10G PON Chipset

10G PON Development Kit

EASY PRX321 REF BOARD V1.9.1

Getting Started

MaxLinear Confidential

Revision 1.0, 2022-03-16
Reference ID 620690

Legal Notice

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by MaxLinear, Inc. MaxLinear, Inc. assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced into, stored in, or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of MaxLinear, Inc.

MaxLinear, Inc. does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless MaxLinear, Inc. receives, in writing, assurances to its satisfaction that: (a) the risk of injury or damage has been minimized; (b) the user assumes all such risks; (c) potential liability of MaxLinear, Inc. is adequately protected under the circumstances.

MaxLinear, Inc. may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from MaxLinear, Inc., the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

MaxLinear, the MaxLinear logo, any MaxLinear trademarks (MxL, Full-Spectrum Capture, FSC, G.now, AirPHY, Puma, and AnyWAN), and the MaxLinear logo on the products sold are all property of MaxLinear, Inc. or one of MaxLinear's subsidiaries in the U.S.A. and other countries. All rights reserved.

*Other company trademarks and product names appearing herein are the property of their respective owners.

MaxLinear, Inc.
5966 La Place Court, Suite 100
Carlsbad, CA 92008
Tel.: +1 (760) 692-0711
Fax: +1 (760) 444-8598
www.maxlinear.com

Current:	**Revision 1.0, 2022-03-16**
Previous:	**None**
Page	**Initial release**

Table of Contents

List of Figures

List of Tables

Preface

The 10G PON Development Kit EASY PRX321 REF BOARD V1.9.1 is a reference and demonstration platform for 10G PON Chipset PRX321 devices. This Getting Started document describes the basic steps for initial operation.

To simplify matters, the following synonyms are used:

PRX321

Synonym used for the 10G PON Chipset PRX321

EASY PRX321 REF BOARD

Synonym used for the 10G PON Development Kit EASY PRX321 REF BOARD

Organization of this Document

This document is organized as follows:

- **Chapter 1, Introduction to the EASY PRX321 REF BOARD**
  This chapter provides basic information about the EASY PRX321 REF BOARD.
- **Chapter 2, Bootloader Preparation**
  This chapter describes how to install, recover, or upgrade the U-Boot environment on the EASY PRX321 REF BOARD.
- **Chapter 3, Installing the 10G PON Chipset Software Image Using the Ethernet Interface**
  This chapter provides instructions on how to install the 10G PON Chipset software image.
- **Chapter 4, Updating the Marvell* AQR113 Firmware**
  This chapter provides instructions on how to modify the 10G PHY to support different IEEE Ethernet standards.
- **Chapter 5, Basic PON Operation**
  This chapter shows how use the utilities built into the 10G PON Chipset software in a GPON environment.
- **Chapter 6, Optical Transceiver Customization**
  This chapter provides instructions on how to add support for additional optical transceivers.
- **Literature References**

Document Conventions

These text formatting conventions are used throughout this document.

- *Italicized font* indicates important notes.
- `Monospace font` is used for console output, file names, or directories.
- **`Bold monospace font`** is used for commands the user must enter or to highlight important information in console output.

1 Introduction to the EASY PRX321 REF BOARD

This chapter lists the contents of the package and the features of the EASY PRX321 REF BOARD.

1.1 Contents of the Package

The package contains these items:

- EASY PRX321 REF BOARD V1.9.1
- A 110 - 240 VAC / 12 VDC power supply
- A USB cable with Type-A to microUSB Type-B connectors
- An XGS-PON Optical Transceiver SFP+ module with blue-colored bi-directional SC/UPC connector

1.2 Board Overview

Figure 1 shows the EASY PRX321 REF BOARD V1.9.1.

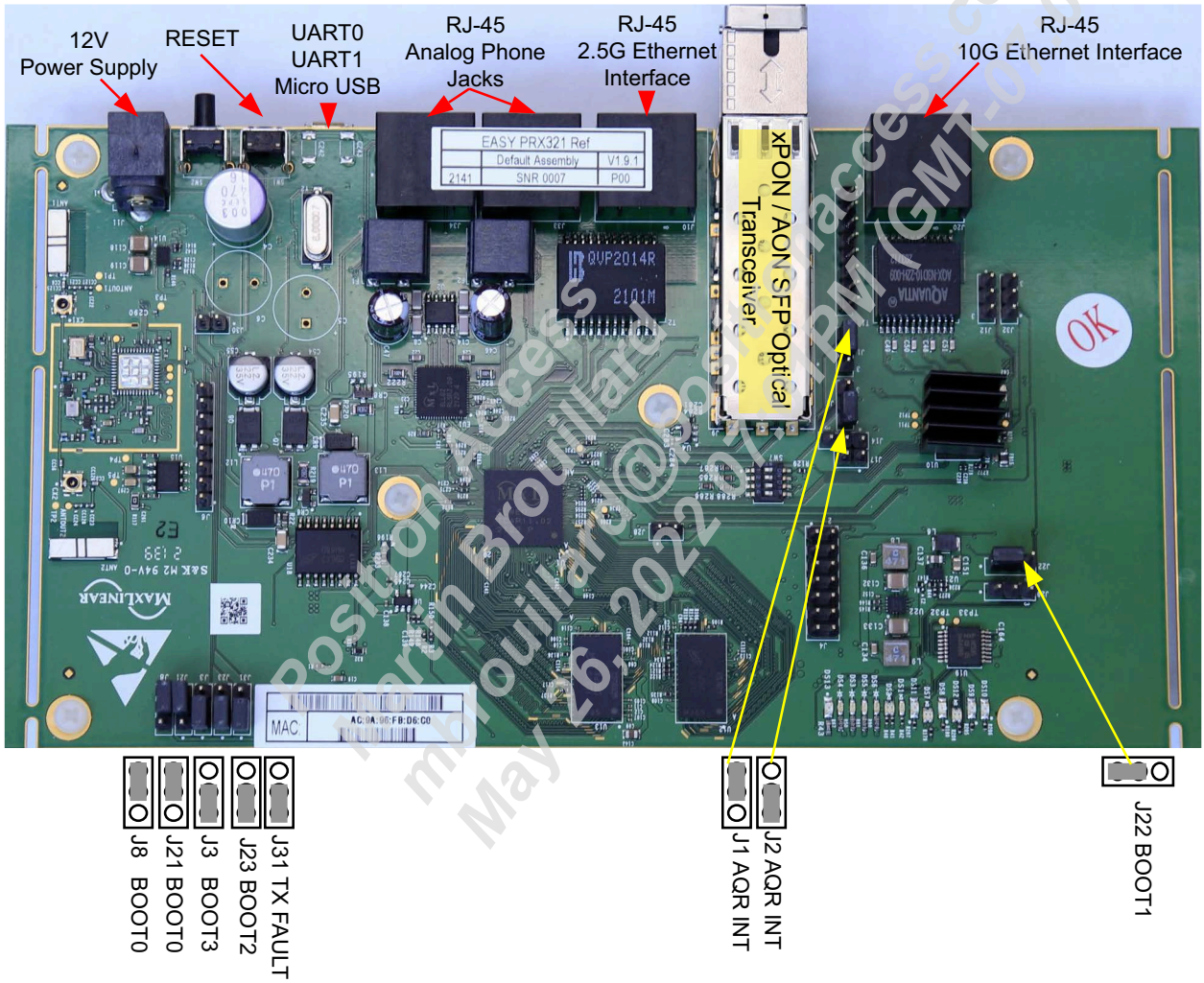

Figure 1 EASY PRX321 REF BOARD V1.9.1

1.3 EASY PRX321 REF BOARD V1.9.1 Relevant Features

These are notable features for the EASY PRX321 REF BOARD V1.9.1:

- The power solution is provided by Texas Instruments* and Richtec*.
- The two POTS interfaces allow the connection of analog phones.
- A DECT device is not mounted on the EASY PRX321 REF BOARD V1.9.1.
- An FTDI* FT2232D device provides two serial ports (UART0 and UART1) for debugging purposes. These ports are accessible via a microUSB Type-B connector.
- The 10G Ethernet interface supports 10GBASE-T or 1000BASE-T and auto-negotiates to the fastest available mode. This capability allows host PCs which do not support 10GBASE-T to connect to the EASY PRX321 REF BOARD.
  - The 10G Ethernet interface does not support 10BASE-T and 100BASE-T modes.
  - The Ethernet interfaces are dedicated for user traffic, updating the 10G PON Chipset software image, or for allowing access control, such as SSH or TFTP.
- The default boot mode for standard operation is Quad Bit NAND (QSPI), Bad Block Management, and ONFI mode enabled.
- MaxLinear provides the EASY PRX321 REF BOARD with a pre-installed 10G PON Chipset Linux* software image
  - Updating the software image requires an Ethernet connection between the EASY PRX321 REF BOARD and the host PC.

1.4 Initial Configuration

This chapter describes how to connect to and operate the EASY PRX321 REF BOARD.

Complete these steps to connect to the EASY PRX321 REF BOARD:

1. Connect the USB cable:
   a) Connect the Type-A connector to the host PC.
   b) Connect the microUSB Type-B connector to the EASY PRX321 REF BOARD.
2. Power on the EASY PRX321 REF BOARD:
   a) Connect AC/DC power supply to an AC power outlet.
   b) Connecting the power connector from the AC/DC power supply to the EASY PRX321 REF BOARD.
3. Wait for Microsoft* Windows* to detect the FTDI* FT2232D device and install the required drivers. These drivers are available at FTDI*'s website.
4. Open a terminal emulation software and verify Microsoft* Windows* installed the UART0 and UART1 interfaces as COM ports. **Figure 2** provides an example of how these interfaces appear when using the TeraTerm* terminal emulation software
5. Configure the terminal software with these parameters when connecting to the EASY PRX321 REF BOARD:
   115200 baud
   8-bit data
   No parity
   1 stop bit
   No flow control
6. Initiate the connection to the EASY PRX321 REF BOARD.
7. (Optional) Connect an Ethernet cable from the EASY PRX321 REF BOARD's 10G Ethernet port and the host PC when updating the 10G PON Chipset Linux* software image, see **Section 2.1.2**.

Note: Sometimes the EASY PRX321 REF BOARD's console does not appear after power cycling. When this occurs, restart the terminal software and reconnect to the COM port.

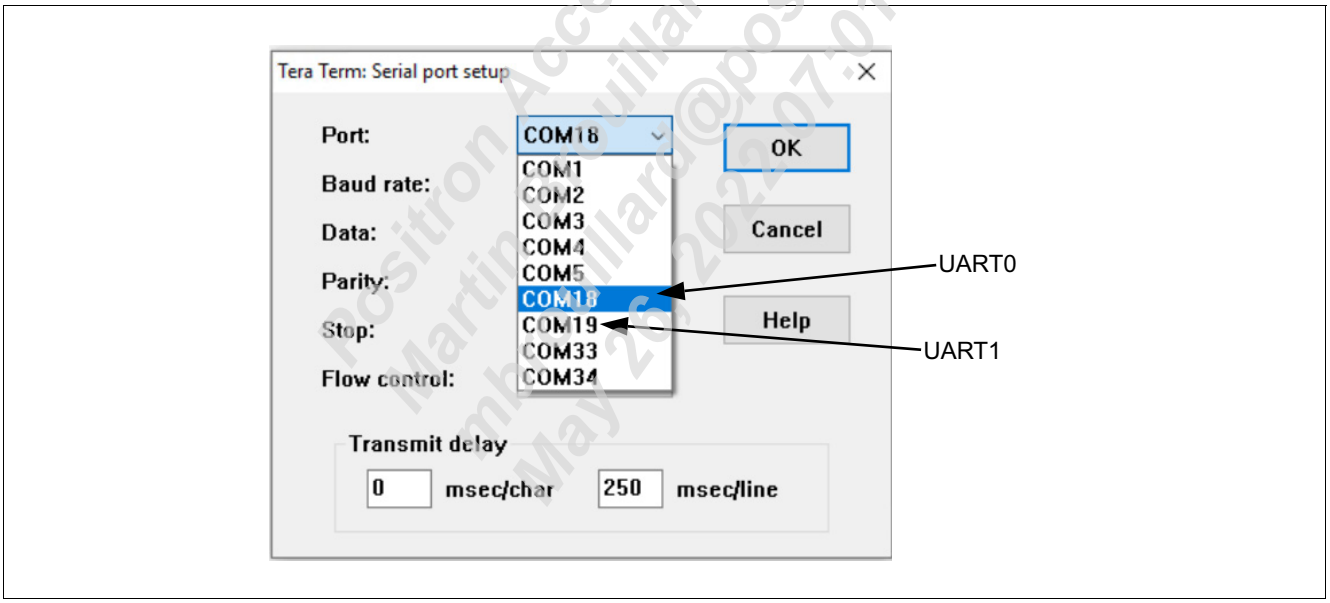

Figure 2 UART0 and UART1 Detected as COM Ports

Note: The assignment of COM port numbers may vary from PC to PC.

2 Bootloader Preparation

Updating U-Boot on the EASY PRX321 REF BOARD is required in two situations:

- To restore a corrupted or deleted flash memory.
- To update the 10G PON Chipset software.

The EASY PRX321 REF BOARD's hardware jumpers control the board's boot mode. See **[1]** for more information on supported boot modes.

2.1 Boot Mode Selection

Certain conditions require modifying the default boot mode.

See **Section 2.1.1** for instructions on repairing U-Boot using the serial interface.

See **Section 2.1.2** for instructions on updating U-Boot Using the Ethernet Interface.

2.1.1 Repairing U-Boot Using the Serial Interface

This section describes how to repair the U-Boot environment using the serial interface.

1. Configure the jumpers according to **Table 1** and **Figure 3** so that the EASY PRX321 REF BOARD boots from UART0 (`0x4`). Refer to **[1]** for more information on boot modes.
2. Connect to the EASY PRX321 REF BOARD from the host PC using terminal emulation software.
3. Power on the EASY PRX321 REF BOARD.
4. Transfer the `u-boot.asc` file. This file is located in this folder of the 10G PON Chipset SDK:
   `<sdk_folder>/sw/PRX300_DEBUG/uboot-prx321-sfu-qspi-nand/`
   Figure 4 shows an example of using Tera Term* to send this file.
5. This console output shows an example of an EASY PRX321 REF BOARD UART0 boot sequence:

```
ROM VER: 2.1.0
CFG 04
B
UART
*************************DdrOk
<<< LINES INTENTIONALLY REMOVED >>>
*************************
U-Boot 2016.07-MXL-v-3.1.241 (Feb 02 2022 - 07:00:52 +0000), Build: prx321-sfu-
qspi-nand
4Kec
      Watchdog enabled
DRAM:  128 MiB
NAND:  device found, Manufacturer ID: 0x2c, Chip ID: 0x24
256 MiB
Bad block table found at page 131008, version 0x01
Bad block table found at page 130944, version 0x01
*** Warning - bad CRC, using default environment
In:    serial
Out:   serial
Err:   serial
Reset cause: POR RESET
Net:   No ethernet found.
Watchdog timer stopped
run flash_flash to bring up the kernel
Hit any key to stop autoboot:  0
```

6. Interrupt the boot process when the `Hit any key to stop autoboot` message appears.
7. Prepare the host PC's terminal software to transfer the `u-boot-nand.bin` file from the `\uboot-prx321-sfu-qspi-nand\` folder[1)] using the Kermit protocol.
8. Enter this command from the EASY PRX321 REF BOARD serial console:

```
PRX300 # loadb 0xa0400000
## Ready for binary (kermit) download to 0xA0400000 at 115200 bps...
## Total Size      = 0x00034ec0 = 216768 Bytes
## Start Addr      = 0xA0400000
```

9. Transfer the file.
   For example, from Tera Term use the **File > Transfer > Kermit > Send**... menu option.
10. Wait until the file has successfully transfers to the EASY PRX321 REF BOARD. This file is temporarily stored in DDR memory.

1) The location of this file depends on the option selected during the build process.

11. Prepare to write the NAND flash memory by first erasing it using this command:

```
PRX300 # nand erase 0 40000
NAND erase: device 0 offset 0x0, size 0x40000
Erasing at 0x20000 -- 100% complete.
OK
```

12. Write the U-Boot image to the NAND flash memory using this command:

```
PRX300 # nand write a0400000 0 40000
NAND write: device 0 offset 0x0, size 0x40000
262144 bytes written: OK
```

13. Power off the EASY PRX321 REF BOARD.
14. Set the EASY PRX321 REF BOARD to boot from the QSPI NAND flash memory by configuring the hardware jumpers as indicated in **Figure 1**.
15. Power on the EASY PRX321 REF BOARD.
16. Verify that the EASY PRX321 REF BOARD boots from U-Boot.

Proceed to **Chapter 3** to install the 10G PON Chipset software image.

Table 1 EASY PRX321 REF BOARD UART Boot Jumper Configuration

Value	Name	RefDes
0	boot0	J21
0	boot0	J8
0	boot1	J22
1	boot2	J23
0	boot3	J3

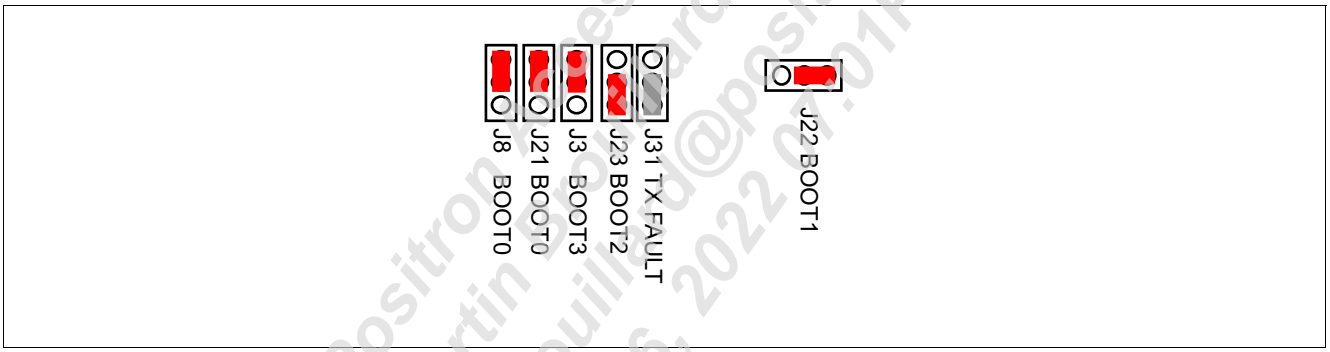

Figure 3 EASY PRX321 REF BOARD UART Boot Jumper Configuration

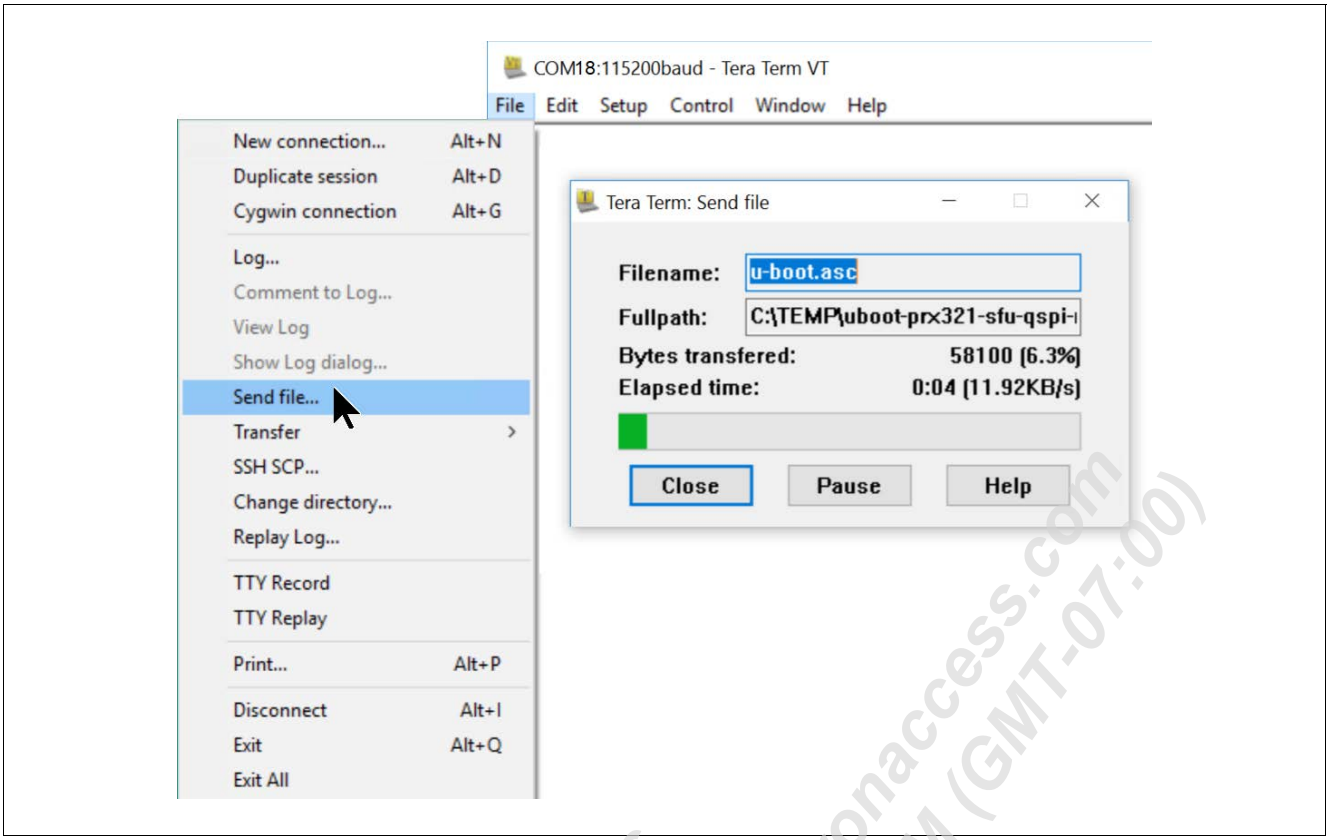

Figure 4 UART ASC File Transfer

2.1.2 Updating U-Boot Using the Ethernet Interface

This section describes how to update U-Boot using the 10G Ethernet interface.

Attention: The instructions provided in this section do not work when using UART boot.

1. Set the U-Boot environment's networking variables in accordance with the network and host PC configuration to allow downloading the U-Boot image via Ethernet port.

```
PRX300 # setenv ethaddr AC:9A:96:F0:A2:60
PRX300 # setenv ipaddr 192.168.1.1
PRX300 # setenv gatewayip 192.168.1.100
PRX300 # setenv netmask 255.255.255.0
PRX300 # setenv serverip 192.168.1.100
PRX300 # setenv tftppath
PRX300 # saveenv
```

   a) Ensure that the `ethaddr` MAC address is correct. The MAC address is printed on the EASY PRX321 REF BOARD label.
2. Connect an Ethernet cable from the EASY PRX321 REF BOARD's 10G Ethernet port (J20) to the host PC/LAN. The link from the 10G Ethernet to the host PC must support 10GBASE-T or 1000BASE-T. 100BASE-T and 10BASE-T are not supported.
3. Verify network connectivity between the EASY PRX321 REF BOARD and the host PC by issuing a ping command:

```
PRX300 # ping 192.168.1.100
Using prx300-eth device
host 192.168.1.100 is alive
```

4. Ensure the U-Boot variables `serverip` and `tftppath` are correctly configured before executing this command:
5.

```
PRX300 # run update_uboot
Using prx300-eth device
TFTP from server 192.168.1.100; our IP address is 192.168.1.1
Filename 'uboot-prx321-sfu-qspi-nand/u-boot-nand.bin'.
Load address: 0x82000000
Loading: ################
         41 KiB/s
done
Bytes transferred = 216768 (34ec0 hex)
NAND erase: device 0 offset 0x0, size 0x100000
Erasing at 0xe0000 -- 100% complete.
OK
NAND write: device 0 offset 0x0, size 0x34ec0
 216768 bytes written: OK
```

3 Installing the 10G PON Chipset Software Image Using the Ethernet Interface

When the U-Boot environment installation is complete and operational, install the 10G PON Chipset software, which consists of a Linux* kernel and file system, to the NAND flash memory. This chapter describes how to perform the installation.

1. Interrupt the boot process when the `Hit any key to stop autoboot` message appears.
2. Verify network connectivity from the EASY PRX321 REF BOARD and the host PC by issuing a ping command:

```
PRX300 # ping 192.168.1.100
Using prx300-eth device
host 192.168.1.100 is alive
```

3. Install the software image to the flash memory using the Ethernet interface with this command:

```
PRX300 # run update_fullimage
ubi0: attaching mtd1
ubi0: scanning is finished
ubi0: empty MTD device detected
ubi0: attached mtd1 (name "mtd=6", size 108 MiB)
ubi0: PEB size: 131072 bytes (128 KiB), LEB size: 126976 bytes
ubi0: min./max. I/O unit sizes: 2048/2048, sub-page size 2048
ubi0: VID header offset: 2048 (aligned 2048), data offset: 4096
ubi0: good PEBs: 864, bad PEBs: 0, corrupted PEBs: 0
ubi0: user volume: 0, internal volumes: 1, max. volumes count: 128
ubi0: max/mean erase counter: 1/0, WL threshold: 4096, image sequence number: 0
ubi0: available PEBs: 820, total reserved PEBs: 44, PEBs reserved for bad PEB
handling: 40
Using prx300-eth device
TFTP from server 10.91.163.13; our IP address is 10.91.184.123
Filename 'lede-intel_mips-prx300-PRX321_SFU_QSPI_PON-squashfs-fullimage.img'.
Load address: 0x82000000
Loading: #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #######
         759.8 KiB/s
done
Bytes transferred = 11643560 (b1aaa8 hex)
Image contains header with name [PON_1.16.0.r2]
Volume kernelB not found!
Creating dynamic volume kernelB of size 2172944
Erasing NAND...
Erasing at 0x120000 -- 100% complete.
```

```
Writing to NAND... OK
Image contains header with name [MIPS 4Kec Bootcore]
Volume bootcoreB not found!
Creating dynamic volume bootcoreB of size 5484864
Erasing redundant NAND...
Erasing at 0x160000 -- 100% complete.
Writing to redundant NAND... OK
Image contains header with name [UGW RootFS]
Volume rootfsB not found!
Creating dynamic volume rootfsB of size 3985616
Erasing NAND...
Erasing at 0x120000 -- 100% complete.
Writing to NAND... OK
Image contains header with name [PON_1.16.0.r2]
Volume kernelA not found!
Creating dynamic volume kernelA of size 2172944
Erasing redundant NAND...
Erasing at 0x160000 -- 100% complete.
Writing to redundant NAND... OK
Image contains header with name [MIPS 4Kec Bootcore]
Volume bootcoreA not found!
Creating dynamic volume bootcoreA of size 5484864
Erasing NAND...
Erasing at 0x120000 -- 100% complete.
Writing to NAND... OK
Image contains header with name [UGW RootFS]
Volume rootfsA not found!
Creating dynamic volume rootfsA of size 3985616
Erasing redundant NAND...
Erasing at 0x160000 -- 100% complete.
Writing to redundant NAND... OK
Volume rootfs_data not found!
PRX300 #
```

4. Once the NAND flash update is complete, power cycle the system or soft-reset it using this command:

```
PRX300 # reset
```

4 Updating the Marvell* AQR113 Firmware

This chapter describes how to temporarily update the firmware of the onboard Marvell* AQR113 device which provides the EASY PRX321 REF BOARD's 10 Gbps Ethernet capability. Updating this device's firmware is useful for testing and evaluating different firmware versions provided by Marvell*. This procedure is performed from the U-Boot environment and uses the hardware MDIO interface between the PRX321 and Marvell* AQR113 to transfer the firmware.

Complete these steps to temporarily replace the firmware:

1. Interrupt the boot process when the `Hit any key to stop autoboot` message appears.
2. Verify network connectivity from the EASY PRX321 REF BOARD and the host PC by issuing a ping command:
```
PRX300 # ping 192.168.1.100
Using prx300-eth device
host 192.168.1.100 is alive
```
3. Configure the GPIOs required to enable the MDIO interface:
```
PRX300 # gpio set-mux 10 1
PRX300 # gpio set-mux 11 1
```
4. Verify MDIO connectivity and read the firmware version stored on the Marvell* AQR113 using these commands:
```
PRX300 # mdio read mdio1 0 0x1E.0x20
Reading from bus mdio1
PHY at address 0:
30.32 - 0x506
PRX300 # mdio read mdio1 0 0x1E.0xC885
Reading from bus mdio1
PHY at address 0:
30.51333 - 0x17
PRX300 # mdio read mdio1 0 0x1.0xC41D
Reading from bus mdio1
PHY at address 0:
1.50205 - 0xb038
PRX300 # mdio read mdio1 0 0x1.0xC41E
Reading from bus mdio1
PHY at address 0:
1.50206 - 0x733
```
5. Note these register values and proceed to the next step. For details on possible values refer to [2].
6. Transfer the Marvell* AQR113 firmware from the host PC to the EASY PRX321 REF BOARD's DDR memory via TFTP using this command[1]:
```
PRX300 # tftp 0xa0400000 AQR-G4_v5.6.7-AQR_MaxLinear_FalconMtn.cld
Using prx300-eth device
TFTP from server 10.91.163.13; our IP address is 10.91.184.95; sending through
gateway 10.91.160.1
Filename 'AQR-G4_v5.6.7-AQR_MaxLinear_FalconMtn.cld'.
Load address: 0xa0400000
Loading: ############################
 66.4 KiB/s
done
Bytes transferred = 393218 (60002 hex)
```


1) Refer to Marvell*'s website and customer support to obtain firmware files.

7. Transfer the firmware file from the EASY PRX321 REF BOARD's DDR memory to the Marvell* AQR113 using this command:[1)]

```
PRX300 # aqc 0xa0400000 0x00060002
Firmware Address [0xa0400000] Size [0x60002]
Dump Information :
Current Revision : 0x506
PHY : AQR 113C
CRC check good on image file (0xA9D8)
Loading IRAM: Size
61675.....................................................................
.........................................
Image load good - mailbox CRC-16 matches (0xA207)
Loading DRAM: Size
31091.....................................................
Image load good - mailbox CRC-16 matches (0x100E)
Done
```

8. Verify that the Marvell* AQR113's firmware was successfully replaced by inspecting the reported version number using these commands and comparing to the values in step 4:

```
PRX300 # mdio read mdio1 0 0x1E.0x20
Reading from bus mdio1
PHY at address 0:
30.32 - 0x506
PRX300 # mdio read mdio1 0 0x1E.0xC885
Reading from bus mdio1
PHY at address 0:
30.51333 - 0x78
PRX300 # mdio read mdio1 0 0x1.0xC41D
Reading from bus mdio1
PHY at address 0:
1.50205 - 0xb196
PRX300 # mdio read mdio1 0 0x1.0xC41E
Reading from bus mdio1
PHY at address 0:
1.50206 - 0x79f
```

   This updated firmware version persists on the Marvell* AQR113 as long as the EASY PRX321 REF BOARD remains powered.

9. Continue to boot the 10G PON Chipset software image with this firmware using this command:

```
PRX300 # run bootcmd
```


1) It is possible that an incorrect PHY ID is reported when using an older version of U-Boot. Ignore this error as long as the version number is correctly identified in the subsequent steps.

5 Basic PON Operation

This chapter describes how to work with PON networking by using the EASY PRX321 REF BOARD as an ONU.

5.1 Prerequisites

Complete these steps to set up your environment:

1. Ensure the EASY PRX321 REF BOARD is powered off.
2. Insert the provided Superxon LTD* SFP+ XGS-PON optical transceiver module into the SFP+ port of the EASY PRX321 REF BOARD.
3. Connect one end of an optical cable with SC/UPC connector into the optical transceiver. **Figure 5** shows an example of an SC/UPC connector.
4. Connect the other end of the optical cable into the OLT.
5. Power on the EASY PRX321 REF BOARD and boot into the 10G PON Chipset software Linux* console.

Attention: Always use at least 10 dB of attenuation or more, if directly connected to an OLT. Maintain proper safety protocols when working with laser equipment to avoid personal injury or damage to equipment.

Figure 5 SC/UPC Connector

5.2 Monitoring the PLOAM State

Use one of the following commands to verify the PLOAM state:

- `root@prx321-sfu-qspi-pon:/# pon ploam_state_get`
  `errorcode=0 current=51 previous=40 time_prev=805`
- `root@prx321-sfu-qspi-pon:/# pon psg`
  `errorcode=0 current=51 previous=40 time_prev=805`

Use the `pond` application to continuously monitor the PLOAM state. Run the application in the background and observe the console output while disconnecting and reconnecting the optical cable. This console output shows an example of the behavior during this activity:

```
root@prx321-sfu-qspi-pon:/# pond&
alarm Loss of GEM channel delineation set
alarm Loss of downstream synchronization set
ploam state: previous - 51, current - 60
alarm Loss of signal set
```

```
alarm Loss of GEM channel delineation cleared
ploam state: previous - 60, current - 11
alarm Loss of signal cleared
alarm Loss of downstream synchronization cleared
ploam state: previous - 11, current - 12
ploam state: previous - 12, current - 23
alarm OMCI Integrity Key has changed triggered
ploam state: previous - 23, current - 40
ploam state: previous - 40, current - 51
```

5.3 Retrieving the ONU Serial Number

Use one of the following commands to print the ONU serial number:

- `root@prx321-sfu-qspi-pon:/# pon serial_number_get`
  `errorcode=0 serial_no="INTC96FBD6C0"`
- `root@prx321-sfu-qspi-pon:/# pon sng`
  `errorcode=0 serial_no="INTC96FBD6C0"`

5.4 Changing the Vendor ID of the ONU

The vendor ID is part of the ONU serial number. When changing the vendor ID is required, modify the line of the configuration file which contains the serial number.

```
root@prx321-sfu-qspi-pon:/# vi /etc/config/gpon....
option nSerial 'INTC96FBD6C0'
```

5.5 Viewing the PON Status with the pontop Application

`pontop` is a full screen text-based user interface application which collects and displays information about the PON connection. This information originates from 10G PON Chipset software daemons which are constantly updated in the background. After launching `pontop`, the following menu of options appears.

```
root@prx321-sfu-qspi-pon:/# pontop
Help
….
?     Help                              s     Status
c     Capability and Configuration      l     LAN Interface Status & Counters
w     Active alarms                     g-s   GEM/XGEM Port Status
g-c   GEM/XGEM Port Counters            g-d   GEM/XGEM port DS Counters
g-u   GEM/XGEM port US Counters         e-d   GEM/XGEM port Eth DS Cnts
e-u   GEM/XGEM port Eth US Cnts         f     FEC Status & Counters
t     GTC/XGTC Status & Counters        p-s   Power Save Status
p-c   PSM Configuration                 a-c   Allocation Counters
p-d   PLOAM Downstream Counters         p-u   PLOAM Upstream Counters
o-s   Optical Interface Status          o-i   Optical Interface Info
d-b   Debug Burst Profile
```

The first and third columns list the key, or key sequence, used to access each information screen. Each screen refreshes at a one-second interval.

Viewing the Optical Interface Information

To view the optical interface information, type the **o** key followed by the **i** key:

```
SFP+ information                                  Status
Vendor name                                     : SUPERXON LTD.
```

```
Vendor oui                                          :
Part number                                         : SOGX2699-PSGA
Revision                                            : 10
Serial number                                       : 97961212300217
Date code                                           : 210609
Wavelength                                          : 1270 nm

Options
------------
Power level declaration                             : Power Level 1
Paging implemented indicator                        : No
Retimer / CDR indicator                             : No
Cooled Transceiver declaration                      : No
Linear Receiver Output implemented                  : No
Receiver decision threshold implemented             : No
Tunable transmitter technology                      : No
RATE_SELECT functionality implemented               : No
TX_DISABLE implemented                              : Yes
TX_FAULT implemented                                : Yes
Inverted loss of signal implemented                 : Yes
Loss of signal implemented                          : No

DMI type
-------------
Digital monitoring implemented                      : Yes
Internally calibrated                               : Yes
Externally calibrated                               : No
Received power measurement type                     : Average
Address change required                             : No

Enhanced options
---------------------
Optional Alarm/Warnings flags implemented           : Yes
Soft TX_DISABLE control and monitoring implemented  : Yes
Soft TX_FAULT monitoring implemented                : Yes
Soft RX_LOS monitoring implemented                  : Yes
Soft RATE_SELECT ctrl and monitoring implemented    : No
Application select control implemented              : No
Soft RATE_SELECT control implemented                : No
Compliance                                          : SFF-8472 Rev 9.5


Page: Optical Interface Status
OPTION                                               VALUE
Optical transceiver temperature                     : 46 deg C / 319 K
Transceiver supply voltage                          : 3.30 V
…
```


Viewing the GEM/XGEM Port Counters

To view the GEM/XGEM port counters, type the **g** key followed by the **c** key:

```
GEM Index GEM ID u/s packets  u/s bytes d/s packets d/s bytes Key Errors
```

```
0           1   254          12192     250         12000     0
1           1022 0           0         0           0         0
```

Viewing the GEM/XGEM Port Status

To view the GEM/XGEM port status, type the **g** key followed by the **s** key:

```
GEM Index GEM ID Alloc Id Alloc Id st. Data/OMCI Max.Size Encr. k.r. Direction
0         1      1        Valid        OMCI      1980     None       DS + US
1         1022   1025     Valid        Ethernet  1628     None       DS + US
```

5.5.1 Viewing pontop Information from the Command Line

It also possible to print a single status message from `pontop` to the console without launching the entire application. This method is useful when checking the device status while performing other tasks.

```
pontop -b -g "<information screen>"
```

Replace the text between the double quotation marks with the desired option described in Section 5.5. For example, this command prints the optical interface status a single time to the console:

```
root@prx321-sfu-qspi-pon:/# pontop -b -g "Optical Interface Status"
```

The printed output is the same as the information displayed in the `o-i` example in Section 5.5.

6 Optical Transceiver Customization

This chapter describes how to add support for new optical transceivers. This chapter assumes that the selected new optical transceiver module supports the desired operation mode and is compatible with the equipment used in the intended deployment scenario. **Table 2** lists the optical transceivers the 10G PON Chipset software supports by default. These files are available in the `/etc/optic-db/` folder, also referred to as a database. The file name convention uses a `manufacturer-modelnumber` format.

Table 2 Optical Transceivers Supported by the EASY PRX321 REF BOARD by Default

Manufacturer	Model	Configuration File Name
Eoptolink*	EOLS-GU-25-D	`eoptolinkinc-eols-gu-25-d`
	EOLS-GU-25-DJ	`eoptolinkinc-eols-gu-25-dj`
Ligent Photonics Inc*	LTW2601-BC	`ligentphotonics-ltw2601cbc`
Lightron*	LWEKR-Rxx8-A	`lightroninc-0013c5-lwekrrxx8a`
MaxLinear	EASY PRX126 REF BOARD	`intel-m2180`
		`intel-m2181`
		`macom-m2180`
		`macom-m2181`
Source Photonics*	SPPS-27-XE-R3-CDFD	`sourcephotonics-spps27xer3cdfd`
Superxon*	SOGX2699-PSGA	`superxonltd-sogx2699-psga`
Wuhan Telecommunication Devices*	RTXM166-401	`wtd-rtxm166-401`
	RTXM166-501	`wtd-rtxm166-501`
	RTXM167-431	`wtd-rtxm167-431`

The EASY PRX321 REF BOARD includes with a Superxon* SOGX2699-PSGA transceiver. For reference, the contents of the `superxonltd-sogx2699-psga` is:

```
package 'optic'

config 'optic' 'common'
        option 'mode' 'xgspon' # only for identification
        option 'bias_hold_time' '70840'
        option 'bias_setup_time' '28980'
        option 'serdes_hold_time' '3288945'
        option 'serdes_setup_time' '0'
        option 'burst_idle_pattern' '0x0000FFFF'
        option 'burst_en_mode' '0'
        option 'tx_en_mode' '0'
        option 'tx_pup_mode' '0'
        option 'sd_polarity' '1'
```

6.1 Transceiver Identification

With every factory reset (firstboot command) or 10G PON Chipset software image update, the software attempts to automatically identify the installed optical transceiver by reading the type information over the I2C bus.

The 10G PON Chipset software attempts to assign the transceiver model by comparing against each of the files in the `/etc/optic-db/` the folder. When the software does not find the transceiver, it assigns the default values.

This example I^2C output indicates a Superxon* SOGX2699-PSGA device was identified:

```
root@prx321-sfu-qspi-pon:/# hexdump -Cv /sys/bus/i2c/devices/0-0050/eeprom
00000000 03 04 01 00 00 00 00 00 00 00 00 03 63 00 14 c8 |............c...|
00000010 00 00 00 00 53 55 50 45 52 58 4f 4e 20 4c 54 44 |....SUPERXON LTD|
00000020 2e 20 20 20 00 00 00 00 53 4f 47 58 32 36 39 39 |.   ....SOGX2699|
00000030 2d 50 53 47 41 20 20 20 31 30 20 20 04 f6 00 ce |-PSGA   10  ....|
00000040 00 1c 00 00 39 37 39 36 31 32 31 32 33 30 30 32 |....979612123002|
00000050 31 37 20 20 32 31 30 36 30 39 20 20 68 f0 02 fa |17  210609  h...|
00000060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
00000070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
00000080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
00000090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
000000a0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
000000b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
000000c0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
000000d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
000000e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
000000f0 32 59 39 32 31 33 36 30 30 31 37 31 36 20 20 20 |2Y92136001716   |
```

To initiate the transceiver identification process, run the `firstboot` command:

```
root@prx321-sfu-qspi-pon:/# firstboot
This will erase all settings and remove any installed packages. Are you sure? [N/y]
y
ubi0_6 is mounted as /overlay, only erasing files
root@prx321-sfu-qspi-pon:/# reboot
...
Committed image: A
ubi0: attaching mtd1
...
MIPS: set unaligned_action to 'SHOW'
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga-10-prx321-sfu-qspi-
pon' configuration
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga-prx321-sfu-qspi-
pon' configuration
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga-10' configuration
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga' configuration
[optic-db] Applied '/etc/optic-db/superxonltd-sogx2699-psga' configuration
...
```

6.2 Selecting New Transceiver Values

The 10G PON Chipset amends the `/etc/config/optic` file when a transceiver is identified. This file stores the device's basic parameter values, such as signal polarity and LOS/SD. The values in this file are copied from the database of known transceivers located in the `/etc/optic-db/` folder. It is possible to add support for new transceivers by modifying the `/etc/config/optic` file directly. However, this method is only valid for evaluation and testing purposes because these values are lost whenever the `firstboot` command is executed, or the board image is updated.

Table 3 shows the parameters and values in the `/etc/config/optic` file which are required when adding temporary support for new transceivers.

Table 3 Required Parameter and Values for New Transceivers

Parameter	Function and Possible Values
`option tx_en_mode '0'`	Enables or disables TX polarity. • **0: Autodetect** • 1: Active Low • 3: Active High
`option tx_pup_mode '0'`	Sets the power-up polarity (if used). • **0: Autodetect** • 1: Active Low • 3: Active High
`option burst_idle_pattern '0x0000FFFF'`	Sets the idle burst pattern. It is possible to adapt any idle burst pattern as long as the device and mode support it.
`option sd_polarity '1'`	Sets LOS or SD polarity.
`option aon_mode '0'`	This option must be used if the transceiver is an Active Ethernet device instead of PON.
`option aon_pol '0'`	Sets the polarity of an "active" Ethernet transceiver.

Note: Single quotes (') are required around the value.

Enabling persistent support for new transceivers requires compiling a new version of the 10G PON Chipset software, with the new device files placed in the `/etc/optic-db/` database folder. It is mandatory to then reinstall this new version to the EASY PRX321 REF BOARD. **Section 6.4** provides information regarding the file creation and naming process.

6.2.1 Autodetecting the Transceiver Light State

The autodetect function is a very important hardware feature of the PRX321. The TX burst control signal, TX_DISABLE, must be connected to a pull-up/down resistor to keep the transceiver light in the OFF state. To allow more flexibility when testing new transceivers, the switch at RefDes SW7, near the SFP+ port, provides this pull-up/down functionality for transceivers which lack them.

Figure 6 shows the relevant switches highlighted in red. Switch 3 enables pull-up and switch 4 enables pull-down. To disable both, leave both switches in the OFF position.

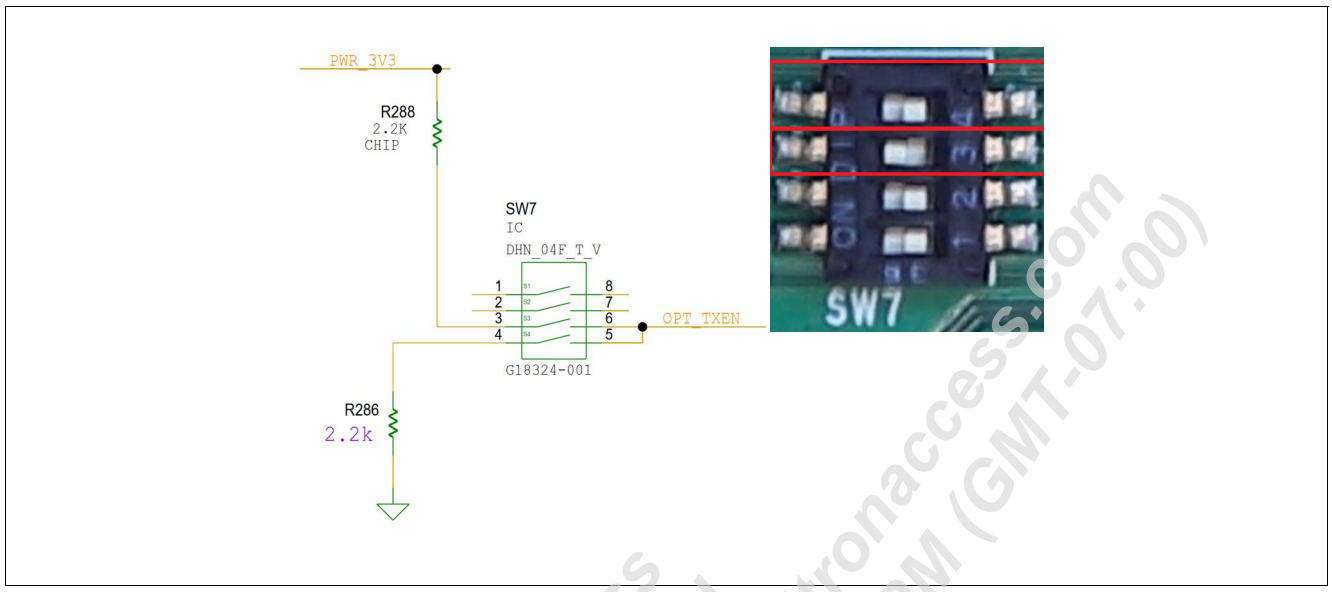

Figure 6 Switch SW7 for Transceiver TX Pull-Down/Up Selection

The PRX321 device detects the initial disable state during board power-up and defines the active state as the inverse of the initial state. This prevents transceivers from sending rogue signals before the EASY PRX321 REF BOARD has fully initialized or the firmware download has completed.

6.2.2 Calibrating the Optical Burst Position

After a new transceiver is recognized, perform a detailed burst calibration to adjust timing and ensure proper functionality. Refer to **[3]** for detailed instructions.

6.3 Enabling XGS-PON, XG-PON, or G-PON Mode

The EASY PRX321 REF BOARD supports these PON modes:

- XGS-PON (ITU G.9807): 10 Gbps downstream / 10 Gbps upstream
- XG-PON (ITU G.987): 10 Gbps downstream / 2.5 Gbps upstream
- G-PON (ITU G.984): 2.5 Gbps downstream / 1.25 Gbps upstream

The EASY PRX321 REF BOARD is configured to use XGS-PON mode by default. Edit the `/etc/config/gpon` configuration file to match the target ONT's operation mode. The selected optical ONU transceiver must support the targeted mode.

```
root@prx321-sfu-qspi-pon:/# vi /etc/config/gpon
config gpon 'ploam'
        option ploam_timeout_0 '65535'
        option ploam_timeout_1 '10000'
        option ploam_timeout_2 '100'
        option ploam_timeout_3 '65535'
        option ploam_timeout_4 '0'
        option ploam_timeout_5 '0'
        option ploam_timeout_6 '10000'
        option ploam_timeout_cpl '0'
        option regID ' 0x4D 0x41 0x58 0x4C 0x49 0x4E 0x45 0x41…
        option nSerial 'INTC96FBD540'

config gpon 'ponip'
        option pon_mode 'xgspon'
        option iop_mask '18'
```

Complete these steps to enable different modes:

1. Replace `xgspon` with either `gpon` or `xgpon`.
2. Save the file.
3. Power cycle the EASY PRX321 REF BOARD.

6.4 Identifying and Naming Transceiver Files

Multiple functions control the identification of transceivers. Understanding how these functions work allows developers to properly name files and add them to the database located at `/etc/optic-db/`.

The function `transceiver_names_get()` in the script `/etc/init.d/pondb` determines the vendor name, part number, and associated information of the transceiver.

```
transceiver_names_get() {
    local vendor_name="$(read_eeprom 20 16 | normalize)"
    local vendor_oui="$(read_eeprom 37 3 | hexdump -ve '1/1 "%.2x"')"
    local part_number="$(read_eeprom 40 16 | normalize)"
    local revision="$(read_eeprom 56 4 | normalize_revision)"
```

The `optic_files_get()` function returns a list of files in a dedicated order according to these criteria[1)]:

1. When the transceiver and MaxLinear board name are correctly identified, a file name is returned that consists of a string, such as:
   `superxonltd-sogx2699-psga-urx851-eva`
2. The transceiver name alone is returned:
   `superxonltd-sogx2699-psga`
3. **`Default`** is returned when neither the board name nor transceiver name is identified.
   `default`

```
optic_files_get() {
    local board="$(pon_board_name)"

    if [ -n "$board" ]; then
        # prepend directoy name and append board name
        #Example: /etc/optic-db/superxonltd-sogx2699-psga-urx851-eva
        transceiver_names_get | prepend "$OPTIC_DB_LOCATION/" | append "-$board"
    fi
    #We prepend directory name
    #Example: /etc/optic-db/wtd-001cad-rtsm166-501-1.0
    transceiver_names_get | prepend "$OPTIC_DB_LOCATION/"

    echo "$OPTIC_DB_LOCATION/default"
```

The function `config_apply()` iterates over the list and uses the first file found to update the optical configuration using UCI commands. Place the new files, with names which comply with the rules described in this section, in the `/etc/optic-db/` folder.

1) The order in which files are returned varies depending on the MaxLinear board version/type.

6.5 Verifying New Transceiver Files Function Properly

After a new transceiver file is created, it is important to verify that it functions properly with the EASY PRX321 REF BOARD before compiling a new extended 10G PON Chipset software image.

Verify the new transceiver file functions using this procedure:

1. Insert the new optical transceiver module into the SFP+ port of the EASY PRX321 REF BOARD.
2. Create a new transceiver file in `/etc/optic-db/` database folder with a name and content which adheres to the procedures described in this chapter.
3. Remove the line containing the software version information from the `/etc/config/optic` file:

```
root@prx321-sfu-qspi-pon:/# vi /etc/config/optic
config optic 'common'
....
        option version '1.16.0.r2'
```

4. Call the initialization script and verify the configuration is applied correctly.
   This example console output demonstrates how the 10G PON Chipset identifies the Superxon* SOGX2699-PSGA using the search routine to eventually choose the `superxonltd-sogx2699-psga` file from the database and apply it.

```
root@prx321-sfu-qspi-pon:/# /etc/init.d/pondb start
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga-10-prx321-sfu-qspi-pon' configuration
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga-prx321-sfu-qspi-pon' configuration
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga-10' configuration
[optic-db] Looking for '/etc/optic-db/superxonltd-sogx2699-psga' configuration
[optic-db] Applied '/etc/optic-db/superxonltd-sogx2699-psga' configuration
```

The values from the new transceiver file are copied to the `/etc/config/optic` file. From now on these values are used each time the board is booted, power-cycles, or rebooted.

Literature References

[1] 10G PON Chipset PRX321 (PRX321B0BI/PRX321B1BI/PRX321B2BI) Data Sheet Rev. 3.3

[2] Marvell* AQR113/AQR114/AQR115/AQR113C/AQR114C/AQR115C Gen 4 Multi-Gigabit Ethernet PHY Transceiver Datasheet

[3] 10G PON Subsystem Optical Timing Calibration Application Note Rev. 1.0

Attention: Refer to the latest revisions of the documents.