Table of Contents

Module Assembly and Quality Assurance tests

Access and Contacts

Grey Room in LHCB Hall (ask for key's LHCb secretariat)

It has airconditioning/heating and overpressure (can be switched off during work, but ON during night). The shoes protection is obligatory in the Grey Room. The dust and dirty work is not allowed in the room

To operate one would need:

- valid CERN access to the LHCb P8 surface building, control room, and Assembly hall
- lxplus account and LHCb ELOG registration
- Access to SciFi ProdDB

Contact persons for operation:

Valery Zhukov	joukov@cernNOSPAMPLEASE.ch	Office: +41 (0)22 76 78188	Cell(CERN): +41 75 411 6537
Sebastian Bachmann	sebastian.bachmann@cernNOSPAMPLEASE.ch	Office:	Cell: +48 791 221 186

QA cosmic test operators:

Andreas Guth	andreas.guth@cernNOSPAMPLEASE.ch	Office: +41 (0)22 76 75750	
Alexander Malinin	malinin@cernNOSPAMPLEASE.ch	Office:	Cell(CERN): +41 75 4110546
Alexander Chuzo	alexander.chuzo@cernNOSPAMPLEASE.ch		
Alexander Grigoryev	alexander.grigoryev@cernNOSPAMPLEASE.ch		
Alexander Petrov	aleksandr.petrov@cernNOSPAMPLEASE.ch		
Alexander Dolmatov	aleksandr.dolmatov@cernNOSPAMPLEASE.ch		

Module Assembly:

Antonio Pelligrino	antonio@nikhefNOSPAMPLEASE.nl
Lara Veldt	lveldt@nikhefNOSPAMPLEASE.nl
Robin Schmeitz	robins@nikefNOSPAMPLEASE.nl
Espen	

Grey Room occupation

Period	Activity	Contacts	Comments
1.07.2020-1.08.2020	SciFi module assembly and QA	V.Zhukov, R.Schmeitz	existing benches
-	Velo	F. Sanders	~10m2
-	RICH	M.Booth	
-	UT		

15.09-10.12	SciFi		

Info and Links

SciFi upgrade twiki

Elog SciFi:Module

SciFi indico

Summary moduleQA

Talks

SciFi General:

03.12.2018 SciFi General M.Dziewiecki

31.012019 SciFIiGeneral V.Zhukov

04.03.2019 SciFi General S.Bachmann

11.04.2019 SciFi General A.Guth

16.05.2019 SciFi General M.Dziewiecki

17.06.2019 SciFi General V.Zhukov

25.07.2017 SciFi General Meeting S.Bachmann

02.12.2019 SciFi General M.Dziewiecki

Module Production:

* pdf_15052019.zip: Valery15052019</verbatim>

* bads07072019.pdf: Valery08072019</verbatim>

* slides_module.pdf: Andreas22072019</verbatim>

Current Status

SciFi Status uptodate

Module Assembly at CERN

The module produced from Fiber Mats in HD and NIKHEF (here MOD) are shiped to CERN and finally assembled in the Grey Room. The Cold Box (CB) Shell is connected and sealed, then the SiPM Top Cover(BTC) is attached. The assembled with cold box modue (MODCB) is checked with a few gas tightness tests and then passed to the QA tests in cosmic bench. After QA tests the MODCB are packed in the boxes and transported to the Assembly Hall for the Cframe integration.

The assembly procedure steps can be seen in the assembly checklist :
https://twiki.cern.ch/twiki/pub/LHCb/SciFiFibreModulesQA/Modassembly_v4.pdf</verbatim>

For more information contact Antonio, Lara and Robin.

CERN Cosmic Bench

The cosmic bench is used to test SciFi modules in pairs, such that one can select a track and estimate the angle. This data are used for the QA.

One can also use only one (usually upper) position, similar to the modtest in the modproduction center for the debugging purpose.

Setup

Two main components:

I. The movable rack with:

1. two SPIROC boards for upper and down positions connected to the corresponding USB boxes by ribbons
2. PC(ask for account lhcb-cern): `lhcbscifi.dyndns.cern.ch`
3. two USB boxes:

Up: board37 uplinks:10,11,12,13,14,15,16,17 and Down: board36 uplinks: 20,21,22,23,24,25,26,27

1. two corresponding LV power supply (PS)
2. Keithley power supply for HV, connected to PC
3. HV switch box, with connection to a multimeter (for LT test)

II. Cosmic bench with holders for two standard (5m) modules and Cosmic hodoscope in between (placed on top of the bottom module):

1. plate with 3 scintillators 80x20cm (from AMS01) providing 60x80cm total area, equipped with 2 Hamamtsu RS5900 PM each (HVmax=800V) running in coincidence.
2. NIM crate with: HV, amplifiers, discriminators, logics and the counter. The typical cosmic rate is about 140Hz.

DAQ

Based on the USB box and SPIROC motherboard, see attachments for schematics and connections. Be careful with connections, watch the channels and uplink numerations:

- the module SiPM channels: 0 from the right (look from readout side)
- the LIS channel: 0 is the right, 1 left
- uplink ribbon cables, uplink 10(20) is the right (look on the USB box connectors) The ribbon are made such that the 4 cables are connected up and another 4 down (for each box)

There HV line is connected to the SPIROC boards via sophisticated switch box. It allows to measure HV current precisely for each module. The switches(4) on the SPIROC board allows to disconnect(switch up) HV from each mat SiPMs.

Software

The software is based on the scifiusbboard package used for SPIROC readout via USB, https://gitlab.cern.ch/lhcb-scifi/scifiusbboard The installation requires a special versions of Linux and root, so cant be run on standard PC.

To compile can try run `make all` in the scifiusbboard folder using existing Makefile. Note that if some new files are added, need to rerun qmake after modification of pro file.

One can use `strosmic90` with one or two modules, eg: `strosmic90 halfmodule` or `strosmic90 module` used fortesing in production centers, for some tests.

The modified version to be used at CERN for QA is in HD2CERN `scifiusbboard/utils/modqacern` to run it, type in the appropriate folder: `modqacern module`

It include several features:

- write a tree with events where we have exactly one hit in the upper and down modules
- write a summary files after each run (30min by default)
- have some extra histos in the summary
- some modifications in LIS gain calculations, etc...

Similar to old code in the GUI one can add prefix to all raw files with uplink data, and summary files. Type `start` to start running. After starting the run, it runs 1000 events for the pedestal and then 8000 events for the LIS (takes ~1min), after that the related histograms are appearing in GUI and can be inspected. The raw file (prefix_timestamp.root) is created, and the root tree with selected events with two hits in two planes dualtree_time stamp is filled. When run is finished (30min) another two files are created; summary file A_Summary+prefix+time stamp, and the equivalent Summary_last.root file, the last is always containing the last summary. The created Summary files can be inspected by root browser during the run. The dualtree can be also seen by root, but should be copied first (to avoid locking when program is filling data). If you copy dualtree, dont use dualtree name in the copied files because all files with dualtree names are used further on for the analysis. The final summary file with total statistics should be also saved by taping on `save` after `stop`

One can reprocess raw data by running `modqacern  rawfilenames.root` (in case you miss some summary file) In this case program will recognise the format. To save summary, should type `save` as usual. Note that the summary root files contains two types of histogramms; histo filled with current run, and the accumulated histograms with the whole statistics (*AllRuns*).

The input settings for the `modqacern` is read out from the `usbboard*.cfg` files stored locally (contrary to the strosmic). This cfg files contain information about uplinks and, important, the LIS settings, such as Ibias and Imod. Note that after `lis_tune` the tuned settings are saved not in the local cfg files but in the `scfiusbboard/cfgFiles/` and must be copied to the running folder using `./copy_cfgfiles` script, or manually.

The templates for the scripts used for the running and analysis are in `scifiusbboard/cern_scripts/`. The operation and analysis codes are made as root macros that are called from the scripts created by the `start` script taking into account the locations and configs. The `start` script creates the scripts that can be run from the corresponding folder. It also creates the config files containing all relations. The `cern_scripts` includes:

- `lis_tuning` that make the Imod scan trying to reach target Ly for the LIS using `tune_lis` or `tune_lis_advanced` Also there is a Imod scan code (optional, not in the templates) that just scan Imod.
- `cosmic` that analysing all data called from the `sum_QA` script. The `anadual` process the dualtree with the selected hits (one hit per plane) that allows to calculate cluster size for vertical track, and the

`sumanalqa` that includes the results of the `anadual`, `Summary` files produced in QA and module tests, as well as lis, LT tests and Vbd files, and analzing for bad channels, etc. * `LT` just a query to fill a file with light tightness test
- `vdb` have a script to extract Vbd from SiPM Db for each BTC and store it as a root histo. This part currently can't run on the lhcbscifi PC, that is, the filesare produced elswhere and copied to `/home/lhcb-cern/moduleTest/data/sipm`. Then used by `dac_from_db` script and `sum_QA`
- `ana` example of analysis of sum root files

Computing

The `lhcbscifi` PC is connected to cern network via wifi. To connect externally use `lhcb-cern@lhcbscifi.dyndns.cern.ch`. Outside CERN one have to connect to lxplus first.

The PC have 3 SSD disks (mounted as raid): main working disk 250 Gb, storage 2Tb and archive disk 14Tb. There software is in the `sw` and testing in the `moduleTest` where `data` contain results from MOD tests i HD and NIKHEF, and SiPM testing used to extarct Vbd, while `cern_data` is used for the testing in Cosmic Bench.

The archive data are at `larchive` (14Tb) and `ldata` (2Tb). The `larchive` has two main folders `FAM_old` contain modules tested before 17.05.2019 and `FAM_new` tested after. (Some modules were reassembled and retested with modified ColdBox)

The templates for the scripts used for the running and analysis are in `scifiusbboard/cern_scripts/`

production Db

One would need an account to access it Db There is a common account, ask S,Bachmann. One can access it in lhcbscifi computer by runing `dblogin` . The final modules after QA are in the `Final Module Assemblys/F-Modules` the results are at the `QA on F-modules` The modification (eg. LIS board or BTC) have to be registered in the Db, click Modify, Change things, and reload the modified page. Then use the `restart` to recreate the configs in the same folder, or start from scratch.

QA test procedure

All operation with hardware: moving modules, connections, etc should be performed only by qualified personals (see contact persons for operations) Most of operations on software are done with scripts, but operator should know the procedure behind.

Never switch ON SiPM HV WITHOUT LV PS ON!

The standard procedure is following, more detail will follow:

1. Install. Place two modules on the upper and down cosmic bench positions, and cosmic hodoscope in between. Check the assembly checklists (attached) and prepare a new ModuleQA checklists for each module.

Ensure position of the trigger hodoscope

1. Initialize. Create directory structure, by running script `./start` This will create folder with FAM[uppermodulId]_FAM[downmoduleId] and put there `config.txt, config.root` files and all needed scripts. Ensure that all information is correctly downloaded from the Db by checking `config.txt`. Go to the created folder.

2. Open ELOG. Register the modules to be tested in the System->Module. You need to have an account, and the link should be in the browser. Write all related information about the modules(new, retested, BTC modification,etc) setup(if changed) and all problems during the test.
3. Set Spiroc DACs (offsets to SiPM HV). Use Vbreakdown from Db for each SiPM. For this power LV PS for all USB boxes and run `./dac_from_db`. Check the error messages if appeared (should not). Check content of `vbd/` folder.
4. Perform connectivity test. Use 'old' LIS settings, by first coping existing board36/37 cfg `./copy_cfgfiles` , then go to `test/` folder and run `modqacern module`.

Put meanfull prefix, start run, check Pedestal, LIS, and some cosmics, then stop the run and save summary.

1. Tune LIS. Use Ibias from Db to get optimised Imod by running `./tune_lis`. This takes about 30min and produce cfg file in `sw/` folder and txt files in `lis/` folder.=
2. Perform another test with tuned LIS settings, after =./copy_cfgfiles
3. , go to `test/` folder and run, with prefix afterlis.
4. Perform LighTightness tests (LT). Fill data by running `./check_LT`, check content of `LT/` folder.
5. Run cosmic QA. Usually for >= 6 hours through the night. For this go to `QA/` folder and run `modqacern module` setting QA prefix for files.=
6. Post-process data . After QA is finished, run
7. ./sum_QA= that creates root and txt files in the `sum/` folder, check the files.
8. Upload to Db. Put data into the database by running `./upload_db=`
9. Archive data. Move =FAM_FAM
10. folder to the archive in order to save space, each test usually takes about 40Gb.
11. Dismount. Disconnect modules and store it either on the bench or in the transportation box. Put the QA tests checklists together with assembly ones in the folder attached to the modules.
12. If another side has to be tested, disconnect power plug and ground for the moving rack, and carefully move the rack to other side fixing the Spiroc boards hanging on the rails. Dont need to switch power on PC, you have about 20mins.

After installing, dont forget to move trigger hodoscope to the right place. You can store the trigger hodoscope at the bottom rails of the cosmic bench.

All steps has to be registered in the checklist
https://twiki.cern.ch/twiki/pub/LHCb/SciFiFibreModulesQA/Modtest_v4.pdf: Modtest_v4.pdf</verbatim>
and some important information in the ELOG, see previous entries for examples.

Installation

INITIAL STATE:both USB-Box LV PS OFF, HV PS OFF

1. Disconnect tested modules; LIS and SPIROC using a grabbing tool. Position the SPIROC boards on the rack safely.
2. Remove previous modules from the bench. Move hodoscope beneath the modules on the shell, carefully (with HV on)
3. if needed move the rack to the side for tests. For this disconnect power (without switching off PC) and ground pin. Keep the trigger cables connected(going to hodoscope) and carefully move the rack, holding the SPIROC boards.
4. Install a new module on down position, ensure module position and flatness. Put hodoscope in between to the right place (close to the center) using stoppers on the rails.
5. Ground yourself using braclet, touching the metallic parts.
6. Mount SPIROC readout boards to the module using two M8 screws (for upper and down modules)
7. Connect SiPM flexes. Handle them with care. Ensure that all connectors are fully pushed in. Recommended to clean up the SPIROC motherboard connectors by isopropanol.
8. Connect LIS flex cables.

9. Switch ON both LV PS then HV PS (58.0V) only after LV is ON. Check total current (should be ~0.012A, but higher during ramping)
10. Switch HV OFF

Fill the check list for both modules.

Initializing directory structure for measurements

There's a script for creating a directory structure (and some further helpful scripts) for each new measurement session (one side of a module pair).

1. Open terminal. Go to `moduleTest/cern_data`
2. Type `./start`
3. Answer all questions given by the script
4. If there are problems with accessing database, probably you're not logged in. Break the script and type `./dblogin` If this not working, problem with Db access. One can still work offline(i.e answering NO on Db access during the scripts), but is not recommended, because one need to know all relations of objects. However it might be needed if entries in Db are not correct (eg. wrong BTC, or Ibias in LIS, etc), though it is recommended to correct them

Following folders are created:

- `.../vbd/` - a directory for all breakdown-related stuff `.../vbd/mod_up/` for upper module `.../vbd/mod_dn/` for down module

- `.../lis/` - a directory for LIS tuning results
- `.../QA/` - a directory for cosmic data and results
- `.../test/` - a directory for connectivity and other test cosmic runs
- `.../LT/` -for LightT ightness test
- `.../sum/` - a directory for cosmic data analysis results

Created Config files: `config.txt` is a config file with entered module data, including obtained from Db, and `.config.root` the same, but as root file, this one is used for all scripts, i.e. doesnt make sense to modify the txt file only.

Scripts:

- `tune_lis` - a script for running LIS tuning procedure
- `tune_lis_advanced` - the same, but allowing for modifying of bias currents. For experts.
- `dacfiles_from_db` - copy DAC files from local database to working directory. For experts.
- `dac_from_db` - run `dacfiles_from_db` and then set Spiroc DACs
- `copy_cfgfiles` - copy module config files from USBBOARDPATH to *QA* and *test* directories
- `run_QA` - run modqacern from `QA` directory
- `test_QA` - run modqacern from `test` directory. For experts.
- `sum_QA` - post-process QA data
- `check_LT` - query light tightness result and write them to a file
- `db_login` - a helper for logging into the production database (produces Michal's name on output as a side effect)
- `update_db` - upload all possible data into the production database.
- `copy_results` - copy most important files to the `data/FAM` directory. For experts.
- `restart` - re-create directory structure, config files and/or local scripts. For experts.

More on scripts : Close

Attention: backup script is a part of run scripts. In most cases, running some scripts twice would delete old data. To avoid it, a backup script can be used. To backup a dir, simply go into it and type `backup_me`. This will create a new folder with a general name of `BACKUP_yyyymmdd_hhmmss` and copy or move all files there. The script is interactive and will ask for a short comment (then saved to a `backup_info` file) and whether to delete old data. It can be used anywhere (even outside the created directory structure).

```
#The name of this subdirectory is always BACKUP_yyyymmdd_hhmmss.
#Additionally, a text file in this directory (backup_info) will be created with a user-given info
#All files specified in .backup_ignore will be SKIPPED when archiving.
#All files specified in .backup_blind will be archived, but not relevant for the initial decision
#Currently, only files from top-level directory can be specified, i.e. any subdirectory is treate
#Both files use regular expressions.
```

The following scripts make backups automatically if target directories are not empty:

- `tune_lis=`
- `tune_lis_advanced`
- =run_QA
-

Downloading Vdb info and setting Spiroc DACs

INITIAL STATE:both USB-Box LV PS OFF, HV PS OFF

Note that this procedure is needed once at the beginning of all tests with new SiPM (re)connected to the SPIROC.

The procedure assumes that all information from needed BTC is already downloaded into the `moduleTests/data/sipm` and correctly formatted. If this particular BTC is not in the Db, the Vbd from Db can not be used. SO the test can not be done! You will get error message during start script. Please contact experts (VZ,AG) or produce the files containing the DAC values with the procedure described below.

Automated procedure for using Db:

1. Ensure that HV is OFF, then set both LV PS ON
2. Run `./dac_from_db` It takes a few minutes, check for errors on the screen.

Manual procedure: (if needed):

1. Copy DAC files: type `./dacfiles_from_db` Currently the files are manually downloaded from Db and stored in `moduleTest/data/sipm`. If the corresponding BTC doesnt exist, you are in trouble, contact AG.
2. Ensure that HV is OFF
3. Switch upper PS ON while , lower PS OFF
4. Type `setSpirocDac $USBBOARDPATH/cfgFiles ./vbd/mod_up/dacfile_from_db.txt` to program SPIROC chips for upper system
5. Switch lower PS ON, while upper PS is OFF.
6. Type `setSpirocDac $USBBOARDPATH/cfgFiles ./vbd/mod_dn/dacfile_from_db.txt` to program SPIROC chips for lower system
7. Switch both PS OFF

Fill the check list.

The alternative procedure to have correction for SPIROC DAC is to run `gainAnalysis =  from =scifiusbboard/utils` using the LIS signals. The procedure will follow.

Get files with DAC values from Vbd values stored in the DB

In case the input DAC files in moduleTests/data/sipm are missing for a new BTC, these files can be produced (along with the Vbd root files) using the .py script `calc_all_DACFiles_DB.py` attached to this TWiki as txt file. The script uses pyROOT to handle the Vbd input from the DB and to produce the Vbd histograms that are stored in moduleTests/data/sipm . Since the ROOT installation on the cosmics bench PC does not include pyROOT, please run the script on lxplus and copy the output folders to moduleTests/data/sipm on the cosmics bench PC. Use the following instructions :

1) Log in to lxplus.

2) Setup root, for example via `source /cvmfs/sft.cern.ch/lcg/app/releases/ROOT/6.12.06/x86_64-centos7-gcc48-opt/root/bin/thisroot.sh`

3) In order to get the Vbd input from the production DB, install the DB command line tool as described in: https://scifi.physi.uni-heidelberg.de/index.php/topic,18.msg45/topicseen.html

For this copy the `proddb-0.1,tar.bz2` from web, also attached here. Then untar it, doing `bzip2 -dk proddb-0.2.tar` and `tar -xvf proddb-0.2.tar`. In the directory proddb-0.2/ you find a README, but on lxplus the installation should work without installing further packages, i.e. just do `make`

4) In the directory proddb-0.2/ , run the command `./proddb LIST` and then enter your DB account name and password

5) Set the environmental variable PRODDBDIR to the full path of proddb-0.2/, eg export PRODDBDIR=you pwd

6) Copy calc_all_DACFiles_DB.py attached here and run `python calc_all_DACFiles_DB.py` . This will produce the DAC files and the root files for all BTCs and save them in folders in your cwd (already with the structure used in moduleTests/data/sipm). You can specify the foldernames by editing 'dir_rootFiles', 'path_DACFiles' in the script. It takes about more than 1 hour and about 100 Mb. The script can crash in case of incomplete Db entries. (have to be clarified).

There is no possibility yet to extract a particular BTC, so all available will be extracted.

In case of problems with the installation, please contact AG.

DAC from

Have to run `hrdreadoutForGainCalibration` (copy it from `scifiusbboard/Builds/` and can modify setting inside) for each USB box separately (enter 36 or 37, and power it). The code will scan HV and estimate HV for each value iteratively. It will create the `itx/` folder with iteration, you can check inside if the iterations are converged (by lokking on rms), Then select iteration there and upload to DAC as `setSpirocDac $USBBOARDPATH/cfgFiles ./itx/DacFile*`

Connectivity test

INITIAL STATE:both USB-Box LV PS OFF, HV PS OFF

1. Check tumbler positions on the switch box (bypass is UP, others are full DOWN). Check positions of tumblers on SPIROC boards (all DOWN, by default)
2. Switch LV PS ON.
3. Switch HV ON (58V)
4. Use `./copy_cfgfiles` script to copy `board36.cfg` and `board37.cfg` to working folders (`QA/` and `test/`).

5. Go to `test/` folder and run `modqacern module` Set the prefix for stored files, eg. firsttest, afterlistune, etc.... and start the run. After LIS run, and first cosmics check the histos and save summary. Check the summary file is stored.
6. close the GUI
7. HV is OFF

Fill the check list.

Light Tightness Test

INITIAL STATE:both USB-Box LV PS ON, HV PS OFF

The test can be done before or after QA.

1. Connect multimeter (uA input) to banana cables from HV distribution box.
2. Switch box: all switches DOWN, i.e bypass and both channels.
3. SPIROC boards: all switches DOWN (should be by default)
4. both LV PS are ON
5. Start a run with `test_QA` script to initialise SPIROC chips. This can be skipped if `modqacern` already have run in cosmic or connectivity test.

Use the bright light source and illuminate the area new Cold Box all around, and the module itself especially in the centre and on the side walls. The test is done separately for the Top and Bottom modules.

TOP module:

1. Switch box: Ch1 DOWN, Ch2 in CENTRAL(OFF) position
2. LV is ON, Switch on HV PS (58.0V)
3. Wait until readout stabilises ~5min, then write currents from multimeter to checklist.
4. Switch multimeter to relative mode
5. Switch on bright lamp and illuminate different parts of coldbox and connections to the module, write down the highest current increase (dI)
   < 5 uA: OK
   5-15 uA: Not critical, but try to locate the leak and leave a comment
   > 15uA: locate and repair the leak, leave a comment.
6. Switch lamp OFF
7. Switch HV OFF

BOTTOM module:

1. Switch box: Ch2 DOWN, Ch1 in CENTRAL position,
2. LV is ON, Switch on HV PS (58.0V)
3. Wait until readout stabilises ~5min, then write currents from multimeter to checklist
4. Switch multimeter to relative mode
5. Switch on bright lamp and illuminate different parts of coldbox and connections to the module, write down highest current
6. Switch lamp OFF
7. Switch HV OFF
8. Distribution box: set bypass UP, set both channel all DOWN.

Fill the checklist. Use `check_LT` script for saving data on the PC. Write the ambient temperature (from the clima panel), then the quiescent current (currebt after stabilization, usually about 15uA), and then the maximum deviation (typical <0.5 uA), for the top and bottom modules.

LIS tuning

INITIAL STATE: both USB-Box PS OFF, HV PS OFF

1. Switch both LV ON, then HV ON (V=58V)
2. Open terminal, go to right `moduleTest/cern_data/FAMxxxxxxX_FAMxxxxxxX` directory
3. Type `./tune_lis`
   The configuration will be fetched from a root file generated by initialisation script. You don't need to enter it again.
   Use `tune_lis_advanced` if there's a need to modify bias currents. This is an advanced option, so don't use it without understanding.
4. It takes about 30min to calibrate both modules
5. Text files with report will be generated at the end (`module_name.lis.txt`) and put into `lis` subdirectory. Check the files and fill the check list.
6. Config files will be configured for the last tuned setting and stored in the `$USBBOARDPATH/cfgFiles`.
7. Repeat the test run using new lis tuning parameters. Go to the main folder `./copy_cfgfiles`, go to the `test/` and

run `modqacern module` (put prefix afterlistun for runid), check the LIS spectra and decide to use it or not. (In case of the bad spectra may change Ibias, see FAQ).

Note that the LIS driver are connected in the staggered way to the mats, that is the mats 0 and 2 are connected to one LIS, and 1,3 to another one LIS board. The connection of fibers (o or 2, and 1 or 3) in each LIS boards is not defined during assembly, but determined after the LIS tune procedure, and stored as a mapping in the file.

'Kink position' - checking Ibias

When using `./tune_lis`, a root script `lis_get_kink.c` is called automatically after tuning. It produces another set of text files (`*.lis.kink.txt`) with so called *kink position* calculated. Shortly speaking, it's a linear extrapolation of the two points of LY vs Mod characteristics to zero LY. These two points come from LIS tuning to standard 1.1 and 1.5 p.e.

We believe that the kink position is an indirect measure of the correctness of Ibias setting. Shortly speaking: We expect kink position to be **20-30 units. Lower values** may indicate **too high Ibias** setting. Please consult an expert in this case. Lowering Ibias might be required.

This step was introduced on 19.11.2019. For earlier data, one can call the script manually. It's in `$USBBOARDPATH/cern_scripts/lis_tuning/`

Advanced LIS tuning

This will ask for the Ibias, allowing to change it from the nominal values read from Db. Note that the Ibias and Imod after lis tune are stored in the `/home/lhcb-cern/sw/scifiusbboard/cfgFiles/board*cfg` and then can be copied to the QA and test folders using `./copy_cfgfiles` script. These Ibias and Imod will appear in the `lis/` folder in the results files. Note that the used Ibias are not stored in the SiPM Db section, but as listuning txt files after the QA test.

QA cosmic test

INITIAL STATE: both USB-Box PS OFF, HV PS OFF

1. Use `copy_cfgfiles` script to copy `board36.cfg` and `board37.cfg` to working folders (`QA` and `test`).
2. LV is ON, then HV is ON (58V)

3. Use `run_QA` script for starting DAQ software, it will start `modqacern module` in the `QA/` folder. Set the *prefix* for saved root files and click start, use QA prefix for final runs used in analysis.
4. run for >6 hours, which corresponds to more than 1000000 selected tracks (in dualtree). Each run consists of 3 parts: Pedestal 1000triggers,LIS run 8000triggers,Cosmics 30min.During the run 3 trees are filled in the corresponding root file with the name prefix_unixtime_data_time.root. Also the selected dualtree tree for events with only 1 hit in one module is stored in the dualtree_unixtime.root file. New run is started automatically each 30min, the running can be stopped by clicking `stop`.Then slick `save` to save the Summary file.
5. stop the run, save, check that output files in the `QA/` folders are OK (latest dualtree and Summary)
6. close the GUI, switch HV OFF, and then LV OFF.

Then new run is started automatically. The running can be stopped by clicking `stop`. Then slick `save` to save the Summary file.

At the end of all tests switch HV and then LV off.

Data post-processing

To post-process the data collected in the QA run, simply type `./sum_QA` from the FAM_FAM folder. Attention: post-processing might take a significant amount of time (> few s).

The following files will be generated in the .../sum_QA directory:

- `sum_FAMxxxxx_X.root` (x2) - the main output file with data, for each module
- `sum_FAM_xxxxx_X.pdf` (x2) - a pdf summary of the above
- `BTCxxxxx_Vbd.root` (x2) - a file with breakdown voltages copied from `data/BTC=`
- =mod_FSMxxxxx_X.root
- (x2) - a file with HD/Nikhef QA data for a module, copied from `data/FSM=`
- =dual_FAMxxxxx_X.root
- (x2) - an intermediate file with Cern cosmic data processed
- `modcb_FAMxxxxx_X.root` (x2) - another file of this kind
- `sumbadch_FAM_xxxxx_X.txt` - a text file with a summary of bad channels
- `sumgaps_FAM_xxxxx_X.txt` - a text file with a summary of mat-SiPM gap calculation

The script will also create in `QA/` folder a final `dualtree.root` file with all dualtree merged. If there is some copied, delete it prior running the script. It will be also a link created to the last `Summary.root` file. (in case of modqacern crashes, see FAQ)

The summary root files after processing contains results from the current tests(modcb), as well as from FSM test in module production centers(mod), and SIPM testing.:

sum*.root : Close

```
 TH2D   h2mod_LIS;1
 TH1D   h1mod_LIS_gain;1
 TH1D   h1mod_LIS_pe;1   h1mod_LIS_pe
 TH1D   h1mod_Cosm_prof;1   12826435 entries / 2048 channels = 6262 entries per channel
 TH1D   h1mod_Cosm_sig;1   2048 entries / 2048 channels = 1 entries per channel
 TH1D   h1mod_Cosm_sigcl;1   2048 entries / 2048 channels = 1 entries per channel
 TH1D   h1mod_LISbad;1
 TH1D   h1mod_Cosmsbad;1   2048 entries / 2048 channels = 1 entries per channel
 TH1D   h1mod_Cosmpbad;1   12826435 entries / 2048 channels = 6262 entries per channel
 TH1D   h1mod_allbad;1
 TH1D   h1mod_mask;1
 TH2D   h2modcb_LIS;1
 TH2D   h2modcb_LIS_S;1   LIS Scurve
 TH1D   h1modcb_LIS_gain;1
 TH1D   h1modcb_LIS_pe;1   h1modcb_LIS_pe
```

```
TH1D   h1modcb_LIS_rms;1   LISrms
TH1D   h1modcb_LIS_p0;1   LIS5pe
TH1D   h1modcb_LIS_p5;1   LIS0pe
TH1D   h1modcb_LIS_p0fr;1   hs5pe
TH1D   h1modcb_LIS_p5fr;1   hs0pe
TH1D   h1modcb_Ped_noise;1   proj
TH1D   h1modcb_Cosm_prof;1   11968521 entries / 2048 channels = 5844 entries per channel
TH1D   h1modcb_Cosm_sig;1   2048 entries / 2048 channels = 1 entries per channel
TH1D   h1modcb_Cosm_sigcl;1   2048 entries / 2048 channels = 1 entries per channel
TH1D   h1modcb_Cosm_dnoise;1   Noise Down
TH1D   h1modcb_Cosm_deff;1   Eff Down
TH1D   h1modcbdual_prof;1   h1d_posa
TH1D   h1modcbdual_sig;1   h1d_siga
TH1D   h1modcbdual_sigcl;1   h1d_sigcla
TH1D   h1modcbdual_cls;1   h1d_clsa
TGraphErrors   gmodcbdual_cls0;1
TGraphErrors   gmodcbdual_sig;1
TGraphErrors   gmodcbdual_sigcl;1
TH1D   h1modcb_Pedbad;1   proj
TH1D   h1modcb_Cosmnbad;1   Noise Down
TH1D   h1modcb_Cosmsbad;1   h1d_sigcla
TH1D   h1modcb_Cosmsbadfr;1   h1d_sigcla
TH1D   h1modcb_Cosmpbad;1   h1d_posa
TH1D   h1modcb_LISbad;1
TH1D   h1modcb_allbad;1
TH1D   h1modcb_noisebad;1   proj
TH1D   h1modcbdual_rsigcls0;1   hsig
TH2D   h2modcbdual_sig2cls0;1   h2modcb_sig2cls0
TH1D   h1modcb_matbad;1
TH1D   h1modcb_sipmbad;1
TH1F   h1modcb_Vbd;1   Vbd_from_DB
TH1D   h1modcb_lt;1   h1modcb_lt
TH1D   h1modcb_pos;1   h1modcb_pos
TH1D   LIS_mat_mapping;1   LIS mat mapping
TH1D   LIS_bias_currents;1   LIS bias currents
TH1D   LIS_mod_currents_ly_15_tp_100;1   LIS mod currents: ly=1.5p.e. tp=10.0ns
TH1F   LIS_measured_LY_ly_15_tp_100;1   LIS measured LY: ly=1.5p.e. tp=10.0ns
TH1D   LIS_mod_currents_ly_11_tp_100;1   LIS mod currents: ly=1.1p.e. tp=10.0ns
TH1F   LIS_measured_LY_ly_11_tp_100;1   LIS measured LY: ly=1.1p.e. tp=10.0ns
```

The analysis procedure is following. * some results from former tests in modproduction centers are copied from `moduleTest/data/FSM`. as `mod_FSMxxx.root`. The FSM files are already preprocessed.to get correct channelo rdering The corresponding histos habe `mod` prefix. * the SiPM Vbd are copied from `moduleTest/data/sipm/` taking into account correct ordering ad `BTC_XXX.root`. * the latest summary files in the `QA/` folder is taken as link `Summary.root` and after processing is stored as `modc_FAMxxx.root`. * all `dualtreexxx.root` files are merged to a `dualtree.root` file and the dualtree.root is analysed for the vertical tracks and results are stored for each module in `dual_FAMxxx.root` * all stored root files are analysed together and final results are stored into `sumFAMxxx.root` files for each module, as well as pdf (with some selected plots) and txt files with lists.

Archiving results locally

After the tests the whole directory has to copied into `moduleTests/cern_data/larchive`. From 17.05.2019 all data are stored in `FAM_new` folder assuming the modules are (re)produced with the modified BTC, that is, some modules are retested. There are a few scripts:

- `process_sum_all` - reprocess all data with `sum_QA` scripts
- `collect_sumroot` - collect all `sum_FAMxxx.root` into `sumroot/` folder
- `collect_pdf` -collect pdf files from `sum/` folders into `pdf/=`
- =collect_lis
- - same for lis information, into `lis/` folder

- `collect_sum` -collect all `sum/` folders into `sum/FAM/=`
- =collect-QAsummary
- -collect all `Summary.root` files into `fam/FAMxxx/` folders

To make a local copy in `data/FAM/`, type `./copy_results`.

See the example code below to see which files are copied

more: Close

```
mkdir -p $DATAPATH/FAM/FAM00019
 mkdir $DATAPATH/FAM/FAM00019/sideA
 cp ./config.txt $DATAPATH/FAM/FAM00019/sideA/
 cp ./config.root $DATAPATH/FAM/FAM00019/sideA/
 mkdir $DATAPATH/FAM/FAM00019/sideA/sum
 cp ./sum/sum_FAM00019_A.root $DATAPATH/FAM/FAM00019/sideA/sum/
 cp ./sum/sumgaps_FAM00019_A.txt $DATAPATH/FAM/FAM00019/sideA/sum/
 cp ./sum/sumbadch_FAM00019_A.txt $DATAPATH/FAM/FAM00019/sideA/sum/
 cp ./sum/sum_FAM00019_A.pdf $DATAPATH/FAM/FAM00019/sideA/sum/
 mkdir $DATAPATH/FAM/FAM00019/sideA/lis
 cp ./lis/FAM00019A.lis.txt $DATAPATH/FAM/FAM00019/sideA/lis/
 mkdir $DATAPATH/FAM/FAM00019/sideA/LT
 cp ./LT/lt.txt $DATAPATH/FAM/FAM00019/sideA/LT/
```

Putting results into production Db

Attention: Putting data into database is a SEMI-automated process and needs an active supervision of the operator. Depending on the current state of the database and results, some data may remain not uploaded. One of these cases is when some relevant data already exists in the database.

To start, go to FAM directory and type `./update_db`. Entering DB credentials might be necessary.

FAQ and Troubleshouting

Moving boxes in the grey room and Assembly hall

Usually Robin is organizing this. Need to persons at least. For movement between Grey room and assembly hall need to use a special trolley. The module in the box has to be properly secured and the cover closed. Avoid humid weather for procedure. In the assembly hall the boxes are stored in the mezzanine, so you need a crane people. All work in assembly hall has to be coordinated with a responsible person (eg. Blake L., Sune J.).

Bad SiPM -SPIROC connections and bad channels

Note that the ch1754 on the upper board is abd, please use connector 'saver'!

These connectors are critical, the used connectors are supposed to sustain ~50 connections. On the SiPM flex side the connectors are used only a few times for BTC testing, while on the SPIROC motherboard side many times, hence can have loose contacts. It is recommended to clean them with isopropanol prior to each connection. Check the alignment of connectors before switching LV ON.

There are several problems can be related to bad connections:

- One bad, misaligned connector can cause LV problem, or even break PS(fuse). Sometimes one can hear a sound from PS in case of bad load due to

bad connection.

- Some bad channels seen in the LIS scan with no or little light. Usually these bad channels are spread all other the module in a random way.

After identification of these bad channels and related connector, have to investigate the cause, see below. If it doesn't help one can use so called connector 'saver' board. In case of persistent problem, one can exchange the SPIROC board. It might happen the problem is on the SiPM flex side, then nothing can be done.

- Extra noise in LIS spectra (large width of LIS peaks).

Procedure for the anticipated bad connection(all steps should be written in ELOG), has to be done for each 'bad' channel case if there is more than one.:

- identify well the connector where is the bad channel, write in elog
- disconnect it and investigate with the lens and bright light if there some dust or damages on the SPIROC side or in the SiPM side. Check the SiPM flex.
- reconnect and check the LIS response, note in ELOG if problem is gone/stays. Repeat this procedure 5 times; disconnect/connect, run LIS.

One can envisage 3 cases; a)problem is gone after first reconnect b) bad channel stays in all attempts c) intermittent appear/disappear.

- in case of case a) stop investigation and start QA.
- Case b) first try to clean up the SiPM side of the connector and the SPIROC side(use isopropanol and 'clean' tissue), then connect and repeat the LIS.

If problem is gone go to QA, if not try to use 'saver' small board on the SPIROC side. Check the LIS LY with the 'saver' board and if the problem is gone, go to QA. If not try to use an another 'saver' board, and if 'bad' channel still persists stop the investigation and go to QA. After this test one can spot problem in the SPIROC side of the connector. One have to check if the same channel appears in other modules.

- Case c) bad channel is not stable. Disconnect 'saver' board if it was connected and it didn't help.

Try to clean up both side of the connector on the SPIROC and SiPM sides. Check 5 times if problem is gone or persistent. If the the bad channel is gone go to QA. If it shows up constantly, disconnect and do an another thorough visual inspection on both sides, then go to QA. If the bad channel is still intermittent go one to the next step

- For the persistent case c) try to reconnect a few times to get good connection, checking with LIS. Then use a heater (borrow it in electronic lab upstairs or in assembly hall) and heat the area near connector to <50C during the run and see if this bad channel appear again during the LIS runs. If indeed the bad channel shows up again go to next step, otherwise go to QA.
- if there is a thermal dependancy of bad channel, try to use the 'saver' board and repeat test with the heater. If bad channel doesn't appear go to QA.

This means the problem is related to the SPIROC side. If 'saver' board didnt' help, remove it, connect and check again LIS, then go to QA. In this case most likely the SiPM side is problematic and have temperature dependancy. Inspect agin the SIPMm connector and flex. Then go to QA.

- During QA test of the modules with any detected 'bad' channel during investigation, check if this bad channel appear/disappear during the cosmic tests.

This can be done by checking Summary files saved after each cosmic run of 30 min.

Bad USB connections and corrupted USB libs.

This problem manifests as crashing programs or scripts that rely on the USB box, for example `modqacern` or `dacfiles_from_db`. The the most likely reason is the LV PS. Check that their are powered on. If yes, disconnect LV connector and carefully check all voltages, there are fuses inside can be broken.

After checking that the USB boxes are powered, check the USB connectors on the USB box and PC. If no joy, try to recompile all Libs in `scifiusbboard/` If nothing helps for both USB boxes, the problem might be in PC. One can also check the status of USB drivers `lsusb`. If only one box is problematic, check the USB cables and connectors carefully, try to replug in the PC side.

Can't upload DAC values

See above, most like USB connections or missing LV PS. But it can be that the related BTC files are not available. Can happen for newly produced BTC, check in `moduleTests/data/sipm` folder. Then contact experts.

Bad LIS connections and replacing LIS driver

The flex or connector for the LIS driver can be broken. There are a few spare that can be used, but one have to observe the connector orientation. The connectors does'nt have locks, i.e. can be connected either way. Moreover different boards, flexes production version can have different orientations! Check carefully the specifications.

If LIS driver has to be replaced, the Db entry should be modified and the FAM_FAM folder recreated using the new Db. The new configs will be created. The previous folder can be copied with some prefix, or deleted.

LIS tune: can't get enough

LIS tune is used to find the LIS driver setting that deliver light enough to see 5 p.e. with >1% of events (from total 8000). This corresponds to the average 1.5 p.e. for each driver (512 channels, or one mat). Obviously across the mat the light yield from LIS is not uniform, usually rising toward side where the light fiber is connected to the light bar, however the average per driver (mat) is used sofar. This 1.5 p.e. target may not be reached for some mats, especially mat 0 and 2 that have light bar far from the fiber mat, which is covered by the black foil. This black foil can reduce the light input from the light bar. The LIS light can be increased by increasing duration of the LIS pulse. The SPIROC readout can provide only Tw=10ns pulse, while the final PACIFIC readout up to 20ns (the relation of LIS LY and Tw is not linear, above 20ns not much gain). Thus during LIS tune two target values are used 1.5 p.e. and 1.1. p.e. that last corresponds to the 15ns pulse, and therefore two LIS settings.

The LIS driver is steered by two values in the settings. The Ibias parameter define a constant current of GBLD(laser driver) Icons= 2 mA + Ibias x 0.16 mA. The 2.5mA is subtracted in LIS mezzanine to keep Icons~0 if no Ibias defined. The max Iconst= 42mA and should be big enough to be above thresholds for Light-Icurr curve. The optimal Ibias values are measured at RWTH and are extracted from Db. The Ibias=10 corresponds to the Iconst=15.5mA that covers most of LIS driver threshold values. Increasing Iconst above threshold value doesnt make sense and just increases the SiPM current. However for some LIS drivers it helps, but one have to observe the increase of SiPM constant current (can see rough measurements in Kethley during LIS run), or increase of the LIS peaks in the LIS spectra. This last can be better seen in the summary root file(can produce ot for the incomplete QA run), see *himodcb LIS rms*, and compare with *h1modcb Ped noise* without LIS light. One can also see the actual distribution of LIS LY for all channels in *LIS_pe* histo. The Imod parameter define the laser pulse amplitude that can be up to 12mA, which corresponds to the Imod~60 maximum value. If after LIS tune, the target values for 1.1 are not reached even for maximum Imod, one can do following:

- try to increase/(in case of low light) or decrease (in case of large current) Ibias, checking the width of LIS peaks. To do this, identify which driver it is and rerun using `tune_lis_advanced` that allows to insert Ibias manually. After that do `copy_cfgfiles` and `modqacern` to see results.

If helps after a few iterations, use this Ibias for QA. The final LIS settings will be stored in output txt files in the `lis/` folder and used for summary files and Db upload. Note that the initial Ibias in the Db for each LIS driver remains enchanged.

- exchange LIS driver with smaller Ibias (as in Db). Usually this means slightly large LIS light output. The exchange means you have to change the Db entry for the corresponding module and rerun `start` script (or use `restart`).
- do nothing, noting in ELOG

Note that all operations should be registered in the ELOG.

Large SiPM current

The KETHLEY HV PS for SiPM has a current limitation, usually 0.009A. The typical operation voltage is 58. 0V (then for each SiPM the voltages are adjusted by the SPIROC DAC). During rump up the current can be large and the voltage can rise slowly. If the target voltage is not reached, that is, the current is above the limit. Wait for current evolution a few minutes, if stable: First check the position of switched on the switch box and SPIROC board. Then check the SiPM flex connectors(clean, reconnect) If everything is OK, you have a problem. First try to identify which module and SiPM flex is causing the problem (by disconnecting). The reason can be a large light leak, try to cover and fix , or electrical problem. In the latest case BTC has to be exchanged most likely.

Note that the current can increase significantly during the LIS tests, this is also a bad sign, can be related to the large Ibias current. Usually the LISspectra lokks bad in this case, and the Ibias can be reduced and lis tuning repeated.

Large current in LT test

The maximum value under illumination is 5 uA. The light leak is usually near the cold box side and can be fixed by metal tape. If the light leak is localized on the module we are in the trouble, because there is no clear procedure how to fix it. Note that some modules can have some leak (<5uA) in the middle of the module. some of these leaks is marked by the tape. This is acceptable if below the limit.

Cosmic hodoscope trigger rate

The typical trigger rate of cosmic hodoscope is about 150 Hz ranging in 120-170 Hz. Much lower or much higher rate can be due to:

- No HV. Check it, switch Off and On.
- some thresholds in discriminators are off (due to defected trimmers). Try to turn threshold a bit, see results,
- some extra RF noise affecting hodoscope readout. Then can try to increase thresholds (loosing efficiency)

PC wifi

The `lhcbscifi` PC doesn't have internet cable connection, it run via wifi board installed in the PC. If there is o connection, check that the wifi is working (using your laptop/mobil). Then try to restart wifi server in PC

Missed Db connection, or no entries

If the relations between objects is not known, the testing is impossible. If some information is missed in Db, better first to find out what and enter in Db. The testing require some information to be downloaded and preprocessed already, eg: * the Vbd for the BTC that is extracted from SiPM testing, the prerpocessing order things in correct way. This is used for SPIROC DAC prior to any measurements. * Ibias current for LIS driver used during LIS tune * the results from modtest in production centers (HD and NIKHEF), used in final analysis

Retesting the module

There are a few reasons to retest module(s); reassembly fixing BTC problem or any other problem without changing components, reassembly with changed components, repetition if previous data are lost/corrupt, some more extensive testing, etc. Important that the previous information is not lost, that is, the existing folder with the same name should be renamed. If only one module is retested, and the latest data should be used for QA, one has to move/rename the old summary files for related module. The `upload_db` should be repeated for corresponding entries to overwrite the old one. All changes should be written in the ELOG.

Crashed QA cosmic test

Most likely not enough space on the main disk (250Gb). Dont forget to move folders from tested modules to archive to clean up the space/ After cleaning can restart the run. After the crash the last summary files and dualtree.root should be available, try to open the files in root. Starting new run would not overwrite the existing one, the dualtree will be appended in postprocessing. The summary files are stored with the timestamp, i.e. status can be recovered. However in this case only last run statistics will be used. In principle the last summary file from crashed run can be merged with the new run, but one has to account for increase of statistics in LIS runs.

System maintenance (for experts only)

Replacing USB box

In order to replace one of existing USB boxes (36 or 37) with a spare one (22), do the following:

1. Locate `$USBBOARDPATH/cfgFiles/board22.cfg.bak`, edit it and put all contents from the relevant file (`board36.cfg.bak` or `board37.cfg.bak`)
2. Go to line **7** of this file and edit it to: `usbBoardID || 22 ||`, save file and close editor.
3. Locate `$USBBOARDPATH/cern_scripts/utils.h`, edit it and modify the relevant define at line 4 or 5 (`USB_ID_TOP` or `USB_ID_BOT`), save file and close editor.
4. If you already created your working directory (`$CERNDATAPATH/FAMxxxxxX_FAMxxxxxxX`), go into it, `./restart` and `./copy_cfgfiles`.
5. Now you are running with new USB Box.

If you have a spare with different USB ID, the procedure is the same; just replace 22 with your USB ID.

Adjusting LV PS voltages

The +/-2V outputs of Spiroc setup PSes are adjustable. It has been checked that their adjustment has significant effect on the readout noise.

In order to adjust the voltages, remove the PS outer cover and locate two miniature pots on the two leftmost channels of the PS (looking from LV output side). To adjust vlotages, use a small screwdriver and a voltmeter to control the voltage. The nominal value now is **2.15V**.

Remarks:

- It's recommended to set both outputs to the same voltage.
- Any increase of voltage should be consulted at least with Michal.
- The lowest 'working' setting is 2.1V. For lower values, on-board voltage regulators for Spiroc chips do not work properly.
- Always tune the PS warmed-up. The diagnostics ('third') setup can be used as a load.

Taking LIS global scan

The LIS scan is a procedure to collect data for a specified range of LIS settings. It is not intended to be used for every module, it's just for studying LIS properties. The existing board.cfg has to be backuped before the scan, because will be overwritten.

To perform a LIS scan:

1. save existing board36/37.cfg in `sw/scifiusbboard/cfgFiles` , they will be overwritten by the scan. One can restore them after the scan.
2. Go to your working directory (FAM_FAM)
3. Go to `lis/` and create a sub-directory for output data, e.g. `scan/`, cd into it. Alternatively, you can use any directory in any place.
4. Run the scanning script: `root -l -q '$USBBOARDPATH/cern_scripts/lis_tuning/scan_lis.c("../../config.root")'`.
   If you run from different directory as stated above, please ensure that you have supplied a correct path to your `config.root` file or leave this parameter empty(`""`) to make the script ask for bias settings..
5. You get a number of files: `lis_raw_??.root` (with raw histograms) and `lisAnalysis_??.root` (with calculated pedestals, gains and LYs), where `??` is a decimal number with current modulation setting for all LIS channels. Ibias for LIS driver settings are taken from `config.root` or from manual input if you don't specify the file.

Remark: The original modulation values in the board.cfg files will be overwritten. Therefore, you should run tune_lis after the scanning procedure or manually recover these settings.

-- ValeryZhukov - 2019-01-28

This topic: LHCb > SciFiFibreModulesQA
Topic revision: r81 - 2020-08-04 - ValeryZhukov