COBHAM

The most important thing we build is trust

© Ian Roman/Volvo Ocean Race

ADVANCED ELECTRONIC SOLUTIONS | AVIATION SERVICES | COMMUNICATIONS AND CONNECTIVITY | MISSION SYSTEMS

A common basic SW and DPU platform for the JUICE mission Instruments

esa

Date: 2016-12-15 (Flight Software Workshop 2016)

Presenters: Daniel Hellström
Cobham Gaisler AB

Presentation does not contain ITAR information

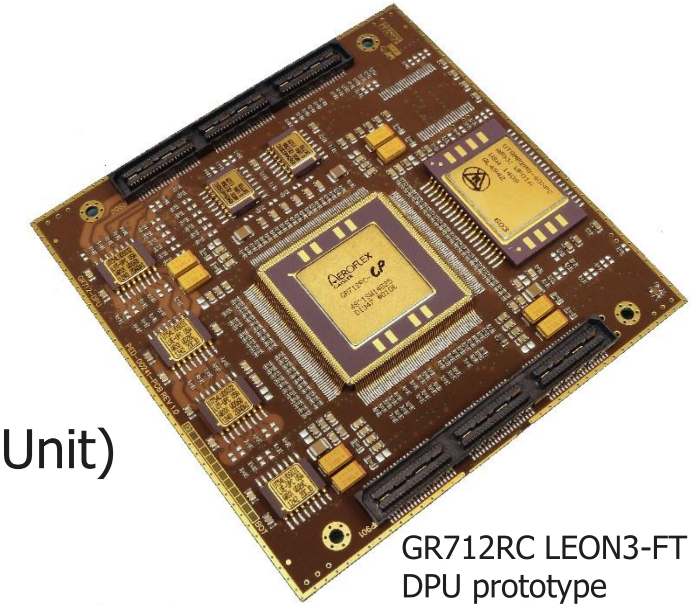
Introduction

- JUICE – JUpiter ICy moons Explorer
  - European Space Agency mission
  - 10 scientific instruments identified potential users of a common platform

- ESA common HW/SW requirements

- Cobham Gaisler has developed a common HW/SW platform for the instruments based on LEON3-FT GR712RC.

- Modular LEON3-FT DPU (Digital Processing Unit)
- Boot SW with SpW/PUS standby
- GR712RC Driver SW for bare-metal and RTEMS



GR712RC LEON3-FT DPU prototype

DPU for JUICE Instruments

Presentation Overview

- ESA JUICE mission
- Project overview
- DPU background and results
- Boot SW
- Driver SW
- Tools used for SW development
- Conclusion & future

ESA JUICE Mission

- JUICE – JUpiter ICy moons Explorer
  - European Space Agency L-class mission
  - Spacecraft destined for the Jupiter system
  - Launch in 2022, cruise until 2030, 3½ years in the Jupiter system.
  - 10 scientific instruments identified potential users of a common platform

- Redundant 100Mbps SpaceWire network
- Instruments redundancy, SpW config:
  - Cold redundant (one SpW per DPUx2)
  - Single-string (two SpW per DPUx1)
- PUS protocol over SpW
- Optional for Instrument Providers to adapt to common HW/SW

DPU for JUICE Instruments

ESA project overview

ESA Activity: DPU for JUICE Instruments

DPU processor section

- GR712RC – LEON3FT
- Boot memory
- Application storage
- Working memory
- FPGA I/O interface
- SpaceWire
- Clocking

Software

- Boot & standby SW
- HW drivers SW
- TSIM2 simulator DPU adaptations

Prototype system

- MB+DPU 4-32MiB SRAM configuration
- MB+DPU 256MiB SDRAM configuration

Project schedule

- Start: February 2015
- PDR: June 2015
- CDR: March 2016
- Delta-CDR: June 2016 (align to new reqs)
- PRE-AR: December 2016 (new services)
- 3:rd party ISVV: H1 2017
- DPU-SIS testing: H1 2017
- AR: Q3 2017 (3:rd party validation input)

Final HW delivered during 2016.

First & second version of SW delivered during 2016.

SW updates are on-going with additional services 2016 & 3:rd party validation 2017.

DPU for JUICE Instruments

Project requirements

- ESA collected common requirements from JUICE Instruments
- ESA compiled HW and SW requirements
    - Functional
    - Performance
    - Quality

Hardware requirements

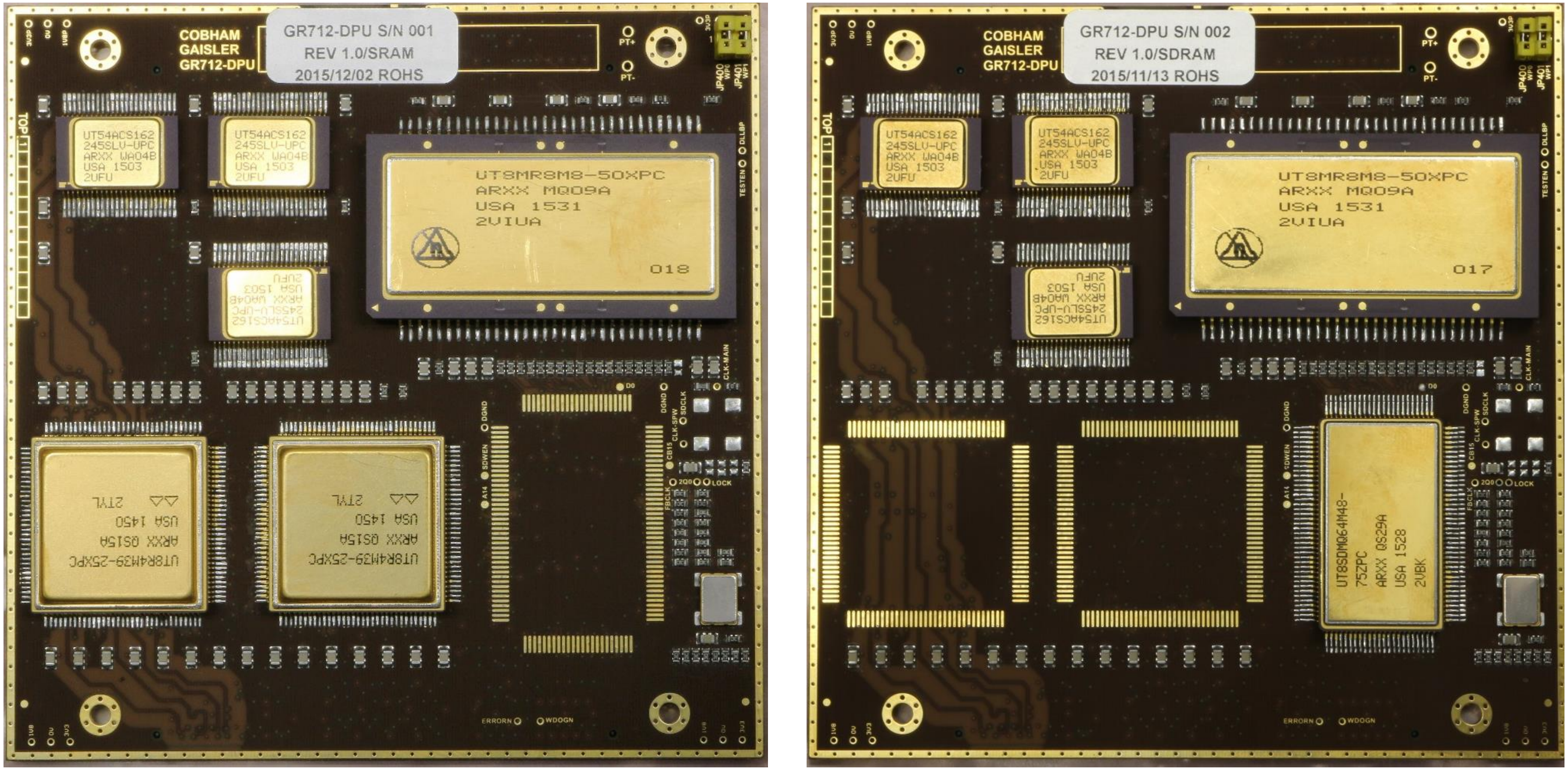
- CPU performance (MIPS, MHz)
- Memories
    - Boot ROM Memory (fit Boot SW, read-only)
    - Application Storage Memory (up to 8 MiB, non-volatile)
    - Main Memory (4 – 256 MiB, volatile)
- Companion chip interface (FPGA)
    - Memory mapped I/O, Interrupts, SpaceWire
- SpaceWire cross-strapping
    - Single string
    - Dual redundant DPU configuration
- Component radiation tolerances:
    - 100 krad (Si) total dose
- EDAC protection of memories

Software requirements

- SW developed according to ESA software engineering standards. Tailoring of the ECSS:
    - ECSS-E-ST-40C “Space Engineering – Software”
    - ECSS-Q-ST-80C “Space Product Assurance – Software Product Assurance”
- Boot SW compliant with tailored of ESA requirements TEC-SWS/10-373/FT "Flight Computer Initialisation Sequence‟ as Payload computer.
    - Init, Self-tests, Application loading, Standby
    - SpW and PUS services,
- RTEMS Low-level Driver SW for DPU interfaces
- Mission specific requirements
- SW shall support all DPU configurations

DPU hardware

SRAM configuration (left) and SDRAM configuration (right)

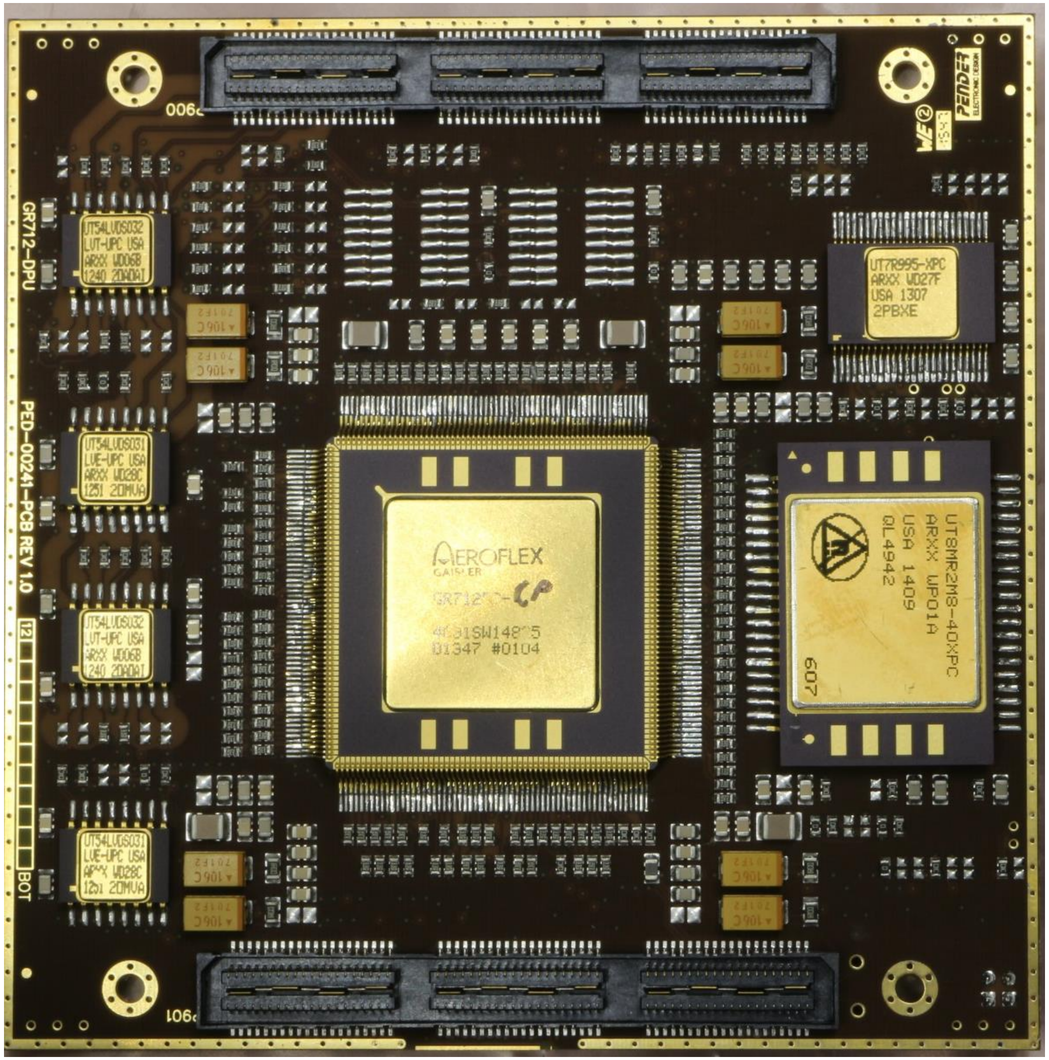

COBHAM GAISLER GR712-DPU
GR712-DPU S/N 001
REV 1.0/SRAM
2015/12/02 ROHS
UT54ACS162 245SLV-UPC ARXX WA04B USA 1503 2UFU
UT8MR8M8-50XPC ARXX MQ09A USA 1531 2VIUA
018
UT8R4M39-25XPC ARXX QS15A USA 1450 2TYL
COBHAM GAISLER GR712-DPU
GR712-DPU S/N 002
REV 1.0/SDRAM
2015/11/13 ROHS
UT54ACS162 245SLV-UPC ARXX WA04B USA 1503 2UFU
UT8MR8M8-50XPC ARXX MQ09A USA 1531 2VIUA
017
UT8SDMQ64M48-752PC ARXX QS29A USA 1528 2VBK
TOP
ERRORN
WDOGN

100mm x 100mm, two-sided mounting

12 layers

DPU hardware

Bottom side – GR712RC CPU


GR712-DPU
PED-00241-PCB REV 1.0
BOT
UT54LVDS032 LVT-UPC USA ARXX WD06B 1240 20ADA1
UT54LVDS031 LVE-UPC USA ARXX WD28C 1251 20MVA
UT7R995-XPC ARXX WD27F USA 1307 2PBXE
AEROFLEX GAISLER
UT8MR2M8-40XPC ARXX WP01A USA 1409 QL4942
PENDER ELECTRONIC DESIGN

2x120pins high-speed connectors

3.3V

1.8V (SRAM + GR712RC iocore)

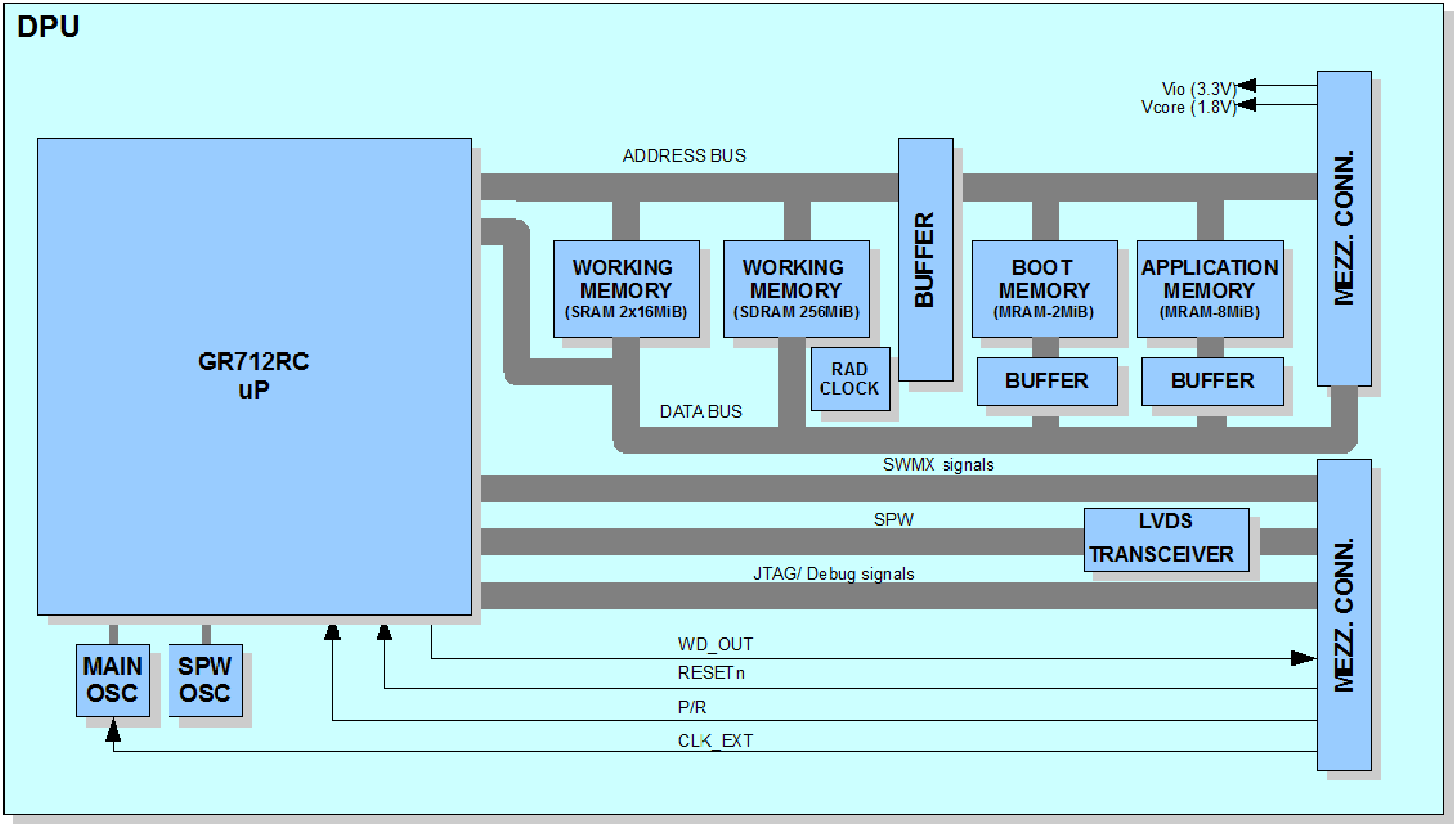
Memory options configurable at assembly

100mm x 100mm

12 layers

DPU architecture

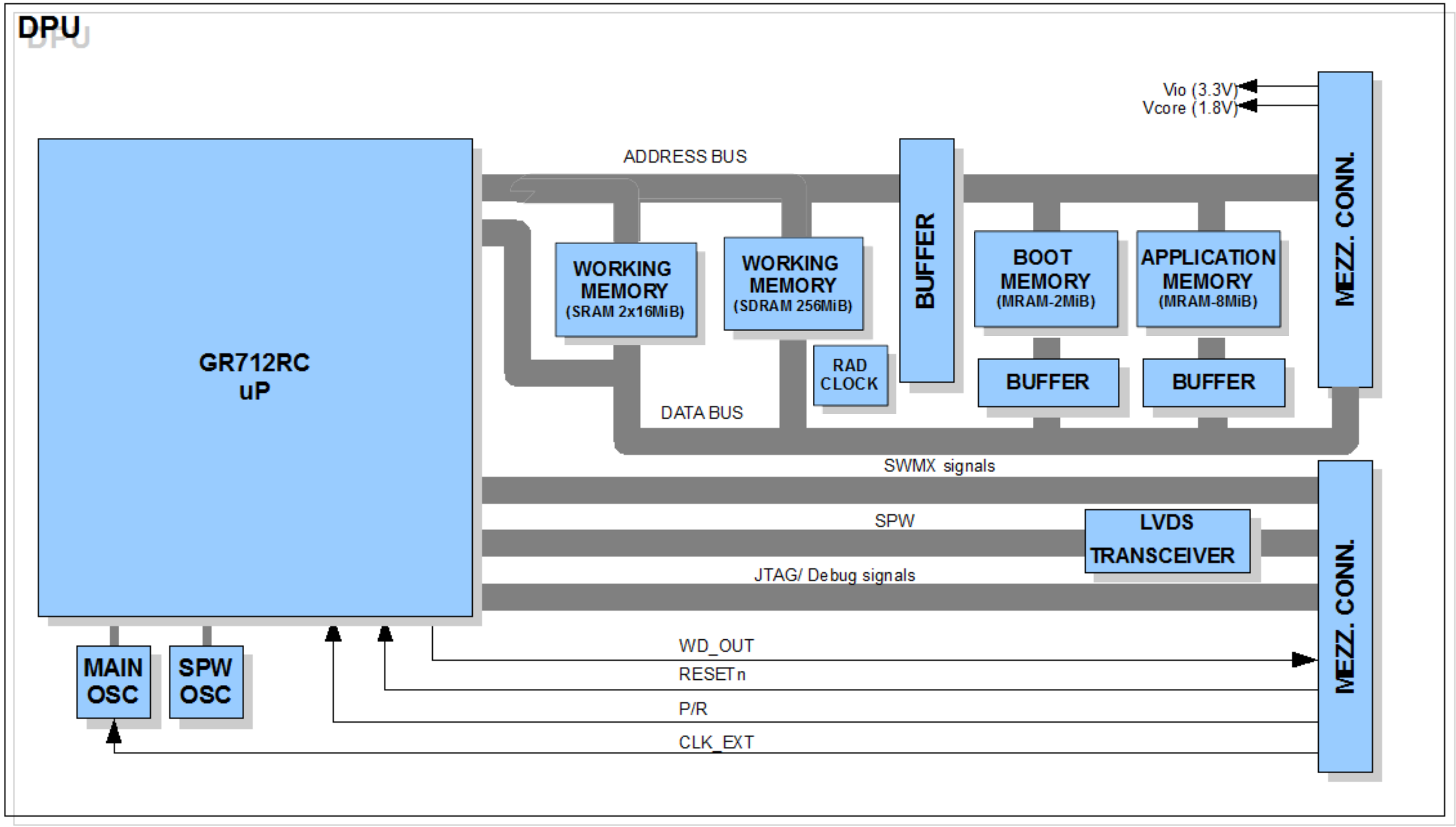
Block diagram for schematics, with two options for working memory



DPU architecture changes

Memory configuration options

PROTOTYPE BOARD



Boot:
MRAM 2 MiB

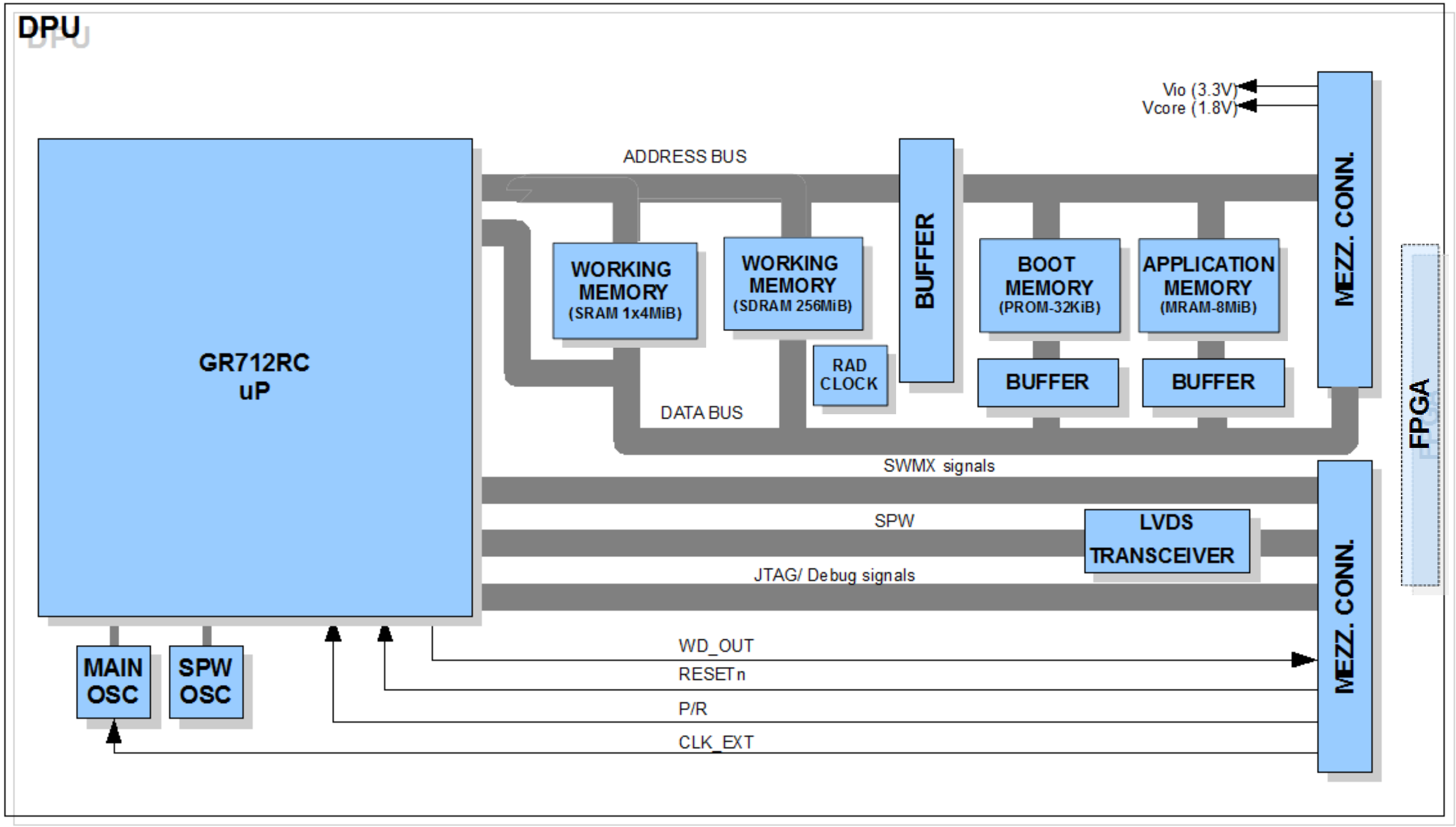
Application:
MRAM 8 MiB

Main memory:
SRAM 2x16 MiB
SDRAM 256 MiB

DPU architecture changes

Memory configuration options

FLIGHT MODEL EXAMPLES



Boot:
PROM 32 KiB

Application:
MRAM 8 MiB

Working:
SRAM 4 MiB
SDRAM 256 MiB

DPU architecture - results

Main memory and Boot memory

Main memory

SRAM interface

- Aeroflex UT8R4M39,UT8R2M39,UT8R1M39
- Total Dose:100 krad(Si),
  SEL Immune: <110 MeV-cm2/mg,
  SEU error rate = 7.3x10-7 errors/bit-day
- **4 - 32MiB** - 32-bit data, 7-bit BCH EDAC

SDRAM interface

- Aeroflex UT8SDMQ64M48 3.0-Gigabit SDRAM
- Total dose: 100 krad(Si),
  SEL Immune 111 MeV-cm2/mg,
  SEU Event Rate: 1.3E-10 events/bit-day
- 64Mx48-bit, allows 16-bit Reed-Solomon EDAC
- Single configuration: **256MiB** effective size
- Aeroflex UT7R995 RadClock used to skew SDRAM clock to meet timing according to WCA results.

PROM Interface

Boot and Application memory both on the PROM memory interface.

GR712RC PROM EDAC supported by design on both PROM and MRAM. CG radiation report **recommends to turn EDAC off** within JUICE mission

Boot PROM

- UT28F256LVQLE - **32KiB** PROM
- Total dose: 1Mrad (Si),
  Onset LET: 40 MeV-cm2/mg,
  SEL Immune > 110 MeV-cm2/m
- Boot SW requires less than 20KiB.

DPU architecture - results

Application storage memory, SpaceWire and FPGA interfaces

SpaceWire

- Aeroflex UT54LVDS031LV/E
- Redundant SpW transceivers for SpW0/1
- Single GPIO power-down transceiver
- Configuration options:
    - Single-string (two interfaces)
    - Dual-redundant (one interface)
- DPU has 50MHz SpW clock. GR712RC PLL (x1,x2,x4 input) & SpW clk divider allows 200/**100**/66.7/50/**40**/33/25/20/10/..MHz.

FPGA I/O companion interface

- SpaceWire
- 8/16/32-bit I/O interface (isolated with buffers) + GPIO for interrupt (optional)

Application Storage MRAM

- Two configurations:
    - 2MiB, Aeroflex UT8MR2M8
    - 8MiB, Aeroflex UT8MR8M8
- Total dose: 1Mrad (Si),
  SEL Immune: 112 MeV-cm2/mg @125C
  SEU Immune: Memory Cell 112 MeV-cm2/mg @25C
- MRAMs has Internal EDAC, corrects single-bit and detects double-bit errors.
- 8MiB configuration has MBE pin connected to GPIO – SW can detect double-bit errors
- ZZ/RST connected to GPIO to enter sleep mode and isolate from radiation.
- MRAM technology benefits:
    - Easy to read and write, fast accesses.
    - Memory cells themselves are not sensitive to SSE, transient effects during read-cycle are guarded by EDAC and easily recovered from toggling ZZ/RST without need for reboot.
    - No scrubbing needed. No reprogramming since >20 years data retention.

DPU performance – simulation results

Results based on Hyperlynx simulations

- Hyperlynx simulations performed for critical cases from datasheet and load analysis.
- Include worst case conditions of supply voltage, temperature and threshold levels, plus actual load situations.
- Table shows maximum values and justifications for limitations.

	Individual memory accesses					Board configuration examples		
	1	2	3	4	5	6	7	8
	SRAM 4 MiB	SRAM 32 MiB	SDRAM	MRAM (Boot / Storage)	PROM (Boot)	SRAM 4MiB 2 MiB MRAM Boot 8 MiB MRAM Stor.	SDRAM 256MiB 2MiB MRAM Boot 8MiB MRAM Stor.	SRAM 4MiB 32 KiB PROM Boot 8 MiB MRAM Stor.
Max. clock freq. (MHz)	100 (read setup)	72.6 (read setup)	44.5 (address setup)	72 (address recovery)	83 (bus contention, no HL-sim)	72 (col. 4)	44.5 (col. 3)	72 (col. 4)
Max. freq w/o WS (MHz)	40.7 (read setup)	29 (read setup)	-"-	21.2-23.2 (SDRAM vs SRAM build, read setup)	18.5-19.9 (SDRAM vs SRAM build, read setup)	21.2 (col. 4)	21.2 (col. 4)	20.2 (read setup)
WS (read) at 50 MHz clock freq.	1 (read setup)	2 (read setup)	N/A	4 (read setup)	4 (read setup)	N/A	N/A	N/A
Peak bandwidth at 50 MHz (MB/s)	67 (4B x 50MHz / 3 cycles)	50 (4B x 50MHz / 4 cycles)	200 (if 4B x 50MHz / 1 cycle)	8.3 (1B x 50MHz / 6 cycles)	8.3 (1B x 50MHz / 6 cycles)	N/A	N/A	N/A

Example for SDRAM address setup: VDD=3.0V, T=+105ºC, Vthres↑=Vinh, load=actual plus FGPA)

DPU design results

Design methodology gives quality document output

The configurable flight design delivered

- DPU User manual
- DPU and Motherboard hardware design document
- DPU schematics configuration document
- Schematics (CAD, BOM, design files)
- PCB layout
- Quality documents
  - Failure Mode and Effects Analysis (FMEA)
  - Radiation analysis
  - Part Stress Analysis (PSA)
  - Timing Analysis (input to WCA)
  - Worst Case Circuit Analysis (WCA)
  - Risk and Feasibility Analysis (input to WCA)
- DPU Interface Control Document
- DPU verification documentation
- DPU Power Calculator
- DPU verification plan, tests, results

Designers reuse & adapt for specific JUICE instrument
The design has been approved by ESA.
DPU & SW property of Cobham Gaisler.

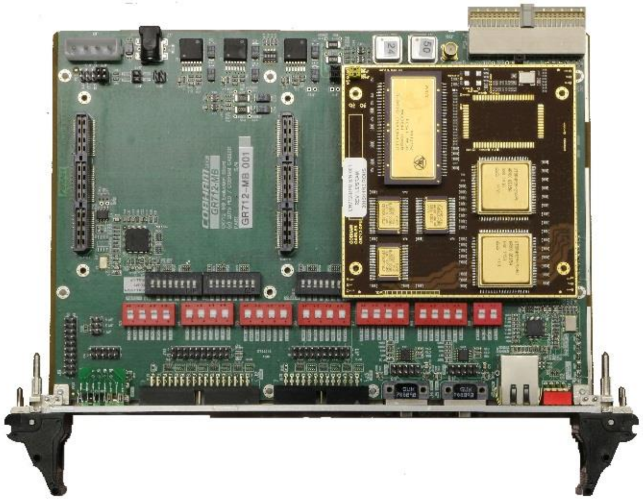
Prototype platform results

DPU prototype board

- Configuration of the DPU architecture
- EM quality for all major components and logic

Motherboard

- Commercial grade components
- I/O connectors and switch matrix
- FPGA expansion slot
- Configurable voltage levels
- Voltage/current measuring circuitry



Boot SW

Overview – from reset to executing application

Boot SW compliant with ESA Requirements TEC-SWS/10-373/FT "Flight Computer Initialisation Sequence‟ for Payloads. Later version now part of SAVOIR.

From reset to executing instrument application software. Executes in PROM uses main memory for data and buffers.

Boot SW, three major parts:

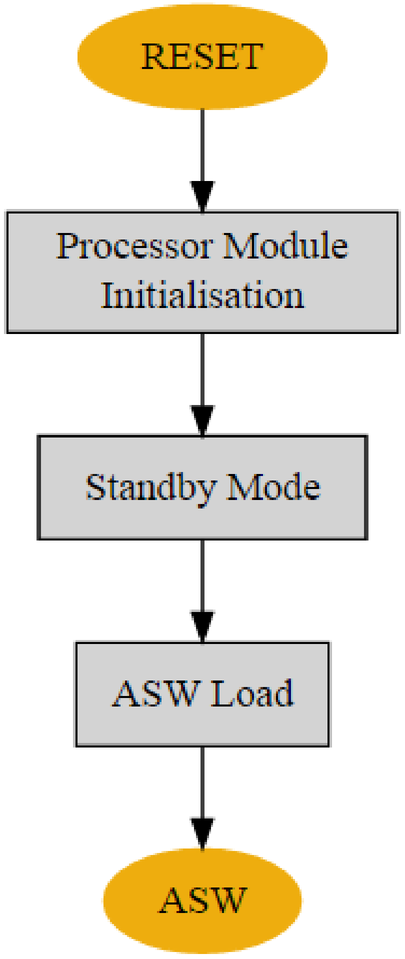
- Initialization and Self-tests of DPU
- Standby, provides PUS services to OBC over SpaceWire
- ASW load

Unit-tests

- Tests SW components on TSIM and HW
- Code coverage

Validation

- “Black box” test use cases on DPU HW controlled by PC
- “White box” test to support unit-tests



Boot SW

Overall Boot SW design

Hardware resources required:

- Interrupts always disabled
- Most exceptions triggers reset/restart
- Does not use FPU, CPU MUL/DIV
- EDAC on RAM, Caches, IU. BOOT/APP no EDAC
- Standby fits into the 16KiB I-Cache. D-cache is used
- Requires ~256KiB RAM
- Requires <20KiB PROM
- SpaceWire interface 0 and 1, RMAP target
- Timer and Latching-Timer (timestamps received time-code for TDP)

Implements a small run-time itself:

- Does not rely on BCC run-time, LibC, RTEMS or 3rd party SW.
- Toolchain independent, but validated using GCC-4.4.2 (BCC).

Code languages:

- Assembly: Run-time, low-level Init before memory, Final ASW boot.
- C-language: Remaining Init, Standby, ASW boot.

No automatic code generation or code reuse.

Configurable design:

- DPU HW configuration, and
- Mission configuration

Boot SW

Initialization sequence

Initialization

- Triggered by a cold or warm restart
- Processor init (Low-level, FPU, IU, cache)
- Clock-gating init
- All GR712RC I/O registers init
- Memory controller initialization
- Self-test (IU, Cache, Memory, Memory Controller) and if successful enabled
- SRAM/SDRAM Memory initialization
- Record result in Boot Report memory area
- Sets up C-environment
- Start Standby mode

Low-level

- Single-vector trapping
- Window over/under-flow handling
- Interrupt always disabled
- Warm restart entry point for Application/Standby – restart without memory init/clearing.

Self-test approach

- Test before enabling/using feature
- Test only what is used by Boot SW to reduce causing potential of triggering errors in Boot SW.
- If a self-test fails it is reported in the Boot Report. If a test generates an exception the boot is restarted (similar to warm) with test disabled and indicated in boot report.

Boot SW

Memory layout

- Instructions always in PROM (not moved)
- DATA/BSS/STACK in RAM
- R/W DATA copied to RAM
- On-chip RAM 192KiB not used
- No dynamic allocation

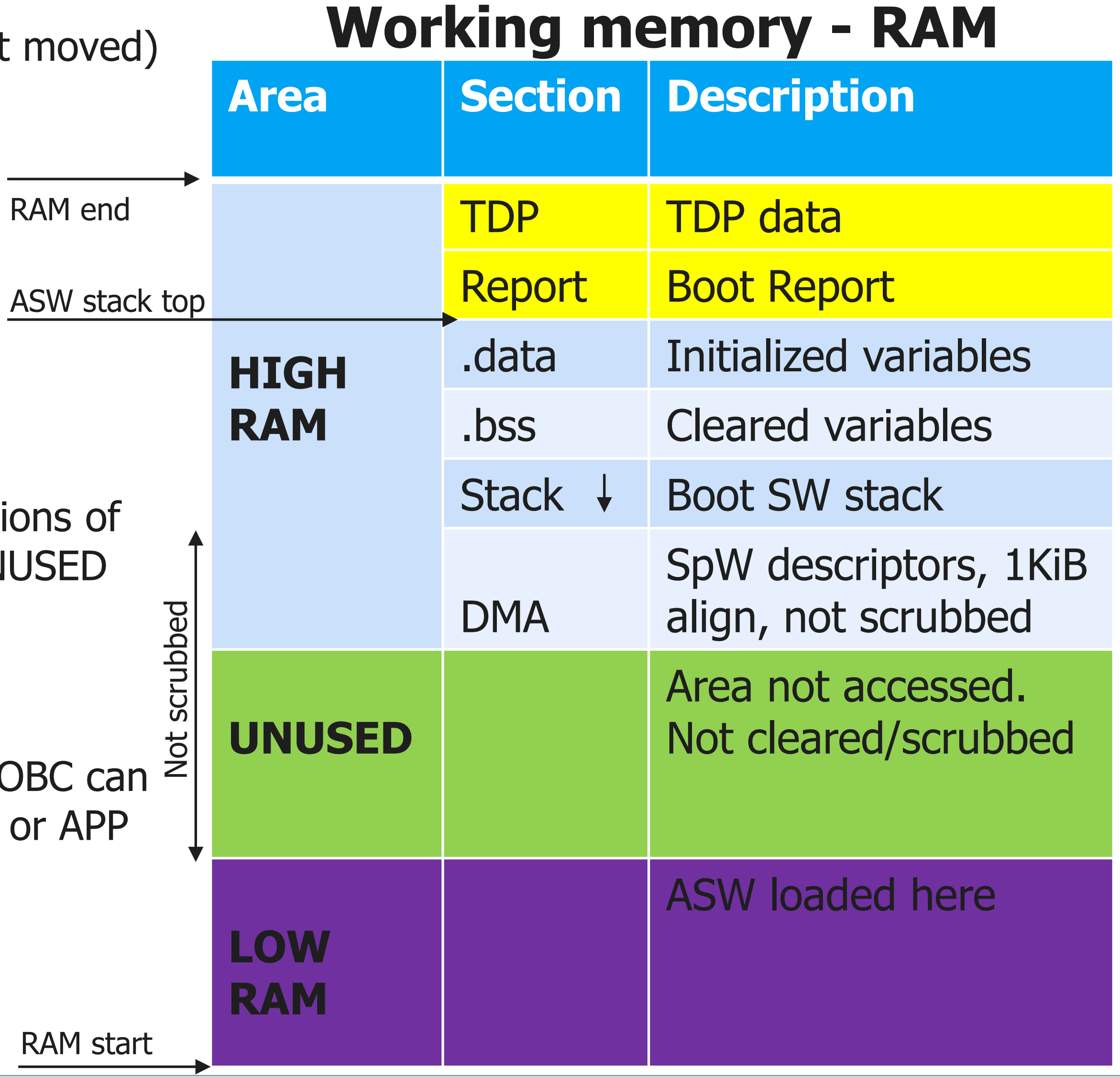
- PUS service can access all RAM
- RMAP can access all RAM
- User configure the size and locations of Memory Areas. LOWRAM and UNUSED sized depending on for example:
  - ASW size
  - RAM size (scrub time, etc.)
- TDP and boot report preserved, OBC can access them regardless of BOOT or APP

🟨 Preserved

🟩 Unused, preserved after warm restart

🟪 Primary ASW area

🟦 Boot SW area



Working memory - RAM

Area	Section	Description
HIGH RAM	TDP	TDP data
	Report	Boot Report
	.data	Initialized variables
	.bss	Cleared variables
	Stack ↓	Boot SW stack
	DMA	SpW descriptors, 1KiB align, not scrubbed
UNUSED		Area not accessed. Not cleared/scrubbed
LOW RAM		ASW loaded here

Boot SW

Standby design

Entering Standby

- Always enter Standby. If SpW link start fail or timeout on SpW PUS TC then Standby will exit and continue to load primary application.
- TM(5,1) Boot Report always sent during boot to let OBC detect power-up.
- SpW link management and selection (in case of dual redundant)

PUS terminal over SpaceWire

- PUS – Packet Utilization Standard
  - ECSS-E-70-41A
- DPU processes TC commands from OBC
- Generate TM responses
- Checksums

Standby Services (maintainence)

- SpaceWire networking - reusing lowlevel parts of SpW driver
- CCSDS Packet Transfer Protocol
- PUS TC/TM services (1, 3, 5, 6, 8, 17)
- Memory service (up/down-load, check) to all memories/registers. Allows partial patching and memory copying.
- House keeping periodic messages
  - Contains information used for validation
- Time Distribution Protocol (TDP) Slave, uses RMAP + time-code (latching-timer)
- Memory scrubber
- Watchdog management
- App storage management. MRAM is held in reset/sleep at all times until accessed by Standby. Write protection, controlled by PUS service and checked by Standby on access to avoid CPU exception.

Boot SW

Standby execution model

Standby uses a simple **Cyclic Execution** model.

Divided into 10 execution slots per second. Each slot are not allowed to execute longer than 1/10s.

Easy to measure and meet deadline with this approach. HK includes worst-case slot time.

Validation results: slot <20ms @50MHz

- TDP runs at least twice a second
- 5 TCs per second
- Configurable scrub rate by sizing scrub block assuming scrub period is 1/10s

Slot	ID	Allocated time
0	link	1/10s
1	tc	1/10s
2	tdp	1/10s
3	tc	1/10s
4	hk	1/10s
5	tc	1/10s
6	scrub	1/10s
7	tc	1/10s
8	tdp	1/10s
9	tc	1/10s

ID	Description
link	Link reconfiguration task
tc	PUS TC task, handles at most one PUS telecommand
tdp	Time Distribution Protocol task for maintaining local time
hk	Housekeeping task
scrub	Memory scrubber task

Boot SW

Application SW load

Load sequence

- Three applications supported:
  - RAM_0 (Optionally selected by Standby)
  - ASM_0
  - ASM_1
- Reads application from ASW format, section by section:
  1. Verify ASW source location
  2. Copy ASW
  3. Verify ASW at destination location
- If CRC failure try next image in order.

No constraints where ASW is located in MRAM or RAM, it is up to user to manage.

At application entry

- Sets up defined HW/SW environment
- Kick Watchdog
- Boot Report and TDP area preserved, stack below Boot Report avoid overwriting

ASW format:

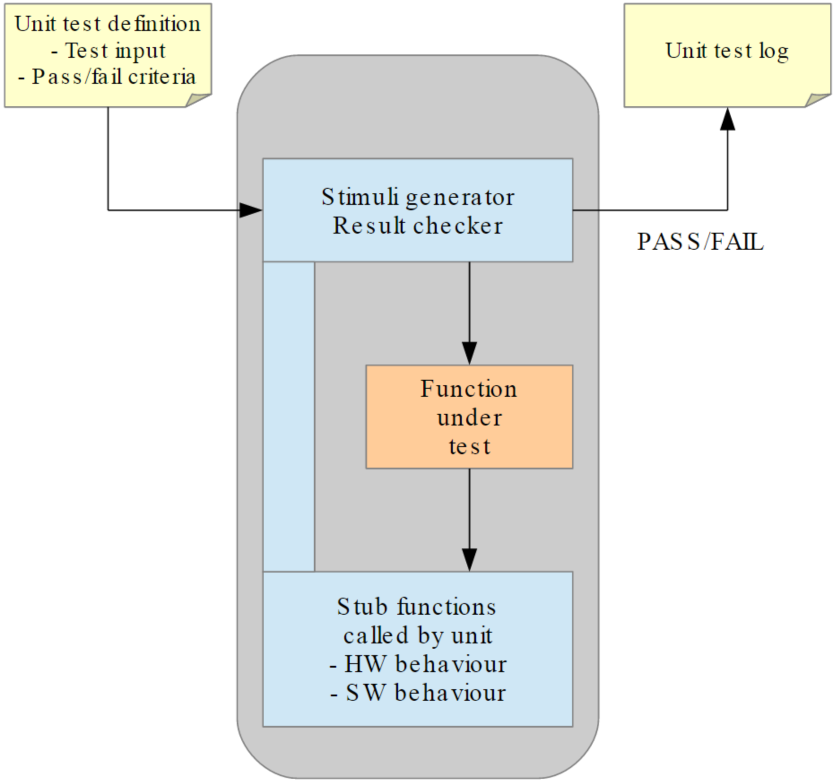
- One Image header (CRCed)
  - Entry Point
  - N Sections (Array length configurable)
    - Enable/Disable (to allow patching)
    - Source address
    - Destination address
    - 32-bit Length
    - Data CRC
- Data Block

Boot SW

Unit testing

Approach

- Automated test framework on
  - DPU with GRMON2,
  - TSIM2 simulator with GR712RC I/O module.
- Instruction code decision coverage on object files from TSIM2.
- Unit-test each component separately. External calls ends up in "Fake" Components that mimics operation and tries to trigger different cases.
- Stimuli function input generator. Verify function output and external calls.



Results:

- 98% code coverage on unit-tested code
- 15-20% of the code tested by validation instead

Boot SW

Validation

Overview

- Ensures that all SW requirements are covered by the implementation.

Test set up:

- DPU board (WD -> reset)
- Linux Workstation
- GRMON2 (debug-link -> DPU)
- GRESB (SpW0 and SpW1 <-> GRESB)
- Test Framework and tests cases

Test Framework

- Developed within activity.
- Automated testing using Tcl scripting and GRMON2 Scripting.
- Wireshark plugin for GRESB PUS TM/TC packet debugging
- Test report creation and logs results.

"Black box" testing

- Scripted model of OBC generating different SpW requests
- Tcl script generates SpW/PUS TC cmds over GRESB SpaceWire network
- Validate SpW/PUS TM responses/reports.
- SpW Link, Time distribution, house keeping, PUS TC/TM protocol etc.
- Standby internal validation parameters to measure worst-case time slot is always active and part of house keeping.

"White box" testing

- Inspect internal Boot SW and system state by GRMON access
- Reaches certain location?
- Correct data/register at a specific location?
- Inject errors to validate self-tests.

Driver SW

Overview

Constraints

- RTEMS-4.10 used. Edisoft RTEMS 4.8 was evaluated but not selected.
- Implemented in C99
- Small OS abstraction layer to isolate OS.
- No active objects like Tasks, Semaphores, MsgQ, Events etc.
- Global Interrupt disabling (PSR.PIL=0xf) for protection.
- Drivers have a “Low-level” approach.

Unit-testing

- Approaches similar to Boot SW
- Worst-case execution measured on HW

Drivers

- SpaceWire
- SPI - user ISR can use low-level driver
- UART - FIFO support, interrupt and polling mode
- GPTIMER, GRTIMER Latching Timer and watchdog interface
- GPIO
- AHB status register
- Clock-gating
- Memory Controller

Validation

- Smaller validation package
- Tests different driver use cases on DPU prototype
- TSIM used for testing some drivers.

Driver SW

SpaceWire driver overview

The major focus of the SW driver library.

- Most complex HW, interface and driver.
- Design inspired by CG's public GRSPW Packet. Reduced the complexity by removing SW FIFOs, work task and interrupt processing.
- Zero-copy Packet DMA interface.
- Driver operations access directly into descriptor table. User sees descriptor table as a FIFO Queue of packets.
- Driver processes sending and receiving of Linked Lists of SpW packet buffers. Six DMA transmission operations:
  - TX_SEND
  - TX_RECLAIM
  - TX_FLUSH
  - RX_PREPARE
  - RX_FLUSH
- SW SpW link interface independent from Packet DMA interface.
- User ISR responsible for calling low-level.

Tools used during SW development

TSIM2 Simulator - used and adapted for DPU

TSIM2 LEON3 simulator with GR712RC I/O module used during the project.

Part of an add-on later in the project TSIM2 was adapted to DPU HW platform to be used as a SW validation tool by Instrument SW developers. The adaptations was made available in a separate DPU4JUICE I/O module, examples and documentation.

- Configuration files: support for every DPU memory configuration.
- Specific getting start documentation.
- APBUART FIFO model implemented.
- SpaceWire model improved, with error injection in order to test SpW/PUS protocol.
- Examples how to validate that MRAM application memory timing vs GPIO reset access is correct

Tools used during SW development

GRESB Ethernet to SpW bridge

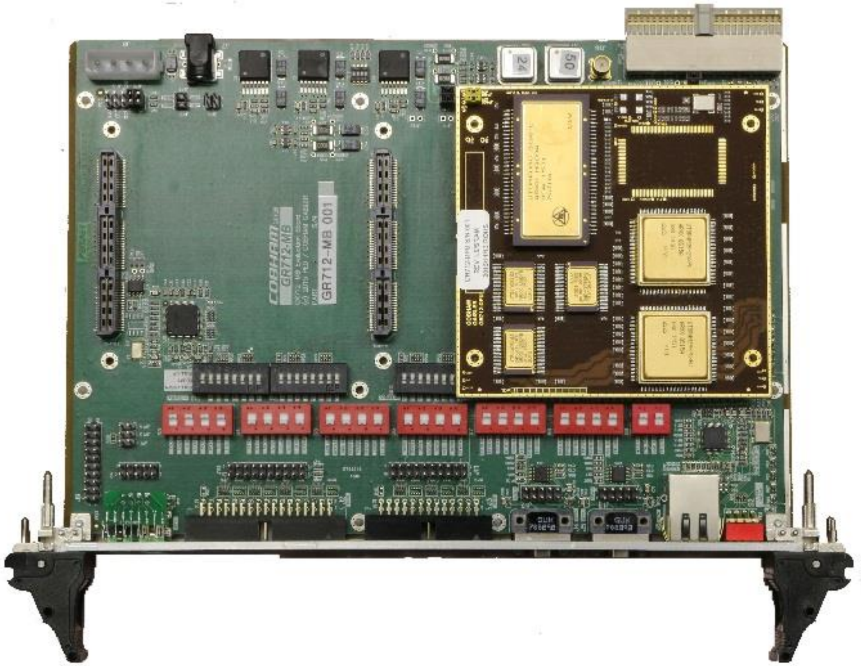
- Easy to use **TCP/IP Ethernet**<->SpaceWire bridge.
- Web interface to configure GRESB routine table
- Fast GRMON debug link to DPU (can replace JTAG).
- Interface multiple GRMONs to multiple DPUs in parallel, i.e. to speed up debugging for dual-redundant Instrument configuration.
- Easy TCP/IP protocol to,
  - Send/receive SpW packets
  - Control link, routing table, link status/errors
  - Send SpW time-code
  - Control 16 GPIO to control/emulate/monitor signals
- Supports bursts of 100Mbit/s SpW interface used in JUICE
- Used in Boot SW validation

Contact: sales@gaisler.com
www.gaisler.com

Tools used during SW development

DPU Motherboard and GR712RC development board

GR712RC Development board,
commercial components



JUICE DPU + Motherboard.
Compact PCI 6u form factor.
(Only 5V power taken from rack)

Future and conclusions

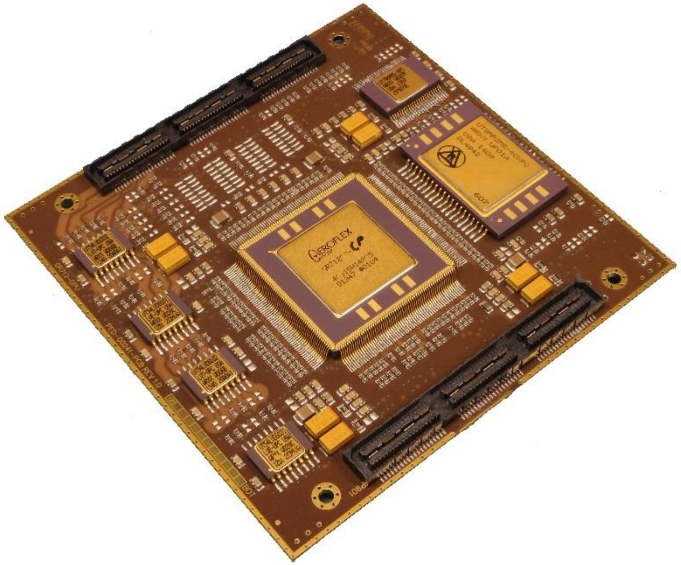
Future

- Application loader and image format part of GR716 ROM code.
- Can be split in two parts and still compatible with ESA “Flight Computer Initialisation Sequence”
  - Init, Self-test, App/standby loader
  - Standby / Application
- Extend to new targets, GR740

Conclusions

- Cobham Gaisler has developed a DPU flight hardware platform
  - Schematics, layout, component selection, analysis, etc.
  - Radiation hardened components
  - Based on GR712RC
- DPU compatible with requirements for the 10 instruments on ESA JUICE spacecraft
- DPU design (parts or in whole) planned to be used in 7 Instruments on-board JUICE
- Cobham Gaisler owned product, not limited to ESA or European area. Possible to reuse in other missions.

Thank you for listening!



Home page: www.gaisler.com

For questions contact: info@gaisler.com

DPU support and tools available

DPU Prototype System

- Motherboard Compact PCI 6U form factor - commercial components
- DPU - EM components supporting all configurations.
- Configured at DPU assembly.
- DPU can cover all memory configurations except Boot PROM. SRAM 4, 8, 16 and 32MiB or SDRAM 256MiB.
- PROM is MRAM to allow easy prototyping.
- Configurations selected by zero-ohm resistors. For example:
  - ROMSN[0] - can be 2MiB MRAM, 8MiB MRAM, or via expansion slot
  - ROMSN[1] - can be 2MiB MRAM, 8MiB MRAM, or via expansion slot
- SDRAM RadClock configurable from MB dip-switches
- JTAG debug-link via MB. 2xUART via USB.
- Ethernet transceivers on MB.
- 2xSpaceWire via DPU transceivers, 2 extra via MB pin-header
- GPIO and GPIN pin header on front panel
- 6xUART via MB external mezzanine (6xRS232 available
- MIL-1553B A&B-buses supported via MB extra mezzanine
- Power regulator with DIP-switch sets VDD in 3 steps/source - 3.3V±10%, 1.8±10%
- i2c TI voltage/current monitors on separate GR712RC, other DPU and FPGA Power lines to monitor. Accessed via USB or GR712RC.

MB FPGA expansion slot

3.3V, 1.8V and configurable (1.2V default)

22 SWMX - 2xSpW, UARTs, GPIO, SPI, I2C,…

Memory bus (Address, data, control)

Clock, reset, Wdog