

Calibration Unit hand-off

The calibration unit (CU)



Flight Hardware : TKR (FM8) & CAL (FM101) on bay 3 (close to the edge) and CAL (FM109) on bay 1. We took photos from [arrival](#) and during [installation](#) in the clean room.

No modifications were done or rework on towers during the beam test. During integration into the 1 x 4 grid one TKR connector (between flex cable 7 and the TEM) of **non-flight TKR** was slightly offset and we added a connector saver to avoid stress in the cable.

Note for mechanical crew : For future operations, note that clearance inside is very small, so exercise care before removing the Inner Shipping Container. Since SLAC has a precision crane this will not be an issue.

Eliazar Ortiz assured us that the clean room (room 104) is continuously monitored for humidity and temperature.

Downloaded data from recorder: max humidity 7%, max temp 33 C, Max force 4 g. Eliazar downloaded all [data](#), which was then provided to John Ku for review. Note: Dates are logged in awkward format: first day was April 24, last day was May 1 2007.

Summary of Results

The CU is different from the LAT in many respects, so is the software to run it. There is essentially only a single local installation allowing to run the CU, right now, and it resides on the lat-btserver - latbt01/lat-bt04 (spare) machines. Here are some notes about the installation itself, the tests that can be performed, the schema files to be used.

No anomalies or failures were experienced over the course of the beam test that would degrade the flight hardware. The fact that CU passed functional tests below demonstrates that.

All data from the beam test can always be accessed through the [beam test database](#). These data is stored at SLAC since all tests were conducted using the SLAC pipeline while at CERN or GSI.

The documentation of all beam test activities is available in this [link](#)

Tests at SLAC were carried out by Carmelo Sgro and Lua Baldini and data reviewed by Eduardo do Couto e Silva and Jana Thayer.

As part of the knowledge transfer process from Pisa to SLAC, Luca and Carmelo taught Jana how to operate the CU.

Tests

Here is the list of functional tests we performed as part of the incoming procedure.

There is no TKR module 1 so in LATTE do not power on this tower otherwise you will get errors.

For charge injection runs, tests are self-contained and results labeled **PASSED** are indeed meaningful (we also inspected the reports). Charge injection tests are done one tower at the time.

For data taking tests we added a reference run taken from the Beam Test at CERN (see below) because the PASS/FAIL condition is affected by how we treat the well-known issue of TKR FIFO errors.

Jana will ask Patrick to back-up all data from the computers. Reference runs for data taking were not linked here because we are behind the firewall and it does not make sense to do all the work manually when the whole disk will be backed up.

Note about tests: When the CU is powered up and initialized after having been off for an extended period of time, trigger parity (and other data acquisition) errors are often observed for the first few hours of running. This problem has been consistently observed over the life of the CU and is believed to be a DAQ issue (**we do not use Flight DAQ modules, nor FSW in the CU**), not a CU problem. Consequently, some events taken within the first few hours after the CU is powered will show evidence of parity errors until the problem resolves on its own on the timescale of a few hours. The quality of the rest of the data is not adversely affected. There has been some investigation on this from our team and Ric Claus, but we never came to a conclusion. Luca Latronico collected all the relevant info in some emails. Jana is aware of the issue and agrees that due to the transient and non-fatal nature of the problem, it is a feature that does not need to be fixed. Further investigation would involve swapping out the GASU and attaching the CU to a more flight-like DAQ, which is in line with the current plans for the unit. This problem has not interfered with the operation of the CU in the past and it is not expected that it will interfere in the future.

TKR

- Noise and gain measurement (TkrNoiseAndGain.py)

- Reference run for the TKR module attached to TEM 3: run 700004421 (**PASSED**)
- Reference run for the TKR module attached to TEM 2: run 700004435 (**PASSED***)

* Note that TKR FM16 (connected to TEM 2) is not flight and it does have its "features" and idiosyncrasies. In particular there are quite a few layers failing in terms of disconnected/defecting channels (Y15, X13, Y11, Y10, Y2). A good reference run for comparison (though quite old, now, and taken with a different software version) is run [700000107](#):

- Noise occupancy measurement (TkrNoiseOccupancy)
 - Reference run for the TKR module attached to TEM 3: run 700004422 (**PASSED**)
 - Reference run for the TKR module attached to TEM 2: run 700004436 (**PASSED**)

CAL

- Standard CPT (CPT_beamtest.py)
 - Reference run for the CAL module attached to TEM 3: runs 700004423-700004433, summary 700004434 (**PASSED**)
 - Reference run for the CAL module attached to TEM 2: runs 700004437-700004447, summary 700004448 (**PASSED**)
 - Reference run for the CAL module attached to TEM 1: runs 700004449-700004459, summary 700004460 (**PASSED**)

Data taking

- Standard end2end (BT 2, nominal conditions, all triggers enabled). See [table of configurations](#)
 - 700004466 ~ 2 kevents
 - 700004467 ~ 12 kevents
 - reference run from BT [70000657](#) Triggers rates for TKR 2 and 3 are slightly different but consistent with results from CERN tests

Installation

Both the LATTE and the INT releases are installed onto lat-bt01 in

```
/usr/local/online
```

For the beam we had to make quite a few specific modifications to the code (both the core stuff and the subsystem scripts) to be eventually able to run a limited subset of the test scripts. Rather than modifying the released code directly, those changes live in a separate folder:

```
/gnfs/home/glast/work/INT/DebugArea
```

which is prepended to the \$PYTHONPATH (i.e. the system searches for hacked versions of the python files here, first, and then looks into the standard installation directory). The files living in this folder can be modified by the user glast (i.e. you don't need to be root).

Links to the relevant configuration files and test scripts also live nearby, the directory structure being the following:

```
/gnfs/home/glast/work/INT/repos/Schema/Calibration (CU_TEM_1.xml, CU_TEM_2.xml, CU_TEM_3.xml)
```

```
/gnfs/home/glast/work/INT/repos/Schema/DataTaking (CU_BT.xml)
```

```
/gnfs/home/glast/work/INT/Scripts/Calibration/TKR (TkrNoiseAndGain, TkrNoiseOccupancy)
```

```
/gnfs/home/glast/work/INT/Scripts/Calibration/CAL (CPT_beamtest)
```

```
/gnfs/home/glast/work/INT/Scripts/DataTaking (end2endBT)
```

Essentially you want to use end2endBT for data taking (and CU_BT.xml as the configuration file) while the charge injection tests must be performed for each module separately using the CU_TEM_xxx.xml files. Note that both the TKR and the CAL modules for the tower under test are declared in the schema file and must be powered on. When you press "select application" in the RunControl main gui you are automatically redirected into /gnfs/home/glast/work/INT/Scripts/DataTaking. Go one level up and then select Calibration to have access to the subsystem level calibration scripts.

Data live either in:

```
/gnfs/data/DVD
```

or:

```
/gnfs/data/LATTE
```

depending on the time of collection.

The end2end data taking configurations are listed [here](#).