

Test Beam Line Status

User Manual

Rafał Bielski,
Gdansk University of Technology, Poland,
DESY Summer Student 2012

September 4, 2012

1 Requirements

- PC with Linux operating system, preferably Scientific Linux or Ubuntu
- TLU, preferably version 0.2c
- (optional) 2 or 3 channel TTI power supply (*3 channels not yet supported!*)

2 Compiling and Running

2.1 ROOT

Before compiling and running the application, one has to be sure to have ROOT installed. If not sure, type `echo $ROOTSYS` in terminal. It should print the directory, where ROOT is installed. If it doesn't, probably ROOT is not installed. Then, if working on a PC in DESY network, one can follow instructions below.

The procedure can be dependend on machine, system and ROOT version, e.g. for 64-bit Red Hat Linux 5.0 or similar and ROOT 5.32 use commands:

```
/afs/desy.de/products/root/amd64_rhel50/5.32.00/  
source bin/thisroot.sh
```

or for any 64-bit system and ROOT 5.28:

```
/opt/products/root64/5.28.00/  
source bin/thisroot.sh
```

2.2 TLU Libraries

Before compiling one has also to make sure that libraries `libtlu.so` and `libeudaq.so` in `lib` directory has been created for current operating system.

2.3 Compiling

To compile the application, enter the `src` directory and use command:

```
make TestBeamLineStatus
```

To remove all objects and rebuild them, use `make clean` before.

2.4 Running the application

To run the application, enter the main application directory (`TestBeamLineStatus`) and use script `runTestBeamLineStatus.sh`. There is an option to disable TTi power supply support: `--tti-disabled`. Use this option, if there is no TTi connected to serial port. Otherwise the application will crash.

The shell script uses Z shell (`zsh`). If it is not available, change it in the script or in the Makefile.

3 Using the application

3.1 Main window

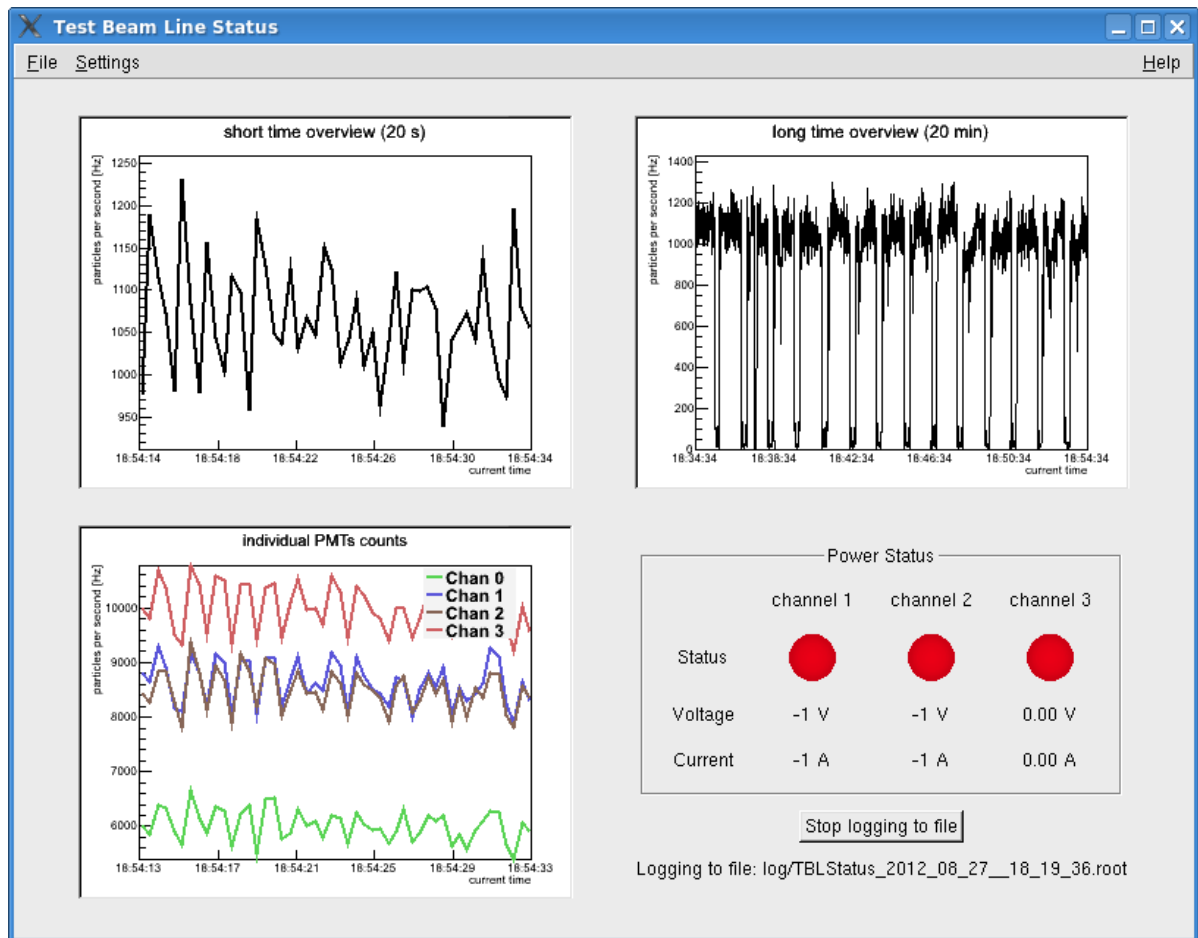


Figure 1: Overview of the main window.

Application's main window contains 3 on-line made plots and a power status box. The top two plots show number of particles per second in different time ranges - short and long. Number of particles represent number of coincidences of hits in detectors, counted using AND mask, which can be changed in TLU settings. The bottom left plot shows hits per second in each of four TLU channels. In the power box, one can check real values of voltage and current supplied by TTi. The LED-like icons show if the values are proper for detectors and TLU. To save currently seen plots to a PDF file go to *File>Save plots as PDF*.

WARNING: While moving or resizing the main window, measurements may be slightly distorted.

3.2 Settings

3.2.1 TLU Settings

In the TLU settings window one can change following settings:

- **short time overview range** - time range on top left plot expressed in seconds.
- **long time overview range** - time range on top right plot expressed in minutes.
- **DUT mask** - mask setting which RJ45 output sockets are used.
- **AND mask** - AND mask for counting coincidences.
- **OR mask** - OR mask for counting coincidences.

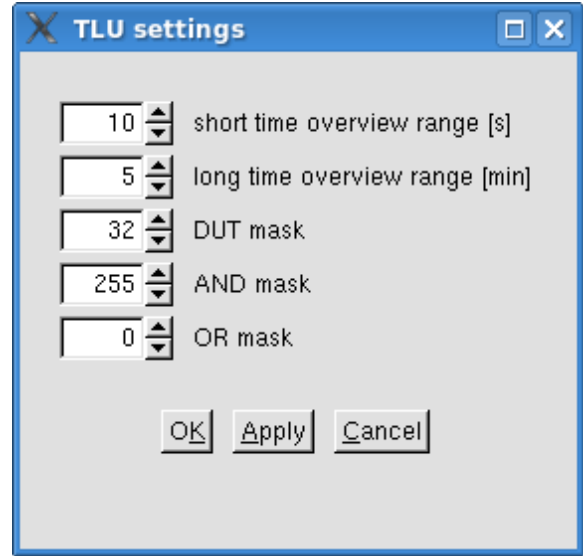


Figure 2: TLU settings window.

Decimal masks numbers are converted to binary numbers, literally telling which channels are used. For example setting AND mask to 6, which is 01100000 means that coincidences in channels 1 and 2 are counted, while setting it to 13, which is 10110000 means that coincidences in channels 0, 2, and 3 are counted.

3.2.2 TTi Settings

Here one can set voltage and current values on TTi power supply. Note that real values will be different from set ones. While voltage should be close to set value, the current set value acts rather as limit, not exact number.

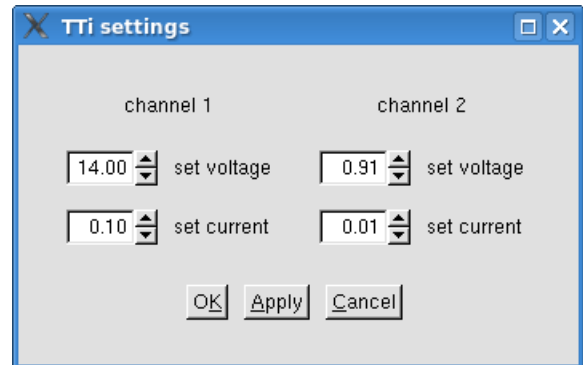


Figure 3: TTi settings window.

3.2.3 Advanced Settings

In the advanced settings window one can change following settings:

- **logging interval** - how often values are logged to logfile if logging is on. Option is applied to the next logfile started, so that one logfile can have only one logging frequency. The more often values are logged, the bigger logfile will be. Default value is 1000ms. Note that this value is set as TTimer object parameter and even though the logging code should be run each n milliseconds, it's never that exact. The logging interval is limited by the time of code execution. There are also some unknown delays caused by TTimer class itself. Note that in tests, real times lower than 80ms have never been reached. Setting this to 0 means 'as fast as possible'.
- **TLU readout interval** - how often current particles and hits values are read from TLU. This value has the same limitations as logging interval.
- **TTi refresh interval** - How often values in power status box are updated. Sending queries to TTi too often may affect measurements and also may take no effect, because TTi response time can be around 1 second. Voltage will not change rapidly and oscillate, so there is no need to check it so often.

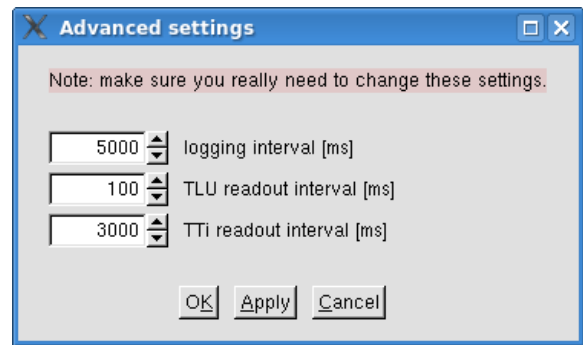


Figure 4: Advanced settings window.

4 Log file

When logging option is turned on, data is saved to .root file in **log** directory. Logging occurs with time intervals, that can be changed in settings (see section 3.2.3). The log file is written when 'Stop logging to file' button is pressed or application is closed. In case of any problems, backup log file is also saved each 10 minutes. If logging interval is bigger than 10 minutes, backup interval is equal to logging interval.

The log file contains one ROOT tree, which contains following branches:

- **ClockTime** - the system time, at which the entry was written.
- **DiffTime** - time in seconds that lapsed from last entry.
- **Part** - particles that have been counted since last entry.
- **PartPerSec** - average particle flux in time from last entry. This value is equal to Part/DiffTime.

- **HitsChN** (N=0..3) - hits that have been counted in each channel since last entry.
- **HitsChNPerSec** (N=0..3) - equal to HitsChN/DiffTime.
- **DutMask, AndMask, OrMask** - current TLU settings (see section 3.2.1).
- **tluInterv** - TLU readout interval (see section 3.2.3).
- **DispSec, DispMin** - current time ranges on plots.

If the TTi support is not disabled, also following branches are saved:

- **TTiInterv** - TTi readout interval (see section 3.2.3).
- **VNset, INset** (N=1,2,(3)) - set values of voltage and current.
- **VNreal, INreal** (N=1,2,(3)) - real read values of voltage and current.