LINUTRONIX
LINUX FOR INDUSTRY

Software Time Sensitive Networking with Linux

TSN/A Conference 2021

Kurt Kanzenbach

Linutronix GmbH

29.09.2021



LINUTRONIX
LINUX FOR INDUSTRY

Agenda

1. Introduction
2. Software TSN
   - Time Synchronization
   - Time Aware Shaping
   - Linux Network Stack
3. Evaluation
4. Conclusion

Time Sensitive Networking

What is TSN?

- Time Sensitive Networking
- *Deterministic* Ethernet
- Defined via IEEE 802.1Q
- *Open and common* Standard

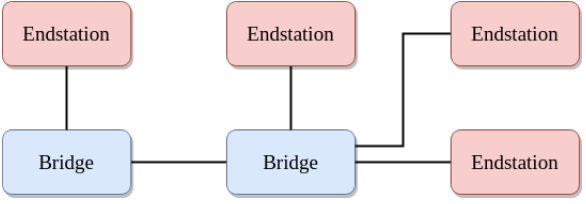


Figure: TSN network

TSN in Software with Linux

- **Hardware TSN:**
  - \+ Excellent precision
  - \+ Low cycle times possible
  - \- Specialized hardware required
  - \- Cost inefficiency
- **Software TSN:**
  - \+ Use of standard hardware
  - \+ Cost efficiency
  - \- Reduced precision
  - \- Increased cycle times

What precision can be achieved with software only solutions?

LINUTRONIX
LINUX FOR INDUSTRY

Agenda

1. Introduction
2. Software TSN
   - Time Synchronization
   - Time Aware Shaping
   - Linux Network Stack
3. Evaluation
4. Conclusion

LINUTRONIX
LINUX FOR INDUSTRY

Time Synchronization

Time Synchronization

- Time Synchronization is essential for TSN networks
- Linux supports PTP: IEEE 1588, IEEE 802.1AS
- Precision depends on message timestamping
- Three different variants available:
  - Software timestamping
  - Hardware timestamping

LINUTRONIX
LINUX FOR INDUSTRY

Linux Timestamping

Linux Timestamping

- PTP has to be supported by NIC/PHY/Switch drivers
- Currently ≈ 50 implementations available:
  - Intel: i210, i225, e1000
  - NXP: enetc, fec, gianfar, dpaa
  - Broadcom: bnxt, tg3
  - ST: stmmac
  - ...

```
root@tsn ~ # ethtool -T enp4s0
Time stamping parameters for enp4s0:
Capabilities:
        hardware-transmit
        software-transmit
        hardware-receive
        software-receive
        software-system-clock
        hardware-raw-clock
PTP Hardware Clock: 0
Hardware Transmit Timestamp Modes:
        off
        on
Hardware Receive Filter Modes:
        none
        all
```

LINUTRONIX
LINUX FOR INDUSTRY

Agenda

1. Introduction
2. Software TSN
   - Time Synchronization
   - Time Aware Shaping
   - Linux Network Stack
3. Evaluation
4. Conclusion

LINUTRONIX
LINUX FOR INDUSTRY

Time Aware Shaping

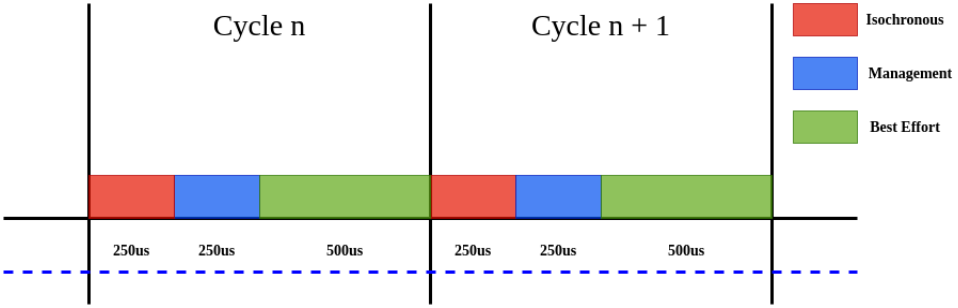


Figure: Example Schedule

LINUTRONIX
LINUX FOR INDUSTRY

TAPRIO

TAPRIO Qdisc

- Time Aware Priority Shaper
- Implementation as Qdisc
- According to 802.1Qbv
- Software variant available
- Parameters:
  - Schedule entries
  - Traffic classes
  - Tx queue mapping

```
root@tsn ~ # tc qdisc replace dev enp4s0 \
   parent root handle 100 taprio \
   num_tc 3 \
   map 2 2 1 0 2 2 2 2 2 2 2 2 2 2 2 2 \
   queues 1@0 1@1 2@2 \
   base-time 1234 \
   sched-entry S 01 2500000 \
   sched-entry S 02 2500000 \
   sched-entry S fc 5000000 \
   clockid CLOCK_TAI
```

LINUTRONIX
LINUX FOR INDUSTRY

Agenda

1. Introduction
2. Software TSN
   - Time Synchronization
   - Time Aware Shaping
   - Linux Network Stack
3. Evaluation
4. Conclusion

LINUTRONIX
LINUX FOR INDUSTRY

Overview

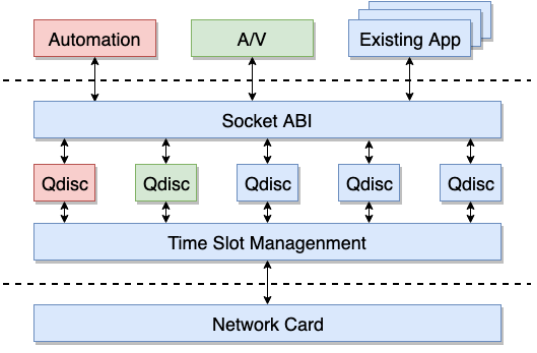


Figure: Linux TSN overview [1]

LINUTRONIX
LINUX FOR INDUSTRY

Linux Network Stack

Linux Network Stack

- Network stack introduces latency and jitter
- Several software optimizations available:
  - **RAW** sockets
  - **mmap** sockets
  - **AF_XDP** sockets
- `PREEMPT_RT` considerations:
  - Interrupt pinning
  - Soft Interrupt handling

Agenda

1. Introduction
2. Software TSN
   - Time Synchronization
   - Time Aware Shaping
   - Linux Network Stack
3. Evaluation
4. Conclusion

Test Setup

	Talker	Listener
Device	PC	mpxls1028 [2]
CPU	4x Intel i5	2x Cortex A72
RAM	4 GiB	4 GiB
NIC(s)	Intel i210/i225	NXP enetc
Kernel	5.11-rt	5.11-rt
Driver	igb/igc	enetc

Table: Test devices

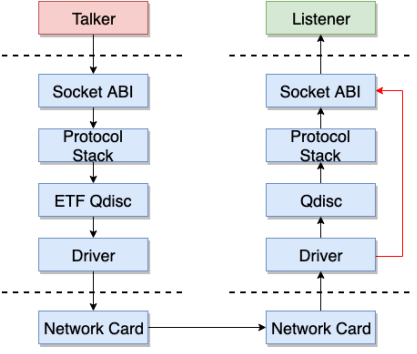
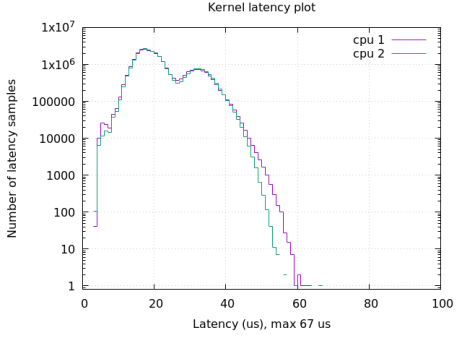
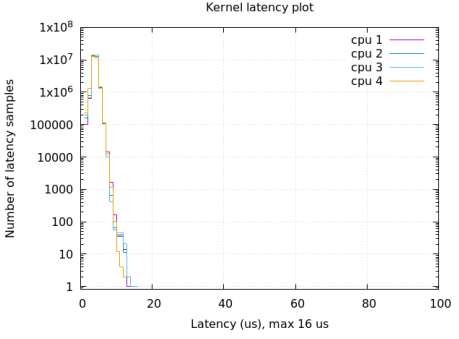


Figure: TSN talker and listener setup

PREEMPT_RT Task Scheduling Latency



(a) RT Tests LS1028A



(b) RT Tests PC

Linux Time Synchronization Precision

Method	Minimum	Maximum
Hardware PHY	10 ns	50 ns
Hardware MAC	10 ns	100 ns
Software	10 µs	100 µs

Table: Time Synchronization Results

- **Hardware PHY**: Hardware generated Time Stamps at PHY level
- **Hardware MAC**: Hardware generated Time Stamps at MAC level
- **Software**: Software generated Time Stamps at driver level

Linux TAPRIO Implementation Latency

Method	Minimum [µs]	Maximum [µs]
Hardware	1	5
Software	23	unbounded
LX Software	21	72

Table: TAPRIO Latency Results

- **Hardware**: Hardware offloaded TAPRIO implementation
- **Software**: Standard Software TAPRIO implementation
- **LX Software**: Linutronix optimized Software TAPRIO implementation

Linux Application Latency: User Space to User Space

Method	Min. [µs]	Max. [µs]	AVG [µs]
XDP/Tx Time	4	53	8
Tx Time	8	53	8
XDP	21	102	22
Regular	21	100	22

Table: Application End to End Latency

- **XDP**: Optimized by eXpress Data Path
- **Tx Time**: Hardware assisted Time Triggered Send
- **Regular**: Standard Linux RAW sockets

Agenda

1. Introduction
2. Software TSN
   - Time Synchronization
   - Time Aware Shaping
   - Linux Network Stack
3. Evaluation
4. Conclusion

Conclusion

Conclusion

- Software TSN can be implemented by using Linux
- Modern Linux technologies such as XDP are helpful
- Hardware Time Stamping for PTP recommended
- Tweaking in regards to TAPRIO required
- Cycle times below 1 ms possible

Thank you for your attention.

Linutronix GmbH

kurt.kanzenbach@linutronix.de
Bahnhofstraße 3
88690 Uhldingen-Mühlhofen



References I

- Thomas Gleixner, “Evolution and current status of TSN in Linux,” in *Proceedings of TSN/A conference 2018*, Nov. 2018.
- MicroSys. (2019, Jun.) System on Module based on NXP QorIQ LS1028A CPU. [Online]. Available: https://microsys.de/fileadmin//user_upload/Images/News/miriac_MPX-LS1028A-DSC0274_small.jpg