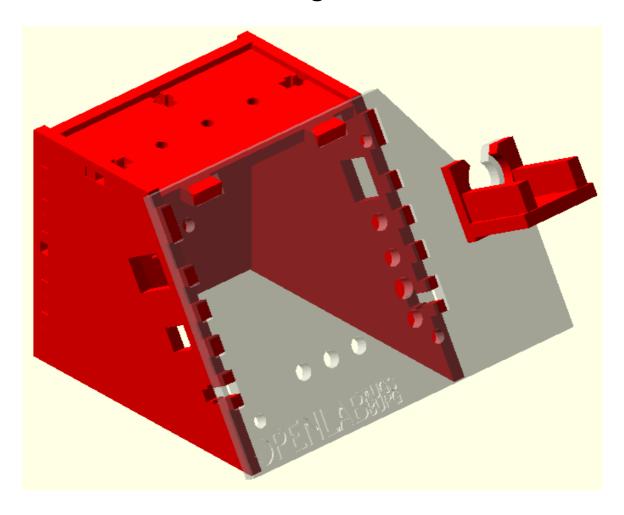
Bauanleitung Lötstation



Inhaltsverzeichnis

Einleitung	3
Bauteile	3
	6
Bestückungsseite der Platine	7
	7
Pinbelegung Arduino Nano	8
Pinbelegung TFT Display	8
Schaltpläne	9
Prinzip der Temperaturmessung	11
Bestückung	12
Software	13
Funktionen der Firmware	14
Bedienung	15

Einleitung

In der Make 5/2016 Seite 12ff wurde eine Lötstation zum Selber bauen vorgestellt. Sie basiert auf einem Arduino Nano, einem 1,8" TFT Display und einer Weller RT Lötspitze. Der Arduino übernimmt die Ausgaben auf dem Display, regelt die Lötspitzentemperatur und bietet dem Anwender mittels zweier Taster ein Benutzerinterface zum Einstellen der Solltemperatur. Vier weitere Taster wählen vorkonfigurierte Temperaturen an bzw. schalten Ein/Aus oder in Standby mit reduzierter Temperatur.

Es werden zusätzlich noch ein paar Widerstände, Kondensatoren, zwei MOSFETs und zwei ICs benötigt. Der eine IC verstärkt den Temperaturmesswert, der andere führt eine Pegelanpassung auf 3,3V für das Display durch. Die Platinendateien und der Quellcode sind auf Github¹ zu finden. Es werden ausschließlich bedrahtete Bauteile verwendet. Damit sollte jeder in der Lage sein, mit seinem "schlechten" Lötkolben diese Lötstation aufzubauen.

Bauteile

Anz.	Bez.	Bauteil	Bild
1	-	Platine	
1	JP1	DC Einbaubuchse	
1	TFT1	1.8" TFT Display (128X160)	To see the second of the secon

¹ https://www.github/to-be-filled-soon

Anz.	Bez.	Bauteil	Bild
1	U1	Arduino Nano V3 ATmega328P Board, 5V/16MHz	CORRECTOR de carine da
1	-	Audio Verlängerungskabel 3,5mm Klinke	
1	R1 (R5)	1/4W 1 K Ω Widerstand (braun-schwarz-rot)	
1	R7	$1/4W~100\Omega$ Widerstand (braun-schwarzbraun)	
1		1/4W 100KΩ Widerstand (braunschwarz-gelb) ²	
5	R3, R9, R10, R11, R12	1/4W 10KΩ Widerstand (braun-schwarz- orange)	
1	R18	1/4W 10MΩ Widerstand (braun-schwarz- blau)	
1	R5	1/4W 220Ω Widerstand (rot-rot-braun) ³	
2	R2, R8	1/4W 5,6KΩ Widerstand (grün-blau-rot)	
1	R4	1/4W 68KΩ Widerstand (blau-grau- orange)	

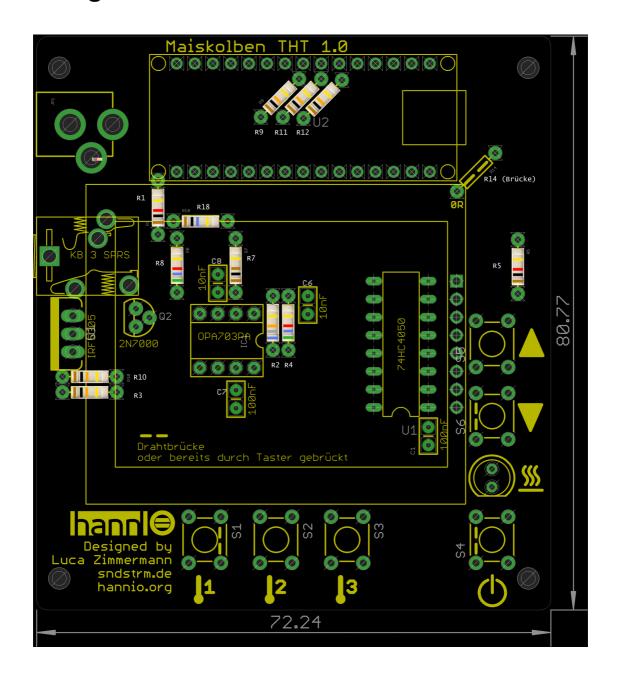
² Dieser Widerstand ist im MAKE-Artikel parallel zum C8 im Schaltplan. Die Prototypen haben wir ohne aufgebaut. Bisher ohne Befund

³ Dieser Widerstand ist im MAKE-Artikel für die LED vorgesehen, im Schaltplan zur Platine wird für die LED ein Widerstand mit $1k\Omega$ benutzt

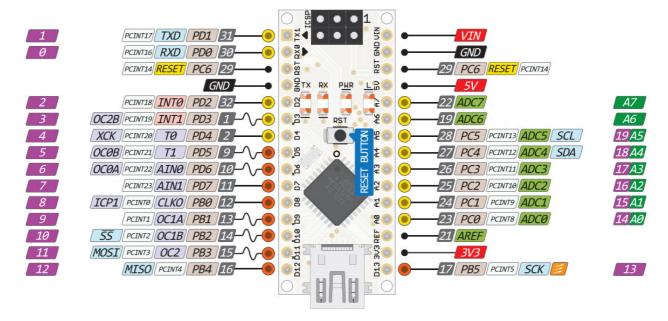
Anz.	Bez.	Bauteil	Bild
1	U1	74HC 4050 6-fach Pegelwandler	TYTTT
4	-	BL 1X10G 2,54 Buchsenleiste	Territorio de la constantina della constantina d
1	КВ	Klinkenbuchse	
2	C6, C8	10nF Keramikkondensator	Gehäusefarbe ocker / hellbraun
2	C1, C7	100nF Keramikkondensator	Gehäusefarbe blau / gelb
1	LED	LED, 5 mm rot	
6	S1 - S6	Taster	

Anz.	Bez.	Bauteil	Bild
1	IC1	OPA 703 PA Operationsverstärker	
1	Q2	N-Kanal 2N 7000 MOSFET	1223
1	Q1	P-Kanal IRF 5305 MOSFET	Re train or hite TO-220AB
1		WELLER RT 1 Lötspitze	25
1	-	MW GS60A12 Netzteil	

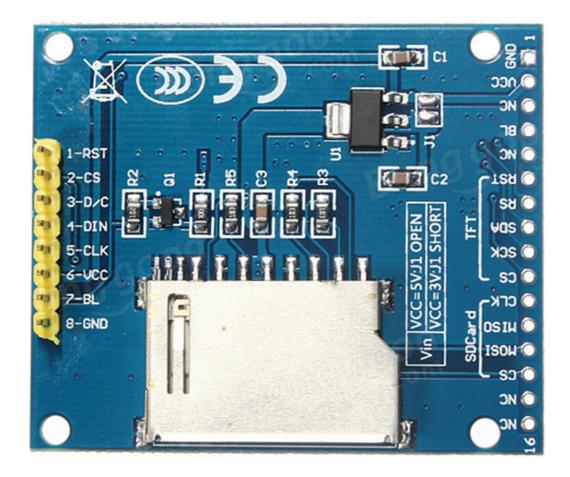
Bestückungsseite der Platine



Pinbelegung Arduino Nano

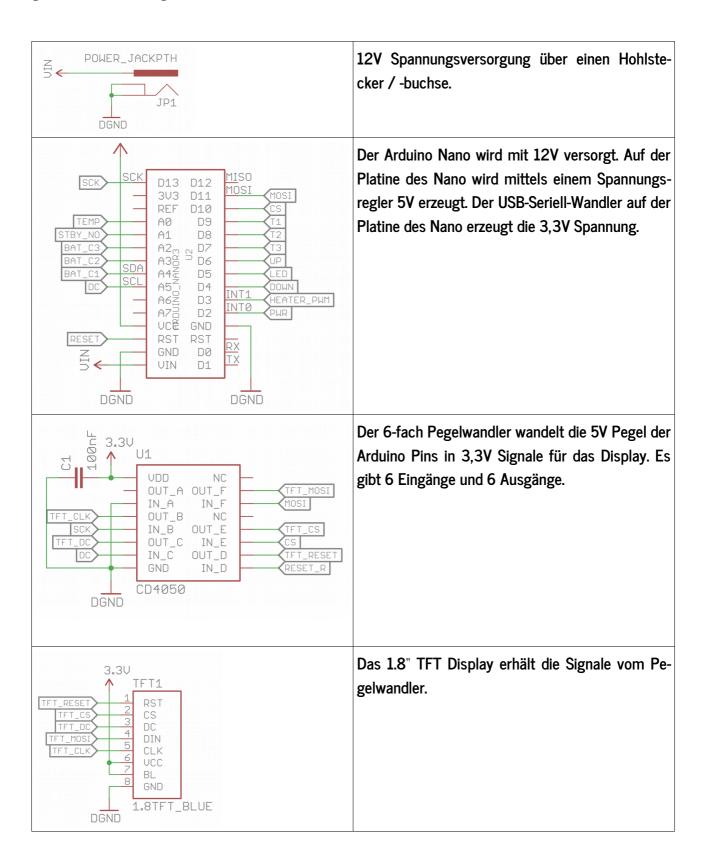


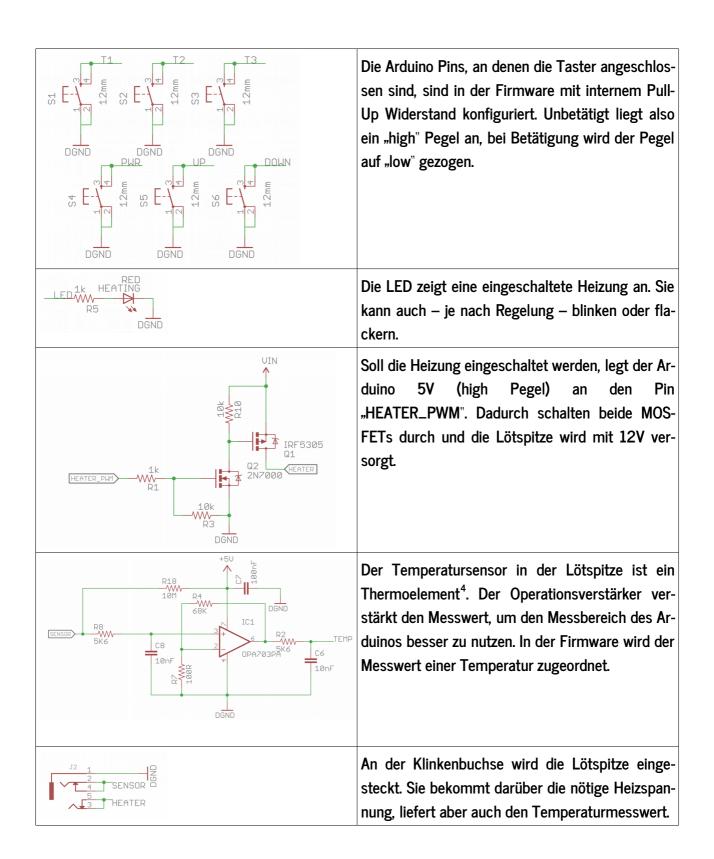
Pinbelegung TFT Display



Schaltpläne

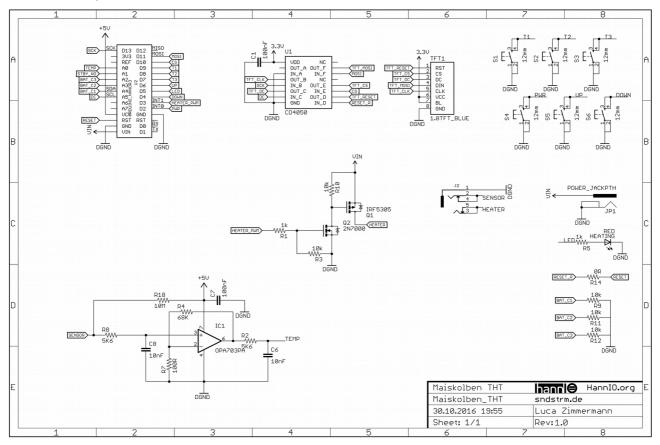
Der besseren Übersicht halber ist der Schaltplan in einzelne Abschnitte geteilt. Zu jedem Abschnitt gibt es eine Erklärung.





⁴ https://de.wikipedia.org/wiki/Thermoelement

Gesamtschaltplan



Prinzip der Temperaturmessung

In der Lötspitze befindet sich ein Thermoelement. Es erzeugt aus Wärme elektrische Energie, deren Spannung man messen kann. Eine Änderung der Temperatur ändert auch die Spannung. Da die Spannung zu klein ist (etwa 16µV pro °C Temperaturänderung), um sie direkt an einem analogen Eingang des Arduinos zu messen, benötigt man den Operationsverstärker. Bei Höchsttemperatur (ca. 450°C) sind etwa 7mV zu messen. Über die Widerstände R4 und R7 wird ein Verstärkungsfaktor von

$$\frac{68000\,\Omega}{100\,\Omega} = 680 \quad \text{eingestellt. Damit ergibt sich eine H\"ochstspannung von} \qquad 680\times0,007\,V = 4,76\,V$$

am analogen Anschluss des Nano. Die Auflösung am analog/digital Wandler ist 10 Bit, d.h. jeder Spannung von OV bis 5V wird ein Integer-Wert zwischen 0 und 1023 zugeordnet. Mit den Formeln

$$\frac{5V}{1023} \times \frac{0,00016V}{680} = 0,45 \qquad Temperatur = Messwert \times 0,45 + 64$$

erhält man dann die Temperatur der Lötspitze.

Das Elektor Magazin hat in der Ausgabe Juli/August 2015 auf S. 94ff eine fast identische Lötstation vorgestellt. Zur Umrechnung des Messwerts werden dort geringfügig andere Werte verwendet:

"Mit einem Temperatursensor wurden die ADC-Werte bei 50 °C (67) und bei 450 °C (1020)

ermittelt. Aus einem Temperaturbereich von 50...450 °C und einem ADC-Bereich von 67...1020 ergibt sich ein Multiplikationsfaktor = (450-50) / (1020-67) = 0,419 Wenn der ADC-Wert 0 ist, beträgt die tatsächliche Temperatur etwa 30 °C. Deshalb wird ein Offset von 30 addiert. Daraus ergibt sich die Formel Temperatur = temp_adc * 0,42 + 30"

Bestückung

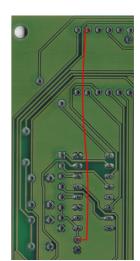
Vorbemerkungen:

- a) Die Platine hat einen Fehler (siehe Punkt 10 unten).
- b) Die Löcher für die Widerstände liegen sehr eng beieinander bitte die Drähte sehr eng am Gehäuse umbiegen.
- c) Die Löcher für den Pegelwandler sind zu klein mit etwas Mühe bekommt man den Chip zwar hinein, man muss aber ziemlich aufpassen, damit man keine Beinchen verbiegt. Besser ist es, die Löcher mit einem 1mm Bohrer aufzubohren. Dabei bitte aufpassen, dass das Lötauge nicht beschädigt wird.
- d) Beim Aufbau der Prototypen sind ein paar Fehler gemacht worden.
 - Die ICs haben auf einer Seite eine Einkerbung. Bitte die ICs so einsetzen, dass die Einkerbung mit dem Bestückungsdruck übereinstimmt.
 - Das selbe gilt für den Arduino Nano. Wird er falsch herum eingesteckt, tötet ihn die 12V Versorgungsspannung.
 - Bei zwei der Prototypen war die Verbindung zum Display wacklig und musste mit Draht nachgebessert werden – bitte sorgfältig Löten!

Aufbau:

- 1. Stiftleisten an den Nano löten. Um sie gerade auszurichten, kann man ein Breadboard zur Hilfe nehmen. Die 2x3 Stiftleiste muss nicht eingelötet werden sie dient lediglich dazu, einen Bootloader auf den Nano zu bringen. Im Lieferzustand ist bereits ein Bootloader auf dem Mikrocontroller drauf.
- 2. Firmware auf den Nano flashen. (Siehe Seite 13)
- 3. Display vorbereiten. J1 am Display muss überbrückt werden!
- 4. DC Einbaubuchse einlöten. Prüfe, ob 12V an den Pins anliegen.
- 5. Buchsenleisten vorbereiten. Der NANO benötigt 2x15, das Display 1x8 Buchsen. Die Buchsenleisten haben jeweils 10 Buchsen. Trenne 2 der 10er Buchsenleisten in eine 5er und eine 4er. Dabei geht eine Buchse verloren. Nun hast du 2x10 + 2x5 für den NANO und 2x4 für das Display.
- 6. Buchsenleisten einlöten. Prüfe, ob 12V an den Pins Vin und GND anliegen. Ein eingesteckter

- NANO sollte die Power-LED aufleuchten lassen.
- 7. R9, R11 und R12 einlöten; die Löcher liegen bei allen Widerständen sehr eng beieinander, deshalb die Drähte nah am Gehäuse umbiegen.
- 8. Brücke R14 einlöten du kannst ein übriges Drahtende eines Widerstands dafür nehmen.
- 9. Pegelwandler und C1 einlöten.
- 10. Auf der Platine ist leider ein Fehler. Der Pegelwandler hat keine Verbindung zum 3,3V Pin des Nano. Diese Verbindung muss mit einem Stück Draht hergestellt werden. Das eine Ende lötest Du an den Pin 3V3 des Nano, das andere Ende kommt an ein Bein des Kondensators C1.
- 11. Nano und Display auf die Buchsenleisten stecken und Stromversorgung einstecken. Das Display sollte nun hell werden und etwas anzeigen. Falls es dunkel bleibt, GND, BL und VCC prüfen auf Durchgang bzw. ob 3,3V Spannung anliegt.
- 12. Die Taster, LED und R5 einlöten. Die LED ist an einer Stelle abgeflacht bitte so ausrichten, dass die flache Stelle mit dem Bestückungsdruck übereinstimmt.



- 13. Beide MOSFETs und die Widerstände R1, R3 und R10 einlöten. Der große MOSFET passt nicht gut auf die Bestückungsseite! Er kann von "unten" eingelötet werrden. Das Kühlblech zeigt dabei natürlich nach innen. Alternativ kann er nach links über die Platine hinausragen dann braucht das Gehäuse an dieser Stelle ein passendes Loch.
- 14. Klinkenbuchse einlöten
- 15. Operationsverstärker, R2, R4, R7, R8, R18, C7 und C8 einlöten.

Software

Zur Übertragung der Firmware auf den Arduino NANO benötigt man die Arduino-Entwicklungsumgebung⁵ und die notwendigen Bibliotheken (Adafruit-GFX⁶, Adafruit-ST7735⁷, Arduino-PID⁸ und TimerOne⁹). Da auf den Boards kein FTDI seriell-USB Wandler ist, sondern ein CH340, muss man den dafür notwendigen Treiber¹⁰ separat installieren. Es kann passieren, dass der Arduino-Ausgang, der die Lötkolben-Heizung einschaltet, während dem Flashvorgang auf High gezogen wird. Der Lötkolben verträgt das nicht. Bitte immer vor dem Flashen den Lötkolben abstecken!

⁵ https://www.arduino.cc/en/Main/Software

⁶ https://github.com/adafruit/Adafruit-GFX-Library/archive/master.zip

⁷ https://github.com/adafruit/Adafruit-ST7735-Library/archive/master.zip

⁸ https://github.com/br3ttb/Arduino-PID-Library/archive/master.zip

⁹ https://github.com/PaulStoffregen/TimerOne/archive/master.zip

¹⁰ https://www.jens-bretschneider.de/aktuelle-treiber-fur-seriell-zu-usb-adapter/

Bis auf die Einschaltmeldung wurde die Firmware unverändert vom ursprünglichen Autor übernommen. Die Quellen stehen auf Github zur Verfügung.

Funktionen der Firmware

Die Arduino-Familie wird in einer C-ähnlichen Programmiersprache programmiert. Es ist zwingend vorgeschrieben, die Funktionen "setup()" und "loop()" zu verwenden. Dabei wird die setup-Funktion gestartet, sobald der Arduino mit Strom versorgt wird. Anschließend wird nach loop() gesprungen und dort verbleibt die Programmausführung auch.

Der Arduino Nano hat als wesentliches Bauteil einen Atmel ATmega328 Mikrokontroller. Dieser Chip hat drei verschiedene Speicher:

Flash Memory 32 KB abzgl. 2 KB für den Bootloader

SRAM 2 KBEEPROM 1 KB

Im Flashspeicher liegt das Programm / die Firmware sowie der Bootloader. Auch Konstanten können hier abgespeichert werden. Insbesondere die Einschaltgrafik ist hier abgelegt. Im SRAM werden alle Variable gespeichert. Bei der Lötstation sind das vor allem die Messwerte, die Solltemperatur und die vorwählbaren Temperaturen. Im EEPROM werden Werte abgespeichert, die vor dem Abschalten verändert wurden und nach dem Einschalten wieder verfügbar sein sollen wie z.B. die Optionen oder die drei vorwählbaren Temperaturen.

Innerhalb von setup() werden vor allem die Anschlüsse konfiguriert. Einige Pins dienen als Eingang, andere als Ausgang. Es wird festgestellt, ob die Lötstation zum ersten Mal in Betrieb genommen wird. Man sieht dann ein Auswahlmenü und soll wählen, ob man rote oder blaue Schrift sieht. Es gibt wohl 2 verschiedene TFT Displaytypen mit einem ST7735 Treiber, die eine unterschiedliche Ansteuerung der Farben benötigen. Beim nächsten Einschalten kommt diese Abfrage nicht mehr. Anschließend kommt eine Begrüßung mit OpenLab Augsburg-Logo. Bei der Inbetriebnahme wird nun in das Optionenmenü gesprungen. Hier kann man "Autoshutdown" und "Heat on boot" Ein-, bzw. Ausschalten. Sind die Optionen einmal gesetzt, kommt man wieder ins Optionenmenü, wenn man die Standby-Taste während der Anzeige des Logos drückt. Zum Schluss werden die Timer initialisiert und eine erste Temperaturmessung durchgeführt.

Die loop() Funktion schaltet die Heizung ein und bedient die serielle Schnittstelle. Man erhält darüber die gespeicherten und gemessenen Werte der Lötstation und hat die Möglichkeit, diese zu

beeinflussen. Die serielle Schnittstelle wird mit 115200 bps initialisiert. Über die Befehle "T", "P", "C" und "O" kommuniziert man mit der Lötstation:

- Txxx neue Solltemperatur setzen; z.B. "T300" für 300°C
- Pxyyy neue vorwählbare Temperatur setzen, z.B. "P2300" für setze 300°C in Speicher 2
- C löscht einen Fehlerzustand
- S Standby
- O Ein- / Ausschalten

Alle anderen Funktionen der Lötstation werden über sog. Interrupt-Serviceroutinen ausgeführt. Die Standby / Ein- / Aus-Taste ist z.B. für den Aufruf des Optionenmenüs für 2,5 Sekunden so konfiguriert. Einmal pro Sekunde wird die Interrupt-Funktion für den Timer1 aufgerufen. Dort wird die Temperatur gemessen, ggf. die Heizleistung angepasst, die Taster abgefragt und die Versorgungsspannung gemessen.

Bedienung

Neben dem Display gibt es als Anzeige für die Heizung eine rote LED. Je nach Grad der Regelung ist diese LED an, aus oder flackert / blinkt. Im Display sieht man dazu die jeweilige Soll- und Isttemperatur. Am unteren Rand des Displays sieht man die drei vorwählbaren Temperaturen.

- Die Tasten △ und ▽ erhöhen bzw. erniedrigen die Solltemperatur. Die Lötstation reagiert darauf und stellt die Regelung entspr. ein. Nach kurzer Zeit sollte die neue Solltemperatur erreicht werden.
- Die Taste schaltet zwischen den Zuständen "On", "Off" und "Standby" um. Im Zustand Off ist die Heizung ausgeschaltet. Im Standby wird die Temperatur auf 150°C heruntergeregelt. Nur wenn die Lötstation eingeschaltet ist, stehen alle Funktionen zur Verfügung.
- Die Tasten 1, 2 und 3 wählen eine der vorwählbaren Temperaturen an. Drückt man die Taste länger, wird die aktuelle Temperatur in diesen Speicher geschrieben und steht künftig zur Auswahl zur Verfügung.