1. **课程设计的环境配置**

本课程设计的目的是设计一个系统，它能够实现对网络上的ARP数据包的解析，从而熟悉ARP数据包的结构，对ARP协议有更好的理解和认识。了解并熟悉Microsoft Visual C++ 6.0开发环境的使用，在该软件平台上进行工程的建立和程序开发，理解和掌握C++语言的程序设计和针对一些常用的网络协议的编程。

**二、 课程设计过程**

1、本程序利用Winpcap驱动为Win32应用程序提供的访问网络底层的能力编写解析ARP数据包的程序。设计步骤如下：

* 1. 输出网卡设备列表。
  2. 选择网卡并打开。
  3. 捕获数据包时，可能需要设置过滤器。
  4. 捕获数据包或者发送数据包。来源：(<http://blog.sina.com.cn/s/blog_4ca1a910010009hd.html>) - 解析ARP数据包（二）——说明书\_星期\_新浪博客

2、在程序设计过程中需要注意网络-主机字节顺序的转化。由于不同计算机系统所采用的数据表示方式不同，对于2B或4B的数据，有的采用低字节地址存放数据的高权值位，而有的却以低地址字节存放数据的低权位值，在网络的数据传输中，我们应该统一表示，所以我们在捕获数据包后，应将数据包头部的表示长度或者类型的数据转换成本地机的表达形式。可以利用函数ntohs（）将网络字节序转换为主机字节序。

3、选择网卡并打开，判断网卡是否可用。

算法流程如下：

1. 取得当前网络设备列表。
2. 选择Ethernet网卡打开，注意判断所选网卡是否为实际存在的可用网卡。
3. 设置过滤器，此处的过滤器正则表达式为“arp”或者“ether proto\\arp”。
4. 捕获数据包并进行处理。（包括输出各IP地址，物理地址，操作类型以及时间）。由于要记录日志文件，为了便于输出流参数，采用pcap\_next\_ex（）函数。

程序流程图如下图所示：

开 始

获取网卡列表

选取Ethernet网卡

打开网卡（混杂模式）

编译设置过滤器

捕获ARP包并将其相应内容输出

图3 程序流程图

**三、 课程设计小结**

通过这次课程设计，我深刻体会到了ARP协议的基本功能与工作方式。在以太网中，一个主机要和另一个主机进行直接通信，必须要知道目标主机的MAC地址。但这个目标MAC地址是如何获得的呢？它就是通过地址解析协议获得的。所谓“地址解析”就是主机在发送帧前将目标IP地址转换成目标MAC地址的过程。ARP协议的基本功能就是通过目标设备的IP地址，查询目标设备的MAC地址，以保证通信的顺利进行。

在这次课程设计过程中，通过查找相关书籍和资料，也使得自己增长了不少知识。从这次课设中，我充分认识到自己相关网络知识的缺乏，虽然也用了很长时间但设计方案还是缺乏一定的可行性，这次课程设计过程中我也认真阅读了相关书籍，对局域网有了一定的了解，所以说这次课设令我收获颇丰。