User's and Developer's Guide for the LEOPARD RaspberryPi Based Data Acquisition System and its Graphical User Interface

Gergő Hamar

Wigner Research Centre for Physics, Budapest
Dept. High Energy Physics
Detectorphysics Group

Budapest, 2014.

Contents

Part I
USER'S GUIDE

1 Introduction

The Leopard project is about to examine photo-electron extraction efficiency and gas gain variation of micropattern based detectors on the submillimeter level. The first attempt to measure the above mentioned quantities was a success [**?**]. Countinuation and upgrade of the setup became an RD51 Common Project in 2013, with the collaboration of the Wigner RCP (Budapest, REGARD Group) and INFN Trieste (COMPASS RICH Upgrade Group).

In Budapest we have decided to build a fast, single channel, cost efficient data acquisition (DAQ) for the Leopard system. It is based on RaspberryPi mini computer with some custom designed boards for data collection and triggering. This manual is about the "Leopard RaspberryPi Based Data Acquisition System" (LRD), and its Graphical User Interface (LRDG).

2 The RaspberryPi

The RaspberryPi [1] (RPi) is a mini computer with a 700 MHz processor and all the main standard computer interfaces and additionally several GPIO pins. These features allow us to build a compact tool for the whole data acquisition system.

As the RaspberryPi is a computer, it runs an operating system, called Raspbian [2], which is a Debian based Linux for the RaspberryPi. The basic data storage unit is a standard SD card, where the OS is located on as well. (For list of usable SD card types check [7].)

The DAQ board uses the GPIO pins, while the USB outputs are used for WiFi connection and control of the HV modules.

The RaspberryPi needs a 5V power ($\approx$ 1A) through its microUSB connector. Eighter its standard power supply or custom designed can be used, however noise level should be checked for non-high-performance units.

Note, there are several models of RaspberryPi at 2014, changing the number of ethernet and USB ports, and varying the GPIO pin configuration. The LRD board was designed to fit onto the GPIO pins of the advanced and popular model : **RaspberryPi Model-B Rev-2**. Possible connection for other models can be checked in the Developer's Guide section of this document.

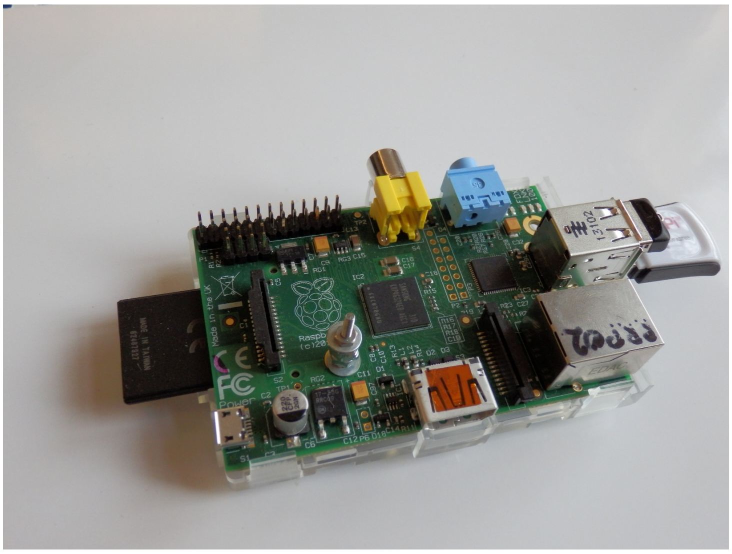


Figure 1: Photo of the *RaspberryPi Model-B* with all its GPIOs mounted. On the left the SD card can be seen, while on the other side there is a USB WiFi reciever and a USB flesh storage unit.

3 The LRD Board

The data acqusion board is a two-sided printed circuit board, designed by the REGARD Group in the Wigner R.C.P. (Budapest).

The incomig analog signal is grounded with 50 Ohms, and goes through a switchable preamplifier (3 switch positions for 0/1/2 stage amplification), and goes towards the LTC1415 analog-to-digital converter [8]. The signal should be in the [0:5] Volt range, and the pedestal can be set via a small potentiometer on the board.

For data taking a trigger shall be proceeded. The trigger is grounded via 50 Ohms, and recieved by a level shifter, where the baseline can tuned via a small potentiometer. The trigger signal should be in the range of [-5:5] Volts, while its level jumps shall not exceed 5 Volts. The processing is edge driven, and its polarity can be choosen a jumper on the trigger reciever part of the board. The selected leading edge can be delayed via a small potentiometer.

However the built-in busy logic prevents data taking during ongoing meaurements, the missed triggers are observed and the corresponding flag is read out; thus making possible a software driven digital signal processing.

The LRD board recieves the control bits from the RPi and passes them directly to the actuator system via TTL signals. It has 6 one-directional pins for this purpose what is essentially enough for direct or indirect actuation as well. The motor control protocol is specified in the LRD Software, thus any type of movement controller can be used.

Communication with the RPi is made through the RPi GPIO pins. The board directly connects to the 26+8 GPIO pins of the RPi, which holds it mechanically from than on as well. There is a place for an additional spacer right above the one on the RPi, what is recommended to be used. Note: On new RPi's the additional 8 pins should be soldered.

For setting the values of the potentiometers a digital oscilloscope may be used to ensure proper tuning of the voltage levels and timing properties. For this purpose two measurement points have been implemented close to the ADC.

A photo of the LRD Board can be seen on figure 2. with marks for all the mentioned main elements. A working setup with cables and board mounted onto the RPi can be seen on figure 3.

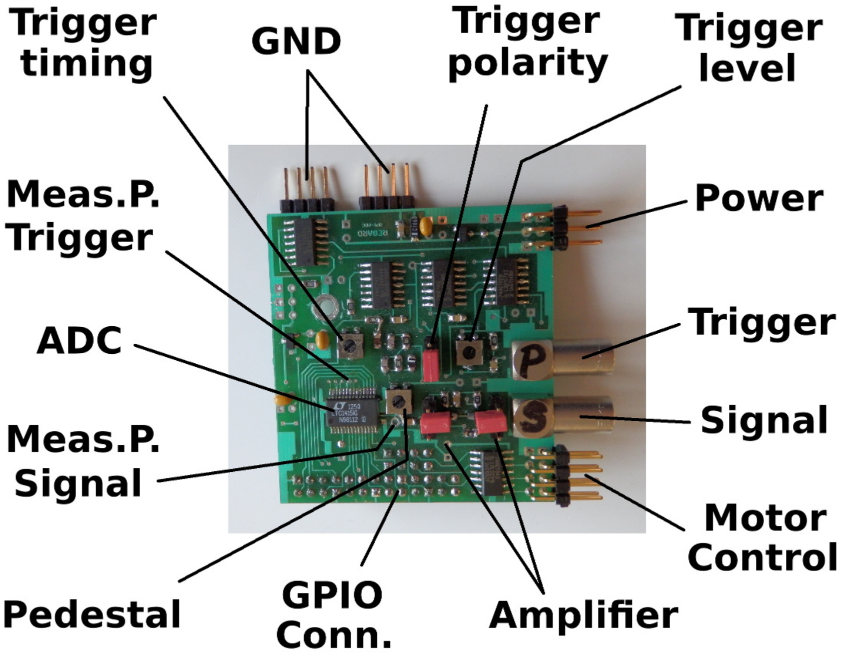


Figure 2: Photo of the Leopard RaspberryPi Daq (LRD) board, with labels.

3.1 LRD Board Connections

Specification of the connectors on the LRD Board:

- Grounding: Two 4x1 male pinheads.
- Power for the board: 3x2 pinhead.
  Middle ones for the power, other four for ground; input voltage should be between 6-10 V.
- Trigger input: LEMO socket with 50 Ohm termination;
  Any kind of a leading edge driven digital pulse, voltages between Ground and PowerVoltage.
- Signal input: LEMO socket with 50 Ohm termination.
- Actuator control: 5x2 flatcable socket.
  2+2 on the sides are ground, 6 middle ones are TTL out.
- RPi connection: 13x2+4x2 female pinhead.
  Directly fits onto the RPi GPIO pins.

3.2 Board hardware settings

Specification of the switches and potentiometers of the LRD Board

- Trigger level potentiometer.
  A level between 0-5 Volts should be set to level with the used input trigger signal. The set threshold should discriminate between the low and high states of the use input logic.
- Trigger pulse polarity switch.
  The converted trigger (TTL) should have positive leading edge. The two states o the switch selects between the direct and the inverted converted trigger, choose where the leading edge is positive.
- Trigger delay potentiometer.
  The proper delay for the trigger's leading edge can be tuned to be in syncron with the maximum of the analog signal.
- Signal postamplification switches.
  There is a pair of switches to set the number of postamplification stages to 0,1 or 2. For each of the three states the two switches should be put into mirror-symmetric positions. The states are 1,0,2 counting from the motor control connector.
- Signal pedestal potentiometer.
  The pedestal of the ADC can be set in the full range, thus eighter positive or negative signals can be measured.

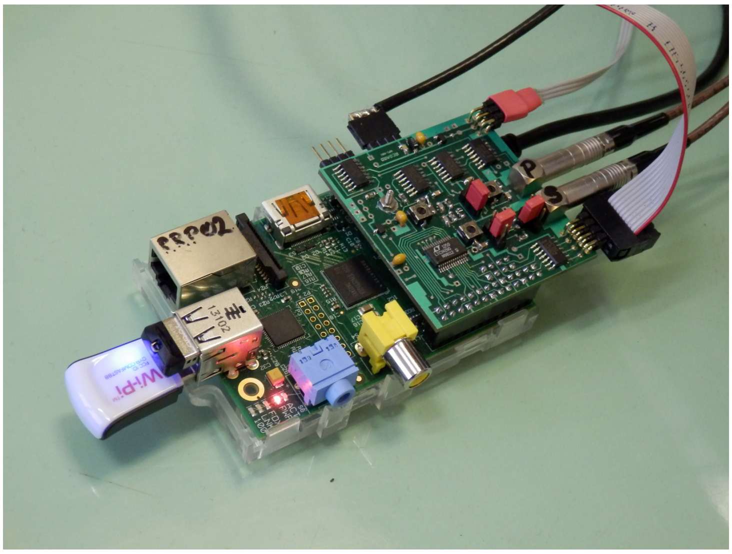


Figure 3: RpiWithBoardAndCables

4 The Leopard setup

The full Leopard setup is a moderate complexity system, its connection schematics is shown in figure 4., while a photo of a working setup can be seen on figure 5.

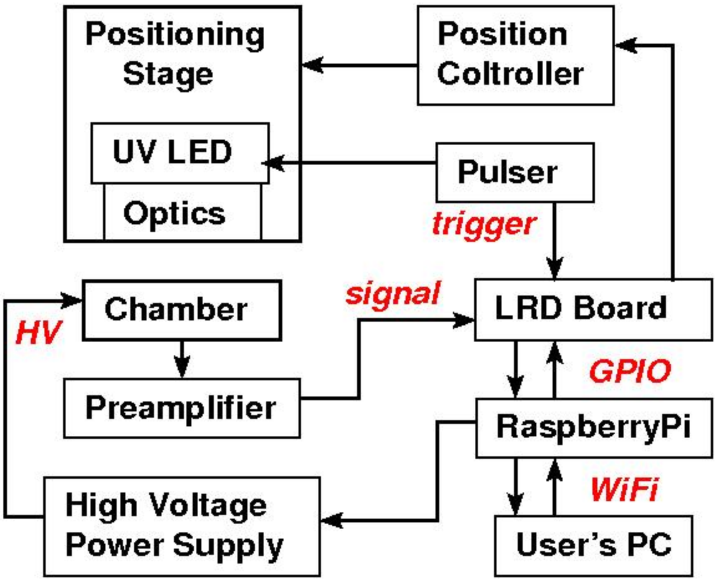


Figure 4: Schematics of a fully equipped Leopard setup.

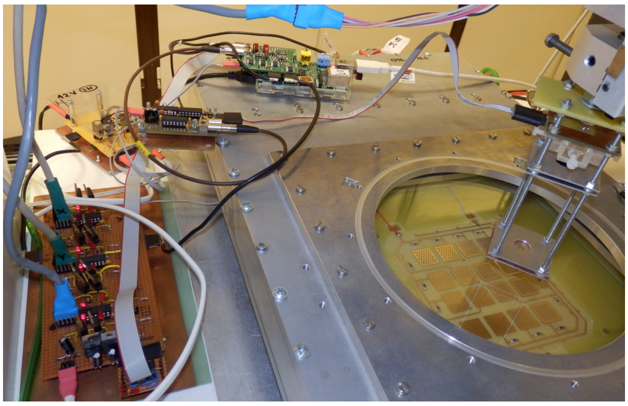


Figure 5: Photo of a Leopard setup at Budapest in 2014. May.

5 LRD Software

The Leopard RPi DAQ Software is written in *C/C++*, to ensure high speed running, modularity, compatibility, and programmer-friendly upgrade possilibity. The DAQ Software runs on the RPi, and encorporates all the recently possible digital communication of the Leopard setup.

5.1 Setting up the RaspberryPi

First of all the operating system of the Raspberry Pi should be settled. Instructions for setting up the SD card with a Raspbian OS can be found in [2] or at [**?**]. Tests and software verifications were done with *Raspbian-Wheezy-2013-09-25*, however all Raspbian should be suitable as well.

The default user for the Raspbian system is "pi" with the password "raspberry", therefore this manual will refer to these as defaults.

The RaspberryPi will be used as a distant machine, therefore the proper net connection should be set up; and there are several possibilies. The RPi can be used with LAN cables, however this is not recommended due to the introduced ground loops, which will increase the noise level. However the RPi does not have built-in WiFi, a simple USB WiFi adapter could used. The WiFi should be configured to connec automatically to the choosen local network. This is the most recommended usage. (Instructions for this can be found for example in [**?**].) The RPi can work as a WiFi hotspot as well, this is only preferred if there are no local wireless network options.

From now on the IP address of the RPi will be named RPIIP.
Please check that the RPi is reachable and usable via logging in.
ssh pi@RPIIP ls

Note, as the usage of the GPIO pins need root permissions, most of the running commands will start with the *sudo* command. The user who will run the LRD should have sudo rights, for the dafault user "pi" this is granted by default.

5.2 Installation of the LRD Software

Copy the ”LeopardRpiDaq” directory to the home of the used RaspberryPi user.

```
scp -r LeopardRpiDaq/ pi@RPIIP:.
```

To achieve maximal compatibility with the current Raspbian used, one should recompile the LRD Software on the actual RPi; this can be done via the compiling script.

```
ssh pi@RPIIP
cd LeopardRpiDaq
source CompileLrd
```

The first time before the LRD Software will be run the inner resistors of the GPIO pins should be reseted.

```
sudo ./ResetGpioPullResistors
```

After this the software is ready to use.

5.3 Starting a run

The Leopard RPi DAQ Software is a command line driven software. One should start it on the RPi from its directory:

```
cd LeopardRpiDaq
```

LRD has some necessary files and several switches to set parameters for the scans. In general the syntax is:

```
sudo ./LeopardDaq [switch [parameter]]
```

The collected data will be written into several output files. The names of the files consist of the run-number and the datatype specifier as an extension.

The detailed list and description of the available switches can be found in section 5.4, and the corresponding input file formats in section 5.5. The structure of the data output files can be found in section 5.6.

5.4 Description of the switches

The Leopard is moderately complex system, however can be used for simple measurements as well. To exploit the wide range of possibilities a system of switches have been introduced. These can to be used to choose actuator movement sequence, do high voltage scans, or charge-up studies with repeated measurements. Most of the functions can be evoked in a way to streightforward setting of some controlled parameters (eg: position, or HV settings).

Most of the switches have a short and a longer name with equal response. And most of the switches need an additional input like filename or simple numbers; these are written in capital letters in the following list.

The basic element of the data recording is the "Measurement Point" (MP), where the collected pulse-height distibution of the many recorded events are stored.

The list of switches for the Leopard RPi DAQ Software are the following:

- -h, --help
  Shows the help, where these switches are listed with their short descriptions.

- -n, --stat NUMBER
  Sets the number of events to be recorded in each MP.

- --prevalence NUMBER
  Sets the prevalence for writing infos to the screen, thus after every NUMBER of MP it gived feedback where the data process is.

- -s, --single
  No actuator movement will be performed. Records spectra only at the current position.

- -sl, --scan-list FILENAME
  Use the specified file for the scan position list. The scan list file should contain the integer triplets for the X,Y,Z coordinates in motorstep units for all the points in interest with the order of desire. Recommended extension for "Leopard scan list" files are ".lsl".

- -sp, --scan-program FILENAME
  To use a standard position scan program defined by the specified file. The scan program should have three lines (for X,Y, and Z), where the Start,Stop, and StepSize have to be defined in motorstep units. Recommended extension for "Leopard scan program" files are ".lsp".

- -hl, --hv-list FILENAME
  Do scans with all the high voltage settings specified in the file. Format deails are specified in the 5.7 section.

- -hp, --hv-program FILENAME
  Do standard high voltage scan on one channel with the range and step specified in the file. Format deails are specified in the 5.7 section.

- --set-hv FILENAME
  Sets the high voltage channels to the first row of the specified hv-list file.

- -r, --repeat NUMBER
  Repeats the specified measurement series NUMBER times.

- -m, --move XX YY ZZ
  Moves the actuator system to position XX,YY,ZZ, and quits. No data taking will be done!

- -f, --force
  Starts the measurement and overwrites the files without question. Useful for starting in the background or with nohup.

- -v, --verbose NUMBER
  Sets the verbose level to write additional information to the screen. Eg. verbose mode 1 shows the calculated yield and gain for each recorded spectrum online.

5.5 Input files

There are three initialization files the LoepardDaq uses. The existance of these files are mandatory, however their content can be changed any time respecting the need of the usage of the system.

- RunNumber.ini
  This file stores the run number, contains only one positive integer number. It is increased automatically for any successful start of the LeopardDaq. In case it is needed, it can be changed simply by overwriting it, like:
  echo 1025 > RunNumber.ini

- XyzPosition.ini
  This file stores the last known position of the 3D actuator system via three positive integer numbers. This info is always updated when the run is finished or stopped. In case of wrong alignment this should overwritten with the proper coordinates, like:
  echo 2000 5000 1000 > XyzPosition.ini

- Settings.ini
  This file stores the hardware settings and comments to the next run, and will be added to the settings file of the run (as an offline logbook for later use, see section 5.6). It should be edited/updated by the user every time before the start of the run, if needed. Any kind of text editor can be used, like:
  vim Settings.ini

For the input file format called by the switches see sections 5.4 and **??** and for examples see section 5.8.

5.6 Data format

For each run there will be at least two output files: one for the settings and one for the recorded data. In case of special runs additional monitoring files are generated as well (eg. map of photo-electron-yield map for the 2D scans).

All these files are named via the runnumber and have specific extension, and are saved in the directory " /Measurements/New/" as "RunRUNNUMBER.EXTENSION".

The settings file (.sett) contains the starting date and time of the run, the run number, all the effects of the applied switches, and the previously edited SettFile.ini (see section 5.5).

The data file contains information on time, posotion, HV settings, and the recorded spectra. All of them has a short name and a format:

- Measurement Point number
  Simple counter for the MPs, increased by one for each MP. Can be efficiently used for offline crosschecks and data integrity.

- Time information (t)
  Format: "t" TimeDiffMicroSec
  Counts the elapsed time in microseconds between this and the previous MP. Sum of these gives the absolute time coordinate of the MP from the start of the run.

- Position information (xyz)
  Format: "xyz" X Y Z
  The three integer numbers are showing the actual position of the 3D moving table in motor steps (2.5 μm).

- HV settings (hv)
  Format: "hv" NHvChannels [HvValue0] [HvValue1] [HvValue2] ...
  Shows the number of HV channels in control, and all the controlled values.

- Spectrum (h)
  Format: "h" Start Support Value0 Value1 Value2 ...
  The histogram of the recorded spectrum in the MP, measured in ADC units. The histogram starts from *Start* and has *Support* number of values: Value0, Value1, ...

5.7 High voltage control

Using the USB ports of the RaspberryPi the LRD is capable to incorporate the control of other external devices. Control of CAEN HV modules are implemented, fot other applications see details in the Developer's Guide.

The LeopardRpiDaq software is capable of communication with the high voltage power supplies using the protocol **CAEN N147X**. (For an up-to-date list check the webpage [?], currently these are CAEN N1470, N1471, N1471H, NDT1470. Tests and verification has been done only with two modules of N1470.)

The communication is done via USB, connection shoud be made directly between the RaspberryPi USB connector and the HV module. The HV module should be set to baude rate of 9600, and remote control (check the datasheet for the used model, or [?]). (Note: after switching a CAEN module to remote control, it have to be rebooted.) These HV modues can be put into daisy chain **??** and controlled via only one USB. The current verion of LRD does not allow to use more USB ports, however are not limited by other than the daisy chain property of the protocoll (upto 16 modules).

Before the first use of the remote control, the communication channel should be configured for the requirements of the HV module via the script:
sudo source DoHvcTtySettings.sh

The format of the HV input files for the switches (see section 5.4) are the following.

For the HV scan on one channel the board and channel number should be given for the proper HV feed, and the scan should be specified with start, end and step values. The file should look like this:

HV *BOARDNUMBER CHANNELNUMBER*
HvProgram *START END STEP*
End

For the HV settings list the file should contain board and channel information for all the HV feeds to be controlled, and the list of the used values, like this:

HV *NUMBER_OF_HV_CHANNELS*
HV1_BOARD HV1_CHANNEL HV2_BOARD HV2_CHANNEL ...
HvSettings *NUMBER_OF_ROWS_IN_THE_LIST*
HV1_VALUE1 HV2_VALUE1 ...
HV1_VALUE2 HV2_VALUE2 ...
...
End

5.8 Examples

To record a single spectrum at the current position with 1M statistics:

```
sudo ./LeopardDaq -s -n 1000000
```

To check charging up with taking 100k statistics for 200 times on a single spot:

```
sudo ./LeopardDaq -s -n 100000 -r 200
```

To do a CriticalLineScan, so doing a one line position scan with a high voltage scan on the drift with 100k statistics you call:

```
sudo ./LeopardDaq -sp Cls.lsp -hp Cls.lhp -n 100000
```

where the files can be eg.: Cls.lsp:

```
4500 4500 1000
8000 9000 20
2500 2500 1000
```

And the HV scan file, Cls.lhp:

```
HV 3 3
HvProgram 2000 3000 100
End
```

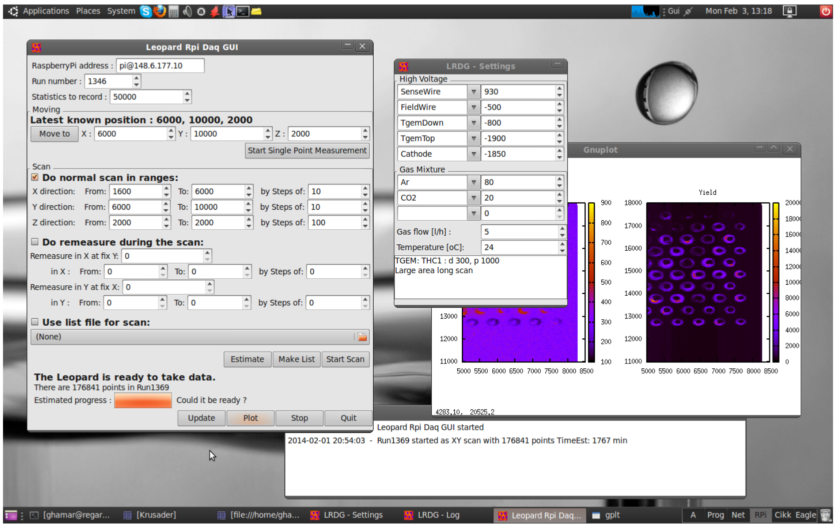
This will record 51 points along Y, with the specified 11 voltages, thus resulting 561 MP spectra.

6 LRDG, the graphical user interface

This section is about the Leopard RaspberryPi DAQ Graphical User Interface (LRDG).

6.1 Installation

6.2 LRDG Panels



Leopard Rpi Daq GUI

RaspberryPi address : pi@148.6.177.10

Run number : 1346

Statistics to record : 50000

Moving

Latest known position : 6000, 10000, 2000

X : 6000 Y : 10000 Z : 2000

Scan

☑ **Do normal scan in ranges:**

X direction:	From: 1600	To: 6000	by Steps of: 10
Y direction:	From: 6000	To: 10000	by Steps of: 10
Z direction:	From: 2000	To: 2000	by Steps of: 100

☐ **Do remeasure during the scan:**

Remeasure in X at fix Y: 0

in X : From: 0 To: 0 by Steps of: 0

Remeasure in Y at fix X: 0

in Y : From: 0 To: 0 by Steps of: 0

☐ **Use list file for scan:**

(None)

The Leopard is ready to take data.
There are 176841 points in Run1369
Estimated progress : Could it be ready ?

LRDG - Settings

High Voltage

SenseWire	930
FieldWire	-500
TgemDown	-800
TgemTop	-1900
Cathode	-1850

Gas Mixture

Ar	80
CO2	20
	0

Gas flow [l/h] : 5

Temperature [oC]: 24

TGEM: THC1 : d 300, p 1000
Large area long scan

```
Leopard Rpi Daq GUI started
2014-02-01 20:54:03 - Run1369 started as XY scan with 176841 points TimeEst: 1767 min
```

Figure 6: LrdgScreenshot

Part II

DEVELOPER'S GUIDE

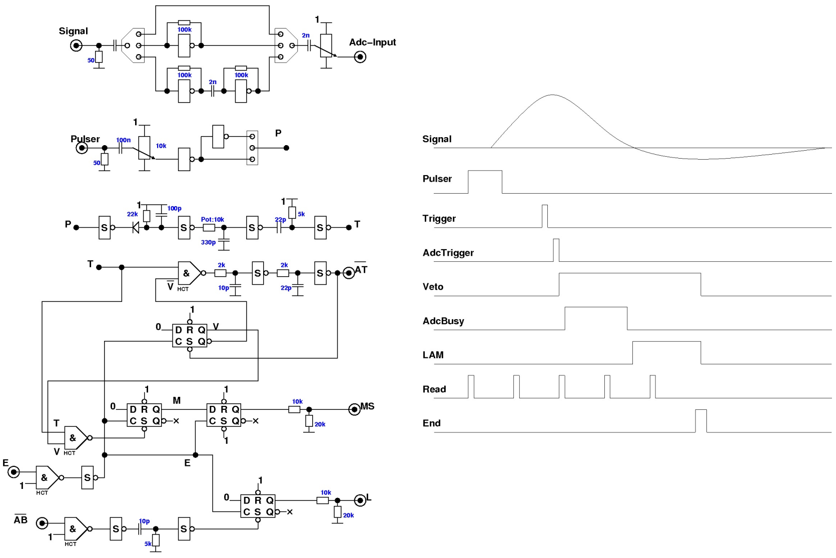


Figure 7: LrdLogicsScheme

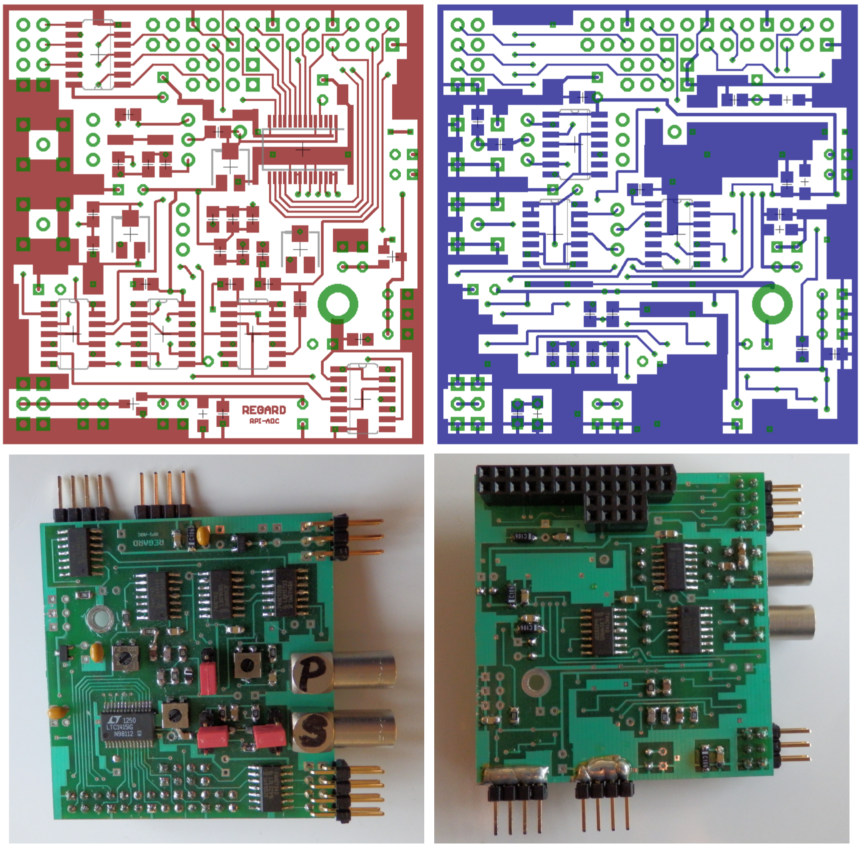


Figure 8: The Eagle schematics and the photo of the top and bottom side of the LRD Board.

.

References

[1] Raspberry Pi webpage
www.raspberrypi.org

[2] Raspbian Operating System
www.raspbian.org

[3] G.Hamar, D.Varga
High Resolution Surface Scanning of Thick-GEM for Single Photo-Electron Detection
Nucl.Instrum.Meth. A 694 (2012) 16

[4] G.Hamar, D.Varga
High granularity scanner for MPGD based photon detectors
Proceedings of Science TIPP2014 (2014) 056

[5] A.László et al.
Experimental determination of GEM avalanche multiplicity distribution
Presentation on the RD51 Collaboration Meeting (2014) Kolkata

[6] G.Kiss et al.
Single electron avalanches in Gas Electron Multipliers
Presentaion on the Zimányi Workshop (2014) Budapest

[7] List of usable SD card for the RaspberryPi
http://elinux.org/RPi_SD_cards

[8] Linear Technologies LTC1415 ADC
http://www.linear.com/product/LTC1415
http://www.linear.com/docs/3720