

# ASTROART 4.0

## CCD CONTROL - USER INTERFACE Version 4.61

### TABLE OF CONTENTS

1	Manual Conventions.....	2
2	Installation.....	2
3	Let's Start.....	2
3.1	Hardware setup .....	2
3.2	Software setup.....	2
4	The Image Page.....	4
5	Filter wheel.....	4
5.1	The Filter wheel Setup page .....	5
5.2	The Filter wheel Filters page.....	5
6	Telescope Control.....	6
6.1	GOTO Control panel.....	8
6.2	Autocenter .....	9
7	Guiding.....	10
7.1	Tutorial.....	10
7.2	Manual Guiding.....	11
7.3	Autoguiding.....	12
7.4	Selfguiding.....	15
8	The Settings Page.....	16
9	Focusing .....	17
10	The Dark/Flat Page.....	18
11	The Sequence Page .....	19
12	Scripts.....	20
12.1	Demos .....	20
12.2	Types, Variables and Functions.....	21
12.3	Loop instructions.....	25
12.4	Conditional instructions.....	27
12.5	Other functions .....	28
12.6	Automatic research.....	29
13	History.....	31
14	Contact Information.....	31

## 1 Manual Conventions

Welcome to the new CCD User Interface of Astroart 4.0. This document contains information about control of CCD cameras, Telescopes and Filter wheels. This manual uses the typefaces and symbols listed below:

- The menu bar, sub-menu and floating menu items are highlighted in bold and embraced in square brackets: submenus commands are noted with an arrow (i.e. **[Tools]** ⇒ **[Star Atlas]**) this means that you should click on the **[Tools]** menu, then click **[Star Atlas]** on the submenu.
- The buttons to be pressed are highlighted in bold and enclosed in a box (i.e. **Setup**) means to click the button labelled 'Setup').

## 2 Installation

The “CCD User Interface” is a plug-in for Astroart which provides a high sophisticated control for your CCD camera. To install this plug-in simply copy the file **piccdgui.dll** into the Astroart directory. To command a particular CCD camera you may need also a driver, available for free at Astroart web page:

[www.msb-astroart.com](http://www.msb-astroart.com)

## 3 Let's Start

The Astroart CCD User Interface contains some simulators for CCD cameras, Telescopes and Filter wheels, which let you test all functions indoor before using your real hardware. Take a look at **Chapter 7** for a quick tutorial.

### 3.1 Hardware setup

At first check the camera position and linking: be sure to securely attach the camera to your focuser. Once you have safely connected the camera to the computer you can turn on all the hardware. Now, start Astroart and select from the menu **[Plug-In]** ⇒ **[CCD Camera]**: this will display the *Setup Page* (see Fig.1)

### 3.2 Software setup

The first option to select is the CCD model: select your CCD camera within the combo box (you may need to download and install the appropriate CCD drivers<sup>1</sup>) then click on **Setup**. If the camera is not correctly detected, we suggest to consult the documentation of the CCD Driver, which is specific for every model. To make some indoor tests use the “CCD Simulator” integrated in Astroart.

To command an auxiliary CCD camera click the second Setup button. The CCD Interface 4.10 let you command up to three cameras:

- A main CCD Camera for imaging and autoguide.
- A guider camera connected to the main camera (SBIG, Starlight-Xpress, etc.)
- A secondary CCD Camera to take short exposures (sky monitoring, etc.) or to autoguide.

---

<sup>1</sup> Read carefully the instructions included with the CCD driver of the selected camera.

A further possibility to command two or more CCD cameras it to execute two sessions of Astroart. This can be very useful if you need to autoguide on the same subject for many hours (example: a variable star). The Astroart session which is guiding could be minimized to icon so it will not disturb the other Astroart session which remains free to take images and process them.

The Setup Page contains some hidden options which can be showed clicking the button **Options**.

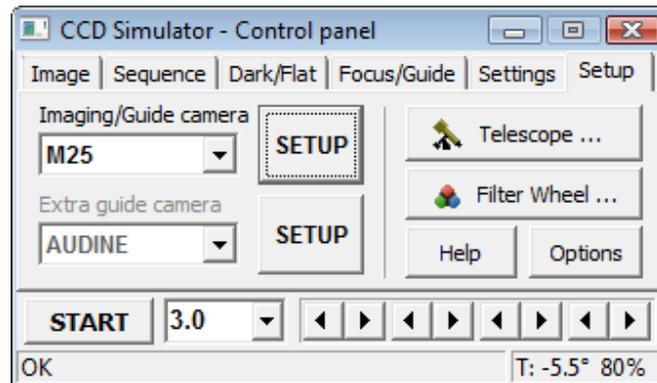


Fig. 1 – The Setup Page

**Visualization (automatic and custom).** The view mode for every new image. If "Auto visualization" is not selected then it's possible to set the minimum and maximum visualization threshold and the transfer function for every new image, for more details see the Astroart on-line guide. The option "Auto (soft)" provides a natural look, "Auto (hard)" is useful for asteroid and supernova search.

**UT and LT.** Universal time and local time, write in the UT-LT box the difference in hours. The button "Check clock" displays for a few seconds the time of the PC to verify it with a precision better than one second.

**Sound on.** Plays a sound to alert the user about important events, likes the lost of the guide star.

**Black background.** Displays a black frame below the Astroart desktop to hide the *Windows Desktop* and all the other applications.

**Visible progress bar.** Enables or disables the red gauge which indicates the download status. On some slow PCs the bar causes stripes over the image.

**Restore panel on top.** On some PCs the CCD control window disappears behind the Astroart desktop, if this happens, select this option.

**High priority.** If selected, during the download the *priority* of Astroart task is increased. This may reduce the noise on some cameras.

**Resync clock.** Select this option to resync the PC clock with the BIOS clock. This can be useful if the CCD driver disables the interrupts. This options does not work under Windows 2000/XP.

## 4 The Image Page

To integrate a new image select the *Image Page* (Fig.2) write with the keyboard an exposure time in seconds (example: “0.002” for two milliseconds) then click the **Start** button. The exposure time can be also set clicking the arrow buttons.

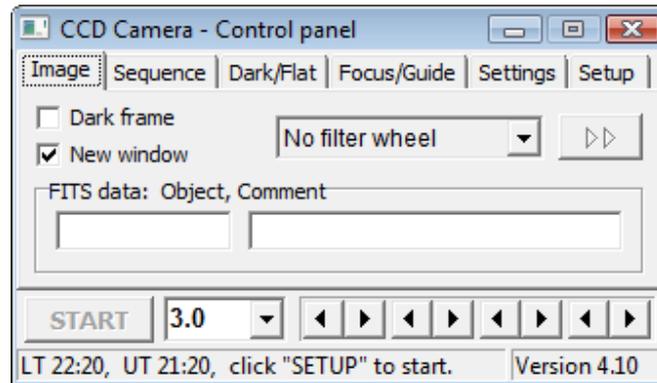


Fig. 2 - The Image page

**Dark frame.** Select this option to close the shutter during the exposition, if your camera has not a built-in shutter you will have to cover the telescope. Click **Start** to integrate the dark frame which will be stored also in memory: if you select the “*Enabled dark frame correction*” option in the *Dark/Flat Page* (see Fig. 15) every new image acquired will be automatically corrected with this dark frame.

**New window.** Every new image is usually displayed into a new window, but if this option is disabled the new image will overwrite the previous one (if they have the same size). This is useful when doing test images or sequences.

**Filter wheel.** Select the filter name and click the arrow button to move the wheel to the desired position.

**FITS Data.** Here you can write the object name and a comment for the FITS header of the image. All other data like date, time, temperature (if supported by the driver) are added automatically.

## 5 Filter wheel

To activate the filter wheel select the *Setup Page* (Fig. 2), click the **Filter wheel setup...** button, then select your model, the serial port, and click on **Connect** (Fig. 3).

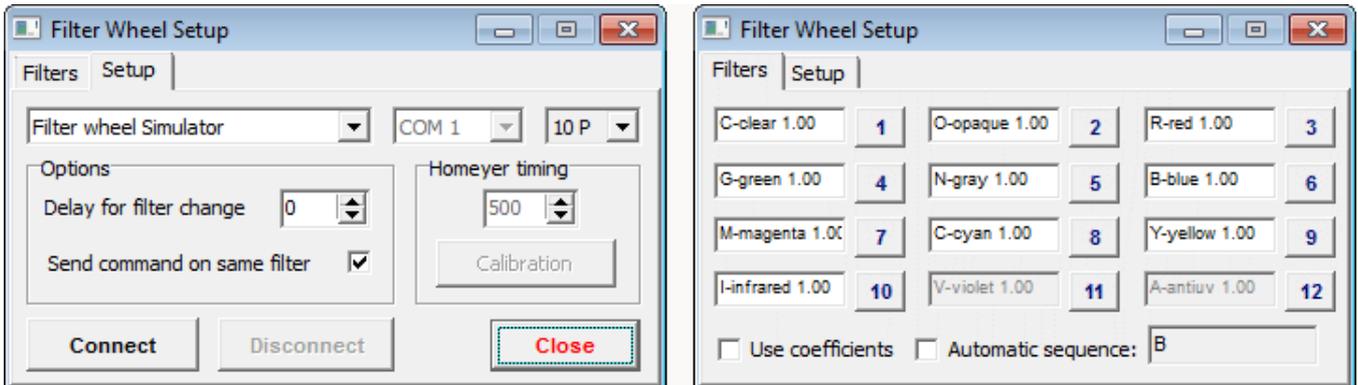


Fig. 3 - The Filter Wheel pages

## 5.1 The Filter wheel Setup page

**Model.** Select from the list your filter wheel. The list may change depending on the model of your CCD camera.

**Com Port.** If your filter wheel is controlled via the serial port, select here which port you are using.

**Send command on same filter.** If enabled, Astroart sends commands to the filter wheel also when the filter is in position yet.

**Homeyer calibration.** The Homeyer filter wheel, if commanded on pin 7, needs to be calibrated with the speed of the PC. If the value measured with the calibration does not work, try to increase it by 10 or 20.

**Delay.** This number indicates how many extra seconds to wait for, after the wheel movements.

## 5.2 The Filter wheel Filters page

**Filter names.** Accordingly to your filter wheel, the appropriate number of filters will be highlighted on the filter page. Take care of the following syntax when you modify the filter name in the text boxes: at first write a letter which identifies the filter. This is very important for the automatic sequences (see below). Then write a minus sign followed by the complete name of the filter and a coefficient: this coefficient could be taken into account during a sequence as a multiplicative factor for the exposure time.

Syntax: `[Filter letter]-[Filter name] [Filter coefficient]`

Example: `B-Blue 1.50` ; where the first letter is used as a flag in sequences.

**Automatic sequence.** If enabled, the filter wheel will be moved before every exposure of an automatic sequence and the first letter of the filter (example: R for Red) will be appended to the filename.

**Use coefficients.** If enabled, the exposure time of every image will be modified by the given coefficient. This may be useful to compensate the relative sensibility of the CCD with every filter.

## 6 Telescope Control

To control the telescope click the button **Telescope Setup...** in the *Setup Page (Fig. 1)*.

In the Telescope Window select the appropriate protocol in the *combo box* (or the Telescope Simulator) and click on **Connect**. (Note that it's possible to *connect* also via the *Guide Window*, see chapter 7). See also the Tutorial available in the next chapter.

**Com Port.** The serial port where the telescope is connected.

**Protocol/Interface.** Select here the protocol used to command the telescope (many new mounts can emulate the GOTO protocol of the LX200, see the user manual of your hardware).

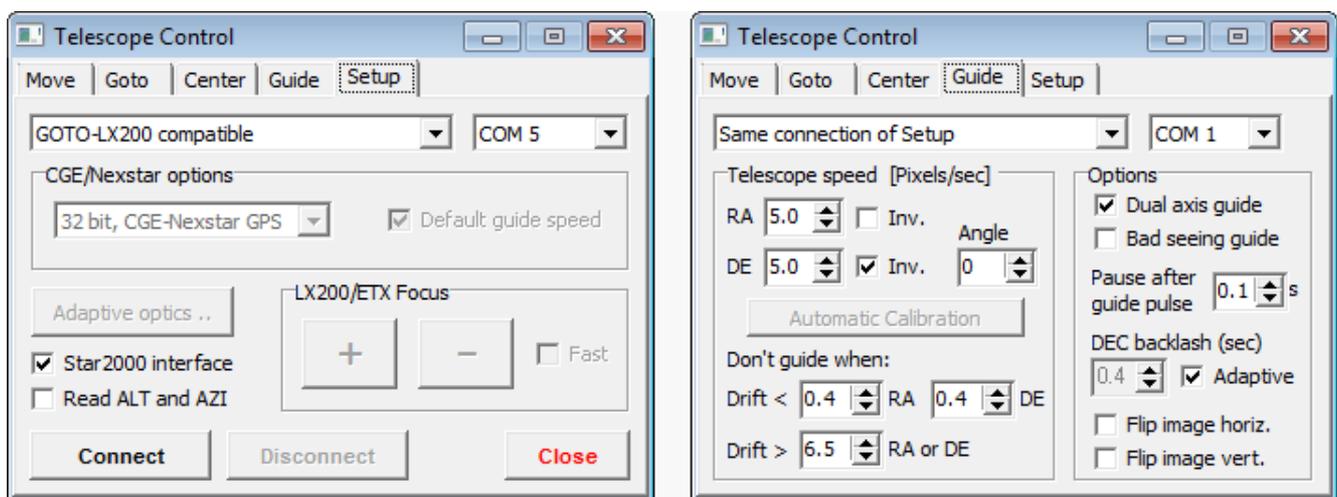


Fig. 4 - The Telescope Setup page

- **Camera Relays.** Relays integrated in some CCD cameras (SBIG, SXV, etc).
- **Telescope Simulator.** A useful simulator to test indoor the Astroart telescope control.
- **LX200 or GOTO compatible.** LX200 and GOTO mounts, STAR2000 in LX200 mode.
- **LX200GPS new firmware.** Latest releases of the LX200GPS with RS232 autoguide commands.
- **ETX or GOTO compatible.** Meade™ ETX telescopes and simple GOTO mounts.
- **ASCOM driver.** Meade™, Celestron™, Astro-physics™, AstroOptiks™, Gemini™, SkySensor™ and many others. This option needs the Ascom libraries available at: <http://ascom-standards.org/>
- **Celestron CGE/Nexstar.** Celestron CGE and Nexstar series.
- **Starlight-Xpress Relay box, ST4 mode.** Autoguider box and STAR2000 relays output.
- **Cookbook Relay Box 300 baud.** Relay boxes designed for the CB245 (300 baud).
- **Cookbook Relay Box 9600 baud.** Relay boxes designed for the CB245 (9600 baud).
- **Audine Relay Box.** Relay boxes originally designed for the Audine camera.
- **VSSI direct cable.** Very Simple Serial Interface cable, a simple connection from the serial port to a telescope mount to correct the R.A. errors only. The signal DTR is the command RA+, the signal RTS is the command RA-. If the telescope is polar aligned this system is sufficient for every purpose.
- **LX200 Shared port with SkyMap PRO.** An inter-process connection with the planetarium SkyMap™ PRO which allows both programs to share the same serial port.
- **MTS-3.** PowerFlex MTS-3 (autoguide and CCD centering only).

- **Interprocess Communication.** Interface with custom programs, see the Plug-in SDK.
- **Verb Interface.** A simple and cheap interface ST-4 compatible, which is connected to the serial port of the PC (a USB->Serial converter may be used too). Does not require any power supply. Designed by P. Mergan, schematics available at the Astroart web site.
- **StarlightXpress Adaptive Optics.** A great tool to guide your images without moving the telescope.
- **Shoestring USB.** A new cable which converts the USB port to a standard ST4 guide port (4 digital signals and ground). More details at: [www.ShoestringAstronomy.com](http://www.ShoestringAstronomy.com)
- **Shoestring Parallel port.** Shoestring GPINT parallel port adapter.

**AO Center.** Click this button to move the AO mirror/prism to the middle position.

**STAR 2000.** Enable this checkbox to initialize a Starlight-Xpress™ STAR2000 interface.

**Read ALT and AZI.** If enabled, Astroart reads the horizontal coordinates from the telescope. This can be slow down the system on some mounts.

**Focus.** These button control the integrated focuser of the LX200. Other focusers can be controlled by the ASCOM autofocus plugin, available at the Astroart web site.

## AUTOGUIDE SETTINGS

---

**Telescope speed and Angle.** The relative speed of the telescope in pixels per second. To measure this value simply move the telescope with the keypad for one second and measure how many pixels it moves. In Astroart this parameter is not critic, an error of  $\pm 50\%$  will be automatically compensated, this means that there is no means to care about the  $\cos \delta$  factor.

The “Inv.” options must be set if the autoguide frames are flipped horizontally or vertically. You may also compensate this also with the “Flip Image...” options in the other panel.

The “Angle” parameter is the rotation of the camera relative to the equatorial axes, again an error of  $\pm 30^\circ$  will be compensated by the autoguid e algorithm. Obviously it is also possible to calculate all these values with the Calibrate button.

**Calibration.** Starts the automatic calibration to find out the telescope speed and the angle.

**Dual axis guide.** If enabled Astroart drives the telescope two axes at a time. This feature is important for altazimuthal telescopes where both motors need to be controlled at the same time. For equatorial mount this option is not recommended because the DEC axis changes very slow depending the precision of the polar alignment, while the fast DEC changes (wind, bad seeing) must be ignored.

**Bad seeing guide.** A special mode for extreme conditions (bad seeing, wind), it's not recommended for normal guide. If this option is not selected the guide will be adaptive in any case, but a different algorithm will be used.

**Don't guide when.** A useful feature to prevent unwanted corrections for small drifts caused by bad seeing. 0.5 pixel is a good compromise between precision and rejection of noise. Values lower than 0.5 should be used only on a very short focal length (photo lens, for example).

**DEC backlash.** Backlash may exist when the declination motor changes its direction. Usually Astroart corrects automatically the DEC backlash (the Automatic correction check box is enabled by

default) unless you set a specific time compensation (in seconds): in this case be careful! Backlash corrections should be always under compensated: an over compensation will lead to an excessive correction and to correct a small error, there could be an overshoot. Since guide speeds in Right Ascension typically are less the sidereal time, no reversal occurs in this axis and you don't need correction for RA backlash.

## 6.1 GOTO Control panel

The Telescope Window can also control the computerized mounts which are GOTO compatible (LX200™, Celestron™, etc.)

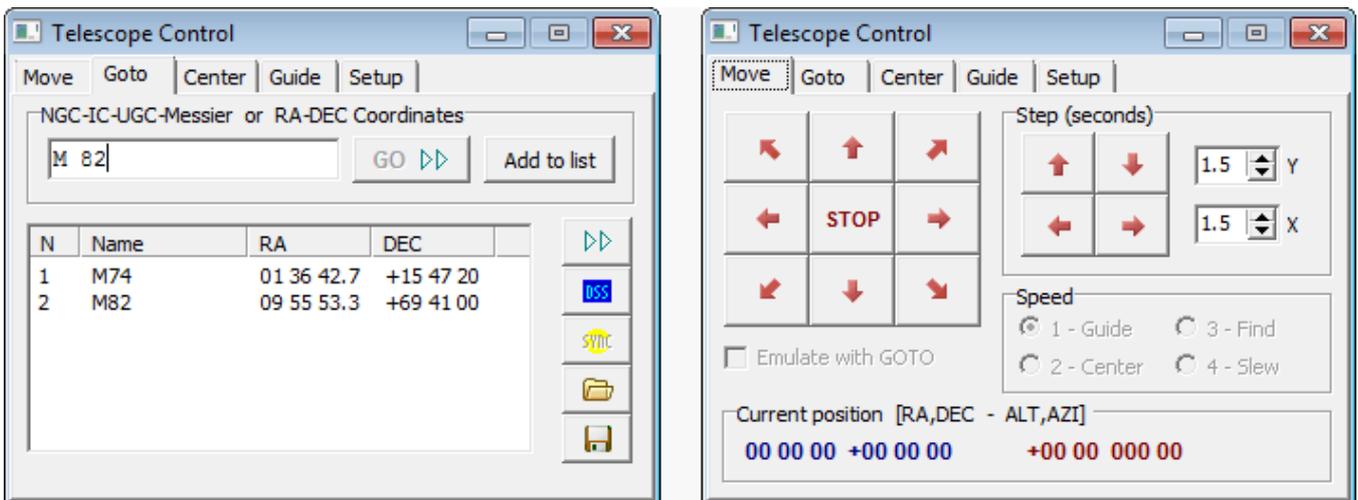


Fig. 5 – The Goto and Move pages

**NGC-IC-UGC-Messier.** Type here the name of a deepsky object (example: N 4565, M 65, U 345 etc.) or some RA/DEC coordinates (example: 18 34.3 +34 56) then click **GO** to slew the telescope to that object. The blank space between the catalogue and the object number is always required ("N4565" won't be accepted).

**DSS.** Opens a Digital Sky Survey image of the object. This function needs the DSS plug-in.

**Resync.** Matches the current position of the telescope with the last object coordinates.

**Custom list.** Manages lists of objects. To create a list write a text file (or a Excel file saved as .CSV) where every rows contains: Name, RA, DEC. The format of the coordinates is free, but the declination must contain the sign. The Name of the object cannot contain spaces. Example:

```

NGC4567 12 34.8 +78 45
"M 67" 12 34 56 +78 23.8
PK456+789 12 12.2 + 34 34 34
"UGC 3456", 12, 14, 16, +34, 54, 34 // Comments this way.
"UGC 4567", 12.23445, -23.23456
    
```

The last two rows are CSV [comma separated values] which can be exported and imported by Microsoft Excel. To delete a row click the right mouse button.

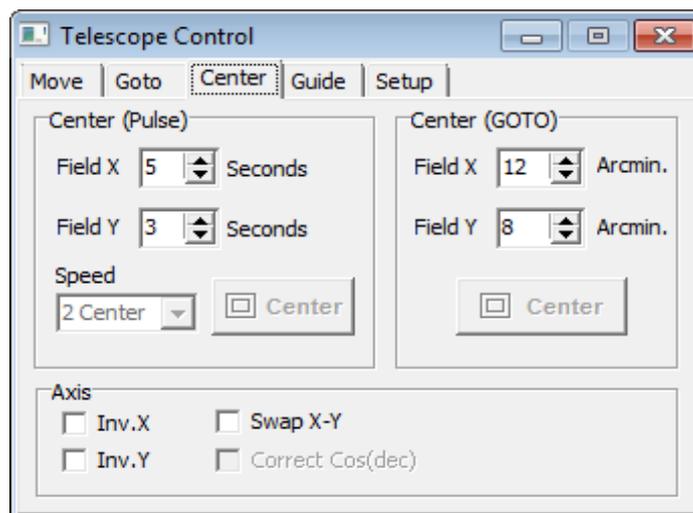
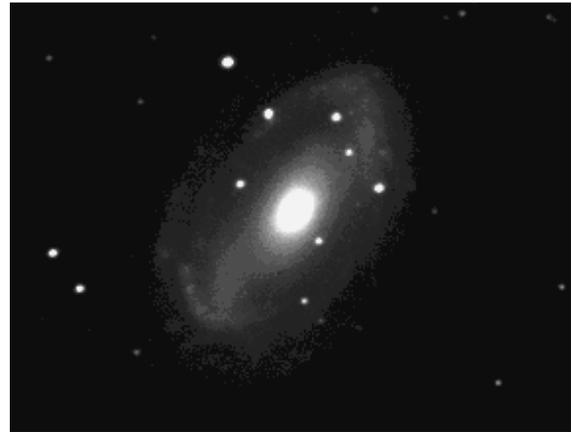
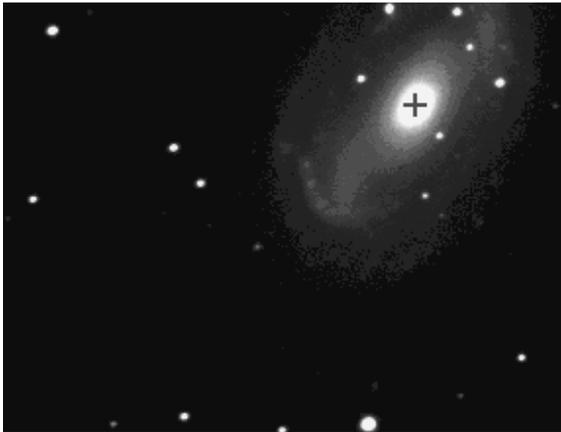
**Arrow buttons.** A “virtual keypad” to move the telescope, useful to center an object. The option “**Step**” will force the duration of every movement of the telescope to a given number of seconds, useful for mosaics and surveys. Please note that some Ascom drivers do not support this function, in this case select the following option:

**Emulate with GOTO.** Select this option if your scope cannot be controlled for simple North, South, East, West commands but supports the GOTO protocol and has a STOPGOTO command. The speed depends by the telescope, probably 1 - 2 degrees per second.

**Speed.** The telescope speed, this option is not supported by some telescope. Remember to select “guide” before autoguiding.

## 6.2 Autocenter

This function centers the object in the field of view of the CCD.



At first select a point over the object (a cross will indicate it) then click the **Center** button. Two methods are available to center the object:

**Center (Pulse).** This is the recommend method. The telescope will be moved using the pulse commands (north, south, east, west) at the given speed. The field of view of the CCD image must be known in seconds(time). This parameter is simply the time necessary for the telescope to move for a full field. The field X can vary slightly with the declination, to correct this effect measure the field X at the equator and enable the option “Cos(declination)”.

**Center (Goto).** The telescope will be moved with a GOTO command (Goto(ra+dx,dec+dy)). This method is not recommend since at the end of the GOTO it won't be possible to sync the position to the object. The field of view of the image must be known in arcminutes. The field X varies with the declination and it's possible to enable the option “Cos(declination)”.

## 7 Guiding

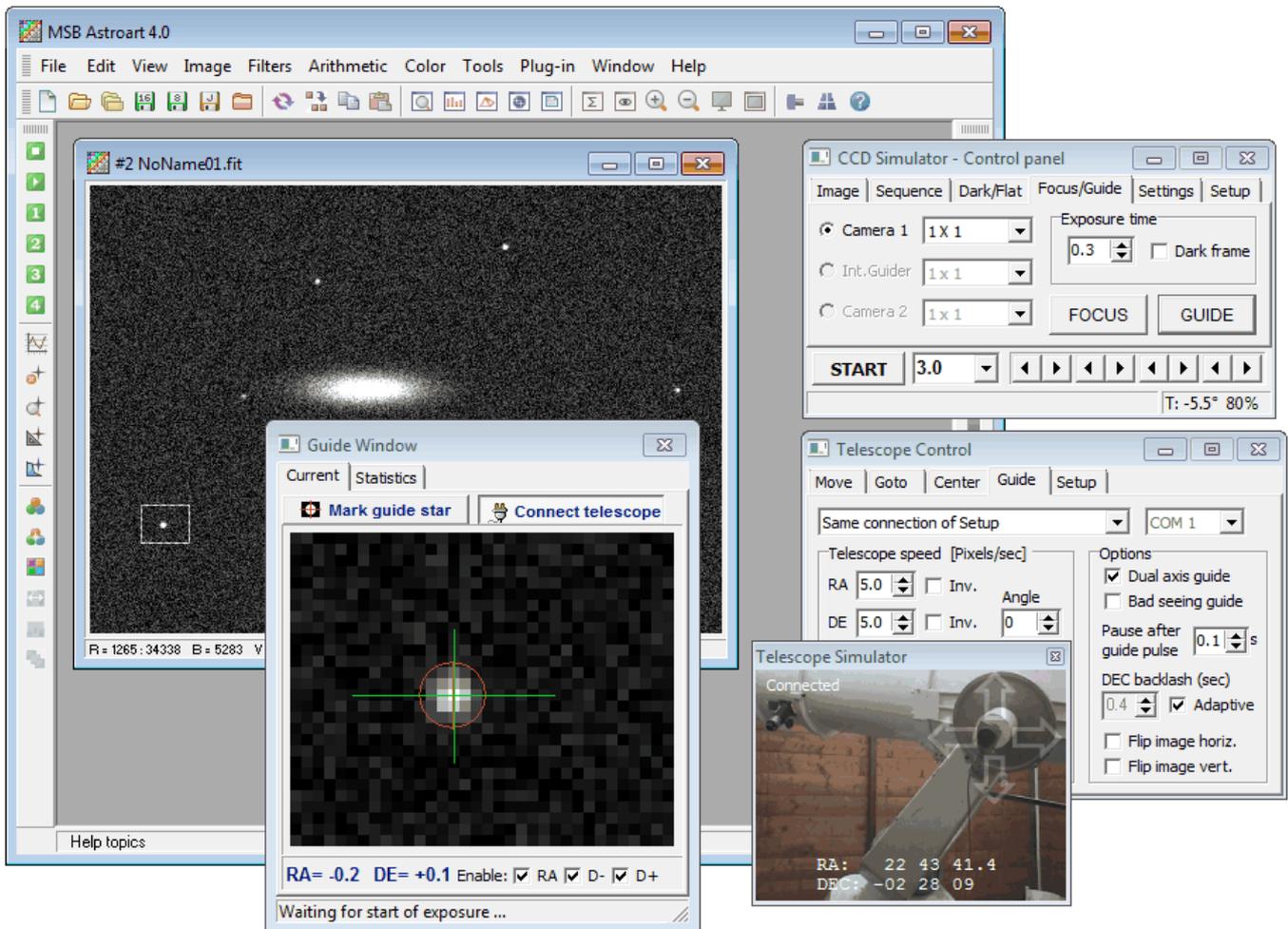
If you follow a star with your telescope at high magnification, you will notice that the position of the star changes, this movement is caused by three causes:

1. A poor polar alignment which could cause a slow drift and a slow rotation of the field of view.
2. The periodic error in the mount's tracking rate: this error results from gears that are slightly out of round. Some mounts have a built-in periodic error corrector called 'PEC'.
3. The random errors due to many causes as dirt, dents and variations in the gears. If the random errors are large and fast enough, they may make an unguided exposure almost impossible.

### 7.1 Tutorial

To quickly understand how the CCD and Telescope work together during an autoguide session, try this step by step tutorial:

1. Select "SIMULATOR" in the driver list and click on "SETUP".
2. Click on "Telescope ..", select "Telescope Simulator" as protocol and click "Connect".
3. Close the Telescope Window.
4. Click "Start" to start an exposure.
5. Draw a small rectangle around a bright star and click "Guide" in the Focus/Guide page.
6. Click "Mark guide star".



7. The telescope simulator will now guide on the reference star.
8. To stop it, click the button "Connect Telescope" in the Guide Window, the star will slowly drift away. Click again the button to restart the guide.

Remember that for a real guided exposure the tracking CCD should be aligned with the axes of the telescope so that motions in Right Ascension and Declination cause the star to move parallel to the CCD array (see Fig.6). Nevertheless Astroart tolerates a quite wide range of rotation, up to 25 degrees.

The CCD User interface is designed to support three kind of guiding: *Manual guiding*, *Autoguiding* and *Selfguiding*.

## 7.2 Manual Guiding

This is the simplest way to guide your exposures and goes back to the times when digital imaging did not exist and the astronomers had to guide with a control pad and an eye at the guide telescope. Now, with the software tools in Astroart you can guide watching the monitor of your PC.

Usually you will use two imaging devices: one camera for guiding (a CCD camera or a WEB-cam) installed on a guide scope, and one camera for imaging (a conventional film camera or a bigger CCD camera) installed on the main telescope.

You will need also a motorised mount with a control pad. The Astroart Guide Window will let you easily guide the telescope watching a reference star on the screen. See: Fig.9.

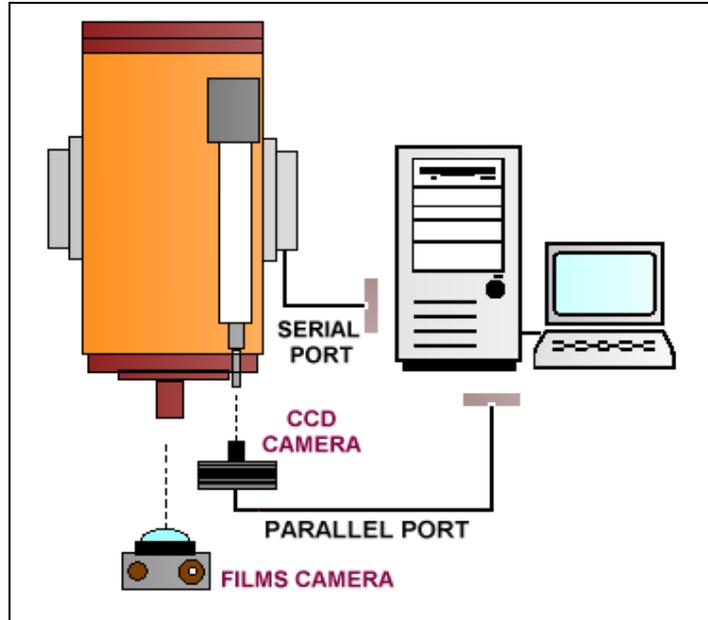


Fig. 6 – A typical telescope set-up for manual guiding: the CCD camera could be a simple WEBCAM.

### 7.3 Autoguiding

Autoguiding means to use a CCD camera (or a WEBCAM) to guide the telescope for another instrument. Astroart measures the position of a reference star (the guide star) on the tracking CCD, and sends the appropriate commands to both the axes of the telescope mount to correct the guiding errors. As explained in fig.8 three scenarios are possible:

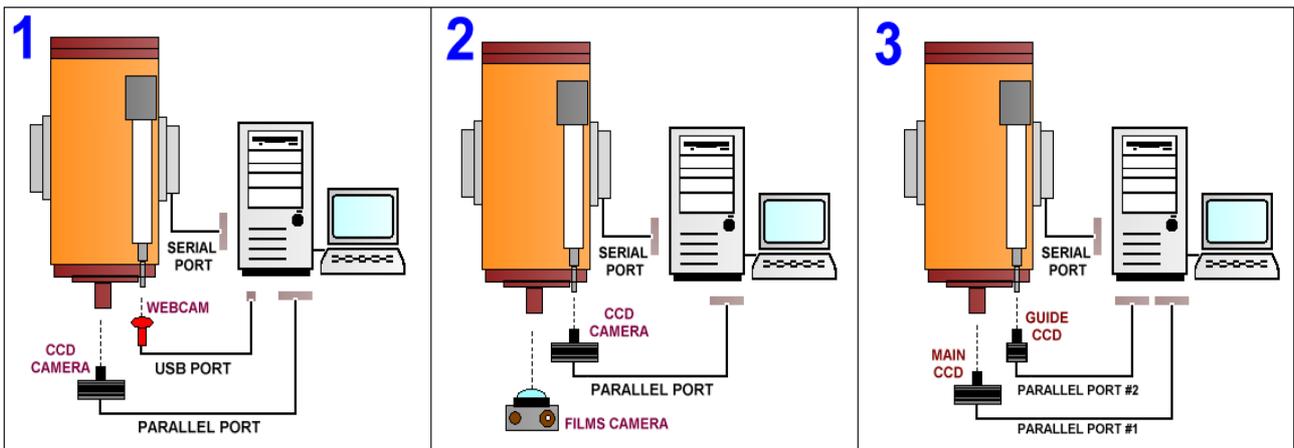


Fig. 7 - Three ways for autoguiding

- **A CCD camera and a WEBCAM as autoguider** : Fig.8-1. You need a conventional CCD camera and a WEBCAM (via USB). Due to the poor sensitivity of most WEBCAMs you'll be able to guide only with a bright star in the field.
- **A FILM camera and a CCD camera as autoguider.** Fig.8-2. Film cameras can still produce stunning images, sometimes better than CCDs. With this setup you can use a CCD or a WEBCAM to guide a film exposure.
- **Two CCD cameras.** Fig.8-3. A main CCD camera as imager and a low cost CCD camera as autoguider.

A further possibility is *selfguiding*. This is equal to the third option (Two CCD Cameras), but the autoguider CCD and the imaging CCD are integrated in the same camera.

To start a guided session using a CCD camera or a WEBCAM with Astroart, you should follow these steps:

1. Take an image with a short exposure time (example: one second): the exposure should be sufficiently long to identify a guide star in the field of view of the CCD, but short enough to let the guiding system works correctly with the same time: you should not exceed 1.5 - 2.0 seconds. To improve the sensitivity of your camera, change the binning (from 1x1 to 2x2).
2. Draw a rectangle around the guide star. A brighter guide star will allow you to use a shorter *exposure guide time* but if your mount is accurate and very stable, you can use longer exposure and therefore dimmer guide stars.
3. Go to the Focus/Guide Page (see Fig. 13), select in the combo box the same exposure time ("exposure guide time") and the same binning of your test image, then click the Guide button.
4. After a while the guide star will appear in the *Guide Window* (see Fig. 8). Click on the **Mark guide star** button: a green cross will automatically mark the initial position of the guide star, while a big red cross will follow the star while its shifts from the original position. If you are going to guide manually you simply have to act on your telescope control pad to keep the green cross over the red one.

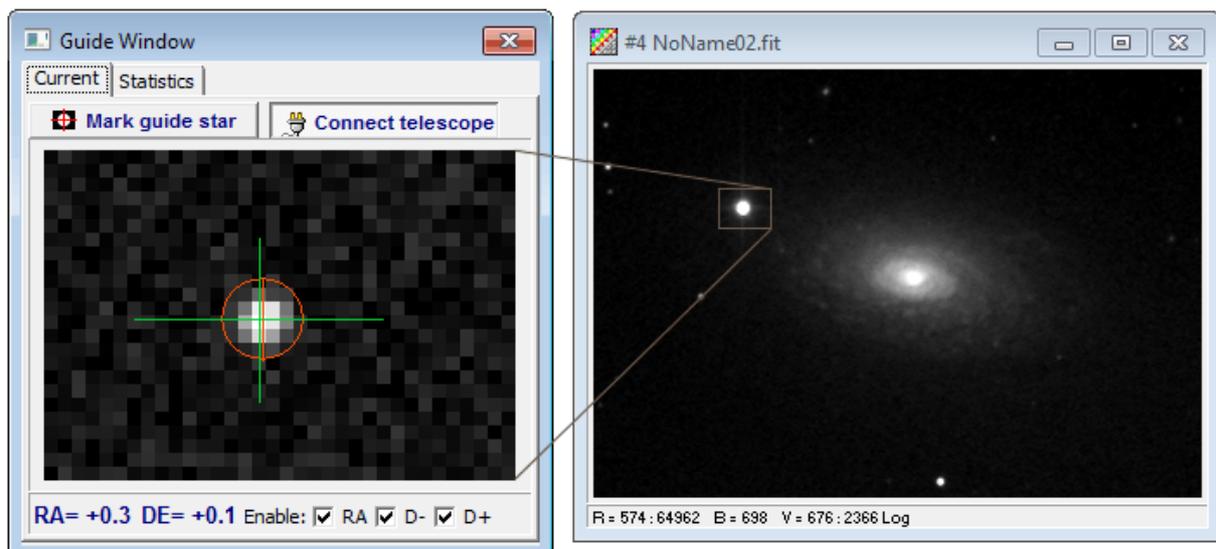


Fig. 8 - The Guide Window

5. Click on the **Telescope** button to activate the telescope and to start the guiding session, remember that the telescope parameters should have been previously set in the *Telescope Window*.
6. If required enable the Guide X and Guide Y checkboxes. If the telescope is well polar-aligned you will obtain better results guiding only in X (RA). If the telescope is not polar-aligned you may enable only the Y+ or the Y- checkboxes (depending on the drift you see), this will prevent unwanted corrections caused by bad seeing.
7. If the corrections go in the wrong direction re-open the *Telescope Setup* window and select *Inverse* for the X axis (or the Y one). If the CCD is mounted rotated of 90 degrees remember to enable the *Swap X-Y* checkbox.
8. Start the film exposure (or the exposure of the second CCD camera).

If the system does not guide well, then it's strongly suggested to watch the behaviour of the star, this will reveal where the problem is, since only 3 scenarios are possible:

- 1) Overcorrection: the scope is much faster than you measured, so every correction brings the star beyond the center. The solution is to increase the option "Telescope speed" or decrease the guide speed of the mount (if available), the best setting is 0.2 - 0.5 X sidereal, 1X is usually too fast.
- 2) Undercorrection: the guide is "lazy" and the star is brought to the center too slowly. This happens rarely, to solve it simply measure again the telescope speed.
- 3) Wrong direction: after a few seconds the star is brought away from the center. The direction of one of the axis must be inverted. Enable the option "Invert X" or "Invert Y".

To discover the problem quickly it's strongly recommended to try to guide on one axis at a time, this will help to understand which problem affects the RA and DEC axis.

**Tip:** The *Guide Window* can be used as a recorder to measure the dX and dY errors of the mount (drifts). The dX and dY errors can be written into a text file which can be copied into the Clipboard and opened later by the Notepad or the Text Editor of Astroart. Select the Statistics page and click the Menu button.

In Fig. 9 is reported a clear example of a mount with a serious drifting along the X axis: within only 12 seconds of recording the drift is of about 12 pixels while the Y axis moves randomly around the 0. You can plot the data to clearly see this effect: in this case the red line represent the X drift along the time. With good approximation the slope of the lines gives you the amount of the drift along the time (in this case about 1pixel/second along the X axis and 0 pixel/sec along the Y axis).

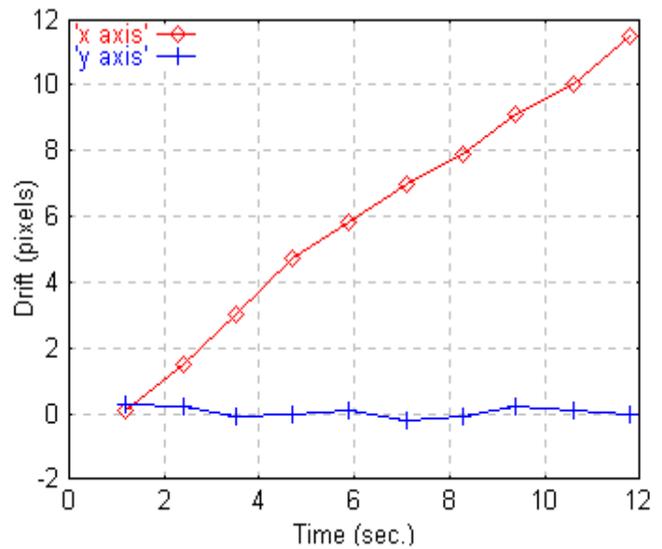
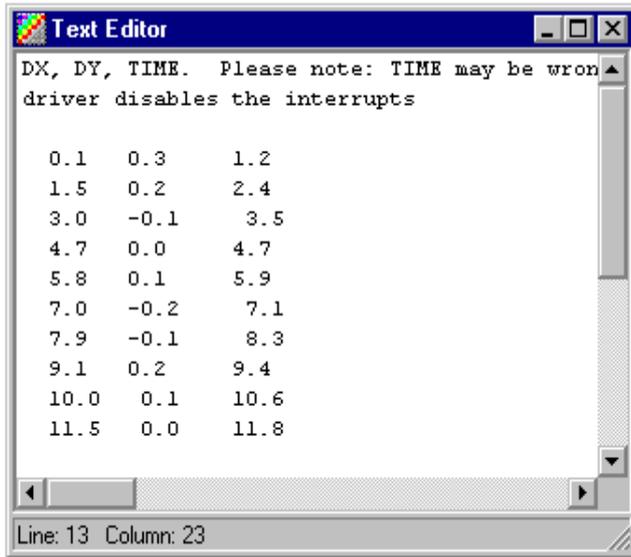


Fig. 9 – You can use the *Guide Window* as a recorder for mapping the guiding errors of your mount: in this simple example, within only 12 seconds of recording we can see a large drift along the X axis (about 1pixel/sec)

## 7.4 Selfguiding

Selfguiding means integrated imaging and autoguiding using a single CCD camera with a built-in autoguider (a second CCD sensor) like the SBIG™ ST cameras or a CCD which can download a part of its image without deleting all the pixels, like the Starlight-Xpress™ MX series.

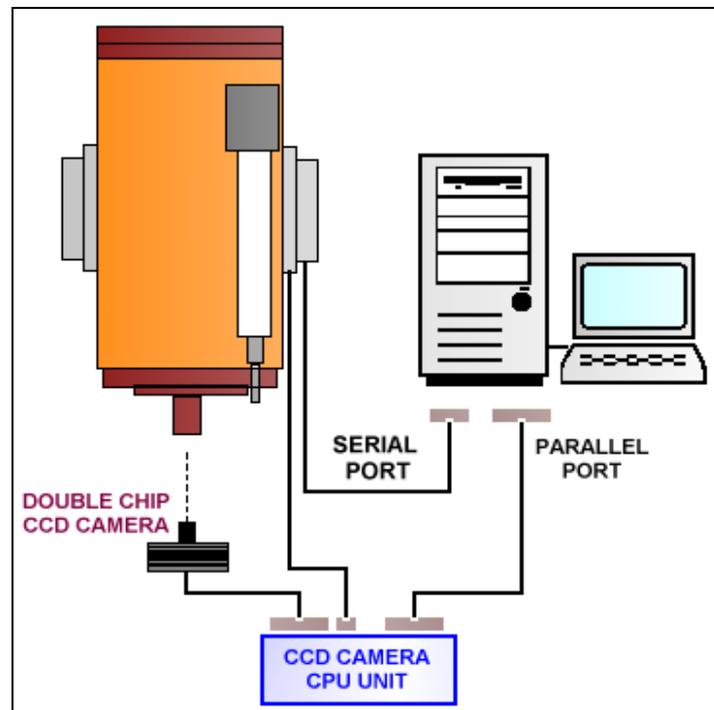


Fig. 10 - A typical telescope setup for autoguiding with a double chip CCD camera

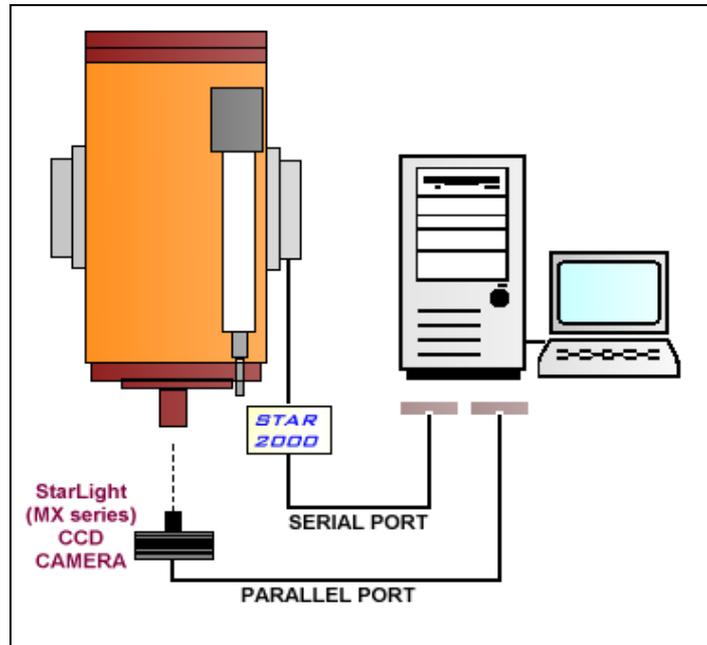


Fig. 11 - The Telescope Setup for Starlight-Xpress MX cameras

## 8 The Settings Page

This page contains all the options to download images from the CCD camera.

**Binning.** If 1x1 is selected then the CCD chip works at full resolution (example: 768x512 for a KAF400). If 2x2 is selected then four pixels will be grouped into one and the final image resolution is 384x256 (for a KAF 400). The advantage of binning is a faster download and better signal to noise ratio. 3x3 and 4x4 are useful and fast to find or center an object.

**Delay.** If this value is different from zero, before each exposure there will be a pause of N seconds. This is useful if you need to setup the telescope before each image.

**Sub-frame.** This a useful feature for planetary imaging at high resolution. To speed-up the download and save space on disk it is possible to acquire only a part of the CCD array. To use this feature download a full frame, select a rectangle on it, then click the **Select from...** button. The frame boundaries will be written into the four edit boxes, (as percentage). If needed it is also possible to input these boundaries by hand.

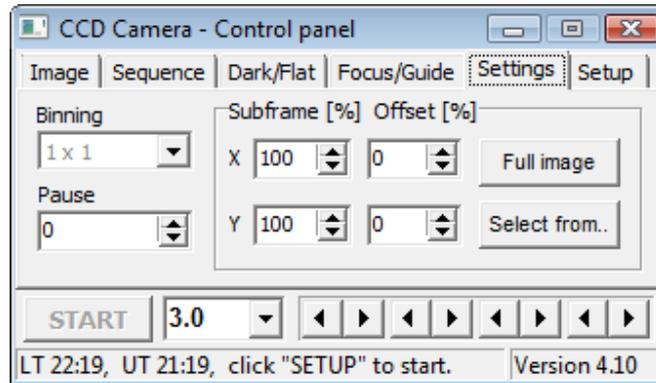


Fig. 12 – The Settings Page

## 9 Focusing

Before focusing you need to integrate a full image and select a rectangle around a bright star. Note that this star should not be saturated.

To integrate a full image with the main CCD camera click the **Start** button. To integrate a full image with the guider camera or the secondary camera click the **Focus** button

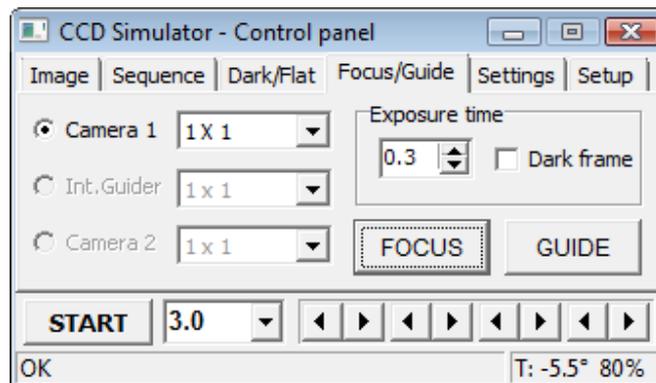


Fig. 13 – The Focus/Guide Page

**Exposure.** The exposure time for every frame (both for focusing and guiding).

**Binning.** The binning factor of every frame, see above.

**Correct Dark frame.** If selected, a dark frame is acquired at the beginning of the focus session: ( if you have not a shutter you should cover the scope before clicking the **Focus** button) Astroart will keep in memory the first frame as a dark frame and every subsequent image will be automatically corrected.

The focus window can be resized in real-time. In the Status Bar take a look at the sharpness indicator (Focus) of the current star: a high value means better focus.

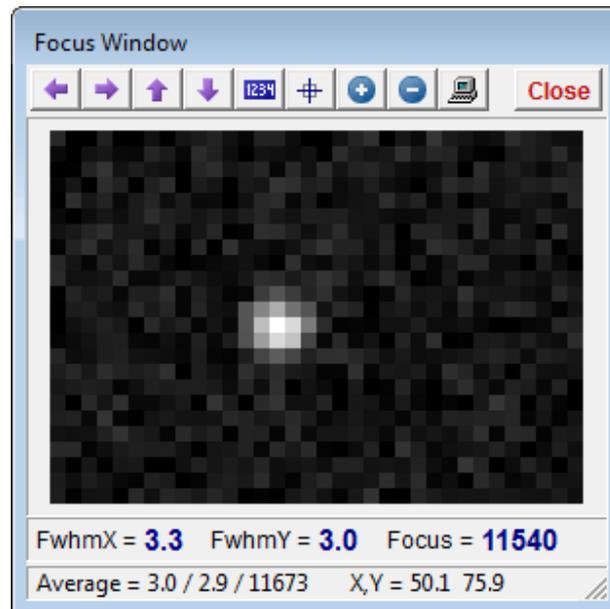


Fig. 14 – The Focus Window

**FwhmX, FwhmY.** “Full width at half maximum” is a measure of the size of the star (in pixel). These parameter are also useful to estimate the best focus, by the way the Focus value is usually more precise.

**X, Y.** The coordinates of the star relative to the full image.

**Buttons.** Click the arrow buttons to move the subframe across the CCD area. The “1234” button displays the Focus parameter (instant value above, average of 4 frames below) and enables the “talking” focus. The “cross” button displays a reference cross. The “+” and “-“ buttons change the exposure time.

**Autofocus.** Click the “PC” button to launch the ASCOM autofocus plugin, which is available at the Plugin Page of the Astroart web site.

**Tip:** the Focus Window can be used to center or find the object you are going to image: take a short exposure, select a rectangle as big as the whole image, go to the *Focus/Image Page* and select *4x4 binning* for a fast download and click on the **Focus** button.

## 10 The Dark/Flat Page

This Page manages the dark frames and flat fields.

**Select from desktop.** Click this button to select a dark frame/flat field from the Astroart desktop, this is useful if you use your own sets of dark frames previously acquired. Obviously the selected image must be of the same size of the CCD array.

**Dark frame correction.** Select this option to disable or enable the dark frame subtraction for every new image. If you change the binning factor the correction will be disabled.

**Flat field correction.** Select this option to disable or enable the flat field correction for every new image.

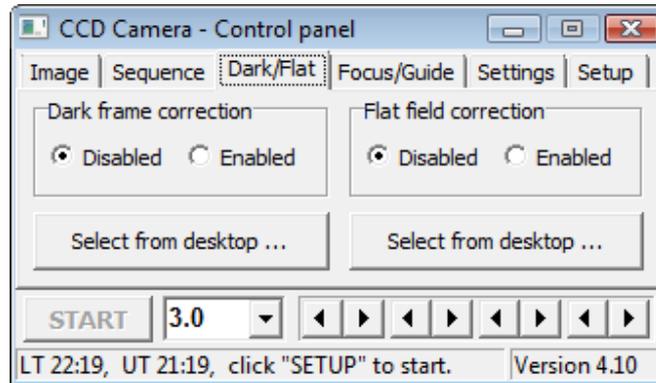


Fig. 15 – The Dark / Flat Page

For best results remember to take 5-10 dark frames and average them to reduce the random noise.

## 11 The Sequence Page

To improve the Signal-To-Noise ratio of CCD images it is often necessary to take a set (sequence) of pictures and average or sum them. This can be done automatically from the Sequence Page clicking the **Start** button.

**Images.** The number of exposures to be acquired, from 1 to 9999.

**Autosave.** If selected every image will be saved with the filename specified in the edit box plus a ordinal number, into the chosen directory. Every image will be also opened into the Astroart desktop, if there is no need for this, disable *New Window* in the *Image* page. The images will be downloaded with all the specifications given in binning, subframes, dark frame, etc.

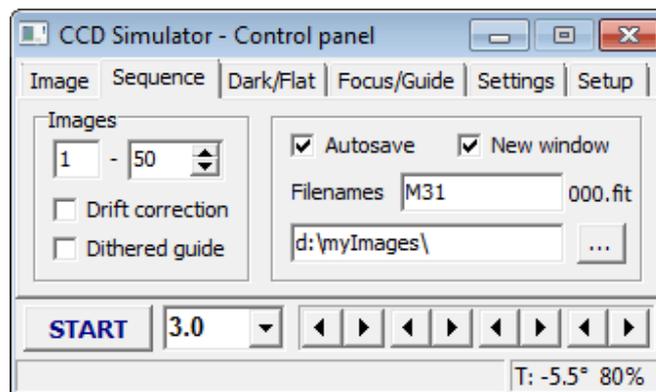


Fig. 16 – The Sequence Page

**Drift correction.** A useful feature for studying variable stars. After each image of the sequence the telescope will be moved to recenter the field. This allows long sequences (many hours) without the worry of drifts caused by bad polar alignment.

**Dithered guide.** This special option can be used during a sequence of autoguided images. Between each image the telescope will be move randomly by +- 0.5 pixels. This means that every image of the sequence will not be aligned with the next. For advanced users only.

**New window.** If selected all images will be displayed into a single window. This function is useful if you need to take and save hundreds of images.

To stop a sequence click the button **Stop** once, and wait a few seconds.

## 12 Scripts

A script is a list of commands which are executed in sequence. Using scripts it's possible to perform very complex tasks, like automatic supernova and asteroid search, programming the telescope and the CCD camera.

The script language of Astroart is called "ABasic". Its syntax is very similar to BASIC (GWBasic™, QuickBasic™, Visual Basic™, VBScript™, etc.)

Example: (test it with the CCD simulator)

```
Camera.Start(10)
Camera.Wait
Image.Save("C:\sample.fit")
```

The first command starts a 10 seconds exposure, the second command waits until the end of the exposure, the last command saves the image.

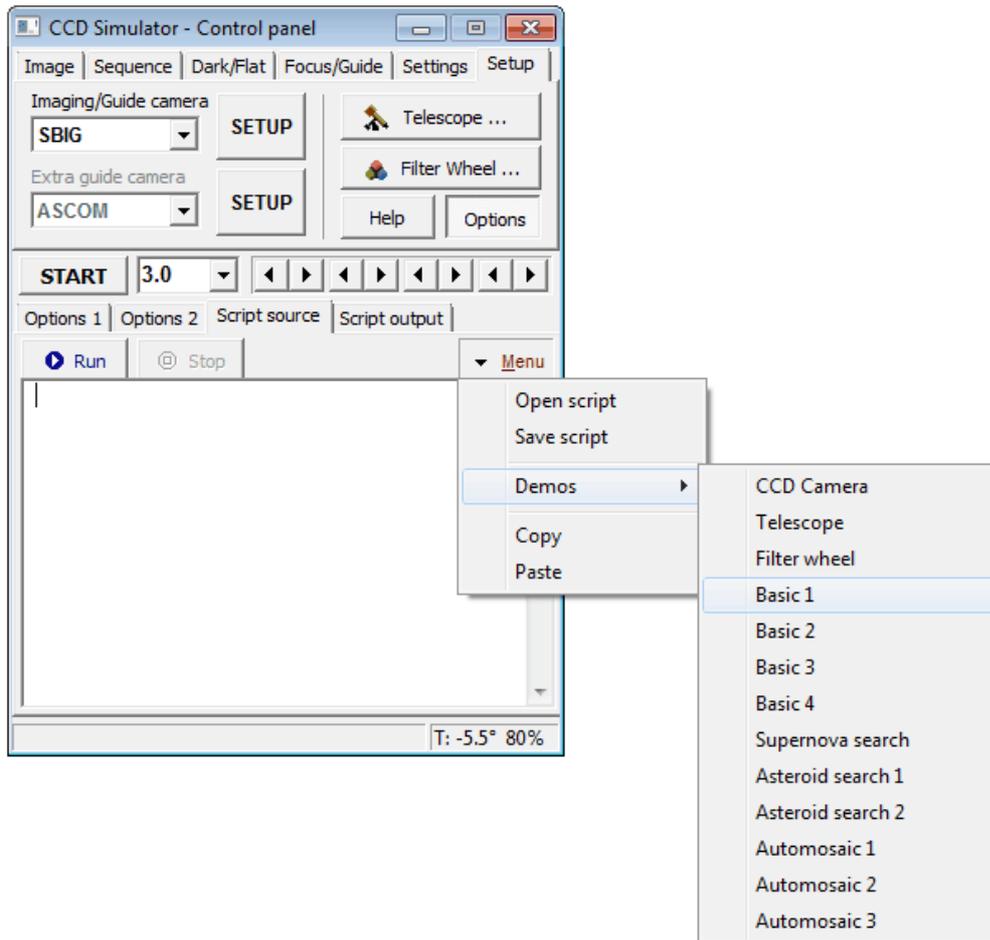
A further example: supernova search on 50 images.

```
for i = 1 to 50
  ra = Telescope.List.Ra(i)
  de = Telescope.List.Dec(i)
  name$ = Telescope.List.Name$(i)
  Telescope.Goto(ra,de)
  Telescope.Wait
  Camera.Start(60)
  Camera.Wait
  Image.Rename(name$ + ".fit")
next i
```

This simple script gets the coordinates and the names of the galaxies from the list in the Telescope Window. For every galaxy the script moves the telescope and starts the exposure.

### 12.1 Demos

To quickly learn the script features try all the demos available in the Menu. If required they will automatically connect the CCD, Telescope and Filter wheel Simulators. This allows to make practice indoor.



## 12.2 Types, Variables and Functions

ABasic supports two types: numbers and strings. A numeric variable contains a number, a string variable contains a string.

### Numeric variables.

They contain a number. The number is internally represented by a floating point value with double precision (64 bit, 15 digits).

Example:

```
x = 10.5
```

```
y = x + 1
```

### String variables.

A string variable contains text. This text may be a single row or a multi-line text. The maximum size of a string variable is 64 MByte.

Example:

```
a$ = "Hello"
```

```
b$ = a$ + "World"
```

The variable "b\$" now contains "HelloWorld"

A single character of a string can be read using square brackets: a\$[1] returns “H” and a\$[2] returns “e” and so on. If the index exceeds the length of the string then it restarts from the beginning, so a\$[6] returns “H” (a\$ was “Hello”).

A single row of a multi-line string can be read using curly brackets.  
Example, if a\$ contains:

```
"This is a  
text placed on  
three rows"
```

Then a\${2} returns “text placed on”. The function **count(a\$)** returns how many lines are contained in a multi-line string.

### Reserved words.

See the documentation available for VBScript, Visual Basic or any other BASIC compiler.

```
IF THEN ELSE ENDIF OR AND NOT MOD REM FOR NEXT STEP BREAK  
CONTINUE WHILE ENDWHILE GOTO GOSUB PRINT INPUT END CLS
```

### Numeric functions.

See the documentation available for VBScript, Visual Basic or any other BASIC compiler.

```
pi() sin(n) cos(n) tan(n) exp(n) ln(n) log10(n) log2(n) sqr(n)  
abs(n) rnd([n]) sgn(n) fix(n) int(n) round(n[,n]) frac(n) asin(n)  
acos(n) atan(n) atan2(n,n) sinh(n) cosh(n) tanh(n) asinh(n) acosh(n)  
atanh(n) degtorad(n) radtodeg(n) modulo(n,n) len(s) val(s) asc(s)  
pause(n)
```

### String functions.

See the demos for more information about these functions.

```
ucase$(s) lcase$(s) ltrim$(s) rtrim$(s) chr$(s) str$(n) mid$(s,n,n)  
hex$(n) left$(s,n) right$(s,n) ltab$(s,n) rtab$(s,n) format$(s,n)  
time$() date$() crlf$() opentext$(s) savetext$(s,s) copytext$(s)  
pastetext$() finddir$(s,s) findfile$(s,s) message(s) ra$(n) dec$(n)  
createdir()
```

### CCD and Telescope functions.

Function	Details	Example
Camera.Start(time[,shutter])	Starts an exposure of <time> seconds. Set <shutter> to zero to integrate a dark frame.	Camera.Start(60,0)

<b>Camera.Wait</b>	Waits until the end of the exposure.	
<b>Camera.Exposing</b>	Returns “1” if a exposure is in progress, otherwise “0”.	
<b>Camera.Binning(mode)</b>	Sets the binning mode. <mode> is a index to the binning list in the “Settings” page of the CCD panel.	Camera.Binning(2)
<b>Camera.SelectDarkFrame</b> <b>Camera.EnableDarkFrame(enable)</b>	Selects the current image as dark frame and automatically enables the correction for the following images.  Enables or disables the correction, 1 = enable, 0 = disable.	Camera.SelectDarkFrame()  Camera.EnableDarkFrame(0)
<b>Camera.Stop</b> <b>Guider.Stop , Guider.Close</b>	Stops the current exposures or the guiding session.	
<b>Guider.Select(n)</b>	Selects which CCD should be used for autoguide: 1 = main ccd, 2 = guide ccd, 3 = secondary camera.	Guider.Select(2)
<b>Guider.MoveReference([dx,dy])</b>	Changes the x and y coordinates of the reference star, to perform the “dithered guide”. If dx and dy are not specified then the shift will be pseudo-random.	Guider.MoveReference()  GuiderMoveReference(-0.3, 0.7)
<b>Camera.Connect([driver])</b> <b>Camera.Disconnect</b>	Connects or disconnects the CCD driver from Astroart.	Camera.Connect(“Simulator”)
<b>Camera.StartAutoguide([x,y])</b> <b>Camera.StopAutoguide()</b>	Starts and autoguide session. If x and y parameters (the coordinates of the guide star) are not given then this command takes a sample image and selects automatically the best star.	Camera.StartAutoguide()  x = Image.GetPointX() y = Image.GetPointY() Camera.StartAutoguide(x,y)
<b>Focuser.Connect</b> <b>Focuser.Disconnect</b>	Connects or disconnects an ASCOM autofocus. Requires the Autofocus plugin.	Focuser.Connect Focuser.Autofocus Focuser.Disconnect
<b>Focuser.Autofocus([x,y])</b>	Starts an autofocus session (requires the Ascom autofocus plugin). If x and y parameters (the coordinates of the focus star) are not given then this command selects automatically the best star from the current image.	Focuser.Autofocus  x = Image.GetPointX() y = Image.GetPointY() Focuser.Autofocus(x,y)
<b>Focuser.GotoRelative(n)</b>	Moves the focuser up or down by a specified amount.	Focuser.GotoRelative(-50)
<b>Focuser.GotoAbsolute(n)</b>	Move the focuser to a given coordinate.	Focuser.GotoAbsolute(1000)

<b>Telescope.Goto(ra,dec)</b>	Moves the telescope to the equatorial coordinates ra (0..24),dec. (-90..+90)	Telescope.Goto(23.4, 45.1)
<b>Telescope.Wait</b>	Waits until the telescope has completed a Goto.	
<b>Telescope.Stop</b>	Stops the telescope.	
<b>Telescope.Ra</b> <b>Telescope.Dec</b>	Returns the current position of the telescope.	x = Telescope.Ra y = Telescope.Dec
<b>Telescope.Pulse(dir\$ [,time])</b>	Moves the telescope for <time> seconds towards the <dir\$> direction ("N","S", "E","W"). If <time> is negative then the direction is inverted. If time is omitted, it moves until the Telescope.Stop command.	Telescope.Pulse("N", 0.5)
<b>Telescope.Speed(n)</b>	Sets the speed for Pulse motion. (1=guide, 2=center, 3=find, 4=slew)	Telescope.Speed(4)
<b>Telescope.List.Open(file\$)</b>	Opens a text file which contains objects and coordinates. See chapter 6.1.	Telescope.List.Open("c:\data\galaxies.txt")
<b>Telescope.List.Count</b> <b>Telescope.List.Clear</b>	Returns how many objects are listed in the Telescope Window.  Clears the list.	n = Telescope.List.Count
<b>Telescope.List.Ra(index)</b> <b>Telescope.List.Dec(index)</b>	Returns the coordinates of the <index>th object of the list. Returns -1 if index is not valid.	x = Telescope.List.Ra(25)
<b>Telescope.List.Name\$(index)</b>	Returns the name of the <index>th object of the list.	a\$ = Telescope.List.Name\$(42)
<b>Telescope.Send(string\$)</b> <b>Telescope.Receive\$</b>	Sends or receives a string to the telescope via the serial port.	Telescope.Send("#Hc#") R\$ = Telescope.Receive\$
<b>Wheel.Connect</b> <b>Wheel.Disconnect</b> <b>Wheel.Filters</b>	Connects, disconnect and returns the number of filters of the filter wheel.	Wheel.Connect n = Wheel.Filters Wheel.Disconnect
<b>Wheel.Goto(n)</b> <b>Wheel.Goto(\$)</b>	Moves the filter wheel to the given filter.	Wheel.Goto(4) Wheel.Goto("R")
<b>Image.Save(filename\$)</b>	Saves the current image.	Image.Save("C:\images\saturn.fit")

<code>Image.Rename(name\$)</code>	Renames the current image.	<code>Image.Rename("jupiter.fit")</code>
<code>Image.Open(filename\$)</code>	Opens an image from disk	<code>Image.Open("C:\moon.tif")</code>
<code>Image.GetKey\$(key\$)</code> <code>Image.GetKey(key\$)</code> <code>Image.SetKey(key\$,value)</code>	Reads or Writes values from the FITS header.	<code>a =Image.GetKey("NAXIS")</code> <code>Image.SetKey("COMMENT","Bad seeing")</code> <code>Image.SetKey("JD",34234)</code>
<code>Image.FlipH</code> <code>Image.FlipV</code> <code>Image.Resize(x,y)</code>	These functions modify the current image.	<code>Image.Resize(320,240)</code>
<code>Image.BlinkAlign</code>	Aligns the current image with the next one inside the Astroart Desktop and blinks them. Requires the Service Pack1.	<code>Image.BlinkAlign</code>
<code>Image.Close</code>	Closes the current image.	
<code>Image.GetPointX()</code> <code>Image.GetPointY()</code>	Return the coordinate of the selected point (or star, or rectangle) on the current image.	<code>x = Image.GetPointX()</code>
<code>Image.DSS(ra,dec,name\$)</code>	Creates a new image from the Digital Sky Survey. Needs the DSS plugin.	<code>Image.DSS(12.034,45.213,"a steroid.fit")</code>
<code>Output.Save(filename\$)</code> <code>Output.Copy</code>	Saves the output panel to disk. Copies the output panel to the Clipboard.	<code>Output.Save("C:\Log.txt")</code>
<code>System.Execute(filename\$)</code>	Executes a new program.	<code>System.Execute("C:\Windows\notepad.exe myfile.txt")</code>
<code>System.Broadcast(message\$,wparam,lparam)</code>	Sends a Windows Message to all windows. This can be used to control other programs. The function is equivalent to:  <code>h = RegisterWindowMessage(message\$)</code> <code>SendMessage(HWND_BROADCAST,h,wparam,lparam)</code>	

## 12.3 Loop instructions

ABasic supports two loop instructions: **For** and **While**.

The easiest way to understand a loop is to look at an example.

The following program prints the numbers from 1 to 10, "a" is the control variable:

```
for a = 1 to 10
  print a
```

```
next a
```

### **The BREAK instruction.**

Exits from a loop. In this following example the loop stops when "a" becomes greater than 5.

```
for a = 1 to 10
  print a
  if a>5 then break
next a
```

### **The CONTINUE instruction.**

Acts as a NEXT instruction. A new iteration starts immediately.

```
for a = 1 to 10
  print a
  if a > 5 then continue
  print "Test"
next a
```

### **FOR-NEXT instruction.**

The complete syntax for this command is:

```
FOR <variable> = <expression> TO <expression> [STEP <constant>]
...
...
NEXT <variable>
```

Examples:

```
for angle = 1+asin(0.4) to 1+asin(0.75) step 0.1
  print angle
next angle
```

```
s = 0
for y = 1 to 10
  for x = 1 to 20
    z = x*y : print z : s = s+z
  next x
next y
print s
```

### **WHILE-ENDWHILE instruction.**

This instruction evaluates a condition at the beginning of the loop. If the condition is false then the cycle stops and execution continues after the ENDWHILE instruction.

Example:

```
a = 1
while a <= 10
  print a
  a = a+1
```

**endwhile**

Since the While command evaluates the condition at the beginning of the loop, the instructions inside the loop can be also never executed. The *Break* and *Continue* instructions can be used in WHILE-ENDWHILE cycles just like in the FOR-NEXT cycles.

## 12.4 Conditional instructions

The IF-THEN instructions evaluates a logical expression and determines the flow of the program based on the result of that expression.

Example of logical expression:

```
a > 5 and b$ = ".fits"
```

```
a >= 3 or not (b <> 5 and b+3 = c)
```

Precedence of operators:

*Higher precedence.*

()

< > <= >= <> =

NOT

AND

OR

*Lower precedence.*

### Syntax.

```
IF <logical expression> THEN
...
...
[ELSE]
...
...
ENDIF
```

### Compact Syntax.

```
IF <logical expression> THEN <instructions> ELSE <instructions>
```

IF instructions can be nested without limits. The ELSE part is optional.

Example:

```
for a = 1 to 10
  if a < 6 then print "-" else print "+"
next a
```

Example:

```
for a = 1 to 10
  if a < 6 then
```

```
print "-"
if a = 5 then print "Half work"
else
print "+"
if a = 10 then print "The end"
endif
next a
```

## 12.5 Other functions

---

### PRINT function.

Prints a text on the “Script output” panel of the CCD window.

```
PRINT [ expression [, expression] [; expression] ]
```

Every expression must be separated by a semicolon or comma; if semicolon is used then a space is added between the writings. If comma is used then a TAB separator is added between the writings.

Example:

```
print "Some math" ; 3+5 ; 4*4*4 ; sin(2.5)
```

### INPUT function.

Shows a dialog window where the user can enter data.

```
INPUT [ <Question> , ] variable
```

Use this function to ask the user for input values. If the user doesn't click the OK button then zero or an empty string is returned.

Example:

```
input "How old are you ?", a
input "Your name ?", n$
print n$ ; "is"; a
```

### MESSAGE function.

Shows a dialog window and wait until the user press “OK”.

Example:

```
message("Ready to start")
```

### COMMENTS.

To write a comment into a script use the keyword “REM” or the symbol “ ‘ ”, example:

```
' This is a comment
REM This is a comment
```

## 12.6 Automatic research

---

A script for automatic research controls the telescope, CCD camera and filter wheel. Usually it consists of three tasks.

1. Open a list of objects(example: Variable stars, Galaxies, etc.)
2. Setup a loop.
3. For every cycle, move the telescope, select the filter, take an image and save it.

For step (1) two methods are possible:

1. The function “Telescope.List.Open(filename\$)” which loads a compatible list (see chapter 6.1). In the script use the commands “Telescope.List..”
2. The function OpenText\$(filename\$) which loads any text file into a string variable. This multi-line string can be parsed using the operators “{}” or the functions “Mid\$”, “Val”, etc.

Example, using lists: (The object list was opened by hand).

```
n = Telescope.List.Count
for i = 1 to n
  ra = Telescope.List.Ra(i)
  de = Telescope.List.Dec(i)
  name$ = Telescope.List.Name$(i)
  print n; name$, ra; de
  Telescope.Goto(ra,de)
  Telescope.Wait
  Pause(4)
  Camera.Start(120)
  Camera.Wait
  Image.Save("c:\images\2004\" + name$ + ".fit")
  Image.Close
next i
```

See all the demo scripts inside the CCD Interface for more information.

## 12.7 Automatic align and blink

---

The function “blinkalign” (available since Astroart 4.0 Service Pack 1) automatically aligns two images and blinks them, to help you in comparing the images. This is useful for search of asteroids, supernovas and comets.

Example: open one image from a recent session, let’s say “M67.fit”, then execute the following script:

```
r$ = "D:\Astroimages\OldReferences\"
Image.Open(r$ + Image.FileName$)
Image.BlinkAlign
Image.Close
Image.Close
```

Where r\$ is a folder of old images of the same field. Your reference image will be opened, compared to the new one, then both images will be closed. To compare an other pair of images, open the next one from the recent folder and execute the script again.

The whole procedure could be made fully automatic iterating through all images of the recent folder, here is a sample script:

```
n$ = "D:\Astroimages\NewImages\  
r$ = "D:\Astroimages\OldReferences\  
im$ = FindFile$(n$, "*.fit")  
n = Count(im$)  
for i = 1 to n  
    Image.Open(n$ + im${i})  
    Image.Open(r$ + im${i})  
    Image.BlinkAlign  
    Image.Close  
    Image.Close  
Next i
```

## 13 History

2005/05/27 - V 3.50. Remote Control, binning for guiding CCDs, minor fixes.  
2005/06/24 - V 3.51. Starlight-Xpress Adaptive Optics now supported.  
2005/09/16 - V 3.60. Guider.MoveReference command, new focus functions.  
2005/11/25 - V 3.61. Shoestring USB, SBIG CFW10 serial, improved remote control.  
2005/12/06 - V 3.62. Shoestring Parallel port, improved remote control.  
2006/04/18 - V 3.70. New autoguide features, new script commands.  
2006/07/21 - V 3.71. Sequence limit increased to 9999 images, fixed two minor bugs.  
2006/11/18 - V 3.72. Relay support for custom cameras.  
2007/05/02 - V 3.80. Autofocus and focusers support.  
2007/09/19 - V 4.00. New script functions.  
2008/02/02 - V 4.10. Up to three cameras can be controlled at once.  
2008/06/16 - V 4.20. SX AO “bump” feature, Meade DSI corrected as secondary camera.  
2008/11/11 - V 4.30. New AO functions, talking focus, Ascom filter wheels.  
2009/02/03 - V 4.40. Script commands for autoguide and autofocus. Minor fixes for Ascom and CGE.  
2009/04/30 - V 4.50. Starlight-Xpress filter wheel supported. Script commands for dark frames.  
2009/09/10 - V 4.51. Script commands for focusers, improved dithered guide, FLI filter wheels.  
2010/08/12 - V 4.60. Compatible with Sequence Generator, new autoguide calibration, scripts for autofocus.  
2011/02/05 - V 4.61. OBJECTRA/DEC FITS keywords, Telescope.Receive\$ script command, Starlight-Xpress filter wheel USB supported.

## 14 Contact Information

**MSB Software.**  
Via Goetz 93, Classe  
48124 Ravenna RA - Italy  
Tel/Fax +39 0544 527265  
Dr. Fabio Cavicchio +39 339 2739548  
Dr. Martino Nicolini +39 349 4004784  
WEB: <http://www.msb-astroart.com>  
WEB: <http://www.msbsoftware.it>  
E-mail: [info@msbsoftware.it](mailto:info@msbsoftware.it)