Index Panel

SLC Control Program

INDEX

SLAC's Software Engineering Newsletter

February 26, 1992 **All that Fits is News to Print** Vol. 6, No. 3

MAJOR fast feedback release

February 29, 1992

Author: *Fast feedback group* **Subsystem:** *Fast feedback* **User Impact:** *Large*
Panel Changes: *Many* **Documents:** *Yes* **Help File:** *Yes*

We have just released a major new version (2.0) of the fast feedback software. This implements *adaptive cascaded feedback* which is designed to keep all the Linac loops from reacting to an upstream change and hence overshooting. In the past we controlled this problem by using very low gains for the Linac loops. With the new software the gains can be raised back up to 1.0 and hence feedback will respond more quickly. This new software represents an addition of 2 man-years of effort to the original 7 man-year project.

To the casual user, feedback's operation has changed very little. It will just perform better. The methods of checking its status, turning it on and off and looking at ring buffers which show how well it is controlling the beam are unchanged. However deep down, quite a bit has changed. It is useful to understand the changes that have been made and the reasons for them.

Cascading

The fast feedback system was implemented in 1990-91. As of August 1991 there were 17 fast feedback loops running using the "new" software. There are two other loops (FB31 and FB69) running "old" software which will be converted to use the new software during the present run. Each of these loops runs on one to three 80386 microprocessors. Loops which use more than one micro have KISNET communications links connecting the micros. The loops typically measure and control the beam at a rate of 20 Hz.

Previously each of these loops operated completely independently from the other loops. There were no communications links between them. However the beam does carry information from one loop to the next.

This is best explained with an example. Suppose we have launch loops (which control the beam positions and angles) at sectors 4 and 6 of the Linac. Suppose the operator suddenly changes an x corrector in sector 3 and nothing else changes. On the following beam pulse, both loops would see that the position and/or angle of the beam changed. Without the cascading explained below, both loops would try to correct the problem by changing their correctors and would temporarily over-compensate.

The basic method of solution involves having each loop send its current measurements of its states (position and angle of the beam) to the immediately downstream loop. This information is known as the cascade vector. The downstream loop multiplies this cascade vector by a matrix made up of transfer matrix elements (from the online accelerator model or measured adaptively) to transform the position and angles from the location of the upstream loop to the location of the downstream loop. This transformed cascade-in vector is then effectively used as extra measurements by the downstream loop. Now the downstream loop is setup to control (*i.e.* try to keep at the set-point) the difference between the states as measured with its own BPMs and the transformed states of the cascade-in vector. **In this way each loop only tries to correct beam disturbances which occur between it and the immediately upstream loop.**

In short, cascading involves the addition of a KISNET communications link from each loop to the immediately downstream loop. This allows the downstream loop to know what the upstream loop is doing so they don't both correct the same beam disturbance and thus overshoot.

Adaption

For cascading to work well, the transfer matrix, T_c, from the upstream to the downstream loop must be known. In this section we consider how accurately this matrix must be known and how to determine it.

The T_c matrix can be easily derived from the model. However this is likely to be inaccurate as there are typically five sectors between adjacent feedback loops and over so many sectors the error in phase advance due to errors in the energy profile are considerable.

Given the relatively long distances between loops, it is necessary to measure T_c directly on the accelerator. This is done with an adaptive algorithm (a one layer neural net). Basically the adaptive algorithm uses the natural movement of the beam caused by things like poorly regulating power supplies or vibrating magnets to measure the transport matrix. We know the beam moves because if it didn't we wouldn't need feedback. It uses the correlations between changes of the states (position and angle) of the up and down-stream loops to figure the transport matrix. It gradually updates its estimate of the transport matrix using this data. This adaption process takes place whenever there is beam. In this way if the accelerator changes (due to a klystron tripping off, or a change of beam intensity), the new transport matrix will gradually be learned. We expect the learning time to be a few minutes and the accuracy of the matrix elements to be 10% to 20%.

Changes to the User Interface

Following is a brief description of the most important changes to the user interface. We don't describe every button, that is left for the HELP files.

- For loops which are cascaded, the ring buffer plot of a state (*e.g.* position) as a function of time now shows the *adjusted state* which has been adjusted with information from the cascade vector of the upstream loop. That is, **it shows how much that state has been perturbed by things between this loop and the one immediately upstream.**
- New ring buffers and canned plots have been created which show the old unadjusted states, that is the absolute position and angle of the beam at the feedback loop's fit point. These are available from the canned plot panel.

- In the past the ring buffers of states showed information which had been filtered by the feedback loop. They now show nearly raw data with no time filtering (or anticipation of the future effects of corrector changes). A simple least squares fit is done to the raw BPM data to get the state values.
- Cascaded loops now have two gains for which you will be prompted when you press the GAIN button. The first one is used when cascade is operational. The second is used when cascade is non-operational for any reason (turned off by an operator, broken KISNET link, upstream loop's correctors are max'ed out *etc*). Normally this second gain is smaller than the first to minimize the overshoot which occurs without cascading.
- The feedback summary displays show whether cascade is on or off.
- There is a new panel accessed via the [CASCAD +CALIB PANEL] button on the main FEEDBACK SELECT PANEL. From this panel one can control the state of cascade and do normal (single loop) calibrations. One can turn cascade or adaption on and off. They should both normally be on. If cascade misbehaves badly, then it can be turned off. In this case one should usually leave adaption on so it can keep tracking the changes of the transport matrix. Some parameters used by adaption (such as the learning rate) can be set. These parameters should only be tweaked by feedback experts. We expect that it will not be necessary to adjust them after the initial commissioning. Finally, the present value of the transport matrix calculated by adaption can be saved and restored to configurations. This provides a recovery method if the system badly misbehaves. HELP gives the details about all of this.
- On the STATE panel, more states are listed. There are entries for the adjusted and unadjusted states. You are only allowed to change setpoints for the adjusted states (as that is what the loop is actually controlling). History plots are available for all of the states.

More Information

For a more thorough explanation about how all of this works, ask your friendly controls software group secretary for the following documents. They provide 36 pages of bed-time reading.

-- `Cascaded Fast Feedback Top Level Design`

-- `Requirements and Design Overview for Adaptive Cascaded Feedback`

Analog Status Configurations

February 13, 1992

Author: *Daniel Van Olst* **Subsystem:** *Configurations* **User Impact:** *Moderate*
Panel Changes: *Few* **Documents:** *No* **Help File:** *Yes*

The configuration facility has been expanded to allow the saving of analog status channel information. Information is organized on a per-channel basis for easy use. Also, the analog status display software has been enhanced to show the difference between actual and configuration values.

Analog status configuration regions are selected on their own panel, off the main configuration panel. There are about a dozen regions, one for each significant region of the machine. Each of these regions includes every analog status channel in their area (with the exception of '   ', 'SPARE ', 'NONENONE' and 'USED ').

Two kinds of saves can be performed for analog status items: ACT, which saves scaled data for the channels; and CON, which saves the last loaded values for the channels. Saving from DES is not allowed for analog status channels.

Loading an analog status configuration moves the values in the configuration file to the database into the new secondary DCON.

The analog status displays now have an additional mode which make use of this. In addition to ABS and REF, a user may now toggle through to the CON mode. This causes the analog status displays show the value (scaled data)-(loaded configuration value).

Activating an analog status configuration has no meaning and is not permitted.

The configuration file display is slightly different for analog devices compared to timing and general database items. Items are green if they are within 5% of the file value. If they are more than 5% above the file value, they are magenta; and if more than 5% below they are red.

Since analog status items may vary considerably with beam rate, configuration saves for regions on the analog configuration panel will also save all the non-zero beam rates at the time of the save. (Note that these are the beam rates at one moment in time; the beam rates shortly before or after this might be considerably different. Use with caution.)

CATER V4.2

February 20, 1992

Author: *Robert C. Sass* **Subsystem:** *CATER* **User Impact:** *Minor*
Panel Changes: *None* **Documents:** *No* **Help File:** *Yes*

The following changes have been implemented for V4.2 of CATER:

1. CATER now presents the same printer list options for all reports: Compose-it-yourself and predefined. This list now includes the LWMOSG and LWVAC Appletalk printers in the MOSG and VAC shops respectively. Internal changes have been made so that additional printers can be added fairly easily.
2. There is now consistent handling of automatic user name insertion for all cases where a user's initials can be entered. This includes both Hardware and Software problem reports, solution entries, problem and solution modification, and problem closeout.
3. There is an expanded micro validation list for the positron area.
4. Tabs instead of spaces are used as delimiters for the export format. This should make it easier to upload CATER information into spreadsheets and other PC tools.

FIFO Device Support

February 13, 1992

Author: *T. Gromme, S. Levitt* **Subsystem:** *micro and SCP* **User Impact:** *Small*
Panel Changes: *Some* **Documents:** *None* **Help File:** *None*

Code has been added to the micro and the SCP to support the Bi Ra 6032 camac-to-camac communication module, known in the database as primary FIFO. Note that although this device is in principle bidirectional, we use it only for receiving beam-synchronous data from experiment-based computers such as PC's or MAC's. Data received through this device (the first instance of which is in

CA13) is now available in BPM displays and buffered data acquisitions. On the BPM "measurement definition" panel, the sequence through which the "device" button toggles now includes FIFO. The "buffered acquisition" text display has been altered slightly to accommodate the large number of channels that FIFOs can have (a maximum of 32). To keep the display tidy, instead of trying to display *all* the channels and running over several lines, only a group of channels is displayed. This selection is based on the value of the [TOGGLE DATA A] button. The display shows a few channels before and after the selected channel. The other channels may be displayed by hitting the toggle button to move the selection along.

Summary Displays

February 11, 1992

Author: *Ralph Johnson* **Subsystem:** *Any* **User Impact:** *Some*
Panel Changes: *None* **Documents:** *No* **Help File:** *No*

The source files which define the SCP summary displays (dcxp displays) and those files which define the summaries for the SIP process (SIP files) can now have "ALL*" specifications for micros, units, and channels. During compilation of these files, the expanded list of all devices is saved in separate diagnostic files. These files are in the "SIP_DIAG:" and "DSP_DIAG:" directories and have file extensions of ".TEMP".

PNET/CAR counters display

February 13, 1992

Author: *Tony Gromme* **Subsystem:** *micro and SCP* **User Impact:** *Small*
Panel Changes: *None* **Documents:** *None* **Help File:** *None*

Code supporting the "CAR counters" display (found on the network index panel) has been changed so that for each micro, the displayed counters count messages received from either the new PNET Multibus module or the old CAR Camac module, whichever is in use by the given micro.