Debugging NPS Thermal Readback LabVIEW Program

Aaron Brown
2022-06

The thermal readback program for Hall C's NPS runs as a background program on a cRIO. The cRIO has been installed in the SHMS detector hut in Hall C. When the program was run after installation, some new issues were noticed that had not presented during development in the EEL building.

The LabVIEW program is designed to collect sensor values from the Keysight mainframe and send them (along with the various limits associated with each signal type) to shared variables that are bound to EPICS process variables (PVs). The first issue noticed is that some of these PVs are not being communicated with by LabVIEW. The temperature values (which are of the double datatype) are being sent with no problem, but none of the Boolean signals are being sent to the shared variables.

- **After cRIO installation in Hall C noticed that some PVs were not receiving values from the thermal readback LabVIEW program**

- **Debugging in progress**

FIG.1. Screenshot of monitoring portion of front crystal zone temperatures Phoebus screen. Unlike the temperatures and averages, the Boolean status indicators are not being written to by LabVIEW and have a dashed border to indicate there is no assigned value.

Debugging NPS Thermal Readback LabVIEW Program

    The first step taken to resolve this issue was to ensure that the Boolean shared variables were not being overwritten in another section of the code. After verifying that this was not the case, the next step was to change the datatype of the Boolean EPICS PVs to double. Since the temperature signals (also of the double datatype) are being sent to EPICS with no problem we figured that this may be a solution.

    So far, the attempted solutions has resolved the issue. As it stands now, the temperature signals are the only ones being communicated from LabVIEW to EPICS. The next attempted solution will be to create a new LabVIEW EPICS client that only communicates with a small number (approximately four) of EPICS PVs. This will hopefully tell us if there is a problem with the current EPICS client library (the number of variables may be too high for the shared variable engine to handle all at once) or if there is a problem with shared variable datatypes.