Technische Universität Braunschweig

INSTITUTE OF COMPUTER AND NETWORK ENGINEERING

Implementation of the SoCWire Protocol (SoCP) within the Dynamic Reconfigurable Processing Module

F. Bubenhagen[(1)], H. Michel[(1)], H. Michalik[(1)], B. Fiethe[(1)]
B. Osterloh[(2)]
W. Sullivan[(3)], A. Wishart[(3)]
J. Ilstad[(4)]
[(1)]IDA TU Braunschweig, [(2)]DSI GmbH, [(3)]Astrium Ltd UK, [(4)]ESA/ESTEC

INSTITUTE OF COMPUTER AND NETWORK ENGINEERING

Outline

1. Introduction
2. SoCWire
3. Dynamic Reconfigurable Processing Module
4. SoCWire Protocol (SoCP)
5. Conclusion

General Requirements for On-Board Processing

Future space missions require scientific instruments providing:

- High-performance on-board processing due to high resolution sensors
- High degree of flexibility and autonomy

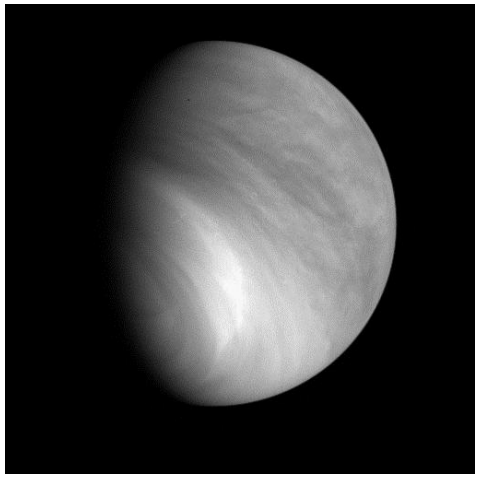
Boundary conditions:

- Limited downlink capacity
- Limited resources (mass, power)

Available On-Board Processing

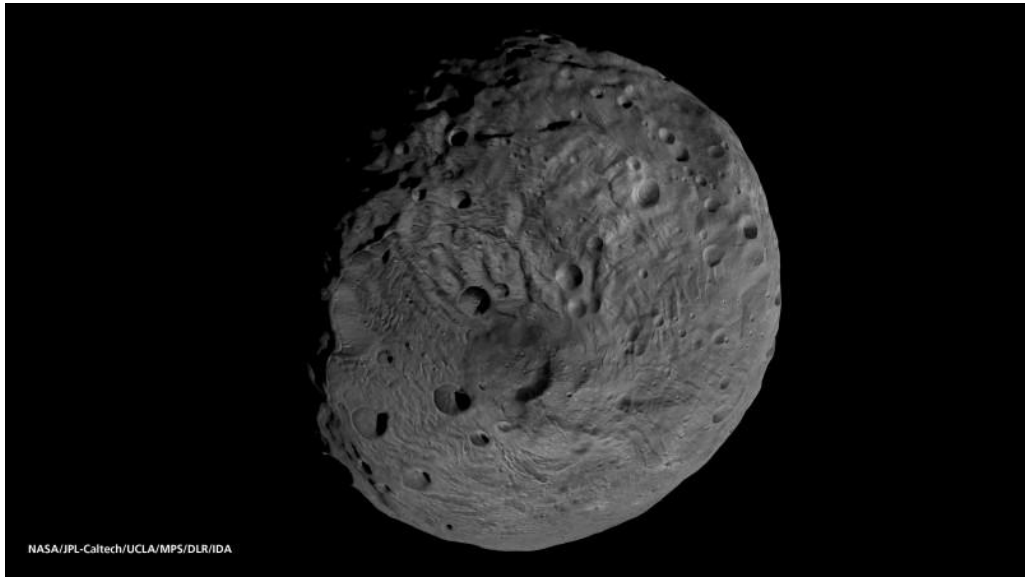
Based on Xilinx SRAM FPGAs:

- Implementation of advanced System-on-Chip
- Proven solution for scientific space instruments when appropriate mitigation techniques against radiation effects are applied
- High flexibility, reconfigurability has only been used in the development phase (VMC, DawnFC)



Venus Montoring Camera (VMC)

Dawn Framing Camera (DawnFC)



Future Adaptable On-Board Processing

- The FPGAs dynamic partial reconfiguration feature allows exchange of processing modules during runtime
  → reduced resource and power consumption
- Data transmission between interfaces (sensors, spacecraft, control), processing blocks and data memories requires network-on-chip infrastructure
- Already introduced the SpaceWire based SoCWire as an effective and viable solution for a System-on-Chip network
- Pure SoCWire network includes only the lower communication layers up to packet level

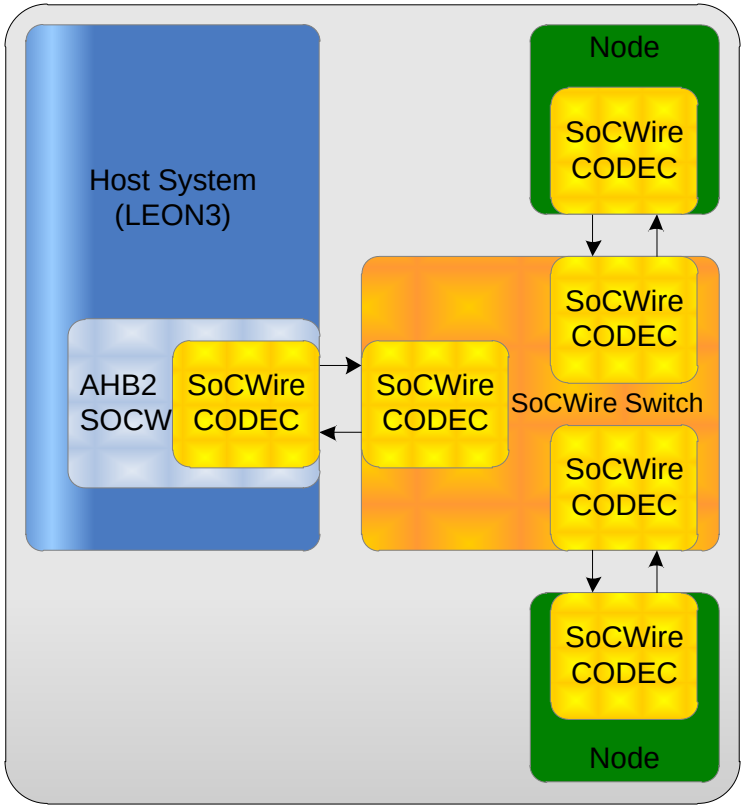
→ Further protocol is required for higher level communication between different nodes

Outline

1. Introduction
2. SoCWire
3. Dynamic Reconfigurable Processing Module
4. SoCWire Protocol (SoCP)
5. Conclusion

SoCWire - Features

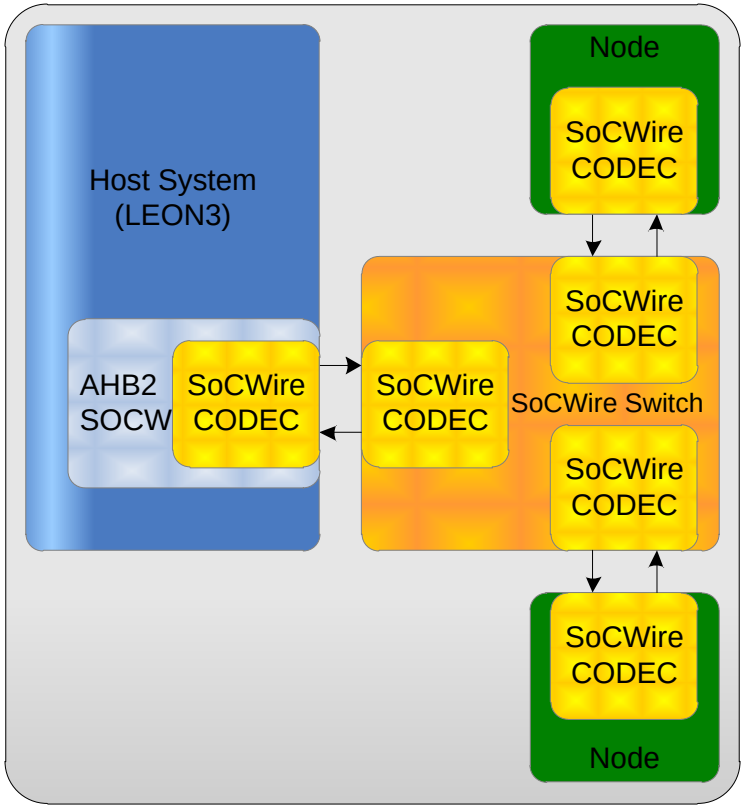
- SpaceWire based Network-on-Chip infrastructure
- Inherited SpaceWire features: link initialization, flow control, detection of link errors, link error recovery, low resource utilization)
- Parallel, synchronous interface with scalable data word width
- Provides isolation of Partial Reconfigurable Modules (PRMs) from host system within a reconfigurable FPGA
- Hot-plug ability



SoCWire - Network

SoCWire network consists of:

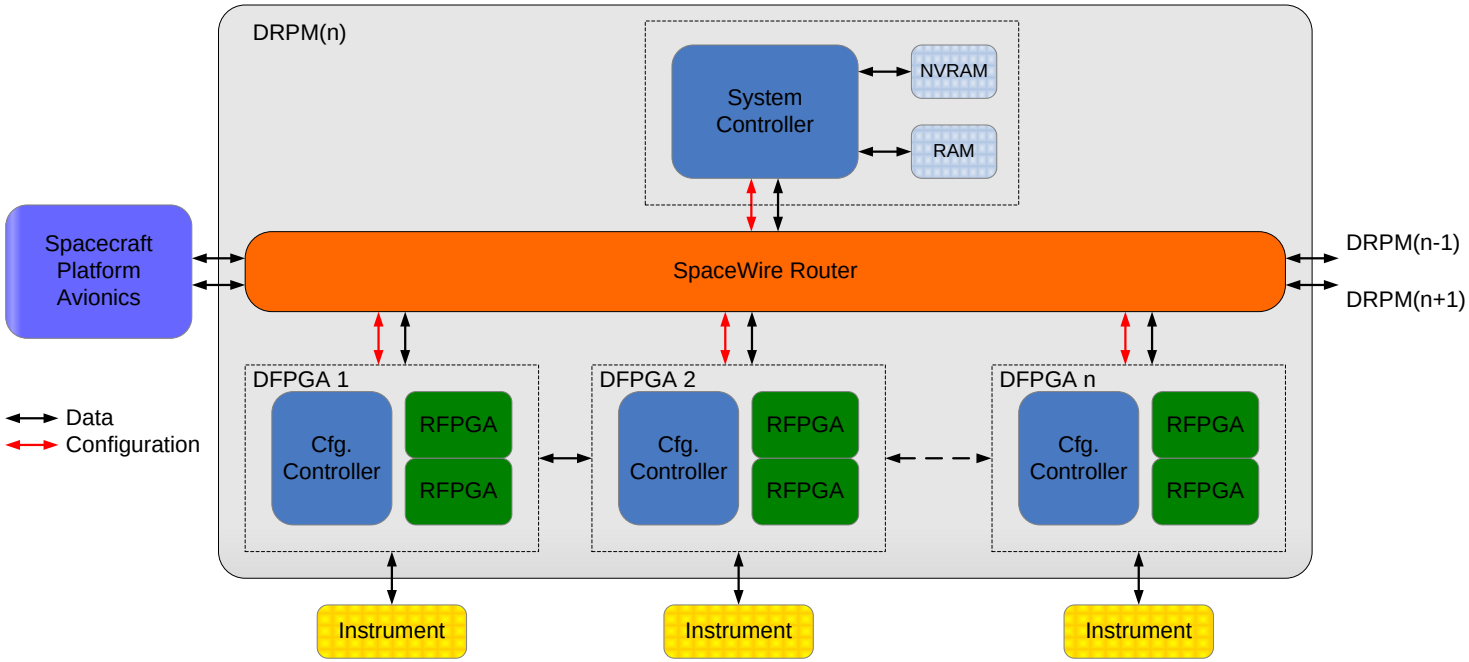
- SoCWire nodes (interface nodes, processing nodes, …)
- SoCWire Switch for node interconnection
- Host system:
  1. LEON processors (network control)
  2. AHB2SOCW bridge with DMA engine (allows high speed data transmission between SoCWire side and memories/interfaces on AMBA bus)



Outline

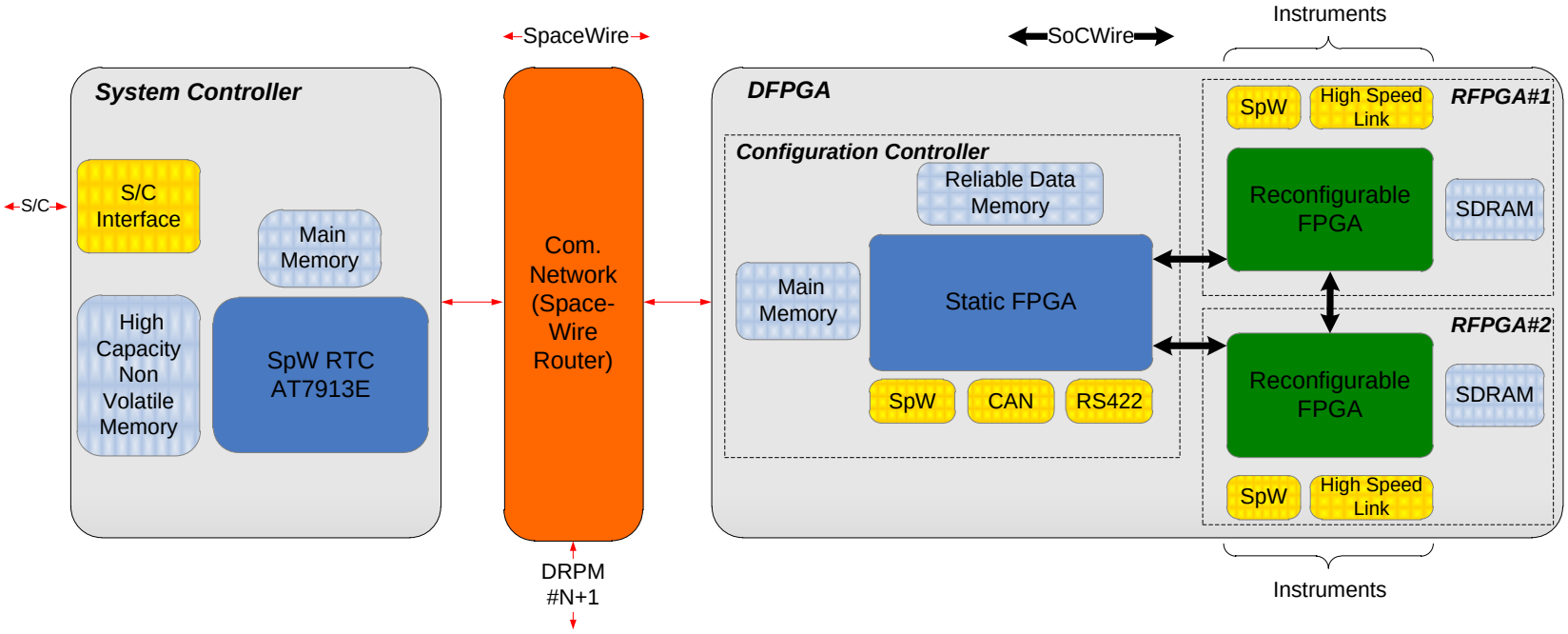
1. Introduction
2. SoCWire
3. Dynamic Reconfigurable Processing Module
4. SoCWire Protocol (SoCP)
5. Conclusion

DRPM - Architecture



- Dynamic Reconfigurable Processing Module (DRPM) is currently in development and validation under ESA contract
- Primary objective of the study is to provide a development environment demonstrating the feasibility of reconfigurable FPGA technology for flight programs
- Modules of the demonstrator platform are equipped with devices and interfaces also available in space qualified versions

DRPM - Demonstrator



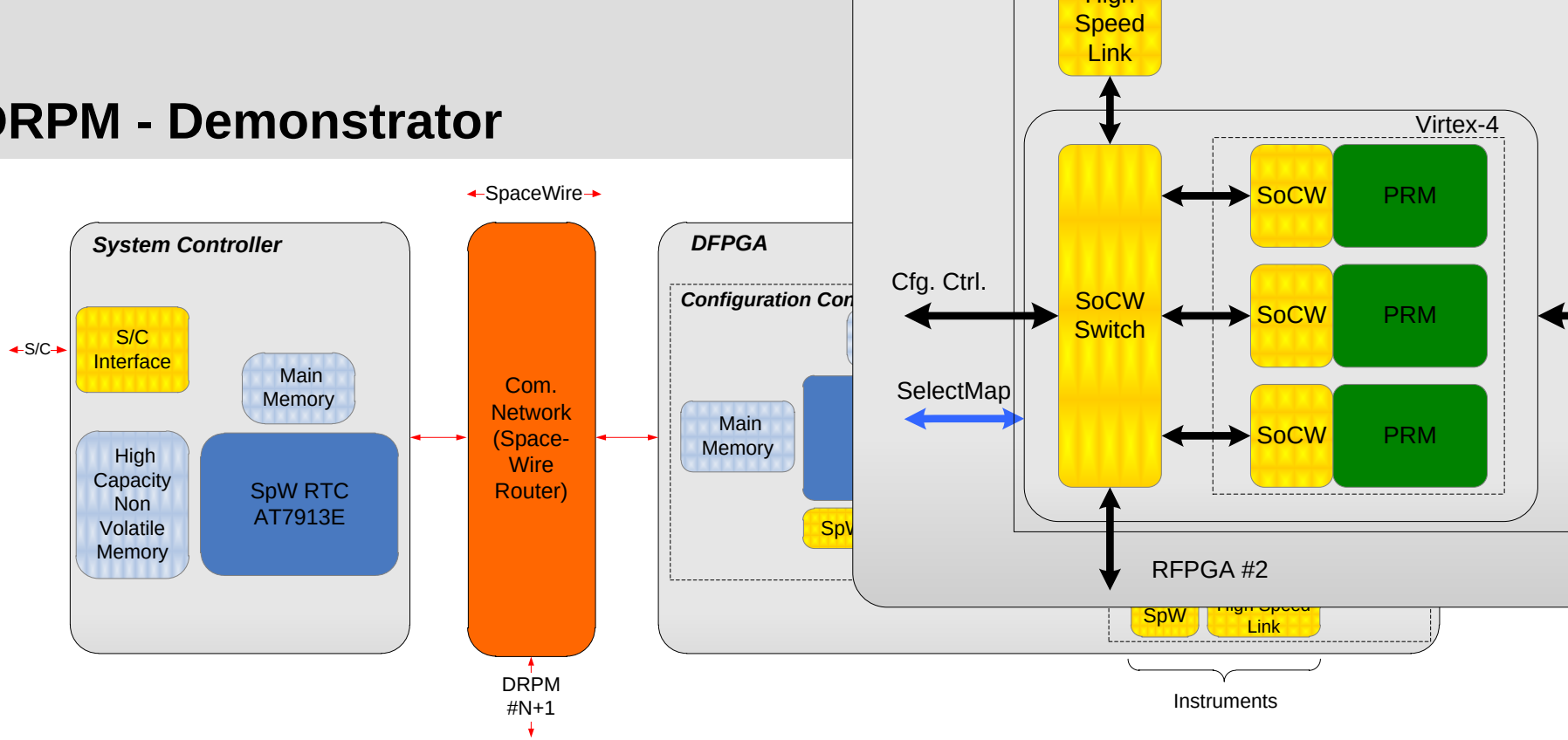
Modular concept:

- System Controller (Overall system control and supervision)
- SpaceWire Router (Sub-module interconnection and expandability)
- Dynamically reconfigurable FPGA (DFPGA) module (Processing unit)

→ Processing capacity scalable by addition of further DFPGA modules

→ Redundancy by addition of DFPGA modules with identical functionality

DRPM - Demonstrator



DFPGA consists of:

- Configuration Controller (CC) for configuration management and includes the SoCWire network host system (LEON processor)
- 2 Reconfigurable FPGA (RFPGA) modules equipped with Virtex-4 devices
- “External ”SoCWire network for CC and RFPGA interconnection
- “Internal” SoCWire network within FPGA(s) for processing and interface node interconnection

Outline

1. Introduction
2. Dynamic Reconfigurable Processing Module
3. SoCWire
4. SoCWire Protocol (SoCP)
5. Conclusion

SoCWire Protocol (SoCP) - Requirements

- Interconnected SoCWire CODECs represent only the physical link with a protocol defined up to packet level (Data + EOP)
- A higher protocol is required to allow data to traverse the network

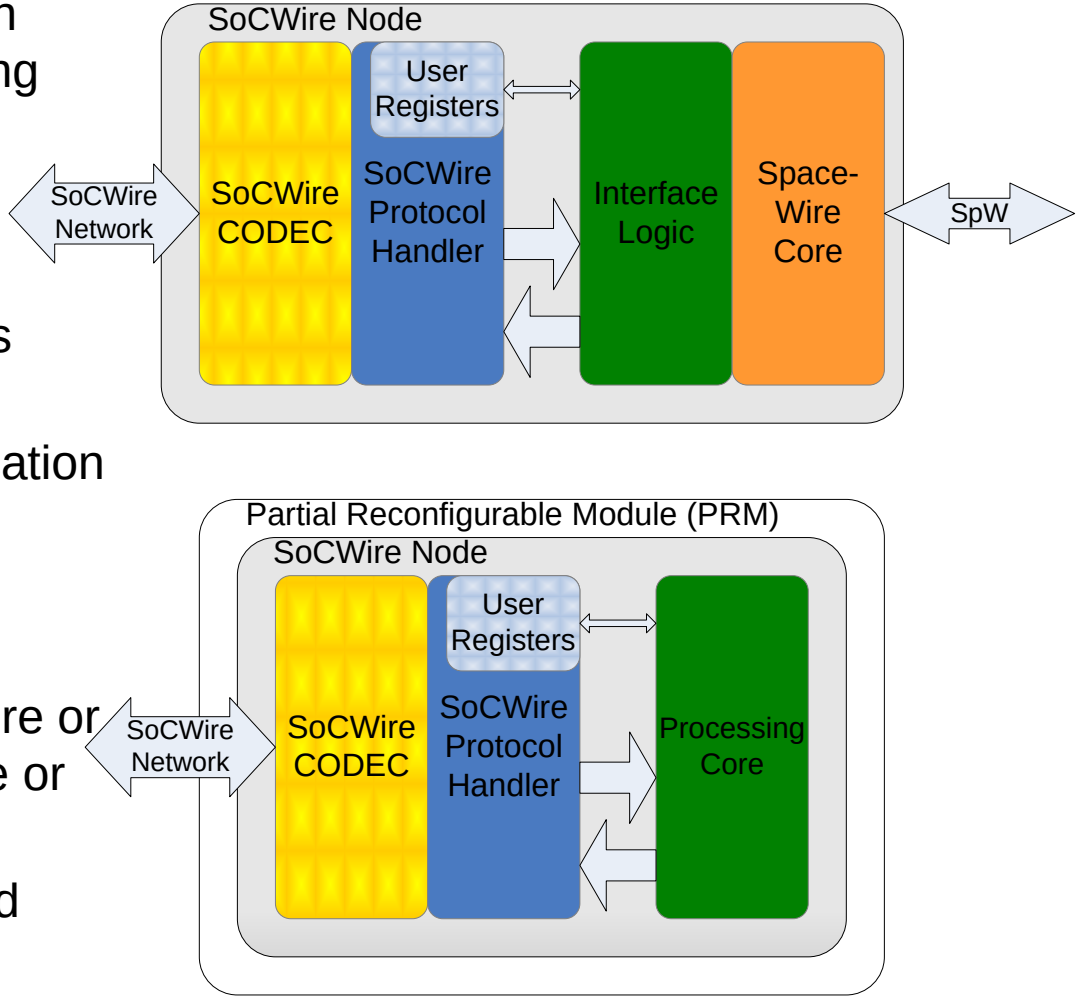
- Requirements which apply to a protocol to be considered:
  - Hardware implementation
  - Minimal resource usage (instantiation for each SoCWire node required)

- Constraints for protocol implementation:
  - Number of SoCWire Switches in a path is limited to three
  - Port 0 of a SoCWire Switch must be the route to the host system
  - Data width is either 16 or 32bit
  - Number of ports in a SoCWire Switch is limited to 16

SoCWire Protocol (SoCP) - Implementation

- SoCP handler is placed between SoCWire CODEC and processing core or interface logic

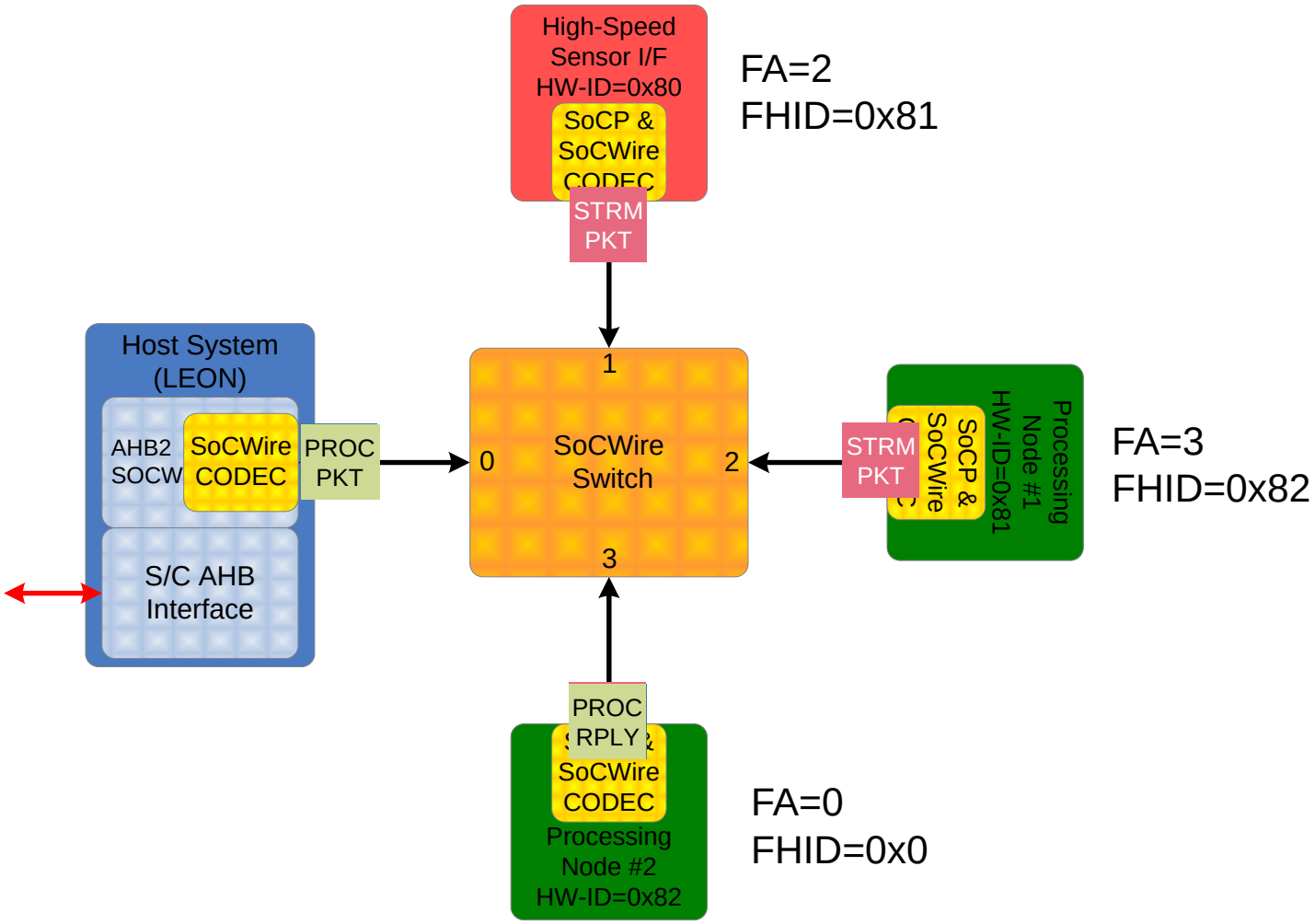
- SoCP handler features:
  - Command and Reply format is inspired by RMAP
  - Reply to requests (from application S/W)
  - Supply the processing core or interface logic with data
  - Send data from processing core or interface logic to another node or host system
  - Provide register for control and general purpose use



SoCWire Protocol (SoCP) - Transaction Types

Transaction Type	Description
Process and Reply	Packet generated by host system, processed by a node with reply sent back to host system
Write Register and Reply	Write a register (Reply is sent by node)
Read Register and Reply	Read a register (Reply is sent by node)
Streaming Data Transmission	Streaming data from a source to a destination node
Plug and Play Init Message	Initialization message sent by a node to host system

SoCWire Protocol (SoCP) - Example



SoCWire Protocol (SoCP) - Process/Streaming Packet Definition

Packet definition for *Process Request/Reply* transfers and *Streaming Data* transfers

Address:

- Up to three addresses for node to node or host to node transfers
- Always three addresses with value 0 for node to host transfers

Hardware ID:

- Unique ID of every node (destination)

Transaction/Packet counter:

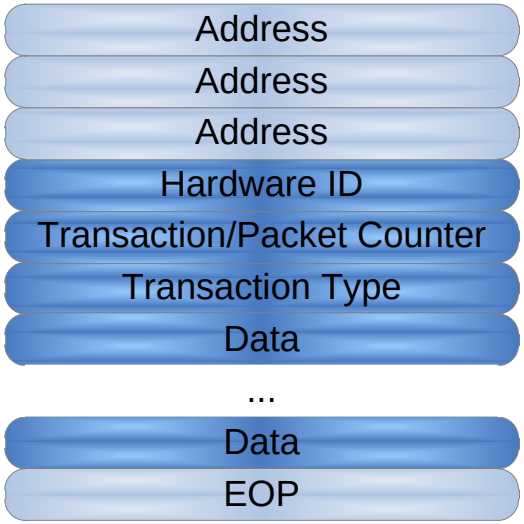
- Sequence Counter

Transaction Type:

- Defines the type of transaction and contains status information with replies

Data:

- Up to 2064 data words (*Process Request/Streaming* transfer)



SoCWire Protocol (SoCP) - Register Packet Definition

Packet definition for *Register Read/Write Request/Reply* transfers

Address, Hardware ID, Transaction/Packet Counter and Transaction Type:

- Same as for *Process Request/Reply and Streaming* Transfer

Register Address:

- One of up to 16 user/control register

Register Data/Data Inverted:

- Data for the addressed register

EOP:

- Every packet is terminated by an EOP token

SoCWire Protocol (SoCP) - Error Detection

- *Hardware ID*, *Transaction/Packet Counter*, *Transaction Type* and *Register Address* fields are protected by a parity bit
- *Register Data* is protected by an inverted copy of the register value to be read or written
- The *Hardware ID* is verified within the destination core and the packet is only processed when there is a match between packet *Hardware ID* and core *Hardware ID*

Limitations:

- If protection of *Process* data or *Streaming Transfer* data is required, then this has to be done within the processing core
- No autonomous error handling, so e.g. a retry mechanism has to be implemented in the software protocol handler running on host system

SoCWire Protocol (SoCP) - Resource Utilization

	SoCP IP Core		SoCW CODEC IP core wo/ RAM	
Device	**Cells/Slices**	**Utilization[%]**	**Cells/Slices**	**Utilization[%]**
XQR4VSX55	115	0.47	272	1.11
XQR4VLX200	115	0.13	272	0.31
XQ5VFX130T	67	0.33	160	0.78
RTAX2000S/SL	269	0.83	754	2.34
RT3PE3000L	368	0.49	932	1.24

Comparison of resources required by SoCP IP core and SoCWire CODEC IP core (both configured to 16bit data word width)

→ Less than 0.5% resources of a SRAM based FPGA

→ SoCP IP core requires only half the resources of a SoCWire CODEC

Outline

1. Introduction
2. Dynamic Reconfigurable Processing Module
3. SoCWire
4. SoCWire Protocol (SoCP)
5. Conclusion

Conclusion

- DRPM architecture is an effective and viable implementation of a reconfigurable hardware for future space missions
- SoCWire provides a fault-tolerant, high speed System-on-Chip infrastructure for exchange of packets between processing and interface nodes
- SoCWire Protocol:
  - An efficient protocol implementation
  - Tailored to the needs of a SoCWire network
  - Small resource utilization
  - Inherent error detection capabilities
- Key features of DRPM architecture including SoCWire and SoCP will be implemented within the Solar Orbiter PHI instrument

THANK YOU FOR YOUR ATTENTION!

Questions?