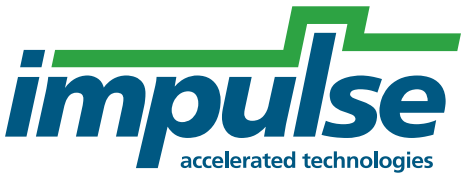


Application Note

Using Impulse C with BlueCat Linux 5.4.2 on MicroBlaze via FSL

Mei A. Xu
Impulse Accelerated Technologies, Inc.

Overview

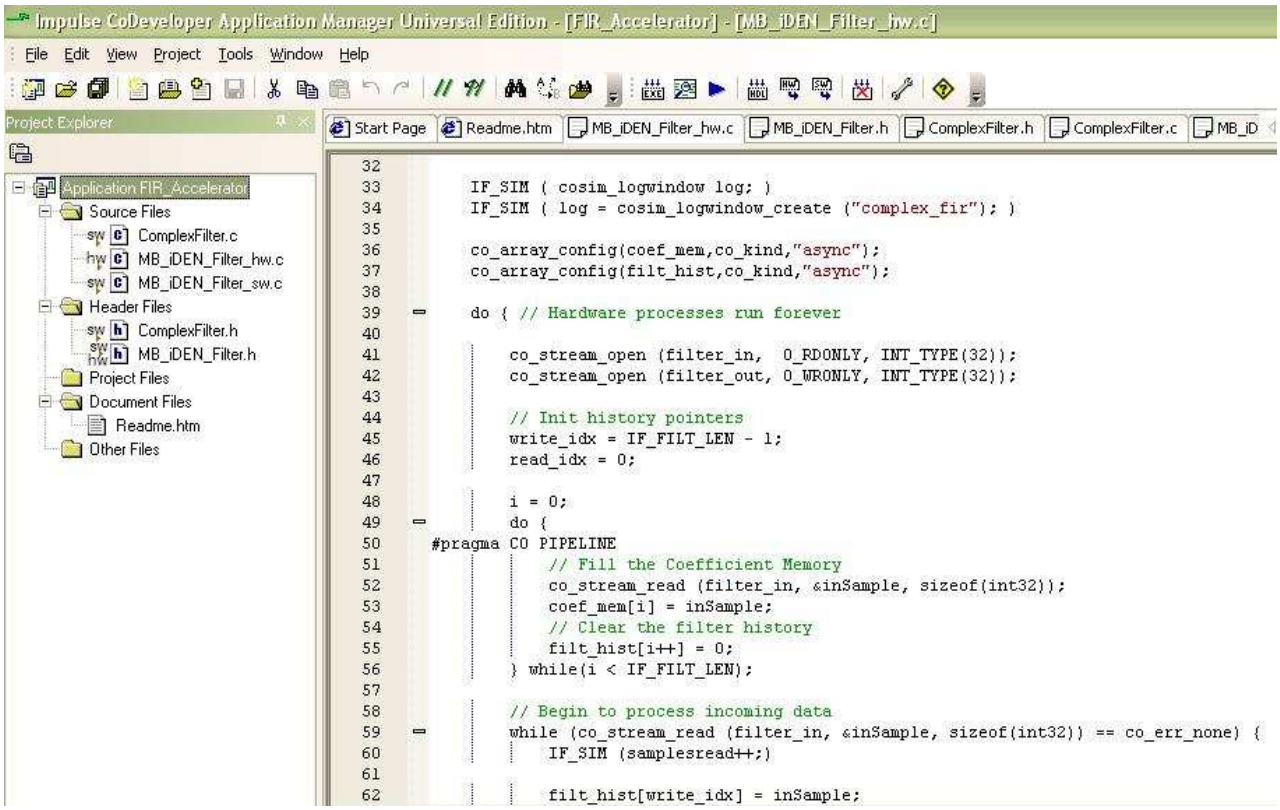
This application note describes how to incorporate an Impulse C generated hardware module with a MicroBlaze microprocessor running the LynuxWorks BlueCat Linux 5.4.2 operating systems, using the Xilinx FSL interface for data communication. A complex FIR filter application is used as an example to demonstrate the design method.

This example makes use of the Spartan-3A DSP 1800 Board, which features is a Spartan-3A FPGA with a MicroBlaze soft processor. This tutorial also assumes you are using the Xilinx EDK 9.2 (or later) development tools. BlueCat Linux package version 5.4.2 from LynuxWorks provides a Linux kernel and the development tools needed to compile Linux for the MicroBlaze embedded processor.

This application note assumes you are already familiar with the design flow from Impulse C into the Xilinx EDK tools. If you are not familiar with this design flow, please read the MicroBlaze tutorials provided with Impulse CoDeveloper, in the Help and Support section of the CoDeveloper Start Page.

Complex FIR Example in CoDeveloper

In CoDeveloper, open the Complex FIR project as shown below. For details about this example, please refer to Impulse C Xilinx MicroBlaze Tutorial #2.



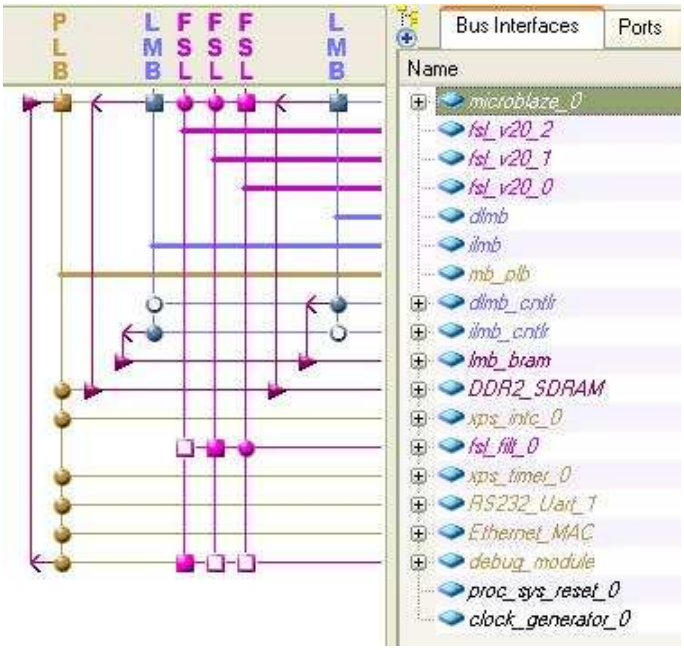
In the Project Options Dialogue, Generate Tab, choose “Xilinx BlueCat Linux MicroBlaze FSL” as the Platform Support Package {PSP).

Then generate HDL and export the generated hardware and software using the “Generate HDL”, “Export Hardware” and “Export Software” toolbar buttons.

Building Hardware in Xilinx EDK

In EDK, created a new project called ComplexFIR. Refer to Xilinx MicroBlaze Tutorial #2 for details of how to set up the MicroBlaze and its peripherals.

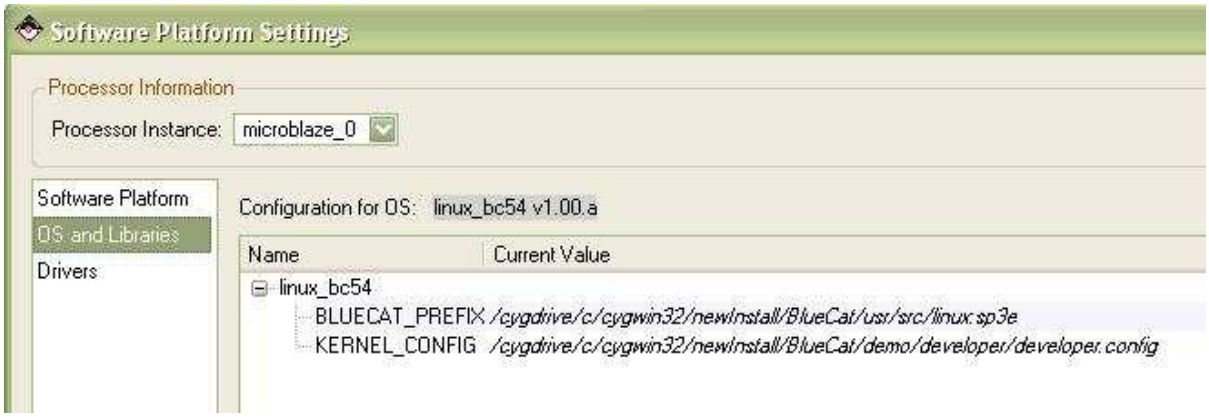
When using BlueCan Linux, besides the two FSLs for input and output streams, a third FSL is used to connect MicroBlaze and the debugger module XMD for fast downloading of the Linux image.



The address map of the system is as follows:

Instance	Peripheral	Base Address	High Address
dlmb_cntlr/ilmb_cntlr	lmb_bram_if_cntlr	0x00000000	0x00001fff
RS232_Uart_1	xps_uartlite	0x84000000	0x8400ffff
Ethernet_MAC	xps_ethernetlite	0x81000000	0x8100ffff
DDR2_SDRAM	mpmc	0x88000000	0x8fffffff
xps_intc_0	xps_intc	0x81800000	0x8180ffff
xps_timer_0	xps_timer	0x83c00000	0x83c0ffff
debug_module	mdm	0x84400000	0x8440ffff
fsl_v20_0/1/2	fsl_v20		

In Software Platform Settings, choose “linux_bc54” as OS. In the OS and Libraries Tab, set the parameters as follows:



Then generate libraries and drivers. This action will update the configuration files for the Linux kernel. Generate a bitstream for the design. Choose “bootloop” as the software application for BRAM initialization.

Download the bitstream to the FPGA.

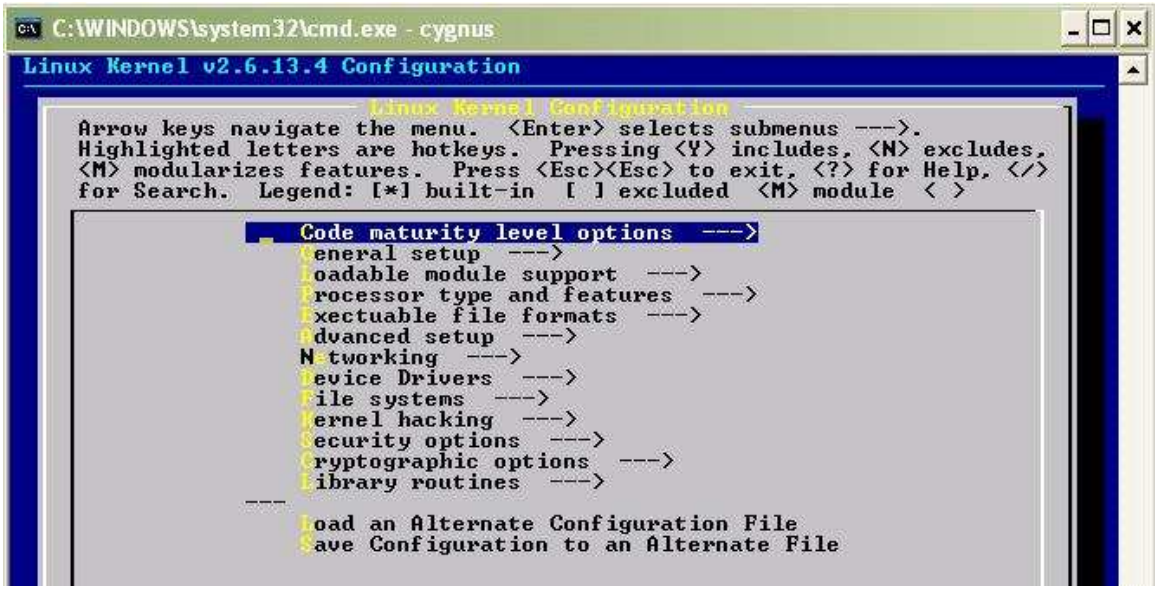
Setting Up BlueCat Linux Environment

Install BlueCat Linux 5.4.2 according to the instruction provided by LynuxWorks. Here the installation directory is: C:\cygwin32. The patch “5p4p2PATCH” is needed to install the FSL driver.

The following command lines are used to set up the environment:

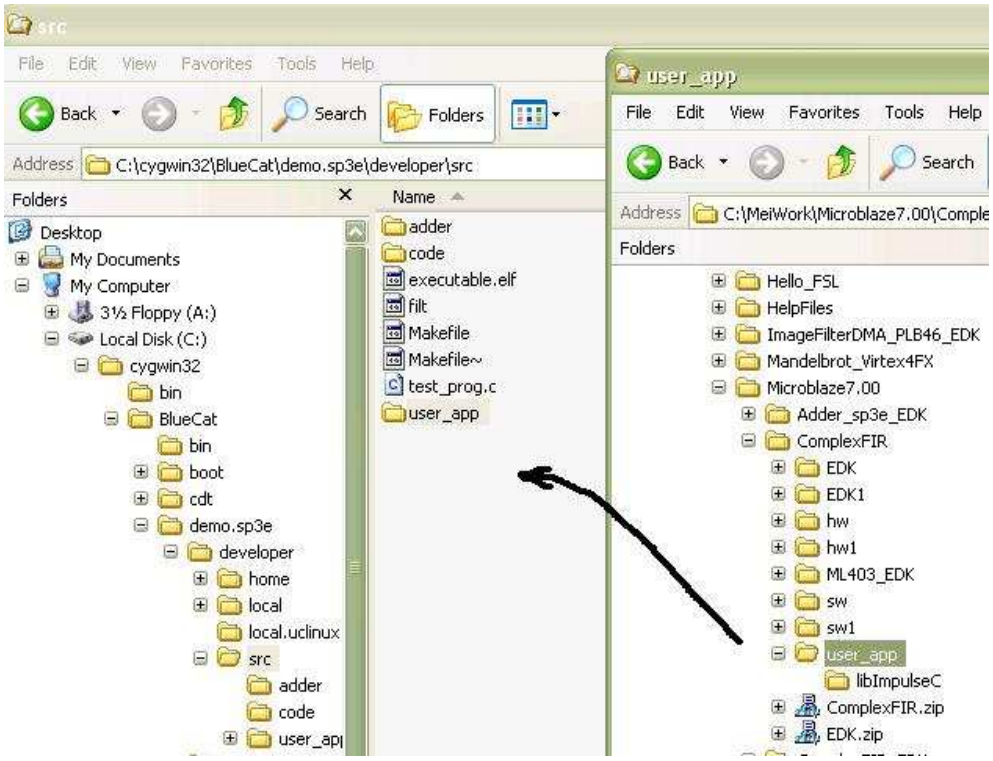
```
BlueCat:bash-3.00$ exit
exit
C:\cygwin32>cygnus
"TARGET_PATH is c:\cygwin32"
Warning: no CDROM found!
bash-3.00$ cd BlueCat/
bash-3.00$ ./SETUP.sh
bash-3.00$ ./SETUP.sh sp3e
bash-3.00$ _
```

Command “make menuconfig” can be used to modify the Linux kernel configuration. This is optional; we will just use the default configuration here.



Building the Complex FIR Software into the Linux File System

Copy the software code and libraries from the Impulse C project directory to the BlueCat Linux developer source directory as follows:



Next, modify the Makefile in $BlueCat/demo/developer directory. Add the following lines:

```
cd src/user_app/libImpulseC; make all
cd src/user_app; make all
```

Add the following line to the developer.SPEC:

```
cp ./src/user_app/user_app tmp
```

Next, build the kernel by running:

```
make clean
make all
```

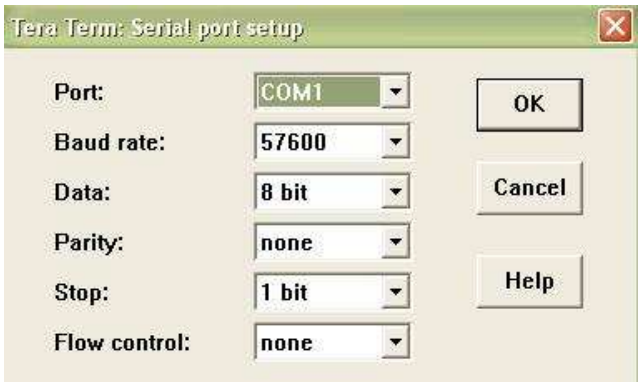
This will build the Linux kernel image with the example application. The current PSP supports up to 4 FSLs. Minor modifications to the source code can be done to accommodate more FSLs (up to 8). The following Linux commands are used to add FSLs as device:

```
mknod /dev/fsl0 c 10 230
mknod /dev/fsl1 c 10 231
```

The major number is 10, minor numbers can be 230 and up.

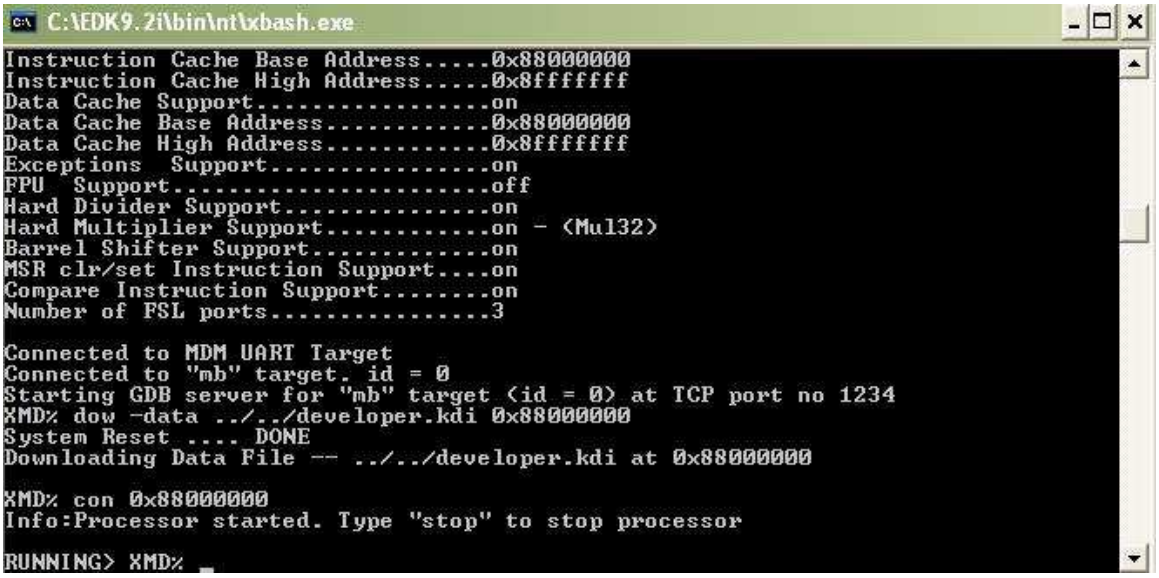
Download Linux Image to Target Board

First, open the Tera Term or other terminal application. Set the serial port to 57600, 8-N-1.



Next, open the XMD from EDK. Download the kernel image “developer.kdi” to the DDR SDRAM starting at address 0x88000000. This will take a few minutes to complete. Then start execution.

```
dow –data ../developer.kdi 0x88000000
con 0x88000000
```



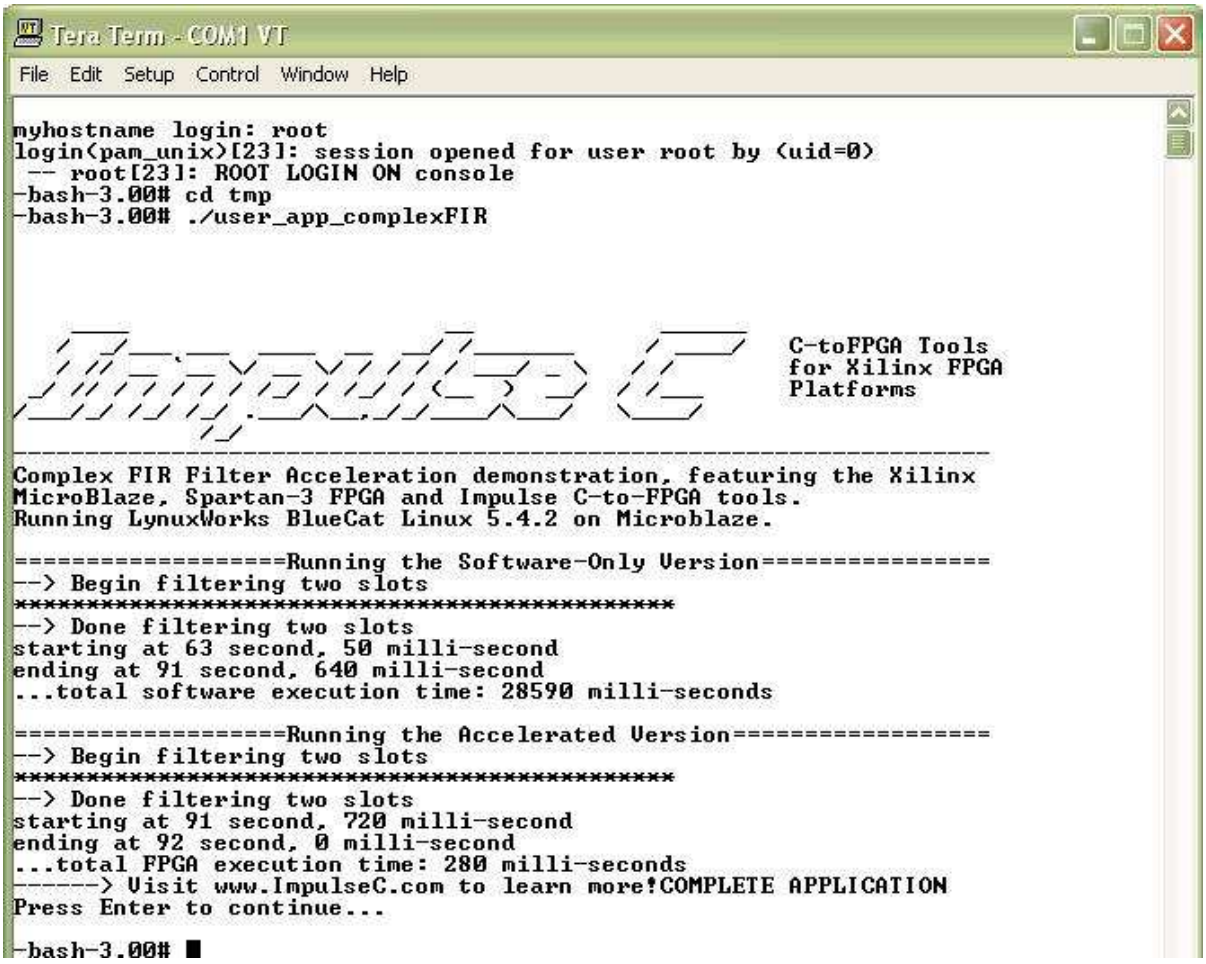
Watch the Tera Term window for linux booting messages.

Log in as “root”.

The Complex FIR application is in the tmp directory.

```
cd tmp
./user_app
```

The Complex FIR filter will execute as follows:



Conclusion

The execution time with hardware acceleration is 102 times faster than the software only version.

Compared to the execution on standalone OS on MicroBlaze, which runs only 163 ms, the BlueCat Linux does introduce latency in terms of execution time. But this latency is more than made up for by the hardware acceleration achieved using Impulse C.