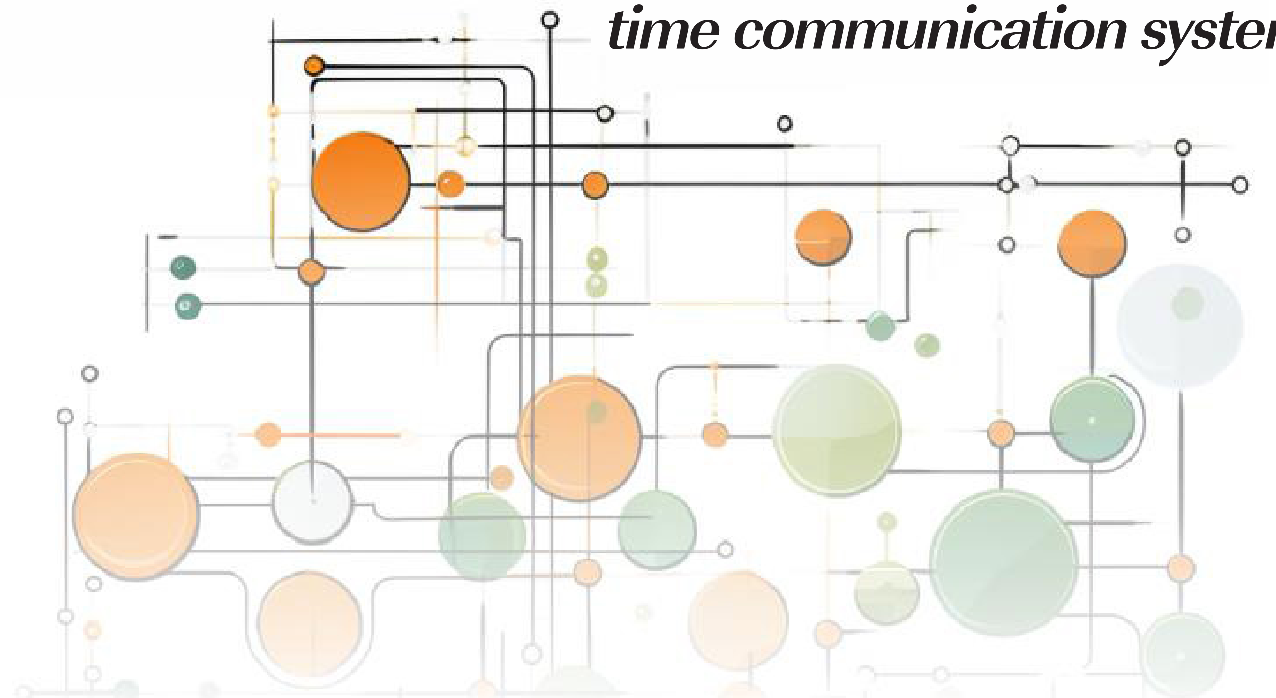
# Real-time communication: Part II - Demands of real-time communication systems



(Source: Embedded Systems Academy)

*Within this article series, Olaf Pfeiffer from Embedded Systems Academy is setting in perspective the timing requirements for different real-time capable communication systems, such as CAN, CANopen, and real-time Ethernet.*

In this second article, the author shows the different timeframes required by different applications and reviews what this means for the communication system used. The Part I, published in the December 2023 issue of the CAN Newsletter magazine, describes how to select the right real-time timeframe for certain applications. In "Part III – The temporal dynamics of CAN-based systems" (planned for the next issue) the author applies the findings from Embedded Systems Academy (EmSA) to CAN and CANopen, giving recommendations on "how to use" (configure) the communications to meet the demands found earlier. The last article "Part IV – From theory to practice: CANopen source code configuration" shows which optimization options are typically available when working with CANopen source code, here, on example of Micro CANopen Plus from EmSA.

## The demands of real-time communication systems

The ever-increasing complexity and demands of modern real-time applications necessitate robust and reliable communication systems. As established in the first part of this series, these applications span a wide spectrum of response time requirements, from seconds to milliseconds, and their success is often contingent on the precise timing of their responses. Consequently, the chosen communication system must be capable of meeting these stringent timing constraints.

However, achieving the desired real-time capabilities is not the sole consideration. In many cases, these systems also need to ensure the safety of users, equipment, and the surrounding environment. Additionally, given the growing threat landscape, ensuring the security of these communication systems has become equally critical. Balancing these requirements – real-time responsiveness, safety, and security – is a multifaceted challenge. In this second part, the specific attributes and considerations are investigated that make a communication system capable of fulfilling these demands.

## Evolution of real-time communication systems

Over time, the demands on real-time communication systems have evolved and become more stringent. In the early days, the primary focus was on achieving real-time requirements with a reasonable level of reliability. It was often deemed sufficient if the system could process and transmit data within the specified timeframes, even if occasional errors occurred.

As technology advanced and systems became more sophisticated, the need for safety became apparent. "Some-what reliable" was no longer adequate, especially for applications where human safety, product quality, or system operation was at stake. To address these concerns, specific protocols were developed to ensure that real-time systems could operate safely, even in the face of faults or disruptions. The importance of safety grew, particularly in critical applications such as transportation or medical devices.

More recently, as real-time systems increasingly became interconnected and even accessible over the internet, security emerged as another crucial consideration. With the potential for cyber-attacks and unauthorized access, it became necessary to safeguard not only the data but also the integrity and availability of the communication system itself.

Today, a comprehensive real-time communication system must meet all three criteria: real-time responsiveness, safety, and security. It is no longer advisable to start from scratch when designing an embedded communication system for any real-time application. Once, it was quite common for developers to take an ad-hoc approach, such as repurposing one of the serial ports to share it among multiple nodes, effectively creating an EIA-485-style network. However, this approach does not accommodate the increasing complexity of real-time systems.

and between devices. In general, it is not advisable to push any system "to its limits," so any networking technology you choose should have enough capability to accommodate your application's growth over time.

## Are there safety and security requirements?

Once you've established the applicable timeframe for your application, it is crucial to determine what safety and security measures are necessary. If your application must adhere to specific safety standards or certifications, your choices regarding communication networks will automatically narrow. This article focuses on the real-time requirements. When conducting your research, double-check the latest developments – all active fieldbus organizations and committees are continually working on improving both safety and security.

*Figure 1: Consider whether any signals require synchronization (Source: Embedded Systems Academy)*

## Is there a best fit?

In German, there's a saying "Es gibt keine eierlegende Wollmilchsau," which can be translated to "there is no one-size-fits-all solution" or, more literally, "there is no egg-laying wool-milk pig." This saying applies here as well. Regrettably, there is no single networking technology that is universally suitable for all applications. Each application has its unique set of requirements and constraints, making it necessary to carefully evaluate and select the appropriate communication technology and protocols. Therefore, it is essential to consider the specific needs of the application and match them with the most suitable networking technology available, taking into account factors such as required throughput, real-time responsiveness, safety, and security.

## The basics: How much data, how often?

First, assess the overall architecture of your system. In addition to real-time requirements and the timeframe within which a complete control step must be executed, consider the total number of inputs and outputs required, their distances apart, and the number of signals and their data lengths that need to be exchanged within each timeframe

## Are there synchronization requirements?

Consider whether any signals require synchronization, meaning that inputs should be captured at the same moment in time. Synchronization is critical for applications where multiple inputs are combined. In real-time communication systems, synchronization plays an important role in ensuring accurate data transmission and interpretation. Some applications demand synchronization due to their nature (e.g., syncing multiple manipulators working on the same material simultaneously), while other effects might be more subtle: Consider a scenario where an analog sensor generates input data every 100 ms based on its internal timer. The transmission of this data onto a network also occurs every 100 ms, triggered by a separate network timer. If these timers are not synchronized, they may gradually drift apart, leading to two possible scenarios:

1. Duplicate data transmission: If the network timer's window is shorter than the sensor's, the sensor may not have generated new input data by the time the network is ready to transmit. In this case, the same data could be transmitted twice.
2. Data loss: If the sensor's timer window is shorter than the network's, a new value may be generated before the previous one has been transmitted. This situation can lead to skipped or lost data.

The impact of these scenarios greatly depends on the signal and its usage. For instance, if the value represents temperature and the main processing unit only needs to know if it falls within the correct range, these scenarios have no effect. However, if it is a counter or a rapidly changing signal representing a wave, missing or duplicated data may have serious consequences.

## Other considerations

When selecting a real-time communication system, there are many additional considerations: Are off-the-shelf products, development, and diagnostic tools readily available? Can it easily integrate with existing (or planned) systems? If hard real-time of single milliseconds is a requirement, such integration may need to go "deeper" into a system, potentially requiring custom software at the lowest levels of the hardware.

## Too many choices...

Understanding the specific requirements of your application – real-time responsiveness, safety, security, system architecture, and synchronization – can guide you in selecting a suitable communication network. If you start at zero, a potential starting point for gaining an overview of available fieldbus networks is the Wikipedia entry titled "Fieldbus." However, note that this list captures only a fraction of the available fieldbus networks. The domain of industrial communication networks is vast and continuously evolving, with many fieldbus networks, some not even officially standardized. Beyond the widely-recognized fieldbus networks, many networks, often crafted by manufacturers or consortia, cater to specific applications or industries. They might offer distinct features, specialized protocols, or proprietary technology tailored to certain application needs.

For instance, the Controller Area Network (CAN) is a versatile communication network supporting numerous applications through its specialized higher-layer protocols. Higher-layer protocols such as J1939 cater to commercial vehicles (for example construction, agriculture), standardizing message formatting and signaling to facilitate manufacturers in crafting interoperable components. NMEA 2000, by the National Marine Electronics Association (NMEA), aids the integration of marine electronics, streamlining the configuration and management of intricate marine systems. CANaerospace, designed for aerospace, meets the distinct demands of avionic systems, ensuring reliable data exchange in aircrafts.

The CANopen protocol, with its flexibility, boasts many device and application profiles, such as those for elevators, emergency vehicles, and CleANopen for waste collection vehicles. These profiles determine the communication behavior and data structures for devices or entire applications, simplifying the development process. Moving beyond CAN, more than 10 different solutions exist for Ethernet-capable real-time communication, each targeting varied applications. As a general rule, if your application's real-time requirement is roughly 100 ms or more, you have a multitude of choices since most embedded communication networks or fieldbuses can fulfil these demands, even for more extensive systems. However, for vast machinery (spanning several hundred meters of cable and beyond), scrutinizing communication runtime and throughput is essential. For stringent real-time requirements, as short as 10 ms or even less, it's imperative to diligently review which network technologies can satisfy your needs. Typically, a time-triggered communication system (available on CAN, Ethernet, and other platforms) is the most deterministic. Here, each signal with real-time requirements is allocated an exclusive timeslot, ensuring predictable signal transmission.

## What's next?

As Embedded Systems Academy provides a comprehensive expertise in CAN and CANopen communications, the next part III of this series will focus on CAN and CANopen as examples for the many embedded communication systems available. The author will explore its suitability for diverse systems with real-time requirements, highlighting achievable response times, areas demanding meticulous attention, and situations that push boundaries, suggesting the evaluation of alternatives. ◀

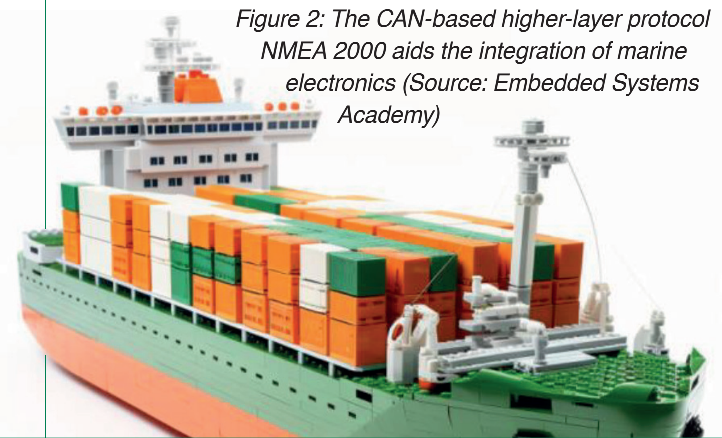


*Figure 2: The CAN-based higher-layer protocol NMEA 2000 aids the integration of marine electronics (Source: Embedded Systems Academy)*

**Author**

Olaf Pfeiffer
EmSA (Embedded Systems Academy)
info@esacademy.com
www.esacademy.de