Architecting Network Latencies for Mixed Criticality In-Vehicle Applications

Lulu Chan, Bart Vermeulen, Nicola Concer, Philip Axer – NXP Automotive

October 2018

Introduction

An Automotive networking revolution is underway, driven by the faster data speeds and higher data capacity required for safety and comfort for increasingly autonomous and connected vehicles. Well-established CAN-centric in-vehicle networks (IVNs) can no longer support the large and diversified amount of data needed by the new features customers expect from their next vehicles. The ever-increasing bandwidth required to support always-connected telematics units, high-resolution head-units, radar sensors, surround-view cameras and more have transformed CAN-centric IVNs into domain-based in-vehicle architectures. In these architectures, a central gateway connects to several domain electronic control units (ECUs) using Automotive Ethernet communication technology. Each domain ECU implements specific functions such as driver assistance, infotainment, and body control.

Automotive Ethernet(1) is the de-facto standard solution for the communication backbone of these architectures, because of its full-duplex and scalable bandwidth, and the wealth of associated standards, tools and solutions available in the adjacent consumer and industrial markets. To meet specific time-constrained application requirements, a set of time-sensitive networking (TSN) standards is released by the IEEE, the governing standardization body of Ethernet. These standards enable network functions such as distributed node synchronization, deterministic or bounded network latencies between end nodes, and network-path redundancy.

In this whitepaper we discuss IEEE 802.1Qbv "Enhancements for Scheduled Traffic," a specific TSN standard which defines mechanisms to achieve deterministic communication latencies between network end nodes. This paper is designed for professionals who are interested in the application of IEEE 802.1Qbv in an Automotive Ethernet network.

In Section 1, we introduce the TSN task group, the IEEE 802.1Qbv standard and the IEEE 802.1AS standard. Section 2 presents four relevant "what if" scenarios to highlight the impact of having scheduled traffic capabilities, as well as a description of our physical Automotive Ethernet setup. In Section 3, we reveal the cost/benefit analysis for each of these scenarios before presenting our final conclusions in Section 4.

Table of Contents

Time-Sensitive Networking .... Page 2

Evaluation Scenarios and Physical Network Setup .... Page 4

Evaluation and Results .... Page 5

Conclusion .... Page 10

References .... Page 11

1. Time-Sensitive Networking

Overview

The IEEE 802.1 TSN task group[2] is working on a set of IEEE 802.1 standards to provide deterministic services in IEEE 802 networks, and to support guaranteed frame transport with low and bounded latency, low packet delay variation, and low packet loss. This task group evolved from the former IEEE 802.1 Audio/Video Bridging (AVB) task group[3], originally focusing only on A/V applications. The TSN task group extends the standards created by the AVB task group by adding supporting mechanisms for more application types. Key TSN mechanisms include time synchronization, stream reservation, quality of service, redundancy and security.

IEEE 802.1Qbv – Enhancements for Scheduled Traffic

An automotive Ethernet switch enables the sharing of a physical Automotive Ethernet link between multiple users and extends the point-to-point physical link to full networks. Typically, Ethernet frames go through three subsequent stages after entering a standard Automotive Ethernet switch:

1. Ingress stage: Incoming Ethernet frames are received and go through various checks, after which they may get dropped, or are stored after optional modifications.
2. Forwarding stage: Stored frames are forwarded to the egress queues of one or more ports. The specific egress port(s) of an Ethernet frame depend on its source/destination MAC address and VLAN ID. The egress queue depends on the priority of the frame.
3. Egress stage: Frames in an egress queue are selected for transmission following a strict priority mechanism, meaning that the frames that are present in higher priority queues are selected before the frames that are present in lower priority queues.

An Automotive Ethernet switch that supports the mechanisms in the IEEE 802.1Qbv standard handles time-scheduled traffic in the network by using a time-aware shaper (TAS)[4]. This shaper is needed for applications where messages need to be delivered within tight timing constraints, irrespective of how much other data is also communicated over the same shared physical network. Examples of such applications are sensor to sensor-fusion data delivery and processing, onboard interprocessor communication between SOCs and control MCUs, intra-domain communication for specific signals, and control traffic delivery without contention from competing traffic. Figure 1 illustrates the mechanisms specified in the IEEE 802.1Qbv standard, in the egress stage of an Ethernet switch.

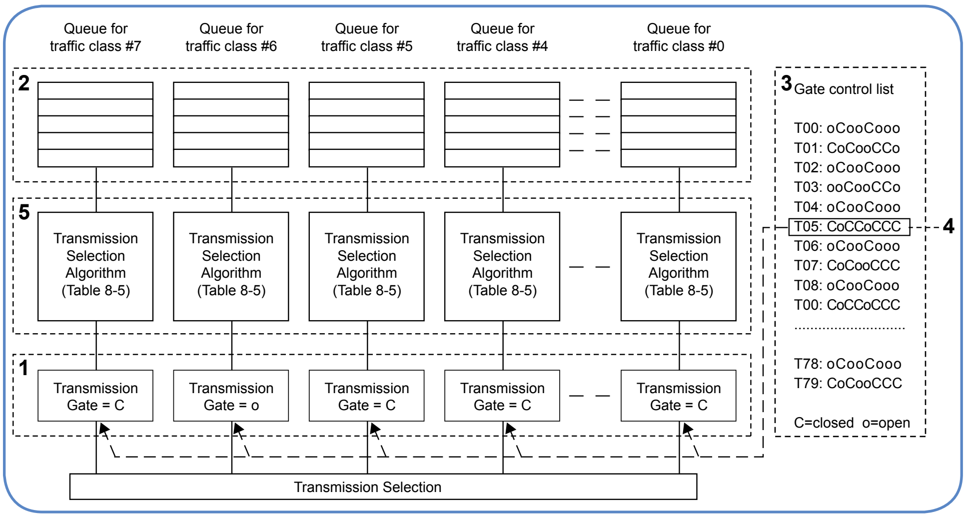

Figure 1 Egress stage of an Ethernet switch with frame selection using IEEE 802.1Qbv transmission gates[4]

Transmission gates (1) are added to the egress queues (2) of the switch that control whether frames in these queues are eligible for transmission. These gates are in turn controlled by a gate control list (3) with gate control entries (4). Each entry specifies for each egress queue whether the gate is open ("o") or closed ("C"), and for how long. Frames in an egress queue are eligible for transmission selection following the other selection algorithms (5), if and only if that queue is open and will remain open for the duration it takes to transmit the frame. This means that no frames from a queue should still be in transmission when its gates is closed. This mechanism prevents any ongoing transmission from a closed queue from potentially blocking the immediate transmission of a frame from another queue during its scheduled open slot.

The Ethernet switch can guarantee the immediate transmission of a timing-critical Ethernet frame when it arrives at a queue's gate at the moment it is scheduled to be open while the other gates are closed. In a mixed-criticality traffic scenario in an IVN, having this IEEE 802.1Qbv support means we can guarantee that no additional delay is introduced in the Ethernet switches by the ongoing transmission of other traffic. Figure 2 shows an example of mixed criticality over two cascaded ethernet switches, where the critical traffic is scheduled to go through without additional delay. There are exceptional cases where such a guarantee cannot be achieved even when Qbv is supported; for example, when the switch cannot know the transmission time of a frame. We will not go into these exceptional cases in this paper.

The timing-critical traffic source and the switch need to synchronize their actions in time for this Qbv mechanism to be effective and to guarantee the lowest latency through the switch. For a communication path through a larger network, we need to synchronize all switches with the sources of timing-critical traffic to obtain this lowest end-to-end latency guarantee through the network. We can use the clock synchronization mechanism described in the IEEE 802.1AS standard for this.

IEEE 802.1AS – Clock Synchronization

The start and execution of the IEEE 802.1Qbv schedule in switches can be synchronized following the Avnu Automotive Profile in the IEEE 802.1AS standard(5). This gives all TSN-capable switches a common view of time provided by a grandmaster clock in the network. The switches can use this common view of time to traverse through their gate control lists and thereby coordinate the opening and closing of the gates on their egress queues. When all components in a network have a common view of time:

- Sensor fusion applications can have a sense of the originating time of a frame, when the sensor nodes timestamp their frames when their payload information was sampled.
- Media applications can have a sense of the presentation time of a frame, where the application end nodes receive the timestamp at which they should present their media.
- Timing-critical applications can agree on a network communication schedule over time to guarantee the lowest end-to-end latency for control traffic by setting all switches to block all other traffic queues when the timing-critical traffic is expected to come through.

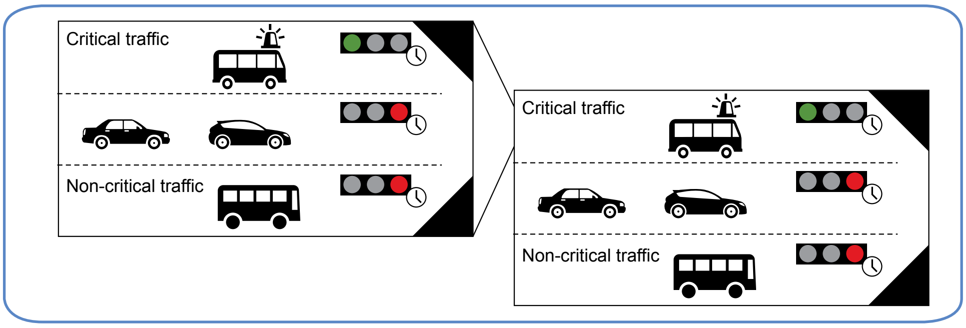

Figure 2: Minimizing end-to-end latency for safety-critical traffic in cascaded switches using IEEE 802.1Qbv

2. Evaluation Scenarios and Physical Network Setup

There are many ways to configure an IVN to support a mixed-criticality traffic scenario. In the remainder of this paper, we evaluate four “what if” scenarios relevant to highlight the benefits and costs of having IEEE 802.1Qbv’s scheduled traffic capabilities in your network:

1. What if the critical traffic source is the only user of the network?
2. What if the critical traffic must share the network with other users?
3. What if you have a critical periodic traffic source that you cannot synchronize with your network?
4. What if you need to minimize the jitter at the receiver of the scheduled traffic?

To evaluate these four scenarios, we built a physical network setup with two Automotive Ethernet switches and various end nodes to send and receive traffic with different criticality. Figure 3 shows a block diagram of this setup and Table 1 lists the hardware we used and their Ethernet traffic characteristics.

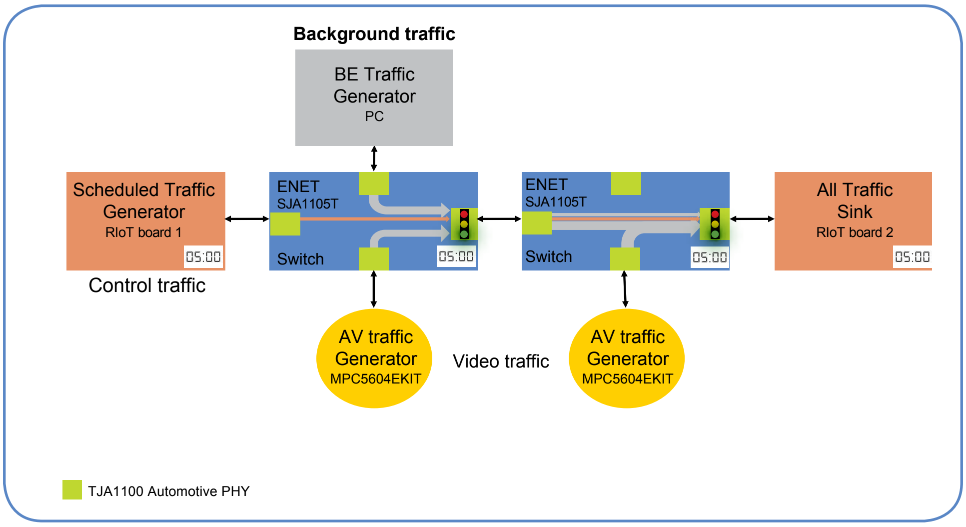

Figure 3: Block diagram of the hardware setup created to evaluate the impact of Qbv features

Component	Purpose	Traffic type	Period	Frames per period	Average frame size	Average output data rate
RIoT board 1	Source	gPTP	1 s	5	~72 B	3.7 kbit/s
RIoT board 1	Source	ST	1 s	1	64 B	512 bit/s
PC	Source	BE	-	-	1518 B	100 Mbit/s
MPC5604EKIT	Source	AV	-	-	1438 B	2 x ~25 Mbit/s
SJA1105T Application Board	Switch	Mixed	-	-	-	100 Mbit/s
RIoT board 2	Sink	Mixed	-	-	-	-

Table 1: Components used in the hardware setup

The traffic sources generate best effort (BE), audio/video (AV), and scheduled traffic (ST), forming the mixed-criticality traffic use case in all four scenarios. In this setup, we use iPerf(6) on a PC to produce best effort traffic in the form of maximum-sized UDP frames at 100 Mbit/s. We use two NXP® MPC5604EKIT(7) cameras to generate video traffic in the form of AVTP frames(8) at approximately 25 Mbit/s each, and we instrumented RIoT board(9) 1 to send out a scheduled traffic Ethernet message at a 1-second interval, starting at a fixed point in time.

Additionally, the scheduled traffic generator acts as the gPTP grandmaster in all scenarios, using the IEEE 802.1AS gPTP protocol to synchronize the scheduled traffic generator with the Ethernet switches to align their Qbv schedules.

For the Ethernet switches, we use the NXP SJA1105T Automotive Ethernet switch(10). In these switches, we assign the best effort traffic, video traffic and gPTP traffic to the egress queues with priority 0, 3 and 6 respectively. We map the scheduled traffic to the egress queue with priority 1 or 7, depending on the evaluation scenario. We use the support for time-aware shaping of the switches in our hardware setup to help illustrate the benefits and cost of using IEEE 802.1Qbv in our four evaluation scenarios.

3. Evaluation and Results

Using this hardware setup, we evaluated four relevant "what if" scenarios to highlight the impact of having scheduled traffic capabilities in an in-vehicle network. In this evaluation, we look at two key application performance metrics; end-to-end latency and jitter. We define end-to-end latency as the time it takes for a scheduled frame from transmission out of its source to reception at the sink, as shown in Figure 4. We define jitter to be the deviation from the expected periodicity of the scheduled traffic at the traffic sink in our setup.

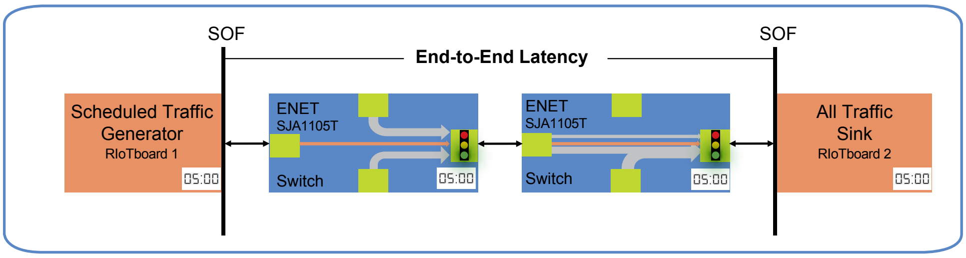

Figure 4: End-to-end latency in our physical setup

Figure 5 shows an example of the cyclic schedule we use in the physical setup for the scheduled traffic. The time slot in the middle of the schedule is when the scheduled traffic is expected to arrive. The figure shows an "ooo" schedule, which we define as a schedule where the queue for the scheduled traffic is both open inside and outside its time slot. Alternatively, in the third and fourth scenario, we evaluate the use of a "CoC" schedule, where the queue for the scheduled traffic is closed outside its time slot. The time slot duration is set to 200 μs, to cover the jitter in the scheduled traffic generator.

In the results, we normalize all measured end-to-end latencies to a reference point, which is the measured latency for a 64-byte frame to travel across the network without competing traffic present. The purpose of this normalization is to observe only the additional latency caused by competing traffic, added on top of the lowest achievable end-to-end latency.

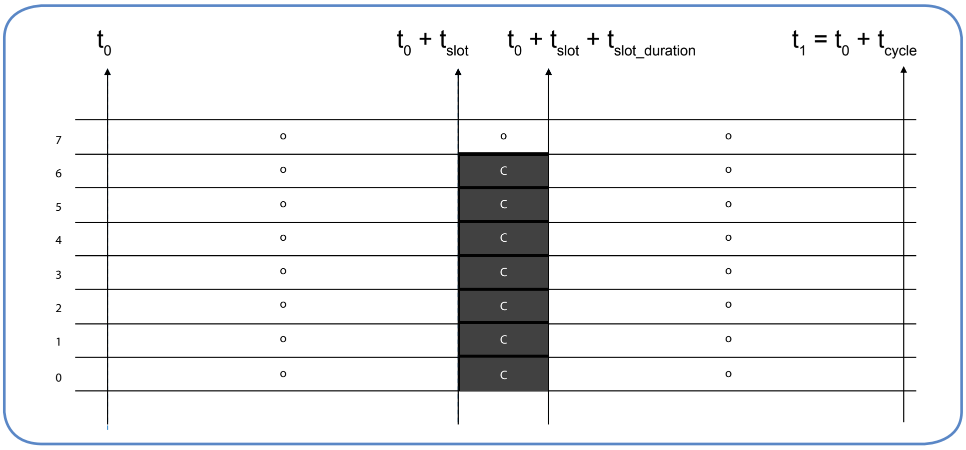

Figure 5: Example of an "ooo" Qbv schedule in the switch

Scenario 1: What if the critical traffic source is the only user of the network?

We expect the best achievable end-to-end latency to occur when the scheduled traffic experiences no delay through the network due to the presence of other traffic. When the Qbv schedule is applied, the network is scheduled to ensure this best end-to-end latency for the critical traffic that the schedule is created for. When the critical traffic is the only user of the network, the Qbv schedule should make no difference to the end-to-end latency compared to the case without the Qbv schedule. Figure 6 shows the added latency results of the case for the scheduled traffic without any other traffic present, both with and without IEEE 802.1Qbv support enabled in the switches. The result is the same for both cases, having no additional latency caused by interfering traffic. This is the expected behavior as there is no other traffic present that can block the transmission of the scheduled traffic in the Ethernet switches, meaning that the scheduled traffic will not experience any additional delay.

Scenario 2: What if the critical traffic must share the network with other users?

Figure 7 shows the measured latency results for the traffic from the scheduled traffic generator in case the best effort and video traffic defined in Table 1 are added to the network. This figure shows a varying additional latency between 0 and ~500 μs in case a Qbv schedule is not used. The variation in the added latency for the scheduled traffic without a Qbv schedule in the switches depends on the egress priority that we assign to the scheduled frame, and whether the frame arrives at the switch during the transmission of another Ethernet frame. As expected, when we use a Qbv schedule in the Ethernet switches, we again observe that there is no added latency caused by interfering traffic.

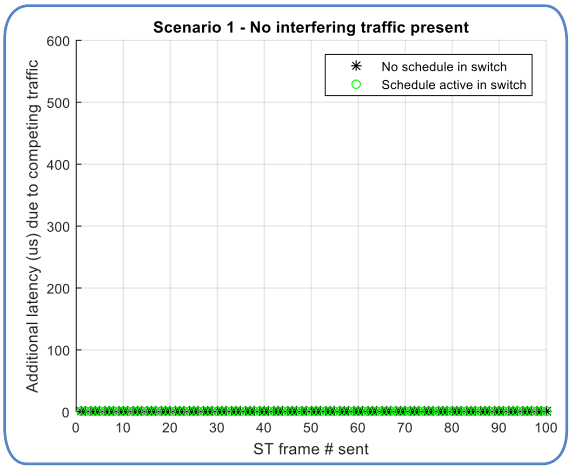

Figure 6: Impact of scheduling without interfering traffic

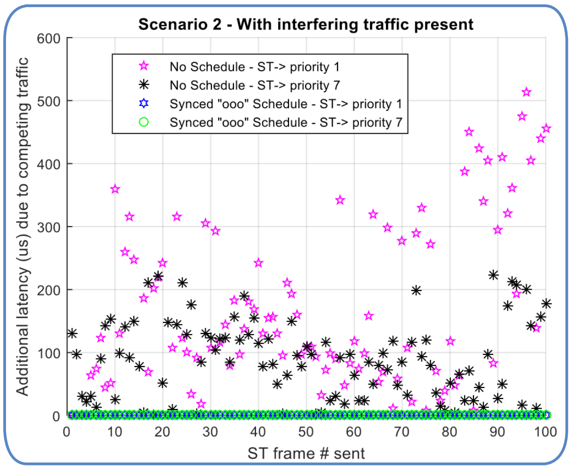

Figure 7: Impact of scheduling with interfering traffic

We also examined the jitter of the scheduled traffic at the source and the sink. Figure 8 and Figure 9 show the results we obtained with and without a Qbv schedule in the switches. Comparing the two figures, we observe that having the Qbv schedule in the switches limits the amount of jitter in the arrival time of the scheduled traffic frames at the sink to the amount of jitter in the transmission time of the scheduled traffic at its source.

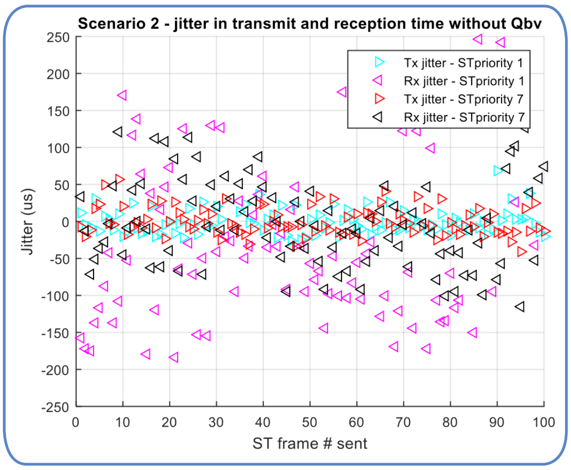

Figure 8: Rx jitter without Qbv

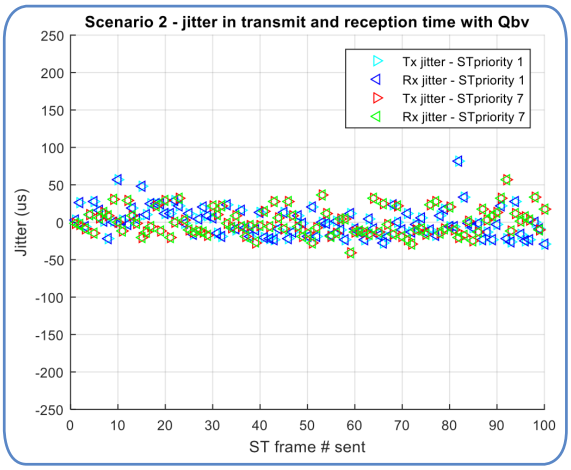

Figure 9: Rx jitter with Qbv

Scenario 3: What if you have a critical periodic traffic source that cannot be synchronized to your network?

Figure 10 and Figure 11 show the impact on the end-to-end latency results by a Qbv schedule that does not have the same time awareness as the source of the scheduled traffic. Figure 10 shows the results using an "ooo" schedule, while Figure 11 shows the results of a "CoC" schedule.

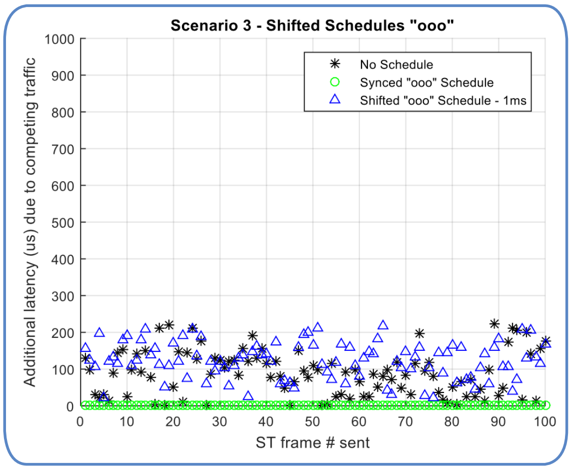

Figure 10: Schedule always open, impact schedule offset

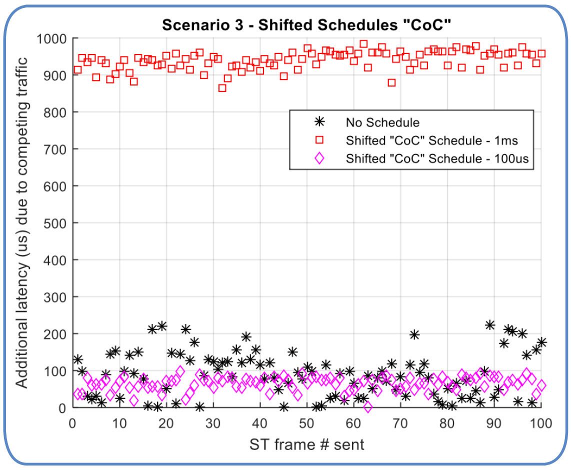

Figure 11: Schedule closed outside its scheduled time slot

Considering that the Ethernet switches are synchronized to the grandmaster, the Ethernet source sends its scheduled traffic frames ahead of its scheduled slot by a fixed time offset in this evaluation scenario. As can be observed in Figure 10, when the egress queue of the scheduled traffic is configured to be always open (as shown in Figure 5), the worst case added latency behavior of an unsynchronized schedule will be the same as the scenario with no Qbv schedule at all. When the egress queue of the scheduled traffic is configured to be closed outside of its scheduled time slots, we can see in Figure 11 that the added latency results then depend on the time offset of the scheduled traffic source. What can also be observed in Figure 11 is that the addition to the end-to-end latency for a 100 µs shift is more stable and has a lower worst case value than the situation without a Qbv schedule. This tells us that a specific configuration of the schedule can limit the worst case added latency. Based on the results above, we observe that:

- The worst case end-to-end latency is strongly dependent on the configured Qbv schedule, which means that the added latency for scheduled traffic over Ethernet switches is controllable by Qbv configuration.
- The worst case end-to-end latency with a Qbv schedule can be comparable to the worst case end-to-end latency without a Qbv schedule.
- The best case end-to-end latency with a synchronized Qbv schedule is comparable to the lowest achievable end-to-end latency over the network with no other traffic present, i.e., dedicated to the use of the scheduled traffic.
- The added latency behavior in general is as listed in Table 2. The table assumes the scheduled frame is the only one in its queue, and that other priority traffic is present.

Assigned Priority of the Scheduled Traffic	Max. Added Latency		
	No Qbv	Qbv, always open	Qbv, open only in slot
Highest priority, arrives within its slot	1 frame	None	None
Lower priority, arrives within its slot	N*1 frame	None	None
Highest priority, arrives outside its slot	1 frame	1 frame	1 cycle
Lower priority, arrives outside its slot	N*1 frame	1 cycle	1 cycle

N = number of frames that could be selected for transmission before the scheduled traffic frame due to higher priority

Table 2: Maximum added latency to a scheduled traffic frame per Ethernet switch

Scenario 4: What if you need to minimize the jitter at the reception of the scheduled traffic?

For some time-constrained applications, having the lowest achievable end-to-end latency may not be the main concern. Instead, the jitter may be the main application performance metric. The Qbv scheduling mechanism can also be used to minimize the jitter at the reception of the scheduled traffic.

In the results from scenario 2, we observed that the "ooo" schedule limits the amount of jitter at the sink to the jitter at the source. In this scenario, we evaluate how to use a shifted "CoC" schedule to minimize the jitter even further.

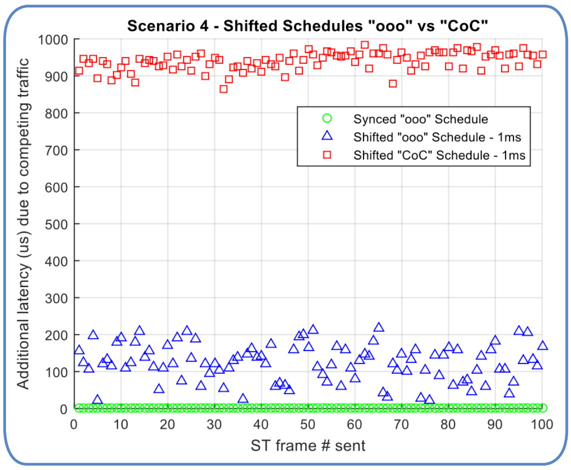

Figure 12: Impact on latency by shifted "CoC" schedule vs "ooo"

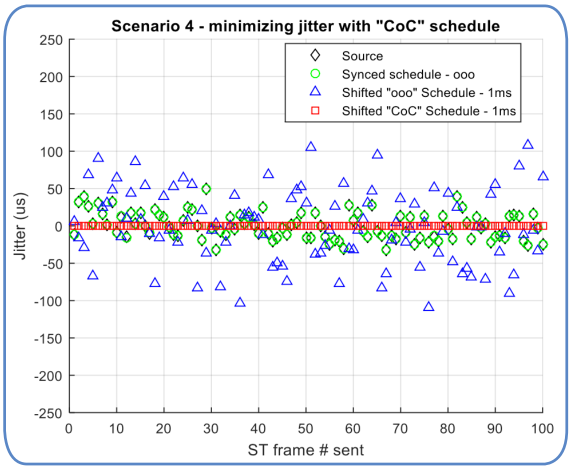

Figure 13: Impact on jitter by shifted "CoC" schedule vs "ooo"

Figure 12 shows us what we already know from scenario 3, where using a shifted "CoC" schedule increases the end-to-end latency depending on the offset of the shift. Figure 13 now shows us the impact of a shifted "CoC" schedule on the jitter. As can be observed from this figure, the jitter at the reception of the scheduled traffic is now minimized, even with regards to the jitter at the source.

4. Conclusion

In this whitepaper, we have explained the main concepts of IEEE 802.1Qbv and shown benefits and costs of using Qbv scheduling on the end-to-end latency and jitter performance of scheduled traffic for timing-constrained applications. Our experiments were done using NXP and other commercially available components. The evaluation consisted of comparing the end-to-end latency and jitter results for scenarios with and without a Qbv schedule, the presence of other criticality traffic, the priority assigned to the scheduled traffic, the synchronicity of the scheduled traffic source with the schedule on the Ethernet switches, and the configuration of the schedule.

Experiments show that with Qbv scheduling, it is possible to achieve the lowest possible latency through a network for periodically scheduled traffic frames, regardless of the priority of the frame and other traffic present. In the case of simple strict priority arbitration without Qbv scheduling, the end-to-end latency of a frame will be increased due to the presence of other traffic, with the increase being dependent on the assigned priority and timing of the competing traffic. Using Qbv scheduling is one solution to ensure lowest end-to-end latency for the traffic frames of a time-constrained application in a network with mixed criticality concurrent traffic.

To achieve this lowest possible latency, it is necessary that the scheduled slots in the Qbv are chosen large enough to take this jitter of the source into account. It is also necessary for the scheduled traffic source to be synchronized to the switches in the network, and that the switches support the Qbv mechanisms in hardware. The cost to the system also lies in the reduced bandwidth given to the other traffic in the network, when it is blocked for the duration of the scheduled slot.

We also show that it is possible to minimize the jitter of scheduled traffic on reception at the sink, at the cost of added latency. To minimize the jitter, the egress queue for the scheduled traffic in the switch will be closed outside the scheduled slots, and the open time slot in the schedule of the switch will have a time offset with the scheduled traffic source. The resulting added latency is dependent on the jitter at the source and the schedule offset in the switches to the scheduled traffic source. The cycle time of the schedule caps the additional end-to-end latency.

Anticipating future in-vehicle application requirements and subsequently evaluating possible solutions, as done in this whitepaper, is key to enable highly connected and autonomous vehicles in the very near future. As a leader in IVN and other automotive solutions, NXP can provide both product-based and technical support for manufacturers wanting to stay at the forefront of innovative in-vehicle networking.

5. References

[1] S.B. Carlson, J. Farkas, N. Finn, D. Pannell, et al, "IEEE 802 Ethernet Networks for Automotive", accessed 23 Aug 2018. [Online]. Available: http://www.ieee802.org/802_tutorials/2017-07/tutorial-Automotive-Ethernet-0717-v02.pdf

[2] "IEEE 802.1 Time-Sensitive Networking Task Group", accessed 23 Aug 2018. [Online]. Available: http://www.ieee802.org/1/pages/tsn.html

[3] "IEEE 802.1 Audio Video Bridging Task Group", accessed 23 Aug 2018. [Online]. Available: http://www.ieee802.org/1/pages/avbridges.html

[4] "IEEE Standard for Local and metropolitan area networks – Bridges and Bridged Networks. Amendment 25: Enhancements for Scheduled Traffic," IEEE Std 802.1Qbv-2015, Dec 2015.

[5] "IEEE Standard for Local and metropolitan area networks – Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks," IEEE 802.1AS-2011, Feb 2011.

[6] "iPerf – The TCP, UDP and SCTP network bandwidth measurement tool", accessed 23 Aug 2018. [Online]. Available: https://iperf.fr/en/

[7] NXP Semiconductors, "NXP MPC5604EKIT: Development Kit", accessed 23 Aug 2018. [Online]. Available: https://www.nxp.com/products/processors-and-microcontrollers/power-architecture-processors/mpc5xxx-55xx-32-bit-mcus/ultra-reliable-mpc56xx-32-bit-automotive-and-industrial-microcontrollers-mcus/nxp-mpc5604ekitdevelopment-kit-enabling-video-over-ethernet-with-nxp-mpc5604e-mcu:MPC5604EKIT

[8] "IEEE Standard for a Transport Protocol for Time-Sensitive Applications in Bridged Local Area Networks," IEEE Std 1722-2016, Dec 2016.

[9] "RIoTboard | element14", accessed 23 Aug 2018. [Online]. Available: https://www.element14.com/community/community/designcenter/single-board-computers/riotboard

[10] NXP Semiconductors, "OM14510-SJA1105TJP: Automotive Ethernet Switch Application Board", accessed 23 Aug 2018. [Online]. Available: https://www.nxp.com/products/analog/interfaces/in-vehicle-network/ethernet/automotive-ethernet-switches/automotive-ethernet-switch-application-board:OM14510-SJA1105TJP

How to Reach Us:

Home Page: **www.nxp.com**
Web Support: **www.nxp.com/support**

USA/Europe or Locations Not Listed:
NXP Semiconductors USA, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.nxp.com/support

Europe, Middle East, and Africa:
NXP Semiconductors Germany GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.nxp.com/support

Japan:
NXP Japan Ltd.
Yebisu Garden Place Tower 24F,
4-20-3, Ebisu, Shibuya-ku,
Tokyo 150-6024, Japan
0120 950 032 (Domestic Toll Free)
www.nxp.com/jp/support/

Asia/Pacific:
NXP Semiconductors Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@nxp.com

Document Number: QBVSOLUTIONSWPA4 REV 1