PicoSkyFT based RTU Demonstrator

ESA PECS Contract: 4000120900/17/NL

TEC-ED & TEC-SW Final Presentation Day - 11th of December 2018, ESTEC/NL

Dejan Gačnik (CTO), dejan.gacnik@skylabs.si

skylabs

Outline

- picoRTU-D System Objectives
- Distributed vs centralised architecture
- System trade-off
- picoRTU-D Architecture (RTI, UI, RTC, FW Architecture)
- picoRTU Assembly concept
- picoRTU System ecosystem
- picoSkyFT IP Core in NG-Medium
- Conclusion
- Follow up

skylabs

PicoRTU-D System
Objectives

- Reduce costs and schedules
  - Common interface (ECSS-E-ST-5015C – CANbus extension protocol)
  - Standardised user interfaces (ECSS-E-ST-50-14C - Spacecraft discrete interfaces)
  - Qualified units (HW and FW, only FW parameterisation is required)
- Decentralization
  - Miniaturisation and local intelligence
  - Consider SAVOIR RTU Functional and Operability Requirements
- Modularity
  - Off-the-shelf qualified picoRTU units
  - Efficiently copes with late requirement changes, as it is a matter of firmware reconfiguration with changing or adding new units.
- Enhancing European technology independence
  - Using mostly EU components to build picoRTU-D

skylabs

Distributed vs centralised architecture

Definition: “A distributed system is a network that consists of autonomous computers that are connected using a distribution middleware.”

- Decentralization of the system with a modular approach benefits in:
  - saving overall mass,
  - harness reduction and simplifications,
  - lower verification efforts and accelerates AIT/AIV
  - decreases development times of the S/C.
- Placing units near the source of measurements
  - the signal quality is improved by reducing noise coupling or
  - placed close to the sink, and
  - the overall EMI is improved.
- The modular system based on off-the-shelf qualified (HW and SW) generic units has additional important advantages:
  - Reduces developme times, as only parametrization of firmware needs to be done, and picoRTU will already fulfil requirements of discrete interfaces for acquisition and actuation.
  - Eases verification during AIT/AIV, thanks to picoRTU-EGSE supporting software.
  - Modular system efficiently copes with late requirement changes, as it is a matter of firmware reconfiguration with changing or adding new units.

skylabs

System trade-off

- **Communication Interfaces for RTI**
  - Evaluation: CAN, SpW (EIA-644), TTTech ETH, M-LVDS (EIA-899), RS422/485, FlexRay, JESD204B, MIL-1553B
  - Specifications:
    - multi-point interface and without additional routers or control devices (lower complexity)
    - adequate number of units ~32 or more
    - data throughput < 10 Mbps
    - heritage
  - CAN ISO 11898-2 PHY interfaces and ECSS-E-ST50-15C application layer protocol was selected as the most adequate:
    - higher number of supported devices on the bus,
    - already supported error detection mechanisms,
    - high maturity in the automotive industry,
    - flight heritage on several missions,
    - already available rad-hard and rad-tolerant parts,
    - Up to 1 Mbps deems sufficient (possibility of 10 Mbps with CAN-FD).
    - in company know-how.

CANopen®

EUROPEAN COOPERATION
ECSS
FOR SPACE STANDARDIZATION

skylabs

System trade-off cont'd

- **RTU User Standard Interfaces**
  - Evaluation: ESCC-E-ST-50-14C interfaces as LV-HPC, HV-HPC, HC-HPC, ASM, TSM1, TSM, BSM, BDM, Serial digital interfaces specifically ISD, OSD and BSD.
  - Selection criteria:
    - Select the most relevant interfaces, based on research of publicly available information.
  - Interfaces selected based on research the most commonly used:
    - ASM
    - TSM1 and TSM2
    - BSM
    - HPC-HV
    - HPC-HC
    - RS485/422
    - Digital IOs

skylabs

EUROPEAN COOPERATION
ECSS
FOR SPACE STANDARDIZATION

System trade-off cont'd

- **User Interfaces Types and picoRTU unit variants**
  - Evaluation: Identify the most optimum selection of number of user interfaces and variety of picoRTU units.
  - Selection criteria:
    - Use as less as possible of various picoRTU units.
  - Final results units and respective number of interfaces:

picoRTU-D variant \ Interfaces	CAPs	PowerIO HVCs	Digital IOs	RS485/422 interfaces
picoRTU-AQ (Analog acquisition unit)	20	0	0	0
picoRTU-HP (High power unit)	0	8	0	0
picoRTU-COMM (Digital and communication unit)	0	0	64	12
picoRTU-D (Generic unit)	10	4	14	2

skylabs

System trade-off cont'd

- Final results of analysis: interfaces vs. number of units per given reference mission:

	picoRTU system				Conventional solution		
Mission	Nb. of boards	Volume (dm^3)	Mass (kg)	Pwr (W)	Volume (dm^3)	Mass (kg)	Pwr (W)
LISA Pathfinder	54	8.35	13.5	89.1	N/A	N/A	N/A
WorldStar	25	3.86	6.25	70	14.7	8	10
Helios II	33	5.10	8.25	92.4	19.7	12	12
Spot4	1	0.15	0.25	2.8	4.5	2.4	2.1
PROBA-3	18	2.78	4.5	29.7	N/A	N/A	N/A

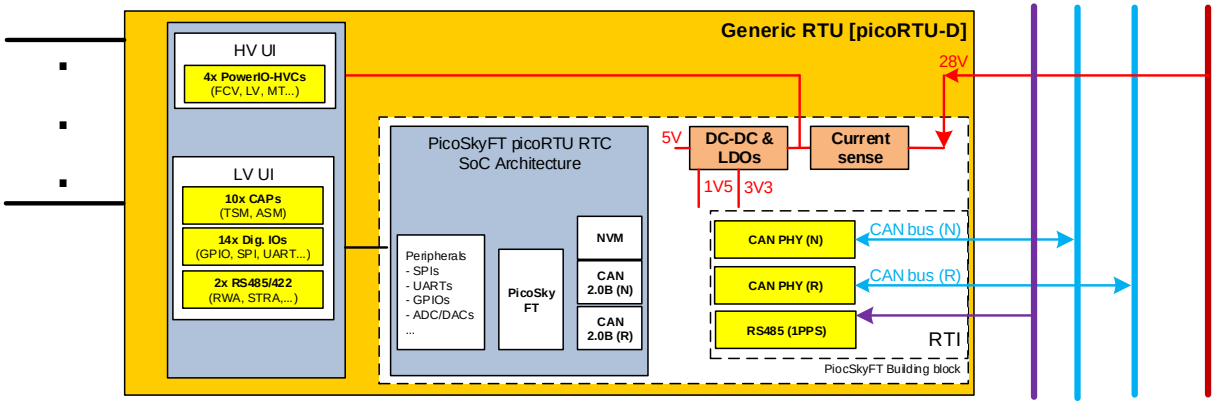
Power consumption is something that need to be revised.

skylabs

picoRTU-D

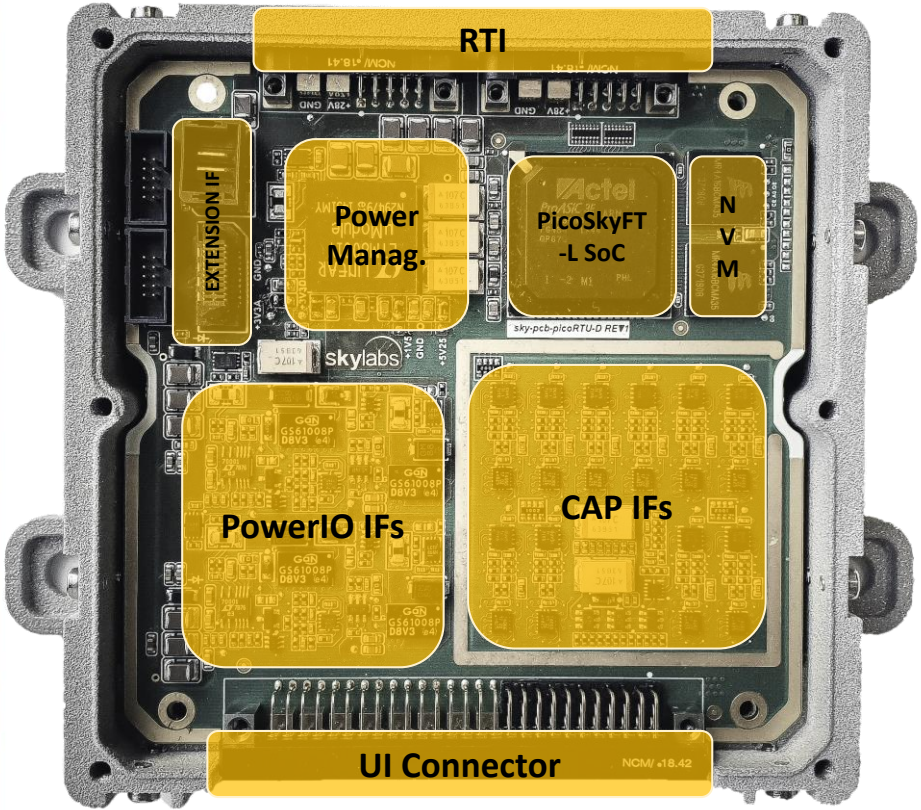
Architecture

- picoRTU-D designed and optimized for distributed system, enhanced by local intelligence.
- Modular system over common RTI: redundant CAN, 28V power supply, 1PPS input.
- Scalable by stacking two or more units into a stack: internal extension connector
- User interfaces: ComboAnalogPin-CAP, PowerIOs, digital IOs, several communication interfaces (e.g. RS485/422, TWI, UART…)
- Precise time synchronization accuracy between picoRTU units (less then 100ns).
- Rich local housekeeping acquisition and conditioning.
- Dedicated internal extension interface for stack (28V power rails, CAN interface, and 1PPS)



picoRTU-D

Architecture cont'd



	Description	Comment
Discrete User interfaces	4x PowerIO-HVC 10x CAP (organized in groups by 4+4+2) 14x Digital IO	High voltage (28V) ASM, TSM1, BSM, Sun sensors 3.3V IOs (GPIO, SPI, TWI, UART)
Communication User Interfaces	4x RS485/422 links 2x UART 2x SPI 2x TWI	Available on Digital IOs Available on Digital IOs Available on Digital IOs
PowerIO capability	< 5A per powerIO @ 28V	Maximum current depends on thermal constraints within the stack.
Remote Terminal Interface	CAN bus for TM/TC	ECSS-E-ST-50-15C comp.
On-board storage NVM SRAM	 2 MB MRAM (1M x 32) 24 kB (FPGA SRAM)	 Firmware and parameters Data memory
A/D resolution	12 bits	
Supply voltage	Unregulated 28 V DC	(from 24V to 32V)
Power consumption	< 2.8W @ 85°C	without PowerIO
Single unit dimensions:	107 x 107 x 13.5 mm	
Operation temperature	-20°C to +60°C	
Storage temperature	-40°C to +85°C	
Mass:	< 250 g	

skylabs

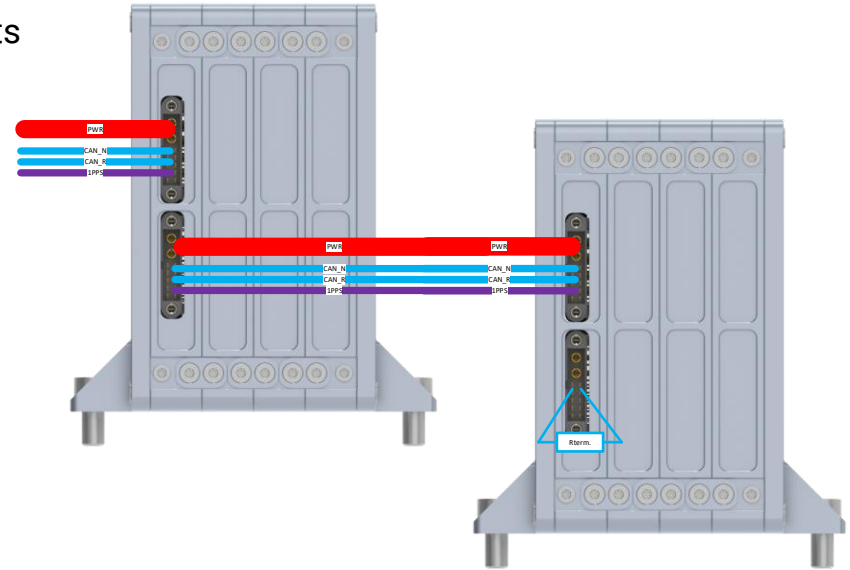
picoRTU-D RTI

skylabs

picoRTU-D

Remote Terminal Interface

- Redundant CAN interface (follows ECSS-E-ST-50-15C)
  - Based on CANopen and tailored to ECSS
    - Described in CIA 301 specification
    - Used in industrial and automotive applications
  - Defines redundancy mechanism
  - Also contains hardware requirements
    - Topology
    - Transceiver characteristics
    - Fault tolerance
  - Flight proven/heritage
- Common power bus topology (overcurrent protection on unit level)
- 1PPS (RS485) interface for high accuracy time synchronization



skylabs

picoRTU-D

Remote Terminal Interface - CANopen

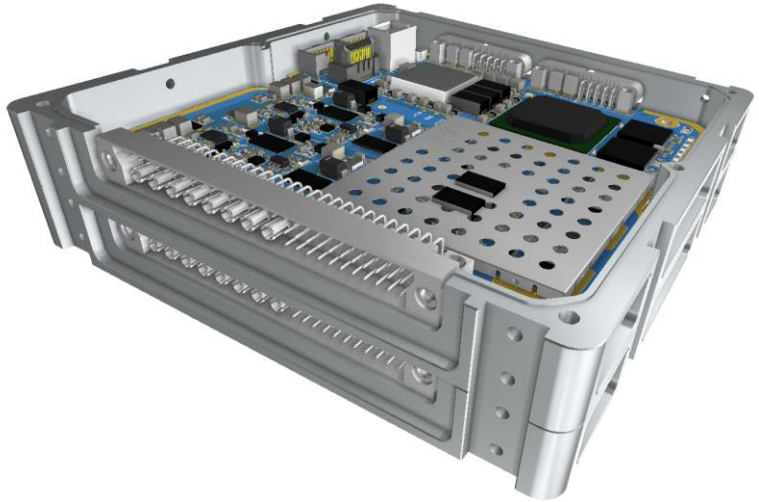
- Object dictionary is the TM/TC channel
- Data from dictionary is accessed through:
  - SDO protocol - on demand (request/response)
  - PDO protocol (unsolicited)
    - Can be cyclic (SYNC, PPS) or asynchronous (on change)
- Larger data transferred over SDO Block transfer service
  - TM/TC list upgrade
  - Firmware patching
  - Logs download
  - TM list acquisition results download
- Periodical heartbeat messages
  - CANopen master knows node is alive
  - Redundancy management according to ECSS-E-ST-50-15C

- 244 objects in the dictionary
  - 54 Housekeeping TM
  - 155 UI TM
  - 10 TM lists
  - 5 System logs
  - 1 System TC entry (256 TC)
  - 1 UI TC entry (256 TCs)
  - 15 HW configuration
  - 3 Binary transfers

CANopen®

EUROPEAN COOPERATION
ECSS
FOR SPACE STANDARDIZATION

skylabs



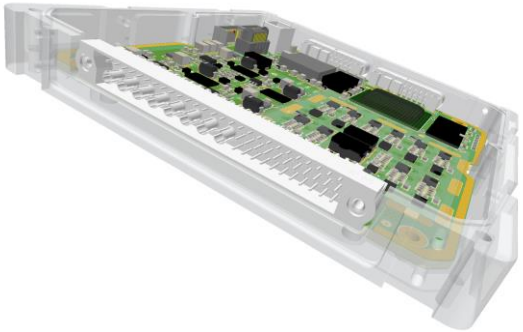
picoRTU-D UI

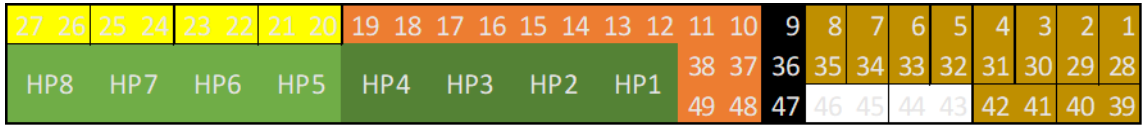
skylabs

picoRTU-D

Standardised user interfaces

- Multifunctional CAP (ComboAnalogPin) interface
  - 20 single ended analog inputs, or
  - 10 differential analog inputs
- PowerIO interface
  - 4 Half-bridge (Arm/Fire feature)
  - 2 Full-bridge configuration
- Digital IO interface
  - 14 digital IOs, configured as IO, UART, TWI, etc…
- Communication interface
  - 2x Rx and 2c Tx (RS485/RS422)





4x powerIO (High Side only, PIO_HSx)
4x powerIO (Low Side only, PIO_LSx)
GND
10x CAP
14x Digital
RS485 (2x RX & 2x TX)
Not Connected

skylabs

picoRTU-D

Standardised user interfaces - CAP

- CAP interface requires two physical connector pins and compliant with ECSS-E-ST-50-14C:
  - ASM for continuous measurement of voltages
  - TSM for temperature sensors
  - Bi-level Discrete monitor (BDM),
  - Sun-sensor equipment
- All CAP interfaces are acquired synchronous and time-tagged (not multiplexed)
- On-board CAPs are organized in groups
  - enables cross-strapping functionality between different groups (unit or system level).
- CAP interface IF is configurable via picoRTU-D configurator.
- Each CAP channel can be configured as a differential or single ended interface according to the desired function.
- Resistance range for TSM is parametrizable per channel through product order. (2 channels adj. current source)
- CAP interfaces have a recalibration feature.

	CAP Pin 1	CAP Pin 2	Sample rate [ksps]	Resolution	Input voltage [V]
Differential ASM	Diff_P	Diff_N	2[3]	12 bit	0 to 5
Single ended ASM	ASM1	ASM2	2	12 bit	0 to 5
Single ended TSM	TSM1	TSM2[1,2]	2	12 bit	0 to 5
BDM	BDM	GND[1]	2	12 bit	0 to 5
Sun sensor	SS1	SS2[1]	2	12 bit	0 to 5 mA

[1] Signal depends on the grounding scheme (e.g. common ground). Within the same group this signal is configured as input and needs to be connected to GND.
[2] TSM1 and TSM2 in table does not denote the interface type as of ECSS standard; referenced here only as a first and second sensor input.
[3] Higher resolution can also be configured.

picoRTU-D

Standardised user interfaces - PowerIO

- PowerIO interface supports:
  - HV-HPC (ECSS-E-ST-50-14C)
  - HC-HPC (ECSS-E-ST-50-14C)
  - DC motor,
  - Brushless motor,
  - Stepper motor
- PowerIO interface IF is configurable via picoRTU-D configurator.
- When using a polarity independent load (e.g. resistive heating elements) HW redundancy can be achieved simply by connecting load between two PowerIO lines
- Arm/Fire functionality supported by using two different HW switches that have independent TC commands and configurable dead time in between.
- Parametrizable settings such as min/max fault voltage, and current limit.

	Current limit / SC current [mA]	Max. Fault Voltage [V]
HV-HPC	≤ 180 (@29V) / 400	29
HC-HPC	≤ 600 (@29V) / 1000	29
DC motor driver	Up to 2000 (@29V) / 3000	29
Brushless motor driver[1]	Up to 2000 per winding (@29V) / 3000	29

skylabs

picoRTU-D

Standardised user interfaces - DigitalIO & Comm IF

- On top of Set and Reset functionality, digital IO interface configuration supports several communication interfaces, such as:
  - SPI with several Chip Selects (SPI4SPACE supported with external phy. circuit), TWI, UART, GPIO, Dallas 1-Wire temperature sensors
- Digital IO interface type is configurable via picoRTU-D configurator.
- These pins are supported by a 3.3 V LVCMOS/TTL level interface, connected to the FPGA through a series resistor (not compliant with ECSS-E-ST-50-14C).
- RS485/422 interface is supported through dedicated UI connector pins. Two configurations are possible:
  - 2 half duplex configurations or
  - 1 full duplex configuration.
  - When an interface is configured to work in a RS422 network its enable signals are toggled accordingly.

skylabs

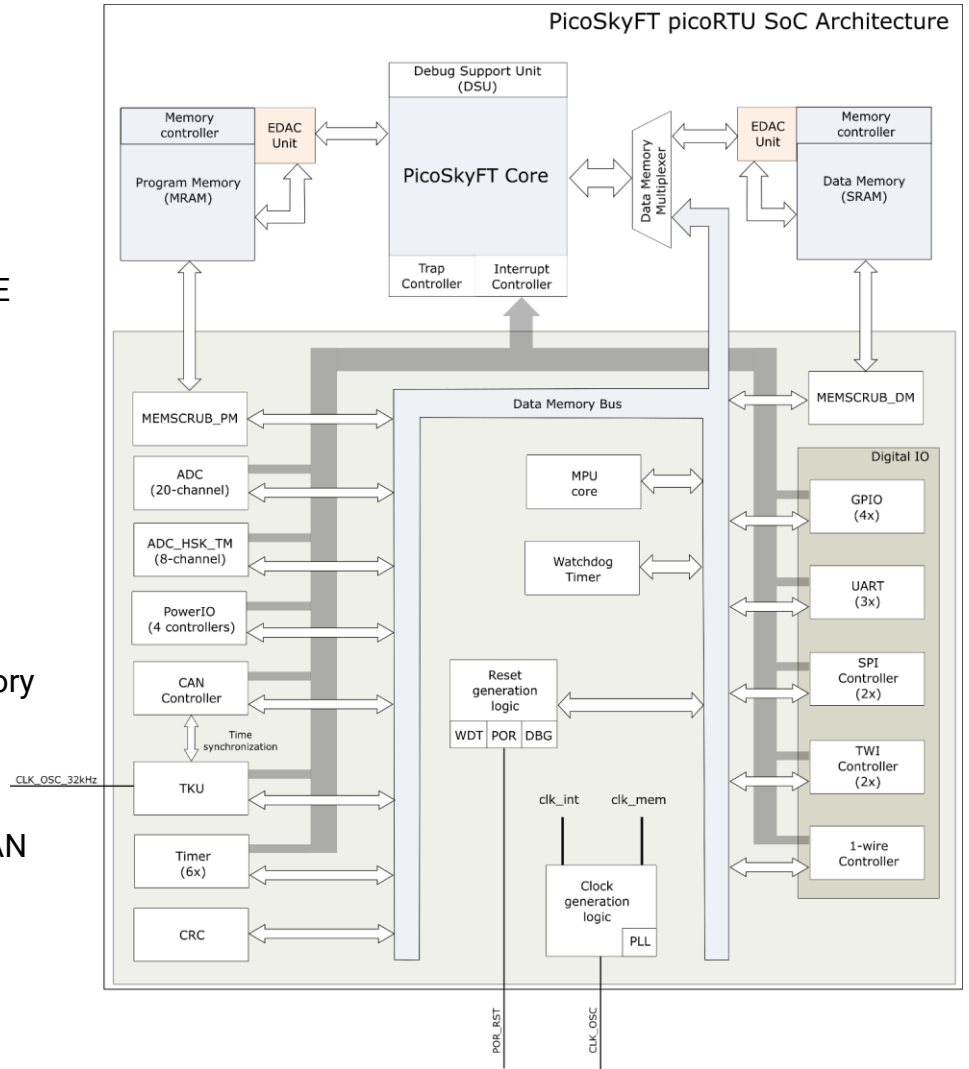
picoRTU-D RTC

skylabs

picoRTU-D

SoC Architecture

- PicoRTU SoC designed on ProASIC3E FPGA family (some test performed with NG-Medium)
- Based on SkyLabs **PicoSkyFT** processor with a set of PicoLIB peripheral IP cores
- FT features like EDAC protected memory, built-in register file parity check, memory scrubber
- PicoRTU SoC also implements Memory Protection Unit for data and program memory.
- Special hardware accelerated synchronization mechanism over CAN provides time accuracy is less then **100 ns** between units.



picoRTU-D

SoC Architecture cont'd

- DMA, NVIC, and DIT are planned to be add into SoC.
- Memory setup:
  - Internal SRAM for Data memory
  - external MRAM for Code
  - and ROM (FPGA LUTs) for emergency mode operation
- Data memory configured as 8k x 2B and 16k x 2B
- Program memory configured as 1M x 4B

SoC resources utilization (A3PE3000)	
COMB:	28916
SEQ:	8088
CORE:	37004 (49,17 %)
RAM:	96 (RAM4K9), 4 (RAM512x18) (89,29 %)
IOs:	197 (57,77 %)
PLL:	2 (33,33 %)

Performance characteristics (A3PE3000)	
clk_int:	**12,389 MHz**
clk_mem:	309,406 MHz
clk_osc:	250 MHz

Table: FPGA resource utilization and performance

skylabs

picoRTU-D

Firmware Architecture

- Objective is to have single, universal and qualified firmware (discrepancy to ECSS might be a dead code). Based on FreeRTOS, safeRTOS could be considered.
- Mission specific behaviour can be set by parameters
  - Stored in non-volatile memory
  - Some parameters are written at production (serial number, UI configuration etc.)
  - System parameters define CAN ID, baud rate, active firmware image and if unit is active
  - User interface parameters define behaviour of user interfaces
- Even if system parameters are corrupted (despite of dual image and other protection mechanisms), it is still possible to communicate with picoRTU through predefined CAN ID and baud rate.
- Self-checking mechanism

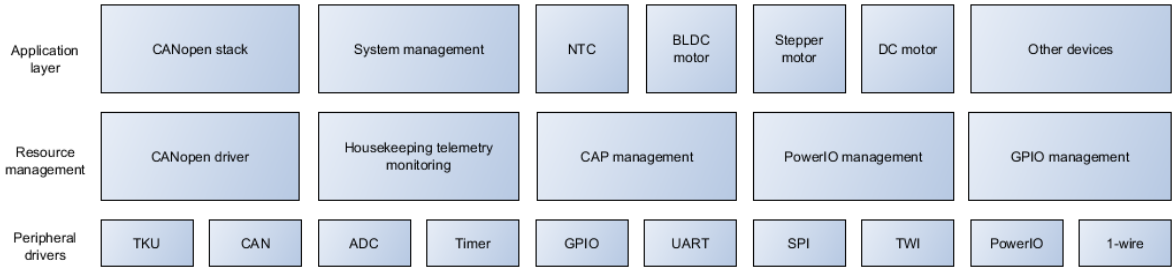


Table: Firmware high level architecture

picoRTU-D

Firmware Architecture cont'd

- Operation modes:
  - Bootloader (with emergency mode and minimal CANopen stack implementation)
  - Initialization (T)
  - Parameterization (T)
  - Halt (T)
  - Stand-by
  - Operational
- Following SAVOIR RTU operation and functional requirements

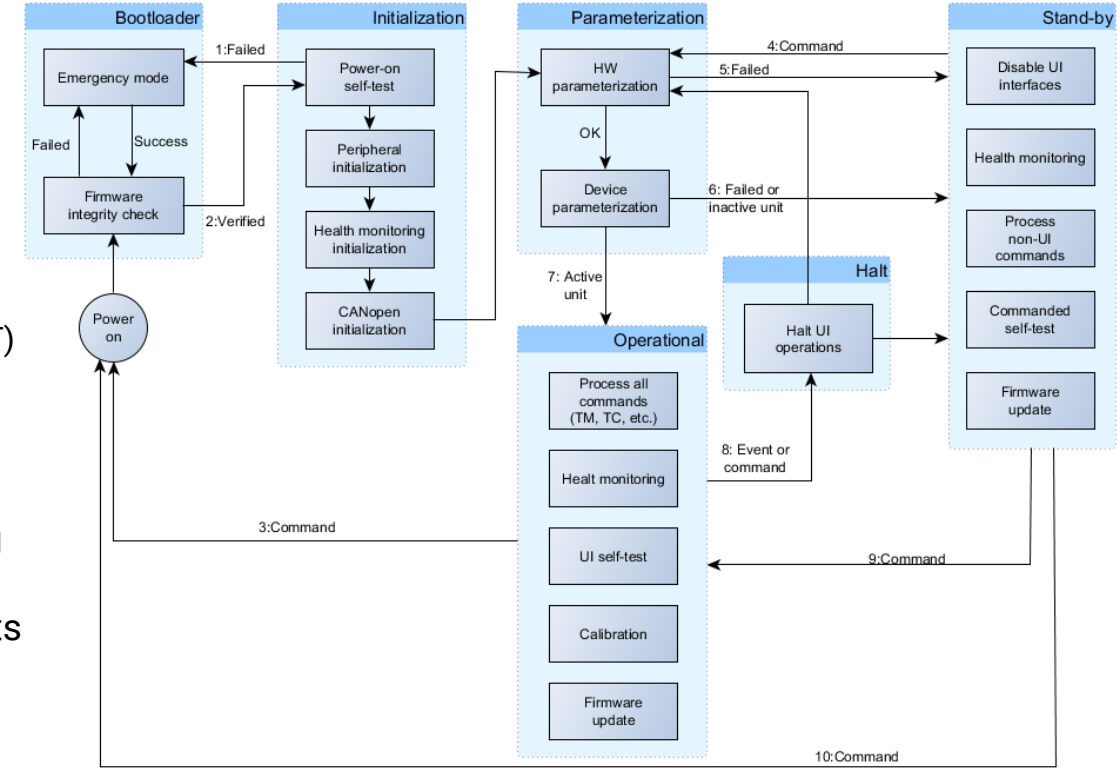


Figure: Firmware operation state diagram (SAVOIR compliant)

picoRTU System

Time synchronization

- picoRTU supports Spacecraft Elapsed Time (SCET)
  - Mandated by ECSS-E-ST-50-15C
  - Dedicated peripheral for timekeeping (TKU)
  - TC scheduler depends on this time
- Synchronization is performed by receiving special PDO with time from CANopen master
  - CAN controller detects PDO automatically and sets TKU without software interaction
  - Time accuracy between distributed units less then 100 ns.
- To improve relative time accuracy of on-board time a 1 PPS can be used.

skylabs

picoRTU-D

Firmware Architecture - Telemetry acquisition

- TM Direct reading through SDO protocol
  - System level, housekeeping, UI TM can be read on demand, cyclically logged and then read on demand, or send on change.
  - Each TM value has always a precise associated timestamp and can be can be read separately or together with TM data
  - Any TM can be read at any time
- TM can be logged in temporal picoRTU buffer
  - Depending on the picoRTU configuration, every X sample is stored
  - Read trough binary transfer mechanism
- PicoRTU-D unit logs are stored in NVM
- Telemetry lists supported (details on next slide)

skylabs

picoRTU-D

Firmware Architecture - Telemetry acquisition cont'd

- Loaded pre- and updated in- flight in non-volatile memory
- When TM list acquisition is triggered, all telemetry indicated by the list is stored in a buffer.
- Telemetry list buffer is read through SDO Block transfer service
  - Buffer integrity assured by checksum.
- Acquisition is triggered by reading special CANopen dictionary entry
  - Commanded acquisition can be done through SDO read
  - Cyclic acquisition can be setup by setting PDO to read that special entry cyclically, e.g. every N SYNC packets
    - Telemetry list buffer still has to be read by SDO

skylabs

picoRTU-D

Firmware Architecture - Housekeeping telemetry

- Exposes condensed information needed to determine overall system status and health.
- Rich local housekeeping TM is tracked:
  - Voltages on power lines (28V, 5V, 2V, 1.5V)
  - Current drawn from 28V power line
  - DC/DC, FPGA and PowerIO temperatures
  - Memory and unit statistics (Nb. of: SEC, DED, resets, etc.)
  - CAN redundancy status
  - Operating state status
  - Spacecraft elapsed time
- The Housekeeping TM can be logged in cyclic buffer
- The Housekeeping TM is locally processed and conditioned -> Master can be notified with simple OK/NOK.

skylabs

picoRTU-D

Firmware Architecture - Telecommand

- User interface and system telecommands
  - Controlling PowerIO, DigIO, Comm IF
  - Commanding advanced self-test routine
  - Commanding transition to other operational state
- Single commands can be executed by writing to special CANopen dictionary entry through SDO protocol
- TC lists supported:
  - Contain predefined list of TC and time delays between them
  - Can be loaded pre- or in- flight
  - Multiple triggers are supported:
    - Delay relative to TC list start command
    - Absolute SCET time
    - CANopen SYNC event + delay (for each TC command)
    - PPS event + delay
  - Can be periodically executed if trigger is CANopen SYNC or PPS event
  - Only for UI TCs

skylabs

picoRTU-D

Firmware Architecture – Firmware upgrade

- False safe patching
  - Two banks for two firmware images available
  - One bank is active and protected
  - Second one is inactive and can be patched
- Integrity of each bank is insured by CRC
- At power up, bootloader selects active firmware bank based on system parameters and integrity checks
  - If both firmware images are corrupted, a emergency mode is entered with minimal CANopen stack support

skylabs

picoRTU System

Novel modular fixture design

- **Novel modular fixture design** removes conventional backplane.
- Stack can be **vertically or horizontally** mounted.
- Stack limitation up **to 15 picoRTU units** (preliminary simulation based on ECSS random and sinusoidal vibration requirements)

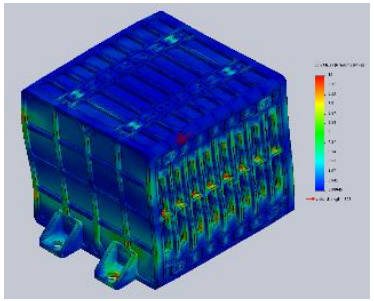


Figure: Vertical (above) and horizontal (below) fixation FEM vibration simulation

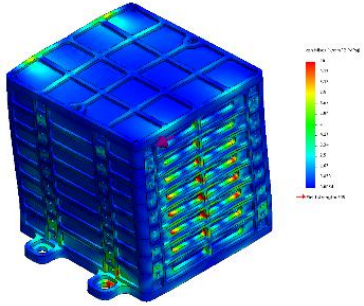


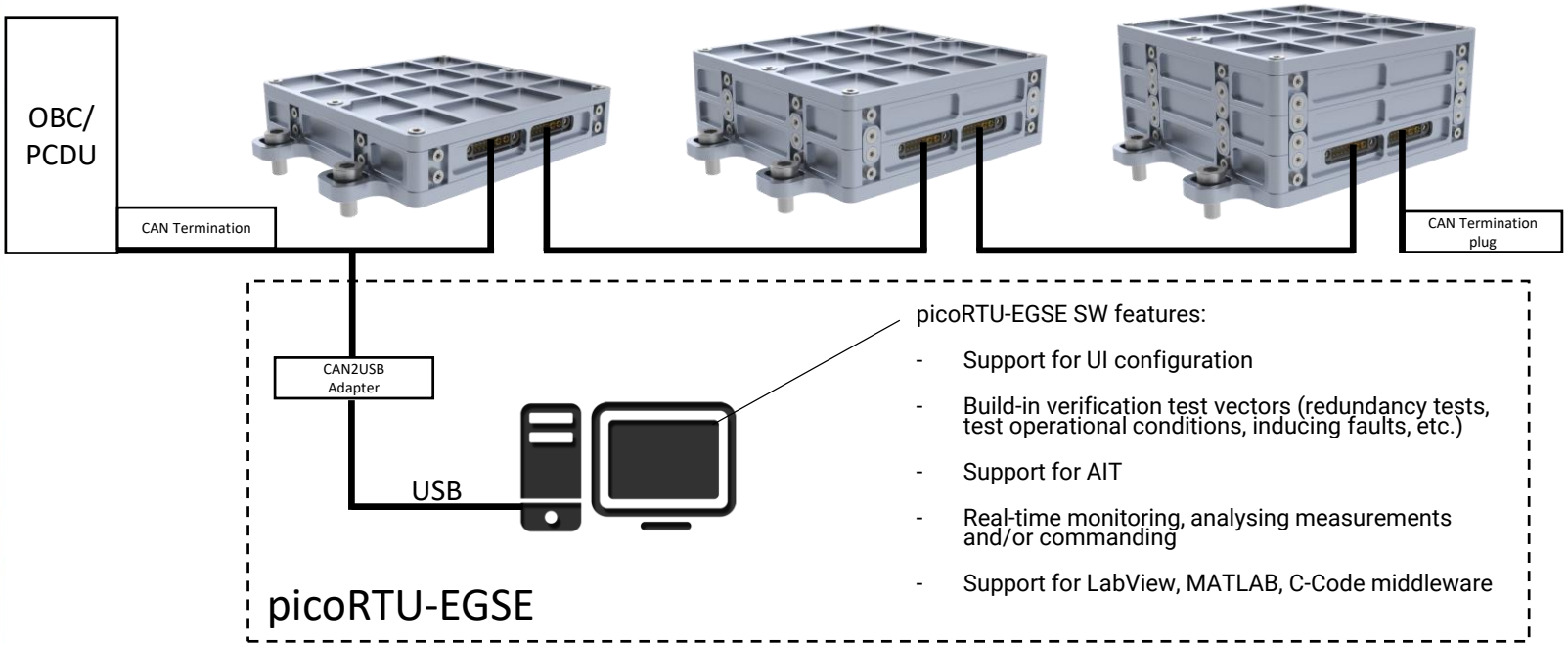
Figure: Vertical (above) and horizontal (below) fixation example

skylabs

picoRTU System

Distributed picoRTU system ecosystem

- On-board system and supporting environment enables quick assembly, configuration and validation.

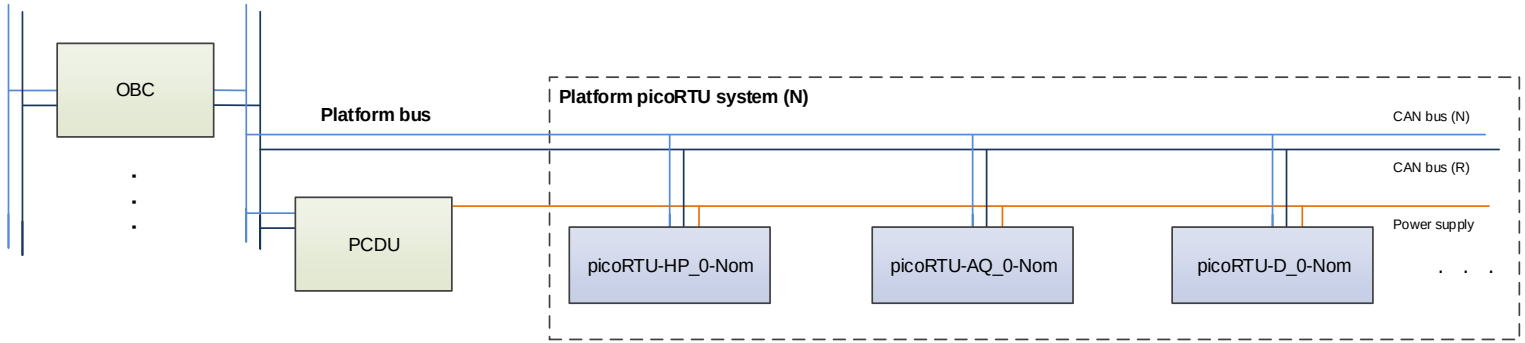


skylabs

picoRTU System

Distributed system scenario 1

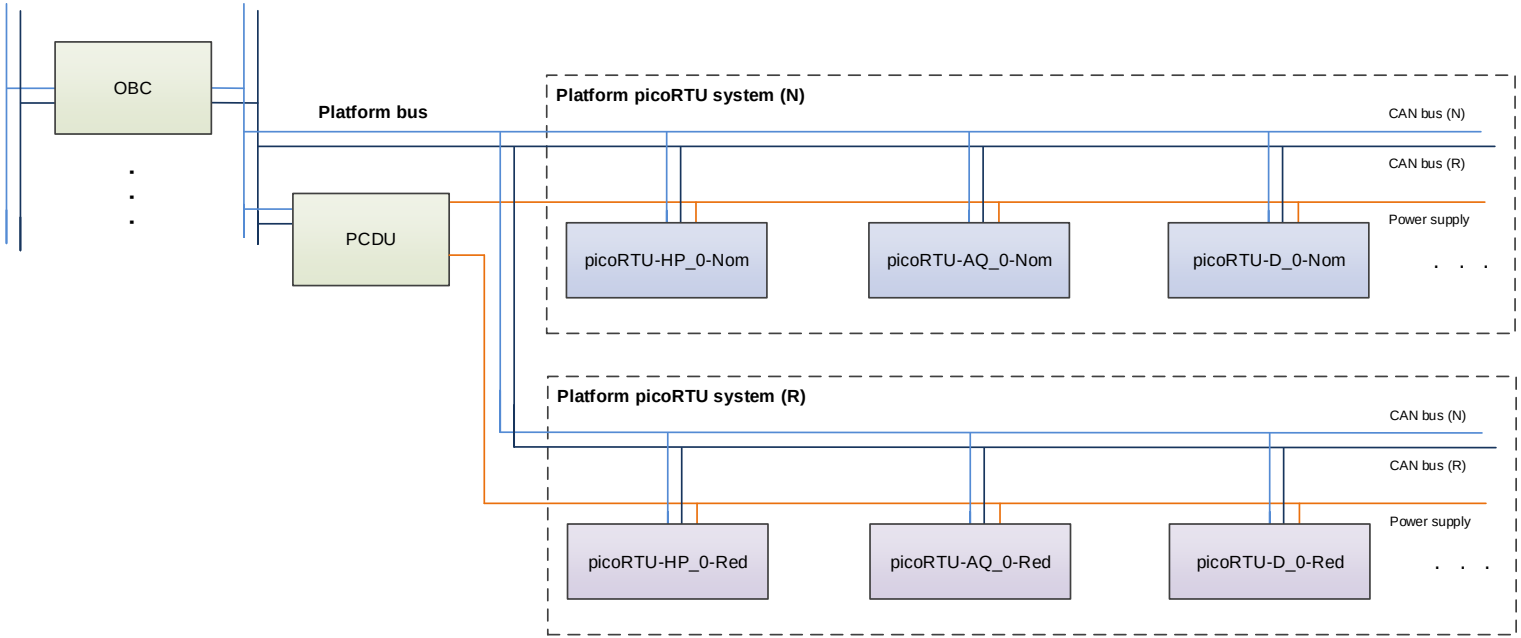
- Configuration scenario 1: Single string RTU system (Nominal)



picoRTU System

Distributed system scenario 2

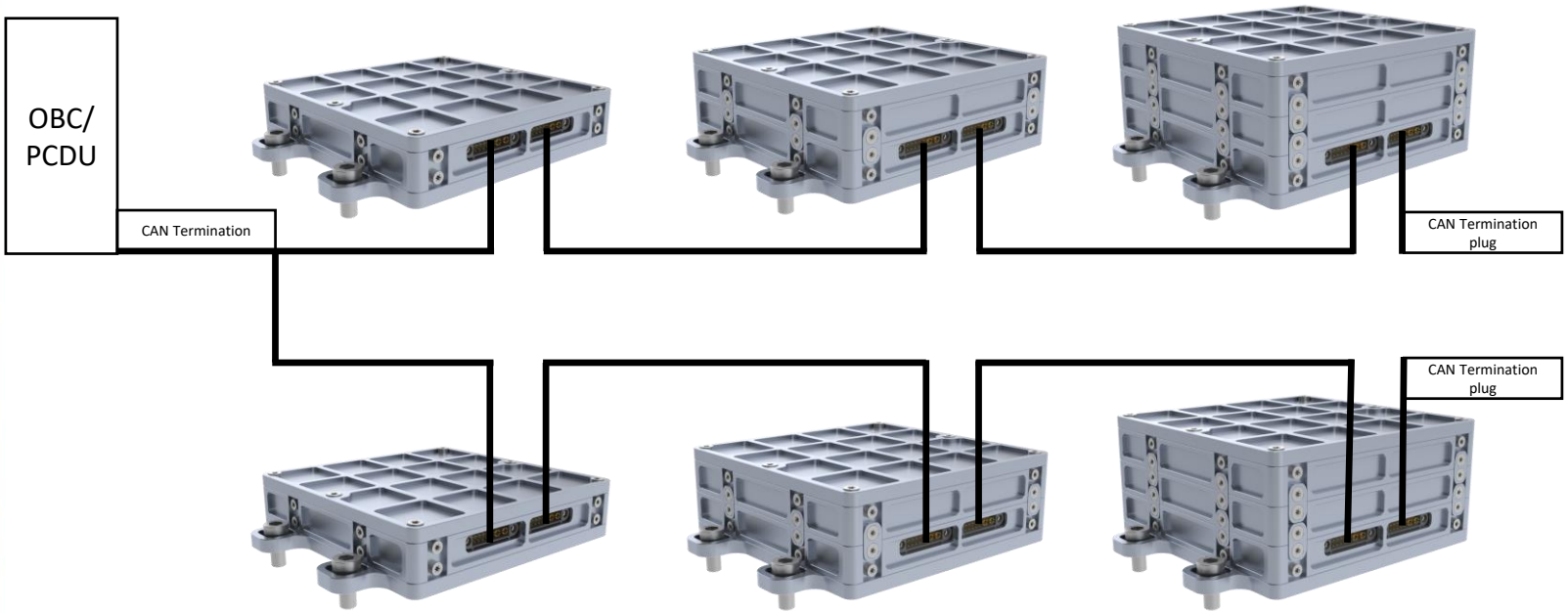
- Configuration scenario 2: Redundant RTU system (N+R) – redundant power bus



picoRTU System

Distributed system scenario 2 cont'd

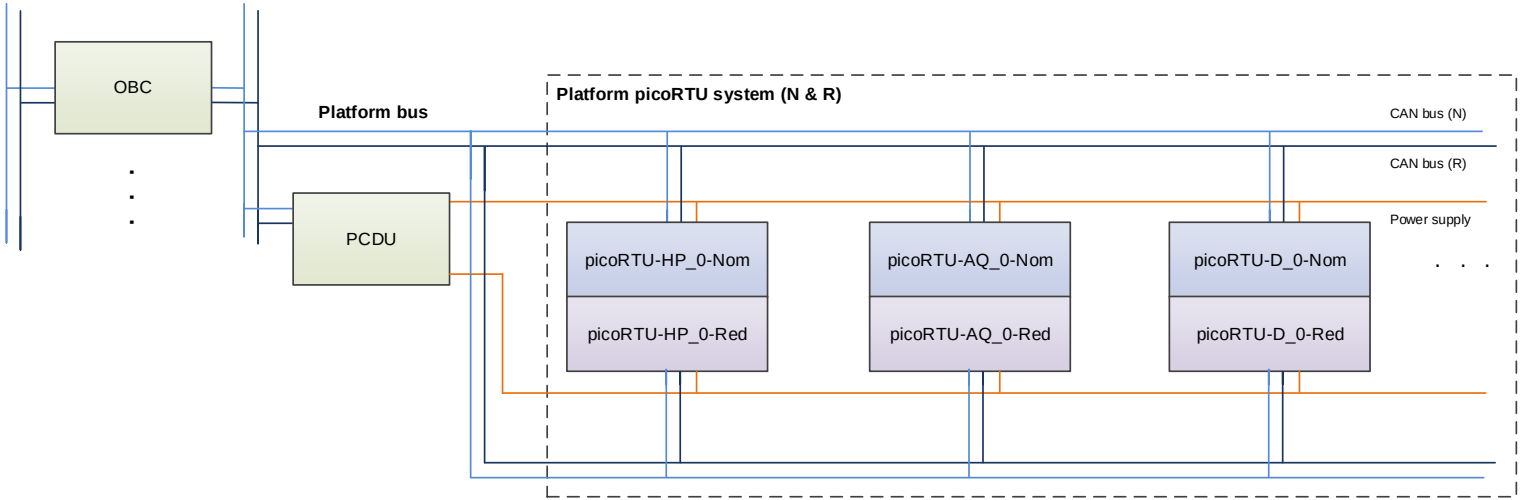
- Configuration scenario 2: Redundant RTU system (N+R) – redundant power bus



picoRTU System

Distributed system scenario 3

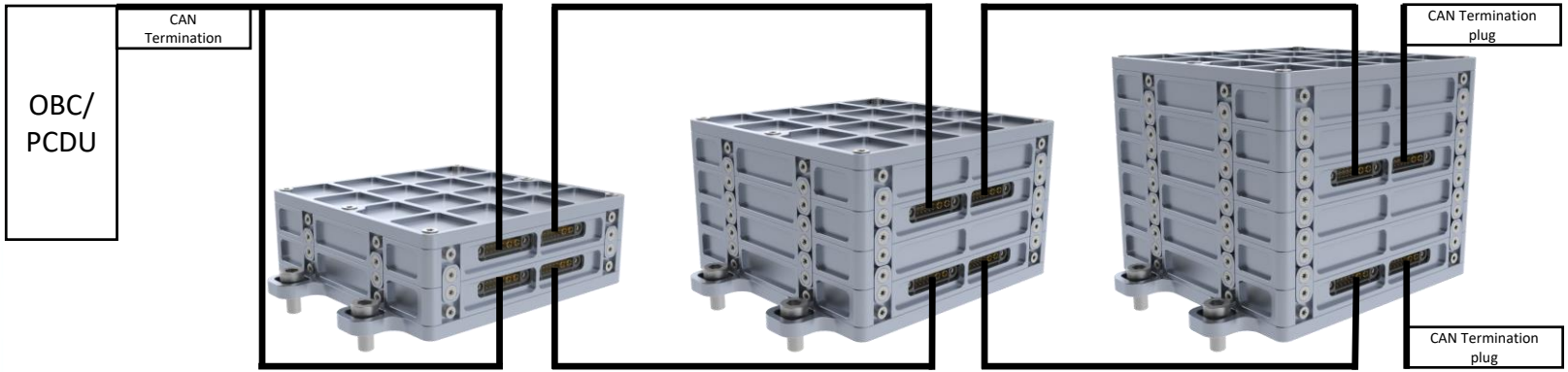
- Configuration scenario 3: Redundant RTU system in single stack (N+R) – redundant power bus



picoRTU System

Distributed system scenario 3 cont'd

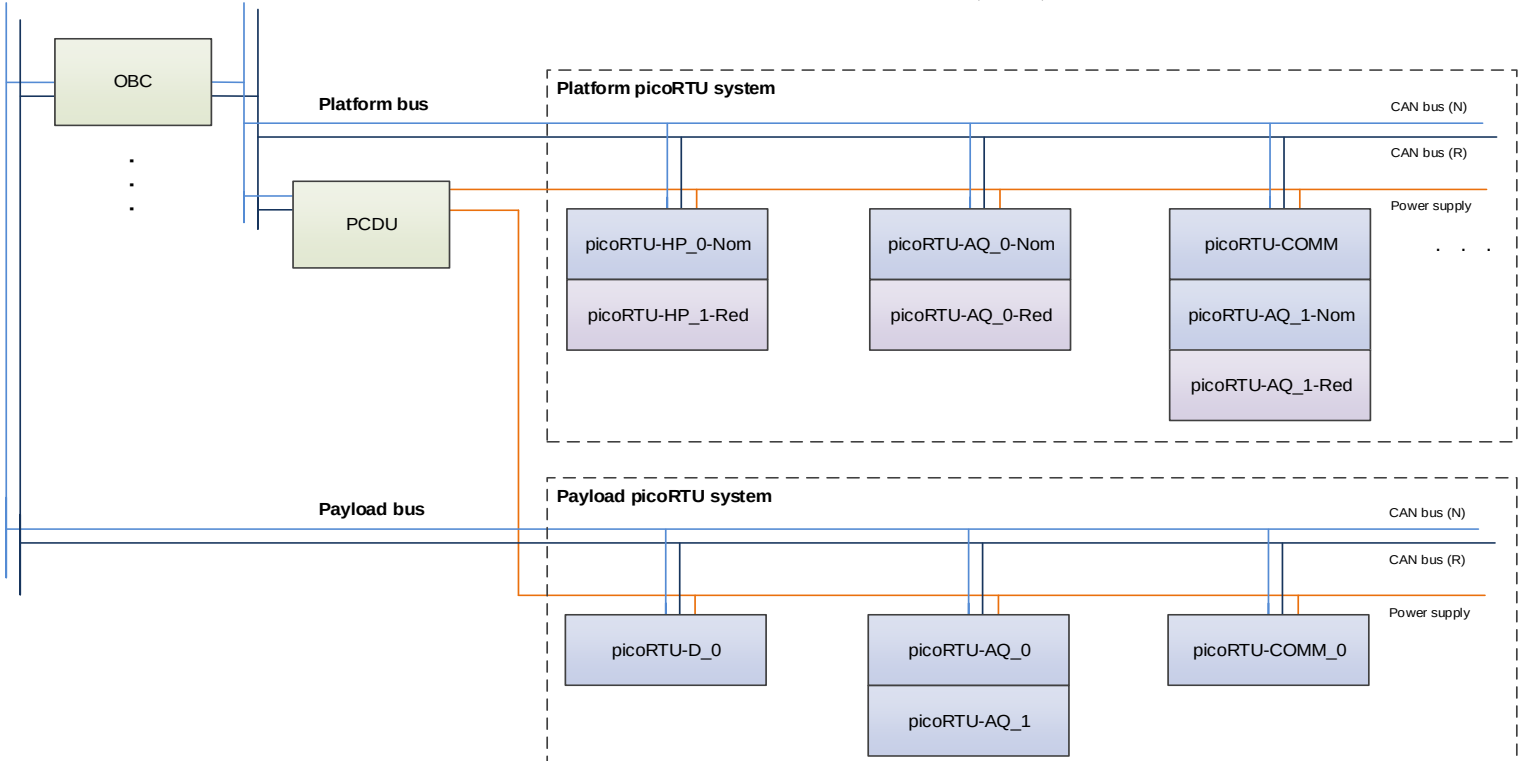
- Configuration scenario 3: Redundant RTU system in single stack (N+R) – redundant power bus



picoRTU System

Distributed system scenario 4

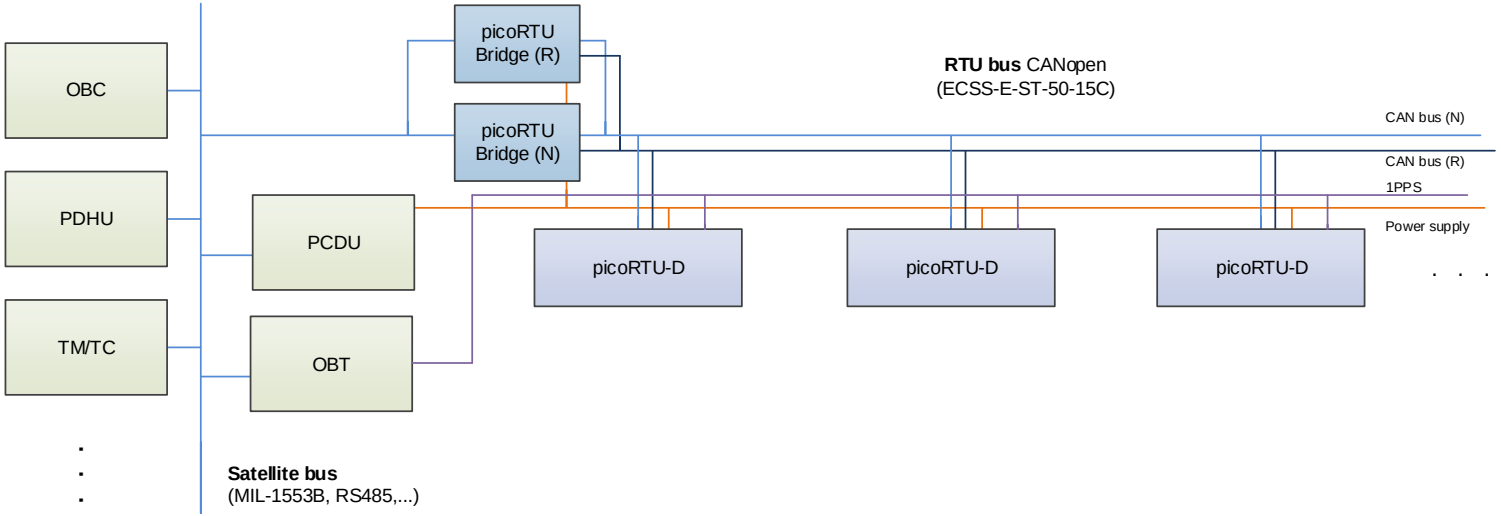
- Configuration scenario 4: Redundant RTU system in single stack (N+R) – redundant power bus



picoRTU System

Distributed system scenario 5

- Configuration scenario 5: Single string RTU system (Nominal) over PicoRTU-Bridge



picoRTU-D

Preliminary verification

- 164 Technical specifications defined (following SAVOIR RTU TS)
- 103 Preliminary testing procedures defined and successfully tested
  - functional,
  - electrical characteristics
  - environmental,

Figure 1: TVC Thermal vacuum chamber testbench

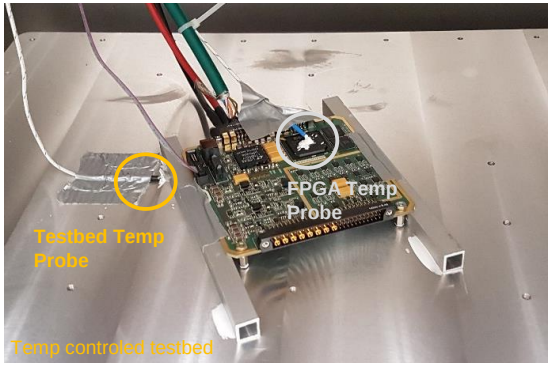


Figure 2: DUT - PicoRTU-D on thermal testbed

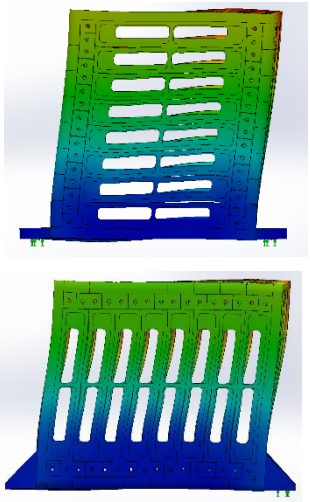


Figure 4: FEM simulation to verify structure vibration resistance compliance

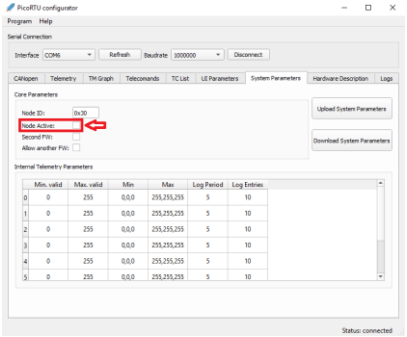
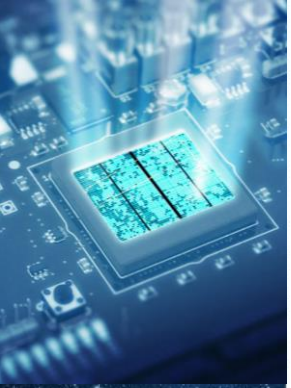


Figure 3: picoRTU-EGSE - PicoRTU-D configurator tool



Activity CCN: PicoSkyFT IP Core in NG-Medium

Budget: 31k € (ESA PECS Contract: 4000120900/17/NL-CCN1), total 231k €

Duration: 2 months

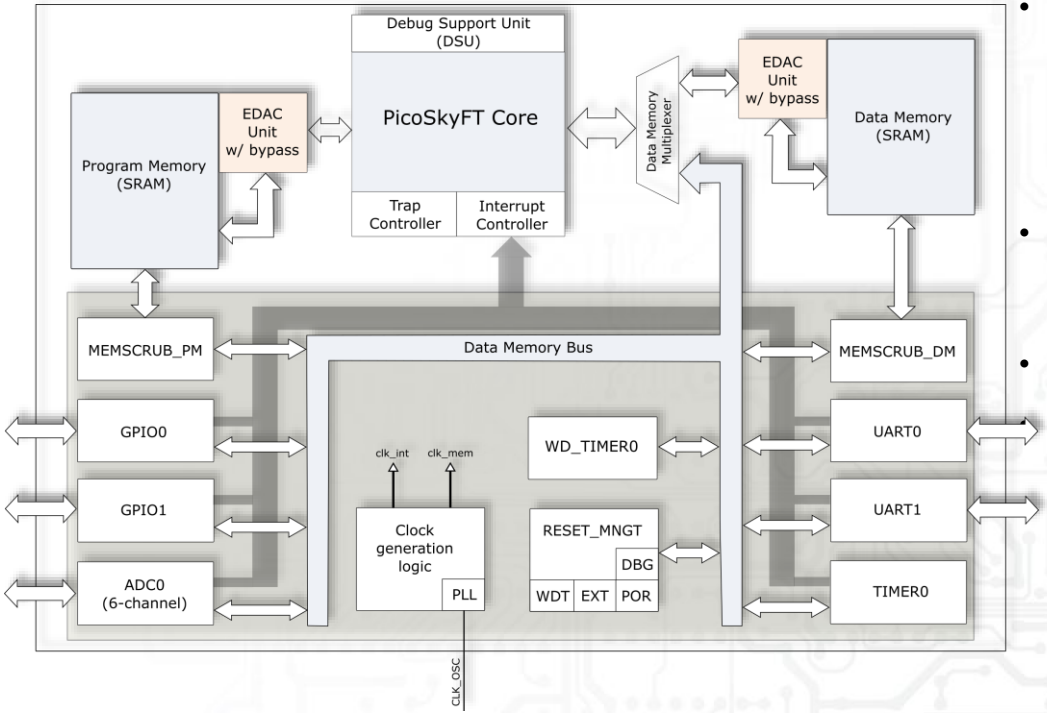
Contractor: SkyLabs (Slovenia)

Objectives:

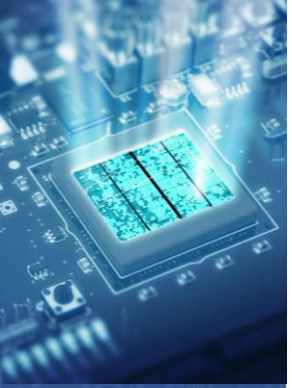
Mapping PicoSkyFT into RadHard FPGA is a major step in the PicoRTU product development roadmap. In order to foster European technological independency, SkyLabs, together with NanoXplore, pursues integration of PicoSkyFT processor core into European RadHard FPGA technology. This represents also a major step forward for Europe, as it represents fully ITAR/EAR free solution for European market, and also opens new market frontiers for European companies. The WP tasks defines mapping of PicoSkyFT into NanoXplore NG-Medium FPGA technology and enables also later characterisation with high energy proton beam.

PicoSkyFT IP Core in NG-Medium

PicoSkyFT RadTest SoC



- SoC with PicoSkyFT-L processor
- Using internal SRAM (EDAC protected):
  - DM: 8 kB
  - PM-UM code: 16 kB
  - PM-SM code: 12.5 kB
- Register file 32 8-bit registers (parity protected 8D + 2P) – not TMR'ed
- SoC frequency 10 MHz
- Single clock & reset tree



PicoSkyFT IP Core in NG-Medium

DUT PicoSkyFT SoC

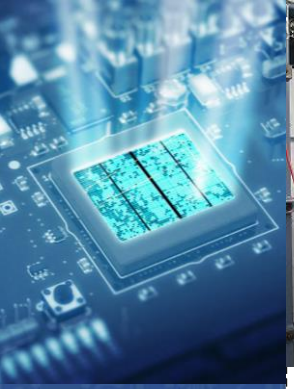
DUT-NG-Medium

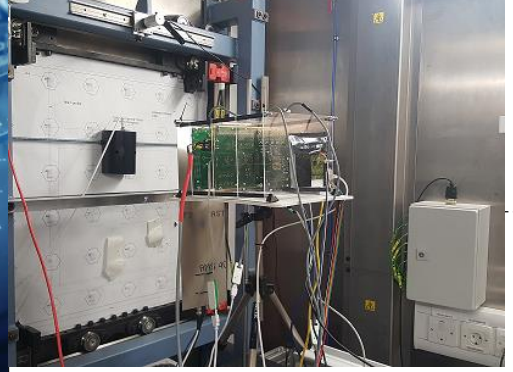
- FPGA NanoXplore NG-Medium
- Supervisor (SM) code in SRAM (EDAC protected)
- RHBD configuration:
  - no TMR
- Reliability factors:
  - SM code in SRAM
  - Both scrubbers not working
  - ADCs were not present in hardware

DUT-ProASIC3

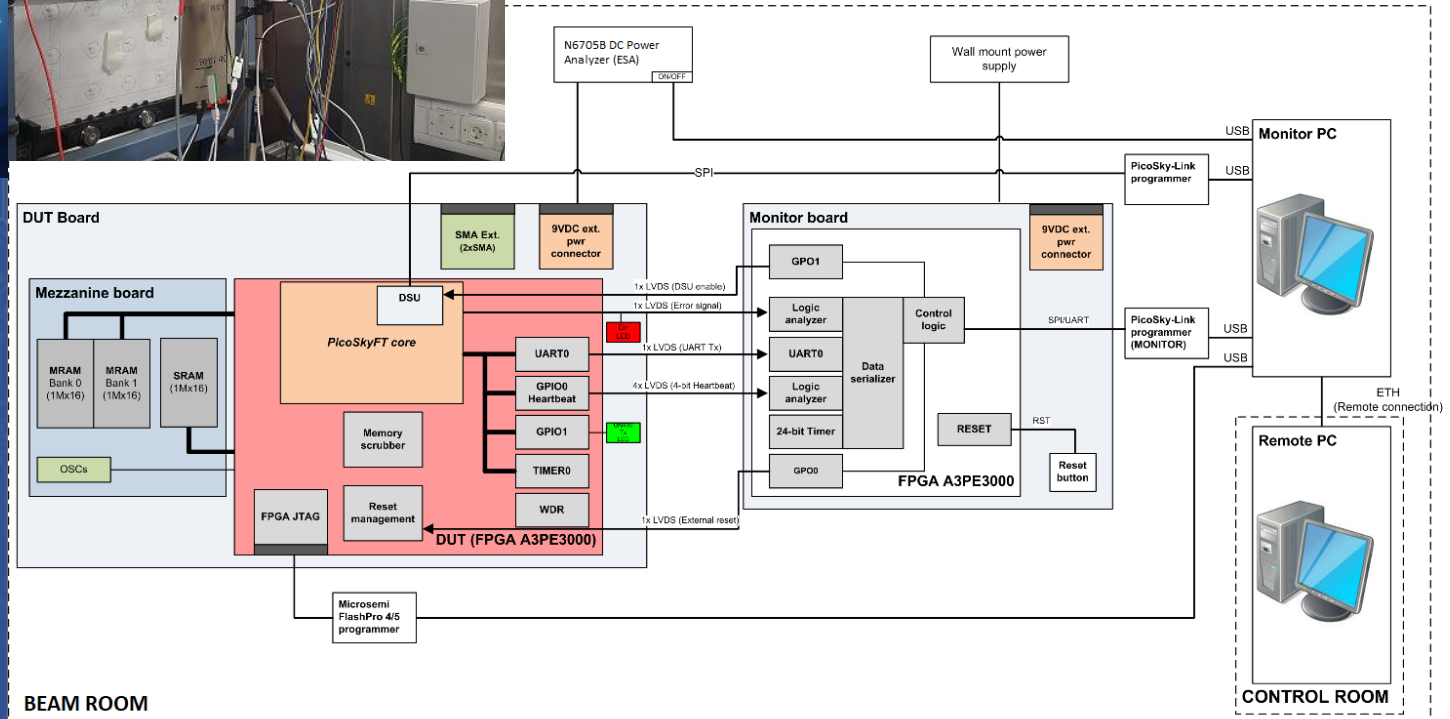
- FPGA Microsemi ProASIC3 (A3PE3000L-PQ208)
  - FPGA Core bias voltage 1.5V
- Supervisor (SM) code impl. with VersaTiles
- RHBD configurations:
  - no TMR,
  - full TMR,
  - pTMR (PicoSkyFT partial TMR, peripherals full TMR)
  - pTMR with enhanced reset logic

skylabs





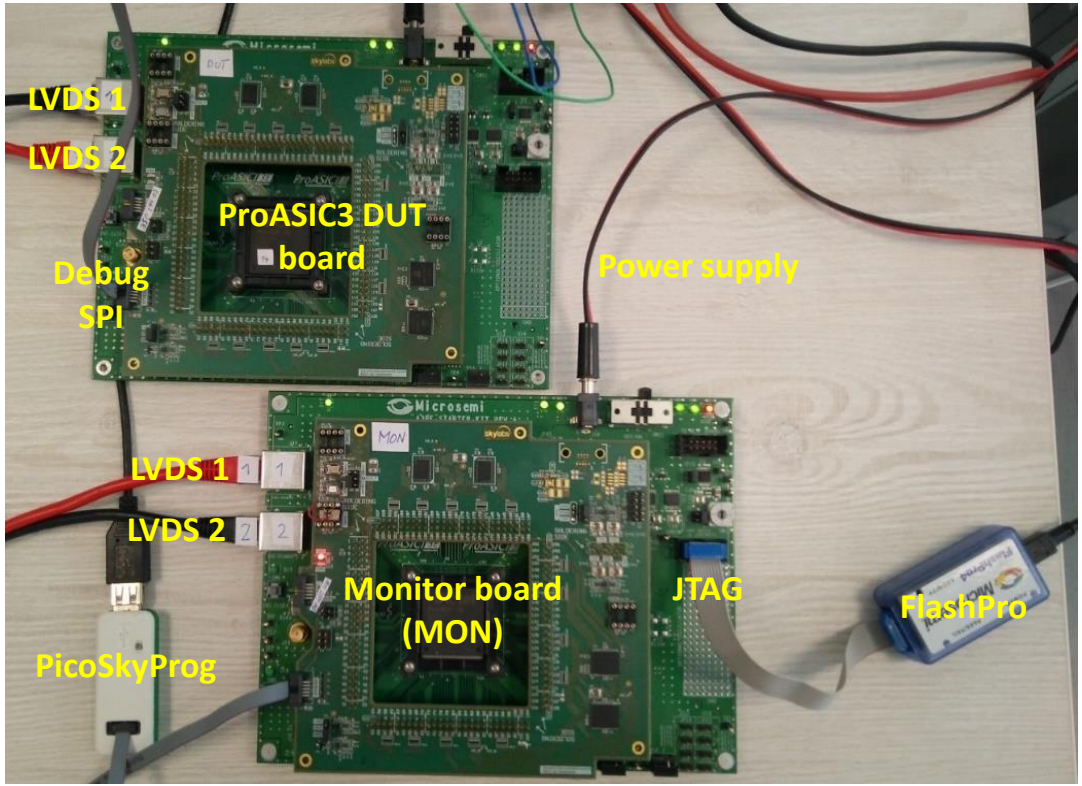
Rad. testing platform



- For ProASIC reused setup from Rad testing at PIF-PSI (09/2017) and CERN (11/2017)
- For NG-Medium EKv2 was extended with add-on board (monitor board was reused)

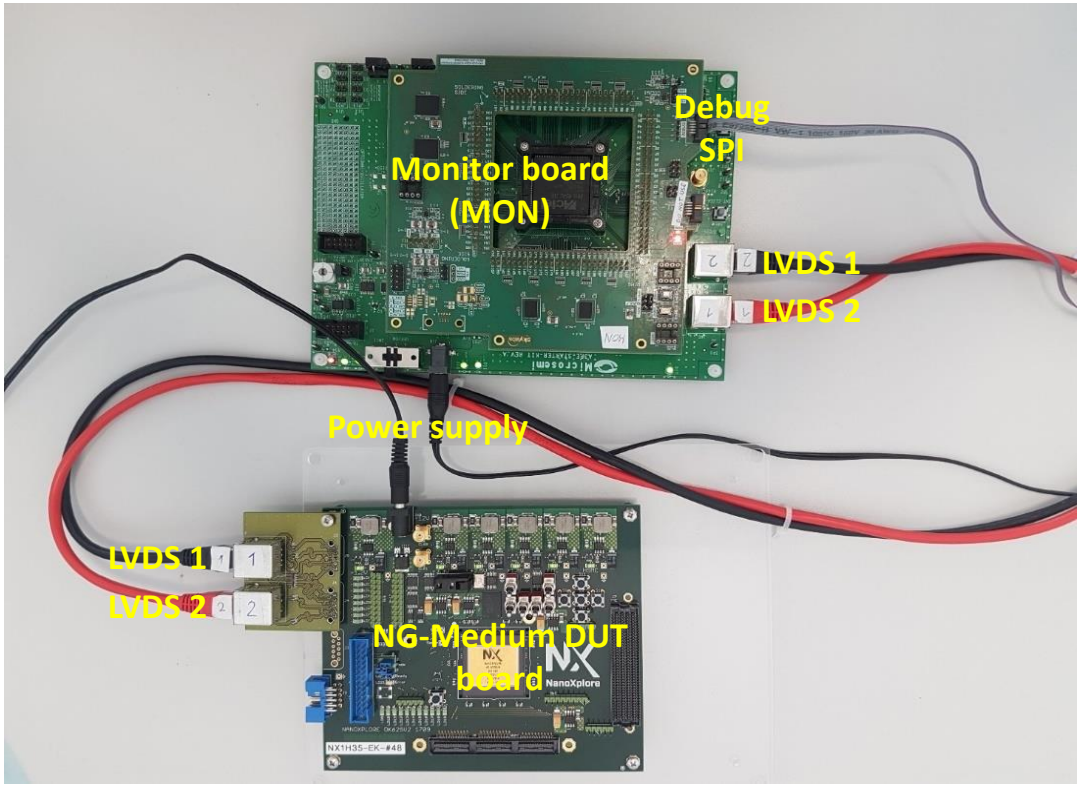
Rad. testing platform

ProASIC3 DUT



Rad. testing platform

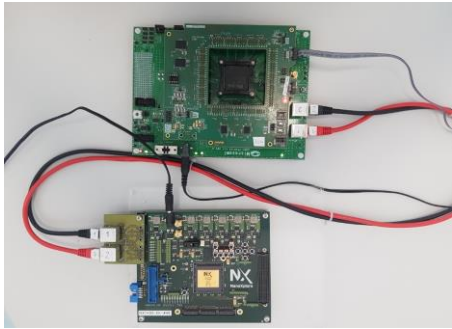
NG-Medium DUT

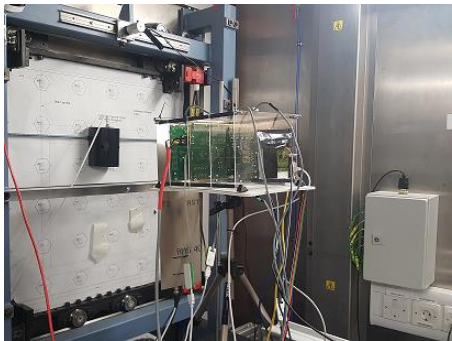


PicoSkyFT IP Core in NG-Medium

Conclusion and results

- PicoSkyFT SoC successfully mapped to NG-medium
  - PicoSkyFT-Large processor, data and program memory w/ EDAC, RegFile FT, Memory scrubber unit, GPIOs, Timer, Watchdog timer, UARTs, ADC controller...
  - Successfully map SoC design and perform NSEE testes
- Utilization and performance
  - 16kB SRAM (w/ EDAC), 1 PLL, CMIC(5.3ms), 26% 4-LUTs (8%DFF)
  - Max freq. 13.44MHz (actual SoC freq. is 10MHz)
  - Environment: Ubuntu 16.04 LTS, NanoXmap 2.8.5, NxBase2 2.1.1
- NSEE test summery @ ChipIR ISIS by Uni MB
  - DUT 24h under the beam / fluence 4,6E+11 neutrons/cm2
  - Proven 100% Safe operation of processor and FIT rate of 0.18
  - Observed events: SEU in embedded SRAM, SET on LVDS line (width of 50ns), SEU in DFF or SETs in combinatorial logic (can not be determine), LVDS IO driver loss (most likely CMIC reset the FPGA, due to SEU in configuration cell).
- Further work required to improve PicoSkyFT radiation testbench for NG-Medium, thus better classification of SEE values and xSections can be provided.





skylabs

Conclusion

- PicoRTU-D represents a good example of advantages using distributed systems, thanks to available soft-cores processor and its complete ecosystem.
- A novel fixation design is presented and shown to be robust to support up to 15 units within single picoRTU stack.
- The modular system based on off-the-shelf qualified (HW and SW) generic units has important advantages:
  - Reduces S/C development times, as only parametrization of picoRTU system firmware needs to be done, and picoRTU will already fulfil requirements of discrete interfaces for acquisition and actuation.
  - Eases and accelerates the verification during AIT, thanks to picoRTU-EGSE supporting software.
  - Modular system efficiently copes with late requirement changes, as it is a matter of firmware reconfiguration with changing or adding new units.

skylabs

Follow up

Follow up activity:

- Required support on system level requirements to defined (picoRTU-EGSE, Bridge,..)
- Detailed trade-off in **Interfaces vs size.** Anticipated additional support from ESA and any other interested primes) in order to provide the best fit of picoRTU systems for various mission sized and interfaces requirements.
- Definition of specifications for two different picoRTU radiation resistance hardware (***RH for GEO*** vs ***COTS+ for LEO***), but considering fully equivalent functionally between both units types.
- Further work required to improve PicoSkyFT radiation testbench for NG-Medium, thus enabling better classification of gathered SEE results and applying proper mitigation techniques for RH picoRTU units.
- Revised power management (PM) within a single stack. Fully distributed (each unit makes its own power rails) vs. base stack unit with PM for all stacked units)

skylabs

Thank you

SkyLabs d.o.o.

Poljska ulica 6

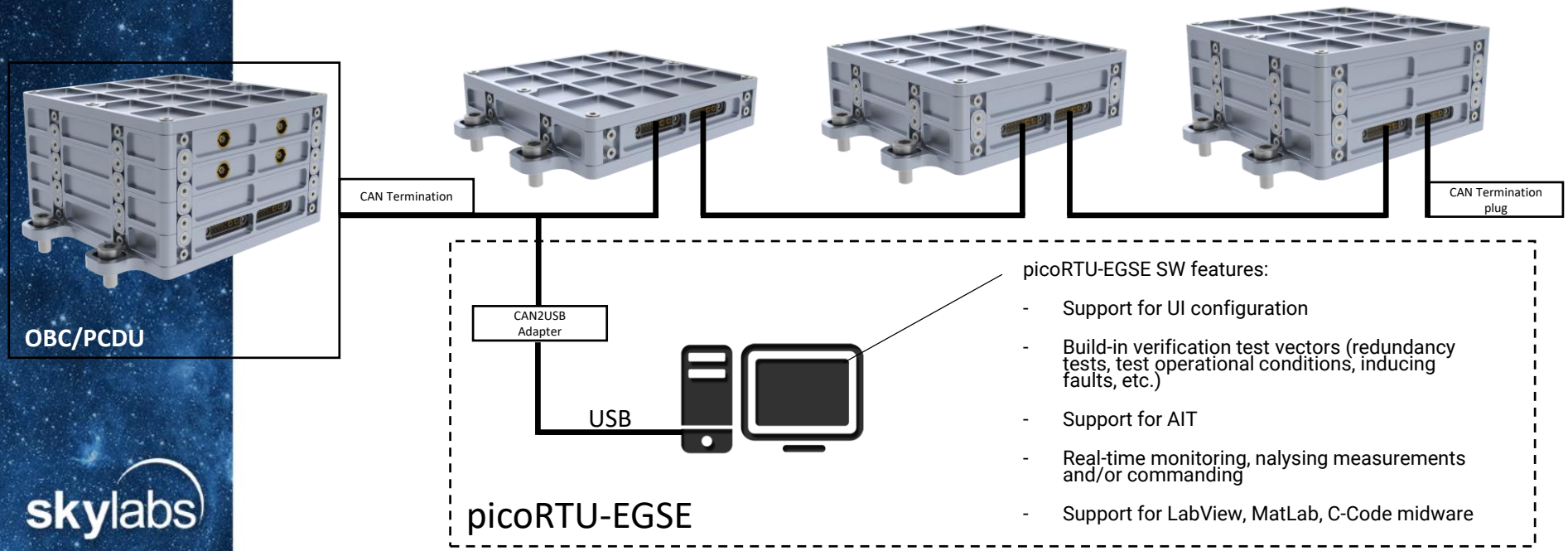
SI-2000 Maribor

info@skylabs.si

picoRTU System

Distributed picoRTU system ecosystem

- On-board system and supporting environment enables quick assembly, configuration and validation.



skylabs