Mavic Flashing Faq V0.7 by vk2fro & dreadwing007

Flashing the Mavic Pro Drone is bloody simple with the pyduml.py script. You will need:
a linux or mac box (live usb will work for linux). Without help from dreadwing007, the linux side would have been a fail!

First lets set up the environment. Open a terminal and complete these to satisfy dependencies
a mavic pro drone and connecting cable
the script (pyduml.py) - https://github.com/hdnes/pyduml
(mac only) terminal: /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
python (linux) should already be there (mac) brew install python
pip (common to both OS's)
pathlib (linux) sudo pip install pathlib (mac) sudo pip install pathlib
pyusb (linux) SF pyusb project (mac) pip install pyusb
libusb (linux) libusb.org (mac) brew install libusb
adb (if you would like root)
a firmware file (check #archived_fw_flashing for my links (vk2fro) or grab them off hostile's git (this one if your reading from there).
a half hour and a coffee/beer

<<<IDENTIFY THE DRONE PORT>>>

open up a terminal and cd to /dev with the mavic connected and powered on and type "ls tty*" and you'll find the by repeating the command after disconnecting the bird and seeing which item disappears from the list. (in my case, tty.usbmodem14E5). On a linux box (mint 18.1) we have discovered that both a spark and a mavic show up as ttyACM0. In essence, you are looking for the single device that appears and disappears conicidentally as you unplug and replug the drone.

<<<FLASHING>>>

FULLY CHARGE A BATTERY!!! do not try to flash with a depleted battery - it'll only end in tears and a dead mavic.

To flash, put the desired firmware into the pyduml.py folder

open a teminal and cd into the pyduml folder - if its in you home directory, cd ~/pyduml (don't forget the ~)

type mv <firmware name> dji_system.bin (this changes the name to work with the script) example “mv V01.03.0700_Mavic_dji_system dji_system.bin”

execute python pyduml.py /dev/<your.modem.number> (on my mac tty.usbmodem14E5) (on dreadwing007’s linux box: /dev/ttyACM0).

example: MAC “python pyduml.py /dev/tty.usbmodem14E5”
LINUX “sudo python pyduml.py /dev/ttyACM0”

Flashing takes around 10 minutes. Watch the lights on the mavic. The drone may reboot during the procedure but don’t touch it. Once it chimes and the front beacons stop flashing, its done. Be patient - 10 minutes seems like an awfully long time, but you don’t get a progress bar like when you flash with assistant. Do not disconnect the drone when the script says its finished - thats only the upload portion!

If your rooted, adb in and issue “busybox tail -f /ftp/upgrade/dji/log/upgrade00.log” if you want to monitor the progress.

<<ROOTING>>

To root, rename fireworks.tar dji_system.bin and re run the script as above. Power cycle the drone 2 minutes after running the script. DO NOT TURN OFF YET after the power cycle.

Now adb devices, and you should see the mavic listed. (RedHerring has Fangs!). adb shell into it and execute these two lines (use copy and paste, a typo = a brick!) in the shell:

“mount -o remount,rw /system”
VERY IMPORTANT - DONT COCK THIS NEXT LINE UP OR YOU’LL GET A BRICK!
“echo /system/bin/adb_en.sh >> /system/bin/start_dji_system.sh”
“reboot”

This makes the root permenant.