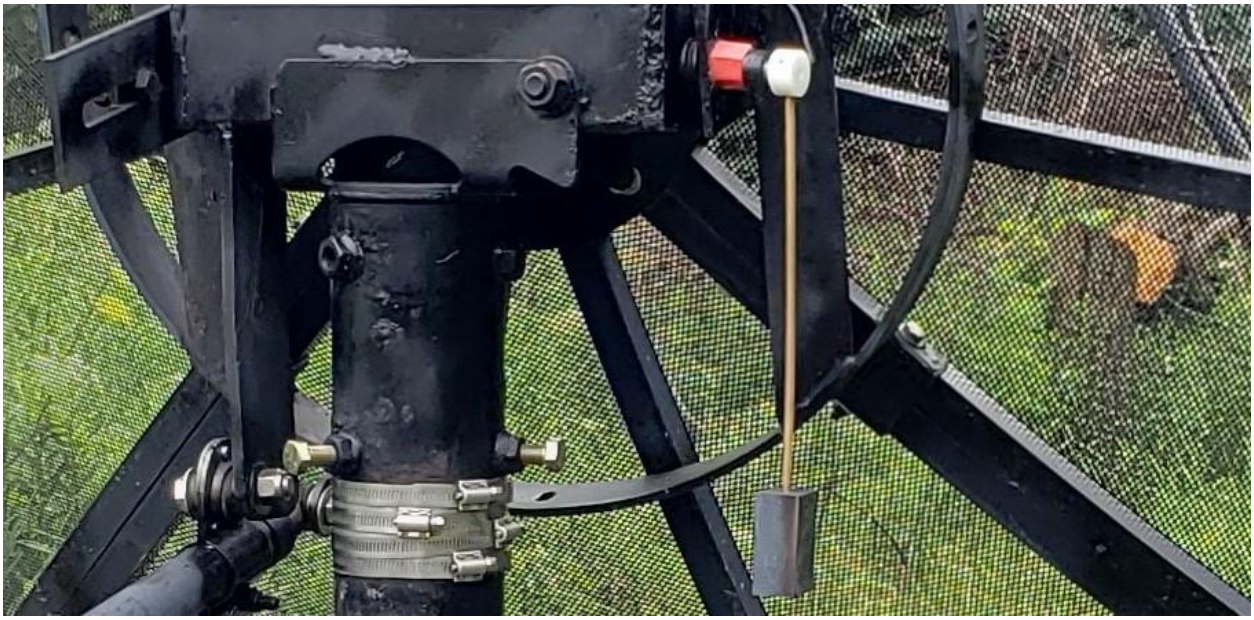
A simple Arduino-based AZ-EL readout/display

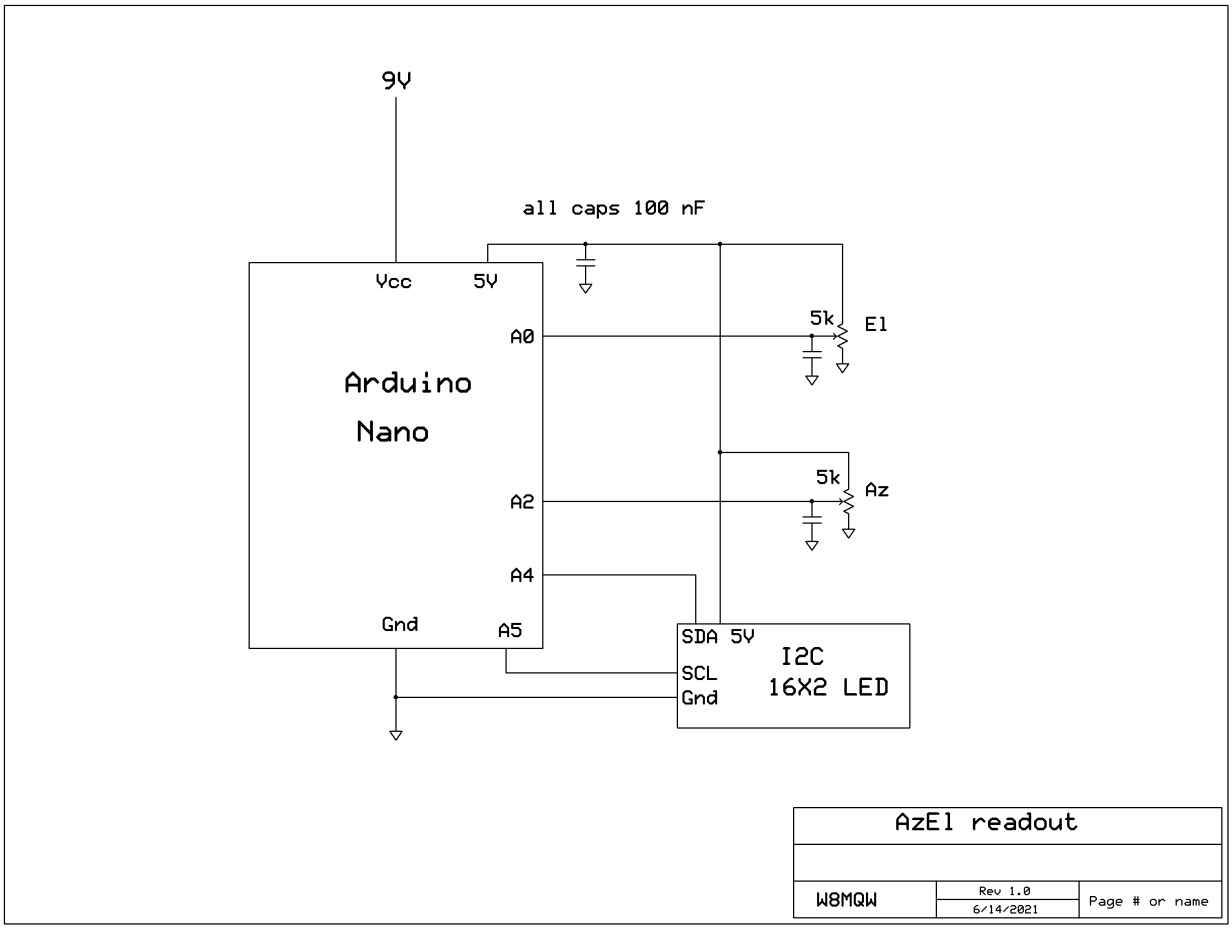
Chuck MacCluer w8mqw

I would like to share a simple, accurate, and very inexpensive technique that I use to readout the position of my 1296 dish. The azimuth is read by a 5k potentiometer driven by a robotics belt:

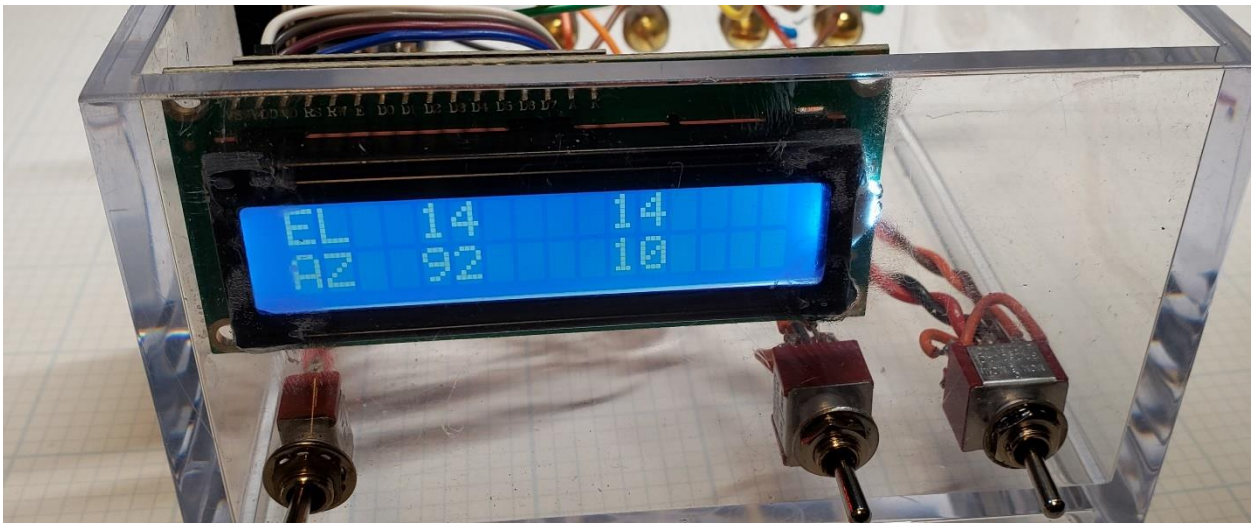
The EL is read by a 5k potentiometer affixed to the moving dish with a pendulum dangling from its shaft:



Five volts is applied to the top of each pot, the bottom grounded, and the wiper read by the Arduino in the shack. The circuit is simplicity itself:



The visual display is an inexpensive 16x2 LED with a I2C daughterboard that is mounted simply with dabs of glue at its four corners into a transparent box available from [1]:



(The second column of numbers in the above display is the A-D capture of the 1024-level resolution of the wiper voltage appearing at the analog inputs of the Arduino. These numbers are handy to have during calibration of the readout as you tweak the scaling functions at the end of the program.)

The above photo is my complete manual control box controlling the dish --- the left-most switch is ON/OFF, the middle and right-most switches are the momentary AZ and EL control.

[1] Acrylic Bloc. The size shown is 4 x 4 x 1.5 by Russel-Hazel

https://www.containerstore.com/s/russell-hazel-acrylic-box/d?q=acrylic%20bloc&p=0&ps=52&productId=11007603

SCRIPT

```
/* This reads two pots, one on dish azimuth, the other on elevation and displays via I2C to a 16x2 1602A LCD display*/

#include <Wire.h>

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

//**************** SETUP *********************//

void setup() {

  lcd.init();

  lcd.backlight();

  lcd.setCursor(0, 0);

  lcd.print("EL");

  lcd.setCursor(0, 1);

  lcd.print("AZ");

 }/************ MAIN *************************/

void loop() {

   char  value[5];  //this will be the string place holder for each value printed

   int ELpot = analogRead(0);   //read elevation pot (returning a number between 0 and 1023)

 int AZpot = analogRead(2);   //read azimuth pot  (ditto)
```

```
  lcd.setCursor(4, 0);  //ready to write elevation in degrees
  dtostrf(scaleEL(ELpot),3,0,value);  //convert scaled EL reading to a string
  lcd.print(value);   //write elevation to LCD
  lcd.setCursor(10, 0);
  dtostrf(ELpot,4,0,value);
  lcd.print(value); //write EL A/D bits  */

  lcd.setCursor(4, 1);  //ready to write azimuth in degrees
  dtostrf(scaleAZ(AZpot),3,0,value);  //convert scaled AZ to a string
  lcd.print(value); //write AZ string to LCD
  lcd.setCursor(10, 1);
  dtostrf(AZpot,4,0,value);  //convert A/D bit reading to a string
   lcd.print(value);          //write AZ A/D bits
delay(500);
}/************ FUNCTIONS **********************/
int  scaleEL(int el) //translates elevation pot A-to-D reading (0--1023)to actual elevation
  {float deg = 0.28159 * el - 26.57; /* this uses the calibration data and point-slope formula*/
   int    EL = int(deg + 0.5);
return EL;
}  /*********************************/
int scaleAZ(int az)       // this translates azimuth pot A-to-D reading into actual azimuth
  {float deg = 0.2571 * az + 59.2; //after a careful Sun track
            /* the above uses the calibration data and point- slope formula*/
      int      AZ = int(deg + 0.5);
  return AZ;
}
```