Deterministic Industrial Network Communication: Fundamentals

Ankush Meshram

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
ankush.meshram@kit.edu

Technical Report IES-2017-04

Abstract: Industrial networks came into existence with the third industrial revolution to support manufacturing and automation. Over the years, there has been technical advancement in different aspects of networking technologies in order to make production and governing automation efficient and intelligent. This also brought along advancing threats leading to the need of advancements in counterattacking or prevention methods. However, to contribute in challenging the Advanced Persistent Threats (APTs) the understanding of the fundamentals of industrial communication is needed. Determinism is at the core of automation, hence this report comprehends various literature sources on the industrial network communication strategies to achieve deterministic industrial network communication.

1 Introduction

Networks have become an integral part of manufacturing over the years replacing point-to-point communications at all levels. At lower levels in factory infrastructure, networks provide higher reliability, visibility and diagnosability and enable capabilities such as distributed control, diagnostics, safety and device interoperability. At higher levels, networks can leverage Internet services to enable factory-wide automated scheduling, control, and improve data storage and visibility. Industrial networks were introduced considering varying requirements of factory automation, distributed process control, home automation, and of critical

Infrastructures such as energy distribution as well as transportation. Appropriate networking technology evolved simultaneously within the application field. The three major influences identified for industrial network evolution [Zur14] are:

- Communication engineering for data transmission over large telephone networks in telecommunication sector.
- Instrumentation and measurements systems with parallel buses to account for limited data processing speed and real-time requirements for synchronization.
- Computer science with high-level communication protocol designs, such as WANs and LANs, leading to gradual change of analog to digital systems in telecommunication sector.

Fieldbus systems were the landmark in industrial networks evolution for automation, which replaced traditional expensive point-to-point cabling of devices to central control room. It brought concepts of decentralization, modularity to extend installations, and communication between intelligent devices, capable of data preprocessing, for transferring process data, and parameterization and configuration purposes. The idea of computer-integrated manufacturing (CIM) comprehended the structure of information flow required for automation in a hierarchical model — to create a transparent, multilevel network — called automation pyramid [SSKD11]. It comprised of 5 or more levels, in order of lowest level at the bottom to highest level at top: Field level (sensor/actuator), Process level (Programmable Logical Controller (PLC), Human Machine Interface (HMI), Cell level (Operator station), Factory level (Manufacturing execution systems (MES)), Company level (Enterprise resource planning (ERP)). Fieldbuses populated the field, process and cell levels while bridging the gap between lower levels which traditionally consisted of point-to-point connections to higher level networks. The distinction between lower level and higher level networks of the automation pyramid is maintained by fieldbus systems. The popularity of Ethernet as the LAN technology in automation and its penetration of all levels of this pyramid to process level is likely to replace mid-level fieldbus systems. Industrial Ethernet is resulted in reduction of the levels in the automation hierarchy, and ultimately to flattening out of pyramid to at most three or two levels.

In further sections, first we will build on communication fundamentals to understand industrial communication paradigms. This is followed by section 3 on

relation between industrial communication and determinism. In section 4, we elaborate on the foundations of deterministic industrial protocols and ending the report with short summary.

2 Fundamentals of Industrial Communication

2.1 Communication Layers

Introduction of the ISO/open system interconnection (OSI) seven-layers reference model for data communication has been the foundation for development of complex communication protocols [Zim80, DZ83] (Figure 2.1). There are three important concepts to understand before dwelling into layers:

1. *Protocol*, is a set of rules and convention that communication layer N of open system must confer to communicate with layer N of another open system. Rule sets of each layer define the respective layer protocol.
2. *Service*, defines the functionality of services offered by one layer (service provider) to layer above it (service user). The OSI model doesn't enforce the way services are implemented.
3. *Interface*, specifies interface between layers with services offered by the lower layer to the upper layer. It also defines access methods with parameters and what results to expect.

An application system sends information to another system through packaging data at top layer and requesting services of layer below to transmit data, repeating until lowest layer. On the way down the layers, the data of the application process are augmented by layer-specific data needed to execute the respective protocols. These data are typically address and control information that is mostly combined in a protocol header. In addition, the data may be segmented into individual packets to match the allowed maximum packet size for a given layer. This way, the number of bits being actually transmitted can be significantly larger than the pure user data provided by the application process,and the communication overhead can be substantial. The receiving system strips additional information of the peer layer to recover data for the application process. The layers of the OSI model, bottom-up, are briefly described next.

- Layer 1, or Physical Layer, presents all mechanical, physical, optical, electrical, and logical properties of the communication system to upper layers necessary for transferring data frames.

- Layer 2, or Data Link Layer, is responsible for data frame formation from bits with frame's coding and checking for transmission errors (via Cyclic Redundancy Check). It is subdivided into the logical link control (LLC) and medium access control (MAC). LLC takes care of error detection mechanism and sets up the connection to layer 3. MAC links to layer 1 and controls who is able to transmit when.

- Layer 3, or Network Layer, establishes routing paths between origin and destination nodes of end-to-end connections while assigning special target addresses. The transmission paths are optimized to reduce congestion in the presence of multiple physical transmission mediums with varying transmission speeds and not exceed maximum allowed delay.

- Layer 4, or Transport Layer, sets up the end-to-end connection and splits up the data in small numbered packets whenever either the data size is too big or transmission times are long. The peer layer on the receiving system takes care of recombining the individual packets in the right order.

- Layer 5, or Session Layer, synchronizes the communication between participating systems while handling the authentication and identification of devices. To perform its synchronization task effectively it introduces any synchronization markers to resume after communication breakdown.

- Layer 6, or Presentation Layer, codes the transmitting data and its interpretation on the receiving system. It interprets the syntactic bit sequence of data into a character and its semantic meaning, such as currency or physical units.

- Layer 7, or Application Layer, provides the interface between the application and the communication unit for transparent representation of communication. It defines the procedures or protocol processes of various application functions for calling up data, file transfer, etc. It is designed in such a way that a system accesses information through its communication unit without the need to know the functions of underlying layers.

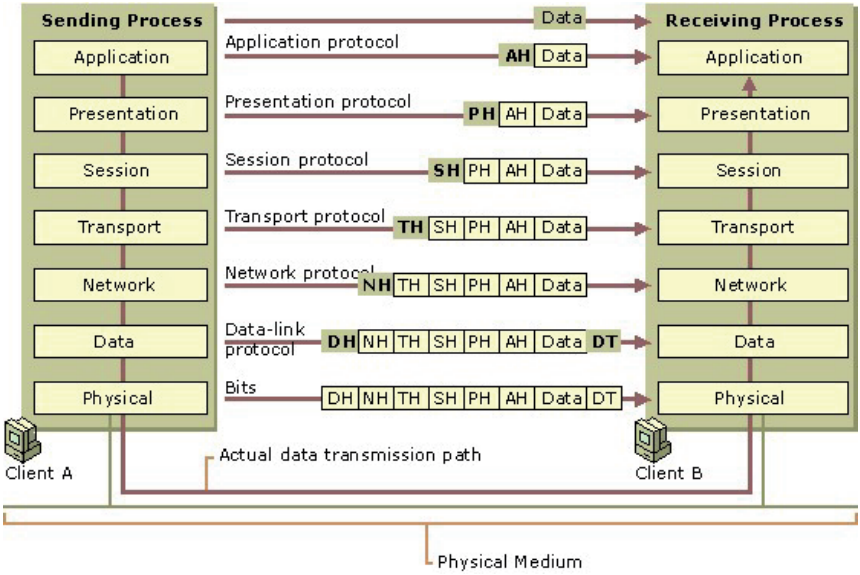

Figure 2.1: ISO/OSI stack with frame headers [Shi12].

Following the hierarchical OSI model, it is possible to set up complex communication system between heterogeneous systems on different layers. Through the use of repeaters, one can overcome the limitations of a given physical layer. The interconnecting devices share a common data link layer. Bridges interconnect different networks by translating data and protocols on layer 3. Routers link networks on layer 4, whereas gateways (especially, application layer gateways) interconnect entirely different communication systems on the application layer.

2.2 Communication Types and Services

Two distinct techniques are used in data communications to transfer data – connection-oriented and connection-less [Zim80]. A connection-oriented method (virtual circuit service) requires a session connection be established before any data can be sent. This method guarantees that data will arrive in the same

order. Connection-less method (datagram service) doesn't require a session connection between sender and receiver. There is no guaranteed data arrival however it is useful for periodic burst transfers. TCP (Transmission Control Protocol) is a connection-oriented transport protocol, while UDP (User Datagram Protocol) is a connectionless network protocol, both operating over IP at the Transport Layer.

The interface between neighboring service provider and service user layers is called service access point (SAP). The user data of layer N to be sent across the network for its peer layer is encoded as service data unit (SDU) and passed on to lower layers in the hierarchy for further processing. The communication between two peer layers is governed by the rule sets (protocol) that can only be understood by them. This information is added to SDU with interface control information (ICI) and protocol control information (PCI) to form a cohesive protocol data unit (PDU).

There are four primitive operations through which interaction between layers occur – Request, req, Indication, ind; Confirmation, con; and Response, res. Service user layer invokes request and response while resulting confirmation and indication originate from corresponding service provider. The combination of these primitive operations can be categorized into 3 major service categories – unconfirmed service, confirmed service, and acknowledged service. The unconfirmed and confirmed service comprise of a request, an indication and a confirmation, whereas acknowledged service comprise all the operations.

2.3 Communication Mechanisms

The multitude of application domains of automation systems have different timing and consistency requirements which varies too within the application areas such as manufacturing, process automation, etc [Tho05].

The timing behavior of a technical process can be conceptualized as either state-based or event-based. In the state-based approach, the status of internal state variables (temperature, pressure) of the process are continuously sampled and transmitted in discrete-time for continuous process control and monitoring. The event-based approach transmits data only when the process state changes and are well-suited for discrete processes or subprocesses which can be modeled as a state machine.

The traffic in industrial networks could be periodic or aperiodic based on how often process data is accessed. Periodic, or cyclic, traffic follows time-slot-communication strategy where each state variable is assigned a dedicated slot in the available bandwidth based on the a priori information of its data generation rate or sampling time. The update rate in a periodic traffic is usually dynamic adapting to the sampling rate demand of current process state information and the data exchange is connection-less. The periodic data is handled through buffers following the FIFO (first-in-first-out) structure where older values are overwritten by latest data. Aperiodic, or acyclic, traffic is generated on demand in an event-based manner and transmitted when free communication bandwidth is available or idle time is reserved between time slots of periodic traffic. Aperiodic data is connection-oriented where acknowledgment is used to allow for re-transmission of lost configuration data. Queues handle aperiodic data where messages are not overwritten and no new data is accepted when the queue is full.

Based on the consistency of accessing information in automation systems, the traffic data can be classified as continuously updated process data and on-demand parameterization data. The process data are real-time data and could be periodic or aperiodic requiring strict delivery timing to be meaningful for process control. The process data in the events of transmission errors could be reconstructed from historical data via interpolation. However, aperiodic process data additionally requires that no data lass occurs or atleast detected in due time through appropriate mechanisms. The parameterization, or configuration, data are non-real-time and usually aperiodic though session information, authentication information or updated communication parameters are transmitted periodically or quasi-periodically. The configuration data contains necessary information to set up or adjust the operation of automation system and needs guaranteed consistent delivery across the system.

2.4 Network Topologies

The star topology was the default wiring structure in automation before fieldbus was introduced. The PLC is at the center connected to I/O elements individually. The line, or bus, topology evolved as most efficient replacement to star-like point-to-point cabling and was quickly adapted network topology. The nodes are all connected in one single line. In the ring topology, nodes are arranged one after another in the form of a chain where each nodes has two independent interfaces

for input and output. It is very fast and deterministic method to exchange data with low jitter (variance of time delay) as nodes don't need explicit addressing. A variant of the ring topology is daisy-chain structure where nodes are cascaded like a string of pearls. In the tree topology, nodes are arranged in hierarchical composite network structure where each node could be a root for a lower-level segment. The root nodes usually have routing capabilities, so that the data traffic can at least partly be confined to individual areas of the network. In the mesh networks, there exists multiple paths between nodes through the network. It requires appropriate routing strategies to keep messages from circling in the network and causing congestion.

2.5 Medium Access Control

The topology being used by the networking technology influences the selection of the medium access control (MAC) method, or vice versa [Zur14].

The data transfer mechanism can be classified into two – single-master (or master-slave) and multimaster. In the single-master approach follows centralized communication architecture where the master either retrieves data from its slaves following request-response communication or synchronizes time slots with slaves to send their data. Such networks are usually single-segment structures with limited size and found at lowest levels of automation pyramid. Within the multimaster approach, participating nodes have equal rights over the communication medium and share it in a democratic way. These networks are found on the middle level of automation pyramid.

Time division multiple access (TDMA) is the actual MAC strategy used over multiplexing methods such as frequency division multiple access (FDMA), code division multiple access (CDMA), or space division multiple access (SDMA) for industrial communication. In TDMA, the network nodes shares the bandwidth and communicate sequentially. The basic methods of multiple access follows either centralized approach by polling or time-slot-based techniques, or in decentralized way by token passing or random access methods.

Polling is a master-slave mechanism where a slave node sends information only when explicitly called upon by the master node. In the network, alternate poll messages from master to each of its slave and responses from them is observed. Polling is strictly cyclic where the master polls all the slaves sequentially and

restarts the cycle. It's cyclic behavior suits well for periodic traffic where process variables are polled equidistantly. Polling polls data data either by explicit node addressing or process variable identifier irrespective of device generating the values. The later variant is also called as central polling. Strict polling doesn't offer functionality for aperiodic traffic and slaves cannot become active themselves to send event of an alarm condition. However, there exist alternate mechanisms to rectify these disadvantages.

Time-slot-based method divides available transmission time on the medium into distinct slots where slaves access the medium at its assigned time slot. The cyclic polling in its essence too partitions the polling cycle into time windows, however in time-slot-based method slaves can send the data themselves without a request from central master. Time-slot-based methods are mostly referred as TDMA. Synchronous TDMA equally distributes time slots whereas asynchronous TDMA dynamically distributes time slot according to amount of data to be sent. Aperiodic traffic is accommodated between the cyclic time slots. Based on the mechanism incorporated to synchronize slots there are two variants of TDMA – centralized and decentralized. In the centralized approach, a dedicated master sends some sort of synchronization message at the start of the cycle followed by nodes exchanging data in their pre-assigned time slots. On the contrary, in the decentralized approach all the nodes synchronizes themselves without explicit node to initiate cycle. Either explicit clock synchronization mechanisms or set of timers that set operation to a stable state state are used by the nodes.

Token Passing (TP) method of medium access is based on a special piece of information, called token, passed on between peer network nodes and only the node possessing it can initiate the data transfer. A set of rules ensure its fairness and its detection when lost or duplicated. Compared to time-slot mechanisms, when a node possessing the token doesn't have any data to send it passes the token on to the next node thus saving time. TP can be implemented either explicitly through a dedicated short message or implicitly through distributed, synchronized access counters (ACs) included in all nodes. The explicit form of TP uses target token rotation time (T_{tr}) to enforce the maximum time duration to posses the token. The implicit TP uses two counters, ACs and Idle Bus Bit Period Counter (IC), included in every master to simulate the token.

Another peer-to-peer communication based medium access method is random access where a network node tries to access the communication medium whenever it wants to without any imposition. This is also called carrier sense multiple access (CSMA). However, the major drawback of this approach is collisions when several nodes try to send data at same time, even if they noticed idle communication line before sending. The variants of CSMA deals with collisions in different ways to avoid bandwidth wastage and communication delays. In CSMA-CD (collision detection), collisions are detected by sending nodes which aborts the data transfer and wait for a random time before trying to send again. In *p*-persistent CSMA variant , the waiting time depends on the value of probability (*p*) that the node will try again in a certain time interval after collision. The probability of each node is adaptable to the estimation on its backlog and the monitored network load. The widely used CSMA-CA (collision avoidance) variant uses asymmetric symbols for coding the bits on the communication line, so that when two different bits are sent at a time, the dominant one wins over the recessive one. It is also called CSMA-BA (bitwise arbitration).

2.6 Communication Paradigms

There are 2 basic communication paradigms governing the information exchange between two or more network entities [Zur14]. The first approach is built upon the cooperation of actions or functions into which more complex process can be decomposed. This paradigm is called client-server, where the responsibility to interpret information lies with the sender. The service or data providing entity is called server, and the service requesting entity is called client. The server becomes active only when its services are requested by the client, hence this paradigm suits well for state-based traffic handled in some scheduled manner.

The other approach concentrates on exchanged data rather than actions and the responsibility of its interpretation lies with the receiver. The publisher-subscriber and producer-consumer paradigms follows the data-oriented approach. In the publisher-subscriber paradigm, the information is produced by the publisher and mulicasts on the network to be listened by the subscribers. The producer-consumer model is similar to the publisher-subscriber and only differs in broadcast communication of information. There are two variants of publisher-subscriber models based on how the information exchange is initiated – pull-type, the publishing action is triggered by a centralized publishing manager,

	Client-Server	**Producer-Consumer**	**Publisher-Subscriber**
Communication relation	*Peer-to-peer*	*Broadcast*	*Multicast*
Communication type	*Connection-oriented*	*Connection-less*	*Connection-less*
Communication service	*Confirmed,unconfirmed, acknowledged*	*Unconfirmed, acknowledged*	*Unconfirmed, acknowledged*
MAC type	*Mono-master(polling, centralized TDMA), multimaster (CSMA, TDMA or Token Passing)*	*Multimaster (TDMA, centralized polling or random access)*	*Multimaster (TDMA, centralized polling or random access)*
Application class	*Parameter transfer, cyclic communication*	*Event notification, alarms, error, synchronization*	*State changes, event-oriented signal sources (eg. switches)*

Table 2.1: Properties of Communication Paradigms

and push-type, publishers become active themselves without centralized manager triggered by a timer or an event. Interestingly, in order for the subscription of the subscribers to correct communication group or multicast group the client-server-type communication is used.

The properties of these three communication paradigms are summarized in Table 2.1 [TMV95].

3 Industrial Networks and Determinism

The technical selection of networks for a particular application revolves around evaluating and balancing quality of service (QoS) parameters. Two parameters which are evaluated to find the balance between network components competing for limited bandwidth and time to deliver information between end components are network average speed and determinism. Network speed is a function of network access time and bit transfer rate. On the other hand, determinism is a measure of the ability to communicate data consistently from end to end within a

guaranteed time. The MAC component of network protocols defines the mechanism for delegating network bandwidth for optimized communication (eg. large packets with low determinism vs small packets with high determinism).

The basic QoS measures of industrial networks incorporates the speed and bandwidth of a network (i.e. how much data can be transmitted in a time interval), the delay and jitter associated with data transmission (time for a message to reach its destination and repeatability of this time), and the reliability and security of the network infrastructure.

The bandwidth of an industrial network is the number of bits that can be transmitted per second. The Ethernet-based industrial networks support data rates of 100 Mb/s or 1 Gb/s. The speed is the inverse of the data rate, thus the time to transmit 1 bit of data over the network, T_{bit} = 10 ns for 100 Mb/s Ethernet. The transmission time for a message on the network can be computed from the network's data rate, the message size and the distance between two nodes. It is considered a deterministic time in a network system. The transmission time (T_{tx}) can be written as the sum of the frame time and the propagation time:

$$T_{tx} = T_{frame} + T_{prop}.$$

Where, T_{frame} is the time required to send the packet across the network, and T_{prop} is the time for a message to propagate between any two devices.

The typical transmission speed in a communication medium is 2 x 10^8 m/s which means the propagation time T_{prop} is negligible, for example, T_{prop} = 67.2 μs for 2500 m Ethernet. The frame time (T_{frame}) depends on the size (in bytes) of data/message (N_{data}), the overhead (N_{ovhd}), padding used to meet minimum frame size requirement (N_{pad}), and the bit time (T_{bit}). Some protocols need extra bytes based on the bit-stuffing mechanism they use (N_{stuff}). The frame time can be expressed as:

$$T_{frame} = [N_{data} + N_{ovhd} + N_{pad} + N_{stuff}] \times 8 \times T_{bit}.$$

A network's time delay is defined as the total time between the sampled or computed data being available at source node and it being received and decoded at the destination node. The jitter is the variability in the delay. Many techniques have been developed to handle constant time delays however large variability in time delays is difficult to compensate for. The total time delay (T_{delay}) depends on the

preprocessing time taken at the source node for data encapsulation and encoding (T_{pre}), waiting time of the node when network is busy (T_{wait}), transmission time to send data across the network (T_{tx}) and the postprocessing time of the received data at the destination node for data decoding and postprocessing (T_{post}). T_{wait} is a function of the MAC mechanism of the protocol and can be computed based on network traffic, how many nodes are there, the relative priority of these nodes and the messages they are sending, and how much data they send. T_{pre} and T_{post} depend on the device and can be major sources of delay and jitter in a network. In an equation form, the total time delay is:

$$T_{delay} = T_{pre} + T_{wait} + T_{tx} + T_{post}.$$

The reliability of data transmission medium in a network gets affected by electromagnetic interference resulting in data corruption. To increase the reliability handshaking mechanism can be used. Acknowledgment messages (ACK) are sent between the devices to confirm the data delivery. If no ACK is received, the data is resent. However, the handshaking techniques increases the required overhead and thus decreasing the overall effective bandwidth.

Security of networked systems is another concern as the networks and operating systems are vulnerable to Internet-based attacks and viruses. Most industrial fieldbuses were not designed to be highly secure and relied on the principle of "security by obscurity" instead of authentication or encryption techniques. The intent of incorporating security in the network is usually to prevent misuse of process data than counteracting network attacks. Firewalls are installed to prevent unknown traffic from entering the network and for secure encrypted transmission virtual private network (VPN) is used. In the recent years, there has been vast progress in development of intrusion detection/prevention systems (IDS/IPS) to handle increasingly complex attacks [DNVHC05, ZJS11, McM17, BG16].

4 Industrial Communication Protocols

The first implementation of the full ISO/OSI seven-layer stack was manufacturing automation protocol (MAP) developed as a framework for the comprehensive control of industrial processes covering all automation levels. However, its complexity made implementations costly and unjustifiable for general-purpose.

Learned from the failure of MAP, further automation protocols stack was reduced and layers were combined based on the domain requirements for simplicity [GH+13] (Figure 4.1). The Internet is governed by protocols based on fully functional TCP/IP stack consisting only physical, network, transport and application layers. The IEC 61158 fieldbus standard reduced model (EPA) consists of only three layers — physical, data-link and application. Many fieldbuses are single-segment networks with limited where routing functionality and end-to-end control is not necessary. Thus, network and transport layer are removed. Also, fieldbuses were not designed for sophisticated tasks hence session and presentation layers are also not needed. However, when the layer 3 and layer 4 functions are needed they can be placed either in layer 2 or layer 7. Furthermore, layer 7 always covers layer 5 and 6 functionality.

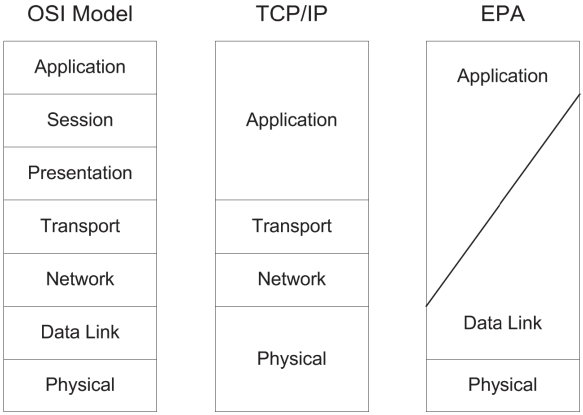

Figure 4.1: Reduced ISO/OSI stack comparison [GH+13].

As discussed earlier, Ethernet has penetrated the automation network. However, the office Ethernet didn't support deterministic capabilities hence couldn't be used for lower levels of automation pyramid. Collisions can occur on the network, and messages must be retransmitted after random amounts of time. To address this inherent nondeterminism, many different flavors of Ethernet were proposed for use in industrial automation. An effective solution in recent years has been the utilization of switches to manage the Ethernet bandwidth utilizing TDMA approach among time-critical nodes giving rise to switched Ethernet.

Switching technology does eliminate collisions, but delays inside the switches and lost packets under heavy load conditions are unavoidable also with switches. The hard real-time requirements of drive controls can't be made through these industrial Ethernet solutions. This led to development of Real Time Ethernet (RTE) standard IEC 61784.

For different application domains there are different RT performance requirements which require different implementations to achieve determinism. RTE implementations are all based on the TCP/IP model (Figure 4.2) and can be classified based on transmission time [Dec05] as follows:

- A low-speed class for human control with transmission time around 100 ms. This timing requirement is typical for the case of humans involved in the system observation, for engineering and for process monitoring. This requirement may be fulfilled with the use of Ethernet cabling and TCP and UDP for non-RT communications. This approach is called 'on top of IP' where the application layer is responsible for scheduling communication to meet the requirements. It is possible to communicate over network boundaries transparently. However, such communication introduce non-deterministic delays and the scheduling device must be equipped with adequate resources. The industrial protocols Modbus/TCP and EtherNet/IP are based on this RTE implementation.

- In the second class, for process control, the transmission time requirement is below 10 ms. This is a requirement for most tooling machine control system like PLCs. To reach this timing behavior, modification of the TCP/IP stack may be done only at the application level to use standard data packets and the transport level may be modified to use custom ethertypes for real-time communications. This approach is 'on top of Ethernet' where custom ethertypes are defined in the Ethernet frame alongside standard types such as IP. The network components and connected devices must have the knowledge of the custom protocols. Often the custom ethertypes will be given dedicated bandwidth or priority within the network. Ethernet Powerlink is the widely popular protocol within this approach-based protocols.

- The last and most demanding class is imposed by motion controls requiring a cycle time less than 1 ms with jitter not more than 1 μs. This can

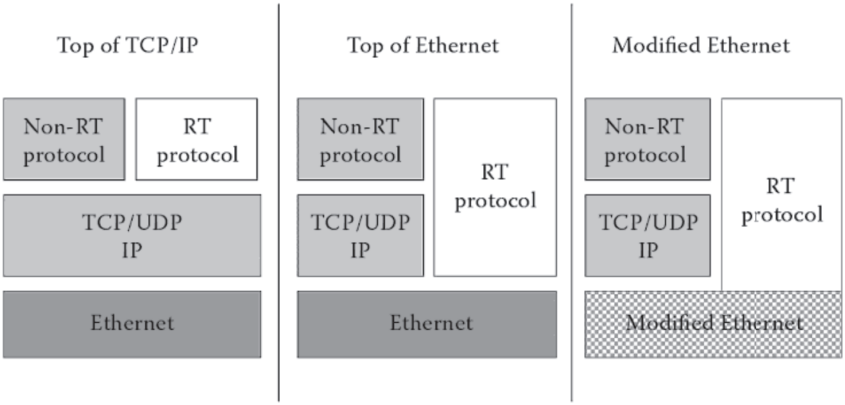

Figure 4.2: RTE implementations.

only be reached when the Ethernet data-link layer is may be modified to apply mechanisms and infrastructure that allow for real-time communication. This approach is called 'modified Ethernet' which enables non-standard topologies such as rings or buses to be implemented. To enable these topologies, the switching functionality is integrated inside the field device. The modifications are mandatory for all devices inside the RT segment but allow non-RTE traffic to be transmitted without modifications. Certain variants of PROFINET (Ethertype 0x8892), EtherCAT (Ethertype 0x88A4) and SERCOS (Ethertype 0x88CD) protocol types follow this approach to provide 1 ms transmission time requiring customized hardware.

5 Summary

This report outlined how industrial communication paradigms differ w.r.t. various communication fundamentals. The important aspect of Quality of Service (QoS) and related parameters were discussed in brief. At the end, we looked

at why and how OSI/ISO model is reduced for automation system requirements, and the approaches based on the TCP/IP model for Real Time Ethernet implementations.

Bibliography

[BG16] Anna L Buczak and Erhan Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2):1153–1176, 2016.

[Dec05] J-D Decotignie. Ethernet-based real-time and industrial communications. *Proceedings of the IEEE*, 93(6):1102–1117, 2005.

[DNVHC05] Dacfey Dzung, Martin Naedele, Thomas P Von Hoff, and Mario Crevatin. Security for industrial communication systems. *Proceedings of the IEEE*, 93(6):1152–1177, 2005.

[DZ83] John D Day and Hubert Zimmermann. The OSI reference model. *Proceedings of the IEEE*, 71(12):1334–1340, 1983.

[GH+13] Brendan Galloway, Gerhard P Hancke, et al. Introduction to industrial control networks. *IEEE Communications Surveys and Tutorials*, 15(2):860–880, 2013.

[McM17] David McMillen. Security attacks on industrial control systems. Technical report, Technical Report. IBM, 2017.

[Shi12] Aaron Shi. OSI model data flow. `http://aaronshi.blogspot.de/2012/11/data-link-layer-add-both-header-and.html`, 2012. [Online; accessed 01-March-2018].

[SSKD11] Thilo Sauter, Stefan Soucek, Wolfgang Kastner, and Dietmar Dietrich. The evolution of factory and building automation. *IEEE Industrial Electronics Magazine*, 5(3):35–48, 2011.

[Tho05] J-P Thomesse. Fieldbus technology in industrial automation. *Proceedings of the IEEE*, 93(6):1073–1101, 2005.

[TMV95] J-P Thomesse, Zoubir Mammeri, and L Vega. Time in distributed systems cooperation and communication models. In *Distributed Computing Systems, 1995., Proceedings of the Fifth IEEE Computer Society Workshop on Future Trends of*, pages 41–49. IEEE, 1995.

[Zim80] Hubert Zimmermann. OSI reference model – the ISO model of architecture for open systems interconnection. *IEEE Transactions on communications*, 28(4):425–432, 1980.

[ZJS11] Bonnie Zhu, Anthony Joseph, and Shankar Sastry. A taxonomy of cyber attacks on SCADA systems. In *Internet of things (iThings/CPSCom), 2011 international conference on and 4th international conference on cyber, physical and social computing*, pages 380–388. IEEE, 2011.

[Zur14] Richard Zurawski. *Industrial communication technology handbook*. CRC Press, 2014.