
OpenSpinCoater

Release 1.0.0

Apr 29, 2020

Contents:

1	Overview	3
2	Hardware	11
2.1	Microcontroller	11
2.2	LCD	11
2.3	Motor	12
2.4	Power Supply	12
2.5	Switching Transistor(s)	12
3	Software	13
4	Indices and tables	15

OpenSpinCoater is a design for a simple [spin coating machine](#), built using inexpensive parts. The microcontroller is an Arduino Uno, and the motor is a 12 volt, 4-pin computer fan. It should be able to be constructed for around the AUD\$50 mark, well below that of [commercial devices](#).

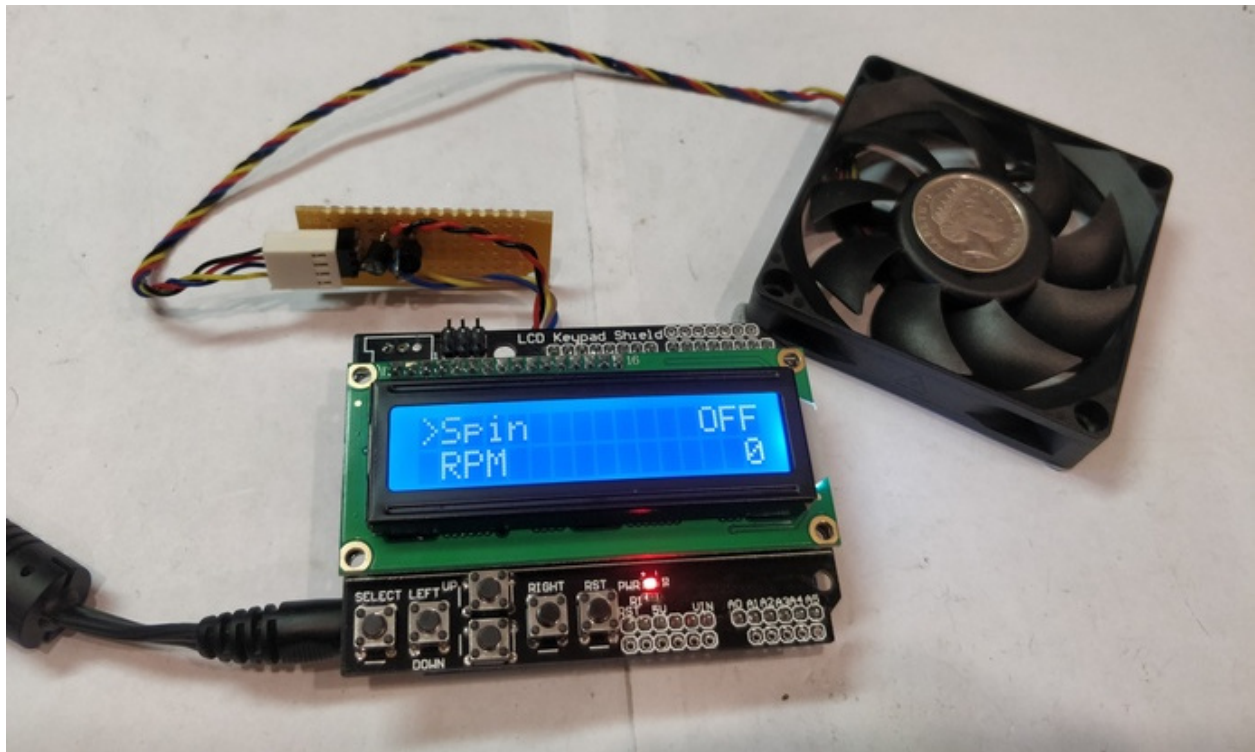
Source code: <https://gitlab.com/ptapping/openspincoater>

Documentation: <https://openspincoater.readthedocs.io/>

CHAPTER 1

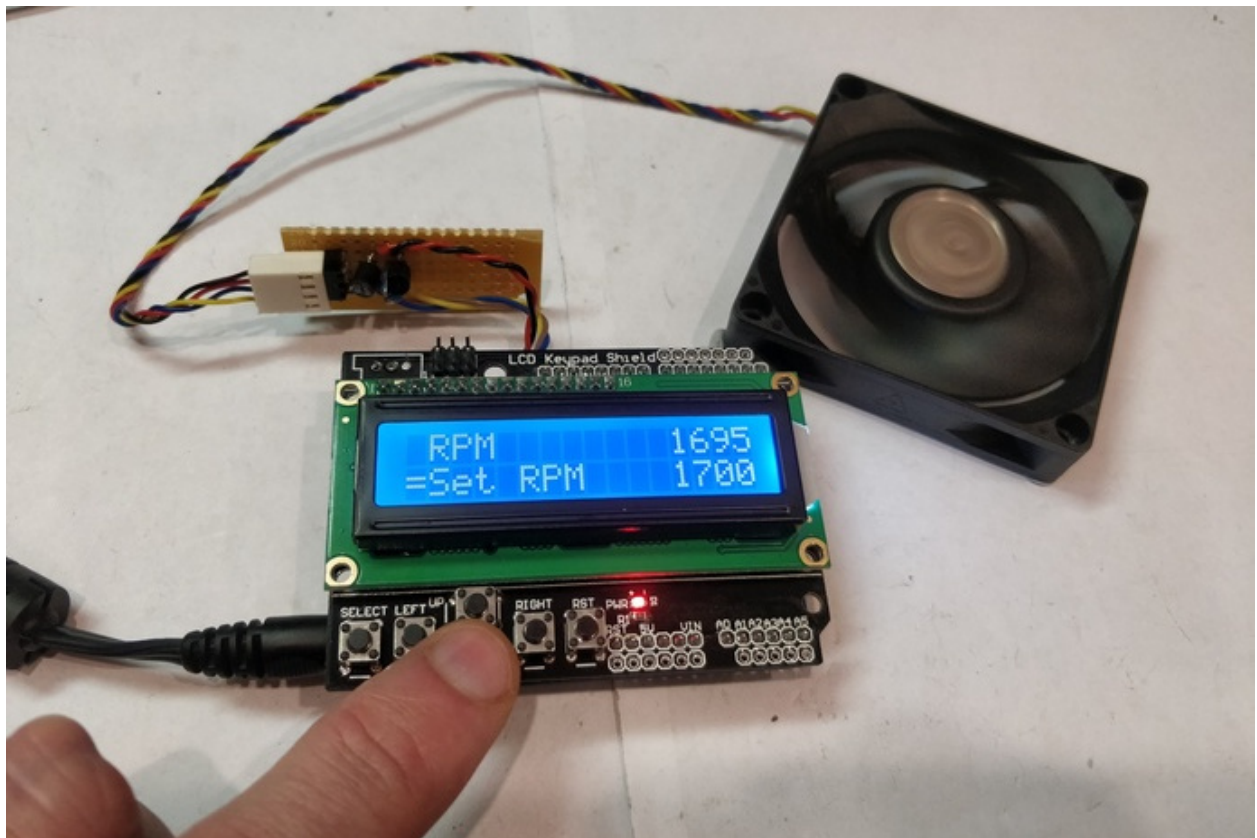
Overview

Here's some photographs of the prototype in operation. Selecting the `Spin` menu item starts or stops the fan rotation.



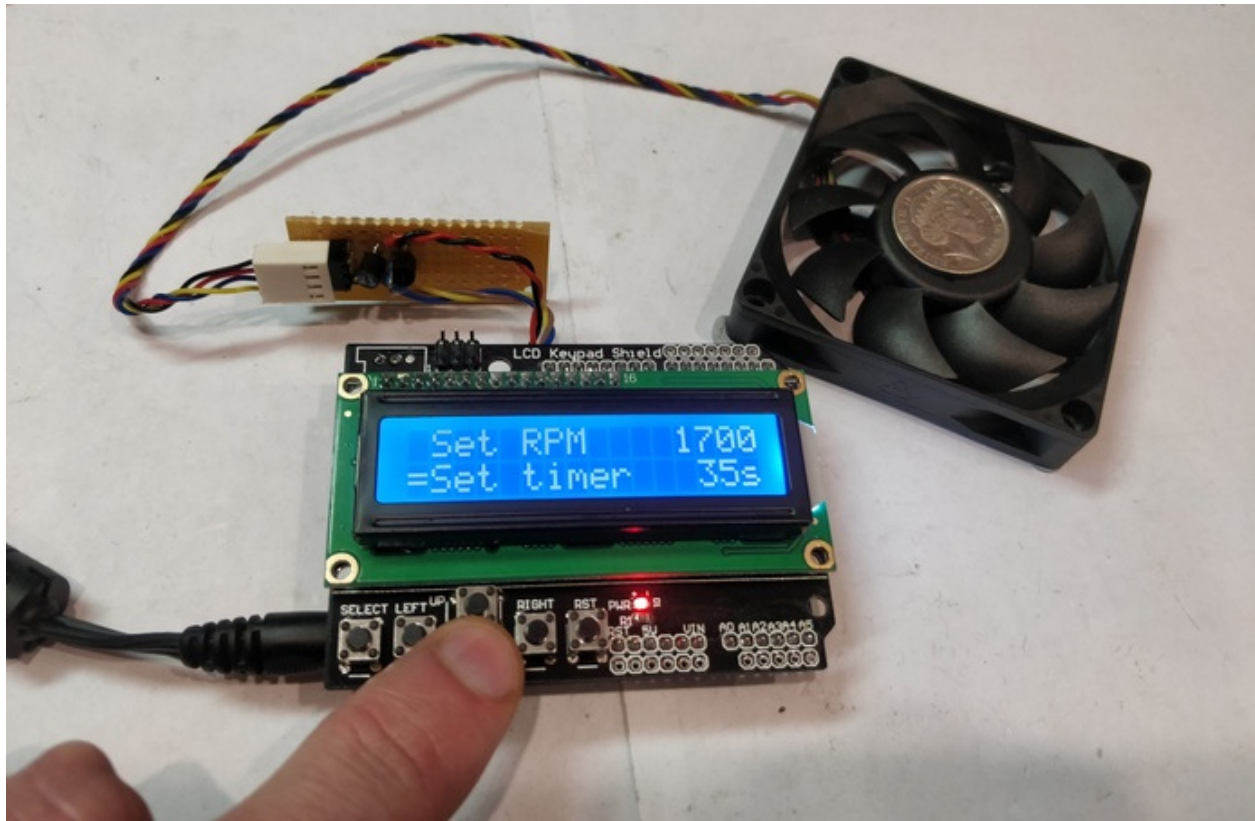


The RPM line displays the currently measured rotation speed, in revolutions per minute. Selecting Set RPM allows the desired rotation speed to be adjusted with the up and down buttons.

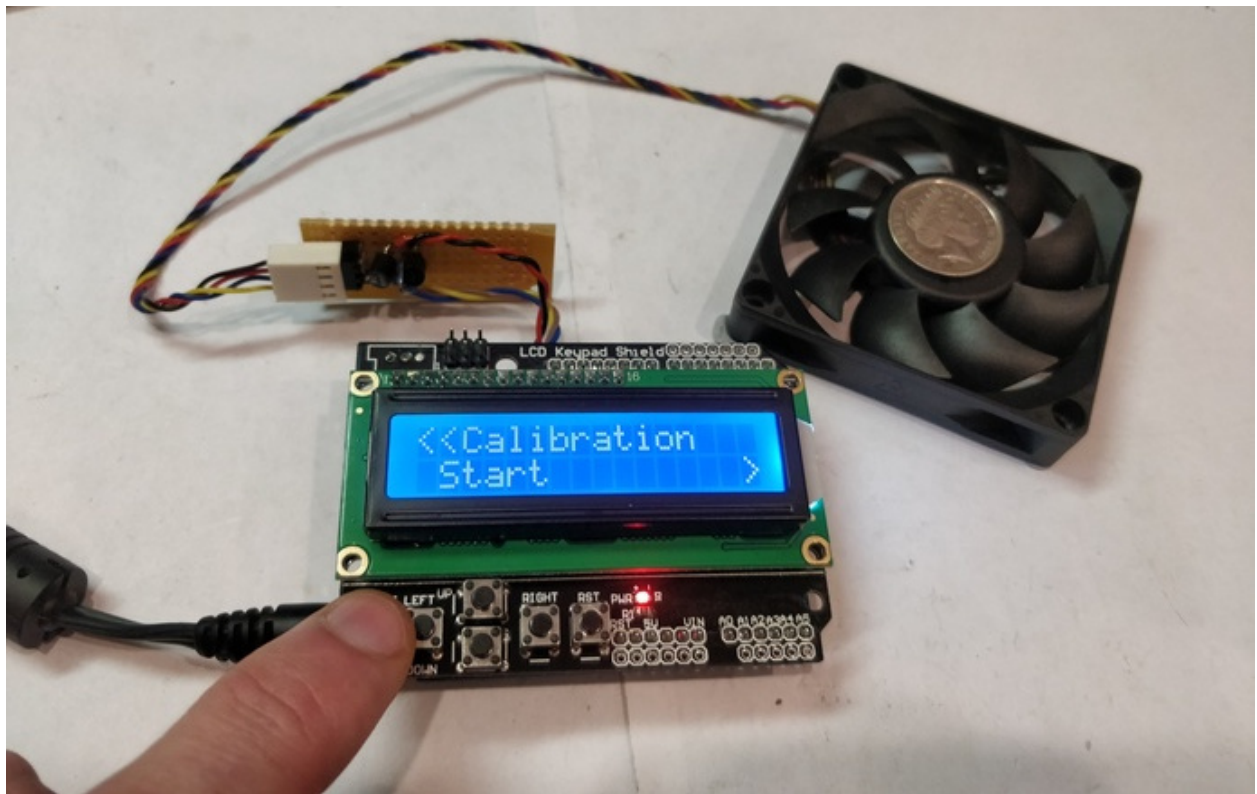
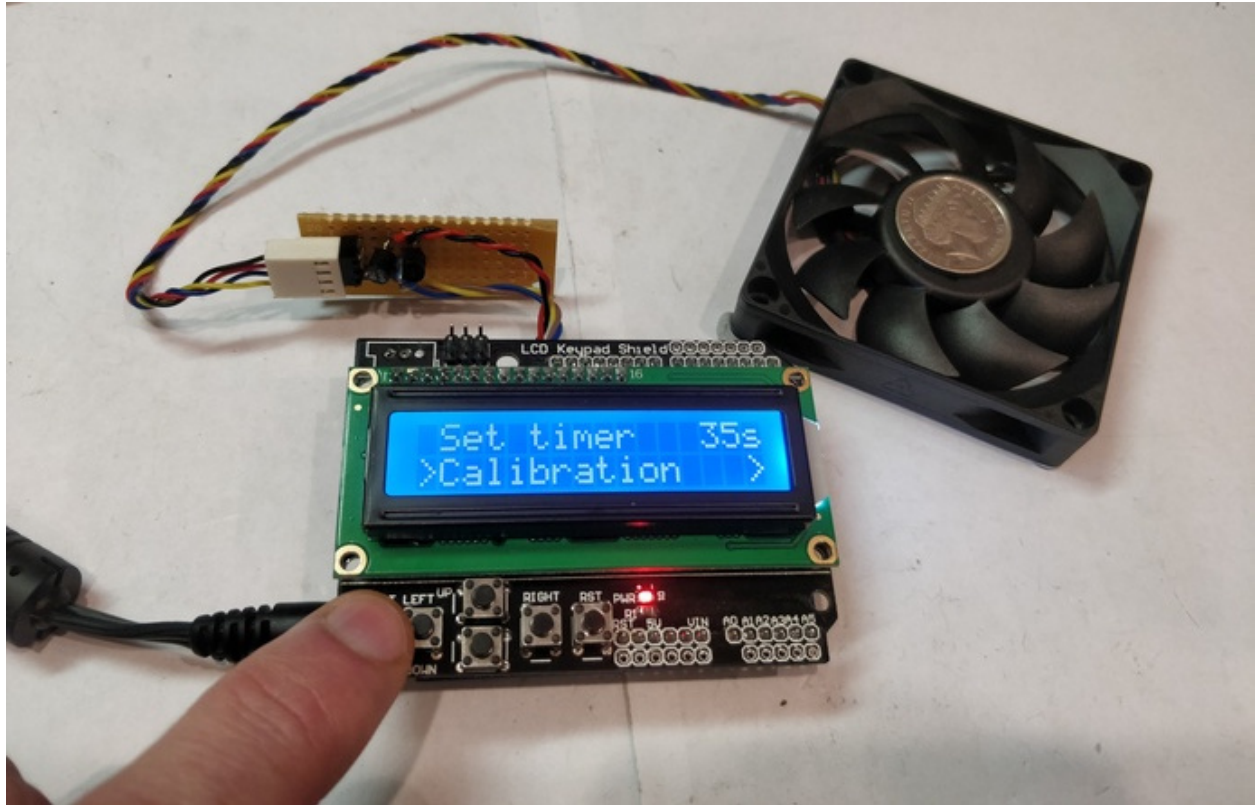


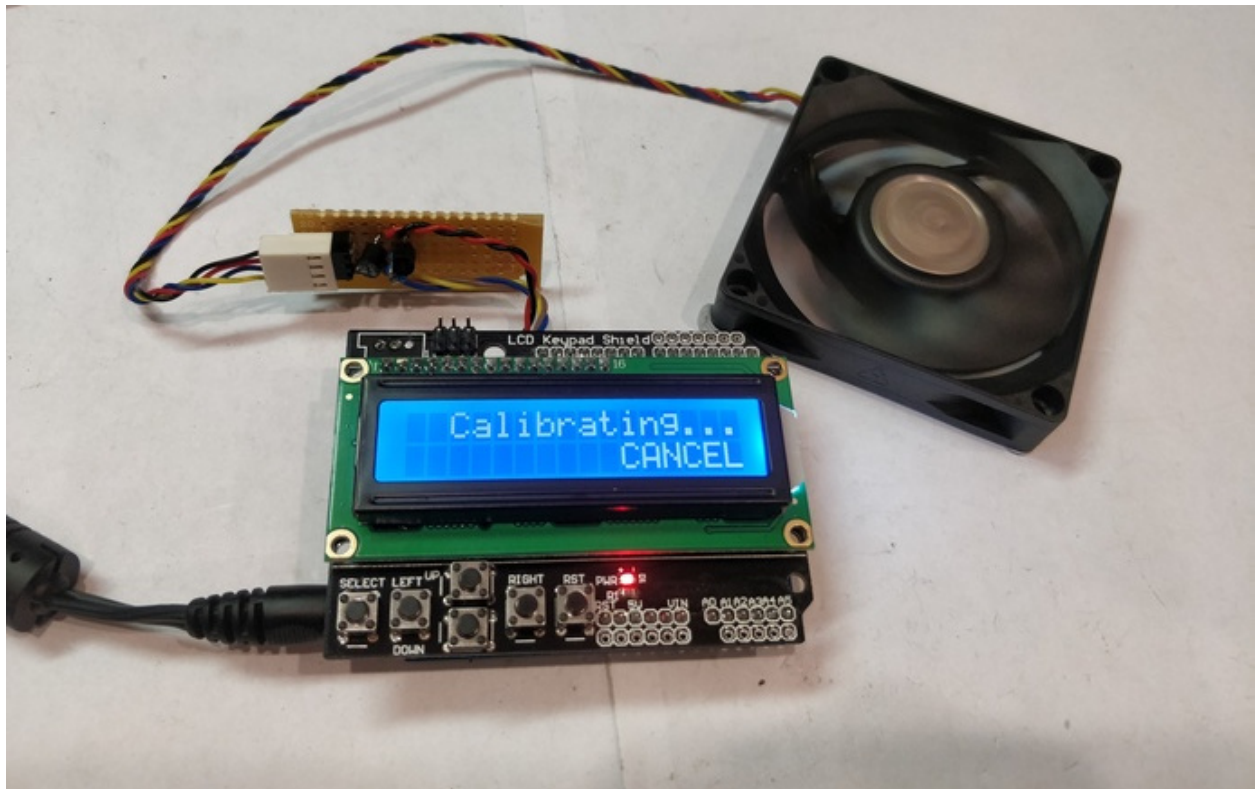
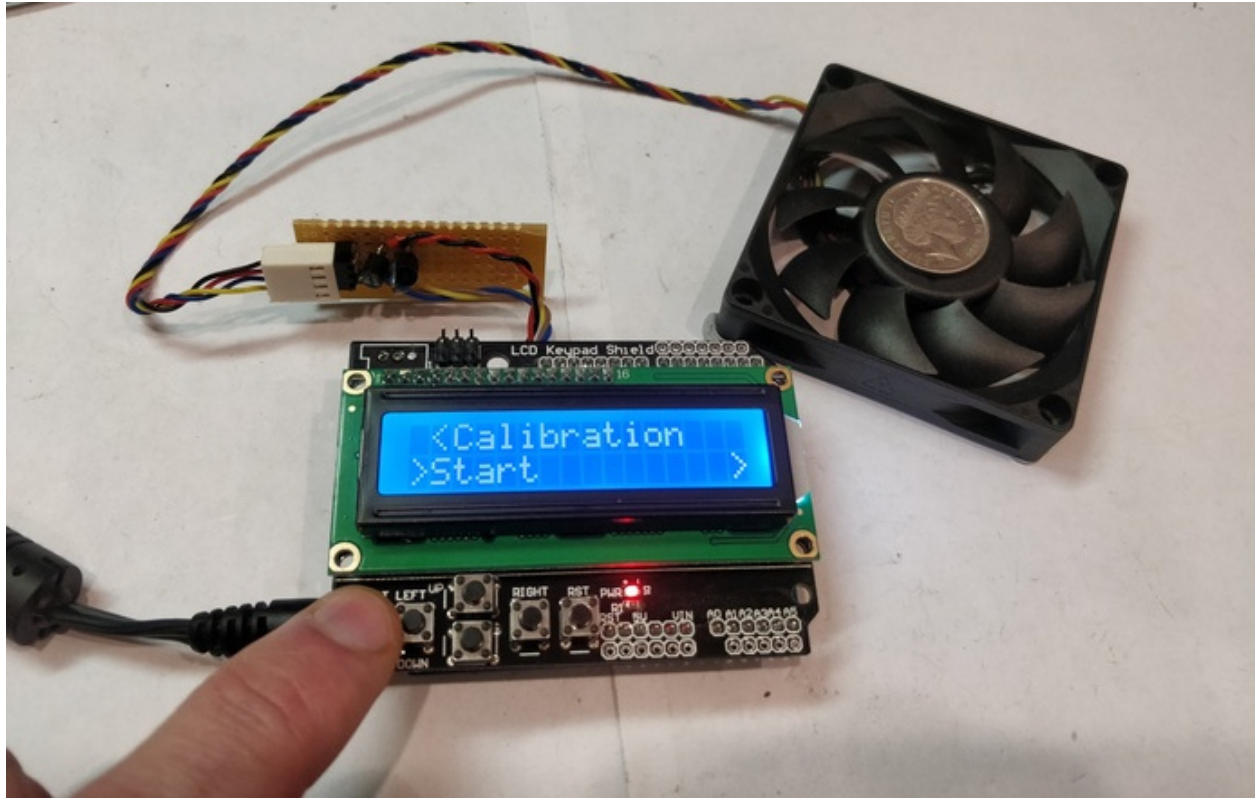
There is an optional timer which will automatically stop the spinning after a set duration. Set the timer to 0 to spin

indefinitely.

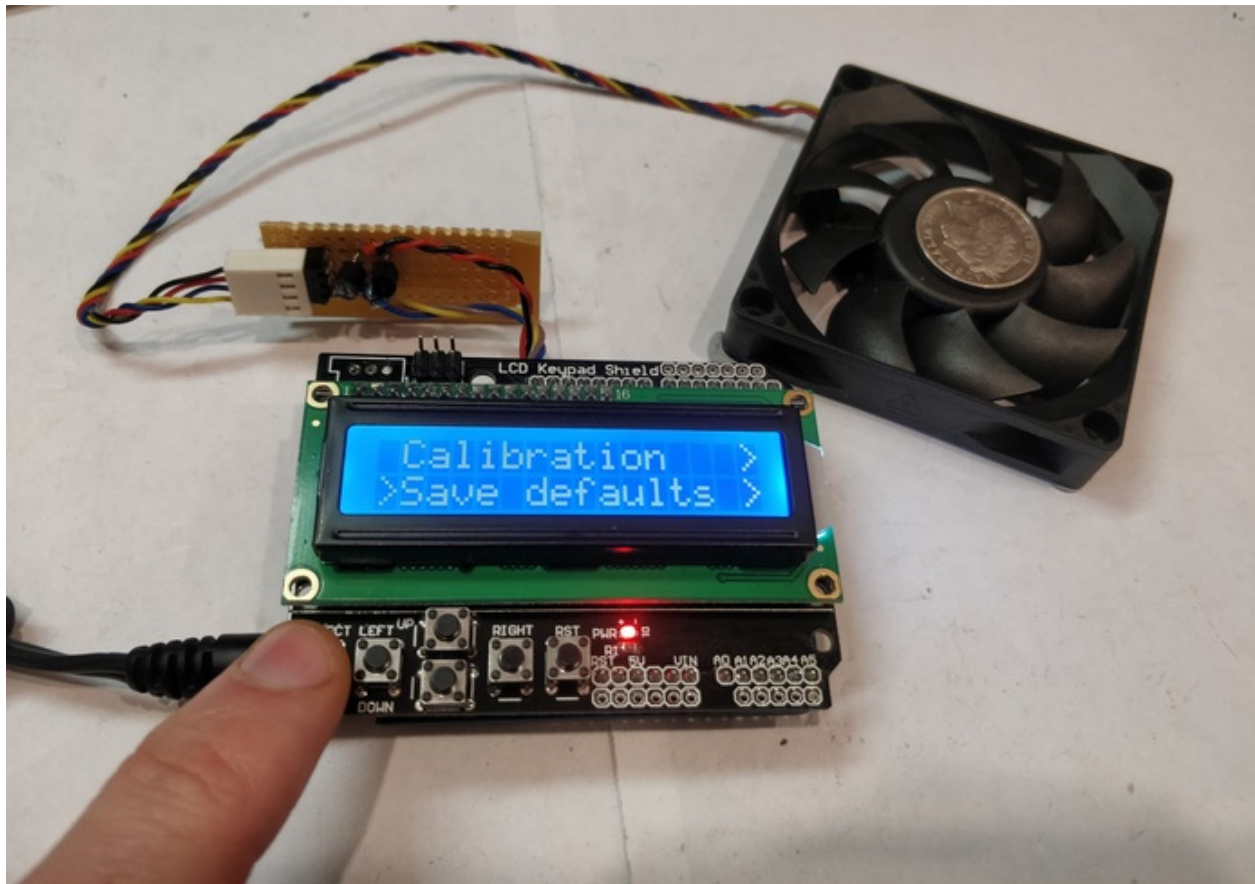


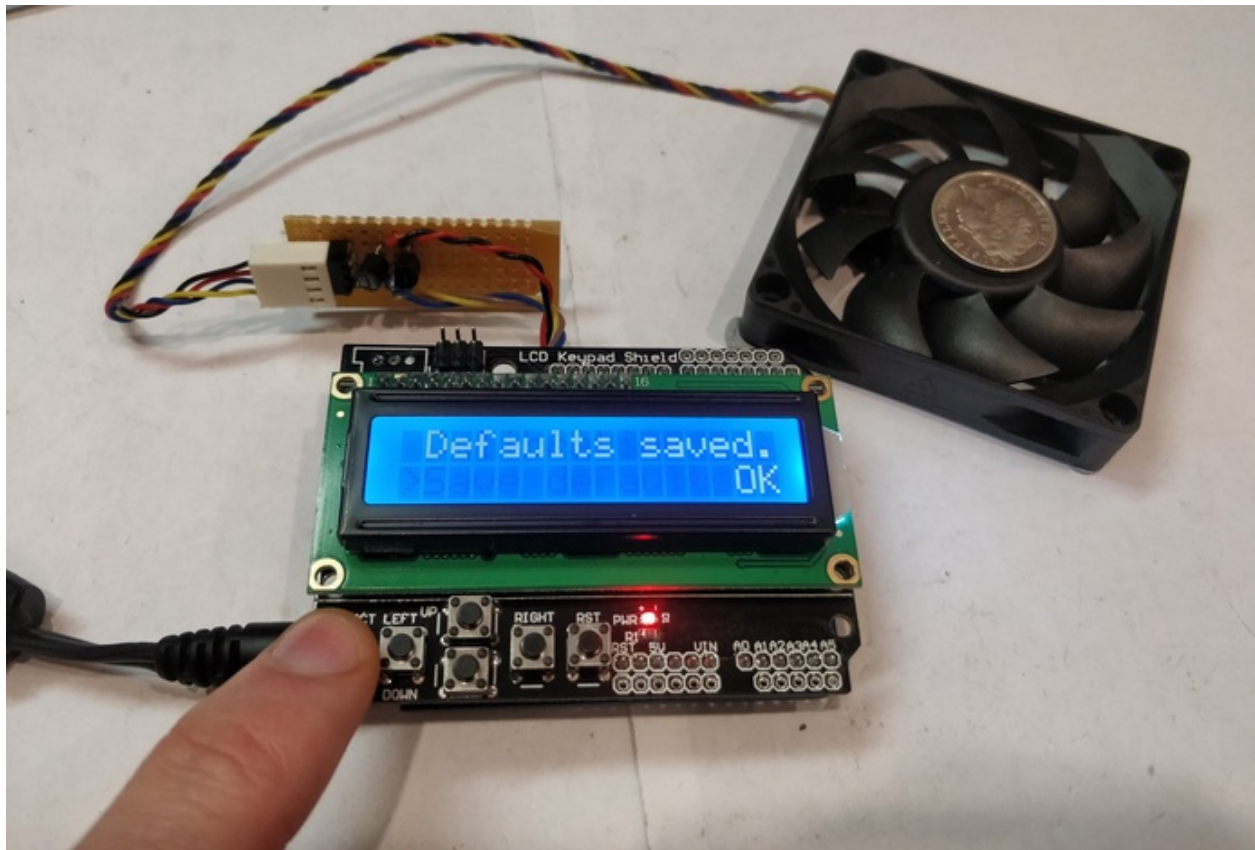
The target RPM speed is achieved using a calibration table. Changing the power supply voltage, fan type, load etc will require the calibration table to be re-built. If the measured RPM is not matching the set RPM, run the calibration procedure. It should ramp up the spin speed, and complete in 1–2 minutes.





The currently set RPM and timer values can be saved to the device to use as defaults for initial power on. Note that the calibration table is automatically saved on completion.





The hardware consists of

- Microcontroller, [Arduino Uno](#) or compatible.
- Display, a 2x16 LCD shield with control buttons, AKA the [dfRobot shield](#).
- Motor, a 4-pin (PWM-controlled) computer fan.
- Suitable [power supply for the Arduino board](#).
- Switching transistor(s), fan connector.

2.1 Microcontroller

The Arduino Uno is inexpensive, and the Chinese “compatible” clones work fine. The pin connections are:

- 2: Fan RPM sense. To pin 3 on fan connector.
- 3: Start. To input of switching transistor(s).
- 11: PWM out. To pin 4 on fan connector.

Additionally, wires are soldered to the negative and positive terminals of the Arduino’s DC barrel jack for ground, and to supply 12 V to the switching transistor(s), respectively.

2.2 LCD

The “dfRobot” shield integrates a 2-row, 16-column LCD with control buttons connected via resistor matrix to the Arduino A0 analog input. Only the up, down, and select buttons are used.

2.3 Motor

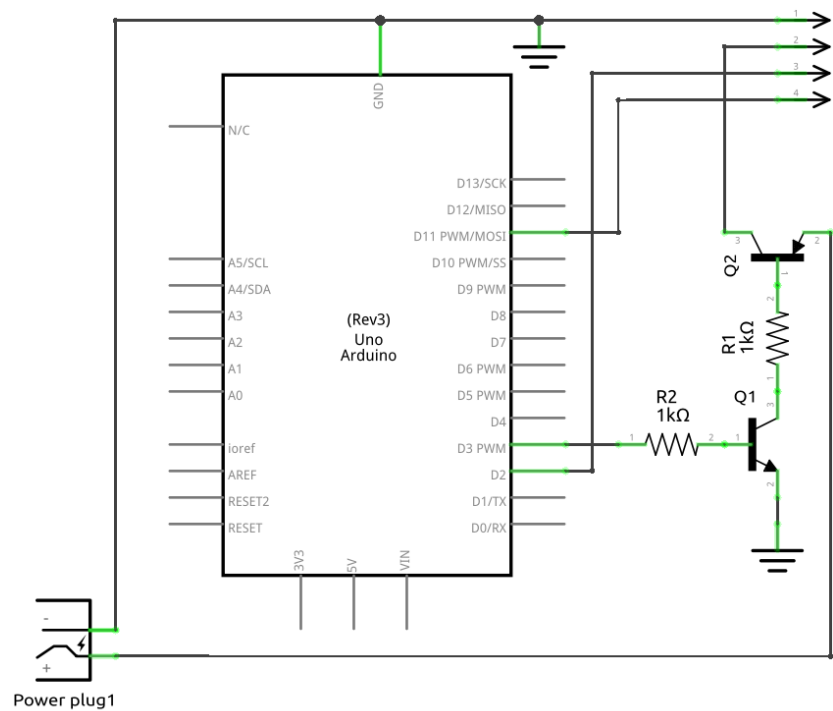
Computer fans are cheap and easily available, and can probably be salvaged from discarded computers. The prototype uses a 60 mm fan from an old AMD heatsink. Ensure the fan is a 4-pin type, which uses a pulse-width-modulated signal to control rotation speed.

2.4 Power Supply

Almost any DC plugpack with 2.1 mm barrel jack will work. Voltage should be 9 or 12 volts, which will affect the RPM range of the fan used. If the RPM range is too high using a 12 V adaptor, change it over to a 9 V one.

2.5 Switching Transistor(s)

The 12 V to the fan needs to be switched using the 5 V output from the Arduino. For the prototype, an NPN+PNP pair of BJTs were used, as they were in my parts box. A single n-channel FET would work and reduce the part count. Ensure the transistor(s) selected can handle the current draw from the fan, approximately 1 A. The schematic for the BJT pair is as follows:



fritzing

CHAPTER 3

Software

The code is written for use on an Arduino Uno (or compatible) microcontroller, but should run on other boards subject to modifications. It uses the [IOAbstraction](#), [tcMenu](#) and [LiquidCrystalIO](#) libraries.

Compile and upload using the Arduino software in the usual manner.

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`