Gene Automation

Table of contents

Disclaimer	2
Previous Gene Modifications, which still relate to the newer version:	2
Changes in new version (Rev 3)	3
Test of roaster and differences in the cup	7
Tasting results:	9
Final words – future work (?)	11
Parts list	12

Disclaimer

This site, I or the manufacturer of Gene Café CBR-101 shall not be responsible or liable, directly or indirectly, for any damage or loss caused or alleged to be caused by or in connection with use of or reliance of this information. No approval has been given by the manufacturer of Gene Café CBR-101 for any modifications to the Gene Café CBR-101. Any modifications done should be carried out by qualified electrical personnel.

Applying this mod to your Gene CBR 101 or similar roaster will not only void your warranty but be irreversible where cuts to the casing are made. Also you should know what you are doing when dealing with electricity and heat. This document makes no claim for correctness or completeness.

Previous Gene Modifications, which still relate to the newer version:

What might be interesting from my former forum post might be the accuracy of the modified roaster regarding the temperature control and the internal temperatures during the roast. Figure 1 shows the difference between the temperature at the heating element and the temperature after the bean chamber. Also note that the metal heating element cools down more quickly than the drum containing no beans. Internal temperatures seem fairly constant, although slightly above 50 °C.

The set temperature is, except for the beginning of the roast, quite near to the setpoint and fluctuates around it with a much smaller deviation than in the original Gene software (Figure 2). The settings for the PID controlling the heater can be changed (and saved) in my software to find the sweet spot for each roaster model and heating element.

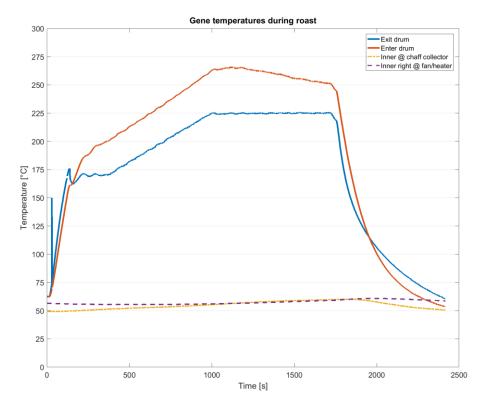


Figure 1: Gene temperature measurements during roast.

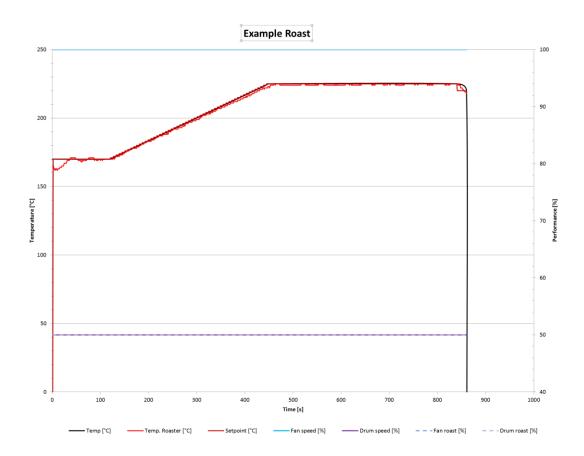


Figure 2: Plotted log data of example roast. Setpoint (dark red) and Temp (black) are the same and should be compared to the measured temperature Temp Roaster (red). Vertical line at the end indicates the start of the cooling cycle (not displayed).

Changes in new version (Rev 3)

Since previous versions of this automation required an external box with the hardware, for the new version of the Gene automation I wanted to have everything attached to the roaster so one only needs to get the machine and not an additional cable and the box with the Arduino and so forth, since I have no fixed setup for the roaster. To do so, the biggest challenge was to get all the electronics in the original chassis. For this I soldered a small circuit board holding a power-supply unit to convert from 48 V AC to 24 V DC, two transistor circuits for the fan and the drum, some additional I/O for the end-switch of the drum, power to the original relay for the heating element (which is always on and the reason for the 24 V event despite the drum motor running on 12 V) and power to the new SSR (controlled by the Arduino). This board can then be attached to the Arduino directly to the outputs. The resulting block of electronics amazingly fit into the roaster near the front where the buttons are located, just after cutting one of the middle posts.

As mentioned in my first version of the automation mod, I replaced the original fan by a 24 V one which has the same proportions since the original 12 V fan broke at some point in testing due to overvoltage. If you want to change the fan (which I would recommend for the 24 V version), look for a 75x75x30 mm 24 V radial fan e.g. for 3D printer.

Since the SSR needs cooling, it remains mounted on the right back of the roaster with most of the metal heatsink showing. Don't forget to add a ground/safety to all external metal parts (Figure 3).



Figure 3: SSR mounting.

The k-type thermocouple I kept on using is the one in the exhaust, when the air comes from the drum into the black carbon like part. I cut an additional hole in the back, opened the component, inserted the thermocouple (which should fit really snug) and reassembled it again. Also it might be possible to find a longer thermocouple and exchange it for the already installed one, making a change to this part unnecessary (Figure 4). Using only a thermocouple in the heating element, which is encased in metal, will give unusable results since it changes temperature more quickly than the cold/hot coffee beans.





Figure 4: Left image: k-type thermocouple (middle) and original Gene thermocouple (top). Right image: Thermocouple amplifier. For some reason the version number (V2) got printed on backwards.

I added an LCD panel to show the current roasting progress in a rough graph and added the exact times underneath each setpoint (Figure 7). Additionally the current and end time of the overall roast and a display of the set and actual roaster temperature at the current setpoint are displayed in the top left corner. The visual feedback of the graph is a big support during roasting since one can see changes quicker even if the scaling is not as precise as on a larger screen. Sadly I could not find a more suitable location for the display on the roaster and thus just hot glued it to the left side. The right would be very close the heating element and would probably also melt the glue on the outside of the casing.

If you are interested in this mod, you will find the compiled code attached. You should be able to build the hardware on your own using the tips in this document as well as the included images and Google. Changes to the hardware for future models are very unlikely since everything is already controllable and a second temperature sensor doesn't benefit the cause. So I think main changes will be in the software, for which an external USB-plug for future changes might be a good idea. I used the short USB-cable that came with the Arduino and put it through the venting grid on the bottom in the middle back of the roaster (after removing the middle plastic rod and mounting the transformer to the most left by drilling a new hole in the base plate). The pin configuration can be taken from Table 1 and a list of the required parts is in chapter Parts list.

Table 1: Wiring with notes.

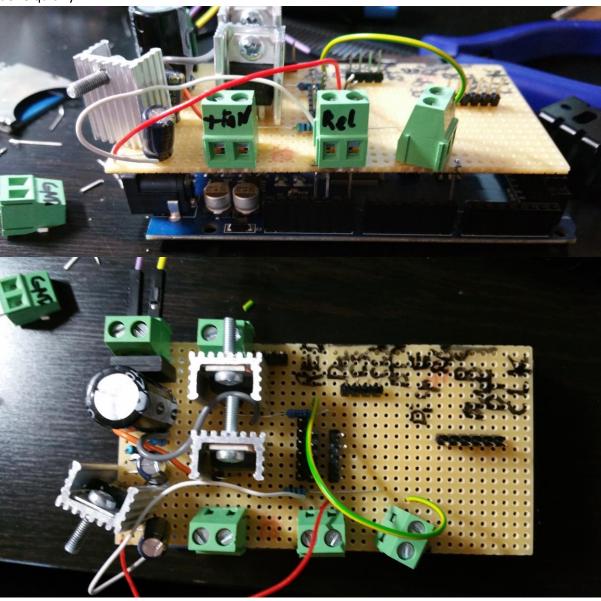
Function:	Pin:	Remark:
Drum	7	Use 2 kΩ resistor for transistor
Rotation end switch	A1	
Fan	2	Use 4.7 kΩ resistor for transistor
SSR	11	
Thermocouple CLK	4	
Thermocouple CS	5	
Thermocouple DO	6	
Encoder blue CLK	20	
Encoder blue DT	19	
Encoder blue SW	18	
Encoder red CLK	13	
Encoder red DT	12	
Encoder red SW	3	
SD MOSI	51	
SD MISO	50	
SD CLK	52	
SD CS	53	

With all components connected to ground and on the positive wire to the Arduino. Double check to use the 24 V circuit for powering the new 24 V fan as well as the original 12 V motor (default in software at 48 % to not overpower the motor; 12 V model did not break on me yet and an additional 12 V circuit is not necessary). The original fan can also be used, if the maximum power is regulated to about 48 % in the options, but at some point my fan broke, which is not so funny at 1st crack during a roast. Also you could just change the power supply to 12 V for everything and remove the original relay from the roaster but I am still not sure if the circuit is needed as a better protection for the heating element, so I kept it (Figure 5).



Figure 5: Original heater relay.

Here are some images from my hardware setup to give you an idea what it might look like. I know this can be planned better but I did some prototypes for different projects and just wanted to get this done quickly.



I also removed the original connectors from the Gene components and replaced them with some that are actually available in Germany to be able to detach the hardware more quickly. You can only use the screw-in types and label the cables or solder the connectors directly to the board. Since this is a new revision of my old setup it got quite messy.

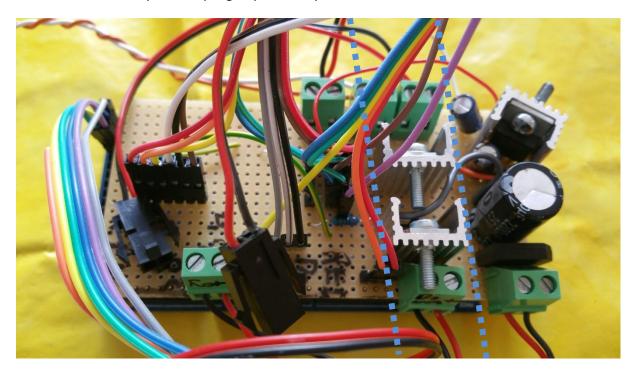


Figure 6: Complete hardware for the automated Gene. To the right of the dotted lines is the power supply which reduces Gene's internal 40 V AC to 24 V DC. Between the dotted lines are two cooled transistors for motor and fan control and the rest is just I/O and some resistors.

Test of roaster and differences in the cup

I'd like to conclude this Gene automation document/thread by adding an example what I use it for, which probably is what most people use their roasters for (roasting delicious fresh coffee). As mentioned in the video I wanted to taste how much different roast profiles with and without breaks and slopes or linear heating behavior change the flavor profile in the cup. Therefore I created five different profiles from the top of my head and roasted 110 g coffee each (Figure 8). In this case all samples are Nicaragua Arabica from Jinotega – El Paraiso, Juan de Dios Castillo Blandón, which is directly imported from Café Fausto GmbH in Munich. After two days I brewed 15 g of coffee to approx. 30 g in 25 s using a Portaspresso hand grinder and a Londinium LR (Figure 9).

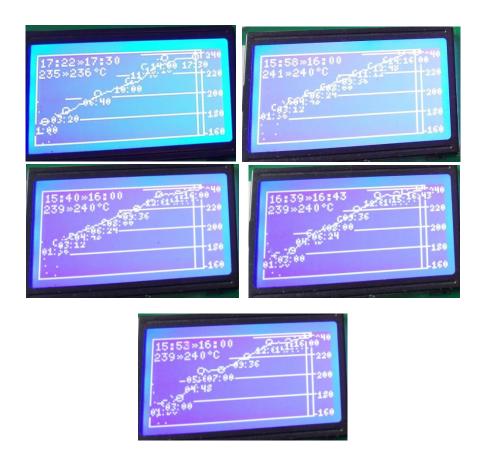


Figure 7: Five test profiles. From top left 1) Non-linear, 2) linear, 3) Break at 235 °C for 2 min, 4) same as 3 but with a non-linear slope at the beginning, 5) Break at 210 °C and 235 °C for 2 min each



Figure 8: Resulting coffee. Middle: test 1, top left: test 2, top right: test 3, bottom left: test 4, bottom right: test 5

Tasting results:

The first profile from the video should be a non-linear rising temperature slope which has a small belly and is above the linear temperature (Figure 7).

For me this roast tasted slightly sour, earthy and mild with a body of 3 out of 5.

For the linear profile of test 2 I would say the cup was not well defined and has no sourness to it, tastes kind of empty with a body of 2 (out of 5).

My taste notes for test 3 and test 4 would read that the profile with the linear heating and a break at 235 °C for two minutes produces a slightly fruity-sweet espresso with a hint of beans and bread/malt while the same profile with a slightly non-linear heating before the temperature rest results in a slightly more sour than sweet espresso with a beany taste. Both with a body of 4/5.

The last test with the double stop roast profile resulted in a balanced, slightly fruity (cherry?) espresso with a medium (3/5) body.

I looked up the Nicaragua flavor profile after my tests and it is supposed to be round, deep chocolaty. So I would probably next try a slightly longer test 3, which can be roasted freely or be planned in advance and just be corrected during roast, if needed.







Figure 9: Brewing process.

Final words - future work (?)

What is nice about the new machine is not only being able to reproduce the roasts quite faithfully for myself, but also to have a platform for exchange of opinions with others. Due to the high reproducibility, it would be easy to exchange the roast profile for a coffee you liked very much, and for others to see what you did and taste what other people prefer in their cups. I know you can also do it with Artisan but then you still have to have a lot of skill to reproduce the profile. This mod would balance out the differences for all Gene users who have it installed.

What I also tried but did not work out yet, or is not implemented in the current version, is amongst others:

- Detecting and marking first and second crack by temperature:
 - → Too many error readings of temperature sensor, might tweak code and try again.
- Keeping fan on between roasts if temperature is above a certain limit
 - → Can be used on red button exit of cooling cycle as seen in the video
- Custom heating curves, how the gaps between each pair of setpoints is filled
 - → Did not work too well with the Gene since its power is limited (see Figure 10)

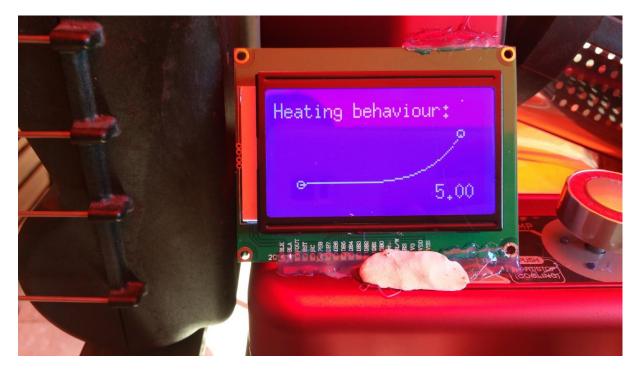


Figure 10: Example of user changeable heating behavior as menu setting.

At the moment there are some minor changes that might be useful to add but most of them are just candy and not needed for day to day roasting. What might be fun would be a Bluetooth connection from the Arduino and an App to monitor and control the roaster. If someone enjoys developing Apps and has the patience to get a good looking design we could team up for the next revision...;)

Parts list

Overall:

- 1x ATmega2560-16AU CH240G Mega 2560 R3 Board
- o 1x 12864 LCD 128x64 px ST7920
- 2x 360 degree rotary encoder module for Arduino (keeps it simple)
- 1x Circuit board
- o 1x Max31855 type k thermo element breakout board for 3 to 5 V
- o 1x K-type thermocouple
- 1x SD-card reader for Arduino
- o 1x SSR relay 25 A, 24 V 380 V
- Cable shoes to professionally connect SSR
- 1x Aluminium heatsink for SSR
- Screws, nuts, bolts for SSR/heatsink, heatsink/Gene, mounting heatsinks to transistors, ...
- o Thermal compound (SSR, transistors, Im350)
- o Connectors of your choice, maybe you can find the original ones from the Gene
- o Headers to connect the Arduino to the breadboard, different length of wire, ...

Power supply:

- o 1x KBP206 Single Phase 2.0 ampere bridge rectifier
- 1x lm350t adjustable voltage regulator (adjust to 24 V with resistors of your choice, see internet for manual)
- 1x 25 V 2200 μF electrolytic capacitor
- 2x 25 V 220 μF electrolytic capacitor

- Motor/Fan:

- o 3x Aluminium transistor heatsink (one for the lm350)
- o 2x NPN-220 TIP120 lc darlington-transistor
- o Perhaps 1x 75x75x30 mm 24 V radial fan to replace the existing 12 V one