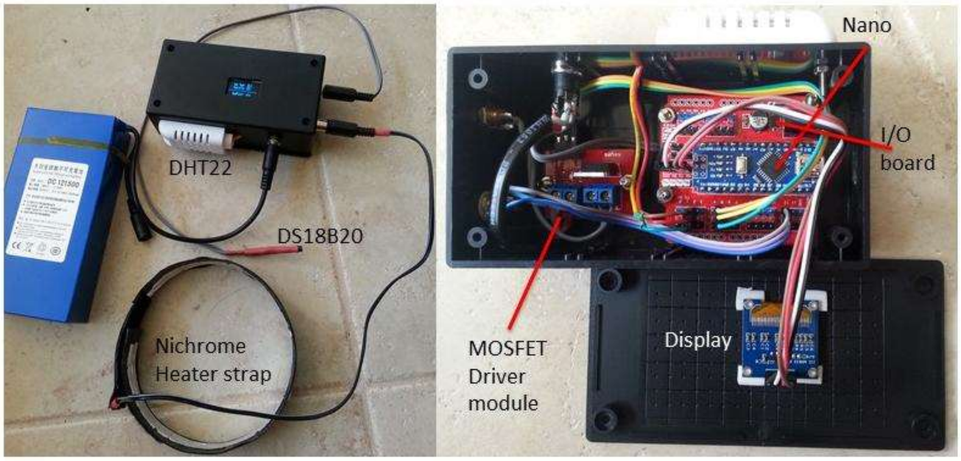
Update 1 Jan, 2018

EASY TO MAKE ARDUINO DEW HEATER CONTROLLER

I needed a dew heater, so thought it was a good time to revisit the excellent Dew Heater controller built in IceInSpace 2010 by Bob Stephens. I had two reasons for doing this:

1. There are some great modular arduino products which make it extremely easy to make. The Nano has a great extension I/O board so all the components CAN BE attached with pre-made connectors. There is NO circuit board making. There is NO soldering onto circuit boards. The ONLY soldering is for the plugs and sockets to connect it to the outside world.
2. Arduino based modules are now very cheap. Apart from the heater straps it costs ~$40 for a 1x channel build (Jan-Feb 2016).
3. Except for the heater it uses very little power. In fact, you can leave it on all the time to monitor temp/humidity – only plug in the heater if needed.

The 1st attached pic show it connected and its inside connections. This is the build with just 1x heater/sensor.



How the controller works & software

This is the same as before. The controller is similar to the build by Bob Stephens (with additional liberation of software from other sources). It is Arduino based and feedback controlled, briefly:

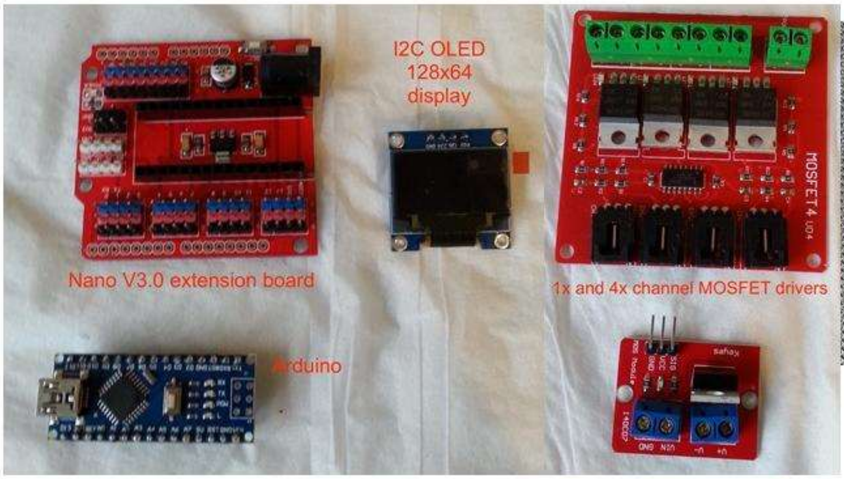
- DHT22 senses ambient temperature & humidity. The dew point is calculated.
- DS18B20 temperature sensor placed inside the heater strap senses the temperature of the element to be heated. You'll need 1x for each bit of glass to be heated.
- Calculate the difference = (heated element - ambient dew point).
- Send a PWM output from the Arduino based on this difference. I have used a threshold for activation of the heater of 5 or 6^{o}C above ambient dew point (this is what most seem to use - can be changed in the software). So the PWM output = 0% at +5 ^{o}C (and above this) to 100% at 0 ^{o}C.
- This PWM output drives the heater strap via a MOSFET driver module. The heater output increases as the temperature difference between heated element and ambient decreases.

- The display swaps between showing (1) the ambient temperature, humidity and dew point and (2) the heater temperature, difference to dew point, and heater drive level.
- The software will automatically determine which of the heater sensors are connected and control their heaters; it will also display the data for these sensors/heaters. The other heater MOSFETs can either be (a) turned off or (b) set manually to a level as a % of the maximum for that heater.

Except for the heater it uses very little power. In fact, you can disconnect heater elements and their temperature sensors and it will just monitor ambient temp & humidity.

<u>The parts</u>

Fig #2 shows the main Arduino bits. I have highlighted a possible change. Most of these bits are available on ebay or your local electronics store.



Connectors. Best to buy a big pack
Of single female-female. They come
Together so you can pull of 3x or 4x to
Makes the connections to sensors etc

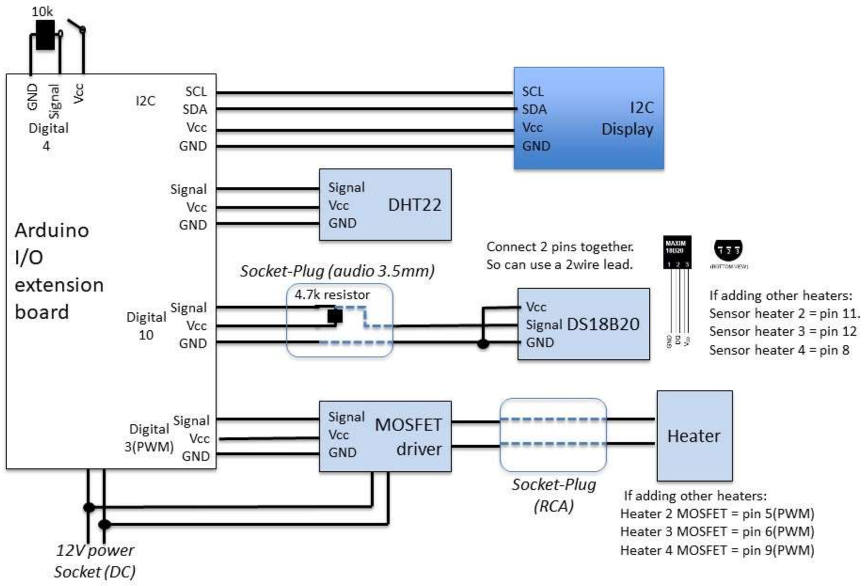
- Arduino bits: Arduino Nano; Nano V3.0 prototype shield I/O extension board expansion module*.
- The sensors: DHT22 Digital temperature/humidity sensor; DS18B20 digital temperature 1-wire sensor (need 1x DS18B20 for each heater that you want feedback temp control). The beauty of these is that they are digital – they are plug and play as they do not need calibrating.
- Display: I used a 0.96" I2C SPI serial 128x64 OLED display - it is small, so if you only want 1x heater channel it all fits in a small light (box) that I Velcro onto the scope. You can use other display eg 20x4 LCD display. Any display is fine just as long it is an I2C serial display as it only require 2x connection leads (it wont use up all of the Arduino analog/digital ports if you need multiple heater channels) plus 2x power leads.
- MOSFET driver for heater: If you only want 1-2 channels get single channel modules. If you want 4 channels there are 4 channel modules. I have found these deliver adequate power for guidescopes up to an 8" SCT. ***NB: if you are worried about power and overheating, use a heatsink on the mosfet, or try a higher power MOSFET. They are available from lots of places.***
- Hardware: ~$10-15 for Jiffy box, plugs/sockets, wires and resistors.

- Heater strap: Whatever you want. You can buy commercial ones. I make my own from nichrome. Its much easier than resistors (and not difficult to connect to wire as some would have you believe). See end for building straps.
- The code allows 1 – 4 heaters. So you will have to get more DS18B20 and MOSFET drivers for multiple channels. This is going to generate more heat so you would have to consider a bigger box and ventilation.

*The nice thing about the extension I/O board is that each Arduino analog/digital/I2C input/output has pins for that input/output & +5VDCC/GND. So you can just plug the pre-made connectors into the IO board. For the MOSFET driver and display just plug the other end into that module. For the sensors (DHT22, DS18B20), just cut one end off and solder onto the appropriate lead or socket (and cover with heatshrink = wonderful stuff).

The circuit

The circuit and how you put it together in Fig #3.



Connect 2 pins together.
So can use a 2wire lead.

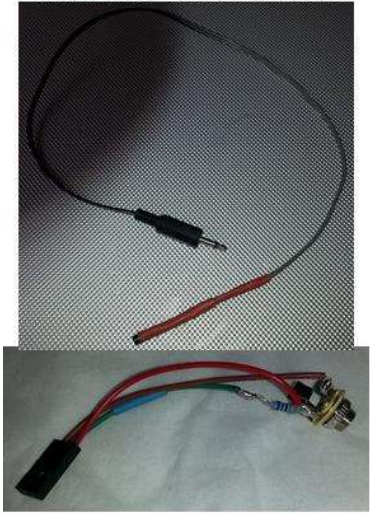
If adding other heaters:
Sensor heater 2 = pin 11.
Sensor heater 3 = pin 12
Sensor heater 4 = pin 8

If adding other heaters:
Heater 2 MOSFET = pin 5(PWM)
Heater 3 MOSFET = pin 6(PWM)
Heater 4 MOSFET = pin 9(PWM)

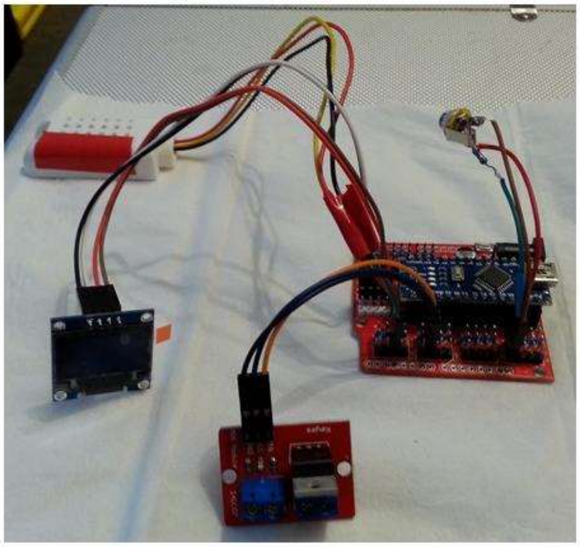
A few things to note:

- The DHT22 attaches to the outside of the box. So get a 3x wire connector, cut off one end, solder to the DHT22 leads and cover soldering with heatshrink (1.5 – 2.55mm). See Fig #4.

DS18B20
1. lead with DS18B20
2. socket with resistor



Put together the basics & test



- Have used parasitic mode for the DS18B20 (so only need 2x wires instead of 3x). So you can't go wrong wiring it up. Just connect the 2 outside ones together – these go to GND; the central one goes to the signal of digital 12. If you cover up all the solder joints and its connections to the 2 core wire with heatshrink it is nice and watertight (see above pic).
- MOSFET driver connections = easy.
- If you have just 1x heater:
    - Heater1: MOSFET = digital pin3 (PWM) & DS18B20 sensor = pin10
- If more heaters:
    - Heater2: MOSFET = digital pin5 (PWM) & DS18B20 sensor = pin11
    - Heater3: MOSFET = digital pin6 (PWM) & DS18B20 sensor = pin12
    - Heater4: MOSFET = digital pin9 (PWM) & DS18B20 sensor = pin8
- The serial display – just needs 4x wires. To attach this to the box – I just cut a hole and stuck it on with two sided tape (you can see it on the 1st Fig).
- Power – run the 12V input to both the Arduino extension IO board and the MOSFET driver.

Getting it going

Before you do the whole build its best to put it together in stages and test each bit.

1. First put Arduino on your PC or Mac.

2. There might be a bunch of updates to boards & libraries.

3. Next you'll need to upload the following libraries used by this program. To do start Arduino & select Sketch/Include Library/Manage Libraries. From here type in each of the following in the search tab. When you find it click on that library and it will come up with an installation button.
    - Adafruit_GFX_Library.

- Adafruit_SSD1306.
- DallaTemperature.
- DHT_sensor_library.
- OneWire.

4. Open the attached sketches & save them. If you are unsure which ones are needed, just compile the program in Arduino and it will tell you which libraries still need to be loaded (as in step 3 above).

5. Before uploading it to your build you need to tell it what Arduino you have. So under Tools – select Board/Arduino Nano; Processor/ATmega238 & Port/"whatever your Arduino comes up as" as this differs for PCs & MACs.

6. Upload up the attached sketch to your arduino.

7. At this stage its best to put it together in stages to check each bit works. I have added sketches that test each bit individually.

8. First the display. The one I got has an I2C address 0x3C, but some have reported that theirs is 0x3D. So if you see nothing try that – its set at the start of "void setup() {".

9. Test the DHT22 so just plug it into digital pin 2. Run the sketch DewHeaterControllerV4_a-test-DHT22. It tests that the sensor which measures ambient temp & humidity. It outputs this to the serial monitor on your computer. To get the serial monitor to work, in Arduino select Tools/Serial Monitor. It will come up with a window that receives input from your Arduino. It should read the data from the DHT22 every couple of seconds, give you the status of it (OK or error) and the temp/humidity & calculate the dew point.

10. Test the DS18B20 – so just plug one into digital pin 10. Run the sketch DewHeaterControllerV4_b-test-DS18B20. It tests that the sensor which measures temperature for each heater. It outputs this to the serial monitor on your computer, as for the DHT22. It will read the data from each DS18B20 every couple of seconds and give you the heater number, the Arduino pin it is on, status (OK or error) and the temp if it was OK.

11. Then try the DS18B20 on pin 11, then 12, then 8 if you have 2 -4 channels – just to check they work. Then try all your DS18B20s at once.

12. Test both the DS18B20 and DHT22 and the MOSFET driver. So plug in both the DHT22 and the DS18B20 (as above). And plug in the MOSFET unit (but don't connect it to a heater starp). Run the sketch DewHeaterControllerV4_c-test-DHT22-DS18B20. It tests both, so will output all of the stuff in (8) & (9) to the serial monitor. You can also check the MOSFETs are activated as they have an LED on them – just blow on the DHT22, the humidity will push the dew point up and activate the MOSFET.

13. Now plug everything, including the display . Check it works – the modules have an LED than turns on when its activated – so you can check before connecting the heater strap. Finally, connect power to the MOSFET (directly from the 12V power source NOT from the Arduino – see the diagram above). Connect a heater strap – check that it heats up.

14. Put it all together in a jiffy box. Upload the sketch DewHeaterControllerVx-x. Clear skies!

Update 1 Jan, 2018

Heater Straps

Building heater straps is cheap and simple. Everyone has their own way of doing it, so I not going to go into detail (can put up detail if needed). I use nichrome wire – much simpler than resistors and its relatively easy to connect nichrome to wire (unlike some make out):

1. Get the correct length of nichrome (comes in 3m lots for $4-5 from Jaycar) so that 12V makes its really warm. The size/wattage really depends upon you scope and how bad your conditions get.
2. You want the nichrome nice and straight (no kinks) – to do this pull it straight by making it red hot over your stove.
3. Cover the nichrome with 1.5mm heatshrink except for 5-10 cm at each end – this will electrically isolate it. Again easier to do it over your stove.
4. Connect the nichrome to wire leads by twisting them together, soldering (just to hold in place) then putting 2x layers of heatshrink over the top to hold the connection together. This works fine – mine has been good for a few months so far.
5. Lay the nichrome on aluminium tape - this will be on the scope side – holds it together & passes heat efficiently. You can get this from Bunning etc.
6. On the other side a few layers of good gaffer tape then felt tape to keep in the heat (both from Bunnings).
7. Put a bit of Velcro (from Bunnings) on the ends to hold it together.