A Modular, Scalable Avionics Architecture for Future Exploration Missions

Christian Fidi[1]
TTTech Computertechnik Ltd., Vienna, Austria

Andrew Loveless[2]
NASA Johnson Space Center, Houston, TX, 77058

Future manned missions to deep space will require vehicle architectures with higher levels of autonomy and fault tolerance. Moreover, such vehicles will have even greater constraints on size, weight, and power – reducing the potential for sparing and increasing the need for hardware commonality. While classical Ethernet is attractive for its flexibility, high throughput, and widespread availability, it cannot meet the needs of systems requiring strict guarantees regarding successful and timely message delivery. TTEthernet extends classical Ethernet with a decentralized clock synchronization service enabling the deterministic delivery of time-triggered messages. Additionally, it provides two forms of event-driven communication, together enabling mixed-criticality traffic to coexist in the same physical network. This paper explores how TTEthernet technology can be leveraged to simplify the design and integration of distributed spacecraft systems.

I. Introduction

NASA aims to expand human exploration into deep space and to the surface of Mars. Unlike in current operations to low earth orbit (LEO), where delivery and return of astronauts and cargo can be accomplished in a matter of hours, any journey to Mars would take months. Moreover, such missions cannot be continuously resupplied from earth, nor quickly aborted in case of emergency. This is an entirely different operating regime, not just due to limited physical access, but also a reduced ability to communicate with Earth-based teams. Future vehicles must be more self-reliant and automated to operate safely and productively in deep space. Moreover, such vehicles will have even more severe power, volume, and mass restrictions. Under the NASA Advanced Exploration Systems (AES) program, technologies are maturated to meet these challenges. One particular area of interest is the advancement of onboard communication systems capable of satisfying the fault tolerance and reliability requirements of future missions. At Johnson Space Center (JSC), the Integrated Power, Avionics, and Software (IPAS) facility is used to evaluate and advance the most promising of these technologies.

This paper describes a novel approach for highly modular and scalable spacecraft avionics based on the deterministic Ethernet technology *TTEthernet*. Today's spacecraft avionics are characterized by a broad variety of processing modules, operating systems, and interfaces for exchanging data. The need for vehicle software to accommodate such an assortment of building blocks is a large contributing factor to the growing cost of software development and integration in spaceflight projects. Similar challenges have triggered developments in other industrial domains – including AUTOSAR in the automotive industry and Integrated Modular Avionics (IMA) in commercial aircraft. These initiatives are all based on open standards for both computing platforms and the interfaces between them. A network technology capable of partitioning traffic with different criticality requirements over the same physical media enables the construction of standardized computing platforms able to perform any number of roles within a vehicle architecture. Gateways between differing vehicle networks can be eliminated and the need for unique software development can be reduced. Specifically, such an approach could:

- Reduce the need to support different network technologies for critical and non-critical traffic.
- Enable the qualification and testing of vehicle subsystems in isolation.

[1] Product Manager, Aerospace, Schönbrunner Str. 7, 1040 Wien, Austria
[2] Network Engineer, Command & Data Handling Branch, 2101 NASA Parkway, Houston, TX

- Enable the reuse of application software developed based on standardized interfaces.
- Separate safety-critical and non-critical applications in both the shared computer and network.
- Leverage synchronization services to enable distributed computing and increased fault-tolerance.

The main characteristics of the envisioned system architecture are the use of general purpose computers (e.g. COTS single board computers) in combination with a Time-Triggered Ethernet network. The Time-Triggered Ethernet standard SAE AS6802[1] specifies a global fault-tolerant synchronization protocol enabling network composability and simpler redundancy management. The underlying time-triggered paradigm provides the means for synchronization of flight software execution and data distribution between computers. Time-Triggered Ethernet provides broad compatibility with established Ethernet standards. By adding TTEthernet switches to an Ethernet network, guaranteed hard real-time communication pathways can be created without impacting existing applications. Within a spacecraft, TTEthernet eliminates the need for dedicated command and control buses by combining low-rate deterministic traffic with high-volume non-critical traffic on a single physical link.

II. TTEthernet Technology

TTEthernet implements time-triggered communication as an additional quality of service (QoS) to standard IEEE 802.3 Ethernet. As illustrated in Figure 1, the time-triggered services are implemented on Layer 2 in parallel to the traditional ISO/OSI layer model. TTEthernet components, such as switches and end systems, provide these services in hardware, enabling the synchronization of their local clocks with the other TTEthernet components in the network. The interaction between TTEthernet participants establishes a common notion of time. Messaging between devices is accomplished according to a static periodic communication schedule, in which TTEthernet components transmit and receive data at predetermined points in time.[2] This time-triggered paradigm ensures message exchange is fully deterministic and free of conflicts. Besides Time-Triggered (TT) traffic, TTEthernet also supports two forms of asynchronous messaging for use in mixed-criticality systems – Rate-Constrained (RC) traffic and Best-Effort (BE) traffic. The identification of the traffic type is based on a field in the corresponding Ethernet destination address. TTEthernet switches guarantee real-time communication by partitioning the physical media based on the bandwidth consumed on the line.

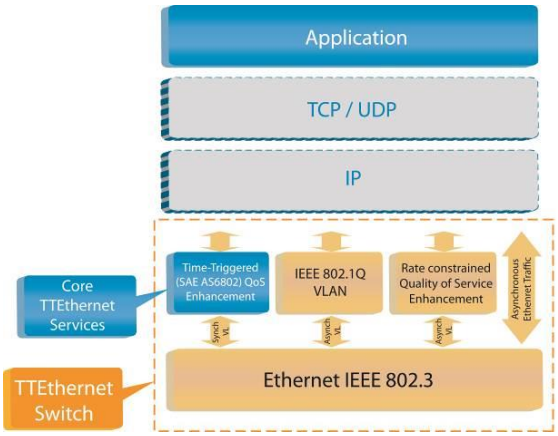

Figure 1. TTEthernet QoS in the OSI model.

TTEthernet's time-triggered and asynchronous traffic classes are interoperable over the same physical layer. This interoperability is illustrated in Figure 2, where frames with different traffic classes are integrated over a single physical link. The dataflow on the link is full duplex, since dedicated physical wires are used for sending and receiving. This structure enables a bandwidth of 100 Mbit/s in both directions. Moreover, the bandwidth can be divided amongst logically independent data paths. Each path may be composed of statically defined time-triggered traffic, rate-constrained traffic with predefined inter-message gaps, or standard Ethernet traffic. Time-triggered traffic is periodic, and different data flows can be assigned different messaging periods. Theoretically, each physical link could support different periods and triggers. However, the scheduling of a large network with high time-triggered bandwidth usage is an NP-complete decision problem. As such, a scheduling tool is needed to generate network configurations with low latency and jitter.

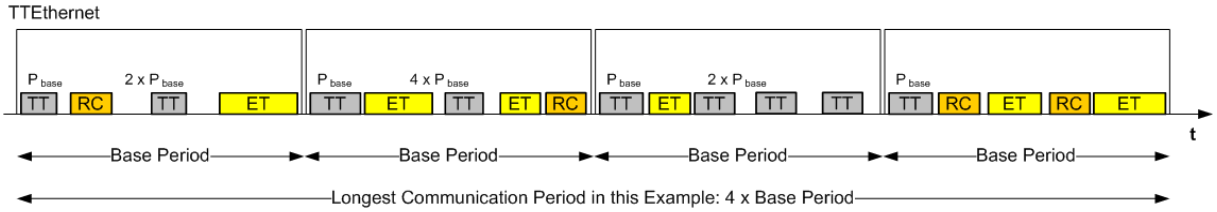

Figure 2. Traffic flow integration over a TTEthernet link.

Event-triggered traffic (rate-constrained and best-effort) can be placed between time-triggered frames and will be handled based on their priorities. As a result, the number of unique interconnections onboard a vehicle can be reduced. When using event-triggered traffic for critical applications, deterministic behavior is generally only possible when bandwidth usage is kept below 25%.[3] However, with a time-triggered access scheme, the bandwidth usage is limited only by the synchronization precision and overhead of the protocol control frames (each 64 bytes).

The time-triggered mechanisms enable the determinism of messages to be well characterized at design time, when the network schedule is initially planned. The tooling enables messages to be scheduled only as they are need by the application software – e.g. frame 1 should be received by end system 3 at a 2 ms offset from the start of a given period. The fact that data exchange is naturally aligned between communicating devices can be exploited to simplify strategies for managing redundancy, constructing voting systems, and performing distributed processing. Moreover, the scalability of TTEthernet from Master-Slave to Multi-Master architectures enables its use for multiple vehicle subsystems with differing criticality and fault-tolerance requirements (e.g. non fault-tolerant to dual fault-tolerant applications).

III. Advanced Tooling and Partitioning

Time-triggered Ethernet is designed for modular mixed-criticality networks in the automotive, aerospace, and industrial domains. The time-triggered nature of its communication makes it appropriate for applications requiring tight control loops, such as command and control onboard spacecraft. Moreover, the technology enables the synchronization of software applications to the network time base. By synchronizing task and network scheduling, it is possible to realize a fully distributed real-time system. Since the configuration of such a system can be quite complex, system-level configuration and verification tools are necessary in addition to those used for network scheduling.

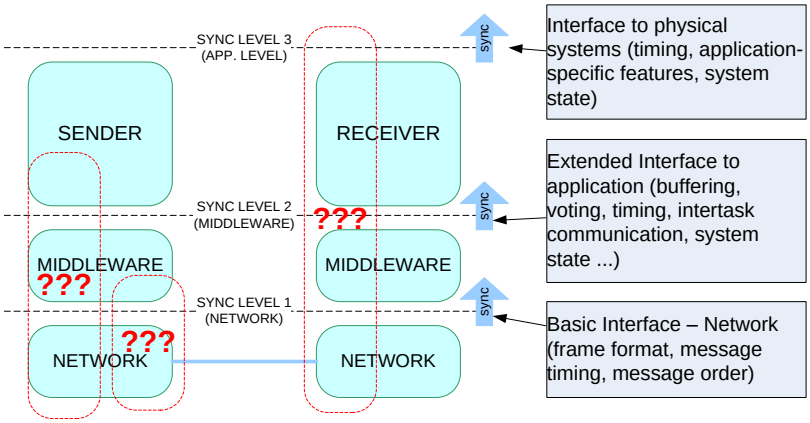

Figure 3. Where are key system interfaces defined?

Such tools provide a higher-level interface enabling the scheduling of logical message flows between applications or partitions rather than network components. This can be accomplished by considering the worst-case time for software to generate data and pass it through the host interface, ensuring that it is available in time for its scheduled transmission. In this way, system-level tools can augment the functionality of the network-level scheduler. The network-level tooling provides the timing model to the system-level tool for the scheduling of messages. Additionally, it creates the configuration files loaded on the physical network devices. A system-level tool is instead responsible for creating operating system and middleware specific configurations, as well as defining interfaces used by the applications. Qualified verification tools are necessary to ensure that configurations at different system layers are consistent with one another.

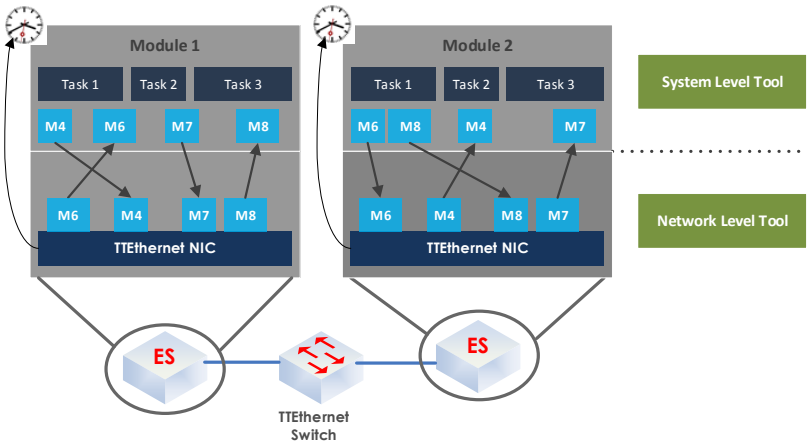

Figure 4. Relationship between network-level and system-level tools.

IV. Modular Approach with TTEthernet

Future manned spaceflight missions will require assembly and re-assembly of different vehicles in orbit to fulfill their mission requirements. In such cases, vehicles will likely be required to dock with and control effectors located within another vehicle, or perform autonomous reconfiguration from high reliability to high availability or low power configurations. This sort of flexibility is naturally supported by standard Ethernet traffic. However, it can also be achieved with fully deterministic time-triggered traffic. Figure 2 5 illustrates a scenario in which multiple vehicles are launched and must be assembled in space. This docking may be permanent (e.g. habitat build-up) or temporary (e.g. resupply). This scenario places requirements on the vehicles' data networks to maintain the necessary QoS when the vehicles are both separate and docked.

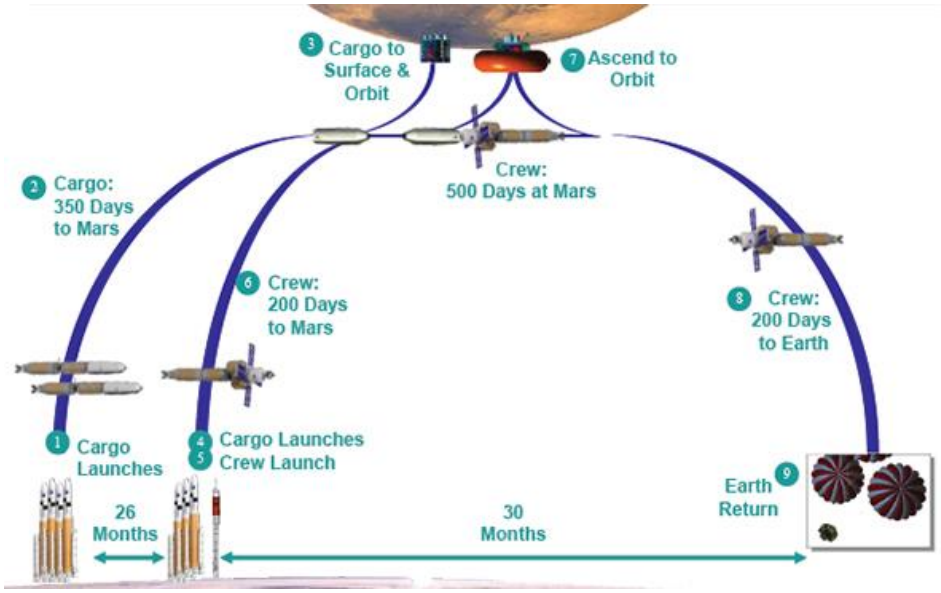

Figure 2. NASA Design Reference Architecture 5.0 for manned missions to Mars.[4]

Figure 6 illustrates how Time-Triggered Ethernet mechanisms allow the networks from two vehicles to be combined, enabling one vehicle to control the other. Accomplishing this task requires the time bases from each vehicle to be merged into the same synchronization domain. Once both networks are synchronized to the same global clock, time-triggered messages can be exchanged between them. In TTEthernet, the combination of vehicle networks is performed through the use of different synchronization priorities. When two vehicle networks are joined, the one

with the lower priority will adopt the time based established by the other. The time base of the higher priority network will not be impacted.

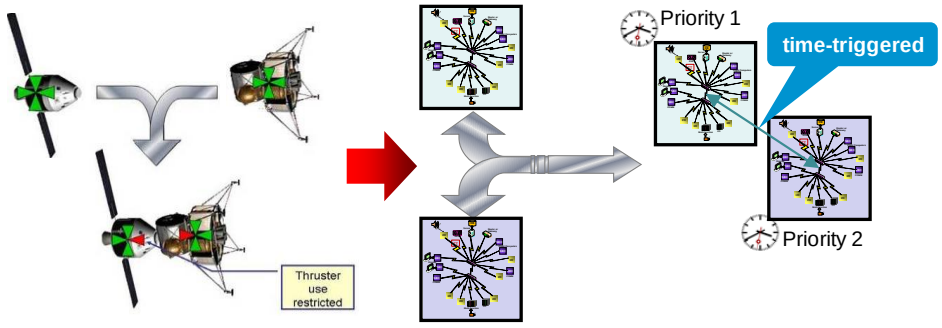

Figure 3. TTEthernet in a system-of-systems architecture.

To enable successful message transmission, the network configurations describing the interface between the two vehicles must be consistent. The easiest way this can be achieved is by pre-allocating the data flows between vehicles. In cases where predefining the traffic patterns is not possible, it is necessary to instead reconfigure those devices involved in the data exchange. To prevent the new data flows from impacting flows which are already scheduled, the tooling must be capable of both 1) reserving time slots for future use and 2) scheduling incrementally as new flows are added.

V. Conclusion

This paper has described how TTEthernet can be used in the design of advanced spacecraft architectures requiring high degrees of autonomy and fault tolerance. The use of three traffic classes allows the network to accommodate data flows of mixed-criticality within the same physical network, reducing the need for unique interconnects between components. Redundancy management is handled transparently by the network hardware, simplifying the fault tolerance scheme and reducing the complexity of the application software. Moreover, this paper introduced how distributed systems can be designed using a system-level approach – synchronizing the various building blocks composing the system and coordinating the configuration of the network and software. By using tools for the specification and verification of system interfaces, the complexity of integrating the software components can be reduced.

References

[1] "Time-Triggered Ethernet," SAE AS6802, Nov. 2016.

[2] Kopetz, H., Ademaj, A., Grillinger, P., and Steinhammer, K., "The Time-Triggered Ethernet (TTE) Design," *Proc. IEEE International Symposium on Object-Oriented Real-Time Distributed Computing*, May 2005.

[3] Hodson, R., Chen, Y., Morgan, D., Butler, A., Schuh, J., Petelle, J., Gwaltney, D., Coe, L., Koelbl, T., and Nguyen, H., "Heavy Lift Vehicle (HLV) Avionics Flight Computing Architecture Study," Tech. rep., NASA Langley Research Center, Hampton, VA, Aug. 2011.

[4] Drake, B., "Human Exploration of Mars Design Reference Architecture 5.0," July 2009.