September 29, 2005
Adam Roe
Northeastern University

L1 Cal Run IIB Control Room Monitoring GUIs

As a Whole

There are now three control room GUIs underway for run IIb L1Cal. Nearing completion is a Low Voltage Power Supply GUI. There is also a data monitoring GUI which will receive energies from the L1 Cal TCC; this is similar in principle to the text output that was on running on the IIa TCC in MCH1. Lastly there is work underway on an error monitoring GUI which will readout error and status registers from the TABs and the GAB via the Bit3 interface. As with the former iteration of this document, the purpose here is to open the discussion about what more would be useful. There seems to not be much need to change the power supply GUI, but the other two are still in early development, and their progress and usefulness is dependent on input. Input on any level is helpful, especially given the timescale within which we would like to see these programs operational.

Low Voltage Power Supply

This is “under control”. The Wiener crates are interfaced via CanBus to the EPICS server. The GUI reads the EPICS record. It receives information on voltages and currents, temperature, fan speed, and status(on/off). As with other EPICS-based monitoring programs, it displays values and colors. These colors represent warning levels: green is good, yellow is a warning and red is a problem. These colors (alarm levels, really) have not yet been set. We do not yet know what standard operational current loads are for the supplies. When we do, they'll be set. This “page” can be picked up and put into any other GUI that uses a similar interface. In terms of shifters, it is worthwhile to note that CalTrack and Layer 0's power supply GUIs will be very similar to this GUI.

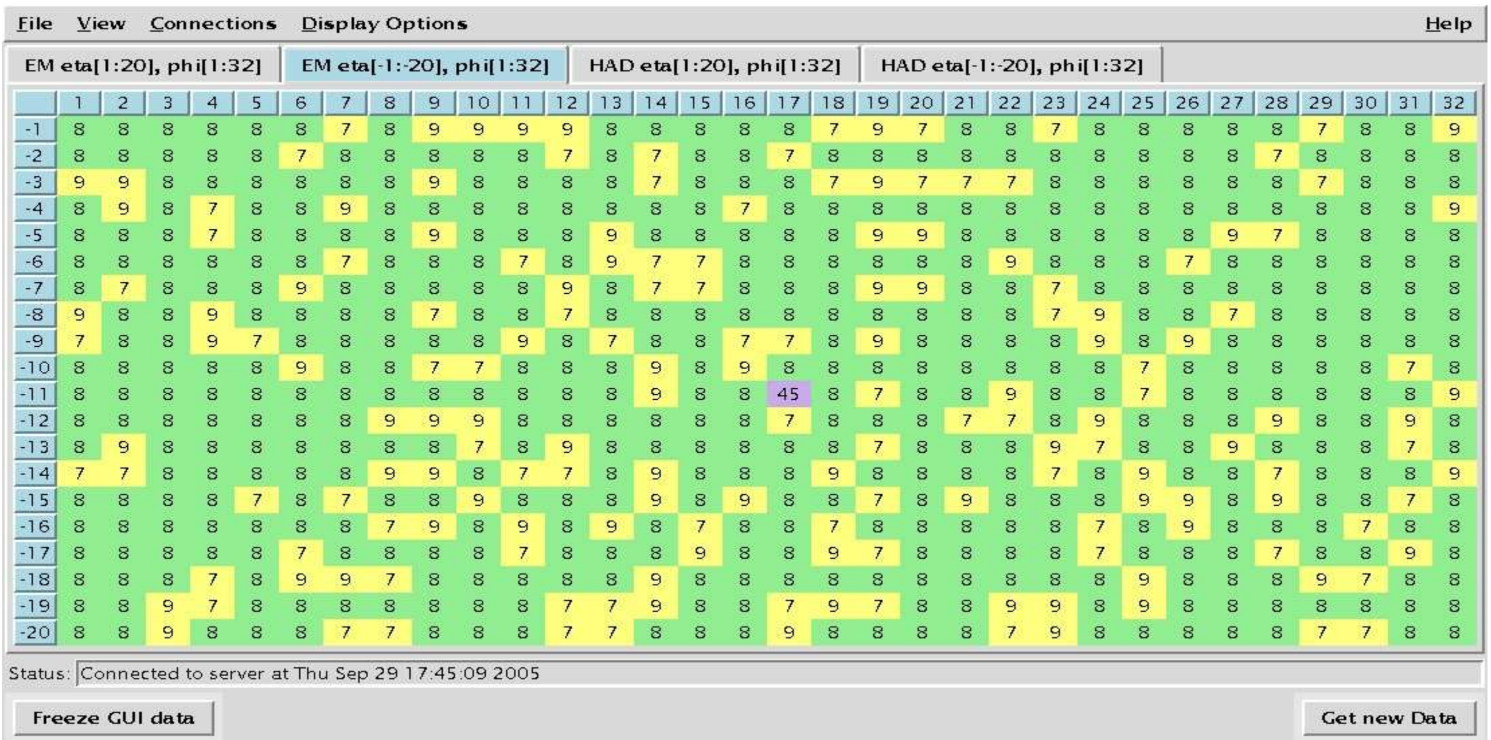
Cal Supply Device	Power	Chan U0 +5.0 V	Chan I0 Load (A)	Chan U1 +5.0 V	Chan I1 Load (A)	Chan U3 +3.3 V	Chan I3 Load (A)	Chan U5 +5.0 V	Chan I5 Load (A)	Temp (deg C)	Fan 1 (rev/sec)	Fan 2 (rev/sec)	Fan 3 (rev/sec)
L1CAL_LVPS_11	on	5.02	18.29	4.99	10.76	3.29	6.42	5.00	9.46	26	55	57	55
L1CAL_LVPS_13	off	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	28	0	0	0

Note that this GUI is running in parallel to the SES and it's functions; there is no conversation between the two. They both receive information from EPICS.

Trigger Tower Energies

This is now functional but incomplete. This GUI monitors data from the L1 Ca; TCC. The most basic function is to display all energies in all Trigger Towers (both EM and HAD portions) for a given timeslice. This GUI has two main purposes. The short-term purpose is to serve as a diagnostic tool during preparation for commissioning and commissioning itself. It's functionality in this respect is highly dependent on how we are commissioning. The longer-term goal is to be a shifter GUI during IIB.

The GUI follows the general visual format of control room GUI's, though it does not function using EPICS. This is displaying a <u>test</u> pattern which I am receiving from a mock server at MSU. It looks like a hot tower, but it's a test pattern that Philippe has sent:



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
-1	8	8	8	8	8	8	7	8	9	9	9	9	8	8	8	8	8	7	9	7	8	8	7	8	8	8	8	8	7	8	8	9
-2	8	8	8	8	8	7	8	8	8	8	8	7	8	7	8	8	7	8	8	8	8	8	8	8	8	8	8	7	8	8	8	8
-3	9	9	8	8	8	8	8	8	9	8	8	8	8	7	8	8	8	7	9	7	7	7	8	8	8	8	8	8	7	8	8	8
-4	8	9	8	7	8	8	9	8	8	8	8	8	8	8	8	7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	9
-5	8	8	8	7	8	8	8	8	9	8	8	8	9	8	8	8	8	8	9	9	8	8	8	8	8	8	9	7	8	8	8	8
-6	8	8	8	8	8	8	7	8	8	8	7	8	9	7	7	8	8	8	8	8	8	9	8	8	8	7	8	8	8	8	8	8
-7	8	7	8	8	8	9	8	8	8	8	8	9	8	7	7	8	8	8	9	9	8	8	7	8	8	8	8	8	8	8	8	8
-8	9	8	8	9	8	8	8	8	7	8	8	7	8	8	8	8	8	8	8	8	8	8	7	9	8	8	7	8	8	8	8	8
-9	7	8	8	9	7	8	8	8	8	8	9	8	7	8	8	7	7	8	9	8	8	8	8	9	8	9	8	8	8	8	8	8
-10	8	8	8	8	8	9	8	8	7	7	8	8	8	9	8	9	8	8	8	8	8	8	8	8	7	8	8	8	8	8	7	8
-11	8	8	8	8	8	8	8	8	8	8	8	8	8	9	8	8	45	8	7	8	8	9	8	8	7	8	8	8	8	8	8	9
-12	8	8	8	8	8	8	8	9	9	9	8	8	8	8	8	8	7	8	8	8	7	7	8	9	8	8	8	9	8	8	9	8
-13	8	9	8	8	8	8	8	8	8	7	8	9	8	8	8	8	8	8	7	8	8	8	9	7	8	8	9	8	8	8	7	8
-14	7	7	8	8	8	8	8	9	9	8	7	7	8	9	8	8	8	9	8	8	8	8	7	8	9	8	8	7	8	8	8	9
-15	8	8	8	8	7	8	7	8	8	9	8	8	8	9	8	9	8	8	7	8	9	8	8	8	9	9	8	9	8	8	7	8
-16	8	8	8	8	8	8	8	7	9	8	9	8	9	8	7	8	8	7	8	8	8	8	8	7	8	9	8	8	8	7	8	8
-17	8	8	8	8	8	7	8	8	8	8	7	8	8	8	9	8	8	9	7	8	8	8	8	7	8	8	8	7	8	8	9	8
-18	8	8	8	7	8	9	9	7	8	8	8	8	8	9	8	8	8	8	8	8	8	8	8	8	9	8	8	8	9	7	8	8
-19	8	8	9	7	8	8	8	8	8	8	8	7	7	9	8	8	7	9	7	8	8	9	9	8	9	8	8	8	8	8	8	8
-20	8	8	9	8	8	8	7	7	8	8	8	7	7	8	8	8	9	8	8	8	8	7	9	8	8	8	8	8	7	7	8	8

Each page displays phi horizontally by eta vertically. There are two pages for each portion, corresponding to +/- eta. Thus there are four pages, which are selected by tabs at the top of the GUI.

<u>Functions and Options</u>

In the IIA system, approximately every 5 seconds the TCC displays TT data for eight consecutive ticks. For Run IIB, the L1 Cal TCC will be able to serve an entire turns' worth of data, 159 successive ticks or 36 populated bunch crossings. The GUI is not a full out diagnostic tool to catch every single problem: an intermittent issue would probably not be caught once, let alone twice. This program can, however, see repetitive issues.

It is worthwhile to note the stage at which the L1 Cal TCC reads data. This data is acquired in parallel to the information being sent to the TAB from the ADF; the data seen here is the same data that the TAB sees.

There are two running modes for the GUI. The default is to automatically ask for a new set of data every x seconds, presumably however often the L1 Cal TCC gets information form the ADFs. This is called (as of yet) "fresh" mode. The other running mode (stale) is to get a new set of data whenever the user requests it. In fresh mode, the button "Freeze GUI data" will stop the automatic requests – it will cut the connection. The button "get new Data" will reconnect in stale mode. The connections menu has three options: cut the connection (the same as o freezing data), connect in fresh mode, or connect in stale mode. If a shifter/user sees a problem, they can still view the data in its various forms even after the connection is cut.

There are many options to consider for analyzing and presenting the TT data. In theory, any number of options can be programmed in, and they can all be user-specified. In reality, we want (as always) to spend the most time programming the most important options.

The current display shows all energies for a given timeslice (i.e. all of the energies in all TT's for a given bunch crossing). The user can specify any of the timeslices to look it, eight in this case. The display can also be in terms of an average, either mode or mean. In the case of a mean, rather than seeing a display pointing you towards a jet or other particles, this gives you longer term information. It could point to a noisy TT, for instance: if a given TT has a mean of 9 (instead of 8) ADC counts over all timeslices over several messages, it could signify accounted for noise. Another choice here would be to display the mode instead of the mean. For any TT, this should be 8 ADC counts (the pedestal). If it is 9 or 7, there could be a consistent miscalibration.

Other ideas include:

-- instead of displaying all values for a given timeslice, display for each TT the max (or min) value among the eight time slices. Again, instead of seeing a display pointing you towards particles, this could give a better idea of long-term activity, like noise level fluctuation.

-- display values in GeV instead of ADC counts.

-- ideas?

<u>Color Displays</u>

Another aspect of the display to consider is color levels. The program follows a similar setup to the color displays in the control room. The basic ideas is that a different

background color shows up for a given value of text, in this case energy.
As of now, 8 is set to green , slight variations are set to yellow, larger to red, and huge to purple. In terms of taking an average (as a mode), the levels could be green for 8, yellow for 7 or 9, and red for anything else, or higher tolerances for mean.

As a whole, there is a question of how closely the GUI should emulate the sliding windows algorithms. We could emulate the algorithms very closely, with a red display as the center of a jet, and yellow as its surrounding 5x5 (in eta-phi space) region. The question here is whether or not it would be useful to do this; this would be a non-trivial task. We have the basics, the question is now what we should do with it.

TAB/GAB Error Readout

As evidenced by the lack of an image, this GUI is still in it's planning stages. Each TAB has 10 sliding windows algorithm chips and a global chip, and each of these chips has a status register. The four LVDS chips on the GAB and the S-30 on the GAB each have status registers as well. Thus there are four levels of errors to monitor, comprising 93 chips. Each chip has a maximum of 16 bits.

The plan is to have a master "something bad happened" display that blinks red or does a jig to get the shifters attention. This would be set off by any of the error bits in any of the chips getting set to high; it is an "or" of all of the errors. As a whole, this GUI is going to be mostly color-based. It can only output a 0 or a 1, so 0's will correspond to green and 1's will correspond to red. In the cases where the readout is not in terms of error but rather in terms of something good, e.g. the bit on the TAB's S-30 which reads running as 1, they will be "not-ed" before "or-ed".

The precise layout is not yet defined. In the end, the main page will not have every single bit shown, but will display "or"s based on sensible choices. For example, the main page will probably show the component/chip on the x-axis and the bit on the y. It will probably display all 5 GAB chips. It will probably have two columns for each TAB – one corresponding to the global chip, one corresponding to an "or" of the appropriate bit on each chip within the TAB. Thus on the main page, bit 1 on TAB 3 displays an "or" of the bit 1 on each sliding windows chip within TAB 3. By clicking on this or-of-bits, it open a pop-up window. This window will show all of the sliding windows chips on the TAB (x-axis) and all of their bits (y-axis). It will also show the TAB's global chip. One of the main advantages of this setup is the ability to see the big picture. Multiple TAB pop-ups can be opened, so a shifter/user can do a TAB-to-TAB comparison to shine light on the cause of a problem.

Technical details regarding how this GUI will get the information from the status registers is yet to be confirmed. Presumably the fiber-optics input/output from the VI Master (now in the readout crate, hooked into tbilisi) – or a similar two-way channel – will be hooked into the TCC, and the TCC will serve the data to this GUI.