

SGL Observatory Automation

ASCOM Motor Focuser Control

Getting Started Guide

Written by Christian Guenther (yesyes)

Document version V1.0 – 20 September 2011

Introduction

SGL Observatory Automation is an open source project to control the creation of software, hardware and interfaces to automate an astronomical observatory using the Arduino open standard, with ASCOM support for appropriate devices.

This guide deals with the Focuser Control part of the project. This allows you to control your DIY motor focuser via the ASCOM standard or via a stand-alone Windows application.

What you need

- Hardware

- A bipolar stepper motor with 1.8 degree steps / 200 steps per revolution or better (less degrees / more steps are better)
- One of the supported stepper motor driver circuits
- One Arduino micro controller board

- Software

- **3rd party software**
 - The Arduino Programming Software
<http://arduino.cc/en/Main/Software>
 - The ASCOM Platform 5.5 or higher
<http://ascom-standards.org/>
- **SGL Observatory Automation Software**
(download from Files section of the SGL Observatory Automation Yahoo! Group)
 - The software that will run on the Arduino board. This is called a sketch and comes in the form of .pde files. These files are text files containing the source code of the sketch. This needs to be compiled and uploaded to the Arduino via USB using the Arduino software.
Files -> SGL Software -> Focuser -> Current Version
SGL_Focuser_Driver_nolcd_2_0_0F.zip.pde
 - The libraries are required to compile the Arduino sketch
Files -> SGL Software -> Focuser -> Current Version
Libraries.zip
 - The SGL Focuser ASCOM driver. This is the software that needs to be installed on the PC that you want to use to control the focuser. This allows you to control the focuser through 3rd party astronomy software that supports ASCOM focuser control.
Files -> SGL Software -> Focuser -> Current Version
SGLAscomFocuser V2_00_00B Setup.exe
 - (optional) The Focuser Control Stand-alone application. This is a separate Windows application that allows you to directly control your focuser without ASCOM support.
Files -> SGL Software -> Focuser -> Current Version
SGL_Focus_StandaloneV1_00_03.zip

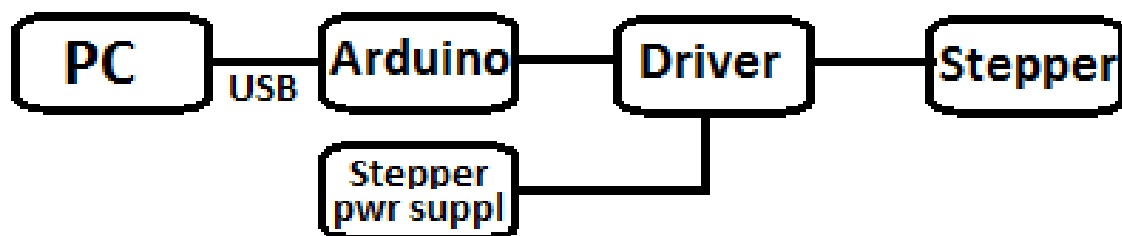
- Skills

- Mechanical skills to mount the stepper motor onto your telescope focuser model. This can be done in many different ways and depends on your specific focuser model.
- Some soldering will be required. Some of the circuits of this project are quite small and sensitive; specifically the Arduino and the EasyDriver. If your soldering skills are a bit rusty or lacking, then I recommend practicing on something less sensitive first.

The Hardware

Overview

This is an overview of how everything needs to be connected



The connection between the PC and the Arduino is a simple USB cable. The Arduino will be powered from the USB bus. The stepper driver connects to the Arduino via a few signal lines and a Ground line (GND). The wires coming out of the stepper motor then connect to the driver board. The power supply for the stepper motor also connects to the driver board.

The Arduino can be any reasonably modern model. The Arduino Duemilanove (Italian for 2009), Uno and Mega are popular. I personally have chosen an Arduino Nano because it's quite small but has the same functionality as the 2009. If you have not dealt with an Arduino before, I recommend having a look at the Arduino website <http://arduino.cc> and get an idea what it is all about.

As for the driver, 3 types are supported: the EasyDriver board, the L293 driver chip and the LadyAda AFmotor board.

I personally have only used the EasyDriver, so I will concentrate on this driver from now on. The procedure will be different for other drivers.

The stepper motor should be a bipolar stepper motor with 1.8 degrees per step (less is better). That's the equivalent of 200 steps per rotation (more is better). The stepper can have 4 or 6 wires but only 4 will be used. These steppers can be salvaged from old printers for example. You can also find some cheap ones on Ebay. You can ask on the Yahoo! Group if you are not sure whether a particular stepper will be suitable.

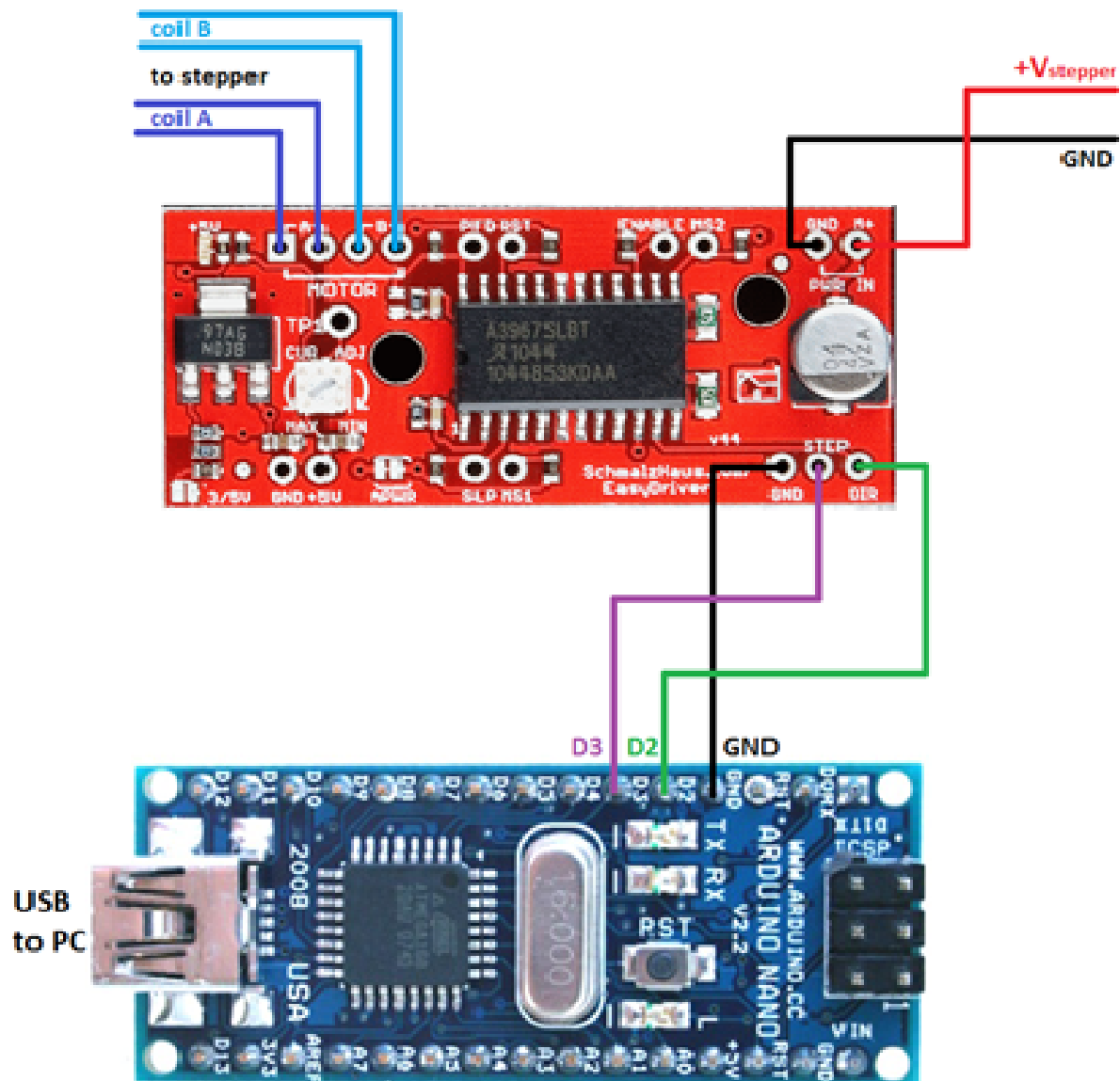
The power supply depends on the stepper. The EasyDriver can handle up to 30Volts. Go a little higher than what the stepper is rated for. That will make sure that the stepper has enough torque to turn the focuser. I'm using a 12V power supply for my 5V rated stepper.

Connecting everything up

On the Arduino side you need to connect 3 wires; one on a GND pin and 2 on digital I/O pins. The default I/O pins are pin 2 for direction and pin 3 for step signals. You can choose different pins and specify the pins you used in the Arduino sketch later.

These 3 wires go to the EasyDriver's DIR, STEP and GND pin. Then connect the stepper power supply to the EasyDriver. The 4 wires of the stepper motor also connect to the EasyDriver. More on that later.

This is what it should look like:



This particular diagram shows an Arduino Nano. The pin names are the same on all Arduinos.

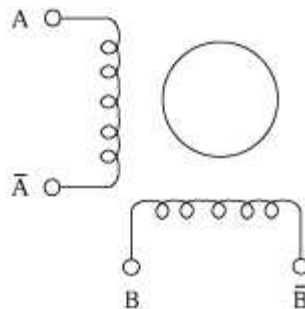
Connecting the stepper

First a word of warning: You must never disconnect the stepper motor while the EasyDriver is powered. Doing this will destroy the driver chip. Make sure the wires don't come loose while you are testing.

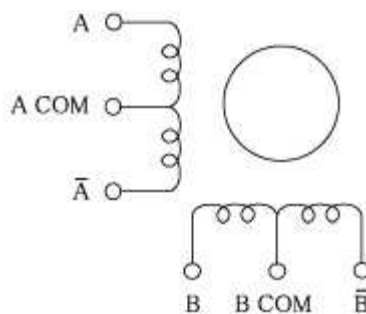
If you have a datasheet for your stepper, then find out which pair of wires is one coil and which pair is the other coil. Connect one pair to A on the EasyDriver and the other pair to B. It does not matter which pair is which. If you connect them the other way around, the stepper will just rotate in the other direction. This can be corrected in the Arduino sketch. Also, each pair does not have polarity. It does not matter which way you connect each pair. If you have a 6 wire stepper, then don't use the middle wire of each coil.

If you do not have a datasheet, you can easily find out the pairs using a continuity tester or better an Ohm meter (required when you have a 6 wire stepper).

For a 4 wire stepper just take the wires and measure continuity or resistance. If you find 2 wires with a low resistance or continuity, that's one coil. The other 2 wires are the other coil.



The 6 wire stepper is a little more tricky. You will need to measure the resistance. Continuity is not enough. You need to find pairs with higher resistance than other combination of wires. In the diagram below there will be higher resistance between A and A than between A and A COM (because the coil is longer). With this type of stepper you will not use A COM and B COM.



Once you know which wires are pairs, connect one pair to the A terminals on the EasyDriver and the other pair to the B terminals.

Mounting the Stepper on the Focuser

There are several ways to connect the stepper shaft with the focuser shaft. You can use a direct coupling; ideally a flexible one. Or you can use gears or belt and pulleys. This is up to you and how you can attach the stepper to the focuser. If you use gears or belt you can choose a gear ratio; the higher gear on the focuser. A higher gear ratio allows for finer adjustment but it takes much longer for the focuser to travel longer distances. I find a 2:1 ratio with a 0.9 degree stepper fine enough.

There is really not much I can suggest how you mount the motor to the focuser. You will need to find some way of attaching a metal bracket to the focuser that holds the stepper in the right place. Make sure you know what every screw on your focuser is for before you loosen any of them trying to find an attachment point.

This how I solved the problem:



A solution using a universal coupling:-

And one with a direct coupling onto the shaft (commercial item):-

The Software

Installing the Arduino Software

You can skip this step if you already have the Arduino Software installed from a different project.

The Arduino Software is an IDE (Integrated Development Environment) that runs on Windows, Mac OS X and Linux. This is where you would develop software that runs on an Arduino and it is also used to upload the sketch onto the Arduino.

For this project the Arduino Software will only be used to make small source code modifications and then upload the sketch to the Arduino board. It is not required when using the focuser. Therefore you don't need to install it on the same PC that you will use to control the focuser. It is, of course, no problem to install it all on the same machine.

First you need to download the Arduino Software from here:

<http://arduino.cc/en/Main/Software>

Then install and configure the Arduino Software. This is explained on the Arduino website here:

<http://arduino.cc/en/Guide/HomePage>

Feel free to follow the examples provided on the Arduino Website. The "blink" example sketch is a good and popular way to check if everything is set up and working correctly.

Copying the Libraries

The libraries required to compile the Arduino sketch. They contain extra bits of source code that the SGL Focuser sketch refers to. All you need to do is to download the libraries from the Yahoo! Group Files section and copy them to where you installed the Arduino Software.

On the website of the SGL Observatory Automation Yahoo! Group go to

Files -> SGL Focus control Easy Driver -> Driver Version 2_00_00

There download **Libraries.zip**

Then open the zip file and copy all its content to the "libraries" folder of the Arduino Software. Let's say you installed the Arduino Software under D:\arduino-0021\ then copy the content of Libraries.zip to D:\arduino-0021\libraries. Make sure you keep the subfolders exactly as they are in the zip file.

Make sure that the Arduino Software is not running while you copy the libraries. If it was running, then close it and start it again for it to recognise the new libraries.

Copying the Sketch

On the website of the SGL Observatory Automation Yahoo! Group go to

Files -> SGL Software -> Focuser -> Current Version

From there download **SGL_Focuser_Driver_nolcd_2_0_0F_zip.pde**

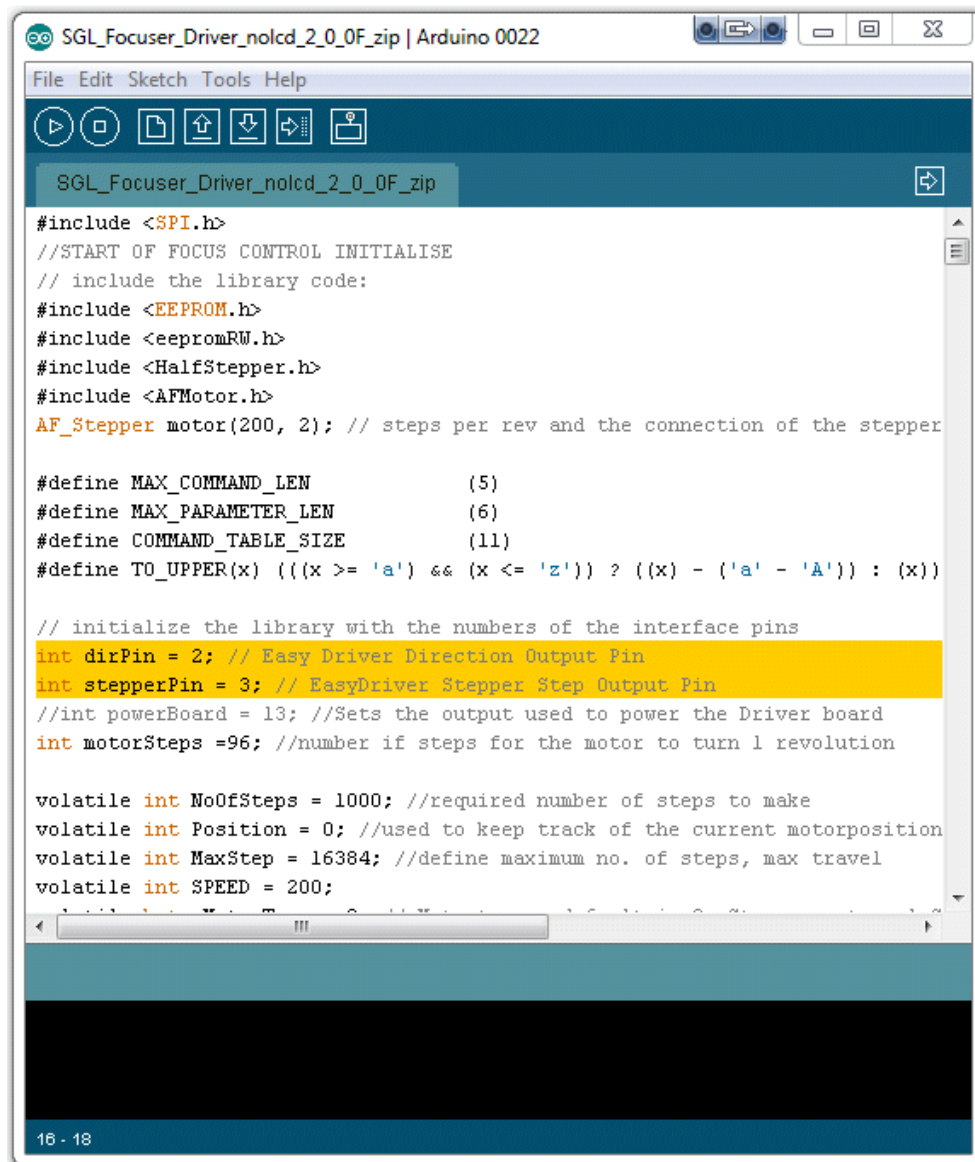
Then open the location where you installed the Arduino Software and go to the subfolder Sketchbook. If this folder does not exist, create it. Under Sketchbook create a new folder with the same name as the .pde file but without the .pde extension. In this case the folder name will have to be **SGL_Focuser_Driver_nolcd_2_0_0F_zip**. Then copy the file **SGL_Focuser_Driver_nolcd_2_0_0F_zip.pde** into that folder.

Modifying and Uploading the Sketch

Depending on how exactly you connected everything up you may need to make some small modifications to the source code of the sketch. You may need to change the I/O pin numbers in the sketch if you used non-default pins. You may also need to change the direction of the stepper depending on how you connected the stepper coils to the EasyDriver.

You need to check and maybe modify the I/O pin numbers now. The stepper direction and a few other things may need to be modified later.

Open the sketch in the Arduino Software.



Locate the following 2 lines

```
int dirPin = 2; // Easy Driver Direction Output Pin
int stepperPin = 3; // EasyDriver Stepper Step Output Pin
```

Edit the pin numbers to reflect your wiring. For example if you connected the DIR input of the EasyDriver to Arduino digital pin 12 and the STEP input to pin 9, edit these 2 lines the following way:

```
int dirPin = 12; // Easy Driver Direction Output Pin
int stepperPin = 9; // EasyDriver Stepper Step Output Pin
```

If you used an analogue I/O pin, prefix the pin number with an A. For example if you connected the DIR input of the EasyDriver to Arduino analogue pin 4 and the STEP input to analogue pin 5, this is what the 2 lines should look like:

```
int dirPin = A4; // Easy Driver Direction Output Pin
int stepperPin = A5; // EasyDriver Stepper Step Output Pin
```

No need to modify anything else at this moment. Click on the Save button in order to save the modifications you made. Now make sure the Arduino board is connected to the PC via USB cable and the correct COM port and Arduino board type selected as explained on the Arduino website. Then

click the Upload button. This will compile the sketch into binary format and upload it to the Arduino board. When the upload has finished the Arduino will start running the sketch immediately.

If you get any errors at this stage, check on the Arduino website for a solution and/or ask for help on the Yahoo! Group.

Installing the Stand-alone Application

The Stand-alone application offers a way of controlling the focuser directly without other software. This can be used to test your build. You can also use this application to control your focuser while observing / imaging if you do not want to use other ASCOM compatible software.

On the website of the SGL Observatory Automation Yahoo! Group go to

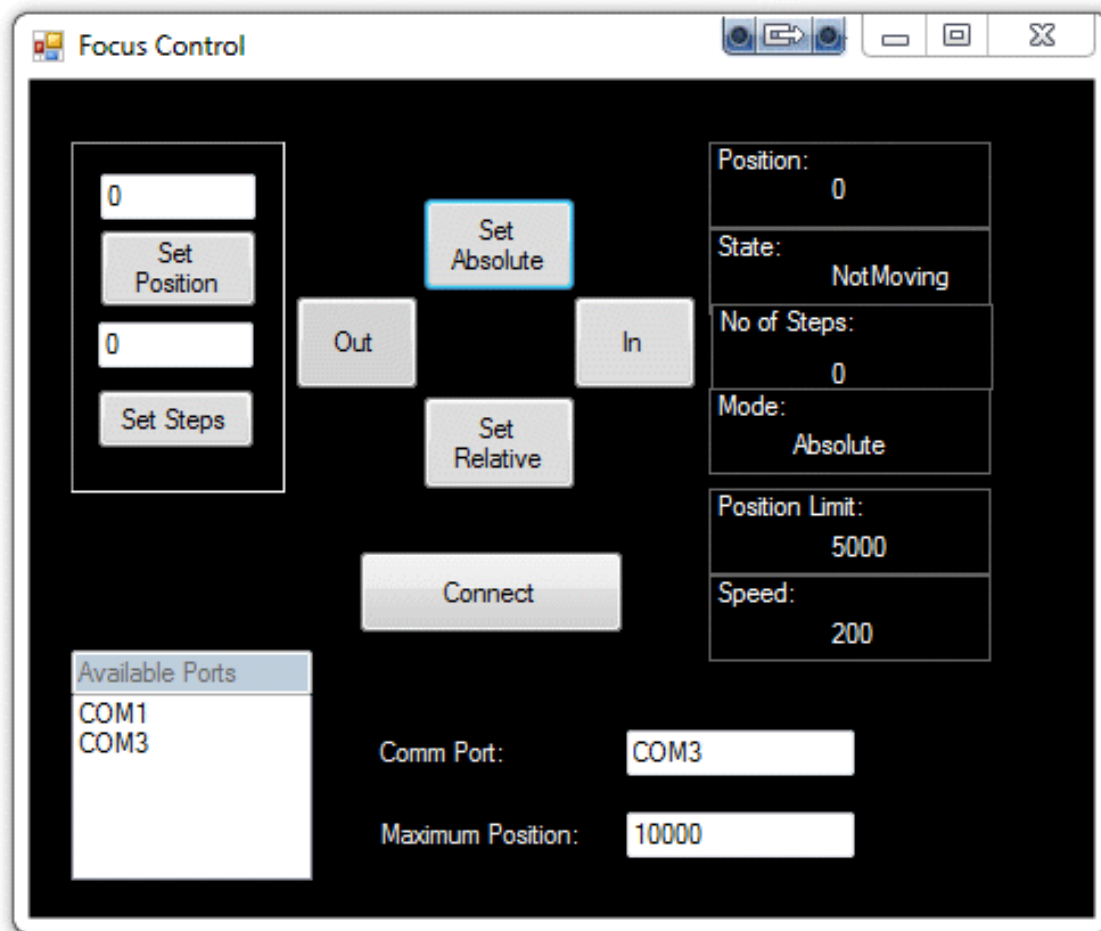
Files -> SGL Focus control Easy Driver

There download **SGL_Focus_StandaloneV1_00_03.zip**

Then extract the files in the zip archive and run setup.exe in order to install the Stand-alone application.

Testing your stepper using the Stand-alone Application

Now it's time to test what you have built and done so far. Connect everything up as described and plug in the Arduino to the PC using the USB cable. Then start the Stand-alone application you just installed. Make sure the Arduino Software is closed at this point because it will keep the COM port to the Arduino open otherwise. A COM port can only be opened in one application at a time.



Now select the COM port that is assigned to your Arduino board and click Connect. Then enter a value in the field above Set Steps. 100 is a good starting point. Then click Set Steps and then on Set Relative. Now when you click on In or Out the stepper should make the number of steps in that direction. 100 steps is probably a very small movement. To make bigger steps, enter a higher number and click Set Steps again.

If the stepper is moving and does so in the right direction, then you're done. The only thing left to do is to install the SGL Focuser ASCOM Driver if you want to use that.

If the stepper does move but in the wrong direction, i.e. the focuser moves out when you click the In button, then you can correct for this in the Arduino sketch. Close the Stand-alone application (for it to close the COM port) and load the sketch into the Arduino Software again. (see **Modifying and Uploading the Sketch**).

In the sketch find the following line

```
boolean Direction = true; // True is one way false is other. Change to false if motor is moving in the wrong direction
```

Change it to

```
boolean Direction = false; // True is one way false is other. Change to false if motor is moving in the wrong direction
```

Then upload the sketch to the Arduino board again. . (see **Modifying and Uploading the Sketch**).

Now close the Arduino Software again and start the Stand-alone application and test again. The stepper should now move in the opposite, correct direction.

If you experience any other issues at this point (for example the stepper is not moving at all) please ask for help on the Yahoo! Group.

Installing the ASCOM Platform

ASCOM stands for Astronomy Common Object Model and represents a standard way of controlling various types of astronomy equipment from a Windows PC. It consists of the equipment independent ASCOM Platform software and ASCOM drivers for each device you want to control.

If you are already controlling other equipment, such as your telescope mount, from your PC, then it is quite likely that you already have the ASCOM Platform installed. Check if it is version 5.5 or higher. If yes, then you're fine and you can skip to the next step. If it is older than version 5.5, then you need to upgrade.

If you don't have the ASCOM Platform installed or you need to upgrade, download the ASCOM Platform software from here:

<http://ascom-standards.org/Downloads/Index.htm>

Installation is straight forward. Just run the downloaded file and follow the on-screen instructions.

Installing the SGL Focuser ASCOM Driver

The SGL Focuser ASCOM Driver is the software that tells the ASCOM Platform how to control this particular focuser. This allows any software that supports ASCOM focusers to control the motor focuser you are building without knowing the specifics of this focuser.

On the website of the SGL Observatory Automation Yahoo! Group go to
Files -> SGL Focus control Easy Driver -> Driver Version 2_00_00
From there download **SGLAscomFocuser V2_00_00B Setup.exe**

Then run this file in order to install the driver. This will just install the ASCOM driver for your focuser. You will not see a new application installed.

Once the driver has been installed you can open the application that supports the ASCOM focuser. This application should have an option somewhere where you can select which focuser to control. The driver you just installed should appear in the list of available focusers as **SGLAscomFocusControl**

Connecting your application with the SGL focuser causes a window to open that looks very similar to the Stand-alone application. You can now control your focuser from your astronomy software and also from this window.

That's it. You're done.

Again, if you have any problems or questions before or while building your motorised focuser, please ask on the Yahoo! Group that you can find here:

http://tech.groups.yahoo.com/group/sgl_observatory_automation/

Good luck with your build and clear skies for actually using it.