 <b>DAVE</b> Simply Embedded  DAVE s.r.l.  <a href="http://www.dave.eu">www.dave.eu</a>		
		VERSION:	<b>1.0.0</b>
		DOCUMENT CODE:	<b>DNNDK001</b>
		NO. OF PAGES:	<b>9</b>

# Neptune



**DN-NDK-001**

*OpenFirmware use cases*



**Printed in Italy**

## **Trademarks**

**Ethernet® is a registered trademark of XEROX Corporation**

**All other trademarks are the property of their respective owners**

## **Copyright**

**All rights reserved. Specifications may change any time without notification.**

## **Company Address**

**DAVE S.r.L.**

**Via Forniz 2**

**33080 Porcia (PN) – Italy**

**Phone: +39 0434 921215**

**e-mail: [info@dave.eu](mailto:info@dave.eu)**

**URL: [www.dave.eu](http://www.dave.eu)**

## **Technical Support (available to registered kit owners only)**

**e-mail: [support-neptune@dave.eu](mailto:support-neptune@dave.eu)**



## Contents

1 - Introduction.....	4
2 - OpenFirmware update operations.....	5
2.1 Updating OFW image from USB.....	5
2.2 Updating OFW image from network.....	5
2.3 Running OFW image from external LPC flash (dual boot).....	6
3 - Add and modify a boot splash image.....	8
4 - Bibliography and useful links.....	9

<i>History</i>		
1.0.0	January 2009	First release



# 1 - Introduction

---

OpenFirmware (OFW) is a hardware-independent firmware/BIOS, developed by Mitch Bradley at Sun Microsystems, and used in post-NuBus PowerPC-based Apple Macintosh computers (though it has been dropped with Apple's transition to Intel processors), Sun Microsystems SPARC based workstations and servers, IBM POWER systems, and PegasosPPC systems, among others. On those computers, Open Firmware fulfills the same tasks as BIOS does on PC computers.

OpenFirmware (OFW) is an implementation of the standard described by IEEE 1275-1994. It is accessed by a user by a Forth-based shell interface. In an embedded system, it is more powerful and flexible than a traditional BIOS, so we've chosen OFW for our Neptune CPU module. For more information about OpenFirmware's features, please refer to [1] and visit <http://www.dave.eu/neptune.html>

This application note describes some useful features of OpenFirmware, trying to be not only a guide for Neptune users, but also a showcase of OpenFirmware's capabilities.

## 2 - OpenFirmware update operations

**WARNING!** In case of power failure in the middle of updating process, OFW could be corrupted and not be able to boot the board any more. If this happens, to restore OpenFirmware it will be necessary the use of a JTAG hardware tool or exploit the dual boot strategy described in section 2.3.

### 2.1 Updating OFW image from USB

The typical updating procedure consists of replacing OFW binary image stored on LPC flash by using OFW itself. To do that you can just type the following command:

```
ok flash <filename>
```

where <filename> is the name of the new image, complete with device descriptor.

Using a USB Memory Stick (either formatted with FAT or ext2), simply copy the OFW image (neptune.rom) on the pen drive and issue the following command:

```
ok flash u:\neptune.rom
```

### 2.2 Updating OFW image from network

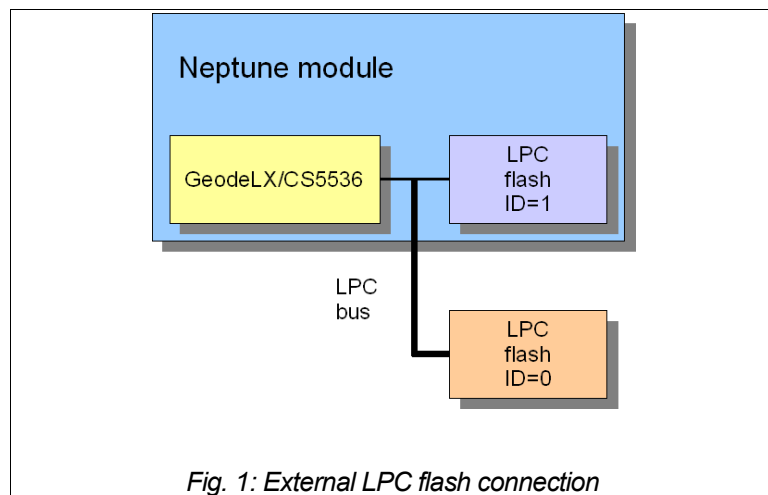
One of the most impressive features of OpenFirmware is its support for a complete TCP/IP stack. If Neptune is connected to a network, OFW can be updated retrieving the image via the TFTP protocol. Assuming that the ip address of the tftp server is

192.168.0.1 and the OFW image is stored in the neptune subdirectory on the server, the command to use is the following:

```
ok flash-net net:192.160.0.1,neptune\neptune.rom
```

## 2.3 Running OFW image from external LPC flash (dual boot)

Since LPC flashes provides a pin-based identification mechanism, more the one flash can be connected to LPC bus at the same time. Geode LX processor will boot from the device with ID set to 0 (see also Fig. 1).



Neptune mounts a LPC flash on-board, but it is highly recommended to include in the design of the host board the dual boot feature adding an external LPC flash which can be configured via dip-switches or jumpers.

In case the host board where Neptune is plugged supports an external flash, simply switching the device ID the system can boot using the OFW image stored into the external LPC, keeping the LPC on Neptune as backup.



The dual boot system can also be used to program OFW image to and from the Neptune LPC flash. Assuming that external flash has been programmed with a valid OFW image and the system is booting from the external flash (ID=0), the image on the Neptune module (ID=1) can be updated using the following command:

```
clone-firmware
```

In this case, this command will copy the portion of OFW from external flash to Neptune flash.

The entire procedure can be completely reversed: booting the system with the LPC flash of the Neptune module (ID=0), the external flash (ID=1) can be programmed using the same `clone-firmware` command.

## 3 - Add and modify a boot splash image

---

Thanks to the efficient architecture and the powerful command line, adding and modifying a boot splash image with OpenFirmware is very simple compared to the complex procedures needed by a classic BIOS. The splash screen image must be converted in a particular 565 16-bit bitmap, but OFW provides a built-in function to do the conversion starting from a more standard 24-bit bitmap.

Assuming that the original 24-bit image named splash.bmp is stored in an usb pen drive, the entire procedure needs just three commands:

```
ok bmp24rgb565 u:\splash.bmp u:\splash.ofw
ok setenv splash-image u:\splash.ofw
ok setenv output-device com1
```

The first command, `bmp24rgb565`, converts the original image and saves it in the usb pen drive; the second command sets an environment variable which instructs OFW on where the image is stored; the third command disables the OFW text output to the screen to show only the image.



## 4 - Bibliography and useful links

### **Bibliography**

[1] Dave Srl, Neptune Software Manual, 2008