Preparation for calibration measurements

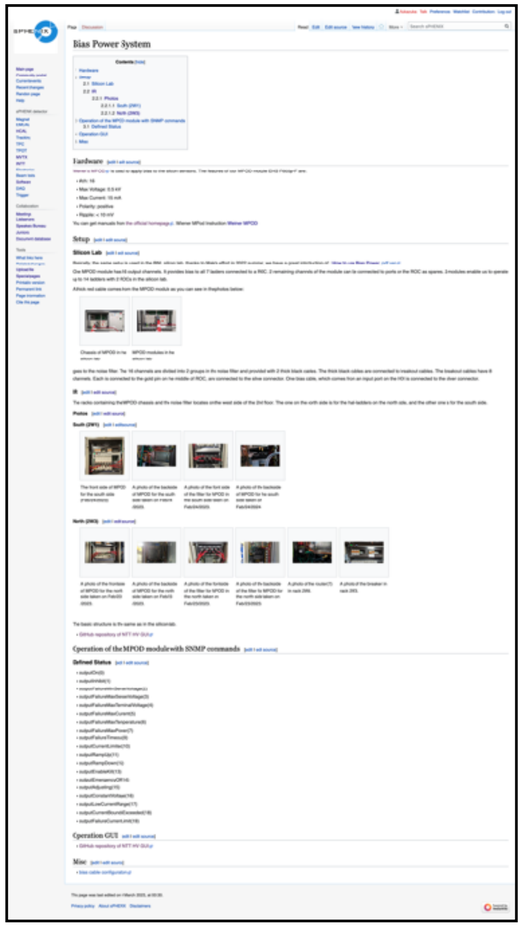
Genki Nukazuka (RIKEN)

What is needed for the ladder tests?

- **Hardware**
  - Cable connections
    - conversion cables (by INTT crews)
    - bias (D. Cacace *et al.*)
    - FPHX power (D. Cacace *et al.*)
    - ROC power (D. Cacace *et al.*)
    - data fibers (D. Cacace *et al.*)
    - slow control fiber (D. Cacace *et al.*)
  - ROC cooling connections, cleaning, and tests (Rob)
  - Power for our power racks
- **Software**
  - LV GUI (Maya): Ready
  - HV GUI (Joseph): we need individual control and current/voltage monitor. Where is that version?
  - Calibration feature in the FELIX firmware (Raul)
  - Analysis software: ongoing
  - Data conservation/management (?):

What is needed for the ladder tests?: HV GUI

- The shifter's GUI doesn't have many features:
  - Individual channel operation
  - Current/Voltage monitor
- Where is our expert's GUI?
- Instruction has to be documented in the sPHENIX wiki, by the way.



Wiki

FELIX

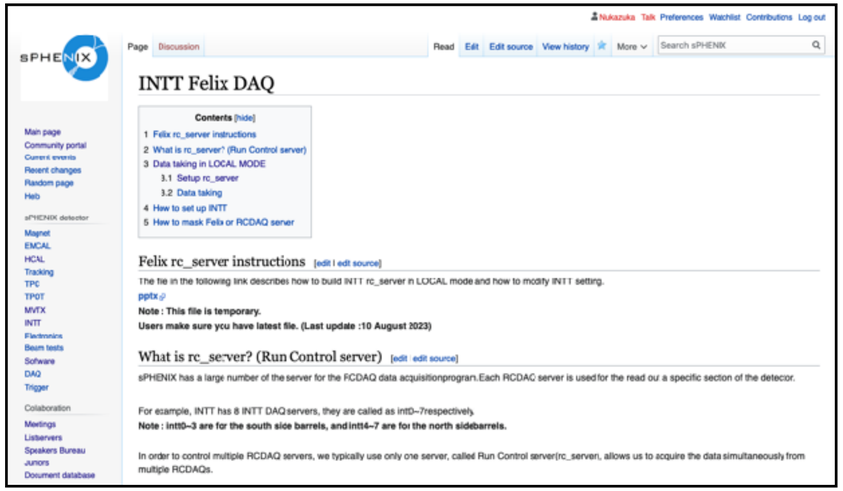
- Raul has been implementing FELIX firmware for calibration measurements.
- Now, he is in the phase of test measurements: measurements with a temporary setup were started.
- A short meeting was held among Raul, Jaein, Akitomo, and me to know the status.
- The idea of the calibration mode is quite simple: running INTT in the streaming readout mode (n_collisions = 127, open_time = 127) with a specific GTM scheduler, and taking data with RCDAQ (not big-partition but small-partition).
- What needs to be implemented/modified:
  - **~/INTT/run.py** in the sPHENIX common directory at 1008:
    Minor modification to change the flag dam::sc_target, which controls whether ROC(s) accept slow control or not, to be enabled during the measurement.
  - **~/operations/INTT/modebits.sh** in the sPHENIX common directory at 1008:
    The scheduler to be loaded needs to be changed.
  - **~/INTT/sphenix_inttpy/run_scripts/fphx_parameters_calib.txt**: Let's check →
    I made another file for the original calibration parameters (fphx_parameters_calib_no_bex.txt) just in case.
  - Channel mask: Noisy chips are masked ch by ch, so it takes time.
    It needs to be updated (low priority).

DAC parameters were optimized by Itaru for the setup with a bus-extender.

Vref	1
#DAC0	18
DAC0	21
#DAC1	19
DAC1	22
DAC2	23
DAC3	27
DAC4	31
DAC5	35
DAC6	39
DAC7	43
N1sel	6
N2sel	4
FB1sel	4
Leaksel	0
P3sel	0
P2sel	4
Gsel	2
BWsel	8
P1sel	5
Injsel	0
LVDS	63

If you don’t remember how to run INTT in local mode…

- Jaein prepared a nice manual for YOU!
  https://wiki.sphenix.bnl.gov/index.php/INTT_Felix_DAQ



INTT Felix DAQ

Contents [hide]

1 Felix rc_server instructions
2 What is rc_server? (Run Control server)
3 Data taking in LOCAL MODE
3.1 Setup rc_server
3.2 Data taking
4 How to set up INTT
5 How to mask Felix or RCDAQ server

Felix rc_server instructions [edit | edit source]

The file in the following link describes how to build INTT rc_server in LOCAL mode and how to modify INTT setting.

pptx

Note : This file is temporary.

Users make sure you have latest file. (Last update :10 August 2023)

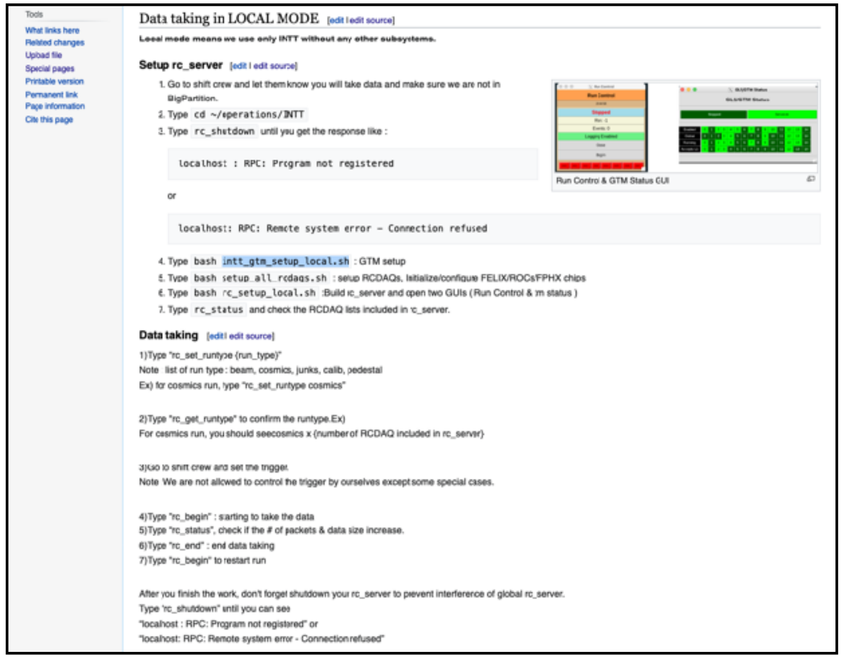
What is rc_server? (Run Control server) [edit | edit source]

sPHENIX has a large number of the server for the FCDAQ data acquisitionprogran.Each RCDAC server is used for the read ou a specific section of the detecor.

For example, INTT has 8 INTT DAQ servers, they are called as intt0~7respectively.

Note : intt0~3 are for the south side barrels, and intt4~7 are for the north sidebarrels.

In order to control multiple RCDAQ servers, we typically use only one server, called Run Control server(rc_server), allows us to acquire the data simultaneously from multiple RCDAQs.



Data taking in LOCAL MODE [edit | edit source]

Local mode means we use only INTT without any other subsystems.

Setup rc_server [edit | edit source]

1. Go to shift crew and let them know you will take data and make sure we are not in BigPartition.
2. Type `cd ~/operations/INTT`
3. Type `rc_shutdown` until you get the response like :

```
localhost : RPC: Program not registered
```

or

```
localhost: RPC: Remote system error - Connection refused
```

4. Type `bash intt_gtm_setup_local.sh` : GTM setup
5. Type `bash setup_all_rcdaqs.sh` : setup RCDAQs, Initialize/configure FELIX/ROCs/FPHX chips
6. Type `bash rc_setup_local.sh` :Build rc_server and open two GUIs (Run Control & rm status)
7. Type `rc_status` and check the RCDAQ lists included in rc_server.

Run Control & GTM Status GUI

Data taking [edit | edit source]

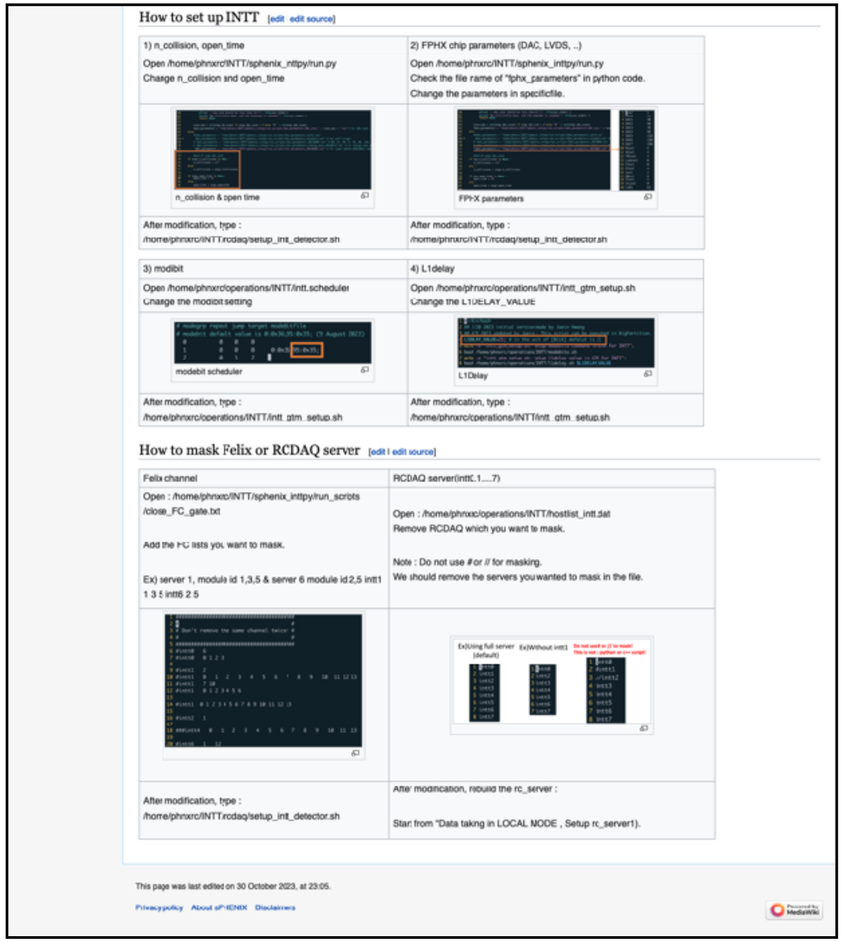
1)Type "rc_set_runtype (run_type)"
Note list of run type : beam, cosmics, junks, calib, pedestal
Ex) for cosmics run, type "rc_set_runtype cosmics"

2)Type "rc_get_runtype" to confirm the runtype.Ex)
For cosmics run, you should seecosmics x (number of RCDAQ included in rc_server)

3)Go to shift crew and set the trigger.
Note We are not allowed to control the trigger by ourselves except some special cases.

4)Type "rc_begin" : starting to take the data
5)Type "rc_status", check if the # of packets & data size increase.
6)Type "rc_end" : end data taking
7)Type "rc_begin" to restart run

After you finish the work, don't forget shutdown your rc_server to prevent interference of global rc_server.
Type 'rc_shutdown" until you can see
"localhost : RPC: Program not registered" or
"localhost: RPC: Remote system error - Connection refused"



How to set up INTT [edit | edit source]

1) n_collision, open_time	2) FPHX chip parameters (DAC, LVDS, ..)
Open /home/phnxrc/INTT/sphenix_inttpy/run.py Change n_collision and open_time	Open /home/phnxrc/INTT/sphenix_inttpy/run.py Check the file name of "fphx_parameters" in python code. Change the parameters in specificfile.
n_collision & open time	FPHX parameters
After modification, type : /home/phnxrc/INTT/rcdaq/setup_intt_detector.sh	After modification, type : /home/phnxrc/INTT/rcdaq/setup_intt_detector.sh
3) modibit	4) L1delay
Open /home/phnxrc/operations/INTT/intt.scheduler Change the modibit setting	Open /home/phnxrc/operations/INTT/intt_gtm_setup.sh Change the L1DELAY_VALUE
modebit scheduler	L1Delay
After modification, type : /home/phnxrc/operations/INTT/intt_gtm_setup.sh	After modification, type : /home/phnxrc/operations/INTT/intt_gtm_setup.sh

How to mask Felix or RCDAQ server [edit | edit source]

Felix channel	RCDAQ server(intt0.1.....7)
Open : /home/phnxrc/INTT/sphenix_inttpy/run_scripts /close_FC_gate.txt Add the FC lists you want to mask. Ex) server 1, module id 1,3,5 & server 6 module id 2,5 intt1 1 3 5 intt6 2 5	Open : /home/phnxrc/operations/INTT/hostlist_intt.dat Remove RCDAQ which you want te mask. Note : Do not use # or // for masking. We should remove the servers you wanted to mask in the file.
	Ex)Using full server (default) Ex)Without intt3
After modification, type : /home/phnxrc/INTT/rcdaq/setup_intt_detector.sh	After modification, reboot the rc_server : Start from "Data taking in LOCAL MODE , Setup rc_server1).

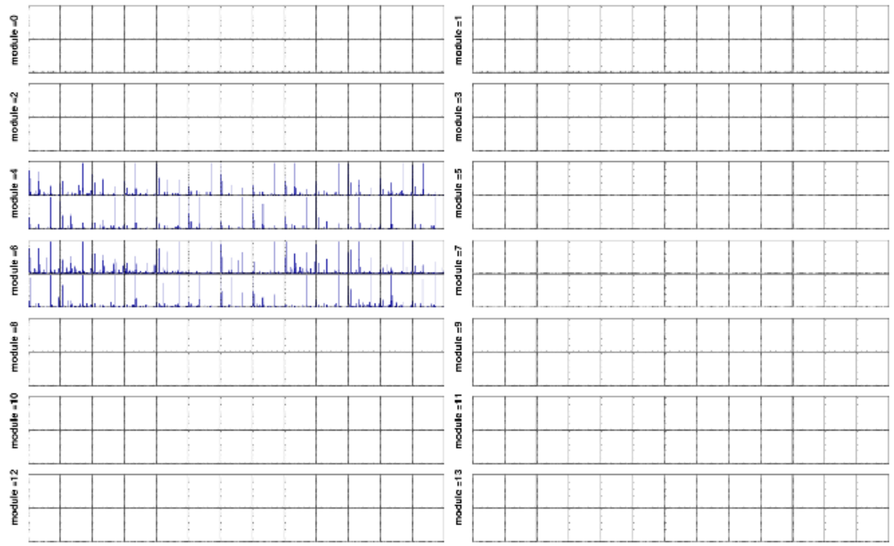
This page was last edited on 30 October 2023, at 23:05.

Privacy policy About sPHENIX Disclaimers

We can follow those step for calibration measurements!

Software status

- **Decoder**: I modified Takashi's event-base decoder to output the hit-base decoder so that we get all hits without any loss. I don't use Fun4All decoder. What we want is DST generated in a proper manner but just quick results.
- **FelixQuickViewer**: It's working well. I'm optimizing it.
- **Ladder map**:
  I cleaned the directories. The latest version in 2023 was copied to the 2024 directory. Symbolic links to them were made under map_ladder. Those symbolics are used by other applications. There are some map_ladder directories which are NOT synchronized…
  - ~/INTT/map_ladder at 1008
  - /sphenix/tg/tg01/commissioning/INTT/map_ladder in SDCC
  - /home/inttdev/INTT/map_ladder in inttdev@inttdaq
    Migration to PostgreSQL was done, but users (apps) need to be modified.



intt_intt1-00014056-0004 (junk data)
FelixQuickViewer itself is working well.

```
[nukazuka@sphnx03 08:01:21 INTT] $ tre -d  map_ladder/
map_ladder/
├── README.md
├── intt0_map.txt -> 2024/intt0_map.txt
├── intt1_map.txt -> 2024/intt1_map.txt
├── intt2_map.txt -> 2024/intt2_map.txt
├── intt3_map.txt -> 2024/intt3_map.txt
├── intt4_map.txt -> 2024/intt4_map.txt
├── intt5_map.txt -> 2024/intt5_map.txt
├── intt6_map.txt -> 2024/intt6_map.txt
├── intt7_map.txt -> 2024/intt7_map.txt
├── 2023
│   ├── silicon_lab
│   └── IR
├── 2024
└── 2025
```

Directory structure under map_ladder.