

Application note

Using PEEDI as a stand-alone FLASH programmer

Version 1.0

1 Introduction

After the development process is over, the project has to meet the masses. The missing link here is the production, where PEEDI again finds its fitting place as a powerful FLASH programmer.

PEEDI may be configured in such way that non high-qualified engineering personnel can easy work with it. For example a worker in a PCB assembling factory programming the already assembled boards.

In this application note I will explain you how to set PEEDI to work alone (without connected to a file server or to a host PC).

2 Let's get to business

First you need to prepare the target configuration file and put it together with the image file to be programmed in a MMC or a SD card.

Apart all the common things that must be set in the target configuration file you have to write all the commands you need to be executed in scripts. For example you may need a script to erase the targets' FLASH, a script to program them, a script to verify them and a script to make this all at once. The scripts are declared in the [ACTIONS] section of the target configuration file like this:

```
[ACTIONS]
AUTORUN = 3
1 = erase_script
2 = program_script
3 = do_all_script
```

The numbers shows what digit is displayed on the front panel LED indicator when the given script is selected by the SELECT button. You can declare up to fifteen scripts starting from 1 to 15.

The AUTORUN parameter specifies which script to be executed automatically a target is detected to be connected to PEEDI. This way you don't even have to press the START button on the front panel to start programming – just connect the next board to be programmed. If you don't need the autorun feature, just comment the AUTORUN parameter using the semicolon sign in the beginning of the line.

You can name the scripts how you wish as long as their names are not identical to another section in the configuration file. For this example I chose names that explain what the scripts do.

A script is a list of PEEDI CLI commands grouped in a common section. Now you can define the scripts for example like I did:

```
[erase_script] ; this will erase all targets' FLASH chips
flash multi erase #all

[program_script] ; this will program all targets' FLASH chips
flash multi program #all card://myimage.bin bin 0x1000000

[do_all_script] ; this will erase and program all targets' FLASH
chips
flash multi erase #all
flash multi program #all card://myimage.bin bin 0x1000000
```

Using the **flash multi** command you can operate on up to four targets simultaneously saving huge amounts of time.

After the configuration file is ready you have to copy it and the image to be programmed to a FAT formatted MMC or SD card. This can be done in two ways – using MMC/SD card reader to your PC or using the PEEDI **transfer** command if you don't have a card reader. To do so, you have to start a file server (FTP, HTTP or a TFTP), connect to PEEDI command line using any telnet application and use the **transfer** command like this:

```
PEEDI - Powerful Embedded Ethernet Debug Interface
Copyright (c) 2005-2006 www.ronetix.at - All rights reserved
Hw:1.2, L:JTAG v1.1 Fw:1.12.2, SN: PD-0000-0123-4567
-----
```

```
peedi> transfer tftp://192.168.1.1/myconfig.cfg card://myconfig.cfg  
transferred 32768 bytes  
file transferred  
peedi> transfer tftp://192.168.1.1/image.bin card://myimage.bin  
transferred 9325 bytes  
file transferred
```

The bolded font shows the things you need to enter assuming that 'myconfig.cfg' is the target configuration file that will be used, 'myimage.bin' is the image file to be programmed and they can be downloaded from a TFTP server with IP address 192.168.1.1.

Never mind which way you copied the files to your card, make sure now it is plugged in the PEEDI's card slot.

Now are two things that left to be set – tell PEEDI to load the target configuration file from the MMC/SD card and use a static network IP address. The last is due to that if the PEEDI is set to get its IP from a DHCP server and there is no server (or the Ethernet cable is unplugged) PEEDI will continuously try to find a DHCP server over and over again, which will significantly prolong the PEEDI' boot time. So to set PEEDI you need to enter the RedBoot command line. To do so, first connect to PEEDI's serial port using a serial cable and any kind of terminal application (for example HyperTerminal). Then press and hold both PEEDI's front panel buttons and reset PEEDI using the back panel button. After a couple of seconds you should see the RedBoot command prompt in the terminal application:

```
Starting Redboot ...  
Platform: PEEDI (ARM7TDMI)  
Copyright (C) 2000, 2001, 2002, 2003, 2004 Red Hat, Inc.  
  
RAM: 0x00000000-0x00800000, [0x0000d228-0x00800000] available  
  
Ethernet eth0: MAC address 02:00:0b:69:2d:92  
IP: 192.168.1.100/255.255.255.0  
Gateway: 192.168.1.1  
TFTP/FTP/HTTP server for RedBoot: 192.168.1.1, DNS server IP: 0.0.0.0  
  
Command line hints:  
  CTRL-P previous line,  CTRL-N next line  
  CTRL-B move backwards, CTRL-F move forwards  
  CTRL-E move to end,   CTRL-A move to beginning  
  CTRL-D erase char,    CTRL-C abort  
  
Command hints:  
  fconfig - change RedBoot and PEEDI configuration parameters  
  update  - update PEEDI firmware  
  
RedBoot>
```

Now use the **fconfig** command to set a static IP and set PEEDI to load the target configuration file from the MMC/SD card like this:

```
RedBoot> fconfig  
Use DHCP for network configuration: no  
Gateway IP address: 192.168.1.1  
Local IP address: 192.168.1.100  
Local IP address mask: 255.255.255.0  
Default TFTP/FTP/HTTP server IP address used by PEEDI: 192.168.1.1  
  
Target config file:  
Examples:  
  http://server/sub_directory/filename.cfg  
  tftp://server/sub_directory/filename.cfg
```

```
card://sub_directory/filename.cfg
ftp://user:password@server/sub_directory/filename.cfg

filename.cfg - load config file from internal EEPROM
Path: card://myconfig.cfg
DNS server IP address:
Redboot telnet port: 23

Firmware update filepath:
<1> - http://www.ronetix.at/download/firmware/fw_peedi_revA_last.bin
<2> - tftp://192.168.3.1/fw_peedi_revA_last.bin
<custom path>
Path: 1
Update RedBoot non-volatile configuration (y/[n])? y
... Erase from 0x019f0000-0x01a00000: .
... Program from 0x007f0000-0x00800000 at 0x019f0000: .

You will have to restart for changes to take effect.

RedBoot>
```

Now everything is set up and PEEDI is ready to work all alone. So, disconnect the Ethernet cable and connect a powered target(s) to PEEDI's JTAG port. If you use the autorun feature the erasing and programming should start immediately after PEEDI have connected to the target(s). If you don't use it, just select desired script to execute using the SELECT front panel button and start it by pressing the START button.

Once a script is started a rotating segment of the front panel LED indicator show that an operation is in progress. If an error occurs during the script execution the script execution is interrupted. No other commands from the script are executed and a blinking code which describes the error is displayed on the LED indicator. Simultaneously the PEEDI speaker beeps as many times as the code error is shown. These are the possible error codes:

- 1 TIMEOUT
- 2 NOT FOUND
- 3 INVALID ARGUMENT
- 4 GENERIC ERROR

If the script execution is passed with no errors, the LED indicator displays again the script number and the speaker beeps once shortly.

Additional information about everything taking place in PEEDI may be obtained on the PEEDI's serial line – the PEEDI boot process, the config file loading, the connection to target, the script selection, the script execution and many more. So I recommend you to observe the information sent by PEEDI through the serial port and check if everything is going as expected until you are sure there are no problems with configuring PEEDI.

Let's get to real business

Now PEEDI is ready to program hundreds or thousands of boards, which in their turn will be ready for hundreds or thousands customers. So hurry up and give PEEDI to your board assembling company or to your production department workers to program your boards.