Programmatically Adding PVs to RICH-II Hardware Interlock System’s EPICS Server

Tyler Lemon
2022-03

Programmatically Adding PVs to RICH-II Hardware Interlock System’s EPICS Server

I continued work on the LabVIEW program for the RICH-II hardware interlock system by developing code in LabVIEW to create an EPICS server with PVs based on a shared variable library in the program’s LabVIEW project.

After creating a CSV configuration using the Python program developed in February 2022 to create a shared variable library in the program’s LabVIEW project, additional code was developed to programmatically create an EPICS server with PVs linked to those shared variables, (figure on slide #2).

The additional code first uses a LabVIEW property to get all variables in the shared variable library containing the variables that will be converted to EPICS PVs (A in figure on slide #2). Next, defines a path on the sbRIO to deploy the EPICS server to (B), undeploys any EPICS server already running on the cRIO and creates an new EPICS server (C), iterates over every variable to create an initial placeholder PV (D), repeats iteration over all PVs (F) and then renames all PVs to follow established RICH PV naming conventions (G). Finally, the code deploys the EPICS server, allowing the program to read from and write to EPICS PVs (H).

A problem that was encountered was that LabVIEW uses a specific naming convention for PVs it creates that does not match established RICH conventions for PV names. The solution to this problem was to develop parts denoted by F and G , (figure on slide #2) to rename the PVs.

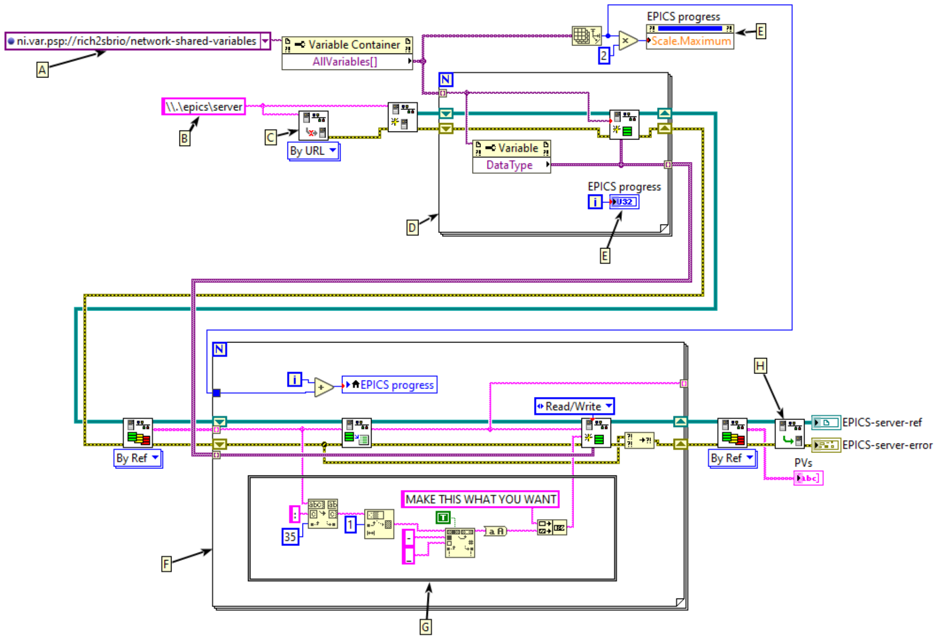
Another problem encountered was that while this section of code was running, it appeared that the LabVIEW program had frozen. This happens because the for-loops used to iterate over all variables (D and F) each must loop over the > 300 shared variables that will be converted to PVs. The solution to this was to add a front panel progress bar indicator to the overall LabVIEW program to indicate that the program not just frozen.

- **Created an EPICS server in RICH-II hardware interlock system’s LabVIEW project**

- **LabVIEW code developed to read variables from a LabVIEW shared variable library, to create and rename variables for EPICS PVs, and to save these EPICS PVs (renamed LabVIEW variables) and deploy them on to an EPICS server**

Programmatically Adding PVs to RICH-II Hardware Interlock System's EPICS Server

As of March 23, the software for this task is complete and has been tested. In the next month, the software will be used in a production environment to create the EPICS server for RICH-II's hardware interlock system during detector assembly.



LabVIEW block diagram developed to programmatically create EPICS server. For slide, code has been rearranged into a column to allow it to be big enough to see components. In actual program, code is one continuous row.