PROXMARK III

COMPILING PROXMARK SOURCE AND FIRMWARE UPGRADING

Version 1

INTENDED AUDIENCE

The Proxmark III is intended for users that are either competent hardware or software developers (preferably both). Users that do not understand the basic principles behind RFID may have difficulty using the device.

The Proxmark III is a RFID development tool. Typically, an "out of the box" proxmark3 with the latest firmware can run acquisitions in LF and HF mode, output traces, decode a number of different RFID credentials and do some operations in ISO 15693 and ISO 14443 a and b modes. If you really want to get the most out of this device, you will need to start enhancing the firmware yourself to suit your needs.

BEFORE YOU START

This document has been created assuming that you have read the relevant getting started guide and configured your development environment accordingly.

For Windows users – Everything in this document is done from the Minimalist GNU terminal window. Start by running "`runme.bat`".

COMPILING PROXMARK SOURCE

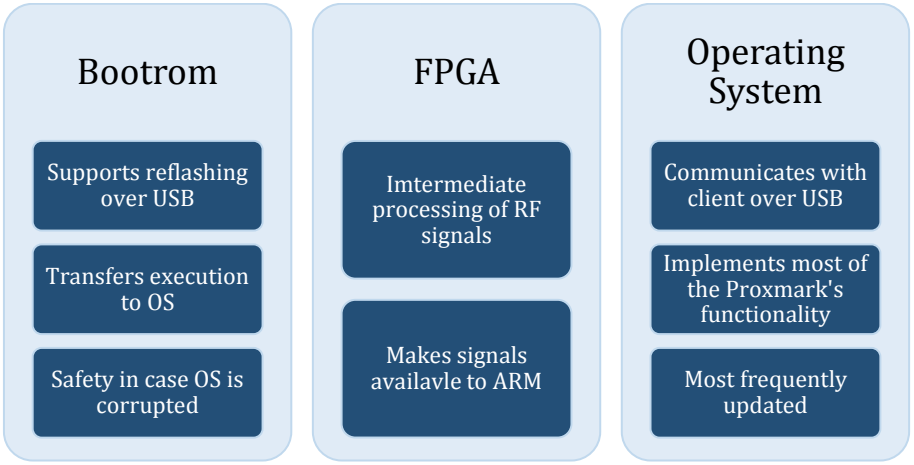
Compiling the Proxmark source should be straight forward. To build the project type "`make clean && make all`". The clean and build process will only take a few seconds before returning you back to the prompt. If the project was built successfully you should be able to run "`./client/proxmark3.exe`". Building the Proxmark project will also produce 3 elf firmware files:

- The bootrom image - "`./bootrom/obj/bootrom.elf`".
- The FPGA image - "`./armsrc/obj/fpgaimage.elf`".
- The OS image - "`./armsrc/obj/osimage.elf`".

If any mention of an error occurs we suggest you request help from the Proxmark community by pasting a copy of the terminal output from the build process and a description of what you were trying to achieve.

FLASHING FIRMWARE

Proxmark firmware is comprised of three logical sections: bootrom, fpga and operating system. The bootrom is a relatively small bit of code that performs some basic hardware initialization, supports reflashing the device over USB and knows how to transfer execution to the operating system. Due to the limited number of features exposed by the bootrom, it is not frequently updated and so you should only rarely need to update it when there is a compatibility conflict.



The FPGA code processes analogue signals coming from the antennas and makes those signals available to the ARM. Like the bootrom code, the FPGA code is not frequently updated. Presently, the operating system is the most frequently updated portion of Proxmark code. It is responsible for receiving and executing most of the commands advertised in the client user interface. Upgrading the bootrom of your Proxmark can brick the device. Please exercise caution when upgrading the bootloader. If the bootloader is corrupted, the only way to restore your Proxmark to working order will be through the use of a JTAG programmer.

FLASHING PROCEDURE

Ensure that you have read the prior section before proceeding. In order to upgrade to the latest version of firmware, you will need to first upgrade the Proxmark's bootloader.

The steps below will upgrade the Proxmark bootloader to the version you checked out previously using the procedure from the getting started guide.

1. Optional – Update your working copy to the latest revision. (Refer to the getting started documentation).
2. If you have not already done so, open up a terminal and go to the "`pm3/client`" directory.
3. Press and hold the button on the Proxmark while connecting it to your computer. Continue to hold the button until the yellow and red LEDs stay lit.
4. Upgrade the Proxmark bootrom by executing the following command:

```
sudo ./flasher -b ../bootrom/obj/bootrom.elf
```

At this point the bootrom has been updated and the Proxmark is now in a position to have its OS upgraded. The following steps will upgrade the Proxmark Operating System and FPGA code to your checked revision.

1. Ensure that the Proxmark is not connected to the PC.
2. Hold down the Proxmark's button and connect it to the PC. After the yellow and red LEDs are lit, execute the command below:

```
sudo ./flasher ../armsrc/obj/fpgaimage.elf
```

1. If the previous step is successful, disconnect the Proxmark.
2. While holding the button, connect the Proxmark to the PC and wait for the yellow and red LEDs to stay lit. Execute the command below:

```
sudo ./flasher ../armsrc/obj/osimage.elf
```

3. Disconnect the Proxmark from the PC and then reconnect it.
4. Launch the client software by executing "`./proxmark3.exe`". The client should successfully connect to the Proxmark. You should see something like this:

```
Connected units:
        1. SN: ChangeMe [bus-0/\\.\libusb0-0001--0x9ac4-0x4b8f]
proxmark3>
```

JTAG RECOVERY PROCEDURE

If for whatever reason the USB upgrade procedure (section above) failed and the Proxmark will no longer boot, you will need to load the bootrom on to the Proxmark using the JTAG interface. This procedure assumes that you have a Segger J-LINK for the recovery process and J-Flash ARM installed on a PC running Windows 7.

1. Plug the Proxmark and the Segger J-LINK in to the computer.
2. Attach the J-LINK to the Proxmark JTAG port.
3. Open J-Flash ARM.
4. Create a new project by selecting **File** > **New project**.
5. Change the project settings by going to **Options** > **Project settings…**

   – or –

   Alt + F7.

6. Click on the Target Interface tab.
7. Change the JTAG speed before init to 200 kHz.
8. Click on the CPU tab.
9. Select the Device radio button and select Atmel AT91SAM7S256 from the drop down list.
10. Change the Clock speed to Auto detection.
11. Apply and OK the changes.
12. Test connectivity to the Proxmark by selecting **Target** > **Connect**.

13. Disconnect from the Proxmark by selecting **Target** > **Disconnect**.
14. Open the data file by selecting **File** > **Open data file…**

    – or –

    Ctrl + O.

15. Open the file "`bootrom.s19`" which should be located in the "`pm3\bootrom\obj`" folder. If it is not there, you will need to re-build the project.
16. Program the Proxmark and verify the process by selecting **Target** > **Program & Verify**

    – or –

    F6.

17. You should expect to see an error saying that the program does not fit into the selected flash sectors. Select Yes to relocate and continue.
18. Disconnect from the Proxmark by selecting **Target** > **Disconnect**.
19. Exit J-Flash ARM.
20. Remove the J-LINK from the Proxmark.
21. Wait a little while and the Proxmark should automatically re-start. You can always un-plug and re-plug the Proxmark if you wish. You can then attempt to re-connect to the Proxmark using the client software.

Thanks to the proxmark3.com guys for letting me use some of their original content!

USEFUL LINKS

The Proxmark forum . . . . . http://proxmark.org/forum/index.php

Proxmark III online store http://proxmark3.com/

Proxmark project downloads http://code.google.com/p/proxmark3/downloads/list

The Proxmark project http://code.google.com/p/proxmark3/

The Proxmark wiki . . . . . http://code.google.com/p/proxmark3/wiki/HomePage?tm=6

Proxmark repository http://proxmark3.googlecode.com/svn/trunk/

Document corrections or comments . . . . . http://proxmark.org/forum/viewforum.php?id=27