High Speed Series X-ray Detector Manual

Including:

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

Ross A. Doyle and Justin Anderson

Copyright © 2012-2013, Rayonix L.L.C.

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

## Introduction to the Manual

Which equipment do I have? ................................................................. 5
Type conventions used in the manual .................................................. 6

## Chapter 1: Capturing data with capxure

Introduction ....................................................................................... 7
Installation .......................................................................................... 7
Program Features .............................................................................. 7
Detector Control Tab .......................................................................... 9
Single Frame Collection Tab .............................................................. 11
Dataset Collection Tab ....................................................................... 14
Image Viewer Tab .............................................................................. 18
Multiple Detector Display ................................................................. 21
Configuration Options ........................................................................ 21

## Chapter 2: Detector Hardware Setup

How to connect and disconnect cooling gas lines ............................. 24
Setting up MX HS System connections from dismantled state ............ 25

## Chapter 3: Detector Startup

Startup Procedure ............................................................................... 27

## Chapter 4. Detector Control Hardware

MX-HS Series Trigger Input and Output Connections .......................... 30

## Chapter 5: Detector Shutdown Procedure

Detector Shutdown Procedure – Normal Method ............................... 31
Detector Shutdown Procedure – Fast Method ..................................... 31
Detector shipping procedure ............................................................... 32

## Chapter 6: Detector System Principles of Operation

X-ray CCD Detector Functionality .................................................... 33
Cooling system............................................................................................................................................................................. 38

Chapter 7: Safety and System Operating Conditions .................................................. 39
  Safety Warnings........................................................................................................................................................................... 39
  Operating Conditions ......................................................................................................................................................................... 40

Chapter 7: Troubleshooting and Service ................................................................. 43
  MX HS Series Troubleshooting guide ................................................................................................................................................... 43
  Rayonix Equipment Service ................................................................................................................................................................. 45

Chapter 8: Image Format (marccd) .......................................................................... 46
  File Size ............................................................................................................................................................................................... 46
  marccd Format Image Header Description ........................................................................................................................................ 46

Chapter 9: The Legacy Remote Mode for HS Detector Control .......................... 53
  Introduction ............................................................................................................................................................................................. 53
  Legacy remote mode communication diagram .................................................................................................................................. 53
  Configuring the legacy remote mode ............................................................................................................................................... 53
  The client program .............................................................................................................................................................................. 54
  Remote commands ............................................................................................................................................................................ 54
  Note on command handling by server application .......................................................................................................................... 58
  State and status values in remote mode version 1 .......................................................................................................................... 58
  Information on background frames and some sample data collection routines .................................................................................. 61
  Compiling the sample programs .................................................................................................................................................... 62

Index .............................................................................................................................................................................................................. 64
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 0.1.

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?

- **MX HS Series** refers to multiple-module X-ray detector systems, featuring tiled mosaics of frame transfer CCD modules. MX170-HS, MX225-HS, MX300-HS, MX340-HS, and MX425-HS feature arrays of 2x2, 3x3, 4x4, or 5x5 elements.

- **LX HS Series** refers to rectangular arrayed module X-ray detector systems, featuring tiled mosaics of frame transfer CCD modules, with thinned walls specific for getting close to the X-ray beam. These systems are designed for applications such as wide angle X-ray scattering (WAXS). The available versions include LX170-HS (1x2 array) and LX255-HS (1x3 array).
• **SX Series** refers to single module X-ray detector systems, featuring the Rayonix frame transfer CCD modules and various custom taper front elements.

**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 setpoint, 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 craydI) 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.

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
- 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
- Set number of Radial Plot bins
- Prompts for the number of radial bins to use. Bin size is the distance from center of circle to farthest corner of image divided by the number of bins.
- **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

**STATUS BAR**

At the bottom 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. On the right-hand side of the status bar is located 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”.
**Detector Control Tab**

The “Detector Control” tab allows the user to set the detector’s readout mode, CCD temperature setpoint, cooler temperature setpoint and shutter close delay. On this tab the detector may also be rebooted and the cooler can be enabled or disabled. Finally, the user can view a dump of the current detector parameter and status values.

**READOUT MODE SELECTION**

This box contains two 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. A low noise readout mode is also available which slows down the readout to yield images with very low read noise.

**GET PARAMETERS/STATUS VALUES**
Press this button to pop up a window containing a list of the current detector parameters and status values.

**REBOOT BUTTON**

Click this button to reboot the detector.

**COOLER POWER TOGGLE**

Un-check this checkbox to turn off the detector cooler. Check the box to turn it on.

**TEMPERATURE SETPOINT**

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°C to -80°C.

**COOLER SETPOINT**

This textbox shows the current cooler setpoint, which is the temperature at the coldhead. This value should not be changed from the default (-273°C) in normal operation.

**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 can occur (because the detector is reading out during the shutter close). If the value is too large, then having inserted an unnecessary delay wastes beam time.

**DETECTOR INFORMATION**

At the top-right hand corner 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.

**SAVE/REVERT**

Clicking the “Save” button will save the changes the user has entered. Pressing “Revert” will revert any changes back to their previous state.
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, choose an exposure type (“Normal”, “Dark” or “Test Image”), binning (e.g. 1920x1920 (2x2)), whether to save frames and which frame types to save. The user should also enter the filename options 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.

CONTINUOUS COLLECTION
Continuous collection mode will collect frames with identical exposure time and resolution until the “Stop” button is pressed. Enter the desired exposure time and hit the “Continuous” button. All other behavior is the same as for Single Frame Collection.

**COLLECT SERIES**

Series collection will collect the entered number of frames with identical exposure time and resolution. Enter the desired number of frames in the textbox and hit the “Go!” button. All other behavior is the same as for Single Frame Collection.

**SETTINGS**

- **Exposure Type**
  - Normal
    - Collect a regular illuminated frame
  - Dark
    - Collect a non-illuminated frame (of given exposure time)
  - Test Image
    - Collect a test image. A test image is generated by the detector controller and can be useful for troubleshooting.

- **Binning**
  - Here the user can choose the desired resolution (binning) for the collection.

- **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 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.

**STOP**

Hitting the “Stop” button will stop continuous or series frame collection after the current frame is done.
**Dataset Collection Tab**

![Image of the Dataset Collection Tab interface]

**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 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.

- **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.

### 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. A blue highlight means the segment is yet to be collected. Yellow highlight means collection of that 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.

- **Frame N**
  - The last frame number to collect for this segment. If “Frame 1” equals 1, this will be the number of frames to collect.

- **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.

- **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 “Num Segments” set to 3 and “Segment Size” set to 2. Frames 2 and 3 have “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**

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. 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 colored blue.

**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
  o The \((x, y)\) pixel coordinates of where the cursor currently is over the image.

• Object Position
  o 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
  o 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
  o This is the intensity value of the pixel under the cursor.

FRAME DISPLAY

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

• Live
  o Images are displayed as they are received from the Rayonix high-speed CCD detector library.

• Monitor Folder
  o 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 file will be displayed.

• Monitor EPICS PV
  o 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
  o 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
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**

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 included in the graph.

- **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 log10.
    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.

**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:

```
num_detectors = 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]
variable_name = 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:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>craydl_config_file</td>
<td>(String). Path to Rayonix craydl library configuration file.</td>
<td>None (Let craydl find it).</td>
</tr>
<tr>
<td>num_detectors</td>
<td>(Integer). Number of cameras in use. Additional windows will be created for displaying frames from the additional cameras.</td>
<td>1</td>
</tr>
<tr>
<td>detector_dist</td>
<td>(Integer). Sample to detector distance (mm).</td>
<td>500</td>
</tr>
<tr>
<td>wavelength</td>
<td>(Double). X-ray source wavelength (Å).</td>
<td>1.0</td>
</tr>
<tr>
<td>default_num_segments</td>
<td>(Integer). Number of segments to display in the dataset.</td>
<td>24</td>
</tr>
<tr>
<td>display_raw_frames</td>
<td>(True/False). Whether to display the raw frames in the frame display area.</td>
<td>true</td>
</tr>
<tr>
<td>display_background_frames</td>
<td>(True/False). Whether to display the background frames in the frame display area.</td>
<td>true</td>
</tr>
<tr>
<td>display_corrected_frames</td>
<td>(True/False). Whether to display the corrected frames in the frame display area.</td>
<td>true</td>
</tr>
<tr>
<td>display_every_n_frames</td>
<td>(Integer). How often to display frame in GUI.</td>
<td>1</td>
</tr>
<tr>
<td>img_folder_monitor_poll_int</td>
<td>(Integer). Time (ms) between checks of folder when running in folder monitoring mode.</td>
<td>1000</td>
</tr>
</tbody>
</table>
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 1 - 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 MX HS System connections from dismantled state

The following figure is an overview of the system components.

- Refer to previous page to learn how to connect/disconnect cooling gas lines.
- If they are not yet attached, “T” 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.
- Connect cooling gas lines. “SEND” fittings always connect to “SEND” fittings (green color). “RETURN” fittings always connect to “RETURN” fittings (red color).
- Connect both power cables to Power Supply (on Power Supply / PC cabinet).
- Connect vacuum hose. (Detector is normally supplied with 1.8m hose using KF-25 fitting for MX225-HS, MX300-HS, and MX340-HS; or ¼ inch Swagelok VCO fitting for LX170-HS, MX170-HS, and LX255-HS.)
- Connect CameraLink connections between head and PC.
Figure 3 – Back view of MX225-HS detector head

Figure 4 - 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. Power on Controller Cabinet cabinet power. This switch is on the BACK of the cabinet, labeled “Rayonix Cabinet.”

2. Boot up the data collection computer using the rack mounted monitor/keyboard. Log in to the xray user account (hsuser, password hsuser), start the capxure software by typing “capture”.

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 5).

Figure 5 - Pfeiffer TMP controls
Use the Scroll buttons (left or right) shown on Figure 5 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.**

The switch is located on the front of the Controller Cabinet, labeled “Main Power for Detector Power Supply.”

![Controller Cabinet front view](image)
5. **Power on the Cooler Cabinet. Then power on both of the Cryocontrollers.**

![Cooler Cabinet power switches](image)

Figure 7 - Cooler Cabinet power switches

6. **Verify that the coolers automatically came on, and wait ~3-4 hours for the detector to cool the detector to the operating temperature (-75°C) 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 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.

![MX-HS Series Shutter and Trigger pinout diagram for TRIGGER connector](image)

Cabling for shutter control: In the pinout diagram in Figure 8, the pins for SHUTTER+ and – are the first output trigger, and are used to output the shutter control signal.

Cabling for triggering the start of fast data series: The default software configuration is that TRIGGER+ and – and TRIGGER2+ and – are both needed to trigger the start and each shift/readout of a data series. (The easiest way to do this is to input the same signal to both sets of pins.) The default configuration of TRIG-OUT2+ and – is for monitoring the busy/ready status of the detector.

Signal configuration: The default signal configuration is 5V TTL. The software configuration can be changed to Open Collector mode instead. Please contact Rayonix for further information.

Note: After connecting a new shutter, the software’s shutter close delay parameter may need to be adjusted (to account for the time it takes for the shutter to close).
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 plate cover over the front window and attach with the thumb screws provided.

2. Turn off the cooler power via the software 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 (> 20°C).

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 5) 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 5) 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 plate cover to protect the detector front window.
• Always ship using the padded crate originally shipped with detector head.
Chapter 6: Detector System Principles of Operation

**X-ray CCD Detector Functionality**

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 9)**

  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 9)**

  The maximum sizes of CCD chips are too small to be direct detectors for many applications (for example, the SX-165 61 x 61 mm² 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. instantaneous 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 10 shows a one-port CCD and a four-port CCD during readout.

![Figure 10 – 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.](image-url)
• **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-designed and manufactured Rayonix Split Frame Transfer CCD sensor. This device has two major advanced features that improve frame rate and duty cycle compared to a conventional sensor: split frame transfer, and 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).

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.

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

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 (SX Series) or maintaining vacuum (MX Series)

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.

For the HS Series, a TMP unit runs constantly and maintains the vacuum during operation, and the vacuum valve is controlled by a system with built-in electronic safety checks. The controller monitors the pressure and closes the valve when the line pressure rises higher than the pressure inside the chamber. Nevertheless, 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 0 Hz before disconnecting. See the previous section about vacuum clamps on the MX Series detector head.

**Operating Conditions**

1. Electrical requirements
## Table 1 – 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).

<table>
<thead>
<tr>
<th>Model</th>
<th>Maximum KVA</th>
<th>Max. Amps @100V</th>
<th>100-120V Circuit Capacity (Amps)</th>
<th>Max. Amps @200 V</th>
<th>200-240V Circuit Capacity (Amps)</th>
<th>Heat Output (Watts)</th>
<th>Heat Output (BTU/Hr)</th>
<th>Cooling Required (Tons)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mar345</td>
<td>1060</td>
<td>10.6</td>
<td>20</td>
<td>5.3</td>
<td>10</td>
<td>400</td>
<td>2000</td>
<td>0.2</td>
</tr>
<tr>
<td>MarDTB</td>
<td>180</td>
<td>1.8</td>
<td>5</td>
<td>0.9</td>
<td>5</td>
<td>100</td>
<td>1000</td>
<td>0.1</td>
</tr>
<tr>
<td>SX-165</td>
<td>810</td>
<td>8.1</td>
<td>15</td>
<td>4.1</td>
<td>10</td>
<td>900</td>
<td>4000</td>
<td>0.2</td>
</tr>
<tr>
<td>MX-225 (std/HE)</td>
<td>1690</td>
<td>16.9</td>
<td>30</td>
<td>8.5</td>
<td>15</td>
<td>1700</td>
<td>6000</td>
<td>0.5</td>
</tr>
<tr>
<td>Pump</td>
<td>610</td>
<td>6.1</td>
<td>10</td>
<td>3.1</td>
<td>5</td>
<td>700</td>
<td>3000</td>
<td>0.3</td>
</tr>
<tr>
<td>Total</td>
<td>2300</td>
<td>23.0</td>
<td>35</td>
<td>(30+10)</td>
<td>11.5</td>
<td>2300</td>
<td>8000</td>
<td>0.7</td>
</tr>
<tr>
<td>MX-300 (std/HE)</td>
<td>3380</td>
<td>33.8</td>
<td>55</td>
<td>(2 x 30)</td>
<td>16.9</td>
<td>(2 x 15)</td>
<td>3400</td>
<td>12000</td>
</tr>
<tr>
<td>Pump</td>
<td>610</td>
<td>6.1</td>
<td>10</td>
<td>3.1</td>
<td>5</td>
<td>700</td>
<td>3000</td>
<td>0.2</td>
</tr>
<tr>
<td>Total</td>
<td>3990</td>
<td>39.9</td>
<td>60</td>
<td>(30+40)</td>
<td>20.0</td>
<td>(15+20)</td>
<td>4000</td>
<td>14000</td>
</tr>
<tr>
<td>MX-325 (std/HE)</td>
<td>3380</td>
<td>33.8</td>
<td>55</td>
<td>(2 x 30)</td>
<td>16.9</td>
<td>(2 x 15)</td>
<td>3400</td>
<td>12000</td>
</tr>
<tr>
<td>Pump</td>
<td>610</td>
<td>6.1</td>
<td>10</td>
<td>3.1</td>
<td>5</td>
<td>700</td>
<td>3000</td>
<td>0.3</td>
</tr>
<tr>
<td>Total</td>
<td>3990</td>
<td>39.9</td>
<td>60</td>
<td>(30+40)</td>
<td>20.0</td>
<td>(15+20)</td>
<td>4000</td>
<td>14000</td>
</tr>
<tr>
<td>MX225HS (idle)</td>
<td>~2000 (tbd)</td>
<td>25 (tbd)</td>
<td>30</td>
<td></td>
<td>14 (tbd)</td>
<td>15</td>
<td>2000</td>
<td>7000</td>
</tr>
<tr>
<td>MX225HS (@100fps)</td>
<td>~2500 (tbd)</td>
<td>25 (tbd)</td>
<td>30</td>
<td></td>
<td>14 (tbd)</td>
<td>15</td>
<td>2500</td>
<td>9000</td>
</tr>
<tr>
<td>Pump</td>
<td>610</td>
<td>6.1</td>
<td>10</td>
<td>3.1</td>
<td>5</td>
<td>700</td>
<td>3000</td>
<td>0.3</td>
</tr>
<tr>
<td>Total</td>
<td>~3100 (tbd)</td>
<td>23.0</td>
<td>35</td>
<td>(30+10)</td>
<td>11.5</td>
<td>20</td>
<td>(15+5)</td>
<td>3500</td>
</tr>
</tbody>
</table>

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 7: Troubleshooting and Service

### MX HS Series Troubleshooting guide

<table>
<thead>
<tr>
<th>problem/symptom</th>
<th>probable cause</th>
<th>fix</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Streaks, horizontal</strong></td>
<td><strong>Shutter problem:</strong> The shutter remains open after the detector has stopped integrating (i.e. exposure is finished) and starts to read out.</td>
<td>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.</td>
</tr>
<tr>
<td>Example: An image of a crystal diffraction pattern shows long vertical streaks starting/ending at diffraction spots.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Blank zone, horizontal stripe in center</strong></td>
<td><strong>Shutter problem:</strong> Shutter opens and closes before the detector starts integrating.</td>
<td></td>
</tr>
<tr>
<td>It looks as if the Image has been cut in half horizontally and the two halves have been shifted apart by the width of the blank zone.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Noise</strong></td>
<td><strong>Experimental Setup:</strong> Air scatter, fibers in beam, diffuse scatter... <strong>Temperature:</strong> The detector is not cooled to the proper operating temperature.</td>
<td>Check the beam path and crystal mount. Check the pressure (detector status window within main software window).</td>
</tr>
<tr>
<td>High background noise in images.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Pressure</strong></td>
<td><strong>Vacuum decayed:</strong> The vacuum in the detector chamber may not be good enough to allow cooling the detector to the optimal operating temperature.</td>
<td>Make sure that the TMP is operating at 1500 Actual Rotation Speed (Act Rotspd), and next make sure that the valve is plugged in to Vac. Valve jack in detector head.</td>
</tr>
<tr>
<td>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.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>problem/symptom</td>
<td>probable cause</td>
<td>fix</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
<td>-----</td>
</tr>
<tr>
<td><strong>Question marks:</strong>&lt;br&gt;The main capture window has question marks for the temperature and pressure readings.</td>
<td><strong>Lost Communication:</strong>&lt;br&gt;The software has lost communication with the detector.</td>
<td>Check: &lt;br&gt;1) Power to controller is on and all voltage LEDs are green, no warning LEDs on.&lt;br&gt;2) All switches are on.&lt;br&gt;3) Cable connectors are OK.&lt;br&gt;4) Fuses are not burned out. Power cycle detector and computer, restart software.</td>
</tr>
<tr>
<td><strong>Blank image</strong></td>
<td><strong>Experimental setup:</strong>&lt;br&gt;No X-rays?&lt;br&gt;&lt;br&gt;<strong>Shutter problem:</strong>&lt;br&gt;Does not open?&lt;br&gt;&lt;br&gt;<strong>Background image:</strong>&lt;br&gt;A frame with large values for all pixels is stored as background. Automatic subtraction of this faulty background produces an all-zero image.</td>
<td>Turn X-rays on.&lt;br&gt;Check shutter manually, make sure all cables are hooked up properly.&lt;br&gt;Collect new background frame.</td>
</tr>
<tr>
<td><strong>“Could not open file...” error message</strong></td>
<td><strong>Permissions wrong.</strong></td>
<td>Type chmod 666 filename (return)</td>
</tr>
<tr>
<td><strong>One or more quadrants of the images are missing.</strong></td>
<td><strong>Bad background frame.</strong></td>
<td>Recollect background</td>
</tr>
<tr>
<td><strong>Beam stop misaligned</strong></td>
<td></td>
<td>A beamstop should be used to prevent very large overexposures of pixels by the direct beam.</td>
</tr>
<tr>
<td><strong>“weak beam”</strong></td>
<td><strong>Aluminum cover on front of detector.</strong>&lt;br&gt;Rotating Anode: <em>Not at full power.</em>&lt;br&gt;Rotating Anode or Synchrotron: <em>Slits completely closed or base is misaligned.</em>&lt;br&gt;Beam conditioning optics are out of alignment.</td>
<td>Remove cover.&lt;br&gt;Slowly turn power up on the rotating anode.&lt;br&gt;Open slits (0.4mm or as desired). Re-align base as described in the manual section called “Error! Reference source not found.” Refer to alignment procedure for your optics.</td>
</tr>
</tbody>
</table>
The coolers are commanded on but the detector temperature does not go down.

<table>
<thead>
<tr>
<th>problem/symptom</th>
<th>probable cause</th>
<th>fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>One or more coolers not running</td>
<td>Check to make sure the pressure gauge indicators show high pressure for the SUPPLY side and low pressure for the RETURN side.</td>
<td>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.</td>
</tr>
<tr>
<td>One cooler running intermittently</td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>Blockage in a coldhead</td>
<td></td>
<td>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).</td>
</tr>
<tr>
<td>One cooler not running / stalled</td>
<td></td>
<td>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.</td>
</tr>
</tbody>
</table>

Rayonix Equipment Service

- 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 8: 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. 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 2 - Example file sizes for various resolutions of Rayonix detectors

<table>
<thead>
<tr>
<th>Resolution</th>
<th>File size (exact) (bytes)</th>
<th>File size (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>512 x 512</td>
<td>528384</td>
<td>0.5</td>
</tr>
<tr>
<td>768 x 768</td>
<td>1183744</td>
<td>1.1</td>
</tr>
<tr>
<td>1024 x 1024</td>
<td>2101248</td>
<td>2</td>
</tr>
<tr>
<td>1536 x 1536</td>
<td>4722688</td>
<td>4.5</td>
</tr>
<tr>
<td>2048 x 2048</td>
<td>8392704</td>
<td>8</td>
</tr>
<tr>
<td>3072 x 3072</td>
<td>18878464</td>
<td>18</td>
</tr>
<tr>
<td>4096 x 4096</td>
<td>33558528</td>
<td>32</td>
</tr>
</tbody>
</table>

marccd Format Image Header Description

Marccd Header Documentation

from C code in frame.h and types.h

Documentation updated by R. Doyle Mon Mar 22 13:53:00 CDT 2010
Documentation updated by M. Blum Tue Oct 11 11:49:20 CDT 2005

Description documents marccd v0.20.0

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 orientation 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 */
# Rayonix High Speed Detector manual, v. 0.1

```
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 succesive 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
```

Copyright © 2012, Rayonix, L.L.C.
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_nsnow; /* 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 preceding imaging pixels, fast direction */
UINT32 prescan/nsnow; /* Number of non-image pixels preceding imaging pixels, slow direction */
UINT32 postscan_nfast; /* Number of non-image pixels following imaging pixels, fast direction */
UINT32 postscan_nsnow; /* Number of non-image pixels following imaging pixels, slow direction */
UINT32 prepost_trimmed; /* trimmed==1 means pre and post scan pixels have been removed */
char reserve1[(64-55)*sizeof(INT32)-16];

/* 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;
INT32 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 image */
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_xtal_to_detector; /* 1000*dist 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_xtal_to_detector; /* 1000*dist 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 around 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 Goniostat 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...
*/

/* 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 parameters to 1024 bytes */

/* Dataset parameters (512 bytes) */
char dataset_comment[512]; /* comments - can be used as
desired */

/* Reserved for user definable data - will not be
     used by Rayonix! */
char user_data[512];

/* char pad[----] USED UP! */
/* pad out to 3072 bytes */

} frame_header;
This manual explains how to set up a client compatible with the program hsserver_legacy. Program hsserver_legacy

**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.

**Legacy remote mode communication diagram**

![Diagram](image)

Figure 11 – Legacy remote mode communication path

**Configuring the legacy remote mode**

The file /opt/rayonix/configuration/craydl/RemoteModeEmulator.conf contains the following configurable variables (with suggested defaults in the provided config 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 2002.

• **ServerLog**: not yet implemented at this time.

Typically no change would be required for these parameters.

**The client program**

A sample client program marccd_client_socket is included in the legacy remote mode files. 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]”.

**Remote commands**

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

<table>
<thead>
<tr>
<th>Remote Mode Command</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>get_size</code></td>
<td><em>hsserver_legacy</em> will answer with the fast (x) and slow (y) dimensions of the data frame.</td>
</tr>
<tr>
<td><code>get_size_bkg</code></td>
<td><em>hsserver_legacy</em> will answer with 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_bin</code></td>
<td><em>hsserver_legacy</em> will answer with the fast (x) and slow (y) binning of the data frame.</td>
</tr>
<tr>
<td><code>set_bin,x,y</code></td>
<td><em>hsserver_legacy</em> will set the fast (x) and slow (y) binning of the data frame.</td>
</tr>
<tr>
<td><code>start</code></td>
<td><em>hsserver_legacy</em> will start integrating data (stop clearing) on the CCD.</td>
</tr>
</tbody>
</table>
**readout,flag[,filename]**

*hsserver_legacy* will stop integrating and start reading the CCD; given filename(s), it will queue the correction and writing of the file to disk.

<table>
<thead>
<tr>
<th>Flag</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<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</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>
</tbody>
</table>

**dezinger,flag**

***TO BE IMPLEMENTED*** *hsserver_legacy* 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>
<thead>
<tr>
<th>Flag</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<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>
</tbody>
</table>

**correct**

*hsserver_legacy* will apply geometric and flatfield corrections to the raw data frame.

**writefile, filename, flag**

***TO BE IMPLEMENTED*** *hsserver_legacy* will write out a data frame to a file on disk. The parameter filename is the name of the file to be written.

<table>
<thead>
<tr>
<th>Flag</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>write raw file</td>
</tr>
<tr>
<td>1</td>
<td>write corrected file</td>
</tr>
</tbody>
</table>

**abort**

*hsserver_legacy* will abort the current operation. Normally this would be done to stop integration and return the CCD to continuous clear mode.

**get_temp**

***TO BE IMPLEMENTED*** Returns the current CCD temperature, or the highest (warmest) CCD temperature in a detector with multiple CCDs.

**get_press**

***TO BE IMPLEMENTED*** Returns the current pressure inside the detector head.

**get_stability**

***TO BE IMPLEMENTED*** Requires valid Baseline stabilization mode license. See manual section describing Baseline stabilization.

**set_stability, target**

***TO BE IMPLEMENTED*** Requires valid Baseline stabilization mode license. See manual section describing Baseline stabilization.

**get_roi**

***TO BE IMPLEMENTED*** Requires valid Region of Interest mode license. See manual section describing Region of Interest.
**set_roi,x0,y0,x1,y1**

***TO BE IMPLEMENTED*** Requires valid Region of Interest mode license. See manual section describing Region of Interest.

**header,header_data**

***TO BE IMPLEMENTED*** hsserver_legacy will accept header_data and interpret item=value pairs to be placed into the data frame header. header_data consists of a list of item=value pairs separated by commas and terminated by a newline (\n). The following items are understood:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type (Units)</th>
</tr>
</thead>
<tbody>
<tr>
<td>detector_distance</td>
<td>float (mm)</td>
</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>start_phi</td>
<td>float (deg)</td>
</tr>
<tr>
<td>rotation_axis</td>
<td>string (omega, chi, kappa, phi, gamma, delta, or xtal_to_detector)</td>
</tr>
<tr>
<td>rotation_range</td>
<td>float (deg)</td>
</tr>
<tr>
<td>source_wavelength</td>
<td>float (angstroms)</td>
</tr>
<tr>
<td>file_comments</td>
<td>string</td>
</tr>
<tr>
<td>dataset_comments</td>
<td>string</td>
</tr>
</tbody>
</table>

**get_state**

hsserver_legacy will answer with the current state of the system.

For remote mode version 1, “state” has been superseded by the more complete “status,” which is returned by the get_state command. The command get_state will return the more complex “status,” which includes the state in the lower 4 bits. Only the states IDLE, ERROR and BUSY will ever be seen. See the section below for the discussion of the version 1 protocol.

The integer numbered states possible in remote mode version 0 are:

<table>
<thead>
<tr>
<th>State Number</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>IDLE</td>
</tr>
<tr>
<td>6</td>
<td>UNAVAILABLE</td>
</tr>
<tr>
<td>7</td>
<td>ERROR</td>
</tr>
<tr>
<td>8</td>
<td>BUSY</td>
</tr>
</tbody>
</table>

**set_state,state**

***NOT IMPLEMENTED*** hsserver_legacy will set the state to the desired state. This is for testing purposes only and has no use in a normally functioning system.
### shutter, flag

`hsserver_legacy` will set the shutter state to either closed or open. (Only if `hsserver_legacy` controls the shutter!) If the MarDTB is used, this function controls the MarDTB shutter. Otherwise, it controls the shutter attached to the shutter input on the detector controller unit.

<table>
<thead>
<tr>
<th>Flag</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>closed</td>
</tr>
<tr>
<td>1</td>
<td>open</td>
</tr>
</tbody>
</table>

### end_automation

`hsserver_legacy` will exit remote mode.

### start_series_timed

![Syntax](image)

Starts a data collection series which is timed by the detector’s internal clock (asynchronous to the user’s experiment). 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). 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.

### start_series_triggered,

![Syntax](image)

Starts a data collection series which is triggered by the user’s input trigger pulses (usually timed to the experimental conditions, but also could be triggered asynchronously using a pulse generator). A background image is required to have already been acquired. `Bulb_mode` default is 0; if set to 1 it requires input trigger signal which starts acquisitions on rising edge and transfer/readout on falling edge. `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). 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.

### set_readout_mode, flag

Program will set the readout mode to one of the following values:

<table>
<thead>
<tr>
<th>Flag</th>
<th>Readout Mode</th>
<th>Gain</th>
<th>Speed</th>
<th>Read bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standard</td>
<td>norm</td>
<td>norm</td>
<td>16</td>
</tr>
<tr>
<td>1</td>
<td>High Gain</td>
<td>high</td>
<td>norm</td>
<td>16</td>
</tr>
<tr>
<td>2</td>
<td>Low Noise</td>
<td>high</td>
<td>med</td>
<td>16</td>
</tr>
<tr>
<td>3</td>
<td>HDR</td>
<td>high</td>
<td>low</td>
<td>18</td>
</tr>
</tbody>
</table>

Note that some data processing programs may need to be updated to read 18 bit HDR (High Dynamic Range) mode files.

### get_readout_mode

Returns the current readout mode setting, with flags defined above in set_readout_mode section.
Note on command handling by server application

In addition to the above commands, it is recommended that any server application implements the following commands (already implemented in the provided marccd_server_socket):

<table>
<thead>
<tr>
<th>Command to server program</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_state</td>
<td>same as above, but queries from the client should be answered directly by the server without querying hsserver Legacy.</td>
</tr>
<tr>
<td>get_size</td>
<td>same as above, but queries from the client should be answered directly by the server without querying hsserver Legacy.</td>
</tr>
<tr>
<td>get_size_bkg</td>
<td>same as above, but queries from the client should be answered directly by the server without querying hsserver Legacy.</td>
</tr>
<tr>
<td>get_frameshift</td>
<td>same as above, but queries from the client should be answered directly by the server without querying hsserver Legacy.</td>
</tr>
<tr>
<td>get_bin</td>
<td>same as above, but queries from the client should be answered directly by the server without querying hsserver Legacy.</td>
</tr>
<tr>
<td>get_state_hist</td>
<td>(Implemented completely in the server.) Answers with the current state and the most recent previous state, separated by commas. (See get_state.)</td>
</tr>
</tbody>
</table>

State and status values in remote mode version 1

In the version 1 protocol, the status of each task is represented in a 4 bit field in the 32 bit state value. To use version 1 instead of version 0, include the appropriate configuration file, marccd_server_v1.conf, instead of the older marccd_server.conf file. This file contains the parameter “remote_mode_version” set to 1.

The task values are:

<table>
<thead>
<tr>
<th>Task Number</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>TASK_ACQUIRE</td>
</tr>
<tr>
<td>1</td>
<td>TASK_READ</td>
</tr>
<tr>
<td>2</td>
<td>TASK_CORRECT</td>
</tr>
<tr>
<td>3</td>
<td>TASK_WRITE</td>
</tr>
<tr>
<td>4</td>
<td>TASK_DEZINGER</td>
</tr>
<tr>
<td>5</td>
<td>TASK_SERIES</td>
</tr>
</tbody>
</table>

The status bits for each task are:

<table>
<thead>
<tr>
<th>Task Status Bit</th>
<th>Task Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x1</td>
<td>TASK_STATUS_QUEUED</td>
</tr>
<tr>
<td>0x2</td>
<td>TASK_STATUS_EXECUTING</td>
</tr>
<tr>
<td>0x4</td>
<td>TASK_STATUS_ERROR</td>
</tr>
<tr>
<td>0x8</td>
<td>TASK_STATUS_RESERVED</td>
</tr>
</tbody>
</table>
Therefore, the state value looks like Figure 12, with eight four-bit fields.

![Figure 12 - State fields in remote mode version 1](image)

Examples state values returned by get_state:

<table>
<thead>
<tr>
<th>State Description</th>
<th>State Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle</td>
<td>0x00000000</td>
</tr>
<tr>
<td>Busy (interpreting command)</td>
<td>0x00000008</td>
</tr>
<tr>
<td>Error (command not understood)</td>
<td>0x00000007</td>
</tr>
<tr>
<td>Acquiring</td>
<td>0x00000010</td>
</tr>
<tr>
<td>Reading</td>
<td>0x00000200</td>
</tr>
<tr>
<td>Reading w/correct and write queued</td>
<td>0x00011200</td>
</tr>
<tr>
<td>Correcting w/write queued:</td>
<td>0x00012000</td>
</tr>
<tr>
<td>Error writing file</td>
<td>0x00040000</td>
</tr>
</tbody>
</table>

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

```c
#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 `hsserverLegacy` code and can be used in the client code:

```c
#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:

```c
/* Get a background 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 */

/* Get a 2nd background frame - This (readout; dezinger) can be repeated if desired */
/* 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,2 */

/* Dezinger to combine 2 background frames into low noise dezingered background 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: dezinger,1 */

/* 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 (not recommended):

- [CLOSE SHUTTER] (make sure shutter is closed)
- start (start integration)
- readout,1 (read data into both raw and background buffers)

The reason it is not recommended is that this method will potentially have zingers in the image. Zingers in the background will be subtracted from data images, leaving the final images with zero intensity spots. In addition, one background can be used multiple times and therefore a zinger in a background will contaminate several images.

Here is a sequence that will make a dezingered bias frame (recommended method):

- [CLOSE SHUTTER] (make sure shutter is closed)
- start (start integration)
- readout,2 (read and copy to Scratch buffer)
- start
- readout,1 (read and copy to Background buffer)
- dezinger,1 (dezinger from Background and Scratch data, put image 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:

- start (start integration)
- [OPEN SHUTTER]
- [WAIT DESIRED TIME]
• [CLOSE SHUTTER]  (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.

Here is a sequence of commands for taking a dezingered data frame:

• 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 \(\text{TIME1} = \text{TIME2} = \frac{\text{total_time}}{2}\). However, if the source has a short decay time, then the times must be \(\text{TIME2}>\text{TIME1}\), calculated so 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). 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.

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

Background subtraction, 37
Cooling system, 29, 30, 32, 33, 39, 40, 41, 44
Correction, image, 37
Detector head, 41
Detector window, 40
  Cleaning, 40
Electronics, 41
File size, 47
Gain
  Detector, 36, 37
Gas lines, connecting and disconnecting, 23
Image headers, 47
Log in, 29
MX HS Detector, 5

Electrical requirements, 41
Operating conditions, 41
Power, 27, 28
Shipping procedure, 33
Shutdown procedure, 32
Startup procedure, 27
Phosphor screen, 34, 40
Quick Start, 7
Resolution, 47
Safety, 40
TMP, 27, 32, 33, 41
Type conventions, 6
Vacuum, 41
Zingers, 37, 38