plug and play – Sercos, the automation bus
Sercos – the automation bus: one of the leading bus systems for applications in industrial sectors for over 20 years. The real-time technology with its millionfold proven quality, the universal application possibilities and high security of investments make the Ethernet system the first choice in mechanical engineering and construction.

An efficient and deterministic communication protocol based on an optical transmission system not susceptible to interference is the foundation for Sercos’ success; today Sercos® is used successfully in the most varied applications in industrial sectors for over 20 years. The real-time technology with its millionfold proven quality, the universal application possibilities and high security of investments make the Ethernet system the first choice in mechanical engineering and construction.

This creates a basis upon which devices from different manufacturers can be combined without any problems. Today, over 4 million real-time nodes are being used on a daily basis in over 500,000 applications.

Sercos III – universal communication for distributed automation solutions

Industrial automation requires real-time-enabled and manufacturer-independent communication solutions. Different kinds of automation devices must be able to be easily and universally connected. The open, IEC consistent universal bus for Ethernet-based real-time communication, Sercos III, meets these demands today with a wide variety of benefits.

Advantages and benefits at a glance

Mechanical engineers and users benefit from a wide range of advantages and benefits:

**Proven**
- Sercos is an open, international standard (IEC 61784, IEC 61158, IEC 61800-7).
- Complete backwards compatibility ensures that it is a long-term investment.
- Leading suppliers of automation systems back Sercos with broad product portfolios.
- Sercos technology is widely accepted in many industries, in particular for high-end applications.
- More than four million real-time nodes are currently being used in more than 500,000 applications – and the number is growing every day.
- Ethernet-standard IEEE 802.3 physics and protocol are used.

**Simple**
- Sercos devices are easy to configure and put into use.
- The cables are easy to connect as neither the physical order of the devices nor the order of the connection to the two Sercos ports is important.
- Maintenance is easy because the devices and their position within the topology are recognized automatically.

**Fast**
- High speed due to the use of Fast Ethernet (1000Mbps full duplex).
- Short running times: the summation frame procedure, the on-the-fly processing and the direct cross communication reduces running times in the network to a minimum.
- Configurable cycle time: The communication cycle can be set between 31.25μs and 65ms – synchronization accuracy << 1μs.

**Efficient**
- Hot plugging can be done without damaging real-time or synchronization characteristics.
- Optimized use of bandwidth through a summation frame procedure and multiplexing procedure.
- All Ethernet-based protocols (including TCP/IP communication) can be transmitted with real-time data at the same time using the same cable.

**Reliable**
- Redundant data transmission ensures high machine and plant availability.
- Synchronization which is accurate to less than one micro-second ensures deterministic and synchronized communication across the whole Sercos network.
- Sercos allows fail-safe communication: Cable breaks are recognized within 25μs which means that data is lost for a maximum of one cycle.
- Robust cables made from copper or fiberglass.

**Economical**
- Sercos energy: save energy and maximize productivity at the same time.
- Machine controls can set components to idle mode.
- Fast and efficient data transmission allows for shorter cycle times and a higher output.

**Flexible**
- Flexible network topologies (ring, line, star/tree structures).
- Comprehensive choice of device profiles for all types of automation devices.
- Innovative communication functions, for example direct cross communication and ring redundancy.

**Safe**
- Safety functions up to SIL3 that are in accordance with IEC 61508 can be implemented with CIP Safety in Sercos.
- Safety-relevant and non-safety-relevant data is transferred over the same cable.
- Devices can securely communicate outside of network boundaries thanks to CIP Safety’s routing capability.

**Independent**
- Sercos technology is independent from manufacturers. The user organization Sercos International e.V. owns all rights to Sercos technology.
- Specifications are maintained and developed by a task force which covers all manufacturers.
- All specifications are freely available.
- You do not have to be a member to use Sercos technology.
Why Ethernet?

Ethernet makes it possible to have a single network infrastructure for communication across all levels of the automation pyramid. The vertical integration of anything from sensors to accounting software opens up new possibilities for operational control. At the same time, modern Ethernet-based networks allow for greater flexibility when installing and expanding control topologies within the production chain when compared with conventional field buses.

Ethernet technology provides a tangible range of benefits:

- recognized and future-proof technology
- many times higher data throughput than field buses
- no proprietary hardware required
- use standard readily available components such as double shielded CAT5e copper cables, connectors and controllers
- flexible and compatible automation systems thanks to international standards

Ethernet technology combines peripheral, driver, safety and office communication in one common medium – simple, economical and efficient.

Without particular specifications, Ethernet cannot satisfy real-time demands and efficiency requirements in automation engineering. It is for this reason that appropriate transmission procedures have to be defined which make the Ethernet efficient and compatible with real-time. Such diversity in real-time Ethernet solutions available on the market does not make it easy to keep perspective. Even specialists sometimes find it difficult to fully evaluate and compare how different real-time Ethernet solutions work and what effects they produce. In the end, an objective comparison of the solutions can only be made when actual application scenarios are available.

What is important is to understand the basic principles of different real-time Ethernet solutions and how they work.
Admittedly, collisions are avoided in switched Ethernet, but fluctuations in the time when transmitting information. In traditional CSMA/CD processes, they are, however, not deterministic. The lack of determinism is due to the fact that while information collisions are indeed flagged and avoided by processing protocols in hardware (image below, left). Real-time protocols in which conventional protocol stacks (layers 3+4, transfer and network layer) are used instead of protocol stacks (image below, center) are more efficient. Further increase in performance is possible by processing protocols in hardware (image below, right). Some real-time Ethernet protocols use their own Ethernet frame format at the same time and are therefore only compatible with Ethernet on the physical layer. On the basis of performance, all real-time Ethernet protocols shown in the image are commonly carried out in specific hardware, not only the version in the right figure.

**Protocol efficiency**

In automation engineering, many users that each have a low amount of data to transmit are typically connected to each other via control and sensor/actuator levels. If this process data is transmitted in individual telegrams, an unfavorable ratio between the Ethernet overhead and the user data volume will be created. In addition, should the user data be less than 46 bytes, telegrams are filled with zero bytes (called padding) to reach the minimum length of 64 bytes. In doing so, valuable bandwidth is not used. For this reason, summation frame telegrams, where real-time data from several users is consolidated into one common telegram, are more efficient.

**Lag time/cycle time**

A determinism transmission time (lag) preferably with a lower jitter is required (motion control applications << 1 µs) for real-time communication. Processing times can be reduced by removing network infrastructure components (switches, hubs) and by processing telegrams through network devices by the network users during the cycle. Cycle times of well under 1 ms can be achieved with more efficient protocols and a faster processing time for telegrams at individual network nodes.

**Synchronization processes**

Automation systems located in different places can be synchronized in different ways.

During a time slot process, synchronization can be deduced from the cyclical protocol. The synchronization process is based on the transmission of a synchronization signal which is cyclically received and analyzed by all network devices. In order to be able to ensure the best possible synchronization, the signal has to be sent and received within a strict time period with the lowest possible time discrepancy.

Another way of increasing the Ethernet’s temporal precision and synchronization is based on the principle of separate clocks (IEC 61588) which are synchronized with each other via telegrams. Separate clocks allow for a precise time base that is not affected by transmission time and time fluctuations in the communications medium. However, since the time base is not able to ensure any determinism when transmitting data, the data always has to be transmitted long enough in advance so that it is available to be processed when it is time for synchronization.

Separate clocks are also used in some real-time Ethernet protocols to minimize jitters during cyclical transmission.

**Topology**

Star topology, which is commonly used for Ethernet connections, is avoided wherever possible in automation engineering since it requires more wiring. Automation devices are better when directly connected, that is, without external infrastructural components instead (“Daisy Chaining”). This ensures that hubs or switches are integrated into terminal devices. In the case of extensive machines and plants, it can be advantageous to add a branch line or a transmission line to one or more devices, or to create a tree or star topology with external infrastructural devices or with a device integrated logic system that has additional ports.

### Comparison of real-time requirement categories

<table>
<thead>
<tr>
<th>Category</th>
<th>Cycle times</th>
<th>Synchronization required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Ethernet communication</td>
<td>Not cyclical</td>
<td>Not synchronized</td>
</tr>
<tr>
<td>Positioning drives, frequency converters, I/O peripheral</td>
<td>4 – 10 ms</td>
<td>&gt; 4 ms</td>
</tr>
<tr>
<td>Drives with peripherals signal processing, high-speed I/O peripheral</td>
<td>250 µs – 4 ms</td>
<td>&lt; 1 µs</td>
</tr>
<tr>
<td>Central drive concepts, highly dynamic metrology applications</td>
<td>31.25 µs – 125 µs</td>
<td>&lt; 1 µs</td>
</tr>
</tbody>
</table>

All standardized real-time Ethernet solutions are listed in IEC 61784 Part 2; each of the protocol specifications for real-time Ethernet solutions is contained in IEC 61158 series of standards. The different kinds of technology described there have Ethernet IEEE 802.3 in common as their transmission medium and protocol. This technology covers the whole plant and process automation sector in terms of how it can be applied. When it comes to their transmission medium and protocol, this technology is placed on bus systems which are used in motion control applications.

**Determinism**

Standard Ethernet is not real-time-enabled because it is not deterministic. The lack of determinism is due to the fact that while information collisions are indeed flagged in traditional CSMA/CD processes, they are, however, not avoided. This results in there being considerable fluctuations in the time when transmitting information. Admittedly, collisions are avoided in switched Ethernet due to full duplex transmission and point-to-point connections. However, additional lag times and non-deterministic delays during high-peak periods have been recorded when using switches.

**Real-time Ethernet: standard hardware vs. specific hardware**

Different processes can be used to make Ethernet compatible with real-time. The simplest form of real-time protocol is placed above the TCP/IP layer and is based on a polling mechanism, or time slot process (image below, left). Real-time protocols in which conventional protocol stacks (layers 3+4, transfer and network layer) are used in real-time protocol (image below, center) are more efficient. Further increase in performance is possible by processing protocols in hardware (image below, right). Some real-time Ethernet protocols use their own Ethernet frame format at the same time and are therefore only compatible with Ethernet on the physical layer. On the basis of performance, all real-time Ethernet solutions are integrated into terminal devices. In the case of extensive machines and plants, it can be advantageous to add a branch line or a transmission line to one or more devices, or to create a tree or star topology with external infrastructural devices or with a device integrated logic system that has additional ports.
Sercos mode of operation

In order to fully meet the demands of modern industrial automation, Sercos provides a high performance protocol. This protocol combines the openness of standard Ethernet with the need for real-time accuracy in automation engineering.

Transmission principle

Sercos communication is based on a time slot process with a cyclical transmission of telegrams that are based on a master slave principle. Cycle times are 31.25 µs, 62.5 µs, 125 µs, 250 µs as well as many times over 250 µs up to a maximum value of 65 ms. Besides automation concepts with centralized signal processing, decentralized automation solutions are also created as a result of this bandwidth during cycle times. In order to meet hard real-time demands despite using Ethernet, a communication cycle is divided into two time slots (channels). Sercos-defined real-time telegrams (Ethertype 0x88CD) are transmitted through a real-time channel that is free of a risk of collision. Parallel to this real-time channel, a UC channel can be configured, in which all other Ethernet telegrams (Ethertype <> 0x88CD) and IP-based protocols such as TCP/IP and UDP/IP can be transmitted. These time slots are called UC channels.

Cycle times and the division of the bandwidth or bus cycle in the real-time and UC channels can be adjusted for each application.

With Sercos, real-time data is sent to IEEE 802.3 in cyclical telegrams with Ethernet protocol type 0x88CD. These make M/S, DCC and SVC, SMP and Safety communication mechanisms available. The replaced data is addressable via standardized functional groups, classes and profiles.

Sercos differentiates between the following kinds of telegrams:

- Master Data Telegram (MDT): The master sends schedule data to the slave devices.
- Acknowledge Telegram (AT): The slaves send their status data to the master and to other slave devices.

The connected devices are recognized in the initialization phase (phase start-up with communication phases CP0 – CP4), they are addressed and configured for application. Each slave is assigned a device channel in the MDT and the AT, which the slave either uses to read from or to write into. Depending on the amount of data, several MDT and several AT are sent by the master per communication cycle. The telegram transports data from device to device. The relevant schedule data is read at each device or the required status data is written in.

Real-time channel

Sercos telegrams that are in the real-time channel are processed on the fly via individual network devices during the cycle. The telegrams are therefore only delayed by a few nanoseconds because the whole protocol process is carried out in hardware. In this way, network performance is independent of protocol stack, CPU performance or software implementation transmission times.

Real-time communication mechanisms available. The replaced data is addressable via standardized functional groups, classes and profiles.

- M/S (Master/Slave): Exchange of functional data between masters and slaves in an M/S connection.
- DCC (Direct Cross Communication): direct cross communication between devices in a DCC connection, either between control systems or between any periphery slaves (e.g. drive, IO, camera, gateway).
- SVC (Service Channel): Exchange of service data that is based on demand in a SVC channel as a component in real-time communication.
- SMP (Sercos Messaging Protocol): Transmitting functional data from several devices in one time slot by using a multiplex process configured in an M/S or DCC connection.
- Safety: Exchange of safety-related data in a M/S or DCC connection, e.g. disabling or approval signals or other set values.
- Sync: Cyclic synchronization trigger for precise, network-wide synchronization

Application

With additional network infrastructure components (such as hubs or switches) are avoided, transmission times are reduced to a minimum throughout the network.

Subsequent communication mechanisms are available in the real-time channel:

- M/S (Master/Slave): Exchange of functional data between masters and slaves in an M/S connection.
- DCC (Direct Cross Communication): direct cross communication between devices in a DCC connection, either between control systems or between any periphery slaves (e.g. drive, IO, camera, gateway).
- SVC (Service Channel): Exchange of service data that is based on demand in a SVC channel as a component in real-time communication.
- SMP (Sercos Messaging Protocol): Transmitting functional data from several devices in one time slot by using a multiplex process configured in an M/S or DCC connection.
- Safety: Exchange of safety-related data in a M/S or DCC connection, e.g. disabling or approval signals or other set values.
- Sync: Cyclic synchronization trigger for precise, network-wide synchronization

"Years of positive experience using Sercos control systems and drives led us to install Sercos as a system bus for our machines and plants. Thanks to Sercos’ real-time performance for motion, safety, vision and I/O as well as the possibility of integrating TCP/IP services easily, cabling and engineering was made considerably easier."

Years of positive experience using Sercos control systems and drives led us to install Sercos as a system bus for our machines and plants. Thanks to Sercos’ real-time performance for motion, safety, vision and I/O as well as the possibility of integrating TCP/IP services easily, cabling and engineering was made considerably easier.

Communication mechanisms in the real-time and UC Channel.

<table>
<thead>
<tr>
<th>Application level</th>
<th>Customer solution</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application</strong></td>
<td><strong>Customer solution</strong></td>
</tr>
<tr>
<td>Encoder profile</td>
<td>Safe Drive profile</td>
</tr>
<tr>
<td>Safe Drive profile</td>
<td>Safe Drive profile</td>
</tr>
<tr>
<td>Safe Drive profile</td>
<td>Safe Drive profile</td>
</tr>
<tr>
<td>Ethernet application</td>
<td>S/IP</td>
</tr>
<tr>
<td>IP</td>
<td>UDP/TCP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Communication level</th>
<th>Serco III communication controller</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware level</strong></td>
<td><strong>Communications</strong></td>
</tr>
<tr>
<td>Serco III communication controller</td>
<td>duplex Fast Ethernet (100 MBit/sec)</td>
</tr>
</tbody>
</table>

MDT = Master Data Telegram
AT = Acknowledge Telegram
UCC = Unified Communication Channel
RTC = Real-Time Channel

Configuration of the Sercos communication cycle

- MDT = Master Data Telegram
- AT = Acknowledge Telegram
- RTC = Real-Time Channel
- UCC = Unified Communication Channel

Translation of telegrams into the Sercos application

Real-time channel

- **MDT** (Master Data Telegram): The master sends schedule data to the slave devices.
- **AT** (Acknowledge Telegram): The slaves send status data to the master.

Real-time channel

- **MDT** and **AT** are sent by the master per communication cycle. The telegrams transport data from device to device. The relevant schedule data is read at each device or the required status data is written in.

Sercos telegrams that are in the real-time channel are processed on the fly via individual network devices during the cycle. The telegrams are therefore only delayed by a few nanoseconds because the whole protocol process is carried out in hardware. In this way, network performance is independent of protocol stack, CPU performance or software implementation transmission times.

Real-time communication mechanisms available. The replaced data is addressable via standardized functional groups, classes and profiles.

- **M/S** (Master/Slave): Exchange of functional data between masters and slaves in an M/S connection.
- **DCC** (Direct Cross Communication): direct cross communication between devices in a DCC connection, either between control systems or between any periphery slaves (e.g. drive, IO, camera, gateway).
- **SVC** (Service Channel): Exchange of service data that is based on demand in a SVC channel as a component in real-time communication.
- **SMP** (Sercos Messaging Protocol): Transmitting functional data from several devices in one time slot by using a multiplex process configured in an M/S or DCC connection.
- **Safety**: Exchange of safety-related data in a M/S or DCC connection, e.g. disabling or approval signals or other set values.
- **Sync**: Cyclic synchronization trigger for precise, network-wide synchronization
UC Channel
Conventional Ethernet communication is synchronized in the so-called Unified Communication Channel (UCC) in the Sercos network, e.g. for e-mail, web services or other proprietary and standardized Ethernet-based protocols. This channel sets itself down onto the Ethernet-layer without tunneling and has a range of benefits:
- It is possible to test and configure slaves even without initializing real-time network and master hardware
- Connection to other automation devices which support another Ethernet-based protocol which is not Sercos
- Directly addressing Sercos devices via MAC or an IP address
- Standard Ethernet devices such as laptops are connected directly to Sercos devices via any free Sercos port
- Full processing power for the application because the master does not have to tunnel or fragment Ethernet packets

S/IP protocol
The S/IP protocol allows data to be exchanged in one cycle using any Sercos devices without requiring a Sercos master or constant Sercos communication. The cycle using any Sercos devices without requiring a full processing power for the application because the master does not have to tunnel or fragment Ethernet packets.

Protocol structure
Sercos telegrams contain a Sercos header and a data field embedded in the Ethernet frame (see image below).

The Sercos header describes which phase the network is in and the position of the MDT and AT telegrams in the communication cycle. The MDT and AT data fields consist of three areas:
- Hot plug field: exchanges data with slaves that have been linked to the network while the operation is running.
- Service channel field: total number of communication channels that exchange acyclic data between master and slaves.
- Real-time data field: is used to create acyclic, cyclic or clock-synchronous connections, and so also real-time communication between any devices in the Sercos network.

Topology
Sercos networks consist of a master which coordinates and at least one slave which carries out automation functions. Generally, the devices are ordered simply and neatly in line or ring topologies. For this purpose, each Sercos device has two communication connections which are connected to the previous and subsequent device via an Ethernet cable – either using a twisted two-wire line (twisted pair) or an optical fiber. As a result of full duplex-enabled Ethernet technology, a logical ring appears in a line topology and a logical double ring appears in a ring topology (see image).

![Sercos-based topologies: Line and ring](image)

"Sercos has a clear and robust data structure. This increases operational reliability and simplifies application development. The network status is always clear and entirely transparent. I can make diagnoses of my plant on each network node with current Ethernet diagnosis tools."

In a line topology, the master is positioned at the beginning of a line or between two lines. The telegrams containing data run through the slaves and are 'looped back' by the last device. All devices analyze the data running in both directions so that all data is guaranteed to reach each device within the cycle independent of the order they are placed in (see direct cross communication). In this way, all devices can be integrated into the network, even across long distances, e.g. assembly lines, without high installation costs.

By adding an additional cable, the Sercos network closes to form a ring; a line is added between the last slave and the master, or two lines between the last slaves. The master feeds the ring in opposite directions over both ports so that the data is also analyzed in two directions in the ring. Besides having all the benefits of a line topology, ring topology provides additional redundant cabling. This means that a break in the ring...
can be compensated for without losing synchronization or causing a breakdown in communication (see ring redundancy).

Single devices or machine modules can also be connected to a line or ring via a branch line or a transmission line. To achieve this, either an infrastructural component with the appropriate branch line ports is integrated into the network or this function is directly integrated into the Sercos device.

Sercos also supports hierarchical, cascading network structures. At the same time, individual network segments are connected to each other via Sercos allowing network structures to be created which are connected in real-time and that are completely synchronized. Cycle times in single segments can be different, e.g. 250 μs for connecting drives and fast I/Os, and 2 ms for connecting control systems. Devices across the entire network can communicate with each other in real-time. Moreover, all devices across the entire network are guaranteed to be synchronized.

Hard and soft master

When it comes to masters, either specific hardware (or hard master) or an alternative standard Ethernet controller (or soft master) can be installed. In soft masters, Sercos-specific hardware functions are relocated to the hardware-related and real-time capable part of the master driver, so that the master-related activation can be realized completely in software. This form of master realization is interesting for PC-based controlling platforms for example.

Wiring

Installing a Sercos network is very easy and does not require any infrastructural components such as switches or hubs. All devices are connected directly to each other via patch or crossover cables. Fast Ethernet technology means that a 100m wire can be strung between two devices. The Ethernet ports on the devices are interchangeable and can even be used to connect standard Ethernet devices (e.g. notebooks) to a Sercos real-time network. This means that every Ethernet and IP protocol on Sercos devices can be accessed without interfering the Sercos network real-time process (online mode), and without needing the Sercos protocol to be activated (offline mode).

Field bus integration

Automation devices which (still) do not have a Sercos interface can be integrated into the Sercos network by using the appropriate gateways. Communication gateways are available for the coupling of field buses for example (e.g. Profibus, CAN), encoder interfaces (e.g. SSI) or sensor/actuator bus (AS-i, IO-Link). Moreover, gateways with axis controller periphery are available to integrate analogue axes. Gateways are either a feature of Sercos devices (e.g. modular I/Os) or are connected to the Sercos network as separate devices.

"With Sercos, we can adjust the network very easily to our current machines and plant structure. The fact that switches and hubs are not required means that installation costs are reduced and cabling is made considerably easier."
Cross communication – Sercos allows for decentralized intelligence and an unlimited capacity to work in real-time because direct communication between all devices is possible. Indirect communication between slaves by bypassing the master would compromise synchronous movements, e.g. by corrupting gantry axes or slowing down reaction times, e.g. when transmitting a trigger signal quickly. Sercos slaves can communicate via cross communication directly and with a minimum of communication dead times – this ensures unlimited real-time communication and intelligent automation structures. Control systems use the same principle to directly communicate with each other.

How it works:
Given that real-time telegrams are always processed in two directions independent of the kind of topology, a direct exchange of data between any given slave is enabled within the communication cycle (see image). This has the advantage that data can always be transmitted between the slaves within an individual communication cycle while causing minimal delay, even with increased cycle times. Moreover, all real-time data is available synchronized – that is, in relation to the general communication cycle – at every point in the network. As a result, data can be processed extremely easily, efficiently and flexibly at individual network nodes. Network diagnosis and monitoring can also be carried out as a result.

Synchronization
Sercos defines how the actual value is recorded and how valid the set value is for processes from different products and manufacturers that are to be synchronized. Each device receives a data telegram with a particular transmission time delay that is appropriate to the device. Unlike other Ethernet bus systems, Sercos derives a general execution time directly from the bus. The device calibrates itself independently with the bus cycle by using the arrival of the MST while taking into account the transmission time during each bus cycle to generate and adjust the internal synchronizing mechanism. As a result, separate clocks are synchronized exactly without exchanging time data that overloads the bandwidth.

The Sercos master simply has to detect the line or ring transmission time and transmit data to each device together with a sum value that has been appropriately configured. Variations in the sum value can delay the stable synchronization signal. Hence delaying the synchroniza-
The problem of breaking or reconnecting the sum or the sum of remaining devices can be formulated in the same way, with the addition that the number of devices is not really specified. This can be carried out at any time, even during low-peak cycle times. This increases the stability of the process in machine connection with fast reac-
tion times and great accuracy.

Oversampling and time stamping
The oversampling process which is integrated into Sercos protocol allows more than one nominal/actual value per cycle to be transmitted. This increases the deli-
crate nature of the process control in extremely critical laser applications, for example, because more data can be consolidated and emitted.

Time stamping also opens up new opportunities for communication across the fixed cycle. This function transmits event-controlled results such as specific measured data and switches outputs independent from the cycle. This increases the stability of the process in complex processing solutions, like those in the semicon-
ductor industry.

Performance
Custom Ethernet protocols from office technology rely on user data being sent as individual packets to each device – embedded in a defined framework made from protocol overheads. The quantity of overheads in data traffic is disproportionately high in small user data pack-
ets such as simple setpoint targets. A simple example of how to use fast Ethernet bandwidth efficiently: if status data of 4 bytes per device for 20 devices were sent indi-
vidually, that would take up 1,680 bytes = 20*84 bytes altogether (smallest packet size with Ethernet: 64 bytes). However, only 80 bytes would be used productively for the application - that’s approx. 5% of the bandwidth, even during low-peak cycle times.

In Sercos telegrams, however, up to 1,494 bytes of all device user data is packed together with an additional 44 bytes of overheads. With packets that are a maxi-

mum size of 1,538 bytes, the bandwidth available for productive data increases to up to 97%.

Efficient use of bandwidth with Sercos

“Sercos’ efficient real-time proto-
cols result in universal, consistent machine connection with fast reac-
tion times and great accuracy.”

Example configurations

<table>
<thead>
<tr>
<th>Cycle time in µs</th>
<th>Cyclic data per device in bytes</th>
<th>Maximum number of devices (without UCC)</th>
<th>Maximum number of devices (with UCC, 250 Bytes = 20 µs)</th>
<th>Maximum number of devices (with UCC, 1,500 Bytes = 125 µs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>31.25</td>
<td>8</td>
<td>7</td>
<td>2</td>
<td>–</td>
</tr>
<tr>
<td>62.5</td>
<td>12</td>
<td>14</td>
<td>8</td>
<td>–</td>
</tr>
<tr>
<td>125</td>
<td>16</td>
<td>26</td>
<td>21</td>
<td>–</td>
</tr>
<tr>
<td>250</td>
<td>12</td>
<td>61</td>
<td>57</td>
<td>30</td>
</tr>
<tr>
<td>250</td>
<td>32</td>
<td>33</td>
<td>31</td>
<td>17</td>
</tr>
<tr>
<td>500</td>
<td>12</td>
<td>122</td>
<td>120</td>
<td>94</td>
</tr>
<tr>
<td>1,000</td>
<td>50</td>
<td>97</td>
<td>95</td>
<td>85</td>
</tr>
<tr>
<td>1,000</td>
<td>32</td>
<td>137</td>
<td>134</td>
<td>120</td>
</tr>
<tr>
<td>1,000</td>
<td>12</td>
<td>251</td>
<td>246</td>
<td>220</td>
</tr>
</tbody>
</table>

“Sercos hardware failure or break in a cable does not result in a breakdown in communication. Instead, we are able to replace defective devices or cables while the system is in operation, which significantly increases machine availability.”
Device models and profiles
The device model from Sercos not only supports pure automation devices but also hybrid devices which combine various applications in one device. It is for this reason that no device profiles but, instead, function-specific profiles are defined by Sercos.

The following profiles have already been specified:
- Generic device profile: functions for diagnosis and management functions in all types of devices
- Drive profile: consistent and continuous functions for controlling of hydraulic, pneumatic and electric drives
- I/O profile: consistent and continuous functions for controlling of modular and non-modular I/O stations
- Encoder profile: consistent and continuous functions for controlling of encoders
- Energy profile: Consistent and continuous functions for reduction of energy consumption due to shutdown, partial machine operation and shutdown.

The parameters of the different profiles are either configured into the real-time telegrams for cyclical transmission or these parameters are accessed via the service channel or the S/IP protocol.

Device description
Sercos offers the possibility to represent all types of automation devices on the field and controlling level in the different phases of their life cycle in a functional and logical manner. The device description language SDDML was developed in order to describe devices and the provided functions for the offline configuration and for their easy display in a generic engineering tool. A configuration interface (SCI) determines the network configuration and defines which slaves must be present and which are optional. The slave devices are identified via criteria which are described in the file. Furthermore, the configuration file describes the parametrization of the individual devices by means of a generic procedure. In addition, the parametrization of the master can be carried out.

FDT/DTM for Sercos
Sercos makes use of the open and manufacturer-independent FDT technology to standardize the communication between field devices and software engineering tools. It is for this purpose that device manufacturers may deliver a Sercos device with a DTM (Device Type Manager) adjusted to the device. In this case, the DTM is directly integrated into the corresponding framework application. However, an SDDML-conforming device description file can also be converted into a corresponding Sercos device DTM by means of a generic conversion (based on general rules). In this connection, the device description may already be available as file or be directly generated from the parameter set directly stored in the device or accessible (online) via the bus system.

Simplified processes – from engineering to maintenance
The configuration, startup, diagnostics and maintenance of Sercos networks is quite simple. The user can focus on the application – Sercos manages the network and supports the user in startup and diagnostics. All devices are connected with industry-compatible standard CAT5e cables or fibre optic cables. After an initialization phase, the network is synchronized and operational. Newly installed devices (hot plugging) are integrated into the communication and real-time data exchange during running operation.

“We have made very good experiences with the excellent degree of standardization of the drive profile whereby it was possible for us to integrate devices of various manufacturers to our control and to commission the machine in a remarkably short time.
Your advantages

- Commissioning without presets thanks to automatic device identification and address assignment. If required, an individual address pre-selection or address presetting via selector is possible.
- Automatic detection and compensation of duplicate device addresses.
- Simple and robust cabling on commissioning and maintenance since both Sercos III ports of the device operate identically and do not have to be distinguished when cabling.
- Simple stocking of spare parts, patch as well as crossover cables can be used.
- Commissioning is possible without master hardware due to simple integration of service PCs into the Sercos III network.
- Full diagnostic possibilities such as automatic identification of the topology and connection sequence of the users, localization and redundancy concerning cable breaks.
- Repairs and modifications of the facility without any impairment of the rest of the network by hot plugging.
- Possibility of vertical integration thanks to the option to include Ethernet based protocols.
Safety-relevant data is transmitted as a safety data container. The data container is stored in the relevant real-time device channel just like standard data – in an MDT just as in an AT. A multiplex protocol, SMP (Sercos Messaging Protocol), is used to transmit safety data that has been scanned differently without losing bandwidth despite there being shorter bus cycles. CIP Safety is a network protocol for functional safety. It has been certified by TÜV Rheinland for installation in applications with a safety integrity level of 3 (SIL3) and satisfies the IEC standard 61508 for functional safety (“functional safety of safety-related electric/electronic/programmable electronic systems”).

CIP Safety on SERCOS is a protocol for transmitting safety-relevant data via Sercos – defined in cooperation with the ODVA and certified according to IEC 61508 up to SIL3. There is no need for additional cabling for a safety bus because signals are simply transmitted alongside other real-time data on the Sercos network. The integration of drive, periphery and safety bus as well as standard Ethernet in one single network simplifies handling and reduces hardware and installation costs. Integrated safety control systems and homogeneous safety solutions can be carried out with ease.

With CIP Safety on Sercos, data is safely transmitted to the same medium using the same connection as the rest of the communication. The functionality of transport protocol and non-medium dependent CIP safety protocols lies in the end devices, which allows standard and safety devices to be operated simultaneously on the same network. Safe communication is possible across and between all network levels. The master, therefore, does not necessarily have to be a safety controller, but can also route data without having to interpret it. Safe Sercos slave devices can be connected to each other without safety controllers and can communicate securely via direct cross communication in the shortest reaction times. This gives the user real flexibility when setting up safety network architecture, when installing safety programmable controllers or when directly transmitting safety data between sensors and actuators. Moreover, by using a standard CIP network, communication from safety devices in a subnet to/from safety devices in another subnet is made seamlessly.

A single safety protocol for Sercos, EtherNet/IP and DeviceNet enables our machines and systems to be universally connected and allows safety-related process data to be transmitted.
When the application requires a Sercos ring for redundant data transmission of real-time data and therefore no free Sercos port is available, an IP switch must be integrated into the ring or into a device. Its function is to connect and disconnect the EtherNet/IP packets into the Sercos ring. The EtherNet/IP devices can be arranged in different topology types: star and line topologies as well as DLR (Device Level Ring).

The highly efficient Sercos telegrams ensure that only a part of the existing bandwidth is used for the real-data exchange. Sercos needs for an application with 64 drives only 400 microseconds and a 2 milliseconds cycle for example. This means that 1.6 milliseconds are available for the transmission of TCP/IP and EtherNet/IP telegrams.

Since the UC Channel sits directly on the Ethernet layer, TCP/IP and EtherNet/IP as well as other Ethernet users can be connected to the network without any additional hardware. Tunneling of the protocols is not required. Even before a Sercos III communication is initiated by the master, the network users can exchange data via TCP/IP, EtherNet/IP and the S/IP protocol specified by Sercos.

“The utilization of a uniform network infrastructure for Sercos and EtherNet/IP devices is an innovative approach to reducing the number of communication interfaces and therefore the hardware complexity in machines and facilities significantly. The continuous networking increases the operating efficiency in engineering and in the operation of the facilities.”

The common infrastructure complements the Sercos solution portfolio as, alongside the extensive Sercos product range, EtherNet/IP devices of various manufacturers can be implemented additionally. With this concept, the number of communication interfaces and therefore the hardware complexity will be significantly reduced in machines and facilities. The continuous networking increases the operating efficiency in engineering and in the operation of the facilities.
Member companies of Sercos International