175 Cabot Street – Suite 210
Lowell, MA 01854 USA
www.cspi.com
NASDAQ: CSPI

How do I determine which version of firmware is running on my Myri-10G Network Adapter and do I need to upgrade it?

Model:

Myri-10g Network Adapters

Software:

DBL, Sniffer, and Driver.

Operating System:

Supported Operating Systems based on the release. Please refer to the Release Notes and User Guides for the specific release.

Information:

With Myri-10G Network Adapters, the EEPROM firmware is actively running just after reset in order for the device to be properly recognized (the firmware has a role in initializing the pcie-link, and then managing some pci-config-space initialization from BIOS/OSes). So even when using Myri-10GE (or DBL, Sniffer10G, or MVA drivers), it is possible for the EEPROM firmware to have an influence on whether the network driver will initialize properly (the driver checks that the state of the card is good before replacing the active firmware on the card by the Myri10GE/DBL/Sniffer10G/MVA firmware included in the driver, and inherits some state from the EEPROM firmware).

We recommend a firmware upgrade only when some problem related to the early phases of BIOS/OS/device initialization has been seen. After the driver has finished initializing, what was in the EEPROM no longer has any influence (that's only 99.99% true, since theoretically there is some NIC state managed by the eeprom firmware which is inherited by the driver, but in practice that little inherited state has always been irrelevant to driver operations).

As a general rule, we recommend that all adapters in the configuration/cluster have the same version of the EEPROM firmware.

How do I determine which version of the firmware is running on the Myri-10G Network Adapter?

At any given point, there is an active firmware running on the embedded NIC-SRAM. Out of reset, the first **active firmware** is taken from the EEPROM and put into NIC-SRAM and controls the adapters for initialization purposes (pcie-negotiation, OS/BIOS configuration). Later when the drive (e.g. Myri10GE or DBL) is loaded, if some **runtime firmware** is available in the /**lib/firmware** directory, the driver will switch the **active firmware** in the NIC-SRAM to that version. That's why there are two different firmware versions into play (one from EEPROM, and one from **/lib/firmware**). Or, in the **MYRI10GE_BUILTIN_FW=1** case, from the **myri10ge*/firmware/*eth*h firmware** header files used by the driver at compile time.

To determine the version of the firmware stored in the EEPROM (not always the same as the **active/running firmware**), you need to run a told called **myri_info** available from the Myri-10G Network Adapter Tool Kit that we provide for the following OSes: Linux, FreeBSD, and Windows.

On Linux, untar the file, and run the command:

```
# myri-tools-<firmware-version>-linux/myri_info -v
```

On FreeBSD, untar the file, and run the command:

```
# myri-pcie-tool-<firmware-version>-freebsd/myri_info –v
```

On Windows, the self-extracting exe file will create a top-level directory:

```
cd \Program Files\myritool [ or \Program Files (x86)\myritool on 64bit
systems]myri_info –v
```

Example of **myri_info** output:

```
# ./bin/myri_info

pci-dev at 46:00.0 vendor:product(rev)=14c1:0008(00)
        behind bridge root-port: 40:0e.0 10de:005d (x8.1/x8.1)
Myri-10G-PCIE-8A -- Link x8
    EEPROM String-spec:
        MAC=00:60:dd:47:76:a9
        SN=315632
        PWR=100
        PC=10G-PCIE-8AL-R
        PN=09-03365
        TAG=ze_tools-1_4_5_rc1

    EEPROM MCP, PRESENT, length = 169528, crc=0x8d1d3a8b
        ETH ::1.4.5 -P- 2006/08/22 16:02:21 myri10ge firmware
        Bundle: exec_len=136752, PCI-ROM-len = 32768
    Running MCP:
ETH ::1.4.45 -- 2009/08/22 18:57:06 myri10ge firmware
```

Myri_info tells you which firmware is stored in the EEPROM and also tells you which firmware is currently active on the NIC (network adapter). Specifically:

The EEPROM firmware is the one after the EEPROM MCP line

The currently active firmware is the one after the Running MCP line.

At boot, before the driver is loaded, **active** and **eeprom firmware** are the same. After the driver has replaced the initial firmware with its own, the **Running MCP/firmware** and **EEPROM firmware** will generally be different.

TAG= indicates the jtag-bytecode part of the EEPROM, and what was last programmed in the EEPROM.

For any given **TAG,** the jtag-bytecode and the **EEPROM MCP** constitute the "primary NIC-bootloader (and fallback firmware)".

Lastly **Running MCP**, tells what is the currently active MCP (the MCP might come from either **EEPROM MCP** or the driver).

For instance on a machine where MX is loaded, the last lines of **Myri_info** look like:

```
# myri_info
..
  Running MCP:
MX10::1.2.0rc1cvs2006.05.18.04.00 xxx@xxxx:/nfs/home/xxxx/tiki-cvs Tue Jun  6
04:54:03 EDT 2006
```

You can also see the **Myri_info** whether an Etherboot/EFI/gPXE ROM part is included as part of **EEPROM MCP**, an etherboot-enabled eeprom will look like (the critical part is the presence of PCI-ROM-len):

```
..
    EEPROM MCP, PRESENT, length = 169528, crc=0x8d1d3a8b
        ETH ::1.4.5 -P- 2006/08/22 16:02:21 myri10ge firmware
        Bundle: exec_len=136752, PCI-ROM-len = 32768
    Running MCP:
```

```
ETH ::1.4.45 -- 2009/08/22 18:57:06 myri10ge firmware
```

Revision	Date	Change
1	6/23/2016	Initial Draft
2	7/20/2016	Feedback