KING – 70cm Telescope

—

RoCon Software Manual

Doc. No.	KING-MPIA-DOC-DET-004
Short Title	KING 70 cm Rocon
Issue	4.110
Date	April 20, 2023

Prepared Richard J. Mathar (mathar@mpia.de, ☎382) April 20, 2023
Name Date Signature

Approved N. Surname
Name Date Signature

Released N. Surname
Name Date Signature

Change Record

Issue	Date	Sect.	Reason/Initiation/Documents/Remarks
1.055	2020-02-24	all	created
4.110	April 20, 2023	all	SVN version 74

Contents

List of Figures

1 OVERVIEW

KING is a telescope with a 70 cm (27 inches) diameter of the primary at the east dome of the Elsässer Laboratory on the MPIA campus. This manual describes aspects of maintaining and using the software of the readout electronics (ROE) of the CCD camera. It is available in the ~ccd/king/sw/ccd/doc and /opt/ccd/share/doc directories of the KING computer, and the SVN under https://svn.mpia.de/gulli/king/sw/ccd/doc/ for people who have an account there.

1.1 Acronyms

ADC	analog-to-digit conversion
CCD	Charge Coupled Device
DHCP	Dynamic Host Configuration Protocol
EOC	End of Conversion (of an ADC)
ESO	European Southern Observatory http://www.eso.org
FITS	Flexible Image Transport System http://fits.gsfc.nasa.gov
FOV	field of view
FPGA	Field programmable gate array
FTP	File Transfer Protocol https://en.wikipedia.org/wiki/File_Transfer_Protocol
GUI	Graphical User Interface
HDU	header-data unit (of FITS)
HEASARC	High Energy Astrophysics Science Archive Research Center https://heasarc.gsfc.nasa.gov/
IDL	Interactive Data Language http://www.uni-giessen.de/hrz/software/idl/
KING	Königstuhl Instrument zum in den Nachthimmel Gucken https://svn.mpia.de/trac/gulli/king
LED	Light Emitting Diode
LST	Local sidereal time
MPIA	Max-Planck Institut für Astronomie, Heidelberg https://www.mpia.de
PDF	Power Density Function https://en.wikipedia.org/wiki/Spectral_density
PLX	PLX Technology, http://www.broadcom.com/products/pcie-switches-bridges/software-dev-kit
RAM	Random Access Memory
ROE	Readout Electronics
RPM	RPM Package Manager http://docs.fedoraproject.org/en-US/Fedora_Draft_Documentation/0.1/html/RPM_Guide/index.html
SQL	Structured Query language https://en.wikipedia.org/wiki/SQL
SSD	solid state disk https://en.wikipedia.org/wiki/Solid-state_drive
SVN	Subversion http://subversion.apache.org
UTC	Universal Time Coordinated

VNC Virtual Network Computing
https://en.wikipedia.org/wiki/Virtual_Network_Computing

1.2 References

References

[1] I. F. W. Group, Definition of the flexible image transport system (FITS), version 4.0 (2016).
URL https://fits.gsfc.nasa.gov/fits_standard.html

[2] A. H. Rots, P. S. Bunclark, M. R. Calabretta, S. L. Allen, R. N. Manchester, W. T. Thompson, Representation of time coordinates in FITS. time and relative dimension in space., Astron. Astrophys. 574 (2015) A36. doi:10.1051/0004-6361/201424653.

[3] F. Hormuth, Astralux, high angular resolution astrometry with an electron multiplying ccd, Master's thesis, Max Planck Institute of Astronomy, Heidelberg (2007).
URL http://www.mpia.de/ASTRALUX/Papers/AstraLux_Diploma_Thesis_2007.pdf

[4] M. W. Buie, General analytical telescope pointing model, $q = T \sin\varphi + x\cos\delta$ is missing in (6) (16 Feb. 2003).
URL https://www.boulder.swri.edu/~buie/idl/downloads/pointing/pointing.pdf

[5] E. J. Spillar, D. Bumbrill, G. L. Grasdalen, R. R. Howell, The wyoming infrared observatory telescop software system, Pub. Astr. Soc. Pac. 105 (1993) 616–624. doi:10.1086/133205.

[6] A. V. Filippenko, The importance of atmospheric differential refraction in spectrophotometry, Publ. Astron. Soc. Pac. 94 (1982) 715–721.

[7] J. R. Ramos, ROCON REad-out Controller Board (Nov. 2009).
URL webdavs://sk1/geirs/roe3MPIA/Roconv3-Draft.pdf

2 OBSERVER

2.1 Basics of Operation

The KING is a telescope which has no interface to any type of local or remote concurrent computer standards. In fact the control is embodied by a 1980's standalone system of electronics; the observer moves the telescop and dome with the handles and control desk in the dome. There is no such thing as a software that supports operating the telescope.

This manual describes the readout process of the CCD—at the Cassegrain focus of the telescope—, which is essentially controlled by (i) an Ethernet connection from a dedicated computer named `king` to a readout electronics and (ii) a standalone data transfer line (a dedicated fiber pair) back to the computer. So *that* aspect of the operation merely requires access to the shell of that dedicated computer.

Finally, data reduction (the process of extracting information from the FITS images generated with the CCD control software) may be done on any computer, supposing the data have been copied there.

2.2 Info pages

Online man pages are available with the command

```
info king
```

provided by the texinfo tool set, if a line like

```
export INFOPATH=${INFOPATH}:/opt/ccd/share/info
```

was added to the `.bashrc` file. The first pages of that content are reproduced here:

KING RoCon Command Interface

King 70cm Telescope Software
Max-Planck Institute of Astronomy, Heidelberg 22 June 2022

Richard J. Mathar mathar@mpia-hd.mpg.de

Table of Contents

ambiTemp.py

syntax: `ambiTemp.py`

This is a python program which reads the temperature of the LabJack sensor in the dome in Kelvin. It prints the value, converted to degrees Celsius, to the standard output.

astron2hdr

This program inserts the information obtained from `nova.astrometry.net` into the primary header of a FITS file. If the program is called with two command line arguments, the first is the ASCII or FITS file with the astrometry information and the second the FITS file to be updated.

WCS from fits

syntax: `astron2hdr astro.wcs fitsimg.fits`

If the first command line argument is the name of an existing file with the suffix `.wcs` and the second command line is the name of an existing file, the program assumes both are FITS files, where the first file has been created with the `solve-field` program of the astrometry.net package. The program copies the values of the `CRVAL`, `CUNIT`, `CD` and `CRPIX` keywords of the primary header of the first FITS file to primary header of the second FITS file.

WCS from ASCII

syntax: `astron2hdr astro.asc fitsimg.fits`

syntax: `astron2hdr fitsimg.fits < astro.asc`

If the program is called with one command line argument or the first command line argument does not end on `.wcs`, it must be the FITS file to be updated and the information must be fed into the program from the standard input.

The program scans the ASCII input for two types of lines:

- Lines that contain the string `RA,Dec` or `CRVAL` followed in the same line by an opening-closing pair of parentheses. The two numbers in the parentheses (separated by a comma) are interpreted as the right ascension and declination in degrees.
- Lines that contain the string `Pixel scale:` or `pixel scale` followed by a number which is interpreted as a pixel scale in arcseconds per pixel. If no such line is found in the astrometric solution but the FITS header contains a `XBINNING` keyword, a pixel scale of this binning factor times 0.554 arcseconds is assumed. If this keyword is not in the FITS header, a binning factor of 2 is assumed.
- Lines that contain the string `CRPIX` followed on the same line by an opening-closing pair of parentheses. The two numbers in the parentheses are interpreted as the position of the two equatorial coordinates in the FITS image. If that coordinate pair is not in the ASCII input, the program assumes that this is in the middle of the FITS image.

So this is tuned towards an input that is copied-and-pasted from the final log lines shown in the astrometry.net solution.

The program assumes a 2.61 degree rotation of the image away from the North=up orientation. It does not try to use the `CD` specification that might be found in the astrometric solution because that is usually containing only two digits.

ccdcom

syntax: `ccdcom [-v] [-d] [--] nr1 nr2 nr3 ...`

Logs into the RoCon, sends a single bare command line, and logs out.

Obviously, using this requires a deep understanding of how the patterns and voltages are applied to the control electronics of the CCD.

The option `-v` may be used to let the program generate more verbose output of what it is doing.

The option `-d` is a developer's aid to debugging, which means it uses the file `/tmp/com.log` for loging purposes and uses the file `comflag` in the current working directory to indicate with 1 or 0 whether the program is currently logged into the RoCon or not.

The bare command line is the blank-separated sequence of the nr1, nr2 and so on, which are numbers in either the usual decimal format or numbers starting with 0x which occur for example in the RAM-level of the FPGA registers. The first argument, nr1, will be 33 if programming the detector or ADC FPGA, and 49 if talking to some parts of the relay box. If any of these numbers in the arguments is negative (i.e., starts with a minus-sign), the option `--`, two repeated dashes, must be used to indicate that all followup command line arguments are arguments in the narrow sense, not options.

ccdlock

syntax: `ccdlock`

Adds a flag to a table of the tecs SQL data base on the local computer, indicating that the CCD is in use. This is a simple semaphore mechanism to prevent parallel use of the readout electronics at the same time by remote users and on-site users.

The `ccdlock` and `ccdunlock` pair is included in the `ccdread` program, such that the standard user does not need to know this command.

ccdread

syntax: `ccdread [-g {5|20|50|100}] [-b binning] [-e exptime] [-E exptime] [-f filename] [-F basename] [-u observername] [-o objectname] [-z genericfitsheader] [-w roistartend] [-r Nexp] [-Z] [-h]`

This is a python wrapper script to call the binary `ccdread3`.

- The option `-e` defines an exposure time for a SCIENCE image (shutter open time) in seconds. If the option is missing, zero seconds will be used.

- The option `-E` defines an exposure time for a DARK image (shutter remains closed) in seconds. If the option is missing, zero seconds will be used.

  The software refuses to start readouts if `-e` and `-E` are used at the same time.

- The option `-o` sets the object keyword in the FITS header if the shutter is open and the sun below the horizon. It is recommended to use IAU style names which start with 'J', have 6 digits for the right ascension (hours, minutes, seconds), the sign of the declination, and 6 digits for the declination, as taken from the telescope control panel. [This nomenclature makes only sense for SCIENCE exposures, open shutter open dome and at night time. We plan to apply Hormuth's pointing model to these names if they start with 'B' instead of 'J', and to add WCS parameters into the FITS files in the future. A name convention of this kind is desirable to allow re-calibrations of the pointing model in the future.] Example:

  ```
  ccdread -o B891031+025053
  ```

- The option `-u` adds a string of the observer's name to the FITS header. If not used, the Linux login name is inserted. As a precaution against splitting the commands in the Linux shell, one needs to add quotes around the arguments of options if these contain white space, here for example `ccdread -u 'R. J. Mathar'`.

- The option `-f` sets the name of the FITS output file. If that file exists, it will be overwritten. If the option `-f` is not used, a name containing a UTC timestamp will be used. If the option is not used, the `filename test.fits` will be used.

- The option `-w` produces images taken by sampling only a subwindow of the full detector region.

  - If this has the format with an `x` separating two positive numbers like `-w 400x400` it will produce a 400 times 400 pixels image centered in the middle of the chip.
  - If this has the format `-w x0:y0-x1:y1` it defines a rectangle of pixels by lower left and upper right pixels.
  - If this has the format `-w x0:y0+w+h` it defines a rectangle of pixels by lower left coordinates, width and height.

  All these coordinate origins, widths and heights need to be set as if the binning factor were 1; they refer to a window in the unbinned full 2k x 2k region. The coordinates start at x=1, y=1 for the pixel in the lower left corner (i.e., the FITS convention). If the `-w` is not used, the full area of the CCD plus some 30 columns of overscan space will be used, i.e., the upper x-value is slightly larger than 2048.

  Note that the time needed for the readout (ADC conversion) is approximately proportional to the area (product of width and height) of that window.

  Example:

  ```
  ccdread -o J891031+025053 -b 3 -w 300:300+400+700 -f test.fits
  ```

- The `-g` may be used to select one of the 4 supported gain factors. If the option is not used a default of 5 will be used.

- The `-b` followed by a small positive integer selects a binning factor along the horizontal and vertical direction. If the option is not used, a default of 2 will be used. Note that binning is implemented as a charge accumulation in the electronics before triggering a conversion with a Analog-to-Digital-Converter (ADC). So in practice the speed of the readout is proportional to the square of the binning factor (i.e., the number of pixels of the CCD that trigger one conversion). Obviously the resolution of the image is inversely proportional to the binning factor, and the average ADU value proportional to the square of the binning factor (with the thread of saturation). A standard binning of 2 makes sense because the resolution of the CCD is roughly half an arcsecond and the seeing is probably in the range of 1 to 2 arcseconds. Example

```
ccdread -b 3
```

For an exposure series that wants to extract widths of stars to determine a best focus, a binning factor of 1 is probably better.

- The option `-r` followed by a positive integer number repeats the exposure `Nexp` times, keeping the other parameters constant. The exposures are stored in a FITS cube in the output file. (Full-frame images with binning 2 contain 2 MBytes per read; the output files may become huge, and the file is generated at the very end of the last exposure.) The options `exptime` is the exposure time spent on each single image, even if `Nexp` is larger than one.
- The `-z` followed by a keyword-value expression adds additional keywords into the FITS file that will be generated. This means to insert the currently inserted filter and other information concerning the exposure that is not available to the data acquisition. The option may be repeated to add more than one line into the header. The expression has the format of the keyword, an equal sign, the value and optionally a sharp followed by a comment. To avoid premature syntactical breakup by the Linux shell, these expressions ought to be enclosed in single quotes. Examples:

```
ccdread -w 100:100+1000+1000 -z ’myint=700#yo’ -z ’myflo=3.14#no’ -z ’mytxt=yhoooo
```

It is strongly advised to add the `T_FOCUS` keyword for series of exposures intended to find a best M2 focal position, where the value is read of the control panel:

```
ccdread -b 1 -w 1900:1900+200+200 -z ’T_FOCUS=7.8#M2 position’
```

Otherwise one would have to use `fv` or a similar method to reintroduce the value into the FITS files before calling `fits2fwhm`.

- The option `-Z` uses tile-compression to reduce the size of the FITS file to typically a fourth of the standard size. If your FITS pipeline programs do not understand that file format, use `funpack` to uncompress the file.
- The option `-h` gives a short usage summary.

Use of options not in this list lead to error messages and do not start an exposure. Note that the underlying `ccdread3` always wipes the chip before the first read of the `Nexp` exposures. There is no command line switch to enable or disable this wiping at any particular time before or after the exposure.

If the program prints *No FITS files produced*, one of the following might have happened:

- You hit *CTRL-c* before at least one of the (possibly repeated) exposures finished. This triggered the exception handling of the subprocess `ccdread3` detailed in See [ccdread3], page 4.
- You tried to save data in a directory where you have no write permission. [Try to create a file there with something akin `touch tmptrash.txt` to see whether this is the cause.]

- The disk partition where the files would appear is full. Check this condition with `df -h`, and delete old files there, for example any redundant TIF-files.

ccdread3

syntax: `ccdread3 [-v] [-g {5|20|50|100}] [-b binning] [-e exptime] [-E exptime] [-f filename] [-u observername] [-o objectname] [-w roistartend] [-s] [-A nAdc] [-k filename] [-r Nexp] [-Z] [-D confdir]`

This is a C++ executable which gets an image by reading the CCD.

- The option `-v` enhances output information (but still in the limits defined by the log level options of the `Makefile.am` at compile time). The option also triggers that the data stream of all 16bit unsigned integers obtained through the fiber is written into a file with the same name as `filename` but the suffix `.fits` replaced by `.dat`. The typical invocation would be

  ```
  ccdread3 -v |& tee ccdread3.log
  ```

  ```
  less -R ccdread3.log
  ```

- The option `-g` should be one of the 4 integers and sets the the gain parameter in the electronics. If the option is not used, the value is read from the `ccdread.par` file in the configuration directory.
- The option `-b` sets a small positive integer binning factor. The same factor is used for the horizontal and vertical axis. If the `-b` is not used, a binning factor of 2 will be used.
- The option `-e` defines an exposure time for a *science* image (shutter open time) in seconds. If the option is missing, zero seconds will be used.
- The option `-E` defines an exposure time for a *dark* image (shutter remains closed) in seconds. If the option is missing, zero seconds will be used.

  If neither the `-e` nor the option `-E` is used, the shutter will remain closed.

- The option `-f` sets the name of the FITS output file. If that file exists, it will be overwritten. If the option `-f` is not used, a name containing a UTC timestamp will be used. For quick tests that produce no lasting data it is recommended to use `-f test.fits`. Note that producing the file will fail if the directory part of the file name points to a non-existing directory or to a directory without write permission for the login/Linux account.
- The option `-o` sets the object keyword and adds WCS coordinates in the FITS header if the shutter was open and the sun below the horizon. [This means for example that the `Analysis->Coordinate Grid` within `ds9` works.] It is recommended to use IAU style names which start with 'J', have 6 digits for the right ascension (hours, minutes, seconds), the sign of the declination, and 6 digits for the declination. The can be read from the telescope control panel. [This nomenclature makes only sense for SCIENCE exposures, open dome and open shutter. We plan to apply Hormuth's pointing model to these names if they start with 'B' instead of 'J', and to add WCS parameters into the FITS files in the future.]
- The option `-s` lets the computer wait for data on the CHAN1 of the OPTPCI, not CHAN0 (the default). This is a debugging option to check that the fiber pairs on the

ROE or the computer side have not been swapped: it allows to select the alternative fiber without walking to the computer and manually re-plugging the fiber heads.

- The option `-u` adds a string of the observer's name. to the FITS header. If not used, the Linux login name is inserted. As a precaution against splitting the commands in the Linux shell, one needs to add quotes around the arguments of options if these contain white space, here for example `ccdread -u 'R. J. Mathar'`.
- The option `-k` adds additional generic FITS lines to the FITS header which are read from the ASCII file that is the argument of the option. Each line should have the format of a keyword, an equal sign, a value and optionally a slash and a comment.
- The option `-w` produces images with a window into the full 2kx2k region. The argument has the format `[xfirst:xlast,yfirst:ylast]`, where the range of horizontal and vertical coordinates has the usual FITS offset of 1, meaning, the coordinate (1,1) is in the lower left edge of the image. Example: `-w [300:350,400:470]` produces an image which is 51 pixels wide and 71 pixels high (smaller if the binning factor is larger than one).
- The option `-r` followed by a positive integer number repeats the exposure `Nexp` times, keeping the other parameters constant. The exposures are stored in a FITS cube in the output file. The options `exptime` is the exposure time spent on each single image, even if `Nexp` is larger than one.
- The option `-A` followed by a number from 1 to 4 reads that many ADC in the pattern for transfer to the fibers. Because the patterns can only submit 32-bit words and an ADC value is a 16-bit value, the number of 16-bit values is actually rounded up to an even number by optionally flushing one ADC pixel datum twice into the fiber. If the option is not used, a value of 2 is assumed, which means the main pattern triggers conversion for all 4 ADCs, but only values of 2 of these are transferred to the computer (and only 1 of these was from the CCD, the other open-source ADC noise).
- The option `-Z` uses tile-compression to reduce the size of the FITS file to typically a fourth of the standard size. If your FITS programs do not understand that file format, use `funpack` to uncompress the file.
- The option `-D` specifies a directory which contains the `*.par` configuration files. The directory name must end on a slash. This is a developer's debugging option. If the option is not used, the default is `/opt/ccd/conf/`.

Other options on the command line will be ignored.

The program catches many interrupt signals (most importantly the terminate signal of users who hit *`CTRL-c`*) to clean up the data transfer and to close the shutter. If repeated exposures with the `-r` were scheduled, the FITS file will only contain the images that had already been transferred to the computer before the interrupt occured.

The program may create TIFF files according to the 6.0 specification (gray scale, 32bit IEEE floating point). Use a program like `display` that understands that file format to display the TIFF image. This is mainly for people who need a representation with full pixel resolution (which may not be available from screenshots) and/or do not have FITS viewers and/or want to avoid artefacts from other lossy data formats.

The name `ccdread3` exists for historical reasons. The integer 3 just means that (as of 2015) it was a third version of a program that was maintained at that time.

The difference between `ccdread3` is essentially that `ccdread` includes a safety guard in form of a `ccdlock`/`ccdunlock` pair.

ccdstart

syntax: `ccdstart [-v] [-p]`

Switch on relay box, set up readout electronics (ADC and patterns) with skeleton default parameters.

If the switch `-p` is used, a series of log files named `...log` is created which reprints information that was read from the parameter files.

If the switch `-v` is used, progress on what is done and from where information is created is more verbose. This would probably be used with redirection of the output to a file so one can examine it later:

```
ccdstart -v |& tee ccdstart.log
```

```
less -R ccdstart.log
```

ccdstop

syntax: `ccdstop`

Powers down all four DAC channels and switches off the relay box at the CCD.

If you want to power off the CCD electronics, always execute `ccdstop` first. Failure to execute `ccdstop` before somone powers off the CCD electronics may cause severe damage to the CCD chip, because it basically means to remove power from the various layers of the chip with unpredictable differential voltages or spikes. The importance of `ccdstop` is in removing power and disconnecting the chip in a well-defined sequence of steps.

A reset cycle would look like:

```
ccdstop -v
# here power off instrumentation, wait a few seconds power on
ccdstart
# a sequene readouts until the night is over
ccdread
ccdread
....
ccdstop
```

If you are not sure in which state the CCD is when working, you can do `ccdstop` followed by `ccdstart` and it should be fine. If the recent `ccdstop` was *not* followed by a `ccdstart`, all electronics of the ROE is actually disconnected from the detector, the ADCs produce grizzled noise patterns, and the FITS images look accordingly.

ccdtemp

syntax: `ccdtemp [-V] [-v] [-p] [-i tcp://xxx.yyy.zzz.ttt:PORT]`

Reads the temperature of the W&T 57710 module, which is the CCD temperature measured by a PT100 sensor, via the Ethernet. The number in units of Celsius is printed on standard output. Since November 2020, the W&T values are rescaled with the formula C->0.9213*C-3.058 better to match the environment temperatures, because the CCD temperatures appear to be uncalibrated and systematically too high.

- If the option `-v` is used, more verbose debugging output is printed.
- If the interface to the module in the file `/opt/ccd/conf/ccdtemp.par` is not correct, it can be manually modified here with the `-i`. Example:

  ```
  ccdtemp -v -i tcp://149.217.42.221:80
  ```

  If neither the option `-i` is used nor the file `/opt/ccd/conf/ccdtemp.par` is readable, the default of the interface becomes `tcp://wut70cm.mpia.de:80`.
- If the option `-p` is also used, the value is printed into two files (of the current working directory):
  - The temperature in Celsius into the file `wtemp.dat` as a bare number (without line feed). Any previous value is overwritten.
  - The bare value in Celsius, a blank and a time stamp (GMT, indicated by a +0000 offset in the string) is appended to the file `wtempout.dat`, followed by a line feed.
- If the option `-V` is used, a table of temperatures of the past is printed: In each line a UTC time stamp, the CCD temperature and the dome temperature in Celsius.

ccdtemp.plt

syntax: `ccdtemp.plt` [-w] [-t yyyy-mm-dd:yyyy-mm-dd]

Plots the ambient and CCD temperatures over time for some section of the past.

If the option `-t` is used with 2 day specifications of the format 4-digit-year, 2-digit month and 2-digit day, separated by dashes and a colon as indicated, the time slot is restricted to the specified interval of the two days. If the option is not used, approximately one week of the most recently recorded data is displayed.

If the option `-w` is not used, the plot is closed with the standard window close button in the upper right corner. One may zoom into a portion of the graph by dragging with the right mouse cursor across a rectangular section of the graph.

If the option `-w` is used, the plot is uploaded to the anonymous FTP server `ftp://ftp.mpia.de/pub/ccd/` as `king70cmTemp.pdf` and `king70cmTemp.asc`, and no interactive manipulation is available. This option is in an experimental manner currently used five times per hour in a `crontab` of the ccd account, and the file also accessible from a link in the `https://svn.mpia.de/trac/gulli/king` web page.

ccdunlock

syntax: `ccdlock`

Erases a flag to a table of the tecs SQL data base on the local computer, indicating that the CCD is no longer in use. This releases the semaphore that has been created by `ccdlock`.

This program may for example be needed if an ongoing exposure was interrupted by a power failure, so the `ccdread` did not actually call `ccdunlock` while finishing.

dfits

syntax: `dfits [-x xtnum] file.fits [file.fits ...]`

Prints the headers of the FITS files of the command line. The default is to print only the primary header; the option `-x` can be used with a positive integer to print the header of a FITS extension or with `-x 0` to print the headers of all data-header units. Example

```
dfits -x 0 *.fits
```

ds9loop

syntax: `ds9loop [ds9option ...] directory [directory ...]`

Calls the viewer `ds9` in a loop over FITS files in directories. All command line arguments that start with a dash are interpreted as `ds9` command line options, and all other arguments are interpreted as directories. A directory specified as a dot means the current working directory.

This is meant as a simple interface to look at all FITS files in directories without selecting file names one by one.

The user needs to close the `ds9` GUI to get to the next FITS file in the director(ies).

Examples

```
ds9loop .
ds9loop /data1/Panic
ds9loop -mosaicimage /data1/Panic
ds9loop -multiframe .
```

fits2csv

syntax: `fits2csv`

syntax: `fits2csv [-c cfgfile|-k kwlist] [-o outfile.csv] [-d dbgLevel] dir1 [dir2 ...]`

The program extracts values of a set of header keywords from all FITS files in specified directories, and generates a text file in tabular comma-separated-values (CSV) format. That file could then be opened for example with open-office (selecting the comma as the field separator).

The program opens an interactive GUI if called without command line options. In the GUI one should first select directories that contain the FITS files of interest, then edit the table of the header keywords of interest, then finally the name of the output file, and start the data extraction with a button at the bottom.

If called with command line options, the options `-c` or `-k` must be present and be followed by an argument. This calls the program as a batch program which is working in the background, so the `outfile.csv` still changes after the Linux command line prompt has reappeared.

All arguments not related to the two options are interpreted as names of existing directories that contain FITS files to be scanned.

- The option `-c` followed by the name of an existing configuration file reads the keywords from the configuration file. Each line in the configuration file specifies one FITS header keyword as a regular expression, followed by white space and an non-negative integer number. The number counts HDU's, so if it is zero the keyword should be searched in the primary HDU, if it is 1 in the second HDU and so on. Lines in the configuration file that have the sharp (#) as the first non-white character are skipped. Lines that do not have an integer after the first blank are also skipped.
- The option `-k` followed by a comma-separated list of keywords specifies regular expressions for the keywords to be extracted from the primary header units. So the major differences to the `-c` option is that header lines in other than the primary HDU will not be scanned, and that there is no need to create an intermediate configuration file. Note that the comma-separated list of keywords must usually be enclosed by single quotes on Linux/Unix systems because the shell would otherwise try to interpret the commas and stars in the regular expressions by its own syntax rules.

  The options `-c` and `-k` can be used at the same time. Then the list of keywords is the unordered concatenation of the two lists of keywords of the individual options. (Duplicates are not removed.)
- The option `-o` followed by the name of an output file tells the program to add the comma-separated values to that file. If the option is not used, the program writes to standard output.
- The `-d` followed by an integer (0 or 1) implies a debug level. If not given, 0 is the default and minimizes verbosity.

Example:

```
cd /dataB/readout2
fits2csv -c $HOME/fits2csv.cfg -o nightLog.csv 20151211 20151212

where $HOME/fits2csv.cfg contains

RA 0
DEC 0
EXPTIME 0
HIERA.*ITIME 0
```

Example:

```
fits2csv -c $(geirs_build)/test/fits2csv.carmenes -o nightLog.csv -d 1 201[56]-*
```

To get a list of FITS files of the current working directory where the XBINNING value is 4, one may put

```
XBINNING 0
```

into `$HOME/fits2csv.cfg` and filter the output with `awk` such that only lines with a third field equal to 4 pass:

```
fits2csv -c $HOME/fits2csv.cfg . | awk -F',' '{if ($3 == 4) print $2}'
```

kingPoint

syntax: `kingPoint [-v] [-h] [-H relhum] [-t ambiCelsi] [-T YYYY-MM-DDTHH:MM:SS] [-p hPa] [-l lambdanm] [-f cfgFile] {J|B}HHMMSSsDDMMSS`

syntax: `kingPoint [-v] [-h] -B {J|B}HHMMSSsDDMMSS`

Applies the pointing model corrections to a ra/dec designation.

The mandatory command line argument is a sky coordinate in the IAU style, a capital letter, 6 digits of the right ascension without the colons (2 digits hour, 2 digits minutes, 2 digits seconds), the sign of the declination (plus or minus), and the 6 digits of the declination (2 digits degrees, 2 digits arcminutes, 2 digits arcseconds). Each 2-digit substring must be padded with leading zeros if any of these 6 numbers is smaller than 10. If the leading capital letter is a `J`, the program runs through the pointing model and produces a corrected position. If the leading letter is anything else, the program again computes a corrected designation, but then flips the correction by moving the output along a great circle to the other side by the same angular distance. Although not precisely correct, this is is within one or two arcseconds the same as reverting the pointing model (i.e., as converting from the nominal coordinates to J2000). So the leading character triggers conversion from control desk to catalog positions or vice versa.

- The option -h just prints a short usage line.
- The option -H is followed by a relative humidity in the range of 0 to 100. If not used 50 percent are used.
- The option -t is followed by an ambient temperature in degrees of Celsius. If not used a default of 10 is used.
- The option -T followed by a ISO specification of a UTC of observation (1 or 2 hours off from the local German time zone depending on daylight savings), four-digit year, 2-digit month where January is 01, 2-digit day in the range 01 to 31, a capital T, and hours, minutes and seconds separated by colons. If the option is not used the current time is used by default.
- The option -p followed by a number is the ambient pressure in units of hPa. If not used a default of 948 is used.

  The options `-H`, `-t` and `-p` are causing slight adjustments in the refractive index of air to model the deviation between observed and astrometric zenith angles. The corrections are unimportant in practice. One might set the pressure explicitly to zero to investigate the current direction of the parallactic angle.
- The option -l followed by a number defines the observing wavelength (a sort of effective mean of the passband of the filter) in nanometers. If not used a default of 550 is used.

- If the option -v is used, the program prints some derived parameters (azimuth, elevation, local sidereal time, hour angle) and derived ra and dec coordinates to standard output. Note that this uses Hormuth's telescope position if the `-B` is absent, so times like the local sidereal time and hour angle may be off by a few seconds from the actual values of the dome.
- If the option `-f` is used, the option argument must be the name of a readable file providing the parameters c[0..4] and d[0..5] of the pointing model. If the option is not used, the parameters of the 2005 calibration are used. The file may contain lines that start with sharps (#) or empty lines, both of which are ignored. The format of the line that contains the fit parameters must have at least the following 12 white-space separated fields: (i) a name of a telescope model, which is ignored, for compatibility with the early IDL program, (ii) 5 floating point numbers for the c-parameters, (iii) 6 floating point numbers for the d-parameters. Any other fields in the line beyond these are ignored.
- If the option -B is used, the program converts FK4 (B1950) coordinates to FK5 (J2000) coordinates or vice versa, assuming zero proper motions and zero parallax. So this is not an application of a telescope specific pointing model but an aid of matching the two stellar coordinate systems. Which of these two directions is chosen depends on the final argument: If this starts with `B` a conversion from FK4 is done, and if it starts with `J` a conversion from FK5 is done. The coordinate system of the output coordinates is indicated by starting either with a `J` or with a `B`.

Warning: this is a blind application of the pointing model mentioned in Hormuth's 2006 thesis, translated from IDL to C++.

One may often get a humidity from `https://kurpfaelzer-gleitschirmflieger.de/WETTER/start.html` or `https://www.lsw.uni-heidelberg.de/wetter/`, but the pressure ought to be the one at the 560 m altitude of the telescope, not one reduced to sea level.

Example:

```
kingPoint J100000+600010
kingPoint B100000+600010
kingPoint -H 10 -t 5 -p 700 -l 800 -T 2006-03-01T00:13:13 J100000+600010
kingPoint -B J100000+600010
kingPoint -B -v J100000+600010
```

rmOldFiles.sh

syntax: `rmOldFiles.sh directory percentage [fnamepattern] [..fnamepattern]`

Removes the oldest files in some directory and subdirectories on the disk to free disk space.

The first argument of the shell script is the directory in which files are located which are candidates for removal. The candidate files include this directory and recursively all its subdirectories.

The second argument of the script is an integer number which is a percentage of maximum used disk space to achieve by removal of files. The script looks with the `df` command at the filling level of the disk that holds the directory, and iteratively removes the oldest candidate files until the usage falls below that percentage.

The optional one to five trailing arguments are file name patterns applied to filter the names of files that are candidates for removal. If no such argument is added, the filter `*.fits` is used, which essentially means that FITS files are removed, but no others. To prevent expansion of wildcard patterns by the current shell, these file name patterns must be enclosed by pairs of single quotes.

There is no guarantee that after the program finishes the disk usage has actually fallen below the targeted percentage, because there may be not enough (or not large enough) files in the candidate list to let the disk use drop as intended.

This example deletes files with the suffix `.bak` in the directory `/disk-d/data` until at least 10 percent of the disk are unused:

```
rmOldFiles.sh /disk-d/data 90 '*.bak'
```

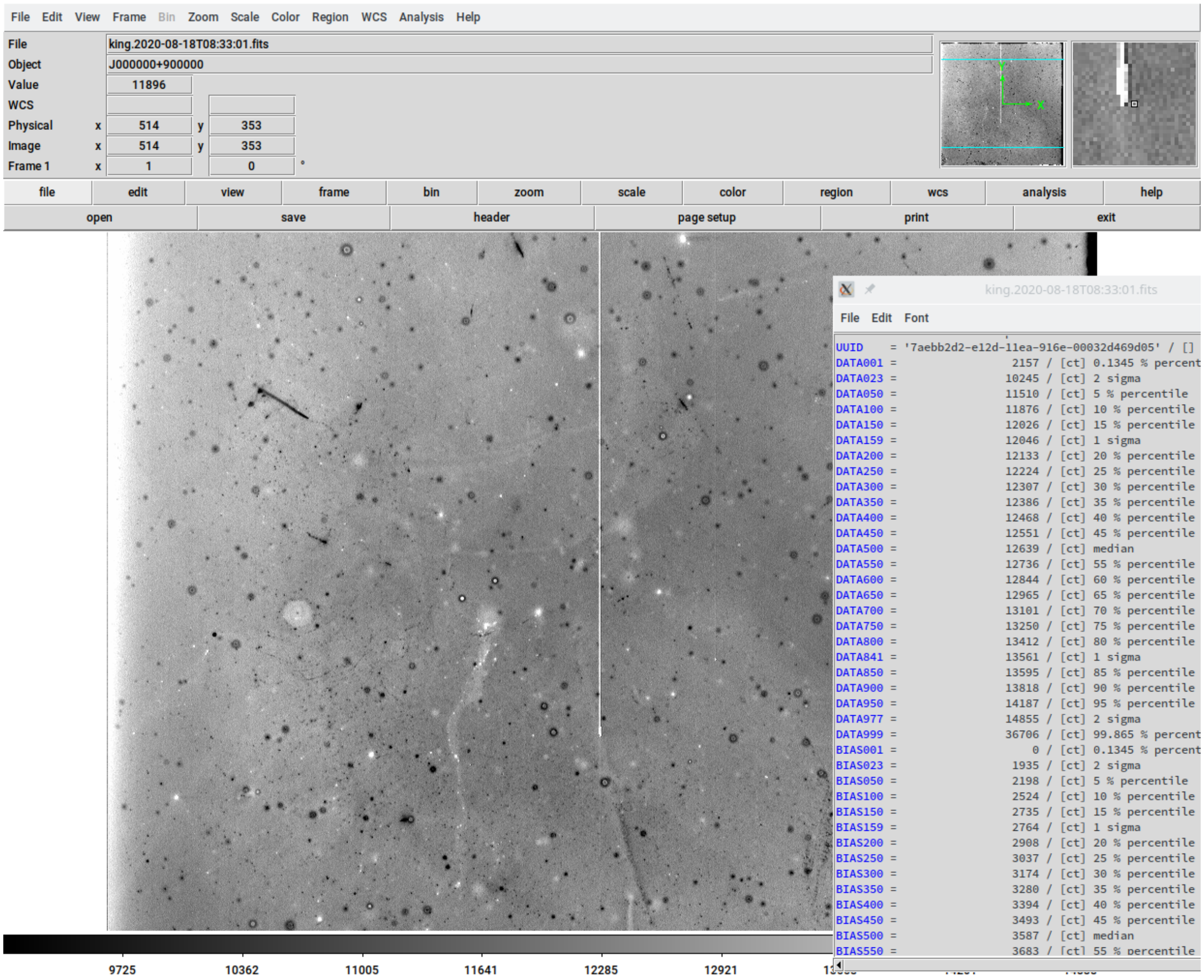
This example deletes FITS files in the directory `/disk-d/data` until at most 85 percent of the disk are used:

```
rmOldFiles.sh /disk-d/data 85
```

It is entirely possible to erase entire file systems indefinitely by using wild card patterns like `*.*`*. The files are not moved into any sort of trash bins for later removal, but in fact removed!*

Index

(Index is nonexistent)


File Edit View Frame Bin Zoom Scale Color Region WCS Analysis Help
file edit view frame bin zoom scale color region wcs analysis help
open save header page setup print exit
9725
10362
11005
11641
12285
12921
File Edit Font

File king.2020-08-18T08:33:01.fits
Object J000000+900000
Value 11896
WCS
Physical x 514 y 353
Image x 514 y 353
Frame 1 x 1 0 °

king.2020-08-18T08:33:01.fits

```
UUID    = '7aebb2d2-e12d-11ea-916e-00032d469d05' / []
DATA001 =                 2157 / [ct] 0.1345 % percent
DATA023 =                10245 / [ct] 2 sigma
DATA050 =                11510 / [ct] 5 % percentile
DATA100 =                11876 / [ct] 10 % percentile
DATA150 =                12026 / [ct] 15 % percentile
DATA159 =                12046 / [ct] 1 sigma
DATA200 =                12133 / [ct] 20 % percentile
DATA250 =                12224 / [ct] 25 % percentile
DATA300 =                12307 / [ct] 30 % percentile
DATA350 =                12386 / [ct] 35 % percentile
DATA400 =                12468 / [ct] 40 % percentile
DATA450 =                12551 / [ct] 45 % percentile
DATA500 =                12639 / [ct] median
DATA550 =                12736 / [ct] 55 % percentile
DATA600 =                12844 / [ct] 60 % percentile
DATA650 =                12965 / [ct] 65 % percentile
DATA700 =                13101 / [ct] 70 % percentile
DATA750 =                13250 / [ct] 75 % percentile
DATA800 =                13412 / [ct] 80 % percentile
DATA841 =                13561 / [ct] 1 sigma
DATA850 =                13595 / [ct] 85 % percentile
DATA900 =                13818 / [ct] 90 % percentile
DATA950 =                14187 / [ct] 95 % percentile
DATA977 =                14855 / [ct] 2 sigma
DATA999 =                36706 / [ct] 99.865 % percent
BIAS001 =                    0 / [ct] 0.1345 % percent
BIAS023 =                 1935 / [ct] 2 sigma
BIAS050 =                 2198 / [ct] 5 % percentile
BIAS100 =                 2524 / [ct] 10 % percentile
BIAS150 =                 2735 / [ct] 15 % percentile
BIAS159 =                 2764 / [ct] 1 sigma
BIAS200 =                 2908 / [ct] 20 % percentile
BIAS250 =                 3037 / [ct] 25 % percentile
BIAS300 =                 3174 / [ct] 30 % percentile
BIAS350 =                 3280 / [ct] 35 % percentile
BIAS400 =                 3394 / [ct] 40 % percentile
BIAS450 =                 3493 / [ct] 45 % percentile
BIAS500 =                 3587 / [ct] median
BIAS550 =                 3683 / [ct] 55 % percentile
```

Figure 1: Reference image of the warm CCD with closed shutter and zero seconds integration time.

Calling

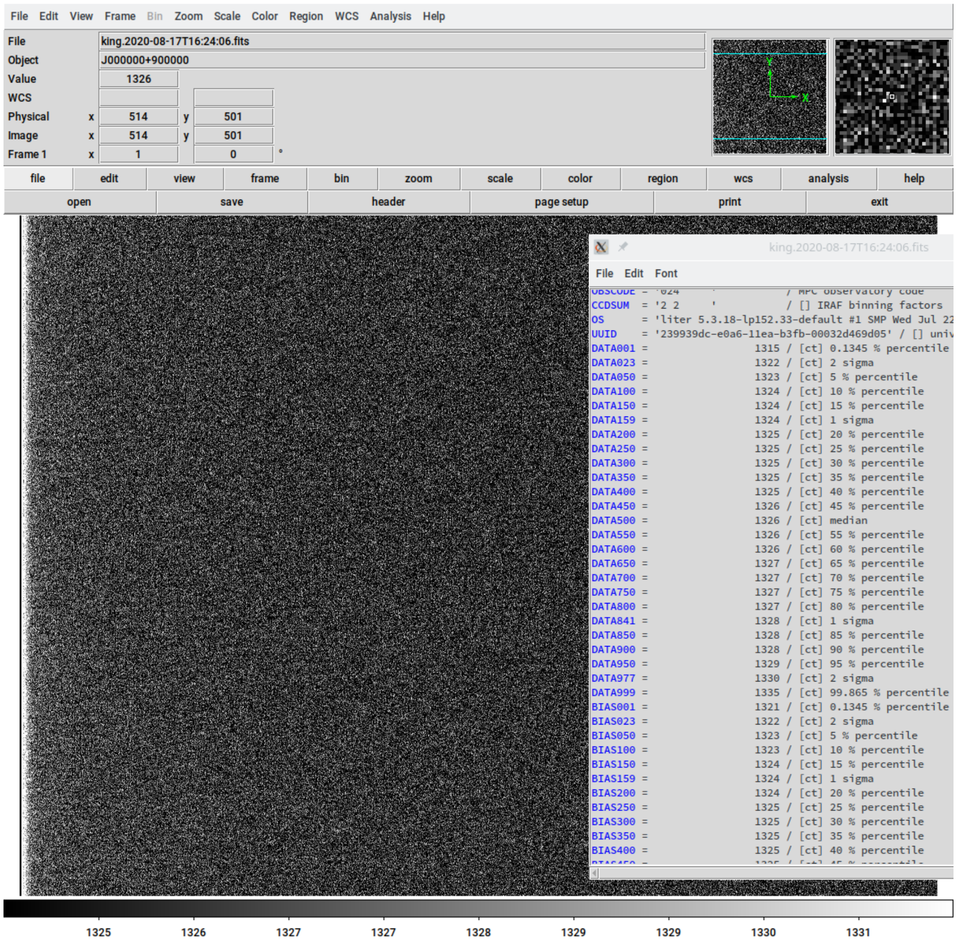
```
ccdread
```

should create a FITS file looking roughly as in Figure 1. If the CCD is cold, the output has a much lower mean level as in Figure 2. If the ADC's have been disconnected, a sort of open-collector noise as in Figure 3 results.

2.3 Prerequisites

2.3.1 ccdread

Readout of the CCD requires that the CCD eletronic boxes are powered, the fiber pair of the CCD is connected to the OPTPCI board at the rear side of the computer, that the computer is powered and that a `ccdstart` was executed after the most recent repowering of the CCD electronics and after the most recent `ccdstop`.

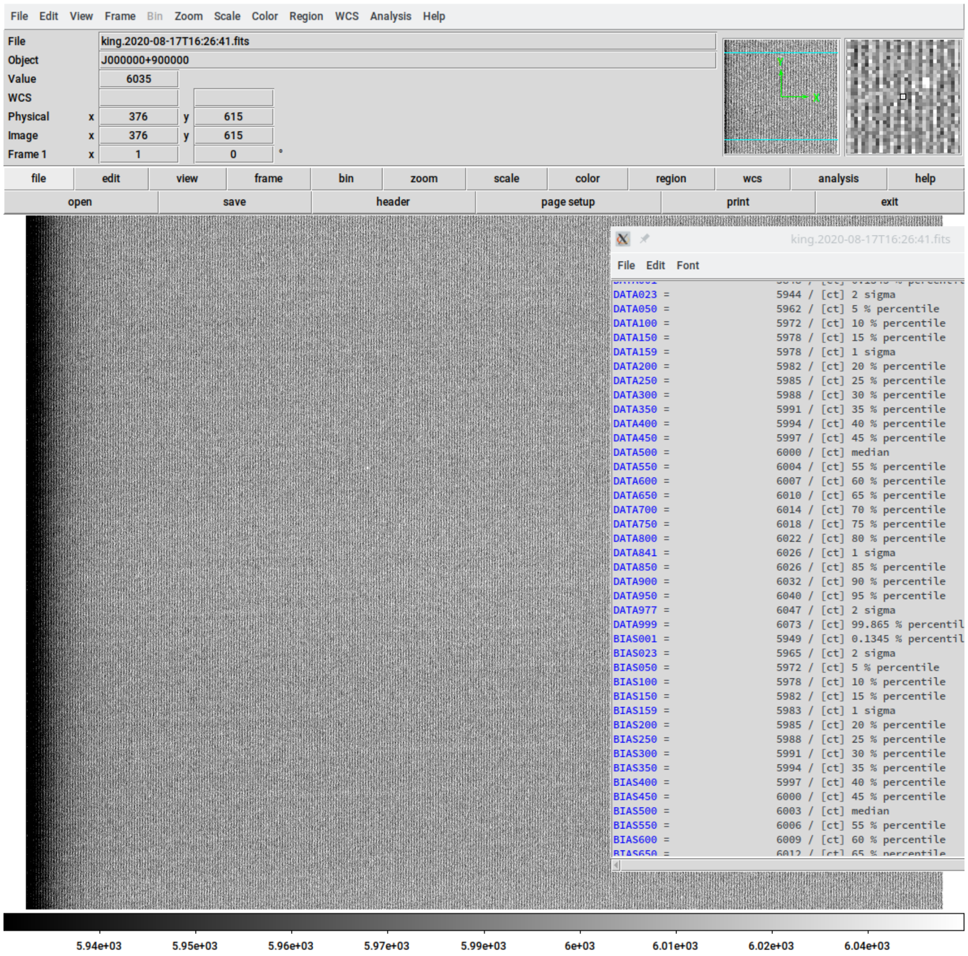

File Edit View Frame Bin Zoom Scale Color Region WCS Analysis Help
file edit view frame bin zoom scale color region wcs analysis help
open save header page setup print exit
File Edit Font
1325 1326 1327 1327 1328 1329 1329 1330 1331

File king.2020-08-17T16:24:06.fits
Object J000000+900000
Value 1326
WCS
Physical x 514 y 501
Image x 514 y 501
Frame 1 x 1 0

king.2020-08-17T16:24:06.fits

```
OBSCODE = '024     '           / MPC observatory code
CCDSUM  = '2 2     '           / [] IRAF binning factors
OS      = 'liter 5.3.18-lp152.33-default #1 SMP Wed Jul 22
UUID    = '239939dc-e0a6-11ea-b3fb-00032d469d05' / [] univ
DATA001 =                 1315 / [ct] 0.1345 % percentile
DATA023 =                 1322 / [ct] 2 sigma
DATA050 =                 1323 / [ct] 5 % percentile
DATA100 =                 1324 / [ct] 10 % percentile
DATA150 =                 1324 / [ct] 15 % percentile
DATA159 =                 1324 / [ct] 1 sigma
DATA200 =                 1325 / [ct] 20 % percentile
DATA250 =                 1325 / [ct] 25 % percentile
DATA300 =                 1325 / [ct] 30 % percentile
DATA350 =                 1325 / [ct] 35 % percentile
DATA400 =                 1325 / [ct] 40 % percentile
DATA450 =                 1326 / [ct] 45 % percentile
DATA500 =                 1326 / [ct] median
DATA550 =                 1326 / [ct] 55 % percentile
DATA600 =                 1326 / [ct] 60 % percentile
DATA650 =                 1327 / [ct] 65 % percentile
DATA700 =                 1327 / [ct] 70 % percentile
DATA750 =                 1327 / [ct] 75 % percentile
DATA800 =                 1327 / [ct] 80 % percentile
DATA841 =                 1328 / [ct] 1 sigma
DATA850 =                 1328 / [ct] 85 % percentile
DATA900 =                 1328 / [ct] 90 % percentile
DATA950 =                 1329 / [ct] 95 % percentile
DATA977 =                 1330 / [ct] 2 sigma
DATA999 =                 1335 / [ct] 99.865 % percentile
BIAS001 =                 1321 / [ct] 0.1345 % percentile
BIAS023 =                 1322 / [ct] 2 sigma
BIAS050 =                 1323 / [ct] 5 % percentile
BIAS100 =                 1323 / [ct] 10 % percentile
BIAS150 =                 1324 / [ct] 15 % percentile
BIAS159 =                 1324 / [ct] 1 sigma
BIAS200 =                 1324 / [ct] 20 % percentile
BIAS250 =                 1325 / [ct] 25 % percentile
BIAS300 =                 1325 / [ct] 30 % percentile
BIAS350 =                 1325 / [ct] 35 % percentile
BIAS400 =                 1325 / [ct] 40 % percentile
BIAS450 =                 1325 / [ct] 45 % percentile
```

Figure 2: Reference image of the CCD at −82° C with closed shutter and zero seconds integration time.


File Edit View Frame Bin Zoom Scale Color Region WCS Analysis Help
file edit view frame bin zoom scale color region wcs analysis help
open save header page setup print exit
File Edit Font
5.94e+03
5.95e+03
5.96e+03
5.97e+03
5.99e+03
6e+03
6.01e+03
6.02e+03
6.04e+03

```
File        king.2020-08-17T16:26:41.fits
Object      J000000+900000
Value       6035
WCS
Physical  x 376   y 615
Image     x 376   y 615
Frame 1   x 1       0 °
```

king.2020-08-17T16:26:41.fits

```
DATA023 =                 5944 / [ct] 2 sigma
DATA050 =                 5962 / [ct] 5 % percentile
DATA100 =                 5972 / [ct] 10 % percentile
DATA150 =                 5978 / [ct] 15 % percentile
DATA159 =                 5978 / [ct] 1 sigma
DATA200 =                 5982 / [ct] 20 % percentile
DATA250 =                 5985 / [ct] 25 % percentile
DATA300 =                 5988 / [ct] 30 % percentile
DATA350 =                 5991 / [ct] 35 % percentile
DATA400 =                 5994 / [ct] 40 % percentile
DATA450 =                 5997 / [ct] 45 % percentile
DATA500 =                 6000 / [ct] median
DATA550 =                 6004 / [ct] 55 % percentile
DATA600 =                 6007 / [ct] 60 % percentile
DATA650 =                 6010 / [ct] 65 % percentile
DATA700 =                 6014 / [ct] 70 % percentile
DATA750 =                 6018 / [ct] 75 % percentile
DATA800 =                 6022 / [ct] 80 % percentile
DATA841 =                 6026 / [ct] 1 sigma
DATA850 =                 6026 / [ct] 85 % percentile
DATA900 =                 6032 / [ct] 90 % percentile
DATA950 =                 6040 / [ct] 95 % percentile
DATA977 =                 6047 / [ct] 2 sigma
DATA999 =                 6073 / [ct] 99.865 % percentil
BIAS001 =                 5949 / [ct] 0.1345 % percentil
BIAS023 =                 5965 / [ct] 2 sigma
BIAS050 =                 5972 / [ct] 5 % percentile
BIAS100 =                 5978 / [ct] 10 % percentile
BIAS150 =                 5982 / [ct] 15 % percentile
BIAS159 =                 5983 / [ct] 1 sigma
BIAS200 =                 5985 / [ct] 20 % percentile
BIAS250 =                 5988 / [ct] 25 % percentile
BIAS300 =                 5991 / [ct] 30 % percentile
BIAS350 =                 5994 / [ct] 35 % percentile
BIAS400 =                 5997 / [ct] 40 % percentile
BIAS450 =                 6000 / [ct] 45 % percentile
BIAS500 =                 6003 / [ct] median
BIAS550 =                 6006 / [ct] 55 % percentile
BIAS600 =                 6009 / [ct] 60 % percentile
BIAS650 =                 6012 / [ct] 65 % percentile
```

Figure 3: Reference image of the CCD at −82° C with closed shutter and zero seconds integration time, after `ccdStop`. The message of this figure is: if the CCD is cold and the background level is near 6000 counts and a very regular shower of vertical stripes appears, the image is not reading the CCD because a previous `ccdStart` was missing.

Note that there is no requirement that the entire control electronics of the telescope and its drives are switched on to read the CCD. Only the `Instrumentierung` switch on the control desk in the dome must be enabled/green; the others and the coordinate and LST information may all be unlit/off.

2.3.2 Temperature log

For a periodic log of the two temperatures again (i) the CCD electronic boxes need to be powered by switching `on` the `Instrumentierung` on the control desk, and (ii) the computer that stores the ambient and CCD temperatures must be on. (Hibernating the computer is equivalent to switching it off here.) For the CCD temperature the WuT block must be connected to the Ethernet (MPIA science net), and for the ambient temperature the LabJack must be reachable from the computer via the local switch. Network accessiblities can be checked with

```
ping wut70cm # CCD temperature device
ping teles  # LabJack temperature near control desk
ping rocon  # readout electronics for ccdstart/read/stop
```

on the logging computer's Linux shell. Figure 7 shows how this accesses the WuT via the MPIA network and the LabJack UE9 via the NETGEAR switch.

2.4 FITS output

The current software produces a single FITS file with 16-bit data in the primary HDU. If a repetition factor > 1 was used with `ccdread` or `ccdread3`, the consecutive images are placed in a FITS cube.

Some keywords detailing the World Coordinate System are included if the shutter was open and the sun below the horizon. Apart from a small rotation by 2.6°, the image is not flipped, and North is up and East is left in the sky images.

According to the FITS standards, the `DATE` is the time stamp when the file was written, and `DATE-OBS` is when the exposure started. The interval in between comprises the exposure time(s) plus the time to convert the pixels with the ADCs and to transmit them to the computer.

The `EXPTIME` is the exposure time of a single image, even in the cases with multiple exposures in a FITS cube.

The `DATE-`i keywords are the start-of-frame times (what would be times of opening the shutter) if the option `-o` of `ccdread/ccdread3` was used to generate more than one frame. These are time stamps of the i-th slice into the data cube. `DATE-1` should be the same as `DATE-OBS`.

A histogram of the pixel values in the region of the detector (columns and rows from 128 to 1920 in the unbinned pixel scale) is put in the `DATA`iii keywords and an equivalent histogram of the overscan region (right from column 2048, unbinned) is put in the `BIAS`iii keywords. The three digits iii are percentiles (multiplied by 10) of the histogram: `DATA500` is the median of the data section, `DATA159` is the ADU value below which 15.9% of the data remain, and `841` is the ADU value below which 84.1% of the dta remain. Subtraction of `DATA841` and `DATA159` values gives the 1σ-spread in the data, for example. For data files with more than one exposure (cubes) these are the statistics of all the images in the file.

The keywords OS is the name and operating system of the computer on which the data acquisition was run.

CCDTEMP and AMBITEMP are the two temperatures of Section 2.5 at the start of the exposure, in Celsius.

DETSEC is the unbinned size of the image contained in the 2048 × 2048 region, i.e., excluding any overscan regions. If the windowing options of ccdread/ccdread3 were used, it will be the unbinned rectangle of this subwindow within the 2048 × 2048 region.

DATASEC is the size of the image contained in the 2048 × 2048 region, shrunk by the binning factor, and translated so the lower left corner is at $[1, 1]$.

BIASSEC is the size of the image outside the 2048 × 2048 region (actually at horizontal values > 2048 in the overscan area), shrunk by the binning factor, and translated so the lower left corner is at $[1, 1]$. Warning to ds9 users: to view the overscan section beyond the x-value of 2048 in the main window, you need to remove the flag Use DATASEC in the Scale menue.

The three OBSGEO coordinates are the geodetic coordinates of the East Elsässer Dome at the Königstuhl [1, 2].

LST is the local sidereal time measured in seconds in the range 0–86400. It is derived from DATE-OBS and OBSGEO-L. The comment rephrases this in the HH:MM:SS format.

UUID is a type-1 Universal Unique Identifier and rephrases the DATE key.

PIXSCALE is the mean pixel scale equivalent to the 15 μm size of the actual pixels; it does *not* change with the binning.

CREATOR is the software the created the FITS file; currently this is a identifier into the SVN repository of the MPIA software engineers in the technical department.

Many other keywords related to the telescope, the F-ratio, diameter, observatory code and so on are fixed.

Note that the software produces the standard checksums; it becomes easy to figure out which naïve, FITS-unaware software packages manipulated the file afterwards without recognizing the standards.

ccdread3 adopts an ESO standard to use time stamps with colons in the FITS file names. There is an obstacle while copying these files with scp to other computers, because scp may interpret these files as hosts and users. To remove that problem use full path names for the files when copying, in the style of `scp ./myfile:hugo.fits user@host:remote` by adding dot-slash in front of the names for example.

The FTOOLS of the HEASARC have been compiled on the computer such that programs like ftcopy to select subimages or slices, like chimgtyp to change image types, like ftpixcalc to add or subtract images, and fv to modify FITS header lines are available.

2.5 Temperatures

There is a sensor element connected with the dewar accessible by a web module of W&T shown in Figure 7. The simplest way to get the temperature is to use any type of web browser to access port 80:

```
firefox wut70cm.mpia-hd.mpg.de:80
```

In standard operation, which means if cooled with liquid nitrogen to 77 K, that could theoretically be close to −196, but in reality it will typically stay above −120.

Within `ccdread3` `ccdtemp` is called which uses an equivalent `GET` on the `WWW` server to retrieve this temperature and to insert it into the FITS header.

An ambient temperature measured at some point in the control desk of the dome is caught through a LabJack UE9 sensor. The UE9 temperature may be polled with a `python` program like

```
#!/usr/bin/env python

import time
import ue9

# Open Labjack UE9
lj=ue9.UE9(ethernet=True,ipAddress="192.168.3.200")

# Read once every 10 seconds until interrupted
while True:
  data_tdome1 = lj.getTemperature()-273.15
  print(data_tdome1)
  time.sleep(10)
```

and that is what (essentially) the script `ambiTemp.py` does.

The mechanisms build into the `ccdread3` insert these temperatures into the FITS headers, keywords `CCDTEMP` and `AMBITEMP`, so the standard observer would not need these tools.

Apparently the two raw temperatures differ by an offset of 5 degrees; this is likely a result of inacurate calibrations or of a warm-up of the dewar by continuous operation of the electronics. Since 2020-11, the output of the WuT reading has been modified by a linear fit recalibrated with the LabJack tempertaure, see `ccdtemp` in Section 2.2.

Plots of the temperatures as a function of time are in a link of `https://svn.mpia.de/trac/gulli/king/wiki/WikiStart/Observing#no2` . Many so-called modern browsers disallow to look into these `ftp` directories. If that is the case, one needs to download the PDF file to look at the plot:

```
wget --no-passive-ftp ftp://ftp.mpia.de/pub/ccd/king70cmTemp.pdf
evince king70*.pdf &
```

If you are already logged into the `king` computer, one can also look at the plot of the local directory with the browser:

```
firefox file:///opt/ccd/share/king70cmTemp.pdf &
```

or with the PDF viewer:

```
okular /opt/ccd/share/king70cmTemp.pdf &
```

Temperatures are absent for all time intervals when (i) the scripts were inhibited because the `king` computer was shut down or sent into hibernation or (ii) when the control electronics was switched off so the WuT electronics (at least) is unreachable.

A weather station at the Landessternwarte with wind and humidity data is `https://www.lsw.`

uni-heidelberg.de/wetter/. A weather station near the Königstuhl Bergbahn is https://holfuy.com/de/weather/1503.

2.6 PyRAF

The call

```
cd
mkiraf
```

is needed to generate the `login.cl` once for each user account that intends to use IRAF or PyRAF. The command `mkiraf` is available *after* the PyRAF intallation (see below).

2.7 Jupyter

The `jypter notebook` must be called explicitly from the Linux shell as

```
jupyter --port=8080 notebook
```

This is currently based on python 3.10. If there are problems of loading dedicated modules of some sort, you may try to go back to python 3.8, i.e. call

```
jupyter-notebook-3.8 --port=8080
```

`ipython` can be called right from the Linux shell:

```
ipython
```

2.8 Pointing Model

There is a note in the Hormuth thesis [3] that a pointing model has been calibrated. Model parameters of Buie's model [4, 5] are apparently left in the `/opt/kappa70` files on `liter`. By reverse enineering of the IDL code and data files, this model may be summarized as follows: The position of the telescope was assumed to be at longitude λ =+08:43:15° and latitude φ=+49:23:55° and altitude 500 m. (Note that this differes from the recommended values of Chapter 3.5.15. Hormuth's position is approximately 400 meters north-west off the dome.)

The major open question is what the reference frame of the pointing coordinates at the control desk of the telescope is. One may speculate that this is a B1950 frame, so given the inauguration of the telescope in 1986, we have at least a precession of 0.47° (by the standard rule of thumb of 360 degrees in 26,000 years, i.e., 1700 arcsec, 3000 unbinned pixels, $1\frac{1}{2}$ times the detector size) in the 34 years since then. It is not clear which portion of the pointing model is not a model of the gear-ratios, encoders and polar axis of the telescope but catching an Earth axis motion of the past decades. The

- online form http://www.robertmartinayers.org/tools/coordinates.html
- https://heasarc.gsfc.nasa.gov/cgi-bin/Tools/convcoord/convcoord.pl
- routine `kingPoint` of section 2.2 from the computers Linux shell with the option `-B`

may be used to convert B1950 ↔ J2000.

2.8.1 Refractive index

The assumption of the refractive index of humid air $n(\lambda, P, t, r)$ was [6]

$$\hat{n} \equiv 64.328 + \frac{29498.1}{146 - \frac{1}{\lambda^2}} + \frac{255.4}{41 - \frac{1}{\lambda^2}}, \tag{1}$$

where λ is the wave length of light in microns.

$$p \equiv P \frac{1 + (1.049 - 0.0157t)P \times 10^{-6}}{720.883(1 + 0.003661t)}, \tag{2}$$

where P is the atmospheric pressure in mm Hg and t the atmospheric temperature in °C.

$$\hat{t} \equiv 100 - t; \quad l \equiv 2.8808 - \frac{5.67\hat{t}}{274.1 + t - 0.15\hat{t}}; \quad f \equiv \frac{r}{100} \times 10^l, \tag{3}$$

where r is the relative humidity in percent.

$$w \equiv (0.0624 - \frac{0.000680}{\lambda^2}) \frac{f}{1 + 0.003661t}, \tag{4}$$

$$\bar{n} \equiv (\hat{n} - w)p; \quad n = 1 + 10^{-6}\bar{n}. \tag{5}$$

The astrometric zenith angle z^v (the angle assuming no atmosphere, vacuum) is then assumed to be related to the observed zenith angle z^o by $\sin z^o = \sin z^v / n$.

2.8.2 Model Fit

The fitted model parameters are

c_0	c_1	c_2	c_3	c_4	d_0
1129.57126679	92.38913925	17.71283515	50.60759643	-10.75546509	-1162.07858759

d_1	d_2	d_3	d_4	d_5	$\hat{\alpha}$	$\hat{\delta}$
238.63414620	-45.31772836	23.96596299	1.12467965	0.14276478	17.5	-157.0

where c_i and d_i are in arcseconds.[1]

The model uses azimuths a, hour angle h, declination δ, elevation a and its complemnt, the zenith angle $z = \pi/2 - a$. Auxiliary parameters are

$$x' = \cos h; \quad y' = \sin h; \quad z' = \sin\varphi \cos\delta - x' \cos\varphi \sin\delta; \tag{6}$$

$$\hat{\delta} = c_0 - c_1 x' - c_2 y' - c_3 z' + c_4 \times 0. \tag{7}$$

Then the fixed declination is $\delta^f = \delta - \frac{\hat{\delta}}{3600}$ in degrees. Further auxiliary parameters are

$$s' = \frac{1}{\cos\delta}; \quad t' = \tan\delta; \quad q' = \sin\varphi \tan\delta + \cos\varphi \cos h; \quad h' = -c_1 y' t' + c_2 x' t' + c_3 s' y' \cos\varphi; \tag{8}$$

$$\hat{h} = d_0 + d_1 s' + d_2 t' + d_3 q' + d_4 a + d_5 \{h + 180°\}, \tag{9}$$

[1] These parameters are copied from the configuration files in the `kappa70` directory, where the telescope has been described as `KINGW1, telescope west`. This may be misleading; the telescope has always been in the east dome of the laboratory. . .

where the curly parentheses $\{\ldots\}$ mean the argument is wrapped into the range from 0 to 360°. The fixed hour angle then is

$$h^f = h - \frac{\hat{h} + h'}{3600}. \tag{10}$$

The application of the pointing model consists of the following steps:

1. Start from a J2000/FK5 equatorial α, δ coordinate. Apply proper motion or parallax correction to forward this to the epoch (although with the current pointing errors and plate scale considering proper motions is fruitless in practise);
2. rotate the two coordinates from J2000 through Earth precession/nutation to the epoch;
3. project these two (vacuum) coordinates to the local (horizontal) coordinate system, apply refraction correction to the elevation/zenith and revert the transformation to obtain "observed" equatorial coordinates;
4. use these observed equatorial coordinates as inputs to the pointing model and compute "fixed" output coordinates.

2.9 Dome Slit Model

Summary: In principle the dome slit position would be aligned with the azimuth of the telescope pointing, but since the German Mount ("asymmetrical equatorial mount") of the telescope moves the telescope axis almost always away from the dome center, we summarize some geometric parameters that should allow to realign the dome rotation with the current telescope pointing.

The local topocentric horizontal coordinate system may defined by a right-handed orthogonal Cartesian system of axes, where $+x$ points to north, $+y$ to the west, and $+z$ to the zenith. The right ascension axis of the mount points to the North Celestial Pole, which is

$$\vec{N} = \begin{pmatrix} \sin\theta_p \cos\phi_p \\ \sin\theta_p \sin\phi_p \\ \cos\theta_p \end{pmatrix} \tag{11}$$

in that coordinate system, where for our rough purposes $\phi_p \approx 0$, $\theta_p = \pi/2 - \varphi$ and $\varphi \approx 49.39545°$ is the geodetic latitude of the telescope. The declination axis is orthogonal to this axis and turns around the r.a. axis by a position angle α_p at the characteristic diurnal rate of 15 arcsec per second. We define the origin of α_p to be zero when the dec. axis is horizontal westwards along $(0, 1, 0)$—which is the standard parking position at the zenith. The dec. axis moves in a plane given by the normalized cross product of $\vec{N} \times (0, 1, 0)$, so the unit vector from the intersection of the two axes (the "intersection" for short) along the dec. axis to the telescope is

$$\vec{T} = \cos\alpha_p \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \sin\alpha_p \frac{1}{\sqrt{\cos^2\theta_p + \sin^2\theta_p \cos^2\phi_p}} \begin{pmatrix} -\cos\theta_p \\ 0 \\ \sin\theta_p \cos\phi_p \end{pmatrix}. \tag{12}$$

The vector from the intersection to the pointing axis of the telescope is $d\vec{T}$, where $d \approx 0.900$ m is taken from a drawing in the mid 1970's. The pointing direction of the telescope is

$$\vec{P} = \begin{pmatrix} \sin z \cos A \\ -\sin z \sin A \\ \cos z \end{pmatrix}, \tag{13}$$

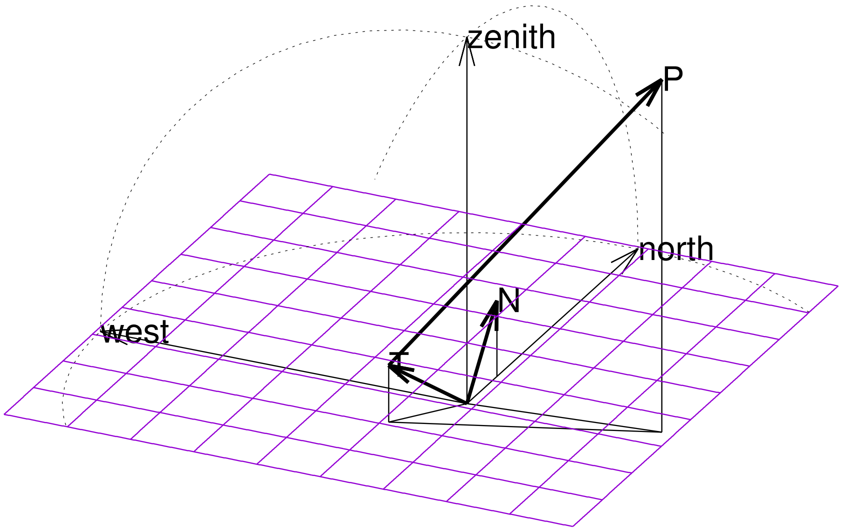


Figure 4: Off-axis telescope-slit model: right ascension axis $\vec{N}$, declination axis $\vec{T}$, and telescope pointing direction $\vec{P}$ in a local horizontal coordinate system. Azimuth is $A = 70°$ and zenith distance $z = 40°$, but the dome slit needs to be $\phi_D \approx 83°$, $\theta_D \approx 28°$. $\alpha_p \approx 61°$.

where A is the azimuth (in the convention north=0, east=90°), and z the zenith distance in the range from 0 to 90°.

The encoder angle α_p is fixed by requiring that $\vec{T}$ and $\vec{P}$ are orthogonal, the dot product of the previous two equations to be zero:

$$\tan\alpha_p \frac{1}{\sqrt{\cos^2\theta_p + \sin^2\theta_p\cos^2\phi_p}}[\cos\theta_p \tan z\cos A - \sin\theta_p\cos\phi_p] + \tan z\sin A = 0. \tag{14}$$

As mentioned above $\cos\phi_p \approx 1$, neglecting corrections within the pointing model, so

$$\tan\alpha_p = \frac{\tan z\sin A}{\sin\theta_p - \cos\theta_p\tan z\cos A}. \tag{15}$$

There is the usual 180° offset ambiguity of the result α_p (equivalent to swapping the counter weight and telescope position on the dec. axis). This can probably be resolved by requiring that the z-component of $\vec{T}$ remains positive, $\alpha_p > 0$, because the telescope should stay “above” the r.a. axis. In practise the unit vector $\vec{T}$ is not computed by plugging this angle into Eq. (12) but directly by the normalized cross product

$$\vec{T} = \frac{1}{\cos\delta}[\vec{P}\times\vec{N}]. \tag{16}$$

We assume that the center of the dome is at the intersection, and that the dome is spherical with a radius $R \approx 3.7$ m—measured by a folding yard stick. So the telescop axis hits the dome at

$$d\vec{T} + t\vec{P} = R\begin{pmatrix}\sin\theta_D\cos\phi_D\\ -\sin\theta_D\sin\phi_D\\ \cos\theta_D\end{pmatrix}, \tag{17}$$

such that the dome should be open into the direction ϕ_D, where 0 is north and 90° is east. This vector equation is three equations with the three unknowns t, θ_D and ϕ_D. Squaring the previous equation and using that $\vec{T} \perp \vec{P}$ gives $d^2 + t^2 = R^2$, so

$$\frac{d}{R}\vec{T} + \sqrt{1-(d/R)^2}\vec{P} = \begin{pmatrix} \sin\theta_D \cos\phi_D \\ -\sin\theta_D \sin\phi_D \\ \cos\theta_D \end{pmatrix}. \tag{18}$$

$\tan\phi_D$ is then fixed by the ratio of the y- and x-coordinates of the vectors (with no sign ambiguity because $\sin\theta_D > 0$ and separate signs for $\cos\phi_D$ and $\sin\phi_D$ are known). As a check, for $d/R \to 0$ we get $\phi_D \to A$ and $\theta_D \to z$, as expected with these sign conventions.

A python function which implements this is:

```
#/usr/bin/env python3

import math
import numpy

def domeAz(z, A, phip=0.0):
    ’’’
    input z is star zenith distance in radians
    input A is star azimuth in radians, N=0, E=pi/2
    input phip is the actual azimuth of the telescope RA axis. Zero for all practical work.
    output is two angles [phi,theta] in radians of dome opening. First angle
    is azimuth, second angle the polar angle where 0 is the dome top

    >>> z = math.radians(17.0)
    >>> A = math.radians(33.0)
    >>> slit = domeAz(z,A)
    >>> print("dome azimuth",math.degrees(slit[0]),"polar angle",math.degrees(slit[1]))
    ’’’
    # MPIA Elsaesser lab geographic latitude, degrees
    phiKing = 49.39545
    # equivalent local zenith distance of North Celestial Pole (NCP)
    thetap = math.pi/2.0 - math.radians(phiKing)
    # star direction in topocentric Cartesian system
    P = [math.sin(z)*math.cos(A), -math.sin(z)*math.sin(A), math.cos(z)]
    # NCP in local Cartesian system related from spherical angles
    N = [math.sin(thetap)*math.cos(phip), math.sin(thetap)*math.sin(phip), math.cos(thetap)]

    T = numpy.cross(P,N)
    len = numpy.linalg.norm(T)
    # direction vector of telescope declination axis, normalized to unit length
    T = T/len

    # distance ra axis to telescope axis (=length of dec axis), meters
    d = 0.900
    # dome radius, meters, estimate
    R = 3.7
    # ratio axis distance to dome radius, unitless
    dR = d/R
    # vector where telescope axis hits dome
    dom = numpy.add(numpy.multiply(dR,T),numpy.multiply(math.sqrt(1-dR**2),P))
    # split vector on dome hull into polar angle theta and azimuth phi
    thetaD = math.acos(dom[2])
    phiD = math.atan2(-dom[1],dom[0])
    return [phiD, thetaD]
```

Since there are no plans to upgrade the dome mechanis with a computerized control system, this is just an academic exercise for now.

2.10 Astronomy

2.10.1 Seeing and Focus

The fastest way to fit a star blob in a FITS image to a Gaussian is apparently by opening it in `siril`: Call `siril` with a command line argument which is your FITS file, and drag with the left mouse button a small square box around one of the stars, and it shows in the `fwhm` field at the lower left a measure in units of pixels.

2.10.2 Astrometry Package

The astrometry.net package is installed on the computer, see https://manpages.debian.org/testing/astrometry.net/solve-field.1.en.html for the main use case and the `astron2hdr` command in Section 2.2.

2.11 Remote Desktops

2.11.1 ssh

The secure shell daemon has been enabled, so the usual remote access is via

```
ssh -X staff@king.mpia-hd.mpg.de
```

or

```
ssh -X praktikum@king.mpia-hd.mpg.de
```

Alternatively call

```
vinagre
```

select `ssh` as the protocol, `praktikum` or `staff` as the user, `king` as the computer and click on `connect`.

2.11.2 vnc

For remote access for heavy-load GUI applications, the `vnc` server and clients are available. The typical use on the remote workstation is to build a tunnel to `king`, to start the VNC server there, and then to connect to the associated port of the server from the remote computer through the tunnel: [Note that the display number N reported by the server is port $5900 + N$ on the remove machine]

```
remote> ssh -X praktikum@king
king> vncpasswd # remember the password! needed for vncviewer below
king> vncserver -autokill # example response: king:2
remote> ssh -N -L 5922:localhost:5902 praktikum@king # take 5922 because smalle probably already in
remote> # ssh -N will appear to hang, keep the window open and open another one
remote> vncviewer localhost:22 # click on cancel for "authentication required.."
```

Logout from the remote session as usual by clicking with the right-most button in a free part of the window manager of the remote screen, select `leave` and click on `OK` (but do *not* shutdown the computer). Terminate the `ssh -N...` with `CTRL-C`.

2.11.3 rdp

To connect remotely from a machine where a `remmina` client has been installed, call

```
remmina
```

select the Protocol `RDP` and type in `king`, edit the connection profile (second icon on the upper left) for user `praktikum` or `staff`, insert the password, increase the `remmina` screen to approximately the size it should have during the session[2] or set the resolution to `Use client resolution` and press `Connect`. Clicking with the right mouse button in the remote session screen offers a logout submenue to terminate the session.

2.11.4 zoom

To support zoom sessions, the software was installed by downloading the RPM package for openSUSE from `https://zoom.us/download?os=linux`, and running

```
zypper install zoom_openSUSE_x86_64.rpm
```

as `root`. See Figure 9 for the microphone and speaker lines.[3]

3 MAINTAINER

3.1 Cables

A schematics of the Ethernet and fiber cabling of the workstation for the telescope is in Figure 7.

[2] apparently there is no option to resize windows during the session maintaining the resolution if `Use initial window size` is selected

[3] Apparently there is no build-in mic or speaker; users need to bring their own head phones...

The NETGEAR switch is separating the 192.x.x.x subnet from the outer world. It ensures that there is no interference with other 192.168.3.x devices that are switched on in the MPIA network.

The main computer is `king.mpia.de`, a PicoSYS 2614 with Core i7-7700T CPU, 16 GB RAM, 256 GB SSD (ASTRO4162910000, SN 1908063047) with openSUSE 15.3. It looks like in Figures 8–9.

After connecting the power supply at the rear, the small round silver button at the front is pressed to boot the computer. A weak beeb tone will be heard. (No monitor, keyboard or mouse is needed to boot the computer). At the rear side of the computer, two red Light Emitting Diodes (LEDs) of the OPTPCI card are on, and two green LEDs will start to blink.

The recommended way to shut down the computer is to run

```
poweroff
```

in the Linux shell as `root`. The LEDs at the rear side of the computer will turn off. Optionally remove all cables (power, USB, monitor, Ethernet).

3.2 SVN

The software is backed up by a Subversion (SVN) installation on the `svn.mpia.de` computer. So commands like

```
cd ~ccd/king
svn info --depth infinity
svn log                                              # read comments of commits
svnversion                                           # print SVN version number of cwd
svn checkout -r21 https://svn.mpia.de/gulli/king r21 # check out revision 21
```

ought to work for the `ccd` user.

The file `~ccd/.subversion/config` specifies `global-ignores` such that in particular FITS files and files with the suffixes `.dat` and `.txt` are not added to the repository.

3.3 PLX (Pex8311) Driver

3.3.1 Automated Loading

To load the driver each time the computer is (re)booted `plxload*.service` is copied to `/etc/systemd/system/`, then

```
systemctl enable plxload8311
systemctl status plxload8311
```

These steps are not needed and actually fail if no PLX device (read: no OPTPCI board) is found on the local bus system. Caveat: if this is automatism is not added, each invocation of `ccdread` or any of the tests involving the OPTPCI board needs to call either the wrapper script

```
plxstartup
```

or

```
/sbin/service plxload8311 restart
```

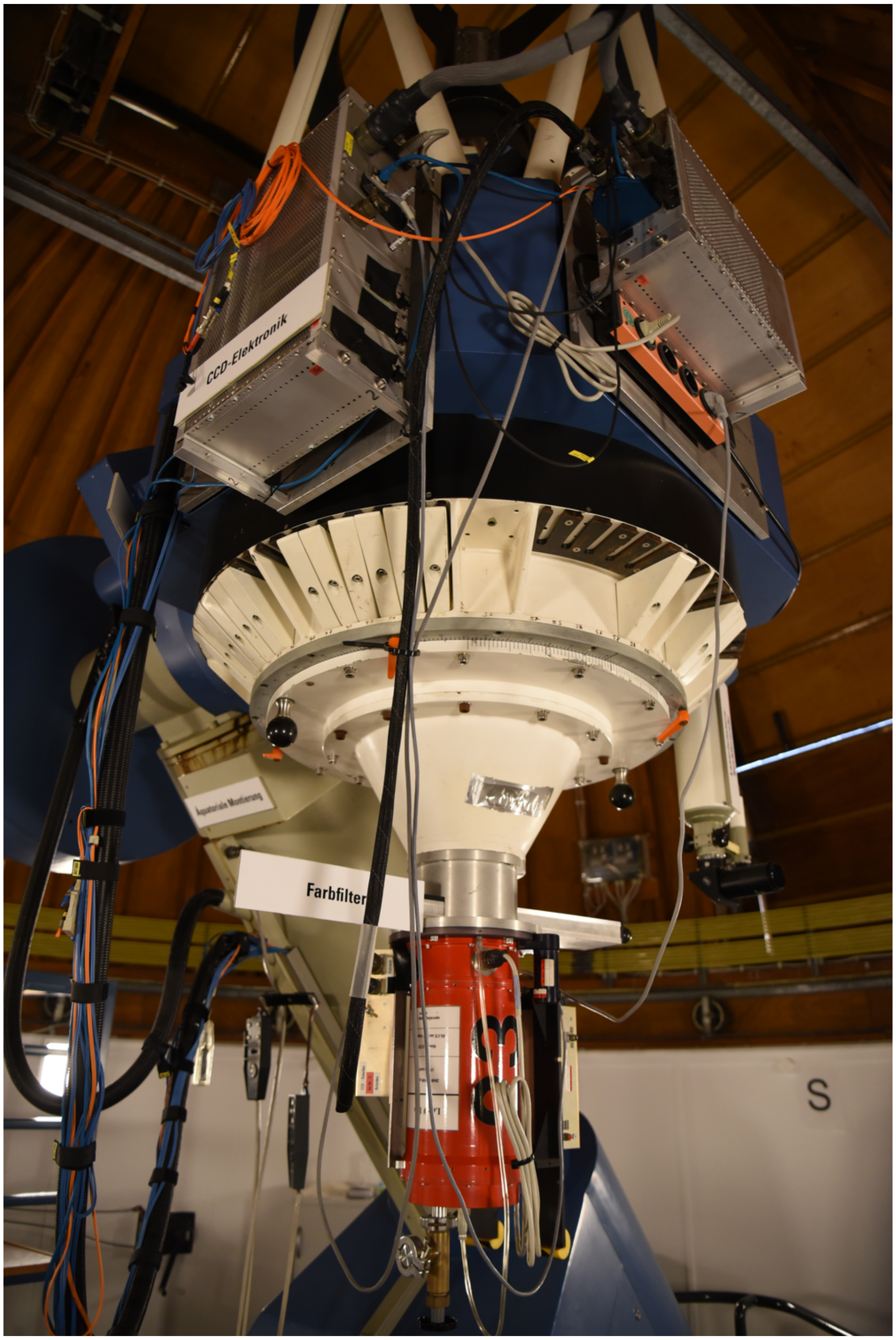


Figure 5: Overview of the relevant electronics. The red tube at the bottom is the CCD dewar. The box to its left is the relay box, the box to its right the temperature control. Mounted near the primary mirror (outside the blew ring) are the grey racks of the CCD electronics (left) and the ROE (right). Underneath the ROE is the external power supply.

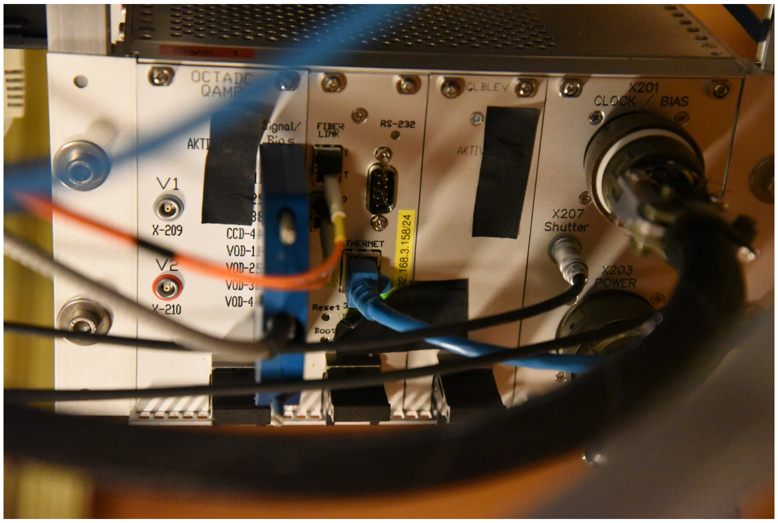


Figure 6: Front panel of the ROE. The fiber pair of the orange cables ends at the computer as pictured in Fig. 8. The blue ethernet cable ends at the NETGEAR switch of the local network.

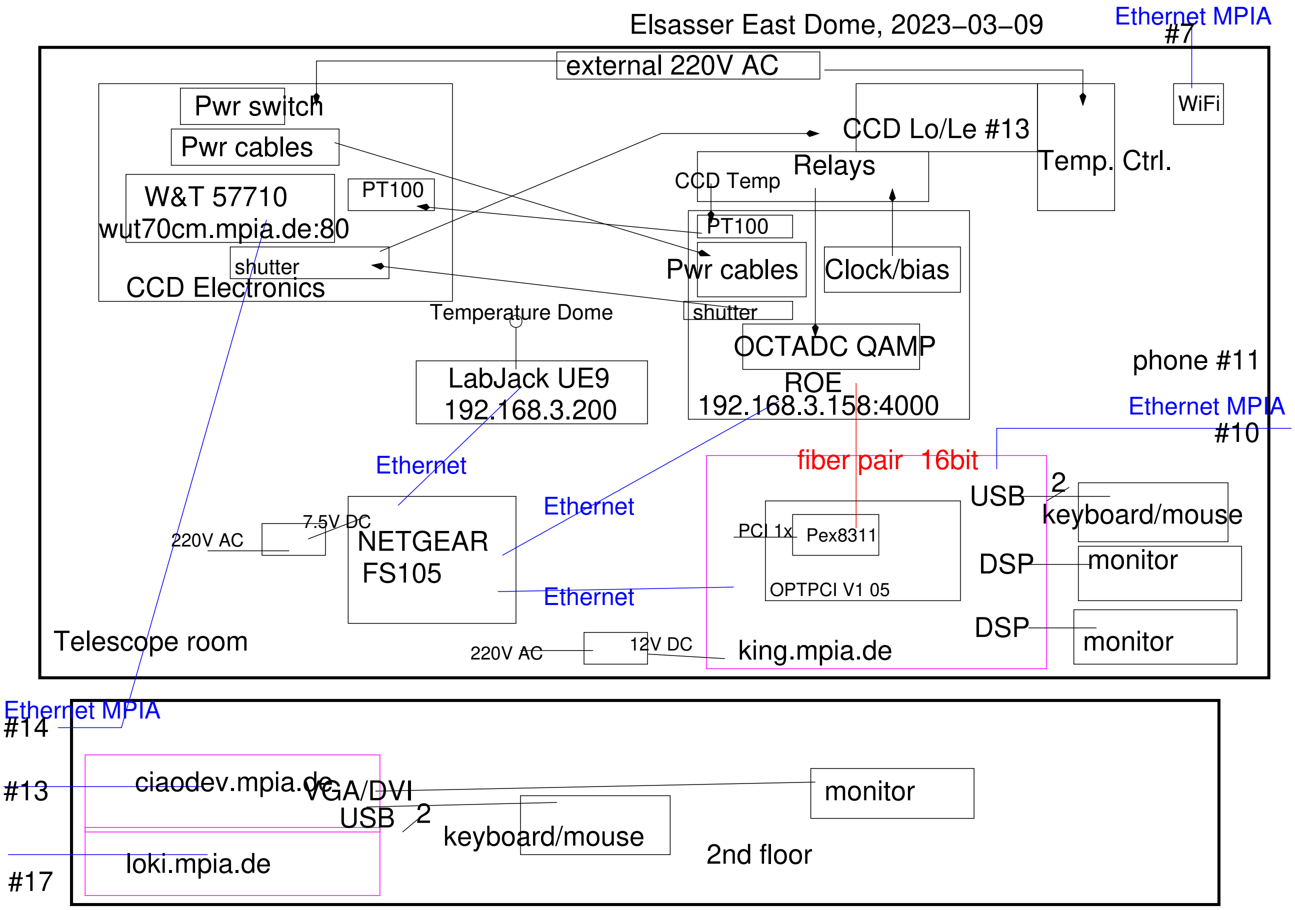


Figure 7: Cabling of equipment in the East Elsässer Telescope Dome. The phone number in the telescope room is ☎364. Computers are shown in magenta and the Ethernet connections in blue.

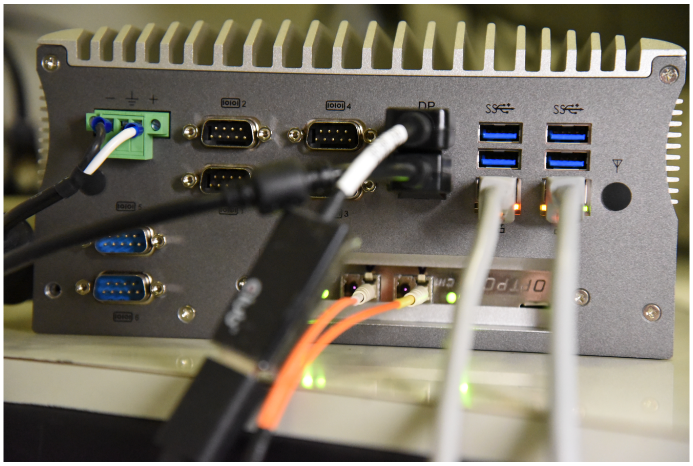


Figure 8: Rear side cabling of the PicoSys 2614 computer. The left of the two grey RJ45 Ethernet cables is connected to the wall (MPIA Internet), the right is connected to the switch.
One of the two orange fibers from the ROE is plugged into CHAN1, the other into CHAN0 of the OPTPCI. The sending connectors of CHAN0 and CHAN1 are not used. To get nonzero data, the fiber head with the yellow cap is to be inserted into CH1, the fiber with the white cap into CH0 (with the setup on the ROE in 2021-02).

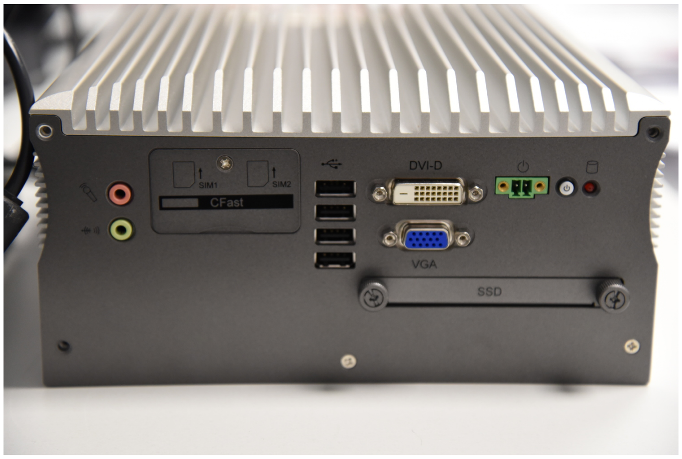


Figure 9: Front of the PicoSys 2614 computer. The current setup is that the two monitors are connected to the display ports (DP) on the rear side, and mouse and keyboard with two of the four USB ports at the front side.

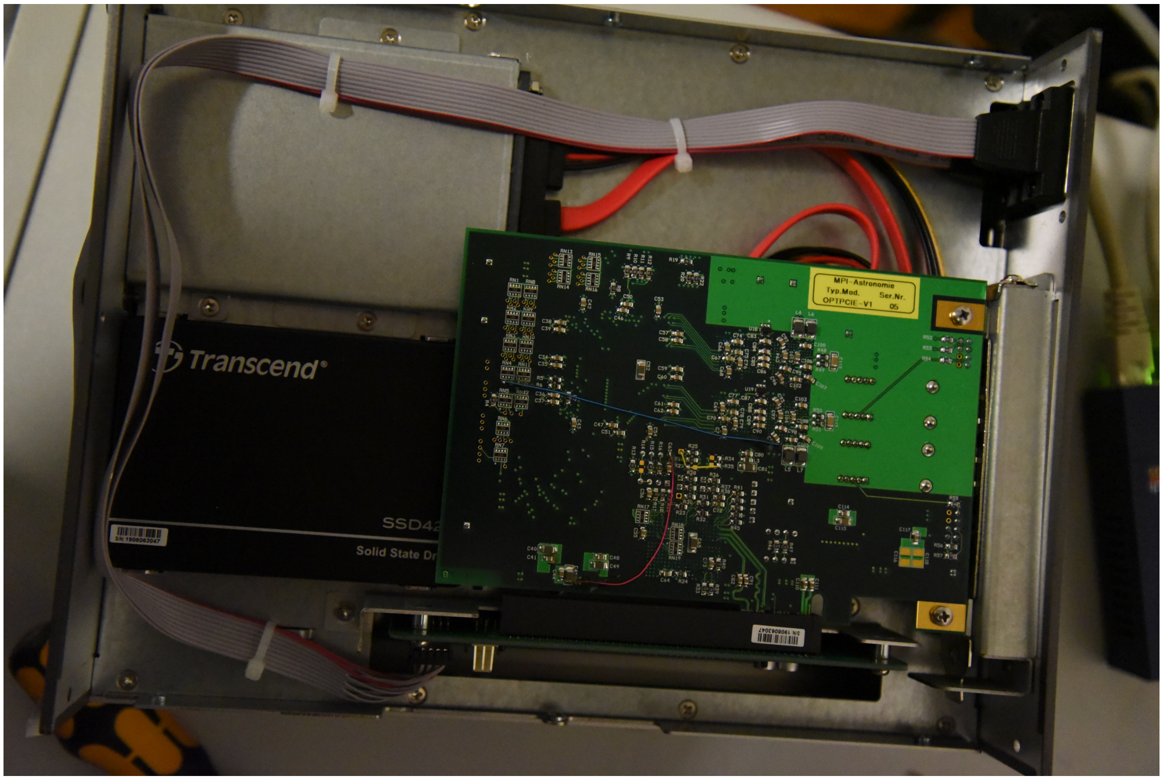


Figure 10: OPTPCI-e board plugged into the PicoSys 2614 computer.

at least once (which needs `root` privileges).

A simple check of successful loading of the driver is that

```
lsmod | fgrep -i Plx
```

contains the `Plx8311` entry and that

```
/sbin/service --status-all | fgrep -i plx
```

contains a line which mentions `loaded active (openSUSE) or loaded (CentOS) or [ + ] plx (Ubuntu)`.

Call

```
/sbin/lspci -v | grep -E 'Plx(8311|9656)' # CentOS openSUSE
lspci -v | grep -E 'Plx(8311|9656)' # Ubuntu
```

so see which boards are plugged into the computer.

If you have root permissions,

```
cat /proc/vmallocinfo | fgrep Plx
```

should show three lines for each OPTPCI board plugged into the computer. Starting `yast2`, moving into the `Security center and hardening` menu, selecting the `Configure` of `Enable basic system services` should also indicate the Plx drivers enabled. If `lsmod` does not show the driver, scan the system logs:

```
journalctl | fgrep plxload
```

Note that the `chkconfig` activates driver loading at computer run level changes; you won't see the driver in the services until the next reboot or a manual interaction via `plxstartup, plxload8311` as above.

3.3.2 Compilation

If

```
zypper up
```

installs a new Linux kernel in `/usr/src`, the PLX driver of the `OPTPCI` board must be recompiled (as `root`) and the CCD software be re-linked with the recompiled driver:

```
cd king/sw/ccd/Rocon_dev/Cfiles/srdata
cp Broad*.zip /usr/src # only needed for SDK > 8.23
./INSTALL.plx
```

To load the new driver, the computer should either be rebooted or the driver be reloaded.

Each time the driver is recompiled, the ROE software must be recompiled as in Section 3.6, because it is linked with the binaries in the `/usr/src` directory. If `zypper up` has installed a new Linux kernel, the steps are (as root)

```
chkconfig --del plxload8311 # remove (temporarily) to avoid boot propblems with old driver
reboot now                  # ensure that new kernel is active
INSTALL.plx                 # recompile PLX driver
chkconfig --add plxload8311  # add new driver to forthcoming reboots
```

3.4 Temperature logs

The two relevant temperatures are logged by the syslog of the operating system and the MySQL server of the operating system by calling `temp2jrnl` in a cronjob.

```
journalctl
```

should show at least a few log lines. If this responds with `...due to insufficient permissions`, check that the directory `/var/log/journal` exists. If not, switch to persistent (and per-user) journaling. by changing

```
#Storage=auto
```

to

```
Storage=persistent
```

in `/etc/systemd/journald.conf` and reboot or restart the journal with

```
systemctl restart systemd-journald
```

as superuser.

A suitable `crontab(1)` line for the `crontab -e` entry is

```
0-57/3 * * * * PATH=$PATH:/opt/ccd/bin ; temp2jrnl 1>/dev/null 2>&1
```

so each 3 minutes the temperatures are logged (20 per hour, 480 per day, 85 thousand per year). Note that calls like `journalctl --vacuum-time=...` or switching to another computer may delete data of the journal; the data in the SQL are not affected then.

A call like

```
journalctl --utc -o short-iso -q SYSLOG_IDENTIFIER=king70 -n 3360
```

would extract the temperatures of the past seven days from the system logger. The entries contain in the payload just two bare numbers, the CCD temperature and the dome temperature in Celsius. The logs stored in the SQL data base are printed with

```
mysql -u tecs -pxxxx
select * from tecs.environ ;
quit
```

where `-pxxxx` is the current password. The time stamps are in the UTC time zone. The call of `mysqldump(1)` may create a snapshot such that `delete from tecs.environ` could be used to start from an empty data base without losing data.

If `gnuplot(1)` has been installed with

```
zypper install gnuplot
```

one may call

```
ccdtemp.plt
```

to get a graphical output of the past data like in Figure 11. There is an auxiliary crontab as seen with

```
crontab -l
```

which calls `ccdtemp.plt` with the option `-w` to upload the temperature file to the MPIA anonymous FTP server.

3.5 External packages

3.5.1 Required texinfo

For the `king.texi` compilation we need

```
zypper install texinfo
```

3.5.2 Required LabJack

To read the LabJack sensor with the python script above, one needs to install the corresponding https://github.com/labjack/LabJackPython package:

```
curl -L https://api.github.com/repos/labjack/LabJackPython/tarball > LabJackPython.tar.gz
tar xzf LabJackPyhthon.tar.gz
cd labjack-LabJack*
```

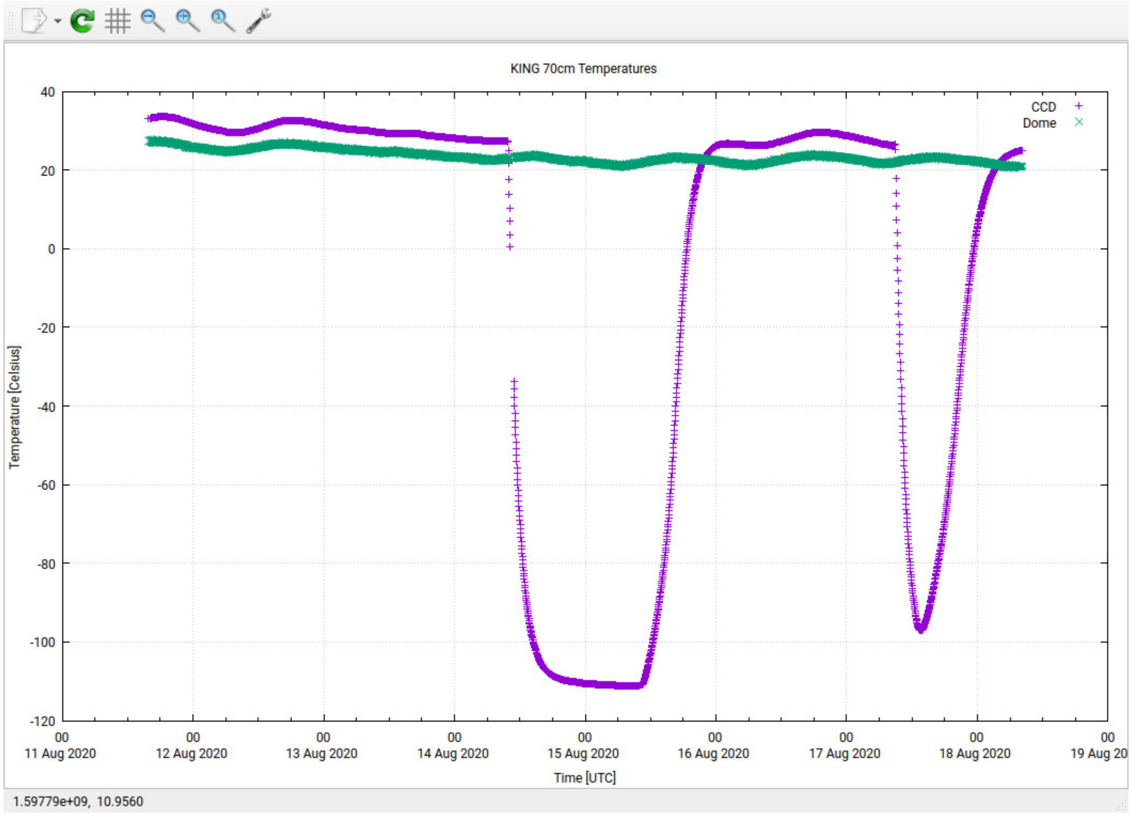


Figure 11: Example of the GUI with `ccdtemp.plt` in August 2020.

```
python3 setup.py install
```

This will generate a file like `/usr/lib/python3.10/site-packages/LabJackPython-2.0.4-py3.6.egg` such that the `import ue9` of python finds the module.

3.5.3 Required mysql-connector

For ccd(un)lock a python package is needed:

```
zypper install python310-mysql-connector-python
```

A problem in June 2022 existed where the 8.0.19 version of that package was not compatible with some requirements of python 3 exception chains which lead to error messages each time an open connection to the data base went out of scope. This was patched by upgrading to 8.0.29:

```
pip3 install --upgrade mysql-connector-python
```

3.5.4 FTOOLS

The heatools have been compiled in the `staff` account as follows:

1. Ensure that you have a recent version of compilers of your operating system, including `gfortran`. You may also need to install the `libXt-devel` package such that `X11/Intrinsics.h` is known.

   ```
   zypper install gcc-fortran libXt-devel # openSUSE
   ```

2. Download the source code from the download page. Select the `Source code` (CentOS or openSUSE or both), not any precompiled binaries, and select the `General-Use FTOOLS`, deselect the `Attitude`, `Caltools`, `HEASim`, `HEASPtools`, `HEAGen`, `Time` and `Xspec` packages which are not of much interest to optical Astronomy, and click `Submit`. Download everything (roughly 200 MB) to `$HOME/heasoft-6.30.1.src.tar.gz`.

3. Unbundle with

   ```
   cd $HOME
   rm -rf heasoft-6.30.1
   tar xzf heasoft-6.30.1src.tar.gz
   ```

4.
   ```
   umask a+rx
   cd heasoft-6.30.1/BUILD_DIR # depending on tar version also ..-6.22.1...
   ./configure # CentOS 7 or openSUSE Leap
   make |& tee build.log
   make install |& tee install.log
   chmod +x headas-init.*
   ```

5. add to `$HOME/.bash_login` or `$HOME/.bashrc` (details of the libc will probably differ, current CentOS 7 systems end in `libc2.17`):

```
export HEADAS=${HOME}/heasoft-6.30.1/x86_64-pc-linux-gnu-libc2.31
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HEADAS}/lib
. $HEADAS/headas-init.sh >& /dev/null
```

and make sure that your terminals are login terminals.

3.5.5 IAU SOFA

The library Standards of Fundamental Astronomy was downloaded as an RPM packages iausofa-c-.....rpm from https://download.opensuse.org/repositories/home:/rjmathar/openSUSE_Leap_15.3/x86_64/ and installed as root

```
cd ~root
zypper install iausofa-c-*.rpm
```

It is used within astro.cxx to calculate sidereal times.

3.5.6 PyRAF

pyraf 2.2.1 was installed with

```
pip3 install --upgrade pip
zypper install xorg-x11-devel # required for X11 header files
pip3 install pyraf
```

iraf 2.17 was installed as root with

```
zypper install libcurl-devel readline-devel libexpat-devel
curl -L https://api.github.com/repos/iraf-community/iraf/tarball > iraf.tar.gz
tar xzf iraf.tar.gz
cd iraf-community*
export IRAFARCH=linux64
./install
make linux64
make sysgen 2>&1 | tee build.log
cd ; mkiraf
```

accepting the defaults (/tmp for image directories,...). A problem here is that mkiraf will manipulate ~/.bashrc and ~/.bash_login, and create ~/.bash_profile; the latter usually will not exist and should be removed. The graphical support was installed via

```
zypper install python3-tk libXaw3d-devel python3-matplotlib tcl-devel tcllib # as root
```

and as the standard user

```
curl -L https://api.github.com/repos/iraf-community/x11iraf/tarball > x11iraf.tar.gz
tar xzf x11iraf.tar.gz
cd iraf-community-x11iraf-*
```

Then the PREFIX variable in Makefile was changed to

```
# PREFIX = /usr/local
```

```
PREFIX = ~/.local
```

The x11iraf was then compiled with

```
make
make install
```

A problem under openSUSE 15.4 is that the linker linking with -ltcl may fail because only the library with the explicit version is present, so as root a

```
cd /usr/lib64
ln -s libtcl8.6.so libtcl.so
```

will be needed. It then is useful (perhaps not required) to add

```
export IRAFARCH=linux64
```

in ~/.bashrc and

```
export PYTHONPATH=${PYTHONPATH}:$HOME/.local/lib/python3.6/site-packages:/usr/lib64/python3.6
# not required but useful to support the ax_python_devel.m4 macro
export PYTHON_VERSION=3
```

in ~/.bash_login.

3.5.7 Optional python3 libs

It was checked with

```
pip3 install matplotlib
```

that the matplotlib is installed and with

```
pip3 search scipy
```

that scipy is installed. photutils was installed with

```
pip3 install photutils
```

3.5.8 Optional sextractor

The SExtractor www.astromatic.net of was downloaded from the openSUSE build service and installed with

```
zypper install sextractor-2.25.0-lp152.35.1.x86_64.rpm
```

3.5.9 Optional siril

The source code can be obtained from https://siril.org. Compilation (under openSUSE) after downloading the source code:

```
bunzip2 siril*tar.bz2
tar xf siril-0.99.8.1.tar
```

```
# tar xf siril-0.99.10.tar # this version does not compile yet...
zypper install fftw3-devel gsl-devel cfitsio-devel cmake
zypper install libconfig-devel opencv-devel intltool glib2-devel json-glib-devel exiv2 libgexiv2-
devel gtk3-devel
cd siril
./autogen.sh
make
make install
```

There is apparently an alternative by downloading the package from https://build.opensuse.org/package/show/openSUSE:Factory/siril or from https://build.opensuse.org/package/show/graphics/siril and installing it with

```
zypper install libavcodec58
zypper install siril-0.99.6-1.4.x86_64.rpm
...
```

but the main problem then is that this RPM demands some codec packages that match *exactly* the versions on the current computer…

3.5.10 Optional xpa

The binary package from https://build.opensuse.org/package/show/home:rjmathar/xpa is installed in the home directory of root:

```
cd ~root
zypper install xpa*.rpm
```

If this is done before compiling the main package, `ccdread3` will send new FITS files with the `xpa` protocol to an existing `ds9` session (if there is one). If otherwise the `xpa` is *not* found at compile time, `ccdread3` will simply pop up a new `ds9` screen with the new image.

3.5.11 Optional Aladin

The Aladin Desktop of http://aladin.u-strasbg.fr/java/nph-aladin.pl?frame=downloading was installed in `/opt/ccd/bin`:

```
tar xf Aladin.tar
mv Aladin/Aladin Aladin/Aladin.jar /opt/ccd/bin
```

3.5.12 Optional

```
zypper install a2ps astyle python3-autopep8 python3-scipy htop nedit
```

3.5.13 fits2csv

The header keyword extractor `fits2csv` is installed from the openSuse build service with

```
zypper install apache-commons-compress
zypper addrepo obs://home:rjmathar/openSUSE_Leap_15.3 rjmathar
# so now exists /etc/zypp/repos.d/rjmathar.repo with  a line
# baseurl=https://download.opensuse.org/repositories/home:/rjmathar/openSUSE_Leap_$releasever
zypper refresh
zypper in nom-tam-fits de-mpg-mpia-rjm-fits2csv
```

3.5.14 fits2fwhm

`fits2fwhm` is a program to reduce a series of FITS files with variable position of an optical element (that is the secondary mirror for the KING telescope) to extract a position that minimizes apparent diameters of the stars. It is obtained from the openSuse build service with

```
zypper addrepo obs://home:rjmathar rjmathar
zypper refresh
zypper in fits2fwhm
```

3.5.15 Optional XEphem

Elwod Downye's XEphem for the openSUSE operating system: download the binary package from from https://software.opensuse.org/download.html?project=Education&package=xephem and install it:

```
cd ~root
zypper install xephem-3.7.7-lp151.7.1.x86_64.rpm
```

This will put catalogs and site files in `/usr/lib/xephem`. If older variants of the package exist, they can be removed beforehand with

```
rpm -qa | fgrep -i xephem
rpm -e xephem-3.7.7-lpxxxx # insert output of the query here....
```

To have a reasonable default, edit `~/.xephem/XEphem` to read

```
XEphem.Elevation:                            560.0 m
XEphem.Lat:                                  49:23:44
XEphem.Long:                                 -08:43:25
XEphem.Pressure:                             948 hPa
XEphem.Sitename:                             MPIA, Heidelberg
XEphem.TZName:                               CET-1CEST
XEphem.TZone:                                -1:00:00
XEphem.TZone.Local.set:                      False
```

(Geographic longitudes in `XEphem` use an unconventional sign where west is positive, which is often used in US American software.) This file needs to be edited two times a year when the German daylight savings convention switches from `-1:00:00` to `-2:00:00` UTC offsets, but everybody can change the offset in the master menu as well.

Additional catalogs havee been added into `/usr/lib/xephem/catalogs` according to https://vixra.org/abs/1802.0035. The may be loaded with the `Data`→`Files` menu and in that submenue with the `files` menue.[4]

3.5.16 Optional astrometry.net

The plate solver was downloaded from https://astrometry.net/use.html and compiled with

```
zypper install python3-devel python3-numpy-devel python3-astropy swig libnetpbm-devel pam-
devel libjpeg8-devel libopenjpeg1
export NETPBM_INC="-I/usr/include/netpbm"
export NETPBM_LIB="-L/usr/lib64 -lnetpbm"
tar xzf astrometry.net-latest.tar.gz
cd *-0.89
make
make py
make install # isntalls into /usr/local/astrometry
find /usr/local/astrometry -type f -exec chmod a+r {} \;
find /usr/local/astrometry -type d -exec chmod a+x {} \;
cd /usr/local/astrometry/data
curl http://data.astrometry.net/4100/cfg > cfg
for i in 07 08 09 10 11 12 13 14 15 16 17 18 19 20 ; do
    curl http://data.astrometry.net/4100/index-41${i}.fits > index-41${i}.fits
done
```

and the `PATH` variable was extended to `/usr/local/astrometry/bin` such that the `solve-field` executable is in the path.

3.6 Compilation

```
cd ~ccd/king/sw
autoreconf -i -s
./configure
# ./configure --prefix=$HOME # while testing with $PATH containg $HOME/bin
make install |& tee  make.log
```

It's recommended to run at least one test readout afterwards:

```
cd ~/ccd/king/sw/ccd/Rocon_dev
ccdstart
ccdread3
ccdstop
```

3.7 Initialization

The `ccdlock` and `ccdunlock` commands store a simple flag into a `mysql` database that is used similar to a semaphore to prevent parallel use of the CCD by multiple users. Initially this database

[4] At startup **`xephem`** loads only a small catalog of ≈ 8800 stars, which is too small for any reasonable sky coverage in the FOV of the telescope.

does not exist and must be created before the lock and unlock commands can be used. The associated script is `Rocon_dev/ccdMySqlCrea`.

One of the requirements here is that the SQL server is running, which is a default in modern Linux systems. If not, run `yast2` as `root`, select `services`, and check that `mariadb` is running.

The status of the SQL database for the `ccdlock` and `ccdunlock` is examined with

```
/sbin/service mysql status
```

If not running it can be started with

```
sudo systemctl enable --now mariadb
sudo mysql_secure_installation
```

where the password was set to the same as the root password on the computer.

3.8 Operating System

3.8.1 Local Network

The RoCon and the LabJack 192.x.x.x addresses have been added to `/etc/hosts` such that commands like `ping` of Section 2.3.2 work from the Linux shell.

3.8.2 DHCP

To work with DHCP on the `king` computer under the `wicked` network manager, the file `/etc/wicked/local.xml` needs to be added according to https://bugzilla.opensuse.org/show_bug.cgi?id=3D1080832#c5, so it reads

```
  <config>
    <addrconf>
      <dhcp4>
        <create-cid>rfc2132</create-cid>
      </dhcp4>
    </addrconf>
  </config>
```

If installed, one can use `yast2`, click on the `network settings` the `Overview` and for the network card on `Edit` and then select `DHCP` for the `Dynamic Address`.

3.8.3 WOL

Wake-on-lan has been installed by adding a file `/etc/system/system/wol.service`

```
[Unit]
Description=Configure wake-up on LAN
After=network-online.target

[Service]
Type=oneshot
```

```
ExecStart=/sbin/ethtool -s eth0 wol g

[Install]
WantedBy=basic.target
```

and `systemctl enable wol.service` such that `wol 00:03:2d:46:9d:05` from a remote computer ought to bring the computer back online if someone put it to sleep. See also the output of `ethtool eth0`.

3.8.4 Memory Defragmentation

The crontab of `root` contains a line

```
28 * * * * sync ; echo 3 > /proc/sys/vm/drop_caches 2>> /opt/ccd/cron.err
```

which regularly drops the caches. The caches contain copies of all the FITS and other files that are produced by the software, which may cause memory allocation problems to get the large consecutive pages required within the PLX driver routine of `ccdread3`. The `Cfiles/Makefile.am` contains an `install-exec-hook` which allows the `ccdread3` to use an equivalent call at the start of the exposure, and it will attempt to drop the caches if the free memory appears to fall below 6 GB.

3.8.5 ccdstop with timeout

Because many operators forget to call `ccdstop` when they are done, a crontab(1) of the type

```
17 * * * * PATH=$PATH:/opt/ccd/bin ; ccdstopTout >/dev/null 2>/dev/null
```

should be added to most users such that after a timeout of 10 hours (the default) after the last `ccdread` at least one `ccdstop` is issued. A different timeout (in units of hours) can be specified as a command line parameter after the `ccdstopTout`. This is basically a precaution against CCD damage. Because people sometimes leave the telescope for a few hours (and may run automated exposure sequences), a timeout much less than 3 hrs is not recommended.

3.8.6 NTP

The computer runs the network time protocol as a client. This is set up with `yast2`, the menu `Network Services: NTP Configuration`, setting the flag on `Start NTP Daemon Now and on Boot`, and adding `time.mpia.de` at the top of the list of Synchronization Servers. See `/etc/chrony.conf`.

3.8.7 Remote Desktops

3.8.7.1 ssh The secure shell daemon has been enabled with the `yast2` editor selecting the `Services Manager`, and setting the `sshd` entry to `On Boot`.

3.8.7.2 vnc The `vnc` server and clients have been made available selecting `remote administration without session management` in `yast2`, `Network Services`, `Remote Administration(VNC)`.

3.8.7.3 remmina The desktop support of `remmina` was added with

```
zypper install remmina remmina-plugin-rdp remmina-plugin-vnc remmina-plugin-www remmina-plugin-xdmcp
```

3.8.7.4 vinagre The desktop support of `vinagre` was added with

```
zypper install vinagre
```

3.8.7.5 rdp Remote desktop protocal of `rdp` was added with

```
zypper install xrdp xorgxrdp
```

and then adding the start of `xrdp` and `xrdp-sesman` on boot in the `yast2 Services Manager`. The command-line alternative is

```
systemctl enable xrdp
systemctl start xrdp
systemctl enable xrdp-sesman
systemctl start xrdp-sesman
```

3.8.8 NX server

If the command `nxplayer` is not found under Linux, use the full path name of the installation to start (`/usr/NX/bin/nxplayer`) or add `/usr/NX/bin` to the `PATH`.

The server was installed by downloading the RPM package from the company and installed (as root) with

```
zypper install nomachine_7.2.3_8*.rpm
```

Just after installation, the NX support is running under an openSUSE system (`ps -elf | fgrep nx`), because `/etc/systemd/system/multi-user.target.wants` contains a `nxserver.service` entry.

To disable this automated start each time the computer boots, use `/sbin/yast2`, the `System` submenue with the `Service Manager`, and disable the `nxserver`. In this case one needs to activate the service explicitly (as root) either from the same menue or by calling `/etc/NX/nxserver --startup`.

3.8.8.1 x2go x2go was installed with

```
zypper ar https://download.opensuse.org/repositories/utilities/openSUSE_Leap_15.3/ makepasswd
zypper ar https://download.opensuse.org/repositories/X11:/RemoteDesktop:/x2go/openSUSE_Leap_15
zypper refresh
zypper in makepasswd
zypper in x2goserver
```

Note that

```
zypper ar obs://X11:RemoteDesktop:x2go/openSUSE_Leap_15.3 x2go
zypper refresh
zypper in x2goserver
```

does not work because package like `makepasswd` are not found at that repository (for now).

3.8.9 Boot Screen

To get some useful information on the booting process (without hitting F1), `/boot/grub2/grub.cfg` was edited to change `splash=silent` to `splash=verbose`, followed by `grub2-mkconfig`.

3.9 Parameter files

3.9.1 4noise.par

3.9.2 adcpatt.par

This is the ADC pattern which is triggered once by the readout process via `ccdread3`. It selects (in that order) ADC's number 2, 1, 3 and 4 for transfer via the fiber; see `dac-1.par` for the bit mask associated with the last integer in the 511 commands.

The 560 command selects the OCTADC board (bit number 5 set).

The 512 command loops once through the sequence of the 4 boards, with a divisor of 2 (period of 30.3 ns); one trigger means submitting the 4 values needs 18 × 30.3 ns = 0.54μs. Because that is shorter than the approximately 1 MHz speed of the ADCs on the board, the pattern is capable of transmitting all data before triggered by the next EOC in the main pattern generator.

The files `adcpatt`*i*`.par` select ADC number 2 ($i = 1$), ADC 2 and 1 ($i = 2$), ADC 1, 2 and 3 ($i = 3$) and ADC 1, 2, 3 and 4 ($i = 4$) for transmission through the fiber. The standard operator is only interested in `adcpatt1.par`, because the other ADCs are not connected to the detector.

Note that `adcpatt3.par` and `adcpatt1.par` are implemented by sending one of the ADC values twice. [The reason is that the fiber transmits always 32bit-values, which means 2 ADC values (2 16-bit data) a time; it would be complicated to synchronize the basic protokoll by submitting an odd number of values per trigger.] So `adcpatt1.par` sends the value of ADC 2 twice per trigger, `adcpatt2.par` sends ADC's 2 and 1 per trigger, `adcpatt3.par` sends ADC's 2, 1, 3 and 2 (again) per trigger, and `adcpatt4.par` sends ADC's 2, 1, 3 and 4 per trigger.

The line endings in files like these that are send in a block to the ROE are carriage return plus line feed, because the `srdata` interface reads them with `fgets` and just copies them over, where the ROE expects both, carriage return and line feed. For the same reason there is no privision for comments in these files.

3.9.3 ccdname.par

3.9.4 ccdread.par

This is a volatile file, which means it may be created for each individual read. It collects the principal parameters of the setup, exposure time, shutter status, the two binning factors, coordinates of the rectangle of the region that is read, and FITS file to be created. Because `ccdread` converts all its parameters into command line arguments of `ccdread3`, calling `ccdread` or `ccdread3` do not generate such a file; the remaining principal use of the file is to characterize the continuous scan within `ccdstart`.

The file contains at least 10 lines (all followup lines are ignored) which show a keyword, a colon and a value. Only the values are important, and must (in that order) contain

- an exposure time in seconds
- a shutter open specification (0= closed, 1 = open)
- a horizontal binning factor ≥ 1
- a vertical binning factor ≥ 1
- a gain parameter from the set shown in the `ccdread3` command
- a lower left horizontal coordinate of the radout window (≥ 1)
- a lower right horizontal coordinate of the radout window (≥ 1)
- a lower left vertical coordinate of the radout window (≥ 1)
- a upper right vertical coordinate of the radout window ($\geq 1, \leq 2048$)
- a FITS file name

3.9.5 ccdtemp.par

Contains basically the TCP/IP information on the interface to the W&T sensor, currently in the science net. This should be the same IP4 address as the output of

```
nslookup wut70cm
```

Not if interest to the standard observer.

3.9.6 chip.par

3.9.7 chipname.par

3.9.8 dac-1.par

The 4 files `dac-`i`.par` for $i = 1 \ldots 4$ contain programs for the ADC pattern generator that (once triggered) select the ith ADC (and always the same OCTADC board) and send it once into the fiber. The divisor value of the clocking is always 0 in the 512 command (7.5 ns), but there are 5 periods that select and de-select the board, so the actual duration of triggering such a pattern once is 75 ns.[5]

The board-select parameter in the 560 command is always 0x20, which is setting bit number 5 for the OCTADC board.

[5] Obviously one may have chosen a different divisor and less than 5 periods here.

The channel select bit is compatible with the binary mask 01010101 for the four ADC's. $i = 1$ is 00000001, $i = 2$ is 00000100, $i = 3$ is 00010000 and $i = 4$ is 01000000. The last parameter in the 511 command is the assocated decimal value, 1, 4, 16 or 64.

The `dac-1.par` file is post-loaded to the ROE in `ccdread3` after the standard readout and fiber transfer are terminated (see `adcpatt.par` above) and if the `cycles` parameter in the `loc.par` files has the value 1. That is a type of debugging aid supposed one has clamped an oscilloscope to the output lines and would like to get rid of the data of the ADC's 2 to 3 that are usually not of interest but may be transferred in the standard readout pattern.

3.9.9 datastream.par

3.9.10 focus.par

3.9.11 interface.par

3.9.12 lan.par

3.9.13 layinf.par

Contains pairs of numbers per line: a non-negative start index into the registers of layer 1 of the FPGA, and a clocking floating point number, which is a period in units of microseconds to be applied to layer 0. The end-index is implicitly defined as one less than the followup start index. The file must contain at least 14 lines (trailing lines will be ignored). There is no provision for comment lines (but material after the floating point column will be ignored for that matter). [7, §5.2.2]

The speed of the readout is governed by the time needed for a single pixel. Layer 2 includes lines 0 to 26 (inclusive) of layer 1 to shift by a pixel, so this needs 0.2+3.9+6+0.3+0.9+6+0.2+0.2+0.1+0.2 μs = 18 μs. (Note that this is a 60 kHz rate and much slower than what the ADC on the ROE could sustain.) For a binned-2 row this amounts to approximately 1050×18 μs = 18.9 s. The vertical shifts are 8×80 μs of lines 27–34 (inclusive) of layer 1 per line, which adds to roughly 0.6 s for 1024 lines. Altogether this illustrates the prototypical 20 s to obtain a frame.

3.9.14 loc.par

3.9.15 offset.par

3.9.16 qamp_volts.par

3.9.17 rampat.par

3.9.18 set-offset.par

3.9.19 voltnames.par

The final contents of the file contains a sequence of positive integer numbers which may be send with the command "33 1013" in that order to the RoCon. Read by `potvar`.

3.9.20 wut-IP.par

Contains basically the TCP/IP information on the interface to the W&T sensor. Not if interest to the standard observer.

3.10 Data files

If `ccdread3` is called with the `-v` option, the PLX driver in `plxread.c` writes all all raw data received from the ROE into a file which has the same name as the FITS file but the suffix `.dat`.

These is four or two unsigned 16-bit values for each (binned) pixel, including the values of the 3 or 1 dummy ADC channels that are not detector data. Whether these are two or four depends on the use of the `-A` option in `ccdread3`. One way to look at these is

```
od -d temp.dat
```

or to skip the byte index and the 3 dummy channels in each output line with

```
od -d temp.dat | awk 'print $2,$6'
```