

Cloudy Nights → Equipment Discussions → ATM, Optics and DIY Forum



\$37 Dobsonian DSC for DIY makers

Started by Adun , Aug 26 2017 07:58 PM

DIY, dob, mount,

Page 1 of 18

Adun

Posted 26 August 2017 - 07:58 PM

This project is an implementation of a simple Digital Setting Circles, that evolves from my previous DSC project, but now using an inexpensive optical encoder (and GT2 timing belt+gear) instead of the magnetometer.

This DSC achieves the following:

- Easy and inexpensive to build, for a DIY maker or Arduino enthusiast.
- Can be attached to dobsonian mounts, included tabletops.
- High resolution: 10.200 counts per revolution (compare to the 11.748 counts per revolution of the SkyWatcher Goto Dobs)
- Instead of a wired handheld control, it communicates with Android/iOS apps like Skysafary 5 (plus or pro), DSC-Browser and others via WiFi.

My prototype is a removable version installed on a Zhumell Z114 tabletop:



You can see I'm using a woodworking clamp to affix the encoder in the position required to have proper tension in the timing belt. This way, the DSC is very easy to remove from the scope without leaving any permanent marks.

Parts list:

\$3.05 NodeMCU ESP8266 development board (version 12E) ([link](https://www.aliexpress.com/item/1pcs-lot-NodeMcu-Lua-WIFI-Internet-of-Things-development-board-based-ESP8266-esp-12e-for-arduino/32775934156.html) (<https://www.aliexpress.com/item/1pcs-lot-NodeMcu-Lua-WIFI-Internet-of-Things-development-board-based-ESP8266-esp-12e-for-arduino/32775934156.html>))

\$4.20 LSM303DLHC sensor module ([link](https://www.aliexpress.com/item/1-pcs-GY-511-LSM303DLHC-Module/) (<https://www.aliexpress.com/item/1-pcs-GY-511-LSM303DLHC-Module/>))

[E-Compass-3-Axis-Accelerometer-3-Axis-Magnetometer-Module-Sensor/1956617486.html](#))

\$17.99 Signswise 6oop/r Incremental Rotary Encoder DC5-24v 6mm shaft ([link](#)
(<https://www.amazon.com/gp/product/BooUTIFCVA>))

\$9.99 85 teeth GT2 timing pulley 10mm shaft ([link](#) (<https://www.aliexpress.com/item/POWGE-1pcs-85-Teeth-GT2-Timing-Pulley-Bore-5-6-35-8mm-for-width-6mm-GT2/32773386105.html>))

\$0.82 20 teeth Gt2 timing pulley 6.35mm shaft ([link](#) (<https://www.aliexpress.com/item/20tooth-Bore-6-35mm-GT2-Synchronous-Pulley-Aluminium-Timing-Gear-For-Width-6mm-Belt-3D-Printers/32808150854.html>))

\$1.05 GT2 timing belt (size depends on your telescope size), I used 400mm for the Z114 ([link](#)
(<https://www.aliexpress.com/item/Free-Shipping-3D-printer-belt-closed-loop-rubber-GT2-timing-belt-400-2GT-6-teeth-100/32477498985.html>))

Subtotal parts cost: **\$37.1**

Add some solder, cables, a 5V power source, a couple resistors, and screws for attaching the encoder to the base whichever way you want. A maker can build this DIY DSC for much less than the cost of two high resolution optical encoders and a NexusDSC.

The key to this new version of my DSC is the use of an inexpensive optical encoder for azimuth. The \$17.99 encoder I used has 6ooppr, which using quadrature produces 2400 counts per revolution, which would not be enough for a DSC, but using GT2 timing pulleys of 85 and 20 teeth (for a ratio of 4.25 to 1) we can now get 10200 counts per revolution on azimuth, for just a fraction of the price of a 10k encoder.

This is how the GT2 pulleys and belt look in the base of my Z114:



For simplicity and cost savings, altitude is driven by the same high-res accelerometer I used in my previous DSC, the LSM303DLHC which has more than enough resolution to match the azimuth encoder.

The microcontroller is a very inexpensive ESP8266, I used the nodeMCU variant which seems to be tolerant of the 5V inputs from the encoder. You can use any 5V power source you want, I'm using a power bank:



The large box is the power bank, the smaller box is housing the main circuit with the ESP8266.

To use it you just connect your tablet to the WiFi access point created by the device, configure SkySafari 5 Plus (or Deep Sky Browser, or other program) to connecto the IP address of the device (by default: 1.2.3.4 port 4030), select "Basic Encoders" for Telescope Type, and voilà! SkySafari is now connected to your scope.

I've published the schematics for the circuit (which is really very simple) and the source code for the microcontroller on [GitHub](https://github.com/vlaate/DobsonianDSC) (<https://github.com/vlaate/DobsonianDSC>).

Edited by Adun, 26 August 2017 - 08:01 PM.

hamishbarker

Posted 27 August 2017 - 06:59 AM

top work! Next on my projects list after I finish a polar axis VNS (vertical north segments) platform for my homebrew 8"/f6 dobsonian. (Tonight's activity was cutting out and filing smooth the first of the two north segments from the bottom of a \$1 dump shop aluminium frying pan. Play abandoned at 10:30pm due to it being too late in the evening to risk disturbing neighbors with metal cutting.)

tony_spina

Posted 27 August 2017 - 09:18 AM

Great work Adun

lakland5

Posted 27 August 2017 - 10:09 AM

Thanks Adun! I'll try to get this made before the next new moon 😊
Best,
Ric Ashley, Evanston, IL (lakland5)

synfinatic

Posted 28 August 2017 - 11:05 PM

Nice work Adun! I love the mix of technology and the low price!

mark cowan

Posted 28 August 2017 - 11:35 PM

Very cool. 

aeajr

Posted 29 August 2017 - 11:09 AM

I need to study this.

Adun

Posted 29 August 2017 - 05:18 PM

synfinatic, on 29 Aug 2017 - 06:05 AM, said:

Nice work Adun! I love the mix of technology and the low price!

The 3 enabling technologies:

* Low cost 60oppr encoder and GT2 gears+belt are meant for 3D printing, and the current boom in 3D printing is driving their costs down.

* The high resolution accelerometers are used a lot for drones, and there is another boom in drones that is driving the cost of accelerometers to very low levels.

* The microcontroller, ESP8266 is a rather recent spin on the arduino/microcontroller ecosystem, a spin driven by the up and coming boom of in the "internet of things", where toasters and fridges are expected to be connected to the internet, and make your coffee when you awaken, etc. The makers of the dominant wifi microcontrollers have a lot to gain, so they are betting big. I almost can't believe I can get an 80 Mhz 32 bit microcontroller with integrated wifi, for \$3.05 retail.

So there you have it: 3D printing, drones, and IOT are three technology booms that are converging to make this completely unrelated gadget (dobsonian push-to) cheaper than ever. Hopefully it will permeate commercial prices soon.

solitary

Posted 31 August 2017 - 07:03 AM

Wow! This deserves to be stickied. Can't wait to start on this project myself. The ESP8266 is a really versatile controller, I've enjoyed using it in the past.

hamishbarker

Posted 04 September 2017 - 03:16 AM

hamishbarker, on 27 Aug 2017 - 1:59 PM, said:

top work! Next on my projects list after I finish a polar axis VNS (vertical north segments) platform for my homebrew 8"/f6 dobsonian. (Tonight's activity was cutting out and filing smooth the first of the two north segments from the bottom of a \$1 dump shop aluminium frying pan. Play abandoned at 10:30pm due to it being too late in the evening to risk disturbing neighbors with metal cutting.)

Equatorial platform is now operational, I will post separately! A bit of a frankenstein but works, with caveat (board is too flexy, need to add more stiffening).

hamishbarker

Posted 05 September 2017 - 05:34 PM

Microcontroller, sensor board and one of the pulleys has arrived. Hopefully china post is not going to be too slow on the rest of the parts. I also have a (rather pricier at \$34 bosch bno055 fusion sensor board (mag/gyro/accel), so I'll install them both and see how comparable the two systems are.

Adun

Posted 05 September 2017 - 07:28 PM

hamishbarker, on 06 Sept 2017 - 12:34 AM, said:

Microcontroller, sensor board and one of the pulleys has arrived. Hopefully china post is not going to be too slow on the rest of the parts. I also have a (rather pricier at \$34 bosch bno055 fusion sensor board (mag/gyro/accel), so I'll install them both and see how comparable the two systems are.

A couple things you can check out while you wait:

- * The bottom screw from your mount might be a bit short for the added height of the GT2 gear. I had to get a larger screw for mine, at the local hardware store, so that it was long enough to go through: bottom board, top board, nut, GT2 gear.
- * The large pulley is meant to stand still (relative to the bottom board) while the whole scope rotates around it. This means the screw needs to be affixed to the bottom board so that it doesn't rotate, you may want to find a "non permanent" way to do it at first. I used a flat bike hexagon wrench with double sided tape.

Also with the sensor and microcontroller you can try the DSC with azimuth standing at zero. This way you can start ironing out the software, and finding calibration values for the sensor board (the maximum and minimum values for each axis), as well as how to connecting to it from a tablet or PC. You can get that done while the encoder, pulley and other parts arrive.

hamishbarker

Posted 05 September 2017 - 11:12 PM

Yeah getting software etc going sans azimuth encoder while waiting for the rest of the parts was my plan.

Big question will be how far from the tube i need to mount the fusion sensor to avoid stray magnetic effects as my dob has a galv sheet steel duct for the main tube.

Built my scope myself so there will be no need to be precious about how the large pulley is fixed. I'm planning on drilling and tapping an m4 screw into the centre of the main bolt.

hamishbarker

Posted 07 September 2017 - 04:24 AM

Hi Vladimir,

I tried flashing the board but had an error as I didn't have the following header files because they were not in the github repository. Were they some general ones from arduino ide or did you write them and forget to include on the upload to git?

Cheers,
hamish

Missing files:

```
#include <Encoder.h>
#include <Wire.h>
#include <LSM303.h>
#include <ESP8266WiFi.h>
```

hamishbarker

Posted 07 September 2017 - 05:14 AM

don't worry, got the libraries sorted and uploaded code to the board.

Adun

Posted 07 September 2017 - 07:53 AM

hamishbarker, on 07 Sept 2017 - 12:14 PM, said:

don't worry, got the libraries sorted and uploaded code to the board.

Hmm maybe I should add some comments in the code clarifying which libraries to import.

Edited by Adun, 07 September 2017 - 07:53 AM.

hamishbarker

Posted 07 September 2017 - 08:38 AM

I think your wiring diagram might not match the code, but I might have just misread it.

Code says join GPIO 0 (D3) to SDA and GPIO2 (D4) to SCL while the diagram shows the reverse. I followed the wiring and it didn't work, then checked the code, reversed connections and it works (previously it was failing to find any i2c devices).

altitude sensing by the LSM303 is working, so all ready for the encoder and other bits when they turn up. Might try to get the BNO055 also working and make a small box to stick on top of binoculars with alt/az display. Even better would be to open up a pair of binocs and install tiny dim LED numerals in the focal plane so that one could sweep around, find something interesting and then be able to read off alt/az numbers to look up said interesting object.

Edited by hamishbarker, 07 September 2017 - 08:40 AM.

Adun

Posted 07 September 2017 - 04:57 PM

hamishbarker, on 07 Sept 2017 - 3:38 PM, said:

I think your wiring diagram might not match the code, but I might have just misread it.

Code says join GPIO 0 (D3) to SDA and GPIO2 (D4) to SCL while the diagram shows the reverse. I followed the wiring and it didn't work, then checked the code, reversed connections and it works (previously it was failing to find any i2c devices).

I just checked the code, and I think you are right about the I2C wires.

The diagram is as I designed it, but upon building it, since I reused the sensor from another project (it was already in the tiny black box, with wiring already done) I may have quickly reversed them in code, and after I glued the box I didn't recheck and failed to update the diagram. I'll fix that on GitHub.

hamishbarker, on 07 Sept 2017 - 3:38 PM, said:

altitude sensing by the LSM303 is working, so all ready for the encoder and other bits when they turn up.

You can fine-tune the "EMA_WEIGHT" and "MEASUREMENT_PERIOD" parameters -if you want- to your liking. Using a MEMs accelerometer to read altitude (inclinaiton) requires the use of a smoothing algorithm to get a stable reading (otherwise, the accelerometer provides very jumpy readings from one milliscond to the next, making the skysafari cursor jump).

This smoothing algorithm introduces tiny bit of lag between when you move the telescope, and when you see the cursor move in your sky map. You can tune this lag to your preference using those "EMA_WEIGHT" and "MEASUREMENT_PERIOD" parameters:

EMA_WEIGHT is a percentage, a higher value will make the cursor catch faster with the scope motion, but too high will allow the accelerometer oscillation to get through. You'd notice it when the scope is standing still and the reading oscillates a little. A lower EMA_WEIGHT kills this oscillation and gives you higher resolution, but adds lag. I'm using 5%, but you can tune it to whatever you feel more comfortable with.

MEASUREMENT_PERIOD is a millisecond amount. A lower value will make the cursor catch faster with the scope motion, reducing the lag. You can try to balance EMA_WEIGHT and MEASUREMENT_PERIOD to equilibrate the responsiveness of the accelerometer to the exact point you like. In another project (magnetometer DSC) I used as low as 20ms, but since there's an encoder here, and WiFi, I didn't want to run the risk of the ESP8266 missing interrupts (encoder steps) so I used a conservative 50ms, but it could perhaps go lower. I didn't want to test the limits of the ESP8266.

Edited by Adun, 07 September 2017 - 11:05 PM.

hamishbarker

Posted 07 September 2017 - 09:23 PM

Yep thanks for the tips and of course the code in the first place.

Will try the magnetic azimuth based code and also incorporate and compare with bno055 sensor performance while waiting for the parts.

GShaffer

Posted 08 September 2017 - 03:04 PM

Link to this excellent topic has been added to the sticky "Best of ATM" at the top of the forum.

hamishbarker

Posted 09 September 2017 - 06:16 AM

hooked up the Node MCU with both the LSM303 and BNO055 sensors now, but have only run them individually as I can't be bothered merging the adafruit code and types for BNO055 with the LSM303 library. But there is an adafruit library for the LSM303 so I guess I could run them simultaneously - although I wonder if either of them puts out any stray magnetic field which might interfere with the other.

Anyway, AS USUAL WITH NEW EQUIPMENT ready to use - it is cloudy. And not just common-or-garden variety little bit of overcast, rather a series of hefty cold fronts and lots of snow falling on the mountains across the bay. I did some test-alignment indoors (pointing to light fittings and spots on the ceiling) with the BN0055 and the performance seemed pretty good. I used Vlad's code for eliminating problems around 360 to zero transition (which can otherwise mess up averaging position smoothing algorithms) and it seemed pretty smooth and stable at all azimuth and altitudes. It will be interesting to run it under the sky.

I also had a couple of ideas about how it should actually be possible to use a similar printer timing belt and cheap encoder for altitude also, in case the accelerometer/gyro/magnetic sensor tactic doesn't perform well enough. the advantage over the more standard altitude encoder on a stick arrangement is that both encoders could be mounted to the rocker board. the altitude encoder would have the big timing gear mounted to the center of rotation of the altitude bearings. The belt could be demounted for breakdown. Both encoders would be physically very close to each other, so no cables, and just wifi to starry night running on the phone.

. Advantage would be no fragile/breakable altitude encoder mounting stick. And potentially the same timing belts could also be serpentine run around an extra pulley so that it could be driven by a stepper motor for full goto functionality. All pretty

much wireless and all four shafts (2 encoders, 2 steppers) mounted quite close, potentially in a single, demountable enclosure for transport breakdown.

I'll make a sketch to illustrate my thoughts.

Edited by hamishbarker, 09 September 2017 - 06:16 AM.

Adun

Posted 09 September 2017 - 09:40 AM

The go-to idea is interesting.

I considered adding steppers, but I thought it would require a lot more gear reduction for tracking to work, since steppers have much less resolution than the 60oppr optical encoder.

But maybe there's a small gearbox that could be used.... Or maybe just try a servo?

I didn't solve that though experiment. I started thinking an EQ platform would be best, which on my latitude (3°N) would be more akin to barn door (per Ed Jones advise)... I'm waiting for a linear drive to experiment on that.

AlanT

Posted 09 September 2017 - 10:03 AM

Adun, on 07 Sept 2017 - 2:53 PM, said:

hamishbarker, on 07 Sept 2017 - 12:14 PM, said:

don't worry, got the libraries sorted and uploaded code to the board.

Hmm maybe I should add some comments in the code clarifying which libraries to import.

I'm also having trouble finding the right libraries. I'd would greatly appreciate some links or instruction on which are the correct ones.

Thanks

Adun

Posted 09 September 2017 - 12:12 PM

AlanT, on 09 Sept 2017 - 5:03 PM, said:

I'm also having trouble finding the right libraries. I'd would greatly appreciate some links or instruction on which are the correct ones.

Thanks

I just added clarifications on the GitHub page and in the source code.

When installing the libraries on the Arduino IDE, these are the ones you need to choose:

Encoder Library by Paul Stoffregen (I used version 1.4.1)

https://www.pjrc.com...bs_Encoder.html (https://www.pjrc.com/teensy/t..._Encoder.html)

LSM303 Library by Pololu (I used version 3.0.1)

<https://github.com/p.../lsm303-arduino> (<https://github.com/pololu/lsm303-arduino>)

The other libraries, Wire (I2C) and ESP8266 I used are just the built-in ones from the Arduino IDE

Edited by Adun, 09 September 2017 - 03:01 PM.

hamishbarker

Posted 10 September 2017 - 03:50 AM

I'll

Adun, on 09 Sept 2017 - 4:40 PM, said:

The go-to idea is interesting.

I considered adding steppers, but I thought it would require a lot more gear reduction for tracking to work, since steppers have much less resolution than the 60oppr optical encoder.

But maybe there's a small gearbox that could be used.... Or maybe just try a servo?

I didn't solve that though experiment. I started thinking an EQ platform would be best, which on my latitude (3°N) would be more akin to barn door (per Ed Jones advise)... I'm waiting for a linear drive to experiment on that.

Push-to does keep the system simpler, lower power and less whiney motors.

There are these 1/16 geared, 32 step steppers, very cheap on fleabay = 512 steps per rev. might be ok for visual. Not sure if the input can be rotated or if some sort of friction drive would be necessary so that one could still push to objects or fairly quickly hand-slew to a general vicinity.

I guess one could rig up friction drive roller from the stepper to the flange of the encoder belt pulley, that way when push-to slewing, the belt pulley could slide along on the stepper drive roller.

I have built an EQ platform (top board was too flexy so I rebuilt it today - clamps and glue are on now), and bought a little basic package celestron RA drive for \$30 as it all seemed too hard and the package drive option seemed pretty good value (and it works great). Having now tested the waters of microcontroller programming and seeing how easy the arduino stuff is, those plus steppers seems a good choice, I think, for drives or camera tracking platforms/barn doors in future especially with things like the nodeMCU being so cheap! And I guess it would be easy to implement a guiding port into such a controller.

[edited] I could write a step by step howto for the libraries with screenshots and add to the github page if Vlad adds me temporarily as contributor.

Edited by hamishbarker, 11 September 2017 - 02:39 AM.

Equipment Discussions → ATM, Optics and DIY Forum → Solar projector telescope Started by VilleM , Today, 11:33 AM 3d printing , ATM , DIY , Visual	2 replies 98 views VilleM Today, 01:04 PM
Equipment Discussions → Mounts → Recommended scope for iOptron skyguider pro Started by Adus72 , Today, 07:14 AM Astrophotography , Beginner and 3 more...	12 replies 148 views Adus72 Today, 03:01 PM
Equipment Discussions → ATM, Optics and DIY Forum → Looking for a partner to build a simple lens-mount assembly (3-D printed housing + diamond-turned lens) Started by 97yusuf23 , Today, 02:35 AM 3d printing , Optics , Mount and 3 more...	0 replies 143 views 97yusuf23 Today, 02:35 AM
Equipment Discussions → Mounts → Celestron AVX Mount Started by tcmattox589 , Yesterday, 11:23 PM Celestron , Mount , Visual	6 replies 210 views Martinbruce Today, 11:32 AM
Equipment Discussions → Equipment (No astrophotography) → Telescope Refractor Astromaster Celestron 90 F11 after update of focuser Started by Reg_Astro , Yesterday, 02:26 PM Celestron , Accessories , EQ , Mount and 2 more...	2 replies 136 views Jon Isaacs Yesterday, 04:05 PM

Cloudy Nights → Equipment Discussions → ATM, Optics and DIY Forum



Cloudy Nights LLC
Cloudy Nights Sponsor: [Astronomics](#)