

Exp 3: 地址解析协议 (ARP)

目的: 学习地址解析协议 ARP (Address Resolution Protocol) 的基本原理。

摘要: 本实验可以使用 ITS101 和 MDDL 语言发出 ARP Request 帧, 让学生了解 ARP 协议的执行方式。另一方面也可将 ITS101 与 PC 直接联机, 观察并学习 PC 上的 ARP 列表(ARP Table)的功能。

时间: 3 hrs。

一、网络拓扑

A: Single LAN

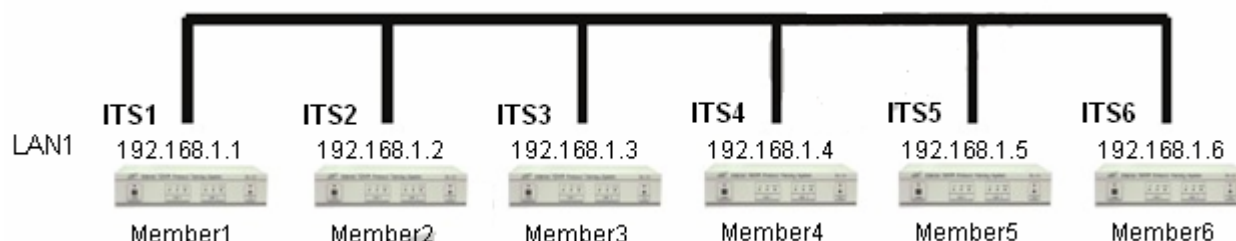


图 3.1

B:

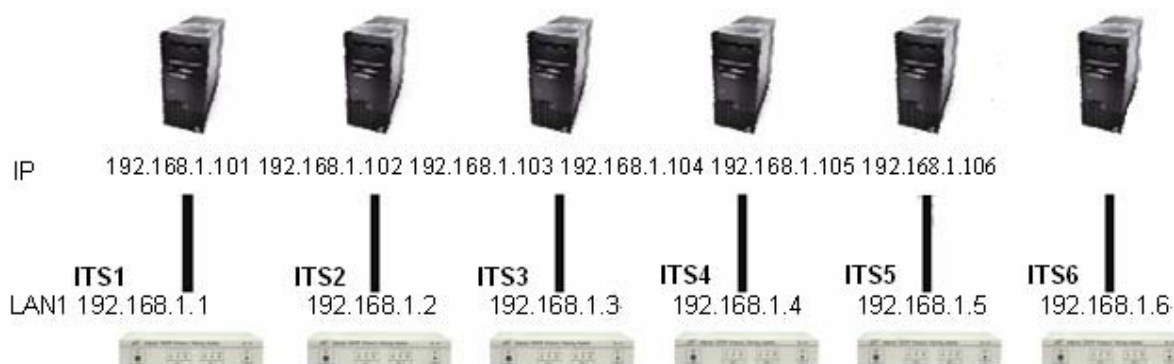


图 3.2

二、技术背景

地址解析协议(ARP), 是属于英特网协议(IP)中的一种协议, 主要功能为对应 IP 地址和物理地址(MAC address), 此协议工作在网络层(Internet Layer), 更具体一点来说, 就是当 IP 要被使用在以太网网络时的一种沟通机制。图 3.3 为 ARP 的运行机制。说明了计算机 X 要如何连接在同局域网络中的计算机 Y, 并传送 IP 报文(IP datagram)。

如果用自然语言来说明ARP请求(ARP Request)的报文内容,大致上就像:「谁是 Y.Y.Y.Y? 请告诉 X.X.X.X」。其中 X.X.X.X 和 Y.Y.Y.Y 分别代表了计算机 X 与 Y 的 IP 地址。

ARP的运作,是在以太网广播传送,见图3.3(a),即ARP请求是被加在以太网广播帧(Ethernet broadcast frame)内传送出去的,ARP帧的类型值是 0x0806。广播之后,所有同一个局域网络内的主机系统都会收到这个请求帧,此时只有其中一个主机,也就是计算机 Y 会响应一个 ARP 回复(ARP Reply)帧,而其它系统将会忽略不予响应。目标系统计算机 Y 通过 ARP回应帧会告诉计算机 X:「我是 hh:hh:hh:hh:hh:hh, YYYY 就是我。」其中的hh:hh:hh:hh:hh:hh 为计算机 Y 的MAC地址。如此一来计算机 X 就可以透过ARP,用计算机Y的IP地址找寻到实体的计算机 Y 的MAC地址,如图 3.3(b)所示。

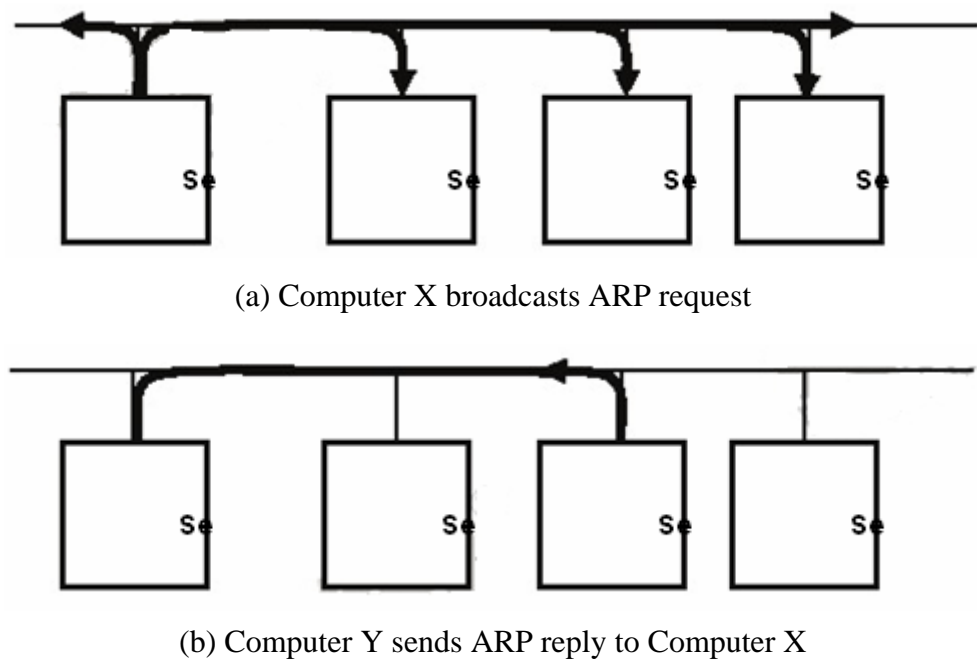


图 3.3

ARP 被加入在以太网帧(Ethernet frame)的图示如下:

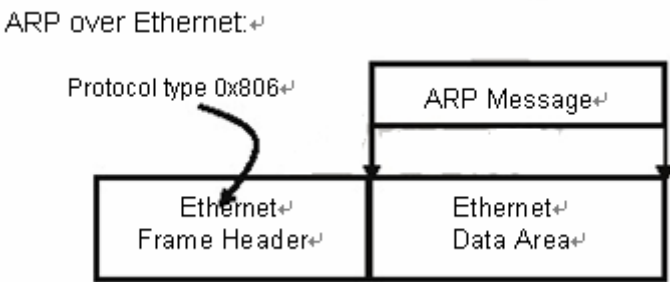


图3.4

ARP 协议结构图如下:

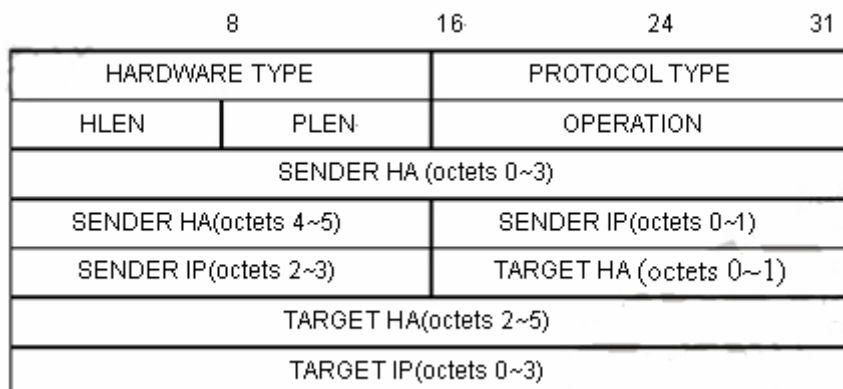


图3.5

硬件类型(Hardware Type): 0x0001 表示该帧的实体层走的是以太网(Ethernet), 协议类型(Protocol Type): 0x0800 表示归属于因特网协议(IP), 硬件地址长度(HLen): 6 表示 MAC 地址长度为 6 bytes, 协议地址长度(PLen): 4 表示 IP 地址长度为 4 bytes (IPv4), 操作类型(Operation): 0x0001 表示 ARP 询问 0x0002 表示 ARP 回复。图 3.6 提供一个 ARP 询问(request)帧的范例。

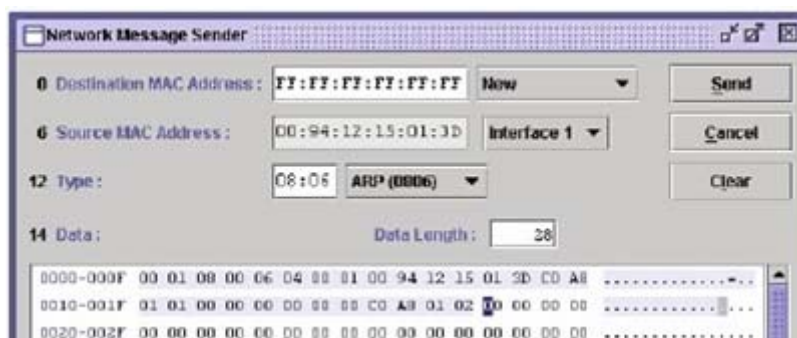


图 3.6

三、实验步骤

1、网络拓扑连线

- 1) 在 Hubox 上将网络连线如图 3.7 所示。本实验可让 6 台 ITS 连接为一个完整的实验网络。6 台 ITS 分别指定名称为 Member1 到 Member6。

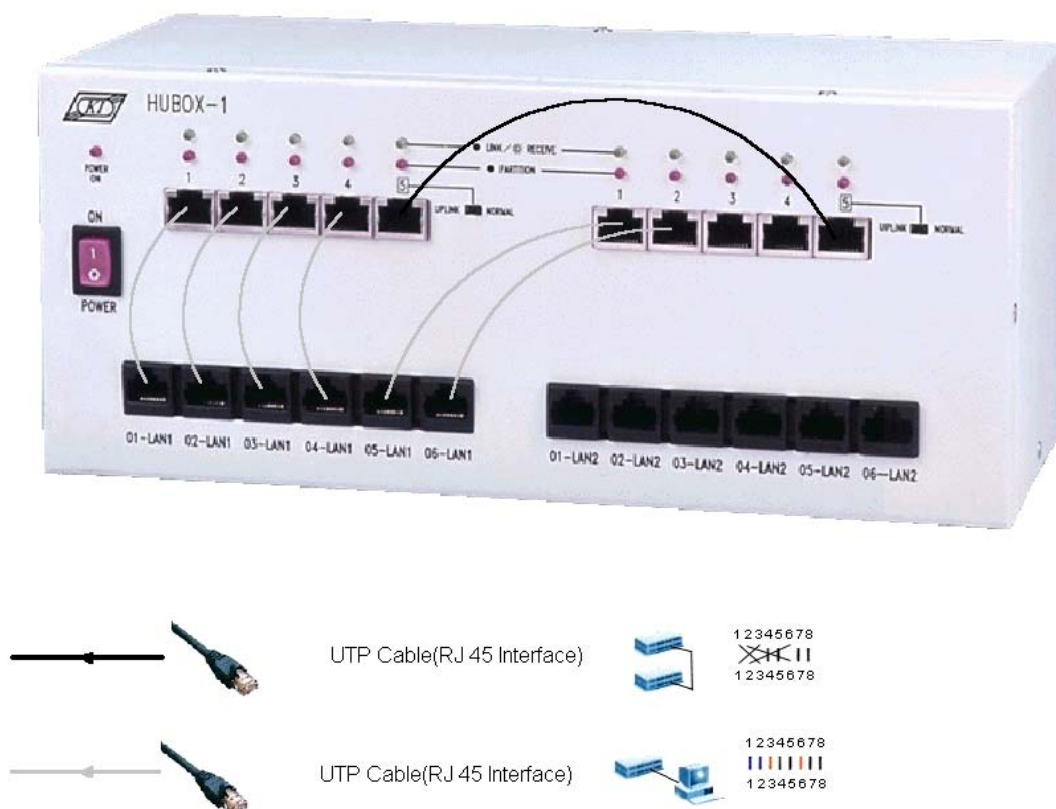


图 3.7

我们需要将六台 ITS 分成三个小组(两两配对):Member1 与 Member2 一组;Member3 与 Member4 一组; Member5 与 Member6 一组。

2、发送 ARP 查询帧

A. 状态设定

- 1) 执行 **XCLIENT.BAT**， 打开 KCodes Network Explorer.
- 2) 从 **Tool menu** 里找到 **Network Configuration**， 如图 3.8 所示， ITS 会打开一个可进行网络功能设定 (Network Configuration) 的窗口。
- 3) 参照网络拓扑 A (Single LAN)， 将自己的 IP 地址输入到 **IP Setting Address of Interface 1** 的对话框中， 并输入 “**255.255.255.0**” 到 **Subnet Mask** 文本框中。
- 4) 选择 **Host** 模式， 并单击 **Set & Close** 按钮。

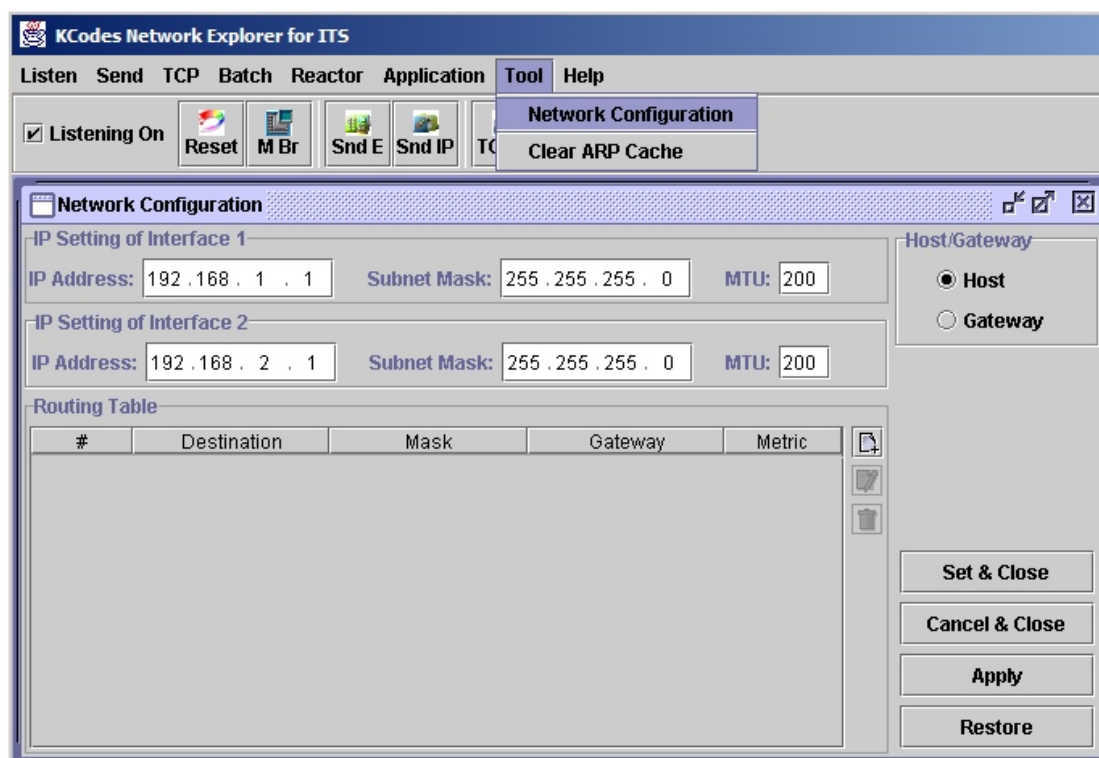


图 3.8

B. 发送 ARP 请求帧给搭档 (Partner)

- 5) 勾选 **Listening On**。
- 6) 从 **Listen menu** 中选择监听等级 (**Listening Level**)，将 **Interface Frames** 打勾。
- 7) 再从 **Listen menu** 中点击 **New Memorized Message Brower**。网络信息浏览器 (Network Message Browser) 将会被打开，开始及时监听整个网络上的信息传输。
- 8) 从 **Send menu** 里，单击 **Send Interface Frame**，打开 Network Message Sender 对话框。
- 9) 参考图 3.6，编辑一个标准的 ARP 请求帧，并传送给和你配对同组的搭档，然后在机会同一时间，你应该会收到一个从搭档那里反馈的标准 ARP 回复帧。

3、利用 MDDL 发送与接收 ARP

A. 调用 ArpServer.mddl 程序

- 10) 打开一个新的网络信息浏览器并将 **Listening On** 打勾。
- 11) 从 **Reactor menu** 中找到并运行 **MDDL Reactor Panel**，打开 MDDL 编辑平台。

- 12) 在 MDDL 编辑平台中, 单击 **Load** 按钮。调用 ArpServer.mddl 程序 (路径为 C:\XClient\Data\Mddl\Tutorial\Ex03\ArpServer.mddl), 最后按下 **Upld** 按钮, 载入到 ITS 中。
- 13) 载入程序后, 你可以再次发送一个 ARP 请求帧给你的搭档, 几乎同时你会收到一个从你搭档那边反馈回来的 ARP 回应帧 (MDDL 程序判断后发送的 ARP 回应帧)。
- 14) 如图 3.5 所示, 使用双反斜线“//”屏蔽 ArpServer.mddl 程序代码中的 **DISCARD_MESSAGE** 这行程序。然后再重新载入该程序至 ITS 中 (单击 Upld 按钮), 如果一切无误, 当所有 Member 在发送 ARP 请求帧给搭档后, 将会收到两个一模一样的 ARP 回应帧, 其中一个来自于 MDDL 程序, 另一来自原本系统的自动回复。

```

ETHER_IN_HANDLER ANY
{
    IF( S.ETHER_TYPE!=CNST_ETHER_TYPE_ARP ||
        S.ETHER_ARP_OP!=CNST_ETHER_ARP_OP_ARPREQ ||
        S.ETHER_ARP_IPADDRTRGT!=MYIP(INTERFACE()))
        RETURN;

    SEND_OUT_ETHER_FROM_INTERFACE INTERFACE() WITH_DATA
    {
        T
            = S
        ,
        T.ETHER_MACADDRDST      = S.ETHER_MACADDRSRC
        ,
        T.ETHER_MACADDRSRC      = MYMAC(INTERFACE())
        ,
        T.ETHER_ARP_OP           = CNST_ETHER_ARP_OP_ARPREPLY,
        T.ETHER_ARP_MACADDRSNDR = MYMAC(INTERFACE())
        ,
        T.ETHER_ARP_IPADDRSNDR  = MYIP(INTERFACE())
        ,
        T.ETHER_ARP_MACADDRTRGT = S.ETHER_ARP_MACADDRSNDR
        ,
        T.ETHER_ARP_IPADDRTRGT  = S.ETHER_ARP_IPADDRSNDR
        ,
        // DISCARD_MESSAGE;
    }
}

```




图 3.9

B. 调用 ArpRequest.mddl 程序

- 15) 按 **Reset** 按钮, 重置网络信息浏览器。
- 16) 打开 MDDL 程序编辑平台。
- 17) 在编辑界面中, 单击 **Load** 按钮, 调用 ArpRequest.mddl 程序 (路径为 C:

\XClient\Data\Mddl\Tutorial\Ex03\ArpRequest.mddl), 最后单击 **Upld** 按钮。

18) 所有 Members 可以自行发送 IP 给你的搭档, 如果系统还不知道你搭档的 MAC 地址, 就会发送 ARP 询问帧。

4、观察 PC 上的 ARP 列表

19) 见图 3.10, 我们在 PC 端使用 RS-232 线连接 ITS, 并从 ITS 端的 LAN1 接口用一条交叉线连接 PC 的网卡, 整个拓扑结构请参考网络拓扑 B。

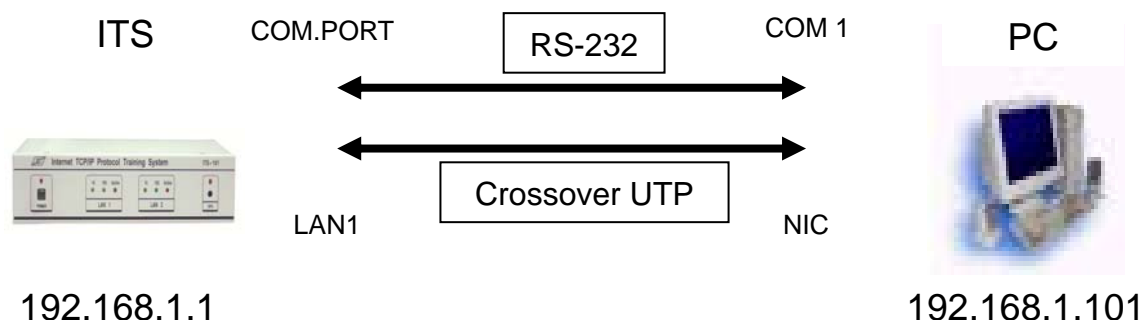


图 3.10

20) 确定所有的 Member 的操作系统是 Windows XP。

21) 从 XP 系统中找到“控制面板”, 再找到“网络连接”, 如图 3.11 所示。

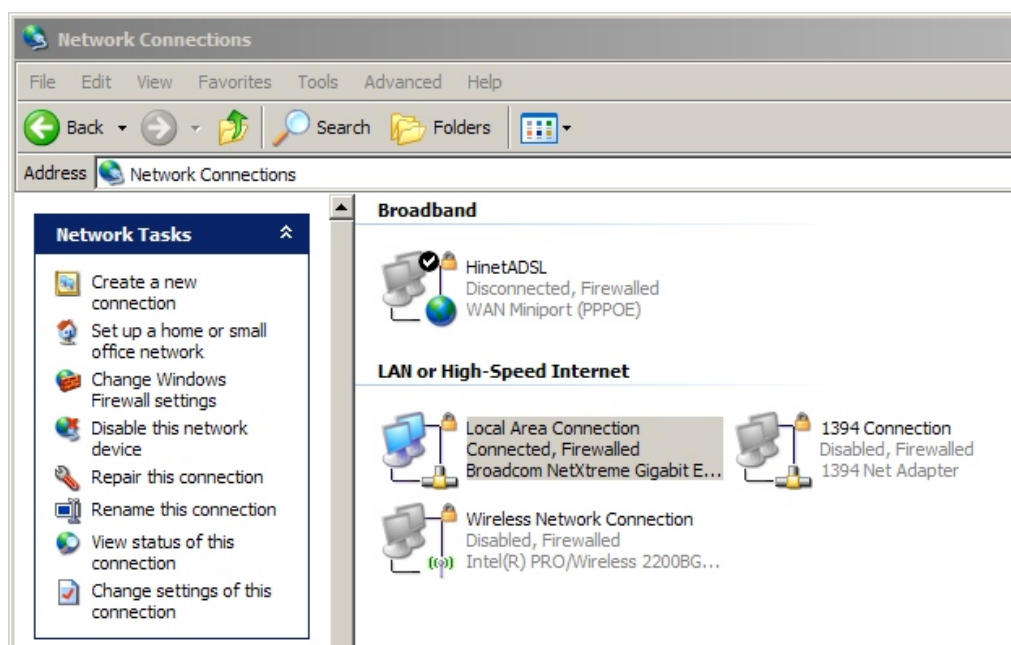


图 3.11

22) 打开其中的本地连接界面, 进入本地连接属性对话框, 如图 3.12 所示。

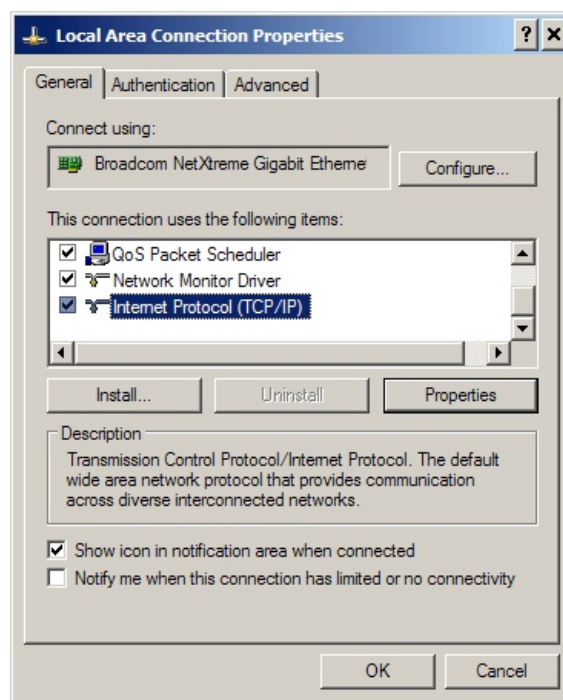


图 3.12

23) 在众多选项里找到 **Internet Protocol (TCP/IP)**，并点击属性按钮。此时，将会打开一个视窗界面如图 3.13 所示。选择“使用下面的 IP 地址”自定义 IP 地址和子网掩码。在本范例中，我们可以输入“**192.168.1.101**”到 IP 地址栏，输入“**255.255.255.0**”到子网掩码栏，设定完成后点击确定按钮。

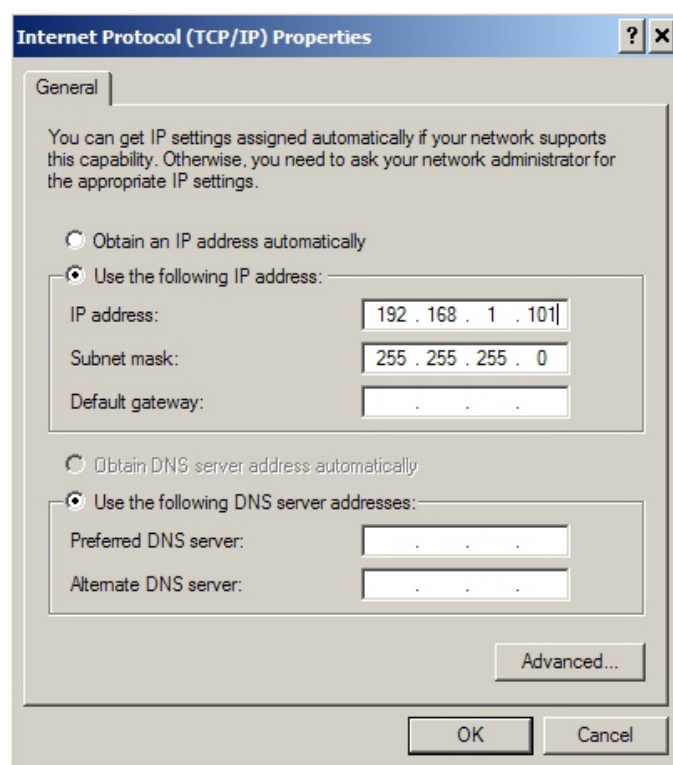
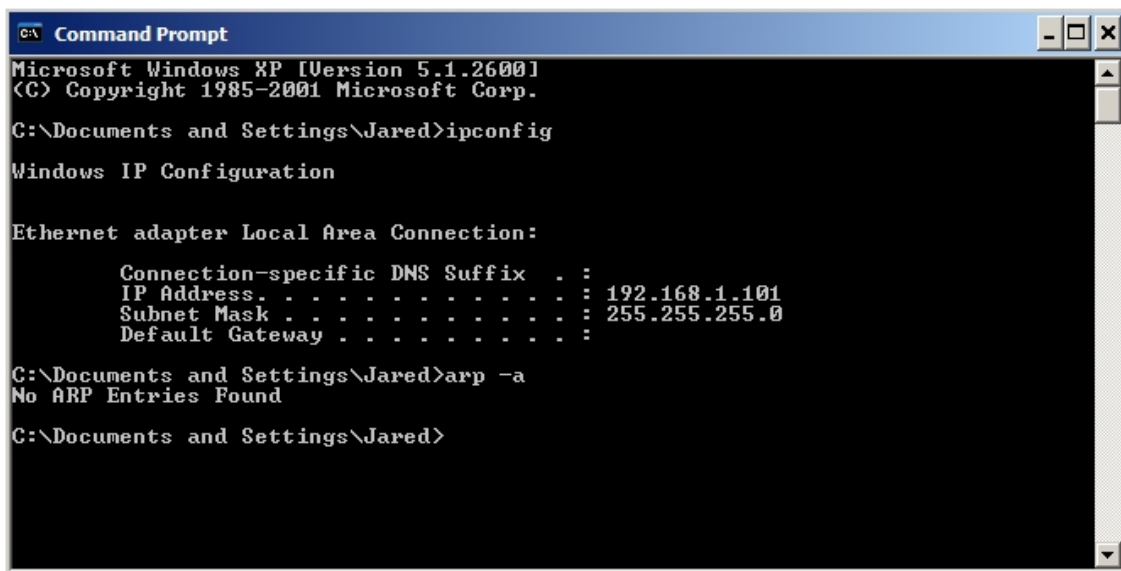


图 3.13

24) 再从 XP 系统的附件中, 运行 **Command Prompt**. (**All Programs >> Accessories >> Command Prompt**)。然后输入指令 **ipconfig**. 我们将会看见 PC 上的 windows IP 设定值。如图 3.14 所示, 我们再输入指令 **arp -a**, 将可以看见 PC 中的 ARP 列表, 目前是空的。



```

C:\> Command Prompt
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Jared>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    IP Address . . . . . : 192.168.1.101
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 

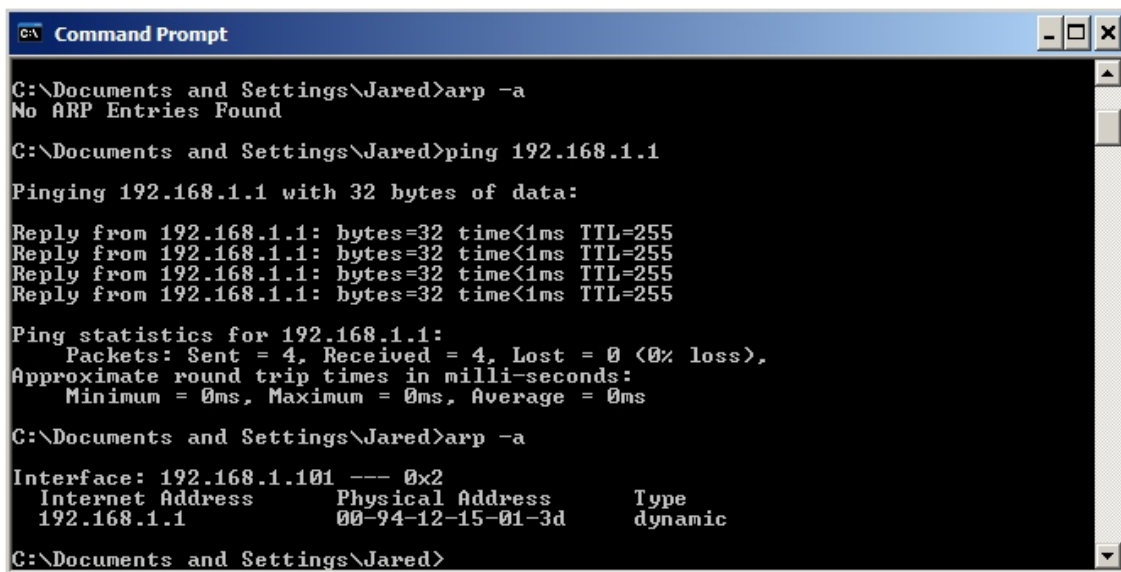
C:\Documents and Settings\Jared>arp -a
No ARP Entries Found

C:\Documents and Settings\Jared>
  
```

图 3.14

25) 进一步我们再用另一个指令 **ping <your ITS IP address>**, 运行后每一位 Members 都会收到 4 次成功回应帧, 如图 3.15 所示。

26) 再重新输入指令 **arp -a**, 我们可以看到 ARP 列表里, 已经多加了一项每个人自己的 ITS IP 地址。



```

C:\> Command Prompt

C:\Documents and Settings\Jared>arp -a
No ARP Entries Found

C:\Documents and Settings\Jared>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Documents and Settings\Jared>arp -a

Interface: 192.168.1.101 --- 0x2
    Internet Address      Physical Address      Type
    192.168.1.1           00-94-12-15-01-3d    dynamic

C:\Documents and Settings\Jared>
  
```


图 3.15

四、实验讨论

- 1、在实验步骤 15 中，当我们加载 ArpServer.mddl 并且将 DISCARD_MESSAGE 这行程序 disable 后，如何判断哪一个 ARP 回复封包是来自于 MDDL 程序？

提示：

```
T.ETHER_MACADDRDST= S.ETHER_MACADDRSRC,
T.ETHER_MACADDRSRC= MYMAC(INTERFACE()),
T.ETHER_TYPE= {0xAA,0xAA},
T.ETHER_ARP_OP= CNST_ETHER_ARP_OP_ARPREPLY,
T.ETHER_ARP_MACADDRSND = MYMAC(INTERFACE()),
```



- 2、请讨论下列的程序所含的功能，为什么我们需要它？

```
TIMER_WITH_PERIOD 200
{
    FOR_EVERY_ELEMENT_IN_POOL 20
    {
        PE[21, 22] -= 200;
        IF( PE[21, 22] > 0 )
        {
            CONTINUE;
        }
        ELSE
        {
            REMOVE_CURRENT_POOL_ELEMENT;
        }
    }
}
```

- 3、What is the dynamic type in your PC's ARP table?
- 4、How to show your ARP table in ITS?

提示：

```
GENERATE_USER_MSG WITH_DATA
{
    TARGET = "Hi I am here"
}
```

REACTOR PROGRAMS**1、ArpRequest.mddl**

```

IP_OUT_HANDLER

{

    IF(ISMYIP(S.IP_ADDRDST) || S.IP_ADDRDST==CNST_IP_ADDR_BROADCAST)

        RETURN;

    IF      (S.IP_ADDRDST.[0,2] == MYIPADDR(1).[0,2])
    {
        VAR1[0,3]          = S.IP_ADDRDST;
        VAR2[0]             = 1;
        S.IP_ADDRSRC        = MYIP(1);
    }
    ELSE IF(S.IP_ADDRDST.[0,2] == MYIPADDR(2).[0,2])
    {
        VAR1[0,3]          = S.IP_ADDRDST;
        VAR2[0]             = 2;
        S.IP_ADDRSRC        = MYIP(2);
    }
    ELSE
    {
        VAR1[0,3]          = { 192, 168, 0, 201 };    // Default gateway.
        VAR2[0]             = 1;                      // Interface corresponding to the default gateway.
        S.IP_ADDRSRC        = MYIP(1);
    }

    S.IP_LEN                = LENGTH(S)                ;
    S.IP_HEADERCHKSUM = {0, 0}                        ;
    S.IP_HEADERCHKSUM = CHECKSUM(S.IP_HEADER)          ;

```

```

LOOK_FOR_ONE_ELEMENT_IN_POOL 20 WITH_CONDITION
(VAR1[0,3]==PE[0,3])
{
    SEND_OUT_ETHER_FROM_INTERFACE PE[20,20] WITH_DATA
    {
        T.ETHER_MACADDRDST      = PE[4,9]                ,
        T.ETHER_MACADDRSRC      = MYMAC(PE[20,20])        ,
        T.ETHER_TYPE             = CNST_ETHER_TYPE_IP      ,
        T.ETHER_DATA             = S
    }

    PE[21, 22]                  = 20000;

    DISCARD_MESSAGE;
}
ELSE
{
    ADD_TO_POOL 2 WITH_LIFETIME 20000 WITH_DATA
    {
        T.[0,]                  = VAR1[0, 3],
        T.[LENGTH(T),]          = S
    }

    SEND_OUT_ETHER_FROM_INTERFACE VAR2[0] WITH_DATA
    {
        T.ETHER_MACADDRDST = CNST_MACADDR_BROADCAST      ,
        T.ETHER_MACADDRSRC = MYMAC(VAR2[0])              ,
        T.ETHER_TYPE       = CNST_ETHER_TYPE_ARP         ,
        T.ETHER_ARP_HARDTYPE = CNST_ETHER_ARP_HARDTYPE_ETHER ,
        T.ETHER_ARP_PROTOTYPE = CNST_ETHER_ARP_PROTOTYPE_IP ,
        T.ETHER_ARP_HARDSIZE = CNST_ETHER_ARP_HARDSIZE_ETHER ,

```

```

T.ETHER_ARP_PROTOSIZE = CNST_ETHER_ARP_PROTOSIZE_IP      ,
T.ETHER_ARP_OP        = CNST_ETHER_ARP_OP_ARPREQ         ,
T.ETHER_ARP_MACADDRSNDR = MYMAC(VAR2[0])                 ,
T.ETHER_ARP_IPADDRSNDR  = MYIP(VAR2[0])                   ,
T.ETHER_ARP_MACADDRTRGT = {0x00, 0x00, 0x00, 0x00, 0x00, 0x00},
T.ETHER_ARP_IPADDRTRGT  =
VAR1[0,3]              ,
T.[LENGTH(T),]         = "test"
}
DISCARD_MESSAGE;
}
}

```

```
ETHER_IN_HANDLER ANY
```

```

{
  IF( S.ETHER_TYPE==CNST_ETHER_TYPE_ARP &&
      S.ETHER_ARP_OP==CNST_ETHER_ARP_OP_ARPREPLY &&
      ISMYMAC(S.ETHER_MACADDRDST))
  {
    ADD_TO_POOL 20 WITH_DATA
    {
      T[0,3]          = S.ETHER_ARP_IPADDRSNDR ,
      T[4,9]          = S.ETHER_ARP_MACADDRSNDR,
      T[10,13]        = S.ETHER_ARP_IPADDRTRGT ,
      T[14,19]        = S.ETHER_ARP_MACADDRTRGT,
      T[20,20]        = INTERFACE()            ,
      T[21,22]        = 20000
    }
  }
}

```

```
FOR_EVERY_ELEMENT_IN_POOL 2 WITH_CONDITION
```

```

(S.ETHER_ARP_IPADDRSNDR==PE[0, 3])

{

    SEND_OUT_ETHER_FROM_INTERFACE INTERFACE() WITH_DATA

    {

        T.ETHER_MACADDRDST  = .ETHER_ARP_MACADDRSNDR      ,
        T.ETHER_MACADDRSRC   = YMAC(INTERFACE())           ,
        T.ETHER_TYPE         = NST_ETHER_TYPE_IP           ,
        T.ETHER_DATA         = PE[4,]

    }

    REMOVE_CURRENT_POOL_ELEMENT;

}

DISCARD_MESSAGE;

}

ELSE IF( S.ETHER_TYPE==CNST_ETHER_TYPE_ARP &&

        S.ETHER_ARP_OP==CNST_ETHER_ARP_OP_ARPREQ)

{

    LOOK_FOR_ONE_ELEMENT_IN_POOL 20 WITH_CONDITION

    (S.ETHER_ARP_IPADDRSNDR==PE[0, 3])

    {

        PE[4,9]              = S.ETHER_ARP_MACADDRSNDR;
        PE[21, 22]           = 20000;

    }

    ELSE

    {

        ADD_TO_POOL 20 WITH_DATA

        {

            T[0,3]            = S.ETHER_ARP_IPADDRSNDR ,
            T[4,9]            = S.ETHER_ARP_MACADDRSNDR,

```

```

        T[10,13]          = S.ETHER_ARP_IPADDRTRGT ,
        T[14,19]          = S.ETHER_ARP_MACADDRTRGT,
        T[20,20]          = INTERFACE()           ,
        T[21,22]          = 20000
    }
}
}
ELSE IF( S.ETHER_TYPE==CNST_ETHER_TYPE_IP )
{
    LOOK_FOR_ONE_ELEMENT_IN_POOL 20 WITH_CONDITION
    (S.ETHER_DATA.IP_ADDRSRC==PE[0, 3])
    {
        PE[4,9]            = S.ETHER_MACADDRSRC;
        PE[21, 22]         = 20000;
    }
    ELSE
    {
        ADD_TO_POOL 20 WITH_DATA
        {
            T[0,3]          = S.ETHER_DATA.IP_ADDRSRC,
            T[4,9]          = S.ETHER_MACADDRSRC      ,
            T[20,20]        = INTERFACE()           ,
            T[21,22]        = 20000
        }
    }
}
}

TIMER_WITH_PERIOD 200
{

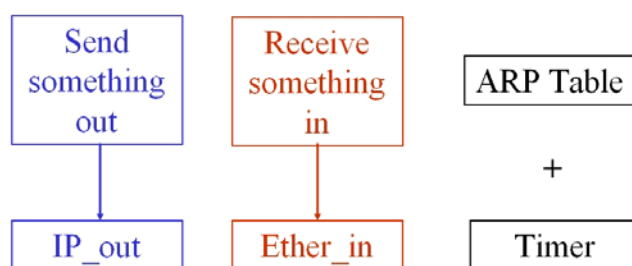
```



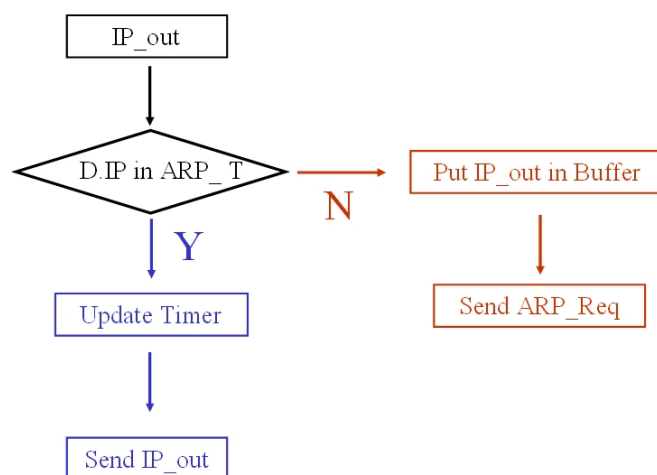
```
FOR_EVERY_ELEMENT_IN_POOL 20
{
    PE[21, 22] -= 200;

    IF( PE[21, 22] > 0 )
    {
        CONTINUE;
    }
    ELSE
    {
        REMOVE_CURRENT_POOL_ELEMENT;
    }
}
}
```

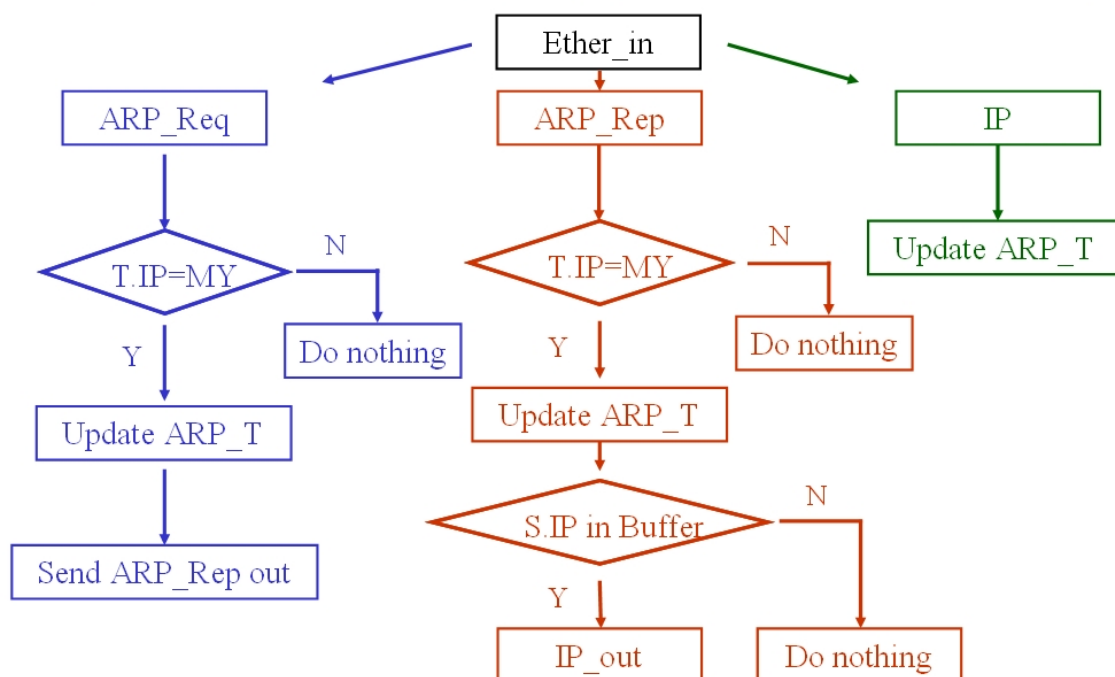
提示 : ARP Mechanism



IP_OUT_HANDLER



ETHER_IN_HANDLER



2、ArpServer.mddl

```

ETHER_IN_HANDLER ANY
{
    IF( S.ETHER_TYPE!=CNST_ETHER_TYPE_ARP ||
        S.ETHER_ARP_OP!=CNST_ETHER_ARP_OP_ARPREQ ||
        S.ETHER_ARP_IPADDRTRGT!=MYIP(INTERFACE()))
        RETURN;

    SEND_OUT_ETHER_FROM_INTERFACE INTERFACE() WITH_DATA
    {
        T
            = S
            ,
        T.ETHER_MACADDRDST      = S.ETHER_MACADDRSRC
            ,
        T.ETHER_MACADDRSRC      = MYMAC(INTERFACE())
            ,
        T.ETHER_ARP_OP          = CNST_ETHER_ARP_OP_ARPREPLY,
        T.ETHER_ARP_MACADDRSNDR = MYMAC(INTERFACE())
            ,
        T.ETHER_ARP_IPADDRSNDR  = MYIP(INTERFACE())
            ,
        T.ETHER_ARP_MACADDRTRGT = S.ETHER_ARP_MACADDRSNDR
            ,
        T.ETHER_ARP_IPADDRTRGT  = S.ETHER_ARP_IPADDRSNDR
    }
    DISCARD_MESSAGE ;
}

```