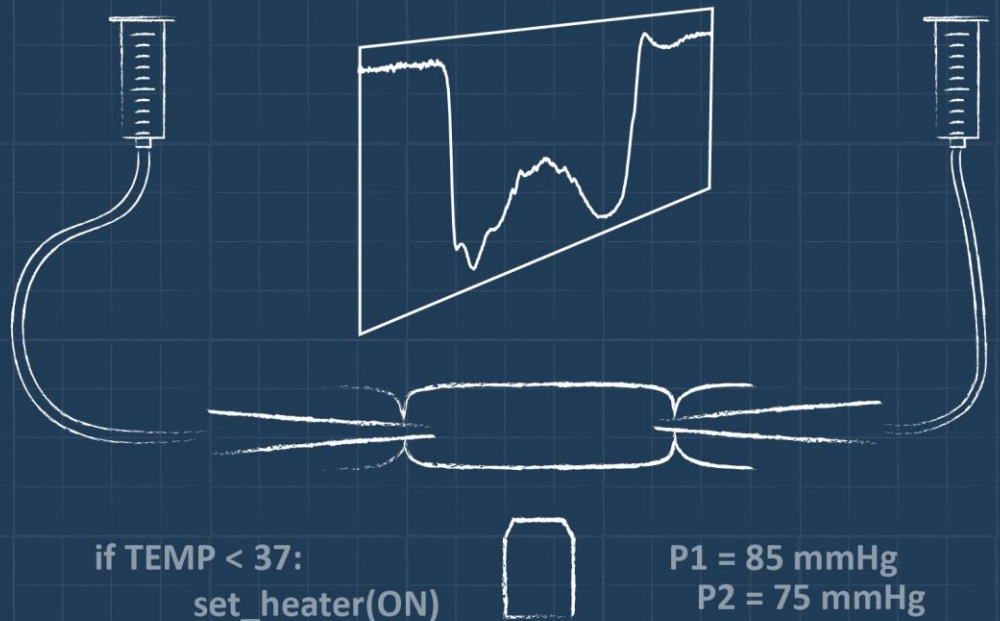


VASOTRACKER 2.0

PRESSURE MYOGRAPHY
BUT BETTER



VasoTracker 2

Temperature & Pressure Monitor User Manual

January 2025

Developed by Calum Wilson & Matthew Lee at the University of Strathclyde, with help from Nathan Tykocki at Michigan State University.

Email: vasotracker@gmail.com



Table of Contents

Table of Contents	0
Overview	1
Parts List	2
Building the VasoTracker Temperature Monitor	3
Option 1: Using the A000077 Protoshield	
Option 2: Using the Sparkfun Protoshield	
Programming the VasoTracker Temperature Monitor	
Building the VasoTracker Pressure Monitor	7
Pressure Monitor Connections and Wiring	
Wheatstone Bridge Shield Connections	
Wheatstone Bridge Shield Setup	
Mapping transducer Connections	
Stacking the Arduino Shields	
Programming the VasoTracker Temperature Monitor	
Integrating the Temperature and Pressure Monitor	15
Setting up the VasoTracker Pressure Monitor	18

Overview



Parts List

VasoTracker Temperature & Pressure Monitor Parts

3D Printed Parts

3D Printed Enclosure	VasoTracker	-	1	£0	£0	GitHub
----------------------	-------------	---	---	----	----	------------------------

Prices correct 10/2024	Total	£0.00
------------------------	-------	-------

VasoTracker Temperature Monitor

Component	Supplier	Product #	Qty	Price	Total	Supplier Link
Arduino Uno**	Arduino	RB-Ard-34	1	£26.50	£26.50	RobotShop
12V power supply	RS	903-7048	1	£8.33	£8.33	RS
Arduino Proto Shield	Skinflint	A000077	1	£13.50	£13.50	SkinFlint
LCD Shield	RobotShop	RB-Cyt-73	1	£3.64	£3.64	RobotShop
10k NTC Thermistor	Adafruit	372	1	£3.06	£3.06	DigiKey
Resistor kit	RobotShop	RES-E12	1	£13.80	£13.80	RobotShop
Jumper wire kit	RobotShop	RB-Cix-02	1	£3.77	£3.77	RobotShop
2-pin Screw Terminal	RobotShop	RB-Dfr-465	1	£0.76	£0.76	RobotShop


Prices correct 10/2024	Total	£73.36
------------------------	-------	--------

VasoTracker Pressure Monitor

Component	Supplier	Product #	Qty	Price	Total	Supplier Link
Arduino Uno**	Arduino	RB-Ard-34	1	£26.50	£26.50	RobotShop
12V power supply	RS	903-7048	1	£8.33	£8.33	RS
Wheatstone bridge amplifier shield	RobotShop	RB-Onl-38	1	£45.63	£45.63	RobotShop
LCD Shield	RobotShop	RB-Cyt-73	1	£3.64	£3.64	RobotShop
Inline pressure transducers	Honeywell	26PCDFG5G	2	£74.55	£149.10	Mouser
12" 4-pin jumper wire	RobotShop	RB-Spa-1107	2	£0.60	£1.20	RobotShop
Arduino stackable headers	RobotShop	RB-Spa-928	1	£1.67	£1.67	RobotShop
Female to female jumper	RobotShop	RB-Gog-65	1	£4.31	£4.31	RobotShop
Heatshrink tubing	RobotShop	RB-Gog-47	1	£1.92	£1.92	RobotShop

Prices correct 10/2024	Total	£242.30
------------------------	-------	---------

Total Price	£315.66
-------------	---------

 **The software and instructions provided only work for Arduino Uno boards with the ATmega microcontroller chip. Other variants (e.g., SAMD51 or ESP-32 chips) cannot be used.**

Building the VasoTracker Temperature Monitor



Option 1: Using the A000077 Protoshield

The temperature monitor uses a simple voltage divider to convert the thermistor resistance (and thus temperature) to voltage

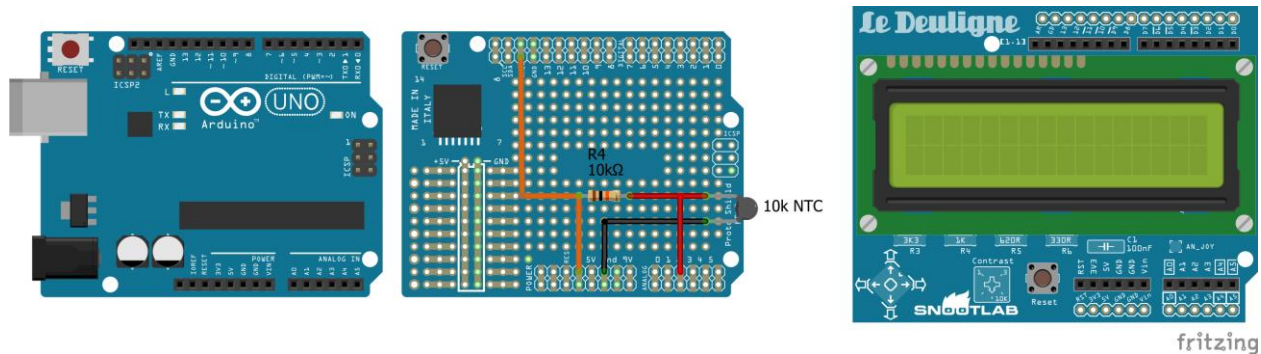
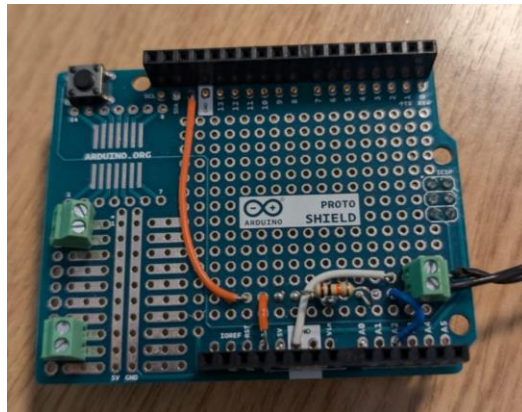


Figure 1 – The VasoTracker 2.0 Temperature Monitor Wiring Diagram

1. Build the temperature sensor circuit on the Arduino Proto shield.

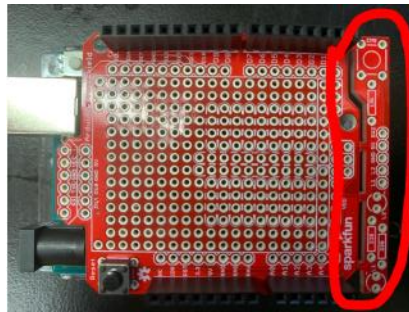


2. Stack the completed Proto shield on top of the Arduino Uno.
3. Stack the LCD shield on top of the Arduino Uno.
4. **(Optional)** 3D print an enclosure to protect your VasoTracker pressure monitor. You can place the device in its own enclosure or use the dual enclosure designed for VasoTracker 2.0. The STL files for the enclosure below can be found on the [VasoTracker 2 GitHub](#).

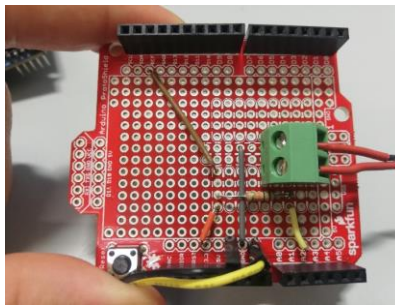
Option 2: Using the Sparkfun Protoshield

If you struggled to source the A000077 Protoshield, you may have purchased the Sparkfun version...

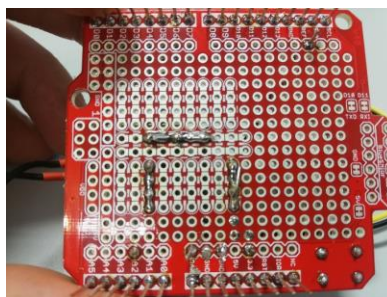
1. Snap off the test part of the sparkfun board It's not needed.



2. Build circuit by soldering the wires/components into place.



3. Ensure wires/components are connected using solder bridges on the bottom side of the protoboard.



Programming the VasoTracker Temperature Monitor

1. Step 1: Download and install the Arduino IDE. This can be done using either Windows or Mac OSX; however, the VasoTracker software installer only runs in Windows.

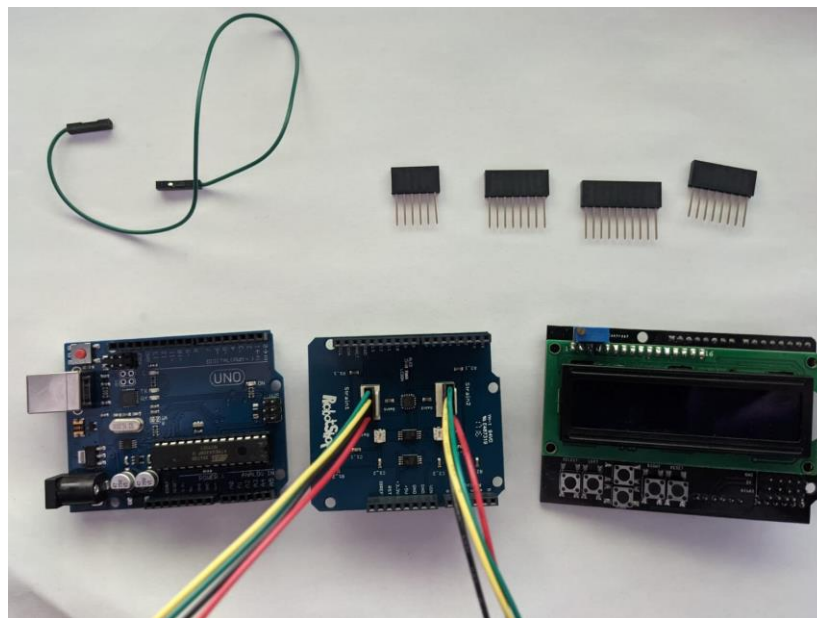


If you want to run VasoTracker on iOS then you must install from the source code. Instructions for doing so can be found in the VasoTracker Software User Manual

2. Download the latest VasoTracker Pressure Monitor Arduino code from GitHub.
3. Unzip the “VasoTracker Temperature_Monitor_VT2” folder. The folder name must not be changed.
4. Connect the Arduino to your computer using the Arduino USB cable.
5. Using the Arduino IDE software, open “VasoTracker_Temperature_Monitor_VT2.ino”.
6. Configure the Arduino IDE for your Arduino Uno.
7. Tools → Board → Arduino Uno
8. Tools → Port → COM X (Arduino Uno). “X” will vary by computer and system, depending on how many USB peripherals are attached.
9. Step 7: Upload the code to the Arduino by clicking the upload button. No modifications to the code are required if you are doing a standard build as outlined above.



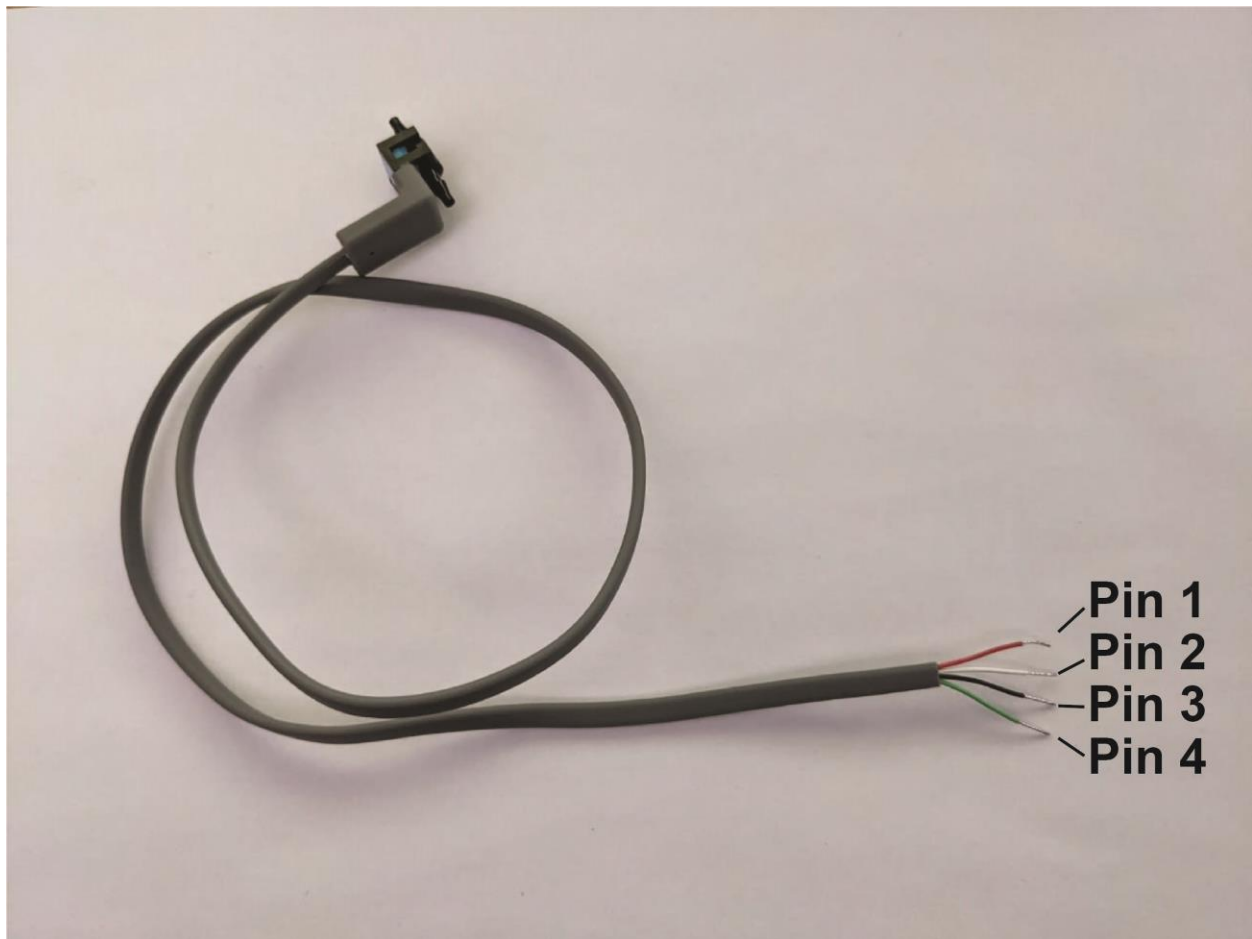
Building the VasoTracker Pressure Monitor



Pressure Monitor Connections and Wiring

The 26PCDFG5G pressure transducer has four pins. This version of the sensor comes with a handy cable. There are four wires:

- **Pin 1 (Red wire):** Vs
- **Pin 2 (White wire):** Signal +
- **Pin 3 (Black wire):** Ground
- **Pin 4 (Green wire):** Signal -



Wheatstone Bridge Shield Connections



The following instructions are for use with the supplied Honeywell pressure transducer cable. There is a crossover within the cable header. If you make your own cable then the Pin order is different

The Wheatstone shield uses the following Arduino pins:

- **A0: Sensor 1 input**
- **A1: Sensor 2 input**
- **5V, 3.3V, GND: to power the shield**

Each pressure sensor needs to be connected to one of the 4-pin Molex connectors on the Wheatstone bridge shield. The Molex pins are as follow:

- **Molex Pin 1: Vs**
- **Molex Pin 2: Signal +**
- **Molex Pin 3: Signal -**
- **Molex Pin 4: Ground**

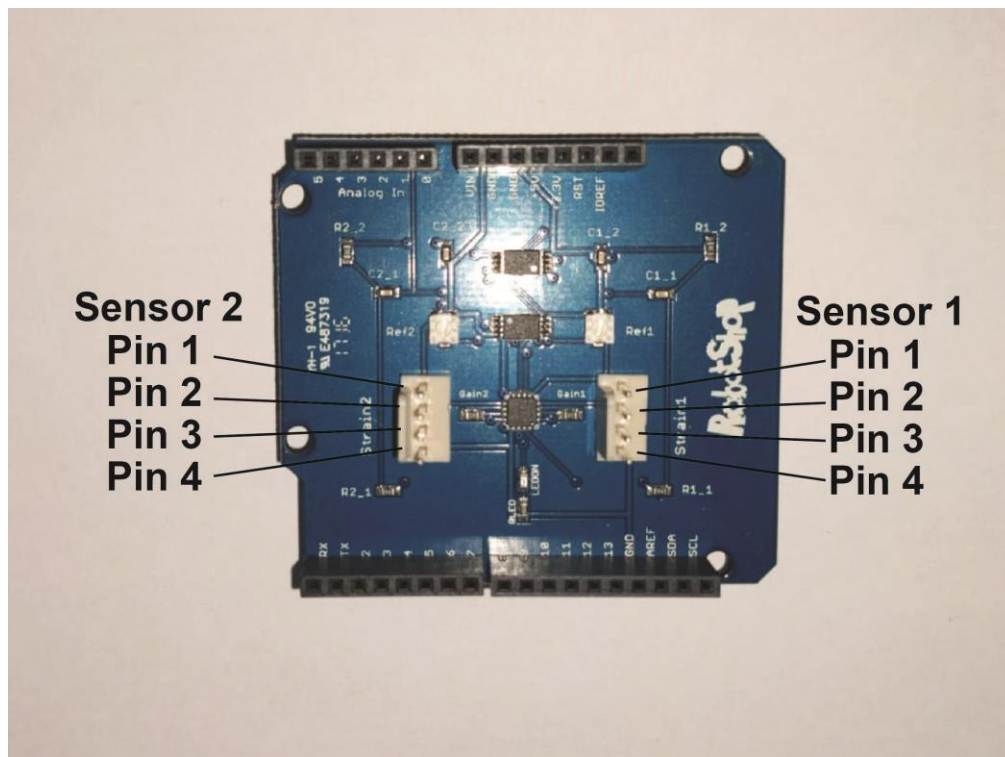


Figure 2 – Wheatstone bridge pinout.

Wheatstone Bridge Shield Setup



Each channel on the Wheatstone bridge shield has its own reference voltage that is set using specific channel potentiometers. Calibrating these is necessary to ensure that the shield works as intended.

- Documentation for the Wheatstone Bridge Shield can be found on the RobotShop website.
- Users need to perform the calibration procedure detailed in the “Test Procedure” pdf file (found here).
- If you are too lazy to read this, then please turn both potentiometers fully clockwise, then turn back 1/4 - 1/8 of a turn so that the screw is parallel to the Arduino headers.
- <https://www.robotshop.com/en/strain-gauge-load-cell-amplifier-shield-2ch.html>
- The Wheatstone bridge shield contains two potentiometers

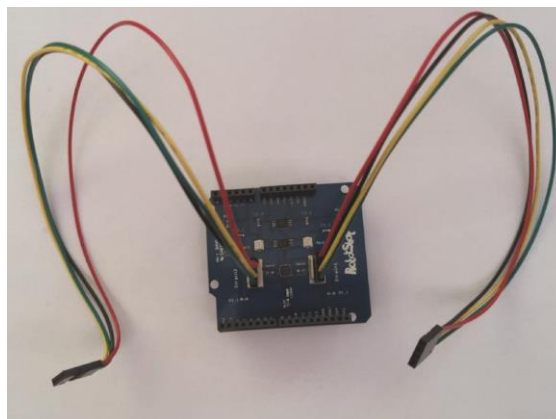
Mapping transducer Connections

The table below maps the pressure transducer connections to the Molex connector pins on the Wheatstone bridge shield:

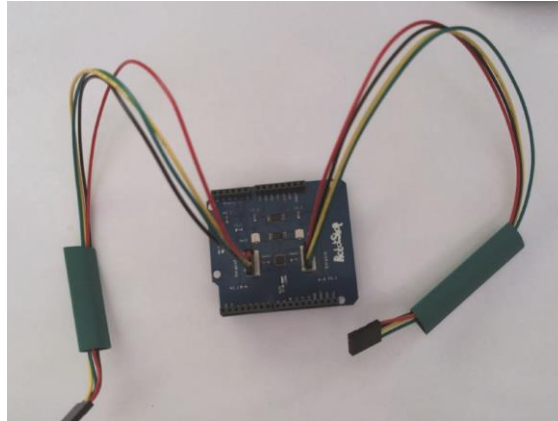
Connection	Pressure sensor pin	Wheatstone bridge pin
Vss	Pin 1 (Red wire)	Molex Pin 1
Signal +	Pin 2 (White wire)	Molex Pin 2
Ground	Pin 3 (Black wire)	Molex Pin 4
Signal -	Pin 4 (Green wire)	Molex Pin 3

Table 1 - Mapping of pressure transducer connections to Wheatstone bridge shield

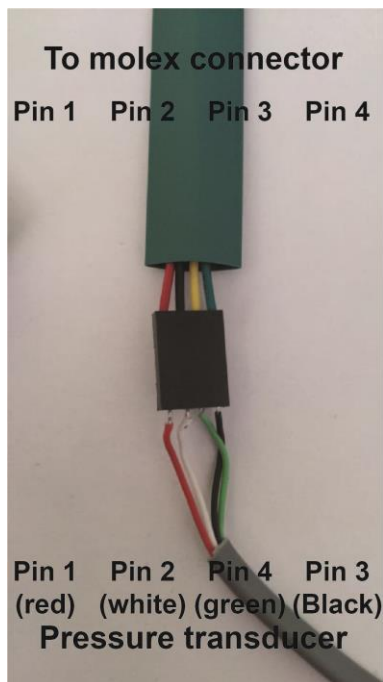
1. **Connect the 4-pin jumper wires to the Molex connectors on the Wheatstone bridge.**



2. Slide a piece of heatshrink tubing over the jumper wires

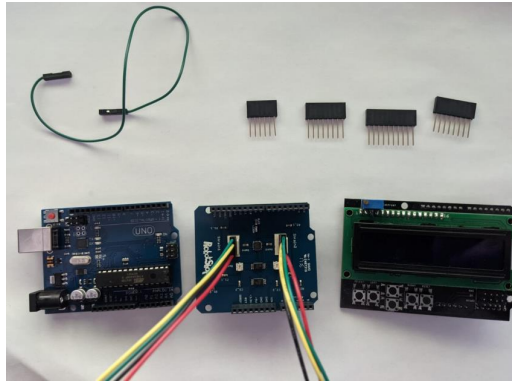


3. Connect the pressure transducer cable to the jumper wire as per Table 1.
4. Slide a piece of heat shrink tubing over the jumper wires. Heat gently (with heat gun or soldering iron) to shrink. Be careful not to burn the heat shrink tubing or damage the wiring insulation.

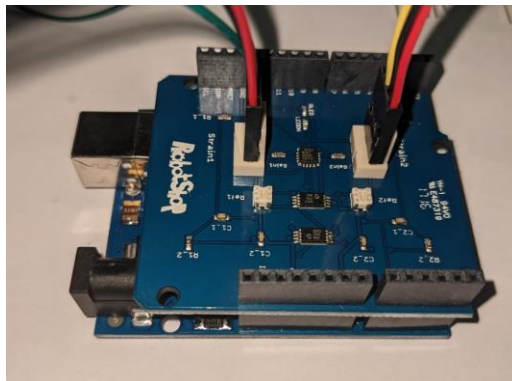


Stacking the Arduino Shields

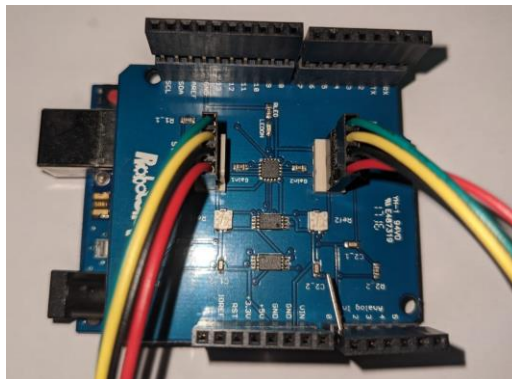
1. Gather the Arduino, shields, and stackable headers.



2. Stack the wheatstone bridge shield on top of the Arduino Uno.

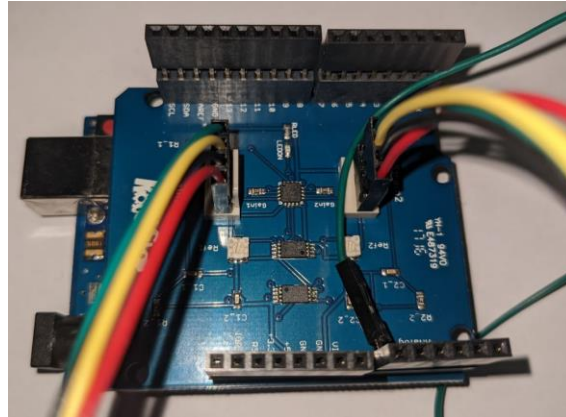


3. Place stackable headers on top of the Wheatstone Amplifier Shield. These provide clearance for the pressure transducer connections). Bend the A0 pin on the stackable header so that it does not connect to the Arduino Uno board (see Figure 8).

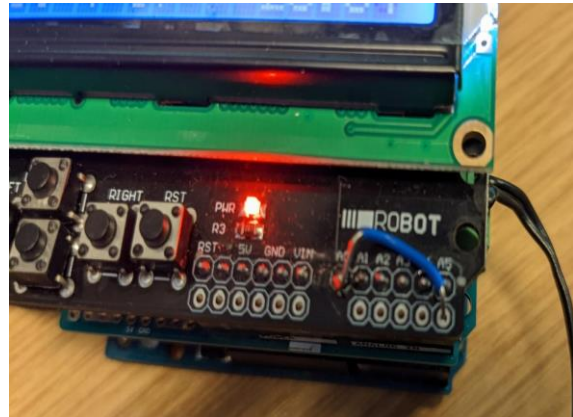
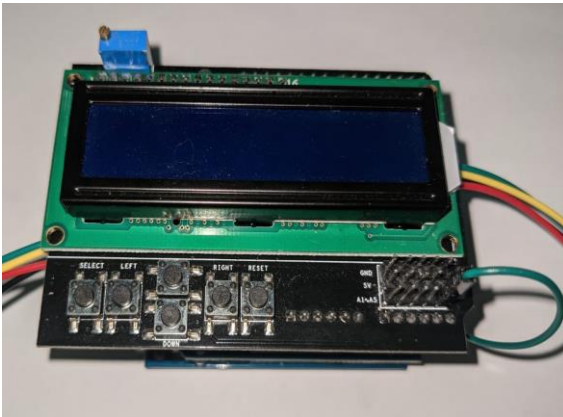


The LCD shield normally uses the A0 pin for the button commands. However, the strain gauge also uses the A0 (and A1) inputs on the Arduino. To get around this problem, we route the LCD A0 pin to A5.

4. Connect the jumper cable to the bent A0 pin.



5. Stack the LCD shield on top of the headers. Route the jumper cable to the A5 pin.



Alternative method for rerouting LCD pins: Remove the A0 pin from the LCD shield (heat it up with a soldering iron and pull it out using pliers). Solder a jumper wire from A0 to A5 on the board. This is a better way to do the above.

6. **(Optional) 3D print an enclosure to protect your VasoTracker pressure monitor.** STL files for a single enclosure can be found on the [VasoTracker 2 GitHub](#). Instructions for connecting the Temperature and Pressure Monitors in a single enclosure are available [here](#).



Programming the VasoTracker Temperature Monitor

1. **Download and install the Arduino IDE.** This can be done using either Windows or Mac OSX; however, the VasoTracker software installer only runs in Windows.



If you want to run VasoTracker on iOS then you must install from the source code. Instructions for doing so can be found in the VasoTracker Software User Manual

2. **Download the latest VasoTracker Pressure Monitor Arduino code from GitHub.**
3. **Unzip the “VasoTracker_Pressure_Monitor_VT2” folder.** You must keep all the files in this folder together, and the folder name must not be changed.
4. **Connect the Arduino to your computer using the Arduino USB cable.**
5. **Within the “VasoTracker_Pressure_Monitor_VT2” folder, you will find three files: (1) “pressure.cpp”; (2) “pressure.h”; and (3) “VasoTracker_Pressure_Monitor_VT2.ino”.** Using the Arduino IDE software, open “VasoTracker_Pressure_Monitor_VT2.ino”.
6. **Configure the Arduino IDE for your Arduino Uno.** If you purchased an Arduino Uno clone board (e.g., Adafruit Metro, please see the instructions for configuring the board at the supplier’s website).
 - Tools → Board → Arduino Uno
 - Tools → Port → COM X (Arduino Uno). *“X” will vary by computer and system, depending on how many USB peripherals are attached.*
7. **Upload the code to the Arduino by clicking the upload button.** No modifications to the code are required if you are doing a standard build as outlined above.

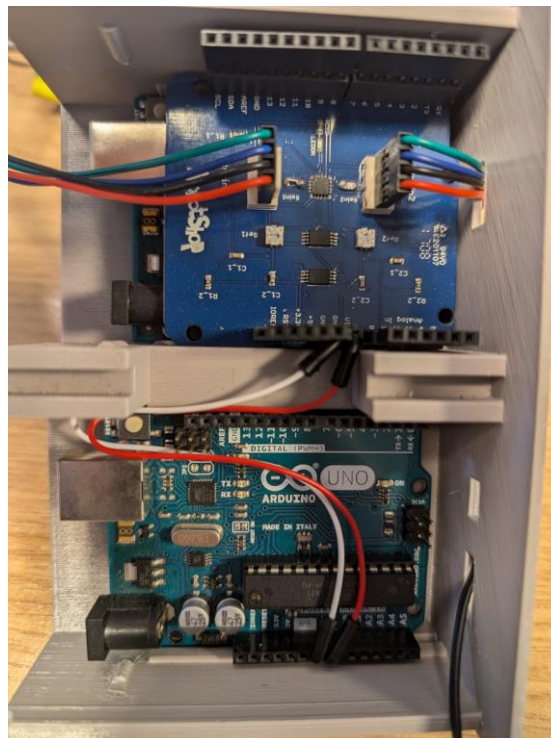


Integrating the Temperature and Pressure Monitor



The VasoTracker Temperature and Pressure Monitors can either be held as independent units, or can be enclosed in a single enclosure and powered from a single power supply.

1. **Download the VasoTracker 2 CombiBox and 3D print all of the parts (including two sets of buttons).**
2. **Slide the Arduino from the Temperature Monitor into the lower slot in the enclosure.**
3. **Slide the Arduino + Wheatstone Shield from the Pressure Monitor into the upper slot in the enclosure.**
4. **Use jumper wires to connect the VIN and GND of the Wheatstone shield to that on the Temperature Monitor Arduino.** You may need to remove two pins from the stackable headers to achieve this.



5. **Add the remaining shields to the Arduino's (temperature monitor protoshield circuit, and LCDs). Use stackable headers to raise the height of the Pressure Monitor LCD.**
6. **Add the front plate and the buttons.**
7. **Power via the pressure monitor Arduino.**



Setting up the VasoTracker Pressure Monitor



On first use, the VasoTracker pressure monitor needs to be calibrated. Once this has been done, the calibration values are saved into memory on the Arduino Uno and can be reloaded each time the pressure monitor is used – even after disconnecting the power. We recommend that you recalibrate the sensors once a week. We use a Big Ben manometer, but any calibrated manometer will suffice, such as a blood pressure cuff (with the cuff removed).



1. Use your tubing and connectors of choice to connect the first transducer to the manometer.
2. On the recalibrate screen (Figure 15), use the up/down buttons until the screen reads “YES” and press the “ENTER” button (left most on LCD screen).



3. The next screen is used to set the low pressure used for calibration (Figure 16). Use the UP/DOWN (10 mmHg increments) or left/right buttons (1 mmHg increments) to set the low calibration pressure to 0 mmHg. Save this value by pressing “ENTER”.



4. The next screen shows the reading from the pressure sensor (Figure 17). Ensure the manometer is set to the pressure set in the previous step (0 mmHg). Press “ENTER” to save this value.



5. The next screen is used to set the high pressure used for calibration (Figure 18). Use the UP/DOWN (10 mmHg increments) or left/right buttons (1 mmHg increments) to set the high calibration pressure to 160 mmHg. Save this value by pressing “ENTER”.



6. The next screen shows the reading from the pressure sensor (Figure 19). Ensure the manometer is set to the pressure set in the previous step (160 mmHg). Press “ENTER” to save this value.



7. Repeat Steps 3-6 for the second pressure transducer.
8. Connect the pressure sensors to your pressure myograph system and use with VasoTracker. If used with VasoTracker pressure myograph software, the pressure values will be saved along with blood vessel diameter data.





www.vasotracker.com