

**71**,

b UNIVERSITÄT BERN

# **Light readout software and QA/QC** ArgonCube 2x2 Electronics and Readout Integrations, 25 August, 2022

**Alexander Selyunin** 



## Installed software

- ACD-SRV02 is used for software installation
- DHCP configured on srv02 on enp7s0f0 interface
- Docker container for ADC GUI software :
  - <u>ksu</u> (Root required to run the docker)
  - docker run -it --rm -p5900:5900 --network host quay.io/kozhukalov/adc64-gui
- VNC to access the ADC GUI from remote PC
  - ssh -L5900:localhost:11590 acdemo@acd-gw01.fnal.gov (ssh tunnel to gw01)
  - ssh -L11590:localhost:5900 acd-srv02 (ssh tunnel to srv02 from gw01)
  - VNC connection to localhost:5900
    - cd <u>build/adc64</u>
      - <u>./afi-adc64 (ADC oscilloscope gui)</u>
    - cd <u>build/adc64-system</u>
      - ./afi-adc64-system (data acquisition application)

	loc	ainost:5900 (acd-srv02.thal.gov:20) - VNC viewer							
<u>File Options Tools He</u>	elp								
File Options Tools He Trigger Timer Lemo(TTL) Threshold Rising	Readout Window ize 1024 atency 0	DSP MAF MAF selector BLC thr 0 Invert signal							
► Start Stop Write file Channels setup Octal ADC1 ▼ ✓ Show sparse markers									
1,000 800 600 400 200									
0	200	400 600	)						

ADCs need to be powered up and connected to the network switch in order to check that we can readout ADC!!!

Туре	Serial Slot	IP Address	t, °C	Event	Trig on XOff	Synch to ADC64	
4							
Run State	Main Log	Statistic					
🔘 Start	[11:47:59]	[thread:GUI thr	ead] Rem	oteControl	Server disconnect	ed: fsm -> Error	
Stop	[11:47:59] [11:47:59]	[thread:GUI thr [thread:GUI thr	ead] "Adc ead] "Adc	DeviceMgr DeviceMgr	Fsm" Request new Fsm" state change	v state: Error e Init to Error	
	[11:47:59]	[thread:GUI thr	ead] "Adc	DeviceMgr DoviceMgr	Fsm" state time: -	9222192747187 pKov(int) "AdcDovicol	м.
	[11:47:59]	[thread:GUI thr	ead] "Adc	DeviceMgr	Fsm" state change	e Error to Ready	i vi ý
	[11:47:59]	[thread:GUI thr [thread:GUI thr	ead] "Adc ead] "Adc	DeviceMgr DeviceMgr	Fsm" state time: 0 Fsm" input: reque	) estState: (none) : .setu	un
	devSet: Q	Set() ; deviceSe	tupKeys: (	QMap();			-1-
	[11:47:59] devSet: Q	[thread:GUI thr Set() ; deviceSe	ead] "Adc tupKeys: (	DeviceMgr QMap() ;	Fsm" input: reque	estState: (none) ; setu	up
	[11:47:59]	[thread:GUI thr	ead] "Can	't find or o	pen json config file	e in /root/.config/AFI E	le
	[11:47:59]	[thread:GUI thr	ead] "Can	't find or o	pen json config file	e in /root/.config/AFI E	le
	Adc64/def	ault/default.json	" eadl OObi	iect: Canno	ot create children f	for a parent that is in a	a
	thread.	tin cad.oor tin		Jeet. canno	cereate emaren i	or a parene diac is in t	
	(Parent is ) current th	MultiCastListene read is QThread	er(0x55bd (0x55bd68	686fa0e0), 34c9930)	parent's thread is	QThread(0x55bd687	ba
		•					

800





### Installed software

#### 1. SSH to the RPi which serves the SiPM PS control board

#### 2. Activate software environment

- source .supplr\_venv/bin/activate

#### Use the following cli commands to control it:

list of all available commands supplr **supplr board-status** print out the connected board (defines board number which is 0 for Module 1) **supplr reset --board <board number>** turn all the voltages off set the voltage [up to 120V\*] for a particular mezzanine [0-3] and mezzanine channel number [0-31]

### **HV** control software

- supplr set-channels-volt --board <board number> --voltage <voltage value> set the same voltage for all channels [128 channels in total]
- supplr set-channel-volt --board <board number> --mez <mez number> --mezch <mez channel number> --voltage <voltage value>
- supplr read-hv-supply-voltage --board <board number> reads the voltage value which is set on the laboratory power supply [SiPM bias]



## Installed software

### **VGA** control software

#### There are two options:

command line interface: gainr set -c config.yaml. The config format is given below. 2. You can install the client part of `Gainr` software doing the following:

> git clone <u>https://git.jinr.ru/greenlab/gainr.git</u> cd gainr python3 -m venv .venv source .venv/bin/activate poetry install

Configure

cat >	• con1	fig.yam	L < <e0f< th=""><th>=</th><th></th><th></th><th></th><th></th></e0f<>	=				
endpo	int:	http://	′ <raspb< td=""><td>berry-</td><td>ру-ір&gt;</td><td>:5000</td><td>)</td><td></td></raspb<>	berry-	ру-ір>	:5000	)	
# Cha	nnel	levels	must b	be int	egers	from	0 to	240
chann	els:							
0:	0							
1:	0							
2:	0							
3:	0							
4:	0							
5:	0							
6:	0							
7:	0							
EOF								

#### Set the output levels on VGA control

gainr set -c config.yaml

- 1. The client part is already installed on the Raspberry Pi. Use ssh to connect the RPi on VGA control board and then use the

240 corresponds to 24dB amplification factor 0 corresponds to 0dB amplification factor



- Connectivity check of the power lines for pre-amps
  - Power the E-PCBs to check the current draw -> 1.06A for all 16 E-PCBs or 265mA per VGA adapter card
- Channel by channel signal check -> LED system implementation required:
  - Power the LED by pulse Generator and provide TTL trigger for ADC
  - Power the SiPMs (~56V for all channels)
  - Run ADC GUI to check if all signals present

### **QA/QC** procedures

