DCS Control/Monitoring for Inner Pixel Integration

General Information

This page collects various DCS control/monitoring information that include LV/HV control and monitoring, and monitoring of temperature, pressure, humidity etc. for both Lab setups and B33 integration. Interlocks have a dedicated page separately.

Collection of specs for LV/HV PSU and associated infrastructure

B84 EPP Lab Setup

The B84 EPP lab DAQ readout and PSU communications are supported by the rack at the 'ATLAS corner', network switch and lab supplied on the test bench. The ITk DAQ Network page has detailed description of the readout network with the RDDEV111 host server and X display/terminal with fastx3. Key equipment that require remote power controls can take regular 110V A/C power from ports served by the TripLite PDU in the rack.



ISEG Control/Monitoring

Many PSUs and DCS monitoring components are from ISEG that are hosted in Wiener MPOD crates. An example control/monitoring of an ISEG HV module in the Wiener MPOD crate using SNMP in EPP lab from RDDEV111 through fastx3 on your laptop browser could look like this:

● ● ● 〈 〉 🔟							0 1 0	
🝐 🧮 My Sessions FastX	TastX StarNet Communications							+
sudong@fastx3:~ _ D	x iseg Control Version 1.1.7.0							_ 🗆 x
File Edit View Search Terminal Help	File System Module Channel	Help						
[sudong@fastx3 ~]\$ ssh rddev111 sudong@rddev111's password: Lost Jerster The Mer 10 0451-02 2021 form fortw2 alon storford adv	1 Modules found. Connected to	o E08F2 sn. 7	200223 at Slo	et 1				
Last togin: Thu Mar 11 04:51:05 2021 From Tastx5.stac.stanford.edu	Slot 1	Vset (V)	Vmeasure (V)	Vnominal (V)	Iset (mA)	neasure (mA)	nominal (mA)	Status CV CC
-hash-4.2\$	Channel 0	49.00	-49.00	-2,000.00	F 4.000	L -0.0000 µA	4.000	On
-bash-4.2% cd iseg/isegSnmpControl-SL71-x64-1.1.7.0/	Channel 1	0.00	-0.04	-2,000.00	F 4.000	L 0.0001 µA	4.000	Off
-bash-4.2\$ bash ./start.sh	Channel 2	0.00	-0.03	-2,000.00	Г 4.000	L 0.0001 µA	4.000	Off
Using Profile path: "/home/sudong"	Channel 3	0.00	-0.02	-2,000.00	Г 4.000	L 0.0001 µA	4.000	Off
"libWIENER_SNMP.so" loaded succesfully	Channel 4	0.00	-0.01	-2,000.00	r 4.000	L 0.0002 µA	4.000	Off
true true true	Channel 5	0.00	-0.01	-2,000.00	r 4.000	L 0.0003 µA	4.000	Off
true true true true true true true true	Channel 6	0.00	-0.01	-2,000.00	r 4.000	L -0.0000 µA	4.000	Off
true true true true true true	Channel 7	0.00	-0.05	-2,000.00	r 4.000	L 0.0002 µA	4.000	Off
true true true true true true								
true true true true true true								
true true true								
true true true true true								
true true true true true true true true								
true true true true true true true true								
*** Initialise SNMP ***								1
*** Translate OIDs ***	- Module Information		-Module Sup	olies and Temp	erature	Module S	ettings	
CNMP service for best #100 100 4 202# served	Serial Number: 7200223							
SNMP session for nost "192.168.4.203" opened	Firmura Name 50052		+ 5 Volt (V)	: 5.0		Voltage L	.imit (%):	103
refreshidble(channel): 20 ms	Firmware Name: E08F2		+ 24 Volt (V	0. 24.0		Current I	imit (%).	102
refreshable(chapped): 12 ms	Firmware Release: 6.73		+ 24 VOIL (V): 24.0			current	Currenc Linne (70). 105	
refreshTable(channel): 11 ms	Channel Number: 8		- 24 Volt (V)	: n/a		Voltage F	Ramp Speed (%): 1
refreshTable(channel): 12 ms	Device Class: 26							
refreshTable(channel): 12 ms			Temperatur	e (°C): 26.0		Current F	Ramp Speed (%): 50
refreshTable(channel): 12 ms								
refreshModule: 16 ms								
refreshTable(channel): 12 ms								
refreshTable(channel): 11 ms								
Sixkne	Module Summarv I Module C	ontrol Mo	dule Status	Module Event	Mask Ch	annei Control	Channel Sta	tus Channed]▶
sudong@fastx3:~ iseg Control Version 1.1.7.0								

The Wiener MpodC controller itself is always on for basic communication to the crate when the crate is powered. To access the ISEG modules in the crate, the red on/off button on the MpodC controller module need to be ON to bring up power to the rest of the crate to make the ISEG modules visible.

The initial setup of the control software from a session on RDDEV111:

- wget https://iseg-hv.com/download/SOFTWARE/isegSNMPcontrol/current/isegSnmpControl-SL71-x64-1.1.7.0.zip
- unzip isegSnmpControl-SL71-x64-1.1.7.0.zip
- (one initial step already done by Matthias for initial common shared setup was to drop WIENER-CRATE-MIB.txt into /usr/share/snmp/mibs/)

Then each time to run control/monitoring session:

- cd isegSnmpControl-SL71-x64-1.1.7.0/
- bash ./start.sh
- (First run may ask for module IP address in IP dialog).
- Simply click Vset boxes to set desired values and right click Status box to get pull down menu to turn on/off etc.

Raspberry Pi Based Control/Monitoring

Many small lab test setups and some part of the B33 integration setup are standardizing to the use of Raspberry Pi 4 to mediate the ethernet connection to the ITk DAQ network for communication and connecting to various types of sensors using dedicated shields. The control/monitoring program running on Raspberry Pi can also reach out through the ITk DAQ network to e.g. Agilent PSU through PROLOGIX ETH-GBIP adaptor for integrated control of module power. Details of the setup can be found at https://docs.google.com/document/d/1MMfeTJIQxq9gqNzj3Ho_N8L8SXRJY5iV2XjEkD1vXNc/edit? usp=sharing

blocked URL	blocked URL					
Widgetloads Pi-SPi-8Al+ readout interface is widely in many test setups. It support 4 types of signals 4-20mA/5V VDC/10V VDC/Temp (NTC) which can be easily reconfigured to host a combination of these by solder fill of the appropriate pads at the back of the card.						

Accessing Pi

to access the pi start by accessing rddev111 and then from there the pi via the commands.

ssh YOUR_USER_NAME@rddev111.slac.stanford.edu

ssh pi@raspi_cooling

(for sp chain testing the pi is at pi@ssc_raspi)

Interlock

Nothing should be needed to be done for the interlock. It should do everything on it's own. There is a script in /home/pi/dcs-interlock/Interlock titled restart_Interlock.sh that can be run if the interlock is tripped and doesn't automatically reset itself after a few minutes.

DCS

The DCS system is not quite finished it is however working in its current state.

To work the DCS access the pi as above then run the scripts in /home/pi/dcs-interlock/DCS

- To start the power supply
 - ./start_peltier.py -V 10 10 0 -A 1 1 0 -C 110

Where the three numbers after the V are the voltages and the three numbers after the A are the currents and then the three numbers after the C are the channels so 110 actives the first two channels and turns the third one off.

- ./start_module.py -V 10 -A 1.4
- ./stopDCS.py
 - Turns off all the channels

Note that this is working such that any computer with access to the Message broker on rddev111 can run this without access into the pi.

To add a new machine to the list you need to add the vhost and set permissions so on rddev111

```
sudo -u rabbitmq rabbitmqctl add_vhost {new_vhost_name}
sudo -u rabbitmq rabbitmqctl set_permissions -p "SP_chain" "Username" ".*" ".*"
```

Monitoring

Currently the monitoring is done via grafana. To access it start by forwarding the required server port to your local machine via the command

Then in any web browser on your machine go to

http://localhost:3000/d/0YxPBumnk/power-of-dcs_coolings?orgId=1&from=now-5m&to=now&refresh=5s

From here you can see all the current data from the cooling stand.

(for the sp chain testing the grafana page is http://localhost:3000/d/rbFcf6W7k/sp-chain-dashboard?orgId=1&refresh=5s)

The DCS setup for the 19-1 Ring in the QC box is located at: http://atlascr.slac.stanford.edu:3000/d/cokMRQnVz/p19-1-ring-dcs-interlock? orgld=1&refresh=5s

You can see NTC and differential Vmon values for the quad and triplets, the DCS readout status and counter, as well the raw values from the MOPS emulator channels (see https://docs.google.com/spreadsheets/d/1Br6QM8IWy_frLFVSGQFpPXXCCUxYvTwKuwFI2_wtjt8/edit#gid=0 for the channel DCS mapping). Readout is controlled by DCS_Script.py in pi@dcs-pi-interlock (raspberry) and is run via a cron job. The job can be temporarily suspended via

crontab -e (comment out the line)

pgrep -f DCS_Script.py

kill <PID>

୍ଷ	፡፡ B General / p19-1 Ring DCS/Interlock ☆		🗤 🛱 🕲 🔿 Las	t6hours × Q 것 5s × 무
Q	Quad a	DCS Read-out Counter		
88	23.5		 T1 Chipb Last *: 21.2 T2 Chip4 Last *: 21.2 	25800
Ø			Q1 Last *: 21.2	25400
¢			Q4 Last *: 21.1	25200
			Q5 Last *: 21.5	25000
	21.5		— Q7 Last *: 21.5	24800
			Q8 Last *: 21.2	24600
	21		- 010 Last *: 21.2	24400
	20.5		- CO2 output Last *: 23.5	24200
	09:30 10:00 10:30 11:00 11:30 12:00 12:30	13:00 13:30 14:00 14:30 15:00	— CO2 input Last *: 21.8	10:00 12:00 14:00 — DCS Read-out Counter
	Quad DeltaV	Triplet DeltaV ~		DCS Read-out Status
	0.0150			100
		0.100		
		0.0800		60
	0.00500	0.0600		
	 Caracter Schembergerung wird der Sterfanzeiten Andere zu Sterfanzeiten Sterfanzeiten Schembergerung von der Landeren zu Kruck zu eine Preise Schembergerung der Schembergerung zu eine Andere Sterfanzeiten Schembergerung von Schembergerung der Sche	0.0400		
63	warmer of the second	0.0200		
125	-0.0050			
Ū	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	10:00 11:00 12:00 13:00 — T1 ChipA — T1 ChipB — T1 ChipC — T2 ChipA — T3 ChipC	14:00 15:00 	10:00 12:00 14:00 DCS Read-out Status
8	Quad Vmon (unsubtracted)	π	plet Vmon (unsubtracted)	
?	0.0160	0.120		

Interlock build/connections/installation

Infomation about how the interlock and DCS is all hooked together can be found in the ICD document.



Installation instructions are on gitlab with all the source code at https://gitlab.cern.ch/nyoung/dcs-interlock

[please add a) instructions to connect rasPI and display monitoring info. b) Instruction to Program/compile/install monitoring programs to run on rasPi.]