

# 数据链路层

---

数据链路层使用的信道主要由以下两种类型：

1. **点对点信道**。这种信道使用**一对一**的点对点通信方式。
2. **广播信道**。这种信道使用**一对多**的广播通信方式，因此过程比较复杂。广播信道上连接的主机很多，因此必须使用专门的共享信道协议来协调这些主机的数据发送。

本章最重要的内容是：

1. 数据链路层的点对点信道和广播信道的特点，PPP协议。
2. 数据链路层的三个基本问题：封装成帧，透明传输和差错检测。
3. 以太网MAC层的硬件地址。
4. 适配器、转发器、集线器、网桥、以太网交换机的作用以及使用场合。

## 使用点对点信道的数据链路层

---

### 数据链路和帧

所谓**链路**是从一个节点到相邻节点的一段物理线路，这段线路可以有线也可以无线，而中间没有其他的交换节点，这意味着一段链路只是一条路径的组成部分。

**数据链路**是一个概念，这是因为在传输数据的时候，除了物理设备以外，还需要一套通信协议来完成数据的传输，两者结合才能叫做数据链路。最常见的是**网络适配器**，它完成了数据链路层和物理层的功能。从另一个层面理解，数据链路是物理链路和逻辑链路的结合，**物理链路就是物理层设备，逻辑链路就是通信协议**。

**帧**是点对点信道的数据链路层的协议数据单元。在互联网中，网络层协议数据单元就是**IP数据报**。

### 三个基本问题

- 封装成帧。
  - 在存放在帧的实际数据之外添加一个帧首部和帧尾部，这样可以方便**帧的定界**。而为了方便帧的传输，定义了一个**MTU**以控制传输长度不至于长到降低传输成功率。帧的首尾部自然会有一些控制信息。
  - 如果发现由帧的开始，但是没有结束，这是一个不完整的帧，应当舍弃。
  - 如果使用ASCII码传输的时候，可以考虑使用ASCII中定义的开始(0x01, **EOH**)与结束(0x04, **EOT**)来进行帧定界。对于比较简单的也会使用这些默认设定。
- 透明传输。
  - 这里的**透明**代表**当一个实际存在的事物看起来却好像不存在一样**。
  - 发生条件是发现一个开始符以后遇到了一个多个结束符的情况，如果不做额外处理会导致提前结束接收，链路层协议会找到一个错误的边界。
  - 这里可以采用**字符填充**解决这个问题，在之前的内容中，可以加入ESC(0x1B)，此时如果需要发送0x1B，那么就发送两个0x1B即可，这样就完成了ESC的转义。
- 差错检测。这里就要提到**CRC校验**。具体方式可以查找对应的资料。
  - 这里需要指出的是如果在数据链路层仅仅使用CRC，那么只能做到对于帧的无差错接受，或者说能以**非常接近于1的概率认为传输过程中没有差错**。CRC**不能**对于帧与帧之间的关系进行评判。

- 帧与帧之间的关系除了正确顺序之外，还有**帧丢失**、**帧重复**或**帧失序**三种。过去OSI对于这类问题的探讨则是需要增加**帧编号**、**确认**和**重传**机制，即如果收到正确的帧就要向发送端发送确认消息，如果一段时间内没有发送则需要重发，直到有回复为止。
- 但是现在则是采取区别对待，因为网络环境的改善导致因链路质量不好而出错的概率大大降低。对于比较好的链路（比如有线传输），数据链路层不用**确认和重传**机制，如果数据链路层传输数据时出现了差错并需要改正，那么改正差错的任务交给上层协议完成；对于质量比较差的协议就需要确认和重传机制。

还有一个概念就是**误码率**，代表着一段时间内传输错误的bits所占所有bits的比率。

## 点对点协议PPP

鞭尸一下HDLC（笑）。这一部分课本讲的其实不好，很多内容都没有涉及到，这里推荐[这个链接](#)，

### PPP协议的特点

PPP协议在用户通过接入网介入到ISP中会使用，因为这是一个点对点的数据链路层协议。PPP协议在1992年，由IETF提出，在1994年变成了RFC 1661。

### PPP协议应满足的要求

IETF认为，PPP协议应该满足如下需求，这在RFC 1547中已经有定义了，主要有：

- 简单处理。处理方式应当尽可能简单，最复杂的部分在TCP中，IP协议也比较简单，因此这个协议也没有必要提供更多的功能。因此对于链路层的帧，不需要纠错，不需要序号，也不需要流量控制。
- 封装，就是帧定界符；透明性，同之前；多种网络层协议；多种类型的链路。
  - 注意PPPoE：PPP over Internet，1999年，RFC 2516，这个协议把PPP帧封装在以太网帧中。
- 差错检测；检测连接状态；最大传送单元；网络层地址写上；数据压缩协商。
  - PPP协议必须能够及时发现链路是否有问题，以及在链路正常之后能够重新发现。
  - PPP协议必须对于每一种类型的点对点链路设置MTU的标准默认值。
  - PPP协议必须提供一种机制是通信的两个网络层

PPP协议不支持多点线路，支支持全双工链路。

### PPP协议的组成

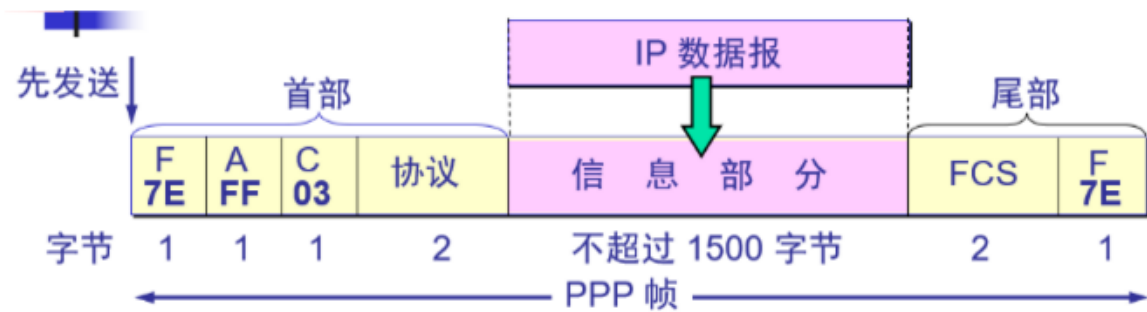
PPP协议分三个部分。

1. 一个将IP串行链路的方法。
2. 一个用来建立、配置和测试数据链路的**链路控制协议LCP**，在RFC 1661中定义了11种。
3. 一套**网络控制协议NCP**，其中每一个协议支持不同的网络层协议。

### PPP协议的帧格式

#### 各字段的意义

这个协议的内容一张图就能大致理解，后面就是简要介绍。



如果写入了数字，就是写死的内容，FCS的就是CRC的帧检验序列。写一部分有固定内容：0x0021为IP数据报，如果是0xC021，就是LCP数据，如果是0x8021就是网络层的控制数据。

## 字节填充

规则如下（RFC 1662）：

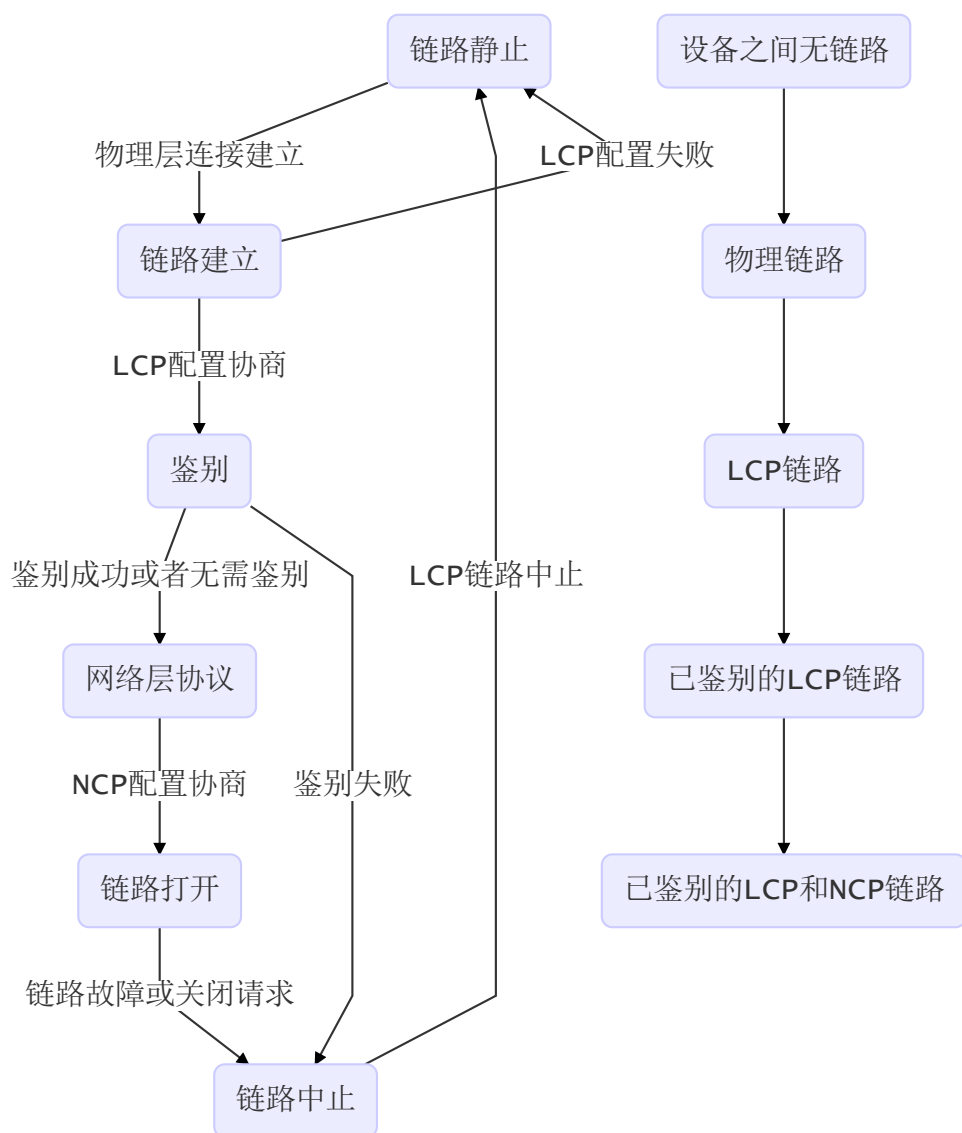
1. 定义转义符为0x7D，每一个出现0x7E的前面放一个转义符。
2. 如果出现了一个转义符，就重复一次。
3. 如果出现了控制字符，就在之前放一个转义符。

## 零比特填充

如果连续出现了五个1，那么在后面加一个0，这主要在同步传输中使用。

## PPP协议的工作状态

那么问题来了，PPP是怎么工作的？



其中会有一些重点需要强调的：

1. **链路静止**是起始与终止状态。
2. **链路建立**状态需要进行LCP协商，一般链路的一段可以发送以下几种响应中的一种。
  1. **配置确认帧(Configure-Ack)**，所有选项都接受。
  2. **配置否认帧(Configure-Nak)**，所有选项都理解但不能接受。
  3. **配置拒绝帧(Configure-Reject)**，选项有的无法识别或不能接受，需要协商。
  4. 配置选项包括链路上的最大帧长MTU，使用的**鉴别协议**，以及不使用PPP帧中的地址和控制字段（这两个字段的内容是固定的）。
3. LCP建立以后就会进入**鉴别状态**，这一状态，只允许发送传送LCP协议的分组、鉴别协议的分组以及检测链路质量的模组。若使用**口令鉴别协议(PAP)**，则需要发起通信的乙方发送身份标识符和口令，也可以使用**口令握手鉴别协议(CHAP)**。
4. 在网络层协议状态下，**NCP**会根据网络层的不同协议交换网络层特定的网络控制分组。如果运行的是IP协议，那么PPP在配置每一端的IP模块时，就要使用**IPCP(IP控制协议)**，IPCP分组也使用PPP帧来处理。

5. 在进入**链路打开**状态下，PPP端点可以发送分组，两个PPP断点还可以发送回送**请求LCP分组(Echo-Request)**和**回送LCP回答分组(Echo-Reply)**，以检查链路的状态。
6. 数据传输结束后，可以由链路的一端发出**终止请求LCP分组(Terminate-Request)**请求终止链路连接，在收到对方发来的**终止确认LCP分组(Terminate-Ack)**后，转到链路终止状态。

这一段课本讲解的其实没有很深入的探讨，挖一个坑，应该会使用非常完全的PPP协议来替换掉，以保证讲解足够明白和深刻。

## 使用广播信道的数据链路层

### 局域网的数据链路层

局域网最主要的特点是：**网络为一个单位所拥有，且地理范围和站点数目均有限**。以太网基本已经霸占了局域网市场。

局域网的网络拓扑有**星形网**（Hub实现），**环型网**，**总线网**和**树形网**（书上没有提到过）。多台主机主要考虑的问题是共享信道，主要有两种解决方法，一个是**静态划分信道**，一种是**动态媒体接入控制**（随机接入和受控接入）。随机接入需要解决碰撞检测和解决的网络协议。

对于局域网而言，链路层被拆解成两个子层，**链路逻辑控制LLC**和**媒体接入控制MAC**，不过在1990年代以后，LLC协议就已经消失了，主要原因是DIX Ethernet V2的占有率提升和IEEE 802.3占有率几乎消失。

现代的计算机连入局域网都会使用适配器（网卡），网卡的硬件地址（MAC地址）会在适配器中写死，而IP地址则放在计算机的存储器中。在连入完成以后，网卡通过中断的方式告知计算机有帧需要接受，而计算机需要发送IP数据报时只需要把东西发给网卡就可以把东西丢出去，得益于PPP协议和当下的状况，计算机的数据从网卡走出去以后一般都会走向一个类似于集线器的终端，之后把数据发给其他网卡。

### CSMA/CD协议

这里书上讲的不错，有原因有解决方案，也有引申的问题。

在早期的以太网中，通常会使用**总线型**来处理，而且会采用**无连接**的工作方式，即**不必先建立连接就可以直接发送数据**，**适配器对发送的数据帧不进行编号，也不要求对方发回确认**。这样处理的原因是在比较近的局域网中，信号的质量往往很好，因此网络必须要尽最大的能力进行速度提升，代价就是降低一定的可靠性。

除此以外，以太网发送数据都是用**曼彻斯特编码**，优点和缺点都非常明显。优点显然是可以根据电压很方便把数据转移出来，缺点显然是这个频带宽度.....

总线存在一个问题，就是如果一个东西在传输数据，那么其他就不能传输数据，否则就会产生干扰，因此产生**CSMA/CD协议**来解决这个问题。

CSMA/CD的核心在于以下三点：

1. **多点接入**。即多个计算机在这个网络中。
2. **载波监听**。不论在发送之前还是在发送中，每个站都必须不停地检测信道。
3. **碰撞检测**。适配器边发送数据边检测信号的变化，如果超出预期则直接停止全部的发送，并产生一个等待时间，之后再发送。

于是乎需要解释一个问题：如果保证同一时间只有一个站点在发送数据，即在信道上没有信号的时候发送不就行了吗？其实这个问题忽略了一个日常情况下可以忽略的条件：时延。

在局域网的分析中，假设单程端到段的时间为 $\tau$ ，在信号传送到对面之前的 $\delta$ 时间内对面也发送了数据，那么就会在 $2\tau - \delta$ 的时刻，本机接收到了对面发送的信息正在干扰自己，因此需要退避。而 $0 \leq \delta < \tau$ ，那么就会有一个退避算法的解释。

1. 以 $2\tau$ 作为**基本退避时间（争用期）**，在这一段时间内，两边都应该停止发送数据。
2. 考虑到更多的机器一起工作的条件下，因此需要考虑**额外的时间**才开始重传，这一段时间应当是争用期的整数倍，假设为 $r$ ，那么就有一些关系。

$$r \in [0, 2^k - 1) \quad (1)$$

$$k = \min(\text{重传次数}, 10) \quad (2)$$

3. 当重传次数到了16次还是传不出去，那么应该丢弃该帧，并向上级报告。
4. 在以太网中，默认 $2\tau = 51.2\mu s$ 。

但是这会发生一个不小的问题：如果帧太小的话，那么，就会在A发送出去以后B没有接收到之前，B也发送了数据，但是双方都不知道被干扰了，因此必须要求64字节是起码要有的，同时要求长度小于64字节的帧都是由于冲突而异常终止的无效帧。

除此以外，还有**强化碰撞**的概念，就是认为发生一个碰撞来让用户知道了碰撞，只需要发送一个32/48bits的数据。以太网还要求帧的最小间隔是 $9.6\mu s$ ，这是为了方便刚刚收到数据帧的站进行数据清理。

## 使用集线器的星型拓扑

总线要求粗同轴电缆，因此存在一个成本问题，于是后来就是细同轴电缆，最后是双绞线和一个集线器就可以解决问题。

集线器还有一些特点如下：

1. 从表面上看，这是一个星形网，但是在逻辑上任然是一个总线网，使用的还是CSMA/CD协议，同一时刻最多只允许一个站发送数据。
2. 一个集线器有很多个接口，因此，一个集线器像一个多接口的转发器。
3. 集线器工作在物理层，仅仅是简单的转发比特，**不进行碰撞检测**。
4. 集线器采用了专门的芯片，进行自适应串音回波抵消，每个比特再转发之前进行再生整形并重新定时。

## 以太网的MAC层

### MAC地址

MAC地址又名**硬件地址**，**物理地址**，用于网络中的标识，而怎么找到这个MAC是路由的任务。严格的讲，局域网的“地址”应当是每一个站的“名字”或标识符。

对于MAC而言，IEEE使用了48位来标识每一个网卡，其中前24bits用于标识**组织唯一标识符OUI**，后面24bits交给厂家自己处理，只要不重复就行，这种方法得到的叫做**EUI-48(适配器标识符，Extended Unique Identifier)**。注意：

1. 有可能几个公司一起买一个EUI，也有可能一个公司买了几个EUI。
2. 前面24bits不一定所有的都用于OUI，IEEE规定**I/G(Individual/Group)**位为第一字节的最低位，如果为0表示单个站地址，如果为1表示组地址，用于多播。
3. IEEE同时把第一字节的最低第二位定义为**G/L(Global/Local)**位，如果位0表示**全球管理**（全球内唯一），如果为1则是**本地地址**，这时用户客人已分配网络上的地址。采用2字节地址字段时全都是本地管理。但是以太网似乎不想管这玩意。

所有的适配器应当都能识别**单播**和**广播**，而**多播**则通过编程方法来识别。除此以外以太网控制器可以设置成为**混杂方式**，偷偷把所有东西悄悄接收下来，这种使用方法对于两类人特别有用：黑客和网络工程师。

### MAC帧

MAC帧有两种标准，一种是以以太网V2标准，一个是IEEE的802.3标准，不过更多的是V2标准这看上去是平行于PPP帧的一种标准。在IP层下来的数据前面加上一些东西。

6	6	2	46~1500	4
目的地址	源地址	类型	IP 数据报	FCS

其中目的地址和源地址是MAC地址，类型是对于上一层使用的协议的标注，比如IP数据报则是0x0800，而0x8137则是NovellIPX发过来的，最后的FCS自然是检验序列。

不过MAC帧有一个问题，就是什么时候结束？实际上因为使用的是曼彻斯特编码，因此根本不需要，在最后的时候输出电压也就不会改变了。

在MAC帧包装完成之后，需要在之前加入8bytes用来时钟同步，其中前面7bytes用于时钟同步，最后一个bytes用于帧开始定界符。

这个时候，要求MAC帧超过64字节用于CSMA/CD，这意味着IP数据报至少要46个字节，如果长度不够则需要强行添加内容，在返回给IP协议时，因为IP协议的第一个字段就是长度，所以不需要做什么判断。

## 扩展的以太网

### 在物理层扩展以太网

就是用集线器树状扩展以太网。

优点：

- 原来不同碰撞域的局域网计算机能够跨碰撞域的通信。
- 扩大了局域网覆盖的地理范围。

缺点：

- 碰撞域增大了，但是总的吞吐量没有提高。
- 如果不同的碰撞域使用不同的速率，那么连接在一起的时候就是所有的速率里面最低的速率运行。

### 在数据链路层扩展以太网

最早的方法时使用网桥，根据MAC帧的目的地址进行转发和过滤，但是注意网桥只能链接两端。后来1990年出现了**交换式集线器**，很快淘汰了网桥，常被称为**以太网交换机**，虽然本质上是多接口的网桥，一般工作在全双工方式。相互通信的主机都是独占传输媒体，无碰撞的传输数据。

最大的区别在于以太网交换机有缓存能力，可以在数据繁忙的时候暂存一下数据。和网桥相似的地方在于**帧交换表**，可以自学习了解每个端口放了什么主机。只有在发送端发送数据的时候，才会把接口和主机信息进行映射。如果学习过的端口过了一段时间就会自动删除。

但是交换机会存在一个bug，就是在进行冗余设计的时候会把以太网交换机的端口互联，从而发生绕圈子的情况，这个问题在802.1D标准协议中制定的**生成树协议STP**，在不改变拓扑的情况下剪掉某些逻辑上的链路。

以太网有几大根基，其中两个是CSMA/CD协议和以太网的帧格式，在交换机缠身以后CSMA/CD协议已经消失。

## 虚拟局域网

**虚拟局域网VLAN**，在IEEE 802.1Q中定义是这样的：

虚拟局域网VLAN是由一些局域网网段构成的与物理位置无关的逻辑组，而这些网段具有某些共同的需求。每一个VLAN的帧都有一个明确的标识符，指明发送这个帧的计算机属于哪一个VLAN。

虚拟局域网可以把连接在同一个交换机的计算机分隔开，并且可以跨越交换机，这有效限制了接收广播信息的计算机数。在**802.1Q**标准下，加入一个4bytes的VLAN标记，其中前面两个字节为定死的0x81，后面的为用户优先级，**规范格式指示符CFI**和**VLAN标识符VID**。

## 高速以太网

### 100BASE-T以太网

1995年IEEE把100BASE-T定位正是标准，代号IEEE 802.3u，可以进入全双工方式，也可以在半双工方式下工作。注意：**细缆以太网升级到快速以太网的必须重新布线**。而且，因为速度的提升，以太网帧的长度不能改变，因此争用期减小为1/10，帧之间的最小间隔也是1/10。

以太网有一个很重要的参数 $a$ ，这个参数是以太网传输数据效率的标志，为了高效， $a$ 必须很小。而且 $a$ 满足

$$a = \tau / T_0 \quad (3)$$

众所周知， $T_0$ 是帧长度和速度的比值，帧长度没有减小，速度上升了10倍，因此 $\tau$ 必须减小到1/10，进而最大长度变成了100m。

### 1Gbit以太网

1997年IEEE通过802.3z标准以支持1Gbit的以太网，也是现在基本普及的网络，甚至部分城市的宽带已经不是100BASE-T以太网能容纳得了的了。

这种网络在半双工条件下使用CSMA/CD协议，但是全双工就不使用。

在半双工条件下，不能单纯减小最大长度了，10m根本不够用，因此Gbit以太网的最大长度还是100m，但是采用了**载波延伸**的方法，最短帧长不变，但是争用期长度变成了512字节。

但是这又会引来一个问题，64字节的帧填满到512字节，会导致数据的利用率极大降低，因此Gbit以太网新加了一个功能就是**分组突发**，可以把很多个短帧进行合并到满足要求。

### 10Gbit以太网和更快的以太网

只工作在全双工模式，也不使用CSMA/CD协议，因此传输距离大大提高了，只需要考虑信号衰减。

### 使用以太网进行宽带接入

- 以太网接入的重要特点是它可提供双向的宽带通信，并且可根据用户对带宽的需求灵活地进行带宽升级。
- 采用以太网接入可实现端到端的以太网传输，中间不需要再进行帧格式的转换。这就提高了数据的传输效率和降低了传输的成本。