



Firmware Upgrade Process

Application note: firmware upgrade and recovery

The firmware upgrade process for the Pure Photonics tunable laser is, in general, a convenient and straightforward process. The Pure Photonics GUI and CLI provide several ways to perform the upgrade. This application note describes both processes.

In some instances, the firmware upgrade process does not complete properly. In this case, a temporary firmware (loaded in a different location of the micro-processor memory) will be active upon start-up of the laser. This firmware has limited functionality and is only intended to allow recovery after failure of the firmware upgrade process.

The failure of the firmware upgrade process is either an incorrect setting of the communications interface (often related to the use of an USB interface) or a power failure (on the +3.3V line) during the upload process. This application note describes on how to recover (both through CLI and GUI).

Before proceeding (either with upgrade or with recovery), first ensure that the USB communications interface is properly installed, as per the instructions in section 1.

For recovery refer to section 5.

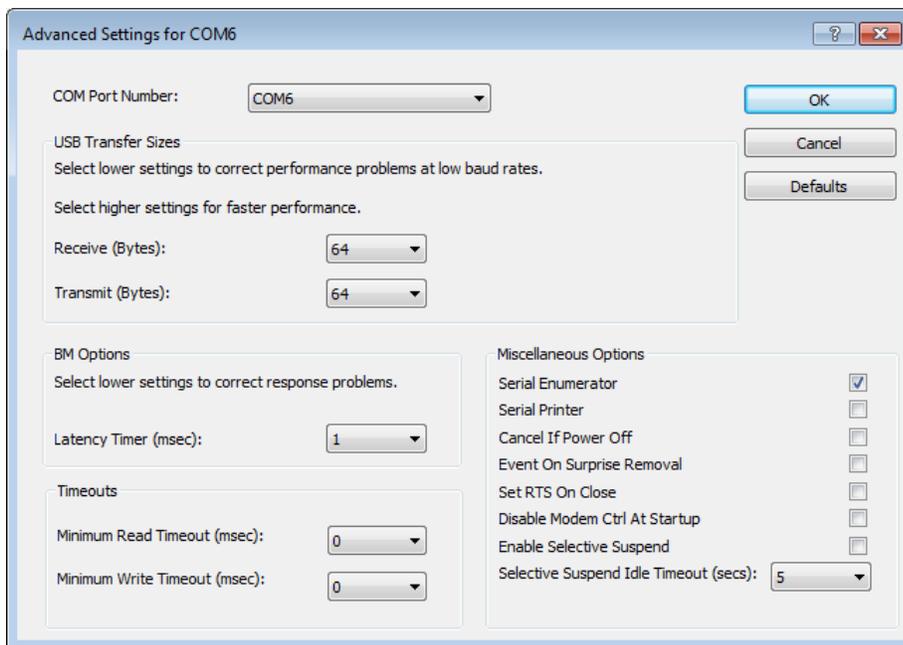
1. Configuring (micro) USB interfaces

The native communication with the module is through a low-voltage RS-232 interface. Several options exist where the customer-facing interface has been changed to a USB interface. For example, the PPCL300 comes with a micro-USB interface option, the CoBrite package comes with a standard USB interface and for computers without a COM port an RS-232 – USB dongle is used.

The standard USB interface is not optimized for serial communications on the RS-232 interface. Due to differences in protocol and optimization algorithms for the USB port, it installs with non-optimal settings. For most operations, this is not a problem, however for a firmware upgrade a proper configuration is required.

We recommend to make the below changes when the USB interface gets registered. On every computer this is only needed once for each USB interface/device.

- a. Open the ‘Windows Control Manager’
- b. Open ‘Hardware and Sound’
- c. Open ‘Device Manager’
- d. Find the COM ports
- e. Right-click the USB serial port and select ‘Properties’
- f. Select the tab ‘Port Settings’ and click the ‘Advanced’ button
- g. **Set the latency timer value to 1msec.**
 - a. You can also select the COM-port designation here (we recommend port # <10).
 - b. We recommend to set the ‘USB transfer sizes’ to the lowest possible setting (do this for both receive and transmit)
- h. Close the windows and start using the device



2. Firmware upgrade with the GUI

The GUI based firmware upgrade currently only works with ITLA type lasers (serial starting with CRTN). For upgrade of micro-ITLA (starting serial CRTM or PP70) the CLI should be used.

The Graphical User Interface is a set of tools to demonstrate and control the different features of the laser. It uses interfaces that are completely accessible to the customer and hence any feature in the GUI can be fully implemented by the customer in their own system.

The GUI is available for download on the Pure Photonics website, under the download section. The zip file needs to be downloaded and extracted to a separate directory. In the directory there will be 'firmware upgrade.bat' file. Double-clicking this link will launch the firmware upgrade GUI.

Upon launch, a main window with 3 steps will be visible. Only step 1 can be activated. Press 'step 1' and enter the COM port of the device in the field that appears (e.g. 'COM1'). If the device is a CoBrite unit, click the 'CoBrite' checkbox. Then press Connect. After 1-2 seconds a 'step 1 report' will appear with information on the unit that is connected.

In case that the COM port is occupied (used by another program) or not present a failure message will appear that says '*Serial Connection Not Established. Issue with the port.*'

In case the device is not powered-on or not connected a failure message will appear that says '*Serial Connection Not Established. Issue with the device.*'

If the 2nd failure message appears and it has been ensured that the device is connected and powered on, a recovery may be needed (see section 4 and 5)

After successful connection with the device the 'step 2' and 'step 3' buttons will enable.

Press 'step 2' button and select 'change dir'. Now you can select the directory where the firmware file (.ray file) is located. After selection, all the .ray files in that directory will be listed. Select the .ray file that you would like to load to the device and press 'select'. The name and path of the selected file will appear.

Now press 'step 3' and press 'upgrade'. A warning message will appear that says '*After you press the upgrade button, please make sure that there are no power interruptions.*'. Acknowledge the box and the firmware upgrade process will start. The process will take about 2.5 minutes for a good interface. For slow interfaces (and CoBrite units) the process can take 25 minutes. Please be patient as the progress bar may freeze.

After completion of the upgrade the progress bar will disappear.

3. Firmware upgrade with the CLI

The Command Line Interface is a tool to directly access the registers of the tunable laser. The CLI is available for download on the Pure Photonics website, under the download section. The zip file needs to be downloaded and extracted to a separate directory. In the directory there will be a .exe file to run the program.

Use the following sequence to perform firmware upgrade:

- `it.connect(1,9600)`
 - The first parameter is the COM port number. This may vary dependent on your configuration
 - The second parameter is the current baudrate of the device. Most devices start up with baudrate 9600
- `it.release()`
 - make sure that you get a response here. If not, something is seriously wrong and more trouble shooting is required. Contact Pure Photonics.
 - If the response is FW 0.0.0, then the temporary firmware version is active, indicating that the previous firmware upgrade did not terminate as intended.
- `it.baudrate(115200)`
 - 115200 is the highest available baudrate, resulting in the fastest upgrade
- `it.upgrade('application',r'c:\.....\.....ray')`
 - The item within parenthesis is the path to the .ray file that you wish to upload

After completion of the upgrade the interface will say: '*Seconds elapsed 140. Init_Run OK*'.

4. Firmware upgrade directly through RS-232

The OIF defined sequence for firmware download is provided in the below table.

Step	Host Sends	Module Responds
1	Write the DLConfig register indicating the type of transfer coming and asserting INIT_WRITE=1 and TYPE. Code might be boot code, run time code, FPGA code, CPLD code.	Module responds by initializing extended address registers (0x0E-0x0F) and may return a pending operation flag if pre-configuration will take longer than the allowed response time (§11.2).
2	Host writes to the extended address register (0x10) with the file data, 2 bytes at a time.	Module receives file data 2 bytes at a time and asserts a pending operation flag as necessary. Each out-bound packet response indicates if any errors have occurred.
3	Host writes DLConfig and asserts DONE=1.	Module completes transfer and performs any clean-up operations related to the write sequence. The module may respond with a pending operation flag.
4	Host writes DLConfig and asserts INIT_CHECK=1	The module performs the consistency check and sets the VALID bit accordingly in DLStatus. The module may respond with a pending operation flag.
5	Host reads DLStatus and checks for VALID=1	Modules responds with the DLStatus information
6	Host writes DLConfig with INIT_RUN=1 and asserts RUNV set to the same value as TYPE in step 1.	Module either (1) responds and is running the new requested code segment or (2) the module responds with a pending operation flag and begins the process of running the requested code segment.

The data is contained in the .ray file (binary data). It is important to append a 0 data value to the data if the length of the data is an odd number.

For a successful firmware update, make sure the unit is not enabled and make sure the power supply (especially the 3.3V) is stable.

In our CLI implementation we follow the following sequence (referencing the steps in the above table):

- STEP 1
 - Write Abort to DIConfig (0x02 to 0x14). If unit indicates an error, then don't proceed.
 - Write InitWrite to DLConfig with type=3 (0x3001 to 0x14). If unit indicates an error, then don't proceed.
- STEP 2
 - Binary read the ray file.
 - Add 0 value to the end if the file length is odd.
 - Feed data to module to EAR register (0x10) in increments of 2 bytes
 - The data in the ray file is big endian, meaning that the first byte is mapped to the upper 8 bits of the data field and the second byte to the lower 8 bits of the data field.
 - After each read verify that there is no error in the response. If the unit indicates an error log it, but proceed.
- STEP 3
 - Flush read buffer (just to be sure).
 - Write DONE to DIConfig (0x04 to 0x14).

- STEP 4
 - Write InitCheck to DIConfig (0x10 to 0x14). If unit indicates pending, check NOP until pending flags are dropped. If unit indicates an error, don't proceed.
- STEP 5
 - Read DIStatus (0x15). If VALID field (0x01) is not 1, don't proceed.
- STEP 6
 - Write InitRun to DLConfig (0x20 to 0x14) with type=3. If unit indicates and error, don't proceed.
 - Wait 0.5 seconds.

For completeness DIConfig (register 0x14):

15	14	13	12	11	10	9	8
TYPE				RUNV			
7	6	5	4	3	2	1	0
Reserved (0x0)		INIT_RUN	INIT_CHECK	INIT_READ	DONE	ABRT	INIT_WRITE

DIStatus (register 0x15):

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
														IN_USE	VALID

5. Recovery after failure of firmware upgrade

There are two methods to recover. The first one is to perform the process in section 3.

The second method is by using the GUI. You can copy the 'firmware upgrade.bat' file and rename it to 'firmware recovery.bat'. Right click the file and choose edit. Replace '*FWupgrade*' by 'recovery' (without the parenthesis).

Now double click the '*recovery.bat*' file and go through step 1, step 2 and step 3. This will restore the device and upload a reference firmware to the unit. From that point you can then load the firmware version that you would like to upgrade to.