ATC02 – Software and operational manual

rev. 03 - 20160331 - author: Alessio Galliazzo

Table of contents

Tables

Images

1. Description

1. Software requirements

CPU:	Any intel x86 or x64 compatible CPU with 1.8GHz of clock
O.S.:	Windows Vista or newer 32 or 64 bit
Hardware interfaces:	A serial port or an USB to serial converter or Ethernet to serial converter ASCOM Platform v6.2

Table 1: Hardware/Software requirements

2. Software architecture and basis

The new version of ATC Remote software is based on a client/server architecture.

The server part is used to manage all the data that need to be sent/received to the ATC02 board.

The server opens the serial connection between ATC02 and other software running on the PC.

The server also manages the timing sequences that poll data from ATC02 unit and, if anything change in the stored variables values, it sends data to clients.

This part of software is also used to "share" a port between many applications, so in the same time you could connect with ATC Remote GUI, with some custom developed software and also with one (or more) ASCOM driver connection. This could be useful using software like MaximDL or FocusMax at the same time.

The client side is used by ASCOM platform driver and our GUI. This part is only used as user interface to read and command the ATC02.

Every time the ATC GUI or ASCOM Driver is launched check for server running, if the client application cannot find a suitable server the application automatically starts the server.

The server is never closed by client application so if you need to switch between application closing one and opening another one the COM port is not reopened.

All the applications are written in C# and are based on .NET 4.5 Framework.

2. ATC Library server

As told in software description the role of the server in the software architecture is to interface the real world with the virtual one.

The server manages the most part of the work cause it is responsible to poll the ATC02 periodically to get variations in parameters, to interpret the ATC02 serial protocol and translate it to a more flexible network protocol, to store data and check differences in the values and to send the right command to ATC02 to permit the control of your telescope.

Since version 3.x the software was completely rewritten and the new software is capable to enqueue commands that will be executed when the ATC02 is free to proceed so, for example, if you move

the BFL to the 0 position and - while the focus is moving - you set the fan speed to 50%, the speed of fan will vary just when the focuser will stop.

This is a big enhancement cause permit the user to launch a command and forget about it.

Another advantage of having a server process running is that you can connect to the telescope from a remote PC as a “native” client without using software like VNC or other desktop remotization program.

1. Network interface

At the start of server program it opens a listening TCP socket on all local addresses on port 4005 waiting for a connection.

2. Network protocol basis

We choose to use a human readable network protocol to help custom developers to implement themselves software and better debug it.

The protocol is based on sentences starting with dollar ('$') and ending with a newline ('\n') characters. All the sentences are sent as characters and can be read with a standard telnet client. If some parameters are needed after the command a bar (“|”) is used as separator.

Supported command list are quite small cause that commands are only used to incapsulate data. Here are the implemented commands:

$CONNECT\n	Open the serial connection between the server and the ATC02 board
$DISCONNECT\n	Close the serial connection if opened. After this command is sent the server store data to file to be able to do a fast start on the next connection
$ISCONNECTED\n	Get the current serial connection status
$SETPORT|comport\n	Set the com port that is used for the connection
$GETPORTS\n	Get a list of available ports in the host system.
$GET|ParameterName\n	Get the value of the specified parameter
$SET|Param|Value\n	Set the value of the specified parameter
$OPENSHUTT\n	Open the primary mirror protection shutter
$CLOSESHUTT\n	Close the primary mirror protection shutter
$FINDOPTIMA\n	Move the focuser in its optimal position
$READSOB\n	Read the optimal shift of focuser (this command should be used only by Officina Stellare staff to tune the backfocus position)

Table 2: Client to server commands

Every time that server process a command successfully it will reply with the proper response so, for example, if a $CONNECT command is issued then the server will reply to all the clients with a $CONNECTED response.

The response has the same format of commands. It's strongly suggested to the developers that want

to interface their own software with our server to implement all kind of responses cause these are sent to every connected client also if the client doesn't ask for something.

Here are a list of responses:

$CONNECTED\n	This response indicate that the server is connected to the ATC02 unit and everything is working as expected. This response is sent after a $CONNECT command or after a $ISCONNECTED command.
$DISCONNECTED\n	This response indicate that the server close its communication with the ATC02 unit. The com port is now available to other software and all data collected from the last poll cycle from ATC02 is now stored on a file. This command is sent after a $DISCONNECT command or after a $ISCONNECTED command.
$ERROR| ErrorDescription	This response indicate that something is going wrong with the server. A short description of the command is reported as parameter. More detailed informations will be found in the server log files.
$GET|Parameter|Value	This is the response to a $GET request. This response return the value of the specified parameter. It is also sent when, during the poll cycle, the server notice that a parameter change its value.
$OK	This is the response to a $SET parameter. When the ATC02 understand and execute the SET command, this response is sent.
$COM|comport	This is an item of the list of available port in your system.

Table 3: Response from server to clients

a) Parameters

As told in previous paragraph the $SET and $GET command need a specific payload as parameter. Here you can find a list of these parameters:

General parameters:

Parameter (case sensitive)	Type	Description
FanSpeed	Int (0-100)	Current fan speed in percentage
ShutterStatus	Enum Read only	Current shutter status, accepted values are Opened (0), Closed (1), Moving (2), Error (3)
isDisplayOff	boolean	A flag indicating that the ATD display unit is enabled or disabled. True means that the display is disabled, False means that display is enabled. When disabling the ATD form the software both display and pushbutton are ignored by firmware
IsShutterAvailable	Boolean Read Only	Flag indicating if the primary mirror protection shutters is installed on the system
FirmwareVersion	Float Read Only	Current installed firmware ATC02 version
isCelsius	Boolean Read Only	True if the degrees returned from temperature probes are expressed in Celsius, False if Fahrenheit
UserInformation	String Read Only	A string containing customer information
UserInformation2	String Read Only	Same as above
TelescopeModel	Int(1-30) Read Only	ID For the telescope model
AmbientTemperature	Float Read Only	Current environment temperature
AmbientPressure	Float Read Only	Current environment pressure
AmbientHumidity	Float Read Only	Current environment humidity (RH %)
DewPoint	Float Read Only	Current calculated dew-point
TiltAngle	Int(-90 - 90) Read Only	The current tilt angle for the telescope
RollAngle	Int(-90 - 90) Read Only	The current roll angle for the telescope

Table 4: General paraemters

Primary mirror related parameters:

Parameter (case sensitive)	Type	Description
PrimaryMirrorHeaterMaintainStaticTemperature	boolean	Status of the fixed point temperature control for M1, true if enabled, false if disabled
PrimaryMirrorStaticTemperature	float	Set point for fixed point temperature control for M1
PrimaryMirrorDegreesOverDP	float	Set point for following dew-point temperature control mode for M1 (if set to 0 the control is disabled)
PrimaryMirrorPWMValue	Int (1-10)	Current value for M1 heater PWM (1 means 10%, 10 means 100%)
PrimaryMirrorTemperature	Float Read only	Current mirror temperature
PrimaryMirrorHeaterMaintainDegreesOverDP	Boolean Read only	Flag indicating if the dew-point following temperature control is enabled or not
PrimaryHeaterActive	Boolean Read only	Flag indicating the current state of the heater: true the heater is ON running with PWM, false heater is OFF

Table 5: Primary mirror parameters

Secondary mirror related parameters:

Parameter (case sensitive)	Type	Description
SecondaryMirrorHeaterMaintainStaticTemperature	boolean	Status of the fixed point temperature control for M2, true if enabled, false if disabled
SecondaryMirrorStaticTemperature	float	Set point for fixed point temperature control for M2
SecondaryMirrorDegreesOverDP	float	Set point for following dew-point temperature control mode for M2 (if set to 0 the control is disabled)
SecondaryMirrorPWMValue	Int (1-10)	Current value for M1 heater PWM (1 means 10%, 10 means 100%)
SecondaryMirrorTemperature	Float Read only	Current mirror temperature
SecondaryMirrorHeaterMaintainDegreesOverDP	Boolean Read only	Flag indicating if the dew-point following temperature control is enabled or not
SecondaryHeaterActive	Boolean Read only	Flag indicating the current state of the heater: true the heater is ON running with PWM, false heater is OFF

Table 6: Secondary mirror parameters

Secondary mirror focuser parameters:

Parameter (case sensitive)	Type	Description
BFL	float	The current BFL position
BacklashCompensationEnabled	boolean	Current status for the backlash compensation feature for secondary mirror linear actuator (M2 Focuser), true for backlash compensation enable, false for backlash compensation disabled
BacklashCompensationDirectionIn	boolean	A flag that identify the direction for the backlash compensation, true for inward movements, false for outwards movements
BacklashCompensationValue	Int (0-99)	Value of the amount for the backlash compensation expressed in cent of millimeter on the focal plane
OptimalBFLPosition	float	Optimal BFL position for the instrument.
BFLMaximumDelta	Int (0-9999) Read only	Value for the maximum delta for M2 movement expressed in cent of millimeters. Maximum value for the BFL is *OptimalBFL+BFLMaximumDelta*, minimum value is *OptimalBFL-BFLMaximumDelta*
IsEncoderAvailable	Boolean Read only	True if the secondary mirror is driven by a servomotor, false for normal – feedback less – linear motor
EncoderFirmwareVersion	String Read only	The servomotor firmware version
isMoving	Boolean Read Only	True if M2 is moving, false if is stopped at the target position

Table 7: Secondary mirror focuser parameters

3. Runtime files

The library server uses a couple of files that are self generated. All the files are located in the AppData\Roaming user folder under the Officina Stellare subfolder. In this directory you will also find all log files.

The interesting files are:

- Parameters.xml:

This files stores all the information of your telescope. Every time that the $DISCONNECT command is sent by the network interface this file is updated with the latest working parameters of your ATC.

We choose to use a standard XML file to better fit our customer requirements (it can be read with a lot of different software and there are a lot of library that can parse this file).

In customer's interfaces there are no needs to directly access this file cause, when starting the ATC library server, all the parameters are automatically loaded in the working memory and used as a base to every further update.

The file is based on a main tag named “Parameters” and a set of child tags named Property or Field.

Both Property and Field tag has an attribute called “name” use to specify what parameter the tag is referred to. Inside the tag there is a string representation of the parameter value.

So, for example, this is a common parameters.xml file:

```
<Parameters>
        <Property name="BFL">207,00</Property>
        <Property name="PrimaryMirrorTemperature">21,5</Property>
        ...
        ...
        <Field name="ShutterStatus">2</Field>
</Parameters>
```

- process.txt

  This file simply contains the fully executable path for the library server. This file is used as a reference for clients to know how to start the server. This file is created at installation time and it will never be deleted. Deleting this file broke the ability for client to automatically start the server when needed. To regenerate this file is enough to start the server manually.

4. Logging files

The server process has his own logging system that can be used to debug the current status of the server and better understand every problem that will be encountered.

The logger module is an HTML file writer that store log information in a table containing the timestamp of the logged event, the module that produce that event the severity of the event (starting from LOG_VERBOSE to LOG_ERROR) and a log message that explain the event.

Different log files is produced by the ATC library server to log different processes:

- ATCLibraryServer_yyyymmddhhmmss.html

  This file log is taken from the ATC Library Server class and contains the communication log between ATC02 and PC

- ATCLibServerProgram_yyyymmddhhmmss.html

  This file contains the main log for the process. Here you can find startup errors, network protocol comunication errors and basic functionality messages.

- NetworkInterfaceServer_yyyymmddhhmmss.html

  This file contains the log for the listener thread of the network interface. It's usually quite small and doesn't tell you so much about software running parameters.

3. ATC Remote GUI

This is the ATC Remote GUI main window. When you first start the application this is what you will see.

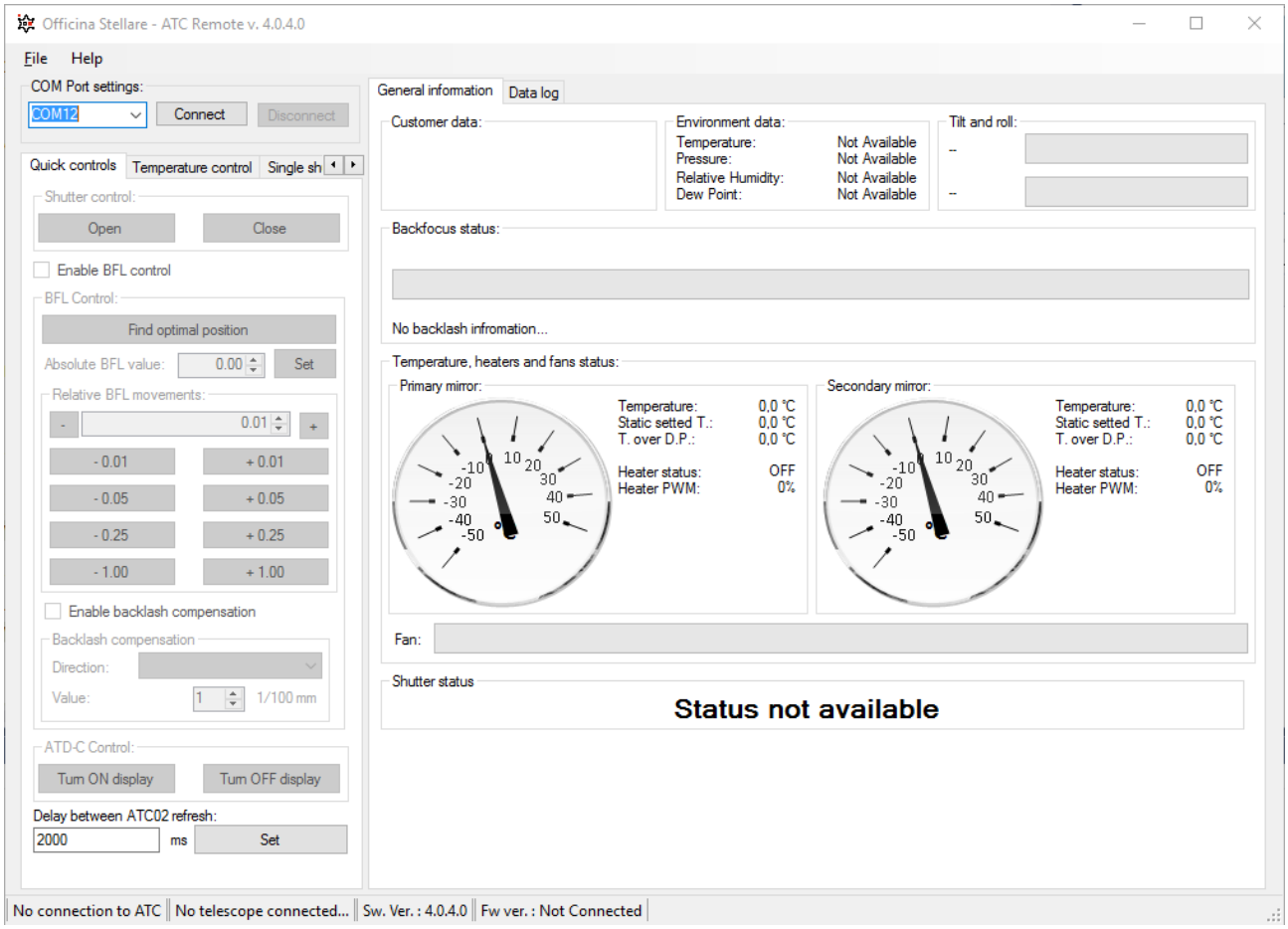


Fig. 1: ATC Remote main window

1. *Graphical user interface*

The main software GUI is composed by 4 different parts.

On the top side (red part on Fig. 2) you have a menu bar: a common way to interact with user and to do the very basic software functions like change settings, showing the about information, and exiting the application.

On the left side (green part on Fig. 2) you have the user control interface. With this part of the GUI you can drive your telescope changing the working

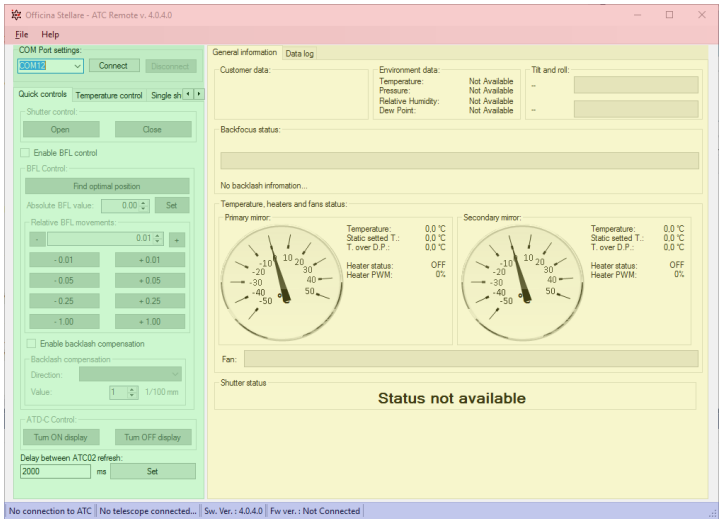


Fig. 2: Main window - Colored

parameters like fans, heaters, BFL, etc.

On the right side (yellow part on Fig. 2) you have all the feedback from the actual telescope situation. You can see the current BFL value, environmental data, M1 and M2 information, fans and shutters status. This part is also splitted in 2 different tabs, one for the actual situation and one used to display graph about the data collected during the software runtime.

On the bottom of the window (blue part on Fig. 2) you can see some useful information like the connection status, the telescope type, the software and firmware version.

a) The menu bar

On the top part of the window you can see a menu bar.

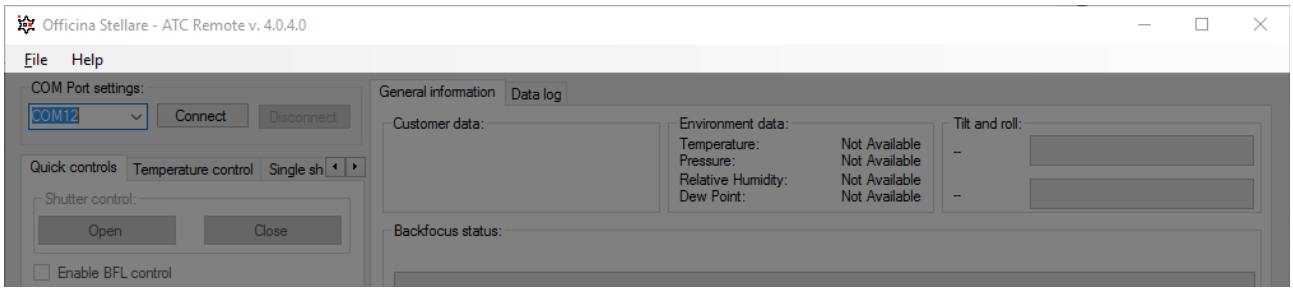
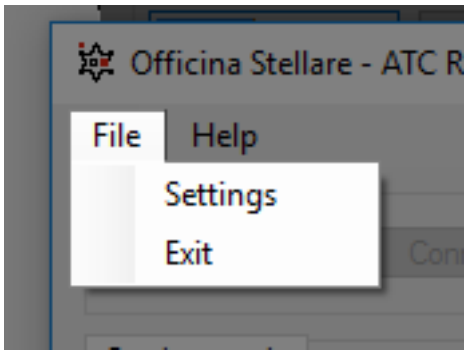


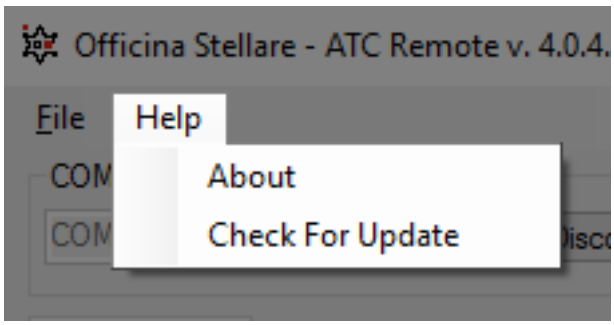
Fig. 3: Top menu bar



File menu is used to change settings and to exit the application.

You can also exit the application by clicking on the top right "X" that you find in the title bar.

Fig. 4: File menu



The Help menu show you the "about" information for the application and has the option to check for software update.

Fig. 5: Help menu

b) Telescope control:

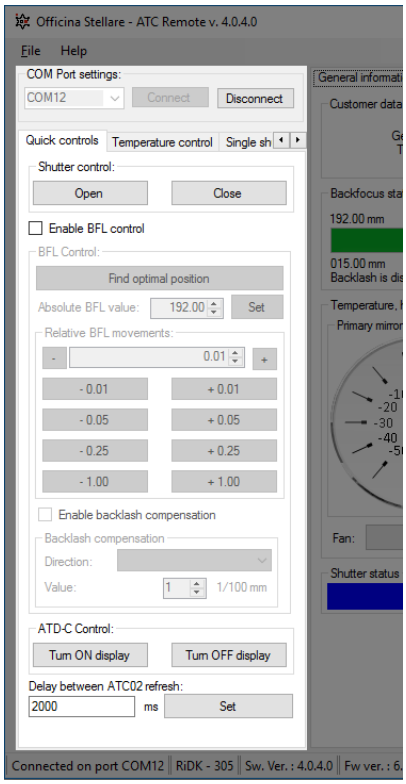


Fig. 6: Telescope control

The main telescope control part of the GUI is used to connect and send command to the ATC02 unit on the telescope.

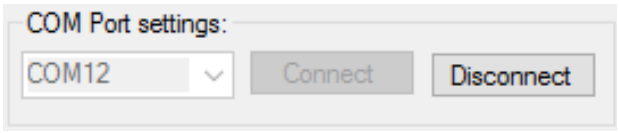


Fig. 7: Com port selector

The COM port is selected with the combobox that you can see in fig. 7.

At startup ATC Remote software will populate the list of available serial port connected on the PC. The last used serial port is pre-setted at the software startup.

Warning: if you are using an USB To Serial adapter ensure to connect the adapter on the USB port before turn on both ATC Remote GUI and ATC Library Server: As the ports are automatically populated at startup, if you connect the USB adapter after starting ATC Remote you need to close and reopen all the software.

Connect and Disconnect button is used to connect to or disconnect from the ATC02 unit. These pushbuttons are grayed according to the connection status to avoid double erroneous connection commands.

Quick control:

Quick control window allow you to make the first operation on the telescope. From this window you can:

- Open/Close the primary mirror protection shutters
- Enable the BFL control
- Homing and move the secondary mirror focuser
- turn on/off the ATD controller
- Change the polling interval between ATC02 queries

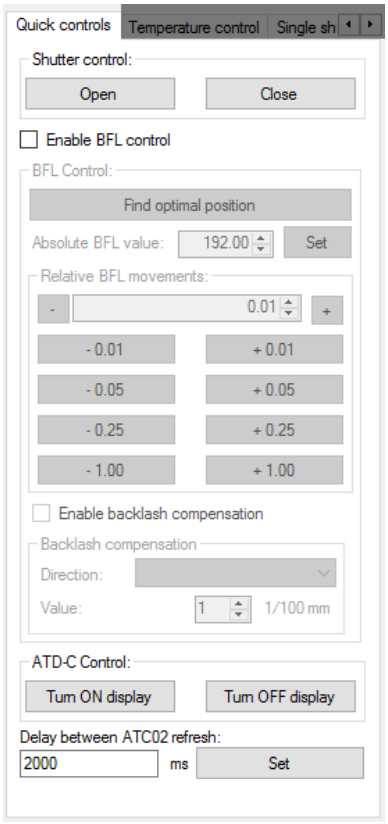


Fig. 8: Quick control

Shutter control:

Fig. 9: Shutter control

This is quite self explanatory: the open button will open the primary mirror protection shutters, the close button will close them.

BFL Control:

The BFL is the backfocus length of the instrument. If you ordered an instrument with secondary mirror focuser system, acting on this part of software, you can control the linear actuator that is moving the secondary mirror.

The **Find optimal position** button is used to start an homing procedure, fully controlled by ATC02 unit. Clicking on this button the ATC will move the focus inward until it reach the homing sensor and then return to the factory setted optimal focus position.

Warning: Is strongly suggested to proceed to a Find optimal position almost once every couple of night. This ensure a fix starting point for the focusing operation.

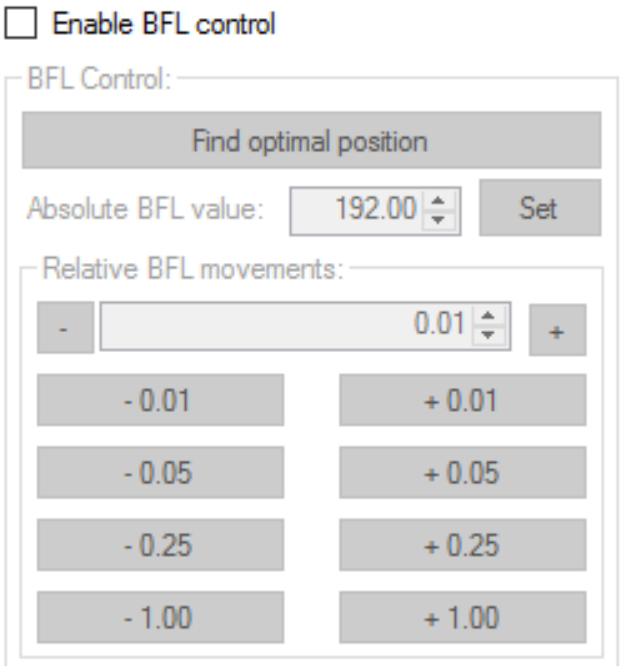


Fig. 10: BFL Control

Absolute BFL value allows you to move the BFL to a specific position. If, for example, you see that your stars are focused at 200.23mm of BFL you can tell ATC to position the BFL in this precise position without auto-focusing mechanism or any kind of calculation.

Relative BFL movements allow you to change the BFL starting from the current position. It can be useful if you are quite near the focus and you need small corrections moving the BFL through small steps. You can choose a custom value using the numeric selector and the "+" and "-" buttons or use the standard steps buttons just under the numeric selector.

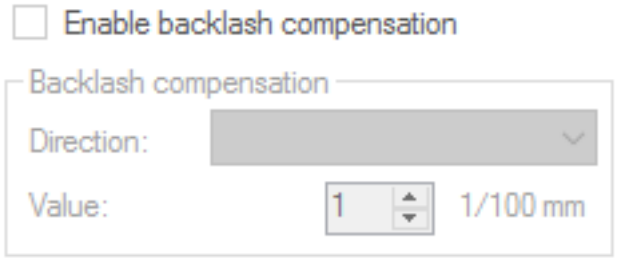


Fig. 11: Backlash compensation

The **backlash compensation** is used to compensate the error caused by an inversion of movement of the secondary mirror linear actuator. This particular function is usually not necessary cause we choose to use high precision backlash compensated linear actuator.

Enabling this feature allow you to choose the direction in which you intend to apply the compensation (INward or OUTward) and the amount (expressed in 1/100 of mm) of the compensation.

ATD-C Control:

This part is used to turn on/off the ATD. If you purchased the handpad with your telescope you may want to disable it.

At every ATC02 startup the ATD is turned on again by default.

Fig. 12: ATD Control

Delay between ATC02 refresh:

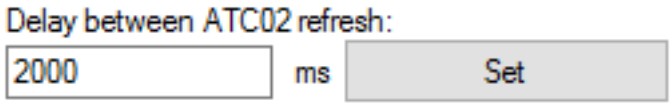


Fig. 13: ATC Delay settings

The **Delay between ATC02 refresh** express how many often the ATC02 unit is polled for its values. Lower delay produce faster response time but it can also flood ATC command queue causing longer delays.

We strongly suggest to never go under 1500 ms. 2000 ms is a good choice for normal astronomical application.

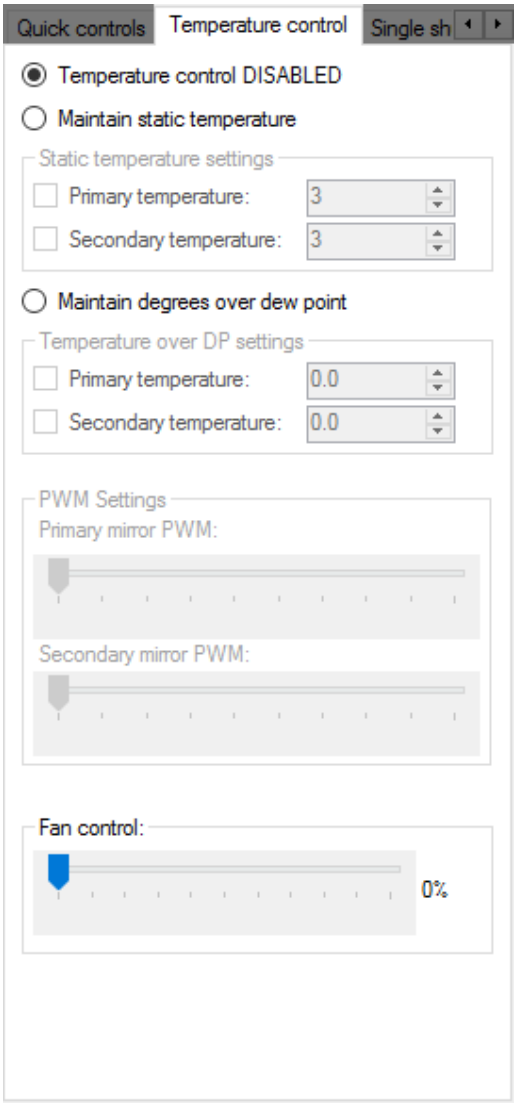


Fig. 14: Temperature control

Temperature control:

All Officina Stellare's telescopes has the ability to manage the mirror temperature to avoid the condensation of dew on the optical surfaces.

There are 2 ways to control the mirror temperature:

- Static temperature mode
- Follow dew point mode

Static temperature:

In this mode of operation the primary mirror and secondary mirror is maintained at a stable temperature chosen by the user.

This operation mode is selected by checking the **"Maintain static temperature"** radio button, checking the **"Primary temperature"** and/or **"Secondary temperature"** checkbox and setting each temperature to a specific value.

Follow dew point:

In this mode the mirror temperature is constantly maintained few degrees over the actual calculated dew point.

To enable this mode of operation you have to check the **"Maintain degrees over dew point"** radio button, check **"Primary temperature"** and/or **"Secondary temperature"** and set an amount of degrees over dew point to maintain. There is an upper limit of delta T allowed of 9.9 degrees.

Mirror temperature and telescope operation consideration:

Is **strongly suggested** to **turn off** (default condition) **all the heater** while using the telescope if not necessary. The scope of the mirror heater is only to avoid the formation of dew on the mirrors optical surfaces so, if the dew point is far enough to the current temperature is useless to use the heaters.

If you really need to heat the mirror is suggested to use the static temperature and let the mirror heat up for at least 30/40 minutes before starting to use the telescope. This allow the glass in the mirror to have the same temperature in every single point and avoid mirror deformations caused by hot spot on the mirror itself.

PWM Settings:

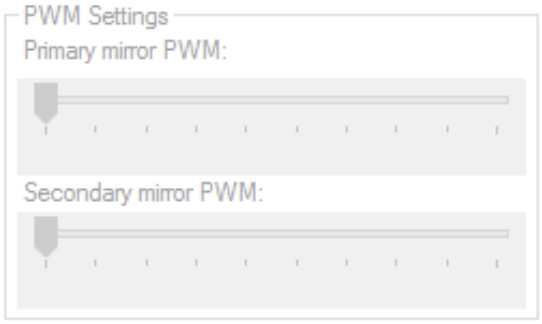


Fig. 15: Heater PWM

PWM, also known as Pulse With Modulation, is a common technique used in electronics to limit the power delivered to an electronic load.

In this particular case we use PWM to limit the power of the mirror heater.

Low value of PWM means that mirror will take more time to warm up but also that there are less thermal stress that can deform the mirror itself.

Higher values of PWM means that mirror will warm up faster but there are the possibilities that the mirror has some momentary deformations that will block you from acquiring good stars.

We suggest to keep these values under 30%. Higher values can be used without damaging the instrument but that can cause a momentary deformation that make the star shape not round.

Fans:

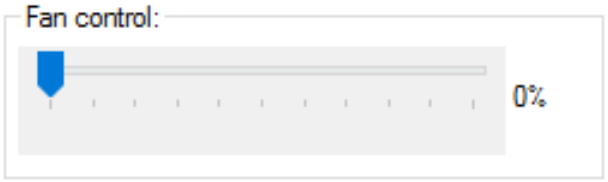


Fig. 16: Fans

Fans are used to acclimate the instrument before starting the sky observations. We suggest to turn on fans at 100% in the same moment that you open the dome. After some time, when you start to observe the sky, you can slow down the fans at 20%. This allow the telescope to maintain the temperature equals to the dome temperature and avoid turbulence to have an impact to the observation.

Single shutter control:

Fig. 17: Single shutter control

Single shutter control tab allow you to control each shutter motor as single unit. This control is not active by default and need a particular cable to connect an RS485 port to the OSBus capable shutter motor.

RS485 Serial port:

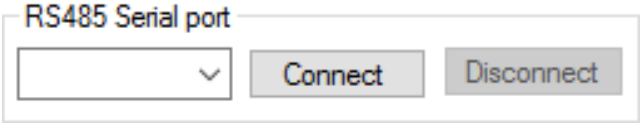


Fig. 18: RS485 Port selector

This part is used to select the RS485 interface used to connect the PC to the OSBus shutter motor.

The port list is automatically populated at the startup. Clicking on connect will open the serial port and start the OSBus listener process. Clicking on disconnect will release the serial port and the software stops to listen for OSBus commands.

Auto poll shutter status:

Fig. 19: Auto poll shutter status

Checking the "Auto poll shutter status" checkbox will enable a fixed delay poll cycle used to update the shutter status automatically. This will help you while moving single motor per time in the way you can have a feedback on the current motor status without have to click on the Read button of each shutter motor.

Test BUS:

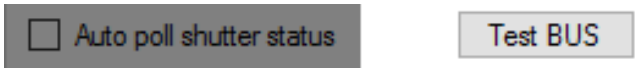


Fig. 20: Test BUS

This button will send some commands to check if all the motors on the bus are ready to operate and are correctly addressed. The results are shown in the bottom log listbox

Motor control:

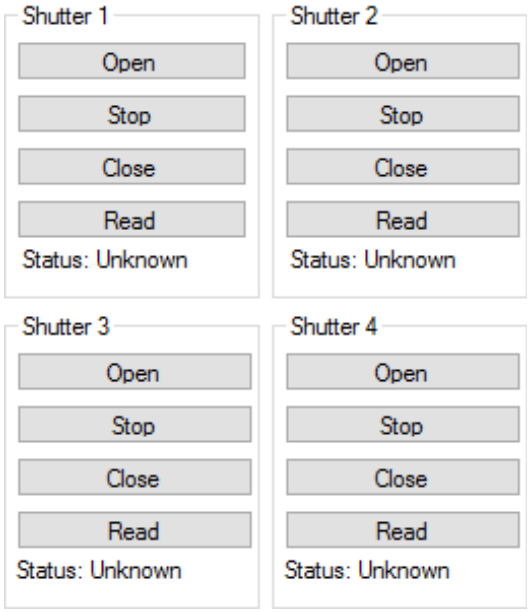


Fig. 21: Motor control

Shutter motors are addressed with number starting from 1 to 4.

The number of motor identify a position. Watching the telescope from M2 to M1 with the mount on the bottom you have:

Shutter nr. 1	Left
Shutter nr. 2	Right
Shutter nr. 3	Top
Shutter nr. 4	Bottom

The motor is in the open position when the flap is not covering the mirror.

The opening sequence is 1,2 – delay – 3,4.

The closing sequence is 4,3 – delay – 2,1.

If you drive the motor manually you **MUST** ensure that flaps **1,2 are over flaps 3,4** while in closed position. If this is not the case you can operate each single motor to restore this condition.

Grouped by motor number you can find 4 buttons and one label:

- Open: Open the shutter flap
- Stop: Stop in a mid way the shutter flap
- Close: Close the shutter flap
- Read: Read the current status
- Status: Report the latest read status.

If the “Auto poll shutter status” is checked, then the label is automatically updated during time.

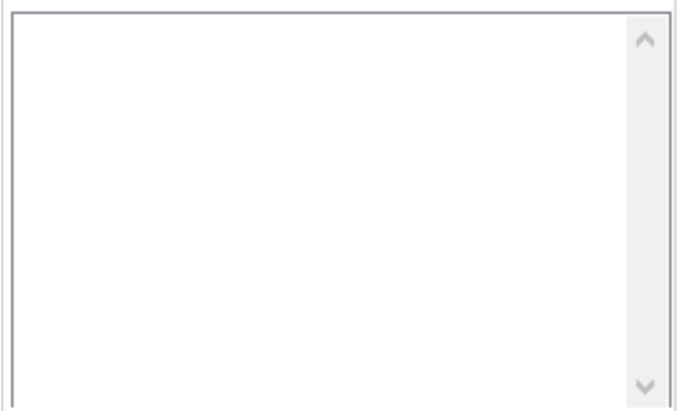


Fig. 22: Debug listbox

Debug listbox:

In this list box, found at the bottom of the single shutter control tab, you can read some debug informations like missed ACK from motors.

c) Telescope status:

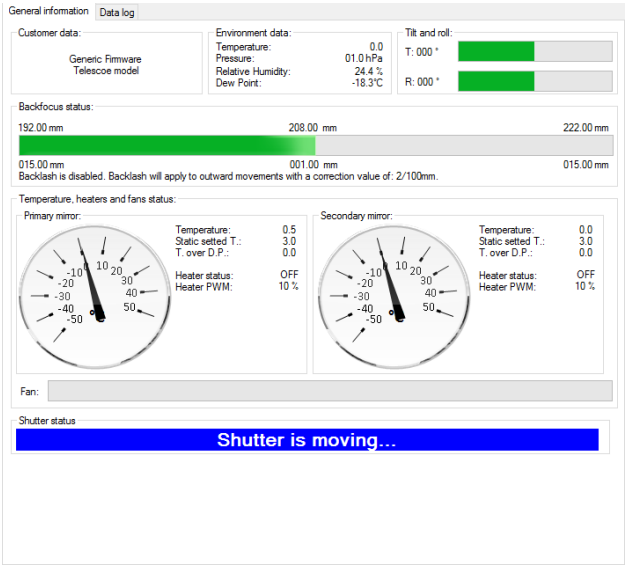


Fig. 23: Telescope status window

Fig. 24: Telescope data logger

The telescope status windows is divided in 2 tab pages. In the first one, named "General information", there are some general and specific telescope information, in the second one, named "Data log", there are some graphs that log some interesting parameters like temperature, inclination, pressure and humidity.

Telescope status – General information:

This part of the software is used for 2 main reasons: the first one is to show data received by sensor displaced on the telescope like temperature sensors, humidity, pressure, tilt and roll, the second one is to have a direct feedback of the current status of the telescope. For example if you change the fans speed, after some seconds, the fan progress bar will move according to the new speed. This is useful cause you have a direct feedback of the issued commands.

Customer information, Environmental information, Tilt and roll:

Fig. 25: Customer, Environmental and Tilt and Roll information

The top part of the General information tab is composed by some read-only information.

The first group report the customer information and the telescope model with serial number.

The second group report some environment information like ambient temperature, pressure, relative humidity and the calculated dew-point.

The third group uses the new environment sensor integrated accelerometer to calculate a tilt and roll angles. This can be used for a second feedback of telescope position.

Mirrors temperature and heater status and fans status:

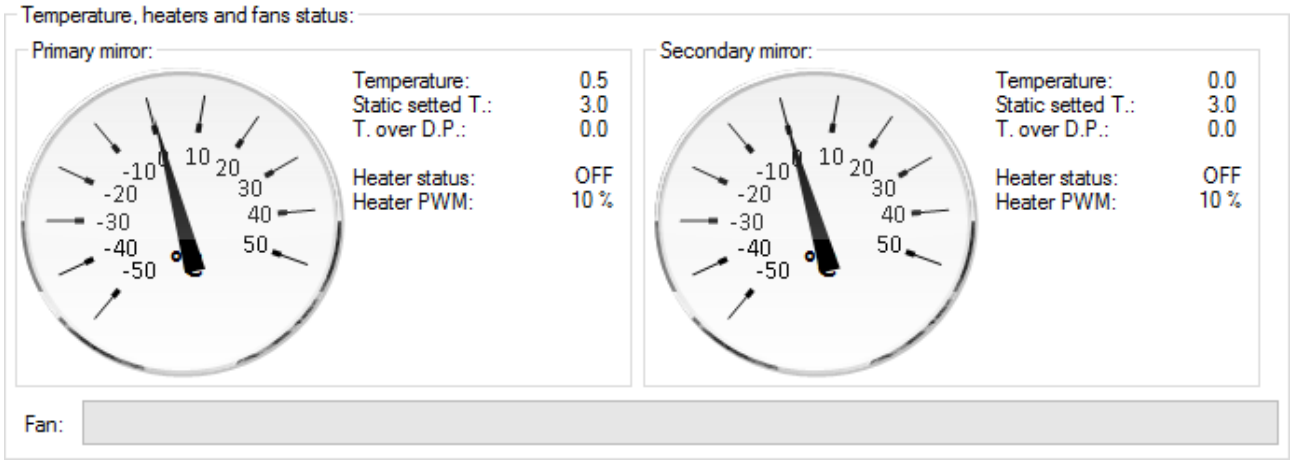


Fig. 26: Mirrors temperature and heater status and fans status

This part is used to show the current mirrors temperature status and the fans power.

For each of M1 and M2 we have:

- Temperature: the current mirror temperature
- Static setted T: temperature setted in the maintain static temperature controls. This has a white background (Fig. 26) if the temperature control is off, it turns red when the temperature control is enabled (Fig. 27).
- T. over D.P.: temperature setted in the maintain degrees over dew point control.
- Heater status: label indicating if the heater is ON or OFF
- Heater PWM: the power that ATC use to drive the heater. As told in previous chapters if this is over 30% can cause mirror deformations so, in this particular case, the label has a red background (Fig. 28).

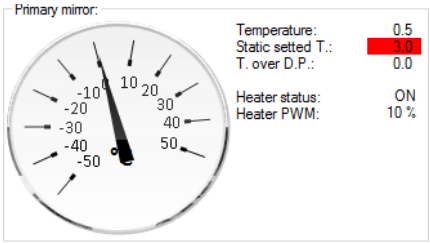


Fig. 27: Temperature control ON

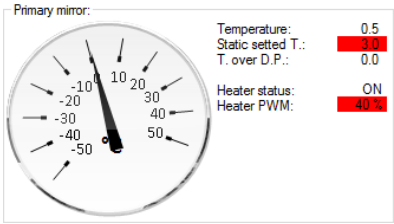


Fig. 28: PWM over 30%

The fan progress bar represent the current speed of fans. It goes from 0 to 100%.

Shutter status:

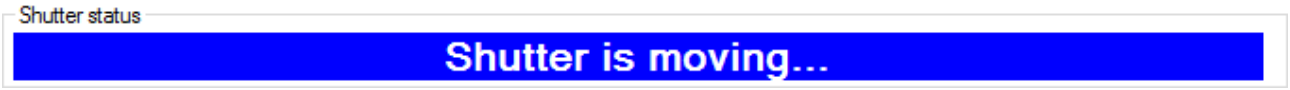


Fig. 29: Shutter status indicator

This label indicate the current shutter status according to SH02 interface. The status are:

- Opened: Label reporting "Shutter is opened" in white text on red background
- Moving: Label reporting "Shutter is moving" in white text on blue background
- Closed: Label reporting "Shutter is closed" in black text on green background

Telescope status – Data log:

These 2 graphs is used to log the data acquired by the temperature sensor and the environmental sensor. The sampling frequency is setted in the settings window. All data is also saved to a CSV file that is easily read by a spreadsheet software like Excel, OpenOffice or Google Sheets.

d) Bottom status bar

Fig. 30: Bottom status bar

The bottom status bar reports some information about the connection and the software.

In particular you can find (from left to right)

- Serial port which is connected to ATC02 unit
- Model of the telescope
- ATC Remote software version
- ATC02 unit firmware version

2. Settings window:

The settings window is accessed through the top menu bar choosing File → Settings.

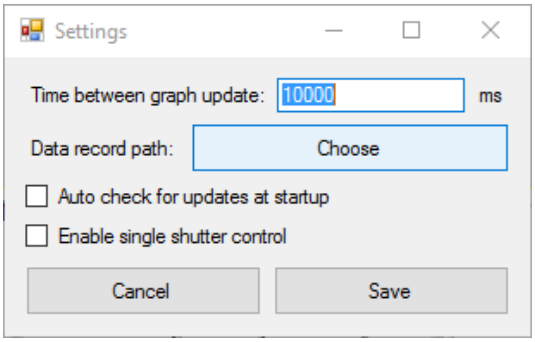


Fig. 31: Settings window

There are some options that the user can set to tune the ATC Remote software.

a) Time between graph update

This parameter, expressed in milliseconds, is the delay between the data store into the Telescope status Data log window. You can chose your preferred value without any problem but remember: using too low values means that the software save the data more often to the disk then a larger file will be generated. We strongly suggest to have this delay bigger than 10,000ms. 60,000ms can be a good value.

b) Data record path:

Clicking on the choose button will prompt you a folder selection window. The selected folder will be the storage path for the ATC data log file.

c) Auto check for update at startup (unchecked by default):

With this checkbox checked, the software automatically check for a new version at every startup. If a new version is available it will prompt you for the download. At the moment there is no auto-download/auto-install function.

d) Enable single shutter control (unchecked by default):

This checkbox will enable the Single shutter control tab in the control window.

4. ATC-CLI – Command line interface:

Starting from version 4.0.3.8 the ATC Remote package has also a command line interface. It can be used to retrieve the information of the telescope status and sensor or to drive by console the telescope system.

```
ATC Command Line Interface v.1.0.0.0
Usage:
  -c --comport  Set the ATC02 com port (required)
  -g --get      Get a specific value. Possible variables are:
                BFLMaximumDelta
                BFL
                FanSpeed
                PrimaryMirrorHeaterMaintainStaticTemperature
                PrimaryMirrorStaticTemperature
                PrimaryMirrorDegreesOverDP
                PrimaryMirrorPWMValue
                SecondaryMirrorHeaterMaintainStaticTemperature
                SecondaryMirrorStaticTemperature
                SecondaryMirrorDegreesOverDP
                SecondaryMirrorPWMValue
                ShutterStatus
                BacklashCompensationEnabled
                BacklashCompensationDirectionIn
                BacklashCompensationValue
                isDisplayOff
                AmbientTemperatureOffset
                AmbientPressureOffset
                AmbientHumidityOffset
                DewPointOffset
                PrimaryMirrorTemperatureOffset
                SecondaryMirrorTemperatureOffset
                PrimaryMirrorTemperature
                PrimaryMirrorHeaterMaintainDegreesOverDP
                SecondaryMirrorTemperature
                SecondaryMirrorHeaterMaintainDegreesOverDP
                AmbientTemperature
                AmbientPressure
                AmbientHumidity
                DewPoint
                TiltAngle
                RollAngle
                FirmwareVersion
                isCelsius
                OptimalBFLPosition
                UserInformation
                UserInformation2
                TelescopeModel
                PrimaryHeaterActive
                SecondaryHeaterActive
                IsEncoderAvailable
                EncoderFirmwareVersion
                AmbientTemperatureRawValue
                AmbientPressureRawValue
                AmbientHumidityRawValue
                PrimaryMirrorTemperatureRawValue
                SecondaryMirrorTemperatureRawValue
                IsShutterAvailable
                isMoving
                isConnected
  -s --set      Set a specific value
  -v --value    Used with -s indicate the value
  -t --shutter  Open/Close shutter (open=1 or close=0)
```

Fig. 32: ATC-CLI Help message

Fig. 31 show the capabilities of ATC-CLI software.

With ATC-CLI you can use software like CCD Autopilot to open the shutters, turn on the Fan or the heaters, and so on.

ATC-CLI.exe is located in the installation folder of ATC Remote suite.

Some examples are:

To open the shutters:

```
ATC-CLI -cCOM1 -t1
```

To close the shutters:

```
ATC-CLI -cCOM1 -t0
```

To set the fan speed at 100%:

```
ATC-CLI -cCOM1 -sFanSpeed -v100
```

To get the fan speed:

```
ATC-CLI -cCOM1 -gFanSpeed
```

5. Software upgrade:

Software updates can be automatically checked by the apposite menu row found in the Help top menu.

The user can proceed with a check for update and, if a new version is found, the user will be prompt to download.

Before installing the latest version the user need to uninstall the previous one.

To install the software is enough to follow the procedure after double clicked on the downloaded installer.

6. Troubleshooting:

Problem:	**Solution:**
When I click on connect nothing appear data is not populated on the screen.	Ensure that you choose the right serial port
	Ensure that the power supply is on and delivering power to ATC02 unit.
	Ensure that the red LED is on on the ATC02 unit.
At the first connect some information is not shown	Open ATC02 Remote software, select the right serial port, click on Connect, wait a couple of minutes, click on disconnect and restart both ATC Library Server and ATC Remote GUI. Try to connect. If the problem persist contact customer care service at support@officinastellare.com
When I open ATC Remote the software crash.	Maybe the process.txt file is corrupted. Goes to installation directory (usually located on "c:\Program Files (x86)\Officina Stellare\") and start manually ATCLibServer.exe. Then start ATC Remote software from your start menu. At the next startup everything should work.

For every other problems feel free to contact our customer service sending a mail to suppor@officinastellare.com

7. Changelog

- 20160331 – Added parameters in chapter 2.2.a, correction of some wrong information.
- 20160122 – Revised with some correction in syntax, Added the troubleshooting section.
- 20160121 – First version