NETWORK DESIGN FAQs

Frequently Asked Questions:

ETHERNET MAC AND PHY

William Wong, Embedded /Systems /Software Editor

How do you implement a single-chip Ethernet microcontroller?

The trick is to incorporate the microcontroller, Ethernet MAC, and PHY on a single chip, thereby eliminating most external components. This enables the MAC and PHY to be matched and reduces the overall pin count and chip footprint. It can also lower power consumption, especially if power-down modes are implemented.

What is an Ethernet MAC?

The MAC is the media access controller. The Ethernet MAC is defined by the IEEE-802.3 Ethernet standard. It implements a data-link layer. The latest MACs support operation at both 10 Mbits/s and 100 Mbits/s. This crop typically implements the MII.

What is the MII?

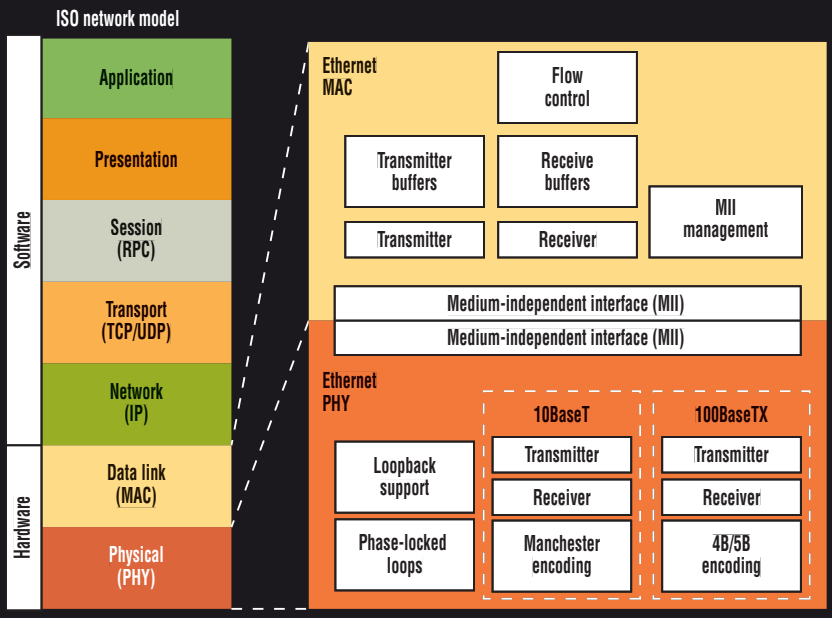
The Media Independent Interface (MII) is an Ethernet industry standard defined in IEEE 802.3. It consists of a data interface and a management interface between a MAC and a PHY *(Fig. 1)*. The data interface consists of a channel for the transmitter and a separate channel for the receiver. Each channel has its own clock, data, and control signals. The MII data interface requires a total of 16 signals. The management interface is a two-signal interface—one signal for clocking and the other for data. With the management interface, upper layers can monitor and control the PHY.

What is an Ethernet PHY?

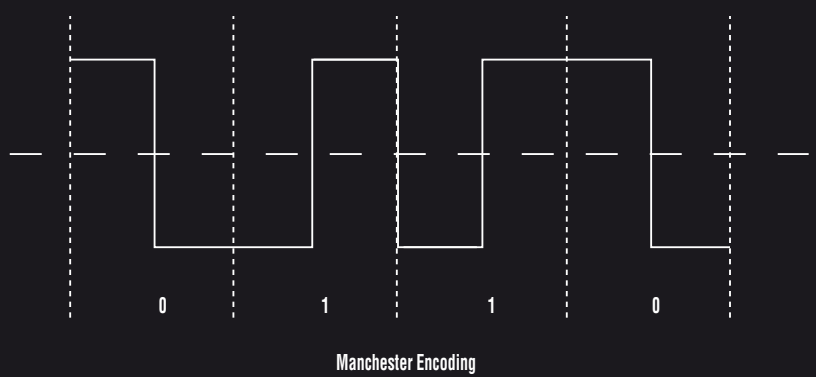
The PHY is the physical interface transceiver. It implements the physical layer. The IEEE-802.3 standard defines the Ethernet PHY. It complies with the IEEE-802.3 specifications for 10BaseT (clause 14) and 100BaseTX (clauses 24 and 25).

What's so hard about putting an Ethernet PHY on the same chip as the MAC?

The PHY incorporates a signifi-



1. The Ethernet MAC and PHY implement the bottom two layers of the International Organization for Standardization/Open System Interconnect (ISO/OSI) stack. The MAC interfaces with the PHY through an MII. The typical 10/100 PHY Ethernet implementation incorporates separate 10BaseT and 100BaseTX interfaces.



2. Manchester Encoding requires transitions up to twice the frequency of data throughput, but it allows easy clock synchronization.

Sponsored by Freescale Semiconductor

cant amount of analog hardware, while the MAC is typically an all-digital component. The chip footprint and the mixed analog/digital architecture are why MACs were first incorporated in microcontrollers, leaving the PHY off-chip. More flexible, higher-density chip technology has allowed both the MAC and PHY to reside on the same chip.

Are external components other than the RJ-45 jack still required?

Yes. The PHY provides most of the analog support, but the typical implementation still requires a half-dozen discretes and a local-area network magnetics isolation module. The latter is typically a 1:1 ratio transformer. These primarily protect the PHY from electrical abuse.

Why are the 10BaseT and 100BaseTX PHY implementations different?

The packet description for both implementations is essentially the same, but the signaling mechanism for the two is completely different. This prevents one hardware implementation from easily handling both speeds. The 10BaseT uses Manchester encoding, while the 100BaseTX uses 4B/5B encoding.

What is Manchester Encoding?

Manchester Encoding, also called Manchester Phase Encoding, implements each bit as a phase change *(Fig. 2)*. A "1" is normally a rising edge in the middle of a cycle, while a "0" is a falling edge. A transition at the end of a cycle is ignored and may be required depending upon the prior bit value.

What is 4B/5B encoding?

4B/5B encoding is one of many block coding methods. It takes a 4-bit block and encodes it into a 5-bit block. This allows the 5-bit block to always have at least two "1" transitions so clock synchronization will always be possible within a 5-bit block. The approach imposes 25% overhead. ED Online 9177