

## 实验内容

- 实验内容：以太网协议设计与实现
  - 以太网协议设计
  - 以太网协议实现

## 以太网协议设计与实现

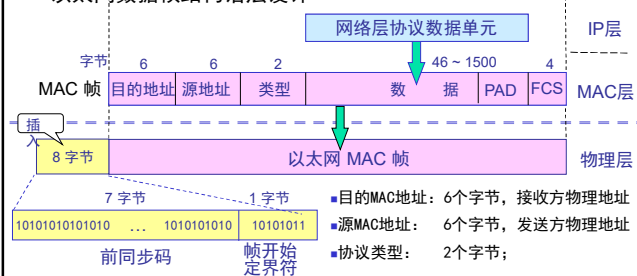
实验内容：以太网协议设计与实现

实验功能要求：

- (1) 发送端：数据帧构造，发送；
- (2) 接收端：数据帧接收，解析；
- (3) 完成一个文件的传输；

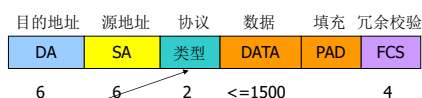
## 实验内容:协议设计与实现

- 一、以太网协议设计
- 以太网数据帧结构语法设计



## 实验内容:协议设计与实现

- 一、以太网协议设计
- 以太网数据帧结构语法设计

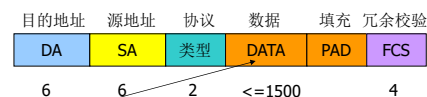


协议类型字段：表示上层协议类型，接收方利用该字段将MAC帧数据（DATA）交付给上层该协议。

■ 0x0800：表示上层为IP协议；0x8137：表示IPX协议；0x0806时，表示ARP协议；

## 实验内容:协议设计与实现

- 一、以太网协议设计
- 以太网数据帧结构语法设计

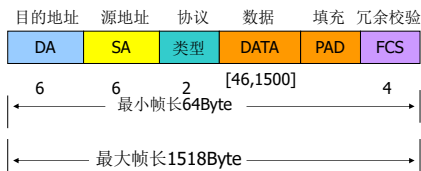


DATA字段：表示要传送的网络层协议数据单元，网络层协议数据单元应是字节倍数，最大数据长度为1500个字节，最小为46个字节？。

## 实验内容:协议设计与实现

### 一、以太网协议设计

#### 以太网数据帧结构语法设计



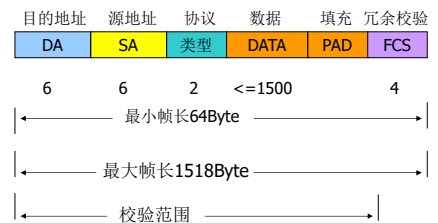
#### PAD（填充）：

- (1) 数据帧要求有MTU（46B），最小帧长为64个字节:(18+PDU)
- (2) 如果实际的PDU数据长度小于46个字节，必须在PAD字段上填充若干字节的0，使PDU和PAD字段的总长度不小于46个字节；否则，接收节点会把超短帧作为“帧碎片”滤掉，不予接收。

## 实验内容:协议设计与实现

### 一、以太网协议设计

#### 以太网数据帧结构语法设计



FCS（帧校验序列）：采用32位CRC校验。  
生成多项式：G(X)-32，CRC-32

## 实验内容:协议设计与实现

### 二、以太网协议实现

#### 开发接口

- WINDOWS系统提供的WINPCAP接口（或者LINUX 下libpcap接口）
- 下载地址<http://www.winpcap.org/>;
- <http://www.winpcap.org/devel.htm> 下载WinPcap developer's pack 包解压，里面有docs, include, lib, example等文件夹；

## 实验内容:协议设计与实现

### 二、以太网协议实现

#### 1、发送方流程

- (1) 定义数据帧数据结构；
- (2) 从文件中读取数据（46-1500字节）；
- (3) 计算CRC校验码；
- (4) 封装以太网数据帧；
- (5) 读取本地网卡列表；
- (7) 选择本次通信网卡序号（0,1,2?）；
- (8) 初始化本次通信网卡；
- (9) 发送数据帧，返回（2），直到文件数据发送完。

```
//ethernet header
struct ethernet_header
{
    u_int8_t dest_mac[6];
    u_int8_t src_mac[6];
    u_int16_t ethernet_type;
};
```

## 实验内容:协议设计与实现

### 二、以太网协议实现

#### 2、接收方流程

- (1) 读取本地网卡列表；
- (2) 选择本次通信网卡序号（0,1,2?）；
- (3) 初始化本次通信网卡；
- (4) 接收数据帧；
- (5) 数据帧正确性检查：
  - 1) 目的地址匹配或者二层广播地址；2) 是否碎片帧（小于64B）；3) CRC校验码验证；
- (6) 将帧首部及尾部各个字段解析并利用十六进制屏幕打印；
- (7) 依据协议类型（0x0800），将帧数据部分写入文件，返回（4），直到文件接收完成。

## 检查点

- 两人一组，自由组队；
- 分别显示发送方和接收方数据帧内容（首部+数据+尾部），二者需保持一致。