

**Embedded multi-core systems for  
mixed criticality applications  
in dynamic and changeable real-time environments**

Project Acronym:

**EMC<sup>2</sup>**

**Grant agreement no: 621429**

<b>Deliverable no. and title</b>	<b>D4.10 – Report describing the evaluation of final implementation, evaluation of innovation results</b>	
<b>Work package</b>	WP4	Multi-core Hardware Architectures and Concepts
<b>Task / Use Case</b>		T4.4 Increased availability and dynamic reconfiguration
<b>Subtasks involved</b>	T4.4 Increased availability and dynamic reconfiguration	
<b>Lead contractor</b>	Infineon Technologies AG Dr. Werner Weber, mailto: <a href="mailto:werner.weber@infineon.com">werner.weber@infineon.com</a>	
<b>Deliverable responsible</b>	UTIA Jiri Kadlec, <a href="mailto:kadlec@utia.cas.cz">kadlec@utia.cas.cz</a> , + 420 2 6605 2216	
<b>Version number</b>	v1.0	
<b>Date</b>	01/04/2017	
<b>Status</b>	Final version	
<b>Dissemination level</b>	Public	

**Copyright: EMC<sup>2</sup> Project Consortium, 2017**

## Authors

Participant no.	Part. short name	Author name	Chapter(s)
04D	UTIA	Jiri Kadlec, Zdenek Pohl Lukas Kohout	Dynamic Reconfiguration of Asymmetric Multiprocessing Platform
18D	Sundance	Flemming Christensen	EMC2-DP HW platform provider
15E	Tecnalía	Elena Terradillos Asier Alonso Muñoz Daniel Múgica	Reliable and Self-Healing Dynamic Reconfiguration Manager (DRM) on LADAP platform (Virtex-5)
15F	TASE	Manuel Sanchez	
16A	Chalmers	I. Sourdis	Resilient Reconfigurable Multiprocessor arrays: probabilistic analysis of availability

## Document History

Version	Date	Author name	Reason
V0.1	25/02/2017	Jiri Kadlec, UTIA	Initial template with UTIA input
V0.2	26/02/2017	Flemming Christensen Sundance	Sundance exploitation path
V0.3	27/02/2017	I. Sourdis, Chalmers	Resilient reconfigurable multiprocessor arrays
V0.4	11/03/2017	Elena Terradillos Jiri Kadlec	Reliable and Self-Healing Dynamic Reconfiguration. References. Draft version for the internal WP4 review.
V0.5	12/03/2017	Jiri Kadlec	Performance tables for EMC2-DP moved to Annex A Licensing conditions (related to the EMC2-DP demonstrator evaluation packages) are moved to the Annex B
V0.6	18/03/2017	Jiri Kadlec	Released for the internal review to be performed by Flemming Christensen, Sundance Multiprocessor Technology Ltd (SMT)
V1.0	29/03/2017	Jiri Kadlec	Final version
	01/04/2017	Alfred Hoess	Final editing and formatting, deliverable submission

## Publishable Executive Summary

This report is focusing on the evaluation of final implementation, evaluation of innovation results of demonstrators prepared by WP4 partners active in *task T4.4 increased availability and dynamic reconfiguration* in the EMC<sup>2</sup> project. It describes the following two demonstrators:

- Dynamic Reconfiguration of accelerators on Zynq device on the industrial EMC2-DP-V2 platform (UTIA with Sundance Multiprocessor Technology Ltd (SMT))
- Reliable and Self-Healing Dynamic Reconfiguration – (Tecnalia with TASE)
- Resilient reconfigurable multiprocessor arrays – (Chalmers)

We can position the presented technology in this wider picture of WP4 in *Figure 1*.

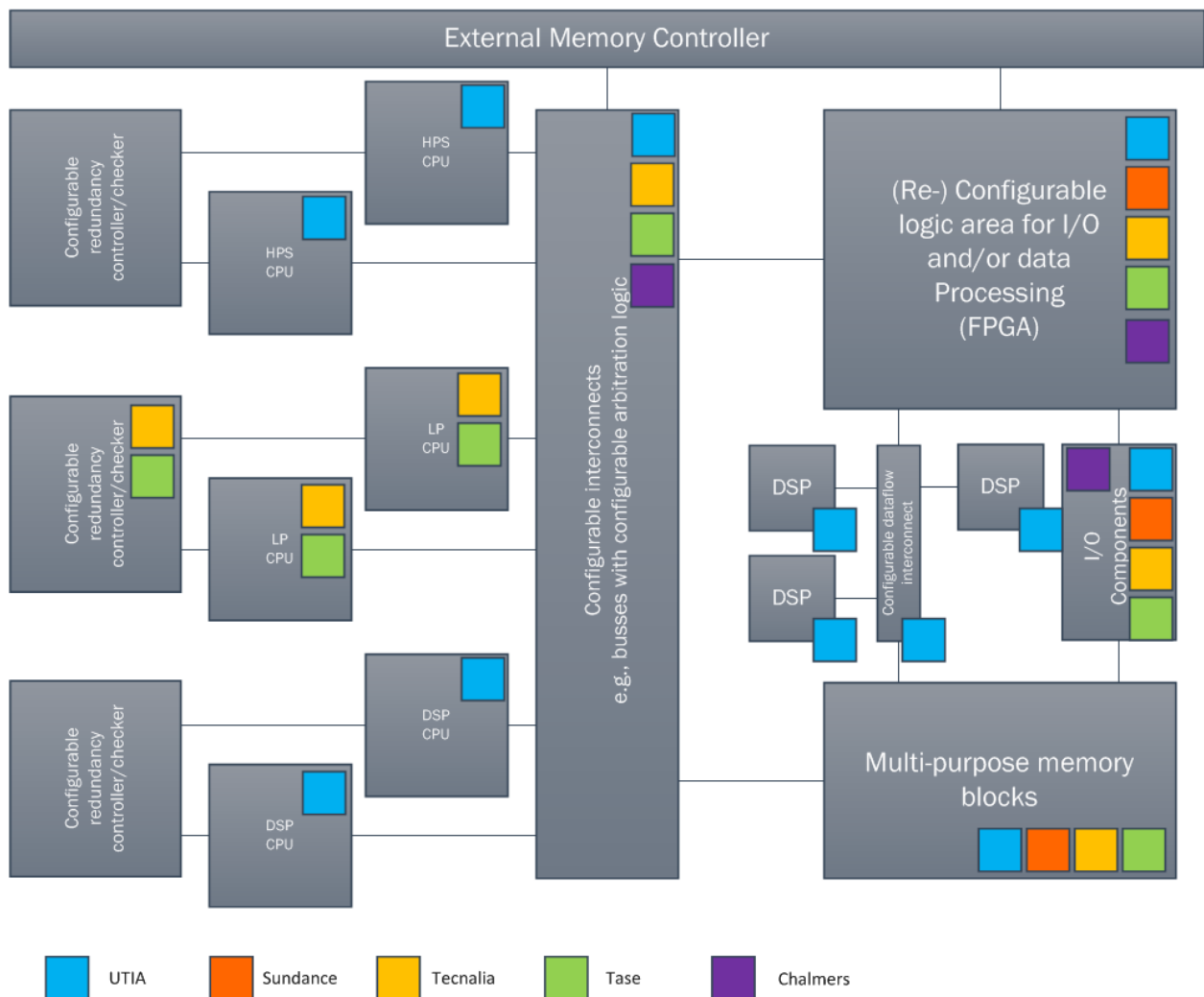


Figure 1: A generic hardware architecture and a partner activity by sector

## Table of contents

1. Introduction .....	6
2. Dynamic Reconfiguration of Accelerators on Zynq Platform .....	7
2.1 Introduction .....	7
Development period M6-M18 (10.2014-9.2015) .....	7
Development period M19-M24 (10.2015-3.2016) .....	7
Development period M25-M30 (4.2016-9.2016) .....	8
Development in the period M31-M36 (10.2016-3.2017) .....	8
Final Demonstrator [11] - Full HD HDMI I/O and three EdkDSP Accelerators.....	9
2.2 Architecture of the final demonstrator [11] .....	10
2.3 Configurations of Video Processing Accelerators and EdkDSP Accelerators: .....	12
General setup of all demos of the final demonstrator [10]: .....	12
Main objectives of the demos in [10]: .....	12
Edge detection .....	13
Motion detection .....	13
Measurements of acceleration and resources used for the final demonstrator [11] .....	13
Floating point performance .....	14
2.4 Summary of parameters of the final demonstrator [11].....	14
2.5 Conclusions .....	15
Final demonstrator [11] with Zynq XC7Z030-1I device [26] .....	15
2.6 Exploitation .....	15
UTIA AV CR v.v.i. ....	15
Sundance Multiprocessor Technology Ltd. ....	16
2.7 Conditions for the access to the final UTIA demonstrator [11].....	16
3. Reliable and Self-Healing Dynamic Reconfiguration Manager (DRM) on LADAP platform (Virtex-5).....	17
3.1 Update on final DRM design.....	18
3.2 DRM design functional validation .....	19
4. Resilient Reconfigurable Multiprocessor arrays: probabilistic analysis of availability and efficiency .....	26
4.1 Probabilistic Analysis of a mixed-grained reconfigurable array .....	27
4.2 Highlights of our T4.4 contribution.....	29
5. Conclusions .....	30
6. References .....	31
7. Annex A – Performance Measurements for the Final EMC2-DP Demonstrator.....	34
7.1 Project sh01: Edge detection with single HW accelerator and 3x EdkDSP .....	34
7.2 Project sh02: Edge detection with two HW accelerators and 3x EdkDSP .....	36
7.3 Project sh03: Edge detection with three HW accelerators and 3x EdkDSP .....	38
7.4 Project md01: Motion detection with chain of HW accelerators, 3x EdkDSP .....	40
8. Annex B – UTIA Licensing of the EMC2-DP Demonstrator Evaluation Packages.....	42
The evaluation version of the package [11] can be downloaded from UTIA www pages [10] free of charge.....	42
Vivado projects with the evaluation version of the (8xSIMD) EdkDSP IP for the Artemis EMC2 project partners.....	44
Vivado projects with the release version of the (8xSIMD) EdkDSP IP .....	46
Disclaimer .....	48

## List of figures

Figure 1: A generic hardware architecture and a partner activity by sector .....	3
Figure 2: EMC2-DP-V2 carrier, TE0715-03-30-11 module. Edge detection demo with 3 HW data paths and LMS filter in one of the three (8xSIMD) EdkDSP accelerators and the ILA debugger.....	10
Figure 3: Full HD HDMII-HDMI platform with three (8xSIMD) EdkDSP accelerators. ....	11
Figure 4: Sundance exploitation path .....	16
Figure 5: System architecture diagram of the Reliable and Self-Healing DRM.....	18
Figure 6: Timing violations when trying to implement the DRM design at 125 MHz .....	19
Figure 7: DRM implementation at 100 MHz – most critical paths timing.....	19
Figure 8: w SEU simulation command .....	20
Figure 9: b SEU simulation command.....	20
Figure 10: Configuration Memory Monitor detects frame with one correctable bit error .....	21
Figure 11: Configuration Memory Monitor detects frame with two bit errors .....	22
Figure 12: Configuration Memory Monitor detects global CRC error .....	22
Figure 13: Scrubbing code at main loop level .....	23
Figure 14: FAR checking routine .....	24
Figure 15: SEFI_FLAG assertion after unsuccessful FAR checking .....	24
Figure 16: Example of CG and FG reconfiguration constructing 3 fault-free components: component-1 (A{FG}, B2, C1, D1), component-2 (A1, B3, C2, D2), and component-3 (A3, B4, C3, D4). ....	27
Figure 17: Project sh01 - Acceleration and HW resources used. ....	34
Figure 18: Power consumption of sh01 demo without ILA.....	35
Figure 19: Power consumption of sh01 demo with ILA.....	35
Figure 20: Project sh02 – Acceleration and HW resources used. ....	36
Figure 21: Power consumption of sh02 demo without ILA.....	37
Figure 22: Power consumption of sh02 demo with ILA.....	37
Figure 23: Project sh03 - Acceleration and HW resources used. ....	38
Figure 24: Power consumption of sh03 demo without ILA.....	39
Figure 25: Power consumption of sh03 demo with ILA.....	39
Figure 26: Project md01 - Acceleration and HW resources used .....	40
Figure 27: Power consumption of md01 demo without ILA .....	41
Figure 28: Power consumption of md01 demo with ILA .....	41

## 1. Introduction

This deliverable describes evaluation of final implementation, evaluation of innovation results for the EMC2 partners involved in task T4.4. This task is dedicated to increasing the availability of a multicore system employing a safety management unit and dynamic reconfiguration. Presented technology can be used in the EMC2 living labs/demonstrators.

Work package 4 is focusing on four technology lanes:

- (1) Heterogeneous Multiprocessor SoC architectures
- (2) Dynamic reconfiguration on HW accelerators and reconfigurable logic
- (3) Networking
- (4) Virtualisation and verification technologies

This deliverable describes mainly technology lanes (1) and (2). Contributions to these technology lanes are indicated in each chapter.

UTIA describes final demonstrator [11] consisting of the application note and the evaluation package for the EMC2-DP-V2 platform [23] provided by partner SMT. The demonstrator supports implementation of reconfiguration of HW functionality as part of the Asymmetric Multiprocessing on the 28nm Zynq device. It is described in Chapter 2.

The dynamic reconfiguration is operating with three floating point accelerators implemented in the programmable logic fabric. Accelerators are individually reconfigured at run-time by changing their internal controller firmware. These controllers are scheduling vector floating point operations. This dynamic runtime re-programmability enables an increase in the availability of the complete system in case of a permanent error, because the accelerated tasks can be dynamically moved from one accelerator to another. This is handled by the software running on the MicroBlaze soft-core processor, operating in the asymmetric multiprocessing mode together with the ARM processor of the Zynq platform. The triple redundancy is supported in case of three EdkDSP accelerators. The main contributors are UTIA and SMT.

In Chapter 3, Tecnia and TASE describe the implementation of complete dynamic partial reconfiguration of an FPGA for the space domain. A platform will be built which allows dynamic reconfiguration of a Xilinx Virtex 5 FPGA with support of reliability and self-healing related features. It will take into account these issues, in the context of subtasks T4.3 and T4.4 and in relation to the living lab 3 (space applications).

Theoretical results related to resilient reconfigurable multiprocessor arrays are described by partner Chalmers in Chapter 4.

The last two chapters contain summary of the evaluation of the final implementation of demonstrators and References.

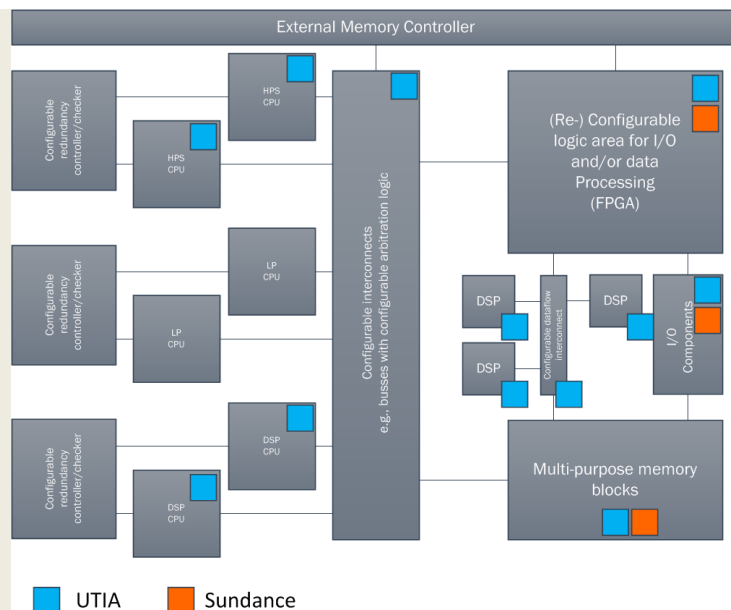
Appendix A provides summary of measured parameters (power consumption, performance, resources used) for the EMC2-DP-V2 platform [23].

Appendix B provides licensing conditions for the released evaluation packages for the final demonstrator [11].

## 2. Dynamic Reconfiguration of Accelerators on Zynq Platform

Technology lanes covered:

- *Heterogeneous Multiprocessor SoC architectures*
- *Dynamic reconfiguration on HW accelerators and reconfigurable logic*
- *Networking*
- *Virtualization and verification technologies*



### 2.1 Introduction

This chapter describes UTIA demonstrator of dynamic reconfiguration of accelerators.

#### Development period M6-M18 (10.2014-9.2015)

Initial development was completed in period M18 (see D4.6 [1] and D4.7 [2]). Runtime reconfigurable accelerators have been implemented in Xilinx Vivado 2013.4 with released C demo projects for Xilinx SDK 2013.4 for the development board ZC702 with chip XC7020-1C. Main outcomes of the development:

- Documented flow supporting the boot, co-debug and execute the ARM Cortex A9 programs
- Compare performance of EdkDSP reconfigurable computing elements and NEON accelerator of ARM.

The ZC702 board is used by universities and it is one of the base demo boards used by Xilinx when releasing new concepts and demos. However, the board is not really suitable for embedded industrial applications due to its size. See public application notes and evaluation packages [20], [21].

#### Development period M19-M24 (10.2015-3.2016)

In the next development period ending in M24 (see D4.8 [3] and D4.16 [6]), we have introduced compatibility of the EdkDSP reconfigurable computing element with the Xilinx software defined system on chip environment (SDSoC 2015.4) [34]. Main outcomes of development:

- Documented flow supporting the boot, co-debug and execute the ARM Cortex A9 programs for
  - Design in SDSoC 2015.4 environment (C/C++ system level compiler based on LLVM)
  - Deployment of exported HW in SDK 2015.4 (Free C development environment)
  - Designed as SDSoC 2015.4 BSP with Full HD HDMI-In and HDMI-Out and single (8xSIMD) EdkDSP reconfigurable accelerator
- Board support package for the SMT industrial grade EMC2-DP-V1 development platform

The EMC2-DP-V1 HW platform used the Zynq XC7Z015-1C device in the TE0715-03-15-1C System on module. The space in the programmable logic was sufficient for a single (8xSIMD) EdkDSP floating point accelerator.

The public application notes and evaluation packages [18] document released designs working with different grades of the Zynq XC7Z020 device. Designs support the HDMI-HDMI interface and work on the TE701-05 carrier [30].

### Development period M25-M30 (4.2016-9.2016)

In D4.9 report [4], we have presented results achieved in period M25-M30. At this stage, we have been able to support (8xSIMD) EdkDSP reconfigurable computing elements with the Xilinx software defined system on chip environment (SDSoC 2015.4) [34]. Main results:

- Documented flow supporting the boot, co-debug and execute the ARM Cortex A9 programs for
- Two (8xSIMD) EdkDSP reconfigurable computing elements for Zynq XC7Z020-2I device on the TE0720-03-2IF SoM module.
- Three (8xSIMD) EdkDSP reconfigurable computing elements for Zynq XC7Z030-1I device on the TE0730-03-30- 1I SoM module with and without the In-circuit Logic Analyzer (ILA)
- Design in SDSoC 2015.4 environment [34] with export to SDK 2015.4 [33].
- Designed as SDSoC 2015.4 BSP with Full HD HDMI-In and HDMI-Out with two and three (8xSIMD) EdkDSP reconfigurable accelerators enabling the dynamic task migration and the triple redundancy in case of the Zynq XC7Z030-1I device on the TE0715-03-30- 1I SoM module [26].
- Board support package for Zynq XC7Z020-2I device on the TE0720-03-2IF SoM module [25] and TE0701-05 carrier [30]. See application notes and evaluation packages [15], [16] and [17].
- Board support package for Zynq XC7Z030-1I device on the TE0715-03-30-1I SoM module [26] and SMT's EMC2-DP-V2 carrier [22], [23], [24]. See application note and evaluation design [14]. It describes board support packages (Standalone and Linux) with Full HD HDMI Input/Output.

### Development in the period M31-M36 (10.2016-3.2017)

In this chapter of the D4.10 report, we present final results achieved in period M31-M36 with description of parameters of the final demonstrator on the EMC2-DP-V2 carrier [22], [23], [24]. We support three (8xSIMD) EdkDSP reconfigurable floating point accelerators with the Xilinx software defined system on chip environment (SDSoC 2015.4) [33] for Zynq XC7Z030-1I device on the TE0720-03-30-1I [25] SoM module and the standalone EMC2-DP-V2 carrier board [21], [22], [23] developed by SMT.

We have realised application notes and the evaluation designs for the final demonstrator [11]. It includes the HW accelerated Full HD HDMI video processing and the HW accelerated floating point filters computed in (8xSIMD) EdkDSP accelerators on the largest Zynq platform with the Kintex PL fabric.

This final demonstrator [11] is supported by free Xilinx Vivado 2015.4 and SDK 2015.4 tool chain [33] for the standalone EMC2-DP-V2 carrier board [24] developed by SMT. It demonstrates:

- 3 edge detection video processing designs (sh01, sh02, sh03)
  - These demos document the possibility to define different HW paths by different source C/C++ functions. This is important for covering of the borders lines of the parallel processed parts of the frame.
  - HW accelerators can be programmed for the number of processed micro-lines.
  - These demos enable efficient, synchronised parallel execution of accelerated data paths and ARM Cortex A9 standalone C code.
- 1 motion detection video processing design (md01)
  - This demonstrates the pipelined parallel execution of HW video processing accelerators.
  - HW accelerators work with fixed number of processed micro-lines (1080 micro-lines) in this case.

Each Full HD demo includes also the HW accelerated computation of two DSP filters. These single precision floating point filters are computed on one of the three (8xSIMD) EdkDSP run-time reprogrammable single precision floating point accelerators with these properties:



- C programs can be compiled for single MicroBlaze processor and for one of the three EdkDSP accelerators. Compiled C (and ASM) code can be executed by the accelerators, without the need to re-compile the design in Vivado 2015.4 [33].
- C programs for the MicroBlaze processor and for the three (8xSIMD) EdkDSP accelerators can be edited in the same SDK 2015.4 environment used for ARM Cortex A9 programming and debug.
- The three EdkDSP accelerators can run different programs in parallel and perform run-time change of tasks, task migration.
- Design is supporting the run-time re-programming of each of the (8xSIMD) EdkDSP accelerators, under the control of the user-defined MicroBlaze C program.
- The MicroBlaze processor executes its program and utilizes data located in the top 256 MBytes of the 1Gbyte DDR3 memory. This region is also accessible by ARM processor. ARM initiates and controls the content of program and data executed by the MicroBlaze.
- ARM and MicroBlaze programs use HW mutex for synchronization.

### **Final Demonstrator [11] - Full HD HDMI I/O and three EdkDSP Accelerators**

The application note [11] describes HW platform performing integration of three runtime reprogrammable (8xSIMD) EdkDSP floating point accelerators. These accelerators work in parallel with an edge detection or motion detection video processing algorithms. Source of video data is an HDMI input with resolution 1920x1080p60 (Full HD). The platform is composed from these HW building blocks (boards):

- Zynq XC7Z030-1I device on System on Module TE0715-03-30-1I from Trenz Electronic [26].
- Carrier board EMC2-DP-V2 with FMC connector from SMT [22], [23], [24].
- AES-FMC-HDMI-CAM-G FMC HDMI I/O extension board from Avnet [32].
- RS232 serial interface.

All implemented Full HD video processing algorithms have been developed, debugged and tested in Xilinx SDSoC 2015.4 environment [34].

SW algorithms have been compiled by Xilinx SDSoC 2015.4 system level compiler (based on the Xilinx HLS compiler) to Vivado 2015.4

HW projects, and compiled by Xilinx Vivado 2015.4 [33] to bitstreams for Zynq XC7Z030-1I device.

Created SW access functions controlling the HW accelerators have been exported from the Xilinx SDSoC 2015.4 environment to the Xilinx SDK 2015.4 [33] SW projects as static libraries for the standalone ARM Cortex A9 processor C programs.

See the running final demonstrator in *Figure 2*.

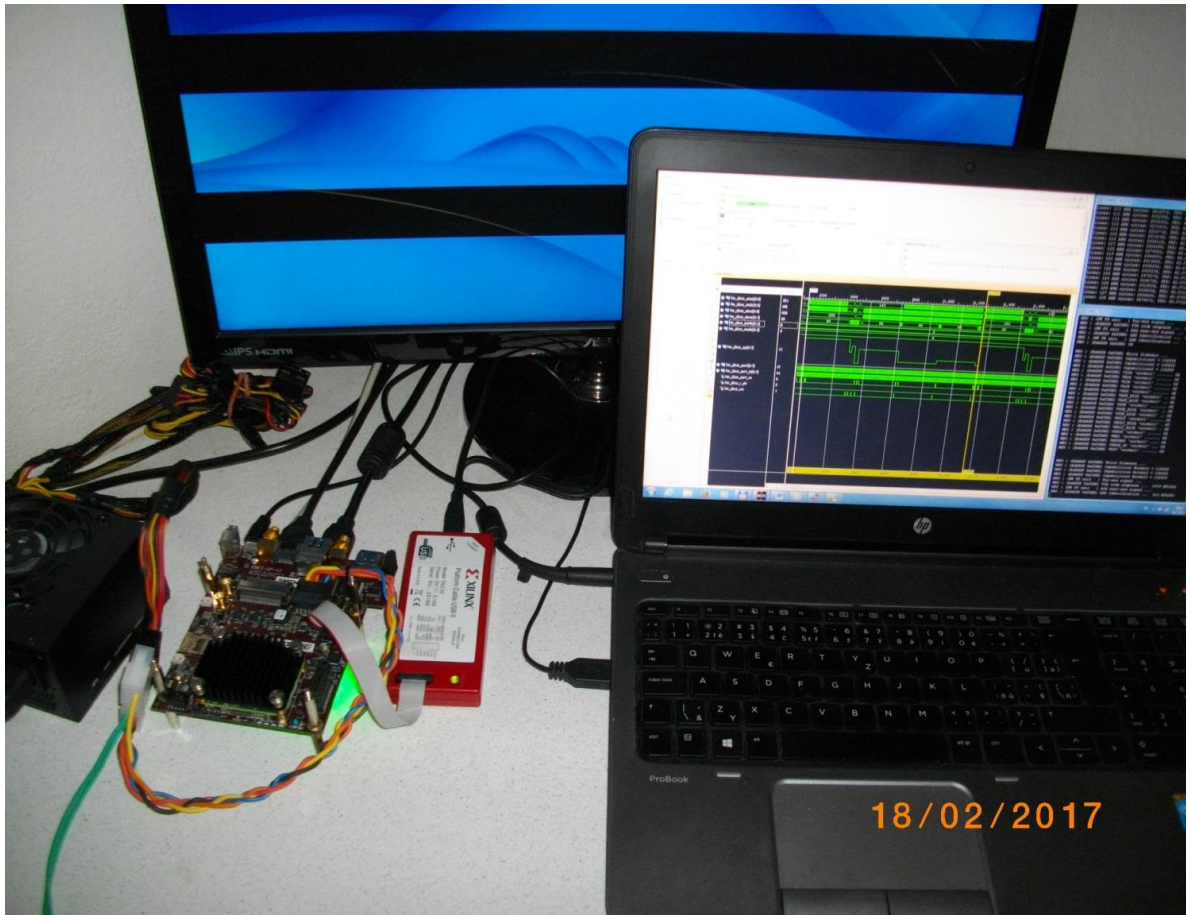


Figure 2: EMC2-DP-V2 carrier, TE0715-03-30-11 module. Edge detection demo with 3 HW data paths and LMS filter in one of the three (8xSIMD) EdkDSP accelerators and the ILA debugger.

## 2.2 Architecture of the final demonstrator [11]

The final demonstrator [11] works with system on module [26] with Xilinx Zynq device XC7Z030-1. It has two ARM Cortex A9 processors operating at 666 MHz. Memory controller of the Zynq device provides the DDR3 memory access ports for ARM processors as well as memory access for the programmable logic (PL) part of the device. The Zynq PL is used for (See Figure 3):

1. Three UTIA EdkDSP (8xSIMD) floating point processors (operating at 150 MHz) connected to Xilinx MicroBlaze 32bit processor (operating at 125 MHz).
2. Input chain of video processing Full HD data to input video frame buffers. The input video DMA (VDMA) controller is operating at 150 MHz.
3. Video processing HW accelerators and data movers defined in Xilinx SDSoC 2015.4 environment. These accelerators are controlled from the ARM Cortex A9 C programs compiled in SDK 2015.4 SW projects. These HW accelerators are operating at 200 MHz.
4. Chain of output video processing IPs connects output frame buffers to the Full HD display by HDMI cable. The output VDMA controller is operating at 150 MHz.

Three EdkDSP is 8xSIMD floating point accelerators are reprogrammable in runtime by change of firmware for the build-in PicoBlaze6 8bit controllers. These controllers are serving as schedulers of vector operations performed in the EdkDSP is 8xSIMD floating point data paths. These schedulers are programmed by simple C programs compiled by UTIA C compiler and assembler. These compilers respect the minimal resources of the PicoBlaze6 controllers.

The three EdkDSP 8xSIMD floating point accelerators are controlled by single 32 bit MicroBlaze processor. The MicroBlaze processor executes larger C programs from the DDR3 memory. Algorithms can benefit from execution of selected operations on three EdkDSP coprocessors. The EdkDSP coprocessors are connected to the MicroBlaze by local dual ported memories.

MicroBlaze C program can take benefit of the potential overlap of data communication from DDR3 to the EdkDSP dual-ported memories (managed by the MicroBlaze processor) and the parallel computations performed in the three EdkDSP accelerators and controlled locally by the three PicoBlaze6 sequencers. All designs include also the video processing chain of Full HD I/O IPs controlled by the ARM processor via the Axi-lite control bus operating at 125 MHz.

ARM Cortex A9 processor performs the global initialization and synchronisation of the video processing chain. The Arm program and the FPGA image is downloaded to the board from the Xilinx SDK 2015.4 via USB JTAG to the 1GB DDR3 located on the Zynq system on module.

System can be also started directly from the SD card with help of the ARM FSBL loader. ARM processor performs the initialization of program for the MicroBlaze processor. This MicroBlaze program contains also the initial firmware for the three EdkDSP accelerators. The ARM processor also initiates the HDMI video input and video output IPs.

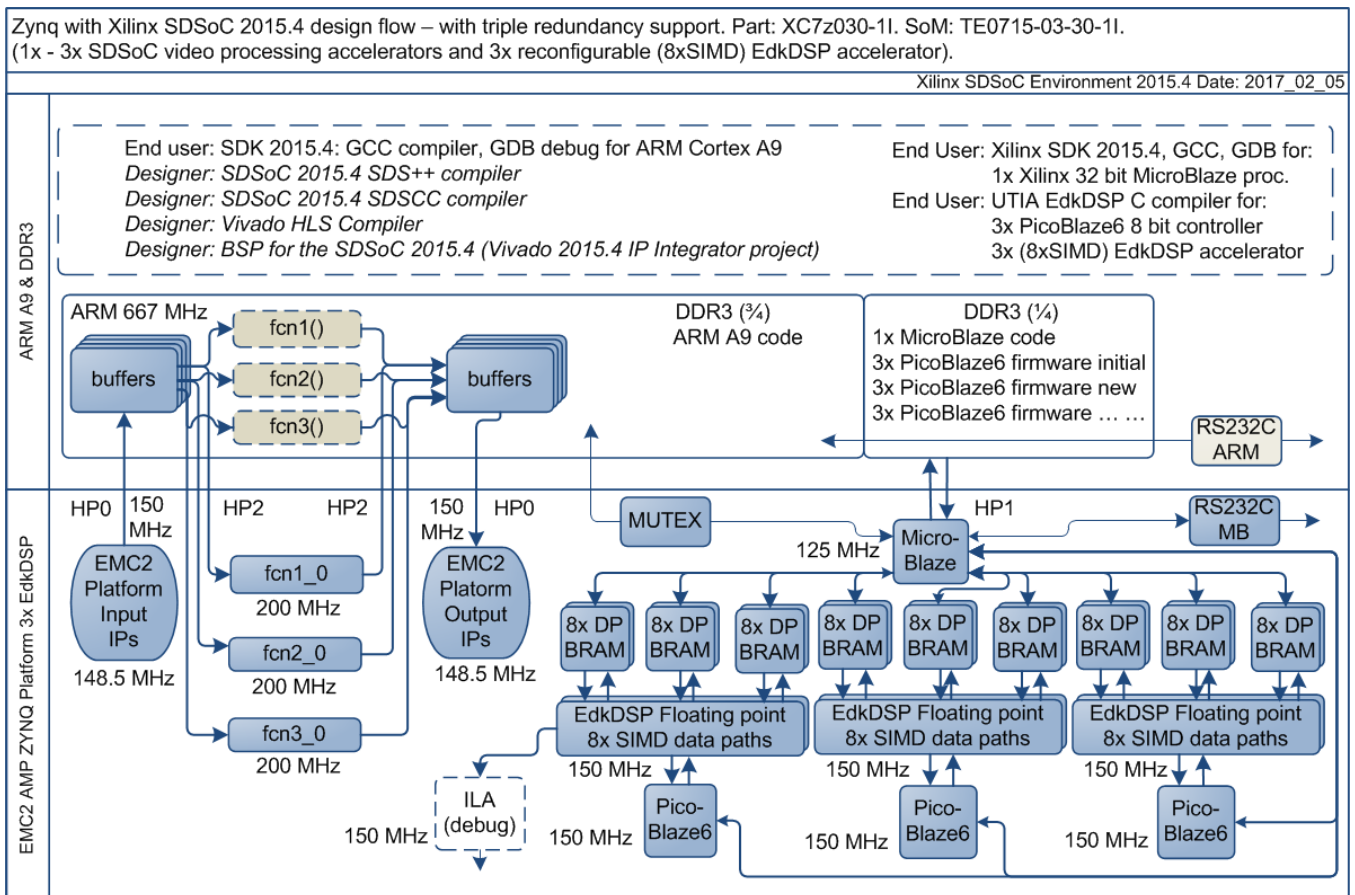


Figure 3: Full HD HDMII-HDMIO platform with three (8xSIMD) EdkDSP accelerators.

### 2.3 Configurations of Video Processing Accelerators and EdkDSP Accelerators:

We present the resources used and the accelerations reached for the final accelerator [11] in Appendix A. We have measured several configurations:

- The MicroBlaze with the three (8xSIMD) EdkDSP are present in HW. The MicroBlaze computes floating point FIR filter on one of the three EdkDSP accelerators and the video accelerator chain computes selected video processing algorithm in HW.
- The MicroBlaze with the three (8xSIMD) EdkDSP are present in HW. The MicroBlaze computes floating point LMS adaptive filter on one of the three EdkDSP accelerators and the video accelerator chain computes selected video processing algorithm in HW.
- The MicroBlaze with the three (8xSIMD) EdkDSP are present in HW. The MicroBlaze computes in SW (only with its internal HW floating point unit) the FIR or LMS filter in parallel to the dedicated video processing accelerator HW chain. None of the three (8xSIMD) EdkDSP accelerators is used.
- MicroBlaze is present in HW. It computes in SW (only with its internal HW floating point unit) the FIR or LMS filter in parallel to the dedicated video processing accelerator HW chain. The (8xSIMD) EdkDSP accelerators are not present in the PL logic.

Video processing algorithms have been implemented in SW by ARM Cortex A9 processor, first. The ARM C/C++ code was compiled with -O3 optimisation (but without use of the NEON accelerator) in the SDSoC 2015.4 environment [34].

The related HW resources include the MicroBlaze with the three (8xSIMD) EdkDSP present in the PL part of Zynq device and only the basic HDMI input and output HW support.

The evaluation designs with HW accelerators have been created from the selected C/C++ functions in SDSoC 2015.4 environment. New HW design have been generated and exported into the final set of SDK 2015.4 projects. The resulting demos are included in the evaluation package [11].

#### General setup of all demos of the final demonstrator [10]:

- ARM Cortex A9 processor of Xilinx Zynq device XC7Z030-1I [26] executes standalone C application programs performing initialisation and synchronisation of the HW accelerated video processing chains.
- Enclosed C programs for ARM, MicroBlaze and PicoBlaze6 sequencers can be modified by the user and recompiled in single Xilinx SDK 2015.4 development framework [33].
- Video signal input with resolution 1920x1080p60.
- Data are processed in HW into the YCrCb 4:2:2 (16 bit per pixel) format and stored by video DMA (VDMA) controller to input video frame buffers (VFBs) reserved in the DDR3.
- HW DMA controller(s) send data from the input VFBs to the processing HW accelerators in the programmable logic (PL) part of Zynq.
- Another HW DMA controller(s) send processed data from HW to output VFBs in DDR3.
- Second part of the HW VDMA writes data to the Full HD display with HDMI.

#### Main objectives of the demos in [10]:

- To demonstrate how to install, compile, modify and use the enclosed SW projects in the SDK 2015.4 [33].
- To demonstrate the HW accelerated video processing algorithms and the acceleration in comparison to the original ARM Cortex A9 SW versions of video processing algorithms.
- To demonstrate parallel execution of predefined video processing HW paths with C user code on ARM.

- To demonstrate HW accelerated video processing algorithms and the accelerated floating point FIR/LMS filters computed by the 8xSIMD EdkDSP run-time re-programmable floating point accelerator.
- To evaluate power consumption of several system configurations. See Annex A.

### Edge detection

The edge detection algorithm detects edges in each frame are marked as white and remaining part of the frame is set as black.

The edges are detected by a Sobel filter. Each pixel is filtered by a 3x3 2D FIR filter. A nonlinear decision on the output of the filter provides decision if the pixel is part of an edge or not. All computation is performed in fixed point. Input to the Sobel filter is the video signal with each pixel converted to the monochrome 8bit format.

Demos **sh01**, **sh02** and **sh03** provide accelerated HW computation of edge detection with 1, 2 or 3 parallel HW data paths. Computation of horizontal border line is resolved in case of sh02 and sh03. All these demos support synchronised parallel execution of user defined C code on ARM while the HW data paths perform accelerated video processing.

HW demos are using 1, 2 or 3 DMA HW channels as input from DDR3 to 1, 2 or 3 Sobel filters. Another 1, 2 or 3 DMA HW channels support output from Sobel filters to the DDR3. Demos are linked with static libraries libsh01.a, libsh02.a or libsh03.a.

### Motion detection

The motion detection algorithm detects and performs visualisation of moving edges. The moving edges are identified by two Sobel filters performing FIR filtering (similar to the above described edge detection) on pixels with identical coordinates but from two subsequent video frames. A difference of these filtered results is computed. This difference signal is finally filtered by the median filter.

Resulting signal is used for the nonlinear binary decision if the analysed pixel is part of a moving edge or not. If the pixel is part of a moving edge, it is assigned red colour and merged with the original colour video signal. Resulting output video signal is unchanged, with the exception of red colour marked moving edges.

Demo **md01** provides accelerated HW computation with one parallel HW data path. HW demo is using 2 DMA HW channels for reading from two subsequent video frame buffers located both in the DDR3 to the video processing chain of accelerators performing the motion detection. Another DMA HW channel performs parallel write of results to the DDR3. Demo is linked with static library libmd01.a.

### Measurements of acceleration and resources used for the final demonstrator [11]

The acceleration results have been measured as a ratio of the frame per second (FPS) reached by the accelerator and the FPS reached by the initial SW implementation on ARM in the SDSoC 2015.4 [34].

In case of SW implementation –O3 optimisation was used. HW support for the HDMI I/O data movement by the dedicated VDMA HW channels was used in all cases.

## Floating point performance

This section summarises measured sustained single precision floating point performance of the system in parallel to the accelerated video processing for the final demonstrator [11]:

**1411 MFLOP/s** on the 150 MHz (8xSIMD) EdkDSP (FIR filter in floating point on single 8xSIMD accelerator).

**12 MFLOP/s** on the 125 MHz MicroBlaze processor (with the MB single precision floating point unit in HW).

The controllers inside each (8xSIMD) EdkDSP accelerators are reprogrammed by the firmware compiled from C code with the use of the UTIA EDKDSP C compiler. Each accelerator can be programmed with two firmware programs. Designs can swap firmware in the runtime in only few clock cycles. The alternative firmware can be downloaded to the (8xSIMD) EdkDSP accelerator controllers in parallel with the execution of the current firmware.

This is demonstrated by swap of the firmware for the FIR filter (room response) to the firmware for adaptive LMS identification of the filter coefficients in the acoustic noise cancellation demo. This also demonstrates the mechanism and support for the move from one task to another task on the same accelerator.

Each of the three (8xSIMD) EdkDSP accelerators can deliver single-precision floating point results, which are bit-exact identical to the reference software implementation running on the MicroBlaze with the Xilinx HW single precision floating point unit.

Annex A contains measurement results for the final demonstrator [11].

## 2.4 Summary of parameters of the final demonstrator [11]

The 28nm Kintex-based programmable logic part of the Zynq XC7Z030-1I device [26] is capable of implementation in three UTIA (8xSIMD) EdkDSP floating point accelerators together with the Full HD video processing chain for the real-time video processing.

The combination of single 32bit MicroBlaze with three instances of the (8xSIMD) EdkDSP single precision floating point accelerators brings additional capability to compute floating point operations (single precision) with the performance 1411 MFLOP/s (in case of FIR filter) on single (8xSIMD) EdkDSP accelerator at the expense of relatively moderate increase of total power consumption of the system.

Instantiation of the 125 MHz MicroBlaze processor with three instances of the 150MHz (8xSIMD) EdkDSP accelerators enables to work with the triple redundancy and, in parallel, execute HW accelerated video processing algorithms. The optional in-circuit logic analyser (ILA) is capable of triggering and visualizing up to 32k of data samples at 150MHz clock rate for the first of the three (8xSIMD) EdkDSP accelerators. This is very useful for debugging of sequences vector operations and addresses generated by the sequencer of the EdkDSP accelerator.

Designs debugged and developed in the high level SDSoc 2015.4 environment [34] are exported for the end-user in form of SDK 2015.4 [33] projects. The released evaluation package with SDK 2015.4 projects provide sufficient freedom for the end-user to make certain SW adaptations and customisations of the final application without the need to understand app low level details of accelerator IP cores, of the Vivado 2015.4 project.

The initial SDSoC 2015.4 board support package is not needed in the released precompiled SDK 2015.4 projects. The SDSoC 2015.4 license is also not needed to run and modify them in the SDK 2015.4 SW project. See Annex A of this report for the detailed measurement results.

## 2.5 Conclusions

This chapter documents these general observations and conclusions related to the final demonstrator:

### Final demonstrator [11] with Zynq XC7Z030-1I device [26]

- The 28nm Kintex-based programmable logic part of the Zynq XC7Z030-1I device [26] is able to implement in parallel three UTIA (8xSIMD) EdkDSP floating point accelerators together with the Full HD video processing chain for the HDMI-HDMI real-time video processing.
- The combination of a single 32bit MicroBlaze with three instances of the (8xSIMD) EdkDSP floating point accelerator brings additional capability to compute in floating point (single precision) and triple redundancy with the performance 1411 MFLOP/s in case of FIR filter on single EdkDSP accelerator at the expense of relatively moderate increase of total power consumption of the system.
- Instantiation of MicroBlaze + three instances of the (8xSIMD) EdkDSP enables it to work with triple redundancy and it provides possibilities for design of video systems with the same number of parallel video processing chains as in the case of smaller XC7Z020-2I device [25].
- There are PL resources for the in-circuit logic analyser (ILA) capable of context sensitive triggering and monitoring of 32K data samples at 150 MHz clock rate of the EdkDSP accelerators. This is useful for the debug of the EdkDSP firmware.

Released application note and evaluation designs [11] document how designs debugged and developed in the high level SDSoC 2015.4 environment [33] can be exported to the end-user in form of SDK 2015.4 [32] projects.

The released evaluation package [11] with SDK 2015.4 projects provides space for the end-user to make high level SW adaptations and customisations of the final application without the need to understand the complete low level details of used IP cores, the Vivado 2015.4 project, and the SDSoC 2015.4 board support package.

## 2.6 Exploitation

### UTIA AV CR v.v.i.

UTIA exploitation plan strategy is based on the release of the evaluation package [11] with accompanying application note for the triple redundancy designs for the XC7Z030-1I part [26] with ILA options for debug.

This application note and the evaluation package have been released on UTIA public www server dedicated to the EMC2 project as the final demonstrator [11]:

<http://sp.utia.cz/index.php?ids=projects/emc2> [23]

in the subpage:

<http://sp.utia.cz/index.php?ids=results&id=s30i1hm4>

As the follow up, UTIA will offer commercial licensing for:

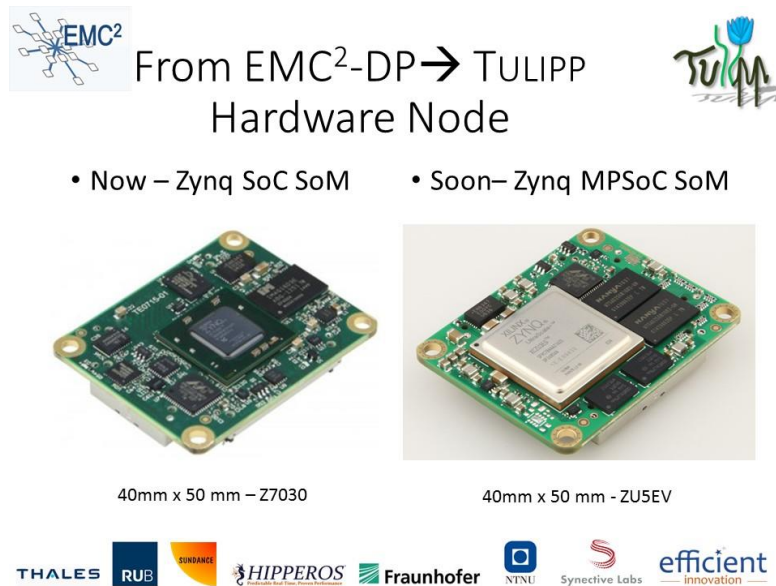
- BSP for the Linux target (for the Xilinx SDK 2015.4 [33]) running on the ARM with the triple redundancy based on 3x (8xSIMD) EdkDSP accelerators for the XC7Z030-1I part [26] with the ILA option for the firmware debug.
- BSP for the Xilinx Zynq XC7Z030-3E device [27] (for the Xilinx SDK 2015.4 [33]). This device offers additional performance gain due to the fast-grade device. (Clock of the ARM processor is up to 1GHz in case of [27] in comparison to the 667 MHz limit of [26]).

## Sundance Multiprocessor Technology Ltd.

Our joint efforts with UTIA have produced some valuable demonstrators for the EMC2 Partners that has chosen to use the EMC2-DP [22], [23], [24] for their own development. See press release [38].

It has enabled a new H2020 project – [www.tulipp.eu](http://www.tulipp.eu) – to gain a head-start in their goal of creating a “Low-Power Image Processing Platform” with the aim of creating a lasting impact.

The result has already, after 12 months, been an eco-system of users that are interested in the concept - <http://tulipp.eu/advisory-board-members/>



*Figure 4: Sundance exploitation path*

The EMC2-DP [22], [23], [24] will also become a mainstream platform for SMT’s range of COTS products. See: <http://www.sundance.technology/som-carriers/pc104-boards/>

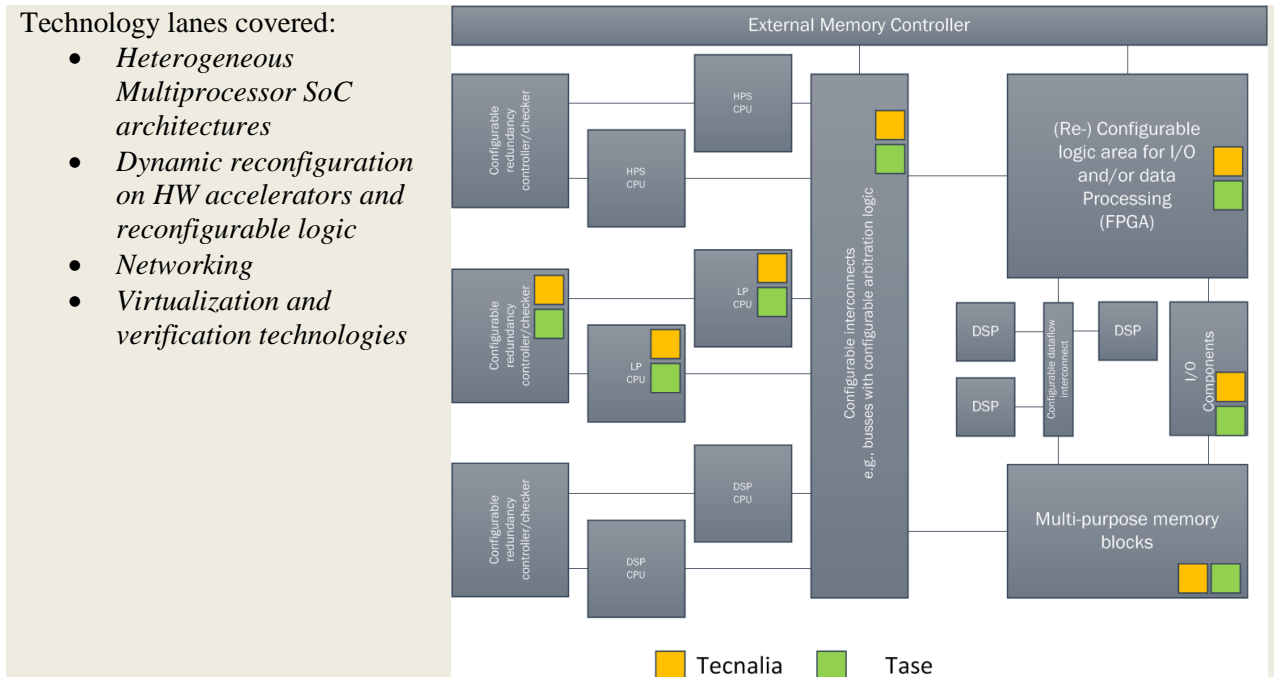
It will be enhanced during the summer with the latest generation of Zynq UltraScale+ MPSoC. See *Figure 4*.

## 2.7 Conditions for the access to the final UTIA demonstrator [11]

The UTIA demonstrator application note and the UTIA evaluation package [11] have several options for licensing. Details are described in Annex B of this D4.10 deliverable.



### 3. Reliable and Self-Healing Dynamic Reconfiguration Manager (DRM) on LADAP platform (Virtex-5)



A Reliable and Self-Healing Dynamic Reconfiguration Manager (DRM) is the solution proposed by TASE and TECNALIA in order to perform partial reconfiguration of a Virtex-5QV FPGA in a safe (reliable) way in the space environment, which is the main technical breakthrough pursued in WP9 - Use Case 1. This solution has been designed in the context of WP4, while its robustness will be evaluated in WP9. In deliverable D4.8 [3] the design, implementation and validation of a basic DRM was presented. This DRM consisted of a MicroBlaze embedded system, implemented in the static area of the Virtex-5 device, without fault-tolerance elements and capability to control the programming of two reconfigurable partitions. Validation of that design was done using the Xilinx ML506 Virtex-5 evaluation board. After that, the basic DRM design was migrated to the LADAP platform developed by TASE, and all the specified fault-tolerance elements have been included to make it reliable and self-healing. The final DRM hardware design and its implementation process were described in detailed in previous deliverable D4.9 [4].

Figure 5 shows the final architecture block diagram of the Reliable and Self-Healing DRM designed by TECNALIA and implemented in the Virtex-5 device of the LADAP platform. It is externally controlled and monitored by a LEON processor implemented by TASE in the RTAX device of the hardware platform.

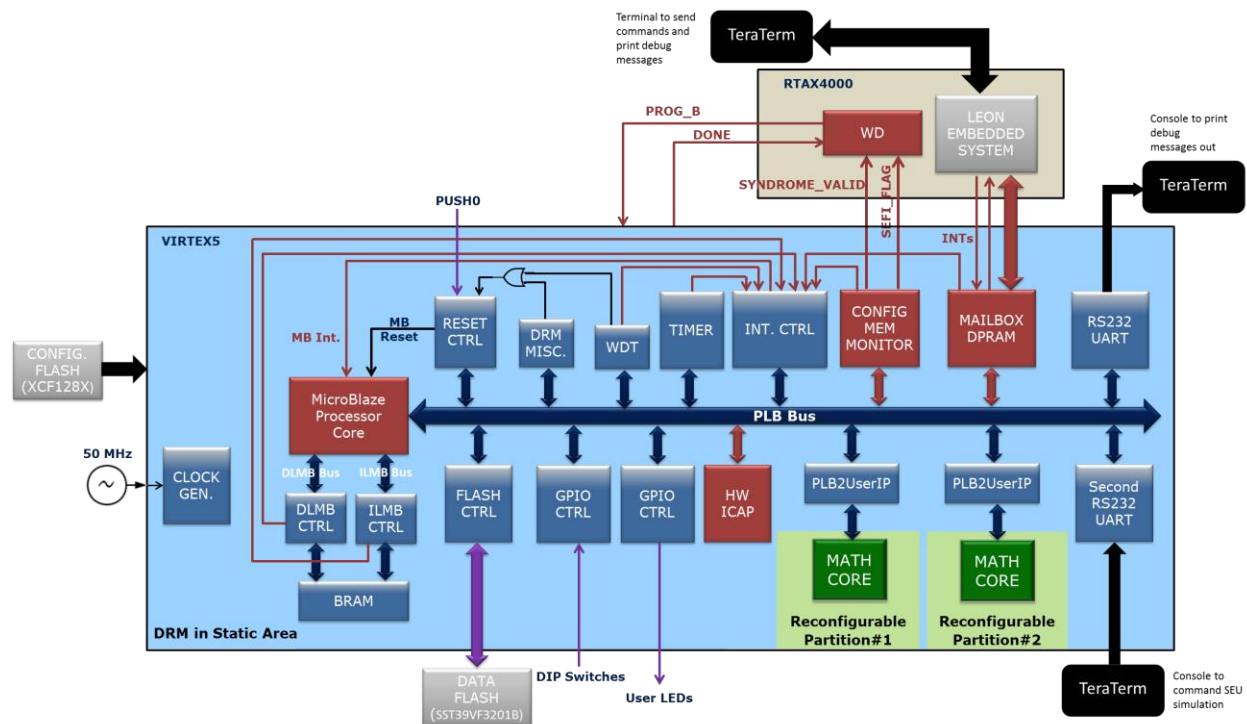


Figure 5: System architecture diagram of the Reliable and Self-Healing DRM

The rest of this section describes a final design update, required from the description presented in D4.9 [4], and the tests to validate the functionalities that make the DRM reliable and self-healing.

### 3.1 Update on final DRM design

In project month M30, when deliverable D4.9 was elaborated, the DRM implementation did not include the MicroBlaze softcore protection functionality. It has been included during the last project period (in M34). This functionality consists in enabling the MicroBlaze fault-tolerance features. In our case this turns into protection of the LMB Block RAM content with ECC (caches and MMU are not used in the DRM MicroBlaze configuration) and configuration of the LMB BRAM Interface Controller for the typical ECC use case. Additionally, MicroBlaze performs periodic scrubbing of the entire LMB BRAM using the *microblaze\_scrub()* function provided by Xilinx.

The DRM design without the MicroBlaze softcore protection functionality worked at 125 MHz system frequency. However, after enabling this functionality the system clock frequency must be decreased and set to 100 MHz, since the 125 MHz constraint cannot be met. This happens because when adding the ECC logic for the LMB memory the combinatorial path is much longer since no extra clock cycle has been added for LMB accesses. This fact is explained in the Xilinx Community Forum answer at <https://forums.xilinx.com/t5/Embedded-Processor-System-Design/MicroBlaze-Fault-Tolerance-Frequency-Limitations/m-p/268844/highlight/true#M6749>.

The next figure shows part of the PlanAhead timing report when trying to implement the DRM design at 125 MHz. As it can be seen, all the setup timing violations are related with the LMB BRAM memory.

Name	Slack	From	To	Total Delay	Logic Delay	Net %	Stages	Source Clock	Destination Clock
TS_clock_generator_0_clock_generator_0_SIG_PLL0_CLKOUT0 = PERIOD TIMEGRP									
Path 5	-5.817	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_16	13.745	4.406	67.9	9	clk_125_0000MHz	clk_125_0000MHz
Path 12	-5.699	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_16	13.721	4.588	66.6	11	clk_125_0000MHz	clk_125_0000MHz
Path 1	-5.840	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_17	13.710	4.406	67.9	9	clk_125_0000MHz	clk_125_0000MHz
Path 6	-5.775	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_16	13.699	4.406	67.8	9	clk_125_0000MHz	clk_125_0000MHz
Path 3	-5.831	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_17	13.697	4.406	67.8	9	clk_125_0000MHz	clk_125_0000MHz
Path 2	-5.836	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.686	4.588	66.5	11	clk_125_0000MHz	clk_125_0000MHz
Path 16	-5.657	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_16	13.675	4.588	66.4	11	clk_125_0000MHz	clk_125_0000MHz
Path 4	-5.827	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.673	4.588	66.4	11	clk_125_0000MHz	clk_125_0000MHz
Path 9	-5.754	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_0	13.583	4.406	67.6	9	clk_125_0000MHz	clk_125_0000MHz
Path 10	-5.716	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_19	13.582	4.406	67.6	9	clk_125_0000MHz	clk_125_0000MHz
Path 8	-5.755	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_0	13.573	4.406	67.5	9	clk_125_0000MHz	clk_125_0000MHz
Path 14	-5.697	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_0	13.559	4.588	66.2	11	clk_125_0000MHz	clk_125_0000MHz
Path 7	-5.768	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_19	13.558	4.588	66.2	11	clk_125_0000MHz	clk_125_0000MHz
Path 13	-5.698	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_0	13.549	4.588	66.1	11	clk_125_0000MHz	clk_125_0000MHz
Path 15	-5.659	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_19	13.534	4.406	67.4	9	clk_125_0000MHz	clk_125_0000MHz
Path 11	-5.711	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_19	13.510	4.588	66.0	11	clk_125_0000MHz	clk_125_0000MHz
Path 17	-5.639	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.489	4.567	66.1	11	clk_125_0000MHz	clk_125_0000MHz
Path 18	-5.633	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.483	4.564	66.1	11	clk_125_0000MHz	clk_125_0000MHz
Path 19	-5.630	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.476	4.567	66.1	11	clk_125_0000MHz	clk_125_0000MHz
Path 20	-5.624	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.470	4.564	66.1	11	clk_125_0000MHz	clk_125_0000MHz
Path 21	-5.610	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.465	4.588	65.9	11	clk_125_0000MHz	clk_125_0000MHz
Path 22	-5.610	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.465	4.585	65.9	11	clk_125_0000MHz	clk_125_0000MHz
Path 25	-5.590	lmb_bram/lmb_bram/ramb36_4	lmb_bram/lmb_bram/ramb36_17	13.456	4.585	65.9	11	clk_125_0000MHz	clk_125_0000MHz
Path 23	-5.601	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.452	4.585	65.9	11	clk_125_0000MHz	clk_125_0000MHz
Path 24	-5.601	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_17	13.452	4.588	65.9	11	clk_125_0000MHz	clk_125_0000MHz
Path 27	-5.581	lmb_bram/lmb_bram/ramb36_4	lmb_bram/lmb_bram/ramb36_17	13.443	4.585	65.9	11	clk_125_0000MHz	clk_125_0000MHz
Path 26	-5.581	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_18	13.382	4.588	65.7	11	clk_125_0000MHz	clk_125_0000MHz
Path 29	-5.571	lmb_bram/lmb_bram/ramb36_2	lmb_bram/lmb_bram/ramb36_19	13.361	4.567	65.8	11	clk_125_0000MHz	clk_125_0000MHz
Path 28	-5.571	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_13	13.152	4.406	66.5	9	clk_125_0000MHz	clk_125_0000MHz
Path 30	-5.566	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_13	13.151	4.406	66.5	9	clk_125_0000MHz	clk_125_0000MHz
TS_clock_generator_0_clock_generator_0_SIG_PLL0_CLKOUT1 = PERIOD TIMEGRP									

Figure 6: Timing violations when trying to implement the DRM design at 125 MHz

The next figure shows that implementation meets the 100 MHz constraint and that the most critical paths are still related with the LMB BRAM memory.

Name	Slack	From	To	Total Delay	Logic Delay	Net %	Stages	Source Clock	Destination Clock
TS_clock_generator_0_clock_generator_0_SIG_PLL0_CLKOUT0 = PERIOD TIMEGRP									
Path 2	0.064	lmb_bram/lmb_bram/ramb36_12	microblaze_0/microbl...ux[11].Gen_Instr_DFF	9.495	3.557	62.5	13	clk_100_0000MHz	clk_100_0000MHz
Path 3	0.076	lmb_bram/lmb_bram/ramb36_13	microblaze_0/microbl...ux[17].Gen_Instr_DFF	9.416	3.415	63.7	11	clk_100_0000MHz	clk_100_0000MHz
Path 4	0.076	lmb_bram/lmb_bram/ramb36_13	microblaze_0/microbl...ux[17].Gen_Instr_DFF	9.416	3.412	63.8	11	clk_100_0000MHz	clk_100_0000MHz
Path 5	0.096	lmb_bram/lmb_bram/ramb36_13	microblaze_0/microbl...ux[16].Gen_Instr_DFF	9.373	3.415	63.6	11	clk_100_0000MHz	clk_100_0000MHz
Path 6	0.096	lmb_bram/lmb_bram/ramb36_13	microblaze_0/microbl...ux[16].Gen_Instr_DFF	9.373	3.412	63.6	11	clk_100_0000MHz	clk_100_0000MHz
Path 7	0.109	lmb_bram/lmb_bram/ramb36_18	microblaze_0/microbl...ux[11].Gen_Instr_DFF	9.433	3.403	63.9	11	clk_100_0000MHz	clk_100_0000MHz
Path 8	0.117	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_5	9.836	4.105	58.3	11	clk_100_0000MHz	clk_100_0000MHz
Path 9	0.122	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_5	9.836	4.105	58.3	11	clk_100_0000MHz	clk_100_0000MHz
Path 10	0.125	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_5	9.828	4.102	58.3	11	clk_100_0000MHz	clk_100_0000MHz
Path 11	0.128	lmb_bram/lmb_bram/ramb36_12	microblaze_0/microbl...ux[35].Gen_Instr_DFF	9.431	3.597	61.9	13	clk_100_0000MHz	clk_100_0000MHz
Path 12	0.129	lmb_bram/lmb_bram/ramb36_12	microblaze_0/microbl...ux[35].Gen_Instr_DFF	9.430	3.595	61.9	13	clk_100_0000MHz	clk_100_0000MHz
Path 13	0.130	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_5	9.828	4.102	58.3	11	clk_100_0000MHz	clk_100_0000MHz
Path 14	0.142	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_10	9.742	4.105	57.9	11	clk_100_0000MHz	clk_100_0000MHz
Path 15	0.147	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_10	9.742	4.105	57.9	11	clk_100_0000MHz	clk_100_0000MHz
Path 16	0.150	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_10	9.734	4.102	57.9	11	clk_100_0000MHz	clk_100_0000MHz
Path 17	0.155	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_10	9.734	4.102	57.9	11	clk_100_0000MHz	clk_100_0000MHz
Path 18	0.167	lmb_bram/lmb_bram/ramb36_10	lmb_bram/lmb_bram/ramb36_5	9.786	4.105	58.1	11	clk_100_0000MHz	clk_100_0000MHz
Path 19	0.168	lmb_bram/lmb_bram/ramb36_10	lmb_bram/lmb_bram/ramb36_5	9.785	4.102	58.1	11	clk_100_0000MHz	clk_100_0000MHz
Path 20	0.172	lmb_bram/lmb_bram/ramb36_10	lmb_bram/lmb_bram/ramb36_5	9.786	4.105	58.1	11	clk_100_0000MHz	clk_100_0000MHz
Path 21	0.173	lmb_bram/lmb_bram/ramb36_18	microblaze_0/microbl...ux[35].Gen_Instr_DFF	9.369	3.443	63.3	11	clk_100_0000MHz	clk_100_0000MHz
Path 22	0.173	lmb_bram/lmb_bram/ramb36_10	lmb_bram/lmb_bram/ramb36_5	9.785	4.102	58.1	11	clk_100_0000MHz	clk_100_0000MHz
Path 23	0.175	lmb_bram/lmb_bram/ramb36_13	microblaze_0/microbl...ux[38].Gen_Instr_DFF	9.344	3.434	63.2	11	clk_100_0000MHz	clk_100_0000MHz
Path 24	0.175	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_2	9.726	4.105	57.8	11	clk_100_0000MHz	clk_100_0000MHz
Path 25	0.175	lmb_bram/lmb_bram/ramb36_13	microblaze_0/microbl...ux[38].Gen_Instr_DFF	9.344	3.431	63.3	11	clk_100_0000MHz	clk_100_0000MHz
Path 26	0.178	lmb_bram/lmb_bram/ramb36_1	microblaze_0/microbl...ux[16].Gen_Instr_DFF	9.430	3.211	65.9	11	clk_100_0000MHz	clk_100_0000MHz
Path 27	0.179	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_2	9.726	4.105	57.8	11	clk_100_0000MHz	clk_100_0000MHz
Path 28	0.183	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_2	9.718	4.102	57.8	11	clk_100_0000MHz	clk_100_0000MHz
Path 29	0.187	lmb_bram/lmb_bram/ramb36_9	lmb_bram/lmb_bram/ramb36_2	9.718	4.102	57.8	11	clk_100_0000MHz	clk_100_0000MHz
Path 30	0.189	lmb_bram/lmb_bram/ramb36_10	lmb_bram/lmb_bram/ramb36_5	9.764	4.084	58.2	11	clk_100_0000MHz	clk_100_0000MHz

Figure 7: DRM implementation at 100 MHz – most critical paths timing

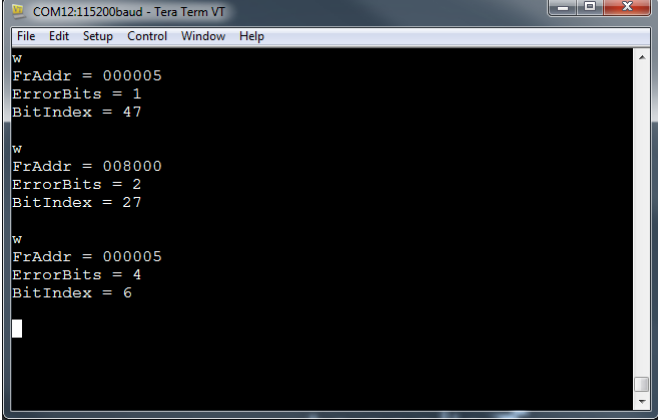
Another minor update is related to the DRM system memory map shown in D9.5 [9]. The address space associated to the LMB BRAM interface controllers has been increased from 256 bytes to 1 Kbyte, that is, from 0x8B000000 to 0x8B0003FF for the DLMB BRAM interface controller, and from 0x8B100000 to 0x8B1003FF for the ILMB BRAM interface controller.

### 3.2 DRM design functional validation

By project month M30, the DRM validation tests done were those related with the communication with the external LEON processor and those related with partial reconfiguration. During the last project period tests have been done to validate the rest of functionalities, mainly the features that make the DRM reliable and self-healing.

In order to check that FPGA configuration memory errors are detected and corrected, the functionality to insert errors in the configuration memory has been developed. This feature simulates the effect of SEUs and works as follows:

MicroBlaze listens to the secondary UART and a terminal application such as Tera Term is connected to it. If MicroBlaze receives the **w** command, it displays the "FrAddr = " prompt and a valid 24-bit frame address as a 6-digit hexadecimal number must be provided. Then, MicroBlaze displays the "ErrorBits = " prompt and the number of bits that must be toggled has to be provided. Up to 4 bit errors per frame can be inserted. Finally, MicroBlaze displays the "BitIndex = " prompt, and a decimal number from 0 to 1,311, representing the first bit in the frame to be toggled, must be provided. If more than one bit error has to be inserted, the adjacent bits will be toggled. The next figure shows the **w** SEU simulation command insertion:



```

COM12:115200baud - Tera Term VT
File Edit Setup Control Window Help
w
FrAddr = 000005
ErrorBits = 1
BitIndex = 47

w
FrAddr = 008000
ErrorBits = 2
BitIndex = 27

w
FrAddr = 000005
ErrorBits = 4
BitIndex = 6

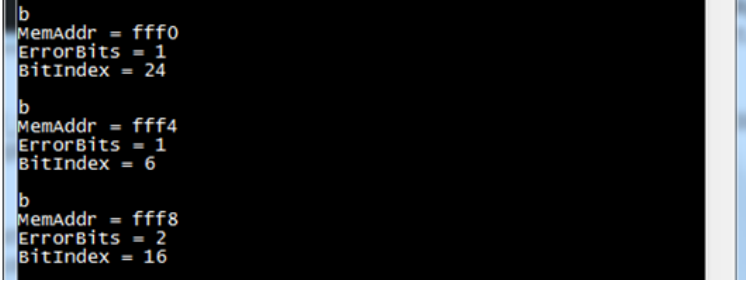
```

Figure 8: **w** SEU simulation command

After receiving this command, MicroBlaze reads the specified configuration frame, toggles the specified number of bits and writes the frame back through the HWICAP core.

In order to test that errors in the MicroBlaze LMB BRAM memory are detected and corrected, the functionality to insert errors in the LMB BRAM memory has been developed. It works in a similar way to the insertion of errors in the FPGA configuration memory:

MicroBlaze listens to the secondary UART. If it receives the **b** command, MicroBlaze displays the "MemAddr = " prompt and a valid 16-bit frame address as a 4-digit hexadecimal number must be provided. Then, MicroBlaze displays the "ErrorBits = " prompt and the number of bits that must be toggled has to be provided. Up to 2 bit errors can be inserted in a BRAM word. Then, MicroBlaze displays the "BitIndex = " prompt, and a decimal number from 0 to 39, representing the first bit in the memory word (32-bit data + 8 ECC bits) to be toggled, must be provided. If more than one bit error has to be inserted, the adjacent bits are toggled. The Fault Injection Data and ECC registers of the LMB BRAM Interface Controller are used to insert the errors. The next figure shows the **b** SEU simulation command insertion:



```

b
MemAddr = fff0
ErrorBits = 1
BitIndex = 24

b
MemAddr = fff4
ErrorBits = 1
BitIndex = 6

b
MemAddr = fff8
ErrorBits = 2
BitIndex = 16

```

Figure 9: **b** SEU simulation command

The validation tests and their results are described next:

1. Initialization of the data flash with the block type 000 configuration frames of the initial full FPGA design, when the version of the image stored in the data flash is not equal to the version of the design loaded in the FPGA.

MicroBlazes reads these configuration frames through the HWICAP core and writes them into the data flash to be used later when full device scrubbing or scrubbing of a frame with two bit errors must be done.

It has to be noticed that after reading the block type 000 configuration frames through the ICAP port, the readback CRC logic didn't re-start although a DESYNC command was issued. This problem has been solved by sending the DESYNC command twice.

2. Configuration memory errors detection by the Configuration Memory Monitor peripheral. The validated features are:

- Detection of frame with one bit error
- Detection of frame with two bit errors
- Detection of global CRC error
- Generation of frame address and frame number
- Translation of bit index in the case of frame with one bit error
- Configuration Memory Monitor interrupt assertion and detection by MicroBlaze

The next figures are snapshots of the Xilinx ChipScope Pro Analyzer tool that show the Configuration Memory Monitor peripheral detecting errors in the configuration memory and issuing to MicroBlaze the corresponding scrub action. The three possible cases have been captured: (1) one correctable bit error in a frame; (2) two bit errors detected in one frame; and (3) 4 bit errors in one frame; the ECCERROR signal is not asserted, only the CRCERROR signal is asserted.

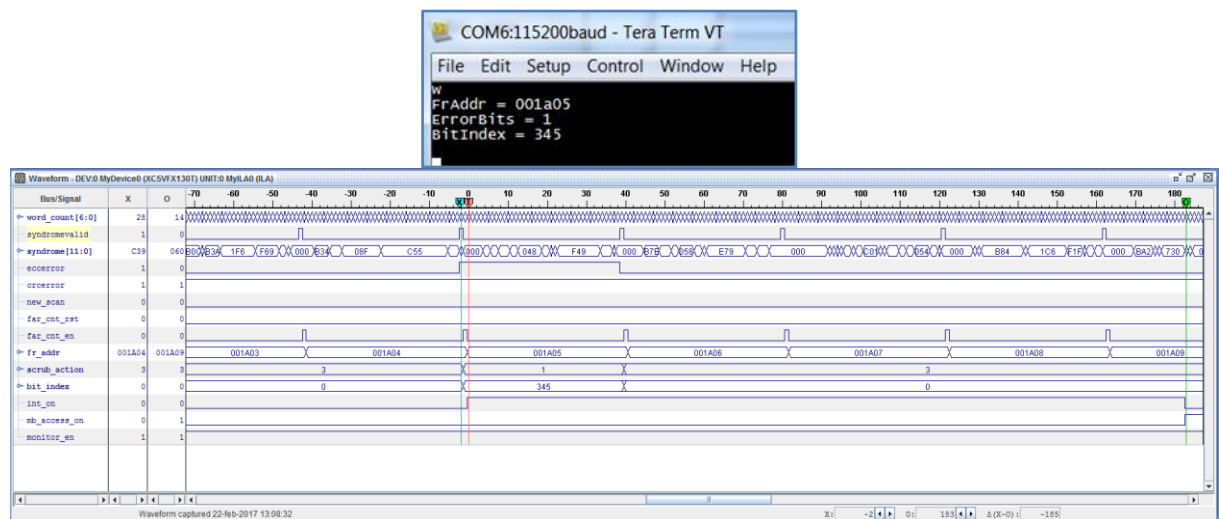


Figure 10: Configuration Memory Monitor detects frame with one correctable bit error

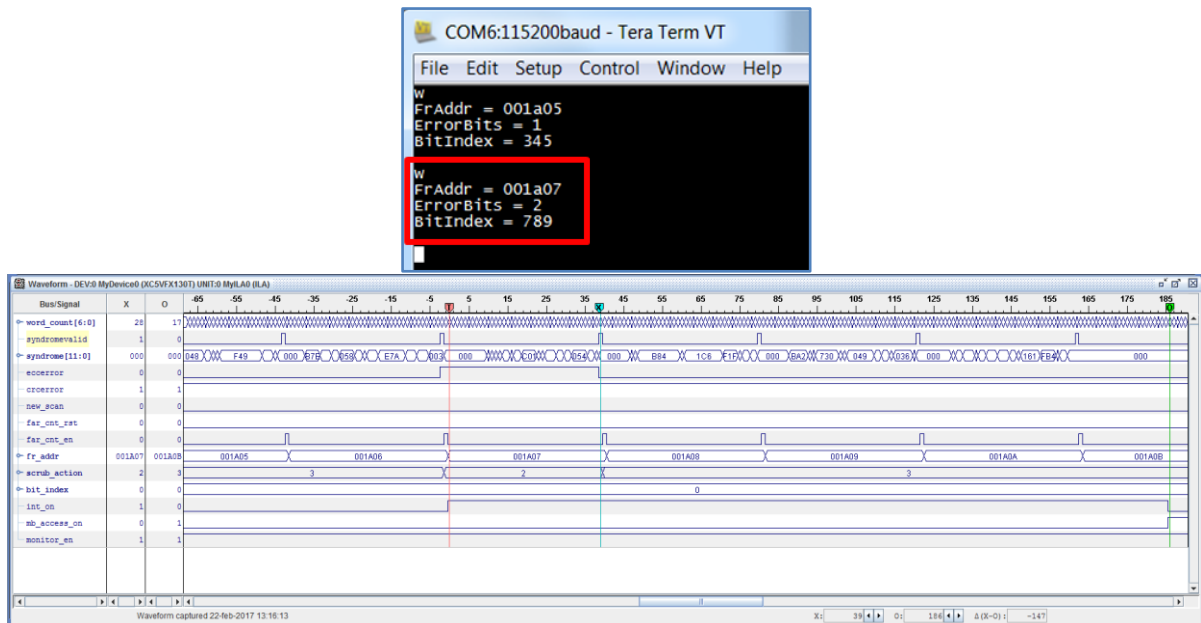


Figure 11: Configuration Memory Monitor detects frame with two bit errors

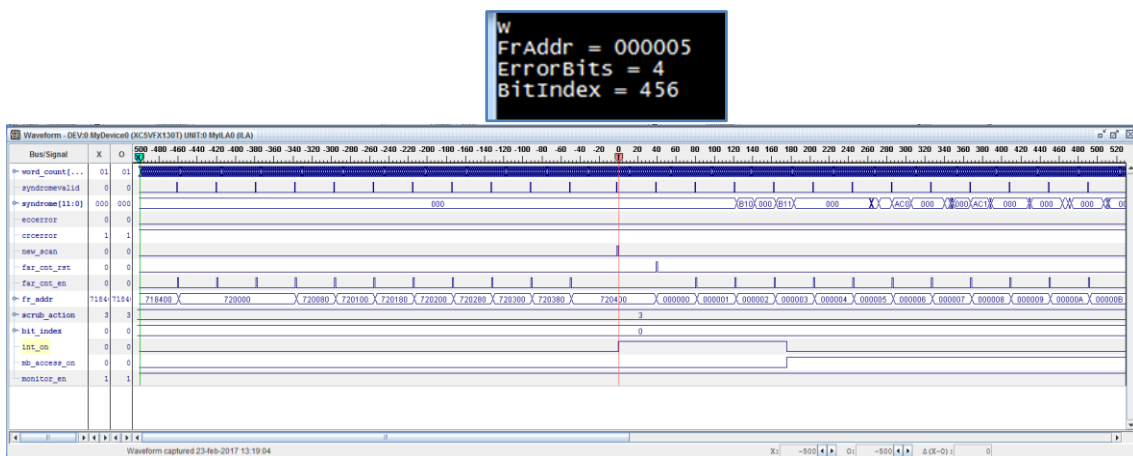


Figure 12: Configuration Memory Monitor detects global CRC error

3. Scrubbing actions performed by MicroBlaze. Validated features are:

- Scrubbing configuration frame with one bit error. MicroBlaze reads the frame through the ICAP port, toggles the error bit and writes the corrected frame back.
- Scrubbing configuration frame with two bit errors. Since the error positions are unknown, MicroBlaze gets the good configuration data from the external data flash and writes it into the FPGA configuration memory through the ICAP port.
- Full device scrubbing. As in the previous case, good configuration data is read from the data flash.
- The three types of scrubbing tests are done injecting errors in frames belonging to each reconfigurable partition and to the static area.
- Full device scrubbing has been tested when different images are loaded into the configurable partitions. The action is validated with the Perform Operation command checking the result of the arithmetical operation performed by the math core in each reconfigurable partition.

4. Frame Address Register (FAR) checking and assertion of SEFI\_FLAG output when checking fails. This has been tested because in the final design implementation injecting one bit error at bit index 235 or 236 in the configuration frame with address 0x001A05 makes FAR checking fail. After error injection, the Configuration Memory Monitor warns MicroBlaze to scrub the frame indicating the error bit position. *Figure 13* shows the code executed by MicroBlaze. As it can be seen, MicroBlaze checks the FAR register before any scrubbing action.

```

/***** Main Loop *****/
while (1){
  if (scrub_cmd_req != 0){
    // Set MbAccessOn bit to 1
    Data = CONFIG_MEM_MONITOR_mReadReg(XPAR_CONFIG_MEM_MONITOR_0_BASEADDR, CONFIG_MEM_MONITOR_SLV_REG1_OFFSET);
    Data = Data | 0x00000001;
    CONFIG_MEM_MONITOR_mWriteReg(XPAR_CONFIG_MEM_MONITOR_0_BASEADDR, CONFIG_MEM_MONITOR_SLV_REG1_OFFSET, Data);

    // Check FAR register
    Status = FARCheck(&HwIcap);

    // If FAR checking ends successfully, perform scrub action
    if (Status == XST_SUCCESS) {
      Status = FakeSEFICheck(&HwIcap);
      Status = ScrubCfgMem(&HwIcap, XPAR_GENERIC_EXTERNAL_MEMORY_MEM0_BASEADDR, scrub_cmd_req, scrub_addr, scrub_fr_num, scrub_bit_index);
      switch(scrub_cmd_req){
        case SCRUB_ONE_BIT_ERROR_FR:
          ScrubAction1Cnt++;
          break;
        case SCRUB_TWO_BIT_ERROR_FR:
          ScrubAction2Cnt++;
          break;
        case SCRUB_FULL_DEVICE:
          ScrubAction3Cnt++;
          break;
      }
    }
    else{ // Set SEFI flag on
      Data = Data | 0x00000002;
      CONFIG_MEM_MONITOR_mWriteReg(XPAR_CONFIG_MEM_MONITOR_0_BASEADDR, CONFIG_MEM_MONITOR_SLV_REG1_OFFSET, Data);
    }

    // Set MbAccessOn bit to 0
    Data = Data & 0xFFFFFFFE;
    CONFIG_MEM_MONITOR_mWriteReg(XPAR_CONFIG_MEM_MONITOR_0_BASEADDR, CONFIG_MEM_MONITOR_SLV_REG1_OFFSET, Data);

    // Clear scrub command request flag
    scrub_cmd_req = 0;
  }
}

```

*Figure 13: Scrubbing code at main loop level*

When the FARCheck function (see *Figure 14*) is executed step by step in debug mode, the first time a mismatch is detected between the value written in the FAR register (0x001AE500) and the read value (0x00000000). Therefore, the SEFI\_FLAG output is asserted and the scrub action is not done. If the PROG\_B signal is disconnected from the RTAX FPGA, the Virtex-5 is not reconfigured, and, since the error in the configuration memory has not been corrected, the Configuration Memory Monitor warns MicroBlaze again to scrub the same frame. This second time, when the FARCheck function is executed, the access to read the FAR register through the ICAP port fails. While the PROG\_B signal is disconnected from the RTAX FPGA, the process is repeated. From the third time on, any access through the ICAP port fails.

```

int FARCheck (XHWIcap *HWIcapInstancePtr) {
int Status;
u32 RegData;

#ifdef PRINT_ANSWER
    xil_printf ("FAR checking... ");
#endif

    // Write FAR checking constant into FAR register
    Status = XHWIcap_WriteConfigReg(HWIcapInstancePtr, XHI_FAR, FAR_CHECK_VALUE);
    if (Status != XST_SUCCESS){
        return XST_FAILURE;
    }

    // Read FAR register
    Status = XHWIcap_ReadConfigReg(HWIcapInstancePtr, XHI_FAR, &RegData);
    if (Status != XST_SUCCESS){
        return XST_FAILURE;
    }

    // Send DESYNC command
    Status = XHWIcap_CommandDesync(HWIcapInstancePtr);
    if (Status != XST_SUCCESS) {
        return XST_FAILURE;
    }

    // Check read FAR value
    if (RegData == FAR_CHECK_VALUE){
#ifdef PRINT_ANSWER
        xil_printf ("FAR check success \r\n");
#endif
        return XST_SUCCESS;
    }else{
#ifdef PRINT_ANSWER
        xil_printf ("FAR check failed \r\n");
#endif
        return XST_FAILURE;
    }
}

```

Figure 14: FAR checking routine

To recover the DRM from this situation, the PROG\_B signal has to be connected to the watchdog implemented in the RTAX. When this watchdog detects the SEFI\_FLAG signal high, it asserts the PROG\_B signal low for at least 250 ns and the Virtex-5 is reconfigured.



Figure 15: SEFI\_FLAG assertion after unsuccessful FAR checking

Note: regarding fake SEFI detection and ICAP START command generation, this feature is implemented, but it cannot be tested. To validate the implementation we need to force the DONE bit of the STAT register to 0, but this is a read only register

5. PLB Timer configuration. Validated features are:
  - PLB Timer interrupt assertion and detection by MicroBlaze
  - Timer period

Validation has been done by means of messages printed out on the Tera Term console.



6. DRM internal WatchDog Timer interrupt assertion and detection by MicroBlaze. Validation has been done by means of messages printed out on the Tera Term console.
7. SEU statistic counters validation. These counters are:
  - a) Single-Error Configuration Memory Frames counter: number of corrupted (and corrected) frames with only one error bit found during readback.
  - b) Double-Error Configuration Memory Frames counter: number of corrupted (and corrected) frames with two error bits found during readback.
  - c) Full Scrubbing Processes counter: total number of full scrubbing processes required because of unidentified uncorrectable bit errors.
  - d) Periodical Full Device Scrubbing Processes counter: number of full device scrubbing processes performed when periodical blind scrubbing is enabled.
  - e) DP1SBitErrCnt: number of single bit errors detected when reading from the mailbox DPRAM where LEON stores the command requests.
  - f) DP1DBitErrCnt: number of double bit errors detected when reading from the mailbox DPRAM where LEON stores the command requests.
  - g) DP2SBitErrCnt: number of single bit errors detected when reading from the mailbox DPRAM where DRM writes the command answers.
  - h) DP2DBitErrCnt: number of double bit errors detected when reading from the mailbox DPRAM where DRM writes the command answers.
  - i) MbBramCECnt: number of correctable single bit errors detected and corrected in the MicroBlaze LMB BRAM memory.
  - j) MbBramUECnt: number of uncorrectable errors detected in the MicroBlaze LMB BRAM memory.

It has been checked that these counters are increased accordingly to the number of errors injected in the FPGA configuration memory (counters a, b and c), to the configured time period for periodical full device scrubbing (counter d) and to the number of errors injected in the LMB BRAM memory (counters i and j). These counters are cleared after sending the answer to a “Retrieve SEU Statistics” command request.

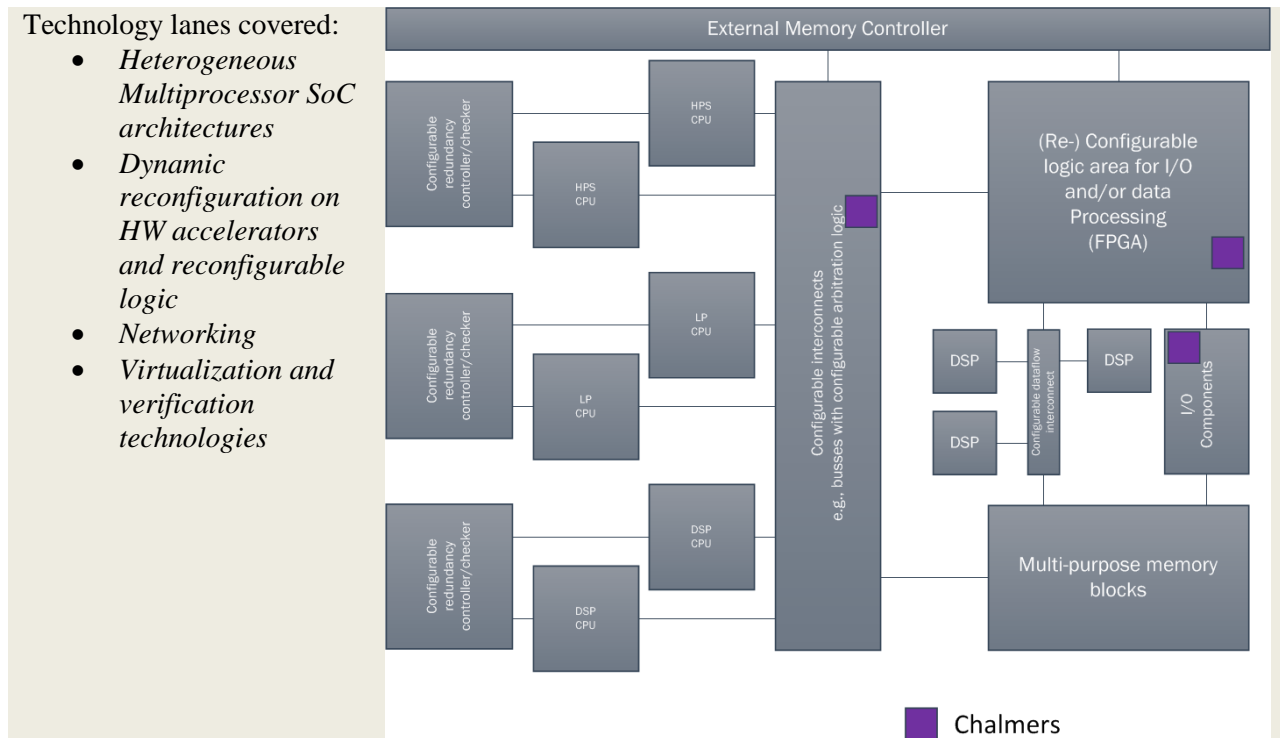
Counters associated to errors in the mailbox DPRAM (e, f, g and h) have been validated by simulation. Practical validation has not been possible since the capability to inject errors in the BRAM blocks of the DPRAM is not available for Virtex-5 devices.

8. DRM Reset commanded by LEON by means of the Configure SEU Mitigation command has been successfully tested.
9. Validation of the MicroBlaze softcore protection features. This includes:
  - ILMB/DLMB BRAM Interface Controller interrupt assertion and detection by MicroBlaze
  - BRAM memory address correction when one bit error is injected
  - Statistic counters
  - Periodic LMB BRAM scrubbing

It has been tested that correctable bit errors in the MicroBlaze LMB BRAM memory are corrected. While uncorrectable errors do not have any impact on the DRM application execution, the DRM keeps working as expected. When uncorrectable errors negatively impact on the DRM application execution, the DRM stops answering to the command requests from LEON. This is an indication for the LEON to force Virtex-5 reconfiguration.

After completing these tests, the DRM reliability and self-healing functionalities are successfully validated. Evaluation of the results from the point of view of their robustness and innovation has been done in WP9 and the conclusions are presented in deliverable D9.6.

#### 4. Resilient Reconfigurable Multiprocessor arrays: probabilistic analysis of availability and efficiency



In T4.4, Chalmers performed a probabilistic analysis to evaluate the availability of the reconfigurable multicore architecture designed in T4.2. As reported in D4.5, in T4.2 Chalmers designed and implemented a reconfigurable processor array, which is able to provide coarse-grain (CG) and fine-grain (FG) reconfigurability to mitigate permanent faults as depicted in the generic *Figure 16*. Coarse-grain reconfigurability allows replacement of faulty processor parts by identical (presumably spare parts) borrowed by a neighboring processor. To exemplify, a pipeline stage of a processor in *Figure 16* (A, B, C, or D) could be replaced when damaged by the same, identical stage of the neighboring processor using the switches and vertical wires that connect the two processors. Fine-grain reconfigurability offers further resources to instantiate more processor parts to replace faulty processor components, acting as a wild card.

In D4.7, we (Chalmers) described a preliminary probabilistic analysis that evaluates the benefits of coarse grain reconfigurability. In periods 2 and 3 of the project, we extended this analysis to include the fine-grain reconfigurability (besides the coarse grain considered in D4.7). The outcome of this analysis was used to evaluate our architecture as reported in D4.5 (Section 3.10) in terms of reliability, performance and energy efficiency.

This probabilistic analysis resulted in the following publication [37]:

*"A Probabilistic Analysis of Resilient Reconfigurable Designs", Alirad Malek, Stavros Tzilis, Danish Anis Khan, Ioannis Sourdis, Georgios Smaragdos and Christos Strydis, in International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2015.*

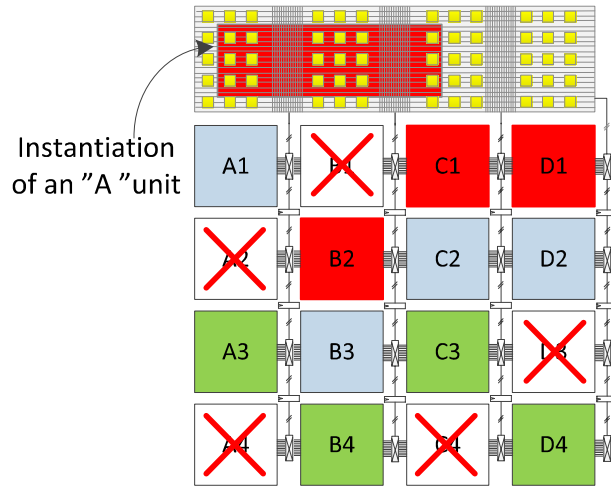


Figure 16: Example of CG and FG reconfiguration constructing 3 fault-free components: component-1 (A{FG}, B2, C1, D1), component-2 (A1, B3, C2, D2), and component-3 (A3, B4, C3, D4).

#### 4.1 Probabilistic Analysis of a mixed-grained reconfigurable array

Considering the generic design described in Figure 16, we next present an overview of our probabilistic analysis developed for assessing the fault-tolerance of our architecture. In particular, we derive the probability of having exactly  $M$  fault-free components out of  $N$  ( $P_{ff}(N, M)$ ) as well as the probability of having at least  $M$  fault-free components out of  $N$  ( $P_{\geq}(N, M)$ ) for different numbers of faults in area  $A$  for two different design approaches: (i) coarse-grain reconfigurability and (ii) a mix of coarse- and fine-grain reconfigurability which offers the option of a faulty Substitutable Unit (SU) to be replaced either by an identical spare unit (coarse-grain replacement) or by fine-grain logic.

Assuming that an area  $A$  is divided into  $N$  SUs of equal size and that there are  $k$  faults in total in the area with a uniform random distribution, then the probability of a SU to have exactly  $i$  out of the  $k$  faults is:

$$P_{f=i}(k, N) = \binom{k}{i} \times \left(\frac{1}{N}\right)^i \times \left(1 - \frac{1}{N}\right)^{k-i} \quad (1)$$

Then, the probability of a SU to be faulty,  $P_{sf}(k, N)$ , is equal to the sum of probabilities of having 1, 2, ...,  $k$  faults:

$$P_{sf}(k, N) = \sum_{i=1}^k P_{f=i}(k, N) \quad (2)$$

Considering that the  $P_{sf}$ , for a specific number of faults  $k$  in the area, of a single SU is known based on the equation 2, it is possible to calculate the probability of having exactly  $M$  fault-free SUs out of  $N$ , denoted as  $P_{ff}(N, M)$ :

$$P_{ff}(N, M) = \binom{N}{M} \times P_{sf}^{N-M} \times (1 - P_{sf})^M \quad (3)$$

It is also possible to expand this formula to get the probability of having at least  $M$  non-faulty SUs out of  $N$  for a specific  $P_{sf}$  here indicated as  $P_{\geq}$ :

$$P_{\geq}(N, M) = \sum_{i=M}^N P_{ff}(N, i) \quad (4)$$

which is the sum of probabilities of having  $i = \{M, \dots, N\}$  fault-free SUs.

In a design with coarse-grain reconfigurability, each component is divided into a specific number of SUs. Considering the example illustrated in *Figure 16* which depicts four components each of which are divided into four SUs, it can be observed that for having a fault-free (working) component, there should be at least one fault-free SU at each column available. Therefore, the number of fault-free components that can be constructed is defined by the minimum number of fault-free SUs in each column. For calculating the probability of having a specific number of fault-free components, we first calculate the probability of having at least  $M$  fault-free SUs out of  $N$  in one column, which is derived by equation 4:

$$P_{\geq}^{col}(N, M) = P_{\geq}(N, M) \quad (5)$$

Expanding this formula over the total number of columns will result in the probability of having at least  $M$  SUs in each column, which is equal to the probability of having at least  $M$  fault-free components available in a coarse-grain (cg) reconfigurable design:

$$P_{\geq}^{cg}(N, M) = (P_{\geq}^{col}(N, M))^c \quad (6)$$

The exponent  $c$  is the total number of columns, which also defines the coarse-grain granularity. In order to find the probability of having exactly  $M$  fault-free components in this case, we exclude from the probability of having at least  $M$  fault-free components the probability of having at least  $(M+1)$  fault-free components:

$$P_{ff}^{cg}(N, M) = P_{\geq}^{cg}(N, M) - P_{\geq}^{cg}(N, M+1) \quad (7)$$

As mentioned above, fine-grain logic can also be used in order to instantiate different SUs and increase the resilience in the presence of permanent faults. We can calculate the probability of having a specific number of fault-free components when both fine-grain and coarse-grain reconfigurability is employed by modifying the probabilities for coarse-grain designs derived above.

*Figure 16* can be used again as an example that depicts a design with four components, each divided into four SUs, having both fine- and coarse-grain reconfigurability. When only coarse-grain replacement is used, having two faulty SUs on one column and at most one in the remaining columns limits the number of fault-free components to two. In a coarse-grain design the above event is included in the probability of having exactly two fault-free components. Introducing fine-grain logic to replace one faulty SU, rescues one additional component in the above example. Consequently, this event (two faulty units at one column and at most one in the remaining columns) should be included in the list of events which are counted under the probability of having three fault-free components and should be removed from the events that are considered under the probability of having two fault-free components.

Finding events that should be removed or added to the "coarse-grain" probability depends on the number of SUs that can fit inside the fine-grain logic. In this work we only consider the case where the fine-grain can be used to replace exactly one faulty SU. As all events are independent, it is possible to separately compute their probability and append it to the  $P_{ff}^{cg}(N, M)$  of equation 7.  $P_{append}^{+}(N, M)$  denotes the probability of events originally considered in the probability of having  $(M-1)$  fault-free components but when using fine-grain logic are becoming part of the probability for  $M$  fault-free components. Similarly,  $P_{append}^{-}(N, M)$  is the probability of events originally counted in  $P_{ff}^{cg}(N, M)$  but using fine-grain replacement moves them to the probability of having  $(M+1)$  fault-free components.

$$P_{append}^{+}(N, M) = \binom{c}{1} \times [P_{ff}(N, M-1)]^1 \times [P_{\geq}(N, M)]^{c-1} \quad (8)$$

where  $c$  is the number of columns.

Similarly,  $P_{append}^{-}(N, M)$  can be calculated as:

$$P_{append}^{-}(N, M) = \binom{c}{1} \times [P_{ff}(N, M)]^1 \times [P_{\geq}(N, M+1)]^{c-1} \quad (9)$$

Then, the probability of having exactly  $M$  fault-free components out of  $N$  for a design which uses fine-grain logic in addition to coarse-grain replacement is:

$$P_{ff}^{cg+fg}(N, M) = P_{ff}^{cg}(N, M) + P_{append}^+(N, M) - P_{append}^-(N, M) \quad (10)$$

Similar to the previous cases, in order to find the probability of having at least  $M$  fault-free components out of  $N$  in a design with both fine and coarse-grain reconfigurability, it is possible to add all probabilities of having exactly  $M, M+1, \dots, N$  fault-free components:

$$P_{\geq}^{cg+fg}(N, M) = \sum_{i=M}^N P_{ff}^{cg+fg}(N, i) \quad (11)$$

The probabilities of having at least  $M$  fault-free components out of  $N$  in the different design approaches considered ( $P_{\geq}^{cg}(N, M)$ ,  $P_{\geq}^{cg+fg}(N, M)$ ) are used in the evaluation section to measure the probability of a design to guarantee a particular availability of components.

The probabilities of having exactly  $M$  fault-free components out of  $N$  in the considered design alternatives ( $P_{ff}^{cg}(N, M)$ ,  $P_{ff}^{cg+fg}(N, M)$ ) are used to evaluate the average number of fault-free components in a given area as described below. For a specific number of faults,  $k$ , the above probability ( $P_{ff}^j(N, M)$ , where  $j$  is ``cg" or ``cg+fg"') is used to calculate the average number of fault-free components, as the weighted average of the individual probabilities:

$$\bar{x} = \sum_{i=1}^N P_{ff}^j(N, i) \times i \quad (12)$$

where  $N$  is total number of components.

## 4.2 Highlights of our T4.4 contribution

The above analysis was the basis for quantifying the availability of the proposed adaptive reconfigurable multicore architecture of D4.5. That is the number of faults sustained in a system composed of the proposed processors before failure, as well as the average availability of the multicore array for different fault densities.

Our results presented in detail in D4.5 (Section 3.10) reveal that mixing fine-grain logic with a coarse-grain sparing approach tolerates up to  $3\times$  more permanent faults than component redundancy and  $2\times$  more than any other purely coarse-grain solution.

Component redundancy is preferable at low fault densities, while coarse-grain and mixed grain reconfigurability maximize availability at medium and high fault densities, respectively.

## 5. Conclusions

The deliverable *D4.8 - Description of detailed design of architecture and API for error handling and redundancy of accelerators* [3] described solutions for the dynamically reconfigurable subsystems.

The deliverable *D4.9 – Description of first demonstrator and start of final evaluation phase* has been building on D4.8 [3] results. It provided first measurements and results from the evaluation phase. The specification of preliminary architecture has been described in the D4.6 deliverable [1]. The D4.7 deliverable [2] provided description of detailed design of architecture and API for error handling and redundancy of accelerators within the EMC2 WP4 work package. It related mainly to the technology developed in Task 4.4 and also to several demonstrators described in the WP4 milestone-description deliverables D4.15 [4], D4.16 [5] and D4.17 [7].

This deliverable *D4.10 – Report describing the evaluation of final implementation, evaluation of innovation results* presents the final implementation of the technology developed in the task T4.4 on the commercial EMC2-DP HW platform [23]. The released final demonstrator [11] is compatible with the Xilinx SDSoC 2015.4 environment [34].

It is actively advertised and exploited by the company SMT. See [22].

The references [10] – [21] provide list of publicly released application notes and evaluation packages.

The demonstrator developed by Tecnalía and TASE is used in the EMC2 living lab 3 (Space applications).

Main technology achievement/innovation reported in D4.10:

- Speed optimized asymmetric multiprocessing designs [11] for the standalone, industrial grade, EMC2-DP-V2 development platform [23]. This platform is produced by the EMC2 WP4 partner Sundance Multiprocessor Technology.
- The released final demonstrator [11] is compatible with the Xilinx software defined system on chip environment (SDK 2015.4) [33].
- It supports the triple redundancy dynamically reconfigurable 3x (8xSIMD) EdkDSP floating point accelerators.

The D4.10 deliverable provides publicly accessible description of results achieved by project partners in the area of dynamic reconfiguration serving for increased error handling capabilities and for increased redundancy of accelerators.

UTIA contribution is addressing the reconfiguration of HW as part of the **asymmetric multiprocessing package** [11] on the 28nm Zynq device on the **industrial grade EMC2-DP HW** [23] developed by SMT. Floating point accelerators are reconfigured in the run-time by change of the firmware of internal scheduling controllers.

The joint work of project partners Tecnalía and TASE resulted in the final demonstrator with well-defined error-handling capabilities and the **increased reliability of FPGA partial reconfiguration** for the space domain. These results are used in the EMC2 living lab 3 (Space applications).

Finally, the summary of results of **generic research of resilient reconfigurable multiprocessor arrays** has been presented by project partner Chalmers. It complements the application-oriented results of the first two teams of partners.

The final demonstrator [11] on the industrial platform [23] is used by several project partners of the EMC2 project and it is already evaluated also by companies and research organizations outside of the project.

## 6. References

### EMC2 Deliverables

- [1] EMC2 D4.6 – Specification of preliminary architecture and API respecting the requirements for error handling and redundancy of accelerators”; ver1.0; Released 31/03/2015.
- [2] EMC2 D4.7 – Description of detailed design of architecture and API for error handling and redundancy of accelerators; V1.3; Released 30/9/2015
- [3] EMC2 D4.8 – Description of first demonstrator of architecture and API for error handling and redundancy of accelerators; V1.0; Released 25/03/2016.
- [4] EMC2 D4.9 – Description of first demonstrator and start of final evaluation phase; V0.5; Released 21/09/2016.
- [5] EMC2 D4.15 – First demonstration platforms for use in LLs. Preliminary EMC2 WP4 demonstrator milestone description deliverable; V1.1; Released 27/11/2014.
- [6] EMC2 D4.16 – Enhanced demonstration platforms including basic innovative techniques. Enhanced EMC2 WP4 demonstrator milestone description deliverable; released 29/10/2015.
- [7] EMC2 D4.17 – Final demonstration platforms including basic innovative techniques. V1.0 Released 3/10/2016
- [8] EMC2 D9.2 – Space Application Concept Report
- [9] EMC2 D9.5 – Space Applications Detailed Description; V1.0; Released 17/12/2016.

### Public Application Notes with Evaluation Packages

- [10] UTIA public www server dedicated to the EMC2 project  
<http://sp.utia.cz/index.php?ids=projects/emc2>
- [11] Jiří Kadlec, Zdeněk Pohl, Lukáš Kohout: Full HD Video Processing in HW with three EdkDSP 8xSIMD Accelerators for TE0715-30-1SoM on EMC2-DP-V2 Carrier. Public application note with the evaluation package. Rev. 1. Released 20/2/2017.  
<http://sp.utia.cz/index.php?ids=results&id=s30i1hm4>  
[http://sp.utia.cz/results/s30i1hm4/s30i1hm4\\_2015\\_4.pdf](http://sp.utia.cz/results/s30i1hm4/s30i1hm4_2015_4.pdf)
- [12] Jiří Kadlec, Zdeněk Pohl, Lukáš Kohout: SDSoC 2015.4 Standalone BSP with Full HD HDMI In-Out with SW and HW Demos for Zynq System-on-Module TE0715-03-30 and Sundance EMC2-DP-V2 Platform. Public application note with the evaluation package. Rev 1. Released 12/8/2016.  
<http://sp.utia.cz/index.php?ids=results&id=s30i1h2>  
[http://sp.utia.cz/results/s30i1h2/s30i1h2\\_2015\\_4.pdf](http://sp.utia.cz/results/s30i1h2/s30i1h2_2015_4.pdf)
- [13] Jiří Kadlec, Zdeněk Pohl, Lukáš Kohout: Full HD HDMI In-Out HW-Accelerated Demos for Zynq System-on-Module TE0715-03-30-1I and Sundance EMC2-DP-V2 Platform. Public application note with the evaluation package. Rev. 4. Released 21/7/2016.  
<http://sp.utia.cz/index.php?ids=results&id=s30i1h1>  
[http://sp.utia.cz/results/s30i1h1/s30i1h1\\_2015\\_4\\_te0720.pdf](http://sp.utia.cz/results/s30i1h1/s30i1h1_2015_4_te0720.pdf)
- [14] Lukas Kohout, Zdenek Pohl, Jiri Kadlec: EMC2-DP HDMI in HDMI out Platform. Public application note with the evaluation package. Rev 1. Released 8/8/2016.  
<http://sp.utia.cz/index.php?ids=results&id=emc2-dp-platform>  
<http://sp.utia.cz/results/emc2-dp-platform/emc2-hio-appnote-v2.pdf>
- [15] Jiří Kadlec, Zdeněk Pohl, Lukáš Kohout: Asymmetric Multiprocessing with MicroBlaze, EdkDSP Accelerator and Toshiba Sensor Video Processing for low cost Zynq on TE0720-03-1CF SoM on TE0701-05 Carrier. Public application note with the evaluation package. Rev. 2. Released 14/7/2016.  
<http://sp.utia.cz/index.php?ids=results&id=t20c1tm1>  
[http://sp.utia.cz/results/t20c1tm1/t20c1tm1\\_2015\\_4\\_te0720.pdf](http://sp.utia.cz/results/t20c1tm1/t20c1tm1_2015_4_te0720.pdf)
- [16] Jiří Kadlec, Zdeněk Pohl, Lukáš Kohout: Asymmetric Multiprocessing with MicroBlaze, EdkDSP Accelerator and Toshiba Sensor Video for Automotive grade Zynq on TE0720-03-1QF SoM on TE0701-05 Carrier. Public application note with the evaluation package. Rev. 2. Released 14/7/2016.

- <http://sp.utia.cz/index.php?ids=results&id=t20q1tm1>  
[http://sp.utia.cz/results/t20q1tm1/t20q1tm1\\_2015\\_4\\_te0720.pdf](http://sp.utia.cz/results/t20q1tm1/t20q1tm1_2015_4_te0720.pdf)
- [17] Jiří Kadlec, Zdeněk Pohl: Evaluation of Asymmetric Multiprocessing for Zynq System-on-Modules TE0720-02-2IF, TE0720-02-1CF, TE0720-02-1QF with Carrier Board TE0701-05. Public application note with the evaluation package. Released 21/11/2015.  
[http://sp.utia.cz/index.php?ids=results&id=emc2\\_amp\\_on\\_zynq\\_trenz\\_2015\\_2](http://sp.utia.cz/index.php?ids=results&id=emc2_amp_on_zynq_trenz_2015_2)  
[http://sp.utia.cz/results/emc2\\_amp\\_on\\_zynq\\_trenz\\_2015\\_2/Utia\\_EdkDSP\\_Vivado\\_2015\\_2 EMC2\\_2\\_te0720\\_te0701.pdf](http://sp.utia.cz/results/emc2_amp_on_zynq_trenz_2015_2/Utia_EdkDSP_Vivado_2015_2 EMC2_2_te0720_te0701.pdf)
- [18] Lukáš Kohout, Jiří Kadlec, Zdeněk Pohl: Video Input/Output Demonstration for Trenz TE0701-05, TE0720-02-1CF, TE0720-02-1QF, TE0720-02-2IF and Avnet HDMI Input/Output FMC Module. Public application note with the evaluation package. Rev. 1. Released 28/8/2015.  
<http://sp.utia.cz/index.php?ids=results&id=te0701-05-te0720-fmc-imageon>  
<http://sp.utia.cz/results/te0701-05-te0720-fmc-imageon/te0701-05-te0720-fmc-imageon-appnote-2014.4.pdf>  
<http://sp.utia.cz/results/te0701-05-te0720-fmc-imageon/te0701-05-te0720-fmc-imageon-appnote-2015.2.pdf>
- [19] Zdeněk Pohl: Dynamic Programmable Logic Reconfiguration for Zynq. Public application note with the evaluation package. Rev.1. Released 18/12/2014.  
<http://sp.utia.cz/index.php?ids=results&id=plreconf>
- [20] Jiří Kadlec, Zdeněk Pohl: Asymmetric Multiprocessing on Zynq ZC702 board with EdkDSP Accelerators for Xilinx Vivado 2013.4 Design Flow. Public application note with the evaluation package. Rev. 3. Released 19/11/2014.  
[http://sp.utia.cz/index.php?ids=results&id=Utia\\_EdkDSP\\_Vivado\\_2013\\_4 EMC2](http://sp.utia.cz/index.php?ids=results&id=Utia_EdkDSP_Vivado_2013_4 EMC2)  
[http://sp.utia.cz/results/Utia\\_EdkDSP\\_Vivado\\_2013\\_4 EMC2/Utia\\_EdkDSP\\_Vivado\\_2013\\_4 EMC2\\_ZC702.pdf](http://sp.utia.cz/results/Utia_EdkDSP_Vivado_2013_4 EMC2/Utia_EdkDSP_Vivado_2013_4 EMC2_ZC702.pdf)
- [21] Jiří Kadlec: Asymmetric Multiprocessing on Zynq with EdkDSP accelerators on Xilinx ZC702 board – ISE 14.5. Public application note with the evaluation package. Rev. 4. Released 5/11/2014.  
[http://sp.utia.cz/index.php?ids=results&id=Utia\\_EdkDSP\\_145\\_ZC702](http://sp.utia.cz/index.php?ids=results&id=Utia_EdkDSP_145_ZC702)  
[http://sp.utia.cz/results/Utia\\_EdkDSP\\_145\\_ZC702/Utia\\_EdkDSP\\_145 EMC2\\_ZC702.pdf](http://sp.utia.cz/results/Utia_EdkDSP_145_ZC702/Utia_EdkDSP_145 EMC2_ZC702.pdf)

### HW Components, System on Modules, Carrier Boards

- [22] Overview of COTS PC/104 carrier cards and boards offered by Sundance Technology.  
<http://www.sundance.technology/som-carriers/pc104-boards/>
- [23] EMC2-DP C/104 OneBank Carrier for SoC Modules. Block diagram. Sundance Technology  
<http://www.sundance.technology/som-carriers/pc104-boards/emc2-dp/>  
<http://www.sundance.technology/wp-content/uploads/2015/03/emc2-dp.pdf>
- [24] Timoteo Garcia: EMC2-DP BOARD IO PCIe/104 OneBank™ Carrier for 40mm x 50mm SoM + VITA57.1 FMC™ Modules. Sundance Technology.  
 Document Issue Number: 1.0\_z712-15-30; Original Issue Date: 17/03/2016  
[http://www.sundance.technology/wp-content/uploads/2015/12/EMC2-Board-IO-v1.0\\_z712-15-30.pdf](http://www.sundance.technology/wp-content/uploads/2015/12/EMC2-Board-IO-v1.0_z712-15-30.pdf)
- [25] TE0720-03-2IF; Part: XC7Z020-2CLG484I; Trenz Electronic.  
<http://shop.trenz-electronic.de/en/TE0720-03-2IF-Xilinx-Zynq-module-XC7Z020-2CLG484I-ind.-temp.-range-1-Gbyte>
- [26] TE0715-03-30-1I; Part: XC7Z030-1SBG485I; Trenz Electronic.  
<https://shop.trenz-electronic.de/en/TE0715-03-30-1I-Xilinx-Zynq-Z-7030-SoC-Micromodule-XC7Z030-1SBG485I-ind.-temp.-range>
- [27] TE0715-03-30-3E; Part: XC7Z030-3SBG485E; Trenz Electronic.  
<https://shop.trenz-electronic.de/en/TE0715-04-30-3E-SoC-Micromodule-with-Xilinx-Zynq-XC7Z030-3SBG485E-ind.-temp.-range>
- [28] Heatsink for TE0720, spring-loaded embedded; Trenz Electronic.  
<https://shop.trenz-electronic.de/en/26922-Heatsink-for-TE0720-spring-loaded-embedded?c=38>
- [29] Heatsink for TE0715, spring-loaded embedded; Trenz Electronic.



- [30] <https://shop.trenz-electronic.de/en/26923-Heatsink-for-TE0715-spring-loaded-embedded?c=346>  
TE0701-05 Carrier Board for Trenz Electronic 7 Series.  
[https://www.trenz-electronic.de/fileadmin/docs/Trenz\\_Electronic/carrier\\_boards/TE0701/REV05/Documents/TRM-TE0701-05.pdf](https://www.trenz-electronic.de/fileadmin/docs/Trenz_Electronic/carrier_boards/TE0701/REV05/Documents/TRM-TE0701-05.pdf)
- [31] TE0701-06 Carrier Board for Trenz Electronic 7 Series.  
[https://www.trenz-electronic.de/fileadmin/docs/Trenz\\_Electronic/carrier\\_boards/TE0701/REV06/Documents/TRM-TE0701-06.PDF](https://www.trenz-electronic.de/fileadmin/docs/Trenz_Electronic/carrier_boards/TE0701/REV06/Documents/TRM-TE0701-06.PDF)
- [32] AES-FMC-HDMI-CAM-G; FMC card with HDMI I/O and CAM interface. Avnet.  
<http://products.avnet.com/shop/en/ema/3074457345623664802>

### **SW/HW Tools and Design Flows**

- [33] Vivado HLx Web Install Client - 2015.4. Xilinx.  
<http://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/vivado-design-tools/2015-4.html>
- [34] SDSoC - 2015.4 Full Product Installation. Xilinx.  
<http://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/sdx-development-environments/sdsoc/2015-4.html>
- [35] UG744, Partial Reconfiguration of a Processor Tutorial v14.5, April 2013. Xilinx.
- [36] XAPP887, PRC/EPRC: Data Integrity and Security Controller for Partial Reconfiguration, v1.1, June 2012. Xilinx.

### **Conference papers**

- [37] Alirad Malek, Stavros Tzilis, Danish Anis Khan, Ioannis Sourdis, Georgios Smaragdos and Christos Strydis, in International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2015.

### **Press releases**

- [38] Sundance Multiprocessor Technology Ltd. integrates the new Xilinx SDSoC development environment on to its EMC2-Z7030 Zynq SoC-based PC/104 single board computer family. Press release issued by SMT:  
[http://www.artemis-emc2.eu/fileadmin/user\\_upload/Publications/EMC2-Z7030-SDSoC\\_Sundance\\_20160217.pdf](http://www.artemis-emc2.eu/fileadmin/user_upload/Publications/EMC2-Z7030-SDSoC_Sundance_20160217.pdf)

## 7. Annex A – Performance Measurements for the Final EMC2-DP Demonstrator

### 7.1 Project sh01: Edge detection with single HW accelerator and 3x EdkDSP

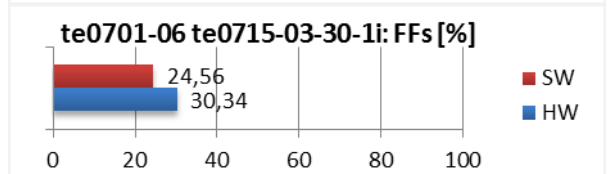
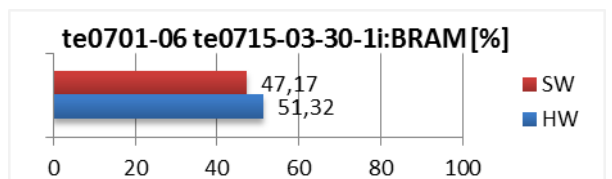
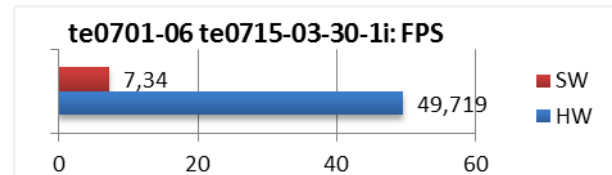
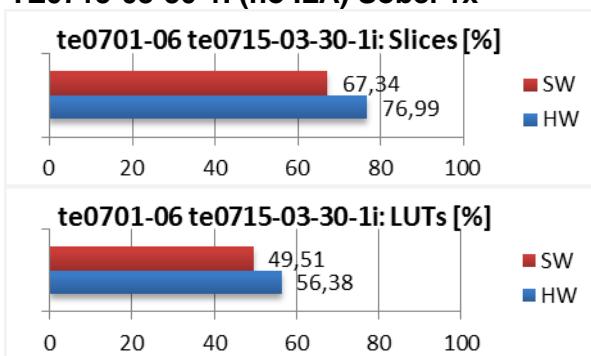
The accelerated data path runs at 200 MHz and includes HW version of the function:

- sobel\_filter\_hitle1

with one data-mover controlling one input and one output AXI DMA channel to the DDR3.

Acceleration by HW: **6.77 x**

#### TE0715-03-30-1I (no ILA) Sobel 1x



#### TE0715-03-30-1I (with ILA) Sobel 1x

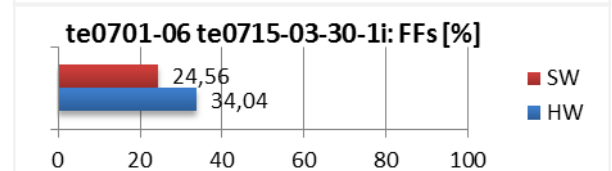
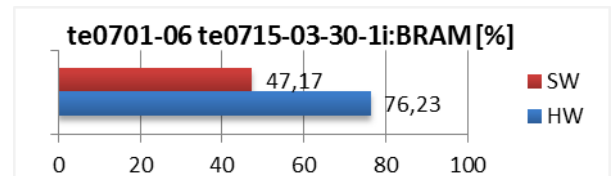
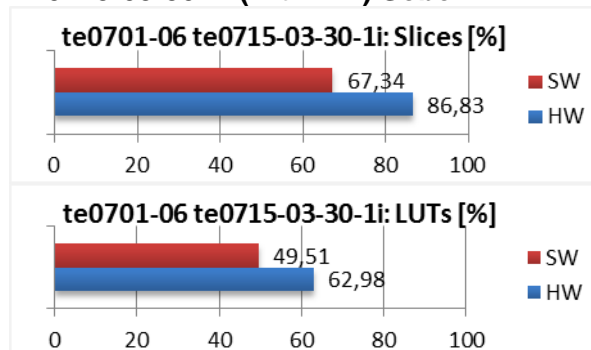


Figure 17: Project sh01 - Acceleration and HW resources used.

#### Power consumption

Power consumption of complete working system has been measured. See Figure 18 and Figure 19:

- ARM A9 + video I/O and HW accelerators + MicroBlaze
- ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW instantiated
- ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW filters are present. One instantiated EdkDSP accelerators is computing the LMS or FIR filter in floating point:
  - One EdkDSP HW accelerator computing LMS Filter: 429 mW/GFLOP/s
  - One EdkDSP HW accelerator computing FIR Filter: 293 mW/GFLOP/s
  - One EdkDSP HW accelerator computing LMS Filter with ILA: 442 mW/GFLOP/s
  - One EdkDSP HW accelerator computing FIR Filter with ILA: 302 mW/GFLOP/s

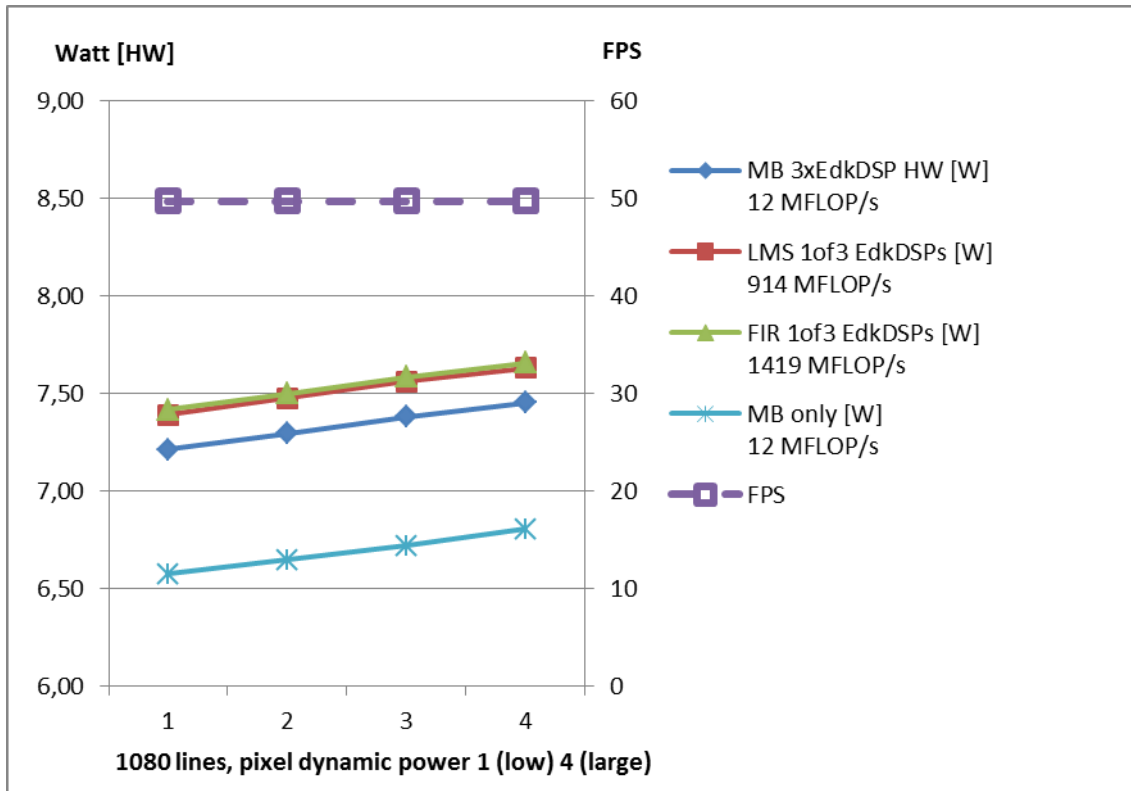


Figure 18: Power consumption of sh01 demo without ILA

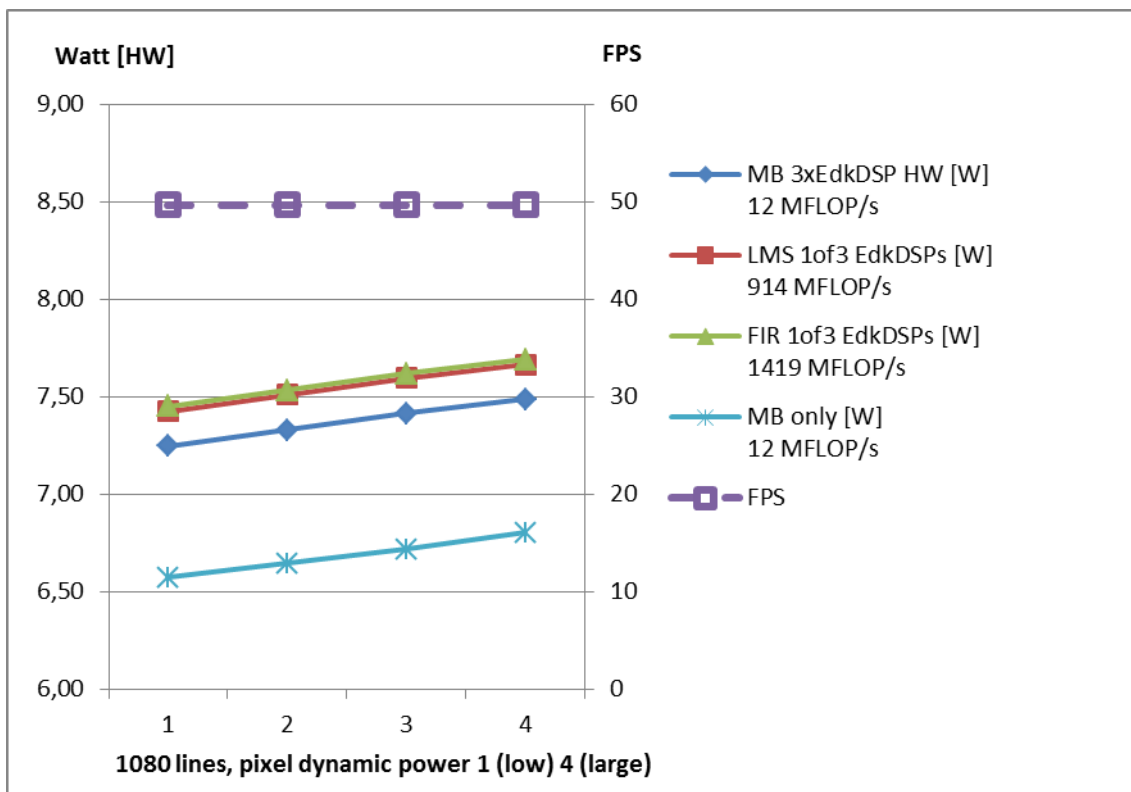


Figure 19: Power consumption of sh01 demo with ILA

## 7.2 Project sh02: Edge detection with two HW accelerators and 3x EdkDSP

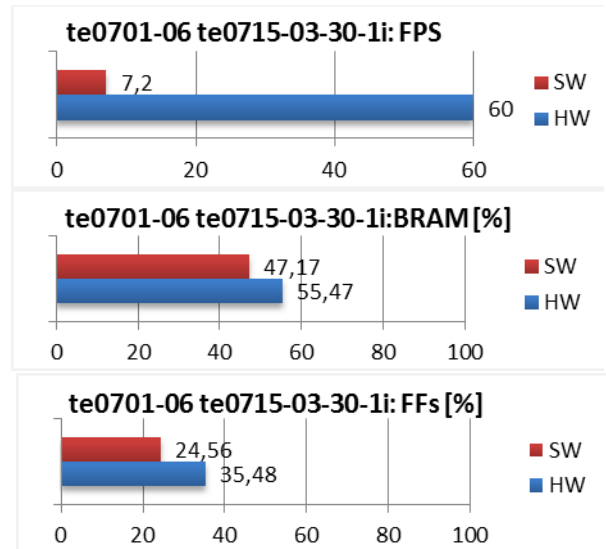
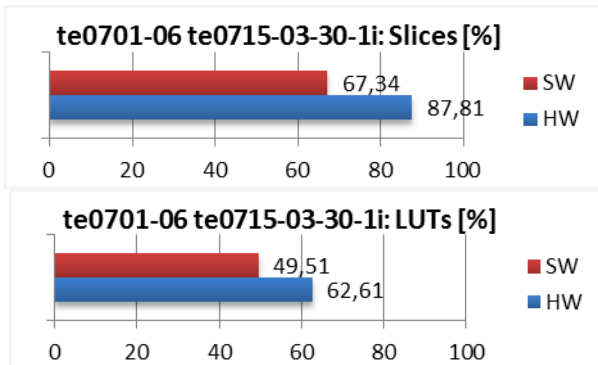
The two parallel accelerated data paths run at 200 MHz and include HW version of these functions:

- sobel\_filter\_htile1
- sobel\_filter\_htile2

with two data-movers controlling two input and two output AXI DMA channels to the DDR3.

Acceleration by HW: **8.33 x**

TE0715-03-30-1i (no ILA) Sobel 2x



TE0715-03-30-1i (with ILA) Sobel 2x

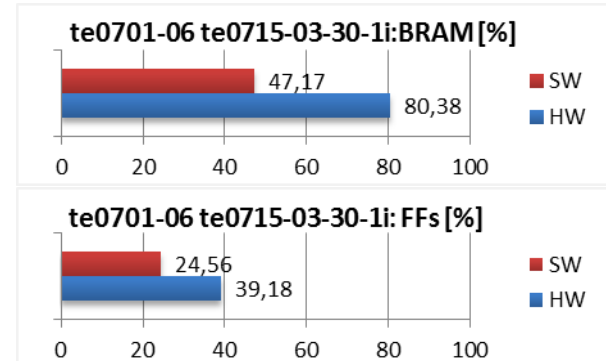
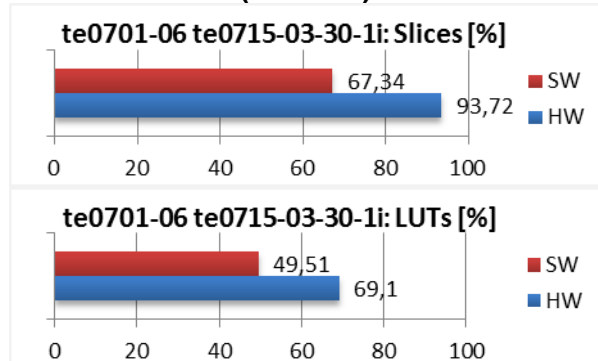


Figure 20: Project sh02 – Acceleration and HW resources used.

### Power consumption:

Power consumption of complete working system has been measured. See Figure 21 and Figure 22:

- ARM A9 + video I/O and HW accelerators + MicroBlaze
- ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW instantiated
- ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW filters are present. One instantiated EdkDSP accelerators is computing the LMS or FIR filter in floating point:
  - One EdkDSP HW accelerator computing LMS Filter: 420 mW/GFLOP/s
  - One EdkDSP HW accelerator computing FIR Filter: 288 mW/GFLOP/s
  - One EdkDSP HW accelerator computing LMS Filter with ILA: 438 mW/GFLOP/s
  - One EdkDSP HW accelerator computing FIR Filter with ILA: 299 mW/GFLOP/s

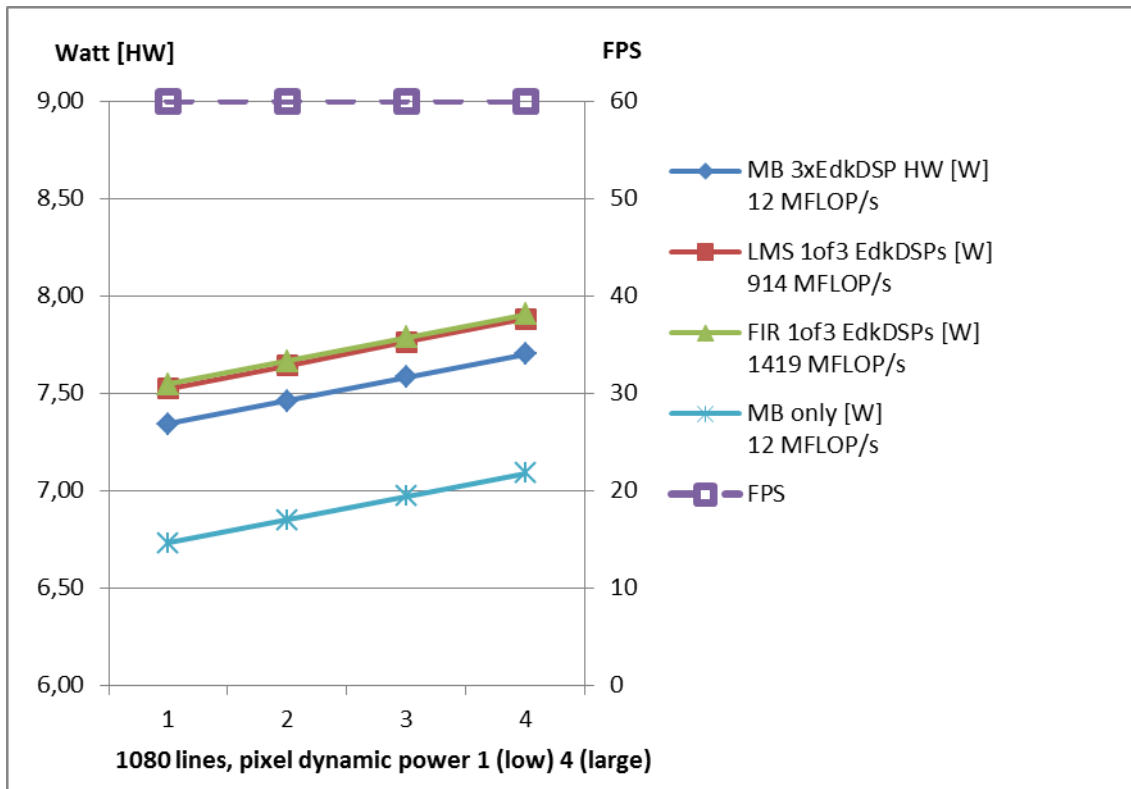


Figure 21: Power consumption of sh02 demo without ILA

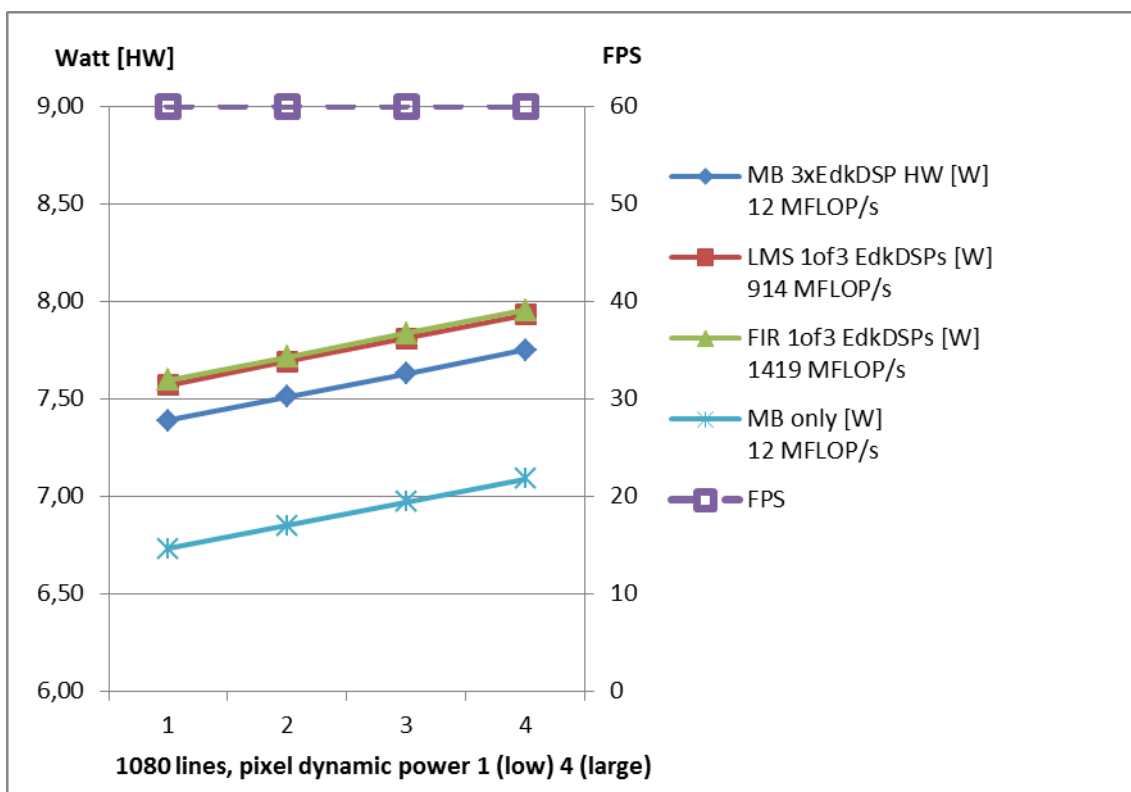


Figure 22: Power consumption of sh02 demo with ILA

### 7.3 Project sh03: Edge detection with three HW accelerators and 3x EdkDSP

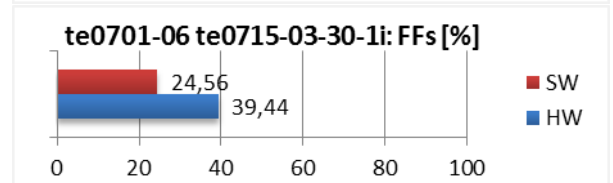
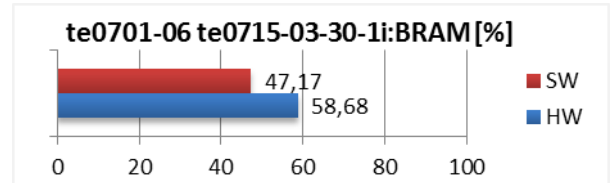
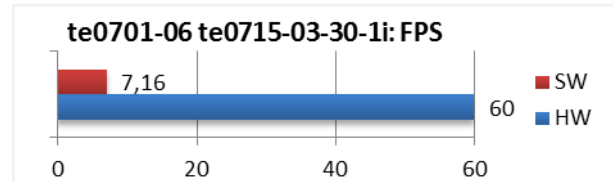
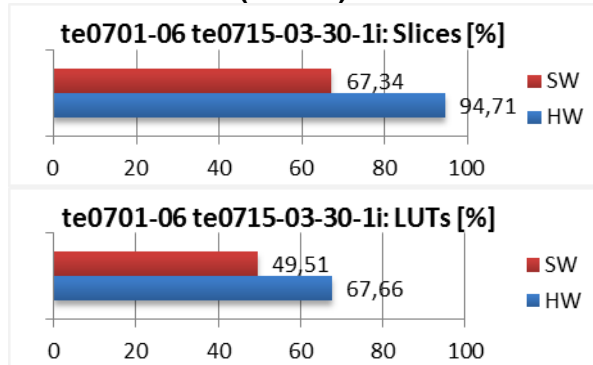
The three parallel accelerated data paths run at 200 MHz and include HW version of these functions:

- sobel\_filter\_htile1
- sobel\_filter\_htile2
- sobel\_filter\_htile3

with three data-movers controlling three input and three output AXI DMA channels to the DDR3.

Acceleration by HW: **8.38 X**

#### TE0715-03-30-1I (no ILA) Sobel 3x



#### TE0715-03-30-1I (with ILA) Sobel 3x

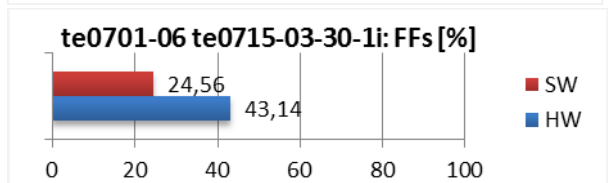
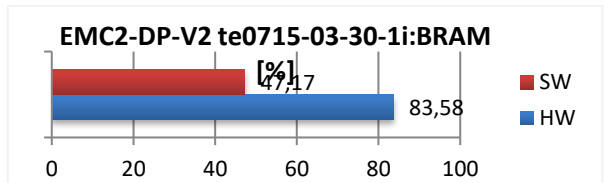
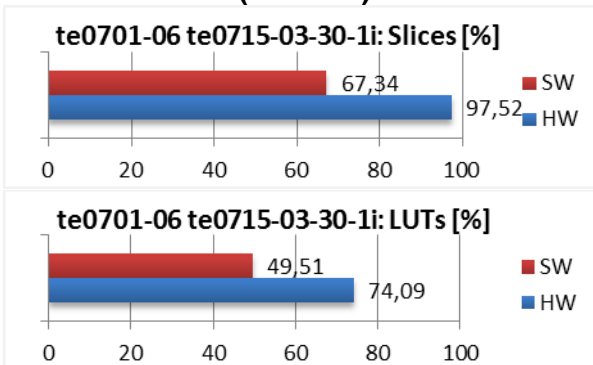


Figure 23: Project sh03 - Acceleration and HW resources used.

#### Power consumption

Power consumption of complete working system has been measured. See Figure 24 and Figure 25 :

- ARM A9 + video I/O and HW accelerators + MicroBlaze
- ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW instantiated
- ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW filters are present. One instantiated EdkDSP accelerators is computing the LMS or FIR filter in floating point:
  - One EdkDSP HW accelerator computing LMS Filter: 420 mW/GFLOP/s
  - One EdkDSP HW accelerator computing FIR Filter: 288 mW/GFLOP/s
  - One EdkDSP HW accelerator computing LMS Filter with ILA: 433 mW/GFLOP/s
  - One EdkDSP HW accelerator computing FIR Filter with ILA: 296 mW/GFLOP/s

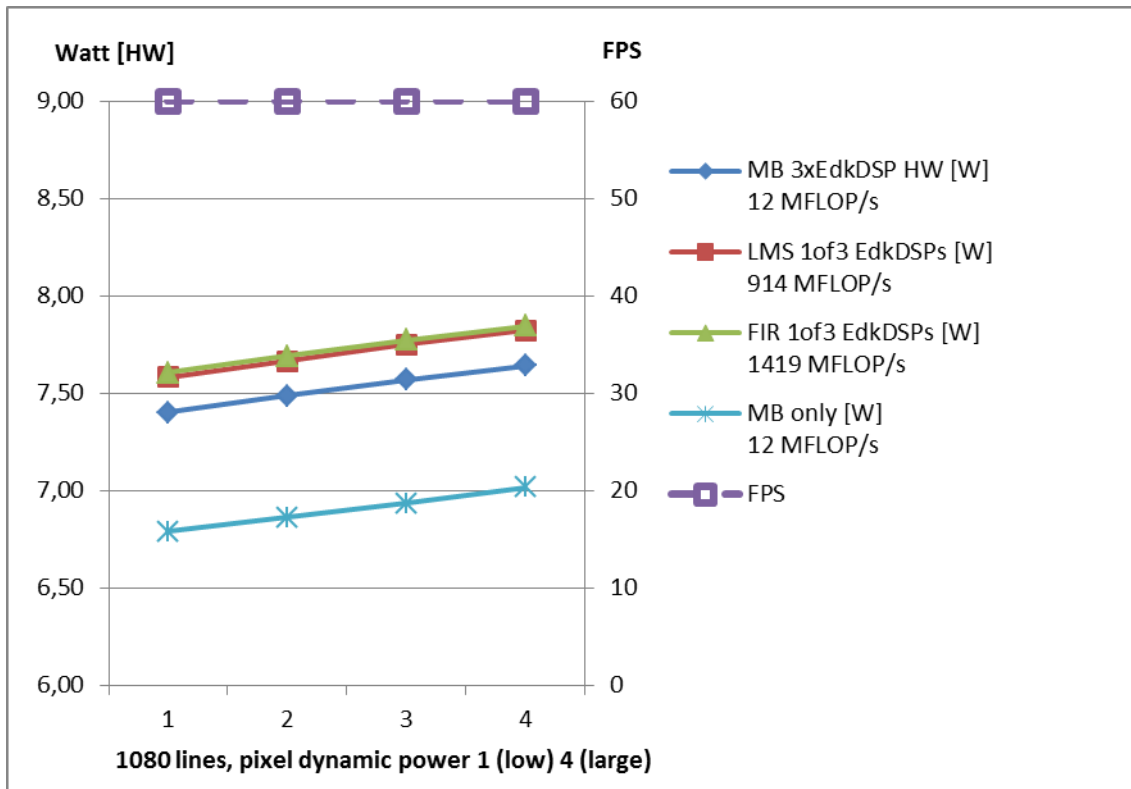


Figure 24: Power consumption of sh03 demo without ILA

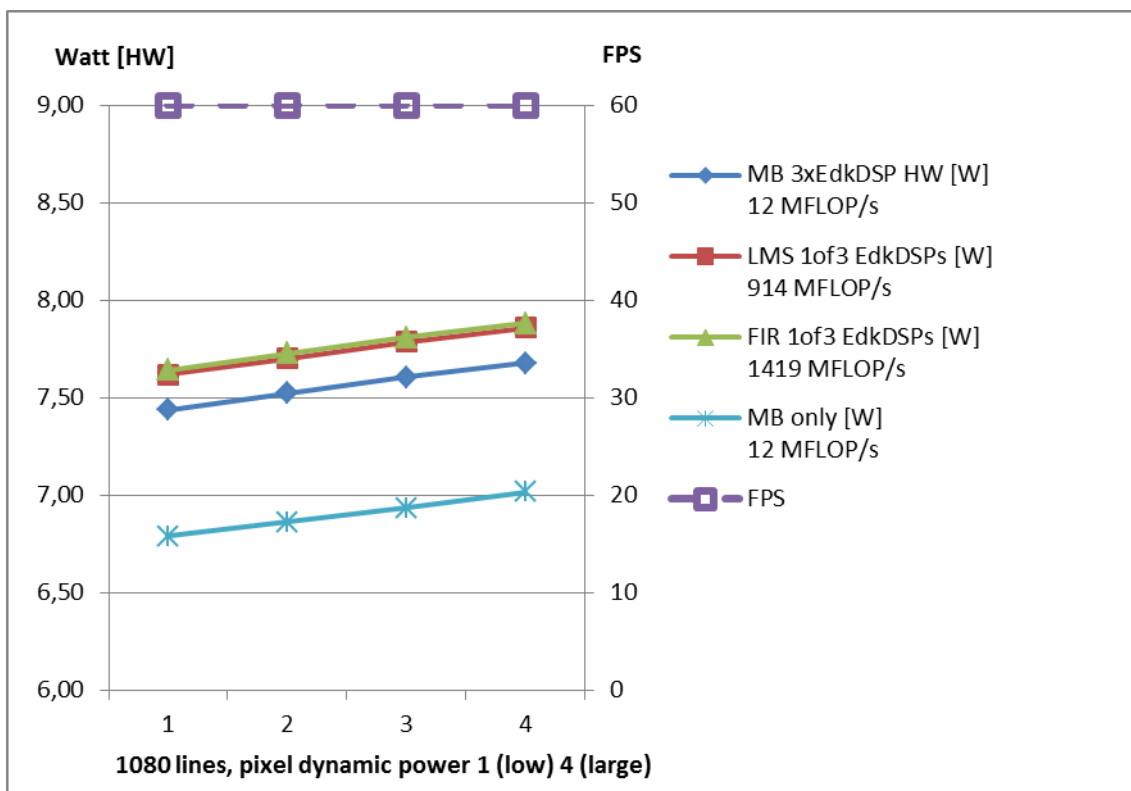


Figure 25: Power consumption of sh03 demo with ILA

## 7.4 Project md01: Motion detection with chain of HW accelerators, 3x EdkDSP

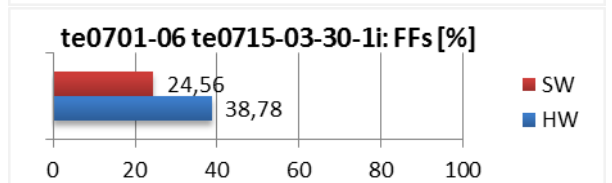
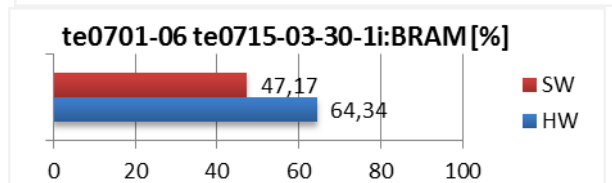
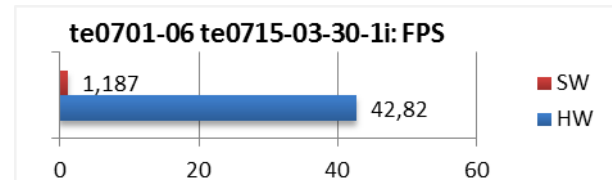
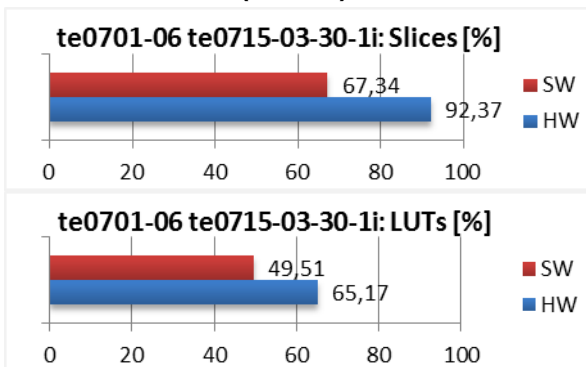
Motion detection is implemented as single accelerated data path run at 200 MHz. It processes data from two subsequent video frames and it includes chain of HW versions of these functions:

- (1) pad (two instances)
- (2) sobel\_filter\_pass sobel\_filter
- (3) diff\_image
- (4) median\_char\_filter\_pass
- (5) combo\_image
- (6) ext

four data-movers controlling the 200 MHz input AXI DMA channels from the DDR3 and one data-mover controlling the 200 MHz output AXI DMA channel to the DDR3.

Acceleration by HW: **35.45 x**

### TE0715-03-30-1I (no ILA) Motion Det. 1x



### TE0715-03-30-1I (with ILA) Motion Det. 1x

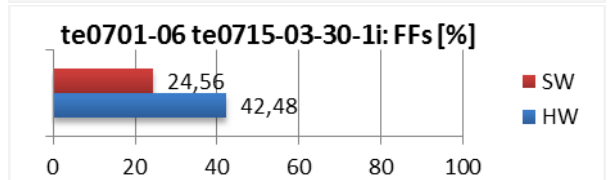
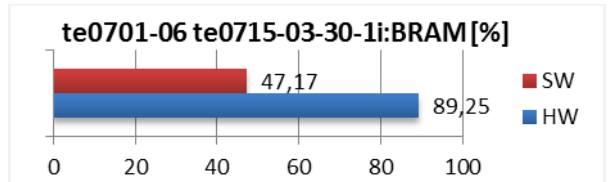
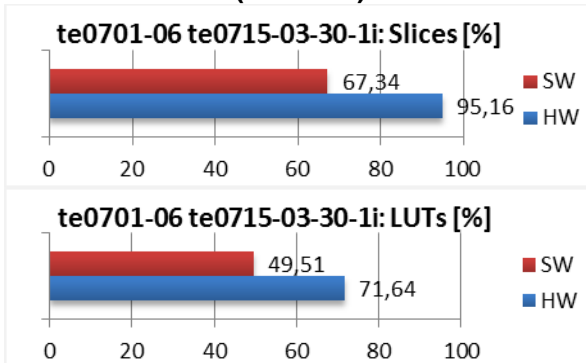


Figure 26: Project md01 - Acceleration and HW resources used

### Power consumption

Power consumption of complete working system has been measured. See Figure 27 and Figure 28:

- ARM A9 + video I/O and HW accelerators + MicroBlaze
- ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW instantiated
  - ARM A9 + video I/O and HW accelerators + MicroBlaze + 3x EdkDSP HW filters are present. One instantiated EdkDSP accelerators is computes LMS or FIR filter:
    - One EdkDSP HW accelerator computing LMS Filter: 433 mW/GFLOP/s
    - One EdkDSP HW accelerator computing FIR Filter: 296 mW/GFLOP/s
    - One EdkDSP HW accelerator computing LMS Filter with ILA: 425 mW/GFLOP/s
    - One EdkDSP HW accelerator computing FIR Filter with ILA: 290 mW/GFLOP/s



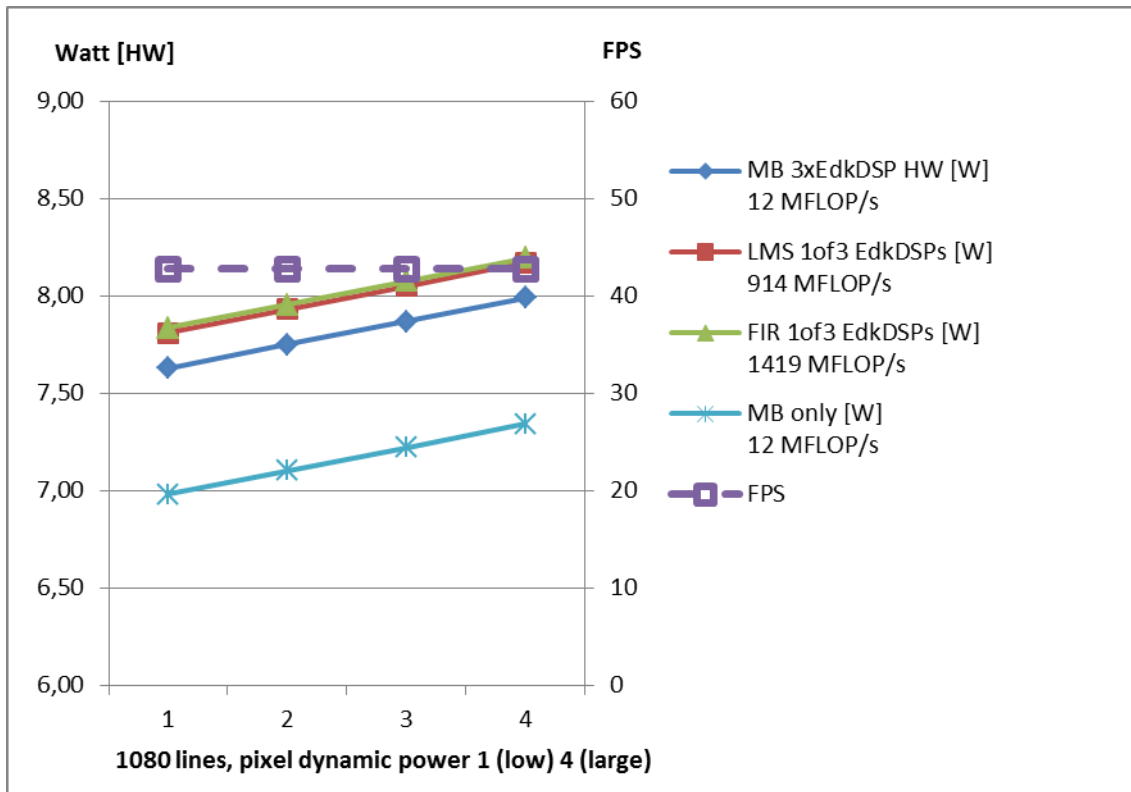


Figure 27: Power consumption of md01 demo without ILA

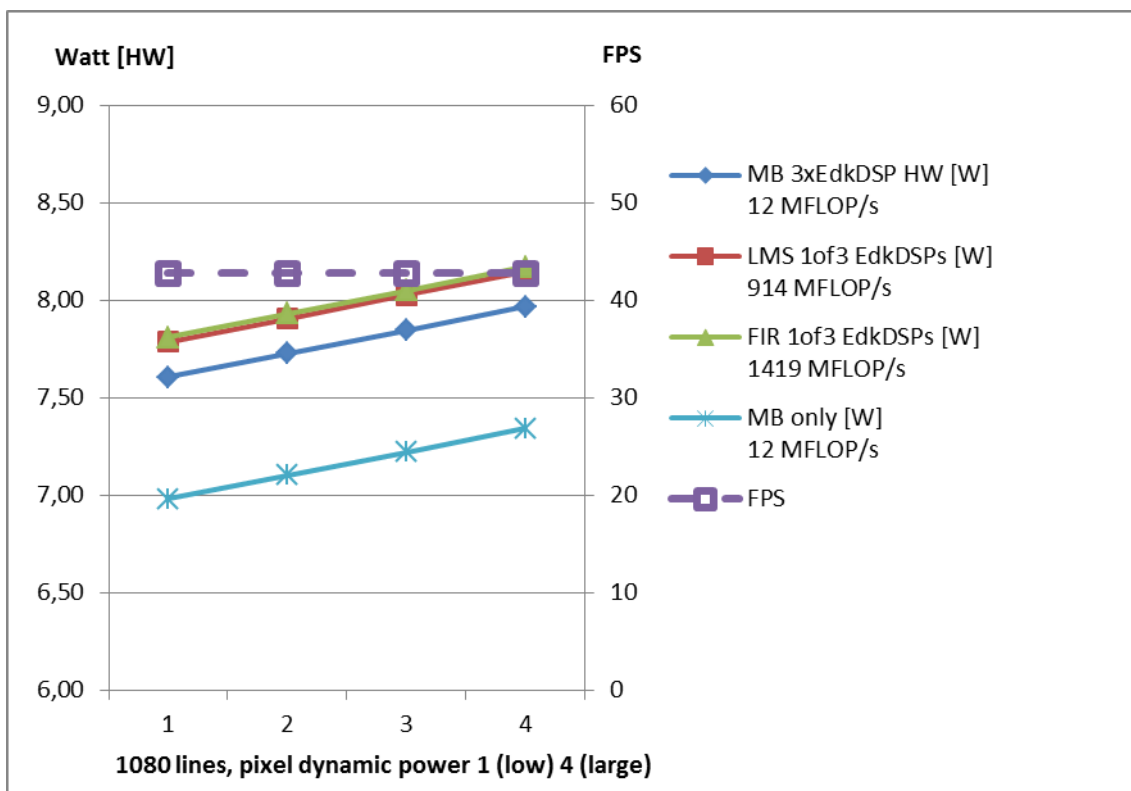


Figure 28: Power consumption of md01 demo with ILA

## 8. Annex B – UTIA Licensing of the EMC2-DP Demonstrator Evaluation Packages

The evaluation version of the package [11] can be downloaded from UTIA www pages [10] free of charge.

### Deliverables:

The evaluation package includes evaluation bitstreams with three (8xSIMD) EdkDSP accelerators working in parallel with the HW-accelerated edge detection and motion detection algorithms for the Full HD HDMI-HDMI video processing on the Trenz TE0715-03-30-11 module [26] located on the SMT's EMC2-DP-V2 carrier [23] with the FMC card [32].

The evaluation package [11] includes bitstreams compiled with the evaluation version of the UTIA (8xSIMD) EdkDSP HW accelerator IP core. Evaluation IPs compiled in the enclosed bitstreams:

<b>bce_fp12_1x8_0_axiw_v1_10_c</b>	Evaluation version of the AXI-lite interface
<b>bce_fp12_1x8_40</b>	Evaluation version of the floating point data path

This evaluation version of the UTIA (8xSIMS) EdkDSP accelerator is compiled into bitstream with an HW limit on number of vector operations.

The termination of the nonexclusive, non-transferable evaluation license of this evaluation IP core is reported in advance by the demonstrator on the RS232 terminal. The evaluation designs run again after the reset.

The evaluation package [11] includes SDK 2015.4 SW projects with source code for MicroBlaze processor and ARM processor. SW projects support the family of UTIA (8xSIMD) EdkDSP accelerators for the Trenz TE0715-03-30-11 module [26] on SMT's EMC2-DP-V2 carrier board [23].

The evaluation package [11] includes SDK 2015.4 SW projects with C source code for ARM Cortex A9 processor (32bit) in standalone mode, C source code for MicroBlaze and C source code for the EdkDSP PicoBlaze6 controller.

The evaluation package [11] includes these static libraries for ARM Cortex A9 processor (32bit) for standalone mode:

<b>libfmc_imageon.a</b>	SDK 2015.4 UTIA static library with interface functions for video IP cores
<b>libwal.a</b>	SDK 2015.4 UTIA static library with EdkDSP API for MicroBlaze
<b>libsh01.a</b>	SDSoC 2015.4 static library for HW accelerator in project sh01
<b>libsh02.a</b>	SDSoC 2015.4 static library for HW accelerator in project sh02
<b>libsh03.a</b>	SDSoC 2015.4 static library for HW accelerator in project sh03
<b>libmd01.a</b>	SDSoC 2015.4 static library for HW accelerator in project md01

These libraries have no time restriction. Source code of these libraries is not provided in this evaluation package.

The evaluation package [11] includes these binary applications for Ubuntu:

<b>edkdspcp</b>	EdkDSP C pre-processor binary for Ubuntu in VMware Workstation 12 Player.
<b>edkdspcc</b>	EdkDSP C compiler binary for Ubuntu in VMware Workstation 12 Player.
<b>edkdspasm</b>	EdkDSP ASM compiler binary for Ubuntu in VMware Workstation 12 Player.

These binary applications have no time restriction. The user of the evaluation package has nonexclusive, non-transferable license from UTIA to use these utilities for compilation of the firmware for the Xilinx

PicoBlaze6 processor inside of the UTIA EdkDSP accelerators in precompiled designs. The source code of these compilers is owned by UTIA and it is not provided in the evaluation package.

The evaluation package [11] includes demonstration firmware in C source code for the Xilinx PicoBlaze6 processor for the family of UTIA EdkDSP accelerators for the Trezz TE0715-03-30-1I module [26] on SMT's EMC2-DP-V2 carrier board [23].

The evaluation package also includes compiled versions of this firmware in form of header files .h. These compiled firmware files can be used for initial test of the UTIA EdkDSP accelerators on the Trezz Electronic TE0715-03-30-1I module [26] on the SMT's EMC2-DP-V2 carrier board [23] without the need to install the UTIA compiler binaries and the Ubuntu image under the VMware Workstation 12 Player.

On email request to [kadlec@utia.cas.cz](mailto:kadlec@utia.cas.cz) , UTIA will send DVD with the Ubuntu image with pre-installed compiler binary files free of charge. The image can be played in the VMware Workstation 12 Player.

HW boards are not part of deliverables. HW can be ordered separately from [22], [23], [26] and [32].

Any and all legal disputes that may arise from or in connection with the use, intended use of or license for the software provided hereunder shall be exclusively resolved under the regional jurisdiction relevant for UTIA AV CR, v. v. i. and shall be governed by the law of the Czech Republic. See also the Disclaimer section.

## Vivado projects with the evaluation version of the (8xSIMD) EdkDSP IP for the Artemis EMC2 project partners.

This evaluation package includes Vivado 2015.4 projects for the Trenz Electronic TE0715-03-30-11 module [26] located on the SMT's EMC2-DP-V2 carrier [23] with the FMC card [32] with the evaluation version of the (8xSIMD) EdkDSP accelerator IP for the partners in the Artemis EMC2 project can be ordered from UTIA AV CR, v.v.i., by email request for quotation to [kadlec@utia.cas.cz](mailto:kadlec@utia.cas.cz).

UTIA AV CR, v.v.i., will provide to the EMC2 project partner quotation by email. After confirmation of the quotation by the customer, UTIA AV CR, v.v.i., will send to the customer this invoice:

The Vivado 2015.4 projects for the Trenz TE0715-03-30-11 module [26] located on the SMT's EMC2-DP-V2 carrier [23] with the FMC card [32] with the evaluation version of the (8xSIMD) EdkDSP accelerator IP for the partners in the Artemis EMC2 project  
(Without VAT) 0,00 Eur

After receiving confirmation from the EMC2 project partner about the zero-invoice received, UTIA AV CR, v.v.i. will send within 5 working days by standard mail printed version of this application note together with DVD with the Deliverables described in this section.

### Deliverables:

The evaluation package for EMC2 partners includes the Vivado 2015.4 design projects which can be modified and recompiled by the EMC2 partner. The evaluation version of the UTIA (8xSIMD) EdkDSP accelerator is provided as part of the Xilinx Vivado 2015.4 design projects. Evaluation IPs included:

**bce\_fp12\_1x8\_0\_axiw\_v1\_10\_c** Netlist of the evaluation version of the AXI-lite interface  
**bce\_fp12\_1x8\_40** Netlist of the evaluation version of the floating point data path  
 This netlist evaluation version of the UTIA (8xSIMS) EdkDSP accelerator has an HW limit on number of vector operations.

EMC2 project partners have nonexclusive, non-transferable license from UTIA to integrate this evaluation netlist into their own Vivado 2015.4 designs and to compile them to unlimited number of bit-streams for the Xilinx ZYNQ xc7z030-1I and xc7z030-1C devices. This nonexclusive, non-transferable license has no time restriction.

The source code of the evaluation versions of the (8xSIMS) EdkDSP accelerator is the IP core owned by UTIA and the source code of it is not provided in the evaluation package to the EMC2 partners. The UTIA (8xSIMD) EdkDSP HW accelerator IP core is compiled with an HW limit on the number of vector operations.

The termination of the nonexclusive, non-transferable evaluation license is reported in advance by the demonstrator on the RS232 terminal. The evaluation designs run again after the reset.

The evaluation package for EMC2 partners includes SDK 2015.4 SW projects with C source code for ARM Cortex A9 processor (32bit) in standalone mode, C source code for MicroBlaze and C source code for the EdkDSP PicoBlaze6 controller.

The evaluation package for EMC2 project partners includes these static libraries for ARM Cortex A9 processor (32bit) for standalone mode:

**libfmc\_imageon.a** SDK 2015.4 UTIA static library with interface functions for video IP cores  
**libwal.a** SDK 2015.4 UTIA static library with EdkDSP API for MicroBlaze  
**libsh01.a** SDSoC 2015.4 static library for HW accelerator in project sh01  
**libsh02.a** SDSoC 2015.4 static library for HW accelerator in project sh02

**libsh03.a**                   SDSoC 2015.4 static library for HW accelerator in project sh03  
**libmd01.a**                   SDSoC 2015.4 static library for HW accelerator in project md01

These libraries have no time restriction. Source code of these libraries is not provided in this evaluation package.

The evaluation package for EMC2 partners includes SDK 2015.4 SW projects with source code for MicroBlaze processor and ARM processor. SW projects support the family of UTIA (8xSIMD) EdkDSP accelerators for the Trenz Electronic TE0715-03-30-1I module [26] on SMT's EMC2-DP-V2 carrier board [23].

The evaluation package for EMC2 partners includes these binary applications for Ubuntu:

**edkdsppp**           EdkDSP C pre-processor binary for Ubuntu in VMware Workstation 12 Player.  
**edkdspc**           EdkDSP C compiler binary for Ubuntu in VMware Workstation 12 Player.  
**edkdspasm**        EdkDSP ASM compiler binary for Ubuntu in VMware Workstation 12 Player.

These binary applications have no time restriction. The user of the evaluation package has nonexclusive, non-transferable license from UTIA to use these utilities for compilation of the firmware for the Xilinx PicoBlaze6 processor inside of the UTIA EdkDSP accelerators in precompiled designs. The source code of these compilers is owned by UTIA and it is not provided in the evaluation package.

The evaluation package for EMC2 partners includes demonstration firmware in C source code for the Xilinx PicoBlaze6 processor for the family of UTIA EdkDSP accelerators for the Trenz Electronic TE0715-03-30-1I module [26] on SMT's EMC2-DP-V2 carrier board [23].

The evaluation package for EMC2 project partners also includes compiled versions of this firmware in form of header files .h. These compiled firmware files can be used for initial test of the UTIA EdkDSP accelerators on the Trenz TE0715-03-30-1I module [26] on the SMT's EMC2-DP-V2 carrier board [23] without the need to install the UTIA compiler binaries and the Ubuntu image under the VMware Workstation 12 Player.

On email request to [kadlec@utia.cas.cz](mailto:kadlec@utia.cas.cz) , UTIA will send DVD with the Ubuntu image with pre-installed compiler binary files free of charge. The image can be played in the VMware Workstation 12 Player.

HW boards are not part of deliverables. HW can be ordered separately from [22], [23], [26] and [32].

Any and all legal disputes that may arise from or in connection with the use, intended use of or license for the software provided hereunder shall be exclusively resolved under the regional jurisdiction relevant for UTIA AV CR, v. v. i. and shall be governed by the law of the Czech Republic. See also the Disclaimer section.

## Vivado projects with the release version of the (8xSIMD) EdkDSP IP

This release package includes Vivado 2015.4 projects for the Trenz Electronic TE0715-03-30-1I module [26] located on the SMT's EMC2-DP-V2 carrier [23] with the FMC card [32] with the release version of the (8xSIMD) EdkDSP accelerator IP with no HW limit on number of vector operations can be ordered by a customer from UTIA AV CR, v.v.i., by sending email request for quotation to [kadlec@utia.cas.cz](mailto:kadlec@utia.cas.cz). UTIA AV CR, v.v.i., will provide quotation by email. After confirmation of the quotation by the customer, UTIA AV CR, v.v.i., will send to the customer this invoice:

Vivado 2015.4 projects for the Trenz TE0715-03-30-1I module [26] located on the SMT's EMC2-DP-V2 carrier [23] with the FMC card [32] with the release version of the (8xSIMD) EdkDSP accelerator IP with no HW limit on number of vector operations.

(Without VAT)

400,00 Eur

After receiving payment, UTIA AV CR, v.v.i. will send to the customer within 5 working days (by standard mail) the printed version of the application note together with a DVD with deliverables described in this section.

### Deliverables:

The release package includes the Vivado 2015.4 design projects which can be modified and recompiled by the customer. Release IPs included:

<b>bce_fp12_1x8_0_axiw_v1_10_c</b>	Release netlist of the evaluation version of the AXI-lite interface
<b>bce_fp12_1x8_40</b>	Release netlist of the evaluation version of the floating point data path
<b>bce_fp12_1x8_30</b>	Release netlist of the evaluation version of the floating point data path
<b>bce_fp12_1x8_20</b>	Release netlist of the evaluation version of the floating point data path
<b>bce_fp12_1x8_10</b>	Release netlist of the evaluation version of the floating point data path

This release netlist versions of the UTIA (8xSIMS) EdkDSP accelerators have **no HW limit on number of vector operations**. The customer has a nonexclusive, non-transferable license from UTIA to integrate these netlists into own Vivado 2015.4 designs and to compile these netlists to an unlimited number of bit-streams for designs for the Xilinx ZYNQ xc7z030-1I and xc7z030-1C devices. This nonexclusive, non-transferable license has no time restriction. The source code of the (8xSIMD) EdkDSP accelerator IP is owned by UTIA and it is not provided in the release package to the customer.

The release package includes SDK 2015.4 SW projects with C source code for ARM Cortex A9 processor (32bit) in standalone mode, C source code for MicroBlaze and C source code for the EdkDSP PicoBlaze6 controller

The release package includes these static libraries for ARM Cortex A9 processor (32bit) for standalone mode:

<b>libfmc_imageon.a</b>	SDK 2015.4 UTIA static library with interface functions for video IP cores
<b>libwal.a</b>	SDK 2015.4 UTIA static library with EdkDSP API for MicroBlaze
<b>libsh01.a</b>	SDSoC 2015.4 static library for HW accelerator in project sh01
<b>libsh02.a</b>	SDSoC 2015.4 static library for HW accelerator in project sh02
<b>libsh03.a</b>	SDSoC 2015.4 static library for HW accelerator in project sh03
<b>libmd01.a</b>	SDSoC 2015.4 static library for HW accelerator in project md01

These libraries have no time restriction. Source code of these libraries is not provided in the release package.

The release package includes SDK 2015.4 SW projects with source code for MicroBlaze processor and ARM processor. SW projects support the family of UTIA (8xSIMD) EdkDSP accelerators for the Trenz TE0715-03-30-1I module [26] on SMT's EMC2-DP-V2 carrier board [23].

The release package includes these binary applications for Ubuntu:

**edkdsppp**      EdkDSP C pre-processor binary for Ubuntu in VMware Workstation 12 Player.  
**edkdsppc**      EdkDSP C compiler binary for Ubuntu in VMware Workstation 12 Player.  
**edkdspasm**     EdkDSP ASM compiler binary for Ubuntu in VMware Workstation 12 Player.

These binary applications have no time restriction. The user of the evaluation package has nonexclusive, non-transferable license from UTIA to use these utilities for compilation of the firmware for the Xilinx PicoBlaze6 processor inside of the UTIA EdkDSP accelerators in precompiled designs. The source code of these compilers is owned by UTIA and it is not provided in the evaluation package.

The release package includes demonstration firmware in C source code for the Xilinx PicoBlaze6 processor for the family of UTIA EdkDSP accelerators for the Trenz TE0715-03-30-1I module [26] on SMT's EMC2-DP-V2 carrier board [23].

The release package also includes compiled versions of this firmware in form of header files .h. These compiled firmware files can be used for initial test of the UTIA EdkDSP accelerators on the Trenz TE0715-03-30-1I module [26] on the SMT's EMC2-DP-V2 carrier board [23] without the need to install the UTIA compiler binaries and the Ubuntu image under the VMware Workstation 12 Player.

On email request to [kadlec@utia.cas.cz](mailto:kadlec@utia.cas.cz) , UTIA will send DVD with the Ubuntu image with pre-installed compiler binary files free of charge. The image can be played in the VMware Workstation 12 Player.

HW boards are not part of deliverables. HW can be ordered separately from [22], [23], [26] and [32].

Any and all legal disputes that may arise from or in connection with the use, intended use of or license for the software provided hereunder shall be exclusively resolved under the regional jurisdiction relevant for UTIA AV CR, v. v. i. and shall be governed by the law of the Czech Republic. See also the Disclaimer section.

## Disclaimer

This disclaimer is not a license and does not grant any rights to the materials distributed herewith. Except as otherwise provided in a valid license issued to you by UTIA AV CR v.v.i., and to the maximum extent permitted by applicable law:

(1) THIS APPLICATION NOTE AND RELATED MATERIALS LISTED IN THIS PACKAGE CONTENT ARE MADE AVAILABLE "AS IS" AND WITH ALL FAULTS, AND UTIA AV CR V.V.I. HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and

(2) UTIA AV CR v.v.i. shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under or in connection with these materials, including for any direct, or any indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or UTIA AV CR v.v.i. had been advised of the possibility of the same.

### Critical Applications:

UTIA AV CR v.v.i. products are not designed or intended to be fail-safe, or for use in any application requiring fail-safe performance, such as life-support or safety devices or systems, Class III medical devices, nuclear facilities, applications related to the deployment of airbags, or any other applications that could lead to death, personal injury, or severe property or environmental damage (individually and collectively, "Critical Applications"). Customer assumes the sole risk and liability of any use of UTIA AV CR v.v.i. products in Critical Applications, subject only to applicable laws and regulations governing limitations on product liability.