2015

Developing an EPICS IOC in LabVIEW

Kenneth Butler
Virginia Commonwealth University

David Shires
Virginia Commonwealth University

Michael Talbott
Virginia Commonwealth University

Team Members:
Kenneth Butler, David Shires, Michael Talbott
Faculty Advisor:
Preetam Ghosh
Sponsor:
Jefferson Labs
Sponsor Advisor:
Christiana Wilson

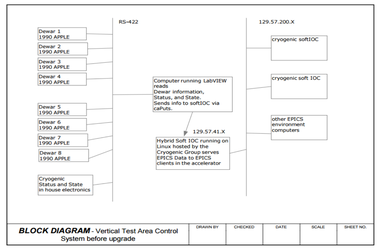
COMPUTER SCIENCE

Developing an EPICS IOC in LabVIEW

CAPSTONE DESIGN EXPO 2015

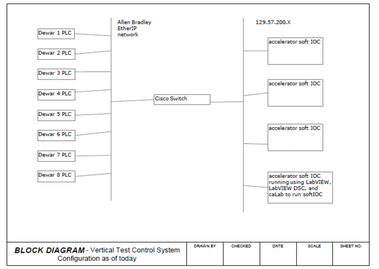
Block Diagrams

The goal of this project was to prove (or disprove) the ability of caLabs soft IOC, which is an interface between LabVIEW and EPICS, to serve data over a network.
The larger project this will be implemented in is the upgrade of the superconducting test facility, called the Vertical Test Area.
Jefferson Labs builds superconducting cavities for particle accelerators around the world. Once these superconducting cavities are built they need to be cooled with liquid helium and injected with high power radio frequency (RF).
The output power of the radio frequency is then measured to determine proper quality. There are 8 dewars that hold the liquid helium that is used to cool the cavities to the proper temperature.



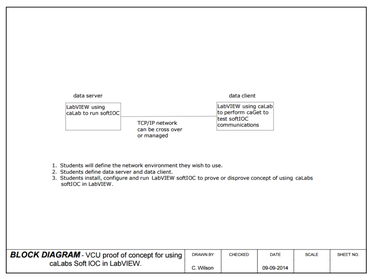
Original System

- Apple II computers read dewar data and relayed information to a computer running labVIEW
- This data is sent to a soft IOC running on Linux



Proposed System

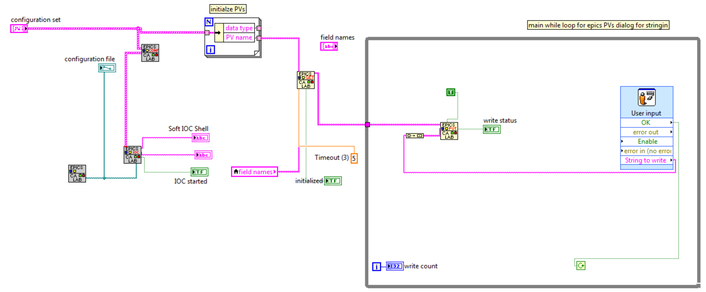
- Dewar information is read by Programmable Logic Controllers (PLCs) then relayed to a Cisco switch
- Data is then read using soft IOCs through labVIEW and caLabs



Our Task

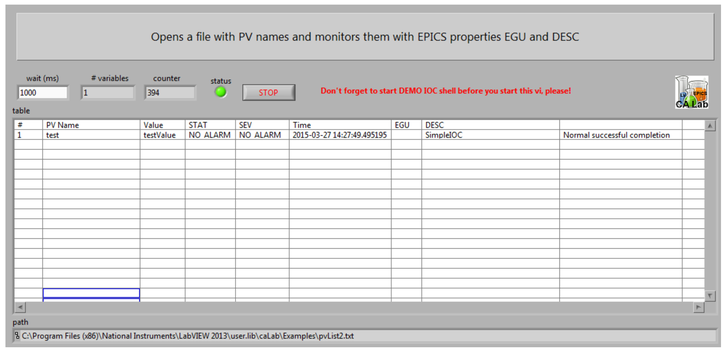
- Prove data can be passed from server to client over a TCP/IP connection using Windows LabVIEW and caLabs
- Test reliability of data transfer
- Develop UI for testing softIOCs

Proof of Concept



Initial Proof of Concept

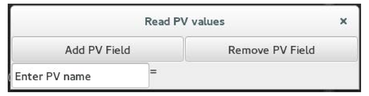
- This LabVIEW program acts as a softIOC server
- Process Variables (PVs) are given a name and a value (via string input) and broadcast over the network on port 5065



Initial Proof of Concept

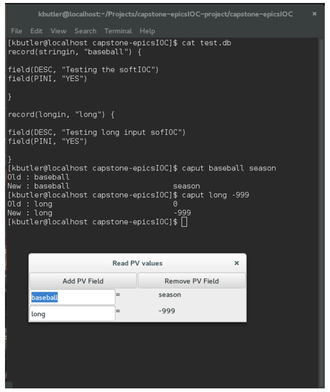
- This LabVIEW program acts as the softIOC client
- It searches the network for PV names given in list format
- Displays the PV values for the names listed (Test input displayed above)

UI Demo



User Interface

- Interface was designed in python using pyepics and wxpython libraries
- Can dynamically add or remove PV fields for monitoring PV values
- Takes PV name as string input (Enter PV name)
- Outputs PV value associated with that PV name



Demo

- This is an example of the UI reading two PVs and displaying their values
- When a PV value is changed the UI updates the value field accordingly
- (In the background) using the EPICS command line tools to set the values for the PVs

In Conclusion

Our concept has proven that LabVIEW, caLabs, and EPICS can be served over a Windows platform and over a network.

In the future, these findings will serve as a basis for Jefferson Labs to move forward in their implement this system on their network.

VCU School of Engineering

Make it real.

VIRGINIA COMMONWEALTH UNIVERSITY