

How to Upgrade X410-X440 MCU Firmware

Contents

- 1 Application Note Number and Authors
- 2 Revision History
- 3 Abstract
- 4 Procedure from Linux Host
 - ◆ 4.1 Procedure
- 5 Procedure on USRP device
 - ◆ 5.1 Procedure

AN-111 by Michael Dickens (draft of 2024-07-15)

This document details how to download, extract, and install the X410/X440 USRP microcontroller ("MCU") firmware updates. This specific guide shows how to do the update to MCU firmware version 0.0.13823-82c52e33a for any revision X410/X440 USRP (currently 5, 6, or 7).

The Procedure for the Read Only (RO) MCU firmware can be done from a Linux host computer/server or on the USRP itself assuming the X410/X440 USRP boots and can stay active and running for a minimum of 20 minutes. The Read/Write (RW) MCU firmware *must* be loaded from the Linux host computer/server; hence we recommend starting with the first Procedure on Step (1) and if that works then both the RO and RW MCU firmware will be successfully update. If you start with the second Procedure at Step (15) then only the RO portion will be updated, so you would then need to run the installer from the Linux host computer/server to get the RW portion updated.

Both are covered here; select the one that matches your current X410/X440 USRP situation & if possible, start with the first Procedure to try to get both the RO and RW firmware updated in one set of commands.

NOTE: This guide assumes the following:

1. A Linux host computer (we use Ubuntu / aptitude-based; others will work);
2. The X410/X440 USRP is accessible to the Linux host computer via a USB cable connected to the USRP USB JTAG port;
3. The X410/X440 USRP has power but is *not* powered on / is currently shut down;
4. The X410/X440 USRP is the only USRP device connected in any form to the Linux host computer (just in case);
5. This procedure *should* work regardless of whether the USRP can boot or not.

1. Make sure `openocd` is installed on the Linux host computer/server. The following command works for Ubuntu / aptitude-based: `sudo apt-get -y install openocd`.

2. Make sure the USRP and host computer are connected via the USB cable as noted in (b) above.

3. Use `screen` to access the MCU on serial port 2:

```
sudo screen /dev/serial/by-id/usb-Digilent_USB_Device_XXXX-if02-port0 115200
```

NOTE: `xxxx` will be some unique string for this specific serial device and Linux configuration, for example `25163520B2FE`. If you copy/paste the first part of the command, up to `usb-` and then press the `<tab>` key once or twice then all the options will be listed; use the entry for `if02` and then finish off the command as noted with `115200`.

4. Press the `<return>` key and the prompt `>` should appear; assuming it does then enter the command:

```
eeppromdump mb
```

NOTE: The first few lines of output will look something like the following

For an X410 USRP:

```
cache hit mb
```

```
usrp_eepprom_board_info (0x10) pid: 0x0410, rev: 0x0005, compat_rev: 0x0005, serial: 320B2FE
```

```
usrp_eepprom_module_info (0x11) pid: 0xa410, rev: 0x0003, serial: 32190B4
```

For an X440 USRP:

```
cache hit mb
```

```
usrp_eepprom_board_info (0x10) pid: 0x0440, rev: 0x0006, compat_rev: 0x0006, serial: 3284FCB
```

```
usrp_eepprom_module_info (0x11) pid: 0xa440, rev: 0x0004, serial: 32896F8
```

5. Note the `compat_rev` part of `usrp_eepprom_board_info`, in this case for the X410 USRP `5` and for the X440 USRP `6`; this value can currently be `5`, `6`, or `7` and, in the future, higher numbers.

6. Detach from screen via pressing the keys `^a` (`<control>`) then `a`) then `\`, then for the prompt (at the bottom) press the `y` key, which will come back to the usual shell prompt.

7. Download and extract the archive to the Linux host computer to the `~/Downloads` directory:

```
cd ~/Downloads
```

```
wget https://files.ettus.com/binaries/misc/upgrade_mcu_titanium_v0.0.13823-82c52e33a.tar.gz
```

```
tar xzf upgrade_mcu_titanium_v0.0.13823-82c52e33a.tar.gz
```

```
cd upgrade_mcu_titanium_v0.0.13823-82c52e33a
```

8. Run the "from host" MCU updater script:

```
sudo ./flash-firmware-from-host.sh
```

NOTE: This script will print the following:

This script updates the microcontroller firmware (RO and RW). The change is persistent across power cycles for the RO part. Running this script incorrectly can brick the device. This script may take upto a minute to run successfully.

Ensure that only one X410/X440 device is connected to the host machine at a time.

Steps:

1. Power off the X410/X440 device by running `"shutdown -h now"` or `"poweroff"` command from the Linux console. But leave the power connected and turned on.
 2. Connect your X410/X440 device using the USB JTAG port to the host machine using a USB cable.
 3. Ensure that only one X440/X410 device is connected at a time. No other USRP device should be connected.
 4. Run this script.
 5. Do not disconnect USB cable while the script is running as it may result in bricking of the device.
 6. Power cycle the device after script has completed running successfully.
- Updating the microcontroller firmware (RO part) is only required if the Ettus Research support told you to do so.

Press `"y"` to continue

9. Press the `y` key; the script will print:

Please enter the motherboard `compat_rev` number:

10. Enter the `#` from the `compat_rev` info from step (5) (in this case `5` for the X410 USRP or `6` for the X440 USRP but use whatever you determined & the printout below will change accordingly), then press the `<return>` key. The script will now print: The `compat_rev 5` is within the supported range.

Hardware Rev: 5

Will use firmware file `/home/uhd/Desktop/upgrade_mcu_titanium_v0.0.13823-82c52e33a/ec-titanium-rev5.bin`

Press `"y"` to continue

11. Press the `y` key; the script will now print out something like the following then terminate as well as try to boot the USRP:

```
Flashing ec-titanium-rev5.bin on hardware...
```

```
Open On-Chip Debugger 0.11.0
```

```
Licensed under GNU GPL v2
```

```

For bug reports, read
http://openocd.org/doc/doxygen/bugs.html
titanium
/home/uhd/Desktop/upgrade_mcu_titanium_v0.0.13823-82c52e33a
DEPRECATED! use 'adapter driver' not 'interface'
DEPRECATED! use 'adapter speed' not 'adapter_khz'
Info : clock speed 2000 kHz
Info : JTAG tap: stm32f4x.cpu tap/device found: 0x4ba00477 (mfg: 0x23b (ARM Ltd), part: 0xba00, ver: 0x4)
Info : JTAG tap: stm32f4x.bs tap/device found: 0x06441041 (mfg: 0x020 (STMicroelectronics), part: 0x6441, ver: 0x0)
Info : stm32f4x.cpu: hardware has 6 breakpoints, 4 watchpoints
Info : stm32f4x.cpu: external reset detected
Info : starting gdb server for stm32f4x.cpu on 3121
Info : Listening on port 3121 for gdb connections
Info : JTAG tap: stm32f4x.cpu tap/device found: 0x4ba00477 (mfg: 0x23b (ARM Ltd), part: 0xba00, ver: 0x4)
Info : JTAG tap: stm32f4x.bs tap/device found: 0x06441041 (mfg: 0x020 (STMicroelectronics), part: 0x6441, ver: 0x0)
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x080001f8 msp: 0x20000400
Info : device id = 0x30006441
Info : flash size = 512 kbytes
Info : JTAG tap: stm32f4x.cpu tap/device found: 0x4ba00477 (mfg: 0x23b (ARM Ltd), part: 0xba00, ver: 0x4)
Info : JTAG tap: stm32f4x.bs tap/device found: 0x06441041 (mfg: 0x020 (STMicroelectronics), part: 0x6441, ver: 0x0)
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x080001f8 msp: 0x20000400
shutdown command invoked

```

12. After the USRP shuts down, power it back up. Repeat command (3) then at the > prompt issue the command `version`; the output should look like the following for when the `compat_rev` is 5; the `# rev5` will match that from Step (5):

```

Chip: stm stm32f412
Board: 0
RO: titanium-rev5_v0.0.13823-82c52e
RW: titanium-rev5_v0.0.13823-82c52e
Build: titanium-rev5_v0.0.13823-82c52e
2024-03-04 15:58:54 @c09af2c5ef57

```

13. Power on the USRP if it did not automatically do so after Step (12).
14. Make sure that the USRP is connected to the host computer via a QSFP28 10/100 GbE link and/or the RJ-45 1 GbE link and that the networking is configured for your desired use-case, then run the following and see if the device comes up:

```
uhd_find_devices
```

NOTE:

1. If you followed Steps (1) through (12) and if this command successfully finds the USRP then you are done! Congratulations on successfully upgrading the MCU firmware on your X410/X440 USRP!
2. If the USRP does not boot or if this command does not succeed in finding the USRP, then try this whole procedure section again starting at Step (3). If you tried this already and the X410/X440 USRP is still having issues, contact Ettus support for assistance.

NOTE: This guide assumes the following:

1. The X410/X440 USRP has internet access, or is networked to the Linux host computer via the RJ-45 1 GbE or 10/100 GbE QSFP28 ports;
2. The X410/X440 USRP is powered on;

15. Power on the USRP if it did not automatically do so after Step (13), or if jumping here because the USRP successfully boots.
16. Make sure that the USRP is connected to the host computer via a QSFP28 10/100 GbE link and/or the RJ-45 1 GbE link and that the networking is configured for your desired use-case, then run the following and see if the device comes up:

```
uhd_find_devices
```

NOTE: Assuming this command successfully finds the USRP note the `mgmt_addr`. Typical values are 192.168.10.2 through 192.168.40.2, but this entirely depends on how the networking is configured. Let's call this value (whatever it actually is) `MGMT_ADDR`.

17. Login to this USRP via the UHD-found `MGMT_ADDR`:

```
ssh root@MGMT_ADDR
```

18. Download the archive, either directly or from the networked host computer attached to the USRP:

```
wget https://files.ettus.com/binaries/misc/upgrade_mcu_titanium_v0.0.13823-82c52e33a.tar.gz
```

19. Extract the archive and move into the directory:

```
tar xzf upgrade_mcu_titanium_v0.0.13823-82c52e33a.tar.gz
cd upgrade_mcu_titanium_v0.0.13823-82c52e33a
```

20. Run the "from device" MCU updater script:

```
./flash-firmware-from-device.sh
```

NOTE: This script will print the following:

This script updates the microcontroller firmware (RO part). The change is persistent across power cycles. Incorrect updates may brick the device and may need you to use the host based update process. This script may take upto a minute to run successfully.

Updating the microcontroller firmware (RO part) is only required if the Ettus Research support told you to do so.

Press "y" to continue

21. Press the `y` key; the script will now print out something like the following:

This script will flash ec-titanium-rev5.RO.flat to the device

old RO version: titanium_v0.0.13797-11efcf3e6

new RO: titanium-rev5_v0.0.13823-82c52e

Press "y" to continue

22. Press the `y` key (again); the script will now print out the following (note that the USRP here is `rev5`; `rev6`, `rev7` etc will show that revision number in the text instead of 5; this revision number is automatically detected by the `device` version of the install script):

```

# Disable periodic led control commands to reduce i2c activity to the controller.
ledctrl led0 none
/usr/sbin/ectool --interface=dev reboot_ec RW
/usr/sbin/ectool --interface=dev flashread 0x0 262144 ec-titanium-rev5.RO.flat.old
Reading 262144 bytes at offset 0...
done.
# This call may throw some i2c errors due to long erase times
/usr/sbin/ectool --interface=dev flasheraseasync 0x0 262144
Erasing 262144 bytes at offset 0...
ioctl -1, errno 16 (Device or resource busy), EC result 255 (<unknown>)
done.
# Sleep and try erase again, this should succeed.
sleep 3
/usr/sbin/ectool --interface=dev flasheraseasync 0x0 262144
Erasing 262144 bytes at offset 0...
done.
/usr/sbin/ectool --interface=dev flashwrite 0x0 ec-titanium-rev5.RO.flat
Reading 84032 bytes from ec-titanium-rev5.RO.flat...
Writing to offset 0...
Write size 112...
done.
copying new firmware files
'ec-titanium-rev5.bin' -> '/lib/firmware/ni/ec-titanium-rev5.bin'
'ec-titanium-rev5.RW.bin' -> '/lib/firmware/ni/ec-titanium-rev5.RW.bin'

```

23. Reboot the USRP:

reboot

24. All done! If you want to verify that the firmware is updated, then repeat step (12) and the output for RO should now be the following for rev5 USRPs; the other entries might still be different:

```
Chip: stm stm32f411
Board: 0
RO: titanium-rev5_v0.0.13823-82c52e
RW: titanium-rev5_v0.0.13823-82c52e
Build: titanium-rev5_v0.0.13823-82c52e
2024-03-04 15:58:54 @c09af2c5ef57
```

For rev6 and newer USRPs, the rev# text will not be present and the Chip might be different:

```
Chip: stm stm32f412
Board: 0
RO: titanium_v0.0.13823-82c52e
RW: titanium_v0.0.13823-82c52e
Build: titanium_v0.0.13823-82c52e
2024-03-04 15:58:54 @c09af2c5ef57
```