



LINUX SOLUTIONS FOR A  
RECONFIGURABLE WORLD

## **PetaLogix Spartan3E-500 Embedded Linux Demo**

Release 1.00.a 4<sup>th</sup> September 2006

Author: John Williams

## 1 Introduction

Thank you for evaluating the PetaLogix Embedded Linux demonstration package for the Xilinx Spartan3E-500 board.

This document will take you through the steps necessary to download the demo onto your board.

There are two different versions of this demonstration package. The “**one-time**” demo will not change the contents of any flash memories or configuration PROMs on your board, and can be used to take a quick look.

The “**persistent**” demonstration programs the FPGA configuration PROM, and the Intel NOR Flash on the board, so that the board will automatically boot up into the Embedded Linux system.

### 1.1 *Supported platforms*

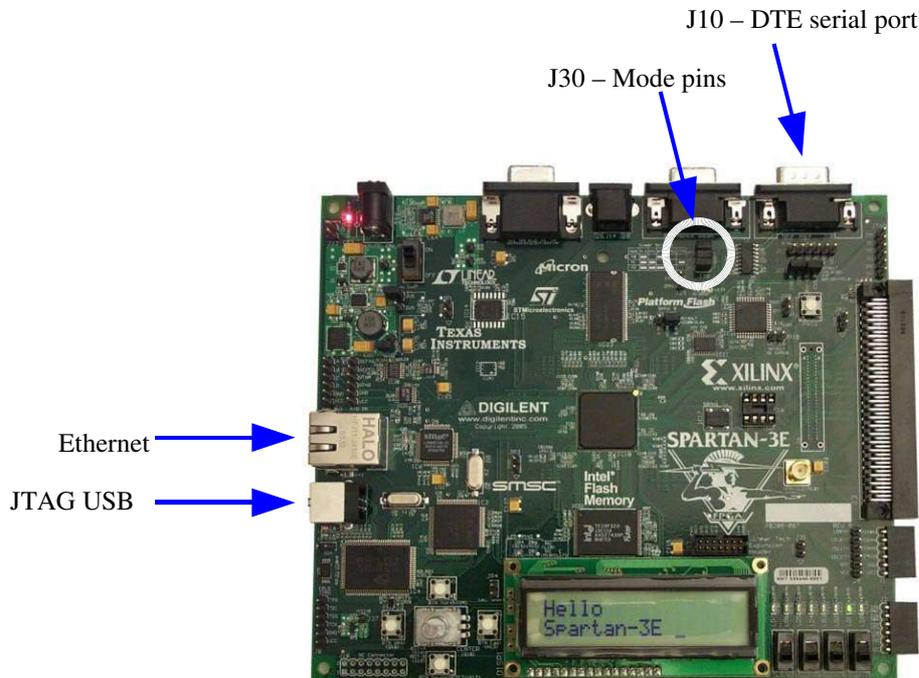
- Windows XP
- Linux

### 1.2 *Required Tool versions*

- ISE8.1 Service Pack 3
- EDK8.1 Service Pack 2

## 2 Preliminaries

Refer to the following figure when setting up the board.



### 2.1 Obtaining the demo package

If you haven't done so already, download the demo package from the PetaLogix website, via

[http://www.petalogix.com/resources/reference\\_designs](http://www.petalogix.com/resources/reference_designs)

Unzip the PetaLogix-Spartan3E-500-demo.zip archive, and change into the directory that is created.

### 2.2 USB cable (required)

Make sure your Xilinx tools are properly installed, including the Platform Cable USB drivers. Search [www.xilinx.com](http://www.xilinx.com) for “Platform Cable USB FAQ” for assistance.

Connect a standard USB cable between your PC, and the USB port on the board (J18)

### 2.3 Serial Port (required)

Connect a 9-pin Null Modem Serial cable between your PC and the “DTE” socket on the board (J10).

Note: The “DCE” socket (J9) is not used in this demonstration.

## PetaLogix Spartan3E-500 Demonstration Package

Start a serial console program (e.g. *hyper-terminal* on Windows, or *minicom* on Linux), and set it to the following

- 115200 baud
- software flow control
- 8 data bits, 1 stop bit
- no parity

### 2.4 Ethernet (recommended)

Connect a standard ethernet cable between the board (J19) and your PC, or a 10/100 ethernet hub/switch.

By default, the board will boot with an IP address of 192.168.0.50. You can easily change this later. Ask your system administrator to assign you a temporary IP address.

Note: due to reduced logic resources on the Spartan3E-500 FPGA, the Xilinx EthernetLite controller is used. This adapter provides basic ethernet connectivity, and is not intended for high bandwidth applications.

### 2.5 Board Jumper Settings

The jumper switches on the board should be in their factory default positions. Most importantly, the following should be observed:

- J30 (Mode) : All closed

## 3 One-time demo

Power on the board, making sure it is connected as described above.

Open up a shell window (Cygwin on XP, or xterm on Linux).

Change into the demo directory, and run the *boot-onetime* script:

```
$ ./boot-onetime.sh
Configuring the FPGA...
Downloading and booting the kernel, please be patient
Done..
Connect hyperterminal or minicom at 115200 baud.
Login using root/root as the username and password
```

You should see the kernel boot messages appear on your serial console, and be presented with

a login prompt. See the section “Using the Demo” below for what to do next.

You will need to repeat this step each time you power up the board. Alternatively, use the “persistent demo” as described below.

## 4 Persistent demo

The persistent demo writes an FPGA configuration into the Platform Flash devices on the board, and also a Linux kernel image into the Intel NOR flash. Do not use this demo if you wish to preserve the contents of these non-volatile memories.

Note that the default contents of these devices are available for download from the Xilinx website, if you do wish to restore them.

First, run the *prepare-persistent* shell script. This will configure the PROM and FPGA, and boot the board into a kernel image. You must press “ENTER” when prompted, to confirm you wish to overwrite the PROM contents. Press CTRL+C if you change your mind.

```
$ ./prepare-persistent.sh

This script will change the contents of the Platform Flash memory
on your Spartan3E-500 board. Press Ctrl+C to cancel now if you
do not wish this to happen, otherwise press ENTER.

Configuring the Platform Flash, please do not interrupt this step
Downloading and booting the initial kernel, please be patient
Done..
Connect hyperterminal or minicom at 115200 baud.
Login using root/root as the username and password
```

After this step completes, it still remains to program the kernel image into flash. See the relevant section in “Using the demo” for details.

## 5 Using the Demo

### 5.1 Logging in

Once the demo starts, login with username “root” and password “root”

### 5.2 Setting the IP address (ethernet required)

After booting, you can run the “setip” script to change the IP address of your MicroBlaze board, e.g.

```
# setip 192.168.0.10
```

For the one-time demo, you need to run this script each time you reboot the board. In the persistent mode, the change will be committed to flash, and the IP address will be set correctly next time you boot the board.

### 5.3 **Directional buttons and rotary encoder**

The Spartan3E board has four directional pushbuttons, and a rotary encoder, suitable for user-input type applications. PetaLogix developed a custom IP core and Linux device driver to integrate these controllers into the standard Linux input subsystem.

The pushbuttons are mapped to appear like directional arrow keys on a regular keyboard, while the rotary knob itself is mapped as a miscellaneous “dial” device.

You can test this out with the “*keypad-test*” demo program.

```
# keypad-test
```

PetaLogix are the experts at integrating custom hardware with the Linux kernel. We concentrate on delivering a complete Linux operating system platform, ready for you to do your application code development.

### 5.4 **Webserver (ethernet connection required)**

The demo contains a small footprint, but fully featured embedded webserver. Point your browser to the IP address (default is 192.168.0.50, or whatever address you used with the “*setip*” script above).

#### 5.4.1 **Upload new kernel image (persistent demo only, ethernet required)**

The default webpage in the demo includes a CGI-BIN web application, to upload a new kernel image and write it into flash.

Once you have connected to the embedded webserver, click the “Browse...” button, and browse to the file

*PetaLogix-Spartan3E-500-demo/images/image-persistent.bin*

Then, click the “Upload kernel image” button. The upload and flash writing will take a minute or two. On the console windows you will see the board reboot when it has completed.

This demonstrates one of the great features of Embedded Linux – remote upgradability. Not only Linux kernels, but with the right design even new FPGA bitstreams can be uploaded remotely via a web browser, using standard tools and applications.

Contact PetaLogix to talk about adding this functionality to your next project.

### 5.5 **GPIO demo**

From the MicroBlaze console, you can use the *gpio-test* program to drive the LEDs and query the switches on the board. Run *gpio-test* without any arguments to get a help screen. Some useful examples are:

```
# gpio-test -c                # Display a cyclic pattern on the LEDs
# gpio-test -o 0xFF           # Turn on all LEDs
```

## PetaLogix Spartan3E-500 Demonstration Package

```
# gpio-test -o 0x0          # Turn off all LEDs
# gpio-test -n 1 -i        # Read the switches
```

### 5.6 Telnet remote access (ethernet connection required)

You can use the standard telnet application to connect remotely to the board, e.g.:

```
$ telnet 192.168.0.50
```

Secure access such as openSSL and ssh is also possible – contact PetaLogix for details.

### 5.7 C++ demos

As of 3<sup>rd</sup> Quarter 2006, C++ applications for Embedded Linux on MicroBlaze are fully supported. Some simple C++ demo applications show the use of static constructors, exception handlers, and the iostreams libraries (see `/bin/cxx_*`)

### 5.8 Shell command line completion

The demo is configured with a fully-featured shell, permitting tab-based command line completion, command history and editing, and other features you've come to expect.

### 5.9 Games

Some classic text adventure games are included in the demo, “dungeon” and “advent4”:

```
# dungeon
Welcome to Dungeon.                This version created 11-MAR-91.
You are in an open field west of a big white house with a boarded
front door.
There is a small mailbox here.
```

It is also possible to connect remotely to these games (ethernet connection required). For `dungeon`, telnet in on TCP port 8798, or port 8898 for `advent4`:

```
$ telnet 192.168.0.50 8798
Trying 192.168.0.50...
Connected to 192.168.0.50 (192.168.0.50).
Escape character is '^]'.
Welcome to Dungeon.                This version created 11-MAR-91.
You are in an open field west of a big white house with a boarded
front door.
There is a small mailbox here.
>
```

## 6 Contact

Call PetaLogix, and see what a difference we can make in your next MicroBlaze project.

- North America ( +1) 408 916 5649
- Australia / Asia Pacific ( +61) 7 3102 3790
- Email [enquiries@petalogix.com](mailto:enquiries@petalogix.com)
- Skype: [petalogix.enquiries](https://www.skype.com/en/contacts/petalogix/enquiries)