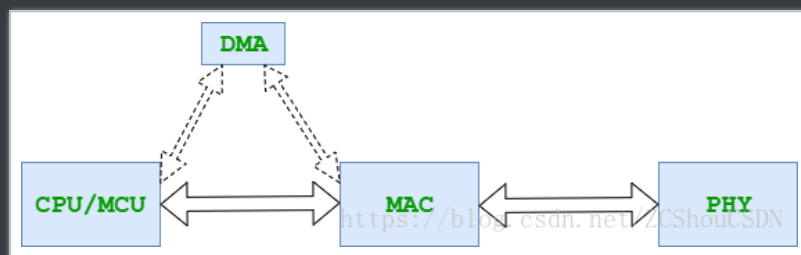


Network 之二 Ethernet (以太网) 中的 MAC、MII、PHY 详解 ethernet phy-CSDN 博客

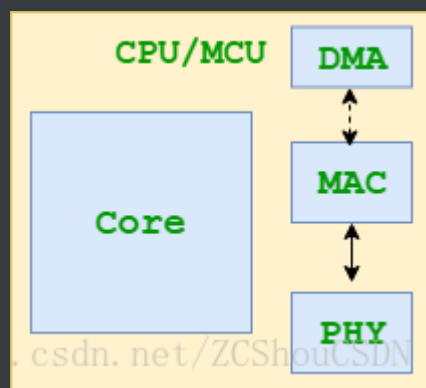
结构

从硬件的角度看，以太网接口电路主要由 MAC (Media Access Control, MAC) 控制器和物理层接口 PHY (Physical Layer, PHY) 两大部分构成。如下图所示：

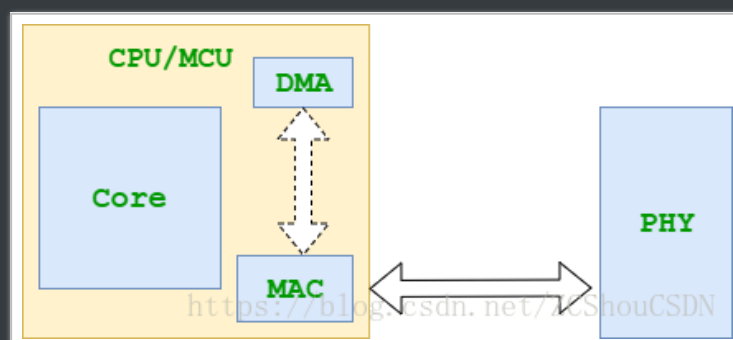


但是，在实际的设计中，以上三部分并不一定是独立分开的。由于，PHY 整合了大量模拟硬件，而 MAC 则是典型的全数字器件。考虑到芯片面积及模拟 / 数字混合架构的原因，通常，将 MAC 集成进微控制器而将 PHY 留在片外（现在，更灵活、密度更高的芯片技术已经可以实现 MAC 和 PHY 的单芯片整合）。

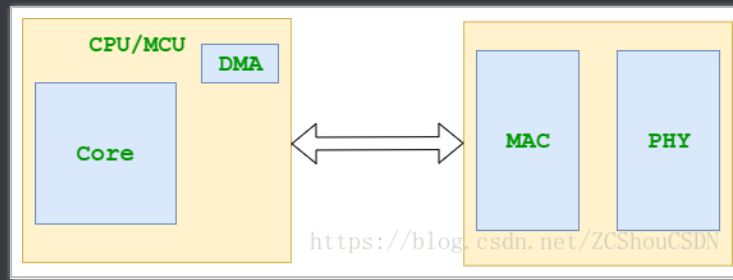
- CPU 集成 MAC 与 PHY，目前来说并不多见。



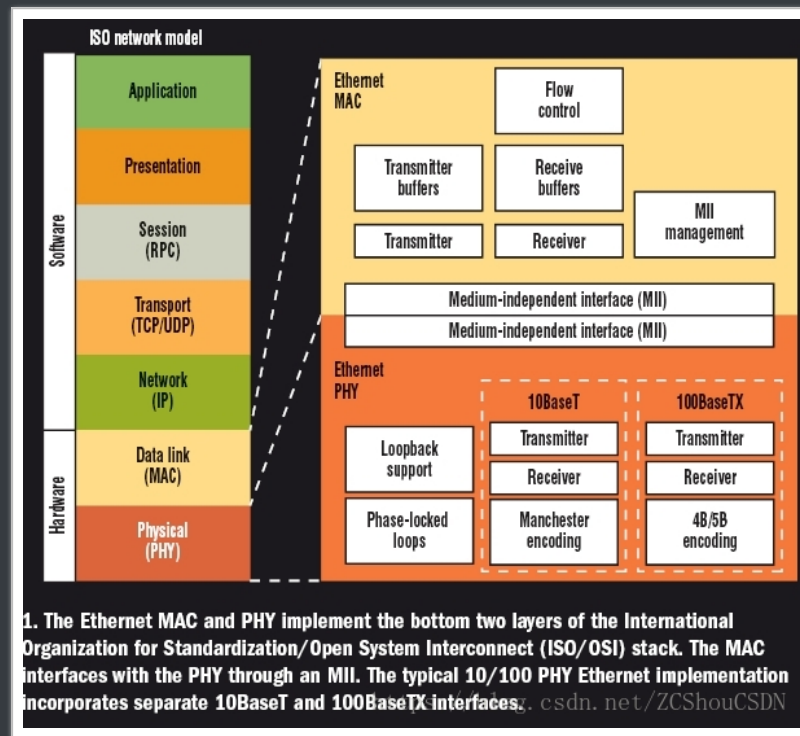
- CPU 集成 MAC，PHY 采用独立芯片，这种比较常见。



- CPU 不集成 MAC 与 PHY, MAC 与 PHY 采用集成芯片 (形成一个独立的网卡), 这种也比较常见。

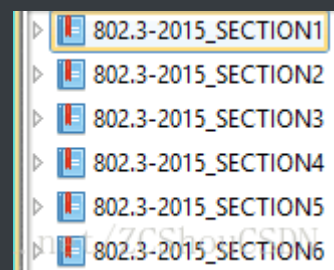


MAC 及 PHY 工作在 OSI 七层模型的数据链路层和 物理层。
具体如下



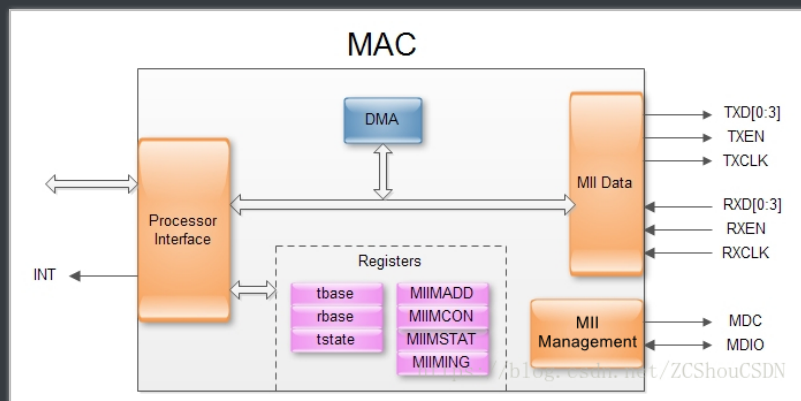
IEEE802.3 标准文档下载地址:

<https://ieeexplore.ieee.org/browse/standards/get-program/page/>。它分为很多个 section。标准文档贼长, 没有全部看过, 大概第一部分是 10Mb/s 以太网 (其中有关于 MAC 的规范), 第二部分是 100BASE-T 100Mb/s 基带网络介绍, 第三部分是 1000 Mb/s 基带网络介绍, 第四部分是 10 Gb/s 基带网络介绍。但是, 并不是说每部分都是独立的。



什么是 MAC

MAC (Media Access Control, MAC) 即媒体访问控制。该部分有两个概念：**MAC 可以指的是一个硬件控制器，也可以指 MAC 通信协议**。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。MAC 硬件大约就是下面的样子了：



关于协议，详见 [Network 之五 TCP/IP 协议族、工作流程、\(TCP、IP、MAC、VLAN\) 常用协议格式详解](#)

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（逻辑链路控制）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 Windows 及 Linux 里面可以用 `arp -a` 的命令查看 ARP 表。

```
PS C:\Users\ZCShou> arp -a
Interface: 192.168.100.214 --- 0x5
Internet Address      Physical Address      Type
192.168.100.1          38-ad-be-a4-c1-05     dynamic
192.168.100.7          a0-ae-11-17-21-9b     dynamic
192.168.100.66         6c-4b-90-f7-d0-70     dynamic
192.168.100.207        00-0c-29-9e-a0-89     dynamic
192.168.100.255        ff-ff-ff-ff-ff-ff     static
224.0.0.22            01-00-5e-00-00-16     static
224.0.0.251           01-00-5e-00-00-fb     static
224.0.0.252           01-00-5e-00-00-fc     static
228.100.100.3          01-00-5e-12-b8-03     static
239.255.255.250       01-00-5e-7f-ff-fa     static
255.255.255.255       ff-ff-ff-ff-ff-ff     static

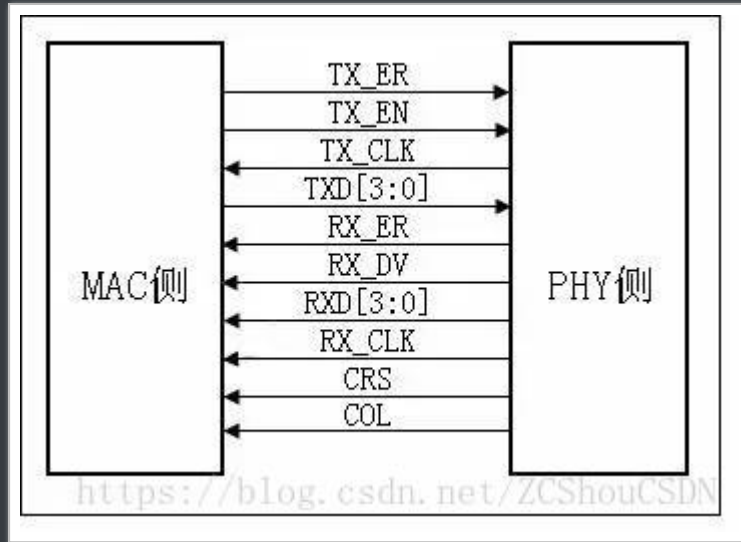
PS C:\Users\ZCShou>

CSDN @ZC-Shou
```

收到数据帧的时候也是一样，做完 CRC 校验以后，如果没有 CRC 校验错误，就把帧头去掉，把数据包拿出来通过标准的接口传递给驱动和上层的协议栈。最终正确的达到我们的应用程序。

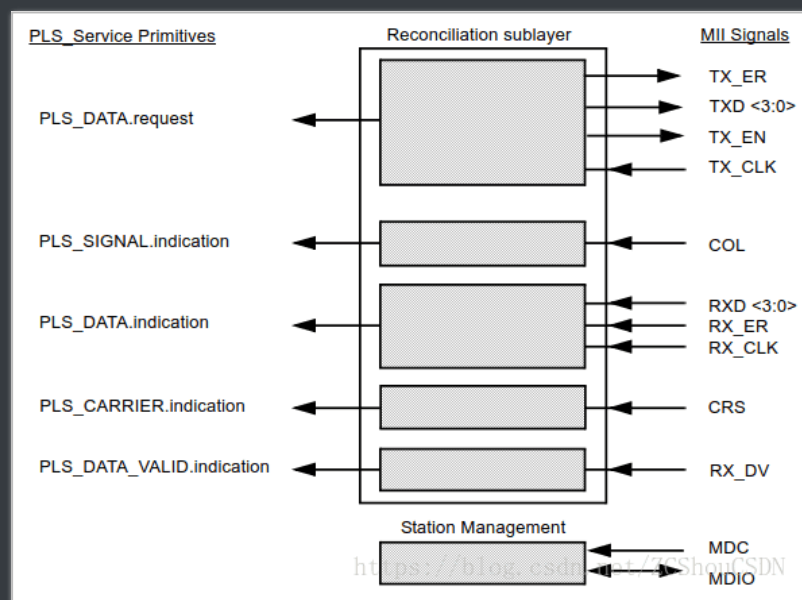
什么是 MII

II (Media Independent Interface) 即媒体独立接口，II 接口是 MAC 与 PHY 连接的标准接口。它是 IEEE-802.3 定义的以太网行业标准。II 接口提供了 MAC 与 PHY 之间、PHY 与 STA (Station Management) 之间的互联技术，该接口支持 10Mb/s 与 100Mb/s 的数据传输速率，数据传输的位宽为 4 位。II 接口如下图所示：



“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。802.3 协议最多支持 32 个 PHY，但有一定的限制：要符合协议要求的 connector 特性。此外，提到 MII，就有可能涉及到 RS，PLS，STA 等名词术语，下面讲一下他们之间对应的关系。

所谓 RS 即 Reconciliation sublayer，它的主要功能主要是提供一种 MII 和 MAC/PLS 之间的信号映射机制。它们 (RS 与 MII) 之间的关系如下图：



MII 的 Management Interface 是与 STA (Station Management) 相连的。

关于本节，具体可参考 IEEE 以太网标准 802.3 的 22.3 Signal timing characteristics 节，其中包含时钟

信号等更详细内容。

MII 接口主要包括四个部分。一是从 MAC 层到 PHY 层的发送数据接口，二是从 PHY 层到 MAC 层的接收数据接口，三是从 PHY 层到 MAC 层的状态指示信号，四是 MAC 层和 PHY 层之间传送控制和状态信息的 MDIO 接口。

| MII接口的MAC模式定义 | | | | | MII接口PHY模式定义 | | | | |
|-----------------|----|-----------|-----------------------|------------------|-----------------|----|-----------|-----------------------|------------------|
| 信号名称 | 数量 | Direction | Description | Belong To | 信号名称 | 数量 | Direction | Description | Belong To |
| TX_CLK | 1 | Input | Transmit Clock | MAC 层到物理层的发送数据接口 | TX_CLK | 1 | Output | Transmit Clock | MAC 层到物理层的接收数据接口 |
| TXD[0:3] | 4 | Output | Transmit Data | | TXD[0:3] | 4 | Input | Transmit Data | |
| TX_ER | 1 | Output | Transmit Coding Error | | TX_ER | 1 | Input | Transmit Coding Error | |
| TX_EN | 1 | Output | Transmit Enable | | TX_EN | 1 | Input | Transmit Enable | |
| RX_CLK | 1 | Input | Receive Clock | 物理层到MAC层的接收数据接口 | RX_CLK | 1 | Output | Receive Clock | 物理层到MAC层的接收数据接口 |
| RXD[0:3] | 4 | Input | Receive Data | | RXD[0:3] | 4 | Output | Receive Data | |
| RX_ER | 1 | Input | Receive Error | | RX_ER | 1 | Output | Receive Error | |
| RX_DV | 1 | Input | Receive Data Valid | | RX_DV | 1 | Output | Receive Data Valid | |
| COL | 1 | Input | Collision Detected | 物理层到MAC层的状态指示信号 | COL | 1 | Output | Collision Detected | 物理层到MAC层的状态指示信号 |
| CRS | 1 | Input | Carrier Sense | | CRS | 1 | Output | Carrier Sense | |
| Total bus Width | 16 | | | | Total bus Width | 16 | | | |
| MDC | 1 | Output | Management Clock | MAC和物理层控制状态信息 | MDC | 1 | Input | Management Clock | MAC和物理层控制状态信息 |
| MDIO | 1 | IO | Management Data I/O | | MDIO | 1 | IO | Management Data I/O | |

MII 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 (transmit coding error) :** TX_ER 同步于 TX_CLK，在数据传输过程中，如果 TX_ER 有效超过一个时钟周期，并且此时 TX_EN 是有效的，则数据通道中传输的数据是无效的，没用的。注：当 TX_ER 有效并不影响工作在 10Mb/s 的 PHY 或者 TX_EN 无效时的数据传输。在 MII 接口的连线中，如果 TX_ER 信号线没有用到，必须将它下拉接地。
- **TXD<3:0> (transmit data) :** TXD 由 RS 驱动，同步于 TX_CLK，在 TX_CLK 的时钟周期内，并且 TX_EN 有效，TXD 上的数据被 PHY 接收，否则 TXD 的数据对 PHY 没有任何影响。
- **TX_EN：** 发送使能。TX_EN 由 Reconciliation 子层根据 TX_CLK 上升沿同步进行转换。
- **TX_CLK (transmit clock) :** TX_CLK (Transmit Clock) 是一个连续的时钟信号（即系统启动，该信号就一直存在），它是 TX_EN、TXD、TX_ER（信号方向为从 RS 到 PHY）的参考时钟，TX_CLK 由 PHY 驱动 TX_CLK 的时钟频率是数据传输速率的 25%，偏差 $\pm 100\text{ppm}$ 。例如，100Mb/s 模式下，TX_CLK 时钟频率为 25MHz，占空比在 35% 至 65% 之间。

- **COL (collision detected)** : COL 不需要同步于参考时钟。The behavior of the COL signal is unspecified when the duplex mode bit0.8 in the control register is set to a logic one (自动协商禁止, 人工设为全双工模式), or when the Auto-Negotiation process selects a full duplex mode of operation。即半双工模式信号有效, 全双工模式信号无效。
 - **RXD<3:0> (receive data)** : RXD 由 RS 驱动, 同步于 RX_CLK, 在 RX_CLK 的时钟周期内, 并且 RX_DV 有效, RXD 上的数据被 RS 接收, 否则 RXD 的数据对 RS 没有任何影响。While RX_DV is de-asserted, the PHY may provide a False Carrier indication by asserting the RX_ER signal while driving the value <1110> onto RXD<3:0>。
 - **RX_ER (receive error)** : RX_ER 同步于 RX_CLK, 其在 RX 通道中的作用类似于 TX_ER 对于 TX 通道数据传输的影响。
 - **RX_CLK**: 它与 TX_CLK 具有相同的要求, 所不同的是它是 RX_DV、RXD、RX_ER (信号方向是从 PHY 到 RS) 的参考时钟。RX_CLK 同样是由 PHY 驱动, PHY 可能从接收到的数据中提取时钟 RX_CLK, 也有可能从一个名义上的参考时钟 (e.g., the TX_CLK reference) 来驱动 RX_CLK。
 - **CRS (carrier sense)** : CRS 不需要同步于参考时钟, 只要通道存在发送或者接收过程, CRS 就需要有效。The behavior of the CRS signal is unspecified when the duplex mode bit0.8 in the control register is set to a logic one(自动协商禁止, 人工设为全双工模式), or when the Auto-Negotiation process selects a full duplex mode of operation, 即半双工模式信号有效, 全双工模式信号无效。
 - **RX_DV (Receive Data Valid)** : RXD_DV 同步于 RX_CLK, 被 PHY 驱动, 它的作用如同于发送通道中的 TX_EN, 不同的是在时序上稍有一点差别: 为了让数据能够成功被 RS 接收, 要求 RXD_DV 有效的时间必须覆盖整个 FRAME 的过程, 即 starting no later than the Start Frame Delimiter (SFD) and excluding any End-of-Frame delimiter。
- 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，通过管理接口，MAC 就能监视和控制 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。一个是时钟信号 (**MDC (management data clock)**) 。另一个是数据信号 (**MDIO (management data input/output)**) 。
- **MDC：** 由站管理实体向 PHY 提供，作为在 MDIO 信号上传送信息的定时参考。MDC 是一种非周期性的信号，没有最高或最低时间。无论 TX_CLK 和 RX_CLK 的标称周期如何，MDC 的最小高低时间应为 160 ns，MDC 的最小周期为 400 ns。
- **MDIO：** 是 PHY 和 STA 之间的双向信号。它用于在 PHY 和 STA 之间传输控制信息和状态。控制信息由 STA 同步地针对 MDC 驱动并且由 PHY 同步地采样。状态信息由 PHY 针对 MDC 同步驱动并由 STA 同步采样。

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由 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。

II 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 II 简化出来的标准，比如 RMII、SMII、GMII 等。

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 II 有更少的 I/O 传输。RMII 口是用两根线来传输数据的，II 口是用 4 根线来传输数据的，GMII 是用 8 根线来传输数据的。II/RMII 只是一种接口，对于 10Mbps 线速，II 的时钟速率是 2.5MHz 就可以了，RMII 则需要 5MHz；对于 100Mbps 线速，II 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!

GMII(Gigabit MII)

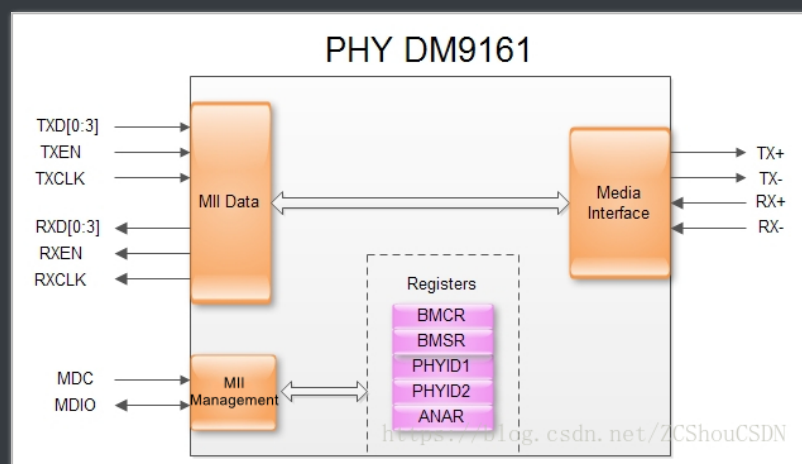
GMII 是千兆网的 MII 接口, 这个也有相应的 RGMII 接口, 表示简化了的 GMII 接口。GMII 采用 8 位接口数据, 工作时钟 125MHz, 因此传输速率可达 1000Mbps。同时兼容 MII 所规定的 10/100 Mbps 工作方式。GMII 接口数据结构符合 IEEE 以太网标准, 该接口定义见 IEEE 802.3-2000。

- **发送器:** 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)。
- 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)
- **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：** 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。
- MDC：配置接口时钟
- MDIO：配置接口 I/O

什么是 PHY

PHY (Physical Layer, PHY) 是 IEEE802.3 中定义的一个标准模块，STA (station management entity, 管理实体，一般为 MAC 或 CPU) 通过 SMI (Serial Manage Interface) 对 PHY 的行为、状态进行管理和控制，而具体管理和控制动作是通过读写 PHY 内部的寄存器实现的。一个 PHY 的基本结构如下图：



PHY 是物理接口收发器，它实现 OSI 模型的物理层。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 寄存器在 IEEE802.3 标准的 **22.2.4**

Management functions 节有介绍，但不涉及所有的寄存器，个别寄存器需要到其它章节中看，当然，文档里面也提到该在哪里找到哪个寄存器。

PHY 寄存器

PHY 寄存器的地址空间为 5 位, 从 0 到 31 最多可以定义 32 个寄存器 (随着芯片功能不断增加, 很多 PHY 芯片采用分页技术来扩展地址空间以定义更多的寄存器), IEEE802.3 定义了地址为 0-15 这 16 个寄存器的功能, 地址 16 - 31 的寄存器留给芯片制造商自由定义, 如下表所示。

| Register address | Register name | Basic/Extended | |
|------------------|--------------------------------------------------|----------------|----------|
| | | MII | GMII |
| 0 | Control | B | B |
| 1 | Status | B | B |
| 2,3 | PHY Identifier | E | E |
| 4 | Auto-Negotiation Advertisement | E | E |
| 5 | Auto-Negotiation Link Partner Base Page Ability | E | E |
| 6 | Auto-Negotiation Expansion | E | E |
| 7 | Auto-Negotiation Next Page Transmit | E | E |
| 8 | Auto-Negotiation Link Partner Received Next Page | E | E |
| 9 | MASTER-SLAVE Control Register | E | E |
| 10 | MASTER-SLAVE Status Register | E | E |
| 11 | PSE Control register | E | E |
| 12 | PSE Status register | E | E |
| 13 | MMD Access Control Register | E | E |
| 14 | MMD Access Address Data Register | E | E |
| 15 | Extended Status | Reserved | B |
| 16 through 31 | Vendor Specific | Reserved | Reserved |

1. 官方介绍请参考 IEEE802.3 标准的 22.2.4 Management functions 节。

2. 上图的 B 和 E 表示了, 在特定接口下, 寄存器是基本的还是扩展的。例如: MII 接口下只有 0 和 1 寄存器是基本的, 其它的是扩展的。**注意: 所有扩展是指留给 IEEE 以后的扩展特性用, 不是给 PHY 厂商的扩展, PHY 厂商自定义的只能是 16~31 号寄存器**

3. 在 IEEE 标准文档及某些 PHY 手册中, 某寄存器的比特 (bit) 用 $x.y$ 表示, 如 0.15 表示第 0 寄存器的第 15 位。

Control Register (Register 0)

寄存器 0 是 PHY 控制寄存器, 通过 Control Register 可以对 PHY 的主要工作状态进行设置。应该保证控制寄存器每个位的默认值, 以便在没有管理干预的情况下, 上电或复位时 PHY 的初始状态为正常操作状态。Control Register 的每一位完成的功能见下。

| Bit(s) | Name | Description | R/W ^a |
|--------|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| 0.15 | Reset | 1 = PHY reset 0 = normal operation | R/W SC |
| 0.14 | Loopback | 1 = enable loopback mode 0 = disable loopback mode | R/W |
| 0.13 | Speed Selection (LSB) | 0.6 0.13 1 1 = Reserved 1 0 = 1000 Mb/s 0 1 = 100 Mb/s 0 0 = 10 Mb/s | R/W |
| 0.12 | Auto-Negotiation Enable | 1 = enable Auto-Negotiation process 0 = disable Auto-Negotiation process | R/W |
| 0.11 | Power Down | 1 = power down 0 = normal operation ^b | R/W |
| 0.10 | Isolate | 1 = electrically Isolate PHY from MII or GMII 0 = normal operation ^b | R/W |
| 0.9 | Restart Auto-Negotiation | 1 = restart Auto-Negotiation process 0 = normal operation | R/W SC |
| 0.8 | Duplex Mode | 1 = full duplex 0 = half duplex | R/W |
| 0.7 | Collision Test | 1 = enable COL signal test 0 = disable COL signal test | R/W |
| 0.6 | Speed Selection (MSB) | 0.6 0.13 1 1 = Reserved 1 0 = 1000 Mb/s 0 1 = 100 Mb/s 0 0 = 10 Mb/s | R/W |
| 0.5 | Unidirectional enable | When bit 0.12 is one or bit 0.8 is zero, this bit is ignored. When bit 0.12 is zero and bit 0.8 is one: 1 = Enable transmit from media independent interface regardless of whether the PHY has determined that a valid link has been established 0 = Enable transmit from media independent interface only when the PHY has determined that a valid link has been established | R/W |
| 0.4:0 | Reserved | Write as 0, ignore on read | R/W |

^aR/W = Read/Write, SC = Self-clearing.
^bFor normal operation, both 0.10 and 0.11 must be cleared to zero; see 22.2.4.1.5.

<https://blog.csdn.net/ZCShouCSDN>

^aR/W = Read/Write, SC = Self-clearing.

^bFor normal operation, both 0.10 and 0.11 must be cleared to zero; see 22.2.4.1.5.

<https://blog.csdn.net/ZCShouCSDN>

- **Reset:** 通过将位 0.15 设置为逻辑 1 来完成复位 PHY。该操作应将状态和控制寄存器设置为其默认状态。因此，此操作可能会改变 PHY 的内部状态以及与 PHY 关联的物理链路的状态。**复位过程中 Bit15 保持为 1，复位完成之后该位应该自动清零。** 在复位过程完成之前，PHY 不需要接受对控制寄存器的写入操作，并且在复位过程完成之前写入 0.15 以外的控制寄存器位可能不起作用。复位过程应在 0.15 位设置的 0.5 s 内完成。

1. 一般要改变端口的工作模式（如速率、双工、流控或协商信息等）时，在设置完相应位置的寄存器之后，需要通过 Reset 位复位 PHY 来使配置生效。
2. 该比特位的默认值为 0。

- **Loopback:** 当位 0.14 被设置为逻辑 1 时，PHY 应置于环回操作模式。当位 0.14 置位时，PHY 接收电路应与网络介质隔离，并且 MII 或 GMII 处的 TX_EN 断言不应导致网络介质上的数据传输。当位 0.14 置位时，PHY 应接受来自 MII 或 GMII 发送数据路径的数据，并将其返回给 MII 或 GMII 接收数据路径，以响应 TX_EN 的断言。当位 0.14 置位时，从断言 TX_EN 到断言 RX_DV 的延迟应小于 512 BT。当位 0.14 置位时，除非设置了位 0.7，否则 COL 信号应始终保持无效。清 0.14 位为零允许正常操作。

Loopback 是一个调试以及故障诊断中常用的功能，Bit14

置 1 之后，PHY 和外部 MD 的连接在逻辑上将被断开，从 MAC 经过 MII/GMII（也可能是其他的 MAC/PHY 接口）发送过来的数据将不会被发送到 MDI 上，而是在 PHY 内部（一般在 PCS）回环到本端口的 MII/GMII 接收通道上，通过 Loopback 功能可以检查 MII/GMII 以及 PHY 接口部分是否工作正常，对于端口不通的情况可用于故障定位。

1. 需要注意的是，很多时候 PHY 设置 Loopback 后端口可能就 Link down 了，MAC 无法向该端口发帧，这时就需要通过设置端口 Force Link up 才能使用 Loopback 功能。
2. 该比特位的默认值为 0。

- **Speed Selection: Bit13 和 Bit6 两位联合实现** 对端口的速率控制功能。链接速度可以通过自动协商过程或手动速度选择来选择。通过将位 0.12 清零来禁用自动协商时，允许手动速度选择。
- 当禁用自动协商并将位 0.6 清除为逻辑 0 时，将位 0.13 设置为逻辑 1 将 PHY 配置为 100 Mb / s 操作，并将位 0.13 清除为逻辑 0 将 PHY 配置为 10 Mb/s 操作。
- 当禁用自动协商并将位 0.6 设置为逻辑 1 时，将位 0.13 清零为逻辑 0 会选择 1000 Mb / s 的操作。将位 0.6 和 0.13 设置为逻辑 1 的组合保留用于未来的标准化。
- 当使能自动协商时，可以读取或写入位 0.6 和 0.13，但位 0.6 和位 0.13 的状态对链路配置没有影响，位 0.6 和位 0.13 不需要反映当它被读取时链接。

如果 PHY 通过比特 1.15: 9 和比特 15.15: 12 报告它不能工作在所有速度时，则比特 0.6 和 0.13 的值应该与 PHY 可以操作的速度相对应。并且任何试图将该位设置为无效的操作均将被忽略。

1. 对 Speed Selection 的修改设置，往往需要复位端口才能配置生效。因此在设置该位置的时候需要检查自动协商的设置并通过 Bit15 复位端口。
2. 位 0.6 和 0.13 的默认值是根据位 1.15:9 和 15.15:12 所指示的 PHY 可以操作的**最高数据速率**的编码组合。

- **Auto-Negotiation Enable:** 自动协商过程应通过将位 0.12 设置为逻辑 1 来启用。如果位 0.12 设置为逻辑 1，则

位 0.13、0.8 和 0.6 不对链路配置和除了自动协商协议规定之外的站操作产生影响。 如果将位 0.12 清零为逻辑 0，则无论链路配置和自动协商过程的先前状态如何，位 0.13、0.8 和 0.6 都将确定链路配置。

如果 PHY 通过位 1.3 报告它缺乏执行自动协商的能力，则 PHY 应在位 0.12 返回零值。 如果 PHY 通过位 1.3 报告它缺乏执行自动协商的能力，则位 0.12 应该始终写为 0，并且任何尝试将 1 写入位 0.12 都应该被忽略。

必须注意的是，对于 1000BASE-T 接口，自动协商必须打开。

- **Power Down:** 通过将位 0.11 设置为逻辑 1，可以将 PHY 置于低功耗状态。 清 0.11 位为零允许正常操作。 PHY 在掉电状态下的具体行为是特定实现的。 处于掉电状态时，PHY 应响应管理事务。 在转换到掉电状态期间和处于掉电状态期间，PHY 不应在 MII 或 GMII 上产生寄生信号。

当位 0.11 或位 0.10 被设置为逻辑 1 时，PHY 不需要满足 RX_CLK 和 TX_CLK 信号功能要求。 在位 0.11 和 0.10 清零后，PHY 应在 0.5 s 内满足 22.2.2 中定义的 RX_CLK 和 TX_CLK 信号功能要求。

1. Power Down 模式一般在软件 shut down 端口的时候使用，需要注意的是端口从 Power Down 模式恢复，需要复位端口以保证端口可靠的连接。
2. 该位的默认值为 0。

- **Isolate:** 通过将位 0.10 设置为逻辑 1，PHY 可能被迫将其数据路径与 MII 或 GMII 电隔离。 清零位 0.10 允许正常操作。 当 PHY 与 MII 或 GMII 隔离时，它不会响应 TXD 数据包和 TX_EN, TX_ER、GTX_CLK 的输入。并且它的 TX_CLK, RX_CLK, RX_DV, RX_ER, RXD 数据包、COL 和 CRS 输出均应为高阻态。 当 PHY 与 MII 或 GMII 隔离时，它将响应管理事务（MDC/MDIO 接口的信号）。

1. IEEE802.3 没有对 Isolate 时 MDI 接口的状态进行规范，此时 MDI 端可能还在正常运行。Isolate 在实际应用中并没有用到。
2. 由于目前很多百兆的 PHY 芯片其 MAC 接口主流的都是 SMII/S3MII，8 个端口的接口是相互关联的，一个端口设置 Isolate 可能会影响其他端口的

正常使用，因此在使用中注意不要随意更改 bit10 的状态。

- **Restart Auto-Negotiation:** 如果 PHY 通过位 1.3 报告它缺乏执行自动协商的能力，或者如果自动协商被禁用，则 PHY 应在位 0.9 返回零值。如果 PHY 通过位 1.3 报告它缺乏执行自动协商的能力，或者如果禁用了自动协商，则应将位 0.9 始终写为 0，并且任何尝试将 1 写入位 0.9 应被忽略。

Bit9 置 1 将重新启动端口的自动协商进程，当然前提是 Auto-Negotiation Enable 是使能的。

一般在修改端口的自动协商能力信息之后通过 Bit9 置 1 重新启动自动协商来使端口按照新的配置建立 link。

- **Duplex Mode:** 可以通过自动协商过程或手动双面选择来选择双工模式。通过将位 0.12 清零来禁用自动协商时，允许手动双面选择。
 - 当禁用自动协商时，将位 0.8 设置为逻辑 1 将 PHY 配置为全双工操作，并将位 0.8 清零以将逻辑 0 配置为用于半双工操作的 PHY。
 - 当启用自动协商时，可以读取或写入位 0.8，但位 0.8 的状态对链路配置没有影响。

如果 PHY 通过位 1.15: 9 和 15.15: 12 报告它只能在一个双工模式下工作，则位 0.8 的值应该与 PHY 可以工作的模式相对应，并且任何尝试改变将该位 0.8 修改为无效指的操作应被忽略。

对 Duplex Mode 的修改配置也需要复位端口才能生效。

- **Collision Test:** 冲突信号 (COL) 测试开关。在需要对 COL 信号进行测试时，可以通过 Bit7 置 1，这时 PHY 将输出一个 COL 脉冲以供测试。实际测试操作中也可以将端口配置为半双工状态，通过发帧冲突来测试 COL 信号，因此该配置实用价值不大。
- **Unidirectional enable:** 如果 PHY 通过比特 1.7 报告它不具备编码和传输来自媒体独立接口的数据的能力，而不管 PHY 是否确定已建立有效链路，则 PHY 应在比特 0.5 中返回零值，并且任何尝试写一个到位 0.5 应该被忽略。

寄存器 1 是 PHY 状态寄存器，主要包含 PHY 的状态信息，大多数 bit 的值都是由芯片厂家确定的，每一个 bit 的功能在表 3 种已有详细说明。

| Bit(s) | Name | Description | R/W ^a |
|--------|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| 1.15 | 100BASE-T4 | 1 = PHY able to perform 100BASE-T4 0 = PHY not able to perform 100BASE-T4 | RO |
| 1.14 | 100BASE-X Full Duplex | 1 = PHY able to perform full duplex 100BASE-X 0 = PHY not able to perform full duplex 100BASE-X | RO |
| 1.13 | 100BASE-X Half Duplex | 1 = PHY able to perform half duplex 100BASE-X 0 = PHY not able to perform half duplex 100BASE-X | RO |
| 1.12 | 10 Mb/s Full Duplex | 1 = PHY able to operate at 10 Mb/s in full duplex mode 0 = PHY not able to operate at 10 Mb/s in full duplex mode | RO |
| 1.11 | 10 Mb/s Half Duplex | 1 = PHY able to operate at 10 Mb/s in half duplex mode 0 = PHY not able to operate at 10 Mb/s in half duplex mode | RO |
| 1.10 | 100BASE-T2 Full Duplex | 1 = PHY able to perform full duplex 100BASE-T2 0 = PHY not able to perform full duplex 100BASE-T2 | RO |
| 1.9 | 100BASE-T2 Half Duplex | 1 = PHY able to perform half duplex 100BASE-T2 0 = PHY not able to perform half duplex 100BASE-T2 | RO |
| 1.8 | Extended Status | 1 = Extended status information in Register 15 0 = No extended status information in Register 15 | RO |
| 1.7 | Unidirectional ability | 1 = PHY able to transmit from media independent interface regardless of whether the PHY has determined that a valid link has been established 0 = PHY able to transmit from media independent interface only when the PHY has determined that a valid link has been established | RO |
| 1.6 | MF Preamble Suppression | 1 = PHY will accept management frames with preamble suppressed 0 = PHY will not accept management frames with preamble suppressed | RO |
| 1.5 | Auto-Negotiation Complete | 1 = Auto-Negotiation process completed 0 = Auto-Negotiation process not completed | RO |
| 1.4 | Remote Fault | 1 = remote fault condition detected 0 = no remote fault condition detected | RO/ LH |
| 1.3 | Auto-Negotiation Ability | 1 = PHY is able to perform Auto-Negotiation 0 = PHY is not able to perform Auto-Negotiation | RO |
| 1.2 | Link Status | 1 = link is up 0 = link is down | RO/ LL |
| 1.1 | Jabber Detect | 1 = jabber condition detected 0 = no jabber condition detected | RO/ LH |
| 1.0 | Extended Capability | 1 = extended register capabilities 0 = basic register set capabilities only | RO |

^aRO = Read only, LL = Latching low, LH = Latching high

寄存器中各位的详细说明如下：

- **100BASE-T4 ability:** 当读为逻辑 1 时，位 1.15 指示 PHY 有能力使用 100BASE-T4 信令规范执行链路发送和接收。当读为逻辑 0 时，位 1.15 表示 PHY 缺乏使用 100BASE-T4 信令规范执行链路发送和接收的能力。
- **100BASE-X full duplex ability:** 当读为逻辑 1 时，位 1.14 指示 PHY 有能力使用 100BASE-X 信令规范执行全双工链路传输和接收。当作为逻辑 0 读取时，bit1.14 表示 PHY 缺乏使用 100BASE-X 信令规范执行全双工链路传输和接收的能力。
- **100BASE-X half duplex ability:** 当读为逻辑 1 时，位 1.13 指示 PHY 有能力使用 100BASE-X 信令规范执行半双工链路传输和接收。当读为逻辑 0 时，位 1.13 指示 PHY 缺乏使用 100BASE-X 信令规范执行半双工链路传输和接收的能力。
- **其他同类型的值意义基本与上面几个相同：指示 PHY 所具有的工作模式能力，不再一一说明。**

- **Unidirectional ability:** 当读为逻辑 1 时，位 1.7 指示 PHY 具有编码和传输来自媒体独立接口的数据的能力，而不管 PHY 是否确定已建立有效链路。当读为逻辑 0 时，位 1.7 指示 PHY 只有在 PHY 确定已建立有效链路时才能从媒体独立接口传输数据。
- **MF preamble suppression ability:** 当读为逻辑 1 时，位 1.6 指示 PHY 能够接受管理帧，而不管它们是否在 22.2.4.5.2 中描述的前导码模式之前。当读为逻辑 0 时，位 1.6 指示 PHY 不能接受管理帧，除非它们之前是 22.2.4.5.2 中描述的前导码模式。
- **Auto-Negotiation Complete:** 当读为逻辑 1 时，位 1.5 指示自动协商过程已完成，并且由自动协商协议（条款 28 或条款 37）实施的扩展寄存器的内容是有效的。当读为逻辑 0 时，位 1.5 指示自动协商过程尚未完成，并且扩展寄存器的内容由自动协商协议的当前状态定义，或者为手动配置写入。如果自动协商通过清除位 0.12 禁用，则 PHY 应在位 1.5 返回零值。如果 PHY 缺乏执行自动协商的能力，它还应在位 1.5 返回零值。

在调试以及异常故障处理时，可以通过该位寄存器的状态判断 AN 是否成功，从而进一步的检查 AN 相关的设置是否正确，或者芯片的 AN 功能是否正常等。

- **Remote Fault:** 当读为逻辑 1 时，位 1.4 表示检测到远程故障状态。故障类型以及故障检测的标准和方法是 PHY 特定的。远程故障位必须使用锁存功能来实现，以便发生远程故障将导致远程故障位置位，并保持置位状态直至被清除。每当通过管理接口读取寄存器 1 时，远程故障位应清零，并且还应通过 PHY 复位清零。

远端错误指示位。Bit4=1 代表连接对端 (Link Partner) 出错，至于出错的具体类型以及错误检测机制在规范中并没有定义，由 PHY 的制造商自由发挥，一般的厂商都会在其他寄存器 (Register16-31 由厂商自行定义) 指示比较详细的错误类型。在与端口相关的故障查证中，Remote Fault 是一个重要的指示信息，通过互联双方的 Remote Fault 信息（可能要加上其他的具体错误指示），可以帮助定位故障原因。

- **Auto-Negotiation ability:** 当读为逻辑 1 时，位 1.3 指示 PHY 有能力执行自动协商。当读为逻辑 0 时，位 1.3 指示 PHY 缺乏执行自动协商的能力。

- **Link Status:** 当读为逻辑 1 时, 位 1.2 指示 PHY 已经确定已建立有效链路。 当作为逻辑 0 读取时, 位 1.2 指示该链接无效。 确定链路有效性的标准是 PHY 特定的。 链路状态位应该使用锁存功能来实现, 以便发生链路故障情况将导致链路状态位清零并保持清零, 直到通过管理接口读取。 此状态指示旨在支持在 30.5.1.1.4, aMediaAvailable 中定义的管理属性。

实际应用中一般都是通过 Bit2 来判断端口的状态。而且, 一般的 MAC 芯片也是通过轮询 PHY 的这个寄存器值来判断端口的 Link 状态的 (这个过程可能有不同的名称, 比如 BCM 叫做 Link Scan, 而 Marvell 叫做 PHY Polling。) 如前所述, 在 AN Enable 的情况下, Link Status 的信息只有在 Auto-Negotiation Complete 指示已经完成的情况下才是正确可靠的, 否则有可能出错。

- **Jabber Detect:** 当作为逻辑 1 读取时, 位 1.1 指示已经检测到爆音条件。 此状态指示旨在支持 30.5.1.1.6 中定义的管理属性, aJabber 和 30.5.1.3.1 nJabber 中定义的 MAU 通知。 检测 Jabber 条件的标准是 PHY 特定的。 Jabber 检测位应该使用锁存功能来实现, 以便发生 Jabber 条件将导致 Jabber 检测位置位, 并保持置位状态直至被清除。 每次通过管理接口读取寄存器 1 时, Jabber 检测位应清零, 并且还应通过 PHY 复位清零。

IEEE802.3 对 Jabber 的解释是 “A condition wherein a station transmits for a period of time longer than the maximum permissible packet length, usually due to a fault condition”。这一位指示的是 Link Partner 发送的时间超过了规定的最大长度。值得注意的是, Jabber Detect 只有在 10BASE-T 模式下才有意义, 100 和 1000M 模式是没有定义 Jabber 这一功能的。

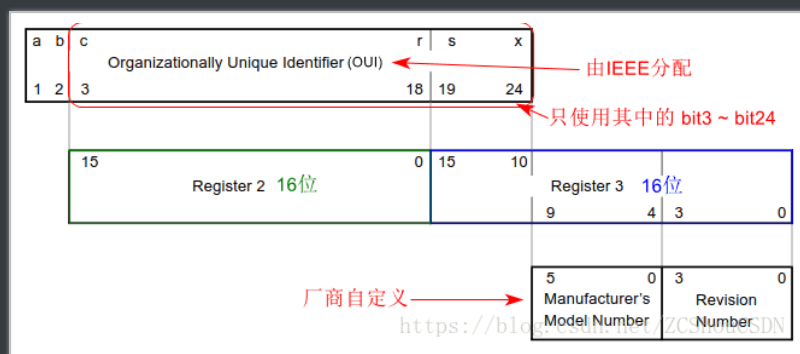
PHY Identifier Register

寄存器 2 和 3 存放 PHY 芯片的型号代码, 寄存器 2(PHY ID1) 为高 16 位, 而寄存器 3(PHY ID2) 为低 16 位。由芯片制造商自行定义, 实际应用中软件通过读取这两个寄存器的内容可以识别 PHY 的型号和版本。

PHY 标识符应由 IEEE 分配给 PHY 制造商的组织唯一标识符 (OUI) 的 (只需要使用第 3 至第 24 位, 共 22 位), 加上 6 位制造商的型号以及 4 位制造商的修订版编号组成。 PHY 标识符旨在提供足够的信息来支持 30.1.2 中所要求的 oResourceTypeID 对象。

OUI 的第三位分配给位 2.15, OUI 的第四位分配给位 2.14,

依此类推。位 2.0 包含 OUI 的第十八位。位 3.15 包含 OUI 的第十九位，位 3.10 包含 OUI 的第二十四位。位 3.9 包含制造商型号的 MSB。位 3.4 包含制造商型号的 LSB。位 3.3 包含制造商版本号的 MSB，位 3.0 包含制造商版本号的 LSB。具体如下图所示：



Auto-Negotiation Advertisement Register (Register 4) (R/W)

寄存器 4 是自动协商的能力通告寄存器，在 AN Enable 的前提下（见寄存器 0），端口根据该寄存器的相关配置将自动协商信息通过 FLP 在 MDI 上进行通告。当 AN 配置为 Disable 状态的时候，寄存器 4 的配置将不起作用，端口的工作模式由控制寄存器中的配置决定。

该寄存器包含 PHY 的通告能力，它们将在自动协商期间传送给其链接伙伴。基本页的位定义在 IEEE 标准的 28.2.1.2 中定义。上电时，在自动协商开始之前，该寄存器应具有以下默认配置：

- **Selector Field (4.4: 0)：** 被设置为适当的代码，如 IEEE 标准的附件 28A 中所规定。
- **Reserved (4.14)：** 被设置为逻辑 0。
- **Technology Ability Field (4.11: 5)：** 根据 MII 状态寄存器（寄存器 1）（1.15: 11）中设置的值或等效值设置。另见 28.2.1.2.3 和附件 28D。

| Bit(s) | Name | Description | R/W ^a |
|--------|--------------------------|-------------------------------|------------------|
| 4.15 | Next Page | See 28.2.1.2 | R/W |
| 4.14 | Reserved | Write as zero, ignore on read | RO |
| 4.13 | Remote Fault | See 28.2.1.2 | R/W |
| 4.12 | Extended Next Page | See 28.2.1.2 | R/W |
| 4.11:5 | Technology Ability Field | See 28.2.1.2 | R/W |
| 4.4:0 | Selector Field | See 28.2.1.2 | R/W |

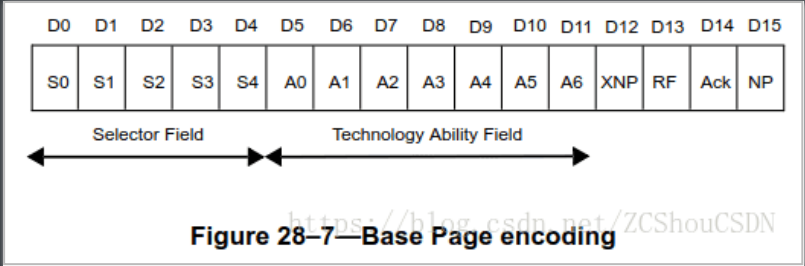
^aRO = Read only, R/W = Read/Write.

https://blog.csdn.net/ZCSnouCSdn

Link codeword encoding (基本链路码字)

在 FLP Burst 内传输的基本链路代码字（基本页面）应该传达如图 28-7 所示的编码。自动协商功能可以使用下一页功能支持其

他页面。 下一页交换中使用的链接代码字的编码在 28.2.3.4 中定义。 在 FLP Burst 中, D0 应该是第一个传输的位。



Next Page function

下一页功能使用标准的自动协商仲裁机制来允许交换任意的数据。 数据由可选的下一页信息携带, 其遵循用于基本链接码字的传输和确认过程。 定义了四种类型的下一页编码: 消息页面, 未格式化页面, 扩展消息页面和扩展的未格式化页面。

关于该部分, 具体见 IEEE 标准的 28.2.3.4 Next Page function。

在 IEEE 标准中, Auto-Negotiation Advertisement Register 中的各部分全部是在独立章节中进行介绍的。具体如下:

- **Selector Field:** 选择器字段 (S [4: 0]) 是一个 5 位宽的字段, 编码 32 个可能的消息。 链路码字中的选择器字段 S [4: 0] 应用于识别自动协商发送的消息的类型。 下表列出了可能发送的消息的类型。 随着新消息的发展, 该表格将相应更新。

| Table 28A-1—Selector Field value mappings | | | | | |
|-------------------------------------------|----|----|----|----|---------------------------------------------------------------|
| S4 | S3 | S2 | S1 | S0 | Selector description |
| 0 | 0 | 0 | 0 | 0 | Reserved for future Auto-Negotiation development |
| 0 | 0 | 0 | 0 | 1 | IEEE Std 802.3 |
| 0 | 0 | 0 | 1 | 0 | IEEE Std 802.9a-1995 (withdrawn) |
| 0 | 0 | 0 | 1 | 1 | IEEE Std 802.5v-2001 (withdrawn) |
| 0 | 0 | 1 | 0 | 0 | IEEE Std 1394 |
| 0 | 0 | 1 | 0 | 1 | INCITS |
| 0 | 0 | 1 | 1 | X | Reserved for future Auto-Negotiation development ^a |
| 0 | 1 | X | X | X | Reserved for future Auto-Negotiation development |
| 1 | X | X | X | X | Reserved for future Auto-Negotiation development |

^aFor up-to-date information on the allocation of Auto-Negotiation Selector Fields see <http://www.ieee802.org/3/selectors/selectors.html>

未指定的组合保留供将来使用。 不会传输选择器字段的预留组合。*我们所接触的以太网 PHY 遵从 IEEE802.3 规范, Selector Field=0001, 该区域不可随意更改 (很多 PHY 将此区域设计为只读寄存器, 以免被修改)。*

- **Technology Ability Field:** 技术能力字段 (A [6: 0]) 是一个 7 位宽的字段, 其中包含指示选择器字段值特定的支持技术的信息。 这些位被映射到各个技术, 以便能够针对单个选择

器字段值并行通告能力。附录 28B.2 和附件 28D 描述了 IEEE 802.3 选择器的技术能力字段编码。链接代码字中可能会公布多种技术。设备应支持其宣传的技术的数据服务能力。

| Table 28B-1—Technology Ability Field bit assignments | | |
|------------------------------------------------------|--------------------------------------------------|-----------------------------|
| Bit | Technology | Minimum cabling requirement |
| A0 | 10BASE-T | Two-pair Category 3 |
| A1 | 10BASE-T full duplex | Two-pair Category 3 |
| A2 | 100BASE-TX | Two-pair Category 5 |
| A3 | 100BASE-TX full duplex | Two-pair Category 5 |
| A4 | 100BASE-T4 | Four-pair Category 3 |
| A5 | PAUSE operation for full duplex links | Not applicable |
| A6 | Asymmetric PAUSE operation for full duplex Links | Not applicable |

- **Extended Next Page:** 扩展下一页 (XNP) 被编码在基本链路码字的位 D12 中。扩展下一页位指示本地设备在设置为逻辑 1 时支持扩展下一页的传输，并指示本地设备在设置为逻辑 0 时不支持扩展下一页。Extended Next Page 的使用与协商的数据速率，媒体或链接技术是正交的。扩展下一页位按照 28.2.3.4 中的扩展下一页功能规范使用。
- **Next Page:** 无论选择器字段值还是链接码字编码，下一页 (NP) 都在 D15 位编码。支持附加链接码字编码的传输和接收是可选的。如果不支持 Next Page 功能，NP 位应始终设置为逻辑 0。如果设备实现下一页功能并希望进行下一页交换，则应将 NP 位设置为逻辑 1。设备可以实现下一页功能，并通过将 NP 位设置为逻辑 0 来选择不参与下一页交换。下一页功能在 28.2.3.4 中定义。
- **Remote Fault:** 远程故障 (RF) 编码在基本链路码字的位 D13 中。默认值是逻辑零。远程故障位为传输简单的故障信息提供了一种标准的传输机制。当自动协商通告寄存器 (寄存器 4) 中的 RF 位设置为逻辑 1 时，发送的基本链路码字中的 RF 位被设置为逻辑 1。当接收到的基本链路代码字中的 RF 位被设置为逻辑 1 时，如果存在 MII 管理功能，则 MII 状态寄存器 (寄存器 1) 中的远程故障位将被设置为逻辑 1。

Auto-Negotiation Link Partner ability register (Register 5) (RO)

寄存器 5 保存的是本端 PHY 接收到的对端 PHY 所通告的端口能力，寄存器 5 的结构和寄存器 4 基本一致。

Auto-Negotiation Expansion Register (Register 6) (RO)

寄存器 6 保存了 PHY 自动协商过程的异常信息。从这个寄存器其中我们可以获取到 Link Partner 子否支持自动协商以及自动协商下一页有没有收到的信息。

Table 28–5—Expansion register bit definitions

| Bit(s) | Name | Description | R/W ^a | Default |
|--------|-------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|------------------|---------|
| 6.15:7 | Reserved | Write as zero, ignore on read | RO | 0 |
| 6.6 | Receive Next Page Location Able | 1 = Received Next Page storage location is specified by bit (6.5) 0 = Received Next Page storage location is not specified by bit (6.5) | RO | — |
| 6.5 | Received Next Page Storage Location | 1 = Link Partner Next Pages are stored in Register 8 0 = Link Partner Next Pages are stored in Register 5 | RO | — |
| 6.4 | Parallel Detection Fault | 1 = A fault has been detected via the Parallel Detection function. 0 = A fault has not been detected via the Parallel Detection function. | RO/ LH | 0 |
| 6.3 | Link Partner Next Page Able | 1 = Link Partner is Next Page able 0 = Link Partner is not Next Page able | RO | 0 |
| 6.2 | Next Page Able | 1 = Local Device is Next Page able 0 = Local Device is not Next Page able | RO | 0 |
| 6.1 | Page Received | 1 = A New Page has been received 0 = A New Page has not been received | RO/ LH | 0 |
| 6.0 | Link Partner Auto-Negotiation Able | 1 = Link Partner is Auto-Negotiation able 0 = Link Partner is not Auto-Negotiation able | RO | 0 |

^aRO = Read only, LH = Latching high.<https://blog.csdn.net/ZCShouCSDN>

Auto-Negotiation Next Page transmit register (Register 7) (R/W)

自动协商下一页发送寄存器包含在支持下一页功能时要发送的下一页链接码字。（见表 28-6）内容在 28.2.3.4 中定义。上电时，该寄存器应包含默认值 2001H，该值表示消息代码设置为 Null 消息的消息页面。该值可以由设备希望传输的任何有效的下一页消息代码替换。写入该寄存器应将 mr_next_page_loaded 设置为 true。

Table 28–6—Next Page transmit register bit definitions

| Bit(s) | Name | Description | R/W ^a |
|--------|--------------------------------|----------------------------|------------------|
| 7.15 | Next Page | See 28.2.3.4 | R/W |
| 7.14 | Reserved | Write as 0, ignore on read | RO |
| 7.13 | Message Page | See 28.2.3.4 | R/W |
| 7.12 | Acknowledge 2 | See 28.2.3.4 | R/W |
| 7.11 | Toggle | See 28.2.3.4 | RO |
| 7.10:0 | Message/Unformatted Code field | See 28.2.3.4 | R/W |

^aRO = Read only, R/W = Read/Write.<https://blog.csdn.net/ZCShouCSDN>

Auto-Negotiation Link Partner Received Next Page register (Register 8) (RO)

Table 28–7—Link Partner Received Next Page register bit definitions

| Bit(s) | Name | Description | R/W ^a |
|--------|--------------------------------|--------------|------------------|
| 8.15 | Next Page | see 28.2.3.4 | RO |
| 8.14 | Acknowledge | see 28.2.3.4 | RO |
| 8.13 | Message Page | see 28.2.3.4 | RO |
| 8.12 | Acknowledge 2 | see 28.2.3.4 | RO |
| 8.11 | Toggle | see 28.2.3.4 | RO |
| 8.10:0 | Message/Unformatted Code Field | see 28.2.3.4 | RO |

^aRO = Read only.

<https://blog.csdn.net/ZCSnouCSDN>

未完待续。。。

附件

1. 无

参考文档

- IEEE Standard for Ethernet
- 网络

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎^{beta}，[点击查看详细说明](#)

