Control System Software

**Author:** Spence Clark  
**Subsystem:** Controls Software  
**Panel Changes:** None  
**Documents:** No  
**User Impact:** None  
**Help File:** None

February 6, 1990

We are pleased to find so many individuals outside the Controls Software group interested in writing software for use in the SLC Control System. This collaboration can lead to a better control system and help to alleviate the backlog of software requests. It can also lead to greater understanding of the systems from the user point of view. This insight will aid in the specification of new software based on everyone’s enhanced perception of the complexities and subtleties of the software.

We need to put some procedures in place for implementing user-generated software in order to enhance overall effectiveness and to maintain system integrity. It is essential that the documentation and implementation are up to standards so that the Controls Software group can maintain the software in the future. We will be happy to work with you to explain how things work and to help you get your code running properly. We can achieve this if everyone uses the same set of code development procedures used in the Software group.

1. All software procedures need a specification of what the code should do and how it should interact with the user and the machine. This functionality should be supported by a detailed design which lays out how the new software will be implemented (e.g. new modules to be written, changes to existing modules, panels). Please feel free to come talk with us about this documentation, since the extent and detail required will scale with the complexity of the task.

2. We will review the requirements and the design and offer constructive suggestions on better implementation strategies, and guidance on where to find and how to use supporting software that already exists. We can also save you a great deal of time learning how to edit, compile, link, and test.

3. Read the Programmer’s Guide which describes the software development environment, including standard procedures and tools. Also read the Basic User’s Guide which will give you an introduction to the control system and the SCP environment and will explain subsystems, support subroutines and utilities. This is the core reference manual for the SLC control system and how to work with it. You can get more details or assistance from your favorite software engineer.

4. Study the Fortran and C software standards to see how the actual code should be structured and written. This is critical for the maintenance and modification of the code in the future. These documents are in flux, so ask for a current version.

5. As the last phase of testing, a “walk through” should be arranged with one of the software supervisors.
to verify functionality and look for possible improvements to the user interface. When the software is ready for release, contact Tanya Boysen to arrange for Code Review. One of the programmers will go over the code to check for adherence to standards and look for potential problems. The librarian for the appropriate subsection of the SCP can then put the new code into CMS and release it into production.

Again, we are pleased to see user interest in writing code, and will do whatever we can to help make it a useful and successful experience. Feel free to approach your favorite software engineer, or talk to Nancy Spencer or myself. Good luck.

Auto Ranging for Final Focus Quad Scans

February 3, 1990

Author: Nan Phinney
Subsystem: Final Focus
Panel Changes: None
Documents: No
User Impact: Small
Help File: None

The Final Focus tuning software for Quad scans has been enhanced to automatically select a range centered on the current value of BDES for the Quad to be scanned. The preselected number of steps and step size are used, and a new value of the scan start is calculated. The algorithm takes into account the instability of the supplies near zero and will shift the range by a fraction of the step size to avoid settings too close to zero. The range will be updated whenever the user Accepts Best Value or Enters a New Quad Value. It is also updated before displaying the range whenever the software detects that BDES has changed since the last time the range was displayed.

The user may override the automatic range selection by entering any range desired. This will remain in effect only until the Quad BDES changes, at which point a new start will be calculated by the auto range function (using the selected steps and step size). The default step size for SQ3 scans has also been changed to 0.5 as this was found to be more useful during the January run.

There are also some minor changes to the Accept Best Value algorithm for all Quad or Knob scans using deflections. A weighted average of the best value is calculated from the fits to both the electron and positron data, if both fits are good. In this release, the change will no longer be implemented if it is smaller than the errors.