AVB - Generation 2
Latency Improvement Options

v2 – Nov, 2011

802.1 AVB Face to Face – Atlanta

Don Pannell - dpannell@marvell.com

We need to know the Requirements!

Start with the Automotive Requirements from previous Presentations by others

Automotive Network Requirements

From the public presentation:

[Revised] QoS requirements for Automotive Ethernet backbone systems

By:

Yong Kim (Broadcom)
Junichi Takeuchi (Renesas)
Masa Nakamura (Envital)

Sept 7, 2011

QoS requirements for automotive control data class

Performance requirements for automotive control data class

- Maximum latency: 100 us / 5 AVB hops
  - Guaranteed latency
  - Topology independent
  - Automotive control data class to have higher priority than SR classes
    - Maximum 2 priority classes (e.g. Control data class and SR class A)

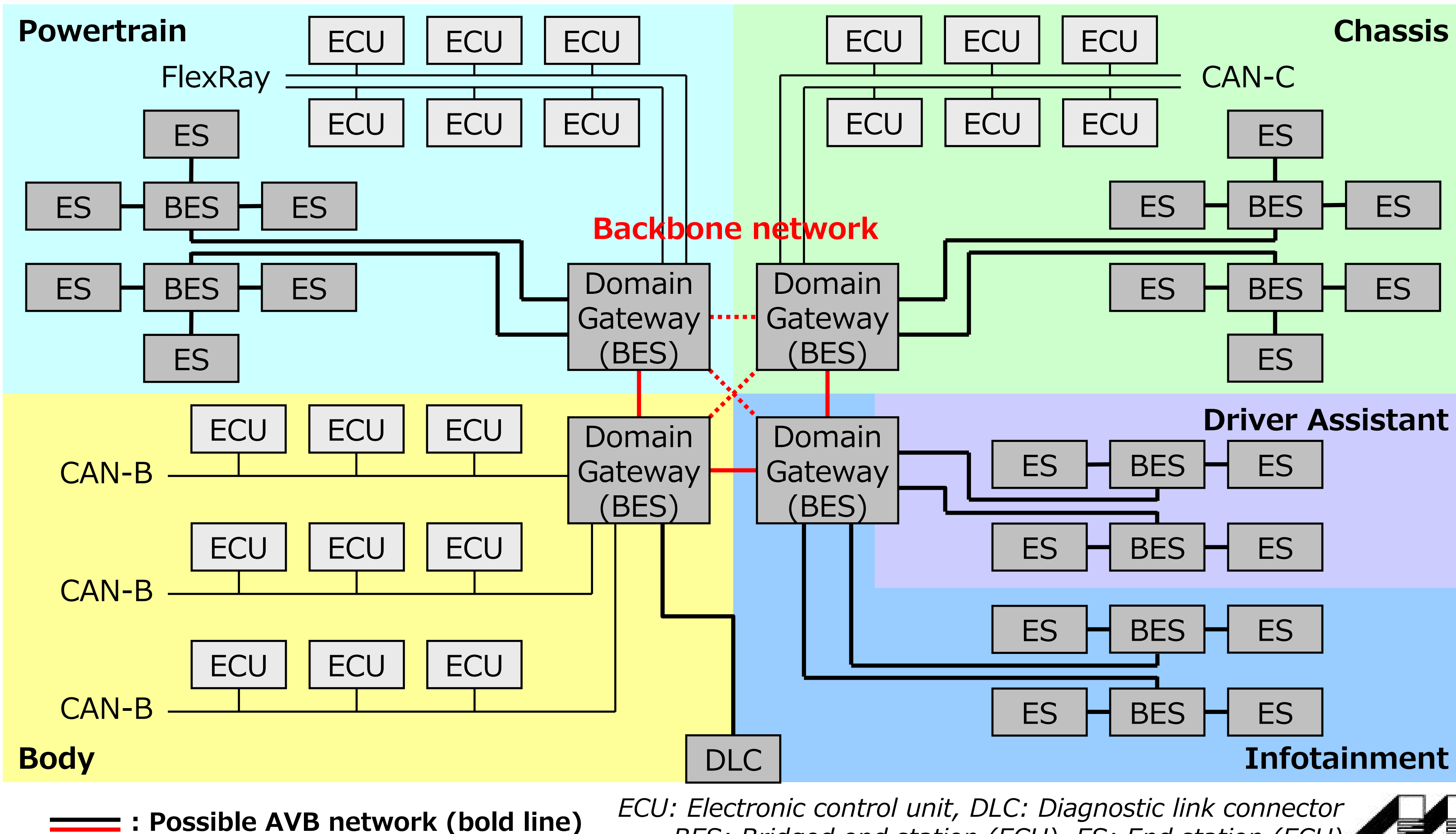
Preconditions for performance requirements

- Network type: Dedicated network in a vehicle
- Network attributes
  - Maximum AVB hop count: 7
  - Maximum number of nodes (bridged end station & end stations): 32
  - Maximum cable length: 24 m
  - Maximum end-to-end cable length: 30 m
- Automotive control data class attributes
  - Maximum data size (payload size): 128 bytes @FE ~ 256 bytes @GE
  - Maximum number of simultaneous transmission: 8 @FE ~ 32 @GE
  - Transmission period: 500 us
- Payload size for other/lower traffic classes: 256 bytes @FE ~ 1500 bytes @GE

These are our best estimates derived from multiple assumptions of the current and future automotive applications.

Example next-generation automotive network architecture

Ethernet AVB applied to automotive control data transmission between domain gateways and in powertrain/chassis domains

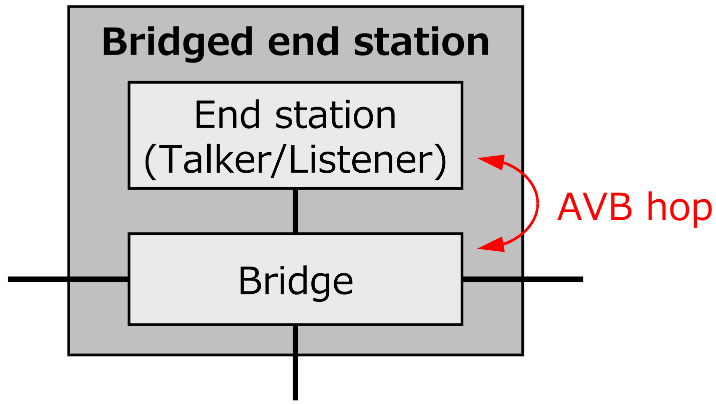


ECU: Electronic control unit, DLC: Diagnostic link connector
BES: Bridged end station (ECU), ES: End station (ECU)

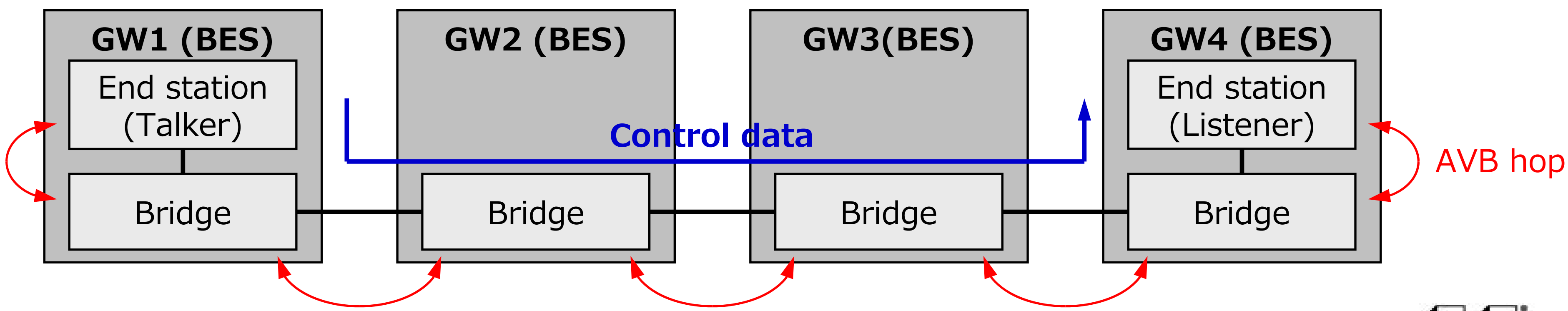
Consideration of AVB hop counts

Assumption for bridged end stations

- A bridged end station consists of a bridge and a end station (talker/listener).
- One AVB hop needs to be counted inside of bridged end stations.



Example control data transmission in a backbone network



AVB hop = 5

How can these Requirements be Met?

Specifically:
Less than 100 uSec Latency over 5 Hops

Goals Using Fast Ethernet

- **Latency less than 100 uSec over 5 Hops**
- **Control Data is 8 x 128 payload bytes every 500 uSec**
  - 128 payload bytes + 22 frame overhead = 150 byte frames + 20 byte gap
  - Time to Tx Control Frame = 13.600 uSec (170 bytes * 8 bits * 10ns)
  - 8 frames = 8 * 170 bytes or 1,360 byte times for the low latency data
  - Time to Tx 8 frames = 108.8 uSec (13.600 uSec * 8) FAIL!
  - We have a problem here as its already over the 100 uSec goal!
- **The next 2 slides show clarified requirements confirmed by Masa Nakamura that gives a solution to this problem:**
  - The 1st slide shows the problem by assuming all 8 flows start at the same time (what the above math assumes)
  - The 2nd slide shows how to solve the problem by staggering when the 8 flows show up at the Talker – something that can be done by the previous networks (both CAN and FlexRay) else this can't work!

Consideration for automotive QoS requirements

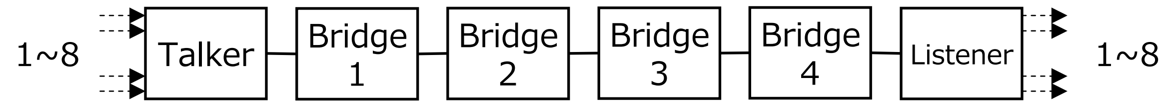
Case 1

●Preconditions

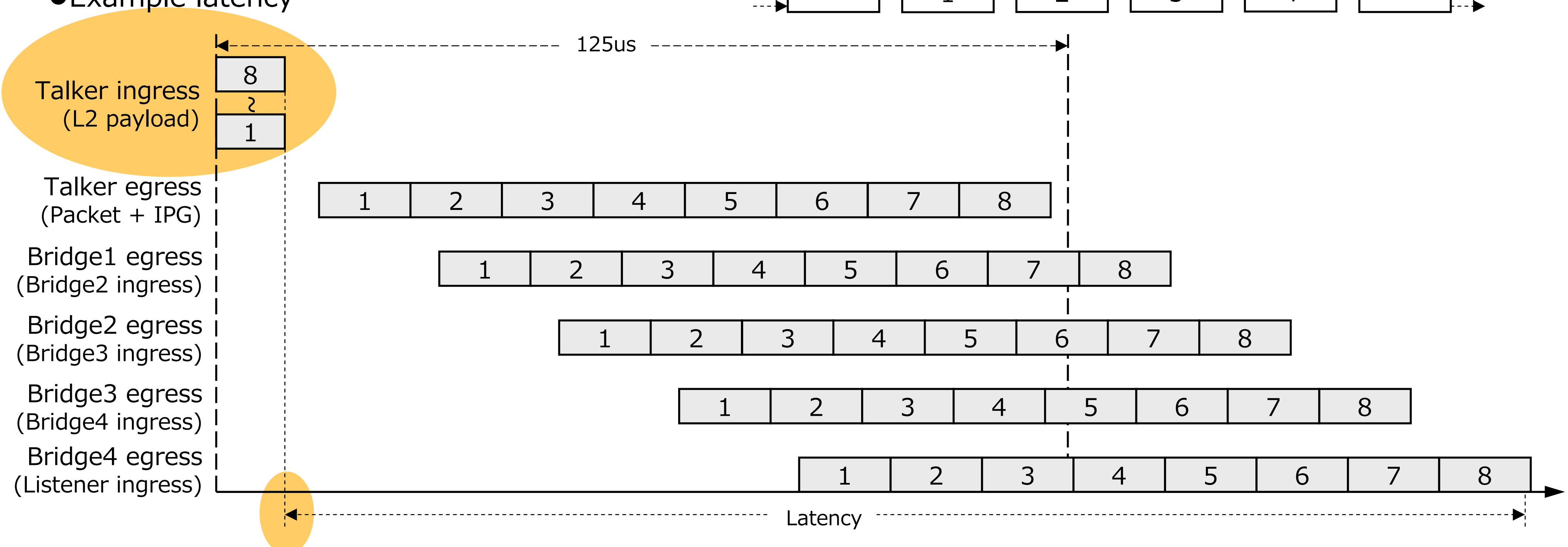
• Clock time for application software execution and network time for data transmission are not synchronized.

• The start time of data transmission is not coordinated.

⇒ Allowed simultaneous transmission



●Example latency



* Not included cable propagation delay

Consideration for automotive QoS requirements

Case 2

●Preconditions

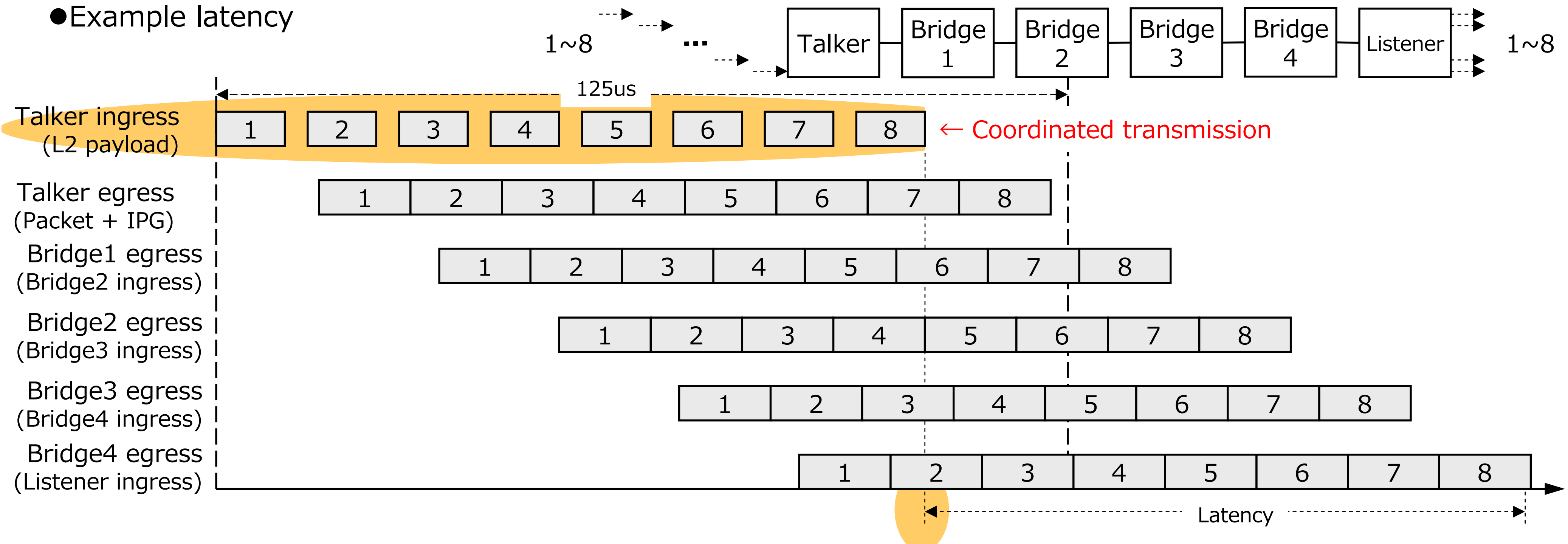
• All ECUs are synchronized.

• Clock time for application software execution and network time for data transmission are synchronized.

• The start time of data transmission could be coordinated to avoid overlapped (simultaneous) transmission.

⇒ Coordinated transmission

●Example latency



* Not included cable propagation delay

FE Latency w/Clarified Req. – No Interference

Latency of each 128 byte (of payload) Control Data Frame:

- Time to Tx one frame out of the Talker
- + Time to get the frame out of the 1st Bridge
- + Time to get the frame out of the 2nd Bridge
- + Time to get the frame out of the 3rd Bridge
- + Time to get the frame out of the 4th Bridge

At this point the frame is fully inside the Listener

- With the exception of cable delay which is insignificant & will be ignored
  - 100 meters of cable delay is ~538ns so 30 meters of cable is 162ns

Time to Tx frame out of the Talker =

- Time to Tx Control Frame = 12.000 uSec (150 bytes * 8 bits * 10ns)
  - 150 = 128 payload + 22 frame overhead
  - There is no IFG/Preamble overhead Assuming no interfering frame

FE Latency w/Clarified Req. – No Interference

- **Time to get the frame out of any of the Bridges (assuming Store & Forward Bridges):**
  - Time to receive the Frame (t_{RX})
    - –This is already included by the previous hop's Tx Time
  - Time to process the Frame (t_{BRIDGE})
    - –Data sheet spec – but will assume 2 x 512 bit times or 10.240 uSec in FE
  - Time to Tx the Frame (t_{TX})
    - –12.000 uSec assuming no interference
  - Total in this case is: 10.240 uSec + 12.000 uSec = 22.240 uSec
- **For 4 Bridge Hops this is: 4 x 22.240 uSec = 88.960 uSec**
- **Total with the Talker added is: 12.000 uSec + 88.960 uSec = 100.960 uSec – a bit over due to t_{BRIDGE}**
  - t_{BRIDGE} is the only controllable parameter in this example!

With No Interference, the FE Automotive Goals can be met (with zero margin) if the delay of each Bridge is 10.000 uSec or less

Specifically:
Less than 100 uSec Latency over 5 Hops
(with Coordinated Transmission into the Talker)

Goals Using Gig Ethernet

- **Latency less than 100 uSec over 5 Hops (no change)**
- **Control Data is 32 x 256 payload bytes every 500 uSec**
  - 256 payload bytes + 22 frame overhead = 278 byte frames + 20 byte gap
  - Time to Tx Control Frame = 2.384 uSec (298 bytes * 8 bits * 1ns)
  - 32 frames = 32 * 298 bytes or 9,536 byte times for the low latency data
  - Time to Tx 32 frames = 76.288 uSec (2.384 uSec * 32) PASS!
    - This is if all 32 frames arrive at the Talker at the same time
- **We will assume the 32 frames arrive staggered such that the Talker latency only needs to Tx one Frame**
  - Same assumption that was made in the FE case

GE Latency w/Clarified Req. – No Interference

- **Latency of each 256 byte (of payload) Control Data Frame:**
  - Time to Tx one frame out of the Talker
  - + Time to get the frame out of the 1st Bridge
  - + Time to get the frame out of the 2nd Bridge
  - + Time to get the frame out of the 3rd Bridge
  - + Time to get the frame out of the 4th Bridge
- **At this point the frame is fully inside the Listener**
  - With the exception of cable delay which is insignificant & will be ignored
    - 100 meters of cable delay is ~538ns so 30 meters of cable is 162ns
- **Time to Tx frame out of the Talker =**
  - Time to Tx Control Frame = 2.224 uSec (278 bytes * 8 bits * 1ns)
    - 278 = 256 payload + 22 frame overhead
    - There is no IFG/Preamble overhead Assuming no interfering frame

GE Latency w/Clarified Req. – No Interference

- **Time to get the frame out of any of the Bridges (assuming Store & Forward Bridges):**
  - Time to receive the Frame (t_{RX})
    - This is already included by the previous hop's Tx Time
  - Time to process the Frame (t_{BRIDGE})
    - Data sheet spec – but will assume 2 x 512 bit times or 1.024 uSec in GE
  - Time to Tx the Frame (t_{TX})
    - 2.224 uSec assuming no interference
  - Total in this case is: 1.024 uSec + 2.224 uSec = 3.248 uSec
- **For 4 Bridge Hops this is: 4 x 3.248 uSec = 12.992 uSec**
- **Total with the Talker added is: 2.224 uSec + 12.992 uSec = 15.216 uSec PASS! With lots of margin**
  - t_{BRIDGE} is the only controllable parameter in this example!

With <u>No</u> Interference, the GE Automotive Goals can be met (with >84 uSec margin) if the delay of each Bridge is 1.024 uSec or less

Specifically:
Less than 100 uSec Latency over 5 Hops
(with Coordinated Transmission into the Talker – without Coordinated Transmission into the Talker the margin is reduced to <10 uSec)

Now add in the Interfering frames per the Automotive Requirements

Specifically:
Less than 100 uSec Latency over 5 Hops
(with Coordinated Transmission into the Talker)
with a Payload size for other/lower traffic classes:
256 bytes @ FE ~ 1500 bytes @ GE

FE Latency w/Clarified Req. – w/Interference

- FAIL!
- There is zero margin left in the FE case!
- If any hop (Talker or Bridge) cannot start right away with the Control Data frame due to needing to finish even 1 byte of an interfering frame, the 5 hop latency will be greater than the 100 uSec requirement!

GE Latency w/Clarified Req. – w/Interference

- **Latency of each 256 byte (of payload) Control Data Frame:**
  - Time to Tx one frame out of the Talker (after a Max interfering frame)
  - + Time to get the frame out of the 1st Bridge (after a Max interfering frame)
  - + Time to get the frame out of the 2nd Bridge (after a Max interfering frame)
  - + Time to get the frame out of the 3rd Bridge (after a Max interfering frame)
  - + Time to get the frame out of the 4th Bridge (after a Max interfering frame)
- **At this point the frame is fully inside the Listener**
- **Time to Tx frame out of the Talker (after Max interfering) =**
  - Time to Tx Max Frame + IFG + Preamble
    - 1500 + 22 + 20 = 1542 bytes * 8 bits * 1ns = 12.336 uSec
  - +Time to Tx Control Frame = 2.224 uSec (278 bytes * 8 bits * 1ns)
    - 278 = 256 payload + 22 frame overhead
  - = 12.336 uSec + 2.224 uSec = 14.560 uSec

MARVELL

GE Latency w/Clarified Req. – w/Interference

- **Time to get the frame out of any of the Bridges (assuming Store & Forward Bridges):**
  - Time to Tx the Max size Interfering Frame ($t_{TX\text{-}INTERFERING}$)
    - 1500 + 22 + 20 = 1542 bytes * 8 bits * 1ns = 12.336 uSec
  - +Time to Tx the Control Frame (t_{TX})
    - 2.224 uSec assuming no interference
  - Total in this case is: 12.336 uSec + 2.224 uSec = 14.560 uSec
  - t_{BRIDGE} is not used here as $t_{TX\text{-}INTERFERING}$ is the larger of the two numbers
- **For 4 Bridge Hops this is: 4 x 14.560 uSec = 58.240 uSec**
- **Total with the Talker added is: 14.560 uSec + 58.240 uSec = 72.800 uSec PASS! With lots of margin**
- **And this uses Today's STD! GE w/No Changes!**

With Any Interference, the FE Automotive Goals cannot be met

Specifically:
Less than 100 uSec Latency over 5 Hops
(with Coordinated Transmission into the Talker)
with a Payload size for other/lower traffic classes of 256 bytes

With Interference, the GE Automotive Goals can be met without any new IEEE STD! needed!

Specifically:
Less than 100 uSec Latency over 5 Hops
(with Coordinated Transmission into the Talker such that the 32 Control frames are transmitted back-to-back out of the Talker – doable with >27 uSec of margin)

Can the FE case be Fixed?

Proposals on the Table

- Pre-emption
- Time Aware Shapers
- Change the Requirements?

Pre-emption

Pre-emption as Defined for Automotive

- **Lower Priority Frames can be Pre-empted but each Framelet down the wire must be a good Frame**
  - Pre-emption to occur on a Link-by-Link basis
  - Re-constructed Frame must be 100% as it was before pre-emption
  - Each Framelet will have a good CRC
  - Each Framelet must be at least 64 bytes in size
  - Pre-empt on Modulo boundaries (easier for the receiver) – assume 64 bytes for now – is that 64 bytes on the wire or 64 bytes of the re-constructed Frame (which would be more on the wire due to the 'Tag')?
  - Pre-emptable Frames will need extra Tag bytes added
- **With these Assumptions What is the Added Latency due to an Interfering Frame that is Pre-empted?**

Pre-emption as Defined for Automotive

- **The Framelet needing to be a Good Frame of 64 Bytes is the Driving Factor**
  - A 127 byte Frame cannot be Pre-empted due to the remainder being too small – unless padding is done
    - Padding will modify the re-constructed Frame unless the original Frame size is transferred to the receiver in the 'Tag'
    - How much padding is OK? Up to 63 bytes for a 65 byte frame?
- **So the Interfering Delay of a Pre-emptable Frame is somewhere between 64 Bytes and 128 Bytes**
  - To be determined
- **Lets Assume it's the best case of 64 Bytes**
  - But remember, worst case is more likely 128 bytes
  - Fixed break points (modulo boundaries) add to this issue

FE Latency w/Clarified Req. – w/Pre-emption

- **Latency of each 128 byte (of payload) Control Data Frame:**
  - Time to Tx one frame out of the Talker (after a 64B pre-empted frame)
  - + Time to get the frame out of the 1st Bridge (after a 64B pre-empted frame)
  - + Time to get the frame out of the 2nd Bridge (after a 64B pre-empted frame)
  - + Time to get the frame out of the 3rd Bridge (after a 64B pre-empted frame)
  - + Time to get the frame out of the 4th Bridge (after a 64B pre-empted frame)
- **At this point the frame is fully inside the Listener**
- **Time to Tx frame out of the Talker (after 64B interfering) =**
  - Time to Tx Framelet + (IFG + Preamble)
    - 64 + 20 = 84 bytes * 8 bits * 10ns = 6.720 uSec
  - +Time to Tx Control Frame = 12.000 uSec (150 bytes * 8 bits * 10ns)
    - 150 = 128 payload + 22 frame overhead
  - = 6.720 uSec + 12.000 uSec = 18.720 uSec

FE Latency w/Clarified Req. – w/Pre-emption

- **Time to get the frame out of any of the Bridges (assuming Store & Forward Bridges):**
  - Time to process the Frame (t_{BRIDGE})
    - Data sheet spec – but will assume ~2 x 512 bit times or 10.000 uSec in FE
  - +Time to Tx the 'rest' of the Interfering Framelet ($t_{TX\text{-}FRAMELET}$)
    - 64 + 20 = 84 bytes * 8 bits * 10ns = 6.720 uSec
  - +Time to Tx the Frame (t_{TX})
    - 12.000 uSec
  - Total in this case is: 10.000 uSec+6.720 uSec+12.000 uSec = 28.720 uSec
  - t_{BRIDGE} is used here as $t_{TX\text{-}FRAMELE}$ cannot start until the frame is ready to Tx
- **For 4 Bridge Hops this is: 4 x 28.720 uSec = 114.880 uSec**
- **Total with the Talker added is: 18.720 uSec + 114.880 uSec = 133.600 uSec FAIL! By the 5 x Framelets (5x6.720 uSec)**

FE Latency w/Clarified Req. – w/Pre-emption

- **Assume GE Performance Bridges are Used with FE Links**
  - Time to process the Frame (t_{BRIDGE})
    - Data sheet spec – but will assume ~2 x 512 GE bit times or 1.024 uSec
  - +Time to Tx the 'rest' of the Interfering Framelet ($t_{TX-FRAMELET}$)
    - 64 + 20 = 84 bytes * 8 bits * 10ns = 6.720 uSec
  - +Time to Tx the Frame (t_{TX})
    - 12.000 uSec
  - Total in this case is: 1.024 uSec+6.720 uSec+12.000 uSec = 19.744 uSec
  - t_{BRIDGE} is used here as $t_{TX-FRAMELE}$ cannot start until the frame is ready to Tx
- **For 4 Bridge Hops this is: 4 x 19.744 uSec = 78.976 uSec**
- **Total with the Talker added is: 18.720 uSec + 78.976 uSec = 97.696 uSec - <u>But this assumes 64 byte Interfering frames where 128 bytes is more likely the worst case</u>**

FE Latency w/Clarified Req. – w/Pre-emption

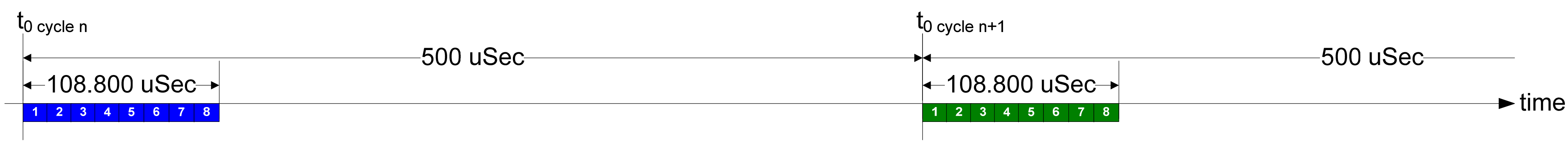
- **Assume GE Performance Bridges are Used with FE Links with more realistic 128 byte Interfering frames**
  - Time to process the Frame (t_{BRIDGE})
    - Data sheet spec – but will assume ~2 x 512 GE bit times or 1.024 uSec
  - +Time to Tx the 'rest' of the Interfering Framelet ($t_{TX\text{-}FRAMELET}$)
    - 128 + 20 = 148 bytes * 8 bits * 10ns = 11.840 uSec
  - +Time to Tx the Frame (t_{TX})
    - 12.000 uSec
  - Total in this case is: 1.024 uSec+11.840 uSec+12.000 uSec = 24.864 uSec
  - t_{BRIDGE} is used here as $t_{TX\text{-}FRAMELE}$ cannot start until the frame is ready to Tx
- **For 4 Bridge Hops this is: 4 x 24.864 uSec = 99.456 uSec**
- **Total with the Talker added is: 23.840 uSec + 99.456 uSec = 123.296 uSec FAIL! – with realistic Interference**

Time Aware Shapers

1st Need to Look at the Source Streams

Requirements state:

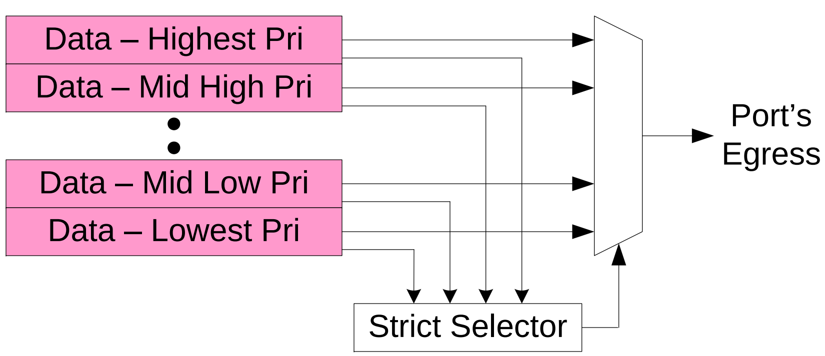
- 8 x 128 data bytes at FE every 500 uSec – or –
  - Is (128 + 22 + 20) x 8 = 108.800 uSec of ultra low latency data every 500 uSec
- 32 x 256 data bytes at GE every 500 uSec
  - Is (256 + 22 + 20) x 32 = 76.288 uSec of ultra low latency data every 500 uSec
- Assuming the previous networks (CAN and FlexRay) can schedule their ultra low latency data (already shown this needs to be done) then the preferred scheduling is:



- This gives 391 uSec for non-ultra low latency data every 500 uSec
- Every low latency network I have seen schedules its data transmission at regular intervals as these technologies do

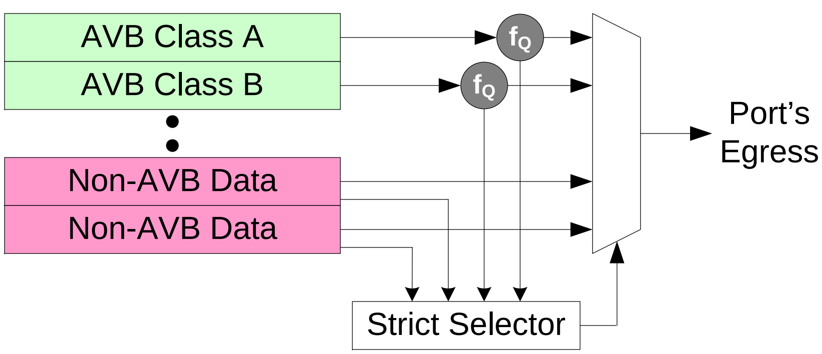
2nd Look at Typical Egress Queues

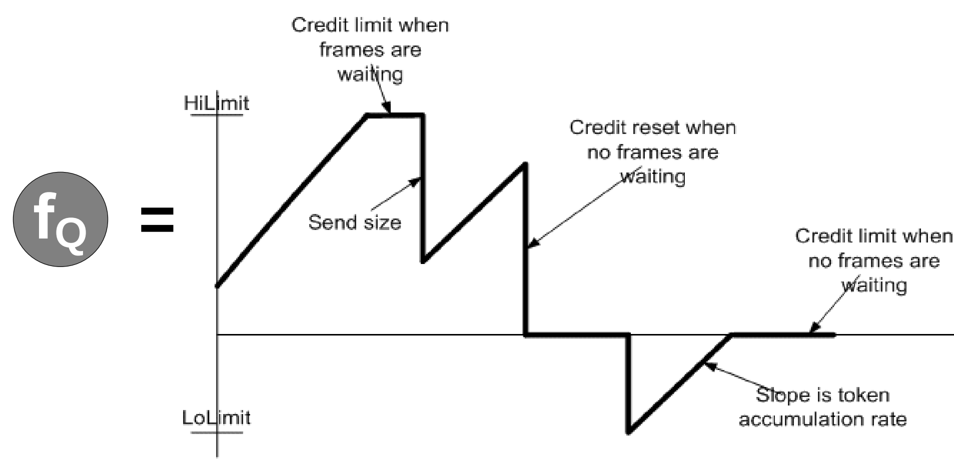
- **IEEE Standards before AVB Standardized two Possible Shapers**
  - Strict (as shown)
    - Transmit all Highest Pri frames until the queue is empty
    - Then transmit the next highest priority queue until it is empty, etc.
  - Or, Proprietary
- **Strict gets a Signal that a Queue has Data or not and the highest Queue with Data is selected**



3rd Look at what AVB's Qav Added

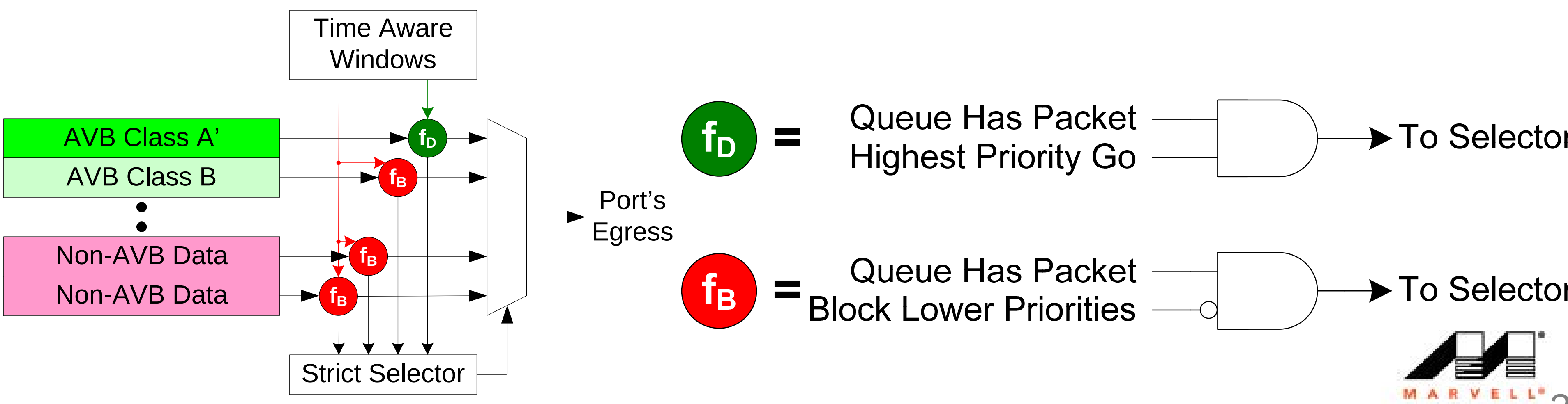
- IEEE 802.1 Qav added a 3rd Standardized Shaper
- The AVB Queues are Typically the top two Queues with a Strict Scheduler
- The AVB Queues add Qav Shapers that Selectively prevents the Strict Selector from 'seeing' Frames that are in the AVB queues.
- This spreads out Bursts of AVB data most of the time





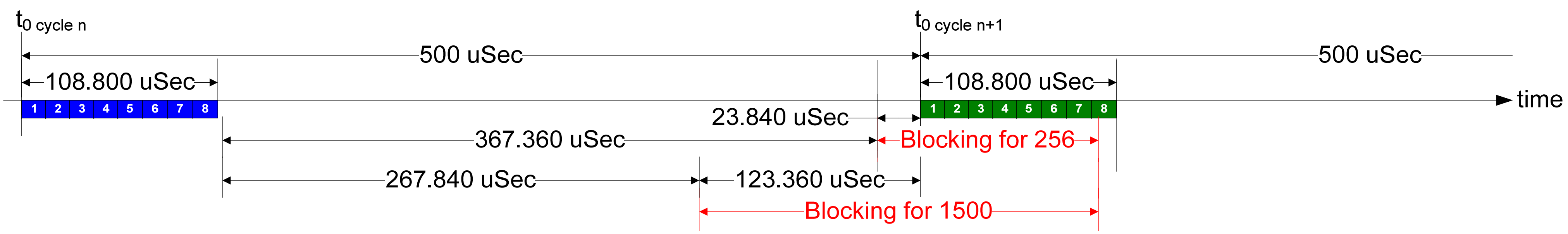
Time Aware Blocking Shapers - TABS

- **Time Aware Blocking Shapers are quite simple**
  - Just an AND gate as shown below
- **They Can work in Parallel with the Qav Shapers**
- **Most of the Logic is generating the Time Aware Windows**
  - One signal is the De-Blocking signal or ‘Highest Priority Goes’
  - The other is an Inverted Blocking signal or ‘Lower Priorities OK’



Usage of the Time Aware Windows

- **Two Blocking Windows are show Below**
  - The 23.840 uSec before t_0 window is for the stated requirement of 256 byte Other traffic Class data
  - The 123.360 uSec before t_0 window is for a supportable1500 byte Other traffic Class data
- **With 500 uSec Cycles there is plenty of time for Other traffic Class data – Even 1522 byte frames**



Advantages of TABS (Time Aware Blocking Shapers)

- AVB Bridges are Already Time Aware with 802.1AS
- All that is needed is to add a Global Time Window circuit and a Standard way to configure them (via MIBs)
- The actual Shaper is then simply a gate or two of logic per queue
- The Ingress & Egress 802.3 MACs do not need to change
- The Queuing System does not need to change
- They work together with the existing 802.1Qav Shapers
- They produce the lowest possible latency as there is no interference from the lower priorities

FE Latency w/Clarified Req. – w/TABS

- **Latency of each 128 byte (of payload) Control Data Frame:**
  - Time to Tx one frame out of the Talker (there is not interfering frame)
  - + Time to get the frame out of the 1st Bridge (there is not interfering frame)
  - + Time to get the frame out of the 2nd Bridge (there is not interfering frame)
  - + Time to get the frame out of the 3rd Bridge (there is not interfering frame)
  - + Time to get the frame out of the 4th Bridge (there is not interfering frame)
- **At this point the frame is fully inside the Listener**
- **Time to Tx frame out of the Talker =**
  - Time to Tx Control Frame = 12.000 uSec (150 bytes * 8 bits * 10ns)
    - 150 = 128 payload + 22 frame overhead
  - There is no interfering frame

FE Latency w/Clarified Req. – w/TABS

- **Time to get the frame out of any of the Bridges (assuming Store & Forward Bridges):**
  - Time to process the Frame (t_{BRIDGE})
    - Data sheet spec – but will assume ~2 x 512 bit times or 10.000 uSec in FE
  - +Time to Tx the Frame (t_{TX})
    - 12.000 uSec
  - Total in this case is: 10.000 uSec + 12.000 uSec = 22.000 uSec
- **For 4 Bridge Hops this is: 4 x 22.000 uSec = 88.000 uSec**
- **Total with the Talker added is: 12.000 uSec + 88.000 uSec = 100.000 uSec PASS!**
- **This is the same as the Non-Interfering case as expected**

Other Possible Issues

- **What About Data Going in the Other Direction?**
- **Since the paths are Full Duplex, the opposite data direction is not different**
  - True for both Pre-emption and Time Aware Blocking Shapers

- **What about Two or More Sources to One Destination?**
- **The worst case latency will increase unless the data slots are scheduled to not overlap**
  - The Data into each End Station Source was assumed to have been timed already for the FE case to work as required so this is just an extension
  - This is true for Time Aware Blocking Shapers
  - For Pre-emption this is much harder to do as the Interference due to the Framelets is unpredictable

Summary for the Automotive Example

- **Pre-emption with FE Link Speeds**
  - 123.296 to 133.600 uSec FAIL!
- **Time Aware Shapers with FE Bridges & FE Links**
  - 100.000 uSec PASS!
- **Today's GE (no pre-emption nor Time Aware Shapers)**
  - 72.800 uSec PASS!
- **Time Aware Shapers with GE Link Speeds**
  - 15.216 uSec PASS!
- **Change the Requirements?**
  - Don't have to that!
- **There are many solutions for this Automotive problem, but Pre-emption is not one of them**

Summary Of Time Aware Shapers

- **Time Aware Shapers are easy to add**
  - They are just another Shaper to add to the option list
- **Time Aware Shapers are consistent with 802 architecture**
  - They are added in the same place as previous shapers
- **Time Aware Shapers produce the lowest possible latency**
  - By definition there are no interfering frames
- **Requires Time Aware transmission of low latency frames**
  - All AVB Talkers are Time Aware already
  - Consistent with previous low latency network architectures
- **Time Aware Shapers w/ GE Links gets the lowest latency**

We can get very low latency if we Engineer it right

Questions?

We can get very low latency if we Engineer it right

Thank You