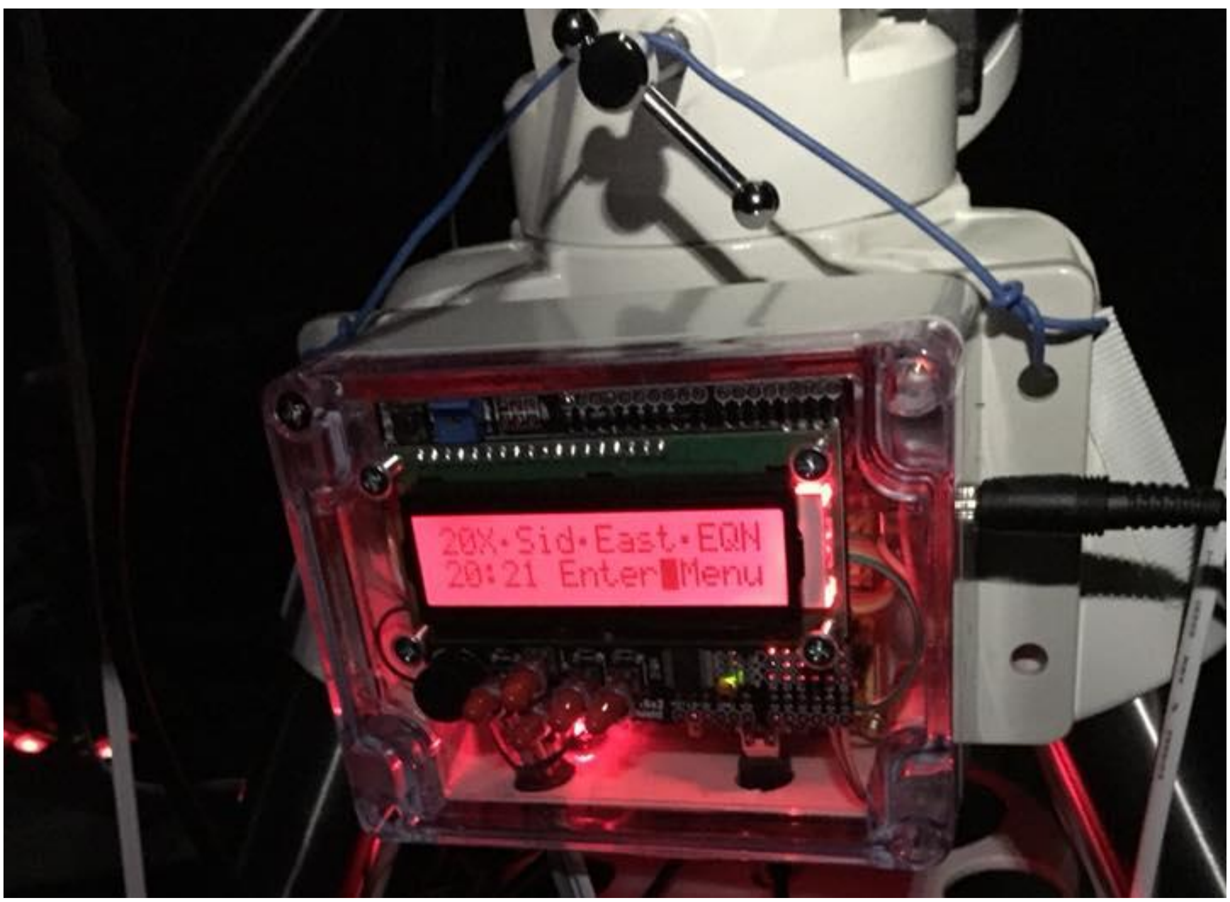
Z-Field Dual Axis Controller
Assembly Instructions and User's Manual

Version 1.04

By Kevin Cobble

Z-Field Learning, LLC



Disclaimer

The instructions in this document, any software, any online information, and any parts or documentation are released into the public domain on an As Is basis with no warranties or guarantees expressed or implied. While the author has taken steps to try to insure the accuracy of the information provided, users intending to use this information, software and/or purchased parts do so at their own risk and usage of the product and/or code is deemed to be acceptance of this risk. The author accepts no responsibility for damage caused to any parts, equipment or self by your use of these ideas, documentation and/or software. This includes any loss of income and any or all other loses that may be incurred.

This project involves the use of products and construction practices that involve some risk to the user. This includes soldering irons, chemical adhesives, and electrical shock hazards. The user assumes the liability of insuring all safety procedures for all activities are adhered to. This manual does not include full instructions or safety procedures for the use of devices used in construction including, but not limited to, soldering irons, hot melt glue guns, chemical adhesives, scissors, pliers, screwdrivers, hobby knives, drill press, hand drills, or any other tools or processes. The user assumes all risks associated with the use of these tools as well as any liability involved in their use.

The procedures outlined in this manual should only be performed by an adult or under adult supervision.

Copyright Restrictions

Table of Contents

Disclaimer ... ii
Copyright Restrictions ... ii
Figures ... v
Tables ... vi
Preface ... vii
1 Introduction ... 1
2 Important First Steps ... 2
2.1 Optional Equipment ... 4
2.2 Operational Notes ... 5
3 Setting up the Arduino IDE ... 6
3.1 Downloading the Z-Field Dual Axis Controller Arduino Code ... 7
4 Building the Z-Field Dual Axis Controller ... 8
Step 1: Drill the holes in the plastic enclosure. ... 8
Step 2: Solder the Diode to the Motor Controller Board ... 10
Step 3: Prepare the wire ... 10
Step 4: Wire the Power Harness ... 11
Step 5: Solder Power Harness to the Motor Control Board ... 12
Step 6: Install the motor boards ... 13
Step 7: Install the power harness ... 14
Step 8: Check the Arduino IDE ... 14
Step 9: Build and install the East/West (Tilt) switch and connector ... 15
Step 10: Attach the Stepper Motors ... 17
Step 11 Attaching the RGB LCD Display ... 18
Step 13 Run all test software again. ... 21
Step 14 Final Touches ... 21
5 Setup and Operation ... 21
5.1 Installing the Arduino Code ... 21
5.2 Basic Operation ... 21
5.2.1 User Menu ... 22
5.2.2 Guide Speed Menu ... 22
5.2.3 Tracking Rate Menu ... 22
5.2.4 Display Menu ... 22

5.2.5 Configure Mount Menu 23
5.2.6 GoTo Menu 23
5.2.7 Menu Summary 24
5.3 Status Messages 24
5.4 Configure the Controller 25
5.5 Changing the defaults 26
5.6 Updating the Software 27
6 Other Software 27
6.1 ASCOM 27
6.1.1 Celestron ASCOM driver 27
6.1.2 ASCOM Pad 27
6.2 PHD 2 28
6.3 Planetarium Software 28
6.3.1 Cartes du Ciel Star Charts 29
7 Optimizing the controller 31
7.1 Modifying Drive Rates 31
8 Optional Keypad 32
9 Abbreviations 34
10 Troubleshooting 35

Figures

Figure 2-1 Meridian Flip 4
Figure 2-2 Finder Scopes and Red Dot finder 5
Figure 3-1 Set Processor/Bootloader 6
Figure 3-2 Loading the Adafruit_RGBLCDShield Liberary. 6
Figure 4-2 Proper Placement of the Drilling Templates 9
Figure 4-1 Finished Enclosure 9
Figure 4-3 Install Grommets 9
Figure 4-4 Solder Zener Diode across jumper pins. 10
Figure 4-5 Cut Wires as shown. 10
Figure 4-6 Reset Circuit 11
Figure 4-7 Power Connector 11
Figure 4-8 Wiring Power Harness 11
Figure 4-9 Completed power harness 11
Figure 4-10 Control board power connections 12
Figure 4-11 Testing Voltage 12
Figure 4-12 Voltage with Nano installed 12
Figure 4-13 A4489 Motor Boards Installed 13
Figure 4-14 A4489 Motor Board Before Heat Sink is Installed 13
Figure 4-15 DRV8825 motor boards installed 13
Figure 4-16 DRV8825 with heat sink installed 13
Figure 4-17 Attach Stand Offs to Control Board 13
Figure 4-18 Bottom of Enclosure with screws and feet installed 14
Figure 4-19 Power Harness Installed 14
Figure 4-20 Tilt switch in housing on back of RA shaft. 15
Figure 4-21 Tilt switch attached to phono cable 15
Figure 4-22 Tilt switch jack with wires attached 15
Figure 4-23 Enclosure with tilt switch jack installed 16
Figure 4-24 Tilt Switch Housing with notches 16
Figure 4-25 Back of tilt switch in housing 16
Figure 4-26 Tilt Switch glued into cap 16
Figure 4-27 Stepper cables 17
Figure 4-28 DEC Stepper Driver Connections, RA motor is similar to the DEC motor connections shown 17
Figure 4-29 RGB LCD Display with button extensions added 19
Figure 4-30 RGB LCD Connections 19
Figure 4-31 Connect the LCD Display to the Motor Control 20
Figure 4-32 Display mounted in clear top. 20
Figure 5-1 Key Layout 21
Figure 5-2 Initial Startup Screen 22
Figure 6-1 Celestron Driver Control 28
Figure 6-2 ASCOM PAD Setup Windows 28
Figure 6-3 Location/Time error message 29
Figure 7-1 Drive RateChangeCalculator.xls display 31
Figure 8-1 Keypad Wiring Diagram 33
Figure 8-2 Remote Keypad outside and inside 33
Figure 8-3 Where to solder the Keypad wires on the back of the LCD Display 33

Tables

Table 1-1 Advantages of the Z-Field Dual Axis controller 1
Table 1-2 Operational Steps for a GoTo 2
Table 2-1 Parts List 3
Table 4-1 Required Tools 8
Table 5-1 Summary of the Menu System 24
Table 5-2 Status Messages 25
Table 5-3 Default Configuration Code 26

Preface

There once was a professor giving a lecture on Astrophysics. He wrote an equation on the board and as he wrote a second equation he said, "It's obvious that …". A student raised his hand and said, "I don't see that it is obvious at all!". The professor stood back and studied the board for several minutes, then he went to a nearby desk, pulled out a piece of paper and started writing furiously. After about 30 minutes he went back to the board and proclaimed, "Yes, it's obvious!" and continued his lecture. The reason for this story is to point out that somethings what may seem obvious to me may not be obvious to you. Every effort has been made to make these instructions as simple as possible. In a few cases I've also included notes on the reason behind certain steps. All efforts have been made to allow for the use of inexpensive, commercially available boards that only need to be modified slightly to work in our application.

But, if at any time you do not understand something, or do not understand how a step is to be completed, please feel free to email me for clarification. If you do not understand, there are probably others that do not as well and I will modify these instructions to make it easier to understand.

Also, if you find a mistake please let me know as well so it can be rectified.

Finally, if you have any suggestions on how to make the design or these instructions better, let me know as well. I have not included much on the Arduino software and how it works, other than the comments in the code itself. At some points it may be difficult to understand what or why certain code is done the way it is. If you have suggestions and or comments let me know.

My email is info@z-field.com.

1 Introduction

This manual describes the construction and operation of a controller unit that can be used with the Celestron CG-4 telescope mount if it has the Dual Axis Motor Kit. It is intended to replace the controller that originally came with the kit. The advantages of this controller over the original are:

1. In addition to the 2x, 4x and 8x speeds you get 0.5x guide, 16x and 50x. Although set to 0.5x, the guide rate can be changed by software such as PHD 2.
2. Turbo mode: By pushing a guide button (say DEC north) then pushing the opposite button (DEC south) at the same time, it goes to 16x speed while both are pushed. Useful for making short bursts to move the scope when you are in 2x or 4x mode.
3. The Display can be backlit in several colors including Red for night vision (shown in the picture).
4. Sidereal, Solar and Lunar Rate
5. Computer Interface
    - ASCOM compliant interface
    - Emulates the Celestron CGEM mount
    - Can be controlled by software such as Sky-X, Starry Night Pro and Cartes du Ciel SkyChart (freeware) (Stellarium will not work)
        - Although not a true GOTO mount you only need to aim at one bright star then the software will synchronize (Sync) the location of the mount to the software. You can GOTO step from there.
    - Can be controlled by ASCOM Pad (replaces the push buttons with on screen buttons)
    - Can interface with PHD 2 to allow for autoguiding
6. The Sync and GOTO functions can be accessed from the keypad so you don't need a computer
7. If you enter the time, date and location at startup it can give Sidereal Time.
8. Assumes an eyepiece or camera with about a 1-degree Field of View (FOV)

Table 1-1 Advantages of the Z-Field Dual Axis controller

This controller is an inexpensive way to achieve some of the functions of a GoTo mount, but, cannot do all the functions of a fully automated, GoTo system. The stepper motors used on the CG-4 mount were not meant to do high speed GoTo operations and are limited to a maximum speed of 50 times (50x) the sidereal rate. That means that it would take about 15 minutes to go from horizon to horizon.

Even if the speed of the motors were not a concern, the CG-4 mount does not have encoders on the axes so any GoTo has to be "open loop", that is, without anything to measure exactly where it is. Because of that limitation, as you go beyond GoTo distances of 5-10 degrees the errors in the system build up to the point that it will miss the intended object by more than a degree.

This controller has been designed to allow the user to "star-hop" from one location to another, correcting for the errors in the mount at each hop. Using this method, it is actually quite easy to go long distances across the sky to find dim objects you could not locate manually.

1. The method of operation for a GoTo is:
2. Locate a bright star and center it in the FOV of the telescope eyepiece/camera
3. Using telescope control software, or manual entry with the key pad, put the location of the star in the controller memory (Synchronize or Sync)
4. Using telescope control software, or manual entry, put the location of the object you want to GoTo in the controller
5. The telescope will move (slew) to the new location.

6. Using the telescope control software or the keypad, move the object to the center
7. Repeat Steps 4-6 until you reach the object of interest

Table 1-2 Operational Steps for a GoTo

Although this may sound involved, it is actually pretty simple and, depending on how many hops you need, does not take that long.

Because it can be computer controlled, software such as ASCOM Pad can be used to replace the push button hand controller for centering and guiding objects making it much easier to frame and take astronomical photographs.

This controller has been designed to look like a Celestron CGEM mount to any ASCOM compliant software. But, since it does not have the full functionality of a CGEM mount, there may be instances where your software may not operate correctly. For instance, a CGEM mount will automatically do a Meridian Flip (Explained in the next section). For safety reasons this controller will not do a GoTo operation across the Meridian. It is up to the user to make certain that the normal tracking movement does not drive the telescope into the mount (same as the Celestron controller).

2 Important First Steps

This section will explain important information as to how some of the parts that are used in the assembly work and why they are important. Below is a list of required parts. The first 20 are contained in the small parts kit available from www.z-field.com. The other items must be purchased separately including the motor controller, LCD display and Enclosure. The links to purchase these are given on www.z-field.com as well.

Small Parts Kit includes:

1. RJ11-4 cable with connectors

Bag 1:

1. (9) Interconnect wires (5- 8” long, 4- 4” long with connectors on each wire)
2. Red/black power wire (10”)
3. Coax cable with 3.5mm (1/8”) plug

Bag 2:

1. (4 sets) #4 screw & locknuts
2. (4 sets) Circuit board standoff + screws+ nuts
3. (5) Button Extensions
4. 3/8”ID Grommet
5. 1/4”ID Grommet
6. Small Zip Tie
7. Rubber bumpers for bottom of the enclosure
8. (4) Heat Shrink 4.5" long

Bag 3:

1. 1N5337BG Zener diode or equivalent
2. Red power LED with built in resistor
3. DPDT power Switch
4. 2.1mm Barrel power jack

5. 3.5mm (1/8”) mono audio jack
6. Tilt Switch
7. Tilt Switch Housing (White tube)
8. 100uf Electrolytic Capacitor

Not Included (links to the first three are on the www.z-field.com website):

1. Motor Control CNC Shield with Arduino and A4988 stepper controllers and mini USB programming cable
2. LCD RGB 16x2 + keypad +Buzzer Shield (Optional)
3. 4.5"x3.5"x2.2"(115mmx90mmx55mm) ABS Junction Box Universal Project Enclosure w PC Transparent Cover
4. 12 Volt power supply (at least 2.5 amps)
5. Optional remote
6. Finder-scope (Red Dot or others see Section 2.1)

Table 2-1 Parts List

The CNC Shield is the heart of the controller and includes the Arduino Nano that is the main processor of the unit. The LCD RGB display includes not only the 16 x 2 LCD display module, but also the keypad and buzzer used to communicate with the user. It is connected to the CNC Shield through an I2C 4 wire connection as well as a single wire for the buzzer connected to the D12 pin of the Arduino. These are all housed inside the Project Enclosure.

This unit is powered from a 12-volt power supply (not included) that is connected to the 2.1mm power connector. The 12v supply powers the stepper motors as well as the Arduino, LCD board and other circuitry. Since many users may want to power the unit from a 12v battery and/or car accessory port, the actual voltage from these sources can range up to 17v. However, the voltage regulator on the Arduino Nano has a maximum input voltage of 12v. The 1N5337BG diode is used to drop the input voltage by about 5 volts to allow for running the board at up to 17v. It should be noted that the stepper motors run at the full input voltage, but, they are rated for up to 24 volts. **Do not run the board at higher than 12v if this diode is not in place!**

The Arduino Nano has internal circuitry to force a reset whenever a serial port is connected or reset. Since this can interfere with some of the software you may want to use with this unit, there is external circuitry that keeps this reset from happening. You will notice that there is a 100uf capacitor on one side of the On/Off switch. This overrides the internal reset circuitry and is only enabled when the 12v power is applied. However, this reset is required in order to program the Arduino. To program the Arduino, power off the 12v enabling the internal reset, the Arduino will get power from the USB connector and can be programmed.

You will notice that three stepper motor drivers come with the CNC Shield. This project only uses 2. The third one may be used in a future upgrade so I would recommend keeping it installed on the board so you don’t lose it.

Just a quick explanation about the tilt switch: German Equatorial Mounts (GEM), like the CG-4, have a problem when an object is straight overhead. If you are tracking something in the east as it passes overhead the mount will make the telescope go past vertical and run the risk of driving the scope into the tripod legs. So, when the scope reaches vertical, you need to move the scope from the west side of the mount over to the east side of the mount where it can continue to track. This is called a Meridian Flip (see Figure 2-1). This flip effectively reverses the direction of the DEC axis so the controller needs to know which side of the mount the telescope is on to perform GOTO operations. In addition, if the time and location of the telescope are entered, the controller will not allow the telescope to slew during a GoTo operation across the Meridian to keep the telescope from being damaged. This happened to me during the first GoTo test and the scope was almost driven into the tripod. Fortunately, it stopped right before any damage occurred.

If you are trying to do a GOTO operation and attempt to go across the meridian the Controller will beep and will not move. (see Section 5.3 Status Messages)

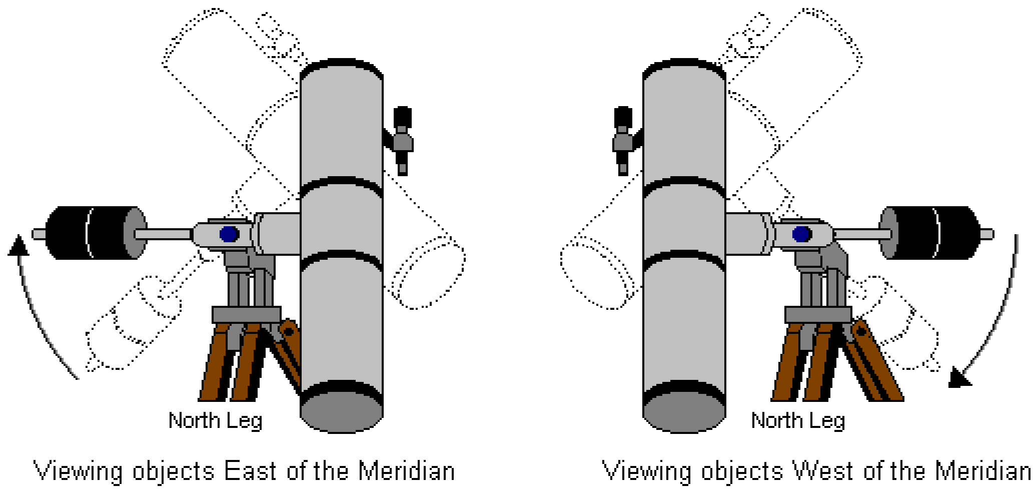


Figure 2-1 Meridian Flip

2.1 Optional Equipment

A quick note on what is not included in the kit. The user will need to supply 12V DC at 2.5 amps. Either an AC adapter (usually less than $10 on Amazon) or a 12V battery or Car Accessory Port. The included power jack on the controller takes a standard 2.1mm X 5.5mm barrel plug. The 6V battery pack that came with the CG4 Dual Axis Motor Kit will not work through this port because of the diode.

There are instructions later in this document that describe how to make an optional hand remote that is separate from the controller. If you use the controller standalone (without connecting to the computer) it may be worth constructing this, but, you may want to wait until you use it a while to decide. If the optional hand control is wired per the instructions, it may also be used as an ST-4 auto-guider port, however, since you can auto-guide through software on the computer and the USB interface, this port is probably not needed.

To initially align the telescope on the bright star/object to do a GoTo you may want to use some sort of finder-scope.

shows the setup I used to test the controller. It is pretty much all finder scopes. One method is to use a star diagonal and a wide-angle eyepiece in your telescope. If you use a 40mm eyepiece on an 80mm f/6 telescope this would give about a 4° field of view. Cost for this method ranges from about $50-$200 depending on which star diagonal and eyepiece you buy. On the right, in Figure 2-2, is a 70mm Celestron Travelscope. With the included 20mm eyepiece this gives about a 3° field of view at a cost of about $70. The advantage of this telescope is that it can also be used as a guide scope. However, you would need to build a cross bar (described in the next paragraph).

One disadvantage of the previously described finders is that 3-4 degrees FOV may not be enough to locate the star of interest. On the far left in Figure 2-2 is a Telrad Finder. This is what is known as a zero-power finder in that it simply projects a set of concentric rings onto a piece of glass you sight through. It makes a great finder for what is needed to align the initial star and cost is around $45. As a side note, if you are ever at McDonald Observatory in south Texas, you will see a Telrad finder on the side of the 107" telescope. At the top of the telescope in Figure 2-2 is my pill bottle solar finder. I simply drill a small hole in the middle of the lid and it projects an image of the sun on the back of the bottle. Centering that image will center the sun in the telescope. It works well and is free. But, it only works on the sun, and to a lesser extent, on the moon.

Finally, on the back of the telescope, and shown in the right image in Figure 2-2, is a red dot finder. These are

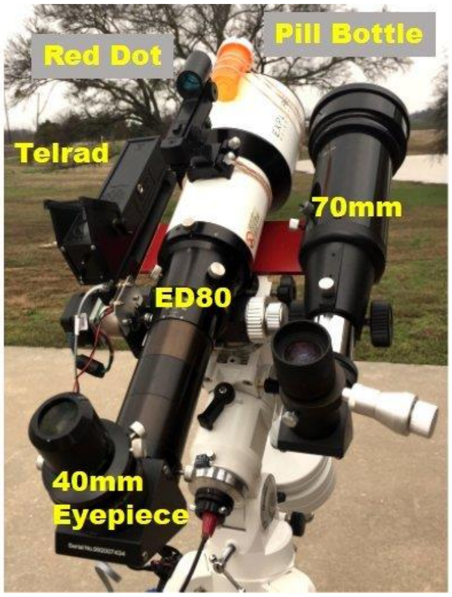


Figure 2-2 Finder Scopes and Red Dot finder

inexpensive, about $15-$20 plus $15 for a base, and work very well. You can see my high-tech method of mounting it on the telescope using several rubber bands. If you do not already have a finder I would consider one of these (this one is from SVBONY purchased on Amazon). It can easily be mounted on your telescope, and in my tests did an excellent job of lining the telescope/camera on the first star. One finder that I have left off the list is a green laser. Since I live right under the approach into Love Field in Dallas, and just south of the approach into DFW airport, I would never use a laser finder. If you do want to use the Telrad and/or Travelscope, you can make a cross-plate like the red one shown out of 3”x12” TEMCo 1/4 Inch 3"x12" 6061 Aluminum (Amazon). I drilled holes in the middle of the plate and mounted it to the top of the dovetail that holds the telescope to the mount and drilled holes in either end to attach the Telrad and Travelscope. The top of the dovetail on the 80mm scope I have is flat, however, the Daystar 80mm is not, so you may need to shim it to make it parallel with the tube.

2.2 Operational Notes

To perform GoTo operations either manually or through a software package you will first need to set the Latitude and Longitude as well as the Date and Time. This can be preformed by some software. I recommend using the software if it will do it or you can enter it manually using the built-in buttons.

For the GoTo operation to work you must polar align the mount first. Without proper polar alignment the GoTo operation will not work correctly. You can use the instructions given in the Citizen Cate manuals or use other methods such as http://astropixels.com/main/polaralignment.html . The freeware program PHD 2 can also be used to drift align the mount: https://openphdguiding.org/PHD 2_Drift_Alignment.pdf. I use a product called the QHYCCD PoleMaster (http://www.qhyccd.com/PoleMaster.html) that works great, but costs $300.

As with the manual operation of the CG-4 mount, if you allow the telescope to track past the meridian (telescope pointed straight up) you run the risk of running the telescope into the mount itself. It is up to the operator to make certain this does not happen. The controller does do a meridian check if you have gone to the object through a GoTo command and will beep and display the error message “Merid Xing”.

Error messages will only be displayed on the Controller Display as there is not a way to send messages to the PC.

3 Setting up the Arduino IDE

Install the Arduino IDE from the Arduino webpage at https://www.arduino.cc/en/Main/Software (click on the installation package for your computer type).

The first thing you want to do is set the Board and Processor. Click on the Tools tab on the upper toolbar. You’ll see a menu similar to Figure 3-1. Go to Board and go down the menu to Arduino Nano and select it. It should also then select the Processor: as ATmega328P (Old Bootloader). If not, go the processor tab in the Tools Menu and select it. (See Figure 3-1) Note that newer Arduino Nanos may have the ATmega328P.

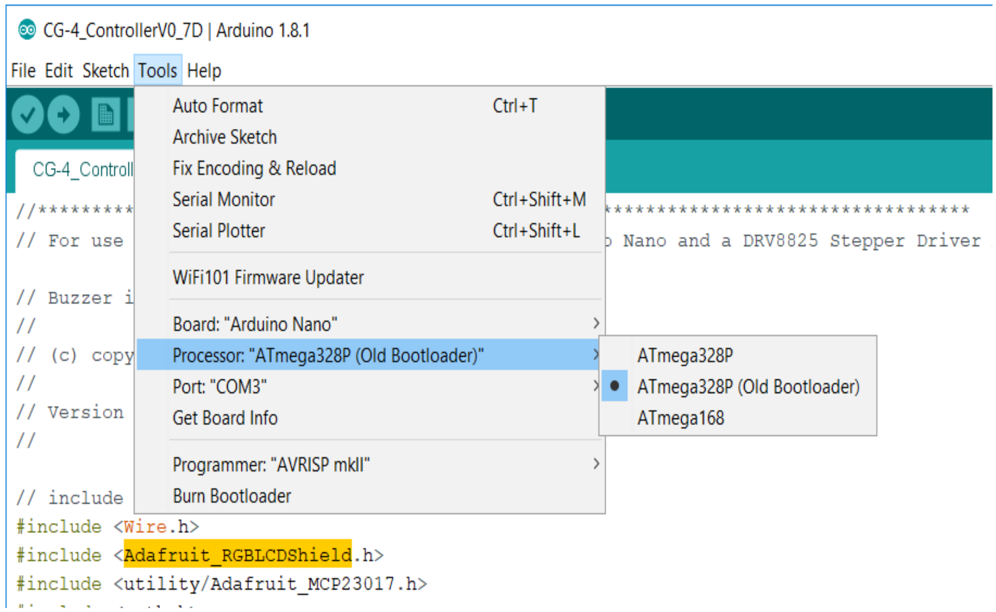


Figure 3-1 Set Processor/Bootloader

If the Arduino download does not work try setting the other ATmega328P setting and see if it works.

You will need to download the Adafruit-RGB-LCD-Shield-Library to use the RGB LCD. You can download it from https://github.com/adafruit/Adafruit-RGB-LCD-Shield-Library . Click on the Clone or Download button and select download ZIP file. After downloading open the Arduino IDE and click on Sketch > Include Library > Add .ZIP library then navigate to the file you just downloaded and select it. (Figure 3-2) It should now be loaded ready to use.

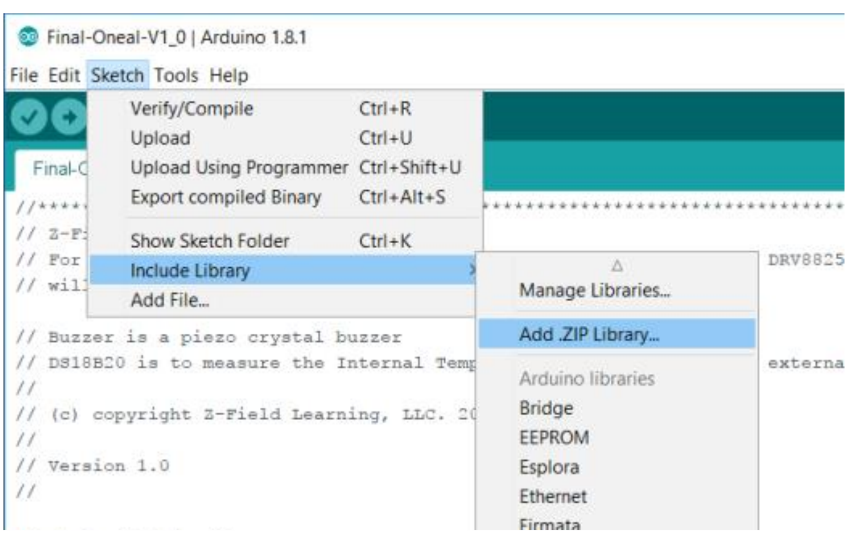


Figure 3-2 Loading the Adafruit_RGBLCDShield Liberary.

Do the same thing to load the eepromanything library from https://github.com/collin80/EEPROMAnything .

3.1 Downloading the Z-Field Dual Axis Controller Arduino Code

Go to www.z-field.com and click on the link for the Z-Field Dual Axis Controller. Click on the Downloads menu at the top of the page. This should take you to the Downloads section. Click on the Arduino Software to download the Z-Field Controller Software.zip file. Open the .zip file and extract it to a directory of your choice. It will create subdirectories:

- Z-Field Controller Software
    - Box-Template.pdf
- Arduino
    - Test Routines
        - Blink
            - Blink.ino
        - RGBLCD-Button-Buzzer_test
            - RGBLCD-Button-Buzzer_test.ino
        - StepperDriverTest
            - StepperDriverTest.ino
        - TiltSwitch
            - TiltSwitch.ino
    - Z-Field_ControllerVx_x where x_x is the current version number.
        - Z-Field_ControllerVx_x .ino

When instructed go to one of these .ino files and double click on them to load the Arduino IDE with that file. Click on the checkmark in the blue-green circle on the left. This will compile the code. If it completes there should be no error messages. If there are errors then you might need to load a library, but it should complete OK. Click on the green -> button to download to the Arduino.

If you want you can go to the Tools menu and select the Serial Monitor button and it will bring up a Serial Terminal. You'll need to have the Controller connected to a USB port. At the bottom right click the baud button and select 9600 baud. You're now ready to go.

4 Building the Z-Field Dual Axis Controller

Before construction of your Controller make sure you have all the parts listed in parts list in Table 2-1. In addition to these parts you will need the following tools:

1. Phillips head screw driver
2. Small flathead screw driver
3. Wire Cutters
4. Wire strippers (or carefully strip wires with the wire cutters)
5. Pliers
6. Soldering Iron. Any iron should work as all of the connections, are wire to wire or wire to pin.
7. Heat Gun for heat shrink tubing (a good, hot hairdryer may work)
8. Drill with Drill bits: 1/2", 5/16", 1/4", 5/32"
9. X-Acto knife or similar
10. Multimeter, with a continuity detector (Amazon, Walmart, electronics or auto parts store)
11. Hot melt glue gun and glue sticks.

Table 4-1 Required Tools

Step 1: Drill the holes in the plastic enclosure.

You will need to download the files from the z-field.com website. There should be a file called Box-Template.pdf. Print out this file. The first page contains templates for drilling the box. These are inside a 7" x 9.5" box. Make certain that this rectangle is exactly 7" x 9.5". If not, you will need to use the Custom Scale box under the PDF print menu to scale it to the right size. If it is not the right size the holes will be in the wrong place and the boards will not fit. You can print it on regular paper and use double sided tape to hold the template on the box, or print it on a full sheet label, but that is costlier. If using double sided tape, make certain each hole mark has a piece of tape under it to hold it to the box while drilling. The templates also list what size drill you should use for each hole. The clear top is more fragile than the lower bottom so I cover the entire top of the clear lid to the box with masking tape (either regular or the blue Painter's Tape) and tape the template on top to help keep the plastic from cracking while drilling. Since the corners of the box are curved I put a center line on the side and front templates. Simply put a mark half way on these sides to help line up the template to the box. Alignment of the top and bottom templates is critical, the front template is a bit less critical, the ½" hole to the right is the opening for the USB cable. If this ends up out of alignment you can just enlarge the hole carefully to allow the cable to fit. The alignment of the two sides is not critical. Each template is marked with which side it should be on. Figure 4-2 shows how to place the Templates. The preferred method is to drill the holes on a drill press, but you can use a hand drill if you are very careful. Drill very slowly, particularly on the clear top, to keep from cracking the plastic. When finished drilling make sure there are no burrs on the holes. I usually take a larger bit and holding just the bit in my hands, spin it in the holes to take off any rough edges. You might want to use masking tape on the drill bit to keep the edges from cutting your hand. On the ½" holes I carefully used a X-Acto knife to remove any burrs.

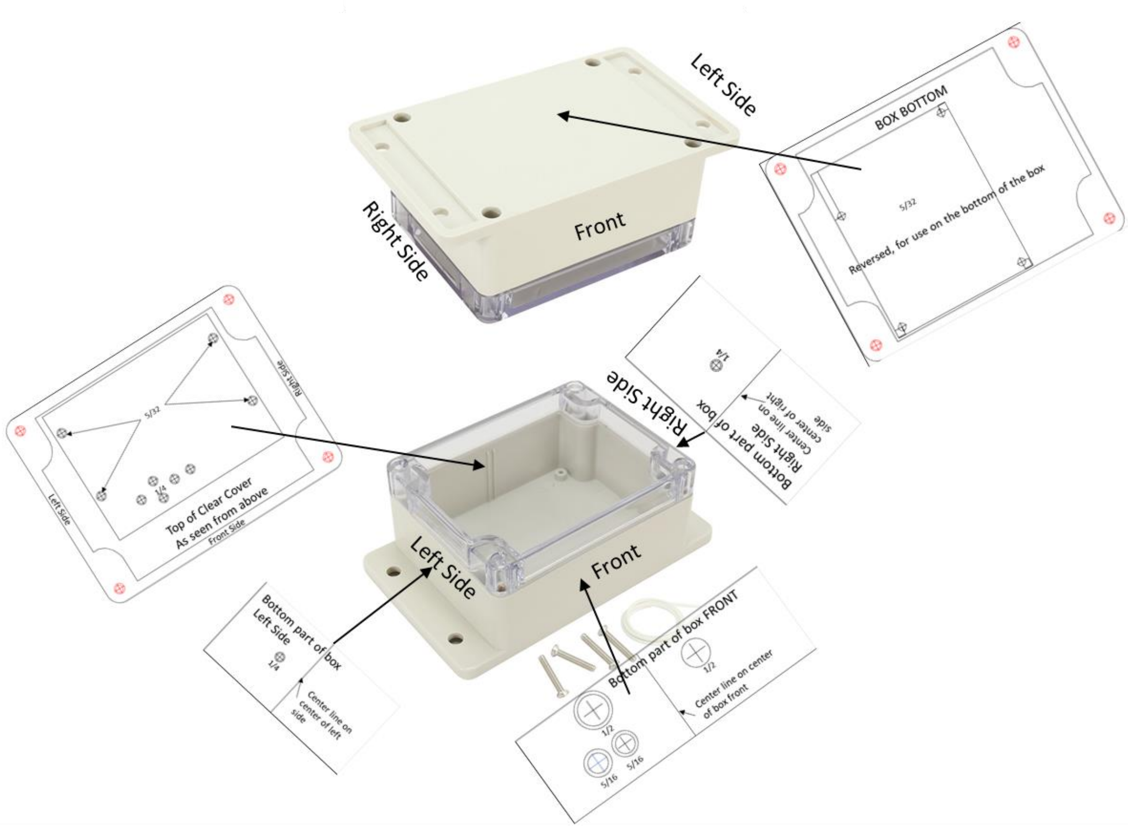


Figure 4-2 Proper Placement of the Drilling Templates

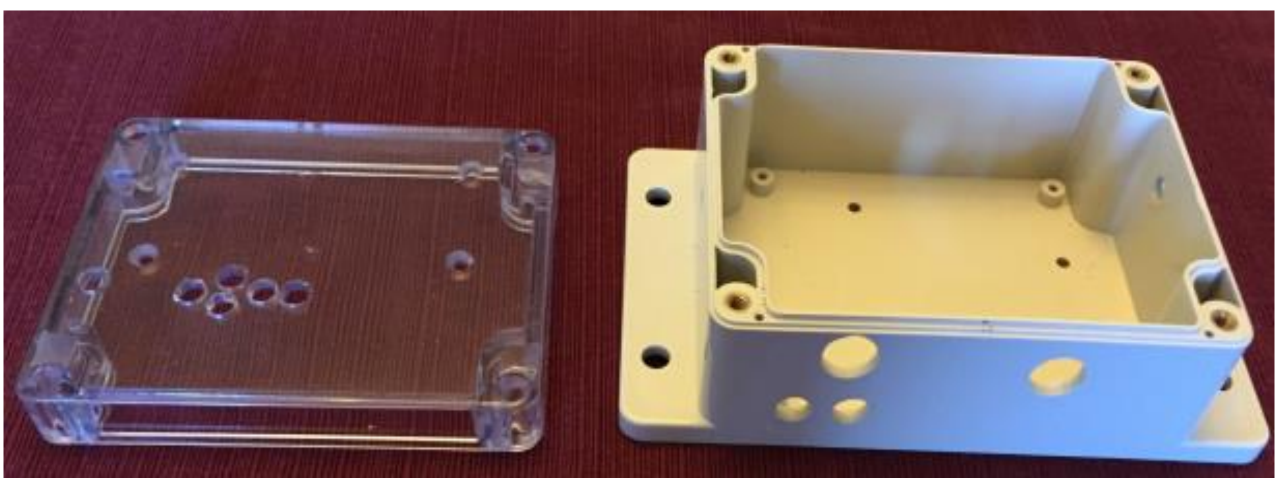


Figure 4-1 Finished Enclosure

Once drilled the enclosure should like Figure 4-1. I used a ¼" drill bit to carefully countersink the 4 holes in the top of the clear lid that will hold the LCD display so that the screws sit flush with the top.

Figure 4-3 Install Grommets

Now install the grommets into the two holes on the front of the lower part of the enclosure shown in Figure 4-3. The thickness of the enclosure makes the grommets somewhat difficult to install. I found it useful to warm the grommets in my hand first to make them a bit more flexible. Note that the power jack shown will be installed in a later step. The larger grommet is not that necessary so if you cannot get it installed you can leave it out.

Theory: You will notice that the CNC shield has a power connector. Do not use this connector! As mentioned in Section 2, the voltage into the board could exceed 12V so we need to put a Zener diode in line with the power input. This is done in Step 2. We will solder the 12V leads in Step 3.

Step 2: Solder the Diode to the Motor Controller Board

Locate the Zener Diode that came with your small parts kit. It should look like the figure to the left. Bend the lead coming out of the end with the band around it 180° so that it lines up with the lead coming out of the other end of the diode (as shown).   Cut the longer lead to make them even. At the top of the motor control board there are two pins labeled Mot_VOT_Sel which probably have a jumper across them. Remove the jumper, if it is there, and carefully solder the diode as shown in Figure 4-4. Be very careful not to get the diode too hot. Tin the leads (using the soldering iron add some solder to the two pins and the diode leads). Then attach the diodes to the pins. Make certain it is soldered exactly as shown with the lead from the band end attached to the pin farthest from the power connector!

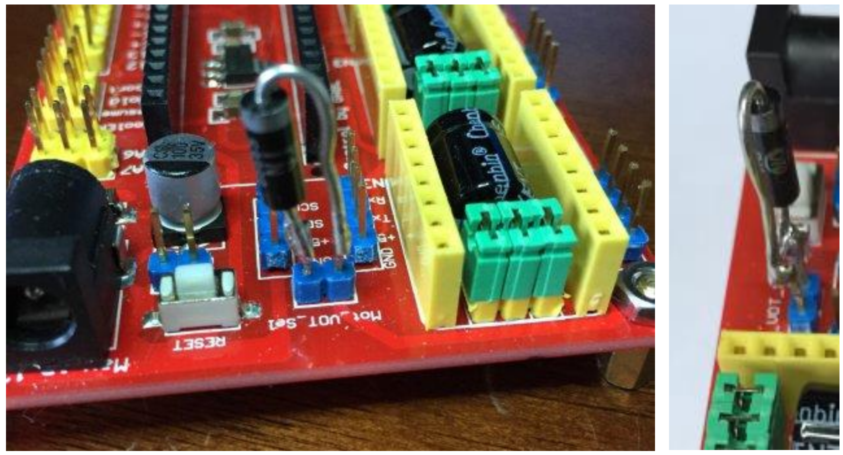


Figure 4-4 Solder Zener Diode across jumper pins.

Step 3: Prepare the wire

Take the wire out of the plastic bag. There should be a long (8”) set of 5 with connectors on each end, a short (4”) set of 4 with connectors on each end and a 10” length of red/black wire. Take the long, 8” five-wire set and cut it into two pieces, 3-1/2” long and 4-1/2” long. Cut the set with 4 conductors into two equal pieces 2” long. Finally, separate the black and red wires and cut the red wire in half so you have two 5” pieces. Leave the black wire uncut. See Figure 4-5. On the long set you either received the colors Brown, Red, Orange, Yellow, Green or Black, White, Gray, Violet, Blue. This will be important later on. On the shorter wires the color doesn’t matter.

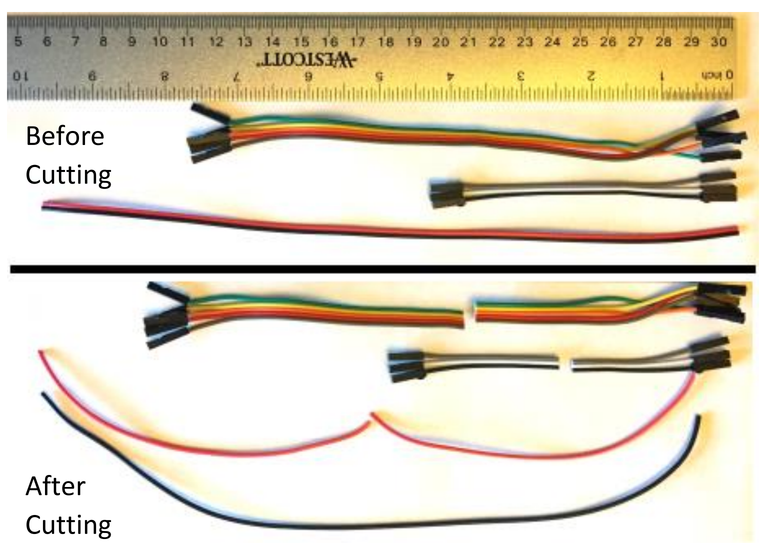


Figure 4-5 Cut Wires as shown.

Take two of the 4-1/2” long sections of heat shrink tubing and cut each into nine, ½” long pieces for use later. You can cut more later if needed.

Step 4: Wire the Power Harness

Wire the power wires as shown in the figure to the right (Note: Heat shrink not shown for clarity). Begin by soldering the negative lead of the 100uf capacitor to the outside pin of the switch. Take two of the 3-1/2” pieces of wire cut from the 8” set. Use the Green and Yellow or the Violet and Blue depending on which set you received. Slip a piece of the heat shrink onto the Green (Blue) and Yellow (Violet) wires, strip about ¼” of the insulation and solder the Green (Blue) wire to the Positive lead of the capacitor and solder the Yellow (Violet) wire to the center pin of the power switch (See Figure 4-8 and Figure 4-6). Slip the heat shrink down over the connections and heat with a heat gun to shrink.

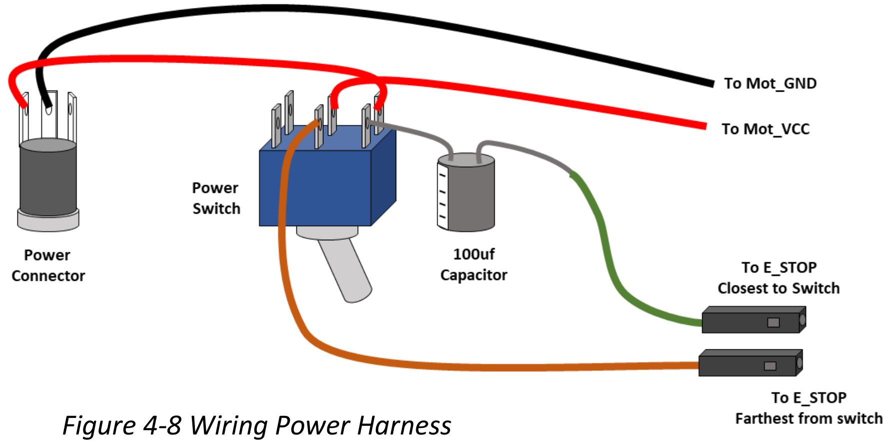


Figure 4-8 Wiring Power Harness

Take one of the 5” long pieces of Red Wire and slip on a piece of heat shrink and strip ¼” of insulation. Position the power connector so that one tab is to the right, one opposite you and one to the left (see Figure 4-8 and Figure 4-7). Solder the red wire to the tab on your left as shown in the figures. Repeat with the black wire soldering it to the center tab as shown in the figures. Slip the heat shrink over the soldered connections and shrink with a heat gun.

Figure 4-6 Reset Circuit

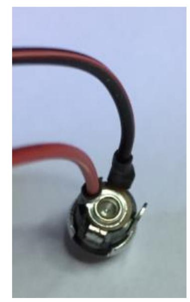


Figure 4-7 Power Connector

Take the other end of the red wire you just soldered, strip the insulation and slip a piece of heat shrink onto it, and solder it to the switch on the tab opposite the capacitor as shown in Figure 4-8. Take the remaining 5” piece of red wire, slip on a heat shrink, strip ¼” of insulation and solder it to the center terminal opposite the Yellow (Violet) wire as shown in the figures. Slip the heat shrink down over the tabs and shrink with a heat gun. The assembly should look like Figure 4-9.

At this point it would be a good idea to strip the black and red wires about 1/4”, connect a power supply to the power connector and check the voltage across these two wires. It should read 12V (or whatever power supply you are using) and should go on and off when you switch the switch.

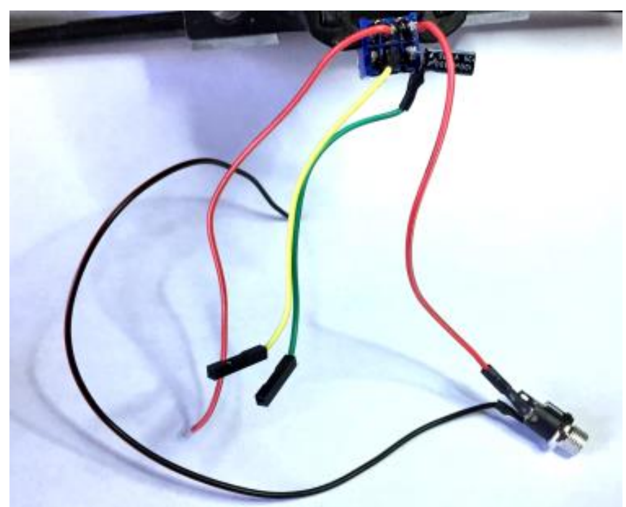


Figure 4-9 Completed power harness

Step 5: Solder Power Harness to the Motor Control Board

Take out the Motor Control CNC Shield. The Arduino Nano and the Motor Control circuits **should not** be installed. Solder the red wire from the switch to the Mot_VCC connection (Near the X) and solder the black wire to the Mot_GND connection (Near the Y). (See Figure 4-10) Take the LED provided in the kit and solder the red wire to the remaining hole in the Mot_VCC and solder the black lead to the remaining hole in the Mot_GND as shown in Figure 4-10.

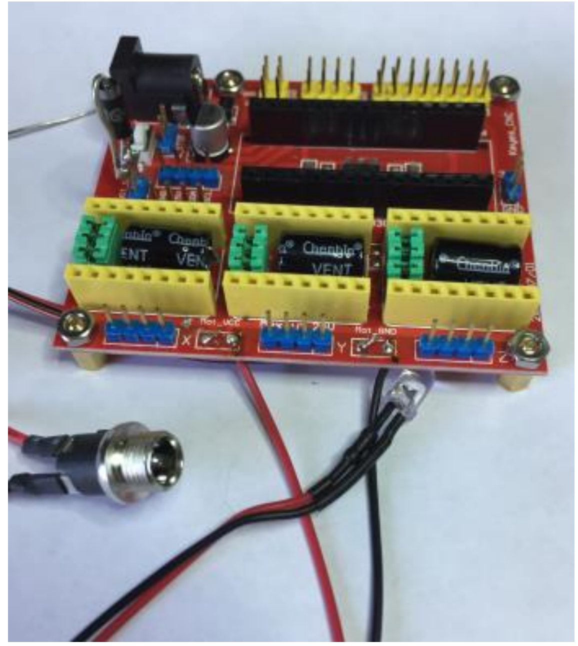


Figure 4-10 Control board power connections

Make certain that there are no boards plugged into the Motor Control Board, plug a 12V power supply into power jack and switch on the power. The LED should light. If it does not light, disconnect power and try reversing the LED leads and see if that works.

Using a Multi-meter set to DC voltage, place the Ground/Common lead on the Mot_GND and place the Voltage lead on one of the leads from the diode installed in Step 1. On the side closest to the Mot_VCC lead it should read 12 volts (or whatever your power supply is). On the other lead it should read about 9 volts or so, see Figure 4-10. If both sides are 12V there is a problem with how you soldered on the diode. Make certain it is installed like Figure 4-4 and measure again. If it still measures about the same voltage on each leg the diode may be bad. Make sure it is Positive 9 volts and not Negative 9 volts. If it is negative your power leads may be reversed.

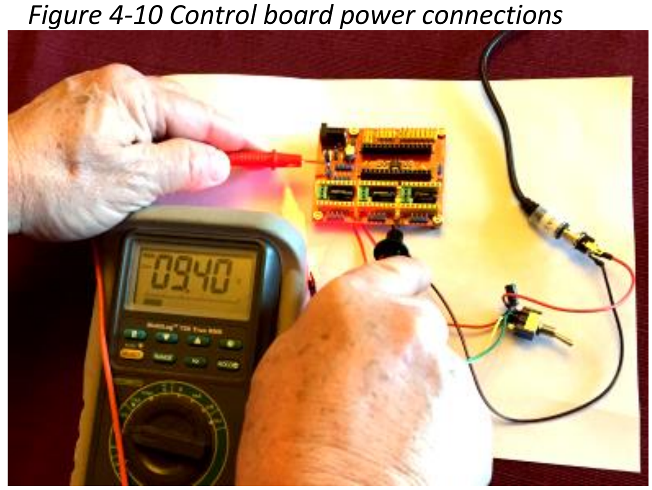


Figure 4-11 Testing Voltage

If the voltage measures less than 12 volts go ahead and install the Arduino Nano onto the Control Board. Make sure it is installed as shown in Figure 4-12. You don't need to install the motor control boards at this time. Measure the voltage as you did previously. Now, one voltage should read 12 volts and the other around 7 volts. Why is the voltage different than before? Without something attached to the diode the voltage "floats" up to about 9.4 volts. With the Nano drawing some of the current the voltage is now 12 volts minus the 5 volts of the Zener Diode to give around 7 volts. Note that both the LED you installed and the LED on the Nano should be on.

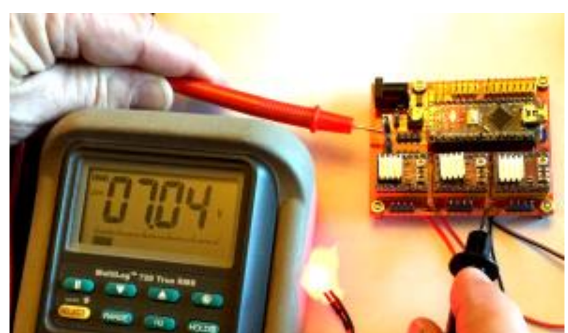


Figure 4-12 Voltage with Nano installed

Before going to the next step make sure the three jumpers between the yellow sockets on each of the motor boards are installed as shown in Figure 4-10. These control the stepping motion of the stepper motor and all three sets of three should be installed in each of the motor control board slots.

Step 6: Install the motor boards

Your Motor Control CNC shield came with three motor boards each with a small heat sink. Carefully peel the paper cover on the bottom of the heatsink to reveal the adhesive beneath then press the heatsink onto the chip on the board, see Figure 4-16. Do this for all three motor boards.

Figure 4-16 DRV8825 with heat sink installed

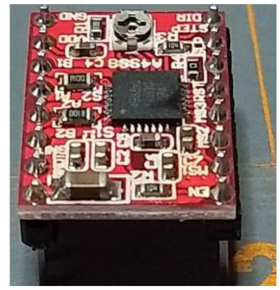


Figure 4-14 A4489 Motor Board Before Heat Sink is Installed

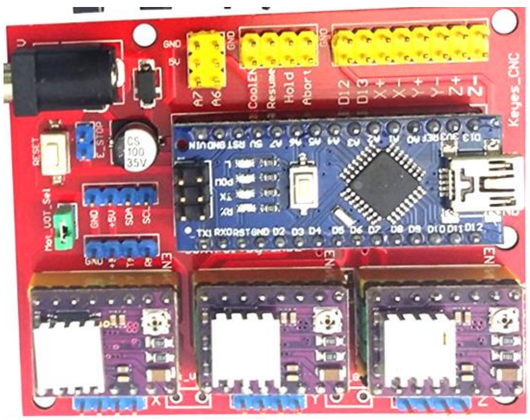


Figure 4-15 DRV8825 motor boards installed

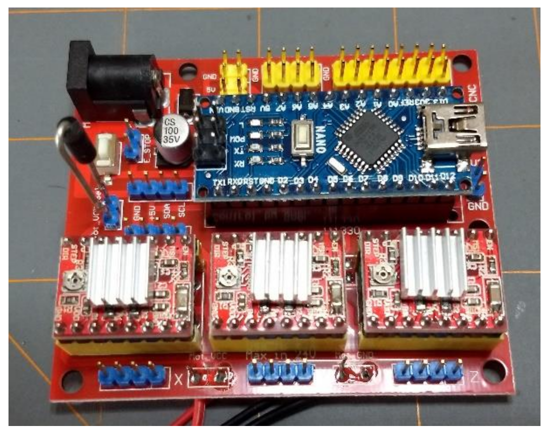


Figure 4-13 A4489 Motor Boards Installed

The CNC shield comes with one of two motor boards, the DRV8825 shown in Figure 4-16 with a purple color board or the A4489 shown in Figure 4-14 with a red color board. Take a minute to be sure you know which board your controller came with. Then Install the motor boards into the sockets on the Motor Control CNC Shield as shown in Figure 4-15 if you have the purple DRV8825 boards or in Figure 4-13 if you have the red A4489 boards. Note the direction. Make certain your board looks like the figure appropriate to your boards before proceeding. Note that for the DRV8825 purple board the small adjustment screw is to the right while on the A4489 red board they are to the left. The pins labeled A1, A2, B1 and B2 should be closest to the blue header pins that attach to the stepper motors and farthest from the Arduino. Note that on the DRV8825 the labels are on the bottom of the board.

Figure 4-17 Attach Stand Offs to Control Board

Take the 4 circuit board standoffs, 4 M3 screws and 4 M3 nuts out of the bag. Push the screw part of the standoff through the holes in the control board and attach it with the M3 nuts as shown in Figure 4-17. Do the same with the remaining standoffs. With the standoffs attached check the fit of the control board in the enclosure. Looking from the bottom of the enclosure you should be able to align the standoffs with the mounting holes. If not, you may need to carefully enlarge some of the holes to where they match. Put the M3 screws through the bottom of the enclosure and engage with the standoffs on the Control Board.

There are 4 small rubber bumpers in your kit. Remove each one from the backing and place near the 4 corners of the bottom of the enclosure. This will keep the metal screw heads from scratching any surface it is placed on. Your enclosure should now look like Figure 4-18. If you look through the larger hole to the right of the smaller grommet you should see the USB connector nearly centered. Try plugging in a USB cable to ensure it fits. If not, you may need to remove the board and enlarge the hole until it does.

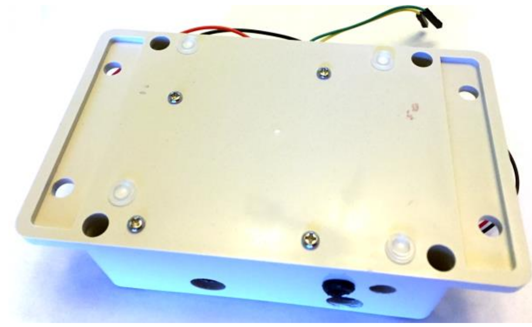


Figure 4-18 Bottom of Enclosure with screws and feet installed

Step 7: Install the power harness

The power harness should be installed as shown in Figure 4-19. Push the LED through the hole in the small grommet. Push the top of the power jack through the hole to the left of the LED and secure with the included nut. Push the switch through the ¼" hole in the left side of the enclosure and attach with included washers and nut. Do not attach the wires from the switch to the E-Stop pins just yet.

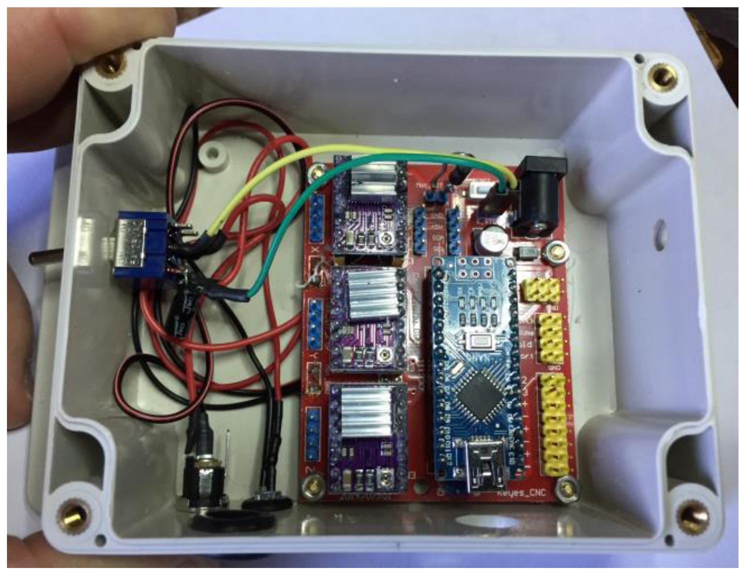


Figure 4-19 Power Harness Installed

Step 8: Check the Arduino IDE

Install the Arduino Nano board onto the Controller board a shown in Figure 4-19. Connect a USB cable from your computer to the Arduino Nano on the controller board. Go to the directory you installed the Test Routines software to. Go to the Blink directory and double click on Blink.ino. This will bring up the Arduino IDE and the Blink software. Upload the code to the Arduino Nano and, after it uploads, it should blink the LED on top of the Nano once per second. If you are having trouble uploading check the Processor Section of the Tools Menu and switch between ATmega 328P and ATmega 328P (Old Bootloader) to see if it will upload. Also, double check the COM port. You may need to bring up the Device Manager to check to see what port it is on. Under the Ports (COM & LTP) section you should see USB_SERIAL CH340 (COM#) where # is the COM port the Nano is attached to. At this point the Nano is not really connected to anything but the USB cable so it should upload normally and blink the LED. If you are having trouble check the Arduino websites for help on getting it running.

Once you can upload to the Nano attach the Green (Blue) and Yellow (Violet) wires to the E_Stop pins as shown in Figure 4-8 and Figure 4-19. Make certain the wire from the left most of the E_Stop pins goes to the positive side of the Capacitor. The wire from the other pin should go to the switch. Plug in a 12-volt power supply and make sure the power LED as well as the LED on the Nano light up when power is supplied.

Now try to upload the Blink.ino file again. If it does not upload check to make sure the power switch is off. The Nano will not be able to load code from the computer while the 12V power switch is on. If it did upload turn on the 12V power and re-upload the Blink routine. You should get an error that it cannot upload. If it can upload with the switch in both positions, recheck the connections to the E_Stop connector and the capacitor on the power switch. Note that when the power switch is on, the Nano cannot upload even if the 12V power is not connected. If the switch is on the Arduino IDE will attempt to upload for several minutes then give an error. This is similar to the (Old Bootloader) problem so if you are having problems uploading try both fixes. Usually, even after it is fixed you might get an Orange warning: **An error occurred while uploading the sketch message**. So, try uploading a couple of times to see if it works.

Figure 4-20 Tilt switch in housing on back of RA shaft.

Step 9: Build and install the East/West (Tilt) switch and connector

Carefully strip about 1 inch of the outer plastic jacket from the audio cable with the 3.5mm connector on the end. I carefully scored the jacket with an X-Acto knife, but don't cut too deep as you'll cut the wires. You should see a white inner wire surrounded by a copper braid. Carefully pull the braid from around the inner wire and twist it to form a lead (Figure 4-21). Strip about ¼" of insulation from the white wire and slip a piece of heatshrink over each lead. Take the tilt switch (shown in Figure 4-21) and bend the leads down as shown. Using the soldering iron put some solder onto each lead as well as onto each wire on the phono cable. It doesn't matter which lead is soldered to which wire, you will calibrate it later. Then solder the wires to the tilt switch as shown. Move the heat shrink over the leads and shrink it using a heat gun.

Figure 4-21 Tilt switch attached to phono cable

Take two of the remaining 3-1/2" wires cut from the long leads with the connectors on the end and strip ¼" of insulation from the ends. Slip a piece of heatshrink onto each lead and solder them to the 3.5mm (1/8") audio jack from the kit as shown in Figure 4-22. Move the heatshrink over the connector tabs and shrink.

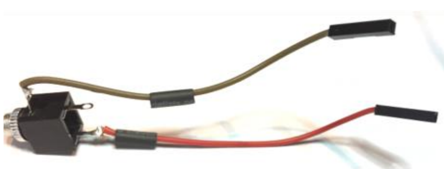


Figure 4-22 Tilt switch jack with wires attached

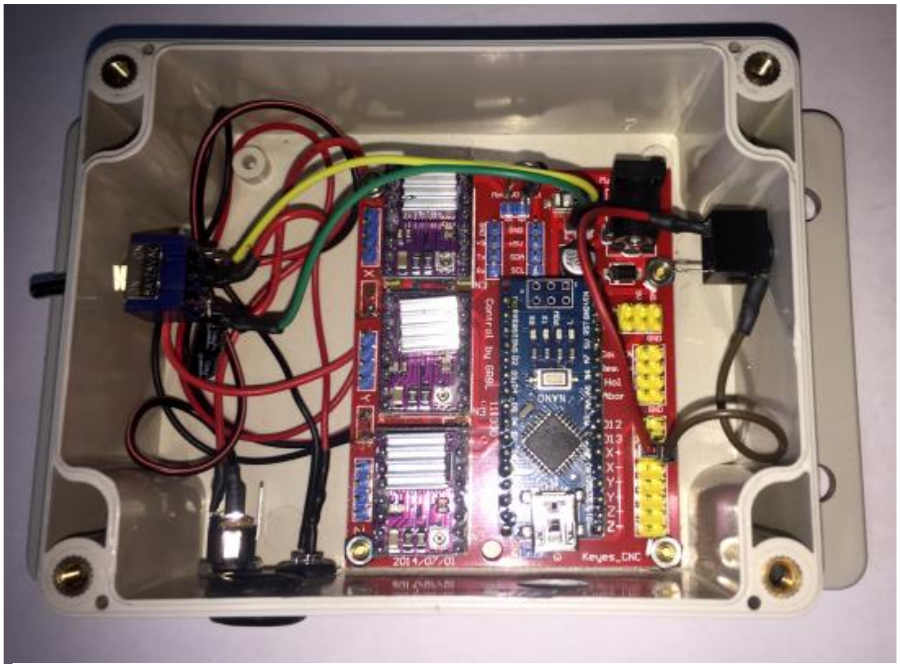


Figure 4-23 Enclosure with tilt switch jack installed

Install the jack through the remaining hole in the right side of enclosure bottom. Your controller should now look like Figure 4-23. As in the figure connect the leads from the jack to the X+ pins as shown. The lead from the side of the jack should go to the outside pin and the lead from the back of the jack should go to the pin closest to the X+ symbol.

Go to the Test Routines directory then to TiltSwitch and double click on TiltSwitch.ino to bring it up in the Arduino IDE. Connect the phono plug to the phono jack you just installed and upload the TiltSwitch routine. Go to the Tools menu and select Serial Monitor. You should see the words East and/or West being printed every second. Rotate the tilt switch and you should see it switch back and forth between East and West. If not, check your wiring to be sure everything is connected per the instructions. Be certain the tilt switch works before proceeding.

Figure 4-24 Tilt Switch Housing with notches

Figure 4-26 Tilt Switch glued into cap

Find the 1” long, white tube. This is the tilt switch housing. It fits in the opening at the rear of the Right Ascension shaft of the CG4 Mount (Figure 4-20). You will need to cut two notches in the top with an Xacto knife about ¼” deep like are shown in Figure 4-24. Make certain the tilt switch with bent leads, see Figure 4-21, will fit into these notches.

Once the switch is working run the wire with the 3.5mm plug on the end through the housing so that the tilt switch nests in the notches you cut (Figure 4-26)

Figure 4-25 Back of tilt switch in housing

Make sure the tilt switch is parallel with the end of the cap. It doesn’t have to be perfect, just close. Then hotmelt glue the switch into the notch in the housing (Figure 4-26). You will align this later. Turn the housing around and hot melt glue the cable end so that it doesn’t pull out, Figure 4-25

The tilt switch is important so that the controller will know which way the telescope is pointing, and therefore, which side of the mount it is on. Put the tilt switch into the Right Ascension axis using the set-screws on the back of the RA axis as shown in Figure 4-20. The tilt switch should be horizontal. You can align it now using the TiltSwitch.ino (or later using the final code). With the telescope on top and parallel with the RA axis, as in Figure 4-20, twist the tilt switch housing around until the serial display reads West when twisted slightly clockwise and East when twisted slightly counter clockwise. Set it about in the middle of these and it should be aligned good enough. Unlock the RA axis and move the scope to point to the west and the display should read West, then move it to point east and make certain the display says East. If it reads the reverse twist the housing 180 degrees and align per the previous paragraph. After you have finished aligning the tilt switch you should use a Sharpie to make a mark on the housing and another mark on the RA axis so you know where to align it if you need to remove it. The tilt switch can be unplugged from the Controller so it can stay attached to the mount.

Step 10: Attach the Stepper Motors

Find the 8 short (2") wires with connectors on the end you cut in the third step. Take the long, white telephone cable with the phone type plugs on the ends and cut it in half. Mark one of the cables on each end with a piece of tape or some other method of identifying it. This will be the RA cable. Slip the cables through the large hole in the front of the enclosure (see Figure 4-27). Carefully strip off about an inch of the outer jacket of each cable being careful not to cut or damage the wires inside. I find that it is better to use an X-Acto knife and starting 1 inch from the end cut to the end of the jacket slicing parallel with the wire length and, hopefully, between the wires inside. Then, fold the jacket back, pull out the 4 wires and snip off the excess jacket with scissors or wire cutters. You should now see black, red, green and yellow wires.

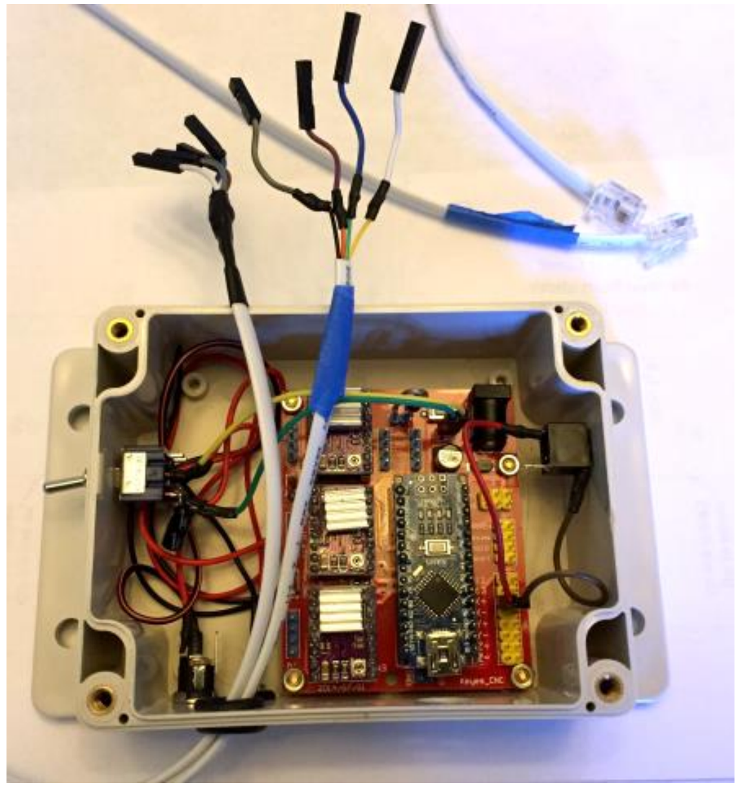


Figure 4-27 Stepper cables

Strip about ¼" off of each of these wires as well as ¼" from the wires with the connectors. Slip a piece of shrinktube onto each wire and solder them to the wires from the phone cable as shown. Color doesn't really matter. Slip the heatshrink down over the connection and seal with a heat gun. Do the same for the remaining cable. Make sure you can see the phone cable wire colors as you will need to attach these by color to the connectors in the next step.

Connect the connectors to the Motor Control Board as shown in Figure 4-28. The wires from the DEC motor need to go in the Y connector block and the wires from the RA motor need to connect to the X block. The placement of the wires is important! I find it is easier to start

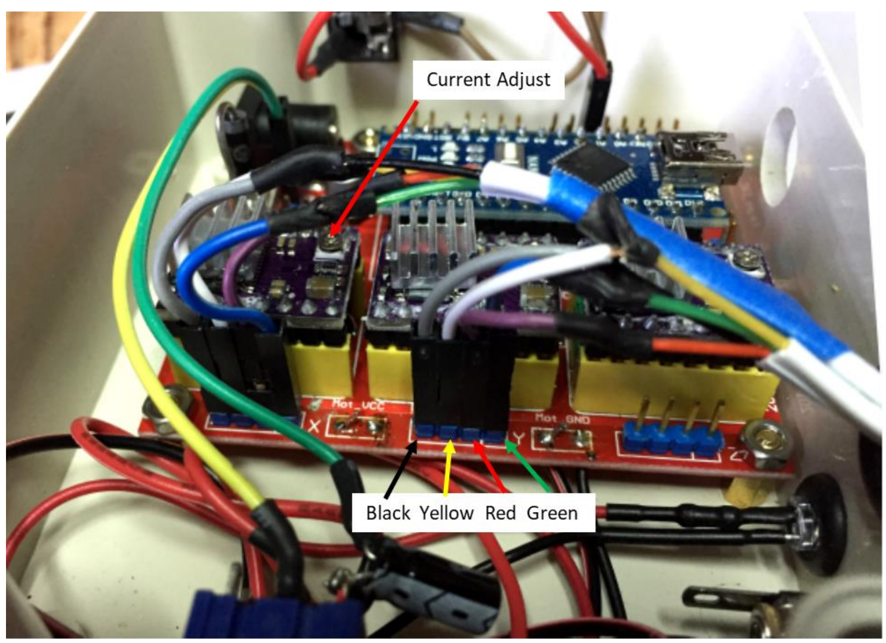


Figure 4-28 DEC Stepper Driver Connections, RA motor is similar to the DEC motor connections shown

with the DEC cable. The colors refer to the colors of the phone cable wires, not the wires with the connector attached! Plug the Green wire onto the pin closest to the Y label. Then plug the Red wire next to it followed by the Yellow and finally the Black wire farthest from the Y label. Do the same for the RA wire starting with Green next to the X label, then Red, Yellow and finally Black farthest from the X label. Your controller is now ready to test.

Plug the RA cable into the RA motor on your telescope mount. Plug the DEC cable into the DEC motor. Plug a 12V power supply into the DC power jack, make sure the switch is turned OFF. Attach a USB cable from the Nano to your computer. In the Test Routines directory go to the subdirectory for the StepperDriverTest and double click on the StepperDriverTest.ino file to load it into the Arduino IDE. With the power switch OFF upload the code to the Arduino. Go to the Tools menu and open the Serial Monitor. Now, turn on the 12V power supply. The motors should now start moving. The Serial Monitor should display what motor is moving and which direction. Make sure both motors move in each direction.

If the motors are not moving immediately turn off the 12V power and check your connections. Check to make certain each of the motor boards is correctly seated in its socket and is facing the correct way. If one or both still do not move you can check your connections with a multi-meter in the continuity check position. Remove the stepper wires for the motor not moving from the Motor Controller board. On one side of each connector attached to the phone cable is small hole you can see the metal connector through. Press the multi-meter probes into the small holes and make sure they are contacting the conductor. Check to see if there is a connection between the Black and Yellow wires. Also check to see if there is a connection between the Green and Red wires. If not, redo the connections between these connectors. If you do have to redo the connections make sure there is continuity between the Red and Green bare wires and the Black and Yellow bare wires. If not make certain that the connector is firmly seated in the motor. If none of this works check between all the wires and see if there are two sets that are connected. If so you can connect these to the Controller Board.

Finally, if the connections appear to be OK, but the motor is still not turning but are buzzing you should take a small screwdriver and adjust the small current adjust screw (Figure 4-28) on the top of the associated motor board. Turn it clockwise until the motor begins to move the turn it about ½ turn further clockwise.

If this doesn’t work and one motor works and the other doesn’t, then you may have a bad motor board. Try swapping out the motor board with the unused one in the Z slot.

Also, plug the motors back into the original CG4 controller and make certain they still work.

If none of these help, contact me and I’ll see if we can get it to work.

With the StepperDriverTest.ino running take a small screwdriver and on each of the small motor control boards there is a small screw adjust. This adjusts the current for the stepper motor. Turn this screw counter-clockwise until that motor stops (it will probably be buzzing but not moving). Now, turn the screw clockwise until the motor begins to move then turn it about ½ turn further clockwise. Do this for both the RA and Dec driver boards.

Later, once everything is working, you might want to go back and snip off each of the single connectors and solder the phone/stepper cable wires directly to the pins. This will provide a much more robust connection, but, makes it difficult to make any future changes.

Step 11 Attaching the RGB LCD Display

Take the RGB LCD Display out of its package and place it face up. The first thing is to cover up the obnoxious, green LED on the front. Figure 4-29 shows the display with the button extensions we’ll add in the next step. Note the red blob next to the IC just right of center. This is the Green power LED. I covered mine with red nail polish, but anything that will cover it up will work, paint, small piece of electrical tape, whatever you have available. The problem is that it will mess with your night vision so it is better to eliminate it now while it is accessible.

You should have the 4-1/2" set of 5 wires with connectors on the end remaining. It should be either Brown, Red, Orange, Yellow or Black, White, Gray, Violet and Blue. I used the first one in my build so I'll be using those colors with the other color set in parenthesis.

You'll notice that I have clipped off all the pins on the lower part of the Display Board with wire cutters. The board was built to fit onto a Arduino Uno, but these pins may contact wires below them in the enclosure so I cut them off. Figure 4-30 shows how to make the connections. This is the upper and lower sides of the LCD Display. Strip about ¼" of insulation from the wires with the connectors and solder them to these holes, use the figure as a reference.

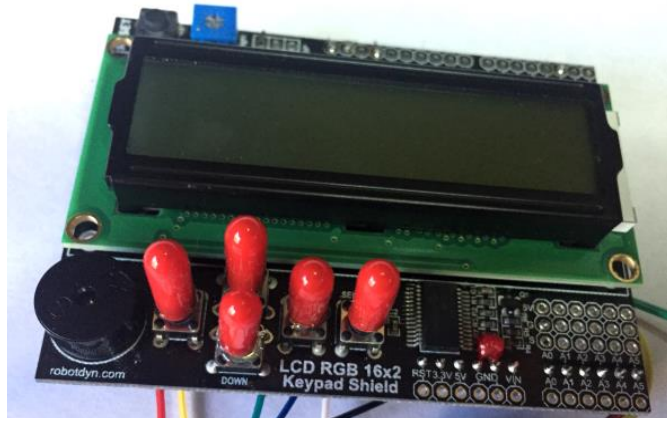


Figure 4-29 RGB LCD Display with button extensions

Color	**Hole**
Brown (Black)	G
Red (White)	ICSP Pin 2
Orange (Gray)	SDA
Yellow (Violet)	SCL
Green (Blue)	2

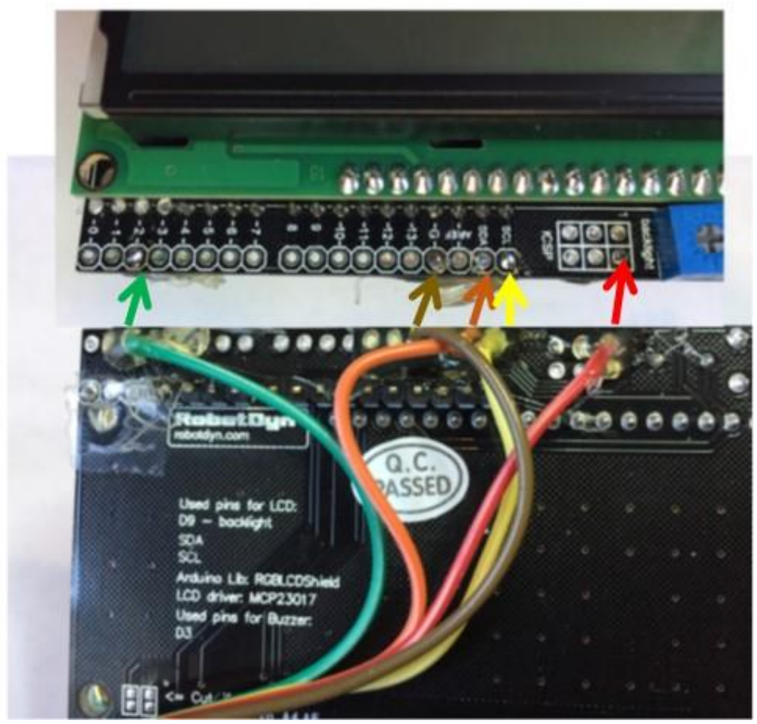


Figure 4-30 RGB LCD Connections

Now temporarily connect the LCD Display to the Motor control board. Figure 4-31 shows how to connect the display. Make the following connections:

Color	**Pin**
Brown (Black)	Gnd
Red (White)	+5V
Orange (Gray)	SDA
Yellow (Violet)	SCL
Green (Blue)	D12

Note that D12 is down next to the Nano board. Make sure you use the pin closest to the Nano.

Once you have made the connections, connect the USB cable to the Nano and go to the Test Routines Directory and then to the RGBLCD-Button-Buzzer_Test subdirectory. Click on the RGBLCD-Button-Buzzer_Test.ino file to load it into the IDE. Make sure the 12V switch is OFF and upload the code to the Nano. You do not need to turn on the 12V at all for this test. After uploading you should see Hello World on the first line of the display and numbers counting up on the second line of the display. Press each of the 5 buttons. The background color of the LCD should change and it should read which button was pressed and the background color in the first line accompanied by a different frequency beep. If you do not see anything recheck your connections. Make certain the Gnd is going to the G hole, the +5V goes to the second hole in the ICSP, SDA is going to SDA and SCL is going to SCL. If you do not hear the buzzer beep make certain the Green(Blue) wire is on the D12 pin closest to the Nano and not the D13 pin.

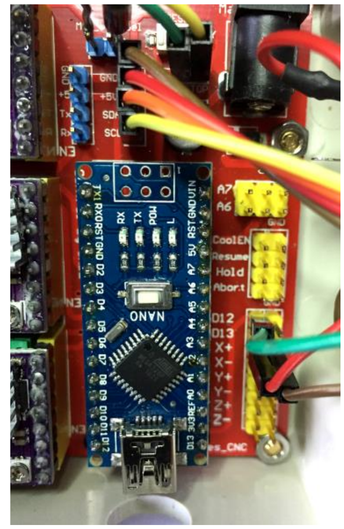


Figure 4-31 Connect the LCD Display to the Motor Control

If everything works then it is a good idea to put some hotmelt glue on the wire connections on the back of LCD Display. This will make it far more difficult to accidently pull out the wires if/when you need to remove the display from the enclosure.

Now, remove the connections and set the board face up in front of you. Find the 5 red button tops (See Figure 4-29). Don't glue them on the buttons yet. They are not stiff enough to allow proper use of the buttons as is. You need to find something to put inside to stiffen them up. I used toothpicks, but thick weed-eater string will work as will 3mm 3D printer string. Once you find something to use cut them into 7mm lengths. I had to use 2 toothpicks per extender. Put a drop of hotmelt glue on one end of the stiffener material and push it to the bottom of the red button extender. Then, place another drop of hotmelt glue on top of one of the buttons on the LCD display board. Pushing down, put the extender onto the button, make certain it is seated. It should look like the button extenders in Figure 4-29. Repeat for the remaining 4 buttons.

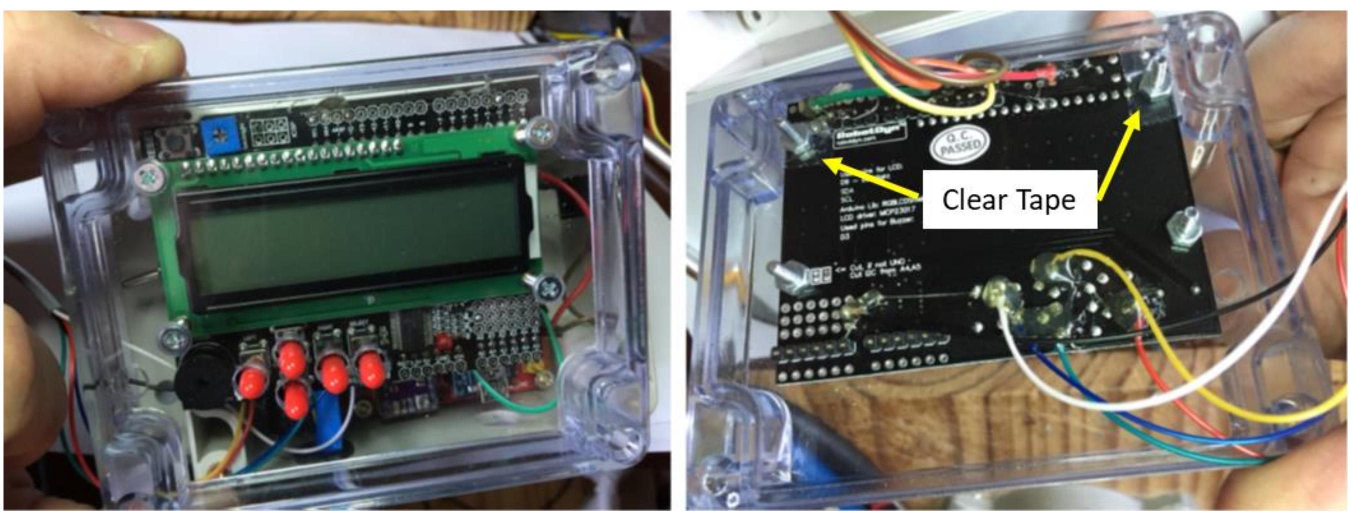


Figure 4-32 Display mounted in clear top.

Now it is time to mount the Display into the clear top of the enclosure. Before doing that, I was concerned that the screws may make contact with some of the exposed solder joints on the Display board. Take a couple of pieces of regular or electrical tape and put them over the holes at the top of the screw holes at the top of the LCD display. Take and X-acto knife and cut the tape in front of the hole to allow the screw to go through.

Take the LCD Display and carefully guide the button extensions through the holes in the clear enclosure top. See Figure 4-32. The four screw holes in the display and display board should match up with the holes in the clear top. Take the four 4-40 1" long screws and push them through the clear top then through the holes on the display board and put a locknut on each one. Very carefully tighten the screws. Too tight and the clear top may crack, so tighten them just enough to hold the display snug against the clear top. The nuts are locknuts so they should not come loose under normal conditions. Ignore the 6 wires attached to the bottom, these are for the optional keypad described later.

Step 13 Run all test software again.

Reconnect the display to the Motor Control Board as described earlier (Figure 4-31) and set the clear enclosure top onto the enclosure bottom and make certain it fits without squishing anything. You may need to adjust the wires a bit inside for everything to fit.

Now is a good time to rerun the test software. Connect the stepper motors, attach the 12V power cable and rerun each of the test routines to make certain everything still works. It is easy to knock the wires off of the pins so make certain everything is connected and working. If so you can attach the top to the bottom for now with the screws that came with the enclosure. There is a white string gasket that came with the enclosure, but since it isn't water tight anyway I would not use it.

Step 14 Final Touches

Once you have finished everything and either added the optional keypad or decided not to, take the small zip tie connector and wrap it around the cables coming through the large grommet on the inside of the enclosure to provide strain relief and keep the wires from being pulled out of the box.

Congratulations! You have now completed the hardware portion of the project!

5 Setup and Operation

This section describes the setup and operation of the Z-Field Dual Axis Controller using the buttons on the LCD display, or the optional hand controller. The next section will describe how to use the controller with your computer.

5.1 Installing the Arduino Code

If you have not already downloaded and installed the Arduino IDE and associated libraries, please do so now following the instructions in Section 3. Go to Z-Field_ControllerVx_x directory where X.X is the latest version. Double click on the Z-Field_Controller Vx_x.ino file to load it into the Arduino IDE. Connect the Z-FIELD Controller Unit to the computer with a USB cable and make certain the 12V switch on the Controller is in the off position. Upload the Arduino code to the controller. The Z-Field Controller should beep whenever it starts. Connect the Z-Field Controller to the stepper motors on your CG-4 telescope mount, connect a 12V power supply and turn on the Controller. It should beep and the RA motor should start driving the telescope at the sidereal rate. Operation of the Controller is explained in the next section.

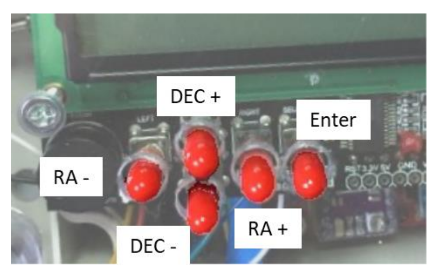


Figure 5-1 Key Layout

5.2 Basic Operation

Once you have downloaded the Arduino software, the Z-Field Dual Axis Controller should beep (if you have the LCD shield with buzzer installed) and you should see a display that looks like Figure 5-2. The key layout is shown in Figure 5-1. The leftmost 4 keys control the movement of the telescope mount just like the keys on the original CG-4 controller. However, there is an additional key to the right that is used to access the user menu

Figure 5-2 Initial Startup Screen

items. The display is shown in Figure 5-2. The initial configuration default is for the backlight to be Red, you'll learn how to change this in the User Menu section. The first line of the display (16X·Sid·East·EQN) shows the current guide rate setting (16X), the current drive rate (Sidereal), which side of the meridian the telescope is pointing (East) and the drive mode (EQN) for Equatorial North. The second line (8:05 Enter=Menu) shows the time (it is wrong until it is set properly) and what happens if you hit the enter key (it enters the User Menu). Pressing any of the other buttons will move the telescope in either the RA or DEC axis depending on keys pressed. If you press one of the keys, say DEC +, then press the opposite key (DEC -) while still holding the previous key, it will start moving at 16X no matter what guide rate is set. I call this Turbo mode and it is useful when aiming the telescope to move it quickly in a particular direction without having to constantly change the guide rate.

If you hear the buzzer beeping there is a good chance there is a problem. It **does** beep when entering the submenus, that is normal. A list of all status/error messages is given in Section 5.3.

5.2.1 User Menu

Menu
Exit

A summary of the Menu system is shown in Table 5-1. Pressing the Enter key when in the Default display will bring up the User Menu. The display should now show Menu on the first line and Exit on the second line as shown at right. The upper line will show which menu you are in and the lower line will show you what action will occur if you press the Enter key. If you press Enter here you will exit and go back to the Default screen shown in Figure 5-2. Pressing the left/right buttons (RA+ / RA-) will move you to different menus, pressing the up down buttons (DEC+ / DEC-) will move you through the items in that submenu. Exit is the only item in this submenu so the up/down buttons won't matter.

5.2.2 Guide Speed Menu

Speed
0.5X

Pressing the right button will get you to the Speed Menu shown to the right. Pressing the up/down buttons will scroll through the guide speeds: 0.5X, 2X, 4X, 8X and *16X. Note the "*" by *16X. This indicates that this is the current guide speed. To select a new guide speed, scroll to that speed and press enter. You will see the "*" appear next to it indicating it is the currently selected speed and it will now appear in the Default Screen.

5.2.3 Tracking Rate Menu

Rate
*Sidereal

Pressing the Right button again will bring up the Rate menu as shown at right. Note that the * is next to *Sidereal indicating it is the currently selected Drive rate. Scrolling up or down will bring up Solar and Lunar. Pressing the Enter button when any of these are visible sets the Drive Rate to that rate.

5.2.4 Display Menu

Display
*Buzzer ON

Pressing the Right button again will bring up the Display menu as seen at right. Scrolling through this menu allows you to turn on/off the buzzer and set the LCD Backlight to different colors: Off, Red, Green, Yellow, Blue, Teal, and White, the first one turning the backlight off entirely.

5.2.5 Configure Mount Menu

Configure Mount
Sidereal Time

Time: 08:51:01
SID.: 18:26:43

Set Latitude
33:08:32 N

Set Longitude
096:39:30 W

Set Time DST?
08:59:52 N

Set Date M/D/Y Z
02/20/2018 -06

Going right again will bring up the Configure Mount menu as shown at right. Note that the first item is Sidereal Time. Pressing Enter here will display the Sidereal Time based on the current local time and your location. If you have not set up the Local Time or location it will send you to the Set Location menu item to confirm/set the location and time. The controller uses your location and time to calculate the Sidereal Time so that it can avoid a Meridian flip. You will need to press Enter again to get out of the Sidereal Time display. Going down you will see Set Location. Pressing enter will bring up the display to set Latitude at right. By moving the cursor, you can set the degrees, minutes and seconds of the latitude by pressing the up and down keys. You can also change N (North) to S (South) if needed. Pressing enter will take you to the Set Longitude Menu. Enter your Longitude similar to how you entered your Latitude. You can change W (West) to E (East) if needed. Pressing enter takes you back to the upper menu and the Set Time/Date submenu item. Pressing enter here will take you to the Set Time window shown at the right. As you did with the Lat/Long enter the current time. Be sure to set N (No) or Y (Yes) as to whether you are currently in Daylight Savings Time. Pressing Enter takes you to the Set Date Screen. As with the previous menus, set the date. The last 3 digits (Z) are Time Zone so be sure to set this. Do not modify this for DST as the time/date calculations take this into account as long as you set the DST in the Time Menu.

After this is the *EQN and EQS to set Equatorial North or South.

Configure Mount
Enter=Defaults

Below this is Reset defaults. Every time you access and Exit the User Menu the changes you make are saved to non-volatile memory for use the next time you turn on the Controller. If, for some reason, things get real messed up you can press this key and it will reload the defaults set in the Arduino code. The display will show Enter=Defaults. Pressing enter here will reset the values to the default values. Press any other key to abort the operation. In the Configuring the Controller section you will be instructed on how you can change these default values.

Configure Mount
Version x.x

The last submenu item is Code Version. Pressing enter will show the window at right displaying the current version of the Z-Field Controller Arduino Code. X.X is the version number. Pressing any key returns you to the submenu.

5.2.6 GoTo Menu

GOTO
Enter Coords

Enter RA
02:54:24

Enter DEC
89:20:02 +

Pressing Right again brings you to the GOTO menu**.** If you go Left from the Menu / Exit point you will reach this menu. **This menu is only needed if you are going to do manual GOTO**. Or if you access the Stop Motors subcommand. You do not need this if you are accessing GOTO functions from the computer! The first menu item is Enter Coords, which allows you to enter the Right Ascension and Declination. You enter these similar to how you entered Lat and Long. What happens to these coordinates is determined by the next two submenu items. Enter RA allows you to enter the Right Ascension in Hours:Minutes:Seconds format. Pressing Enter brings up the Enter Dec box. Enter in Degrees:Minutes:Seconds and change + to – if needed. I should note here that the default is the RA and DEC for Polaris since this should be visible for the majority of users. Pressing Enter takes you back to the GOTO submenu.

Pressing Down you will see Sync. This is the first step in a Manual GOTO operation. You need to point the telescope at a bright object, then using the Enter Coords item above, enter the coordinates of the bright object.

Go down to the Sync item and press Enter. This tells the Controller the coordinates of where the telescope is currently pointed.

02:54 +89:20
02:32 +85:20

Pressing down again and you will see GoTo. If you have just done a Sync you need to go back to the Enter Coords item and enter the coordinates of where you want to move to. It should be within about 5 degrees of where you are pointing. After entering the coordinates, move down to the GoTo item and press Enter. At this point the display shows the coordinates of where you are pointing on the top line and where you are going to on the second line as shown at right. Pressing any key but enter will abort the GOTO operation and put you back at the GOTO screen. If you press Enter, the scope will move to the new coordinates.

Warning: Before performing a GoTo operation the Time and Location need to be set! The accuracy of the GoTo also depends on how well you are aligned on the pole. You need to have aligned either using a drift method (https://openphdguiding.org/tutorial-drift-alignment-with-PHD 2/) or using a polar scope or QHYCCD PoleMaster (http://www.qhyccd.com/PoleMaster.html)

Note that the software will not allow a GoTo that crosses the meridian. If you attempt to do so the controller will Beep with a warning message: “Merid Xing” being displayed in the status section of the display.

The last item in this menu is Stop Motor. This allows you to stop both drive motors. If you messed up in the previous GOTO operation and you realize the scope is slewing across the sky, way past where you wanted to go, then pressing Enter at this item will stop the motors and stop the GOTO operation. However, it will leave the telescope pointing to an unknown position so you will need to go back and realign on a bright object, enter the coordinates of that object, then do a Sync, etc. If the display is showing *Stop Motor then the motor should be stopped. Pressing Enter again will restart the RA tracking motor. This command is also useful if Drift Aligning the mount.

5.2.7 Menu Summary

A summary of the SubMenus is given below. Note that the motion is circular so if you go left from the Exit submenu you will enter the GOTO submenu. And going right from the GOTO submenu will enter the Exit submenu.

Menu	**Speed**	**Rate**	**Display**	**Configure Mount**	**GOTO**
Exit	.5X	Sidereal	Buzzer ON	Sidereal Time	Enter Coords
	2X	Solar	Buzzer OFF	Set Location	Sync
	4X	Lunar	Backlight OFF	Set Time/Date	Goto
	8X		Red	EQN	StopMotors
	16X		Green	EQS	
			Yellow	Reset defaults	
			Blue	Code Version	
			Violet		
			Teal		
			White		

Table 5-1 Summary of the Menu System

5.3 Status Messages

The following Status/error messages may appear in the status portion of the display accompanied by a beep. The higher frequency the beep the more serious the message is. Pushing any of the buttons clears the message.

Display	Buzzer	What to do
Menu	500hz beep for .2 sec.	This is normal, nothing needs to be done
MenuTimeOut	100hz beep for .1 sec.	If you go into the menu system and do not exit, after 1 minute the Controller will force an exit and give this message to let you know. No further action is required.
Bad Menu	200hz beep for .5 sec.	You should never see this. It means the software has gotten somewhere without a command. If you do see this report it to info@z-field.com.
PC Change	100hz beep for .1 sec.	This is to let you know that code on the computer software has made a change. Will usually occur when setting the location or time from the computer. No action required/
PTO ## ###	100hz beep for .1 sec.	PC Time Out. This means the software was waiting for data from the PC that never came. The ## ### should be two numbers. Send these numbers along with which software you were using to info@z-field.com.
Track Err	2500hz beep for .2 sec.	Means that the tracking routines have sensed an error. May need to reset the controller. If this continues to occur contact info@z-field.com.
Track Off	2500hz beep for .2 sec.	Normal. Response to the Turn Motors Off command. This is just to let you know the motors are off.
Merid Xing	2500hz beep for .2 sec.	If you attempt to do a GoTo across the Meridian, the Controller will give this message and not allow it. Can occur under both manual and software control. To go to an object on the other side of the Meridian you need to reposition the scope to a bright object on the same side as the object you are attempting to move to and redo the GoTo operation.
SetDateTim	2500hz beep for .2 sec.	If you attempt a GoTo operation and you have not set the Location and/or Date you will get this message. If you are doing a manual GoTo you should be automatically pointed to the Location Menu that also takes you to the Enter Date/Time menu. If you are using software to control the mount make sure you have set the Location and Date/Time either in software or manually.

Table 5-2 Status Messages

5.4 Configure the Controller

Plug the RA stepper cable into the RA motor and DEC cable into the DEC motor as you did when you were testing the hardware. Turn on the Controller. If you haven't aligned the East/West tilt switch do so now. Put the tilt switch into the Right Ascension axis using the set-screws as shown in Figure 4-20. The tilt switch should be horizontal. With the telescope on top and parallel with the RA axis, as in Figure 4-20, twist the tilt switch housing around until the LCD display reads West when twisted slightly clockwise and East when twisted slightly counter clockwise. Set it about in the middle of these and it should be aligned good enough. Unlock the RA axis and move the scope to point to the west and the display should read West, then move it to point east and make certain the display says East. If it reads the reverse twist the housing 180 degrees and align per the previous paragraph. After you have finished aligning the tilt switch you should use a Sharpie to make a mark on the housing and another mark on the RA axis so you know where to align it if you need to remove it.

You should now be ready to run using the commands given in the previous section.

5.5 Changing the defaults

You will probably want to change the default values in the Arduino code to match your situation. To do so, load the current version of the controller software (You will need to do this each time you upgrade) and search for "void defaultConfigs" . You should see:

```
void defaultConfigs()
{
  Configs.Zfield=99;          // If the first character is 99 that says the data has been written previously and is valid
  Configs.eeAccess=0;              // Number of times the data has been accessed
  Configs.RAEQN=false;            //Direction of RA step moves motor counter-clockwise in EQN, clockwise in EQS
  Configs.DECEQN=false;           //Direction of DEC step moves motor clockwise in EQN, counter-clockwise in EQS
  Configs.enBUZZ=true;              //Enable Buzzer 0-Buzzer Off  1-Buzzer allowed
  Configs.EQN=true;                // True = EQ North  False=EQ South
  Configs.guideSpeed=5;              // 1= .5x 2=2X, 3=4X, 4=8X, 5=16X, 6=40x (Slewing only)
  Configs.keys=0;                 // 0=Normal 1=Reverse RA 2= Reverse DEC 3=Reverse RA and DEC, 4=Flip RA/DEC
5=Flip+Rev RA, 6= Flip+ Rev DEC, 7=Flip+RevRA+RevDec
  Configs.blColor=1;               //  1= Red, 2= Green, 3=Yellow, 4=Blue 5=Violet 6=Teal 7=White 8=Off
  Configs.LatDeg=0;
  Configs.LatMin=0;
  Configs.LatSec=0;
  Configs.LatDir=0;
  Configs.LongDeg=0;
  Configs.LongMin=0;
  Configs.LongSec=00;
  Configs.LongDir=1;
  Configs.itemEnable=itemEndef;
```

Table 5-3 Default Configuration Code

Ignore Zfield and eeAccess as these are used to determine if the data is valid and the number of times it has been accessed. If you find that for some reason (maybe your stepper cables are wired backward) that the motors move backward you can change that with the RAEQN and DECEQN. You can change from false to true to change the base direction. You do not need to change these if you are using EQS as it is handled in the stepper driver.

enBUZZ=true enables the buzzer, changing it to false will default to buzzer off.

EQN, guideSpeed, drvRate are described in the comments. Keys=0; is not implemented yet, but would flip the keys to reverse DEC+/DEC-,etc.

The last ones are the Lat and Long. You can change these to your Latitude and Longitude. LatDir=0 for North and 1 for South. LongDir = 1 for West and 0 for East.

itemEnable sets the "*" items in the submenus so don't change it.

Do not change slewDelay=1500 unless I tell you to! This sets the minimum delay for the stepper motors to function properly. Changing this can cause the motors to stop working!

EastSide=true sets the default on the tilt switch. Since this is calibrated when you set up the mount you shouldn't have to change this.

5.6 Updating the Software

From time to time you should check the z-field.com site to see if there are new updates to the software. If there is a new version you will need to download the software and reload the Controller with the new software per Section 5.1.

6 Other Software

6.1 ASCOM

In order to use the Controller with any commercially available software you will need to download and install the latest ASCOM platform from https://ascom-standards.org/ . The site gives instructions on how to do this. ASCOM is a standard platform that allows a common command set to be used across software.

6.1.1 Celestron ASCOM driver

After installing the ASCOM platform you will need to download and install the Celestron driver from https://ascom-standards.org/Downloads/ScopeDrivers.htm . Scroll down the page until you see Celestron Unified (6.0.6338, June 2017) and Download and install this driver.

6.1.2 ASCOM Pad

ASCOM PAD is freeware that will run the Z-Field Controller through the ASCOM Celestron Driver. You can use it to move the telescope using your mouse from your computer. It is a very useful piece of software and I would recommend installing it to test out the Z-Field Controller. You can download it from http://eq-mod.sourceforge.net/apindex.html at the bottom of the page.

After installing the software, open it up and go to the Devices tab and scroll down to Mount then Setup. A Mount Control Setup window should appear. Click on the Select button under the ASCOM frame. The ASCOM Telescope Chooser window will appear. Click the down arrow and select the Celestron Telescope Driver. Then click the Properties… button. Your screen should now look like Figure 6-2. In the COM Port window select the COM port the Z-Field Controller is attached to. See Section 3 on how to find what COM Port it is using. Note that you can also set EQN or EQS here as well. Once you have set the COM port click OK on the two ASCOM windows and close the Mount Control Setup window. The window shown in Figure 6-1 should now pop-up. This is actually from the Celestron ASCOM driver, but I think it is better than the ASCOM PAD interface for running the mount. If you are using other software that uses the Celestron ASCOM driver you will not need to run the ASCOM Pad program at the same time. The buttons will move the mount in that direction and the selection window at the bottom will select a speed that the mount will move. The CG-4 Mount is limited to 50x Sidereal so anything faster than that will default to 50x. At the bottom of the ASCOM Pad window you will see several symbols: Star, Moon and Sun. Pressing these will change to drive rate to Sidereal, Lunar and Solar respectively.

ASCOM PAD has some quirks. Often it takes several attempts to get it to open the Celestron driver so if it doesn't work at first try it again. If you cannot see the COM port in the setup window click on the Advanced button and click on Show All COM Ports. You will notice there is not a close or exit button. This is my biggest complaint about ASCOM PAD. The only way I know to close it is to go to the hidden icons (Right side of the

MENU bar at the bottom of the Windows screen) and click the ^ arrow. You should see the ASCOM Pad icon. Right click and choose Exit.

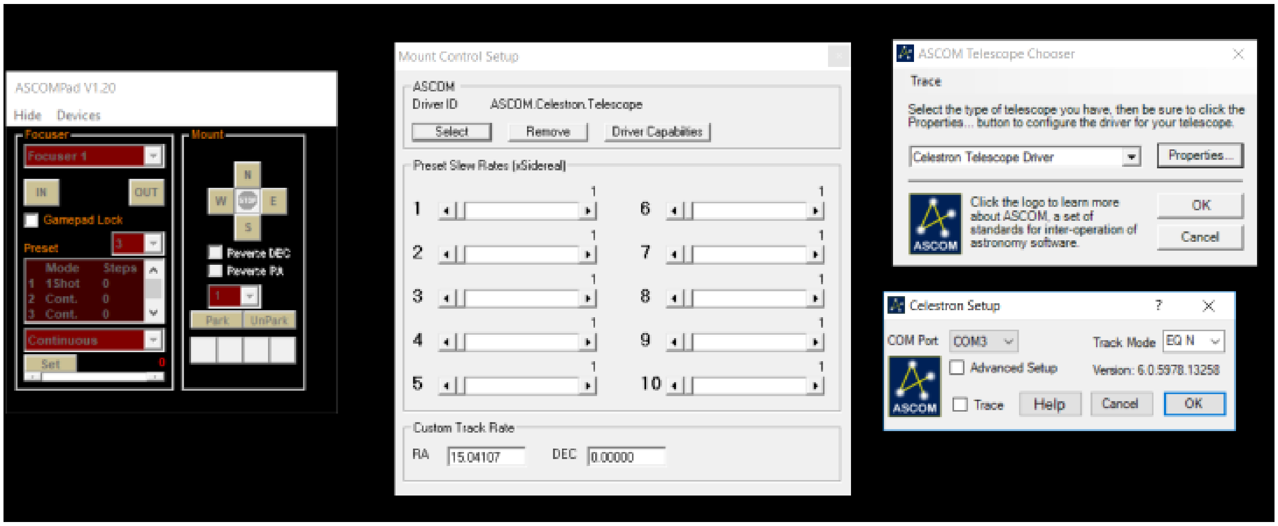


Figure 6-2 ASCOM PAD Setup Windows

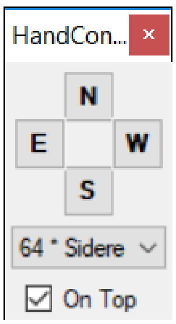


Figure 6-1 Celestron Driver Control

6.2 PHD 2

The Z-Field Controller supports PHD 2, a guiding program that uses the ASCOM platform. PHD 2 can also be used to drift align your mount: https://openphdguiding.org/tutorial-drift-alignment-with-PHD 2/ . When it says to "Slew" you will need to manually position your scope to a star near that location.

Information on PHD 2 including instructions on how to download and install are at https://openphdguiding.org/ . Note that to do guiding you will need a separate guide scope (or off-axis guider) and camera. As a side note, guide cameras used with PHD 2 often have an ST-4 autoguide port that will interface with the Optional Keypad port described in Section **Error! Reference source not found.**. Even if you have i nstalled the Keypad interface, PHD 2 usually runs better through the ASCOM interface directly to the controller.

6.3 Planetarium Software

The Z-Field Controller will work with any planetarium software that supports the ASCOM platform. However, be advised that it is very limited. Since it is not a true GoTo mount system you need to realize the limitations. In general the software needs to support the Synchronize (Sync) as well as GoTo functions. The way the GoTo in the Z-Field Controller works is by pointing it at a star, Selecting the same star using the planetarium software, then then issuing the Sync function. This command sends the coordinates of the star you are pointed at to the Z-Field Controller so it knows where the scope is pointed. Then, when the software issues a GoTo command it knows how to move the motors and in what direction. Because the Z-Field Controller is Open Loop (it does not have anything to measure exactly where it is pointing), you should only select nearby objects to GoTo, usually less than 10 degrees away. You then need to recenter the object you went to in the field of view of either the camera or eyepiece (whichever you are using) the do another GoTo hop to the next object. Using this method it is possible to go all the way across the sky. However, if you have to do a Meridian Flip you will need to find another starting star and start the process over.

I have tested the Z-Field controller with Starry Nights Pro and know that it works. I have also tested it with Stellarium. But, Stellarium does not have a Sync function do you cannot really use it.

I have not tested Sky X, but I'm certain it will work. In the next section I will describe how to use Cartes du Ciel, a freeware planetarium like software that works very well.

If you plan to use other software becides Cartes du Ciel, you should still read the next section to get an idea of how to use the Controller with your software.

6.3.1 Cartes du Ciel Star Charts

I've found that this freeware program is an excellent program to control the Z-Field Controller. You can find download and installation instructions at https://sourceforge.net/projects/skychart/ . I will not explain all the functions of this software, except the ones needed to setup the Z-Field Controller and do simple GoTos. I leave the rest as an exercise for the reader. On initial startup it should ask you your location. Enter it and your time zone information. If you move to a different location be sure to go to the Setup Menu and select Observatory to enter your new coordinates. It is important that you have the correct coordinates and time zone entered in the software (and your computer) for the software to work correctly with the Z-Field Controller. After setting up your location and time information, go to the Telescope menu at the top and click Telescope Settings. Go to the Telescope Tab and make sure ASCOM is selected. Exit that menu then go to the Telescope menu and select Connect Telescope. That will bring up the Driver Selection window. Click the Select button and, as with the ASCOM Pad software select the ASCOM.Celestron.Telescope and set it up as described in the ASCOM Pad section above. Now click the Connect button at the bottom. The Red patches should turn green and you should see the Celestron Control window appear that allows you to move the scope using their buttons. Don't close the Driver Selection window yet! You should see your Latitude and Longitude displayed in the Observatory frame. If it is not correct exit the window and go to the Setup – Observatory window to enter the correct location and time zone information. Then come back to the Telescope-Connect window. If the Latitude and Longitude are correct press the Set Location button. This sends the location information to the Z-Field Controller. Now press the Set Time button to send the time to the Controller as well. You should see the time change on the LCD display of the Controller. You can exit the Telescope Connect window and you should be ready to control the Z-Field Dual Axis Controller with the software.

NOTE: If the Time and Location have already been set using the buttons on the controller you will get a message when you press Location or Time shown in Figure 6-3. This just means that the location and/or time have already been set. I would recommend disconnecting and shutting down the Controller then reconnecting and powering back on so that you use the software to set the location and time as it is more accurate.

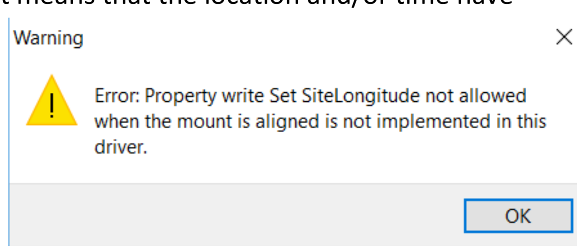


Figure 6-3 Location/Time error message

If this is the first time you use the software you might want to go to the Setup – Display menu. Under the Findercircle (eyepiece) Tab you can select what sized circles to display the telescope field of view. Make certain that at least one is selected. You can play with this later. As stated earlier, you need to manually point the telescope at a bright star. I prefer a bright star as the locations of the

Moon and planets change. But, the Moon certainly is easy to find and center. Once you have the bright object centered in your camera or eyepiece you need to right click on that object in Cartes du Ciel and select Centre on. This will move the object to the center of the screen. Right clicking on the object again, go down the popup to Telescope. This will pop up a new menu. Select Sync: (object name). A popup window will appear asking you if the scope is pointed at the object. Click Yes. A set of rings should appear over the object that correspond to the Finder Circles set by the Display menu. The software will do the Sync with the Z-Field controller sending it the coordinates of the object. The Z-Field Controller now knows where it is pointing. Now select a nearby (<10 degrees) object, right click, move down to Telescope and select Slew: (object name). The telescope will now move to that object. You should see the finder rings, that show the location of the telescope, change on the screen as the scope moves. Since the DEC axis actually moves 3 times faster than the RA axis the DEC movement will usually finish first and you'll see it only moving in RA at the end, this is normal. As stated earlier, because the control is open loop the object will probably not be in the center of the camera field of view and/or the eyepiece. Be sure to manually (using either the Celestron HandControl that popped up earlier or the Z-Field Controller buttons) move the object to the center. There is no need to re-Sync as the Controller and the software know where the scope is pointing. Repeat this process until you reach the object of interest.

You should also set the gain up on your camera to allow it to see fainter objects. Once you find the object of interest you can turn the gain back down to do imaging.

It is best to go out, on an evening when you have some time, and practice with the software and controller so you get an idea of how it functions. If the scope seems to stop moving it is possible that it has reached its limit vertically and you need to do a meridian flip. You need to do the flip manually, then you'll need to re-center a bright star, Sync and start the process over again. Also, make certain that the 12V power is turned on as it is very easy to forget to turn it on. Even without the 12V on, the software will appear to be operating fine, except it is unable to move the stepper motors. However, even though the motors are not moving, it will appear to be moving on the screen.

7 Optimizing the controller

7.1 Modifying Drive Rates

The drive rates (Sidereal, Solar, Lunar) for the Z-Field Dual Axis Controller rely on the clock of the Arduino Nano, which operates at 16Mhz. However, the actual clock rate may vary slightly between different Nanos which may result in the drive rates not being accurate. In the config_t structure there are three variables, SidRate, SolRate and LunRate. In the void defaultConfigs() routine these are set to the default values:

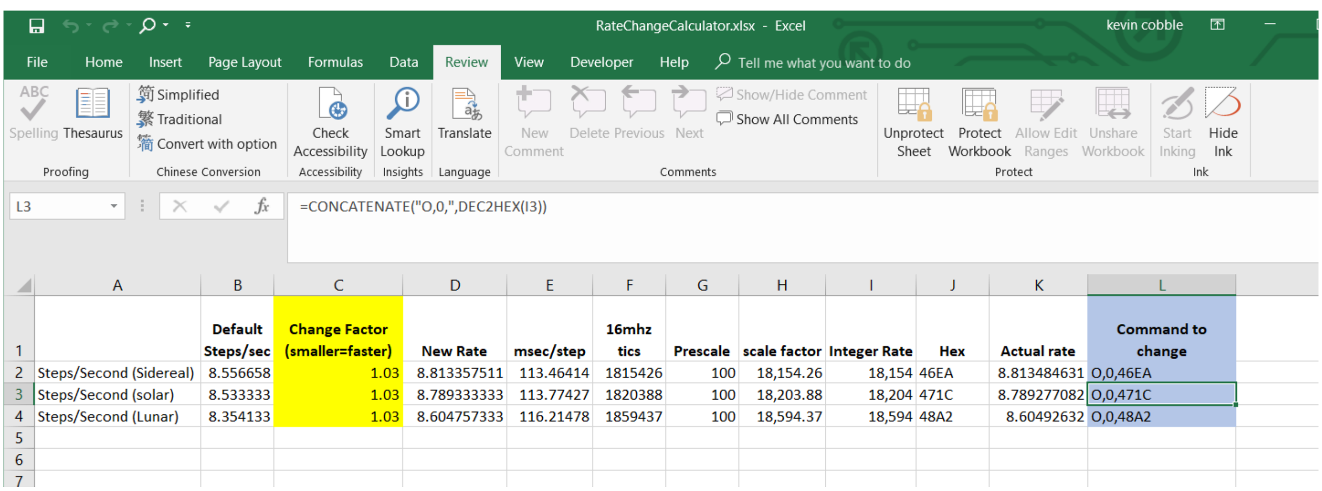
a. SidRate=18699;
b. SolRate=18750;
c. LunRate=19152;

The interrupt timer uses these values to determine how many of the 16Mhz clock tics to count before going to the stepper motor routine to move the stepper motor. In the stepper motor routine there is a further counter that divides these values by 2, then by another value determined by the slew rate in effect. For the basic drive rate this is divided by 50, giving a final division of 100. Other values less than 50 give the ability to do 0.5x, 2x, 4x, 8x and 16x speeds for moving using the buttons.

One way to change these are to change them directly in the code. Ultimately, if you decide you need a different rate than the defaults, you can do this. Once they are set you will need to go into the Configure Mount Menu and do a Reset Defaults to load these into the saved configuration.

WARNING! And this is a big warning: if you make these values too small (less than 10000), it can affect the ability of the controller to function and may lock it up. There are checks in the software that if these values are less than 15000 it will load the default instead. However, if it locks up you'll need to start with a clean version of the code, then change the *#define ForceConfigs* 0 to 1. That will force the code to reload the configuration from scratch and save the new version. Once it is reset, be sure to put this value back to 0 or it will keep loading the defaults.

The better method is to load them from a serial terminal program (such as the one on the Arduino or RealTerm or some other terminal program). There is an Excel program, DriveChangeCalculator.xlsx, included with the code download after V1.01 that will help you find the right value to send to the controller.



	Default Steps/sec	Change Factor (smaller=faster)	New Rate	msec/step	16mhz tics	Prescale	scale factor	Integer Rate	Hex	Actual rate	Command to change
Steps/Second (Sidereal)	8.556658	1.03	8.813357511	113.46414	1815426	100	18,154.26	18,154	46EA	8.813484631	O,0,46EA
Steps/Second (solar)	8.533333	1.03	8.789333333	113.77427	1820388	100	18,203.88	18,204	471C	8.789277082	O,0,471C
Steps/Second (Lunar)	8.354133	1.03	8.604757333	116.21478	1859437	100	18,594.37	18,594	48A2	8.60492632	O,0,48A2

Figure 7-1 Drive RateChangeCalculator.xls display

When you open the serial terminal make certain there is no other software running that connects to the controller (ASCOMPad, Cartes du Ciel, etc.) as this may not allow the serial terminal to connect. Also, be certain the No Line Ending is set if you are using the Arduino monitor. You may need to use the Device Manager to find the serial port to connect to. Once you are connected you can issue the O command to change the drive rate. The command structure is O,#,XXXX where # is 0 for Sidereal, 1 for Solar and 2 for Lunar drive rate to change. XXXX is the number you want to change it to in Hexadecimal.

Using the Excel spreadsheet, you can modify the Change factor around 1 and it will calculate the correct command to send. In the Command to change column you can copy the command and paste it in your serial terminal to change it. By issuing an N command (Just send N) it should respond with the current settings of the drive rates.

Column B of the spreadsheet is the default steps per second the stepper motor needs to take for the default drive speeds. Column C is the Change Factor, New Rate is the modified steps per second, msec/step is just that, 16mhz tics, in floating point. Scale factor is the new number you are going to download to the controller (if you are permanently modifying the code as stated above use these values). Hex gives the rate in hexadecimal. Actual rate is the steps per second using the integer stepping rate. Command to change give the command to copy and paste in your serial terminal program to send to the controller.

The Arduino software checks to make certain that the value you enter is in the range 15000 to 25000.

When you issue the rate change command to the controller it will set the new value, set the current drive mode to that rate (if you were in Sidereal mode and set the Solar rate it will set the drive to Solar mode) then it will save the values to permanent memory.

You do not need to put these values into the actual code (as stated earlier) as they will be picked up when the configuration is read from memory even if you load an updated version, unless otherwise stated in that version.

I would recommend making small changes to the Change Factor (+/- .01) as the difference from the default rate will be small.

To set the sidereal rate I point the camera at a bright star and turn on the reticle in the Firecapture software and put the bright star in the center of the reticle. Then observe if the star drifts to the west (lower the change factor by .01 steps) or to the east (raise the change factor by 0.1 steps). Do the same with a sunspot on the sun for the solar rate and a crater peak for the Lunar rate.

8 Optional Keypad

This section describes the construction of a keypad that will attach to the LCD Display board and will allow you to keep the Z-Field controller on the mount or nearby table and use the keypad to control it remotely from the eyepiece or computer. If you are going to primarily run the controller from the computer with only occasional standalone work, building the keypad may not be necessary. But, if you plan to use the controller primarily without a computer then I would recommend building one. The keypad can be added at any time. If installed per the instructions below it can also act as an ST-4 Autoguider port.

You will need the following parts (I put a link next to each with the parts I used, but you can substitute similar parts):

Plastic Enclosure (any enclosure that the buttons fit in will work.
https://www.amazon.com/gp/product/B019GFLNV4/ref=oh_aui_search_detailpage?ie=UTF8&psc=1

5 momentary contact push buttons
https://www.amazon.com/gp/product/B0094GP7SQ/ref=oh_aui_search_detailpage?ie=UTF8&psc=1

Six conductor telephone extension STRAIGHT cable https://www.amazon.com/TENINYU-Conductor-Straight-Telephone-Extension/dp/B06WRWCV11/ref=sr_1_fkmr0_1?ie=UTF8&qid=1521158556&sr=8-1-fkmr0&keywords=6c6p+telephone+extension+6+foot

Six conductor, RJ12 Straight Inline Coupler (Make certain it is Straight Thru and not Reversed) https://www.amazon.com/gp/product/B004C4XO6M/ref=oh_aui_search_detailpage?ie=UTF8&psc=1

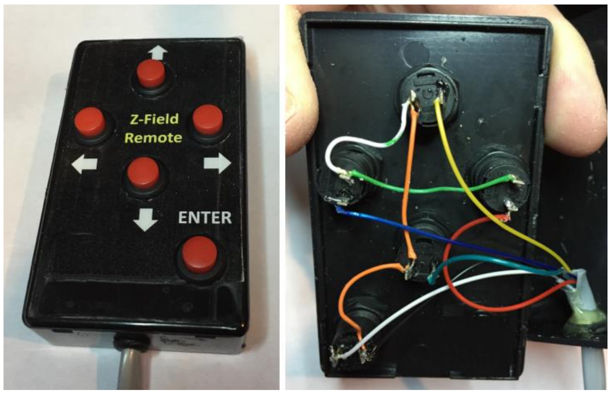


Figure 8-2 Remote Keypad outside and inside

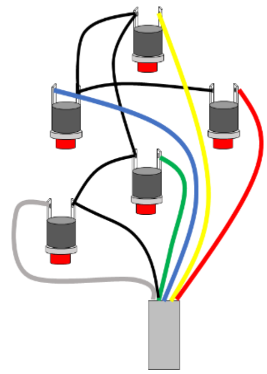


Figure 8-1 Keypad Wiring Diagram

Figure 8-1 and Figure 8-2 show how the remote is setup. Drill 5 holes into the top of the enclosure that are the right size to hold the push buttons you bought. Install the buttons into the case top. Drill a hole in the box side bottom for the cable. Cut an end of the 6-conductor telephone cable about 8 inches long and set aside. On the long piece, trim about 3 inches of the phone cable jacket off and run the stripped end through the hole in the bottom. Then, attach the wires according to the diagram. The colors ARE important!

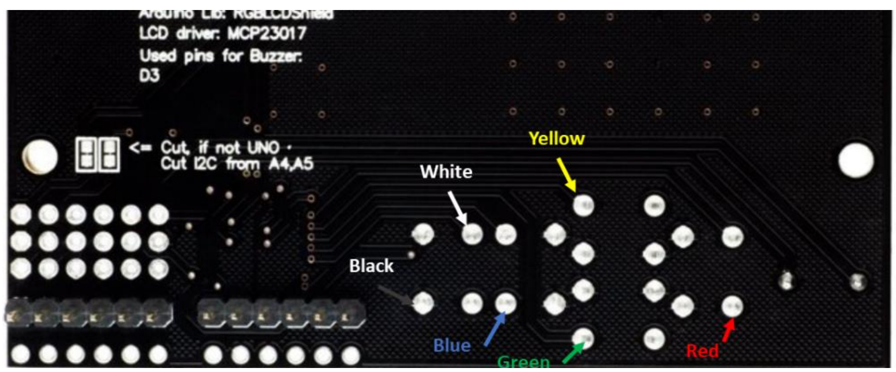


Figure 8-3 Where to solder the Keypad wires on the back of the LCD Display

Take the short end you set aside and strip about 2 inches of the jacket off the end. Run the stripped end through the large grommet in the Z-Field Controller enclosure. The connector should be hanging outside the enclosure.

Strip and solder the wires to the back of the LCD Display according to Figure 8-3. Also, see Figure 4-32 to see how I did it. Now, connect the two connectors on the phone cables with the coupler and you're done. Run the RGBLCD-Button-Buzzer_Test test routine to check to make certain it is wired correctly.

Once it is working I would recommend you hot melt glue the wires to the back of the LCD module to provide strain relief.

9 Abbreviations

The Unit, Controller and Z-Field Controller all refer to the Z-Field Dual Axis Controller.

Arduino Nano- The main controller board for this Z-Field controller.

DEC- Declination

EEPROM- Electrically Erasable Programable Read Only Memory, Memory on the Arduino that where the program is stored.

FOV- Field of View, Area of sky, in degrees, seen through a telescope using the eye, a camera or other instrument.

GEM- German Equatorial Mount, The CG-4 is a GEM

LCD – Liquid Crystal Display

RA- Right Ascension

RGB- Red – Green – Blue

10 Troubleshooting

The most likely part to fail is the Arduino Nano. If you input too high a voltage the voltage regulator on the Nano will be damaged and the unit will not work. You can buy replacement Nanos on Amazon or other online sources.

The stepper controller board (either DRV8825 or A4988 depending on what board you purchased) are usually very robust so you probably will not have a problem with them. The stepper control kit comes with three so there is a spare if necessary.

Problem	What to do
Unit does not turn on	Make certain there is power to the board. If it works when connected to the USB plug but not to the 12V check the 12V power supply to make certain it is working. Also check that power is getting to the board by measuring the voltage on both sides of the Zener Diode as shown in the assembly instructions. Check to see that the LED on the Arduino Nano board is flasing about once a second or so. If not there may be a problem with the code and/or nano board. If there are no lights on the Nano board even when plugged into the USB of the computer than the Nano board may need to be replaced. If there are lights, but they are not blinking try reloading the Arduino software.
LEDs are on but nothing is showing on the screen	If the Backlight on the LCD is not on check the connections to the LCD. IF the backlight is on then try reloading the Arduino software.
Motors do not run, but are buzzing	Adjust the current adjust screw on the small motor control boards as described at the end of Step 10.
Motors do not run and no buzzing	Check the connections of the motors as described in Step 10.
Only East (or West) shows on the display when the RA axis is rotated	Check the Tilt switch as described in Step 9. Also, take the tilt switch assembly out of the RA axis and listen very carefully as you twist it to see if you can hear the ball inside moving. If not, or if it still does not work, you may need a new tilt switch.
The Arduino software will not download	Make certain the power switch is turned off per the instructions in Step 8. If you cannot download with the switch in either on or off position check the Processor setting on the Arduino IDE per Section 3. Sometimes you need to turn it off, disconnect then reconnect the USB and try again.