



Ethernet TSN Nano Profil – Migrationshelfer vom industriellen Brownfield zum Ethernet TSN-basierten IIoT

Sebastian Schriegel¹, Alexander Biendarra², Thomas Kobzan¹, Ludwig Leurs³,
Jürgen Jasperneite^{1,2}

¹Fraunhofer IOSB-INA
32657 Lemgo

{sebastian.schriegel, thomas.kobzan, juergen.jasperneite}
@iosb-ina.fraunhofer.de

²Institut für industrielle Informationstechnik (inIT)
Technische Hochschule Ostwestfalen-Lippe
32657 Lemgo

{alexander.biendarra, juergen.jasperneite}@th-owl.de

³Lehrstuhl für Werkzeugmaschinen und Steuerungen (WSKL)
TU Kaiserslautern
67663 Kaiserslautern
leurs@rhrk.uni-kl.de

Zusammenfassung. IEEE 802.1 Ethernet TSN ermöglicht Netzwerkkonvergenz für zeitsensitive Applikationen und Protokolle und ist damit eine wichtige Basistechnologie für universelle, einheitliche und durchgängige industrielle Kommunikation vom Sensor bis zur Cloud. Es ist allerdings nicht einheitlich definiert, welche TSN- und Ethernet-Funktionen mit welchen Ressourcen (z. B. Speicher) dafür notwendig sind bzw. genutzt werden sollen. Hersteller, die IEC und die IEEE arbeiten deshalb daran Ethernet TSN für den Einsatz in der industriellen Automation in einem engeren Rahmen zu definieren als die IEEE 802.1-Standardfamilie dies tut. Das entsprechende Profil IEC/IEEE 60802 TSN Profile for Industrial Automation soll 2021 fertig sein. Die hohe Einsatz- und Anforderungsbandbreite (Sensor bis Cloud) macht eine homogene Lösung aber schwer und die Entwicklung und Verbreitung der Technologie in den Anwendungen wird sehr viel Zeit benötigen. In diesem Beitrag wird ein TSN-Nano Profil vorgeschlagen, das einerseits Ethernet TSN für die Feldebene (Sensoren und Aktoren) skalieren soll und andererseits passende heutige IEC-Echtzeit Ethernet-Hardware mit speziellen Umsetzungsmethoden per Firmware mit TSN-Funktionen updaten kann und so die Markteinführung von Ethernet TSN in der Automation und auch das Retrofitting von existierenden Anlagen vereinfacht und beschleunigt. Es wird vorgeschlagen das Ethernet TSN Nano Profil in das IEC/IEEE 60802 TSN-IA-Profil aufzunehmen und so eine Profilskalierung für einfachste Geräte zu erzielen und Retrofitting zu ermöglichen.

1 Entwicklungstendenzen von Ethernet TSN im Applikationsfeld industrielle Automation und Motivation für ein Ethernet TSN Nano Profil

Heute werden in der industriellen Automation Echtzeit-Kommunikationssysteme eingesetzt, die teilweise IEEE Ethernet-Technologie enthalten, spezifische Anforderungen nach Echtzeit oder Robustheit mit IEC-Standards ergänzen und untereinander nicht interoperabel sind (z. B. PROFINET, Sercos III, EtherCAT: OT-Operation Technology). Weiterhin werden IT-Technologien wie IP und OPC UA für die vertikale Vernetzung verwendet, die Kopplung zur OT erfolgt über Gateways. Für Industrie 4.0-Applikationen ist eine flexible und durchgängige Vernetzung vom Sensor bis in die Cloud notwendig, die als IIoT – Industrial Internet of Things bezeichnet wird und die hierarchisch organisierte Automatisierungspyramide auflöst [WO07] [SC18]. IEEE 802.1 Ethernet TSN (Time Sensitive Networks) wird als Kommunikationsinfrastruktur gesehen, welche dies leisten kann, da verschiedene zeitsensitive und nicht-zeitsensitive IT- und OT-Protokolle gleichzeitig (konvergent) das Netzwerk nutzen können. Neben Konvergenz verspricht Ethernet TSN Flexibilität (stoßfreie Re-Konfiguration) und aufgrund der Verwendung von IEEE-konformer Hardware Gerätekostenoptimierung (IEEE Ethernet-Standard-Chips) und Kommunikationsleistungssteigerung (z.B. Link Speeds > 100 Mbit/s) [SC17] [IM11] [SC11]. Es handelt sich bei Ethernet TSN um eine Reihe neuer IEEE 802.1 Ethernet Bridging Standards. Verschiedene Interessensgemeinschaften und Nutzerorganisationen kombinieren Ethernet TSN mit Kommunikationsprotokollen und Methoden der Gerätemodellierung und Konfiguration zu Gesamtlösungen. Beispiele sind die Profibus-Nutzerorganisation (PROFINET over TSN) [XS17], die EtherCAT Technology Group (EtherCAT TSN Profile) [EC18], Sercos International (Sercos over TSN) [SE18] oder verschiedene Bestrebungen OPC UA mit TSN zu kombinieren. Viele der neuen TSN-Standards erfordern eine Umsetzung in Hardware, also neue Kommunikations-ASICs. Dies ist aufwendig und insbesondere für die große Gerätevielfalt eine Herausforderung für die Komponentenhersteller. Hinzu kommt, dass es verschiedene Ausprägungen (Profile) der zu TSN gehörenden IEEE 802.1-Standardfamilie in Systeme und Chips sowie spezifischer Chipeigenschaften wie Speicher, Latenzzeit, Queues oder Zeitstempelauflösung geben kann. Die IEEE und die IEC arbeiten deshalb an einen gemeinsamen TSN-Profil für die industrielle Automation (IEC/IEEE 60802 TSN-IA), welches TSN für das Anwendungsfeld der industriellen Automation in einem engeren Rahmen definieren soll als die IEEE 802-Standardfamilie dies tut. Im Jahr 2021 soll das Profil fertiggestellt sein. Durch Ethernet TSN soll die Netzwerktechnik auf dem Layer 2 in der industriellen Automation also einheitlicher und damit überhaupt erst einmal potential interoperabel werden, eine vollständige Homogenität wird aber auch aufgrund der hohen Einsatzbreite vom Sensor bis in die Cloud nicht erreicht und eine Markt- und Applikationsdurchdringung wird Jahrzehnte dauern. In der Zeit können viele neuen Anwendungen wie z. B. datenbasierte Services oder flexible Anlagen nicht oder nur mit einer beschränkten Leistung umgesetzt werden oder müssen umständlich und teuer z.B. mit Gateways und parallelen Netzwerken umgesetzt werden.

Dieser Aufsatz beschreibt Ethernet TSN aus der Sichtweise der Heterogenität, Hürden bei der Marktdurchdringung und dem aktuellen Status der Profilbildung in Kapitel 2. Die Updatefähigkeit von heutigen Echtzeit-Ethernet-Geräten (inkl. Anlagenretrofitting), Skalierung von Ethernet TSN für einfachste Geräte (Sensor- und Aktorebene) und Konfiguration solcher TSN-Systeme sind die Themen dieses Beitrages. Ziel der Betrachtungen ist eine einfache und schnellere Migration hin zu Ethernet TSN-basierten Industriellen Netzwerken (inkl. Retrofitting) und Skalierbarkeit von TSN-Profilen für einfachste Geräte. In Kapitel 3 wird dazu das Ethernet TSN Nano-Profil als kleinster gemeinsamer TSN-Funktions- und Ressourcennenner und potentielle Update-Chance für (IEC-Standard-) Echtzeit-Ethernet-Geräte vorgestellt und Verfahren beschrieben mit denen in heterogenen Netzwerken TSN-Geräte Geräte mit weniger TSN-Funktionen (die z.B. basierend auf IEC-Echtzeit Ethernet-Hardware) im Verbund unterstützen. Kapitel 4 zeigt darauf aufbauend mit welchen Modellen heterogene Netze, die aus Geräten mit unterschiedlichen Profilen und unterschiedlichen TSN-funktionsstärken aufgebaut sind, konfiguriert werden können und dass Echtzeiteigenschaften mit solchen Netzen erreichbar sind.

2 Heterogenität von Ethernet TSN in Profilen, Systemen und Geräten

Ethernet TSN wird grundsätzlich als Technologie gesehen, welche das Potenzial hat die industrielle Kommunikation auf Basis von IEEE-Standards zu vereinheitlichen. Der Anspruch ein so breites Anforderungsfeld vom Sensor bis zur Cloud mit einer einheitlichen Lösung zu bedienen ist hoch. Vollständig homogene Netzwerke sind selbst innerhalb der TSN-Anwendungsdomäne Industrieautomation aus mehreren Gründen nicht absehbar:

- Divergente Anforderungen der Applikationen vom Sensor bis in die Cloud: Updateraten, Verfügbarkeitslevel, Synchronität, Kostensensitivität, Verlustleistung, Rekonfiguration und Dynamik, Geräte- und Datenmengen, Topologien
- Divergente Geräteklassen vom Sensor bis in die Cloud mit z. B. Bridged Endstations mit 2 Ports und Link Speeds zwischen 0,01 und 1Gbit/s im Feld (Maschinen und Anlagen) oder TSN im Fabriknetzwerk mit z. B. Switches mit mehr als 48 Ports und Link Speeds zwischen 1 und 10 Gbit/s
- Verschmelzung von IT und OT durch konvergente Netze: Es kann im Prinzip jedes Protokoll und jede zeitsensitive Anwendung auf jede andere treffen. Der Bedarf und die Existenz von TSN-Profilen zeigen, dass Homogenität, wenn überhaupt, nur innerhalb von Anwendungsdomänen erreichbar ist.
- Time-to-Market- und Kosten-Druck bei den Herstellern (Notwendigkeit eines schnellen Aufbaus großer Gerätevielfalt): Geräte und Systeme werden auf dem Markt gebracht, die nur einen Teil der TSN-Standards unterstützen
- Differenzierungsstrategien
- Investitionsschutz bestehender Geräte und Systeme und unterschiedliche Migrationswege
- Anlagenretrofitting

Die Heterogenität entsteht in der Auswahl der IEEE-Standards (IEEE 802.1, IEEE 802.3), die für das Gerät oder das System zum Einsatz kommen sollen (z.B. Preemption ja oder nein und für welche Link Speeds), in der Auslegung und Implementierung der Standards in Systeme (für IEEE 802.1AS z.B. die Zeitstempelauslösung oder für das Switching die Größe der Pufferspeicher) und in der Konfiguration und Gerätemodellierung. Die folgende Tabelle zeigt IEEE Ethernet-Standards und Funktion sowie implementierungsabhängige Leistungskriterien in den beiden linken Spalten. Dazu sind TSN-Profile dargestellt, die versuchen TSN für spezifische Anwendungsdomänen und Protokolle enger zu definieren und so Homogenität zu fördern. Die Tabelle zeigt die zum Zeitpunkt der Einreichung dieses Papiers verfügbaren Informationsstand auf Basis veröffentlichter Dokumente.

Wie ein Netzwerk aussehen kann, das heterogen aufgebaut ist, zeigt [Abbildung 1](#). Die Topologie besteht aus Multiport-Gigabit-Switches, die neben Preemption, Synchronisation und allen Traffic Shapern viel Pufferspeicher besitzen (in der Abbildung als Typ C bezeichnet), 100 Mbit/s-Zweiportbridges, die auf IEC-Echtzeit Ethernet-Hardware basieren (Typ D) sowie Geräten die zwischen diesen beiden Funktions- und Ressourcenextremen liegen (Typen A und B).

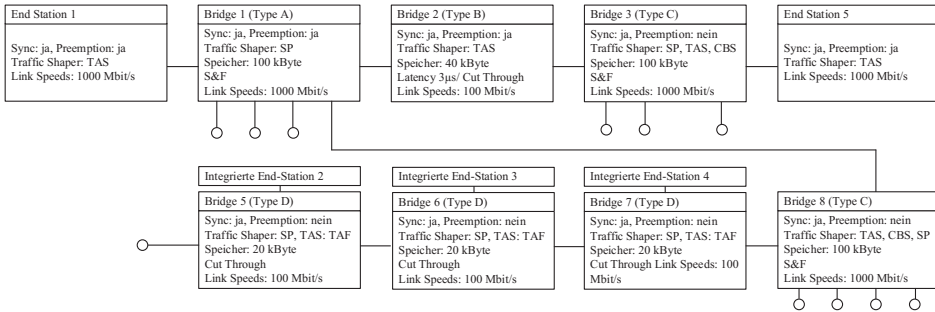


Abb. 1. Heterogen aufgebautes Ethernet-Netzwerk

Es entstehen folgende Fragen:

- Wie können existierende IEC-Echtzeit Ethernet-Systeme und die dafür verfügbare Gerätevielfalt möglichst gut zu Ethernet TSN migriert werden? Wie können bestehende Anlagen im Sinne eines Retrofittings mit TSN einfach umgerüstet werden und so z.B. von daten-basierten Services profitieren?
- Wie kann Ethernet TSN für einfache Geräte skaliert werden?
- Wie können in heterogenen Netzwerken Echtzeiteigenschaften (max. Latenz, Synchronisationsgenauigkeit) realisiert und garantiert werden?

In diesem Beitrag werden sich ergänzende Lösungskomponenten beschrieben:

Kapitel 3: TSN Nano Profil für die industrielle Automation: Skalierung von Ethernet TSN für einfache Geräte und Firmware-Update-Möglichkeit für IEC-Echtzeit Ethernet-Geräte

Kapitel 4: Ethernet TSN-Konfigurationsmodellvarianten für heterogene Netze

3 Ethernet TSN Nano-Profil – Skalierung von Ethernet TSN für die Feldebene und Retrofitting von IEC Echtzeit Ethernet-Hardware und -Anlagen

Das Ethernet TSN-Nano Profil ist so aus Funktionen und Ressourcen zugeschnitten, dass die Anforderungen an die Hardware den heutigen IEC-Echtzeit Ethernet-Mechanismen möglichst ähnlich sind. Damit werden die Chancen maximiert, dass die TSN Nano-Funktionalität per Treiberupdate auf heute verfügbare und für gängige Echtzeit-Ethernet-Protokolle im Einsatz befindliche Hardware installiert werden kann (muss für die spezifischen Geräte geprüft werden). Die im Profil enthaltenen Funktionen leiten sich aus Kompromissen zwischen Hardwareanforderungen, durchgängiger, interoperabler und anforderungserfüllender (TSN-) Kommunikation (IEC/IEEE 60802 Use Cases [IE18]: isochrone Kommunikation: garantierte maximale kleine Latenzzeit und maximale Latenzzeit ohne Übertragungsverluste) und Möglichkeiten IEC-Echtzeit Ethernet-Funktionen für TSN zu verwenden ab. Die am stärksten verbreitete Geräteklasse sind Bridged Endstations mit zwei externen Ports. Der Beitrag fokussiert sich deshalb auf diese Geräteklasse. Die folgenden Standards und Funktionen sollen unterstützt und teilweise mit speziellen vereinfachten Verfahren umgesetzt werden.

Funktion/ Standard		Umsetzungsverfahren, Parameter und Ressourcen
IEEE 802.3 MAC		2 externe Ports 100 Mbit/s, 1 interner Port
IEEE 802.1AS		Sync-ID 20 Workingclock
IEEE 802.1Q	VLAN-Prioritäten, Queues, Switch	2 Queues mit je 1,5 kByte Speicher je Port, Cut Through
	Time Aware Shaper	Guard Band, TAF (Time Aware Forwarder) Abschn. 3.1
	VLAN-ID und MAC	Einträge nur für lokalen Port, keine VLAN-ID-Auswertung

Per-Stream-Filtering and Policing (IEEE 802.1Qci-2017) und das Remapping von VLAN-Prioritäten sind im Ethernet TSN Nano-Profil nicht enthalten. Diese Funktionen müssen aber nicht zwangsläufig an den TSN-Domänengrenzen aktiviert werden, sondern können auch von den TSN-Switches ausgeführt werden, an die die Ethernet TSN-Nano-Linientopologien angeschlossen werden. In einem so aufgebauten und konfigurierten TSN-Netzwerk ist die VLAN-ID-Auswertung in den Ethernet TSN-Nano-Geräten auch nicht zwingend notwendig. Diesbezüglich folgen in den nächsten beiden Unterkapiteln spezielle Umsetzungsverfahren für Time Aware Shaper.

3.1 Time Aware Forwarder: Funktionsmodus für den Time Aware Shaper für Bridged Endstations mit eingeschränkter Weiterleittabelle und VLAN-Auswertung

Als IEC-Hardwarefreundliche Umsetzungsvariante des Time Aware Shaper wird hier Time Aware Forwarding (TAF) vorgeschlagen. TAF basiert darauf, dass gegenüber Traffic Shaping bei Geräten mit 2 Ports Frames einfach nur weitergeleitet werden müssen und eine Prüfung von MAC und VLAN nur für den lokalen Empfang notwendig ist. Um das Time Aware Forwarding (TAF) zu realisieren, sind 2 Queues notwendiger Gates zeitgesteuert zyklisch geöffnet und geschlossen. Der garantierte Startzeitpunkt für die Time-Aware Queue Q1 wird über ein Guard Band (GB) abgesichert. In Q1 werden Frames eingereiht, die während einer entsprechenden Empfangs-Gatezeit empfangen wurden (RX-Gate Q1) (keine IEEE-Funktion, kommt aus der IEC). Die folgende [Abbildung 2](#) zeigt Beispiele wie das Queuing und Forwarding funktionieren.

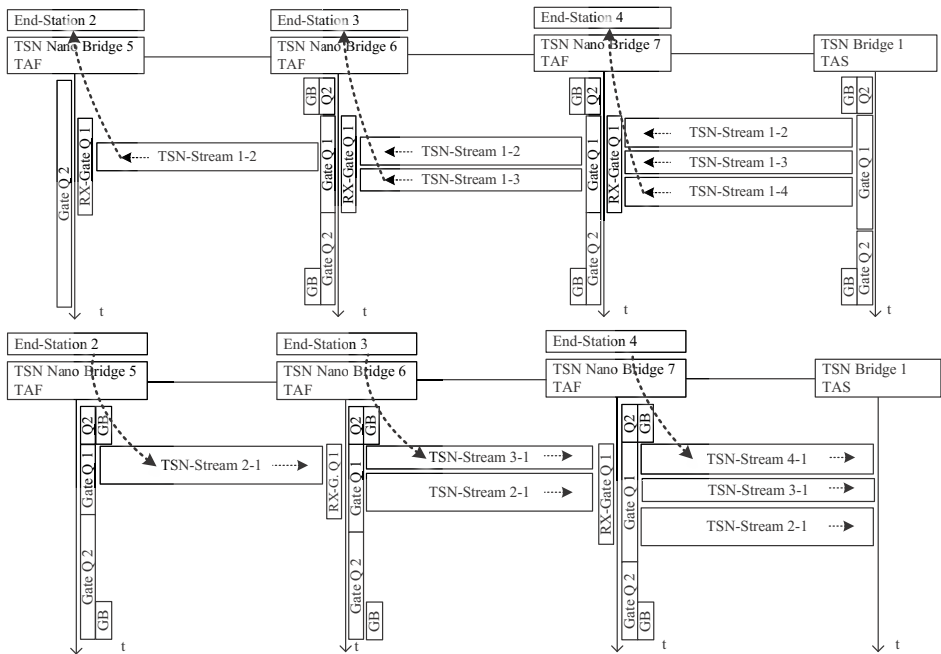


Abb. 2. Funktionsprinzip Time Aware Forwarder: Queuing für zeitsensitive Frames mit RX-Gate

Nachteile:

- 1.) TAF basiert darauf, dass alle Frames die nicht für die lokale Applikation bestimmt sind, ohne Prüfung weitergeleitet werden (wie z.B. bei SERCOS III), das Normalverhalten von TSN-Streams ist aber verwerfen und nicht fluten. IEEE-Konformität müsste durch Standardisierung von TAF noch erfolgen.
- 2.) Da Preemption nicht unterstützt wird, TAS aber schon, sinkt die effektive Bandbreitennutzung für azyklische, ungeplante Kommunikation (Q2) und es muss eine Mindestzeit für die maximal MTU zwischen den TAS-Fenstern eingehalten werden.
- 3.) TAF ist für die Traffic-Klasse isochron (*garantierte maximale kleine Latenzzeit*) TAS ebenbürtig, weil die TSN-Frames zeitgeplant übertragen werden was durch eine Netzwerkconfiguration sichergestellt ist. Je nach Netzwerkdiameter und der Anzahl von dezentralen Kommunikationspunkten (Multi-Controller) muss dies für die Traffic-Klasse *maximale Latenzzeit ohne Übertragungsverluste* bewertet werden.

Da die Anforderung insbesondere an die TAS-Funktion (Guard Band, Queue-Auswahl auf Basis RX-Gate) eine IEC-Echtzeit Ethernet-Funktion ist (die wiederum sehr heterogen ist), wird im folgenden Kapitel 3.2 noch eine Variante vorgeschlagen bei der die Bridges in einer Linientopologie nur eine Queue benötigen und Nachbargeräte sowie eine übergeordnete Konfiguration die Funktion nachbilden.

3.2 Domain-based Time Aware Forwarder: Funktionsmodus für garantierte niedrige Latenz in Linientopologien mit Geräten ohne Time Aware Shaper

Um eine niedrige Latenz, insbesondere in einem 100 Mbit/s-TSN-Netzwerk zu garantieren, ist der Time Aware Shaper notwendig. Verfügen Chips nicht über TAS-Funktionen, steigen insbesondere in Linientopologien die Latenzen stark an. Um Geräte in einem zeitsensitiven Netzwerk nutzen zu können, wird hier ein Funktionsmodus vorgestellt, bei dem die Nachbargeräte dieser Geräte oder Domänen eine Time Aware Forwarder-Domäne bilden. Für diese Domäne werden Übertragungszeitschlitze für die azyklische Kommunikation so geplant, dass diese die (ebenfalls geplante) zyklische Kommunikation nicht stört und so spezifische TAS-Hardware nicht notwendig ist. Da die vollständige Kommunikation in der Domäne also zeitgeplant ist, wird kein lokales Traffic Shaping benötigt und die Geräte kommen entsprechend mit einer Queue aus. Die Kontrolle der azyklischen Kommunikation stützt sich dabei auf folgende Funktionen:

- 1.) Die Geräte senden von der lokalen Applikation (End-Station) nur zeitgesteuert in die D-TAF-Domäne (also auch die azyklischen Frames). Zur Optimierung können die für die einzelnen Geräte geplanten Kanäle für die azyklischen Frames auf Kommunikationsphasen aufgeteilt werden was eine hochfrequente Kommunikation (kleine Zykluszeit) der zyklischen Kommunikation auch in so einer Domäne ermöglicht.

- 2.) Die (Voll-TSN-) Geräte an den D-TAF-Domänengrenzen ([Abbildung 2](#): Bridge 1) nutzen eine verlängerte Gatezeit der zeitsensitiven Q1 um die D-TAF-Domäne vor ungeplanten Frames zu schützen.

In der vollständigen D-TAF-Domäne werden azyklische Frames nie gepuffert, sondern immer im Cut Through-Verfahren weitergeleitet. [Abbildung 3](#) zeigt eine D-TAF-Domäne im heterogenen TSN-Netzwerk.

Nachteile (zusätzlich zu TAF):

- 1.) Das D-TAF-Verfahren verursacht eine verringerte Bandbreite für azyklische Kommunikation. Unter der Annahme, dass TSN Nano-Profil-Geräte oder -Linien als Kammzähne an eine leistungsfähige TSN-Hauptdomäne angeschlossen werden, könnten die verschlechterten Eigenschaften als unkritisch angesehen werden (da es hier um die weniger stark beanspruchten Netzwerkzweige geht).
- 2.) Eine D-TAF-Domäne inkl. ihrer Nachbargeräte muss explizit konfiguriert werden. Dies kann eine zentrale Konfiguration übernehmen oder es könnte eine dezentrale unterlagerte D-TAF-Konfiguration Anwendung finden.
- 3.) Die Topologie wird eingeschränkt: Eine D-TAF-Domäne muss entweder von TAS-fähigen Bridges umsäumt oder der D-TAF-End-Port der Linie muss offen sein.

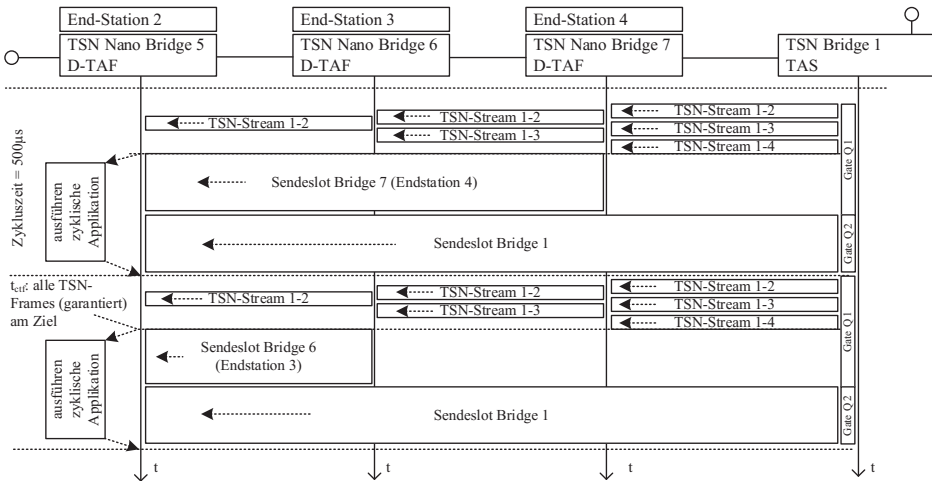


Abb. 3. Ort-Zeit-Diagramm und Konfiguration des D-TAF (hier Outbound-Kommunikation, Inbound-Kommunikation (Upstream) zur Steuerung sieht identisch aus)

4 Ethernet TSN-Konfigurationsmodellvarianten für heterogene Netze

In diesem Kapitel werden grundlegende TSN-Konfigurationsvarianten beschrieben und gezeigt, dass Echtzeiteigenschaften in heterogenen Netzwerken erreichbar sind und wie sich TSN-Nano-Profilgeräte dabei einfügen. **Abbildung 4** zeigt ein Beispiel für applikationstaktsynchrone (isochrone: garantierte maximale kleine Latenzzeit) Kommunikation in einem heterogenen Netz, bei der zu einem garantierten spätesten Zeitpunkt t_{ctf} (ctf – cyclic traffic finished) der Kommunikationszyklus abgeschlossen ist.

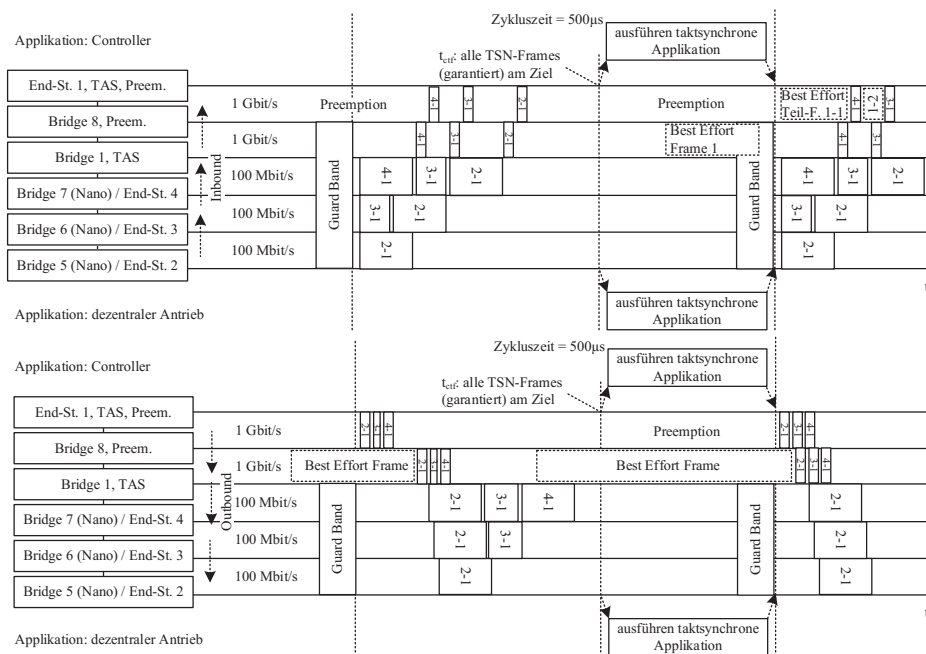


Abb. 4. Taktsynchrone Kommunikation für die beiden Kommunikationsrichtungen

Alle hier vorgestellten TSN-Konfigurationsmodi basieren darauf, dass keine Sendezeitpunkte in den Bridges verwendet werden, sondern das Senden der TSN-Endpoints (Talker) zum synchronisierten Zyklusbeginn erfolgt. Bei TSN funktioniert dieser Modus, da nicht ausschließlich im Cut Through-Verfahren weitergeleitet werden muss, sondern auch das Zwischenspeichern von Frames möglich ist. Gegenüber z.B. PROFINET IRT enthält die Konfigurationslogik aus diesem Grund ein Element mehr: die Berechnung und Prüfung ob die einzelnen Bridges (dies ist insbesondere für die Multiport-Bridges relevant) genug Speicher für die berechnete Sendereihenfolge haben. **Abbildung 5** zeigt die Elemente einer TSN-Konfigurationslogik.

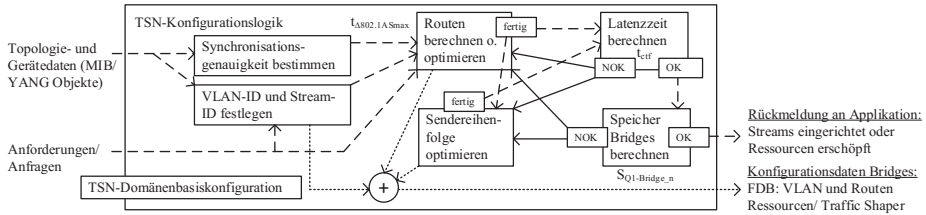


Abb. 5. Elemente einer TSN-Konfigurationslogik

Je nach Informationslevel, die der Konfigurationslogik zur Verfügung steht und den Applikationsanforderungen (Stream: QoS und Verortung der Talker und Listener in der Netzwerktopologie) sind unterschiedliche TSN-Funktions- und Konfigurationsmodi notwendig oder möglich. Die TSN-Konfigurationslogikelemente Latenzzeit- und Speicherberechnung und Sendereihenfolge sind dann unterschiedlich ausgeprägt. Die Berechnung der Routen kann z.B. mit dem Dijkstra-Algorithmus vorgenommen werden. Darauf wird hier nicht eingegangen. Die Berechnung der Synchronisationsgenauigkeit wurde in [SC15] beschrieben und wird hier auch nicht weiter behandelt.

Applikationsanforderungen		Zykluszeiten, Datenmengen		
		Taktsynchrone Applikation	Freilaufende Applikation	
Informationslevel		Kommunikationstyp: Garantierte maximale kleine Latenzzeit	Kommunikationstyp: Maximale Latenzzeit und keine Überlastverluste	
Geräte Modelle	Single-Controller (Inbound-Outbound-Unterscheidung)	Keine Pfad-redundanzen	TSN-Funktionsmodus: fluten Kap. 4.2	TSN-Funktionsmodus: fluten Kap. 4.2
		Pfad-redundanzen	nicht möglich	nicht möglich
	Multi-Controller (inkl. Konvergenz) (kein zentraler Komm.-Punkt)	Keine Pfad-redundanzen	TSN-Funktionsmodus: fluten Kap. 4.1	TSN-Funktionsmodus: fluten Kap. 4.1
		Pfad-redundanzen	nicht möglich	nicht möglich
Geräte Modelle und Topologie	Single-Controller (zentraler Kommunikationspunkt)	Keine Pfad-redundanzen	TSN-Funktionsmodus: Pfadplanung und (optional) Sendereihenfolgeoptimierung Kap. 4.3	TSN-Funktionsmodus: Pfadplanung und Speicherauslastung Kap. 4.3
		Pfad-redundanzen	TSN-Funktionsmodus: Pfadplanung und (optional) Sendereihenfolgeoptimierung Kap. 4.4 (nur Ausblick)	TSN-Funktionsmodus: Pfadplanung Kap. 4.3
	Multi-Controller (inkl. Konvergenz) (kein zentraler Komm.-Punkt)	Keine Pfad-redundanzen	TSN-Funktionsmodus: Pfadplanung und (optional) Sendereihenfolgeoptimierung Kap. 4.4 (nur Ausblick)	TSN-Funktionsmodus: Pfadplanung Kap. 4.3
		Pfad-redundanzen	TSN-Funktionsmodus: Pfadplanung und (optional) Sendereihenfolgeoptimierung Kap. 4.4 (nur Ausblick)	TSN-Funktionsmodus: Pfadplanung Kap. 4.3

4.1 TSN-Funktionsmodus ohne Topologiewissen in der Konfigurationslogik

In diesem Funktionsmodus leiten Bridges alle TSN-Frames an alle Ports weiter (außer auf dem den es empfangen worden ist), es erhält also auch jeder Switch für jeden Port die gleiche Konfiguration. Es dürfen also keine Ringe in dem Netz vorkommen (was ja Topologiewissen ist).

1. Bestimmen Worst Case Topologie: Linientopologie, alle Frames (Inbound und Outbound) über den maximalen Netzwerkdiаметer.



2. Bestimmen Worst Case Link Speed-Kombinationen (maximal kleinste Bandbreite B)
3. Bestimmen Worst Case Preemption-Kombinationen (maximal wenig Links mit Preemption)
4. Summen Delays: n Links, m Bridges (inkl. egress Traffic Shaper, Queueing, GB) und k Frameübertragung

$$t_{ctf} = \sum_1^n t_{LinkDelay_n} + \sum_1^m t_{BridgeDelay_m} + \sum_1^k t_{Frame_k} + t_{\Delta_{802.1ASmax}}$$

$$t_{LinkDelay_n} = f \begin{cases} \text{Online} \\ \text{Offline} \end{cases}$$

$$t_{BridgeDelay_m} = f \begin{cases} \text{Traffic Shaper} \\ \text{Latenzzeit} \\ \text{Preemption} \\ \text{Guard Band} \end{cases} \quad t_{Frame_k} = f \begin{cases} \text{Länge} \\ \text{Bitrate Link} \end{cases}$$

5. Berechnung des Speicherbedarfes für Queue 1 S_{Q1} : Worst Case für die Speicher: Sterntopologie: Alle End-Station werden an eine Bridge angeschlossen und je Port p wird ein Durchsatz D (t) in die Queue 1 eingespeist und mit einer Bandbreite B_{egress} geleert.

$$S_{Q1-Bridge_n}(t) = \left[\sum_{p=0}^{p=n-1} \int_{t=0}^{t=t_{ctf}} D_{ingress_n} dt \right] - B_{egress_{port_n}} D_{ingress_n}(t)$$

$$= f \begin{cases} \text{ingress Link Speed} \\ \text{Frame}_k \end{cases}$$

Der Vorteil diesen Funktionsmodus ist, dass jede Heterogenität möglich ist und das Netzwerk robust und dynamisch ist. Topologieänderungen können ohne die TSN-Konfiguration zu ändern erfolgen, Multi-Controller können genutzt werden. Nachteilig ist, dass keine Ringe möglich sind, die garantierbare Latenzzeit bzw. kleinste Zykluszeit hoch ist und die Ressourcen verschwenderisch eingesetzt werden.

4.2 Spezieller TSN-Funktionsmodus für Netzwerke mit einem zentralen Kommunikationspunkt (Single-Controller Networks)

Der Funktionsmodus ist gleich zu dem des vorherigen Kapitels 4.1. Da es aber einen zentralen Kommunikationspunkt in der TSN-Domäne gibt, kann Inbound- und Outbound-Kommunikation unterschieden werden (de facto Topologiewissen). Dies führt zu Vereinfachungen in der Berechnung der notwendigen Kommunikationsfenster und auch zu einer deutlichen Reduzierung der Zeitfenster t_{ctf} .

Berechnung des Kommunikationsfensters $t_{ctf-outbound}$ und $t_{ctf-inbound}$

$$t_{ctf-inbound} = \sum_1^n t_{LinkDelay_n} + \sum_1^m t_{BridgeDelay_m} + \sum_1^k t_{Frameinbound_k} + t_{\Delta_{802.1ASmax}}$$

$$t_{ctf-outbound} = \sum_1^n t_{LinkDelay_n} + \sum_1^m t_{BridgeDelay_m} + \sum_1^k t_{Frameoutbound_k} + t_{\Delta_{802.1ASmax}}$$

Berechnung des Speichers für Queue 1 S_{Q1}

$$S_{Q1-Bridge_n}(t) = \left[\sum_{p=0}^{p=n-1} \int_{t=0}^{t=t_{ctf}} D_{ingress_n} dt \right] - B_{egressport_n}$$

Das Verfahren ist robust, bei einer Topologieänderung muss aber eine Neuberechnung und teilw. Neukonfiguration erfolgen. Um ein Optimum bzgl. Kosteneffizienz bei Erfüllung der Anforderungen zu erreichen, muss die Topologie bereits während der Anlagenplanung bekannt sein. Das ist in der Netzwerktechnik und auch in der Echtzeitkommunikation nicht neues, bei Feldbus-Systemen war dies aber sehr einfach überschaubar. Da die Sendereihenfolge nicht optimiert ist, wird die für die gegebenen physikalischen Netzwerkressourcen nicht die optimal mögliche kleinste Latenzzeit t_{ctf} erreicht.

4.3 Spezieller TSN-Funktionsmodus mit Topologiewissen und Sendereihenfolgeoptimierung für Single-Controller Networks

1. Routen bestimmen (z.B. Dijkstra-Algorithmus)
2. Sendereihenfolge festlegen Outbound: Frame mit der längsten Route zuerst
3. Summen Latenzzeiten berechnen: Link, Bridge und Frameübertragung (Kommunikationsfenster t_{ctf})

$$t_{ctf-inbound} = \sum_1^n t_{LinkDelay_n} + \sum_1^m t_{BridgeDelay_m} + \sum_1^k t_{Frameinbound_k} + t_{\Delta_{802.1ASmax}}$$

für

$$\sum_1^n t_{LinkDelay_n} > \sum_1^m t_{BridgeDelay_m} + \sum_1^k t_{Frameinbound_k}$$

4. Berechnung des notwendigen Speichers für Queue 1 S_{Q1} :

$$S_{Q1-Bridge_n}(t) = \sum \int_t^{t=t_{ctf}} D_{ingress_n} dt - B_{egressport_n}$$

Mit dem Vorgehen kann die minimal erreichbare Latenzzeit erzielt werden, die Konfiguration einer Multi-Controller-TSN-Domäne ist aber so nicht möglich.

4.4 Ausblick allgemeine TSN-Funktion und -konfiguration für konvergente Netzwerke (inkl. Multi-Controller Networks, kein zentraler Kommunikationspunkt)

Ein konvergentes Netzwerk, das verschiedene zeitsensitive Applikationen und Protokolle von unterschiedlichen Kommunikationspunkten aus gleichzeitig überträgt, ist mit vielen der dargestellten Konfigurationsverfahren der letzten drei Kapitel nicht konfigurierbar, da sie auf Vereinfachungen basieren. Dies ist der Inhalt weiterer Arbeit.

5 Zusammenfassung

IEEE 802.1 Ethernet TSN ermöglicht Netzwerkkonvergenz für zeitsensitive Applikationen und Protokolle und ist damit eine wichtige Basistechnologie für universelle, einheitliche und durchgängige industrielle Kommunikation vom Sensor bis zur Cloud. Es ist allerdings nicht einheitlich definiert, welche TSN- und Ethernet-Funktionen mit welchen Ressourcen (z.B. Speicher) dafür notwendig sind bzw. genutzt werden sollen. Hersteller, die IEC und die IEEE arbeiten deshalb daran Ethernet TSN für den Einsatz in der industriellen Automation in einem engeren Rahmen zu definieren als die IEEE 802.1-Standardfamilie dies tut. Das entsprechende Profil IEC/IEEE 60802 TSN Profile for Industrial Automation soll 2022 fertig sein. Die hohe Einsatz- und Anforderungsbandbreite (Sensor bis Cloud) macht eine homogene Lösung aber schwer. Dazu kommen Kompromisse zwischen Entwicklungszeit und -kosten die inhomogene TSN-Lösungen fördern. In diesem Beitrag wird ein TSN-Nano Profil vorgeschlagen, das einerseits Ethernet TSN für die Feldebene (Sensoren und Aktoren) skalieren soll und andererseits passende heutige IEC-Echtzeit Ethernet-Hardware mit speziellen Umsetzungsmethoden per Firmware mit TSN-Funktionen updaten kann und so die Markteinführung von Ethernet TSN in der Automation und auch das Retrofitting von existierenden Anlagen vereinfacht und beschleunigt. Es wird vorgeschlagen das Ethernet TSN Nano Profil in das IEC/IEEE 60802 TSN-IA-Profil aufzunehmen und so eine Profilskalierung für einfachste Geräte zu erzielen und Retrofitting zu ermöglichen.

Literatur

- [WO07] Martin Wollschläger, Thilo Sauter, and Jürgen Jasperneite: The future of industrial communication. In: in IEEE Industrial Electronics magazine. IEEE, Mar 2017.
- [IE18] IEC/IEEE 60802 TSN Profile for Industrial Automation, online: <https://1.ieee802.org/tsn/iec-ieee-60802-tsn-profile-for-industrial-automation/>
- [SC17] Sebastian Schriegel, Carsten Pieper, Sergej Gamper, Alexander Biendarra, and Jürgen Jasperneite: Vereinfachtes Ethernet TSN-Implementierungsmodell für Feldgeräte mit zwei Ports. In: Kommunikation in der Automation - KommA 2017, Magdeburg, Germany, Nov 2017.
- [XS17] Xaver Schmidt: The goals of integration of TSN into PROFINET. In: TSN/A Conference 2017.
- [SE18] SERCOS: Sercos over TSN. Online: www.sercos.de, 2018.
- [EC18] Karl Weber: Whitepaper EtherCAT Technology Group - EtherCAT and TSN - Best Practices for Industrial Ethernet System Architecture. Online: www.ethercat.org, 2018.
- [SC07] Schriegel, Sebastian; Jasperneite, Jürgen: Investigation of Industrial Environmental Influences on Clock Sources and their Effect on the Synchronization Accuracy of IEEE 1588, International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS 2007), Vienna, 2007.
- [PS18] TonTong Wang, Norman Finn, Xinyang Wang: TSN Profile for Network Slicing. Online: <http://www.ieee802.org/1/files/public/docs2018/new-tsn-wangtt-TSN-profile-for-network-slicing-0718.pdf>
- [P5G18] Applicability of TSN for 5G services: Online: <http://www.ieee802.org/1/files/public/docs2018/new-tsn-wangtt-applicability-of-TSN-for-5G-services-0318-01.pdf>
- [SPN18] 802.1DF Standard: Time-Sensitive Networking Profile for Service Provider Networks
- [In-V18] 802.1DG - Standard: Time-Sensitive Networking Profile for Automotive In-Vehicle Ethernet Communications
- [ULL18] Ahmed Nasrallah, Akhilesh Thyagaturu, Ziyad Alharbi, Cuixiang Wang, Xing Shao, Martin Reisslein, and Hesham ElBakoury: Ultra-Low Latency (ULL) Networks: The IEEE TSN and IETF DetNet Standards and Related 5G ULL Research. In: <https://arxiv.org/pdf/1803.07673.pdf>
- [SC11] Schriegel, Sebastian; Jasperneite, Jürgen: Taktsynchrone Applikationen mit PROFINET IO und Ethernet AVB. In: Automation 2011 – VDI-Kongress, Baden Baden, Juni 2011.
- [TE18] Whitepaper online: <http://www.ieee802.org/24/802.24-smart-grid-whitepaper.pdf>
- [TA18] Tutorial Automotive Ethernet: online: http://www.ieee802.org/802_tutorials/2017-07/tutorial-Automotive-Ethernet-0717-v02.pdf
- [IM11] Imtiaz, Jahanzaib; Jasperneite, Jürgen; Schriegel, Sebastian: A Proposal to Integrate Process Data Communication to IEEE 802.1 Audio Video Bridging (AVB). In: 16th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2011) Toulouse, France, Sep 2011.
- [SC15] Schriegel, Sebastian; Biendarra, Alexander; Ronen, Opher; Flatt, Holger; Leßmann, Gunnar, Jasperneite, Jürgen: Automatic Determination of Synchronization Path Quality using PTP Bridges with Integrated Inaccuracy Estimation for System Configuration and Monitoring. In: ISPCS 2015, Beijing, 2015.
- [KO18] Thomas Kobzan, Sebastian Schriegel, Simon Althoff, Alexander Boschmann, Jens Otto, Jürgen Jasperneite: Secure and Time-sensitive Communication for Remote Process Control and Monitoring. In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Torino, Italy, September 2018.

- [SC18] Schriegel, Sebastian; Kobzan, Thomas; Jasperneite, Jürgen: Investigation on a Distributed SDN Control Plane Architecture for Heterogeneous Time Sensitive Networks. In: 14th IEEE International Workshop on Factory Communication Systems (WFCS) Imperia (Italy), Jun 2018.
- [SK18] Stephan Kehrer: Time-Sensitive Networking (TSN) in Brownfield Applications. In: TSN/A Conference 2018
- [KD18] Keynote-Discussion: The Migration-Plans to TSN – Panel on the Battle of Fieldbus Protocols. In: TSN/A Conference 2018, Stuttgart
- [CBdb] P802.1CBdb – FRER Extended Stream Identification Functions. Online: <https://1.ieee802.org/tsn/802-1cldb/>

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.

