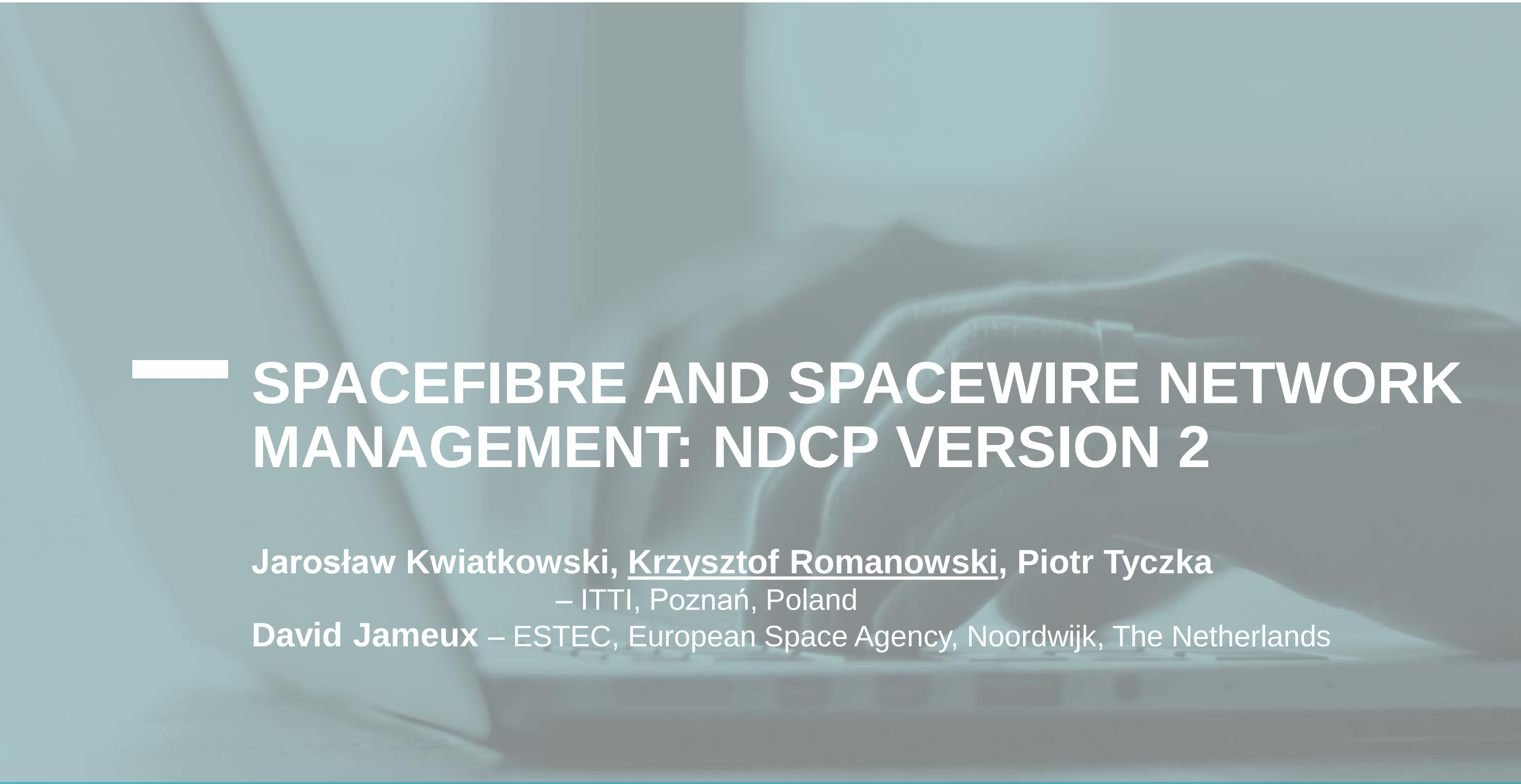


SPACEFIBRE AND SPACEWIRE NETWORK MANAGEMENT: NDCP VERSION 2

Jarosław Kwiatkowski, Krzysztof Romanowski, Piotr Tyczka
– ITTI, Poznań, Poland
David Jameux – ESTEC, European Space Agency, Noordwijk, The Netherlands

9th International SpaceWire and SpaceFibre Conference
Pisa, 17-19 October 2022

OUTLINE

- Network management, RMAP, and NDCP
- NDCP v. 2 principles
- NDCP v. 2 fields
- Demonstrator
- Conclusion

NETWORK MANAGEMENT

- Basic network management tasks:
  - Discover network devices
    - Presence
    - Identity
    - Configuration – get parameters
  - Discover networks (connection topologies)
  - Configure
    - Devices –set parameters
    - Networks – via configuring devices

- Network management data structure:
  - Management Information Base (MIB), specified in standards
    - SpaceWire: ECSS-E-ST-50-12C Rev.1 (2019)
    - SpaceFibre: ECSS-E-ST-50-11C (2019)
    - Similar, though with some differences – see table below.
  - Network Management Service with *fields*, specified in NDCP (formerly PnP) draft standard
    - Most recent version: ECSS-E-ST-50-54 Draft 1.8 (2016)

MANAGEMENT-RELATED DEFINITIONS

notion	SpaceWire (2019)	SpaceFibre (2019)
management parameter	(3.2.72) configuration parameter, control variable or status variable of a SpaceWire node or routing switch used to manage its operation	(3.2.78) configuration parameter, control parameter or status value of a SpaceFibre node or routing switch used to manage its operation
network manager	–	(3.2.85) **node** that configures, controls and monitors the status of the SpaceFibre network
MIB requirements – in section 5.2 “Protocol stack and interface architecture”	(5.2.6a) {MIB} shall provide two services: 1. A set parameter service which writes control or configuration information to the other SpaceWire layers. 2. A get status service which reads the status or current configuration or control values of the other SpaceWire layers.	(5.2.7a) {MIB} shall provide one service: 1. A Management service which writes control or configuration information to all SpaceFibre layers and which reads the status or current configuration or control values of all SpaceFibre layers.
	{MIB} shall be responsible for configuring, controlling and monitoring the operation of the other SpaceWire layers.	–
	{MIB} shall have direct access to the relevant configuration parameters, control parameters and status parameters in the network layer, data link layer, encoding layer and physical layer.	{MIB} shall have direct access to the relevant configuration parameters, control parameters and status parameters in the Network layer, Data Link layer, Multi-Lane layer, Lane layer and Physical layer.
MIB requirements – in a separate detailed section	(5.7) {MIB} specifies the way in which the operations of other SpaceWire layers are configured, controlled and monitored.	(5.9.1a) {MIB} shall configure, control and monitor the operation of the other SpaceFibre layers.
	(5.7.2) a. {MIB} shall provide a means for a **user application in a node** to configure and control **that node**. b. {MIB} shall provide a means for **a user application in a node** to read the current configuration, control and status information of **that node**. c. {MIB} shall provide a means for **a configuration port** in a routing switch to configure and control **that router**. d. {MIB} shall provide a means for a **configuration port** in a routing switch to read the current configuration, control and status information of **that router**.	(5.9.2) a. The **Remote Memory Access Protocol**, (RMAP) in compliance with the ECSS-E-ST-50-52, shall be used for remote configuration, control and monitoring of SpaceFibre **networks**. b. Network configuration, control and monitoring should be carried out using virtual network VN0 only. c. Network configuration, control and monitoring shall only be carried out by **network managers**. d. Only network managers shall be permitted to be RMAP initiators on virtual network VN0.

Cf. 6.5 “MIB service interface” – very similar

NETWORK ADDRESSING AND DISCOVERY

- No permanent unique device addresses ☹. Addresses may be:
  - Path addresses – depend on network topology
  - Logical addresses – depend on network topology and configuration of devices (routing tables)
- Thus – no way to address a *specific* device without knowing the topology; discovery or prior definition of the topology of the network is required.
- Path addressing is used in topology discovery (and in NDCP in general).
- Unique device IDs (temporary – for the NMT operation time) need to be assigned by the NMT (since there may be redundant paths or loops, this is indispensable for topology discovery).
- NDCP introduces the notion of ownership of a peripheral device (a network device being managed) by the control device (the NMT) and mechanisms that restrict writing configuration by non-owners, in order to limit the risk of overwriting the configuration of the peripheral device by another control device (e.g. a redundant NMT).
  - RMAP does NOT have that functionality.
- Explicit rather than implicit return addresses are crucial for topology discovery.

KEY DEVICE FEATURES FOR NMT OPERATION

Feature	Support in NDCP fields	Support in RMAP registers	
		Generally	In SpW-10X
Device ID	**Y**: *Device ID* field in *Device Identification* field set	**N** (unless a general purpose register is available)	**Y**: *Router Identity Register*
Device ownership	**Y**: *Owner Logical Address, Ownership Port, Owner Address*, and *Owner Address Length* fields in *Device Identification* field set	**N** (unless some general purpose registers are available)	**N** (may be partly simulated with *General Purpose Register*)
Protection against configuration overwriting by non-owners	**Y**	**N**	**N**
Availability of explicit return address	**Y**: *Return Port* field in *Device Identification* field set	**N** (unless supported by a particular implementation)	**Y**: *Return Port* field in *Network Discovery Register*

- With RMAP, at least a freely writeable general purpose register is indispensable.
- There are devices that fail to effectively write a requested value into such a register.

NDCP VS RMAP

- NDCP provides standardized addressing and contents of its fields as the Network Management Service.
- RMAP is standardized as to frame format and Read/Write/Read-Modify-Write services only.
- Different devices can have different organization of memory:
  - register contents: meaning, types, units, ranges, reset and default values,
  - register sizes and numbers,
  - memory address maps,
  - addressing modes (byte- vs. word-oriented),
  - byte order (most-significant-first vs. least-significant-first),
  - RMAP command types implemented (e.g. single address type vs. incrementing address type).
- No single RMAP-based management method for all RMAP-supporting devices.
- NDCP provides the return port information as standard.
- RMAP – does or does not provide, depending on specific implementation (device)

WHY IS THE RETURN PORT IMPORTANT?

- RMAP standard allows a conforming implementation on a target device to use an implicit return address or implicit partial return address for sending its reply packets – e.g. to send the reply via the same port on which the command arrived, without the need to specify that port in the reply address field of the RMAP header.

- SpW-10X routing switch, as an example, adopts this convention: the first element of the return address path, i.e. the return port, is implicit, however …

- … it is available for reading in the *Return Port* field of the SpW-10X register set.

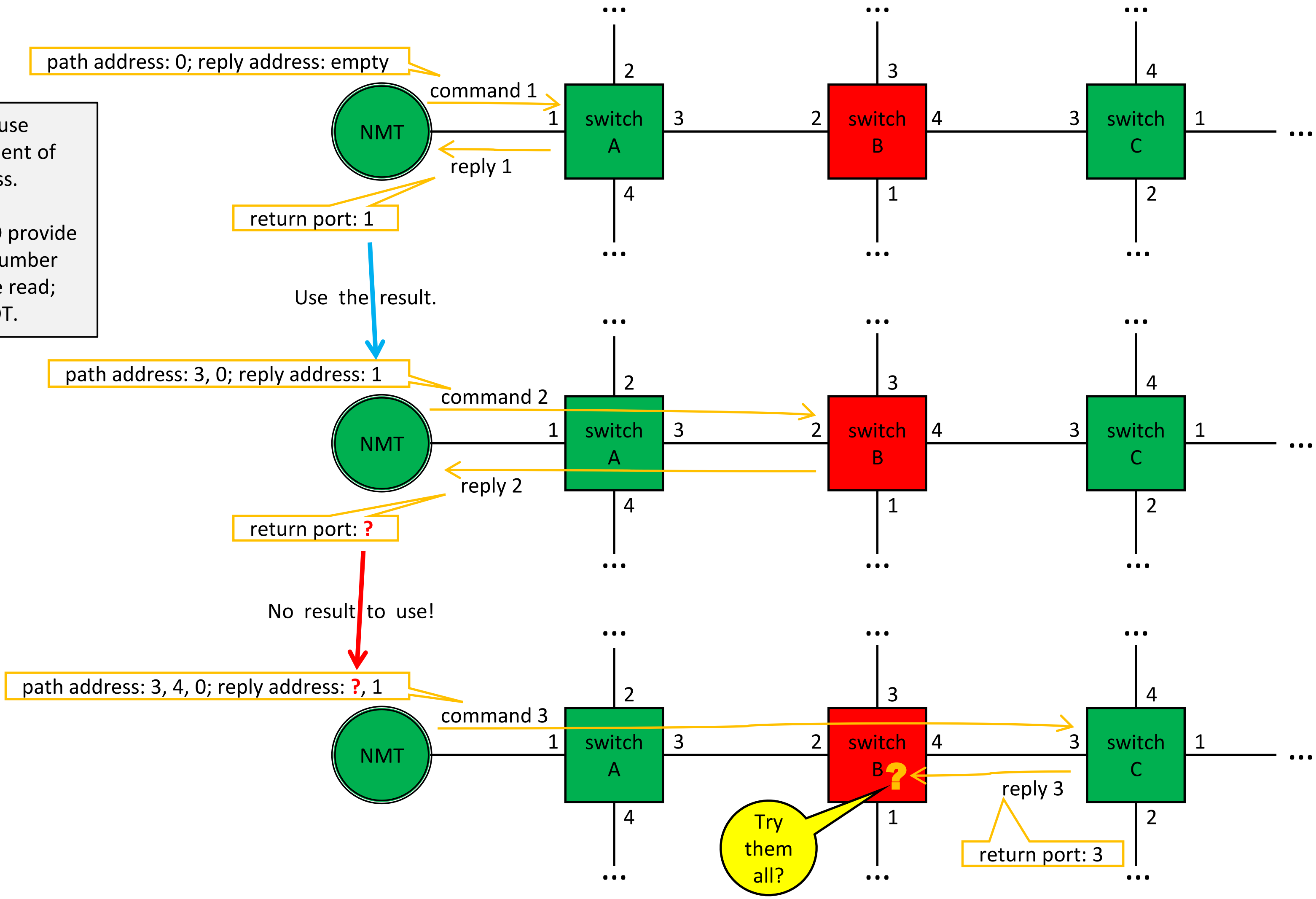
- By reading that field, the NMT can get the number of the port of the switch via which it sent the read command and can use that information for building the model of the network topology. ☺

- There are devices that use implicit return addressing but do not make the return port information available. ☹

IF NMT DOES NOT GET RETURN PORT INFO FROM SWITCH ‘B’…

Switches A, B, C use implicit first element of the return address.

Switches A, C DO provide the return port number in a register to be read; switch B does NOT.



path address: 0; reply address: empty

command 1

reply 1

return port: 1

Use the result.

path address: 3, 0; reply address: 1

command 2

reply 2

return port: ?

No result to use!

path address: 3, 4, 0; reply address: ?, 1

command 3

reply 3

return port: 3

Try them all?

TOWARDS STANDARDIZED MANAGEMENT: NDCP2

- A proposition for standardizing RMAP use for SpaceFibre network management was presented at the previous SpW/SpFi Conference:
  - Felix Siegle and Alessandro Leoni, “Standardization efforts for a network management and discovery protocol for SpaceFibre,” Proc. 8th Int. SpaceWire Conf. Long Beach 2018

- The *FiMan* project (ITTI) proposed an alternative: to extend NDCP to cover SpaceWire, SpaceFibre, and mixed SpaceWire/SpaceFibre networks.
  - This is called NDCP v.2
  - Based on the same principles as the original NDCP (PnP), referred here to as v.1:
    - Communications protocol with packet structure and field addressing
    - General structure of Network Management Service fields: field sets of 32-bit fields, each field identified by the quad:
      - application (service) index
      - protocol index
      - field set identifier
      - field identifier
    - Protocol ID unchanged from NDCP v.1 (i.e. 3, which is still at a proposal status)
    - Detailed organization and contents of the fields – modified, but support for v.1 retained, with a subfield used for identifying the specific version
  - Although neither SpW nor SpFi standards refer to nonhomogeneous devices (with both types of ports in the same device), such devices do exist, so the proposed field organization is common for both technologies.

NDCP V.2 FIELDS

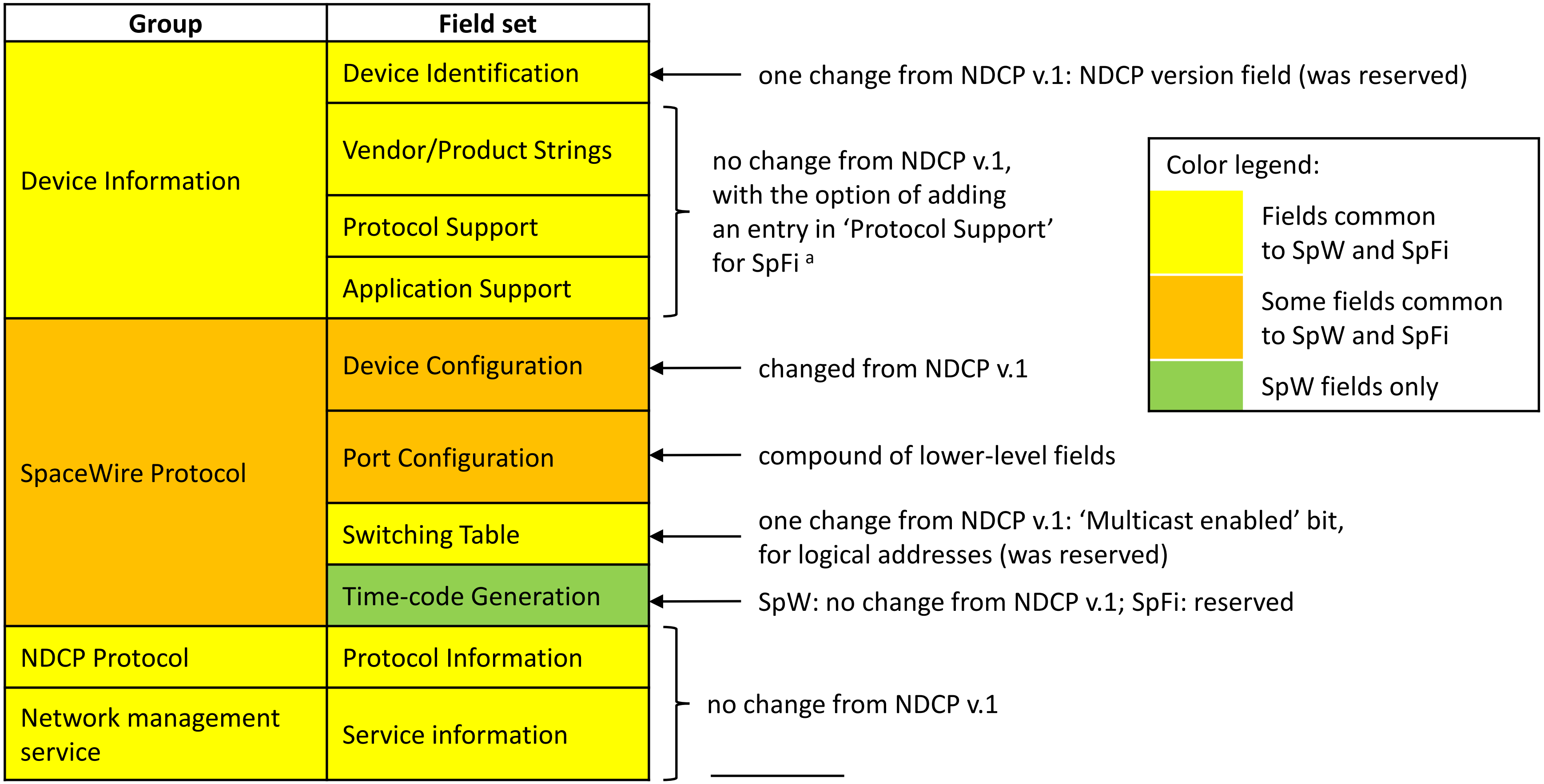
- Sources for the proposition of specific NDCP fields:
  - SpW standard (2019), sections on management parameters and MIB
  - SpFi standard, sections on management parameters and MIB, including the tables with SpFi configuration parameters and SpFi status parameters
  - NDCP v.1 (PnP) draft standard
  - Paper of Siegle & Leoni (2018)
  - Analysis done in the FiMan project
- Levels of the hierarchy of fields:

Field hierarchy level	Corresponding index or identifier in NDCP field addressing
Group	Pairs composed of an application index and a protocol index
Field set	Field set identifier
Field subset (optional; up to 3 levels, for port parameters)	No explicit corresponding addressing identifier; this level is only used for naming related adjacent fields
Field	Field identifier

COMMON SUPPORT OF NDCP V.1 AND V.2 BY NMT

- A problem for NMT application: how to detect which version, the old (v.1) or the new (v.2), is supported by a particular managed device, so as to interpret its data correctly?
- An unused – *reserved* – field of NDCP v.1 was selected.
- Note that according to NDCP v.1 draft standard, the reserved fields have to be readable by the application and to return zero when read.
- Thus zero read by the NMT at the selected field indicates the device is NDCP v.1-compliant.
- NDCP v.2-compliant device should instead return a non-zero value, which will be interpreted as indicating the NDCP version number.
- There are known exceptions: device models that violate the NDCP v.1 requirement of returning zero when reading a reserved field and instead return consistently a specific non-zero value. These exceptions are handled case-by-case: the NMT treats such values as zero.
- The NDCP field selected as the version indicator is byte 0 of the Version field of the Device Identification field set:

NDCP V.2 FIELD ORGANIZATION – HIGHER LEVELS



Group	Field set
Device Information	Device Identification
	Vendor/Product Strings
	Protocol Support
	Application Support
SpaceWire Protocol	Device Configuration
	Port Configuration
	Switching Table
	Time-code Generation
NDCP Protocol	Protocol Information
Network management service	Service information

Device Identification ← one change from NDCP v.1: NDCP version field (was reserved)

Vendor/Product Strings, Protocol Support, Application Support: no change from NDCP v.1, with the option of adding an entry in 'Protocol Support' for SpFi [a]

Device Configuration ← changed from NDCP v.1

Port Configuration ← compound of lower-level fields

Switching Table ← one change from NDCP v.1: 'Multicast enabled' bit, for logical addresses (was reserved)

Time-code Generation ← SpW: no change from NDCP v.1; SpFi: reserved

Protocol Information, Service information: no change from NDCP v.1

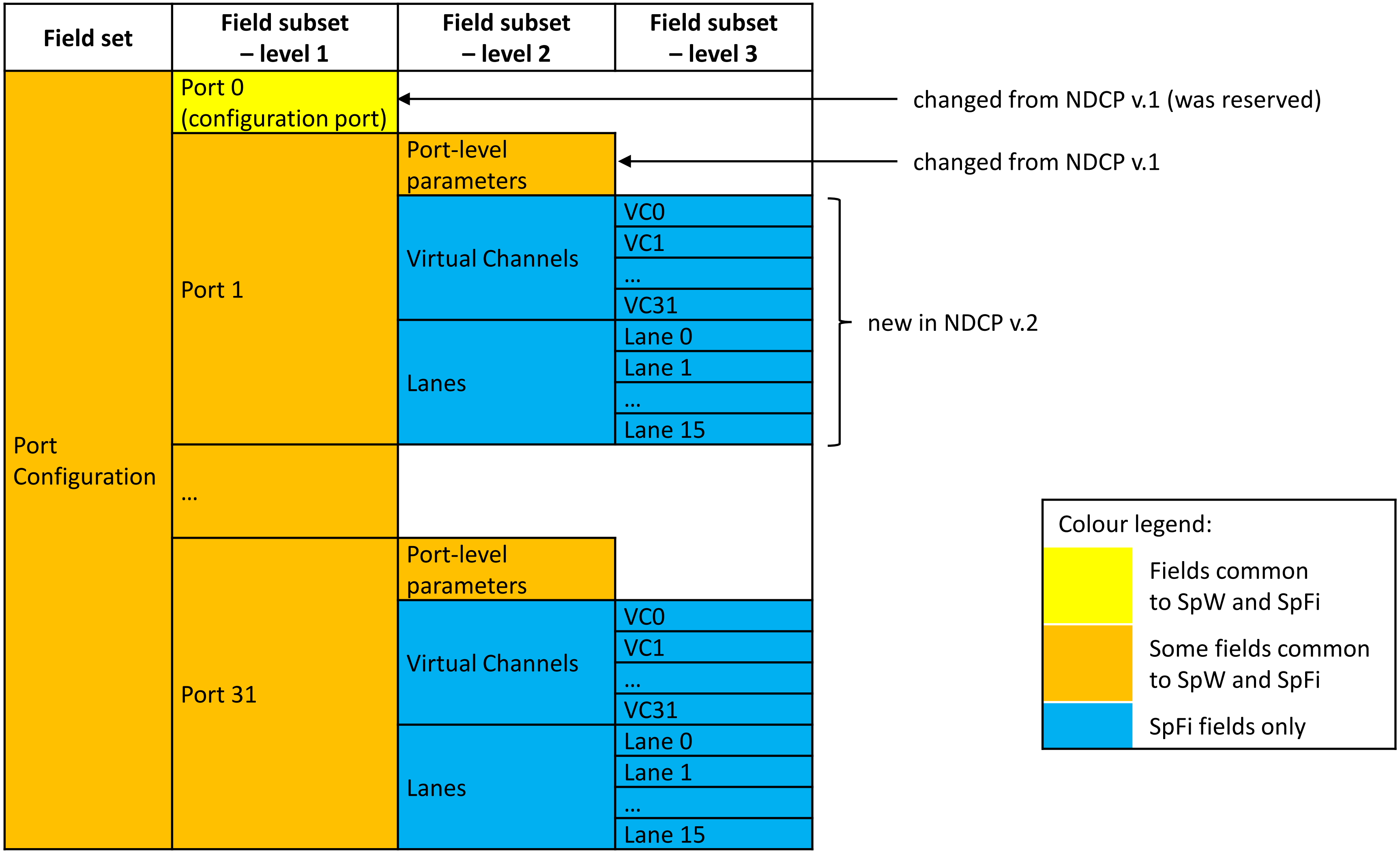
Color legend:

- 🟨 Fields common to SpW and SpFi
- 🟧 Some fields common to SpW and SpFi
- 🟩 SpW fields only

---

[a] to indicate the device supports SpFi. (This is optional: the information is available anyway from checking all port types.) The proposed value is 255, since it is not used by the Protocol ID standard for any protocol.

NDCP V.2 FIELD ORGANIZATION – LOWER LEVELS



Field set	Field subset – level 1	Field subset – level 2	Field subset – level 3
Port Configuration	Port 0 (configuration port)		
	Port 1	Port-level parameters	
		Virtual Channels	VC0
			VC1
			…
			VC31
		Lanes	Lane 0
			Lane 1
			…
			Lane 15
	…		
	Port 31	Port-level parameters	
		Virtual Channels	VC0
			VC1
			…
			VC31
		Lanes	Lane 0
			Lane 1
			…
			Lane 15

changed from NDCP v.1 (was reserved)

changed from NDCP v.1

new in NDCP v.2

Colour legend:

🟨	Fields common to SpW and SpFi
🟧	Some fields common to SpW and SpFi
🟦	SpFi fields only

NDCP FIELD ORGANIZATION PRINCIPLES

- Hierarchical order adopted for NDCP v.2:
  - Logical units of Device Information, Protocol, Device Configuration, Port Configuration on top
  - Technology-specific subdivision (SpW, SpFi) below that, as specific parts of the Port Configuration group
- Possible alternative with reverse order:
  - Technology-specific subdivision on top, with separate SpaceWire Protocol group and SpaceFibre Protocol group, and below them Port field subsets of the same technology type in a group
  - Simpler for homogeneous devices (having all ports of a single type: either SpW of SpFi) (and might use less memory for SpW)
  - However, problematic for nonhomogeneous devices with ports of both types, common port numbering, and common switching table
- Packing of device parameters in NDCP fields:
  - Fields are 32-bit long
  - Parameters may be of different sizes, often shorter
  - Parameters packed into fields so as to minimize memory usage while avoiding mixing read-only with read-write parameters in the same field

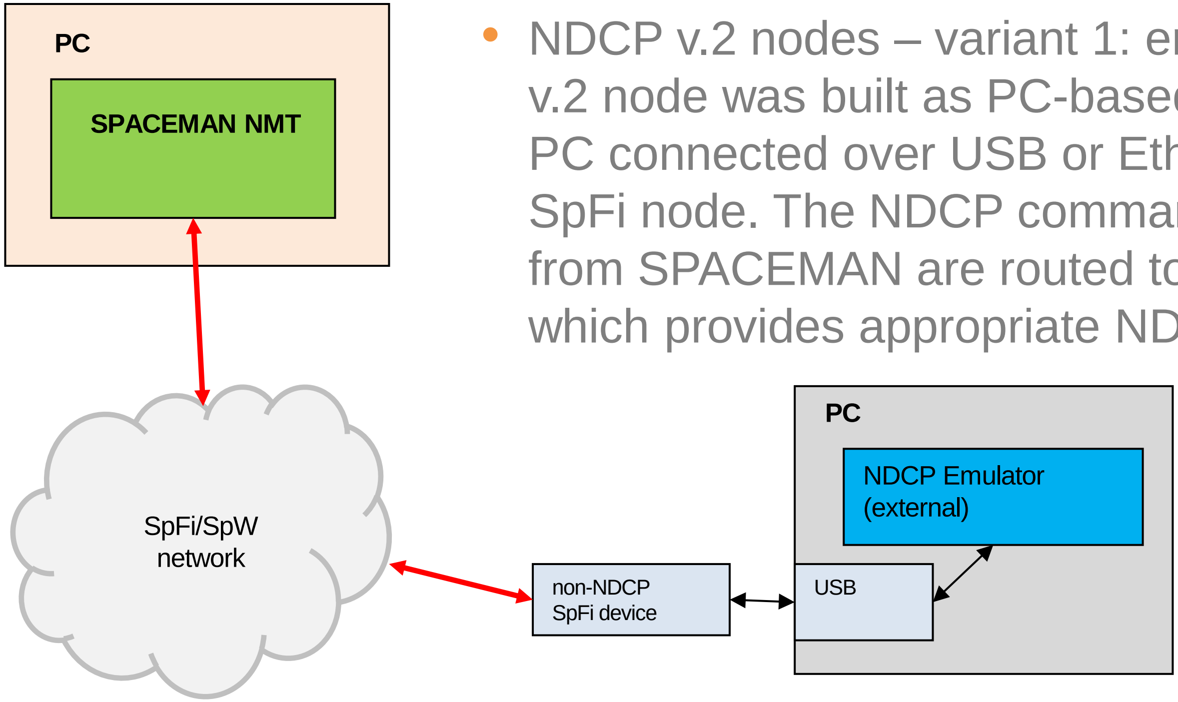
SOME OF THE SPFI PARAMETERS

(sub)field name	common for SpW /SpFi?	present in:		width (bits)	range	unit	information held	reset value	read-only?	static?	source	notes
		node	switch									
Device Configuration field set												
Device Type	Y	Y	Y	32	0-max	n/a	bits 8-31: device type code bits 0-7: field set definition version		RO	Y	ITTI	
Routing Switch Status	Y	N	Y	32	bit map	n/a	TBD		RO	N	[SpFi] Tab.5-37	Format not in [SpFi]
Broadcast time-out interval	N	Y	Y	32	0-max	clock cycle or µs (TBD)	time; if µs, then range is 0 to ca. 4295 s with minimal increment 1 µs		RW	N	[SpFi] Sect.5.8.12.2	Range and unit not in [SpFi]
Broadcast Channel association valid	N	Y	Y	1	0-1	n/a	1: device associated with Broadcast Channel 0: not associated with any broadcast channel	0 (TBD)	RW	N	[SpFi] Sect.5.8.12.1	
Broadcast Channel	N	Y	Y	8	0-255	n/a	broadcast channel number associated with the device	0 (TBD)	RW	N	[SpFi] Sect.5.8.12.1	
Invalid output port error	N	N	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Sect.5.8.8.3k	May be part of Routing Switch Status.
Port Configuration field set												
Port Type	Y	Y	Y	32	0-max	n/a	bits 8-31: port type code bits 0-7: field set structure version		RO	Y	ITTI	
Network Discovery	Y	Y	Y	1	0-1	n/a	attribute flag	1: use port for discovery; 0: don’t	RO	Y	[NDCP1]	
Number of Virtual Channels	N	Y	Y	5	0-31	virtual channel	actual number of virtual channels=value+1		RO	Y	ITTI	
Number of Lanes	N	Y	Y	4	0-16	lane	actual number of lanes=value+1		RO	Y	ITTI	
16-bit CRC error	N	Y	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Tab.5-37	
Frame Error	N	Y	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Tab.5-37	
CRC-8 error	N	Y	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Tab.5-37	
Sequence error	N	Y	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Tab.5-37	
Error recovery buffer empty	N	Y	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Tab.5-37	
Number of error recovery attempts	N	Y	Y	32	0-max	attempt	counter		RO	N	[SpFi] Tab.5-37	Range not in [SpFi]; [S&L] uses 6 bits
Link Reset Caused by Protocol Error	N	Y	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Tab.5-37	
Far-End Link Reset	N	Y	Y	1	0-1	n/a	status flag		RO	N	[SpFi] Tab.5-37	
Alignment State	N	Y	Y	2	0-2	n/a	one of 3 states		RO	N	[SpFi] Tab.5-37	'Reserved’ for single-lane ports
Bandwidth Credit Limit	N	Y	Y	32	0-max	word	limit on the number of words	imple-mentation	RW	N	[SpFi] Tab.5-36	Range not in [SpFi]; [S&L] uses 32 bits as well

NDCP V.2 DEMONSTRATOR IMPLEMENTATION (1)

- SPACEMAN NMT application was extended so as to:
  - support SpaceFibre for managed devices and management gateway devices
  - support the NDCP v.2 protocol in addition to NDCP v.1

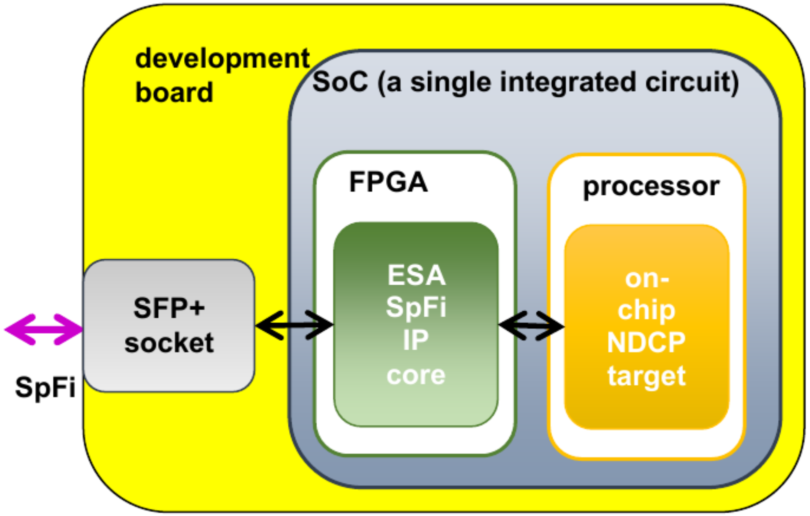
- NDCP v.2 nodes – variant 1: emulator of an NDCP v.2 node was built as PC-based software, with the PC connected over USB or Ethernet to a SpW or SpFi node. The NDCP commands the node gets from SPACEMAN are routed to the emulator, which provides appropriate NDCP responses



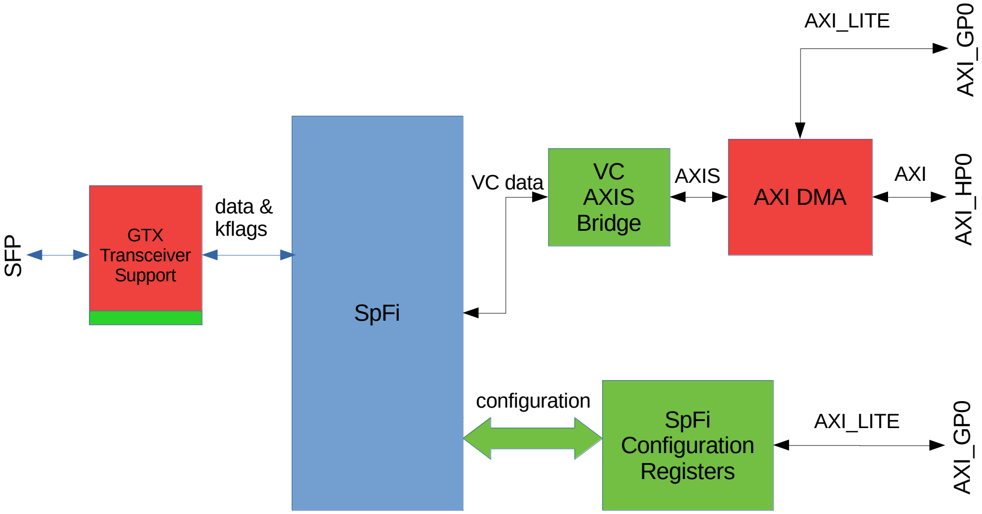
NDCP V.2 DEMONSTRATOR IMPLEMENTATION (2)

- NDCP v.2 nodes – variant 2: built on Zynq SoC development boards, with physical SpW and SpFi connectors added, and loaded with IP cores obtained from ESA IP Core Portfolio for:
  - SpW+RMAP (U.of Dundee)
  - SpFi (Gaisler)

on top of which ITTI added implementation of NDCP v.2 for SpW and SpFi

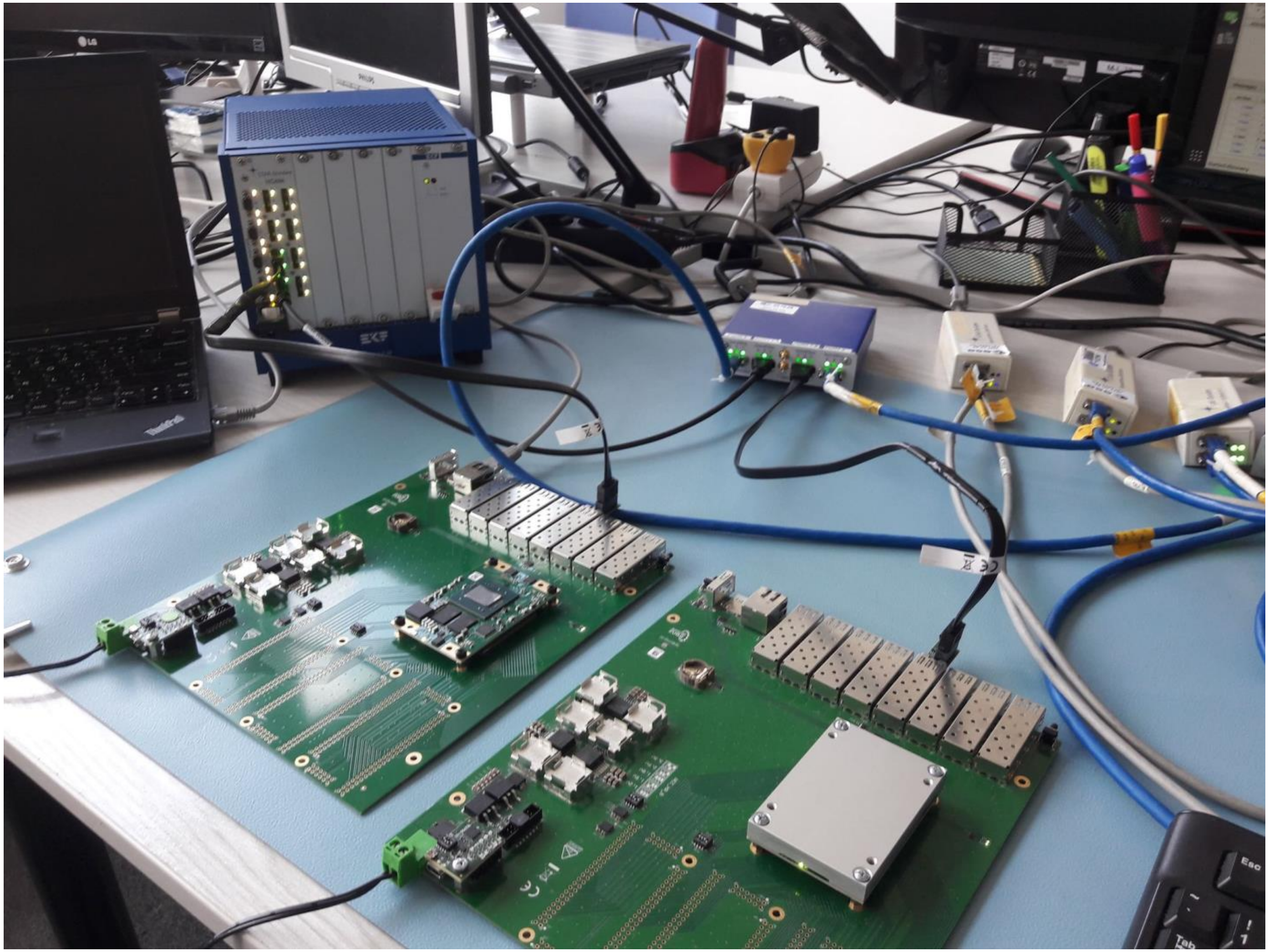


ZYNQ-BASED V.2-ENABLED SPFI NODE

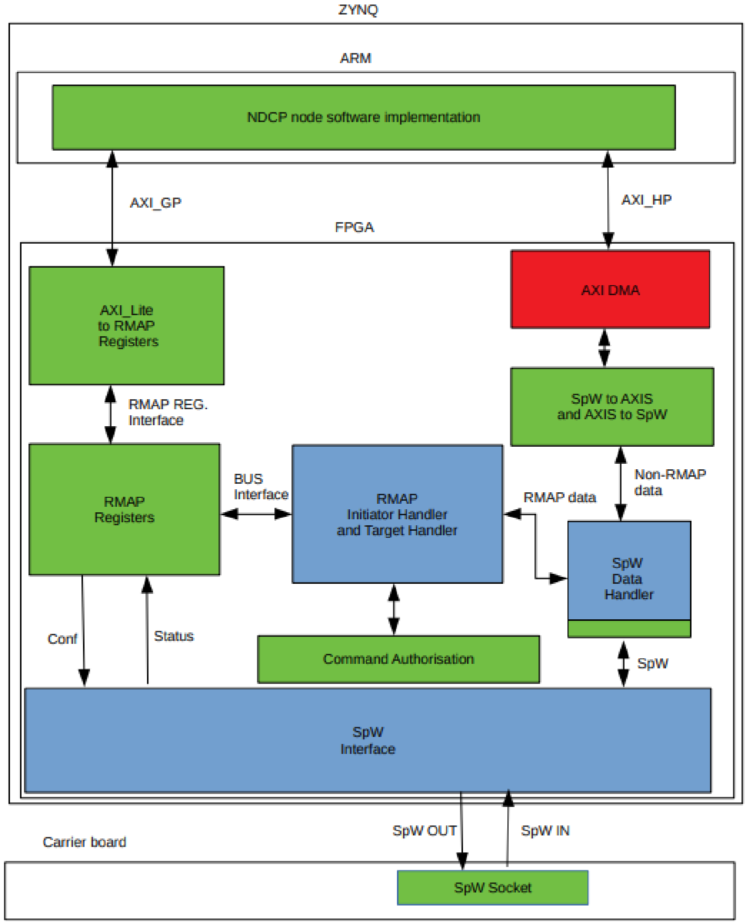


Colour legend: blue: ESA IP core
red: Xilinx IP core
green: ITTI IP core
red+green: generated by ITTI based on Xilinx templates

SPW/SPFI NETWORK WITH TWO V.2-ENABLED SPFI NODES



ZYNQ-BASED V.2-ENABLED SPW NODE



ZYNQ

ARM

NDCP node software implementation

AXI_GP

AXI_HP

FPGA

AXI_Lite to RMAP Registers

AXI DMA

RMAP REG. Interface

SpW to AXIS and AXIS to SpW

Non-RMAP data

RMAP Registers

BUS Interface

RMAP Initiator Handler and Target Handler

RMAP data

SpW Data Handler

Conf

Status

Command Authorisation

SpW

SpW Interface

Carrier board

SpW OUT

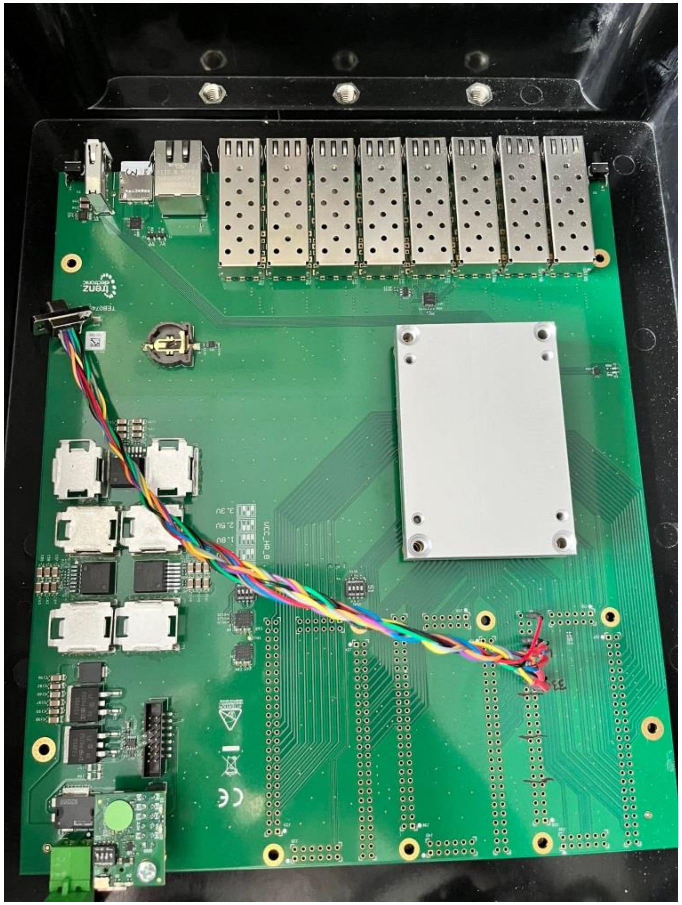
SpW IN

SpW Socket

🟦 Ipcore from ESA

🟩 ITTI ipcore

🟥 Xilinx ipcore



SPW NETWORK WITH ONE V.2-ENABLED SPW NODE

THANK YOU FOR YOUR ATTENTION

Krzysztof.Romanowski@itti.com.pl

ITTI Sp. z o.o., ul. Rubież 46, 61-612 Poznań, Poland
www.itti.com.pl