

Port Embedded Linux to XUP Virtex-II Pro Development Board

Presented by
Aswin Ramakrishnan

Points of Interest

- Embedded Linux
- XUP Virtex-II Pro Development Board

Embedded Linux

- What?
 - Linux on Embedded Systems
 - Static - For limited resources
- Why?
 - Easily Customizable
 - Free
 - Rich code resource
- Who?
 - Motorola, Nokia, Panasonic, UT Starcom, Cisco, Philips, Linksys

Embedded Linux

- The Boot-loader
- The Linux Kernel
- The Root Filesystem

SoPC – System on Programmable Chip

- What?
 - Includes CPU + Mixed Signal Arrays of ICs
 - FPGAs
- Why?
 - Flexibility
 - Speed and Reliability
- How?
 - The Core
 - Configurable Analog and Digital blocks
 - Programmable Routing and Interconnect
 - BSP provides support for OS I/O

Xilinx XUP Virtex II Pro

- FPGA Development System
- PowerPC 205 Processors
- DDR SDRAM DIMM that can accept up to 2Gbytes of RAM
- 10/100 Ethernet port
- USB2 port (for FPGA configuration only)
- Compact Flash for FPGA configuration and data storage
- PS/2 and RS232 port

Xilinx XUP Virtex II Pro

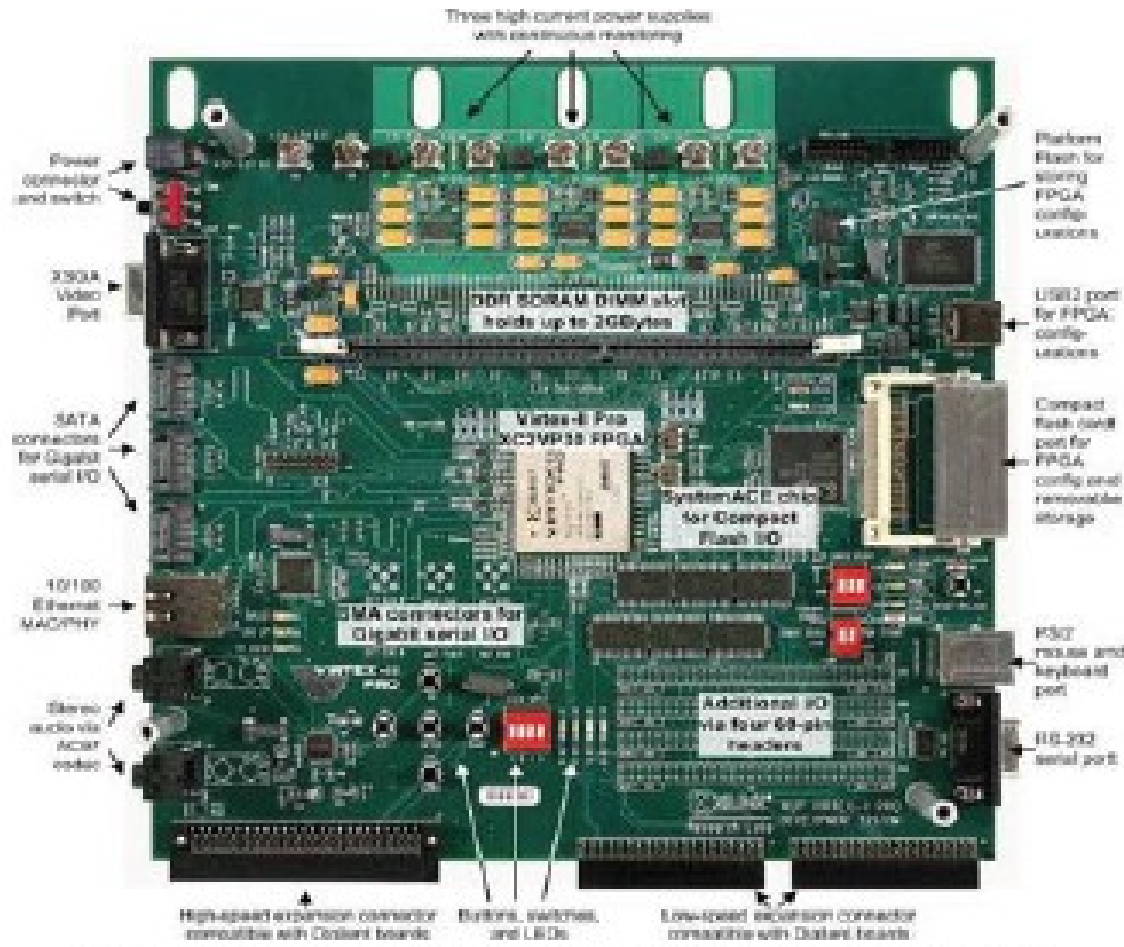


Figure 2. XUP Virtex-II Pro Development System

Porting Embedded Linux – Development Flow

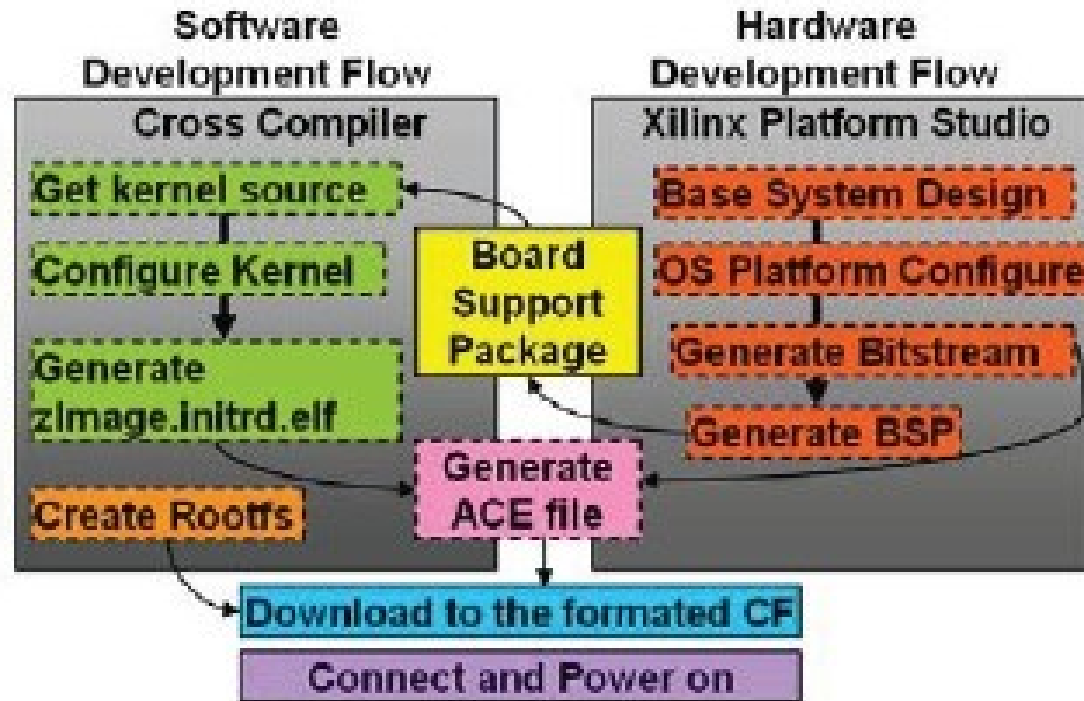


Figure 1. System Development Flow

Base System Building

Table 1. The major IP cores' parameter

IP Core Name	Parameter
PPC405	300MHz
OPB_UART16550	Use interrupt
OPB_SystemACE	Use interrupt
OPB_Ethenet_MAC	No DMA and use interrupt
PLB_DDR_256MB	256MB and use interrupt
PLB_BRAM_IF_CNTL R	128KB

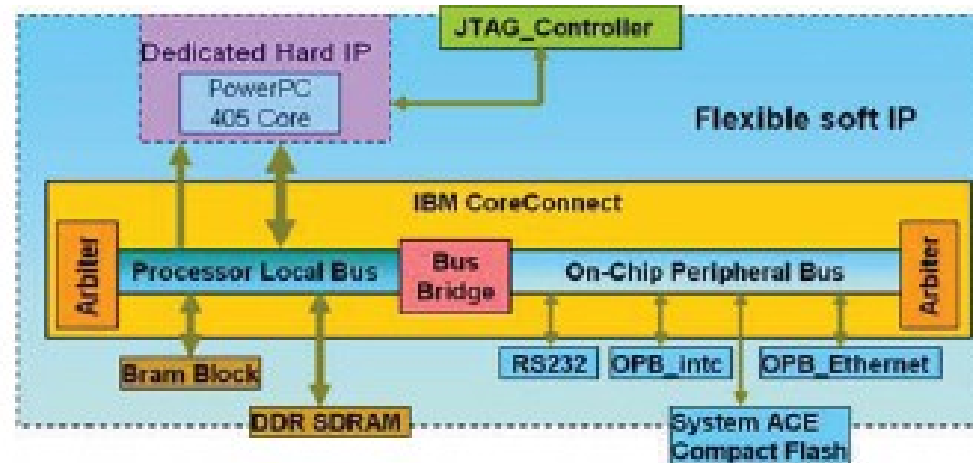
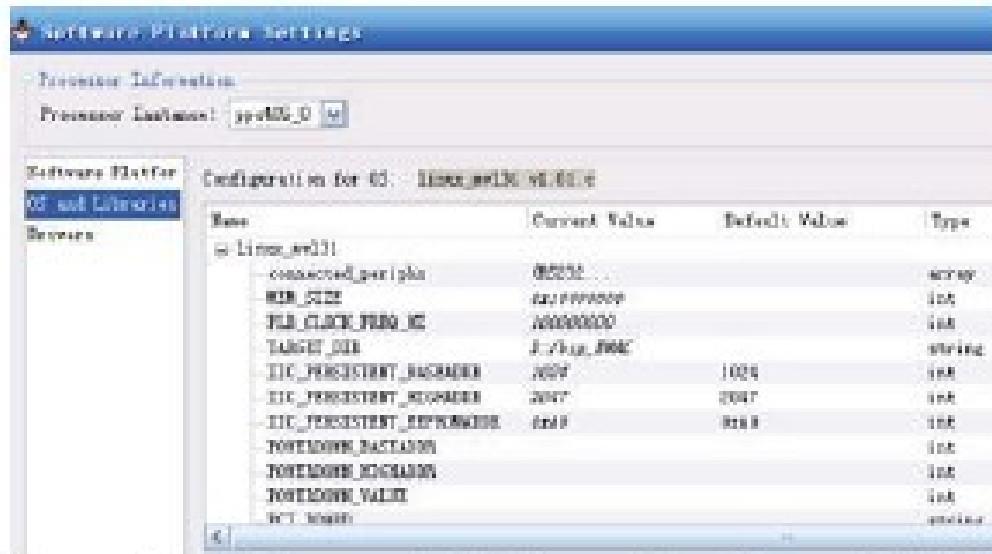


Figure 3. Base System Architecture

OS Platform Configure



The screenshot shows a window titled 'Software Platform Settings'. Under the 'Processor Information' section, the 'Processor Name' is 'ppc602_0'. The 'Software Platform' section is expanded to show 'Linux_armv7l'. Below this, a table lists various configuration parameters for this platform.

Name	Current Value	Default Value	Type
connected_peripherals	0x0000		string
MEM_SIZE	0x00000000		int
PLB_CLOCK_FREQ_MHz	100000000		int
TARGET_DIR	./target/armv7l		string
IIC_PERSISTENT_BROADCAST	0x01	0x00	int
IIC_PERSISTENT_ADDRESS	0x07	0x07	int
IIC_PERSISTENT_RESPONSE	0x01	0x00	int
JOYDOWN_DURATION			int
JOYDOWN_THRESHOLD			int
JOYDOWN_VALUE			int
WCT_PATH			string

Figure 4. Software Platform Settings Table

Generate BSP and Bitstream

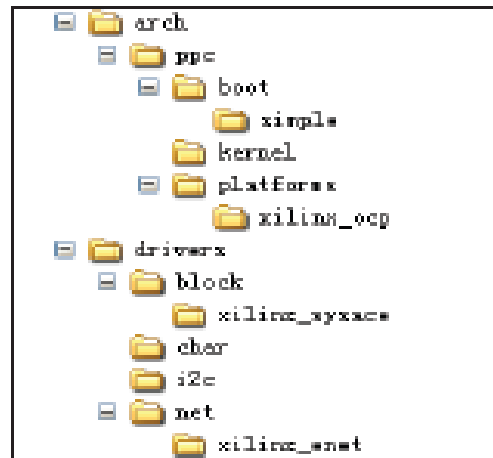


Figure 5. Generated BSP directory structure

Build cross-compiling environment for PPC

- `tar -zxvf crosstool-0.43.tar.gz`
- `cd crosstool-0.43`
- Edit `demo-powerpc-405.sh` file and amend similar content like that:
 - `TARBALLS_DIR=$HOME/crosstool-0.43`
 - `RESULT_TOP=/opt/crosstool`
 - `sh all.sh -notest`
 - `sudo mkdir /opt/crosstool`
 - `sudo chown $USER /opt/crosstool`
 - `sh demo-powerpc-405.sh`

Getting the kernel sources

- `wget http://www.bitmover.com/bk-client2.0.shar`
- `sh bk-client2.0.shar`
- `cd bk-client2.0`
- `sh demo.sh`
- `./bkf clone bk://ppc.bkbits.net/linuxppc_2_4_devel
linuxppc-2.4.26`
- `export PATH=/opt/crosstool/gcc-3.3.6-glibc-
2.3.2/powerpc-405-linux-gnu/bin:$PATH`

Configure and Compile kernel

- make menuconfig
- Processor Type:40x
- Machine Type:Xilinx-ML300
- TTYs0 device and default console:UART0
- Networking support: Enable
- Default bootloader kernel arguments: Enable
- Initial kernel command string: "console = ttyS0, 38400 ip=on root=/dev/xsysace/disc0/part3,rw"

Configure and Compile kernel

Block Devices

- Xilinx on-chip System ACE: Enable
- RAM disk support: Enable
- Default RAM disk size:16384

Character devices

- Standard/generic(8250/16550 and compatible UARTs) serial support: Enable
- File systems
- Virtual memory file system support: Enable
- Second extended fs support: Enable

Build Root FileSystem

- cd mkroofs
- Modify some parameters (listed as following) in mkrootfs.sh:
 - LFS= the prefix location of the rootfs
 - CC= cross compiler
 - TARGET_PREFIX=location of cross toolchain library file
 - BUILD_TOOLS= directory of cross toolchain
 - PPC_KERNEL= directory of kernel source
 - PPC_KERNEL_VERSION=2.4.26
- Replace “busybox-1.00-pre2” with desired busybox directory
- sh mkrootfs.sh

Building ACE File

Edit genace.opt file, the content as following:

```
-jprog  
-board user  
-target ppc_hw  
-hw download.bit  
-elf zImage.initrd.elf  
-debugdevice devicenr 1 cpunr 1  
-ace system.ace
```

Download & Test

- Format flash
- Set the COM Port
- Baud Rate: 38400
- Boot Linux

```
loaded at:      00400000 004DF1E4
board data at: 004DC13C 004DC154
relocated to:  0040564C 00405664
zimage at:     00405B53 004DBC0A
avail ram:     004E0000 10000000

Linux/PPC load: console=ttyS0,115200 ip=on root=/dev/xsysace/disc
0/part3 rw init=/bin/sh
Uncompressing Linux...done.
Now booting the kernel
```