

Speckle Instrument GUI - Linux User Guide

Dave Mills (rfactory@theriver.com) – February 2021

Table of Contents

1. Introduction.....	2
2. Installation.....	2
3. Graphical user interface.....	4
3.1 The main window.....	4
3.1.1 Configurations menu.....	5
3.1.2 Temperature menu.....	5
3.1.3 Set ROI's menu.....	6
3.1.4 Tools menu.....	7
3.1.5 Filter Transmission & Efficiency Curves.....	9
3.2 Mimic Diagram.....	12
3.3 Observations.....	13
3.3.1 Number of Image Sets to Acquire Per Target.....	14
4. Desktop layout.....	15
4.1 Recovery options.....	18
4.2 Controlled shutdown.....	19
5. Log files.....	19
6. Updating the Software.....	20
6.1 Recompiling the shared libraries (Non Ubuntu only).....	20
7. Changing hardware components.....	21
8. Database.....	22
9. Example FITS header.....	23
10. Command Line usage (experts only).....	25

1. Introduction

The Speckle Instrument GUI has been developed by Dave Mills (The Random Factory, Tucson, AZ) in collaboration with the Speckle Instrument PI (Steve Howell) and collaborators (Nic Scott, and Mark Everett - KPNO). It provides comprehensive control over each hardware component, and the ability to acquire single frame or Data cube (time series) Observations in multiple sequences, filters, binning factors, and image geometries. Observation metadata is logged in the image headers, and also in a local database.

2. Installation

The GUI and accompanying sources can be obtained from the Github repository.

```
cd $HOME  
git clone https://github.com/therandomfactory/speckle-control
```

These commands will place the files in the directory *\$HOME/speckle-control*. Although it is possible to install the software to a different location, this is not recommended as it will be necessary to manually change the location in some of the scripts included with the drivers.

Run the software and drivers installation script

```
cd $HOME/speckle-control  
sudo ./install
```

Configure the USB devices for rw access

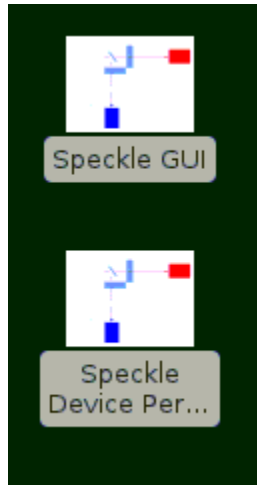
```
cd $HOME/speckle-control  
./setDevicePermissions
```

For a completely new installation, refer to section 7 of this guide.

Once this setup has been completed, the interface can be started with the command

```
~/speckle-control/startspeckle2
```

These USB permissions can also be set using the desktop icon, and the program can also be launched with an icon double-click as well.



To check for Speckle software updates , run the script,

`$HOME/speckle-control/checkUpdate`

if an update is available you will be offered the option to install it and the Andor camera drivers.

If an Andor SDK update is available , run the script,

`$HOME/speckle-control/andorUpdate`

which will download and install it, and rebuild any software as necessary.

3. Graphical user interface.

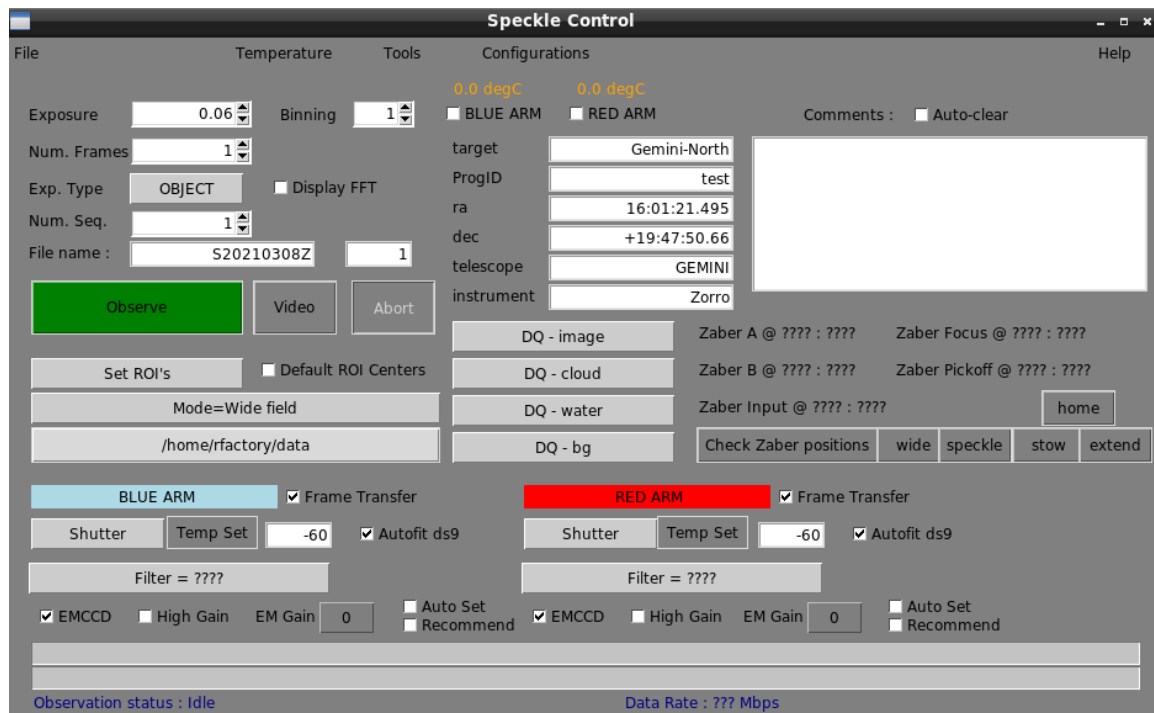
The graphical user interface provides easy access to the major functions such as image acquisition, temperature control, and device setup and configuration.

The program will open a small main window, and then create a message window which shows the progress of the system startup operations.

Once the message window closes, the system is ready for use. The cameras are initialized, and temperature control has been switched on.

Three terminal log windows appear at the bottom of the screen, left and right are per camera, and the central one is the general observation log.

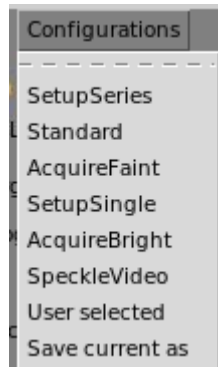
3.1 The main window



Most of the time the controls in this window will be the focus of observing activities.

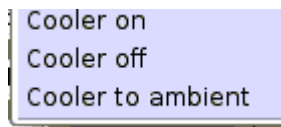
The following elements are provided :

3.1.1 Configurations menu



This menu provides quick setup for a range of commonly used observing or setup configurations. Each is a simple script (the sources can be found in *\$HOME/speckle-control/config-scripts*, and any new scripts which are added to this directory will be available as menu options after a GUI restart)

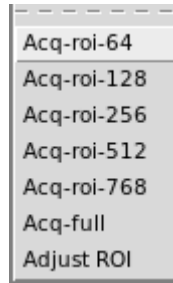
3.1.2 Temperature menu



This menu provides control over the camera cooler usage. Cooling may be switched on or off, and the “Cooler to ambient” option may also be selected (this is also automatically applied when the camera is shutdown).

The actual temperature set-points are individually controlled using entry boxes in the main window.

3.1.3 Set ROI's menu



This menu provides control over the data acquisition geometry. A range of ROI (Region of Interest) sizes can be selected, or the geometry can be reset to include the full frame. If an ROI is chosen, then an image will be taken with each camera, and a default ROI of the requested size will be automatically generated centered on the brightest target in the image(s). If it is necessary to manually adjust the calculated ROI's, then select that option and use the ds9 controls to move them, and then click OK on the dialog.

It is also possible to set default per camera ROI centers in the

andorsConfiguration.<telescope>

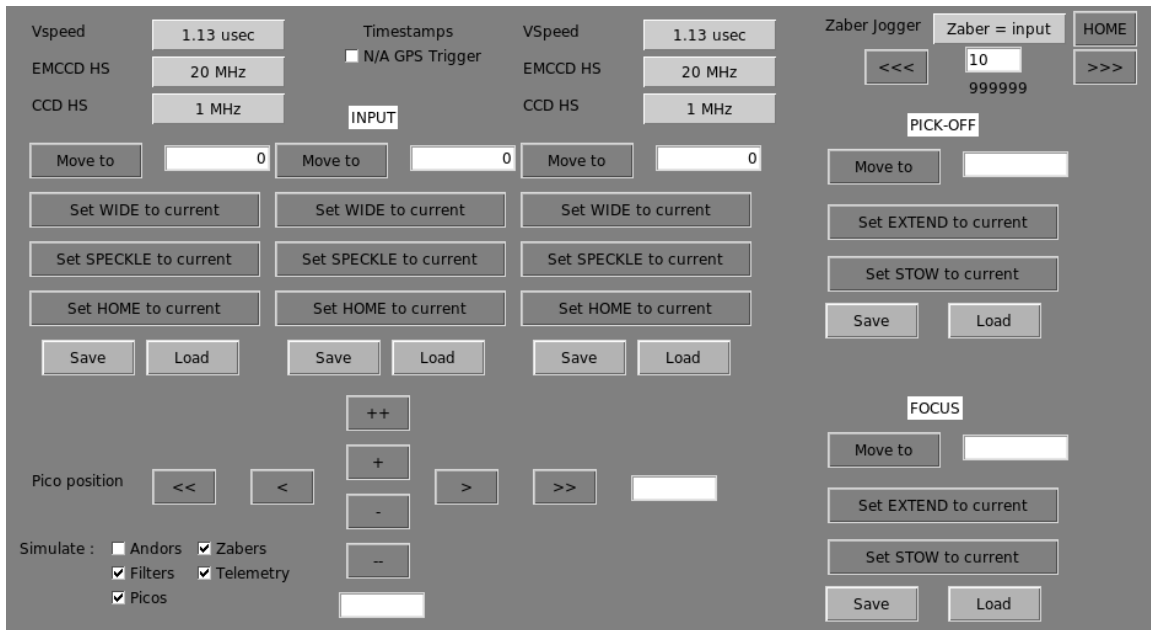
configuration file. If the *Default ROI Centers* checkbox is selected then the ROI's will be positioned so their centers are at the pixel coordinates specified in the configuration file.

3.1.4 Tools menu

This menu provides access to a set of commonly used options. There are four main types of item, GUI window visibility/mode, Zaber stage motions, pop-up windows, and resetting hardware components.

Tools	Configurations
Engineering	
Observing	
Filter Selection	
Camera status	
Plot timings	
ds9 red	
ds9 blue	
Audio notification on	
Audio notification off	
Allow seperate exposures	
Identical exposures	
Mimic diagram	
HOME all stages	
zabers to wide mode	
zabers to speckle mode	
zaber red wide	
zaber red speckle	
zaber blue wide	
zaber blue speckle	
zaber input wide	
zaber input speckle	
zaber focus extend	
zaber focus stow	
zaber pickoff extend	
zaber pickoff stow	
Reconnect Telemetry service	
Show Accumulate option	
Hide Accumulate option	
Reset Device Permissions	
Reconnect Filter Wheels	
Reconnect Cameras	
Reconnect Zabers	
Watch log	
View database	

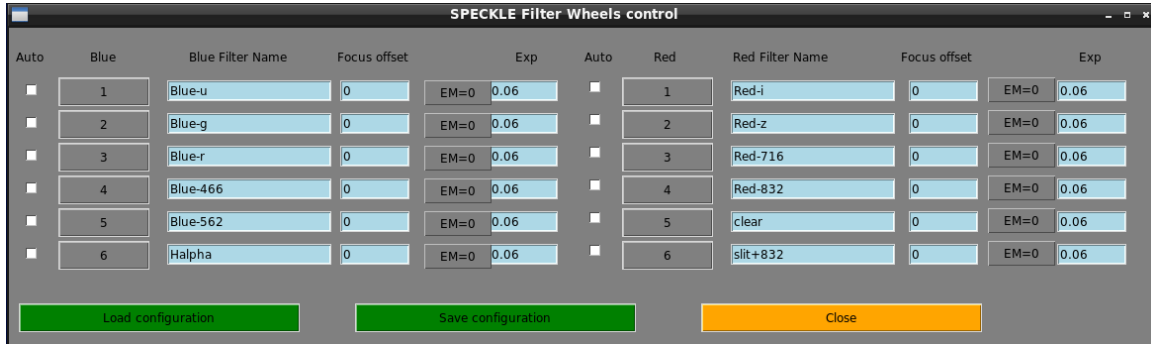
The *Engineering* option re-sizes the main window to make visible an extra set of controls generally used for equipment characterization and setup.



The detailed readout parameters of each camera can be manipulated, and the Zaber station positions edited and loaded/saved. For Gemini, extra controls for the Focus and Pickoff stages, and the a pico stage jogger are also included.

The *Observing* option returns the main window geometry to the default , hiding the Engineering controls.

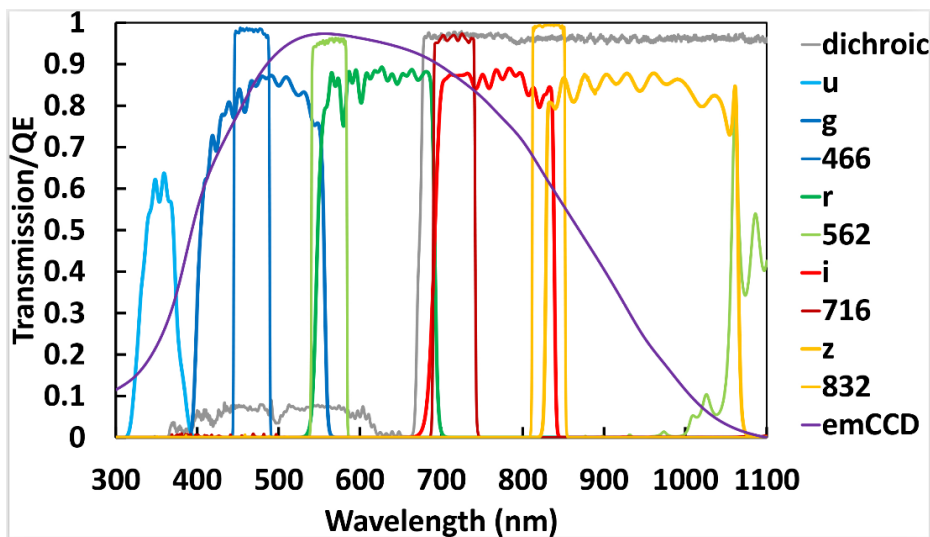
The *Filter Selection* option opens the Filter Wheel control window.



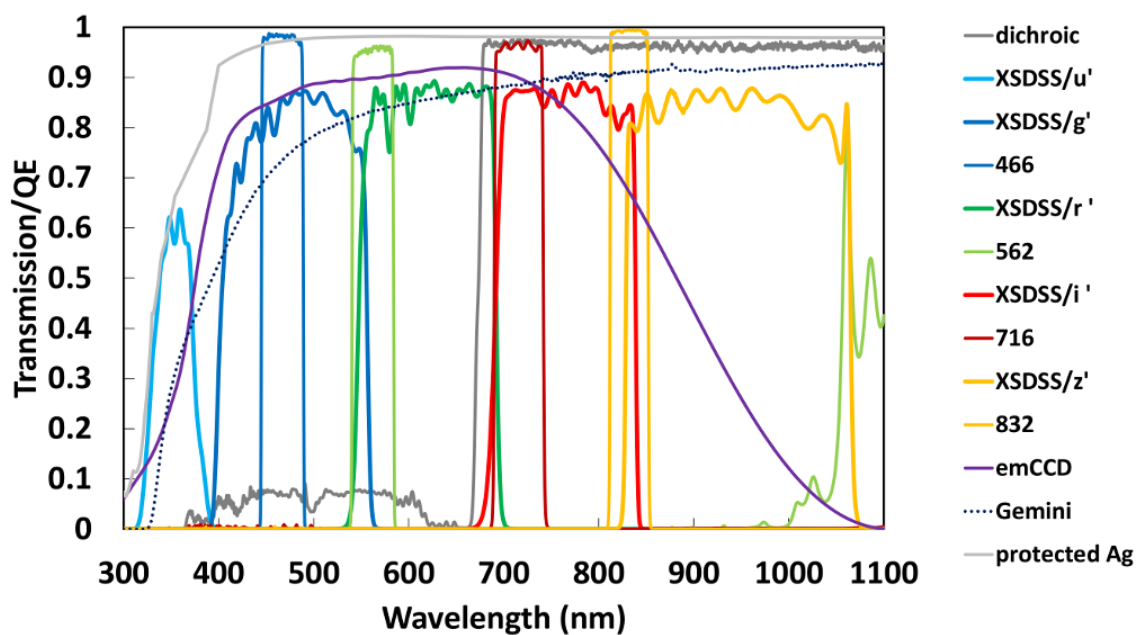
This provides options to rename filters and load/save the configurations. There is a placeholder for providing focus offset but this is not yet implemented. There are also options to set different exposure times for each filter; **this is an experimental feature and will not play well with the display of *Observation Status* progress bars.**

3.1.5 Filter Transmission & Efficiency Curves

Speckle instruments use a dichroic beamsplitter to separate the incoming light (at 686nm) into blue and red channels before focusing on the two identical cameras, which operate simultaneously. The speckle filter choice will be one of 467nm or 562nm paired with one of 716nm or 832nm. NESSI's SDSS filters are also listed below (although not used for speckle imaging). Data are in nanometers and fractional efficiencies as quoted by the manufacturer.



The two Gemini instruments are equipped with these filters.

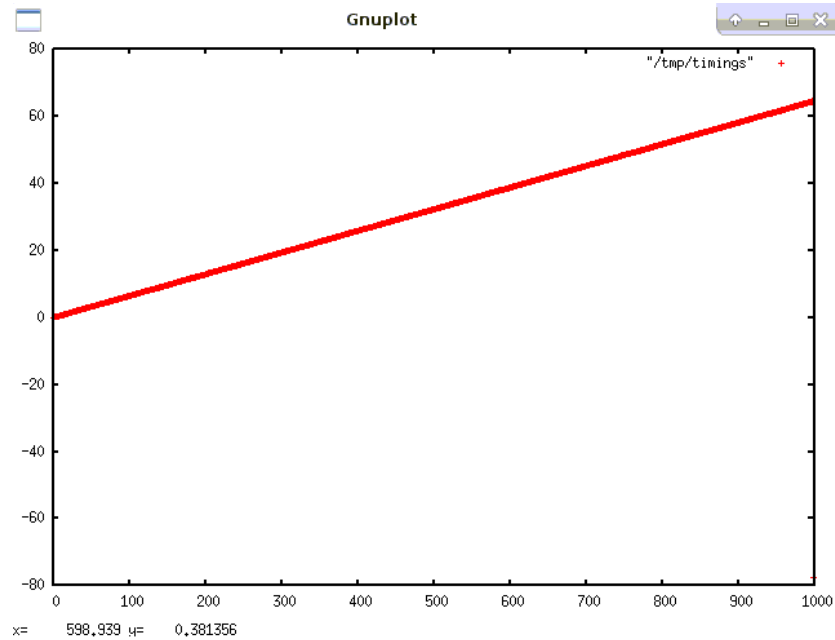


The *Camera Status* option opens a window showing the current settings of the main camera configuration and readout parameters. The values are updated after each

Camera Configurations		
	Red Arm	Blue Arm
Shutter	???	???
FrameTransferMode	???	???
OutputAmplifier	???	???
EMAdvanced	???	???
EMCCDGain	???	???
EMHSSpeed	???	???
HSSpeed	???	???
VSSpeed	???	???
PreAmpGain	1	1
ReadMode	???	???
AcquisitionMode	???	???
KineticCycleTime	???	???
NumberAccumulations	???	???
NumberKinetics	???	???
AccumulationCycleTime	???	???
TExposure	???	???
TAccumulate	???	???
TKinetics	0.04	0.04
Refresh		
Close		

observation, or when the *Refresh* button is clicked.

The *Plot timings* option opens a file selection dialog. Selecting a data cube image-name will plot the time history of that cube's exposures (delta times with 0 = 1st frame time).



The stored data in the FITS table is in seconds (system clock time)

The plotted data can also be examined in the file /tmp/timings after a plot.

The figure shows a Terminal window with the following text:

```
nessi:more /tmp/timings
0.0
0.06500005722045898
0.12899994850158691
0.1940000057220459
0.25800013542175293
0.3229999542236328
0.3880000114440918
0.4519999027252197
0.5160000324249268
0.5810000896453857
0.6459999084472656
0.7100000381469727
0.7739999294281006
0.8389999866485596
0.9030001163482666
0.9679999351501465
1.0329999923706055
1.0970001220703125
1.1610000133514404
1.2260000705718994
1.2899999618530272
1.3550000190734863
1.4189999103546143
1.4839999675750732
1.54800000972747803
1.6129999160766602
1.6770000457763672
1.7409999370574951
1.80599994277954
1.871000051498413
1.934999942779541
2.0
2.064000129699707
2.128000020980835
2.193000078201294
```

Options to open/close the Mimic diagram, Camera status, and Filters windows.

A set of the options to command the motion of the relevant Zaber stages to the requested position(s). Feedback on the positions can be seen in the Mimic diagram, and in the debug log window.

Other options control whether separate per filter exposure times are allowed (experimental), and control over resetting various hardware components in case of problems.

The *Show/Hide Accumulate* options control whether the corresponding control is visible.

Audio notifications (at end of observation) may be switched on or off.

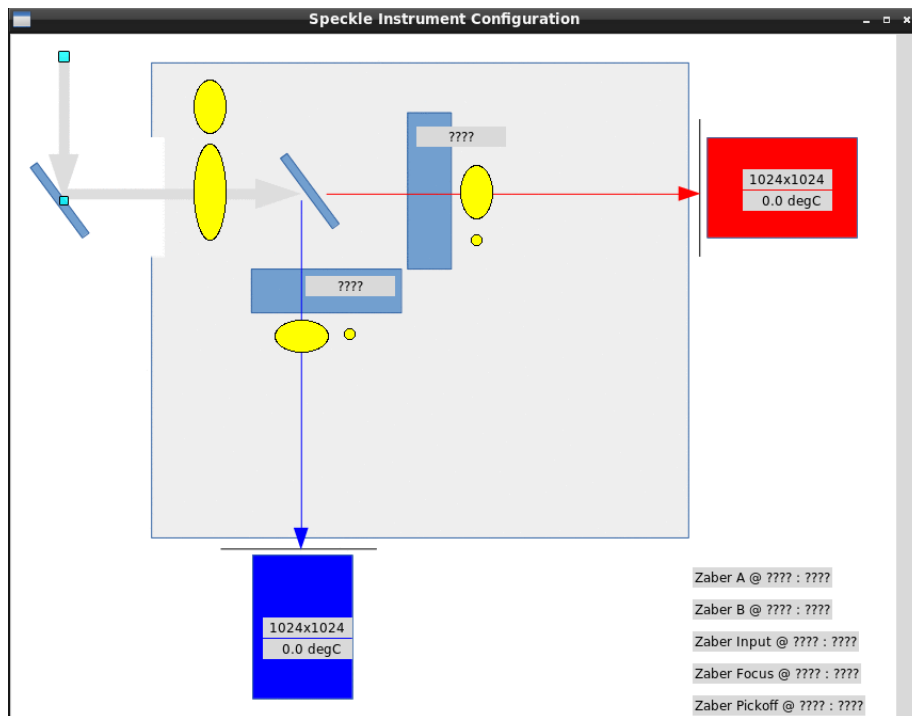
The *ds9red/ds9blue* options restart the corresponding image viewers if they have crashed.

The *Watch Log* option opens a log window which tracks the FITS headers of the observations.

The *View Database* option opens a terminal connected to the local observations database (accepts Mysql queries)

3.2 Mimic Diagram

The Mimic diagram window shows the at-a-glance configuration of the instrument.



3.3 Observations

The top left section of the main window contains a group of controls related to the sequencing and initiation of observations.

The *Exposure* time can be specified (in seconds) using the spinbox , or a value can be typed into the entry box area.

The *Num Frames* to take can be specified using the spinbox, or a value can be typed into the entry box area. If greater than 1, this specifies the number of exposures in each datacube. In this mode an array of (TAI) timing information about the exposures is also included in the FITS file as a Binary table Extension.

The *Exp Type* menu can be used to select common exposure types. Dark, Flat, etc.

This has little effect except over the shutter control, but the type is recorded in the image headers.

The *Num Seq* spinbox can be used to repeat a set of observations multiple times.

The *Accum* spinbox can be used to select the number of exposures to be accumulated before each camera readout. The exposures are thereby “co-added” by the camera (the visibility of this option is selected using the Tools menu options). This is normally used in conjunction with time series (datacube) operations.

The *File name* entry box is used to specify the base name for the FITS files. It will be expanded to add Sequence and Frame number where appropriate as the files are stored.

The current frame number is shown to the right , and will auto-increment as data is taken.

The *Observe* button start a sequence of observations (can also be just a single frame).

The *Video* button starts a display only sequence , it must be canceled using the *Abort* button before data acquisition Observations can commence.

The *Binning* spinbox controls the binning factor in both x and y dimensions.

The *Display FFT* option chooses whether to display the raw image data, or to display an FFT of the data instead.

The right side of the main window is focused on the meta data which will be included in the FITS headers. Some of this is automatically populated with data from the Telescope telemetry services. There are also menus for selecting a variety of Data Quality (DQ) specifications, and a comments area (this area may be flagged to auto-clear after each exposure if required). These are added to the FITS headers.

The current state of each camera (enabled, temperature) is prominently displayed top-center of the main window. Checkboxes can be used to enable/disable each camera.

A set of buttons provides quick access to check and set the Zaber stage positions.

The lower section of the window contains the major camera settings controls.

From here, the temperature set-point, Filter, Shutter state, Frame Transfer mode, EM mode and gain can be changed. There are also options to enable EM gain advisory popups, and to Auto set the gain. A checkbox controls the display of the images in ds9 which can be set to autofit or not (zoom to displayed frame size)

The bottom of the window has two progress bars, the upper shows the progress of a sequence of frames in a datacube image, and the lower shows the number of sequences completed.

3.3.1 Number of Image Sets to Acquire Per Target

The performance of speckle imaging is quite sensitive to conditions like seeing, so there are no strict rules to follow for determining the ideal number of image sets to acquire on a target of given brightness.

Observers targeting stars fainter than $V=13$ should plan on acquiring multiple image sets and those observing brighter stars may also benefit from taking multiple sets. This will depend on how they balance better contrast depth/image quality vs. number of targets visited. Multiple image sets per star can also help under less than optimal observing conditions and, given the several minutes needed to set up observing of each new target, many users may want to devote comparable time to exposures.

Each image set requires 1 minute of telescope time. Acquiring a target with a short slew requires 3 minutes and with a long slew, 5 minutes. Since a science target requires a point source observation, additional time is needed for that (about 4 minutes). Refer to the guide on estimating observing time for more information.

Note that we have found the signal-to-noise ratio for detecting secondary sources in speckle images does not grow as rapidly with exposure time as it would in traditional CCD imaging (ie. with the square root of time). Proposers may not expect to achieve the same contrast limits on faint stars as bright ones (5 magnitudes may be achievable on 12th magnitude stars and 3 magnitudes on 14th magnitude stars.) The table below only suggests numbers of image sets to take for various magnitude stars:

V or R # image sets	
<12	1-3
12-12.5	3
12.5-13	5
13-13.5	7
>13.5	9

(Mark Everett (everett@noao.edu)).

4. Desktop layout

A recommend desktop layout is illustrated below. The main visible components are

ds9red image viewer for the Red arm camera images.

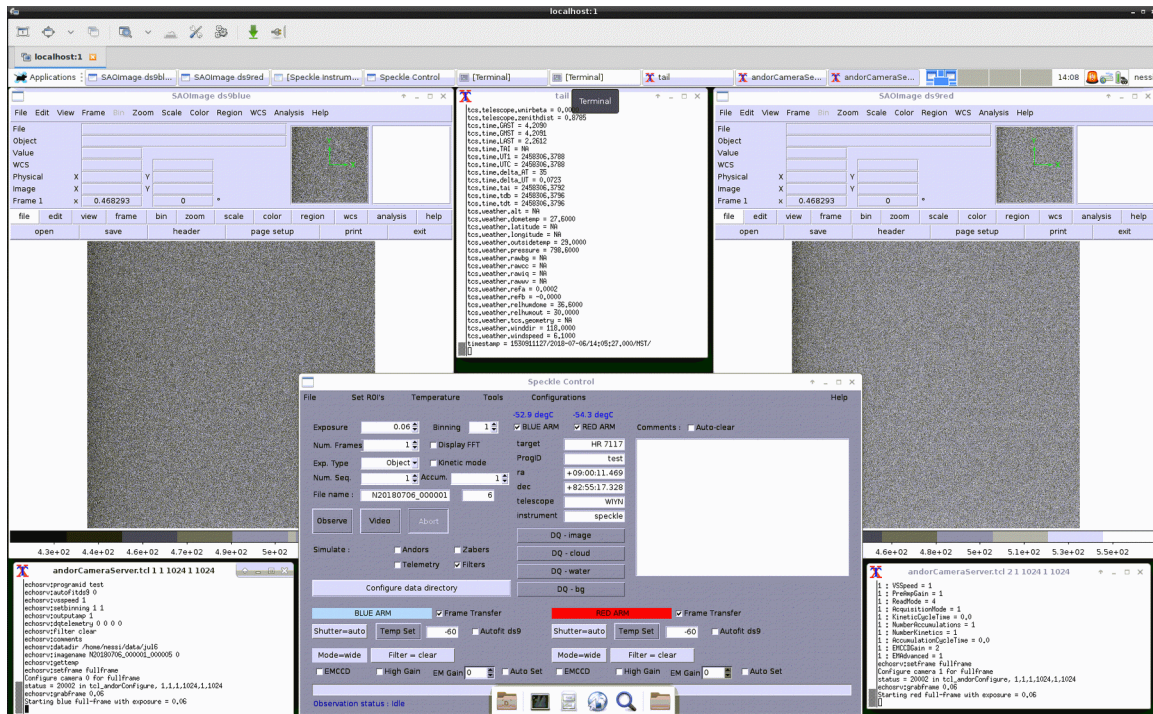
ds9blue image viewer for the Blue arm camera images.

Top-center xterm showing the debug log.

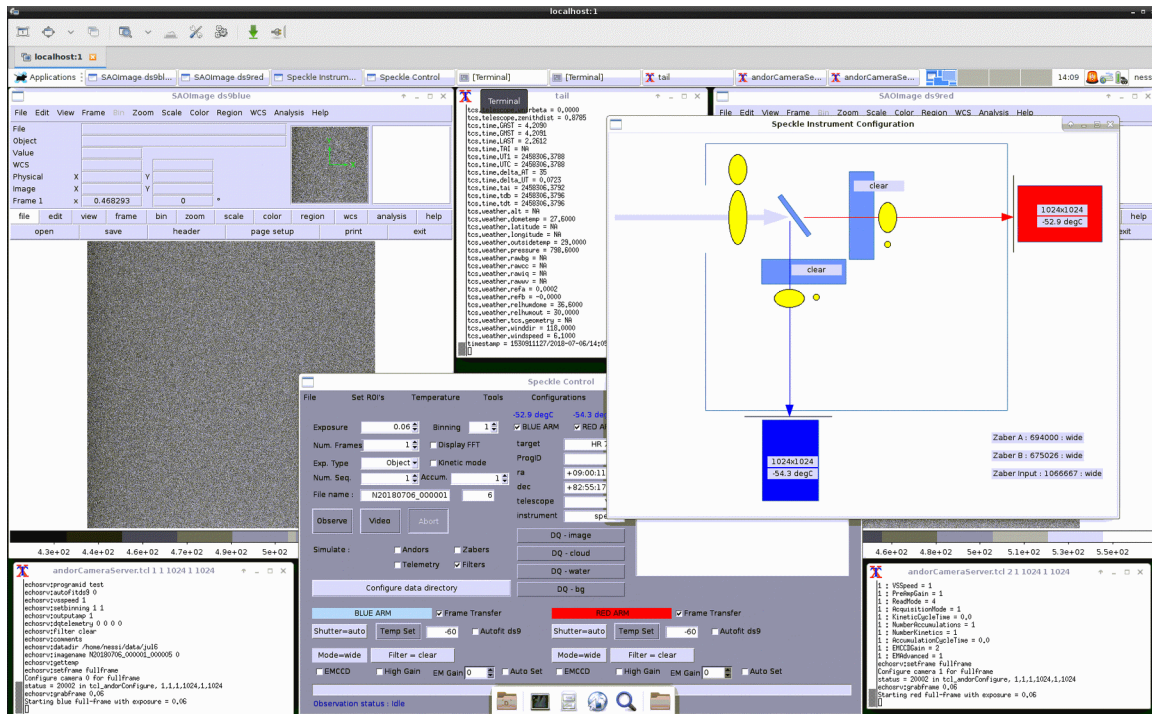
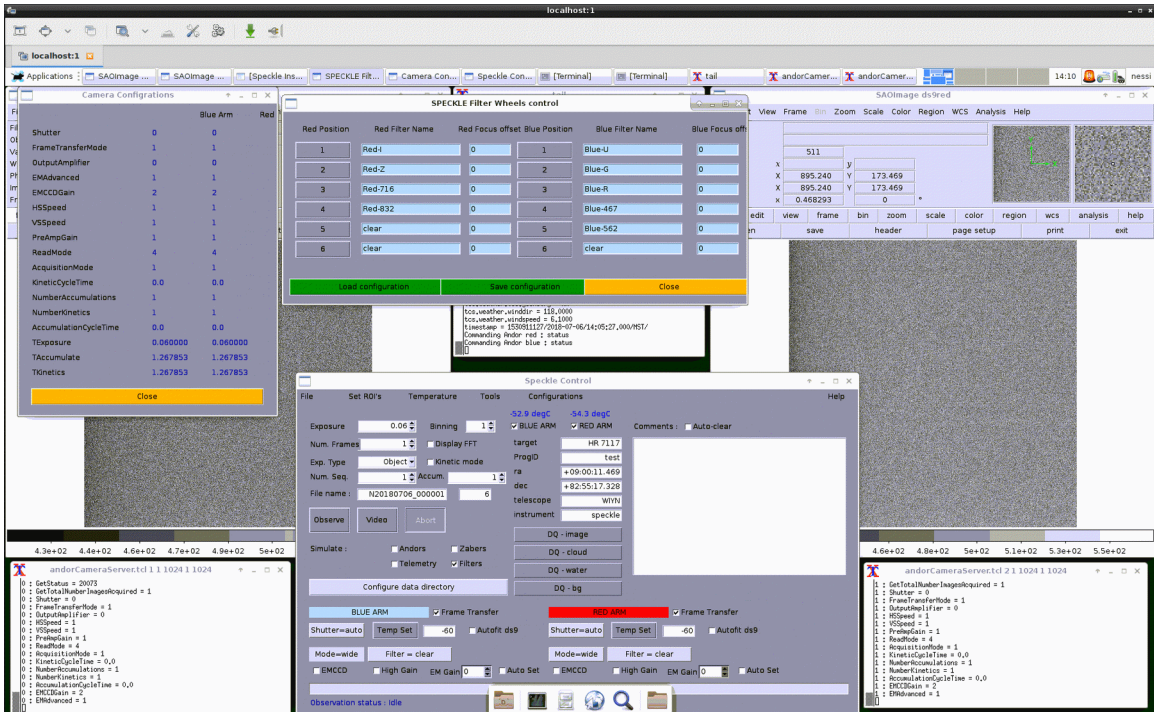
Lower left xterm showing the Red camera server operations

Lower right xterm showing the Blue camera server operations.

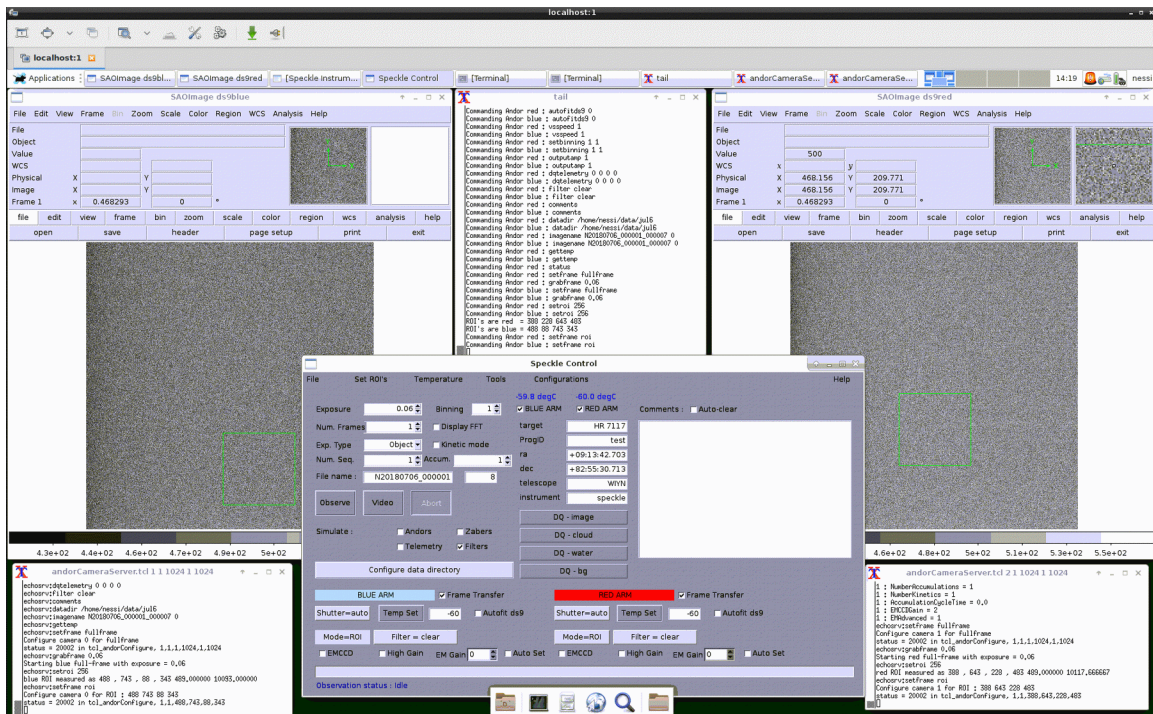
Mid-screen main GUI window.



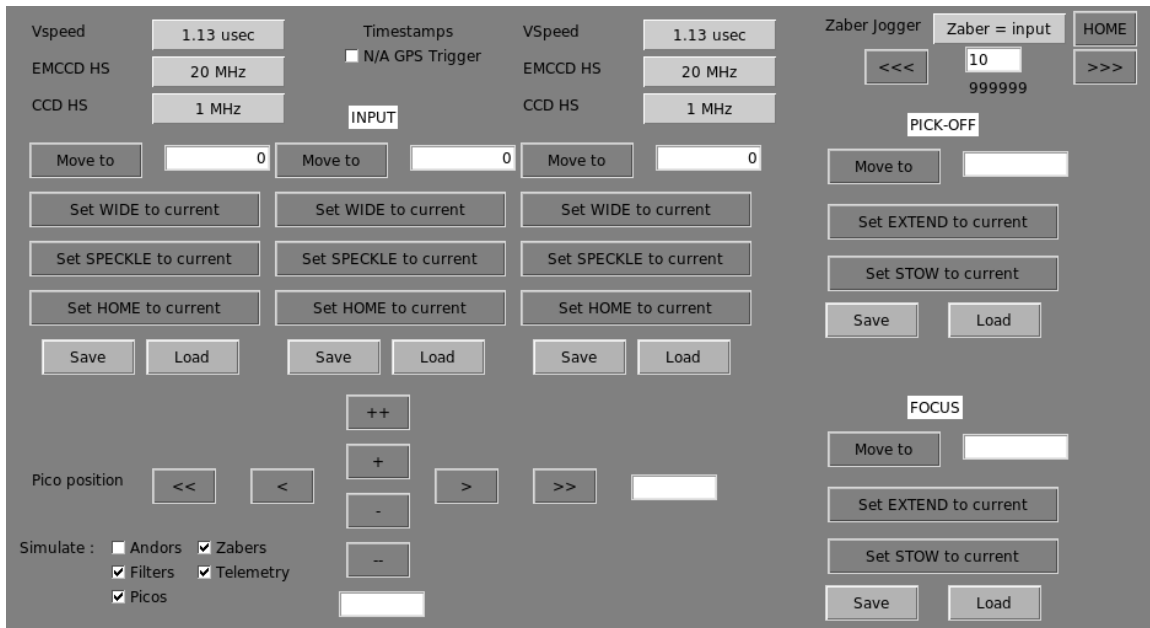
The following example also show typical popup windows for the Mimic diagram and Filter Wheels and Camera status windows.



This desktop shows an example of the appearance after an ROI selection , each ds9 window shows the chosen region, and the numeric parameters can be seen in the camera server log windows.



The Engineering mode (Tools menu) adds a set of low level controls to the bottom of the main GUI window



The detailed exposure modes for each camera can be set (Vertical transfer speed, Readout clock speed for EM and Normal readout modes, Horizontal transfer speed).

Low level control of the Zaber stages, and the ability to load and save new defaults.

For Gemini installations, control of the Focus and Pickoff stages, and a fine adjustment “jogger” for the Pico stage position.

In the lower left corner there is also a set of checkboxes to choose a Simulation mode for hardware which is non-operational or absent.

There is also a placeholder for a future feature to select GPS hardware timestamps (currently each datacube image also contains a FITS table containing the system clock times (double precision seconds) of the start of Acquisition time of each frame. These may be examined using the *Plot Timings* option on the Tools menu.

4.1 Recovery options

If you experience problems with operation of the software or particular hardware components, there are a set of options in the *Tools* menu. These can be used to Reconnect to the telemetry server, cameras, filter wheels, or zaber stages.

There is also a cleanRestart icon and script which will shutdown the software, power cycle the filter wheel, cameras and zaber stages, and then restart the software.

4.2 Controlled shutdown

To shutdown the software, select *Shutdown* from the *File* menu. This will command the cameras to ramp up to ambient temperature and then close the camera control servers and GUI components.

5. Log files

All GUI controlled operations are logged to disk. The files are named according to the system clock at the startup and stored in the /tmp directory.

```
nessi:ls /tmp/speckl*.log
/tmp/speckleLog_1530322241.log /tmp/speckleLog_1530502749.log
/tmp/speckleLog_1530322463.log /tmp/speckleLog_1530502891.log
/tmp/speckleLog_1530322723.log /tmp/speckleLog_1530503583.log
/tmp/speckleLog_1530323111.log /tmp/speckleLog_1530504217.log
/tmp/speckleLog_1530323495.log /tmp/speckleLog_1530504290.log
/tmp/speckleLog_1530323737.log /tmp/speckleLog_1530504332.log
/tmp/speckleLog_1530324010.log /tmp/speckleLog_1530504425.log
/tmp/speckleLog_1530324888.log /tmp/speckleLog_1530504611.log
/tmp/speckleLog_1530325368.log /tmp/speckleLog_1530504737.log
/tmp/speckleLog_1530325752.log /tmp/speckleLog_1530504795.log
/tmp/speckleLog_1530326089.log /tmp/speckleLog_1530505286.log
/tmp/speckleLog_1530327038.log /tmp/speckleLog_1530505602.log
/tmp/speckleLog_1530327151.log /tmp/speckleLog_1530506620.log
/tmp/speckleLog_1530332355.log /tmp/speckleLog_1530508085.log
/tmp/speckleLog_1530335084.log /tmp/speckleLog_1530508483.log
/tmp/speckleLog_1530335813.log /tmp/speckleLog_1530511001.log
/tmp/speckleLog_1530336849.log /tmp/speckleLog_1530511098.log
/tmp/speckleLog_1530398480.log /tmp/speckleLog_1530511224.log
/tmp/speckleLog_1530398755.log /tmp/speckleLog_1530546285.log
/tmp/speckleLog_1530398794.log /tmp/speckleLog_1530546434.log
/tmp/speckleLog_1530398814.log /tmp/speckleLog_1530548907.log
/tmp/speckleLog_1530399172.log /tmp/speckleLog_1530553512.log
/tmp/speckleLog_1530399516.log /tmp/speckleLog_1530553964.log
/tmp/speckleLog_1530403411.log /tmp/speckleLog_1530554786.log
/tmp/speckleLog_1530403900.log /tmp/speckleLog_1530556213.log
/tmp/speckleLog_1530404046.log /tmp/speckleLog_1530556852.log
/tmp/speckleLog_1530405721.log /tmp/speckleLog_1530557089.log
/tmp/speckleLog_1530407045.log /tmp/speckleLog_1530557647.log
/tmp/speckleLog_1530410430.log /tmp/speckleLog_1530560460.log
/tmp/speckleLog_1530410786.log /tmp/speckleLog_1530561133.log
/tmp/speckleLog_1530410917.log /tmp/speckleLog_1530561509.log
/tmp/speckleLog_1530411091.log /tmp/speckleLog_1530562077.log
```

6. Updating the Software

A *checkUpdates* desktop icon and script are provided to automatically check for Speckle software updates (on github) , and optionally download and install them, recompiling sub packages as needed. These also perform updates to the Andor drivers if needed.

You can also separately update just the Andor software using the *andorUpgrade* script.

6.1 Recompiling the shared libraries (Non Ubuntu only)

The default *install* script assumes a Ubuntu Linux system. If it is necessary to rebuild the compiled code on a different flavor of Linux, the only component which might prove challenging is the VIPS package.

Low level functionality is provided in C/C++ for speed , and this code is wrapped using tcl and loaded into the interpreter at runtime.

To move the code to a different version of Linux it may be necessary to recompile the libraries in the following directories.

andor
ccd
filterWheel
guider

Each has either a Makefile (*ccd*, *guider*,*vips*) or a set of build steps (*andor/buildlib* , *oriel/buildlib*)

The Vips library may present more difficulty due to it's many dependencies.

The package can be normally be recompiled using the GNU standard incantations

./configure --prefix=/home/speckle/speckle-control --without-python
make install

If the configure step does not work , try

make AUTOCONF=: AUTOHEADER=: AUTOMAKE=: ACLOCAL=:
make install AUTOCONF=: AUTOHEADER=: AUTOMAKE=: ACLOCAL=:

7. Changing hardware components

If it becomes necessary to change out either Filter Wheel or Camera components, the appropriate configuration files will need adjustment. The configuration files are in the `$HOME/speckle-control` directory

`andorsConfiguration.[telescope]`

`filtersConfiguration.[telescope]`

In each case the serial number information will need to be updated.

The Filter Wheel serial numbers can be found using the `lsusb` command

The other configuration files are less likely to need changes.

`picomotorConfiguration.[telescope]` controls Pico stage defaults

`wcsPars.[camera].[mode].[telescope]` control image world coord system

`zabersConfiguration.[telescope]` controls Zaber stage defaults

```
nessi:lsusb
Bus 001 Device 002: ID 8087:8001 Intel Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 003 Device 016: ID 05e3:0612 Genesys Logic, Inc. Hub
Bus 003 Device 017: ID 136e:0012 Andor Technology Ltd.
Bus 003 Device 018: ID 136e:0012 Andor Technology Ltd.
Bus 003 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 002 Device 003: ID 8087:0a2a Intel Corp.
Bus 002 Device 044: ID 104d:1011 Newport Corporation
Bus 002 Device 043: ID 104d:1011 Newport Corporation
Bus 002 Device 042: ID 0403:6001 Future Technology Devices International, Ltd FT
232 Serial (UART) IC
Bus 002 Device 041: ID 05e3:0610 Genesys Logic, Inc. 4-port hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
nessi:
nessi:
nessi:lsusb -v -s 002:043 | grep iSerial
    iSerial          128 061D088E010F5400
nessi:lsusb -v -s 002:044 | grep iSerial
    iSerial          128 1B18177A01135400
nessi:
```

8. Database

The camera servers automatically log information about each image to the a local database (Mysql). The database is named “speckle” and the table name is “Speckle_Observations”. It can be viewed using the *mysql* command line program. e.g.

```
mysql -user=root speckle
select * from Speckle_Observations LIMIT 10;
```

Or using a Mysql capable tool e.g. the TOPCAT gui

The TOPCAT GUI shows the 'Table List' with the query '1: select * from Speckle_Observations'. The 'Current Table Properties' panel displays the following information:

- Label: select * from Speckle_Observations
- Location: select * from Speckle_Observations
- Name: jdbc:mysql://localhost/speckle#select * from Speckle_Observations
- Rows: 5,818
- Columns: 25
- Sort Order: (indicated by a yellow arrow icon)
- Row Subset: All
- Activation Action: (no action) ☐ Broadcast Row

The 'SAMP' section shows 'Messages: 0' and 'Clients: 0'.

The 'Table Browser for 1: select * from Speckle_Observations' displays a table with the following columns: exposureStart, exposureEnd, Filter, amplifier, numex..., numAc..., window, colBin..., rowBin..., RA, Declination, dqImage, dqCloud, dqWater, and dqBG. The table contains 5,818 rows of data.

	exposureStart	exposureEnd	Filter	amplifier	numex...	numAc...	window	colBin...	rowBin...	RA	Declination	dqImage	dqCloud	dqWater	dqBG
5800	1.530573E9	1.530573E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:53:02.780	+87:52:38.838	0	0	0	0
5801	1.530573E9	1.530573E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:54:14.16	+87:52:39.23	0	0	0	0
5802	1.530573E9	1.530573E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:54:14.16	+87:52:39.23	0	0	0	0
5803	1.530578E9	1.530578E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+21:44:54.319	+88:24:28.219	0	0	0	0
5804	1.530578E9	1.530578E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+21:44:54.319	+88:24:28.219	0	0	0	0
5805	1.530915E9	1.530915E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+10:13:13.532	+82:56:18.104	0	0	0	0
5806	1.530915E9	1.530915E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+10:13:17.85	+82:56:18.141	0	0	0	0
5807	1.530917E9	1.530917E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+10:41:19.991	+82:56:33.363	0	0	0	0
5808	1.530917E9	1.530917E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+10:41:27.812	+82:56:33.423	0	0	0	0
5809	1.530918E9	1.530918E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+11:05:25.176	+82:56:42.548	0	0	0	0
5810	1.530918E9	1.530918E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+11:05:27.716	+82:56:42.561	0	0	0	0
5811	1.530919E9	1.530919E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:45:54.867	+88:27:46.997	0	0	0	0
5812	1.530919E9	1.530919E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:45:54.867	+88:27:46.997	0	0	0	0
5813	1.530919E9	1.530919E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:46:23.701	+88:27:47.28	0	0	0	0
5814	1.530919E9	1.530919E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:46:23.701	+88:27:47.28	0	0	0	0
5815	1.530920E9	1.530920E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:54:28.998	+88:27:47.687	0	0	0	0
5816	1.530920E9	1.530920E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+20:54:33.522	+88:27:47.694	0	0	0	0
5817	1.530920E9	1.530920E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+21:00:38.237	+88:27:48.277	0	0	0	0
5818	1.530920E9	1.530920E9	clear	ECMMD Amplifier	1	1	1.1024.1.1024	1	1	+21:00:43.47	+88:27:48.295	0	0	0	0

9. Example FITS header

SIMPLE = T / file does conform to FITS standard
BITPIX = 16 / number of bits per data pixel
NAXIS = 3 / number of data axes
NAXIS1 = 128 / length of data axis 1
NAXIS2 = 128 / length of data axis 2
NAXIS3 = 1000 / length of data axis 3
EXTEND = T / FITS dataset may contain extensions
COMMENT FITS (Flexible Image Transport System) format is defined in 'Astronomy
COMMENT and Astrophysics', volume 376, page 359; bibcode: 2001A&A...376..359H
BZERO = 32768 / offset data range to that of unsigned short
BSCALE = 1 / default scaling factor
CREATOR = 'Linux ANDOR CCD control V2' / Speckle Data-taking program
MJD-OBS = 58998.864815 / MJD at start of obs
JD = 2458999.36481 / Julian Date at start of obs
HDR_REV = '4.00 10-Feb-2020q' / Header-Rev
UTC = '20:45:19.9' / UTC time at end of observation
LAST = '02:54:46.6' / LSTHDR local sidereal time
EPOCH = 2000.000000 / EPOCH of telescope coords
TARGRA = 75.000000 / Telescope target RA
TARGDEC = 89.970000 / Telescope target DEC
RA = '02:53:48.402' / Telescope RA
DEC = '+19:39:24.95' / Telescope DEC
RAOFFST = -1296000.0000 / Telescope RA offset
DECOFFST = -0.000000 / Telescope DEC offset
ZD = 0.000100 / zenith distance
AIRMASS = 1.000000 / airmass at start of exposure
FOCUS = '-0.091 ' / Telescope focus (microns)
OBJECT = 'Zenith ' / Object name
PA = 0.000000 / Instrument position angle
IAA = 270.000000 / Instrument Alignment angle
CRPA = -180.000000 / Current Cass Rotator Position Angle
GUIDING = 'Off ' / Guide state
OBSERVAT = 'Gemini-North' / Originating Observatory
RELEASE = '2020-12-01' / End of proprietary period YYYY-MM-DD
INSTRUME = 'Alopeke ' / Instrument name
DATE-OBS = '2020-05-29' / Date of start of observation in UTC
TIMEOBS = '20:43:46' / from speckle.scope.timeobs
OBSTYPE = 'Object ' / Type of picture (object, dark, etc.)
TELESCOP = 'Gemini-North' / Specific system
RECID = '2020-05-29_Gemini-North' / archive ID for observation
ACT = 0.008544 / Accumulation cycle time
EXPTIME = 0.007369 / Frame Exposure time in seconds
OBSTIME = 1590785111.7608 / UTC at start of observation
EXPENDTM = 1590785120.6196 / UTC at end of observation
HEAD = 'Andor iXon Emccd' / Head model
ACQMODE = 'Kinetics mode' / Acquisition mode

KCT = 0.008544 / Kinetic cycle time
 KINMODE = '1' / Kinetic Mode
 NUMEXP = '1' / from speckle.scope.numexp
 READMODE= 'Image' / Readout mode
 IMGRECT = '1,1024,1,1024' / Image format
 EMCCD = 'On' / EMCCD mode
 FRAMEXFR= 'On' / Frame Transfer mode
 BLCLAMP = 'On' / Bias Clamp
 HBIN = 1 / Horizontal binning
 VBIN = 1 / Vertical binning
 SUBRECT = '152,279,1,128' / Subimage format
 EXPOSURE= 7.369000 / Total Exposure Time
 EMGAIN = 0.000000 / EM Real Gain
 VSSPEED = 1.130000 / Vertical Speed (usec)
 HSSPEED = 1.000000 / Horizontal Speed (MHz)
 OUTPTAMP= 'Electron Multiplying' / Output Amplifier
 PREAMP = 1 / Pre Amplifier Gain
 SERNO = 'X-10405' / Serial Number
 UNSTTEMP= -60.000000 / Unstabilized Temperature
 ACCUM = 1 / Accumulations per frame
 NUMKIN = 1000 / Number of kinetic frames
 FILTER = 'Blue-562' / Filter name
 FLDZABER= 'wide' / Field zaber position
 INPZABER= 'wide' / Input zaber position
 CCDTEMP = -28.049999 / CCD temperature
 RAWIQ = '0' / Image quality
 RAWCC = '0' / Cloud cover
 RAWWV = '0' / Water vapour
 RAWBG = '0' / Background
 HUMIDITY= '16.4' / from tcs.weather.humidity
 OBSID = 'GN-2020A-FT-204' / Observation-ID
 GEMPRGID= 'GN-2020A-FT-204' / Observation-ID
 SPKLESEQ= 2 / SPECKLENFO-sequence
 SPKLEDAT= 'May 29 10:45:21' / SPECKLENFO-timestamp
 CRVAL1 = 19.656667 / Declination of reference pixel [deg]
 CRVAL2 = 43.450000 / RA of reference pixel [deg]
 CTYPE1 = 'DEC--TAN' / Coordinate type
 CTYPE2 = 'RA--TAN' / Coordinate type
 CRPIX1 = 64 / Coordinate reference pixel in X
 CRPIX2 = 64 / Coordinate reference pixel in Y
 CDELTA1 = 0.000000000000 / Rotation angle
 CDELTA2 = -0.000020138889 / Coordinate pixel scale in Y
 CD1_1 = 0.000020138889
 CD1_2 = 0.000000000000
 CD2_1 = 0.000000000000
 CD2_2 = -0.000020138889
 WCSNAME = 'FK5' / World coordinate system type


```
RADECSYS= 'FK5' / Default coordinate system type  
END
```

10. Command Line usage (experts only)

There is a rich set of commands to allow interactive and scripted usage.

To access the command line it is necessary to source the *setup<telescope>* script from the speckle-control directory and then start the GUI manually e.g.

```
source setupGeminiS  
wish  
source gui-scripts/gui2.tcl
```

The following commands are available

Filter Wheel :

```
loadFiltersConfig [filename]  
saveFiltersConfig [filename]  
echoFiltersConfig  
selectfilter arm filter-number  
findWheels  
resetFilterWheel arm
```

Zaber stages :

```
loadZaberConfig [filename]  
saveZaberConfig [filename]  
echoZaberConfig  
zaberPrintProperties  
zaberConnect  
zaberDisconnect  
homeZabers  
zaberCheck  
zaberSetPos name position  
zabersStopAll  
zaberGoto name station
```

Pico Stages : **Gemini only**

loadPicosConfig [filename]
savePicosConfig [filename]
echoPicosConfig
picosConnect
picoCommand axis cmd
picoSet axis parameter value

Andor Cameras :

Commands may be issued from the GUI command line, scripted , or optionally by *telnet* to ports 2001, 2002 (also available when started in normal GUI mode).

When using the command line the syntax is

commandAndor arm “command and parameters” (where *arm* is red or blue)
or *commandAndors “command and parameters”*

When using the *telnet* connection, just type the command and parameters

accumulationcycletime seconds
acquisition index
autofitds9 0/1
baseclamp 0/1
comments comment1|comment2|...
configure hbin vbin vstart vend hstart hend preamp vsspeed ccdhss emccdhs
datadir data-directory
dqtelemetry rawiq rawcc raqvv rawbg
emadvanced index
emccdgain 0/1
fastVideo exposure xs ys dim
fitsbits data-format
forceroi xs xe ys ye

frametransfer index
gettemp
grabcube exposure xs ys dim
grabframe exposure
grabroi exposure xs ys dim
hsspeed amp index
imagename image-name
kineticcycletime seconds
locatestar smooth dim
numberaccumulations count
numberkinetics count
outputamp index
positiontelem input-zaber field-zaber filter
preampgain index
programid program-id
readmode index
reset mode
setexposure seconds
setframe mode
setroi mode
settemperature degrees
shutdown
shutter index
status
version
vsamplitude index
vsspeed index
whicharm