Rayonix MX170-HS

This is where we put all available Rayonix information.

Documentation

doc
include
src
misc files (conf, log, csv)
Rayonix demo history

DAQ Usage

```
Usage: rayonix -i <detid> -p <platform>,<mod>,<chan> [OPTIONS]
Options:
  -i <detid>                  detector ID (e.g. 22 for XppEndstation)
  -p <platform>,<mod>,<chan>  platform number, EVR module, EVR channel
  -u <alias>                  set device alias
  -v                          increase verbosity (may be repeated)
```
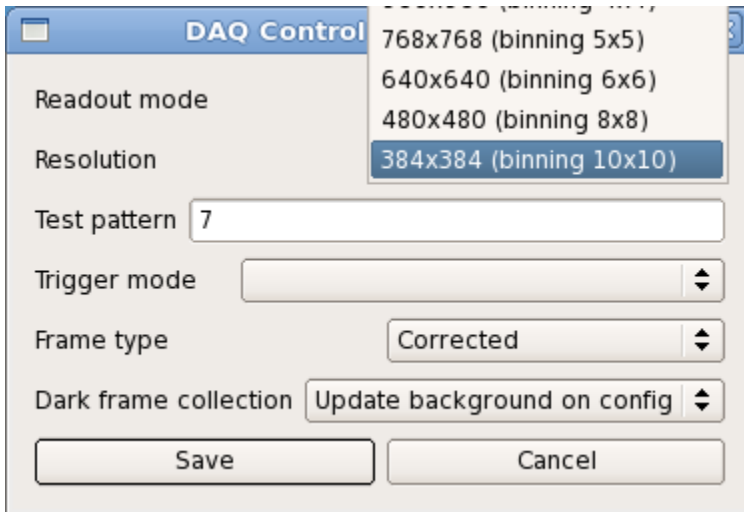
Rayonix Binning Mode Configuration

In standard mode, only 2x2 and 4x4 binning are selectable. These are well tested modes.

To enable additional hardware binning options, select **Options -> Expert Mode** in the DAQ configuration GUI. This enables Rayonix binning modes up to 10x10.

Managing the Rayonix Server

Configuring the Private Network Connection to the Rayonix PC

The Rayonix PC is configured with IP address 10.0.1.101 on the point-to-point 10G link.

The DAQ readout node should be configured with IP address 10.0.1.1 on the point-to-point 10G link.

Here is an example from daq-xpp-rayonix:

```
eth2      Link encap:Ethernet  HWaddr 00:60:DD:44:B4:14
          inet addr:10.0.1.1  Bcast:10.0.1.255  Mask:255.255.255.0
          inet6 addr: fe80::260:ddff:fe44:b414/64 Scope:Link
          UP BROADCAST MULTICAST  MTU:9000  Metric:1
          RX packets:1172819660 errors:0 dropped:0 overruns:0 frame:0
          TX packets:25773 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:9070537182124 (8.2 TiB)  TX bytes:10921615 (10.4 MiB)
          Interrupt:245
```

For example in MFX using eth3:

[koglin@daq-mfx-cspad01 mfxopr 12:19:36] cat /etc/sysconfig/network-scripts/ifcfg-eth3

DEVICE=eth3

ONBOOT=yes

IPADDR=10.0.1.1

NETMASK=255.255.255.0

HWADDR=00:60:DD:45:67:0F

MTU=9000

Running rnxserv on the Rayonix PC

November 2016 Update:

- The **rnxserv-statuslist-lowNoise/rnxserv** binary includes a fix for Rayonix readout mode 3 ("Low Noise").
- Earlier rnxserv versions default to mode 1 ("Standard") if readout mode 3 is requested.
- "Low Noise" readout mode is not recommended for frame rates above 5 Hz.

August 2018 Update:

- The **rnxserv-newcraydl/rnxserv** binary includes support for a new version of the craydl SDK.
- Fixes the long standing binning change crash issue.
- Supports the bigger MX340-HS detector in MFX.

In case the Rayonix PC is restarted, the server process must be started using procServ. Choose a name for the log file ("logfile.txt" in the following example), then run procServ from the ~hsuser/slac directory:

[xpp-control ~]$ **ssh hsuser@con-ics-xpp-rayonix**
[hsuser@makenxwork2 ~]$ **cd slac**
[hsuser@makenxwork2 slac]$ **procServ-2.5.1/procServ --allow -l 30098 -L logfile.txt 30099 rnxserv-newcraydl/rnxserv**
procServ-2.5.1/procServ: spawning daemon process: 8887
No log file specified and stdout is not a file - no log will be kept.

For the new MX340-HS in MFX the command is slightly different since it is RHEL7 host:

[mfx-control ~]$ **ssh hsuser@con-ics-mfx-rayonix**
[hsuser@localhost ~]$ cd slac
[hsuser@localhost slac]$ **/opt/lcls/pkg_mgr/release/procServ/2.7.0-1.1.0/rhel7-x86_64/bin/procServ --allow -l 30098 -L logfile.txt 30099 rnxserv-newcraydl/rnxserv -d 1**
/opt/lcls/pkg_mgr/release/procServ/2.7.0-1.1.0/rhel7-x86_64/bin/procServ: spawning daemon process: 22907

To verify that the server started normally, cat the log file. For an example log file click here.

Alternatively the following sequence of commands will work on both machines (using helper scripts):

Launching rnxserv

```
# for MFX the hostname is con-ics-mfx-rayonix and for XPP it is con-ics-xpp-rayonix
ssh hsuser@con-ics-${HUTCH}-rayonix
cd slac
./startDaqInterface_newcraydl
```

For added log file verbosity, pass the -v flag to rnxserv. This flag can be repeated up to 3 times for maximum verbosity. Maximum verbosity may lead to larger log files and slower performance.

Maximum log file verbosity example (not recommended for frame rates above 10 Hz):

> [xpp-control ~]$ **ssh hsuser@con-ics-xpp-rayonix**
> [hsuser@makenxwork2 ~]$ **cd slac**
> [hsuser@makenxwork2 slac]$ **procServ-2.5.1/procServ --allow -I 30098 -L logfile.txt 30099 rnxserv-newcraydl/rnxserv -vvv**
> procServ-2.5.1/procServ: spawning daemon process: 8887
> No log file specified and stdout is not a file - no log will be kept.

For the new MX340-HS in MFX the command is slightly different since it is RHEL7 host:

> [mfx-control ~]$ **ssh hsuser@con-ics-mfx-rayonix**
> [hsuser@localhost ~]$ cd slac
> [hsuser@localhost slac]$ **/opt/lcls/pkg_mgr/release/procServ/2.7.0-1.1.0/rhel7-x86_64/bin/procServ --allow -I 30098 -L logfile.txt 30099 rnxserv-newcraydl/rnxserv -d 1 -vvv**
> /opt/lcls/pkg_mgr/release/procServ/2.7.0-1.1.0/rhel7-x86_64/bin/procServ: spawning daemon process: 22907

Alternatively the following sequence of commands will work on both machines (using helper scripts):

Launching rnxserv in verbose mode

```
# for MFX the hostname is con-ics-mfx-rayonix and for XPP it is con-ics-xpp-rayonix
ssh hsuser@con-ics-${HUTCH}-rayonix
cd slac
./startDaqInterface_newcraydl -vvv
```

To monitor the output of the remote server running on the Rayonix box, first ssh to daq-xpp-rayonix then run telnet as shown below. This is a read-only procServ console, and it is safe to open multiple instances at the same time.

If you need to restart the remote server, do the same except change the port # from 30098 to 30099. This is a read/write procServ console, and it allows you to send Ctrl-C to the program or Ctrl-X to procServ, restarting the program.

To shutdown Rayonix, first ssh to host machine and stop procServ process. e.g., telnet and kill with Cntr-X after Cntr-T to make sure it does not restart after killing:

 -> ssh daq-mfx-cspad01

 -> telnet 10.0.1.101 30099

Then follow steps to turn off chiller and shutdown.

Note that several temperatures are displayed automatically once per minute.

> [daq-xpp-rayonix ~]$ **telnet 10.0.1.101 30098**
> Trying 10.0.1.101...
> Connected to 10.0.1.101 (10.0.1.101).
> Escape character is '^]'.
> @@@ procServ server PID: 6979
> @@@ Server startup directory: /home/hsuser/slac
> @@@ Child startup directory: /home/hsuser/slac
> @@@ Child started as: rnxserv/rnxserv
> @@@ Child "rnxserv/rnxserv" PID: 22446
> @@@ procServ server started at: Thu Apr 17 17:49:34 2014
> @@@ Child "rnxserv/rnxserv" started at: Thu Apr 17 17:52:34 2014

```
2014-Apr-17 19:14:38,[STATUS
],[STATUS_DATA],-80.35,-78.85,-80.45,-80.05,-113.95,-109.85,36.05,43.75,51.25,36.25,47.85,47.85,40,286,284,288,0,0,1200,32:53:08
2014-Apr-17 19:15:38,[STATUS
],[STATUS_DATA],-80.35,-79.05,-80.45,-79.95,-114.25,-110.05,36.05,43.55,50.95,36.25,47.85,47.85,41,286,285,288,0,0,1200,32:54:08
2014-Apr-17 19:16:38,[STATUS
],[STATUS_DATA],-80.35,-79.05,-80.45,-80.15,-114.35,-110.05,35.05,43.65,50.95,36.25,47.85,47.85,40,286,285,288,0,0,1200,32:55:08
2014-Apr-17 19:17:38,[STATUS
],[STATUS_DATA],-80.25,-79.15,-80.45,-80.25,-114.45,-110.25,36.05,43.35,50.95,36.25,47.85,47.85,40,286,284,288,0,0,1200,32:56:07
2014-Apr-17 19:18:38,[STATUS
],[STATUS_DATA],-80.25,-79.35,-80.35,-80.25,-114.45,-110.35,35.05,43.35,50.95,36.25,47.85,47.85,40,286,284,288,0,0,1200,32:57:07
2014-Apr-17 19:19:38,[STATUS
],[STATUS_DATA],-80.25,-79.45,-80.35,-80.35,-114.55,-110.45,36.05,43.35,50.95,36.25,47.85,47.85,41,286,284,288,0,0,1200,32:58:07
```
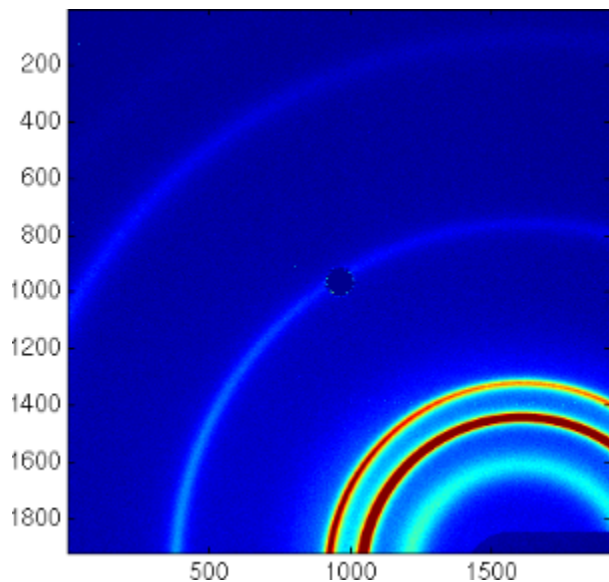
In addition, the "Max Rayonix temperature" will be displayed on this console during the Rayonix configuration cycle.
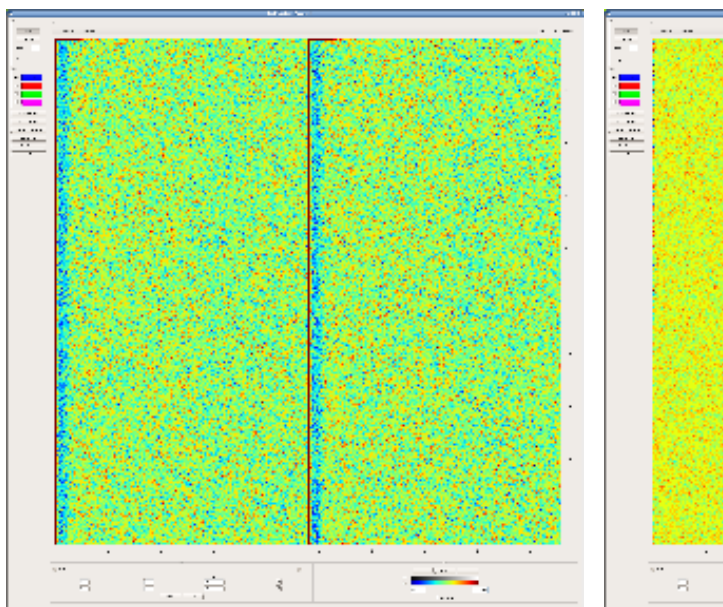
```
18_17:02:30: ** workCommand.cmd == RNX_WORK_OPENDEV **
SetBinning(2) done
==== Max Rayonix temperature: -79.85 C ====
SetExposureTime(0.000) done
SetIntervalTime(0.0) done
18_17:02:30:SetReadoutMode(1) done
18_17:02:30:SetFrameTriggerMode(FrameTriggerType(FrameTriggerTypeFrame)) done
18_17:02:30:SetReadoutMode(1) done
EnableTestPattern(false) done
SendParameters() done
SetAcquisitionUserCB() done
```

Rayonix Screen Shots

LCLS User Image

Rayonix, 2x2 binning, raw

Low-level debug logs can be viewed as root on the Rayonix PC like so:

```
[root@makenxwork2 ~]# tail -25  /var/log/messages
Dec 11 23:25:59 makenxwork2 rnxserv: workThread.c:813 SetupAcquisitionSequence(0) done
Dec 11 23:25:59 makenxwork2 rnxserv: workThread.c:823 Thread #0: StartAcquisition() done
Dec 12 01:30:39 makenxwork2 rnxserv: rnxserv.c:444 state = RNX_STATE_CONFIGURED
Dec 12 01:30:39 makenxwork2 rnxserv: rnxserv.c:584 state = RNX_STATE_UNCONFIGURED
Dec 12 01:30:39 makenxwork2 rnxserv: rnxserv.c:496 Command socket closed
Dec 12 01:30:39 makenxwork2 rnxserv: workThread.c:868 EndAcquisition(true) succeeded
Dec 12 01:30:39 makenxwork2 rnxserv: workThread.c:833 Thread #0: EndAcquisition() done
Dec 12 01:30:53 makenxwork2 rnxserv: rnxserv.c:243 Command socket opened
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:435 SetBinning(2) done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:449 SetExposureTime(0.000) done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:462 SetIntervalTime(0.0) done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:497 SetReadoutMode(1) done
```
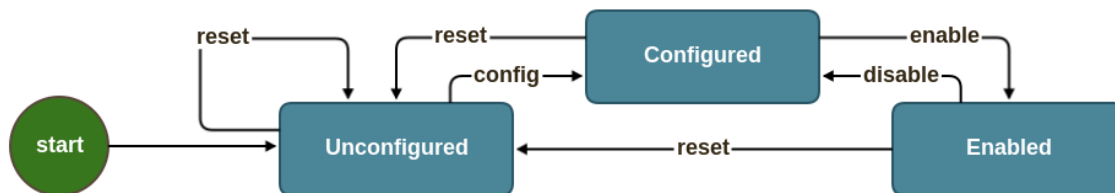
```
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:526 Trigger Mode 0: true frame transfer
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:536 SetFrameTriggerMode(FrameTriggerType(FrameTriggerTypeFrame)) done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:543 DigitalIOSignalTypeOpto
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:555 SetReadoutMode(1) done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:594 EnableTestPattern(false) done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:608 SendParameters() done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:621 SetAcquisitionUserCB() done
Dec 12 01:30:53 makenxwork2 rnxserv: rnxserv.c:274 state = RNX_STATE_CONFIGURED
Dec 12 01:30:53 makenxwork2 rnxserv: rnxserv.c:280 framesize=3686400;epoch=74;deviceID=MX170-HS:109
Dec 12 01:30:53 makenxwork2 rnxserv: rnxserv.c:422 state = RNX_STATE_ENABLED
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:799 STARTACQ: _darkFlag=false _readoutMode=1 _testPattern=0
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:813 SetupAcquisitionSequence(0) done
Dec 12 01:30:53 makenxwork2 rnxserv: workThread.c:823 Thread #0: StartAcquisition() done
[root@makenxwork2 ~]#
```

Rayonix Server State Machine

Notes

Notes from meeting with Michael Blum.
Rayonix demo history