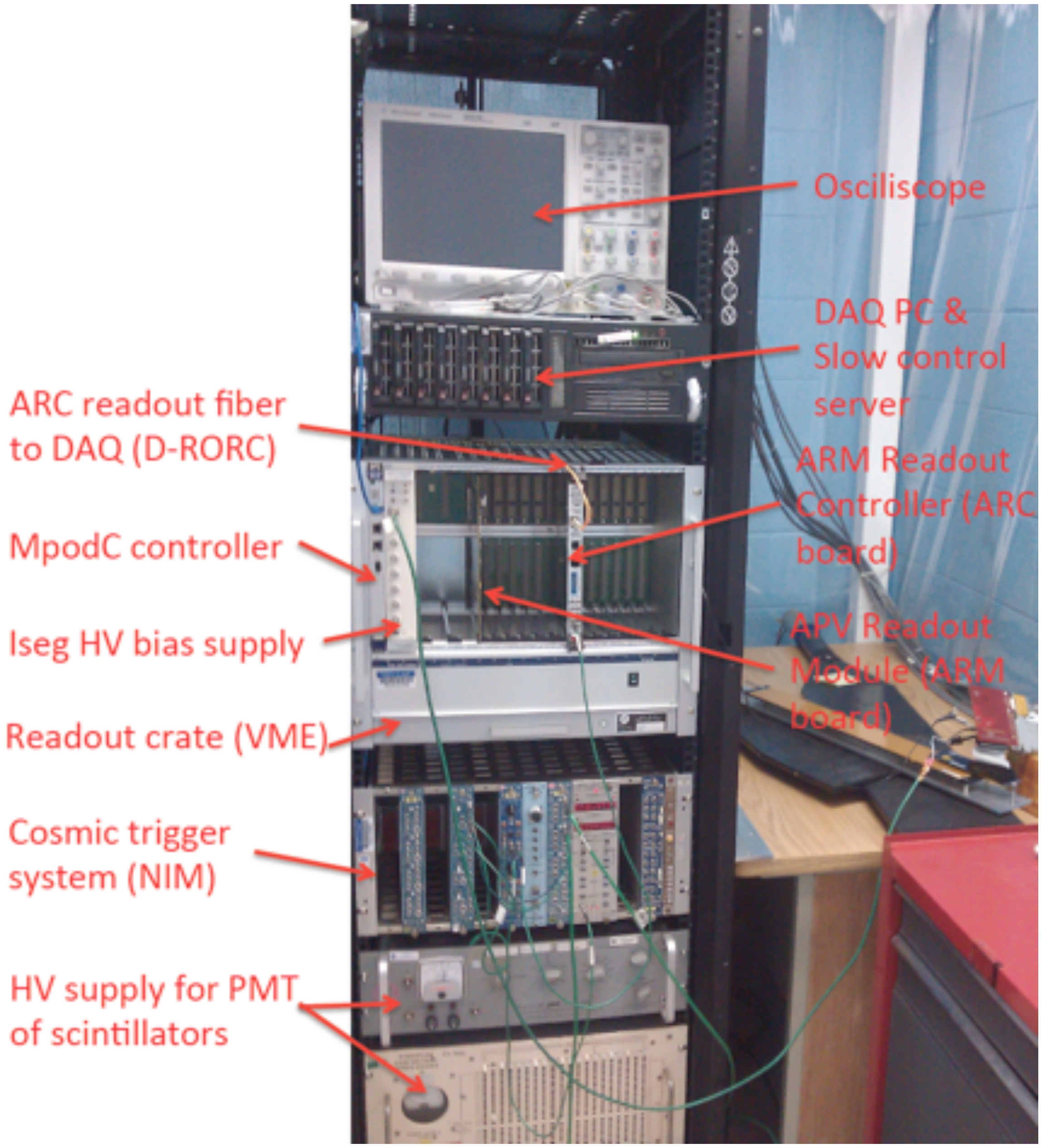
UIC IST Teststand User Manual (v_4)

Yaping Wang
Zhenyu Ye

12/16/2012

1. IST Teststand



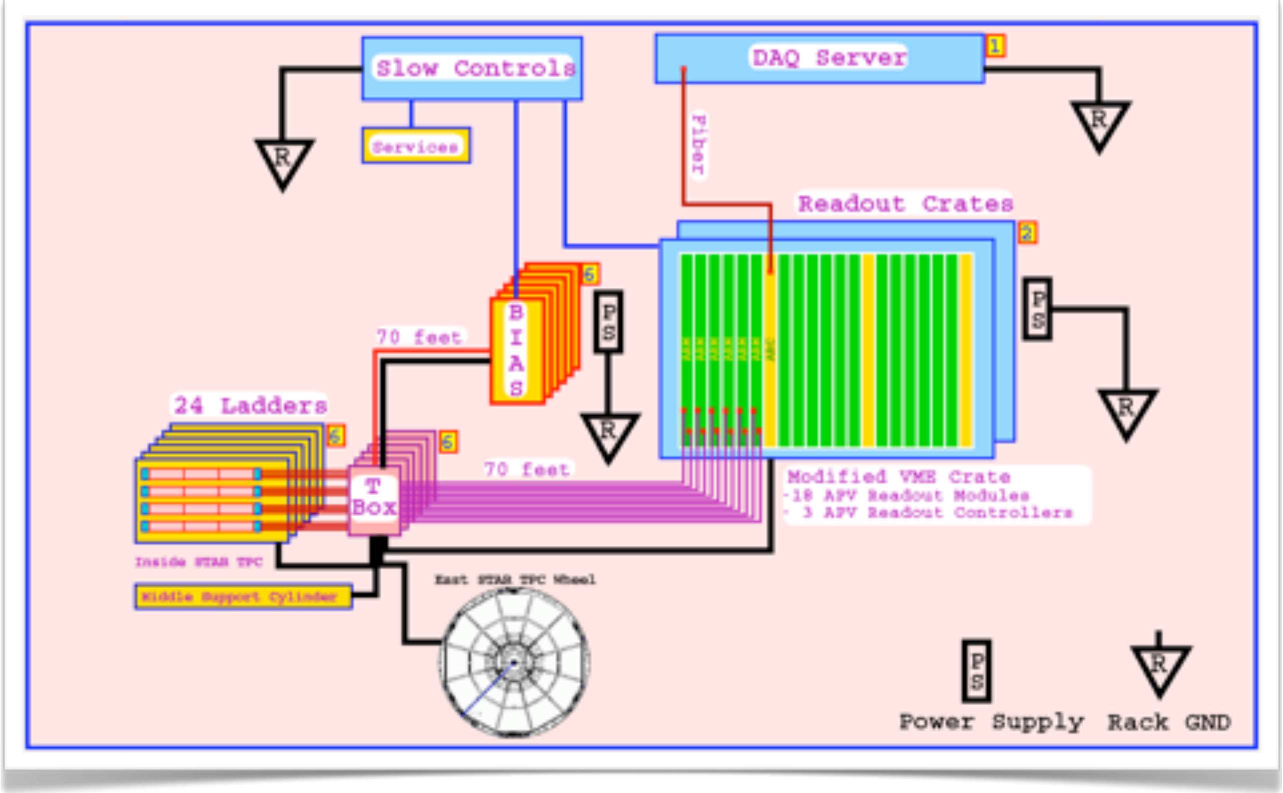
Note:

1) Local DAQ system has been installed at UIC for testing, and the cosmic trigger system is available for assembly.

2) The DAQ PC also provides service as a slow control server (slow control software - EPICS - should be installed, SL 5.3 installed).

3) The Iseg HV module is controlled through MPOD controller. The MPOD controller can be accessed via IP by EPICS software on DAQ server.

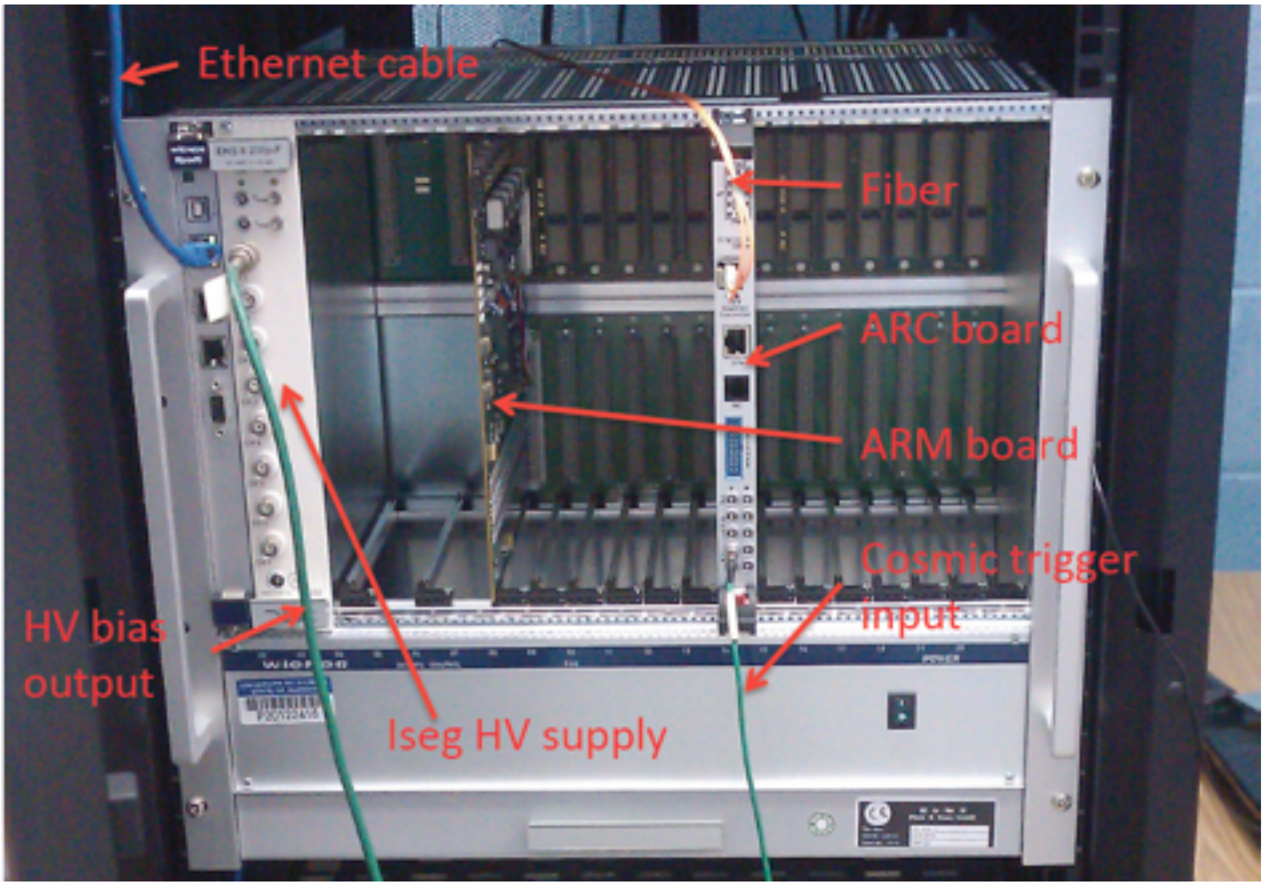
2. IST readout system



IST readout is identical to FGT design

1) For full readout testing of one IST stave, we need one ARM Readout Controller (ARC board) and two APV Readout Modules (ARM boards), and both of them are inserted in the VME crates.
2) FGT ARM Interface card (ABC board) is mounted on the backside of ARM board. The ABC board is used for inter-connection between Patch Panel Board (PPB) and ARM board. One ABC board covers two PPBs (connected by power cables and signal cables).
3) One IST stave has 3*12 APV chips, which were divided into three groups (A, B, C). They need one Transit Board (TB) and three PPBs for full readout. PPB and TB were connected with 50-channel bus.
4) The Data ReadOut Receiver Card (D-RORC, PCI kind) is mounted on the PCI slot of DAQ server, which builds a detector data link with the SIU card on the ARC via short optical fiber.
5) Slow control: (1) EPICS was installed on a Linux server (SL 5.3 suggested); (2) An EPICS package is available on the DAQ PC to control/monitor Iseg HV bias supply. Nominal bias voltage for sensors is set to 60 Volts.

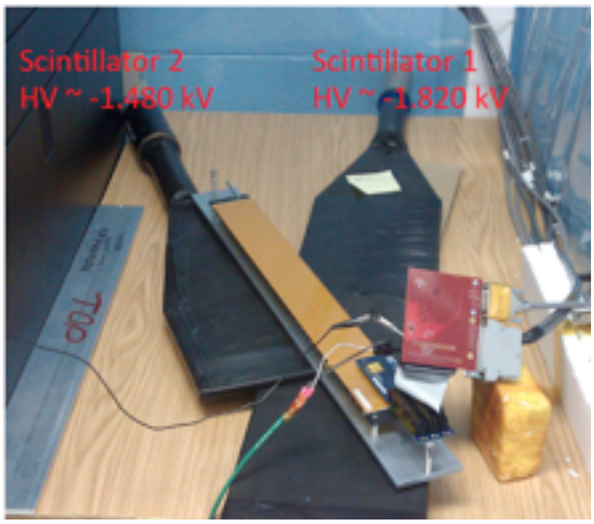
3. IST VME crate

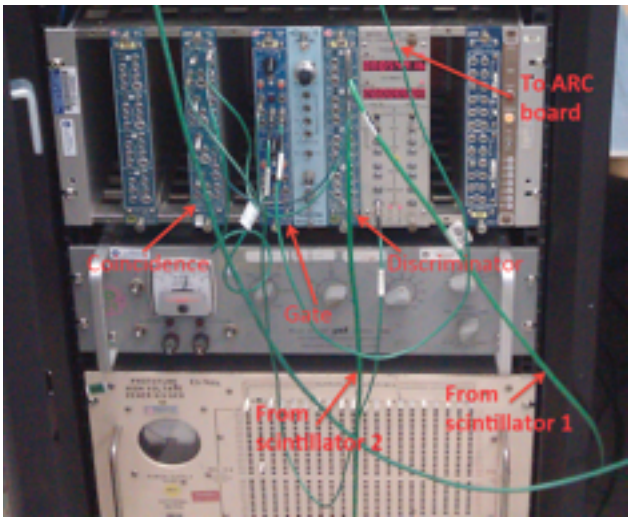


1) The ARC board should be located on the 7th slot of crate (If two ARC boards, the second one should be inserted into the 14th slot), and the SIU card is mounted on the ARC board.
2) The SIU card is connected to the D-RORC card via optical fiber.
3) Cosmic trigger is transmitted to the ARC trigger input for data acquisition.
4) The ARM board 0 locates on the 1st slot, then ARM1, ARM2, etc. Each ARC board covers 6 ARM boards.
5) The Iseg power supply locates on left slot close to MPODc module. The bias HV is set to 60 Volts.
6) The MPODc module should have a fixed IP address, which belongs to the same local network as the DAQ PC. We fixed a IP address for the MPODc, and also distributed a fixed IP for DAQ PC.

Note: Once the cards are available and cables are connected well, we can switch on the crate.
Do check carefully connections before you switch on the crate.

4. Cosmic trigger system





1) Cosmic trigger system consists of two sets of Scintillators + PMT, and the PMTs work at negative HV (The scintillator with bigger size is working at -1.820 kV, and the other smaller one is working at -1.48 kV). Note: (1) Do not directly switch off the HV module once it works with high voltage outputs. Firstly, you should decrease the output voltage step by step. (2) Connections: the PMT HV input “Cathode” connects to the HV supply output; Connect the PMT output signal (BNC connector close to Cathode) to the Discriminator.

(1) The HV for the scintillator 1 is supplied directly by channel 1 the High Voltage Regulated DC Power Supply (Model name HV-1547) at -1.82 kV.

(2) The HV for the scintillator 2 is provided the combination of the Model HV-1547 and Model ES-7092 (Phototube High Voltage Zener Divider). Firstly, we input the -1.82 kV to the Model ES-7092 from output channel 2 of the model HV-1547, and then insert a pin into position 220 (black) at channel 2. Then the output voltage from the channel 2 of the Model ES-7092 is -1.60 kV which is supplied for the scintillator 2.

(3) Check the HV connection carefully, and switch on power button if no question.

(4) Adjust the voltage selection (3*500V/step + 3*100V/step + 2*10V/step).

2) The pulses output from PMTs are transmitted to LRS model 621BL (Discriminator).

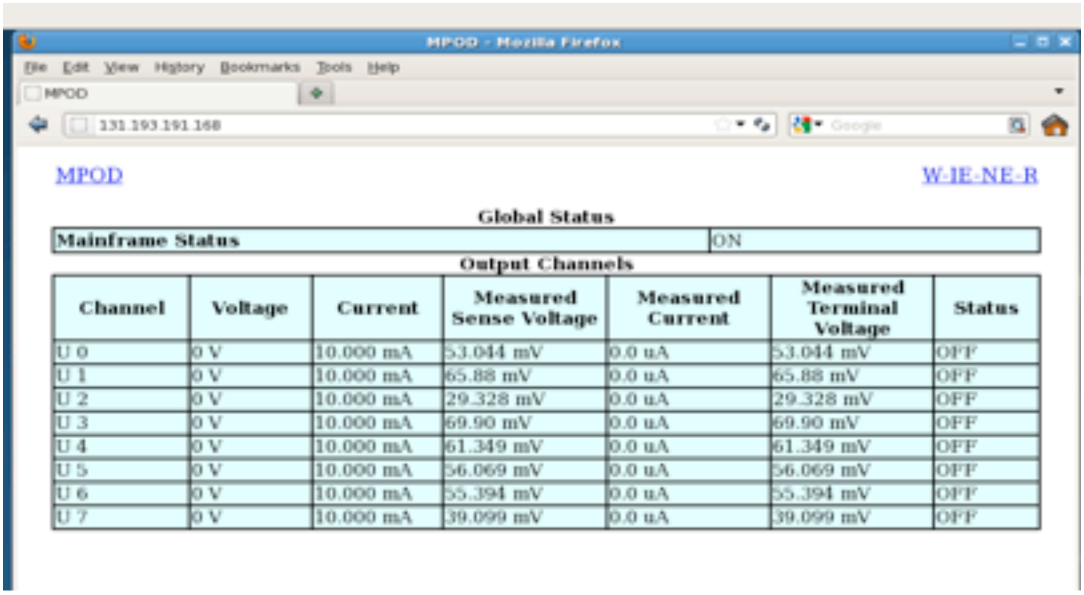
3) The two readout signals from discriminators are sent to LRS model 365AL (DUAL FOUR-FOLD) for coincidence (AND logic calculation).

4) The coincidence output then sent to P\S model 794 (QUAD GATER) for trigger pulse width adjustment. **The trigger signal sent to ARC board should be 1 ms of pulse width.**

5. IP address setting for MPODc (Windows OS)

1) Get file or CD: "MPOD_CD_ROM_002-6"
2) Open the CD and go to sub-directory "NET SNMP", and install "net-snmp-5.5.0-2.x64.exe". No special requirements for installation.
3) Once the net-snmp installed successfully, go to sub-directory "MPOD_Software/MUSEcontrol", and install MUSEcontrolInstall-2.1.1563.0.exe
4) Start program "Muse control" and open menu "System"-->"Configure": Setting IP address, subnet mask. Using the default port number. **MPODc IP address is set to: 131.193.191.168**
5) Save and Exit the program.
6) Once the IP setting successfully for MPODc, we can check now:

Open a web browser and input the MPODc's IP address, then Enter. We can monitor the status of Iseg power supply's outputs.



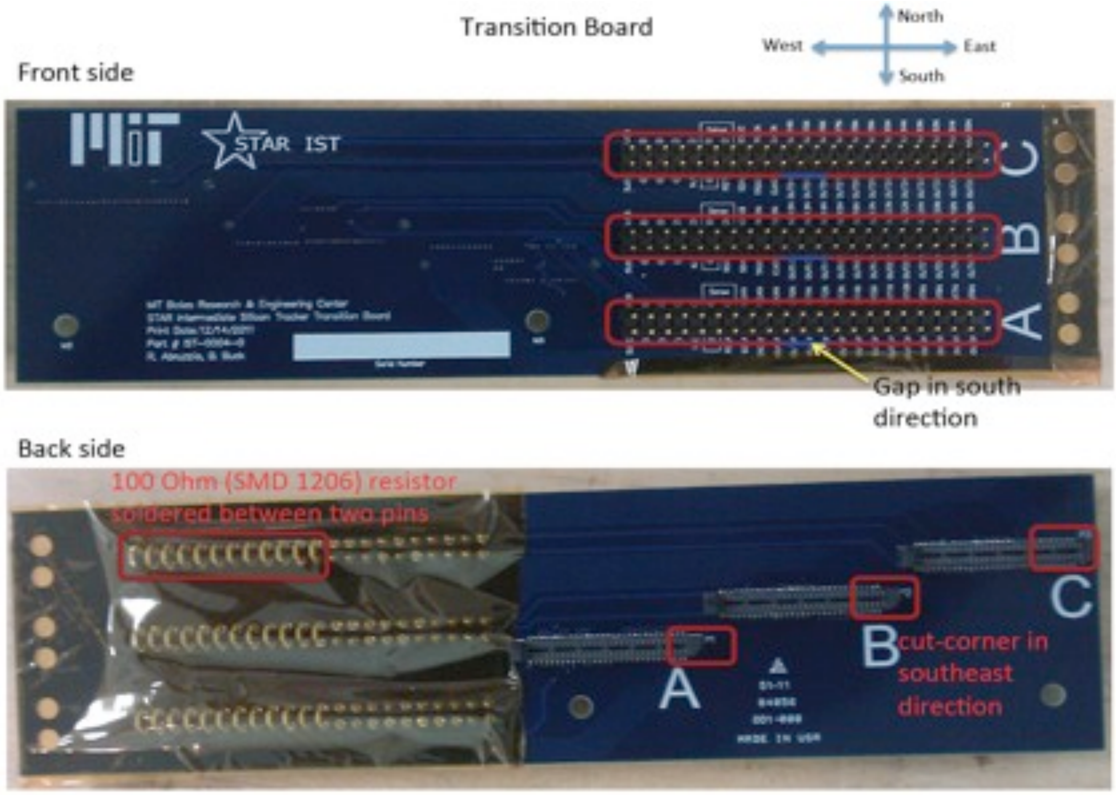
Mainframe Status	ON

Output Channels

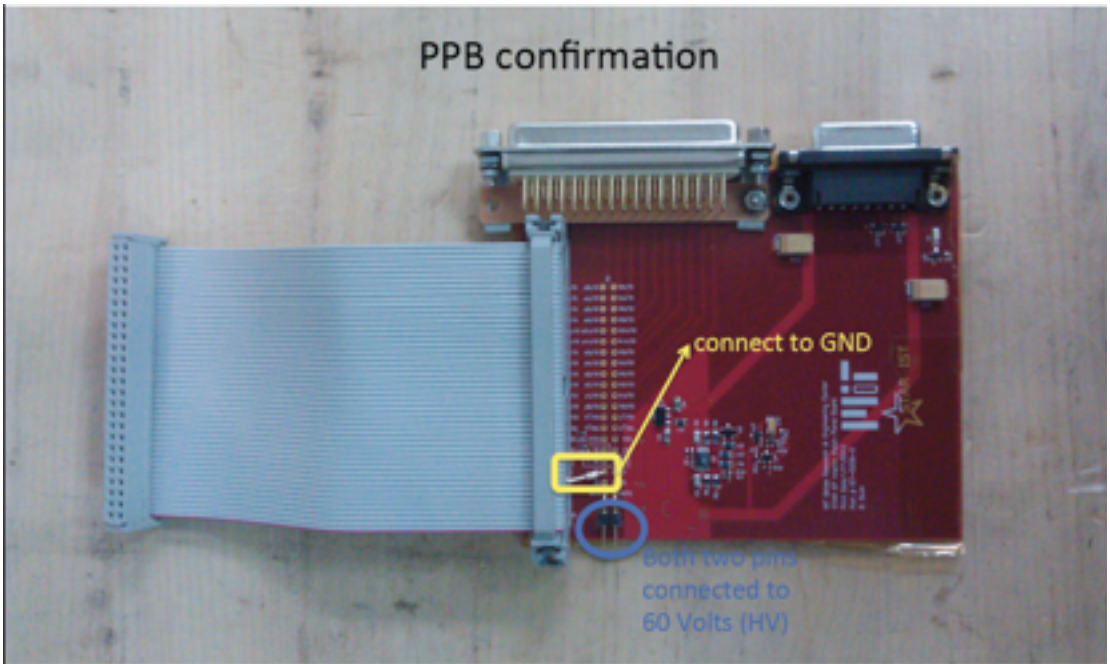
Channel	Voltage	Current	Measured Sense Voltage	Measured Current	Measured Terminal Voltage	Status
U 0	0 V	10.000 mA	53.044 mV	0.0 uA	53.044 mV	OFF
U 1	0 V	10.000 mA	65.88 mV	0.0 uA	65.88 mV	OFF
U 2	0 V	10.000 mA	29.328 mV	0.0 uA	29.328 mV	OFF
U 3	0 V	10.000 mA	69.90 mV	0.0 uA	69.90 mV	OFF
U 4	0 V	10.000 mA	61.349 mV	0.0 uA	61.349 mV	OFF
U 5	0 V	10.000 mA	56.069 mV	0.0 uA	56.069 mV	OFF
U 6	0 V	10.000 mA	55.394 mV	0.0 uA	55.394 mV	OFF
U 7	0 V	10.000 mA	39.099 mV	0.0 uA	39.099 mV	OFF

6. Build the testing chain

1) Transition board (T-Board) and Patch Panel Board (PPB):



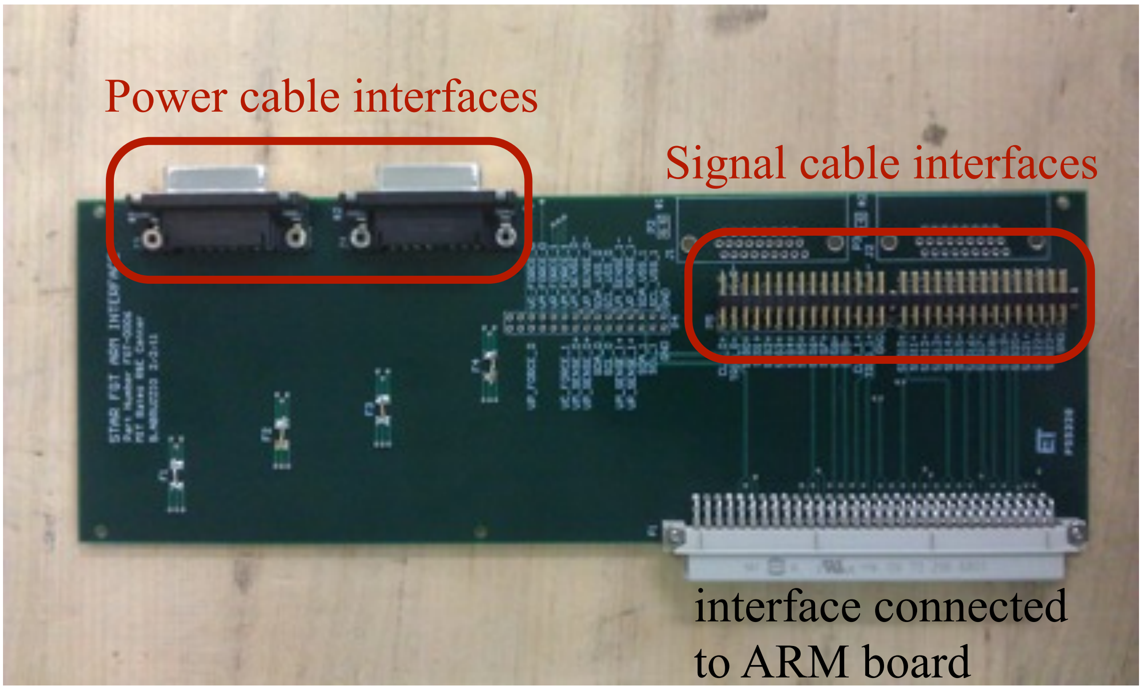
Backside of T-Board is mounted on the IST stave (A, B and C slots). Each slot on the T-Board front-side are connected to PPB board via a 50-channel bus. One T-Board covers to 3 PPBs.



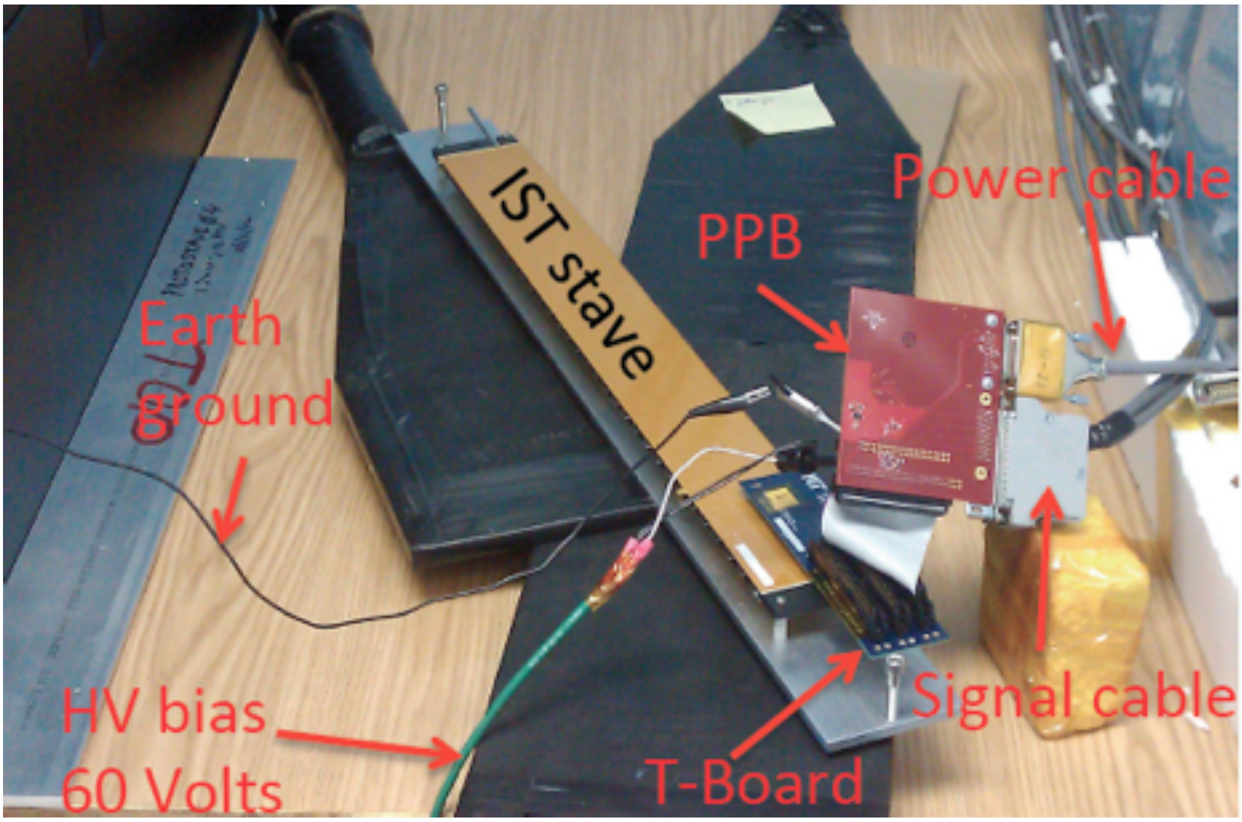
The pin circled in yellow has to be connected to the central ground and as such also defines 0 for the bias voltage.
The two pins that are circled in blue are the bias connections. One pin is connected to the backplane connection of one sensor, the other pin is connected to the backplane of the other sensor.

2) ABC board

ABC board is an interface between PPB and ARM boards, which mounted on the backside of the ARM boards. Each ABC board take charges of communications with two PPB boards (two power cable + two sets of signal cables).



3) Test chain (Note: bias HV is set to 70 Volts in actual test)



6. EPICS software installation (SL linux OS)

1) Installation guide

0. log in starist.phy.uic.edu and run a terminal
  su (to get root permission)
  mkdir /opt/epics
  cd /opt/epics
1. download base, extensions, and medm to current directory
    http://www.aps.anl.gov/epics/download/index.php
2. setting variable:
  export EPICS_HOST_ARCH=linux-x86_64
  export EPICS=/opt/epics
3. cd /opt/epics
    tar xvf baseR3.14.11.tar
    ln -s base-3.14.11 base
    tar xvf extensionsTop_20120904.tar
    tar xvf medm3_1_7.tar
4. cd /opt/epics/extensions/src/
    mv /opt/epics/medm3_1_7 ./medm
5. cd $EPICS/base
    make
6. yum install openmotif-devel.i686 openmotif-devel.x86_64
7. cd /opt/epics/extensions/src/medm
    make
8. setting enviroment variables:
    touch EPICS.sh and EPICS.csh

    vim EPICS.sh
      export EPICS_HOST_ARCH=linux-x86_64
      export EPICS=/opt/epics
      export PATH=$PATH:$EPICS/base/bin/linux-x86_64:$EPICS/extensions/bin/linux-x86_64

    vim EPICS.csh
      setenv EPICS_HOST_ARCH linux-x86_64
      setenv EPICS /opt/epics
      setenv PATH $PATH:$EPICS/base/bin/linux-x86_64:$EPICS/extensions/bin/linux-x86_64
9. sncseq-2.0.12 and asyn4-20 installation (download, extract, and make under the /opt/epics directory)
    (1)
10. install net-snmp service
      su (to get root permission)

yum install net-snmp (net-snmp-devel package is installed, then the commands "snmpd", "net-snmp-config" and "snmptrapd" are available)

11. install net-snmp utils

download net-snmp-5.6.2.tar.gz to via link: http://sourceforge.net/projects/net-snmp/files/net-snmp/5.6.2/

tar zxvf net-snmp-5.6.2.tar.gz
cd net-snmp-5.6.2
./configure (default setting, no special requirement)
make
su
make install

12. test snmp and set community passwords

(1) snmpwalk -v 2c -m +WIENER-CRATE-MIB -c public 131.193.191.168

SNMPv2-MIB::sysDescr.0 = STRING: WIENER MPOD (2688018, MPOD 2.1.1909.0, MPODslave 1.09, MPOD-BL 1.50)
SNMPv2-MIB::sysObjectID.0 = OID: WIENER-CRATE-MIB::sysMainSwitch.0
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (45460426) 5 days, 6:16:44.26
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING:
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 79

(2) snmpwalk -v 2c -m +WIENER-CRATE-MIB -c guru 131.193.191.168 snmpCommunityName

WIENER-CRATE-MIB::snmpCommunityName.public = STRING: "public"
WIENER-CRATE-MIB::snmpCommunityName.private = STRING: "private"
WIENER-CRATE-MIB::snmpCommunityName.admin = STRING: "admin"
WIENER-CRATE-MIB::snmpCommunityName.guru = STRING: "guru"

(3) snmpset -v 2c -m +WIENER-CRATE-MIB -c guru 131.193.191.168 snmpCommunityName.guru s seCrET (setting of password)

(4) snmpwalk -v 2c -m +WIENER-CRATE-MIB -c seCrET 131.193.191.168 snmpCommunityName

WIENER-CRATE-MIB::snmpCommunityName.public = STRING: "public"
WIENER-CRATE-MIB::snmpCommunityName.private = STRING: "private"
WIENER-CRATE-MIB::snmpCommunityName.admin = STRING: "admin"
WIENER-CRATE-MIB::snmpCommunityName.guru = STRING: "seCrET"

13. switch to your home directory

add environment variable: "setenv LD_LIBRARY_PATH /usr/local/lib:/usr/lib:$LD_LIBRARY_PATH" in .cshrc or "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/lib:/usr/lib" in .bashrc

add command line "source /opt/epics/EPICS.sh" in .bashrc or "source /opt/epics/EPICS.csh" in .cshrc

14. Install slow control package

(1) download the slow control package to the /home/ist/

(2) mkdir ist_hv_devel

(3) extract the package to the directory

(4) cd ist_hv_devel

(5) vim configure/RELEASE

EPICS_BASE=/opt/epics/base

TEMPLATE_TOP=$(EPICS_BASE)/templates/makeBaseApp/top

ASYN = /opt/epics/asyn4-20/

SNCSEQ=/opt/epics/sncseq-2.0.12/

(6) replace “fgttester/fgt_hv_devel” with “ist/ist_hv_devel” in all files

(7) correct errors in snmpApp/src/Makefile

(8) vim ist-test_display

source setup_independent_env

setenv EPICS_CA_AUTO_ADDR_LIST NO

setenv EPICS_CA_ADDR_LIST "starist.phy.uic.edu"

cd display

medm -x FGTTestPowerSupply.adl &

(9) vim setup_independent_env

setenv EPICS_CA_SERVER_PORT 45582

setenv EPICS_CA_AUTO_ADDR_LIST NO

setenv EPICS_CA_ADDR_LIST "starist.phy.uic.edu"

(10) mkdir /home/ist/.snmp

mkdir /home/ist/.snmp/mibs/

cp /home/ist/ist_hv_devel/WIENER-CRATE-MIB.txt /home/ist/.snmp/mibs

(11) cd iocBoot/iocsnmp

vim ist-test.cmd

(input Mpodc IP address and replace SYSTEM name “ist-test” with “isttest”)

vim ../../display/FGTTestPowerSupply.adl

(replace SYSTEM name “ist-test” with “isttest”)

vim iocstart

source ../../setup_independent_env

ist-test.cmd

(12) cd

mkdir .snmp

mkdir .snmp/mibs/

cp /home/ist/ist_hv_devel/WIENER-CRATE-MIB.txt ~/.snmp/mibs/

cp /home/ist/ist_hv_devel/ WIENER-CRATE-MIB.txt /usr/share/snmp/mibs/

(13) cd /home/ist/ist_hv_devel

gmake clean

gmake

Reference:

EPICS Installation guide referred links below:

http://nibot-lab.livejournal.com/91787.html

https://pubweb.bnl.gov/~mdavidsaver/epics-doc/epics-starting.html

7. Software preparation for DAQ and Analysis

IST DAQ PC: starist.phy.uic.edu

User account: ist

Password: XXXXXX

1) Software preparation for DAQ

1. log in IST DAQ PC
2. Start the afs service
3. open a terminal window

```
mkdir "cvs" to install software via afs service
cd
vim .bashrc  (add following variables)
      export PATH=$PATH:.
      export CVSROOT=/afs/rhic.bnl.gov/star/packages/repository
      export RTS_PROJECT=STAR
      export PROJDIR=/RTS
      export RTS_DEV_PREFIX=pro_
      export USE_64BITS=1
```

4. change to "cvs" directory

```
cd cvs
klog XXXXX (AFS account at RACF)
cvs checkout StRoot/RTS
cvs checkout online/RTS
```

5. install driver for RORC64 (This step is already done by Tonko)

6. install DAQ system (compiled by Tonko)

```
cd cvs/online/RTS/src/FGT_TEST/
cp /home/tonko/cvs/online/RTS/src/FGT_TEST/fgt_run_daq ./
cp /home/tonko/cvs/online/RTS/src/FGT_TEST/fgt_run_daq.o ./
cp /home/tonko/cvs/online/RTS/src/FGT_TEST/fgt_read_daq ./
cp /home/tonko/cvs/online/RTS/src/FGT_TEST/fgt_read_daq.o ./
cp /home/tonko/cvs/online/RTS/src/FGT_TEST/fgt_run_daq ./
cp /home/tonko/cvs/online/RTS/src/FGT_TEST/ars_lib.o ./
```

2) Software preparation for Analysis

1. log in IST DAQ PC
2. In order to match the analysis library at STAR RCF, we need to make both the Scientific Linux version and gcc version identical to one at RCF. After that, we could share the identical library via AFS service (sl57_x8664_gcc432).

open a terminal window

mkdir gcc and download gcc-4.3.2 package (http://gcc.gnu.org/mirrors.html)

cd gcc/

tar zxvf gcc-4.3.2.tar.gz
mkdir gcc_build
cd gcc_build
su
../gcc-4.3.2/configure --prefix=/usr --enable-shared --enable-threads=posix --enable-__cxa_atexit --enable-c99 --enable-long-long --enable-languages=c,c++ --disable-libstdcxx-pch
make
make install

3. setting RCF environment
   cd ~
   vim .cshrc   (add the following lines in to the script)
      setenv GROUP_DIR /afs/rhic.bnl.gov/rhstar/group
      if (-r $GROUP_DIR/star_cshrc.csh) then
         source $GROUP_DIR/star_cshrc.csh
      endif
4. open a terminal window and switch to your home directory
   mkdir ARMdisplay
   cd ARMdisplay
   copy the latest script "ARMdisplay.C" to the current directory. You can find the latest version at RCF: /star/institutions/mit/nieuwhzs/IST_TESTING/IST_PROTOTYPE_STACK_02-03-04/COSMICS_Sep2012
   Once the script is created first time or edited, you should compile it again:
   rm ARMdisplay ARMdisplay.o
   stardev
   make
5. the excutive file "ARMdisplay" is generated once compiling successfully.

8. Instruction to take data with IST

I. Starting readout crate control

(1) Start EPICS software: the EPICS back-end software is running in the background on IST DAQ PC. Once the IST DAQ PC gets rebooted, the EPICS can not be started automatically.

login to the IST DAQ PC directly:

ssh ist@starist.phy.uic.edu

password: XXXXXX

cd /home/ist/ist_hv_devel/iocBoot/iocsnmp

./iocstart (let it run all the time)

(2) Start the readout crate control GUI:

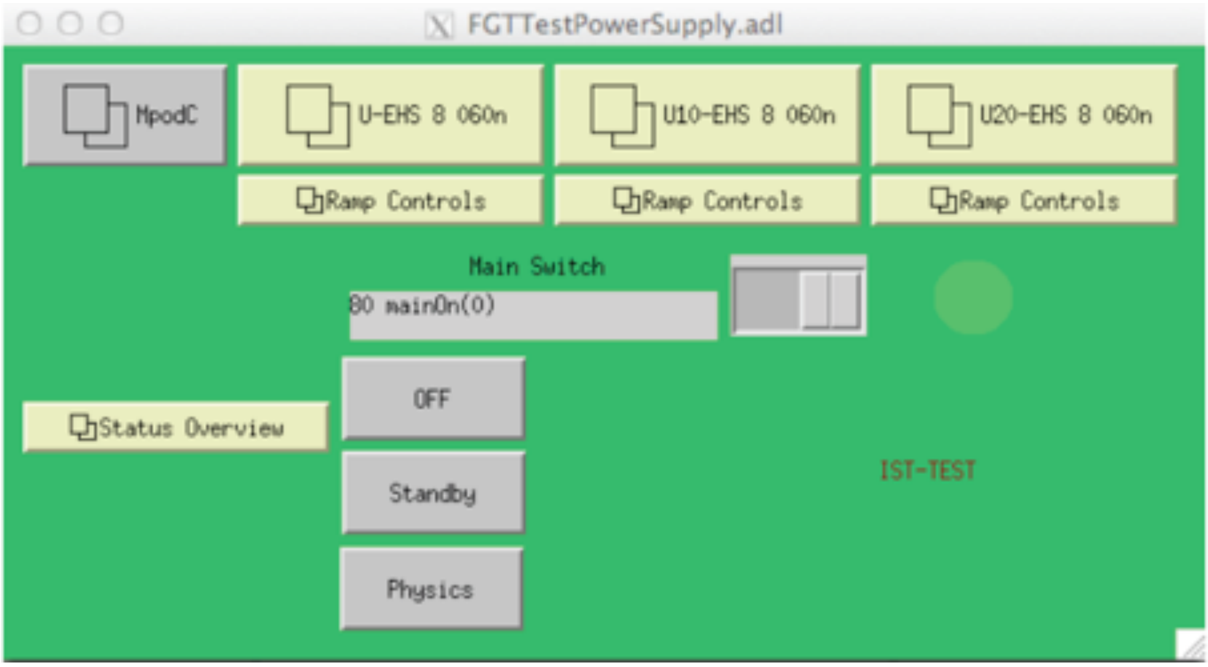
cd ~/ist_hv_devel

./ist-test_display

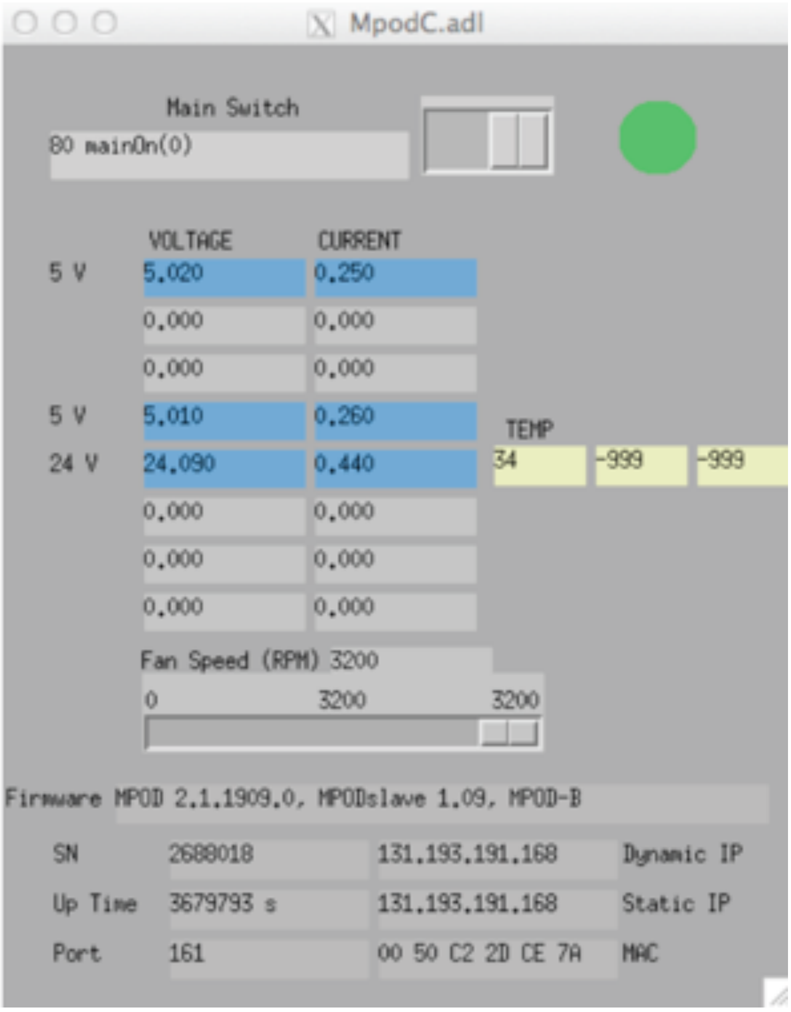
The GUI pops up now and parameters can be set easily.

1) Start the Main Switch (as shown in the following picture left)

click the switch, the circle turns green indicating the readout VME crate is powered on.



2) MpodC information can be found as shown in the picture below:



3) Enable the EHS 8205E HV output:

Click the button “V-EHS 8 060n”, and a new interface is popped up as below:

Firstly, we set the “Voltage” to 60.000

Secondly, switch on the v0 channel by click the switcher.

II.Running the DAQ

(1) login to the IST DAQ PC

go to the IST working directory

cd /cvs/online/RTS/src/FGT_TEST/

(2) when the readout crate is on and everything is hooked up correctly then you should be able to run the DAQ:

./fgt_run_daq -L -c ist_rdo_conf_YPWANG_for_test.txt -C 0x03 -n 100 -S /scratch/data/ist_software_test1.sfs

-L means that the ARC local clock is used, always needs to be there

-C 0x01 means that ARM0 gets used (mask 0x01 for ARM0 and 0x02 for ARM1, 0x03 for both ARM0 and ARM1, etc.)

-c set a configuration file <configuration file>

-n set number of events

-S set output file name. The output should be put in /scratch/data

III. Analyzing DAQ files with ARMdisplay

(1) login to IST DAQ PC

(2) go to the ARMdisplay directory:
cd ~/ARMdisplay/

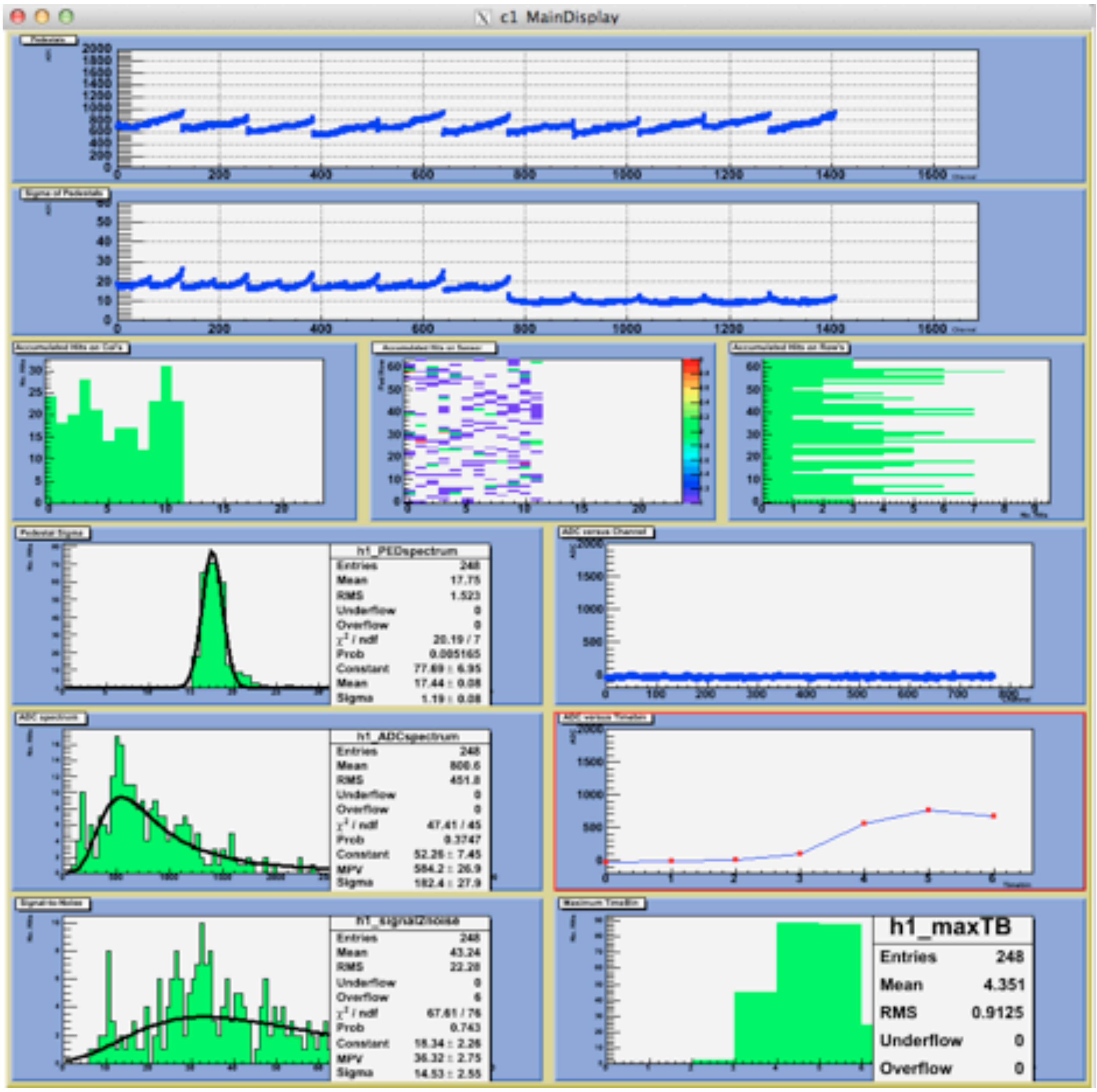
(3) analyze daq file
stardev
./ARMdisplay -f <daq_file_name>

(4) The ARMdisplay contains a lot of messages about missing APV's. It will also make four display windows and depending on how many ARM's were used up to 3 windows. The 4th display is for tracking results.

(5) we can make changes for ARMdisplay.C, and "make" again.

Note: If more than one ARMs were used, we should make changes in the ARMdisplay.C, like readout slot, etc.

(6) The cosmic data can be found here: /scratch/data/


c1_MainDisplay
h1_PEDspectrum
Entries 248
Mean 17.75
RMS 1.523
Underflow 0
Overflow 0
Prob 0.005165
Constant 77.69 ± 6.95
Mean 17.44 ± 0.08
Sigma 1.19 ± 0.08
h1_ADCspectrum
Entries 248
Mean 800.6
RMS 451.8
Underflow 0
Overflow 0
Prob 0.3747
Constant 52.26 ± 7.45
MPV 584.2 ± 26.9
Sigma 182.4 ± 27.9
h1_signal2noise
Entries 248
Mean 43.24
RMS 22.28
Underflow 0
Overflow 6
Prob 0.743
Constant 18.34 ± 2.26
MPV 36.32 ± 2.75
Sigma 14.53 ± 2.55
h1_maxTB
Entries 248
Mean 4.351
RMS 0.9125
Underflow 0
Overflow 0