A SIMPLE ARDUINO FOCUS CONTROLLER

v1.00 (08 March 2021)

BUILD A SIMPLE ARDUINO FOCUS CONTROLLER

Requirements

- Simple focus controller, no push buttons, no display
- Suitable for controlling a NEMA8-NEMA17 stepper motor (400mA per coil)
- Micro-stepping if required
- Supported by Windows/Linux application, ASCOM or INDI drivers

This document describes

- building of a simplified myFP2 controller unit
- sample schematics and strip-board layouts
- the operation of Windows applications and ASCOM drivers available
- how to initially setup the myFP2 controller for your telescope

DISCLAIMER

No obligation or responsibility is accepted on the part of the author, for any mishaps or failures caused by this product or use of this product. Users intending to use this product do so at their own risk and usage of product is deemed to be acceptance of those risks.

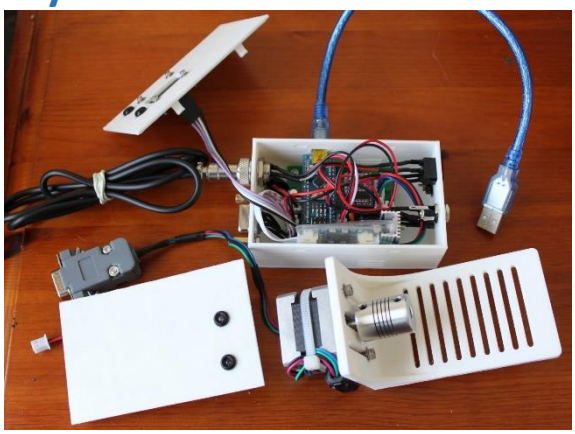
In other words, the author accepts no responsibility to damage caused to any equipment or goods or self by using this project, or loss of income or all other losses that may be incurred.

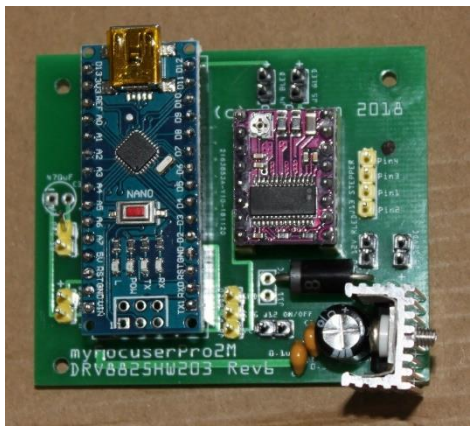
COPYRIGHT RESTRICTIONS

CONTRIBUTIONS

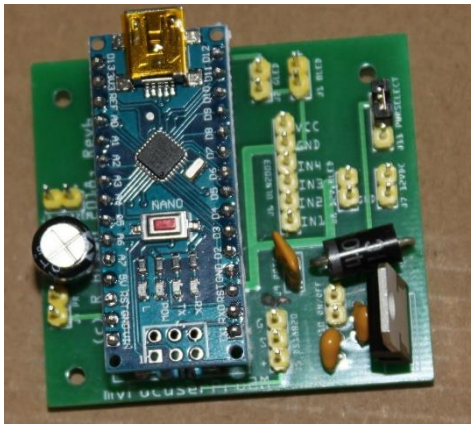
Please contribute to the ongoing development of this project via PayPal and send your donation to user rbb1brown@gmail.com (Robert Brown). All contributions are gratefully accepted.

myFP2M DRV8825





myFP2M ULN2003



https://www.youtube.com/watch?v=Pgm9UNhY1oo

https://www.youtube.com/watch?v=5dNatyl2yk0

https://www.youtube.com/watch?v=SYFPOzRNO2I

Table of Contents

DISCLAIMER

This project is released into the public domain as is where is, with no obligation or responsibility accepted on the part of the author, for any mishaps or failures caused by this product or use of this product. Users intending to use this project or code do so at their own risk and usage of product and code is deemed to be acceptance of those risks. The author(s) accept no responsibility to damage caused to any equipment or goods or self by using the ideas, schematics and code associated with this project, or loss of income or all other losses that may be incurred. No warranty is offered or implied.

COPYRIGHT RESTRICTIONS

The schematic, firmware and other code and ideas are released into the public domain. Users are free to implement these for their personal use, but may NOT sell projects based on (or derived from) this project for commercial gain without express written permission granted from the author(s).

1. You cannot sell kitsets or assembled units to others
2. You cannot copy the design and code and make your own version to sell to others, even if you make some code changes or change pin designations
3. You cannot use these designs and code ideas to make your own focuser and sell versions of those to others

FIRMWARE LICENSE RESTRICTIONS

This code is released under license. If you copy or write new code based on the code in these files you must include to link to these files AND you must include references to the author(s) of this code.
Firmware is released subject to the following restrictions

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), the rights to use, copy, modify, merge, publish, distribute, sublicense copies of the Software (from the authors), **subject** to the following conditions:

1. The above copyright notice shall be included in all copies or substantial portions of this Software.

2. The software may NOT be sold or charged for in any way, either as part of a package, license fee, annual charges or premiums and shall include the copyright notice above.

3. The source-code to any software MUST be made available in full source-code form (including modified source files) and free to download without payment and include the copyright notice above

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

myFP2M FEATURES

- **myFP2M** focus controller with 3D printed case (100Lx62Dx40Hmm, 140gm/5oz)
- Windows application and ASCOM driver
- Linux application and INDI driver
- Support for DRV8825, ULN2003 driver boards, L298N with micro-stepping in software
- DRV8825 stepper motor smoother diodes are standard for better micro-stepping performance
- The controller is connected via a USB connection (high quality 1.5m USB shielded cable provided)
- 1.5m power cable twin core with 2.1mm x 5.5mm male connector
- A 30V 3A fuse (resettable) PPTC provides protection against excessive current
- 12VC power plug and 1m cable (red is positive)
- Backlash in firmware, enable/disable, IN and OUT
- Three speed settings (Slow, Medium, Fast)
- Step Size, enable/disable, specify step size in microns
- Up to 8 focuser pre-sets available
- Controller settings are remembered and saved on chip

SUPPORT VIDEOS

myFP2M-DRV8825 https://www.youtube.com/watch?v=Pgm9UNhY1oo
myFP2M-ULN2003 https://www.youtube.com/watch?v=o1UVqP6QM2Q
myFP2M-L298N https://www.youtube.com/watch?v=SYFPOzRNO2I

FILES YOU NEED TO DOWNLOAD

APPLICATIONS

Windows Link
Linux Link

3D PRINTED CASE

All boards Link

FIRMWARE

All boards Link

PARTS LIST

All boards Link

PCB GERBER FILES AND SCHEMATIC

All boards Link

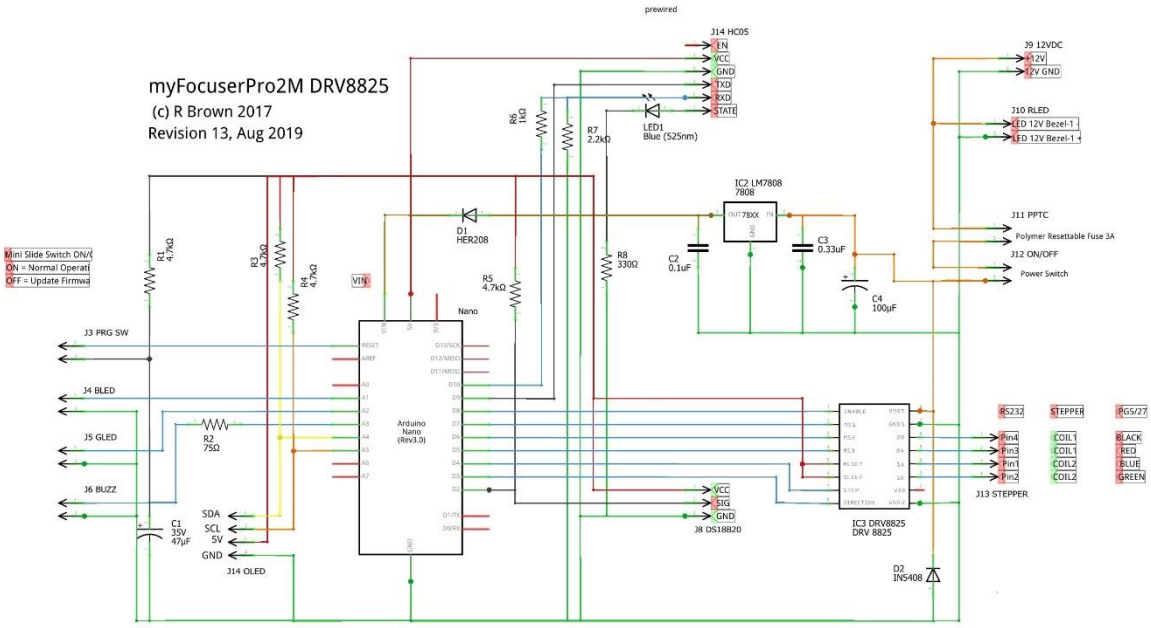
PRECAUTIONS

Never disconnect or connect the stepper motor when the myFP2M Controller is ON or when external power is ON. This will result in damage to the driver board or controller

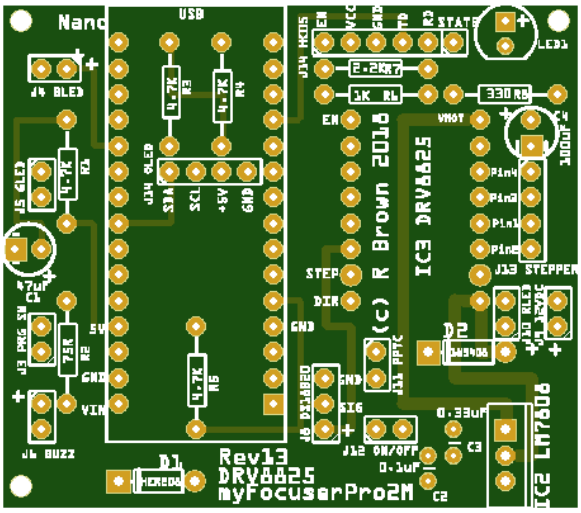
Never disconnect or connect the temperature probe when the myFP2M Controller is ON or when external power is ON. This may result in damage to the probe or controller.

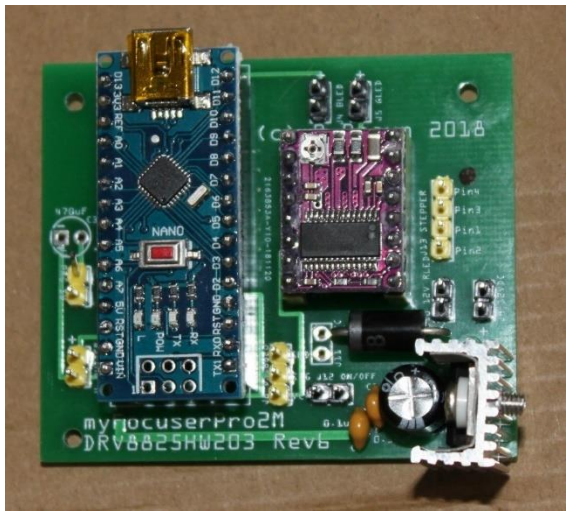
MYFP2M DRV8825

SCHEMATIC

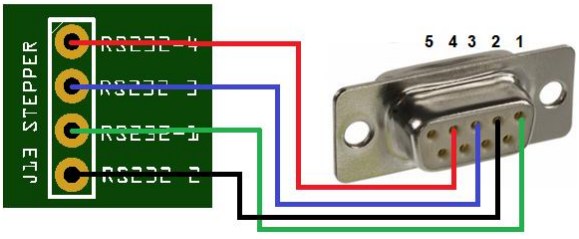


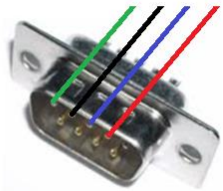
PCB



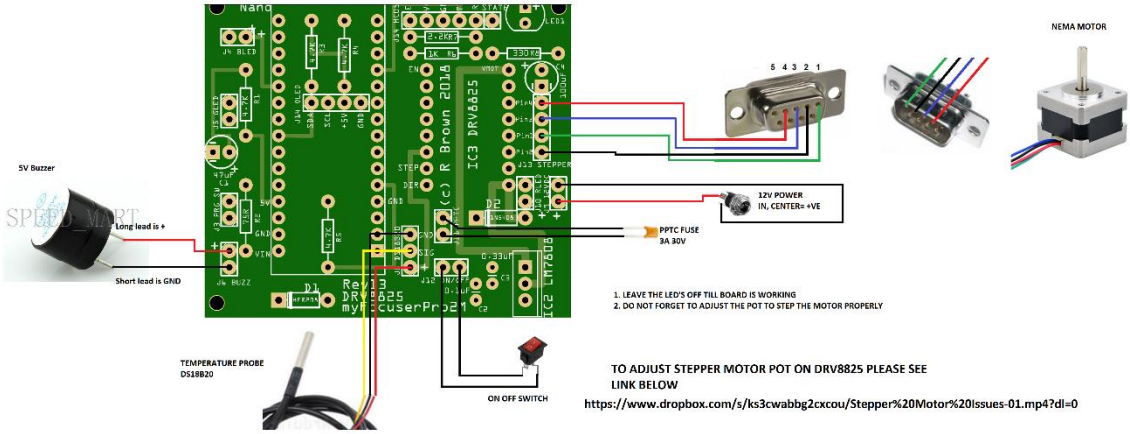


WIRING

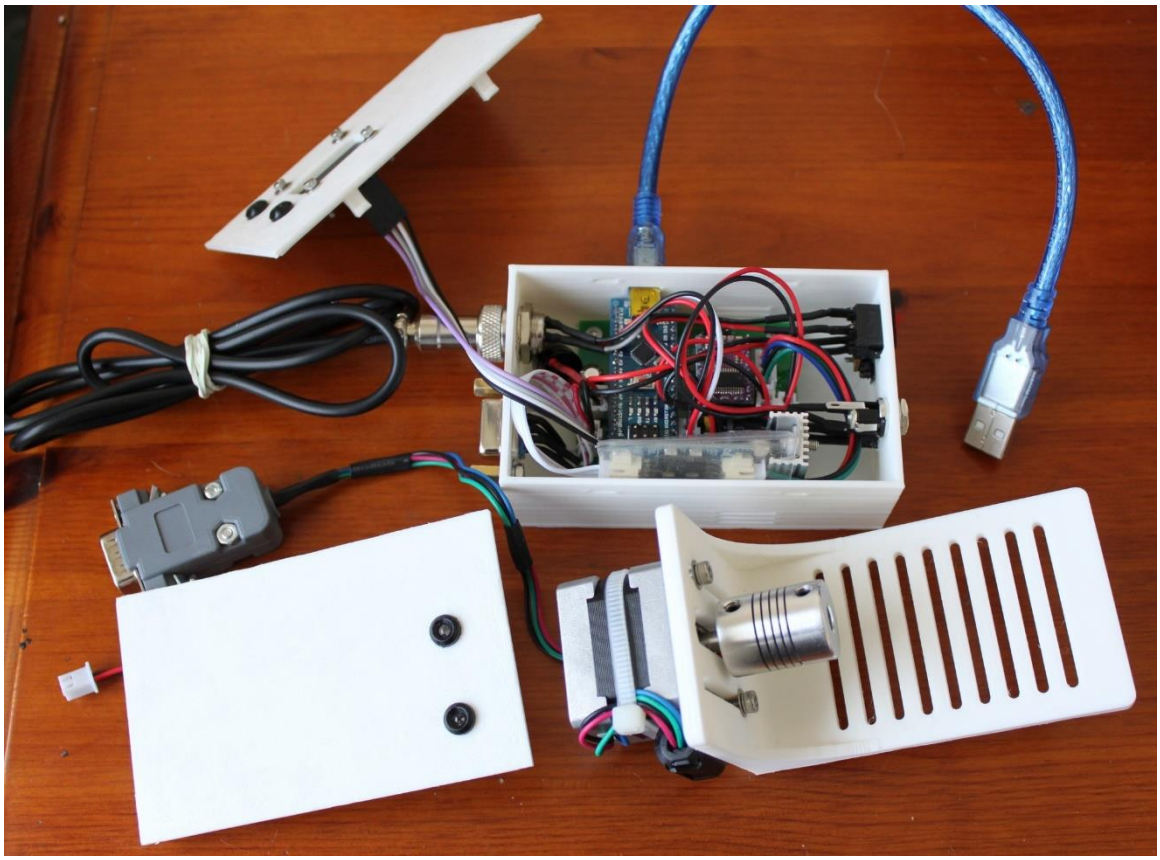




PCB WIRING



PCB ASSEMBLY



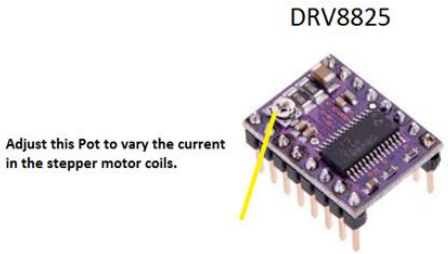
FIRMWARE TO USE

myFP2M-DRV8825-2xx.ino

DRV8825 STEPMODE

The DRV8825 supports FULL, ½, ¼, 1/8, 1/16 and 1/32 stepping in firmware. Settings are saved in EEPROM. To support micro-stepping, the controller has smoother diodes fitted as standard.

DRV8825 CURRENT ADJUSTMENT



The POT is adjusted before delivery.

If you are using a different stepper motor, you may need to adjust the POT on the DRV8825 board to get optimal stepping of the stepper motor. This pot adjusts the current that flows in the coils of the stepper motor.

Adjusting the Stepper Motor manually

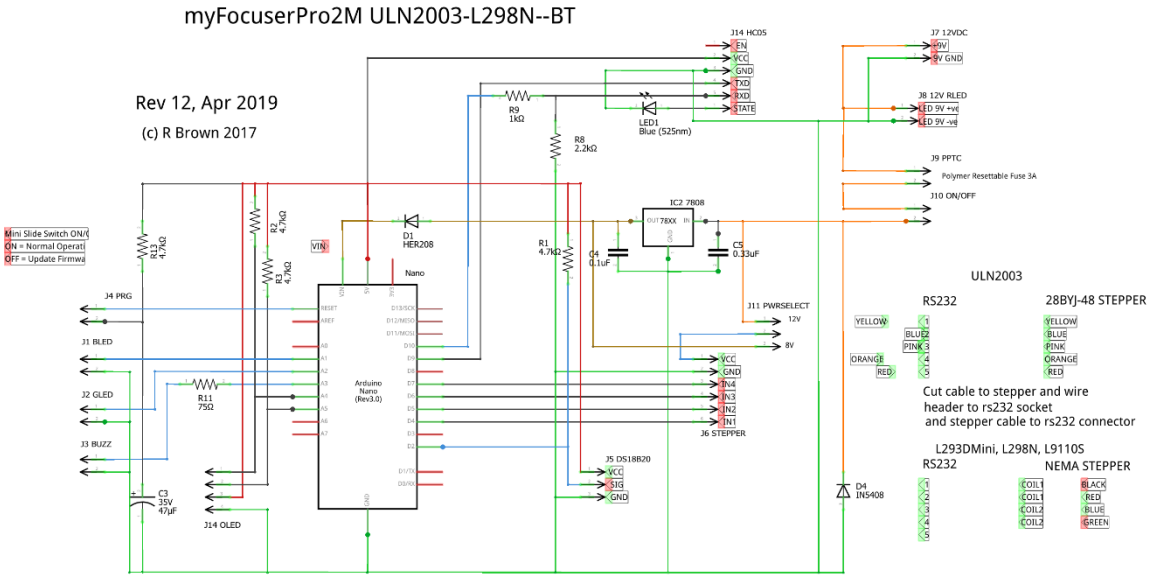
It is best to use a ceramic or plastic screwdriver when adjusting the pot. I would suggest a plastic knitting needle which has the end filed down to look like a screwdriver.

1. With the controller connected via USB, and 12V power to the driver board, set the focuser position to 0 and the Motor Speed to SLOW
2. Enter a focuser position of 5000 and click the Goto button
3. Wind the pot all the way anticlockwise until the motor stops moving
4. Now very slowly turn the pot clockwise until you see the motor start to turn. If the maxSteps is reached, just reset the focuser position to 0 and then type in 5000 for the position and click Goto again
5. Slowly turning the pot, when you see the stepper start to move ok without jerking, then slowly turn no more than 1/8th clockwise from that point
6. It should now be close enough
7. If you go too far then there will be too much current and the motor will run hot. You should use no more than 12V external power

On some driver boards clockwise might be anticlockwise. Once set, then switch to 1/4 stepping and repeat the 0 then 5000 Goto. The motor should run smoothly without missing steps (a missed step will be a sudden jerk which you will be able to feel or hear). If there is any of this, you might need to ever so slightly turn it a little more. Be careful as a little turn can make significant changes in current.

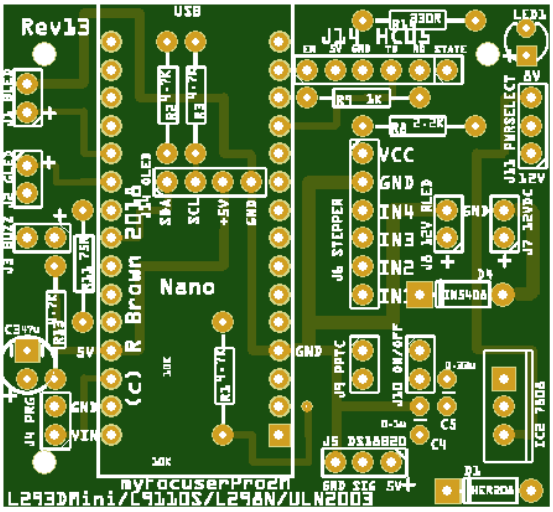
MYFP2M L298N

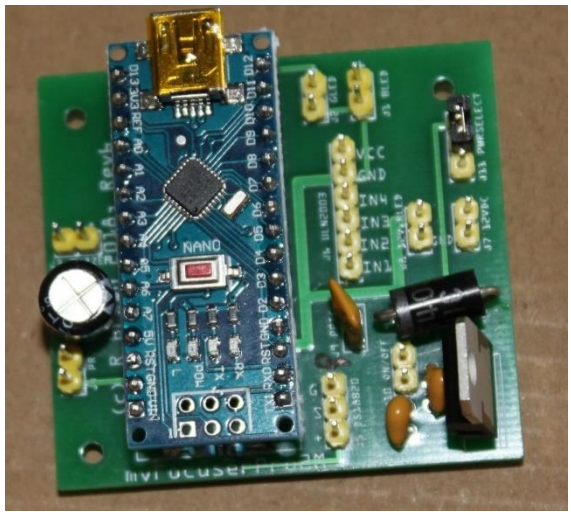
SCHEMATIC



PCB

The L298N version uses the ULN2003 PCB with a L298N driver board.

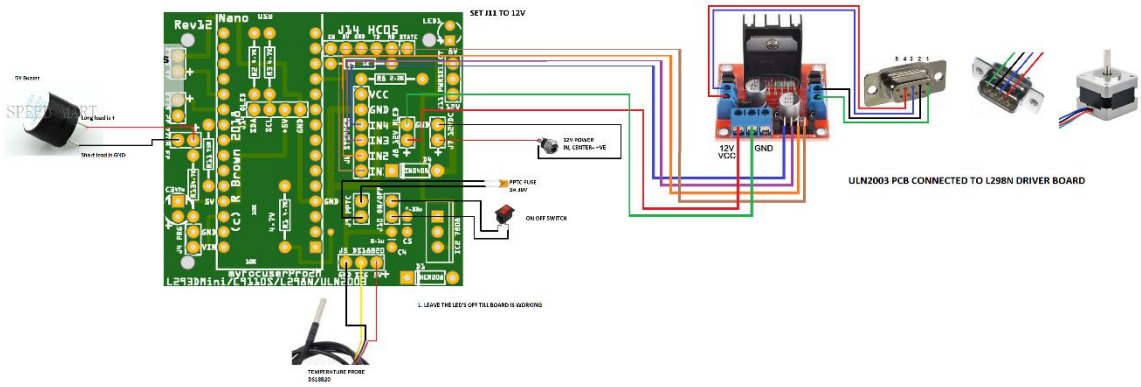




WIRING

// TODO

PCB WIRING



PCB ASSEMBLY

// TODO

FIRMWARE TO USE

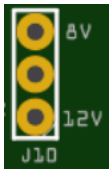
myFP2M-L298N-2xx

L298N STEPMODE

The L298N supports FULL and HALF stepping in firmware. Settings are saved in EEPROM.

J10 8V/12V STEPPER POWER SELECT

J10 can provide either 8V or 12V power to the L298N board. This is normally set to **12V** (middle pin of J10 connected to the 12V side of J10 using a motherboard shunt jumper

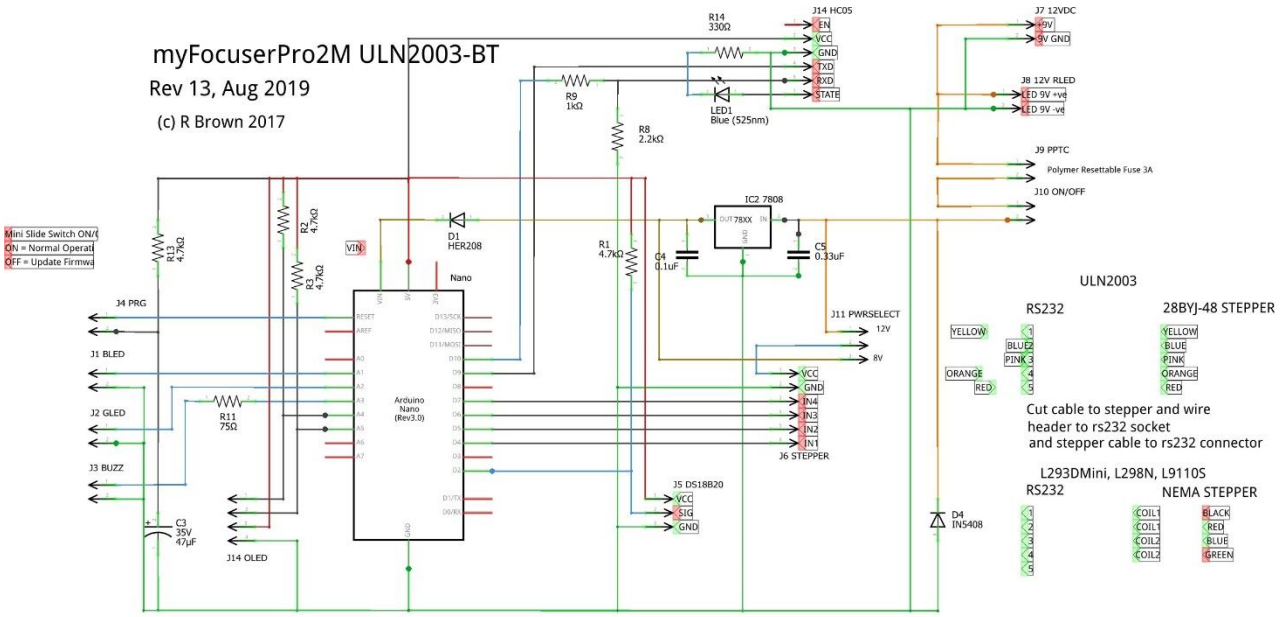


PCB Power select

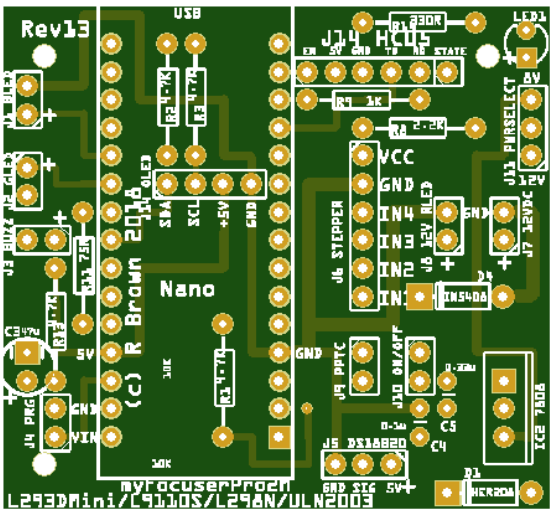
MYFP2M ULN2003

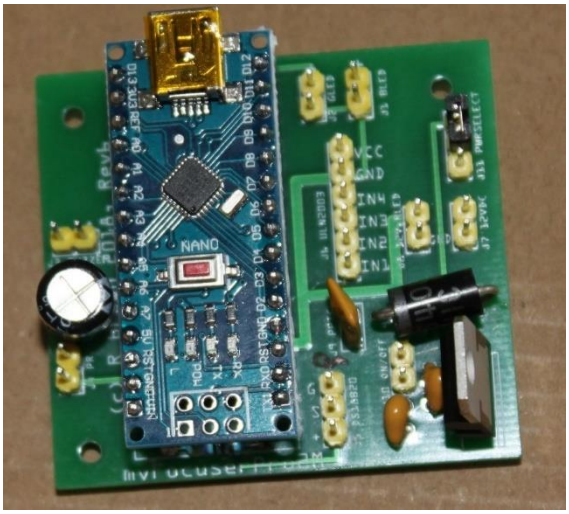
Suitable for telescopes such as MAK or SCT where the focuser knob moves a mirror instead of a draw tube, and guide scopes

SCHEMATIC

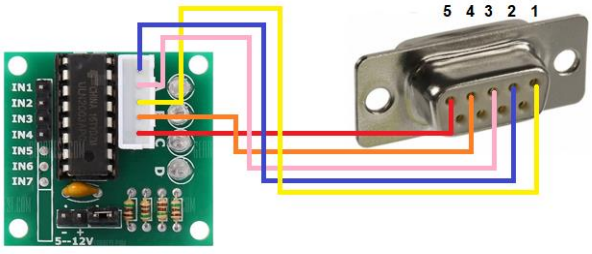


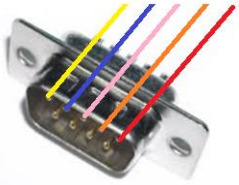
PCB



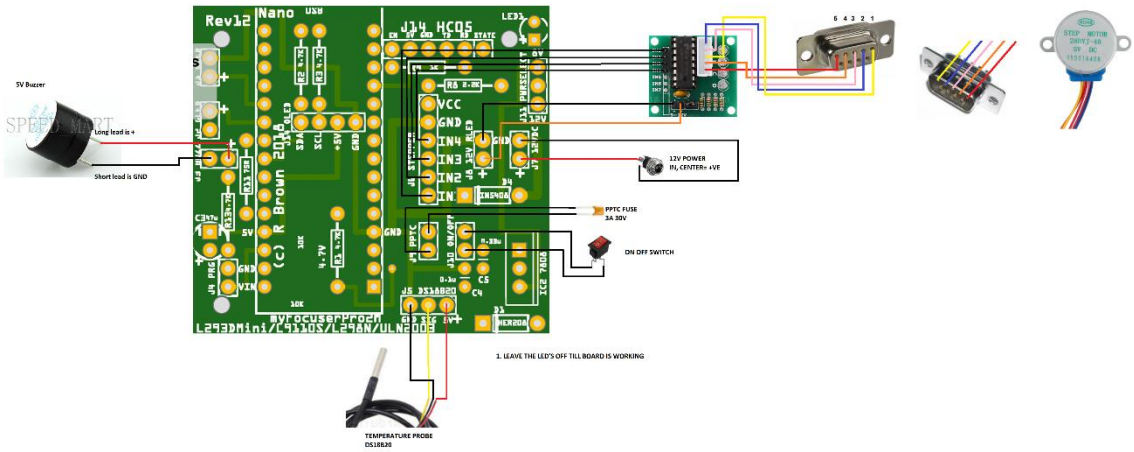


WIRING

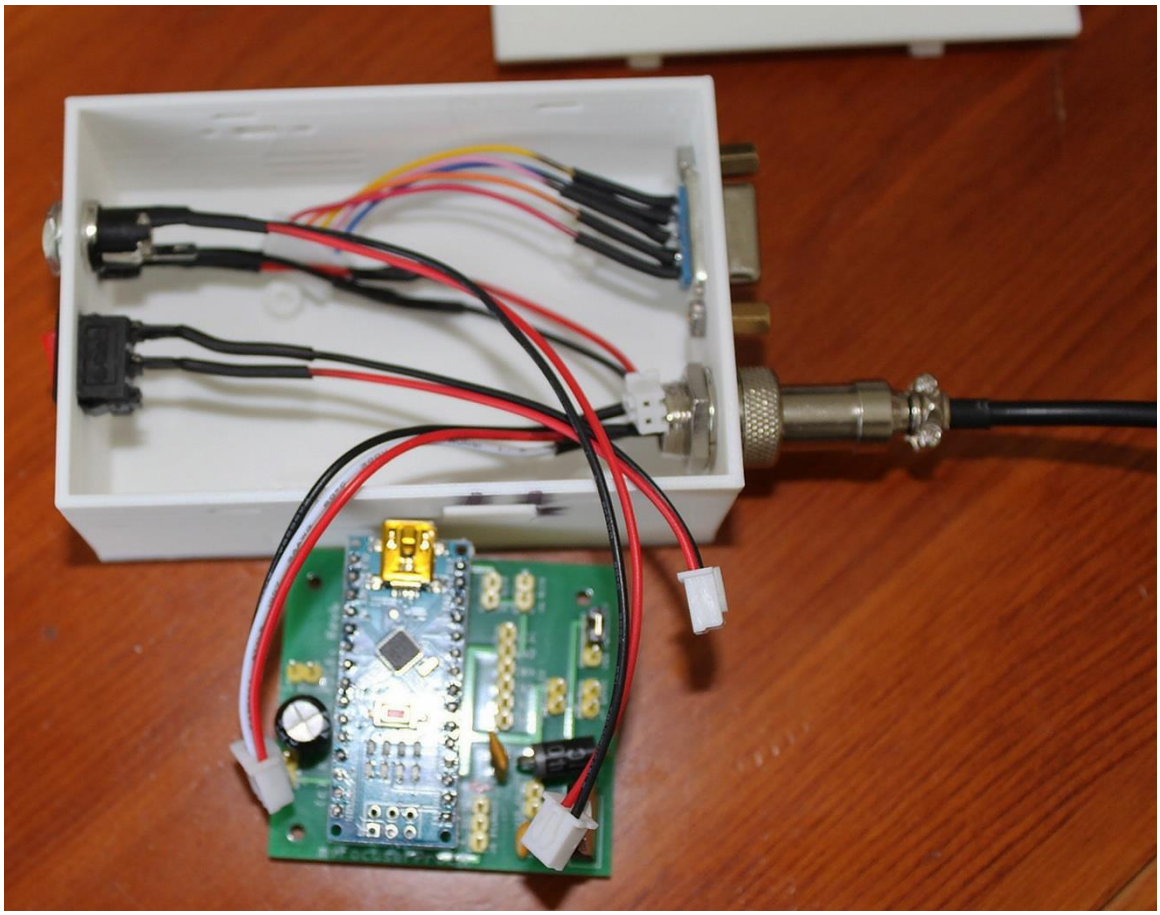




PCB WIRING



PCB ASSEMBLY



FIRMWARE TO USE

myFP2M-ULN2003-2xx

ULN2003 STEPMODE

The ULN2003 supports FULL and HALF stepping in firmware. Settings are saved in EEPROM.

TEMPERATURE PROBE

One temperature probe (DS18B20) is supported, with resolution set by the controller at 0.25 degree of resolution (can be changed). The accuracy of the measured temperature is within 0.5 degrees Celsius.

The probe cable length is either 1m or 2m (length is approximate), specified when ordering.

The temperature probe is wired using a GX-12 aviation style male socket and female connector. A screw lock ring holds the probe securely in place.

Disconnection or reconnection of the probe must be done when power is OFF.

When a temperature probe is connected, the controller supports temperature compensation.

TEMPERATURE PROBE CONNECTORS

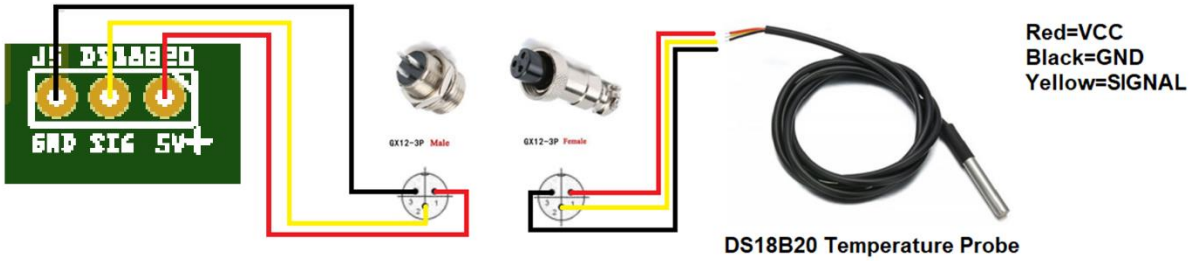
A GX12 female/male connector is used as a reliable way of interfacing the temperature probe.

https://www.ebay.com/itm/GX12-2-3-4-5-6Pin-12mm-Male-Female-Panel-Metal-Aviation-Plug-Socket-Connector/253246268801

Note: The temperature probe is NOT used for reading ambient temperature. The probe is used to read the temperature of the focuser tube. This reading can then be used to adjust the position of the focuser as the tube temperature changes, keeping the image in focus. This is called temperature compensation. The amount by which the focuser tube expands or contracts according to temperature change is called the temperature coefficient, and is measured in motor steps per degree Celsius.

You need to secure the temperature probe where it can actually measure the temperature of the focuser tube. For a refractor this would be about half-way down the tube length.

TEMPERATURE PROBE WIRING



12V POWER CONNECTION

Power to the controller is via a 12V 3A power source.

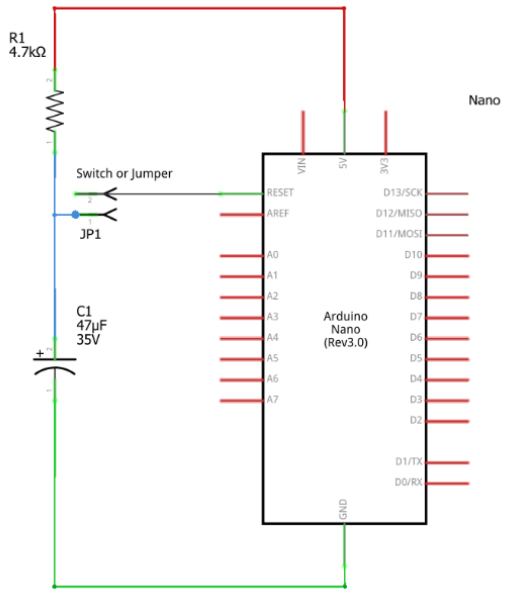
The power socket is 5.5x2.1mm female, and a 5.5mmx2.1mm male plug with 2M of twin core used, red is positive and black is ground.

The centre pin of the connector is positive.

The 12V power is wired through an ON/OFF toggle switch, a PPTC (polymer resettable) 3A fuse and a 12V RED LED (pre-wired), which lights when power is ON to the controller.

CONNECTION RESET PREVENTION CIRCUIT

The following circuit provides a means of **disabling** the Arduino reboot that occurs when connecting to the controller via USB.



myFP2M comes with its own Windows Applications and ASCOM drivers.

When connecting, applications like Moonlite drivers, APT, FocusMax and other applications/drivers attempt to restart the controller so it starts in a known state.

The controller can take up to 3s to respond after a reset, during which time the application can time out and not connect to the controller.

To overcome this timeout, the reset of the Nano on a serial connection request must be prevented. If you wish to prevent this reset when the software connects to a controller, the power-on reset prevention circuit is required.

Connection reset prevention circuit explanation

Purpose: Prevent Arduino resetting when a connection is made on the serial port to the controller

When an application or driver in Windows or ASCOM connects to an Arduino, a reset is applied which forces the Arduino to reset. During the reset, the Arduino cannot respond to requests on the Serial port till the reset is finished. For an Arduino controller such as a focuser controller this can take 2-3 seconds.

If an application connects to the controller a reset happens, the controller cannot respond until 2-3 seconds has elapsed. if the application connects then immediately sends a request without waiting, the application can time out with a serial response error - the moonlit driver for example will only wait 1s.

To prevent the Arduino resetting when a connection is made, which stops the reset cycle and thus making the Arduino respond immediately, we need to stop the reset.

This is done via a capacitor between the RST line and either VCC or GND. There are many ways to do this. Including using resistors.

We need to prevent the RST line from going low, so it has to be held high.

One way is to put a 4.7K resistor from RST to 5V, and then use a 47uF electrolytic (bipolar) capacitor between RST and GND (+ve lead to RST, -ve lead to GND). This keeps the RST line high and prevents the reset from occurring when a connection is made.

The downside is that you will not be able to reprogram the Arduino chip unless the reset prevention circuit is removed. There needs to be some way to remove this circuit (either using jumpers or a switch so that the Arduino chip can be reprogrammed with new firmware updates.

STEPPER MOTORS

The controllers use either a NEMA14 type stepper motor or a 28BYJ-28 geared stepper motor. It is possible to replace the NEMA14 with a geared NEMA17 stepper motor. Standard cable length is approximately 20cm.

NEMA14 0.9° STEPPER MOTOR

The NEMA14 is much lighter and uses 0.9° step movement with 400 steps per revolution. Using microstepping, this gives 800 steps at half-stepping. This stepper is for use with the myFP2-DRV or myFP2-L293D controller units. The shaft size is 5mm/4.5mm notched. This stepper motor is ideal for the majority of focusing solutions. The manufacturer lists the holding torque as 0.11Nm, *significantly* higher than a 28BYJ-48 motor.
NEMA14 Bipolar Stepper Motor .9deg 0.4A 11Ncm 14HM11-0404S

The DRV8825 version of the myFP2M controller features a diode smoother that provides better shaping of motor pulses when using microstepping. At 1/8 microstepping 3200 steps per revolution, the NEMA14 (120g) is a viable alternative to using a PG5 NEMA17 motor (500g).

An extension cable is available.

It is not recommended to run the DRV8825 controller with step modes higher than 1/8 as the torque of the motor is significantly reduced compared to full or half stepping.

28BYJ-48 BIPOLAR GEARED STEPPER MOTOR

This geared unipolar motor can be used with the myFP2ME-ULN and myFP2ME-L293D controller units. The stepper motor has 2048 steps at full stepping or 4096 steps at half-stepping. *This is a lightweight motor suitable for visual use only, guide-scopes, Mak's and SCT's*. The ULN version will have insufficient power to handle heavier imaging combinations.

The shaft size is 5mm/3.2mm notched. The controller boards supply either 8V or 12V via a use selectable jumper. The controller board uses 8V to drive the 5V motor, allowing stepper cable lengths of up to 2M.

In terms of torque, the 28BYJ-48 manufacturer lists

In-traction Torque > 34.3mN.m (120Hz)
Self-positioning Torque > 34.3mN.m
Friction torque 600-1200 gf.cm
Pull in torque 300 gf.cm

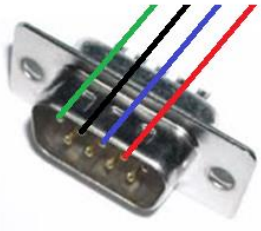
STEPPER MOTOR CONNECTION

Stepper motor connections are via an 9 pin DB9 connector. The 5V stepper motor is provided by default. The 12V version is available as an option. The ULN2003 controller can support both.

28BYJ-48

Pin1-2 Coil-1, Pin3-4 Coil-2, Pin5 Common

NEMA14/NEMA17-PG5/PG27



Pin1-2 Coil-1, Pin3-4 Coil-2

STEPPER MOTOR EXTENSION CABLES

The basic stepper motor cable length is approximately 19cm. This is sufficient if the controller box is mounted close to the focuser. However, if the controller is not mounted near the focuser (such as on the mount) then a stepper motor extension cable is required.

Stepper motor extension cables are available in either ½ metre or 1 metre lengths (actual lengths may be slightly less than indicated).

STEPPER MOTOR MOUNTING BRACKETS

A number of mounting brackets are available, suitable for mounting on a refractor or Crayford focuser. These brackets are designed for specific stepper motors. If the dimensions of the available brackets are not suitable, custom made 3D printed brackets are available on request, provided sufficient measurements are provided.

As these are 3D printed, please be aware that there may be minor variations in print quality and measurements. You may need to gently file or enlarge some slots as required. In addition, the brackets are made from PLA and will deteriorate if regularly exposed to UV or direct sunlight.

The brackets attach to the bottom of the focuser.

Normally the focuser has probably 4 small screws and 1 large screw [the focus lock screw]. The 4 small screws hold the focuser assembly together and it is possible to replace two of these screws and affix the bracket.

You will need to measure the length of the existing screw, then add the depth of the recess, the width of the bracket [4mm], plus the width of washer [1mm].

Your local hardware store or eBay is a good source for screws. Most focusers today use metric screws.

Do not overtighten the screws or bolts holding the motor to the bracket, or the screws used to attach the bracket to the focuser body. This could cause damage to the bracket.

The brackets are printed using 100% infill so are solid and extra holes can be drilled if required.
Brackets can be mounted in two different ways, depending upon whether the drive train is direct connected using a flexible couple, or using a belt and gears.

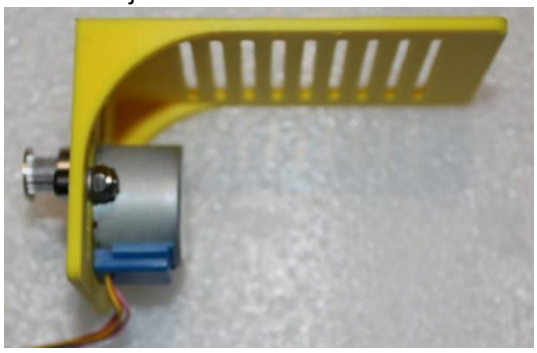
The photos below illustrate the two different ways in which the bracket can be attached to the focuser.

Attaching Stepper Motor direct to focuser shaft using a flexible coupler

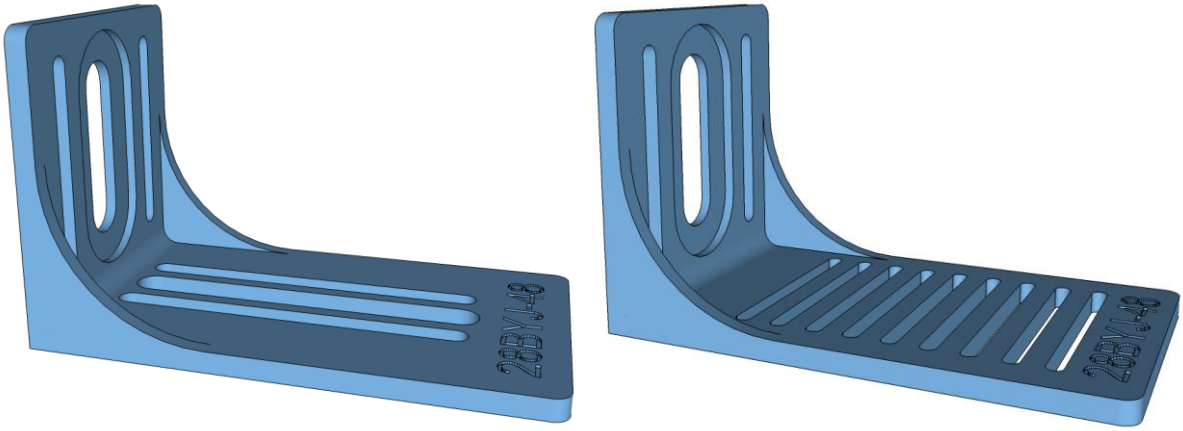
The stepper motor fits to one side of the focuser shaft and the motor uses a flexible couple to attach direct to the focuser shaft. Slots in the bracket allow the stepper motor to positioned at the same height as the focuser shaft as well as the correct distance apart to allow for fitting of the flexible coupler.

Attaching Stepper Motor to focuser shaft using a pulley and belt

The stepper motor fits underneath the focuser. The focuser shaft has a large pulley and is connected to the stepper motor via a belt. Slots in the bracket allow the stepper motor to move up and down providing tension adjustment.

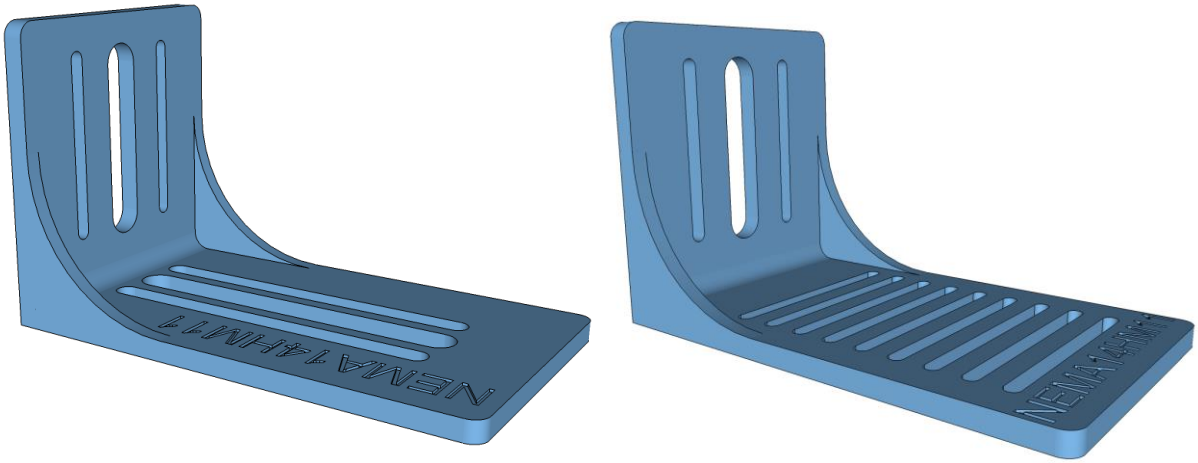


28BYJ-48 BRACKETS



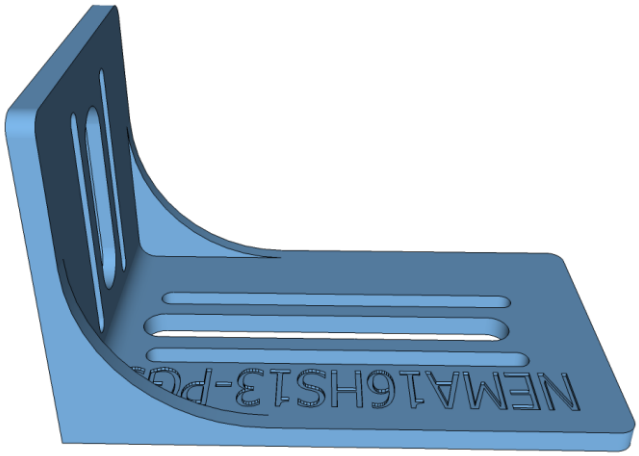
https://www.thingiverse.com/thing:3731843

NEMA14 14HM11-0404S



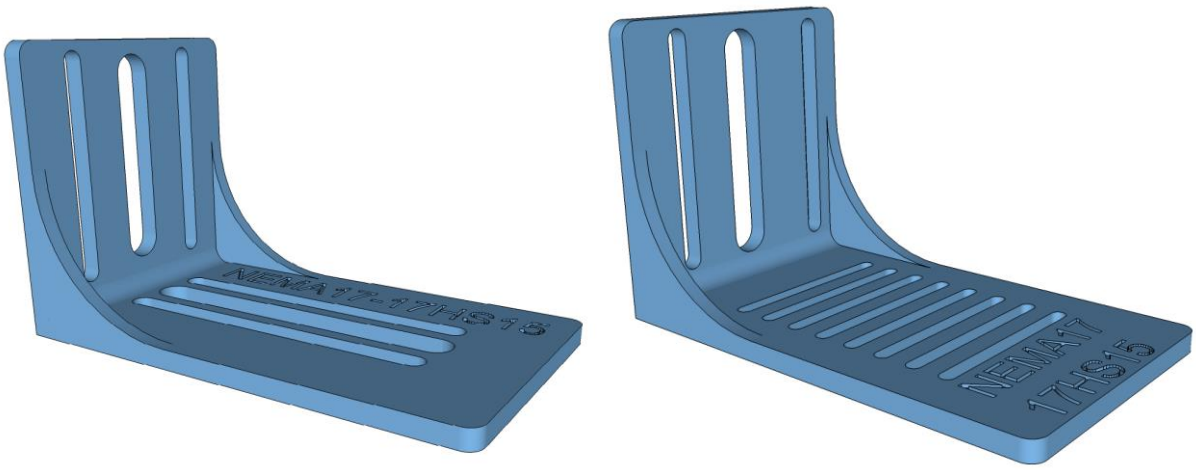
https://www.thingiverse.com/thing:3731855

16HS13-0604S-PG5



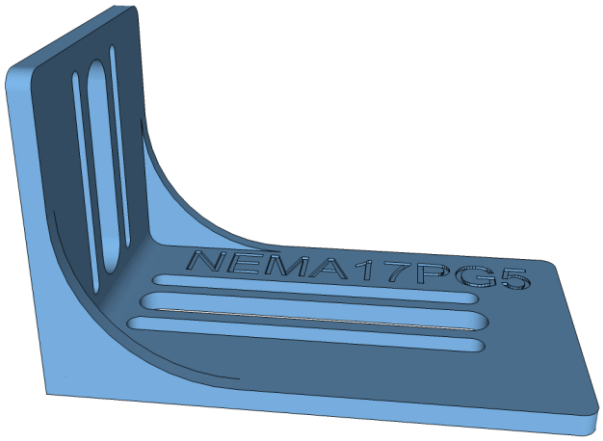
https://www.thingiverse.com/thing:3731917

NEMA17-17HS15-0404S



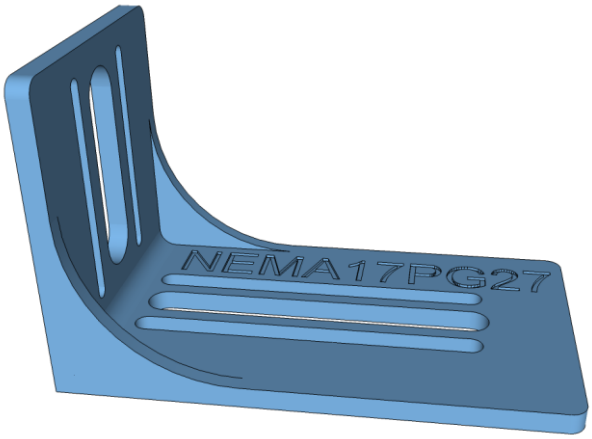
https://www.thingiverse.com/thing:3731927

NEMA17-PG5-0404S



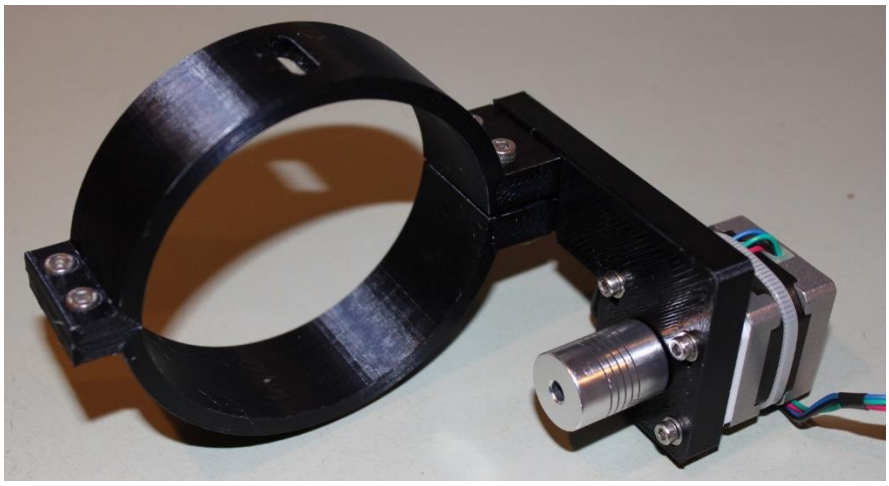
https://www.thingiverse.com/thing:3731932

NEMA17-PG27-0404S



https://www.thingiverse.com/thing:3731940

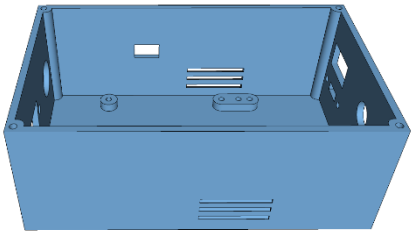
WILLIAM OPTICS FD-80 BRACKET

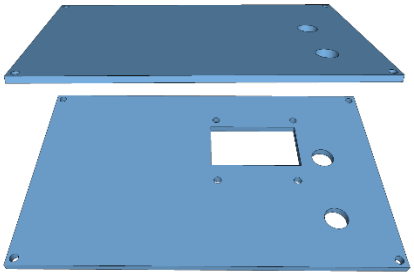


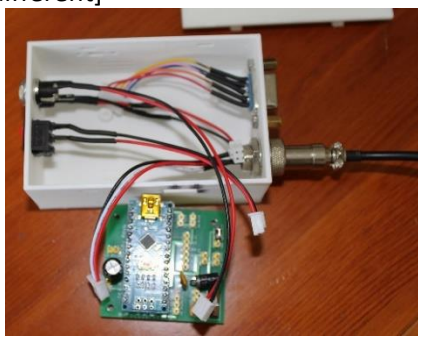
https://www.thingiverse.com/thing:3731836

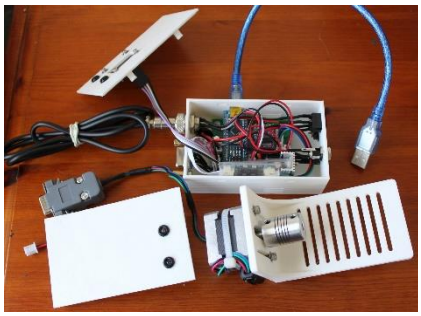
MYFP2M CASE

100mm long x 62mm deep x 36mm high [final version may look slightly different]









ULN2003/L298N Case	https://www.thingiverse.com/thing:3733591
DRV8825 Case	https://www.thingiverse.com/thing:3733583

MYFP2M FIRMWARE

Create a project folder on your computer that will hold all the files required for the controller. Download the firmware zip file from the Sourceforge site.

https://sourceforge.net/projects/arduinoascomfocuserpro2diy/files/myFP2M/Arduino%20Firmware/

HOW TO PROGRAM THE FIRMWARE

Programming is the same as for any other myFocuserPro controller. The controller is connected via a USB cable to the computer. The Arduino IDE is started and the firmware file and com port selected. The target is set to Arduino Nano and then the upload button in the IDE is selected.

The Arduino IDE will automatically place the Nano chip into flash mode and once the upload is completed, the Nano is reset and the controller starts running the firmware.

INSTALL THE ARDUINO IDE

To reprogram a controller requires the Arduino IDE software application. Please **download and install** the latest IDE software from. https://www.arduino.cc/en/main/software [as at writing v1.8.9 – do NOT use the web editor based version]. Once the Arduino IDE is installed, you can start installing the necessary libraries.

COPY THE LIBRARIES FIRST

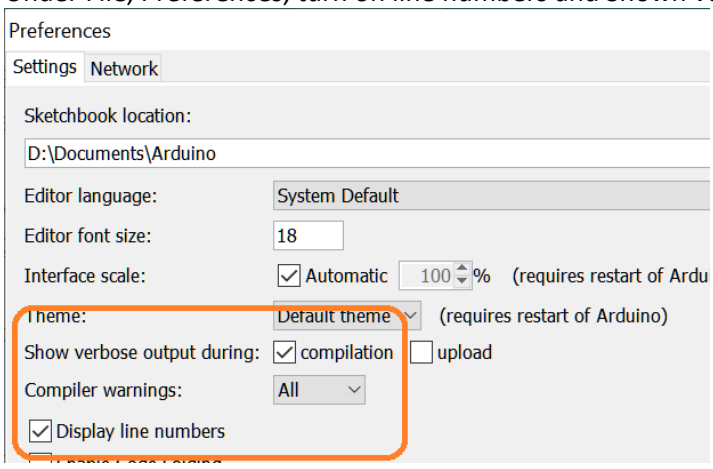
Unzip the firmware file, which will create a myFocuserPro2N Libraries folder. Navigate to the **myFP2M Libraries** folder. Copy **ALL** the folders to the Documents\Arduino\libraries folder (for Windows, for other OS's please consult were the Arduino IDE stores user libraries).

Do not edit or replace any of these provided library files. They have been edited and modified to use with the myFP2M firmware.

Note: *These library folders are the same as provided with the myFocuserPro2.*

ARDUINO IDE CONFIGURATION

Under File, Preferences, turn on line numbers and Shown verbose output on compilation.



FIRMWARE FILES

There is a firmware file to support each driver board. Select the correct firmware file that matches the Driver Board you are using.

MYFP2M FIRMWARE OPTIONS

The firmware files that are used to program the controller support different hardware driver board options as well as focuser options such as temperature probe, display and backlash.

These are either enabled or disabled by the user.

The idea is the user configures the firmware options to match the hardware they have purchased.

The firmware file has most options disabled.

CONFIGURING THE MYFP2M FIRMWARE

The firmware file is comprised of a number of sections in which you will need to change some things.

CHANGES YOU MUST MAKE TO THE MYFP2M FIRMWARE CODE

Before you program the controller, you must make a few changes to the firmware. If you have already programmed the controller, simply make the required changes and then re-program the module again.

A NOTE ABOUT OPTIONS

Any line that begins with 2wo slashes // indicate the line is commented out – ignored by the compiler. To enable an option means making sure that your choice does NOT start with two slashes and the other choices are commented out (begin with two slashes)

// 1: SPECIFY DRIVER BOARD HERE

You first need to specify the Driver board you are using. This is done in section 1. The following example shows the driver board as ULN2003. You should have no need to change the settings in this section

```
// ------------------------------------------------------------------------------------------------------
// 1: SPECIFY DRIVER BOARD HERE
// ------------------------------------------------------------------------------------------------------
// DRIVER BOARDS - Please specify your driver board here, only 1 can be defined, see #define DRVBRD lines
#include "myboards.h"

#define DRVBRD ULN2003

#ifndef DRVBRD
#halt // ERROR you must have DDRVBRD defined
#endif
```

// 2: SPECIFY STEPPER MOTOR HERE

This section is not used in the myFP2M firmware

// 3: SPECIFY HARDWARE OPTIONS HERE

This section in the firmware file lets you enable options like a display or temperature probe. The following example enables the temperature probe, backlash, buzzer and in-out leds.

```
// -------------------------------------------------------------------------------------------------
// 3: SPECIFY HARDWARE OPTIONS HERE
// -------------------------------------------------------------------------------------------------
// Caution: Do not enable a feature if the associated hardware circuits are not fitted on the board
// Enable or disable the specific hardware below

// To enable bluetooth, uncomment the next line (only available on some boards)
//#define BLUETOOTH 1

// To enable the temperature probe, uncomment next line
#define TEMPERATUREPROBE 1

// To enable the OLED DISPLAY uncomment the next line
//#define OLEDDISPLAY 1

// To enable backlash in this firmware, uncomment the next line
#define BACKLASH 1

// To enable the buzzer, uncomment the next line
#define BUZZER 1

// To enable the IN-OUT LEDS, uncomment the next line
#define INOUTLEDS 1
```

There are no further changes to make.

Once you have made the required changes, reprogram the controller with the revised firmware.

GENERAL RULE FOR OPTIONS

Do NOT enable an option if the hardware for that option does not exist.

For example, if you enable the OLED display in the firmware and the hardware has not been added to the controller - then you can expect some pretty annoying and frustrating problems to occur.

Important settings are stored in the controller EEPROM; are remembered from session to session; and can be configured using the Windows Application as part of the Initial Setup of the myFocuserPro2E controller.

Values are retrieved from the controller when connecting using the Windows Application or ASCOM driver. Some of these settings can be updated when connecting to the myFocuserPro2E controller.

Note: Stepper Coil power means that at the end of the move, when the stepper motor is stationary, power is either OFF or ON to the stepper motor coils. If OFF, this saves power, but it might mean that a heavy focuser might start to slip if pointed towards zenith (less of a problem if using a geared NEMA stepper motor). To minimize this slippage, set the Stepper Coil Power to ON. The ON setting consumes power when the stepper motor is not moving. Some stepper motors might get hot in operation if this setting is ON. You will need to check your stepper motor if this is happening. The recommended PG27 stepper motor is fine with Coil Power OFF. If using micro-stepping (any stepping mode other than FULL steps) with a NON-GEARED MOTOR then Coil Power should be set to ON.

Note: Remember NOT to use the Focus Lock Screw on your focuser; leave the screw loose or remove it. The Stepper motor will hold the focuser in place. If you leave the screw in and accidently tighten the focus lock screw, then serious damage can occur to the stepper motor or focuser.

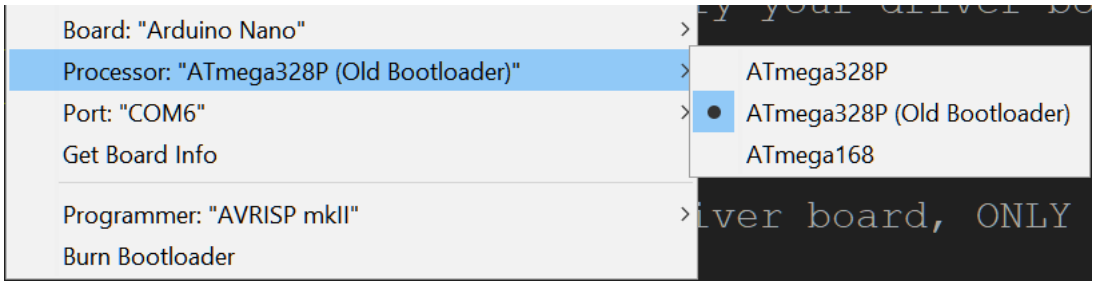
UPLOADING ARDUINO FIRMWARE CODE TO THE UNO/NANO

To upload any firmware to the controller, select the correct board from the Tools > Board menu. Next select the correct serial port from the Tools > Serial Port menu.

Press the upload button  in the Arduino environment. The board will automatically reset and the sketch will be uploaded.

The term “Arduino Firmware Code” refers to the Arduino program (ends in .ino) that contains the focuser code and when executed by the Arduino, makes the chip act like a myFocuserPro2 controller.

Note: Depending on the Arduino Nano chip you are using, with the newer Arduino IDE v1.8.8+ you may need to specify the Old bootloader to reprogram the chip.



CONTROLLER SETTINGS THAT ARE REMEMBERED

Backlash steps in
Backlash steps out
Backlash in enabled state
Backlash out enabled state
Coil power
Delay after move
Display enabled
Display time for each page
Focuser position
Last focuser direction
Maxstep
Motor Speed
Reverse direction
Stepmode
StepSize
StepSize enabled state
Temperature compensation enabled state
Temperature coefficient
Temperature mode (Celsius or Fahrenheit)
Temperature compensation direction

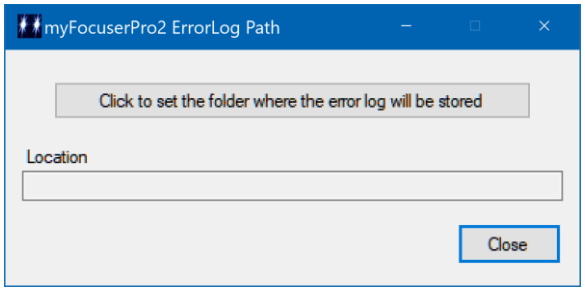
CONTROLLER POWER RESET PREVENTION SWITCH

To reprogram the controller, the power reset prevention switch must be in the OFF position. If the switch is in the ON position, reprogramming the controller will fail with a timeout error.

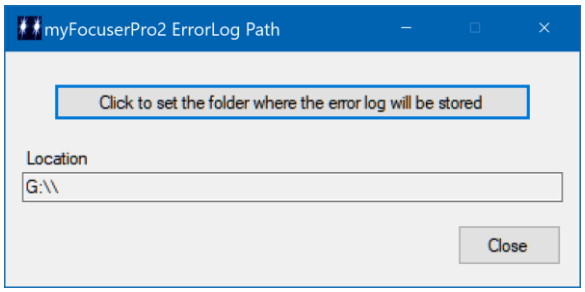
myFP2 WINDOWS APPLICATION

The myFP2 application gives full control of the focuser (supports ALL build options). Note that the majority of settings are remembered/saved by the application or within the controller. The Menu provides access to Exit (quit the program), Settings, Language and About (Copyright message).

The application is arranged as a series of tabs. When first started, the application will ask for the location where data logfiles and error logfiles will be stored.



Click the button and select the desired location. Once selected, the form will look like



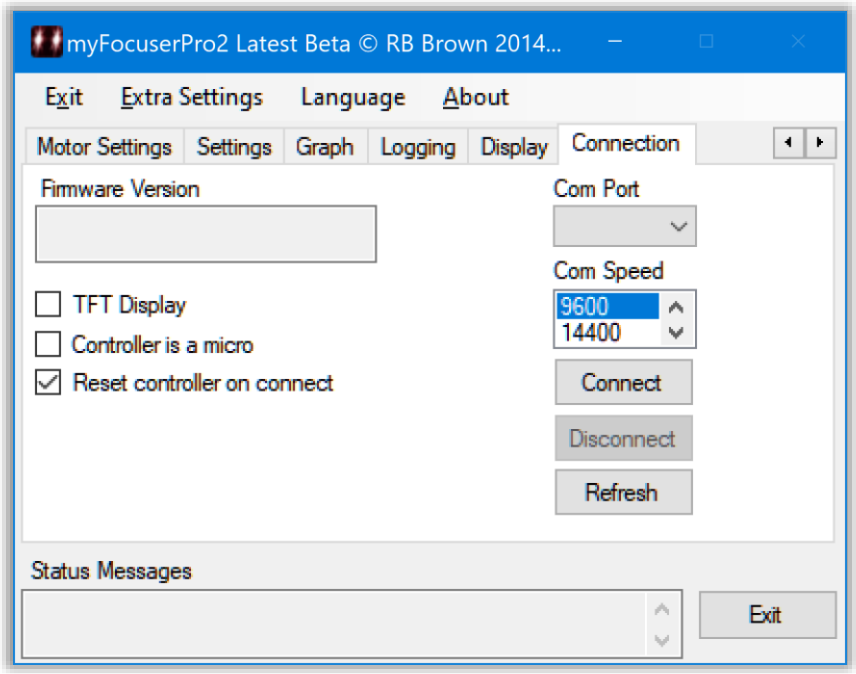
Click the Close button to continue. The application displays the connection tab.

TYPING NEW VALUES INTO TEXT FIELDS

After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.

CONNECTION TAB

The connection tab allows you to specify the serial port [COM port] and speed [9600] to which the controller is connected. Buttons are also provided for Connecting, Disconnecting and Refreshing the list of serial ports.



Select the correct COM port from the drop-down list and select 9600 and the speed. Click Connect to connect to the controller.

Once connection is established, controller parameters will be retrieved and then the Main tab will be displayed.

☑ TFT Display
Do not use for myFP2M

☑ Controller is a Micro
Do not use for myFP2M

☑ Reset Controller on Connect
When checked, will reset the Arduino controller when connecting. The recommended setting in checked so that the Controller starts with a known state.

COM Port - Use the dropdown list to select the correct comport that the myFP2 is connected to

Com Port Speed - Select the baud-rate that matches the firmware in the controller (default 9600)

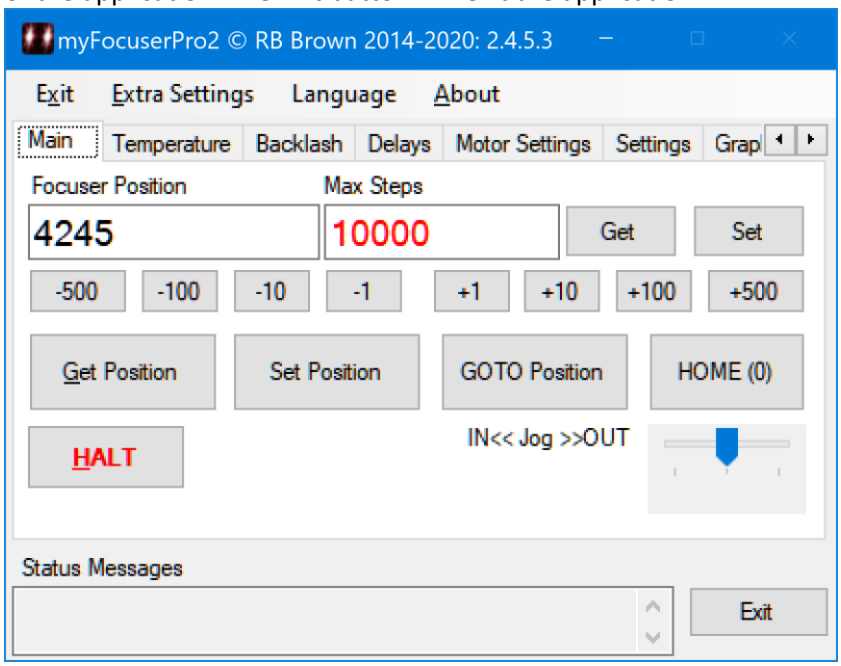
Connect - After selecting the correct comport, click Connect to connect to the myFP2 controller.

Disconnect - disconnects the myFP2 controller.

Refresh - refreshes the list of available comports.

MAIN TAB

The main tab provides basic control of the focus controller, with operations such as move, halt, go and home. The jogging control is also on the main tab. Status messages are displayed in the fields at the bottom of the application. The Exit button will exit the application.



Focuser Position

This text fields displays the current focuser position. This filed can be changed by typing a new value in the text field. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.* If a new value is entered, you can then goto the desired position by clicking the GOTO Position button, or you can change the current focuser position to the new value [without moving the focuser] by clicking the Set Position button.

Goto Position

Type in the desired focus position in the Focuser Position text box (digits only, < maxStep, 0 or > 0) and press the ENTER key to finish the text entry. Next click the GOTO Position button. The focuser will move to the specified position.

Get Position

Returns the current myFP2 controller position in the Focuser Position text box.

Set Position

Type in the desired focus position in the Focuser Position text box (digits only, less than Maximum Position, 0 or greater than 0), press ENTER to terminate the text entry, then click the Set Position Button. The focuser will NOT move but the focuser position is updated.

HOME

The focuser will move to position 0 and stop.

Get Maximum Position

Returns the current myFP2 maxStep value in the Maximum Position text box.

Set Maximum Position
Type in the desired maxStep value in the Maximum Position text box (digits only, greater than FocuserPosition) and terminate the text entry by pressing the ENTER key, then click the Set Button.

HALT
Halts the focuser if currently moving.

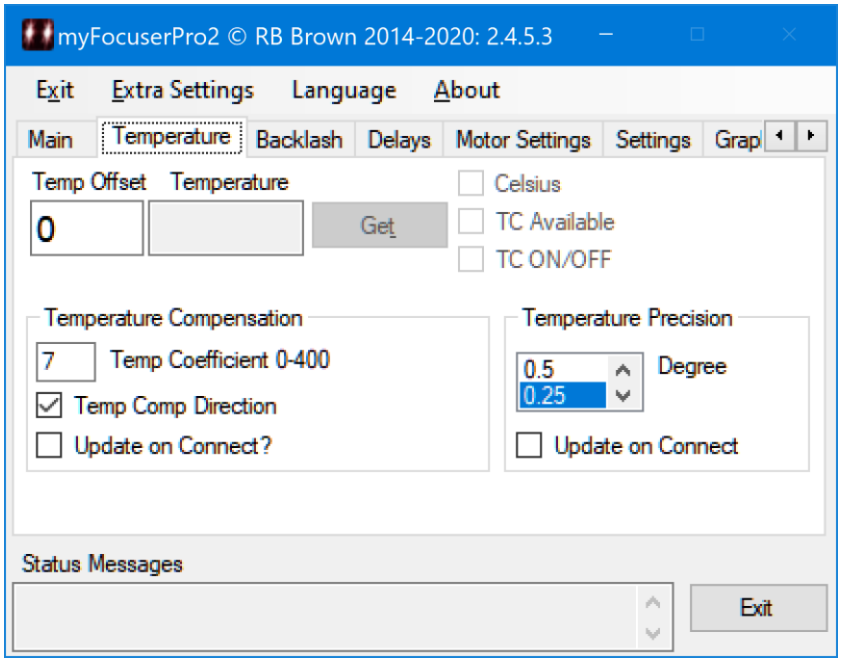
-500 to 500
These buttons provide a means to move the myFP2 controller by a specified number of steps. For example, clicking +10 will move the focuser +10 steps. Moves are not permitted when temperature compensation is enabled.

Jogging
This control lets the user move the focuser under jog control. Dragging the control to the left moves the focuser IN whilst dragging the control to the right moves the focuser OUT. The speed is controlled by the motor speed setting. Jogging continues as long as the mouse is held down or the control is NOT at the midpoint of the control. When the mouse is released, the control is set to 0 and de-activated, jogging stops and the current focuser position is retrieved from the controller. Jogging also stops if the HALT button is clicked. A single click on the control stops the focuser moving by sending a HALT command to the controller.

Status Messages
Text box used to indicate status messages and responses from myFP2 controllers.

EXIT
Exits the application.

TEMPERATURE TAB



Temp Offset

The Temp offset entry-box provides a mechanism for adjusting the returned temperature value for calibration.

Adjustment values range from –3 to +3. For example, typing -1.5 into the entry-box will subtract 1.5 degrees C from the returned temperature value.

A comma or decimal point can be used to signify the decimal point.

To set the temperature offset, click inside the entry-box and type the desired value (for example -1,32) and then press Enter. Once the Enter key is pressed, the entered value is validated (rounded to 2 decimal places and bound checked at -3 and +3) and shown corrected in the entry-box (using a decimal separator of a decimal point).

After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.

Get Temperature

Requests the temperature from the myFP2 controller (if a temperature probe is supported and attached) and display the adjusted value in the Temperature Text Box.

☑ Celsius

When checked displays temperature value in Celsius

☑ TC Available

Updated when connected to a myFP2 controller. If checked, the controller supports temperature compensation

☑ TC ON/OFF

If the controller supports temperature compensation, this checkbox can be used to turn temperature compensation in the controller ON or OFF

Temperature Compensation
Steps per degree (0-400) is the coefficient value, which determines how many steps the focuser will move on a one-degree temperature drop (each focuser is different and this value must be calculated). *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

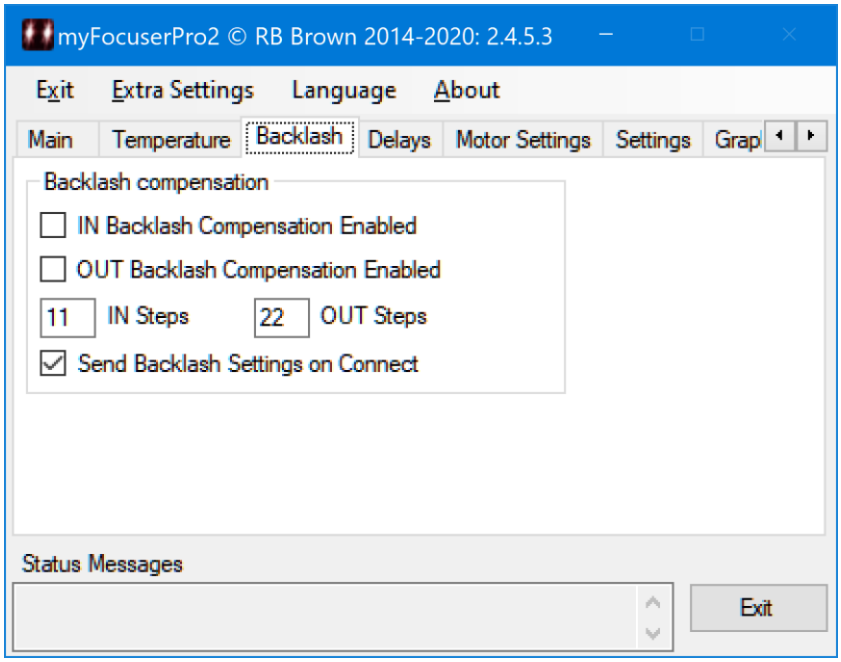
☑ Direction
When checked means that the focuser temperature compensation move will be applied inwards. When unchecked, the focuser temperature compensation move will be applied in an outwards direction.

☑ Update on Connect
When checked means that the entered coefficient value and the Direction value will be sent to the controller when connecting

Temperature Precision
The drop-down box provides a list of the currently available temperature precision settings, from 0.5 degree to 0.06125 degree resolution. The current highlighted value is shown.

☑ Update on Connect
When checked means that the selected temperature precision setting will be sent to the controller when connecting. If unchecked, then the temperature precision setting will be updated once connection to the controller occurs.

BACKLASH TAB



The myFP2 supports backlash for both IN and OUT directions. Backlash is only applied if the move is in the OPPOSITE direction to the previous move, and the number of backlash steps is applied BEFORE the move is executed.

☑ IN Backlash Compensation Enabled
Enable or disable backlash compensation of IN moves

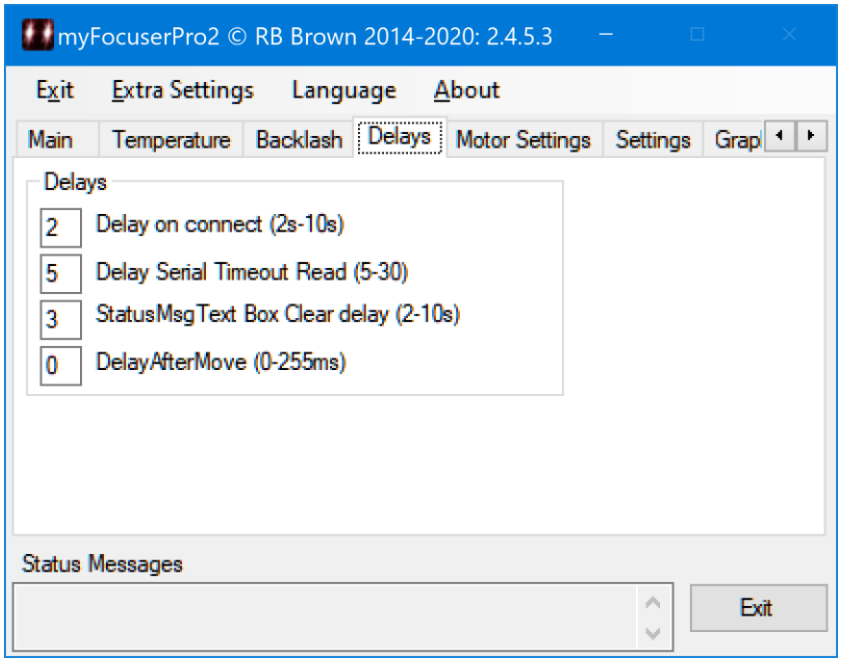
☑ OUT Backlash Compensation Enabled
Enable or disable backlash compensation of OUT moves

IN Steps (0-255)
The number of steps of backlash to apply after the IN move has completed. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

OUT Steps (0-255)
The number of steps of backlash to apply after the OUT move has completed. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

☑ Send Backlash Settings on Connect
Check this box to send the backlash settings to the controller on connect. If unchecked, the values are updated from the controller when connection is made.

DELAYS TAB



Delay (s) 2-10 on Connect
Specify the delay in seconds after connecting that the driver will wait before sending a command to the myFP2 controller. Valid values are 2-10 seconds. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

Delay (s) 5-30 timeout on Serial Read
Specify the delay in seconds that the application will wait when attempting to read from the serial port after sending a command to the myFP2 controller (default = 5). For Bluetooth or slower devices, a value of 8 or 10 may suffice. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

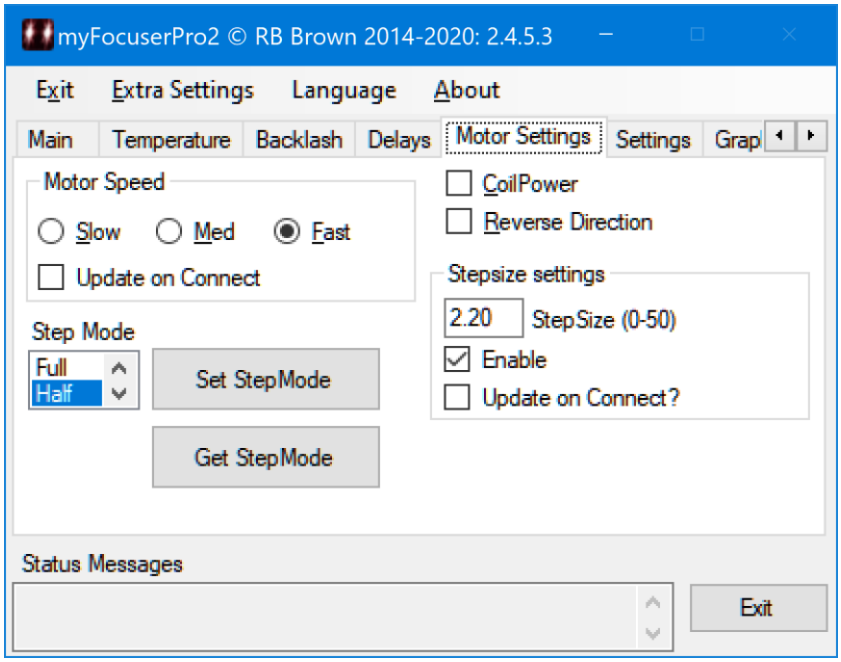
StatusMsg Text Box Clear after delay
After writing text to the Status messages text box, this delay specifies the number of seconds that will elapse before the text box is automatically cleared. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

DelayAfterMove
Specifies the delay in milliseconds (0-255) that is applied after a move, to prevent any blurring of images captured caused by any mechanical residual movement or vibration of the stepper or focuser during focusing.

This is most useful when auto-focusing because the capture time is relatively short in duration.

After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.

MOTOR SETTINGS TAB



Set Step-Mode
To set the Step-Mode, select the desired stepping mode from the dropdown list, then click the Set StepMode button. This should be set only ONCE during the initial setup and after than remain unchanged. Changing stepmode during a session invalidates the focuser position.

Get Step-Mode
Click the Get StepMode to get the current stepping mode from the myFP2 controller

Motor-Speed
This controls the relative speed of the stepper motor. The speed can only be changed when the focuser is NOT moving. If the Update on Connect option is checked, the motor-Speed is sent to the controller when connecting, otherwise the speed setting will be updated after connection to the controller occurs.

☑ Coil Power
When ON coil power is ON [checked] and the stepper coils remain powered after the move is completed

☑ Reverse Direction
When ON [checked] the motor moves in the opposite direction (IN means OUT and OUT means IN)

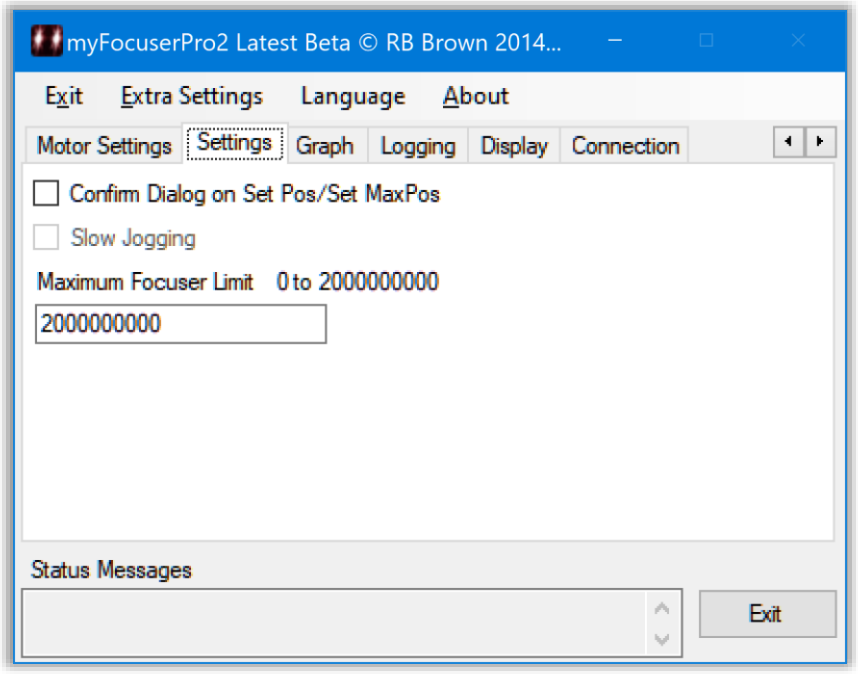
Step Size in microns (0-50)
Defines the StepSize in microns. You can enter your measured / calculated value for your focuser and this will be sent to the myFP2 controller when connecting. This allows client applications using the ASCOM driver to retrieve the StepSize setting from the myFP2 controller. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.* Please see the section in this PDF on determining your focuser step size value.

☑ Enable in controller?

Turns ON or OFF the reporting of the StepSize value from the myFP2 controller. If you do not know your step size value, DO NOT enable this feature.

☑ Update on Connect
Enable the checkbox to send these values to the myFP2 controller on Connect.

SETTINGS TAB



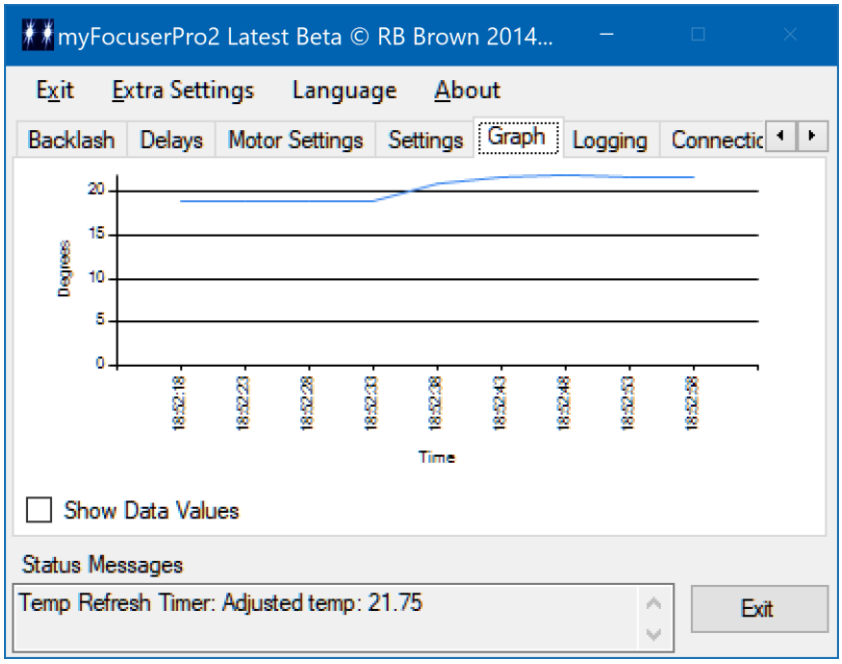
Options MUST BE SET before connecting to the focuser. The settings will be remembered by the application and next time the application is run, these settings will be restored

☑ Confirm Dialog on Set Pos/Set MaxPos
When checked, will display a confirm (Yes/No/Cancel) dialog box when the user clicks on the Set Focuser Position and Set Max Position buttons. Click Yes to confirm and update the value, or click No/Cancel to abort the change and restore the previous setting.

Maximum Focuser Limit
Specify the maximum number of steps for the focus controller and will vary depending upon each user's configuration. You should set this to the maximum number of steps that you have calculated for your focuser [see determining maxStep]. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.* This setting must be specified before connecting to the controller, and will be remembered by the application. See the section on the initial setup of the focuser for further information.

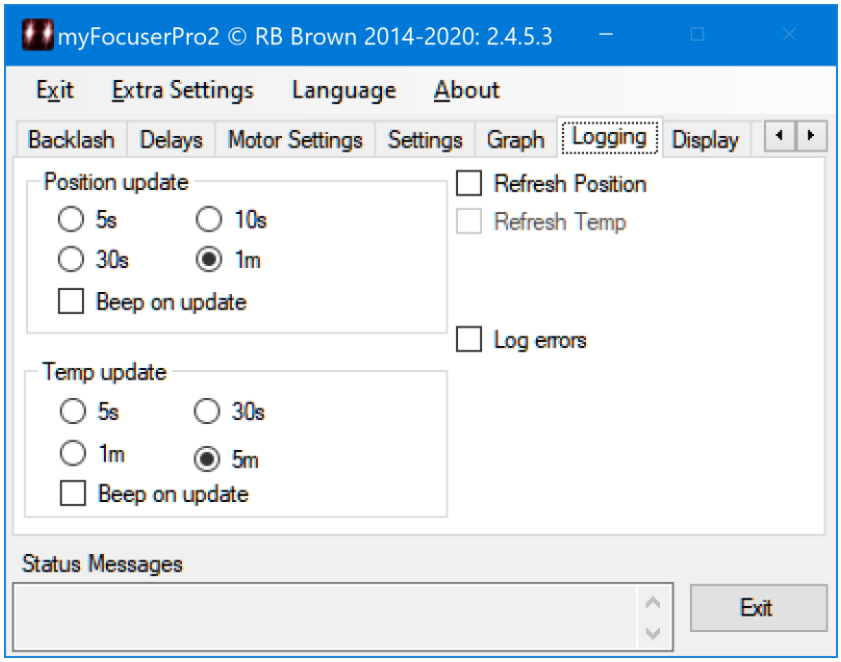
☑ SlowJogging
When checked enables super slow jogging

GRAPH TAB



If refresh temp is enabled [on the Logging tab] then a plot of the measured temperature will be displayed and updated at regular intervals.

LOGGING TAB



☑ Refresh Temp
When ON [checked] the application will periodically poll the myFP2 controller and request a temperature update. The polling interval is set under the settings menu

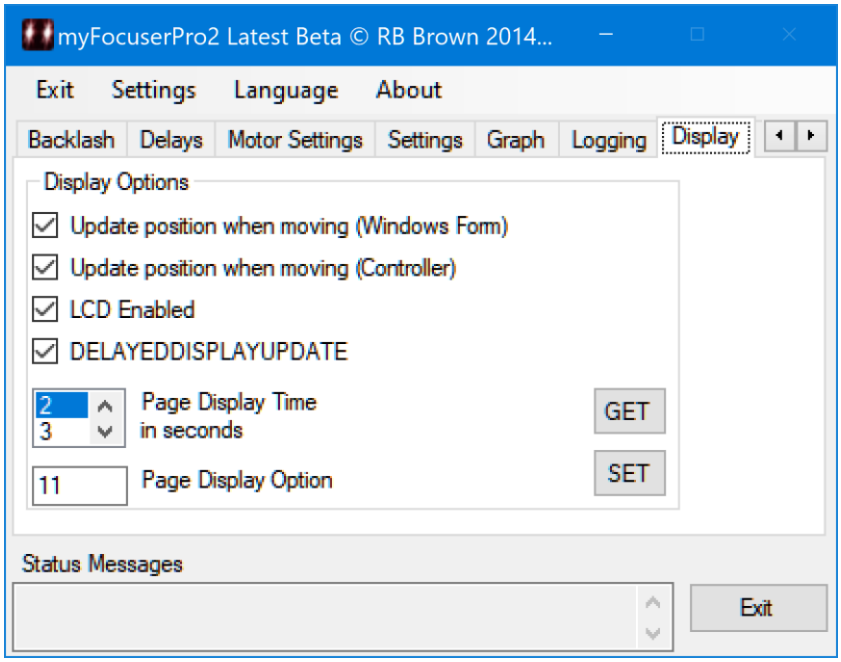
☑ Refresh Position
When ON [checked] the application will periodically poll the myFP2 controller and request a position update. The polling interval is set under the settings menu

Refresh Position
Select the desired (5s, 10s, 30s, 1m) refresh interval for getting the current focuser position from the controller (when connected).

Refresh Temp
Select the desired (5s, 30s, 1m, 5m) refresh interval for getting the current focuser temperature from the controller (when connected). In the Windows app we now have a separate timer for Position and Temperature. For Temperature compensation, you must set the temperature coefficient AND start the Refresh Update timer BEFORE enabling temperature compensation.

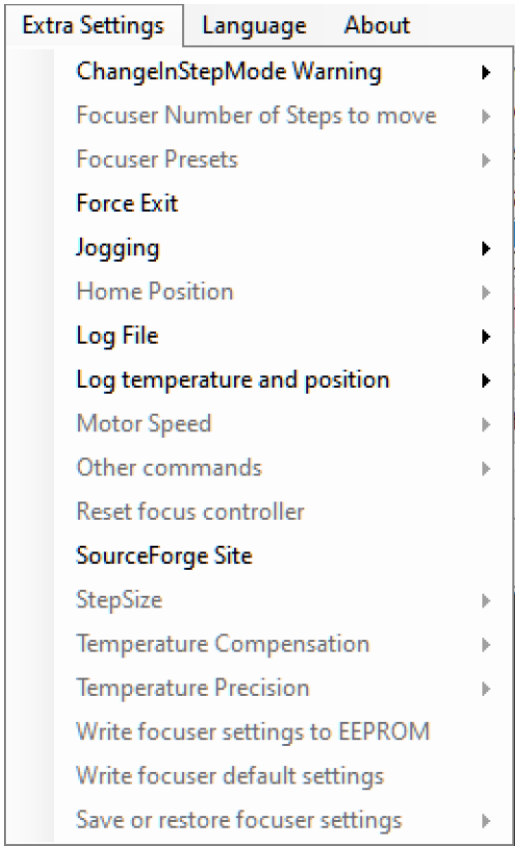
☑ Log Errors
To generate a log file for debugging and trouble-shooting purposes, check the Log Errors checkbox.

DISPLAY TAB



Do not use for myFP2M

THE SETTINGS MENU



Backlash Compensation
A set of options to query the backlash settings or send settings to the controller

ChangeInStepModeWarning
Enabled or disabled. When enabled, it warns the user about changing the StepMode when the focuser is connected.

Focuser Number of Steps to move
When "Settings>Double Step Size" is selected from the menu bar, the step button values are doubled (-500 becomes -1000). The double step size setting is NOT remembered by the application program.

Focuser Presets
This allows you to specify up to FOUR preset focuser positions and move the focuser to any of the four preset positions. These presets are saved by the application.

Force Exit
In the event of problems, this provides a clean method of exiting the program **Jogging** - Get the jogging status and the last jogging direction

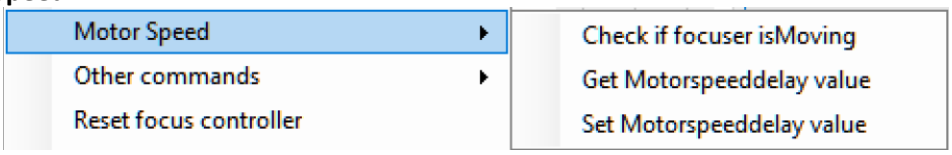
Get Controller Firmware Version
Get the firmware version of the connected myFP2 controller

Home Position
Do not use for myFP2M

Log File
Reset Error Log File path displays a dialog box to specify the drive and directory where the error log file and other log files are to be stored.

Log Temperature and Position
Save the temperature or focuser position to a log file in automate mode.

Motor Speed



Check if focuser isMoving
Gets the moving status of the controller

Get/Set Motorspeed value
Click here to learn more about motor speed values

Reset Arduino Focus Controller
This will restart the Arduino focus controller. Please wait 3-5s before issuing any command.

SourceForge Site
Opens a web browser and display the myFP2 website where you can download the latest drivers, software and documentation.

Step Size
This menu has two options – "Check if StepSize is enabled in the controller" queries the controller and displays if StepSize is enabled, and "Get controller StepSize" displays the current controller setting for StepSize.

Temperature Compensation
Please read the section on temperature compensation.

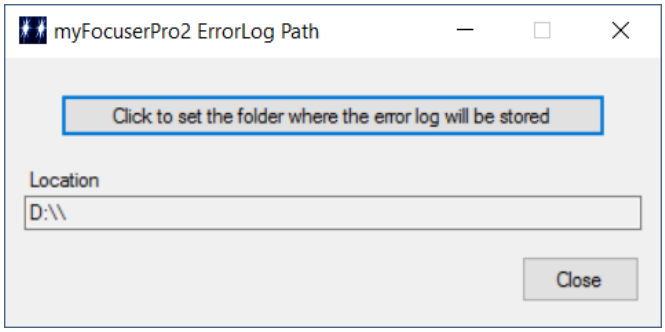
Temperature Precision
This setting allows the user to control the resolution of the DS18B20 temperature probe from 0.5 degrees down to 0.0625 degrees. This setting is remembered by the controller.

Write Focuser Settings to EEPROM
Save the controller settings in EEPROM.

Write focuser default settings
Used to write a standard configuration setting to the myFP2 controller. The default configuration is specified within the Arduino firmware file. This may be useful after upgrading the firmware file to a newer version. For more information, click here.

Save and Restore Focuser/App settings
Provides for the saving and restoring of focuser and application settings. This is designed to be used before and after upgrading the Windows application and myFP2 controller firmware.

THE ERRORLOGPATHNAME FORM

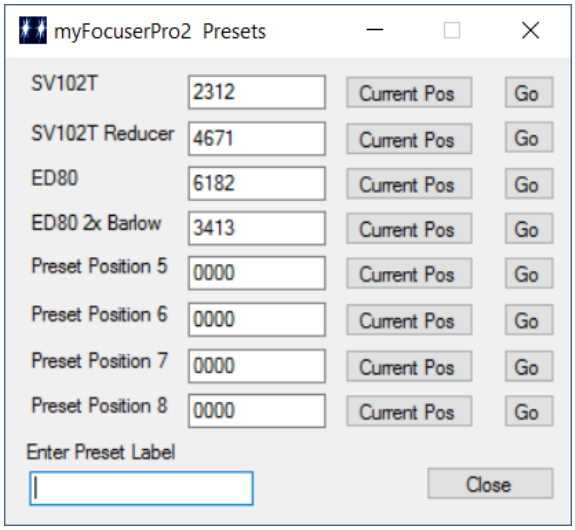


You can specify the directory/folder where the error log file and other settings files will be stored. The path is remembered by the application. This dialog box is accessed from the menu bar Settings>Log File>Reset Error Log File Path

When the application is first installed, this path is set to NULL, so that when the application is run for the first time, this dialog box appears at start-up for the first time.

THE FOCUSER PRESETS SETTINGS FORM

Selecting the "Enter Presets" option from the Focuser Presets menu under the Settings displays the following form



You can enter up to 8 preset focuser positions. The values are only checked when a focuser move is initiated (if the value is wrong then a message will be displayed in the Status messages textbox and the move cancelled).

Typical values must be greater than 0 and less than maxStep. To copy the current focuser position to a preset, click the associated **Current Pos** button.

Once you have entered the preset values, click the **Close** button to close the form.

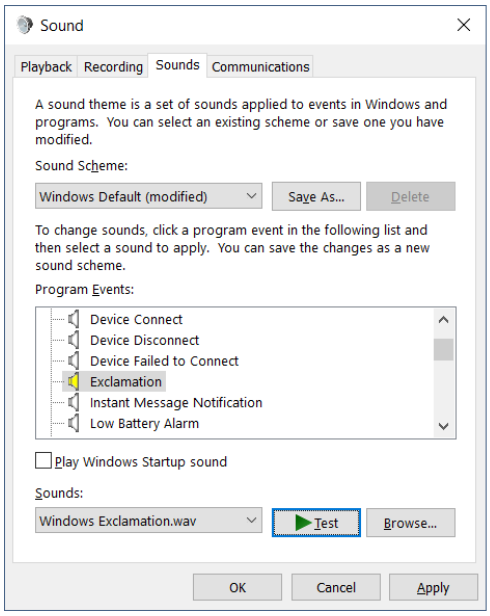
To change a label, first type the new text into the Enter Preset Label textbox (limit of 17 characters) then click on an existing label to replace that labels text.

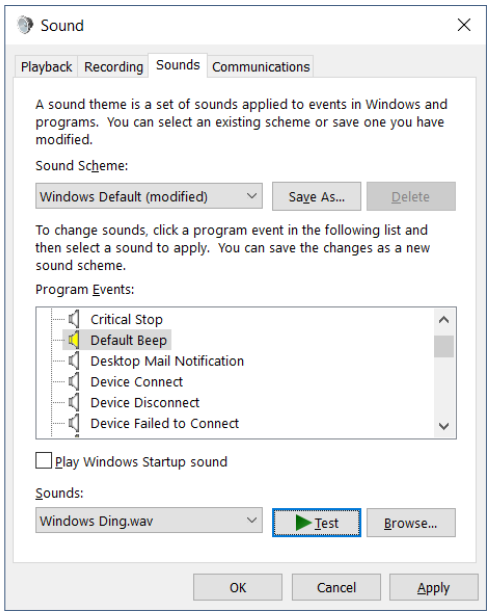
To command the focuser to move to a preset value (assuming one has been set), use the Settings>Focuser Presets menu, as shown, and clicking on one of the presets

Using the Refresh Timer Sounds

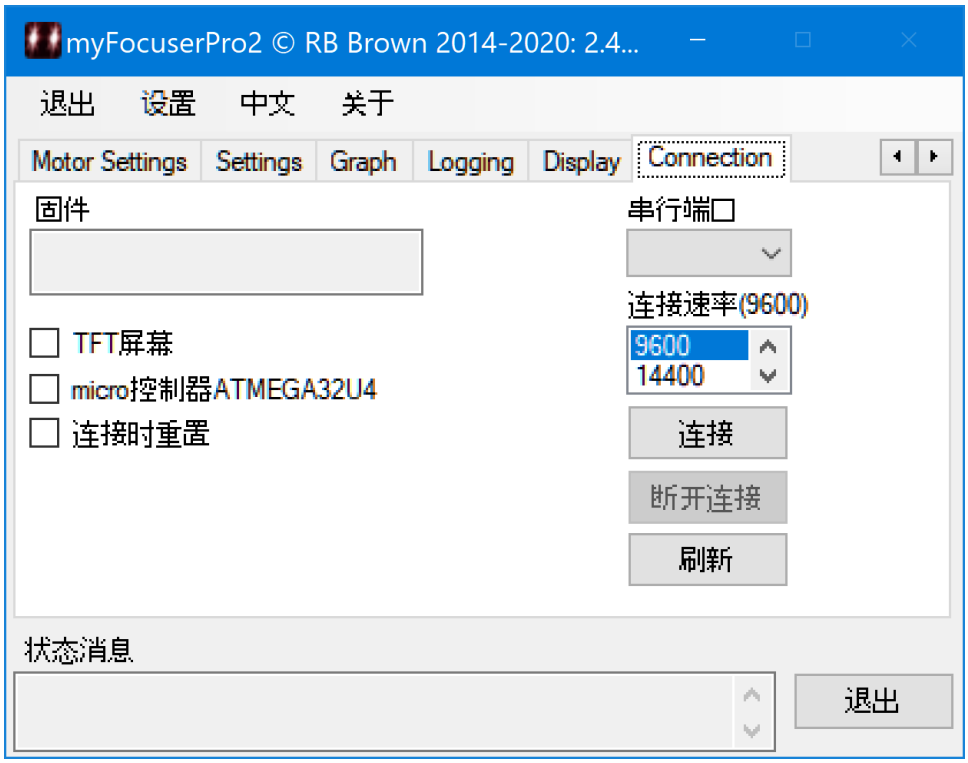
With Windows, the refresh timers can generate system sounds when the update occurs. For the Temperature refresh timer, the Windows system sound is Beep, and Exclamation is used for the Position refresh timer. The sound for each refresh timer is independent and can be enabled or disabled.

These sounds can be configured in the Windows environment by accessing the Sound control.





LANGUAGE SUPPORT

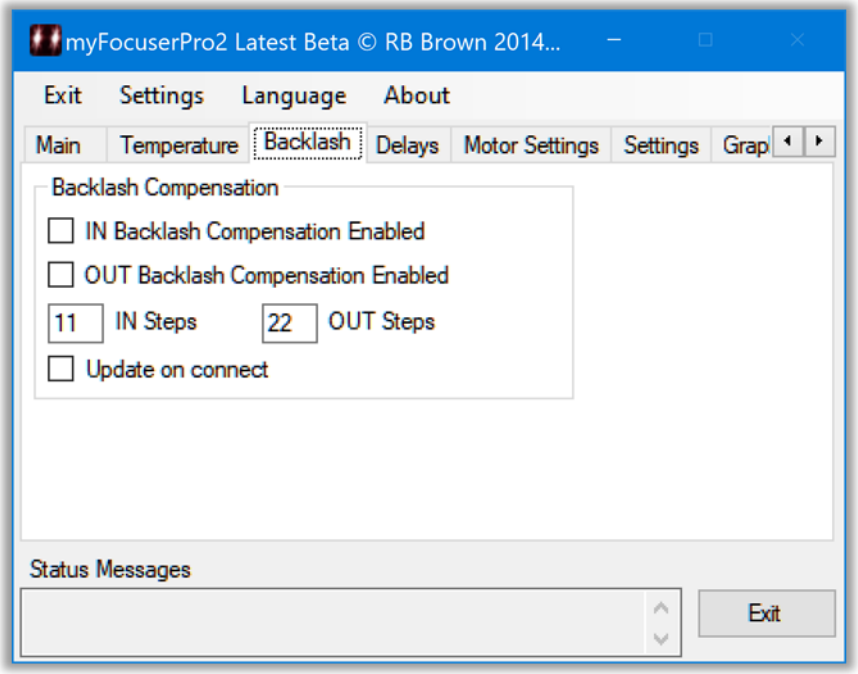


Current supported languages are

- Chinese (Simplified)
- Czech
- English
- French
- German
- Greek
- Italian
- Polish
- Russian
- Spanish

Other languages can be added. Translations have been done using online translation sites and may not be a true or accurate translation.

BACKLASH SETTINGS



NOTES ABOUT BACKLASH

Backlash is only applied if

1. Backlash is enabled in both the settings and firmware, and
2. The new focuser direction is not the same as the previous focuser direction

The backlash compensation in myFP2 works as follows.

1. When a move is requested, the direction of the move is determined to see if it is the opposite direction to the previous move.
2. If the answer is yes, then (if backlash compensation is enabled for the direction being requested) the focuser firmware performs the backlash steps without adjusting the target position (but ensures that it does not exceed the boundaries of 0 and maxSteps).
3. The focuser then performs the move to the desired target position

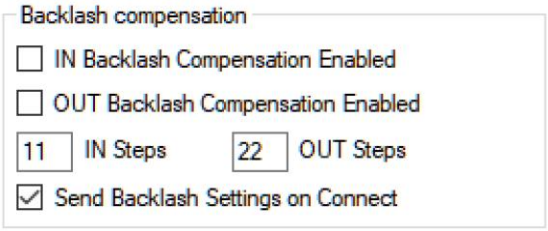
Because the backlash is implemented in this way, the final position will end up being the same position as that requested by the move.

How to determine and set Backlash

To determine backlash, you need the controller fitted to the telescope, with all the controller settings as per normal usage. This can be done during the day.

You will need to use the Windows application to perform the tests below. Remember that backlash is only applied if enabled and the direction of move is opposite to the previous move.

It is best to use a dial indicator (or digital caliper) to determine if the focuser has moved. These can be setup on the end of the focuser lip, eyepiece, imaging camera etc to give an indication of movement. If you do not have one, you could mark a small dot on the larger focuser knob or gear.



Once you have the dial indicator positioned correctly, start the Windows Application (ensure 12V is connected and turned on else the stepper will not move).

Open the Backlash tab and set the backlash as indicated on the left.

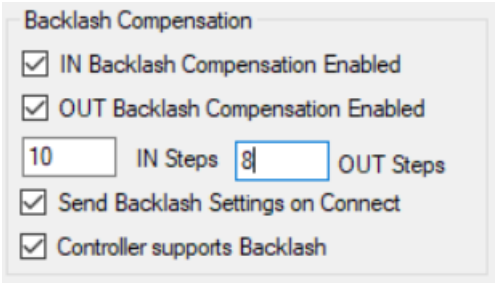
Click on the Connection tab and connect to the controller.

Determine the amount of IN backlash

1. Move 100 steps OUT (100+)
2. Ensure that the dial indicator is positioned correctly
3. Note the focuser position
4. Move 1 step IN
5. After each single step, check the reading of the dial indicator to see if there has been movement
6. If there has been no movement, go back to step 4 and move IN again by one step
7. When you have detected movement, this means the backlash has been overcome. Note the focuser position and the difference between the position at step3 and now is the IN backlash steps
8. Remember this value (write it down, my value was 10)

Determine the amount of OUT backlash

1. Move 100 steps IN (100-)
2. Ensure that the dial indicator is positioned correctly
3. Note the focuser position
4. Move 1 step OUT
5. After each single step, check the reading of the dial indicator to see if there has been movement
6. If there has been no movement, go back to step 4 and move OUT again by one step
7. When you have detected movement, this means the backlash has been overcome. Note the focuser position and the difference between the position at step3 and now is the OUT backlash steps
8. Remember this value (write it down, my value was 8)



Disconnect from the focuser. Open the extra settings screen. Enter your values then click Close.

Connect to the focuser to send the values to the controller.

Backlash and Client Applications

If the client application supports backlash compensation, it is recommended to enable the backlash compensation in the firmware and DISABLE the backlash compensation in client application.

DO NOT have backlash compensation in both the client application and the firmware enabled at the same time.

Backlash and ASCOM drivers

As ASCOM provides no means for handling backlash, the settings will only be able to be configured on the extra settings form from the setup-dialog form for ASCOM (in the same way that MotorSpeed and other values are handled). Any backlash setting can then be sent to the controller when connecting and would then be set and remembered.

Once you have setup the correct backlash values in the controller, you can UNCHECK the *Send Backlash Settings when connecting to controller* checkbox.

Do Not Be Fooled

Do not be fooled by trying to determine backlash visually. Inherently you will get different and inconsistent values. You must make sure that the focuser is outside the critical focus.

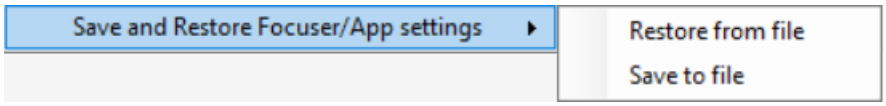
The original goal was to get around 10 steps within the critical focus zone. If you try to determine backlash when the focuser is within the critical focus zone, you may need up to 10 steps or more before any visual indication of change occurs. The exact amount of backlash steps cannot be determined.

Determining backlash using the visual approach is fraught with inherent issues and should be avoided.

THE SETTINGS SAVE/RESTORE OPTIONS

This option has been added as an aid in upgrading the firmware or windows application. Users should not use this option for saving settings between sessions as this is already automated.

These options are designed to work with upgrades to the Arduino firmware or Windows application.



Restore from file - Restores the previously saved myFP2M controller and Windows application settings from a text file in the specified folder. After restoring the settings, the application will save the myFP2M controller settings to EEPROM, reboot the myFP2M controller, and exit the application. Next time the application is started the restored settings will be in effect.

Save to file - Saves the current myFP2M controller and Windows application settings to a text file in the specified folder (the name of the file is automated)

IMPORTANT SETTINGS

By now you should have calculated

- **StepSize** in microns
- **Critical Focus Zone**
- **Step mode** you will be using (Full, Half etc) in order to get around 10 stepper motor steps within the critical focus zone
- **maxStep** being the maximum focuser position

In operating the focuser, you will need to determine the correct settings for **maxStep** that matches your focuser and type of stepper motor being used.

DRV8825/ULN2003/L298N STEP MODE

Step mode is controlled by the application and remembered by the controller.

Once the step mode is set, it should not be changed

MAXSTEPS

To determine **maxStep**, we assume that the start position [0] is ½ a turn out, and we will drive the focuser OUT but stop one half turn before the maximum stop of the focuser. For my focuser this is three full turns of the focuser knob. I am using a myFP2M-DRV8825 controller.

We position the focuser so it is ½ turn out from fully in. We then clamp the coupler so the focuser motor is connected to the focuser shaft. In the myFP2M Windows Application we enter 0 as the current Position and click the SET button to send this value to controller.

The step mode [1/8] I have already set in firmware and reprogramed the controller so I do not need to set that.

I am using a NEMA14 0.9° stepper motor at 1/8 steps. This means my stepper motor has 3200 steps per revolution. As the focuser can travel 3 full revolutions this means there will be a maximum of 3200*3 steps, so we set maxStep to 9600 in the setup dialog box.

In the myFP2M Windows Application we enter 9600 as the Maximum Position and click the SET button to send this value to controller.

For the initial focuser position, we determine the half-way point (0-9600) and so the initial focuser position will be 4800.

Next, we move the focuser from position 0 to the mid-point by entering 4800 as the focuser position and then click the GOTO POSITION button to move the focuser. Once the focuser has stopped moving, we can then wait 30s before closing the application and powering off the focuser. *If you notice that the focuser does not move when the GOTO POSITION command is sent to the controller, it is likely that the direction is incorrect. Try enabling Reverse Direction and then clicking the GOTO POSITION button again.*

To check that everything is set, we turn on the focuser and restart the myFP2M Windows application. You will see that the focuser position will be set to the midway point (in our example 4800) and that the maximum position is set to 9600.

As long as the focuser is not manually moved, or the coupler disconnected, the focuser is now setup with the correct values. Each time we connect to the focuser, the correct settings will be sent to the controller and will be saved so they can be recalled next time we run the software or access the ASCOM driver.

In order for the focuser to work correctly, incorrect values for Maximum Position or setting the zero position incorrectly may cause damage to either the focuser or stepper motor.

Please note that the values will be different for your focuser and these will need to be determined by you in order for the focuser to work correctly. Incorrect values for Maximum Position or setting the zero position incorrectly may cause damage to either the focuser or stepper motor.

It is important that the stepper motor stops and does not try to drive past the minimum and maximum points of your focusers travel.

STEP SIZE

Step size is the amount in microns that the focuser travels for a single step. The controller supports the implementation of step size (ASCOM driver can return the step size if enabled, else the driver returns a not implemented exception).

Be aware that there can be no common value for this as each implementation is different, depending upon the step mode, stepper motor, gearing and connection to the focuser.

If you have some software that needs step size (like Maxim DL), then you will need to calculate the correct value and then use that value in the software application (like Maxim DL) and specify it for your controller.

You can only calculate the step size once your focuser is fully setup.

Note that if you change the stepping mode then the step size will also change. So, the best thing to do is use one step mode and never change the step mode!

To calculate the step size, position the focuser at say 1000 steps. If your focuser has indicator marks note the position. If the focuser does not have any position marks, try to use an electronic calliper to measure how far out the focuser is and use that position as 0. Now you will send a command to move the focuser outwards 1000 steps from its current position. Once the focuser has moved to the new position, take another measurement, and subtract the first measurement from it.

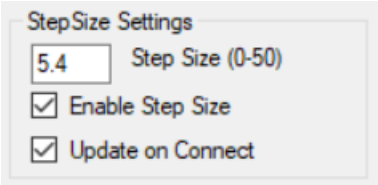
If the first reading was 62mm and the final reading was 87mm, then the distance the focuser actually moved for 1000 stepper motor steps was 25mm. To calculate the step size, divide the distance in microns (to convert a millimetre to a micron multiply by 1000) by the number of steps

25 * 1000 / 1000
= 25000 / 1000
= 25 microns

Note: 1mm = 1000 microns.

The controller implements bounds checking for the value of Step Size, which restricts the Step size to > 0 and < 50

The settings for StepSize and whether it is enabled in the controller is specified in the Extra Settings form of the Windows Application or the Settings form for the ASCOM driver.



To enable stepsize in the controller, you must specify the step size value, check the Enable Step Size box and check the Update on Connect box before connecting to the controller.

STEP SIZE IN SGPRO

SGPRO has a "step size" setting in the focuser properties. This is **NOT** the same as the focuser step size.

To calculate the "step size" value for SGPRO

- Adjust for best focus
- Set the number of focus points to 9
- Note focuser position (example 12321)
- Note HFR value (example 1.9)
- Cycle subs continuous (image capture) - Move focuser out till HFR value is 3 to 5 times the best HFR value (8)
- When this is reached note focuser position (example 12571)
- Deduct original focuser position from current focuser position (12571-12321=250)
- Step size = steps moved (250) * 2 / focus points (9) - 1
- Step size = (250) * 2 / 8
- Step size = 500 / 8 = 55
- Stop subs
- Open equipment profile, focuser section, enter in "step size"

STEPSIZE IN APT

In APT under GEAR, Stepsize does not refer to the stepsize of the controller. In APT stepsize refers to when APT issues a step, how many steps does the focuser move. In our case it should be set to 1.

In focus-aid settings,
FWM Threshold - take about 4-5 images and measure the FWHM. Let's say it came as 4.5, 4.65, 4.4 and 4.45

This means that FWM is varying through seeing by 0.25 so you would set this threshold to a little higher than the variation, say 0.3 or 0.35. Note that this will vary from night to night depending upon seeing.
Fine step - this is closely related to how many steps there are in the CFZ. If you calculated say 10 steps in CFZ then I would set this to maybe slightly less like 7 or 8.

Coarse step - set this to at least fine step * 2, a good number should be number of steps in cfz * 2 (rounded to whole number)

Max moves - 100 should be fine.

NOW YOU NEED TO BE NEAR FOCUS WHEN RUNNING THE AUTO-FOCUS AID - within say 50 steps or so of best focus, unless you change Max Moves.

Note: This information has been gleaned from user feedback and the authors own testing. This might not apply to your particular imaging setup, but may serve as a starting point. All feedback most welcome.

INITIAL SETUP OF YOUR CONTROLLER

To use your controller correctly, you will need to change a few settings first. The settings are best changed with the Windows application and these settings are

Focuser Position
Max Steps
Step Size
Stepmode

SET FOCUSER TUBE ALL THE WAY IN AT 0 POSITION

Move the focuser tube so it is about ½ a turn out from the fully in position.
Affix the stepper motor to the focuser and secure the motor in place, fully engaged.

INSTALL THE WINDOWS APPLICATION

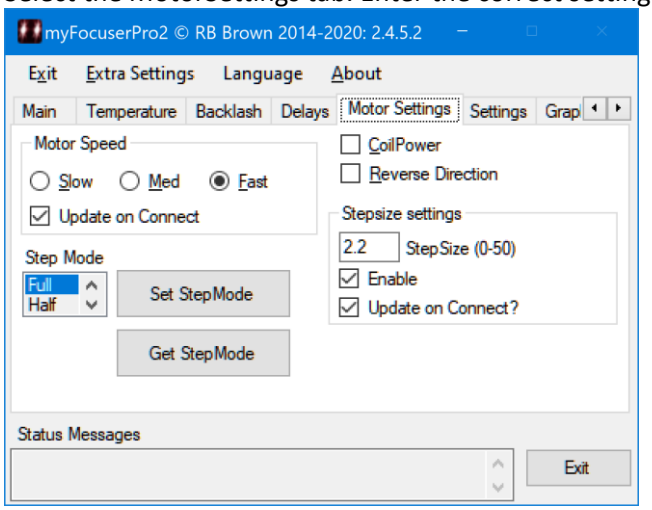
Within the Windows application, the parameters for Step-Size and maxStep (maximum permissible) are set on the Extra Settings menu and *must be set before connecting to the controller*.

SET STEPSIZE AND MAXIMUM VALUE FOR MAXSTEP

a) With the stepper motor disconnected, move the focuser to the in-most position as position 0 (suggest you position the focuser at ½ turn outwards and make that 0)
b) Next clamp the stepper motor in place and engage the stepper with the focuser shaft (tighten screws on clamp etc)
c) Start the windows application BUT do NOT connect to the focuser yet

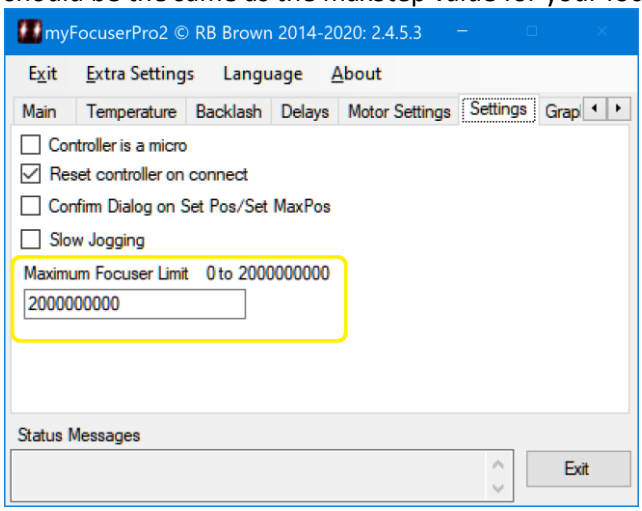
SET STEP SIZE

Select the MotorSettings tab. Enter the correct settings for step size.



SET MAXIMUM FOCUSER LIMIT

Select the Settings tab. This displays the following (note that your values can be different). In all cases, this should be the same as the maxStep value for your focuser.

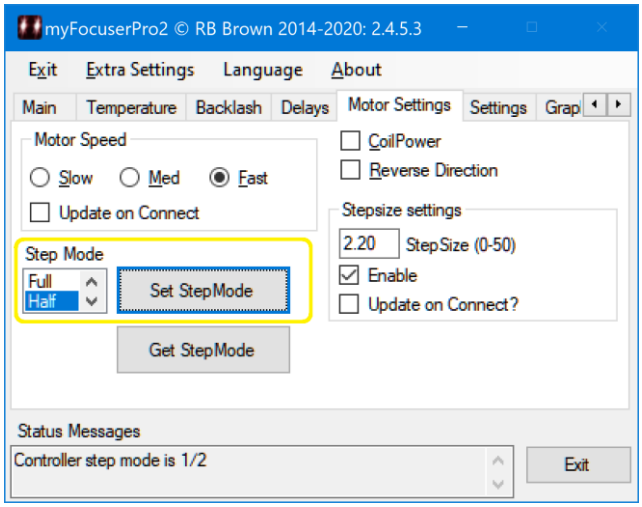


CONNECT TO THE CONTROLLER

Select the Connection tab. Select the correct COM Port and Port Speed [9600] then click the Connect button to connect to the controller.

SET STEP MODE

When the controller connects, the step mode is retrieved from the controller. Select the Motor Settings tab. From the drop down list, choose the correct step mode, then click the Set Step Mode button to update the controller. The step mode is now saved in the controller and will be remembered.

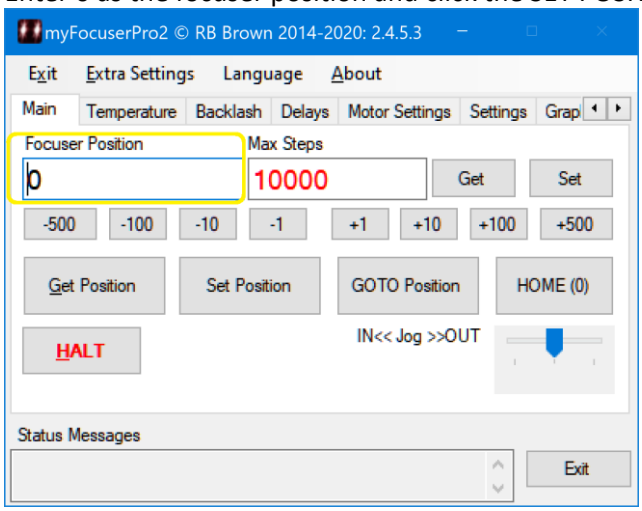


The maximum value of step mode is 128. However, only the RAPS128 and ST6128 driver boards support this level. Each driver board type has its own limit for micro-stepping. If you try to send an invalid step-mode to any driver-board it is ignored by the controller firmware. For instance, sending a step-mode of 64 to a DRV8825 controller will cause the controller to ignore the request and the controller will use the default of half-steps.

For a L298N or ULN2003, the available stepmodes are Full and ½.

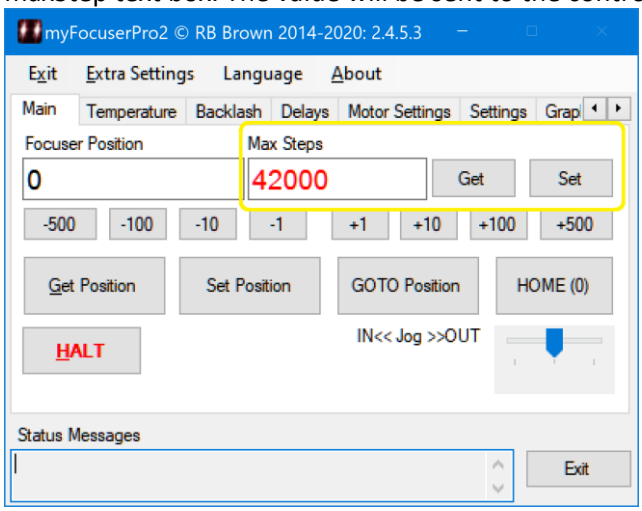
SET FOCUSER POSITION TO 0

Enter 0 as the focuser position and click the SET POSITION button to send the command to the controller.



SET MAXSTEPS

To set the maxStep setting, enter the maxStep value and then click the Set button to the right of the maxStep text box. The value will be sent to the controller and remembered for later use.



CHECK FOCUSER DIRECTION

Now that the focuser has the correct values, you can test the direction setting to ensure the focuser is moving in the correct direction. Reverse direction is on the Motor Settings tab.

This assumes that you have set the stepper current correctly if using a DRV8825/EasyDriver/RAPS128 driver board.

a) In the focuser position text box, enter 0 as the current focuser position and click the SET POSITION button to send the position to the controller
b) Ensure that external power is ON to the stepper motor. The focuser is currently at position 0
c) Click the +100 button
d) If the focuser does not move at all, then click the Reverse Direction button to enable that setting, then click the +100 button again
e) The focuser should move 100 steps outwards
f) Clicking any + button should move the focuser outwards and any – button the focuser should move inwards towards 0

SET COIL POWER

If you are using **microstepping** then Coil Power should be left ON. This is because with micro-stepping the stepper motor can only hold its position if current is flowing in the stepper motor coils. If coil power is OFF, then the stepper will move to the closest full step, and over time this results in the real focuser position not being accurate. Coil Power is on the Motor Settings tab.

The controller will remember stepsize, maxSteps, stepmode, coil power, reverse direction and focuser position.

You can also set other defaults such as Motor Speed, LCD Display Time and other settings at this time.

FROM THIS POINT ON, DO NOT CHANGE THE STEP MODE OR ENTER A NEW FOCUSER POSITION AND CLICK SET POSITION AS THIS WILL ALTER THE CONFIGURATION OF THE STEPPER AND RESULT IN LOSS OF ACCURACY OF POSITION AND ALSO POSSIBLE DAMAGE TO THE FOCUSER OR STEPPER MOTOR.

If you need to set up backlash [on the Backlash tab], consider doing this once you have your focuser configured and working. Please see the section on backlash to determine how these settings can be determined and configured.

CONGRATULATIONS: Your focuser is now ready to use!

MYFOCUSERPRO2-ANDROID-BLUETOOTH

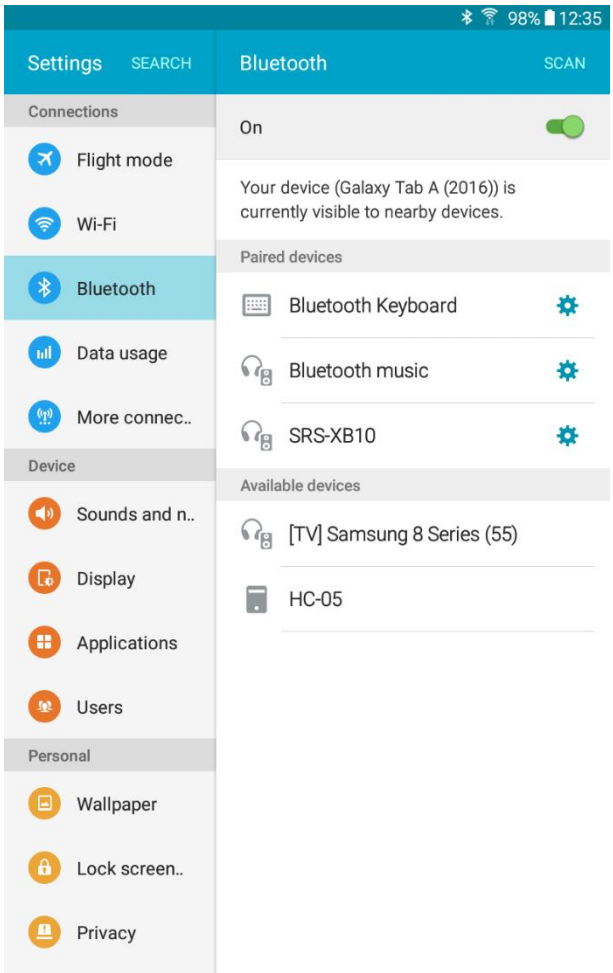
The provided Android application was developed and tested on a Samsung S9+ and Samsung Table A4.

Tips for best operation before starting the application

1. It is best to start with the Bluetooth enabled
2. Turn off screen rotation
3. Disable any screen saver or screen display timeout
4. Disable any running apps which might interrupt the application (such as email, notifications etc)
5. As GPS and Bluetooth consume a lot of battery power, it is best to have the phone connected to a charger when using this application

myFocuserPro2 Bluetooth Android App
https://www.youtube.com/watch?v=I7mUmHR_bx8

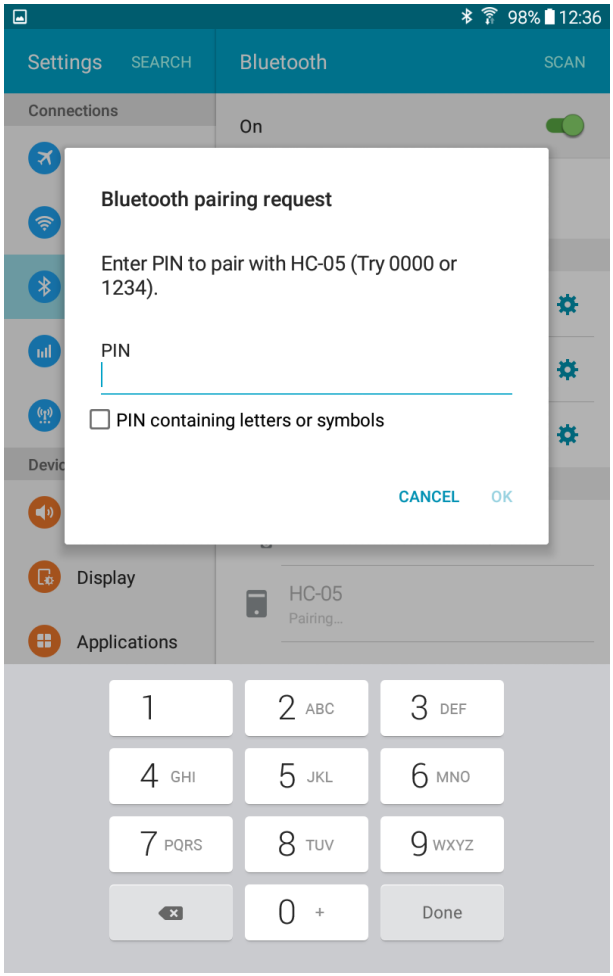
HOW TO PAIR THE BT CONTROLLER - 1



To first setup the HC-05 adapter to be recognized and paired to the Android device, turn on the Bluetooth enabled adapter.

Open the Bluetooth settings of the Android device and you should see a device which is a series of digits separated by colons or might appear as HC-05.

HOW TO PAIR THE BT CONTROLLER – 2

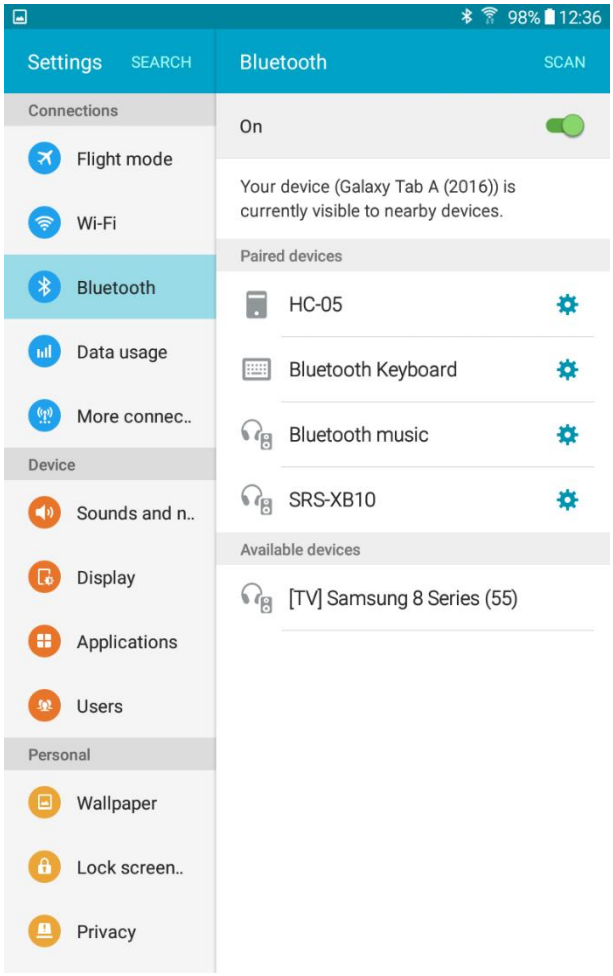


If you tap on the HC-05 Bluetooth device, a pairing request will be established and you will need to enter a pin number to pair the HC-05 device to the Android device.

The pin number is 1234

After entering the pin number, tap the OK text on the dialog box.

HOW TO PAIR THE BT CONTROLLER - 3



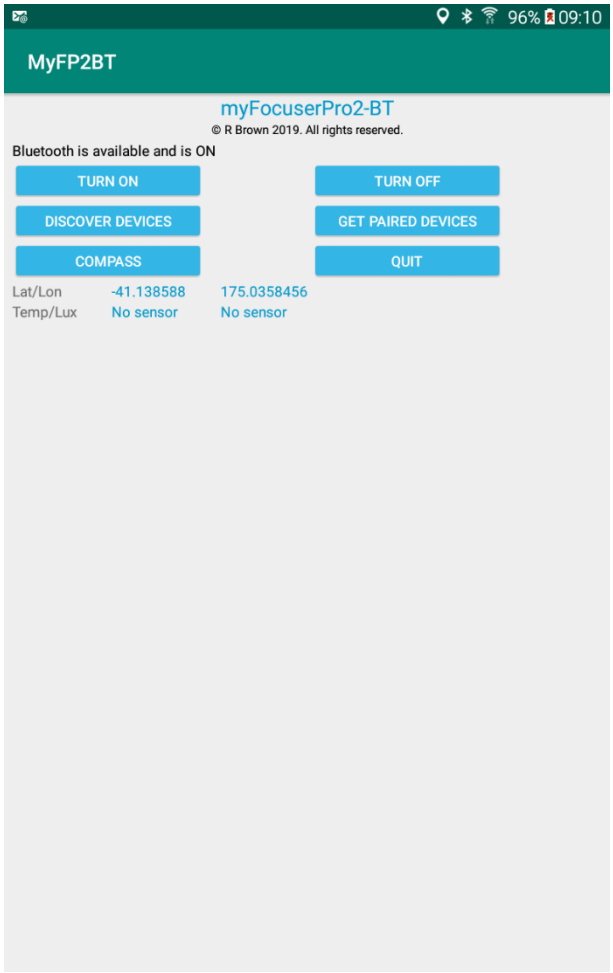
The picture shows the HC-05 as paired.

Once paired, you do not have to repeat the pairing unless you delete the HC-05 device from your Android device.

Next time, all you need to do is power on the controller, enable Bluetooth on your Android device, then run the application.

MAIN ACTIVITY SCREEN

It is best to have the HC-05 enabled controller powered on and paired with the Android device before starting the application.

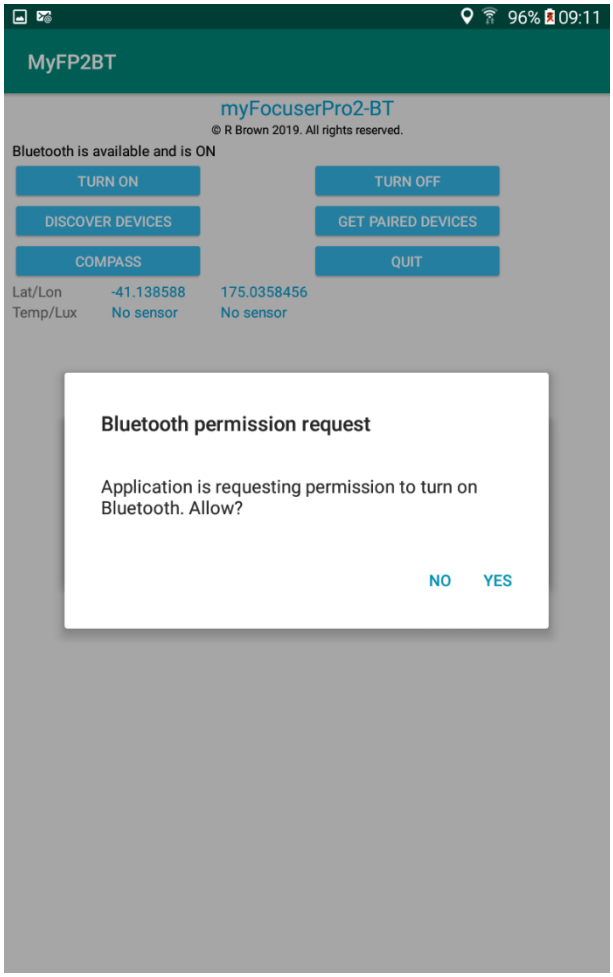


The initial screen gives a number of options

- **Turn On** provides a means to enable the Bluetooth device on the Android phone
- **Turn Off** provides a means to disable the Bluetooth device on the Android phone
- **Discover Devices** scans and displays a listed of discovered Bluetooth devices found by the Android phone
- **Get Paired Devices** lists the currently paired Bluetooth devices for the Android phone
- **Compass** opens a compass to give a bearing heading. Accuracy depends on whether the location sensor on the phone has been calibrated correctly
- **Quit** exits the application

- Latitude and Longitude are the current coordinates of the phone. If shown as 0, this indicates that the Location button has not yet been selected. The values are only updated IF the location button is selected
- Temp/Lux displays the temperature and Lux (light) values if those sensors are present

TURN ON

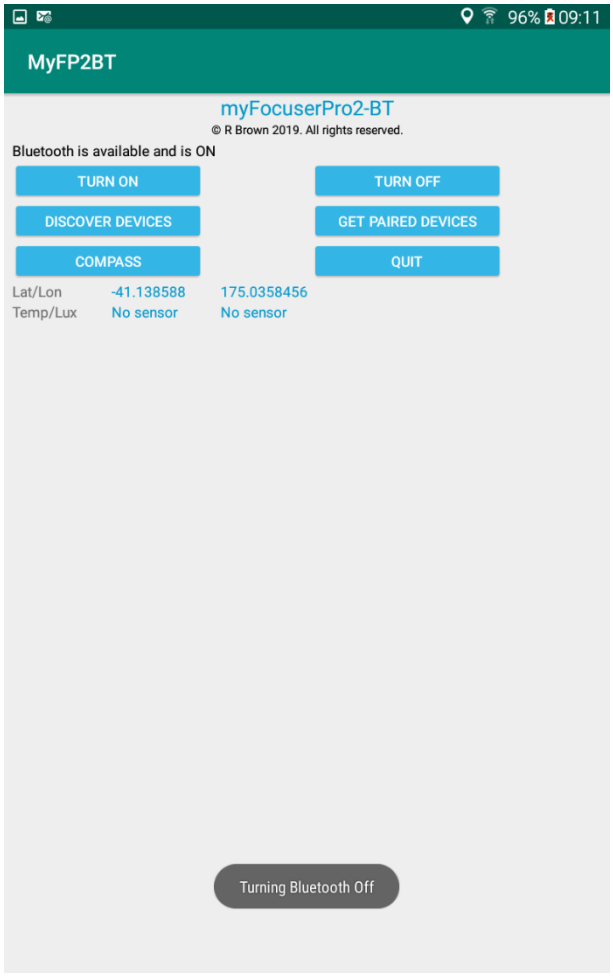


Turns on Bluetooth.

A dismiss dialog is shown.

Tap YES to enable Bluetooth.

TURN OFF



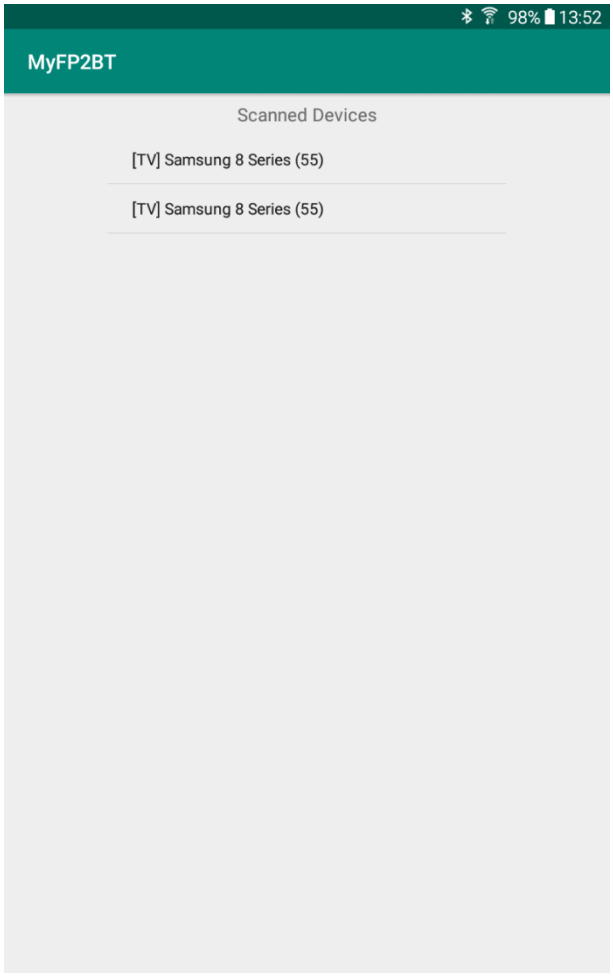
Turns off Bluetooth.

Bluetooth will be turned off.

A toast message confirms Bluetooth is being turned off.

When Bluetooth is OFF, you cannot communicate with the myFocuserPro2 Bluetooth controller.

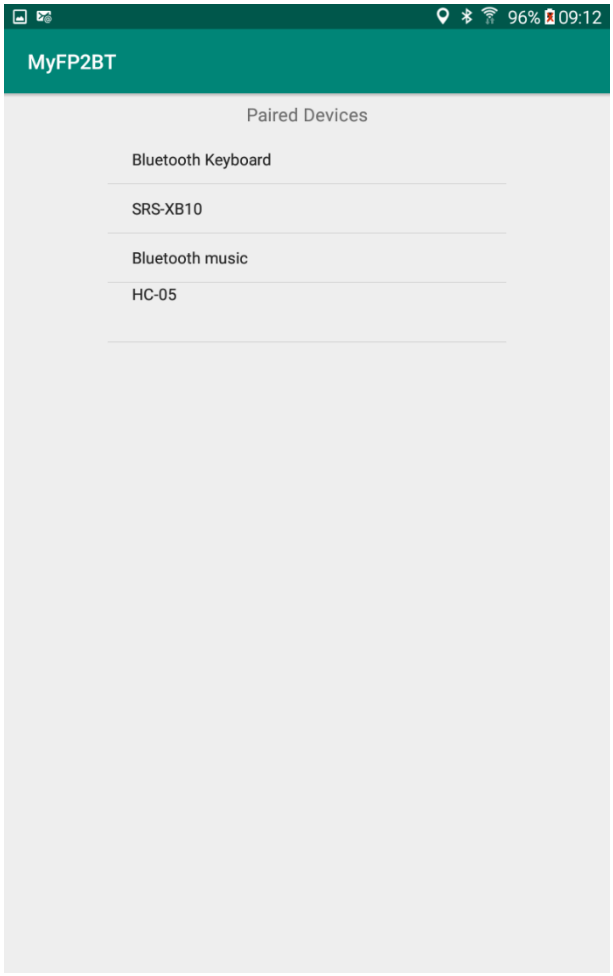
DISCOVER DEVICES



List all available Bluetooth devices that are not paired.

Bluetooth must be turned on for this to function.

GET PAIRED DEVICES



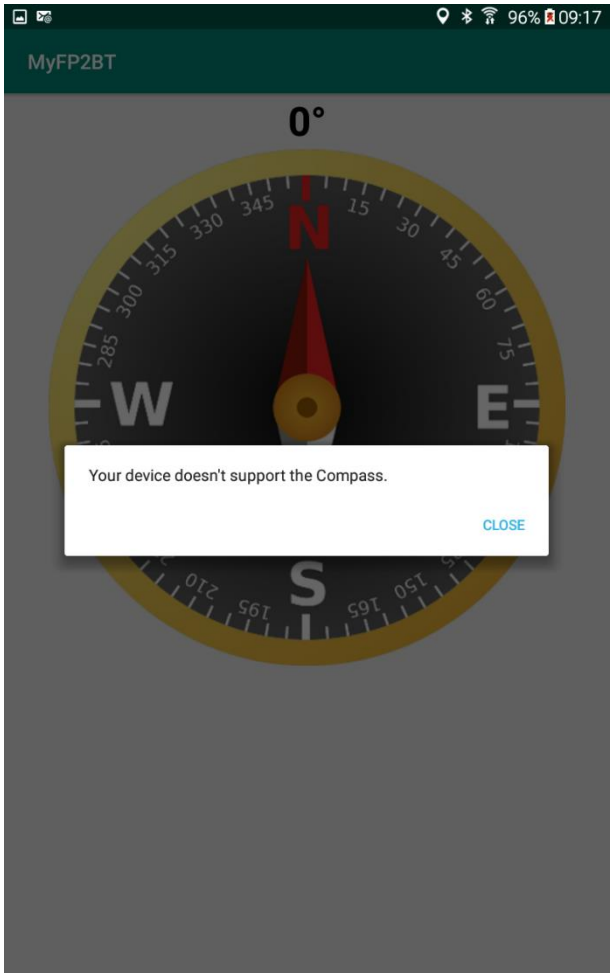
Get all paired Bluetooth devices.

The HC05, if paired, will show in this list.

To connect to the HC-05 device, tap on the item in the list.

Bluetooth must be turned on for this to function.

COMPASS

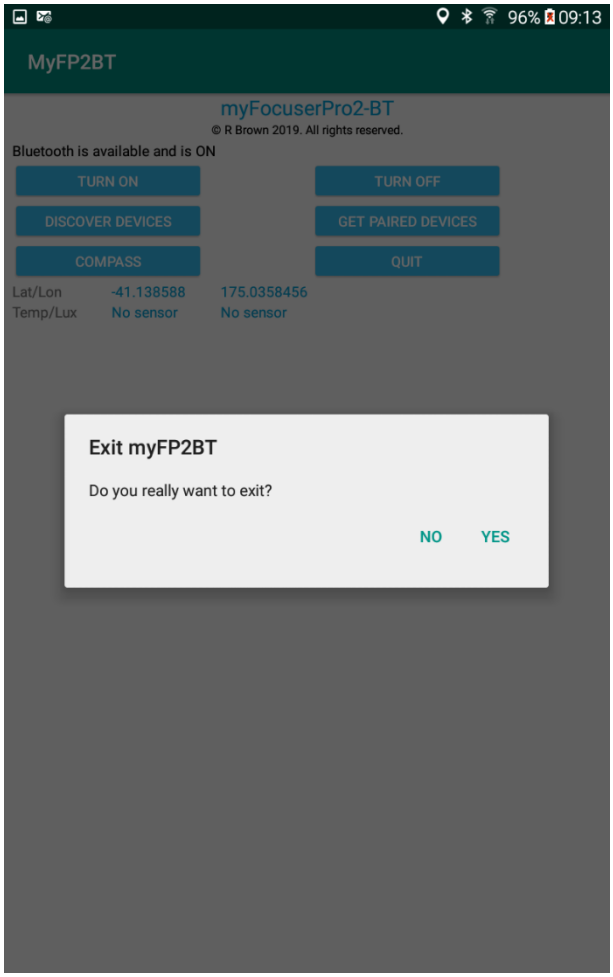


Tapping the Compass button will display a compass based on the user's current phone heading.

To return to the main screen tap the Back arrow.

The phone/tablet must have a magnetic sensor for this to function.

QUIT



When the QUIT button is tapped, a dialog box will appear asking to quit (Yes or No).

If the user taps YES, then the application will exit.

If the user taps NO, the application will return to the Main Activity.

CONNECT-1

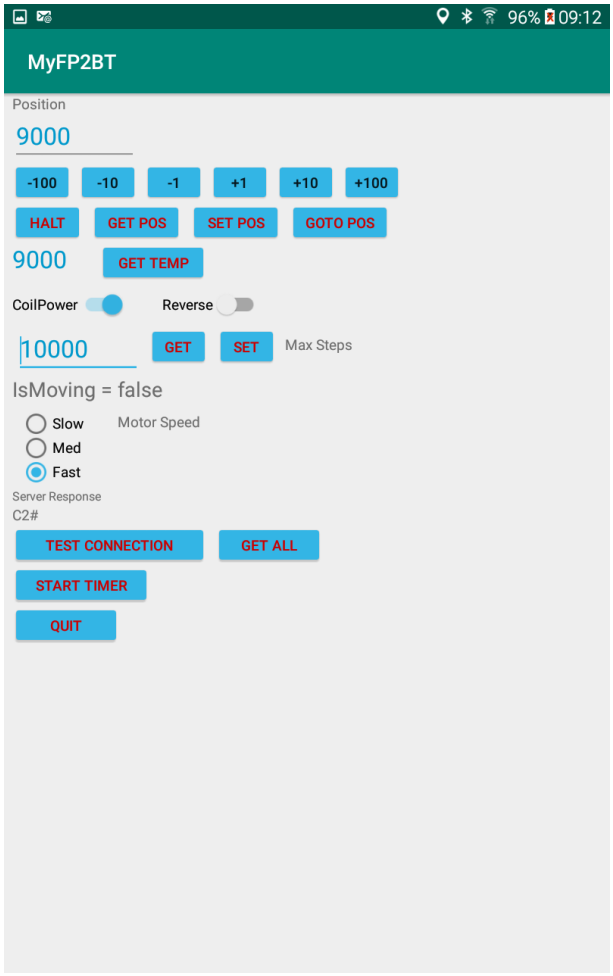
The Connect activity is displayed when the user has done Setup, followed by tapping the Connect button.

Before any commands can be sent to the controller, the user MUST tap the TEST CONNECTION button.

Once connection has been made, the values can be obtained from the controller by tapping the GET ALL button.

CONNECT-2

This example shows an established connection to the Bluetooth controller and the user has tapped the GET ALL button.



All buttons are self-explanatory. For example, to get the current controller temperature, tap the GET TEMP button. To move the focuser 100 steps out, tap the +100 button.

If the user taps in the Position field, the soft-keyboard will appear and the user can change the current position value. Once this is done, the new value can be sent to the controller by tapping the SET POS button.

If the user wants to have the values regularly updated, the START TIMER button is tapped. This starts a timer which will periodically get the position, temperature and ismoving status from the controller.

When the timer is running, the STOP TIMER button will be displayed.

To exit the Connect Activity and return to the Paired Activity tap the QUIT button.

HOW TO INSTALL THE ANDROID BLUETOOTH APPLICATION

Currently the Android BT application is an APK file that is provided for download as an unsigned app and not via Google Play.

SETTING UP YOUR DEVICE

From your smartphone or tablet running Android 4.0 or higher, go to Settings, scroll down to Security, and select Unknown sources. Selecting this option will allow you to install apps outside of the Google Play store. Depending on your device, you can also choose to be warned before installing harmful apps. This can be enabled by selecting the Verify apps option in the Security settings

HOW TO INSTALL THE APK FROM YOUR ANDROID DEVICE?

Just open your browser, find the APK file you want to download, and tap it – you should then be able to see it downloading on the top bar of your device. Once it's downloaded, open Downloads, tap on the APK file, and tap Yes when prompted.

WHERE DO I PUT AN APK FILE ON MY ANDROID?

To get started, download an APK file using either Google Chrome or the stock Android browser. Next, go to your app drawer and click Downloads; here you will find the file you just downloaded. Open the file and install the app. If you downloaded the APK file on your computer, the process is slightly different.

INSTALLING AN APK FILE FROM THE FILE MANAGER

Download the APK file if necessary. If you haven't yet downloaded the APK file onto your Android, do the following: ...

Open your Android's file manager app. ...
Select your Android's default storage. ...
Tap Download. ...
Tap the APK file. ...
Tap INSTALL. ...
Tap DONE when prompted.

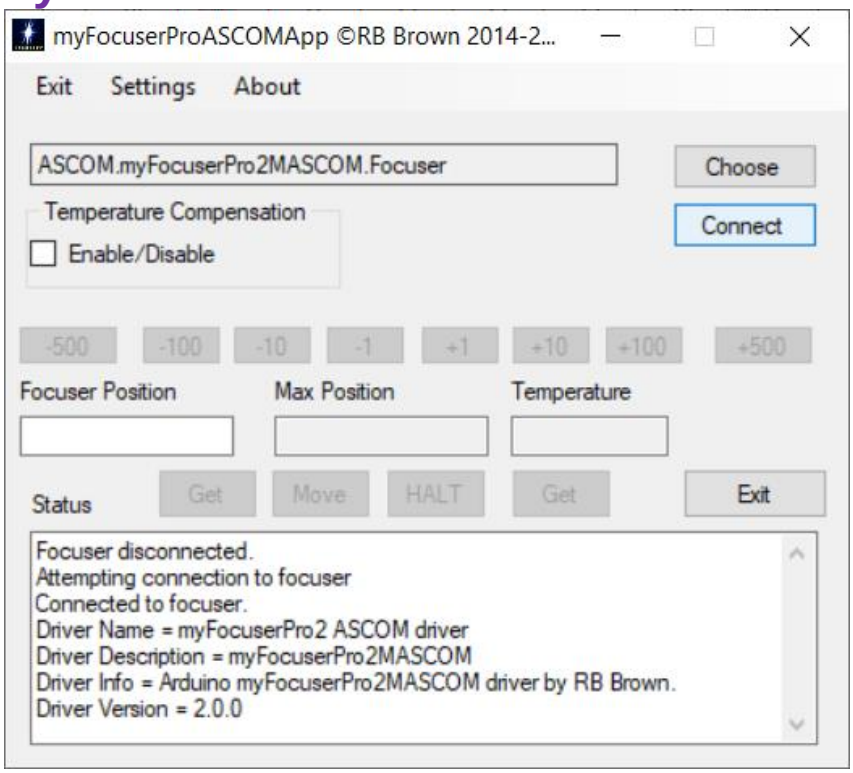
HOW TO CALIBRATE THE COMPASS FOR THE SAMSUNG GALAXY S9+ ANDROID PHONE

1. When you are on the start screen, open the phone app
2. Select the dial keypad to manually enter numbers.
3. Next, type *#0*# on your Samsung Galaxy S9 or S9 Plus smartphone.
4. The service menu will open up. Select "Sensor" and scroll down to the bottom of the next menu.
5. You will see "Magnetic Sensor" and below it a circle with a line and a number behind it. This has the following meaning: Red line with a 0 = The compass sensor must be calibrated Blue line with a 3 = The magnetic sensor is calibrated 6.

To calibrate the compass, rotate the Samsung Galaxy S9 once around all three axes. Do this until the circle shows a blue line and a 3. Only then is your compass calibrated.

Source: https://www.solvemix.com/index.php/android-smartphones/17-year-2018/2522-samsung-galaxy-s9-google-maps-arrow-shows-wrong-direction-compass
Android Smartphones, © Solvemix - quick tips, tricks and solutions

myFOCUSERASCOM APPLICATION TESTER



This a Windows software application that acts as ASCOM client which can talk to any ASCOM focuser driver and allow you to control the focuser.

The **Settings** menu has a number of options, similar to those provided by the Windows Application.

The application is found in the folder **ASCOM Driver Tester** on the CDROM. Run the setup.exe program to install.

myFOCUSERPRO2 ASCOM DRIVER

The myFocuserPro2 ASCOM driver provided comes with an installer program. The ASCOM driver works with ALL driver build options.

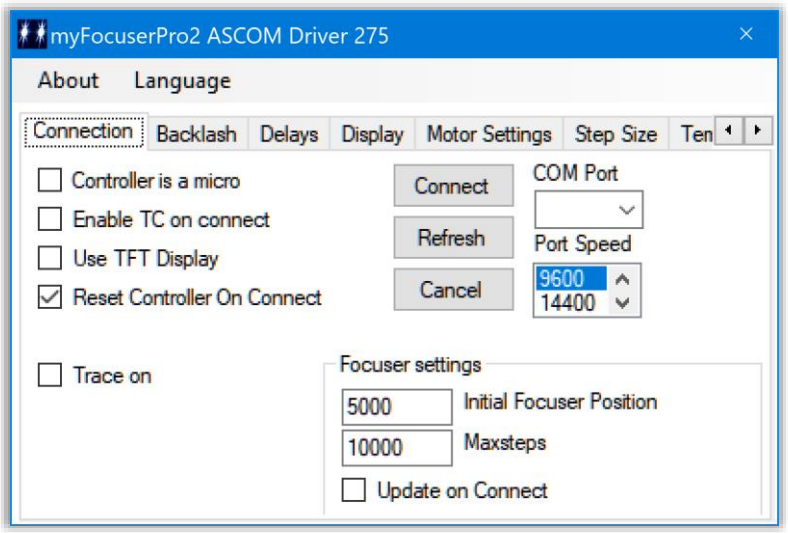
Additional settings are accessed via the Settings menu bar option and must be set before connecting to the controller.

To run two focus controllers using ASCOM, you need to install the secondary ASCOM driver.

IT IS IMPORTANT THAT IF USING BOTH THE ASCOM DRIVER AND WINDOWS APPLICATION TO CONTROL THE FOCUSER THAT YOU USE THE SAME SETTINGS FOR REVERSE DIRECTION, HALF STEPS (or stepmode) AND COIL POWER IN BOTH THE ASCOM DRIVER AND WINDOWS APPLICATION.

ASCOM DRIVER CONNECTION TAB

The connection tab provides the main connection point for the driver to connect to the controller.



Controller is a Micro

Check this box if the Arduino controller is a Micro. For all other controllers, leave unchecked. *This also requires special changes to the Arduino firmware file.*

Enable TC on connection

When checked, temperature compensation will be turned on in the controller once a connection is established.

Use TFT Display

Enable this if the controller is fitted with a TFT display

Reset Controller on Connect

When checked, will reset the Arduino controller when connecting. The recommended setting in checked so that the Controller starts with a known state.

Trace On

Debug messages are written to a trace file

Com Port and Com Port Speed
You need to determine the correct Com Port that the focuser is using (you can get this easily by using the Windows Application). Once you have the Com Port value, you need to select it from the Combo List for the Com Port on the ASCOM setup dialog box before clicking the Connect button. The application saves the selected Com Port value. The default speed is 9600.

FocuserSettings Initial Position
This setting specifies the initial position of the focuser. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

FocuserSettings MaxSteps
This setting specifies the maximum number of steps for the focus controller and will vary depending upon each user's configuration. You should set this to the maximum number of steps that you have calculated for your focuser [see determining maxStep]. This setting must be specified before connecting to the controller and will be remembered by the application. See the section on the initial setup of the focuser for further information. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

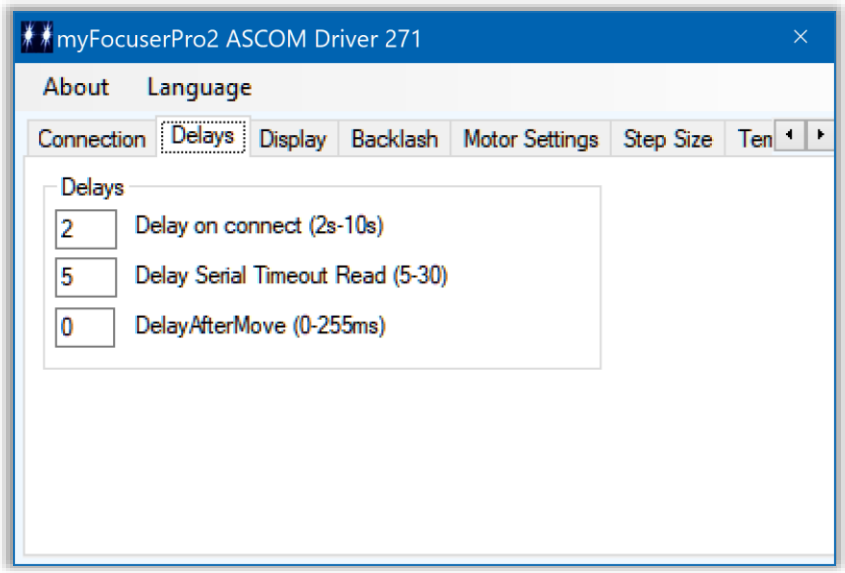
FocuserSettings Update on Connect
If enabled, the Initial focuser position and maxStep values will be sent to the controller when connecting. If unchecked, the focuser will use the previous saved values in the controller.

CONNECT
When all the settings have been set to their desired states, click the CONNECT button.

REFRESH
The refresh button updates the list of com ports shown in the drop-down list of available com ports.

CANCEL
Aborts the setup dialog form

ASCOM DRIVER DELAYS TAB

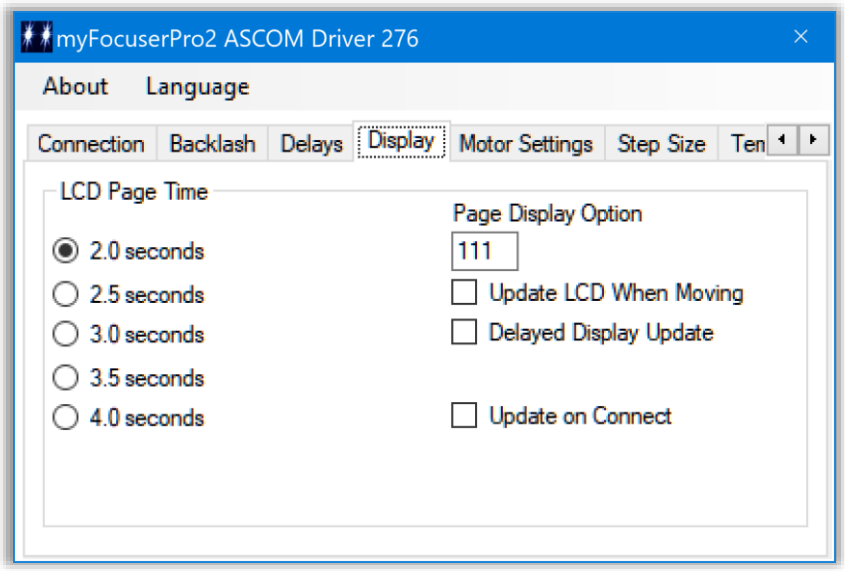


Delay (s) 2-10 on Connect
Specify the delay in seconds after connecting that the driver will wait before sending a command to the myFP2 controller. Valid values are 2-10 seconds. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

Delay (s) 5-30 timeout on Serial Read
Specify the delay in seconds that the application will wait when attempting to read from the serial port after sending a command to the myFP2 controller (default = 5). For Bluetooth or slower devices, a value of 8 or 10 may suffice. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

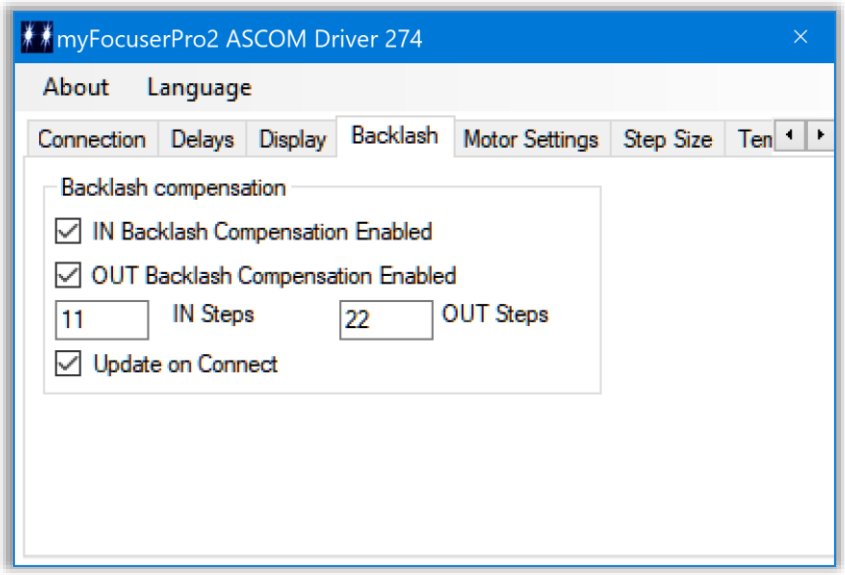
DelayAfterMove (0-255)
Specifies the delay in milliseconds that is applied after move to prevent any blurring of star caused by any mechanical residual movement or vibration of the stepper or focuser during focusing. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

ASCOM DRIVER DISPLAY TAB



MYFP2M BOARDS CAN SUPPORT A DISPLAY BUT ARE NORMALLY BUILT WITH NONE.

ASCOM DRIVER BACKLASH TAB



IN Backlash Compensation Enabled
Enable or disable backlash compensation of IN moves

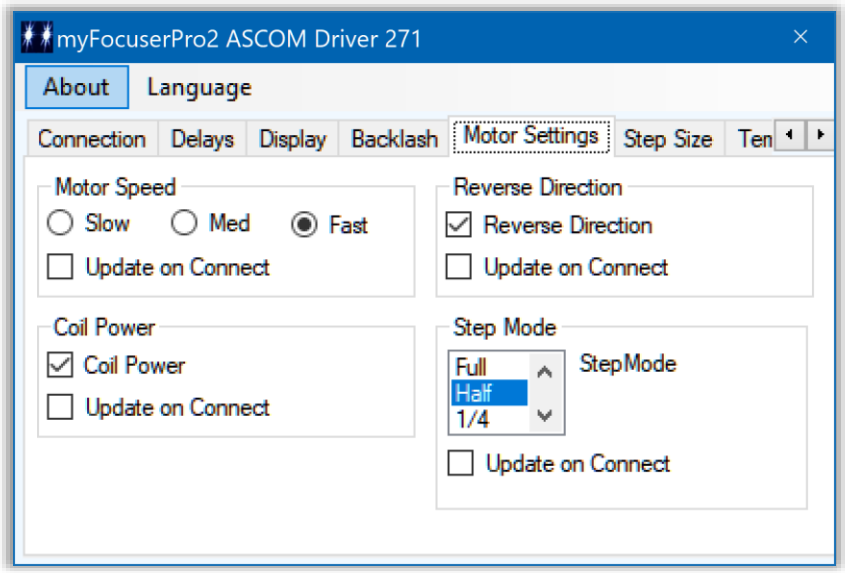
OUT Backlash Compensation Enabled
Enable or disable backlash compensation of OUT moves

IN Steps (0-255)
The number of steps of backlash to apply after the IN move has completed. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

OUT Steps (0-255)
The number of steps of backlash to apply after the OUT move has completed. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

Update on Connect
If checked, the ASCOM driver will send the Backlash Settings to the controller when connecting. If unchecked, Backlash settings will be retrieved from the controller when connecting and saved for future use.

ASCOM DRIVER MOTOR SETTINGS TAB



Motor Speed
The speed of the stepper (delay between steps) can be adjusted in the range Slow-Medium-Fast. The speed setting is remembered by the controller. If the Update on Connect checkbox is checked, then the value of motor speed will be sent to the controller when connecting. If the checkbox is not checked, then the controller will use the last remembered motor speed setting, and the ASCOM driver will retrieve this value when connecting to the controller and save it for future use.

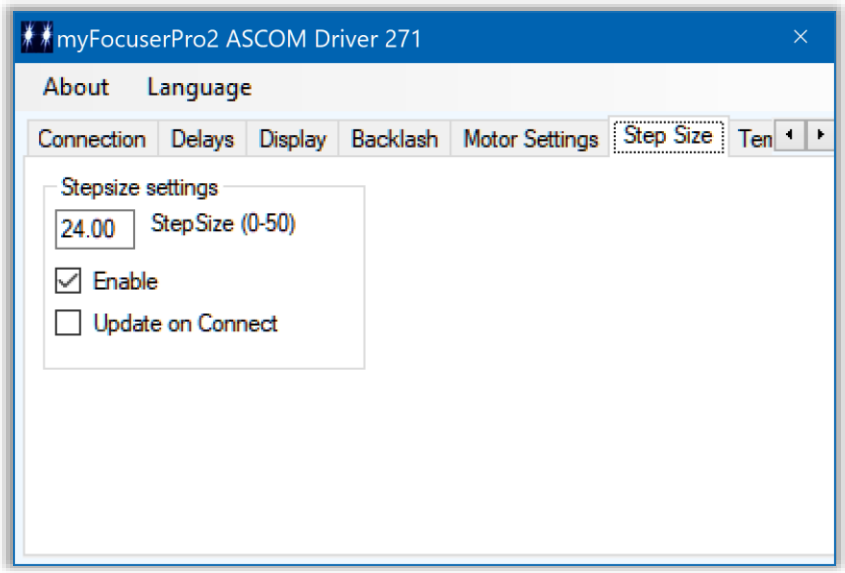
Coil Power
When enabled (ticked), indicates that coil power is ON and the stepper coils are powered after the move is completed. If the Update on Connect checkbox is checked, then the value of Coil Power will be sent to the controller when connecting. If the checkbox is not checked, then the ASCOM driver will retrieve this value when connecting to the controller and save it for future use.

Reverse Direction
When enabled (ticked), indicates that the motor moves in the opposite direction (IN means OUT and OUT means IN). If the Update on Connect checkbox is checked, then the value Reverse Direction will be sent to the controller when connecting. If the checkbox is not checked, then the ASCOM driver will retrieve this value when connecting to the controller and save it for future use.

Step Mode
The desired micro stepping mode.

Update on Connect
If the Update on Connect checkbox is checked, then the value Step Mode will be sent to the controller when connecting. If the checkbox is not checked, then the ASCOM driver will retrieve this value when connecting to the controller and save it for future use.

ASCOM DRIVER STEPSIZE TAB

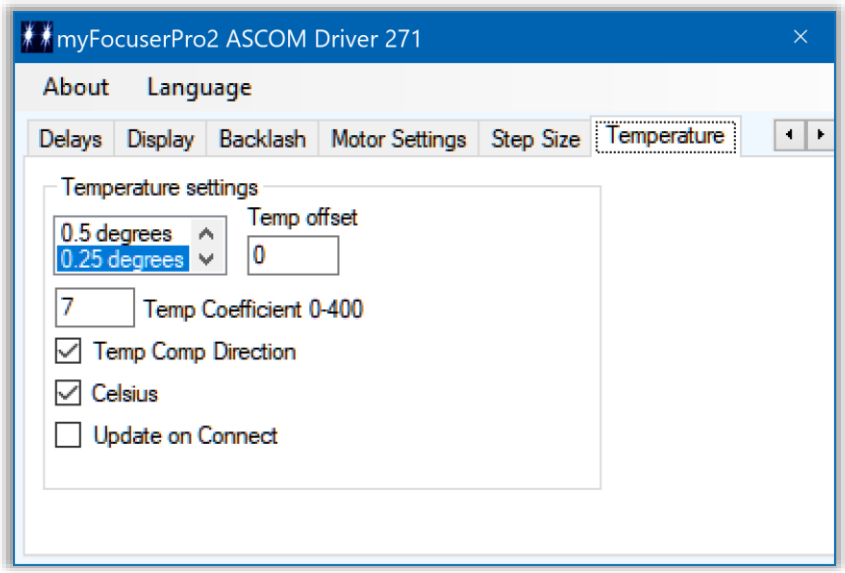


Step Size in microns (0-50)
This setting defines the Step Size in microns. You can enter the calculated value for your focuser and this will be sent to the myFP2 controller when connecting. This allows client applications using the ASCOM driver to retrieve the Step Size setting from the myFP2 controller. Please see the section in this PDF on determining your focuser step size value. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

Enable
This setting turns ON or OFF the reporting of the StepSize value from the myFP2 controller. If you do not know your step size value, DO NOT enable this feature.

Update on Connect
If checked, the ASCOM driver will send the Step Size Settings to the controller when connecting. If unchecked, Backlash settings will be retrieved from the controller when connecting and saved for future use.

ASCOM DRIVER TEMPERATURE TAB



Temperature Resolution
Allows the user to control the resolution of the DS18B20 temperature probe from 0.5 degrees down to 0.0625 degrees (9=0.5, 10=0.25, 11=0.125, and 11=0.0625). This setting is remembered by the myFP2 controller.

Temp Offset
The value to subtract from the temperature reading returned from the myFP2 controller before returning it to the requesting application. Adjustment values range from –3 to +3. For example, typing -1.5 into the entry-box will subtract 1.5 degrees C from the returned temperature value. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

Temperature Coefficient (0-400
The value in steps per degree for temperature compensation, and is specific to each focuser and must be calculated by the user. The myFP2 Windows application has special support for calculating this value. *After typing the new value in the field, you must use the ENTER key to ensure the value is validated. If the value you entered is invalid, then the field is updated with a default value.*

Temperature Compensation Direction
Use this setting to set the direction in which temperature compensation moves are applied. Checked means moves are make Inwards (normal).

Celsius/Fahrenheit
When checked, temperature is displayed in Celsius (default). When unchecked, temperature is displayed in Fahrenheit. This temperature display setting can be changed at any time.

Update on Connect
If checked, the ASCOM driver will send the Temnperature Settings to the controller when connecting. If unchecked, Backlash settings will be retrieved from the controller when connecting and saved for future use.

ASCOM DRIVER CONFORM VALIDATION

Every release of the myFP2 ASCOM drivers are tested against the CONFORM utility. This program tests to ensure that the functions/methods/properties provided by the ASCOM driver are compliant to the ASCOM standard.

All myFP2 drivers are ASCOM compliant and the validation reports are available on the myFP2 site.

TEMPERATURE COMPENSATION AND THE ASCOM DRIVER

An **ASCOM client** has the ability to send a command to an ASCOM driver and enable/disable temperature compensation. The myFP2 ASCOM driver and firmware supports temperature compensation.

Temperature compensation is enabled by the ASCOM Client application. If your client application does not support this then please contact the developer to have this function added to their application.

For temperature compensation to work, you need to calculate the temperature co-efficient for your focuser/telescope. The Windows application can automatically calculate the temperature coefficient and store the value in the controller. Use the Windows application to calculate the temperature coefficient and store this value in the controller. Once set, the ASCOM driver can use this value.

Here is the part of the ASCOM Conform report for the myFP2 related to temp compensation

```
17:30:58.416 TempCompAvailable      OK      True
17:30:58.456 TempComp Read          OK      False
17:30:58.465 TempComp Write         OK      Successfully turned temperature compensation on
17:30:58.473 TempComp Write         OK      Successfully turned temperature compensation off
17:31:06.998 Move - TempComp True           Moving to position: 61748
17:31:07.005 Move - TempComp True   OK      .NET InvalidOperation Exception correctly raised as expected
Conformance test complete
No errors, warnings or issues found: your driver passes ASCOM validation!!
```

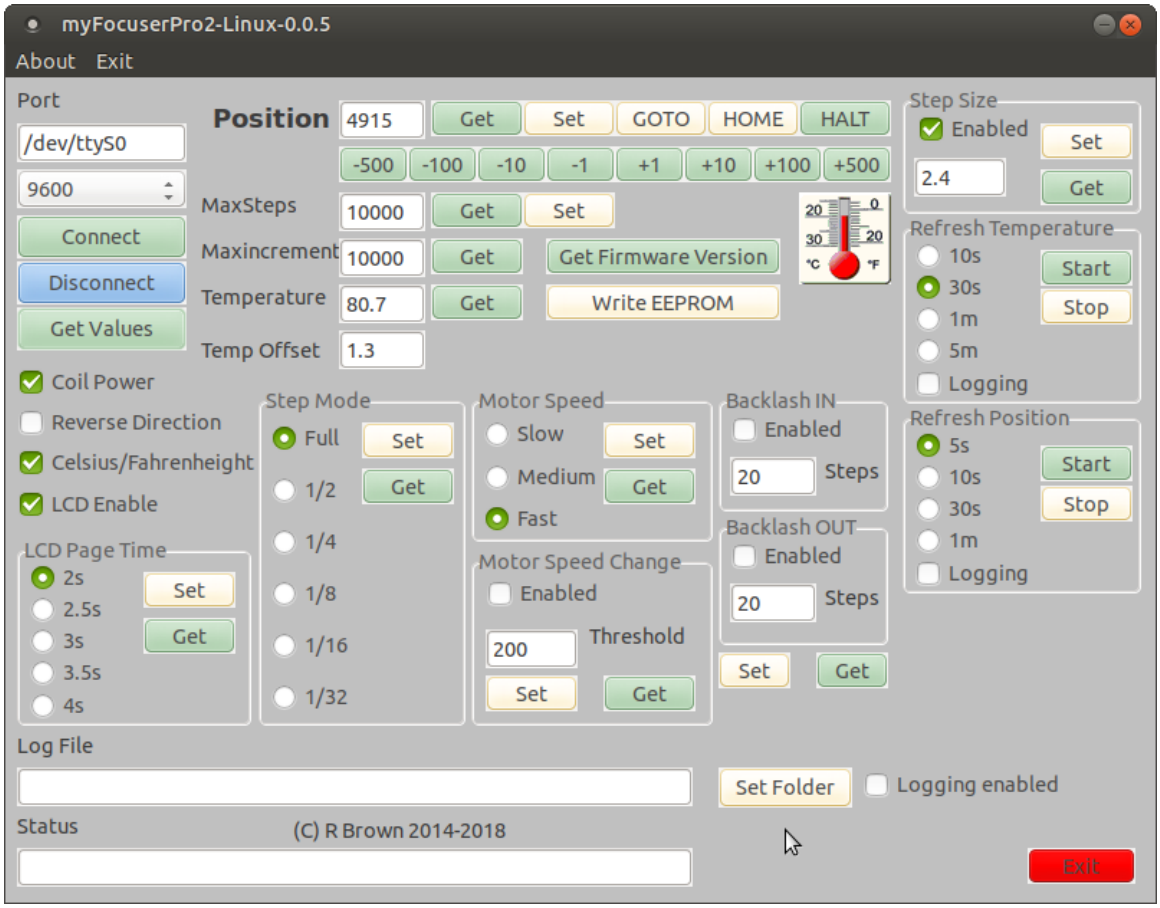
If the client application implements its own temperature compensation support, please disable it. Do not enable the client support if temperature compensation is implemented in the controller.

myFP2ASCOM LANGUAGE SUPPORT

Support is provided for multiple languages. Translation has been done by Google translate. Buttons, Labels, menu items and Hover messages are all translated. If the menu option Language, On Load, Enable is checked then the ASCOM driver will remember the language setting and reapply the setting each time the ASCOM driver is loaded.

myFocuserPro2-Linux

The provided LINUX application was developed and tested on UBUNTU with the MATE desktop environment.



Controller parameters are saved in the local file myFP2M-Linux.ini and sent to the controller when connecting, and are saved when the program exits.

INSTALL INSTRUCTIONS

Copy the myFP2M-Linux AND myFP2M-Linux.ini files to a user folder on your Linux system (like a folder under the desktop)

UBUNTU-MATE

You must be a member of the group tty
Use menu, administration, users and groups, manage groups, tty, properties and ensure that your name is checked, then click OK

chmod +x myFP2M-Linux
or you can right mouse click the file, properties, permissions and check allow executing file as program

I had to right mouse click file and chose run as administrator
It seems it cannot connect to tty0 device without admin privilege

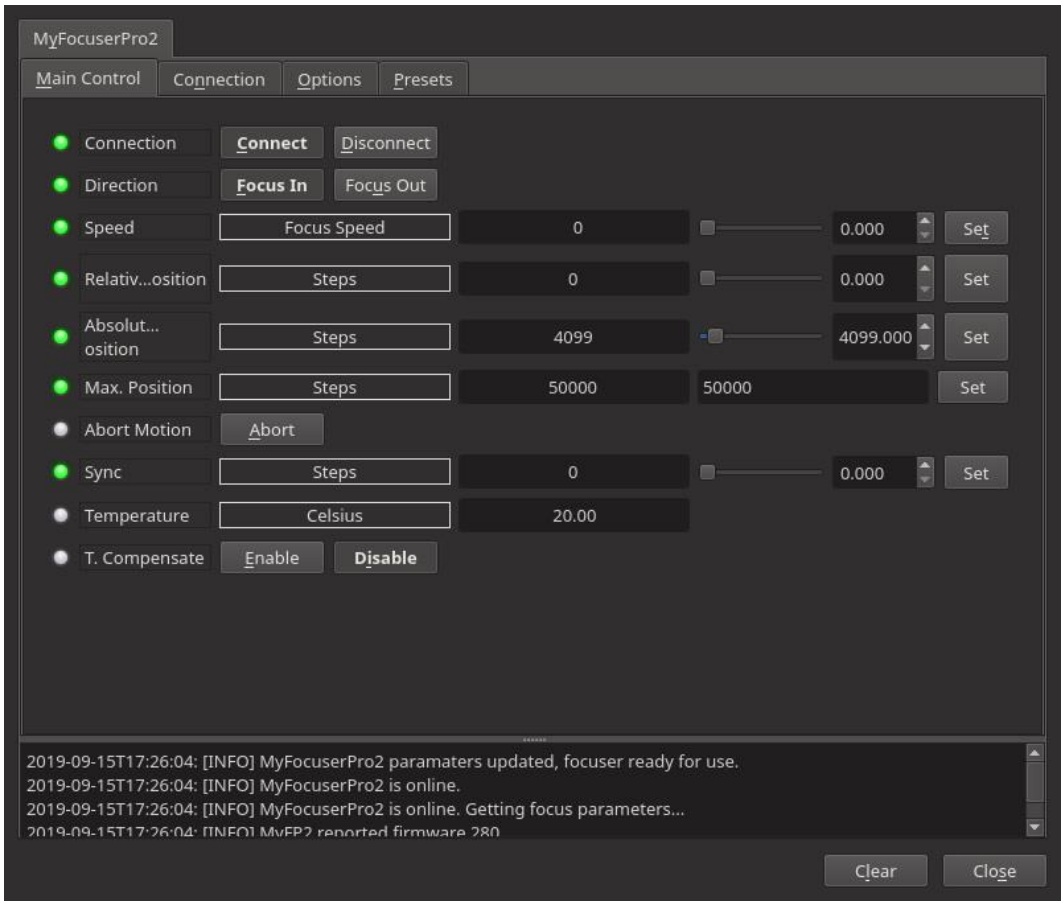
INDI

Thanks to Alan Townshend myFP2 is supported by INDI.

https://indilib.org/devices/focusers/myfocuser-pro-2.html

Current features are:

- Sync
- Absolute and relative position
- Preset positions
- Temperature compensation
- Coil power control
- Display on or off
- Reverse direction
- Joystick control



INSTALLATION

The INDI myFocuserPro2 driver is included with libindi >= 1.8.2
Under Ubuntu, you can install the driver via:

```
sudo add-apt-repository ppa:mutlaqja/ppa
sudo apt-get update
sudo apt-get install indi-bin
```

TEMPERATURE COMPENSATION

Temperature compensation is the automatic adjustment of focus position based on changes in temperature. It will be different for each telescope. This feature is turned ON or OFF by an **ASCOM client** by sending the correct command to the ASCOM driver. The Windows application can talk directly to the controller and enable/disable temperature compensation.

Each user must determine their own compensation value for their equipment. This means calculating a **temperature coefficient** value, which is the number of steps the focuser needs to move to best focus when the temperature changes by 1 degree.

The user waits for the telescope optics to achieve thermal equilibrium and then takes a series of measurements over a period of temperature change. These measurement results are readings of focuser position versus temperature which can be plotted on a graph. The slope of the graph then is the temperature coefficient (ideally the graph should be a linear line but this might not be the case).

Generally, as temperature drops over the course of the evening, the focuser position will move inwards. We are going to record the in movements over changes in temperature.

Generally, temperature compensation is only applied in one direction (inwards or as a fall in temperature reading), though some systems might have compensation in both directions.

Assuming we only do temperature compensation in one direction, this also avoids backlash issues which might accrue from a reverse change in focuser direction.

So, manually, we might do the following

1. Wait for the temperature to stabilize (thermal equilibrium)
2. Slew to a target star, enable mount tracking and guiding software
3. Use a Bahtinov mask (or FWHM value) to get the best focus
4. Wait for the temperature to drop by a specified amount (3 degrees)
5. Refocus
6. Record the temperature and the focus position
7. Calculate the temperature coefficient
8. Update the controller settings and enable temperature compensation
9. Remove the Bahtinov mask and start imaging

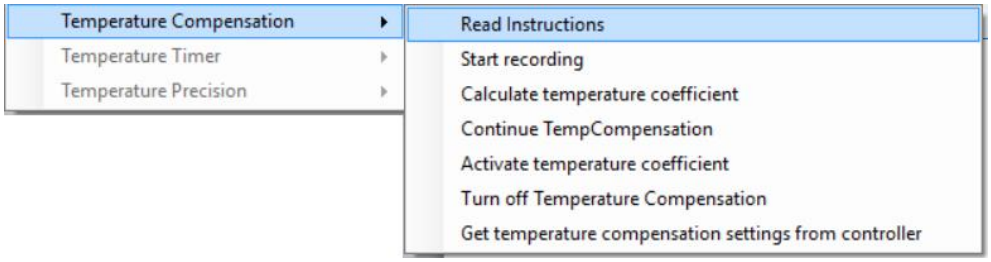
The following pages outline how to do this with the myFP2M controller and application software.

Step 0 Read the Instructions

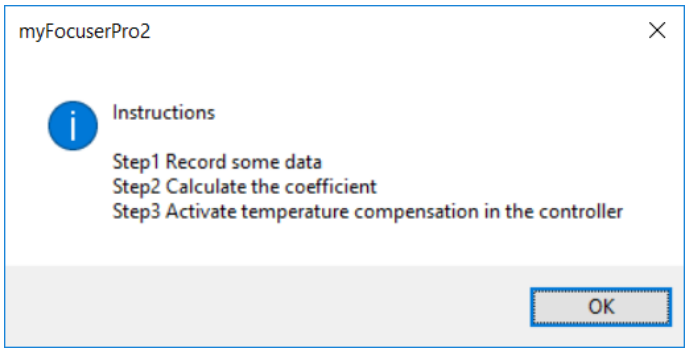
The following assumptions are

1. The temperature probe is located near the optics of the telescope (for a refractor this would be near the front lens cell)
2. The telescope has reached thermal equilibrium. This may take up to 30 minutes or more to occur and varies depending on the conditions and telescope type and size
3. The myFocuserPro2 controller is connected and the Windows Application is running
4. The telescope is at optimal focus and tracking the target star

The first step is to access the **Read Instructions** option from the Temperature Compensation menu of the myFocuserPro2 windows application. Select the Read Instructions option

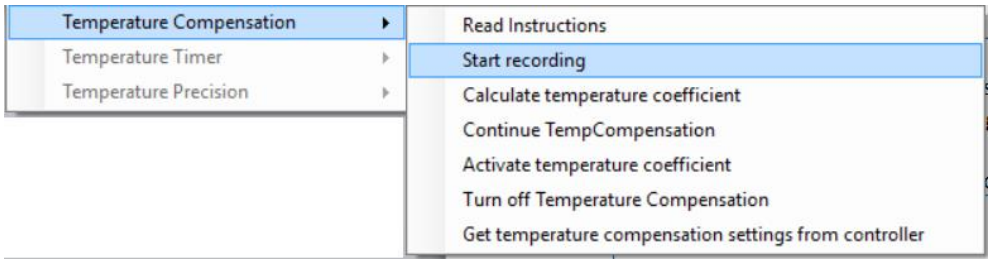


This displays the following MessageBox of the steps you must follow to determine and apply a temperature coefficient value and enable temperature compensation for your myFP2M controller. The steps are performed one after the other in order. If you make a mistake, simply start again from Step 1.

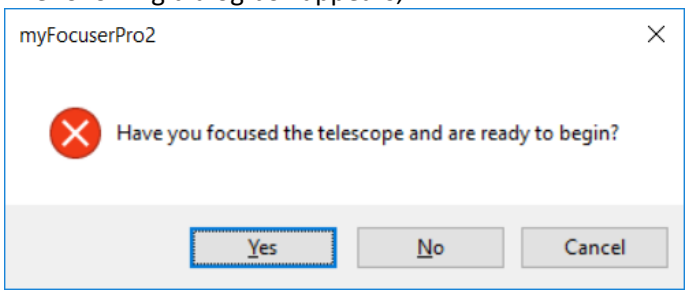


Step 1 Start Recording

The next step is to access the **recording** option from the Temperature Compensation menu of the myFocuserPro2 windows application. Select **Start recording**

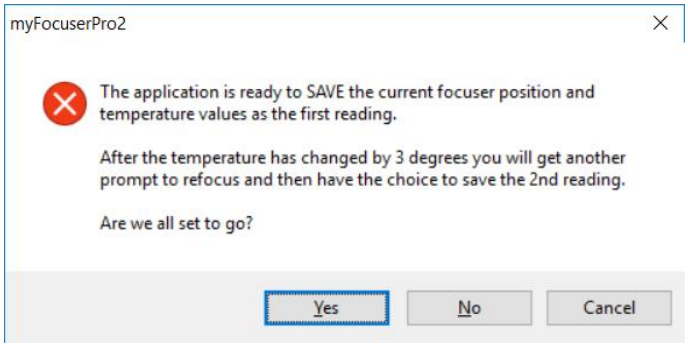


The following dialog box appears,

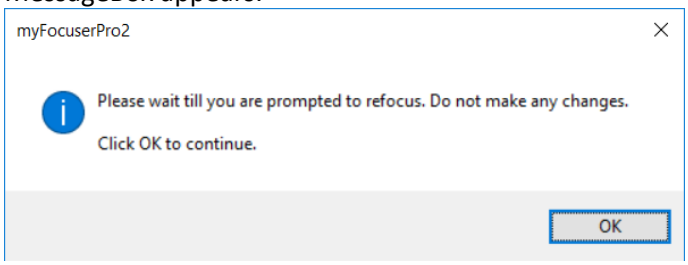


If you have already focused the telescope within the last few minutes, click **Yes**, else click No and refocus the telescope before starting again.

After clicking **Yes**, the following dialog box appears,



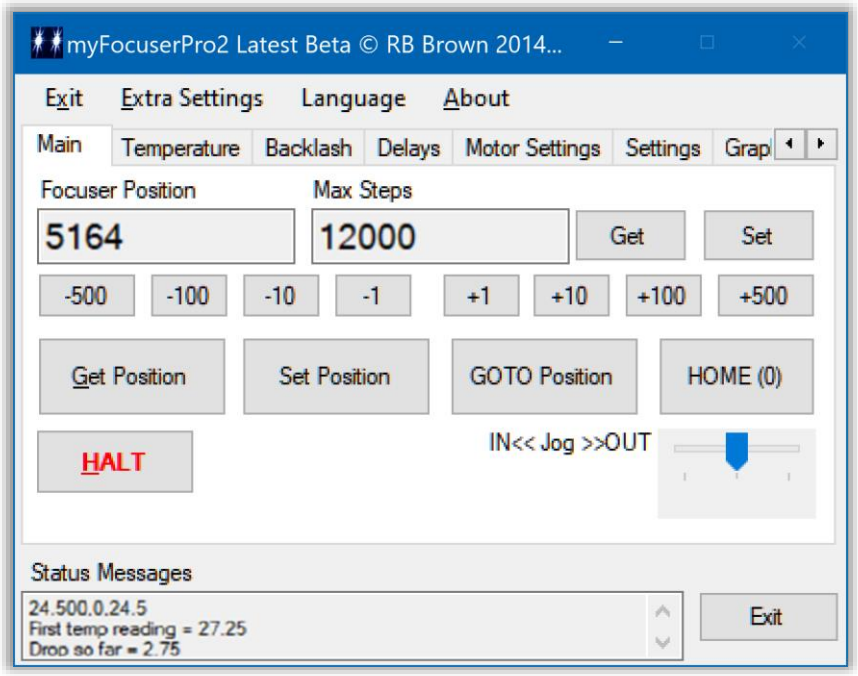
When you click **Yes**, you will be automatically taken to the record dialog option and the following MessageBox appears.



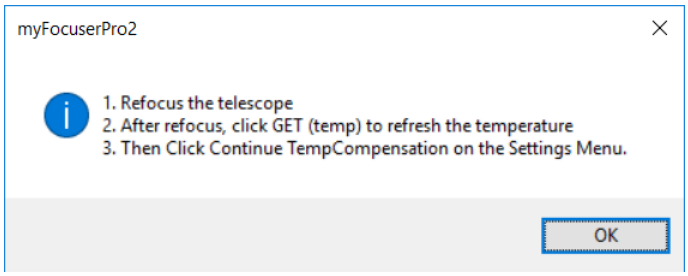
At this point, the program has automatically

- saved the current temperature and focuser position
- started the auto refresh timer for the temperature updates and enabled it to a 10s refresh cycle
- started to monitor the temperature change

Just wait for the application to monitor the temperature and prompt you when ready. During this time interval DO NOT MAKE ANY CHANGES TO ANY SETTINGS. You will be able to see the progress in the status textbox, as indicated below

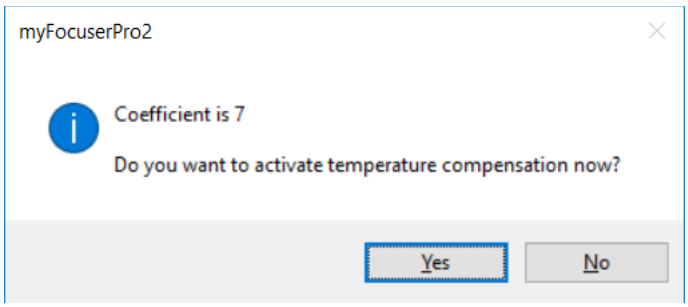


Once the program detects a 3 degree change in temperature, a new dialog MessageBox will automatically appear asking you to refocus the telescope. Click **OK** and then refocus the telescope.

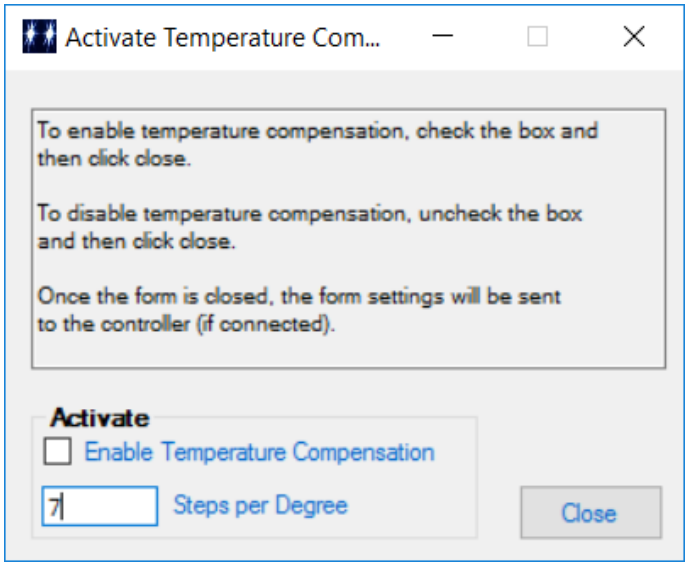


After refocusing, get the latest temperature reading by clicking the GET button for temperature. Once the new temperature value is displayed, click the **Continue TempCompensation** menu option on the settings menu under Temperature Compensation.

The application will automatically calculate the temperature coefficient and display the value in a MessageBox (example below) and include an option for you to now update the controller and enable temperature compensation.



Click **YES** to Goto the next step. The application will now display the Apply Coefficient Form as shown below.



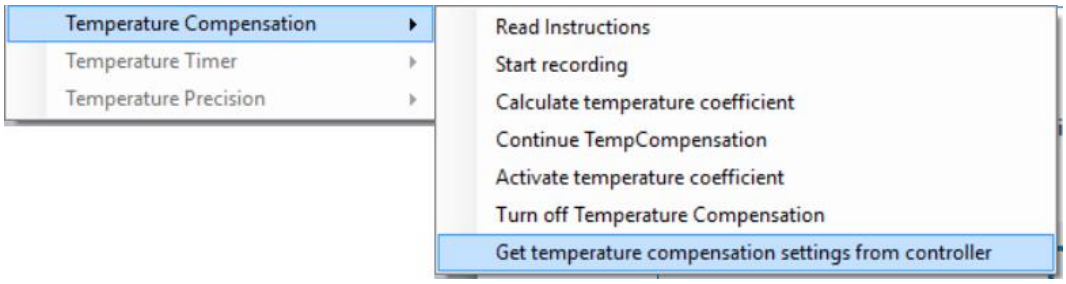
The Steps per Degree is automatically preloaded from the previous calculation. To send this value to the controller, and enable temperature compensation, check the box **Enable Temperature Compensation** and then click the **Close** button.

Note: If you know the temperature coefficient value for your focuser, you can access this menu directly and enter the Steps per Degree value manually.

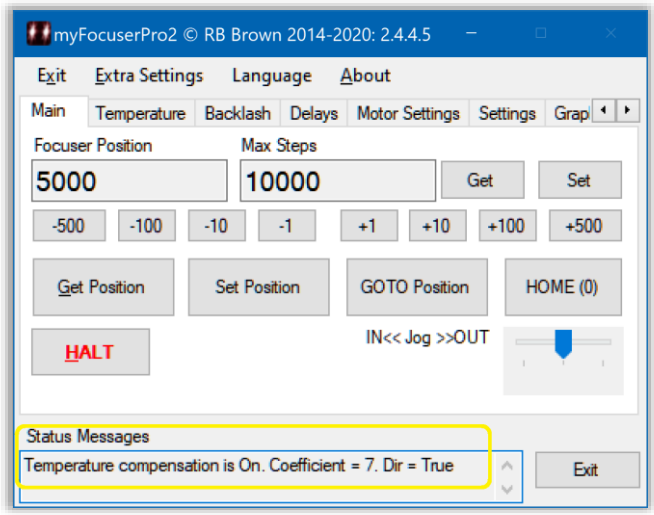
The values are sent to the myFP2M controller and temperature compensation is enabled.

Display the Current myFP2M Controller Temperature Compensation Settings

To display the current temperature compensation settings, select the **Get temperature compensation settings from controller** option



The application will query the controller and display the current settings.
The application will query the controller and display the current settings in the status area.



The Windows Application and ASCOM driver support temperature compensation. However, the controller handles the changes internally when enabled.

AT THE BEGINNING OF THE EVENING SHOULD YOU FOCUS AT LEAST TWO OR THREE TIMES WHEN THE TEMPERATURE CHANGES?

That would depend upon your temperature coefficient setting compared to step-size.
If the temperature coefficient setting was 28 steps, and your step size was 4.2, then this means that for every degree of temperature change, the focuser must move by 28 steps to compensate. This pdf explains how to calculate step size, critical focus zone and temperature coefficient steps.

If the temperature drops by 1 degree, this means we need to move the focuser 28 steps to stay in focus [the temperature coefficient].

The step size is 4.2 and we have 10 steps in the critical focus zone. So that means the focuser has 10 steps where it is in focus. But we need to move 28 steps. which is way outside the critical focus zone and so the image is out of focus and we must move the focuser to stay in focus.

This would mean adjusting the focus every time there is a temperature change of 1 degree.

If you are an imaging taking photos, then you need to

1. wait till thermal equilibrium of the scope has occurred
2. determine your coefficient setting [pdf explains how]
3. enable step size in the focuser and set the step size value
4. set the temp coefficient value in the focuser
5. before doing an imaging run, do an auto focus
6. enable temperature compensation
7. start the imaging sequence

TEMPERATURE COMPENSATION DIRECTION

If you find your images are slowing becoming out of focus as the imaging run capture sequence advances, this means the temperature compensation direction value is set wrong and should be changed to the other direction.

Please see the following video
https://www.youtube.com/watch?v=YXRqP-V1fcM

FINE-TUNING THE STEPPER MOTOR SPEED

Firmware 292+ offers the capability to determine the correct motor speed delay settings used to control the speed of the stepper motor.

In the firmware myBoard,h file, there are THREE delay settings used to control the motor speed for each of the motor speed settings.

There are different values for each board type. For the DRV8825HW203 board, the settings look like

```
#define DRVFAST                 4500        // delays for motorspeed
#define DRVMED                  8000
#define DRVSLOW                 12000
```

If a user wanted to change the motor speed, they would alter these settings by changing the values to a different number. Lower values increase the motor speed and higher values slow the motor speed.

Note: The lowest value is 1 and the highest value is 14000. It is critical that you use values within this range. For some boards the stepper might not speed at all if a low value is used.

To determine the correct value requires a lot of trial and error. Typically you would change a value, reprogram the controller, test the change, then repeat the process till you are satisfied with the result.

Starting with firmware 292+, there is an easier way to determine the correct values without having to reprogram the controller all the time.

Two new firmware calls are available that support the trial and error process. In firmware 292+ the process is simpler by letting you send a new delay setting (for the current motor speed setting) and then you can issue a move command to test the result. This can be repeated without reprogramming the controller till a suitable value is found. You can also change the speed setting and repeat this process to determine suitable values for other speed settings. Once you have the correct values, you can then edit the correct boards settings in the myBoards.h and reprogram the controller.

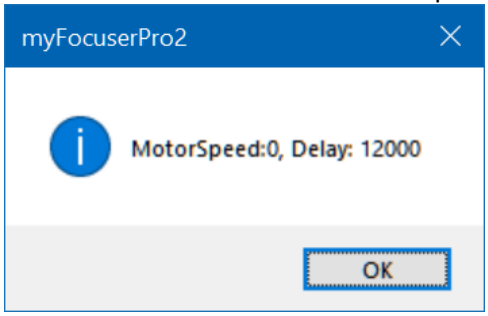
The following steps illustrate how to do this using the latest Windows myFP2 application.

STEP1: LOAD WINDOWS APPLICATION AND CONNECT TO THE myFP2 FOCUSER

STEP2: SET THE FOCUSER SPEED TO FAST

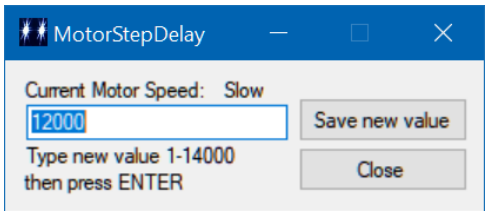
STEP3: SET THE MOTOR SPEED DELAY TO A NEW VALUE

Access the menu Extra Settings, MotorSpeed, Set Motorspeeddelay Value. The application reads the current value of the current motor speed setting (2=FAST) and displays this in a popup window.

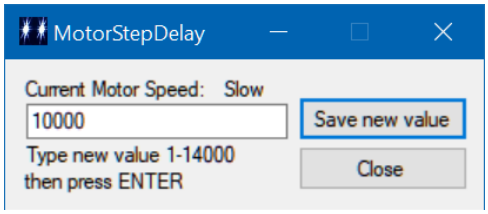


Click OK.

A form is now displayed allowing a new value for the motorstepdelay for FAST to be entered. Type in the new value then press the ENTER key [values are only checked if the ENTER key is pressed]



In this example a new value of 10000 is typed and the ENTER key is pressed. Click **Save new value** then click the **Close** button



The new value is sent to the controller.

STEP4: TEST THE MOTORSPEED USING THE MOVE BUTTONS

For a NEMA motor it is best to use Moves of 100 or 200 steps as this can easily be translated into motor movement [a 200 steps per revolution NEMA motor at HALF steps would go ½ a revolution]. If the motor pully is marked, degrees of rotation can easily be seen when moving.

After step3 setting a new stepdelayvalue, this step involves sending the motor a specified number of steps. Ensure that the motor steps smoothly and there is no missed steps [missed steps can occur if the stepdelayvalue is too low).

It is best to alter values around 500 to 1000 at a time. If the current value is 1, try 500, then try 1000 and so on till you find a suitable value.

STEP5: REPEAT STEP3 AND STEP4 TILL THE VALUE IS SATISFACTORY

When you have determined a suitable value, please write this value down.

STEP6: CHANGE THE MOTOR SPEED TO MEDIUM

STEP7: REPEAT STEPS 3 AND 4 TO ALTER THE STEPDELAY FOR MOTORSPEED MEDIUM

STEP8: CHANGE THE MOTOR SPEED TO SLOW

STEP9: REPEAT STEPS 3 AND 4 TO ALTER THE STEPDELAY FOR MOTORSPEED SLOW

You now have three values representing FAST, MEDIUM and SLOW delays for your stepper motor.

Use the Arduino IDE to edit the myBoards.h file and update the values for your board. In our example, we used the WEMOSDRV8825 board. The values which we found were acceptable were 350, 3000 and 12000.

In the myBoards.h file we look for the section that defines #define DRVBRD DRV8825HW203.
Below is the portion of those lines from the myBoards.h file.

```
#define DRVFAST                 4500         // delays for motorspeed
#define DRVMED                  8000
#define DRVSLOW                 12000
```

We make the necessary changes to DRVFAST, DRVMED and DRVSLOW, save the files, then reprogram the controller.

```
#define DRVFAST                 350         // delays for motorspeed
#define DRVMED                  3000
#define DRVSLOW                 12000
```

WHAT TO DO IF YOU LOSE YOUR FOCUSER SETTINGS

1. Unclamp the focuser coupler connecting the stepper motor to the focuser (or remove belt if using a pulley system)
2. Manually move focuser to initial 0 position (1/2 turn out as described above)
3. Power focuser and start myFP2M Windows application software (not ASCOM driver)
4. Set the step mode to what you used in the initial setup of the controller
5. Enter 0 as the focuser position and click the SET POSITION button
6. Clamp the focuser coupler so that the focuser motor can now drive the focuser
7. Enter your calculated maxStep value into the Maximum Position text box and click the Set Button
8. The focuser is now setup. Enter the position for reasonable focus into the Focuser Position text box and click the GOTO POSITION button to move the focuser to the focus position

UPGRADING FIRMWARE AND SOFTWARE

In MOST circumstances, you can apply the new firmware directly using the Arduino IDE and install the new application by running the setup program.

New releases often occur after you have built and set-up your myFP2M controller.

It is important to realize that new drivers, software and firmware (the Arduino code file) fix issues in previous versions as well as introduce new features. Make sure you save any downloaded firmware and software in a folder in case you want to revert back to a previous version.

Support is freely given concerning any current release. Please contact me and I will do my best to work with you to help resolve any issues you might have. Previous releases are not supported.

I know that updating is a pain, and that sometimes you might be hesitant to change something that works. Having said that, rest assured that newer releases are produced for sound reasons, and they might fix an issue that you might be having.

Sometimes the NOTICE with updates will ask you to uninstall the current software before installing the new release (this occurs with the Windows Application and sometimes the ASCOM driver).

When a new release of firmware or application software is released, before upgrading, save the settings to a file. Then install the update, reload the new application, connect to the controller, and then restore the settings.

Any changes that you make to the either the application or controller settings after a save and before a restore are lost. The save to file option saves BOTH the firmware settings and the software application settings at the time that the save to file option is run.

NOTE: I always move the focuser to position 0 before updating any new firmware. This means that once the new firmware is loaded and the new Windows/ASCOM installed, all I need to do is reset the current focuser position to 0 after reloading all the settings.

Updates for myFP2M software can be found at
https://sourceforge.net/projects/arduinoascomfocuserpro2diy/files/myFP2M/

STEP 1: SAVE SETTINGS

Ensure that the controller is connected. Run the Windows application.

Select from the Settings Menu the Save and Restore Focuser/App settings, then **Save to file**

STEP 2: UPDATE FIRMWARE OR APPLICATION

Close the application. Proceed to update either the firmware or application to the next release.

STEP3: RESTORE SETTINGS

Ensure that the controller is connected. Run the Windows application.

Select from the Settings Menu the Save and Restore Focuser/App settings, then **Restore from file**

Once the settings are restored from the file, the firmware settings are written to EEPROM, the controller is rebooted, and the application will exit (it needs to exit to update the settings).

When the application is restarted, the new settings will take effect.

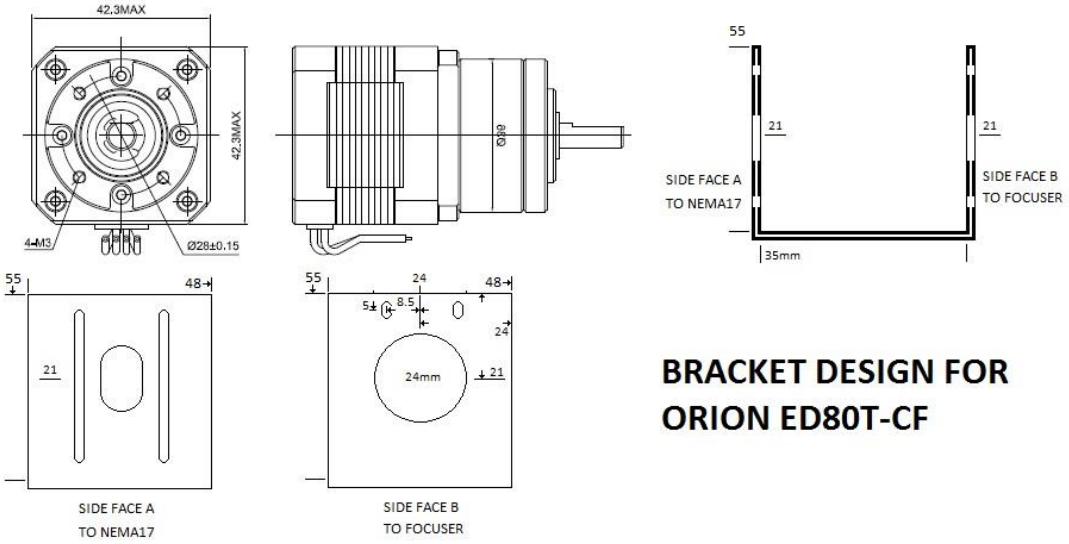
If you have any issues, please feel free to contact me for assistance/advice.

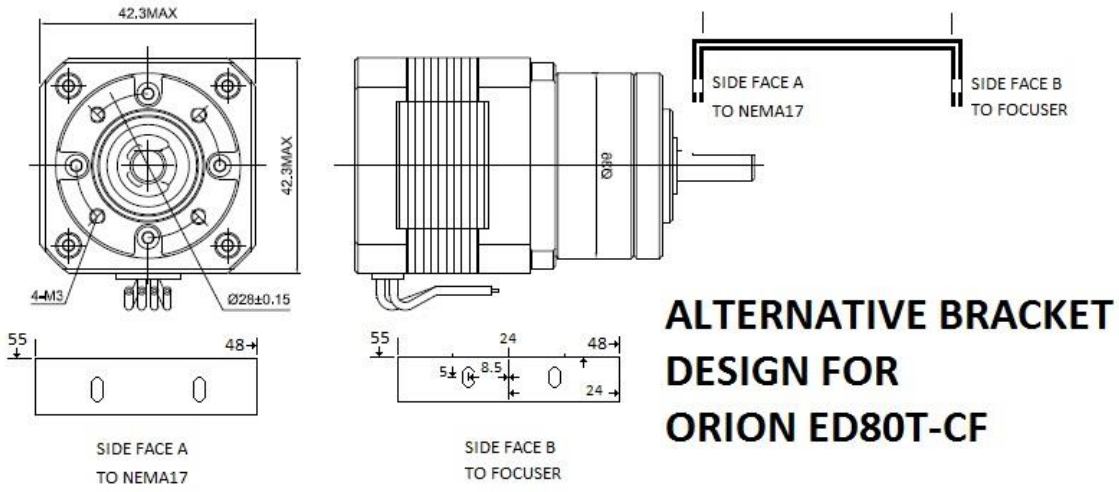
IT IS IMPORTANT THAT TO REALIZE THAT THE FIRMWARE AND WINDOWS APPLICATIONS OFTEN INTRODUCE NEW FEATURES AT THE SAME TIME. WHAT THIS MEANS IS YOU MUST UPDATE THE FIRMWARE AS WELL AS THE WINDOWS APPLICATION AND ASCOM DRIVER TOGETHER. YOU CANNOT RUN THE LATEST WINDOWS APPLCIATION OR ASCOM DRIVER ON FIRMWARE THAT MAY BE SEVERAL VERSIONS EARLIER.

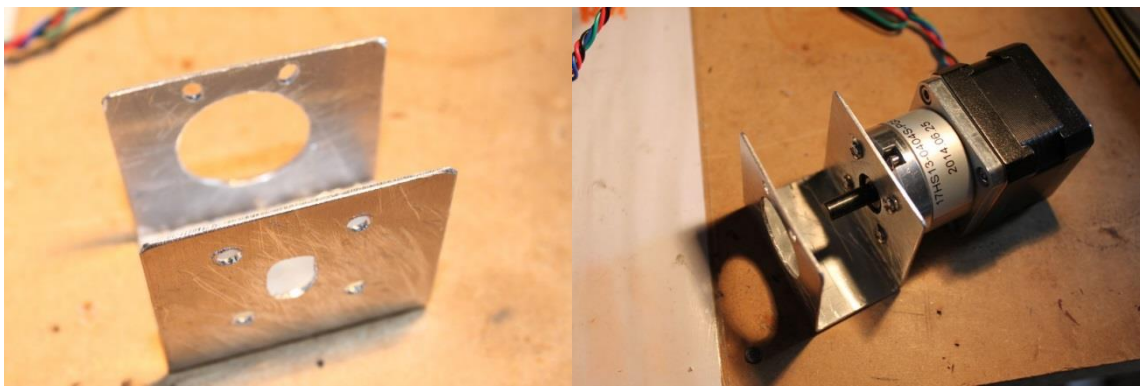
Please see the video
https://www.youtube.com/watch?v=cSvOVw8Djsw

OTHER POSSIBLE BRACKET DESIGNS

The NEMA17 connects via a home-made bracket to the focuser. The following diagram and photos show the U-shaped bracket that is used on the Orion ED80T-CF refractor.







USING A PULLEY AND BELT DRIVE

A belt reduction drive can be used to connect the stepper motor to focuser or DLSR camera lens. You can also put the belt over the focus knob and drive the knob using the belt.

© Speed_Mart, Pitch 2mm, 6mm wide, GT2 pulley and belt

The above pulley 14T has 14 teeth, and a 42T is fitted to the focuser shaft. A GT2 closed belt (length can be calculated once a mounting bracket has been designed and fitted – based on the distance between pulley center's) is used and this gives a ratio of 1:3. This means that using the NEMA14 0.9° motor at full steps which has 400 full steps per revolution, it will take 1200 steps to rotate the focuser (or lens) ONE complete revolution, which is more than adequate for most setups.

Below shows a DLSR setup using a longer 320mm GT2 belt.

If purchasing the pulley and belt separately, ensure that the pitch of the pulley matches that of the belt.

GEARS. WHAT OPTIONS DO I HAVE?

Gears are normally used to provide a reduction, so the small gear is fitted to the stepper motor and the larger gear (more teeth) is fitted to the focuser. This provides a reduction in step size as well as an increase in torque.

Gears are normally using in ratios like 1:2, 1:3 or 1:4
This means

1:1	1:2	1:3
15T	30T	45T
16T	32T	48T
18T	36T	54T
20T	40T	60T
30T	60T	
40T	80T	

3D PRINTING GEARS

Infill 100%
Nozzle 0.4mm
Layer Height 0.2mm
Speed SLOW

GEARS FOR NEMA MOTOR

NEMA motors have shaft sizes like 5mm and 6mm in diameter. In 3D printing, the bore size in the gear is made slightly large to accommodate shrinkage in the printing process. Gears are provided with a M3 hex grub screw [which interlocks with an M3 nut inserted into the base of the gear] to secure the gear to the stepper motor shaft.

All measurements in mm, designed to fit standard Reprap 3D printer GT2 belt (2mm pitch)

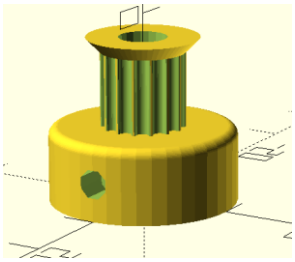
height of tooth part = 8mm
height of hub part = 6mm

measurements below are approx

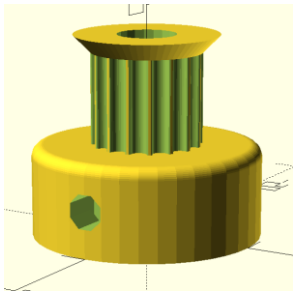
	Pulley D	Hub D
15T	9	20
16T	10	20
18T	11	20
20T	12	20
30T	18	20
32T	19	20
36T	22	20
40T	24	25
45T	26	25
48T	28	25
54T	34	25
60T	38	30
72T	45	30
80T	50	30

30T-80T has 2xM3 hex screws

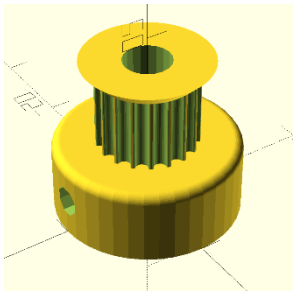
3D PRINTED GEARS NEMA



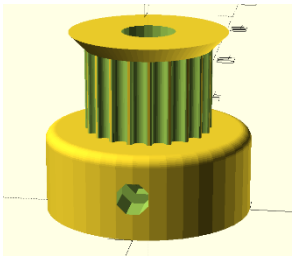
15T



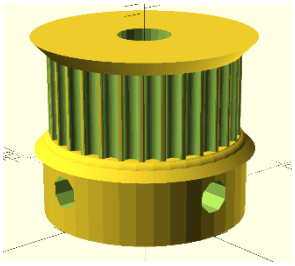
16T



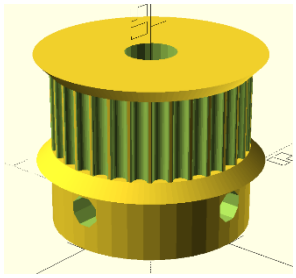
18T



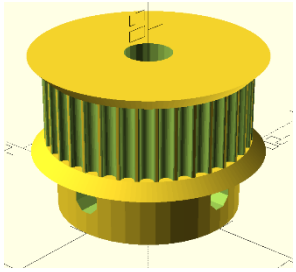
20T



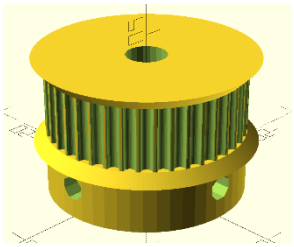
30T



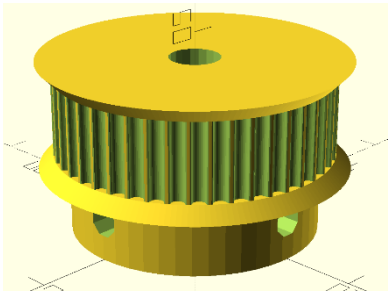
32T



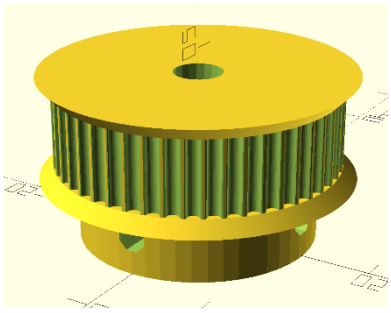
36T



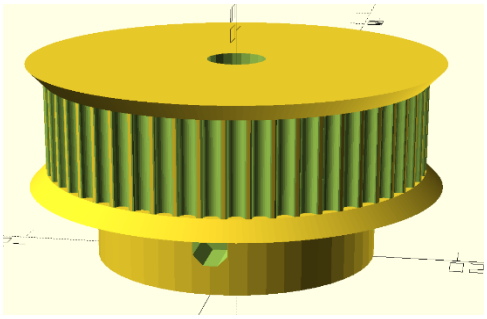
40T



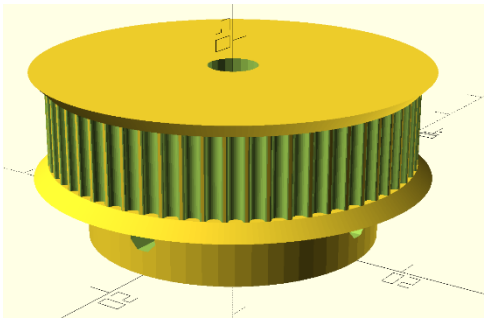
45T



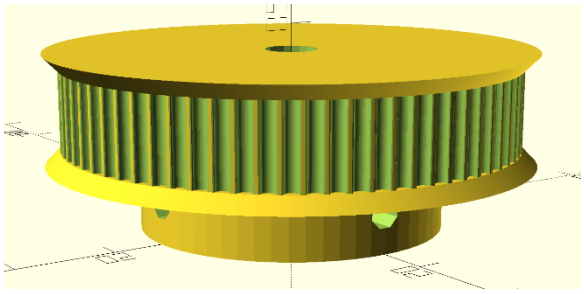
48T



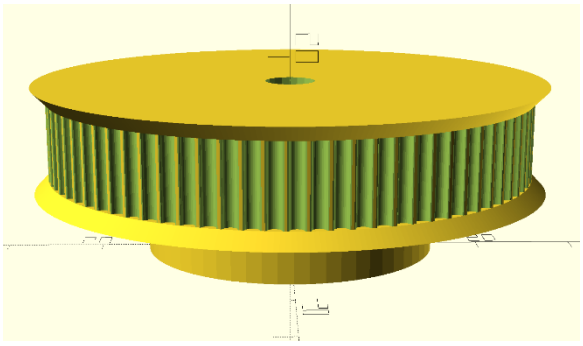
54T



60T

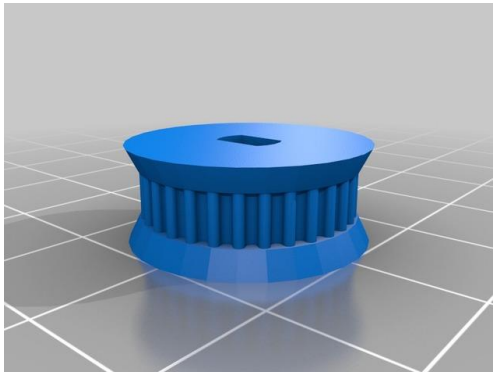


72T

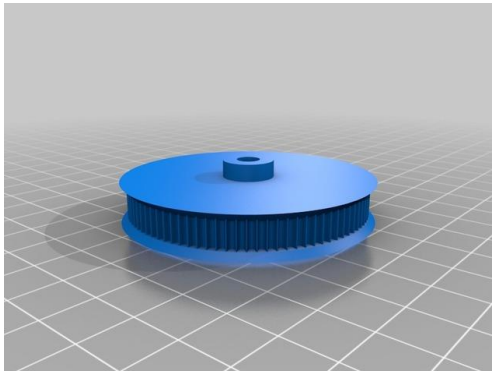


80T

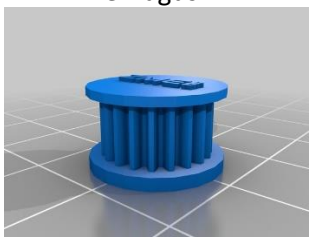
3D PRINTED GEARS 28BYJ-48



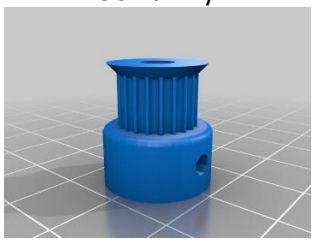
https://www.thingiverse.com/thing:2477530
TheMagusMX



http://www.thingiverse.com/thing:2401712
90T tximy

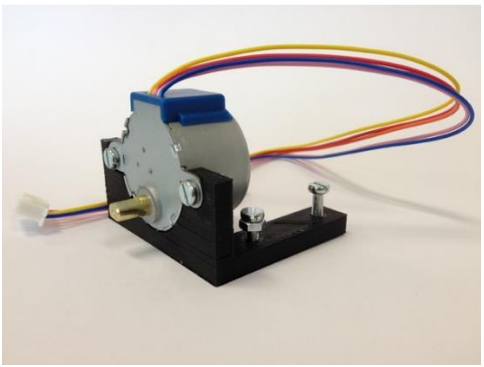


https://www.thingiverse.com/thing:2868283
Zmei242

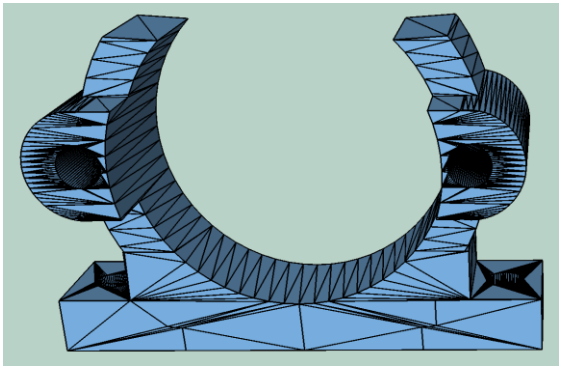


http://www.thingiverse.com/thing:2139089
Pulley GT2 20 Teeth Tuliomfaria

3D PRINTED BRACKET 28BYJ-48



https://www.thingiverse.com/thing:2514652
ewoud_design



http://www.thingiverse.com/thing:1542952
Druidosh

https://www.thingiverse.com/thing:2477530
TheMagusMX

HOW DO I CALCULATE THE LENGTH OF BELT REQUIRED FOR GEARS?

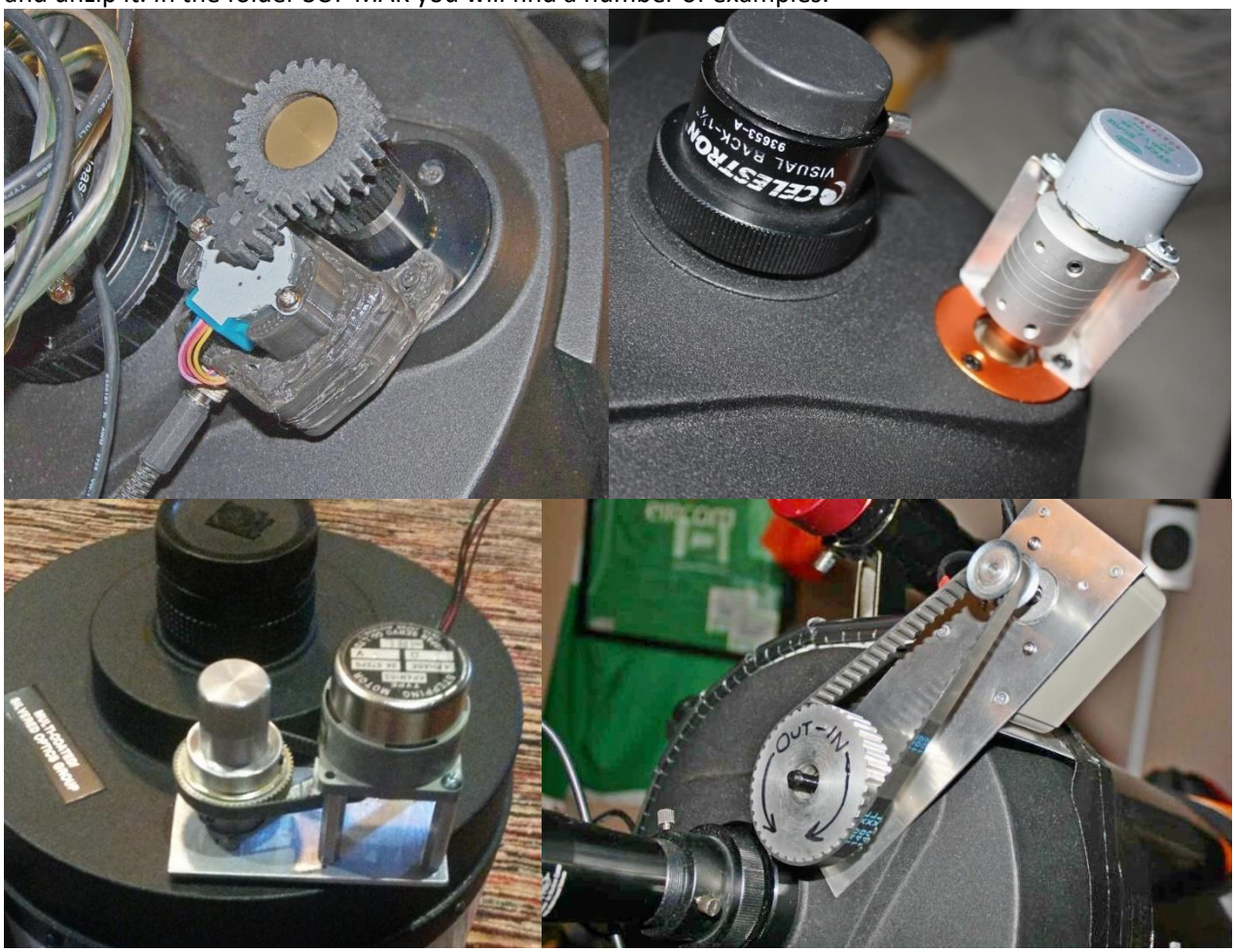
The following spreadsheet is a belt length calculator. If you are using gears, use this calculator to determine the length of the belt you require

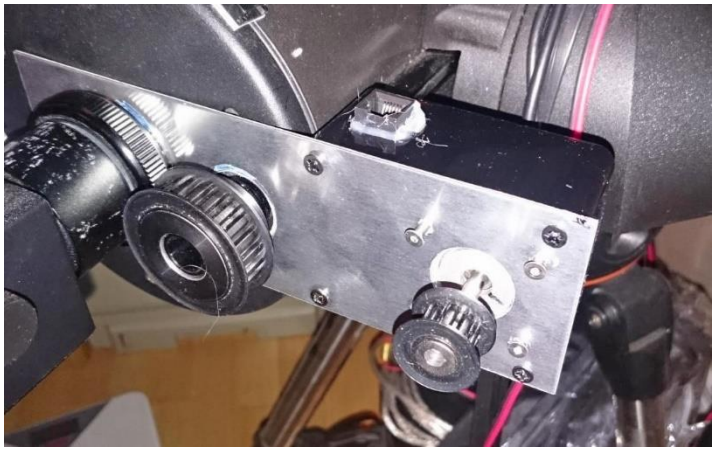
https://sourceforge.net/projects/arduinoascomfocuserpro2diy/files/Documentation/Belt%20Length%20Calculator.xlsx/download

HOW DO I CONNECT MY STEPPER MOTOR ON AN SCT?

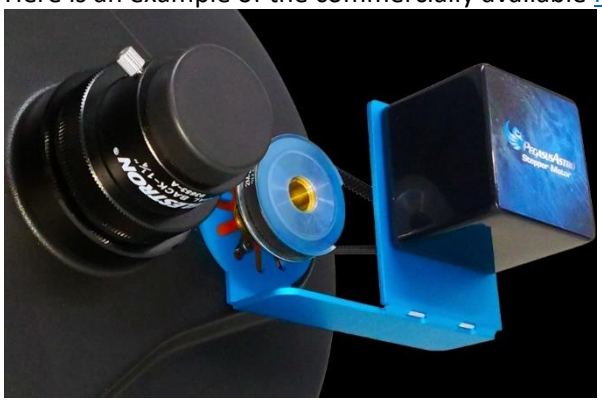
There are many images on the Internet showing users innovative solutions to attaching a stepper motor to an SCT. Download this file
https://sourceforge.net/projects/arduinoascomfocuserpro2diy/files/MOUNTING%20BRACKETS%20AND%20IDEAS/User%20Builds%20and%20Bracket%20Examples.zip/download

and unzip it. In the folder SCT-MAK you will find a number of examples.





Here is an example of the commercially available Pegasus Astro controller bracket



BRACKET MOUNTING IDEAS

https://sourceforge.net/projects/arduinoascomfocuserpro2diy/files/Documentation/Mounting%20Examples.zip/download

There are lots of pictures from contributors in how to mount the stepper motor on various telescopes. These are is in the folder **Mounting Bracket Examples** included with the firmware zip file.

COMPARING STEPPER MOTORS

	Weight	Steps per rev	Current	Holding Torque	Size (mm)
28BYJ-48 5V	30g	2048/4096	160-320mA	3.43N.cm	35x20
NEMA14 0.9	120g	400 at Full steps	400mA	11Ncm	35x35x28
		3200 at 1/8 stepping			
NEMA 17 PG5	500g	1026 at Full steps	400mA	121.2Ncm	42x42x61
NEMA 17 PG27	500g	5370 at Full Steps	400mA	300Ncm	42x42x70

Do NOT use stepper motors with a current rating higher than 500mA per phase coil

PARTS LIST myFP2M-ULN2003 (Rev7 PCB)

myFP2M-ULN2003	Quantity
Arduino Header Pins 40, Breakable 2.54mm 1pc of 40 pins	1
myFP2E-ULN PCB	1
M2*3mm hex bolt stainless steel (secure PCB)	2
DB9 female socket panel mount	1
XH2.54 5p header cable	1
Posts brass M3*6*5 Male-Female spacer hex	2
M3 hex nut stainless steel	2
M3 Spring washers, Stainless Steel 316	2
Arduino Nano 328P CH340G	1
15P Headers for Nano	1
LM7808 Voltage Regulator TO-220 package	1
TO-220 Heat sink + screw 20*15*10	1
Ceramic Capacitor 0.33uf	1
Ceramic Capacitor 0.1uf	1
GX 12mm 3P female/male connector pair	1
XH2.54 3P header cable	1
1N5408 3A Diode	1
PTC Polymer Resettable Fuse 3A	1
Power Socket Female Panel Mount 12V 2.1mm x 5.5mm	1
XH2.54 2p header cable	1
Power Plug Male 2.1mm x 5.5mm	1
1.5m cable for 12V power 2 core 5A Black/Red	1
12V Prewired LED Red 3mm with Bezel	1
5V Prewired LED Blue	1
5V Prewired LED Green	1
3mm Bezel	2
XH2.54 2p header cable	3
On/Off toggle switch press fit (power)	1
XH2.54 2p header cable	1
Mini Miniature On/Off 1P2T 3-Pin 2 Position Slide Switch SPDT	1
XH2.54 2p header cable	1
M2*7mm hex bolt stainless steel (secure SPDT)	2
M2*5*0.3 washer stainless steel	2
M2 nut	4
Capacitor 470uF 35V	1
Motherboard shunt jumpers	1
Buzzer Continuous Tone 5V	1
XH2.54 2p header cable	1
3D Printed Case	1
M2 x 7mm Hex bolt stainless steel (secure cover)	4
M2*5*0.3 mm washers	4
ULN2003 Board and 28BYJ-48 5V motor	1

USB Cable Type B Mini Male to USB Type A 1.5m Shielded	1
Heat shrink tube 2mm 2m length	1
CDROM	1
CDROM Paper Flap Sleeve	1
TEMPERATURE PROBE	
DS18B20 Temperature Probe Waterproof with 1m cable	1
DS18B20 Temperature Probe Waterproof with 2m cable	1
STEPPER MOTOR	
28BYJ-48 Unipolar 4-phase and ULN2003 driver	1
RS232 DB9 Female socket plus shell	1
Ferrite EMI Noise Suppressor Clip On 7mm	1
3mm Allen key	1
Washers, Stainless Steel, M3*7*0.5	2
Screws M4*10mm	2
M4 lock nut	2
STEPPER MOTOR EXTENSION CABLE	
RS232 crimp male	1
RS232 crimp female	1
Flat ribbon cable 1m (10p) 1.27mm spacing	1

Flexible Shaft Coupler 5-6mm (optional-specify when ordering)	1
M1.5 Allen Key	1

PARTS LIST myFP2M-DRV8825 (Rev7 PCB)

myFP2M-DRV8825	Quantity
Arduino Header Pins 40, Breakable 2.54mm 1pc of 40 pins	1
myFocuserPro2-DRV PCB	1
M2*3mm hex bolt stainless steel (secure PCB)	2
DB9 female socket panel mount	1
XH2.54 4p header cable	1
Posts brass M3*6*5 Male-Female spacer hex	2
M3 hex nut stainless steel	2
M3 Spring washers, Stainless Steel 316	2
DRV8825	1
TL-Smoother	1
Capacitor 100uF 35V	1
8P Headers for DRV8825 2.54mm	2
Heat sink 10mmx13mmx13mm DRV8825	1
Arduino Nano 328P CH340G	1
15P Headers for Nano 15P 2.54mm	2
LM7808 Voltage Regulator TO-220 package	1
TO-220 Heat sink + screw 20*15*10	1
Ceramic Capacitor 0.33uf	1
Ceramic Capacitor 0.1uf	1
GX 12mm 3P female/male connector pair	1
XH2.54mm 3p header cable (for temperature probe)	1
1N5408 3A Diode	1
PTC Polymer Resettable Fuse 3A	1
12V Prewired LED Red 3mm with Bezel	1
5V Prewired LED Blue	1
5V Prewired LED Green	1
3mm Bezel	2
XH2.54 2p header cable	3
On/Off toggle switch press fit (power)	1
XH2.54 2p header cable	1
Mini Miniature On/Off 1P2T 3-Pin 2 Position Slide Switch SPDT	1
XH2.54 2p header cable	1
M2*7mm hex bolt stainless steel (secure SPDT)	2
M2*5*0.3 washer stainless steel	2
M2 nut	4
Capacitor 470uF 35V	1
Buzzer Continuous Tone 5V	1
XH2.54 2p header cable	1
Power Socket Female Panel Mount 12V 2.1mm x 5.5mm	1
XH2.54 2p header cable	1
Power Plug Male 2.1mm x 5.5mm	1
1.5m cable for 12V power 2 core 5A Black/Red	1

3D Printed Case	1
M2*7mm Hex bolt stainless steel (secure cover)	4
M2*5*0.3 mm washers	4
USB Cable Type B Mini Male to USB Type A 1.5m Shielded	1
Heat shrink tube 2mm 2m length	1
CDROM	1
CDROM Paper Flap Sleeve	1

TEMPERATURE PROBE	
DS18B20 Temperature Probe Waterproof with cable 1m	1
DS18B20 Temperature Probe Waterproof with 2m cable	1

STEPPER MOTOR	
NEMA14 Bipolar 11Ncm Stepper Motor (0.4A, 10V)(Shaft=5mm)	1
RS232 DB9 male socket plus shell	1
Ferrite EMI Noise Suppressor Clip On 7mm	1
Plastic cable tie 200mm	1
2.5mm Allen key	1
Washers, 3D printed	4
Washers, Stainless Steel, M3*7*0.5	4
Screws M3*10mm	4

STEPPER MOTOR EXTENSION CABLE	
RS232 crimp male	1
RS232 crimp female	1
Flat ribbon cable 1m (10p) 1.27mm spacing	1

Flexible Shaft Coupler 5-6mm (optional-specify when ordering)	1
M1.5 Allen Key	1