

Using the SSD as root file system for Raspbian

We have seen a number of questions related to booting the Pi from the SSD on the Multi-function shield. While it is not technically possible to *boot* the Pi from the SSD, the Pi can be configured to use the SSD to store and retrieve the root filesystem. This provides a number of benefits, including:

- Increased storage size for the root filesystem – it is almost certain that the SSD you choose will be substantially larger than your SD card. This should also impact reliability as there will be more pages in the wear leveling pool for the higher capacity SSD.
- Faster storage access – the SSD is a faster medium than the SD card. This will result in performance increase, especially for applications that read and write a lot of data.
- Direct access – all reads and writes of files in the main filesystem will occur in the SSD, without having to manually mount and access data through a mount point.
- Fewer writes to SD card – since the root filesystem and the swapfile have been moved to the SSD, the only writes to the SD card will be editing files in the SD card partition mounted on `/boot`.

The kernel boot configuration files will still be read from the SD card partition mounted on `/boot` on each boot. Here we note that SD cards are degraded by repeated writes, but should not be degraded by only repeated reads.

The initial setup to use the SSD for root partition has been quite simplified by the folks at Adafruit Industries. They have provided a pdf [guide](#) and a github hosted setup [script](#). Be careful though. Read the guide, and know that this operation will destroy any contents or formatting on the SSD (or other device!) at `/dev/sda`.

Before running the script, make sure the SSD is plugged in and accessible (running `lsusb` should show an entry for the drive) and make sure you are connected to the internet (you can confirm this by running `ping www.google.com`). The steps to run the script are quite simple:

```
pi@raspberrypi ~ $ git clone git://github.com/adafruit/Adafruit-Pi-ExternalRoot-Helper.git
```

(If, rather than cloning a local version of the script anonymously you would like to use ssh and your git account, run `git clone git@github.com:adafruit/Adafruit-Pi-ExternalRoot-Helper.git` as directed in the Adafruit guide instead. We do not suggest this for a simple script.)

```
pi@raspberrypi ~ $ cd Adafruit-Pi-ExternalRoot-Helper
```

```
pi@raspberrypi ~ $ sudo ./adafruit-pi-externalroot-helper -d /dev/sda
```

You will have to confirm some operations by typing 'y'. As the script instructs, after it is done, you must run `sudo reboot` to enact the new configuration. After doing this, you can run a few commands provided in the Adafruit pdf guide to confirm everything went as planned.

```
pi@raspberrypi ~ $ df -h
```

will show the additional free space in the rootfs.

```
pi@raspberrypi ~ $ readlink /dev/root
```

will return 'sda1', which is the first partition on the SSD. We can check that it is mounted on /:

```
pi@raspberrypi ~ $ mount | grep root
/dev/root on / type ext4 (rw,noatime,data=ordered)
```

then we can double check that the drive is available by uuid and label:

```
pi@raspberrypi ~ $ ls -l /dev/disk/by-uuid/
```

```
pi@raspberrypi ~ $ ls -l /dev/disk/by-label/
```

We can also check the write speed like this:

```
pi@raspberrypi ~ $ sync; echo 3 | sudo tee /proc/sys/vm/drop_caches
;dd if=/dev/zero of=test.tmp bs=500K count=1024
```

and the read speed:

```
pi@raspberrypi ~ $ sync; echo 3 | sudo tee /proc/sys/vm/drop_caches
;time dd if=test.tmp of=/dev/null bs=500K count=1024
```

And finally, after all that we will want to delete the test file with:

```
pi@raspberrypi ~ $ rm test.tmp
```

The original root filesystem will remain on the SD card after it is copied from the SSD. There will also be backup of the original `/boot/cmdline.txt` name `/boot/cmdline.txt.bak`. To recover from a failed boot or go back to using the SD card, all you have to do it plug the SD card into a computer and rename `/boot/cmdline.txt` to be something else like `cmdline.txt.ssd` and rename the backup `cmdline.txt.bak` `cmdline.txt`. Then plug the SD card back into the pi and it will boot from the SD card as before.