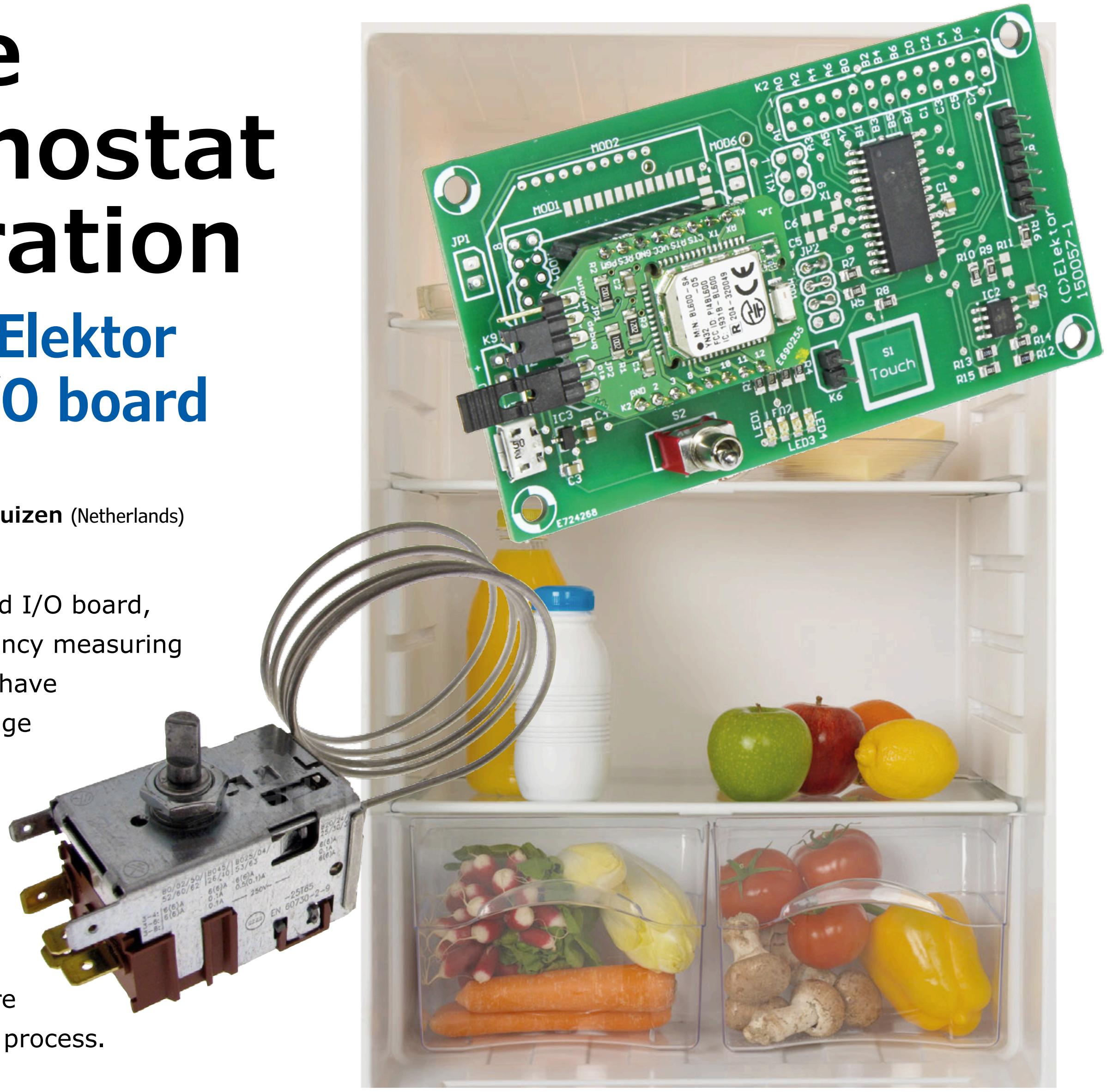
Fridge Thermostat Calibration

Using the Elektor Android I/O board

By **Elbert Jan van Veldhuizen** (Netherlands)



If you have an Android I/O board, you don't need any fancy measuring equipment when you have to calibrate a new fridge thermostat. The only things you need to add are a resistor and an NTC. An app on an Android mobile phone accurately tracks the temperature during the calibration process.

Fridges have the unwelcome tendency to go wrong at times. Fortunately, the fault is usually not caused by the expensive compressor. More often than not, it's the cheaper thermostat that's broken. Replacing a thermostat will be a piece of cake for the typical Elektor reader. However, it will be more difficult to find exactly the same type of thermostat. The author found that the specific thermostat required for the fridge would cost five times as much as a universal thermostat. Furthermore, the supplier couldn't guarantee that it was an exact replacement for that type of fridge. The author decided to buy a cheap universal thermostat and calibrated it using the Android I/O board and a homemade app. It wasn't really worth the effort to build a separate device for a job that only needs doing once every five to ten years. But if you have a spare Android I/O board lying about, it won't take long to make a fridge thermostat calibrator!

Thermostat operation

Before we look in detail at the fridge thermostat calibrator, we first have a look at how a fridge thermostat works and what the differences are between the various types.

A fridge thermostat has a capillary tube in which a liquid evaporates (often the same type of liquid that is used in the fridge for cooling). When it warms up, more gas is created. This increases the pressure, which forces a diaphragm to move, closing a switch that turns on the compressor. When it becomes cooler, the gas condenses, which causes the diaphragm to retract, turning off the switch.

The electrical circuit of the fridge is shown in **Figure 1**. This circuit also shows a heating element used for automatic defrosting, although this may not be present in all fridges. When the compressor is switched on, the heating element is shorted and will therefore be off. When the compressor is turned off, a small current flows through the resistor in the thermostat and through the heating element, all via the motor of the compressor. However, this current is too small to make the motor turn. The heating element warms up the cooling element of the fridge to above freezing, which causes any ice on it to melt and to drain away through a small hole in the fridge.

Different types of thermostat will have different values of resistors. However, you can easily reuse the resistor from the old thermostat.

The thermostat measures the temperature of the cooling element (rather than

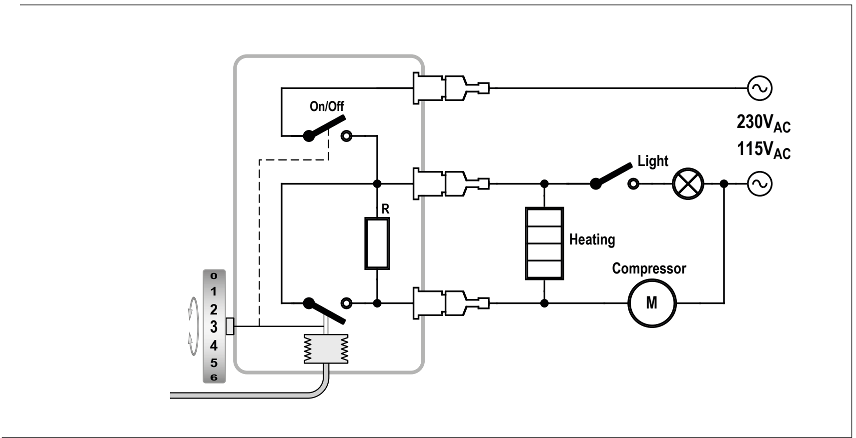


Figure 1. Circuit diagram for the thermostat and the wiring in the fridge.

Figure 2. The external (left) and internal (right) adjustment screws on the thermostat.

the temperature in the fridge!). In a combination fridge the cooling element cools down to a temperature between −15 and −25 degrees Celsius (turn-off point). The thermostat then switches off and lets the temperature rise all the way to above the freezing point, which starts the automatic defrosting (turn-on point). The compressor will then be turned on again. The temperature control on the thermostat determines the turn-off point. The screen dump in **Figure 4** shows the graph for a complete cycle.

The measuring circuit is extremely simple

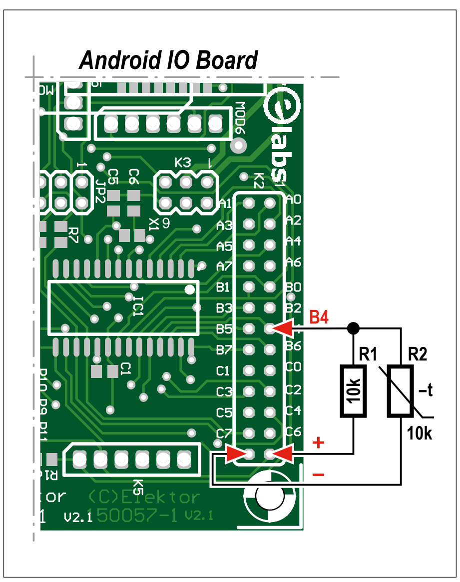


Figure 3. The 'measuring circuit' consists of a resistor and an NTC.

Table 1. Effect of the adjustment screws on the switching temperature.

	turn-off temperature	turn-on temperature
1 x CW external	+3 °C	+2 °C
1 x CW internal	−2 °C	+2 °C

There are two screws on the back of the thermostat (**Figure 2**). One screw is fixed to the casing (from here on called the external screw) and one screw is inside the thermostat (the internal screw). The adjustment of both these screws determines the turn-on and turn-off temperatures.

To start with, the turn-off temperature has to be low enough to make it cold enough inside the fridge. But if the turn-off temperature is set too low, the fridge will never be able to reach this temperature and the compressor will remain turned on, causing the freezer compartment to freeze solid. The turn-on temperature should be about 4.5 degrees Celsius for most fridges.
In **Table 1** you can see what effect the screws have in the thermostat of the author, to get an idea what to expect. CW means clockwise, 1 turn to the right and '+' indicates that the switch point is at a higher temperature.

Calibration

To calibrate the thermostat you have to set the temperature control to its maximum setting (usually on '6'). Turn the fridge on and adjust the thermostat so that the compressor turns off after a long time (between 1 to 2 hours, for example). When it is warming up you should adjust the turn-on temperature. The fridge will then cool again, when you should adjust the turn-off temperature. You should repeat this process several times until you're happy with the result. You can now set the temperature control to a central position and check that the fridge settles down to a normal temperature (about 3 to 4 degrees Celsius).

It is important to keep a log of all events during this process. Which temperature did you measure? Which screws did you turn, and how much by and in which direction? It will be useful if you make a mark on the screwdriver, so you can easily see how much you turn it by. You can determine from the log if you're going in the right direction, and it will be easy to reverse any changes made, if necessary. It is important to note that it will take several cycles before the fridge settles down to a stable temperature, so you will need some patience.

The circuit

The circuit diagram is shown in **Figure 3** and is extremely simple. It consists of a potential divider with an external NTC, which is connected to pin B4. The Android I/O board already has an NTC on board, which is connected to pin B3. When the Android I/O board is put in the fridge (Bluetooth and WiFi can go through a metal fridge), then B3 will measure the fridge temperature. The external NTC has to be mounted at the end of the capillary tube. In some fridges this is easy because the cooling element in the fridge sticks out a little bit. In other fridges the capillary tube has to be pushed through a whole to get to the cooling element. In that case you should use an SMD for the NTC, which is connected via two thin wires. This NTC can then be mounted to the end of the capillary tube using some duct tape. As an alternative, the NTC can be mounted onto the cooling element in the fridge (also using duct tape) and covered with some isolation foam. In that case you won't measure the exact cooling temperature, but it will be close enough for the calibration.

The app

A screen dump of the app is shown in **Figure 4**. 'T1' is the NTC connected to pin B4 and T2 is the NTC mounted on the Android I/O board (pin B3). The app displays the measured temperature, including the minimum and maximum measured temperatures. It also gives a graphical representation of the temperature over the last hour. From the menu you can reset the minimum and maximum temperatures, and clear the graph. The software in the app is based on part 2 of the Android I/O board Etch Control example (Elektor November & December 2015). In onCreate() the GUI is defined; a loop in a thread takes care of the regular transmission of commands to the Android I/O board, and a message handler receives the measured data and passes it on to the GUI.

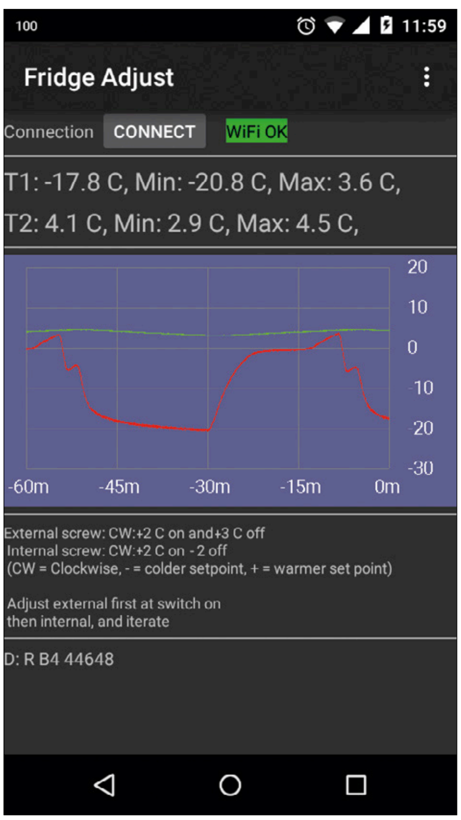


Figure 4. Screendump of the calibration App.

What is different here is the graph. An imageView is created in the GUI for the space taken up by the graph. When the graph is created, a bitmap is made in memory that has the same dimensions as the imageView. A Canvas is defined in de bitmap, where the graph is made using the drawLine, drawPoint and drawText commands. The color is defined by the Paint object. Once everything has been drawn via the Canvas in the bitmap, this bitmap is copied to the GUI and it becomes visible to the user. In **Listing 1** you can see (an abstracted version of) the code for this.

As usual, you can download the apk file and the source code for this app from the magazine website [1].

Good luck with calibrating your fridge thermostat!

(150516)

Web Link

[1] www.elektormagazine.com/150516

Listing 1. An example for creating the graph.

```
public void makegraph() {
        Paint paint;
        Bitmap bg;
        Canvas canvas;
        int color;

        double xm=imGraph.getWidth();
        double ym=imGraph.getHeight();
        bg = Bitmap.createBitmap(xm, ym, Bitmap.Config.ARGB_8888);
        canvas = new Canvas(bg);
        paint = new Paint();

        // draw line
        paint.setColor(Color.parseColor("#808080"));
        // ...
        // x1,y1 x2,y2 begin and end point
        canvas.drawLine(x1,y1,x2,y2, paint);

        // draw text
        int textsize=(int)(xm/24);
        paint.setColor(Color.WHITE);
        paint.setTextSize(textsize);

        // draw point
        paint.setColor(Color.parseColor("#FF0000")); //red
        canvas.drawPoint(x1,y1, paint);

        // Make graph visible
        // in onCreate(), imGraph is defined as:
        //     ImageView imGraph;
        //     imGraph = (ImageView) findViewById(R.id.idGrafiek);
        imGraph.setImageBitmap(bg);
}
```