

Ecal Six-Supermodule Heater Controls

Marc McMullen
2023-08

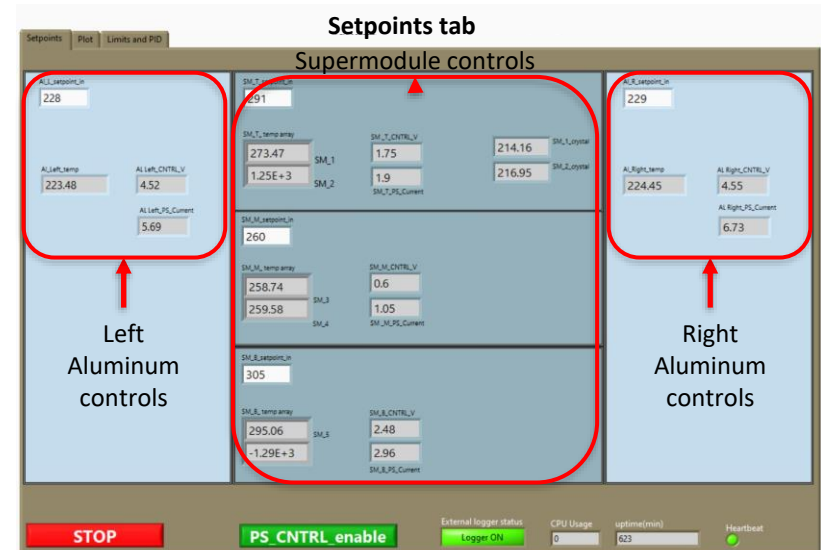
This month I completed and tested LabVIEW-based software to control the output of five supplies used to provide power to all the heaters of the ECal six-supermodule teststand. The user interface has three tabs. A setpoints tab to set the temperatures, a plot tab to observe the progress of the system, and a tab which has the PID limits and software limits.

The latest version of the software has five, PID-controlled channels. Three channels of supermodule control, each channel controls two supermodules. Each supermodule has two heaters, wired in parallel. The other two channels control a stack of three aluminum bars. Each bar has three heaters.

Initially, I used only two PID controls in the software. One for supermodules and the other for aluminum bars. In this configuration, each channel would cycle through the PID control individually. The PID would read the channel temperature and pass the control voltage for that channel to the supply. However, there seemed to be some crosstalk between channels using this method. So I added separate PID controls for each channel, for a total of five PID controls. This eliminated the crosstalk issue.

The basic functions of the software, channel control and monitoring, worked as expected. However, the software crashed three times in a period of four days while being run as a LabVIEW Real-Time application. The program is stable when the software is started remotely. Currently, the cause of the crashes is not known.

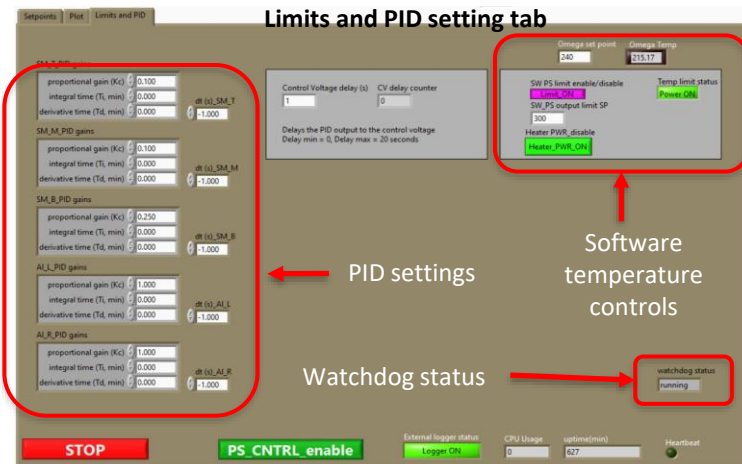
- Develop PID based controls for the ECal six-supermodule test stand
- Test and debug the controls system



EIC Beamline Teststand Sensor Upgrade

Due to the nature of the power supply, it is important that the system is in a safe condition if the controls are down. The supply naturally outputs maximum power when it does not sense a control voltage. For this reason, I added a watchdog loop and a safe state loop with it. The watchdog monitors the software and will reboot the cRIO if the software starts. The safe-state loop sets the control voltages to a low setpoint (0.6V) and opens the heater power relays so that no power can go to the heaters until the software reboot has completed and the user enables the heaters.

In addition to these upgrades, I also added configuration files for the temperature setpoints and PID settings. These update while the software is running if the temperature setpoint or PID values change. The latest values are loaded to the controls when the software boots.



In conclusion, I have developed and implemented LabVIEW-based software to control the temperature of the ECal six-supermodule teststand. The system works as expected. The software has redundant safety systems which will prevent over-temperature events, live monitoring, and datalogging. These features make the software suitable for the remote control of a heating system. This system has been tested and is scheduled to move to Hall A for in-beam testing in the coming weeks.

