

VxWorks 下基于 BSD4.4 规范的网络程序设计

张保山

摘 要 介绍实时多任务嵌入式操作系统 VxWorks 下基于 BSD4.4 规范的网络程序设计原理, 并提供了高实时应用软件中网络通信的 TCP、UDP 及组播的程序设计架构。

关键词 VxWorks BSD4.4 规范 网络程序设计

BSD(Berkeley Software Distribution, 伯克得软件发布中心) 的套接字(socket), 目前已经成为网络通信应用程序^[1, 2]的主流接口。本文以某高实时应用系统为例, 介绍实时多任务嵌入式操作系统 VxWorks 下基于 BSD4.4 规范的网络程序设计, 对其它高实时应用软件的网络程序

设计具有重要的参考价值。

1 VxWorks5.4 实时嵌入式操作系统及其 Tornado2.0 开发环境

VxWorks 是美国 WindRiver 公司提供的嵌入式操作系统, 它是实时操作系统方面的典范,

层次总排序结果如下,
主准则层

A	B ₁	B ₂	B ₃
1	0.122	0.230	0.648

次准则层

B ₁	B ₁₁	B ₁₂	B ₁₃
0.122	0.014	0.041	0.068

B ₂	B ₂₁	B ₂₂
0.230	0.0575	0.173

B ₃	B ₃₁	B ₃₂	B ₃₃	B ₃₄	B ₃₅
0.648	0.028	0.085	0.141	0.197	0.197

由层次单排序检验所得各层次一致性指标可得层次总排序的一致性指标:

B 层对 A 层的指标: $C.I. = 0.003$, $C.R. = 0.004 < 0.1$, 具有满意的一致性指标。

$B_{1i}(i = 1, 2, 3)$ 层对 B_1 层的指标: $C.I. = 0.03$, $C.R. = 0.051 < 0.1$, 具有满意的一致性指标。

$B_{2i}(i = 1, 2)$ 层对 B_2 层的指标: $C.I. = 0$, 具有完全一致性。

$B_{3i}(i = 1, 2)$ 层对 B_3 层的指标: $C.I. = 0.06$, $C.R. = 0.005 < 0.1$, 具有满意的一致性指标。

因此, 层次总排序具有满意的一致性指标。

3 结论

本文在层次分析法的原理上, 建立了舰载防空武器系统作战效能层次模型, 深入地分析了防空武器系统作战效能。通过对防空武器系统作战效能的分析可以解决如下问题:

(1) 在许多备选方案中, 采用哪个方案能够研制出最有效的系统?

(2) 以正在研制的武器系统与类似的武器系统相比, 前者是否优于后者?

(3) 防空武器系统的主要技术性能变化会对系统效能产生什么影响?

(4) 防空武器系统的主要技术性能参数之间是否存在使系统达到最佳匹配关系? 如何确定他们之间的最佳匹配关系。

参考文献

- 1 戴自立. 现代舰船作战系统. 兵器工业出版社, 1999. 8
- 2 周文松. 某型舰空导弹武器系统作战效能分析. 海军航空工程学院硕士学位论文, 1994. 3
- 3 许诚, 张福光, 吴进煌. 某型导弹武器系统装艇作战效能探讨. 海军航空工程学院学报, 1992. 4
- 4 朱齐丹, 华克强. 全武器系统优化配置的理论研究. 中国船舶科技报告, 1994. 6
- 5 朱齐丹, 安晓东. 层次分析法在舰船作战能力分析中应用. 火力与指挥控制, 1996 年第 4 期

VxWorks5.4 提供了对 TCP/IP 网络栈的支持,方便用户的剪裁,与之相适应的开发工具 Tornado2.0 提供了对嵌入式软件设计的集成开发环境,大大缩短了软件的开发周期。

本文的程序设计是基于 BSD4.4 规范的,基于 BSD4.4 规范 VxWorks 标准的网络栈有丰富的网络协议及工具^[3],包括:PPP(点对点协议)、IP(网际协议)、UDP(用户数据报协议)、TCP(传输控制协议)、DNS(域名系统)、ARP(地址解析协议及代理地址解析协议)、RARP(反向地址解析协议)、Sockets(TCP,UDP,组播;套接字)、RPC(远程过程调用)、FTP(文件传输协议)、NFS(网络文件系统)、rlogin 和 telnet(远程登录)、ICMP(网际控制报文协议)、IGMP(网际组管理协议)、SNMP(简单网络管理协议)等。

2 网络程序设计

2.1 基于 TCP 的网络程序设计

TCP (Transmission Control Protocol, 传输控制协议) 提供一种服务器-客户端通信的可靠的面向流的通信方式。TCP 提供了可靠的数据流服务,即服务器与客户端通信时需要建立应答确认的握手机制。VxWorks 下 TCP 服务器与客户机的通信模式见图 1。

建立 TCP 套接字的函数原型如下:

```
int CreateTCP(char *serverName, unsigned short wPortNum, int TcpType);
```

参数说明

*serverName :服务器的名字或 IP 地址;
wPortNum :无符号整型数,使用的端口号;

TcpType :TCP 套接字类型,其类型为:

```
enum eTCP_TYPE
```

```
{
```

```
TCP_CLIENT = 0, /* 客户端
```

套接字 */

```
TCP_SERVER = 1 /* 服务器套接字 */
```

套接字 */

```
};
```

返回值

成功时返回建立套接字的文件描述符,否则返回 ERROR。

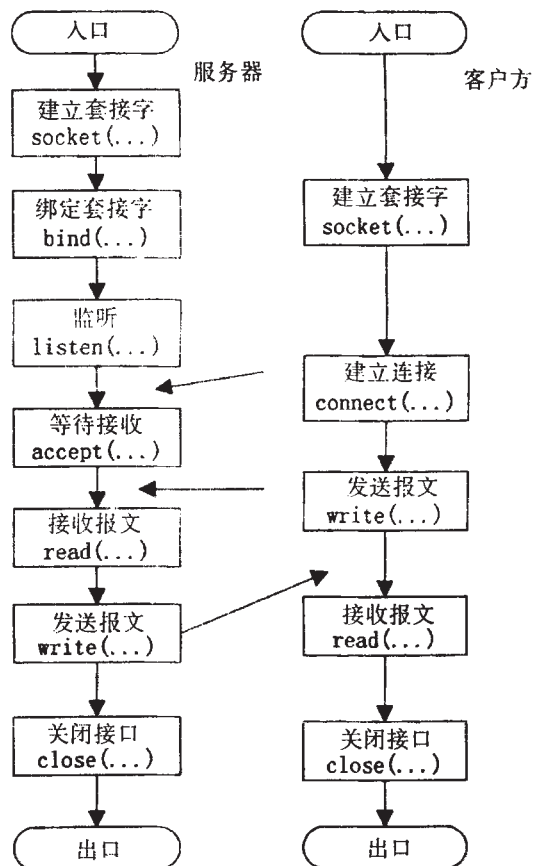


图 1 VxWorks 下 TCP 服务器与客户机的通信模式

建立 TCP 套接字流程图见图 2。

2.2 基于 UDP 的网络程序设计

UDP (User Datagram Protocol, 用户数据报协议) 提供一种服务器-客户端通信的不可靠无连接的通信方式,即服务器与客户机之间在交换信息时不需要建立相应的握手信号,信息发出去不管,因此在高实时领域网络协议普遍采用 UDP 时,省去了确认应答的过程。VxWorks 下 UDP 服务器与客户机的通信模式见图 3。

创建 UDP 套接字流程图见图 4。

考虑到网络节点既可充当服务器,又可充当客户端的角色,在进行 UDP 程序设计时对创建 UDP 套接字进行了统一的考虑。

创建 UDP 套接字的函数原型如下:

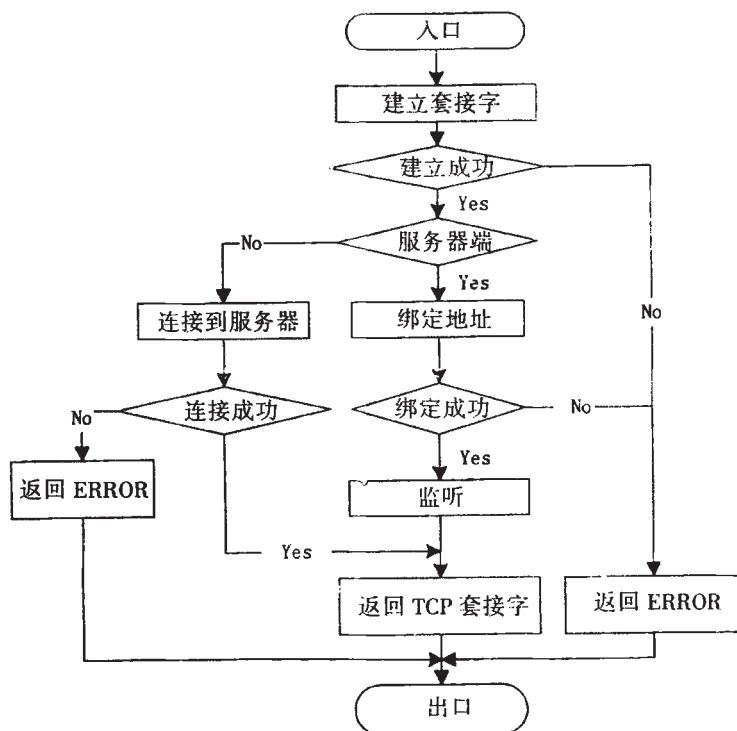


图 2 创建 TCP 套接字流程图

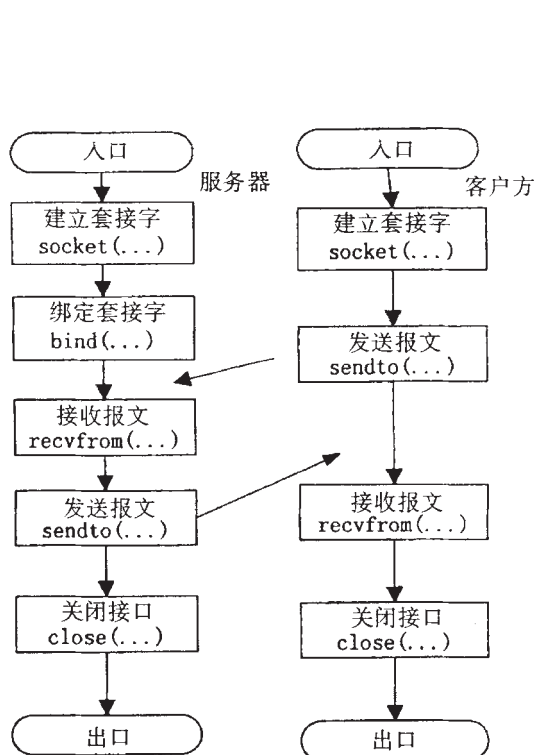


图 3 VxWorks 下 UDP 服务器 - 客户端通信模式

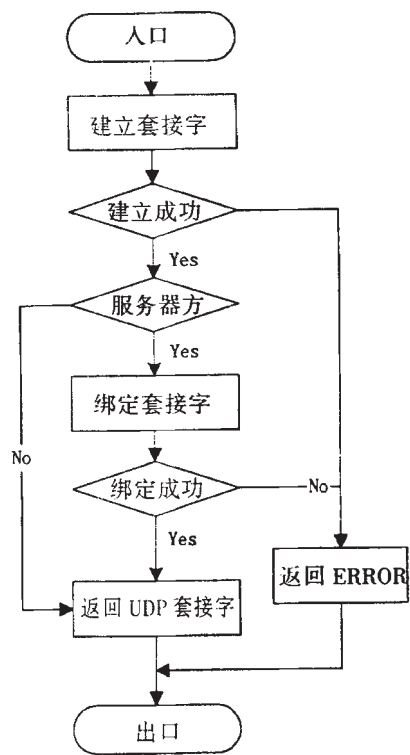


图 4 创建 UDP 套接字流程图

```
int CreateUDP(char *serverName, unsigned
short wPortNum, int UdpType);
```

参数说明

*serverName :服务器的名字或 IP 地址 ;
wPortNum :无符号整数型 ,使用的端口号 ;
UdpType :UDP 类型 ,其类型为 :

```
enum eUDP_TYPE
{
```

```
    UDP_CLIENT = 0, /* 客户端套
```

```
接字 */
```

```
    UDP_SERVER = 1 /* 服务器套
```

```
接字 */
```

```
};
```

返回值

成功时返回建立套接字的文件描述符 ,否则返回 ERROR。

2.3 基于组播的网络程序设计

对 VxWorks 操作系统 ,其高层协议 (TCP/ IP)支持能力强 ,因此 VxWorks 下点对点、广播通信方式易于实现 ,但广播发送意味着信息发送者要向网络上的每个节点交付一个分组的副

本 ,其缺点是严重消耗了网上的资源 ,增加了网络的负载 ,一种比较好的方式是使用组播 ,组播也称为多播 (multicast) ,高实时应用领域的网络通信在多个节点收到同一报文时 ,往往采用组播 ,这样允许网络上的每个节点是否参与到组播中去。此时 ,网络系统中各节点采用 C 类 IP 地址 ,组播则使用 D 类网络地址 ,Internet 的网络地址分类及其范围见表 1。

表 1 网络地址的分类及其范围

网络地址类别	范围
C 类	192. 0. 1. 0 ~ 223. 255. 255. 0
D 类	224. 0. 0. 0 ~ 239. 255. 255. 255

组播是将同一数据包传送到多个 IP 地址 ,组播是 IP 层的特性 ,但访问该函数 ,应用程序要用到 UDP 套接字。组播一数据包 ,发送方 (客户) 仅向组播地址发送即可 ,它们使用正常的 UDP 套接字。如果欲接收组播发送的数据包 ,应用程序需加入组 ,必须设置相应的套接字选项。选项的描述如下表 2。

表 2 组播选项设置参数

命 令	参 数	描 述
IP_MULTICAST_IF	struct in_addr	选择组播缺省接口
IP_MULTICAST_TTL	CHAR	设置组播缺省的生存时间 (TTL)
IP_MULTICAST_LOOP	CHAR	允许/禁止组播的循环 LoopBack
IP_ADD_MEMBERSHIP	struct ip_mreq	加入组播
IP_DROP_MEMBERSHIP	struct ip_mreq	脱离组播

设置组播相应的选项调用函数 setsockopt
()。

组播的函数主要如下 :

a) 加入组地址

```
int mcastJoinGroup(.....);
```

b) 脱离组地址

```
int mcastLeaveGroup(.....);
```

c) 组播接收初始化

```
int mcastRecvInit(.....);
```

d) 退出组播

```
int mcastRecvQuit(.....);
```

e) 设置组播的生存时间

```
int SetCastTTL(.....);
```

上述程序在成功时返回 OK ,否则返回 ERROR。

组播接收初始化创建组播服务器方的套接字流程图见图 5。

2.4 高实时通信中的网络程序设计

高实时网络体系结构与 ISO/OSI 网络七层结构略有不同 ,其对照表见表 3。

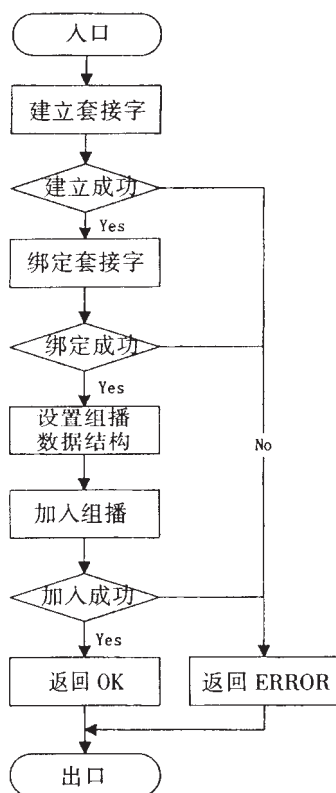


图5 创建组播套接字流程图

由表 3 可以得出如下结论，高实时系统中网络体系结构的分层及其使用的协议是在满足 ISO/OSI 网络分层规定的基础上，进行适当剪裁以满足高实时网络通信的要求。

高实时通信的一个比较典型的应用是水面舰艇的舰炮武器系统。图 6 是某舰炮武器系统的网络图，其中 ZK-1、ZK-2、……、ZK-n 是作战系统的节点，它们连接在作战系统网络上，HK-1、HK-2、HK-3、……、HK-m 是舰炮武器系统的节点，例如跟踪雷达、光电跟踪仪、捷联式垂直基准、火控设备等网络节点，它们均连接在火控系统网络上，HK-1 与 HK-2 作为一个组接入作战系统网络，接收作战系统的控制及信息。这种分布式的网络体系结构是建立在以太网基础上，通过双冗余网络交换机进行通信，完成信息的可靠高速实时传输，方便系统的扩充与剪裁，满足海军对舰炮武器系统通用化、系列化、组合化的要求。

在具体软件开发过程中，UDP 与 TCP 的不同（组播由于采用 UDP，可以归入 UDP 一类），对于网络报文的接收有 4 种不同的方式。

表 3 OSI/ISO 网络协议分层与高实时网络协议分层对照表

OSI/ISO 网络分层	协议种类	高实时网络分层	高实时网络协议种类
应用层	FTP、SMTP、TELNET、NFS 等	应用层	
表示层			
会话层			
传输层	TCP、UDP	传输层	UDP
网络层	IP、ARP、RARP、ICMP、IGMP	网络层	IP、ARP、ICMP
数据链路层	100BASE-TX、10BASE-T、FDDI、Token Ring 等	链路层	100BASE-TX 10BASE-T
物理层	双绞线、光纤、同轴缆等	物理介质层	双绞线、光纤

- 循环的无连接 UDP 接收方式；
- 并发的多任务的无连接 UDP 接收方式；
- 循环的有连接 TCP 接收方式；
- 并发的多任务的有连接 TCP 接收方式。

在循环的网络报文接收方式中，服务器在接收完一个节点的数据后，再处理下一个节点

的报文接收，并发的多任务方式则是对与本节点通信的每一个节点建立一个任务，其优先级相同，通过轮转时间片的方式完成网络报文的接收。在舰炮武器系统网络报文接收过程中，可以采用循环接收方式，也可以采用并发多任务的方式。图 7 是循环无连接 UDP 接收方式的流程图。

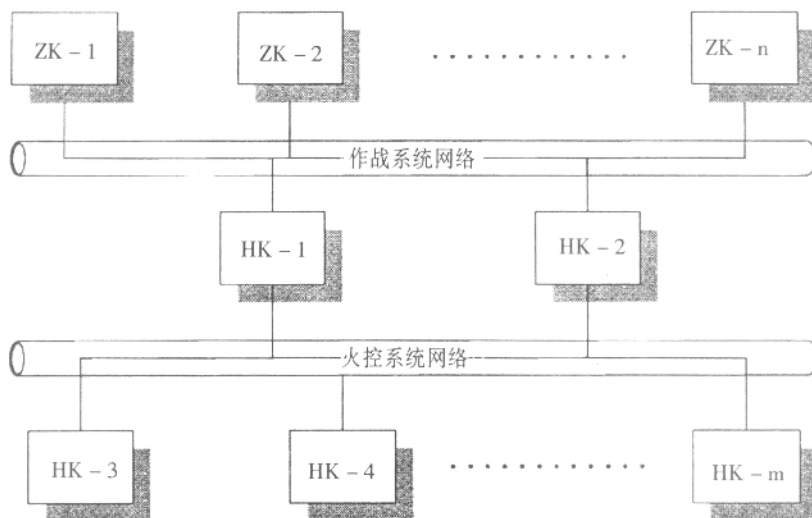


图6 某高实时网络体系结构图

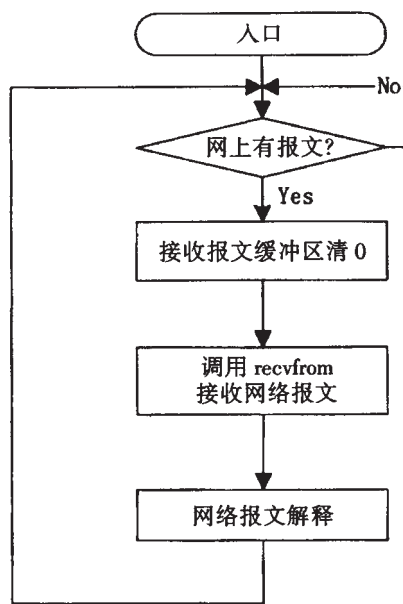


图7 循环无连接 UDP 接收方式的流程图

网络报文的发送则要考虑到网络流量, 尤其当系统网络上某个节点未开机或者故障时, 发送方应适当减少网络流量, 当该节点正常工作时, 应能恢复到正常工作情况下的网络流量。一种常用的方法是通过报文计数器来动态控制网络流量, 高实时系统中网络各节点一般采用周期发送的方式, 通信速率较高 (一般 50

次/秒)。在主导任务中, 将该报文计数器加 1, 当该报文计数器大于某一固定门限时, 在发送时则降低网络上的流量, 可以每秒发一次, 否则按 50 次/秒发送。在网络报文接收任务中如果收到该网络报文则将报文计数器清 0, 可以重新恢复到 50 次/秒的通信速率。减少网络流量的另一方法是在 UDP 报文发送过程中, 对关键信息人为增加“确认应答”标志, 或者状态改变时发送。这样就达到了合理安排网络流量, 保证网络负载平衡的目的。

3 结束语

本文中所提到的网络程序设计原理及方法, 已经成功应用于某舰炮武器系统的高实时网络通信中, 达到预期效果。以上程序在 Tor-nado2.0 中调试通过。

参考文献

- 1 Internetworking with TCP/IP Volume I: Principles, protocols and Architecture, Third Edition, by Douglas E. Comer, 1995.
- 2 Internetworking with TCP/IP Volume III: Client - Server Programming and Applications, by Douglas E. Comer and David L. Stevens.
- 3 VxWorks Network Programmer's Guide 5.4, Edition 1.