Fraunhofer IPMS

Whitepaper

Time Sensitive Networking
A TSN Implementation on Intel FPGA Base

Introduction

Currently being developed by the IEEE, Time Sensitive Networking (TSN) is a set of standards aimed at increasing the degree of determinism in switched Ethernet networks according to existing IEEE 802.1 and 802.3 standards. In essence, TSN looks to realize Ethernet networks with:

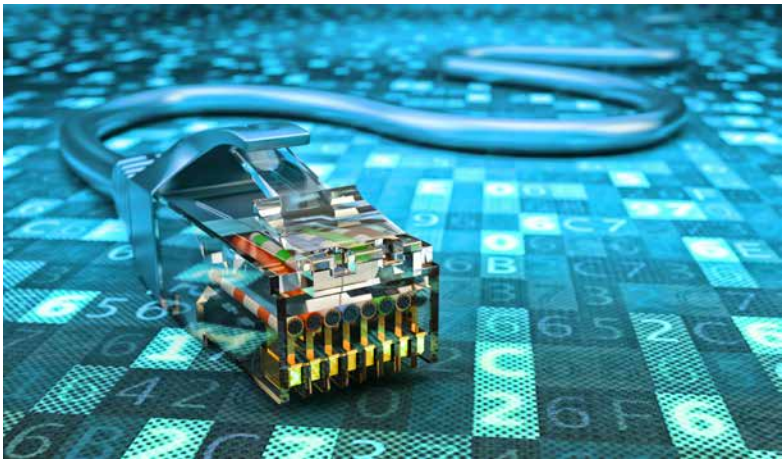
- guaranteed end-to-end latencies for real-time critical data traffic
- transmission of (time-) critical and uncritical data traffic over a converged network
- low packet losses
- low latency fluctuation (jitter)

Such Ethernet networks provide users the following advantages:

- Higher-level protocol layers can share a common, a converged network infrastructure
- Real-time control available for use outside of the operational technology (OT) area (process control, automation, etc.)
- No manufacturer dependencies relevant to real-time capable fieldbus systems

This whitepaper presents a simplified example to explain the benefits of TSN and demonstrates TSN implementation for endpoint applications on an Intel FPGA (Cyclone© V SoC) platform using the Fraunhofer IPMS TSN-EP IP-core. The Fraunhofer IPMS TSN-EP IP-core integrates a real-time MAC and supports the following TSN sub-standards:

- IEEE 802.1AS - Timing and Synchronization
- IEEE 802.1Qav - Forwarding and Queuing
- IEEE 802.1Qbv - Enhancements for Scheduled Traffic



TSN in Use – An Example

Sensors supply information needed for control and regulation in automated processes. These measured values are sent to control units for further processing. Calculated manipulated variables are transmitted to actuators to facilitate process intervention. Sensors are equipped with varying communication interfaces, depending on transmission requirements. Sensors with high power requirements and large volumes of data to be transmitted, such as those in camera systems, operate with network-compatible communication interfaces.

In the automation technology sub-area of robotics, interaction often occurs among locally distributed optical sensor systems such as time-of-flight cameras, robot systems actuators, and other actuating elements. Tasks such as positioning workpieces in a conveyor system are realized through these interactions. In such cases, data with high bandwidth requirements is transmitted in addition to the sensor data which, in most cases, must be available in real time. Therefore, in addition to other bus systems, a variety of Ethernet-based fieldbus systems that meet the various determinism and real-time data transmission requirements have emerged for this area of automation. Using multiple systems within a single network often leads to hardware incompatibilities which are currently solved with cost-intensive gateways in today's automated networks. TSN-enabled components allow different protocols to be used in parallel in a single network without the need to connect special hardware.

As shown in Figure 2, image processing data and sensor data are transmitted over a switched TSN network. This could be a camera used to detect position synchronized with a handling robot in order to carry out any position correction needed to move on to the next step in the process. Time-critical sensor data and best-effort data are simultaneously sent via the same network.

Time-critical data - Cyclic real time traffic

Used for real-time critical periodic controller-to-controller, controller-to-I/O, or device-to-device communication.

Non-critical Data – Best effort

In this example, the following best-effort data traffic is sent in the background:

- Simple network management protocol (SNMP)
- OPC UA status
- Web server
- Image and video data

TSN standards make real-time communication over a switched Ethernet network possible without the need for proprietary fieldbus systems and their required gateways.

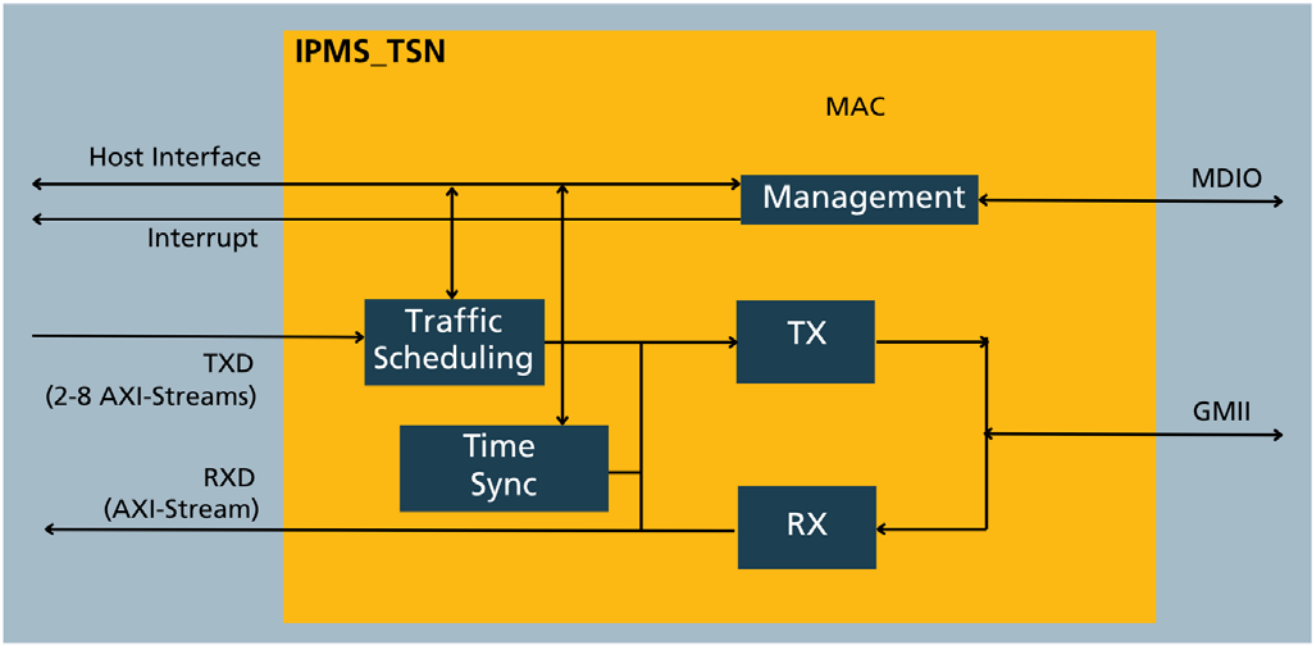


Fig. 1: Block diagram TSN-EP IP-core

TSN Implementation – Intel FPGA Basis

Offering a high degree of flexibility to support customer-specific solutions for small and medium quantities at a good price/performance ratio, field-programmable gate arrays (FPGA) provide a solid platform for TSN-based applications. FPGAs also provide I/O flexibility and the ability to integrate additional Ethernet protocols on a single device. This examples illustrates a TSN-EP IP-core implementation on an Intel FPGA from the Cyclone® series.

For the setup, a base board (NOVPEK™CVL) with an attached SODIMM module (NOVSOM®CVL) based on the Cyclone® series from Intel is used. The board is equipped with a KSZ9031RN PHY. The PHYs are connected to the TSN-EP IP-core via RGMII (reduced gigabit media independent interface) and MDIO (management data input/output). In comparison to GMII, RGMII saves resources by reducing the number of pins from 25 to 12. In contrast, RGMII uses double data rate (DDR) signaling, i.e. signal edges for both rising and falling clock edges.

The PHY is configured via the MDIO-interface of the TSN-EP IP-core. The Intel® SoC FPGA Embedded Development Suite (SoC EDS) design tool which also includes ARM Development Studio 5 is used. As an alternative to a hard processor core, a soft core such as NIOS II can also be used on the Cyclone® V GX Starter Kit.

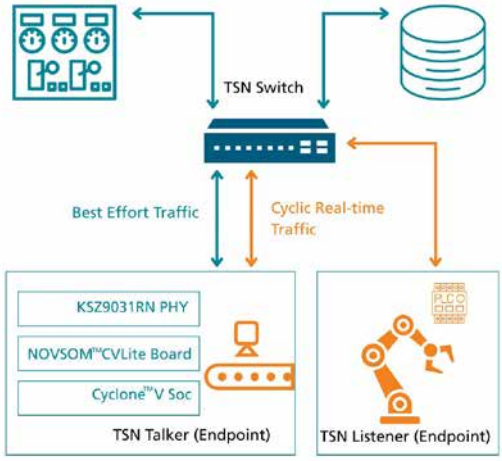


Fig. 2: Example of a switched TSN network

Hardware Description

The Fraunhofer IPMS TSN-EP IP-core is connected on the CPU side via an APB interface and an interrupt line. The core also provides an RX AXI stream and a configurable number of TX AXI streams, which are connected to the HPS of the Cyclone® V SoC via a DMA IP-core in this example (blue in Figure 3).

On the PHY side, the TSN-EP IP-core is connected via MII, GMII or RGMII. In order to work with the DDR registers of the I/O elements in the FPGA part of the Cyclone® V SoC, the ALTDDIO_IN or ALTDDIO_OUT IP from Intel are used. The TSN-EP IP integrates Intel's Quartus Prime® FPGA design software with the included QSys system integration tool that makes it easy to integrate Intel IP-core components (e.g. HPS, DMA, CDC-FIFOS).

With Quartus 18.1, the hardware description (HDL) and the QSYS project can be used to generate a bit file (.sof), which is then compressed as an RBF file (.rbf). Furthermore, handoff files (.xml, .c or .h) which describe the configuration of the HPS are generated.

Utilization of the FPGA part of the Cyclone® V SoC with integrated TSN-EP is shown in Figure 2. Orange corresponds to the utilization by the TSN-EP IP-core, blue to the used Intel IP cores such as the DMA controller or the ABP bus interface.

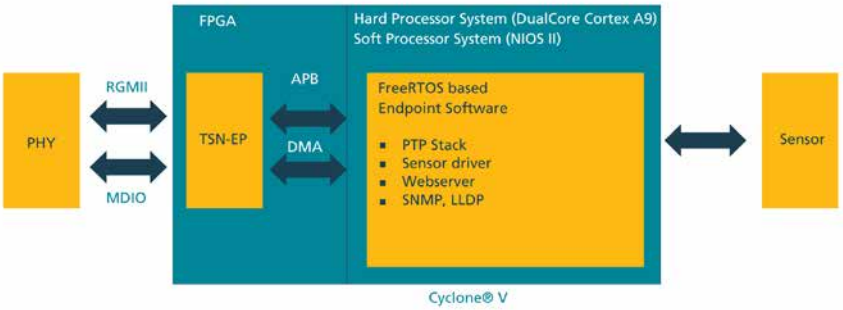


Fig. 3: Schematic layout FPGA Cyclone ® V SoC with integrated TSN -EP IP -core

Software Description

The program code uses FreeRTOS (10.2), the Intel / Altera HW libraries, and the IPMS PTP stack to control the TSN-EP IP-core. The FreeRTOS scheduler operates at a clock frequency of 1000 Hz and reserves 128 kbytes of memory for use as HEAP.

A Command Line Interface is provided via a UART interface to control the system. DRAM generates an one-megabyte dynamic data buffer for Ethernet data packets. Software on the HPS is executed directly from the DRAM, whereby the available memory of a gigabyte is by far not used.

An .ELF file is created from the compiled program code and then converted into a binary format (.bin). Software is started using U-Boot, the boot loader.

Summary

TSN allows the transmission of critical real-time and non-critical data traffic over a converged Ethernet network, meeting various jitter, bandwidth and latency requirements. FPGAs offer a wide-range of possibilities for quickly bringing customer-specific TSN-capable solutions into productive use.

	Resource	Usage	%
1	Logic utilization (ALMs needed / total ALMs on device)	4,534 / 41,910	11 %
2	ALMs needed [=A-B+C]	4,534	
1	[A] ALMs used in final placement [=a+b+c+d]	5,306 / 41,910	13 %
2	[B] Estimate of ALMs recoverable by dense packing	831 / 41,910	2 %
3	[C] Estimate of ALMs unavailable [=a+b+c+d]	59 / 41,910	< 1 %
4	Difficulty packing design	Low	
5			
6	Total LABs: partially or completely used	689 / 4,191	16 %
7			
8	Combinational ALUT usage for logic	7,138	
9	Combinational ALUT usage for route-throughs	1,307	
10			
11	Dedicated logic registers	6,923	
12			
13	Virtual pins	0	
14	I/O pins	132 / 499	26 %
15	I/O registers	246	
16			
17	Hard processor system peripheral utilization		
18			
19	M10K blocks	51 / 553	9 %
20	Total MLAB memory bits	0	
21	Total block memory bits	316,136 / 5,662,720	6 %
22	Total block memory implementation bits	522,240 / 5,662,720	9 %
23			
24	Total DSP Blocks	0 / 112	0 %
25			
26	Fractional PLLs	1 / 6	17 %

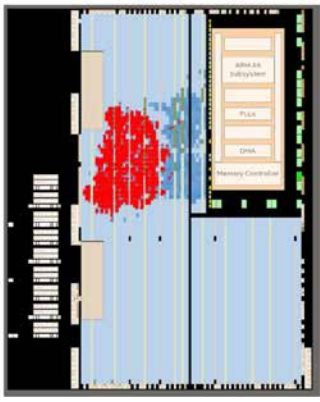


Fig. 4: Utilization of FPGA part of Cy clone ® V SoC with integrated TSN-EP IP Core

Contact

Stephan Kube
+49 351 88 23 - 1211
stephan.kube@ipms.fraunhofer.de

Fraunhofer IPMS
Maria-Reiche-Str. 2
01109 Dresden, Germany
www.ipms.fraunhofer.de

ABOUT FRAUNHOFER IPMS

The Fraunhofer Institute for Photonic Microsystems IPMS stands for applied research and development in the fields of industrial manufacturing, medical technology and improved quality of life. Our research focuses on miniaturized sensors and actuators, integrated circuits, wireless and wired data communication and customized MEMS systems.

Fraunhofer IPMS has many years of experience in the design and development of IP cores for automotive communication and has a family of TSN IP cores. Many users around the world use Fraunhofer IPMS IP cores in the automotive, aerospace and automation industries, among others. The multidisciplinary IP design team at Fraunhofer IPMS, with expertise ranging from computer architectures, network structures and RTL design to the implementation of electronic systems, is also available as a competent development partner for application-specific adaptations of IP cores and their integration into complex network architectures.