

以太网设计 FAQ：以太网 MAC 和 PHY

问：如何实现单片以太网微控制器？

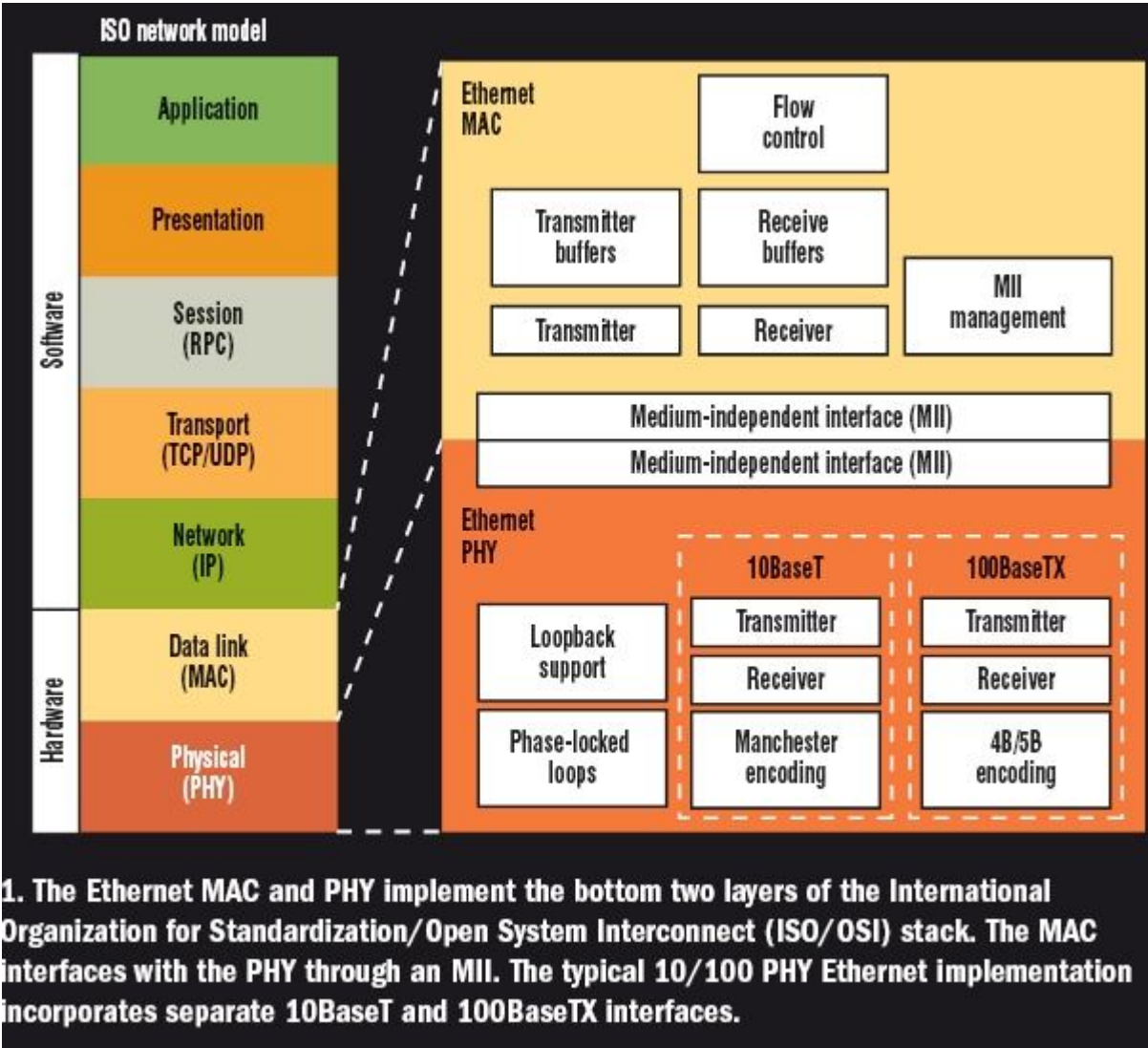
答：诀窍是将微控制器、以太网媒体接入控制器(MAC)和物理接口收发器(PHY)整合进同一芯片，这样能去掉许多外接元器件。这种方案可使 MAC 和 PHY 实现很好的匹配，同时还可减小引脚数、缩小芯片面积。单片以太网微控制器还降低了功耗，特别是在采用掉电模式的情况下。

问：以太网 MAC 是什么？

答：MAC 就是媒体接入控制器。以太网 MAC 由 IEEE-802.3 以太网标准定义。它实现了一个数据链路层。最新的 MAC 同时支持 10Mbps 和 100Mbps 两种速率。通常情况下，它实现 MII 接口。

问：什么是 MII？

答：MII 即媒体独立接口，它是 IEEE-802.3 定义的以太网行业标准。它包括一个数据接口，以及一个 MAC 和 PHY 之间的管理接口(图 1)。数据接口包括分别用于发送器和接收器的两条独立信道。每条信道都有自己的数据、时钟和控制信号。MII 数据接口总共需要 16 个信号。管理接口是个双信号接口：一个是时钟信号，另一个是数据信号。通过管理接口，上层能监视和控制 PHY。



问：以太网 PHY 是什么？

答：PHY 是物理接口收发器，它实现物理层。IEEE-802.3 标准定义了以太网 PHY。它符合 IEEE-802.3k 中用于 10BaseT(第 14 条)和 100BaseTX(第 24 条和第 25 条)的规范。

问：造成以太网 MAC 和 PHY 单片整合难度高的原因是什么？

答：PHY 整合了大量模拟硬件，而 MAC 是典型的全数字器件。芯片面积及模拟/数字混合架构是为什么先将 MAC 集成进微控制器而将 PHY 留在片外的原因。更灵活、密度更高的芯片技术已经可以实现 MAC 和 PHY 的单芯片整合。

问： 除 RJ-45 接口外，还需要其它元件吗？

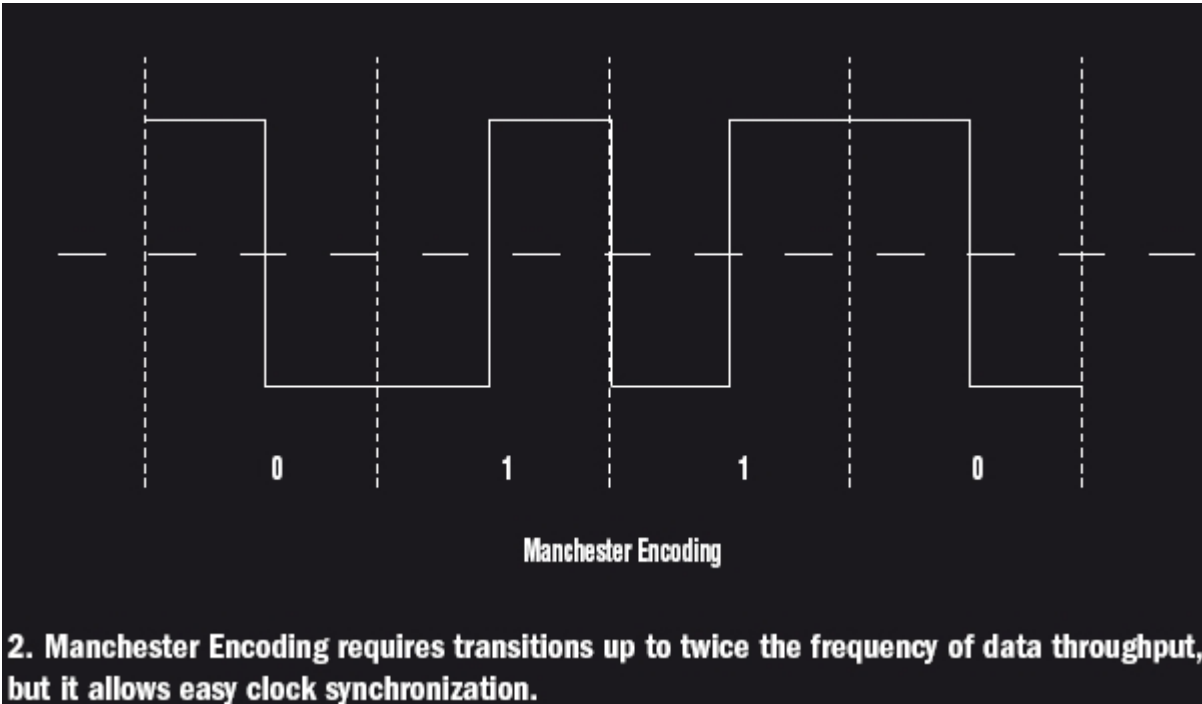
答：需要其它元件。虽然 PHY 提供绝大多数模拟支持，但在一个典型实现中，仍需外接 6、7 只分立元件及一个局域网绝缘模块。绝缘模块一般采用一个 1：1 的变压器。这些部件的主要功能是为了保护 PHY 免遭由于电气失误而引起的损坏。

问： 10BaseT 和 100BaseTX PHY 实现方式不同的原因何在？

答：两种实现的分组描述本质上是一样的，但两者的信令机制完全不同。其目的是阻止一种硬件实现容易地处理两种速度。10BaseT 采用曼彻斯特编码，100BaseTX 采用 4B/5B 编码。

问： 什么是曼彻斯特编码？

答：曼彻斯特编码又称曼彻斯特相位编码，它通过相位变化来实现每个位(图 2)。通常，用一个时钟周期中部的上升沿表示“1”，下降沿表示“0”。周期末端的相位变化可忽略不计，但有时又可能需要将这种相位变化计算在内，这取决于前一位的值。



问： 什么是 4B/5B 编码？

答：4B/5B 编码是一种块编码方式。它将一个 4 位的块编码成一个 5 位的块。这就使 5 位块内永远至少包含 2 个“1”转换，所以在一个 5 位块内总能进行时钟同步。该方法需要 25%的额外开销。

作者：William Wong，嵌入式/系统/软件编辑，《Electronic Design》

有关 MAC、PHY 和 MII

以太网（Ethernet）是一种计算机局域网组网技术，基于 IEEE 制定的 IEEE 802.3 标准，它规定了包括物理层的连线、电信号和介质访问层协议的内容。以太网是当前应用最普遍的局域网技术。它很大程度上取代了其他局域网标准，如令牌环、FDDI 和 ARCNET。历经 100M 以太网在上世纪末的飞速发展后，目前千兆以太网甚至 10G 以太网正在

Ethernet 的接口实质是 MAC 通过 MII 总线控制 PHY 的过程。

1. MAC

MAC 是 Media Access Control 的缩写，即媒体访问控制子层协议。该协议位于 OSI 七层协议中数据链路层的下半部分，主要负责控制与连接物理层的物理介质。在发送数据的时候，MAC 协议可以事先判断是否可以发送数据，如果可以发送将给数据加上一些控制信息，最终将数据以及控制信息以规定的格式发送到物理层；在接收数据的时候，MAC 协议首先判断输入的信息并是否发生传输错误，如果没有错误，则去掉控制信息发送至 LLC 层。以太网 MAC 由 IEEE-802.3 以太网标准定义。

2. MII

MII 即媒体独立接口，“媒体独立”表明在不对 MAC 硬件重新设计或替换的情况下，任何类型的 PHY 设备都可以正常工作。包括分别用于发送器和接收器的两条独立信道。每条信道都有自己的数据、时钟和控制信号。MII 数据接口总共需要 16 个信号，包括：

transmit data - TXD[3:0]

transmit strobe - TX_EN

transmit clock - TX_CLK

transmit error - TX_ER/TXD4

receive data - RXD[3:0]

receive strobe - RX_DV

receive clock - RX_CLK

receive error - RX_ER/RXD4

collision indication - COL

carrier sense - CRS

一般说来，包括：

IC 对 PHY 作读取与写入用的一组信号：MDC(clock),MDIO(data)

做为 data sampling reference 用的两组 clock，频率应为 25MHz(TX_CLK,RX_CLK)

各 4-bit 的输出、输入 Bus(TX[0:3],RX[0:3])

通知对方准备输入数据的输出、输入的启动信号(TX_EN)

输出、输入的错误通知信号(TX_ER,RX_ER)

得到有效输入数据的通知信号(RX_DV)

网络出现拥塞的 colision 信号(Col)

做为 carrier 回复用的信号(CRS)

电位可使用+5V 或+3.3V

MII 以 4bit,即半字节方式双向传送数据，时钟速率 25MHz，其工作速率可达 100Mb/s。MII 管理接口是个双信号接口，一个是时钟信号，另一个是数据信号。通过管理接口，上层能监视和控制 PHY。MII 界面传递了网络的所有数据和数据的控制，而 MAC 对 PHY 的工作状态的确定和对 PHY 的控制则是使用 SMI (Serial Management Interface) 界面通过读写 PHY 的寄存器来完成的。PHY 里面的部分寄存器是 IEEE 定义的，这样 PHY 把自己的目前的状态反映到寄存器里面，MAC 通过 SMI 总线不断的读取 PHY 的状态寄存器以得知目前 PHY 的状态，例如连接速度，双工能力等。当然也可以通过 SMI 设置 PHY 的寄存器达到控制的目的，例如流控的打开关闭，自协商模式还是强制模式等。不论是物理连接的 MII 总线和 SMI 总线还是 PHY 的状态寄存器和控制寄存器都是有 IEEE 的规范的，因此不同公司的 MAC 和 PHY 一样可以协调工作。当然为了配合不同公司的 PHY 的自己特有的一些功能，驱动需要做相应的修改。RMII(Reduced Media Independant Interface)简化媒体独立接口,是标准的以太网接口之一，比 MII 有更少的 I/O

传输。

关于 RMII 口和 MII 口的区别：

RMII 口是用 2 根线来传输数据

MI I 口是用 4 根线来传输数据

GMII 是用 8 根线来传输数据

MI I/RMII 只是一种接口，对于 10M 线速，MI I 的时钟是 2.5M，RMII 则是 5M；对于 100M 线速，MI I 的时钟是 25M，RMII 则是 50M。

The standard MI I features a small set of registers:

- * Basic Mode Configuration (#0)
- * Status Word (#1)
- * PHY Identification (#2, #3)
- * Ability Advertisement (#4)
- * Link Partner Ability (#5)
- * Auto Negotiation Expansion (#6)

MI I/RMII 用于传输以太网包，在 MI I/RMII 接口是 4/2bit 的，在以太网的 PHY 里需要做串并转换、编解码等才能在双绞线和光纤上进行传输，其帧格式遵循 IEEE 802.3(10M)/IEEE 802.3u(100M)/IEEE 802.1q(VLAN)。

以太网帧的格式为：前导符+开始位+目的 mac 地址+源 mac 地址+类型/长度+数据

+padding(optional)+32bitCRC，如果有 vlan，则要在类型/长度后面加上 2 个字节的 vlan tag，其中 12bit 来表示 vlan id，另外 4bit 表示数据的优先级！

3. PHY

PHY 是物理接口收发器，它实现物理层。包括 MI I/GMII（介质独立接口）子层、PCS（物理编码子层）、PMA（物理介质附加）子层、PMD（物理介质相关）子层、MDI 子层。

100BaseTX 采用 4B/5B 编码。PHY 在发送数据时，收到 MAC 过来的数据（对 PHY 来说，没有帧的概念，对它来说，都是数据而不管什么地址，数据还是 CRC），每 4bit 就增加 1bit 的检错码，然后把并行数据转化为串行流数据，再按照物理层的编码规则把数据编码，再变为模拟信号把数据送出去。收数据时的流程反之。PHY 还有个重要的功能就是实现 CSMA/CD 的部分功能。它可以检测到网络上是否有数据在传送，如果有数据在传送中就等待，一旦检测到网络空闲，再等待一个随机时间后将送数据出去。如果两个碰巧同时送出了数据，那样必将造成冲突，这时候，冲突检测机构可以检测到冲突，然后各等待一个随机的时间重新发送数据。这个随机时间很有讲究的，并不是一个常数，在不同的时刻计算出来的随机时间都是不同的，而且有多重算法来应付出现概率很低的同两台主机之间的第二次冲突。通信速率通过双方协商，协商的结果是两个设备中能同时支持的最大速度和最好的双工模式。这个技术被称为 Auto Negotiation 或 NWAY。隔离变压器把 PHY 送出来的差分信号用差模耦合的线圈耦合滤波以增强信号，并且通过电磁场的转换耦合到连接网线的另外一端。RJ-45 中 1、2 是传送数据的，3、6 是接收数据的。新的 PHY 支持 AUTO MDI-X 功能(也需要隔离变压器支持)。它可以实现 RJ-45 接口的 1、2 上的传送信号线和 3、6 上的接收信号线的功能自动互相交换

网卡工作在 osi 的最后两层，物理层和数据链路层，物理层定义了数据传送与接收所需要的电与光信号、线路状态、时钟基准、数据编码和电路等，并向数据链路层设备提供标准接口。物理层的芯片称之为 PHY。数据链路层则提供寻址机构、数据帧的构建、数据差错检查、传送控制、向网络层提供标准的数据接口等功能。以太网卡中数据链路层的芯片称之为 MAC 控制器。很多网卡的这两个部分是做到一起的。他们之间的关系是 pci 总线接 mac 总线，mac 接 phy，phy 接网线（当然也不是直接接上的，还有一个变压装置）。

由此可见，MAC 和 PHY，一个是数据链路层，一个是物理层；两者通过 MI I 传送数据。

网卡的组成工作原理

1.认识网卡，我们上网必备组件之一。

网卡工作在 OSI 的最后两层：物理层和数据链路层，物理层定义了数据传送与接收所需要的电与光信号、线路状态、时钟基准、数据编码和电路等，并向数据链路层设备提供标准接口。物理层的芯片称之为 PHY。数据链路层则提供寻址机构、数据帧的构建、数据差错检查、传送控制、向网络层提供标准的数据接口等功能。以太网卡中数据链路层的芯片称之为 MAC 控制器。很多网卡的这两个部分是做到一起的。他们之间的关系是 pci 总线接 mac 总线，mac 接 phy，phy 接网线（当然也不是直接接上的，还有一个变压装置）。

下面继续让我们来关心一下 PHY 和 MAC 之间是如何传送数据和相互沟通的。通过 IEEE 定义的标准 MII/GigaMII(Media Independent Interfade, 介质独立界面)界面连接 MAC 和 PHY。这个界面是 IEEE 定义的。MII 界面传递了网络的所有数据和数据的控制。而 MAC 对 PHY 的工作状态的确定和对 PHY 的控制则是使用 SMI(Serial Management Interface)界面通过读写 PHY 的寄存器来完成的。PHY 里面的部分寄存器也是 IEEE 定义的，这样 PHY 把自己的目前的状态反映到寄存器里面，**MAC 通过 SMI 总线不断的读取 PHY 的状态寄存器以得知目前 PHY 的状态**，例如连接速度，双工的能力等。当然也可以通过 SMI 设置 PHY 的寄存器达到控制的目的，例如流控的打开关闭，自协商模式还是强制模式等。

我们看到了，不论是物理连接的 MII 界面和 SMI 总线还是 PHY 的状态寄存器和控制寄存器都是有 IEEE 的规范的，因此不同公司的 MAC 和 PHY 一样可以协调工作。当然为了配合不同公司的 PHY 的自己特有的一些功能，驱动需要做相应的修改。

一片网卡主要功能的实现就基本上是上面这些器件了。其他的，还有一颗 EEPROM 芯片，通常是一颗 93C46。里面记录了网卡芯片的供应商 ID、子系统供应商 ID、网卡的 MAC 地址、网卡的一些配置，如 SMI 总线上 PHY 的地址，BOOTROM 的容量，是否启用 BOOTROM 引导系统等东西。

很多网卡上还有 BOOTROM 这个东西。它是用于无盘工作站引导操作系统的。既然无盘，一些引导用必需用到的程序和协议栈就放到里面了，例如 RPL、PXE 等。实际上它就是一个标准的 PCI ROM。所以才会有一些硬盘写保护卡可以通过烧写网卡的 BootRom 来实现。其实 PCI 设备的 ROM 是可以放到主板 BIOS 里面的。启动电脑的时候一样可以检测到这个 ROM 并且正确识别它是什么设备的。AGP 在配置上和 PCI 很多地方一样，所以很多显卡的 BIOS 也可以放到主板 BIOS 里面。这就是为什么板载的网卡我们从来没有看到过 BOOTROM 的原因。

2.工作过程

PHY 在发送数据的时候，收到 MAC 过来的数据(对 PHY 来说，没有帧的概念，对它来说，都是数据而不管什么地址，数据还是 CRC)，每 4bit 就增加 1bit 的检错码，然后把并行数据转化为串行流数据，再按照物理层的编码规则(10Based-T 的 NRZ 编码或 100based-T 的曼彻斯特编码)把数据编码，再变为模拟信号把数据送出去。收数据时的流程反之。现在来了解 PHY 的输出后面部分。一颗 CMOS 制程的芯片工作的时候产生的信号电平总是大于 0V 的(这取决于芯片的制程和设计需求)，但是这样的信号送到 100 米甚至更长的地方会有很大的直流分量的损失。而且如果外部网线直接和芯片相连的话，电磁感应(打雷)和静电，很容易造成芯片的损坏。

再就是设备接地方法不同，电网环境不同会导致双方的 0V 电平不一致，这样信号从 A 传到 B，由于 A 设备的 0V 电平和 B 点的 0V 电平不一样，这样会导致很大的电流从电势高的设备流向电势低的设备。我们如何解决这个问题呢？这时就出现了 Transformer(隔离变压器)这个器件。它把 PHY 送出来的差分信号用差模耦合的线圈耦合滤波以增强信号，并且通过电磁场的转换耦合到连接网线的另外一端。这样不但使网线和 PHY 之间没有物理上的连接而换传递了信号，隔断了信号中的直流分量，还可以在不同 0V 电平的设备中传送数据。

隔离变压器本身就是设计为耐 2KV~3KV 的电压的。也起到了防雷感应(我个人认为这里用防雷击不合适)保护的作用。有些朋友的网络设备在雷雨天气时容易被烧坏，大都是 PCB 设计不合理造成的，而且大都烧毁了设备的接口，很少有芯片被烧毁的，就是隔离变压器起到了保护作用。

发送数据时，网卡首先侦听介质上是否有载波（载波由电压指示），如果有，则认为其他站点正在传送信息，继续侦听介质。一旦通信介质在一定时间段内（称为帧间缝隙 IFG= 9.6 微秒）是安静的，即没有被其他站点占用，则开始进行帧数据发送，同时继续侦听通信介质，以检测冲突。在发送数据期间，如果检测到冲突，则立即停止该次发送，并向介质发送一个“阻塞”信号，告知其他站点已经发生冲突，从而丢弃那些可能一直在接收的受到损坏的帧数据，并等待一段随机时间（CSMA/CD 确定等待时间的算法是二进制指数退避算法）。在等待一段随机时间后，再进行新的发送。如果重传多次后（大于 16 次）仍发生冲突，就放弃发送。

接收时，网卡浏览介质上传输的每个帧，如果其长度小于 64 字节，则认为是冲突碎片。如果接收到的帧不是冲突碎片且目的地址是本地地址，则对帧进行完整性校验，如果帧长度大于 1518 字节（称为超长帧，可能由错误的 LAN 驱动程序或干扰造成）或未能通过 CRC 校验，则认为该帧发生了畸变。通过校验的帧被认为是有效的，网卡将它接收下来进行本地处理

网卡的原理及测试技术

网卡充当计算机和网络缆线之间的物理接口或连线将计算机中的数字信号转换成电或光信号,称为 NIC（ network interface card ）。数据在计算机总线中传输是并行方式即数据是肩并肩传输的，而在网络的物理缆线中说数据以串行的比特流方式传输的，网卡承担串行数据和并行数据间的转换。网卡在发送数据前要同接收网卡进行对话以确定最大可发送数据的大小、发送的数据量的大小、两次发送数据间的间隔、等待确认的时间、每个网卡在溢出前所能承受的最大数据量、数据传输的速度。

一、网卡的基本构造

网卡包括硬件和固件程序（只读存储器中的软件例程），该固件程序实现逻辑链路控制和媒体访问控制的功能，还记录唯一的硬件地址即 mac 地址，网卡上一般有缓存。网卡须分配中断 irq 及基本 i/o 端口地址，同时还须设置基本内存地址（base memory address）和收发器（transceiver）。

网卡的控制芯片是网卡中最重要元件，是网卡的控制中心，有如电脑的 cpu，控制着整个网卡的工作，负责数据的的传送和连接时的信号侦测。早期的 10/100m 的双速网卡会采用两个控制芯片（单元）分别用来控制两个不同速率环境下的运算，而目前较先进的产品通常只有一个芯片控制两种速度。

晶体震荡器负责产生网卡所有芯片的运算时钟，其原理就象主板上的晶体震荡器一样，通常网卡是使用 20 或 25hz 的晶体震荡器。

boot rom 插槽如无特殊要求网卡中的这个插槽处在空置状态。一般是和 boot rom 芯片搭配使用，其主要作用是引导电脑通过服务器引导进入 win9x。

boot rom 就是启动芯片，让电脑可以在不具备硬盘、软驱和光驱的情况下，直接通过服务器开机，成为一个无硬盘无软驱的工作站。没有软驱就无法将资料输出，这样也可以达到资料保密的功能。同时，还可以节省下购买这些电脑部件的费用。在使用 boot rom 时要注意自己使用何种网络操作系统，通常有 boot rom for nt, boot rom for unix, boot rom for netware 等，boot rom 启动芯片要自行购买。

eprom从前的老式网卡都要靠设置跳线或是 dip 开关来设定 irq、dma 和 i/o port 等值，而现在的网卡则都使用软件设

定，几乎看不见跳线的存在。各种网卡的状态和网卡的信息等数据都存在于这颗小小的 **eeeprom** 里，通过它来自动设置。内接式转换器只要有 **bnc** 接头的网卡都会有这个芯片，并紧邻在 **bnc** 接头旁，它的功能是在网卡和 **bnc** 接头之间进行数据转换，让网卡能通过它从 **bnc** 接头送出或接收资料。

rj-45 和 **bnc** 接头 **rj-45** 是采用双绞线作为传输媒介的一种网卡接口，在 **100mbps** 网中最常应用。**bnc** 是采用细同轴电缆作为传输媒介

信号指示灯在网卡后方会有二到三个不等的信号灯，其作用是显示目前网络的连线状态，通常具有 **tx** 和 **rx** 两个信息。**tx** 代表正在送出资料，**rx** 代表正在接收资料，若看到两个灯同时亮则代表目前是处于全双工的运作状态，也可由此来辨别全双工的网卡是否处于全双工的网络环境中（见上图两个接口的中间部分）。也有部分低速网卡只用一个灯来表示信号，通过不同的灯光变换来表示网络是否导通。

二、网卡的分类

三、网卡测试技术

基于操作系统的测试

网卡一个重要的性能是看其是否支持多种网络操作系统，比较流行的网络操作系统有 **windowsnt**、**unix** (**linux**、**freebsd**、**sco**、**solaris**、**hp** 底)、**novell**、**dec** 等。同时网卡应能够支持多种的网络协议，如 **tcp/ip**、**ipx/spx**、**apple**、**netbeui** 等。

基于主机的兼容性测试

硬件上的兼容性也是非常重要的一个方面，尤其在笔记本电脑上兼容性问题比较突出，根据本人的实际经验，甚至某些名牌的网卡在一些笔记本电脑上也存在较为严重的兼容性问题。在服务器或台式电脑方面这些问题不常出现。

网卡传输速率测试（数据吞吐量）

测试网卡的传输速率一般有硬件和软件两种方法，硬件是利用一些专用的仪器如网络分析仪、**smartbits smartcards** 等其他一些设备，利用 **icmp echo** 请求和 **udp** 数据包来检测数据流量。通常测试的项目有以下几方面：

autonegotiation test

测试网卡速率、全双工/半双工和流控协商。协商决定着是否通过“暂停帧 **pause frame**”来允许流量控制。

arp test

测试网卡是否能对 **arp** 请求做出正确回应及是否在规定时间内应答。这个时间由测试者进行设置。

error test

测试网卡处理错误 **frame** 的能力，通常在较低的传输速率下进行此项测试（**0.5%**传输速率），有以下几个方面的测试：网卡接收正确的 **frame**，作出处理。

网卡接收到存在 **crc** 校验错的 **frame**，网卡将其丢弃。

网卡接收到传输顺序错误的 **frame**，网卡将其丢弃。

网卡接收到含有少量错误 **bits** 的 **frame**，网卡应全部接收并处理。

网卡接收到超小 **frame**，网卡应将其丢弃。

网卡接收到超长 **frame**，网卡应将其丢弃。

packets loss test

rfc 规定测试网卡在各种传输带宽利用率下的处理 **frame** 的能力，从初始化数据传输到传输速率的不断变化一直到传输结束，检查 **frame** 的丢失情况。

throughput test

数据吞吐量的测试也是 **rfc** 规定的一项测试内容，测试的结果反映出传输的最大带宽的利用率，每秒处理的 **frame** 和每秒处理的 **bits** 数量。

back-to-back test

同样此项测试也为 **rfc-2544** 的规定，测试在一个设定的最大传输速率下网卡可处理的并发 **frame** 的数量。最终反映出在不丢失数据包的情况下可并发传输的最大 **frame** 数量。

利用软件测试通常是利用 **zd** 的 **netbench** 来测试，一般只利用其测试网卡的最高传输速率。测试时要组成一个网络结构，一台 **windowsnt server** 服务器，若干个 **windows9x** 或 **windowsnt station** 客户端，传输大容量的文件如 **100mbps**，测试的结果将反映出网卡的最高传输速率。另一个测试项目是测试网卡对较小的数据包请求的回应能力，这里有必要

讨论一下 tcp/ip 的 ping 命令的机制。ping 是利用发送和接收 icmp echo 报文，来检测链路状态和协议设置。数据链路层封装的是 frame，大小在 64k~1518k 之间，当发送 frame 时，网卡接受到 frame 时首先要读取帧头和帧尾的 mac 地址，当 mac 地址相匹配时再接封装读取 ip 地址。当网卡连续接收到 frame 时，要对每一个 frame 做出处理，当网卡或是系统无法处理这些数据包时，这些数据包将被丢弃。这种情况多发生在连续发送非常小的 frame 时。ping 的机制是发送一个 icmp 报文，接收到一个 icmp echo 后再发送下一个 icmp 报文。所以较小的连续的 frame 会对网卡和系统造成较大的压力。在 netbench 中，有一项测试就是测试网卡或系统对连续的小数据包的处理能力。

稳定性测试

一块好的网卡应该具有良好的稳定性，具体讲就是在不同的工作环境下和不同的工况下应具有稳定的表现。通常测试主要是高温和传输大文件测试。

高温测试一般是在 30~35 摄氏度下连续运行网卡的测试程序达一定的时间比如 2 小时以上，检测网卡高温下的稳定性。pcmcia 接口的网卡一般有两种 32 位的和 16 位的，前者又称为 cardbus 网卡，数据带宽由 16 位增加到 32 位，使得 pcmcia 的网卡发热量成为一个显著的问题。

另一个测试是传输大的文件，某些品质较差的网卡在传输大容量的文件比如 2gbps 以上的文件时容易出错。

综上所述，在测试一块网卡时要进行全面的软、硬件及兼容性测试，可根据具体的应用和不同的要求，有机的选择测试项目，正确反映网卡的性能指标。
