此文章已于 14:03:36 2016/7/14 发布到 MADHEX

网口扫盲

**[网口扫盲一:网卡初步认识](http://www.cnblogs.com/jason-lu/p/3198424.html)**

网络适配器又称网卡或网络接口卡(NIC),英文名Network Interface Card.它是使计算机联网的设备.平常所说的网卡就是将PC机和LAN连接的网络适配器.网卡(NIC) 插在计算机主板插槽中,负责将用户要传递的数据转换为网络上其它设备能够识别的格式,通过网络介质传输.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.

它的主要技术参数为带宽,总线方式,电气接口方式等.它的基本功能为:从并行到串行的数据转换,包的装配和拆装,网络存取控制,数据缓存和网络信号.

网卡的主要工作原理:发送数据时, 计算机把要传输的数据并行写到网卡的缓存,网卡对要传输的数据进编码(10M以太网使用曼切斯特码,100M 以太网使用差分曼切斯特码), 串行发到传输介质上.接收数据时, 则相反.

**1. 网卡的基本构造**

以最常见的PCI 接口的网卡为例,一块网卡主要由 PCB 线路板,主芯片,数据汞,金手指(总线插槽接口) ,BOOTROM,EEPROM,晶振,RJ45接口,指示灯,固定片等等,以及一些二极管,电阻电容等组成.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)

1. 网卡的控制芯片: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,负责数据的传送和连接时的信号侦测.早期的10/100Mbps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.

常见的 10/100/1000M bps自适应网卡芯片有 Intel 的8254\* 系列,Broadcom 的BCM57\*\*系列,Marvell的  
88E8001/88E8053/88E806\*系列,Realtek的RTL8169S-32/64,RTL8110S-32/64(LOM),RTL8169SB,RTL8110SB(LOM) ,RTL8168(PCI Express) ,RTL8111(LOM,PCI Express) 系列,VIA 的VT612\*系列等等.

1. 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.千兆网卡使用62.5MHz或者125MHz晶振.
2. 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操作系统.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.
3. eeprom: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这颗小小的eeprom里,通过它来自动设置.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
4. 数据汞:这是消费级PCI 网卡上都具备的设备,数据汞也被叫做网络变压器或可称为网络隔离变压器.它在一块网卡上所起的作用主要有两个,一是传输数据,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不同电平的连接网线的另外一端;一是隔离网线连接的不同网络设备间的不同电平,以防止不同电压通过网线传输损坏设备.除此而外,数据汞还能对设备起到一定的防雷保护作用.
5. 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.
6. 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.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.
7. WOL:有些网卡会有WOL的功能, WOL网络开机的功能(wake on line).它可由另外一台电脑,使用软件制作特殊格式的信息包发送至一台装有具wol功能网卡的电脑,而该网卡接收到这些特殊格式的信息包后,就会命令电脑打开电源,目前已有越来越多的网卡支持网络开机的功能.

**2. 网卡的分类**

* 以传输速率可分为:

10Mbps网卡,100Mbps网卡,1000Mbps网卡,10GMbps网卡.目前常见的三种架构有10baset,100basetx与base2,前两者是以rj-45双绞线为传输媒介,传输速率分别为10Mbps和100Mbps.而双绞线又分为category 1至category 5五种规格,分别有不同的用途以及频宽,category通常简称cat,只要使用cat5规格的双绞线皆可用于10/100mbps的网卡上.而10base2架构则是使用细同轴电缆作为传输媒介,传输速率只有10Mbps.这里提到的10Mbps或100Mbps是指网卡上的最大传送速率,而并不等于网络上实际的传送速度,实际速度要考虑到传送的距离,线路的品质,和网络上是否拥挤等因素,这里所谈的bps指的是每秒传送的bit(1个byte=8个bit).而100Mbps则称为高速以太网卡(fast ethernet),多为PCI/PCI-E接口.当前市面上的pci网卡多具有10/100/1000Mbps自动切换的功能,会根据所在的网络连线环境来自动调节网络速度.1000 Mbps以太网卡多用于交换机或交换机与服务器之间的高速链路或backbone.

* 以接口类型可分为:

ISA接口网卡,PCI/ PCI-X/ PCI-E接口网卡,USB接口网卡和笔记本电脑专用的PCMCIA接口.现在的ISA接口的网卡均采用16bit的总线宽度,其特性是采用programmed i/o的模式传送资料,传送数据时必须通过cpu在i/o上开出一个小窗口,作为网卡与pc之间的沟通管道,需要占用较高的cpu使用率,在传送大量数据时效率较差. PCI接口的网卡则采用32bit的总线频宽,采用bus master的数据传送方式,传送数据是由网卡上的控制芯片来控制,不必通过i/o端口和cpu,可大幅降低cpu的占用率,目前产品多为10/100Mbps双速自动侦测切换网卡.

* 以传输方式可分为:

半双工网卡,全双工网卡.半双工网卡无法同一时间内完成接收与传送数据的动作,如10base2使用细同轴电缆的网络架构就是半双工网络,同一时间内只能进行传送或接收数据的工作,效率较低.要使用全双工的网络就必须要使用双绞线作为传输线才能达到,并且也要搭配使用全双工的集线器,要使用10base或100basetx的网络架构,网卡当然也要是全双工的产品.

* 以传输介质可分为:

rj-45双绞线的网卡与bnc的同轴电缆两种,有的网卡同时具有两种接头,可适用于两种网络线,但无法两个接头同时使用.另外还有光纤接口的网卡,通常带宽在1000 Mbps.

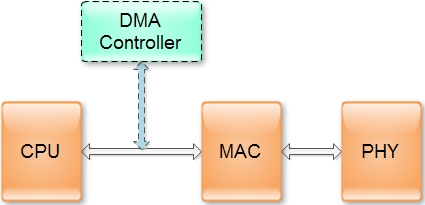
* 其它网卡:

从网络传输的物理媒介上还有无线网卡,利用2.4GHz的无线电波来传输数据.目前ieee有两种规范802.11和802.11b,最高传输速率分别为2Mbps和11Mbps,接口有PCI,USB和PCMCIA几种.

[**网口扫盲二:Mac与Phy组成原理的简单分析**](http://www.cnblogs.com/jason-lu/p/3196096.html)

**1. general**

下图是网口结构简图.网口由CPU、MAC和PHY三部分组成.DMA控制器通常属于CPU的一部分,用虚线放在这里是为了表示DMA控制器可能会参与到网口数据传输中.

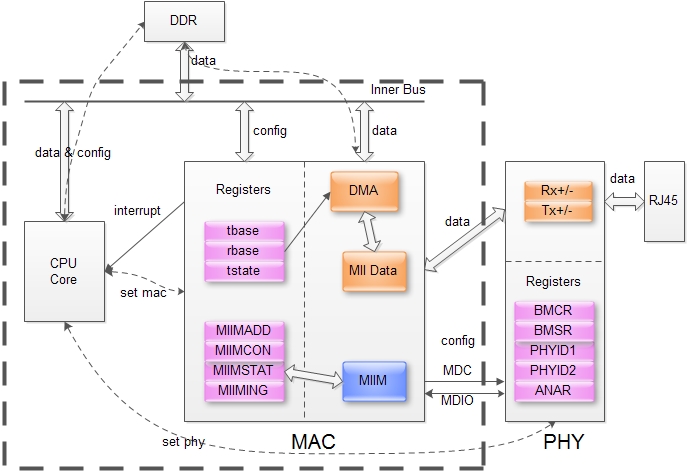


对于上述的三部分,并不一定都是独立的芯片,根据组合形式,可分为下列几种类型:

1. CPU集成MAC与PHY;
2. CPU集成MAC,PHY采用独立芯片;
3. CPU不集成MAC与PHY,MAC与PHY采用集成芯片;

本例中选用方案二做进一步说明,因为CPU总线接口很常见,通常都会做成可以像访问内存一样去访问,没必要拿出来说,而Mac与PHY之间的MII接口则需要多做些说明.

下图是采用方案二的网口结构图.虚框表示CPU,MAC集成在CPU中.PHY芯片通过MII接口与CPU上的Mac连接.



在软件上对网口的操作通常分为下面几步:

1. 为数据收发分配内存;
2. 初始化MAC寄存器;
3. 初始化PHY寄存器（通过MIIM）;
4. 启动收发;

**2. MII**

MII接口是MAC与PHY连接的标准接口.因为各厂家采用了同样的接口,用户可以根据所需的性能、价格,采用不同型号,甚至不同公司的phy芯片.

需要发送的数据通过MII接口中的收发两组总线实现.而对PHY芯片寄存器的配置信息,则通过MII总的一组串口总线实现,即MIIM（MII Management）.

下表列出了MII总线中主要的一些引脚

|  |  |  |
| --- | --- | --- |
| **PIN Name** | **Direction** | **Description** |
| TXD[0:3] | Mac to Phy | Transmit Data |
| TXEN | Mac to Phy | Transmit Enable |
| TXCLK | Mac to Phy | Transmit Clock |
| RXD[0:3] | Phy to Mac | Receive Data |
| RXEN | Phy to Mac | Receive Enable |
| RXCLK | Phy to Mac | Receive Clock |
| MDC | Mac to Phy | Management Data Clock |
| MDIO | Bidirection | Management Data I/O |

MIIM只有两个线, 时钟信号MDC与数据线MDIO.读写命令均由Mac发起, PHY不能通过MIIM主动向Mac发送信息.由于MIIM只能有Mac发起, 我们可以操作的也就只有MAC上的寄存器.

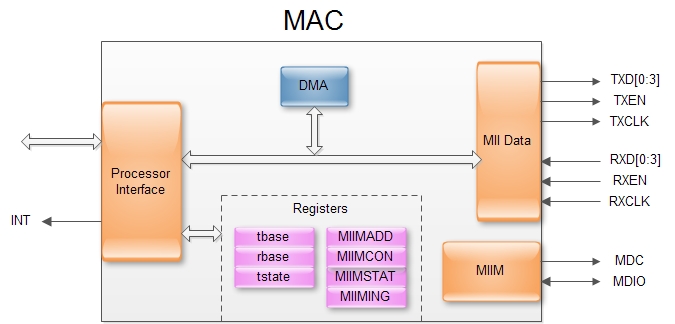
**3. DMA**

收发数据总是间费时费力的事,尤其对于网络设备来说更是如此.CPU做这些事情显然不合适.既然是数据搬移, 最简单的办法当然是让DMA来做.毕竟专业的才是最好的.

这样CPU要做的事情就简单了.只需要告诉DMA起始地址与长度, 剩下的事情就会自动完成.

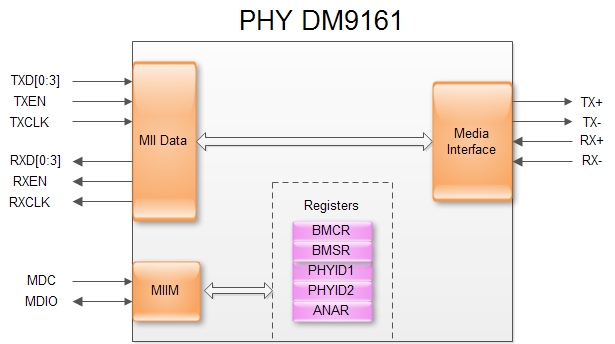
通常在MAC中会有一组寄存器专门用户记录数据地址, tbase与rbase, cpu按MAC要的格式把数据放好后, 启动MAC的数据发送就可以了.启动过程常会用到寄存器tstate.

**4. MAC**



CPU上有两组寄存器用与MAC.一组用户数据的收发,对应上面的DMA;一组用户MIIM,用户对PHY进行配置.两组寄存器由于都在CPU上,配置方式与其他CPU上寄存器一样,直接读写即可.数据的转发通过DMA完成.

**5. PHY**



该芯片是一个10M/100M Ethernet网口芯片

PHY芯片有一组寄存器用户保存配置,并更新状态.CPU不能直接访问这组寄存器,只能通过MAC上的MIIM寄存器组实现间接访问.同时PHY芯片负责完成MII总线的数据与Media Interface上数据的转发.该转发根据寄存器配置自动完成,不需要外接干预.

[**网口扫盲三:以太网芯片MAC和PHY的关系**](http://www.cnblogs.com/jason-lu/p/3195473.html)

### 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即Media Access Control,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.该层协议是以太网MAC由IEEE-802.3以太网标准定义.最新的MAC同时支持10Mbps和100Mbps两种速率.

以太网数据链路层其实包含MAC(介质访问控制)子层和LLC(逻辑链路控制)子层.一块以太网卡MAC芯片的作用不但要实现MAC子层和LLC子层的功能,还要提供符合规范的PCI界面以实现和主机的数据交换.

MAC从PCI总线收到IP数据包(或者其他网络层协议的数据包)后,将之拆分并重新打包成最大1518Byte,最小64Byte的帧.这个帧里面包括了目标MAC地址、自己的源MAC地址和数据包里面的协议类型(比如IP数据包的类型用80表示).最后还有一个DWORD(4Byte)的CRC码.

可是目标的MAC地址是哪里来的呢?这牵扯到一个ARP协议(介乎于网络层和数据链路层的一个协议).第一次传送某个目的IP地址的数据的时候,先会发出一个ARP包,其MAC的目标地址是广播地址,里面说到：”谁是xxx.xxx.xxx.xxx这个IP地址的主人？”因为是广播包,所有这个局域网的主机都收到了这个ARP请求.收到请求的主机将这个IP地址和自己的相比较,如果不相同就不予理会,如果相同就发出ARP响应包.这个IP地址的主机收到这个ARP请求包后回复的ARP响应里说到:”我是这个IP地址的主人”.这个包里面就包括了他的MAC地址.以后的给这个IP地址的帧的目标MAC地址就被确定了.(其它的协议如IPX/SPX也有相应的协议完成这些操作.)

IP地址和MAC地址之间的关联关系保存在主机系统里面,叫做ARP表,由驱动程序和操作系统完成.在Microsoft的系统里面可以用arp-a的命令查看ARP表.收到数据帧的时候也是一样,做完CRC以后,如果没有CRC效验错误,就把帧头去掉,把数据包拿出来通过标准的借口传递给驱动和上层的协议客栈,最终正确的达到我们的应用程序.

还有一些控制帧,例如流控帧也需要MAC直接识别并执行相应的行为.

以太网MAC芯片的一端接计算机PCI总线,另外一端就接到PHY芯片上,它们之间是通过MII接口链接的.

**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."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.它包括一个数据接口,以及一个MAC和PHY之间的管理接口.

* 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,包括TX\_ER,TXD<3:0>,TX\_EN,TX\_CLK, COL,RXD<3:0>,RX\_EX,RX\_CLK,CRS,RX\_DV等.MII以4位半字节方式传送数据双向传输,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.其管理是使用SMI(Serial Management Interface)总线通过读写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.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.

MII支持10Mbps和100Mbps的操作,一个接口由14根线组成,它的支持还是比较灵活的,但是有一个缺点是因为它一个端口用的信号线太多,如果一个8端口的交换机要用到112根线,16端口就要用到224根线,到32端口的话就要用到448根线,一般按照这个接口做交换机,是不太现实的,所以现代的交换机的制作都会用到其它的一些从MII简化出来的标准,比如RMII,SMII,GMII等.

RMII是简化的MII接口,在数据的收发上它比MII接口少了一倍的信号线,所以它一般要求是50MHz的总线时钟.RMII一般用在多端口的交换机,它不是每个端口安排收,发两个时钟,而是所有的数据端口公用一个时钟用于所有端口的收发,这里就节省了不少的端口数目.RMII的一个端口要求7个数据线,比MII少了一倍,所以交换机能够接入多一倍数据的端口.和MII一样,RMII支持10Mbps和100Mbps的总线接口速度.

SMII是由思科提出的一种媒体接口,它有比RMII更少的信号线数目,S表示串行的意思.因为它只用一根信号线传送发送数据,一根信号线传输接受数据,所以为了满足100Mbps的总线接口速度的需求,它的时钟频率就达到了125MHz,为什么用125MHz,是因为数据线里面会传送一些控制信息.SMII一个端口仅用4根信号线完成100Mbps的传输,比起RMII差不多又少了一倍的信号线.SMII在工业界的支持力度是很高的.同理,所有端口的数据收发都公用同一个外部的125MHz时钟.

GMII是千兆网的MII接口,这个也有相应的RGMII接口,表示简化了的GMII接口.

**MII总线**

在IEEE802.3中规定的MII总线是一种用于将不同类型的PHY与相同网络控制器(MAC)相连接的通用总线.网络控制器可以用同样的硬件接口与任何PHY .

**GMII(Gigabit MII)**

GMII采用8位接口数据,工作时钟125MHz,因此传输速率可达1000Mbps.同时兼容MII所规定的10/100 Mbps工作方式.

GMII接口数据结构符合IEEE以太网标准.该接口定义见IEEE 802.3-2000.

**发送器**:

* GTXCLK——吉比特TX..信号的时钟信号(125MHz)
* TXCLK——10/100Mbps信号时钟
* TXD[7..0]——被发送数据
* TXEN——发送器使能信号
* TXER——发送器错误(用于破坏一个数据包)

注:在千兆速率下,向PHY提供GTXCLK信号,TXD,TXEN,TXER信号与此时钟信号同步.否则,在10/100Mbps速率下,PHY提供TXCLK时钟信号,其它信号与此信号同步.其工作频率为25MHz(100M网络)或2.5MHz(10M网络).

**接收器:**

* RXCLK——接收时钟信号(从收到的数据中提取,因此与GTXCLK无关联)
* RXD[7..0]——接收数据
* RXDV——接收数据有效指示
* RXER——接收数据出错指示
* COL——冲突检测(仅用于半双工状态)

**管理配置**

* MDC——配置接口时钟
* MDIO——配置接口I/O

管理配置接口控制PHY的特性.该接口有32个寄存器地址,每个地址16位.其中前16个已经在"IEEE 802.3,2000-22.2.4 Management Functions"中规定了用途,其余的则由各器件自己指定.

**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.

RMII口是用两根线来传输数据的,MII口是用4根线来传输数据的,GMII是用8根线来传输数据的.MII/RMII只是一种接口,对于10Mbps线速,MII的时钟速率是2.5MHz就可以了,RMII则需要5MHz;对于100Mbps线速,MII需要的时钟速率是25MHz,RMII则是50MHz.  
  
MII/RMII用于传输以太网包,在MII/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!

**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.包括MII/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.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.

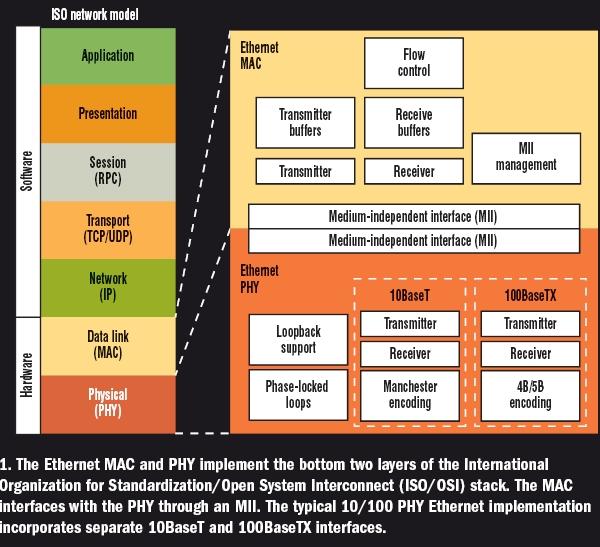
PHY在发送数据的时候,收到MAC过来的数据(对PHY来说,没有帧的概念,对它来说,都是数据而不管什么地址,数据还是CRC.对于100BaseTX因为使用4B/5B编码,每4bit就增加1bit的检错码),然后把并行数据转化为串行流数据,再按照物理层的编码规则把数据编码,再变为模拟信号把数据送出去.收数据时的流程反之.PHY还有个重要的功能就是实现CSMA/CD的部分功能.它可以检测到网络上是否有数据在传送,如果有数据在传送中就等待,一旦检测到网络空闲,再等待一个随机时间后将送数据出去.如果两个碰巧同时送出了数据,那样必将造成冲突,这时候,冲突检测机构可以检测到冲突,然后各等待一个随机的时间重新发送数据.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.

许多网友在接入Internt宽带时,喜欢使用”抢线”强的网卡,就是因为不同的PHY碰撞后计算随机时间的方法设计上不同,使得有些网卡比较”占便宜”.不过,抢线只对广播域的网络而言的,对于交换网络和ADSL这样点到点连接到局端设备的接入方式没什么意义.而且”抢线”也只是相对而言的,不会有质的变化.

现在交换机的普及使得交换网络的普及,使得冲突域网络少了很多,极大地提高了网络的带宽.但是如果用HUB,或者共享带宽接入Internet的时候还是属于冲突域网络,有冲突碰撞的.交换机和HUB最大的区别就是:一个是构建点到点网络的局域网交换设备,一个是构建冲突域网络的局域网互连设备.

除此之外PHY还提供了和对端设备连接的重要功能并通过LED灯显示出自己目前的连接的状态和工作状态让我们知道.当我们给网卡接入网线的时候,PHY不断发出的脉冲信号检测到对端有设备,它们通过标准的”语言”交流,互相协商并却定连接速度、双工模式、是否采用流控等.通常情况下,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AutoNegotiation或者NWAY,它们是一个意思–自动协商.

具体传输过程为,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.



**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: 网卡上除RJ-45接口外,还需要其它元件吗?**

答:PHY和MAC是网卡的主要组成部分,网卡一般用RJ-45插口,10M网卡的RJ-45插口也只用了1,2,3,6四根针,而100M或1000M网卡的则是八根针都是全的.除此以外,还需要其它元件,因为虽然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.

另外,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.

为了解决以上问题Transformer(**隔离变压器**)这个器件就应运而生.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.

隔离变压器本身是个被动元件,只是把PHY的信号耦合了到网线上,并没有起到功率放大的作用.那么一张网卡信号的传输的最长距离是谁决定的呢?

一张网卡的传输最大距离和与对端设备连接的兼容性主要是PHY决定的.但是可以将信号送的超过100米的PHY其输出的功率也比较大,更容易产生EMI的问题.这时候就需要合适的Transformer与之配合.作PHY的老大公司Marvell的PHY,常常可以传送180~200米的距离,远远超过IEEE的100米的标准.

RJ-45的接头实现了网卡和网线的连接.它里面有8个铜片可以和网线中的4对双绞(8根)线对应连接.其中100M的网络中1,2是传送数据的,3,6是接收数据的.1,2之间是一对差分信号,也就是说它们的波形一样,但是相位相差180度,同一时刻的电压幅度互为正负.这样的信号可以传递的更远,抗干扰能力强.同样的,3,6也一样是差分信号.

网线中的8根线,每两根扭在一起成为一对.我们制作网线的时候,一定要注意要让1,2在其中的一对,3,6在一对.否则长距离情况下使用这根网线的时候会导致无法连接或连接很不稳定.

现在新的PHY支持AUTO MDI-X功能(也需要Transformer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.有的PHY甚至支持一对线中的正信号和负信号的功能自动交换.这样我们就不必为了到底连接某个设备需要使用直通网线还是交叉网线而费心了.这项技术已经被广泛的应用在交换机和SOHO路由器上.

在1000Basd-T网络中,其中最普遍的一种传输方式是使用网线中所有的4对双绞线,其中增加了4,5和7,8来共同传送接收数据.由于1000Based-T网络的规范包含了AUTOMDI-X功能,因此不能严格确定它们的传出或接收的关系,要看双方的具体的协商结果.

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.

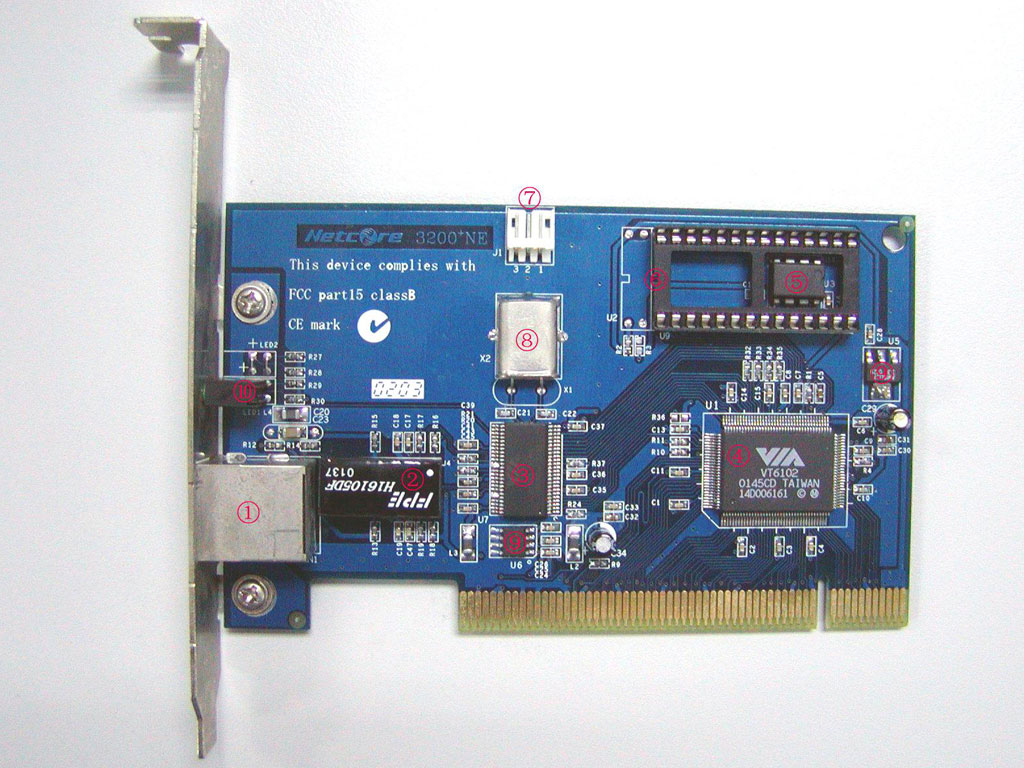
很多网卡上还有BOOTROM这个东西.它是用于无盘工作站引导操作系统的.既然无盘,一些引导用必需用到的程序和协议栈就放到里面了,例如RPL,PXE等.实际上它就是一个标准的PCI ROM.所以才会有一些硬盘写保护卡可以通过烧写网卡的BootRom来实现.其实PCI设备的ROM是可以放到主板BIOS里面的.启动电脑的时候一样可以检测到这个ROM并且正确识别它是什么设备的.AGP在配置上和PCI很多地方一样,所以很多显卡的BIOS也可以放到主板BIOS里面.这就是为什么板载的网卡我们从来没有看到过BOOTROM的原因.

最后就是电源部分了.大多数网卡现在都使用3.3V或更低的电压.有的是双电压的.因此需要电源转换电路.

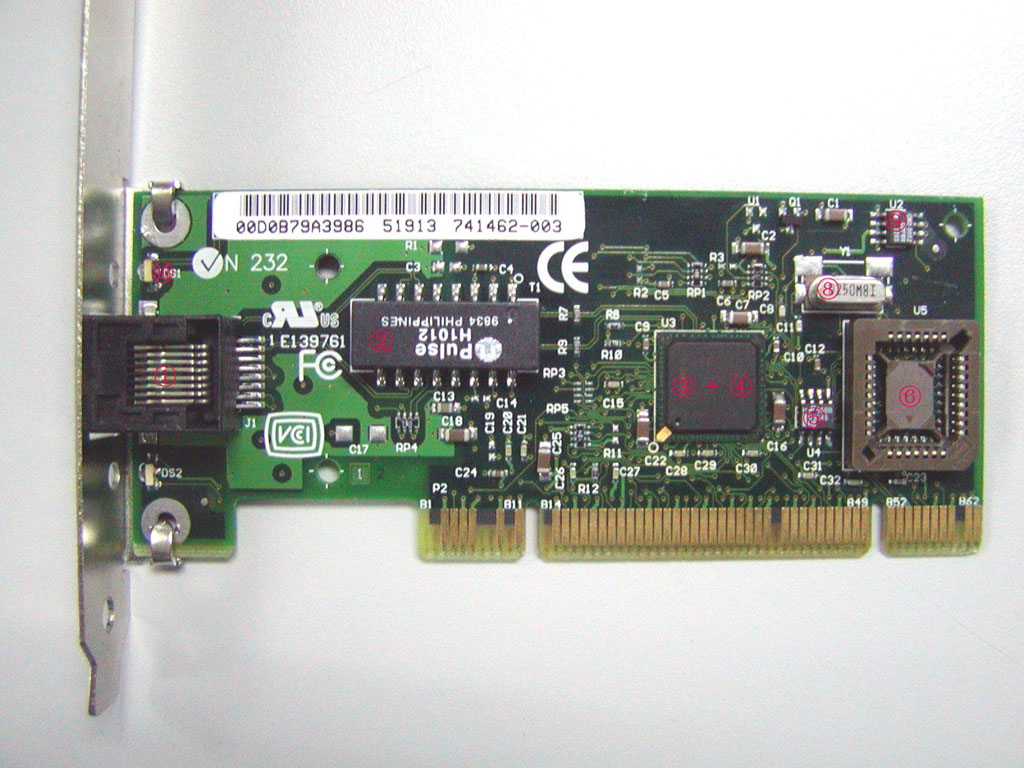
而且网卡为了实现Wake on line功能,必须保证全部的PHY和MAC的极少一部分始终处于有电的状态,这需要把主板上的5V Standby电压转换为PHY工作电压的电路.在主机开机后,PHY的工作电压应该被从5V转出来的电压替代以节省5V Standby的消耗.(许多劣质网卡没有这么做).

有Wake on line功能的网卡一般还有一个WOL的接口.那是因为PCI2.1以前没有PCI设备唤醒主机的功能,所以需要着一根线通过主板上的WOL的接口连到南桥里面以实现WOL的功能.新的主板合网卡一般支持PCI2.2/2.3,扩展了PME#信号功能,不需要那个接口而通过PCI总线就可以实现唤醒功能.

我们现在来看两个图



MAC和PHY分开的以太网卡



MAC和PHY集成在一颗芯片的以太网卡

上图中各部件为:

①RJ-45接口

②Transformer(隔离变压器)

③PHY芯片

④MAC芯片

⑤EEPROM

⑥BOOTROM插槽

⑦WOL接头

⑧晶振

⑨电压转换芯片

⑩LED指示灯

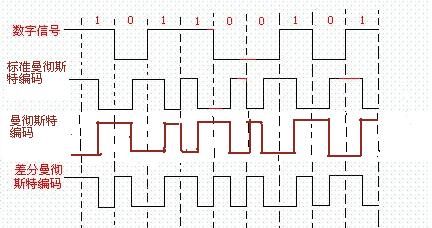
网卡的功能主要有两个:一是将电脑的数据封装为帧,并通过网线(对无线网络来说就是电磁波)将数据发送到网络上去;二是接收网络上其它设备传过来的帧,并将帧重新组合成数据,发送到所在的电脑中.网卡能接收所有在网络上传输的信号,但正常情况下只接受发送到该电脑的帧和广播帧,将其余的帧丢弃.然后,传送到系统CPU做进一步处理.当电脑发送数据时,网卡等待合适的时间将分组插入到数据流中.接收系统通知电脑消息是否完整地到达,如果出现问题,将要求对方重新发送.

**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.

**问:网卡的MAC和PHY间的关系?**

答: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).

PHY和MAC之间是如何传送数据和相互沟通的.通过IEEE定义的标准的MII/GigaMII(Media Independed 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.ETHERNET的接口实质是MAC通过MII总线控制PHY的过程.

**问:网线上传输的是模拟信号还是数字信号?**

答:是模拟信号.因为它传出和接收是采用的模拟的技术.虽然它传送的信息是数字的(并不是传送的信息是数字的信号就可以叫做数字信号).

简单的例子:我们知道电话是模拟信号,但是当我们拨号上网的时候,电话线里传送的是数字信息,但信号本身依旧是模拟的.然而ADSL同样是通过电话线传送的,却是数字信号.这取决于它传出和接受采用的技术.

**问:若操作系统没有加载网卡驱动,网卡虽然在系统设备树上,但网卡接口创建不了,那网卡实际能不能接收到数据?**

答:这里面有很多细节, 我根据Intel网卡的Spec大概写了写, 想尽量写的通俗一些,所以没有刻意用Spec里的术语,另外本文虽然讲的是MAC/PHY,但光口卡的(SERDES)也是类似的.

1. PCI设备做reset以后进入D0uninitialized(非初始化的D0状态, 参考PCI电源管理规范),此时网卡的MAC和DMA都不工作,PHY是工作在一个特殊的低电源状态的;
2. 操作系统创建设备树时,初始化这个设备,PCI命令寄存器的 Memory Access Enable or the I/O Access Enable bit会被enable, 这就是D0active.此时PHY/MAC就使能了;
3. PHY被使能应该就可以接收物理链路上的数据了,否则不能收到FLP/NLP, PHY就不能建立物理连接.但这类包一般是流量间歇发送的;
4. 驱动程序一般要通过寄存器来控制PHY, 比如自动协商speed/duplex, 查询物理链路的状态Link up/down;
5. MAC被使能后, 如果没有驱动设置控制寄存器的一个位(CTRL.SLU )的话, MAC和PHY是不能通讯的, 就是说MAC不知道PHY的link已经ready, 所以收不到任何数据的.这位设置以后, PHY完成自协商, 网卡才会有个Link change的中断,知道物理连接已经Link UP了;
6. 即使Link已经UP, MAC还需要enable接收器的一个位(RCTL.RXEN ),包才可以被接收进来,如果网卡被reset,这位是0,意味着所有的包都会被直接drop掉,不会存入网卡的 FIFO.老网卡在驱动退出前利用这位关掉接收.Intel的最新千兆网卡发送接收队列的动态配置就是依靠这个位的,重新配置的过程一定要关掉流量;
7. 无论驱动加载与否, 发生reset后,网卡EEPOM里的mac地址会写入网卡的MAC地址过滤寄存器, 驱动可以去修改这个寄存器,现代网卡通常支持很多MAC地址,也就是说,MAC地址是可以被软件设置的.例如,Intel的千兆网卡就支持16个单播 MAC地址,但只有1个是存在EEPROM里的,其它是软件声称和设置的;
8. 但如果驱动没有加载,网卡已经在设备树上,操作系统完成了步骤1-2的初始化,此时网卡的PHY应该是工作的,但因为没有人设置控制位(CTRL.SLU)来让MAC和PHY建立联系,所以MAC是不收包的.这个控制位在reset时会再设置成0;
9. PHY可以被软件设置加电和断电, 断电状态除了接收管理命令以外,不会接收数据.另外,PHY还能工作在Smart Power Down模式下,link down就进入省电状态;
10. 有些多口网卡,多个网口共享一个PHY, 所以BIOS里设置disbale了某个网口, 也未必会把PHY的电源关掉,反过来,也要小心地关掉PHY的电源;
11. 要详细了解PHY,最终还是要熟悉IEEE以太网的相关协议.

本文引用:<http://blog.csdn.net/woodstar123/article/details/3324368>

<http://blog.csdn.net/yayong/article/details/5334565>