Exp 4：ICMP Ping 和 Checksum

目的：学习 ICMP 协议和校验码checksum的计算。

摘要：可将ITS用交叉线直接与 PC 连接，在PC端用“ping”指令练习发送ICMP报文此外，通过MDDL语言，学生可以更清楚了解ICMP 协议的工作流程与作用。

时间：3 hrs。

一、网络拓扑

A：Single LAN

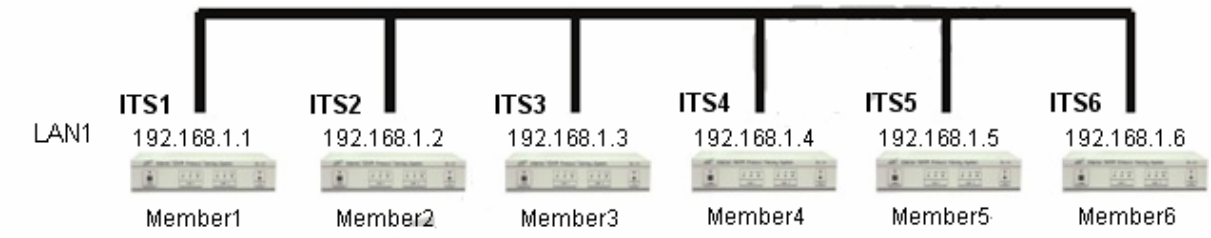


图4.1

B：

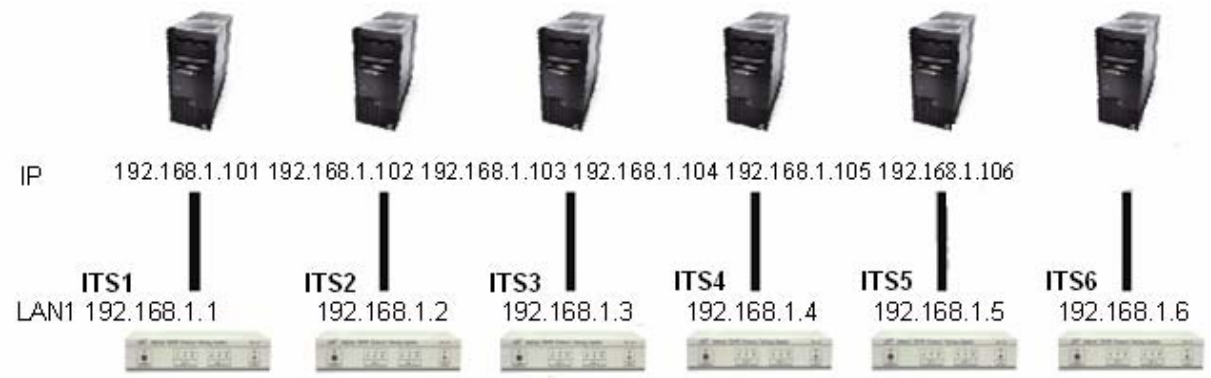


图4.2

二、技术背景

因特网控制消息协议ICMP（Internet Control Message Protocol）是使用在当IP报文传送发生问题时所产生的响应消息，它也能显示出不能到达的IP网络，当节点过负载，当错误发生在IP包头信息等等，该协议也常被使用在确认路由器是否正确绕送数据包到清单中的目标地址。

计算机中的ping指令会发出名为Echo Request的ICMP报文，主要用来是验证网络上

的点对点联机是否正常，同时也能收集一些联机效能的信息。例如：量测报文来回时间(RTT，Round Trip Time)和远程主机失去响应后的时间数值。计算机主机的ping指令发出后，每个Echo Request 报文都包含一个顺序码(从0开始)，这个顺序码会累加，并有一个时间戳记(timestamp)值显示传输时间。而每一次传回的 Echo Reply报文都会被接受，计算出 RTT(in milliseconds)并用文字显示成一列在屏幕上。

每个 ICMP报文里都包含了校验码(checksum)字段，checksum 是一种为了保证数据包的正确性，计算方式是把报文的数据依序排成一栏的16位整数，全部相加后一起使用 1 补码计算，计算出来的值就是checksum字段的数值。当报文被目的端接收后会将该报文的数据再计算一次checksum与封包 checksum字段的数值做比较以判断整个报文的数据是否正确。

Note：

1的补码（反码）计算：1's 补码溢位位会被加在最后面而 2's 补码溢位位会被丢弃。

1 的补码（反码）：交换 0 与 1 的位值。

ICMP 协议虽然与 IP 协议同在第三层，但 ICMP 本身格式并不具备传送能力，而是会被加在 IP packets 里，再通过 Ethernet frames 传送，其形式如下：

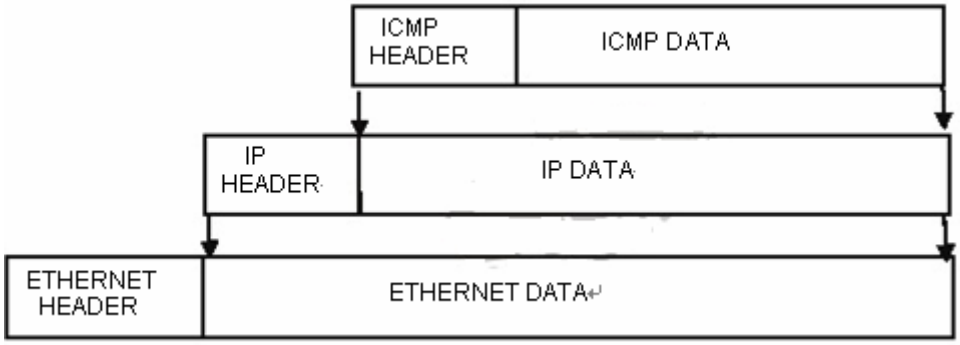


图4.3

此外，ICMP 并没有统一的报文格式以供全部 ICMP 消息使用，不同的ICMP类别分别有不尽相同的报文字段。ICMP协议中，Echo Request(type 8)和Echo Reply(type 0)封包格式如下：

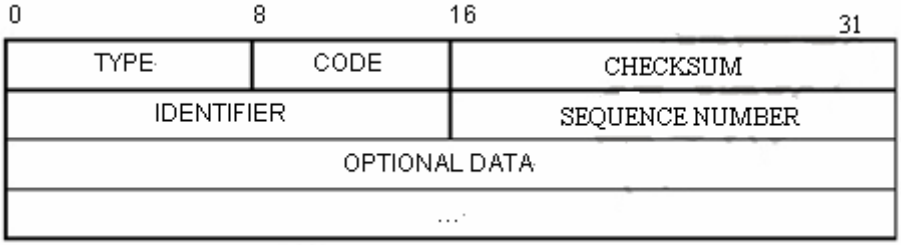


图4.4

TYPE(8 bits)：指定 ICMP报文类型(request (8) 或 reply (0))

CODE(8 bits)：码与 TYPE 搭配使用表示该报文的功能，在 ICMP 询问及响应时皆设为‘0’

IDENTIFIER(16 bits)：用来辨识响应报文对应的询问报文

SEQUENCE NUMBER(16 bits)：报文传送序号

OPTIONAL DATA(长度不定)：选择性数据

图4.5为一个标准的 ICMP Echo Request 封包内容：

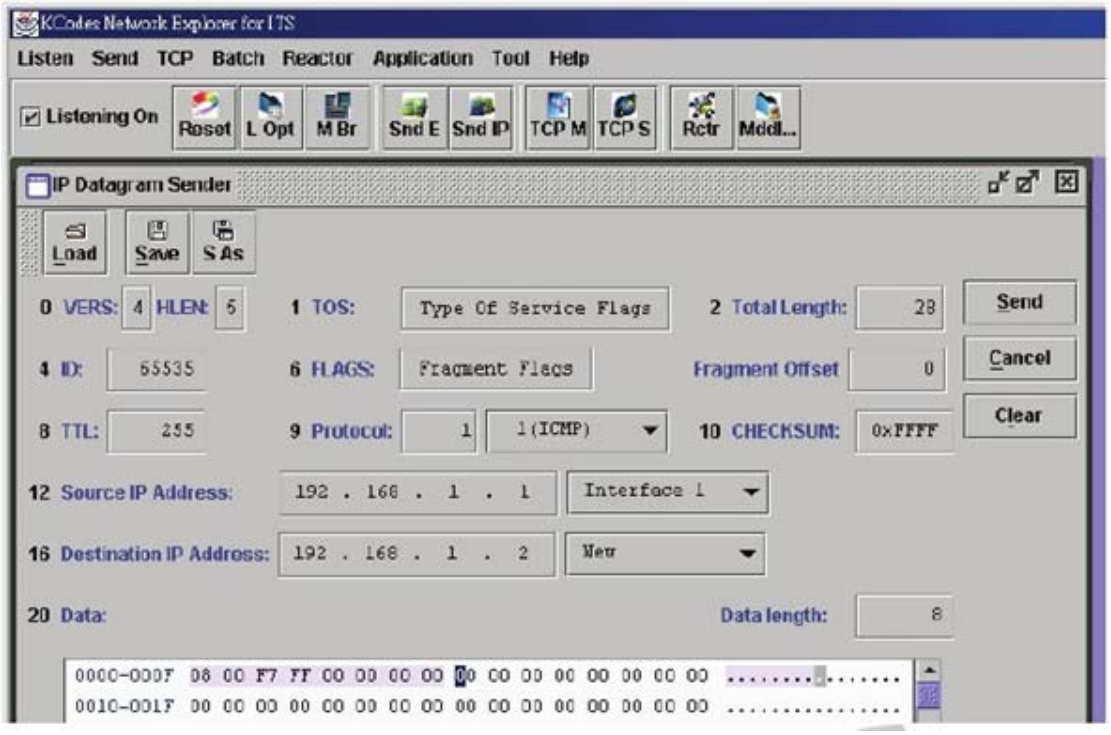


图 4.5

三、实验步骤

1、从 PC 上观察 ICMP

1）见图 4.6，我们在 PC 端使用 RS-232 串口线连接 ITS 机器，并用交叉线从 ITS 端的 LAN1 口连接到 PC 的网卡上，整个拓扑请参考网络拓扑 B。

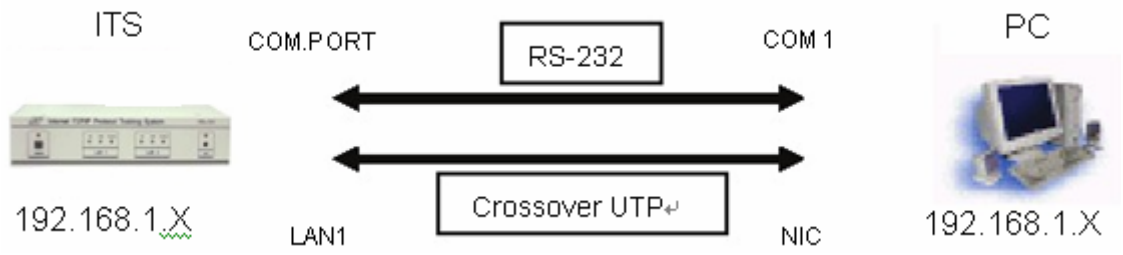


图 4.6

2）参考实验 3 内容“查看 PC 机上的 ARP 列表”的实验步骤，设定本机电脑的 IP 地址和子网掩码。本实验的范例中，输入的 IP 地址为“**192.168.1.101**”，如图 4.7 所示。

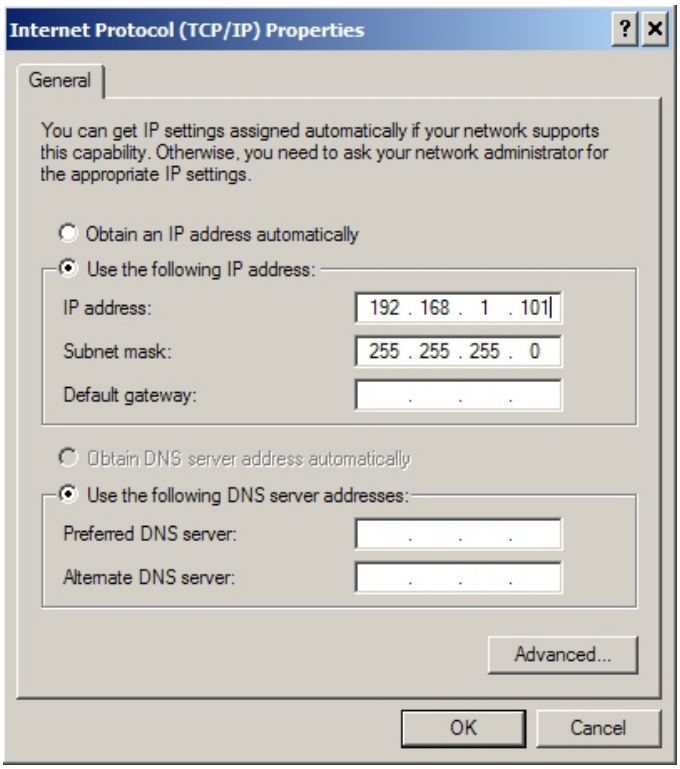


图 4.7

3）执行 XCLIENT.BAT，打开 ITS 应用程序界面 KCodes Network Explorer。勾选 Listening On。

4）从 Listen menu 中选择监听等级（ Listening Level ），将 Interface Frames 打勾。

5）再从 Listen menu 中选择 New Memorized Message Brower 。如图 4.8 所示，网络信息浏览器（Network Message Browser）将会被打开，可以及时监听整个网络上的信息传输。

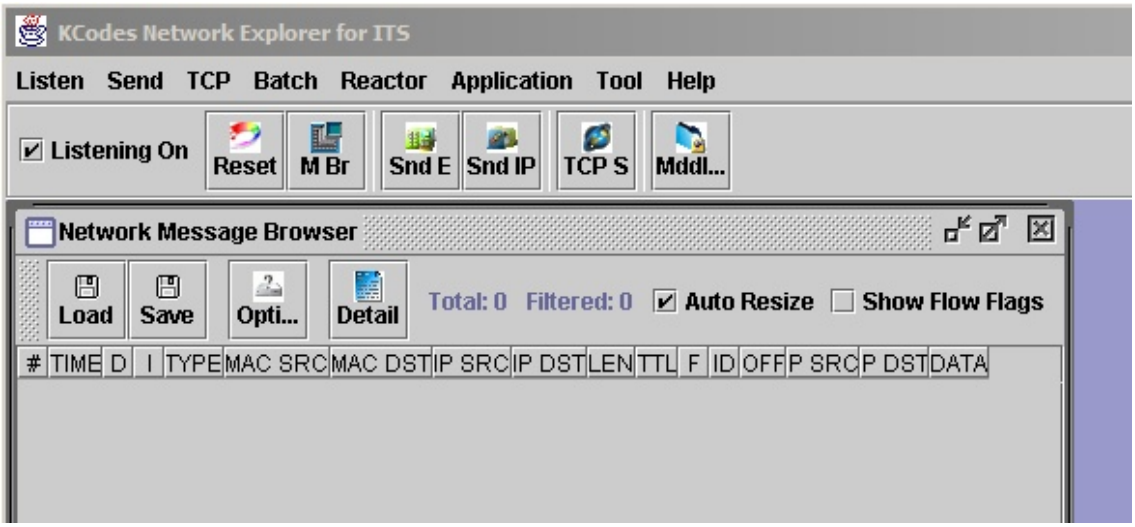


图 4.8

6）打开 windows 的命令行界面（Command Prompt）. 输入 **ping <*your ITS IP address*>** 命令，执行后每一位 Member 应该都会收到成功的 4 次 reply，如图 4.9 所示.

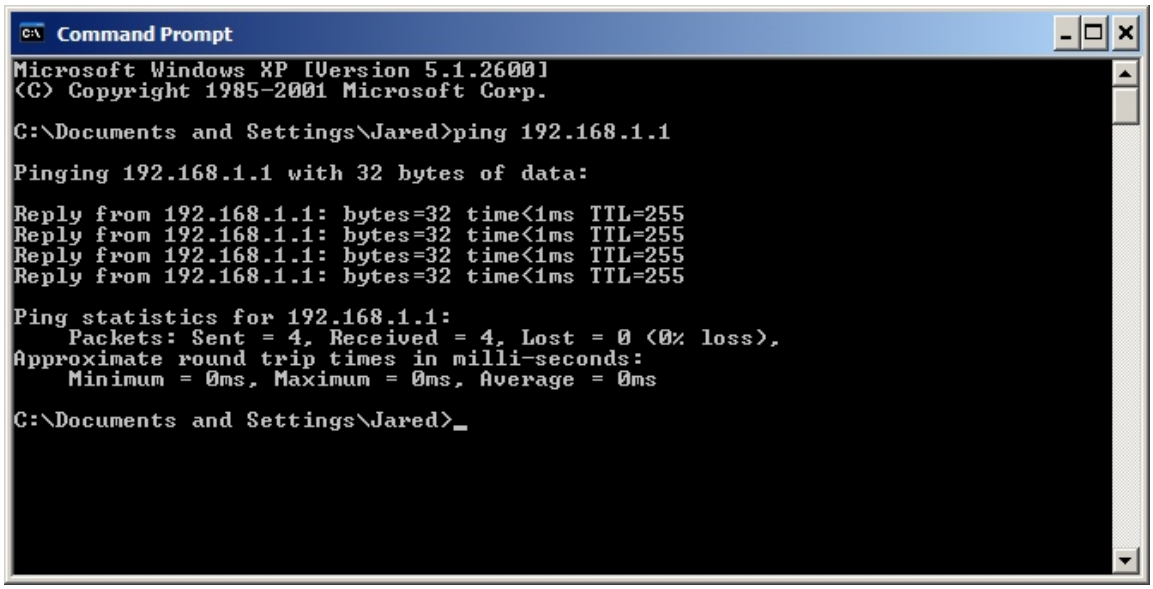


图 4.9

7）见图 4.10，从网络信息浏览器中，我们可以看到数个来自 PC 端或 ITS 端的 ICMP 报文，选择任意一个报文并单击 **Detail** 按钮，即可观察到更详细的报文内容。

Network Message Browser

Load | Save | Opti... | Detail | Total: 8 Filtered: 8 | Auto Resize | Show Flow Flags

#	TIME	D	I	TYPE	MAC SRC	MAC DST	IP SRC	IP DST	LEN	TTL	F	ID	OFF	P SRC	P DST	DATA
1	10:33:16.50	I	1	IP/ICMP	00:14:38:15:9D:C8	00:94:12:15:01:3D	192.168.1.101	192.168.1.1	74/60	128		10,316	0			08.00.2C.5C.02.00...
2	10:33:16.50	O	/1	IP/ICMP	00:94:12:15:01:3D	00:14:38:15:9D:C8	192.168.1.1	192.168.1.101	74/60	255		25	0			00.00.34.5C.02.00...
3	10:33:17.51	I	1	IP/ICMP	00:14:38:15:9D:C8	00:94:12:15:01:3D	192.168.1.101	192.168.1.1	74/60	128		10,317	0			08.00.2B.5C.02.00...
4	10:33:17.51	O	/1	IP/ICMP	00:94:12:15:01:3D	00:14:38:15:9D:C8	192.168.1.1	192.168.1.101	74/60	255		26	0			00.00.33.5C.02.00...
5	10:33:18.51	I	1	IP/ICMP	00:14:38:15:9D:C8	00:94:12:15:01:3D	192.168.1.101	192.168.1.1	74/60	128		10,318	0			08.00.2A.5C.02.00...
6	10:33:18.51	O	/1	IP/ICMP	00:94:12:15:01:3D	00:14:38:15:9D:C8	192.168.1.1	192.168.1.101	74/60	255		27	0			00.00.32.5C.02.00...
7	10:33:19.51	I	1	IP/ICMP	00:14:38:15:9D:C8	00:94:12:15:01:3D	192.168.1.101	192.168.1.1	74/60	128		10,319	0			08.00.29.5C.02.00...
8	10:33:19.51	O	/1	IP/ICMP	00:94:12:15:01:3D	00:14:38:15:9D:C8	192.168.1.1	192.168.1.101	74/60	255		28	0			00.00.31.5C.02.00...

图 4.10

2、发送 ICMP

A. ITS 状态设定

1）在 Hubox 上将网络连接如图 4.11 所示，再接下来的步骤里，需要两两配对，所以我们会有三个小组：Member1 与 Member2 一组，Member3 与 Member4 一组，Member5 与 Member6 一组。

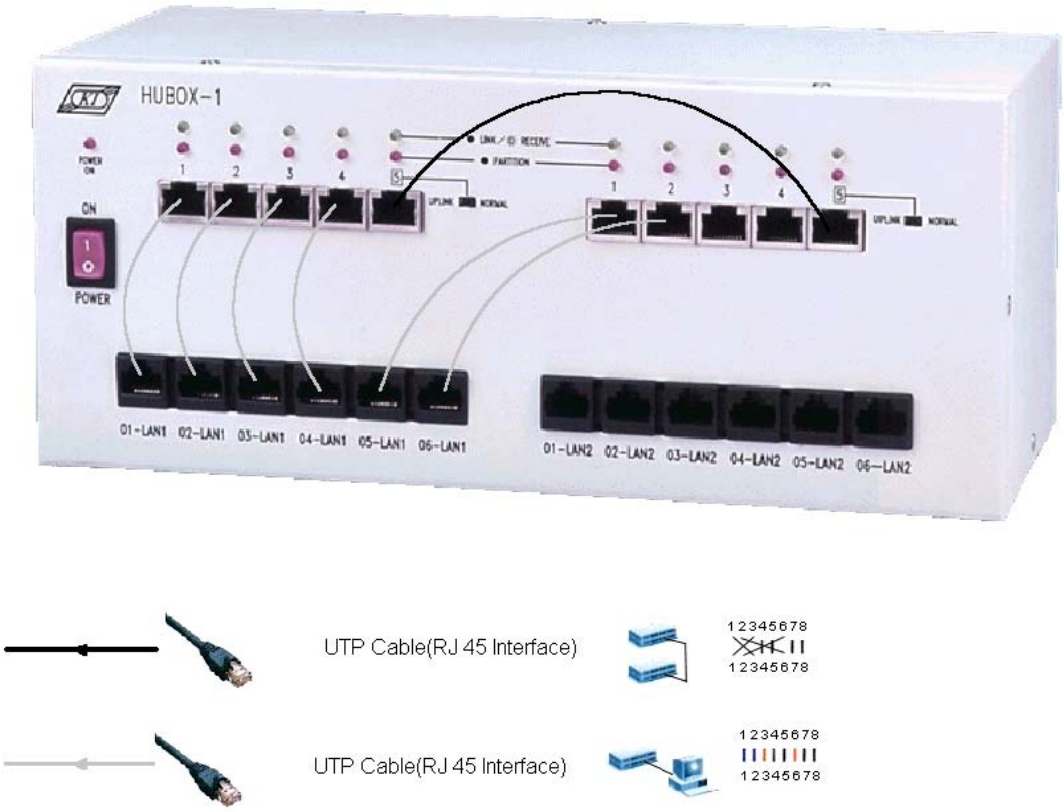


图 4.11

2）打开网络配置设定界面 **Network Configuration** 。

3）参照网络拓扑 A，将自己的 IP 地址输入到 **IP Setting of Interface 1** 的文本框中，并且输入**"255.255.255.0"** 到子网掩码的文本框中，见图 4.12。

4）选择为 **Host** 模式，并单击 **Set & Close** 按钮。

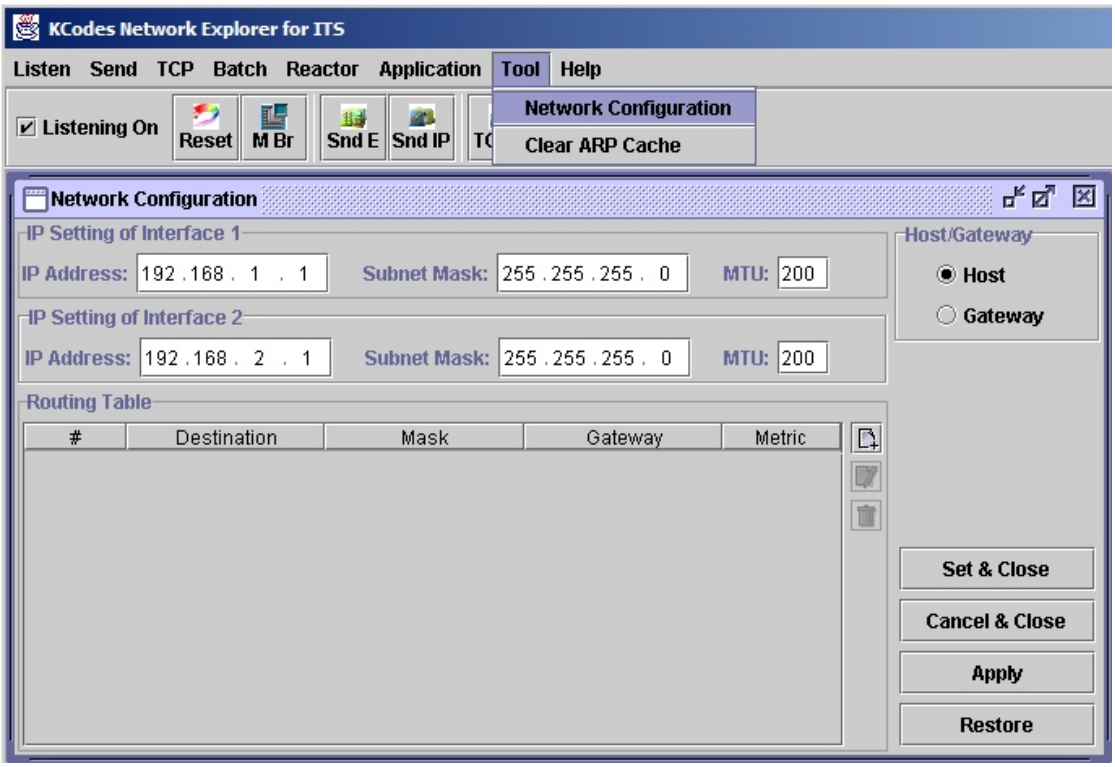


图 4.12

B. 发送 ICMP Echo Request

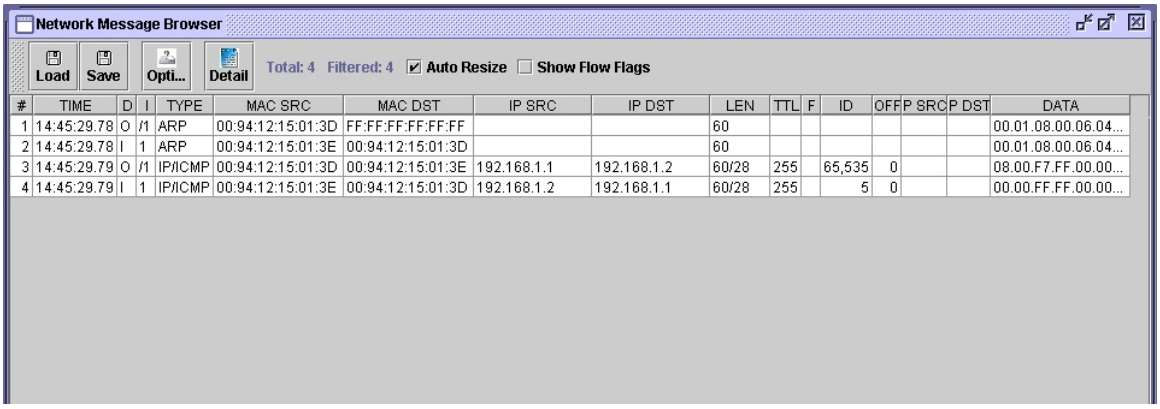
5）勾选 **Listening On**。

6）从 **Listen menu** 中选择监听等级（**Listening Level** ），将 **Interface Frames** 打勾。

7）在从 Listen menu 下单击 **New Memorized Message Brower** 打开网络信息浏览器，开始及时监听整个网络上的信息传输。

8）从 Send menu 中选择 **Send Interface Frame** ，打开以太网帧发送界面（Network Message Sender）。

9）参考图 4.5, 编辑一个 ICMP Echo Request 并发送给你的搭挡，然后几乎在同一时间，你应该会收到一个从搭挡那端发回来的 ICMP Reply 报文，如图 4.13 所示。



#	TIME	D	I	TYPE	MAC SRC	MAC DST	IP SRC	IP DST	LEN	TTL	F	ID	OFF	P SRC	P DST	DATA
1	14:45:29.78	O	/1	ARP	00:94:12:15:01:3D	FF:FF:FF:FF:FF:FF			60							00.01.08.00.06.04...
2	14:45:29.78	I	1	ARP	00:94:12:15:01:3E	00:94:12:15:01:3D			60							00.01.08.00.06.04...
3	14:45:29.79	O	/1	IP/ICMP	00:94:12:15:01:3D	00:94:12:15:01:3E	192.168.1.1	192.168.1.2	60/28	255		65,535	0			08.00.F7.FF.00.00...
4	14:45:29.79	I	1	IP/ICMP	00:94:12:15:01:3E	00:94:12:15:01:3D	192.168.1.2	192.168.1.1	60/28	255		5	0			00.00.FF.FF.00.00...

图 4.13

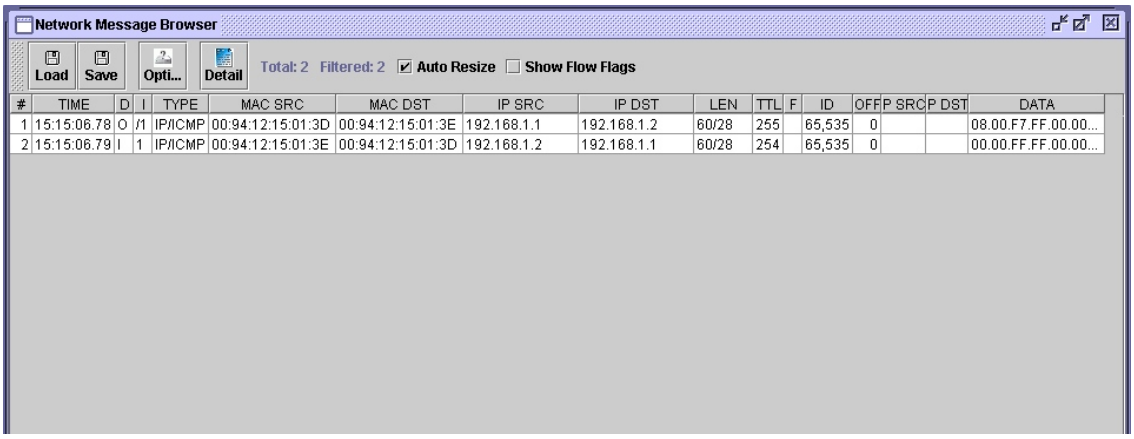
3、从 MDDL 平台自动回复 Echo Reply 报文

10）重新打开一个新的网络信息浏览器（Network Message Browser）.勾选 **Listening On**。

11）从 Reactor menu 中找到并执行 **MDDL Reactor Panel** ，打开 MDDL 编辑工具。

12）在程序编辑界面上，单击 **Load** 按钮.调用 IcmpEchoResponseFull.mddl 程序（路径为 C: \XClient \Data \Mddl \Tutorial \Ex04 \IcmpEchoResponseFull.mddl），最后单击 **Upld** 按钮。

13）试着发送一个 ICMP Echo Request 报文给你的搭挡，几乎在同一时间，你将收到一个从你搭挡端发回来的（由 MDDL 程序判断后发送出的）ICMP Reply 报文，如图 4.14 所示.



#	TIME	D	I	TYPE	MAC SRC	MAC DST	IP SRC	IP DST	LEN	TTL	F	ID	OFF	P SRC	P DST	DATA
1	15:15:06.78	O	/1	IP/ICMP	00:94:12:15:01:3D	00:94:12:15:01:3E	192.168.1.1	192.168.1.2	60/28	255		65,535	0			08.00.F7.FF.00.00...
2	15:15:06.79	I	1	IP/ICMP	00:94:12:15:01:3E	00:94:12:15:01:3D	192.168.1.2	192.168.1.1	60/28	254		65,535	0			00.00.FF.FF.00.00...

图 4.14

四、实验讨论

1、在发送 ICMP 这个实验阶段中，请试着发出 checksum 错误的 ping 给 ITS 和 PC，然后观察其变化。

2、分析与讨论 ARP packet 在此实验中所扮演的角色或其功能。

3、在以下几个情况下，试着使用你的 ITS 发出 IP 封包(IP datagram)给其它 ITS：

1）目的 IP 地址(destination IP address)为同一个子网络但无人使用

2）目的 IP 地址不在同一个子网络(subnet)

3）将 IP header 中的 protocol type 设为 50

REACTOR PROGRAM

1、**IcmpEchoResponseFull.mddl**

```
IP_RECEIVED_HANDLER
{
    IF(S.IP_PROT != CNST_IP_PROT_ICMP || S.IP_DATA.ICMP_TYPE_CODE !=
CNST_ICMP_TYPE_CODE_ECHOREQ)
        RETURN;

    SEND_OUT_IP WITH_DATA
    {
        T                          = S                                ,
        T.IP_TTL                   = {0xFF}                           ,
        T.IP_ADDRSRC               = S.IP_ADDRDST                     ,
        T.IP_ADDRDST               = S.IP_ADDRSRC                     ,
        T.IP_DATA.ICMP_TYPE_CODE   = CNST_ICMP_TYPE_CODE_ECHOREPLY ,
        T.IP_DATA.ICMP_CHKSUM      = {0x00, 0x00}                     ,
        T.IP_DATA.ICMP_CHKSUM      = CHECKSUM(T[20,])                 ,
        T.IP_LEN                   = LENGTH(T)                        ,
        T.IP_HEADERCHKSUM          = {0, 0}                           ,
        T.IP_HEADERCHKSUM          = CHECKSUM(T.IP_HEADER)
    }
    DISCARD_MESSAGE;
}
```