

# 计算机网络第三章

---

@author:yelihu

@time:2019年09月17日

@email:[87160265@qq.com](mailto:87160265@qq.com)

## 数据链路层

---

数据链路层属于计网的低层，数据链路层主要使用的信道有如下两种

- 点对点信道
- 广播信道

学习点对点信道中的PPP协议以及利用广播信道实现的局域网之前，首先搞懂什么是数据链路层的三个基本问题。

---

## 三个基本问题

数据链路层的协议解决的最基本的三个问题分别如下：

- 将数据封装成帧：帧的**首部**和**尾部**用于**封装数据报**，来确定一个**单位**的帧。帧里面包含一个数据字段和若干首部字段。控制字符放在帧的最前面，表示首部的开始。另一个控制字符表示帧的结束。我们将128个ASCII码限定出来33个限定字符，其中的SOH和EOT分别作为控制字符中帧的帧定界符的名称。
- 透明传输：为了防止数据链路层在数据报中找到类似EOT之类的控制字符，对帧的定界造成干扰，我们可以加入字节填充的方法，在数据报里的类似控制消息的字符之前填充一个"转义字符"十六进制为1B，名称为ESC。
- 差错监测：传输过程可能会出现差错，相对于信噪比的概念在差错监测中有个误码率的概念。传送十个"01"出现了一个本该是0的，接收端却收到了1，那么这1个差错除以10，误码率就是

$$\frac{1}{10}$$

我们一般使用**循环冗余检测 CRC**的方法，来实现差错检测。接收端通过 **CRC**的规则来校验**帧监测序列 FCS**。实现了差错监测，实现了一定范围内的正确。

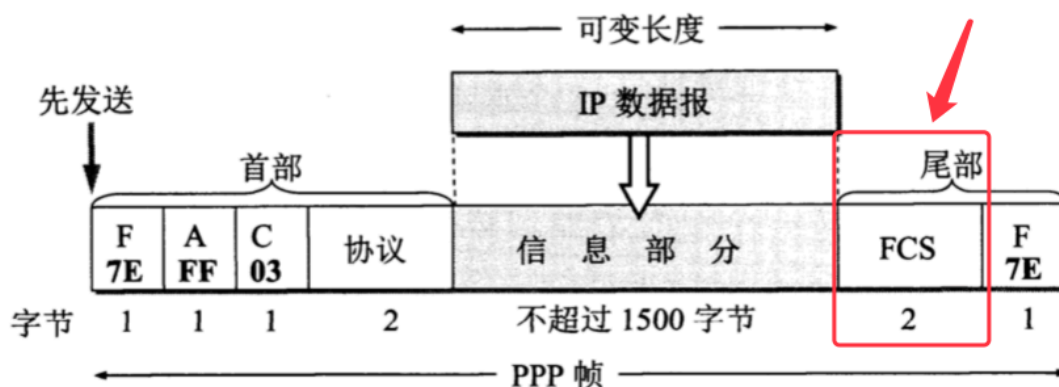
---

## CRC的描述

首先要知道，**循环冗余检测 CRC**这种方法，校验错误的主体是**冗余码**。

什么是冗余码？

冗余码就是为了在数据帧上，为了实现差错检测而附加在数据后面的一小串二进制数字，通常也叫做帧检验序列 FCS。



CRC 校验的计算规则如下：

首先发送端将数据划分为组，若一组K个bit，例如：M = 101001，CRC想要做到的就是在M数字后面加上一个N位的冗余码。最后发送出去的数据和冗余码一共为

$$K + N (\text{位})$$

这N为冗余码，我们可以这样求

首先强调一点：模2运算是进位和借位的，详情见教科书P74注①

1. 将这个数先左移N位，也就是乘以

$$2^N$$

那么原来的M后面也就是加上了N个0，拓展成了M+N这么长。将这个新的长度的M称为M'

2. 按照事先约定好的一个数P，这个P的位数为N+1，我们将M'

$$\frac{M'}{P}$$

就能分别得到一个商Q和一个余数R，其中R的位数为N位(为什么呢？比如当M= 1001，N=3，M' = 1001000，P = 1000，这时候N = 3，N+1 = 4) 如下运算①

$$\frac{M' (M \text{ 加上 } N \text{ 位左移产生的 } 0)}{P} = \frac{1001000}{1101(N+1 \text{ 位})} = 1110 (\text{商}) + 011 (\text{余数, } n \text{ 位})$$

3. 上面就得到了这个除法产生的N位余数，我们将这个余数作为冗余码，拼接在数据M(即1001后面)。这个商在此次运算中没什么用。
4. 最后发出去的数据+冗余码即为：1001011
5. 接收端将此次接收到的数据都除以约定好的P (1101)，如下运算②

$$\frac{1001111}{1101} = 1111 \text{ 余 } 0 (\text{整除})$$

上面的结果显示，经过模2运算校验之后，是整除的(余数为0)，所以校验正确，说明数据在链路中传播的时候，没有收到干扰造成某些位数0变1/1变0。如果不能整除，说明某些位在传播过程中变化。

上面计算的详细过程如下

$$\begin{array}{r} \phantom{(p)} 1101 \sqrt{1001 \underline{000}} \rightarrow \text{左移3位} \\ \phantom{(p)} \underline{1101} \\ \phantom{(p)} 1000 \\ \phantom{(p)} \underline{1101} \\ \phantom{(p)} 1010 \\ \phantom{(p)} \underline{1101} \\ \phantom{(p)} 1110 \\ \phantom{(p)} \underline{1101} \\ \phantom{(p)} 011 \text{ (余数)} \end{array}$$

运算①

(P) 1101  $\sqrt{1001011}$

1101

1000

1101

1011

1101

1101

0

余数

链路: (link), 一个点到另外一个点的物理线段, 中间没有任何节点

数据链路：除了link，要需要一个通讯协议来控制数据传输。实现协议的硬件和软件在一起组成了数据链路。

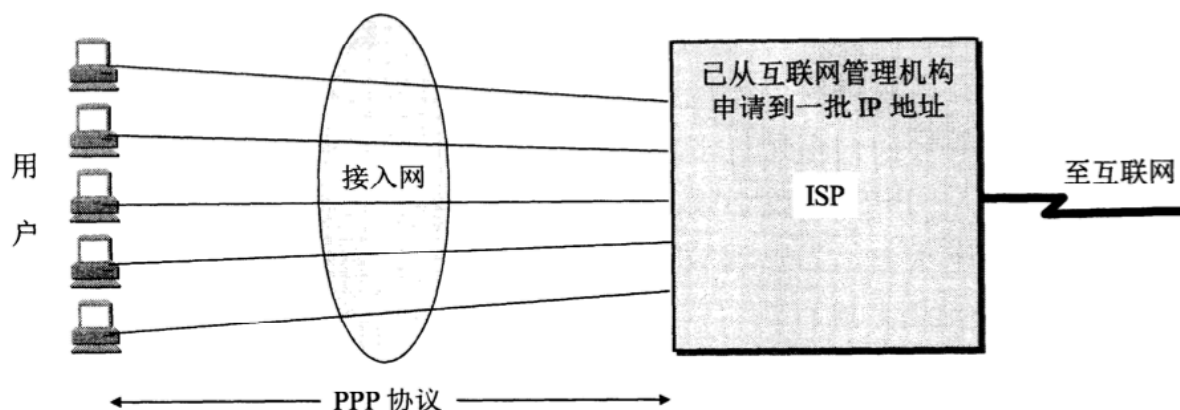
帧：

数据链路层上传输的数据单位是帧

## 点对点协议PPP

全程Point-to-Point Protocol点对点协议，目前使用最广泛的数据链路层协议。

### 特点



端在访问网络层获取IP的时候，就使用PPP协议

特点如下

- 简单
- 成帧
- 透明
- 多种网络层协议支持：一个链路，多个网络协议支持
- 多类型链路支持：多类型的链路支持
- 差错监测：接收端丢弃有差错的帧
- 检测连接状态：链路是否正常？
- 最大传送单元：除去首部和尾部，中间的数据的最大大小。太小效率低，太大容错率低，一旦一个bit出错，整个帧全部出错。
- 网络层地址协商机制
- 数据压缩协商

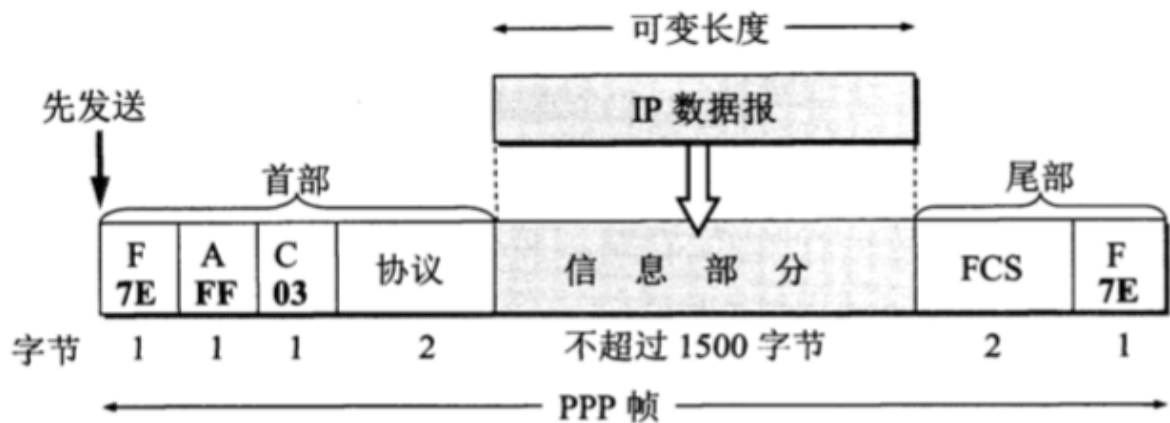
---

### 组成

- 方法：IP数据报（针对网络层是IP协议）封装到串行的链路上
- 链路控制协议
- 网络控制协议

帧的格式

帧的格式如下



图上说明，左边首部首先发送，顺序如下

- 第一个byte是0x7E，固定的填写这个值。F部分是flag标志位，标志这个帧的开始
- 第二个是FF和第二个是03，分别是地址字段和控制字段，按照教科书的意思，这里两个字段没意义。
- 第四个是协议，不同网络层协议填写不同内容的字段，比如当发送的是**IP数据报**的时候，这里的内容是0x0021
- 中间这部分就是需要传输的数据，一般不超过1500byte。
- 尾部第一块称为fcs，上文提及，用于差错检测
- 最后一部分结束位，7E，和开头一样。

A部分称为地址字段，这里的数据为0xFF，没有实际意义。因为PPP是点对点的协议。

如何实现透明传输的问题，比特填充和零比特填充？

### 透明传输的解决

#### 字节填充

将每一个出现在数据中的0x7E，转变为0x7D和0x5E，如果

#### 零比特填充

每出现五个连续的1就填充一个0，来防止被误认为F，这个0在接收端会被删除

010**01111110**001010  
会被误认为是标志字段F

0 1 0 0 1 1 1 1 1 0 1 0 0 0 1 0 1 0

↑  
发送端填入 0 比特

0 1 0 0 1 1 1 1 1 0 1 0 0 0 1 0 1 0

接收端删除填入的 0 比特

## PPP协议总结

- 点对点
- PPP最基本的步骤
- 数据链路层需要处理的三个问题和PPP协议如何处理这三个问题

---

## 广播信道的数据链路层

---

### 局域网

#### 局域网定义

...

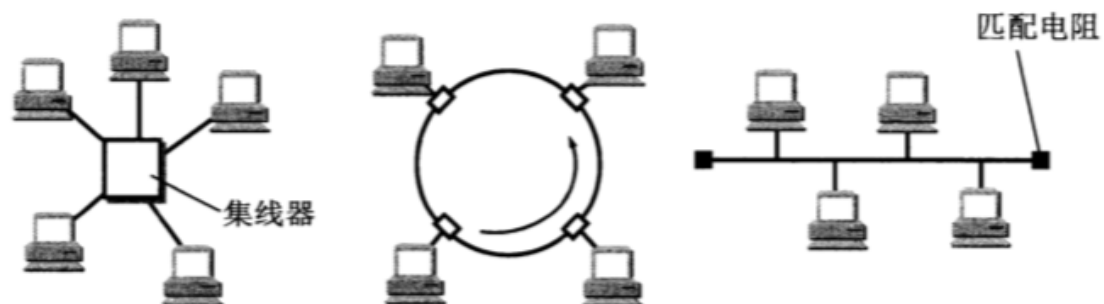
#### 局域网特点

##### 局域网特点如下

- 广播特点：比如教师机播放的PPT可以给学生机播放
- 设备位置灵活调整
- 提高了系统的：
  - 可靠性
  - 可用性
  - 生存性

---

#### 局域网拓扑结构



- 
- 星型：单点故障问题严重，如果中间的集线器HUB坏了...
  - 环形
  - 总线型
- 

#### 媒体共享技术



如何共享信道？如下两种类型

### 静态划分信道

- 频分复用
- 时分复用
- 波分复用
- 码分复用

### 动态划分信道

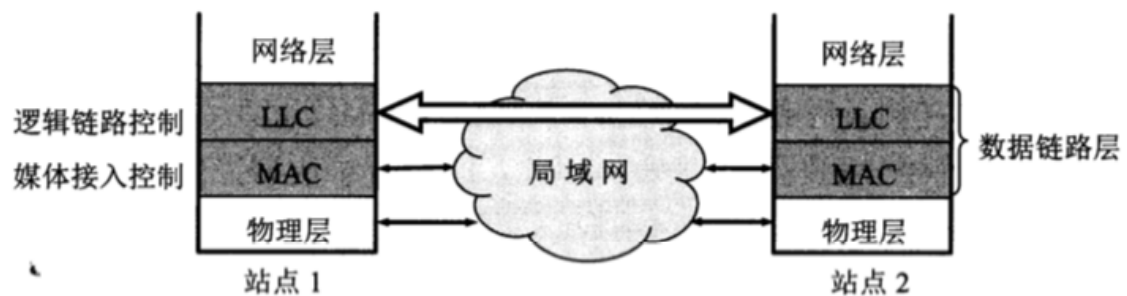
- 随机接入
- 受控接入：多点线路探寻、轮询

---

## 以太网

十九世纪末有人认为光通过"以太"来传播

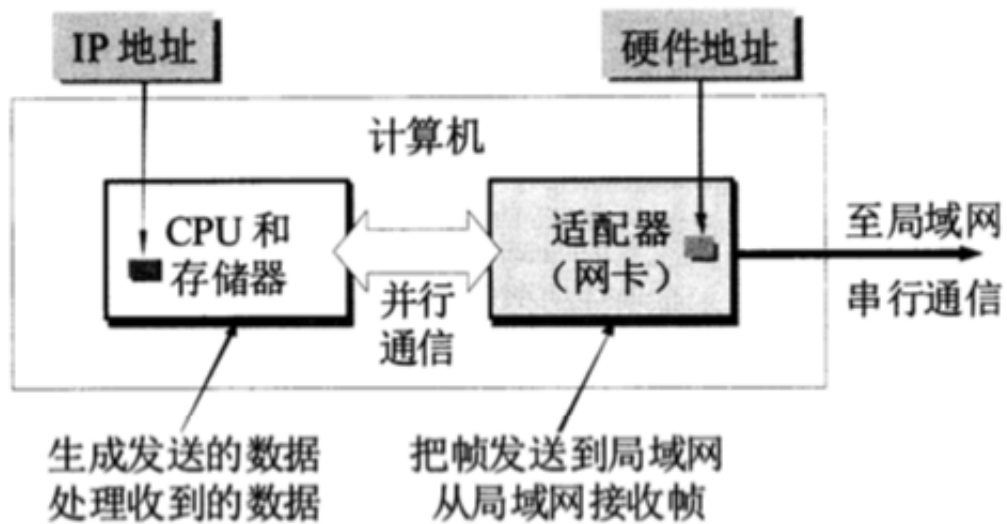
以太网的标准：DIX和IEEE 802.3：将链路层分为两个子层：逻辑链路控制LLC和媒体接入控制MAC(在Win10里面显示的是物理地址，该地址分隔符为"-")



---

### 适配器的作用

每一网卡在出厂的时候都有一个mac地址分配



并行通信的位宽由电脑是几位的机子决定，32位或者64位

## CSMA/CD协议

最初的以太网都是许多计算机连接在一个bus总线上



一个信号可以发送给所有计算机

如何实现指定机子接收？加上物理地址

## 缺点

所有机子都可以截获某一个由A机发往B机的数据。

