
TECHNICAL ARTICLE: DISTRIBUTED PROCESSING AT SPEED: SAFETYNET P ETHERNET FIELDBUS

Compared with today's fieldbus systems, Ethernet in its current form is just too complex. This affects scan times, precision/frequency of measurements, data flow and processor power, among other things. As far as the automation system is concerned, the performance of the process computer as well as the communication must satisfy growing requirements. As a modern Ethernet-based fieldbus system, SafetyNET p matches these needs with ease. At the same time it is simple to install and delivers the operational reliability associated with the best of fieldbus systems.

SINCE THE INTRODUCTION of the SafetyBUS p safe fieldbus system in 1999, safety engineering through software-based communication has been taken forward into the world of Ethernet. SafetyNET p is the result of the consistent further development from its fieldbus origins.

While the idea of putting wide-ranging applications onto a traditional fieldbus is unattractive, the same proposition using Ethernet is reasonable. With safety-related systems, the reaction time required to control a potentially dangerous action has always been the decisive factor in driving development. With a data bandwidth some two orders of magnitude better conventional fieldbus, multiple functions carried over Ethernet become possible. SafetyNET p was always designed to operate as a publisher-subscriber network and therefore does not require a control centre. This enables modular system architecture and individually tailored subnetworks.

SafetyNET p - from here on referred to as SNp - is a multi master bus system. All the devices on the network have equal rights. The bus scan time of SNp can be adapted to suit the application requirements. A minimum bus scan time of 62.5µs can be achieved. As a result, it's even possible to use the protocol in a frequency converter control loop between a rotary encoder and a speed regulator.

Other highly dynamic applications are also possible, of course. Jobs and events can be recorded and executed with high precision across the entire network. This is a requirement for real time applications. A jitter of around 100ns can be achieved in real time control loops.

The protocol includes a safe data channel, which is certified for data transfer in accordance with SIL 3 of IEC 61508. Both safety-related and non safety-related data is transferred via the same bus cable. Non safetyrelated subscribers have direct access to safetyrelated data and can use this data for further non safety-related processing tasks. SNp is extremely flexible, not just when it comes to selecting a suitable bus scan time, but also when deciding on the appropriate topology: linear, star, tree and ring topologies are all supported.

The RTFL (**Real Time Frame Line**) communication principle is suitable for intra cell communication, as it allows the fastest scan times. RTFN (**Real Time Frame Network**) mode is used at higher levels, as it offers maximum coexistence capability with existing services. The interface with the application is made via widely used CANopen technology. Existing CANopen devices can be converted to SNp devices simply by changing the transport layer. Intelligent mechanisms that keep the device's processing power requirements to a minimum are used for communication within the SNp devices. This reduces the costs for simple field devices such as I/O modules, for example.

Systems that don't have real time requirements below the millisecond range can use conventional Ethernet controllers. SNp uses Ethernet technology. The interface depends on the required performance level: If fastest possible communication is required, the RTFL communication principle is used, which is based on Ethernet OSI Layer 2 (MAC Frames). For communication via mixed Ethernet based networks, from cell to cell or in general networks, TCP/IP or UDP/IP communication is used. The standard COTS Ethernet infrastructure can be used if the performance is satisfactory. This includes connectors, cables, routers, switches, gateways or communication channels.

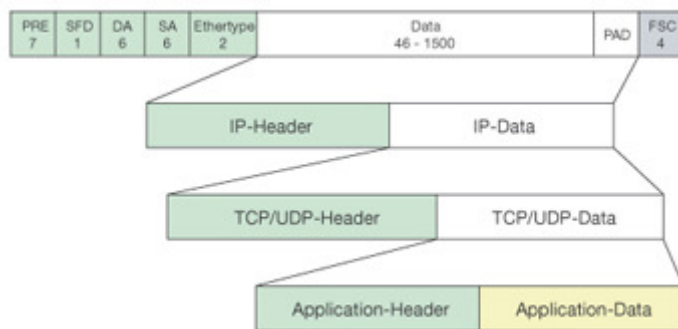
Ethernet as a fieldbus

Ethernet was originally developed to exchange large amounts of data between mainframe computers, without any specific requirements for real time behaviour. The Ethernet technology must therefore be

optimised to enable it to meet fieldbus requirements for deterministic communication, simple design and good overall performance.

An Ethernet data frame is designed for the transfer of large data packets. The minimum size of such a frame is 48 bytes. If smaller data amounts are sent, the unused bytes are still transmitted. This procedure is called padding. Based on an Ethernet frame with a 26-byte header, the 48-byte frame size would prove inefficient for a 16-bit I/O device as the two bytes of user data would contrast with an overhead of 72 bytes. Even at a bus load of 100%, the proportion of usable data per connection would never be higher than 3%.

The Ethernet frames in the transmitter and receiver are processed in a protocol stack. This software stack is typically processed in the device's microcontroller thus presenting a burden on scant processor resources. The processing time within the processor is even longer than the time it takes to transmit the data via the Ethernet cable. What's more, the faster the bus communication, the greater the required processor performance. It's clear, therefore, that the processor performance required for Ethernet communication is considerably greater than that required for the customary fieldbuses. If TCP/IP data frames are used to transport the data, all the devices will need IP addresses. Here too, the work involved in assigning addresses is significantly greater than that required for fieldbuses. Today's standard Ethernet networks usually feature a tree or star structure. One or more centralised switches are connected directly to the individual end devices. This means that each end device requires a switch port. If several switches are used, these are usually networked in star formation. This type of cabling is too complex and costly for use as a fieldbus system.



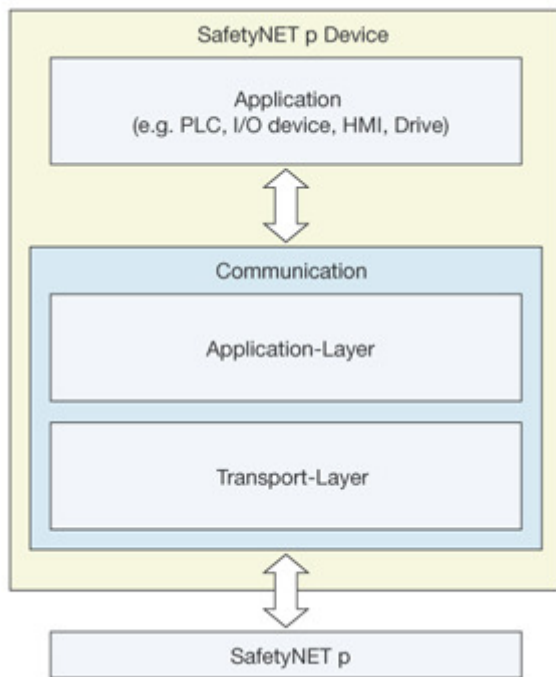
Standard Ethernet Frame with TCP/UDP IP usable data

To enable the Ethernet subscribers to communicate with each other, the switches on the star points of an Ethernet cabling structure act as a hub. They receive Ethernet frames at one port and, once the frame has been received in full, they pass it on to the relevant destination port. If the destination port is unable to receive data, the switch will buffer the frame temporarily. This process, plus the internal processing time within the switch, generally causes small but incalculable delays, which can be regarded as jitter. The above points needed to be resolved to enable Ethernet to be used as a fieldbus.

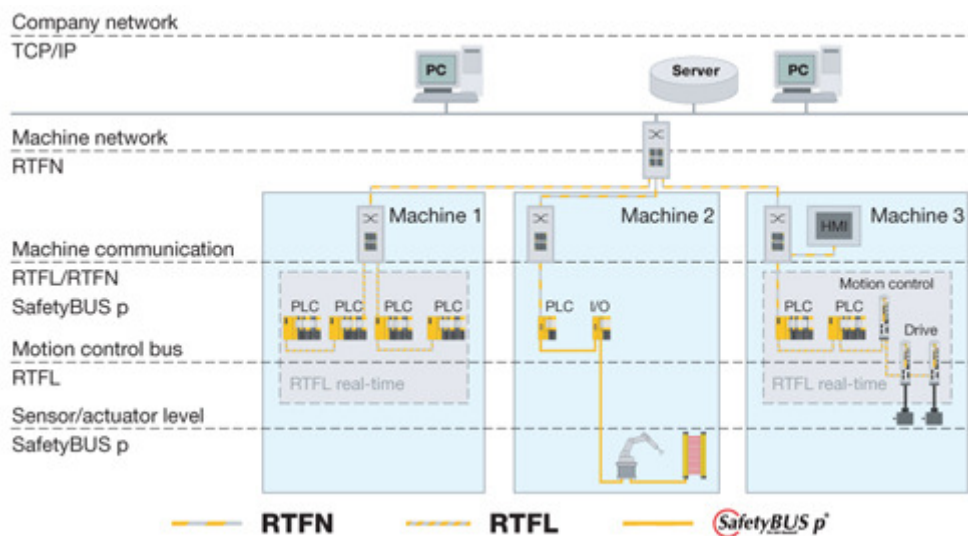
The SafetyNET p device model

Each SNp device has two principal tasks plus others depending on type. The first of these is the device's actual function, i.e. the operation of the controller, I/O module, operator terminal, drive or sensor. In quite general terms this function is referred to as the application for the purposes of this article. j2

The second central task is to transfer information from and to the application. This transfer occurs via SNp and is integrated within the communication system, which is subdivided into two parts. The interface between the application and communication system resides in OSI Layer 7, in the application layer. The application layer converts the data and functions that the application is intended to execute or communicate via SNp into generic object oriented formats and device-independent communication functions. These application layer functions apply generally and are converted into SNp-specific communication mechanisms within the transport layer. This approach allows the same application layer to be used for different transport mechanisms. In the case of SNp, this means that extremely fast intra cell communication via RTFL as well as inter cell communication via RTFN can both be implemented in the same application layer.



Snp device model. Application layer functions are converted into Snp-specific communication mechanisms within the transport layer. This approach allows the same application layer to be used for different transport mechanisms - RTFN and RTFL



Snp in automation. In the classic automation pyramid, different bus systems are assigned to different levels within the hierarchy. Depending on the hierarchy level, the bus system has various tasks to fulfil and must satisfy various requirements relating to bandwidth, reaction times and available services.

There is generally a safety bus right up to process control level which communicates safety related data. The total number of devices is generally highest at the sensor/actuator level, and lowest at the corporate network level. Thanks to segmentation into smaller units, a manageable number of devices can be combined at each level of the automation pyramid. In contrast, the size of each information unit from the sensor/actuator to the corporate network continually increases. While the standard information unit at sensor/actuator level is a single bit, at machine level a data width of one to two data words is generally used. At corporate network level, data sizes in Kbytes and higher are common.

Distributed real time applications

Today's fieldbus systems use simple data telegrams for synchronisation between the subscribers in the network. The sender forwards a telegram to individual or multiple recipients, who write the data to the outputs as soon as it is received. This method fails to consider the transmission time via the bus,

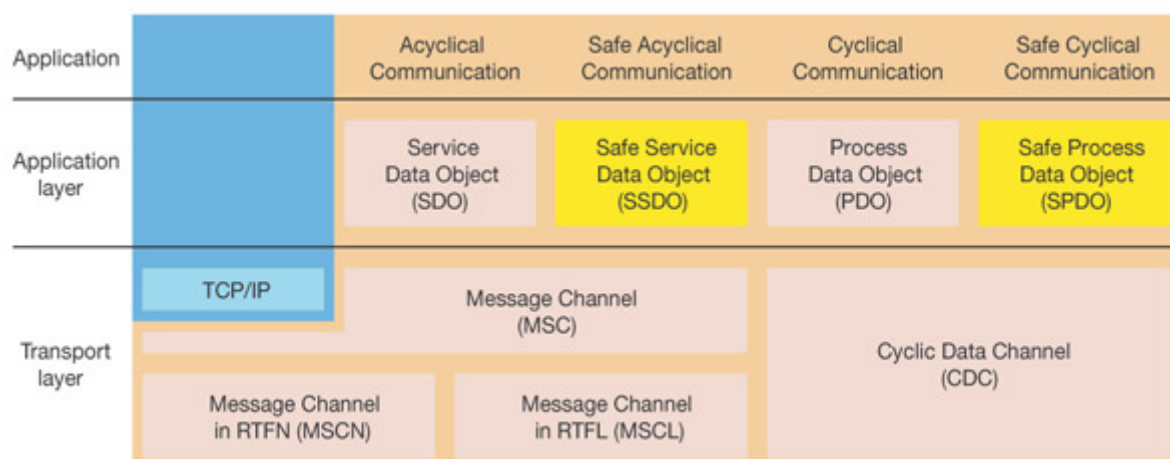
which may be dependent on interconnected routers, nor does it consider any potential collisions or waiting times on the transport medium itself. As a result, precise synchronisation between the individual bus subscribers is impossible.

Applications with fast process cycles, in particular real time applications distributed across several subscribers, require more precise mechanisms. These higher requirements can be achieved using events with network wide synchronisation. For this it is necessary to introduce mechanisms which will trigger actions within the network at the relevant point in time. To establish such methods network wide, each network subscriber needs an accurate clock which is synchronised with the other subscribers. The accuracy of the clocks and of the synchronisation determine the accuracy of the overall system. The jitter, the inaccuracy of the clocks in relation to each other, is used to gauge the accuracy. This jitter is measured in nanosecond time values.

The key to overall system precision lies partly in the accuracy of the clocks themselves. SNp uses a master clock to align continuously the clocks on the other subscribers. Based on precise, synchronous clocks, an action request can be sent to the subscribers in advance, which then perform the action at the required point in time. Communication is therefore separate from execution. This means it's possible to synchronise the way data is read in or output within the overall network.

All devices on SNp can synchronise their device clocks, if required to do so by the respective application. The Precision Time Protocol (PTP) standardised in IEEE 1588 can be used to synchronise the SNp devices in RTFN. The Precise Clock Synchronisation protocol (PCS), which is optimised for use in RTFL, is also used. Where networks use both RTFL and RTFN subnetworks, the RTFL master clock synchronises between the two methods.

The SNp communication model describes the communication system and its two components, the application layer and the transport layer. The application layer converts the existing device function on the application side into a generic functionality on the SNp side. This way it is possible to re use application programs across the range of devices. Emulation of the CANopen application layer represents another simplification, which is equally helpful to both manufacturers and users of SNp devices. Application data and application functions are written into application objects and are stored in an object directory. The object directory is accessible as an index table via SNp. Acyclical data is read and written using the SDO transfer protocol. Cyclical data is communicated via the PDO transfer protocol. The benefit of emulating the CANopen application layer is that device manufactures can port existing device profiles directly on SNp and re use them. The SDO and PDO transfer protocols are mapped to MC and CDC mechanisms on the SNp transport layer. As two different transport mechanisms are used on SNp (RTFL and RTFN), there are two transport layer implementations.



Communication channels. Communication requirements vary depending on the operating status of the overall network or individual devices. Phases for starting or stopping the device, device programming, diagnostics and parameter setting are primarily phases in which data transfer is not time critical. The relevant data packets are generally larger than during actual operation. In these cases, acyclical communication mechanisms are used. When the device is actually in operation, much of the data to be communicated is cyclical. In other words, it is transmitted at regular intervals. The scan time plays a key role in this.

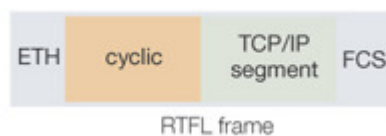
Parameters or diagnostic data need to be communicated at the same time. The required service will determine whether the data is transmitted cyclically or acyclically. During actual operation, non safety-related operating data is transmitted in addition to safety-related data. safety-related data is cyclical in nature, as is the non safety-related data.

RTFL communication

The Real Time Frame Line transport layer is optimised for the fastest real time applications. Typically the devices are networked in a linear structure, as with traditional fieldbus systems. All the bus subscribers have equal rights. Data is exchanged in accordance with the publisher/subscriber principle. As a publisher, each device can provide data to the other devices via Snp. It doesn't matter which subscriber or subscribers actually read the data. For their part, other subscribers can read the published data from individual subscribers or all subscribers. Even for subscribers it is irrelevant who has produced and published the data. This way it is possible to exchange data efficiently between all the subscribers.

The communication mechanism used by RTFL is a very fast cyclical data transfer in one single or more Ethernet data frames per cycle. The CDC, and MSC communication channels accessible from the application side are mapped to the RTFL cyclical data transfer by the transport layer. Basic communication is initiated by a special device called the Root Device (RD). The Ethernet frame generated within the Root Device is then transferred to the other devices (OD - Ordinary Device). The ODs fill the Ethernet frame with data to be published and extract from the Ethernet frame the data to be read. The devices are addressed via their MAC address. Each RTFL network requires just one Root Device.

TCP/IP communication via RTFL TCP/IP data can be transmitted in an RTFL line, without affecting the system's real time capability. Standard TCP /IP subscribers are connected to the first or last subscriber on an RTFL line. The TCP/IP data is segmented transmitted in the acyclical range of an RTFL telegram.



TCP/IP communication on RTFL

RTFN communication

The Real Time Frame Network transport layer can be used at process control and cell level, where standard Ethernet protocols are in demand and the requirements for real time are lower. RTFN is used to network the RTFL real time cells and to connect standard Ethernet subscribers, such as visualisation devices or service PCs.

The RTFN level typically has a tree topology, as used in office communication with conventional Ethernet. Switches are used to connect the network subscribers in individual point to point connections. RTFN can use two different mechanisms. In closed networks, the Ethernet MAC frame is used, as defined for Ethernet OSI Layer 2. The CDC is mapped to individual, unidirectional point to point connections. The devices are addressed directly via their MAC address. If a response is required from the addressee, this must establish its own connection. The UDP frame defined for Ethernet OSI Layer 4 is also used. The devices are addressed by their IP address. If IP-based communication is used, the RTFN frames may also be routed from network to network! With RTFN there is no distinction between RD and OD.

Publisher-subscriber relationship

Snp uses publisher/subscriber technology for the cyclical transfer of data. Each Snp subscriber publishes its own data and also has access to all the data published by the other subscribers. An off line configuration tool determines which subscriber publishes which data or subscribes to it. Data is also allocated to the PDOs via the configuration tool as part of the network configuration. The PDO configuration can also be adjusted during operation via SDO messages. The device description files (EDS file) typically describe which data is published by which device. However, this data can also be read interactively from the devices. Once the required communication data is known, the configuration

tool ensures optimum distribution of the data to individual frames on the CDC and MSC channels. The configuration tool can also be used to set up devices that wish to subscribe to data from other subscribers. An allocation table is used to determine which subscriber uses which published data. The configuration tool specifies the communication profile segment (CPS) for each individual device as a result of this allocation. The respective CPS for the individual devices can be loaded into the devices online via SNp. The devices can either store the data locally or load it from a configuration server during the system start up phase.

Application layers

The application layer represents the interface between the application and the communication system. All of a device's SNp functions are made accessible via the application layer. The application layer represents the interface between the user and the communication system. SNp uses the widespread CANopen application layer. From the user's point of view, SNp communication is handled in a way that's familiar from CANopen devices.

Basic IP telegram types

The Internet Protocol, IP for short, is named and used in conjunction with the TCP/IP protocol family. Its main function is to address data packets and transfer them into a connectionless packet oriented network (Routing). All stations and end devices have their own address in the network, the IP address. In accordance with IP Version 4, the IP address is 32 Bits long. It is divided into 4 bytes, separated by dots. For example: 127.0.0.1 Each byte may have a value from 0 to 255. IP V6 has a 128 bit address and was launched in order to expand the address space.

UDP The user datagram protocol (UDP) is a very reduced connectionless network protocol which belongs to the family of internet protocols. The task of UDP is to transfer data which is transmitted over the internet to right application on the target system. Therefore UDP uses so called ports. The port number of the receiving service is also transmitted. With a checksum UDP is also able to detect failures in the data transmission. However as a connectionless protocol UDP has no measures for telegram repetition. This must be performed by the upper layers.

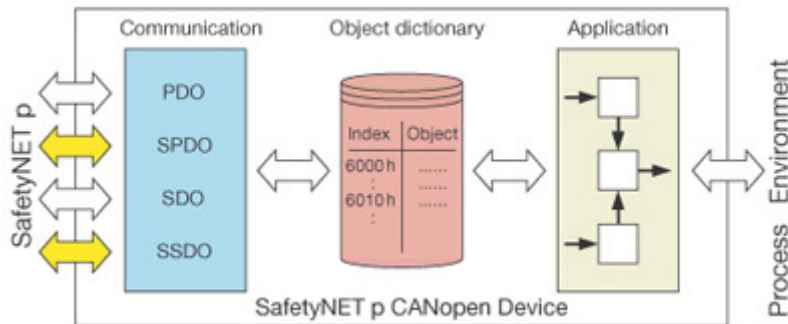
A modular machine design means that complex systems are decomposed into mechatronic units with separate functions. This reduces the complexity of the overall system. Machine manufacturers can standardise the mechanical and electrical engineering to a high degree. For example, a packaging machine consists of several function units: Feed conveyors, cartoners, filling unit, gluing device etc. If these function units are designed independently in terms of mechanics, hardware and software, they can be reused or exchanged on new machines or machine versions, without any major modifications. The individual function units can be commissioned separately before they are finally joined together. On SNp there is no central instance, as there is on master/slave systems. That is why SNp is the optimum communication system for distributed control systems and is ideally suited to the implementation of modular plant and machine concepts.

CANopen application layer

CANopen is an open fieldbus standard published by CIA (CAN in Automation). Combined with the CAN bus protocol it provides a standardised industrial application layer. This contains the standard for communication as well as the technical and functional performance data, which enables distributed automation units to be networked. Application objects are used, which are defined in device profiles. The core elements of CANopen are the device profiles, which define the uniform functions and standardised parameters and objects for the various applications and devices. With these profiles as a base it is possible to achieve a high degree of uniformity, even among devices from different manufacturers and device classes. Even today, corresponding profiles are available for all common device types found in automation. For example: Digital and analogue I/O modules, drives, valve terminals, control systems and rotary encoders, etc.

As the CANopen device profiles are not committed to a specific communications medium, they can be used for SNp. From the perspective of today's CANopen users, this means that they can fall back on familiar interfaces. Manufacturers too can largely adopt the CANopen implementation on existing devices when implementing SNp. The central task of the object directory is to act as a link between the application and the communication system. Essentially this is where objects are grouped and

standard communication and device parameters are stored. Safe PDOs are also stored in the object directory, but with an additional safeguard. Write access to safety-related objects is only permitted via SSDOs (Safe Service Data Objects). Read only access to safety-related data is permitted via both SDOs and SSDOs. A distinction is made between safety-related and non safety-related PDOs.



CANopen object directory

Safety communication

For the purposes of machine safety, the processing and communication of safety-related process signals requires special measures compared with conventional data transfer. From a certain level of safety onwards, safety-related signals are recorded and processed with multi channel redundancy in order to overcome safely any potential errors in the devices or periphery. Additional security mechanisms must be implemented within the protocol in order to overcome any communication errors. Communication errors that need to be overcome with safety-related communication are: Repetition Loss, Insertion, Incorrect sequence, Falsification, Delay, Mixing of safety-related and non safety-related signals.

SNp's safe application layer enables safety-related data to be transmitted over the same bus cable as non safety-related data. This safe communication was developed in accordance with relevant standards, such as EN IEC 61508. All the security mechanisms enabling safe communication are encapsulated within the protocol, so users and integrators do not have to worry as to whether communication is failsafe. With the exception of actual safe subscribers, all the other network components such as switches and cables are regarded as non safety-related - in accordance with the black channel principle.

For safe communication on SNp, SPDOs (Safe Process Data Objects) are used for cyclical communication, while SSDOs (Safe Service Data Objects) are used for acyclical communication. Safe [cyclical] process data objects have the following format:

- PID (Packet Identifier): Used with the SID for unique packet identification.
- Length: Complete length of packet in bytes
- Process data: Safe process data
- SID (Safe ID): Bit unique network wide ID, through which both the sender and the SPDO are uniquely identifiable.
- Counter Number: 8 Bit cyclical counter for life sign monitoring on the application layer
- CRC: Bit CRC of PID, Length, Process Data, SID and Counter No.



Safe cyclical process data objects

SSDOs (Safe Service Data Objects) are used for safe, acyclical data. For example, SSDOs are used to download the configuration to the safe SNp subscribers. Safe SDO communication uses the MSC data channel and provides the same communication options available with standard communication

via the RTFN/RTFL MSC. A 32-bit CRC is used for data security. If the data packets are too large to be secured using the 32-bit CRC, they can be downloaded in segmented form.

RTFL transport layer

The RTFL transport layer is an implementation for real-time communication for the highest speed requirements.

SNp's RTFL transport layer transmits data cyclically. Telegrams are passed along a line, from device to device. It is irrelevant whether the devices are actually connected in a line or are positioned in various network branches. What's important is that each device has precisely one predecessor and one successor within the network.

In the description which follows, this sequence is called the 'logical line'. Data transfer is initiated in the Root Device (**RD**) as part of each cycle. The RD creates an RTFL Ethernet Frame and sends this to the first Ordinary Device (**OD**). This OD writes the data it wants to publish into the Ethernet frame and sends this on to the next OD. The devices within the line are addressed via their MAC address. Each device knows the MAC address of both its predecessor and its successor in the logical line.

When the RTFL Ethernet frame has reached the final OD in the logical line and this OD has added the data it wishes to publish to the frame, the OD then takes the data it needs from the fully filled Ethernet frame. The OD then sends the data frame back along the same logical line, but in the reverse direction. All the other ODs can now take the data that's relevant to them from the Ethernet frame. In this way it's possible for each subscriber to exchange its status data with every other subscriber, as the Ethernet frame crosses the network.

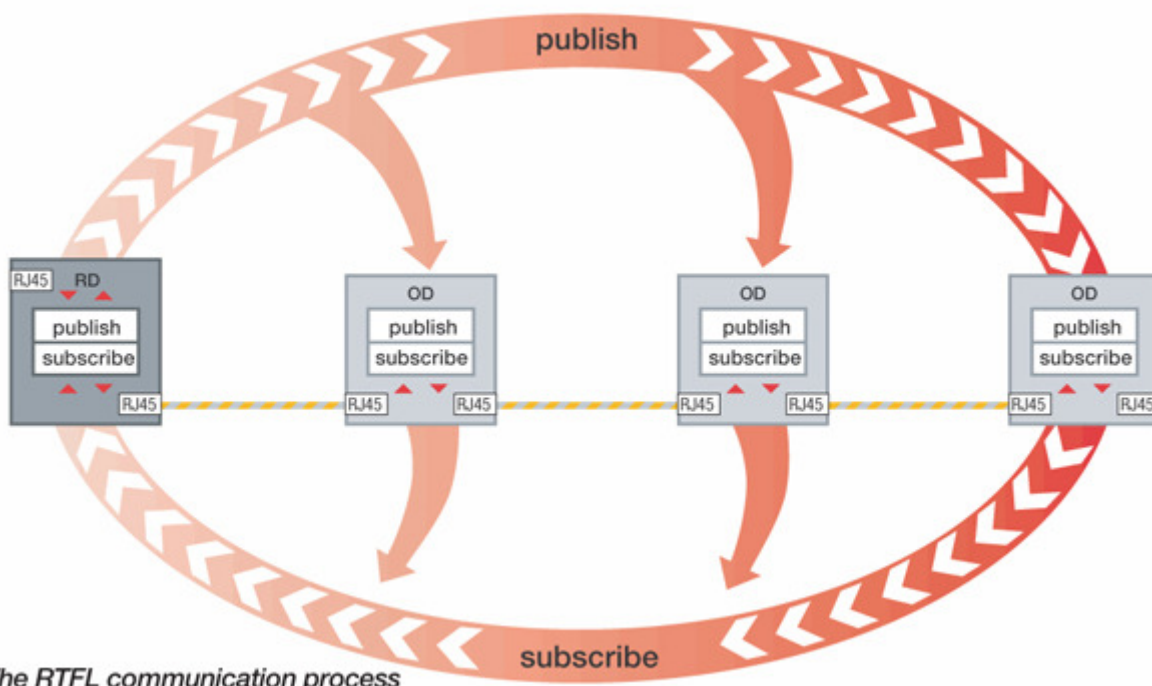
As each OD has at least two Ethernet interfaces, the linear structure is easy to achieve, without any additional network components. Equally, the Ethernet specification is met in full at each point in the network. If the ODs are connected in a physical line as described above, an additional special feature of SNp helps to accelerate the process. Telegrams received by a device at one port can be read or written as they pass through and are forwarded to the other port before they have been received in full. This mechanism is called 'Cut Through' mode.

In a network with a physical linear structure, the predecessor and successor of each device is unique. It is possible to produce physical topologies other than a logical line, but the latency times of the needed switches can increase the scan time as well as the jitter.

Device types in RTFL

The Root Device (RD) organises and controls the SNp data flow. It contains the following functionalities: Bus master (topology scan, clock synchronisation, configuration); Frame pump (to create the blank Ethernet frame).

The SNp configuration tool provides the RD with the configuration data. The RD reads the current network topology and configures the logical line. Each OD is provided with the MAC address of its neighbour. In its function as frame pump, the RD sends the RTFL frames as part of each cycle. An RTFL network must contain just one RD.



An Ordinary Device (OD) is a publisher and/or subscriber in an RTFL SNp network. Each OD has at least two Ethernet interfaces.

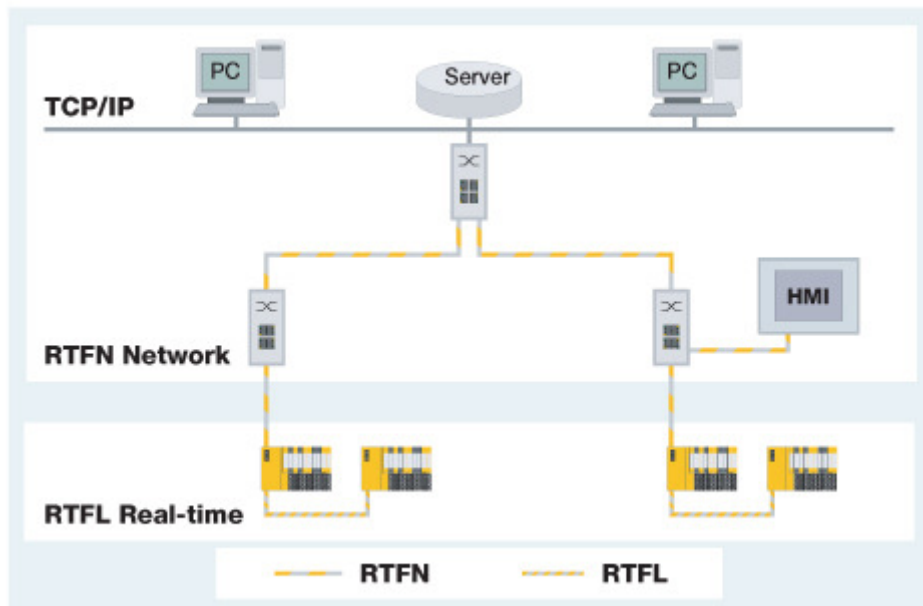
The Gateway acts as a link between RTFL and RTFN. It can also operate as an interface between SNp and a standard Ethernet. Both RDs and ODs can implement the function of a Gateway.

RTFL network management

The Root Device is responsible for network management in an RTFL network. The RD stores the network configuration data specified during engineering and distributes it to all the ODs when the network is started up. The RD also manages a list of all the subscribers in RTFL during runtime. This is called a mapping table and it contains the MAC address of each OD, along with its device ID, device address and other information that's required for network management.

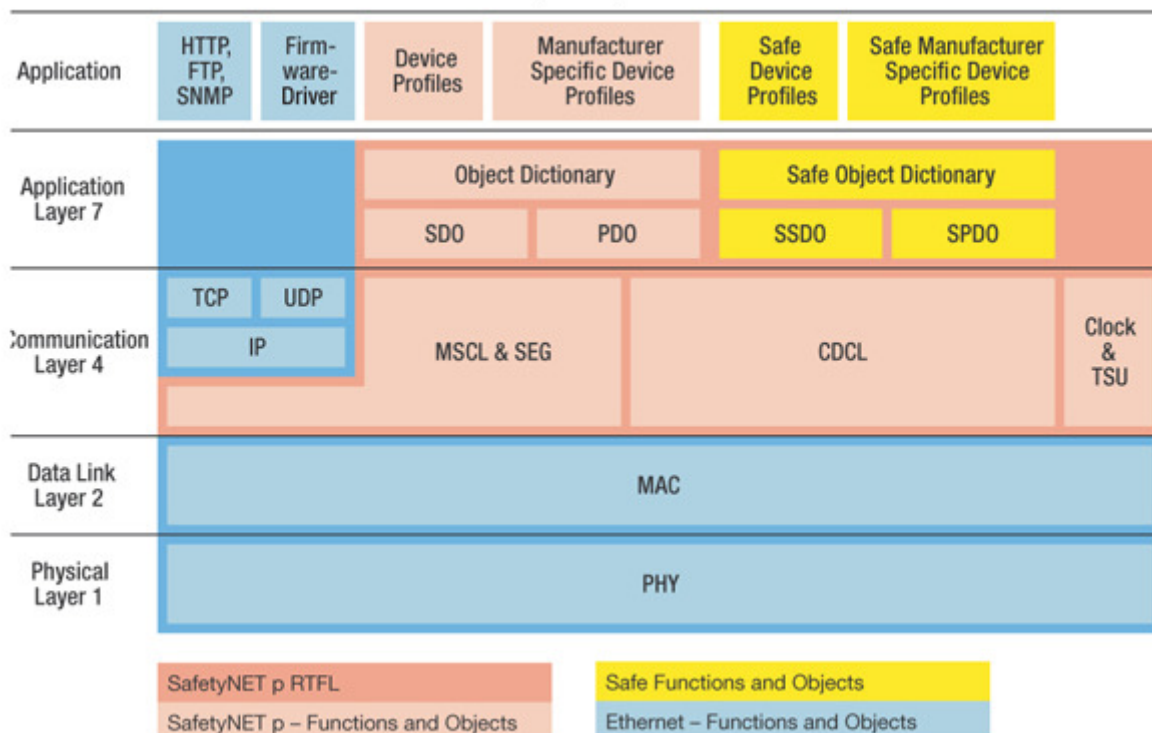
Root Devices can be assigned their IP address from an IP based network automatically by a DHCP Server or manually via an engineering tool.

There are two options for uniquely identifying an OD as a physical device: Through a unique physical position in some network topologies, particularly linear topologies, it's possible to identify the device uniquely via its physical position. The benefit of this option is that there is no need to make any settings or configurations on the device if a device is exchanged. The new subscriber merely has to be started up in the same physical position. Through a name stored in the device. Devices may be given a name. Subscribers with names are not recognised uniquely through their position but by their name. The name can be stored securely (in non volatile memory), on a chip card for example. This option is always required when the physical position cannot be clearly identified, for example on a tree structure.



Division of the networks. Real Time Frame Networks handle general Ethernet traffic while cyclical - and thus deterministic traffic - is carried on RTFL. Both network types support Safety-related systems.

SafetyNET p RTFL



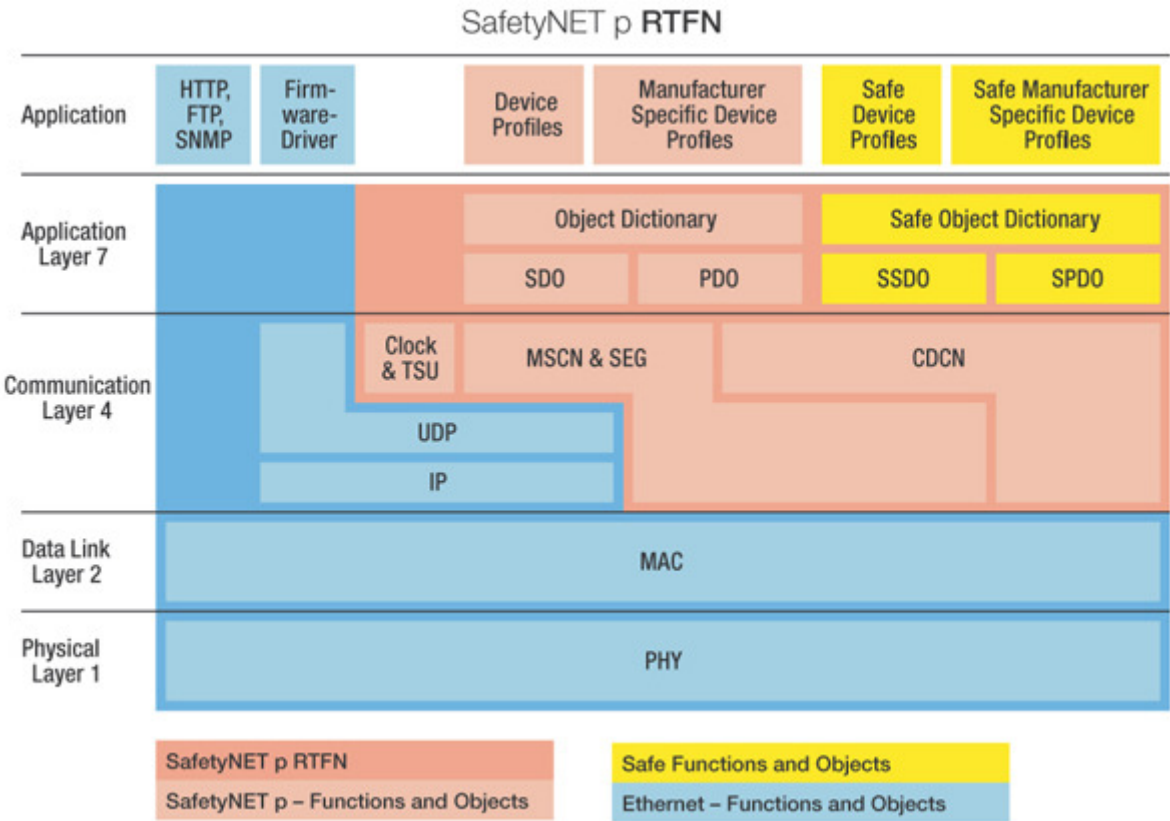
RTFL reference model. The diagram shows an overview of the RTFL protocol design. The individual functions of the application layer and transport layer are also assigned.

RTFN transport layer

SNp's RTFN transport layer is used at both cell and machine level. Because of the required compatibility with standard Ethernet, no special measures to control general Ethernet traffic are permitted, so conventional Ethernet ports and software protocol stacks are sufficient. This means that an RTFN subscriber can be implemented using a standard office PC with a standard Ethernet card.

Individual point to point connections can be established between the individual subscribers on a SNp network. These will be based either on MAC frames or UDP/IP, depending on the communication channel. MAC frames are used particularly for time critical data such as cyclical data. The required protocol stack contains little overhead, so this is the fastest means of communication in RTFN. In comparison with MAC frames, UDP/IP is less powerful, concerning its real time behaviour. For this reason UDP/IP is primarily used when the routing capability of the UDP/IP frame is required. TCP/IP offers the necessary services for data that is not time critical. The data security and automatic telegram repetition of the TCP protocol is also useful. Both UDP and TCP use the devices' IP addresses for addressing.

Communication via RTFN uses the following communication channels: Cyclic Data Channel (CDCL) for cyclical process data Message Channel (MSCN) for non cyclical process data All the remaining Ethernet services are also available for subscribers connected to the same network.

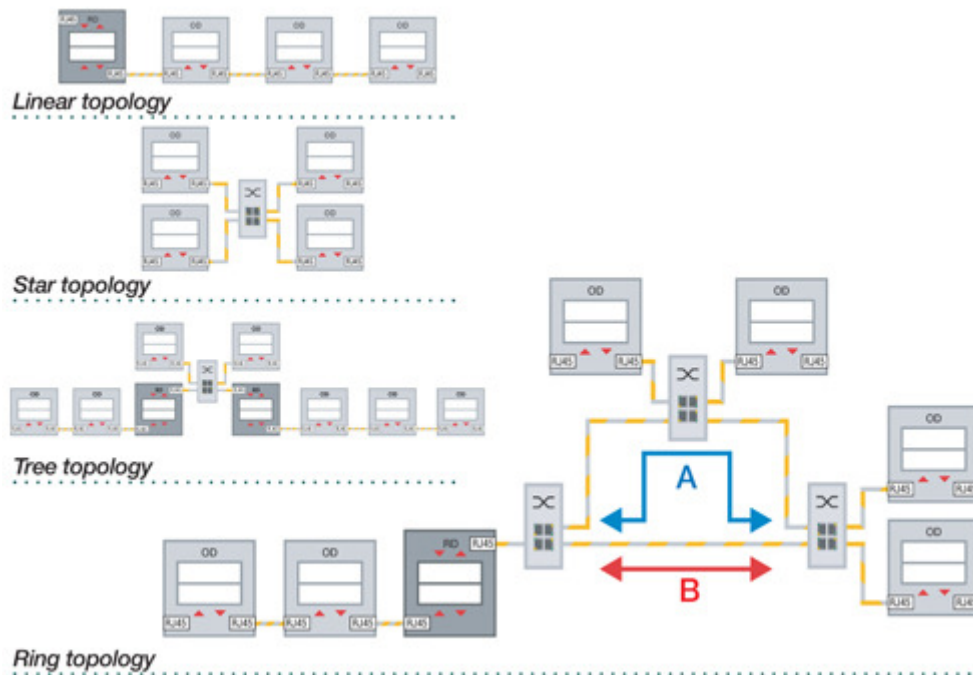


RTFN model. This transport layer is used at both cell and machine level. Because of the required compatibility with standard Ethernet, no special measures to control general Ethernet traffic are permitted, so conventional Ethernet ports and software protocol stacks are sufficient.

Topologies

The ability to implement various topologies with SNp means that the system is extremely flexible to adapt to the plant or machine layout. Linear, star, tree and ring topologies are possible with SNp. Dynamic structures are also supported.

A linear topology is created by connecting several bus subscribers in a line, without branches. This topology is the structure familiar from fieldbus systems. RTFL is optimised for linear topologies. Each RTFL subscriber has two Ethernet ports, making it simple to create a linear topology. There is no need for any additional network components, such as switches. This saves costs and installation work. Linear topologies can also be implemented using RTFN, but performance will be lower than with RTFL due to the addition of the switch latency times.



A star topology is formed when several subscribers are connected at a single point. On SNp, this connection is made via a switch. The communication partners communicate in point to point connections. RTFN is the preferred communication method for star topologies. However, RTFL star topologies can also be realised, using switches. In this case it's important to note that the scan time and jitter will be increased.

A tree topology is the combination of a star and linear topology. On SNp, tree topologies can be implemented by connecting several RTFL segments via RTFN.

A ring topology enables greater availability of a network. Various switches offer the necessary functionality. SNp subscribers can be connected in ring format using switches that support ring redundancy. The increased availability in the ring structure becomes clear if you disconnect the ring at a single point - due to a fault or simply by disconnecting a ring subscriber. The topology changes to linear and all subscribers are still accessible.

it is possible to add or remove subscribers during operation (hot plugging). This makes sense when changing tools or when using mobile visualisation devices. A flag is set during the engineering phase and this indicates whether or not the subscriber is present in the line, without an error being displayed.

Connectors for SNp

The ability to control the connection technology in an industrial environment is a key criterion in gaining acceptance for an industrial communication system among users. The Ethernet connectors commonly used in the office environment are suitable in this case. Essentially, three types of connector are specified for SNp. Others may be specified to suit the specific applications such as M12 and sealed RJ45 types.

Network cabling on SNp devices is based on the IAONA guideline: *Industrial Ethernet Planning and Installation Guide*. SNp uses Industrial Ethernet cables which conform to a minimum of CAT5. The cable cross section is AWG22/7. With this it's possible to achieve maximum permitted cable runs of 100m between to connected devices.

Security

The terms security and safety are often confused. The difference is that safety in the automation context normally describes a system's failsafe behaviour. However, the two terms can be

interdependent. An automation system that is insufficiently protected against malicious attacks cannot guarantee failsafe behaviour under certain circumstances.

Where the risk of attacks from hackers was previously limited to the LAN network, the automation is now also under threat. Some of the common security strategies from the office environment can be transferred to the factory, but additional hazards must also be considered and any measures adapted to the factory environment.

The security concept on Snp is a combination of security regulations and hardware/software solutions. In precise terms the concept segments the network into security cells. It's necessary to activate specifically the type of communication that is to be permitted from one cell to another. Snp devices have security data sheets, which clearly state the communication channels the devices need in order to fulfil their function. Special industrial firewalls enable security cells to be implemented in an industrial environment. Access to Snp devices is protected against unauthorised access through multi stage authentication. Internet, encryption mechanisms can be used.

Snp implementation

Simple, flexible and economical implementation options are the key to gaining acceptance of an industrial communication system among manufacturers of automation devices. These requirements have been considered on Snp: various options are available for the implementation of Snp devices. This way the manufacturer can reduce the work involved in interface implementation to a minimum. The application layer for RTFL and RTFN devices is identical. RTFL and RTFN implementations differ basically in the transport layer, where RTFL uses an FPGA chip.

RTFL communication provides two options for implementation: Implementation via preprogrammed FPGA Implementation or via IP cores in different FPGAs.

The application layer is implemented as software on a microcontroller. This may be the same microcontroller as the one containing the application itself. A dual channel architecture is required on safe devices. To implement the safe application layer, two microcontrollers must be available to process the safe protocol stack. These may be same microcontrollers as those on which the application for the automation device is implemented.

RTFN implementation requires no special hardware support; in other words, RTFN can be implemented in the form of a driver on a normal PC with a standard Ethernet card and TCP/IP stack.

Safety Network International is the independent, open user group that originated from the SafetyBUS p Club formed back in 1999. Since it was founded it has worked to promote the propagation and ongoing development of the two industrial communication systems SafetyBUS p and SafetyNET p

www.safety-network.de