

TTE Switch Lab Space

User Manual

Document type:	User Manual
Document number:	D-863-M-05-001
Document version:	0.3.4
Date:	2019-07-24
Status:	Draft
Product version:	n/a

TTTech Computertechnik AG

Schoenbrunner Str. 7, A-1040 Vienna, Austria, Tel. +43 1 585 34 34 – 0, Fax +43 1 585 34 34 – 90, office@tttech.com

No part of the document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the written permission of TTTech Computertechnik AG. Company or product names mentioned in this document may be trademarks or registered trademarks of their respective holders. TTTech Computertechnik AG undertakes no further obligation or relation to this document.

Contents

List of Figures	4
List of Tables	5
List of Code Listings	6
Legal Disclaimer	7
Regulatory Information	8
Typographic Conventions	9
Important Notice	10
Hardware Setup	11
1 Introduction	12
2 Overview of the TTE-Switch Lab Space	13
2.1 Identification	13
2.2 External Interfaces	13
2.3 Functional Features	14
2.3.1 TTEthernet Implementation	14
2.3.2 ARINC 664p7 Implementation	15
2.4 Physical Specifications	15
2.5 Power Supply	15
2.6 Environmental Operating Ranges	15
2.7 Standards Compliance	15
3 Functional Description	16
3.1 Block Diagram	16
3.2 Interfaces	16
3.2.1 Networking Interfaces	16
3.2.1.1 Ethernet Ports	16
3.2.2 Debug and Maintenance Interfaces	17
3.2.2.1 GPIO Interface	17
3.2.2.2 RS-422 Interface	18
3.2.2.3 SpaceWire Interface	18
3.2.2.4 USB Interface	18
3.2.2.5 JTAG Interface	18
3.2.3 Human-Machine Interfaces	19
3.2.3.1 Power Button	19
3.2.3.2 Reset Button	19
3.2.3.3 Status LEDs	19
3.3 Technical Data	19
3.3.1 Cooling	19
3.3.2 Size	20
3.3.3 Weight	20
3.3.4 Electrical Characteristics	20
4 Getting Started	21

5	State Machine	22
5.1	State Machine Overview	22
5.2	Operational Modes of the Firmware	22
5.2.1	INIT Mode	22
5.2.2	ADDRESS Mode	23
5.2.3	CONTROLLER Mode	23
5.2.4	RECONFIGURE Mode	24
5.2.5	ERROR Mode	25
6	Configuring the TTE-Switch Lab Space	26
6.1	Creating a Switch Configuration	27
6.1.1	Using the TTE-Tools for Device Configuration	27
6.1.1.1	Configuring the Switch	29
6.1.1.2	Configuring the Internal End System	30
6.1.1.3	Configuring Best-Effort Routes	31
6.1.2	Using a Python Script to Configure the Network	32
6.2	Uploading the Switch Configuration	32
6.2.1	Preparing Files for an Upload	33
6.2.2	Uploading TFTP Files	33
6.3	Downloading the Switch Configuration	34
6.4	Recovering from Errors	34
7	Diagnosis	36
8	Troubleshooting	37
8.1	Basic Checks	37
8.2	SpaceWire Hot-Plugging Issues	37
8.3	The <code>arp</code> Command Fails	37
8.4	The <code>tftp</code> Command Does Not Work	37
8.5	No Link Is Established	37
8.6	Customer Support Information	38
Appendix A Default Configuration		39
Appendix B MIBs		39
B.1	TT-MIB	39
B.2	TT-SWE-VNV-MIB	39
B.3	TTE-ES-1-7-MIB	47
Glossary		55
Bibliography		57
Index		58

List of Figures

1	The TTE-Switch Lab Space	13
2	Product identification label	13
3	Front view of the TTE-Switch Lab Space	13
4	Rear view of the TTE-Switch Lab Space	14
5	Block diagram of the TTE-Switch Lab Space	16
6	GPIO	17
7	The state machine	22
8	Schematic overview of dataloading	26
9	TTE-Tools data flow overview	28

List of Tables

1	Ports of the TTE-Switch Lab Space	17
2	RJ45 copper port LED status	17
3	Pin assignment of the GPIO interface	18
4	Pin assignment of the RS-422 interface	18
5	Front panel status LEDs of the switch	19
6	List of files in the flash memory	32
7	Recovery from errors	35

List of Code Listings

1	Mapping the ports	29
2	BE management configuration	30
3	First part of the BE route configuration	31
4	Second part of the BE route configuration	31
5	Default configuration settings	39
6	TT-MIB definitions	39
7	TT-SWE-VNV-MIB definitions	39
8	TTE-ES-1-7-MIB definitions	47

Legal Disclaimer

THE INFORMATION GIVEN IN THIS USER MANUAL IS GIVEN AS A SUPPORT FOR THE USAGE OF THE TTE-SWITCH LAB SPACE ONLY AND SHALL NOT BE REGARDED AS ANY DESCRIPTION OR WARRANTY OF A CERTAIN FUNCTIONALITY, CONDITION OR QUALITY OF THE TTE-SWITCH LAB SPACE. THE RECIPIENT OF THIS USER MANUAL MUST VERIFY ANY FUNCTION DESCRIBED HEREIN IN THE REAL APPLICATION.

TTTECH COMPUTERTECHNIK AG PROVIDES THE USER MANUAL FOR THE TTE-SWITCH LAB SPACE "AS IS" AND WITH ALL FAULTS AND HEREBY DISCLAIMS ALL WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OR COMPLETENESS, OR OF RESULTS TO THE EXTENT PERMITTED BY APPLICABLE LAW. THE ENTIRE RISK, AS TO THE QUALITY, USE OR PERFORMANCE OF THE USER MANUAL, REMAINS WITH THE RECIPIENT. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW TTTECH COMPUTERTECHNIK AG SHALL IN NO EVENT BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOSS OF DATA, DATA BEING RENDERED INACCURATE, BUSINESS INTERRUPTION OR ANY OTHER PECUNIARY OR OTHER LOSS WHATSOEVER) ARISING OUT OF THE USE OR INABILITY TO USE THE USER MANUAL, EVEN IF TTTECH COMPUTERTECHNIK AG HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TTTECH COMPUTERTECHNIK AG MAKES NO WARRANTY OF ITS PRODUCTS, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW DISCLAIMS ALL LIABILITIES OR DAMAGES RESULTING FROM OR ARISING OUT OF THE APPLICATION OR USE OF THESE PRODUCTS.

Regulatory Information

NOTE: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/ TV technician for help.

NOTICE:

Changes or modifications made to this equipment not expressly approved by (TTTech Computertechnik AG) may void the FCC authorization to operate this equipment.

Standards Compliance

The TTE-Switch Lab Space conforms to the following standards:

- **Directive 2014/30/EU (EMC Directive)**
 - EN 55032:2015 – *Electromagnetic compatibility of multimedia equipment – Emission Requirements*
 - EN 55035:2016 – *Electromagnetic compatibility of multimedia equipment – Immunity Requirements*
- **Directive 2014/35/EU (Low Voltage Directive)**
 - EN 62368-1:2014 – *Audio/video, information and communication technology equipment – Part 1: Safety requirements*
- **Directive 2011/65/EU (ROHS)**
 - EN 50581:2012 – *Technical documentation for the assessment of electrical and electronic products with respect to the restriction of hazardous substances*
- **Directive 2012/19/EU (WEEE)**
- **FCC Title 47, Part 15B**
 - *Unintentional Radiators*

Typographic Conventions

Element	Typographic format	Example
Source code examples, parameters, file names, directory and path names	Courier new	tt_start_OS; C:\TTTech\TTPos\
GUI menu names and entries	Boldface	Click File and then Open to open the ...dialog.
Keyboard keys, GUI buttons	Keystroke symbols	Click OK or press Enter to confirm the settings made.
Command-line interface commands	Terminal	ls -l /dev/null

Important Notice

<i>General</i>	The TTE-Switch Lab Space is a non-repair item except for the fuse.
----------------	--

<i>Handling</i>	Only carry the TTE-Switch Lab Space by holding its carrying handles.
-----------------	--

<i>Modifying</i>	The TTE-Switch Lab Space shall not be opened and the screws shall not be loosened.
------------------	--

Hardware Setup

Required Hardware

Several components are needed to get the TTE-Switch Lab Space installed in a rack. The following list shows which of these components are included with the switch and which are not.

- Screws and fasteners for rack mounting – *tbd*
- Ethernet cables – *not included*

Installation Instructions

Compatible Hardware

1 Introduction

The TTE-Switch Lab Space is a 25x Ethernet port switch based on the TTE-Controller HiRel (TT6802-2-SW-B) and serves as a development platform for TTEthernet.

TTEthernet is a fault-tolerant real-time communication protocol for safety-relevant systems that makes it possible to conveniently configure the deterministic processing of **critical** Ethernet traffic (time-triggered, ARINC 664 P7 [1]) and **non-critical**, standard Ethernet traffic (IEEE 802.3 [4]) in one physical infrastructure.

Switching Function

The TTE-Switch Lab Space is a high-performance deterministic Ethernet switch and provides 25 Ethernet ports in total:

- 6x 100/1000 Mbit/s ports
- 19x 100 Mbit/s ports

These ports can also be used to monitor traffic. The TTE-Switch Lab Space has built-in mechanisms for traffic policing and fault isolation.

Virtual Links and Protocol Support

The TTE-Switch Lab Space allows configuring up to 4096 virtual links (VL). Virtual links can be configured with 8 priorities and a **bandwidth allocation gap (BAG)** of 0.5 ms to 1600 ms. The network configuration is stored in the non-volatile memory of the switch (256 Mbit). It is optionally possible to configure IEEE 802.1Q VLANs. Profiled IP/UDP, redundancy management, and traffic shaping are implemented in hardware.

Data Loading and Diagnostics

The TTE-Switch Lab Space features an integrated CPU for management. Switch management relies on SNMP. Data loading is possible through TFTP.

2 Overview of the TTE-Switch Lab Space

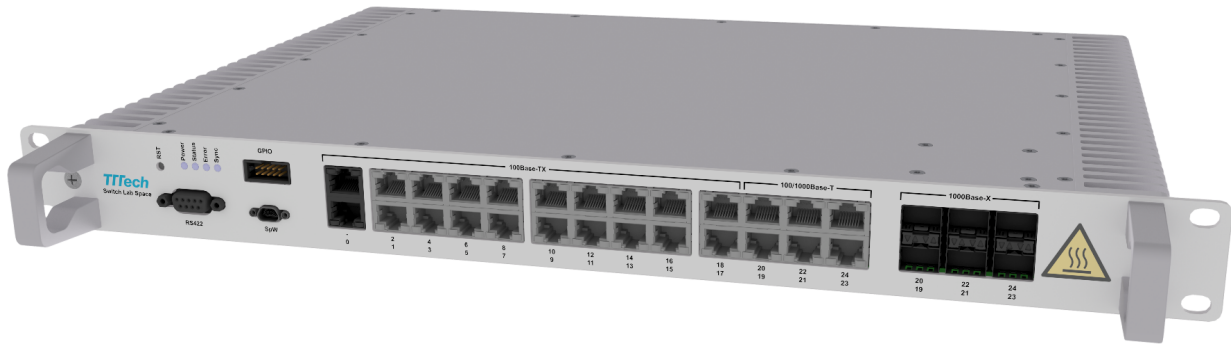


Figure 1: The TTE-Switch Lab Space

2.1 Identification

Each TTE-Switch Lab Space has a label at its rear that provides the following information:

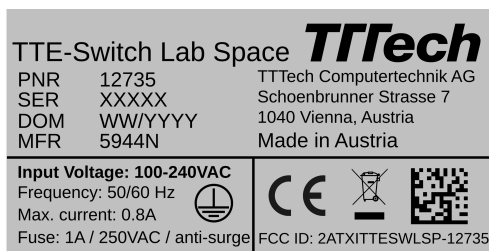


Figure 2: Product identification label

- Product Name: TTE-Switch Lab Space
- PNR: Part Number
- SER: Serial Number
- DOM: Date of Manufacturing
- MFR: Manufacturer
- A data matrix with the product serial number on the bottom right.

2.2 External Interfaces

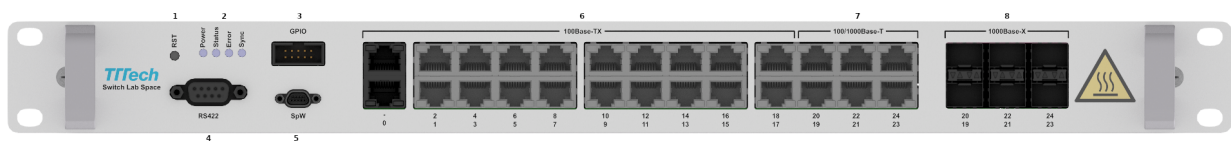


Figure 3: Front view of the TTE-Switch Lab Space

1. Reset
2. 4 status LEDs on the front panel (Power/Status/Error/Sync)
3. GPIO
4. RS-422 serial interface (can be used for debugging and maintenance)
5. SpaceWire (can be used for debugging and maintenance)
6. 19 x 100 Mbit/s full-duplex Ethernet ports (100Base-TX via RJ45)

7. 6 x 100/1000 Mbit/s Ethernet ports (1000Base-T via RJ45)
8. 6 x 100/1000 Mbit/s Ethernet ports (1000Base-X via SFP)

NOTE

The Ethernet port labeled – is not functional.



Figure 4: Rear view of the TTE-Switch Lab Space

9. USB connector (provides access to the DSU/UART1 interface)
10. JTAG connector for factory testing and programming
11. IEC 60320-1 C14 type power connector and a power button. The switch has an electrical fuse which protects the external power rail so that a failing switch does not cause the system power supply rail to fail. Both the power connector and the power button serve as disconnect devices.

NOTE

Only connect the TTE-Switch Lab Space to a socket-outlet that features a grounding wire.

2.3 Functional Features

The TTE-Switch Lab Space is intended as a development platform for TTEthernet and provides the following features:

- A high-speed deterministic network
- Rate-constrained traffic (fully compliant with ARINC 664p7 [1])
- Time-triggered traffic (SAE AS6802 [6])
- Best-effort traffic (IEEE 802.3-2005 [4])

The TTE-Switch Lab Space is mains-operated and is developed as rack-mounted electronic ground support equipment (EGSE) for TTEthernet development and as a functionally-representative demonstrator of space-grade components of TTTech Computertechnik AG's space products (i.e. Switch Controller Space and End System Controller Space).

Apart from the standard switch functions (including best-effort, rate-constrained and time-triggered Ethernet), the switch also includes support for the following utilizations:

- Access to a TTEthernet End System via SpaceWire using RMAP
- The possibility to access the UART0 interface for debugging purposes
- The ability to use physical layers other than the 1000Base-T via SFP module slots

2.3.1 TTEthernet Implementation

- 8 sub-schedules
- 8 clock synchronization masters
- 4096 virtual links
- Store-and-forward switch architecture

2.3.2 ARINC 664p7 Implementation

- Policing, filtering, and switching engine for bandwidth control and traffic prioritization
- Integrity and error checking of frames
- 4096 virtual links with up to 8 priorities, with restrictions of their associated ports
- 4096 BAGs
- BAGs freely configurable from 0.5 ms to 1600 ms
- BAG configuration granularity of 100 μ s
- Jitter and BAG resolution of 8 μ s
- Support for ICMP (ping), SNMPv1 and TFTP dataloading
- Configuration data can be programmed through TFTP

Additional packaging material and additionally packed accessories, e.g. a mains power cord, may be part of the delivered product but are not considered to be part of the system in this document.

NOTE

2.4 Physical Specifications

- 19-inch rack housing: 1 height unit
- Size: 346 mm x 483 mm x 44 mm

2.5 Power Supply

- AC voltage: 100-240 V, 50 to 60 Hz

2.6 Environmental Operating Ranges

- Operating temperature: 0 °C to +60 °C
- Storage temperature: -55 °C to +85 °C

2.7 Standards Compliance

- **ARINC Specification 664P7-1**: The switch is fully compliant to ARINC 664 part 7 (deterministic Ethernet networking) [1].
- **IEEE 802.3™-2005** (switching, flow control) [4].
- **IEEE 802.1Q™-2011** [3].
- **IEEE 1588-2008** [5]: The switch supports the IEEE 1588 end-to-end transparent clock mode. The clock of the switch is not synchronized to the IEEE 1588 Master Clock.
- **SAE AS 6802** [6]: The switch supports the SAE AS 6802 network synchronization and start-up mechanism (fault-tolerant TTEthernet clock synchronization protocol).

3 Functional Description

This section describes the functionality of the TTE-Switch Lab Space.

3.1 Block Diagram

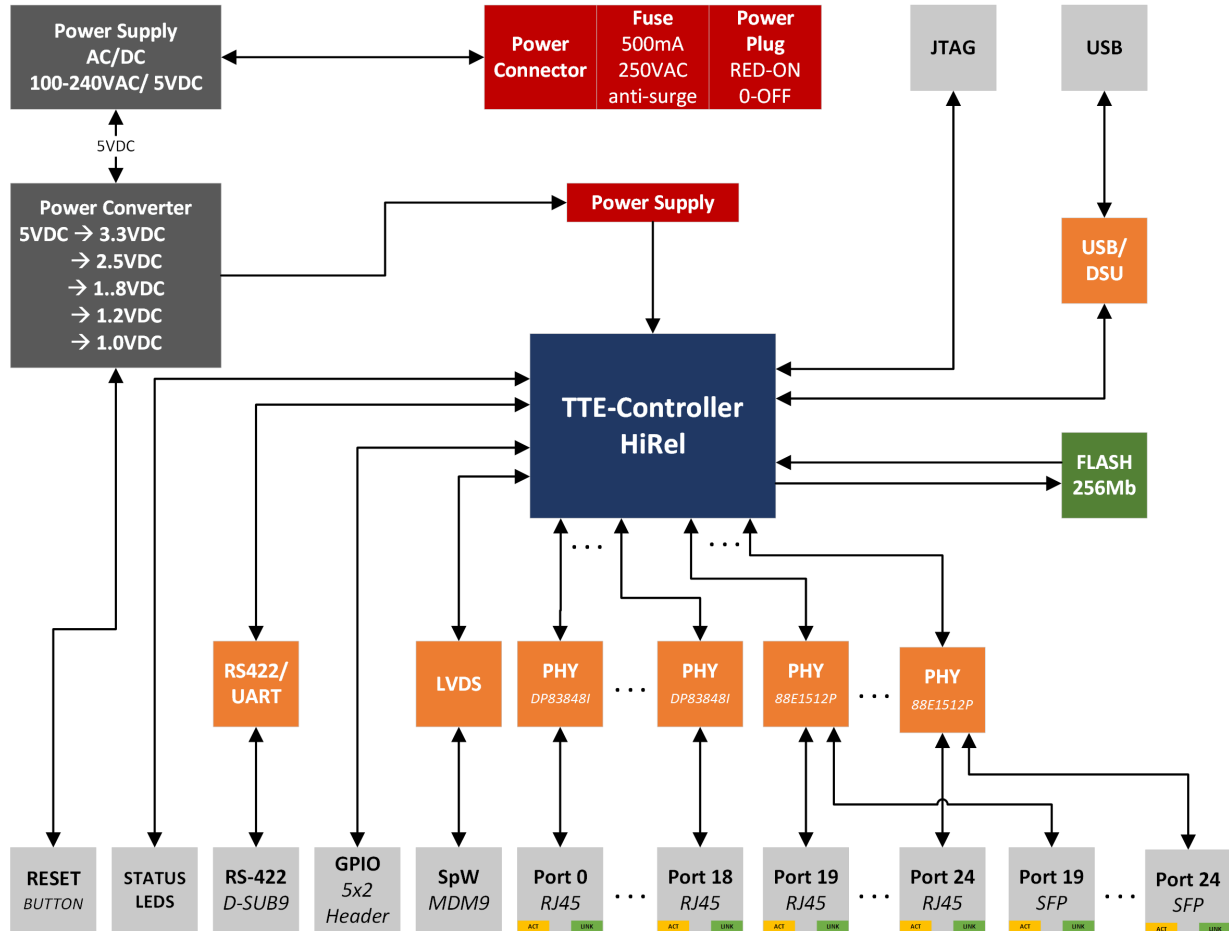


Figure 5: Block diagram of the TTE-Switch Lab Space

3.2 Interfaces

The TTE-Switch Lab Space features several interfaces that can be divided into networking interfaces, debug and maintenance interfaces, and human-machine interfaces. These interfaces are described in the following sections.

3.2.1 Networking Interfaces

This section describes the networking interfaces of the TTE-Switch Lab Space.

3.2.1.1 Ethernet Ports

Each Ethernet port (see Figure 3 on page 13) features two integrated LEDs: a **Link** and an **Activity** LED.

Port	Description
-	This Ethernet port is not functional.
0...18	General purpose, full-duplex RJ45 Ethernet copper ports configurable for 100Base-TX.
19...24	General purpose, full-duplex RJ45 Ethernet copper ports configurable for 100/1000Base-T.
19...24	General purpose SFP ports configurable for 1000Base-X, suitable to operate 1000Base-X-compliant SFP modules.

Table 1: Ports of the TTE-Switch Lab Space

Each of the 1000Base Ethernet ports can either be used as RJ45 port or as SFP port, but not as an RJ45 and SFP port at the same time.

NOTE

Hot-plugging SFP modules (i.e., inserting a module during runtime) is only supported if the switch is configured to use the 1000Base-X ports. Otherwise, a reset is required to configure SFP modules.

NOTE

Port LED	Description
	The Link LED illuminates steadily when a link is established with the connected internal end system.
	The Activity LED flashes when communication activity is detected (frame transmission or reception).
	No link and no activity detected.

Table 2: RJ45 copper port LED status

3.2.2 Debug and Maintenance Interfaces

The TTE-Switch Lab Space interfaces described in this section are not meant for static cable installations and serve for debug and maintenance purposes only.

3.2.2.1 GPIO Interface

The GPIO interface is used to access the GPIO interface of the TTE-Controller HiRel. This interface is used for general purposes as can be seen in Table 3 on the next page.

The GPIO pins (see Figure 6 on this page) are used to select which configuration will be loaded at boot.

When using a jumper between the pins 9 and 10, the default configuration is loaded. If no jumper is used, the user configuration is loaded.



Figure 6: GPIO

Pin No.	Feature	Description	Default
1	IRQ0	Internal end system external trigger 0.	Output high
3	IRQ1	Internal end system external trigger 1.	Output high
5	IRQ2	Integration cycle start interrupt.	Output high
7	NA	NA	Input

Pin No.	Feature	Description	Default
9	Default/user configuration	When connected to GND, the default configuration is loaded. Otherwise, the user configuration is loaded.	Input
2,4,6	+3.3VDC	Pins connected to +3.3VDC.	–
8,10	GND	Pins connected to ground.	–

Table 3: Pin assignment of the GPIO interface

3.2.2.2 RS-422 Interface

The RS-422 interface (see Figure 3 on page 13) is used to access the UART0 interface of the TTE-Controller ASIC. This interface is used for debugging.

Connector type: 9-pin D-sub female connector. The control interface conforms with the RS-422 standards and enables connection to one of the COM ports of a PC using a serial 1:1 adapter cable (DB-9 connector to DB-9 socket). The settings are 115200 baud, 8 data bits, one stop bit, no handshake, no parity.

RS-422 Constraints

The cable length must not exceed 3 m.

Pin No.	Signal Name	Signal Type
1	RXD–	Receive data output
2	RXD+	Receive data output
3	TXD+	Transmit data output
4	TXD–	Transmit data output
5	GND	Ground
6	CTS–	Clear to send
7	CTS+	Clear to send
8	RTS+	Request to send
9	RTS–	Request to send

Table 4: Pin assignment of the RS-422 interface

3.2.2.3 SpaceWire Interface

The SpaceWire interface (see Figure 3 on page 13) is a host interface that is used to read and/or write all registers of the ASIC. The internal end system can be access by using the correct SpaceWire driver.

3.2.2.4 USB Interface

The TTE-Switch Lab Space provides access to the DSU/UART1 interface of the TTE-Controller HiRel through the USB interface (see Figure 4 on page 14). The USB interface is used for factory firmware programming.

3.2.2.5 JTAG Interface

The TTE-Switch Lab Space features a standard JTAG interface (see Figure 4 on page 14) on the back panel. The JTAG connector is used for factory testing.

3.2.3 Human-Machine Interfaces

This section describes the human-machine interfaces of the TTE-Switch Lab Space.

3.2.3.1 Power Button

The power button is located right next to the power connector on the rear panel of the switch (see Figure 4 on page 14). To turn on the power, press the power button into the I position.

The power button also serves as a disconnect device.

NOTE

3.2.3.2 Reset Button

The reset button (see Figure 3 on page 13) is recessed and can only be pushed with a thin tool, such as a pencil. This prevents that the reset button is pushed by accident. A long press of the reset button of approximately 10 seconds (the Status LED will light up green) will load the default configuration.

3.2.3.3 Status LEDs

In operation, the front panel LEDs (see Figure 3 on page 13) of the TTE-Switch Lab Space indicate the following status information:

Power	Status	Sync	Error	Description
				The <i>Power</i> LED lights up green when the power supply of the switch is turned on.
				The firmware is running without active TTEthernet synchronization. If the Default mode was selected, then the default configuration is loaded and the Status LED lights up green. If the User mode was selected, then the user configuration is loaded (see Table 3 on the facing page) and the Status LED lights up green only if the user configuration file is valid.
				The firmware is running with active TTEthernet synchronization. If the Default mode was selected, then the default configuration is loaded. If the User mode was selected, then the user configuration is loaded. The Sync and Status LEDs light up green.
				The firmware is running without active TTEthernet synchronization. No user configuration is loaded or the uploaded user configuration file has an error. The Error LED lights up red.
				The debug support unit is active (e.g., during a firmware upload)"
				The <i>Power</i> LED lights up red if the PCB registers excess voltage.

Table 5: Front panel status LEDs of the switch

3.3 Technical Data

3.3.1 Cooling

The switch is constructed for passive cooling only and can withstand an operational ambient temperature ranging from 0 °C to +60 °C.

When the switch is used in environmental conditions with temperatures exceeding +45 °C, it is mandatory to mount the switch in a suitable rack to avoid the risk of burns or other personal injury.

NOTE

3.3.2 Size

The TTE-Switch Lab Space complies with the IEC 60297 [2] standard and comes with a housing that takes up one rack unit (1U). Size: 346 mm x 483 mm x 44 mm.

3.3.3 Weight

The TTE-Switch Lab Space weighs a maximum of 5 kg.

3.3.4 Electrical Characteristics

Power Supply The TTE-Switch Lab Space supports the following input voltages:

- Voltage: 100 VAC - 240 VAC
- Frequency: 60 Hz - 50 Hz

Power Consumption The TTE-Switch Lab Space consumes a maximum of 45 W.

Grounding

- For ESD protection, Signal Ground (GND) is connected with a 5 x 1 nF and 2 x 10 M Ω to CHASSIS-GND.
- Signal Ground (GND) is AC/DC-coupled with CHASSISGND, which is connected to the housing and the metal-plated connectors (Ethernet RJ45, SFP Cages, D-SUB DE-9), which can be accessed from the outside.

4 Getting Started

The Python scripts mentioned in this and subsequent sections require Python 2.7. Depending on the Python installation on the host system, the Python command will be either `python` or `python2`. The examples in this document use `python2` for consistency. The Python scripts delivered to the customer were tested on Ubuntu version 14.04.

NOTE

This section describes how to get started with the TTE-Switch Lab Space in 4 steps.

1. Make sure that the switch is connected to a power outlet and switched off.
2. Use a jumper and connect the pins according to Table 3 on page 18 in order to select the default configuration of the switch.
3. Power on the switch.

The Power LED will turn on (see Table 5 on page 19). When the switch has successfully finished its start-up, the Power LED and the Status LED will remain green. This means the switch is running the default configuration (see Table 5 on page 19).

4. Connect a host PC directly to one of the 25 front panel Ethernet ports using a standard Cat5 or Cat5e patch cable.
 - Set the IP address of the host PC to the same subnet as the switch, for example to 10.10.10.20.
 - Set the subnet mask of the host PC to 255.255.0.0.
 - Set the transmission speed to 100 Mbps for port 0–18 and to 1000 Mbps for ports 19–24.
 - Open a **Command Prompt** to verify that the switch is operational.
 - Type `arp -s 10.10.10.10 A6-A7-A8-00-01-20 10.10.10.20`
10.10.10.10 is the default IP address, A6-A7-A8-00-01-20 is the default MAC address. The host PC has to be on the same subnet as the TTE-Switch Lab Space. See Section 8.3 on page 37 if the `arp` command does not work.
 - Type `ping 10.10.10.10` and press **Enter** (see Section A on page 39).
If no data packet is lost, the ping was successful (see Listing Listing 1 on page 29).

```
Pinging 1 0 . 1 0 . 1 0 . 1 0 with 32 bytes of data :
Reply from 1 0 . 1 0 . 1 0 . 1 0 : bytes =32 time =12 ms TTL =128
Reply from 1 0 . 1 0 . 1 0 . 1 0 : bytes =32 time =9 ms TTL =128
Reply from 1 0 . 1 0 . 1 0 . 1 0 : bytes =32 time =7 ms TTL =128
Reply from 1 0 . 1 0 . 1 0 . 1 0 : bytes =32 time =7 ms TTL =128

Ping statistics for 1 0 . 1 0 . 1 0 . 1 0 :
Packets : Sent = 4 , Received = 4 , Lost = 0 (0% loss) ,
Approximate round trip times in milliseconds :
Minimum = 7 ms , Maximum = 12 ms , Average = 8 ms
```

See Section A on page 39 for the functionality of the Default Configuration.

NOTE

5 State Machine

5.1 State Machine Overview

The embedded software of the TTE-Switch Lab Space consists of two main parts:

- Bootloader
- Firmware

The bootloader is stored permanently in the internal ROM of the TTE-Controller HiRel. It cannot be modified or updated. The bootloader initializes the CPU and loads the firmware image. The bootloader then verifies the integrity of this firmware image through a CRC check. If the CRC is successful, the bootloader executes the firmware image.

The firmware then starts in INIT mode. In this mode, it initializes the rest of the system, executes the Built-In Self-Tests (BISTs), and verifies the integrity of all configuration images in the attached flash device.

The runtime system of the firmware provides several operational modes. Each operational mode consists of specific runtime system tasks that can only be executed in this mode. Several other tasks are shared by several modes. See Figure 7 on this page for a visual representation of the embedded software and its process sequence.

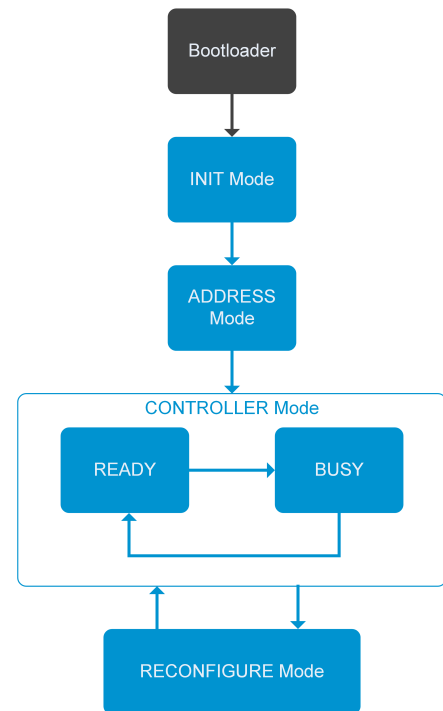


Figure 7: The state machine

5.2 Operational Modes of the Firmware

The firmware has the following operational modes: INIT, ADDRESS, CONTROLLER, RECONFIGURE, and ERROR.

5.2.1 INIT Mode

In this mode, the firmware starts and initializes the rest of the system before it executes the BISTs and verifies the configuration images.

Identification

- Shared memory.Operation mode = 0x0000 14E5
- No response on the network
- Shared memory.Lifesign value is undefined

Operation

- The firmware initializes all peripherals of the TTE-Switch Lab Space and the PHYs.
- The firmware loads the configuration of the internal end system that was selected through the GPIO interface (see Section 3.2.2.1 on page 17) or a long-press of the reset button (see Section 3.2.3.2 on page 19).
- The firmware configuration files are loaded.
- The Power-up Built-In Tests (PBITs) are performed.

User Operations

None.

Transitions

- If successful, the firmware transitions to ADDRESS mode.
- In the case of an error, the firmware transitions to ERROR mode.

Remarks

There is no interaction with the user on the network.

5.2.2 ADDRESS Mode

In this mode, the firmware sets the internal software network stack to the data provided by the network configuration.

Identification

- Shared memory.Operation mode = 0x0000 14FB
- No firmware communication on the network
- Shared memory.Lifesign value increments every 5 ms

Operation

- Set up the network stack.
- Set the network address (MAC and IP) to a fixed value.
- Continuous internal end system diagnostics.
- Continuous switch diagnostics.
- Continuous Built-In Tests (CBITs) are performed.

User Operations

None.

Transitions

- If successful, the firmware transitions to CONTROLLER mode.
- In the case of an error, the firmware transitions to ERROR mode.

Remarks

There is no interaction with the user on the network.

5.2.3 CONTROLLER Mode

In this mode, the user can interact with the TTE-Switch Lab Space. This includes the upload (Section 6.2 on page 32) and download (Section 6.3 on page 34) of files, status information requests (ICMP echo requests, SNMP get requests), and the acceptance of user commands (SNMP set requests). With a user-specific command, the user can switch the operational mode of the firmware to the RECONFIGURE mode.

Identification

- Shared memory.Operation mode = 0x0000 14FC
- The firmware responds on the network to ICMP, SNMP, and TFTP requests.
- The firmware communicates with best-effort traffic only.
- Shared memory.Lifesign value increments every 5 ms

Operation

- SNMP
- ICMP ping
- TFTP
- Continuous internal end system diagnostics
- Continuous switch diagnostics
- Continuous Built-In Tests (CBITs) are performed

User Operations

- Telemetry (SNMP)
- Network management (ICMP ping)
- Commands (SNMP set requests)
- File upload (TFTP)
- File download (TFTP)

Transitions

- At the user's request the firmware transitions to RECONFIGURE mode.
- In the case of an error, the firmware transitions to ERROR mode.

Remarks

None.

5.2.4 RECONFIGURE Mode

In this mode, the firmware configures the internal end system and the switch engine with the configuration selected through the GPIO interface (see Section 3.2.2.1 on page 17) or a long-press of the reset button (see Section 3.2.3.2 on page 19). Once the reconfiguration is complete, the firmware will switch back to CONTROLLER mode.

Identification

- Shared memory.Operation mode = 0x0000 1502
- There is no firmware communication on the network.
- Shared memory.Lifesign value increments every 5 ms

Operation

- Reconfigure the internal end system and switch engine with the user's configuration file.
- The firmware configures the host interrupts according to the firmware configuration file.
- The flash memory is disabled.

User Operations

None.

Transitions

- If the reconfiguration is successful, the firmware transitions to CONTROLLER mode.
- In the case of an error, the firmware transitions to ERROR mode.

Remarks

None.

5.2.5 ERROR Mode

This operational mode is triggered when a non-recoverable error occurs. The firmware disables the CPU and remains fail-silent (i.e., the CPU does not operate on an external interface any more). Also, the Error LED will light up.

Identification

- Shared memory.Operation mode = 0x0000 151B
- There is no firmware communication on the network.
- The shared memory.Lifesign does not advance over time.

Operation

- The firmware is turned off.
- The firmware is fail-silent on the network.

User Operations

The user needs to reset the TTE-Switch Lab Space.

Transitions

None.

Remarks

None.

6 Configuring the TTE-Switch Lab Space

The device and network configuration files for the TTE-Switch Lab Space are created (see Section 6.1 on the next page) and stored in the flash memory. The firmware maintains a file system on the flash device to this end.

The device configuration files can be updated through TFTP only. A Python script lets the user prepare the configuration files for the TFTP transfer. The Python script will add the header, CRC checksum, and hash values that the firmware needs to check that the configuration files were transferred correctly before they are stored.

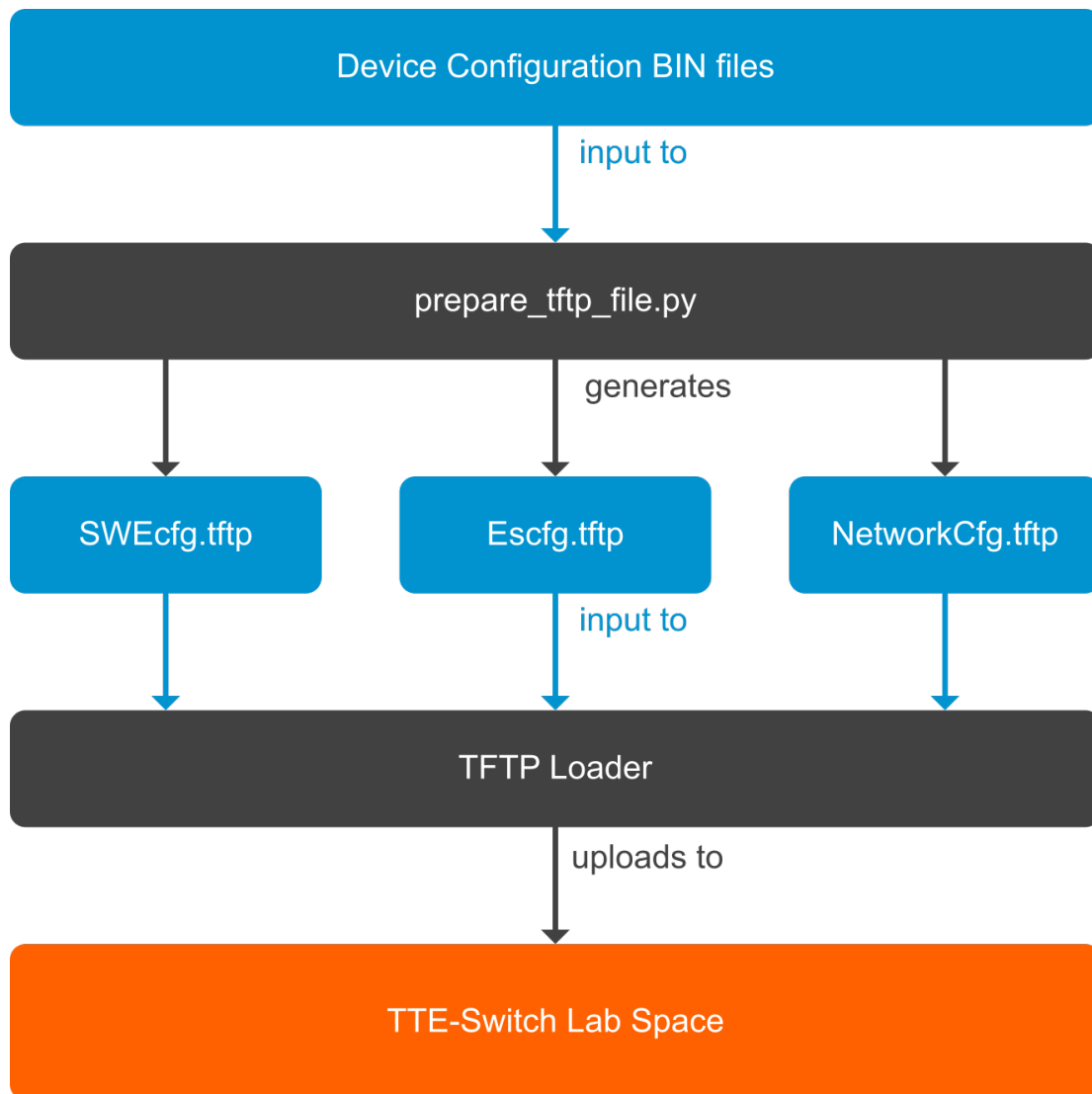


Figure 8: Schematic overview of dataloading

NOTE Any dataloading software that supports TFTP can be used on the host system to transfer configuration files to the TTE-Switch Lab Space.

NOTE Both the SWEcfg.tftp and the Escfg.tftp configurations must be uploaded to obtain a working setup.

6.1 Creating a Switch Configuration

In order to upload files to the TTE-Switch Lab Space, it is recommended to use the default configuration. It is also possible to upload files with the user configuration on the condition that the internal end system is reachable with best-effort traffic. The target files (also see Table 6 on page 32) are as follows:

- Internal End System Configuration File
- Switch Engine Configuration File
- Network Configuration File

The internal end system configuration is required in order to communicate with the firmware. In addition, it is also necessary to configure the switch engine to route packets meant for the internal end system. If the switch engine does not forward any frames to the internal end system and/or the internal end system is not configured correctly, the switch cannot be reconfigured until a hard reset to its default configuration is done.

The device configuration includes the internal end system and switch engine configuration files that are created with the TTE-Tools. The network configuration file is created with a hex editor by modifying the network parameters.

These files should be replaced and will run in user configuration mode.

6.1.1 Using the TTE-Tools for Device Configuration

The configuration for the switch is created by the TTE-Tools (minimum recommended version: 5.3.1037) – TTE-Plan, TTE-Build Network Configuration, and TTE-Build Device Configuration (see Figure 9 on the next page).

- **TTE-Plan** is the TTEthernet network design tool. Based on input provided in a network description XML file, TTE-Plan creates the network configuration in a user-convenient way and calculates the TTEthernet schedule for the network. The network description XML file describes the high-level communication requirements for the system, e.g., physical and logical topology, virtual links (VLs), including their IDs, timing requirements and possible frame sizes, as well as synchronization parameters and requirements, e.g., the SAE AS 6802 clock.
- **TTE-Build Network Configuration** knows the specifics of all supported TTEthernet devices. The tool extracts the data from the network configuration, calculates the parameters for the individual devices, and generates the device configuration files.
- **TTE-Build Device Configuration** converts the device configurations from the XML representation into binary configuration images required by the TTEthernet switches and TTEthernet end systems. The XML schemas used to describe these specifications will be publicly available and allow for the highest level of flexibility when TTE-Build Device Configuration is integrated with third-party tools or customer-specific tool chains.

A detailed description of how to create a configuration with the TTE-Tools can be found in the TTE- Plan User Manual [9] and the TTE-Build User Manual [8]. The TTE-Tools model the entire network, not just a single device. Therefore, it is necessary to specify an end system for each port of the switch that a virtual link has as source or destination. The virtual links must be configured from end system to end system.

NOTE

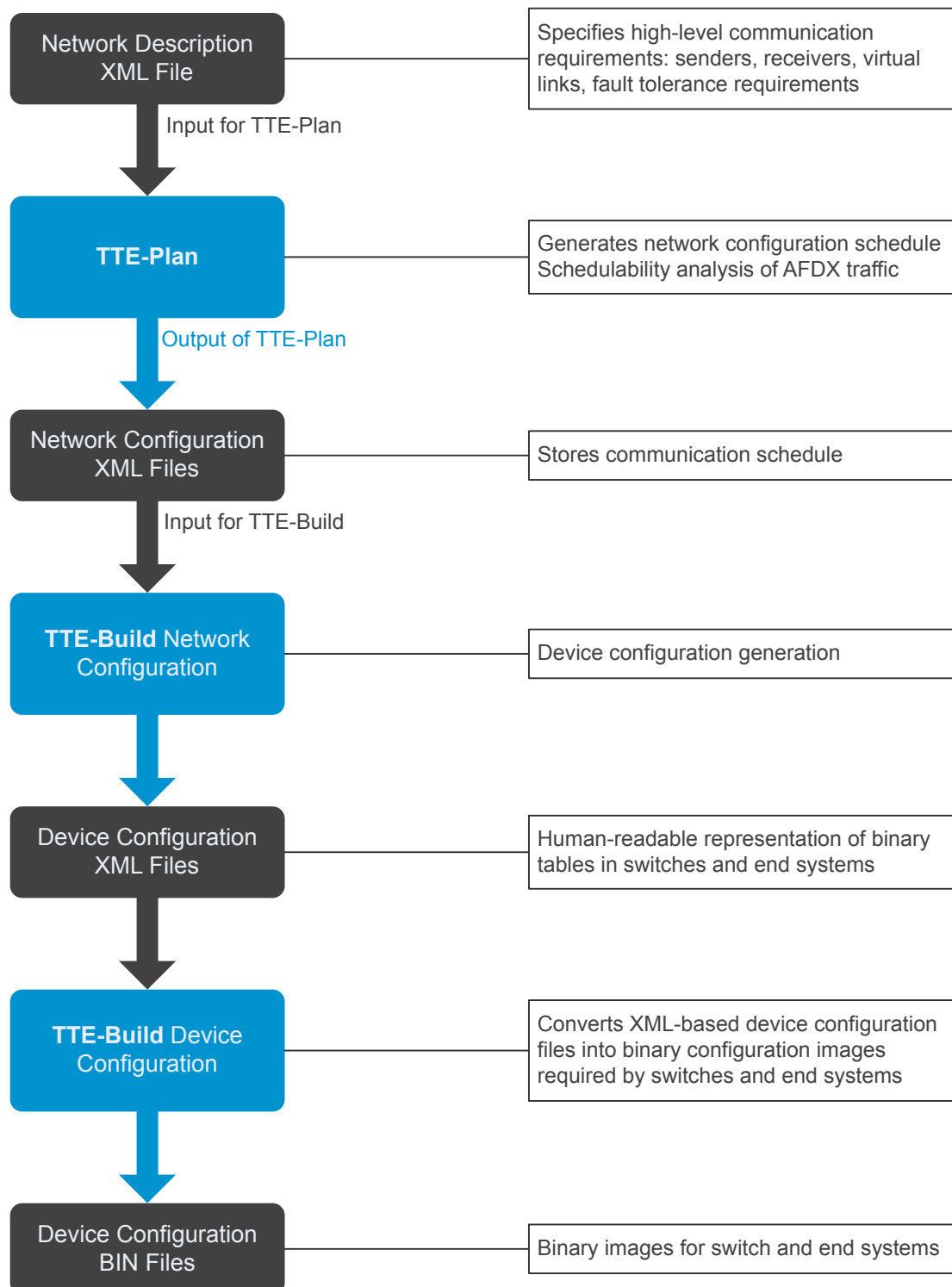


Figure 9: TTE-Tools data flow overview

6.1.1.1 Configuring the Switch

In the network description XML file, a switch device must be created.

For **TTE-Tools version 5.3 or below**, the device target is TTE-Switch_Controller_Space_ASIC:

```

1  <device xsi:type="topo:Switch" name="swlsp_sw0_0">
2  <port name="swlsp_sw0_P0" targetId="PHY.25"/>
3      <port name="swlsp_sw0_P1" targetId="PHY.22"/>
4      <port name="swlsp_sw0_P2" targetId="PHY.21"/>
5      <port name="swlsp_sw0_P3" targetId="PHY.7"/>
6      <port name="swlsp_sw0_P4" targetId="PHY.8"/>
7      <port name="swlsp_sw0_P5" targetId="PHY.9"/>
8      <port name="swlsp_sw0_P6" targetId="PHY.10"/>
9      <port name="swlsp_sw0_P7" targetId="PHY.11"/>
10     <port name="swlsp_sw0_P8" targetId="PHY.12"/>
11     <port name="swlsp_sw0_P9" targetId="PHY.13"/>
12     <port name="swlsp_sw0_P10" targetId="PHY.23"/>
13     <port name="swlsp_sw0_P11" targetId="PHY.24"/>
14     <port name="swlsp_sw0_P12" targetId="PHY.14"/>
15     <port name="swlsp_sw0_P13" targetId="PHY.15"/>
16     <port name="swlsp_sw0_P14" targetId="PHY.16"/>
17     <port name="swlsp_sw0_P15" targetId="PHY.17"/>
18     <port name="swlsp_sw0_P16" targetId="PHY.18"/>
19     <port name="swlsp_sw0_P17" targetId="PHY.19"/>
20     <port name="swlsp_sw0_P18" targetId="PHY.20"/>
21     <port name="swlsp_sw0_P19" targetId="PHY.6"/>
22     <port name="swlsp_sw0_P20" targetId="PHY.5"/>
23     <port name="swlsp_sw0_P21" targetId="PHY.4"/>
24     <port name="swlsp_sw0_P22" targetId="PHY.3"/>
25     <port name="swlsp_sw0_P23" targetId="PHY.2"/>
26     <port name="swlsp_sw0_P24" targetId="PHY.1"/>
27     <port name="swlsp_sw0_0_P0" targetId="PHY.0"/>
28     <syncConfig xsi:type="sync:AS6802SyncConfig" name="swlsp_sw0_PSYNC"
        targetId="AS6802.0" syncRole="syncCompressionMaster"
        syncPriority="//@syncDomain[name='SYNC_DOMAIN_1']
        /@syncPriority[name='PRIORITY_1']"/>
29     <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
        addrMask="FF:FF:FF:FF:FF:FF" srcPorts="//@device[name='swlsp_sw0_0']
        /@port[name='swlsp_sw0_0_P0']"
        dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P1']
        //@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P2']
        //@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P3']"/>
30     <deviceTarget
        href="platform:/plugin/com.ttttech.ttetools.models.targetdevice/data
        /TTE_Switch_Controller_Space_ASIC.targetdevice#"/>
31 </device>

```

Listing 1: Mapping the ports

For **TTE-Tools above version 5.3**, the device target is TTE_Switch_Lab_Space:

1 This feature will be available upon release of the corresponding TTE-Tools release.

Synchronization only needs to be configured if time-triggered traffic is to be transmitted in the network. According to the SAE AS6802 [6] standard, a TTEthernet network can have 1..n synchronization domains

and 1..n synchronization priorities.

- **port**
The attribute `port` is used to define all the necessary ports of the switch. The user must define 25 physical ports, from P0 to P24.
- **syncConfig**
Set the attribute `syncRole` to `syncCompressionMaster` when configuring the switch as the TTEthernet compression master and reference a `syncPriority`, or set the attribute `syncRole` to `syncNone` if no TTEthernet clock synchronization is required. The synchronization port AS6802.0 must be defined if TTEthernet clock synchronization is required.
- **deviceTarget**
This attribute specifies the name of the device target. For **TTE-Tools 5.3 and below**, use `TTE_Switch_Controller_Space_ASIC`. For **TTE-Tools x.x**, use `TTE_Switch_Lab_Space`.

6.1.1.2 Configuring the Internal End System

The management CPU of the switch uses an internal end system to send and receive frames for ICMP (ping) and diagnostics via SNMP. ICMP and SNMP via best-effort traffic are always possible by selecting the default configuration.

Best-Effort Management Configuration

If a certain configuration apart from the default configuration also requires management services via best-effort traffic, the corresponding routes must be defined between the switch and the internal end system. In this case, the following settings must be made in the network description to configure the internal end system for the switch:

- Create a switch-internal management end system.
- Specify the name of the device target.
- Define a physical port and a host port.
- Specify the MAC interface.

For **TTE-Tools version 5.3 or below**, the device target is `TTE-ES_Controller_Space_ASIC`:

```

1  <device xsi:type="topo:EndSystem" name="ESx">
2    <port name="swlsp_sw0_ES_P1" targetId="PHY.1"/>
3    <deviceTarget
      href="platform:/plugin/com.tttech.ttetools.models.targetdevice/
      data/TTE_ES_Controller_Space_ASIC.targetdevice#/" />
4    <hostInterface name="swlsp_sw0_ES_PHOST" targetId="HOST.0">
5      <macInterface name="swlsp_sw0_ES_MAC" address="22:22:22:22:22:22"/>
6    </hostInterface>
7  </device>

```

Listing 2: BE management configuration

For **TTE-Tools version 5.3.1027 and above**, the device target is `TTE_Switch_Lab_Space_Internal_ES`:

```

1  This feature will be available upon release of the corresponding
   TTE-Tools release.

```

- **port**
The attribute `port` is used to specify all the necessary ports of the internal end system. `PHY.1`, `PHY.2` and `PHY.3` specify the physical ports.

- **deviceTargets**
This attribute specifies the name of the device target. For **TTE-Tools 5.3 and below**, use TTE_ES_Controller_Space_ASIC. For **TTE-Tools x.x**, use TTE_Switch_Lab_Space_Internal_ES.
- **hostInterface**
The attribute HOST.0 specifies the host port and the attribute address of element <macInterface> specifies the MAC address of the internal end system.

6.1.1.3 Configuring Best-Effort Routes

If a management service such as ARP support (ICMP ping) and diagnostics via SNMP shall be available, this has to be correctly defined and configured. For this purpose, broadcast best-effort routes from each switch port to the management port have to be defined in the network description XML. These broadcast best-effort routes are necessary to have the switch accept the messages and handle them accordingly. The following section describes how these broadcasting routes are defined.

These best-effort route configurations consist of two parts:

In the first part, a BE broadcast route from the internal end system port (swlsp_sw0_0_P0) to all switch ports (P1, P1, ..., P1) needs to be created as follows:

```

1  <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
    addrMask="FF:FF:FF:FF:FF:FF" srcPorts="//@device[name='swlsp_sw0_0']/
    @port[name='swlsp_sw0_0_P0']"
    dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P1']"
    //@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P2']
    //@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P3']"/>
2
3  <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
    addrMask="FF:FF:FF:FF:FF:FF" srcPorts="//@device[name='swlsp_sw0_0']/
    @port[name='swlsp_sw0_0_P0']"
    dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P4']"
    //@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P5']
    //@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P6']"/>

```

Listing 3: First part of the BE route configuration

- **destinationMacAddress**
Set the attribute destinationMacAddress to FF:FF:FF:FF:FF:FF to define a broadcast route.
- **addrMask**
Set the attribute addrMask FF:FF:FF:FF:FF:FF to define a broadcast route.
- **srcPorts**
This concerns the source physical port of the route, swlsp_sw0_0_P0.
- **dstPorts**
This concerns the destination physical port of the route, all physical ports.

In the second part of the best-effort route configuration, it is necessary to define routes from every physical port of the switch that is a source to the swlsp_sw0_0_P0 destination as follows:

```

1  <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
    addrMask="FF:FF:FF:FF:FF:FF"
    srcPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P1']"
    dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_0_P0']"/>
2
3  <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
    addrMask="FF:FF:FF:FF:FF:FF"
    srcPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P2']"
    dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_0_P0']"/>

```

```

4
5 <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
  addrMask="FF:FF:FF:FF:FF:FF"
  srcPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P3']"
  dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_0_P0']"/>
6
7 <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
  addrMask="FF:FF:FF:FF:FF:FF"
  srcPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P4']"
  dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_0_P0']"/>
8
9 <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
  addrMask="FF:FF:FF:FF:FF:FF"
  srcPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P5']"
  dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_0_P0']"/>
10
11 <bestEffortRoute destinationMacAddress="FF:FF:FF:FF:FF:FF"
  addrMask="FF:FF:FF:FF:FF:FF"
  srcPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_P6']"
  dstPorts="//@device[name='swlsp_sw0_0']/@port[name='swlsp_sw0_0_P0']"/>

```

Listing 4: Second part of the BE route configuration

6.1.2 Using a Python Script to Configure the Network

The `create_network_config.py` Python script makes it possible to create a network configuration without the TTE-Tools.

Example

```
python2 create_network_config.py --mac A6:A7:A8:0:1:20 --ip 10.10.10.81
--gateway 10.10.10.1 --subnet 255.255.0.0 NW_cfg.bin
```

NW_cfg.bin

The name of the configuration binary that will be created.

6.2 Uploading the Switch Configuration

The memory of the flash storage device is organized as a set of files as listed in Table 6 on this page. These files are typically accessed via TFTP.

File Name	Maximum File Size	Protection	Purpose
TTC_EScfg.bin	1 MB	None	The user internal end system (ES) configuration.
TTC_SWEcfg.bin	1 MB	None	The user switch engine (SWE) configuration.
TTC_NetworkCfg.bin	128 kB	None	The network configuration for the internal end system.

Table 6: List of files in the flash memory

NOTE

The firmware does not check the data of the files, only the integrity of the files as a whole. Therefore, it is the user's responsibility to upload the correct data files to the file system in the flash memory.

It is possible to use any TFTP client to upload the files. If the files are not uploaded correctly, the Error LED will light up red when trying to enter the user configuration mode (see Table 5 on page 19). It is recommended to use the default configuration when uploading configuration files.

NOTE

6.2.1 Preparing Files for an Upload

To upload a file with TFTP, it must be converted to a specific binary format. The script `prepare_tftp_file.py` is called as follows:

```
python2 prepare_tftp_file.py [-e] [-h] [-l] [--verbose] [-v] [-f file]
-s source <filename.bin> <filename.tftp>
```

<code>-e</code>	Force encoding with 64/72-bit ECC protection, must be used only for converting a firmware binary. Not needed when source type FIP is used.
<code>-h</code>	Print this usage help and exit.
<code>-l</code>	List available file names in the file system and exit.
<code>--verbose</code>	Verbose output, print additional information during processing.
<code>-v</code>	Print the version of the script and exit.
<code>-f file</code>	Name of the file in the file system in the flash memory.
<code>-s source</code>	Type of the source file, either Image or FIP.
<code><filename.bin></code>	Binary input file, e.g., a switch engine or network configuration file.
<code><filename.tftp></code>	Output file, i.e., the converted input file, ready for an upload via TFTP.

Example The following command converts a stand-alone end system configuration for an upload via TFTP:

```
python2 prepare_tftp_file.py -s Image ES_Config.bin ES_user_cfg.tftp
```

6.2.2 Uploading TFTP Files

Once a TFTP file has been generated, it can be uploaded to the respective target binary files.

Example 1

```
tftp -i <ipaddress> PUT NW_cfg.bin TTC_NetworkCfg.bin
```

Example 2

```
tftp -i <ipaddress> PUT ES.tftp TTC_EScfg.bin
```

Example 3

```
tftp -i <ipaddress> PUT SWE.tftp TTC_SWEcfg.bin
```

`<ipaddress>` The actual IP address of the switch (default: 10.10.10.10).

6.3 Downloading the Switch Configuration

Files can be downloaded from the TTE-Switch Lab Space through any TFTP client. The user therefore has to know the preconfigured IP address.

In comparison with the upload process (where the file must be converted), a file downloaded from the TTE-Switch Lab Space is a plain binary file without any header, hash, ECC protection, or other modification. To upload such a file again, the `prepare_tftp_file.py` script needs to be run.

```
tftp -i <ip address> GET <target.bin> <filename.bin>
```

<ip address>	The actual IP address of the switch (default: 10.10.10.10).
<target.bin>	The target configuration file that will be downloaded, e.g., <code>TTC_NetworkCfg.bin</code> , <code>TTC_EScfg.bin</code> , or <code>TTC_SWEcfcfg.bin</code> .
<filename.bin>	The output binary file, i.e., the file converted by <code>prepare_tftp_file.py</code> .

Example

The following command downloads a binary from the flash memory:

```
tftp -i 10.10.10.10 GET TTC_EScfg.bin ES_user_cfg.bin
```

6.4 Recovering from Errors

The following error conditions may apply during operation. These descriptions are for general guidance only.

Error	Possible Cause	Mitigation
Firmware does not respond on the network	• Network cable not connected • Wrong network speed • Wrong MAC address • Firmware in ERROR mode • Wrong port configured for PHY	• Check UART for critical error • Check network cable • Set network speed to 100 Mbit • Check for UART error messages
SNMP error response on GET request	OID not supported.	No mitigation.
SNMP error response on SET request	OID not supported.	No mitigation.
TFTP error response	• File corrupted • Wrong file name provided	• See Section 6.2 on page 32 and Section 6.3 on the current page. • See Table 6 on page 32.
SNMP trap is not transmitted	User configuration is not correct.	• Check the current firmware mode via SNMP.
No response to SNMP GET requests	The firmware is in ERROR mode.	Check the UART for an error message.
SNMP timeout	• Firmware in wrong mode • Firmware in ERROR mode • Wrong network address (default configuration) • Wrong network address (user configuration)	• Check the last message on the UART. • Check the addresses used in the user configuration.

Error	Possible Cause	Mitigation
TFTP timeout	• Firmware in wrong mode • Firmware in ERROR mode • Wrong network address (default configuration)	• Check the last message on the UART. • Check the addresses that was sent by the SNMP SET request.

Table 7: *Recovery from errors*

7 Diagnosis

Management Information Database

The Management Information Database (MIB) describes the managed objects that can be retrieved via SNMP Version 1 (SNMPv1) on UDP port 161. Make sure to use the correct IP address according to your loaded configuration. The default configuration has the IP address 10.10.10.10 (see Section B on page 39).

In your preferred SNMP client, make sure to choose SNMPv1 and disable all other versions of the SNMP protocol. The MIB files listed below can be loaded into any SNMPv1-compliant SNMP client. The MIB files contain the following diagnostic information about the switch status:

- TT_MIB-TTE-SWE (see Section B.2 on page 39)
Contains all relevant status information and error counters of the switch engine.
- TT-MIB-TTE-ES (see Section B.3 on page 47)
Contains all relevant status information and error counters of the switch-internal end system. For more detailed information about the switch-internal end system (see the Pegasus End System Interface Control Document [7]).

8 Troubleshooting

8.1 Basic Checks

If you encounter a malfunction of the TTE-Switch Lab Space, there are some basic checks you can perform:

- Does the power fuse work? If the power fuse works, but the LEDs of the switch still do not light up, have the AC wall socket checked to see if the wall socket is working as intended.
- Is the pin programming of the GPIO interface correct for the desired configuration?
- What do the status LEDs display? See Section 3.2.3.3 on page 19 for details.
- Are all temperatures within the allowed ranges? See Section 3.3 on page 19 for details and check the SNMP client output for the temperature values of the different sensors.

8.2 SpaceWire Hot-Plugging Issues

If you run into hot-plugging issues with SpaceWire, the following procedure has been proven to work during tests:

1. Plug in the SpaceWire cable
2. Power on the SpaceWire component
3. Power up the TTE-Switch Lab Space

Alternatively, hit the reset button upon connecting the SpaceWire cable to the TTE-Switch Lab Space.

8.3 The arp Command Fails

The arp command needs to be run as administrator.

If you get an error message that states that The ARP entry addition failed: Access is denied., find out the interface name (i.e., the name of the network card) and add it to the list:

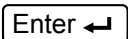
1. `netsh interface show interface`
2. `netsh interface ip add neighbors "<interface name>" <gateway IP> <gateway MAC>`

Example

```
netsh interface ip add neighbors "Ethernet 4" 10.10.10.10 A6-A7-A8-00-01-20
```

8.4 The tftp Command Does Not Work

To use the TFTP client through the tftp command, TFTP has to be enabled in the operating system. On Microsoft Windows this can be achieved as follows:

1. Open the  menu.
2. Type `Turn Windows features on or off` and press .
3. Select the checkbox next to **TFTP Client** and press **OK**.

8.5 No Link Is Established

If no link is established, autonegotiation may be the culprit. A crossover cable is needed to create a link in this case.

A crossover cable is also needed if the two devices that are to be linked both have autonegotiation set to **disabled**.

8.6 Customer Support Information

Company address:

TTTech Computertechnik AG
Schoenbrunner Strasse 7
A-1040 Vienna, Austria

For technical assistance and support regarding TTTech products, please contact our customer support:

- E-Mail: support@tttech.com
- If there are problems with your TTE-Switch Lab Space, please have the serial number ready to speed up the processing of your support request.

A Default Configuration

The default configuration is part of the firmware image of the switch. Therefore, the default configuration is not part of the user configuration and cannot be changed via TFTP.

For default/user configuration selection, see Section 3.2.2.1 on page 17.

The settings of the default configuration are:

```

1      IP: 10.10.10.10
2      MSK: 255.255.0.0
3      GW: 10.10.10.1
4      MAC: A6:A7:A8:00:01:20

```

Listing 5: Default configuration settings

B MIBs

B.1 TT-MIB

```

1  TT-MIB DEFINITIONS ::= BEGIN
2
3      IMPORTS enterprises
4          FROM RFC1155-SMI;
5
6      tttech      OBJECT IDENTIFIER ::= { enterprises 38355 }
7      technology  OBJECT IDENTIFIER ::= { tttech 100 }
8      platformSW  OBJECT IDENTIFIER ::= { technology 1 }
9      ttEthernet  OBJECT IDENTIFIER ::= { technology 2 }
10     ttp         OBJECT IDENTIFIER ::= { technology 3 }
11
12     products    OBJECT IDENTIFIER ::= { tttech 200 }
13     common      OBJECT IDENTIFIER ::= { products 1 }
14     smc         OBJECT IDENTIFIER ::= { common 1 }
15
16     aerospace   OBJECT IDENTIFIER ::= { products 3 }
17     spaceAsic   OBJECT IDENTIFIER ::= { aerospace 1 }
18     aeroSwitch  OBJECT IDENTIFIER ::= { aerospace 3 }
19
20 END

```

Listing 6: TT-MIB definitions

B.2 TT-SWE-VNV-MIB

```

1  TT-SWE-VNV-MIB DEFINITIONS ::= BEGIN
2
3
4      IMPORTS
5      ttEthernet FROM TT-MIB
6      Counter, OBJECT-TYPE FROM RFC1155-SMI;
7

```

```

8      tteSwitchingEngineVnV          OBJECT IDENTIFIER ::= { ttEthernet 4 }
9
10     tteSweDevId OBJECT-TYPE
11         SYNTAX      OCTET STRING (SIZE(0..50))
12         ACCESS      read-only
13         STATUS      mandatory
14         DESCRIPTION
15             "TTE-SWE Device ID."
16         ::= { tteSwitchingEngineVnV 1 }
17
18     tteSweDevRev OBJECT-TYPE
19         SYNTAX      OCTET STRING (SIZE(0..50))
20         ACCESS      read-only
21         STATUS      mandatory
22         DESCRIPTION
23             "TTE-SWE Device revision."
24         ::= { tteSwitchingEngineVnV 2 }
25
26     tteSweDevItfRev OBJECT-TYPE
27         SYNTAX      OCTET STRING (SIZE(0..50))
28         ACCESS      read-only
29         STATUS      mandatory
30         DESCRIPTION
31             "TTE-SWE Device Interface Revision."
32         ::= { tteSwitchingEngineVnV 3 }
33
34     tteSweUsrCfgTable OBJECT-TYPE
35         SYNTAX      OCTET STRING (SIZE(0..50))
36         ACCESS      read-only
37         STATUS      mandatory
38         DESCRIPTION
39             "User configuration table. This is part of the TTEthernet
40             configuration.
41             NOTE: Only the first 252 characters can be read here."
42         ::= { tteSwitchingEngineVnV 4 }
43
44     tteSweComMonMode OBJECT-TYPE
45         SYNTAX      INTEGER (0..2)
46         ACCESS      read-only
47         STATUS      mandatory
48         DESCRIPTION
49             "COM/MON mode detected by the TTE-SwE.
50             Possible values are:
51             0: TTE_SWE_CM_MODE_MON
52                 COM/MON setup is active, core works as MON.
53             1: TTE_SWE_CM_MODE_COM
54                 COM/MON setup is active, core works as COM.
55             2: TTE_SWE_CM_MODE_STANDALONE
56                 COM/MON setup is deactivated, core works standalone."
57         ::= { tteSwitchingEngineVnV 5 }
58
59     tteSweComMonError OBJECT-TYPE
60         SYNTAX      INTEGER (0..67108863)
61         ACCESS      read-only
62         STATUS      mandatory
63         DESCRIPTION

```

```

64         "COM/MON error detected by the TTE-SwE on the respective port.
65         Each port is represented by a bit, port 0 at bit 0, port 1 at
66         bit 1 and so on.
67         0: No error occurred.
68         1: A COM/MON mismatch was detected at the respective port."
69     ::= { tteSwitchingEngineVnV 6 }
70
71 tteSweOutOfMemCond OBJECT-TYPE
72     SYNTAX      INTEGER (0..67108863)
73     ACCESS      read-only
74     STATUS      mandatory
75     DESCRIPTION
76         "Out-of-memory conditions detected by the TTE-SwE.
77         0: No error occurred.
78         1: An out-of-memory condition occurred in one of the
79         TTE-SwE's memory partition."
80     ::= { tteSwitchingEngineVnV 7 }
81
82 tteSweAccessMode OBJECT-TYPE
83     SYNTAX      INTEGER (0..2)
84     ACCESS      read-write
85     STATUS      mandatory
86     DESCRIPTION
87         "Access mode for the TTE-SwE IP. Possible values are:
88         0: TTE_SWE_CM_ACC_MODE_COM_MON
89         Request is sent to COM and MON module.
90         1: TTE_SWE_CM_ACC_MODE_COM
91         Request is sent to COM only.
92         2: TTE_SWE_CM_ACC_MODE_MON
93         Request is sent to MON only. NOTE: If not MON is given in
94         the system an status query might fail or provide invalid
95         data."
96     ::= { tteSwitchingEngineVnV 8 }
97
98 tteSweSyncState OBJECT-TYPE
99     SYNTAX      INTEGER (0..8)
100    ACCESS      read-only
101    STATUS      mandatory
102    DESCRIPTION
103        "Current state of the synchronization engine.
104        Possible values are:
105        0: INIT
106        1: INTEGRATE
107        2: UNSYNC
108        3: WAIT 4 CYCLE START
109        4: CA ENABLED
110        5: WAIT FOR IN
111        6: TENTATIVE SYNC
112        7: SYNC
113        8: STABLE"
114    ::= { tteSwitchingEngineVnV 9 }
115
116 tteSweClockCorrSmallest OBJECT-TYPE
117     SYNTAX      INTEGER
118     ACCESS      read-only
119     STATUS      mandatory

```

```

120     DESCRIPTION
121         "Smallest correction value ever applied by the clock
122         synchronization algorithm."
123     ::= { tteSwitchingEngineVnV 10 }
124
125     tteSweClockCorrLargest OBJECT-TYPE
126         SYNTAX      INTEGER
127         ACCESS      read-only
128         STATUS      mandatory
129         DESCRIPTION
130             "Largest correction value ever applied by the clock
131             synchronization algorithm."
132         ::= { tteSwitchingEngineVnV 11 }
133
134     tteSweNumberOfSyncLoss OBJECT-TYPE
135         SYNTAX      INTEGER(0..65535)
136         ACCESS      read-only
137         STATUS      mandatory
138         DESCRIPTION
139             "Number of transitions of the clock synchronization state
140             machine from SYNC state or STABLE state to anything else than
141             SYNC state or STABLE."
142         ::= { tteSwitchingEngineVnV 12 }
143
144     tteSweMembershipVectorCurrent OBJECT-TYPE
145         SYNTAX      INTEGER(0..255)
146         ACCESS      read-only
147         STATUS      mandatory
148         DESCRIPTION
149             "Membership value of the in-schedule integration protocol
150             control frame received within the most recent acceptance
151             window that has the largest weight.
152             NOTE: The field contains valid data only if the IP is
153             synchronized."
154         ::= { tteSwitchingEngineVnV 13 }
155
156     tteSweMembershipVectorSmallest OBJECT-TYPE
157         SYNTAX      INTEGER(0..255)
158         ACCESS      read-only
159         STATUS      mandatory
160         DESCRIPTION
161             "Membership vector with the smallest number of flags set of
162 all
163             in-schedule integration frames ever processed by the
164             synchronization state machine. The field will maintain its
165             value also when synchronization is lost."
166         ::= { tteSwitchingEngineVnV 14 }
167
168     tteSweMembershipVectorLargest OBJECT-TYPE
169         SYNTAX      INTEGER(0..255)
170         ACCESS      read-only
171         STATUS      mandatory
172         DESCRIPTION
173             "Membership vector with the largest number of flags set of all
174             in-schedule integration frames ever processed by the
175             synchronization state machine. The field will maintain its

```

```

175         value also when synchronization is lost."
176 ::= { tteSwitchingEngineVnV 15 }
177
178 -----
179 -- TTE-SWE Ethernet Port Status Table --
180 -----
181 tteSweEthPortStatusTable OBJECT-TYPE
182     SYNTAX  SEQUENCE OF TteSweEthPortStatusTableEntries
183     ACCESS  not-accessible
184     STATUS  mandatory
185     DESCRIPTION
186         "TTE-SWE Ethernet Port status table."
187     ::= { tteSwitchingEngineVnV 16 }
188
189 tteSweEthPortStatusTableEntry OBJECT-TYPE
190     SYNTAX  TteSweEthPortStatusTableEntries
191     ACCESS  not-accessible
192     STATUS  mandatory
193     DESCRIPTION
194         "TTE-SWE Ethernet Port status table entry."
195     INDEX   { tteSweEthPortName }
196     ::= { tteSweEthPortStatusTable 1 }
197
198 TteSweEthPortStatusTableEntries ::=
199     SEQUENCE {
200         tteSweEthPortName
201             OCTET STRING (SIZE(0..50)),
202         tteSweEthPortTxBytes
203             OCTET STRING (SIZE(0..50)),
204         tteSweEthPortTxFrames
205             Counter,
206         tteSweEthPortRxBytes
207             OCTET STRING (SIZE(0..50)),
208         tteSweEthPortRxFrames
209             Counter,
210         tteSweEthPortNoLossBePolicing
211             Counter,
212         tteSweEthPortNoLossCtPolicing
213             Counter,
214         tteSweEthPortNoLossUnknownVl
215             Counter,
216         tteSweEthPortNoLossCrcError
217             Counter,
218         tteSweEthPortNoLossLengthError
219             Counter,
220         tteSweEthPortNoLossSofError
221             INTEGER(0..255),
222         tteSweEthPortNoLossAlignmentError
223             INTEGER(0..255),
224         tteSweEthPortNoLossMiiError
225             INTEGER(0..255),
226         tteSweEthPortNoLossRunt
227             INTEGER(0..255)
228     }
229
230 tteSweEthPortName OBJECT-TYPE

```

```

231     SYNTAX  OCTET STRING (SIZE(0..50))
232     ACCESS  read-only
233     STATUS  mandatory
234     DESCRIPTION
235         "Name of the Ethernet port."
236     ::= { tteSweEthPortStatusTableEntry 1 }
237
238 tteSweEthPortTxBytes OBJECT-TYPE
239     SYNTAX  OCTET STRING (SIZE(0..50))
240     ACCESS  read-only
241     STATUS  mandatory
242     DESCRIPTION
243         "Number of bytes sent on the respective Ethernet port."
244     ::= { tteSweEthPortStatusTableEntry 2 }
245
246 tteSweEthPortTxFrames OBJECT-TYPE
247     SYNTAX  Counter
248     ACCESS  read-only
249     STATUS  mandatory
250     DESCRIPTION
251         "Number of frames sent on the respective Ethernet port."
252     ::= { tteSweEthPortStatusTableEntry 3 }
253
254 tteSweEthPortRxBytes OBJECT-TYPE
255     SYNTAX  OCTET STRING (SIZE(0..50))
256     ACCESS  read-only
257     STATUS  mandatory
258     DESCRIPTION
259         "Number of bytes received on the respective Ethernet port."
260     ::= { tteSweEthPortStatusTableEntry 4 }
261
262 tteSweEthPortRxFrames OBJECT-TYPE
263     SYNTAX  Counter
264     ACCESS  read-only
265     STATUS  mandatory
266     DESCRIPTION
267         "Number of frames received on the respective Ethernet port."
268     ::= { tteSweEthPortStatusTableEntry 5 }
269
270 tteSweEthPortNoLossBePolicing OBJECT-TYPE
271     SYNTAX  Counter
272     ACCESS  read-only
273     STATUS  mandatory
274     DESCRIPTION
275         "Number of frames which were dropped based on the best
276 effort
277         policing operation."
278     ::= { tteSweEthPortStatusTableEntry 6 }
279
280 tteSweEthPortNoLossCtPolicing OBJECT-TYPE
281     SYNTAX  Counter
282     ACCESS  read-only
283     STATUS  mandatory
284     DESCRIPTION
285         "Number of frames which were dropped based on the
286         critical-traffic policing operation."

```

```

286         ::= { tteSweEthPortStatusTableEntry 7 }
287
288     tteSweEthPortNoLossUnknownVl OBJECT-TYPE
289         SYNTAX Counter
290         ACCESS read-only
291         STATUS mandatory
292         DESCRIPTION
293             "Number of frames which were dropped because the Virtual
Link
294             ID has not been configured for this port."
295         ::= { tteSweEthPortStatusTableEntry 8 }
296
297     tteSweEthPortNoLossCrcError OBJECT-TYPE
298         SYNTAX Counter
299         ACCESS read-only
300         STATUS mandatory
301         DESCRIPTION
302             "Number of frames received on the port carrying a wrong CRC
303             checksum."
304         ::= { tteSweEthPortStatusTableEntry 9 }
305
306     tteSweEthPortNoLossLengthError OBJECT-TYPE
307         SYNTAX Counter
308         ACCESS read-only
309         STATUS mandatory
310         DESCRIPTION
311             "Number of frames that are longer than 2047 bytes."
312         ::= { tteSweEthPortStatusTableEntry 10 }
313
314     tteSweEthPortNoLossSofError OBJECT-TYPE
315         SYNTAX INTEGER(0..255)
316         ACCESS read-only
317         STATUS mandatory
318         DESCRIPTION
319             "Number of frames that start with a byte other than 0x55 or
320             0xD5, have a byte other than 0xD5 being the first byte that
321             is different than 0x55 or 0xD5, or that terminated before
322             the SOF delimiter byte."
323         ::= { tteSweEthPortStatusTableEntry 11 }
324
325     tteSweEthPortNoLossAlignmentError OBJECT-TYPE
326         SYNTAX INTEGER(0..255)
327         ACCESS read-only
328         STATUS mandatory
329         DESCRIPTION
330             "Number of frames having a length that is not a multiple of
331             eight bits (at the given line speed)."
332         ::= { tteSweEthPortStatusTableEntry 12 }
333
334     tteSweEthPortNoLossMiiError OBJECT-TYPE
335         SYNTAX INTEGER(0..255)
336         ACCESS read-only
337         STATUS mandatory
338         DESCRIPTION
339             "Number of frames that terminated with the MII error input
340             being asserted."

```

```

341         ::= { tteSweEthPortStatusTableEntry 13 }
342
343 tteSweEthPortNoLossRunT OBJECT-TYPE
344     SYNTAX  INTEGER(0..255)
345     ACCESS  read-only
346     STATUS  mandatory
347     DESCRIPTION
348         "Number of frames that do not have a SOF error, alignment
349         error, nor MII error, but are shorter than 64 bytes."
350     ::= { tteSweEthPortStatusTableEntry 14 }
351
352 -----
353 -- TTE-SWE Memory Partition BE Status Table --
354 -----
355 tteSweMemPartBeStatusTable OBJECT-TYPE
356     SYNTAX  SEQUENCE OF TteSweMemPartBeStatusTableEntry
357     ACCESS  not-accessible
358     STATUS  mandatory
359     DESCRIPTION
360         "TTE-SWE memory partition status table for best effort
361         traffic."
362     ::= { tteSwitchingEngineVnV 17 }
363
364 tteSweMemPartBeStatusTableEntry OBJECT-TYPE
365     SYNTAX  TteSweMemPartBeStatusTableEntry
366     ACCESS  not-accessible
367     STATUS  mandatory
368     DESCRIPTION
369         "TTE-SWE memory partition status table entry for best effort
370         traffic."
371     INDEX   { tteSweMemPartBeDropCounter }
372     ::= { tteSweMemPartBeStatusTable 1 }
373
374 TteSweMemPartBeStatusTableEntry ::=
375     SEQUENCE {
376         tteSweMemPartBeDropCounter
377         Counter
378     }
379
380 tteSweMemPartBeDropCounter OBJECT-TYPE
381     SYNTAX  Counter
382     ACCESS  read-only
383     STATUS  mandatory
384     DESCRIPTION
385         "Number of frames dropped due to lack of best effort memory
386         partition space."
387     ::= { tteSweMemPartBeStatusTableEntry 1 }
388
389 -----
390 -- TTE-SWE Memory Partition CT Status Table --
391 -----
392 tteSweMemPartCtStatusTable OBJECT-TYPE
393     SYNTAX  SEQUENCE OF TteSweMemPartCtStatusEntry
394     ACCESS  not-accessible
395     STATUS  mandatory
396     DESCRIPTION

```

```

397         "TTE-SWE memory partition status table for critical
traffic."
398     ::= { tteSwitchingEngineVnV 18 }
399
400     tteSweMemPartCtStatusEntry OBJECT-TYPE
401         SYNTAX      TteSweMemPartCtStatusEntry
402         ACCESS      not-accessible
403         STATUS      mandatory
404         DESCRIPTION
405             "TTE-SWE memory partition status table entry for critical
traffic."
406         INDEX      { tteSweMemPartCtDropCounter }
407     ::= { tteSweMemPartCtStatusTable 1 }
408
409     TteSweMemPartCtStatusEntry ::=
410         SEQUENCE {
411             tteSweMemPartCtDropCounter
412             Counter
413         }
414
415     tteSweMemPartCtDropCounter OBJECT-TYPE
416         SYNTAX      Counter
417         ACCESS      read-only
418         STATUS      mandatory
419         DESCRIPTION
420             "Number of frames dropped due to lack of virtual link memory
partition space."
421     ::= { tteSweMemPartCtStatusEntry 1 }
422
423 END
424

```

Listing 7: TT-SWE-VNV-MIB definitions

B.3 TTE-ES-1-7-MIB

```

1  TTE-ES-1-7-MIB DEFINITIONS ::= BEGIN
2
3
4  IMPORTS ttEthernet FROM TT-MIB
5      Counter, OBJECT-TYPE FROM RFC1155-SMI;
6
7
8      tteEndSystem1-7  OBJECT IDENTIFIER ::= { ttEthernet 3 }
9
10     tteItfRev OBJECT-TYPE
11         SYNTAX      Counter
12         ACCESS      read-only
13         STATUS      mandatory
14         DESCRIPTION
15             "TTEthernet End System Interface Revision."
16     ::= { tteEndSystem1-7 1 }
17
18     tteDevId OBJECT-TYPE
19         SYNTAX      Counter
20         ACCESS      read-only

```

```

21         STATUS          mandatory
22         DESCRIPTION
23             "TTEthernet End System Device ID."
24         ::= { tteEndSystem1-7 2 }
25
26     tteDevRev OBJECT-TYPE
27         SYNTAX          Counter
28         ACCESS          read-only
29         STATUS          mandatory
30         DESCRIPTION
31             "TTEthernet End System Device Revision Number."
32         ::= { tteEndSystem1-7 3 }
33
34     tteSyncState OBJECT-TYPE
35         SYNTAX          Counter
36         ACCESS          read-only
37         STATUS          mandatory
38         DESCRIPTION
39             "TTEthernet Clock Synchronization State."
40         ::= { tteEndSystem1-7 4 }
41
42     tteSyncLoss OBJECT-TYPE
43         SYNTAX          Counter (0..65535)
44         ACCESS          read-only
45         STATUS          mandatory
46         DESCRIPTION
47             "The number of transitions of the clock
48             synchronization state
49             machine from SYNC state or STABLE state to any
50             other state
51             than SYNC or STABLE."
52         ::= { tteEndSystem1-7 5 }
53
54     tteDropIpFrag OBJECT-TYPE
55         SYNTAX          Counter
56         ACCESS          read-only
57         STATUS          mandatory
58         DESCRIPTION
59             "Number of frames dropped because of IPv4 fragmentation
60             errors."
61         ::= { tteEndSystem1-7 6 }
62
63     tteDropNoMem OBJECT-TYPE
64         SYNTAX          Counter
65         ACCESS          read-only
66         STATUS          mandatory
67         DESCRIPTION
68             "Number of frames dropped that could not be stored in the
69             frame
70             memory because the complete frame memory was full."
71         ::= { tteEndSystem1-7 7 }
72
73     tteUserData OBJECT-TYPE
74         SYNTAX          OCTET STRING (SIZE(0..127))
75         ACCESS          read-only
76         STATUS          mandatory

```

```

73      DESCRIPTION
74          "User Configuration Data. This is part of the TTEthernet
75          configuration.
76          NOTE: Only the first 128 characters can be read here."
77      ::= { tteEndSystem1-7 8 }
78
79  tteMinClkCorr OBJECT-TYPE
80      SYNTAX      INTEGER
81      ACCESS      read-only
82      STATUS      mandatory
83      DESCRIPTION
84          "The smallest correction value ever applied by the clock
85          synchronization algorithm."
86      ::= { tteEndSystem1-7 9 }
87
88  tteMaxClkCorr OBJECT-TYPE
89      SYNTAX      INTEGER
90      ACCESS      read-only
91      STATUS      mandatory
92      DESCRIPTION
93          "The largest correction value ever applied by the clock
94          synchronization algorithm."
95      ::= { tteEndSystem1-7 10 }
96
97  -- Channel statistics table
98
99  chStatTable OBJECT-TYPE
100     SYNTAX      SEQUENCE OF ChStatEntry
101     ACCESS      not-accessible
102     STATUS      mandatory
103     DESCRIPTION
104         "A table containing channel statistics columns."
105     ::= { tteEndSystem1-7 11 }
106
107  chStatEntry OBJECT-TYPE
108     SYNTAX      ChStatEntry
109     ACCESS      not-accessible
110     STATUS      mandatory
111     DESCRIPTION
112         "Information about a particular table row."
113     INDEX      { tteRxBytes }
114     ::= { chStatTable 1 }
115
116  ChStatEntry ::=
117     SEQUENCE {
118         tteRxBytes
119         Counter,
120         tteTxBytes
121         Counter,
122         tteSizeErr
123         Counter,
124         tteCrcErr
125         Counter,
126         tteSofErr
127         Counter,
128         tteAlignErr

```

```

129         Counter,
130     tteMiiErr
131         Counter,
132     tteRxFrames
133         Counter,
134     tteTxFrames
135         Counter,
136     tteVlNotFound
137         Counter
138 }
139
140 tteRxBytes OBJECT-TYPE
141     SYNTAX      Counter
142     ACCESS      read-only
143     STATUS      mandatory
144     DESCRIPTION
145         "Number of received bytes on the channel."
146     ::= { chStatEntry 1 }
147
148 tteTxBytes OBJECT-TYPE
149     SYNTAX      Counter
150     ACCESS      read-only
151     STATUS      mandatory
152     DESCRIPTION
153         "Number of sent bytes on the channel."
154     ::= { chStatEntry 2 }
155
156 tteSizeErr OBJECT-TYPE
157     SYNTAX      Counter
158     ACCESS      read-only
159     STATUS      mandatory
160     DESCRIPTION
161         "The number of frames that are either shorter than 64 bytes,
162         longer than 1518 bytes, or whose length does not match
163         the value of the EtherType/Length field of the Ethernet
164         frame."
165     ::= { chStatEntry 3 }
166
167 tteCrcErr OBJECT-TYPE
168     SYNTAX      Counter
169     ACCESS      read-only
170     STATUS      mandatory
171     DESCRIPTION
172         "The number of frames that terminated with a checksum error."
173     ::= { chStatEntry 4 }
174
175 tteSofErr OBJECT-TYPE
176     SYNTAX      Counter
177     ACCESS      read-only
178     STATUS      mandatory
179     DESCRIPTION
180         "The number of frames that start with a byte other than 0x55,
181         have a byte other than 0xD5 being the first byte that is
182         different from 0x55, or that terminated before the SOF
183         delimiter byte."
184     ::= { chStatEntry 5 }

```

```

184
185     tteAlignErr OBJECT-TYPE
186         SYNTAX      Counter
187         ACCESS      read-only
188         STATUS      mandatory
189         DESCRIPTION
190             "The number of frames having a length that is not a multiple
of
191             8 bits (at the given line speed)."
```

192 ::= { chStatEntry 6 }

```

193
194     tteMiiErr OBJECT-TYPE
195         SYNTAX      Counter
196         ACCESS      read-only
197         STATUS      mandatory
198         DESCRIPTION
199             "The number of frames that terminated with the MII error input
200             being asserted."
```

201 ::= { chStatEntry 7 }

```

202
203     tteRxFrames OBJECT-TYPE
204         SYNTAX      Counter
205         ACCESS      read-only
206         STATUS      mandatory
207         DESCRIPTION
208             "The number correctly received frames on the channel."
```

209 ::= { chStatEntry 8 }

```

210
211     tteTxFrames OBJECT-TYPE
212         SYNTAX      Counter
213         ACCESS      read-only
214         STATUS      mandatory
215         DESCRIPTION
216             "The number transmitted frames on the channel."
```

217 ::= { chStatEntry 9 }

```

218
219     tteVlNotFound OBJECT-TYPE
220         SYNTAX      Counter
221         ACCESS      read-only
222         STATUS      mandatory
223         DESCRIPTION
224             "The number of critical traffic frames dropped at the
respective
225             channel (since power-on or reset or the most recent read
access
226             of the host layer to this field) because their VL ID did not
227             produce a hit in the Input VL Lookup Table."
```

228 ::= { chStatEntry 10 }

```

229
230     -- Partition Statistics table
231
232     parStatTable OBJECT-TYPE
233         SYNTAX      SEQUENCE OF ParStatEntry
234         ACCESS      not-accessible
235         STATUS      mandatory
236         DESCRIPTION
```

```

237         "A table containing partition statistics."
238     ::= { tteEndSystem1-7 12 }
239
240     parStatEntry OBJECT-TYPE
241         SYNTAX      ParStatEntry
242         ACCESS      not-accessible
243         STATUS      mandatory
244         DESCRIPTION
245             "Information about a particular table row."
246         INDEX       { tteDropFull }
247     ::= { parStatTable 1 }
248
249     ParStatEntry ::=
250         SEQUENCE {
251             tteDropFull
252                 Counter,
253             tteMemThreshold
254                 Counter
255         }
256
257     tteDropFull OBJECT-TYPE
258         SYNTAX      Counter
259         ACCESS      read-only
260         STATUS      mandatory
261         DESCRIPTION
262             "This counter is incremented when a message that could be
263 stored
264             in an input port of partition i was dropped because
265             the corresponding input port was already full (only in case of
266             a queue). The counter is 24 bits wide and does not wrap."
267     ::= { parStatEntry 1 }
268
269     tteMemThreshold OBJECT-TYPE
270         SYNTAX      Counter
271         ACCESS      read-only
272         STATUS      mandatory
273         DESCRIPTION
274             "Indicator that the amount of free memory buffers exceeds
275             the configured threshold on the partition. 0 means threshold
276             was not reached. 1 means the threshold was reached."
277     ::= { parStatEntry 2 }
278
279     tteRmDropSpecial OBJECT-TYPE
280         SYNTAX      INTEGER
281         ACCESS      read-only
282         STATUS      mandatory
283         DESCRIPTION
284             "This flag will be set whenever the MAC drops a frame because
of
285             ARINC 664 redundancy management if the sequence number of
286             the frame received did not match the sequence number of
287             the frame of the same VL that most recently passed RM."
288     ::= { tteEndSystem1-7 13 }
289
290     tteRmIdxSpecial OBJECT-TYPE
291         SYNTAX      INTEGER

```

```

291     ACCESS      read-only
292     STATUS      mandatory
293     DESCRIPTION
294         "Index of the entry within the IC/RMTable of the configuration
295         file that was processed when most recently triggering setting
296         the tteRmDropSpecial flag.
297         This value is only valid after reading tteRmDropSpecial first
298         and if tteRmDropSpecial indicates an error. Otherwise its
299         content is undefined."
300     ::= { tteEndSystem1-7 14 }
301
302     tteRmDrop OBJECT-TYPE
303         SYNTAX      INTEGER
304         ACCESS      read-only
305         STATUS      mandatory
306         DESCRIPTION
307             "This flag will be set whenever the MAC drops a frame because
308             (a) it carried the same sequence number as the most recent
309             frame of the same VL that passed the check in case of ARINC
310             664
311             redundancy management or (b) it was received too close to
312             the most recent frame of the same VL that passed the check in
313             case of time-triggered redundancy management ."
314         ::= { tteEndSystem1-7 15 }
315
316     tteRmIdx OBJECT-TYPE
317         SYNTAX      INTEGER
318         ACCESS      read-only
319         STATUS      mandatory
320         DESCRIPTION
321             "Index of the entry within the IC/RM Table of the
322             configuration
323             file that was processed when most recently triggering setting
324             the tteRmDrop flag.
325             This value is only valid after reading tteRmDrop first
326             and if tteRmDrop indicates an error. Otherwise its
327             content is undefined."
328         ::= { tteEndSystem1-7 16 }
329
330     tteDropIpSize OBJECT-TYPE
331         SYNTAX      Counter
332         ACCESS      read-only
333         STATUS      mandatory
334         DESCRIPTION
335             "Number of frames dropped because of incorrect size of IP
336             packet."
337         ::= { tteEndSystem1-7 17 }
338
339     tteMinMembVec OBJECT-TYPE
340         SYNTAX      INTEGER
341         ACCESS      read-only
342         STATUS      mandatory
343         DESCRIPTION
344             "The membership vector with the smallest number of
345             flags set of all in-schedule integration frames
346             ever processed by the synchronization state

```

```
345         machine."
346     ::= { tteEndSystem1-7 18 }
347
348 tteMaxMembVec OBJECT-TYPE
349     SYNTAX      INTEGER
350     ACCESS      read-only
351     STATUS      mandatory
352     DESCRIPTION
353         "The membership vector with the largest number of
354         flags set of all in-schedule integration frames
355         ever processed by the synchronization state
356         machine."
357     ::= { tteEndSystem1-7 19 }
358
359 tteAggrMembVec OBJECT-TYPE
360     SYNTAX      INTEGER
361     ACCESS      read-only
362     STATUS      mandatory
363     DESCRIPTION
364         "The aggregated membership value of in-schedule
365         integration protocol control frames received
366         within the most recent acceptance window."
367     ::= { tteEndSystem1-7 20 }
368 END
```

Listing 8: *TTE-ES-1-7-MIB definitions*

Terms and Abbreviations

Entry	Description
ASIC	Application-Specific Integrated Circuit
BAG	Bandwidth Allocation Gap
BE	Best Effort
BIST	Built-In Self-Test
Best-Effort (BE) Traffic	Ethernet traffic that is not critical traffic (IEEE 802.3 standard traffic [4]). BE traffic will be serviced with lowest priority.
CBIT	Continuous Built-In Test
COM	Communication port, a serial port that acts as a serial communication interface.
CRC	Cyclic Redundancy Check
DSU	Debug Support Unit
EAN	European Article Number
Fail-silent	A fail-silent system either provides a service as intended or no service at all, i.e., it fails silently.
GPIO	General-Purpose Input/Output
ICMP	Internet Control Message Protocol
JTAG	Joint Test Action Group
LVDS	Low-Voltage Differential Signaling
MIB	Management Information Database
PBIT	Power-up Built-In Test
PCB	Printed Circuit Board
PHY	Physical Layer
QR Code	Quick Response Code
RC	Rate Constrained
RMAP	Remote Memory Access Protocol
Rate-Constrained (RC) Traffic	TT Ethernet traffic that is used for applications with less stringent determinism and real-time requirements than strictly time-triggered applications (ARINC 664 avionics traffic). RC traffic is used for safety-critical aerospace applications that depend on highly reliable communication and have moderate temporal quality requirements, e.g., multimedia systems.
S/N	Serial Number

Entry	Description
SFP	Small Form-factor Pluggable, a compact, optical module transceiver.
SNMP	Simple Network Management Protocol
SpW	SpaceWire
TFTP	Trivial File Transfer Protocol
TTE	Time-Triggered Ethernet (<i>this abbreviations is used as a prefix only</i>).
TT	Time-Triggered
Time-Triggered (TT) Traffic	TT Ethernet traffic that is used for applications with stringent determinism and real-time requirements (IEEE 1588-compatible clock synchronization service, real-time control [5]). TT traffic guarantees that bandwidth and latency are predefined for each application. TT traffic is used for safety-critical aerospace applications that depend on highly reliable communication and have high temporal quality requirements, e.g., closed loop control systems.
UART	Universal Asynchronous Receiver-Transmitter
UDP	User Datagram Protocol
VLAN	Virtual Local Area Network
VL	Virtual Link
TTEthernet	Time-Triggered Ethernet, also <i>TTE</i> when used as a prefix.

References

- [1] ARINC. ARINC Specification 664P7-1, *Aircraft Data Network, Part 7, Avionics Full-Duplex Switched Ethernet Network*. International Standard, Aeronautical Radio, Incorporated, 2009.
- [2] IEC. IEC 60297-3-105 Part 3-105: *Dimensions and design aspects for 1U chassis*. International Standard, International Electrotechnical Commission, 2004.
- [3] IEEE. 802.1Q™-2011 – *Standard for Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks*. International Standard, Institute of Electrical and Electronics Engineers, 2005.
- [4] IEEE. IEEE 802.3™-2005 – *Standard for Ethernet*. International Standard, Institute of Electrical and Electronics Engineers, 2005.
- [5] IEEE. IEEE 1588-2008 – *Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, Precision Time Protocol (PTP) Version 2. International Standard, Institute of Electrical and Electronics Engineers, 2008.
- [6] SAE. SAE AS 6802, *Time-Triggered Ethernet*. International Standard, SAE International, 2011.
- [7] TTTech. *Pegasus End System (Full IP) Interface Control Document*. Interface Description D-PEGTTE-ID-10-004, TTTech, 2013.
- [8] TTTech. *TTE-Build User Manual*. User Documentation D-TTE-G-01-002, TTTech, 2014.
- [9] TTTech. *TTE-Plan User Manual*. User Documentation D-TTE-G-01-010, TTTech, 2014.

Index

— Symbols —		— S —	
TTEthernet implementation	14	Standards compliance	15
		State machine	22
— A —		— T —	
ARINC 664p7 implementation	15	Troubleshooting	37
— B —			
Block diagram	16		
— C —			
Configuring the TTE-Switch Lab Space	26		
Customer support	38		
— D —			
Debug and maintenance interfaces	17		
Diagnosis	36		
— E —			
Environmental operating ranges	15		
External interfaces	13		
— F —			
Functional description	16		
Functional features	14		
— G —			
Getting started	21		
— H —			
Human-machine interfaces	19		
— I —			
Identification	13		
Interfaces	16		
Introduction	12		
— N —			
Networking interfaces	16		
— O —			
Overview	13		
— P —			
Physical specifications	15		
Power supply	15		