High Speed Series X-ray Detector Manual

Including:

- ***capxure*** – detector control GUI and image viewer
- **MX HS Series** X-ray Detector System device manual
- **LX HS Series** X-ray Detector System device manual
- **SX HS Series** X-ray Detector System device manual

Ross A. Doyle and Justin Anderson

Rayonix L.L.C.
1880 Oak Avenue, Ste. 120
Evanston, IL 60201 USA
http://www.rayonix.com

Table of Contents

Introduction to the Manual

Welcome and thank you for choosing our X-ray instruments.

This is intended to be a complete manual for users of Rayonix's High Speed series X-ray Detectors, based on custom-designed frame transfer CCD modules. These detectors include: MX HS, LX HS, and SX HS detectors. This manual also covers the software control of this equipment via the program *capxure*. At the time of writing, the latest *capxure* release was version 0.9.1.

Not all items in the manual are applicable to the equipment you are using. Also, the program *capxure* is highly configurable and the way you are operating it may look different (e.g. different buttons and more or fewer windows or tabs). Please check the chapter and section overview that you are reading before trying to apply the information.

Many detectors will be operated by one-time users; these users should start with the instructions in Chapters 1 and 2, and remember not make physical changes to the detector (cabling, etc.) or software configuration changes without the permission of the beamline or laboratory staff.

Explanations about hardware and software are side-by-side when they cannot be separated in any natural way. In these cases the user must shift attention back and forth between the computer and the detector.

We are constantly trying to improve what we provide to you. Thus, suggestions and problems about the manual, and software bugs reports and feature requests are welcome. Suggestions for the manual can be emailed directly to our general email account at info@rayonix.com. For software bug reports and new feature requests, please see our bug server at: http://www.rayonix.com/bugs. Create an account and log in to report a bug. Please include as much detail about the bug as possible.

Which equipment do I have?

In order to subdivide the equipment for explanatory purposes, the following divisions are made. They differ in the number of overall components, i.e. one detector head with various numbers of equipment racks and coolers.

- **Large HS detectors include 4x4 array systems (or larger).** These are the MX300-HS and MX340-HS. The major components are detector head, vacuum pump, two cabinets with two coolers each, and one cabinet containing power supply and computer.

- **Medium HS detectors include the 3x3 MX225-HS.** The major components are detector head, vacuum pump, one cabinet with two coolers, and one cabinet containing power supply and computer.

- **Small HS detectors include four or fewer modules in various arrays.** These are the 2x2 array MX170-HS, the 1x3 array LX255-HS, the 1x2 array LX170-HS, and various versions of single module SX HS systems (SX30-HS and SX85-HS). The major components are the detector head, vacuum pump, and one rack containing one cooler, power supply, and computer.

Type conventions used in the manual

Courier font (e.g. "`capxure --help`") is used to indicate commands to be entered into a terminal window, or characters copied from a terminal window.

Chapter 1: Capturing data with capxure

Introduction

capxure is the GUI software for collection and control of the Rayonix high-speed CCD detector (HS Series). The software allows the user to collect single frames, a series of frames or a detailed dataset of frames. The user may also control detector settings such as cooler power and temperature set-point, readout speed and frame resolution. Also included is image viewing which will display collected frames - simultaneously in separate windows if there is more than one detector present. Tools are available which allow the user to display statistics for part or the entire frame. The software interfaces with the Rayonix high-speed detector library ("RxDetector" in *craydl*) or may be set to monitor a folder for new images to display. There is also a feature whereby the GUI will monitor an EPICS ("Experimental Physics and Industrial Control System" software) process variable for indication of a new image and then retrieve the image for display from EPICS itself.

Installation

The program should be pre-installed on the controller computer, which ships with the detector. It is distributed as an RPM file and can be installed with the command (run as "root" user) "rpm –i filename.rpm". Upgrading to a new version can be accomplished with the command (again as "root") "rpm –U filename.rpm". Installing or upgrading will automatically make a backup of any configuration files changed on site. capxure depends on Qt and the boost libraries as well as the Rayonix craydl library and any of it's dependencies.

Starting the program

Typically the program is started by logging in to the detector pc as Rayonix HS User (username `hsuser`, password `hsuser`), opening a terminal window by right clicking on desktop and selecting "Opening in terminal window," and typing `capxure`.

Command Line Options

Usage:

```
capxure [-h | --help (help)] [-v (version)] [-c </config/file/path>]
[-V (view only mode)] [-A (admin mode)] [-P (proprietary mode)] [-m
<directory to monitor> -p <monitor filter>]
```

Program Features

IMAGE MENU

The program's "Image" menu has the following options:

- Load frame
    - For loading a data frame into the display section of the image tab.

- Playback as video
  - Will playback the images as a video, in filename alphabetical order starting with the current frame.
- Playback as looping video
  - As above except playback the frames in a continuous loop. When reaching the end frame, the next frame will be the first filename alphabetically in the series.
- Stop video playback
  - Stops video playback of frames.
- Radial Plot as Q
  - To set the Radial Plot graph of the image tab to display distance units of “Q” ($\frac{4\,\pi\,\sin{(\theta)}}{\lambda}$)
- Radial Plot as pixels
  - To set the Radial Plot graph of the image tab to display distance units of pixels
- Write radial plot values to disk
  - Writes a file containing the values in the radial plot
- Scale display linearly
  - Shows the image with linear grayscale using the selected numbers
- Scale display as log
  - Shows the image with log grayscale
- Scale display with gamma value
  - Prompts to enter a gamma value for gamma grayscaling the image
- Show Radial Plot CSV
  - Displays a window with a comma-separated-value list of the plot’s values (distance to bin start, mean value of pixels in bin).
- Set Line Plot width
  - Prompts for the width to use in plotting pixel values about the user-drawn line
- Invert pixels
  - Inverts the pixel gray scale display so that more intense pixels are lighter than less intense pixels.
- Show image header
  - Displays a dialog with information about the displayed image.

STATUS BAR

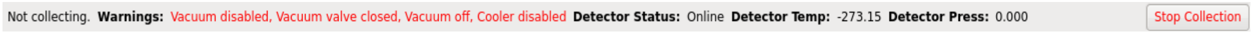

Not collecting. **Warnings:** Vacuum disabled, Vacuum valve closed, Vacuum off, Cooler disabled **Detector Status:** Online **Detector Temp:** -273.15 **Detector Press:** 0.000

Figure 1 - Capxure Status Bar

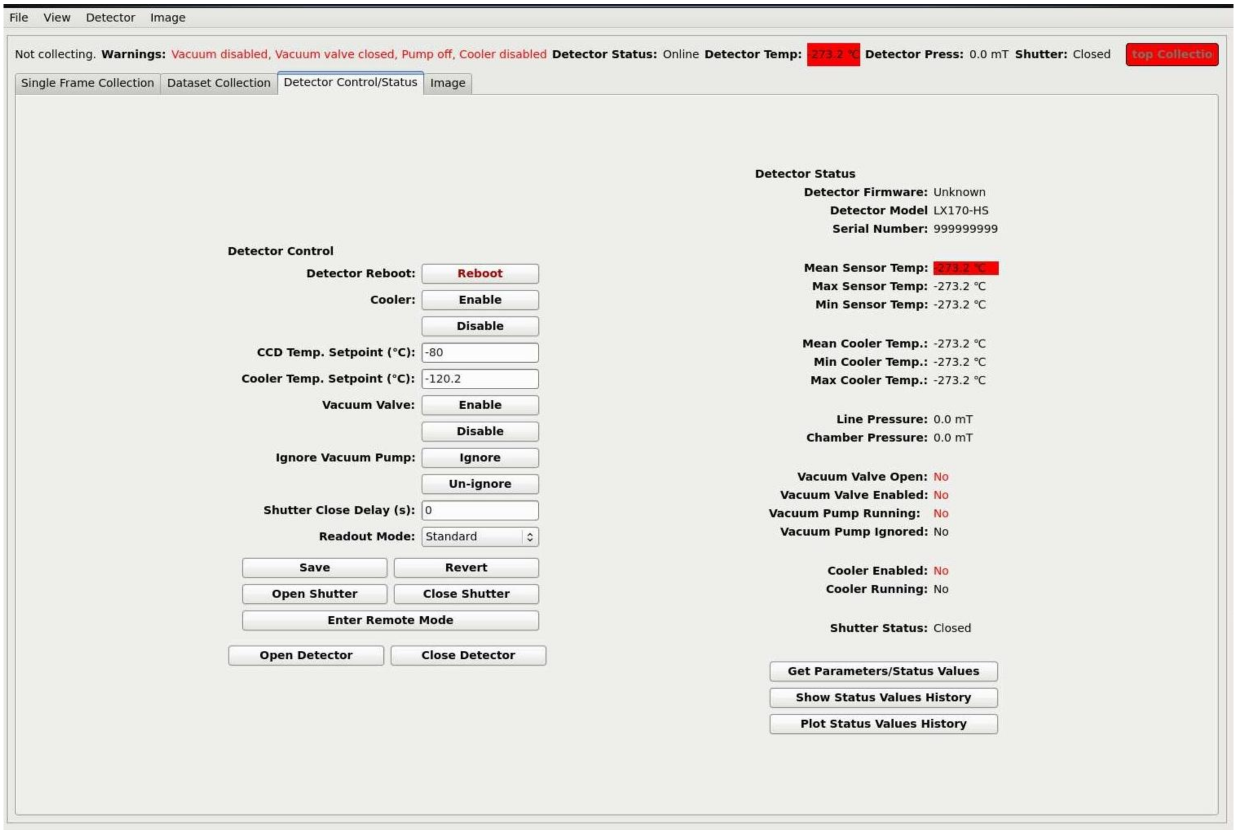
At the top of the program window is located a status bar which displays information about the state of image collection and the detector. The collection status is the first item on the left-hand side. It may read “Not collecting”, “Collecting single frame”, “Collecting dataset”, “Monitoring folder” or “Monitoring EPICS”, depending on the type of collection that is occurring. Next is the “Warnings” field, which will show any warnings for the system such as “Vacuum disabled” or “Cooler disabled”. Last there are the items “Detector Status”, “Detector Temp” and “Detector

Press” which will display the detector status, temperature and pressure, respectively. The “Detector Status” may have values such as “Offline”, “Online” or “Error”. There is also a “Stop Collection” button on the right side of the status bar so that collection can easily be stopped from any tab in the program.

USING TABS IN CAPXURE

With default settings, the program displays the following tabs in the main window: Single Frame Collection, Detector Control/Status, and Image. The program can also be configured to have the Dataset Collection tab. Tabs can be broken off the main window by double clicking. They can be replaced to the main window by clicking the X in the top corner of the broken off tab.

Detector Control Tab



Not collecting. **Warnings:** Vacuum disabled, Vacuum valve closed, Pump off, Cooler disabled **Detector Status:** Online **Detector Temp:** -273.2 °C **Detector Press:** 0.0 mT **Shutter:** Closed

Detector Control

Detector Reboot:
Cooler:
CCD Temp. Setpoint (°C): -80
Cooler Temp. Setpoint (°C): -120.2
Vacuum Valve:
Ignore Vacuum Pump:
Shutter Close Delay (s): 0
Readout Mode: Standard

Detector Status

Detector Firmware: Unknown
Detector Model LX170-HS
Serial Number: 999999999

Mean Sensor Temp: -273.2 °C
Max Sensor Temp: -273.2 °C
Min Sensor Temp: -273.2 °C

Mean Cooler Temp.: -273.2 °C
Min Cooler Temp.: -273.2 °C
Max Cooler Temp.: -273.2 °C

Line Pressure: 0.0 mT
Chamber Pressure: 0.0 mT

Vacuum Valve Open: No
Vacuum Valve Enabled: No
Vacuum Pump Running: No
Vacuum Pump Ignored: No

Cooler Enabled: No
Cooler Running: No

Shutter Status: Closed

Figure 2 - Capxure Detector Control/Status tab

The “Detector Control” tab allows the user to set the detector’s readout mode, CCD temperature set-point, cooler temperature set-point and shutter close delay. On this tab the detector may also be rebooted or re-opened and the cooler and vacuum can be enabled or disabled. The user can view a dump of the current detector parameter and status values or plot the temperature and pressure history on a graph. There is a section on the right side of the tab showing the current detector, cooler and vacuum statuses.

REBOOT BUTTON ("Detector Reboot")

Click this button to reboot the detector. (Caution – This is not required during normal operation.)

COOLER ENABLE/DISABLE BUTTONS

These buttons will enable and disable the detector cooler. The status change should be reflected in the status area on the right side of the tab soon after pressing one of these buttons. During normal service, the cooler should remain enabled.

TEMPERATURE SETPOINT ("CCD Temp. Setpoint (C)")

This textbox displays the current detector target temperature. This value should not be changed from the default in normal operation. A typical default value is -70℃ to -80℃.

COOLER SETPOINT ("Cooler Temp. Setpoint (C)")

This textbox shows the current cooler set-point, which is the temperature at the cold head. This value should not be changed from the default (-273℃) in normal operation.

VACUUM VALVE ENABLE/DISABLE BUTTONS

(CAUTION – This button should not be used in normal service). These buttons will enable and disable the detector's vacuum valve. The status change should be reflected in the status area on the right side of the tab soon after pressing one of these buttons. You should not disable the vacuum valve unless directed to by Rayonix.

IGNORE VACUUM PUMP ("Ignore/Un-ignore" Buttons)

(CAUTION—This button should not be used in normal service). These buttons will cause the detector to ignore and un-ignore the vacuum pump status. The status change should be reflected in the status area on the right side of the tab soon after pressing one of these buttons. You should not ignore the vacuum pump unless directed to by Rayonix.

SHUTTER CLOSE DELAY

This is a delay (in seconds) between when the "close shutter" command is sent to the time that the detector is read out. The ideal value depends only on how fast the shutter mechanism works. If the value is too short, then streaking or other imaging artifacts can occur (because the detector is reading out during the shutter close). If the value is too large, the only downside is that the unnecessary delay wastes beam time.

READOUT MODE SELECTION

This box contains the options for desired readout mode. Normal mode (high speed) will read out the detector in a mode that is optimized for most scientific applications, including those requiring high speed. The following table shows all- available readout modes:

Setting number	Readout Mode	Gain	Speed	Read bits	File bits
0	Standard	norm	norm	16	16
1	High Gain	high	norm	16	16
2	Low Noise	high	med	16	16
3	HDR	high	low	18	32*

Setting number	Readout Mode	Gain	Speed	Read bits	File bits
4	Turbo	norm	highest	16	16
5	HDR16	high	low	18	16

Table 1 - Available readout modes for HS detectors. ***Note that some data analysis programs may need to be updated to read 32 bit marccd format images.***

SAVE/REVERT

Clicking the "Save" button will save the changes the user has entered (Such as Readout Mode). Pressing "Revert" will revert any changes back to their previous state.]

OPEN SHUTTER / CLOSE SHUTTER BUTTONS

These buttons are used to manually cause the detector shutter signal to change state (open/closed)

ENTER REMOTE MODE

This button is used to start the legacy remote server. See page 60 for more information. Alternatively, one could start the remote server without using capxure (quit capxure and then start the program *hsserver_legacy*).

OPEN DETECTOR/CLOSE DETECTOR BUTTONS

Use these buttons to open or close the connection to the detector. Used for testing or troubleshooting purposes.

DETECTOR STATUS

On the right side of this tab is information specific to the current detector. Here the user will find the detector's serial number, firmware revision number and model number. You will also find the current status of the cooling and vacuum.

GET PARAMETERS/STATUS VALUES

Press this button to pop up a window containing a list of the current detector parameters and status values.

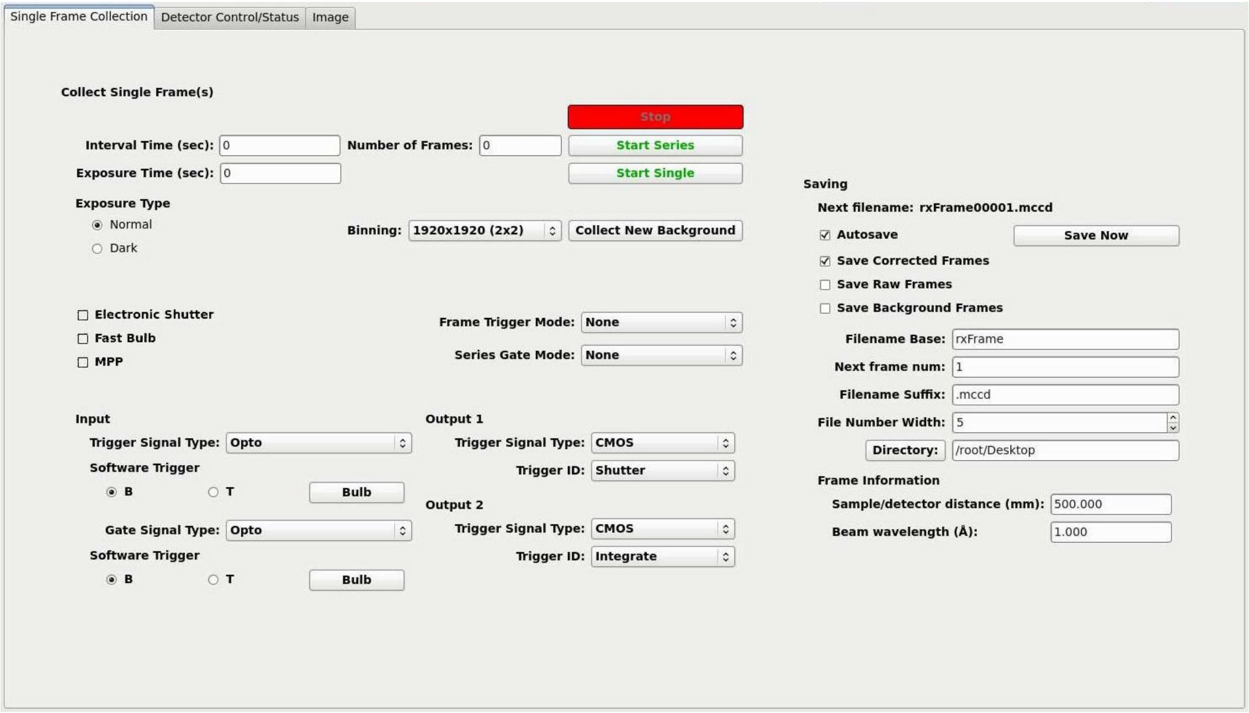
SHOW STATUS VALUES HISTORY

This button will open a dialog box containing the "STATUS_TITLES" and "STATUS_DATA" lines from the RxDetector log file. You will be able to see the history of the detector temperatures and pressures.

PLOT STATUS VALUES HISTORY

Pressing this button will display a plot of up to the last week's worth of temperature and pressure values for the detector. While looking at the plot, use the right mouse button to drag a box over a section of interest to zoom in to a narrower time range.

Single Frame Collection Tab



Collect Single Frame(s)

Interval Time (sec): 0
Number of Frames: 0
Exposure Time (sec): 0

Exposure Type
● Normal
○ Dark

Binning: 1920x1920 (2x2)

□ Electronic Shutter
□ Fast Bulb
□ MPP

Frame Trigger Mode: None
Series Gate Mode: None

Input
Trigger Signal Type: Opto
Software Trigger
● B ○ T
Gate Signal Type: Opto
Software Trigger
● B ○ T

Output 1
Trigger Signal Type: CMOS
Trigger ID: Shutter
Output 2
Trigger Signal Type: CMOS
Trigger ID: Integrate

Saving
Next filename: rxFrame00001.mccd
☑ Autosave
☑ Save Corrected Frames
□ Save Raw Frames
□ Save Background Frames
Filename Base: rxFrame
Next frame num: 1
Filename Suffix: .mccd
File Number Width: 5
Directory: /root/Desktop

Frame Information
Sample/detector distance (mm): 500.000
Beam wavelength (Å): 1.000

Figure 3 - Single Frame Collection Tab

On the “Single Frame Collection” tab are controls for collecting one single frame, a series of a specified number of identical frames and for continuously collecting identical frames. There are also settings on this tab for which frames to save (i.e. corrected, raw and background), save file name, exposure type to use (i.e. Normal, Dark or Test Image) and binning.

SINGLE FRAME COLLECTION

To collect a single frame, first choose an Exposure Type:

- “Normal” will output a shutter signal if Output 1 or Output 2 are set Trigger ID “Shutter.”
- “Dark” v

Choose the Binning (e.g. 1920x1920 (2x2)), and set various options, such as triggering options and whether to save frames and which frame types to save. The user should enter the filename if they are saving frames. Finally, enter the Exposure Time and click on the “Start” button. The frame will be collected, optionally saved and displayed on the “Image” tab.

COLLECT SERIES

Series collection will collect the entered number of frames with identical resolution. Choose an Exposure Type “Normal” or “Dark” as described above. Enter the desired number of frames in the “Number of Frames” box. Click “Start Series” button. (Refer to the Series Data Collection Example table on page 16 and the Timing Diagrams on page 73 for useful information.)

SETTINGS

- Exposure Time (sec)
    - The exposure time, in seconds.
- Interval Time (sec)
    - The interval time, in seconds, for series data collection.
- Exposure Type
    - Normal
        - Collect a regular X-ray illuminated frame (detector will output a shutter signal if Output 1 or Output 2 are set Trigger ID “Shutter”).
    - Dark
        - Collect a dark (non-illuminated) frame (will not output the shutter signal even if an output is set to Trigger ID “Shutter”).
- Binning
    - Here the user can choose the desired resolution (binning) for the collection.
- Left side checkbox options
    - Electronic Shutter
        - Normally should be clicked on. Causes frame transfer at both beginning and end of integration, when in any non-frame transfer mode (timed exposures or bulb mode). Necessary for shutterless data collection.
    - Fast Bulb
        - Normally should be clicked on. Used in Bulb mode. On=Keep output shutter signal high during entire series. Off=Shutter opens and closes for each frame.
    - MPP
        - Should be used for long (greater than 10 sec) bulb mode or triggered exposures to reduce the dark current. When set On, there is a 1sec delay between trigger and frame transfer (either for frame trigger or bulb trigger), because switching to MPP mode takes time.
        - Unnecessary for short (less than 10sec) bulb mode or triggered exposures.
        - Automatically controlled (no matter the state of this checkbox) when the detector controls the frame timing. When timed data frames are greater then 10sec, MPP mode is used.
- Frame Trigger Mode
    - Option “None” – no frame trigger, to use detector internal timer.
    - Option “Frame” – rising edge of input frame trigger causes electronic shutter followed by readout and simultaneous image acquisition. *See timing diagram on page 74*.
    - Option “Bulb” – images are timed by electronic shutter / image acquisition when rising edge input trigger received, then ending when falling edge received. Like an old-fashioned camera bulb mode.
    - *Note that selecting “Frame” or “Bulb” will **require** the use of a Frame Trigger signal, and without it, no frames will be collected.*
- Series Gate Mode
    - Option “None” – no series gate, begin series immediately.
    - Option “Start” – series begins upon rising edge of input Series Gate signal.

    - Option“Gate” – series begins with rising edge of Series Gate input signal, and can be paused and resumed with falling edge and rising edge.
    - *Note that selecting “Start” or “Gate” will* ***require*** *the use of a Series Gate signal, and without it, no frames will be collected.*
- Input
    - Trigger Signal Type options *(used for input Frame Trigger)*
        - Opto *(either Opto or Opto Inverted recommended to protect detector from noise or improper voltage on input trigger signal)*
        - Opto Inverted
        - CMOS Pulldown
        - CMOS Pullup
        - CMOS Pulldown Inverted
        - CMOS Pullup Inverted
        - Software
    - Software Trigger *(used for input Frame Trigger)*
        - B: For bulb-like behavior of software trigger button. Equivalent of rising edge when clicking the button and falling edge when the button is released.
        - T: For time-like behavior of software trigger button. First click is equivalent of rising edge signal and next click is equivalent of falling edge signal, and so on.
        - Bulb button: Clicking this button has the same function as the detector receiving an input Frame Trigger signal, when the Frame Trigger Signal Type is set to “Software.” Useful for testing purposes.
    - Gate Signal Type *(used for input Series Gate)*
        - Opto *(either Opto or Opto Inverted recommended to protect detector from noise or improper voltage on input trigger signal)*
        - Opto Inverted
        - CMOS Pulldown
        - CMOS Pullup
        - CMOS Pulldown Inverted
        - CMOS Pullup Inverted
        - Software
    - Software Trigger *(used for input Series Gate)*
        - B: For bulb-like behavior of software trigger button. Equivalent of rising edge when clicking the button and falling edge when the button is released.
        - T: For time-like behavior of software trigger button. First click is equivalent of rising edge signal and next click is equivalent of falling edge signal, and so on.
        - Bulb button: Clicking this button has the same function as the detector receiving an input Series Gate signal, when the Gate Signal Type is set to “Software.” Used for testing purposes.
- Output 1
    - Trigger Signal Type options *(used for Output signal 1)*
        - CMOS

        - Opto
    - Trigger ID: Output 1 can be defined to produce any of the following signals.
        - Shutter
        - Integrate
        - Frame
        - Line
        - Shutter Opening
        - Shutter Closing
        - Shutter Active
        - Trigger Wait
        - Trigger ACK *(Trigger acknowledge)*
        - Trigger Fall Wait
        - Trigger Fall ACK *(Trigger acknowledge)*
        - Gate Armed
        - Gate
- Output 2
    - Trigger Signal Type options *(used for Output signal 2)*
        - CMOS
        - Opto
    - Trigger ID: Output 2 can be defined to produce any of the following signals.
        - Shutter
        - Integrate
        - Frame
        - Line
        - Shutter Opening
        - Shutter Closing
        - Shutter Active
        - Trigger Wait
        - Trigger ACK *(Trigger acknowledge)*
        - Trigger Fall Wait
        - Trigger Fall ACK *(Trigger acknowledge)*
        - Gate Armed
        - Gate
- Frame Information
    - Distance (mm) – Detector distance value entered here will be incorporated into the frame header data. Note, this field is linked to the identical one on the "Dataset Collection" tab and changing the value here will automatically change it in the other.
    - Wavelength (Å) – Beam wavelength value entered here will be incorporated into the frame header data. As with "distance", this field is linked to the identical one on the "Dataset Collection" tab and changing the value here will automatically change it in the other.
- Saving
    - Save Frames
        - Check this box to save collected frames to disk
    - Save Corrected Frames

        - Check this box to save the corrected frames to disk (typically users will use this option and this option only)
    - Save Raw Frames
        - Check this box to save uncorrected frames to disk. This is useful for troubleshooting image correction.
    - Save Background Frames
        - Check this box to save the background frames to disk. This is useful for troubleshooting background frame issues.
- Filename Options
    - Filename base
        - This will be the base of the filename used when saving frames to disk.
    - Filename suffix
        - This text will be appended to the end of the filenames saved to disk.
    - File Number Width
        - The number of digits to use for the frame serial number part of the filename. e.g. If set to 4, frame #1 would be 0001.
    - Next frame num
        - The frame number of the next frame to be collected
    - Next filename
        - This is the filename that will be used for the next frame saved to disk.
    - Save directory
        - The directory to save the frames in. Clicking the "Browse" button underneath this textbox will allow the user to browse for the directory to use.

COLLECT NEW BACKGROUND

Pressing this button will cause the detector to collect a new background frame. Normally background frames are only collected if the frame parameters change or if they become stale.

STOP

Hitting the "Stop" button will stop continuous or series frame collection after the current frame is complete.

SERIES DATA COLLECTION EXAMPLES

Exposure Time	Interval Time	Frame Trigger Mode	Result
0	0	None	Frame transfer exposures at the fastest possible rate for selected binning
0	> 0	None	Timed exposures at the specified interval time, separated by 1ms frame transfer. Note that if Interval time is too short (shorter than fastest possible rate for selected binning), then the actual Interval time will not be the selected Interval time; it will be slower than fastest possible rate, because interval timer timeouts will be ignored when detector is in busy (readout) state.

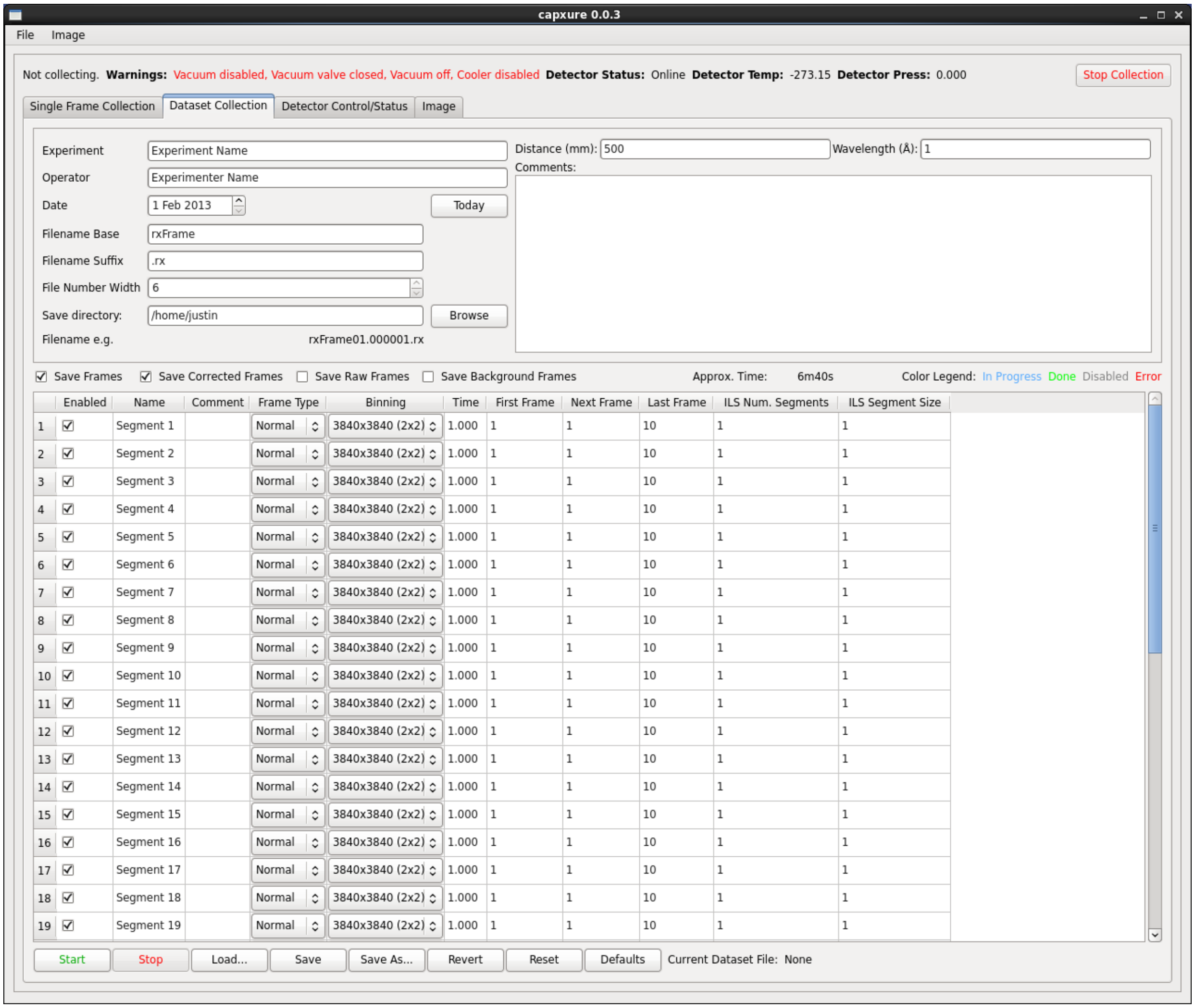
Exposure Time	**Interval Time**	**Frame Trigger Mode**	**Result**
> 0	0	None	*(Not commonly used)* Timed exposures separated by the full readout time
> 0	> Exposure Time	None	Timed exposures on specified intervals
0	N/A	Frame	Frame transfer triggered by rising edge of input trigger
> 0	N/A	Frame	Timed exposures triggered by rising edge of input trigger
N/A	N/A	Bulb	"Bulb Mode." Timed exposures started by rising edge of input trigger, and ended (readout) by falling edge.

Table 2 - Series data collection example parameters and results

Notes: Please refer to the example timing diagrams on page 73 for additional information. In all examples, set **Series Gate Mode** to "None" for no gating, "Start" to start data collection when Input 2 rising edge received, or "Gate" for Input 2 retriggerable gate (rising edge causes sensitivity to Input 1 triggers or Interval timer, falling edge causes insensitivity).

Troubleshooting tip – if Frame Trigger Mode or Series Gate Mode is set to options besides "None," then the proper Input signal(s) is required to collect any data frames.

Dataset Collection Tab



Not collecting. **Warnings:** Vacuum disabled, Vacuum valve closed, Vacuum off, Cooler disabled **Detector Status:** Online **Detector Temp:** -273.15 **Detector Press:** 0.000

Experiment	Experiment Name
Operator	Experimenter Name
Date	1 Feb 2013
Filename Base	rxFrame
Filename Suffix	.rx
File Number Width	6
Save directory:	/home/justin
Filename e.g.	rxFrame01.000001.rx

Distance (mm): 500 Wavelength (Å): 1

Comments:

☑ Save Frames ☑ Save Corrected Frames ☐ Save Raw Frames ☐ Save Background Frames Approx. Time: 6m40s Color Legend: In Progress Done Disabled Error

	Enabled	Name	Comment	Frame Type	Binning	Time	First Frame	Next Frame	Last Frame	ILS Num. Segments	ILS Segment Size
1	☑	Segment 1		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
2	☑	Segment 2		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
3	☑	Segment 3		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
4	☑	Segment 4		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
5	☑	Segment 5		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
6	☑	Segment 6		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
7	☑	Segment 7		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
8	☑	Segment 8		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
9	☑	Segment 9		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
10	☑	Segment 10		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
11	☑	Segment 11		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
12	☑	Segment 12		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
13	☑	Segment 13		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
14	☑	Segment 14		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
15	☑	Segment 15		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
16	☑	Segment 16		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
17	☑	Segment 17		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
18	☑	Segment 18		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1
19	☑	Segment 19		Normal	3840x3840 (2x2)	1.000	1	1	10	1	1

Current Dataset File: None

Figure 4 - Dataset collection tab

By default this tab will not appear in the program window. To enable this tab, quit the program, edit the file /opt/rayonix/configuration/capxure.conf (make a backup copy before making any changes) and set enableDatasetTab = true. Then restart the program.

EXPERIMENT INFO AND COMMENTS

- Experiment
    - Enter a name for the dataset/experiment.
- Operator
    - Enter the user name or initials.
- Date
    - Enter the date of the experiment. The user can click on the “Today” button to the right to automatically enter today’s date.
- Filename base
    - This will be the base of the filename used when saving frames to disk.

- Filename suffix
    - This text will be appended to the end of the filenames saved to disk.
- File number width
    - The number of digits to use for the frame serial number part of the filename. e.g. If set to 4, frame #1 would be 0001.
- Save directory
    - The directory to save the frames in. Clicking the “Browse” button to the right of this textbox will allow the user to browse for the directory to use.
- Filename e.g.
    - This is an example of what the filename used for frames saved to disk will look like.
- Distance (mm)
    - You can enter a value for the sample to detector distance here and it will be incorporated into the frame header data. Note, this field is linked to the identical one on the “Single Frame Collection” tab and changing the value here will automatically change it in the other.
- Wavelength (Å)
    - Here, you can enter a value for the beam wavelength and it will be incorporated into the frame header data. As with “distance”, this field is linked to the identical one on the “Single Frame Collection” tab and changing the value here will automatically change it in the other.
- Comments
    - Enter any comments regarding this dataset/experiment.

SAVE OPTIONS

- Save Frames
    - Check this box to save collected frames to disk
- Save Corrected Frames
    - Check this box to save the corrected frames to disk (typically users will use this option and this option only)
- Save Raw Frames
    - Check this box to save uncorrected frames to disk. This is useful for troubleshooting image correction.
- Save Background Frames
    - Check this box to save the background frames to disk. This is useful for troubleshooting background frame issues.

Approx. Time

The program will display the total dataset collection time here, not including any possible dead time between frames or segments and not including any needed background frame collections. This value is meant to give the user a general idea of how long the dataset will take to complete.

COLOR LEGEND

To the right of the save frame options is a dataset segment color legend. This legend tells the user what the segment highlight colors mean. White rows (with no highlight) have yet to be started. A blue highlight means the segment is in progress. Green highlighted segments have already been collected. A grayed-out segment is disabled and a red highlighted segment indicates there is an error with the segment.

DATASET SEGMENTS GRID

Each line in the grid contains the settings for the corresponding segment of collection. Following are the definitions of each column:

- (Leftmost column - unnamed)
  - The segment number
- Enabled
  - A checkbox which, when checked, enables the segment for collection
- Name
  - A name for the segment
- Comment
  - A comment for the segment
- Frame Type
  - Normal
    - For this segment, collect normal, illuminated frames
  - Dark
    - For this segment, collect dark, non-illuminated frames
  - STOP
    - Do not collect this segment or any segments after it
- Binning
  - Here the user can select the resolution (binning) to use when collecting this frame.
- Time
  - The exposure time to use for this segment (seconds)
- Frame 1
  - The first frame number to use for this segment (usually 1)
- Next Frame
  - The next frame that will be collected. For a new dataset, this will be equal to "Frame 1". This number is incremented as collection proceeds.
- Last Frame
  - The last frame number to collect for this segment. If "Frame 1" equals 1, this will be the number of frames to collect.
- ILS Num. Segments
  - This is an "interleaved segments" option whereby the user can collect frames from successive segments in a staggered manner. This field sets how many segments are in this interleaved segment group. For instance, if set to 3 then this segment and the two following segments are combined. Frames will be collected from this segment, then the following segment and then the next following segment. After that, collection will come back to this segment and repeat as before. Note: In our example of three segments being interleaved, this parameter would be ignored in the next two segments.
- ILS Segment Size

  - This “interleaved segment” parameter indicates the number of frames to collect in this segment before moving on to the next segment in the interleaved segment group. It only makes sense to have a value here if this segment is part of a group of interleaved segments.
  - Example: In frame #1 we have “ILS Num. Segments” set to 3 and “ILS Segment Size” set to 2. Frames 2 and 3 have “ILS Segment Size” set to 3 and 4, respectively. The frames would be collected as follows:
    - 1, 1, 2, 2, 2, 3, 3, 3, 3, 1, 1, …..

BOTTOM BUTTONS

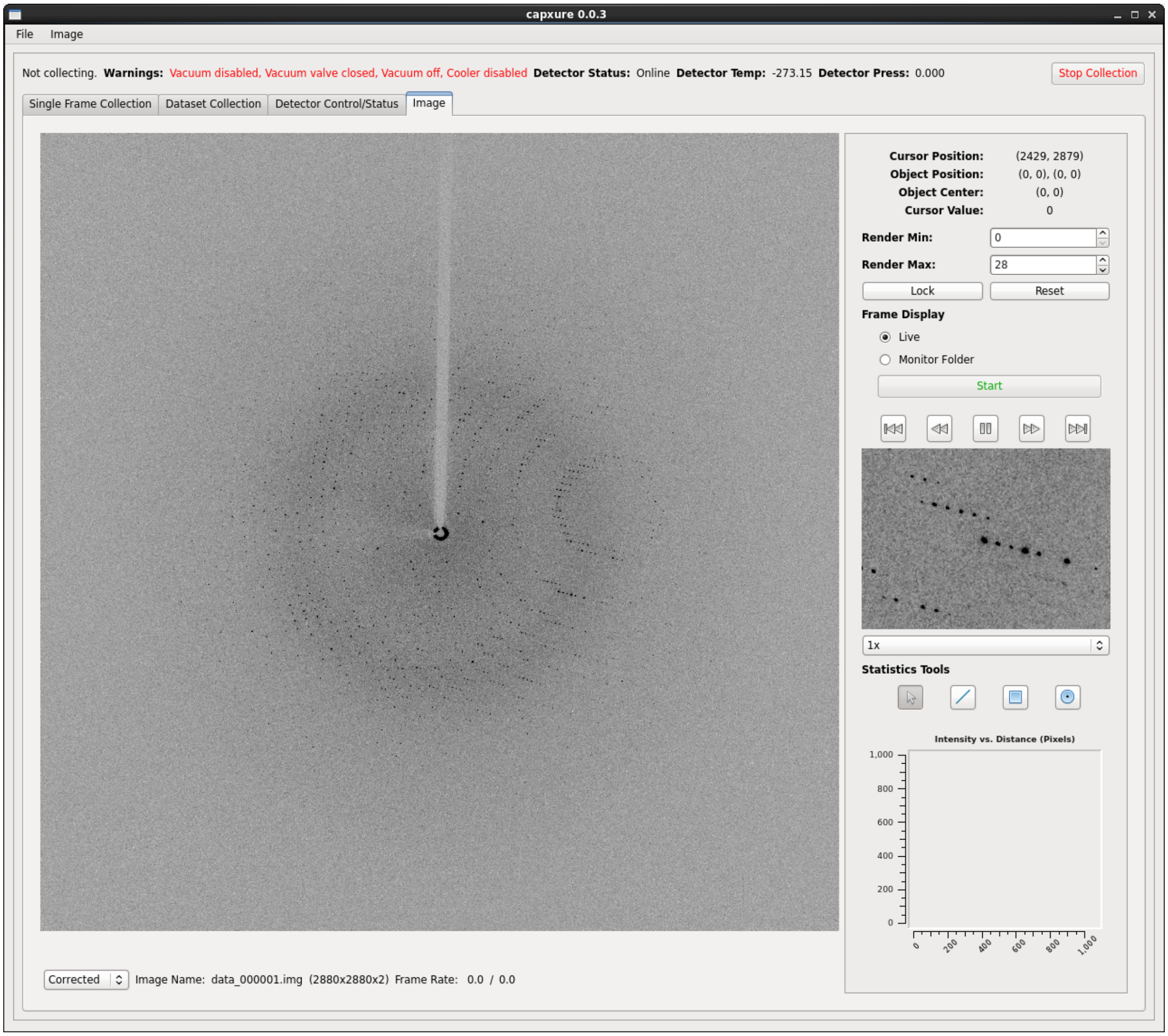
At the bottom of the Dataset Collection tab are the following buttons:

- “Start”
  - Starts collection of the dataset
- “Stop”
  - Stops collection of the dataset
- “Load…”
  - Prompts for a saved dataset parameters file to load from disk
- “Save”
  - Saves the dataset parameters to a file on disk. Will save to current dataset parameters file or, if none loaded, will prompt for one to use.
- “Save as…”
  - Prompts for dataset file path to save the dataset parameters to disk.
- “Revert”
  - Reverts the dataset parameters back to the last loaded or saved file values
- “Reset”
  - Resets each segment’s “Next Frame” value to its “Frame 1” value. This will effectively allow the user to restart this dataset collection from the beginning.
- “Defaults”
  - Resets all of the dataset parameters to their default values. Useful when starting a new dataset from scratch.

CURRENT DATASET FILE

To the right of the buttons at the bottom of the screen is a text field that displays the current dataset parameters file being used. This is the filename that was last loaded from or saved to disk.

Image Viewer Tab



capxure 0.0.3

Not collecting. **Warnings:** Vacuum disabled, Vacuum valve closed, Vacuum off, Cooler disabled **Detector Status:** Online **Detector Temp:** -273.15 **Detector Press:** 0.000

Cursor Position: (2429, 2879)
Object Position: (0, 0), (0, 0)
Object Center: (0, 0)
Cursor Value: 0

Render Min: 0
Render Max: 28

Frame Display
● Live
○ Monitor Folder

Corrected Image Name: data_000001.img (2880x2880x2) Frame Rate: 0.0 / 0.0

Figure 5 - Image view tab

The “Image” tab contains an area for image display and tools for graphing statistics about an area of the image. On this tab are also options for how the program will get frames to display here. You can set the “frame rendering limits” to control the contrast. There is also a selector for which type of frame is being displayed, i.e. “Corrected”, “Raw” or “Background”.

IMAGE AREA

The largest area on this tab is where data frames will be displayed. Areas that have no image data to display will be white.

COORDINATE AREA

At the top of the right-hand side of this tab are the image pixel coordinates, which are updated as the user interacts with the image. The definitions are:

- Cursor Position
  - The (x, y) pixel coordinates of where the cursor currently is over the image.
- Object Position
  - This displays the top-left, bottom-right ((x_0, y_0), (x_1, y_1)) corner pixel locations of an object drawn on the image by the user (e.g. Line Tool, Box Tool, Radial Plot Tool).
- Object Center
  - Here is displayed the coordinate of the center of the object drawn on the image by the user (e.g. Box Tool, Radial Plot Tool).
- Cursor Value
  - This is the intensity value of the pixel under the cursor.

Render Min/Max

Here you can set the minimum intensity value, below which will be shown as white pixels. You can also set the maximum intensity value, values above which will be shown in black. Pixels between these values will be scaled in gray.

LOCK (Rendering Limits)

Activating the “Lock” button here will lock the rendering limits so that they will not be recalculated when a new frame is displayed.

Reset (Rendering Limits)

Pressing the “Reset” button will rescale the rendering limits to their calculated optimum values.

FRAME DISPLAY

Here the user can choose how the images displayed in this tab are obtained:

- Live
  - Images are displayed as they are received from the Rayonix high-speed CCD detector library.
- Monitor Folder
  - If this option is selected, the user will be prompted for a folder path and filename base, which the program will monitor. The program will check the folder at a configurable interval for files matching the base name (the filenames will match as long as they start with the basename). The most recently modified image file will be displayed and the program will continually check for new image files.
- Monitor EPICS PV (Optionally activated via config file option)
  - EPICS is the “Experimental Physics and Industrial Control System” software in use at some synchrotrons. This option will tell the program to monitor an EPICS “Process Variable” for change and, when it changes, to load an image from EPICS. When selecting this option, the user will be prompted to supply the

EPICS record name prefix (e.g. “13RAYONIX1”), EPICS camera record name (e.g. “cam1”) and the EPICS image record name (e.g. “image1”). The program will register with the EPICS program to be told when the e.g. 13RAYONIX1:image1:UniqueId_RBV variable changes, indicating that there is a new image ready. When that variable changes, the program will obtain the new image from EPICS via the e.g. 13RAYONIX1:image1:ArrayData variable. Note: EPICS must be running and communicating with the detector for this option to work.

- “Start” button
  - After setting up folder monitoring or EPICS PV monitoring, click on this button to begin. The button will then change into a “Stop” button which, when clicked, will stop folder or EPICS monitoring.

IMAGE NAVIGATION BUTTONS

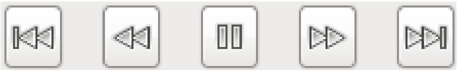


Figure 6 - Image navigation buttons

Under the frame display options is a set of buttons for navigating to previous/next frames on disk. Image cueing is available when collecting in “Live” mode if the images are being saved to disk. Press the “Pause” button to pause image display and the cue buttons will then be enabled. The user can display the previous frame, first frame, next frame or last frame on disk. The program determines next/previous frames by looking for the file in the working directory whose name is next/previous alphabetically and whose file type is the same as the displayed image. After the user is done browsing other frames, they can press the “Play” button to resume display of new images. The cue buttons will then be disabled again.

When collecting in Monitor Folder mode, image cueing is always available since the frames are already stored on disk. Use as with “Live” mode.

Image cueing is not available in EPICS PV Monitoring mode since the program does not know the filename or path.

ZOOM WINDOW

Under the image navigation buttons is the image zoom window. The zoom window displays a zoomed-in subset of the image. The zoom window can be centered by clicking on a point in the main image display or within the zoom window itself. The user can select the zoom level via the drop-down menu below the zoom window. At the 32x zoom level, intensity values are written on top of the pixels. As the user moves the pointer inside of the zoom window, the Cursor Position coordinates will be updated to reflect the position over the image.

STATISTICS TOOLS

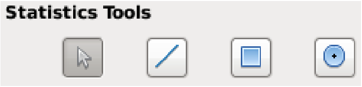


Figure 7 - Statistics tools for images

Under the zoom window is a set of icons corresponding to image statistics tools. They are defined as follows:

- Arrow Tool
  - The default tool. Used to select a point on the image as the center of the zoom window.
- Line Tool
  - Allows the user to draw a line over the image. A plot of pixel intensity vs. distance along the line will be drawn in the plot area. Right clicking on the line tool will allow the user to set the line width. Pixels within this distance from the line will be averaged together along the line.
- Box Tool
  - The box tool allows the user to draw a box over a portion of the image. A histogram will be displayed of the pixel count vs. intensity value for the pixels within the box. Also displayed are the mean and sigma for the intensities within the box. Clicking inside the box and holding the right mouse button will allow the user to reposition the box by dragging it to a new location.
- Radial Plot Tool
  - This tool is for marking the center point to use for displaying a radial plot such as is useful for SAXS data. The center point is indicated by the user's placement of the crosshair at the center of the drawn circle. Right clicking on the tool icon will prompt the user for settings to use when making the plot (i.e. Plot distance as Q or pixels, set number of bins, show plot CSV). Under the plot area are checkboxes for plotting the intensity and/or bin as $\log_{10}$. Clicking inside the circle and holding the right mouse button will allow the user to reposition the circle by dragging it to a new location.

DISPLAYED IMAGE NAME AND BUFFER SELECTION

Under the image display area is the filename of the currently displayed image. A filename will be displayed if the image has been loaded from or saved to disk. Otherwise, there will be a generic identifying name and timestamp for the image.

To the right of the image name is the displayed frame buffer selector. This drop-down menu allows the user to select the currently displayed image buffer, i.e. "Corrected", "Raw" or "Background". The selector is only available when displaying live images.

Next is displayed the image dimensions (pixels fast x pixels slow x depth in bytes). After that is a frame rate display which shows the display rate and the rate at which frames are being received from the RxDetector library.

Multiple Detector Display

If your installation has more than one detector, you can have the program display image feeds from the other detectors in separate windows. These extra windows are identical to the contents of the "Image" tab of the main program.

To enable this feature, set the number of detectors in the configuration file as such:

numDetectors = 3

In this example of 3 detectors, two additional image viewer windows will be opened when the program is started.

Currently, it is expected that the image viewer windows will be used in "folder monitoring" mode when more than one detector is present.

Configuration Options

The program reads a configuration file named "capxure.conf" from the directory the program was started from. The syntax of the configuration file is as follows:

```
[section name]
variableName = value
```

Currently, all values are given in the section named "[main]".

A sample configuration file ("capxure.conf") is included in the installation. The possible values for the configuration file are defined as follows:

Parameter Name	**Description**	**Default Value**
craydlConfigFile	(String). Path to Rayonix craydl library configuration file.	None (Let craydl find it).
numDetectors	(Integer). Number of cameras in use. Additional windows will be created for displaying frames from the additional cameras.	1
defaultNumSegments	(Integer). Number of segments to display in the dataset.	24
displayRawFrames	(true/false). Whether to display the raw frames in the frame display area.	true
displayBackgroundFrames	(true/false). Whether to display the background frames in the frame display area.	true
displayCorrectedFrames	(true/false). Whether to display the corrected frames in the frame display area.	true
capxureLogFile	(String). Path to save log file to.	capxure.log
capxureLogLevel	(String). Lowest log level to write. Options include FATAL, ERROR, WARNING, INFO, VERBOSE, DEBUG, TRACE.	INFO

enableEpicsMonitor	(true/false). Set to true to enable the EPICs monitoring controls on the Image tab.	false
enableDatasetTab	(true/false) Makes the dataset tab appear in the program.	false
enableContinuousButton	(true/false) Makes a Continuous button appear in the Single Frame collection tab. This button causes the program to image indefinitely, with no maximum frame limit.	false
videoFrameDelay_ms	(Integer). When playing back images as video, this is the delay between frame displays in milliseconds.	0
enableAdministratorMode	(true/false) Causes more options to appear in the program. Some controls can damage the detector, so use with caution.	false
printMonitoredDirectories	(true/false) When set true, the program will print out a message when a new directory is added to the watched directories list. This happens if users create new directories under the watched directory.	true
monitoredDirectoryPathFile	(String). When set (to some file), capxure will read this file, which should contain a text string that is the desired monitor directory path. This file can be changed during program usage and the program will then change the monitor folder to whatever the file now reads. Note: If both this file and the –m <path> command line option are both used, the program will write the command line <path> into this file when the program starts.	None

Table 3 - Capxure configuration options, usually found in capxure.conf

Chapter 2: Detector Hardware Setup

Note to one-time detector users: normally this procedure has been done for you. Also, users should not generally make any physical changes to the instrument (cables, etc.) without the beamline staff permission.

This section describes how to connect the MX HS detector system. If the detector is already connected and you want to start and use the system, please go to the following chapter.

How to connect and disconnect cooling gas lines

Because the cooling connections require tools to connect and disconnect, they should normally be the first connections made when connecting the system together (and last disconnections to be done when dismantling system).

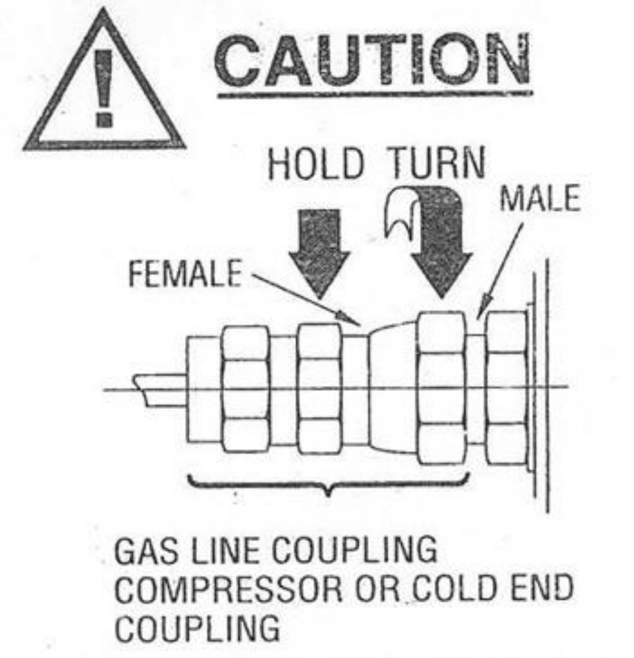


Figure 8 - Gas line coupling

- Caution—the gas is flammable and leaks must be vented immediately
- Use two wrenches when connecting or disconnecting gas lines
- Have caps ready to cap both valves when disconnected
- Gas line must be aligned, with minimal torque from the hose, when making or breaking a coupling
- Complete each disconnection or connection without delay
- When making connections, tighten to 14-16 N.m (10-12 lb-ft)

Setting up Small HS System connections from dismantled state

The “Small” HS systems have from one to four Rayonix frame-transfer CCD sensors in the arry, including the following systems:

- **SX30-HS** (Single module 30mm system)
- **SX85-HS** (Single module 85mm system)
- **LX170-HS** (1x2 array system)
- **LX255-HS** (1x3 array system)
- **MX170-HS** (2x2 array system)

The major components are shown in the following overview:

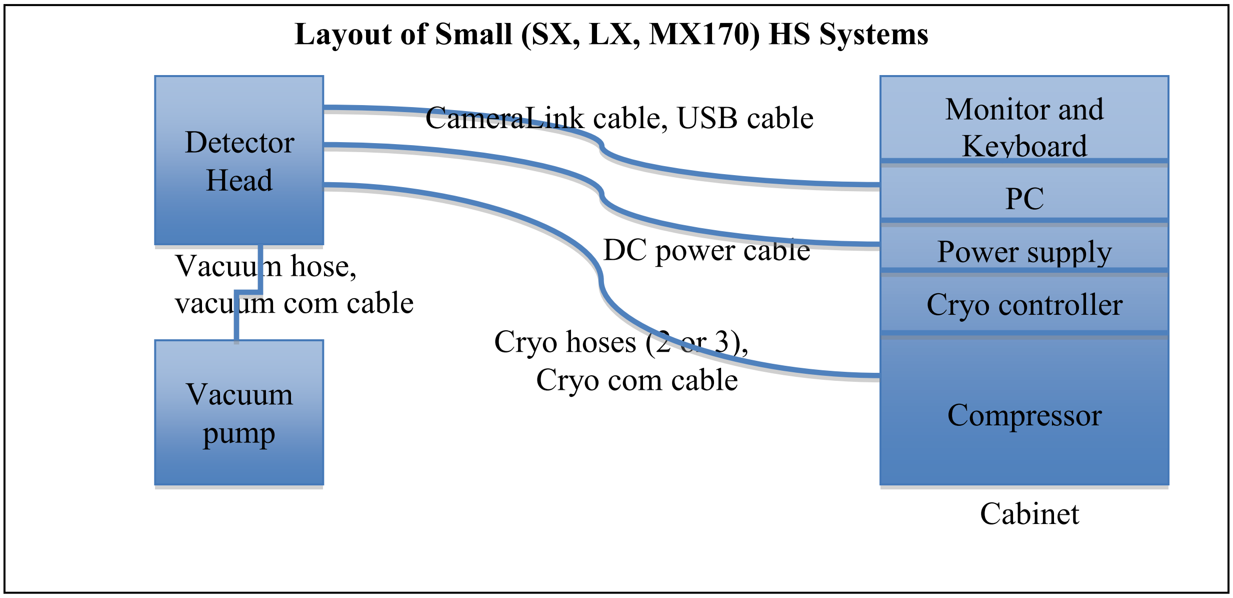

Layout of Small (SX, LX, MX170) HS Systems
Detector Head
CameraLink cable, USB cable
Monitor and Keyboard
PC
DC power cable
Power supply
Vacuum hose, vacuum com cable
Cryo controller
Cryo hoses (2 or 3), Cryo com cable
Vacuum pump
Compressor
Cabinet

Figure 9 – Hardware Overview of “Small” HS systems. ****Note there are three Cryo hoses for systems with 3 or 4 modules (LX255/MX170), but only two for systems with 1 or 2 modules(LX170, SX).***

1. Refer to page 28 to learn how to connect/disconnect cooling gas lines.

2. *(Skip this step for an SX series or LX170-HS detector.)* If the system is a 1x3 or 2x2 array (LX255-HS or MX170-HS), there are three cooling gas lines, two SUPPLY and one RETURN. A “Y” cryo adapter must be attached on the RETURN side fittings of the detector head, between Cryo A RETURN and Cryo B RETURN.

3. Connect cooling gas lines. “SEND” fittings always connect to “SEND” fittings (green color). “RETURN” fittings always connect to “RETURN” fittings (red color). When there are two RETURN fittings next to each other on the cabinet, it doesn’t matter which one is used.

4. Connect DC power cable from Detector Head to Power Supply.

5. Connect vacuum hose. Note: 1x1, 1x2, 1x3, and 2x2 HS detectors (SX, LX, and MX170-HS models) use ¼ inch male Swagelok VCO fitting on detector head. Standard vacuum pump supplied with the detector has a KF-40 (aka QF-40) fitting.

6. Connect CameraLink and USB connection between detector head and PC. CameraLink cable should be connected to **CameraLink 0** connector on PC. USB should be connected to USB 2.0 port on the computer, usually on a PCI card, not on the computer case or motherboard.

Setting up Medium/Large sized (3x3 and 4x4) HS System connections from dismantled state

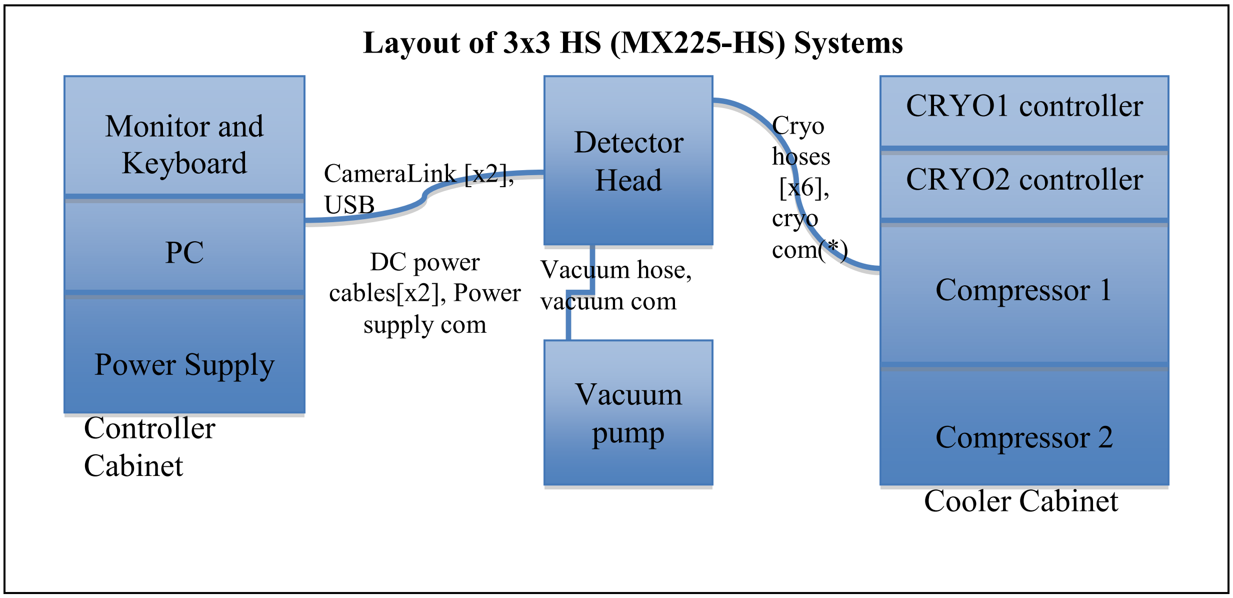


Figure 10 – Hardware Overview of 3x3 HS, the MX225-HS. ***Note: Alternatively, cryo com cable can connect between Cooler Cabinet and Controller Cabinet.***

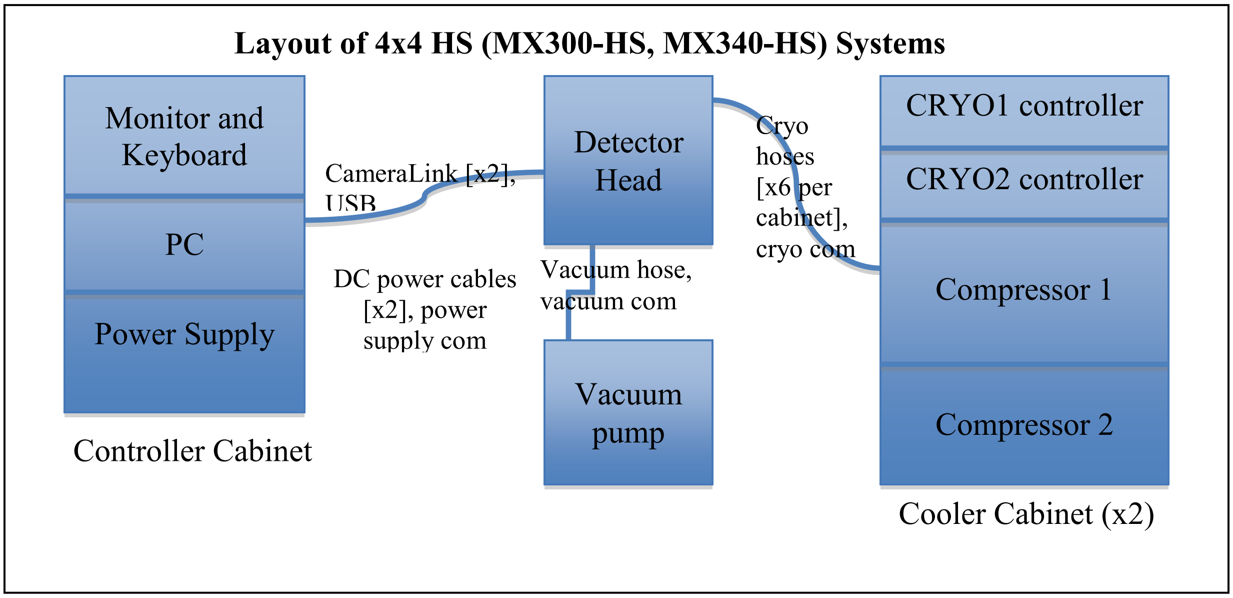


Figure 11 – Hardware Overview of 4x4 HS Detectors, including MX300-HS and MX340-HS

1. Refer to page 28 to learn how to connect/disconnect cooling gas lines.

2. If they are not yet attached, "Y" cryo fittings should be attached to the detector head before other connections. They are used on the detector head RETURN side fittings. They should be connected between Cryo A and B, and Cryo C and D (for 4x4 systems, Cryo D and E, and Cryo F and G).

3. Connect cooling gas lines. "SEND" fittings always connect to "SEND" fittings (green color). "RETURN" fittings always connect to "RETURN" fittings (red color).

4. Connect both power cables from Detector Head to Power Supply (on Power Supply / PC cabinet).

5. Connect vacuum hose. Note: Vacuum fitting type on 3x3 and 4x4 HS detectors — MX225-HS, MX300-HS, and MX340-HS — is KF-25 (aka QF-25). Vacuum fitting type on standard Rayonix pump is KF-40 (aka QF-40).

6. Connect CameraLink and USB connection between detector head and PC. CameraLink cable should be connected from CameraLink 1 and 2 on detector head to **CameraLink 0 and 1** connectors on PC. USB should be connected to USB 2.0 port on the computer, usually on a PCI card, not on the computer case or motherboard.

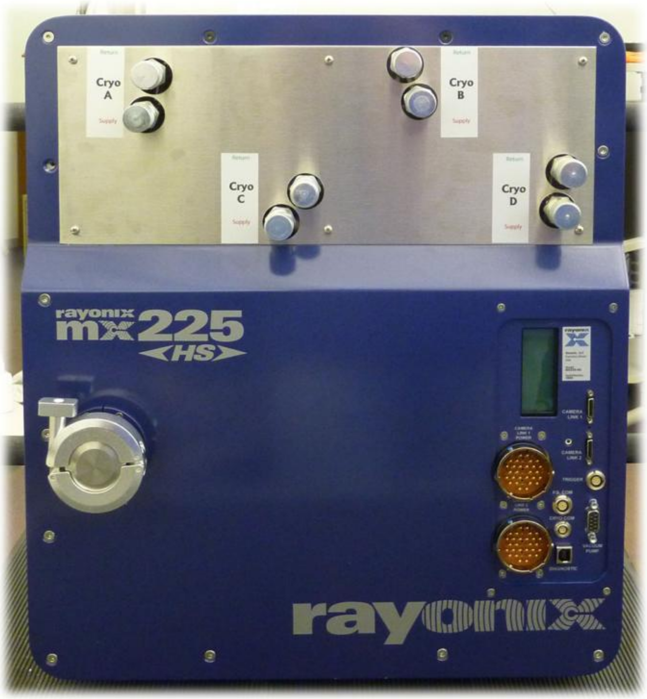


Figure 12 – Back view of MX225-HS detector head

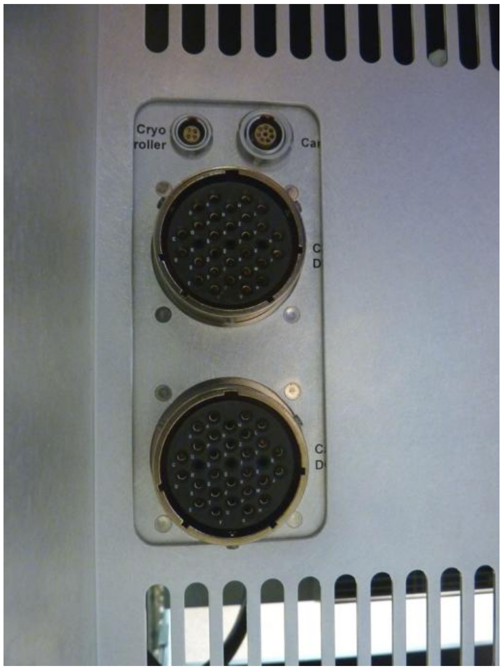


Figure 13 - Power supply connectors on Controller Cabinet

Chapter 3: Detector Startup

Note that the hardware setup (connecting hoses and cables) is explained in Chapter 2: Detector Hardware Setup.

Startup Procedure

1. **Make sure detector computer is already running (by logging in from other terminal or checking rack-mounted monitor/keyboard). If NOT running, Power on Controller Cabinet cabinet power. This switch is on the BACK of the cabinet, labeled "Rayonix Cabinet." Then boot up the data collection computer using the rack mounted monitor/keyboard.**

2. **Log in to the xray user account (hsuser, password hsuser), start the capxure software by typing "capxure".**

3. **Turn on the TMP (Turbomolecular vacuum pump).**

The following instructions are for Pfeiffer TMP system:

If the pump power is off, turn the green switch on (located back next to the power cable). It will take a few moments for the pump to initialize.

Press the button with the circle symbol on the front face of the pump to start (bottom right of Figure 14).

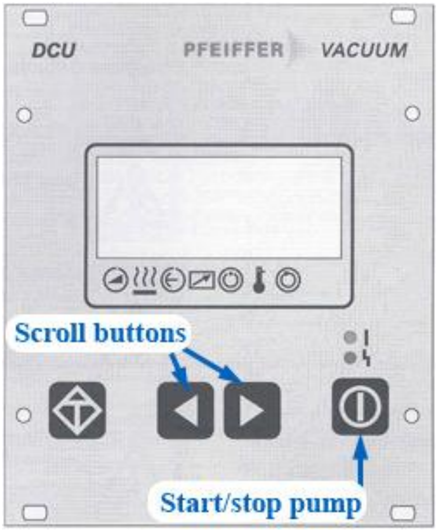


Figure 14 - Pfeiffer TMP controls

Use the Scroll buttons (left or right) shown on Figure 14 to go through the menus, until you find the display "**Act rotspd**" (Actual Rotation Speed). Watch **Act rotspd** increase slowly. When it reaches 1500 Hz (or is equal to **Set RotSpd**, if different than 1500 Hz), continue to the next step.

4. Turn on the detector power supply.

On smaller systems (SX, LX, or MX170-HS), use the Cryocontroller power switch, which also powers the Power Supply.

On medium or larger systems (3x3 or 4x4), the switch is located on the front of the Controller Cabinet, labeled “Main Power for Detector Power Supply” or “High Speed Mosaic Power Supply.”

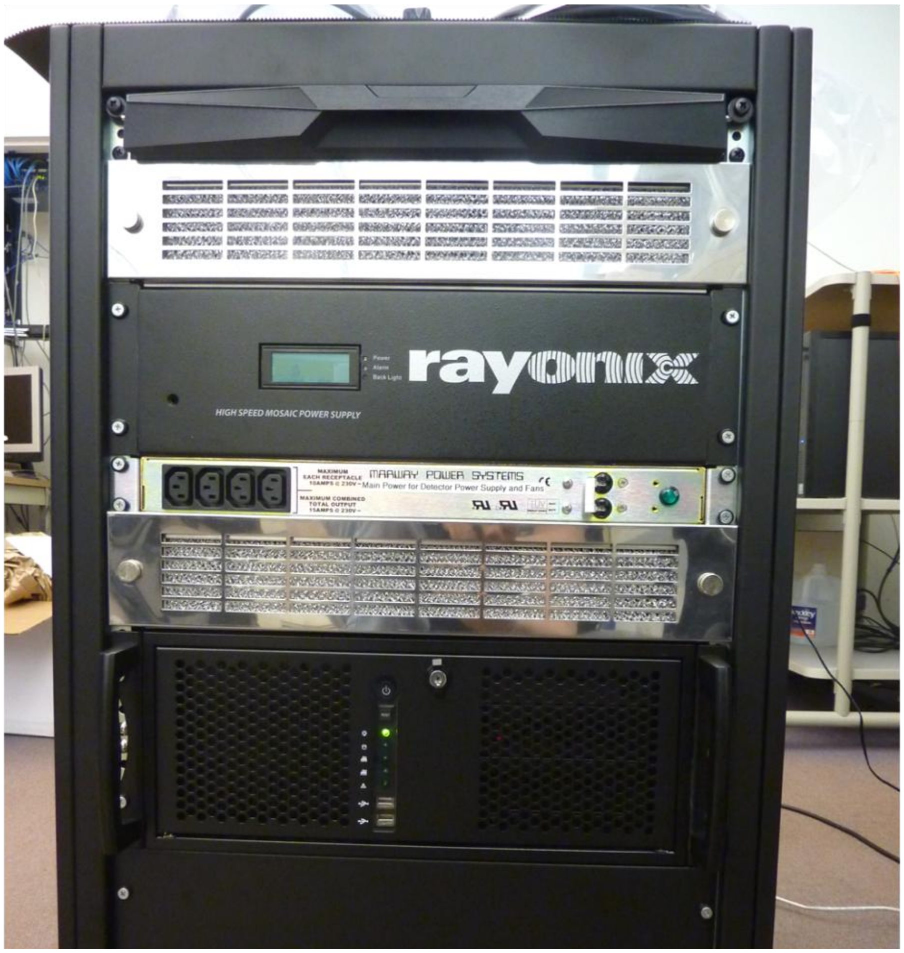

Power
Alarm
Back Light
rayonix
HIGH SPEED MOSAIC POWER SUPPLY
MAXIMUM EACH RECEPTACLE 10AMPS @ 230V~
MARWAY POWER SYSTEMS
Main Power for Detector Power Supply and Fans
MAXIMUM COMBINED TOTAL OUTPUT 15AMPS @ 230V~

Figure 15 – MX225-HS Controller Cabinet front view

5. ***(This step is for 3x3 and 4x4 systems only)* Power on the Cooler Cabinet(s). Then power on all of the Cryocontrollers.**

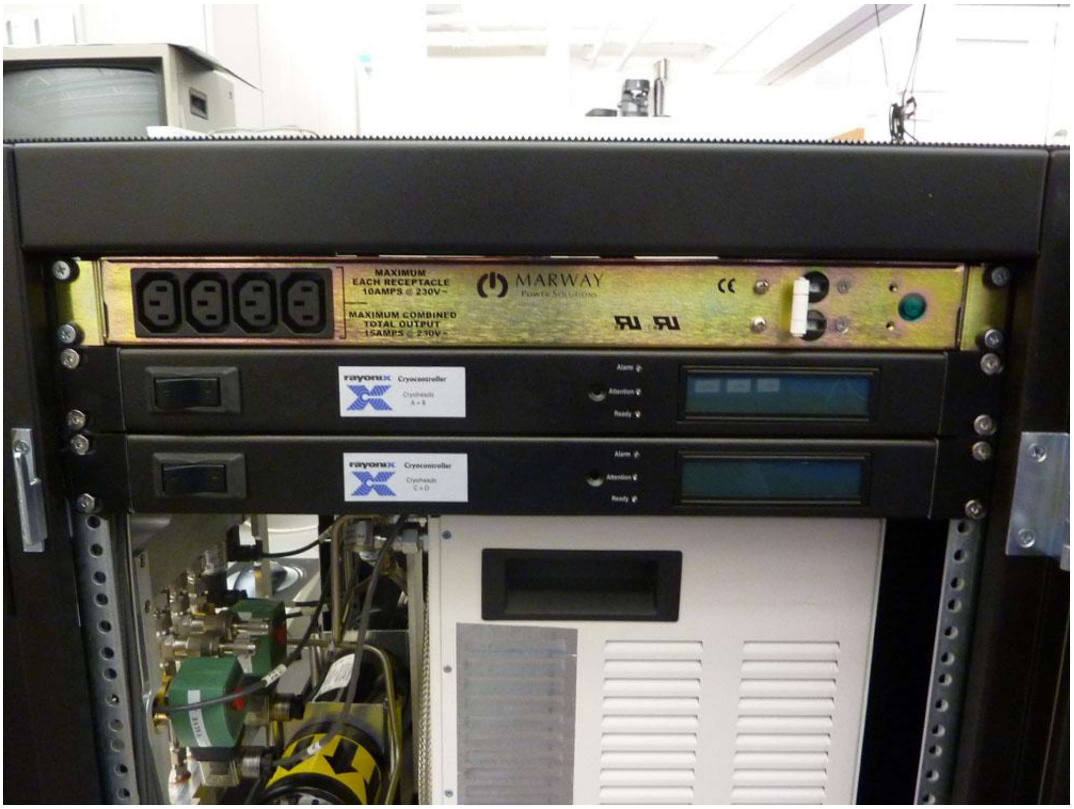


Figure 16 - Cooler Cabinet power switches

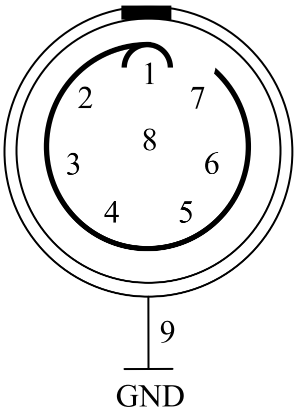
6. **Check vacuum pressure status in the capxure GUI. Pressure needs to be below 100mTorr for the coolers to come on.**

7. **Enable the cooling by clicking the button in the capxure detector control tab. Verify that the coolers came on, and wait ~3-4 hours for the detector to cool the detector to the operating temperature (-75C) before collecting data.**

Chapter 4. Detector Control Hardware

MX-HS Series Trigger Input and Output Connections

The trigger inputs and outputs for the MX-HS series detectors are done through a single LEMO style connector on the detector head labeled, "TRIGGER." The connector accepts an 8-pin LEMO plug, p/n FGG.1B.308.CLAD52. A plug to make such a cable is included with each new detector, normally found in the tan Rayonix tool bag.



Pin order as shown looking at solder cups.
Pin 1 indicated by notch and semicircle.

1: Trig-Out1 +
2: Trig-Out1 –
3: Trigger1 +
4: Trigger1 –
5: Trig-Out2 +
6: Trig-Out2 –
7: Trigger2 +
8:Trigger2 –
9: Ground

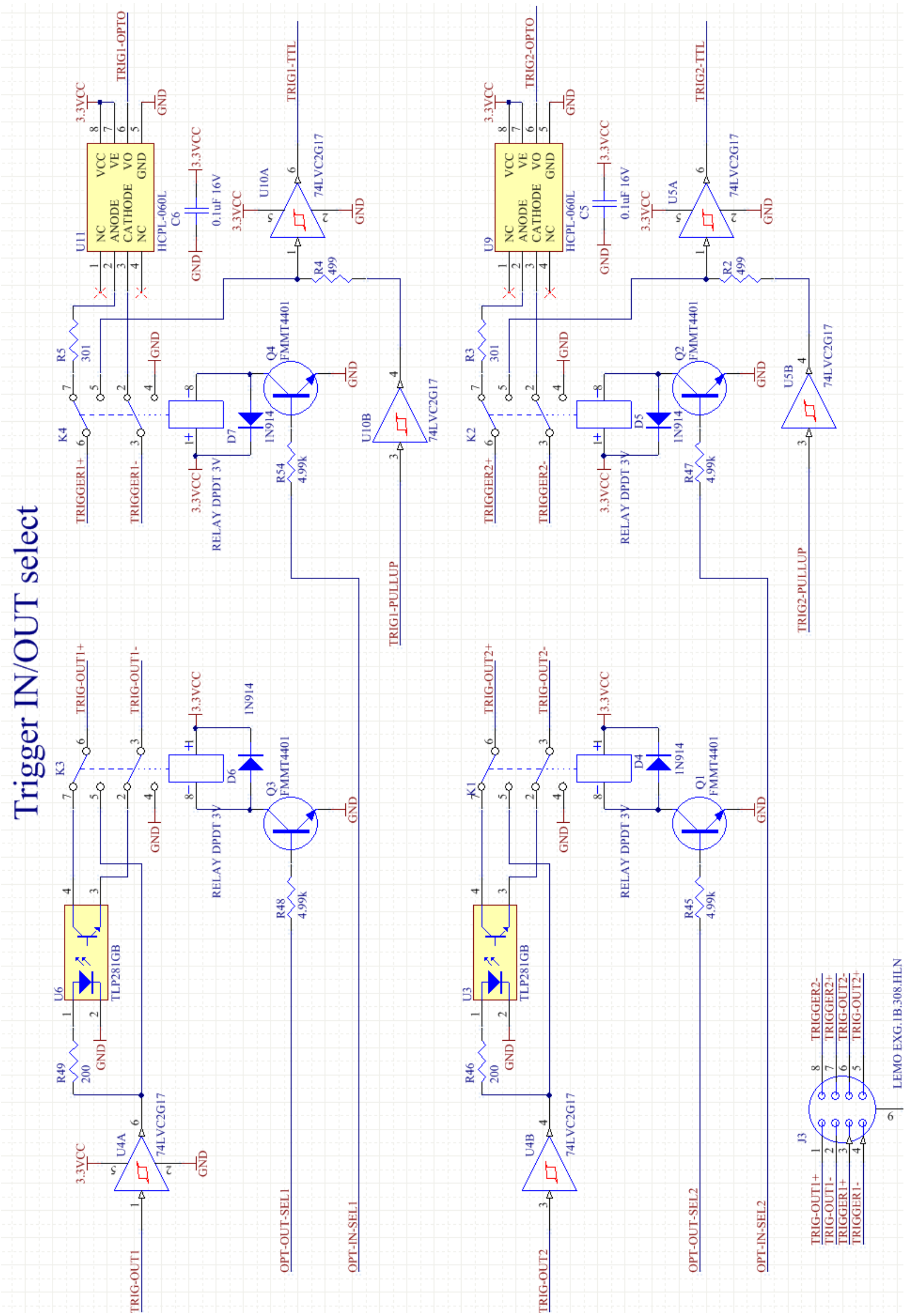
Figure 17 MX-HS Series Pinout Diagram for TRIGGER Connector

Input cabling for triggering fast data series: Trigger1 + and – is used for the input Frame Trigger, and Trigger2 + and – are used for the input Series Gate signal. Depending on the options the user selects when starting a data series or data frame, both inputs may be needed to trigger the start of the series and/or each shift/readout of the data series.

Output cabling for fast data series and/or external shutter control: The possible signal identities of Trig-Out1 and Trig-Out2 are listed on page 15. Commonly they may be assigned to Shutter (to control an external shutter) and Trigger Wait (to indicate that the detector is ready to accept the next trigger signal). (*Note: If an external shutter is used, the software's shutter close delay parameter may need to be adjusted, to account for the time it takes for the shutter to close.*)

Signal configuration: The default input and output signal type is 3.3V LVTTL (CMOS). Other signal types such as Opto (opto-isolated open collector mode) can be selected via the capxure GUI (explained on page 14) or via the remote mode interface (explained on pages 62, 64, and 67).

Circuit diagram of trigger connections:


Trigger IN/OUT select
U11
NC
ANODE
CATHODE
NC
VCC
VE
VO
GND
HCPL-060L
3.3VCC
TRIG1-OPTO
GND
C6
0.1uF 16V
3.3VCC
GND
U10A
74LVC2G17
TRIG1-TTL
R4
499
R5
301
K4
TRIGGER1+
TRIGGER1-
RELAY DPDT 3V
D7
1N914
Q4
FMMT4401
R54
4.99k
U10B
74LVC2G17
TRIG1-PULLUP
U9
HCPL-060L
3.3VCC
TRIG2-OPTO
C5
0.1uF 16V
U5A
74LVC2G17
TRIG2-TTL
R2
499
R3
301
K2
TRIGGER2+
TRIGGER2-
D5
1N914
Q2
FMMT4401
R47
4.99k
U5B
74LVC2G17
TRIG2-PULLUP
K3
TRIG-OUT1+
TRIG-OUT1-
D6
1N914
Q3
FMMT4401
R48
4.99k
U6
TLP281GB
R49
200
U4A
74LVC2G17
TRIG-OUT1
OPT-OUT-SEL1
OPT-IN-SEL1
K1
TRIG-OUT2+
TRIG-OUT2-
D4
1N914
Q1
FMMT4401
R45
4.99k
U3
TLP281GB
R46
200
U4B
74LVC2G17
TRIG-OUT2
OPT-OUT-SEL2
OPT-IN-SEL2
J3
LEMO EXG.1B.308.HLN
TRIG-OUT1+
TRIG-OUT1-
TRIGGER1+
TRIGGER1-
TRIGGER2-
TRIGGER2+
TRIG-OUT2-
TRIG-OUT2+

Chapter 5: Detector Shutdown Procedure

Note: even when not being used for a long period of time (several weeks or even more), it is fine to leave the detector operational, with cooler running. For very long, planned shutdowns (longer than at least a month), the following procedure should be used.

Detector Shutdown Procedure – Normal Method

1. Place the aluminum or plastic plate cover over the front window and attach with the screws provided.

2. Turn off (Disable) the cooler power via the capxure GUI interface. Wait about **3 hours** while the system warms up; continue to next step after the CCD temperature indicator in software shows CCD temperature readings at room temperature (> 20C).

3. Power off the detector Power Supply switch on the FRONT of the Controller Cabinet.

4. Stop the TMP by pressing the button on the front face (as shown in Figure 14) and allow it to spin down to Actual Rotation Speed ("Act Rotspd" indicator) of 0 Hz. This takes a few minutes.

5. Power off the TMP by flipping the green switch on the back of the TMP.

6. Log off and use Linux command to power down the detector control PC.

7. Power off the Controller Cabinet power switch on the BACK of the Controller Cabinet.

8. Power off the Cooler Cabinet power switch.

9. If any cooling hoses will be disconnected (for moving, shipping, storage, etc.) make sure that step 2 above has been followed – cooling connections should not be broken while the detector is cold (at operating temperature). *Note that it is strongly recommended never to make disconnections if not necessary.*

Detector Shutdown Procedure – Fast Method

This method can be used when the beamline staff do not have time to monitor the system while it warms up. It will not harm the system to use this fast shutdown method. The reasons to use the normal method (above) when possible are to make sure 1) a log of the warmup is recorded as a possible troubleshooting aid in the future, 2) the vacuum pumps out the detector head during the warmup, potentially saving time pumping the head before the next cooldown, and 3) if the system *will be disconnected immediately after warmup, the beamline staff can confirm the detector warmed up before making any disconnections.*

1. Place the aluminum plate cover over the front window and attach with the thumb screws provided.

2. Power off the detector Power Supply switch on the FRONT of the Controller Cabinet.

3. Stop the TMP by pressing the button on the front face (as shown in Figure 14) and allow it to spin down to Actual Rotation Speed ("Act Rotspd" indicator) of 0 Hz. This takes a few minutes.

4. Power off the TMP by flipping the green switch on the back of the TMP.

5. Log off and use Linux command to power down the detector control PC.

6. Power off the Controller Cabinet power switch on the BACK of the Controller Cabinet.

7. Power off the Cooler Cabinet power switch.

8. If any cooling hoses will be disconnected (for moving, shipping, storage, etc.) wait at least 3-4 hours for the detector to warm up to room temperature before making disconnections. *It is strongly recommended never to make disconnections if not necessary.*

Detector shipping procedure

- Always contact Rayonix before shipping, because most service issues can be solved remotely.
- An RMA number is required for shipping the system to Rayonix for service.
- If shipping is required, usually only the detector head needs to be shipped, **not the entire cabinet(s) or coolers** (Rayonix service personnel will advise about this).
- Always follow the shutdown procedure above, before disconnecting anything.
- Always cap any disconnected valves at both ends with valve caps.
- Always place the aluminum or plastic plate cover to protect the detector front window.
- Always ship using the special padded crate originally used to ship the detector head.

Chapter 6: Detector System Principles of Operation

X-ray CCD Detector Functionality

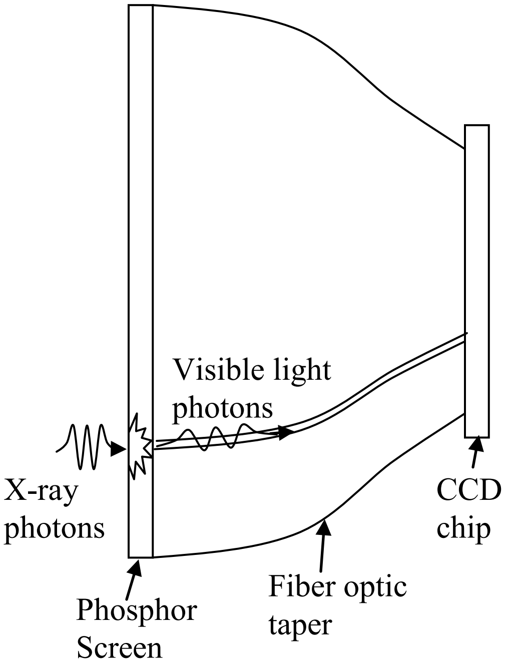


Figure 18 - Side view of CCD X-ray detector (not to scale)

Several important steps occur between the X-rays incident on the detector surface during exposure and the image that is displayed on the computer screen after readout.

- **X-ray conversion to light by phosphor screen (see Figure 18)**

  X-rays, after passing through the front window of the detector, are stopped by a phosphor screen (e.g. gadolinium oxysulfide), which converts them into visible light. This is mainly due to the fact that a CCD sensor has peak quantum efficiency in the visible region and very low quantum efficiency for detecting X-ray directly.

- **Light demagnification onto CCD (see Figure 18)**

  The maximum sizes of CCD chips are too small to be direct detectors for many applications (for example, the SX-165 61 x 61 mm^2 CCD chip is one of the largest available). Therefore, a large area of light from the phosphor screen is demagnified by a fiber optic taper onto the CCD sensor.

- **Integrating detector (vs. counting detector)**

  A CCD-based detector is an integrating detector, as opposed to a counting detector. That means that individual X-rays incident on the detector are not distinguished as they strike; rather, the detector begins in a cleared state, and then integrates the incident signal for a specified amount of time, and finally reads out the total signal accumulated in each pixel. Counting detectors count each photon (or whatever particle is being detected) as it strikes, and can have some advantages if the experiment requires detection of only one or two photons per pixel, but they have tradeoffs such as lower quantum efficiency of the X-ray interaction with the sensitive surface, and significant count rate limitation (meaning nonlinear detection efficiency vs. instaneous X-ray intensity).

- **CCD sensors**

  A CCD is a solid-state layer of silicon, an insulating layer of silicon dioxide, and a transparent array of electrodes on top (made of polysilicon), corresponding to the array of pixels. Photons pass through the polysilicon and silicon dioxide, and are absorbed in the silicon. Valence electrons are knocked free from the silicon and migrate to the electrodes. During the integration state, the number of electrons in each electrode accumulates, and is proportional to the number of incident photons.

  The two dimensions of the CCD array are referred to as the parallel direction and the serial direction. The CCD is read out at one or more corners of the array. The accumulated charge in the pixels is shifted line by line, in the parallel direction, until it reaches the serial register(s). A serial register is at the edge of the CCD and is the row of pixels at which readout occurs. The voltage of the accumulated charge in each pixel is amplified and then digitized with an A/D (analog to digital) converter. Then the serial register shifts in the serial direction and reads the value of the next pixel. Figure 19 shows a one-port CCD and a four-port CCD during readout.

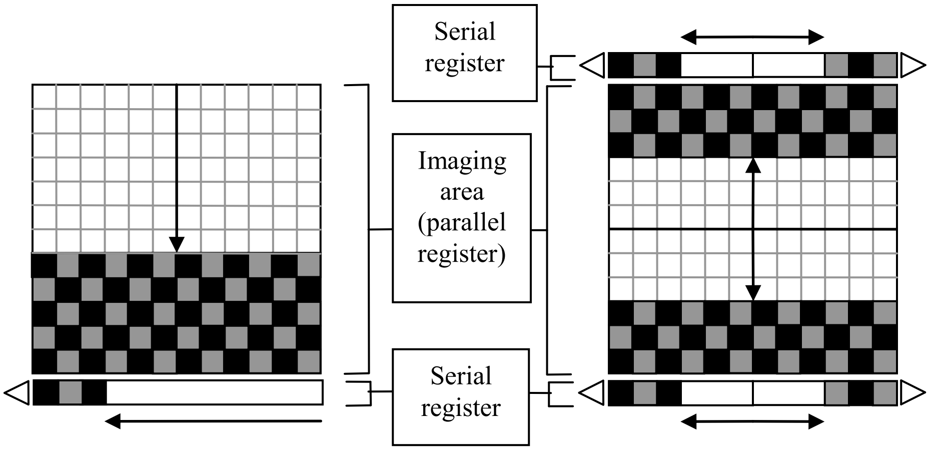


Figure 19 – One-port CCD (LEFT) and four-port CCD (RIGHT) diagrams, shown during readout. Checkered pattern indicates accumulated charge. Arrows indicate the directions in which the charge shifts.

- **Analog to digital conversion and data path**

  During readout, for each pixel an analog voltage is received from the readout channel on the CCD. On each channel this signal passes through an amplifier, is modified by an analog offset to match the A/D (Analog to Digital) converter input, and then is converted to a digital signal by an A/D converter. The location where A/D conversion takes place is different for each detector. For the SX Series, the analog signals from the four quadrants travel along four separate cables to the electronics controller (in the same cabinet in the cooler), and digital signals are sent from the electronics controller to the computer. For the MX Series detectors, the A/D converters are contained in the detector head, and the digital signals are sent from the detector head to the computer.

- **High efficiency back-illuminated CCD sensors vs. standard front-illuminated**

  Standard CCD devices, called front-illuminated CCDs, are built with the polysilicon electronics layers coated on a relatively thick slab of silicon substrate. The visible light photons detected by the device must pass through the polysilicon layers to reach the silicon layer, and absorption in those polysilicon electronics layers reduces the light-gathering quantum efficiency to about 0.35. (This is not the same as the QE of the detector for X-rays, which is determined by the phosphor screen).

  Back-illuminated CCD devices are built by using a layer of silicon which is etched thin and then coated with polysilicon electronics layers on one side. Then the silicon substrate side is illuminated, rather than the polysilicon side. The light detected by these devices strikes the silicon directly, and therefore has higher quantum efficiency, up to 0.7 or higher. The HE (high efficiency) versions of Rayonix CCD detectors use back-illuminated CCD devices. In addition to the improvement in electro-optical gain and signal vs. noise for each X-ray, the back-illuminated devices used also have very low noise readout electronics.

- **Rayonix Split Frame Transfer CCD sensor**

  The High Speed (HS) series detectors use a custom-desgned and manufactured Rayonix Split Frame Transfer CCD sensor. This device has three major advanced features that improve frame rate and duty cycle compared to a conventional sensor: 1) split frame transfer, and 2) highly parallelized 16-port per chip readout (64 ports total for 2x2 array MX HS detector, 144 ports for 3x3 array, 256 readouts for 4x4 array), and 3) 18-bit readout mode and data path.

  Split frame transfer is a method for buffering the analog CCD data before readout, allowing successive data frames to be collected with only 1msec dead time between them. The sensor has a buffer region on either side of the imaging area. Charge is shifted to the buffer region in 1msec, and the next image can be acquired while the first image is read out. Charge is clocked independently in the image acquisition area and the buffer region.

  Highly parallelized 16-port per chip readout reduces the readout time without sacrificing linearity and low noise capability of other scientific CCD sensors.

The HS detectors feature 18-bit readout mode and data path that can be used by accessing the "HDR" - mode setting in the software configuration, or the data collection menu in capxure software. It increases the dynamic range and sensitivity at the lowest signal levels. Alternatively, standard 16-bit mode is available. There is a somewhat lower maximum frame rate, and larger file size, when using the HDR mode. Make sure your data processing software can handle 18-bit input frames. HDR16 mode is also available, which does all readout and processing using 18-bit mode, but upon the final step, downsamples and saves as a 16-bit frame. There is an advantage in noise and dynamic range vs. standard 16-bit mode.

- **Transfer to computer workstation**

  The digital pixel values for the image are transferred to the computer workstation via a CamerLink high speed interface.

- **Background subtraction and image correction**

  Normally, the images displayed on the screen and saved to files are corrected images. The correction process has three major steps: background subtraction, geometric correction, and flat field correction.

  Background subtraction is necessary for two reasons: 1) the voltage level that corresponds to 0 signal may be different for each channel, and 2) the bias voltages in the electronics may drift over time or due to ambient temperature change. The software by default needs to have a background image already collected and stored in the background buffer before starting to collect a data images. One background can be used for many images, so by default the program reads the bias level of the detector twice and produces a dezingered background (otherwise, one zinger could contaminate many frames; see about zingers below). The background must be periodically recollected in case of any drift in the bias levels.

  The very cold operating temperature of Rayonix HS series CCD chips and resulting low dark current saves the user one very important time-consuming step. Background images really only need to be bias readouts (with zero integration time), as opposed to true dark current images with exposure time the same as the desired exposure time. For even a 1000 sec exposure, the dark current noise equivalent of the signal from about one 12 keV X-ray photon is accumulated (at the standard resolution).

  After background subtraction, a flat field image is applied to correct for the optical gain differences that may exist from pixel to pixel (due to permanent transparency variations in the fiber optic taper, as well as any variations in the phosphor).

  Finally, a spatial correction is applied. This is a geometric mapping created by careful calibration; it corrects for things like an overall "pincushion" distortion present in many fiber optic tapers.

- **Baseline stabilization** ***(not yet implemented)***

  The MX HS Detectors have native electronics with good baseline stability (better than ±0.5 ADU). However, the software allows the user to further stabilize the baseline of successive images, with a small tradeoff in frame rate capability. When this function is used with the maximum stability setting, the baseline stabilized to about ±0.01 ADU.

  This feature is useful for certain types of measurements that require comparisons between successive data frames that include, for example, subtracting (or adding) two data frames, such as one often must do in small angle scattering experiments. Baseline instability can make it appear that there are slightly more or slightly less X-rays across the entire detector (or readout channel) in a data frame. That is different than the read noise, which has no net effect on the average. A stable baseline is less critical for data analysis in which a background value is calculated by measuring the background around each individual spot on the same data frame (e.g. single crystal crystallography experiments).

  The method of improving the baseline is by an overscan technique. When this option is “on,” extra blank pixels are read out from the CCD after each line of the CCD is read out from the serial register. In the program memory, a temporary data frame which is larger than the normal data frame is recorded, and the pixels outside the imaging area are used to compute the baseline. The tradeoff will be a modest increase in readout time. For better accuracy, more extra blank pixels are required to establish a better baseline, causing the readout time and small calculation overhead to increase. The software allows the user to choose the desired accuracy, to optimize either the frame-to-frame dead time or the baseline stability.

- **Zingers**

  Images collected by CCD X-ray detectors can have small streaks and spots of varying intensity, which are not due to incident X-rays. These are known as zingers, and the number of them in each image is proportional to the integration time. They are random events that have two sources: radioactive decay of thorium contaminants in the glass used for fiber optic tapers, and cosmic rays. We use fiber optic tapers with the lowest thorium content available, but still some contamination will always be present. In addition, it is unavoidable that cosmic rays constantly rain down through the atmosphere and can strike the glass or phosphor screen causing a signal, and the detector is sensitive to some of these stray particles as well as the X-rays that the user is trying to measure. A big reduction in cosmic ray zingers (but not the radioactive decay zingers) could be obtained by moving the laboratory under several hundred meters of rock, like some highly sensitive physics experiments, such as neutrino detectors.

  We can take advantage of the fact that zingers are random events, however. One common solution to zingers in long exposures is to dezinger two data frames (in *capxure* or *marccd*, use the multi-read function). That is, collect two identical X-ray images and merge the images together into one; apply a statistical test to each pixel, and if the intensity in one image is much higher than the other, use the lower; if the intensities are statistically similar, average them.

Dezingering does require special care that the two images are truly identical (same X-ray dose, same movement of the sample, etc.); otherwise the statistical test will yield unpredictable results. In particular, if the X-ray beam is not constant intensity, or the sample is decaying, then the exposure times and diffractometer motions must compensate for that. If there are significant differences between the frames, then the artifacts created by dezingering may yield worse results than simply using normal, single-read images with zingers in them. Though they are not aesthetically pleasing, some kinds of data analysis can tolerate many zingers.

Cooling system

CCD sensors must be cooled in order to reduce dark current (a constant, additive buildup of thermal electrons that accumulate as the detector is integrating). For scientific imaging, CCD sensors are operated at extremely cold temperatures; MX HS Series detectors operate the CCD sensor(s) at -70 C or below. The operating temperature is factory-determined individually for each CCD and should never be changed. There is no reason to run the CCD at a warmer temperature. Attempting to operate at a much colder temperature (e.g. below -120 C) could negatively affect imaging performance or even damage some components due to stress.

The cooling method used is a closed-cycle refrigeration system, similar to what cools most refrigerators and air conditioners. Compressed gas is pumped along the SUPPLY cooling hose into the detector head. There, gas is forced through a small opening into a chamber called a “cold head” (with a thermal conductor connected to the CCD sensor support). Gas expansion causes cooling of the cold head. The expanded gas is recollected from the RETURN hose and compressed in a compressor (the white box (or boxes in the case of MX Series) inside the cabinet. The same gas continuously cycles through the system. Filter dryers that eliminate moisture from the gas are included in most systems; they are large black cylinders connected within the SUPPLY lines.

Chapter 7: Safety and System Operating Conditions

Safety Warnings

1. **PCC Compact Cooler (A.K.A. CryoTiger) cooling system**

   The cooling system uses refrigerant gas that is flammable (trade name: PT-30). If a gas leak occurs, the room should be vented immediately and flames and sparks (if any) extinguished.

   Disconnecting the gas lines should be limited or avoided if possible (it is during connection and disconnection of lines that the system is at greatest risk for gas leaks). If the lines must be disconnected, it is very important that the detector is warmed up to ambient temperature first. This may take up to three hours after shutting off the cooling. In addition, we recommend reading carefully how to connect and disconnect the lines, and have valve caps (both male and female) available to cap valves immediately if they are leaking after disconnection.

   Do not leave any valve loosened by not fully disconnected or connected; in this state, the dual valve plunger seal could be compressed, allowing a slow leak of gas to escape.

   Cap any valves of any components that are disconnected (e.g. when shipping a detector head).

2. **Electrical system**

   The detector must be protected from electrical transient events from the mains power system. Failure to isolate the detector from transients risks damage to the CCD.

3. **The detector window**

   The detector window is made of 200 µm thick Beryllium coated with Paralene-N, and is recessed behind the front flange by 5 mm. The material can easily be scratched (although surface scratches would not normally affect imaging). Harder direct hits can shatter the Beryllium (ruining the phosphor screen underneath, or even damaging the fiber optic taper).

   The window must be protected from moving diffractometer components, for example, when using a custom goniostat or very large goniometer head. These components should have hardware and software limits that prevent them from driving a device into the window.

   In most cases the Beryllium window does not require routine cleaning. However, if soiled, it may be cleaned with a mild solvent, such as low-odor mineral spirits, and a camera lens-quality disposable cloth. Use a light touch.

   When the detector is not in use, or being moved, or especially when being shipped, we recommend attaching the included aluminum cover for protection.

4. Detector electronics

Do not disconnect the power cable or signal cable between the detector head and the electronics while the electronics are powered on (similarly, do not connect this cable *after* powering on the electronics). This power cable delivers operating voltages to all of the static and clocked voltages at the CCD. If disconnection occurs, a loud alarm signal is turned on and remains on until the detector is reinitialized.

5. Opening the detector head

The detector head has no user-serviceable parts that are accessible by removing covers. Removing the vacuum flange could be disastrous if the system is at normal vacuum conditions. In addition, no vacuum clamps should ever be released if they are "inside" the valve (between the valve and the head innards). Check closely to make sure a clamp is outside the valve before releasing. **Any clamp that requires a tool to open should not be opened by the user!**

For all detectors, removing the front window may result in rendering the detector unusable until serviced and recalibrated at Rayonix. If the window becomes damaged for some reason, contact Rayonix immediately for advice about what to do next.

6. Refreshing the vacuum or maintaining vacuum (MX/LX/HS Series)

For the HS Series, a TMP unit runs constantly and maintains the vacuum during operation. Special care must be taken with the vacuum system. Sudden catastrophic release of the vacuum can damage the small wires on the CCD chip and render the detector unusable. The vacuum valve is controlled automatically by a system with built-in electronic safety checks. The controller monitors the line pressure and chamber pressure, as well as communicating with a specially instrumented TMP pump. The controller will close the valve when the line pressure rises higher than the pressure inside the chamber, or when the pump spins down or powers off (or is not connected). The safety checks should not be tested if not necessary. Powering the detector controller off will always close the valve. In addition, disconnecting the vacuum hose while the TMP is operating, or before its spin-down is completed after powering off, can cause serious damage to the TMP. Watch the rotation speed (under the "Act Rotspd" menu) go to <100 Hz before disconnecting. See the previous section about vacuum clamps on the MX Series detector head.

Operating Conditions

1. Electrical requirements

Table 4 – Total apparent power consumption, recommended circuit capacity, and cooling requirements for Rayonix X-ray equipment. ***(Notes: MX-225 (std/HE), 300, and 325 cabinets are NOT wired internally for operation at 100-120V. MX-300 and MX-325 systems require two separate receptacles, although they may share one circuit breaker. Heat output of Mar345 estimated based on 80% exposure duty cycle—8.0 min exp, 1.5 min scan, and 0.5 min erase).***

Model	Maximum KVA	Max. Amps @100V	100-120V Circuit Capacity (Amps)		Max. Amps @200V	200-240V Circuit Capacity (Amps)		Heat Output (Watts)	Heat Output (BTU/Hr)	Cooling Required (Tons)
Mar345	1060	10.6	20		5.3	10		400	2000	0.2
MarDTB	180	1.8	5		0.9	5		100	1000	0.1
SX-165	810	8.1	15		4.1	10		900	4000	0.3
MX-225 (std/HE)	1690	16.9	30		8.5	15		1700	6000	0.5
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total	2300	23.0	35	(30+10)	11.5	20	(15+5)	2300	8000	0.7
MX-300/325 (std/HE)	3380	33.8	55	(2 x 30)	16.9	30	(2 x 15)	3400	12000	1.0
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total	3990	39.9	60	(30+40)	20.0	30	(15+20)	4000	14000	1.2
SX85-HS LX170-HS LX255-HS MX170-HS						1x20				
@ 100fps						1x20				
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total										
MX225-HS (idle)	~2000 (tbd)	25 (tbd)	30		14 (tbd)	2x15		2000	7000	0.6
MX225-HS 100fps	~2500 (tbd)	25 (tbd)	30		14 (tbd)	2x15		2500	9000	0.8
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total (@100fps)	~3100 (tbd)	23.0	35	(30+10)	11.5	20	(15+5)	3500	12000	1.0
MX300-HS MX340-HS						3x15				
@ 100 fps						3x15				
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total										

The detector must be connected to properly installed incoming mains AC power, which matches the electrical setup of the system (factory set up for each country, indicated by labels next to power cord). It is important that an electrical transient surge protector be included somewhere in the incoming mains AC power to the detector.

2. Temperature

The operating ambient temperature range of the detector is 15°C to 35°C. The non-operating ambient temperature range of the detector is -10°C to 50°C. The detector must be allowed to stabilize within the operating temperature range before it is powered on.

3. Humidity

The operating humidity range for the detector is 10% RH to 50% RH. The non-operating humidity range for the detector is 5% RH to 95% RH. Note that the detector must not be operated when condensation is forming on any electrical components.

4. Altitude

The detector is rated to operate from sea level to 3000 meters (10000 feet) elevation. The non-operating altitude range is the same.

5. Vibration

The detector must not be subject to either high-impact (>3.5 g) accelerations, or to steady-state low-level mechanical vibration. Shock absorbing interfaces must be used in instances where either condition might otherwise be exceeded.

6. Aggressive vapors

The detector system must not be exposed to aggressive vapors. Specifically, salt-laden air causes micro-crystals of salt to form on all of the components inside the detector electronics unit and the detector head. These ultimately lead to low-level signal interconnects, which could damage the CCD.

Any other corrosive air may also introduce faults that could damage the CCD.

The air flowing over the fans and consequently over the components must also not contain micro-particles (dust) that can build into electrically conductive macro-particles, because of potential signal interconnects that cause damage to the CCD.

Chapter 8: Troubleshooting and Service

MX HS Series Troubleshooting guide

problem/symptom	probable cause	fix
Streaks, horizontal Example: An image of a crystal diffraction pattern shows long vertical streaks starting/ending at diffraction spots.	***Shutter problem:*** The shutter remains open after the detector has stopped integrating (i.e. exposure is finished) and starts to read out.	Lubricate with graphite or WD-40 into brass bushing around shutter shaft. Toggle shutter manually (the switch is on underside of base, close to the detector side edge of the phi-motor housing). If this does not help, call Rayonix for instructions.
Blank zone, ***horizontal stripe in center*** It looks as if the Image in each module has been cut in half horizontally and the two halves have been shifted apart by the width of the blank zone.	***Shutter problem:*** Shutter opens and closes before the detector starts integrating.	Check that shutter is not sticking open or closed. Check software control sequence. Check shutter close delay parameter.
Noise High background noise in images.	***Experimental Setup:*** Air scatter, fibers in beam, diffuse scatter... ***Temperature:*** The detector is not cooled to the proper operating temperature.	Check the beam path and crystal mount. Check the pressure (detector status window within main software window).
Pressure The pressure reported in the detector status window is higher than 100 mtorr when the detector is cold or higher than 500 mtorr when warm.	***Vacuum decayed:*** The vacuum in the detector chamber may not be good enough to allow cooling the detector to the optimal operating temperature.	Make sure that Vac valve status is Open, and TMP is operating at either 1200 or 1500 Actual Rotation Speed (Act Rotspd). Check cable connections. If Vac valve still not Open, in some cases, rebooting the detector in the capxure GUI may solve the problem.

problem/symptom	probable cause	fix
Question marks: The main capxure window has question marks for the temperature and pressure readings.	***Lost Communication:*** The software has lost communication with the detector.	Check: 1) Power to controller is on and all voltage LEDs are green, no warning LEDs on. 2) All switches are on. 3) Cable connectors are OK. 4) Fuses are not burned out. Power cycle detector and computer, restart software.
Blank image	***Experimental setup:*** No X-rays? ***Shutter problem:*** Does not open? ***Background image:*** A frame with large values for all pixels is stored as background. Automatic subtraction of this faulty background produces an all-zero image.	Turn X-rays on. Check shutter manually, make sure all cables are hooked up properly. Collect new background frame.
"Could not open file..." error message	***Permissions wrong.***	Type chmod 666 filename (return)
One or more quadrants or channels of the images are missing.	***Bad background frame.*** ***Bad electronic card***	Recollect background image Contact Rayonix
Beam stop misaligned		A beamstop should be used to prevent very large overexposures of pixels by the direct beam.
"weak beam"	**Aluminum cover on front of detector.** Rotating Anode: ***Not at full power.*** Rotating Anode or Synchrotron: ***Slits completely closed or base is misaligned.*** ***Beam conditioning optics are out of alignment.***	Remove cover. Slowly turn power up on the rotating anode. Open slits (0.4mm or as desired). Re-align base as described in the manual section called "**Error! eference source not found.** Refer to alignment procedure for your optics.

problem/symptom	probable cause	fix
The coolers are commanded on but the detector temperature does not go down.	**One or more coolers not running**	Check to make sure the pressure gauge indicators show high pressure for the SUPPLY side and low pressure for the RETURN side.
	One cooler running intermittently	Make sure Cooler Cabinet door and panels remains closed for fan to efficiently cool the cabinet! Check log file for temperature fluctuations. Cooler may also be overheating intermittently due to cooler fan not working. Order parts from Rayonix.
	Blockage in a coldhead	Check temperature readout of cold heads in software. If one cold head much warmer than others, try warming up system and perform "drying" procedure (warm up system, set cold head to +30C, turn on cooler for 10 min, repeat 2x, then cool down).
	One cooler not running / stalled	Try power cycling cooler. Cooler power switch functions as a circuit breaker. Try to flip switch to "on" position. If no response, check fuse. (UNPLUG THE COOLER BEFORE CHECKING FUSES!) There are two fuses in the power entry module in the cooler itself - where the power cord enters.

Rayonix Equipment Service

For equipment service, please contact us by any of the following methods:

Email: info@rayonix.com
Phone: +1-847-869-1548 (Toll free in the U.S. and Canada: 1-877-629-XRAY)
Fax: +1-847-869-1587

Chapter 9: Image Format (marccd)

Many crystallographic and X-ray diffraction data analysis programs are already able to process *marccd* format images. However, a complete description of the format is included here, so that any program could be written to process the images.

File Size

Images collected in native format of *marccd* are TIFF images, with a 4 kilobyte header, and two byte depth (four byte depth if 18-bit "HDR" readout mode- was used). The file size is (header size + nfast*nslow*byte_depth); therefore, a 3072 x 3072 image collected on a MX-225 will have size (4096 + 3072*3072*2) bytes = 18878464 bytes, or approximately 18 MB. The table below shows some example resolutions produced by various Rayonix X-ray detectors:

Table 5 - Example 16-bit file sizes for various resolutions of Rayonix HS detectors

Resolution	File size (exact) (bytes)	File size (MB)
384 x 384	299008	0.3
480 x 480	464896	0.5
640 x 640	823296	0.8
768 x 768	1183744	1.2
960 x 960	1847296	1.8
1280 x 1280	3280896	3.2
1920 x 1920	7376896	7.2
3840 x 3840	29495296	28.8

Timestamp

The headers of images collected with HS series detectors have a preecise timestamp with nanosecond resolution. The interpretation of the timestamp is as follows:

```
char acquire_timestamp[32]  MMDDhhmmYYYY.ss \0 XXXXXXXXX \0
```

where **M** = Months, **D** = Days, **h** = hours, **m** = minutes, **Y** = Years, **s** = seconds, **0** = Null, **X** = nanoseconds (the nine digits after the decimal). The same format applies for `header_timestamp` and `save_timestamp`.

marccd Format Image Header Description

```
Marccd Header Documentataion

from C code in frame.h  and types.h

Documentation updated by R. Doyle Feb 20 2015

Description documents marccd v0.20.20


Summary of file structure:
```

```
|-- 1024 bytes TIFF HEADER -------------|
|-- 3072 byte frame_header structure ---|
|-- nfast*nslow*depth byte image -------|

The full header, as written to the file, is a TIFF header.
The initial 1024 bytes are a minimal TIFF header with a standard
TIFF TAG pointing to the image data and a private TIFF TAG
pointing to this header structure.  As written by mmx/marccd, the
frame_header structure always begins at byte 1024 and is 3072
bytes long, making the full header 4096 bytes.

Immediately following the header is the image - it is of
arbitrary size defined by the header fields nfast, nslow and
depth. The total size is nfast * nslow * depth bytes.

The meanings of the data types should be self evident:
(example:  UINT32 is an unsigned 32 bit integer)
The exact C language definition is machine dependent but these
are the most common definitions on a 32bit architecture cpu.
#define UINT16  unsigned short
#define INT16   short
#define UINT32  unsigned int
#define INT32   int

Currently frames are always written as defined below:
      origin=UPPER_LEFT
      orientation=HFAST
      view_direction=FROM_SOURCE


/* This number is  written into the byte_order fields in the
   native byte order of the machine writing the file */
#define LITTLE_ENDIAN   1234
#define BIG_ENDIAN      4321

/* possible orientations of frame data (stored in orienation
field) */
#define HFAST                   0  /* Horizontal axis is fast */
#define VFAST                   1  /* Vertical axis is fast */

/* possible origins of frame data (stored in origin field) */
#define UPPER_LEFT              0
#define LOWER_LEFT              1
#define UPPER_RIGHT             2
#define LOWER_RIGHT             3

/* possible view directions of frame data for the given
orientation and origin (stored in view_direction field) */
#define FROM_SOURCE             0
#define TOWARD_SOURCE           1

/* possible types of data (in data_type field) */
#define DATA_UNSIGNED_INTEGER   0
#define DATA_SIGNED_INTEGER     1
#define DATA_FLOAT              2

#define MAXIMAGES 9
#define MAXSUBIMAGES 4096
```

```
     #define MAXFRAMEDIMENSION       8192

typedef struct frame_header_type {
      /* File/header format parameters (256 bytes) */
      UINT32 header_type; /* flag for header type  (can be used as
                            magic number) */
      char header_name[16];        /* header name (MARCCD) */
      UINT32 header_major_version; /* header_major_version (n.) */
      UINT32 header_minor_version; /* header_minor_version (.n) */
      UINT32 header_byte_order;    /* BIG_ENDIAN (Motorola,MIPS);
                                       LITTLE_ENDIAN (DEC, Intel) */
      UINT32 data_byte_order;  /* BIG_ENDIAN (Motorola,MIPS);
                                 LITTLE_ENDIAN (DEC, Intel) */
      UINT32 header_size;      /* in bytes */
      UINT32 frame_type;       /* flag for frame type */
      INT32  magic_number;     /* to be used as a flag - usually to
                                 indicate new file */
      UINT32 compression_type; /* type of image compression    */
      UINT32 compression1;     /* compression parameter 1 */
      UINT32 compression2;     /* compression parameter 2 */
      UINT32 compression3;     /* compression parameter 3 */
      UINT32 compression4;     /* compression parameter 4 */
      UINT32 compression5;     /* compression parameter 4 */
      UINT32 compression6;     /* compression parameter 4 */
      UINT32 nheaders;         /* total number of headers      */
      UINT32 nfast;            /* number of pixels in one line */
      UINT32 nslow;            /* number of lines in image     */
      UINT32 depth;            /* number of bytes per pixel    */
      UINT32 record_length;    /* number of pixels between successive
                                 rows */
      UINT32 signif_bits;      /* true depth of data, in bits  */
      UINT32 data_type;        /* (signed,unsigned,float...) */
      UINT32 saturated_value;  /* value marks pixel as saturated */
      UINT32 sequence;         /* TRUE or FALSE */
      UINT32 nimages;          /* total number of images - size of each
                                  is nfast*(nslow/nimages) */
      UINT32 origin;           /* corner of origin             */
      UINT32 orientation;      /* direction of fast axis       */
      UINT32 view_direction;   /* direction to view frame      */
      UINT32 overflow_location;/* FOLLOWING_HEADER, FOLLOWING_DATA */
      UINT32 over_8_bits;      /* # of pixels with counts > 255 */
      UINT32 over_16_bits;     /* # of pixels with count > 65535 */
      UINT32 multiplexed;      /* multiplex flag */
      UINT32 nfastimages;      /* # of images in fast direction */
      UINT32 nslowimages;      /* # of images in slow direction */
      UINT32 darkcurrent_applied; /* flags correction has been applied
                                 - hold magic number ? */
      UINT32 bias_applied;       /* flags correction has been applied –
                                 hold magic number ? */
      UINT32 flatfield_applied;  /* flags correction has been applied –
                                 hold magic number ? */
      UINT32 distortion_applied; /* flags correction has been applied –
                                 hold magic number ? */
      UINT32 original_header_type;     /* Header/frame type from file
                                 that frame is read from */
      UINT32 file_saved;         /* Flag that file has been saved,
                                 should be zeroed if modified */
      UINT32 n_valid_pixels;     /* Number of pixels holding valid data
```

```
                           - first N pixels */
UINT32 defectmap_applied; /* flags correction has been applied –
                           hold magic number ? */
UINT32 subimage_nfast;       /* when divided into subimages (eg.
                           frameshifted) */
UINT32 subimage_nslow;       /* when divided into subimages (eg.
                           frameshifted) */
UINT32 subimage_origin_fast; /* when divided into subimages (eg.
                           frameshifted) */
UINT32 subimage_origin_slow; /* when divided into subimages (eg.
                           frameshifted) */
UINT32 readout_pattern;      /* BIT Code - 1 = A, 2 = B, 4 = C,
                           8 = D */
UINT32 saturation_level;     /* at this value and above, data are
                           not reliable */
UINT32 orientation_code;     /* Describes how this frame needs to
                           be rotated to make it "right" */
UINT32 frameshift_multiplexed;/* frameshift multiplex flag */
UINT32 prescan_nfast;    /* Number of non-image pixels preceeding
                           imaging pixels - fast direction - also
                           used with CTBL for ROI.A.x */
UINT32 prescan_nslow;    /* Number of non-image pixels preceeding
                           imaging pixels - slow direction - also
                           used with CTBL for ROI.A.y */
UINT32 postscan_nfast;   /* Number of non-image pixels followng
                           imaging pixels - fast direction - also
                           used with CTBL for ROI.B.x */
UINT32 postscan_nslow;   /* Number of non-image pixels followng
                           imaging pixels - slow direction - also
                           used with CTBL for ROI.B.y */
UINT32 prepost_trimmed;  /* trimmed==1 means pre and post scan
                           pixels have been removed */
UINT32 orig_frameformat_serial_length; /* Used for CTBL with
                           region-of-interest to store full CTBL
                           serial length */
UINT32 orig_frameformat_parallel_length; /* Used for CTBL with
                           region-of-interest to store full CTBL
                           parallel length */
char reserve1[12];

/* Data statistics (128) */
UINT32 total_counts[2];  /* 64 bit integer range = 1.85E19*/
UINT32 special_counts1[2];
UINT32 special_counts2[2];
UINT32 min;
UINT32 max;
UINT32 mean;             /* mean * 1000 */
UINT32 rms;              /* rms * 1000 */
UINT32 n_zeros;          /* number of pixels with 0 value  - not
                           included in stats in unsigned data */
UINT32 n_saturated;      /* number of pixels with saturated value
                           - not included in stats */
UINT32 stats_uptodate;   /* Flag that stats OK - ie data not
                           changed since last calculation */
UINT32 pixel_noise[MAXIMAGES]; /* 1000*base noise value (ADUs) */
char reserve2[(32-13-MAXIMAGES)*sizeof(INT32)];

/* Sample Changer info */
```

```
char          barcode[16];
UINT32        barcode_angle;
UINT32        barcode_status;
/* Pad to 256 bytes */
char reserve2a[(64-6)*sizeof(INT32)];

/* Goniostat parameters (128 bytes) */
INT32 xtal_to_detector; /* 1000*distance in millimeters */
INT32 beam_x;           /* 1000*x beam position (pixels) */
INT32 beam_y;           /* 1000*y beam position (pixels) */
INT32 integration_time; /* integration time in milliseconds */
INT32 exposure_time;    /* exposure time in milliseconds */
INT32 readout_time;     /* readout time in milliseconds */
INT32 nreads;           /* number of readouts to get this img */
INT32 start_twotheta;   /* 1000*two_theta angle */
INT32 start_omega;      /* 1000*omega angle */
INT32 start_chi;        /* 1000*chi angle */
INT32 start_kappa;      /* 1000*kappa angle */
INT32 start_phi;        /* 1000*phi angle */
INT32 start_delta;      /* 1000*delta angle */
INT32 start_gamma;      /* 1000*gamma angle */
INT32 start_xtal_to_detector; /* 1000*distance in mm (dist in
                                 um)*/
INT32 end_twotheta;     /* 1000*two_theta angle */
INT32 end_omega;        /* 1000*omega angle */
INT32 end_chi;          /* 1000*chi angle */
INT32 end_kappa;        /* 1000*kappa angle */
INT32 end_phi;          /* 1000*phi angle */
INT32 end_delta;        /* 1000*delta angle */
INT32 end_gamma;        /* 1000*gamma angle */
INT32 end_xtal_to_detector; /* 1000*distance in mm (dist in um)*/
INT32 rotation_axis;    /* active rotation axis (index into above
                           ie. 0=twotheta,1=omega...) */
INT32 rotation_range;   /* 1000*rotation angle */
INT32 detector_rotx;    /* 1000*rotation of detector  X */
INT32 detector_roty;    /* 1000*rotation of detector around Y */
INT32 detector_rotz;    /* 1000*rotation of detector around Z */
INT32 total_dose;       /* Hz-sec (counts) integrated over full
                           exposure */
char reserve3[(32-29)*sizeof(INT32)]; /* Pad Gonisotat parameters
                           to 128 bytes */

/* Detector parameters (128 bytes) */
INT32 detector_type;    /* detector type */
INT32 pixelsize_x;      /* pixel size (nanometers) */
INT32 pixelsize_y;      /* pixel size (nanometers) */
INT32 mean_bias;        /* 1000*mean bias value */
INT32 photons_per_100adu;       /* photons / 100 ADUs */
INT32 measured_bias[MAXIMAGES]; /* 1000*mean bias value for each
                           image*/
INT32 measured_temperature[MAXIMAGES];  /* Temperature of each
                           detector in milliKelvins */
INT32 measured_pressure[MAXIMAGES];     /* Pressure of each
                           chamber in microTorr */
/* Retired reserve4 when MAXIMAGES set to 9 from 16 and two
fields removed, and temp and pressure added
char reserve4[(32-(5+3*MAXIMAGES))*sizeof(INT32)];
*/
```

```
/* X-ray source and optics parameters (128 bytes) */
/* X-ray source parameters (14*4 bytes) */
INT32 source_type;      /* (code) - target, synch. etc */
INT32 source_dx;        /* Optics param. - (size microns) */
INT32 source_dy;        /* Optics param. - (size microns) */
INT32 source_wavelength;/* wavelength (femtoMeters) */
INT32 source_power;     /* (Watts) */
INT32 source_voltage;   /* (Volts) */
INT32 source_current;        /* (microAmps) */
INT32 source_bias;           /* (Volts) */
INT32 source_polarization_x; /* () */
INT32 source_polarization_y; /* () */
INT32 source_intensity_0;    /* (arbitrary units) */
INT32 source_intensity_1;    /* (arbitrary units) */
char reserve_source[2*sizeof(INT32)];

/* X-ray optics_parameters (8*4 bytes) */
INT32 optics_type;           /* Optics type (code)*/
INT32 optics_dx;             /* Optics param. - (size microns) */
INT32 optics_dy;             /* Optics param. - (size microns) */
INT32 optics_wavelength;     /* Optics param. - (size microns) */
INT32 optics_dispersion;     /* Optics param. - (*10E6) */
INT32 optics_crossfire_x;    /* Optics param. - (microRadians) */
INT32 optics_crossfire_y;    /* Optics param. - (microRadians) */
INT32 optics_angle;          /* Optics param. - (monoch. 2theta -
                                microradians) */
INT32 optics_polarization_x; /* () */
INT32 optics_polarization_y; /* () */
char reserve_optics[4*sizeof(INT32)];
char reserve5[((32-28)*sizeof(INT32))]; /* Pad X-ray parameters
                                to 128 bytes */

/* File parameters (1024 bytes) */
char filetitle[128];         /* Title */
char filepath[128];          /* path name for data file */
char filename[64];           /* name of data file */
char acquire_timestamp[32];  /* date and time of acquisition */
char header_timestamp[32];   /* date and time of header update */
char save_timestamp[32];     /* date and time file saved */
char file_comment[512];      /* comments  - can be used as
                                desired   */
char reserve6[1024-(128+128+64+(3*32)+512)]; /* Pad File  to 1024
                                bytes */

/**************************************************/
//Optional interpretation of above times for higher resolution
//char acquire_timestamp[16]; /* date and time of
                           acquisition %m%d%H%M%Y.%S */
//char acquire_timestamp_nanoseconds[16];/* nanosecond portion -
                           zero-filled if present, else NULLs */
//char header_timestamp[16]; /* date and time of header update */
//char header_timestamp_nanoseconds[16];/* nanosecond portion -
                           zero-filled if present, else NULLs */
//char save_timestamp[16];    /* date and time of file saved  */
//char save_timestamp_nanoseconds[16]; /* nanosecond portion -
                           zero-filled if present, else NULLs */
/**************************************************/
```

```
/* Dataset parameters (512 bytes) */
char dataset_comment[512];    /* comments  - can be used as
                           desired   */
/* Reserved for user definable data - will not be used by Mar! */
char user_data[512];
/* char pad[----] USED UP! */     /* pad out to 3072 bytes */

} frame_header;
```

Chapter 10: Legacy Remote Mode for HS Detector Control

This manual explains how to set up a client compatible with the remote mode (using program ***hsserver_legacy*** or ***capxure***).

Introduction

The legacy remote mode server is a method of controlling the Rayonix HS (High Speed) series of detectors by emulating the old *marccd* style of remote mode. In this way the detector can be controlled by the user's control software. Data acquisition controls, such as changing binning, collecting data images and data series, setting header information, and saving files are available through this interface. An institution might prefer to use this mode if they have previously controlled Rayonix detectors via *marccd* remote mode and want to quickly get going with minimal changes to their control software. The location of the server program is usually ***/opt/rayonix/bin/hsserver_legacy***.

Legacy remote mode communication diagram

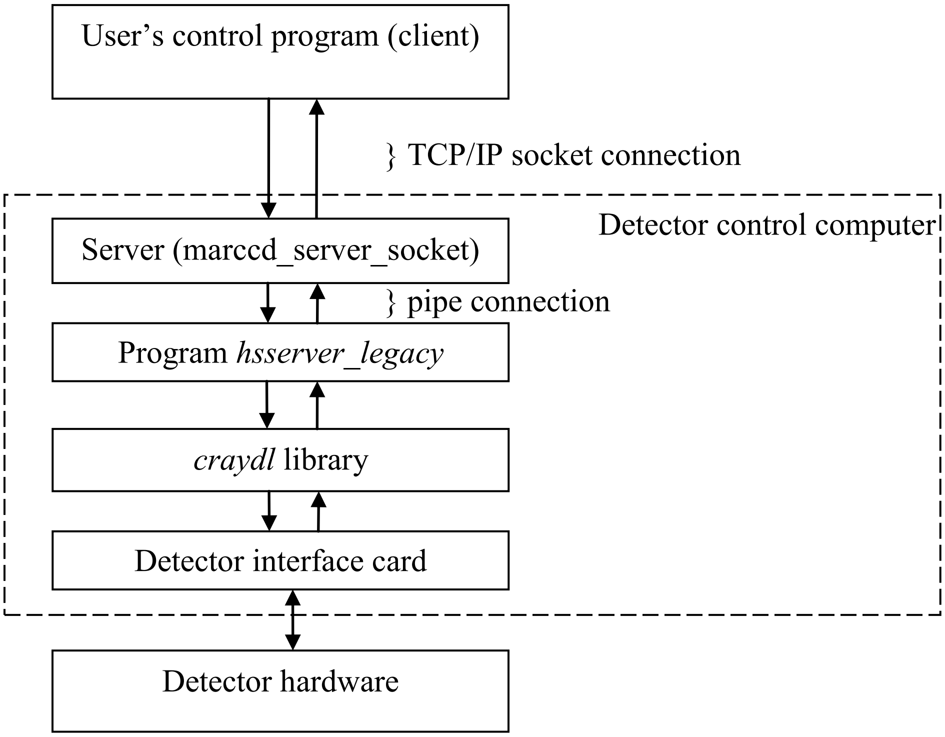

User's control program (client)
} TCP/IP socket connection
Detector control computer
Server (marccd_server_socket)
} pipe connection
Program hsserver_legacy
craydl library
Detector interface card
Detector hardware

Figure 20 – Legacy remote mode communication path

Configuring the legacy remote mode

The file /opt/rayonix/configuration/RemoteModeEmulator.conf contains the following configurable variables (with suggested defaults in the file).

- **ServerEnvironment:** if an environment variable is required by the control program, it can be inserted here (usually not required)
- **ServerCommand:** usually marccd_server_socket.
- **ServerArguments:** the port number to be opened by the server. It should match that looked for by the client. The sample client program provided uses port number 2222.
- **ServerLog:** not yet implemented at this time.

Typically no change would be required for these parameters.

The client program

The client program can be written in any language It functions like a telnet session to the socket program, into which text commands (described in the next section) may be entered to drive *hsserver_legacy*. Users will need to incorporate this or a similar client into the controlling program they wish to use.

Type ./marccd_client_socket to start the program. The user may try typing in the commands below (such as get_state, or get_bin, etc.) in order to verify that indeed the *hsserver_legacy* program is executing these commands.

Alternatively, for testing purposes a telnet session may be used to connect to the server program and enter commands one at a time by hand. Typically the following command would be issued: "`telnet LOCALHOST [port number]`" where port number would be 2222 (or whatever port number according to the ServerArguments configuration parameter).

Remote commands

The program *hsserver_legacy* understands the following remote mode commands:

Remote Mode Command	**Effect**
`[command],help`	The program will return usage information about `command` *(not available for all commands)*.
`abort`	The program will abort the current operation. Returns CCD to continuous clear mode and server program to state 0x0.
`get_bin`	Returns the fast (x) and slow (y) binning of the data frame.
`set_bin,x,y`	Sets the fast (x) and slow (y) binning of the data frame.
`correct`	The program will apply geometric and flatfield corrections to the raw data frame.

<table>
<tr><td><code>dezinger,flag</code></td><td>The program will calculated a "dezingered" frame from two stored frames. One of the source frames is the System Scratch frame. The second source frame and the destination are specified with the flag.
<table>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>0</td><td>use and store into the latest data frame.</td></tr>
<tr><td>1</td><td>use and store into the current background frame</td></tr>
<tr><td>2</td><td>use and store into system scratch storage (not useful; frame dezingered with itself)</td></tr>
</table></td></tr>
<tr><td><code>get_electronic_shutter</code></td><td>Returns the electronic shutter function state. 0 = disabled, 1 = enabled.</td></tr>
<tr><td><code>set_electronic_shutter ,flag</code></td><td>Sets the electronic shutter function. Flag 0 = disabled, 1 = enabled. Electronic shutter normally should be on. Causes frame transfer at both beginning and end of integration, when in any non-frame transfer mode (timed exposures or bulb mode). Necessary for shutterless data collection.</td></tr>
<tr><td><code>end_automation</code></td><td>The program will exit remote mode.</td></tr>
<tr><td><code>get_fastbulb</code></td><td>Returns the Fastbulb function state. 0 = disabled, 1 = enabled.</td></tr>
<tr><td><code>set_fastbulb,flag</code></td><td>Sets the Fastbulb function. 0 = disabled, 1 = enabled. Normally should be Enabled. This pertains to Bulb mode. When enabled, keeps output shutter signal high during entire series. When disabled, shutter opens and closes for each frame.</td></tr>
<tr><td><code>get_frameshift</code></td><td><b>***NOT IMPLEMENTED***</b></td></tr>
<tr><td><code>set_frameshift,flag</code></td><td><b>***NOT IMPLEMENTED***</b> This function is of little use because HS detector is frame-transfer type CCD sensor.</td></tr>
<tr><td><code>gate[,flag]</code></td><td>Software gate emulator that mimics the detector receiving an input gate signal. Gate signal type must be set to <b>Software</b>.
<table>
<tr><th>gate flag</th><th>Action</th></tr>
<tr><td>none</td><td>Returns the state of the software gate function (0 = low, or 1 = high)</td></tr>
<tr><td>0</td><td>Lowers or keeps low the software gate (equivalent to falling edge signal input).</td></tr>
<tr><td>1</td><td>Raises or keeps high the software gate (equivalent to rising edge signal input).</td></tr>
<tr><td>time</td><td>Bulb gate with time specified as a floating point number with seconds unit.</td></tr>
</table></td></tr>
<tr><td><code>get_gate_signal_type</code></td><td>Returns the gate signal type.</td></tr>
<tr><td><code>set_gate_signal_type ,flag</code></td><td>Sets the gate signal type. Flag can be the following: <b>Opto, Opto Inverted, CMOS Pulldown, CMOS Pullup, CMOS Pulldown Inverted, CMOS Pullup Inverted, Software</b>. Software gate uses the command “gate” shown above.</td></tr>
<tr><td><code>get_gate_type</code></td><td>Returns the gate type.</td></tr>
</table>

<table>
<tr><td><code>set_gate_type,flag</code></td><td>Sets the gate type. Flag = 0, 1, or 2. Flag 0 = “None” – no series gate, begin series immediately. Flag 1 = “Start” – series begins upon rising edge of input Series Gate signal. Flag 2 = “Gate” – series begins with rising edge of Series Gate input signal, and can be paused and resumed with falling edge and rising edge.
<i>Note that selecting “Start” or “Gate” will <b>require</b> the use of a Series Gate signal, and without it, no frames will be collected.</i></td></tr>
<tr><td><code>header,header_data\n</code></td><td>The program will accept <code>header_data</code> and interpret item=value pairs to be placed into the data frame header. <code>header_data</code> consists of a list of item=value pairs separated by commas and terminated by a newline (<code>\n</code>). The following items are understood:
<table>
<tr><th>Parameter</th><th>Type (Units)</th></tr>
<tr><td>beam_x</td><td>float (mm)</td></tr>
<tr><td>beam_y</td><td>float (mm)</td></tr>
<tr><td>exposure_time</td><td>float (sec)</td></tr>
<tr><td><i>The following are used for giving starting values for the respective items:</i>

theta / twotheta,
omega,
chi,
kappa,
phi,
gamma,
delta,
distance,
wavelength,
energy</td><td>

float (deg),
float (deg),
float (deg),
float (deg),
float (deg),
float (deg),
float (mm),
float (mm),
float (angstroms),
float (eV)</td></tr>
<tr><td>rotation_axis</td><td>string (twotheta, omega, chi, kappa, phi, gamma, delta or distance)</td></tr>
<tr><td>rotation_range</td><td>float (deg)</td></tr>
<tr><td>file_comments</td><td>string</td></tr>
<tr><td>dataset_comments</td><td>string</td></tr>
<tr><td>detector_distance</td><td>DEPRECATED; use distance</td></tr>
<tr><td>xtal_to_detector</td><td>DEPRECATED; use distance</td></tr>
<tr><td>start_phi</td><td>DEPRECATED; use phi</td></tr>
<tr><td>source_wavelength</td><td>DEPRECATED; use wavelength</td></tr>
</table></td></tr>
<tr><td><code>help</code></td><td>Returns a list of all commands understood. Useful for confirming that the version of the legacy remote server understands the command that you are trying to issue. Most often the help command is typed manually when connected via a telnet session.</td></tr>
<tr><td><code>is_state</code></td><td><b>***DEPRECATED***</b> Use <code>get_state</code> instead.</td></tr>
</table>

<table>
<tr><td>get_mode</td><td>Returns the remote mode version. The legacy remote mode is version 2, but backwards compatible with version 1. If the client software requires the answer to be “1,” the software can be configured to return version 1 (using the variable ReportRemoteModeVersion in file RemoteModeEmulator.conf).</td></tr>
<tr><td>get_mpp</td><td>Returns the MPP mode state. 0 = disabled, 1 = enabled.</td></tr>
<tr><td>set_mpp,flag</td><td>Sets theMPP mode. Flag 0 = disabled, 1 = enabled. MPP mode should be used for long (greater than 10 sec) bulb mode or triggered exposures to reduce the dark current. When Enabled, there is a 1sec delay between trigger and frame transfer (either for frame trigger or bulb trigger), because switching to MPP mode takes time. Unnecessary for short (less than 10sec) bulb mode or triggered exposures. Automatically controlled (no matter the state of this variable) when the detector controls the frame timing. When timed data frames are greater then 10sec, MPP mode is used.</td></tr>
<tr><td>get_output1_signal_id</td><td>Returns the Output 1 signal ID.</td></tr>
<tr><td>set_output1_signal_id ,flag</td><td>Sets the Output 1 signal ID. Flag options are: <b>Shutter, Integrate, Frame, Line, Shutter Opening, Shutter Closing, Shutter Active, Trigger Wait, Trigger ACK, Trigger Fall Wait, Trigger Fall ACK, Gate Armed, Gate</b>.</td></tr>
<tr><td>get_output1_signal_type</td><td>Returns the Output 1 signal type.</td></tr>
<tr><td>set_output1_signal_type ,flag</td><td>Sets the Output 1 signal type. Flag options are: <b>Opto, CMOS</b>.</td></tr>
<tr><td>get_output2_signal_id</td><td>Returns the Output 2 signal ID.</td></tr>
<tr><td>set_output2_signal_id ,flag</td><td>Sets the Output 2 signal ID. Flag options are: <b>Shutter, Integrate, Frame, Line, Shutter Opening, Shutter Closing, Shutter Active, Trigger Wait, Trigger ACK, Trigger Fall Wait, Trigger Fall ACK, Gate Armed, Gate</b>.</td></tr>
<tr><td>get_output2_signal_type</td><td>Returns the Output 2 signal type.</td></tr>
<tr><td>set_output2_signal_type ,flag</td><td>Sets the Output 2 signal type. Flag options are: <b>Opto, CMOS</b>.</td></tr>
<tr><td>get_preset</td><td>***NOT IMPLEMENTED***</td></tr>
<tr><td>set_preset</td><td>***NOT IMPLEMENTED***</td></tr>
<tr><td>get_press</td><td>Returns the current pressure inside the detector head</td></tr>
<tr><td>readout,flag[,filename]</td><td>Stops integrating (or clearing, if issued directly from clearing mode), and starts reading the CCD. Given filename(s), it will queue the correction and writing of the file to disk
<table>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>0</td><td>read data into raw data frame storage</td></tr>
<tr><td>1</td><td>read data into background frame storage <b>(Note that a dezingered multiple readout is automatically collected by a single readout,1 command)</b></td></tr>
<tr><td>2</td><td>read data into system scratch storage</td></tr>
<tr><td>3</td><td>read data into data frame storage and do NOT correct [and write uncorrected frame]</td></tr>
</table></td></tr>
<tr><td>get_readout_mode</td><td>Returns the current readout mode setting, with flags defined above in set_readout_mode section.</td></tr>
</table>

<table>
<tr><td><code>set_readout_mode,flag</code></td><td>Program will set the readout mode to one of the following values:
<table>
<tr><th>Flag</th><th>Readout Mode</th><th>Gain</th><th>Speed</th><th>Read bits</th><th>File bits</th></tr>
<tr><td>0</td><td>Standard</td><td>norm</td><td>norm</td><td>16</td><td>16</td></tr>
<tr><td>1</td><td>High Gain</td><td>high</td><td>norm</td><td>16</td><td>16</td></tr>
<tr><td>2</td><td>Low Noise</td><td>high</td><td>med</td><td>16</td><td>16</td></tr>
<tr><td>3</td><td>HDR</td><td>high</td><td>low</td><td>18</td><td>32</td></tr>
<tr><td>4</td><td>Turbo</td><td>norm</td><td>highest</td><td>16</td><td>16</td></tr>
<tr><td>5</td><td>HDR16</td><td>high</td><td>low</td><td>18</td><td>16</td></tr>
</table>
Note that some data processing programs may need to be updated to read <b>HDR</b> (High Dynamic Range) mode files, which are 32 bit (compared to 16 bit standard images). -</td></tr>
<tr><td><code>reboot</code></td><td>Reboots the detector controller. (Caution – this is not needed in normal operation.)</td></tr>
<tr><td><code>get_roi</code></td><td><b>***TO BE IMPLEMENTED***</b> Returns the region of interest.</td></tr>
<tr><td><code>set_roi,x0,y0,x1,y1</code></td><td><b>***TO BE IMPLEMENTED***</b> Sets the region of interest. Setting (x0,y0,x1,y1) = (0,0,0,0) turns off the region of interest.</td></tr>
<tr><td><code>shutter,flag</code></td><td>The program will set the shutter state to either closed or open. Only functions if the either Output 1 or Output 2 trigger ID set to “Shutter,” and the appropriate cable connects from the “Trigger” connector on the detector head to the user shutter.
<table>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>none</td><td>Returns the current shutter state (0 or 1)</td></tr>
<tr><td>0</td><td><b>manual/closed</b>
Shutter signaled to close. Signal falls or remains low on configured Output.</td></tr>
<tr><td>1</td><td><b>manual/open</b>
Shutter signaled to open. Signal rises or remains up on configured Output.</td></tr>
</table></td></tr>
<tr><td><code>get_size</code></td><td>Returns the fast (x) and slow (y) dimensions of the data frame.</td></tr>
<tr><td><code>get_size_bkg</code></td><td>Returns the fast (x) and slow (y) dimensions of the stored background frame (0,0 if no background frame is yet present).</td></tr>
<tr><td><code>get_stability</code></td><td><b>***TO BE IMPLEMENTED***</b> Returns the baseline stability target.</td></tr>
<tr><td><code>set_stability,target</code></td><td><b>***TO BE IMPLEMENTED***</b> Sets the baseline stability target as a floating point number. Causes extra blank pixels to be read in addition to each frame to accurately maintain baseline stability. Target = 0.0 turns off the baseline stabilization.</td></tr>
<tr><td><code>start</code></td><td>The program will start integrating data (stop clearing) on the CCD.</td></tr>
</table>

`start_series`
`[,n_frames]`
`[,first_frame_number]`
`[,integration_time]`
`[,interval_time]`
`[,frame_trigger_type]`
`[,series_trigger_type]`
`[,filename_base]`
`[,filename_suffix]`
`[,number_field_width]`
	Starts a data collection series with the specified parameters. A background image is required to have already been acquired.
	N_frames is the number of frames in the sequence (default 1). **First_frame_number** is the number of the first frame for the filename (default 1). **Integration_time** is the time duration (in seconds) of each exposure. **Interval_time** is the time between the starting of each exposure, in seconds (must be equal to or greater than integration_time plus one millisec, otherwise there will be fewer than one frame per interval, since an interval timeout is ignored during an exposure).**Frame_trigger_type** is defined in table below. 0 = not triggered (asynchronous), 1 = triggered frame transfer, 2 = bulb mode, and 3 = "LCLS mode." **Series_trigger_type** allows gating the data collection (can be used with any frame trigger modes). 0 = not gated, 1 = starts on rising edge (no stop), 2 = retriggerable gate, starts on rising edge, stops on falling edge. Filenames are defined as [**filename base**][**number field**][**filename suffix**] where **number_field_width** is an integer that defines the number of digits of the field. **Filename_base** contains the path (e.g. "/mnt/ssd/test_image").

Exposure time (sec)	**Interval time (sec)**	**Frame_trigger_type**	**Behavior**
0	0	0	Frame transfer at maximum possible rate
0	> 0	0	Timed exposures of specified interval, separated by 1ms frame transfer
> 0	0	0	Timed exposures separated by full readout time
> 0	> Exp time	0	Timed exposures on specified intervals
0	N/A	1	Frame transfer triggered by rising edge of external source
> 0	N/A	1	Timed exposures triggered by rising edge of external source
N/A	N/A	2	"Bulb mode." Timed exposures started by rising edge of and ended (readout) by falling edge of external source
0	0	3	****Not yet implemented**** **"LCLS mode."** Frame transfer/clear data (no readout) on input Trig1. Readout on input Trig2. Useful for high rep rate XFEL experiment that only has X-ray data in some XFEL pulses.

<table>
<tr><td><code>get_state</code></td><td>Returns the current state of the system.
The state is a 32 bit field which which includes the status of the server program in the lower 4 bits, and other task status information in the other bits (see section below, starting on page 68, for a full discussion of state handling).</td></tr>
<tr><td><code>set_state,state</code></td><td><b>***NOT IMPLEMENTED***</b> Sets the state to the desired state. This is for testing purposes only.</td></tr>
<tr><td><code>get_temp</code></td><td>Returns the current CCD temperature, or the highest (warmest) CCD temperature in a detector with multiple CCDs</td></tr>
<tr><td><code>trigger[,flag]</code></td><td>Software trigger emulator that mimics the detector receiving an input trigger. Trigger signal type must be set to <b>Software</b>.
<table>
<tr><th>trigger flag</th><th>Action</th></tr>
<tr><td>none</td><td>Returns the software trigger state (0=disabled, or 1=enabled)</td></tr>
<tr><td>0</td><td>Disables the software trigger (equivalent to falling edge signal input).</td></tr>
<tr><td>1</td><td>Enables the software trigger (equivalent to rising edge signal input).</td></tr>
<tr><td>time</td><td>Bulb trigger with time specified as a floating point number, with seconds unit.</td></tr>
</table></td></tr>
<tr><td><code>get_trigger_configuration</code></td><td>***DEPRECATED***</td></tr>
<tr><td><code>set_trigger_configuration</code></td><td>***DEPRECATED***</td></tr>
<tr><td><code>get_trigger_signal_type</code></td><td>Returns the trigger signal type.</td></tr>
<tr><td><code>set_trigger_signal_type,flag</code></td><td>Sets the trigger signal type. Flag can be the following: <b>Opto, Opto Inverted, CMOS Pulldown, CMOS Pullup, CMOS Pulldown Inverted, CMOS Pullup Inverted, Software</b>. Software trigger uses the command “trigger” shown above.</td></tr>
<tr><td><code>get_trigger_type</code></td><td>Returns the trigger type.</td></tr>
<tr><td><code>set_trigger_type,flag</code></td><td>Sets the trigger type. Flag = 0, 1, or 2. Flag 0 = “None” – no frame trigger, to use detector internal timer. Flag 1 = “Frame” – rising edge of input frame trigger causes electronic shutter followed by readout and simultaneous image acquisition. <i>See timing diagram on page 74</i>. Flag 2 = “Bulb” – images are timed by electronic shutter / image acquisition when rising edge input trigger received, then ending when falling edge received. Like an old-fashioned camera bulb mode.
<i>Note that selecting “Frame” (flag 1) or “Bulb” (flag 2) will <b>require</b> the use of a Frame Trigger signal, and without it, no frames will be collected.</i></td></tr>
<tr><td><code>writefile,filename,flag</code></td><td>Writes data frame from selected memory buffer to a file on disk. Parameter <code>filename</code> is the name of the file to be written.
<table>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>0</td><td>write raw file</td></tr>
<tr><td>1</td><td>write corrected file</td></tr>
<tr><td>2</td><td>write background file</td></tr>
<tr><td>3</td><td>write scratch file</td></tr>
</table></td></tr>
</table>

State and status values in remote mode version 1 / legacy remote mode

The status returned by get_state is represented by eight 4-bit fields, for a total of a 32-bit value. It looks like Figure 21:

unused	series	dezinger	write	correct	read	acquire	state

Figure 21 – Status fields in remote mode version 1

The server **"state"** is represented in the lowest 4 bits of the status, and is defined differently than other statuses. It will only ever have the following number integer values (representing states IDLE, ERROR and BUSY):

State Number	State
0	IDLE
6	UNAVAILABLE
7	ERROR
8	BUSY

The other seven 4-bit fields represent the following tasks:

Task Number	Task
0	TASK_ACQUIRE
1	TASK_READ
2	TASK_CORRECT
3	TASK_WRITE
4	TASK_DEZINGER
5	TASK_SERIES
6	unused

The status bits for each task are:

Task Status Bit	Task Status
0x1	TASK_STATUS_QUEUED
0x2	TASK_STATUS_EXECUTING
0x4	TASK_STATUS_ERROR
0x8	TASK_STATUS_RESERVED

Examples state values returned by get_state:

Idle	0x00000000
Busy (interpreting command)	0x00000008
Error (command not understood)	0x00000007
Acquiring	0x00000010
Reading	0x00000200
Reading w/correct and write queued	0x00011200
Correcting w/write queued:	0x00012000
Error writing file	0x00040000
Collecting a series	0x02000000

These are the C definitions of masks for looking at task state bits:

```
      #define STATUS_MASK             0xf

      #define TASK_STATUS_MASK(task)  (STATUS_MASK <<
      (4*((task)+1)))
```

These are some convenient macros for checking and setting the state of each task. They are used in the *hsserver_legacy* code and can be used in the client code:

```
      #define TASK_STATUS(current_status, task) (((current_status)
      & TASK_STATUS_MASK(task)) >> (4*((task) + 1)))

      #define TEST_TASK_STATUS(current_status, task, status)
      (TASK_STATUS(current_status, task) & (status))
```

The following is an example of pseudo C code to do an exposure sequence:

```
/* Get a backround frame */
   /* Wait for detector to NOT be reading */
   do {
     /* send: get_state */
     /* put result in state */
   }  while (TEST_TASK_STATUS(state, TASK_READ,
TASK_STATUS_EXECUTING));

   /* send: readout,1 */

/* A dezingered backround frame is automatically collected from four
images and put in the background buffer */

/* Get a sequence of data frames */
     while(1) {

        /* Wait for detector to NOT be acquiring (i.e. it has at least
         * started the previous read) */
           do {
              /* send: get_state */
              /* put result in state */
           }  while (TEST_TASK_STATUS(state, TASK_ACQUIRE,
TASK_STATUS_EXECUTING));

        /* Start detector frame acquisition */
           /* send: start */

        /* Wait for detector to start acquiring (this is very
         * important, so that no X-rays are on the detector during
         * readout; here could be a delay of approximately the
         * readout time) */
           do {
              /* send: get_state */
              /* put result in state */
           }  while (!TEST_TASK_STATUS(state, TASK_ACQUIRE,
TASK_STATUS_EXECUTING));

        /* Do exposure "stuff" here */

        /* End acquisition by starting readout, (correction and write
         * will be automatically queued and executed.) */

           /* send: readout,0,filename */
     }
```

Information on background frames and some sample data collection routines

The following are possible sequences of commands that you may implement in your remote mode control of *hsserver_legacy*. We assume here that your facility has implemented its own shutter control.

Either a "bias" frame (a background with zero integration time) or a non-zero time "dark" frame must always be collected and put in the Background buffer, to be subtracted from the data. Because of the extremely low CCD operating temperature, our X-ray detectors have minimal dark current; thus taking the time to collect a dark frame (as opposed to a bias frame) is usually not necessary, even for very long x-ray exposure times of data.

Here is the simplest and quickest method of collecting a Background image:

- [CLOSE SHUTTER] *(make sure shutter is closed)*
- `readout,1` *(the program automatically collects four images, averages and dezingers them, and puts the result in background buffer)*

The background doesn't have to be retaken for every data image taken, but generally should be retaken at the start of every new data set, or once every half hour, whichever is sooner (depending on the thermal stability of the hutch). For the SX Series detector, if a mismatch in the level of the 4 quadrants of data frames is noticed, the bias is probably drifting and should be recollected (and maybe should be set to be collected more often).

To collect a data image in single-frame mode (first collect Background image using above method, then proceed):

- `start` *(start integration)*
- [OPEN SHUTTER]
- [WAIT DESIRED TIME]
- [CLOSE SHUTTER]
- `readout,0,FILENAME` *(read data into raw frame buffer; queue the correction; corrected data are written to the filename)*

Note that in normal operation, neither the background frame nor the raw (uncorrected) data frame need to be saved.

To collect a timed series of 100 frames at 5Hz using shutterless mode (first collect Background image using above method, then proceed):

- [OPEN SHUTTER]
- `start_series,100,1,0,0.2,0,0,FILENAME,.mccd,5` *(collects the data series – note that the chosen parameters do not use any input synchronization to other experimental apparatus)*
- [WAIT UNTIL TASK_SERIES FINISHED]
- [CLOSE SHUTTER]

To take a dezingered data frame in single-frame mode (first collect Background image using above method, then proceed):

- `start` *(start first integration)*
- [OPEN SHUTTER]
- [WAIT TIME1]
- [CLOSE SHUTTER]
- `readout,2` *(read data into raw buffer and copy to Scratch)*
- `start` *(start second integration)*
- [OPEN SHUTTER]
- [WAIT TIME2]
- [CLOSE SHUTTER]
- `readout,0` *(read data into raw frame buffer)*
- `dezinger,0` *(dezinger from raw and Scratch data; data sent to raw buffer)*
- `correct` *(apply correction; data sent to "corrected" buffer)*
- `writefile,IMAGE,1` *(write data from corrected frame buffer to file)*

The dezinger operation goes through every pixel of the two (or multiple) separate reads of the detector, and compares the values. If the two values are very different, as determined by a statistical test, then the lower value is accepted and the higher value is discarded. If the values are statistically close enough, then they are averaged.

Because a statistical test is used, special care must be taken to make dezingered data frames. Each exposure must truly be the same (same X-ray dose, same movement of the sample or no movement of the sample, and very little decay or other change in sample). Otherwise the dezinger operation will yield unpredictable results.

If the source has constant intensity, then TIME1 = TIME2 = total_time/2. However, if the source has a short decay time, then the times must be TIME2>TIME1, calculated so that that both frames have equal dose, within a few percent.

Compiling the sample programs

The source programs to run hsserver_legacy are typically located in /opt/rayonix/src/marccd_server. Along with these instructions you should obtain a tar file called example_remote_server.tgz). Note that at this time, the sample client still uses the old protocol (with no series capability). If you have not already done so, unzip and untar the file in a new directory by typing "`tar -zxvf example_remote_server.tgz`."

Included in the untarred files will be:

dsmar_utils.c
dsmar_utils.h
Makefile
Makefile.bak
marccd.c
marccd_client_socket.c
marccd_server_pipe.c
marccd_server_socket.c
remote_mode_manual.pdf
socket_utils.c
socket_utils.h

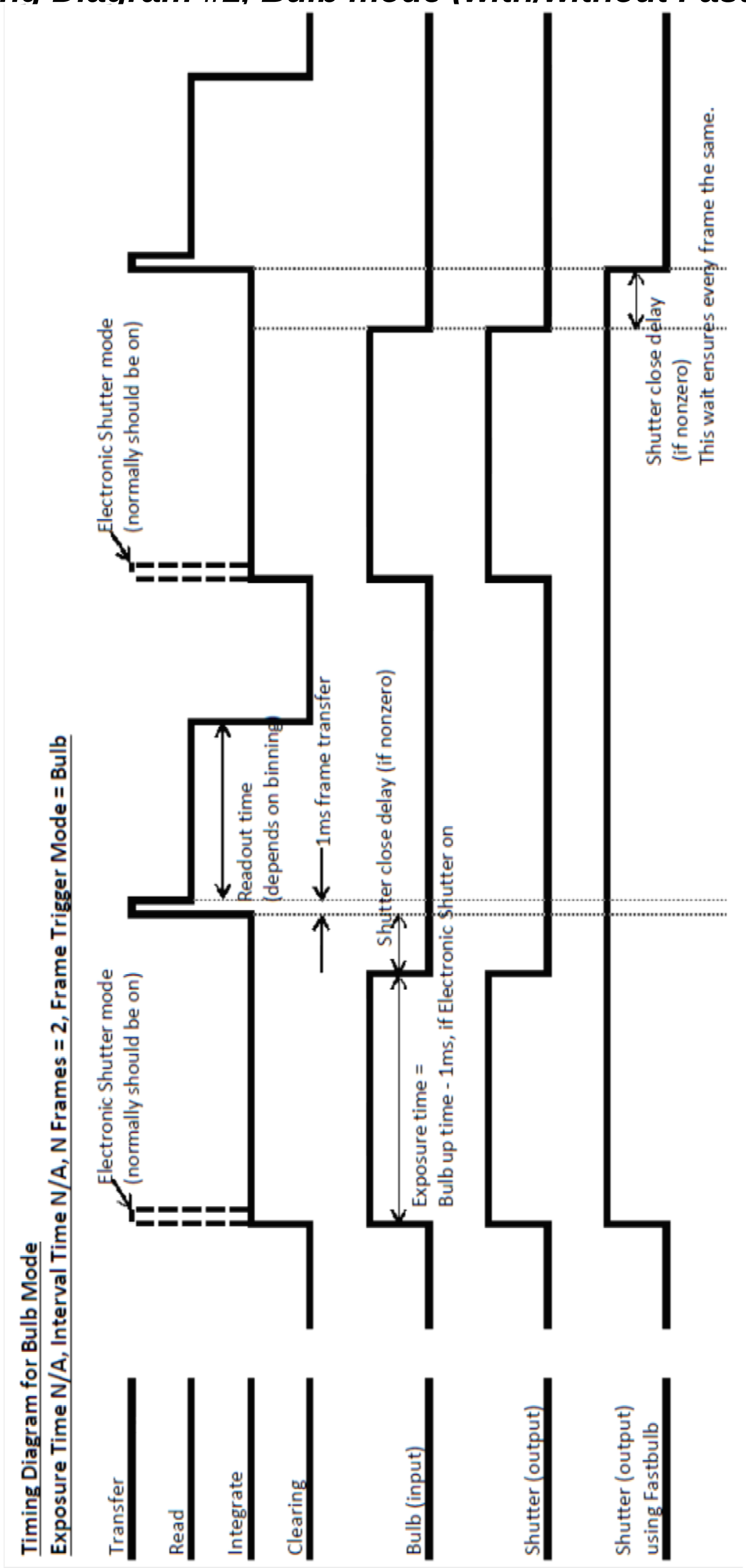
Before compiling any programs, type "`make depend`" in the current directory to update the dependencies in the Makefile to match the compiler libraries on your computer.

Compile marccd_client_socket.c and marccd_server_socket.c by typing "`make marccd_client_socket`" and "`make marccd_server_socket`." The file marccd_server_pipe.c is also provided as a sample to show how a connection can be made with *hsserver_legacy* using pipes, but in the example that follows, the programs with socket connections are used.

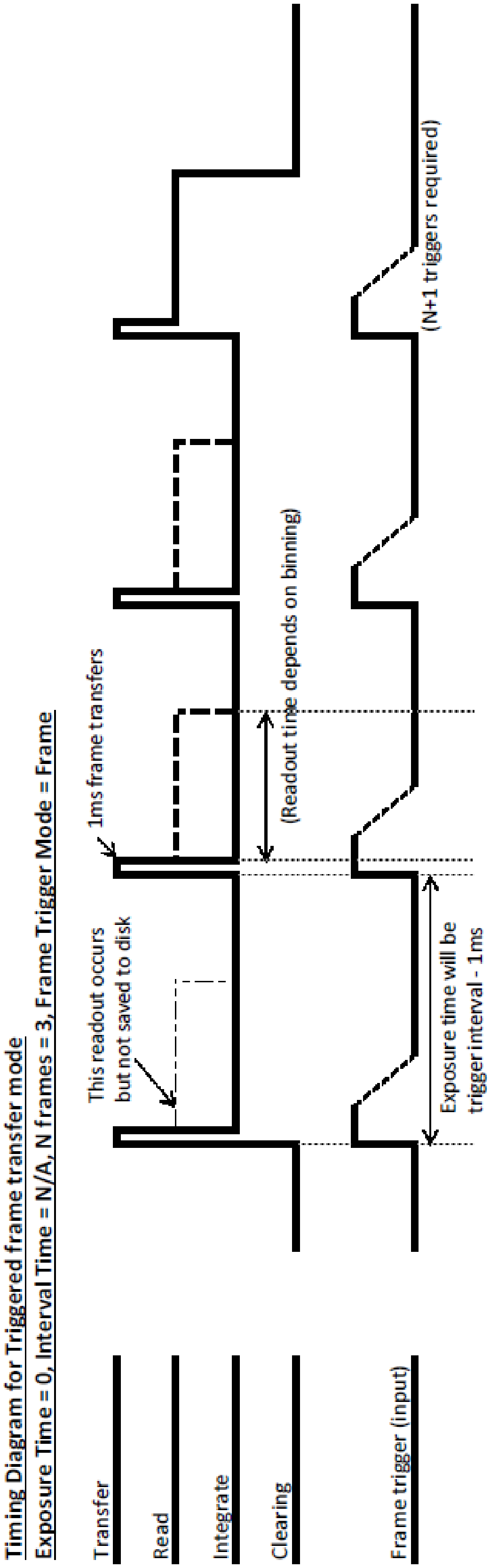
Compliled versions of marccd_server_socket and marccd_client_socket should be installed in /opt/rayonix/bin.

Chapter 11: Example Timing Diagrams

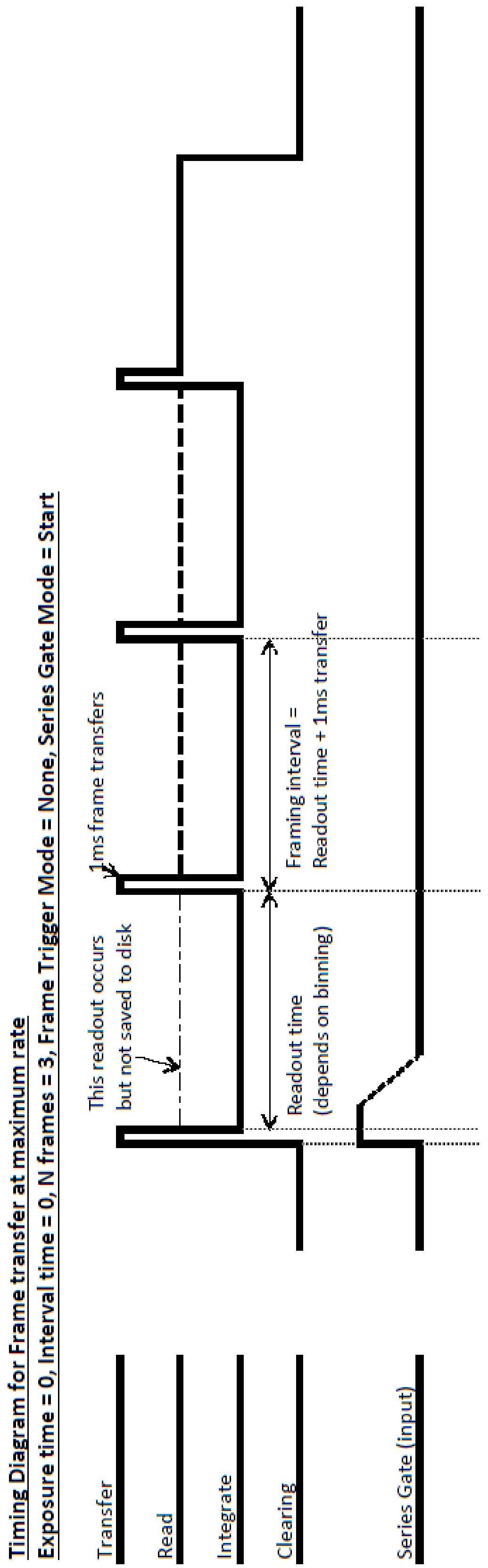
Example Timing Diagram #1, Bulb mode (with/without Fastbulb)


Timing Diagram for Bulb Mode
Exposure Time N/A, Interval Time N/A, N Frames = 2, Frame Trigger Mode = Bulb
Transfer
Read
Integrate
Clearing
Bulb (input)
Shutter (output)
Shutter (output) using Fastbulb
Electronic Shutter mode (normally should be on)
Exposure time = Bulb up time - 1ms, if Electronic Shutter on
Readout time (depends on binning)
1ms frame transfer
Shutter close delay (if nonzero)
Electronic Shutter mode (normally should be on)
Shutter close delay (if nonzero)
This wait ensures every frame the same.

Example Timing Diagram #2, Triggered frame transfer mode


Timing Diagram for Triggered frame transfer mode
Exposure Time = 0, Interval Time = N/A, N frames = 3, Frame Trigger Mode = Frame
Transfer
Read
Integrate
Clearing
Frame trigger (input)
This readout occurs but not saved to disk
1ms frame transfers
(Readout time depends on binning)
Exposure time will be trigger interval - 1ms
(N+1 triggers required)

Example Timing Diagram #3 – Frame transfer at maximum possible rate, and Series Gate


Timing Diagram for Frame transfer at maximum rate
Exposure time = 0, Interval time = 0, N frames = 3, Frame Trigger Mode = None, Series Gate Mode = Start
Transfer
Read
Integrate
Clearing
Series Gate (input)
This readout occurs but not saved to disk
1ms frame transfers
Readout time (depends on binning)
Framing interval = Readout time + 1ms transfer

Index