Application Note

Akademie věd České republiky
Ústav teorie informace a automatizace AV ČR, v.v.i.

Internet of Things Building Blocks for Xilinx Artix7 FPGA with UTIA EdkDSP Accelerators. Vivado 2013.4 Designs with SW demos.

Jiří Kadlec
kadlec@utia.cas.cz
phone: +420 2 6605 2216
UTIA AV CR, v.v.i.

Revision history:

Rev.	Date	Author	Description
1	5.11.2014	Jiří Kadlec	Description of precompiled Vivado 2013.4 Artix7 designs with EdkDSP accelerators and examples of use in the IoT applications.
2	17.11.2014	Jiří Kadlec	Fixed typo on page 56

Acknowledgements:

This work has been partially supported by the Eniac JU project THINGS2DO “Thin but Great Silicon 2 Design Objects”, project number ENIAC JU 621221 and 7H14007 (Ministry of Education Youth and Sports of the Czech Republic [6].

Table of contents

1. Summary

1.1 Building Blocks for the Internet of Things and the Embedded Parallel Computing

This application note describes precompiled Vivado 2013.4 Artix7 designs with the floating point EdkDSP accelerators and examples of use of several basic design objects used in the IoT applications. The MicroBlaze SoC design with the AXI bus is based on the Xilinx BIST (build in self-test) provided by Xilinx for the Artix7 AC701 board and the Vivado 2014.3 design flow. The network HW controller is supporting 1Gbit/100Mbit/10Mbit standards with HW DMA and a SW stack based on the lwIP library and the Xilinx application note XAPP1026 [3], [4]. The MicroBlaze processor is controlling 6 EdkDSP floating point accelerators. Each accelerator is organised as 8xSIMD reconfigurable data path, controlled by a PicoBlaze6 controller. This evaluation package is provided by UTIA for the Xilinx AC701 board with the 28nm Artix7 xc7a200t-2 FPGA part. This application note explains how to install and use the demonstrator on Windows7, (32 or 64 bit) and Xilinx AC701 board [1], [2]. These key features are demonstrated:

- WWW server running on Artix7 AC701 board with the lwIP stack running in RAW mode or SOCKET mode with the Xilkernel support of POSIX compatible threads.
- TFTP server running on Artix7 AC701 board with the lwIP stack running in RAW mode or SOCKET mode.
- RAM based file system with files in the DDR3 memory on the AC701 board.
- 6 reprogrammable floating point accelerators for local embedded computing on the Artix7 28nm chip.
- Demo implementation of an adaptive acoustic noise cancellation on 1 of the 6 accelerators is computing the recursive adaptive LMS algorithm for identification of regression filter with 2000 coefficients in single precision floating point arithmetic with this sustained performance
    - 761,0 MFLOP/s on a single 125 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 in the real time the firmware in only few clock cycles in the runtime.
- 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 125 MHz (8xSIMD) EdkDSP accelerator is 100x 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.
- The floating point 2000 tap coefficients FIR filter (acoustics room model) is computed by single 125 MHz (8xSIMD) EdkDSP accelerator with the floating point performance of 1126 MFLOP/s. The peak performance (only theoretical) of a single 125 MHz (8xSIMD) EdkDSP accelerator is 2 GFLOP/s.
- The peak performance of six 125 MHz (8xSIMD) EdkDSP accelerators implemented in this demo design is 12 GFLOP/s (this is only theoretical peek figure).
- This evaluation package presents two (8xSIMD) EdkDSP accelerator families: one family without pipelined floating point divider data path and one family with a single pipelined floating point divider data path. The members of both families differ by size and by supported vector floating point operations.
- The floating point applications are scheduled inside of the EdkDSP accelerator by the Xilinx PicoBlaze6 processor [5]. Each PicoBlaze6 firmware program has maximal size of 4096 (18 bit wide words).

1.2 What is included

The evaluation package includes precompiled Vivado 2013.4 Artix7 designs with floating point EdkDSP accelerators and examples related to the IoT applications in form of Xilinx SDK 2013.4 SW projects for Windows 7 (32 or 64bit):

- 8 evaluation versions of precompiled Artix7 designs. Each design contains one MicroBlaze and six instances of the EdkDSP accelerators. Each accelerator has 8xSIMD floating point data paths and programmable PicoBlaze6 controller for scheduling of floating point vector operations in the accelerator. The MicroBlaze works with 100 MHz system clock and EdkDSP acelerators use 125 MHz clock. The Microblaze processor works with 1Gb Ethernet with DMA controller and 1GB DDR3 memory. Designs are compiled in Xilinx Vivado 2013.4.
- UTIA is providing source code for the demo applications and SW projects for the Xilinx SDK 2013.4. These source code projects are compiled with the UTIA library libwal.a serving for the EdkDSP communication and the library libmfsimage.a with the initial file system supporting simple www server GUI.
- The included evaluation versions of the UTIA EdkDSP accelerators have HW limitation of maximal number of performed vector operations.
- The UTIA EDKDSPC C compiler is provided as 4 binary applications for Ubuntu in the VMware Player.
- The firmware for accelerators is provided in source code and also in format of binary files to enable the initial evaluation of the EdkDSP accelerators without the need to install the EDKDSPCC C compiler.

- UTIA partners of the Eniac THINGS2DO [6] projects, can get from UTIA the release version of Vivado 2013.4 HW design projects with the evaluation versions of the EdkDSP accelerators (in the Vivado 2013.4 IP netlist format) for free. See chapter 6 for specification of deliverables for the Eniac THINGS2DO [6] project partners with license details.

- Release versions of Vivado 2013.4 HW design projects and release version of EdkDSP accelerators for the Xilinx AC701 board is offered by UTIA. All customers can order and buy from UTIA the release version of this demo. It includes the Vivado 2013.4 HW design projects with the EdkDSP accelerators (in the Vivado 2013.4 IP netlist format) with the HW limitation of maximal number of performed vector operations removed. See sections 7 of this application note for specification of deliverables and license details.

2. Description of EdkDSP Accelerators in IoT Demonstrators

2.1 Description of EdkDSP accelerators and evaluation designs

This application note describes how to set-up and use of 8 HW designs running on one MicroBlaze processor with six (8xSIMD) EdkDSP accelerators on Xilinx AC701 board. See Figure 1 and Figure 2.

Demonstrators serve for evaluation of the IoT objects equipped with two floating point accelerator families on the Xilinx Artix7 xc7a200t-2 part:

- **bce_fp11_1x8_0_axiw_v1_[10|20|30|40]_a** is a family of four versions of floating point EdkDSP accelerators with 8 SIMD data paths.
- **bce_fp12_1x8_0_axiw_v1_[10|20|30|40]_a** is similar family of four versions of floating point EdkDSP accelerators with 8 SIMD data paths extended by a pipelined floating point division (FPDIV) in a single data path.

The four grades [10|20|30|40] of the EdkDSP accelerator differ in HW-supported vector computing capabilities:

The area optimized accelerators bce_fp11_1x8_0_axiw_v1_10_a and bce_fp12_1x8_0_axiw_v1_10_a perform vector floating point operations FPADD, FPSUB in 8 SIMD data paths.

The accelerators bce_fp11_1x8_0_axiw_v1_20_a and bce_fp12_1x8_0_axiw_v1_20_a perform vector floating point operations FPADD, FPSUB in 8 SIMD data paths plus the vector floating point MAC operations in 8 SMD data paths for length of the vector 1 up to 10. These accelerators can be used in applications like floating point matrix multiplication with row and column dimensions <= 10.

The accelerators bce_fp11_1x8_0_axiw_v1_30_a and bce_fp12_1x8_0_axiw_v1_30_a support identical operations as the bce_fp11_1x8_0_axiw_v1_20_a and bce_fp12_1x8_0_axiw_v1_20_a plus the floating point vector by vector dot products performed in 8 SIMD data paths. It is optimized for parallel computation of up to 8 FIR or LMS filters, each with size up to 255 coefficients. It is also effective in case of floating point matrix by matrix multiplications, where one of the dimensions is large (in the range from 11 to 255).

Finally, the accelerators bce_fp11_1x8_0_axiw_v1_40_a and bce_fp12_1x8_0_axiw_v1_40_a support identical operations as the bce_fp11_1x8_0_axiw_v1_30_a and bce_fp12_1x8_0_axiw_v1_30_a plus an additional HW support of dot product. It is computed in 8 data paths with the HW supported wind-up into single scalar result.

The bce_fp11 versions of 8xSIMD accelerators has no support for pipelined vector floating point division and it is suitable for applications like FIR filters or adaptive LMS filters with no need for floating point division.

The bce_fp12 versions of 8xSIMD accelerators are larger in comparison to the bce_fp11 equivalents and support in a single data path the pipelined vector floating point division. Accelerators are suitable for applications like adaptive normalised NLMS filters and the square root free versions of adaptive RLS QR filters and adaptive RLS LATTICE filters.

Figure 1: Demonstration of IoT application combining the 1 Gb ethernet, www server and TFTP server with 6x (8xSIMD) EdkDSP floating point accelerators on Xilinx AC701board with Artix7 FPGA.

Ten HW designs precompiled in Vivado 2013.4 combine MicroBlaze and six 8xSIMD EdkDSP accelerators. All designs demonstrate use of single instance of 8xSIMD EdkDSP floating point accelerator on 32bit AXI-lite bus of the Xilinx MicroBlaze soft-core processor on the Xilinx Artix7 AC701 FPGA board with system clock of MicroBlaze 100 MHz and EdkDSP accelerators 125 MHz. See Figure 2.

Common properties of precompiled Vivado 2013.4 evaluation designs:

- The EdkDSP floating point accelerators are reconfigurable during runtime by change of firmware.
- All HW evaluation designs have been compiled in Xilinx VIVADO 2013.4 with SW projects for SDK 2013.4.

Presented HW accelerators can results in better POWER per MFLOPS ratio for certain class of DSP applications in comparison to the computation on MicroBlaze with HW floating point support.

The demonstrator includes source code of set of SW demos prepared for easy import of projects and compilation in the Xilinx SDK 2013.4.

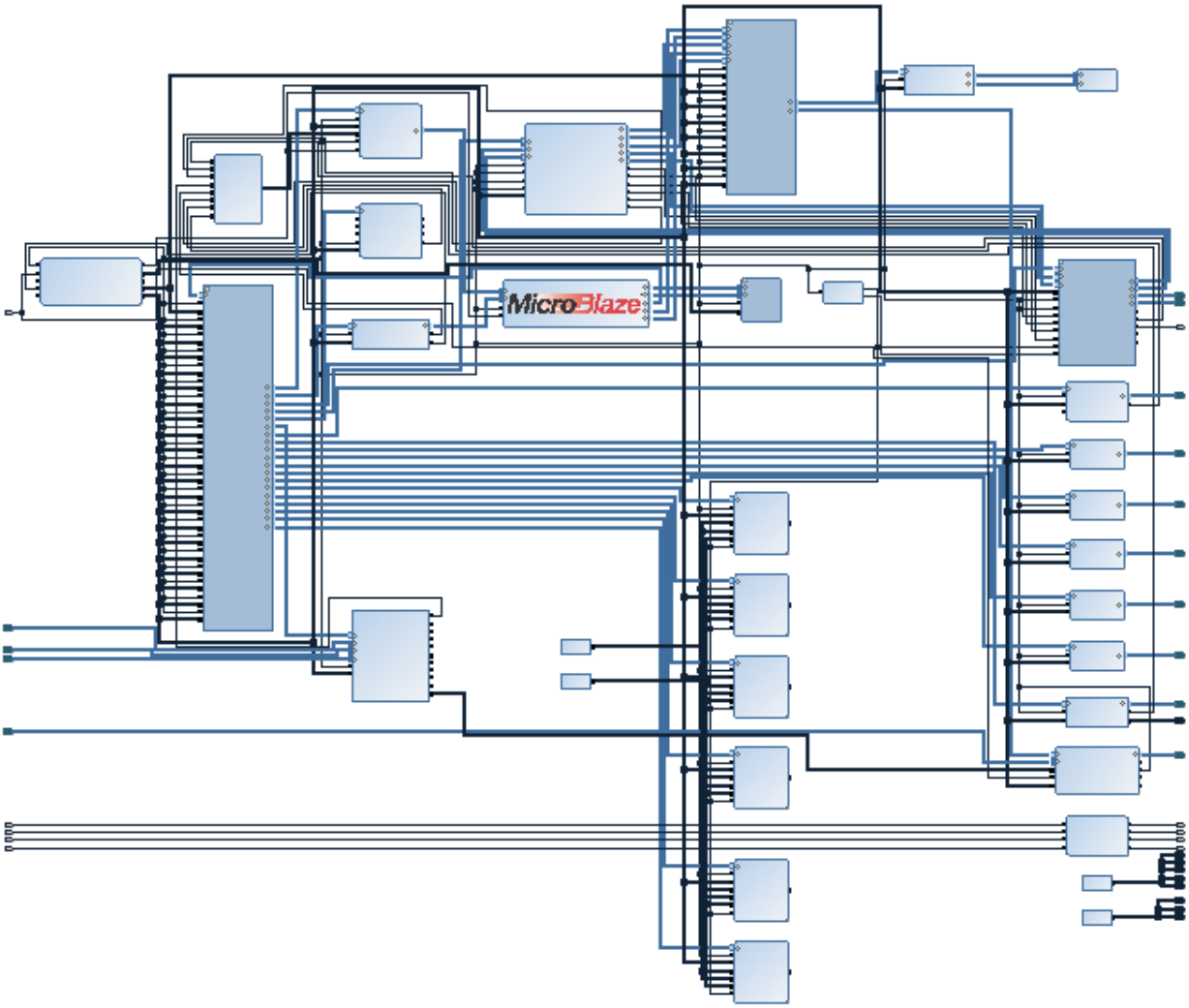

Figure 2: Design with 6 EdkDSP accelerators in Xilinx Vivado; 2013.4 IP Integrator.

Figure 2 describes the SoC with MicroBlaze, 1Gb Ethernet and six EdkDSP accelerators. The design is based on the Vivado 2013.4 AC701 BIST (built-in self-test) reference design from Xilinx.

The internal Program and data RAM memory is set to 128 KB size. Data width is 128 bit to support burst operations. The 6 EdkDSP (8xSIMD) floating point accelerators are memory mapped on the 32 bit AXI-lite bus.

Each accelerator has reserved 1 MB of address space. See Figure 3 and Figure 4 for the resources used by the designs.

2.2 Resources used by the designs

The resources used by the 10 presented designs are summarised in Figure 3 and Figure 4.

7a200t-2	fp	fp	fp	fp	fp	Design size			Performance	
	Add Mul	Mac	Dot Prod	S8 Prod	Div	FFs %	LUTs %	Bram No (of)	LMS Mflop/s	FIR Mflop/s
ac701_bist						10	21	74 (365)		
(6x) fp11_1x8_10	8x					20	52	254 (365)		
(6x) fp11_1x8_20	8x	8x				22	55	254 (365)		
(6x) fp11_1x8_30	8x	8x	8x			24	63	254 (365)		
(6x) fp11_1x8_40	8x	8x	8x	1x		24	64	254 (365)	**(6x) 761**	**(6x) 1126**

Figure 3: Resources used by MicroBlaze and 6x (8xSIMD) EdkDSP, no FP division

7a200t-2	fp	fp	fp	fp	fp	Design size			Performance	
	Add Mul	Mac	Dot Prod	S8 Prod	Div	FFs %	LUTs %	Bram No (of)	LMS Mflop/s	FIR Mflop/s
ac701_bist						10	21	74 (365)		
(6x) fp12_1x8_10	8x				1x	21	56	254 (365)		
(6x) fp12_1x8_20	8x	8x			1x	24	59	254 (365)		
(6x) fp12_1x8_30	8x	8x	8x		1x	26	68	254 (365)		
(6x) fp12_1x8_40	8x	8x	8x	1x	1x	26	69	254 (365)	**(6x) 761**	**(6x) 1126**

Figure 4: Resources used by MicroBlaze and 6x (8xSIMD) EdkDSP, with FP division

The ac701_bist design describes resources used by the MicroBlaze SoC without EdkDSP accelerators. The internal block RAM memory is set to 32KB and 128KB. Please, notice, that the Xilinx reference ac701_bist design [2] works with internal block RAM memory set to 32KB and 1MB.

All designs with EdkDSP accelerators (fp11 and fp12) work with

- 48 single precision 3-stage pipelined floating point add/sub units each performing up to 125 MFLOP/s
- 48 single precision 4-stage pipelined floating point multiply units each performing up to 125 MFLOP/s
- 6 PicoBlaze6 controllers with 125 MHz system clock, each executing 62,5 Mil. instructions/s
- The 100MHz MicroBlaze processor working with one single precision pipelined floating point add/sub unit and one single precision pipelined floating point multiply unit, 32 KB data cache and 32 KB instruction cache.

The fp12 designs work in addition with

- 6 single precision 16-stage pipelined floating point divide units each performing up to 125 MFLOP/s.

The designs use accelerators with different HW supported operation. This is reflected in the difference of resources used by the designs. See Figure 3 and Figure 4.

2.3 Use of external DDR3 memory

Presented FPGA designs are running on the Xilinx AC701 development board [1], [2]. See Figure 1. It is using the 1GB DDR3 memory with clock signal 400 MHz. The DDR3 is connected to Xilinx Artix7 xc7a200t-2 FPGA by 64 bit wide data path.

2.4 Re-programmability of EdkDSP accelerators

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 1126 MFLOP/s in case of 2000 tap FIR filter computation and 761 MFLOP/s in case of the adaptive 2000 tap LMS filter identification demo. All designs have six (8xSIMD) EdkDSP units.

The Xilinx PicoBlaze6 processor has fixed configuration with size of the program memory 4096 (18 bit wide) words, 64 Bytes scratch pad RAM memory and the interrupt vector in the address 1023.

The (8xSIMD) EdkDSP accelerator works with 2 program memories. Each program memory has 4096 (18bit wide) words. Both program memories are accessible by MicroBlaze processor via AXI-lite bus. The MicroBlaze application can write new firmware to the currently unused program memory, while the PicoBlaze6 is executing firmware from the second program memory.

2.5 Debug of the IoT evaluation designs with the EdkDSP accelerators

All EdkDSP accelerators can communicate with MicroBlaze program. The communication is using the Worker Abstraction Layer (WAL) library API. This API is used for support of writing of the debug information from the worker to the MicroBlaze terminal.

The PicoBlaze6 processors [5] can exchange data and text via the 8 bit communication data path with the MicroBlaze processor. This path is used to communicate parameters to the accelerators and to get messages or reports from accelerators for debugging. Text file with information from the accelerator can be stored in the RAM based file system of MicroBlaze. It can be downloaded to PC via Ethernet for inspection.

Floating point data are accessed by the MicroBlaze processor via the dual ported block memories of accelerators. The MicroBlaze side of the dual-ported memories is mapped into the MicroBlaze memory. The MicroBlaze processor can copy data from the dual ported memories to the DDR3 global workspace and display floating point data in the debugger. The computation in the (8xSIMD) EdkDSP units can overlap with the communication with the DDR3 performed by MicroBlaze. It is supported by data and program cache. A Ping-Pong swap of memory banks is used by the accelerator firmware. The (8xSIMD) EdkDSP firmware is computing (in parallel) in some banks of all dual ported memories and the MicroBlaze is communicating (sequentially) to/from DDR3 in another set of banks of the dual-ported memories. This process can be stopped, inspected and debugged by the MicroBlaze debugger from the SDK 2013.4.

3. Installation and use of the evaluation package

3.1 Import of precompiled HW and SW projects into Xilinx SDK 2013.4

Unzip the evaluation package to directory of your choice. The directory c:\VM_07 will be used in this application note. You will get these directories:

c:\VM_07\d_34_7a

```
01.11.2014  16:11   <DIR>      .
01.11.2014  16:11   <DIR>      ..
01.11.2014  16:09   <DIR>      d_7a200t_fp11_6x8
01.11.2014  16:09   <DIR>      d_7a200t_fp11_6x8_IMPORT
01.11.2014  16:12   <DIR>      d_7a200t_fp11_6x8_v1_10a
01.11.2014  16:12   <DIR>      d_7a200t_fp11_6x8_v1_20a
01.11.2014  16:12   <DIR>      d_7a200t_fp11_6x8_v1_30a
01.11.2014  16:12   <DIR>      d_7a200t_fp11_6x8_v1_40a
01.11.2014  16:10   <DIR>      d_7a200t_fp12_6x8
01.11.2014  16:10   <DIR>      d_7a200t_fp12_6x8_IMPORT
31.10.2014  14:25   <DIR>      d_7a200t_fp12_6x8_v1_10a
31.10.2014  14:24   <DIR>      d_7a200t_fp12_6x8_v1_20a
31.10.2014  14:24   <DIR>      d_7a200t_fp12_6x8_v1_30a
31.10.2014  14:23   <DIR>      d_7a200t_fp12_6x8_v1_
```

Select SDK 2013.4 workspace in c:\VM_07\d_34_7a \d_7a200t_fp12_6x8\SDK_Workspace.
See Figure 5.

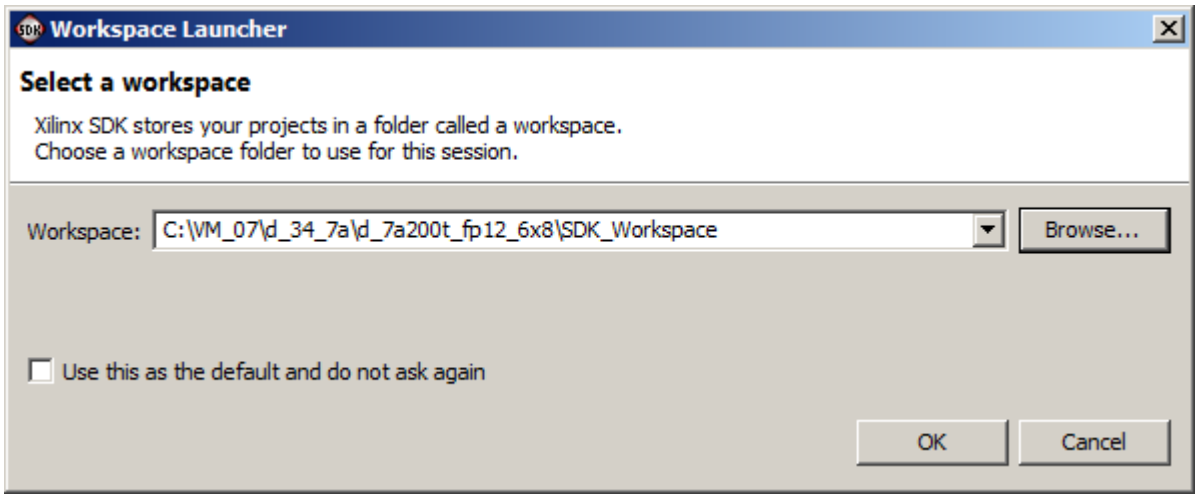

Figure 5: Select the SDK Workspace

Add **c:\VM_07\d_34_7a\d_7a200t_fp12_6x8\repo_edkdsp** path to the UTIA EdkDSP repository. See Figure 6.

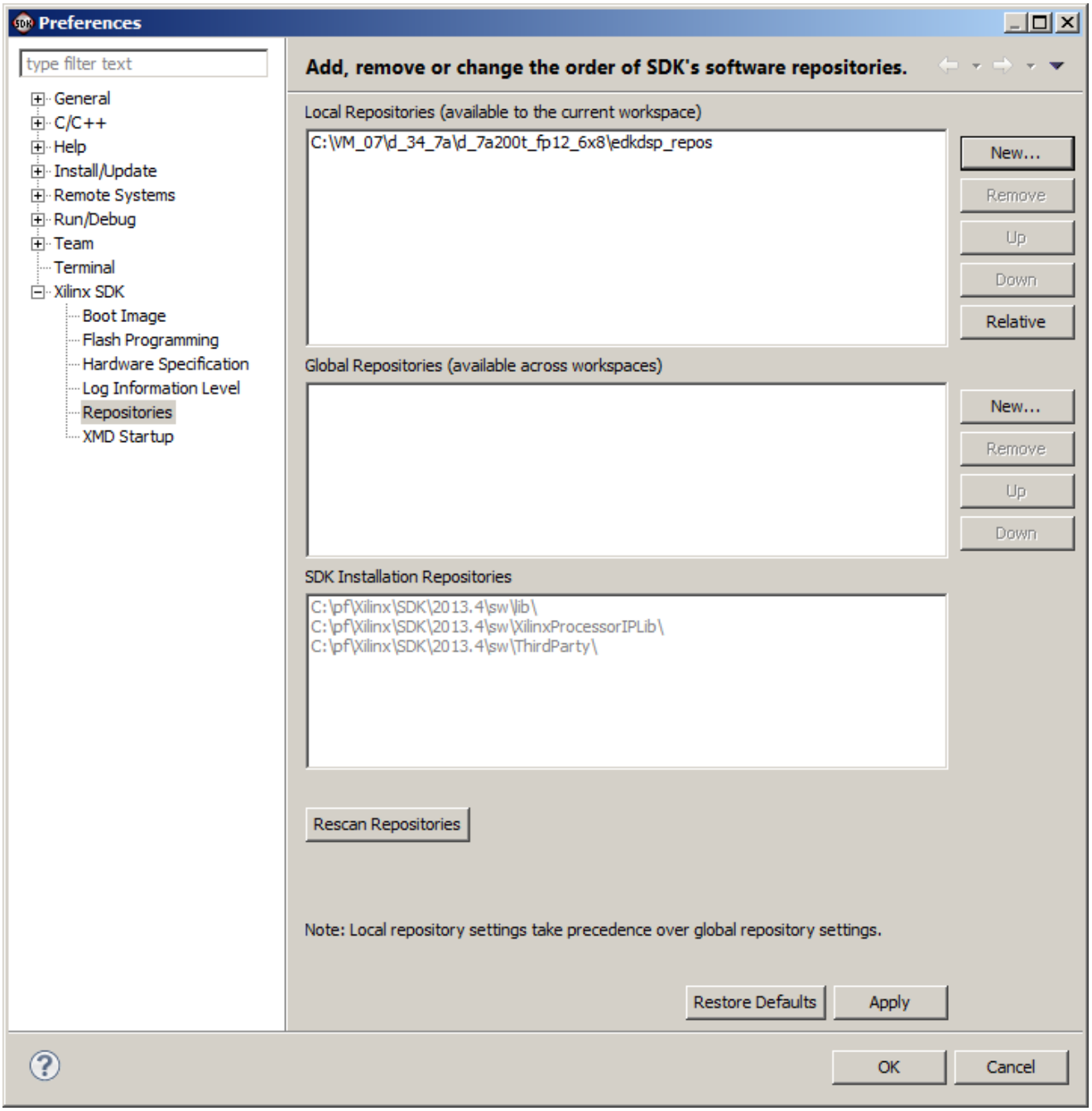

Figure 6: Include the UTIA EdkDSP Repository

Click on the “Rescan Repositories” button. Click on the “Apply button”, and finally click on the OK button. The path to the SW drivers has been defined.

In SDK, select File -> New -> Project … -> Xilinx -> Hardware Platform Specification. See Figure 7.
Click on the Next button.

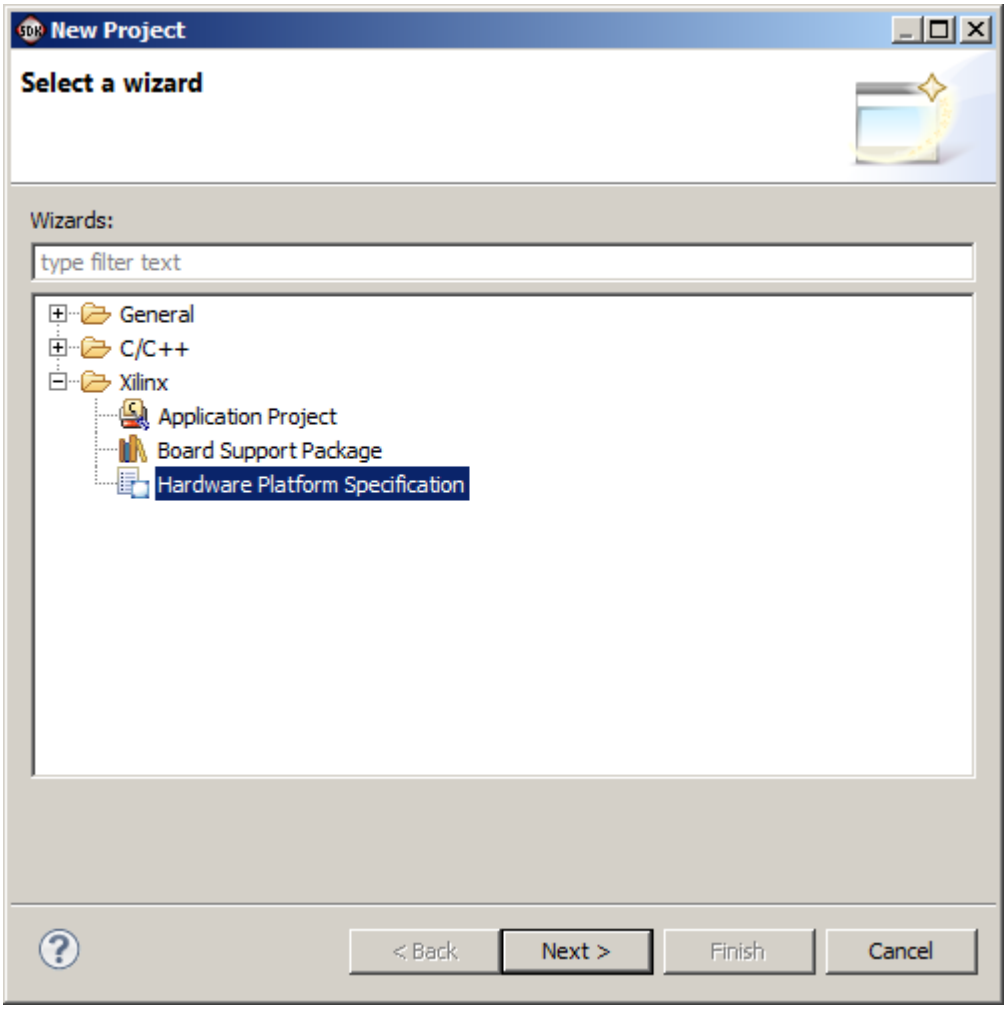

Figure 7: Specify the hardware platform

In the "New Hardware Project" screen, fill into the Project name: hw_platform_0
In the New Hardware Project screen, fill into the Target Hardware Specification:

c:\VM_07\d_34_7a\d_7a200t_fp12_6x8_v1_40a\SDK\SDK_Export\hw\system.xml

This will specify one of the 8 precompiled HW designs present in the evaluation package. See Figure 8.

We have selected the **d_7a200t_fp12_6x8_v1_40a** design, demonstrating the use of six instances the UTIA EdkDSP accelerators, all with 8xSIMD data path, with floating point single data path division. All six (8xSIMD) accelerators compiled in this design have identical capabilities defined by the IP core: bce_fp12_1x8_0_axiw_v1_40_a .

Click on "Finish" button to finalize the selection of the precompiled HW design. See Figure 8.

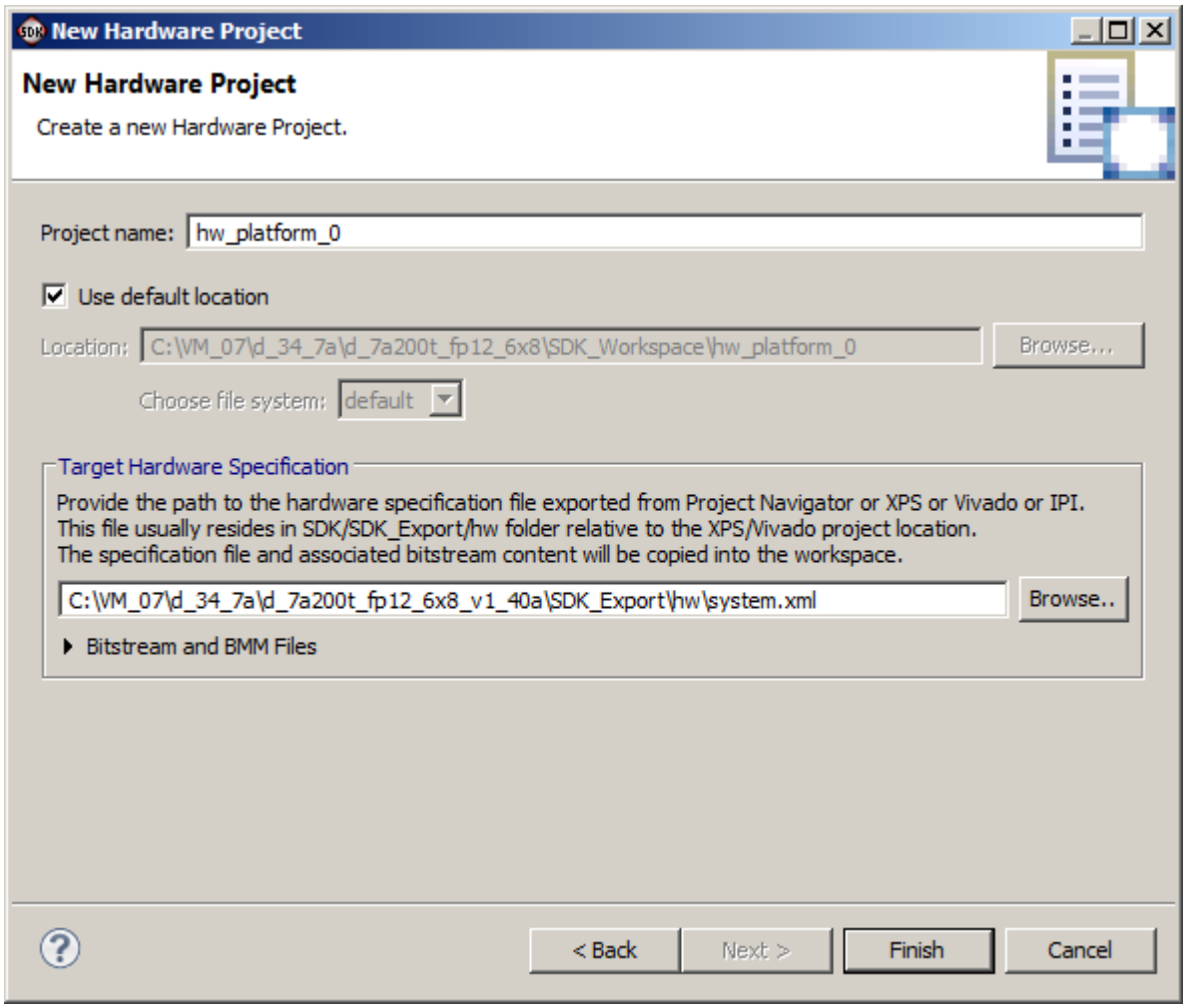

Figure 8: Use the name "hw_platform_0" and select one of the provided xml design descriptions

SDK is interpreting the system.xml and presents HW cores of in the design. See Figure 9.

The hardware platform "hw_platform_0" has been created.

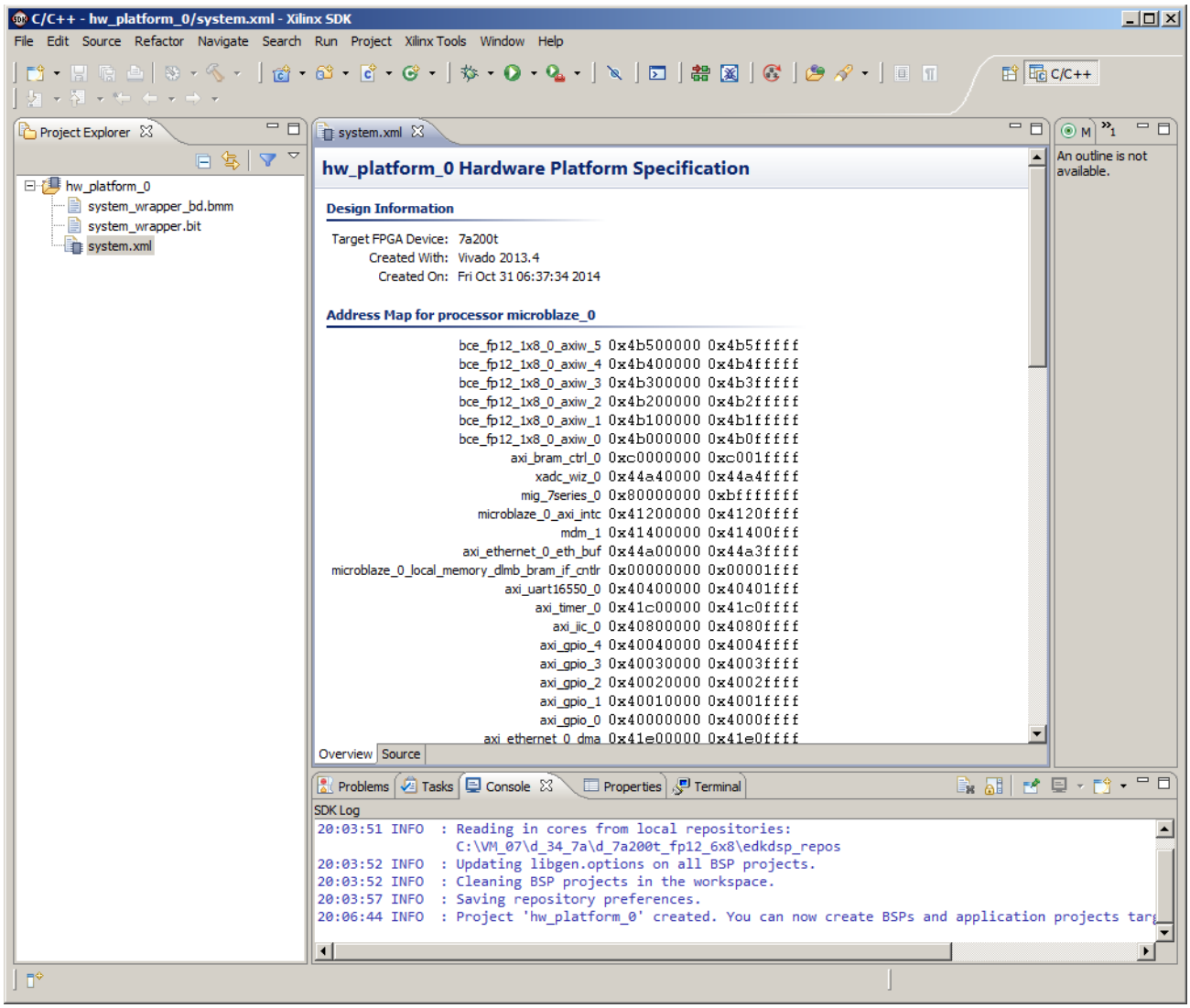

Figure 9: Hardware platform with the MicroBlaze processor and the address map

SW projects can be imported into SDK now. Select:

File -> Import -> General -> Existing Projects into Workspace
Click on Next button. See Figure 10.

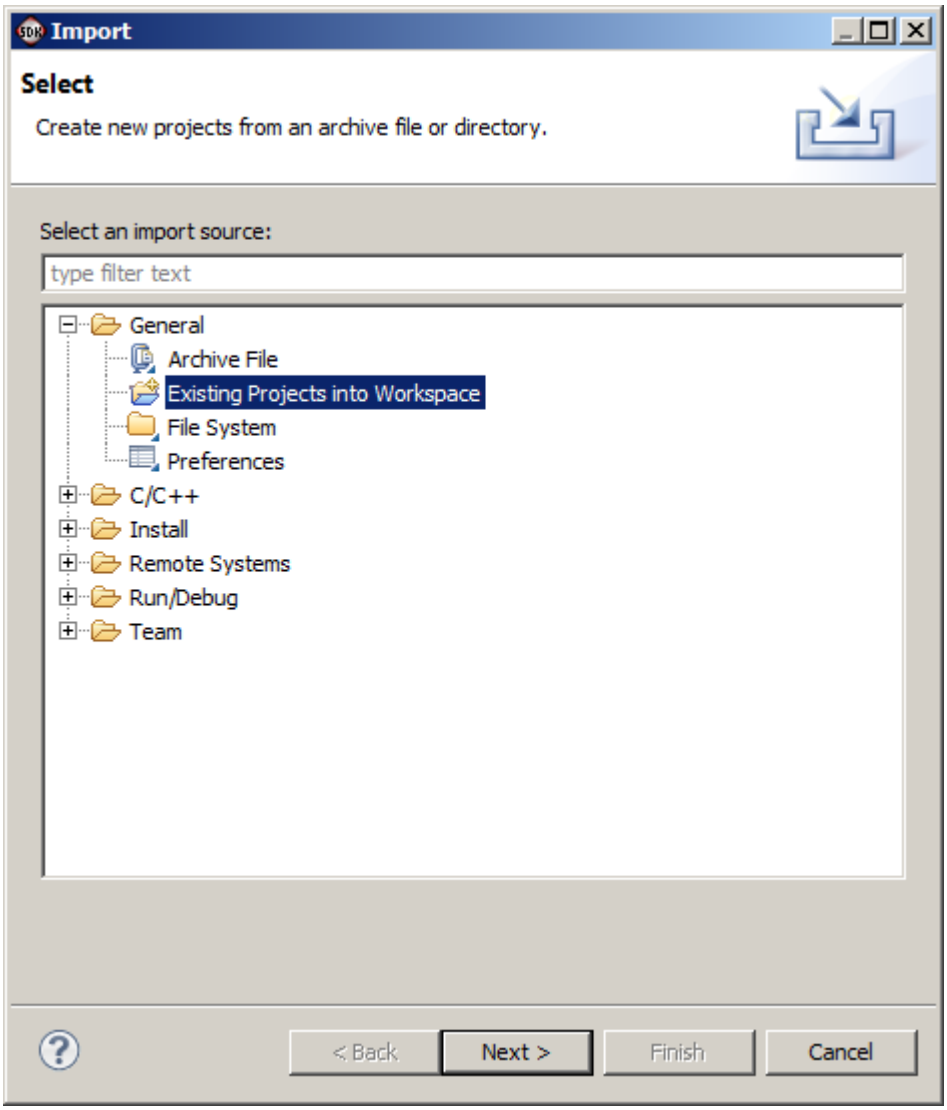

Figure 10: Import existing projects into workspace

Select the directory with projects to be imported. See Figure 11.

c:\VM_07\d_34_7a\d_7a200t_fp12_6x8_IMPORT

Set the “Copy projects into workspace” check box.
Click on Finish button. See Figure 11.

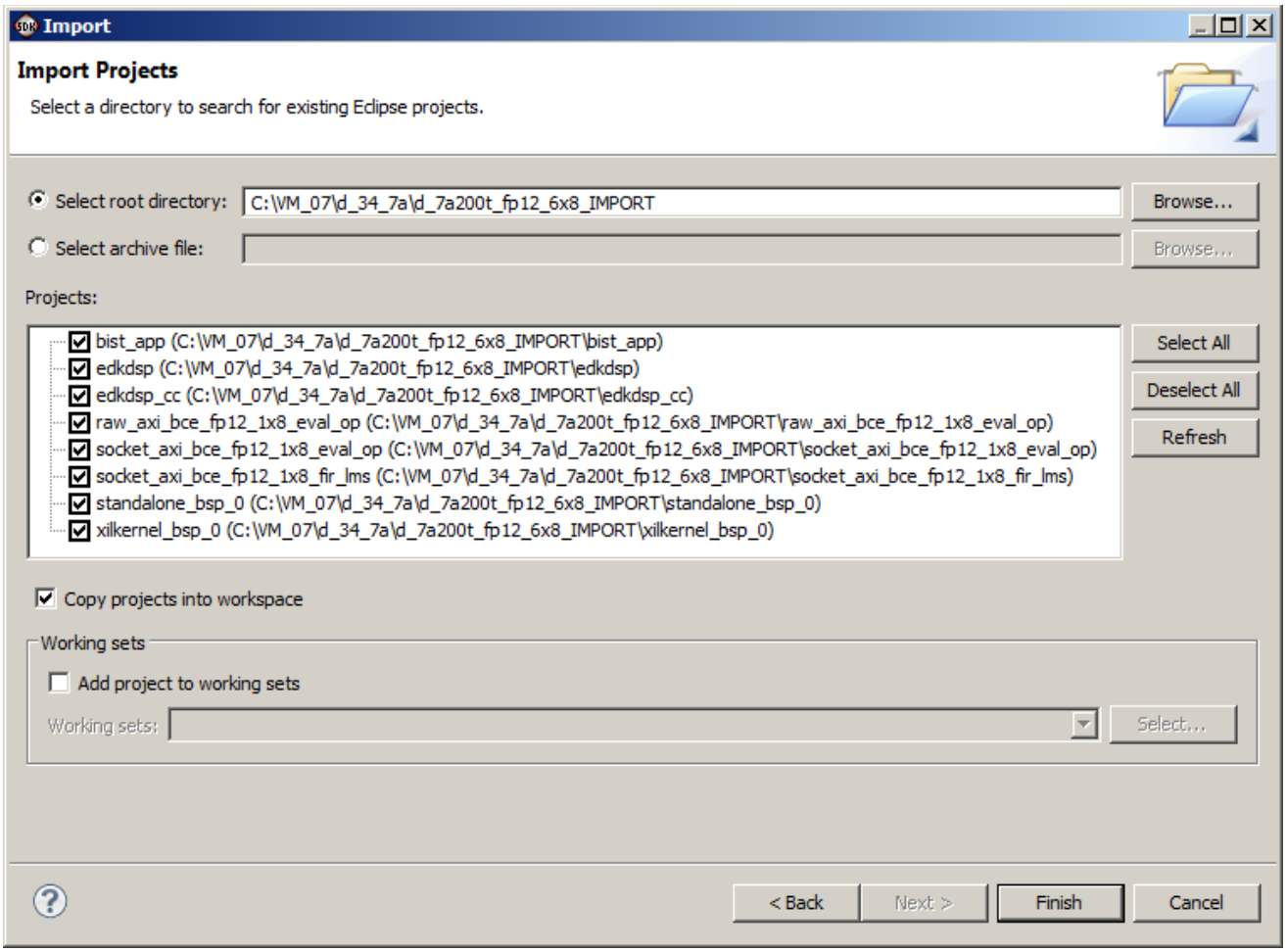

Figure 11: Select copy projects into workspace and finish the import of all projects.

All the UTIA EdkDSP SW projects are imported into SDK workspace from the directory **c:\VM_07\d_34_7a\d_7a200t_fp12_6x8_IMPORT**

Process of compilation will start automatically. This first compilation of all SDK SW projects can take several minutes to finish. It should finish without errors. See Figure 12.

3.2 Evaluation of demo projects

The “bist_app” project in the “Project Explorer” window of the SDK 2013.4 is only slightly modified version of the Xilinx BIST SW application project. The RAM memory test is adjusted for the 128 KB RAM. See Figure 12.
The “edkdsp” project is extending the “bist_app” with tests of the EdkDSP accelerator, without Ethernet.
The “raw_axi_bce_fp12_1x8_eval_op” project is extending the “edkdsp” with RAW version of the lwIP Ethernet www server GUI, the TFTP file server and the RAM based file system.
The “socket_axi_bce_fp12_1x8_eval_op” project is extending the “edkdsp” with SOCKET version of the lwIP Ethernet www server GUI, the TFTP file server and the RAM based file system.
The “socket_axi_bce_fp12_1x8_fir_lms” project is demonstrating the floating point FIR filter and LMS filter computation on a single (8xSIMD) EdkDSP accelerator with the SOCKET version of the lwIP Ethernet www server GUI, the SOCKET version of the TFTP file server and the RAM based file system.

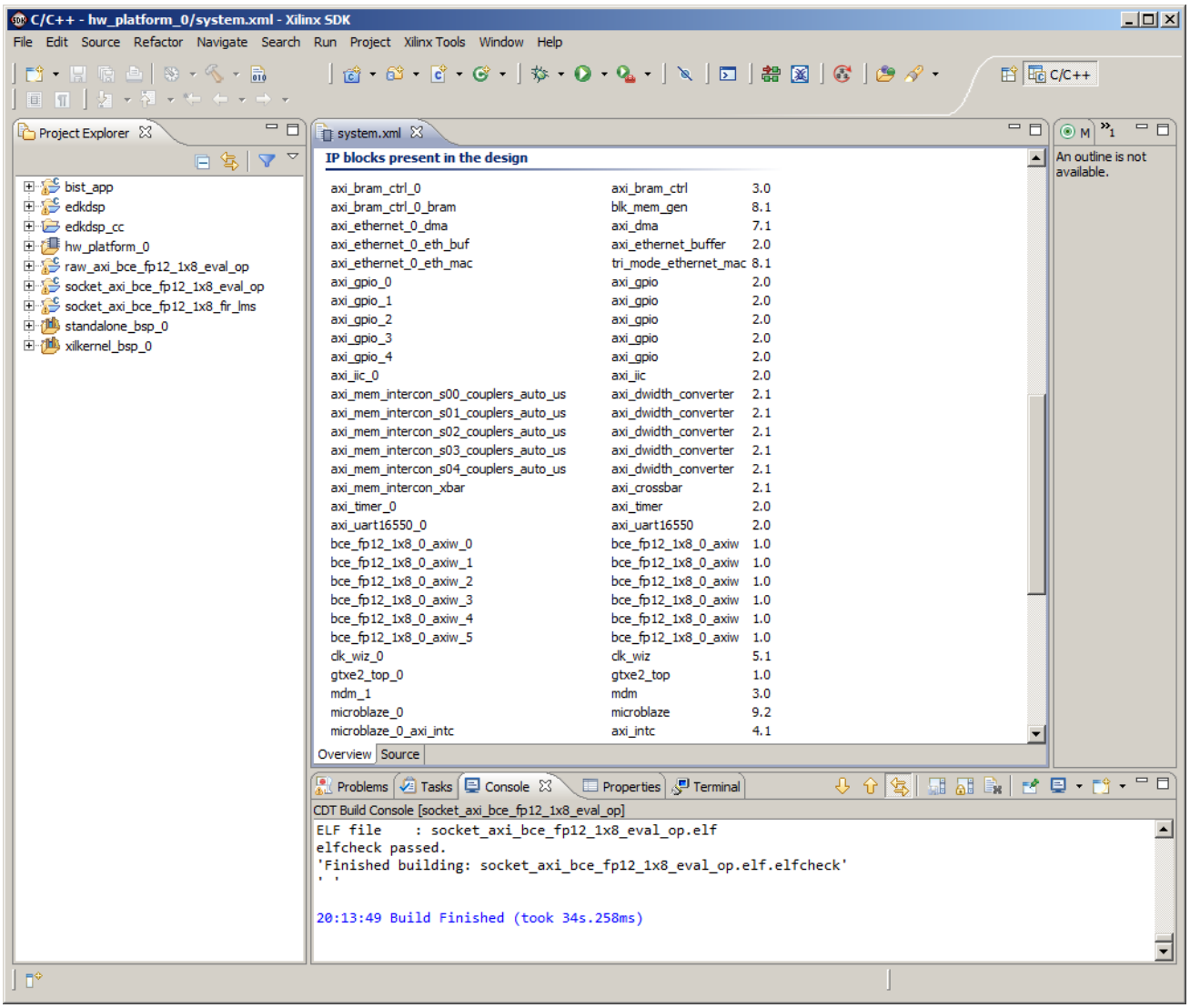

Figure 12: All projects are compiled. See IP blocks present in the design.

Connect the jtag and serial line USB cables to your AC701 board. Switch ON the board.

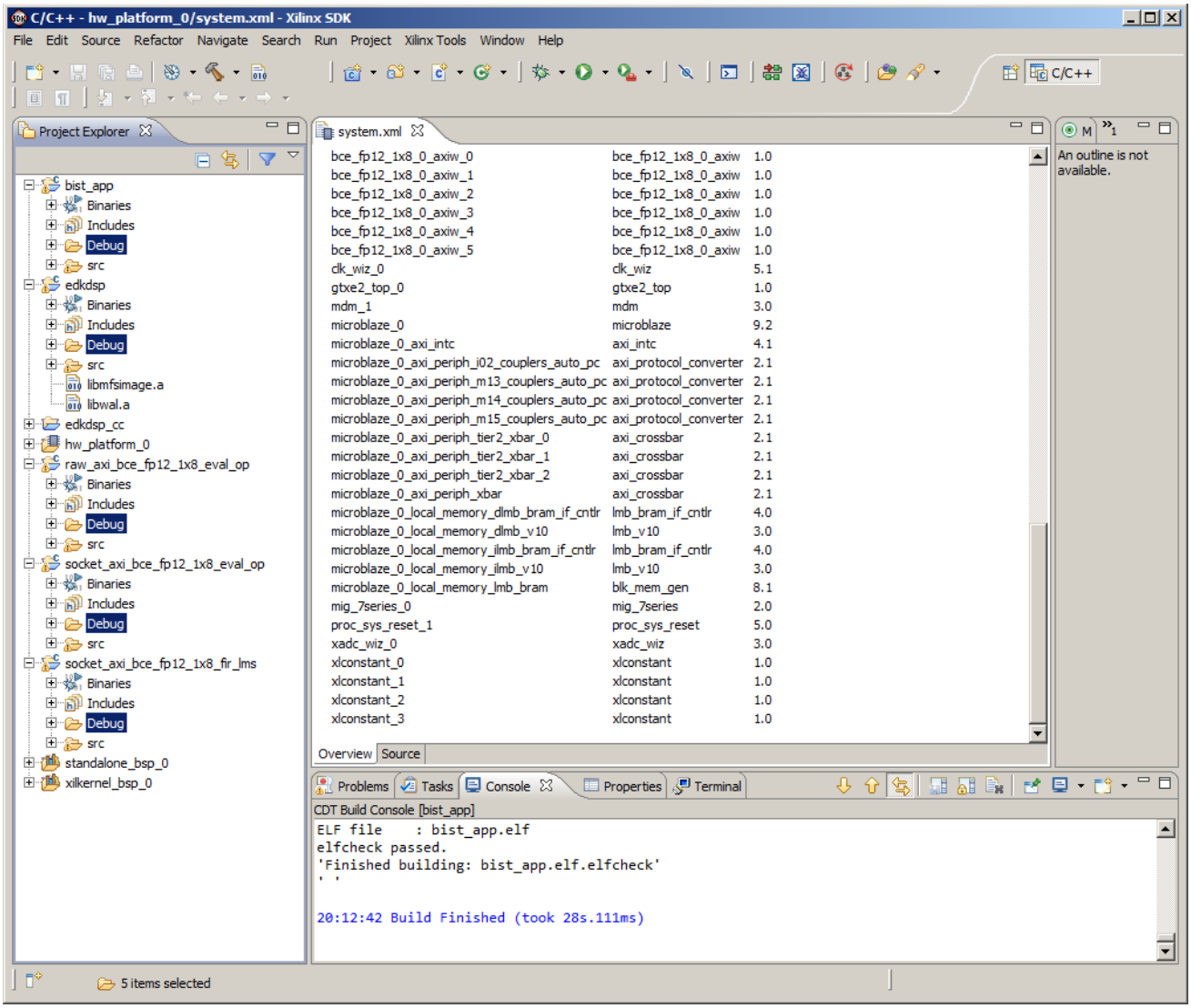

Figure 13: Set all projects for Release and delete all Debug subdirectories

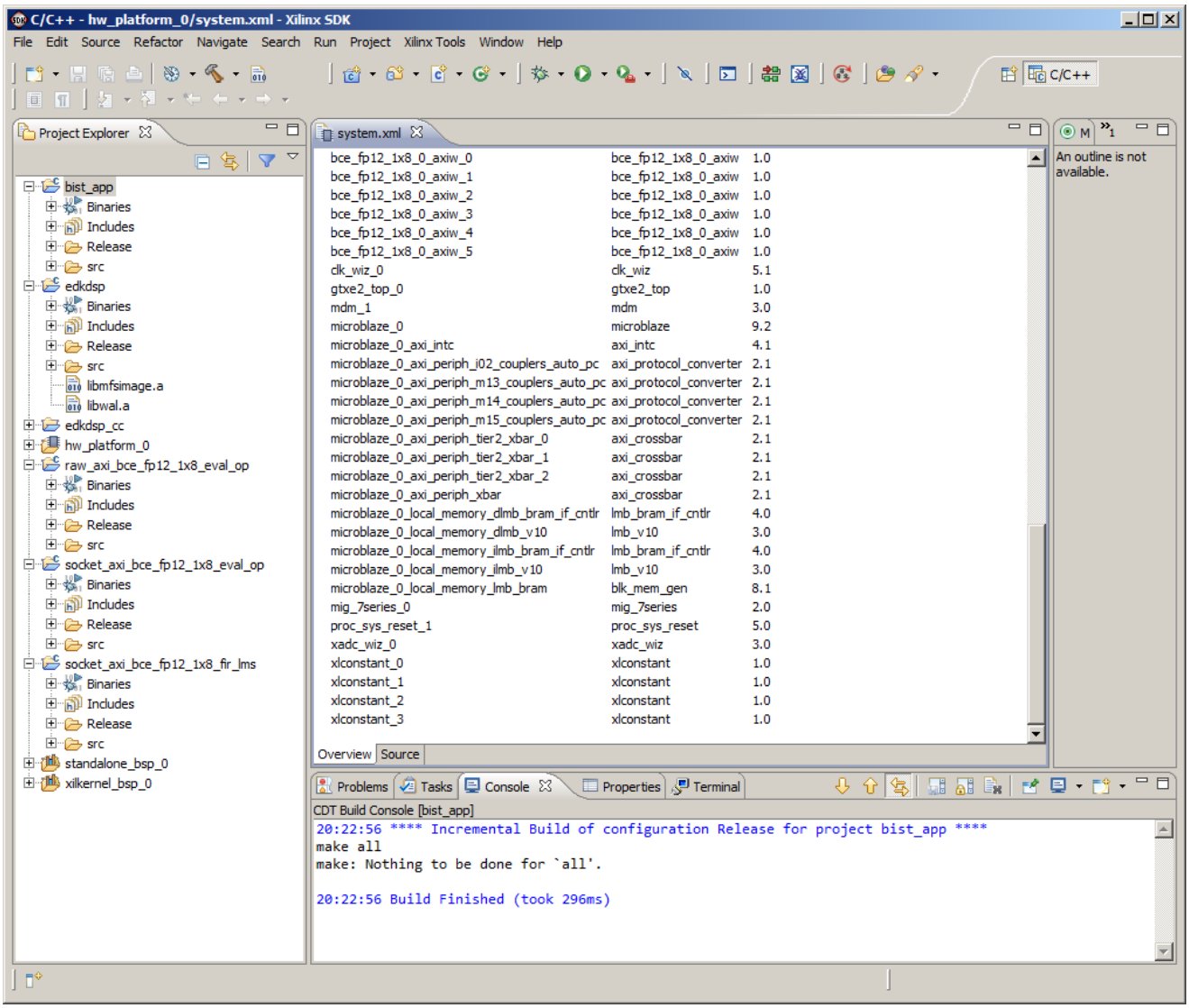

Figure 14: All projects are recompiled for release.

On PC, start PuTTY terminal. Set 9600 baud and “Flow control” to None. See Figure 15 and Figure 16.

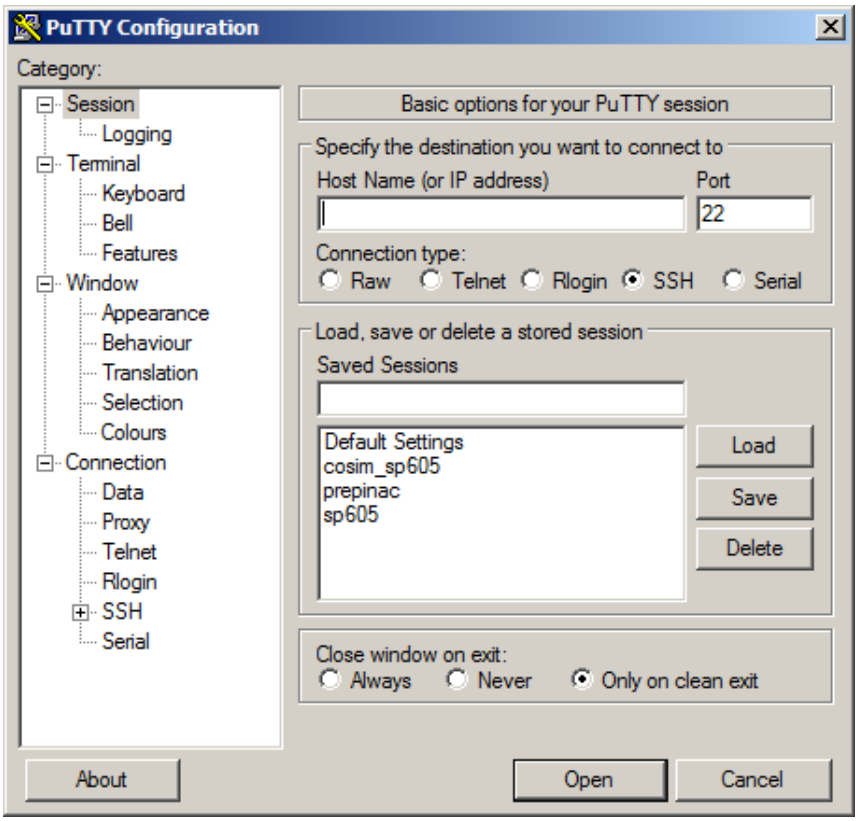

Figure 15: Open PuTTY terminal.

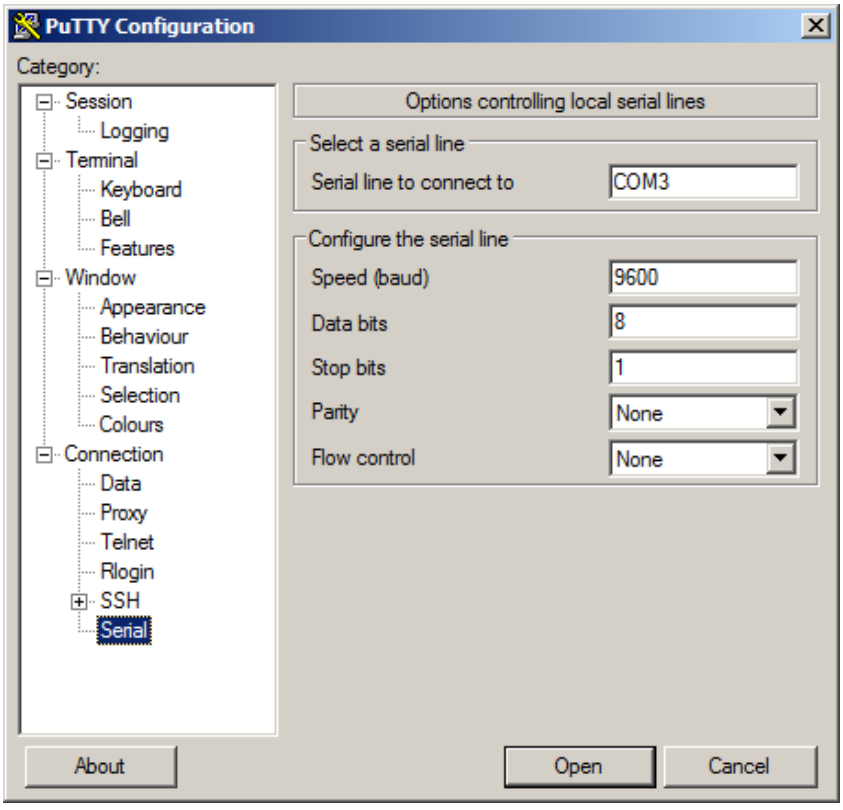

Figure 16: Select "Serial", select your COL port, set speed to 9600 and flow control to None.

3.3 Ethernet point to point connection with PC

The SDK SW projects included in this evaluation package demonstrate integration of the UTIA EdkDSP accelerator together with the Xilinx 1Gb Ethernet controller. The connection to the Ethernet is based on two versions of the LwIP SW:

- Raw versions of SDK SW projects use raw version of the LwIP library without real-time OS.
- Socket versions of SW projects use the socket version of LwIP on top of the Xilinx XilKernel.

Set your PC Ethernet connection to point-to-point with the fixed IP address:

192.168.8.2

All included UTIA EdkDSP projects are setting the IP address of the AC701 board to:

192.168.8.10

This setting enables the direct point to point Ethernet connection.

3.4 Boot of the bitstream

Program the AC701 board by selecting in SDK:
Xilinx Tools -> Program FPGA

C:\VM_07\d_34_7a\d_7a200t_fp12_1x8\SDK_Workspace\hw_platform\system.xml

Click on the “Program” button. See Figure 17.

The AC701 board is programmed with the system_wrapper.bit. The MicroBlaze is running in the initial bootloop from internal FPGA RAM.

3.5 Boot of the application

The SW bist_app.elf application from the “bist_app” project can be downloaded to the DDR3 memory and started. Select the “bist_app” project in the project navigator.

In SDK, select:
Run -> Run Configuration -> Xilinx C/C++ ELF

Click on the “New launch configuration” in the Run configuration screen and the bist_app.elf project executable is ready for download to DDR3 via the jtag cable. Click on “Run” button to download the executable. See Figure 18.

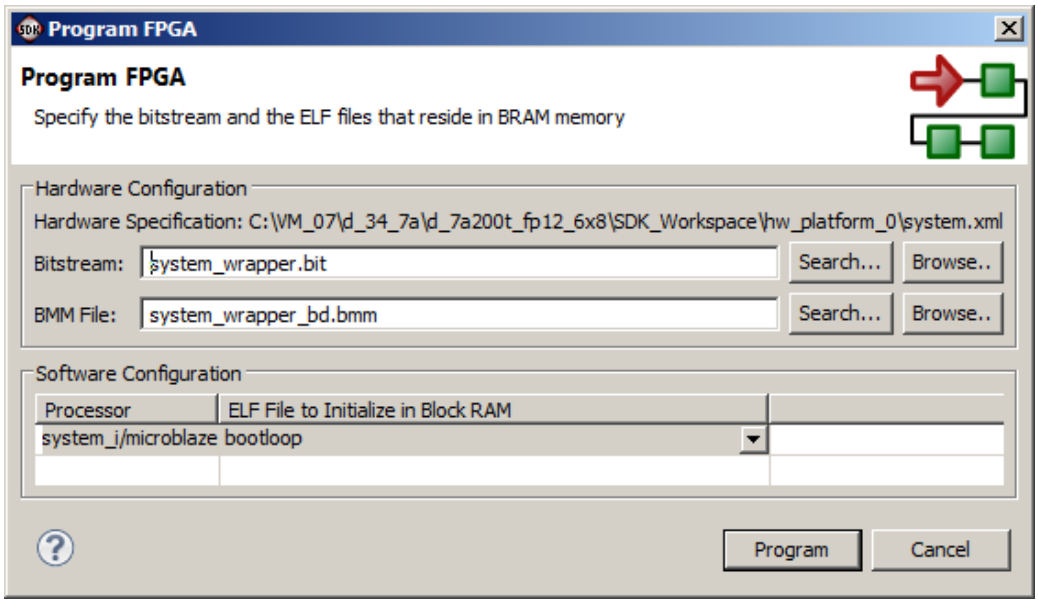

Figure 17: Program AC701 board.

Click on the “Program” button.

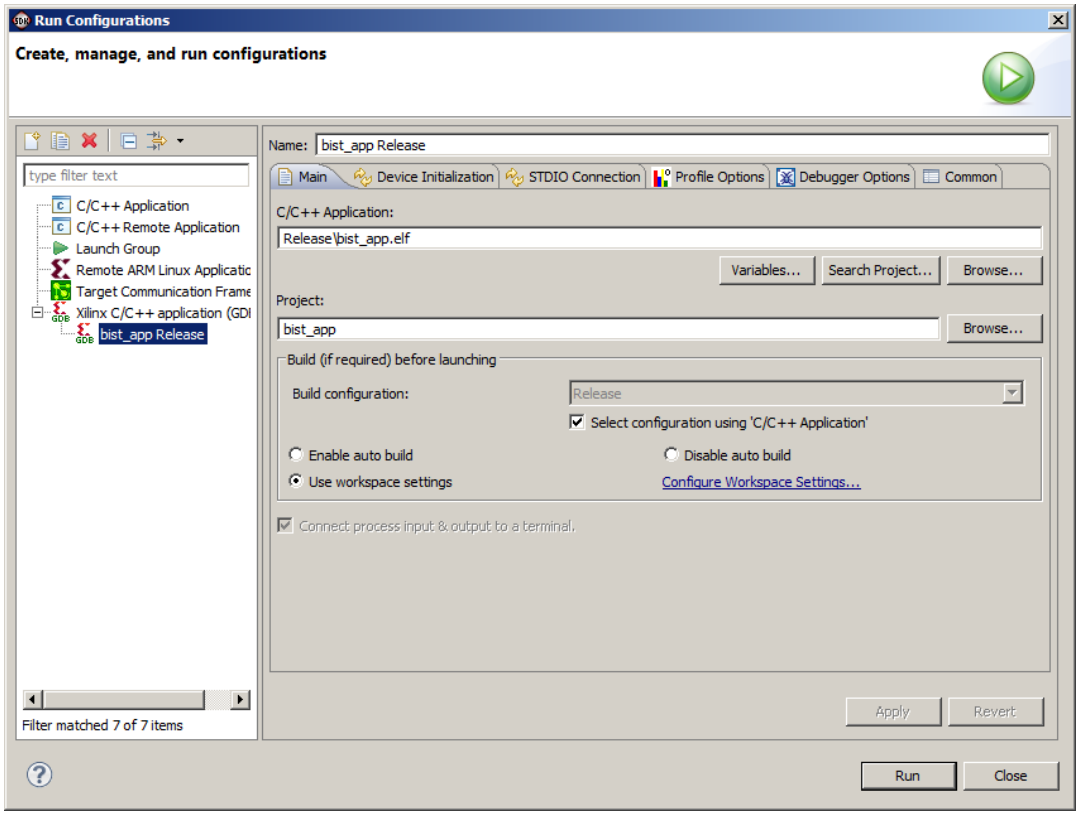

Figure 18: Select “bist_app.elf” code.

Run the application bist_app.elf by clicking on Run.

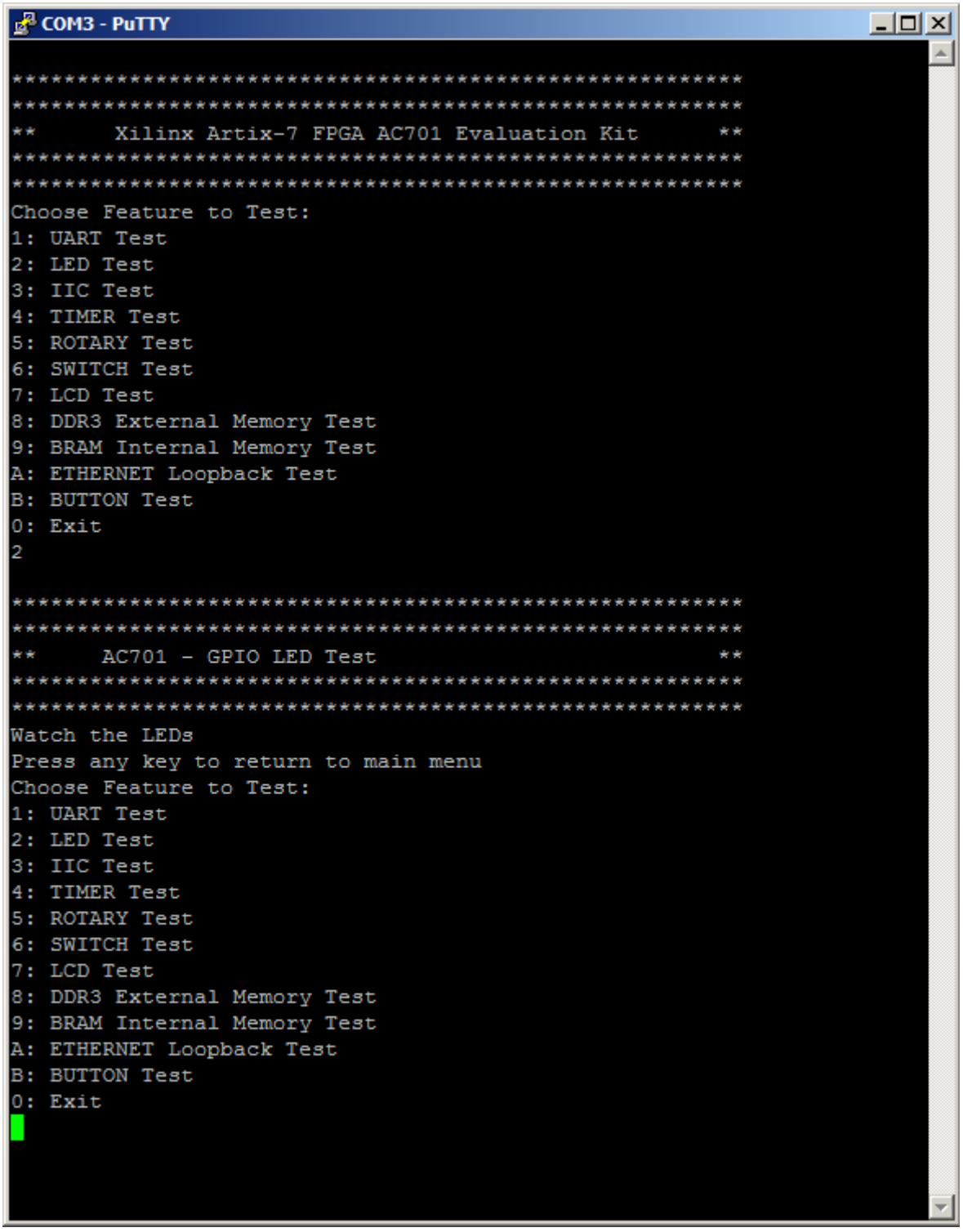

Figure 19: Run bist_app.elf and select tests from the terminal keyboard (PC).

The Xilinx **bist_app** demo serves for test of the MicroBlaze peripherals. Stop hardware from SDK.

Download again the bitstream (chapter 3.4), select the **edkdsp** project for download (chapter 3.5), run it to see the extended menu enabling tests of the EdkDSP accelerator. See Figure 20.

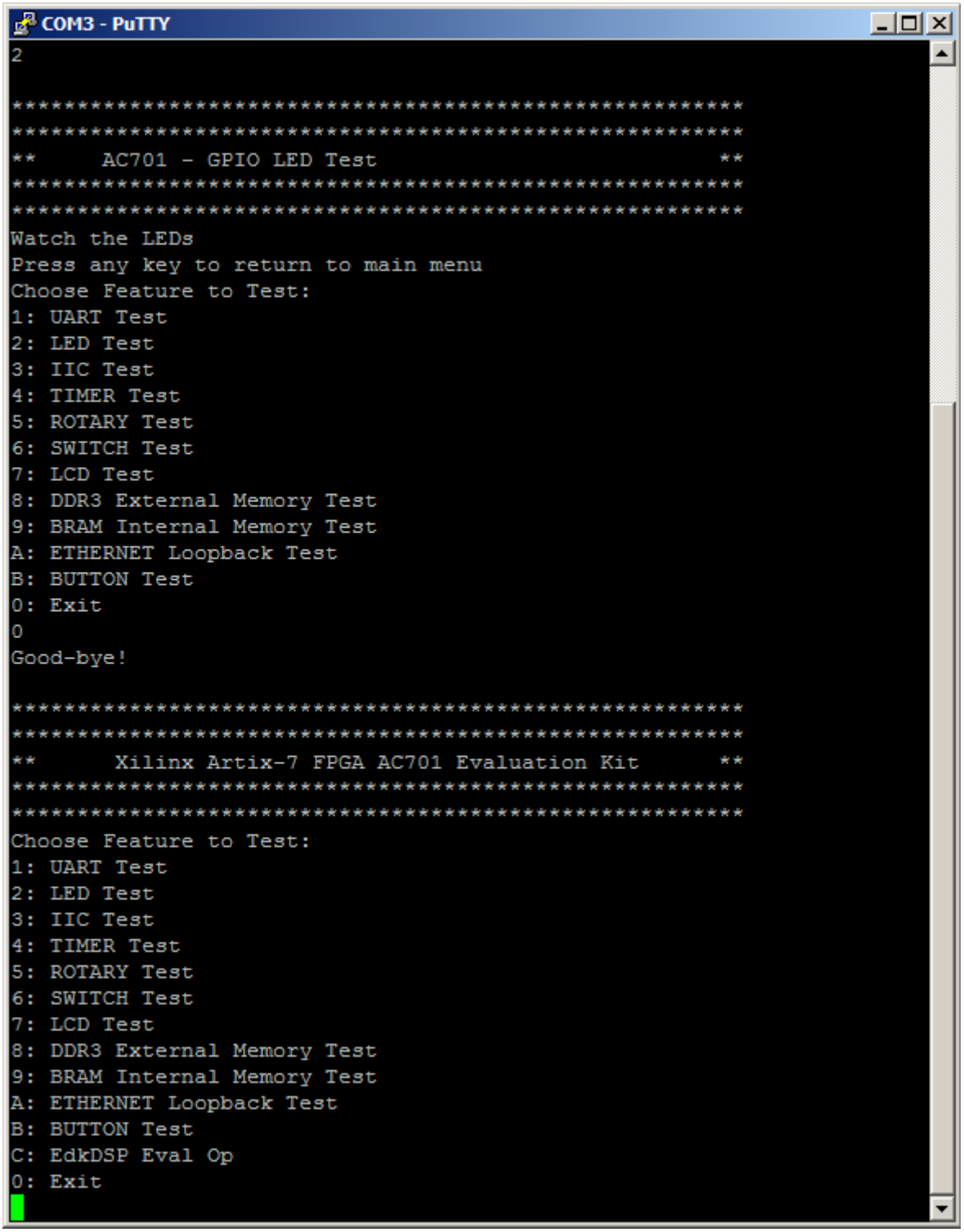

Figure 20: Run the edkdsp.elf application and select the EdkDSP Eval Op test.

Select the C option from the terminal keyboard to run test of the EdkDSP accelerator. See Figure 21.

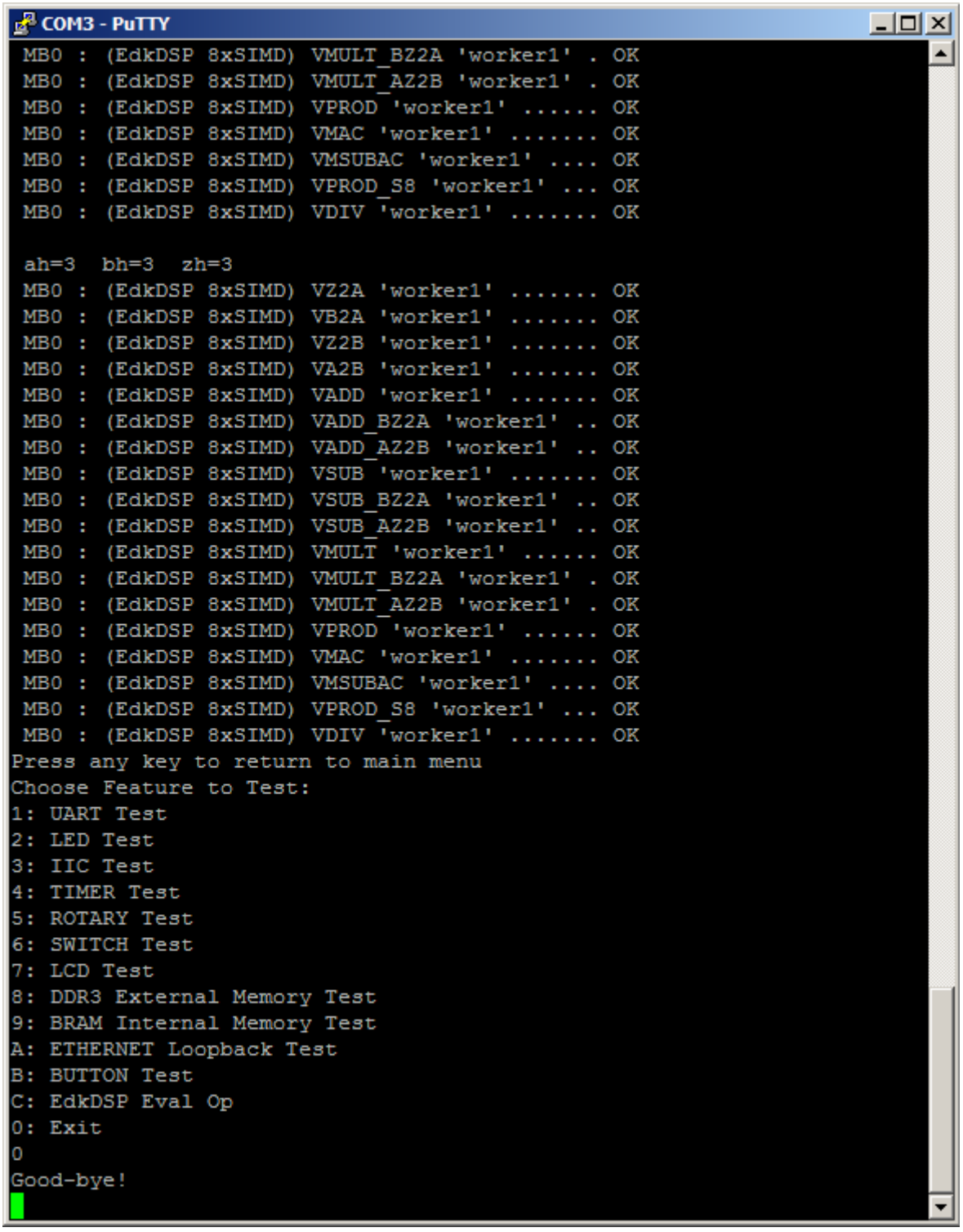

Figure 21: The EdkDSP basic vector floating point operations have been tested.

Stop hardware from the SDK. Download again the bitstream (chapter 3.4), select the **raw_axi_bce_fp12_1x8_eval_op** project for download (chapter 3.5) and run it. See Figure 22.

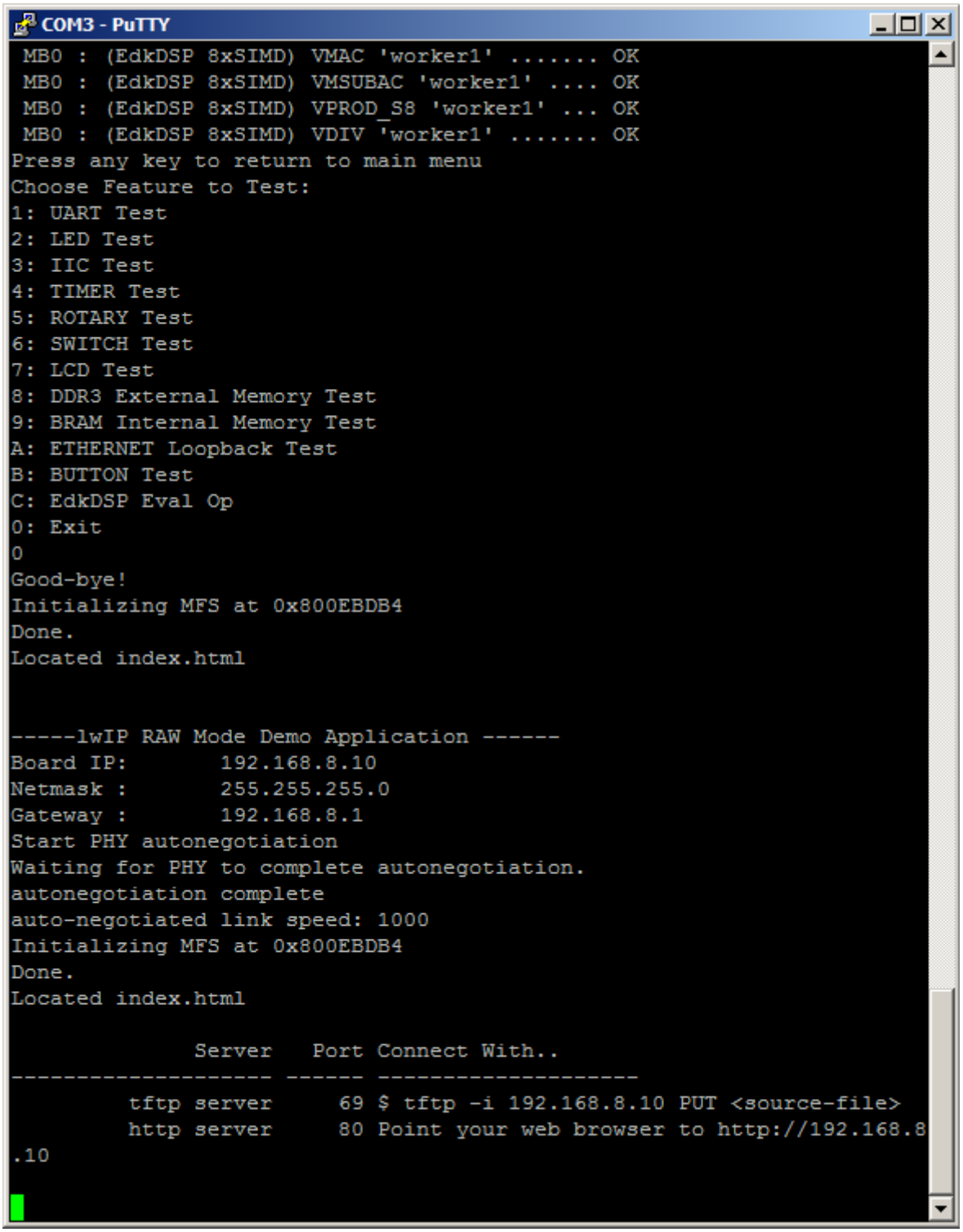

Figure 22: Select "raw_axi_bce_fp12_eval_opl.elf application to test the lwIP services in RWW mode.

The RAW version of the tftp server and the RAW version of the http server have been started on the Artix7 MicroBlaze processor. Open www browser in (Internet Explorer) client and connect to the board address http://192.168.8.10/

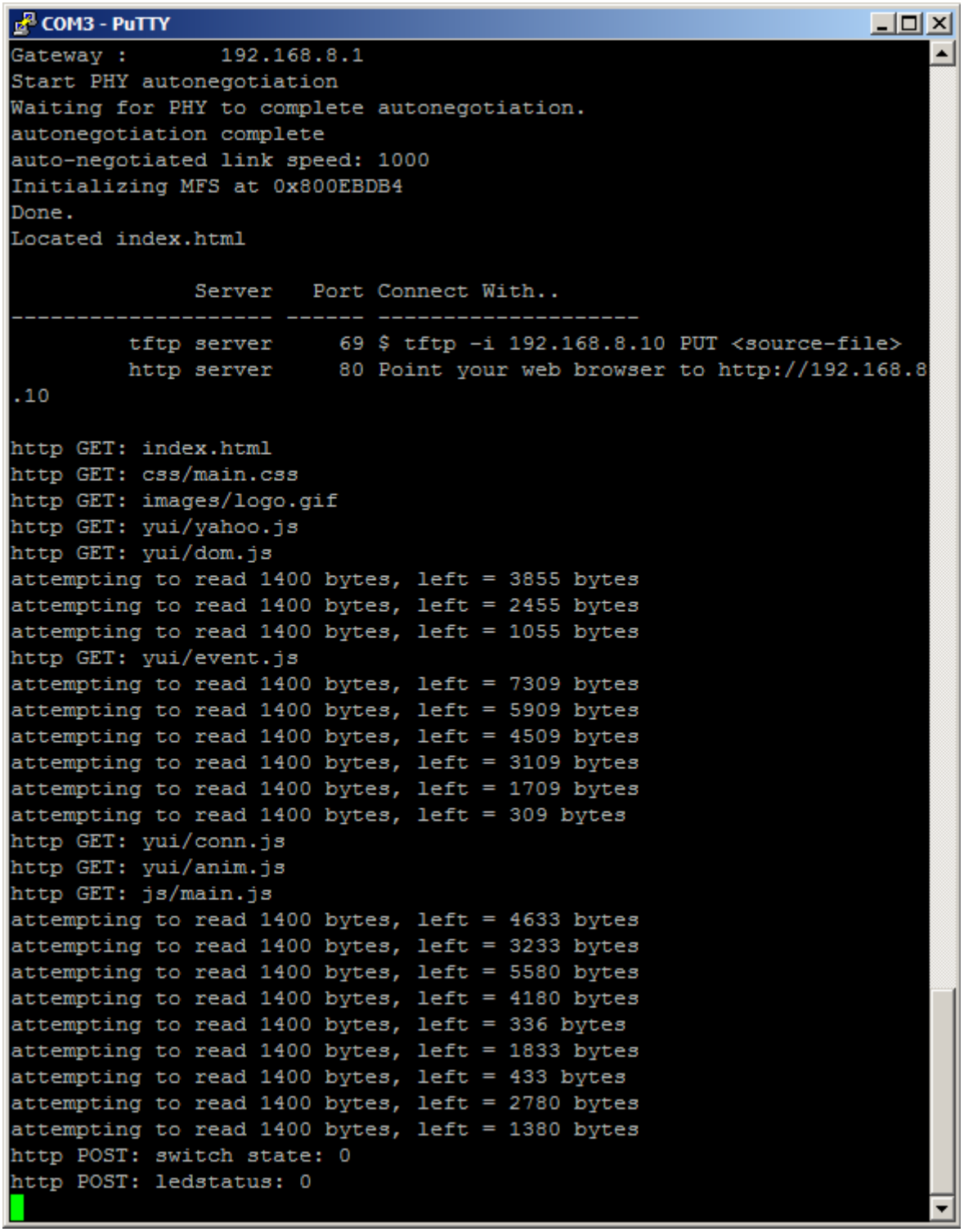

Figure 23: The Java Script has been loaded from the FPGA RAM based file system to your brawser.

Support script files are downloaded to the PC from the Artix7 file system and the interface page is started. See Figure 23 and Figure 24.

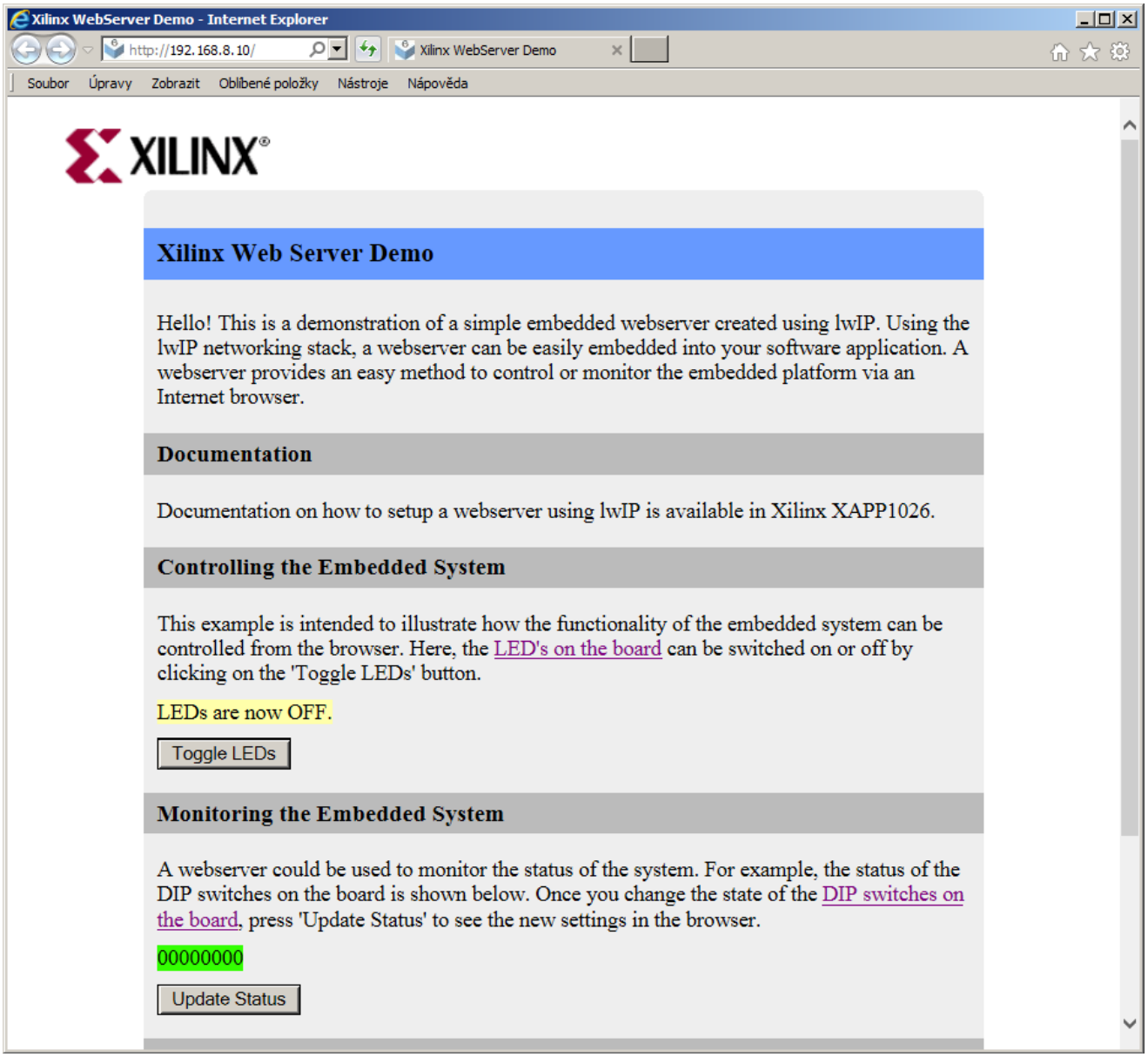

Figure 24: The demo www server is evaluating the basic GUI for communication from the web browser client to the Artix7 application working as an embedded server providing 1 G bit point to point connection.

The **Update Status** button serves to get the DIP switches status. The **Toggle LEDs** button is toggling the led output on the board and starts the EdkDSP accelerator evaluation. See Figure 25. The SW application is testing presence of an updated firmware in the RAM based file system of the board. If it is not present, the default firmware is used.

The file FP1101.TXT is open for WR in the RAM based file system. It will store text messages from the tested EdkDSP accelerator.

The capabilities of all 6 EdkDSP accelerators are displayed next. This information is based on the reply from the initialised accelerators. Test is performed. Finally the top directory of the RAM based file system is listed together with the information about used and free blocks in the RAM based file system. See Figure 25.

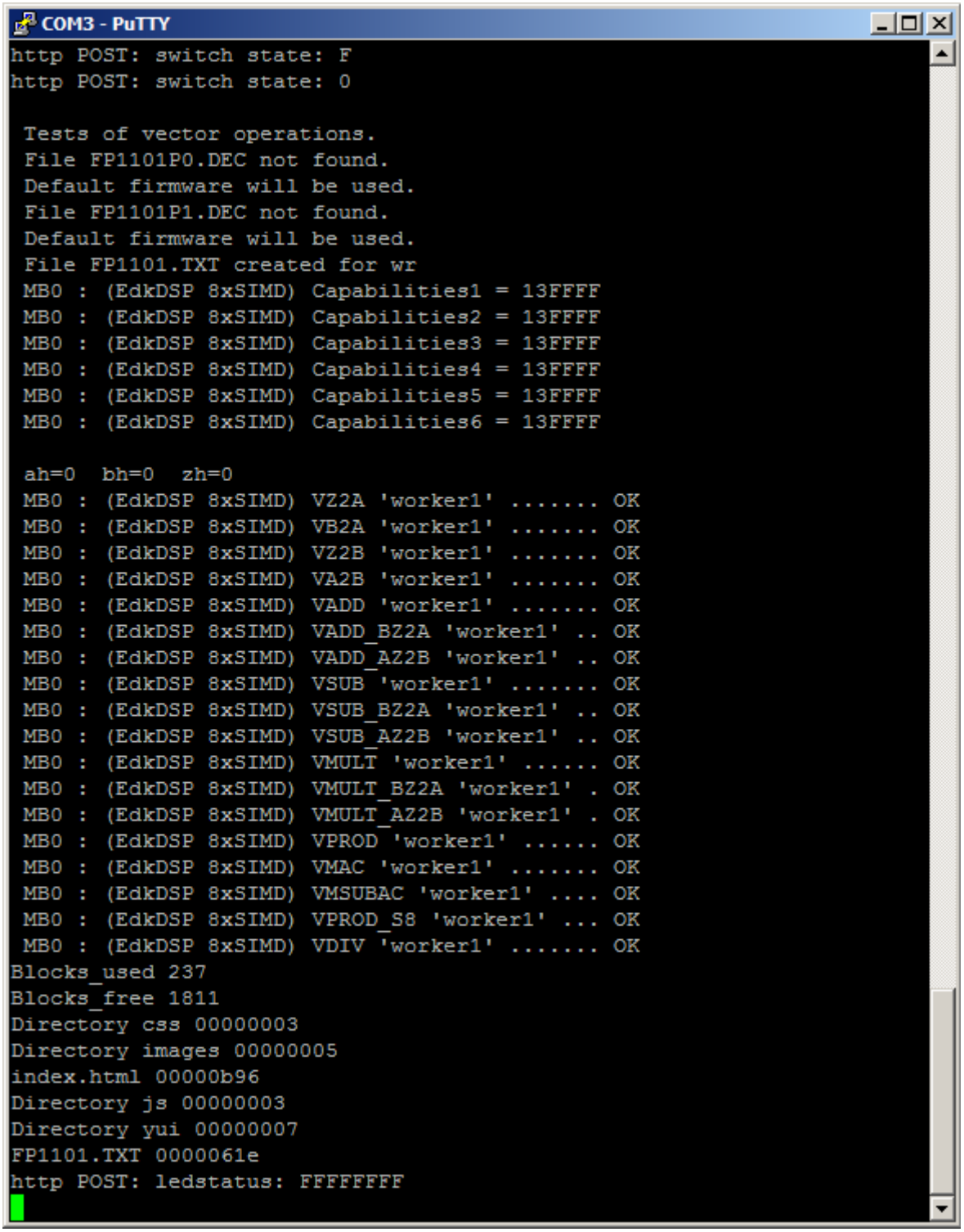

Figure 25: Test of basic operations has been started from the web browser GUI TOGLE LED button. The listing of the top level directory of the RAM based file system is provided to the terminal.

Close the web browser. Close the application running on the Artix7 from the SDK (click on the Red square icon on top of the console and next on the X icon to close the debugger session).

Download again the bitstream (chapter 3.4), select the **socket_axi_bce_fp12_1x8_eval_op** project for download (chapter 3.5) and run it. See Figure 26.

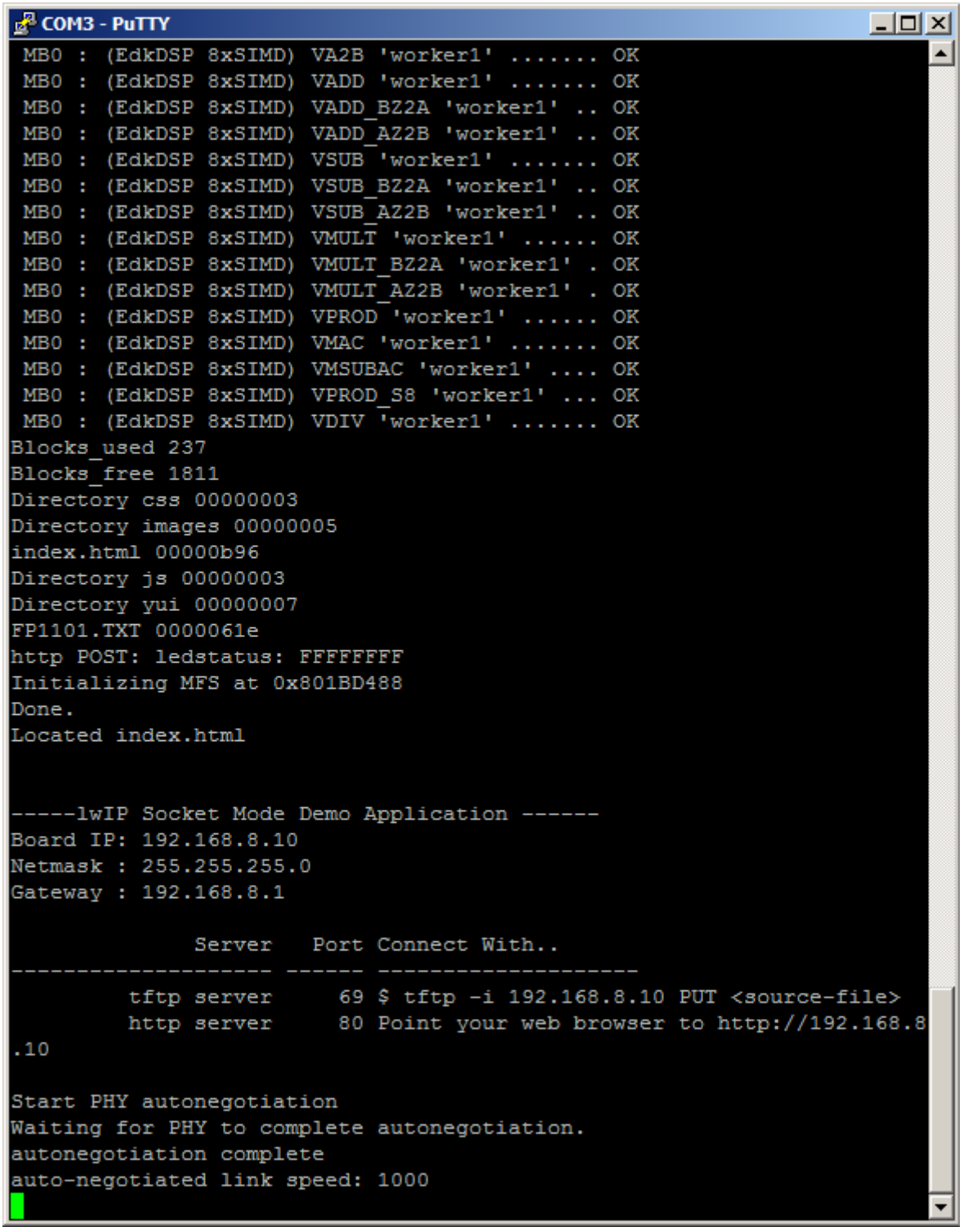

Figure 26: Start the socket_axi_bce_fp12_1x8_eval_op.elf demo application, working on top of the Xilkernel OS..

The SOCKET version of the tftp server and the http server have been started on the Artix7 MicroBlaze processor. Open www browser (Internet Explorer) client and connect to the board address: http://192.168.8.10/

Click on the **Toggle LEDs** button to toggle the led output on the board and to starts the EdkDSP accelerator evaluation. The SOCKET version of the server supports both buttons in parallel. See Figure 27.

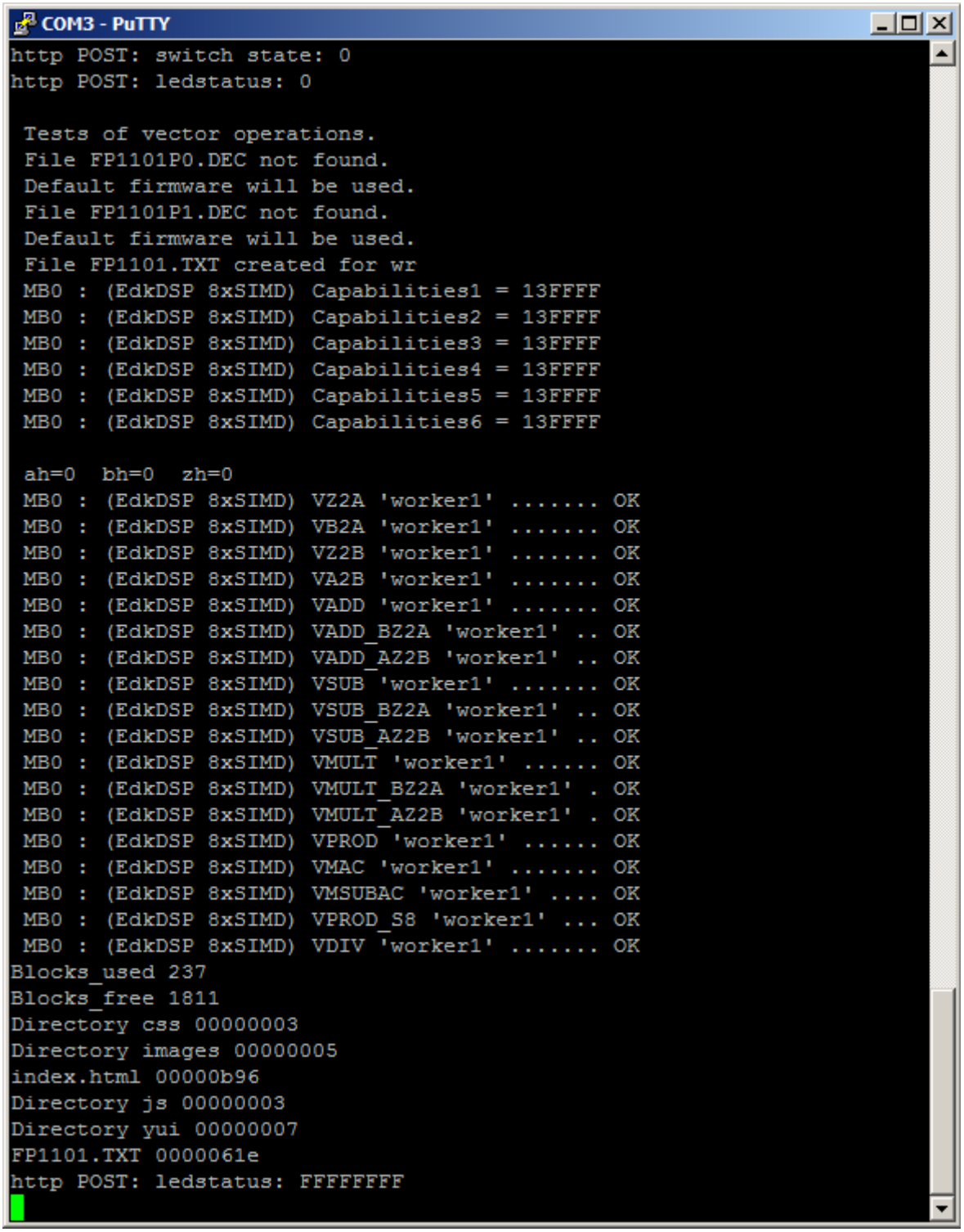

Figure 27: Test of vector operations is started from the www browser GUI. It is served by the lwIP library working on top of the Xilkernel.

Close the web browser. Close the socket based application running on the Artix7 from the SDK.
Download again the bitstream (chapter 3.4), select the socket_axi_bce_fp12_1x8_fir_lms project for download (chapter 3.5) and run it.

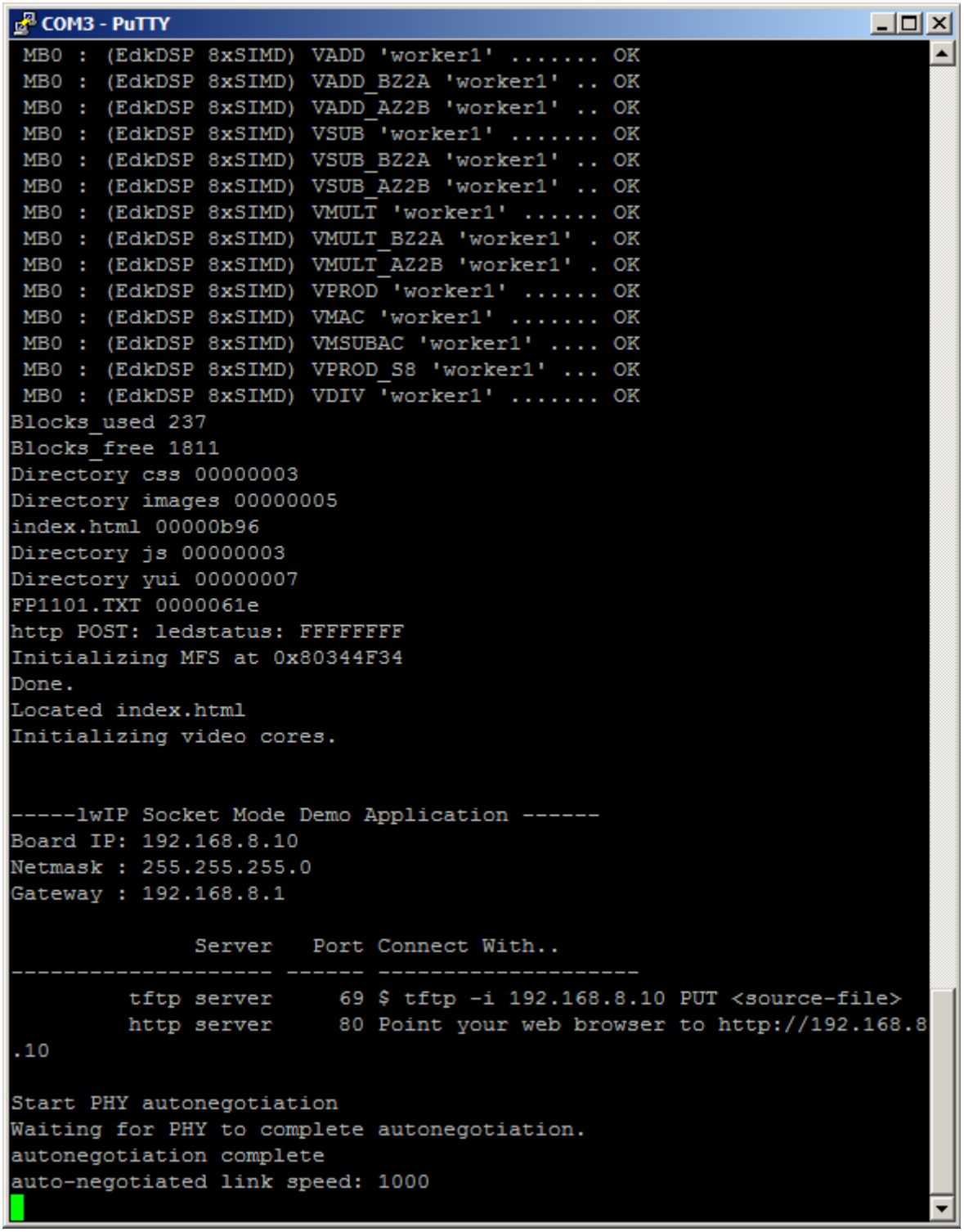

Figure 28: Start the socket_axi_bce_fp12_1x8_fir_lms.elf application.

The SOCKET version of the TFTP and HTTP servers have been started on the Artix7 MicroBlaze processor. Open www browser (Internet Explorer) client and connect to the board address: http://192.168.8.10/

Click on the **Toggle LEDs** button to toggle the led output on the board and starts the FIR and LMS filter computation on single (8xSIMS) EdkDSP accelerator. See Figure 29.

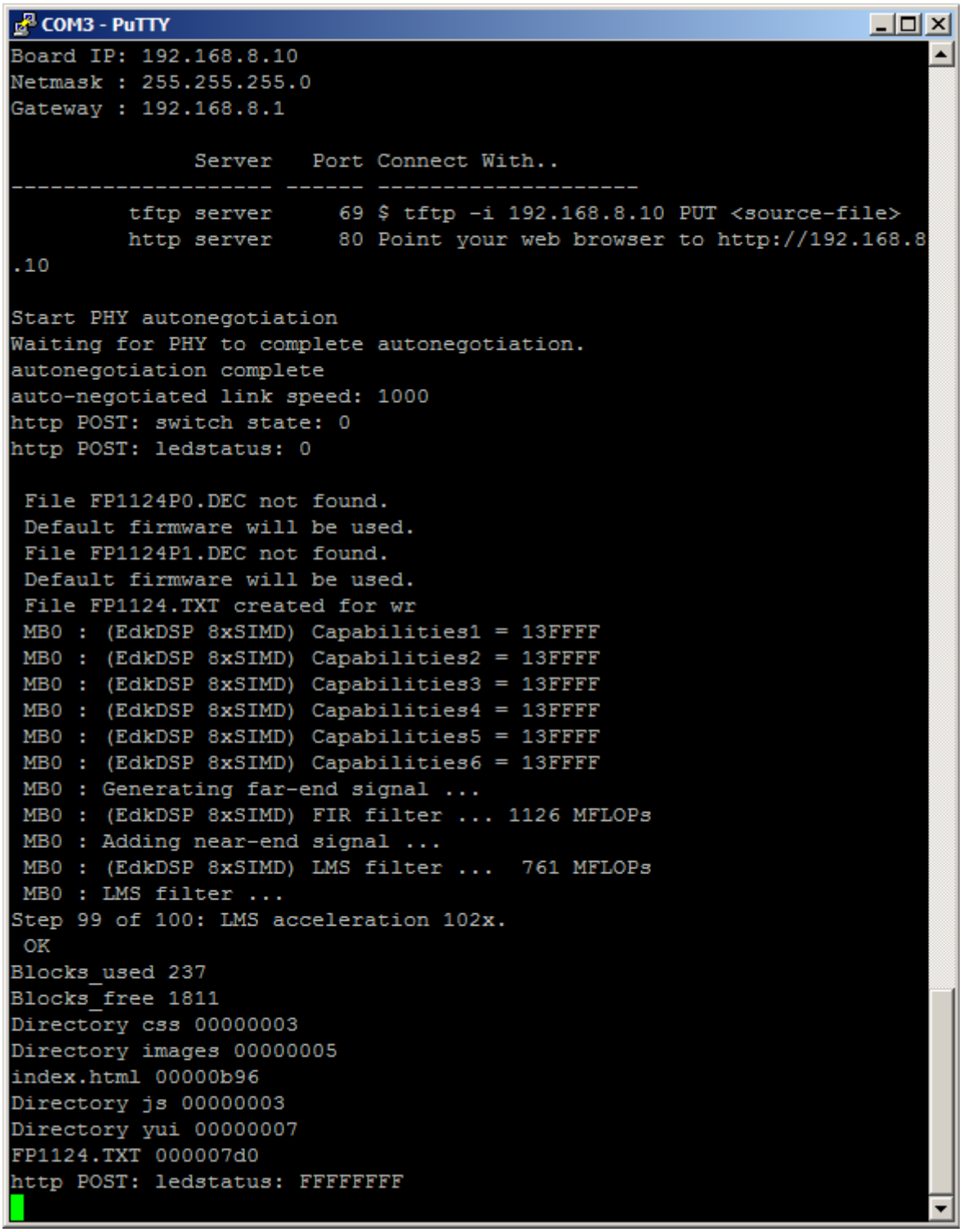

Figure 29: The FIR and LMS computation is started from the web browser GUI. The performance of single EdkDSP accelerator is measured and compared to the performance of MicroBlaze processor with HW floating point unit.

The performance for FIR and LMS is displayed and the speedup in comparison to the MicroBlaze is reported during the MicroBlaze verification run. The result from the EdkDSP is identical to the MicroBlaze result.
Close browser. Stop the Artix7 application.

3.6 Use of the C compiler for the EdkDSP firmware with download from Ethernet

This section is describing the use of the UTIA EdkDSP C compiler to recompile the firmware for the PicoBlaze6 controller present in each of the six (8xSIMD) EdkDSP accelerators in the AC701 board.

In SDK Project Explorer, open the project edkdsp_cc and the subdirectory edkdsp_cc/a. See Figure 30. It contains C source code of the EdkDSP accelerator firmware and Ubuntu scripts for the compilation. The compiled versions of firmware are already present in the demonstrated applications in form of headers for the MicroBlaze applications. This helps to evaluate the EdkDSP accelerators without installation of the C compiler for the EdkDSP.

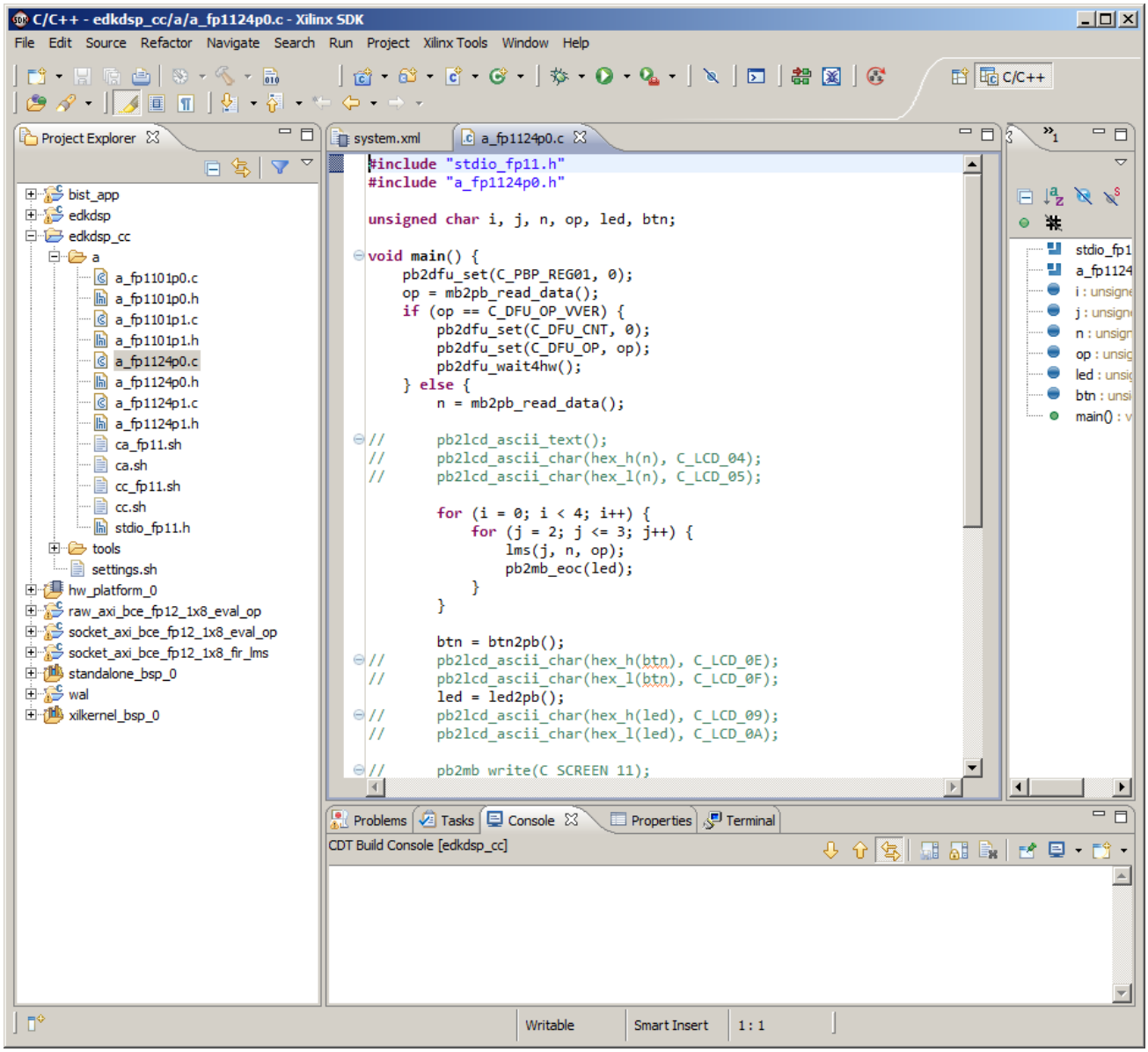

Figure 30: Evaluate the included C code for reprograming of the EdkDSP accelerators.

The UTIA EdkDSP C compiler is provided as implemented as several Ubuntu binary applications.
The "VMware player" software and the compatible Ubuntu image version is needed to run the UTIA EdkDSP C compiler on Windows 7 (64bit or 32bit) PC.

The Ubuntu image used in UTIA needs two DVD disks (8GB) for installation. That is why it is not included as part of the evaluation package. If you would need this image, write an email request to kadlec@utia.cas.cz to get these two DVD with correct Ubuntu image from UTIA (free of charge).

Install the VMware Player software (64bit or 32bit) on your PC.
In VMware Player open the Ubuntu_EdkDSP package. See Figure 31.

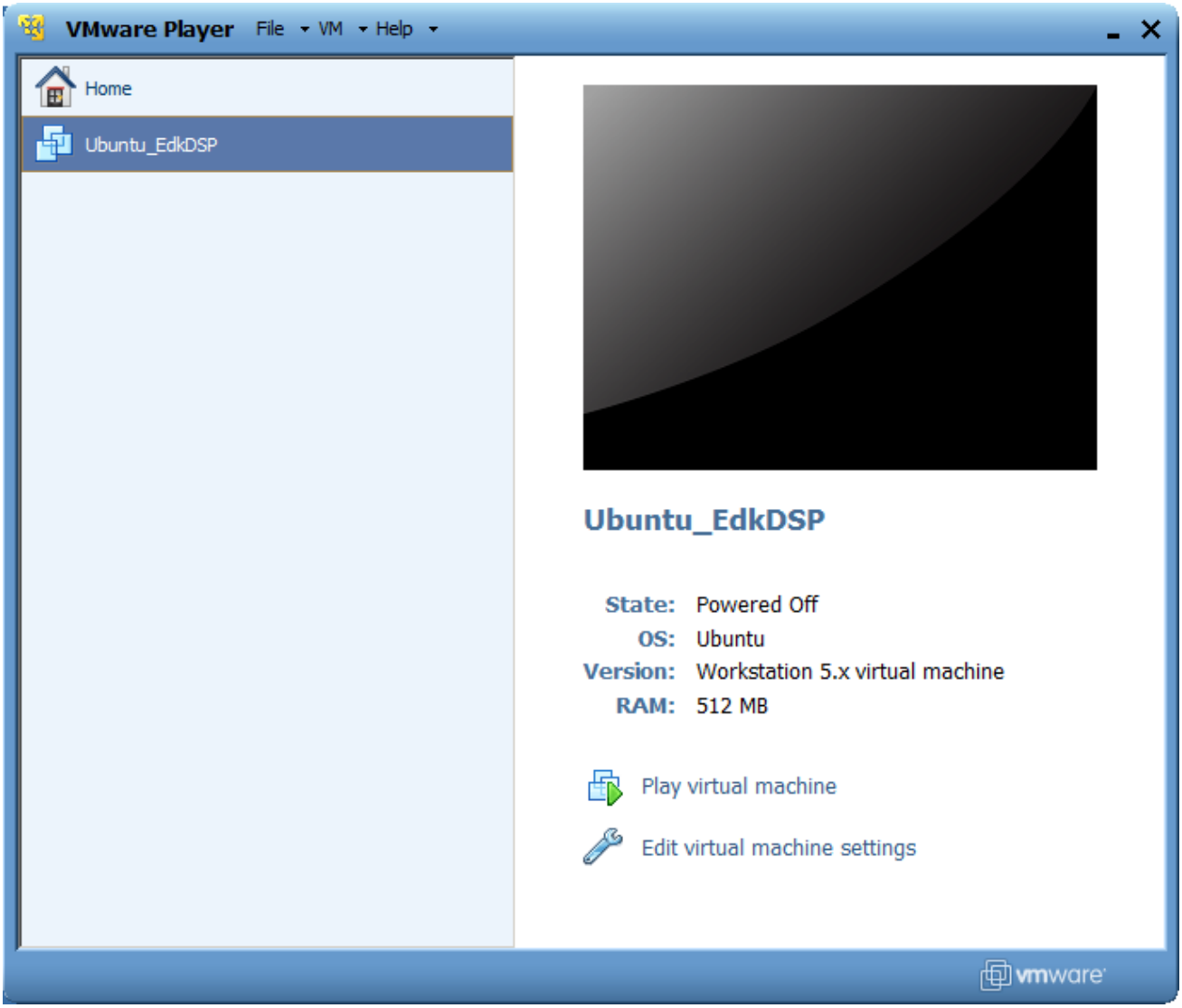

Figure 31: Start the VMware Player to run the C compiler for the EdkDSP accelerators as an Ubuntu binary user application.

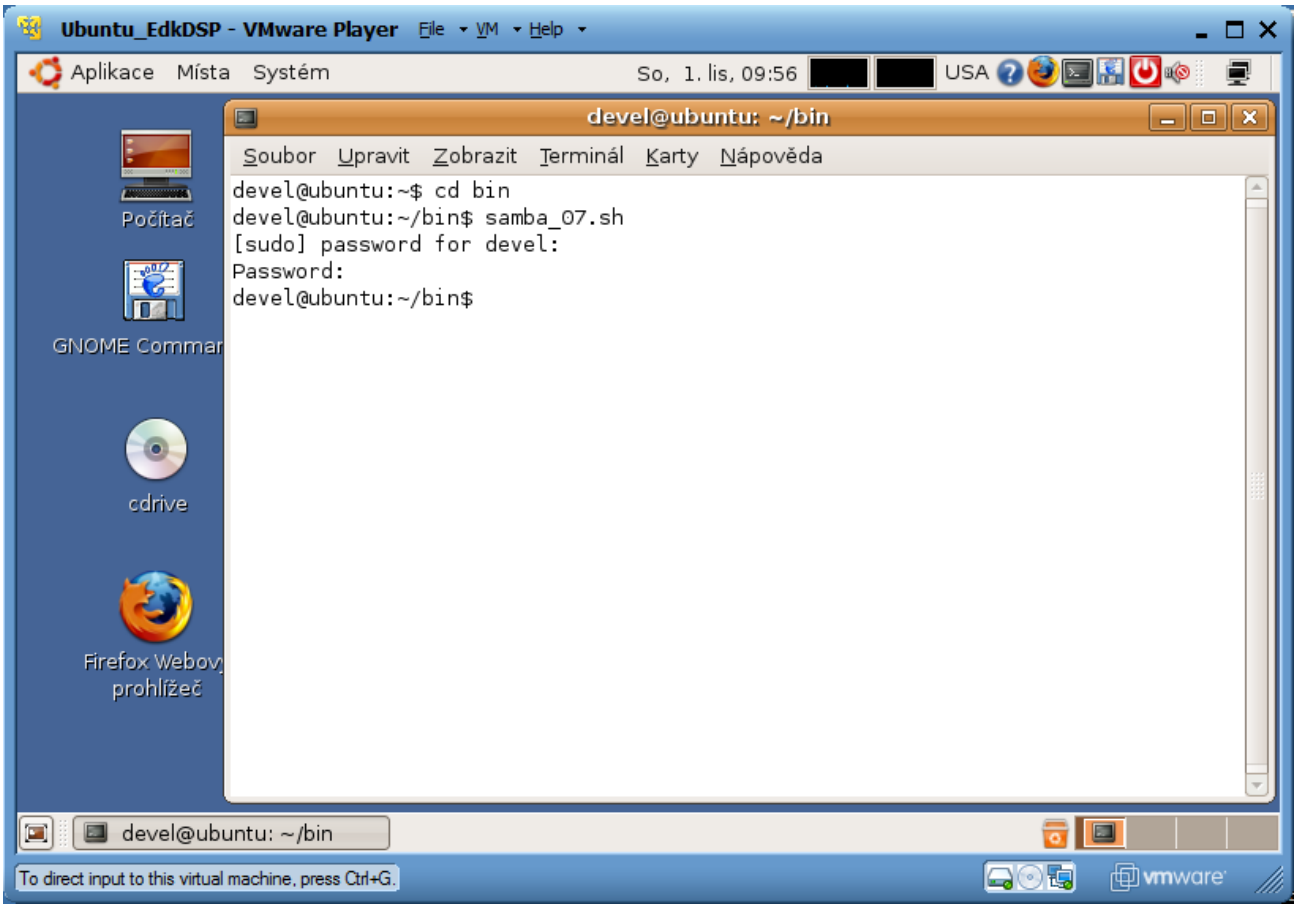

Figure 32: Mount the Windows 7 directoy c:\VM_07as /mnt/cdrive in Ubuntu

Open the VMware Player and select the “Ubuntu_EdkDSP” image. The Ubuntu will start.
Login as:
User: devel
Pswd: devuser

The PC directory c:\VM_07 needs to be shared by Windows 7 with Ubuntu.
In Windows 7, set the directory c:\VM_07 and its subdirectories as shared with the __vmware_user__ for Read and Write. In Ubuntu, open terminal and mount the PC directory c:\VM_07 to Ubuntu.
The Windows 7 c:/VM_07 directory is mounted to the Ubuntu OS as: /mnt/cdrive
This process has been automated by the script samba_07.sn in my case. See Figure 32.

In Ubuntu terminal, change the directory to:
$ cd /mnt/cdrive/d_34_7a/d_7a200t_fp12_6x8/SDK_Workspace/edkdsp_cc

The EdkDSP C compiler utilities have to be on the Ubuntu PATH. This is done by sourcing the settings.sh script in this directory. Type in Ubuntu terminal (See Figure 33):
$ source settings.sh
In Ubuntu terminal, change the directory to the example directory (See Figure 33):
$ cd a
devel@ubuntu:/mnt/cdrive/d_34_7z/d_7z020_fp12_4x8/SDK_Workspace/edkdsp_cc/a$

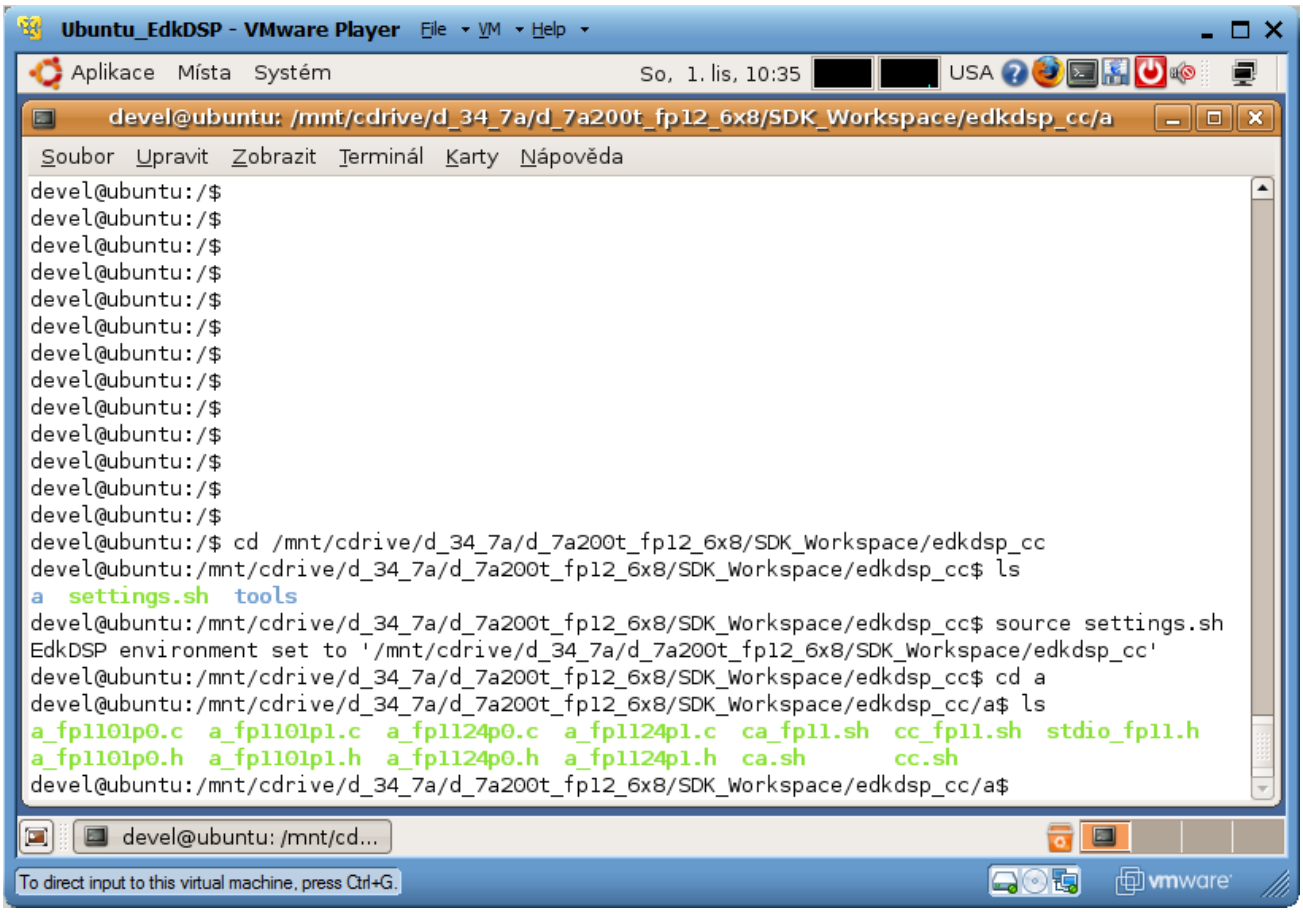

Figure 33: Source the path to the EdkDSP C compiler tools.

In SDK, open the C source code of the current firmware for the EdkDSP accelerator in the file edkdsp_cc/a/a_fp1101p0.c

See the original listing in Figure 34.
To demonstrate the compilation and new firmware download via Ethernet, We will change the message going from EdkDSP PicoBlaze processor to the MicroBlaze and to the FP1101.TXT log file from I=00; to Input=00.

Uncomment the four commented lines from // pb2mb_Write ('n'); to // pb2mb_Write ('t');
See Figure 34.
Save the modifications.

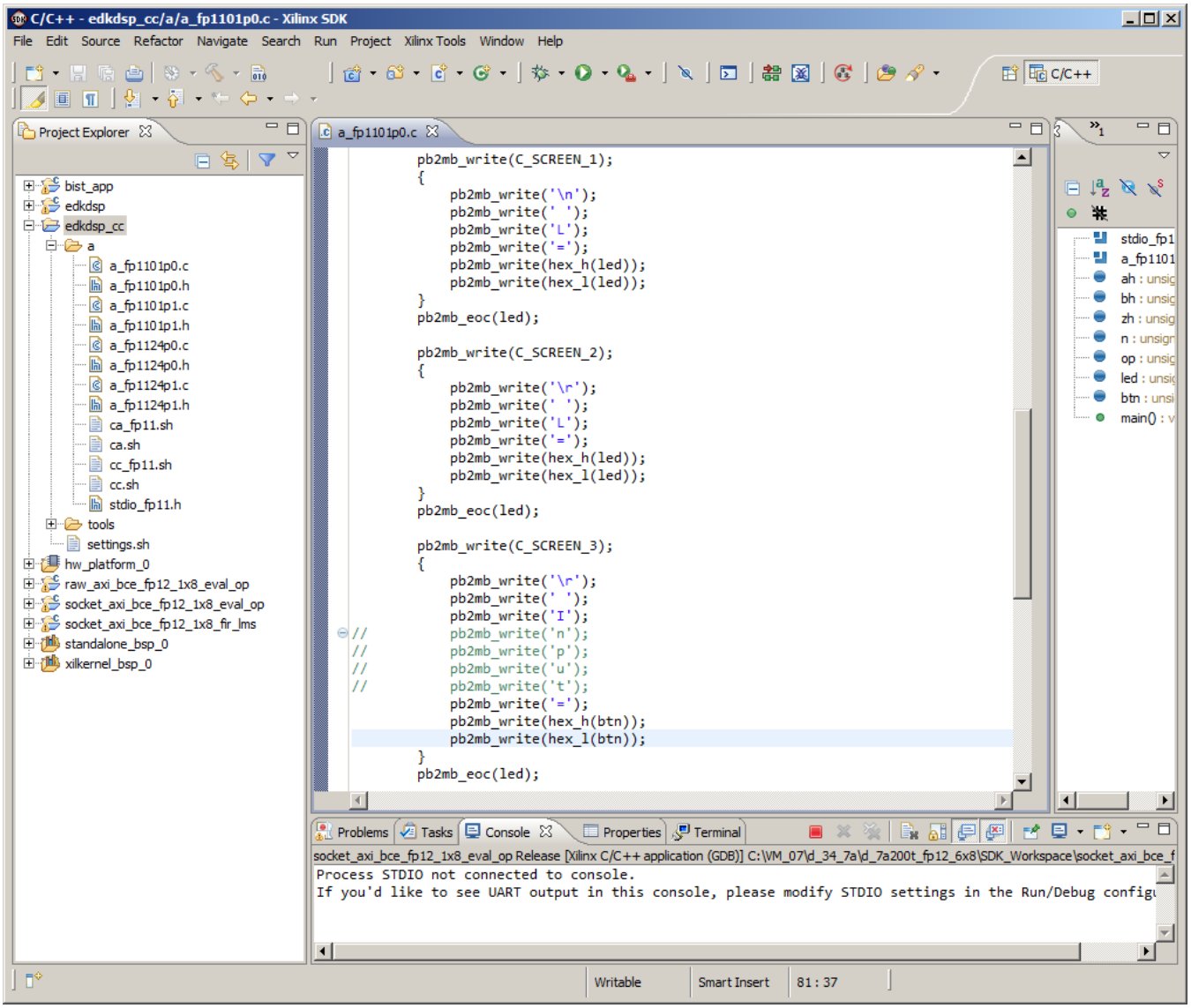

Figure 34: See the details of communication from the accelerator to MicroBlaze in the original code.

We will demonstrate the complete process related to the compilation, download of results from Artix7 to the PC and upload of the bitstreem to the Artix7 now.

Start the application socket_axi_bce_fp12_1x8_eval_op.elf and open the www browser and start the demo run by clicking on the Toggle LEDs button.
See Figure 35 and Figure 36

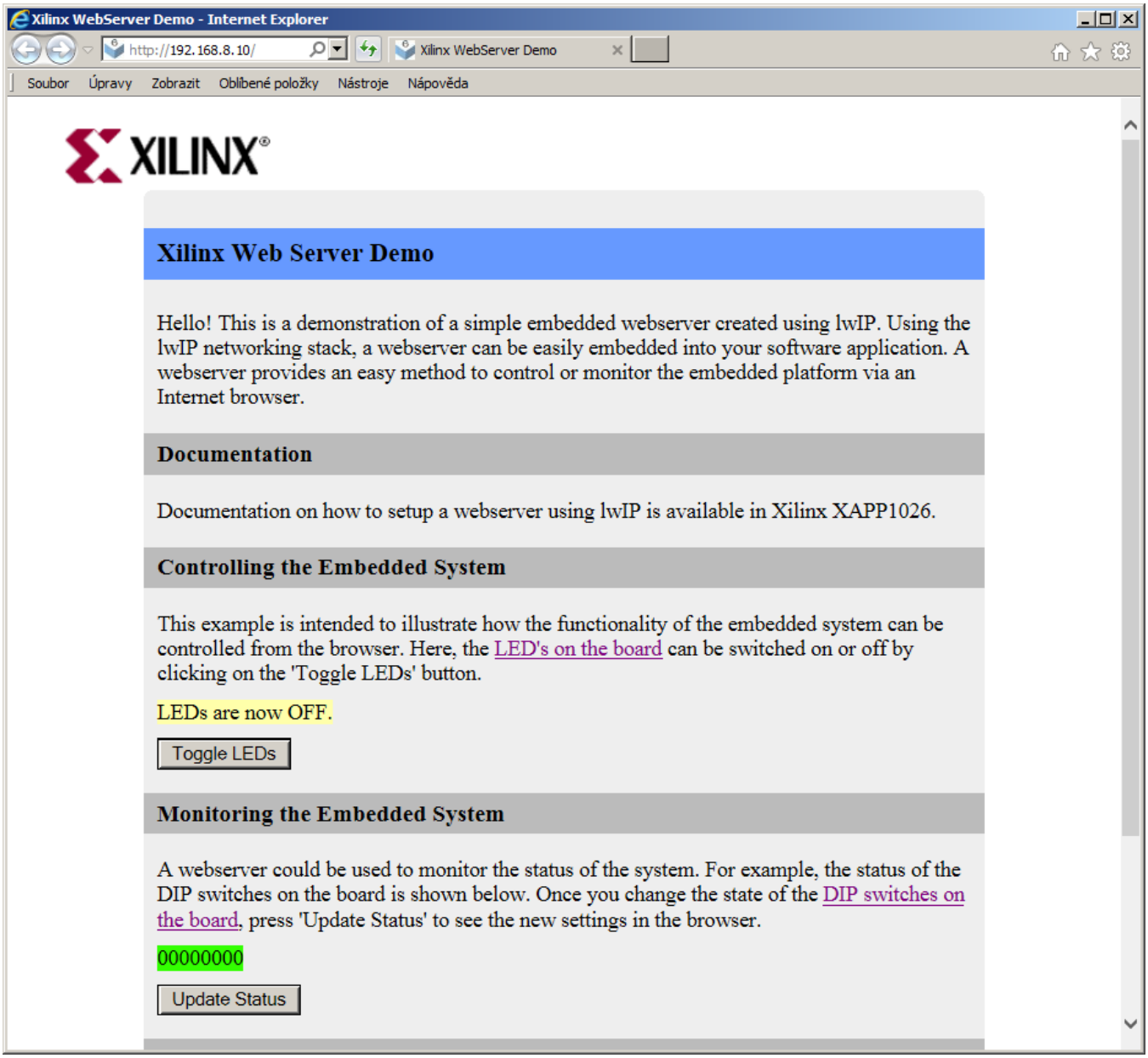

Figure 35: Start test from the web brawser GUI by Toggle LEDs button.

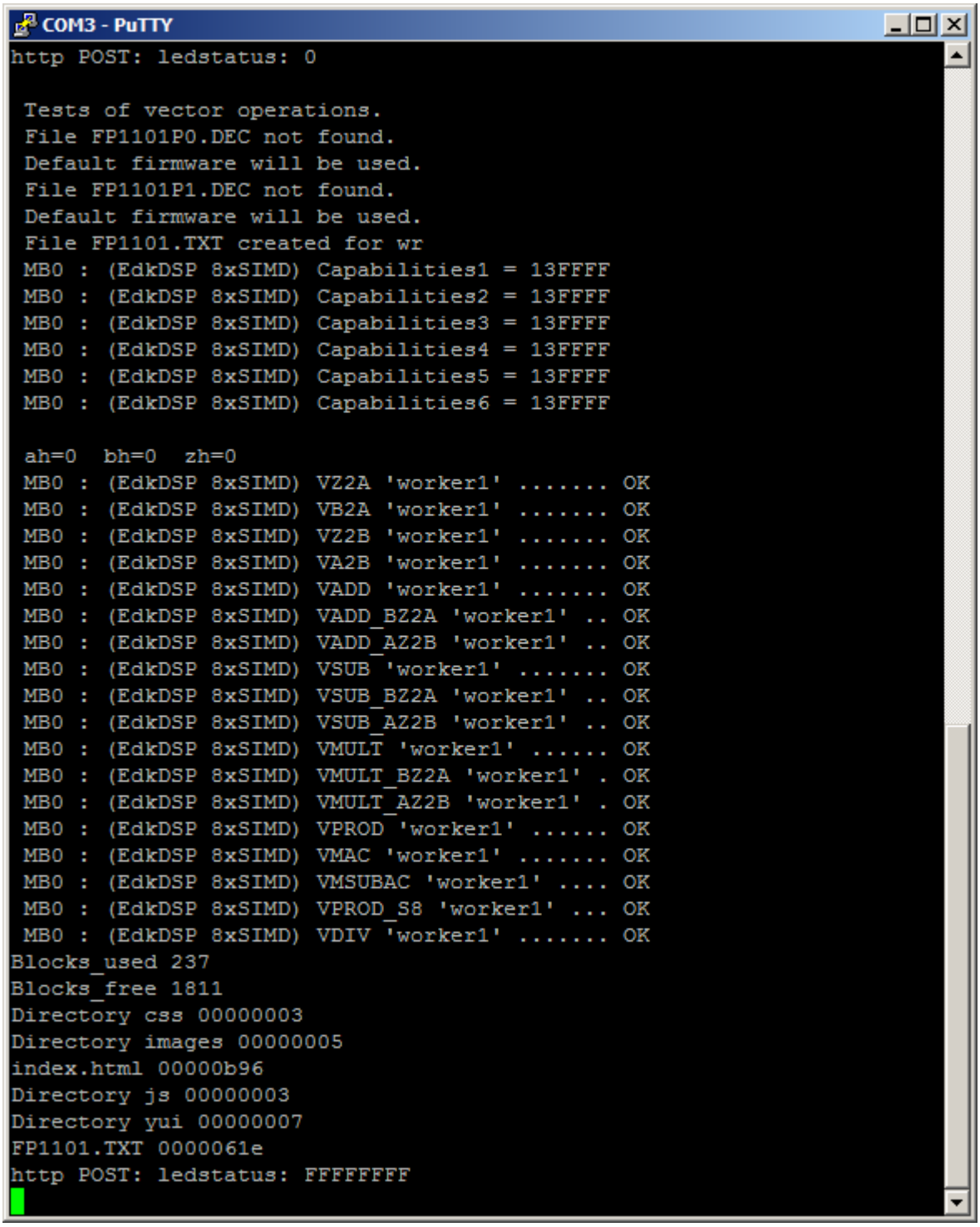

Figure 36: Test has been performed and the tested EdkDSP accelerator created data file FP1101.TXT in the RAM file system located in the DDR3 of the AC701 board.

Open the TFTP application on your PC as a TFTP client connected to the Artix7 host 192.168.8.10 with Port 69. See Figure 37. Select Local (PC) file to:
c:\VM_07\ d_34_7z\d_7z020_fp12_4x8\SDK_Workspace\edkdsp_cc\a\FP1101.TXT
and Remote File (Artix7 file system) to:
FP1101.TXT
See Figure 37 and Figure 38 for the selection of the PC file location. Click on Get to download the file.

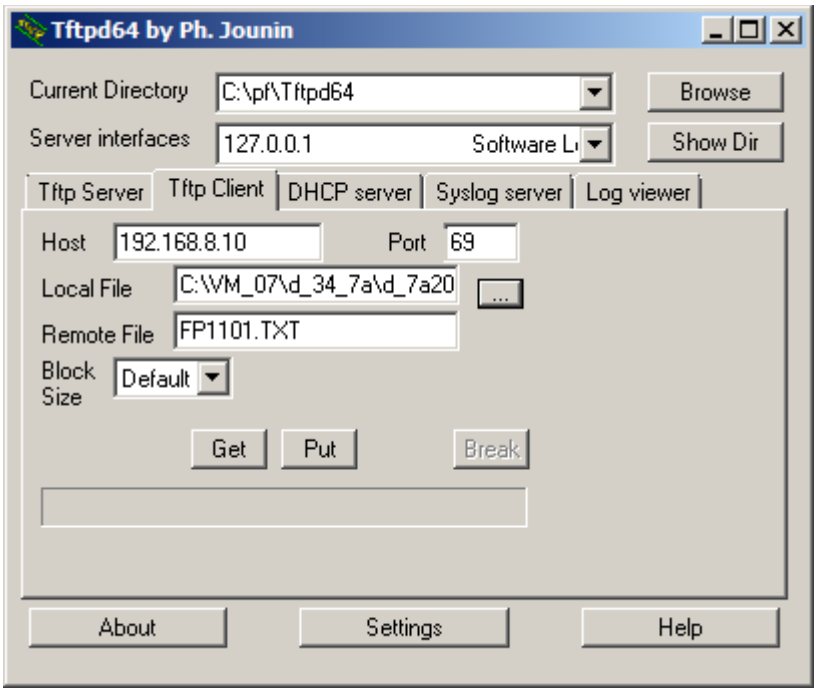

Figure 37: Start TFTP client and get the file FP1101.TXR from the Artix7 FPGA to PC via Ethernet.

The EdkDSP firmware after the compilation is presented in Figure 36.

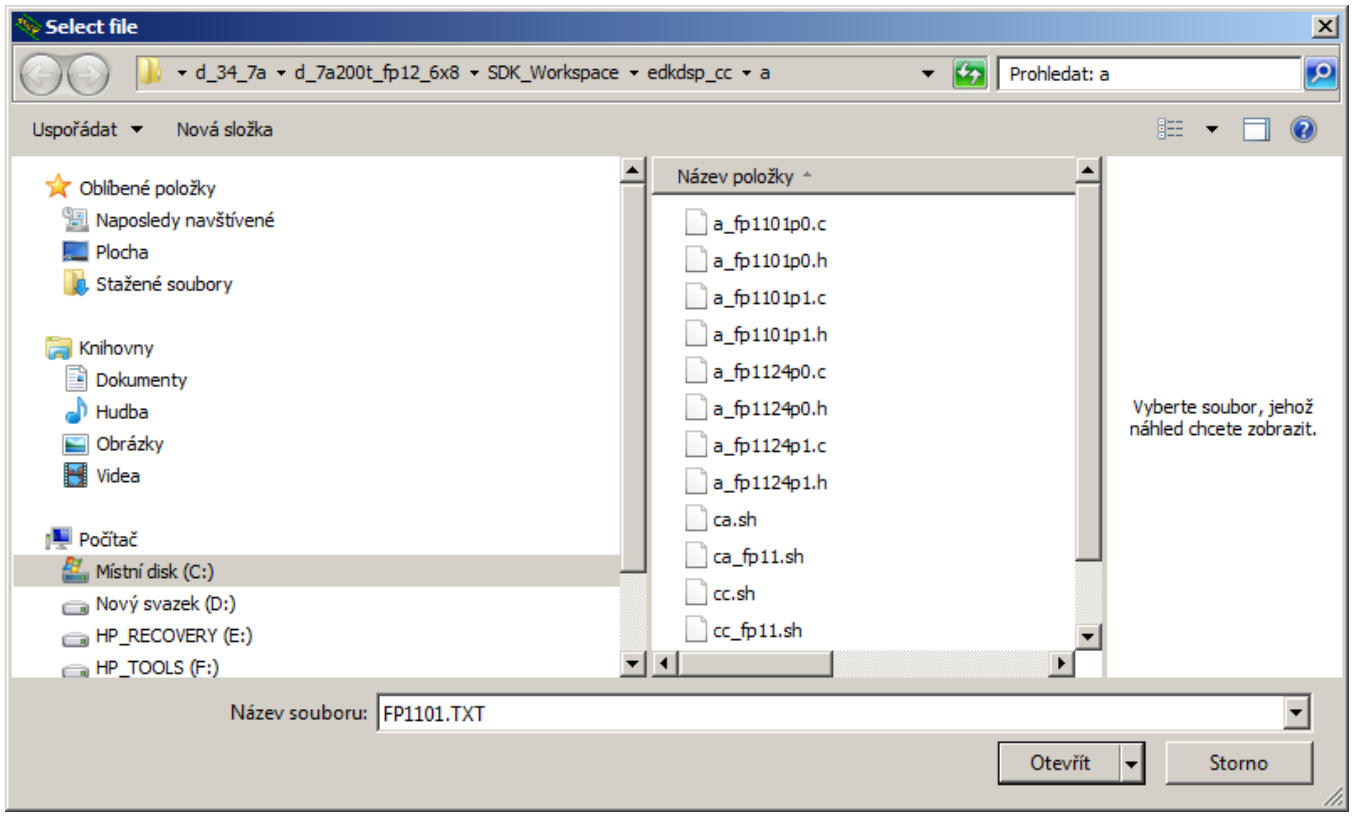

Figure 38: Select the directory where you want to get the FP1101.TXT file.

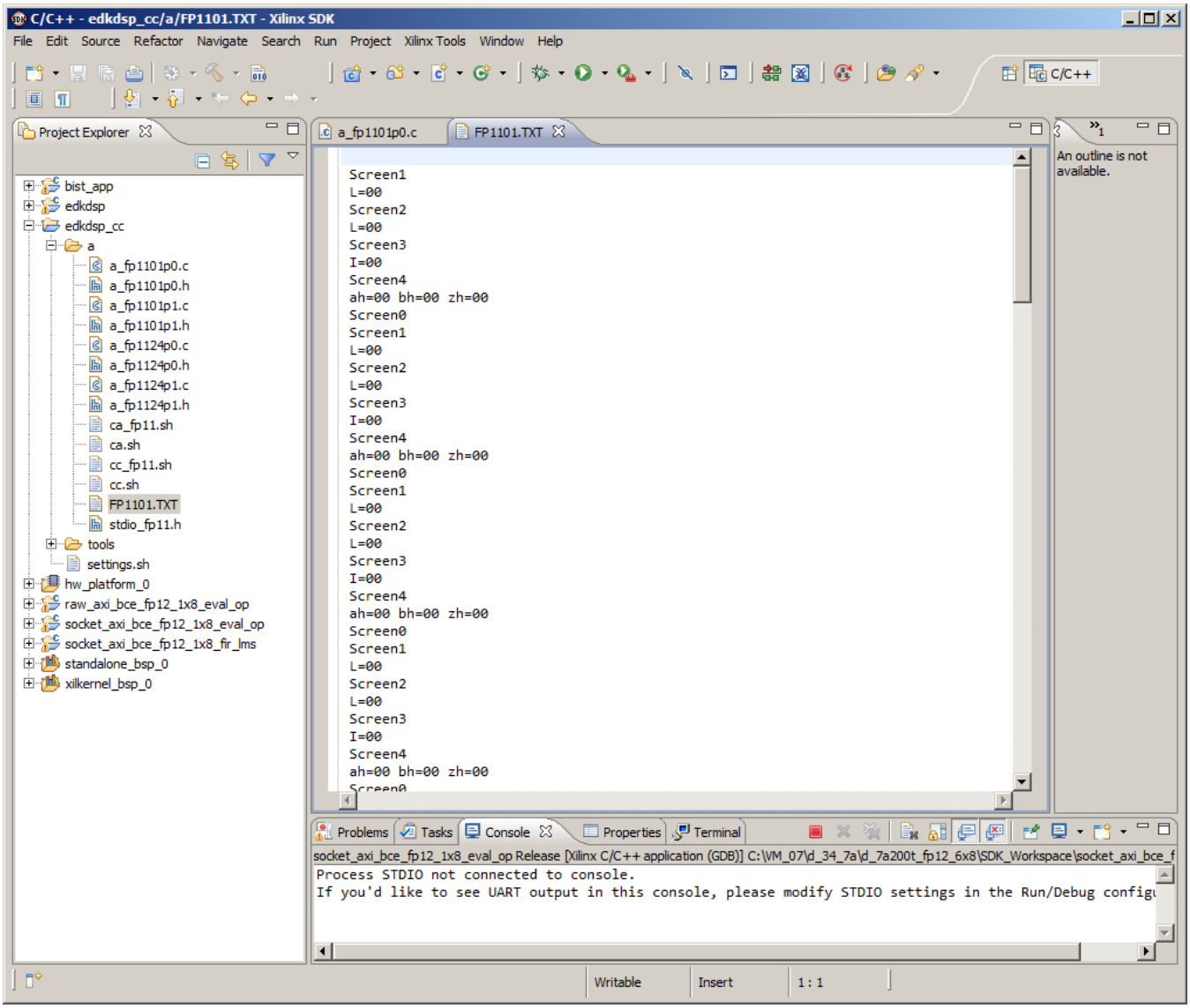

Figure 39: In SDK, Refresh the edkdsp_cc/a directory (by F5) to see the received FP1101.TXT file downloaded from the server running on the Artix7 FPGA. Notice that the input data are printed as I=00.

Refresh the project explorer view by F5. The uploaded log file FP1101.TXT can be open. See Figure 39. The PicoBlaze6 original firmware is writing I=00 to the log file as expected.

Keep the application running on the Artix7 together with the browser GUI.

Compile the modified firmware source code by script cc_fp11.sh with parameter a.
Type in the Ubuntu terminal:
$ cc_fp11.sh a

This will compile and assemble all four C firmware programs to header files with the firmware binary code (See Figure 40):
a_fp1101p0.c is compiled to FP1101P0.DEC
a_fp1101p1.c is compiled to FP1101P1.DEC
a_fp1124p0.c is compiled to FP1124P0.DEC
a_fp1124p1.c is compiled to FP1124P0.DEC

This compiled firmware can be uploaded from PC to the running demo application in the Artix7 chip.

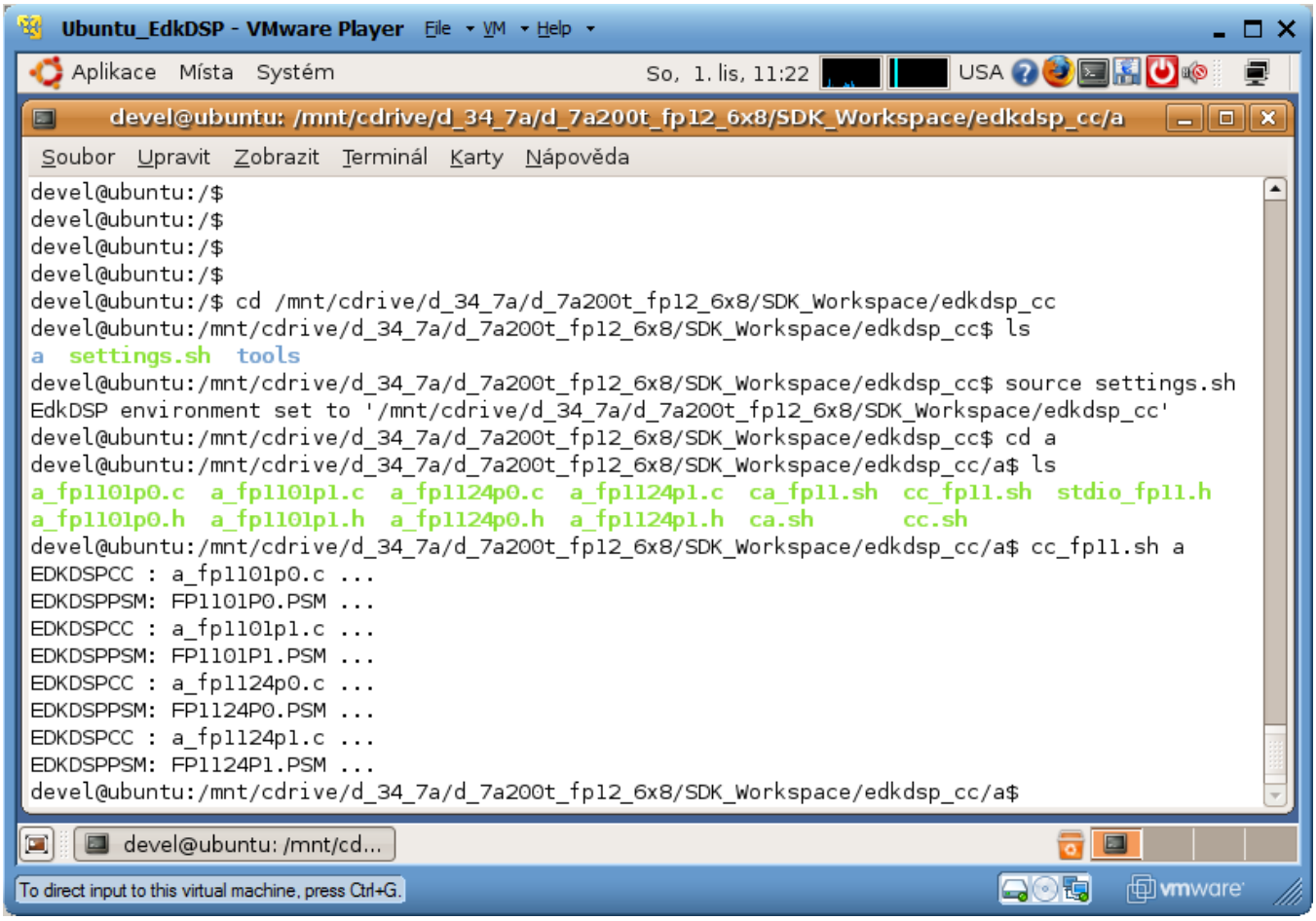

Figure 40: Compile the C code with uncommented lines to display Input=00 instead of I=00

Upload the compiled firmware from PC to the Artix7 File system. See Figure 41 - Figure 44.

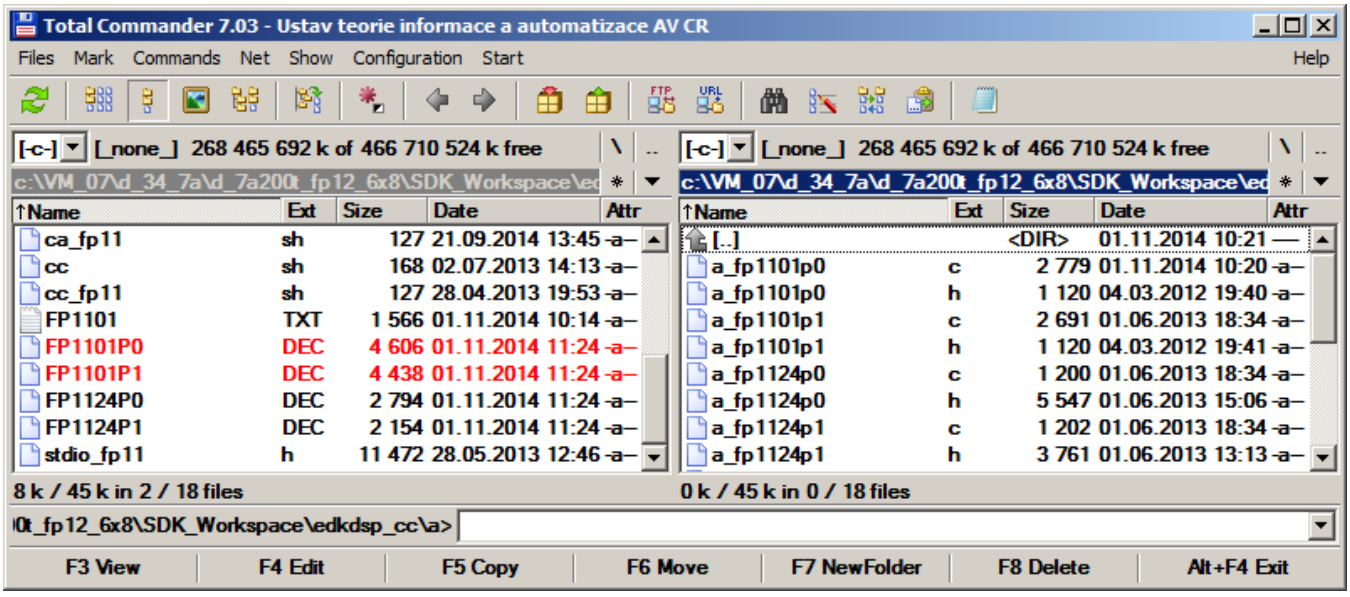

Figure 41: Select compiled binaries and download them to the Artix7 FPGA by the TFTP client.

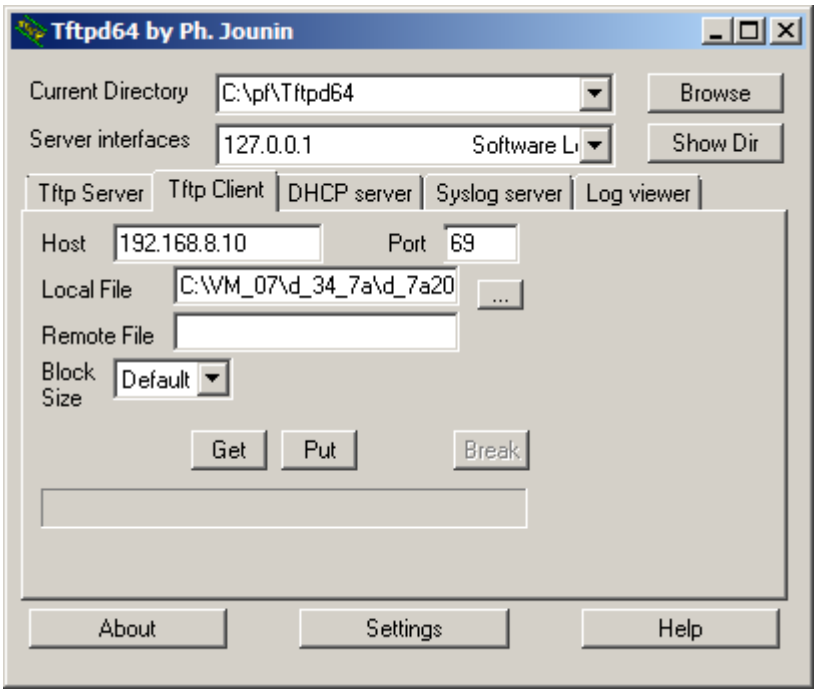

Figure 42: Drag and drop the 2 binary program files to upload them to the Artix7 file system.

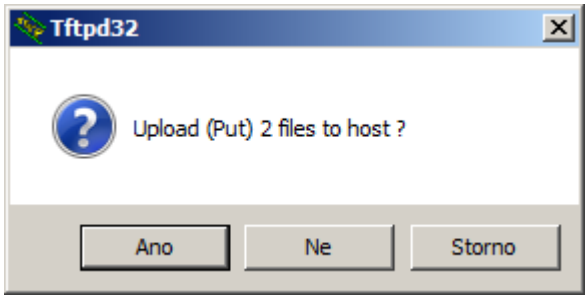

Figure 43: Confirm Ano (yes in Czech…)

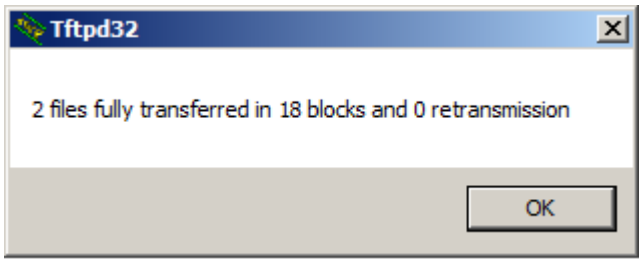

Figure 44: The TFTP server is indicating number of blocks uploaded to Artix7 file system.

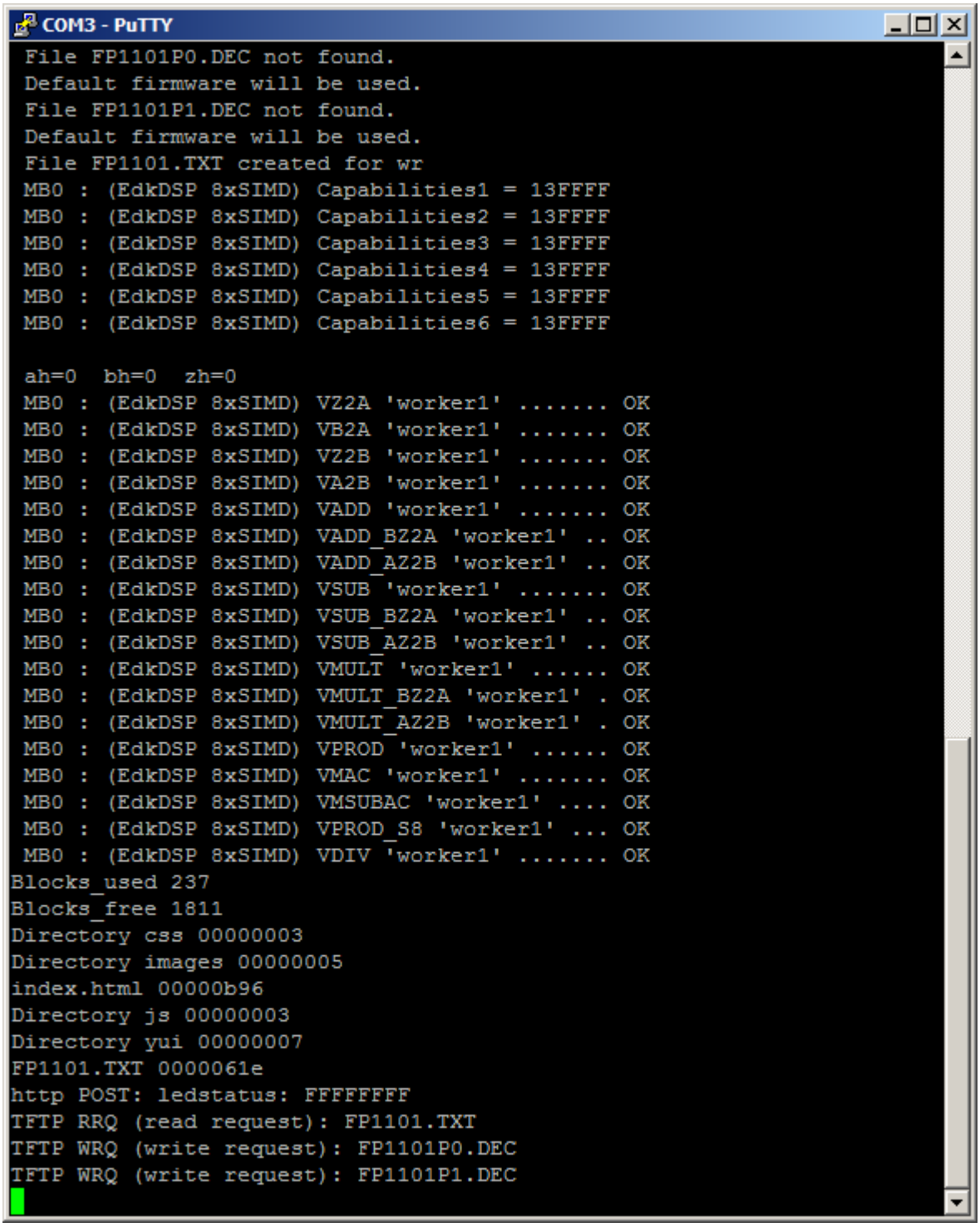

Figure 45: The TFTP server is indicating on the terminal that the 2 files have been received.

The TFTP server running on the Artix7 MicroBlaze is informing about the uploaded firmware files. See the last two lines in Figure 45.

Start second test of the EdkDSP accelerator by clicking on the Toggle LEDs button in the www browser user interface. See Figure 46. Firmware files have been found, and firmware of the tested EdkDSP accelerator have been updated. Tests have been performed and the log file FP1100.TXT stored in the Artix7 RAM based file system. See Figure 46.

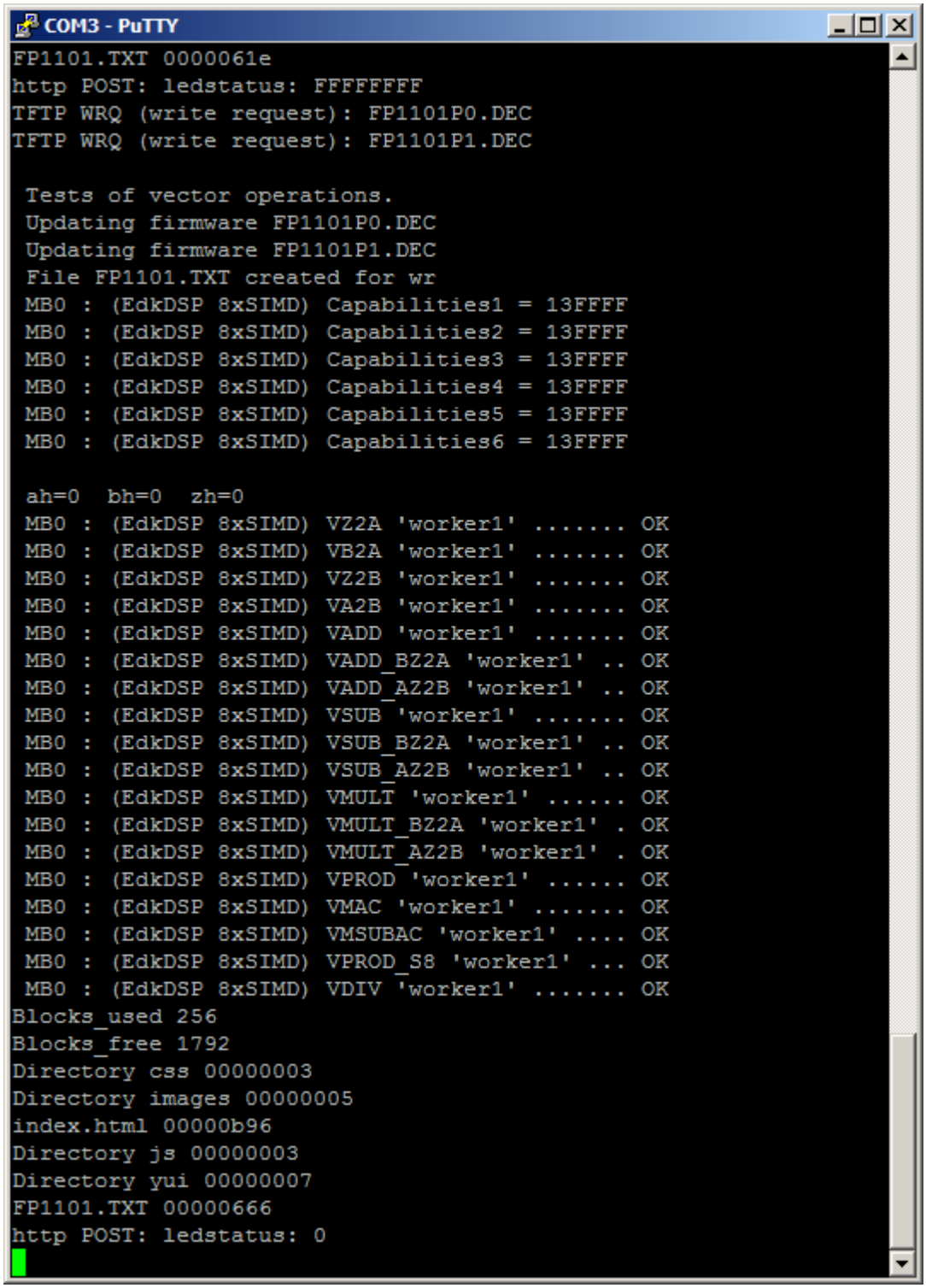

Figure 46: Run next test from the web browser GUI, download the resulting file FP1101.TXT to PC, refresh the edkdsp_cc/a directory and see the file in the SDK. Input=00 is now written to the file. This corresponds to the modified C source for the EdkDSP accelerator.

Download the FP1101.TXT file to PC with the TFTP client applicatiob, and see its cintent in the SDK. The messages from the tested EdkDSP accelerator have been modified to Input=00. See Figure 47.

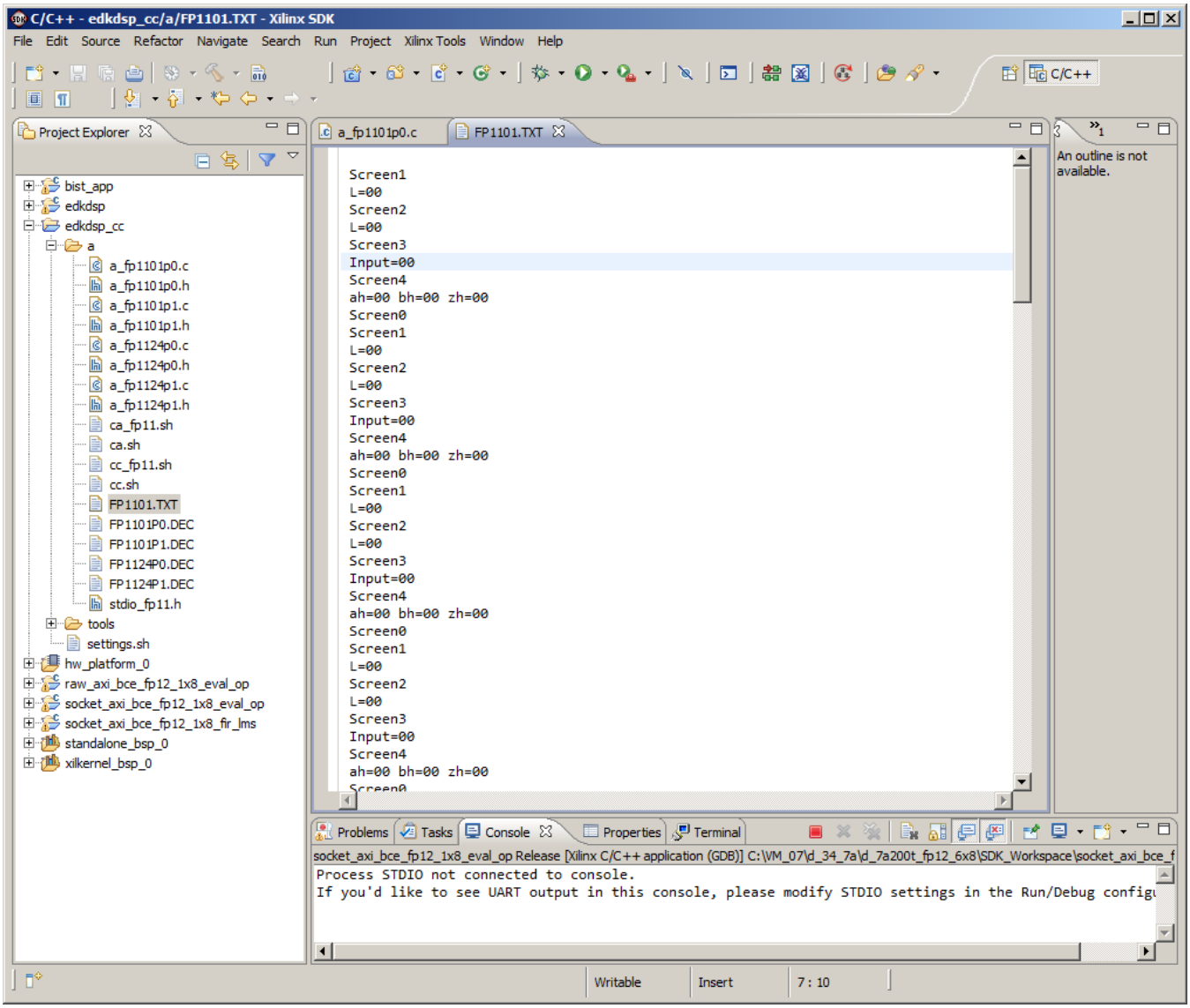

Figure 47: The console output indicates that 2 firmware files have been found and used to reprogram the tested EdkDSP accelerator before test. The resulting FP1101.TXT is bigger due to the longer output text (I=00) replaced bt (Input=00).

We have demonstrated the process of compilation, download of files from the Artix7 chip to PC and upload of compiled firmware from PC to the Artix7 and its EdkDSP accelerators.

Close the browser application and stop the application on the Artix7 MicroBlaze processor.

3.7 Use of the C compiler for the EdkDSP firmware witout Ethernet

This section is describing the use of the UTIA EdkDSP C compiler to recompile the firmware for the PicoBlaze6 controller present in each of the six (8xSIMD) EdkDSP accelerators in the AC701 board for simple application without internet connectivity. The edkdsp project in the SDK project explorer will be used as an example.

The firmware C source code examples can be compiled by the script ca_fp11.sh with parameter a.
Type in the Ubuntu terminal (See Figure 48):
$ ca_fp11.sh a

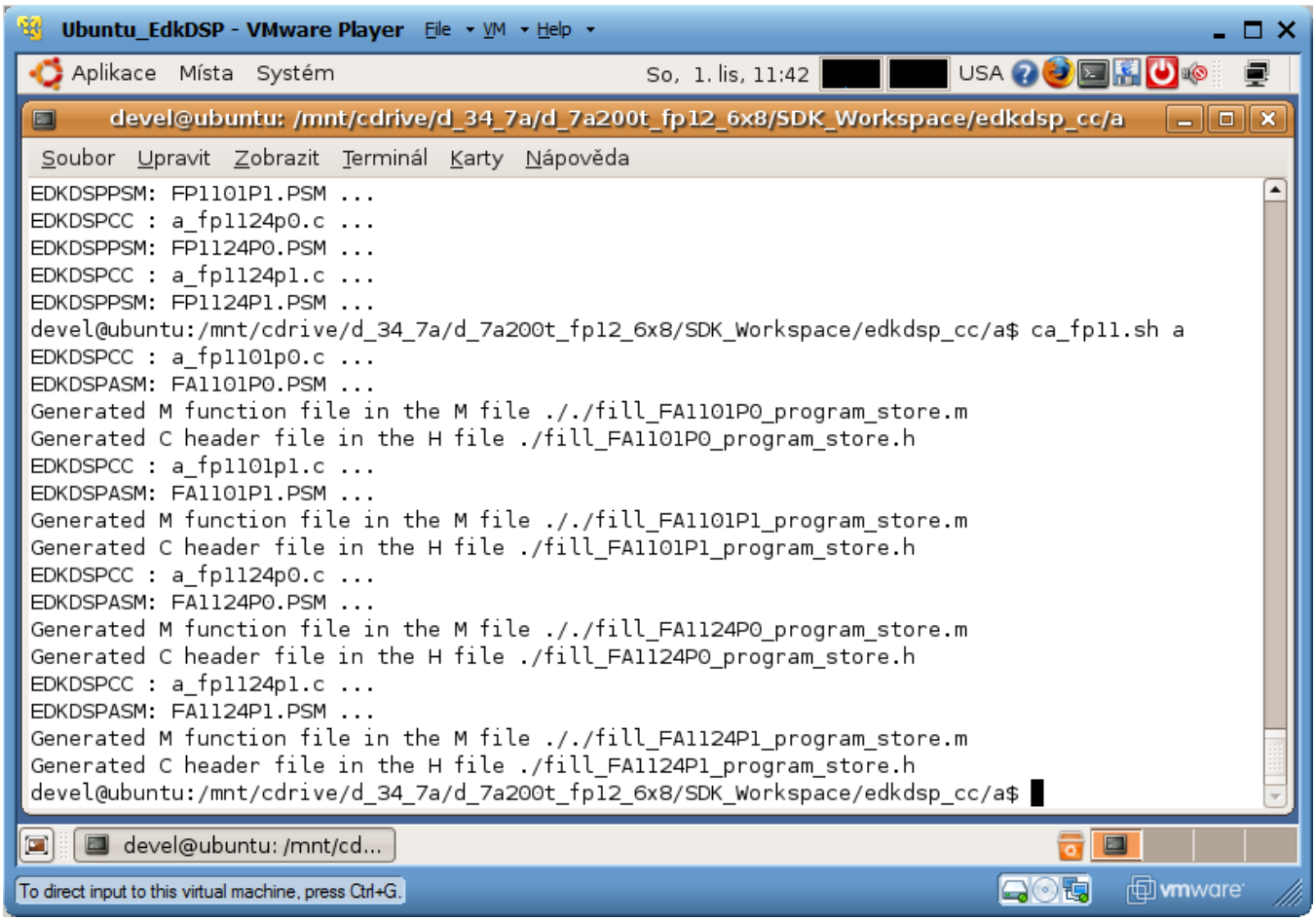

Figure 48: Compile the C source code for the accelerator by the EDKDSPCC compiler with the edkdspasm assembler. It will create the assembler source code and firmware binary in format of C .h header files. These headers can be used for inclusion into the edkdsp demo project (without the TFTP file server).

This will compile and assemble all four C firmware programs to header files with the firmware binary code:
a_fp1101p0.c is compiled to fill_FA1101P0_program_store.h
a_fp1101p1.c is compiled to fill_FA1101P1_program_store.h
a_fp1124p0.c is compiled to fill_FA1124P0_program_store.h
a_fp1124p1.c is compiled to fill_FA1124P0_program_store.h

Copy and paste the compiled headers into the src directory of the MicroBlaze project “edkdsp” of the SDK

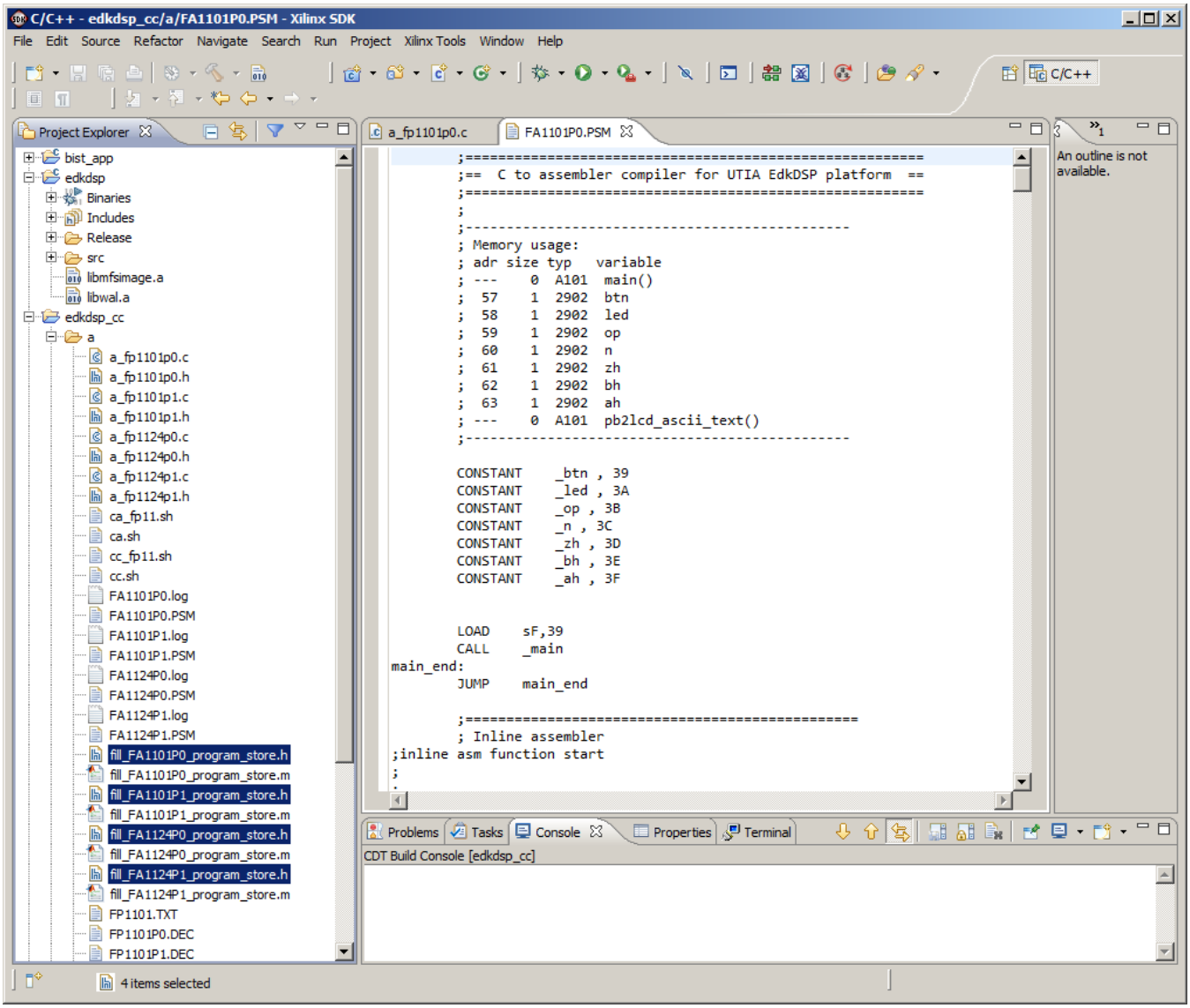

Figure 49: Select firmware header files and Ctrl-C Ctrl-V them to the edkdsp/src directory.

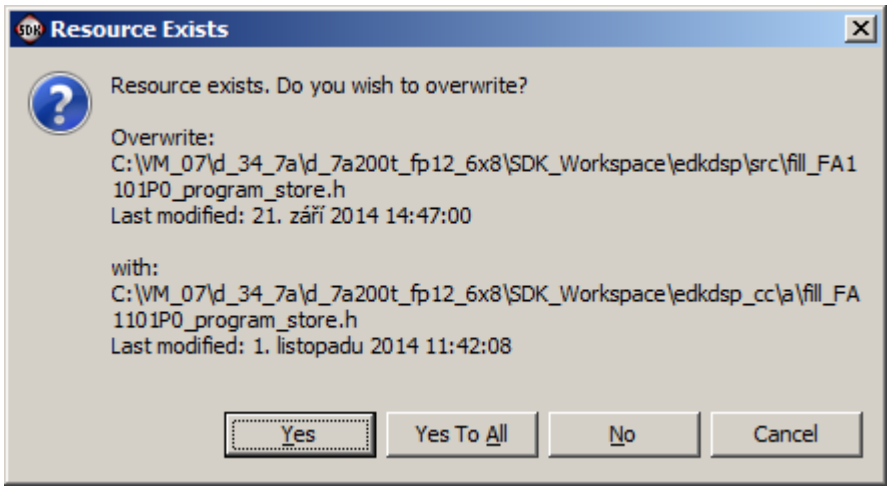

Figure 50: Confirm to overwrite multiple files

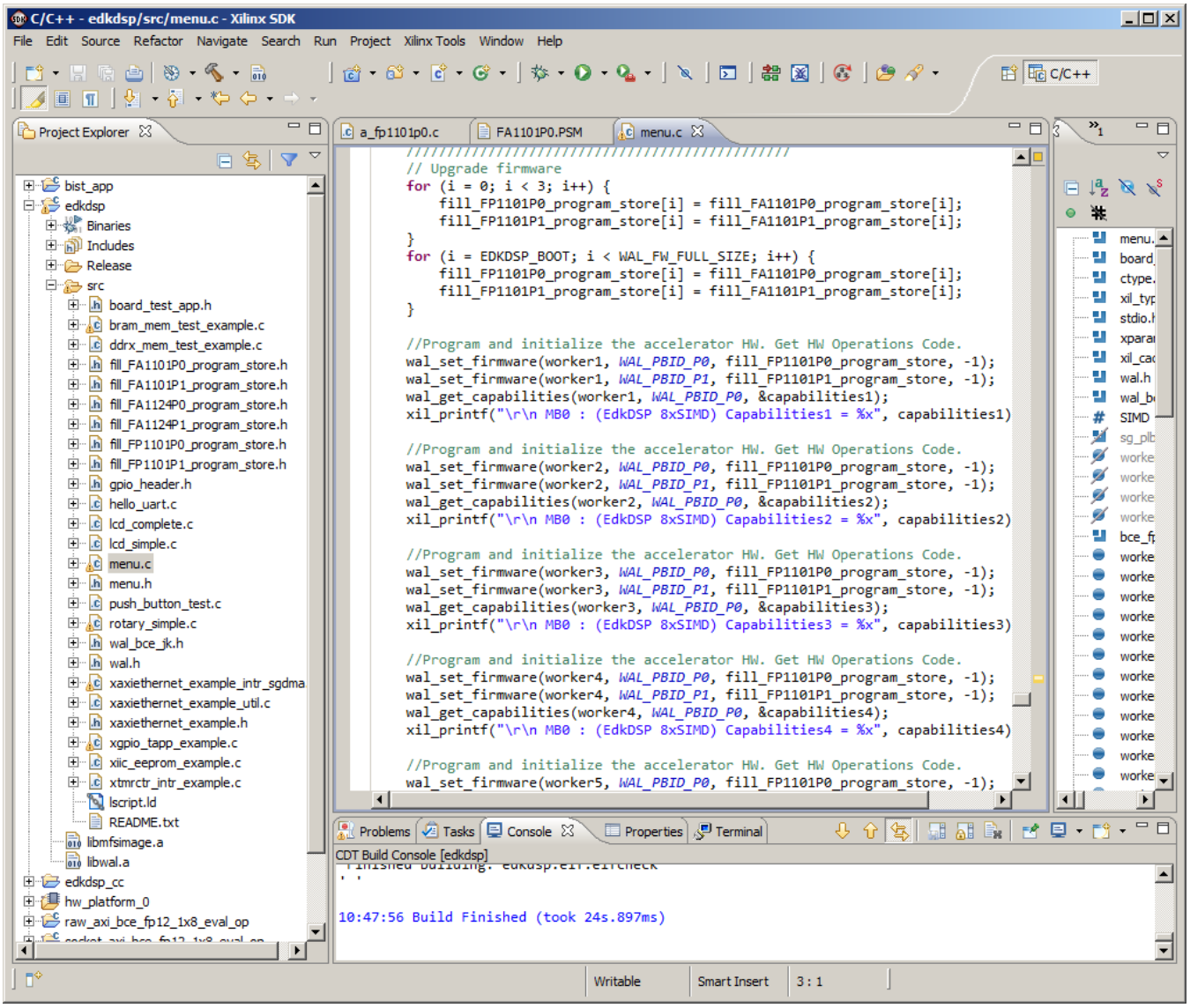

Figure 51: See the updated edkdsp/src directory and section of the Microblaze source code, where the recompiled modified firmware is updated and EdkDSP accelerators are programmed.

Notice also the listing of the firmware in the assembler in Figure 49.
Figure 51 is presenting the firmware update section of the C code in the Microblaze edkdsp project.

In SDK, recompile the edkdsp project, to reflect the change of the firmware in header files.

To test new firmware, download the bitstream, and run the recompiled edkdsp.elf application. See Figure 52.

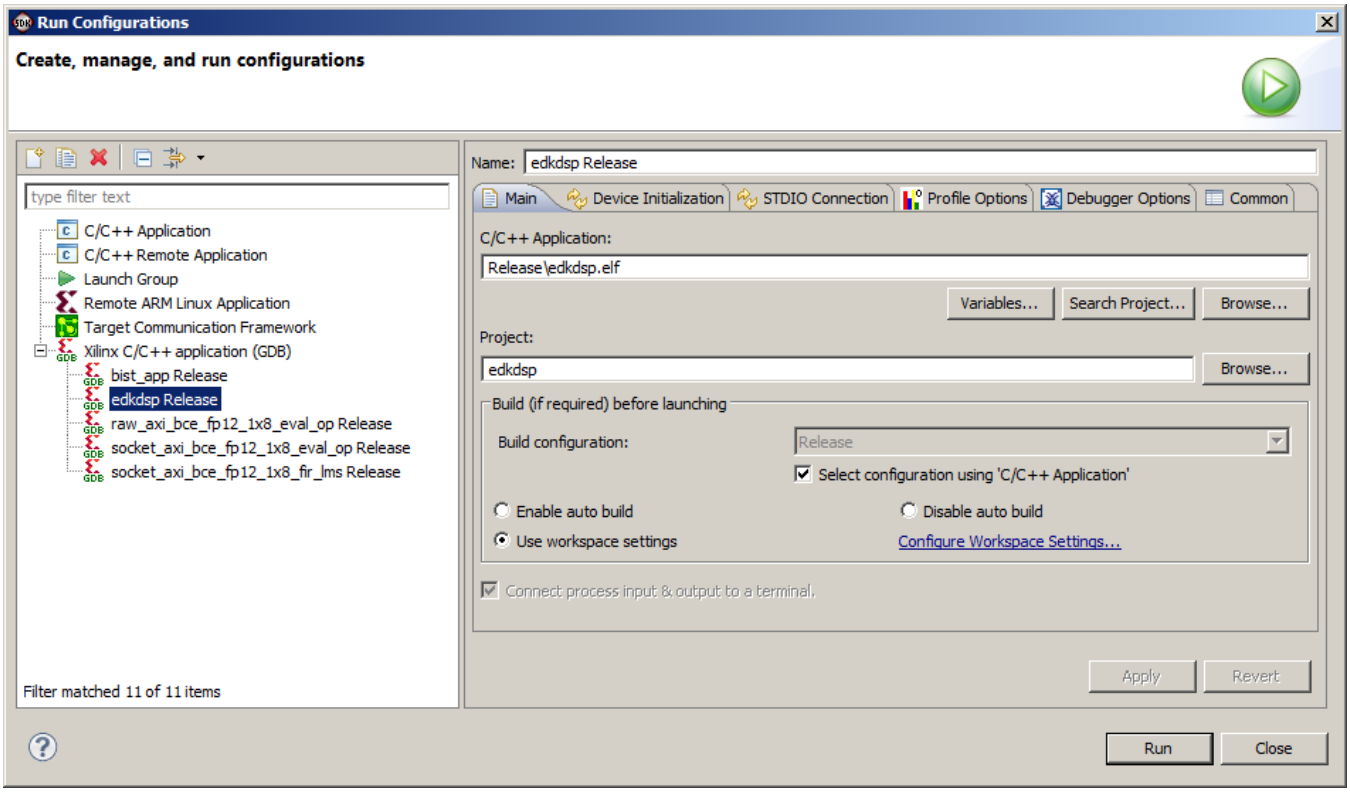

Figure 52: Recompile edkdsp project, download the .bit file and run the edkdsp.elf on Artix.

Figure 53 is presenting the initial menu of the edkdsp application.
Type C to select test of the EdkDSP operations.
Figure 54 is presenting results of the test of the EdkDSP accelerator with modified firmware.
Type 0 to wxit from the edkdsp simple menu.
Close the debug session from SDK console (the X icon).

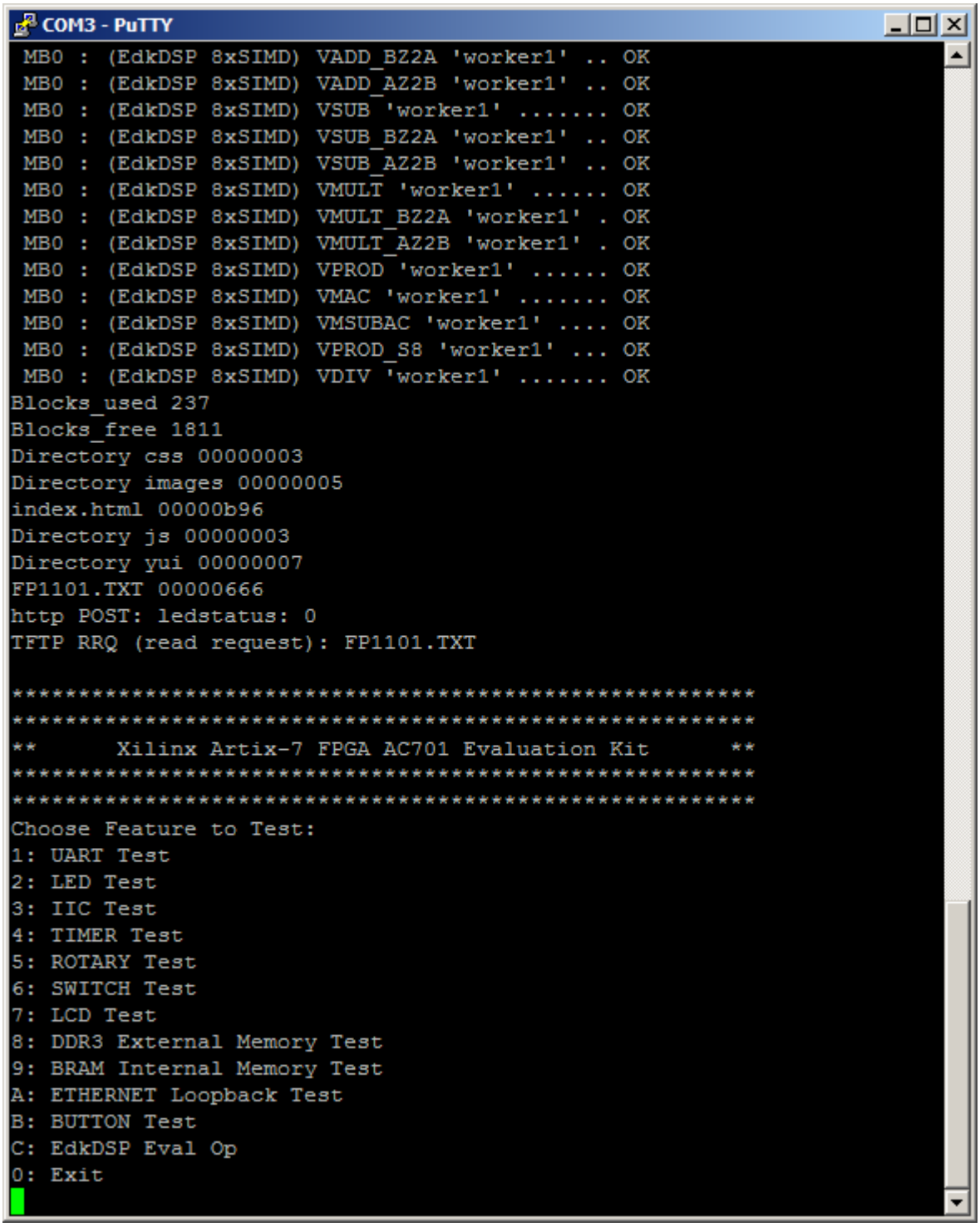

Figure 53: Test the EdkDSP accelerator with the new firmware from the menu (type C)

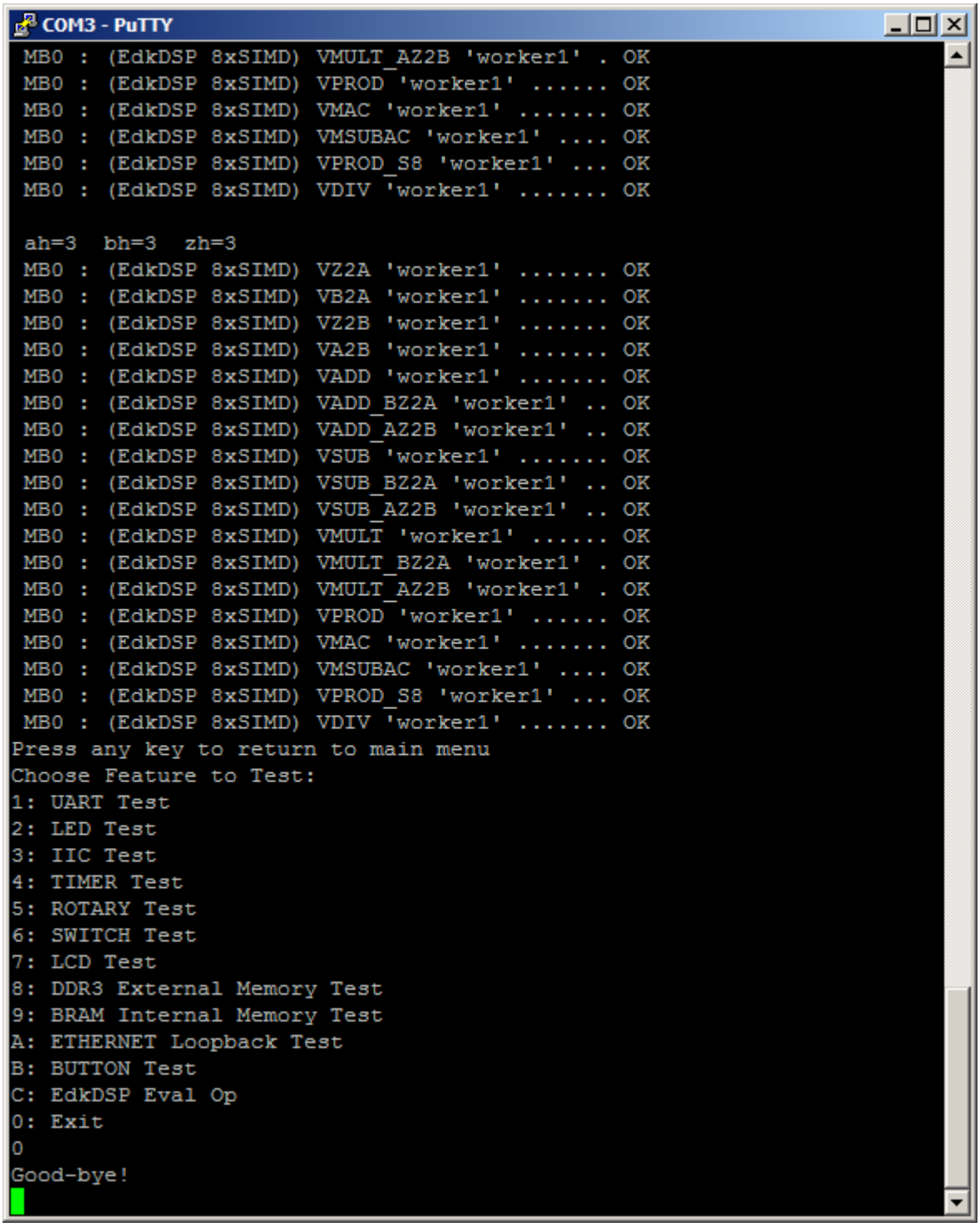

Figure 54: See the result of test of all basic vector operations performed on the EdkDSP accelerator with the recompiled firmware. Results remain identical with the Microblaze reference.

4. References

[1] AC701 Evaluation Board, for the Artix-7 FPGA User Guide UG952 (v1.2) August 28, 2013
http://www.xilinx.com/support/documentation/boards_and_kits/ac701/ug952-ac701-a7-eval-bd.pdf

[2] AC701 Built - In Self Test, Flash Application, December 2013
http://www.xilinx.com/support/documentation/boards_and_kits/ac701/2013_4/xtp194-ac701-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] Eniac JU project THINGS2DO “Thin but Great Silicon 2 Design Objects”, project number ENIAC JU 621221.
http://things2do.space.com.ro/
http://sp.utia.cz/index.php?ids=projects/things2do

5. Evaluation version of Vivado 2013.4 Artix7 designs

The enclosed **Evaluation version of precompiled Vivado 2013.4 Artix7 designs with evaluation versions of UTIA (8xSIMD) EdkDSP accelerator cores** can be downloaded from UTIA www pages free of charge and used for evaluation together with the six UTIA (8xSIMD) EdkDSP accelerators.

The evaluation package includes one DVD or the www download package with these deliverables:

8 precompiled designs with UTIA (8xSIMD) EdkDSP accelerators for Xilinx Artix7 AC701 board [1], [2] compiled in Xilinx Vivado 2013.4. The UTIA (8xSIMD) EdkDSP accelerators are compiled with HW limit on number of vector operations. The termination of the evaluation license is reported in advance by the demonstrator on the terminal.

The evaluation package includes SDK 2013.4 SW projects with source code for MicroBlaze processor. SW projects support the family of UTIA (8xSIMD) EdkDSP accelerators for the Xilinx AC701 board [1], [2].

The evaluation package includes this compiled library:

libwal.a	EdkDSP api (SDK 2013.4, MicroBlaze) for EdkDSP accelerators on AC701 board.
libmfsimage.a	The library with file system supporting simple www server GUI.

The library **libwal.a** has no time restriction. The evaluation license is provided by UTIA only for the use with the family of UTIA EdkDSP accelerators designed for the Xilinx AC701 board. Source code of this library is owned by UTIA and it is not provided in this evaluation package.

The evaluation package includes these binary applications for Ubuntu:

edkdsppp	EdkDSP C pre-processor binary for Ubuntu (x86 PC) under the VMware Player.
edkdspcc	EdkDSP C compiler binary for Ubuntu (x86 PC) under the VMware Player.
edkdspasm	EdkDSP ASM compiler binary for Ubuntu (x86 PC) under the VMware Player.
edkdsppsm	EdkDSP ASM compiler binary for Ubuntu (x86 PC) under the VMware Player.

These binary applications have no time restriction. The user of the evaluation package has license from UTIA to use these utilities for compilation of the firmware for the Xilinx PicoBlaze6 processor inside of the UTIA EdkDSP accelerators in the 8 precompiled designs for the Xilinx AC701 board. The source code of these compilers is owned by UTIA and it is not provided in the evaluation package.

The evaluation package includes demonstration firmware in C source code for the Xilinx PicoBlaze6 processor for the family of UTIA EdkDSP accelerators for the Xilinx AC701 board.

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 Xilinx AC701 board without the need to install the UTIA compiler binaries and the Ubuntu (x86 PC) OS image under the VMware Player.

On email request to kadlec@utia.cas.cz , UTIA will send 2 DVD CDs (8GB) with the Ubuntu (x86 PC) image for the VMware Player free of charge.

6. Release version of Vivado 2013.4 Artix7 designs for THINGS2DO project partners

The **release version of Vivado 2013.4 Artix7 designs with evaluation versions of UTIA (8xSIMD) EdkDSP accelerator cores for THINGS2DO [6] project partners** can be ordered from UTIA AV CR, v.v.i., by email request for quotation to kadlec@utia.cas.cz. UTIA will provide quotation by email. After the confirmed order received by email to kadlec@utia.cas.cz, UTIA AV CR, v.v.i. will deliver (by standard mail to the THINGS2DO project partners) a printed version of this application note together with 3 DVDs with deliverables described in this section. UTIA AV CR, v.v.i., will also send to the THINGS2DO project partner (by email) and by the standard mail the invoice for:

Release version of Vivado 2013.4 Artix7 designs
with evaluation versions of UTIA (8xSIMD) EdkDSP accelerator cores
for THINGS2DO [6] project partners (without VAT) 0,00 Eur

The package includes this application note and the EdkDSP DVD with these deliverables:

8 precompiled designs with UTIA (8xSIMD) EdkDSP accelerators for Xilinx AC701 board, compiled in Xilinx Vivado 2013.4. The UTIA (8xSIMD) EdkDSP accelerators are compiled with HW limit on number of vector operations. The termination of the evaluation license is reported in advance by the demonstrator on the terminal.

The **Release version of Vivado 2013.4 Artix7 designs with evaluation versions of UTIA (8xSIMD) EdkDSP accelerator cores for THINGS2DO [6] project partners** include all 8 Vivado 2013.4 design projects and the evaluation versions of the UTIA (8xSIMD) EdkDSP accelerators provided in form of netlisted IP cores generated in Xilinx VIVADO 2013.4:

bce_fp11_1x8_0_axiw_v1_10_a
bce_fp11_1x8_0_axiw_v1_20_a
bce_fp11_1x8_0_axiw_v1_30_a
bce_fp11_1x8_0_axiw_v1_40_a
bce_fp12_1x8_0_axiw_v1_10_a
bce_fp12_1x8_0_axiw_v1_20_a
bce_fp12_1x8_0_axiw_v1_30_a
bce_fp12_1x8_0_axiw_v1_40_a

These evaluation versions of UTIA (8xSIMS) EdkDSP netlist pcores are compiled with an HW limit on number of vector operations. **THINGS2DO [6] project partners** have license from UTIA to integrate these evaluation netlists into their own VIVADO 2013.4 designs and to compile them to unlimited number of bit-streams for designs on Xilinx Artix7 FPGAs. This license has no time restriction. The source code of the evaluation versions of (8xSIMS) EdkDSP accelerators is an IP owned by UTIA and it is not provided in the release package to the THINGS2DO project partners.

The package for the THINGS2DO [6] project partners includes the SDK 2013.4 SW projects in source code for MicroBlaze as described in this application note. Projects support the evaluation versions of the UTIA (8xSIMD) EdkDSP accelerators (in the netlist pcore format) for the Xilinx AC701 board.

The package for the THINGS2DO project partners includes the library:

libwal.a	EdkDSP api (SDK 2013.4, MicroBlaze) for EdkDSP accelerators on AC701 board.
libmfsimage.a	The library with file system supporting simple www server GUI.

The library **libwal.a** has has no time restriction. The evaluation license is provided by UTIA only for the use with the family of UTIA EdkDSP accelerators designed for the Xilinx AC701 board. Source code of this library is owned by UTIA and it is not provided in this evaluation package.

The package for the THINGS2DO project partners includes these binary applications for Ubuntu:

edkdsppp	EdkDSP C pre-processor binary for Ubuntu (x86 PC) under the VMware Player.
edkdspcc	EdkDSP C compiler binary for Ubuntu (x86 PC) under the VMware Player.
edkdspasm	EdkDSP ASM compiler binary for Ubuntu (x86 PC) under the VMware Player.
edkdsppsm	EdkDSP ASM compiler binary for Ubuntu (x86 PC) under the VMware Player.

These binary applications have no time restriction. The THINGS2DO project partners have license from UTIA to use these utilities for compilation of the firmware for the Xilinx PicoBlaze6 processor inside of the UTIA EdkDSP accelerators in the 10 precompiled designs for the Xilinx AC701 board. The source code of these binaries is owned by UTIA and it is not provided in the evaluation package.

The package includes demonstration firmware in C source code for the Xilinx PicoBlaze6 processor for the family of UTIA EdkDSP accelerators for the Xilinx AC701 board.

The package also includes compiled versions of this firmware in form of header files .h. These compiled firmware files can be used to evaluate the UTIA EdkDSP accelerators on the Xilinx AC701 board without the need to install the UTIA compiler binaries and the Ubuntu (x86 PC) OS image under the VMware Player.

The release package deliverables also includes two DVDs with the Ubuntu (x86 PC) image for the VMware Player (free of charge). This image is provided to ease the installation of the UTIA EdkDSP C compiler on Windows 7 (32bit or 64bit) in the VMware Player.

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.

7. Release version of Vivado 2013.4 Artix7 designs

The **release version of Vivado 2013.4 Artix7 designs with the release version of the UTIA (8xSIMD) EdkDSP accelerator cores** can be ordered from UTIA AV CR, v.v.i., by email request for quotation to kadlec@utia.cas.cz. UTIA will provide quotation by email. After the confirmed order received by email to kadlec@utia.cas.cz, UTIA AV CR, v.v.i. will deliver (by standard mail) to the customer the printed version of this application note together with 3 DVDs with deliverables described in this section. UTIA AV CR, v.v.i., will send to the customer (by email) and by the standard mail the invoice for:

Release version of Vivado 2013.4 Artix7 designs with the release version of UTIA (8xSIMD) EdkDSP accelerator cores (without VAT) 400,00 Eur

The release package includes this application note and the EdkDSP DVD with these deliverables:

8 precompiled designs with UTIA (8xSIMD) EdkDSP accelerators for Xilinx AC701 board [2], compiled in Xilinx Vivado 2013.4. The UTIA (8xSIMD) EdkDSP accelerators included in these designs are compiled with **no HW limit on number of vector operations.** Therefore, all these precompiled designs of the release package run on AC701 without limitations of the evaluation package.

The release package includes all 8 Vivado 2013.4 design projects. The UTIA (8xSIMD) EdkDSP accelerators are provided in the form of netlist IP cores generated in Xilinx VIVADO 2013.4:

bce_fp11_1x8_0_axiw_v1_10_a
bce_fp11_1x8_0_axiw_v1_20_a
bce_fp11_1x8_0_axiw_v1_30_a
bce_fp11_1x8_0_axiw_v1_40_a
bce_fp12_1x8_0_axiw_v1_10_a
bce_fp12_1x8_0_axiw_v1_20_a
bce_fp12_1x8_0_axiw_v1_30_a
bce_fp12_1x8_0_axiw_v1_40_a

These UTIA (8xSIMS) EdkDSP netlist pcores have **no HW limit on number of vector operations.** The user of the release package has license from UTIA to integrate these netlists into its own VIVADO 2013.4 designs and to compile them to unlimited number of bit-streams. This license has no time restriction. The source code of the (8xSIMS) EdkDSP accelerators is an IP owned by UTIA and it is not provided in the release package to the customer.

The release package includes SDK 2013.4 SW projects in source code for MicroBlaze as described in this application note. Projects support the family of UTIA (8xSIMD) EdkDSP accelerators for Xilinx AC701 board [2].

The release package includes the library:

libwal.a	EdkDSP api (SDK 2013.4, MicroBlaze) for EdkDSP accelerators on AC701 board.
libmfsimage.a	The library with file system supporting simple www server GUI.

The library **libwal.a** has has no time restriction. The evaluation license is provided by UTIA only for the use with the family of UTIA EdkDSP accelerators designed for the Xilinx AC701 board. Source code of this library is owned by UTIA and it is not provided in this release package.

The release package includes these binary applications for Ubuntu:

edkdsppp	EdkDSP C pre-processor binary for Ubuntu (x86 PC) under the VMware Player.
edkdspcc	EdkDSP C compiler binary for Ubuntu (x86 PC) under the VMware Player.
edkdspasm	EdkDSP ASM compiler binary for Ubuntu (x86 PC) under the VMware Player.
edkdsppsm	EdkDSP ASM compiler binary for Ubuntu (x86 PC) under the VMware Player.

These binary applications have no time restriction. The user of the evaluation package has license from UTIA to use these utilities for compilation of the firmware for the Xilinx PicoBlaze6 processor inside of the UTIA EdkDSP accelerators in the 8 precompiled designs for the Xilinx AC701 board. The source code of these compilers is owned by UTIA and it is not provided in the release 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 Xilinx AC701 board.

The release package also includes compiled versions of this firmware in form of header files .h. These compiled firmware files can be downloaded into the UTIA EdkDSP accelerators for the Xilinx AC701 board without the need to install UTIA compiler binaries and the Ubuntu (x86 PC) OS under the VMware Player.

The release package deliverables also includes two DVDs with the Ubuntu (x86 PC) image for the VMware Player (free of charge). This image is provided to ease the installation of the UTIA EdkDSP C compiler on Windows 7 (32bit or 64bit) in the VMware Player.

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.

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.