Radiation Software Information

Written by Nicolas Douard (nicolasdouard@gmail.com) for The University and Manchester under MIT license.

The "radmon" units are Radiation Monitoring units. They perform measurement and send data to the radmon server. The server, known as "radser" is a FTP server on which all the data is uploaded.

Two program are executed on the radmon units on startup: “RadiationMonitor” and “RadiationUpload”.

- RadiationMonitor is written in C++ and handles monitoring (I/O) and logging to text files.
- RadiationUpload is written in Python and handles periodic uploads to the FTP server.

These programs are automatically started with console output when the unit starts. In case of error, both automatically restart due to the fact they are started through bash scripts within an infinite “while” loop. To stop a program manually, do Ctrl+C on its console window.

To restart both programs, stop them using Ctrl+C and execute “run.sh” which is located in /home/pi/radmon or simply reboot the unit using Menu > Shutdown > Reboot.

RadiationMonitor

RadiationMonitor reads “radmon.cfg” and won’t start if this file isn’t properly set. It should contain the name of the measurement unit on the first line with no spacing or character other than alphabetic and numbers. This value is used to set the name of log files.

The data is logged to monthly files. When a new month begins, the unit automatically restarts and a new file Is created.

RadiationUpload

RadiationUpload reads and writes “radser.cfg”. If “radser.cfg” doesn’t exist, it will create a new default file. The user than has to edit this file with proper information (server IP, username, password) and restart the program.

RadiationUpload also reads “radmon.cfg” to determine the name of the current log file. It won’t start without that file.

The current log is automatically uploaded to the server every 10 minutes. **It is possible to change this duration by editing “RadiationUpload.py”** with a text editor. You simply have to change the expression on line 120. Currently, it is as follows: “threading.Timer(600, Main).start()”.

The “600” value corresponds to the sleep duration in seconds between two uploads. If you want to set it to upload every 30 minutes, just change “600” for “1800”. RadiationUpload.py is a Python program and Python programs are interpreted, not compiled, so saving the file and closing the editor will be sufficient to update the delay. **If you do this on the Raspberry Pi directly, make sure to stop the program before editing the file and restart it afterwards.**

/home/pi/radmon Folder

This folder should contain the following files:

- RadiationMonitor
- RadiationUpload.py
- radmon.cfg
- radmon.sh
- radser.cfg
- radupl.sh
- run.py
- run.sh

On startup, the system runs “run.py” which calls “run.sh”. “run.sh” launches console instances for “radmon.sh” and “radupl.sh”. “radmon.sh” invokes “RadiationMonitor” and “radupl.sh” invokes “RadiationUpload.py”. The programs themselves are invoked in a loop so that they will restart if they crash.

The files “radmon.cfg” and “radser.cfg” are config files as described previously.

Log files and FTP server organisation

The log files are monthly and named using this convention: “unit_log_Y-M.txt” where “unit” is the unit name (for example radmon1), “Y” is the current year (for example “2016”) and “M” is the current month (for example “08”). A possible name is: “radmon1_log_2016-08.txt”.

The log files are stored in /home/pi/radmon on the units.

On the FTP server, one folder is created for each unit. For example, the “radmon1” folder will be attributed to the “radmon1” unit and will contain all monthly logs that unit uploads to the server. This name is the one set in the “radmon.cfg” file.

Radmon Unit Setup from scratch (for quick setup, use the image instead)

1. Install Raspbian using NOOBS: https://www.raspberrypi.org/documentation/installation/noobs.md

2. Connect the Raspberry Pi to the Internet, set the hostname of the unit in Menu > System > Hostname, change the password here as well and set the localisation settings which are available in: Menu > Preferences > Localisation

3. Follow this tutorial to install arduPi libraries (from 3. to 3.5, make sure to compile the library manually in 3.5): https://www.cooking-hacks.com/documentation/tutorials/raspberry-pi-to-arduino-shields-connection-bridge/

Rebuild RadiationMonitor by putting the source file RadiationMonitor.cpp into /home/pi/cooking/arduPi and run "g++ -lrt -lpthread RadiationMonitor.cpp arduPi.o -o RadiationMonitor".

4. Install the following Python packages: configparser, pathlib, ftplib.

https://docs.python.org/3/installing/

5. Copy the "radmon" folder into /home/pi. Copy the newly compiled RadiationMonitor (3.) from /home/pi/cooking/arduPi to /home/pi/radmon.

6. Make radmon.sh and run.sh executable by doing the following: right click on the file > properties > permissions tab > execute: change to "Only owner and group".

At this point, clicking on "run.sh" should start the two console applications.

7. Add /home/pi/run.py to autostart by doing the following:

run "sudo nano ~/.config/lxsession/LXDE-pi/autostart"

add "@sudo python3 /home/pi/radmon/run.py" on the first line

press Ctrl+X to save and exit

This allows the two programs to start automatically with the desktop.

Radmon Unit Setup from image

Clone image to SD

Note: these instructions are for **Windows only**.

Required software:

- SDFormatter (https://www.sdcard.org/downloads/formatter_4/) or HP USB Disk Storage Format Tool if the SD contains a previous system installation. (http://files.extremeoverclocking.com/file.php?f=197)
- Win32 Disk Imager https://sourceforge.net/projects/win32diskimager/

1. Insert the SD card back into your computer.

2. Format your disk using one of those tools. With SDFormatter, simply pick the drive and click format. This doesn't work for SD cards with a previous system installation as there are several partitions Windows doesn't recognize.

3. Open Win32DiskImager and browse for your image file. Select your device from the Device dropdown just as you did before.

4. Click "Write" to write the image to the SD card.

5. When it finishes, eject the SD card and re-insert it into your Raspberry Pi.

Configure Unit

1. Change the hostname by going to Menu > Preferences > Raspberry Pi Configuration > System tab and change 'Hostname' to something else than 'radmon1' (possibly 'radmon 2', 'radmon 3' etc.)

2. Change the password. If you don't, it will be the same as radmon1's password as it is an image from that unit.

3. Open the file manager (Menu > Accessories > File Manager) and browse to "home/pi/radmon". There are two file you need to edit here: radmon.cfg and radser.cfg

4. Double click radmon.cfg to open it in the text editor. Change "radmon1" to the unit name (normally the same as hostname). Close the window and save.

5. Double-click radser.cfg to open it in the text editor. This file defines the IP of the server (radser) as well as the username and the password used to log in. Change the IP to the current IP of the server (radser) on the local network. Close the window and save.

6. Restart the unit (Menu > Shutdown > Reboot).