Switch Timing Parameters for Datasheets

Thomas Enzinger, B&R Industrial Automation
Astrit Ademaj, TTTech

15.6.2020

Problem Statement

For design and configuration of a real-time communication systems the timely behavior of the components needs to be known upfront.

Manufacturers of standardized components for such systems need a possibility to describe this timely behavior in a standardized way.

Typical workflow:

1. **Definition of overall timing requirements for data exchange**

   based on

   - planned production speed
   - location of sensors/actors
   - other mechanical/physical requirements

2. **Selection of communication technology and topology**

3. **Selection of automation components**

   → How to verify whether the overall system can meet the requirements for data exchange ?

Timing parameters:

- Signal runtime on media
    - No need for standardization, simple statement of [time/distance]
- Time errors dependent PTP implementation
    - Standardization ongoing in TSN-IA profile

- **Switch forwarding delays**

    → Standardization for Datasheets required !

Switch forwarding delays …

… must be **predictable**, independent of the features or strategies used in the TSN domain:

- **Strategy with individual Qbv configurations on each switch**

  Needed to calculate the individual Qbv configurations

- **Strategy with identical Qbv config on all switches in the domain**

  Needed to calculate the common Qbv schedule

- **Strategy without Qbv**

  Needed to calculate the worst case end-to-end latency

Existing definition in 802.1Qcc

802.1Qcc, Subclause 12.32.1 defines the following “Bridge Delay attributes”:

- independentDelayMin
- independentDelayMax
- dependentDelayMin
- dependentDelayMax

“Each set of Bridge Delay attributes is accessed using three indices: ingress port, egress port, and traffic class”

Missing information for offline description:

The following parameters are not considered in 802.1Qcc, because the bridge will provide the parameters based on the current state at the moment when the attributes are accessed:

- Link speed of the addressed ports
- Selected bridge features, like traffic selection mechanism and shapers

Large number of attributes:

Example 1:

Bridge ports: 5
Link speeds: 2 (e.g. 100 Mbit/1Gb)
Traffic classes: 2 (e.g. relevant classes for isochronous path computation)

Number of attribute **sets**: **160** = (ports)*(ports-1)*(speeds^2)*classes

Example 2:

Bridge ports: 8
Link speeds: 3 (e.g. 10 Mbit/100 Mbit/1Gb)
Traffic classes: 2

Number of attribute **sets**: **1008**

!! Each **set** might contain up to 4 attributes !!

!! Variation of activated features (Strict Prio, Preemption, Qbv, …) **not yet considered** !!

Assumption for standard switches:

- Delays are **independent** of ports numbers or direction
- Delays are **dependent** on link speed and used features
- Datasheet allows to state attribute sets using ‘wildcards’ (e.g. from Port <any> to Port <any>)
- Datasheet allows to state particular feature sets (based on the selectable features in the PICS)

Example:

Port [1..n] – Port [1..n], 100 M - Gbit, Feature Set A
Port [1..n] – Port [1..n], 1G - 100 M, Feature Set A
Port [1..n] – Port [1..n], 100 M - 100 M, Feature Set A
Port [1..n] – Port [1..n], 1G – 1G, Feature Set A

→ Only 4 attribute sets per Feature Set required

What we need:

- Standardized way to define feature sets (shapers, traffic class, …)
- Standardized to define attribute sets
- Possibility to use wildcards (Port <any> to Port <any>)
- Possibility to define separate attribute sets for special ports (e.g. 1 POF port on 8-port switch)

Thank you for your attention

Thomas Enzinger, B&R Industrial Automation
Astrit Ademaj, TTTech

15.6.2020