Index Panel

SLC Control Program

SLC INDEX

SLAC's Software Engineering Newsletter

October 18, 1995 | **All that Fits is News to Print** | Vol. 9, No. 7

SLC and PEP-II INDEX Touch Panels

October 9, 1995

Author: *Kenneth Underwood* **Subsystem:** *All* **User Impact:** *Some*
Panel Changes: *Some* **Documents:** *No* **Help File:** *No*

Introduction

A request from the PEP-II project for a separate INDEX touch panel which would exclude many non PEP-II touch panels precipitated a fundamental change to the way touch panels are selected. The reason for the change can best be shown by an example.

Every touch panel has (or should have) a button to select the INDEX touch panel. Under the old scheme there could not be two INDEX touch panels so one would have to be named differently. But there are a large number of touch panels that could be used with either the SLC or PEP-II INDEX touch panels. If the [INDEX] button is pushed on an arbitrary touch panel, just how would the desired INDEX touch panel be selected?

To avoid unnecessary duplication an arbitrary touch panel should be able to select an SLC or PEP-II INDEX touch panel depending upon the current machine operation. There are a few other touch panels such as the BPMSTART touch panel that also fall into this catagory, so the implementation should support any arbitrary touch panel name besides INDEX.

Local Machine Mode

The implementation that is used to solve the ambiguous touch panel name problem starts with the local machine mode which is unique to each SCP. The local machine mode is initialized on SCP startup from the global machine mode, but can be changed as required for the desired machine operation. Some of the SCP behavior from CUD displays to feedback loops is based upon the machine mode. The local machine mode can be changed with the multi-state button [MACHIN MODE] found on the DISPLY touch panel located just off the SLC and PEP-II INDEX touch panels.

To select the desired touch panel the local machine mode is first translated into a touch panel type. This touch panel type is then used along with the current touch panel name and touch panel mode to select the correct touch panel appropriate to the current mode of operation.

Touch Panel Type

The touch panel type determines which touch panels can be selected. More than one local machine modes may translate into a single touch panel type. For example the SLC, POL, SLC10PPS, and POL10PPS local machine modes are all translated into the SLC touch panel type, while the PEP local machine mode is translated into the PEP2 touch panel type.

The touch panel type is then used in conjunction with the selected touch panel name to determine which of the possibly several touch panels to display. The actual touch panel that is displayed also depends upon the touch panel mode that is currently in effect.

Touch Panel Name

The touch panel name determines which touch panel file is used to create the touch panel screen. For example the INDEX touch panel name previously referred to the INDEX.PNL touch panel file. This file would then be used (usually a compiled version) to create the buttons and text you see on the touch panel screen.

Now the touch panel file name is prefixed with the touch panel type that is currently selected. These types include SLC, PEP2, and NLCTA. For example the PEP-II INDEX touch panel file name is $PEP2$INDEX and the NLCTA INDEX touch panel file name is $NLCTA$INDEX. However, you will find no touch panel of the file name SLCINDEX because the universal(see below) INDEX touch panel file name INDEX.PNL (without a prefix) will be used whenever the type specific version does not exist.

Touch Panel Mode

The touch panel mode determines the way touch panels are accessed. The touch panel mode is set by the multi-step button [CHANGE PANEL MODE] found on the DEVPANEL located just off the SLC and PEP-II INDEX touch panels. The current states are "NORMAL", "NO CACHE", or "USE.PNL".

If the current touch panel mode is "USE .PNL" then the SLCPNL logical name directory list will be searched for the selected touch panel file name. Otherwise, only compiled touch panels will be searched for in the touch panel database using the touch panel name and touch panel type as the key.

This can create some inconsistencies (this has always been true) between the external touch panel files and the internal compiled touch panels depending upon the current touch panel mode. Any reports of touch panel problems should include the current touch panel mode, touch panel name, and touch panel type in order to determine the cause of the problem.

Universal Touch Panels

Universal touch panels are simply touch panels that have no touch panel type. If the requested touch panel does not exist with the current touch panel type, then the universal touch panel with the same touch panel name will be used, if it exists. If neither the correct type nor the universal type touch panel exists, an error message will be issued and the current touch panel will remain selected.

Touch Panel Help Display

The touch panel HELP display shows the correct touch panel file name for the currently selected touch panel name and type for use with the EDITPNL touch panel editing facility. The touch panel file name includes the touch panel type as a prefix, if the file currently exists, otherwise the file name will be the universal touch panel file name. For example if the BPMSTART touch panel is currently selected with the PEP2 touch panel type, the HELP display would show the $PEP2$BPMSTART.PNL file name. However, the NTWKINDX touch panel is a universal touch panel, so the HELP display instead shows the NTWKINDX.PNL file name regardless of the touch panel type.

Changing Touch Panel Types

The touch panel type and hence the set of touch panels that can be accessed may be changed in one of two ways. The first involves explicitly changing the local machine mode on the DISPLY touch panel. The second way uses a button push to change the machine mode which then reselects the current touch panel. If you are interested in details, see the implementation behind the [SLC INDEX] and [PEP-II INDEX] buttons on the PEP-II and SLC INDEX touch panels, respectively.

Button Macros

The button macro facility is also touch panel type aware. The button macro directory display as well as direct button macro execution from a touch panel button now depends not only on the touch panel name, but also the touch panel type. There are also universal button macros that do not depend upon the touch panel type, so that they can be displayed and executed regardless of the touch panel type.

However, just because a button macro may be available for execution does not imply that it will run without errors. Most of the old button macros would probably work under the SLC and PEP2 touch panel types, so they were converted to be universal button macros. However, most of these same button macros will probably not execute without errors (if at all) under the NLCTA touch panel type.

When a new button macro is recorded, the user will be prompted for a button macro name and the author's name as before, but now the user will also be asked whether the new button macro is a universal button macro. The new button macro name must be unique within the current touch panel type.

The button macro TPU file name includes the touch panel type as a prefix. For example a button macro named CAMAC_STATUS that was recorded under the PEP2 touch panel type would have a file name $PEP2$CAMAC_STATUS.TPU unless it was declared a universal button macro in which case its name is just CAMAC_STATUS.TPU. Another button macro with the same CAMAC_STATUS button macro name can exist under the SLC touch panel type with a SLCCAMAC_STATUS.TPU file name. Additionally, a universal button macro with a CAMAC_STATUS.TPU file name can exist which will be available for touch panels types other than PEP2 or SLC.

Low-rate beamcodes

October 13, 1995

Author: *Tony Gromme* **Subsystem:** *SCP & MPG* **User Impact:** *Some*
Panel Changes: *One* **Documents:** *No* **Help File:** *Yes*

The following three changes are intended to make it easier to divert a low-rate subset of a production beam for PEP-II bypass line commissioning.

It is now possible to give a beamcode a broadcast pattern of less than 10 Hz. When this is done, KLYS tmasks will be tested against a tmask for that beamcode which is a minimal 10 Hz superset of the beamcode's pattern. For example, if a beamcode has pattern (1H359L), 1 Hz on timeslot 1, then KLYS tmasks will be tested as though the beamcode's pattern were (1H35L), 10 Hz on timeslot 1.

It is now possible to give overlapping patterns to beamcodes in a BGRP, and to specify the order in which beamcodes appear in a BGRP. For example, if beamcode 1 has pattern (1H2L), 120 Hz on timeslots 1 and 4, and in the same BGRP beamcode 8 has pattern (1H359L), 1 Hz on timeslot 1, and beamcode 8 appears earlier in the BGRP display than beamcode 1, then beamcode 8 will "steal" 1 Hz from beamcode 1's 120 Hz, leaving beamcode 1 actually being broadcast at 119 Hz. The MPG still will not permit a BGRP to be activated if the BGRP to be activated has any beamcodes in common with already active BGRP's, or if beamcode patterns in the BGRP to be activated would overlap any beamcode patterns in already active BGRP's.

There is now a button on the BGRP control panel to stop or restart the broadcasting of any given beamcode "on the fly", i.e., without necessarily deactivating or reactivating any BGRP. The "on/off" state of any beamcode is kept in the database in primary BEAM secondary HSTA. The "on/off" state of any beamcode will thus be preserved across deactivation/activation of BGRP's, across IPL'ing the MPG, and across DBINSTALL. The display of names of BGRP's now also shows a list of those beamcodes that are HSTA'ed off but belong to currently active BGRP's. The display of any one BGRP will show the pattern of any beamcode that is HSTA'ed off (and patterns of modifiers belonging to that beamcode) as 0 Hz, in cyan. It is not presently possible to see a display of a BGRP with all its beamcodes on without having those beamcodes actually HSTA'ed on, though of course this generally can be done on the SLC Vax.

FAST FEEDBACK CALIBRATION CHI-SQUARED DISPLAY

10 October 1995

Author: *P. Grossberg* **Subsystem:** *Fast Feedback* **User Impact:** *Small*

Panel Changes: *One* **Documents:** *No* **Help File:** *Yes*

A new fast feedback calibration display has been created which shows the chi-squared values for the most recent calibration. A new button to activate this display is on the FEEDBACK CALB/DIAG panel.

The chi-squared display format is similar to the calibration display with data shown for states versus actuators. Two chi-squared cut values have been defined for color-coding the displayed values: green chi-squareds are smaller than the smaller cut value; yellow chi-squareds lie between the two cut values; and red chi-squareds are greater than the greater cut value. The letters R and Y appear after red/yellow values for clarity on black and white monitors/printers. Header information includes the total number of red/bad chi-squared values.

For compatibility with the new chi-squared display, the calibration display has been changed in minor ways: it now includes the letters R and Y beside red/yellow values; the number of bad chi-squared values appears in its header information; and calibration error estimates are being calculated for use by correlation plots (where error estimate = rms/number-of-good-measurements).

Correlation Plot Data Reload Fix

September 6, 1995

Author: *Ron Chestnut* **Subsystem:** *Correlation Plots* **User Impact:** *Small*
Panel Changes: *None* **Documents:** *No* **Help File:** *None*

The Correlation Plots facility has a little used and limited capability of saving not only button definitions, but data as well to a file and later restoring the data from a file. The buttons on the Correlations Plots Auxilliary Panel are $\boxed{\text{COPY TO FILE}}$ and $\boxed{\text{RELOAD FROM FILE}}$.

Under some conditions the restore function had caused SCP crashes. Now the SCP will complain about what it cannot correctly interpret and not crash. These complaints should now be limited to Correlation Plot buttons with expressions, which cannot be handled. The reload process will however continue; upon termination, the buttons with defined expressions will be blank. The message windows will contain information on what was skipped. There is also some rudimentary checking done that the file has the correct contents, and is not some other data file.

Do you understand PICS thresholds?

Oct 9, 1995

Author: *Karey Krauter* **Subsystem:** *PICS* **User Impact:** *None*
Panel Changes: *None* **Documents:** *None* **Help File:** *No*

Ken Underwood and I had an enlightening discussion a while ago that I thought would benefit anyone else who was unclear on what the meanings are of the various PICS thresholds. The discussion began when I asked Ken about a phone call I received. The caller was worried about a PICS LIMIT_LO trip occurring without a corresponding response from the new MPS system to limit the beam to LIMIT_LO. Myth number one that I've had to have explained to me three times: PICS threshold trips do not mean the beam is tripped to the correspondingly named beam rate.

First, the naming of the PICS thresholds is really a misnomer. These names were chosen for brevity. Here is what the names really should be, if we were to rename them and we all were expert typists:

Current Name	What it really means
FULLRATE	ABOVE_THIS_FORCES_TRIP_TO_LOWER_RATE
LIMIT_LO	BELOW_THIS_PERMITS_RTN_TO_LIMIT_HI
LIMIT_HI	BELOW_THIS_PERMITS_RTN_TO_FULLRATE

FULLRATE threshold. This is the max radiation allowed. If the radiation level detected by the ion chamber is more than this threshold, then the beam frequency must be tripped to the frequency next lower than the currently running frequency. Another good name for this threshold would be TRIP_THRESHOLD since this is the only threshold that dictates when rate should be *lowered*. In other words, if the PICS FULLRATE threshold trips ...

... and the beam rate is FULLRATE then lower it to LIMIT_HI.
... and the beam rate is LIMIT_HI then lower it to LIMIT_LO.
... and the beam rate is LIMIT_LO then lower it to ZERORATE.

LIMIT_LO threshold. If the the radiation level detected by the ion chamber is less than this threshold, then the beam frequency may be permitted to return to LIMIT_HI. This threshold is typically set to be 1/10th of the max radiation allowed. The factor of 1/10 is used because the max radiation level tolerated at the LIMIT_LO (1Hz) beam rate is 1/10th the current of the LIMIT_HI (10Hz) beam rate.

LIMIT_HI threshold. If the radiation level detected by the ion chamber is less than this threshold, then the beam frequency may be permitted to return to FULLRATE. Note that the LIMIT_LO and LIMIT_HI thresholds are the opposite of the FULLRATE threshold in function: these dictate when rate can be *raised*. If the max beam rate is 120Hz, then this threshold is typically set to be 1/12th of the max radiation allowed. If the max beam rate is 30Hz, then this threshold is set to 1/3rd of the max.

For running at 120Hz, the factor of 1/12 is used because the max radiation level tolerated at the LIMIT_HI (10Hz) beam rate is 1/12th the current of the FULLRATE (120Hz) beam rate. For running at 30Hz, the factor of 1/3 is used because the max radiation level tolerated at the LIMIT_HI (10Hz) beam rate is 1/13rd the current of the FULLRATE (30Hz) beam rate.

Let me try to draw a picture to illustrate this. Key turning points in the illustration are explained here.

```
/\          (3)                                                     /\
|            ***                                                     |
|............*.*................FULLRATE threshold, max allowed......|
|            * *                                                     |R
| (1)    (2) * *                                                     |A
|************* *           ***********************************       |D
|              *           * (7)                                     |I
|              *           *                                         |A
|              *           *                                         |T
|              *           *                                         |I
|              *           *                                         |O
|..............*...........*....LIMIT_LO threshold, 1/10th of max....|N
|              *           *                                         |
|              * (4)       *                                         |L
|              *********** *                                         |E
|                    (5) * *                                         |V
|........................*.*....LIMIT_HI threshold, 1/12th of max....|E
|                        ***                                         |L
|                        (6)                                         |
|                                                                    |
+--------------------------------------------------------------------+
TIME ->
```

1. Here is the beam running at 120Hz normally and correctly, with no problems. It is registering a radiation level less than the FULLRATE threshold.

2. Here the beam has had a problem. As a result of this problem, its radiation level elevates to above the FULLRATE threshold. (Unfortunately the illustration shows instantaneous radiation level changes: it really is more gradual of course.)

3. Once the radiation level exceeds the FULLRATE threshold, the new MPS system detects the PICS-FULLRATE trip and drops the beam rate to the next lower frequency of 10Hz. The radiation level begins to lower as a result.

4. At the lower 10Hz beam rate, the radiation level lowers and stabilizes at a new lower level.

5. Here the problem which elevated the radiation level in the first place has been corrected. The radiation level drops off even further in response to the correction.

6. Once the radiation level of the now-correct beam has decreased to less than the PICS-LIMIT_HI threshold, then the beam rate is permitted to return to 120Hz. Its radiation level rises in response to the rise in beam frequency.

7. At the higher 120Hz beam rate, the radiation level rises and stabilizes at its original higher level (but beneath the FULLRATE threshold).

One other point of clarification in the use of PICS's. The little red light on the front panel of the CAMAC card itself? It isn't used, so it is never reset. It may be ignored.

Questions? Ken Underwood has talks about this stuff in his sleep.

New Scp

October 11, 1995

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

There is an upgraded version of the SCP which has now been released. It provides:

1. Keyboard typeahead. You can type dialog input prior to pushing a button to popup the dialog or while it is being popped up.

2. Mouse clickahead. You can queue mouse button pushes while SCP actions are active. They will be processed in order when the current action becomes inactive.

3. Control-X cancel. You can type control-X at anytime to clear all SCP keyboard and mouse input that may be pending. This does not interrupt currently executing code.

4. Control-C cancel. As before, you can type control-C at anytime to cancel or stop current application code execution. This works if the particular application code looks for a control-c having been pushed. Now the control-c also clears all SCP keyboard and mouse input that may be pending as does control-x.

5. Xwindow server errors. Most, if not all, preveously occuring xwindow errors have been eliminated. There are, of course, times when errors may still occur as when there are network problems. However, the regularly occuring ones should no longer happen.

6. Dialog focus. When a dialog pops up it must receive input focus within the Xwindow system prior to being able to receive input. The related code in the SCP has been upgraded to make it more difficult to have any keyboard input missed.

In addition to the items discussed above which are user related, the upgrade adds features which are needed for Epics support and PC based knobs. It also improves our ability to maintain and debug SCP code.

System CUD Display Enhancements

October 16, 1995

Author: *Daniel Van Olst* **Subsystem:** *CUD* **User Impact:** *small*
Panel Changes: *none* **Documents:** *no* **Help File:** *yes*

The SYSTEM CUD display has been modified in order to provide more useful and accurate information on the health of the software and hardware that runs the control system.

The SLC PROCESSES list on the SYSTEM CUD display has been modified in the following ways:

- Only missing or suspended processes are displayed, due to the large number of required ancillary processes (those controlled by the WARMSLC command), and lack of room on the display.

- The SLC PROCESSES list now has the title BAD PROCESSES.

- The same processes are displayed as on the list at the top of the DISPLAY MICRO STATUS display on the NETWORK INDEX (except that processes shown in green on the DISPLAY MICRO STATUS display are not shown).

The PROGRAM SUMMARY list has been removed due to lack of usefulness. It has been replaced with the following information:

- Calf Slots Left: This is the number of additional CALF SCPs that may be started.

- SLCNET Poll Rate: Rate (in hz) at which the VAX is polling SLCNET micros.

- History Buffer Statuses: Normally these should display GOOD. If FAILED is displayed, email is automatically sent to History Buffer Experts. If FAILED is displayed at any other time than Software Controls' normal business hours, a History Buffer Expert should be contacted so that history buffer data is not permanently lost.

In addition, HELP has been added for the SYSTEM CUD display. See HELP for the SYSTEM DISPLAY button on the CUD panel, off the SPECIAL DISPLAYS panel.