应答 bit6	应答 bit5	数据长度
0: Dout	0—有按键	DI	0

4.3.2 Internet 远程通信

1. 服务器与客户端之间为 TCP/IP 协议。数据交互方式如下

(1) 服务器与管理员通讯

查询命令网络协议:

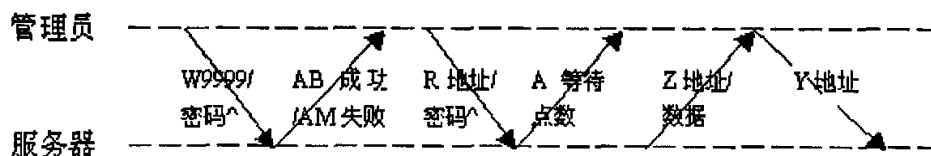


图 4.12 管理员查询命令通讯协议

设置命令网络协议:

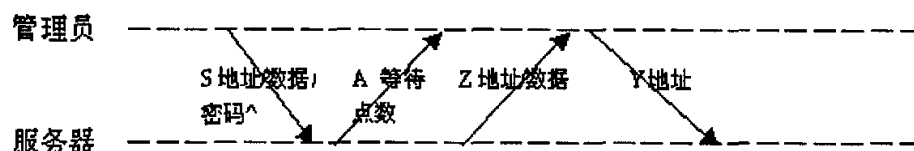


图 4.13 管理员设置命令通讯协议

(2) 服务器与住户通讯

查询命令网络协议:

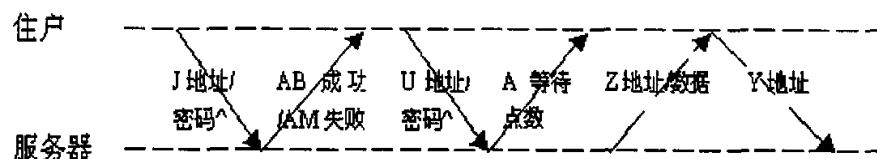


图 4.14 住户查询命令通讯协议

设置命令网络协议:

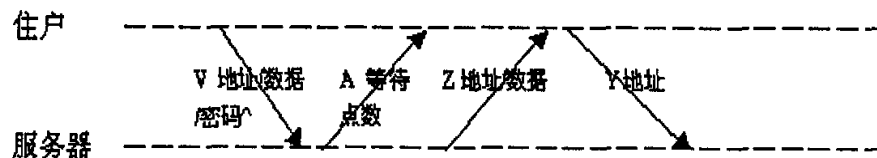


图 4.15 住户设置命令通讯协议

2. 服务器与模块控制程序之间为 UDP 协议 (在此不作详细介绍)

5 系统研究结果分析

5.1 消防逃生程序软件演示

1. 用户管理

1) 用户登录



图 5.1 用户登陆界面

2) 保安注册登记



图 5.2 保安注册界面

2. 系统管理功能演示:

1) 串口表配置



图 5.3 串口表配置界面

2) 通讯口分配

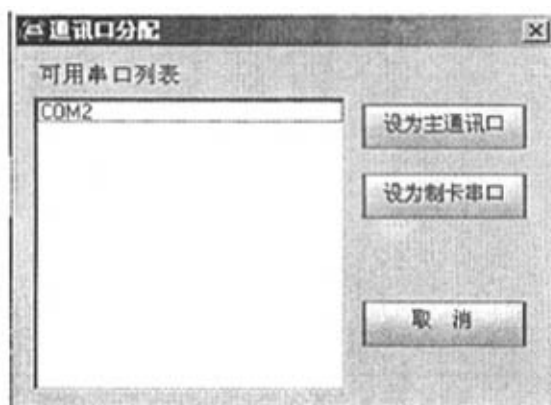


图 5.4 通讯口配置界面

3) 住户信息登记

户主名: 联系方式1:
 姓名: 联系方式2:
 楼号: 联系方式3:
 单元号: 在线标识: ☒ 户主照片:
 楼层号: 双击获取照片
 房间号:
 物理地址:

户主地址	物理地址
1栋 1单元 0-0	80
1栋 1单元 1-1	81

新增 保存 编辑 删除 查询 信息分类:

家庭成员信息

姓名	户主关系	联系方式1	联系方式2

新增 保存 编辑 删除 关闭窗体

图 5.5 住户信息登记界面

4) IC 卡管理

住户地址: 查询:
 身份:
 姓名:
 办理类型:
 办理日期:
 卡号:
 注册状况:

工具栏

注册卡号
 注销卡号
 清除卡号

新增 删除

图 5.6 IC 卡管理界面

5) IC 卡维护

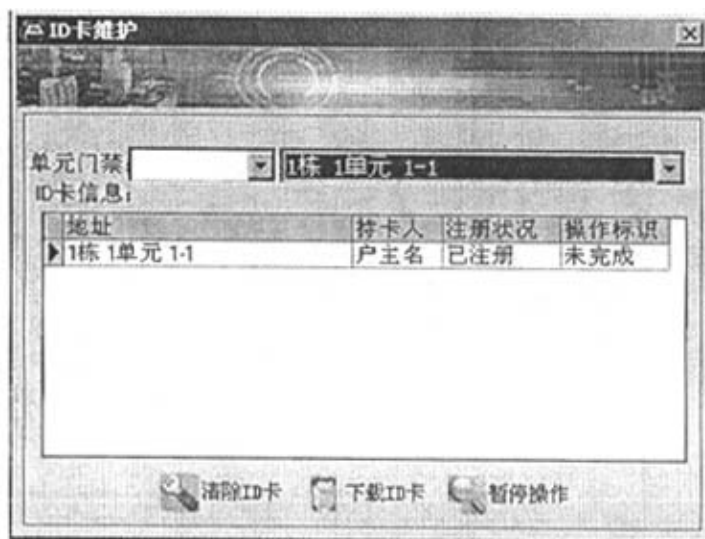


图 5.7 IC 卡维护界面

6) 电子定位

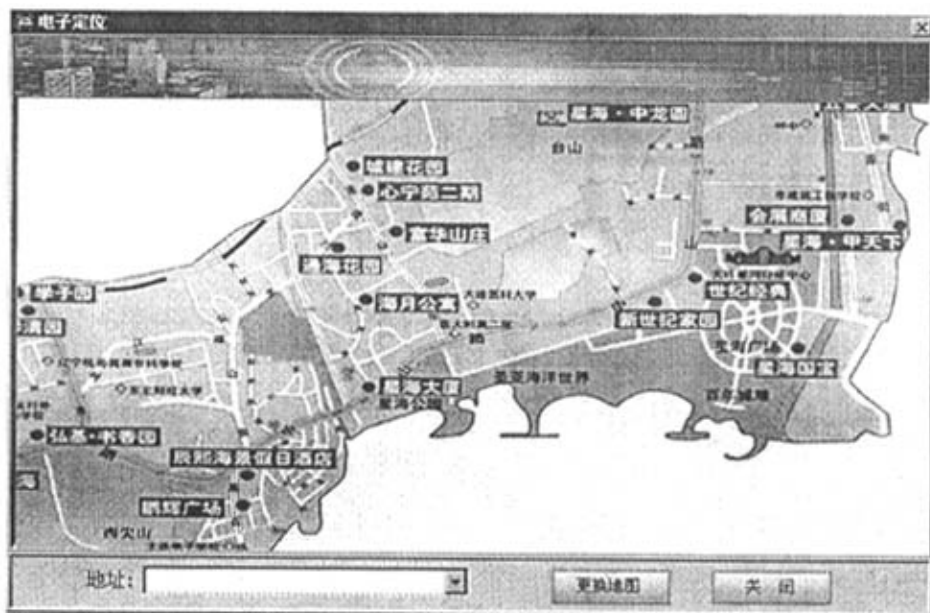


图 5.8 电子定位界面

7) 服务类型设置

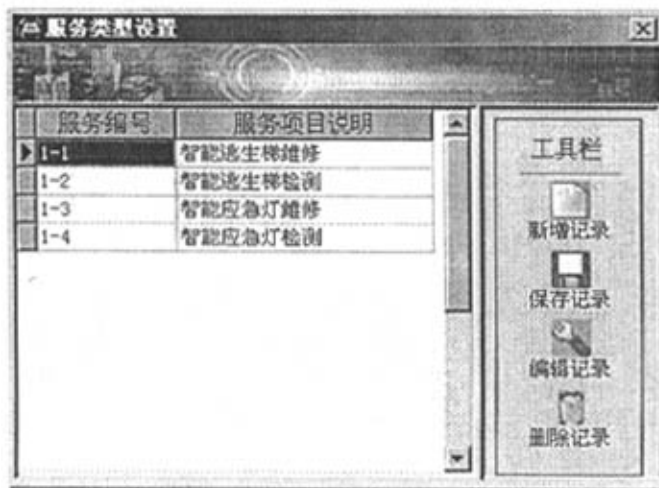


图 5.9 服务类型设置界面

8) 用户权限设置



图 5.10 用户权限设置界面

9) 节点测试



图 5.11 节点测试界面

10) 校正时间



图 5.12 校正时间界面

11) 读取物理地址

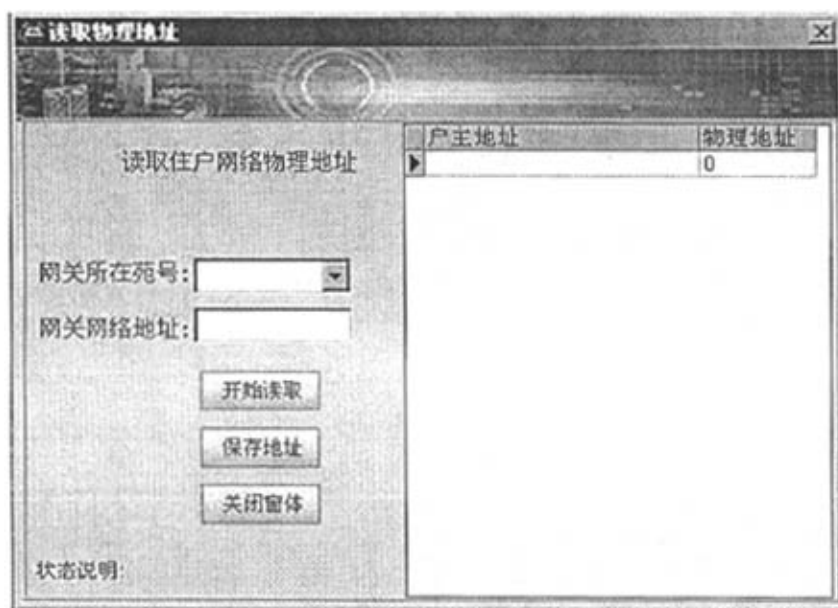


图 5.13 读取物理地址界面

3. 设备管理

1) 应急灯控制

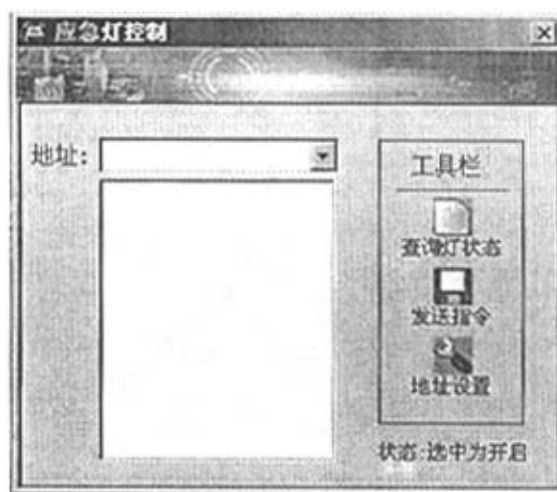


图 5.14 应急灯控制界面

2) 逃生梯控制



图 5.15 逃生梯控制界面

4. 短信管理

1) 短信发送

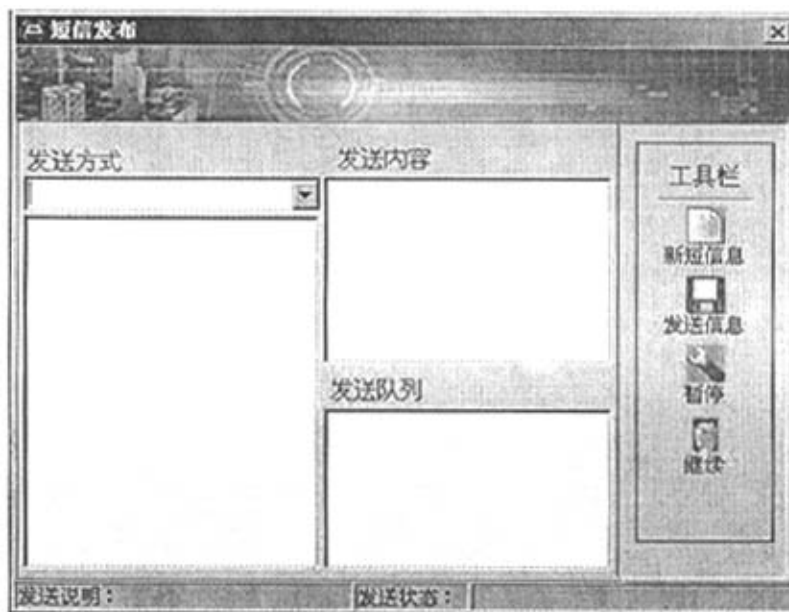


图 5.16 短信发送界面

5. 报警管理

1) 报警查询

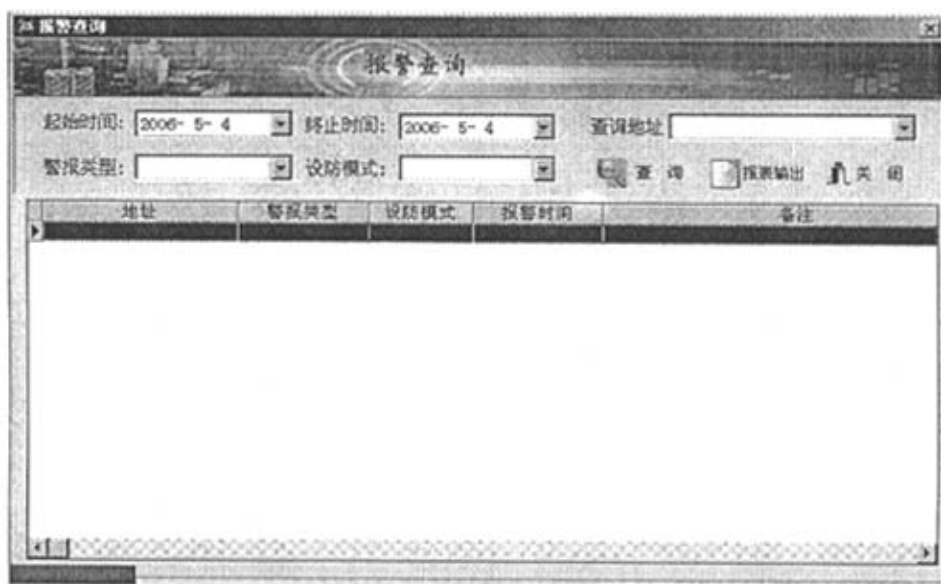


图 5.17 报警查询界面

6. 报表生成

1) 历史记录查询



图 5.18 历史纪录查询界面

2) 故障记录报表生成

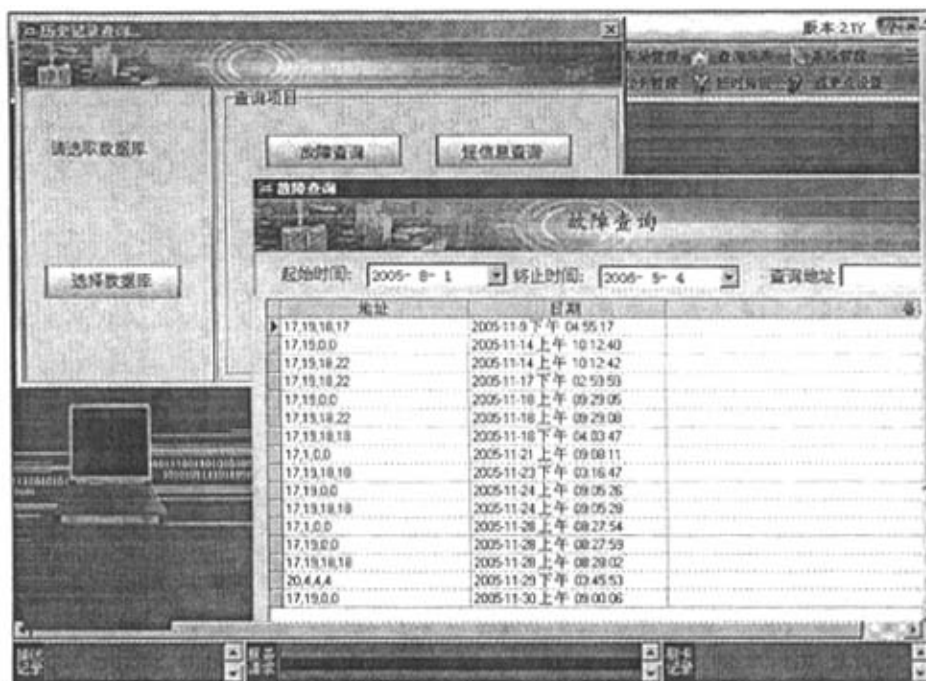


图 5.19 故障报表界面

3) 警情记录报表查询



图 5.20 警情记录报表界面

3) 短消息报表查询



图 5.21 短信报表界面

3) IC 卡刷卡记录报表



图 5.22 IC 卡刷卡记录报表界面

5.2 监控程序软件演示

1. 远程监控系统客户端程序

1) 提交住户信息登陆系统功能

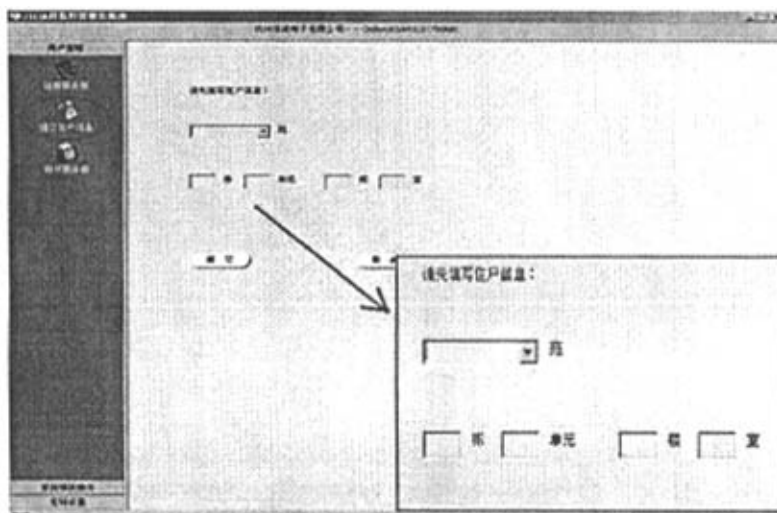


图 5.23 远程监控系统登陆界面

2) 模块状态查询功能

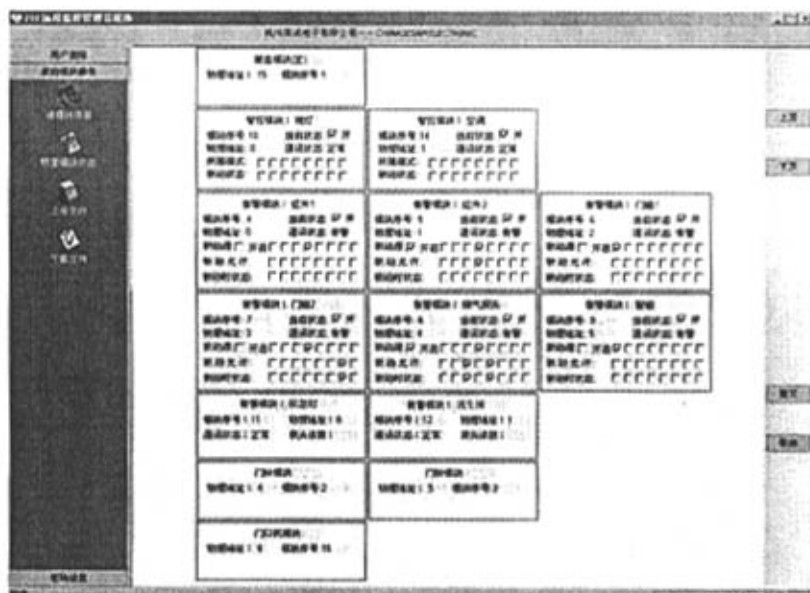


图 5.24 远程监控系统模块查询界面

3) 恢复为上次模块状态功能

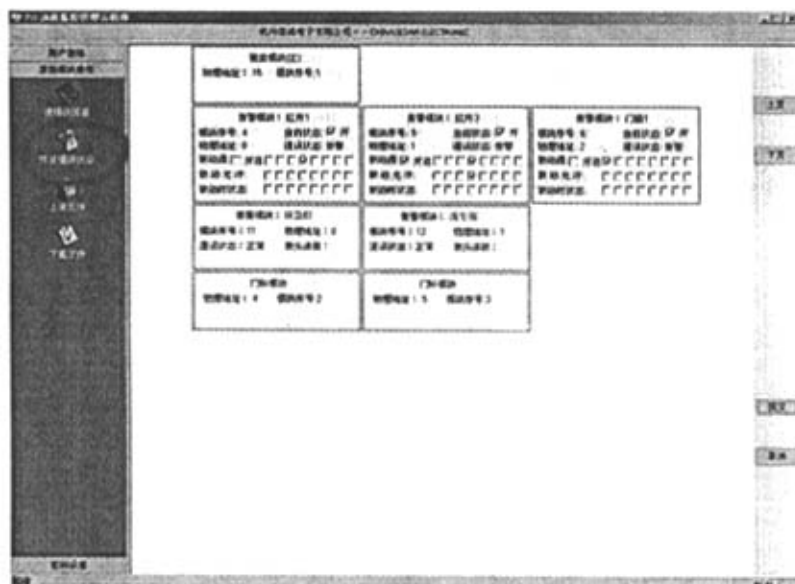


图 5.25 远程监控系统模块恢复界面

4) 文件上传功能

将本地文件上传到远程服务器上。

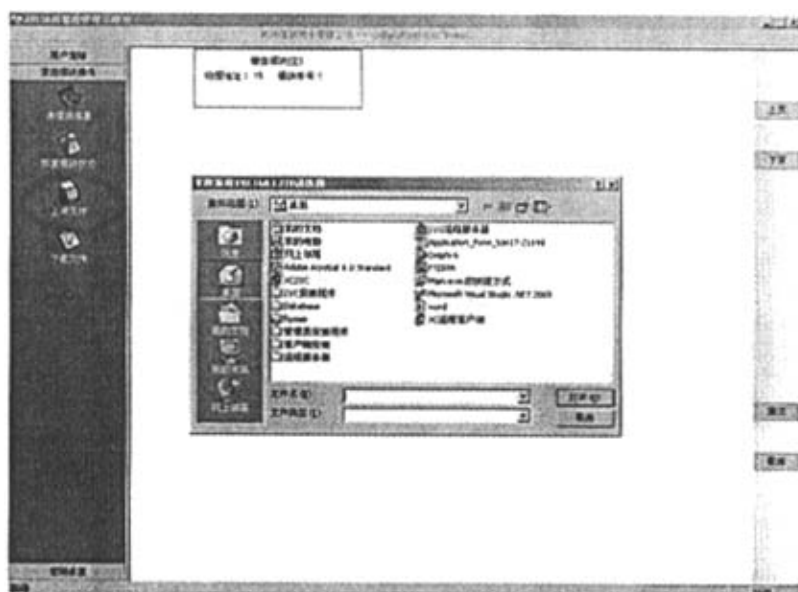


图 5.26 远程监控系统文件上传界面

5) 文件下载功能

将升级程序远程下载到本地机器上。

- 62

- 23、朱红育 史忠科, 一种高速 CAN-LonWorks 总线互联适配器的设计与实现, 电气传动, 2005 年 35 卷 12 期, 38-40
- 24、史雪飞, 胡纪五, LonWorks 现场总线技术中关于节点的设计与应用, 仪表技术与传感器, 2002 年 4 期, 27-30
- 25、程启文 陈建铎 宋彩利, Lon 网络中上位机与智能节点通信方式的研究与实现, 微电子学与计算机, 2005 年 22 卷 7 期
- 26、凌志浩. 从神经元芯片到控制网络.北京: 北京航空航天大学出版社,2002
- 27、彭海霞, 余永权. 基于 LonWorks 技术的智能家居网络节点设计与开发.现代电子技术, 2005,28(12)
- 28、文娟, 萧秋水. Visual C++.NET 网络编程与互联网应用开发 清华大学出版社 2002
- 29、陈元琰. Visual C++.NET MFC 类库应用详解 科学出版社 2004
- 30、George Shepherd David Kruglinski Programming with Microsoft Visual C++.NET Core Reference 清华大学出版社, 2004
- 31、李现勇. Visual C++串口通信技术与工程实践(第二版) 人民邮电出版社 2004.
- 32、Bruce Eckel, Chuck Allison. Thinking in C++ 2nd Edition, Volume 2: Standard Libraries & Advanced Topics. Prentice Hall PTR, 2004.
- 33、张乃孝, 裘宗燕. 数据结构 北京:高等教育出版社 1999-6
- 34、邬宽明, 现场总线技术应用选编(上), 北京: 北京航空航天大学出版社, 2003
- 35、邬宽明, 现场总线技术应用选编(下), 北京: 北京航空航天大学出版社, 2003

致谢

本论文是在王小海教授悉心指导下完成的，首先在此对王教授表示诚挚的谢意。在我的整个研究生学习阶段，无论在专业学习，还是在课题研究和论文撰写过程中，王小海教授都给予了我极大的关心、指导和鼓励。非常感谢王教授对我两年谆谆的教诲。

同时也感谢沈红老师在整个课题研究中给予的平台设计和软件整体设计方面的指导与支持。

在课题进展中，杭州信诚公司董事长樊小刚、技术总监吴建德以及信诚公司消防逃生系统设计小组成员付晓、熊典礼等工程师和同组的徐永超同学给予了我无私、诚恳的指导与帮助。在大家的团结合作下，我的课题才得以顺利的开展。

在研究生期间，教研组的陈隆道老师、章安元老师、祁才君老师等都给予我非常大的关心和指导。在此对你们表示真心地感谢。

另外，感谢实验室的师兄师姐赵云峰、周俊、李翼、李玮、汪盈、李歆裘等，还有同级的同学曹昕鸢、刘中富、费斯，他们在我搜集论文材料上给了我很多帮助，实验室气氛融洽，大家互帮互助，为我的项目发展提供了一个宽松的环境。

衷心感谢我的父母、兄长及男友，是他们在不断关心我，鼓励我，督促我，使我能顺利的完成硕士论文。

使我能顺利的完成硕士论文。

郎春华

二零零六年五月于浙江大学