Video Chain Demonstrator on Xilinx Kintex7 FPGA with EdkDSP Floating Point Accelerators

Jiri Kadlec
Institute of Information Theory and Automation
UTIA AV CR v.v.i.
Prague, Czech Republic
kadlec@utia.cas.cz

***Abstract*— this paper briefly describes basic Kintex7 FPGA video pipe infrastructure for UTIA demonstrator in the ARTEMIS JU project ALMARVI. The video pipeline is combined with the run-time reprogrammable vector floating point EdkDSP accelerators on the same FPGA chip.**

Keywords— floating point accelerators, reconfigurability, video processing, programmable logic.

I. INTRODUCTION

This paper describes the integration and design work resulting in a set of precompiled Vivado 2013.4 Kintex7 designs, integrating the HD video processing chain for the 1920x1080p60 video sensor together with 6 SIMD floating point EdkDSP accelerators optimized for DSP computation. The HDMI Video stream can be processed on a pipe of several development boards. Demo chain of 2 boards is presented in Fig. 1.

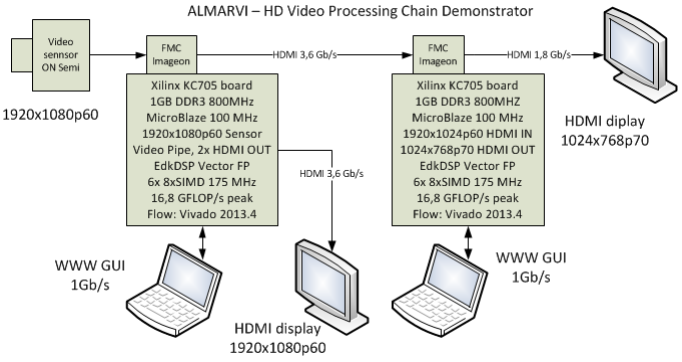

Fig. 1: ALMARVI video processing chain demonstrator on Kintext7 FPGA.

The first KC 705 board is supporting the Vita2000 video sensor, pre-processing of sensor data and the HDMI output in parallel with 6 floating point accelerators. See Fig. 2.

The second KC705 board in Fig. 2 is supporting the HDMI input, the complete video frame buffer storage in the DDR3 memory, the video frame rate and the video resolution conversion again with the 6 floating point accelerators. See Fig. 3. This concepts can be extended to larger systems.

II. SYSTEM DESCRIPTION

Both presented MicroBlaze SoC systems are based on the Xilinx BIST (build in self-test) provided by Xilinx for the Kintex7 KC705 board and the Vivado 2013.4 design flow. See Fig. 2 and Fig. 3. The network HW controller is supporting 1Gbit/100Mbit/10Mbit standards with HW DMA and a SW stack based on the Xilinx LwIP library [3]. The MicroBlaze processor is controlling 6 EdkDSP floating point accelerators. Each accelerator is organised as 8xSIMD reconfigurable data path, controlled internally by an 8 bit PicoBlaze6 controller.

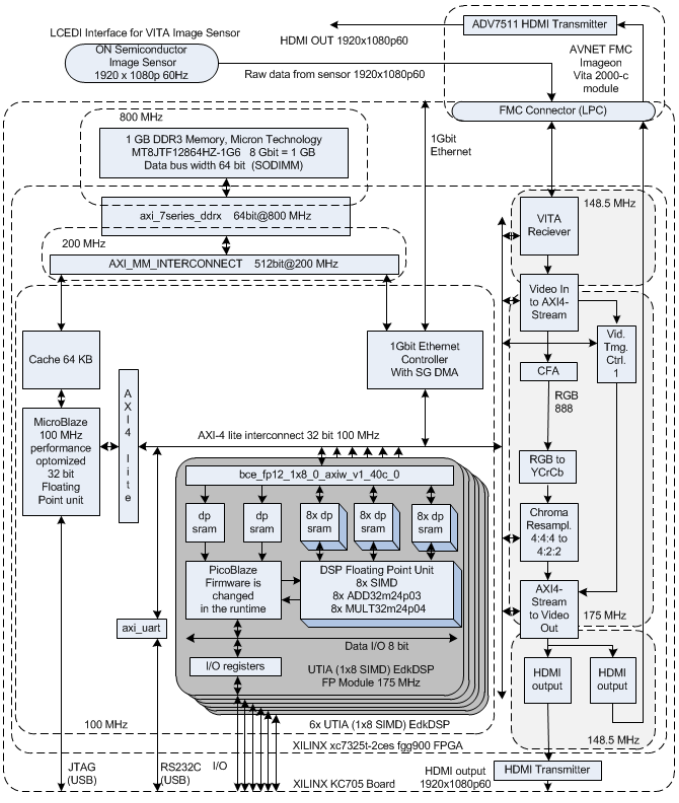

Fig. 2: Kintex7 SoC with six UTIA 8xSIMD EdkDSP vector floating point accelerators and the Vita2000 color video sensor interface.

These computing SoC are extended with the video processing chain for the Avnet Imageon FMC card [8] and the ONsemiconductors Vita2000 video sensor. See Fig. 2 and Fig. 3.

The basic video processing blocks have been designed by the Avnet for the Vivado 2013.3. Designs build on the basic building blocks provided by Xilinx within the Video and Image Processing pack [10]. The video chain also needs an IIC controller for programming of the programmable clock generator chip on the Imageon FMC card [8]. We have ported the Avnet video chain designs [9] from the Vivado 2013.3 to the Xilinx Vivado 2013.4 flow.

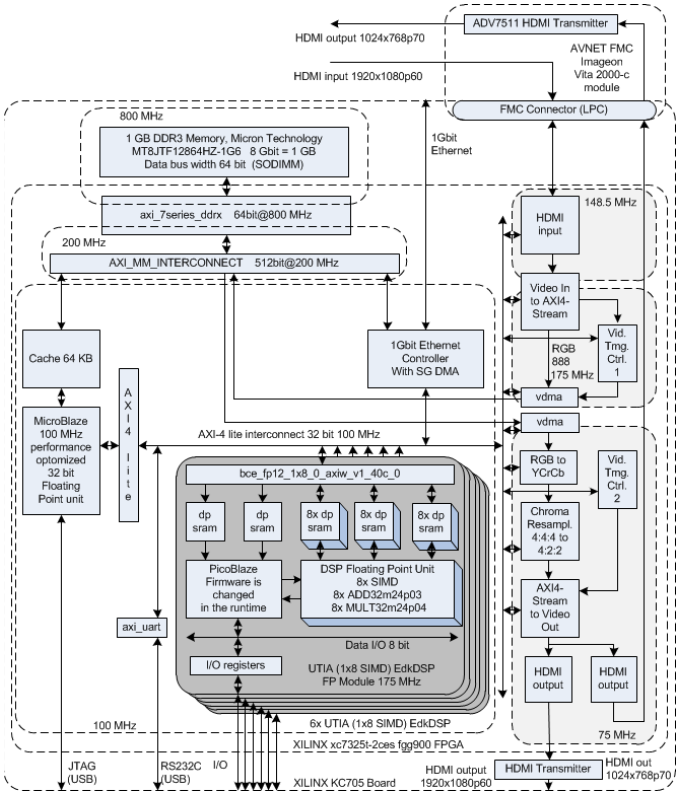

Fig. 3: Kintex7 SoC with six UTIA 8xSIMD EdkDSP vector floating point accelerators, HDMI input, HDMI output and the video frame buffer.

The video sensor Vita2000 is interfaced by the VITA Receiver block. See Fig. 2. Video output is then converted from the raw Bayer format in the color filter array interpolation block CFA. The output is already the RGB 888 HD video 1920x1080p60 corresponding to the video clock rate 148.5 MHz defined by programmable clock. The information about the video frame timing is provided in the Video Timing Controller block. See Fig. 2.

The RGB 888 color pixel data are provided in the AXI4-Stream Interconnect format, supporting the clock rate 175 MHz. This is identical clock to the 175 MHz clock domain of EdkDSP accelerators.

This will help to process some of the pixel data directly by the EdkDSP accelerators in next phase of the ALMARVI project.

The RGB 888 color pixel data AXI4-Stream Interconnect stream with 175 MHz clock is an input to the Imageon HDMI output video chain. See Fig. 2 and Fig. 3.

The RGB 888 format is converted to the YCrCb format and resampled to get 16 bit per pixel colour format for the HDMI chip on the Imageon FMC card. The AXI4-Stream to Video Out block is reconstructing the output video signal from the AXI4-Stream Interconnect data stream format. This block is taking the video timing information from the corresponding video timing controller block 1 or 2. See Fig. 2 and Fig. 3.

Video chain blocks are controlled by the MicroBlaze processor SW. Once configured and connected to the HDMI output, the chain operates autonomously. It provides the RGB 888 AXI4-Stream Interconnect stream clocked at 175MHz.

This is the place for the video processing interfaces and computations in the next stage of project development.

The 175 MHz clock is also driving the 6 EdkDSP (8xDIMD) vector floating point accelerators. See one of these accelerators on Fig. 6. Each of the six accelerators is connected as a slave to the MicroBlaze processor via the 100 MHz AXI4-lite bus.

The UTIA EdkDSP accelerator bce_fp12_1x8_0_axiw_0 has its output signals marked for debug in the Vivado 2013.4 ILA analyser.

The evaluated FIR and LMS filters have 800 coefficients and the computation is performed in 8 SIMD layers of the accelerator. This result in 8 partial floating point vector products followed by an HW supported wind up. FIR and LMS are both computed in blocks of 800 input data. Data is divided into 8 SIMD layers, each with 100 floating point numbers. The RD/WR of block data from/to the external DDR3 is performed by the MicroBlaze. RD/WR operations are performed in parallel with the computation of the accelerator.

The EdkDSP accelerator bce_fp12_1x8_0_axiw_0 is programmed by firmware and executed by the PicoBlaze6 controller inside of the accelerator. The firmware is compiled from C source code. See Fig. 4 for the FIR filter code and Fig. 5 for the LMS filter code.

III. System Performance

The floating point computation of with 800 coefficients FIR filter on the EdkDSP accelerator bce_fp12_1x8_0_axiw_0 is documented on Fig. 6. The Vivado 2013.4 ILA has been configured and compiled to store up to 32K samples of data. All signals are sampled with the clock 175 MHz. We can see the trigger point [T] in red corresponding to the execution of the C code line sending the signal to the ILA. See line 13: in Fig. 4 and Fig. 6.

The vector product of floating point vectors with the dimensions [800, 1] starts in the relative clock position 0. See Fig. 6. It takes exactly 100 clock cycles to start all 8x100=800 floating point MAC (multiply and accumulate) operations. It takes additional 59 clock cycles to finalize the final wind-up floating point additions to get the vector product result and to update data in the corresponding circular data buffers for the next step of the FIR block processing. See Fig. 6.

```
 1: #include "stdio_fp11.h"
 2: #include "a_fp1124p1.h"
 3: unsigned char i, j, n, op, led, btn;
 4: void main() {
 5:         pb2dfu_set(C_PBP_REG01, 0);
 6:         op = mb2pb_read_data();
 7:         if (op == C_DFU_OP_VVER) {
 8:                 pb2dfu_set(C_DFU_CNT, 0);
 9:                 pb2dfu_set(C_DFU_OP, op);
10:                 pb2dfu_wait4hw();
11:         } else {
12:                 n = mb2pb_read_data();
13:                 pb2dfu_set(0x20, 0);
14:                 for (i = 0; i < 4; i++) {
15:                         for (j = 2; j <= 3; j++) {
16:                                 fir(j, n, op);
17:                                 pb2mb_eoc(led);
18:                         }
19:                 }
20:                 btn = btn2pb();
21:                 led = led2pb();
22:                 pb2mb_write(C_SCREEN_0);
23:                 pb2mb_eoc(led);
24:         }
25:         pb2mb_eoc('.');
26:         pb2mb_req_reset('.');
27:         pb2mb_reset();
28:}
```

Fig. 4: C code of the FIR filter on EdkDSP accelerator.

```
 1: #include "stdio_fp11.h"
 2: #include "a_fp1124p0.h"
 3: unsigned char i, j, n, op, led, btn;
 4: void main() {
 5:         pb2dfu_set(C_PBP_REG01, 0);
 6:         op = mb2pb_read_data();
 7:         if (op == C_DFU_OP_VVER) {
 8:                 pb2dfu_set(C_DFU_CNT, 0);
 9:                 pb2dfu_set(C_DFU_OP, op);
10:                 pb2dfu_wait4hw();
11:         } else {
12:                 n = mb2pb_read_data();
13:                 pb2dfu_set(0x20, 1);
14:                 for (i = 0; i < 4; i++) {
15:                         for (j = 2; j <= 3; j++) {
16:                                 lms(j, n, op);
17:                                 pb2mb_eoc(led);
18:                         }
19:                 }
20:                 btn = btn2pb();
21:                 led = led2pb();
22:                 pb2mb_write(C_SCREEN_0);
23:                 pb2mb_eoc(led);
24:         }
25:         pb2mb_eoc('.');
26:         pb2mb_req_reset('.');
27:         pb2mb_reset();
28:}
```

Fig. 5: C code of the LMS filter on EdkDSP accelerator..

The EdkDSP accelerator bce_fp12_1x8_0_axiw_0 is using its 8 single precision floating point ADD units and 8 single precision floating point MULT units. The presented FIR stage corresponds to 1600 floating point operations. It needs 159 clock cycles. See Fig. 6. This is 10.06 floating point operations for each clock cycle. This is 1.76 GFLOP/s for the 175 MHz clock.

The ILA measurement is repeated for the 800 coefficients adaptive LMS filter floating point filter computation on the identical EdkDSP accelerator bce_fp12_1x8_0_axiw_0. See Fig. 7. The HW remains identical, but the firmware program has been dynamically changed (during the runtime) to compute the LMS filter. The differences of the source code are highlighted (C code lines 2: 13: and 16:).

The trigger point [T] corresponds to the execution of the C code line 13: It is sending signal to the ILA HW. See Fig. 5 and Fig. 7. The vector product of floating point vectors with the dimensions [800, 1] starts in the relative clock position 0. See Fig. 7. It takes again exactly 100 clock cycles to start all 8x100=800 floating point MAC (multiply and accumulate) operations. But it takes additional 455 clock cycles to compute the prediction error and to update all 800 regression coefficients and to reach the next step of the FIR block processing.

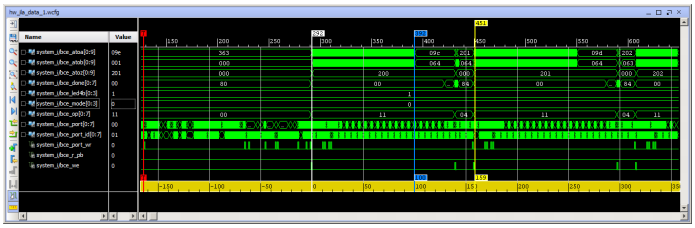

Fig. 6: Computation of the FIR filter on EdkDSP accelerator, 8x100=800 coefficients.

The EdkDSP accelerator bce_fp12_1x8_0_axiw_0 is using again its 8 single precision floating point ADD units and 8 single precision floating point MULT units. The presented LMS stage needs 2400 floating point operations. It needs 555 clock cycles. This corresponds to 4.32 floating point operations for each clock cycle. This is 756 MFLOP/s for the 175 MHz clock.

The EdkDSP accelerator performs better in case of simple 800 coefficients FIR filter. It is slightly less effective in the case of 800 coefficient LMS. This is due to an additional overhead related to internal data movement in case of LMS.

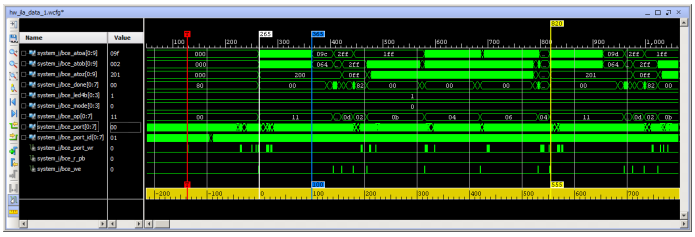

Fig. 7: Computation of the adaptive LMS filter on EdkDSP accelerator, 8x100=800 coefficients.

The synchronisation of the EdkDSP PicoBlaze6 controller with the MicroBlaze is performed by the synchronisation calls to the C function pb2mb_eoc() inside of the double for loop. See line 17: in Fig. 4 and Fig. 5. The corresponding ILA data record can be found in Fig. 8 for the FIR filter case. The FIR filter has processed the block of 8x100=800 data in cca. 16000 clock cycles. The execution of the synchronisation point function takes 395 clock cycles. After this point, the accelerator firmware can continue its FIR computation on new block of 8x100=800 data samples and the MicroBlaze code can perform in parallel the RD and WR operation to get new data from the DDR3 external memory to the MicroBlaze side of the dualported internal 8xSIMD memory structure of the EdkDSP accelerator.

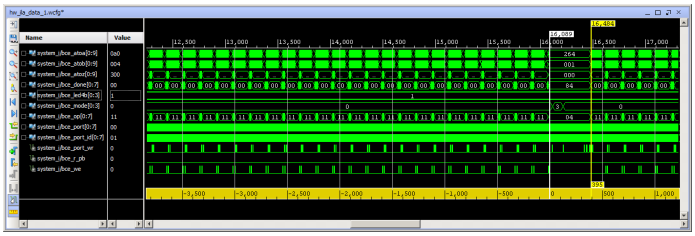

Fig. 8: Synchronisation of computation of the FIR filter on EdkDSP accelerator

The resources used by the ALMARVI video chain demonstrator design used for processing of the VITA2000 sensor are summarised in Table 1.

The floating point performance is measured for the FIR filter and the adaptive LMS filter with the maxilla possible vector length of 2000 coefficients.

Computations are performed with the accelerator firmware listed in Fig. 4 and Fig. 5 on a single 8xSIMD accelerator on 8 data sub-blocks with length 250 (8x250=2000).

TABLE I.

KC705 board, FPGA part: xc7k325t-2	Resources (complete design) See Fig. 2
Floating point paths: ADD, MUL units	48 x
Floating point paths: MAC (it is reusing the same set of 48 ADD, MUL units)	48 x
Floating point Dot Product with wind-up (it is reusing the 48 ADD, MUL units)	48 x
Floating point 8xSIMD Vector Dot Product with wind-up (it is reusing the same set of 48 ADD, MUL units)	6 x
Floating point paths: Division	6 x
Flip Flops used by the design (% of total)	21 %
LUTs used by the design (% of total)	52 %
Block RAMs used by the design	259 of 445

Table 1 describes the resources for the SoC design described in Fig. 2. It includes these main building blocks:

- Basic video processing pipeline is converting the Vita2000 video sensor data to RGB 888 175 MHz AXI4-Stream Interconnect stream and the HDMI video output for the 1920x1080p60 resolution.
- 48 single precision 3-stage-pipelined floating point add/sub units each performing up to 175 MFLOP/s
- 48 single precision 4-stage-pipelined floating point multiply units each performing up to 175 MFLOP/s
- 6 single precision 16-stage-pipelined, floating point division units each performing up to 175 MFLOP/s.
- 6 PicoBlaze6 8-bit controllers with 175 MHz system clock, each executing 87,5 Mil. Instructions/s
- Single 100 MHz 32bit MicroBlaze processor (performance optimized) with one single precision 3-stage pipelined floating point add/sub unit and one single precision 4-stage pipelined floating point multiply unit, 32 KB data cache and 32 KB instruction cache.

Presented FPGA designs are running on the Xilinx KC705 development board [1], [2]. It is using the 1 GB DDR3 memory with clock signal 800 MHz. The DDR3 is connected to Xilinx Kintex7 xc7k325t-2 FPGA by 64 bit wide data path.

The maximal theoretical peak performance of this external DDR3 memory with 800 MHz clock is 1600 (DDR3 transactions) x 8(bytes) = 12,8 GB/s. This is 102,4 Gb/s.

Each (8xSIMD) EdkDSP floating point accelerator subsystem contains one reprogrammable Xilinx PicoBlaze6 8-bit controller and the floating point (8xSIMD) DSP unit. The performance of the accelerator is application specific. In this demo, a single (8xSIMD) EdkDSP unit is delivering sustained 1403 MFLOP/s in case of 2000 tap FIR filter computation and 1012 MFLOP/s in case of the adaptive 2000 tap LMS filter identification demo. Design has six (8xSIMD) EdkDSP units. Each Xilinx PicoBlaze6 processor has 64 Bytes scratch pad RAM memory and the interrupt vector in the address 1023.

PicoBlaze6 controllers are configured with 2 program memory blocks. Each program memory block has 4096 (18bit wide) words. Both program memories are accessible by MicroBlaze processor via AXI-lite bus. MicroBlaze applications can write new firmware to the currently unused program memory, while the PicoBlaze6 is executing firmware from the second program memory. The peak performance of data memories of all six (8xSIMD) EdkDSP accelerators in the design is: 175 MHz (clock) x 4(bytes) x 3(mems) x 8(simd) x 6(instances) = 100.8 GB/s. This is 806.4 Gb/s.

IV. Conclusions

Basic video processing pipeline from the Vita2000 video sensor to RGB 888 175 MHz AXI4-Stream Interconnect stream and to the HDMI video output has been combined with 6 reprogrammable, floating point accelerators for computing on the Kintex7 28 nm chip.

Demo implementation of an adaptive acoustic noise cancellation is computing on 1 of the 6 accelerators the recursive adaptive LMS algorithm for identification of regression filter with 2000 coefficients in single precision floating point arithmetic with this sustained performance:

- 1012,0 MFLOP/s on a single 175 MHz (8xSIMD) EdkDSP accelerator (only 1 of the 6 units is used).
- 7,6 MFLOP/s on the 100 MHz MicroBlaze processor with the floating point HW unit.

The EdkDSP accelerators can be reprogrammed by the firmware. The programming is possible in C with the use of the UTIA EDKDSP C compiler. Accelerators can be programmed with two firmware programs. Designs can swap the firmware in the real time.

The alternative firmware can be downloaded to the EdkDSP accelerators from the internet in parallel with the execution of the current firmware. This is demonstrated by the download of firmware by the TFTP server and by swap of the firmware for the FIR filter room-response to the firmware for the adaptive LMS identification of the filter coefficients in the acoustic noise cancellation demo.

The EdkDSP accelerator is providing single-precision floating point results bit-exact identical to the reference software implementations running on the MicroBlaze with the Xilinx HW single precision floating point unit.

Single 175 MHz (8xSIMD) EdkDSP accelerator is 132 x faster than computation on the performance optimized 100 MHz MicroBlaze with HW floating point unit, in the presented case of the 2000 tap adaptive LMS filter. It is 184 x faster in case of the 2000 tap adaptive FIR filter.

The floating point 2000 tap coefficients FIR filter (acoustics room model) is computed by single 175 MHz (8xSIMD) EdkDSP accelerator with the floating point performance of 1.4 GFLOP/s.

The peak performance of a single 175 MHz (8xSIMD) EdkDSP accelerator is 2.8 GFLOP/s. The peak performance of the six 175 MHz (8xSIMD) EdkDSP accelerators implemented in this demo design is 16.8 GFLOP/s.

An application note describing programming and use of the EdkDSP 8xSIMD accelerators for the Xilinx KC705 board and the related Vivado 2013.4 based evaluation package can be downloaded from the UTIA www server [7].

Acknowledgment

This work has been partially supported by the Artemis JU project ALMARVI "Algorithms, Design Methods, and Many-Core Execution Platform for Low-Power Massive Data-Rate Video and Image Processing", project number Artemis JU 621439 and 7H14004 (Ministry of Education Youth and Sports of the Czech Republic [6].

References

[1] KC705 Evaluation Board for the Kintex-7 FPGA User Guide UG810 (v1.5), July 11, 2014. http://www.xilinx.com/support/documentation/boards_and_kits/kc705/ug810_KC705_Eval_Bd.pdf

[2] KC705 Built-In Self Test Flash Application, December 2013. http://www.xilinx.com/support/documentation/boards_and_kits/kc705/2013_4/xtp195-kc705-bist-c-2013-4.pdf

[3] LightWeight IP (lwIP) Application Examples, Author: Anirudha Sarangi and Stephen MacMahon; XAPP1026 (v3.2); October 28, 2012. http://www.xilinx.com/support/documentation/application_notes/xapp1026.pdf

[4] LightWeight IP Application Examples. Author: Anirudha Sarangi, Stephen MacMahon, and Upender Cherukupaly, XAPP1026 (v5.0) October 24, 2014. http://www.xilinx.com/support/documentation/application_notes/xapp1026.pdf

[5] PicoBlaze 8-bit Embedded Microcontroller User Guide for Extended Spartan 3 and Virtex5 FPGAs; Introducing PicoBlaze for Spartan-6, Virtex-6, and 7 Series FPGAs. UG129 June 22, 2011. http://www.xilinx.com/support/documentation/ip_documentation/ug129.pdf

[6] Artemis JU project Almarvi "Algorithms, Design Methods, and Many-Core Execution Platform for Low-Power Massive Data-Rate Video and Image Processing", project number ENIAC JU 621439. http://www.almarvi.eu http://sp.utia.cas.cz/index.php?ids=projects/almarvi

[7] Computation and Communication Blocks for Xilinx Kintex7 FPGA with UTIA EdkDSP Accelerators. Vivado 2013.4 Designs with SW Demos. http://sp.utia.cas.cz/index.php?ids=results&id=Utia_EdkDSP_Vivado_2013_4_KC705

[8] HDMI Input/Output FMC Module http://www.xilinx.com/products/boards-and-kits/1-44p3z5.html

[9] Vivado 2013.3 FMC-IMAGEON IP cores and tutorials https://www.em.avnet.com/en-us/design/drc/pages/supportanddownloads.aspx?RelatedId=442

[10] Video and Image Processing Pack http://www.xilinx.com/products/intellectual-property/ef-di-vid-img-ip-pack.html#overview

[11] Vivado Design Suite AXI Reference Guide UG1037 (v2.1) November 20, 2014http://www.xilinx.com/support/documentation/ip_documentation/axis_interconnect/v1_1/pg035_axis_interconnect.pdf