ETHERNET FOR SPACE APPLICATIONS: TTETHERNET

Session: Networks and Protocols
Long Paper

Wilfried Steiner, Günther Bauer
TTTech Computertechnik AG, Vienna Austria

David Jameux
European Space Agency ESTEC, Noordwijk, The Netherlands
E-mail: wilfried.steiner@tttech.com, guenther.bauer@tttech.com,
david.jameux@esa.int

ABSTRACT

With the growing complexity of avionics for spacecrafts it may be beneficial to rethink architectural styles in general and adopting network architectures from other application areas in particular. Common network architectures from the civil aerospace domain implement a reliable high-speed backbone bus that integrates lower-speed field-busses. This paper discusses an integrated architecture featuring Ethernet as backbone bus that integrates SpaceWire networks.

1 INTRODUCTION

On-board spacecraft computer networks have to be robust against harsh environments including high-level radiation, which can cause transient upsets, like bit-flips, in a integrated circuits. As a result, one of the driving requirements in hardware deployment for spacecrafts lies in small memory sizes and footprints, which typically leads to specific space products.

Standard Ethernet networks, on the other side, are primarily developed with a focus on consumer electronics and office requirements and do not impose said limits on memory and footprint. On the contrary, in order to avoid message drops the requirements on message buffer memory in Ethernet switches and routers become excessive.

TTEthernet closes the gap between restricting hardware requirements from space applications and excessive hardware requirements from modern Ethernet networks. The key is the introduction of a time-triggered paradigm on top of Ethernet that allows a coordinated and pre-determined usage of the resources present in the network. As a result TTEthernet is scalable for cross-domain usage, which gives a vast economic benefit and significantly accelerates the maturity process of the TTEthernet technology.

This paper gives an introduction to the TTEthernet concepts and arising benefits from TTEthernet deployment in mixed-criticality systems. Furthermore, we present integrated network architectures using TTEthernet as deterministic high-speed backbone that interconnects individual SpaceWire networks. In particular, we discuss dataflow and the flow of synchronization in such a hybrid network, where dataflow is considered in both directions, while the flow of synchronization is considered from TTEthernet to SpaceWire, if required at all.

2 OVERVIEW OF TTETHERNET

2.1 DATAFLOW IN TTETHERNET

TTEthernet specifies services that enable time-triggered communication on top of Ethernet, the TT (Time-Triggered) Services. As depicted in Figure 1, TT Services can be viewed parallel to the common OSI layers: a communication controller that implements the TT Services is able to synchronize its local clock with the local clocks of other communication controllers and switches in the system. The communication controller can then send messages at off-line planned points in this synchronized global time. These messages are said to be time-triggered and it is the task of the off-line planning tool to guarantee that the time-triggered message schedule is free of conflict. By conflict-free we mean that it will never be the case that two time-triggered messages compete for transmission and, hence, no dynamic arbitration actions for the communication medium (for time-triggered messages) are required.

TTEthernet supports communication among applications with different real-time and safety requirements over a single physical network. Therefore, three different traffic classes are provided (see Figure 1): Time-Triggered Traffic (TT), Rate-Constrained (RC) Traffic, and Best-Effort Traffic (BE). If required, the corresponding traffic class of a message can be identified based on a message's Ethernet Destination address.

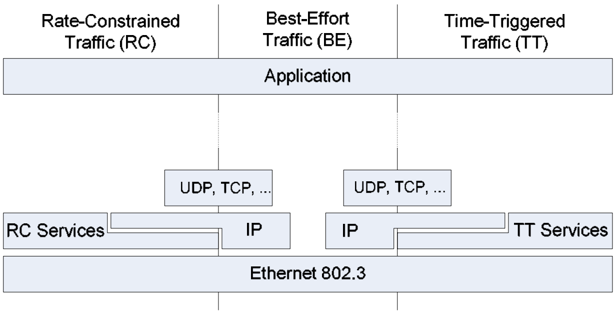

Figure 1: Interaction of standards

As depicted in Figure 1, messages from higher layer protocols, like IP or UDP, can easily be ”made” time-triggered without modifications of the messages' contents itself. This is because the TTEthernet protocol overhead is transmitted in dedicated messages, called Protocol Control Frames, which are used to establish system-wide synchronization upon those components that need to be synchronized. In short, TTEthernet is only concerned with ”when” a data message is sent, rather than with specific contents within a data message.

Time-Triggered (TT) messages are used for applications with tight latency, jitter, and determinism requirements. All TT messages are sent over the network at predefined times and take precedence over all other traffic classes. TT messages are optimally suited for communication in distributed real-time systems.

Rate-Constrained (RC) messages are used for applications with less stringent determinism and real-time requirements than strictly time-triggered applications. RC messages guarantee that bandwidth is predefined for each application and delays and temporal deviations have defined limits. In contrast to TT messages, RC messages are not sent with respect to a system-wide synchronized time base. Hence, different communication controllers may send RC messages at the same point in time to the

same receiver. As a consequence, the RC messages may queue up in the network switches leading to increased transmission jitters. As the transmission rate of the RC messages is bound a priori and controlled in the network switches, an upper bound on the transmission latency can be calculated off-line and message loss is prevented.
Best-Effort (BE) messages implement the classical Ethernet approach. There is no guarantee whether and when these messages can be transmitted, what delays occur and if BE messages arrive at the recipient. BE messages use the remaining bandwidth of the network and have less priority than TT and RC messages.

TTEthernet realizes the dataflow-integration service of messages of different traffic classes in the TTEthernet switches. A more detailed description of time-triggered communication and general dataflow-integration services can be found in [1].

2.2 Fault-Tolerant Synchronization Strategy in TTEthernet

In addition to a non-fault-tolerant Master-Slave clock synchronization service, TTEthernet specifies a fault-tolerant Multi-Master synchronization approach as depicted in Figure 2. In the first step Synchronization Masters send Protocol Control Frames to the Compression Masters. The Compression Masters then calculate an average value from the relative arrival times of these Protocol Control Frames and send out a new Protocol Control Frame in a second step which is used for re-synchronization of the local clocks. The central role of the Compression Master suggests its realization in the switch in the computer network, though this is not mandatory.

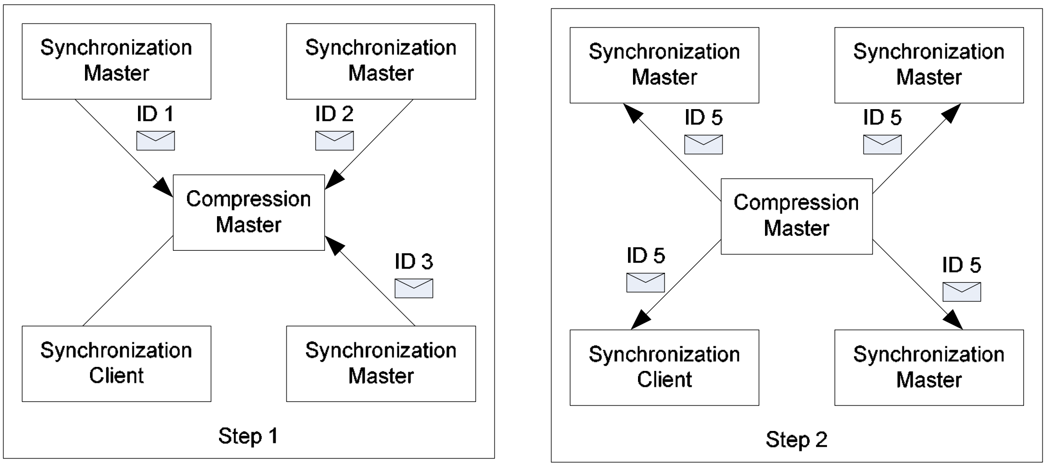

Figure 2: TTEthernet two step fault-tolerant synchronization approach

The scalability of TTEthernet from Master-Slave to Multi-Master gives a vast economic benefit: as TTEthernet can be used throughout different application domains, the cost of the realization of TTEthernet can be decreased significantly. Likewise, the cross domain usage of TTEthernet increases the probability of remaining failure detection in the realization of TTEthernet and by this matures the realization of TTEthernet significantly. This is also called ”proof-by-million” following the concept, that the probability of correctness is directly linked to the number of its implementations. Likewise the cross-domain usage contributes to the ”service history” of TTEthernet, when deployed in systems with a comparable level of criticality.

TTEthernet tolerates multiple inconsistent faults. When configured to a Multi-Master mode, TTEthernet tolerates a fully inconsistent-omission faulty communication path and even an inconsistent-omission faulty end system at the same point in time. This failure mode means that each faulty device can arbitrarily drop messages on any of its incoming communication links and on any of its outgoing communication links with potential inconsistent dropping behavior for each message. TTEthernet therefore allows a more cost-efficient realization of system architectures that require tolerance of multiple concurrent failures in the system. For example said inconsistent failure mode can even be tolerated in a system architecture that consists of only two independent communication channels. Previous realizations of communication architectures that tolerate this failure mode required at least three independent communication channels.

TTEthernet supports arbitrary multi-hop networks and even provides hooks for power-down modes of sub-networks. This can be achieved by implementing more than one Compression Master per communication channel. A system-inherent priority mechanism can be used to switch between Compression Masters, or even use the redundant set of Compression Masters per Channel as hot standby redundancy.

3 MOTIVATION FOR AN INTEGRATED ARCHITECTURE

In this section we discuss benefits of an integrated network architecture that is realized from SpaceWire, Ethernet, and Time-Triggered Services.

3.1 GENERAL BENEFITS

Design Diversity: SpaceWire and Ethernet can be argued as truly diverse network designs. This means that in the case of a design error in one of the two networks, it is highly unlikely that the same design error also occurs in the respective other network.

3.2 BENEFITS OF SPACEWIRE

Simplicity: One of the key design requirements of SpaceWire was simplicity. A simple design reduces the probability of design errors and eases the test and verification process. Furthermore, simplicity contributes to shorten the training efforts.

Efficient Memory Utilization: SpaceWire flow-control provides a natural method of high-efficient memory utilization. As messages do not have to be buffered intermediate in network components, such as routers, memories can be kept small. This, in turn contributes to a low failure rate of transient upsets.

Space-Proofed Technology: As SpaceWire is already used in active space-missions, SpaceWire-based projects can rely on well-established implementation processes.

3.3 BENEFITS OF ETHERNET

Bandwidth: SpaceWire is currently defined up to 400 Mbit/sec and typical realizations of SpaceWire will not exceed 200 Mbit/sec. Ethernet meets higher bandwidth requirements: Ethernet specifies already 1 Gbit/sec and 10 Gbit/sec bandwidths. Ethernet is, thus, suited as backbone network for individual SpaceWire sub-networks.

Group Communication: SpaceWire is a unicast protocol, besides SpaceWire Time-Codes, which are broadcasted. Hence, messages that have to be received by multiple nodes have to be repeatedly sent, leading to inefficient bandwidth utilization. Ethernet supports multicast and broadcast communication, in which the Ethernet switches are capable of relaying the same message to several ports.

Widely-Used Standard: Ethernet is the dominating standard in office communication and current market trends show that Ethernet is getting rapidly adopted for distributed control throughout the application domains, e.g.: Profinet, Powerlink, EtherCAT in industrial controls, ARINC 664-p7 for avionics controls, or upcoming Ethernet-based solutions for automotive applications. There are multiple reasons for this trend, which are not only focusing on components or chip cost: using Ethernet allows utilizing a huge knowledge and user pool. A naïve web-search reveals 103,000,000 Google-hits for Ethernet vs. 114,000 hits for SpaceWire. Furthermore, the development for critical application domains as civil avionics demands a requirements-based development that meets the highest certification standards. Re-use of the certified components for space products certainly increases their quality.

3.4 Benefits of Time-Triggered Services

Time-triggered services establish and maintain a global time, which is realized by the close synchronization of local clocks of the components. The global time forms the basis for other system properties, as for example temporal partitioning, precise diagnosis, efficient resource utilization, or composability.

Temporal Partitioning: The global time can be used as powerful isolation mechanism when components become faulty; we say that the global time operates as “temporal firewall”. In case of a failure it is not possible for a faulty application to untimely access the network. Depending on the location of the failure, either the communication controller itself or the switch will block faulty transmission attempts. Failures of the switch can be masked by powerful end-to-end arguments such as CRCs or by high-integrity designs.

Efficient Resource Utilization: The global time allows the individual components in a network to operate as a coordinated whole. This coordination directly contributes to high-efficient resource utilization. In systems that apply an event-triggered paradigm, as for example an ARINC 664-p7 network, high safety margins on resources, such as memory in switches, must be assumed. This is to guarantee that even in worst-case scenarios, when a multitude of components send at approximately the same point in time, no message will be dropped due to buffer overflows in the switches. The time-triggered paradigm allows the components to coordinate their network utilization and, hence, minimizes such safety margins.

Precise Diagnosis: A global time stamping service simplifies the process of reconstruction of a chain of distributed events. On the other side, the synchronous capturing of sensor values allows to build snapshots of the state of the overall systems.

Composability: The global time allows the specification of components not only in the value domain, but also in the temporal domain. This means that already during the design process of components, the access pattern to the communication network can be defined. The components can then be developed in parallel activities. Upon integration of the individual components, it is guaranteed that prior services are stable and that the individual components operate as a coordinated whole.

4 SpaceWire and TTEthernet Integration

Figure 3 presents four options with increasing functionality on how SpaceWire and TTEthernet may be integrated, (a) from a dataflow perspective and (b) from a synchronization flow perspective. Ethernet links are represented by bold lines, SpaceWire links by thin lines. The TTEthernet domain is represented by four switches (A…D). The SpaceWire domain is represented by four sub-networks (A…D). Within each SpaceWire sub-network, a SpaceWire Router and three SpaceWire nodes (1…3) are depicted. The interface between the TTEthernet and the SpaceWire domain can either be asynchronous (ASYNC) or synchronous (SYNC) and may require a dedicated gateway component (GW). In cases where no dedicated gateway component is present, the gateway functionality is realized within the TTEthernet switches. TTEthernet nodes are not depicted in the figure. However, for a fault-tolerant synchronization of TTEthernet (Multi-Master Mode) also TTEthernet nodes that operate as Synchronization Masters have to be present.

In addition to standard SpaceWire, we also consider SpaceWire Reliability and Timeliness (SpaceWire-RT). For SpaceWire-RT we assume that a Master-Slave clock synchronization service is realized, such that a single SpaceWire node will periodically send Time-Codes. These Time-Codes will then be used by at least a subset of nodes for synchronizing their local clocks as discussed in [1] or directly trigger a remote node to send a message as proposed as isochronous communication in [2]. Note that the objective of this example is rather to present the different integration options, than defining an overall network architecture. However, as discussed in the following (Section 4.5), communication between any two nodes in this network is possible.

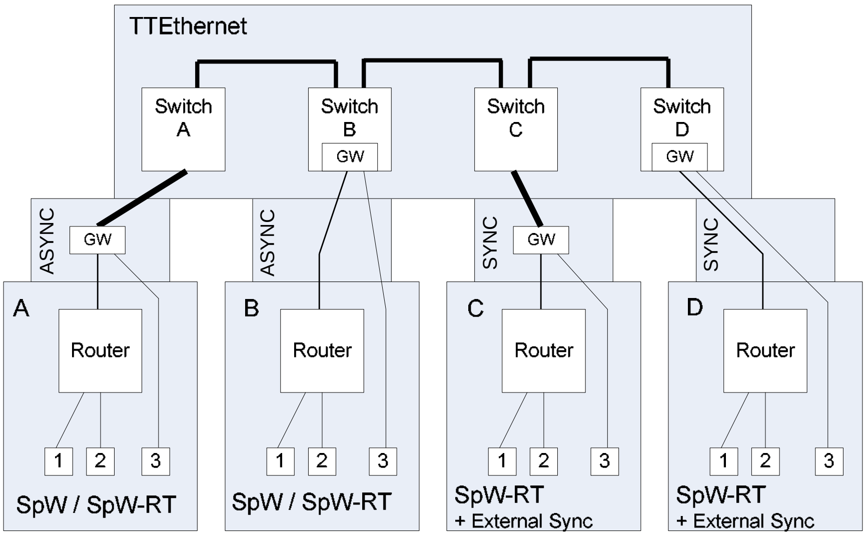

Figure 3: Four Options of SpaceWire - TTEthernet Integration

4.1 Integration Option A: Asynchronous SpaceWire / SpaceWire-RT and TTEthernet (External Gateway)

The simplest form of integration is the implementation of an asynchronous gateway component. The functional requirements on the gateway are in the value domain of converting Ethernet frames to SpaceWire messages and vice versa.

In the temporal domain SpaceWire messages that are relayed into the TTEthernet domain can be sent as either Best-Effort (BE), Rate-Constrained (RC), or Time-Triggered (TT) Ethernet frames, where only the TT type would required the gateway to be synchronized to TTEthernet. The temporal end-to-end guarantees from a SpaceWire node to a TTEthernet node have to be carefully examined and, of course, depend on the traffic type chosen. In case, where the SpaceWire message is mapped to Best-Effort traffic, no delivery guarantees for this message can be given, as this message will be treated with lowest priority in the TTEthernet system. In case, where the SpaceWire message is mapped to Rate-Constrained traffic, the delivery of this message is guaranteed. However, this message may be delayed by other Rate-Constrained and Time-Triggered traffic inside the TTEthernet domain, leading to high transmission latencies. In case, where the SpaceWire message is mapped to Time-Triggered traffic, the delivery of this message is guaranteed with short transmission latency and a jitter, which is significantly less than in Rate-Constrained traffic.

However, as the SpaceWire domain is not synchronized to the TTEthernet domain (in this Integration Option), the SpaceWire message will be delayed already in the gateway until the next sending slot of the gateway is reached. By assigning sending slots with short period and, thus, granting more bandwidth to the gateway, this delay can be minimized.

Furthermore, the gateway can operate as a data concentrator, which means that the gateway collects a magnitude of SpaceWire messages and forms a single Ethernet frame. Of course, this data concentration would not be restricted to messages from a single SpaceWire node, but may very well compress data from different nodes that may or may not be attached to the same SpaceWire router (e.g. messages from node 1 and messages from node 3). In principle, this data concentrator function can be also reversed from the TTEthernet to the SpaceWire domain. This may be of particular use, if only fractions of Ethernet frames communicated in the TTEthernet domain are of relevance in a respective SpaceWire sub-network.

4.2 Integration Option B: Asynchronous SpaceWire / SpaceWire-RT and TTEthernet (Integrated Gateway)

In contrast to Integration Option A, Integration Option B realizes the gateway functionality inside the TTEthernet switch device. The merge of the switch device with the gateway leads to a direct reduction of weight and power and, hence, indirectly to a reduction of cost.

From a fault-tolerance perspective we observe three emerging properties. Firstly, the system reliability increases with the reduction of the number of components in the system. Secondly, the gateway function benefits from design decisions and methods that have been chosen for the TTEthernet switch: in order to restrict the failure modes of a TTEthernet switch it can be implemented as a self-checking pair. In case of TTEthernet this is done by a so called COM/MON design, in which a switch is constructed out of a dual-chip solution. One chip, the commander (COM) acts as switching device, the second chip, acts as monitor (MON) that controls the output of the commander. When the monitor detects a failure of the commander, the monitor resets the COM/MON device, or parts of the COM/MON device, e.g. one outgoing port of the switch. Thirdly, the semantic filter and the leaky-bucket protection mechanisms present in the switch for the TTEthernet domain can be leveraged for SpaceWire. The semantic filter can analyse the semantics of a SpaceWire message and discard this message in case of semantic failures or syntactic inconsistencies (e.g. a checksum failure). The leaky-bucket mechanism checks that a minimum distance between two SpaceWire messages is respected. For example, assume that SpaceWire node 3 becomes faulty and starts to behave as babbling idiot (starts to send arbitrary messages). The leaky-bucket mechanism will silently discard messages if they arrive too close to each other at the gateway and, thus, prevents a monopolization of the network by the faulty node 3. Note, that in such a configuration, the TTEthernet switch can be seen as Central Guardian for a SpaceWire network (in case where the messages from node 3 are sent back to node 1 and 2).

4.3 Integration Option C: SpaceWire-RT with External Synchronization and TTEthernet (External Gateway)

The gateway functionality can even be extended by an external clock synchronization service that synchronizes the SpaceWire domain to the TTEthernet domain. For this, the gateway will synchronize to the TTEthernet domain and send SpaceWire Time-Codes itself into the SpaceWire domain, which are aligned to the TTEthernet global time. This overall synchronization allows the most efficient usage of resources in the integrated architecture, as even the over-sampling effects as discussed in Section 4.1 are avoided.

4.4 Integration Option D: SpaceWire-RT with External Synchronization and TTEthernet (Integrated Gateway)

Again, the gateway can be integrated within the TTEthernet switch with perpetuation of the functionality discussed under the previous integration options. However, as system-wide synchronization is established, also the time-triggered protection mechanism can be implemented. This mechanism extends the leaky-bucket mechanism in that not only a minimum distance between messages is controlled, but also the absolute timing of a message, as the sending slots are a priori defined. This is, in principle, also possible under Integration Option B, but would cause the maintenance of two global times (the TTEthernet and the SpaceWire global time).

4.5 SpaceWire Tunnelling Over Ethernet

The integrated SpaceWire – TTEthernet architecture not only allows a dataflow from SpaceWire to TTEthernet and vice versa, but also tunnelling of SpaceWire messages from SpaceWire over TTEthernet to SpaceWire. This may be of particular usage, (a) to establish spatial partitioning between two SpaceWire sub-networks, and (b) allows TTEthernet to be used as high-speed backbone bus for SpaceWire.

5 Conclusion

This paper has given a short introduction to the TTEthernet technology with a focus on dataflow and synchronization. We discussed benefits of an integrated SpaceWire – TTEthernet architecture and presented four integration options. These integration options allow seamless dataflow and time synchronization between SpaceWire and TTEthernet.

6 References

1. W.Steiner, R.Maier, D.Jameux, A.Ademaj, “Time-Triggered Services For SpaceWire”, SpaceWire Conference, Nara, Japan, 2008
2. S. Parkes. “The Operation and Uses of the Time – Code”, International SpaceWire Seminar, 2003