

Ramdisks for the Raspberry

Posted on **February 4, 2013**

If you use your [Raspberry Pi](#) as a standalone (headless) device, e.g. to measure and log the room temperature, there is no need to write all log files etc. to the SD card. Instead you can mount ramdisks into the relevant directories. This reduces the number of write accesses to the memory card and does not wear out its flash memory.



— Raspberry Pi used as a standalone temperature sensor and logger.

A standalone raspberry might be powered off without shutting down linux by calling *halt* or *shutdown -h now*. If there is a write access to the sd card during power off, it might damage the internal filesystem structure and if you are really unlucky (as has happened to me...), the filesystem is damaged in a way that the raspberry cannot boot anymore. Keep in mind to regularly create backups of your sd card!

At least the following directories are written to by the linux system, so they should be mounted to ramdisks:

- */var/log* – System logfiles
- */var/lock* – Lock files for active processes
- */tmp* – Temporary files

You can do this by adding lines to the file `/etc/fstab`:

1	proc	/proc	proc	defaults	0	0
2	/dev/mmcblk0p1	/boot	vfat	defaults	0	2
3	/dev/mmcblk0p2	/	ext4	defaults,noatime	0	1
4	tmpfs	/tmp	tmpfs	defaults,noatime,mode=1777	0	0
5	tmpfs	/var/log	tmpfs	defaults,noatime,mode=0755	0	0
6	tmpfs	/var/lock	tmpfs	defaults,noatime,mode=0755	0	0

The lines starting with *tmpfs* are the important ones that mount ramdisks to the specified directories. The *tmpfs* filesystem implements a ramdisk that adapts its memory usage to the space needed in the ramdisk. So you do not need to specify its size in the options of `/etc/fstab` and consume only as much ram as is needed by the files in the ramdisk.

If you use a webserver like *apache* or *nginx* on the raspberry, you need to take care of the directory where the server stores its logfiles. This is `/var/log/apache` or `/var/log/nginx`. Both webserver do not create this directory if it is missing and do not start. So you need a solution to create this log directory, before the webserver process is started.

Therefore I have written an init script named *prepare-dirs* that creates this directory, in my case for the *nginx* webserver.

```
#!/bin/bash
#
### BEGIN INIT INFO
# Provides:      prepare-dirs
# Default-Start: 2 3 4 5
# Default-Stop:  0 1 6
# Required-Start:
# Required-Stop:
# Short-Description: Create /var/log/nginx directory on tmpfs at startup
# Description:      Create /var/log/nginx directory on tmpfs at startup
### END INIT INFO

DIR=/var/log/nginx

#
# main()
#
case "${1:-''}" in
    start)
        # create the /var/log/nginx needed by webserver
        if [ ! -d ${DIR} ]; then
            mkdir ${DIR}
            chmod 755 ${DIR}
        fi
        ;;
    stop)
        ;;
    restart)
        ;;
    reload|force-reload)
        ;;
    status)
        ;;
    *)
        echo "Usage: $SELF start"
        exit 1
        ;;
esac
```

The script should be put into `/etc/init.d` and enabled in the boot process by calling

This creates links in the `/etc/rcX.d` directories and makes sure that the script is called with the start or stop parameter when entering or leaving the corresponding runlevels. If you want to know more about the boot procedure on [raspbian](#), see the [Upstart Website](#) or the [Upstart Wikipedia entry](#).

The numbers 01 and 99 tell `update-rc.d` to place the execution of the script at the beginning of the boot process and at the end of the shutdown process. The `nginx` webserver e.g. has the boot priority 02 by default, so it is started after the `prepare-dirs` script has been executed.

After editing `/etc/fstab` and setting up the init script, you should reboot the raspberry, so the ramdisks get mounted. You can check if everything works correctly by calling `mount`. It should print out a list with all mounted filesystems as specified in `/etc/fstab`.

With this setup you can prevent most of the write accesses to the sd card. To guarantee that the root filesystem (the one containing the data on the sd card) is never modified, it should be mounted read-only, so that it cannot be damaged in any way. I hope to have enough time in the next days to try that out.

This entry was posted in Uncategorized and tagged [embedded](#), [linux](#), [raspberry_pi](#) by [andreas](#). Bookmark the [permalink \[http://www.a-netz.de/2013/02/ramdisks-for-the-raspberry/\]](http://www.a-netz.de/2013/02/ramdisks-for-the-raspberry/).

3 THOUGHTS ON "RAMDISKS FOR THE RASPBERRY"



Angie

on [July 3, 2013 at 21:56](#) said:

you can also disable sound:

```
mv /lib/modules/$(uname -r)/kernel/sound /lib/modules/$(uname -r)/kernel/sound.disable
```

that make more free RAM for your applications



john

on [October 19, 2013 at 21:10](#) said:

Thanks very much for this. Very useful indeed.



Keule

on [November 10, 2013 at 17:50](#) said:

in case of Lighttpd Webserver, you should add after line 23:
chown -R www-data:www-data \${DIR}