# AVB Gen 2: the Next Step

Michael Johas Teener

Plumbing Architect and Sr. Technical Director, Broadcom Corp, mikejt@broadcom.com

# Agenda

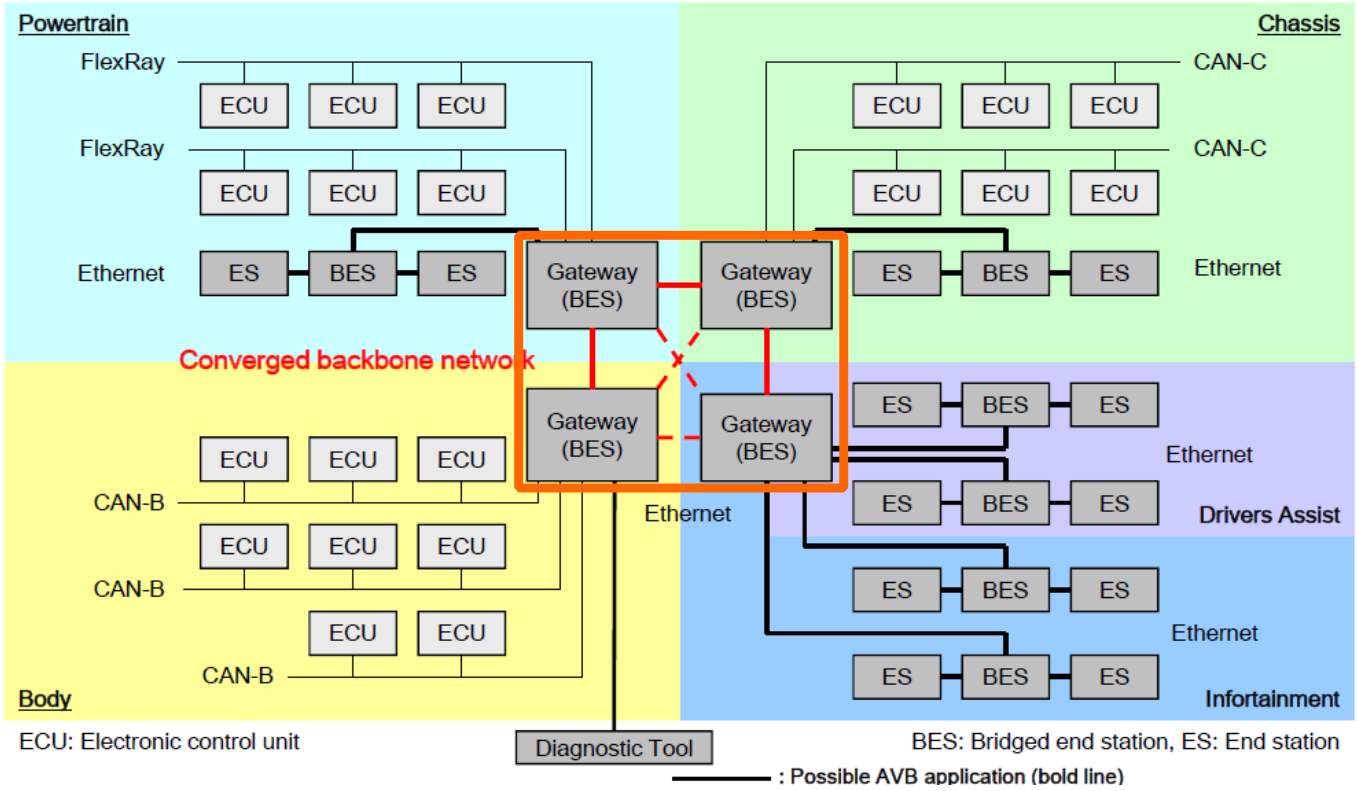
- Requirements for automotive control networks
  - … other markets, too
- Existing tools (AVB Gen 1)
  - and Ethernet 101
- New standards (AVB Gen 2)
  - Scheduled queues
  - Packet preemption
  - Multipathing and redundancy
  - Time synch improvements
- Timeline for standardization and products

Some of this presentation was derived from contributions to the 802.1 public document area. Many thanks to Markus Jochim, Christian Boiger, Norm Finn, Franz-Josef Goetz, Yong Kim, and Don Pannell

# Automotive control network requirements

- Small physical size, many ports, many different data requirements
  - 30 m, 5 hops, perhaps 100 devices
  - Control, sensors, driver assist video, radar, entertainment A/V
  - Gateways to CAN, Flexray, MOST, etc.
- Deterministic and very small delays
  - < 100 µs through 5 hops using 100 Mbit/sec PHY: 20 µs/hop
- Auto-spec reliability
  - Lighter/rugged/low cost cables
  - Difficult electrical/physical environment
- Safety!
  - Redundant paths / fault detection / securityd

# Example Auto Network

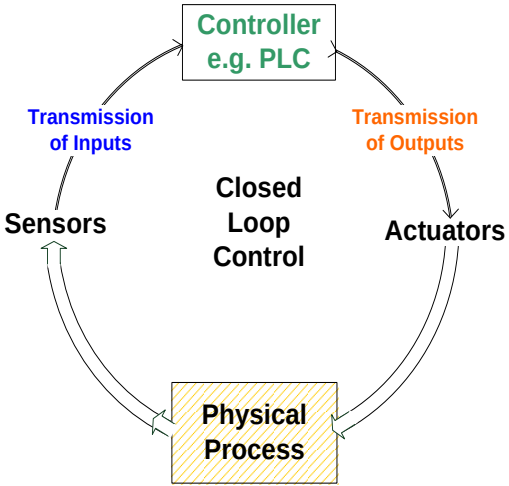


# Other markets / uses for gen 2

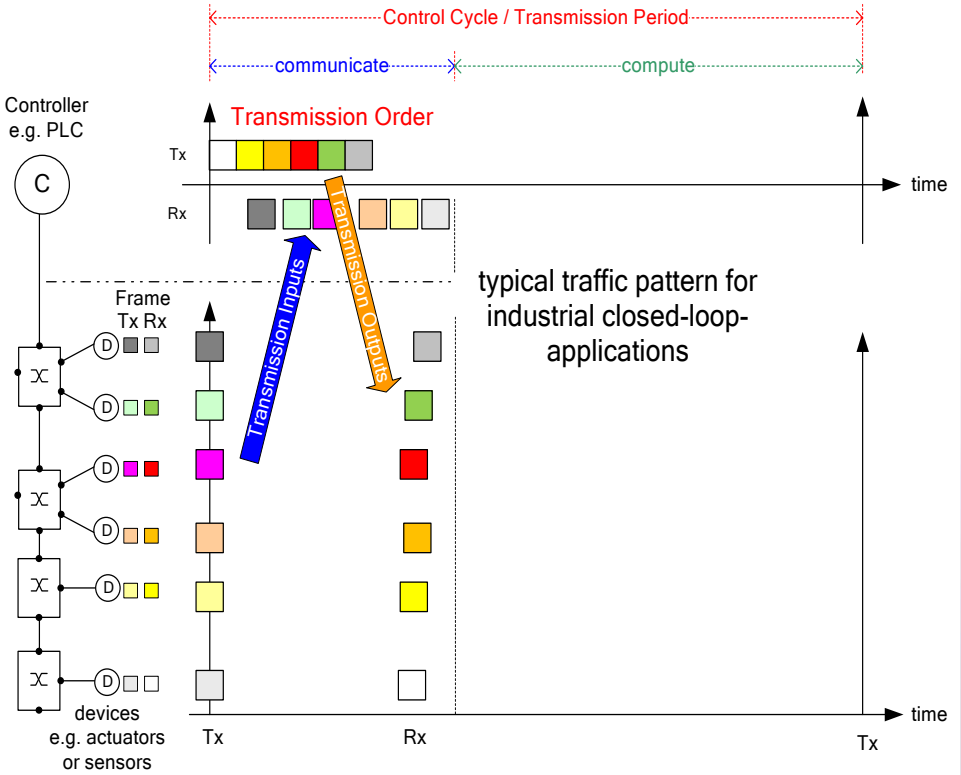
- Industrial control
  - Factory and robot networks
    - Siemens, Rockwell, Hirschmann, NI
- Professional A/V
  - Live performance / large venues
    - Harman, Bosch, Meyer Sound
- Consumer A/V
  - Multiple networks in the home, frequently unreliable
    - Broadcom, Intel, etc.

# Industrial / automotive control loop architecture

**The application model for control traffic is different**




***Typical traffic pattern for control traffic***



# Implications of control loop systems

- Sensor and control messages are scheduled
  - Well known timing, network-wide
  - “Cycle time” ranges from 30 µs to 10 ms, but commonly 125 µs
- Control messages are relatively short
  - Typically 128-256 byte
- Sensor bandwidth highly variable
  - Existing applications are low bandwidth, similar to control
  - Newer machine vision applications require high bandwidth in the local environment
    - Uncompressed/minimally compressed video from camera to “recognition” system

# Existing tools

- Ethernet 101:
  - It's a switched point-to-point network: packets only go where they need to go (CSMA/CD is dead)
    - Switches route traffic only where it needs to go
    - Each link only carries traffic that must travel on that link
    - Endpoint link speed (hence, cost) can be optimized for a particular device … 10M, 100M, 1G, 10G, 40G, 100G …
  - Switches and endpoints are really smart
    - Priorities, security, virtual LANs, cable diagnostics
- So we are well on our way, plus …

# AVB now

## AVB provides some of the solution

- 802.1AS / 1588 provides ±500 ns synchronization

  … but almost 1 sec switchover to new grand master clock

- Stream reservation plus credit-based shaper provides 2ms delays

  … but only for 7 hops and with no significant improvement for higher bandwidth links

- **But most importantly: an architecture for managed traffic**
  - A standards-based way to manage different classes of time-sensitive traffic
  - Control and monitoring via protocols running on end-point devices (vs requiring a centralized network manager)

# AVB does not mean complex!

- A full plug-and-play implementation implies a full stack
  - But note that some vendors do all that within the switch or endpoint IC itself (internal CPU+ROM)
- Well-know configurations can be pre-configured for instant-on
  - If configuration never changes, only minimal recovery parts of the AVB stack is needed
  - Hybrid engineered/plug-and-play architectures also work to support “minimal platform plus options” model.

# Moving to AVB gen 2

- Scheduled queues
- Packet preemption

} Reducing worst case delays

- Multipathing and redundancy
  - Faster recovery on link failure
  - Aggregated links
- Simpler/faster time synch recovery

# Limits on Delay

***The fundamental problem is interfering traffic***

- If a packet has just started being transmitted on a particular egress port, then all traffic, regardless of the priority, must wait until the egress port has completed transmitting that packet

# What is the best we could do?

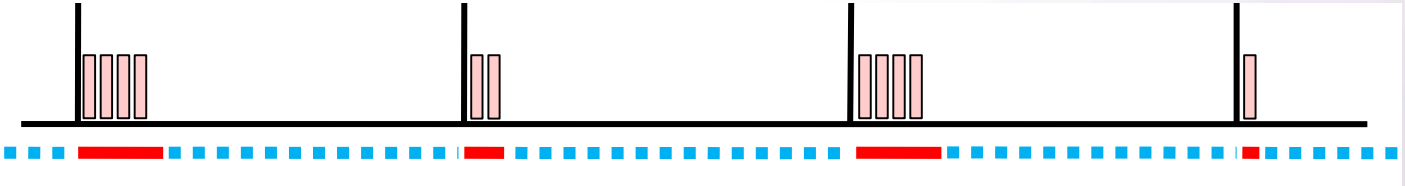
- Assume no interfering traffic
  - 100Mbit switch could have delays as low as 36 µs
    - GigE about 4 µs
- Add cut-through switching
  - 100Mbit switch could have delays as low as 16 µs!
    - GigE about 2 µs!
  - Note that cut-through normally does not help if there is interfering traffic, but in this case we assume no interference
- So, for a 5 hop / 100Mbit network …
  - We would have 80 µs delays …

Not bad!

# Avoiding interfering traffic

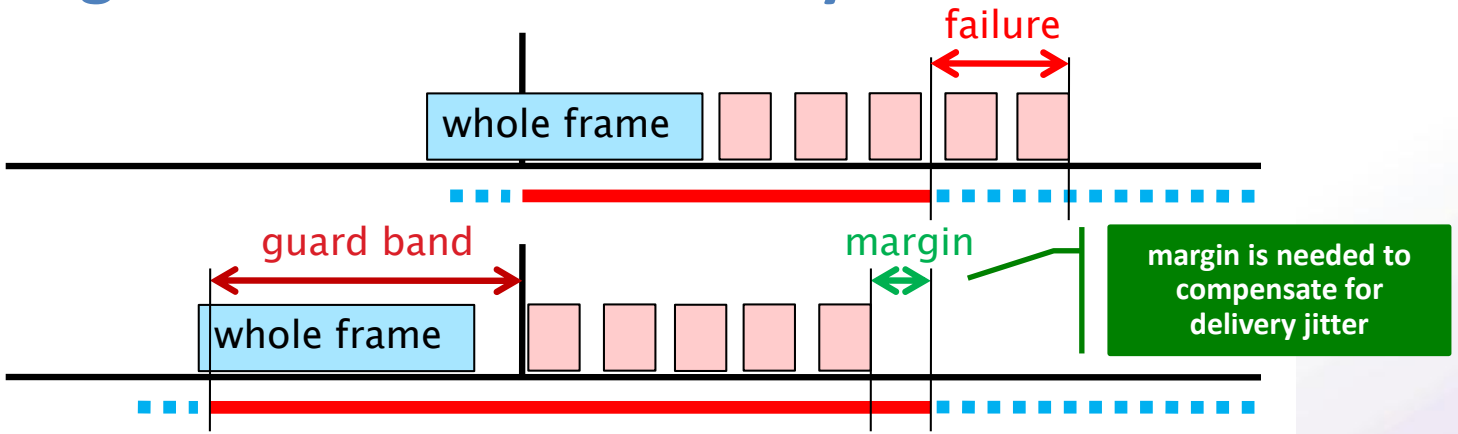
## The “Time Aware” shaper

- Make switches end points aware of the cycle time for control traffic
  - Block non-control traffic during particular windows of time to ensure that the egress port for a control stream is idle when the control traffic is expected
  - Each egress port could have a separate schedule
- Non-trivial calculation in non-trivial networks
  - Requires a fully managed network
  - This is a well understood, but difficult problem, currently implemented in proprietary networks such as Siemens’ “Profinet”



# Time aware shaper issues

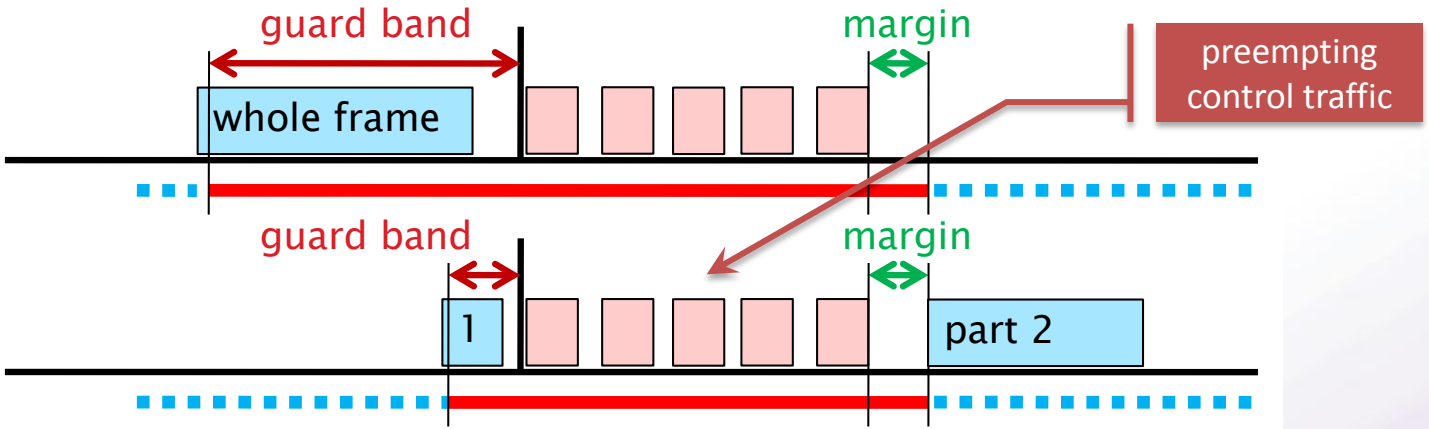
## A “guard band” is necessary




- If an interfering frame starts transmission just before the start of a reserved time period, it can extend critical transmissions outside the window.
- Therefore, a guard band is required before the window starts, equal in size to the largest possible interfering frame.

# Reducing the guard band

## Preemption is a good solution …




- If preemption is used, the guard band need only be as large as the largest possible interfering fragment, instead of the largest possible interfering frame.
- It is easy to see that the smaller the size of the time-reserved windows, the larger the impact of preemption.

# Efficiency of time-aware shaper

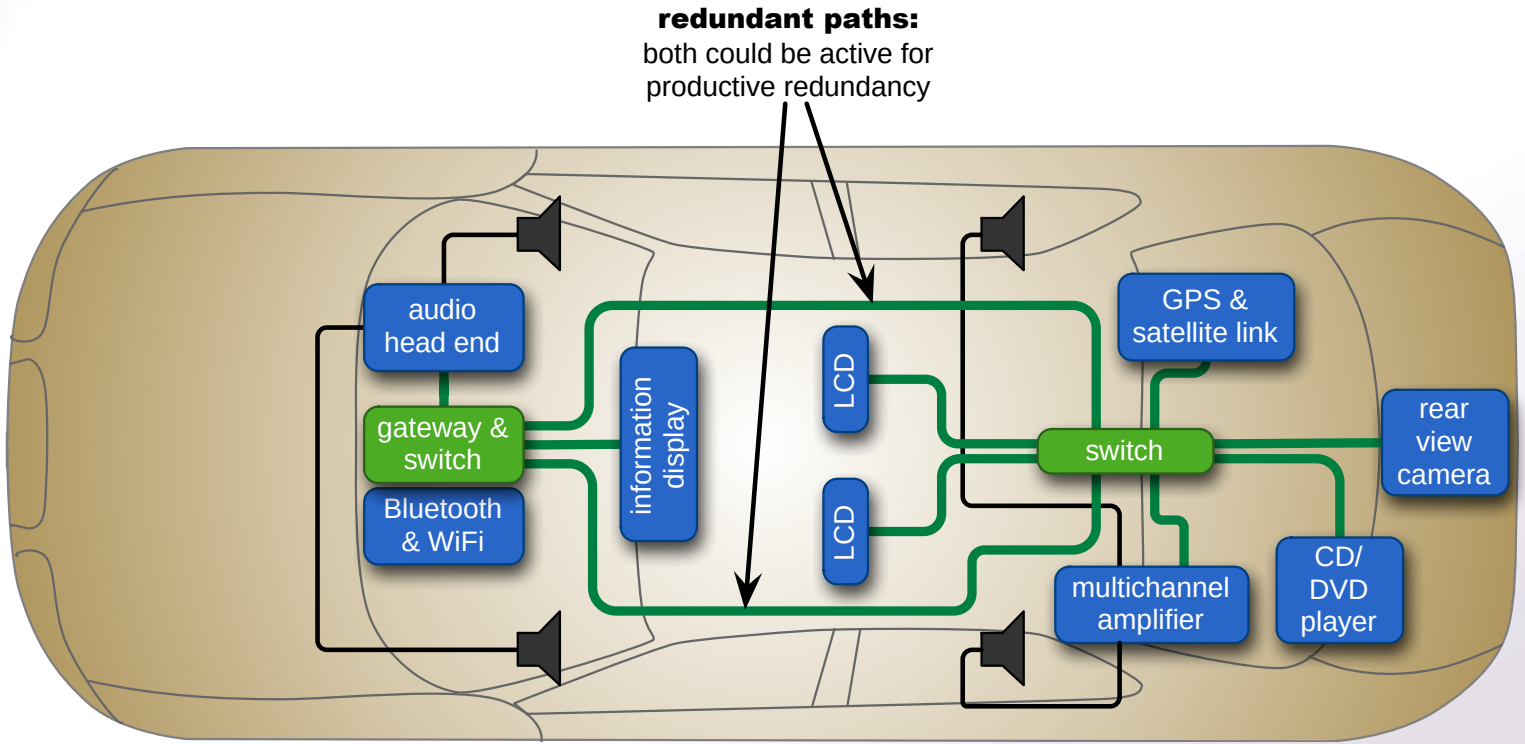
Assume the control traffic consists of a burst of four 128 byte packets and half the window is needed to compensate for delivery jitter

| Control traffic (as a percent of link bandwidth) | Overhead (margin, preamble, interframe gap) | With preemption | Without preemption |
|---|---|---|---|
| 0.1% | 0.2% | 0.3% | 0.5% |
| 1.0% | 2.3% | 2.6% | 5.3% |
| 10% | 23% | 26% | 53% |
| 30% | 69% | 78% | 160% |

# Preemption can be used alone

- Even without the time-aware shaper, worst-case delays for control traffic are reduced
  - For a single control stream, delay could be reduced over 100 µs per 100 Mbit/sec link
    - Over 10 µs per GigE link
  - More complex traffic (more control streams, different schedules) have less of a reduction
    - But still deterministic if the control stream schedule is deterministic
- Greatly reduces the delays caused by interfering best-effort traffic

# Multipathing and redundancy



# Faster recovery on link failure

- Send critical data down separate paths simultaneously
  - Lost link does not kill connectivity, recovery is seamless
- Redundant paths can be used productively
  - Separate streams of non-critical traffic can take different paths between endpoints to avoid congestion
- Redundant links also work for time synch
  - Recovery from lost Grand Master is instant

# When will all this happen?

## Early participation is critical!

- Definition of AVB gen 2 features currently under way
  - Dominated by industrial control, professional A/V, and consumer market representatives
  - Time synch feature set almost frozen
    - 6-9 months
  - Scheduled queues at draft 0.0
    - Feature freeze also in 6-9 months
  - Preemption/multipath one year away
- Completion of standards roughly two/three years from now (2014/2015)

# Thank you!

# References

(all are hyperlinks to the public documents areas of the IEEE 802.1 website)

- AVB - Generation 2 Latency Improvement Options
- QoS requirements for Automotive Ethernet backbone systems
- AVB + Extensions for Industrial Communication
- Preemptive Transmission advantages