AN1120

Ethernet Theory of Operation

Author: *M. Simmons*
Microchip Technology Inc.

INTRODUCTION

This document specifies the theory and operation of the Ethernet technology found in PIC® MCUs with integrated Ethernet and in stand-alone Ethernet controllers.

Ethernet technology contains acronyms and terms defined in Table 1.

APPLICATIONS

Ethernet is an asynchronous Carrier Sense Multiple Access with Collision Detect (CSMA/CD) protocol/interface, with a payload size of 46-1500 octets. With data rates of tens to hundreds of megabits/second, it is generally not well suited for low-power applications.

However, with ubiquitous deployment, internet connectivity, high data rates and limitless range expansibility, Ethernet can accommodate nearly all wired communications requirements. Potential applications include:

- Remote sensing and monitoring
- Remote command, control and firmware updating
- Bulk data transfer
- Live streaming audio, video and media
- Public data acquisition (date/time, stock quotes, news releases, etc.)

THEORY OF OPERATION

Ethernet is a data link and physical layer protocol defined by the IEEE 802.3™ specification. It comes in many flavors, defined by maximum bit rate, mode of transmission and physical transmission medium.

- Maximum Bit Rate (Mbits/s): 10, 100, 1000, etc.
- Mode of Transmission: Broadband, Baseband
- Physical Transmission Medium: Coax, Fiber, UTP, etc.

TABLE 1: ETHERNET GLOSSARY

Term	Definition
CRC	Cyclic Redundancy Check: Type of checksum algorithm used when computing the FCS for all Ethernet frames and the hash table key for hash table filtering of receive packets.
DA	Destination Address: The 6-octet destination address field of an Ethernet frame.
ESD	End-of-Stream Delimiter: In 100 Mb/s operation, the ESD is transmitted after the FCS (during the inter-frame gap) to denote the end of the frame.
FCS	Frame Check Sequence: The 4-octet field at the end of an Ethernet frame that holds the error detection checksum for that frame.
IP	Internet Protocol: Refers either to IPv4 or IPv6.
LAN	Local Area Network or Large Area Network.
MAC	Media Access Control: The block responsible for implementing the Media Access Control functions of the Ethernet specification.
MAC Address	A 6-octet number representing the physical address of the node(s) on an Ethernet network. Every Ethernet frame contains both a source and destination address, both of which are MAC addresses.
MDI	Medium Dependent Interface or Management Data Input.
MDO	Management Data Output.
MDIO	Management Data Input/Output.
MII	Media Independent Interface: Standard 4-bit interface between the MAC and the PHY for communicating TX and RX frame data. In 10 Mb/s mode, the MII runs at 2.5 MHz; in 100 Mb/s mode, it runs at 25 MHz.
MIIM	MII Management: Set of MII sideband signals used for accessing the PHY registers.

TABLE 1: ETHERNET GLOSSARY (CONTINUED)

Term	Definition
OUI	Organizationally Unique Identifier: The upper three octets of a MAC address are referred to as the OUI, and typically are assigned to an organization or company. Microchip's OUI is 00-04-A3h.
Octet	In Ethernet terms, one 8-bit byte.
Packet Buffer	The physical or virtual memory where all transmit and receive packets (frames) are stored.
PHY	The block that implements the Ethernet physical layer.
RAM	Random Access Memory (normally volatile memory).
Receive Buffer	Logical portion of the packet buffer used to store received packets.
RX	Receive.
SA	Source Address: The 6-octet source address field of an Ethernet frame.
SFD	Start Frame Delimiter: The single octet field of an Ethernet frame that marks the start of a frame.
SPI	Serial Peripheral Interface.
SSD	Start-of-Stream Delimiter: In 100 Mb/s Ethernet, the first octet of the preamble is known as the SSD and is encoded differently from the rest of the preamble.
Station Address	The Station Address is the MAC address of the Ethernet node. It is typically compared against the destination address in a received Ethernet frame to determine if the frame should be received or not. On the transmit side, it is typically transmitted as the source address of an Ethernet frame.
Transmit Buffer	Logical portion of the packet buffer used to store packets to be transmitted.
TX	Transmit.
RMII	Reduced Media Independent Interface: A 2-bit version of the MII.
SMII	Serial Media Independent Interface: A 1-bit version of the MII.
NRZI	Non-Return-to-Zero Inverted: A binary code in which a logical one is represented by a signal transition and a logical zero is represented by the lack of a transition.

PROTOCOL STACK

The easiest way to understand the role that Ethernet plays is by looking at a protocol stack, which describes a complete protocol or set of protocols in a layered approach (see Figure 1).

Frame/Packet Encapsulation

To understand how Ethernet works, it is first necessary to understand the concept of packet encapsulation, and how the protocol stack fits into this concept.

Each layer of the protocol stack is responsible for a particular level of functionality. As an example, the physical layer is concerned with the actual electrical transmission of bits across a medium. Each higher layer in the model utilizes the underlying layers in a somewhat independent fashion (meaning little or no overlap in functions between the layers).

This layered approach is implemented through the use of encapsulation. This concept can best be explained using the example shown in Figure 2. This example shows how each layer associated with a web browser session maps to the protocol stack model.

Starting at the application layer, the web browser would generate an HTTP request using an application-specific command. This request would then be passed down to the TCP layer, which would construct a TCP packet consisting of a TCP header and TCP data. The TCP header contains information particular to the TCP protocol, such as packet sequencing information, checksum information and the source and destination port number (HTTP typically has a port number of 80).

At the IP protocol level, an IP datagram is constructed to hold the TCP packet. Similar to the TCP packet, the IP datagram consists of an IP header and IP data. The IP header contains information such as the type of service, checksum information, protocol type (06h for TCP), and the source and destination IP addresses. The data field of the IP datagram contains the complete TCP packet to be transmitted.

At the data link/physical layer, the IP datagram is transported across the network using the IEEE 802.3 protocol. A MAC (IEEE 802.3) frame consists of a MAC header and a MAC payload (data). The MAC header contains information about the MAC frame, such as the source MAC address, the destination MAC address and the length of the frame. The payload field contains the complete IP datagram to be transported.

Note that the various addresses encapsulated within each protocol are different, and typically, have no fixed relationship to one another. In our example, the TCP packet uses a port number, which is typically assigned based on the application layer protocol (i.e., port 80 for HTTP). The IP datagram uses an IP address, which is statically or dynamically assigned out of a pool of available internet addresses, and the MAC frame uses MAC addresses, which are assigned to the particular piece of hardware.

Note 1:	The terms “MAC frame”, “Ethernet frame” and “IEEE 802.3 frame” are used interchangeably in this document.
2:	The terms “packet”, “frame” and “datagram” are often used interchangeably. These terms apply to specific protocols, such as an IEEE 802.3 frame, a TCP packet or an IP datagram.

FIGURE 1: INTERNET PROTOCOL STACK

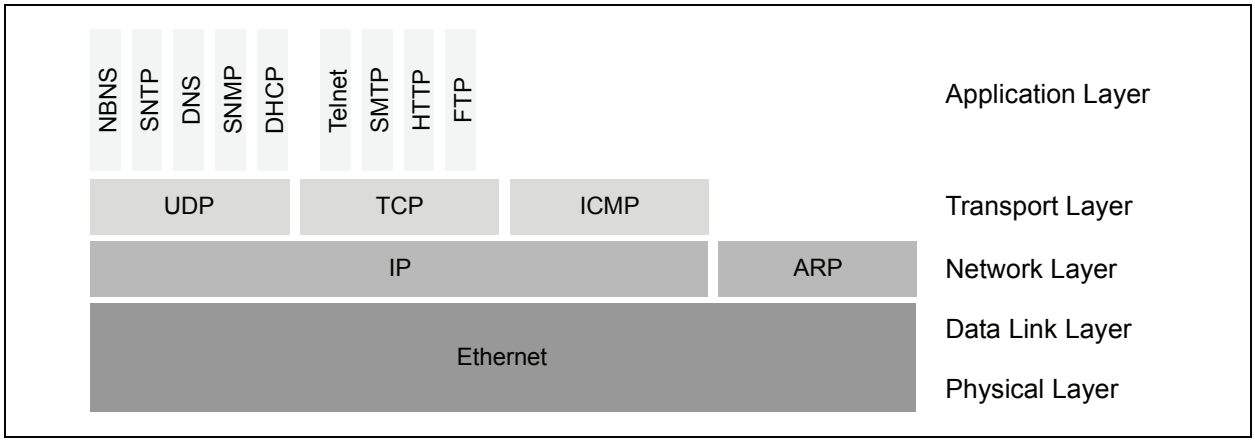
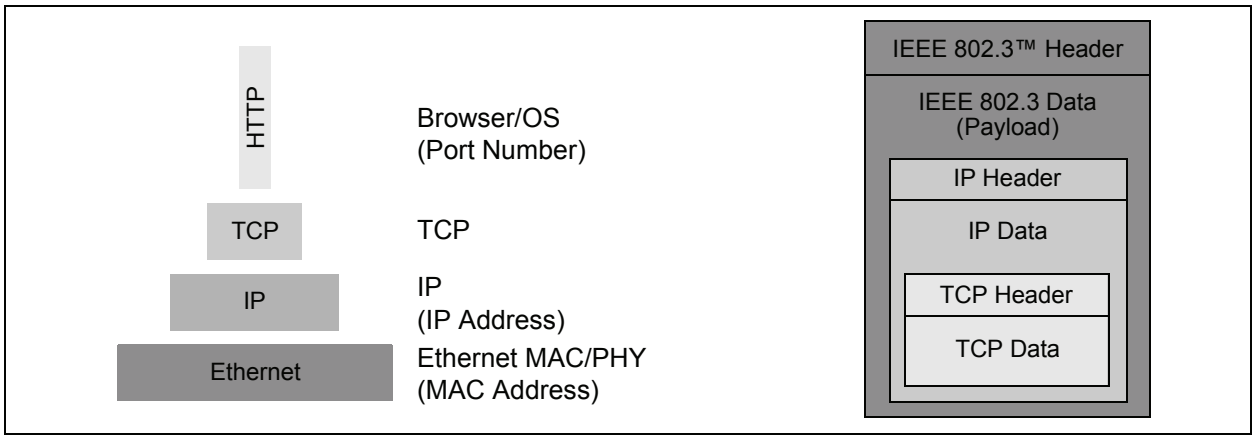


FIGURE 2: DATA ENCAPSULATION EXAMPLE



Application Layer Protocols

The application layer provides the user interface.

When used on top of some lower layer protocols (UDP or TCP – **"Transport Layer Protocols"** section), application layer protocols are usually assigned a port number. For example, HTTP servers are typically associated with port 80.

The following are common application layer protocols associated with the Internet:

Hyper Text Transfer Protocol (HTTP): Used primarily to transfer data associated with browsing of the World Wide Web.

Simple Mail Transfer Protocol (SMTP): Used to transport e-mails across the internet.

File Transfer Protocol (FTP): Used to transfer files or other pieces of data over the internet.

Domain Name System (DNS): Used to translate domain names, such as "microchip.com" into IP addresses.

Dynamic Host Configuration Protocol (DHCP): Used to dynamically assign IP addresses to a particular node from a pool of available IP addresses.

Telnet: Used to establish an interactive TCP connection to a node.

Simple Network Time Protocol (SNTP): Used to allow nodes to synchronize their clocks to a reference clock.

Simple Network Management Protocol (SNMP): Used to monitor network attached devices for conditions that require intervention, such as Faults, etc.

Transport Layer Protocols

The transport layer hides network dependent details from the layers above, including transport address to network address translation, sequencing, error detection/recovery, etc.

When used on top of the IP protocol, transport layer protocols are typically assigned an IP protocol number.

The following are common transport layer protocols associated with the internet:

Transmission Control Protocol (TCP): Provides reliable communication to applications.

User Datagram Protocol (UDP): Provides high performance, but unreliable communication to applications.

Internet Control Message Protocol (ICMP): Used to send network and/or node error or status messages.

Network Layer Protocols

The network layer determines how messages are routed in a network, including QoS (Quality of Service) services, provision of network addresses for the transport layer, etc.

When used on top of Ethernet, network layer protocols are typically assigned an “EtherType”, which is discussed in more detail in the **“Ethernet Frame Format”** section.

The following are common network layer protocols associated with the internet:

Address Resolution Protocol (ARP): Used to translate protocol addresses to hardware interface addresses, such as an IP address to a MAC address.

Reverse Address Resolution Protocol (RARP): Used to translate hardware interface addresses to protocol addresses, such as a MAC address to an IP address.

Internet Protocol (IP): Connectionless network layer protocol used by TCP, UDP, etc.

Physical/Data Link Layer Protocols

The physical layer provides for the transparent transmission of bit streams across physical connections, including encoding, multiplexing, synchronization, clock recovery, serialization, etc.

The data link layer is concerned with the transmission of frames (blocks) in an error-free manner, including frame sequencing, frame flow control, etc.

Ethernet is one of the most common physical/data link layer protocols, and the subject of the remainder of this application note.

PHYSICAL MEDIUM OVERVIEW

As mentioned previously, Ethernet is defined in part by the physical medium over which frames are transmitted. The following is a summary of the more common mediums:

- 1 Mb/s
  - 1Base5: 2 twisted telephone wire pairs
- 10 Mb/s
  - 10Broad36: 1 broadband cable
  - 10Base2: RG 58 coax cable
  - 10Base5: 1 coax cable
  - 10Base-F: 1 optical fiber
  - 10Base-T: 2 pairs UTP CAT3 or better, full-duplex
- 100 Mb/s
  - 100Base-FX: 2 optical fibers, Full-Duplex
  - 100Base-T2: 2 pairs UTP CAT3 or better, full-duplex
  - 100Base-T4: 4 pairs UTP CAT3 or better, half-duplex
  - 100Base-TX: 2 pairs UTP CAT5 or better, full-duplex
- 1 Gb/s
  - 1000Base-CX: Copper jumper cable
  - 1000Base-LX: Long wavelength Multi/Single mode fiber
  - 1000Base-SX: Short wavelength Multi mode fiber
  - 1000Base-T: 4 CAT5e, CAT6 or better pairs

Note 1: UTP – Unshielded Twisted Pair wire

2: CAT3 wires and copper telephone wires are essentially interchangeable.

ETHERNET SPECIFICATIONS

The Ethernet specification (IEEE 802.3) has evolved over the last number of years to address higher transmission rates and new functionality. Table 4 shows the most common specification supplements.

ETHERNET FRAME FORMAT

A basic 10/100 Ethernet frame consists of the following fields, as shown in Figure 3.

Preamble: Seven octets of 55h. In 100 Mb/s operation, the first octet is 4B/5B encoded to /J/K/ (more on what this means later), and is known as the Start-of-Stream Delimiter (SSD). The preamble is present to allow the receiver to lock onto the stream of data before the actual frame arrives.

Start-of-Frame Delimiter (SFD): '10101011b' (as seen on the physical medium). The SFD is sometimes considered to be part of the preamble. This is why the preamble is sometimes described as eight octets.

Destination Address (DA): The 6-octet MAC address of the destination hardware. Please refer to the **"MAC Addresses"** section for information on multicast and broadcast addressing.

Source Address (SA): The 6-octet MAC address of the source hardware.

Length/Type: If the value in this 2-octet field is ≤ 1500 (decimal), this represents the number of octets in the payload. If the value is ≥ 1536, this represents the EtherType (payload type). The following are the most common EtherType values:

- IPv4 = 0800h
- IPv6 = 86DDh
- ARP = 0806h
- RARP = 8035h

Payload (Client Data): The client data, such as an IP datagram, etc. The minimum payload size is 46 octets; the maximum payload size is 1500 octets. While payloads below or above these limits do not meet the IEEE 802.3 specification, there is varied support for these payloads depending on the particular vendor. Please refer to the **"Frame Size"** section for further discussion on this topic.

Pad: Since the minimum payload size is 46 octets, pad octets must be inserted to reach this minimum if the payload size is less than 46 octets.

Frame Check Sequence (FCS): The value of the 4-octet FCS field is calculated over the source address, destination address, length/type, data and pad fields using a 32-bit Cyclic Redundancy Check (CRC).

End-of-Stream Delimiter (ESD): In 100 Mb/s operation, the PHY transmits a /T/R/ symbol pair after the FCS (during the inter-frame gap) to denote the end of the frame.

In 10 Mb/s operation, a special TP_IDL signal (discussed later in this document) and network silence indicates the end of the frame. Like the /T/R/ symbol pair in 100Base-T, this special TP_IDL marker is not considered part of the frame data.

Note:	MAC frames are enumerated in terms of "octets" (one octet = 8 bits).

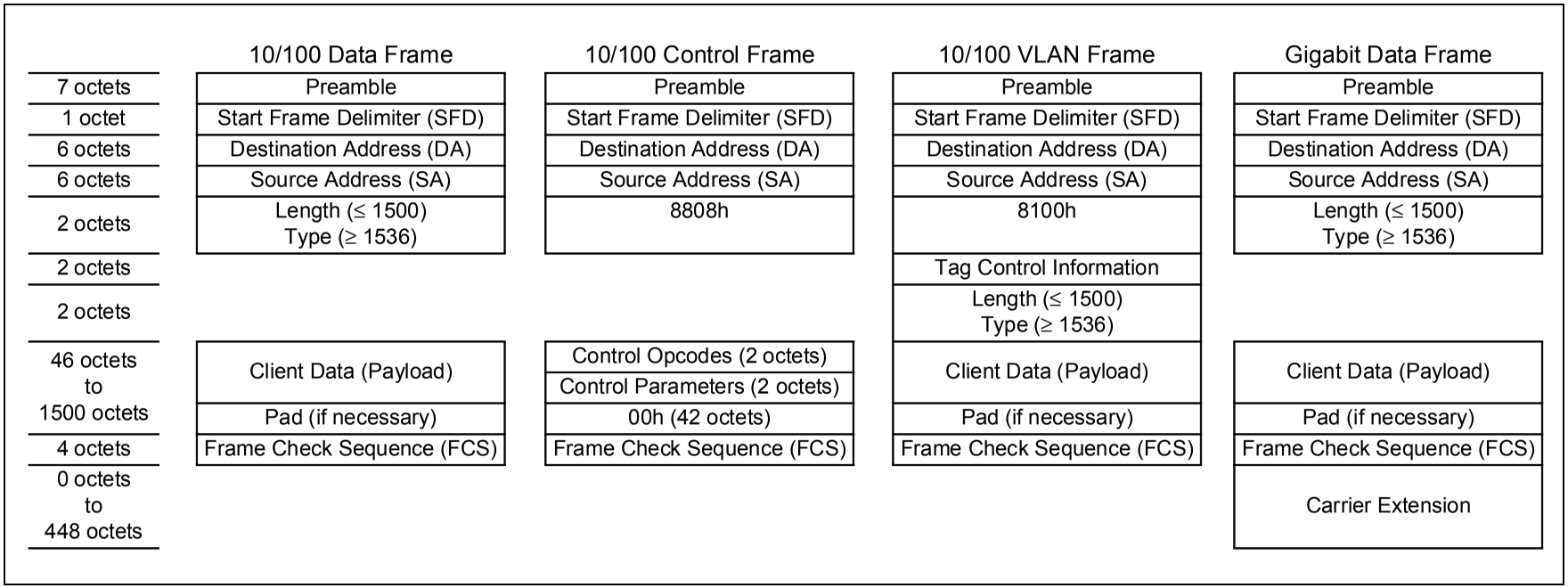
FIGURE 3: BASIC FRAME FORMAT

10/100 IEEE 802.3™ Frame

7 octets	Preamble
1 octet	Start Frame Delimiter (SFD)
6 octets	Destination Address (DA)
6 octets	Source Address (SA)
2 octets	Length (≤ 1500) Type (≥ 1536)
46 octets to 1500 octets	Client Data (Payload)
	Pad (if necessary)
4 octets	Frame Check Sequence (FCS)

Besides the basic frame described above, there are two other common frame types in 10/100 Ethernet: control frames and VLAN tagged frames. Figure 4 shows a comparison between the three common 10/100 frame formats and the gigabit Ethernet frame format.

FIGURE 4: COMMON ETHERNET FRAME TYPES



FRAME SIZE

When discussing IEEE 802.3 frame sizes, the Preamble/SFD is typically not included in the size of the frame. Therefore, the minimum and maximum allowed size of a basic or control frame is 64 octets and 1518 octets, respectively. Conversely, the maximum size for a VLAN tagged frame (described in the **"VLAN Tagged Frames"** section) is defined as 1522 octets.

Frames below the 64-octet limit are often known as "runt" frames, while frames above the 1518-octet limit are often known as "long" or "huge" frames. The term, "jumbo" frames, refers to frames above 1518 octets in 10/100Base-T and to 9000 octet frames in gigabit Ethernet. The term "giant" is sometimes used to refer to frames that are more than 6000 octets long.

In some literature, the term "frame size" refers solely to the payload of the frame. It is, therefore, common to see the term "jumbo frame" defined as a frame with a size of greater than 1500 octets.

Control Frames

Ethernet frames with an EtherType value of 8808h are specified as MAC control frames, and are used to control the flow of frames on a link. Implementation of MAC control features in an Ethernet node is optional.

The first two octets in a MAC control frame payload contain the opcode. Currently, the only standard control frame is a pause frame, which has an opcode and a destination address as follows:

- Opcode: 0001h
- Address: 01-80-c2-00-00-01 (multicast)

A pause frame requests that the station at the other end of the link stop transmitting for a period of time (specified by a 2-octet pause time after the opcode). One pause "quanta" is equal to 512 bit times.

Transmitting a pause frame with a pause time value of 0000h means to cancel any existing pauses in effect.

VLAN Tagged Frames

Virtual Local Area Network (VLAN) tagging adds additional information, known as tag control information, into the frame for the purpose of allowing the creation of networks defined by a logical topology, rather than a physical topology.

MAC ADDRESSES

A MAC address is a 48-bit (6-octet) number unique to every piece of Ethernet hardware. It consists of a 24-bit Organizationally Unique Identifier (OUI) and a 24-bit hardware identifier, as shown in Figure 5.

OUIs are assigned by the IEEE to a particular company or organization (Microchip's OUI is 00-04-A3h), while hardware IDs are assigned by the owner of that particular OUI.

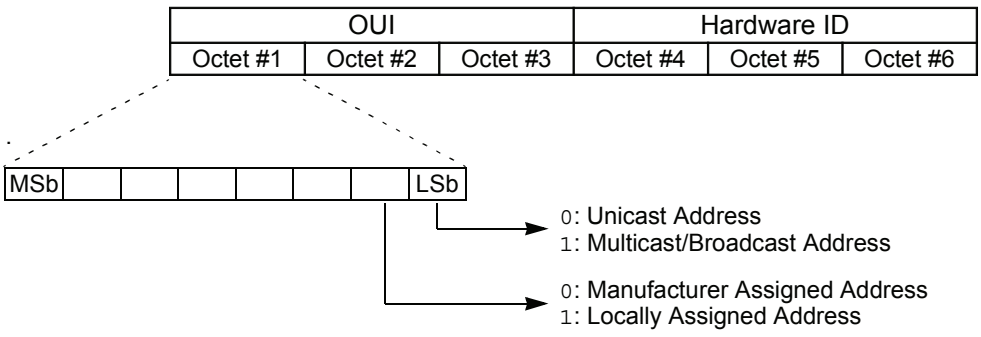
For individuals requiring 4096 MAC addresses or less, an Individual Address Block (IAB) may be purchased. An IAB consists of a reserved OUI (owned by the IEEE) plus 12 bits of reserved hardware identifier, which leaves 12 bits of hardware identifier available to the purchaser, for a total of 4096 unique MAC addresses.

MAC address octets are transmitted high-order (Octet #1) first, while bits within an octet are transmitted low-order, Least Significant bit (LSb) first.

A MAC address whose Least Significant bit of Octet #1 is set as a multicast address is intended for one or more nodes. As an example, pause frames, which have an address of 01-80-c2-00-00-01, are considered multicast packets.

A MAC address of FF-FF-FF-FF-FF-FF is a broadcast address, which is intended for all nodes.

FIGURE 5: MAC ADDRESSES



Example: A Microchip owned MAC Address.

OUI			Hardware ID		
Octet #1	Octet #2	Octet #3	Octet #4	Octet #5	Octet #6
00	04	A3	00	00	01

STREAM CONSTRUCTION/ DECONSTRUCTION

Based on the previous discussion of the protocol layer model and frame encapsulation, we are now ready to discuss the functions of the Ethernet MAC and PHY. The IEEE 802.3 definition of the PHY and MAC layers for 100 Mb/s are shown in Figure 6. What is important to realize from this diagram is that the functions of the Ethernet PHY and MAC, and the interfaces of each, are defined by the IEEE 802.3 specification.

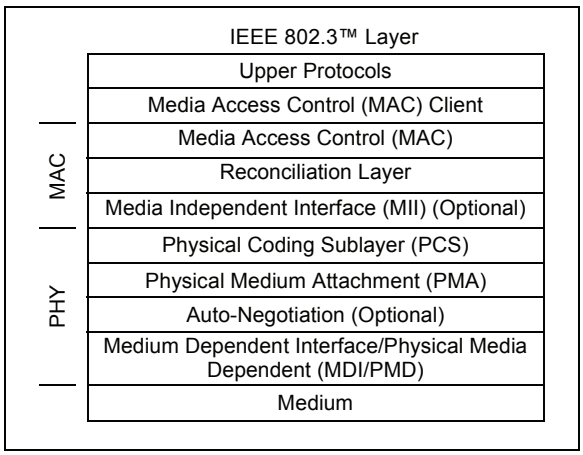
The physical interface to the transmission medium is called the MDI, and changes depending on which medium (twisted pair, fiber, etc.) is used.

The interface between the PHY and the MAC is called the MII, and is composed of a receive path, a transmit path and a management path, which is used to read and write PHY registers. The width of the receive and transmit paths are the same, and is determined by the speed that the MAC and PHY are implementing, as follows:

- 10 Mb/s: 4 bits wide at 2.5 MHz
- 100 Mb/s: 4 bits wide at 25 MHz

Note:	There are also Reduced MII (RMII) and Serial MII (SMII) interfaces defined that are 2-bit and 1-bit wide, respectively.

FIGURE 6: IEEE 802.3™ 100 Mb/s LAYER DEFINITIONS



Reconciliation Layer: Maps the physical status (carrier loss, collision, etc.) to the MAC layer.

Media Independent Interface (MII) (Optional): Provides an n-bit transmit/receive interface to the PHY.

Physical Coding Sublayer (PCS): Encoding, multiplexing and synchronization of outgoing symbol streams (4B/5B encoding, etc.).

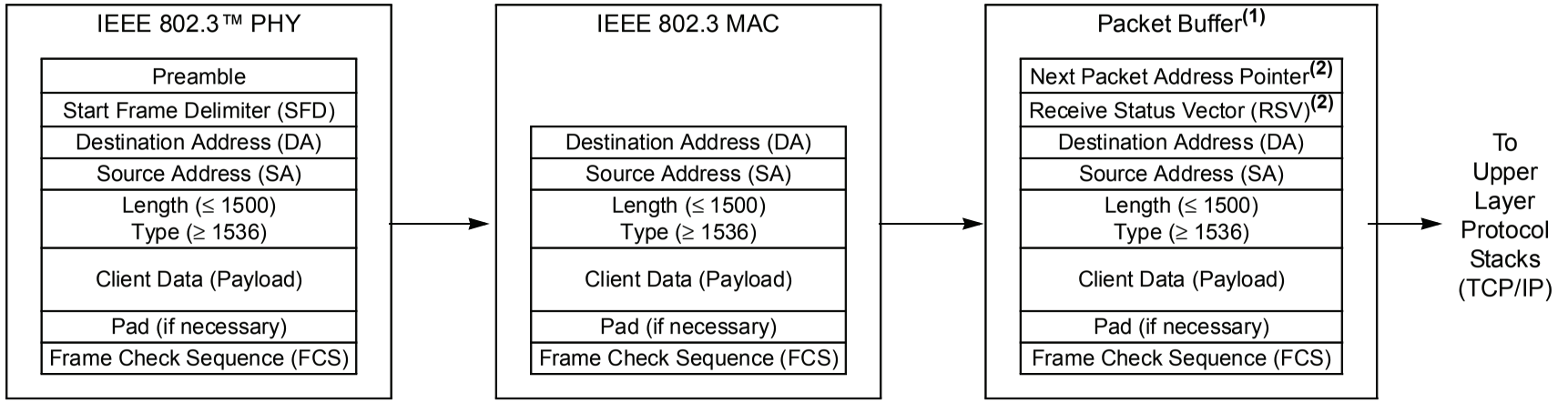
Physical Medium Attachment (PMA): Signal transmitter/receiver (serialization/deserialization of symbol stream, clock recovery, etc.).

Auto-Negotiation (Optional): Negotiation to the highest mode supported by both hosts.

Medium Dependent Interface/Physical Media Dependent (**MDI/PMD):** RJ45, etc.

Medium: UTP, Fiber, etc.

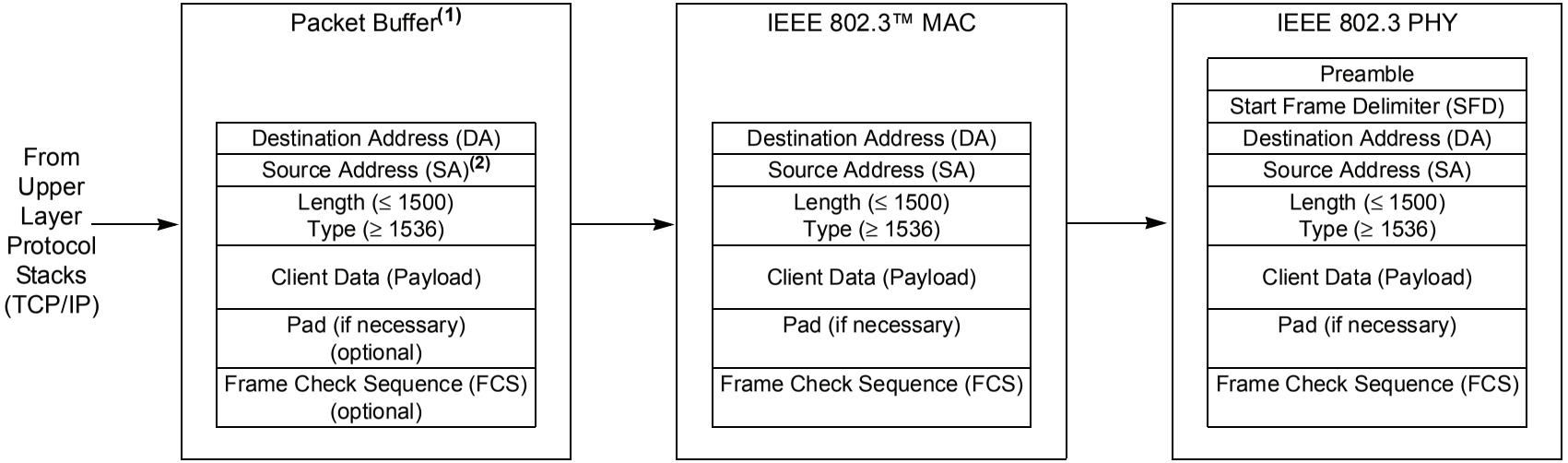
FIGURE 7: STREAM DECONSTRUCTION (RX)



Note 1: Devices other than the ENC family of devices from Microchip may implement the concept of a packet buffer in varying ways.

2: These two fields provide extra information over and above the standard IEE 802.3 frame.

FIGURE 8: STREAM CONSTRUCTION (TX)



Note 1: Devices other than the ENC family of devices from Microchip may implement the concept of a packet buffer in varying ways.

2: Some Microchip devices include functionality to automatically insert the source address into all outgoing frames. In this case, it is not necessary to store the source address in the packet Buffer.

STREAM TIMING

So far, we have discussed how data is assembled and disassembled into packets, and the role of the MAC and the PHY in doing so. What still remains is the actual transmission of the constructed stream over the physical medium.

Before we can understand the timing of IEEE 802.3 frames, we have to understand the reasons behind the timing.

Carrier Sense Multiple Access with Collision Detect (CSMA/CD)

Originally, Ethernet was designed as a protocol to run over a shared medium, as shown in Figure 9. In this topology, each node on the bus has equal access to the bus, but only one node may transmit at a time, and each node transmits half-duplex. Simultaneous transmission from multiple nodes would result in garbled data on the medium, and subsequent loss of data. From this simple example, we can derive some basic requirements for a network protocol:

- Multiple nodes must be able to transmit on a shared medium (Multiple Access).
- Each node must be able to detect when another node is transmitting (Carrier Sense).
- A transmitting node must be able to determine when simultaneous transmission occurs in the case where multiple nodes see the medium as Idle and start transmitting at the same time (Collision Detect).
- When a collision is detected, each node must have a method to determine when to retransmit without each node continually trying to retransmit at the same time (Backoff).

These requirements are met in Ethernet using a scheme known as Carrier Sense Multiple Access with Collision Detect (CSMA/CD).

Before an Ethernet node can begin transmitting, it must first determine whether the medium is active or Idle (Carrier Sense). If the medium is active, then that node must wait until the medium becomes Idle, and then waits a predetermined amount of time after that before starting to transmit. This predetermined amount of time is called the Inter-Packet Gap (IPG), also known as an Inter-Frame Gap (IFG), and is dependent on the speed of the bus, as shown in Table 2. The IPG is used as a recovery time between frames to allow nodes to prepare for reception of the next frame.

However, if multiple nodes are waiting for the medium to become Idle, then they may start transmitting at virtually the same time once the medium becomes Idle. Therefore, all nodes must also have the ability to detect these collisions (Collision Detect).

If the nodes that are trying to transmit on an Idle medium are at physically opposite ends of the medium, and one node starts transmitting just before it sees the transmission from the other node on the medium, then the worst case scenario occurs. As an example, let us assume that Node 1 and Node 4 in Figure 9 both want to transmit. Node 1 starts to transmit, but the data takes some time to propagate down the medium to Node 4. Node 4 starts to transmit just before it sees the data from Node 1. Node 4 will almost immediately detect a collision on the medium, and will transmit a special pattern known as a jam signal onto the medium. This jam signal must now return to Node 1 before it can detect that a collision has occurred by comparing its transmitted data to the received data. This applies to 10Base2, 10Base5 and 10Base-F nodes, where all nodes share a common medium.

FIGURE 9: SHARED BUS TOPOLOGY (10Base2)

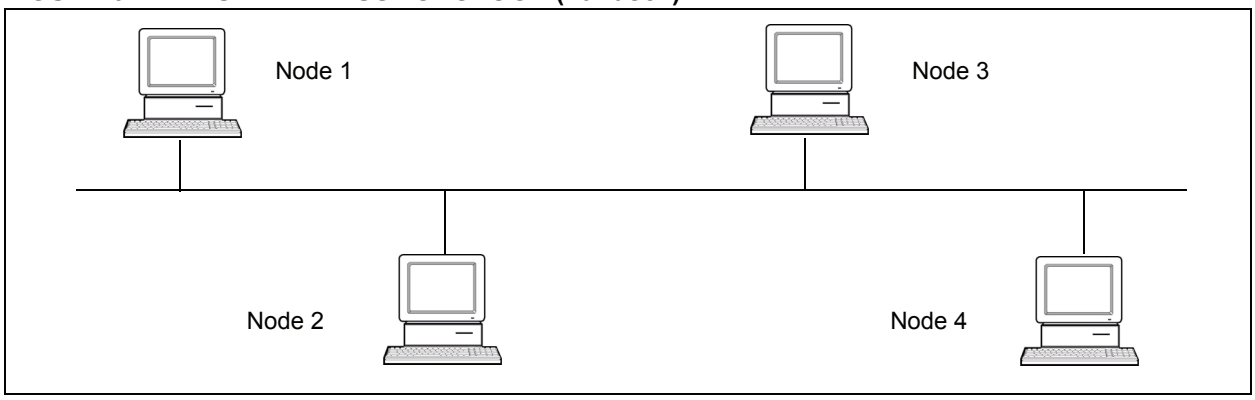


TABLE 2: KEY ETHERNET TIMING PARAMETERS

	Bit Time	IPG Time	Slot Time	Network Diameter (without repeaters)	Network Diameter (with repeaters)
10Base-T	100 ns	9.6 μs	512 Bit Times = 51.2 μs	100m	2500m
100Base-TX	10 ns	960 ns	512 Bit Times = 5.12 μs	100m	205m

This means roughly two times the signal propagation time of the network can occur before all nodes on the medium are ensured to have detected it. This time is known as the collision window or slot time. The slot times for various speeds are shown in Table 2.

The situation we have described is known as an “in-window” collision because the collision is detected within the slot time. If, however, the size of the network is larger than the network diameter, an “out-of-window” or “late” collision can occur. Late collisions are not considered a transmission error like in-window collisions, but are instead considered a problem with the network topology itself. Unlike in-window collisions, late collisions are not dealt with at the physical/data link layers of Ethernet, but rather must be detected and handled by the application software.

Based on the above example, it should be somewhat evident that the collision window is equivalent to the minimum size of the frame. However, increasing the frame size then increases the impact of recovering from a collision.

To this end, the original authors of the IEEE 802.3 specification compromised by coming up with a “reasonable” collision window (referred to as the “Network Diameter” in Table 2) for 10Base-T and 100Base-T Ethernet. The minimum frame size was then set to match the chosen network diameter. It would follow naturally that gigabit Ethernet, which runs at 1000 Mb/s, would have a network diameter 1/10 that of 100Base-T. However, this would result in a practically unusable network diameter of about 20m. Gigabit Ethernet extends the frame size by adding bits at the end of the frame (called “Carrier Extension”) to form an effective minimum frame length of 4096 bits. This results in a network diameter roughly the same as for 100Base-T.

Since the transmission rate for 100Base-T is 10 times as fast as the transmission rate for 10Base-T, the time required to transmit a frame is 1/10 the time. This, in turn, means the slot time is reduced from about 50 μs for 10Base-T to about 5 μs for 100Base-T. Consequently, the network diameter shrinks from 2500m to about 200m.

Note that half-duplex can be used on topologies that do not use a shared bus topology, such as a point-to-point connection (Figure 10). In this case, the TX line of one node is connected to the RX line of the other node, and vice-versa. Consequently, a collision is much easier to detect, as each node can simply look for data on its RX port while it is transmitting. If any data is received while it is transmitting, the linked node must be transmitting as well, and a collision has occurred. This applies to 10Base-T and all 100 Mb/s and gigabit Ethernet nodes.

The last requirement for our network protocol is a method by which each node determines when to retransmit. If every node tries to retransmit at the same time, collisions would continue ad infinitum.

For this reason, Ethernet implements what is known as a binary exponential backoff algorithm, which works as follows:

1. Each node chooses a random delay (in the range from 0 to 1) before attempting its first retransmit.
2. If another collision occurs, each node doubles the range of random delays (now from 0 to 3) and chooses a random delay again.
3. This process repeats (with a range of 0 to 7, 0 to 15, etc.) until no collision occurs or until 10 attempts have been made. At this point, the defined range for each node will be 0 to 1023. In this manner, the range of backoff times increases exponentially with each try, and the probability of a collision rapidly decreases.
4. Six more attempts (for a total of 16 attempts) will be made to retransmit. If a node is still unsuccessful at retransmitting, the frame is dropped, and an excessive collision error is reported. The application software must then detect the dropping of the frame and try to retransmit the dropped frame, if desired.

Full-Duplex Operation

While early Ethernet networks were implemented with a shared medium, and required CSMA/CD, most modern Ethernet networks are configured in a point-to-point (Figure 10) or a star topology (Figure 11), which can be thought of as a collection of point-to-point connections.

In either configuration, as each node is connected to a maximum of one other node, each node may operate in Full-Duplex mode. With a point-to-point/full-duplex configuration, collisions are not possible, and CSMA/CD is therefore not used. Each node may transmit whenever it wants to, within the constraints placed upon transmission by the inter-packet gap.

In addition, the total throughput of the medium is doubled (i.e., from 10 Mb/s to 20 Mb/s or from 100 Mb/s to 200 Mb/s).

Full-duplex operation also has the benefit of removing the limitations in network diameter due to slot times.

Note that not all medium types support full-duplex. In particular, the following types do not:

- 10Base2
- 10Base5
- 10Base-FP
- 10Base-FB
- 100Base-T4

FIGURE 10: POINT-TO-POINT TOPOLOGY

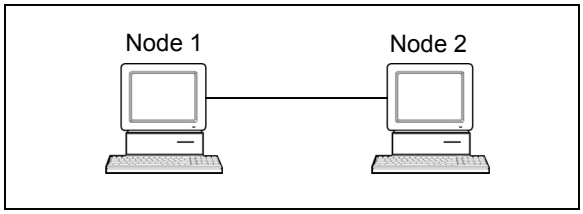
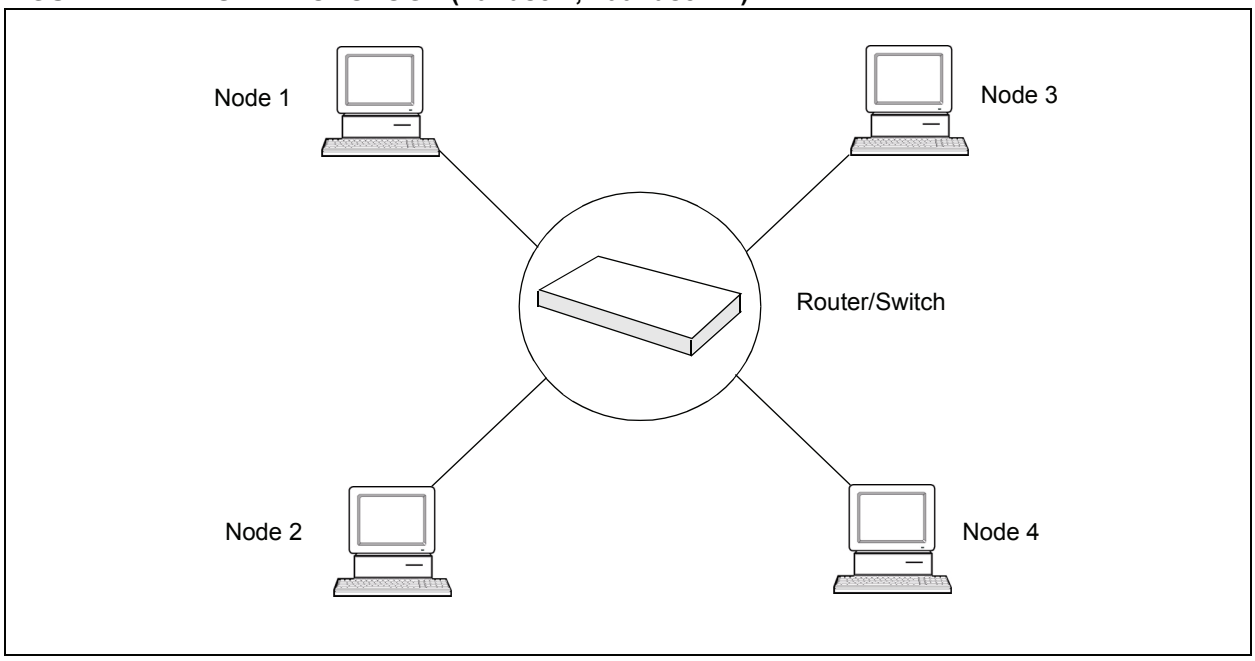


FIGURE 11: STAR TOPOLOGY (10Base-T, 100Base-TX)



10 Mb/s STREAM CONTENTS

There are distinct differences between a 10 Mb/s and a 100 Mb/s stream, so let us discuss the contents and signaling of the 10 Mb/s stream first. This section describes how the frame shown in Figure 3 is actually transported over the physical medium (i.e., CAT5 cable, etc.).

The first step in transmission of a 10 Mb/s stream is to encode the data to be transmitted using Manchester encoding. Manchester encoding encodes a logical '0' as a mid-bit low-to-high or high-to-low transition on the signal, and a logical '1' as the opposite transition. In Ethernet, a logical '0' is encoded as a high-to-low transition, while a logical '1' is encode as a low-to-high transition. See Figure 12 for an example.

Manchester encoding is used because it provides high reliability and the ability to extract the clock from the data stream. However, it requires double the bandwidth of the data to be transmitted.

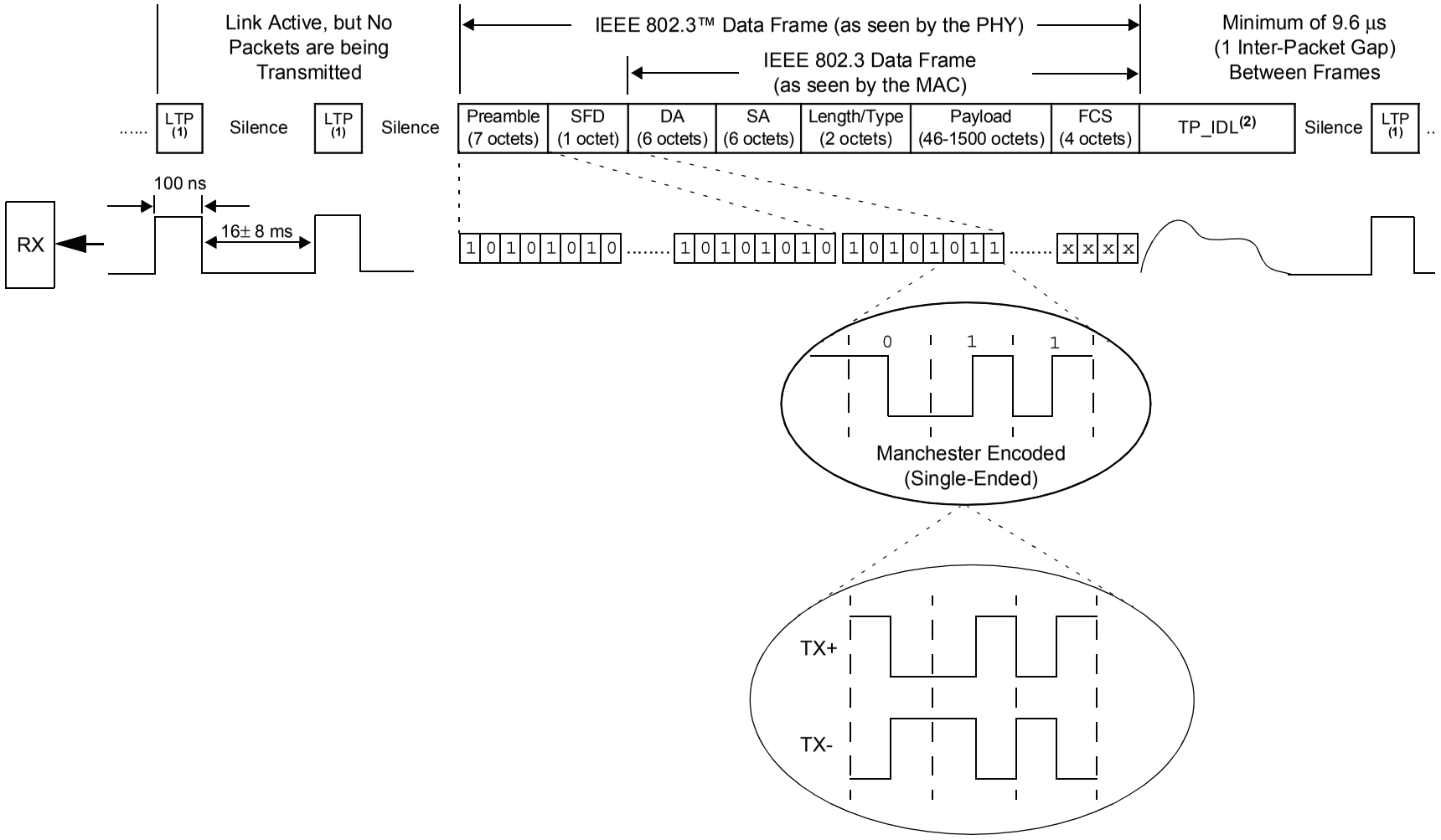
Since 10Base-T Ethernet uses differential signaling, the Manchester encoded signal is transmitted using differential signals, as shown in Figure 12.

The second step in transmission is to wave shape the output signals in order to meet the signal profiles defined in the IEEE 802.3 specification. These profiles are meant to ensure adequate signal propagation over the required lengths on the particular physical medium while minimizing unwanted EMI emissions.

Finally, the signal is transmitted over the cable using either voltage drive or current drive (which one depends on the particular IEEE 802.3 PHY) from an isolation transformer.

The differential voltage levels at the receiver are in the range of 350 mV to 3.1V.

FIGURE 12: 10 Mb/s ETHERNET STREAM(3)



Note 1: The Link Test Pulse (LTP) is also known as a Normal Link Pulse (NLP) and consists of a pulse that is approximately 1-bit time wide. The exact voltage profile for this pulse is described in the IEEE 802.3 specification, section 14.3.1.2.1.

2: The exact voltage profile for TP_IDL is described in the IEEE 802.3 specification, section 14.3.1.2.1.

3: Ethernet frames are transmitted Most Significant Byte (octet) first, Least Significant bit within an octet first.

100 Mb/s STREAM CONTENTS

Because Unshielded Twisted Pair (UTP) wires are low-pass in nature, the same encoding scheme that was used for 10Base-T will not work when we increase the speed by 10x, as is required for 100 Mb/s operation.

In addition, the power transmitted over certain types of physical links (i.e., phone lines, etc.) is limited to be less than approximately 30 MHz by regulatory guidelines. Therefore, a different encoding scheme is required for 100Base-T.

The encoding scheme used in 100Base-TX is known as Multi-Level Transition 3 (MLT3), and is shown in Figure 13. Each logical '0' or '1' is encoded as a transition to one of three levels (hence the '3' in MLT3). The transition is always to the closest voltage level, and always in the same order (-1, 0, +1, 0, -1, ...). A logical '0' is denoted by no transition, while a logical '1' is denoted by a transition.

As an example, consider the bit sequence '11111' shown in Figure 13. Since '1' always equates to a transition, a constant sequence of '1's will give us a transition on every bit, as shown in the figure.

By always transitioning to the closest voltage level, the transition times can always be minimized.

Because MLT-3 requires 4 transitions (-1 to 0 to +1 to 0 to -1) to complete a full cycle, the maximum fundamental frequency is reduced by 4, from 125 MHz to 31.25 MHz. This meets our requirement for power transmission at no higher than approximately 30 MHz.

The non-encoded signal frequency spectrum is 125 MHz, instead of the expected 100 MHz, because of 4B/5B encoding, which is discussed in the next section.

4B/5B Encoding

In addition to the physical encoding of MLT3, 100Base-TX introduces a logical encoding called 4B/5B, or sometimes "Block Coding". There are two primary requirements that 100Base-TX encoding must meet.

First, it must solve the problem of clock recovery in long streams of transmitted '0's. In MLT3, as you recall, a '0' is denoted by the lack of a transition in the transmitted signal. With no explicit clock, the transmit and receive nodes would soon become out of synchronization due to various jitter introducing effects. This would eventually result in the corruption of data.

Secondly, it must allow for transmission of not only data, but also of signaling codes, such as Start-of-Stream, End-of-Stream, Error and Idle.

The solution to these problems that 100 Mb/s Ethernet implements is to encode each 4 bits of data into 5 bits on the transmission medium. The translation from 4 bits to 5 bits is shown in Table 3. This means the actual transmission rate over the physical medium for 100 Mb/s Ethernet is 125 Mb/s.

If we look closely at the coding for all of the codes (except /H/, which is an error code), we will see the actual transmitted value always contains at least two '1's, which will result in a minimum of two transitions in the MLT3 waveform for any data transmitted. This addresses the issue of clock recovery.

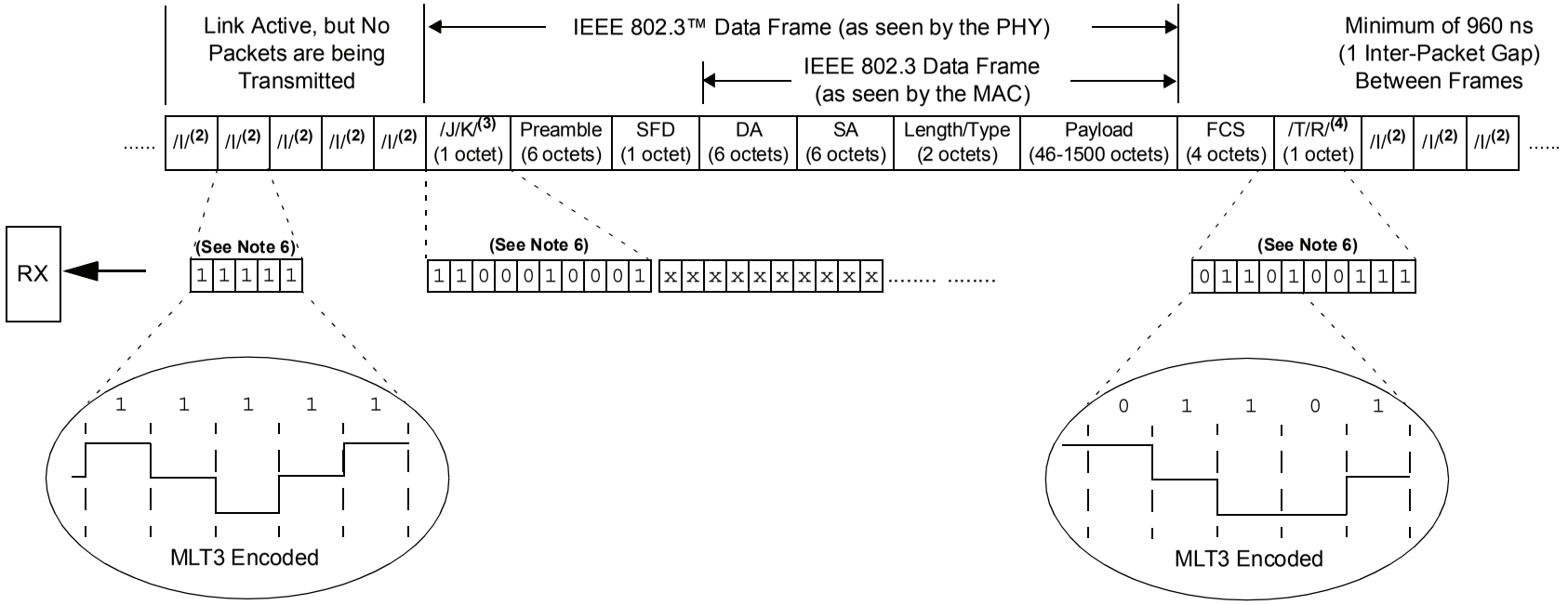
With 2^5 encodings for 16 data values, we now have 16 extra values that can be used to transmit signaling data. These include the following:

- Idle, which replaces the Normal Link Pulses (NLPs) used in 10Base-T
- Start-of-Stream Delimiter (SSD), which replaces the first octet of the Preamble in 10Base-T
- End-of-Stream Delimiter (ESD), which replaces the TP_IDL waveform used in 10Base-T
- Transmit error, which has no equivalent in 10Base-T

TABLE 3: 4B/5B ENCODING

Code	Value	Definition
0	11110	Data 0
1	01001	Data 1
2	10100	Data 2
3	10101	Data 3
4	01010	Data 4
5	01011	Data 5
6	01110	Data 6
7	01111	Data 7
8	10010	Data 8
9	10011	Data 9
A	10110	Data A
B	10111	Data B
C	11010	Data C
D	11011	Data D
E	11100	Data E
F	11101	Data F
I	11111	Idle
J	11000	SSD (Part 1)
K	10001	SSD (Part 2)
T	01101	ESD (Part 1)
R	00111	ESD (Part 2)
H	00100	Transmit Error

FIGURE 13: 100 Mb/s ETHERNET STREAM (AFTER AUTO-NEGOTIATION)[1,5]



Note 1: Due to 4B/5B encoding, each octet seen by the PHY/MAC corresponds to 10 bits on the physical medium. Thus, the data rate on the physical medium is actually 125 Mb/s.

2: /I/ = Idle code group ('11111').

3: /J/K/ = Start-of-Stream Delimiter (SSD) code group ('10001 11000').

4: /T/R/ = End-of-Stream Delimiter (ESD) code group ('00111 01101').

5: Ethernet frames are transmitted Most Significant Byte (octet) first, Least Significant bit within an octet first.

6: Unlike 10 Mb/s Ethernet, signals on the physical medium are scrambled to reduced radiated emissions. The bit streams shown are unscrambled and are shown for the sole purpose of understanding how MLT3 encoding works.

ENCODING/DECODING OVERVIEW

So far, we have discussed the encoding and decoding of 10Base-T, where Manchester encoding is used, and of 100Base-T, where NRZI, MLT3 and 4B/5B encoding are all used.

With Manchester encoding, where a single transition represents a single bit, the 10 Mb/s data rate translates into a 10 MHz bandwidth requirement on the medium. Noise immunity is added through the use of differential signalling on the medium.

How then, do all of the encoding methods employed on 100Base-TX combine to produce a final signal to be transmitted over the medium?

Figure 14 shows a simplified block diagram of a 100Base-TX PHY, with the bandwidth requirements at each stage. From this diagram, we can see that even though the effective data rate of the stream is increased to 125 Mb/s due to 4B/5B encoding, the required bandwidth of the physical medium is actually much smaller than 125 MHz.

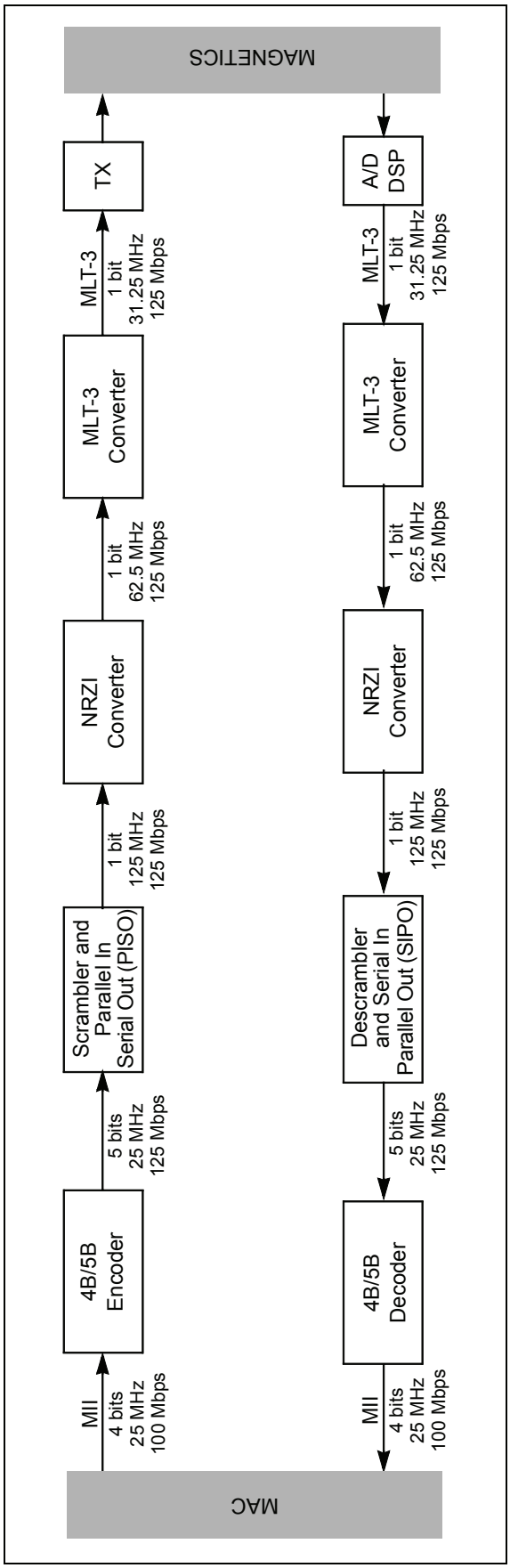


FIGURE 14: SIMPLIFIED 100Base-TX PHY BLOCK DIAGRAM

AUTO-NEGOTIATION

Auto-negotiation is the process by which two nodes communicate their respective abilities (speed, duplex, support for pause frames, etc.) in order to choose the highest common ability for both ends of the link. Auto-negotiation takes place at link initialization, and is backward compatible (i.e., does not break nodes that do not support auto-negotiation). Auto-negotiation is optional for 10Base-T and 100Base-T, but required for gigabit Ethernet.

Auto-negotiation is performed through the use of Fast Link Pulses (FLPs) shown in Figure 15. FLPs are similar to Normal Link Pulses (NLPs), but are transmitted in a burst of 17-33 pulses (called a link code word) between NLPs. Given the minimum inter-space timing of about 62.5 µs for FLPs, and the bits times of 100 ns (10 Mb/s) and 10 ns (100 Mb/s), it should be clear that FLPs are not interpreted as valid data by Ethernet nodes. In fact, FLPs are interpreted by nodes that do not support auto-negotiation as NLPs and are ignored. Nodes that support auto-negotiation, but do not receive any FLPs from the opposite end of the link automatically default to the lowest common ability (typically half-duplex 10Base-T) by default. In addition, some Ethernet PHYs have the capability to distinguish between 10 Mb/s and 100 Mb/s operation (based on the physical encoding seen on the link), a feature known as parallel detection. Of course, it is still possible to configure each end of the link manually to settle on a common ability, but this must be done in software.

It is possible that two nodes both support auto-negotiation, but no common ability is found. In this case, no link is established.

The maximum number of link code words sent by each node is not defined by the IEEE 802.3, but each node that supports auto-negotiation must be capable of transmitting an auto-negotiation base page.

The 16-bit (1 link code word) base page defines the abilities of the node, and is defined as follows:

Bits 0-4: Known as the selector field (S0-S4), this field defines the type of LAN technology used. For IEEE 802.3 Ethernet, this field is set to '10000'.

Bits 5-12: Known as the technology ability field (A0-A7), this field defines the capability of the node.

- **Bits 5-9:** Defines the link type, with the following defined priority (in order of highest priority to the lowest priority):

  100Base-TX full-duplex (bit 3 set)
  100Base-T4 (bit 4 set)
  100Base-TX (bit 2 set)
  10Base-T full-duplex (bit 1 set)
  10Base-T (bit 0 set)

- **Bit 10:** 0 = Pause not enabled
  1 = Pause enabled

- **Bit 11:** Supports asymmetric pause operation for full-duplex links

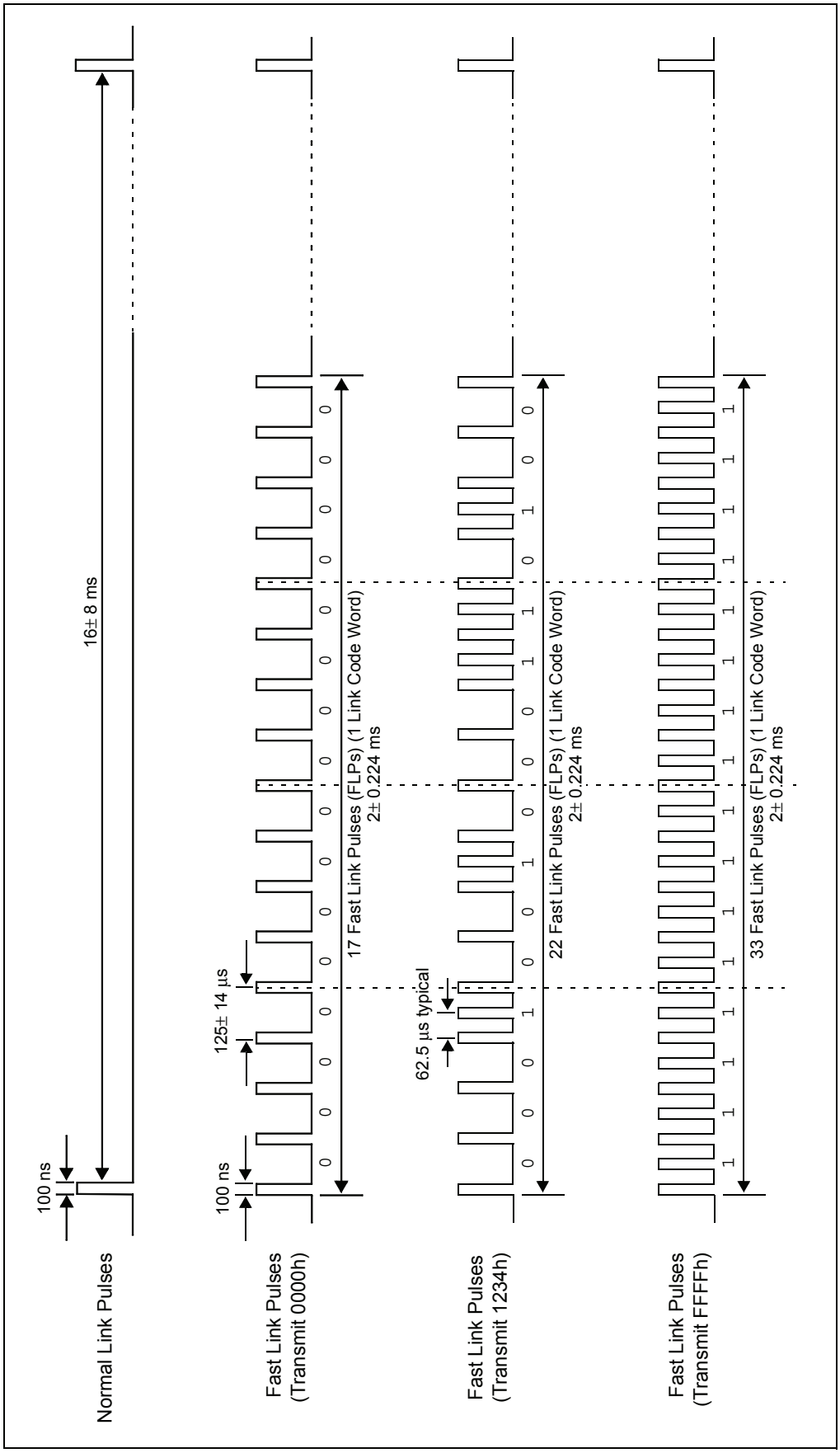
- **Bit 12:** Extended next pages bit, used only with gigabit Ethernet nodes.

Bit 13: Known as the Remote Fault (RF) indicator bit, this bit indicates a remote Fault.

Bit 14: Known as the Acknowledge (Ack) bit, this mandatory bit signals the receipt of an FLP message. An FLP message must be received identically three consecutive times before it is considered correct and Acknowledged.

Bit 15: Known as the Next Page (NP) bit, this bit indicates whether a next page link code word is following the base page. Next page words are used to transmit extra information between linked nodes during auto-negotiation, and is an optional capability.

FIGURE 15: FAST LINK PULSES



AUTO-CROSSOVER

In a properly configured Ethernet connection, the TX port of one node is connected to the RX port of the other node, and vice-versa.

In star topology UTP Ethernet networks, this crossover is typically done in the switch/hub/router's connection to the Ethernet jack. As a result, most UTP Ethernet cables have a 1-to-1 pin mapping between the connectors on the ends of the cable. Cables of this type are commonly referred to as "straight-through cables".

However, a different type of cable exists, called a "crossover cable". This type of cable internally crosses the TX and RX port on one end of the cable to the RX and TX port on the other end of the cable, respectively. This type of cable allows two end Ethernet devices to communicate with each other when directly connected as a point-to-point network. Additionally, crossover cables allow a switch/hub/router to communicate with another switch/hub/router. Using an incorrect cable type will not damage compliant Ethernet nodes, but neither node will be able to communicate or detect a link.

To eliminate cabling mismatches and reduce consumer frustration, a feature called auto-crossover may optionally be implemented in a node. When implemented, an auto-crossover capable node will automatically swap its TX/RX pins between TX and RX until a link is established. In this manner, either a crossover or patch cable may be used with the node with the same results. It is only necessary that one node in a linked pair implement auto-crossover. Most modern switches, routers, etc., implement auto-crossover.

Note that this functionality is different from "auto-polarity", where a node may automatically switch between positive and negative signals on a TX port or on an RX port. The two functions serve different purposes and are unrelated.

Auto-crossover is also sometimes referred to as Auto-MDIX, due to the fact that the crossover ("X" in Auto-MDIX) occurs at the MDI layer in the node (see Figure 6).

REFERENCES

The following documents are referenced in this application note:

- IEEE 802.3 Specification
- Associated IEEE Supplements (see Table 4)

TABLE 4: MOST COMMON SPECIFICATION SUPPLEMENTS

Supplement	Year	Description
IEEE 802.3a	1985	10Base-2 Thin Ethernet
IEEE 802.3c	1985	10 Mb/s Repeater Specification
IEEE 802.3d	1987	Fiber Optic Inter-Repeater Link
IEEE 802.3i	1990	10Base-T Twisted Pair
IEEE 802.3j	1993	10Base-F Fiber Optic
IEEE 802.3u	1995	100Base-T Fast Ethernet and Auto-Negotiation
IEEE 802.3x	1997	Full-Duplex Standard
IEEE 802.3z	1998	1000Base-X Gigabit Ethernet (SX, LX, CX)
IEEE 802.3ab	1999	1000Base-T Gigabit Ethernet over Twisted Pair
IEEE 802.3ac	1998	Frame Size Extension to 1522 Octets for VLAN Tagging
IEEE 802.3ad	2000	Link Aggregation for Parallel Links
IEEE 802.3af	2003	Power Over Ethernet (PoE)

NOTES:

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-572-9526
Fax: 886-3-572-6459

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

01/02/08