openSAFETY –
the key to your safety solution

openSAFETY:
the first open and bus-independent safety standard for all Industrial Ethernet solutions

Save time and expenses: how everyone benefits from openSAFETY

openSAFETY over SERCOS III, EtherNet/IP, Modbus-TCP, POWERLINK, and your fieldbus

www.open-safety.org
openSAFETY:
the first open and bus-independent safety standard for all Industrial Ethernet solutions

With openSAFETY, the EPSG has introduced the world’s first 100% open safety protocol. In fact, openSAFETY is open not only in terms of its legal basis, but also literally open in technical respects: given the protocol’s bus-independence, openSAFETY can be used with all fieldbuses, Industrial Ethernet solutions, or industry-specific communication solutions. The EPSG provides active support for implementing openSAFETY on top of any and all data transfer protocols, and also offers help e.g. for certification and conformance tests. The open source license of this TÜV-certified protocol stack ensures that the technology is a very secured investment for all users.

Highlights at a glance:
- one single, uniform standard for all leading fieldbuses
- maximum productivity due to efficient cross-communication
- reduced commissioning and maintenance time
- automatic safe parameterization
- perfectly suited to safe modular machine concepts
- sole 100% open safety solution
- fastest IEC 61508 SIL3 communication solution
- no risk involved in investment: TÜV certified conformance test
- perfectly suited to back-plane buses

The data transfer protocol that carries safety frame traffic has no bearing at all on the functionality of the safety protocol. This is called the Black Channel principle: all safety-oriented mechanisms are exclusively implemented on the application level, which enables total independence from the underlying transport layer. A brief look at the basic structure of data transfer protocols may serve to illustrate the principle.

Transport and Application Layers

The standardized OSI (“Open Systems Interconnection”) data communication model is the reference scheme for today’s most common non-proprietary data transfer protocols. This system is comprised of seven layers that enable the processing of data in a hierarchically structured way. Every layer represents a stage determined by protocols in which data transfer tasks are carried out according to specific rules. The two fundamental, low-level layers are the so-called Physical Layer and the Data Link Layer, which are also jointly referred to as physical layers. These layers define the physical interface to...
the transmission medium, and provide test functions to check whether an actual connection between a sending and a receiving device is established at all. Ethernet only specifies these two low-level physical layers. Also referred to as “transfer layers,” the third and fourth layer handle the timing and logical order of data transfer as well as data attribution to applications. Comprising all transport-oriented services, these four layers combined can be said to constitute no more than the traffic medium for application data, which, in the OSI model, is attributed to the upper layers. These high-level layers include the Session Layer and Presentation Layer. These two are often grouped with the Application Layer, since all programs and applications directly access all three of them. The Session Layer administers the organization and synchronization of data exchange between applications. E.g. if a connection is interrupted, services on this layer ensure that communication is resumed from the point of interruption once the connection is back in operation. Layer 6, the Presentation Layer, translates system-specific representations into a format that higher-level applications can digest, and ensures syntactically correct data exchange. Other tasks carried out in the Presentation Layer include data compression and data encryption. The top layer in the OSI model is the Application Layer. For this layer, no strict definition of a task or range of tasks is applicable. It provides various services to actual applications that operate outside the scope of this model. Inspired by the OSI model, the illustration below highlights that openSAFETY exclusively specifies the high-level, application-oriented layers of the protocol stack. The safety mechanisms implemented in this layer enable safety-oriented decoding and encoding of payload data pertaining to specific safety-sensitive applications. For the sake of simplicity, the blue area in the center of this illustration covers all the transport-oriented layers 2 through 6. The choice of transport medium, or, more precisely, of a specific data transfer protocol, is of marginal importance.
Save time and how everyone benefits from openSAFETY

“Competition is good for business” – this free market maxim is almost always true, since healthy competition stimulates continual product improvements regarding quality as well as cost-effectiveness. However, safety-oriented software and hardware development is a special case where this rule may not apply, because manufacturers in a market economy are faced with a high investment risk: there are considerable development costs, but there is only comparatively small sales potential. Hence, the automation sector has long been calling for a universal safety protocol, i.e. one that would give all manufacturers a solid economic base for all further development of safety-oriented products. The introduction of openSAFETY marks the first time a standard of this kind has become available that can be used license free by anyone.

Rising development costs for communication systems

A brief review of the history of bus-based automation may serve to illustrate how investment risk has been increasing in line with growing complexity. Ten years after the introduction of the first fieldbus, the market was full of different, competing systems. Infighting over the standardization of the technology ensued. Until the present day, industry experts know this phase as the “fieldbus war,” which only seemed to end with the introduction of the IEC61158 – a weak compromise because still, even this conclusion did not bring any overall system compatibility. Still, even this conclusion did not bring about overall system compatibility. It should be noted, though, that development costs for first generation buses were comparatively low, whereas their market potential was sizable. The entry of Ethernet technology into industrial data communications marked the next stage of development, one that was widely associated with the expectation that, finally, a universal standard would prevail. However, as different manufacturers chose strongly divergent approaches to enable real-time performance with this technology, what followed was the fieldbus war all over again. Amongst a multitude of systems, about half a dozen have been able to claim major portions of the market: Profinet, Modbus-TCP, EtherNet/IP, which all provide soft real-time performance, and the hard real-time systems POWERLINK, EtherCAT, and SERCOS III. There was a moderate increase of development costs for Industrial Ethernet solutions in comparison to conventional fieldbuses. At the same time, the sales potential could not be increased in the same proportion.

Development effort

- Integrated safety technology
- Industrial Ethernet
- Fieldbuses

Sales potential

These graphs visualize how investment risk increases as development becomes more and more complex.
Safety technology – a special case

Things look markedly different, however, for the development and certification effort required to make products designed for use in safety-sensitive areas, whenever these are to comply with the IEC 61508 standard covering the “functional safety of electrical, electronic, and programmable electronic safety-related systems.” In this area, there is a ten-fold increase of development costs over those for non-safe fieldbus technology. Moreover, manufacturers must demonstrate a wide range of experience relating to the interpretation of the standards, and must have specialist know-how in certain methods and procedures. With highly demanding requirements and high costs on the one hand, and, at the current time, a comparably small market for such products on the other, a “battle of the systems” would seriously impede the further development of bus-based safety technology. E.g. sensor makers would face immense efforts and high risk exposure if they were to develop their products in keeping with the safety standards of various different safety protocols.

The solution: openSAFETY

A tried and tested, non-proprietary system, openSAFETY resolves this situation in a way that benefits both manufacturers and users. Thanks to the Black Channel principle that makes the safety protocol suitable for use with all fieldbus and Industrial Ethernet technologies, safety technology manufacturers can focus on one universal system, and must only complete one safety development process to serve all standard fieldbuses. Both the effort involved and the investment risk are drastically reduced. openSAFETY benefits plant and machine operators in much the same way. While they are responsible for safety in their machinery, they usually have no say regarding the communication systems used in it. These are predetermined by the control systems the machine manufacturers have chosen to use. However, openSAFETY gives machinery operators a consistent safety solution for a heterogeneous control network in its entirety.

Benefits for plant operators

- a single, consistent safety standard for an entire line or plant
- for all control systems manufacturers
- perfectly suited to safe modular plant concepts
- minimal commissioning and retrofit time

Benefits for sensor manufacturers

- only one-time development required
- no extraordinary investment risk
- minimal time-to-market
- low costs due to open source
- interoperability guaranteed

openSAFETY constitutes a universal safety standard for an entire production line, irrespective of the control system manufacturer and fieldbus standard used in it. The bus-independent openSAFETY standard therefore reduces costs as well as commissioning time for production facilities as a whole.
Since safety-oriented product development is characterized by particularly challenging conditions, namely high costs and a somewhat small sales potential, finding a way to ensure long-term investment viability is an absolute imperative for system and product manufacturers within this market segment. Investment protection depends on various parameters: the availability and reliability of the hardware and software that is used, a safe legal basis for their use, and, of course, the scope and proper budgetability of development costs. openSAFETY is a safe choice in every respect for product manufacturers and end users. This solution gives them a ready-to-use safety stack with a proven track record of many years in actual applications. In contrast to dedicated, proprietary developments, there are no high development costs for this solution, no lengthy periods of time must be spent on it, and no demanding know-how requirements must be met. As it is already TÜV-certified and incorporated into the IEC 61784-3 standard, openSAFETY also minimizes development risks. Last not least, users also enjoy an extremely sound legal basis for long-term investment viability, since openSAFETY has been made available under the BSD open source license.

4x faster = 16x greater productivity with openSAFETY

Cross-traffic enhances safety: A network featuring cross-traffic capability (pictured on the left) enables direct node-to-node communication for safety devices, whereas, by contrast, communication in a network that does not feature cross-traffic (see right side) must always go through Master and Safety Master nodes. In the latter case, signal transfer times are quadrupled, and an emergency stop is delayed.
Simple is safer: how cross-traffic enhances safety in your machine

Since openSAFETY only specifies the application layer, the performance and reaction times of a safe network that implements this safety protocol depends on the data transfer protocol that is used. Data transfer protocols differ in terms of available bandwidth, the cycle times that can be achieved, and functional features, such as e.g. hot-plugging capability or cross-traffic for data communication on the safe network. Though the reaction time of a safety solution is also determined by the cycle periods in absolute terms, cross-traffic is one capability that will decisively enhance the performance of a safety-oriented system. An original feature of the Ethernet standard, cross-traffic denotes the ability of nodes on a network to communicate directly with each other, with no need for a detour via a Master. In safety-oriented networks, cross-traffic not only enables installations in which safety controllers need not be placed in the center: by allowing for straight and direct communication in hazardous situations, cross-traffic capability also benefits reaction times. Since the emergency stopping distance of an axis (see illustration above) increases with the square of the fault response time and negative acceleration, quadrupling the signal transfer time will result in a 16-fold extension of the emergency stopping distance.

Fault response time for safe operational stop: even minimal extensions of reaction time may fatally increase the emergency stopping distance.

**Benefits for machine manufacturers**
- free choice of safety sensors
- faster reaction times
- tighter safety distances
- higher productivity
- easy commissioning and diagnostics
- facilitation of Machinery Directive implementation
openSAFETY is basically notable for its data transfer definitions, for the high-level configuration services it provides, and especially for its encapsulation of data that is relevant to safety into an extremely flexible telegram format. Indeed, in all applications, openSAFETY uses a frame with a uniform format, no matter whether for payload data transfer, or for configuration or time synchronization purposes. As variable as it is economic, frame length is contingent on the amount of data to be transferred. The safety nodes on the network automatically recognize the content, i.e. frame types and lengths do not have to be configured.

Automatic safe parameter distribution
One special openSAFETY highlight is the automatic safe distribution of parameters: the protocol enables storing all configuration details for safety applications, such as e.g. light curtains, in the safety controller. If a device is exchanged, the safety controller automatically and safely loads the stored configuration onto the swapped application – i.e. users do not need to manually configure the new node when they replace a safety device.

No faults go undetected
openSAFETY uses checksum procedures to perpetually examine whether transferred data content is incomplete, and constantly monitors the data transfer rate. Due to extremely short cycle times, failures are detected almost without any delay. Since all data traffic irregularities will thus be recognized, even unsafe networks do not compromise safety functionality. The following paragraph points out which type of transmission errors may occur, and explains the mechanisms openSAFETY uses to identify or prevent these faults.

Causes of fault
A substantial portion of all data transfer errors results from incorrect data forwarding by gateways. E.g., data duplications may occur if a network is linked to other networks via two gateways, both of which transfer the same set of data. On the other hand, data packets may be lost if a gateway does not pass on data at all, or feeds it into the wrong network. If data packets can only be transferred as a sequence of partial packets because of their length, there is a risk that different transfer routes via various gateways result in mix-ups or erroneous insertions of spe-

<table>
<thead>
<tr>
<th>Faults</th>
<th>Time stamp</th>
<th>Time monitoring</th>
<th>Identifier</th>
<th>CRC Protection</th>
<th>Redundancy with cross-checks</th>
<th>Distinct frame structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duplication</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loss</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Insertion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Incorrect sequence</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delay</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distortion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mix-up of standard and Safety Frames</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The table lists all known transmission errors and openSAFETY's applicable fault recognition mechanisms.
Specific packet segments. The forwarding of data may also be delayed due to high load on a gateway. Another potential source of faulty data is electromagnetic interference, which may distort data, namely by “flipping” single bits or even by destroying entire information sections. Moreover, in networks where standard data as well as safety data are transferred, so-called “masquerades” can occur, i.e. standard data is taken to be safety data due to mix-ups and insertions. This may result in serious malfunctions.

**Fault identification and prevention**

One of openSAFETY’s most crucial mechanisms is the time stamp, which prevents data duplications, mix-ups, and delays. Every data packet is stamped with the current time when it is sent. This stamp enables the receiver to avoid double read-outs, and to determine the chronological sequence of different packets as well as any delays. openSAFETY does not depend on distributed clocks; a special procedure provides for reliable synchronization of all microcontroller clocks within the nodes. Time monitoring is employed in order to prevent faults caused by data loss or excessive delays, i.e. the nodes are continuously monitored for live operation and proper functioning. In addition, as they are prompted for reply, Consumers can tell that the data link remains established. openSAFETY implements this mechanism, which is called “Watchdog,” as a software-based function. The identifier precludes any mix-ups on the receiving end: openSAFETY frames feature a unique, 8-bit or 16-bit identification tag that encodes parts of the address field, the telegram type contained, and the frame type. The most reliable means to identify changes to the original content is the CRC procedure, which uses a key to generate a checksum for each data set, and attaches that as well as the key as a bit sequence to the data set. This checksum is a distinctive encoding of the data set itself. Using the bit sequence and the key, the receiver calculates the original data set, and checks the result against the data set that was received in the clear. If any deviations from the original data content are detected, the message will be ignored.

**Structure of an openSAFETY frame**

openSAFETY duplicates the frame to be transferred and conjoins the two identical frames into one openSAFETY frame. Hence, the openSAFETY frame consists of two subframes with identical content. Each subframe is provided with an individual checksum as a safeguard. The receiver compares the identical content of the two subframes. The probability that the same data is changed or destroyed in two such subframes is extremely low, and even lower the more the frame length increases. That said, even in this extremely exceptional case, the checksums still serve as a corrective. The special format of openSAFETY frames, i.e. the two subframes with their own individual checksums, also makes “masquerades” extremely unlikely to occur, and precludes any erroneous processing of a masked standard message.

**The openSAFETY network**

An openSAFETY network may contain up to 1023 safety domains, with up to 1023 nodes or devices permitted within each of these. Safety domains can extend over different and inhomogeneous networks, and can integrate the safety nodes that are scattered throughout these into one domain. Safe and unsafe devices can be operated within one domain. Gateways allow for intercommunication between different safety domains. openSAFETY enables users to enforce hierarchical separations as well as to establish separate safety zones on a network. Therefore, e.g. installations can be made in one zone, while production in other zones carries on unimpeded. In every domain, a Safety Configuration Manager (SCM) is responsible for continuous monitoring of all safety nodes.
openSAFETY
over SERCOS III

SERCOS III
An open and vendor-independent standard for digital drive interfaces, SERCOS III not only specifies the hardware architecture of the physical connections as well as a protocol structure, but also features extensive profile definitions. For SERCOS III, i.e. the third generation of the Sercos Interface that was originally introduced to the market in 1985, Standard Ethernet serves as the data transfer protocol. This communication system is predominantly used in Motion Control-based automation systems.

How it works
SERCOS III requires dedicated hardware on both the Master and the Slave side. Such SERCOS III hardware relieves the host CPU of all communication tasks, and ensures quick real-time data processing and hardware-based synchronization. The SERCOS user organization provides a SERCOS III IP core to support FPGA-based SERCOS III hardware development. SERCOS III uses a summation frame method. Daisy chain or closed ring cabling is required for the network nodes. Data is processed while passing through a device, using different types of telegrams for different communication types. Due to the full-duplex capability of the Ethernet connection, a daisy chain will actually yield a single ring already, whereas a proper ring topology will produce a double ring, allowing for redundant data transfer. Cross-traffic is enabled by the two communication interfaces every node is equipped with: in a daisy chain as well as a ring network, the real-time telegrams pass through every node on their way back and forth, i.e. they are processed twice per cycle. Hence, devices are capable of communicating with each other within a communication cycle, with no need to route their data through the Master. Besides the real-time channel, which uses time slots with reserved bandwidths to ensure collision-free data transfer, SERCOS III also provides for an optional non-real-time channel. Nodes are synchronized on the hardware level, with a cue taken straight from the first real-time telegram at the beginning of a communication cycle. The Master Synchronization Telegram (MST) is embedded into the first telegram for that purpose. Keeping synchronization offsets below 100 nanoseconds, a hardware-based procedure compensates for runtimes and runtime variations resulting from the Ethernet hardware. Various network

Specification
Layer model for openSAFETY over SERCOS III
segments may use different cycle clocks and still achieve fully synchronized operation.

**Organization**

A registered association, SERCOS International e.V., supports the technology’s ongoing development and ensures standards compliance. Over 50 control system makers and more than 30 drive manufacturers are members of the user organization.

**openSAFETY**

openSAFETY utilizes the Black Channel principle, i.e. it works as an implementation on top of an existing, unaltered SERCOS III solution. The cross-traffic feature provided by SERCOS III is used by openSAFETY for cyclical safe data exchange. SSDOs are transmitted on the non-real-time (NRT) channel. Ideal docking is ensured for openSAFETY via SERCOS III function profiles (FSp).

The openSAFETY over SERCOS III specification contains a complete description of all mechanisms and functions.

---

*Typical ring topology of a safe SERCOS III network*
openSAFETY over EtherNet/IP

**EtherNet/IP**
Initially released in 2000, EtherNet/IP is an open industrial standard developed by Allen-Bradley (Rockwell Automation) and the ODVA (Open DeviceNet Vendors Association). The "Ethernet Industrial Protocol" is essentially a port of the CIP application protocol (Common Industrial Protocol), which was already used by ControlNet and DeviceNet, to the Ethernet data transfer protocol. EtherNet/IP is particularly well established on the American market and is often used with Rockwell control systems.

**How it works**
EtherNet/IP runs on standard Ethernet hardware and uses both TCP/IP and UDP/IP for data transfer. Due to the Producer/Consumer functionality supported by the CIP protocol, EtherNet/IP has various communication mechanisms at its disposal, e.g. cyclic polling, time or event triggers, multicast or simple point-to-point connections. The CIP application protocol differentiates between "implicit" I/O messages and "explicit" query/reply telegrams for configuration and data acquisition. While explicit messages are embedded into TCP frames, real-time application data is sent via UDP owing to the latter protocol’s more compact format and smaller overhead. A VLAN flag in the header of the Ethernet frame is used to prioritize time-critical data. Forming the center of a star topology network, switches prevent collisions of data from devices that are hooked up via openSAFETY over EtherNet/IP

---

### Layer model for openSAFETY over EtherNet/IP

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical</td>
<td>EtherNet Physical Layer</td>
</tr>
<tr>
<td>Data Link</td>
<td>EtherNet CSMA/CD</td>
</tr>
<tr>
<td>Network</td>
<td>IP</td>
</tr>
<tr>
<td>Transport</td>
<td>TCP, UDP</td>
</tr>
<tr>
<td>Application</td>
<td>CIP Message Routing, Connection Management, CIP Data Management Services, Explicit Messages, I/O Messages</td>
</tr>
</tbody>
</table>

---

**Specification**
the center of a star topology network, switches prevent collisions of data from devices that are hooked up via openSAFETY over EtherNet/IP
point-to-point connections. EtherNet/IP typically achieves soft real-time performance with cycle times around 10 milliseconds. The CIPSync and CIPMotion protocol extension, an enhancement that is currently not yet available, and precise node synchronization via distributed clocks as specified in the IEEE 1588 standard are to deliver cycle times and jitter small enough to enable servo motor control.

**Organization**

Two organizations, the Open DeviceNet Vendors Association (ODVA) and ControlNet International, are jointly responsible for the maintenance and ongoing development of CIP technology.

**openSAFETY**

openSAFETY’s Black Channel functionality enables a simple implementation on top of EtherNet/IP. With EtherNet/IP, openSAFETY makes use of the option to establish connections via its own assemblies. Safe communication then proceeds via these assemblies. In this solution, safety devices operate as a Producer and a Consumer at the same time, and are therefore capable of listening to safe data on the network.

All details are documented in the openSAFETY over EtherNet/IP specification.
openSAFETY over Modbus-TCP

Modbus-TCP

Developed as early as 1979 by US PLC maker Gould Modicon (which is now a division of Schneider Electric), the Modbus protocol is considered one of the very first fieldbus systems. Enabling communication between control systems and devices supplied by different manufacturers, the open protocol eventually became something of an industry standard. Modbus is purely an application protocol, i.e. it does not depend on a transmission medium. Conceived and initiated by Schneider Automation, Modbus-TCP draws on the same services and the same object model as the original Modbus varieties, i.e. Modbus ASCII, Modbus RTU (asynchronous data transfer via RS-232 or RS-485), and Modbus Plus (Token Passing). However, this new incarnation uses Ethernet as its data transfer protocol, and uses TCP/IP packets for sending data.

How it works

Unlike Standard Ethernet, Modbus-TCP does not control node access to the network using a CSMA/CD procedure in the Data Link Layer, but handles access control through the Client/Server principle in the Application Layer. That means that a unique address is assigned to every node on the network, and that nodes are not allowed to send data unless a node’s request to do so is acknowledged by a Master with an explicit prompt to proceed. Parameters and data are encapsulated for sending, and are embedded into the payload data container of a TCP telegram. A “Modbus Application Header” (MBAP) is assigned to the payload data to ensure servers can definitely interpret Modbus parameters and instructions upon receipt. Only one Modbus application telegram may be embedded into each TCP/IP telegram. Like all TCP-based protocols (with TCP representing “Transmission Control Protocol”), Modbus-TCP operates based on connections. Prior to actual data transfer, a reliable connection must therefore be estab-
Established between Master and Slave in order to ensure that data is received completely and in the correct sequence. Once that connection is established, the Client and Server can transfer any amount of payload data. For cyclical input and output data transfer, the connection remains permanently in place. For service data, it is only established for the duration of the actual transmission. Server and Client nodes are able to establish and maintain several TCP/IP connections at the same time.

**Organization**

Based in the USA, the Modbus Organization, Inc. (Modbus-IDA), caters to the interests of users and manufacturers of Modbus-TCP-enabled devices.

**openSAFETY**

What is true for all busses is also true for Modbus-TCP: the Black Channel principle separates data transfer mechanisms from the safety layer, i.e. leaves the actual Modbus routines untouched. Ethernet-enabled Modbus is not limited to TCP/IP communication, but may also utilize UDP/IP. openSAFETY exploits this option. Non-cyclical safety data is transferred via TCP/IP frames. Cyclical data, on the other hand, which is duplicated due to openSAFETY’s safeguard mechanisms anyway, is delivered via UDP/IP.

The openSAFETY over Modbus-TCP specification covers all details.
openSAFETY

over POWERLINK

POWERLINK

Developed by B&R in 2001, the real-time Industrial Ethernet protocol POWERLINK is characterized by cycle times in the microsecond range, universal applicability, and maximum network configuration flexibility. A completely patent-free, vendor-independent, and purely software-based real-time system, which has also been available free of charge as an open source version since 2008, POWERLINK requires no proprietary hardware and provides total user independence from licenses and specific vendors. POWERLINK gives users completely integrated CANopen mechanisms on the one hand, and 100% compliance to the IEEE 802.3 Ethernet standard on the other. As a result, POWERLINK provides all Standard Ethernet features including cross-traffic, hot plugging, and a free choice of network topology.

How it works

POWERLINK uses a mixture of timeslot and polling procedures to achieve isochronous data transfer. In order to ensure coordination, a PLC or an industrial PC is designated to be the so-called Managing Node (MN). This manager enforces the cycle timing that serves to synchronize all devices, and controls cyclical data communication. All other devices operate as Controlled Nodes (CN). In the course of one clock cycle, the MN sends so-called “poll Requests” to one CN after another in a fixed sequence. Every CN replies immediately to this request with a “pollResponse,” which all other nodes can listen in on. A POWERLINK cycle consists of three periods: during the “Start Period,” the MN sends a “Start of Cycle Frame” (SoC) to all CNs to synchronize the devices. Jitter, i.e. clock rate inaccuracy due to fluctuations in the cycle, is as low as about 100 nanoseconds. Cyclic isochronous data exchange takes place during the second period (“Cyclic Period”). Multiplexing allows for optimized bandwidth use in this phase. The third period of a cycle marks the start of the asynchronous phase, which enables the transfer of large, non-time-critical data packets. Such data, e.g. user data, is spread out over the asynchronous phases of several cycles. POWERLINK distinguishes between real-time and non-real-time domains. Since data transfer in the asynchronous period supports standard IP frames, routers separate data safely and transparently from the real-time domains.

Specification

Layer model for openSAFETY over POWERLINK
Organization

An independent organization with a democratic charter, the Ethernet POWERLINK Standardization Group (EPSG) was founded by drive and automation industry leaders in 2003. Their common goal is to standardize and to continue to develop and enhance POWERLINK technology. The EPSG cooperates with leading standardization organizations, e.g., CAN in Automation (CiA), IEC, and the Open Source Automation Development lab (OSADL).

openSAFETY

openSAFETY is simply implemented on top of POWERLINK as well, with no impact at all on that base protocol. POWERLINK provides full-fledged cross-traffic. openSAFETY uses this function to achieve extremely brief safe reaction times. All communication for initializing and parameterizing the system goes through POWERLINK’s asynchronous communication channel.

All of openSAFETY’s mechanisms, functions, and potential options are described in the openSAFETY over POWERLINK specification.

Sample safe POWERLINK network: generally speaking, all conceivable topologies are possible!
openSAFETY over your fieldbus

Other than the various familiar, widely used fieldbuses and Industrial Ethernet systems on the market, or even the lesser known, but specified communication protocols, there are also individual, customized bus systems designed for "in-house" use only that many automation applications continue to employ. Found in all sorts of industries, unique implementations of this kind are, in many cases, neither standardized nor certified. Even for such environments, openSAFETY constitutes a suitable, uncomplicated safety solution. Due to true Black Channel operation, which is a native openSAFETY feature, the data transfer protocol never plays a role.

Since openSAFETY verifies the integrity of transferred data at all times, continually monitors the duration of transmissions using special mechanisms, and immediately recognizes any data transfer errors that occur, even single-channel, unsafe transport networks may be used as a basis for communication without compromising safety functionality at all.

What do users have to do?

Users who would like to implement an openSAFETY-based safety solution with their existing data communication system must only ensure that the safety protocol, which is available free of charge on the Internet, is integrated on the Application Layer of their bus system. Anyone in need of help can simply request assistance from the EPSG. In summary, given that openSAFETY is also already TÜV-certified, the basis for implementing safe data transfer capability is provided for free and virtually laid at any user’s doorstep.
Connecting Industries: openSAFETY provides one universal safety solution for all industrial sectors

Time-to-market as a basis for success!
FEEL SAFE ABOUT IT?

The global standard for integrated safety technology significantly reduces the wiring costs, enables faster commissioning, and achieves top machine performance through efficient communication. openSAFETY gives you maximum productivity with certified safety. Compatibility to your Industrial Ethernet solution guaranteed.

www.open-safety.org