Technical White Paper

Real-Time Communication for Industrial Applications

Thomas Mauer

ABSTRACT

Selecting the appropriate processing system for industrial communication depends on the specific requirements of the application. For example, the communication needs of a remote sensor differ significantly from those of a controller like a PLC. Texas Instruments (TI) offers versatile solutions to meet these varying requirements, utilizing the PRU-ICSS and CPSW industrial Ethernet MACs. These MAC implementations support a wide range of industrial communication needs and are available across a scalable processor family, ranging from single-core Arm microcontrollers to multi-core Arm® microprocessors.

Traditionally, industrial communication was handled by a separate subsystem that interfaced with a microprocessor or microcontroller via parallel or serial connections. However, these interfaces could become bottlenecks, limiting data bandwidth and overall system performance based on the user application's demands.

To address these limitations, TI provides single-chip solutions that integrate the MAC, industrial communication stack, and application processor. These solutions reduce the bill of materials (BOM), PCB area, and cost, while also eliminating data exchange bottlenecks by leveraging high-speed internal bus interfaces. This integration ensures that process data transfer between the industrial communication stack/MAC and the application is efficient and unrestricted.

TI's industrial communication solution offers a complete package, including processors, Ethernet PHYs, clocking, power solutions, and evaluation modules (EVMs). These components are complemented by industrial communication stack libraries, software examples, and extensive technical support, providing a robust platform for developing industrial communication systems.

Table of Contents

List of Figures

List of Tables

Trademarks

EtherCAT™ is a trademark of Beckhoff Automation GmbH, Germany.
Code Composer Studio™ is a trademark of Texas Instruments.
Arm® and Cortex® are registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.
PROFINET® is a registered trademark of PROFIBUS Nutzerorganisation e.V. (PNO).
EtherNet/IP® is a registered trademark of ODVA, Inc.
All trademarks are the property of their respective owners.

1 Introduction

1.1 Real-Time Communication in Factories

Real-time communication in factories manages the data exchange between control units such as programmable logic controllers (PLC) and field devices including motor drives, sensors, and actuators. Depending on the manufacturing requirements, this data communication can occur within seconds to microseconds

In slower processes like those in the gas and oil industries, where temperature and pressure changes are gradual, communication occurs in the range of tens to hundreds of milliseconds. For factory automation with conveyor belts and automated machinery, data communication can occur in the range of hundreds of microseconds to tens of milliseconds.

In robotics and motor drive applications, data exchange must occur faster due to the need for rapid updates to motor controllers and position data, ranging from 31.25 microseconds (corresponding to a 32kHz control loop) to a few hundred microseconds.

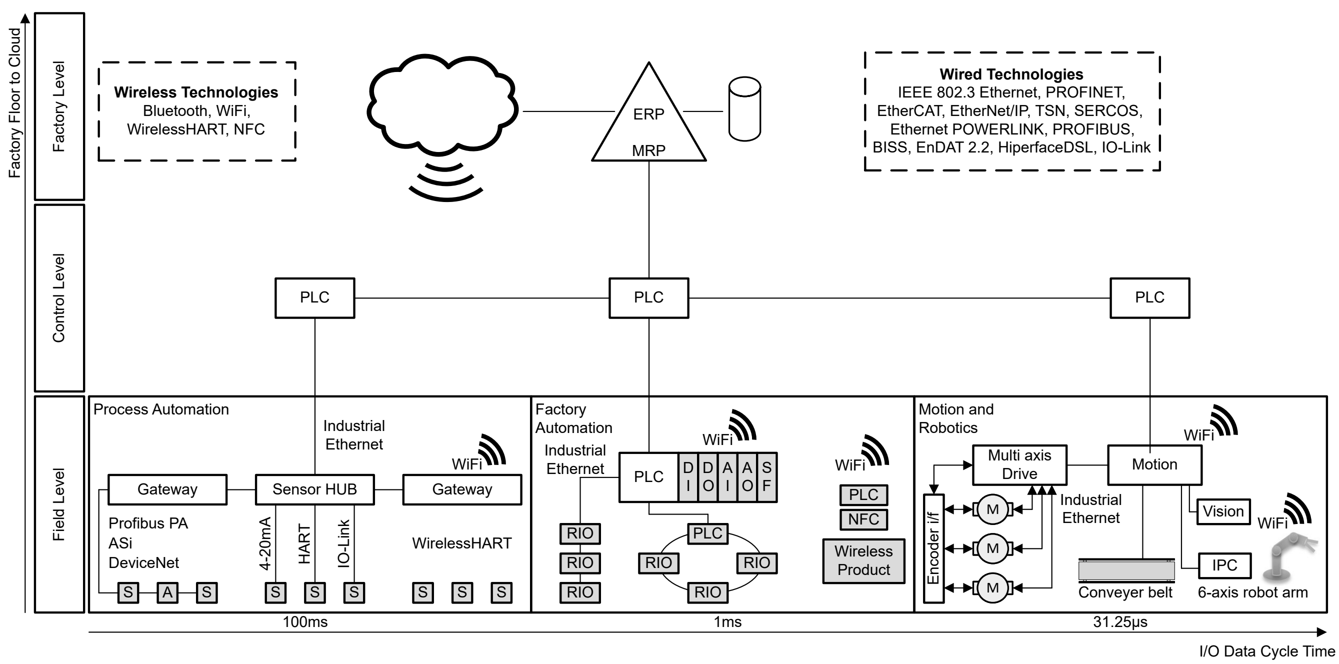

Figure 1-1. Real-Time Communication in Factory

PLCs report status and diagnostic information to upper layers, providing information to the plant control and operator. Industrial networks can also communicate with the cloud, enabling operators to monitor and control the plant from remote locations.

To achieve periodic data exchange, components have to meet requirements for different OSI model layers. For instance, the physical (PHY) layer and media access control (MAC) layer must fulfill specific real-time data communication requirements. Standard Ethernet MACs may introduce jitter and delay or may not support on-the-fly data processing required by specific industrial protocols.

Furthermore, the system's exposure to the internet necessitates security features to protect data communication. The European Cyber Resilience Act, for example, adds such requirements for plant manufacturers.

1.2 Industrial Protocols

Various industrial communication protocols exist, with no single dominant protocol. Different protocols have been established based on market segments such as process automation, factory automation, robotics, and motor drives. These range from serial communication protocols like HART and IO-Link to Ethernet-based protocols like EtherCAT, PROFINET®, and EtherNet/IP®.

In recent years, industrial Ethernet has taken the for front role over the serial based communication because Ethernet adds benefits, which are explained in Section 2.1.

Some protocols are designed to be interoperable with devices from different vendors, commonly seen in factory automation with PLCs, sensors, and actuators. The plan manufacturer purchases different types of vendors for a specific setup; therefore, different equipment from manufacturers needs to work together.

Other systems, such as CNC machinery or multi-carrier systems, which are enclosed systems, may use specialized or proprietary protocols that do not require interoperability with off-the-shelf devices. Nevertheless, such machinery are also built with standardized protocols if they do not need to take advantage of specific features of proprietary protocols.

1.3 Serial and Ethernet-Based Communication Protocols

Historically, serial-based protocols were used for factory automation due to their low cost and ease of use. Examples include DeviceNet, CanOpen, Profibus, and Modbus Serial. However, they have lower communication speeds and limited reach for high-speed communication.

With advancements in Ethernet technology, industrial Ethernet protocols like Ethernet/IP, PROFINET, and EtherCAT have become more prevalent, offering 100Mbps data rates over 100BASE-TX Ethernet. Newer protocols, such as Time-Sensitive Networking (TSN), PROFINET TSN, and CC-Link IE TSN, support 1000Mbps data rates.

2 Industrial Protocols

Various industrial communication protocols exist, with no single dominant protocol. Different protocols have been established based on market segments such as process automation, factory automation, robotics, and motor drives. These range from serial communication protocols like HART and IO-Link to Ethernet-based protocols like EtherCAT, PROFINET, and EtherNet/IP.

Some protocols are designed to be interoperable with devices from different vendors, commonly seen in factory automation with PLCs, sensors, and actuators. Other systems, such as CNC machinery or multi-carrier systems, may use specialized or proprietary protocols that do not require interoperability with off-the-shelf devices.

2.1 Ethernet-Based Communication Protocols

Historically, serial-based protocols were used for factory automation due to their low cost and ease of use. Examples include DeviceNet, CanOpen, Profibus, and Modbus Serial. However, they have lower communication speeds and limited reach for high-speed communication.

With advancements in Ethernet technology, industrial Ethernet protocols like Ethernet/IP, PROFINET, and EtherCAT have become more prevalent, offering 100Mbps data rates over 100BASE-TX Ethernet. Newer protocols, such as Time-Sensitive Networking (TSN), PROFINET TSN, and CC-Link IE TSN, support 1000Mbps data rates.

Industrial Ethernet offers several advantages over traditional serial fieldbus systems:

- **Higher Data Transfer Rates and Higher Bandwidth**: Ethernet supports speeds from 10Mbps to 10Gbps, compared to the lower rates of serial fieldbus. Ethernet also supports high-bandwidth applications like camera streaming, whereas serial fieldbus is limited to simpler tasks.
- **Scalability**: Ethernet is easily scalable with various networking options, while serial fieldbus has fixed node limits and complex wiring.
- **Standardization**: Ethernet is based on widely adopted IEEE standards and industrial Ethernet protocols are based on the Ethernet standard, with protocol specific enhancements and extensions.
- **Network Management and Diagnostics**: Ethernet provides advanced tools for network monitoring and diagnostics.
- **Reduced Wiring Complexity**: Ethernet requires single cable solutions without cable endpoint termination, reducing wiring complexity. Also single pair Ethernet (SPE) reduces the cable strings from 2-pair or 4-pair for standard Ethernet to 1-pair.
- **Enhanced Security Features**: Ethernet includes advanced security measures, unlike the basic security of serial fieldbus. The security function requirements for industrial Ethernet are getting currently defined by the different industrial protocol organizations, as they see the need to make industrial applications within a factory secure against cyber attacks.

Furthermore, there are significant difference between industrial Ethernet and standard Ethernet. Standard Ethernet is the Ethernet type that is used in the office and IT environment. Industrial Ethernet is used in industrial applications like factory automation, grid infrastructure and building automation to exchange periodically process data.

Table 2-1 provides the benefits and differences between industrial and standard Ethernet:

Table 2-1. Benefits and Differences Between Industrial and Standard Ethernet

	Industrial Ethernet	**Standard Ethernet**
Method of data exchange	Determinitic, managed frame exchange, scheduled data transmission to avoid frame collisions.	Best effort, on-managed frame exchange, frame collisions
Robustness	Signal immunity, harsh environments	Consumer environment
Content in data	process data, diagonostic data and Internet Protocol (IP) frames.	IP frames with any content (video, files, web-server).
Equipment type	PLC, remote IO, sensor, actuator, motor drives	PC, Laptop, printer, Internet
Location	Plant floor, factory, power station, building control	Office network, back-end
Media Access Control (MAC)	Specialized MAC implementation (ASIC, FPGA)	Standard Ethernet MAC as in every PC
MAC frame processing	On-the-fly and cut-though	Store-and-forward
MAC error handling	Error handling in MAC	Error handling in OLI layer 3 and above

2.2 Network Topologies

Control systems like programmable logic controller or motion controller are connected to sensors, actuators and drives on the factory floor via different network topologies. The topology is the method of wiring Ethernet cable in between different equipment so that all field devices are logically connected to the control system.

The network topology methods described below also depends on the type of industrial Ethernet protocol used, because some protocols require a specific connection method and do not allow to use the other described methods. For example, EtherCAT is typically wired in a line topology.

Some protocols do allow a combination of multiple network topology. For example PROFINET and EtherNet/IP combine line and star topology to logically connect all devices to the control system.

- **Line Topology**: Build as a linear connection from the PLC to devices. Each device has two Ethernet ports to forward the received Ethernet frame.

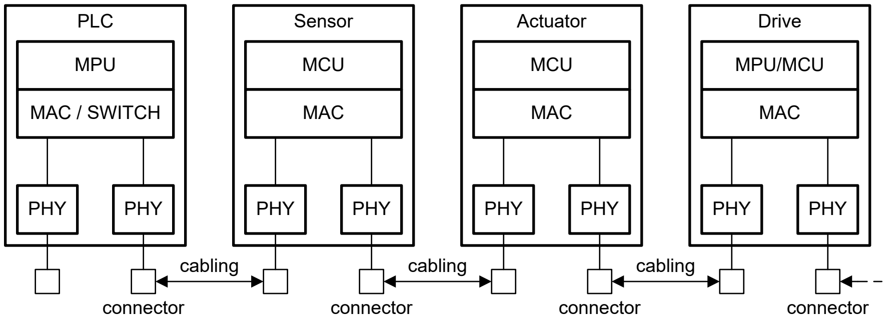

Figure 2-1. Line Topology

- **Ring Topology**: Adds redundancy to the line topology by connecting the last device back to the PLC. With some protocols the PLC sends out the Ethernet frame on both ports. In case there is a ring-break between two devices, for example the Ethernet cable was damaged and there in no link-up between the two devices, the PLC Ethernet frame still reaches all devices in the network with the required process data.

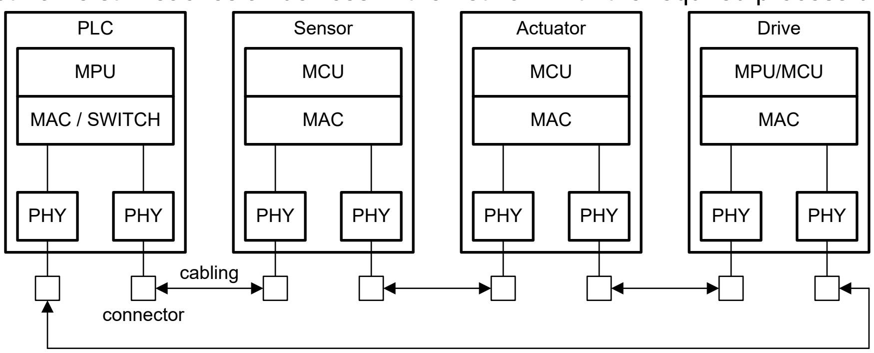

Figure 2-2. Ring Topology

- **Star Topology**: Uses multi-port Ethernet switches to distribute Ethernet packets. This is often combined with line topology. Note that typically specific industrial Ethernet protocol specific Ethernet switches need to be used, such that, off-the-self Ethernet switches does not work with an industrial Ethernet protocol or at least performance degradation is expected.

2.3 OSI Layer Model

The Open System Interconnection (OSI) layer model provides a conceptual framework that standardizes the functions of a communication system into seven distinct layers, facilitating interoperability and communication between different systems and technologies. Each layer serves a specific function and interacts with the layers directly above and below one layer, providing a structured and modular approach to the network communication.

The seven OSI layers are:

- Physical Layer
- Data Link Layer
- Network Layer
- Transport Layer
- Session Layer
- Presentation Layer
- Application Layer

In industrial Ethernet systems, the OSI model is used to standardize communication protocols and makes sure compatibility and interoperability among devices. Industrial Ethernet protocols, for example PROFINET, utilizes various OSI layers, particularly the physical layer for hardware connections, the data link layer for Ethernet frame communication, and the network layer for protocol specific addressing and routing.

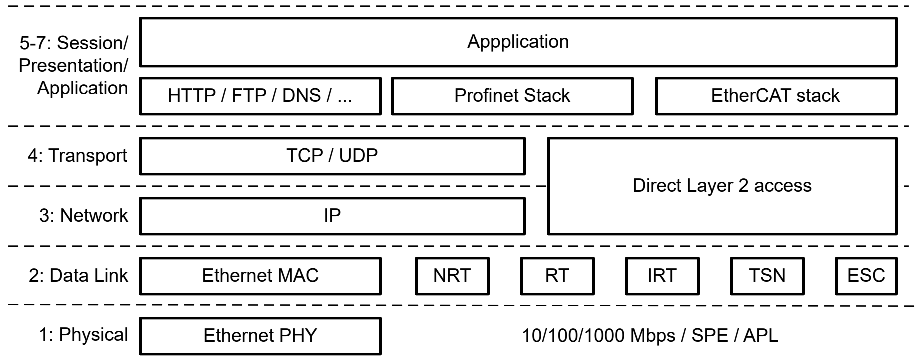

Figure 2-3. OSI Layer

The Physical Layer is handled by the Ethernet PHY. Depending on the protocol either a 10/100Mbit Ethernet PHY is used, or a 10/100/1000Mbit Ethernet PHY. New market trends also ask for single pair Ethernet (SPE) PHY, which also support speeds from 10Mbit/sec to 1000Mbit/sec.

The Data Link Layer is hosting the media access controller (MAC) implementation. Commonly used is here the standard Ethernet MAC for TCP/IP data communication. Industrial Ethernet protocols have some extensions to the MAC layer to support cyclic data exchange with high priority Ethernet frames and specific Ethernet frame handling like cut-through and on-the-fly.

Network and Transport Layer handle the Internet Protocol (IP) and UDP/TCP in Ethernet communication. Industrial Ethernet protocols can also use those three blocks, but more often the stack in Layer 5 of the industrial Ethernet protocol has direct Layer 2 access, and bypassing UDP/TCP and IP block.

The Session Layer runs the industrial Ethernet protocols stack, which is very specific to the industrial protocol.

The Presentation and Session Layer contain the industrial application, which is depends on the customer use case for the device.

2.4 Industrial Ethernet System Block diagram

2.4.1 Two-Port Device

The two-port Ethernet device is used for field devices, to simplify the support of line topology. As each device has already two Ethernet ports with Ethernet switch capability, the line topology can be simply realizes without the need of additional Ethernet hubs or switches.

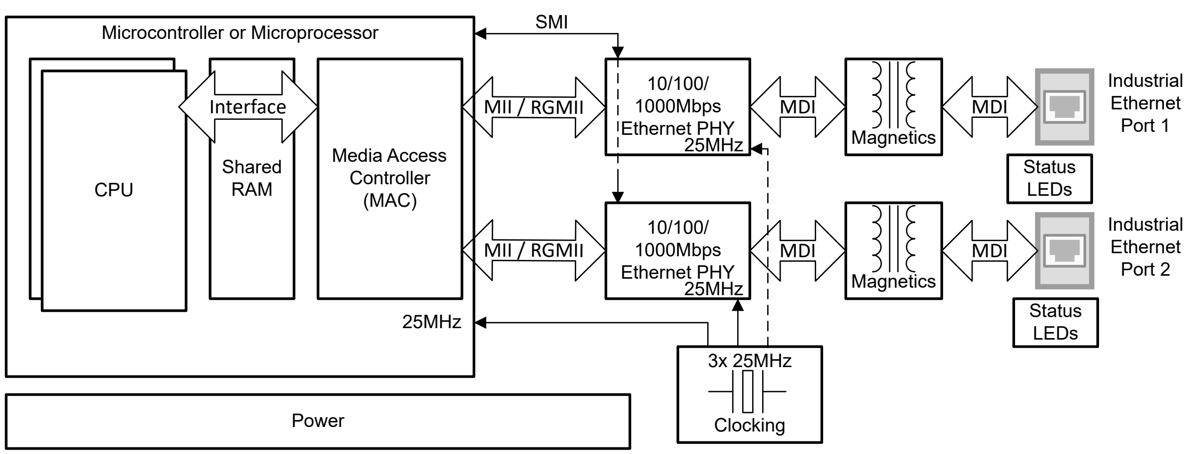

Figure 2-4. Two-Port Ethernet Device

The device with two physical Ethernet ports is also referred to as a 3-port switch. The device has two physical Ethernet ports and one logical port that connects to the host processor. Some protocols like EtherCAT require a 25MHz common clock for the two Ethernet PHYs and the MAC. This is to reduce RX/TX jitter when transferring the Ethernet frame from MAC to PHY and vice versa. When using independent 25MHz clock, the RX/TX jitter can be in the range of 40ns for 100Mbps and 4ns for 1000Mbps PHY speeds.

Depending on the industrial Ethernet protocol, either two 10/100 Mbps Ethernet PHYs or two 10/100/1000Mbps Ethernet PHYs are deployed. The PHYs connect via MII or RGMII interface to the Ethernet MAC. There is also a sideband signal called Serial Management Interface (SMI), consisting of MDIO and MCD lines, to enable register programming of the Ethernet PHY by the MAC. Programming the Ethernet PHY can be useful if the PHY needs to operate in a specific operation mode that is not already configured at the power up of the Ethernet PHY via the boot-strap configuration. For additional details on the boot-strap, see the *DP83826 Deterministic, Low-Latency, Low-Power, 10/100 Mbps, Industrial Ethernet PHY Data Sheet*.

The MAC implementation depends on the supported protocol. There is no common MAC as for example EtherCAT and PROFINET use different type of Ethernet frame handling. For the differences in Ethernet frame handling, see Section 2.6. The MAC makes the process data or Ethernet frames available via the shared RAM.

The CPUs runs the industrial Ethernet protocol stack, the peripheral drivers, additional functions (for example, web-server or UPC-UA database) and the customer application. Depending on the software architecture, those software tasks can be split across different CPU cores.

2.4.2 One-Port Controller

Figure 2-5 shows the system components for a controller, which for example can be a PLC or motion controller. The controller typically runs some PLC application like CoDeSys. The controller has one industrial Ethernet Port that connects to the field devices like sensors, actuators or motors. There is also a second Ethernet port that connects the controller to the Control and Factory Level (see Figure 1-1).

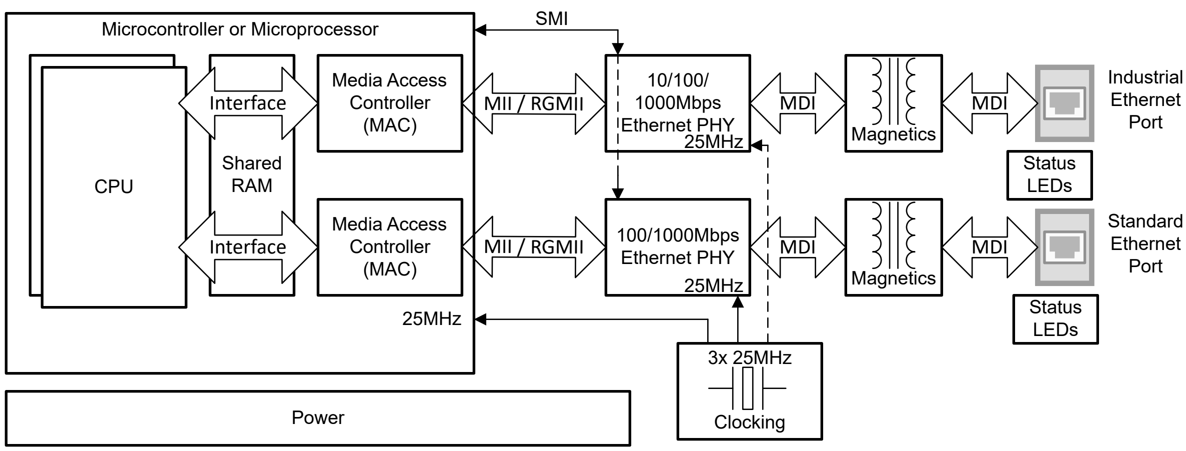

Figure 2-5. One-Port Ethernet Controller

The Ethernet port to the field device can use a specific MAC implemention, but some protocols like EtherCAT controller only requires a standard MAC with time triggered send function.

2.5 Ethernet Physical Layer (PHY)

Industrial Ethernet protocol can use 10BASE-T, 100BASE-TX and 1000BASE-T, with the majority of protocols currently using 100BASE-TX. The Ethernet cabling is using 2-twisted pairs for 10BASE-T and 100BASE-TX and 4-twisted pairs for 1000-BASE-T. The Ethernet data send on the cable pairs is full duplex, which means that the PHY can receive and transmit at the same time. Those Ethernet standards require a cable reach of 100m between two field devices. If a longer cable reach is needed, then a Ethernet hub or Ethernet switch has to be inserted into the Ethernet line.

For 10/100Mbps, shown in Figure 2-6, there are two pairs of Ethernet cable used: one pair is used for transmit and the second pair is used for receive.

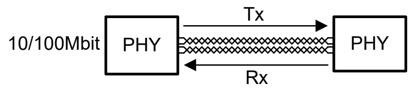

Figure 2-6. PHY 100mbit

For 1000Mbps, there are four pairs of Ethernet cable used. In Figure 2-7, the PHY transmit and receive simultaneously on all four pairs.

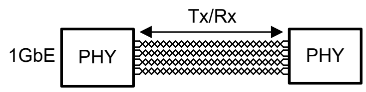

Figure 2-7. PHY 1gbit

The data transfer bandwidth depend on the established LINK speed, and is 10Mbps, 100Mbps or 1000Mbps. The Ethernet PHYs for those IEEE standards do support multi-speed. This means that a 10/100Mbit PHY support the two speeds of 10Mbps and 100Mbps, while a 10/100/1000Mbit PHY supports in addition also 1000Mbps.

Table 2-2 lists industrial Ethernet protocols required for specific PHY features.

Table 2-2. Industrial Ethernet Protocol PHY Features

Feature	Description	Protocol example
Fast-Link-Down (FLD)	The PHY needs to be able to drop an active link when receive errors occur. Such feature is mainly important to industrial protocols that need to loop back Ethernet frames to the controller, like in EtherCAT. A device with a link-down needs to switch to loopback mode once the link-down is detected. FLD works with PHY LINK LED state, which is fed back to the MAC. Note that alternatively the LINK state is read out by the MAC using MDIO/MDC protocol, and this may take several microseconds to detect.	EtherCAT requires <15µsec of FLD
Low transmit and receive latency	PHY latency has to be as low as possible, because in line topology the latency between PLC and last device does sum up.	EtherCAT requires a port to port pass through latency of <1µsec.
Fixed MDI/X and fixed speed	Typically the PHY performs autonegotiation with the link partner PHY. This means the two PHYs negotiate on the fastest speed and about the MDI connection and polarity. This negotiation takes time, and only after the negotiation the link is established and Ethernet frames are exchange. Configuring the PHY to a fixed configuration speed up the link-up time.	PROFINET fast-startup of <500ms

There is a new Ethernet standard adopted to the industrial use case called Single-Pair-Ethernet (SPE). SPE has the advantage of reducing Ethernet cable to one pair over the different Ethernet speed grades of 10Mbps, 100Mbps and 1000Mbps.

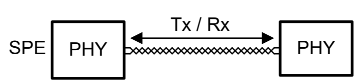

Figure 2-8. PHY SPE

The 10BASE-T1L standard supports 10Mbps of bandwidth with a cable reach of 1km (1000m). This is very good because standard Ethernet only supports 100m as referred to above.

100BASE-T1 and 1000BASE-T1 currently support only a cable reach of 50m and 15 m, which is a reduction of maximum cable length of standard Ethernet, but with the benefit of using 1-pair only.

2.6 Media Access Controller (MAC)

The media access control (MAC) is responsible for managing protocol access to the physical network medium. The main functions consist of:

- Addressing: Each device on the Ethernet network has a unique MAC address, a 48-bit identifier used to distinguish devices on a network. The MAC makes sure that data packets are delivered to the correct device.
- Frame Delimiting: The MAC sublayer defines the frame structure used for data transmission. An Ethernet frame includes a preamble, destination MAC address, source MAC address, type/length field, data payload and frame check sequence (FCS).
- Media Access Control: The MAC sublayer controls how devices access the shared communication medium. Industrial Ethernet MAC derive here from standard Ethernet MAC, and even implement specific operation modes like on-the-fly processing and cut-though data processing. Those methods are explained later in this section.
- Error Detection: The MAC sublayer includes mechanism for detecting errors in receive frames, typically using the Frame Check Sequence (FCS). If an error is detected, the frame is typically discarded and upper layer is getting notified.
- Frame Transmission and Reception: The MAC sublayer handles the actual transmission and reception of frames over the network medium (layer 1). This makes sure that frames are properly formatted, addressed, and send at the correct time. Some IE protocols have different transmission phases, one phase for real-time data (process data), and second one for non-real-time data (standard Ethernet frames). This transmission phases are time multiplexed and referred to as cycle time.

2.6.1 Device MAC

For a device, many times a three-port switch is used. A three-port switch refers to two physical ports, and one port that is connected to the host CPU over a shared memory interface.

The three-port switch does receive and transmit industrial Ethernet packets on the two physical ports. It makes forwarding decisions when receiving an Ethernet telegram from port one, to either put it into the shared RAM interface for the host CPU, or to send it out on Port two. Or it does both, putting the Ethernet frame into the shared RAM and forwarding it, in case the Ethernet frame is a broadcast frame.

Industrial Ethernet MAC for devices for industrial Ethernet typically work in one of the following frame processing modes shown in Table 2-3.

Table 2-3. Frame Processing Modes

MAC Frame Processing Mode	Description	Example
One-the-fly	The Ethernet frame is getting processed by the MAC while it is getting received. The MAC can extract bytes of the Ethernet frame, or insert bytes into the Ethernet frame without altering the length. The MAC does update the CRC checksum at the end of the frame to reflect any modifications. Typical port to port delay is less than 1µs in 100Mbps.	EtherCAT device
Cut-through	The MAC receives the first couple of bytes of the Ethernet frame, typically 16 to 32 bytes. The MAC then analysis the destination MAC and Ethernet frame type to derive a forwarding decision. Once the destination port of the frame has been determined, the frame is getting forwarded, while the reception of the remainder of the packet is still ongoing. The port 1 to port 2 delay depends on the Ethernet speed and is in the range of 1.5µs at Gbit speed and 4µs for 100Mbit speed.	PROFINET and EtherNet/IP
Store-and-forward	This is a legacy MAC mode that is supported by many network interface cards (NIC) found in standard PCs. The MAC receives the complete Ethernet frame and then only performs the forwarding decision. Because of this, the port 1 to port 2 delay depends on the size of the Ethernet frame. At Gbit the forwarding delay is 12.5µs, for 100Mbit it is 125µs.	Standard Ethernet network cards

Table 2-4 shows additional MAC features that some industrial protocol use.

Table 2-4. MAC Features

MAC Mode	Description	Example
Time Triggered Send (TTS)	This is a specific mode in the MAC to precisely time a transmission during the process cycle time. The MAC pre-loads the Ethernet frame into the transmit FIFO together with a transmission time. Once the transmission time expires, the MAC starts the transmission. This method is used for imitating a specific mode time slot in a time synchronized architecture, as well as for time synchronization	EtherCAT controller, Profinet IRT, TSN
Time stamping	The MAC captures time stamping for an received ethernet frame or an transmitted Ethernet frame. Timestamping occurs in the MAC. There are also Ethernet PHYs that include 1588 timestamping protocol. IEEE1588 can be also done in the Ethernet stack on the processor side, or in the industrial Ethernet MAC.	IEEE 1588, Profinet PTCP, TSN gPTP,

2.6.2 Controller MAC

The industrial Ethernet controller typically features one Ethernet port dedicated to connecting industrial Ethernet (IE) devices. Additionally, a second Ethernet port is often included to interface with the network backbone and intranet, providing diagnostic and status information to the management level. The first port's speed is determined by the specific industrial Ethernet protocol in use, commonly operating at 100Mbit/s. The second port usually supports higher speeds, typically 1Gbit/s, to handle the additional data traffic and management functions.

Certain industrial Ethernet protocols, such as EtherCAT, do not necessitate a specific MAC implementation. Therefore, the MAC requirement has to be evaluated based on the particular industrial Ethernet protocol being deployed.

The controller CPU frequently operates a High-Level Operating System (HLOS) such as Linux to support services such as PLC (for example, CoDeSys run-time) and OPC UA. Consequently, a processor with a robust architecture, such as a Cortex® A53 or Cotrex A73, is typically used to meet the performance demands.

2.7 Industrial Protocol Stacks

The industrial Ethernet protocol stack is responsible for managing the specific tasks required to facilitate reliable and deterministic communication in industrial environments. Designed to meet the stringent demands of industrial applications, the stack provides real-time data transfer, high availability, robustness, and seamless integration with various industrial devices and systems.

For example, in an EtherCAT™ device, the Media Access Control (MAC) layer is crucial for forwarding EtherCAT frames with minimal delay while performing necessary frame manipulation. As EtherCAT frames pass through each device, the EtherCAT MAC can insert or extract data at specific locations within the frame. The EtherCAT stack is responsible for configuring the MAC so that the MAC correctly performs these manipulations at the appropriate points in the frame.

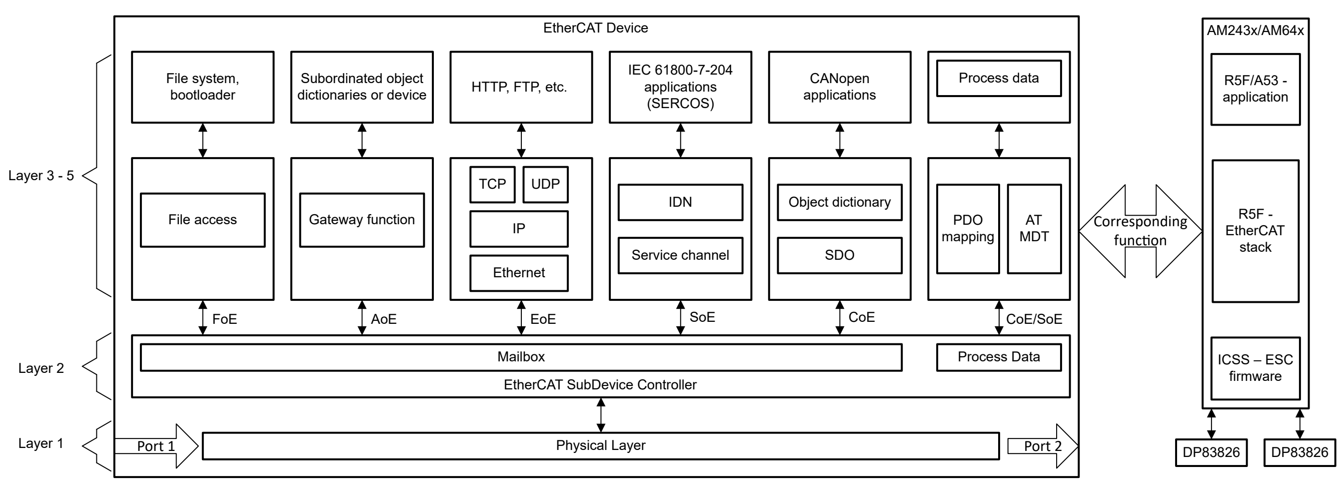

Figure 2-9. EtherCat Stack Example

The EtherCAT stack handles various functions, including:

- Frame Processing and Synchronization: Ensuring precise timing and handling of EtherCAT frames.
- Process Data Handling: Managing the exchange of process data through Process Data Objects (PDOs).
- Network Management: Overseeing network topology and ensuring stable communication between devices.
- Error Handling and Diagnostics: Monitoring and addressing errors within the network.
- Communication Services: Supporting services such as CAN Application Layer over EtherCAT (CoE), File Access over EtherCAT (FoE), Ethernet over EtherCAT (EoE), and Servo Drive Profile over EtherCAT (SoE).
- Slave Information Interface (SII): Facilitating the configuration and identification of devices.
- Functional Safety (FSoE): Ensuring that safety-critical data is handled securely within the EtherCAT framework.

While EtherCAT is a specific example, other industrial Ethernet protocol stacks provide similar functionality tailored to their respective requirements.

Texas Instruments (TI) offers pre-certified stack solutions integrated with TI chips, enabling customers to build their final products without needing to source the stack from third-party suppliers. This single-supplier approach streamlines the development process, providing access to industrial communication stacks, demonstration examples, demo boards, and comprehensive technical support directly from TI.

2.8 Industrial Communication Software Development Kit (SDK)

Texas Instruments (TI) offers an Industrial Communication Software Development Kit (SDK) designed to streamline the development and implementation of industrial communication protocols.

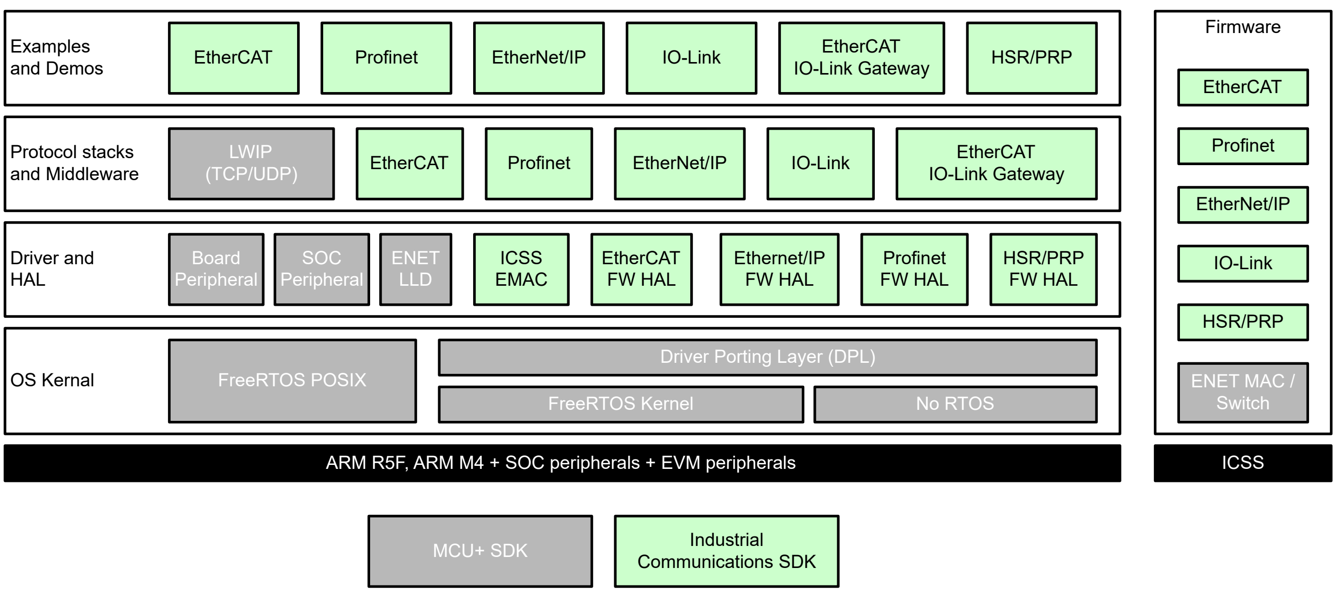

Figure 2-10. TI SDK Stack Example

This SDK provides numerous advantages for developers working in industrial environments:

- **Comprehensive Multi-Protocol Support:** The SDK supports a wide range of industrial communication protocols, including EtherCAT, PROFINET, EtherNet/IP, and more. These protocols are all compatible with the same processor family, leveraging the Programmable Real-time Unit and Industrial Communication Subsystem (PRU-ICSS).
- **High Performance and Real-Time Capabilities:** The SDK is optimized for high-performance Arm-based processors and includes dedicated hardware accelerators such as PRU-ICSS. It supports real-time operating systems (RTOS) as well as bare-metal implementations, delivering the deterministic performance required by industrial communication protocols.

- **Ease of Development:** The SDK simplifies the development process by providing a comprehensive set of protocol stack libraries, software tools, and example applications. Developers can evaluate the industrial communication solution on a TI Evaluation Module (EVM) and quickly jump-start development on custom hardware. Pre-configured examples and reference implementations reduce time to market and accelerate the development process.
- **Scalability and Flexibility:** The modular design of the SDK allows for easy customization and scalability to meet specific application requirements. The SDK is scalable across TI’s microcontroller and processor families, all of which support PRU-ICSS.
- **Integration with the TI Ecosystem:** The SDK integrates seamlessly with TI's development tools, including Code Composer Studio™ (CCS) and the system configuration tool (SysConfig), enhancing the overall user experience. It is also compatible with other TI products, including microcontrollers and analog components like ADCs, offering a complete solution for industrial applications.
- **Extensive Documentation and Support:** The SDK includes comprehensive technical documentation such as data sheets, application notes, and user's guides to assist developers. Direct technical support is provided by TI’s product engineers through the e2e forum, helping to resolve issues and guiding developers through the development process.

TI’s Industrial Communication SDK delivers various industrial Ethernet stacks, including EtherCAT, PROFINET, and EtherNet/IP. The firmware for the PRU-ICSS includes software binaries tailored for the PRU cores, with specific MAC implementations for each industrial communication protocol, such as EtherCAT ESC, PROFINET MAC, EtherNet/IP MAC, and IO-Link Master frame handler.

The SDK structure includes:

- **OS Kernel:** The SDK includes a FreeRTOS implementation, providing the necessary real-time operating system support.
- **Drivers and Hardware Abstraction Layer (HAL):** This section contains peripheral drivers and the HAL, which support the various industrial protocols.
- **Protocol Stacks and Middleware:** This section includes the industrial communication stacks described in Section [ToDo add reference].
- **Examples and Demos:** These provide user software for device-specific functions, such as sensors, actuators, and drives.

The documentation provided with the SDK thoroughly explains how to build and operate each example, making sure that developers have the required resources to build a successful product with industrial communication.

2.9 EtherCAT Device Example Using the AM243x Processor

This example demonstrates how to build and run an EtherCAT-device application using the AM243x processor with the AM243x LaunchPad (LP) development board. Figure 2-11 shows example instructions that will guide you through the process of setting up and executing an EtherCAT device demo.

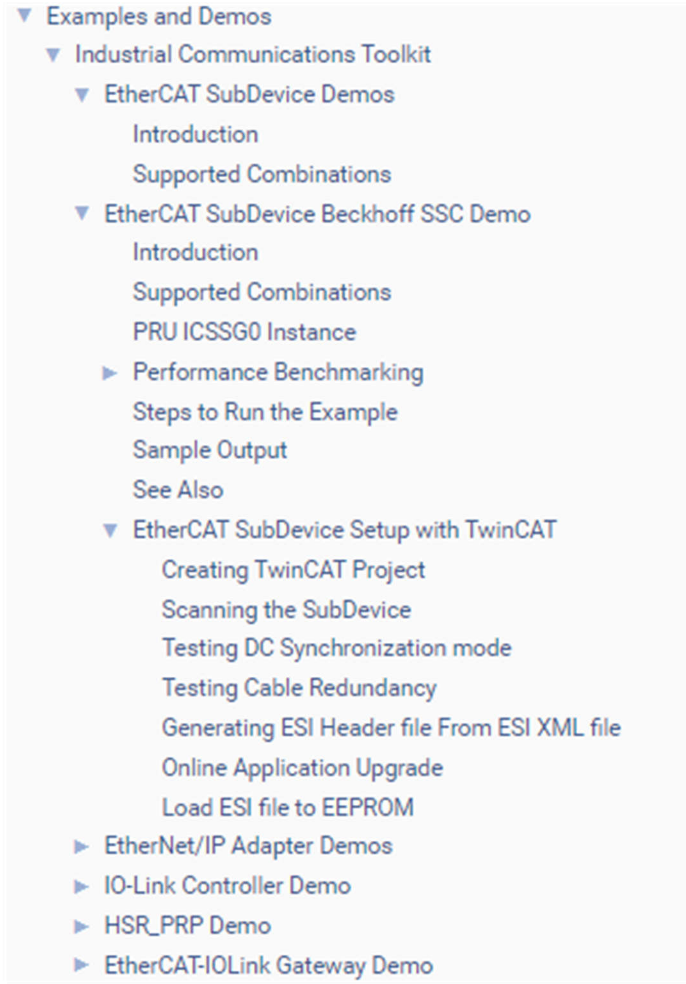

Figure 2-11. Screenshot of EtherCat Software Example

The step-by-step instructions for building the EtherCAT device demo can be found in the Industrial SDK documentation (example:), specifically in the *Examples and Demos* section. This documentation includes detailed guidance on:

- **Importing the Project into Code Composer Studio (CCS):** Instructions on how to properly import the EtherCAT demo project into CCS, including any necessary configurations.
- **Building the EtherCAT Demo Application:** Step-by-step guidance on how to compile and build the EtherCAT device demo using the provided project files and SDK tools.
- **Running the Example with TwinCAT EtherCAT Controller Application:** Detailed steps on how to execute the EtherCAT device demo and interface it with the TwinCAT EtherCAT controller application, demonstrating real-time communication and control.

By following these instructions, developers can effectively evaluate and experiment with EtherCAT communication on the AM243x platform, leveraging TI's comprehensive Industrial SDK.

3 Conclusion

Selecting the appropriate processing system for industrial communication depends on the specific requirements of the application. For example, the communication needs of a remote sensor differ significantly from those of a controller like a PLC. Texas Instruments (TI) offers versatile solutions to meet these varying requirements, utilizing the PRU-ICSS and CPSW industrial Ethernet MACs. These MAC implementations support a wide range of industrial communication needs and are available across a scalable processor family, ranging from single-core Arm microcontrollers to multi-core Arm microprocessors.

Traditionally, industrial communication was handled by a separate subsystem that interfaced with a microprocessor or microcontroller via parallel or serial connections. However, these interfaces could become bottlenecks, limiting data bandwidth and overall system performance based on the user application's demands.

To address these limitations, TI provides single-chip solutions that integrate the MAC, industrial communication stack, and application processor. These solutions reduce the bill of materials (BOM), PCB area, and cost, while also eliminating data exchange bottlenecks by leveraging high-speed internal bus interfaces. This integration ensures that process data transfer between the industrial communication stack/MAC and the application is efficient and unrestricted.

TI's industrial communication solution offers a complete package, including processors, Ethernet PHYs, clocking, power solutions, and evaluation modules (EVMs). These components are complemented by industrial communication stack libraries, software examples, and extensive technical support, providing a robust platform for developing industrial communication systems.