marccd

Version 0.10.17
Software Manual

Including:
- MarCCD X-ray Detector System device manual
- MarMosaic X-ray Detector System device manual
- MarDTB Advanced Goniostat device manual
- MarCSC Cryogenic Sample Changer device manual

By Ross A. Doyle

Portions by Christine R. A. Muchmore and Michael Blum

Copyright © 2000-2006, Mar USA

Mar USA
1880 Oak Avenue
Evanston, IL 60201 USA
http://www.mar-usa.com

Table of Contents

Copyright © 2000-2006, Mar USA

Copyright © 2000-2006, Mar USA

Forward to the Manual

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

This is intended to be a complete manual for users of Mar USA's CCD-based detectors and accessories. These include: the MarCCD X-ray Detector System, the MarMosaic X-ray Detector System, the MarDTB (Mar DeskTop Beamline) Advanced Goniostat, and the MarCSC (Mar Cryogenic Sample Changer). This manual also covers the software control of this equipment via the program *marccd*. At the time of writing, the latest *marccd* release was version 0.10.17.

Not all items in the manual are applicable to the equipment you are using. Also, the program *marccd* is highly configurable and the way you are operating it may look different (e.g. different buttons and more or fewer tabbed windows). 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 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, additions, suggestions, and reports of manual errors and software bugs are welcome. See **Appendix B: Configuration Guide for *marccd*** for information about reporting software bugs. Suggestions for the manual can be emailed directly to our general email account at info@mar-usa.com.

Type conventions used in the manual

MarCCD refers to the X-ray detector system hardware. The word ***marccd*** (in italics) refers to the software program. The program *marccd* is the main software to control all the equipment.

Courier font (e.g. "`marccd -v`") is used to indicate commands to be entered into the computer screen, or characters copied from the computer screen.

Copyright © 2000-2006, Mar USA

Chapter 1: Quick Start Guide

This Quick Start Guide assumes that the detector is already running and is at the proper operating temperature. If you find that this is not the case, you must use the guide in Chapter 4: Detector Start-Up Procedures. Generally, one-time detector users should not make any physical changes to the detector (attaching cables, etc.) without the permission of the beamline or lab staff.

If the marccd *software is already running and the detector is at the operating temperature (the temperature and status is shown in upper portion of the software window; typical operating temperature is between -70 C and -80 C), then the user only needs to get started collecting data. Skip to Chapter 2: Data Collection with* marccd.

MarCCD Quick Start

Log into the computer controlling the MarCCD (default password shown):

```
user: marccd
password: marccd
```

The control and data collection program is called *marccd.* To start the *marccd* software, open a fresh terminal window and type:

```
marccd <return>
```

The *marccd* main window will appear on the screen as in Figure 1. Go to Chapter 2 to begin collecting data with *marccd.*

Copyright © 2000-2006, Mar USA

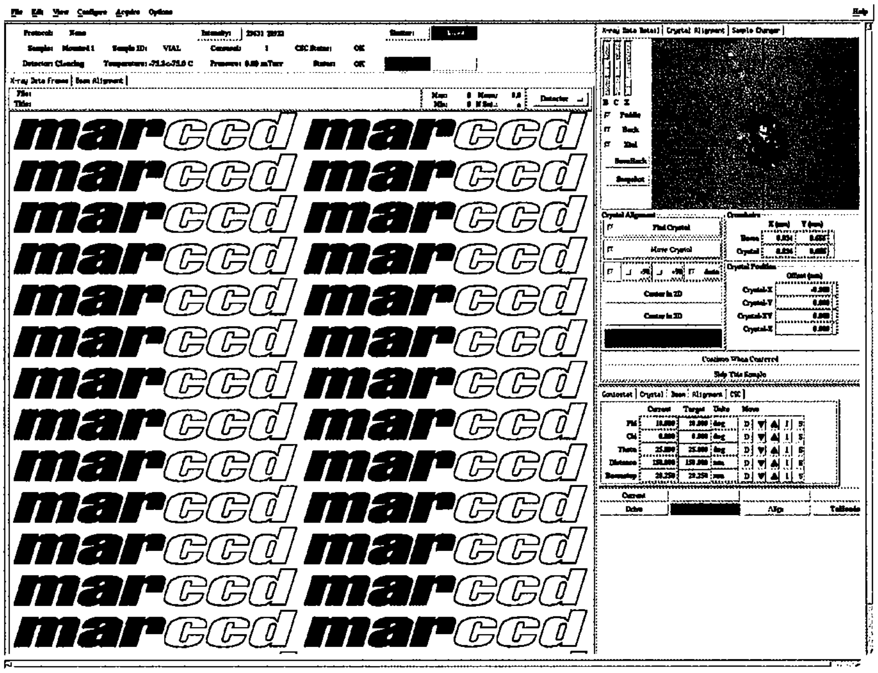


Figure 1 - The *marccd* main window

MarMosaic Quick Start

The control software for MarMosaic detectors is called *marccd.*

Log into the computer controlling the MarMosaic (default password shown):

```
user: marccd
password: marccd
```

To start the *marccd* software, open a fresh terminal window and type:

```
marccd <return>
```

The *marccd* main window will appear on the screen as in Figure 1, above. Go to Chapter 2 to begin collecting data with *marccd.*

Copyright © 2000-2006, Mar USA

Chapter 2: Data Collection with *marccd*

Acquire Single Frame dialog—to take single exposures

Use the left mouse button to choose from the pull-down menus on top of the main *marccd* window.

⇒ Click on Acquire, and then click Single Frame.

The Acquire Single Frame dialog box will appear, as shown in Figure 2. An explanation of each part of this dialog can be found below.

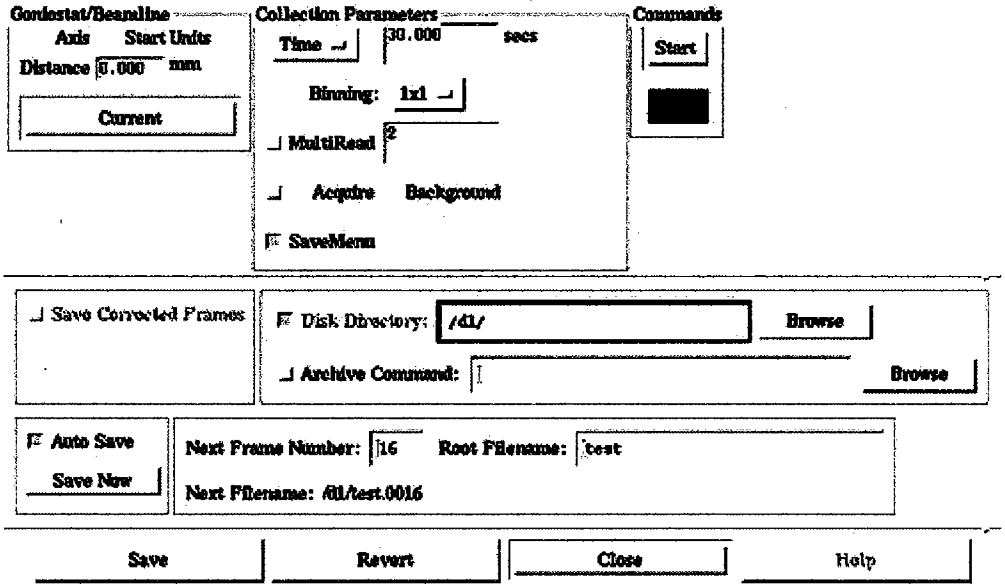

Distance: 0.000 mm
Time: 30.000 secs
Binning: 1x1
☐ MultiRead: 2
☐ Acquire Background
☑ SaveMenu
☐ Save Corrected Frames
☑ Disk Directory: /d1/
☐ Archive Command:
☑ Auto Save
Next Frame Number: 16
Root Filename: test
Next Filename: /d1/test.0016

Figure 2 - Acquire Single Frame dialog

- **Goniostat/Beamline options box**

  The motor axes that can be controlled or set are listed here. They may include: Distance, Phi, Crystal-Z, and others. Before the exposure, the software will drive (or set, if not under software control) the motors to the numbers typed into each box listed. Click "Current" to update the boxes with the current values of the motors.

- **Collection Parameters options box**

  - Select Time or Dose (though Dose Mode may not be available) and type in desired value.
  - Select desired detector Binning.

Copyright © 2000-2006, Mar USA

- Scan Width checkbox (not shown in Figure 2) is used only if a scannable Phi axis exists and is controlled by the program. Check the box to oscillate and enter the desired scan width (starting at value indicated in the Goniostat/Beamline box).
- Click the Multi-read checkbox if desired, and enter the desired number of reads. Multi-read is used for collecting dezingered images, useful for long exposures with weak signals. The X-ray signal must be constant intensity unless Dose mode is used. **Though any number of reads can be entered, we don't believe more than two reads is ever useful and don't recommend a value greater than two**. The exposure Time (or Dose) before each read will be the value entered into the field above divided by the number of reads selected.
- Click the Acquire Background checkbox to force the software to collect a new background image prior to X-ray exposure, by reading out the detector. A background is always subtracted from the diffraction image. If "Acquire Background" is **not** checked, the software will *still take a background frame* if there is a special reason (e.g. no background frame stored, or the stored frame is older than the configured timeout).
- SaveMenu checkbox opens the save menu options.

- **Save menu**

Click the SaveMenu checkbox to open a new section with save options. The Auto Save checkbox is used to automatically write the images to file. If checked, you must enter the following three values:

- Directory
- Root Filename
- Next Frame Number (this counter is automatically increment for each frame collected)

The Archive Command checkbox is explained in Chapter 3. Generally, this checkbox can be left unchecked.

- **Commands**

Once all the options are entered to your satisfaction, go to the bottom row of buttons.
⇒ Click "Start" to start the exposure.
⇒ Click "Close" if you wish to get rid of the panel. The exposure will continue. Note that every time you put the data acquisition menus away, you will be asked to Save, Revert, or Cancel. Click Save to save the current values of each text field and checkbox to be recalled the next time you open the menu. Click Revert to discard the current values. Click Cancel to keep the menu open. In general, click Save.
⇒ To abort an exposure, click on "Stop" on this menu or the main window. A question box will appear and ask whether to stop the image immediately or wait until the end of the exposure.

Viewing different image buffers

Copyright © 2000-2006, Mar USA

To view the most recently collected data frame, find the pop-up button like the one in Figure 3, above the top right corner of the data display area in the main *marccd* window.

Figure 3 - Display buffer selection

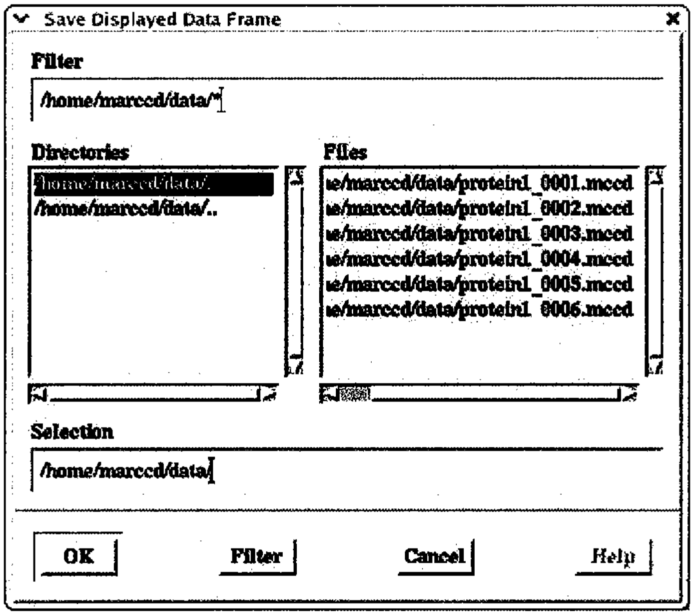
If not already set to "Detector," use the left mouse button to choose "Detector" (rather than "Background" or "Scratch") from this menu. Once the exposure is completed, the frame will be displayed in the "Detector" buffer

Manually saving the displayed data frame

Images collected are *not* automatically saved unless the Auto Save checkbox is checked before collecting the image and a valid directory is given (with storage space available).

To save the image from the buffer you are currently viewing, do the following:

⇒ In the menu bar, click File→Save Displayed Data Frame. The Save Displayed Data Frame dialog box will appear as in Figure 4.



Filter
/home/marccd/data/*

Directories
/home/marccd/data/.
/home/marccd/data/..

Files
ue/marccd/data/protein1_0001.mccd
ue/marccd/data/protein1_0002.mccd
ue/marccd/data/protein1_0003.mccd
ue/marccd/data/protein1_0004.mccd
ue/marccd/data/protein1_0005.mccd
ue/marccd/data/protein1_0006.mccd

Selection
/home/marccd/data/

Figure 4 - Save Displayed Data Frame dialog

- Some hints about using file dialogs:
  - Double-clicking a directory opens that directory
  - Clicking on a file name puts that file name in the Selection box
  - Alternatively, type directly into the Selection box
  - Click the Filter button if the wildcard characters were updated into the Filter, or to refresh the files and directories shown

Copyright © 2000-2006, Mar USA

  - Click cancel to dismiss the menu
  - If used often, file dialogs can be left open and moved to the side of the screen, even when doing other tasks with *marccd*
- Enter the desired file name into Selection, including the path

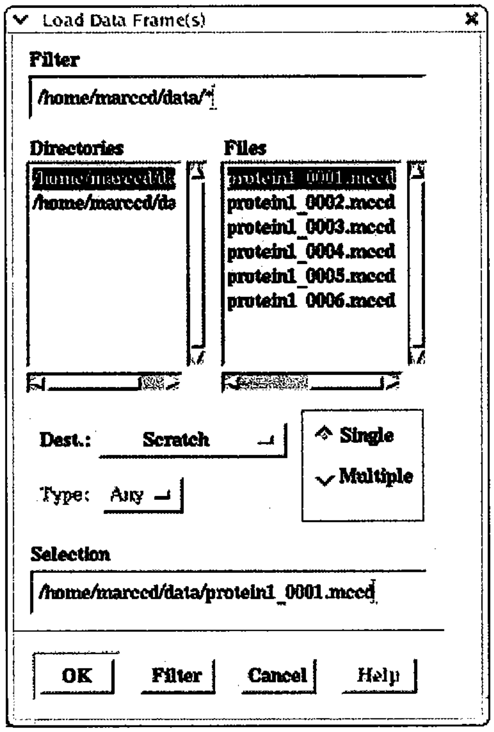
⇒ Click on "OK."

Note: Do not use "Save Displayed Image". The "Save Displayed Image" option saves a picture of the current display in 8-bit TIFF format for other purposes than data processing; it does not save the actual data frame. In addition, that feature is not available for 24-bit displays.

Loading a stored image

To load an image from the hard drive or other medium for viewing in *marccd*, do the following:

⇒ In the menu bar, click File→Load Data Frame(s). The Load Data Frame(s) dialog box will appear as in Figure 5.



Filter: /home/marccd/data/*

Directories: /home/marccd/da

Files:
protein1_0001.mccd
protein1_0002.mccd
protein1_0003.mccd
protein1_0004.mccd
protein1_0005.mccd
protein1_0006.mccd

Dest.: Scratch

Type: Any

◆ Single
✓ Multiple

Selection: /home/marccd/data/protein1_0001.mccd

Figure 5 - Load Data Frame(s) dialog

- See hints about using file dialogs in the previous section
- Enter the desired file name into Selection, including the path
- Select the destination buffer, such as "Scratch"

⇒ Click on "OK."

Copyright © 2000-2006, Mar USA

- Change the image buffer (as explained in the section above) to view the buffer where the image was loaded

Multiple files can be loaded at the same time using the “Multiple” button and extra options. Select a few files to load, and choose Average, Merge, or Sum; then click “OK.”

Acquire Dataset dialog—to collect a data set

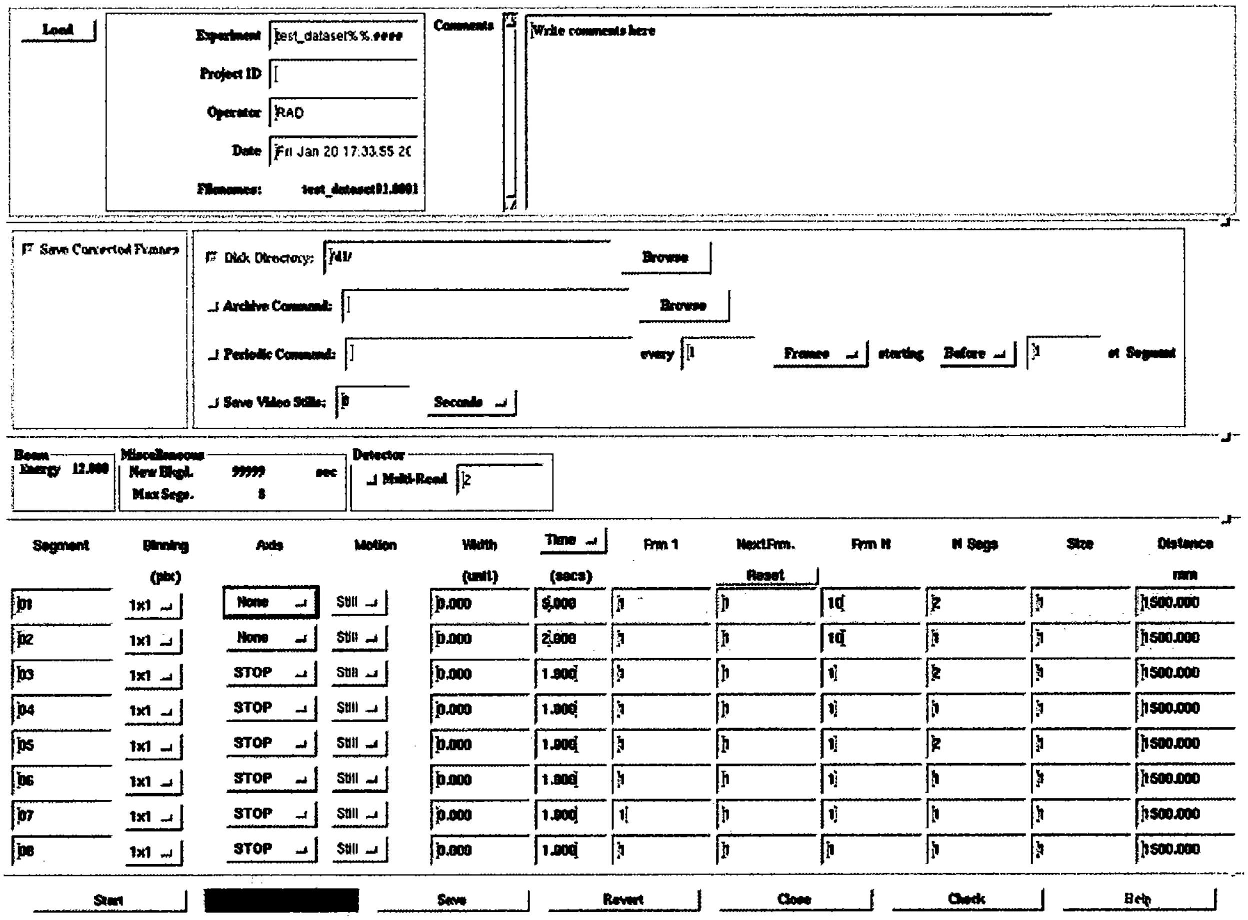
Note: This is just a simple example for collecting a data set. Please read the section Data Collection – Advanced Features in the next chapter of this manual for more sophisticated methods to collect your data. For MarCSC users, additional information is found in the MarCSC chapter.

Use the left mouse button to choose from the pull-down menus on top of the main *marccd* window:

⇒ Click on Acquire→Data Set.

The Acquire Data Set dialog box will appear as shown in Figure 6 (fewer columns, or more columns, or different parameters may be configured on your workstation).

Copyright © 2000-2006, Mar USA



Experiment: test_dataset%%.####
Project ID:
Operator: RAD
Date: Fri Jan 20 17:33:55 20
Filename: test_dataset01.0001
Comments: Write comments here

☑ Save Corrected Frames
☑ Disk Directory: /d1/
☐ Archive Command:
☐ Periodic Command: every 1 Frames starting Before 1 at Segment
☐ Save Video Stills: 0 Seconds

Beam: Energy 12.000
Miscellaneous: New Bkgd. 99999 sec; Max Segs. 8
Detector: ☐ Multi-Read 2

Segment	Binning (pix)	Axis	Motion	Width (unit)	Time (secs)	Frm 1	NextFrm. Reset	Frm N	N Segs	Size	Distance mm
01	1x1	None	Still	0.000	5.000	1	1	10	2	1	1500.000
02	1x1	None	Still	0.000	2.000	1	1	10	1	1	1500.000
03	1x1	STOP	Still	0.000	1.000	1	1	1	2	1	1500.000
04	1x1	STOP	Still	0.000	1.000	1	1	1	1	1	1500.000
05	1x1	STOP	Still	0.000	1.000	1	1	1	2	1	1500.000
06	1x1	STOP	Still	0.000	1.000	1	1	1	1	1	1500.000
07	1x1	STOP	Still	0.000	1.000	1	1	1	1	1	1500.000
08	1x1	STOP	Still	0.000	1.000	1	1	1	1	1	1500.000

Figure 6 - Acquire Data Set dialog

- **Top section (Filenames and Dataset Information)**
  - **Experiment and Filename**: As the Experiment field is filled in, the example filename is shown in the Filename field. Use one or more % characters (two recommended) to designate the segment number, and one or more # characters to designate the image number (four recommended). If no template characters are used (% and/or #), then the following template is used: **Experiment_field%%.####.**
  - **Project ID**: This can be left blank.
  - **Operator**: Write the operator's name or initials to be recorded in image headers.
  - **Date**: Write the date to be recorded in image headers.
  - **Comments**: Enter comments into this box to be recorded in image headers.
  - **Load button**: Click this button to load a data collection protocol (normally found in /home/marccd/protocols). Protocol files are used to automatically fill in the Data Acquisition dialog for commonly used procedures. Loading a protocol file may overwrite the fields above (Experiment, etc.), and it will fill in values to the protocol information in the dialog. See the Dataset Protocols section in Chapter 3 for more detailed information.

- **Second section (File Deposition and Periodic Commands)**

    - **Disk Directory**: Enter the entire path for the disk directory (or click Browse to browse for a directory).
    - **Archive Command**: This field is typically left blank. Archive commands are used to move or copy files after they are recorded to disk. If desired, check the checkbox and enter the archive command. See Chapter 3 for detailed information about Archive Commands.
    - **Periodic Command**: This field is typically left blank. Periodic commands are used for all kinds of tasks relating to a beamline. Anything that can be controlled by a computer and needs to be done during data collection can be done here. If desired, check the checkbox and enter the periodic command and periodicity. See Chapter 3 for detailed information about Periodic Commands.
    - **Save Video Stills**: Set these fields to capture video images of the crystal periodically. See the MarCSC chapter for information about saving video stills during data collection.

- **Third section (Experimental Conditions)**

Check that the goniostat is in the correct position, and beam information is correct by looking at the information here. This information will be recorded to headers to simplify record-keeping later.

Under the "Detector" heading is a Multi-Read checkbox. Check this if desired, and enter the desired number of reads. Multi-read is used for collecting dezingered images, useful for long exposures with weak signals. (Zingers are unwanted stray cosmic rays and nearby radioactive sources picked up by the detector in addition to the diffraction pattern.) Signal must be constant intensity unless Dose mode is used. **Though any number of reads can be entered, we don't believe more than two reads is ever useful and don't recommend a value greater than two**. The exposure Time (or Dose) before each read will be the value entered above, divided by the number of reads selected.

- **Fourth section (Protocol)**

Note: These instructions are for collecting one simple data set. To set up more complicated strategies and for explanations of advanced features, read the Data Collection – Advanced Features section of Chapter 3.

To collect just one data set, make sure the Segment 2 Axis selector is set to "STOP," and type the dataset protocol in Segment 1 (the top row of boxes) in this section.

- **Segment**: Name this Segment "1."
- **Binning**: Choose binning (this column absent if *marccd* is not configured to select binning). Lower binning yields higher resolution data frames.

Copyright © 2000-2006, Mar USA

- **Axis:** Choose “Phi” by clicking the button (if *marccd* doesn’t control the phi axis, or for a still exposure, choose “None”). Selecting “Skip” causes this row to be skipped. Selecting “STOP” stops the data collection at this row.
- **Motion**: Set to “Scan” for a scan motion, or “Still” for a still exposure. If *marccd* is configured to collect dark images, the option, “Dark” may be available, too.
- **Width:** In the case of a phi-scan, type in the oscillation angle in degrees, for example 0.1 for “phi-slicing” or 2.0 for “large oscillation images”. These numbers will depend on the crystal and the type of experiment. If not configured, this column may be absent. For a still shot, the values in this column don’t matter.
- **Time/Dose**: Choose “Time,” then type in the exposure time in seconds. Dose mode requires the software to be reading an ionization chamber (MarDTB or MarCSC).
- **Frm1**: Type in the number you wish to give the first frame of this data set, usually 1.
- **Next Frm**: Type in the frame number at which you wish to start this data set, usually 1.
- **Reset button**: Click this button to restart a dataset from the beginning, or recollect the same data on a different sample. This button sets all “NextFrm” values equal to their corresponding “Frm1” values. If dataset is stopped in the middle of a data collection (with Stop button), continue in the middle of the set by *not* touching this button.
- **Frm N**: Type in the total number of frames to collect.
- **Nsegs**: Type in 1. (This column is not always configured to be available. See the section Data Collection – Advanced Features in Chapter 3.)
- **Size:** If “Nsegs” is 1, then it does not matter what is given for “Size,” type in 1, for example. (Also not always available; see Data Collection – Advanced Features in Chapter 3.)
- **Phi**: (If configured for phi control) Type in the starting Phi position for the frame listed as “NextFrm.”
- **Distance**: Type in the desired sample-to-detector distance in mm.

For as many segments are desired, change Axis to a setting besides, “STOP,” and fill out the values above. The maximum number of possible segments available (rows in this dialog) is configurable. See the Appendix for more information about configuration.

Color indicators: each segment has a color for its row of boxes. The colors have the meaning as follows:

- **Light Blue**: Data will be collected.
- **Violet-Red**: Skip this segment.
- **Light Green**: Finished collecting this segment.
- **Goldenrod**: This segment is currently being collected.
- **Gray**: This segment will not be collected.

- **Acquire Dataset dialog command buttons (bottom row of dialog)**
  - **Start:** Starts the dataset with the requested values.
  - **Stop:** Stops the current dataset (only if a dataset is active).

Copyright © 2000-2006, Mar USA

- **Save:** Saves the current values in the Acquire Dataset dialog to come up the next time the window is opened.
- **Revert:** Reverts all values in the dialog to the most recently saved values
- **Close:** Closes the dialog window
- **Check:** Before starting a dataset, this button may be clicked as a sanity check to verify that all parameters are sensible and that there is space available in the save directory.

- **To start the data set**

  Once all the data is typed in correctly:
  ⇒ Click on the "Check" button in the bottom row. A dialog box will pop up and offer comments on the data collection setup. This is useful for spotting typos and/or mistakes.
  ⇒ If everything looks right, click on the "Start" button in the bottom row.

  The program will now start the data collection. If marccd controls motors such as distance and phi (for example, in the case of a software goniostat or with MarDTB or MarCSC), it will first drive to the starting motor positions, and then begin collecting data.

- **Using other *marccd* program functions during a dataset**

  The user may use other features of marccd during data acquisition. For example, it is normal to load data frames from disk and view them. The Zoom Window and Image toolkit is also available during a dataset.

  However, controls that will cause marccd to interact with the goniostat and detector are not available during a dataset. These include buttons to drive axes, change intensity range, and perform Beam Alignment or Crystal Alignment, and the controls in the Configure Detector menu. Though most off-limits functions should appear disabled (grayed out) during a dataset, the user should avoid them even if they are found to be available. Stop the dataset data collection in order to access those features.

- **To get rid of the data collection dialog box**

  ⇒ Click "Close" in the bottom row of buttons. Closing the dialog does not stop or affect the data collection.

  A question box will appear and ask whether to Save, Revert or Cancel. "Save" saves the current values in the dialog for the next time it is opened. "Revert" throws away changes, and "Cancel" keeps the dialog open.

- **To stop collecting the dataset at any point**

  ⇒ Click the red "Stop" button at the top of the *marccd* main window or in the "Acquire Data Set" dialog.

Copyright © 2000-2006, Mar USA

A question box will ask whether to abort the current frame now or wait until the frame is finished. Aborting in the middle of a frame is currently not available for the MarMosaic detectors, but it is available for MarCCD detectors running updated software. MarMosaic detector users must wait until the current frame is finished.

- **Disk space**

The *marccd* software will check for disk space and warn the user if there is not enough disk space for a data set about to be started. If the disk space runs out during the data collection (either because there was not enough space to begin with, or because other data was written to the same disk in the meantime), then *marccd* will pop up a dialog box with a warning and keep on trying to write the last collected frame. It will not go on to the next frame in the data set until it has successfully saved the previous data frame.

File sizes are discussed in the Appendix.

Copyright © 2000-2006, Mar USA

Chapter 3: Program Features of *marccd* in Detail

Note to one-time users of the detector: some features described here should not be used except by beamline staff responsible for the detector (e.g. changing parameters in Configure-Detector, Configure-Goniostat menus).

Starting the program

The program marccd can be run with a choice of command line options (explained below). Running with no options is the most common usage (start the program simply by opening a new terminal window and typing "`marccd <return>`").

First, the configuration files are read by the program. A separate *marccd* GUI opens, while in the original terminal window, log messages are continuously printed. The location of the log file is explained in the section below.

Command line options

The usage for command line options is as follows (here showing the explicit path for executing the program):

```
/home/marccd/bin/linux/marccd [X11 options] [-fhnPrtVv] [-C
directory] [-c filename] [-p filename] [-e name] [-D none] [-G
none]
```

See the table below for a complete reference of command line options. Option "-h" can be used to display these options and their usages.

Table 1 - Command line options

Option	Explanation
-C directory	Configuration directory. Default is $MARCCDCONF
-c filename	Configuration filename. Default is $MARCCDCONF/marccd.conf
-D type	Detector type (none = No detector).
-G type	Goniostat type (none = No goniostat).
-e name	Default experiment name for dataset.
-f	Force (kill old process, if any).
-h	Show help message (lists all command line options).
-n	Do not read ~/.marccdrc file.
-P	Proprietary mode (may require license key). (**CAUTION—additional features allow user to change detector parameters permanently**)
-p filename	Startup in dataset mode, reading protocol from file.
-r	Startup in remote mode (may require license key).
-t	Print timing statistics.

Copyright © 2000-2006, Mar USA

-v	Print version and exit.
-V	Verbose mode. (Each instance increases verbosity)

Menu Bar

Many control features are accessed via the menu bar (shown in Figure 7). The table below explains those features.

Figure 7 - Menu bar

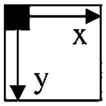
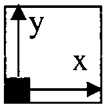
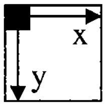
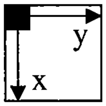
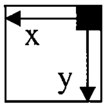
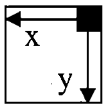
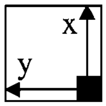
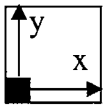
Table 2 - Menu bar description

Item	Option1	Option2	Explanation
File	Load X-ray Data Frame(s)		Opens Load Frame dialog. Use this to load frames into the Scratch buffers for inspection and analysis. Select an image with the mouse or type the path and name in and click "OK." Multiple frames can also be merged or averaged during loading by selecting "Multiple."
	Save Displayed Data Frame		Opens Save Frame dialog. Use this to save the frame buffer that is currently displayed. Type in the filename and click "OK." Save format will be the format selected under Options→ Output File Format.
	Save Displayed Image		(NOT CURRENTLY SUPPORTED) Saves the displayed image on the screen as an 8 bit TIFF, NOT a data image.
	Print		(NOT CURRENTLY SUPPORTED) Prints the image from a histogram graph to a postscript file.
	Quit		Exits the program. A dialog box will prompt for confirmation.
Edit	Copy to	Background	Copies the currently displayed frame to the Background buffer
		Scratch	Copies the currently displayed frame to the Scratch buffer
		Scratch1	Copies the currently displayed frame to the Scratch1 buffer
		Scratch2	Copies the currently displayed frame to the Scratch2 buffer
		Scratch3	Copies the currently displayed frame to the Scratch3 buffer
	Edit Frame Header	Edit Frame Title	Adds a frame title to the header before saving.
		Edit Frame File Comment	Adds a frame file comment to the header before saving.

Copyright © 2000-2006, Mar USA

		Edit Frame Dataset Comment	Adds a frame dataset commend to the header before saving.
	Rotate / Reflect Frame	Rotate Clockwise (+90)	Rotates the image displayed on the screen by 90 degrees clockwise.
		Rotate Counter Clockwise (-90)	Rotates the image displayed on the screen by 90 degrees counter-clockwise.
		Rotate 180	Rotates the image displayed on the screen by 180 degrees.
		Reflect in Horizontal Plane	Reflects the image displayed on the screen in the horizontal plane.
		Reflect in Vertical Plane	Reflects the image displayed on the screen in the vertical plane.
		Reflect in +45 degree Plane	Reflects the image displayed on the screen across the +45 degree diagonal plane.
		Reflect in -45 degree Plane	Reflects the image displayed on the screen across the -45 degree diagonal plane.
	Byte-swap Frame		Byte-swaps the displayed frame (little-endian to big-endian or back).
	Bin / Unbin Frame	Bin Frame	Bins the displayed frame. A text box appears. Type in an integer binning and click, “OK.” (*Note: Information in the displayed buffer is lost when binning a frame, even if it is subsequently unbinned.)*
		Unbin Frame	Unbins a binned frame (up to maximum resolution). No additional information is available; therefore, the added 4x4, 16x16, etc., pixels contain copies of the original pixel values.
View	Show Header		Displays the currently displayed image header. Raw (NxN) image formats contain no header information.
	Color Scheme	B/W Positive	NOT CURRENTLY SUPPORTED
		B/W Negative	Default color scheme. Higher intensity is darker on the screen.
		Color Spectrum	NOT CURRENTLY SUPPORTED
		Pseudocolor	NOT CURRENTLY SUPPORTED
		Egelman	NOT CURRENTLY SUPPORTED
		Caspar	NOT CURRENTLY SUPPORTED
	Scaling	Linear	Linear grayscale
		Log	Log grayscale. A dialog box prompts for a gamma (γ) value to be entered, where gamma is used as follows: $color = \frac{N(\log\gamma(X-\min)/(\max-\min))}{\log\gamma}$ Here, there are N total shades (65536 in a two-byte depth image), X is the value of the pixel being evaluated, and max and min are the max and min shade values in this frame.
	View Direction	From Source	(Default) Displays image as viewed from the source

Copyright © 2000-2006, Mar USA

		Toward Source	Displays image as viewed toward the source.		
	Coordinates	Native		Origin ≡(0,0)	Changes the coordinate system used when viewing images. When pointer is above a pixel on the screen, coordinates are printed below the zoom window.
		automar		Origin ≡(0,0)	
		MOSFLM		Origin ≡(1,1)	
		DENZO		Origin ≡(0,0)	
		BUDDHA		Origin ≡(1,1)	
		XENGEN		Origin ≡(0,0)	
		Xentronics PCS		Origin ≡(0,0)	
		Siemens PC		Origin ≡(0,0)	
Configure	Detector		Opens the Configure Detector Dialog for controlling detector cooler power, etc. See complete explanation in the section below.		
	Goniostat		Opens the Configure Goniostat Dialog for displaying and changing information the software knows about the goniostat. See complete explanation in the section below.		
Acquire	Single Frame		Opens the Acquire Single Frame Dialog, as explained in Chapter 2.		
	Dataset		Opens the Acquire Dataset Dialog, as explained in Chapter 2.		
	Remote Control		(Requires license key—not shown without valid license key) Opens the Remote Control dialog box		
Options	Output File Format	Native (16 bit TIFF)	(Recommended) Native *marccd* image format is two-byte TIFF, with header format as explained in Appendix.		These options change the output format for all files (including single

Copyright © 2000-2006, Mar USA

		Raw (NxN)	Two-byte NxN raw data, no header.	frame and dataset acquisition, and manually saving with File→ Save Displayed Data Frame).
		Mar 180/300 (NxN)	Mar 180/300 format. The only difference is that the number of pixels corresponds to the actual *marccd* frame pixels.	
		FITS	Flexible Image Transport System, common for astronomical images and analysis.	
		Xentronics/Cadmus	Xentronics multiwire detector image format	
		Xentronics/PCS	Xentronics multiwire detector image format	
		Scripps/Filman	Scripps Filman detector image format	
		Siemens/PC	Siemens/PC image format	
		Photometrics	Photometrics image format	
		MSC/RAXIS-II	RAXIS-II image format	
		Fuji/Photon Factory	Fuji image format	
		IPLab	IPLab image format	
		Fischer/Xedar	Fischer/Xedar image format	
		MarResearch	MarResearch image format	
		CHESS/TV6	CHESS/TV6 image format	
		PI/CSMA	PI/CSMA image format	
	Save Preferences		NOT CURRENTLY SUPPORTED	
Help	About		Shows detailed information about the *marccd* executable file being used (author, version, build, etc.)	
	Help		Shows basic help information	
	Report a bug		Opens an internet browser and directs it to the Mar USA Bug Server (http://www.mar-usa.com/bugs). Create an account and log in to report a bug. Include as much detail about the bug as possible.	

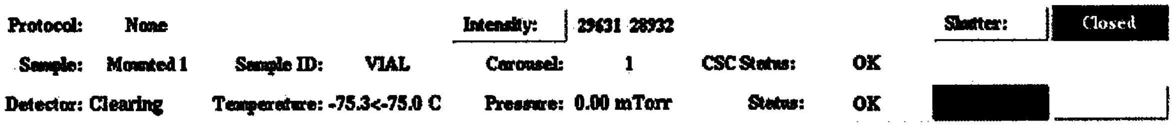
Quitting the program

⇒ Click File→Quit, on the menu bar. A dialog box will prompt for confirmation.

Alternatively, point the mouse at the terminal window where *marccd* was started, and hit Ctrl+C to quit the program.

Copyright © 2000-2006, Mar USA

Status Window



Protocol: None
Intensity: 29631 28932
Shutter: Closed
Sample: Mounted 1
Sample ID: VIAL
Carousel: 1
CSC Status: OK
Detector: Clearing
Temperature: -75.3<-75.0 C
Pressure: 0.00 mTorr
Status: OK

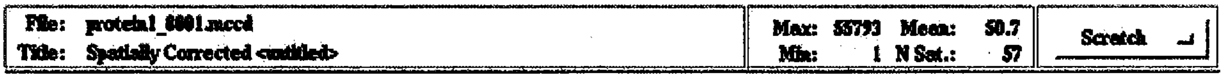
Figure 8 - Status window (shown here for the case that a MarCSC and MarMosaic are present)

The Status Window (shown in Figure 8), gives current information about the detector (and MarDTB and MarCSC sample changer, if available), updated once per second or a few times per second. It contains the following sections:

- **Protocol**: If a protocol file is currently being used for collecting a dataset (see the Dataset Protocols section below), the title is displayed here.
- **Detector**: Current detector activity is shown here. During idle times, detector is always in the Clearing state. Other states are Integrate and Reading.
- **Temperature**: Detector temperature measured at the CCD. Typical operating temperatures are -70 C to -80 C. Detectors with multiple CCD chips (MarMosaic) show a shorthand temperature range for all the CCD chips (e.g. "Temperature: -75.3<-75.0 C" means that all CCD chips have -75.3 C < T < -75.0 C).
- **Pressure**: Pressure inside the vacuum chamber. To maintain such low temperatures, the CCD is kept inside a vacuum environment.
- **Status**: Current known detector or program status is shown here.
- **Shutter**: Shutter state is shown here, and can be controlled manually by pressing the shutter button (only if *marccd* controls the shutter).
- **Intensity**: If *marccd* is configured to get the ionization chambers state (in the case of MarDTB and MarCSC), updated intensity information for the first and second chambers is shown here.
- **Intensity button**: This is used to set the gain of the ionization chambers. Available settings are HIGH (for high intensity region, such as at synchrotrons) and LOW (for low intensity region, such as rotating anodes; if signal is saturating at LOW, try setting to HIGH region). *(VERY IMPORTANT: the gain self-adjusts, so there must be NO X-ray signal going into the MarDTB when changing the gain. The data collection shutter inside the MarDTB is* after *the ionization chambers; therefore a shutter upstream from the instrument must be closed. If gain is set accidentally with X-ray signal, it must be set again after closing an upstream shutter.)*
- MarCSC Information, if available, is shown in the following fields:
    - **Sample**: The sample currently mounted and its carousel position
    - **Sample ID**: The barcode reader's last recognized barcode or VIAL for empty vial
    - **Carousel**: The current carousel motor position under the picker mechanism (not necessarily equal to the position of the mounted sample—although the program will remember the correct position to which the sample should be returned)
    - **CSC Status**: The sample changer status
- **Stop button**: The red Stop button is used to stop any current data acquisition (Single Frame or Dataset). There are also stop buttons with the same functionality in the respective data acquisition menus.

Copyright © 2000-2006, Mar USA

Frame title bar and buffer display pop-up



File: protein1_0001.mccd
Title: Spatially Corrected <untitled>
Max: 55793 Mean: 50.7
Min: 1 N Sat.: 57

Figure 9 - Frame title bar

Information about the currently displayed frame buffer is shown here. Next to “File” the filename is shown, and next to “Title” the frame title is shown. To the right of the title bar is some basic numerical information about the image currently displayed:

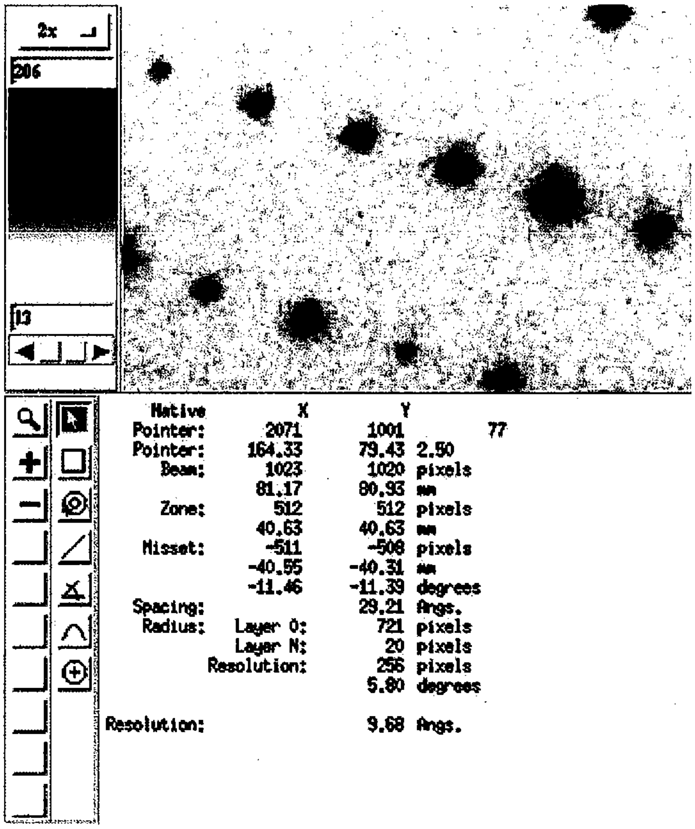
- Max: the maximum intensity value in the frame (excluding saturated pixels)
- Min: the minimum intensity value in the frame
- Mean: the average intensity value (excluding saturated pixels)
- N Sat: the number of saturated pixels (for two-byte depth frames, the value is 65535)

The pop-up button to the right of the Frame Title Bar is used to switch to a different display buffer. Click the button, drag to the desired buffer name, and release. If no frame is currently loaded in the buffer, the *marccd* logo will be displayed.

Inspecting images

Images can be inspected with several zoom, contrast, and numerical tools as shown in Figure 10. These tools are found in the top right of the *marccd* main window, or in some configurations, they may be found on a tab in the top right called “X-ray Data Detail.”

Copyright © 2000-2006, Mar USA



```
     Native          X          Y
   Pointer:       2071       1001           77
   Pointer:     164.33      79.43  2.50
      Beam:       1023       1020  pixels
                 81.17      80.93  mm
      Zone:        512        512  pixels
                 40.63      40.63  mm
    Misset:       -511       -508  pixels
                -40.55     -40.31  mm
                -11.46     -11.39  degrees
   Spacing:                 29.21  Angs.
    Radius:   Layer 0:        721  pixels
              Layer N:         20  pixels
           Resolution:        256  pixels
                             5.80  degrees

Resolution:                  9.68  Angs.
```

Figure 10 - X-ray Data Detail tab and tools

- **Zoom Window**

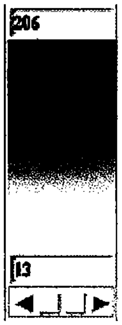
An area from the large display window is shown enlarged in this field. The magnified area within the data frame is chosen by clicking the left mouse button in the desired location in the main frame display window.

  - **Magnification pop-up menu**

2x

This pop-up menu to the top left of the zoom display allows the user to choose the magnification. When it is set to 32, the actual integer values stored for each pixel will be displayed in the zoom window.

- **Contrast control**



All settings in this area apply to both the main display and the zoom display. On top of the box showing the color spectrum in use is a dialog box containing a number. This is the integer number stored for each pixel. Any pixel with this value or higher will be displayed in the color at the top of the display spectrum. On the bottom of the color spectrum is another dialog box. Any pixel with a value equal or less to what is in this box will be displayed in the color at the bottom edge of the color spectrum. To enter a value into

Copyright © 2000-2006, Mar USA

either box by hand, use the mouse to click in the box, type in the desired value, and then hit return.

In the example shown, the top box reads 206, and the color at top edge of spectrum is black; therefore all pixels with a value of 206 or greater will look black in the display (both the large display and the zoom window). The bottom dialog box reads 13, and the bottom edge of the spectrum is white; therefore all pixels with values of 13 or less will be displayed in white. Pixels are shaded in between, and contrast may be either linear or log (changed from View option on the menu bar).

Also below the display color spectrum is another box with two small arrow buttons. If the arrow button pointing left (toward the large data frame display window) is clicked, the contrast is automatically adjusted to display the entire frame. If the arrow button pointing right (to the zoom display) is clicked, the contrast is automatically adjusted for viewing the zoom box.

The color spectrum itself can be adjusted via the mouse by positioning the mouse in the spectrum, holding the left button down and then moving the mouse. If the mouse is moved vertically the brightness is adjusted. By moving the mouse horizontally the contrast is adjusted.

- **Image toolkit**

  - **Show zoom window**

    If this button is clicked on, the area displayed in the zoom window will be outlined by a box in the main display window. As the zoom area is re-located by clicking with the mouse, this box is moved accordingly. The box can also be dragged (with the right mouse button) to reposition the zoomed area. The zoom display will update when the mouse button is released.

  - **Forward**

    Click on this button to automatically load the next frame of a data set into the current display. An error message is displayed if such a frame cannot be found. The filename is determined from the name of the displayed frame. The frame number is assumed to be the rightmost numerical part of the name.

  - **Backward**

    Click on this button to automatically load the previous frame of a data set into the current display. An error message is displayed if such a frame cannot be found. The filename is determined from the name of the displayed frame. The frame number is assumed to be the rightmost numerical part of the name.

  - **Pointer**

    When this button is clicked all other tools are turned off. The Pointer must be selected in order to view the Goniostat menu.

Copyright © 2000-2006, Mar USA

- **Box**

  This tool is used to display a histogram and statistics about the pixels in a region of interest. When this button is clicked, the user can draw a box anywhere in the main display window or Zoom Window. Statistics about the pixels in the box are displayed in the table to the right of the tool buttons.

  To draw the box, hold down the middle mouse button and drag. To move the box, position the mouse in the center of the box and drag the box by holding down the right mouse button while moving the mouse. To change the size of the box, position the mouse on an edge or corner of the box, hold down the right mouse button and drag.

  A histogram is displayed below the image toolkit, and statistics are printed to under the zoom window.

- **Layer lines**

  This tool provides a way to know the angle between a crystal lattice layer and the X-ray beam, using a diffraction pattern. It uses two rings, defining spots in a zero layer and an Nth layer, and a cross marking the beam position.

  The zero-layer ring will always go through the beam position. To move the beam position, you can right-click the cross (but defining the beam position is easier done by switching to the Resolution Ring tool, centering, and switching back to this tool).

  Place the zero-layer ring through a desired "loon" of spots by dragging the circle with the right mouse button. Then position the other ring (the Nth ring) along the desired ring of spots. The angle of the lattice layer appears printed to the right of the toolkit, under the heading "Misset."

- **Line**

  This tool allows the user to draw a line in the display and then inspect a plot of the intensities along this line. The plot will appear under the toolkit.

  Lines may be changed after they are drawn. Point to the approximate middle of the line and right click to drag the line to a different location. Point to one of the endpoints of the line, and right click to drag the single endpoint to a different location.

  Click a second time on the Line tool button to be asked to enter a width of pixels for averaging. If a number greater than 1 is entered, a wide strip of pixels (centered about the line drawn on the screen) is averaged and the plot of intensities is displayed.

- **Angle**

Copyright © 2000-2006, Mar USA

This tool measures the angle between two lines. To draw the angle, use the middle mouse button to draw a line. A short horizontal line will automatically appear attached to the user-drawn line. Use the right mouse button to drag the end of the horizontal line to the desired position. The table to the right of the toolkit buttons will now display the angle between the lines.

Spot profile

This tool allows the user to examine the profile of a spot in the phi direction. When this button is clicked, a dialog box appears. The user is asked to type in the number of frames to be used, with the currently displayed frame at the center of the stack. Note that the currently displayed data frame will be lost unless it is already saved to disk. Once "OK" is clicked, the frames surrounding the currently displayed frame will be automatically loaded into the program memory. A small box can be positioned in the display via the middle mouse button. This is best done in the zoom window. The right mouse button can grab the edges of the box and change the box size. The middle mouse button will move the center of the box to wherever it is clicked in the display. The area within the small box is used for the plot displayed under the table and tool buttons.

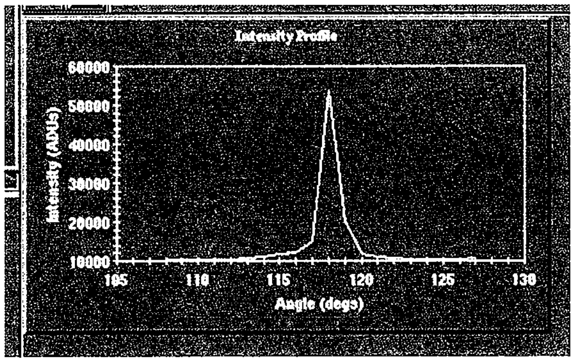


Figure 11 - Spot profile graph window

Figure 11 shows an example plot in which the intensity within the box varied from frame to frame. This tool would be used to examine spot profiles in order to decide on the oscillation range for phi when a data collection is being set up.

Resolution ring

Note: the distance and wavelength must be recorded correctly in the image header for the Resolution Ring to function.

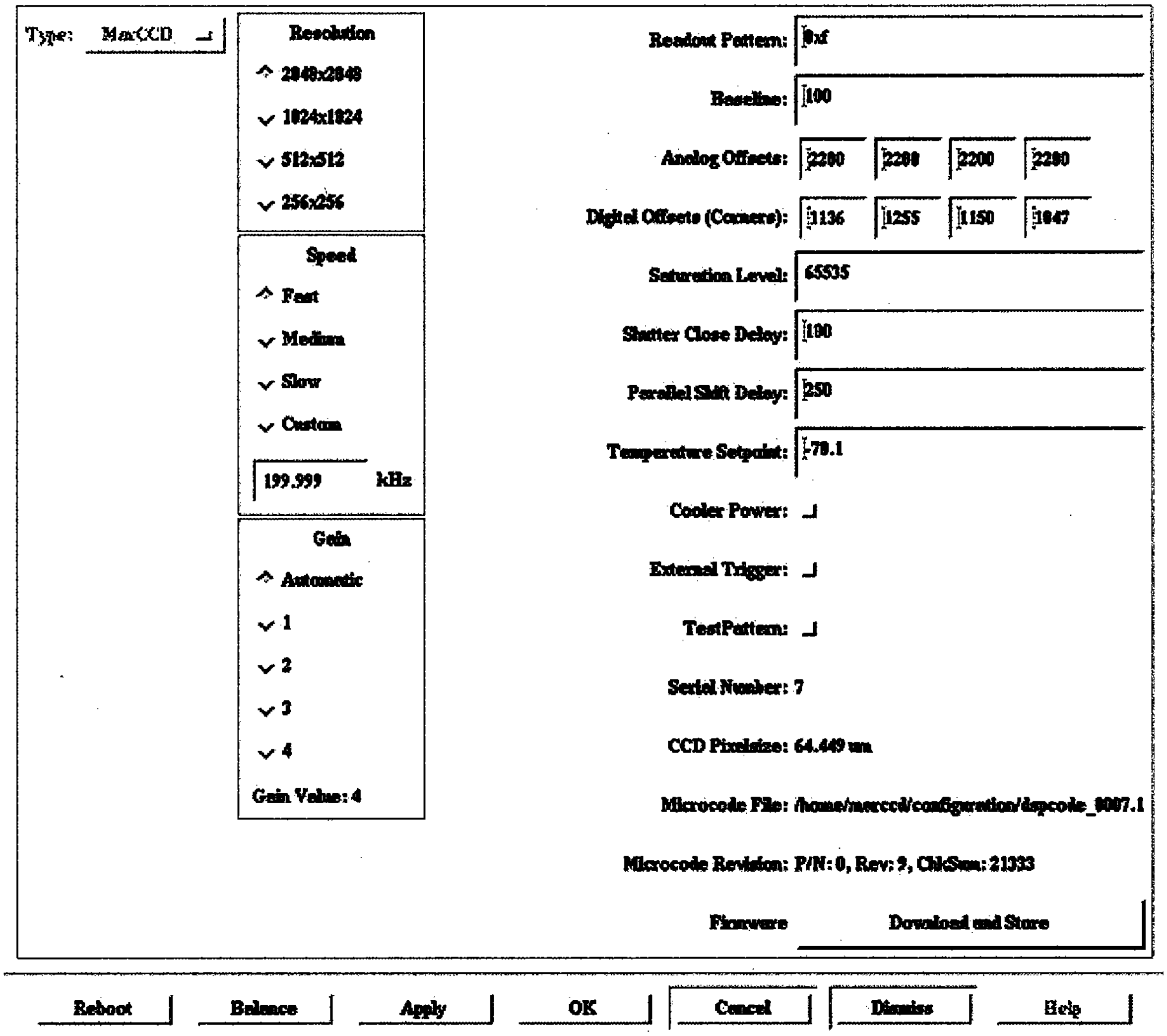
This tool allows the user to 1) define the beam position, and 2) display a resolution ring. Define the beam position by dragging the center of the circle with the right mouse button. The new beam position will be shown in the table to the right of the tool buttons. Right-click on the ring edge and drag the ring to the desired size. The resolution is printed in the window to the right of the toolkit buttons.

Configure Detector Dialog

This dialog looks different for MarCCD and MarMosaic detectors; therefore, each is described separately below.

- **MarCCD Configure Detector dialog**

Open this dialog by clicking Configure→Detector in the menu bar.



199.999 kHz

Gain Value: 4

Readout Pattern: 0xf

Baseline: 100

Analog Offsets: 2200 2200 2200 2200

Digital Offsets (Corners): 1136 1255 1150 1047

Saturation Level: 65535

Shutter Close Delay: 100

Parallel Shift Delay: 250

Temperature Setpoint: -70.1

Serial Number: 7

CCD Pixelsize: 64.449 um

Microcode File: /home/marccd/configuration/dspcode_0007.1

Microcode Revision: P/N: 0, Rev: 9, ChkSum: 21333

Figure 12 - MarCCD Configure Detector dialog

<u>Parameters on the left half of the dialog:</u>

- **Resolution**
  Normally the MarCCD detector is read out at highest resolution (2048x2048 pixels). The user may elect lower resolution, in which case the pixels on the CCD chip will be binned into groups of 4 (or 16). This would speed up the read out, reduce the size of

the files containing the frames, and increase the dynamic range (because there are now 4 (or 16) wells on the CCD chip contributing to each pixel in the finished data frame), but each frame would have only 1024x1024 (or 512x512) pixels for the same size detecting surface. The corresponding change of pixel size is reflected on the lower right half of the Configure/Detector panel as soon as "Apply" is clicked. Highest resolution (unbinned) is recommended (less than 5 sec readout time, very low readout noise), thus 2048x2048 is the default.

- **Speed**
  This is preset to the fastest Speed possible and cannot be changed by the user. It is set to the default value: Fast.

- **Gain**
  In most cases the Gain is set to Automatic, in any case this is preset to the optimal value and cannot be changed by the user. It is set to the default value: Automatic.

Parameters on the right half of the dialog:

These parameters all pertain to the treatment of the signal delivered by the chip.

Obviously, once the system has been calibrated and installed, it is important that the hardware configuration is not changed. For example, if the system was calibrated with channel A from the detector going through the A/D converter A, channel B going through A/D converter B etc.(as determined by the cables going from the detector head to the controller), then the detector must always be set up this way for use as well. Likewise the A/D cards in the controller may not be switched or the calibration will no longer be optimal.

- **Readout Pattern**
  This refers to which channels should be used to read out the chip. For the 4-channel CCDs this should be set to 0xf, causing the system to use all four channels for maximum readout speed (the option to read out on less than four channels is disabled in the distributed software). The default for four-channel detectors is 0xf, for others it varies.

  The prefix 0x indicates that the number which follows is hexadecimal. However, the marccd software currently always interprets the following number as hexadecimal, regardless of the prefix. To avoid confusion the prefix 0x should be used at all times.

  The locations of the four readout channels of the CCD chip as seen in images are shown in Figure 13:

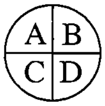


Figure 13 - MarCCD readout channel definitions (as viewed in standard direction—From Source)

Copyright © 2000-2006, Mar USA

Table 3 - MarCCD readout channels

Channel	Quadrant of image (standard view: From Source)	Bit	Value
A	top left	0	1
B	top right	1	2
C	bottom left	2	4
D	bottom right	3	8

The hexadecimal number describing the readout pattern is the sum of numbers in the Value column of the table above, corresponding to the channels to be used. It should not be changed from factory the default (except for overcoming service problems, as explained in the service manual). Non-standard values may require cabling changes to work as expected.

See the table below for examples of how hexadecimal numbers are calculated for readout configurations.

Table 4 - MarCCD readout channel configurations

Example readout configuration	Calculation	Hexadecimal equivalent
Four channels (A, B, C, D)	1+2+4+8=15	15=0xf
Two channels (A and B only)	1+2=3	3=0x3
Two channels (C and D only)	4+8=12	12=0xc

Diagonal combinations (i.e. B and C; A and D), or combinations of three channels are not permitted.

- **Baseline**
  This number is also read in from the default parameter file. It is usually set to 100.0 and is added to each pixel after the digital offset is subtracted. This value should not be changed, the default is 100.

- **Analog Offset**
  These values should be read from the default parameter file and not be changed by hand. Default values vary; typically all four A/D converters must have the same value.

  The leftmost number refers to the A/D converter card furthest away from the controller board, the rightmost number to the A/D converter closest to the controller board. This is independent of how the jumpers are set on the A/D boards. The numbers set here should bring the lowest signal to the lowest end of the voltage range of the A/D converter, i.e. these values can have an effect on the dynamic range.

- **Digital Offsets (Corners)**
  Also factory preset and should always be read from the defaults file. The four numbers refer to the four corners of the CCD chip: first number corresponds to top

left, second number to top right, third number to bottom left, fourth number to bottom right. Normal values are 300 to 600. The four numbers may differ from each other. These numbers may change if a "balance" is performed. To get back to the default numbers, reboot the detector controller and load the defaults.

- **Saturation Level**
  All pixels with values greater than this number will be flagged as saturated in the final image (i.e. set to 65535). This number is usually 65534 or 65535, depending on the type of AD converter used.

- **Shutter Close Delay**
  This value is set in the configuration file. It is a delay 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 beam time is wasted by having inserted an unnecessary delay.

- **Parallel Shift Delay**
  This value is set in the configuration file. It is determined by Mar USA during calibration and generally should not be changed.

- **Cooler Power checkbox**
  Click this button to toggle the cooler on/off, and then click the "Apply" button to execute. The cooler should be "ON" for normal operation.

- **External Trigger checkbox (not available in standard *marccd* version)**
  If this button is checked, the detector will only be read out if an external signal is received on the aux. input of the CCD controller (black box in the cooler cabinet). The expected signal is usually set to TTL, i.e. the rising edge of a positive 5V pulse triggers a detector read-out. A set of jumpers internal to the CCD controller allows this to be set to OPTICALLY ISOLATED OPEN COLLECTOR INPUT. The default is OFF. In most cases this button should be off, i.e. the detector should be read under software control.

- **Test Pattern checkbox (not available in standard *marccd* version)**
  If this button is on, then a test pattern will be generated whenever the detector is read. The test pattern contains all values required to check that no bits are missing in the images. The default is OFF.

- **Serial Number**
  This number must agree with the detector serial number.

- **CCD Pixel Size**

Copyright © 2000-2006, Mar USA

The exact value varies between individual detectors. The pixel size will change if one chooses a different resolution and then click either on "Apply" or "OK". The change is due to the binning of physical pixels into larger pixels for lower resolution.

- **Configuration File**
  This file must exist and agree with the serial number of the detector. It contains the default parameters needed to run each detector properly.

- **Microcode File**
  This code may be customized for each detector and the serial number must match the detector serial number. The file must exist.

- **Microcode Revision**
  This number is currently a composite of a part number and a revision number. It is used to check whether the controller is currently running the same microcode version that marccd is expecting.

Buttons on the bottom of the dialog:

- **Reboot button**
  When this button is clicked, the system will reboot the detector controller. In older versions of marccd a box appears, offering the choice to either reload the current parameters or to load default parameters upon rebooting. These parameters are customized for each detector. Given a choice, the user should choose to load the default parameters.

- **Balance button**
  The Balance operation adjusts the baseline level of raw and background images by recalculating the Digital Offsets. This can be used to make the four quadrants of background images about the same level and therefore easily inspected simultaneously (which is not often necessary). Very rarely, it is used to solve ADC baseline shifts that are causing dead areas of zero intensity in one or more quadrants—see the Troubleshooting section if this problem is occurring.

  When the Balance button is clicked the detector chip is read out twice (to remove zingers) and proper digital offsets for each A/D converter are calculated to produce the same final intensity range on all 4 channels. The 4 channels are then read out a third time to produce an image that demonstrates the effect of the new offsets visually. Balance does not change the gain of the four quadrants.

  Note: a common misunderstanding is that the Balance operation solves problems of signal differences among the four channels appearing in normal images (i.e. corrected images) taken by the MarCCD detector. Balancing does ***not*** *solve such differences, because background frames are always subtracted from corrected images. Differences could occur if an image was acquired with an old (stale) background image. The ADCs drift gradually, so periodically the background must be recollected.*

Copyright © 2000-2006, Mar USA

If using the Acquire Single Frame dialog to collect data, try collecting a new image after checking the checkbox to force a new background to be collected. If quadrant differences persist, see the Troubleshooting section in the Appendix.

- **Apply button**
  When this button is clicked the currently showing data in the window is read and/or acted upon. The user should click this button or "OK" after setting "Cooler Power On." This causes the program to actually turn the cooler power on. Likewise, if the Resolution is to be changed, one must first choose the resolution and then click this button or the "OK" button for the change to take effect.

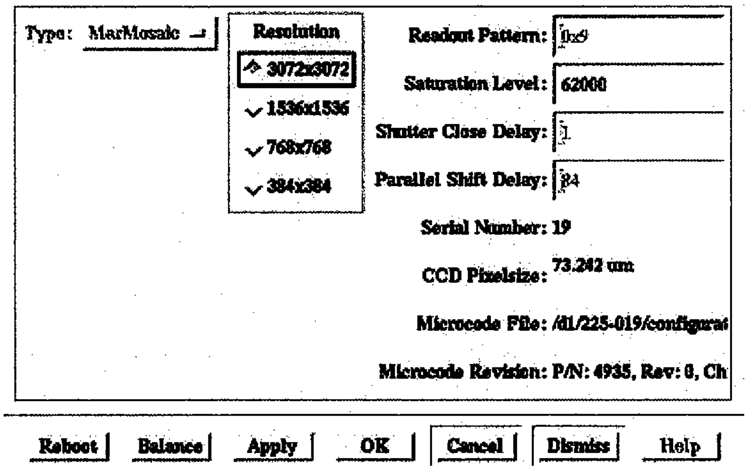
- **OK button**
  When this button is clicked the currently showing data in this window is read and/or acted upon and the window closes. It does the same as clicking the "Apply" button, followed by the "Dismiss" button.

- **Cancel button**
  When this button is clicked the window will close without accepting any changes made after the last "Apply" click. If the window is subsequently re-opened it will not show any changes made after the last "OK" or "Apply" click.

- **Dismiss button**
  When this button is clicked the window will close without storing or acting upon any new data entered after the last "Apply" click. However, if the window is subsequently re-opened the new data will still be visible as typed in.

- **MarMosaic Configure Detector dialog**

Open this dialog by clicking Configure→Detector in the menu bar.



Type: MarMosaic
Readout Pattern: 1x9
Saturation Level: 62000
Shutter Close Delay: 1
Parallel Shift Delay: 84
Serial Number: 19
CCD Pixelsize: 73.242 um
Microcode File: /d1/225-019/configurat
Microcode Revision: P/N: 4935, Rev: 0, Ch

Figure 14 – Configure Detector dialog for MarMosaic

<u>**Parameters in the main part of the dialog:**</u>

Copyright © 2000-2006, Mar USA

- **Resolution**

  Normally the MarMosaic detector is read out at highest resolution (3072x3072 pixels for 3x3 detectors, and 4096x4096 for 4x4 detectors). The user may elect lower resolution, in which case the pixels on the CCD chip will be binned into groups of 4 (or 16). This would speed up the read out, reduce the size of the files containing the frames, and increase the dynamic range (because there are now 4 (or 16) wells on the CCD chip contributing to each pixel in the finished data frame), but each frame would have lower resolution. The corresponding change of pixel size is reflected on the lower right half of the Configure/Detector panel as soon as "Apply" is clicked. Highest resolution is the default and is recommended (1 sec hardware readout time, very low readout noise).

- **Readout Pattern**

  This refers to which channels should be used to read out each CCD chip (all the chips in the MarMosaic are read simultaneously with the same pattern).

  The four readout channels of the CCD chip are arranged physically as shown in Figure 15:

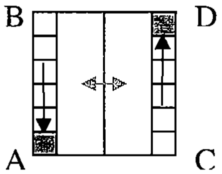


Figure 15 - Readout channel of a CCD chip on a MarMosaic (as viewed in standard direction—From Source). Black arrows indicate readout of pixels in the Fast direction (*note that this is different than the Fast direction of the MarCCD, which is horizontal*). Gray arrows show the Slow direction of line shifts during readout. Gray pixels shown are the pixels at the readout amplifier (when in 0x9 configuration).

The hexadecimal number describing the readout pattern is the sum of the numbers in the Value column in the table below, corresponding to the channels to be used.

Table 5 - MarMosaic readout channels

Channel	Bit	Value
A	0	1
B	1	2
C	2	4
D	3	8

The standard readout configuration is 0x9 (the prefix 0x indicates that the number which follows is hexadecimal), because channels A and D are used. The hex value, 0x9, is calculated according to the table above, so that A and D add up to 1+8=9. The readout pattern should not be changed from the factory default.

- **Saturation Level**

Copyright © 2000-2006, Mar USA

All pixels with values greater than this number will be flagged as saturated in the final image (i.e. set to 65535). This number is usually 62000 for the MarMosaic detectors, depending on the type of AD converter used.

- **Shutter Close Delay**
  This value is set in the configuration file. It is a delay 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 beam time is wasted by having inserted an unnecessary delay.

- **Parallel Shift Delay**
  This value is set in the configuration file. It is determined by Mar USA during calibration and generally should not be changed.

- **Cooler Power checkbox**
  Click this button to toggle the cooler on/off, and then click the "Apply" button to execute. The default is "OFF" but the cooler should be "ON" for normal operation.

- **Serial Number**
  This number must agree with the detector serial number.

- **CCD Pixel Size**
  The exact value varies between individual detectors. The pixel size will change if one chooses a different resolution and then click either on "Apply" or "OK". The change is due to the binning of physical pixels into larger pixels for lower resolution.

- **Configuration File**
  This file must exist and agree with the serial number of the detector. It contains the default parameters needed to run each detector properly.

- **Microcode File**
  This code may be customized for each detector and the serial number must match the detector serial number. The file must exist.

- **Microcode Revision**
  This number is currently a composite of a part number and a revision number. It is used to check whether the controller is currently running the same microcode version that marccd is expecting.

Buttons on the bottom of the dialog:

- **Reboot button**
  When this button is clicked, the system will reboot the detector controller. In older versions of marccd a box appears, offering the choice to either reload the current

parameters or to load default parameters upon rebooting. These parameters are customized for each detector. Given a choice, the user should choose to load the default parameters.

- **Balance button**
  This button has no functionality for the MarMosaic and never needs to be clicked.
- **Apply button**
  When this button is clicked the currently showing data in the window is read and/or acted upon. The user should click this button or "OK" after setting "Cooler Power On." This causes the program to actually turn the cooler power on. Likewise, if the Resolution is to be changed one must first choose the resolution and then click this button or the "OK" button for the change to take effect.
- **OK button**
  When this button is clicked the currently showing data in this window is read and/or acted upon and the window closes. It does the same as clicking the "Apply" button, followed by the "Dismiss" button.
- **Cancel button**
  When this button is clicked the window will close without accepting any changes made after the last "Apply" click. If the window is subsequently re-opened it will not show any changes made after the last "OK" or "Apply" click.
- **Dismiss button**
  When this button is clicked the window will close without storing or acting upon any new data entered after the last "Apply" click. However, if the window is subsequently re-opened the new data will still be visible as typed in.

Configure Goniostat Dialog

Open this dialog by clicking Configure→Goniostat in the menu bar.

The Configure Goniostat dialog was previously used to set most goniostat parameters, but is now obsolete. Currently, these parameters are normally only changed by changing the values in configuration files. Nevertheless, this dialog can still be viewed to check which axes are present and what degree of control the *marccd* software has over them. Limits and speeds are also shown here for each axis. (Although fields are filled in, they may not be meaningful, such as motor speed for changing Wavelength in Angstrom/sec).

For each axis, a set of six checkboxes shows the degree of software control. The checkboxes are under the heading, "Pr/Mv/Mt; Sc/XC/XF," and are defined as follows:

- **Pr**: Present. Axis is present.
- **Mv**: Movable. Axis is movable.
- **Mt**: Motor control. Axis is controlled by *marccd.*

- **Sc**: Scannable. Axis can be scanned, for example, by the Beam Alignment function.
- **XC**: External Control. The software issues commands to control an external device.
- **XF**: External Feedback. The software receives confirmation from the externally controlled axis about its position. (Currently, External Feedback is not supported, so even though move commands are issued, *marccd* does not know the position of the external motor.)

Click Dismiss to close this dialog, and then in the question box, click Abandon to throw away any changes made.

Data Collection – Advanced Features

- **Interleaved segments**

Note: By default, interleaved segments are not available. To check whether the current marccd installation is configured for interleaved segments, open the Acquire Dataset dialog and check if there is an "Nsegs" column available. If not, a line "`dataset_interleaved_segments yes`" keyword must be added to the configuration by the administrators of the software installation (see Appendix for Configuration Guide).

Data collection with interleaved segments allows more sophisticated protocols to be used. Segments are collected in groups that alternate frame-by-frame. One could collect images that alternate between two different distances or wavelengths, or other procedures.

To tie a block of a certain number of segments together, set the value of **Nsegs** in the Acquire Dataset dialog to the number of blocks to be tied (e.g. greater than 1). The parameter **Size** for each segment determines how many frames would be collected before switching to the next tied segment in the block. All segments in a block get the same **Size** number as the first segment in the block. After **Size** number of frames is collected for all the tied segments, the first tied segment is returned to. The block is finished after **FrmN** frames are collected for all the segments in the block.

As an example of that kind of experiment, please see the table below. Other parameters may be available (depending on configuration), such as selecting "phi" as the scan axis.

Table 6 - Interleaved segments example

Segment	Width	Time	Frm 1	NextFrm.	FrmN	Nsegs	Size	Distance
01	1.0	1.0	1	1	15	2	5	100.0
02	1.0	1.0	1	1	15	1	1	150.0
03	1.0	1.0	1	1	10	1	1	50.0

In this case, the following would occur:

- Segment 01: 5 frames at distance 100.0 mm are collected
- Segment 02: 5 frames at distance 150.0 mm are collected

Copyright © 2000-2006, Mar USA

- Segment 01: 5 frames at distance 100.0 mm are collected
- Segment 02: 5 frames at distance 150.0 mm are collected
- Segment 01: 5 frames at distance 100.0 mm are collected
- Segment 02: 5 frames at distance 150.0 mm are collected
- Segment 03: 10 frames at distance 50.0 mm are collected

Nsegs is 2 for segment 01, therefore segments 01 and 02 form a block. Size of 5 in segment 01 determines that 5 frames are collected before switching to the next tied segment, which also will have Size 5 (all tied segments have the same "Size" parameter as the first, no matter what is entered in the column). Finally, because 15 frames are finished from segments 01 and 02, all of segment 03 is collected.

- **Periodic Command(s)**

Periodic commands are commands that are executed between frames, while a data collection is taking place. They can be anything from simple commands or sophisticated scripts. The conditions for executing a periodic command can be every certain number of frames, segments, or fixed amount of time (although the test for whether the time has expired happens only between exposures). The program may be configured to use multiple periodic commands, using the `dataset_max_periodic_commands` parameter (see the Configuration Guide in the Appendix).

As an example, we can enter the Linux command "sleep 5" as a periodic command, check the Periodic Command checkbox, and change the options to "After" every "2" "frames." Thus, during data collection, after every 2 frames, the program will sleep for five seconds (the sleep command in this case freezes the display; a more sophisticated script should be used to make the program delay without freezing the user interface).

Periodic commands can be use both in the Acquire Dataset dialog, and in Dataset Protocol files as well (using the periodic start, start phase, command, and interval parameters explained in the protocols section below).

- **Archive Command**

An archive command is program or script that is run after an image is written to disk to move or treat files in some way, and the latest image filename is passed as an argument to the program. Archive commands are available in both the Acquire Single Frame and Acquire Dataset dialogs, as well as dataset protocol files (by using the `archive_command` parameter). Here is an example: the Linux command "rm" is entered as the Archive Command (and the checkbox to use Archive Command is checked). An image is collected with filename "test.0001." The file, after it is written to disk once, will immediately be removed from the disk.

- **Dataset Protocols**

Protocol files are text files that can be used to easily collect data sets many times without having to type the parameters into the Acquire Dataset dialog each time. There are two ways to use them:

1) Loading a protocol file by hand in the dataset dialog in marccd. To do this, click the "Load" button in the Acquire Dataset dialog, select the file, and then click "OK."
2) Starting *marccd* with the command line -p option (e.g. "`marccd -p protocols/my_protocol_file`").

The “autostart” parameter is used to automatically start data collection, and “autoquit” is used to automatically quit the program after the protocol is finished, for automated operation. Thus a protocol file actually allows a data set to be collected from the command line.

Example protocol files, which can be modified as needed, are normally found in the directory ~/protocols/ (if not found, they may be obtained by contacting Mar USA).

The table below lists all the dataset protocols parameters recognized.

Notes:

All multiple word strings should be in quotes.
Keywords are not case sensitive.
Use # to comment out a line or add comments to protocol file.

Table 7 - Dataset protocol parameters

Parameter	Type	Default	Example
	Explanation		
filename_template	CSTRING	%%.####	test_%%%%.####
	File name template; “%%%%” is segment number; “####” is image number (REQUIRED). If % and # symbols are not used, the format will automatically be filename_template%%.####. This template is overridden by any segment_filename_template_N.		
directory_save	CSTRING	None	“/data/test1/”
	Directory to save images (REQUIRED).		
save_raw	bool	No	no
	Set to yes to save raw images (PROPRIETARY MODE ONLY).		
save_corrected	bool	Yes	yes
	Set to yes to save corrected images (RECOMMENDED).		
save_thumbnail	bool	No	Not Used
	Save a thumbnail of the image (NOT AVAILABLE).		
autostart	bool	No	no
	Automatically start data collection when protocol file loaded (or “marccd –p protocol_file” command line option used).		
autoquit	bool	No	no
	Automatically quit marccd program when finished.		
comment	CSTRING	None	“Lysozyme batch 1”
	Enter comments into image headers.		
operator_name	CSTRING	None	“RAD”
	Enter person's name into image headers for reference.		
archive_command	CSTRING	None	“/bin/echo”

Copyright © 2000-2006, Mar USA

	Archive command executed after each image with this command; most recent filename is appended as an argument to each command.		
periodic_command	CSTRING	None	"/bin/echo hello_world"
	A command executed at specific intervals. Will be executed every [periodic_interval] (e.g. 30) [periodic_mode] (e.g. SECONDS), starting [periodic_start_phase] (e.g. AFTER) the first [periodic_start] (e.g. 100) [periodic_start_mode] (e.g. SECONDS).		
periodic_start	CINT	0	1
	The sequence number at which the periodic_command starts.		
periodic_start_phase	KEYWORD	After	AFTER
	Options: "BEFORE," "AFTER." The phase for the first periodic_command executed.		
periodic_interval	CINT	0	1
	The interval at which periodic_command is executed.		
exposure_mode	KEYWORD	Time	TIME
	Options: "TIME," "DOSE." The desired exposure mode for the dataset if not defined in each segment).		
detector_multiread	CINT	1	1
	Number of CCD reads per frame. 2 reads will collect two frames of half total exposure time, average, and de-zinger. MORE THAN 2 READS IS NOT RECOMMENDED.		
center_crystals	bool	No	yes
	Automatically center each crystal after it is loaded.		
unload_last_sample	bool	Noa	yes
	Automatically put away the last sample into the carousel at end of dataset.		
save_video_stills	bool	No	yes
	Save video still images after each image collected.		
segment_filename_template_N	CSTRING	None	"crystal1_###.image"
	Image name template for a particular segment; overrides filename_template.		
segment_number_N	CINT	N	1
	Segment identification number for this segment. Default is the integer N (in parameter name).		
sample_number_N	CINT	0	14
	Sample number for this segment; will change sample automatically if this value is not 0 and different from previous segment sample number.		
detector_binning_N	CINT	?	2
	Detector binning for this segment. Default binning for MarMosaic detectors is 1; default for MarCCD detectors is 2.		
axis_N	CSTRING	Phi	phi
	Axis to change for this segment using motion_N; must be configured correctly for each detector installation. For example, the phi axis is usually configured to be scannable.		
motion_N	KEYWORD	Scan	SCAN
	Options: "STILL," "SCAN," "OSCILLATE," "DARK." Axis motion of axis_N for this segment.		

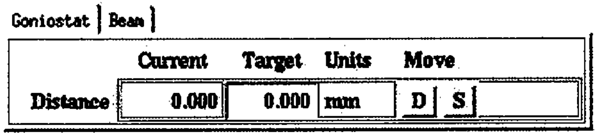
width_N	double	0.0	1.0
	The amount of motion per image of the desired axis		
gap_N	double	0.0	0.0
	Gap advanced along desired axis between each image.		
exposure_parameter_N	double	0.0	3.0
	Exposure parameter (e.g. seconds in the case of time mode, or counts in KHz*sec in the case of dose mode).		
exposure_mode_N	KEYWORD	Time	TIME
	Options: “TIME,” “DOSE.” Exposure mode for this segment.		
frame_first_N	CINT	0	1
	First frame number for this segment.		
frame_next_N	CINT	0	1
	Next frame at which to start taking data. Usually the first, but not always (for example, starting after an interrupted data collection).		
frame_last_N	CINT	0	90
	Last frame number for this segment.		
segment_group_size_N	CINT	1	1
	Segment group size for interleaving segments.		
segment_chunk_size_N	CINT	1	1
	Interleaved segment chunk size.		
comment_N	CSTRING	None	“Lysozyme crystal 14”
	Comment for image headers for this segment; overrides comment parameter.		
phi_first_N	double	0.0	90.0
	Rotates the sample to this phi value at the start of the segment. Requires a phi axis to be present and configured correctly.		
distance_first_N	double	0.0	120.0
	Drives the sample to this distance value at the start of the segment. Requires a distance axis to be present and configured correctly.		
[other axis]_first_N	double	0.0	0.0
	Depending on particular detector installation, other movable axes could be present and able to be driven using parameters of this format. Drives the axis to this value at the start of the segment		

Goniostat menu and control buttons

Note: If the goniostat menu is not currently displayed on the screen, and rather a small histogram or other graph is shown, try clicking the Pointer tool in the X-ray Data Detail tab.

The goniostat menu is used to see the current status of parameters that the software knows about. It is also used to drive motors, initialize, and set values by hand (if configured to do so). This menu is variable depending on the configuration, and two examples are shown here.

Figure 16 shows the simplest case, where the software only contains Distance and Wavelength information.

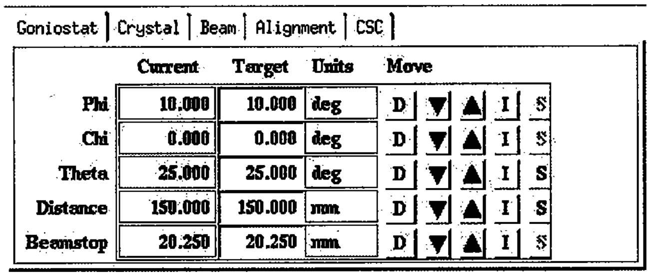


	Current	Target	Units
Distance	0.000	0.000	mm

Figure 16 - Goniostat menu (no goniostat)

If the software is not configured to control these parameters, the "S" button (Set) is used to define the wavelength and distance for recordkeeping (these values are recorded in image headers). Type in a value and click the "S" button. If the software is configured to control parameters Distance and/or Wavelength (by sending external commands), type in new target values and use "D" to Drive to new targets. *(Note: Currently, external goniostat feedback is not supported; therefore, though the current value is shown updated after Drive button is clicked, the software has not actually received a confirmation from the external device that it reached the target.)*

Figure 17 shows the goniostat menu in the case that a MarCSC is present.



	Current	Target	Units
Phi	10.000	10.000	deg
Chi	0.000	0.000	deg
Theta	25.000	25.000	deg
Distance	150.000	150.000	mm
Beamstop	20.250	20.250	mm

Figure 17 - Goniostat menu (MarCSC)

The following tabs are present:

- Goniostat: Motors normally considered part of a goniostat
- Crystal: Crystal control motors for aligning the crystal
- Beam: Beam (wavelength) and beam shaping parameters (slits)
- Alignment: The four axes to align the instrument to the X-ray beam—horizontal and vertical translation and rotation
- CSC: Sample changer motors (manual controls disabled)

There are the following buttons present for most axes:

- **D**: (Drive) Type in a desired value to the left and click the button "D" for drive (alternatively, type in several values and click the "Drive" button below the goniostat dialog). The software will not drive if the target is outside the configured software limits.
- **Arrow buttons**: Click and hold to drive the motor "by hand" in the desired direction; release to stop driving.
- **I**: (Initialize) Click to initialize. This motor will drive to the maximum or minimum hardware limit value and stop. Initializing any Alignment motors will require doing the Beam Alignment procedure again (see Chapter 7).
- **S**: (Set) Caution—this redefines the current value in the software, so it is rarely used for motors controlled directly by *marccd*. Type in a different value and click the Set button to redefine the present position in the software. This is useful for recordkeeping of values

such as Wavelength that may not be controlled by the software, but need to be recorded in image headers.

Below the goniostat menu are several goniostat control buttons, shown in Figure 18. They have the following functions:

Figure 18 - Goniostat control buttons

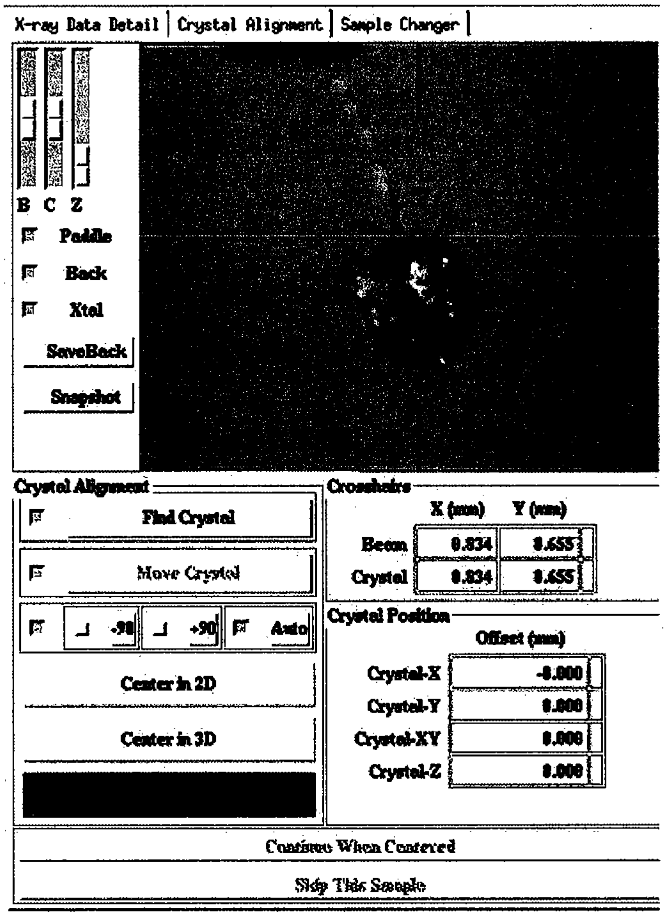
- **Current button**: Clicking this button will cause the software to ask the goniostat for updated parameter values, and all Target value boxes will be filled in with Current values.
- **Drive button**: Use this button to drive one or several software-controlled axes to target values, after typing target values into the goniostat menu above.
- **Stop button**: Stop driving the goniostat axes.
- **Align button**: This button used to open mar345dtb program for aligning the MarDTB. It has been replaced by the Beam Alignment tab (see Chapter 7).
- **ToHeader button**: This button puts the target values from the goniostat menu into the header of the image frame currently displayed (the new header will not be saved to disk unless manually saved with File→Save Data Frame). This is useful for image inspection tools (e.g. Resolution Ring, where the Distance, Wavelength, and beam center are used for computing resolution).

Crystal Alignment Tab

Note: When no MarDTB or MarCSC is present, this tab is normally configured to be absent.

The crystal alignment tab, shown in Figure 19, displays a video image of the crystal from the MarDTB.

Copyright © 2000-2006, Mar USA



Crosshairs

	X (mm)	Y (mm)
Beam	0.834	0.655
Crystal	0.834	0.655

Crystal Position

	Offset (mm)
Crystal-X	-0.000
Crystal-Y	0.000
Crystal-XY	0.000
Crystal-Z	0.000

Figure 19 - Crystal alignment tab

The options available on this tab are different depending on whether the MarDTB has a MarCSC/MarXY (Self-Centering Phi Axis Option) installed or not.

- **Functions available for MarDTB (with or without MarCSC/MarXY options)**

The following scrollbars can be used to adjust the video image display if desired:

- **B** (Brightness) bar
- **C** (Contrast) bar
- **Z** (Zoom) bar

The following buttons are located to the left of the video image:

- **SaveBack button**: click this button when a crystal is not mounted in order to manually save a background into the ~/log/xtal/ directory for automatic crystal centering.
- **Snapshot button**: click this button to save a snapshot of the video image in the directory ~/log/xtal/.

Copyright © 2000-2006, Mar USA

- **Functions available only for MarCSC or MarXY option users**

The following checkboxes, located left of the video image, control physical pieces of equipment on the MarCSC or MarDTB with MarXY (Self-Centering Phi axis, not available on standard MarDTB):

- **Paddle** checkbox: moves the paddle behind the crystal, improving the visual contrast for automatic centering
- **Back** checkbox: turns on the backlight on the paddle
- **Xtal** checkbox: turns on a blue LED light pointed at the crystal for changing contrast which may improve automatic centering)

For MarCSC/MarXY (Self-Centering Phi Axis Option) users, the following buttons are used for alignment from the workstation when not collecting a dataset. These buttons are used for aligning a crystal while not collecting a dataset (for more of a tutorial on aligning crystals, see Chapter 8 for MarCSC or MarXY Self-centering Phi Axis Option users, or Chapter 7 for MarDTB users).

- **Find Crystal button and checkbox**: Finds crystal in current image using the software configured (default is program *marloop*). A background image (with SaveBack back button) is required to be present; if none is present, the software will automatically move the crystal back along the Z axis, out of the field of view of the camera, take a background, and return the crystal to the original position. The checkbox affects whether this procedure is done during the Center in 2D procedure.
- **Move Crystal button and checkbox**: After middle-clicking with the mouse on the crystal position in the video display, click Move Crystal button to drive the crystal to the Beam crosshair position at the center of the image, by moving X and/or Y, and Z axes. The checkbox affects whether this procedure is done when doing the Center in 2D procedure.
- **+90 button and checkbox**: Clicking the button itself drives phi +90 degrees from current position. Selecting the checkbox affects which task is done during the Center in 2D procedure.
- **-90 button and checkbox**: Clicking the button itself drives phi -90 degrees from current position. Selecting the checkbox affects which task is done during an automated procedure (Center in 2D, 3D, etc.).
- **Auto button and checkbox**: The software moves Phi by the amount requested by crystal centering program (default *marloop* uses +90). Selecting the checkbox affects which task is done during the Center in 2D procedure.
- **Center in 2D button**: Clicking this button will do the tasks selected by checkboxes above (Find Crystal and/or Move Crystal, and/or rotate phi +90, -90, etc.). The crystal position is automatically detected by image recognition software (default configured program is *marloop*), and the X and/or Y and Z axes are driven to move the crystal to the center of the video image. Phi is not rotated; therefore, the crystal is not necessarily centered yet (see Chapter 8 to read a description of how to center a crystal).
- **Center in 3D**: Click this button to automatically do the entire algorithm to center a crystal along the phi axis. Images are taken and analyzed between successive phi axis rotations. The algorithm is explained in Chapter 8.
- **Stop button**: click this button to stop the current crystal alignment function being done.

The following crosshair positions (software crosshairs overlain on the video image) are displayed below the video image (*note: X and Y here refer to the horizontal and vertical coordinates corresponding to the center of the video screen, not the X and Y positions of the individual axes*):

- **Beam** (X, Y): The crosshair normally displayed with red lines overlain on the video image (when the pointer points to the image).
- **Crystal** (X, Y): The crosshair normally displayed with yellow lines overlain on the video image (when the pointer is over the image). Clicking the mouse on the image changes the position. This is used for point-and-click centering (see: How to Align a Crystal).

Under the title Offset, the X, Y, and Z positions corresponding to the position of the Crystal Position crosshair are displayed (at the bottom right of the Crystal Alignment tab):

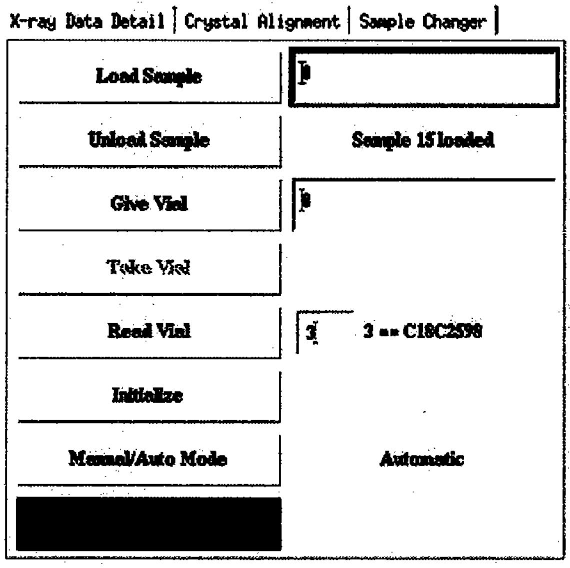
- **Crystal X**: Distance along goniometer head X axis from Beam crosshair on video image to Crystal Position crosshair on video image. Depends on current Phi position.
- **Crystal Y**: Distance along goniometer head Y axis from Beam crosshair on video image to Crystal Position crosshair on video image. Depends on current Phi position.
- **Crystal XY:** This is the relative horizontal position on the screen of the Crystal Position crosshair to the Beam Position. It is the length of the vector made by X and Y positions, projected on the horizontal axis of the screen: $CrystalXY = \sqrt{|X|^2 + |Y|^2}$.
- **Crystal Z**: Position of the Z axis (corresponding to Vertical direction in video screen) of the Crystal Position crosshair.

The following buttons are used when during a dataset data collection, using semi-automatic centering (described in Chapter 8):

- **Continue When Centered button**: After mounting a crystal and before collecting any images with it, it should be aligned to the phi axis using whatever methods desired from the Crystal Alignment tab (or local remote control). Then click this button to collect the images.
- **Skip This Sample button**: Do not center this sample or collect any data on it; continue in dataset to next sample to be loaded from the Acquire Dataset dialog.

Sample Changer Tab (MarCSC users only)

This tab, available only when marccd is configured for a MarCSC Cryogenic Sample Changer, looks as shown in Figure 20:



Sample 15 loaded

3

3 -- C18C2598

Automatic

Figure 20 - Sample Changer tab

- **Load Sample button and text field**: To load a particular sample, type the number of the desired sample into the text field, and then click the button. If the button is not available, the sample changer may be in Manual mode or doing other tasks such as data collection. Stop the other tasks if desired and/or switch to Automatic mode (see Chapter 8 about Manual/Automatic mode).
- **Unload Sample button**: Click the button do unload the currently mounted sample and put away in the carousel. The sample must be Automatic mode and not doing other tasks, or else this button would be grayed out.
- **Give Vial button and text field**: Use Give Vial to replace one or a few individual vials in the carousel without taking the time to pull out the entire carousel. Type a number of a particular sample in the text field and click the Give Vial button; the sample changer will retrieve a vial from the carousel and extend the server arm for a person to take the vial by hand (use gloves). The sample changer waits with the server arm extended until the Take Vial button is clicked (see below).
- **Take Vial button**: This is available only after the sample changer is put in the Give Vial state. While the server arm is extended, put a vial in the server "hand" mechanism to store in the carousel (or, if an empty position in the carousel is desired, leave the server hand empty). Click Take Vial to take the vial and return the sample changer to normal operation.
- **Read Vial button and text field**: Type a sample number in the text field and click this button to remove a vial from the carousel, scan the cap with the barcode reader, and replace the vial back into the carousel. The sample number (carousel position) and barcode ID will be printed to the right of the text field. Read Vial is available even when another sample is mounted (and the sample changer is otherwise idle).
- **Initialize button**: Click this button to initialize the sample changer. Use this button whenever the carousel has been installed. Sample changer motors and Crystal translation motors X and Y are initialized. If a sample is not mounted, Z is also initialized; if a sample is mounted when this button is clicked, the sample is left mounted on the

Copyright © 2000-2006, Mar USA

goniometer head, and the Z axis is not initialized (if Z were initialized, the crystal would be driven out of the cold stream and melted).

- **Manual/Auto Mode button**: Click this button to change between Manual and Automatic mode. ***Before switching to Automatic mode, the user must very carefully determine whether a sample is mounted, and if so, which number in the carousel it is ok to return to. If there are no empty vials (vials without caps) in the carousel, the mounted sample must be removed by hand.*** See Chapter 8 for more information about Manual/Automatic mode.
- **Stop button**: Click this button to stop the task that the sample changer is doing.

Log files

Notes: The base directory for log files is defined by the environment variable MARCCDLOG. Usually it is ~/log/, and below it is assumed to be ~/log/./). ***We recommend never deleting log files! They contain useful information that may help solve problems in the future, for example, understanding when an error condition first appeared.*** *If a file is suspected of being corrupted (for example, it is overly large—several GB), it may be archived by changing to a different name (e.g. "mv stdouterr.log stdouterr.log.1," as long as the new name for the file does not already exist). A fresh file will be created by* marccd.

A detailed log file of the *marccd* program is recorded in the file stdouterr.log (found in the directory ~/log/).

A useful log of dataset data collection information, including each image, parameter, and filename, is recorded in the file ~/log/dataset.log.

If the MarDTB (with or without MarCSC) is used, a "spy" file of specific actions by the instrument is recorded in the file ~/log/mardtb_spy.log.

A log of MarDTB beam alignment scans is recorded in ~/log/beam/. The files are dtb.scan.XXX (where XXX is a three-digit number). They can be viewed as graphs in the Beam Alignment tab by clicking the "Plots" button.

Video images are recorded into the directory ~/log/xtal/. For example, background images are saved as empty.pgm, whether recorded automatically during crystal alignment (MarCSC), or if recorded by clicking "SaveBack." If the "Snapshot" button is clicked in the Crystal Alignment tab, a file snapshot_XXX.pgm, where XXX is a three-digit number, is saved here.

- **Additional logged information for MarCSC and MarXY users only**

For MarCSC and MarXY Self-Centering Phi Axis users, files named "find.XXX.pgm" (where XXX is a 3-digit number) are recorded for each successive alignment in the ~/log/xtal/ directory. The program *marloop*, if used as the image analysis program (default), records images in this directory marloop_crystal.pgm, marloop_crystal-marloop_empty.pgm.

Copyright © 2000-2006, Mar USA

During data collection with the MarCSC, images are recorded for each crystal at the Phi 0 and Phi 90 positions. This is for later visual verification of the accuracy of centering. A montage of these 0 and 90 images is recorded in the data directory being written to. The files, called "marccd_motage_0.pgm" and "marccd_montage_90.pgm," look something like Figure 21:

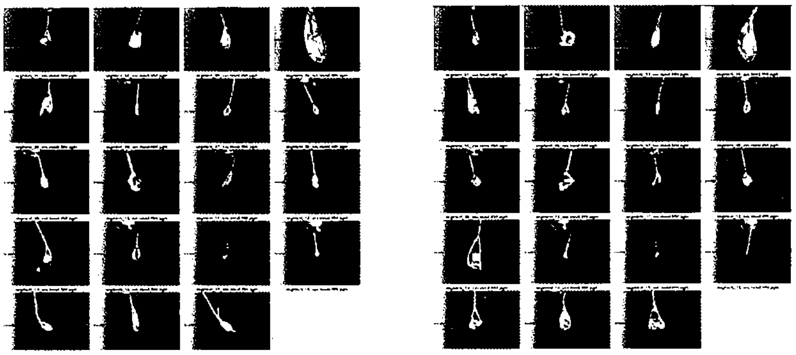


Figure 21 - Montage of crystal images collected for dataset

In the montage images, an empty space is inserted for crystals on which data collection was requested, but automated crystal alignment failed.

Chapter 4: Detector Start-Up Procedures

Note to one-time detector users: normally the detector is running and 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.

If the detector is powered on, the controller and the cooler are running, but the software is not yet running, go to Chapter 1. If the marccd *software has already been started, and the detector is at the operating temperature (typically between -70 C and -80 C), then the user only needs to get started collecting data; go to Chapter 2.*

MarCCD Detector Startup

Note that the hardware setup (connecting hoses and cables) is explained in Chapter 6: Detector Hardware.

1. Turn on the detector power

The big cabinet on rollers provides access to the internal main power switch through an opening in one of the side panels. Reach in and turn the power switch inside on. The switch is lit up red when the power is on.

2. Log on to the workstation and start the *marccd* software

Log into the computer controlling the MarCCD (default password shown):

```
user: marccd
password: marccd
```

To start the MarCCD software type:

```
marccd <return>
```

The *marccd* main window will appear on the screen as in Figure 1 (above).

3. Reboot the detector controller

If the detector has been turned off since the last time this software was run, the controller needs to be rebooted.

Note: It is not necessary to reboot if the software was stopped and restarted (while the detector was left running), or even if the computer has been rebooted. If the status window in the program marccd indicate question marks "?????" for temperature and pressure, then the detector should be rebooted.

Copyright © 2000-2006, Mar USA

Click on Configure→Detector, and the Detector Configuration dialog box will appear.

Click on the Reboot button (bottom left of dialog, as shown in Figure 22).

Figure 22 - Reboot button in MarCCD Configure->Detector dialog

A question box will pop up. Click on "Default Parameters."

Click on "Reboot."

The controller will emit three quick, loud beeps which indicate that it has rebooted successfully.

A moment later, a blue dialog box will appear with the question, "Do you wish to restart the cooling?" Click, the "Yes" box.

Now wait for the detector to cool down, two to four hours (depending on the quality of the vacuum, refrigerant gas pressure, starting temperature, ambient temperature).

When the detector is within two or three degrees of the operating temperature (which can be seen in the Configure→Detector dialog), it is ready to collect data.

MarMosaic Detector Startup

Note that the hardware setup (connecting hoses and cables) is explained in Chapter 6: Detector Hardware.

1. Make sure Controller Power Switch is off and System Power Switch is on.

Note for MarMosaic 300 and 325 users: Do this for both cabinets, in either order. The MarMosaic 225 has only one cabinet.

Controller Power Switch indicated in Figure 23 should be switched to "off" position, if not already.

System Power Switch indicated in Figure 23 should be in "on" position (green light is lit), if not already. This is the component normally labeled "Marway Power Systems."

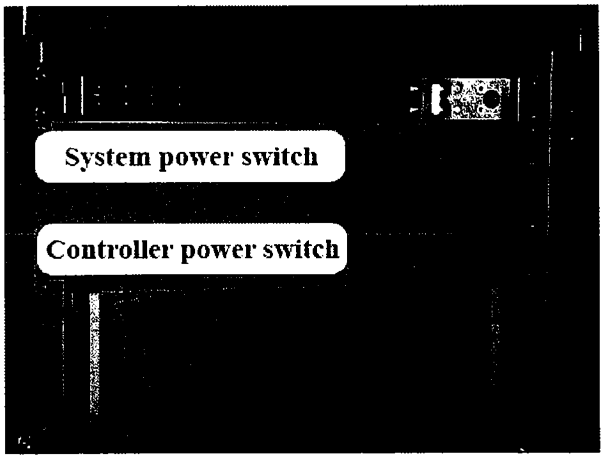


Figure 23 - MarMosaic power switches

2. 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 24).

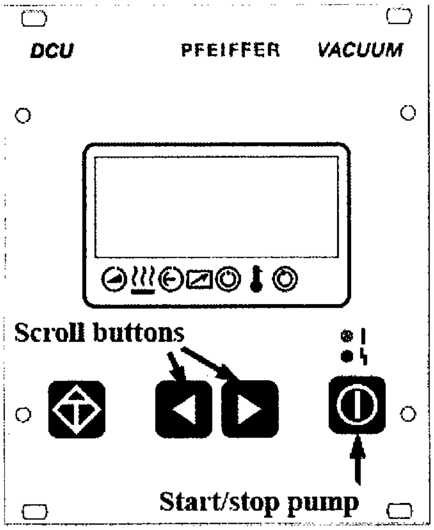


Figure 24 - Pfeiffer TMP controls

Use the Scroll buttons (left or right) shown on Figure 24 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.

3. Turn on the detector power

Copyright © 2000-2006, Mar USA

Important: the vacuum pump should already be attached to the detector head and running (see previous section).

Note for MarMosaic 300 and MarMosaic 325 users: Do this section twice (once for each of the cabinets). The order of powering on does not matter, but the second Controller Power Switch must be powered on within five seconds of the first. The MarMosaic 225 uses only 1 cabinet.

Switch the System Power Switch on inside the MarMosaic cabinet, as shown above in Figure 23. This is the component normally labeled "Marway Power Systems."

Next, switch on the MarMosaic Controller Power Switch, also shown in Figure 23.

Switch on the compressor power switches (if not already in the **on** position) (Figure 25).

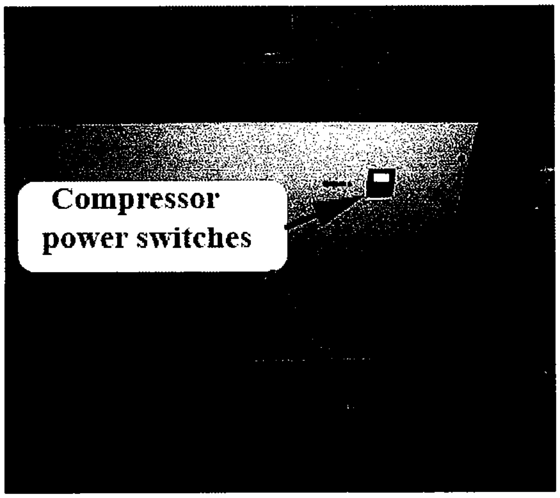


Figure 25 - MarMosaic compressor power switches

The compressors will begin to operate after a time delay of about 45 sec to 1 minute.

The MarMosaic controllers boot themselves and begin cooling the detector automatically; they do not need to be rebooted from the software (unlike the MarCCD controllers).

4. Log on to the workstation and start the *marccd* software

The control software for MarMosaic detectors is called *marccd.*

After switching on detector as above, log into the computer controlling the MarMosaic (default password shown):

```
user: marccd
password: marccd
```

To start the *marccd* software type:

Copyright © 2000-2006, Mar USA

```
marccd <return>
```

The main *marccd* window will appear on the screen as in Figure 1 (above).

Check that the software Status Window is displaying the Temperature and Pressure (if the detector is switched on as in the previous section, but the Temperature and Pressure fields in the marccd status window read “?????”, then jump to the troubleshooting section of this manual).

Now wait for the detector to cool down. Normally it takes about 3-6 hours (depending on the quality of the vacuum, refrigerant gas pressure, starting temperature, ambient temperature).

When the detector is within two or three degrees of the operating temperature (which can be seen in the Configure-Detector dialog; -70 C to -80 C typical), it is ready to collect data.

Copyright © 2000-2006, Mar USA

Chapter 5: Detector Shut-Down Procedures

MarCCD shut-down procedure

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

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

2. Power off the detector electronics cabinet.

3. If any cooling hoses will be disconnected (for moving, shipping, storage, etc.) wait at least 3 hours for the detector to warm up to room temperature before making disconnections. *It is strongly recommended never to make disconnections if not necessary. The detector head can be moved from one room to another without disconnecting by carefully strapping to the cabinet and rolling together.*

MarCCD shipping procedure

- Always contact Mar USA before shipping, because most service issues can be solved remotely.
- If shipping is required, usually only the detector head and electronics controller need to be shipped, **not the entire cabinet or cooler** (Mar USA service personnel will advise about this).
- Always follow the MarCCD 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 special padded boxes for the detector head (which can be obtained from Mar USA).

MarMosaic shut-down procedure

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

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

2. Power off the Controller Power Switch (not the System Power Switch) as shown in Figure 23 (MarMosaic 300 and 325 users, power off both Controller Power Switches in either order).

Copyright © 2000-2006, Mar USA

3. Stop the TMP by pressing the button on the front face (as shown in Figure 24) 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. Power off the System Power Switch (MarMosaic 300 and 325 users, power off both in either order).

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

MarMosaic shipping procedure

- Always contact Mar USA before shipping, because most service issues can be solved remotely.
- If shipping is required, usually only the detector head needs to be shipped, **not the entire cabinet(s) or coolers** (Mar USA service personnel will advise about this).
- Always follow the MarMosaic 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 special crate originally shipped with detector head.

Chapter 6: Detector Hardware

CCD X-ray Detector: Principles of Operation

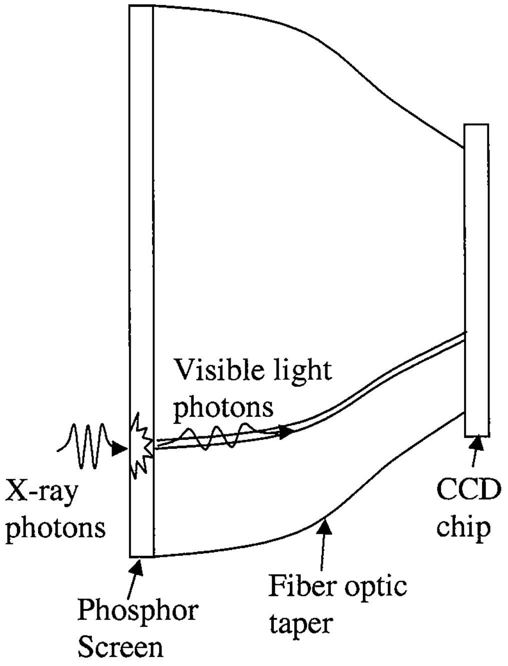


Figure 26 - Side view of CCD X-ray detector (not to scale)

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

- **X-ray conversion to light by phosphor screen (see Figure 26)**

  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 26)**

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

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

Copyright © 2000-2006, Mar USA

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 at the lowest signal levels, but they are usually severely rate-limited compared to CCD detectors.

- **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 27 shows a one-port CCD and a four-port CCD during readout.

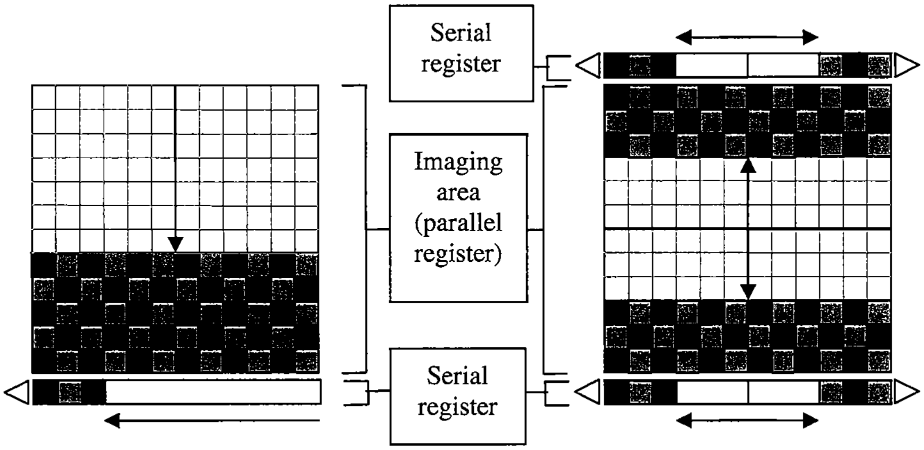


Figure 27 – One-port CCD (LEFT) and four-port CCD (RIGHT) diagrams, shown during readout. Checkered pattern indicates accumulated charge. Arrows indicate the directions in which the charge shifts.

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

During readout, for each pixel an analog voltage is received from the readout channel on the CCD. On each channel this signal passes through an amplifier, is modified by an analog offset to match the A/D (Analog to Digital) converter input, and then is converted to a digital signal by an A/D converter. The location where A/D conversion takes place is different for each detector. For the MarCCD, 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 MarMosaic detectors, the A/D converters are contained in the detector head, and the digital signals are sent from the detector head to the computer.

- **Transfer to computer workstation**

The digital pixel values for the image are transferred to the computer workstation with full-frame DMA (Direct Memory Access), meaning there is no load on the computer processor for transfer.

- **Background subtraction and image correction**

Normal images displayed on the screen and saved to files are corrected images. The correction process has three major steps: background subtraction, spatial 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 the MarCCD and MarMosaic 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 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.

- **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 *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 (MarCCD and MarMosaic)

CCD sensors must be cooled in order to reduce dark current (a constant, additive buildup of thermal noise that accumulates as the detector is integrating). For scientific imaging, CCD sensors are operated at extremely cold temperatures; MarCCD and MarMosaic 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 MarMosaic) 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.

- **Connecting and disconnecting gas lines**

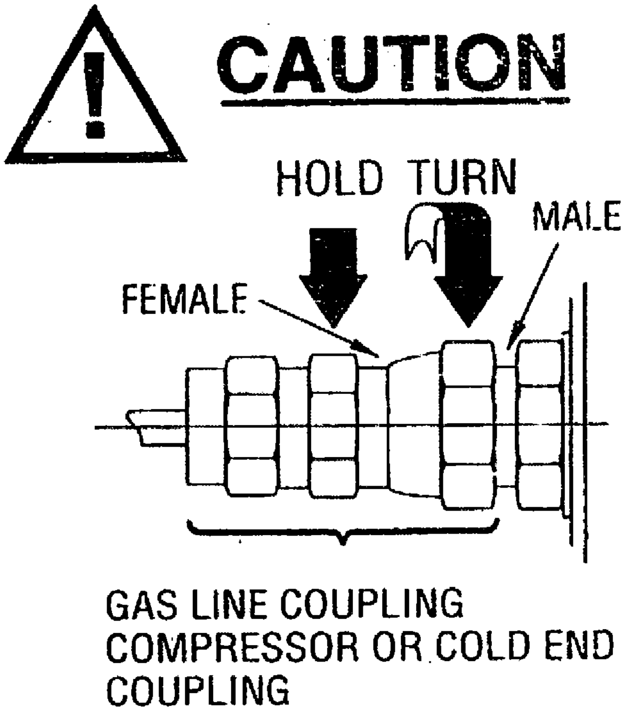


Figure 28 - Gas line coupling

  - 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 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)

Copyright © 2000-2006, Mar USA

MarCCD Hardware Setup

The following describes how to set up the system. A diagram of the hardware connections is shown in Figure 29.

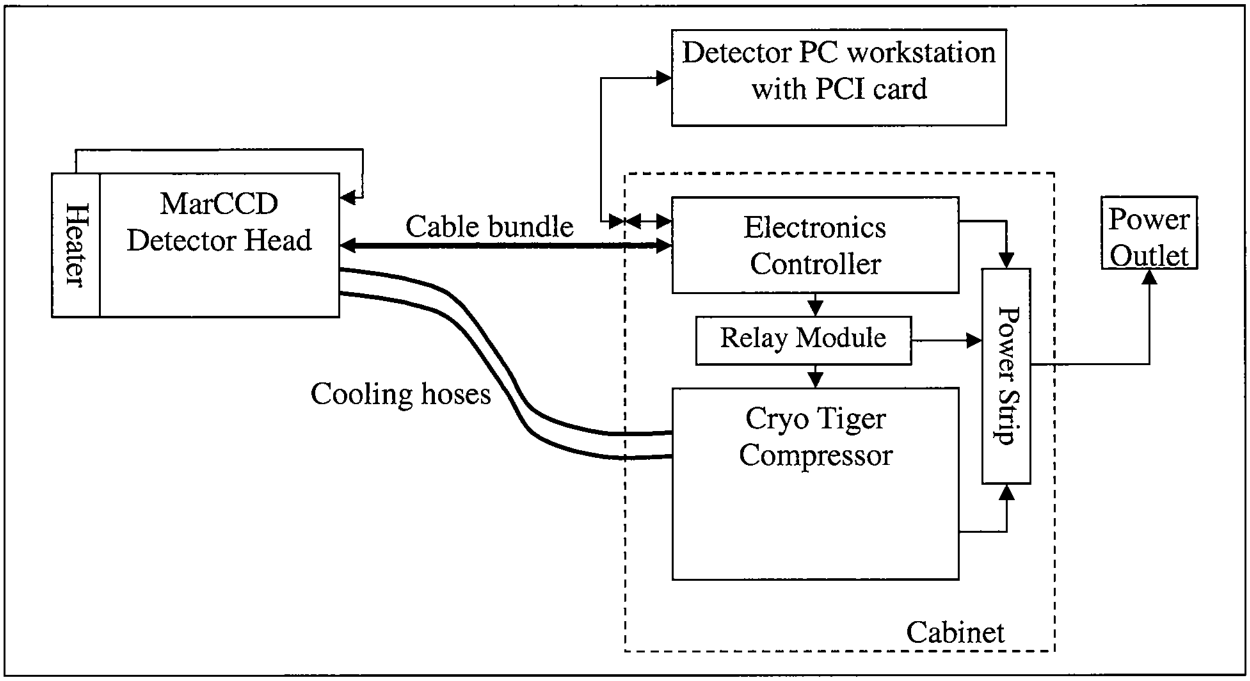


Figure 29 - MarCCD hardware connections diagram

Note: The detector cabinet power should be OFF before continuing.

- Connect cooling hoses. SEND should connect to SEND and RETURN should connect to RETURN. Refer to the section above: Connecting and disconnecting gas lines

- Connect cable bundle as labeled in Figure 30:
    - Gently connect CONTROL CABLE
    - Connect DC POWER IN
    - Connect four ANALOG OUTPUT SIGNAL cables. Notice that the analog signal cables are **NOT** BNC cables! They are Twin-Ax cables with a plastic tab inside and must be inserted in the correct orientation.

Copyright © 2000-2006. Mar USA

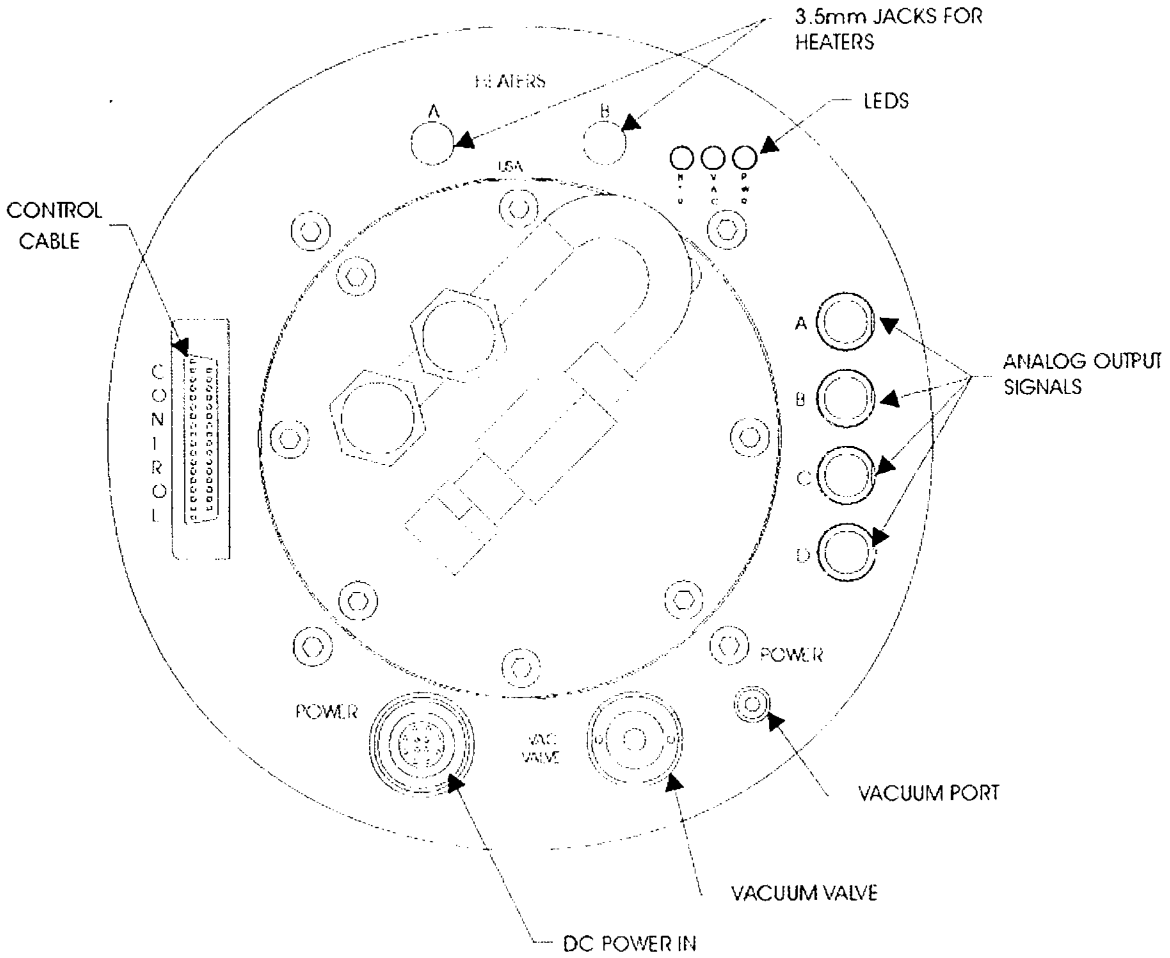


Figure 30 - MarCCD detector head diagram

- Connect cables to controller. Cables for A, B, C, and D must correspond to the colors inserted in A, B, C, and D inputs in the controller. See Figure 31.

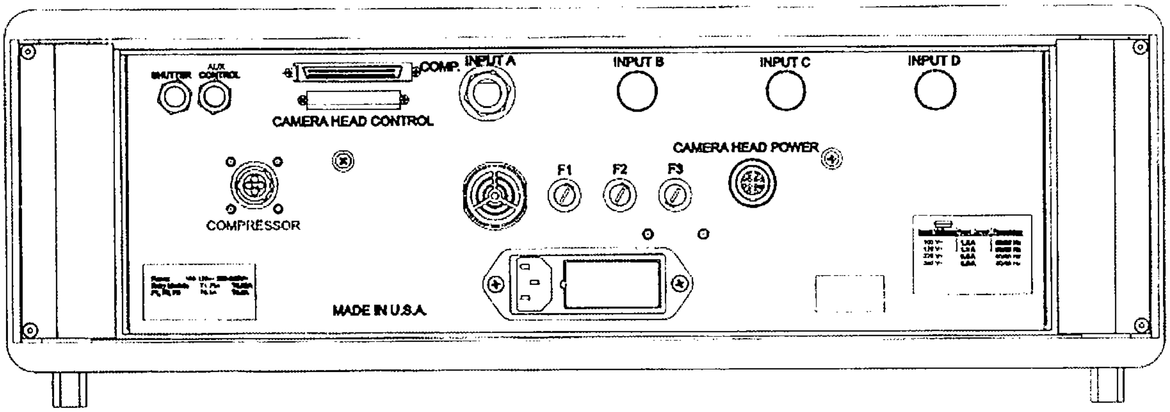


Figure 31 - MarCCD controller diagram

- Connect window ring heater 3.5 MM JACK FOR HEATER (either port A or B, if two heater ports are available)
- Connect signal cable from cabinet to computer (newer systems use fiber optic cable; older systems use 68-pin cable—for these systems, be careful **not** to connect to SCSI port in computer). Make sure connector inside the cabinet is secure, between controller COMP. jack and cabinet housing.
- Now do go to the startup procedure above, labeled, "MarCCD Detector Startup."

Copyright © 2000-2006, Mar USA

MarMosaic Hardware Setup

The following describes how to set up the system. A diagram of MarMosaic 225 hardware connections is shown in Figure 32. The setup for a MarMosaic 300 or 325 is the same, but there is one additional cabinet, with four cooling hoses connected to the detector head.

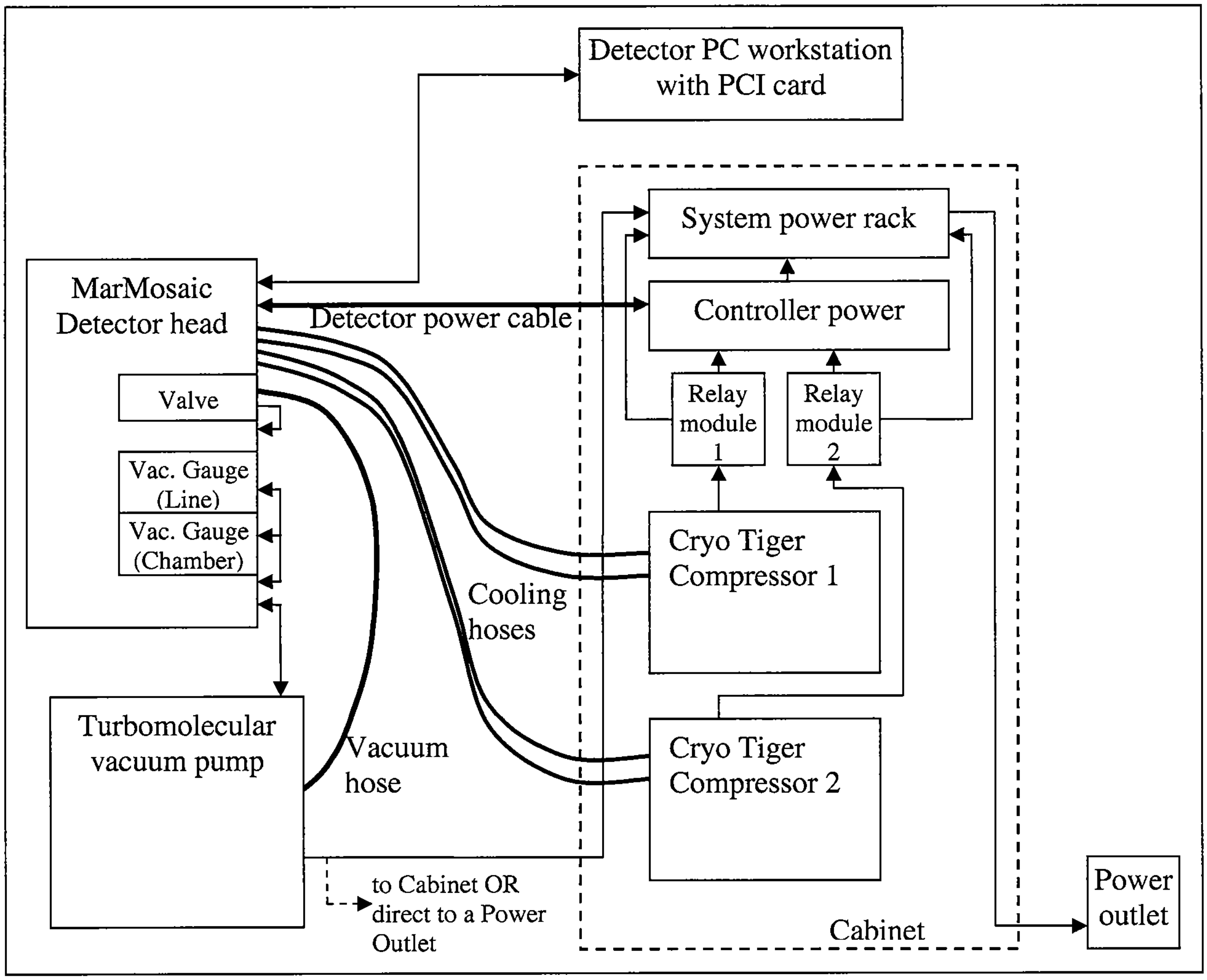


Figure 32 – MarMosaic 225 hardware connections diagram. MarMosaic 300 and 325 have one additional cabinet, with four additional cooling hoses connected to the detector head.

Note: The System Power rack(s) should be powered OFF before continuing (System Power racks are normally labeled "MARWAY POWER SYSTEMS."

- Connect all cooling hoses. Refer to the section above: Connecting and disconnecting gas lines
-
    - Connect "CRYO 1 SEND" on the outside of the cabinet to "CRYO 1 SEND" on the detector head.

  - Connect "CRYO 1 RETURN" on the outside of the cabinet to "CRYO 1 RETURN" on the detector head.
  - Connect "CRYO 2 SEND" on the outside of the cabinet to "CRYO 2 SEND" on the detector head.
  - Connect "CRYO 2 RETURN" on the outside of the cabinet to "CRYO 2 RETURN" on the detector head.
  - For MarMosaic 300 and 325, continue connecting CRYO 3 and CRYO 4, both SEND and RETURN, on the second cabinet.

- Connect the vacuum hose to the TMP (Turbomolecular vacuum pump). The TMP's AC power cord may be plugged directly into the Cabinet's power rack, or alternatively, directly to a 220V power outlet.

- Connect detector power cable between the detector head and cabinet as labeled in Figure 32, (the jack on the detector head is labeled, "POWER"), and twist the outer ring to lock.

- Connect the Vacuum Gauge (Line) and Vacuum Gauge (Chamber) together using the short end of the doubled cord. Connect the long end to the detector head, to the jack labeled, "Vac. gauge."

- Connect the Turbomolecular vacuum pump signal cable to the jack labeled, "Vac. Pump."

- Connect the vacuum valve to the jack on the detector head labeled, "Vac. Valve."

- Connect the fiber optic cable to the computer workstation, and to the jack labeled, "COMPUTER" on the detector head. If the computer workstation is already powered on, it may need to be powered down to reset and rebooted (stray room light into the end of the unconnected fiber optic cable can cause the computer to lock up). If the computer is off, power it on now, but do not start the *marccd* software.

- Now do go to the startup procedure above, labeled "MarMosaic Detector Startup."

Chapter 7: The MarDTB (Mar DeskTop Beamline) Advanced Goniostat

Overview: The MarDTB (Mar DeskTop Beamline)

The MarDTB is a precision single-axis diffractometer and goniostat system, compatible with MarCCD and MarMosaic 225 detectors. It features computer controlled motors for nearly every operation, operated by a real-time controller and Ethernet TCP/IP services. Under software control, the instrument uses four alignment motors, four Tungsten slits, and two ionization chambers to automatically align itself to an X-ray beam (see the Beam Alignment section later in this chapter). Alignment to the X-ray beam can be done at any time from the computer workstation, taking as little as about one minute. A 50x CCD microscope is used for viewing and aligning the crystal, displayed on a small LCD monitor attached to the instrument, and also in real-time in the marccd software Crystal Alignment tab.

Set up the system

The MarDTB requires a standard goniometer head (unless the MarCSC or MarXY Self-centering Phi Axis is installed). Some very large goniometer heads with extension pieces could crash into the surrounding MarDTB surfaces or even the detector face when phi is rotated, so please be sure that the goniometer head diameter is less than about 6cm at the base, and 25mm at the crystal position (the collimating “nose” on the MarDTB may be removed for such large goniometer heads). Please carefully check that it won’t crash.

Whenever a computer-controlled motor is changed by hand, the software will no longer know the position of that axis. It should be reinitialized by clicking the “I” button next to that axis in the Goniostat menu (located center right in the *marccd* main window). The slits and beamstop positions are easily changed by hand (deliberately or by accident).

If the MarDTB has been shipped or physically moved from one location to another since the last data collection, it is a good idea to initialize all of the axes including the alignment motors. After initializing any alignment motor, the beam alignment must be redone.

Load a sample

The standard MarDTB is now shipped with a phi-swing, though some exist without the phi-swing. The phi swing allows the phi axis hardware to rotate in the chi axis up to 90 degrees in order to load crystals directly from a vial full of liquid nitrogen. This is advantageous over loading in the horizontal phi orientation, because the crystal is kept safe and liquid nitrogen is not spilled on the instrument.

There are two ways to rotate chi for loading a sample: using the software goniostat menu (located center right in marccd main window) to drive to a specific angle, or by pressing the

Copyright © 2000-2006, Mar USA

"Chi" button on the local remote control while standing next to the MarDTB. If the local remote is used, press the "Chi" button again to drive back to zero after mounting, for data collection.

Take particular care not to accidentally brush the slit knobs while loading a sample, because they can be changed that way (and would require initialization).

Crystal centering

The crystal image can be seen in the MarDTB LCD screen while standing in front of the instrument (it can also be seen in on the computer, in the *marccd* Crystal Alignment tab when using a video capture card). Center the crystal using the goniometer head's X and Y adjustments. Center the crystal along the Z axis by computer control (or by turning the Z-axis knob by hand at the instrument). Normally, the user is now ready to start collecting data (in the case of a very unstable X-ray beam position, possibly a beam alignment should now be done).

MarDTB Beam Alignment using *marccd*

1. Introduction

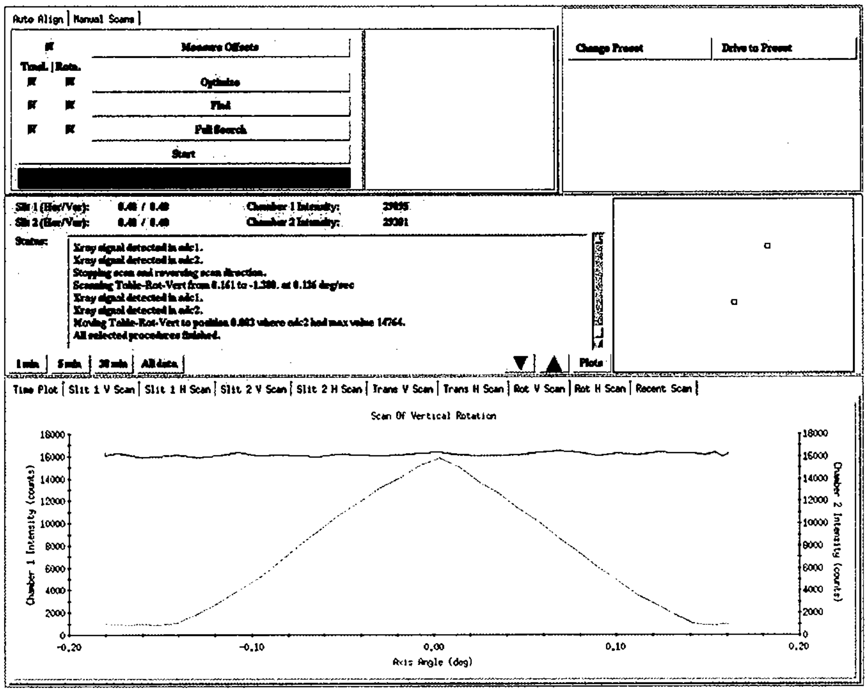
This section explains how to align the MarDTB goniostat to an X-ray beam using the Beam Alignment tab in *marccd*. The scope of this manual covers *marccd* versions 0.9.36 and later.

The MarDTB goniostat has four computer-controlled alignment axes: horizontal and vertical translation, and horizontal and vertical rotation. The MarDTB also uses two ionization chambers, each behind a set of vertical and horizontal slits. These provide enough information and degrees of freedom to align to a linear beam, provided that the beam is close enough to be within the range of motion of the motors. The first ionization chamber, closer to the beam source, is used for aligning the translation, and the second chamber is used to align the rotation. The centers of rotation of the two rotation axes (horizontal and vertical) are close to the first ionization chamber and slits.

The hardware and software has been tested to successfully align to beams as small as 50 microns.

2. Quick Start

Start the program *marccd* and make sure there are no communication problems with the MarDTB (i.e. error messages). Click the Beam Alignment tab in the main *marccd* window, which looks like Figure 33.



Slit 1 (Hor/Ver): 0.40 / 0.40
Slit 2 (Hor/Ver): 0.40 / 0.40
Chamber 1 Intensity: 23035
Chamber 2 Intensity: 23301

Status:

```
Xray signal detected in adc1.
Xray signal detected in adc2.
Stopping scan and reversing scan direction.
Scanning Table-Rot-Vert from 0.161 to -1.300. at 0.126 deg/sec
Xray signal detected in adc1.
Xray signal detected in adc2.
Moving Table-Rot-Vert to position 0.003 where adc2 had max value 14764.
All selected procedures finished.
```

Figure 33 - Beam alignment tab in *marccd*

Make sure that any X-ray shutters in the beam path before the MarDTB are open, so that the MarDTB has a beam to align to. Make sure all of the seven checkboxes in the Auto Alignment Options section are checked.

Click **Start**, and read carefully when prompted for information at the beginning of the procedure. Click **Measure** if prompted with a gray box regarding the Offsets. Click **Default** when prompted with a blue box regarding the slit settings. The procedure should now begin. The algorithm is quite sophisticated and can determine which parts of the search and optimization procedures need to be done.

After the prompts, the whole alignment procedure takes from one to 25 minutes (the bulk of the time is searching the entire range of the slow Vertical Translation alignment motor, up to the point when the beam is found). The procedure is finished when “All selected procedures finished” appears in the central message window, and the grayed boxes return to active.

Click **Stop** to stop the procedure in the middle for some reason. If something goes horribly wrong (e.g. there is no X-ray beam because a beamline shutter is closed), one can return the four alignment axes the original position when the program started by clicking **Drive** and then “Original Position” in the drop-down menu; or one can move the alignment motors to the most recent successful optimized position by clicking **Drive** and then “Optimized” in the drop-down menu. A prompt will follow to confirm or cancel the drive command requested.

3. Configurable Software Parameters for the MarDTB

This section is now found in the Appendix B: Configuration Guide for *marccd.*

4. Using the Auto Alignment features

The six buttons and seven check boxes in this section (shown in Figure 34) are used for the main functionality of the beam alignment program. During automatic alignment, the ionization chamber offsets and noise sigmas are required to distinguish X-ray signal from background noise. These can be measured by the instrument itself automatically (requiring slits to close and open again), or entered by hand by the user (to save time). Because there is a drift in the ionization gauges, there is a default 1800 sec timeout on the measurement of these parameters.

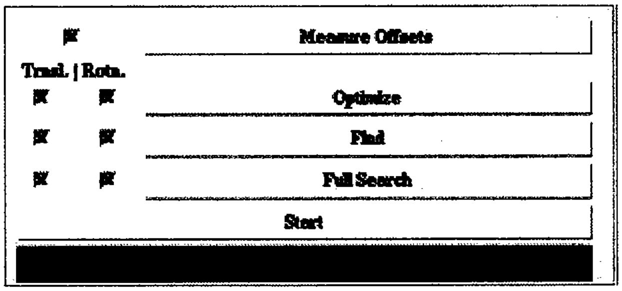


Figure 34 - Auto alignment features

a. **Start** button and the **Measure Offsets** checkbox

The **Start** button should be used most often. It will perform the functions selected using the checkboxes. It has a sophisticated algorithm to test which of the functions need to be done, so all of the checkboxes can be left checked unless there is a specific reason to uncheck (and thus disallow any test for) some functions. The amount of time required for the entire procedure can range from one to a maximum of 25 minutes, depending on whether the MarDTB needs an **Optimize** only, or ends up exhaustively searching the entire range of alignment motors. Movement of the slow vertical translation takes up the bulk of the time during a **Full Search**.

If the box **Measure Offsets** is checked, then the algorithm will *always* measure the offsets and sigmas of the ionization chambers after pressing **Start**. For users unfamiliar to the system, this box should remain checked. If the box is unchecked, the user will sometimes be prompted to decide whether to measure or type by hand the ionization chamber offsets and sigmas. Defaults will be suggested (which are configurable parameters above).

Next, the user will be prompted whether to use the current slit positions, or the default slit positions for the optimize procedure. Typically, the default should be used, if parameters are configured correctly. The program will accept any value of current slit settings; however, settings less than 0.05 mm are not recommended. Note that the optimized beam position and angle can vary for different slit settings.

Finally, the algorithm will begin the rest of the procedures selected by checkboxes. All buttons except the **Stop** buttons are grayed-out. There is an Alignment Message Window that gives occasional updates on the progress. The algorithm is finished when the display reads, "All selected procedures finished" and the grayed-out buttons return to being active.

b. **Stop** button

If the algorithm needs to be stopped at any time, press the Stop button. The **Stop** button in the Manual Scans section has equal functionality. The only cases in which the MarDTB won't stop immediately are during some slit scans.

c. **Measure Offsets** button

Click this button to measure or type in new offsets and sigmas, if one has reason to believe that the offsets have suddenly changed (for example if the ionization chamber gain setting was changed). During automatic alignment, the ionization chamber offsets and noise sigmas are required to distinguish X-ray signal from background noise.

Measurement of the offsets consists of closing the slits in the first chamber to cut out the X-ray signal, recording the offsets for a few seconds, and then returning the slits to their original positions.

d. **Optimize** button and checkboxes

The **Optimize** procedure is the most common procedure that will be done when using a stable X-ray source. It maximizes the X-ray intensity already coming through the slits at the beginning of the procedure, for the four alignment axes.

Click this button to do a stand-alone **Optimize** procedure (note that this procedure will be done automatically by clicking the **Start** button, if enough X-ray intensity is already coming through the slits). The checkboxes for the Translation and/or Rotation determine which axes will be optimized. The checkboxes are also used to determine which axes, if any, are optimized during the entire automated procedure when the **Start** button is clicked.

At the beginning of the stand-alone **Optimize** procedure, the user may be prompted to Measure or type in the ionization chamber Offsets. Then the user is prompted with a question about whether to use the Default slit settings from the configuration file, or use the Current slit settings. Normally, the Default settings should be used. Slit settings less than 0.05 mm are not recommended at this time.

Next, the ionization chambers for the desired axes (Translation or Rotation) are checked to make sure there is some significant X-ray intensity coming through the slits at the desired setting. *This is a prerequisite for doing an Optimize.*

Finally, the alignment axes begin to move and scan. The maximum position for each of the desired axes is determined by scanning completely over each peak—first Translation, then Rotation. After a full successful alignment (both Translation and Rotation), the position of the four axes is remembered by the program and can be accessed later (see below).

e. **Find** button and checkboxes

The **Find** procedure is the second-most common alignment procedure used for fairly stable X-ray sources. It consists searching for the X-ray beam within the maximum opening of the slits (typically about 4 mm), and using a slit scanning and testing procedure to find and center on the position of the X-ray beam. If the beam is less than about 4 mm from the current axis positions, it can be found with this procedure.

Click the **Find** button to do a stand-alone **Find** procedure, along the axes selected with the checkboxes (Translation or Rotation). The checkboxes are also used to determine which **Find** procedures (Translation and/or Rotation) are done when the **Start** button is clicked.

At the beginning of the stand-alone **Find** procedure, the user may be prompted to Measure or type in the ionization chamber Offsets.

Next, the procedure will test whether there is any X-ray intensity at the desired slit settings, to determine if a time-consuming **Find** is even necessary. If no X-rays are detected, the slits are opened their maximum settings, and X-ray intensity is checked again. If there is no X-ray intensity with these slit settings, then Find will fail. Otherwise, it will continue on to do a series of slit scans and tests, and eventually drive the motors to the calculated beam position.

f. **Full Search** button and checkboxes

The **Full Search** procedure searches the entire range of the four alignment motors for the beam and can be quite time-consuming (up to 20 minutes).

Click the **Full Search** button for a stand-alone version of the Full Search procedure, along the axes selected with the Translation and Rotation checkboxes. Note that **Full Search** is best done as a part of the full, automated **Start** procedure; otherwise it will probably leave the motors in only a very roughly aligned position. The checkboxes also used to determine which Full Search procedures (Translation and/or Rotation) are done when the **Start** button is clicked.

At the beginning of the stand-alone **Full Search** procedure, the user may be prompted to Measure or type in the ionization chamber Offsets.

Next, the X-ray intensity is checked successively at the Optimize slit settings, and then at the maximum settings, to make sure that the time-consuming Full Search procedure is necessary. Finally, the alignment motors will scan the full ranges with the slits set at the maximum setting. The procedure will stop the motors and exit normally if X-ray intensity is detected, or with a warning if no beam is found by the end of the procedure.

5. Using the Manual Scans features

Manual scans provide a way to do scans of single axes without doing the entire alignment procedure. These features are provided only for special cases (for example, testing the scan shapes for very oddly shaped beams).

The radio box allows the user to select which of eight axes on which to do a scan:

Vertical Translation
Horizontal Translation
Vertical Rotation
Horizontal Rotation
Slit 1 Vertical
Slit 1 Horizontal
Slit 2 Vertical
Slit 2 Horizontal

The directional radio box lets the user choose whether to scan to the Max of the motor range, or the Min, from the current position.

The slit scans are simple scans. The "table" motor scans (Translation and Rotation), however, contain some amount of processing. Table motor scans will search for an X-ray beam along that axis, then move to the maximum, if any, found.

The **Stop** button can be used to stop the Manual Scan features. Both this and the Auto Alignment Options **Stop** button have the same function.

6. Using the auxiliary features

1. Schematic diagram of alignment motors (Figure 35)

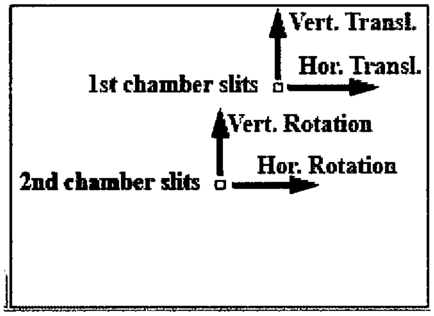


Figure 35 - Diagram of alignment motors

The white box represents the ranges of the “table” alignment motors (horizontal and vertical translation and rotation). Left-right on the screen corresponds to horizontal motors, and up-down corresponds to vertical motors.

The red box is a representation of the Slit 1 motors. The size is computed by the current Slit 1 Horizontal and Vertical positions relative to the translation motor ranges.

The blue box is a representation of the Slit 2 motors. The size is computed by the current Slit 2 Horizontal and Vertical positions relative to the *translation* motor ranges (not to the rotation motor ranges).

2. Graph window and tabs

The graph window has a tabbed interface with 10 tabs. Color graphs show ionization chamber 1 as a red line and chamber 2 as a blue line.

i. Time Plot and the **1 Min**, **5 Min**, **30 Min**, **All Data** buttons

The most recent approximately one hour of ionization chamber intensities are recorded and displayed on this tab. The display can be changed to show the most recent one minute, five minutes, thirty minutes, or the entire data, using the appropriate buttons: **1 Min**, **5 Min**, **30 Min**, and **All Data**.

ii. Eight tabs for the scan axes

All eight axes that may perform scans have a individual tabs: Slit 1 Horizontal, Slit 1 Vertical, Slit 2 Horizontal, Slit 2 Vertical, Horizontal Translation, Vertical Translation, Horizontal Rotation, Vertical Rotation. The most recent scan for each of these motors (if any has been done) is displayed in these tabs. If a scan is currently in progress, the scan is updated continuously. It is impossible to watch a tab other than the scanning axis during a scan.

iii. Recent Scan

The most recent scan is displayed here. More importantly, this tab displays saved scans that are opened using the **Plots**, **Up Arrow**, and **Down Arrow** buttons (see below).

iv. Displaying saved scans using the **Plots**, **Up Arrow**, and **Down Arrow** buttons

Up to 99 recent scans are saved automatically, normally in the [MARCCD_HOME]/log/beam/ directory. *If no such directory exists, no scans are saved.* Filenames are dtb.scan.x or .xx, where the x’s are numbers. The numbers recycled in a rolling fashion.

Copyright © 2000-2006, Mar USA

The recent scans can be displayed in reverse chronological order by using the **Down Arrow** (an upside-down triangle), or oppositely using the **Up Arrow** (a triangle). The scan data are plotted in the Recent Scan tab in the graph window.

Alternatively, specific scans can be displayed with the **Plots** button. A file selection dialog appears. Double click on the desired file, or single click, then click "OK". The scan data are plotted in the Recent Scan tab in the graph window.

3. Preset motor positions

For convenience, a few special motor positions are automatically recorded for easy access: the Optimized Position (the position of the most recent successfully completed Optimize of both Translation and Rotation), and Original Position (the position that the four alignment motors had when the program *marccd* was started). In addition, up to 10 presets can be saved by the user for recall later. This is useful if, for example, the beam position often changes between specific, well-known positions when the energy is changed at a synchrotron beamline.

i. **Change Preset** button and drop-down menu

Click one of the ten buttons **Preset 1-10** in the drop-down menu to change a preset position. A prompt will appear. The user will be given a choice whether to type the values by hand, or select the current motor positions. A second prompt will allow the user to confirm or cancel changing the preset.

ii. **Drive** button and drop-down menu

1. Optimized Position button

Click this button to drive to the most recent successfully completed Optimize position. There is a confirm/cancel prompt before executing the drive command.

2. Original Position button

Click this button to drive to the original position when the *marccd* software was started. There is a confirm/cancel prompt before executing the drive command.

3. Preset 1-10 buttons

Click one of these buttons to drive to a saved preset position 1 through 10. There is a confirm/cancel prompt before executing the drive command.

7. Alignment during data collection (in the Dataset dialog)

Copyright © 2000-2006, Mar USA

This is a feature already available for use in the current marccd version as a "beta" version (for testing). Alignment can be done during a dataset to "chase" an unstable X-ray beam, or to re-optimize the beam automatically when changing wavelengths. Before enabling these features, we recommend that you read this section carefully to understand what the algorithm will do in most cases.

To enable dataset beam alignment, the following line must be added to the configuration file, probably in the ~/.marccdrc or the mardtb_[SN].conf file (*CAUTION—only to be done by the staff responsible for the detector and equipment*):

```
dataset_alignment_enable      yes
```

After restarting the *marccd* software and opening the Acquire Dataset dialog, a new set of options will appear as in Figure 36:

Figure 36 - Dataset beam alignment options

Click the checkbox Align Goniostat and choose from the following options:

- Slit Settings selector

  You must select the slit values used during the alignment. They can be the Default or Current slit values. After any alignment is finished, slits are returned to the current slit values for data collection.

- Before Data Collection box

  Check this box to do an alignment before starting the dataset.

- After Every [N] Minutes/Frames/Segments box and selection

  Check the "After Every" box to do beam alignment periodically. Select the desired periodic counter (Minutes, Frames, or Segments) and enter the desired number of counts, N. For example, with Minutes selected and N=30, an alignment will be done after finishing an image at thirty minutes, and similarly, every thirty minutes thereafter. *Note on Segment counter: interleaved segment method of dataset data collection is too complex to use with this feature. With Segments selected, the algorithm will only align once on the first frame of the Nth segment.*

By default, the dataset alignment will only do the equivalent of the Optimize function. If no X-ray intensity is detected coming through the slits at the desired slit settings either before or after alignment, then the function will fail and go into a five minute pause. After the five minute pause, it will attempt to Optimize again. No data will be collected if the alignment never ends with success (defined as at least some X-ray intensity coming through the slits after Optimize).

For beamlines at which the beam position is likely to change by an amount that is equal or greater to the beam size, then only Optimize during dataset alignment may be insufficient to capture the beam. In this case, we recommend also adding the following line to the configuration file modified above:

```
dataset_alignment_find_enable      yes
```

In this case, the Find and Optimize functions will also be performed whenever alignment is to occur (as if the Find and Optimize checkboxes for 1st and 2nd chambers were selected in the Beam Alignment tab). The maximum slit setting for the MarDTB is approximately 4mm; thus the Find function will search for the beam within a two-millimeter position of the current position.

8. Beam Profile (in development)

Currently, there is no way to create a Beam Profile using the Beam Alignment tab in *marccd.* This function is still preserved in the mar345dtb program, however, which can be called by clicking the **Align** button in the Goniostat tab (or by quitting *marccd* and typing "mar345dtb --nodisp --no345". A profiling feature in *marccd* is under development.

MarDTB Troubleshooting guide

MarDTB Technical Description

Special section: The MarXY (Self-Centering Phi Axis option) for the MarDTB

Magnetic caps

Manual Crystal Centering using the local remote control

Point-and-click centering in the *marccd* software

Automatic Crystal Centering

Chapter 8: The MarCSC (Mar Cryogenic Sample Changer)

Overview: The MarCSC

The MarCSC (Mar Cryogenic Sample Changer) is an automated sample mounting device that is completely integrated into the MarDTB (Mar DeskTop Beamline) advanced goniostat. It features a carousel of 19 samples kept in liquid nitrogen, and removes and loads samples to the goniostat data collection position via automated computer control. Samples must be mounted on standard, commercially available pins of a certain length (usually 18 mm), inserted into SPINE (Structural Proteomics In Europe) standard magnetic caps (normally the bar-coded SPINE standard caps sold by Marresearch GmbH or Mar USA), and loaded into standard, commercially available magnetic vials; these vials are loaded into the carousel. The MarCSC includes a computer-controlled, motorized XYZ goniometer head for alignment of each sample to the X-ray beam. After each sample is mounted from the carousel, a 50X microscope CCD video camera is used for computer controlled sample alignment for the X, Y, and Z axes. The computer program *marccd* is used for both MarCSC and detector control, and fully automated, unattended data collection is possible for a complete carousel of 19 samples. Manual control of the instrument is also available, as well as a high degree of software configurability, for example, to work with an external image analysis program for sample alignment.

Quick Start

This Quick Start section describes operations assuming that the instrument and detector are both on and running, and that a new carousel full of samples has already been loaded into the Dewar of the MarCSC.

Start the *marccd* software

Open a new window at the workstation and start the marccd software by typing "marccd" at the terminal prompt. The software will establish communications with the MarDTB/CSC and the detector.

Load crystals into carousel

Load vials into the carousel under liquid nitrogen, using the transfer station included with the system.

Copyright © 2000-2006, Mar USA

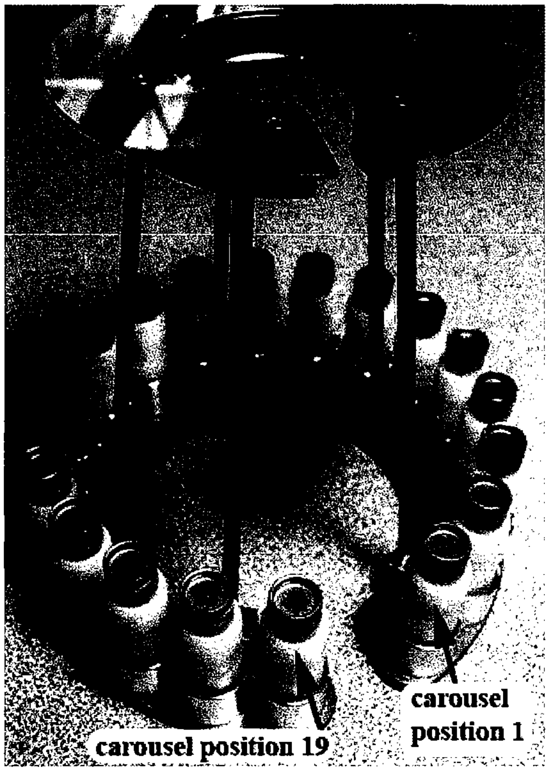


Figure 37 - MarCSC carousel. Positions 1 and 19 are indicated.

Load carousel into Dewar

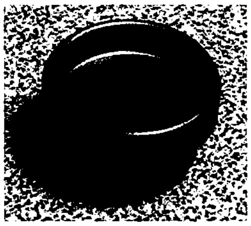
Make sure the instrument is idle (not collecting a data set or being used by any other user). Also make sure that no beam alignment process is occurring.

Initialize and align

Set up data collection and begin

Preparing Your Experiment

Caps



The goniometer head of the MarCSC only accepts special magnetic caps for mounting samples. High-quality magnetic caps with a laser-etched 2D barcode, which conform to SPINE (Structural Proteomics In Europe) standards, are available for sale from Mar USA in North America, and Marresearch GmbH in Europe and worldwide. Caps which do not conform to the SPINE standards cannot generally be expected to function reliably.

Copyright © 2000-2006, Mar USA

Pins and loops

Standard pins with loops, usually of the typical length 18mm, must be bought from another supplier such as Hampton Research, and inserted into the magnetic caps using glue or epoxy. If non-standard length pins are used, the program must be reconfigured (see the Sample Changer section of the Appendix Configuration Guide), or the automatic centering will be very unreliable.

At this time, the MarCSC automatic centering does not detect the location of a crystal within a loop; it only detects the loop position (actually, the centroid of a blob of liquid, loop and crystal). Therefore, in order to expect the automatic crystal centering to be successful, a loop that matches the crystal size must be used. For example, for a 50 µm crystal, a 75 µm loop would be appropriate.

Vials

Sample Transfer Station

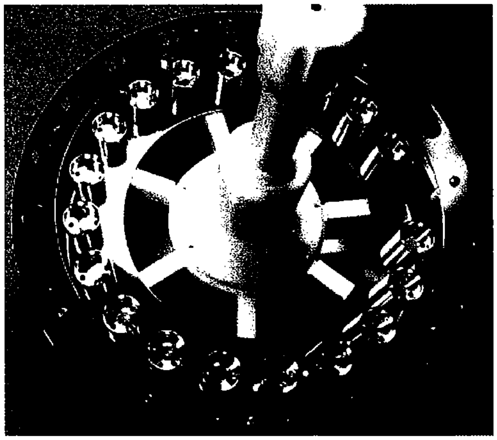


Figure 38 - Sample Transfer Station

Initializing the MarCSC

Via Software Interface

Normally the software is used to initialize the MarCSC. *(IMPORTANT: The user must first inspect the carousel and Dewar to make sure there are no loose vials, nor any vial in the Server Hand mechanism; note the sample number if any sample is currently mounted.)* In the Sample Changer tab in the *marccd* main window, click Initialize. If any sample is currently mounted, the correct sample number must be typed into the software question box. About 20-30 seconds are required to initialize all of the motors. Any currently mounted crystal should remain safely in the cold stream even during initialization.

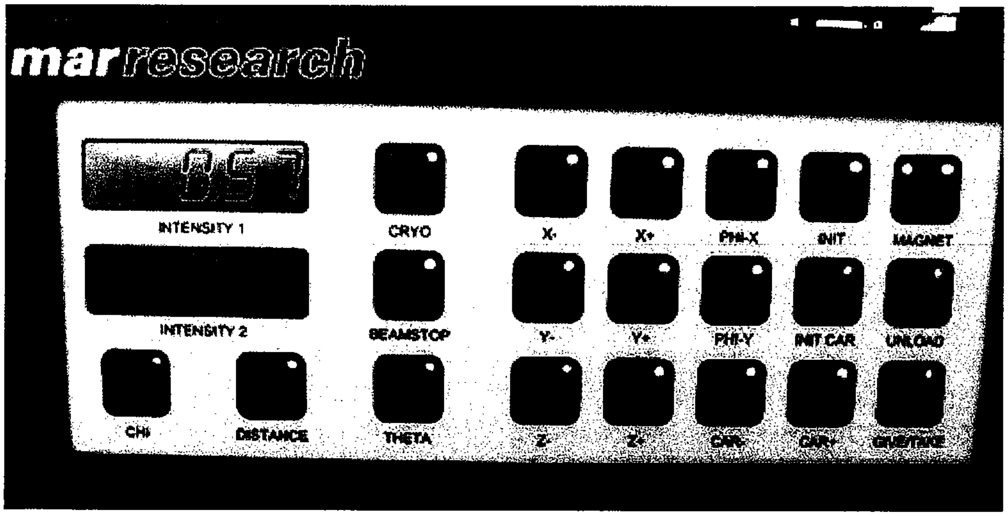
Via Local Remote

Copyright © 2000-2006, Mar USA

There is an Initialize button on the Local Remote box at the instrument. This feature does the same initialization procedure as the software initialize button with one important exception: Crystal-Z is also initialized; therefore, if any crystal is currently mounted, it will be driven out of the cold stream and melted!

Operation of the MarCSC

Local Remote Control



Manual Mode

Manual mode is used when mounting/dismounting samples *by hand* to the instrument. In manual mode, no automatic mounting or dismounting is possible, because the sample changer doesn't "know" which carousel position a mounted crystal came from. Pressing the "Magnet" button on the local remote control button will automatically switch the sample changer to Manual mode. Alternatively, it can be switched by the Manual/Auto Mode button in the Sample Changer tab in *marccd.*

Automatic Mode

The instrument must be in automatic mode in order for it to mount samples from the Carousel, dismount, or read barcodes. To switch from Manual to Automatic, the user must carefully determine which carousel position the mounted crystal (if any) should be returned to; if this mounted crystal is not to be returned to the carousel, it must be removed by hand first. The only way to switch to Automatic mode is the Manual/Auto Mode button in the Sample Changer tab in *marccd.*

Data Collection

Automatic Crystal Centering

The Algorithm Overview

Success and Failure

Copyright © 2000-2006, Mar USA

Video still images

Dataset crystal montage

Crystal Alignment - Specification of crystal centering image analysis API

This section will describe the crystal alignment algorithm, and how to design third-party image analysis software to interact with *marccd* for crystal alignment.

The program *marccd* controls the mounting of new crystals and all interactions with the goniostat and sample changer motors.

After each crystal is mounted, the crystal may need to be centered depending on the user's choice of none, manual, or auto centering. Centering the crystal is a process that must be accomplished in 3 dimensions. To accomplish 3-dimensional centering, *marccd* will use an internal algorithm that relies on acquiring a series of 2-dimensional images at different crystal orientations, and it will use an external, standalone program (`find_crystal`) to analyze the 2-D images and return the probable position of the crystal within each image. Optionally, `find_crystal` may also ask *marccd* to acquire images at additional crystal orientations by requesting Δx, Δy, Δz, Δø. With this option enabled, accomplishing the full 3-D centering of the crystal becomes the responsibility of `find_crystal`, rather than *marccd*. Note that, although `find_crystal` is passed a single 2-D image by *marccd*, it is not required to use that image in its analysis, and it may acquire data from other sources, including multiple images from more than one camera orientation. In theory, given enough data, `find_crystal` could accomplish the 3-D centering of the crystal without needing to reorient the crystal.

A list of configuration parameters for crystal centering is found in Table 8.

Table 8 - Configuration parameters for crystal centering

Parameter	Type	Default Value
crystal_alignment_automatic_enable	Boolean	false
crystal_alignment_find_crystal_command	String	find_crystal
crystal_alignment_video_image_filename	String	crystal.pgm
crystal_alignment_video_image_background_filename	String	empty.pgm
crystal_alignment_find_crystal_3D_enable	Boolean	false
crystal_alignment_supply_crystal_image	Boolean	true
crystal_alignment_supply_background_image	Boolean	true
crystal_alignment_supply_image_number	Boolean	false
crystal_alignment_supply_phi_position	Boolean	false
crystal_alignment_supply_video_parameters	Boolean	false
crystal_alignment_supply_crystal_number	Boolean	false
crystal_alignment_supply_crystal_id	Boolean	false
crystal_alignment_supply_segment_number	Boolean	false

Copyright © 2000-2006. Mar USA

crystal_alignment_use_background	Boolean	false
crystal_alignment_illuminate_background	Boolean	false
crystal_alignment_illuminate_crystal	Boolean	false
crystal_alignment_default_zoom	Float	1.0
crystal_alignment_default_brightness	Float	0.5
crystal_alignment_default_contrast	Float	0.4
video_still__type	String	pgm

Calling `find_crystal`

The program *marccd* will take a series of images at various crystal orientations. As each image is acquired, it will be written to the file `crystal.jpg` and then *marccd* will call the program *`find_crystal`* as follows:

```
find_crystal [-b empty_image] [-f crystal_image] [-n N] [-p
p.ppp] [-v f.ffff,s.ssss] [-s segment] [-c crystal] [-i
identifier]
```

-f empty_image	Name of captured video image (default: crystal.jpg)
-f crystal_image	Name of captured video image (default: crystal.jpg)
-n N	Image number – 0 indicates first image for new crystal
-v f.ffff,s.ssss	Video parameters. Size of image in fast and slow directions (mm)
-p p.ppp	Current phi position in degrees (at phi=0.0, Y motion is in video plane)
-s segment	dataset segment number (This will be 0 for single frame data)
-c crystal	crystal number (carousel position)
-i identifier	string crystal ID (barcode)

[] indicates optional parameter determined by configuration parameter values
The order of parameters is NOT defined by the above specification. Do not rely on the parameters appearing in the above order.

Note that the phi position sent to *`find_crystal`* may not be the true phi position, but is relative to the defined condition that phi is 0.0 when the Crystal-Y motion is parallel to the video plane.

Analysis by *`find_crystal`*

The program *marccd* will expect *`find_crystal`* to analyze the image to identify the center of the crystal and to return information on its standard output in the following form:

Copyright © 2000-2006, Mar USA

`type,fast,slow,2D_status[,delta_x,delta_y,delta_z,delta_phi ,3D_status]`

[] indicates optional data determined by `type`

2-Dimensional Analysis

The simplest form of *`find_crystal`* will perform a standalone analysis of each image to determine the position of the crystal center (or other position which is desired to be moved to the beam position). This is a 2-dimensional analysis and the result is a coordinate pair identifying that position in the image. The program *`find_crystal`* should set the type to 0 and return the fast, slow coordinate pair and a status indicator. If no position can be identified, then the status indicator should be –1. In that case, *marccd* will ignore the fast,slow coordinate and will make a decision about what to do next, which may be to try a different crystal position and repeat *`find_crystal`* on another image, or it may treat the crystal alignment as a failure and move on to the next crystal. If the analysis algorithm produces a coordinate that is at or beyond the edge of the image, then the status indicator should be set to 0 to indicate to *`marccd`* that another attempt to find the crystal should be made after adjusting the probable position toward the center. Note that the fast, slow coordinate pair are in fractional coordinates with the origin, as viewed on a normally oriented video screen, at the upper left. The normal range of coordinate is from 0.0 to 1.0, but note that this is not a requirement. *`Marccd`* will accept and act on a fractional coordinate in the range -1.0 to 2.0 so that *`find_crystal`* may specify a coordinate that is up to 1 field width beyond the edge of the video field.

For a 2-D *`find_crystal`*, `type` should be 0, and only 4 parameters are expected to be returned: `0`, `fast`, `slow`, and `2D_status`.

3-Dimensional Analysis

A more sophisticated form of *`find_crystal`* can perform a 3-dimensional analysis to determine the centering of the crystal. This requires that *`find_crystal`* be able to analyse multiple images acquired at various crystal positions (primarily with different ø positions). To accomplish this, *`find_crystal`* must be able to direct *`marccd`* to move the crystal. *Marccd* will call *`find_crystal`* once per crystal orientation and expect *`find_crystal`* to return instructions for reorienting the crystal for the next image. *Marccd* signals to *`find_crystal`* that a new crystal has been mounted by passing the arguments –N 0, and *marccd* expects *`find_crystal`* to signal the successful analysis of each image be returning a 3D status of 0 and to signal the successful full 3D alignment of the crystal be returning a positive 3D status.

Parameter	Data Type	Description

<table>
<tr><td>type</td><td>integer</td><td>Output type<table><tr><td>0</td><td>2-D only (3-D data is not written)</td></tr><tr><td>1</td><td>3-D only (2-D values must be written, but should be ignored)</td></tr><tr><td>2</td><td>2-D and 3-D data are valid</td></tr></table></td></tr>
<tr><td>fast</td><td>float</td><td>Fractional coordinate of crystal center in fast video direction</td></tr>
<tr><td>slow</td><td>float</td><td>Fractional coordinate of crystal center in slow video direction</td></tr>
<tr><td>2D_status</td><td>integer</td><td>Information about the result<table><tr><td>-1</td><td>Failure – fast and slow data should be ignored.</td></tr><tr><td>0</td><td>Guess – center is likely off-screen, but next image should bring it closer</td></tr><tr><td>1-3</td><td>Limited success - one or more coordinates may be unreliable - to be further defined in the future</td></tr><tr><td>4</td><td>Success, but Z coord is unreliable</td></tr><tr><td>5</td><td>Success</td></tr></table></td></tr>
<tr><td>delta_x</td><td>float</td><td>Requested change in crystal X position (mm)</td></tr>
<tr><td>delta_y</td><td>float</td><td>Requested change in crystal Y position (mm)</td></tr>
<tr><td>delta_z</td><td>float</td><td>Requested change in crystal Z position (mm)</td></tr>
<tr><td>delta_phi</td><td>float</td><td>Requested change in crystal ø position (degrees)</td></tr>
<tr><td>3D_status</td><td>integer</td><td>Information about the result of the 3-dimensional centering<table><tr><td>-1</td><td>Failure – give up on this crystal</td></tr><tr><td>0</td><td>In progress – follow directions for next orientation</td></tr><tr><td>1-5</td><td>Success indicator: 1 is worst, 5 is best. Finished with centering</td></tr></table></td></tr>
</table>

Copyright © 2000-2006, Mar USA

Chapter 9: Safety and System Operating Conditions

Safety Warnings

1. **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 (one exception is earlier-model MarCCD detectors that use internal vacuum valves—if no "Vacuum" port opening is available on back of detector head; for these detectors, precaution must be taken not to touch parts not related to the vacuum system).

For MarMosaic detectors, removing the front 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 Mar USA. If the window becomes dislodged for some reason, contact Mar USA immediately for advice about what to do next.

6. Refreshing the vacuum (MarCCD) or maintaining vacuum (MarMosaic)

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 MarCCD, during the periodic re-evacuation of the vacuum chamber (once every six months to one year), the procedure should be followed carefully, and the detector valve must be closed tightly before the hose is disconnected.

For the MarMosaic, 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 MarMosaic detector head.

7. MarDTB shutter

The MarDTB shutter should be considered to be a data-collection shutter only, not a safety shutter. A separate safety shutter should be used upstream from the instrument.

Operating Conditions

1. Electrical requirements

Table 9 – Total apparent power consumption, recommended circuit capacity, and cooling requirements for Mar X-ray Detectors. ***(Notes: MarMosaic 225, 300, and 325 cabinets are NOT wired internally for operation at 100-120V. MarMosaic 300 and 325 systems require two separate receptacles, although they may share one circuit breaker. Heat output of Mar345 estimated based on 80% exposure duty cycle—8.0 min exp, 1.5 min scan, and 0.5 min erase).***

Model	Maximum KVA	Max. Amps @100V	100-120V Circuit Capacity (Amps)		Max. Amps @200 V	200-240V Circuit Capacity (Amps)		Heat Output (Watts)	Heat Output (BTU/Hr)	Cooling Required (Tons)
Mar345	1060	10.6	20		5.3	10		400	2000	0.2
MarDTB	180	1.8	5		0.9	5		100	1000	0.1
MarCCD	810	8.1	15		4.1	10		900	4000	0.3
MarMosaic 225	1690	16.9	30		8.5	15		1700	6000	0.5
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total	2300	23.0	35	(30+10)	11.5	20	(15+5)	2300	8000	0.7
MarMosaic 300	3380	33.8	55	(2 x 30)	16.9	30	(2 x 15)	3400	12000	1.0
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total	3990	39.9	60	(30+40)	20.0	30	(15+20)	4000	14000	1.2
MarMosaic 325	3380	33.8	55	(2 x 30)	16.9	30	(2 x 15)	3400	12000	1.0
Pump	610	6.1	10		3.1	5		700	3000	0.3
Total	3990	39.9	60	(30+40)	20.0	30	(15+20)	4000	14000	1.2

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.

Copyright © 2000-2006, Mar USA

Appendix A: Installation Guide for *marccd*

Currently, *marccd* runs only on RedHat Linux and closely related distributions such as CentOS. As of the time of writing, Enterprise Linux 4 was the currently recommended distribution. Every attempt has been made to keep the functionality backwards-compatible to older distributions.

The hardware PCI card and Linux driver are not guaranteed to work on all variations of computer hardware (without major additional development effort by us). Therefore, we do not support migrating the PCI card and software to a computer workstation not purchased from Mar USA.

A site-specific and detector-specific CDROM is included with each detector. For a system that must be reinstalled from scratch (due to hard drive failure or major accidental deletion of files, for example), follow the instructions below to reinstall. In addition, the video4linux rpm packages and v4l X-windows module must be installed in order to view a video image of the crystal in the *marccd* software.

If a username "marccd" and directory /home/marccd/ already exist, then it is suggested to save the existing directory with a different name. Login as root and type "`mv /home/marccd/ /home/marccd_old`*."*

Software installation procedure:

- Insert the CDROM into the drive. (The serial number on the CDROM label must match the detector serial number.)
- Open a new terminal window and at the prompt type:

```
mount /mnt/cdrom
cd /mnt/cdrom
./install_marccd
```

- The installation script will prompt with certain questions. Accept default values by hitting the carriage return (recommended).
- In order for the driver for the PCI card to be installed, you must reboot the computer now.

Copyright © 2000-2006, Mar USA

Appendix B: Configuration Guide for *marccd*

Introduction

The appearance and available features of the program *marccd* are highly configurable. Running the same executable, options and controls can be very different depending on how parameters are defined in the configuration files. Parameters are set up by Mar USA when shipping a new detector, and only need slight modification to integrate with the user's existing equipment.

Configuration files

Note 1: The files described here are appropriate for marccd *versions >= 0.8.10. There was a significant change from version < 0.8.10!*

Note 2: ***Only the staff responsible for maintaining the detector should ever make changes to the configuration files! We strongly recommend making backup copies of files before editing them.*** *Do not edit files on Windows computers and then transfer them to Linux.*

Note 3: Files without explicit paths here are assumed to be in the configuration directory (environment variable MARCCDCONFIG).

The *marccd* program reads 3 configuration files in this order (if they are present):

• marccd.conf	The main configuration file
• [USER_HOME]/.marccdrc	The personal configuration file
• marccd_admin.conf	The administrator's configuration file

None of the files are required. If a file is not found, then it is skipped. A different main configuration file can be specified with the -c command line option. The personal configuration file can be skipped with the -n command line option.

Any files explicitly included in these files are read in order. The configuration files are searched for in the configuration directory, which is defined by the environment variable MARCCDCONFIG. The value can be overridden on the command line by the -C option. If the main configuration file is specified on the command line (-c) with a path, then it will not be searched for in the configuration directory.

Any file may include another file with the line,

```
include             filename
```

Includes can be nested.

The parameter format is one parameter name with one or more values to set, followed by a newline character. For example,

Copyright © 2000-2006. Mar USA

```
parameter_name      parameter_value
```

A parameter that is defined twice, or is defined in multiple included files, will get the value assigned from the last occurrence in the last file read. Unknown or mistyped parameter names are ignored. A line can be commented out with # character at the beginning of the line.

Normally, the marccd.conf file includes hardware-specific files for configuring the detector, goniostat, beamline and/or other hardware.

The following files are typically found in the configuration directory:

Category	**File name**	**Purpose**
Configuration	marccd.conf	The main configuration file
	marccdrc	A sample personal preferences file (to copy to ~/.marccdrc)
	marccd_admin.conf	The administrator's configuration file
Detector configuration	detector_none.conf	A dummy configuration for running marccd with no detector
	marccd_####.conf	If this is a MarCCD installation, this file will be present, containing all specific parameters for this detector. The #### refers to the four-digit detector serial number.
	marmosaic_####.conf	If this is a MarMosaic installation, this file will be present, containing all specific parameters for this detector. The #### refers to the four-digit detector serial number.
	marccd_G0xxxx.conf	A dummy G0 controller MarCCD configuration file
	marccd_G1xxxx.conf	A dummy G1 controller MarCCD configuration file
	marccd_G2xxxx.conf	A dummy G2 controller MarCCD configuration file
	marmosaic_3x3xxxx.conf	A dummy 3x3 mosaic configuration file
	marmosaic_4x4xxxx.conf	A dummy 4x4 mosaic configuration file
Goniostat configuration	goniostat_none.conf	A dummy configuration for running marccd with no goniostat. Distance is normally defined as present here; user can explicitly set distance in software between limits defined here. Shutter can be defined as present or not here.
	goniostat_sw.conf	Used with a software goniostat, i.e. marccd sends commands to control an external custom piece of equipment. This file defines which axes are controlled, limits, and units.
	mardtb_xxx.conf	configuration file for MarDTB serial number xxx
	marcsc_xxx.conf	configuration file for MarCSC serial number xxx
	marxy_xxx.conf	configuration file for MarDTB Self-Centering Phi Axis option, serial number xxx
	marbase.conf	configuration file for original marbase and controller

Copyright © 2000-2006, Mar USA

	marbase_lift.conf	configuration file for original marbase and controller with lift
Beamline configuration	synch-fixed.conf	configuration file for a fixed wavelength beamline
	synch-tunable_A.conf	configuration file for a tunable wavelength beamline, wavelength defined in Angstroms
	synch-tunable.conf	configuration file for a tunable wavelength beamline, wavelength replaced by Energy in KeV
	RA-Cu.conf	configuration file for a Cu Rotating Anode generator
Miscellaneous configuration	marccd_server.conf	Remote mode server parameters, protocol 0 (requires license key available from Mar USA)
	marccd_server_v1.conf	Remote mode server parameters, protocol 1 (requires license key available from Mar USA)
	marccd_server_esrf.conf	Remote mode server parameters for TACO device server at ESRF (requires license key available from Mar USA)
	frameshift.conf	Frameshift option parameters for MarCCD (requires license key available from Mar USA)
Detector dsp code (a.k.a. microcode)—defined in detector file	2541D.bin	binary microcode file for marccd 165 controllers with AIA interface
	3129D.bin	binary microcode file for marccd 165 controller with FO interface
	3565-.bin	binary microcode file for marmosaic detectors
Files for program mar345dtb only	config.xxx	configuration file mar345dtb program for mardtb serial number xxx
	MAR_DTB_NO	configuration for mar345dtb program containing mardtb serial number

Using *marccd* to control other motors

The program can also be made to control one or more other motors, or an entire goniostat.

The starting point for a primary software-controlled goniostat is to include the goniostat_sw.conf configuration file (as described above).

A software controlled secondary goniostat or additional motor(s) is called the external goniostat. These additional axes are flagged with the parameter [axis]_external_control.

See the following section about pseudoaxes, which is also applicable to goniostat axes.

Defining a pseudoaxis to do any software-controllable task

For nearly any task that can be controlled by a computer, *marccd* can be made to send those commands via a pseudo-axis and software script. This is useful for tasks that must be done automatically during data collection.

Suppose we have some Foo device attached to a computer somewhere, and a program foocheck is used to get the current value of the Foo device. That could be done during data collection by including the following file (by inserting a line "`include checkfoo.conf`" in an included file, probably marccd.conf).

Here are the contents of a text file that we will create, called checkfoo.conf:

```
# this config file creates a new pseudo axis
#

goniostat_external_paranoid      yes
#goniostat_external_command      motor_server_example.exp
goniostat_external_command       /opt/bin/foocheck.exp -A -B -C

auxillary_20_configurable             yes
auxillary_20_present                  yes
auxillary_20_display_name             CheckFoo
auxillary_20_internal_name            CheckFoo
auxillary_20_device_name              CheckFoo
auxillary_20_motor_group              2
#auxillary_20_unit_name
auxillary_20_motor_driven             yes
auxillary_20_movable                  yes
auxillary_20_forgetful                yes
auxillary_20_automation_enable        yes
auxillary_20_exposure_enable          yes
auxillary_20_scan_enable              no
auxillary_20_scannable                yes
auxillary_20_external_control         yes
#auxillary_20_external_encoder              no
#auxillary_20_has_home_1                    no
#auxillary_20_has_home_2                    no
#auxillary_20_minimum                       0.0
#auxillary_20_maximum                       2.5
auxillary_20_has_limit_minimum        no
auxillary_20_has_limit_maximum        no
auxillary_20_has_user_positions       no no no no
auxillary_20_user_positions           0.0 0.0 0.0 0.0
#auxillary_20_initialization_capable        no
#auxillary_20_initialization_enable         no
#auxillary_20_easy_init_enable              yes
auxillary_20_easy_set_enable          yes
auxillary_20_easy_drive_enable        yes
```

The new parameter CheckFoo will be handled by the external program as defined by goniostat_external_command. A column called CheckFoo will appear in the dataset menu, and the axis will also appear as a scannable axis. One would be able to set up a dataset which includes periodic calls to the foocheck.exp program defined by this file.

Copyright © 2000-2006. Mar USA

For the device control, the program foocheck.exp must be able to interpret the commands:

- marccd_move_abs (with arguments axis, position, speed)
- marccd_set_abs (with arguments axis, position)
- marccd_exposure (with arguments axis, width, time, expose)
- marccd_shutter (with argument "open" or "close")

The ".exp" file extension indicates that this is an Expect script. Other scripting or executable languages may be used.

Instead of creating a new script, an example script is already installed on your system. It contains the main function which any script needs, namely, it interprets the above commands. However, the sections for each command must have calls inserted to execute the desired functions. The example script is at ~/bin/linux/motor_server_example.exp.

Arguments follow each command, including move_to position, speed, etc. Even when a speed or position does not make sense to the call, it could be used an additional flag. In any case, the flags should be handled or disregarded as necessary by the program used to interpret the commands.

Adding the interleaved segments parameter with a line "`dataset_interleaved_segments  yes`" would allow datasets to be devised that have alternating data frames collected and foocheck calls.

List of configurable parameters

These parameters may need to be edited, but the best place for new parameters is a local default file called [MARCCD_HOME]/.marccdrc. Furthermore, editing the parameters in any configuration file should **only be done by the staff responsible for the detector**. *It is strongly recommended to make a backup copy of any configuration file before editing it.*

- **Configurable parameters for the general program**

Parameter name	**default**	**Type (Unit)**	**Explanation**
verbose	0	int	Verbosity level; each additional increment prints more to log and screen
shrink_rendering	BY_AVERAGING	int	Method of resizing data frame larger than main_image_width for display. Choices: BY_AVERAGING (requires more computation), BY_SUBSAMPLING (looks worse)
multithread_enable	1	int	Flag to allow multithreading
multithread_X11_enable	1	int	Flag to allow multithreading in X11
max_processors	0	int	Sets maximum CPUs that will be used for processing, even if they are available. Set 0 to use all available CPUs.
configuration_filename	(none)	string	Normally set to marccd.conf.

configuration_directory	defined by MARCCDCONFIG environment variable	string	Directory containing configuration files
configuration_basename	marccd	string	Base name of files
calibration_directory	defined by MARCCDCALIB environment variable	string	Directory containing calibration tables
log_directory	defined by MARCCDLOG environment variable	string	Log file directory
help_directory	defined by MARCCDHELP environment variable	string	Help and documentation directory
protocol_directory	defined by MARCCDPROTOCOLS environment variable	string	Directory containing protocol files (used by clicking "Load" in Acquire Dataset dialog)
skiprc	0	int	Flag to skip personal configuration (i.e. ~/.marccdrc)
frame_file_format	1	int	Sets default file format. Options: 1=FILE_FORMAT_NATIVE 2=FILE_FORMAT_FOREIGN_RAW 3=FILE_FORMAT_FOREIGN_XENTRONICS1 4=FILE_FORMAT_FOREIGN_XENTRONICS2 5=FILE_FORMAT_FOREIGN_FILMAN 6=FILE_FORMAT_FOREIGN_SIEMENS 7=FILE_FORMAT_FOREIGN_PHOTOMETRICS 8=FILE_FORMAT_FOREIGN_RAXIS2 9=FILE_FORMAT_FOREIGN_FUJI_PF 10=FILE_FORMAT_FOREIGN_FUJI 11=FILE_FORMAT_FOREIGN_IPLAB 12=FILE_FORMAT_FOREIGN_XEDAR 13=FILE_FORMAT_FOREIGN_MAR300 14=FILE_FORMAT_FOREIGN_TV6 15=FILE_FORMAT_FOREIGN_CSMA 16=FILE_FORMAT_FOREIGN_ADSC 17=FILE_FORMAT_FOREIGN_SPECTRAL 18=FILE_FORMAT_FOREIGN_FITS 19=FILE_FORMAT_FOREIGN_MAR345 20=FILE_FORMAT_FOREIGN_MAR_PROFILE 21=FILE_FORMAT_FOREIGN_PGM

frame_math_precision	3	int	Changes precision for image computations. Options: 1=PRESERVE_LSB 2=PRESERVE_MSB 3=PRESERVE_RANGE
main_window_width	128 or main_image_window_width+ 510	int (pixels)	width of main GUI window
main_window_height	128 or main_image_window_heigth+ 82 (+63 if view_mode)	int (pixels)	height of main GUI window
main_image_window_width	main_image_width+4	int (pixels)	width of main image display window
main_image_window_height	main_image_height+4	int (pixels)	height of main image display window
main_image_width	1024	int (pixels)	width of main image
main_image_height	1024	int (pixels)	height of main image
zoom_image_window_width	384	int (pixels)	width of zoom window
zoom_image_window_height	256	int (pixels)	height of zoom window
zoom_image_width	384	int (pixels)	width of zoom image
zoom_image_height	256	int (pixels)	height of zoom image
detector_type	DetectorNone	string	Options: DetectorNone, DetectorMarCCD, DetectorMarMosaic
detector_device	DetectorNone (not used)	string	Options: set to /dev/marCCD or /dev/rmarMosaic
detector_port	0	int	0 (port number not used for CCD detectors)
goniostat_type	GoniostatNone	string	Options: None, Unknown, Virtual, MMX, MMX0, MMX1 , MMX2, Mar, MarSCSI, Mar345V1, Mar345V2, MarNetV1, MarNetV2, MarDTB, SPEC, EPICS, TACO, ACE, GRACE, Remote, Software
goniostat_device	GoniostatNone	string	Name of goniostat for communication. Network host name “mardtb” for MarDTB.
goniostat_port	0	int	use 4451 for MarDTB, 4123 for marbase
correction_table_basename	ccdctb_	string	correction tables are named: basename+_[SN]_.[RES] where SN is the 4-digit serial number, RES is the 4-digit resolution

dataframe_correction_reorientation	R0	keyword	Options: R0, R90, R270, R180, M0, M45, M90, M135. Here, R[N] means rotate N degrees clockwise, and M[N] means mirror about a line at N degrees from horizontal
scratch_frame	none	string	Not Used
flatfield_basename	none	string	Not Used
postflatfield_basename	none	string	Only used for MarMosaic: ccdpff_
render_n_rank	1000	int	Number of divisions of scale for rendering
render_top_rank	997	int	Rank defined as highest intensity color in integrated histogram
render_bottom_rank	5	int	Rank defined as lowest intensity color in integrated histogram
value_display_format	Decimal	keyword	Pixel values for display on screen. Options: Decimal, Hexadecimal, Octal, Binary
coordinate_type_user	CoordsNative	keyword	Default display coordinates. Options: CoordsNative, CoordsAutomar, CoordsMosflm, CoordsDenzo, CoordsBuddha, CoordsXengen, CoordsCadmusPCS, CoordsSiemensPC, CoordsXedar
view_direction	FromSource	keyword	Default display orientation. Options: FromSource, TowardSource
color_scheme	BlackOnWhite	keyword	WhiteOnBlack, BlackOnWhite, WarmOnCool, Xentronics, Egelman, Caspar
color_scaling_method	LinearScaling	keyword	LinearScaling, LogScaling, CustomScaling, HistogramUniformScaling, HistogramGaussianScaling, HistogramCustomScaling
color_gamma	10.0	double	Default suggested gamma for Log scale contrast. Explained in Chapter 3.

Copyright © 2000-2006. Mar USA

display_source	MMXframeCorrectedDetector	int	Default frame displayed. Options include: MMXframeNone, MMXframeDisplayed, MMXframeDetector, MMXframeCorrectedDetector, MMXframeProcessing, MMXframeBackground, MMXframeFlatfield, MMXframeInvertedFlatfield, MMXframePostFlatfield, MMXframeInvertedPostFlatfield, MMXframeXCorrection, MMXframeYCorrection, MMXframeAreaCorrection, MMXframeCorrectionTable, MMXframeDarkOnly, MMXframeDarkStandard, MMXframeBias, MMXframeInputMask, MMXframeOutputMask, MMXframeDefectMap, MMXframeOverlapMap, MMXframeMean, MMXframeSigma, MMXframeStdErr, MMXframeNUsed, MMXframeCrystals0, MMXframeCrystals90, MMXframeScratch, MMXframeScratch1, MMXframeScratch2, MMXframeScratch3, MMXframeSystemScratch, MMXframeSystemScratch1, MMXframeSystemScratch2, MMXframePreCorrectBuffer0, MMXframePreCorrectBuffer1, MMXframePostCorrectBuffer0, MMXframePostCorrectBuffer1
goniostat_alignment_command	none	string	Alignment program launched by align button. Used to be mar345dtb, before beam alignment tab was developed in *marccd.*
goniostat_alignment_command_name	Align	string	Alignment command displayed on goniostat menu button
goniostat_alignment_command_delay	2.0	double (sec)	Delay before Alignment program launched (allowing time for *marccd* to shut down communication with goniostat)
intensity_counts_minimum	0	int (KHz)	If nonzero, this value is used in dataset dialog. If lower than minimum, warning occurs.

Copyright © 2000-2006, Mar USA

intensity_counts_maximum	0	int	If nonzero, this value is used in dataset dialog. If higher than maximum, warning occurs.
corrected_frame_bias	10	int	Bias added to corrected frames to preserve correct statistics
decimal_places_axis	3	int	Decimal places for display of linear motor values
decimal_places_angle	decimal_places_axis	int	Decimal places for display of angular motor values
decimal_places_wavelength	decimal_places_axis	int	Decimal places for display of wavelength
decimal_places_energy	decimal_places_axis	int	Decimal places for display of energy
decimal_places_exposure_time	3	int	Decimal places for display of exposure time
decimal_places_exposure_dose	decimal_places_exposure_time	int	Decimal places for display of exposure dose
punctuation_angle	0	int	Angle display. Choose 0 for decimal degrees, choose 1 for degrees and minutes.
background_update_interval	1800	int	general background update interval; can be overridden by specific parameters for single frame and dataset
single_frame_background_update_interval	background_update_interval	int	background update interval for single frame acquisition (when “Acquire Background” checkbox not checked)
default_filename_template		string	Not Used
default_filename_base		string	Not Used
keep_timing_statistics	0	int	Adding increments increases the number of various data collection timing statistics printed to log file and screen.
demonstration_mode	0	bool	flag to put in demonstration mode (prints fake Temp and Pressure values in Status even when not connected to a detector)
thumbnail_enable	0	int	Enables saving a thumbnail of every image in the dataset (Acquire Dataset dialog).
thumbnail_type	IMAGEFILE_TYPE_PGM	keyword	Default thumbnail type in dataset menu. Can by changed dynamically in menu. Keyword options same as image_type.
thumbnail_size	-1	int	Default maximum dimension for thumbnails saved in dataset menu (can be dynamically changed in menu)

Copyright © 2000-2006, Mar USA

image_type	IMAGEFILE_TYPE_PGM	keyword	Default image type for saving displayed 8-bit images (can be dynamically changed in File→ Save Displayed Image menu). Options are: IMAGEFILE_TYPE_PGM, IMAGEFILE_TYPE_RAW, IMAGEFILE_TYPE_TIFF, IMAGEFILE_TYPE_JPEG, IMAGEFILE_TYPE_GIF, IMAGEFILE_TYPE_PNG, IMAGEFILE_TYPE_EPS, IMAGEFILE_TYPE_MIFF, IMAGEFILE_TYPE_HTML, IMAGEFILE_TYPE_FITS
image_size	2048	int	Default maximum dimension for saving displayed 8-bit images (can be dynamically changed in File→ Save Displayed Image menu)
beam_coordinates	81.5 81.5	double (mm)	Two values for X and Y coordinates of beam (written to images headers for data processing)
beam_slope	0.0 0.0	double	Not Used
beam_reference_distance	100.0	double	Not Used
sidebar_tab_default	“Crystal Alignment”	keyword	Sets which tab is displayed by default when program is started (if tabs are configured to appear). Options are: “X-ray Data Detail” “Crystal Alignment” “Sample Changer”
continuous_mode_enable	no	bool	Enables continuous mode in the Acquire Single Frame dialog. If set “yes,” Continuous button appears, which can be used instead of “Start.” Images are collected continuously until stopped by user.
frame_display_enable	yes	bool	If set “no,” data frame display window disabled
view_mode	no	bool	If “yes,” program functions mostly for data frame viewing only, without goniostat or detector. Goniostat menu and status window are disables, as are many options.
expert_mode	no	bool	Not Used
administrator_mode	no	bool	Not Used
proprietary_mode	no	bool	Proprietary mode of the program. Can also be accessed with command line option. *Not recommended for default use because many detector parameters can be permanently changed.*
administrator_mode_license_key	xxx	string	Not Used
proprietary_mode_license_key	xxx	string	Not Used
acquire_enable	yes	bool	Setting “no” disallows all image acquisition.
acquire_dataset_enable	yes	bool	“Yes” allows Acquire Dataset dialog access.
acquire_single_enable	yes	bool	“Yes” allows Acquire Single Frame dialog access.

pipeline_buffers_n_read	0	int	Max number of read image buffers used in remote mode version 1 data handling. Set 0 for program default.
pipeline_buffers_n_holding	0	int	Max number of holding image buffers used in remote mode version 1 data handling. Set 0 for program default.
pipeline_buffers_n_corrected	0	int	Max number of corrected image buffers used in remote mode version 1 data handling. Set 0 for program default.
configure_enable	yes	bool	Setting “no” disallows Configure options in File menu (e.g. Configure→ Detector and Goniostat)

- **Configurable parameters for *marccd* appearance**

Note: The list of X11 color keywords recognized is normally in [X11root]/lib/X11/rgb.txt. To select an RGB color, precede the value by “#,” e.g. #FFFFFF.

Parameter name	**default**	**Type**	**Explanation**
warning_color	Goldenrod	X11 color or keyword	Status bar warning color
error_color	DarkRed	X11 color or keyword	Not Used
shutter_open_color	DarkRed	X11 color or keyword	Shutter status open color
shutter_closed_color	DarkGreen	X11 color or keyword	Shutter status closed color
go_color	DarkGreen	X11 color or keyword	Not Used
stop_color	Red	X11 color or keyword	Not Used
warning_tape_bright_color	Yellow	X11 color or keyword	Not Used
warning_tape_dark_color	Black	X11 color or keyword	Not Used
progress_bar_trough_color	White	X11 color or keyword	Progress bar color
progress_bar_bar1_done_color	Blue	X11 color or keyword	Not Used
progress_bar_bar2_done_color	Red	X11 color or keyword	Not Used
progress_bar_bar3_done_color	Green	X11 color or keyword	Not Used
progress_bar_bar4_done_color	Violet	X11 color or keyword	Not Used
saturated_color	Red	X11 color or keyword	Color of saturated pixels in images displayed on screen
bubble_help_background_color	LightGoldenRod	X11 color or keyword	Bubble help background color
bubble_help_foreground_color	Black	X11 color or keyword	Bubble help text color
pending_segment_background_color	LightBlue	X11 color or keyword	Dataset pending segment background
pending_segment_foreground_color	Black	X11 color or keyword	Dataset pending segment text
unused_segment_background_color	LightGray	X11 color or keyword	Dataset unused segment background
unused_segment_foreground_color	Black	X11 color or keyword	Dataset unused segment text
skipped_segment_background_color	PaleVioletRed	X11 color or keyword	Dataset skipped segment background

skipped_segment_foreground_color	Black	X11 color or keyword	Dataset skipped segment text
inprogress_segment_background_color	Goldenrod	X11 color or keyword	Dataset in-progress segment background
inprogress_segment_foreground_color	Black	X11 color or keyword	Dataset in-progress segment text
finished_segment_background_color	LightGreen	X11 color or keyword	Dataset finished segment background
finished_segment_foreground_color	Black	X11 color or keyword	Dataset finished segment text
error_segment_background_color	OrangeRed	X11 color or keyword	Dataset error segment background
error_segment_foreground_color	Black	X11 color or keyword	Dataset error segment text
video_cursor_beam_color	DarkRed	X11 color or keyword	Color of beam crosshair overlay on video image
video_cursor_crystal_color	DarkBlue	X11 color or keyword	Color of crystal crosshair overlay on video image

- **Configurable parameters for dataset data collection (Acquire Dataset dialog)**

Parameter	**Default**	**Type (Unit)**	**Explanation**
dataset_interleaved_segments	no	bool	Setting "yes" allows interleaved segments (Nsegs and Size columns will appear in dataset menu).
dataset_resolution_option	no	bool	Setting "yes" allows binning column in dataset menu. Each segment has selectable binning.
dataset_rotation_gap_option	no	bool	Setting "yes" allows rotation gaps in dataset menu. A rotation gap column is created.
dataset_still_exposure_option	no	bool	Setting "yes" allows still exposures in dataset menu under "Motion" column
dataset_dark_exposure_option	no	bool	Setting "yes" allows dark exposures in dataset menu under "Motion" column
dataset_summarize_segments_option	no	bool	Not Used
dataset_multicrystal_enable	no	bool	Allows multicrystal data collection in dataset (for MarCSC, should be "yes")
dataset_max_crystals	-1	int	Maximum number of crystals for data collection menu (for MarCSC, should be 19)
dataset_max_segments	8	int	Max number of segments in dataset menu
dataset_max_periodic_commands	1	int	Max number of periodic commands in dataset menu.
dataset_background_update_interval	-1	int (sec)	Background update interval in seconds for dataset data collection. Set to -1 to make update interval once per segment.
dataset_video_still_enable	yes (if video_input	bool	Enables video still capture option in dataset menu.

	_enable set)		
dataset_warn_nounload	no	bool	Warns if last crystal will in dataset will not be unloaded (unload_last_sample protocol parameter set).
dataset_project_id_enable	yes	bool	Enables project ID field in dataset menu. Used at EMBL for keeping track of authorized projects.
dataset_authorization_enable	no	bool	Enables authorization at EMBL for keeping track of authorized projects.
dataset_authorization_require	no	bool	Forces project ID field to be filled with an approved project ID in dataset menu, or data will no be collected. Used at EMBL for keeping track of authorized projects.
dataset_authorization_edit_enable	no	bool	Enables editing project ID field in dataset menu. Used at EMBL for keeping track of authorized projects.
dataset_authorize_command	none	string	Sets command for asking if project ID field in dataset menu is authorized. Used at EMBL for keeping track of authorized projects.

- **Configurable parameters for remote mode**

Parameter name	**Default**	**Type (Unit)**	**Explanation**
remote_mode_version	0	int	Options are 0 and 1. Version 1 uses more complete status information and allows thumbnails. See Appendix D.
remote_mode_server_environment	same as NETHOST environment variable	string	Remote mode server environment
remote_mode_server_command	same as DSSERVER environment variable	string	Remote mode server program, including path; executed by *marccd* when remote mode entered
remote_mode_server_arguments	same as DSPERSONALNAME environment variable	string	Arguments to remote mode server program. Usually just the port number (must match the port specified by client to connect).
remote_mode_server_log	blank	string	Not Used – Future development may include a server log
remote_mode_server_environment_label	Server Environment	string	Server environment text label in remote mode dialog box
remote_mode_server_command_label	Server Command	string	Server command text label in remote mode dialog box
remote_mode_server_arguments_label	Server Arguments	string	Server arguments text label in remote mode dialog box

Copyright © 2000-2006, Mar USA

remote_mode_server_log_label	Server Log	string	Server log text label in remote mode dialog box
remote_mode_license_key	xxx	string	A license key obtained from Mar USA to operate in remote mode.
acquire_remote_enable	no	bool	Must be set "yes" to use remote mode

- **Configurable parameters for MarCCD detectors**

Parameter	**Default**	**Type (Unit)**	**Explanation**
detector_serial_number	0	int	Must correspond to detector serial number
ccd_controller_type	none	keyword	Must correspond to controller type. Options: Spectral600, Spectral600_G2, Spectral600_G2_A, Spectral600_G2_B.
ccd_sensor_type	none	keyword	Must correspond to CCD type. Options: SiTE424, LM485, FI447, EEV4240
ccd_hardware_serial_configuration	0 0 0 0 0	int array	Array of 5 integers: serial configuration, serial origin, serial length, serial postscan, max serial binning
ccd_hardware_parallel_configuration	0 0 0 0 0	int array	Array of 5 integers: parallel configuration, parallel origin, parallel length, parallel postscan, max parallel binning
ccd_hardware_readout_confuration	0x0	hex int	Native readout configuration for four channels. Channels A, B, C, D assigned bits 0, 1, 2, 3, respectively. Therefore, add numbers 1, 2, 4, 8 for each desired channel. 3-channel readout not possible, and some 2-channel combinations not supported.
ccd_serial_configuration	0 0 0 0 0	int array	Array of 5 integers: serial configuration, serial origin, serial length, serial postscan, serial binning
ccd_parallel_configuration	0 0 0 0 0	int array	Array of 5 integers: parallel configuration, parallel origin, parallel length, parallel postscan, parallel binning
ccd_readout_configuration	0x0	hex int	Default configuration for four channels. Channels A, B, C, D assigned bits 0, 1, 2, 3, respectively. Therefore, add numbers 1, 2, 4, 8 for each desired channel. 3-channel readout not possible, and some 2-channel combinations not supported.
ccd_pixelsize	0.0	double (μm)	Pixel size of the unbinned readout mode (not necessarily supported mode of readout). Can be an array of 2 doubles (serial direction pixelsize first, then parallel)
ccd_readout_speeds	0	int	Number of readout speeds available (up to 4)
ccd_readout_config_1	0 0 0	int array	Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 384615 1 4

ccd_readout_config_2	0 0 0	int array	Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 192315 1 1
ccd_readout_config_3	0 0 0	int array	Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 96155 1 1
ccd_readout_config_4	0 0 0	int array	Factory setting. Not used unless number of readout speeds is 4.
ccd_readout_config_default	0 0 0	int	Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 384615 1 4
ccd_offsets_analog	0 0 0 0	int array	Analog offsets applied to the four quadrants (ADC input voltage offset) in arbitrary units. Example: 1200 1200 1200 1200
ccd_offsets_corner	0 0 0 0	int (ADU)	Digital offsets applied to the four quadrants by software, after analog to digital conversion and transfer to computer. Example: 942 655 821 834
ccd_offsets_adc	0 0 0 0	int	Digital offsets from analog converter only.
ccd_relative_preamp_gains	1.0 1.0 1.0 1.0	double array	Not yet implemented.
ccd_significant_bits	0	int	Significant bits of ADC. Normally 16.
ccd_parallel_shift_delay	0	int (ms)	Factory-determined parallel shift timing of CCD.
ccd_shutter_close_delay	0	int (ms)	Delay of readout after hardware shutter close signal sent. Should be increased for slow-closing shutters (if *marccd* controls the shutter)
ccd_external_trigger	0	int	Not yet implemented. TTL-triggered “bulb” mode.
ccd_baseline_raw	0	int	Raw frame baseline value target achieved during balance operation.
ccd_saturation_level	0	int (ADU)	Signal level at which saturation occurs
ccd_saturation_flag	0	int (ADU)	Pixels at saturation_level are replaced by this flag value, normally 65535 for 16-bit
ccd_temperature_setpoint	0	double (K)	Factory-determined CCD operating temperature
ccd_temperature_offset	0.0	double (K)	Not yet implemented.
ccd_microcode_file	none	string	CCD dsp code for detector operation (must be in configuration directory)
frameshift_mode_enable	no	bool	Setting “yes” may require license key
frameshift_single_file		bool	Not Used
frameshift_lines	0	int	Default number of lines for frameshift, if enabled. May be changed by user in Acquire Dataset and Single Frame dialogs.
frameshift_mode_license_key	none	string	License key to operate frameshift mode

- **Configurable parameters for MarMosaic detectors**

Parameter	**Default**	**Type (Unit)**	**Explanation**
detector_serial_number	0	int	Must correspond to detector serial number
ccd_controller_type		string	Not Used (for MarMosaic)
ccd_sensor_type	none	string	Must correspond to CCD type. Options: SiTE424, LM485, FI447, EEV4240
ccd_hardware_serial_configuration	0 0 0 0 0 0	int array	Array of 6 numbers for native hardware configuration: serial configuration, serial prescan, serial origin, serial length, serial postscan, serial binning
ccd_hardware_parallel_configuration	0 0 0 0 0 0	int array	Array of 6 numbers for native hardware configuration: parallel configuration, parallel prescan, parallel origin, parallel length, parallel postscan, parallel binning
ccd_hardware_readout_configuration	0x0	hex int	Native readout configuration on each CCD chip. Four channels A, B, C, D assigned bits 0, 1, 2, 3, respectively. Therefore, add numbers 1, 2, 4, 8 for each desired channel. 3-channel readout not possible, and some 2-channel combinations not supported.
ccd_hardware_sensors_fast_slow	0 0	int array	Native size of CCD sensor array, in fast direction and slow direction
ccd_hardware_sensor_rows	0x0 0x0 0x0 0x0	hex int array	Hex number represents native CCD sensors present in the array, enumerated by bit value. Each successive number is a row in the slow direction. E.g.: 3x3 array with all sensors present should have "0x7 0x7 0x7."
ccd_serial_configuration	0 0 0 0 0 0	int array	Array of 6 numbers for default configuration: serial configuration, serial prescan, serial origin, serial length, serial postscan, serial binning
ccd_parallel_configuration	0 0 0 0 0 0	int array	Array of 6 numbers for default configuration: parallel configuration, parallel prescan, parallel origin, parallel length, parallel postscan, parallel binning
ccd_readout_configuration	0x0	int	Default readout configuration on each CCD chip. Four channels A, B, C, D assigned bits 0, 1, 2, 3, respectively. Therefore, add numbers 1, 2, 4, 8 for each desired channel. 3-channel readout not possible, and some 2-channel combinations not supported.
ccd_sensors_fast_slow	0 0	int	Default size of CCD sensor array, in fast direction and slow direction
ccd_sensor_rows	0x0 0x0 0x0 0x0	int	Hex number represents default CCD sensors present in the array, enumerated by bit value. Each successive number is a row in the slow direction. E.g.: 3x3 array with all sensors present should have "0x7 0x7 0x7."

ccd_pixelsize	0.0	double (μm)	Pixel size of the default readout mode. Can be an array of 2 doubles (serial direction pixelsize first, then parallel)
ccd_dsi_sample_time	0	int	Factory set DSI sample time; affects readout speed and gain
ccd_readout_speeds	0	int	Number of readout speeds available (up to 4)
ccd_readout_config_1	0 0 0	int	(Currently for MarCCD only) Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 384615 1 4
ccd_readout_config_2	0 0 0	int	(Currently for MarCCD only) Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 192315 1 1
ccd_readout_config_3	0 0 0	int	(Currently for MarCCD only) Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 96155 1 1
ccd_readout_config_4	0 0 0	int	(Currently for MarCCD only) Factory setting. Not used unless number of readout speeds is 4.
ccd_readout_config_default	0 0 0	int	(Currently for MarCCD only) Factory setting: clock speed (in Hz), gain 1, gain 2. E.g. 384615 1 4
ccd_offsets_analog	0 0 0 0	int	(Currently for MarCCD only) Analog offsets applied to the four quadrants (ADC input voltage offset) in arbitrary units. Example: 1200 1200 1200 1200
ccd_offsets_corner	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	int array	(Currently for MarCCD only) Digital offsets applied to the four quadrants by software, after analog to digital conversion and transfer to computer.
ccd_offsets_adc	0 0 0 0	int array	(Currently for MarCCD only) Digital offsets from analog converter only.
ccd_relative_preamp_gains	1.0	double array	Not yet implemented.
ccd_significant_bits	0	int	Significant bits of ADC. Normally 16.
ccd_parallel_shift_delay	0	int	Factory-determined parallel shift timing of CCD.
ccd_shutter_close_delay	0	int	Delay of readout after hardware shutter close signal sent. Should be increased for slow-closing shutters (if *marccd* controls the shutter)
ccd_external_trigger	0	int	Not yet implemented. TTL-triggered “bulb” mode.
ccd_baseline_raw	0	int (ADU)	Raw frame baseline value target
ccd_saturation_level	0	int (ADU)	Signal level at which saturation occurs
ccd_saturation_flag	0	int (ADU)	Pixels at saturation_level are replaced by this flag value, normally 65535 for 16-bit
ccd_temperature_setpoint	0.0	double (K)	Factory-determined CCD operating temperature

ccd_temperature_offset		double (K)	Not Used
ccd_cooler_setpoint	-273.0	double (°C)	(MarMosaic only) Temperature set point of cold head heater
ccd_cooler_offset		double	Not Used
ccd_microcode_file	none	string	CCD dsp code for detector operation (must be in configuration directory)

- **Configurable parameters for all goniostats**

Parameter	**Default**	**Type (Unit)**	**Explanation**
goniostat_device_name	none	string	Goniostat device name
goniostat_logical_name	none	string	Goniostat logical name (Not Used)
goniostat_port	0	int	Goniostat port
goniostat_type_name	none	string	Goniostat type name
goniostat_parameter_file_name	none	string	Goniostat parameter file name
omega_theta_minimum	0.0	double	Anti-collision parameter, if software-controlled Kappa axis existed (Not Used)
omega_theta_maximum	0.0	double	Anti-collision parameter, if software-controlled Kappa axis existed (Not Used)
goniostat_n_detectors	1	int	Goniostat number of detectors (Not Used)
motor_group_name_0	Goniostat	string	Tab name for motor group 0
motor_group_name_1	Crystal	string	Tab name for motor group 1
motor_group_name_2	Beam	string	Tab name for motor group 2
motor_group_name_3	Alignment	string	Tab name for motor group 3
motor_group_name_4	SC	string	Tab name for motor group 4
user_position_name_0	Load	string	Defines button name for first button under Goniostat Tab. Button used for sending preset positions, [axis]_user_positions (only if [axis]_has_user_positions is set "yes").
user_position_name_1	User 1	string	Defines button name for second button under Goniostat Tab. Button used for sending preset positions, [axis]_user_positions (only if [axis]_has_user_positions is set "yes").
user_position_name_2	User 2	string	Defines button name for third button under Goniostat Tab. Button used for sending preset positions, [axis]_user_positions (only if [axis]_has_user_positions is set "yes").
user_position_name_3	User 3	string	Defines button name for fourth button under Goniostat Tab. Button used for sending preset positions, [axis]_user_positions (only if [axis]_has_user_positions is set "yes").
[axis]_display_name	none	string	Display name for [axis]
[axis]_device_name	none	string	Device name for [axis]
[axis]_internal_name	none	string	Internal name for [axis]

[axis]_unit_name	none	string	Parameter unit label for [axis]
[axis]_configurable		bool	Flag allows configurability for [axis]
[axis]_present	no	bool	Flag indicates [axis] is present
[axis]_toggle		bool	Flag indicates [axis] is a toggle-state device
[axis]_motor_driven		bool	Flag indicates [axis] is motor-driven
[axis]_movable		bool	Flag indicates [axis] is movable
[axis]_exposure_enable		bool	Flag indicates [axis] is exposure enabled
[axis]_scan_enable		bool	Flag indicates [axis] is scan enabled
[axis]_oscillation_enable		bool	Flag indicates [axis] is oscillation enabled
[axis]_external_control		bool	Flag indicates [axis] is part of external (secondary software controlled) goniostat
[axis]_has_home_1		bool	Flag indicates [axis] has home 1
[axis]_has_home_2		bool	Flag indicates [axis] has home 2
[axis]_has_limit_minimum		bool	Flag indicates [axis] has limit minimum
[axis]_has_limit_maximum		bool	Flag indicates [axis] has limit maximum
[axis]_default_home		int	Default home. Options: 1 = DRIVE_TO_HOME1 rising edge -1 = DRIVE_TO_HOME1 falling edge 2 = DRIVE_TO_HOME2 rising edge -2 = DRIVE_TO_HOME2 falling edge 3 = DRIVE_TO_LIMIT1 4 = DRIVE_TO_LIMIT2 5 = DRIVE_TO_MIN 6 = DRIVE_TO_MAX 7 = DRIVE_DOWN 8 = DRIVE_UP
[axis]_position		double	Axis current position
[axis]_has_user_positions MAX_USER_POSITIONS	no	bool	If set "yes" then the four buttons under Goniostat Tab will send [axis]_user_positions when clicked.
[axis]_user_positions MAX_USER_POSITIONS		double	Defines preset positions that will be sent to Goniostat Tab, if this axis has [axis]_users_positions set "yes."
[axis]_minimum		double	Software maximum for [axis]
[axis]_maximum		double	Software minimum for [axis]
[axis]_home_1		double	Home 1 position (used when home switch used instead of limit switch for homing)
[axis]_home_2		double	Home 2 position (used when second home switch used instead of limit switch for homing)
[axis]_limit_minimum		double	Limit minimum position (when limit switch used as home position)
[axis]_limit_maximum		double	Limit maximum position (when limit switch used as home position)
[axis]_external_encoder		bool	Not Used
[axis]_steps_per_drive_rev		double	Steps per drive revolution
[axis]_pulses_per_step		int	Pulses per step (for stepper motors that use microsteps)
[axis]_drive_rev_per_axis_rev		double	Drive revolutions per axis revolution
[axis]_antibacklash_size		double	Backlash size

[axis]_antibacklash_speed		double	Backlash speed
[axis]_deadband		double	Deadband (smallest measurable move distance)
[axis]_offset		double	Not Used
[axis]_speed		double	Default speed in position direction
[axis]_speed_negative		double	Default speed in negative direction
[axis]_speed_adjust		double	Parameter to correct speed if controller clock different than computer clock
[axis]_speed_minimum		double	Minimum allowed speed
[axis]_speed_maximum		double	Maximum allowed speed
[axis]_exposure_minimum		double	Minimum exposure scan time, if any
[axis]_exposure_maximum		double	Maximum exposure scan time, if any
[axis]_exposure_speed_minimum		double	Minimum exposure scan speed, if any
[axis]_exposure_speed_maximum		double	Maximum exposure scan speed, if any
[axis]_scan_speed		double	Default scan speed
[axis]_scan_minimum		double	Minimum scan distance, if any
[axis]_scan_maximum		double	Maximum scan distance, if any
[axis]_scan_speed_minimum		double	Minimum scan speed, if any
[axis]_scan_speed_maximum		double	Maximum scan speed, if any
[axis]_acceleration		double	Acceleration
[axis]_reference_speed_1		double	Speed when homing
[axis]_reference_speed_2		double	Speed when homing
[axis]_reference_acceleration		double	Deceleration when hitting limit switch
[axis]_reference_timeout		double	Timeout for homing
[axis]_automation_enable		bool	Flag enables automated movement during dataset. Puts column in Dataset Dialog.
[axis]_motor_group		int	Motor group assignment
[axis]_continuous_move_enable		bool	Flag to make arrow buttons for continuous move available in goniostat menu
[axis]_continuous_move_acceleration_factor		double	Holding arrow button for more than two seconds boosts to second speed (by this factor).
[axis]_initialization_capable		bool	Flag that indicates [axis] can be initialized
[axis]_initialization_enable		bool	Not Used / Obsolete
[axis]_easy_init_enable		bool	Flag to make "I" initialize button available in goniostat menu
[axis]_easy_scan_enable		bool	Not Used
[axis]_easy_set_enable		bool	Flag to make "S" set button available in goniostat menu
[axis]_easy_drive_enable		bool	Flag to make "D" drive button available in goniostat menu
[axis]_forgetful		bool	Flag to make last axis position stored in goniostat ".dat" file. Position is recalled when restarting software.
[axis]_reserved		int	Not Used
goniostat_reserved		int	Not Used

- **Configurable parameters to control an external (i.e. secondary) goniostat**

Copyright © 2000-2006, Mar USA

Parameter	**Default**	**Type (Unit)**	**Explanation**
goniostat_external_command	none	string	Goniostat external command name (if any). The path must be included, or else in the shell path. The program can be made to understand the following argument commands: • marccd_move_abs • marccd_set_abs • marccd_exposure • marccd_shutter
goniostat_external_group_commands	no	bool	When multiple external goniostat axes are to be driven simultaneously, the commands are all sent on the same command line.
goniostat_external_paranoid	yes	bool	For any axis with [axis]_external_control set "yes," the program does not assume the position remains constant (e.g. it could be moved by other program). Move commands are sent even when the last known position by *marccd* is the same as the next target position.

- **Configurable parameters to control a software goniostat**

Parameter	**Default**	**Type (Unit)**	**Explanation**
goniostat_software_command		string	Goniostat software command name (if any). The path must be included, or else in the shell path. The program can be made to understand the following argument commands: • marccd_move_abs • marccd_set_abs • marccd_exposure • marccd_shutter
goniostat_software_read_positions	no	bool	Not Used
goniostat_software_group_commands	no	bool	When multiple software goniostat axes are to be driven simultaneously, the commands are all sent on the same command line.
goniostat_software_paranoid		bool	For any axis in the software goniostat, the program does not assume the position remains constant (e.g. it could be moved by other program). Move commands are sent even when the last known position by *marccd* is the same as the next target position.
goniostat_software_log_commands	no	bool	Not Used
goniostat_software_log_responses	no	bool	Not Used

Copyright © 2000-2006, Mar USA

goniostat_software_block_on_move	yes	bool	Setting "no" disables *marccd* waiting for software command to exit before continuing. Software goniostat will be assumed to be finished with command immediately (caution using this option).
goniostat_software_expose_synchronizes_axis	no	bool	Default "no" setting means that after exposure, a software goniostat scanned axis is assumed to be at exactly the scan end position.
goniostat_software_expose_requires_axis_position	no	bool	Puts extra axis position parameter on command line in marccd_expose command.

- **Configurable parameters specific to the MarDTB**

The following chart shows parameters normally found in the [MARCCD_HOME]/configuration/mardtb_[SN].conf file (where [SN] refers to a three-digit serial number of the goniostat).

Parameter name	**Default**	**Type (Unit)**	**Explanation**
beam_alignment_window_enable	yes	bool	If set to "no" then there will be no Beam Alignment tab in main *marccd* window
beam_alignment_slit1hori	0.6	double (mm)	Defines horizontal slit 1 width for automatic beam optimize procedures; needs to be adjusted for beam size. Setting less than 0.05 mm is not recommended at this time.
beam_alignment_slit1vert	0.6	double (mm)	As above, vertical slit 1 width
beam_alignment_slit2hori	0.55	double (mm)	As above, horizontal slit 2 width
beam_alignment_slit2vert	0.55	double (mm)	As above, vertical slit 2 width
default_beam_intensity_offset_1	20	int (adc units)	The default offsets 1 and 2, and sigmas 1 and 2 are values suggested by the program to the user and should be approximately correct values; **slightly high estimates are safer**. Synchrotron values are typically 10-20 (set when the gain is set low); lab sources have offsets of 100-1000 (set when the gain is set high).
default_beam_intensity_sigma_1	2	int (adc units)	See default_beam_intensity_offset_1 explanation above. Synchrotron values are typically in the range 2-4; lab sources typically in the range of 20-40.
default_beam_intensity_offset_2	20	int (adc units)	See default_beam_intensity_offset_1 explanation above.

default_beam_intensity_sigma_2	2	int (adc units)	See default_beam_intensity_sigma_1 explanation above.
beam_intensity_offset_timeout	1800	int (sec)	Beam alignment algorithm requires measurement or typing of new offset and sigma values if this time has passed since last measurement/typing. Setting to 0 causes the timeout to be infinite, so the algorithm will never require new offsets or sigmas (not recommended).
beam_intensity_offset_prompt_enable	yes	bool	Setting to "no" will eliminate prompting the user to type in offsets and sigmas; the algorithm will measure them automatically, if needed during alignment.
beam_alignment_ignore_errors	no	bool	If set to no (default) the software will detect goniostat errors, such as hitting hardware limit switches, and report in a yellow display box. If the goniostat motors are properly initialized and have the correct values in the configuration file, then this should not happen. To disable error reporting, set this to yes.
dataset_alignment_enable	no	bool	Enables alignment features in the dataset dialog.
dataset_alignment_find_enable	no	bool	If dataset_alignment_enable is set, this feature also enables Find function in addition to Optimize. For particularly unstable beams (if beam position changes by the beam size or greater), set to "yes." Search for beam within maximum slit setting.
beam_alignment_peak_fraction	0.5	double (fraction 0.0-1.0)	This is the "scan to fraction" for individual scans. A peak is considered scanned over if the intensity first rises and then falls to this fraction of maximum; then the program will stop and evaluate the result.
beam_alignment_ignore_errors	no	bool	Setting "yes" turns off MarDTB error checking during scans. If set "no," error checks are done, and errors found, such an axis hitting a limit switch, result in yellow warning messages shown.
beam_alignment_axis_correction_th	0.0	double (mm)	After beam alignment Optimize finishes, a final correction move may be applied with this and following 7 parameters. This is optional horizontal translation correction.
beam_alignment_axis_correction_tv	0.0	double (mm)	Optional vertical translation correction applied after finishing every Optimize.
beam_alignment_axis_correction_rh	0.0	double (deg)	Optional vertical translation correction applied after finishing every Optimize.
beam_alignment_axis_correction_rv	0.0	double (deg)	Optional vertical translation correction applied after finishing every Optimize.

beam_alignment_axis_correction_s1v	0.0	double (mm)	Optional slit 1 vertical correction applied after finishing every Optimize.
beam_alignment_axis_correction_s1h	0.0	double (mm)	Optional slit 1 horizontal correction applied after finishing every Optimize.
beam_alignment_axis_correction_s2v	0.0	double (mm)	Optional slit 2 vertical correction applied after finishing every Optimize.
beam_alignment_axis_correction_s2h	0.0	double (mm)	Optional slit 2 horizontal correction applied after finishing every Optimize.
dataset_beam_alignment_enable	no	bool	Enables option to align beam in Acquire Dataset dialog. See Chapter 5.
dataset_beam_alignment_find_enable	no	bool	If dataset beam alignment already enabled, setting this parameter "yes" allows Find procedure as well as Optimize procedure for beam alignment. Useful if beam expected to often change position by greater than the default slit widths.
dataset_beam_alignment_retry_interval	600	int (sec)	When doing a dataset beam alignment, if the alignment fails (i.e. no beam found at end of alignment procedure), then the dataset will not continue. This is the number of seconds that the program will wait before collecting trying beam alignment procedure again.
beam_param_rotv_cntr_slit1v_d	4.0	double (mm)	The distance along beam axis from the center of vertical rotation to the slit 1 vertical (ideal case is 0; in MarDTB it is known to be 4.0mm) Used for movement corrections during alignment.
beam_param_rotv_cntr_slit2v_d	84.0	double	The distance along beam axis from the center of vertical rotation to the slit 2 vertical (in MarDTB it is known to be 84.0mm) Used for movement corrections during alignment.
beam_param_roth_cntr_slit1h_d	-6.0	double	The distance along beam axis from the center of horizontal rotation to the slit 1 horizontal (ideal case is 0; in MarDTB it is known to be -6.0mm) Used for movement corrections during alignment.
beam_param_roth_cntr_slit2h_d	74.0	double	The distance along beam axis from the center of horizontal rotation to the slit 2 horizontal (in MarDTB it is known to be 74.0mm) Used for movement corrections during alignment.
beam_simulation_enable	no	bool	Enables simulating a beam from a previously recorded beam profile file. Button and checkbox are produced in the GUI when "yes." Used for demonstration or development only.

beam_simulation_xt_offset	0.0	double	Horizontal translation offset of simulated beam (relative to the actual offset recorded in the profile file header)
beam_simulation_yt_offset	0.0	double	Vertical translation offset of simulated beam (relative to the actual offset recorded in the profile file header)
beam_simulation_xr_offset	0.0	double	Horizontal rotation offset of simulated beam (relative to 0 position of axis). Not tested.
beam_simulation_yr_offset	0.0	double	Vertical rotation offset of simulated beam (relative to 0 position of axis). Not tested.
beam_intensity_offset_measure_enable	yes	bool	If set to "no," offsets will never be measured; instead, defaults defined by configuration file will be used. Setting "no" not recommended because of ADC drift.
debug	0	int	OBSOLETE for MarDTB firmware>4.23. Defined a debug level for beam alignment; if 1 or greater, scans pause for time beam_alignment_bug_scan_delay
beam_alignment_bug_scan_delay	0	int	OBSOLETE for MarDTB firmware>4.23. See debug parameter.
goniostat_mardtb_default_heartbeat	1000	int (ms)	MarDTB heartbeat interval required by mar345 and mar345dtb software.
goniostat_mardtb_adc_average	0	int	Number of MarDTB internal ADC measurements to average for each value (0 leaves default unchanged)
goniostat_mardtb_rc_update_time	110	int (ms)	Remote control update time. Also used by beam alignment to query ADCs at this interval for checking intensities.
goniostat_mardtb_notch_frequency	0	int	Frequency of MarDTB internal ADC measurements to average for each value (0 leaves default unchanged)
goniostat_mardtb_shutter_delay	0.0	double (ms)	MarDTB shutter delay
goniostat_mardtb_amptek_present	no	bool	Set "yes" to indicate amptek present
goniostat_mardtb_cryo_present	no	bool	Set "yes" to indicate cryo present
goniostat_mardtb_cryo_offset_angle	-5.0	double (deg)	Cryo offset angle for detector collision prevention calculation
goniostat_mardtb_detector_clearance	48.0	double (mm)	Detector clearance from detecting surface to outer front surface for detector collision prevention calculation
goniostat_mardtb_collision_distance_min	27.0	double (mm)	Min allowed detector distance collision prevention calculation
goniostat_mardtb_amptek_param	50.0	double (mm)	Max amptek distance from crystal for detector collision prevention calculation
goniostat_mardtb_cryo_param	240.0	double (mm)	Cryo max distance from crystal for detector collision prevention calculation
goniostat_mardtb_chi_param	320.0	double (mm)	Chi max distance from crystal for detector collision prevention calculation

goniostat_mardtb_chi_extra_dist	25.0	double (mm)	Chi axis diameter along phi axis for detector collision prevention calculation
goniostat_mardtb_scan_start_bug	no	bool	Obsolete for firmware>4.24. Caused a 1 sec pause before sending all scan commands.
goniostat_mardtb_scan_samples_per_second	10.0	double	Not Used (software does not modify MarDTB default). When scanning an axis, this is the frequency of intensity recordings per second. Therefore, physical spacing between samples is speed (distance/sec) / frequency (samples/sec)
goniostat_mardtb_rc_theta_position1		double	Not Used
goniostat_mardtb_rc_theta_position2		double	Not Used
goniostat_mardtb_rc_theta_acceleration		double	Not Used
goniostat_mardtb_rc_theta_velocity		double	Not Used
goniostat_mardtb_rc_theta_channel		int	Not Used
goniostat_mardtb_rc_distance_position1		double	Not Used
goniostat_mardtb_rc_distance_position2		double	Not Used
goniostat_mardtb_rc_distance_acceleration		double	Not Used
goniostat_mardtb_rc_distance_velocity		double	Not Used
goniostat_mardtb_rc_distance_channel		int	Not Used
goniostat_mardtb_rc_chi_position1		double	Not Used
goniostat_mardtb_rc_chi_position2		double	Not Used
goniostat_mardtb_rc_chi_acceleration		double	Not Used
goniostat_mardtb_rc_chi_velocity		double	Not Used
goniostat_mardtb_rc_chi_channel		int	Not Used
goniostat_mardtb_rc_cryo_position1		double	Not Used
goniostat_mardtb_rc_cryo_position2		double	Not Used
goniostat_mardtb_rc_cryo_acceleration		double	Not Used
goniostat_mardtb_rc_cryo_velocity		double	Not Used
goniostat_mardtb_rc_cryo_channel		int	Not Used
goniostat_mardtb_rc_beamstop_position1		double	Not Used
goniostat_mardtb_rc_beamstop_position2		double	Not Used
goniostat_mardtb_rc_beamstop_acceleration		double	Not Used
goniostat_mardtb_rc_beamstop_velocity		double	Not Used
goniostat_mardtb_rc_beamstop_channel		int	Not Used
goniostat_mardtb_can_first_id	10	int	First module on CAN bus (not on controller board)

Copyright © 2000-2006, Mar USA

goniostat_mardtb_can_velo_microstep_bug	no	bool	Setting "yes" forces program to compensate for a bug in some goniostat firmware versions (which affects easy move buttons). Program normally determines if necessary (when set "no").
goniostat_mardtb_can_velocity_maximum	15000	int	Max velocity for modules on CAN bus (first_id or higher).
goniostat_mardtb_adc_offset_low	100.0	double	Low range (i.e. high gain) offset (set when Intensity button on status window clicked)
goniostat_mardtb_adc_tolerance_low	20.0	double	Low range (i.e. high gain) tolerance (set when Intensity button on status window clicked)
goniostat_mardtb_adc_offset_high	10.0	double	High range (i.e. low gain) offset (set when Intensity button on status window clicked)
goniostat_mardtb_adc_tolerance_high	2.0	double	High range (i.e. low gain) tolerance (set when Intensity button on status window clicked)
video_input_device	none	string	Input video device name (e.g. /dev/video1)
video_input_device_0	none	string	Input video device 0 name (equals video_input_device)
video_input_encoding_0	none	string	Input video device 0 encoding
video_input_encoding_id_0	COMPOSITE_PAL	int	Input video device 0 encoding type
video_input_device_1	none	string	Input video device 1 name, usually used for grabbing stills
video_input_encoding_1	none	string	Input video device 1 encoding
video_input_encoding_id_1	COMPOSITE_PAL	int	Input video device 1 encoding type
video_input_enable	no	bool	Set "yes" to allow video input in crystal alignment window
video_output_enable	no	bool	Not yet implemented
video_input_port	0	int	Video input port, usually 67
video_output_port	0	int	Not yet implemented
video_crosshair_x	0.0	double	Video crosshair x (must be calibrated)
video_crosshair_y	0.0	double	Video crosshair y (must be calibrated)
video_mm_per_pix_x		double	Only used if specified: video millimeters per pixel
video_mm_per_pix_y		double	Only used if specified: video millimeters per pixel
video_field_width	1.670	double (mm)	Video field width
video_field_height	0.0	double (mm)	Video field height
video_image_width	400	int (pix)	Video image width as displayed on screen
video_image_height	0	int (pix)	Video image height as displayed on screen

Copyright © 2000-2006, Mar USA

video_zoom_bug	no	bool	Flag to correct for bug in older Matrox video card driver
video_still_initialize_command	none	string	Video still initialize command. Usually "v4lctl -c /dev/video1 setinput Composite1"
video_still_command	none	string	Video still command. Usually "streamer -c /dev/video1 -q -t 1 -f pgm -A 10 -o"
video_still_type	pgm	string	File type of video still images
video_still_width	768	int (pix)	Width of video still images
video_still_height	576	int (pix)	Height of video still images
crystal_alignment_window_enable	no	bool	Flag makes crystal alignment window appear in *marccd*

- **Configurable parameters specific to the MarCSC**

The following chart shows parameters normally found in the [MARCCDHOME]/configuration/marcsc_[SN].conf file (where [SN] refers to the three-digit serial number of the goniostat).

Parameter name	**Default**	**Type (Unit)**	**Explanation**
goniostat_mardtb_csc_installed	no	bool	Set to "yes" for MarCSC users
crystal_alignment_window_enable	no	bool	When set to yes, will display crystal alignment tab behind the zoom window
crystal_alignment_automatic_enable	no	bool	Flag enables crystal alignment automatic centering
crystal_alignment_automatic_3D_enable	no	bool	Flag enables Center in 3D button in Crystal Alignment tab.
crystal_alignment_find_crystal_command	"find_crystal"	string	Command used for finding the crystal. Command marloop could be replaced by third-party chosen by user. Input parameters are defined below by crystal_alignment_supply_*
crystal_alignment_find_crystal_3D_enable	no	bool	Flag to indicate that external find crystal routine will drive the entire 3D algorithm
dataset_multicrystal_enable	no	bool	Allows multiple crystals in dataset; will change crystals in automatic mode
dataset_max_crystals	-1	int	Maximum number of crystals in a dataset. Normally 19. If set to -1, will be set equal to dataset_max_segments

dataset_max_segments	8	int	Max segments in dataset is this param or dataset_max_crystals, whichever is larger
sample_changer_max_mount_failures	3	int	Maximum mount tries before failure
sample_changer_max_unmount_failures	3	int	Maximum unmount tries before failure
goniostat_mardtb_csc_pin_length_code_default	-18	int	Length code default. Positive value means code always used. Negative value means code used only if code not read by reader.
goniostat_mardtb_csc_pin_length_adjustment	0.0	double (mm)	Pin length adjustment in Z axis, to correct any differences in hardware setup
goniostat_mardtb_csc_pin_length_[1,2,3,4,5]	[12.0, 14.0, 16.0, 18.0, 21.0]	double (mm)	Actual exact lengths corresponding to the five codes (using non-default requires goniostat_mardtb_csc_pin_length_code_modification set to **yes**)
goniostat_mardtb_csc_pin_length_[1,2,3,4,5]_code	[12, 14, 16, 18, 21]	int	Five possible allowed pin length codes in barcode
goniostat_mardtb_csc_pin_length_code_modification	no	bool	If no, does not allow any pin lengths configured to override standard length for that code
goniostat_mardtb_csc_pin_length			Obsolete / Not Used
goniostat_mardtb_csc_permit_error_level	0	int	Barcode reading is used to detect mount / unmount success. This value determines read errors allowed or not: 0 (stop on any error), 1 (stop on critical errors), 2 (ignore errors)
goniostat_mardtb_csc_require_barcode			Obsolete / Not Used
goniostat_mardtb_csc_low_temperature			Obsolete / Not Used
goniostat_mardtb_csc_use_batch_commands			Obsolete / Not Used
crystal_alignment_video_image_filename	“crystal.pgm”	string	JPEG crystal image saved in log directory for analysis
crystal_alignment_video_image_background_filename	“empty.pgm”	string	JPEG image without crystal (background)

crystal_alignment_supply_crystal_image,	yes	bool	*Note: all crystal_alignment_supply_* parameters are command line inputs to the defined crystal_alignment_find_crystal_command.* Setting "yes" causes the name of supplied crystal image to be included in command call (string). Usage: find_crystal –f filename
crystal_alignment_supply_background_image,	yes	bool	Setting "yes" causes the name of supplied background image to be included in command call (string). Usage: find_crystal –b filename
crystal_alignment_supply_image_number	no	bool	Setting "yes" causes the dataset sequence number of current image to be included in command call (integer). Usage: find_crystal –n number
crystal_alignment_supply_phi_position	no	bool	Setting "yes" causes the current phi position to be included in command call (floating point value). Usage: find_crystal –p phi
crystal_alignment_supply_video_parameters	no	bool	Setting "yes" causes the video width and height to be included in command call (floating point values). Usage: find_crystal –v width height
crystal_alignment_supply_segment_number	no	bool	Setting "yes" causes the dataset segment number to be included in command call (integer). Usage: find_crystal –s number
crystal_alignment_supply_crystal_number	no	bool	Setting "yes" causes the mounted crystal number to be included in command call (integer). Usage: find_crystal –c number
crystal_alignment_supply_crystal_id	no	bool	Setting "yes" causes the cap barcode ID to be included in command call (string). Usage: find_crystal –i id_string
crystal_alignment_background_z_motion	-1.5	double (mm)	crystal z position changed by this amount to collect empty background image
crystal_alignment_cryo_safe_position	0.0	double	If > 0.0, crystal alignment routine drives cyro to this "safe" position before moving crystal for background image

Copyright © 2000-2006, Mar USA

crystal_alignment_detector_safe_position	0.0	double	If > 0.0, crystal alignment routine drives detector distance to this "safe" position before moving crystal for background image
crystal_alignment_beamstop_safe_position	0.0	double	If > 0.0, crystal alignment routine drives beamstop distance to this "safe" position before moving crystal for background image
crystal_alignment_theta_safe_position	0.0	double	If > 0.0, crystal alignment routine drives two-theta motor to this "safe" position before moving crystal for background image
crystal_alignment_use_background	yes	bool	Setting "yes" causes background to be collected for crystal alignment routine
crystal_alignment_illuminate_background	yes	bool	Setting "yes" activates background illumination device for crystal alignment routine
crystal_alignment_illuminate_crystal	no	bool	Setting "yes" activates crystal illumination device for crystal alignment routine
crystal_alignment_default_zoom	1.0	double	Sets default video zoom when starting program
crystal_alignment_default_brightness	0.5	double	Sets default video brightness when starting program
crystal_alignment_default_contrast	0.5	double	Sets default video contrast when starting program
crystal_alignment_initial_phi_separation	90.0	double	Angular divisor of first 360 degrees of observation.
crystal_alignment_min_sttp	0.10	double	Minimum Student's T-Test P. If P of outlier is less than this number, then it is thrown out.
crystal_alignment_tolerance	0.100	double	Not Used
crystal_alignment_tolerance_x	0.025	double	Crystal alignment continues until sigma of x position less than this value.
crystal_alignment_tolerance_y	0.025	double	Crystal alignment continues until sigma of y position less than this value.
crystal_alignment_tolerance_z	0.050	double	Crystal alignment continues until sigma of z position less than this value.
crystal_alignment_max_motion	1.5	double (mm)	If requested combined 3D motion greater than this, alignment failure
crystal_alignment_max_motion_x	1.5	double (mm)	If requested X motion greater than this, alignment failure
crystal_alignment_max_motion_y	1.5	double (mm)	If requested Y motion greater than this, alignment failure
crystal_alignment_max_motion_z	1.5	double (mm)	If requested Z motion greater than this, alignment failure

Copyright © 2000-2006, Mar USA

crystal_alignment_max_trusted_motion	0.2	double (mm)	If any one observation makes a move greater than this distance, then new position is not kept in list for averaging.
crystal_alignment_max_trusted_motion_x	0.2	double (mm)	Not Used
crystal_alignment_max_trusted_motion_y	0.2	double (mm)	Not Used
crystal_alignment_max_trusted_motion_z	0.2	double (mm)	Not Used
crystal_alignment_min_trusted_status_2D	5	int	When external alignment routine used for 2D alignment, if it returns reliability status less than this number, then new position is not kept in list for averaging.
crystal_alignment_min_trusted_status_3D	5	int	When external alignment routine used for 3D alignment, if it returns reliability status less than this number, then new position is not kept in list for averaging.
crystal_alignment_min_positions	6	double	Minimum number of crystal positions used for alignment
crystal_alignment_max_positions	16	double	Maximum number of crystal positions used for alignment
crystal_alignment_max_trials	1	double	Not Used

Upgrading *marccd*

Note: There was a major change in configuration files from version 0.8.20 onward. If upgrading from an older executable, we recommend contacting us at Mar USA to help update your configuration files first.

Upgrades to the program can be downloaded via ftp at mar-usa.park.nunet.net. After connecting with an ftp session, login with username anonymous, password your email address. Change to the directory pub/marccd/bin/linux. There is a Release Notes file which explains the changes made to each version. A README file explains how to install the latest version of *marccd.*

Bug reports and feature requests

Please help us improve the software. Report bugs and feature requests, preferably by making an account and bug entry on our public bug server at http://www.mar-usa.com/bugs, or alternatively by email to the main Mar USA account (info@mar-usa.com). Bug reports absolutely require a) the software version; b) enough background information to repeat the problem, such as which buttons were clicked and what settings were used, etc; c) at file attachment with the relevant lines of output from the file [MARCCD_HOME]/log/stdouterr.log (hint: type "`tail -n 1000 ~/log/stdouterr.log > my_logfile.log`"). We will not be able to evaluate simple bug reports, such as, “Software crashed twice,” or, “We did what we always do, but it didn’t work.”

Copyright © 2000-2006, Mar USA

Appendix C: Troubleshooting and Service

MarCCD Troubleshooting guide

problem/symptom	probable cause	fix
Streaks, vertical Example: An image of a crystal diffraction pattern shows long vertical streaks starting/ending at diffraction spots.	***Shutter problem:*** The shutter remains open after the detector has stopped integrating (i.e. exposure is finished) and starts to read out.	Lubricate with graphite or WD-40 into brass bushing around shutter shaft. Toggle shutter manually (the switch is on underside of base, close to the detector side edge of the phi-motor housing). If this does not help, call Mar USA for instructions.
Blank zone, horizontal stripe in center 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.	***Shutter problem:*** Shutter opens and closes before the detector starts integrating.	
Noise High background noise in images.	***Experimental Setup:*** Air scatter, fibers in beam, diffuse scatter... ***Temperature:*** The detector is not cooled to the proper operating temperature.	Check the beam path and crystal mount. Check the pressure (detector status window within main marccd software window).
Pressure The pressure reported in the detector status window is higher than 1 Torr when the detector is cold or higher than 2 Torr when warm.	***Vacuum decayed:*** The vacuum in the detector chamber is probably not good enough to allow cooling the detector to the optimal operating temperature.	Follow the procedure for re-evacuating the CCD chamber (page 6 of this section).
Temperature The detector will not cool below -30 or -40°C.	***Vacuum decayed:*** The pressure in the detector chamber is too high.	Follow the procedure for re-evacuating CCD chamber (page 6 of this section).
Detector will not cool below 0°C.	***Moisture in the system.***	Call Mar USA for service.
The detector temperature cycles between cold and warm	***Cooler fan not working.***	Test cooler fan intake by holding a piece of paper near fan grille. Order parts from Mar USA.

Copyright © 2000-2006, Mar USA

No update of status window The information on the status window in the main marccd window does not change even though data is being collected properly.	**Bug**	Restart software.
Blank image	***Experimental setup:*** No X-rays? ***Shutter problem:*** Does not open? ***Background image:*** A frame with large values for all pixels is stored as background. Automatic subtraction of this faulty background produces an all-zero image.	Turn X-rays on. Check shutter manually, make sure all cables are hooked up properly. Collect new background frame.
No Response (Message in detector status window):	***The controller*** (black box in cooler cabinet) ***cannot communicate with the detector head.***	Check all cables, especially power to detector head and the cable between the controller and the detector head (F.O. cable or older 68-pin cable).
Cooler won't come on ***or*** ***Cooler turns off on its own.***	***Intentional Delay:*** There is about a one-minute delay after asking for cooler power via software before the cooler actually comes on. This is to prevent damage from rapid power-cycling of the cooler. ***No power:*** The power to the cooler is not on? ***Time-delay relay bad.***	Click the check box next to "cooler power" in the "Configure Detector" window to show a check mark. ***Then click on "Apply"*** and wait about one minute. Check power cables, and then switch on the power strip inside the cooler cabinet and the power switch on the compressor itself. Call Mar USA for replacement parts.

Copyright © 2000-2006, Mar USA

Goniostat does not move or moves incorrectly ***The "Configure Detector" window contains nonsense numbers.***	***Communication problem and/or corrupt goniostat file.***	Follow goniostat reset procedure on page?.
"Could not open file..." error message	***Permissions wrong.***	Type chmod 666 filename (return)
One or more quadrants of the images are missing.	***Not all readout-channels are working.***	Check cables between Detector controller (black box inside cooler cabinet) and detector head.
The images look as if they have been cut into 4 quadrants and re-assembled wrong. quadrants mixed up	***Cables*** between Detector head and controller are ***mixed up.***	The four color-coded cables between the 4 readout channels on the detector head and the black controller box inside the cooling cabinet have to connect A (on detector) to A (on controller), B (on detector) to B (on controller), C to C, D to D.
Shutter: won't open ***or*** ***Shutter: won't close***		Exercise shutter manually via the switch on the bottom of the Marbase below the detector-side edge of the phi-motor housing. If that does not help, call Mar USA for instructions.
Phi axis does not turn	***Preload on Phi-axis bearings is too tight.*** ***Contact between gears is too loose or too tight.***	Refer to section ? of this manual.
Beam stop misaligned	***The direct beam should not damage the MarCCD detector.***	To align the beam stop, refer to section ? of this manual.

“weak beam”	Rotating Anode: ***Not at full power.*** Rotating Anode or Synchrotron: ***Slits completely closed or base is misaligned.*** ***Beam conditioning optics are out of alignment.***	Slowly turn power up on the rotating anode. Open slits (0.4mm or as desired). Re-align base as described in the manual section called “MarDTB Beam Alignment using *marccd* Refer to alignment procedure for your optics.
The detector has run against a limit switch and cannot be backed off.	***Limit switch is depressed.***	Try to initialize from the marccd software to both the near and the far end. If it does not work, reach in and push the limit switch open while driving the detector away from it.
The shadow of the Mylar-type beam stop holder is visible in the images.	***Beam stop holder is pulled out.***	Push the bottom part of the beam-stop holder towards the crystal position. Make sure it does not interfere with the goniometer head in any phi position.

- **The temperature and/or pressure readout in the software show "????".**

  This means that the detector controller is either not running at all or not communicating with the computer running the MarCCD software.

  Usually this is fixed by rebooting the detector controller. To do this, choose “Detector” from the “Configuration” menu and then click on the “Reboot” button on the bottom of the Configure Detector panel. A dialog box will appear informing you that the default settings will be loaded upon rebooting the detector controller. Click on “OK.” The detector controller itself (in the large gray box on wheels that also contains the cooler) will beep twice upon rebooting.

  After rebooting the cooling needs to be turned back on as soon as possible. To turn the cooling on, click the cooler button to show a check mark (√) and then click on the large “Apply” button. The cooling will come on about 40 seconds after the “Apply” button was clicked.

- **The cooler is on but the detector temperature readout does not go down.**

When starting from room temperature, it takes about 2 hours before the temperature starts to drop faster. The ideal temperature for data collection may not be reached for 3-4 hours. This is because the entire fiber optics taper is cooled and a solid block of glass this size takes a while to cool to such low temperatures.

If the temperature does not go down in 3-4 hours, the pressure inside the detector may be too high. At room temperature the pressure should usually be less than 1 Torr, anything above 2 Torr may make cooling difficult. See the section on re-evacuating the detector, page ?.

If the temperature goes down for a while but then actually starts to go back up slightly while the cooler is running, it may help to stop the cooler (turn “Cooler Power” off in Configure=>Detector panel, then click on “Apply) and restart it after about an hour. A drastic measure (only if the temperature is RISING while cooler is ON) is to raise the “set” temperature to 30°C and actually warm the detector up a little bit. This can only be done in “Expert Mode” in the “Configure => Detector” panel, by typing “30.0” into the ?? box and the clicking on “Apply”. Leave the detector at 30°C for about one hour, then change the set temperature back to the normal value and try cooling again.

- **“Vac” Light on the detector is on.**

  The light labeled "vac" indicates that the pressure is high enough to justify pumping down the chamber.

- **Cooler not running:**

  If the “Temperature” and/or “Pressure” read “????”, the problem is communication. See page ? on rebooting the detector controller.

  The cooler can only be turned on via the MarCCD software. This is done from the “Configure => Detector” panel: Click on the “Cooler Power” button to show a check mark (√), then click on “Apply”.

  If the cooler does not come on within minutes of turning it on from the software, then the problem is almost certainly a fuse or circuit breaker.

  The first to check is the "circuit breaker" switch - it is on the opposite side of the cabinet from the pressure gauge. You will need to take the other outer cover off of the cabinet.

  REMEMBER TO UNPLUG THE COOLER BEFORE CHECKING FUSES!!

  There are two fuses (1A) in the power entry module in the cooler itself - where the power cord enters.

Then, unfortunately, there are two 10A fuses inside the cooler box itself. Checking/changing these is rather cumbersome and has been changed on later models (serial numbers here?). You must remove the panel that holds the circuit breaker switch.

REMEMBER TO UNPLUG THE COOLER BEFORE REMOVING THE CIRCUIT BREAKER!

The fuses are BEHIND the circuit breaker. It is usually easiest to remove the two screws holding the circuit breaker into its bracket so that you can see/reach the fuses.

Procedure for re-evacuating CCD chamber (MarCCD):

Note: This procedure for MarCCD detectors ONLY! MarMosaic detectors use an external TMP that runs constantly.

Tools needed:

1. 3 mm hex key
2. vacuum pump capable of 0.1 Torr (0.01 Torr is better)
3. vacuum hose with clamp or fitting to attach to a 1/4" OD tube (~ 6mm)
4. Actuator (spanner) for MarCCD systems with external valve—this is supplied in the tan tool bag labeled *marccd.*

- The detector must be warm. Turn the cooling power off: in the Configure Detector menu in the marccd software, click on the “cooling power” button to show NO check mark and then click on “Apply.” Let the detector warm up for until it is close to room temperature (i.e. above +12 C).

The following steps apply to older MarCCD systems with INTERNAL vacuum valves only:

- Switch off power to electronics/cooler crate. You can leave the software running with the configure/detector dialog box up.
- Write down the order in which the blue analog cables are hooked up. They have colour-coded ends for this purpose.
- Remove the electrical cables (6 cables) from the detector head. LEAVE the cooling hoses ATTACHED!
- Remove the 4 socket head cap screws holding the cylindrical cover on the back of the detector and remove the cover by sliding it back over the cooling hoses.
- You should be able to see the brass vacuum fitting at the bottom of the detector and the green valve handle. Attach the vacuum hose to the fitting and secure it with a clamp. You may need to dismount the detector in order to reach the fitting. In this case, the best place to put the detector is on the floor, standing on its front. It is perfectly safe to handle the detector by the cooling tubes on the back.
- Reconnect the two larger (power and control) cables. It is NOT necessary to reconnect the 4 blue analog cables.
- Switch on power to the electronics/cooler.
- Reboot the detector controller from the Configure/Detector window.

Copyright © 2000-2006, Mar USA

The following steps apply to MarCCD detectors with EXTERNAL vacuum valves only:

- Remove the black vinyl cover from the vacuum port (labeled "VAC PORT") at the back of the detector head.
- Attach the vacuum pump via the vacuum hose and secure it with the hose clamp.
- Remove the plastic cover from the vacuum valve at the back of the detector head (labeled "VAC VALVE").
- Gently attach the actuator to the valve, do not turn it yet.

The following steps apply to all MarCCD detectors:

- Check that you get a reading of temperature and pressure.
- Switch on the vacuum pump.
- Slowly turn the valve handle to open the valve while watching the pressure reading in the software. Verify that the pressure starts to go DOWN when the valve is opened. If it goes UP, then CLOSE THE VALVE IMMEDIATELY and verify/improve the vacuum connection to the pump and try again.
- Open the valve fully and allow to pump until the pressure is reasonable. The pressure increases approximately 0.1 Torr/week. It is possible to pump the chamber to 0.01 Torr, but that will only increase the lifetime of the vacuum by about a week and may not be worth the effort. 0.1 to 0.2 is usually good enough. A short vacuum hose will improve the speed of pumping. With a 25cm hose I can usually bring the vacuum to 0.1 Torr in 5-10 minutes and to 0.01 within 20-30 minutes.
- When satisfied with the vacuum, close the valve, disconnect the hose, and switch off the pump. Check that the vacuum is still OK!

For MarCCD systems with internal vacuum valves

- Switch off the power to the electronics.
- Disconnect the power and control cables.
- Replace the cover. replace the 4 screws.
- If you dismounted the detector, then remount it now.
- Reconnect all 6 cables. The color code of should be used to verify that the cables are connected properly. Top to bottom on the detector is ABCD. On the controller ABCD is left to right. Connect A to A, B to B, ... etc... (use note made earlier)
- Switch on power to the electronics/cooler.
- Reboot the detector controller from the Configure/Detector window.

For all MarCCD detectors:

- Enable cooling and click OK or Apply.
- Verify that the cooler comes on. The detector temperature should begin to drop within 10-15 minutes, although occasionally it will take 1 hour to get started. It should reach operating temperature in about 2 hours.

Copyright © 2000-2006, Mar USA

MarMosaic Troubleshooting guide

problem/symptom	probable cause	fix
Streaks, horizontal Example: An image of a crystal diffraction pattern shows long vertical streaks starting/ending at diffraction spots.	***Shutter problem:*** The shutter remains open after the detector has stopped integrating (i.e. exposure is finished) and starts to read out.	Lubricate with graphite or WD-40 into brass bushing around shutter shaft. Toggle shutter manually (the switch is on underside of base, close to the detector side edge of the phi-motor housing). If this does not help, call Mar USA for instructions.
Blank zone, horizontal stripe in center 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.	***Shutter problem:*** Shutter opens and closes before the detector starts integrating.	
Noise High background noise in images.	***Experimental Setup:*** Air scatter, fibers in beam, diffuse scatter... ***Temperature:*** The detector is not cooled to the proper operating temperature.	Check the beam path and crystal mount. Check the pressure (detector status window within main marccd software window).
Pressure The pressure reported in the detector status window is higher than 100 mtorr when the detector is cold or higher than 500 mtorr when warm.	***Vacuum decayed:*** The vacuum in the detector chamber may not be good enough to allow cooling the detector to the optimal operating temperature.	Make sure that 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.
Temperature The detector will not cool below -30 or -40°C.	***Vacuum decayed due to valve closing.***	Call Mar USA for service.
Detector will not cool below 0°C.	***Moisture in the system.***	Call Mar USA for service.
The detector temperature cycles between cold and warm	***Cooler fan not working.***	Test cooler fan intake by holding a piece of paper near fan grille. Order parts from Mar USA.

Copyright © 2000-2006, Mar USA

Question marks: The main marccd window has question marks for the temperature and pressure readings.	***Lost Communication:*** The software has lost communication with the detector.	Check: 1) Power to controller is on and all voltage LEDs are green, no warning LEDs on. 2) All switches are on. 3) Cable connectors are OK. 4) Fuses are not burned out. Power cycle detector and computer, restart software.
Blank image	***Experimental setup:*** No X-rays?	Turn X-rays on.
	Shutter problem: Does not open?	Check shutter manually, make sure all cables are hooked up properly.
	Background image: A frame with large values for all pixels is stored as background. Automatic subtraction of this faulty background produces an all-zero image.	Collect new background frame.
Cooler won't come on ***or*** ***Cooler turns off on its own.***	***Intentional Delay:*** There is about a one-minute delay after asking for cooler power via software before the cooler actually comes on. This is to prevent damage from rapid power-cycling of the cooler.	Click the check box next to "cooler power" in the "Configure Detector" window to show a check mark. ***Then click on "Apply"*** and wait about one minute.
	No power: The power to the cooler is not on?	Check power cables, switch on the power strip inside the cooler cabinet and the power switch on the compressor itself.
	Time-delay relay bad.	Call Mar USA for replacement parts.
"Could not open file..." error message	***Permissions wrong.***	Type chmod 666 filename (return)

Copyright © 2000-2006, Mar USA

One or more quadrants of the images are missing.	***Not all readout-channels are working.*** ***Or, bad background frame.***	Check cables between Detector controller (black box inside cooler cabinet) and detector head. Recollect background
Phi axis does not turn	***Preload on Phi-axis bearings is too tight.*** ***Contact between gears is too loose or too tight.***	Refer to section ? of this manual.
Beam stop misaligned	***The direct beam should not damage the MarMosaic detector.***	To align the beam stop, refer to section ? of this manual.
"weak beam"	Rotating Anode: ***Not at full power.*** Rotating Anode or Synchrotron: ***Slits completely closed or base is misaligned.*** ***Beam conditioning optics are out of alignment.***	Slowly turn power up on the rotating anode. Open slits (0.4mm or as desired). Re-align base as described in the manual section called "MarDTB Beam Alignment using *marccd* Refer to alignment procedure for your optics.
The detector has run against a limit switch and cannot be backed off.	***Limit switch is depressed.***	Try to initialize from the marccd software to both the near and the far end. If it does not work, reach in and push the limit switch open while driving the detector away from it.
The shadow of the mylar-type beam stop holder is visible in the images.	***Beam stop holder is pulled out.***	Push the bottom part of the beam-stop holder towards the crystal position. Make sure it does not interfere with the goniometer head in any phi position.

Mar USA Equipment Service

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

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

Appendix D: The Remote Mode of *marccd*

This section of the manual covers software versions:

marccd ver. 0.6.6 and higher – first remote mode implementation
0.9.2 for get_size_bkg
0.9.9 for set/get_frameshift
0.10.10 for remote mode ver. 1
0.10.17 for image thumbnails

Introduction

It is possible to collect data by marccd under the control of another program. We call this the "Remote mode" of data collection. Data acquisition controls, such as binning, integration, readout, setting header information, and saving files, are available through this interface. An institution might prefer to use this mode if they have developed a user interface that can control many kinds of equipment. Several institutions are currently using this method successfully, including ESRF.

Note: A license key may be required from Mar USA, included in the marccd *configuration file, to operate the remote mode.*

Compiling the sample programs

Along with these instructions you should obtain a tar file called example_remote_server.tgz (generally found in ~marccd/contrib/ but if your release of marccd is older than 0.6.6, you may request a newer file provided from Mar USA). 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.txt
socket_utils.c
socket_utils.h

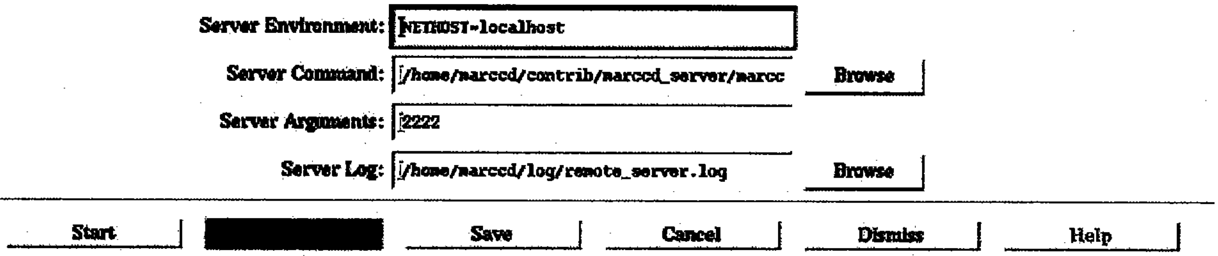
Before compiling any programs, type "`make depend`" in the current directory to update the dependencies in the Makefile to match the compiler libraries on your computer.

Copyright © 2000-2006, Mar USA

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 *marccd* using pipes, but in the example that follows, the programs with socket connections are used.

The remote mode dialog box and the server program

To put the program *marccd* into remote mode, select "Acquire→Remote control" in the menu bar (*if this option is not available, you may need to obtain a license key from Mar USA and put it in the configuration file*). A dialog box should come up with four fields. These four fields have customizable label names and values that are sourced from the file ~marccd/configuration/marccd_server.conf (or marccd_server_v1.conf). By default, the field labels are as shown in Figure 39:



Server Environment: NETHOST=localhost
Server Command: /home/marccd/contrib/marccd_server/marcc
Server Arguments: 2222
Server Log: /home/marccd/log/remote_server.log

Figure 39 - Remote mode dialog box

- **Server Environment:** This field can be left blank. (If an environment variable is required by the control program, it can be inserted here. Some configuration of *marccd* may be required by Mar USA to use this.)

- **Server Command:** Enter marccd_server_socket or click "Browse" to browse for the desired server program.

- **Server Arguments:** Enter 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, so enter 2002.

- **Server Log:** This can be left blank. It is not yet fully implemented at this time.

(Earlier versions of *marccd* probably have a different name for each of these fields, but they should operate similarly.) The configuration file, ~marccd/configuration/marccd_server.conf, is called by the site configuration file, ~marccd/configuration/marccd_site.

When the user clicks "Start" in the Remote control dialog box, *marccd* will execute the specified server program as a child process. The server is responsible for translating commands and

Copyright © 2000-2006, Mar USA

responses between the controlling program and *marccd*. The communication model is shown in Figure 40. Though the client program is shown operating on another computer, in the case of the included sample client program, the client would be running on the detector workstation.

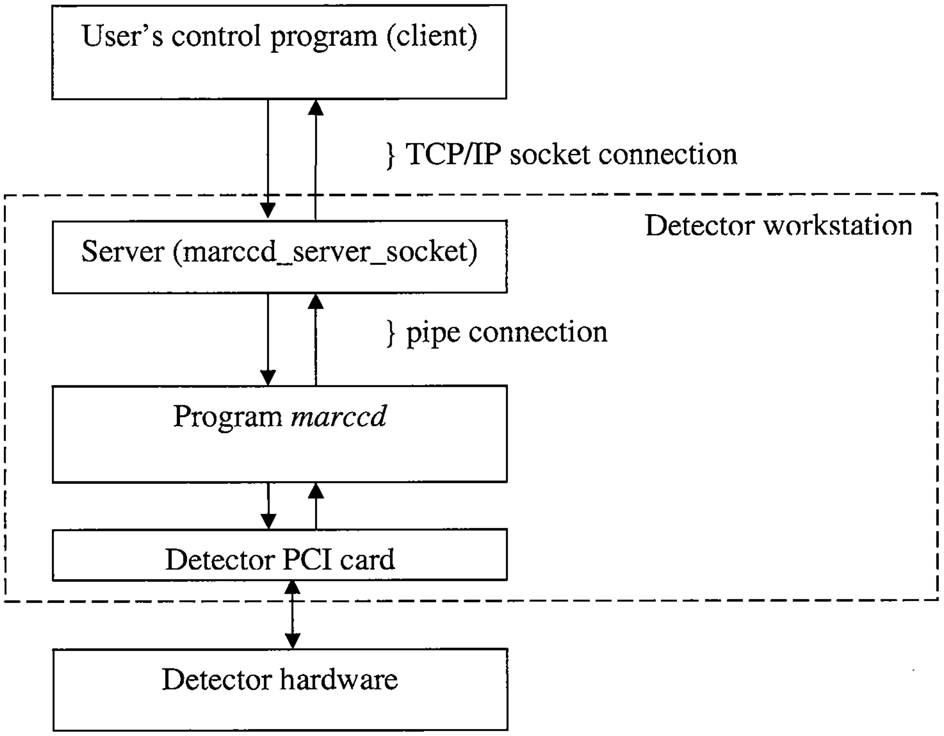


Figure 40 - Remote mode communication path

At the ESRF the server program is MarCCDds. MarCCDds advertises itself to the ESRF device server database system and accepts commands from that system. It interprets those commands and passes commands to *marccd* that *marccd* can understand. MarCCDds is available in the ~marccd/contrib/ directory.

Alternatively, *marccd* can be started directly into the remote mode with the default options by using the -r command line option. The values specified in the configuration file ~/marccd/configuration/marccd_server.conf (or marccd_server_v1.conf) are automatically used.

The client program

Now look at the sample client program marccd_client_socket. It functions like a telnet session to the socket program, into which text commands (described in the next section) may be entered to drive *marccd*. 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 *marccd* program is executing these commands.

Remote commands used by *marccd*

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

<table>
<tr><th>Remote Mode Command</th><th>Effect</th></tr>
<tr><td><code>get_size</code></td><td><i>marccd</i> 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><i>marccd</i> 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><i>marccd</i> 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><i>marccd</i> will set the fast (x) and slow (y) binning of the data frame.</td></tr>
<tr><td><code>get_frameshift</code></td><td><i>marccd</i> will answer with the frameshift parameter (nlines).</td></tr>
<tr><td><code>set_frameshift,nlines</code></td><td><i>marccd</i> will set the frameshift parameter (nlines)</td></tr>
<tr><td><code>set_thumbnail1,type,xsize,ysize</code></td><td><i>marccd</i> will use these values for writing thumbnail images
<table>
<tr><th>Param.</th><th>Meaning</th><th>Examples</th></tr>
<tr><td><code>type</code></td><td>file type</td><td>PGM, RAW, TIFF, JPG, GIF, PNG, EPS, MIFF, HTML, or FITS. (If Image-Magick (convert) is not installed, then only PGM, RAW, and TIFF are supported)</td></tr>
<tr><td><code>xsize</code></td><td>horizontal size in pixels</td><td>set to 0 to give maximum ysize</td></tr>
<tr><td><code>ysize</code></td><td>vertical size in pixels</td><td>set to 0 to give maximum xsize</td></tr>
</table></td></tr>
<tr><td><code>set_thumbnail2,type,xsize,ysize</code></td><td>(same as above)</td></tr>
<tr><td><code>start</code></td><td><i>marccd</i> will start integrating data (stop clearing) on the CCD.</td></tr>
</table>

<table>
<tr><td>readout,flag[,filename],[thumbnail1 filename],[thumbnail2 filename]</td><td>*marccd* will stop integrating and start reading the CCD; given filename(s), it will queue the correction and writing of the file to disk
<table>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>0</td><td>read data into raw data frame storage</td></tr>
<tr><td>1</td><td>read data into background frame storage</td></tr>
<tr><td>2</td><td>read data into system scratch storage</td></tr>
<tr><td>3</td><td>read data into data frame storage and do NOT correct [and write uncorrected frame]</td></tr>
</table></td></tr>
<tr><td>dezinger,flag</td><td>*marccd* will calculated a "dezingered" frame from two stored frames. One of the source frames is the System Scratch frame.
The second source frame and the destination are specified with the flag.
<table>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>0</td><td>use and store into the latest data frame.</td></tr>
<tr><td>1</td><td>use and store into the current background frame</td></tr>
<tr><td>2</td><td>use and store into system scratch storage (not useful; frame dezingered with itself)</td></tr>
</table></td></tr>
<tr><td>correct</td><td>*marccd* will apply geometric and flatfield corrections to the raw data frame.</td></tr>
<tr><td>writefile,filename,flag</td><td>*marccd* will write out a data frame to a file on disk. The parameter `filename` is the name of the file to be written.
<table>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>0</td><td>write raw file</td></tr>
<tr><td>1</td><td>write corrected file</td></tr>
</table></td></tr>
<tr><td>writethumbnail1,filename,flag</td><td>*marccd* will write out a thumbnail image of a data frame to a file on disk. Flags have same meaning as for `writefile` command. See `set_thumbnail1` for format of thumbnails.</td></tr>
<tr><td>writethumbnail2,filename,flag</td><td>(same as above)</td></tr>
<tr><td>abort</td><td>*marccd* will abort the current operation. Normally this would be done to stop integration and return the CCD to continuous clear mode.</td></tr>
</table>

<table>
<tr>
<td><code>header,header_data\n</code></td>
<td><i>marccd</i> will accept <code>header_data</code> and interpret item=value pairs to be placed into the data frame header. <code>header_data</code> consists of a list of item=value pairs separated by commas and terminated by a newline (<code>\n</code>). The following items are understood:
<table>
<tr><th>Parameter</th><th>Type (Units)</th></tr>
<tr><td>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>
</table>
</td>
</tr>
<tr>
<td><code>get_state</code></td>
<td><i>marccd</i> 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>
<tr><th>State Number</th><th>State</th></tr>
<tr><td>0</td><td>IDLE</td></tr>
<tr><td>1</td><td>ACQUIRE</td></tr>
<tr><td>2</td><td>READOUT</td></tr>
<tr><td>3</td><td>CORRECT</td></tr>
<tr><td>4</td><td>WRITING</td></tr>
<tr><td>5</td><td>ABORTING</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>
</table>
</td>
</tr>
<tr>
<td><code>set_state,state</code></td>
<td><i>marccd</i> will set the state to the desired state. This is for testing purposes only and has no use in a normally functioning system.</td>
</tr>
</table>

Copyright © 2000-2006, Mar USA

<table>
<tr><td><code>shutter,flag</code></td><td>marccd will set the shutter state to either closed or open. (Only if marccd 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>
<tr><th>Flag</th><th>Action</th></tr>
<tr><td>0</td><td>closed</td></tr>
<tr><td>1</td><td>open</td></tr>
</table></td></tr>
<tr><td><code>end_automation</code></td><td>marccd will exit remote mode.</td></tr>
<tr><td><code>phi,move_flag, absolute_flag,angle</code></td><td>***CURRENTLY NOT SUPPORTED***
marccd will move the phi axis (only if marccd controls the phi axis!).</td></tr>
<tr><td><code>distance,move_flag, absolute_flag,angle</code></td><td>***CURRENTLY NOT SUPPORTED***
marccd will move the detector distance (only if marccd controls the detector distance!).</td></tr>
</table>

Note on command handling by server application

In addition to the above commands, it is recommended that any server application implements the following commands:

Command to server program	**Action**
`get_state`	same as above, but queries from the client should be answered directly by the server without querying *marccd.*
`get_size`	same as above, but queries from the client should be answered directly by the server without querying *marccd.*
`get_size_bkg`	same as above, but queries from the client should be answered directly by the server without querying *marccd.*
`get_frameshift`	same as above, but queries from the client should be answered directly by the server without querying *marccd.*
`get_bin`	same as above, but queries from the client should be answered directly by the server without querying *marccd.*
`get_state_hist`	(Implemented completely in the server.) Anwers with the current state and the most recent previous state, separated by commas. (See `get_state`.)

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:

Task Number	Task
0	TASK_ACQUIRE
1	TASK_READ
2	TASK_CORRECT
3	TASK_WRITE
4	TASK_DEZINGER

The status bits for each task are:

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

Therefore, the state value looks like Figure 41, with eight four-bit fields

unused	unused	dezinger	write	correct	read	acquire	state

Figure 41 - State fields in remote mode version 1

Examples state values returned by get_state:

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

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

```
#define STATUS_MASK             0xf

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

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

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

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

Copyright © 2000-2006, Mar USA

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

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

   /* send: readout,1 */

/* Get a 2nd backround 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 *marccd.* 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`	

Copyright © 2000-2006, Mar USA

• `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 MarCCD 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]	
• `readout,0`	*(read data into raw frame buffer)*
• `correct`	*(apply correction; data sent to "corrected" buffer)*
• `writefile,FILENAME,1`	*(write data from corrected frame buffer to file)*

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.

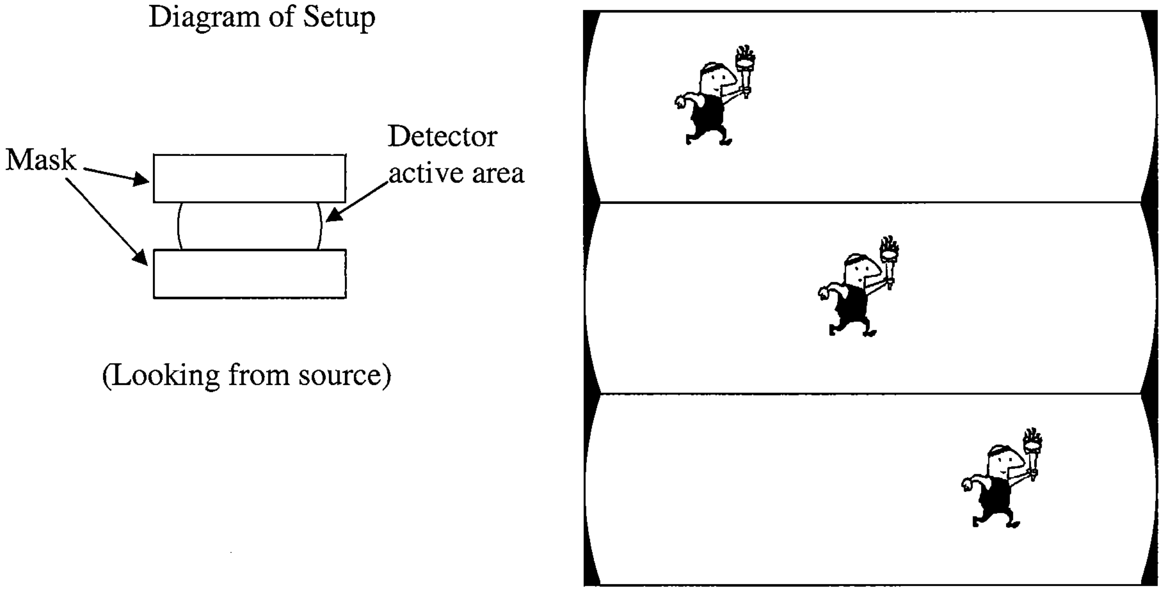
If the source has constant intensity, then TIME1 = TIME2 = total_time/2. However, if the source has a short decay time, then the times must be TIME2>TIME1, calculated so that that both frames have equal dose, within a few percent.

Copyright © 2000-2006, Mar USA

Appendix E: The Frameshift Option for MarCCD

Frameshift is an optional mode of data collection with the MarCCD (not available with MarMosaic). It makes taking data frames separated by only milliseconds possible. Frameshift mode uses part of the CCD chip for imaging and the rest for data storage. After collecting the first image in the sequence, the CCD is triggered to quickly shift a configurable number of pixel rows, exposing fresh rows for the next image. Further triggers can make the CCD shift again and again until the CCD is filled up. Then the entire CCD is read out at the standard rate, and a corrected image is produced showing a collage of the separate exposures.

Example: 1/3 of detector exposed for frameshift imaging



Output Produced

Diagram of Setup

(Looking from source)

Image is 2048 x 2048 total; 3 images possible with frameshift set to 682 lines

Figure 42 - Frameshift example information

The following items are required to utilize the frameshift option:

- frameshift software license key, available for purchase from Mar USA (for *marccd* configuration file)
- pulse generator to trigger the MarCCD to shift (usually synchronized to the experimental setup, such as a shutter or a laser, etc.)
- lead or steel mask covering the detector face, leaving a straight horizontal band across the center of the detector exposed for imaging

Appendix F: Native Image Format for *marccd* Images

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 MarMosaic 225 will have size (4096 + 3072*3072*2) bytes = 18878464 bytes, or approximately 18 MB. The table below shows resolutions produced by various Mar USA X-ray detectors:

Resolution	File size (exact) (bytes)	File size (MB)
512 x 512	528384	0.5
768 x 768	1183744	1.1
1024 x 1024	2101248	2
1536 x 1536	4722688	4.5
2048 x 2048	8392704	8
3072 x 3072	18878464	18
4096 x 4096	33558528	32

Image Headers

The header description for *marccd*:

```
MarCCD Header Documentation
from C code in frame.h and types.h

Documentation updated by M. Blum Wed Jul 29 17:50:56 CDT 1998

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.
```

Copyright © 2000-2006, Mar USA

Immediately following the header is the image - it is of arbitrary size defined by the header fields nfast, nslow and depth. The total size is nfast * nslow * depth bytes.
The meanings of the data types should be self-evident (example: UINT32 is an unsigned 32 bit integer).
The exact C language definition is machine dependent but these are the most common definitions on a 32bit architecture cpu.

```
#define UINT16 unsigned short
#define INT16 short
#define UINT32 unsigned int
#define INT32 int
Currently frames are always written as defined below:
origin=UPPER_LEFT
orientation=HFAST
view_direction=FROM_SOURCE
/* This number is written into the byte_order fields in the
native byte order of the machine writing the file */
#define LITTLE_ENDIAN 1234
#define BIG_ENDIAN 4321
/* possible orientations of frame data (stored in orienation
field) */
#define HFAST 0 /* Horizontal axis is fast */
#define VFAST 1 /* Vertical axis is fast */
/* possible origins of frame data (stored in origin field) */
#define UPPER_LEFT 0
#define LOWER_LEFT 1
#define UPPER_RIGHT 2
#define LOWER_RIGHT 3
/* possible view directions of frame data for
the given orientation and origin (stored in view_direction field)
*/
#define FROM_SOURCE 0
#define TOWARD_SOURCE 1
typedef struct frame_header_type {
/* File/header format parameters (256 bytes) */
UINT32 header_type; /* flag for header type (can be used as
magic number) */
char header_name[16]; /* header name (MMX) */
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 */
UINT32 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 background_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 */
char reserve1[(64-40)*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;
UINT32 mean;
UINT32 rms;
UINT32 p10;
UINT32 p90;
UINT32 stats_uptodate;
UINT32 pixel_noise[MAXIMAGES]; /* 1000*base noise value
(ADUs) */
char reserve2[(32-13-MAXIMAGES)*sizeof(INT32)];
```

```
/* More statistics (256) */
UINT16 percentile[128];
/* 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_gamma; /* 1000*gamma angle */
INT32 start_xtal_to_detector; /* 1000*distance in mm (dist in um)
*/
INT32 end_twotheta; /* 1000*two_theta angle */
INT32 end_omega; /* 1000*omega angle */
INT32 end_chi; /* 1000*chi angle */
INT32 end_kappa; /* 1000*kappa angle */
INT32 end_phi; /* 1000*phi angle */
INT32 end_delta; /* 1000*delta angle */
INT32 end_gamma; /* 1000*gamma angle */
INT32 end_xtal_to_detector; /* 1000*distance in mm (dist in um)*/
INT32 rotation_axis; /* active rotation axis */
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 */
char reserve3[(32-28)*sizeof(INT32)];
/* Detector parameters (128 bytes) */
INT32 detector_type; /* detector type */
INT32 pixelsize_x; /* pixel size (nanometers) */
INT32 pixelsize_y; /* pixel size (nanometers) */
INT32 mean_bias; /* 1000*mean bias value */
INT32 photons_per_100adu; /* photons / 100 ADUs */
INT32 measured_bias[MAXIMAGES]; /* 1000*mean bias value for each
image */
INT32 measured_temperature[MAXIMAGES]; /* Temperature of each
detector in milliKelvins */
INT32 measured_pressure[MAXIMAGES]; /* Pressure of each chamber
in microTorr */
/* Retired reserve4 when MAXIMAGES set to 9 from 16 and two
fields removed, and temp and pressure added char reserve4[(32-
(5+3*MAXIMAGES))*sizeof(INT32)]; */
/* X-ray source and optics parameters (128 bytes) */
/* X-ray source parameters (8*4 bytes) */
INT32 source_type; /* (code) - target, synch. etc */
INT32 source_dx; /* Optics param. - (size microns) */
```

Copyright © 2000-2006. Mar USA

```
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; /* () */
char reserve_source[4*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))];
/* 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_comments[512]; /* comments - can be used as desired */
char reserve6[1024-(128+128+64+(3*32)+512)];
/* Dataset parameters (512 bytes) */
char dataset_comments[512]; /* comments - can be used as desired
*/
char pad[3072-(256+128+256+(3*128)+1024+512)]; /* pad out to 3072
bytes */
} frame_header;
```

Index

Copyright © 2000-2006, Mar USA