

SAMOVAR

Technische Dokumentation

Firmware 7.00

Aktuelle Version: <https://www.samovar-tool.ru/doc/de/device/>
Erstellt 01.10.2026

Inhalt

Interner Abschnitt. Bitte zuerst lesen :)

Erste Schritte

- Build und Flashen mit PlatformIO

- Montage des ESP32

Minimale Konfiguration

- Summer

- Luftdrucksensoren

- Temperatursensoren

- Verwendete Spannungsregler

 - Übersicht der Regler

 - Regler an Samovar anschließen

 - Heizelement

 - Einsatz eines Zusatzheizelements

- Schütz. FI-Schutzschalter. Notabschaltung

Kühlung

 - Einsatz einer Membranpumpe

 - Einsatz einer Kreislumpumpe

 - Ohne Pumpe

- Rücklauf-Füllstandssensor

- Treiber. Schrittmotor

- Fraktionsverteilung

- Not-Aus-Taster. Zusätzliche Notfallsensoren

Empfohlene Konfiguration

- LCD-Display, Encoder, Taste

- Relaismodul richtig anschließen

Erweiterte Konfiguration

- Messgerät

- I2CStepper

- pH-Sensor und sein Anschluss

Zusätzliche Funktionen

- Port-Monitor über Bluetooth

Interner Abschnitt. Bitte zuerst lesen :)

Interne Informationen für alle, die helfen, die Bauanleitung zu erweitern und zu korrigieren.

Diese Informationen richten sich an alle, die an der Erstellung dieser Anleitung mitwirken.

Adresse für Materialien und für die Kommunikation zu dieser Anleitung: device@samovar-tool.ru

Bitte senden Sie Texte im Format .doc oder .txt und Bilder als .jpg. Senden Sie den Text getrennt von den Fotos. Sie können alles in einer Archivdatei zusammenfassen.

Die Anleitung entstand auf Grundlage der allgemeinen Hilfe. Wenn Sie einen defekten Link finden, korrigieren Sie ihn oder ersetzen Sie ihn durch einen Link auf den aktuellen Abschnitt.

- Wenn ein Abschnitt „am falschen Platz“ steht, ist das kein Problem – er lässt sich im Baum ganz einfach verschieben. Konzentrieren wir uns also vorerst nicht darauf.

Was vorgeschlagen wird:

Die Struktur der Anleitung kann im Laufe der Projektentwicklung verfeinert werden.

Jeder Abschnitt sollte möglichst viele Informationen über ein bestimmtes Bauteil enthalten: ein Foto, einen Link, wo man es kaufen kann, die Originalbezeichnung (damit man es auch findet, wenn der Link nicht mehr funktioniert), wie man es anschließt, Besonderheiten, mögliche Ersatzteile, Umbauten, usw.

Sehr erwünscht ist ein Link auf den Forumsbeitrag, in dem das Gerät besprochen wird – Ihre eigenen Beiträge finden Sie leicht, jemand anderes müsste dafür das ganze Forum durchlesen, und das mehr als einmal... .

Also ein Abschnitt pro Gerät; wird ein Abschnitt zu umfangreich, lässt sich leicht ein Unterabschnitt anlegen. (Zum Beispiel wie bei den [Reglern](#).)

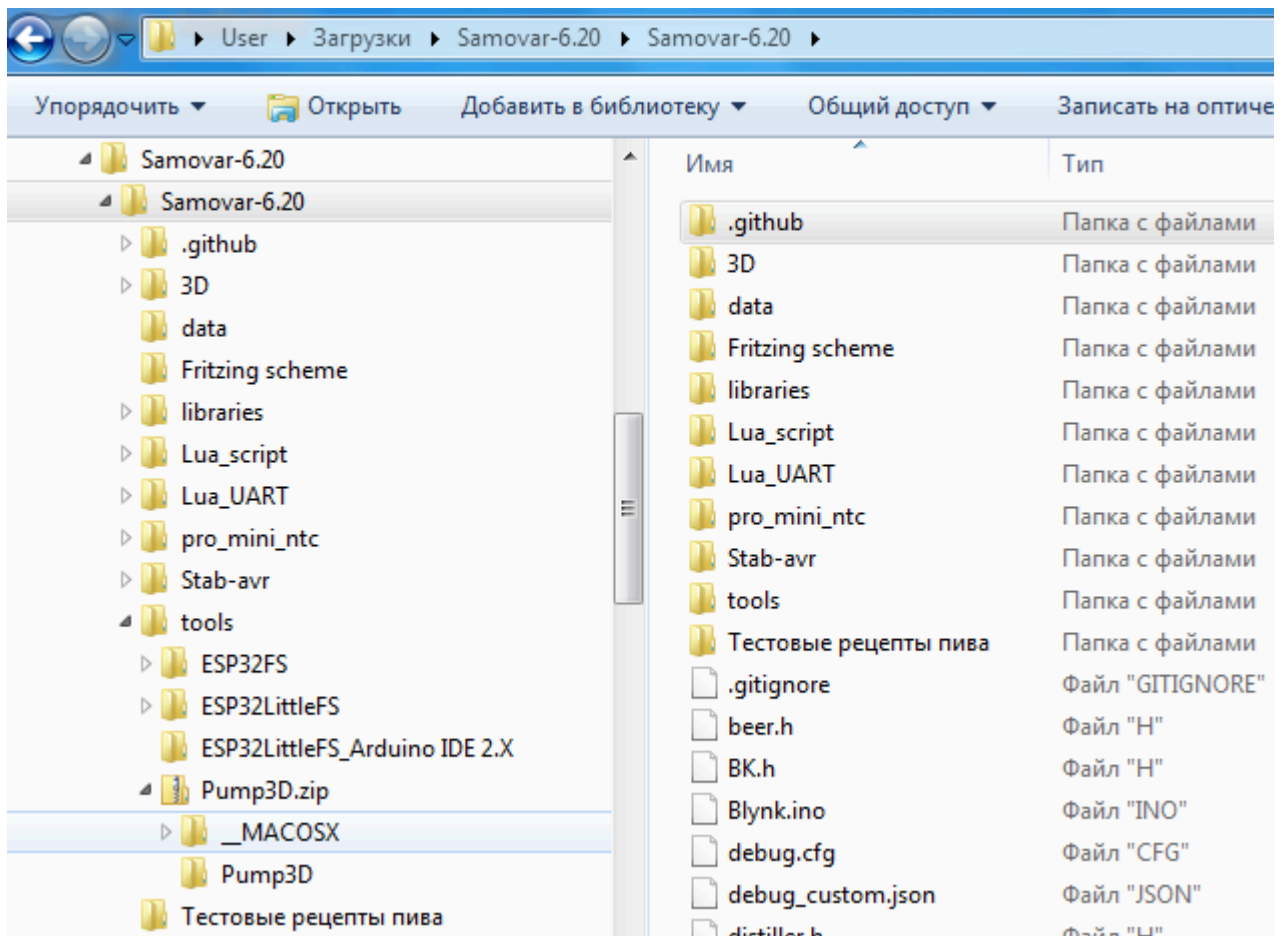
Zusätzliche Informationen über ein bestimmtes Gerät gehören nicht in andere Abschnitte. Erwähnungen – so viele Sie wollen. Setzen Sie einfach einen Link auf den Abschnitt dieses Geräts, das ist alles. Sonst kommt man beim Zusammenstellen sehr leicht durcheinander.

Ausnahmen sind möglich – zum Beispiel der Drucksensor in der Brennblass. Die Beschreibung der Umsetzung mit dem MPX5010D, der direkt an Samovar angeschlossen ist, beginnt an einer Stelle und wird im Abschnitt über das Messgerät von dranek fortgesetzt.

Urheberschaft. Auf Wunsch wird am Ende des Abschnitts ein Link zum Autor des Materials gesetzt.

Erste Schritte

Laden Sie zuerst das [Release](#) mit der neuesten Samovar-Version herunter, entpacken Sie es auf Ihrem Computer und sehen Sie sich den Inhalt an.



Was in den Ordnern liegt:

Hinweis

Beginnen Sie mit der Datei **README.md** im Projektstamm: Sie nennt die unterstützten Platinen, die Einschränkungen bei den Pins und den Installationsablauf.

- **.github** – ein Software-Ordner.
- **3D** – Modelle für den 3D-Druck. Fraktionsteller, Trichter usw.
- **data_raw** – die Quellen der Weboberfläche. Der Ordner **data** ist das fertige Abbild des Dateisystems des Geräts; er wird mit dem Befehl **python3 tools/build_web_assets.py** komplett neu erzeugt und daher nicht von Hand bearbeitet. Die Dateien **program_*.txt** enthalten die Standardprogramme.
- **Fritzing scheme** - Aufbaupläne.
- **libraries** – ein Software-Ordner (Bibliotheken).
- **Lua_script** – eine Beschreibung der Skriptsprache Lua und verschiedene sehr nützliche Skripte.
- **Lua_UART** – wie man einen separaten eigenständigen COM-Port-Monitor baut.

- [pro_mini_ntc](#) – wie man eine Ergänzung zu Samovar (das Messgerät) baut; sie kann auch eigenständig als Messgerät arbeiten.
- [Stab-avr](#) – eine Variante eines Leistungsstabilisators aus fertigen Komponenten, den Samovar steuern kann.
- **tools** – der Konfigurator (`samovar_configurator.py`), die Build-Skripte der Weboberfläche und die automatischen Projektprüfungen.
- **doc** – Beschreibungen der Firmware-Subsysteme (Markdown-Dateien).
- **partitions** und **ESP32-S3** – Partitionstabellen des Flash-Speichers für ESP32 und ESP32-S3; PlatformIO verwendet sie.
- `Stab-avr`, `Lua_UART`, `pro_mini_ntc` – separate Arduino-Sketches; sie sind nicht Teil des Samovar-Builds.
- **Test beer recipes** – einige fertige Bierrezepte zum Laden in Samovar.

In welcher Minimalkonfiguration lässt sich Samovar aufbauen?

Für den ersten Start genügen eine ESP32-Platine und USB-Stromversorgung. Die Firmware kann entweder [in der Arduino IDE](#) oder in PlatformIO gebaut werden. Die Hardware-Einstellungen stehen in **`Samovar_ini.h`**; Benutzeranpassungen, die bei Updates erhalten bleiben sollen, in **`user_config_override.h`**. Fertige Binärdateien sind im Release nicht enthalten.

Ausführliche Anleitungen: [Build in der Arduino IDE](#) und [Build und Flashen mit PlatformIO](#).

Hinweis:

Sensoren und Stellglieder, deren Vorhandensein Samovar nicht prüft, weil es keine Rückmeldung gibt:

- *Schütz / Einschalten des Spannungsreglers (Relais Nr. 1)*
- *Relais der Zusatzheizung / Anheizen des Spannungsreglers (Relais Nr. 4)*
- *Summer*
- *Ventil zum Öffnen des Wassers (Relais Nr. 3)*
- *Rücklauf-Füllstandssensor*
- *Schrittmotor der Schlauchpumpe*
- *Servo des Fraktionstellers*
- *Kühlpumpe*
- *Luftdrucksensor*
- *Drucksensor der Brennblase.*
- *Rührwerk (Relais Nr. 2)*

Sensoren und Stellglieder, deren Vorhandensein Samovar prüft, wenn:

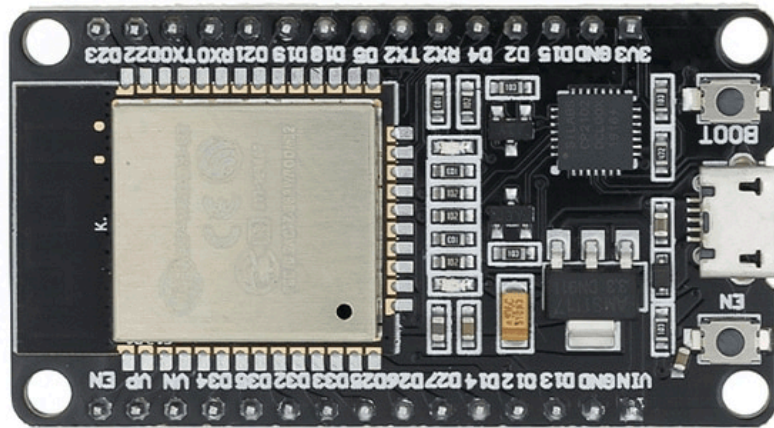
- *Temperatursensoren – wenn der Sensor beim Start des Prozesses vorhanden war. Fehlt er, kommen regelmäßig Meldungen.*
- *Wasserdurchflusssensor – wenn seine Verwendung aktiviert ist.*
- *Heizungsregler – wenn die Prüfung der Reglerverwendung aktiviert ist.*

Richten Sie dann die [WLAN-Verbindung ein](#) und öffnen Sie die Weboberfläche von Samovar. Beim klassischen ESP32 DevKit schaltet der Befehl „Heizung ein/aus“ («Включить/Выключить нагрев») GPIO2 um, an dem bei vielen Platinen die eingebaute blaue LED hängt. Beim ESP32-S3 hängt es von der konkreten Platine ab, ob es eine solche LED gibt.

Ein informativer Artikel über [ESP32](#)

Erläuterung.

ESP32 ist ein Chip, der auf einer Platine sitzt. Die Platine hat zusätzlich einen Spannungswandler, ein WLAN-Modul, Bluetooth und eventuell noch vieles mehr. Hier bezeichnet ESP32 eine Platine mit dem eingebauten Chip.



ESP-32 30P

Es gibt viele Platinenvarianten.

Die Hauptplatine des Projekts ist das **ESP32 DevKit V1 30 pin**, PlatformIO-Umgebung **Samovar**. Unterstützt werden außerdem **LILYGO T-Relay 4** (dieselbe Umgebung **Samovar**; die Platine wird im Konfigurator gewählt) und **ESP32-S3 DevKitC-1** (Umgebung **Samovar_s3**). LILYGO und ESP32-S3 haben eine andere Pinbelegung (siehe `Samovar_pin.h`); die Samovar-Leiterplatte 1.3.2 für das 30-Pin-DevKit passt nicht zu ihnen.

ESP32 ist ein Chip

DevKit V1 – die Version

30 pin - die Anzahl der Kontakte (Beinchen, Pins) auf der Platine

CH340C ist ein Adapterchip für die Datenübertragung über den USB-Bus

Type-C – die Bauform des USB-Steckers

Wo kaufen.

[AliExpress](#) (Suchbegriff: ESP32 smart board, USB flash CH340C WiFi + Bluetooth, ultra-low power consumption («Умная плата ESP32, флэш-накопитель USB CH340C WiFi + Bluetooth, сверхнизкое энергопотребление»))

[Ozon](#) (Suchbegriff: ESP32 WROOM-32 development board WiFi Bluetooth-compatible development board for IOT («ESP32 WROOM-32 Плата для разработки WiFi Bluetooth-совместимая плата для разработки IOT»))

Wenn Sie unsicher sind, ob Sie das richtige ESP32-Modell gewählt haben, fragen Sie im Forum.

Jetzt müssen Sie Samovar mit der Anlage verbinden.

Legen Sie zuerst fest, wie der [ESP32 montiert wird](#).

Entscheiden Sie dann über die Anordnung der Geräte. Es wird entweder ein einziger Block oder mehrere separate, miteinander verbundene Blöcke.

Und schließlich beginnen Sie den Aufbau mit der [minimalen Konfiguration](#) und ergänzen nach und nach die nötigen Komponenten, bis die ehrgeizigsten Ziele erreicht sind.

Fragen und Antworten. FAQ

*** Kann die ESP32 DevKit V1 38 pin-Platine mit 38 Pins verwendet werden?**

*Ja, aber sie lässt sich nicht auf der Leiterplatte montieren, die für das **ESP32 30 pin** ausgelegt ist.*

PlatformIO installieren und Samovar zum ersten Mal flashen

Samovar unterstützt zwei vollwertige Installationswege: die Arduino IDE und den Konfigurator auf Basis von PlatformIO. Diese Seite beschreibt PlatformIO. Für die Arduino IDE nutzen Sie den Abschnitt [„ESP32 für das Hochladen vorbereiten“](#).

Mit dem Konfigurator wählen Sie Platine und Ausstattung, speichern Einstellungen, flashen den ESP32, laden LittleFS separat hoch und öffnen den seriellen Monitor. PlatformIO ist ein Programm, das den Quellcode kompiliert und auf die Platine schreibt.

Windows 10/11 (nur 64 Bit): Laden Sie das [neueste Samovar-Release](#) herunter, entpacken Sie das Archiv in einen einfachen Pfad, zum Beispiel **C:\Samovar**, schließen Sie den ESP32 mit einem datenfähigen USB-Kabel an und doppelklicken Sie auf **flash_windows.bat**. Die Batch-Datei prüft Python und PlatformIO, installiert sie bei Bedarf und öffnet das Einstellungsfenster.

macOS und Linux: Öffnen Sie das Terminal im entpackten Projektordner und führen Sie aus:

```
chmod +x flash_macos_linux.sh
./flash_macos_linux.sh
```

Das Skript prüft Python mit Tkinter und PlatformIO, installiert sie bei Bedarf und öffnet denselben Konfigurator.

Wenn macOS Operation not permitted oder getcwd: cannot access parent directories meldet:

Verschieben Sie den Projektordner aus „Downloads“ in Ihren Benutzerordner und starten Sie das Skript dort. Oder erlauben Sie dem Terminal den Zugriff auf den Ordner „Downloads“ unter „Systemeinstellungen → Datenschutz & Sicherheit → Dateien und Ordner“ und starten Sie das Terminal danach vollständig neu.

Unter macOS erfordert die automatische Installation von Python und Tkinter Homebrew. Unter Linux werden apt, dnf und pacman unterstützt; das System fragt eventuell nach dem sudo-Passwort – dem Administratorpasswort des Computers.

Arbeit im Konfigurator: Wählen Sie ESP32 DevKit, LILYGO oder ESP32-S3, legen Sie die Ausstattung und bei Bedarf die anfänglichen WLAN-Zugangsdaten fest. Die übrigen Parameter verteilen sich auf die Registerkarten „Allgemein“ («Основные»), „Temperaturen“ («Температуры»), „Regler“ («Регулятор»), „BK“ («БК»), „NBK“ («НБК»), „Sensoren“ («Датчики»), „Pumpen“ («Насосы»), „Ausstattung“ («Оборудование»), „Schrittmotor“ («Шаговый двигатель»), „Rektifikation“ («Ректификация»), „Netzwerk“ («Сеть») – das sind dieselben Direktiven, die früher in Samovar_ini.h bearbeitet wurden. Drücken Sie **„Einstellungen speichern“** («Сохранить настройки»), dann **„Flashen“** («Прошить»). Die Einstellungen werden in der Datei **user_config_override.h** neben dem Projekt gespeichert und beim Aktualisieren der Quellen nicht überschrieben. Das Build- und Flash-Protokoll steht unten im Fenster. Drücken Sie bei einer neuen Platine zusätzlich **„LittleFS hochladen“** («Загрузить LittleFS»). LittleFS ist ein Speicherbereich mit der Weboberfläche und den Benutzerdateien. Erneutes Hochladen kann Programme, Lua-Dateien und Protokolle ersetzen, daher fragt der Konfigurator nach einer Bestätigung. Sichern Sie vor dem erneuten Hochladen die benötigten Benutzerdateien über den Dateieditor von Samovar.

Die Schaltfläche **„Serieller Monitor“** («Монитор порта») öffnet ein separates Diagnosefenster (Geschwindigkeit 115200). Bleibt das Flashen lange bei Connecting... hängen, halten Sie BOOT gedrückt, drücken Sie kurz RESET/EN und lassen Sie BOOT los, sobald das Schreiben beginnt.

Aktualisierung über WLAN. Spätere Firmware-Updates sind ohne Kabel möglich. Drücken Sie **„Aktualisieren“** («Обновить») neben dem Feld „Port oder Adresse“ («Порт или адрес»): Neben den COM-Ports enthält die Liste die im Netzwerk gefundenen Samovar-Geräte und die Adresse, die das Gerät nach der Schaltfläche **„IP abrufen“** («Получить IP») im seriellen Monitor gemeldet hat. Sie können die Adresse auch von Hand eingeben (zum Beispiel 192.168.1.37 oder samovar.local). Wählen Sie das Gerät statt eines COM-Ports und drücken Sie dasselbe „Flashen“ oder „LittleFS hochladen“. Voraussetzungen: Die Firmware auf dem Gerät wurde mit aktivierter Option **„Aktualisierung über WLAN erlauben“** («Разрешить обновление по Wi-Fi») gebaut (Registerkarte „Netzwerk“), Computer und Samovar sind im selben Netzwerk, und die Firewall erlaubt eingehende Verbindungen für python.exe. Vollständiges

Löschen des Flash-Speichers, der serielle Monitor und „**ESP neu starten**“ («Перезагрузить ESP») funktionieren nur über USB.

Die Schaltfläche „**Dateieditor**“ («Редактор файлов») öffnet die Dateien des Geräts (Programme, Lua-Skripte, Seiten der Weboberfläche) direkt im Konfigurator, und „**Im Browser öffnen**“ («Открыть в браузере») öffnet die Weboberfläche von Samovar.

Manueller Weg über PlatformIO:

pio run -e Samovar -t upload – ESP32 DevKit oder LILYGO;

pio run -e Samovar_s3 -t upload – ESP32-S3;

pio run -e <environment> -t uploadfs – LittleFS hochladen;

pio run -e <environment> -t monitor – seriellen Monitor öffnen.

Fertige Umgebungen mit anderem Regler: **Samovar_rmvk** (RMVK), **Samovar_sem** (SEM_AVR), **Samovar_no_power** (ohne Regler, mit UART); **Samovar_lua** – mit Lua, **Samovar_alarm_button** – mit Not-Aus-Taste. Benutzerparameter für einen manuellen Build werden in **user_config_override.h** nach dem Vorbild von **user_config_override.example.h** festgelegt.

Fahren Sie nach dem ersten Flashen mit dem [Verbinden von Samovar mit dem WLAN](#) fort.

Sind die Netzwerkdaten bereits im Konfigurator eingetragen und ist im Speicher der Platine noch kein anderes Netzwerk hinterlegt, versucht die Firmware automatisch, sich damit zu verbinden.

Montage des ESP32

An den ESP32 werden viele Peripheriegeräte angeschlossen. Daher muss der ESP32 untergebracht und die Verbindungen müssen organisiert werden.

Wichtig. Die unten angegebene Pinbelegung und die Leiterplatte Version 1.3.2 gelten für den ESP32 DevKit V1 30 pin. Für LILYGO T-Relay 4 (Umgebung **Samovar**, die Platine wird im Konfigurator gewählt) und ESP32-S3 (Umgebung **Samovar_s3**) verwenden Sie die Pinbelegung aus [Samovar_pin.h](#); die Platine 1.3.2 passt physisch nicht zu ihnen.

Назначение в проекте ← Разъем на плате	Обозн.	№ вывода	ESP-WROOM-32 DevKitC v1	№ вывода	Обозн.	Разъем на плате → Назначение в проекте
--	--------	----------	-------------------------	----------	--------	--

Выход 3,3 В	3V3	1		30	V5	Вход. Питание 5В
Общий, масса	GND	2		29	GND	Общий, масса
Мешалка ← (Реле №2) ← IN2	D15	3		28	D13	IN4 (Реле №4) → Разгон
Контактор (PM2) ← (Реле №1) ← IN1	D2	4		27	D12	BUZ1 → Бипер,звук
Управление охлаждением ← PUMP	D4	5		26	D14	IN3 (Реле №3) → Охлаждение (вкл/выкл)
Регулятор напряжения ← UART RX	D16	6		25	D27	Level → Датчик уровня флегмы
Регулятор напряжения ← UART TX	D17	7		24	D26	STEP - Драйвер ШД
Датчики температуры ← DS18B20	D5	8		23	D25	SERVO1 → Сервопривод
Энкодер ← S1 ← ENC1	D18	9		22	D33	EN – Драйвер ШД
Энкодер ← S2 ← ENC1	D19	10		21	D32	DIR – Драйвер ШД
Дисп+ давл ← SDL_LCD+BME680	D21	11		20	D35	ALARM_BTN → Аварийная кнопка
Системный, UART_0: RX	GPIO3	12		19	D34	нет → LUA_PIN
Системный, UART_0: TX	GPIO1	13		18	D39	BTN – Универсальная кнопка
Дисп+ давл ← SDA_LCD+BME680	D22	14		17	D36	WATERSENSOR → Датчик потока ОЖ
Кнопка энкодера ← KEY(ENC1)	D23	15		16	EN	Системный Сброс [LOW]

Еinschränkungen der ESP32-Pins. Die Pins GPIO34–39 sind reine Eingänge und haben keine internen Pull-up-Widerstände: Für die Notfalltaste (D35), die Universalstaste (D39) und den Durchflusssensor (D36) ist ein externer 10-kΩ-Widerstand gegen 3,3 V zwingend, sonst gibt es Fehlauflösungen und eine Notabschaltung direkt nach dem Einschalten. Diese Widerstände sind auf der Leiterplatte bereits vorgesehen. Führen Sie die Leitungen zum Durchflusssensor getrennt von den Stromkabeln des Heizelements. Pin D15 (Relais Nr. 2, Rührwerk) wird beim Reset hardwareseitig hochgezogen – das Relais daran schaltet bei jedem Start kurz ein; verwenden Sie ein Low-aktives Modul, die Relaispolarität wird in den Samovar-Einstellungen festgelegt (Felder Relais 1–4). Die Pull-up-Widerstände des I2C-Busses (Display, Drucksensor, I2CStepper) betragen 4,7–10 kΩ.

Verwendung der Leiterplatte.

Für die Verwendung des ESP32_30pin mit Samovar wurde eine eigene Leiterplatte entwickelt. Darauf werden montiert:

- der ESP32 selbst,
- ein Spannungswandler von 12 auf 5 Volt,
- ein Schrittmotortreiber,

- ein Summer,
- alle Kondensatoren und Widerstände müssen bestückt werden. Empfohlene Werte:

Kondensatoren:

C1, C2 - Elektrolytkondensatoren $500\ \mu\text{F} \times 25\ \text{V}$ (oder 1000×25)

C3, C4 - Elektrolytkondensatoren $500\ \mu\text{F} \times 16\ \text{V}$ (oder 1000×16)

Hinweis

Die Empfehlungen nennen Werte mit großer Spannungsreserve. Mit Spannungsreserve hält selbst ein nicht besonders guter Elektrolytkondensator länger und arbeitet zuverlässiger.

Wenn Sie aber möchten und sich Ihrer Bauteile sicher sind, können Sie:

C1, C2 für 16 Volt wählen

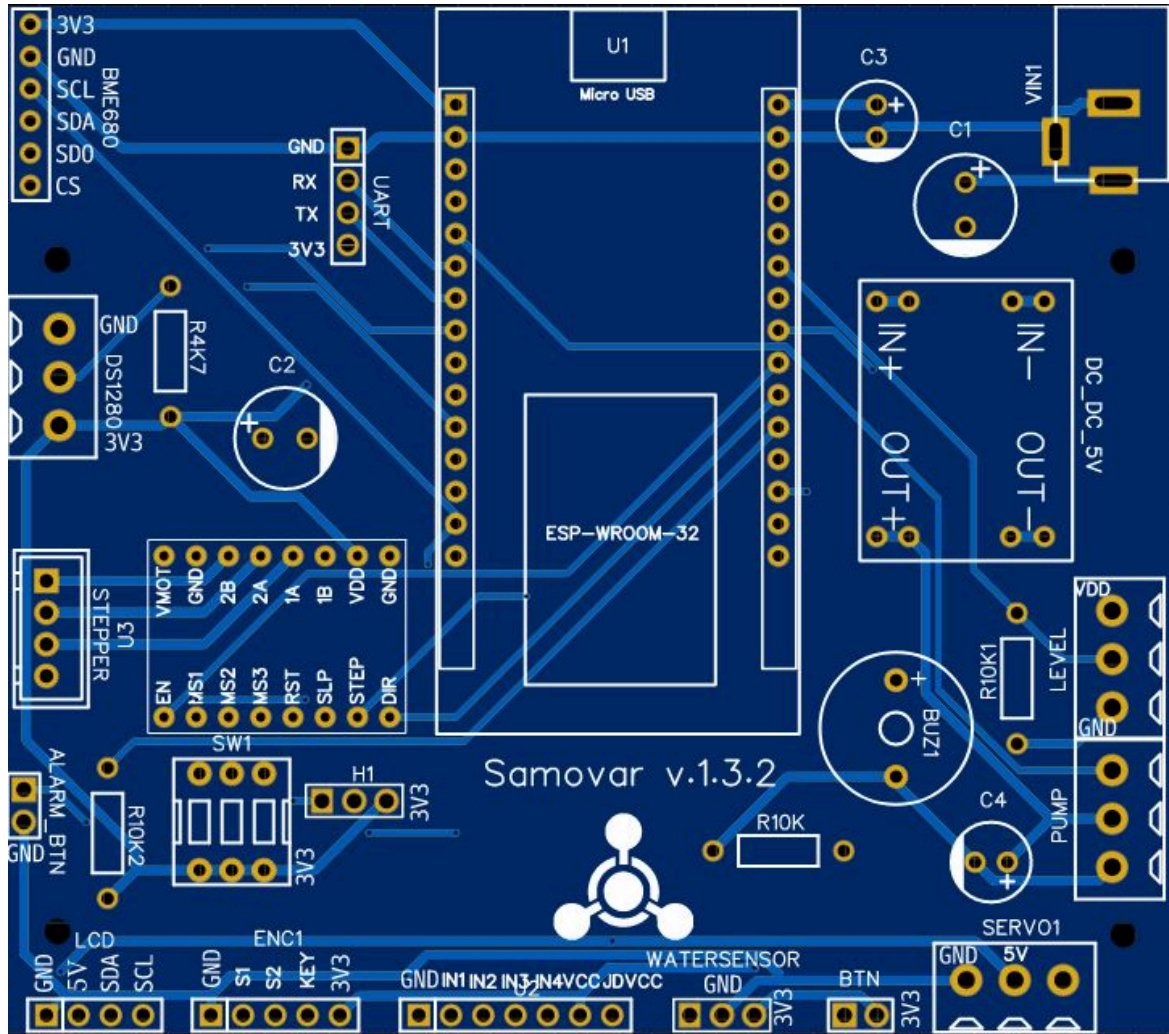
C3, C4 - für 6,3 Volt.

Widerstände: R1-R3 = $10\ \text{k}\Omega$, R4 = $4,7\ \text{k}\Omega$. Alle $0,125\ \text{Watt}$.

- Anschlussstecker. (*Hinweis: Einige Stecker können weggelassen und die Leitungen direkt angelötet werden.*)

Es wird empfohlen, alle Elemente sofort auf der Platine zu bestücken:

Summer, Luftdrucksensor, Widerstände und Kondensatoren, Stiftleiste für den Schrittmotortreiber, Schalter und Stecker. Auch wenn Sie einen Teil der Ausstattung vorerst nicht nutzen wollen. Später erleichtert das die Erweiterung der Möglichkeiten von Samovar erheblich.



Bestückungsplätze und Elemente:

- [ESP-WROOM-32](#) Montageplatz für den [ESP32](#). Es wird empfohlen, ihn auf eine Buchsenleiste (female) zu setzen
- [BME680](#) Anschlussplatz für einen Luftdrucksensor (BMP180, BMP085, BMP280, BME280, BME680).
- [DS18B20](#) Anschluss der Temperatursensoren. Rechts (auf der Platine) befindet sich ein 4,7-kΩ-Pull-up-Widerstand für die Sensoren.
- [STEPPER](#) - Anschluss des Schrittmotors NEMA17 der peristaltischen Abnahmepumpe. Rechts ist der Platz für den Schrittmotortreiber.
- [SW1 und H1](#) - Schrittteiler des Schrittmotors und zusätzliche Steuerung des Schrittmotortreibers
- [ALRM_BT](#) Anschluss der Notfalltaste und der Notfallsensoren.
- [LCD](#) - Anschluss des Displays LCD2004 (andere Displays werden nicht unterstützt)
- [ENC1](#) - Stecker für den Encoder mit Taste.
- [IN1IN2IN3IN4](#) Anschluss des Relaismoduls.

- WATERSENSOR - Stecker für den Durchflusssensor der Kühlung
- [BTN](#) - Multifunktionstaste
- [SERVO1](#) - Anschluss des Servoantriebs des Fraktionstellers
- [PUMP](#) - Steuerung der Kühlpumpe
- [LEVEL](#) - Anschluss des Rücklauf-Füllstandssensors
- [BUZ1](#) - aktiver 3-Volt-Summer
- DC_DC_5V - Spannungswandler von 12 auf 5 Volt.
- VIN1 - 12-Volt-Stecker
- C1, C2 - Elektrolytkondensatoren 500 μ F $\times$ 25 V (oder 1000 $\times$ 25)
- C3, C4 - Elektrolytkondensatoren 500 μ F $\times$ 16 V (oder 1000 $\times$ 16)
- Alle Widerstände 10 k Ω , außer einem Widerstand mit 4,7 k Ω

Woher bekommt man die Platine?

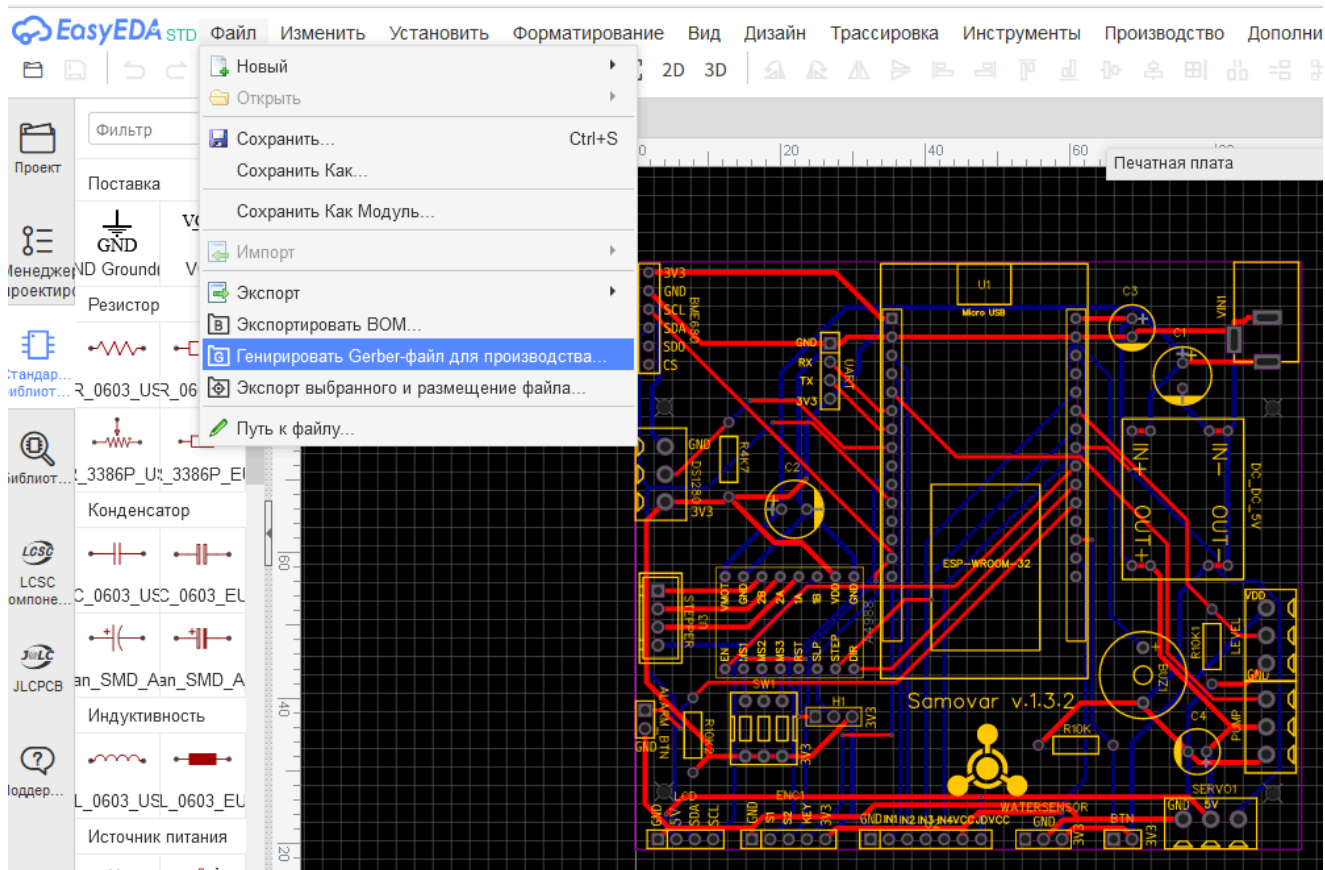
- Fragen Sie im Support-Thread nach. Vielleicht hat jemand, der schon gebaut hat, eine übrig und gibt sie ab.
- Selbst bestellen.

Die entwickelte Platine liegt auf easyeda.com. Derzeit kann man die Fertigung über diesen Dienst nicht bestellen. Den Dienst selbst können Sie aber nutzen. Für die Fertigung brauchen wir die Dateien im Gerber-Format, also die Beschreibung des Leiterplattenprojekts für die Fertigung.

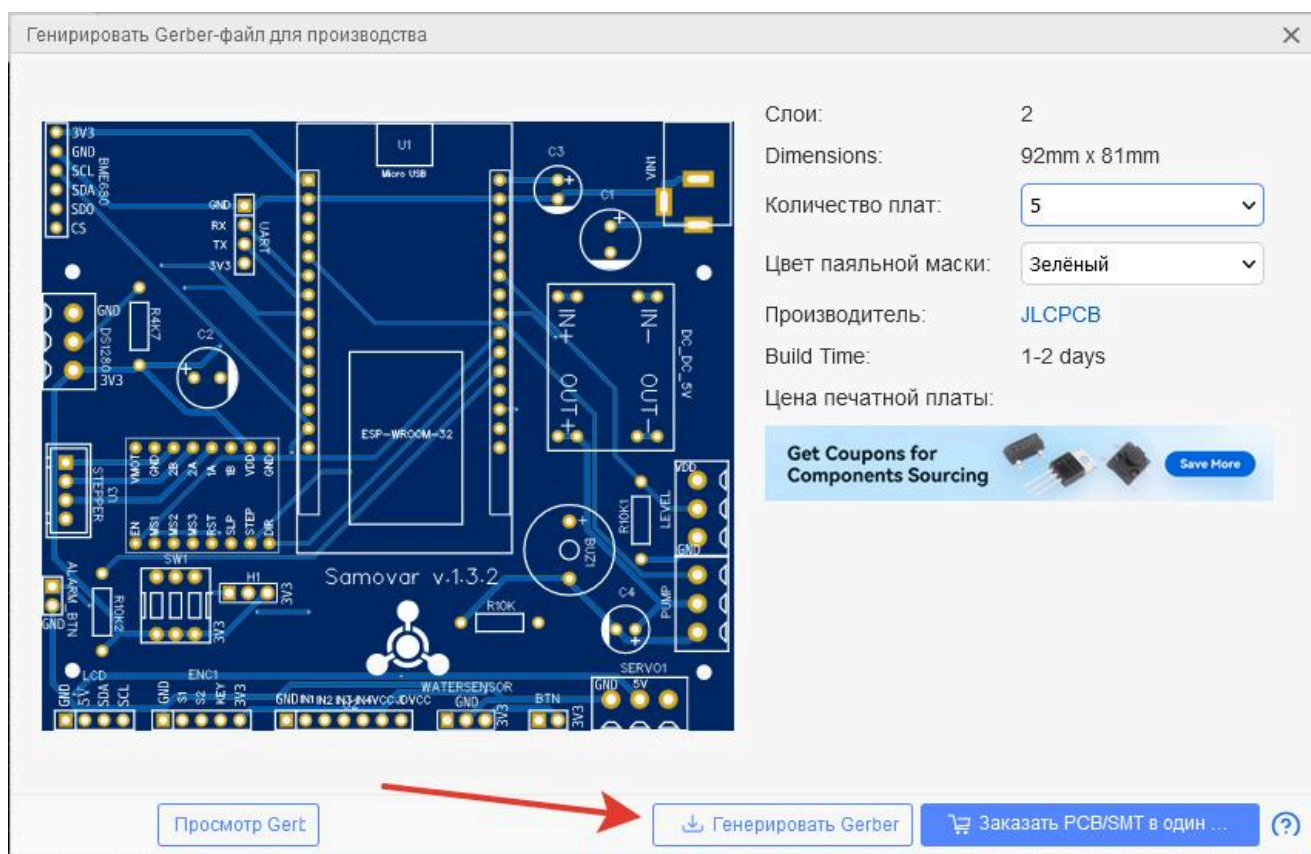
Das Gerber-Archiv für die Fertigung der Leiterplatte Version 1.3.2, Stand September 2024, können Sie [über diesen Link](#) von unserem Server herunterladen.

Oder:

Gehen Sie über das Menü: File>Generate Gerber file for production... (in der russischsprachigen easyeda-Oberfläche: «Файл>Генерировать Gerber-файл для производства...»)



Klicken Sie und wählen Sie nach der Prüfabfrage: „No, generate Gerber file“ («Нет, генерировать Gerber-файл»), es erscheint ein Fenster



Sie können ändern:

- die Anzahl der Platinen - ein Vielfaches von 5 (weniger als 5 kann man nicht bestellen).
- die Farbe der Lötstopmmaske - das ist die Farbe der Platine. Wählen Sie Grün oder Blau. Andere Farben können extra kosten.

Klicken Sie auf generate Gerber. Die erstellte Datei wird als zip-Archiv auf Ihrem Computer gespeichert.

Jetzt müssen Sie die Fertigung bestellen.

Es gibt viele Dienste, die fertigen und – vor allem – liefern. Zum Beispiel nextpcb.com Anleitungen gibt es auf der Website oder im Internet, Bezahlung über AliExpress.

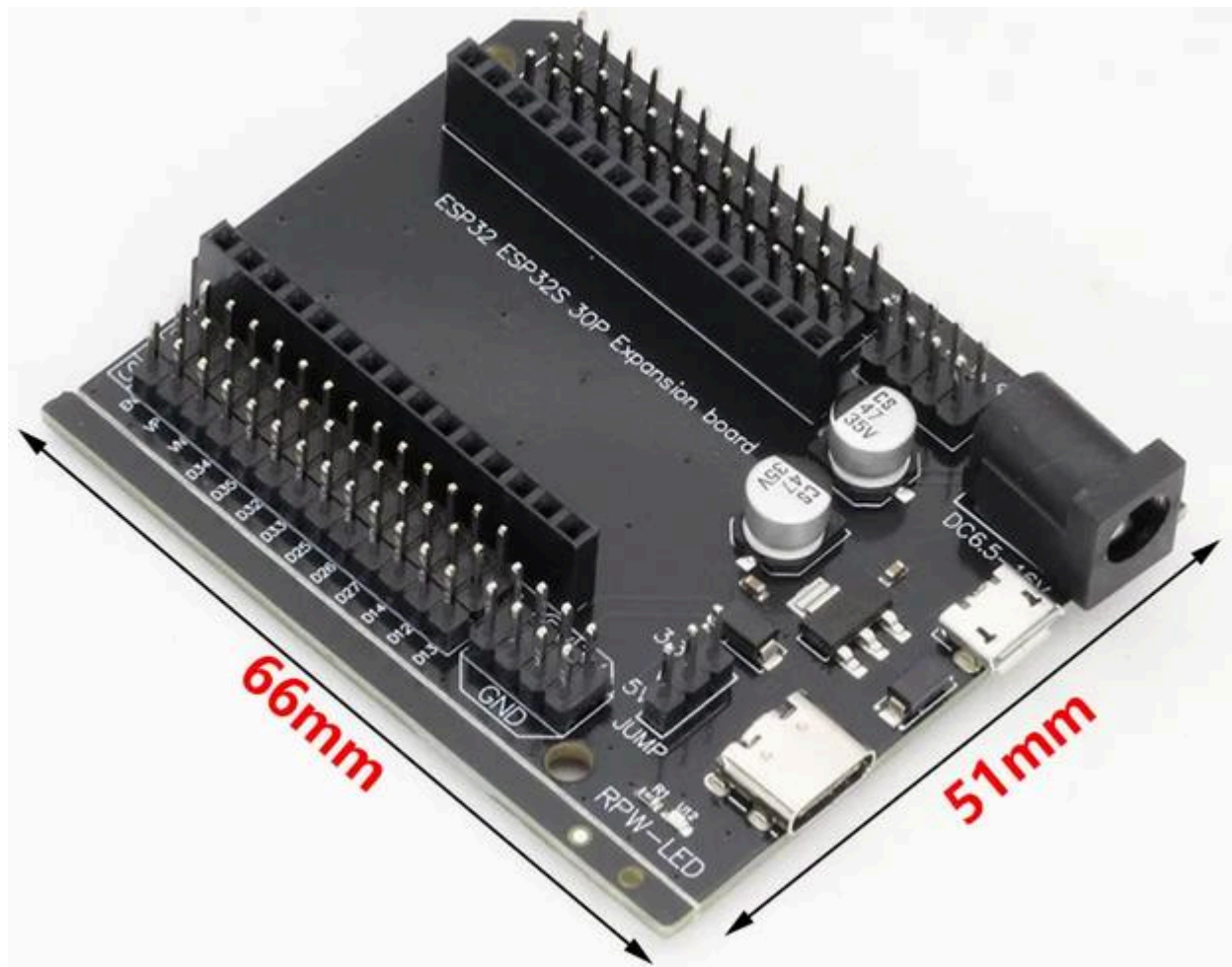
Oder gehen Sie direkt zu AliExpress und geben Sie in die Suche "custom printed circuit board manufacturing" ein. Es erscheinen Angebote von Verkäufern. Wählen Sie nach Bewertungen.

Das Bestellprinzip ist dasselbe - Sie schicken dem Verkäufer das ganze Gerber-Archiv (ist der Dateiname kyrillisch - benennen Sie ihn in lateinische Schrift um). Der Verkäufer ermittelt den Preis und teilt ihn mit. Passt er Ihnen, bezahlen Sie - und in etwa einem Monat haben Sie die Leiterplatten.

5 Platinen kosten mit Lieferung etwa 15-20 USD.

Verwendung eines Steckboards (Shield)

Für den ESP32 gibt es eine spezielle Platine ([ESP32 Shield 30Pin breadboard expansion board](#)) zum schnellen Prototyping verschiedener Geräte. Sie sieht so aus:

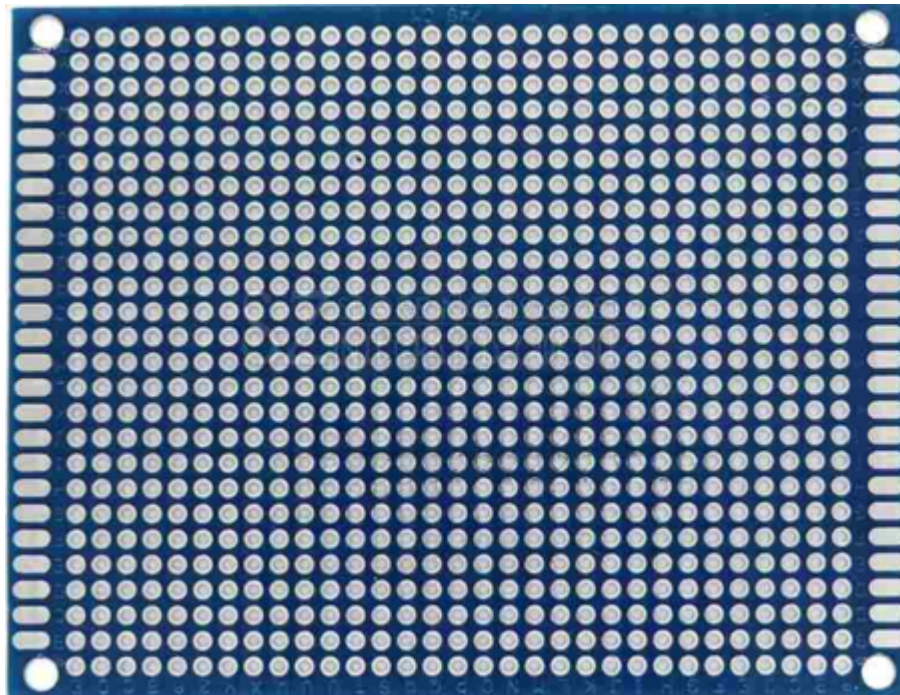


Auf dem Shield sind bereits ein Spannungswandler 16...6,5-5v und USB-Anschlüsse vorhanden - 5 Volt Versorgung. Der ESP32 wird in eine Buchsenleiste gesteckt, die Peripherie wird an die Stifte (Stiftleiste) angeschlossen. Die gesamte Verdrahtung erfolgt mit Leitungen; die Widerstände und ein Teil der Kondensatoren werden entweder auf dem Shield selbst oder an den Gehäusesteckern angelötet.

Ein Shield ist eine ordentliche und recht günstige Lösung. Für den endgültigen Aufbau ist es aber sehr zu empfehlen, alle Leitungen zu verlöten statt sie auf lösbaren Verbindungen zu lassen, sonst folgt ein Ausfall auf den anderen. Shields gibt es auch für ESP32_38 pin. Boards, damit lässt sich auch ein ESP32 mit 38 Pins verwenden.

Verwendung einer Lochrasterplatine.

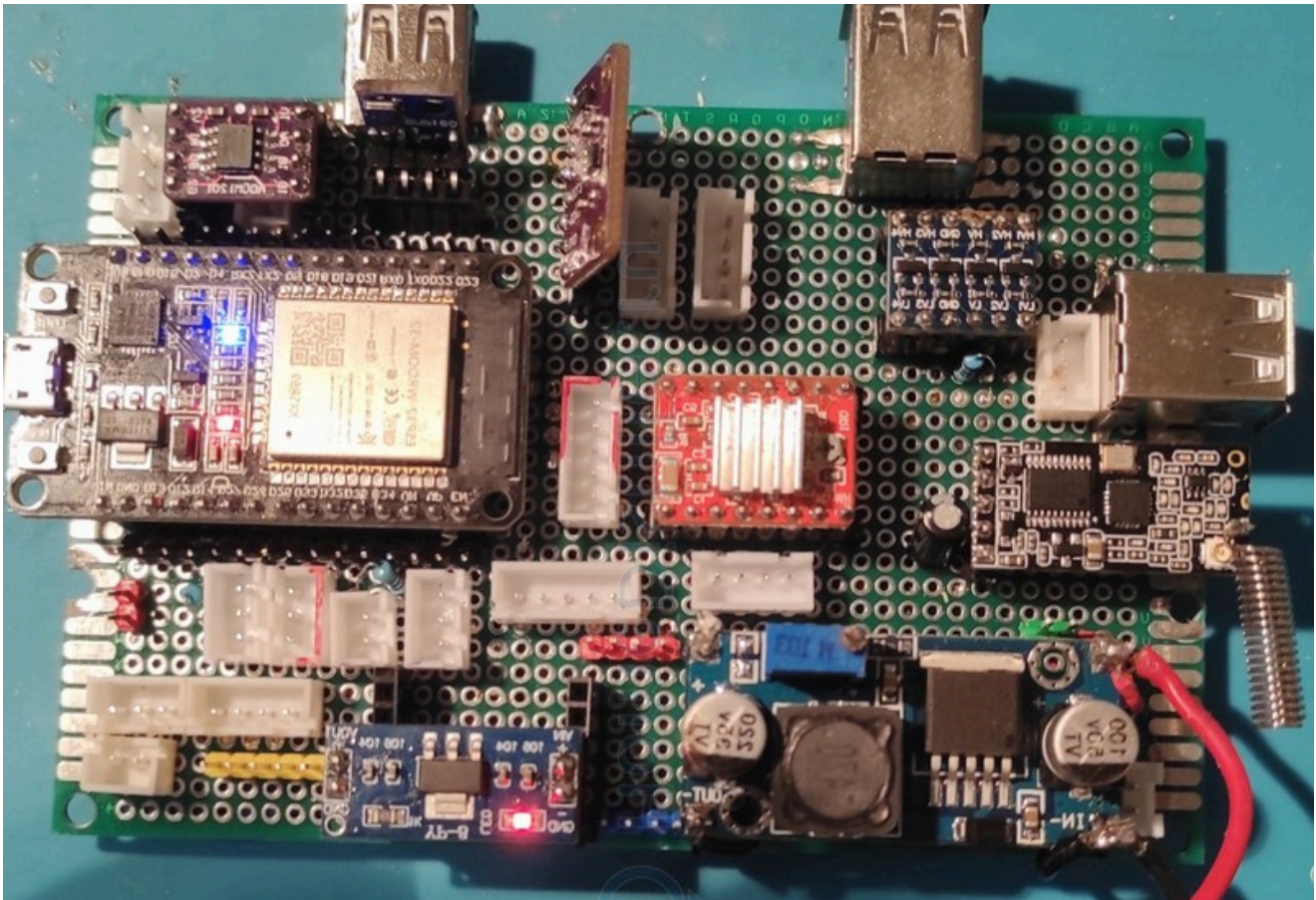
Sie können eine Prototyp-Leiterplatte zum Einlöten von Bauteilen verwenden. (Prototyp-Leiterplatte, Lochrasterplatine, Universalplatine)



Auf einer solchen Platine lassen sich alle Elemente unterbringen, und es bleibt sogar Platz für spätere Erweiterungen. Die Arbeit damit erfordert aber gewisse Fertigkeiten. Eine Warnung - Samovar auf einer solchen Platine aufzubauen ist nicht ganz einfach.

Trotzdem lässt sich das Projekt auf diese Weise umsetzen.

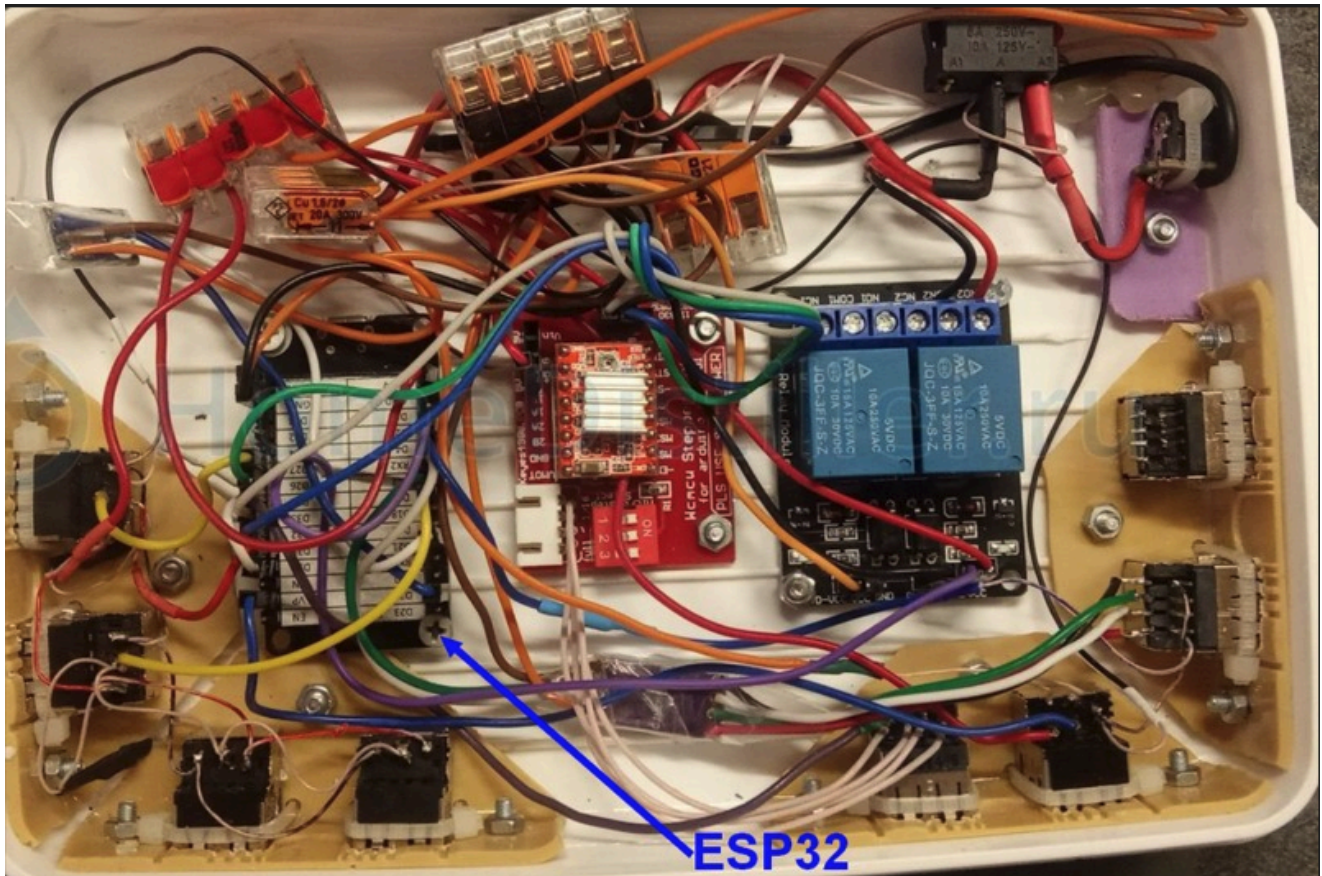
Ein Beispiel für den Aufbau von Samovar auf einer solchen Platine:



Weitere Fotos [im Forumsthread](#).

Aufbau ohne Platine

Und schließlich die letzte Variante - ganz ohne Leiterplatten. Freiverdrahtung. Ein Aufkleber mit den Pinbezeichnungen wird gedruckt und auf die Unterseite des ESP32 geklebt; der ESP32 wird umgedreht und auf Abstandshaltern (zur Kühlung) am Gehäuse verschraubt. Die Leitungen werden mit Steckern vom Typ Dupont an den ESP32-Pins befestigt.
Ein Umsetzungsbeispiel:



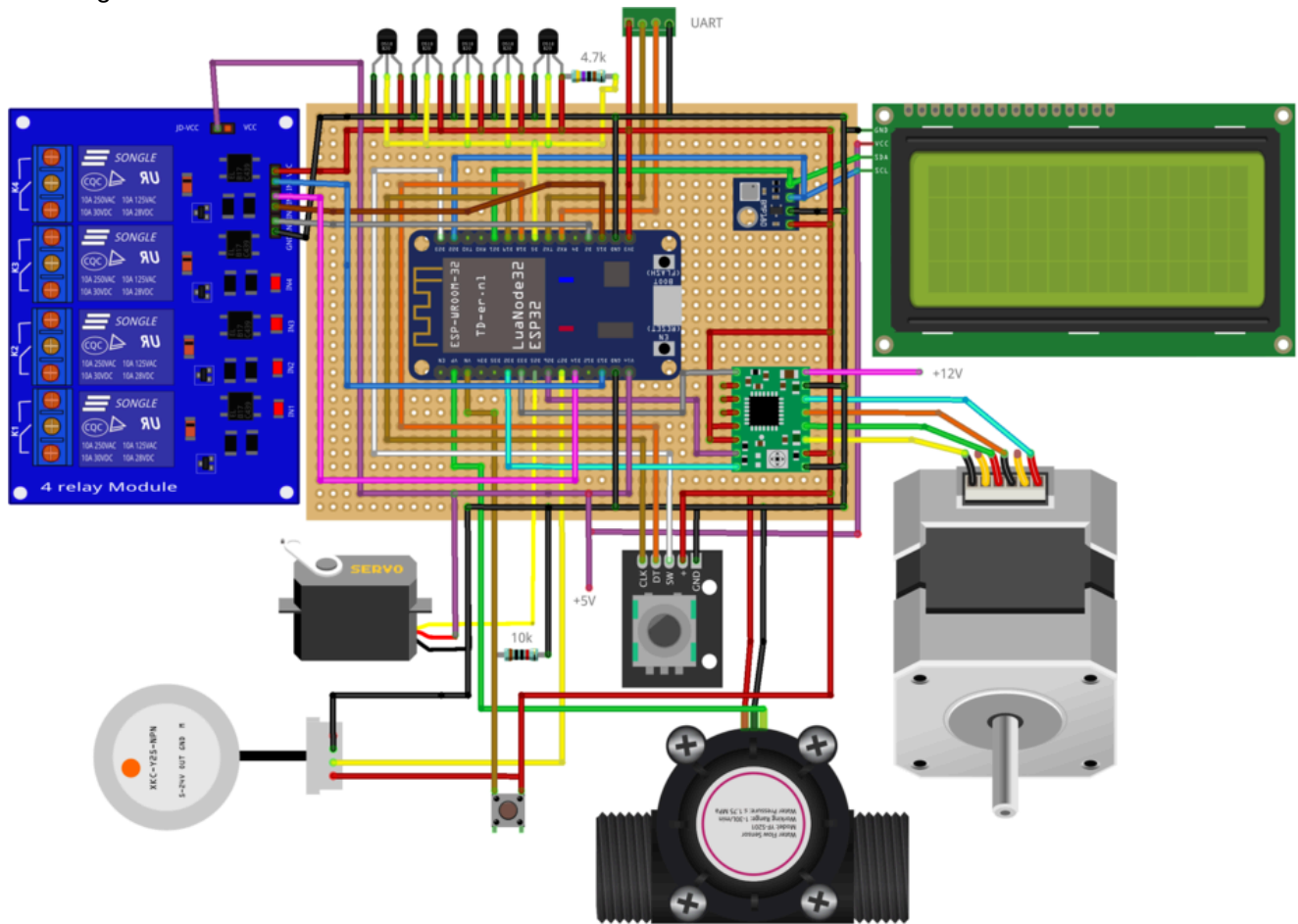
[Im Forum ansehen](#)

Empfehlungen:

- Eine Leiterplatte erleichtert den Aufbau und die anschließende Einrichtung erheblich. Vor allem aber erspart sie die meisten Probleme durch mögliche Fehler und schlechte Verbindungen zwischen ESP32 und Peripherie.
- Wenn Sie mit dem Bau beginnen möchten, aber niemand eine spezielle Leiterplatte abgegeben hat und Sie die Fertigung nicht selbst bestellen wollen - ist ein Shield die beste Variante.

Minimale Konfiguration

Die folgende minimale Konfiguration beschreibt die Überwachung ohne automatische Steuerung des Leistungsteils.



fritzing

Ein Schaltplan in höherer Auflösung liegt im Release-Archiv im Ordner mit dem Fritzing-Schema.

Es wird vorausgesetzt, dass beim Bestücken der Leiterplatte mit dem ESP32 auch der [Luftdrucksensor](#) und der [Summer](#).

eingebaut wurden. Schließen Sie als Nächstes die [Temperatursensoren](#) an. Da der Computer meist an einem Ort und die Kolonne an einem anderen steht, kann der ESP32 von einem beliebigen Handy-Ladegerät versorgt werden.

Hier wird Samovar zu Ihrem Beobachter. Er zeigt Meldungen auf dem Monitor an und sendet sie an Ihr Smartphone: Anheizen beendet, Wasser aufdrehen, Stabilisierung beendet, Temperaturen überschritten, Pause und so weiter. Außerdem sehen Sie alle Temperaturen. Bricht Dampf durch, erscheint eine Meldung; es gibt Diagramme, und Sie können das Verhalten der Kolonne auswerten usw.

Für Stabilität und Sicherheit sind aber Sie zuständig. Steuern kann er noch nichts. Er informiert nur.

Samovar nur mit Temperatursensoren und Ladegerät als Versorgung zu betreiben, ist eher ein Test. So können Sie sich vergewissern, dass ESP32 und WLAN stabil arbeiten.

Für den weiteren Aufbau müssen Sie die Samovar-Platine in ein Gehäuse einbauen und ein leistungsfähigeres Netzteil verwenden. Wenn Sie eine Kühlpumpe einsetzen wollen, brauchen Sie ein 12-Volt-Netzteil mit 80–100 Watt.

Sie können zwei Netzteile verwenden: 40–50 Watt für die Samovar-Platine und den Schrittmotor und 60–70 für die Kühlpumpe.

Nächste Schritte:

- Mit einem stabilisierenden [Spannungsregler](#) können Sie Samovar die Heizungssteuerung fast vollständig überlassen: Anheizen, Stabilisierung, Halten der Temperatur und Abschalten der Heizung am Ende des Prozesses.
- [Notabschaltung](#) wird direkt nach dem Anschluss des Reglers dringend empfohlen.
- [Kühlungssteuerung](#) automatisiert das Halten der eingestellten Temperatur am Auslauf vollständig, öffnet das Wasser zu Beginn und schließt es am Ende des Durchgangs.
- [Rücklauf-Füllstandssensor](#) überwacht das Fluten der Kolonne. Zusammen mit dem Spannungsregler senkt Samovar automatisch die Heizleistung und verhindert, dass Flüssigkeit durch das TCA (Atmosphärenrohr) ausgeworfen wird. Mit diesem Sensor lässt sich die Kolonne während der gesamten Abnahme automatisch im Vorfluten-Modus halten.
- [Abnahmepumpe](#)
- [Fraktionsverteilung](#)
- _

Summer

Meldet Ereignisse: Das Signal ist eine Folge von fünf kurzen Pieptönen. Schalten Sie ihn in der Weboberfläche ein oder aus: „Einstellungen“ («Настройки») → Registerkarte „Allgemein“ («Основные»), Kontrollkästchen **„Summer für Benachrichtigungen verwenden“** («Использовать зуммер для нотификации») (ist das Kontrollkästchen deaktiviert, ist der Summer vollständig abgeschaltet). Auf derselben Registerkarte gibt es das Kontrollkästchen **„Summer bei Programmwechsel einschalten“** («Включать зуммер при смене программы»).

Er wird direkt an einen ESP32-Pin angeschlossen (Anschluss BUZ1 auf der Leiterplatte): D12 beim ESP32 DevKit, GPIO0 beim LILYGO T-Relay 4, GPIO39 beim ESP32-S3.



Aktiver Summer DC 3V (rotes Etikett); 5V (blaues Etikett); 12V (schwarzes Etikett)

Verwenden Sie einen aktiven Summer. Ein passiver funktioniert nicht. Achten Sie beim Einbau auf die Polarität. Die Kennzeichnung sollte auf dem Gehäuse stehen, sonst ist das lange Beinchen der Pluspol.

Wenn Sie einen Summer haben und herausfinden wollen, ob er aktiv ist: Stellen Sie das Netzteil auf 3 Volt ein, schließen Sie den Minuspol an und tippen Sie mit dem Pluspol einige Male an den anderen Kontakt. Ist kein Ton zu hören, erhöhen Sie die Spannung auf 5 Volt.

Hören Sie einen Ton, ein Piepen, ist er aktiv; hören Sie beim Antippen mit dem Pluspol nur Klicken, ist er passiv. Dasselbe gilt für 12 Volt.

Empfohlen wird ein 3-Volt-Summer, aber auch ein 5-Volt-Summer wurde schon verwendet. Die Lautstärke ist durchaus ordentlich.

Eine Lautstärkeregelung für den Summer gibt es nicht. Sie können versuchen, einen Widerstand in die Versorgungsleitung einzufügen, oder die Öffnung einfach etwas mit Klebeband abdecken.

Luftdrucksensoren

Samovar 7.00: Der Sensortyp wird im Konfigurator auf der Registerkarte „Sensoren“ («Датчики») gewählt; Samovar_ini.h muss nur bei einem nicht standardmäßigen Aufbau von Hand bearbeitet werden.

Hinweis.

Samovar funktioniert auch ohne diesen Sensor, es ist aber besser, ihn einzubauen, zumal der Sensor sehr wenig kostet.

Das Projekt kann die Sensoren BMP180, BMP280, BME280 und BME680 verwenden. Einige davon messen nicht nur Luftdruck und Temperatur, sondern auch die Luftfeuchtigkeit und sogar die Luftqualität. Samovar nutzt nur die Luftdruckdaten, daher spielt es keine Rolle, welchen dieser Sensoren Sie verwenden.

Der Sensor wird zusammen mit dem Display an den I2C-Bus angeschlossen: SDA D21 und SCL D22 beim ESP32 DevKit, GPIO33/GPIO32 beim LILYGO T-Relay 4, GPIO8/GPIO9 beim ESP32-S3. Die Messwerte werden in mm Hg umgerechnet und zur Korrektur der Siedetemperatur verwendet.

Die Leiterplatte hat Löt pads für einen Sensor mit 6 Pins, von denen nur 4 angeschlossen sind.

Wenn Sie also einen Sensor kaufen und nicht einen vorhandenen verwenden, nehmen Sie besser den einfachen und zuverlässigen BMP180. Mehr über diesen Sensor lesen Sie [in diesem Testbericht](#).

Der Sensor braucht keine weitere Einstellung.

Achtung

*Da die Sensoren über den I2C-Bus arbeiten, können manche **BMP280**-Sensoren eine andere Adresse haben. Kommentieren Sie in diesem Fall in der Datei*

Samovar_ini.h die Zeile `#define USE_BMP280_ALT` ein

Es sollte also so aussehen:

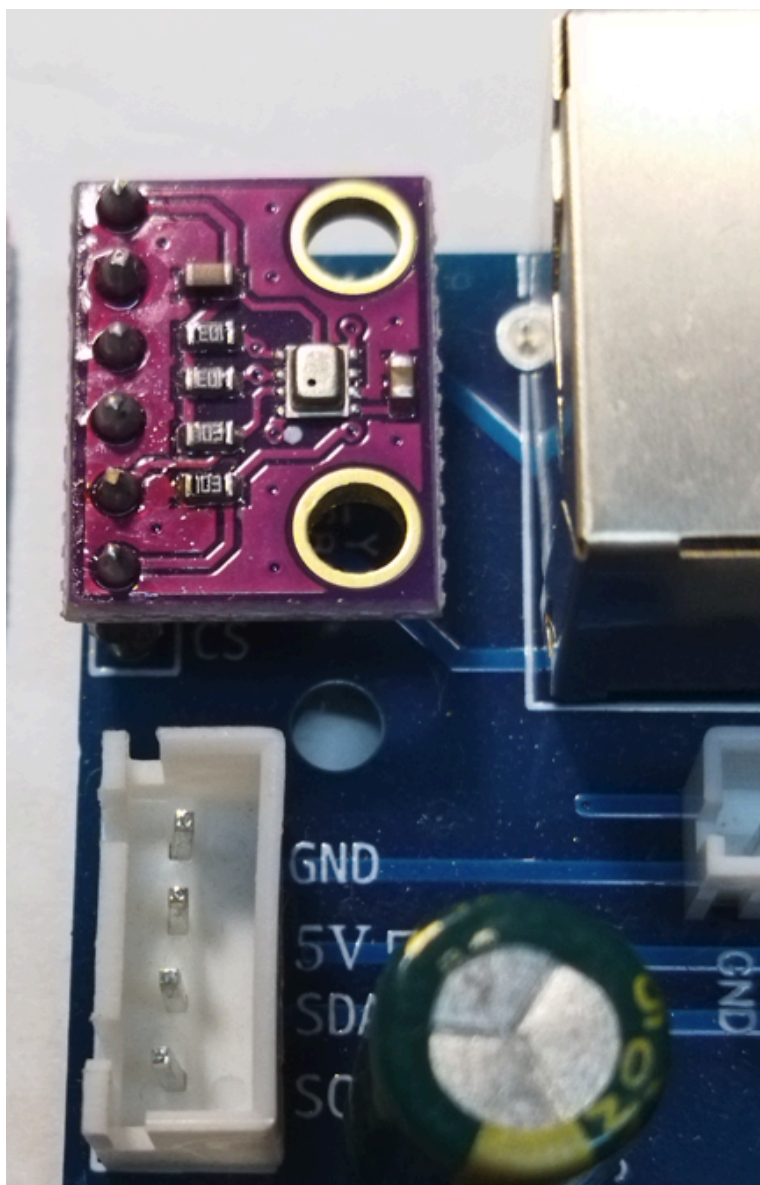
...

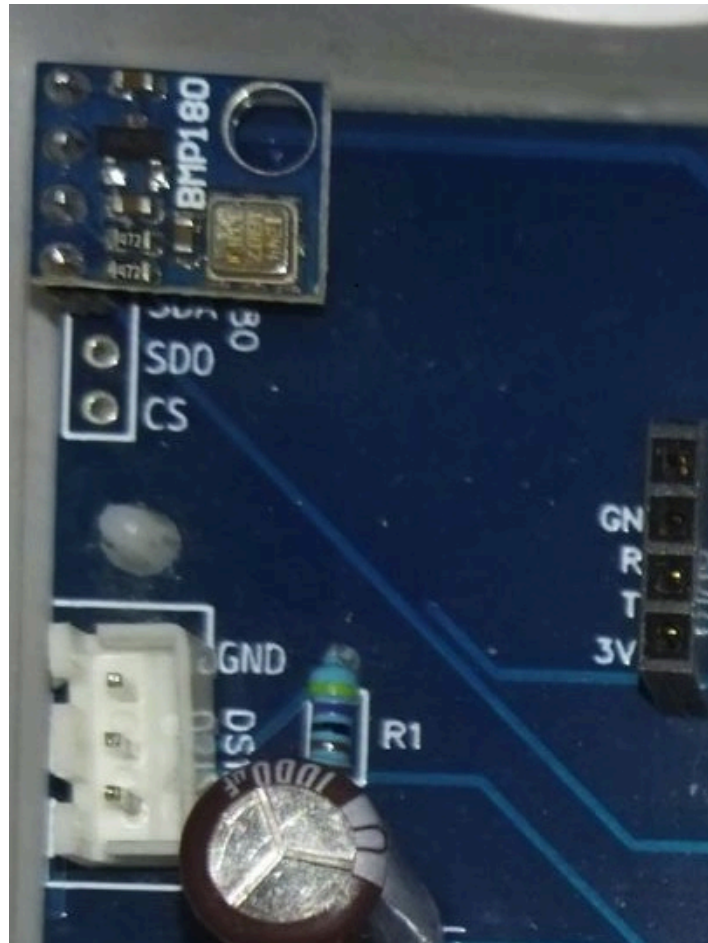
```
//#define USE_BMP280 //use the BMP280 pressure sensor
```

```
#define USE_BMP280_ALT //use the BMP280 pressure sensor with the alternative address.
```

*In der Standardkonfiguration von Samovar 7.00 ist der BME680 aktiviert. Im Konfigurator wird der Sensor auf der Registerkarte „Sensoren“ («Датчики») im Feld „Luftdruck“ («Атмосферное давление») gewählt: „BMP180/BMP085“, „BMP280“, „BMP280, alternative Adresse“ («BMP280, альтернативный адрес»), „BME280“, „BME680“ oder „Nicht verwenden“ («Не использовать»). Lassen Sie beim Kompilieren in der Arduino IDE genau eine passende Direktive in **Samovar_ini.h** aktiv; für einen BMP280 mit alternativer Adresse verwenden Sie `USE_BMP280_ALT`.*

Am besten löten Sie die Sensorplatine direkt auf die Leiterplatte.





Temperatursensoren

Samovar verwendet digitale Sensoren vom Typ **DS18B20**. Schließen Sie sie mit drei Adern an: 3,3 V Versorgung, GND und eine gemeinsame Datenleitung mit einem 4,7-k Ω -Pull-up-Widerstand nach 3,3 V. Datenleitung: GPIO5 beim ESP32 DevKit, GPIO12 beim ESP32-S3 und GPIO27 beim LILYGO T-Relay 4. Alle Sensoren werden parallel an eine Leitung angeschlossen; die Firmware unterscheidet sie anhand ihrer eindeutigen Adressen.

Unterstützt werden fünf Rollen: Dampftemperatur, Kolonnenschuss, Kühlwasser bzw. Rücklauf, Brennblase und TCA (Entlüftungsrohr zur Atmosphäre). Ordnen Sie nach der Installation die erkannten Adressen in den Samovar-Einstellungen den passenden Sensoren zu („Einstellungen“ («Настройки») → Registerkarte „Temperatur“ («Температура»), Felder „Adresse des Dampfsensors vor dem Dephlegmator“ («Адрес датчика пара перед дефлегматором»), „Adresse des Sensors im Kolonnenschuss“ («Адрес датчика в царге»), „Adresse des Wassersensors“ («Адрес датчика воды»), „Adresse des Sensors in der Brennblase“ («Адрес датчика в кубе»), „Adresse des TCA-Sensors“ («Адрес датчика в ТСА»)) und prüfen Sie die Messwerte, bevor Sie mit dem Heizen beginnen. Die Sensoren werden mit 12-Bit-Auflösung abgefragt. Wenn ein Sensor beim Start des Prozesses angeschlossen war und dann verschwindet, meldet Samovar dies regelmäßig.

Verwenden Sie bei langen Leitungen hochwertiges Kabel, zuverlässige Verbindungen und eine separate Stromversorgung für die Sensoren. Parasitäre Versorgung über zwei Adern wird nicht empfohlen.

Verwendete Spannungsregler

Samovar 7.00: Der Reglertyp wird im Konfigurator auf der Registerkarte „Regler“ («Регулятор») gewählt: KVIC, RMVK, SEM_AVR oder „Nicht verwenden“ («Не использовать»).

ACHTUNG!

Alle an Samovar angeschlossenen Regler müssen spannungsmäßig physisch getrennt sein. Das können Relais, Optokoppler, Datenbus-Isolatoren wie ADUM usw. sein.

Schließen Sie die Regler niemals direkt an, sonst gelangt die Netzspannung zu Ihnen und zu Samovar. Das Beste, was dann passieren kann, ist, dass Samovar durchbrennt.

Samovar unterstützt mehrere Varianten. In der Arduino IDE erfolgt die Auswahl über Direktiven in **Samovar_ini.h**. In PlatformIO können Sie eine fertige Umgebung wählen: **Samovar** – KVIC, **Samovar_rmvk** – RMVK, **Samovar_sem** – SEM_AVR, **Samovar_no_power** – ein Build ohne UART-Regler. Es kann immer nur eine Variante gewählt werden.

Spannungsregler mit Steuerung über das UART-Protokoll

Algorithmus für einen Regler mit UART.

Einschalten: Samovar schaltet Relais Nr. 1 ein und wartet die Zeit

SAMOVAR_USE_POWER_START_TIME ab (standardmäßig 2000 ms; im Konfigurator ist das „Startverzögerung des Reglers, ms“ («Задержка запуска регулятора, мс») auf der Registerkarte „Regler“; bei SEM_AVR werden davor zusätzlich 3 s für den Start des Reglers eingeräumt), danach sendet es dem Regler den Arbeitsbefehl. Antwortet der Regler nicht, wird die Heizung mit der Meldung „Zeitüberschreitung des Reglerbefehls“ («Таймаут команды регулятора») abgeschaltet. Relais Nr. 4 wird separat vom Algorithmus des gewählten Modus gesteuert und muss nicht gleichzeitig einschalten.

Ausschalten: Die Relais Nr. 1 und Nr. 4 werden sofort abgeschaltet. Nach 700 ms wird dem Regler der Befehl SLEEP gesendet, nach weiteren 200 ms endet die Stoppssequenz.

1. Regler [KVIC](#)

Voll kompatibel mit Samovar. Die Einstellschrittweite am Encoder beträgt 0,5 Volt. Die Stabilisierung erfolgt auf 0,1 Volt genau. Er kann im Notfall von einem unabhängigen Sensor abgeschaltet werden.

2. Regler [RMVK](#) (mit Samovar funktionieren nur die Versionen Nr. 5 und 6)

Alle RMVK-Modelle haben ab Werk eine galvanische Trennung an den Steuerkontakten, für den Betrieb mit Samovar ist kein Umbau nötig.

Die Einstellschrittweite am Regler und von Samovar aus beträgt 1 Volt. Die Stabilisierung erfolgt auf 1 Volt genau. Version Nr. 6 kann im Notfall von einem unabhängigen Sensor abgeschaltet werden. Über den RMVK steuert Samovar das Anheizen, die Stabilisierung und die Notabschaltung der Heizung vollständig.

Achtung! Der RMVK arbeitet nur mit ganzen Werten, hat also eine Einstellschrittweite von 1 Volt; das bringt Besonderheiten mit sich, wenn Samovar im Modus [Vorfluten](#) arbeitet.

3. Regler SEM AVR

Kompatibel mit Samovar. Eine Besonderheit: In der Samovar-Oberfläche wird die an die Last abgegebene Leistung nicht in Volt (wie bei den beiden vorigen Reglern), sondern in Watt eingestellt und angezeigt.

Für den korrekten Betrieb mit Samovar müssen Sie den Sketch Stab-avr.ino verwenden, der mit Samovar mitgeliefert wird und im Verzeichnis ...Samovar\Stab-avr liegt

Spannungsregler ohne externe Steuerung über Protokolle.

Zum Beispiel:

[RM2](#), [RM2M](#), [RM2-Pro](#) (im Folgenden einfach RM) .

Alle RM-Modelle haben ab Werk eine galvanische Trennung an den Steuerkontakten, für den Betrieb mit Samovar ist kein Umbau nötig.

Die Steuerung dieser Regler beschränkt sich auf zwei Parameter:

- Den Regler selbst ein-/ausschalten.
- Den Modus "Anheizen" starten/beenden

Die Funktion wird mit zwei Relais umgesetzt, deren Kontakte den Regler steuern. Beim Samovar-Befehl "Heizung einschalten"

schalten die Kontakte von Relais Nr. 1 den RM ein. Relais Nr. 4 schaltet den Modus "Anheizen" nur ein, wenn der Algorithmus des gewählten Modus es verlangt.

Relais Nr. 4 öffnet seine Kontakte, und der RM beendet den Anheizmodus. Ist der Durchgang beendet, schaltet Samovar den RM über die Kontakte von Relais Nr. 1 ab.

Hinweis:

- Alle genannten Regler arbeiten als eigenständige Geräte und können mit Samovar auch ohne UART-Verbindung nach dem Prinzip und Algorithmus des RM-Reglers verwendet werden.

Samovar und der Spannungsregler zeigen unterschiedliche Leistungswerte. Wie behebe ich das?

Im Großen und Ganzen ist die absolute Leistung nicht so wichtig, aber gleiche Leistungswerte an Regler und Samovar lassen sich erreichen, und im Grunde gibt es dafür nur einen Weg.

Keiner der Regler, Samovar eingeschlossen, misst die Leistung. Sie berechnen sie wie ein Taschenrechner aus zwei Werten: der aktuell am Heizelement anliegenden Spannung und dem eingegebenen Widerstandswert dieses Heizelements.

Der eingegebene Widerstand ist konstant, die Spannung ändert sich. Die ganze Einstellung läuft auf die Wahl des Heizelementwiderstands hinaus.

Sie können den berechneten Widerstand nehmen: Spannung zum Quadrat geteilt durch die Leistung. Die Spannung steht auf dem Typenschild des Heizelements. Beispiel für ein Heizelement mit 3 kW und 220 Volt: $220 \cdot 220 / 3000 = 16.133333333333$ Ohm.

Oder messen Sie ihn mit einem Multimeter.

Für die Regler **RMVK** und den einfachen Regler auf Basis chinesischer Module (**Stab-avr**).

Achtung! Wenn die Brennblase nicht wirklich beheizt werden soll, klemmen Sie das Heizelement ab.

- Im **RMVK** können Sie Leistung und Spannung eingeben, der Widerstand wird automatisch berechnet. Oder Sie geben den Widerstand ein, und die Leistung wird berechnet. [Video zur Einrichtung](#). Sehen Sie es an, machen Sie es nach und vergessen Sie nicht, den Regler auf Anzeige in Watt umzustellen.

- Stellen Sie beim **Stab-avr** mit dem Potentiometer die momentane tatsächliche Netzspannung ein und geben Sie die Leistung Ihres Heizelements ein.

Achtung! Die Leistung des Heizelements muss beim Stab-avr für eine Netzspannung von 230 Volt eingegeben werden. Hat der Hersteller die Leistung im Datenblatt für 220 Volt angegeben, müssen Sie sie auf 230 Volt umrechnen.

Starten Sie Samovar – Heizung. Im Modus Anheizen etwas einzustellen ist sinnlos – die Spannung ändert sich ständig und mit ihr die Leistung. Geben Sie von Samovar aus den Befehl, am Regler eine bestimmte Spannung einzustellen (etwa 170 Volt beim RMVK oder 2000 Watt beim Stab-avr). Die Regler wechseln in den Stabilisierungsmodus.

Vergleichen Sie die in Samovar und am Regler angezeigten Leistungswerte.

Jetzt gibt es zwei Wege – entweder Sie ändern den Widerstandswert in Samovar (Weboberfläche: Seite HAUPTSEITE - Einstellungen («ГЛАВНАЯ - Настройка») - Allgemein («Основные») - Registerkarte Sonstiges («Прочие»)), was deutlich bequemer ist. Oder Sie ändern den im Regler eingegebenen Leistungswert. Ziel ist, dass die Werte am Regler und in Samovar möglichst nah beieinanderliegen.

Der Regler **Kvic** .

Er hat keine Leistungsanzeige. Deshalb wird nur Samovar auf die Leistung eingestellt, die Sie sehen möchten...

Das ist die erste Möglichkeit

Die zweite ist eine Kette: ein separates [Spannungs- und Strommessgerät](#) – Regler – Samovar.



Вольтметр Амперметр переменного тока 50-300 в 20 A, измеритель мощности KWS, цифровой электронный измеритель мощности, ваттметр переменного тока, Электрический измеритель с функцией сброса 0-а

★★★★★ 4,8 265 отзывов 445 продано

[Характеристики](#) [Описание](#)

Цвет: 100A-Close



Название бренда	Interest Maker
Номер модели	AC 50-300V 20A 100A Voltmeter Ammeter KWS Power Energy Meter LED Digi
Происхождение	Китай
Сертификация	Евротест (CE), RoHS
Размеры	84.6*44*22.3mm
Товары для ремонта	ELECTRICAL

Das ist praktisch, wenn ab und zu verschiedene Heizelemente verwendet und/oder die Heizer des Heizelements selbst umgeschaltet werden.

Das Spannungs- und Strommessgerät misst den aufgenommenen Strom tatsächlich und zeigt die echte Leistung auf seinem Display an. Wenn Samovar also diese Leistung anzeigen soll, ändern Sie einfach im laufenden Betrieb den Widerstandswert des Heizers in Samovar.

Siehe auch:

- Hilfe bei der Auswahl: [Übersicht der Regler](#)
- Neben der Steuerung von Reglern kann Samovar [zusätzlich ein Zusatzheizelement steuern](#)
- Wie Sie den Regler ohne Kabel an Samovar anschließen: [über Funk](#)

Übersicht der Regler

Diese Übersicht hilft Ihnen bei der Wahl eines Reglers für den Betrieb mit Samovar.

Alle unten vorgestellten Regler sind Stabilisatoren der Ausgangsspannung. Sie liefern am Ausgang unabhängig von Netzschwankungen eine stabilisierte Spannung. Sie erhöhen die Spannung nicht. Stabilisiert wird nur, wenn die Ausgangsspannung niedriger als die Eingangsspannung ist. Sie können als eigenständige Geräte arbeiten.

Achtung.

In dieser Übersicht werden Vor- und Nachteile der Regler ausschließlich im Hinblick auf den Betrieb mit Samovar betrachtet.

Regler KVIC

Ein Stabilisator auf Basis eines Arduino-Mikrocontrollers. Sie können ihn selbst bauen, wenn Sie die Bauteile und Erfahrung im Aufbau und/oder in der Reparatur elektronischer Geräte haben. Manuelle Bedienung – ein Encoder.

Anzeige: 0,96-Zoll-LCD

Ein fertiges Gerät kann gekauft werden. Bauform, Ausstattung, Garantie und andere Bedingungen werden individuell mit dem Hersteller vereinbart.

Der Quellcode ist in Assembler geschrieben, offen und [verfügbar](#) für eigene Änderungen.

Vorteile:

- Schneller Start, Stabilisierung der Ausgangsspannung. Bis auf 0,1 Volt genau
- Manuelle Bedienung über einen Encoder mit Taste. Einstellschrittweite 0,5 Volt
- Gut ablesbare Anzeige.
- Anzeige der Spannungsstabilisierung.
- Möglichkeit, die Stabilisierungsspannung manuell vorzugeben, auf die der Regler nach Ausführung des Befehls „Anheizen stoppen“ («Стоп разогрев») umschaltet.
- Steuerung über UART oder I2C
- Reparierbar, das Programm lässt sich bei neuen Versionen aktualisieren. Eine alternative Firmware mit einer Portgeschwindigkeit von 9600 Baud kann installiert werden (standardmäßig 38400 Baud)
- Steuerung des Zusatzheizelements über ein separates Leistungsrelais.
- Ein Anschluss für den externen Befehl „Anheizen stoppen“ («Стоп разогрев»)

- Ein Anschluss für den externen Befehl „Notfall“ («Авария»)
- Sanftes Anlegen der Spannung an die Last beim Start und sanftes Wegnehmen beim Abschalten.

Nachteile:

- **Alle Bauteile des Reglers** liegen auf Hochspannung.
- Zum Austausch des Programms muss eventuell der Mikrocontroller aus dem Regler ausgebaut werden.
- Die vorgegebene Stabilisierungsspannung wird beim Ausschalten oder Neustart nicht im Gerätespeicher gesichert.
- Reparatur und Flashen erfordern besondere Kenntnisse.

„Auf Bestellung“ kaufen oder selbst bauen: [Support-Thread im Forum](#).

Regler RMV-k



Ein Stabilisator im 2DIN-Gehäuse für Hutschienenmontage. Manuelle Bedienung und Einstellung – mit Tasten. Informationsanzeige: eine dreistellige Anzeige. Selbst bauen lässt er sich nicht.

Der Stabilisator ist in 6 Versionen und 5 Ausstattungen erhältlich. (zum Zeitpunkt der Erstellung der Übersicht).

Achtung! Für den Betrieb mit Samovar über UART eignen sich nur die Versionen 5 und 6.

Vorteile:

- Lebensdauer des Produkts – 10 Jahre.

- 6 Monate Garantie
- Spannung wird auf 1 Volt stabil gehalten. Einstellschrittweite 1 Volt
- Speicher für 10 vorgegebene Werte
- Der Widerstand des Heizelements wird für die Leistungsberechnung gespeichert
- Kalibrierung des Voltmeters
- Lässt sich auf Anzeige in Watt umstellen.
- Steuerung durch Samovar über UART (nur Versionen 5 und 6).
- Ein Anschluss für den externen Befehl „Notfall“ («Авария») (ab Version 3).
- Ein Anschluss für den externen Befehl „Anheizen stoppen“ («Стоп разгон») (ab Version 4).

Nachteile:

- Für Reparaturen sind besondere Kenntnisse und Geräte nötig.
- **Alle Bauteile des Reglers** liegen auf Hochspannung.
- Die Firmware lässt sich nicht aktualisieren
- Wird der Regler eigenständig betrieben, ist für die Funktionen „Anheizen stoppen“ («Стоп разгон») und „Notfall“ («Авария») eine zusätzliche Stromquelle nötig (eine 9-V-Blockbatterie genügt)
- Umständliche manuelle Programmierung und Eingabe der Werte in den Speicher.
- Eigenheiten bei der manuellen Bedienung (um die Spannung zu senken, muss man sie z. B. erst um 1 Volt erhöhen und dann senken).

Den Regler können Sie beim Entwickler über einen [Support-Thread im Forum](#) kaufen (nur den Regler, ohne Triac usw.)

Komplett montiert im Gehäuse, fertig zum Anschließen des Heizelements und für den Betrieb, gibt es ihn im [Online-Shop](#) (dort gibt es auch einen Bauteilsatz zum Selbstbau).

Regler Stab-avr

Ein einfacher stabilisierter Leistungsregler für Heizelemente aus chinesischen Modulen

Ein Stabilisator auf Basis eines Arduino-Mikrocontrollers. Bedienung mit Tasten. Anzeige: 0,96-Zoll-LCD. Mit minimalen Lötkenntnissen selbst zu bauen. Nicht im Handel erhältlich.

Der Quellcode ist in C++ geschrieben, offen und lässt sich mit der Arduino IDE selbst ändern.

Vorteile:

- Einfacher Aufbau aus fertigen Modulen.
- Vollständig reparierbar.
- Steuerung durch Samovar über UART
- Das Programm lässt sich aktualisieren, offener Quellcode.
- Alle Bauteile des Reglers liegen auf ungefährlicher Spannung, außer den Leistungsbauteilen.
- Kann ohne Spannungstrennung an Samovar angeschlossen werden (die Spannungspegel von Arduino NANO und ESP32 müssen angepasst werden).
- Externer Anschluss „Anheizen stoppen“ («Стоп разгон»)
- Externer Anschluss „Notfall“ («Авария»)
- Kalibrierung des Voltmeters
- Möglichkeit, den Wert der Ausgangsleistung vorzugeben und zu speichern, auf den der Regler nach Ausführung des Befehls „Anheizen stoppen“ («Стоп разгон») umschaltet.
- Die Nennleistung des Heizelements wird gespeichert.

Nachteile:

- Relativ langer Start der Original-Firmware beim Betrieb mit Samovar.
- Empfohlen wird die angepasste Firmware, die mit dem Samovar-Sketch mitgeliefert wird.

So bauen Sie ihn selbst: [Support-Thread im Forum](#).

Regler RM (RM2M, RM2M-Pro)



Ein Stabilisator im 3DIN-Gehäuse für Hutschiennenmontage. Manuelle Bedienung und Einstellung – mit Tasten. Informationsanzeige: eine dreistellige Anzeige. Selbst bauen lässt er sich nicht.

Der Stabilisator wird von verschiedenen Herstellern angeboten. Entsprechend unterscheiden sich Ausstattung und Garantie. Sie können entweder nur den Stabilisator (ohne Triac usw.) oder ein komplettes, betriebsbereites Gerät kaufen.

Vorteile:

- Die eingestellte Spannung wird auf plus/minus 1 Volt stabil gehalten.
- Einstellschrittweite 1 Volt
- Anzeige der von der Last aufgenommenen Leistung: von 0 bis 9,99 kW.
- Speicher für 10 Spannungsvorgaben.
- Eingebauter Spannungs-Zeit-Timer: von 0 bis 999 Minuten.
- Braucht für die Funktionen „Anheizen stoppen“ («Стоп разгон») und „Notfall“ («Авария») keine zusätzliche Stromquelle

Nachteile:

- Es gibt keinen Steuerkanal.
- Umständliche Programmierung und Eingabe der Werte in den Speicher.

Fazit.

Für den Einsatz mit Samovar werden die ersten drei Regler empfohlen. Wird die Steuerung über UART genutzt, steuert Samovar den Regler vollständig: vom Start des Anheizens über die Änderung der Spannung zu jedem Zeitpunkt bis zum Abschalten der Heizung.

Die Notabschaltung durch ein fremdes Gerät kann optional genutzt werden. Zum Beispiel beim Einsatz von Samovar mit einem PVK (Brennblase mit Dampfmantel), um die Heizung abzuschalten, wenn der Druck im Dampf-Wasser-Mantel überschritten wird. Als Sensor kann ein Kontaktmanometer dienen.

Der Regler **KVIC**. Bequeme manuelle Bedienung; mit der Einstellschrittweite von 0,5 Volt lässt sich die Kolonne recht genau in den Vorfluten-Modus bringen und lange in diesem Zustand halten. Bauen kann ihn aber nur ein erfahrener Elektroniker.

Der Regler **RMV-k**.

Ein stabil arbeitendes Gerät mit recht großer Auswahl an Ausstattungen. Zuverlässig; soweit bekannt, gibt es praktisch keine Beschwerden. Über UART an Samovar angeschlossen, ist er bequem und einfach zu steuern. Sie können ein komplett betriebsbereites Gehäuse zusammen mit den Sensoren „Anheizen stoppen“ («Стоп Разгон») und „Notfall“ («Авария») kaufen. Die Sensoren sind für den eigenständigen Betrieb des Reglers nutzbar und sind Bimetall-Thermoschalter KSD9700. Beim Anschluss an Samovar wird der Sensor „Anheizen stoppen“ nicht verwendet, da der Anschluss für die Datenleitung belegt ist – Samovar steuert den Modus „Anheizen stoppen“ dann selbst. Der Sensor „Notfall“ kann als eigenständige Option verwendet werden. Für Samovar selbst eignen sich die Sensoren nicht, sie können aber als Bauteile für eine zusätzliche [Schutzstufe](#) verwendet werden.

Auch der Regler **Stab-avr** steht ihm in nichts nach. Dass man ihn selbst bauen und reparieren kann, macht ihn sehr attraktiv. Eine Besonderheit dieses Reglers: Die Leistungsvorgabe wird in Watt angezeigt (und nicht in Volt wie bei den oben besprochenen Reglern).

Die Einstellschrittweite beträgt 1/200 der Nennleistung des Heizelements. Wegen der Art der Leistungsvorgabe kann der tatsächliche Wert leicht vom Sollwert von Samovar abweichen (zum Beispiel 1803 W statt 1800). Das ist völlig unkritisch.

Alle Vorteile der Regler **RM** (RM2M, RM2M-Pro) werden durch einen großen Nachteil zunichtegemacht: die fehlende UART-Verbindung. Und damit die Unmöglichkeit, die Spannung am Regler "von außen" vorzugeben. Die Stabilisierungsspannung lässt sich nur manuell einstellen. Beim Betrieb mit dem RM kann Samovar den Regler nur ein-/ausschalten und das Anheizen ein-/ausschalten. Denn dieser Regler bietet keine anderen Steuermöglichkeiten durch fremde Geräte.

Empfehlungen.

Wenn Sie auch nur ein wenig Erfahrung mit dem Lötkolben haben, lohnt es sich, den Regler **Stab-avr** selbst zu bauen.

Wenn Sie ernsthafte Zweifel an den eigenen Fähigkeiten oder keine Zeit haben, lohnt es sich, den Regler **RMV-k** komplett montiert zu kaufen.

Regler an Samovar anschließen

Kabelverbindung.
Regler Kvic und RMVK

Der KVIC-Regler verwendet einen ADUM1201 zur Spannungstrennung. Samovar kommuniziert mit ihm mit 38400 Baud oder mit 9600 Baud, wenn **KVIC_USE_9600** aktiviert ist.

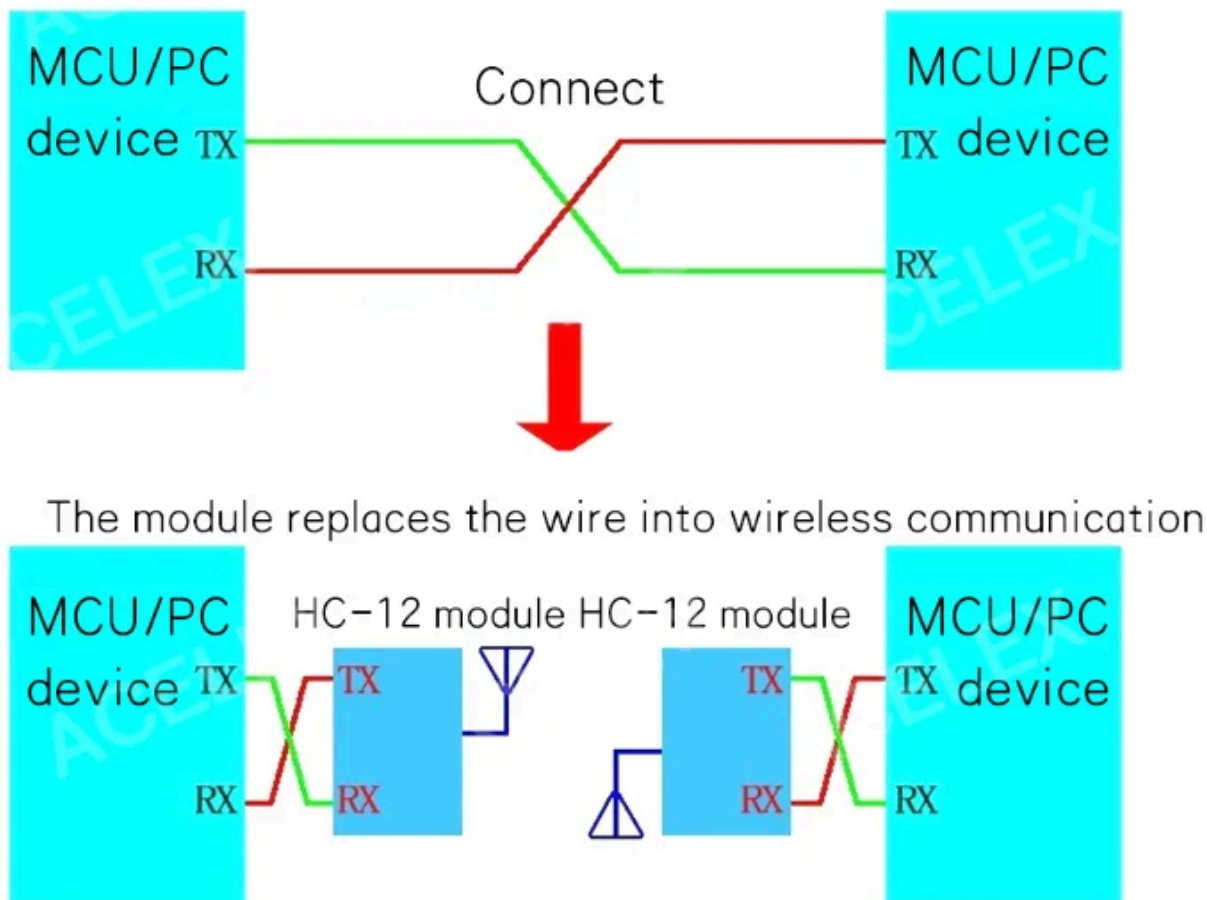
Der RMVK-Regler verwendet zur Trennung Optokoppler PC817. Die Portgeschwindigkeit beträgt 9600 Baud.

Diese Regler werden auf die gleiche Weise an Samovar angeschlossen. Rx des einen Geräts muss mit Tx des anderen verbunden werden.

Auf der Seite von Samovar wird UART2 verwendet (auf der Leiterplatte ist das der Anschluss UART): RX D16 und TX D17 beim ESP32 DevKit, RX GPIO25 und TX GPIO4 beim LILYGO T-Relay 4, RX GPIO18 und TX GPIO17 beim ESP32-S3. Auch der Regler SEM_AVR (Stab-avr) wird an diesen Port angeschlossen, mit 9600 Baud.

Regler drahtlos an Samovar anschließen

Ein Samovar mit einem über UART (COM-Port) gesteuerten Regler lässt sich ohne Kabel mit zwei Funkmodulen (HF-Modulen) HC-12 verbinden. Das ist sehr einfach; schließen Sie sie einfach wie im Schaltplan gezeigt an:



Der ADUM wird hier nicht benötigt, da die Module für Spannungen von 3,1 V bis 5 V ausgelegt sind und daher sowohl an den ESP32 (Samovar) als auch an einen Arduino Nano angeschlossen werden können.

Laut Hersteller beträgt die Reichweite dieses Moduls bis zu 1 km (im freien Gelände und mit externer Antenne), was für unseren Zweck mehr als genug ist. Das Modul ist ab Werk auf 9600 bps eingestellt. Wenn Sie einen über UART gesteuerten Regler STAB-AVR oder RMVK verwenden, schließen Sie einfach ein Modul an Samovar und das andere an den Regler an, wie im Schaltplan gezeigt, und alles sollte funktionieren. Bei einem Kvic-Regler müssen Sie beide Module konfigurieren. Das ist nicht schwer, Sie müssen sie aber an einen Computer anschließen, um die Geschwindigkeit zu ändern, mit der die Module untereinander Daten austauschen.

Beide HC-12-Module müssen auf dieselbe Geschwindigkeit eingestellt sein wie Samovar und der Regler. Für RMVK und SEM_AVR sind das 9600 Baud. Für KVIC verwenden Sie 38400 Baud oder 9600 Baud, wenn **KVIC_USE_9600** im Build aktiviert ist.

Das Modul hat viele Optionen, uns interessiert aber nur eine: das Ändern der Portgeschwindigkeit des Moduls.

Um das Modul in den Einstellmodus zu versetzen, verbinden Sie den Pin SET mit GND. Versorgen Sie das Modul dann mit Strom und schließen Sie es über einen UART-USB-Adapter an einen Computer an (dafür kann ein Arduino Nano dienen). Wenn Sie 38400 Baud gewählt haben, senden Sie den Befehl **AT+b38400** an den erkannten COM-Port und warten Sie auf die Antwort **OK+b38400**. Bei 9600 Baud müssen Sie die Werkseinstellung des HC-12 nicht ändern.

Trennen Sie SET von GND, schalten Sie die Versorgung ab, schließen Sie die Module an Samovar und den Regler an, und alles sollte funktionieren.

Achtung!

Der Kvic-Regler hat keine galvanische Trennung von der Netzspannung; selbst die Antenne des HC-12 liegt auf Netzpotenzial! Das Modul muss daher im Gehäuse des Reglers untergebracht oder anderweitig zuverlässig isoliert werden.

Heizelement

Sie können beliebige Heizelemente verwenden, die Ihr Spannungsregler steuern kann. Hier betrachten wir den gängigsten Typ.

Im Russischen steht die Abkürzung ТЭН (Heizelement) für "Rohrheizkörper". Ein solches Bauteil enthält also so viele Heizer, wie es Rohre hat. Teilen Sie die Gesamtleistung des Heizelements durch die Anzahl der Rohre, erhalten Sie die Leistung eines Rohrs. Alle Rohre haben die gleiche Leistung.



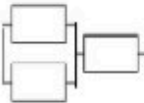
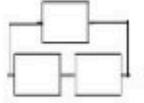
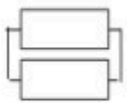
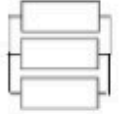
In der Regel wird ein Modell mit 3 kW gewählt. Jedes Rohr hat dann 1 kW. Für 1,5"-Kolonnen ist das eine gute Wahl, denn eine 1,5 Meter hohe Kolonne braucht für den Betrieb im Vorfluten-Modus 1,2–1,4 kW. So bleibt eine Leistungsreserve.

Sie können also eine kombinierte Schaltung verwenden:

- Schließen Sie zwei Rohre über den Regler an; das ergibt 2 kW. Bei der Rektifikation hält der Regler die Ausgangsspannung im Bereich von 170–180 Volt. Das reicht völlig aus, selbst wenn die Netzspannung "einbricht".
- Schließen Sie das verbleibende Rohr über ein Halbleiterrelais an und steuern Sie dieses Relais über die Kontakte von Relais Nr. 4. Das ist dann das Zusatzheizelement.

Für 2"-Kolonnen reicht ein 3-kW-Heizelement bei dieser Schaltung nicht aus. Ein 4,5-kW-Element ist genau richtig.

Durch Umschalten der Rohranschlüsse können Sie deutlich leistungstärkere Heizelemente verwenden, die dann schonender arbeiten. Halten Sie sich natürlich an das Prinzip der vernünftigen Bemessung. Unten finden Sie eine Tabelle möglicher Schaltungen der Heizer im Bauteil.

	1	2	3	4	5	6	7
ТЭНБ							
3 kw	0,33	0,5	0,63	1	1,5	2	3
4,5kw	0,5	0,75	1	1,5	2,25	3	4,5
6kw	0,67	1	1,33	2	3	4	6
9kw	1	1,5	2	3	4,5	6	9
12kw	1,3	2	2,67	4	6	8	12

So lesen Sie die Tabelle – am Beispiel eines 4,5-kW-Heizelements

Wählen Sie in der linken Spalte die Leistung – 4,5 kW. Die obere Zeile zeigt anhand der Nummer das Schaltschema der Heizer.

Nummer 1 ist zum Beispiel die Reihenschaltung aller drei Heizer. Am Schnittpunkt von Zeile und Spalte steht die Leistung, die das Bauteil bei 220 Volt abgibt = 0,5 kW, also nur 500 Watt.

Schaltung Nr. 6 ergibt 3 kW an 2 Heizern und lässt einen Heizer frei – er kann als Zusatzheizelement dienen.

Einsatz eines Zusatzheizelements

Samovar kann ein zusätzliches Zusatzheizelement steuern, und zwar in den Modi "[Rektifikation](#)" («Ректификация»), "[Destillation](#)" («Дистилляция») und "[Bier](#)" («Пиво»)

Um diese Funktion zu nutzen, schließen Sie an die Schließerkontakte von Relais Nr. 4 entweder ein Schütz oder ein Halbleiterrelais an, dessen Laststrom für den zweiten Heizkörper ausreicht.

Betrieb des Zusatzheizelements im Modus „Rektifikation“

Für den Einsatz des Zusatzheizelements müssen Sie im Sketch nichts ändern. Diese Funktion ist standardmäßig aktiviert.

Die unten genannten Schwellentemperaturen werden im Konfigurator auf der Registerkarte „Temperaturen“ («Температуры») eingestellt („Temperatur für das Ende des Aufheizens, °C“ («Температура перехода из разгона, °C»), „Temperatur für das Öffnen der Kühlung, °C“ («Температура открытия охлаждения, °C»), „Schwelle des Zusatzheizelements, °C“ («Порог разгонного ТЭНа, °C»)); beim Kompilieren in der Arduino IDE werden sie mit denselben Direktiven in [Samovar_ini.h](#) festgelegt.

Die Logik ähnelt der des [PM](#) im Modus „Aufheizen“ («Разгон»):

Beim Befehl „Heizung einschalten“ («Включить нагрев») schaltet Samovar nicht nur das Hauptheizelement ein, sondern Relais Nr. 4 schaltet auch das Schütz oder das Halbleiterrelais ein, das wiederum Spannung an das Zusatzheizelement legt. Wenn die Dampftemperatur den in den Einstellungen festgelegten Wert erreicht (standardmäßig 39°):

```
//Значение температуры датчика пара, при котором колонна перейдет из режима  
разгона в рабочий режим  
#define CHANGE_POWER_MODE_STEAM_TEMP 39
```

Relais Nr. 4 öffnet seine Kontakte und schaltet den Zusatzheizkörper ab.

Bei einem über UART gesteuerten Regler (KVIC, RMV-K, SEM_AVR) gibt es zwei weitere Bedingungen: Relais Nr. 4 ist nur eingeschaltet, solange der Status „Aufheizen“ («Разгон») aktiv ist und die Temperatur in der Brennblase nicht über **OPEN_VALVE_TANK_TEMP** liegt (standardmäßig 77°, „Temperatur für das Öffnen der Kühlung“). Sobald die Brennblase sich über diesen Wert erwärmt, wird das Zusatzheizelement abgeschaltet, auch wenn der Dampf 39° noch nicht erreicht hat. Ist der Brennblasensensor nicht angeschlossen, steuert Samovar das Zusatzheizelement nicht. Ohne UART-Regler schaltet Relais Nr. 4 beim Start der Heizung zusammen mit Relais Nr. 1 ein und nach dem Ende des Aufheizens ab.

Wurde der Befehl „Heizung einschalten“ nicht gegeben oder liegt die Sensortemperatur über dem im Sketch festgelegten Wert, schaltet sich das Aufheizen nicht ein.

Betrieb im Modus „Destillation“. Der Ausgang für das Zusatzheizelement schaltet beim Start der Heizung ein. Nach Siedebeginn schaltet er ab, wenn in den Einstellungen das Kontrollkästchen

„Zusatzheizelement beim Kochen verwenden“ («Использовать разгонный тэн при кипении») deaktiviert ist („Einstellungen“ («Настройки») → Registerkarte „Bier“ («Пиво»)); ist es aktiviert, bleibt der Ausgang bis zum regulären Ende der Heizung eingeschaltet.

Betrieb des Zusatzheizelements im Modus „Bier“

Das Heizelement wird genauso angeschlossen wie oben beschrieben.

Der Algorithmus ist jedoch ein anderer

Das Zusatzheizelement wird beim Maischen über den Parameter „Schwelle des Zusatzheizelements, °C“ gesteuert (Registerkarte „Temperaturen“ des Konfigurators); in [Samovar_ini.h](#) ist es diese Zeile:

```
#define ACCELERATION_HEATER_DELTA 4
```

//Если разница между целевой температурой и текущей температурой (для режимов пиво и су-вид) больше заданной, то при использовании регулятора, управляемого по UART дополнительно Реле № 4, будет включать разгонный тэн, при разнице меньше, разгонный тэн отключится

Standardmäßig aktiviert, 4 ist der Wert in Grad C°. Funktioniert nur mit einem über UART gesteuerten Regler. Ohne einen solchen Regler schaltet Relais Nr. 4 immer dann zusammen mit der Hauptheizung ein, wenn die Temperatur unter dem Zielwert liegt, und schaltet mit ihr ab.

Beim Einschalten weist Samovar den Regler an, volle Leistung auf das Hauptheizelement zu geben. Liegt die Temperatur in der Brennblase 4° unter der in der Zeile des Maischprogramms festgelegten, schaltet Samovar Relais Nr. 4 ein, das wiederum das Zusatzheizelement einschaltet. Ist die Differenz kleiner als 4°, schaltet das Zusatzheizelement nicht ein, und es heizt nur der Hauptheizkörper.

Wenn die nächste Temperaturrast abgeschlossen ist oder der Bediener den Befehl „Nächstes Programm“ («Следующая программа») gibt, vergleicht Samovar die Temperatur in der Brennblase mit der in der nächsten Programmzeile festgelegten. Ist die Differenz größer als 4°, wird das Zusatzheizelement zugeschaltet.

Wurde der Befehl „Heizung einschalten“ nicht gegeben oder ist die Differenz kleiner als die eingestellte, schaltet sich das Aufheizen nicht ein.

Während die Würze kocht, ist das Zusatzheizelement eingeschaltet, wenn das Kontrollkästchen „Zusatzheizelement beim Kochen verwenden“ aktiviert ist, sonst ist es aus.

Samovar 7.00: Im Modus „Käseherstellung“ («Сыроварение») nutzt das Ventil für den Salzlakenablass Relais Nr. 4. Schließen Sie daran entweder ein Käseventil oder ein Zusatzheizelement für die anderen Modi an, aber nicht beide Geräte gleichzeitig. In den Modi „BK“ («БК»), „NBK“ («НБК») und „Sous-vide“ («Су-вид») gibt es keine eigene Logik für das Zusatzheizelement: Mit einem UART-Regler wird Relais Nr. 4 darin nicht eingeschaltet.

Schütz. FI-Schutzschalter. Notabschaltung

Zusätzlicher Schutz für Störfälle.

Der Aufbau wird empfohlen.

Der vorgeschlagene Schutz arbeitet unabhängig und schaltet die Anlage in kritischen Störfällen ab.

Die Hauptgefahr ist ein Durchbruch von Alkoholdämpfen. Wenn aus irgendeinem Grund einer der folgenden Fälle eintritt:

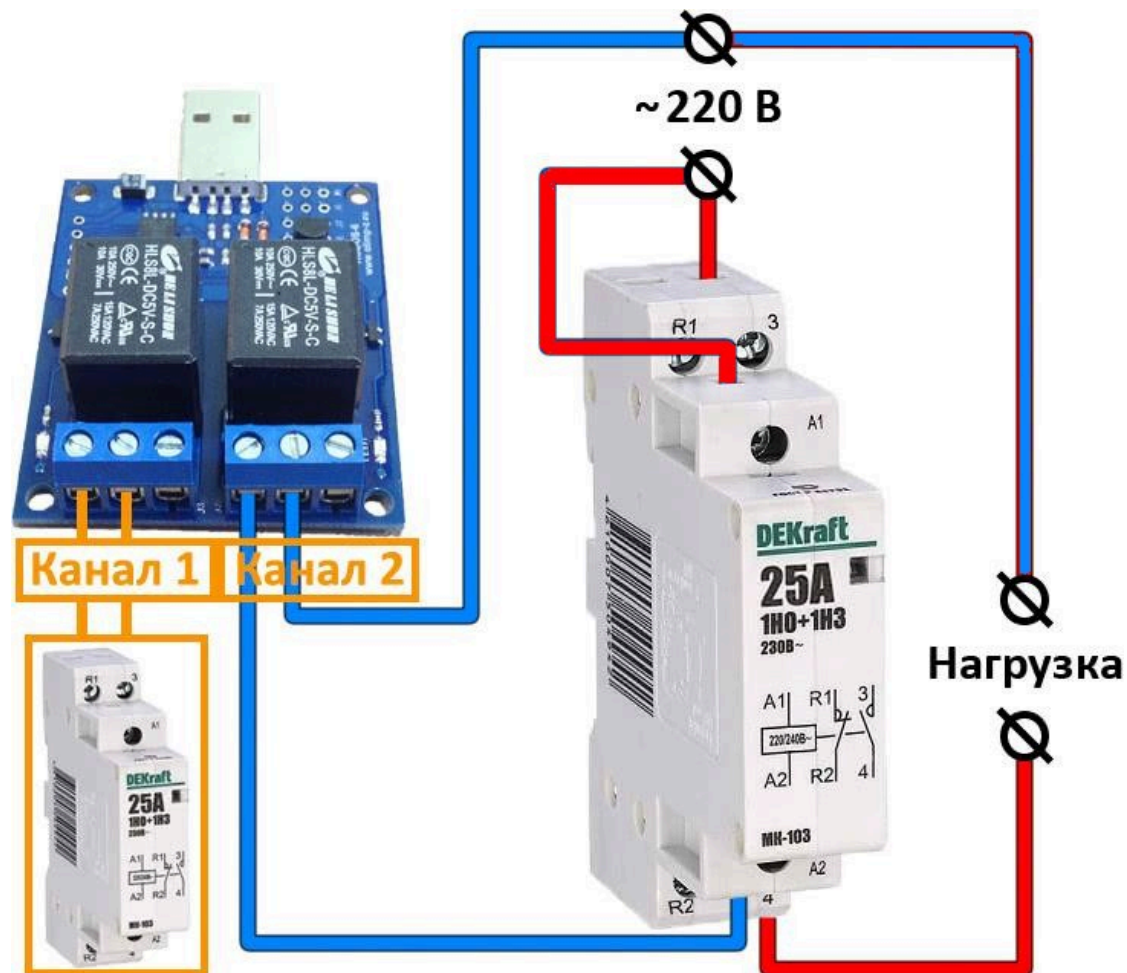
- Der Triac fällt aus (Kurzschluss).
- Der Spannungsregler hängt sich auf oder arbeitet fehlerhaft.
- Die Automatik fällt aus.
- Das 12-V-Netzteil fällt aus.
- Die Kühlung fehlt.

Dann müssen Sie die Heizung abschalten und den Dampfaustritt stoppen. Überwacht wird am TCA (Atmosphärenrohr).

Variante 1

Gilt, wenn zusammen mit Samovar ein Schütz verwendet wird.

Anschluss des Schützes an das [Relaismodul](#)



Sie brauchen einen Bimetall-Thermoschalter KSD9700 NC (Öffner, normalerweise geschlossen) mit einer Auslösetemperatur von 55–70 °C. Am besten mit isoliertem Gehäuse: Keramik oder Kunststoff.

So bauen Sie es auf:

Die Minusspannung vom 12-V-Netzteil wird der Samovar-Platine nicht direkt, sondern über den KSD9700 zugeführt, der am TCA befestigt ist. Am bequemsten geht das über einen zweipoligen Steckverbinder am Samovar-Gehäuse und ein Kabel der nötigen Länge.

So funktioniert es.

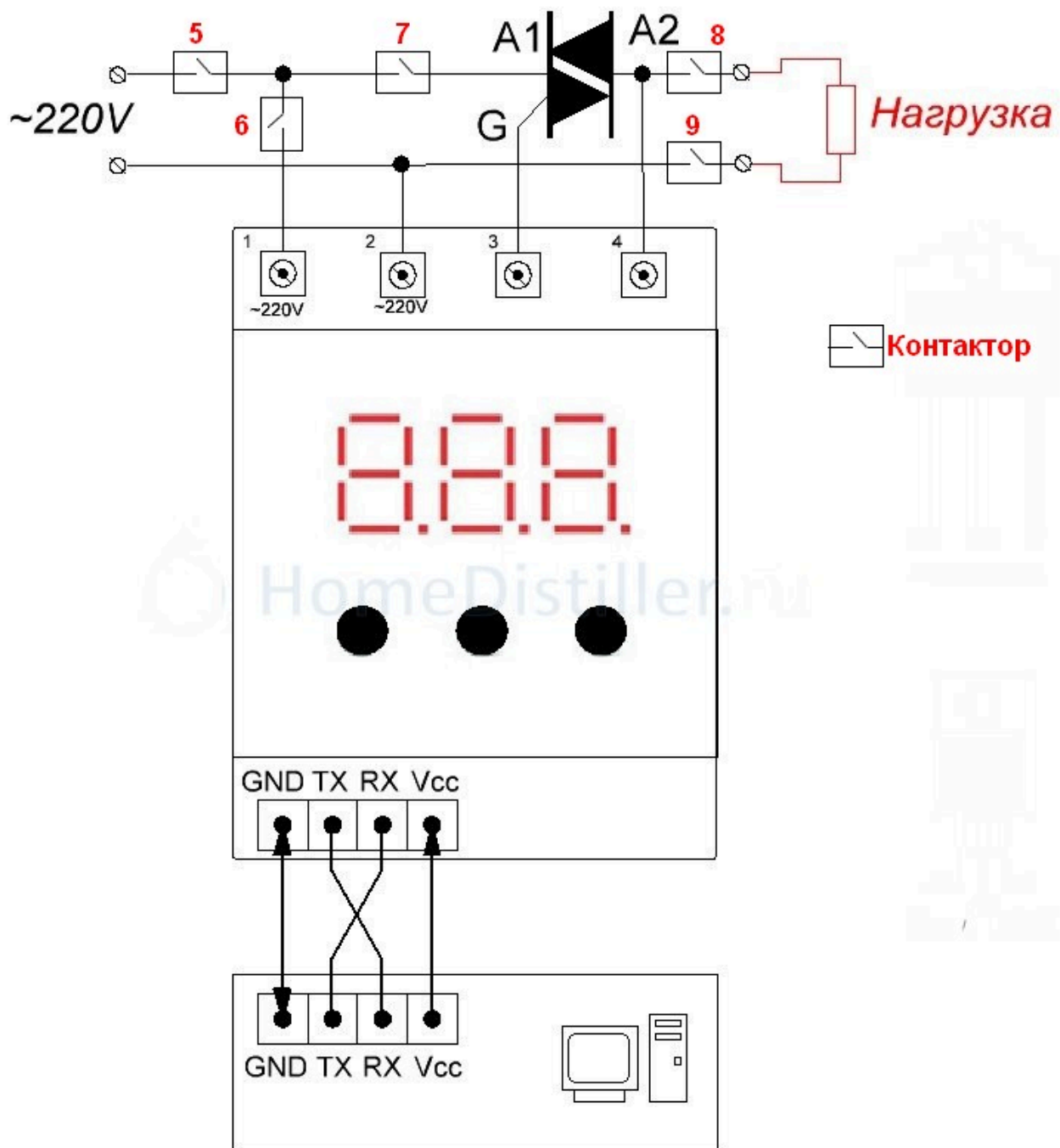
Tritt ein Notfall ein, den Samovar nicht beherrschen kann (Verbindung zum Regler verloren, Triac ausgefallen usw.), beginnt Dampf durch das TCA durchzubrechen. Der Thermoschalter erwärmt sich und unterbricht den Stromkreis. Samovar schaltet ab, das Steuerrelais lässt das Schütz abfallen, und das Schütz trennt seinerseits die Heizung. Der Dampfaustritt hört auf. Wird ein Wasserventil verwendet, schließt es.

Nach einiger Zeit kühlt das TCA ab, der Thermoschalter schließt seine Kontakte, und Samovar schaltet sich ein. Alle anderen Geräte schalten sich aber ohne Ihren Befehl nicht ein.

Klären Sie natürlich zuerst, warum die Abschaltung erfolgt ist.

Und natürlich: Vergessen Sie beim Anschließen von Samovar an die Kolonne, den Steckverbinder einzustecken, schaltet Samovar nicht ein, weil keine Spannung anliegt – das entspricht einem ausgefallenen Netzteil.

Schema möglicher Einbauorte des Schützes



Punkt 5. Das Schütz schaltet ein und versorgt den Regler und gleichzeitig den Triac, der auf Befehl des Reglers Spannung an die Last durchlässt. Das Schütz wird durch den Befehl „Heizung einschalten“ («Включить нагрев») von Samovar eingeschaltet.

Entsprechend braucht der Regler etwas Zeit zum Starten, erst danach erhält er von Samovar den Befehl „Anheizen“ («Разгон») und legt Spannung an die Last. Diese Verzögerung legt der Parameter

SAMOVAR_USE_POWER_START_TIME fest (standardmäßig 2000 ms; im Konfigurator ist das „Startverzögerung des Reglers, ms“ («Задержка запуска регулятора, мс»)) auf der Registerkarte „Regler“ («Регулятор»)), bei SEM_AVR kommen weitere 3 s hinzu. Antwortet der Regler in dieser Zeit nicht, schaltet Samovar die Heizung mit der Meldung „Zeitüberschreitung des Reglerbefehls“ («Таймаут команды регулятора») ab.

Wägen Sie alle Vor- und Nachteile ab, bevor Sie das Schütz an dieser Stelle einbauen.

Punkt 6. Dringend nicht empfohlen. Ein Schütz an dieser Stelle schaltet im Grunde nur den Regler ab, sonst nichts. Der Regler selbst verbraucht so gut wie nichts, und wenn Sie nur den Regler abschalten müssen, genügt ein Schalter für kleine Leistung.

Bei dieser Schaltung heizt das Heizelement weiter, wenn sich der Regler aufhängt, der Triac ausfällt usw.

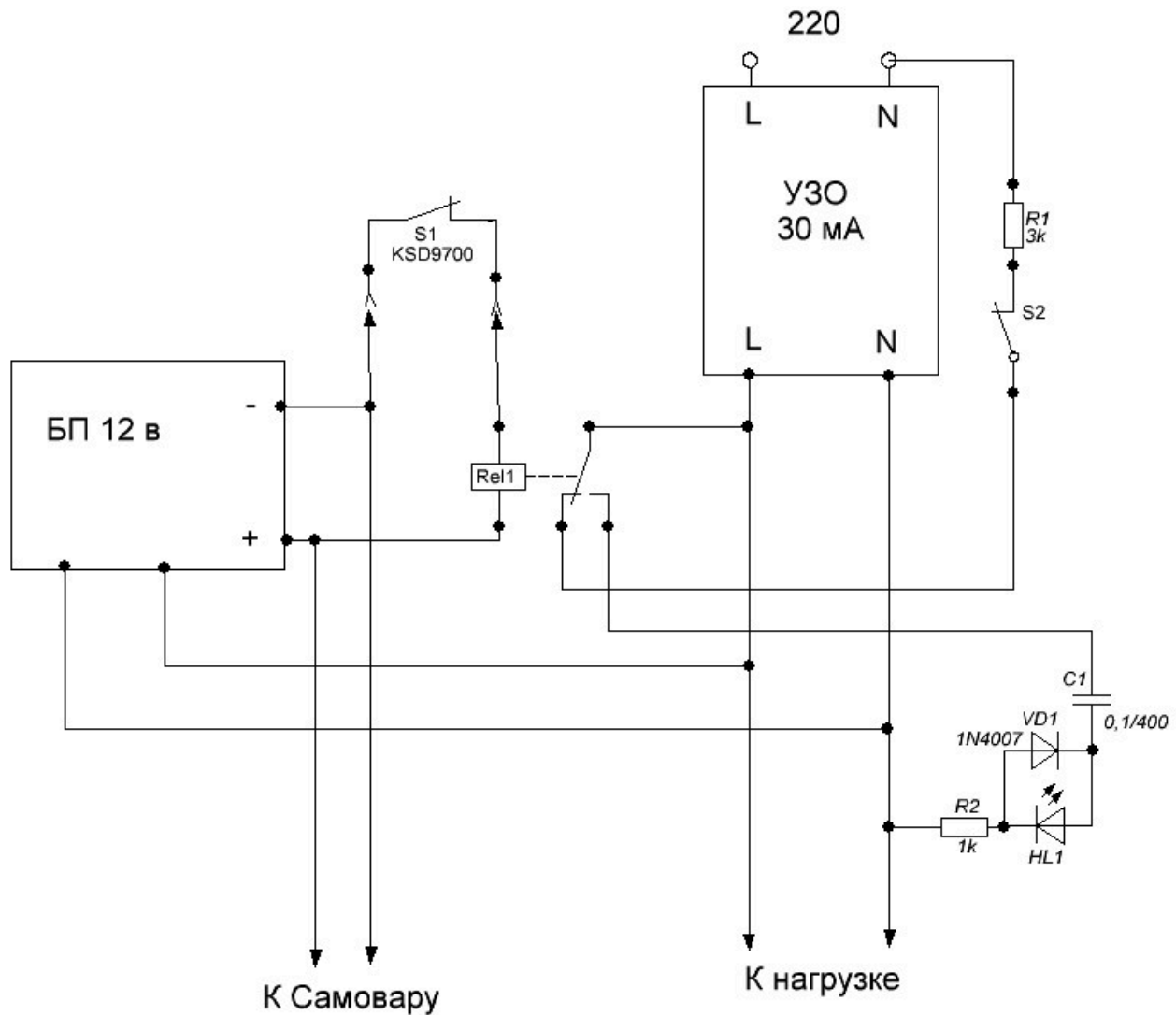
Punkte 7 und 8. Beim Anlegen der Hauptversorgung schaltet der Regler ein. Er startet so lange, wie er braucht. Sie können ihn einstellen, Spannungen vorgeben usw. An die Last gelangt aber nichts, weil das Schütz den Versorgungskreis zum Triac unterbricht. Erst wenn Samovar den Befehl „Heizung einschalten“ («Включить нагрев») gibt, schließt das Schütz, und die Spannung gelangt über den Triac an die Last.

Punkt 9 - für den Einbau empfohlen. Mit dem Schütz an dieser Stelle und geschlossenem Triac ist das Heizelement vollständig spannungsfrei.

Variante 2

Statt eines Schützes können Sie einen FI-Schutzschalter (Fehlerstrom-Schutzschalter) verwenden, der zusätzlich vor Spannungsableitung auf das Gehäuse der Anlage schützt.

Der Sensor S1 (KSD9700, Öffner) wird am TCA angebracht.



So funktioniert es.

Um alles einschließlich Samovar einzuschalten, drücken und halten Sie den Öffner-Taster S2. Eingeschaltet wird über den FI-Schutzschalter selbst. Nach dem Anlegen der Spannung startet das 12-V-Netzteil und versorgt die Spule von Relais Rel1. Die Relaiskontakte (NO) schließen, und die LED HL1 leuchtet. Jetzt kann der Taster S1 losgelassen werden. Die Überwachung ist bereit. Leuchtet die LED nicht und Sie lassen den Taster los, löst der FI-Schutzschalter aus.

Betrachten wir die möglichen Situationen:

- Der Bediener hat vergessen, den Sensor S1 anzuschließen – das Relais schaltet nicht ein – der FI-Schutzschalter löst aus
- Der Triac fällt aus – Dampf durch das TCA – Sensor S1 unterbricht den Stromkreis – das Relais fällt ab – der FI-Schutzschalter löst aus

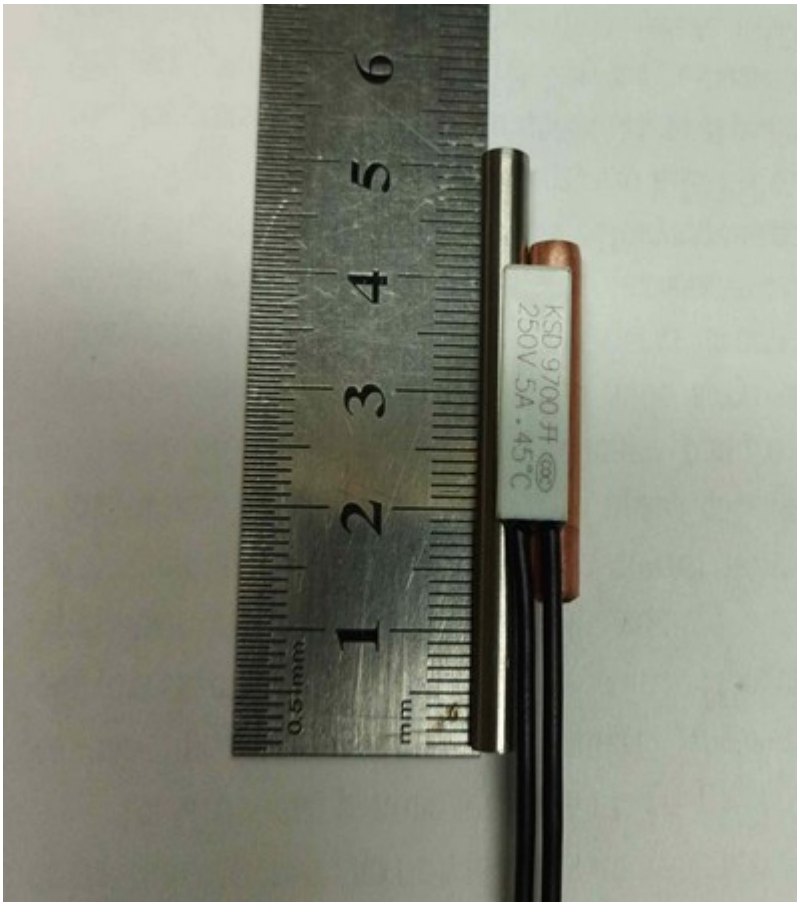
- Unzureichende Kühlung – Dampf durch das TCA – Sensor S1 unterbricht den Stromkreis – das Relais fällt ab – der FI-Schutzschalter löst aus
- Das 12-V-Netzteil fällt aus – das Relais fällt ab – der FI-Schutzschalter löst aus
- Die Automatik und/oder der Regler hängt sich auf – Dampf durch das TCA – Sensor S1 unterbricht den Stromkreis – das Relais fällt ab – der FI-Schutzschalter löst aus.
- Kurzer Stromausfall – bei Rückkehr der Spannung löst der FI-Schutzschalter aus.

Beispiel für den Aufbau der Überwachungseinheit

Die Überwachungseinheit an der TCA-Leitung.

Material: Edelstahlrohr mit 6 mm Durchmesser, Kupferrohr mit 6,1 mm Durchmesser (vorausgesetzt, der 18B20-Sensor hat keine Hülse; hat er eine Hülse, ist das Rohr nicht nötig), ein KSD9700-Sensor, Schrumpfschlauch, Silikonschlauch.

Legen Sie das Edelstahlrohr neben das Kupferrohr (oder neben die Hülse des Temperatursensors). Wählen Sie die Rohrlänge so, dass das Edelstahlrohr ein paar Zentimeter länger ist. Legen Sie den KSD9700-Sensor obenauf und umhüllen Sie das Ganze mit Schrumpfschlauch. Stecken Sie den 18B20-Sensor, der in Samovar als TCA-Sensor zugeordnet ist, in das Kupferrohr. Verschließen Sie zur Sicherheit ein Ende des Kupferrohrs mit einem beliebigen Dichtmittel, damit keine Flüssigkeit eindringt





Bei der Rektifikation setzen Sie das Edelstahlrohr in die Leitung vom TCA des Dephlegmators zum Auffangbehälter für Dampfkondensat ein. Erfahrungsgemäß am besten etwa 30 cm vom TCA entfernt.

Beim Destillieren von Maische und beim Betrieb einer Maischekolonne schließen Sie das Edelstahlrohr direkt an den Produktauslauf an. Das Produkt fließt also durch das Rohr weiter in den Auffangbehälter. Zusätzlich sehen Sie die Produkttemperatur am Auslauf als TCA-Temperatur.

Da Sie die Temperatur des aus dem Kühler austretenden Wassers in Echtzeit regeln können, lässt sich die Pumpe auf die optimale Fördermenge einstellen.



Sie können ganz auf das Kupferrohr verzichten und einen Sensor ohne Hülse direkt an das Edelstahlrohr drücken, müssen dann aber für die Abdichtung des Sensors sorgen, und die Wärmekontaktfläche für den KSD9700 wird ebenfalls kleiner.

Für den korrekten Betrieb muss die Temperatur, bei der Samovar die Heizung anhand des TCA-Sensors abschaltet („Notfalltemperatur TCA, °C“ («Аварийная температура TCA, °C») auf der Registerkarte „Temperaturen“ («Температуры») des Konfigurators; in [Samovar.ini.h](#) ist es die Zeile)

//Maximale TCA-Temperatur, bei der die Leistung abgeschaltet wird

#define MAX_ACP_TEMP 75

5–10° unter der Auslösetemperatur des KSD9700 liegen. Der Standardwert von 75° liegt über der Schwelle des Thermoschalters von 55–70°, er muss also gesenkt werden.

Bricht dann Dampf durch, reagiert zuerst Samovar und gibt den Abschaltbefehl; wird dieser nicht ausgeführt, lösen Schütz oder FI-Schutzschalter aus.

Kühlung

Samovar 7.00: Die Temperatur, bei der die Kühlung öffnet, und die Pumpenparameter werden im Konfigurator auf den Registerkarten „Temperaturen“ («Температуры») und „Pumpen“ («Насосы») eingestellt.

In den Modi Rektifikation und Destillation hält Samovar die eingestellte Kühlwassertemperatur.

Für die Kühlung verwendet Samovar:

1. Relaisausgang Nr. 3: GPIO14 beim ESP32 DevKit und GPIO41 beim ESP32-S3; er steuert das Ventil, das Wasser aus der Leitung zuführt. Oder ein anderes Gerät, das beim Befehl „Wasserzufuhr öffnen“ («Откройте подачу воды») einschalten soll.
2. Den Ausgang zur Pumpensteuerung: GPIO4 beim ESP32 DevKit und GPIO11 beim ESP32-S3; an ihm arbeitet ein PWM-Modulator, der die Drehzahl der Wasserpumpe automatisch ändert. Dieser Ausgang kann auch ein Ventil steuern, das den Wasserdurchfluss erhöht, sofern keine Pumpe verwendet wird.

Der Algorithmus (am Beispiel der Rektifikation, mit dem Original-Sketch). Ist die Temperatur der Brennblass **OPEN_VALVE_TANK_TEMP** erreicht (standardmäßig 77 °C), öffnet das Ventil und die Pumpe läuft an. In den ersten zehn Regelzyklen erhält die Pumpe den Startwert **PWM_START_VALUE** (standardmäßig 40 %), danach hält der Regler die eingestellte Wassertemperatur, ohne unter **PWM_LOW_VALUE** zu gehen.

Die Kühlung wird an zwei Stellen eingestellt:

1. Im Konfigurator (Registerkarten „Temperaturen“ («Температуры») und „Pumpen“ («Насосы»)) oder in der Datei **Samovar_ini.h** – das sind dieselben Parameter, der Konfigurator schreibt sie in diese Datei. Die Zeilen:

```
//Temperatur des Brennblassensors, bei der das Wasserventil öffnet und die  
Wasserpumpe einschaltet
```

```
#define OPEN_VALVE_TANK_TEMP 77
```

Bei einer Brennblassentemperatur von 77° wird der Befehl zum Öffnen des Wasserventils gegeben, gleichzeitig geht das PWM-Steuersignal zum Einschalten der Pumpe raus, und der Bediener erhält die Meldung, dass das Wasser geöffnet werden muss. Die Meldung wird in jedem Fall gesendet, auch wenn weder Pumpe noch Ventil verwendet werden.

```
#define DELTA_T_CLOSE_VALVE 20
```

//Differenz zwischen der eingestellten Kühlwasser-T und der Wasser-T, bis zu der die Pumpe nach Ende des Destillationsprozesses in den Modi Destillation und Rektifikation weiterläuft. Der Wert ist in Grad C°

angegeben und kann geändert werden. Vom Abschalten wird dringend abgeraten. (Details siehe [hier](#))

//Unterer Wert (in Prozent) für den Betrieb des PWM-Reglers der Pumpe. Wählen Sie einen Wert, der den Wasserdurchfluss sicherstellt

#define PWM_LOW_VALUE 10

Bekanntlich braucht ein Motor zum Anlaufen mehr Spannung als zum Weiterlaufen. Diese Zeile legt den kleinstmöglichen Wert in % fest, bei dem der Motor die Wasserpumpe ohne Stillstand weiterlaufen lassen kann.

//Unterer Wert (in Prozent), ab dem der PWM-Regler der Pumpe zu arbeiten beginnt. Wählen Sie einen Wert, bei dem der Motor sicher anläuft

#define PWM_START_VALUE 40

Der Wert in % der Nennspannung, bei dem der Pumpenmotor sicher anläuft.

Wenn Ihre Pumpe also (sofern sie bereits im System eingebaut und mit Flüssigkeit gefüllt ist) bei 4 Volt zuverlässig anläuft und darunter nicht, entspricht das bei einer Nennspannung von 12 Volt 30 % des Nennwerts. Der Wert 40 kann also durch 30 ersetzt werden.

2. In den Einstellungen auf der Registerkarte „Temperatur“ («Температура») – der Sollwert des Wassersensors. Die Pumpe hält diese Temperatur während des Prozesses.

Nach dem Ende der Heizung wird die Kühlung abgeschaltet, wenn das Wasser auf den Sollwert minus **DELTA_T_CLOSE_VALVE** abgekühlt ist und mindestens drei Minuten vergangen sind. Die zweite Möglichkeit: Die zuvor beheizte Brennblass ist unter 70 °C abgekühlt. Im Modus mit TCA-Sensor wird zusätzlich dessen Abkühlung geprüft.

Im Modus **Maischekolonne** steht die automatische Kühlungssteuerung zur Verfügung, nachdem in der Weboberfläche die Schaltfläche „**Automatik**“ («Автомат») eingeschaltet wurde. In den Modi **Bier** und **Käseherstellung** werden Pumpe und Ventil in den Kühl- und Umfüllstufen gemäß Programm eingeschaltet (siehe [Modus Käseherstellung](#)).

Das Ventil wird nur in den Modi und Stufen verwendet, in denen die Firmware Kühlung anfordert.

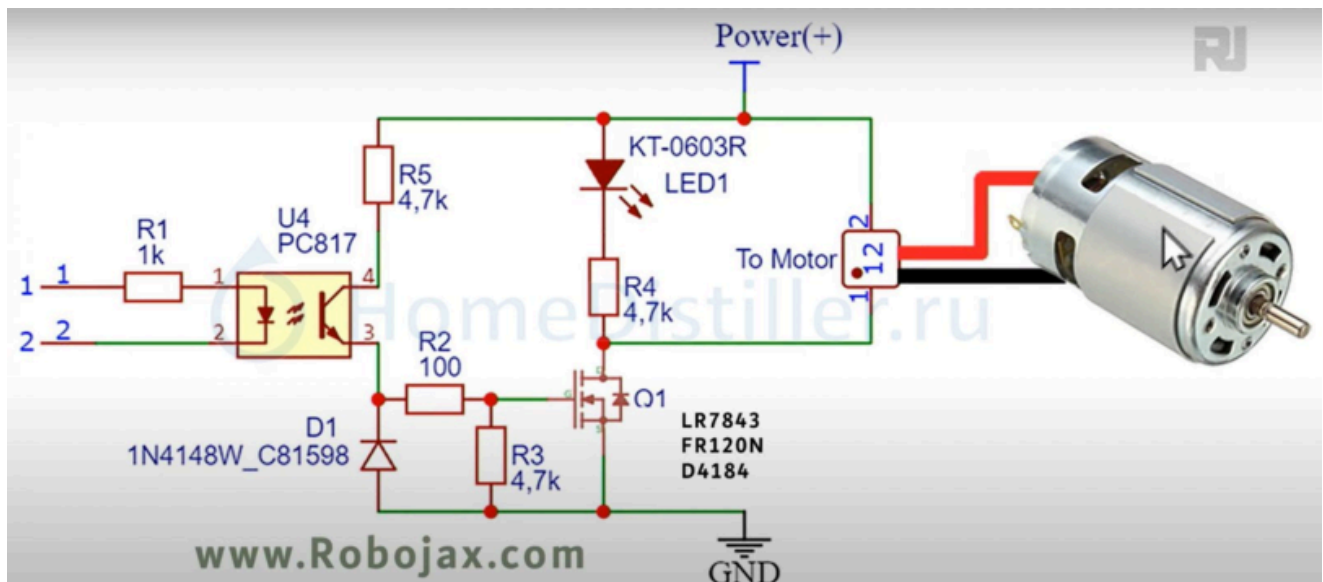
Umsetzung:

Zur Steuerung der Pumpe (oder des Durchflussventils) brauchen Sie ein Zwischenglied: ein Halbleiterrelais oder eine Leistungsplatine.

Achtung.

Manche Halbleiterrelais schalten bei direktem Anschluss an den ESP32 nicht - Spannung und Strom reichen nicht aus.

Empfohlen wird eine Platine mit Feldeffekttransistor (MOSFET) [LR7843](#)



Dieser MOSFET schaltet den Motor auf der Minusseite (Low-Side).
Auch wenn es heißt, dass der MOSFET kaum warm wird, sollten Sie die Platine im Gehäuse lieber im Bereich der Luftbewegung unterbringen.

Weiter: [Membranpumpe](#), [Kreispumpe](#) und [Kühlung ohne Pumpe](#)

Einsatz einer Membranpumpe

Es wird nicht empfohlen, die Pumpe direkt an eine Wasserleitung unter Druck anzuschließen. Sehr empfehlenswert ist ein Zwischenbehälter mit Schwimmerventil.

Ein Schwimmerventil ist einfach und zuverlässig. Es arbeitet jahrelang ohne Probleme. Zum Schutz vor Leckagen können Sie einen Notüberlauf hinzufügen.

Membranpumpen sind meist selbstansaugend, trotzdem ist es besser, das Wasser über einen Anschlussstutzen vom Boden des Behälters zu entnehmen, damit die Pumpe immer Wasser bekommt.

Empfohlene Pumpe: 12 V Versorgung, bis 60 W, Förderleistung 2–3,5 l/min.

Hinweis

Bei der Rektifikation an einer 2"-Kolonne liegt der Wasserverbrauch bei etwa 40–50 l/h. Bei der Destillation mit einem 3-kW-Heizelement ist der Verbrauch höher.

Diese Pumpen werden am häufigsten verwendet.



Obwohl ihre Leistung überdimensioniert ist, haben sich solche Modelle gut bewährt. Beachten Sie die Schraube und die Abdeckung links. Das ist eine Vorrichtung, die bei Überschreiten des Drucks automatisch die Stromversorgung abschaltet. Sie kann einstellbar sein oder nicht. Manche Modelle haben sie gar nicht.

Wenn Sie eine solche Pumpe einsetzen wollen, muss das 12-V-Netzteil von Samovar passend dimensioniert sein, zum Beispiel 12 V 100 W.

Die Pumpe ist laut, berücksichtigen Sie das bei der Wahl des Aufstellorts. *

Wenn Sie die Pumpe nur für die Rektifikation verwenden wollen, genügt eine bescheidenere Pumpe. Dank ihrer geringen Größe lässt sie sich direkt am Behälter oder an der Kolonne befestigen. Auch das

Geräusch einer solchen Pumpe ist geringer.



Betriebsspannung: 12 V DC,
Betriebsstrom: 0,5–0,7 A (Leistung muss über 6 W liegen)
Förderleistung: 1,5–2 l/min (oder in diesem Bereich),
Maximale Saughöhe: 78,7 Zoll (2 m)

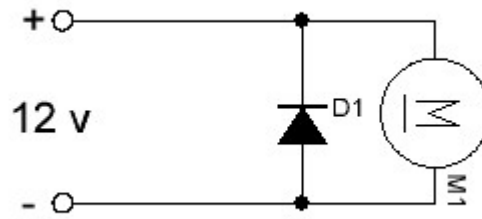
Grundsätzlich können Sie jede 12-V-Membranpumpe verwenden, deren Kenndaten Ihnen passen.

* Es gibt eine Möglichkeit, das Geräusch zu verringern und die Pumpe angenehmer zu machen: Ändern Sie die PWM-Frequenz. Standardmäßig ist sie in der Datei **pumppwm.h** festgelegt; *um diese Datei nicht zu bearbeiten, fügen Sie eine eigene Zeile in **user_config_override.h** ein:*

```
#define PUMP_PWM_FREQ 15
```

15 ist die Standard-PWM-Frequenz der Kühlpumpe in Hertz. Sie können bis zu 450 einstellen. Dieser Parameter ist im Konfigurator nicht verfügbar.

Außerdem müssen Sie eine Freilaufdiode parallel zur induktiven Last setzen. Das klingt beängstigend, ist aber einfach: Bauen Sie eine Diode so nah wie möglich am Motor ein.



Jede Diode ist geeignet, deren Nennstrom nicht kleiner als der Motorstrom ist. Schnelle Schottky-Dioden haben sich gut bewährt, zum Beispiel **1N5844** oder ähnliche.
Etwas ausführlicher – [Beitrag im Thread](#)

Einsatz einer Kreiselpumpe

Es gibt viele Pumpenarten: Membran-, Kolben-, Zahnrad-, Kreiselpumpen und andere. Die für unsere Zwecke verwendeten lassen sich in 2 Typen einteilen: solche, durch die im ausgeschalteten Zustand unter keinen Umständen Flüssigkeit gelangt, und solche, durch die Flüssigkeit im freien Gefälle fließen kann.

Zum zweiten Typ gehören verschiedene Kreiselpumpen mit Laufrad.

Wie erkennt man das? Blasen Sie in den Einlassstutzen; geht die Luft frei durch, ist es dieser Typ.

Zum Einsatz dieses Pumpentyps.

Welche Faktoren beim Betrieb wesentlich sind.

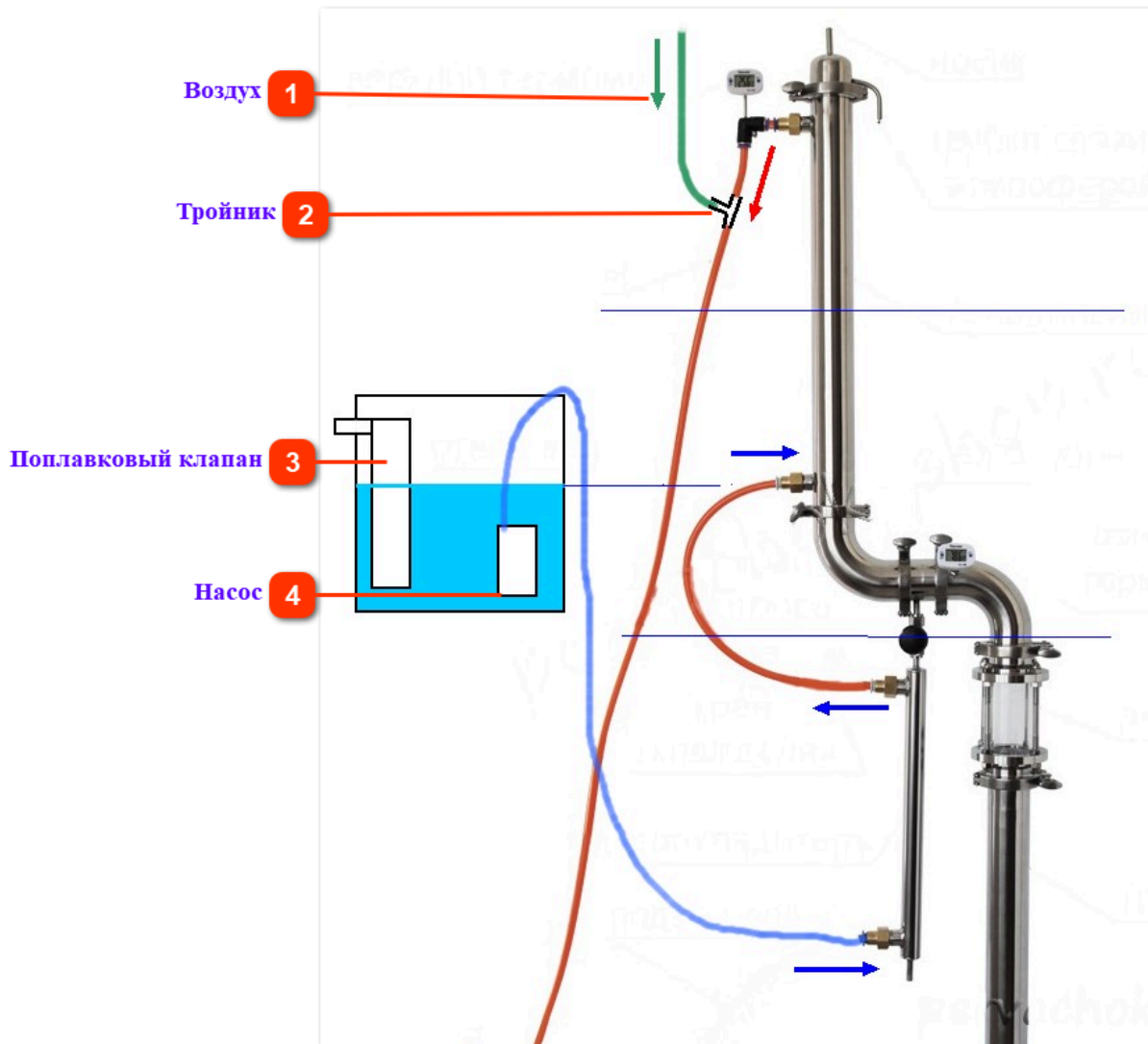
- Der erste Faktor ist die Förderleistung. Wenn der Hersteller angibt, dass die Pumpe 600 l/h fördert und Wasser auf 5 Meter Höhe hebt, heißt das nicht, dass sie in einer Stunde 600 Liter auf 5 Meter Höhe befördert. Entweder sind es 600 Liter auf Höhe der Wasserentnahme, oder 5 Meter Höhe und so viel, wie sie eben herausdrückt (falls sie überhaupt etwas herausdrückt).
- Der zweite Faktor ist die PWM. Sie bestimmt nicht nur die geförderte Menge, sondern auch die Höhe, auf die eine Wassersäule gehoben werden kann.
- Der dritte Faktor ist ein parasitärer, nämlich das unkontrollierte „Durchlaufen im Gefälle“, das im System auftreten kann.

Betrachten wir die Kühlung mit diesem Pumpentyp am Beispiel eines konkreten Aufbaus, den ein Kollege verwendet und der seit mehr als einem Jahr unter der Steuerung von Samovar läuft.

Die Temperaturdifferenz beträgt 1–2 Grad bei einer gehaltenen Auslasstemperatur von 60 °C.

Ein Behälter (3–5 Liter, Hauptsache, Anschlüsse und Pumpe passen hinein) mit Schwimmerventil und einer Aquarium-Tauchpumpe wird auf die Höhe des Dephlegmators gebracht.

Der vom Schwimmer gehaltene Wasserstand im Behälter sollte möglichst auf Höhe des unteren Stutzens des Dephlegmators liegen. Eine Abweichung von 10–15 cm nach oben oder unten ist unkritisch.



1

Luft

Durch dieses Röhrchen tritt Luft ein, um das unkontrollierte „Durchlaufen im Gefälle“ zu verhindern. Im Grunde ist es ein TCA (Entlüftungsrohr zur Atmosphäre) für Wasser

2

T-Stück

Sorgt für die „Strahlunterbrechung“

3

Schwimmerventil

Hält einen konstanten Wasserstand im Behälter.

4

Pumpe

Beim Einsatz eines **Rohrbündel-Dephlegmators**.

Die Pumpe speist den unteren Einlass des Nachkühlers, vom Auslass des Nachkühlers fließt das Wasser zum unteren Einlass des Dephlegmators. Das heiße Wasser tritt am oberen Auslass des Dephlegmators aus.

Bauen Sie am Auslass des Dephlegmators einen Temperatursensor ein und sorgen Sie für eine „Strahlunterbrechung“.

Setzen Sie dazu ein T-Stück in die Ablaufleitung und führen Sie von dort ein Röhrchen nach oben. Wichtig ist, dass das T-Stück über dem Flüssigkeitsspiegel des Behälters mit der Pumpe liegt.

Was dieser Aufbau bewirkt.

Die Pumpe muss nicht „wissen“, wie hoch das Wasser steigt; sie liefert tatsächlich einen Wasserdurchfluss, statt das Wasser mit dem Laufrad in sich selbst umzuwälzen, während sie versucht, Druck aufzubauen, um das Wasser nach oben zu heben.

Die PWM steuert tatsächlich den Wasserdurchfluss.

Die „Strahlunterbrechung“ verhindert unkontrolliertes Durchlaufen im Gefälle vollständig.

Hinweis:

Die „Strahlunterbrechung“ ist sehr wichtig. Im Test ohne Strahlunterbrechung und bei ausgeschalteter Pumpe (also Kühlung nur durch das Durchlaufen im Gefälle durch die Pumpe) stieg die Wassertemperatur am Auslass bei 1800 Watt zugeführter Leistung nicht über 40°.

Wenn ein **Dimroth-Kühler** als Dephlegmator verwendet wird.

Die Pumpe speist den unteren Einlass des Nachkühlers, vom Auslass des Nachkühlers fließt das Wasser zum oberen Einlass des Dephlegmators. Das heiße Wasser tritt am unteren Auslass des Dephlegmators aus.

Das Wasser fließt also dem Dampf entgegen.

Bauen Sie am unteren Auslass des Dephlegmators einen Temperatursensor ein und führen Sie die Ablaufleitung für das heiße Wasser möglichst zuerst nach oben bis zum oberen Einlass des Dephlegmators und dann nach unten zum Abfluss. Nachdem die Ablaufleitung nach unten geht, sorgen Sie ebenfalls für eine „Strahlunterbrechung“. Setzen Sie dazu ein T-Stück in die Ablaufleitung und führen Sie von dort das Röhrchen „TCA für Wasser“ senkrecht nach oben. Achten Sie darauf, dass das T-Stück nicht unter dem Flüssigkeitsspiegel des Behälters mit der Pumpe liegt.

Die Probleme, die bei Kreiselpumpen auftreten, betreffen Besitzer von Membranpumpen kaum.

Das einzige aufgetretene Problem waren sehr kaltes Wasser und ein launischer Motor. Bei eingestellten 15 % Leistung blieb die Pumpe stehen, und bei 20 % war die Auslasstemperatur zu niedrig.

Das kann übrigens passieren, wenn Sie eine Pumpe mit größerer Förderleistung verwenden.

Die Lösung ist ein Rücklauf direkt am Pumpenausgang. Auf den Auslassstutzen der Pumpe wird ein T-Stück gesetzt und ein fester Rücklauf über eine Düse (eine kalibrierte Bohrung) in denselben Behälter angelegt. So können Sie den minimalen Leistungswert anheben, während tatsächlich weniger Kühlflüssigkeit in die Kühlung gelangt.

Das Prinzip des kontrollierten Rücklaufs lässt sich auf jede Pumpe anwenden.

Ohne Pumpe

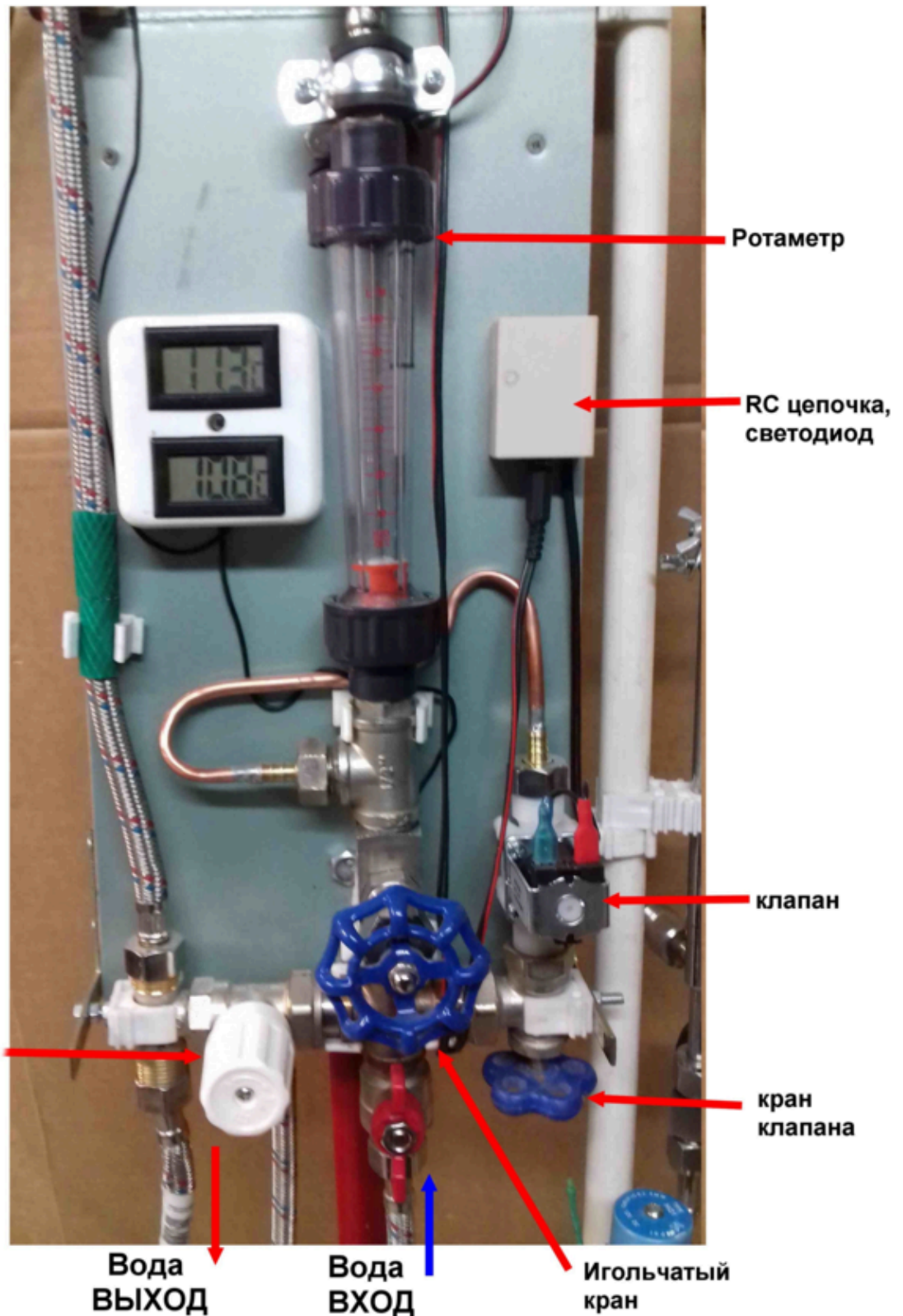
Samovar 7.00: Wasserpumpe und Wasserventil werden im Konfigurator auf der Registerkarte „Pumpen“ («Насосы») aktiviert; die folgenden Direktiven werden nur bei manueller Einrichtung benötigt.

Halbautomatischer Modus

Das Prinzip dieser Methode: Sie arbeitet eher wie ein Wächter (Watchdog), und das funktioniert recht gut.

Das Wasser kommt aus der Leitung (oder aus einem Druckbehälter oberhalb der Kolonne).

Es fließt zu einem Nadelventil und dann zu einem Rotameter, das zur Einstellung und Kontrolle eingebaut ist. Um das Nadelventil herum wird eine Bypassleitung mit einem weiteren Hahn und einem Ventil gebaut.



Die Einstellung erfolgt so:

Drehen Sie das Wasser auf. Schließen Sie das Nadelventil vollständig. Legen Sie zwangsweise Spannung an das Ventil an, sodass es öffnet, und stellen Sie mit dem Hahn des Ventils den Durchfluss ein, zum Beispiel 10 Liter pro Stunde. Nehmen Sie die Spannung weg – das Ventil schließt.

Das Ventil der Bypassleitung wird anstelle der Pumpe an Samovar angeschlossen – an den Pumpensteuer Ausgang (GPIO4 beim ESP32 DevKit, GPIO11 beim ESP32-S3) über einen Transistorschalter oder ein Relais, wie auf der Seite [Kühlung](#) beschrieben.

Deaktivieren Sie im Konfigurator auf der Registerkarte „Pumpen“ («Насосы») das Kontrollkästchen **„Wasserpumpe oder Rührwerk steuern“** («Управлять насосом воды или мешалкой») und tragen Sie im Feld **„Steuerung des Wasserventils“** («Управление клапаном воды») **LOW** ein (oder HIGH, wenn Ihr Schalter bei High-Pegel einschaltet). Der Konfigurator schreibt dies in die Datei Samovar_ini.h, in der sich dieselben Zeilen auch von Hand bearbeiten lassen:

```
//#define USE_WATER_PUMP                //использовать управление ШИМ для
насоса для воды
muss auskommentiert sein
```

und die Zeile:

```
#define USE_WATER_VALVE LOW            //использовать управление
клапаном для регулировки напора воды
muss einkommentiert sein. Solange USE_WATER_PUMP aktiv ist, hat die Direktive
USE_WATER_VALVE keine Wirkung – in der Firmware schließen sie sich gegenseitig aus.
```

Starten Sie dann die Kolonne und stellen Sie mit dem Nadelventil den Wasserdurchfluss so ein, dass die Temperatur am Ausgang des Dephlegmators zum Beispiel 50° beträgt. Bringen Sie die Kolonne in den Arbeitsbetrieb und korrigieren Sie den Durchfluss dabei leicht mit dem Nadelventil. Das war's – die Kolonne wird gekühlt.

Steigt die Wassertemperatur über den Sollwert des Wassersensors, öffnet das Ventil in der Bypassleitung, und zusätzliche 10 Liter pro Stunde fließen in die Kühlung; sobald die Temperatur sinkt, schließt das Ventil.

Das Ventil öffnet, wenn die Wassertemperatur mehr als 1 °C über den Sollwert steigt, und schließt, wenn sie mehr als 1 °C unter den Sollwert fällt. So arbeiten die Modi Rektifikation und Destillation. Im Bier-Modus wird dieser Ausgang nicht verwendet – es öffnet nur das Ventil an Relais Nr. 3; im Käseherstellungsmodus ist das Bypassventil offen, solange die Produkttemperatur mehr als 1 °C über dem Zielwert liegt.

In diesem Modus kann das System recht lange arbeiten. Man sollte es aber nicht lange unbeaufsichtigt lassen, und der Durchfluss muss regelmäßig mit dem Nadelventil nachgestellt werden.

Rücklauf-Füllstandssensor

Samovar 7.00: Aktivierung und Polarität des Sensors werden im Konfigurator auf der Registerkarte „Sensoren“ («Датчики») eingestellt und in user_config_override.h gespeichert.

Der berührungslose Flüssigkeitsstandsensor XKC-Y25-V beruht auf einem kapazitiven Messelement, das auf Änderungen der Dielektrizitätskonstante reagiert. Dadurch erkennt der Füllstandsmelder Flüssigkeit durch verschiedene nichtmetallische Wände hindurch: Kunststoff, Glas, Keramik usw. Der Sensor registriert sogar die Berührung Ihrer Hand. Die maximale Wandstärke und damit die Erfassungsdistanz des Sensors beträgt 12 mm. Durch Metallwände funktioniert der Sensor nicht.

Mehr dazu lesen Sie unter: [Berührungsloser Flüssigkeitsstandsensor XKC-Y25-V](#)

Tatsächlich kann Samovar mit jedem Sensor arbeiten (Laser, induktiv, Kontakt usw.), der mit 3,3 Volt betrieben werden kann (oder mit 5 Volt, sofern ein Pegelwandler verwendet wird) und beim Auslösen eine logische Eins oder eine logische Null ausgibt.

Welcher Sensor verwendet wird, lässt sich in der Datei Samovar_ini.h festlegen

Rücklauf-Füllstandssensor verwenden, der beim Auslösen eine logische Eins ausgibt.

```
#define USE_HEAD_LEVEL_SENSOR //использовать датчик уровня флегмы в голове (датчик P-N-P).
```

Rücklauf-Füllstandssensor verwenden, der beim Auslösen eine logische Null ausgibt

```
#define WHLS_HIGH_PULL //использовать датчик уровня жидкости N-P-N
```

(Alle folgenden Beispiele beziehen sich auf den Sensor XKC-Y25-V P-N-P)

Wahl des Ausgangssignalmodus.

Manche Sensormodelle haben 4 Anschlüsse und eine eingebaute Schaltung zur Invertierung des Ausgangssignals, die über den Zustand des Pins MODE gesteuert wird.

Normaler Modus

Der Pin MODE ist mit dem Pluspol der Versorgung verbunden oder gar nicht angeschlossen: Das Signal OUT wird nicht invertiert.

- Liegt der Flüssigkeitsstand im Erfassungsbereich des Sensors: Die LED leuchtet, der Sensorausgang ist High.

- Liegt der Flüssigkeitsstand außerhalb des Erfassungsbereichs des Sensors: Die LED ist aus, der Sensorausgang ist Low.

Invertierter Modus

Der Pin MODE ist mit Masse verbunden: Das Signal OUT wird invertiert.

- Liegt der Wasserstand im Erfassungsbereich des Sensors: Die LED leuchtet, der Sensorausgang ist Low.
- Liegt der Wasserstand außerhalb des Erfassungsbereichs des Sensors: Die LED ist aus, der Sensorausgang ist High.

Wo er platziert und wie er befestigt wird,

Der Sensor wird am Schauglas angebracht, da er durch Metall nicht funktioniert.

Stufe 1 – provisorisch-dauerhaft.

Sie benötigen:

- eine Plastikkarte
- zwei Foldback-Klammern
- den Sensor
- einen Filzstift
- einen Stufenbohrer
- ein Messer



Nehmen Sie die Karte, befestigen Sie sie mit einer Klammer an den Stäben des Schauglases (das Foto ist nicht besonders gut) und markieren Sie, wo der zweite Stab liegt.

Schneiden Sie sie zu und bohren Sie mit dem Stufenbohrer ein Loch, etwas kleiner als nötig. Schneiden Sie es mit dem Messer nach, sodass der Sensor straff sitzt.



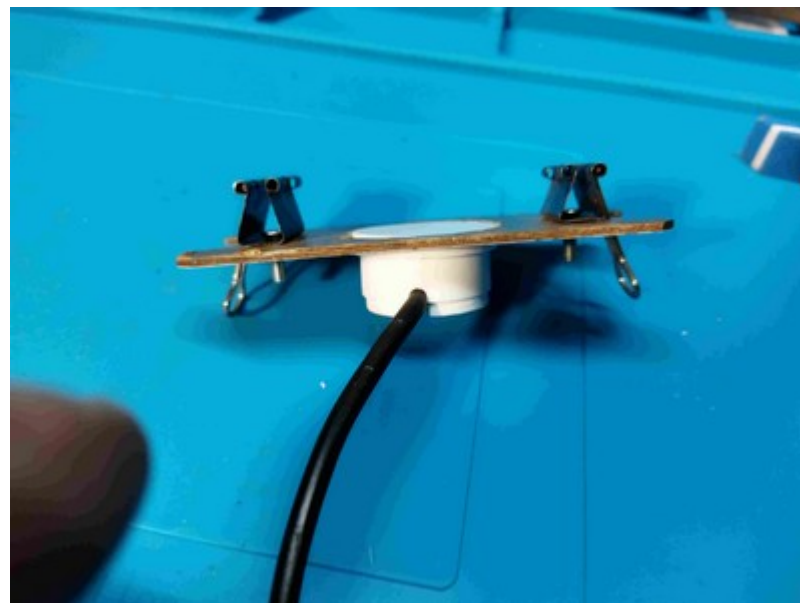
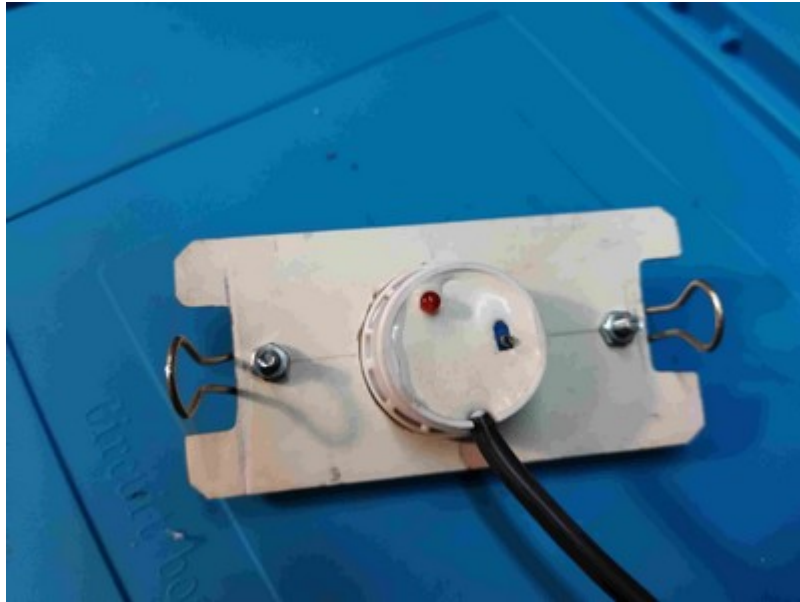
Zeichnen Sie + (Plus) und - (Minus) auf, das ist die Empfindlichkeitseinstellung, nehmen Sie die Abdeckung vom Sensor ab, stecken Sie ihn in das Loch und befestigen Sie ihn mit den beiden Klammern am Schauglas.



Fertig! Setzen Sie die Abdeckung nicht wieder auf – Sie müssen noch am Trimmer drehen, um den Auslösepunkt zu finden.

Dauerhaft.

Nehmen Sie ein Stück steifen Kunststoff und machen Sie darin auf dieselbe Weise ein Loch, jetzt werden die Klammern jedoch angebohrt und mit M3-Senkkopfschrauben befestigt.



Entfernen Sie eine Feder und befestigen Sie das Ganze auf dieselbe Weise am Schauglas.



Was das bringt. Erstens die Steifigkeit der Konstruktion. Zweitens – die Möglichkeit, den Sensor so tief wie möglich und sogar schräg zu montieren.

Der Sensor ist kapazitiv. Stellen Sie ihn mit einem Schraubendreher mit langem isoliertem Griff ein, am besten mit einem ganz oder überwiegend aus Kunststoff.

Schließen Sie den Sensor an den Anschluss LEVEL auf der Leiterplatte an. Die Prüfung ist einfach – anschließen, Hand annähern, und die LED leuchtet auf. Als Auslösung des Rücklauf-Füllstandssensors wertet Samovar einen Zeitraum von 3 Sekunden, in dem die LED des Sensors ununterbrochen leuchtet. Blinkt die LED in Abständen von weniger als 3 Sekunden, gilt das nicht als Auslösung.

Stellt Samovar fest, dass der Sensor ausgelöst hat, erhält der Regler den Befehl, die Spannung zu senken: um 1 V beim KVIC, um 2 V beim RMVK, um 3 % der aktuellen Leistung beim SEM_AVR. Die nächste Absenkung ist frühestens nach 40 Sekunden möglich – der Prozess reagiert träge. Während des Programms „Vorfluten“ («Предзахлёб») ist das Auslösen des Sensors ein normales Ereignis, und es wird keine Meldung über Fluten gesendet.

Nach dem Aufheizen, wenn die Kolonne in den Stabilisierungsbetrieb übergegangen ist und das Schauglas sich erwärmt hat, müssen Sie die Auslöseschwelle einstellen.

Verringern Sie zunächst die Empfindlichkeit des Sensors, bringen Sie die

Kolonne in den Arbeitsbetrieb und beginnen Sie mit der Abnahme des Vorlaufs. Drehen Sie mit einem Schraubendreher die Trimmerschraube, bis der Sensor auslöst, und drehen Sie sie dann ein wenig zurück.

Während der Vorlaufabnahme können Sie die Kolonne durch Erhöhen der Leistung fluten lassen, um sicherzustellen, dass der Sensor korrekt auslöst, und ihn feinzustimmen. Diese Episode beeinträchtigt die Qualität des Endprodukts nicht.

Je feiner die Einstellung, desto früher reagiert der Sensor auf Rücklauf im Schauglas, allerdings sind dann Fehlauslösungen möglich.

Wurde der Sensor abgenommen oder stand die Kolonne lange still, muss die Einstellung wiederholt werden.

Mögliche Probleme.

- Der Sensor ist nicht angeschlossen, aber Samovar meldet, dass er ausgelöst hat.

Bei der Einrichtung des Sketches können Sie im Konfigurator auf der Registerkarte „Sensoren“ («Датчику») das Kontrollkästchen „Rücklauf-Füllstandssensor verwenden“ («Использовать датчик уровня флегмы») deaktivieren oder in der Datei `Samovar_ini.h` die Zeile auskommentieren:

`//#define USE_HEAD_LEVEL_SENSOR` //использовать датчик уровня флегмы в голове (датчик P-N-P)

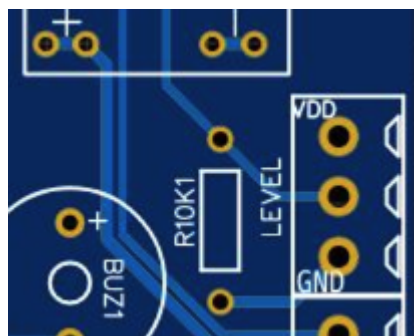
Oder ein anderer Weg:

Kommentieren Sie in der Datei `Samovar_ini.h` die Zeile aus

`//#define IGNORE_HEAD_LEVEL_SENSOR_SETTING` //Если датчик флегмы используется, то игнорировать команду на отключение датчика в WEB-интерфейсе. Dann erscheint in der Oberfläche auf der Registerkarte Einstellungen («Настройку») ein Kontrollkästchen, mit dem Sie die Verwendung des Rücklaufsensors ein- oder ausschalten können.

- Der Sensor ist angeschlossen, löst nicht aus, aber Samovar meldet, dass er ausgelöst hat.

Bei Verwendung des Sensors muss der Widerstand `R1-10k` (neben dem Anschluss `LEVEL`) bestückt sein.



Gibt der Sensor beim Auslösen eine logische Eins aus, muss der Füllstandseingang nach GND gezogen werden: GPIO27 beim ESP32 DevKit oder GPIO19 beim ESP32-S3. Auf der Standard-Leiterplatte für das DevKit ist dieser Pull-down bereits vorgesehen.

Gibt der Sensor beim Auslösen eine logische Null aus, muss der Füllstandseingang nach +3,3 V gezogen werden: GPIO27 beim ESP32 DevKit oder GPIO19 beim ESP32-S3.

Treiber. Schrittmotor.

Samovar 7.00: Drehrichtung und sanfte Beschleunigung des Motors werden im Konfigurator auf der Registerkarte „Schrittmotor“ («Шаровый двигатель») eingestellt.

Das Projekt verwendet Schrittmotortreiber (im Folgenden: Motor) A4988 oder DRV8825.

Welchen Treiber wählen? Lesen Sie diesen Artikel, um sich einen Überblick zu verschaffen; die wichtigsten Unterschiede zwischen diesen Treibern stehen am Ende.

Den Treiber setzt man am besten wie den ESP32 auf eine Stiftleiste.

Vor der Einstellung des Treibers muss der ESP32 auf der Platine sitzen und der Samovar-Sketch darauf hochgeladen sein.

Schließen Sie weder die Stromversorgung noch den Motor an.

Nehmen Sie den Treiber und stecken Sie ihn anhand der Pinbeschriftung auf der Treiberplatine und der Leiterplatte in die Buchsenleiste.

Prüfen Sie mit einem Multimeter im Durchgangsprüfmodus. Die GND-Pins müssen auf Masse liegen, an VMOT müssen +12 Volt vom Versorgungsanschluss VIN anliegen und an VDD +3,3 vom ersten Pin des ESP32.

Legen Sie 12 Volt an den Anschluss VIN an. Sie können noch einmal prüfen, ob die Spannungen anliegen.

Die Leiterplatte hat Schalter für den Treiber:

- H1 – alle drei Pins verbinden
- SW1 – falls Schalter vorhanden sind, alle auf OFF stellen. Falls keine Schalter vorhanden sind – die Jumper vorerst nicht setzen.

Jetzt muss der Treiber eingestellt werden.

Schließen Sie den Motor nicht an.

Ein Treiber ist empfindlich. Es gibt sogar Empfehlungen, ihn bei ausgeschalteter Versorgung einzustellen.

Also: den Trimmwiderstand ein wenig drehen, Versorgung anlegen, Spannung messen, Versorgung trennen – erneut drehen. Und so weiter, bis der gewünschte Wert eingestellt ist.

Es gibt einen anderen, schnelleren Weg.

Ein Schraubendreher mit isoliertem Griff. Verbinden Sie eine Multimeter-Messspitze an beliebiger Stelle der Leiterplatte mit Masse (GND) und

berühren Sie mit der anderen entweder vorsichtig den Trimmer und drehen den Schleifer mit dem Schraubendreher, oder Sie setzen eine "Krokodilklemme" auf die Messspitze, klemmen sie an die Spitze des Schraubendrehers und drehen den Schraubendreher über die Klemme.

Das Vorgehen bei der Einstellung finden Sie unter den Links.

[Einstellung des Schrittmotortreibers DRV8825](#)

[Einstellung des Schrittmotortreibers A4988](#)

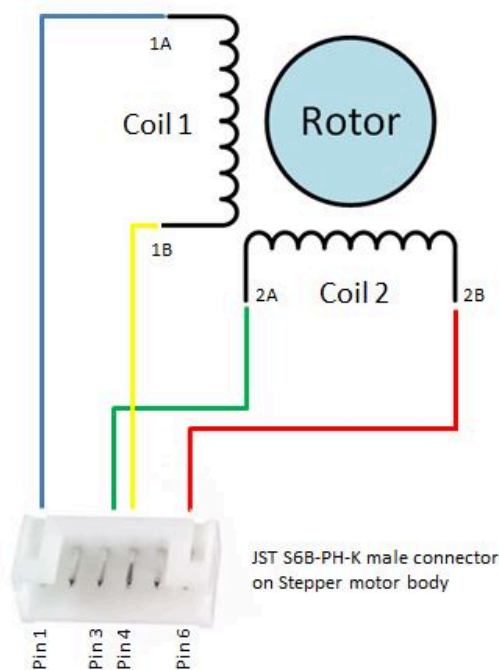
Anschluss des Schrittmotors.

Das Projekt verwendet einen bipolaren Schrittmotor NEMA17. Prüfen Sie zuerst, ob der Motor zum Projekt passt.

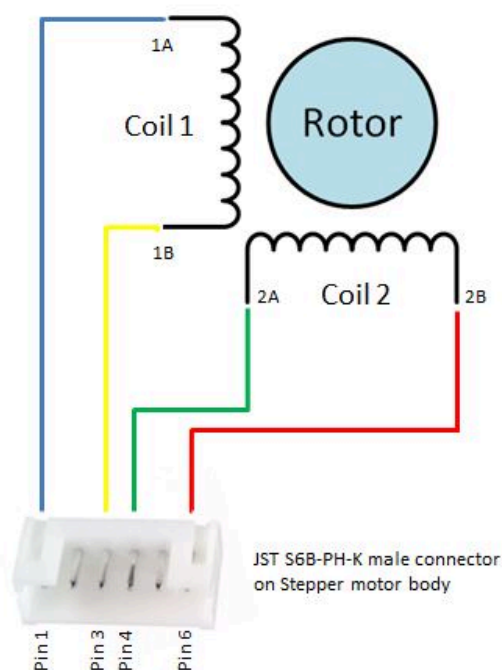
Bipolare Motoren haben zwei unabhängige Wicklungen. Motoren verschiedener Hersteller können unterschiedliche Anschlussbelegungen haben.

Nema 17 internal connections of coils to S6B-PH-K male socket

Crossover



Straight



Stecken Sie den mitgelieferten Stecker mit Kabeln an den Motor, nehmen Sie eine beliebige Ader und suchen Sie mit der Durchgangsprüfung des

Multimeters deren Gegenstück. Das ist eine Wicklung. Markieren Sie beide Adern. Prüfen Sie das zweite Paar. Piept es, ist alles gut.

Nehmen Sie nun eine beliebige Ader des einen Paares und eine beliebige des zweiten. Prüfen Sie sie. Sie dürfen nicht piepen. Piepen sie doch, ist es kein bipolarer Motor.

Schließen Sie nun den Motor an den Anschluss auf der Leiterplatte an. Gemäß dem Bild oben ist 1A-1B die eine Wicklung, 2A-2B die andere. "A" ist der Anfang einer Wicklung und "B" das Ende. Sie können natürlich streng nach Beschriftung anschließen, notwendig ist das aber nicht.

Wichtig ist nur, die eine Wicklung an die Pins 1A-1B des Treibers anzuschließen und die zweite an 2A-2B. Welche wohin kommt, ist egal, und ebenso, ob es der Anfang der Wicklung oder das Ende ist.

Angeschlossen. Legen Sie die Versorgung an. Der Motor dreht sich nicht.

Öffnen Sie die Weboberfläche und geben Sie den Befehl zur Abnahme.

Die Motorwelle sollte sich nun drehen.

Der Motor brummt und dreht sich nicht; der Motor dreht ruckartig

- Prüfen Sie, ob der Motor richtig und überhaupt an den Treiber angeschlossen ist. An den Treiber selbst, nicht nur an die Leiterplatte.

- Prüfen Sie, ob an den Pins RST und SLP des Treibers +3,3 Volt anliegen.

- Drücken Sie in der Weboberfläche auf der Seite Einstellungen, Registerkarte „Pumpe“ («Насос»), auf „Pumpenkalibrierung“ («Калибровка насоса»). Steht im Feld „Geschwindigkeit des Schrittmotors“ («Скорость шагового двигателя») 8000 (oder ähnlich), ändern Sie den Wert auf 800

([Forumsbeitrag](#): ...Ich hatte eine ähnliche Situation bei der Einrichtung von Samovar. Der Unterschied war der Treiber TMC2208. Das Problem ließ sich durch die Wahl der Schrittteilung lösen; stabil lief die Pumpe nur bei 1/8, in den anderen Modi brummte sie entweder oder lief ruckartig mit wilden Aussetzern (obwohl dieser Treiber mit dieser Pumpe an einem anderen Gerät bei allen Teilungen funktionierte; der einzige Unterschied waren die Steuersignale, 5 V statt 3,3 V). Vielleicht hängt es irgendwie mit der Frequenz der STEP-Impulse zusammen; versuchen Sie dann, im Fenster „Kalibrierung des Schrittmotors“ («Калибровка шагового двигателя») den Parameter „Geschwindigkeit des Schrittmotors“ («скорость шагового двигателя») passend zu wählen (etwa 4000 statt 8000). Und dann auf 100 ml kalibrieren.

Denken Sie daran: Die Betriebsmodi des Motors werden bei vollständig getrennter Versorgung der Samovar-Leiterplatte gewählt.)

Dreht sich die Welle in die falsche Richtung, vertauschen Sie die Adern einer der Wicklungen am Anschluss an die Platine.

Hinweis.

Die Datei Samovar_ini.h enthält eine Zeile, mit der sich die Drehrichtung per Software ändern lässt:

*//#define STEPPER_REVERSE //Drehrichtung des Schrittmotors ändern
Im Konfigurator ist das das Kontrollkästchen „Umgekehrte Drehrichtung des Schrittmotors“ («Обратное направление шагового двигателя») auf der Registerkarte „Schrittmotor“ («Шаговый двигатель»); auf derselben Registerkarte gibt es „Sanfte Beschleunigung des Schrittmotors“ («Плавный разгон шагового двигателя») (USE_STEPPER_ACCELERATION). Es ist aber bequemer, die Adern einmal zu vertauschen, als bei jedem Update auf den Parameter zu achten.*

Lassen Sie den Motor über das Abnahmeprogramm mit verschiedenen Geschwindigkeiten laufen. Stellen Sie die Abnahme auf 70–150 ml/h. Halten Sie das Motorgehäuse mit einer Hand fest und versuchen Sie, die Welle mit der anderen anzuhalten; lässt sich die Welle leicht anhalten, kehren Sie zur Treibereinstellung zurück und erhöhen Sie die Spannung ein wenig, buchstäblich um 0,02–0,04 Volt.

Prüfen Sie erneut, wie leicht sich die Welle anhalten lässt.

Kleben Sie den mitgelieferten Kühlkörper auf den Treiberchip.

Jetzt können Sie den Pumpenkopf montieren. Schmieren Sie zuvor die Rollen und den Schlauch mit neutralem Silikonfett. Das Schmieren muss regelmäßig wiederholt werden (alle 5–7 Brände); das verlängert die Lebensdauer von Rollen und Schlauch, und auch der Motor hat es leichter.

Lassen Sie ihn mit der Pumpe laufen und achten Sie darauf, ob es Aussetzer gibt, ob der Motor überlastet klingt und wie heiß Motor und Kühlkörper des Treiberchips werden...

Jetzt können Sie, falls nötig, zur Feineinstellung übergehen.

Ein wenig Theorie.

Anders als herkömmliche Gleich- und Wechselstrommotoren bewegen sich Schrittmotoren in diskreten Schritten, daher der Name „Schrittmotor“.

Jeder Schritt eines Schrittmotors entspricht einer genauen Winkeldrehung.

Üblich sind Schrittwinkel von 1,8° oder 0,9°, das entspricht einer vollen Umdrehung, die in 200 bzw. 400 Schritte geteilt ist.

Standardmotoren NEMA17 haben einen Drehwinkel von $1,8^\circ$ pro Schritt (200 Schritte pro volle Umdrehung)

Der Motor kann in mehreren Modi arbeiten – Vollschrift und Mikroschritt.

Vollschrift bedeutet, dass der Motor die Welle mit 1 Schritt um $1,8^\circ$ dreht

Im Vollschriftmodus hat der Motor das höchste Drehmoment aller Modi.

Mikroschritt ist die Teilung des Standardschritts in mehrere Mikroschritte. Der Standardschritt kann in 2, 4, 8, 16 und 32 Teile geteilt werden

Die Schrittteilung lässt sich mit den Schaltern SW1 einstellen

MS1, MS2, MS3 – Pins zur Wahl der Mikroschrittauflösung. Mit ihnen lässt sich einer von fünf Modi wählen:

Für den Treiber **A4988**

MS1	MS2	MS2	Schritte
aus	aus	aus	Voll
ein	aus	aus	1/2
aus	ein	aus	1/4
ein	ein	aus	1/8
ein	ein	ein	1/16

Für den Treiber **DRV8825**

MS1	MS2	MS2	Schritte
aus	aus	aus	Voll
ein	aus	aus	1/2
aus	ein	aus	1/4
ein	ein	aus	1/8
aus	aus	ein	1/16
ein	ein	ein	1/32

Erläuterung.

Angenommen, der Modus 1/4 ist gewählt. Dann dreht sich die Motorwelle mit 1 Mikroschritt um $1,8^\circ/4 = 0,45^\circ$, das entspricht $200 \cdot 4 = 800$ (oder anders gerechnet $360^\circ/0,45^\circ = 800$) Mikroschritten pro voller Umdrehung der Welle.

Im Mikroschrittmodus hat der Motor ein geringeres Drehmoment, dafür mehr Laufruhe und Genauigkeit (was für das Projekt nicht besonders wichtig ist)
Schrittmotoren erzeugen im Vollschriftmodus mehr Vibration und Geräusch.

Der Mikroschritt verringert Vibration und Geräusch deutlich, wodurch der Motor ruhiger und leiser läuft.

Kurz gesagt: Beobachten und hören Sie, wie Ihr Motor arbeitet. Sie können verschiedene Modi ausprobieren und vergleichen.

Achtung.

Die Betriebsmodi des Motors werden bei vollständig getrennter Versorgung der Samovar-Leiterplatte gewählt.

Nachdem Sie sich für einen Betriebsmodus des Motors entschieden und die DIP-Schalter in die richtige Stellung gebracht (oder die Jumper endgültig eingelötet) haben, müssen Sie die Pumpe kalibrieren. Die Kalibrierung muss jedes Mal wiederholt werden, wenn Sie den Betriebsmodus des Motors ändern. Die Genauigkeit der Abnahme hängt auch davon ab, wie regelmäßig die Pumpe geschmiert wird. Je seltener Sie schmieren, desto geringer ist die tatsächliche Abnahmegeschwindigkeit.

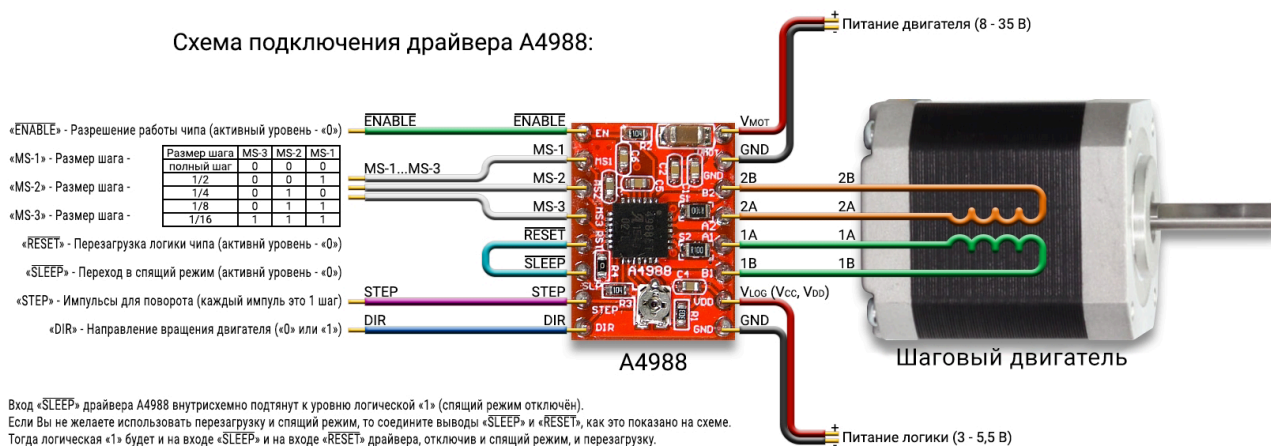
Unterschiede zwischen den Motortreibern DRV8825 und A4988

(vereinfacht, ohne konkrete Details)

- Unterschiedliche Lage des Potentiometers zur Einstellung der Strombegrenzung.
- Der Treiber DRV8825 unterstützt im Mikroschrittmodus 1/32 Schritt.
- Der Treiber A4988 unterstützt höchstens 1/16 Schritt.
- Der Treiber DRV8825 liefert ohne Kühlkörper bis zu 1,5 A pro Phase,
- Der Treiber A4988 liefert ohne Kühlkörper 1 A pro Phase.
- Der Treiber A4988 ist günstiger als der Treiber DRV8825.

Referenzinformationen.

Схема подключения драйвера A4988:



Fraktionsverteilung

Samovar 7.00: Die Korrekturen für die 11 Servopositionen werden im Konfigurator im Feld „Servokorrekturen (11 Zahlen)“ («Поправки сервопривода (11 чисел)») auf der Registerkarte „Ausstattung“ («Оборудование») eingegeben; er schreibt sie in das Array servoDelta in der Datei Samovar_ini.h.

Samovar kann den gesamten Durchgang gemäß dem von Ihnen erstellten Abnahmeprogramm in 11 Teile aufteilen.

Hinweis.

Wenn Sie keine Aufteilung brauchen oder die Behälter selbst wechseln, ist diese Baugruppe nicht nötig. Samovar funktioniert auch ohne sie.

Für den Aufbau brauchen Sie ein Servo, wie es in Arduino-Projekten verwendet wird.

Servo [SG90](#)

(Rc Mini Micro 9g 1.6 kg servo SG90 180° For RC 250 450 for Arduino DIY)



Achtung!

Samovar verwendet ein **Servo mit 180° Drehwinkel**; jedes andere arbeitet nicht korrekt (Servos gibt es mit 90°, 180°, 360° Drehwinkel oder mit Endlosdrehung).

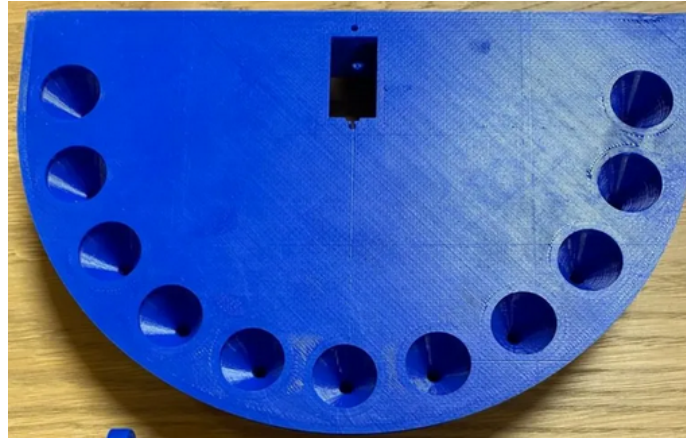
Wählen Sie das Servo vor dem Kauf sorgfältig aus. Im Zweifel lieber verzichten und bei einem anderen Verkäufer kaufen, der den Drehwinkel klar angibt.

Belegung der Leitungen: rot – +5 V, braun – GND, gelb oder orange – Steuersignal. Der Signaleingang wird beim ESP32 DevKit an GPIO25 oder beim ESP32-S3 an GPIO10 angeschlossen.

Zur Prüfung des Servo-Drehwinkels können Sie einen speziellen [Sketch](#) verwenden. Legen Sie den Sketch in dasselbe Verzeichnis wie die Samovar-Dateien (... Documents\Arduino\...) und installieren Sie ihn auf dem ESP32 statt Samovar. Wie man einen Sketch installiert, steht im [Benutzerhandbuch](#). Nach der Installation des Sketches auf dem ESP32 sollte sich das Servo in einer Schleife von der

Ausgangs- in die Endposition und zurück bewegen. Nach der Prüfung des Servos müssen Sie den Samovar-Sketch erneut auf den ESP32 hochladen.

Für das Servo SG90 wurde ein Modell eines Fraktionstellers für den 3D-Druck entwickelt. Die Modelldateien liegen im Verzeichnis 3D, das in jedem Samovar-Release-Archiv enthalten ist. Es sind die Dateien **Samovar_plate.stl** und **Samovar_horn.stl**. Drucken Sie mit einem aceton- und alkoholbeständigen Kunststoff, zum Beispiel PLA.



Das SG90 ist eines der günstigsten und beliebtesten Servos. Bei manchen Nutzern gab es Probleme mit seinem stabilen Betrieb.

Das äußerte sich darin, dass das Servo die gewünschte Position nur mühsam erreichte oder der Hebel irgendwann zu "zittern" begann. Die Erfahrung zeigt, dass das mit einer ungünstigen Führung des Produktschlauchs, seiner übermäßigen Größe, Steifigkeit oder umgekehrt Schlaffheit zusammenhängt. Das SG90 lässt sich ersetzen durch:

Servo [MG996R](#) , MG996, MG995.



Dieses Servo ist teurer als das vorige, aber auch kräftiger (und verbraucht daher mehr Energie), kann ein Kunststoff- oder Metallgetriebe haben (für unsere Zwecke egal) und hat mehr Zubehör.

Samovar verwendet ein **Servo mit 180° Drehwinkel**; jedes andere arbeitet nicht korrekt.

Belegung der Leitungen: rot – +5 V, braun, grau oder schwarz – GND, gelb oder orange – Steuersignal. Der Signaleingang wird beim ESP32 DevKit an GPIO25 oder beim ESP32-S3 an GPIO10 angeschlossen. Auch für diese Servos wurde ein Fraktionsteller-Modell für den 3D-Druck entwickelt. Es sind die Dateien **plate_996_servo.stl** und **horn_996_servo.stl**. Die Modelldateien liegen im Verzeichnis 3D im Samovar-Release-Archiv. Drucken Sie mit einem aceton- und alkoholbeständigen Kunststoff.

Beispiele für Aufbau und Umbau von Fraktionstellern:



Siehe im Forum: [Beitrag 1](#), [Beitrag 2](#)

Korrektur der Positionen

Jedes Servo hat eigene Abweichungen und wird individuell auf die Positionen des Tellers eingestellt. Wird der Fraktionsteller gedruckt oder nach Schablone mit gleichen Abständen zwischen den Trichtern gefertigt, müssen die Drehwinkel korrigiert werden.

Die Drehwinkel des Servos werden in der Datei **Samovar_ini.h** festgelegt (siehe [Hochladen und Aktualisieren des Sketches](#) und [Einrichten des Sketches](#)), das sind die Zeilen:

```
//Korrektur des Servo-Drehwinkels.  
int8_t servoDelta[11] = {0, -2, -3, -4, -3, -2, 0, 0, 0, 0, -2};
```

Die Werte in den geschweiften Klammern sind die Winkelkorrektur für jede **Position**. Es sind insgesamt 11. Die Auffangtrichter werden ab Position null gezählt. Die Korrekturwerte werden in Grad angegeben. Jede Position wird einzeln korrigiert. Das hängt nicht von der Korrektur benachbarter Positionen ab.

Erstellen Sie für die Korrektur ein Test-Abnahmeprogramm für den Modus „Rektifikation“ («Ректификация»)

```
H;450;0.10;0;0.00;45  
H;450;0.10;1;0.00;45  
H;450;0.10;2;0.00;45  
H;450;0.10;3;0.00;45  
H;450;0.10;4;0.00;45  
H;450;0.10;5;0.00;45  
H;450;0.10;6;0.00;45  
H;450;0.10;7;0.00;45  
H;450;0.10;8;0.00;45  
H;450;0.10;9;0.00;45  
H;450;0.10;10;0.00;45
```

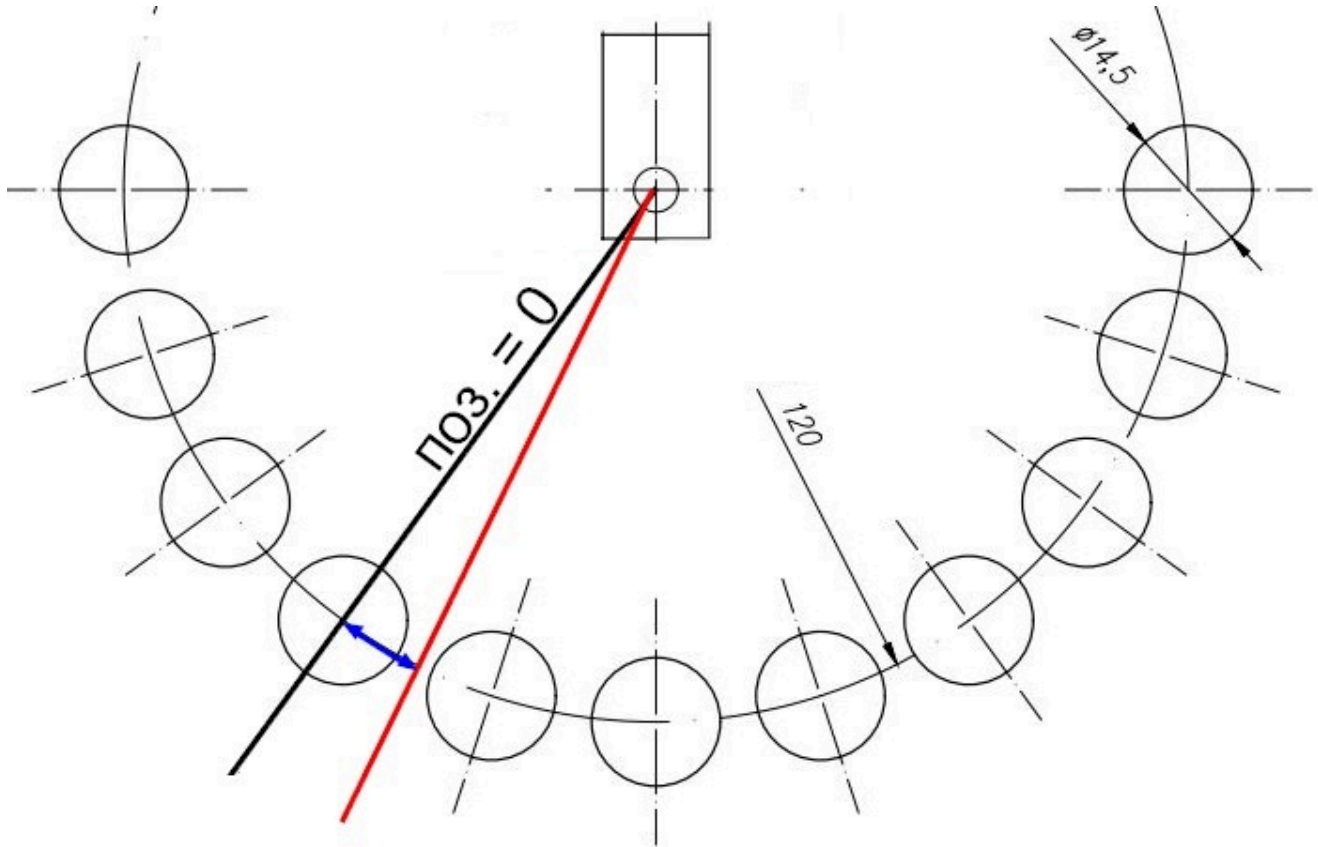
Kopieren Sie diesen Textblock, fügen Sie ihn in das [Textfeld des Abnahmeprogramms](#) auf der Registerkarte „Programm“ («Программа») der Hauptseite der Samovar-Weboberfläche ein und speichern Sie.

Sie können den Hebel noch nicht auf das Servo setzen oder ihn so montieren, dass er sich leicht wieder abnehmen lässt. Schließen Sie das Servo an und schalten Sie Samovar ein. Beim Start bringt Samovar das Servo in die Ausgangsposition. Setzen Sie den Hebel so auf, dass er auf Trichter null zeigt. Geben Sie in Samovar den Befehl „Heizung einschalten“ («Включить нагрев») (das Heizelement muss abgeklemt sein) und danach den Befehl „Abnahme starten“ («Начать отбор»). Der Arm bleibt bei Trichter null.

Der Befehl „Nächstes Programm“ («Следующая программа») bewegt den Arm in die nächste Position – Trichter Nr. 1. Fällt die Armposition mit der gedachten Linie durch die Mitte der Drehachse des Servos und die Mitte des Auffangtrichters zusammen, ist alles gut. Das ist null. Keine Korrektur nötig. Geben Sie erneut den Befehl „Nächstes Programm“, und der Arm fährt in die nächste Position.

Hat der Arm die gewünschte Position nicht erreicht oder ist er darüber hinausgefahren, ist eine Korrektur nötig. Nehmen wir als Beispiel Trichter Nr. 3 (denken Sie daran, dass die Trichter ab null gezählt werden).

Für diesen Trichter beträgt der Korrekturwert im Original-Sketch 4 (minus 4)



Angenommen, der Arm ist "übers Ziel hinausgeschossen" und auf der roten Linie stehen geblieben. Messen Sie die Bogenlänge (blauer Pfeil) zwischen null und der tatsächlichen Position. Sagen wir, es sind 10 mm.

Jetzt brauchen Sie die Bogenlänge für eine Drehung um 1 Grad. Suchen Sie einen Online- [Rechner](#) . Geben Sie den Radius ein (hier 120) und setzen Sie den Winkel = 1.

Wir erhalten: Bei einer Drehung um 1 ° beträgt die Bogenlänge 2 mm. Unser Arm steht 10 mm neben der gewünschten Position.

$10/2=5$ Grad.

Der Arm ist zu weit gefahren und muss zurück, also wird der Wert negativ. Im Sketch steht bereits - 4 (minus 4)

$-4+(-5)= -9$

Die Korrektur des Drehwinkels für Trichter Nr. 3, also Position Nr. 4, beträgt somit minus 9°

Das gilt für ein bestimmtes Servo und einen bestimmten Teller, mit Trichtern auf einem Kreis mit 120 mm Radius.

Gehen Sie auf die gleiche Weise alle Armpositionen durch und korrigieren Sie die Werte in den geschweiften Klammern der Zeile

`int8_t servoDelta[11] = {0, -2, -3, -9, -3, -2, 0, 0, 0, 0, -2};`

Laden Sie den Sketch nach der Korrektur erneut auf Samovar, setzen Sie das Test-Abnahmeprogramm und prüfen Sie, wie der Arm zwischen den Positionen fährt.

Fraktionsteller selbst bauen

Ein Fraktionsteller mit festen Trichterpositionen. Ein Beitrag [im Forum](#). Dort kann auch die Schablone für den Bau heruntergeladen werden.

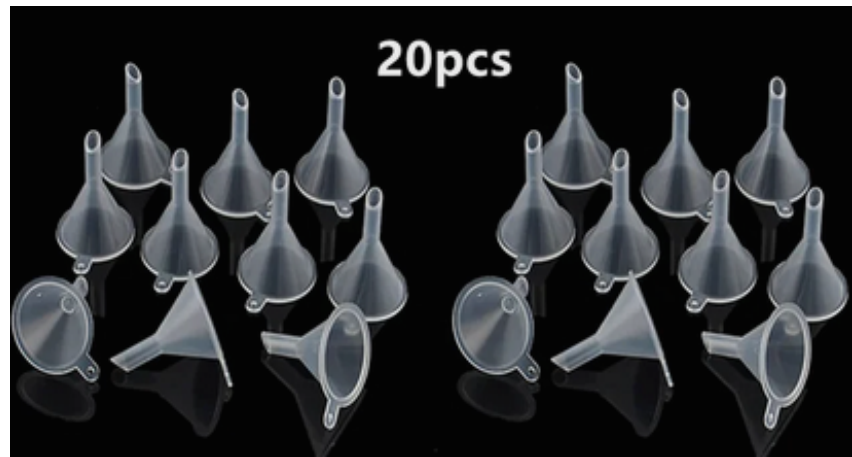
Ein Fraktionsteller, bei dem die Drehwinkel des Servos kaum korrigiert werden müssen.



Der Trick: Die Mittelpunkte für die Bohrungen werden nach der tatsächlichen Position des Hebels angezeichnet.

Bauanleitung [im Forum](#) .

Trichter



- Kaufen:

[Metall-Minitrichter für die Küche mit herausnehmbarem Filter, 5 Stk.](#)

[SUEF kleine Kunststoff-Öltrichter, 2/5/10/20 Stk., Trichter für flüssiges Öl.](#)

[Edelstahl-Minitrichter für die Küche mit herausnehmbarem Filter, 5 Stk.](#)

-

- Auf dem 3D-Drucker herstellen – die Trichterdatei liegt im Verzeichnis 3D.

- Verwenden Sie [Fläschchenkappen](#) oder medizinische [Spritzen](#). Spritzen unter 5 ml werden nicht empfohlen – sie können überlaufen; besser sind 10 oder 20 ml.

Laufen die Auffangtrichter über, ist die einfachste Abhilfe, die Auffangbehälter möglichst tief zu stellen und den Teller höher anzubringen. Die Flüssigkeitssäule im Schlauch vom Trichter zum Behälter beschleunigt den Abfluss zusätzlich.

Behebung möglicher Probleme. *Vorausgesetzt, das Servo funktioniert*

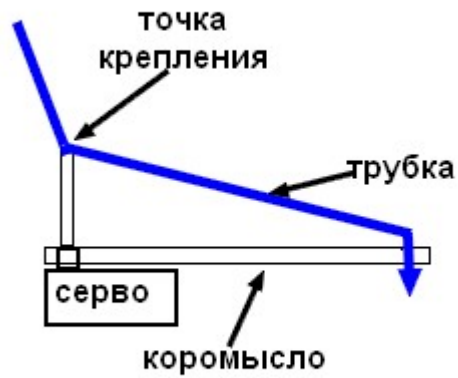
- Samovar startet neu, während das Servo arbeitet - der 12-auf-5-Volt-Wandler hat eventuell zu wenig Leistung, oder der Servomotor zieht zu viel Strom. Was Sie tun können:

- Als Übergangslösung die Kapazität von Kondensator C4 erhöhen. Bei Bedarf auch C3.

- Einen zweiten 12-auf-5-Volt-Wandler einbauen. Massen verbinden und +5 an das Servo legen (aus diesem Wandler können Sie alle 5-Volt-Verbraucher versorgen)

- Der Hebel "zittert" über dem Trichter, auch ohne angeschlossenen Schlauch – zu geringe horizontale Steifigkeit des Hebels.

- Der Hebel "pendelt" mit angeschlossenem Flüssigkeitsschlauch über dem Trichter.
 - Der Schlauch ist zu steif. Er "zieht" den Arm ständig zurück. Das Servo versucht immer wieder, ihn in die Position zurückzubringen.
 - Der Schlauch "baumelt" über dem Teller und zieht den Arm mal in die eine, mal in die andere Richtung.
- Als Möglichkeit: Fixieren Sie den Schlauch zusätzlich in der Mitte der Drehachse des Servogetriebes.



Not-Aus-Taster. Zusätzliche Notfallsensoren

Geprüft mit Samovar 7.00.

Schließen Sie den Taster oder Sensor zwischen dem Notfalleingang und GND an. Er löst bei Low-Pegel aus: GPIO35 beim ESP32 DevKit oder GPIO48 beim ESP32-S3.

Beim Auslösen schaltet Samovar die Heizung, die Wasserzufuhr, die Abnahme und das Rührwerk ab und sperrt einen Neustart bis zum Neustart der Platine. Mehrere Schließer-Taster und -Sensoren können parallel geschaltet werden. Verwenden Sie beim ESP32 DevKit die PlatformIO-Umgebung

Samovar_alarm_button oder fügen Sie **#define USE_ALARM_BTN** in **user_config_override.h** hinzu.

In der Umgebung **Samovar_s3** ist der Notfalleingang GPIO48 bereits aktiviert.

Sensor für ausgelaufenen Alkohol. [Anfang](#). [Umsetzung](#).

Beim ESP32 DevKit ist der Notfalleingang GPIO35, beim ESP32-S3 GPIO48. In beiden Fällen löst er aus, wenn der Eingang mit GND kurzgeschlossen wird.

Für GPIO35 am klassischen ESP32 ist ein externer 10-kOhm-Pull-up-Widerstand nach +3,3 V erforderlich: Die Pins GPIO34–39 haben keinen internen Pull-up. Für GPIO48 am ESP32-S3 aktiviert die Firmware den internen Pull-up.

Auf der Samovar-Platine 1.3.2 ist der Anschluss für den Not-Aus-Taster mit ALRM_BT beschriftet (ältere Unterlagen nennen ihn auch ALARM_BTN). Er ist mit GPIO35 verbunden und benötigt einen 10-kOhm-Pull-up-Widerstand nach +3,3 V.

Links zu den Beiträgen: [Link 1](#), sowie **ais77** [Link 2](#), [Link 3](#).

Empfohlene Konfiguration

Die empfohlene Konfiguration ergänzt den Minimalaufbau um die Stellglieder, die für den automatischen Betrieb nötig sind:

- ein Leistungsregler und ein Schütz zur Steuerung der Heizung;
- ein Wasserventil und bei Bedarf eine Kühlpumpe;
- ein Schrittmotor mit Treiber für die Abnahme;
- Temperatursensoren für Brennblase, Dampf, Kolonnenschuss und Wasser;
- ein Luftdrucksensor;
- ein Rücklauf-Füllstandssensor oder andere Notfallsensoren, die Sie verwenden.

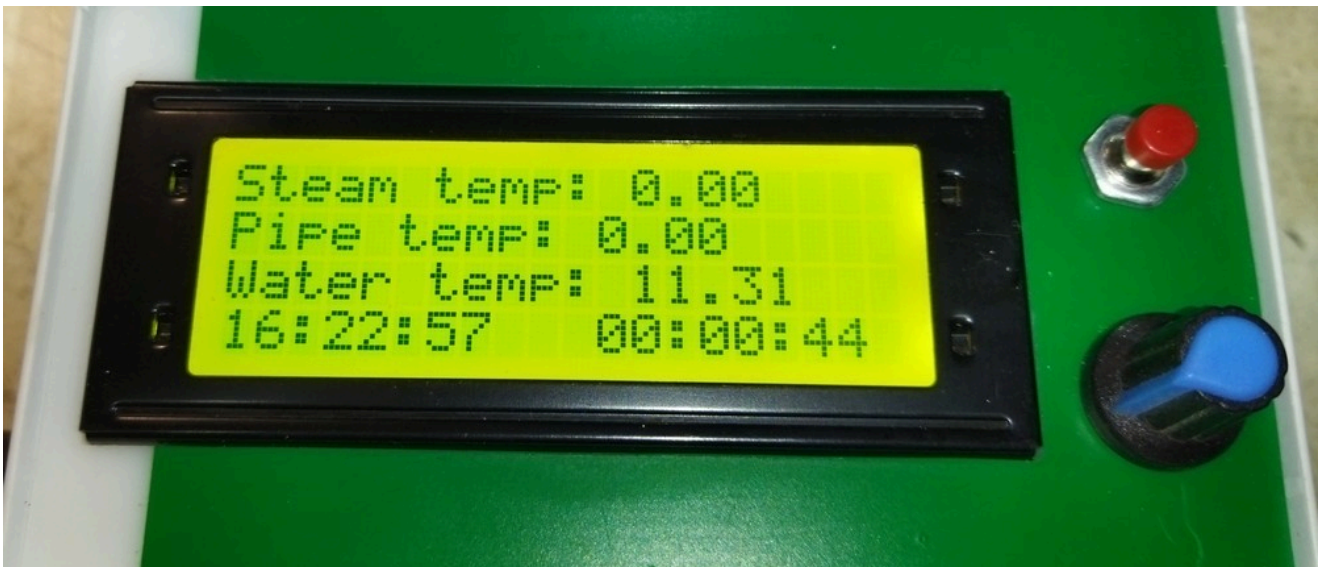
LCD-Display, Encoder und eine separate Taste sind für die Bedienung vor Ort praktisch, aber nicht erforderlich: Die wichtigsten Einstellungen und Bedienelemente sind über die Weboberfläche erreichbar.

LCD-Display, Encoder, Taste

Samovar 7.00: Die Bedientaste wird im Konfigurator auf der Registerkarte „Ausstattung“ («Оборудование») aktiviert.

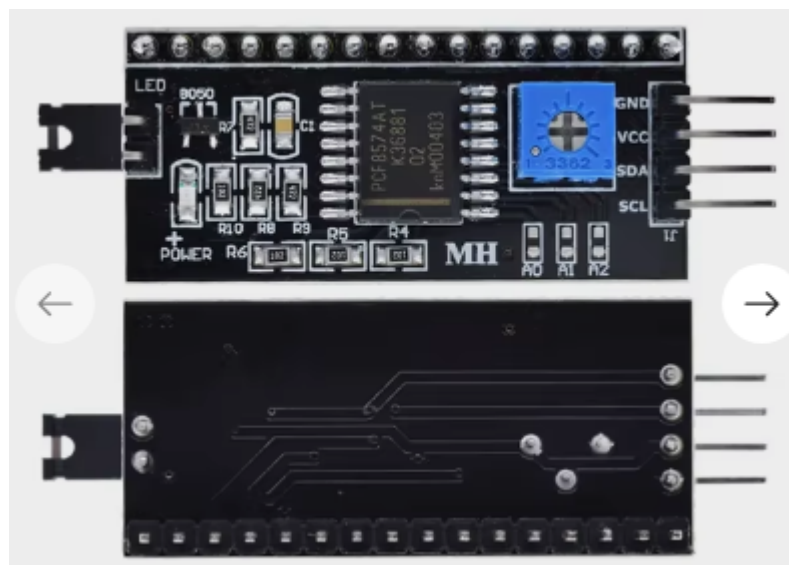
Achtung.

Samovar funktioniert ohne Display, Encoder und Universaltaste. Alle Informationen stehen in der Weboberfläche und bei aktivierter Integration auch in Blynk zur Verfügung. Ein Display ist direkt an der Anlage praktisch; ein Encoder ohne Display wird meist nicht gebraucht, eine separate Taste kann dagegen nützlich sein.



Im Projekt wird ein vierzeiliges Display verwendet: 20X4 LCD2004 mit einem I2C-Modul. Ohne das Modul können Sie das Display nicht an Samovar anschließen.

Das ist das Modul I2C



Und das ist ein 20x4-Display mit aufgelötetem I2C-Modul.



I2C LCD2004 Green

Sie können ein Display mit bereits aufgelötetem Modul kaufen oder beides separat als Set kaufen und selbst löten.

Die Hintergrundbeleuchtung kann blau, grün, gelb oder weiß sein (das beeinflusst den Preis). Wenn Sie die Beleuchtung nicht brauchen oder sie stört, können Sie sie nach der Einrichtung des Displays abschalten, indem Sie die Steckbrücke LED auf dem I2C-Modul entfernen. Der Text bleibt lesbar. Die meisten Displays unterstützen nur lateinische Schrift, manche auch Kyrillisch. Samovar verwendet lateinische Schrift.

Anschluss.

Das Display wird mit 4 Leitungen vom I2C-Modul an die Samovar-Platine angeschlossen, entsprechend der Pinbelegung von Modul und Platine. In der Regel leuchtet beim ersten Einschalten die Hintergrundbeleuchtung, aber es ist kein Text zu sehen. Drehen Sie am Trimpotentiometer auf dem I2C-Modul. Der Text erscheint. Stellen Sie den höchsten Kontrast ein. Damit ist die Einrichtung des Displays abgeschlossen.

Hinweis.

Werden einzelne Zeilen oder Zeichen auf dem Display falsch angezeigt, löten Sie die Verbindungen zwischen Modul und Display sorgfältig nach. Entfernen Sie die Flussmittelreste.

Encoder

Im Projekt wird ein Encoder EC11 auf einer „Breakout-Platine“ verwendet. Er sieht so aus.



Er wird mit 5 Leitungen gemäß Pinbelegung an die Samovar-Platine angeschlossen. Seine Hauptaufgabe ist das „Blättern“ durch die Zeilen auf dem

LCD2004-Display.

Blättert der Encoder die Displayzeilen in die falsche Richtung, tauschen Sie die Leitungen an den Kontakten S1 und S2.

Taste.

Wie Samovar auf einen Tastendruck reagiert, hängt vom gewählten Betriebsmodus ab (siehe [LCD-Display und Universal taste](#)).

Ein Taster mit Schließer genügt. Schließen Sie ihn zwischen dem Eingang und +3,3 V an: GPIO39 beim ESP32 DevKit oder GPIO47 beim ESP32-S3. Der Eingang braucht einen externen Pull-down-Widerstand von 10 kOhm nach GND.

Wenn Sie die Taste nicht verwenden, ist beim ESP32 DevKit kein Widerstand nötig, Sie müssen aber im Konfigurator auf der Registerkarte „Ausstattung“ («Оборудование») das Kontrollkästchen „Bedientaste verwenden“ («Использовать кнопку управления») deaktivieren oder die Taste in der Datei Samovar_ini.h auskommentieren:

```
///define USE_BTN //use the button
```

Beim ESP32-S3 ist der Tasteneingang GPIO47 unabhängig von dieser Einstellung immer aktiv, daher ist der 10-kOhm-Widerstand nach GND auch ohne Taste nötig; sonst kann es zu Fehlauslösungen kommen.

Achtung!

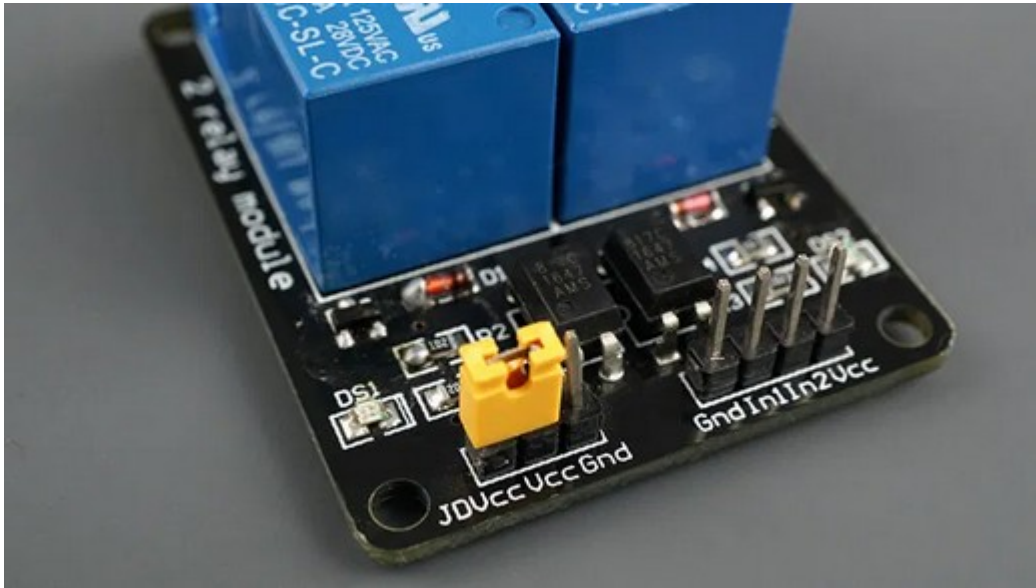
Wird beim Start die separate Taste etwa zwei Sekunden gehalten, startet der WLAN-Zugangspunkt. Wird die Encoder-Taste etwa zwei Sekunden gehalten, werden nur die gespeicherten WLAN-Einstellungen gelöscht. Die übrigen Hardware-Einstellungen werden nicht zurückgesetzt

Relaismodul richtig anschließen

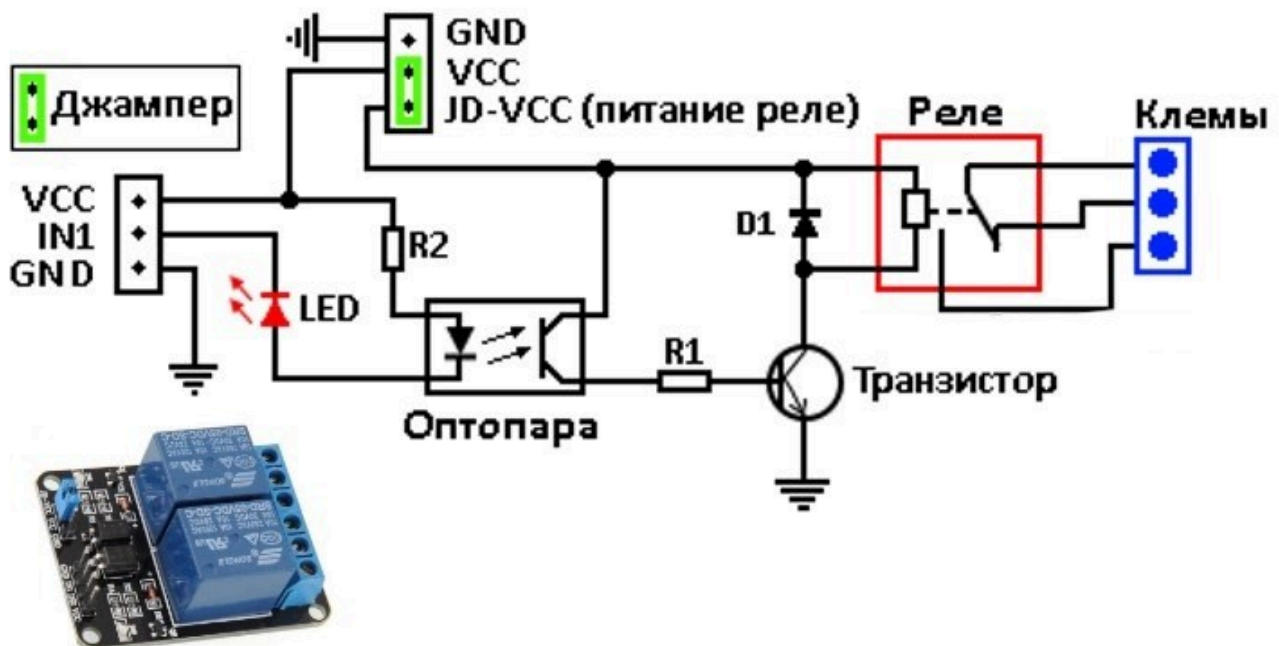
An den ESP32 können Sie Relaismodule für 5 Volt oder 12 Volt anschließen. Das Modul muss eine Steckbrücke JDVCC - VCC haben.

Achtung:

Fehlt die Steckbrücke JDVCC-VCC, hat das Modul keine galvanische Trennung der Stromversorgung. Solche Module sind nicht zu empfehlen.



Schaltplan am Beispiel eines Kanals:



Anschluss an den ESP32:

Der obere Steckverbinder im Schaltplan:

UNBEDINGT! Entfernen Sie die Steckbrücke!

- Legen Sie den Minuspol der Relaisversorgung (-5 oder -12 V, je nach verwendeten Relais) an den Pin GND.
- Der Pin VCC wird nicht verwendet (er bleibt also unbeschaltet).
- Legen Sie den Pluspol der Relaisversorgung (+5 oder +12 V) an den Pin JDVCC.

Der zweite Steckverbinder:

- Verbinden Sie VCC mit +3,3 Volt vom ESP32
- Verbinden Sie IN1, IN2 mit dem gewünschten Ausgang des ESP32. Für eine vollständige galvanische Trennung können Sie den Pin GND dieses Steckverbinders unverbunden mit GND des ESP32 lassen.

So wird das ganze Modul einschließlich der Relais aus einem separaten Netzteil (5 oder 12 Volt) versorgt, und von Samovar kommen nur 3,3 Volt: zum Widerstand R2, zur LED des Optokopplers und zur Anzeige-LED.

Liegt am Pin IN1 eine logische Null, zieht das Relais an; liegt eine logische Eins (+3,3 Volt) an, fällt das Relais ab

Hinweis.

Die meisten Relaismodule sind vom Typ Low Level Trigger (LLT, standardmäßig aktiviert). Das bedeutet: Um das Relais einzuschalten, muss am Steuereingang (IN1) eine logische "0" anliegen. Liegt am Eingang (IN1) eine logische "1", schaltet das Relais ab. Solche Relais werden eingesetzt, wenn der Mikrocontroller viele Geräte steuert, um seine Erwärmung und Stromaufnahme zu verringern. Sie werden auch in Systemen verwendet, in denen das Relais beim Einschalten der Versorgung oder "standardmäßig" eingeschaltet sein muss.

So steuern Sie das Modul mit einer logischen "1" statt einer logischen "0":

- Wählen Sie in der Weboberfläche von Samovar auf der Registerkarte [Einstellungen – Sonstiges](#) («Настройки – Прочие») den gewünschten Auslösepegel des Moduls.

Настройки уровней для реле	
Реле 1	HIGH ▾
Реле 2	LOW ▾
Реле 3	LOW ▾
Реле 4	LOW ▾

Die Felder **rele1...rele4** legen den Pegel fest, bei dem das Relais **einschaltet**. Für das hier beschriebene Modul, das mit logischer Null einschaltet, stellen Sie den Pegel **LOW** ein.

Achtung. Beim ESP32 DevKit ist Relais Nr. 2 (Rührwerk) an GPIO15 angeschlossen. Das ist ein Strapping-Pin, den die Hardware beim Reset der Platine auf High zieht. Ein Modul, das mit logischer Eins einschaltet, schaltet auf diesem Kanal bei jedem Start von Samovar kurz ein. Verwenden Sie für Relais Nr. 2 ein Modul, das mit Low-Pegel (LOW) einschaltet.

Erweiterte Konfiguration

Die erweiterte Konfiguration baut auf der empfohlenen auf und ergänzt Hardware für bestimmte Modi und Einsatzszenarien.

Sie können ein [LCD-Display mit Encoder](#), einen [Servo für die Fraktionsverteilung](#), ein oder zwei [I2CStepper](#)-Module für Rührwerk und Pumpe, eine [Not-Aus-Taste](#), zusätzliche Sensoren, ein [externes Messgerät](#) und Hardware für die [NBK](#) (kontinuierliche Maischekolonne) anschließen. Schließen Sie nur die Teile an, die in der gewählten Build-Umgebung aktiviert und in **Samovar_ini.h** oder **user_config_override.h** konfiguriert sind (am einfachsten über den Konfigurator, der aus **flash_windows.bat** oder **flash_macos_linux.sh** gestartet wird).

Hardware für die Käseherstellung

Samovar 7.00: Der Modus [Käseherstellung](#) nutzt die Standardausgänge der Platine; separate Anschlüsse sind nicht nötig:

Der pH-Sensor ist ein analoges Modul PH-4502C. Sein Ausgang wird an den Analogeingang LUA_PIN angeschlossen: GPIO34 beim ESP32 DevKit, GPIO4 beim ESP32-S3. Der Drucksensor MPX5010DP und die Lua-Funktion analogRead() nutzen denselben Eingang und können daher nicht zusammen mit dem pH-Sensor arbeiten. Die Messwerte werden mithilfe der Kalibrierung in pH umgerechnet, siehe [pH-Kalibrierung](#).

Das Rührwerk ist Relais Nr. 2 (dasselbe wie das Rührwerk im Bier-Modus) oder das Rührwerk des I2CStepper-Moduls mit Adresse 1; die Auswahl wird in der Programmzeile festgelegt.

Das Ablassventil ist Relais Nr. 4. In den anderen Modi steuert dieser Ausgang das Zusatzheizelement; schließen Sie daran also entweder ein Ventil oder ein Heizelement an.

Die Kühlung erfolgt wie in den anderen Modi über eine Wasserpumpe oder ein Ventil; ohne USE_WATER_PUMP oder USE_WATER_VALVE besteht die Kühlstufe die Prüfung im Käseherstellungsprogramm nicht.

Messgerät

Das Messgerät von dranek ist ein separates externes Gerät für zusätzliche Messungen. Es wird für den Grundbetrieb von Samovar nicht benötigt und ist nicht Teil der Samovar-Firmware: Samovar liest seine Werte nicht aus.

Der Quellcode des Messgeräts liegt im Samovar-Repository im Verzeichnis **pro_mini_ntc** (ein ESP8266-Sketch, ein Ordner „Схема и плата“ („Schaltplan und Platine“) mit Schaltplan und Gerber-Dateien sowie ein Ordner „Справочная информация“ („Referenzinformationen“) mit einer Beschreibung). Dieses Verzeichnis ist nicht Teil des Firmware-Builds von Samovar.

[Beschreibung, Schaltplan und Diskussion des Messgeräts im Forum.](#) Verwenden Sie beim Nachbau die Schaltung aus dem ursprünglichen Beitrag des Autors.

I2CStepper

I2CStepper ist ein eigenständiges Modul auf Basis eines Arduino Nano mit eigenem Schrittmotor, vier Relais, einem externen Sensor, einem 16×2-LCD und einem Drehgeber. Die aktuelle Firmware verwendet das Protokoll I2C v3. Das Modul kann als eigenständiges Gerät verwendet werden. Es lässt sich allein über das Menü auf dem 16x2-LCD und den Drehgeber bedienen oder in Verbindung mit Samovar.

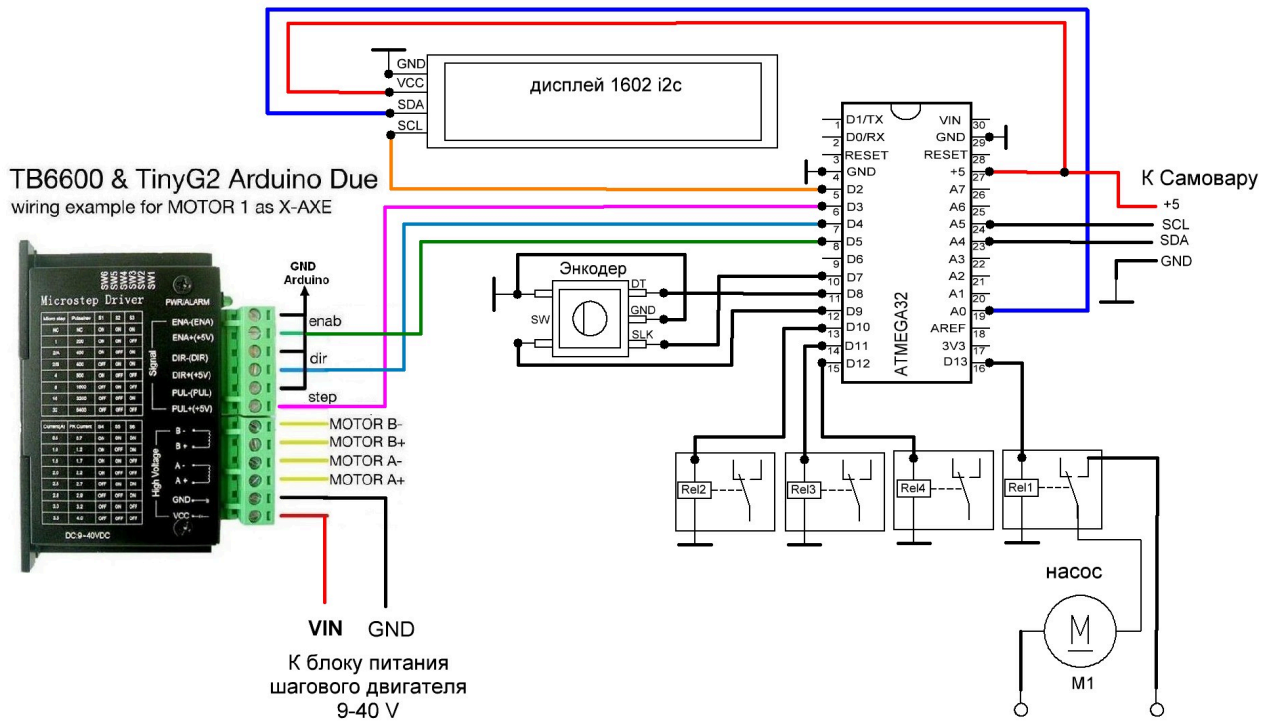
Die Erweiterung wird über den I2C-Bus mit vier Adern an Samovar angeschlossen (wie ein LCD-Display). Samovar erkennt, dass die Erweiterung angeschlossen ist, und kann sie steuern.

Samovar unterstützt bis zu zehn Platinen mit den Adressen 1–10: Eine ungerade Adresse ist ein Rührwerk, eine gerade Adresse eine Pumpen-/Abfüllplatine. Verwendet werden jeweils die Platinen mit der niedrigsten Adresse jedes Typs. Das Rührwerk arbeitet nach Drehzahl, Laufzeit und Pause; die Pumpe arbeitet in ml/h; der Abfüllmodus arbeitet mit vorgegebenem Volumen und vorgegebener Geschwindigkeit.

Meldet das Gerät Relaisunterstützung, lässt sich das erste Relais auf der I2CStepper-Seite schalten, alle vier aus Lua. Den Zweck der Relais legt der Benutzer fest: Die Samovar-Firmware definiert ihn nicht.

Die Module werden beim Start von Samovar auf dem Bus gesucht. Wird ein Modul gefunden, erscheint auf den Seiten der Modi Destillation («Дистилляция»), Maischekolonne («Бражная колонна») und Bier («Пиво») eine Registerkarte „**I2CStepper**“; sie öffnet die Seite zur manuellen Steuerung der Platine: Geschwindigkeit, Schrittzahl, Start und Stopp, Relais 1. Die eigenen Einstellungen der Platine (Adresse, Modus, Geschwindigkeit, Laufzeit und Pause des Rührwerks, Pumpenförderung, Abfüllvolumen und -geschwindigkeit, Schritte pro 1 ml) werden unter Einstellungen («Настройки») → Registerkarte „I2CStepper“ bearbeitet und im Speicher des Nano abgelegt. Die Kalibrierung der externen Pumpe wird mit der Schaltfläche „Kalibrierung der externen Pumpe“ («Калибровка внешнего насоса») auf der I2CStepper-Seite gestartet; das Ergebnis wird in das Feld „Schritte/ml“ («Шагов/мл») der Platine geschrieben.

Anschlussschema



Pinbelegung für den Anschluss der Peripherie:

// Pins für I2C Master - Anschluss des Displays und ggf. weiterer Geräte

// SDA_PIN 0 //A0

// SCL_PIN 2 //D2

// Pins für I2C Slave - Kommunikation mit Samovar

// SDA_PIN A4

// SCL_PIN A5

// Pins für den Schrittmotor

#define STEPPER_STEP 3

#define STEPPER_DIR 4

#define STEPPER_EN 5

//Pins für die Relais

#define MIXER_PUMP_PIN 13 // RELE_PIN1

#define RELE_PIN2 10 // RELE_PIN2

#define RELE_PIN3 11 // RELE_PIN3

#define RELE_PIN4 12 // RELE_PIN4

// Pins für den Drehgeber

#define ENC_CLK 7 //S2

#define ENC_DT 8 //S1

#define ENC_SW 9 //KEY

Tipp.

Wenn Sie die Erweiterung als separates Gerät in einem eigenen Gehäuse verwenden wollen, ist es sinnvoll, sie mit einem USB-A-B-Kabel (Computer – Drucker) an Samovar anzuschließen. Die Versorgung +/- 5 Volt des Moduls und die beiden Signale des I2C-Busses werden direkt vom Display von Samovar abgegriffen. Am Samovar-Gehäuse wird eine USB-B-Buchse montiert.

Am Gehäuse der Erweiterung wird eine USB-A-Buchse montiert.

So erhält der Arduino Nano beim Anschluss an Samovar gleichzeitig Versorgung und I2C-Bus. Beim Einsatz als eigenständiges Gerät wird statt Samovar ein separates Netzteil verwendet, zum Beispiel ein Handy-Ladegerät.

Bei einem eingebauten Netzteil wird an dessen Ausgang eine USB-B-Buchse gesetzt und dasselbe USB-A-B-Kabel zur Stromzufuhr verwendet; das schließt aus, dass versehentlich +5 Volt in Samovar eingespeist werden.

Peripherie.

Der Relaisblock wird mit 5 Volt betrieben, vorzugsweise mit Ansteuerung über Optokoppler. Auch 12-Volt-Relais sind möglich, mit einem [Anschluss](#) nach demselben Prinzip wie bei Samovar

Jede Pumpe ist verwendbar: mit Gleichstrommotor oder mit Wechselstrommotor, einphasig oder dreiphasig. Der Pumpenmotor wird nämlich über ein Relais und dann über einen beliebigen Leistungsschalter gestartet und gestoppt: ein Schütz oder ein Halbleiterrelais. Für kleine Mengen bietet sich die Magnetkupplungspumpe 15RM an. Edelstahl, leise, mit guter Förderleistung.

Für das Rührwerk oder die NBK (kontinuierliche Maischekolonne) ist jeder bipolare Schrittmotor geeignet, sofern Leistung und Drehmoment ausreichen. Empfohlen wird ein Nema 23 mit passendem Treiber.

Die aktuelle Firmware für den Arduino Nano und die Beschreibung des Protokolls v3 finden Sie im [I2CStepper-Repository](#). Prüfen Sie nach dem Umstieg auf das Protokoll v3 die Adresse der Platine, den Modus und die Kalibrierung „Schritte/ml“ («Шагов/мл»): Beim Versionswechsel wird die EEPROM-Struktur neu initialisiert.

Aktuelle Modi:

- Mixer: Rührwerk, Umdrehungen pro Minute, Laufzeit, Pause, Richtungswechsel nach der Pause und Sanftanlauf.
- Pump: kontinuierliche Förderung in ml/h mit optionaler Pause.
- Fill: dosiertes Abfüllen eines vorgegebenen Volumens in ml.

Ein externer Sensor kann die Pumpe stoppen oder pausieren. Hängt der I2C-Bus, stoppt das Modul selbst den Schrittmotor und schaltet die Relais ab; nach der Wiederherstellung ist eine Bestätigung mit dem Drehgeber erforderlich.

Achtung.

Die Lua-Funktion **set_stepper_target(spd, direction, target)** steuert nur die I2CStepper-Pumpe (die Arbeitsplatine mit gerader Adresse): Geschwindigkeit, Richtung und Schrittzahl. Antwortet die Pumpe nicht auf dem Bus, gibt die Funktion 0 zurück; den eingebauten Schrittmotor von Samovar steuert sie nicht.

--set_stepper_target(spd, direction, target) //spd - Geschwindigkeit in Schritten pro Sekunde,

direction - Richtung vorwärts oder rückwärts,

target - Anzahl der Schritte

Bei spd = 0 ist der Schrittmotor gestoppt.

Beispiel

set_stepper_target(950, 0, 1000)

Das Beispiel setzt Geschwindigkeit 950, Richtung 0 und eine Bewegung um 1000 Schritte.

Der pH-Säuresensor und sein Anschluss

Im Modus Käseherstellung ermöglicht der pH-Sensor, dass Samovar in der Phase „pH abwarten“ («Ожидание pH») auf den eingestellten Säuregrad wartet. Das Messset besteht aus einer Tauchelektrode und einer PH-4502C-Platine. Die Elektrode wird an die BNC-Buchse der Platine angeschlossen; deren Analogausgang PO gibt die Spannung an Samovar weiter. Schließen Sie die Elektrode selbst nicht direkt an den ESP32 oder den ADS1115 an: Sie braucht eine Platine zur Signalaufbereitung.

Die Elektrode muss für das zu messende Produkt und dessen Temperatur geeignet sein. Ein Sensor für Wasser kann in Milch durch Eiweiß und Fett verschmutzen; ob er für dauerhaftes Eintauchen geeignet ist, ergibt sich aus dem Datenblatt der Elektrode. Eine Kalibrierung beseitigt weder Verschmutzung noch Verschleiß.

Anschluss über den ADS1115

Der ADS1115 ist ein externer ADC, also ein Wandler von Spannung in eine Zahl. Für den Anschluss benötigen Sie ein ADS1115-Modul, zwei 10-k Ω -Widerstände mit 1 % Toleranz und einen 100-nF-Kondensator. Die Werte beziehen sich auf den Ausgang der PH-4502C-Platine, nicht auf das Elektrodenkabel.

- Versorgen Sie den PH-4502C mit 5 V und den ADS1115 mit 3,3 V. Verbinden Sie die GND-Massen beider Platinen und von Samovar. Prüfen Sie vor dem Anschluss die Pinbeschriftung Ihrer Platine: Die Pinreihenfolge ist von Modul zu Modul verschieden.
 - Verbinden Sie PO des PH-4502C über den ersten 10-k Ω -Widerstand mit AIN0 des ADS1115. Setzen Sie den zweiten 10-k Ω -Widerstand zwischen AIN0 und GND. Schalten Sie den 100-nF-Kondensator parallel zum zweiten Widerstand. Der Spannungsteiler halbiert die Spannung: 5 V an PO werden zu 2,5 V an AIN0. Der Ausgang DO wird für die pH-Messung nicht verwendet.
 - Verbinden Sie SDA und SCL des ADS1115 mit den gleichnamigen Leitungen des I2C von Samovar — der gemeinsamen Zweidraht-Verbindung zu den Modulen. Verwenden Sie die Pins Ihrer ESP32-Platine laut deren Schaltplan; die GPIO-Nummern hängen vom Aufbau ab. Die Pull-ups von SDA/SCL müssen an 3,3 V liegen.
 - Verbinden Sie ADDR mit GND: Die Adresse des ADS1115 ist dann 0x48. Ist diese Adresse bereits von einem anderen Gerät belegt, wählen Sie eine freie ADS1115-Adresse und geben Sie sie in der Build-Einstellung an. Zwei Geräte mit derselben Adresse dürfen nicht an einer Leitung arbeiten.
 - Aktivieren Sie die Build-Einstellung [USE ADS1115](#) mit dem Adresswert, zum Beispiel [#define USE ADS1115 0x48](#), und laden Sie die Firmware hoch. Der pH-Wert wird automatisch von AIN0 gelesen. LUA_PIN bleibt frei. Ist der ADS1115 nicht erreichbar, sind der Start der Käseherstellung und das Speichern der pH-Kalibrierung gesperrt.
- Prüfen Sie die Spannung zwischen AIN0 und GND mit einem Multimeter, bevor Sie an den ESP32 anschließen. Der Messbereich des ADS1115 in der Firmware beträgt $\pm 4,096$ V, erlaubt aber nicht, an den Eingang eine Spannung oberhalb der Versorgung des Chips anzulegen. Kalibrieren Sie nach einer Änderung der Schaltung oder dem Umstieg auf den ADS1115 erneut.

Anschluss ohne den ADS1115

Ist USE_ADS1115 deaktiviert, wird der Ausgang desselben Spannungsteilers statt an AIN0 an LUA_PIN angeschlossen. Der PH-4502C wird mit 5 V versorgt; seine GND wird mit der GND von Samovar verbunden. Legen Sie niemals 5 V direkt an einen Eingang des ESP32. LUA_PIN wird auch vom Drucksensor MPX5010DP genutzt: Schließen Sie nur ein Gerät daran an. Während der Käseherstellung ist dieser Eingang durch die pH-Messung belegt und kann nicht aus Lua gelesen werden.

Wenn die Messwerte beim Einschalten von Geräten springen

Vergleichen Sie zuerst die Messwerte in einem separaten Kunststoffbecher mit Pufferlösung und im Kessel, während Sie Pumpe, Rührwerk und Heizung nacheinander einschalten. Hängen die Sprünge mit

dem Einschalten von Geräten zusammen, sind elektrische Störungen eine wahrscheinliche Ursache. Ein langsames Wandern der Messwerte im Puffer bei ausgeschalteten Geräten erfordert eine Prüfung der Elektrode und des PH-4502C; eine galvanische Trennung behebt deren Defekt nicht.

Um Störungen über die gemeinsame Versorgung und Masse auszuschließen, können Sie den Messteil abtrennen: **Elektrode** → **PH-4502C** → **Teiler 1:2** → **ADS1115** → **ADuM1250** → **ESP32**. Der ADuM1250 ist ein Modul zur galvanischen Trennung von I2C: Es überträgt Daten, ohne die beiden Seiten elektrisch zu verbinden.

- Versorgen Sie den PH-4502C, den ADS1115 und die Messseite des ADuM1250 aus einer separaten USB-Powerbank mit 5-V-Ausgang, die geringe Last unterstützt. In dieser Schaltung erhält auch der ADS1115 5 V; der Spannungsteiler an seinem Eingang bleibt. Während der Messung darf die Powerbank weder geladen werden noch andere Teile von Samovar versorgen.
- Versorgen Sie die dem ESP32 zugewandte Seite des ADuM1250 aus den 3,3 V von Samovar. Schließen Sie SDA/SCL des ADS1115 an die 5-V-Seite und SDA/SCL von Samovar an die 3,3-V-Seite an. Die Pull-up-Widerstände jeder Seite müssen an der Versorgung genau dieser Seite liegen.
- **Verbinden Sie die Massen der beiden Seiten nicht.** Verbinden Sie sie mit einem Draht, einem USB-Kabel oder über andere Geräte, ist die Trennung aufgehoben. Die übrigen I2C-Geräte von Samovar bleiben auf der ESP32-Seite.
- Platzieren Sie PH-4502C und ADS1115 nah beieinander und fern vom Heizungsregler. Halten Sie die PO-Leitung kurz und führen Sie das Elektrodenkabel abseits von Leistungsleitungen. Die ADS1115-Adresse und die Einstellung USE_ADS1115 bleiben gleich.

Dies ist eine Anschlussvariante gegen elektrische Störungen, keine Garantie für die Genauigkeit der Elektrode. Wiederholen Sie nach der Kalibrierung die Prüfung mit dem Einschalten der Geräte; die Schaltung darf beim Schalten keine reproduzierbare pH-Verschiebung zeigen.

Softwarefilter und Einschränkungen

In Firmware mit Medianfilter wird der pH-Wert aus fünf aufeinanderfolgenden Messungen im Abstand von einer Sekunde berechnet. Der Median ist nach dem Sortieren die mittlere Zahl der Reihe: Die Reihe 5,42; 5,41; 3,10; 5,43; 5,42 ergibt 5,42. Der Filter entfernt kurze Ausreißer, behebt aber weder einen konstanten Fehler noch ein langsames Wandern der Messwerte. Bei gleichmäßiger Änderung beträgt die Verzögerung etwa zwei Sekunden.

Der pH-Wert steht zur Verfügung, sobald alle fünf Werte gefüllt sind. Nach einem Lesefehler, einem ungültigen Messwert, einer Pause zwischen Messungen von mehr als fünf Sekunden, einer Änderung der Kalibrierkoeffizienten oder einem Zurücksetzen des Prozesses wird das Fenster neu gefüllt. Ein altes Ergebnis wird nicht als neue gültige Messung ausgegeben. In der Phase „pH abwarten“ («Ожидание pH») wird die Bestätigung des Erreichens der Schwelle 30 Sekunden lang gehalten.

Der Filter gilt für beide Anschlüsse — über den ADS1115 und über LUA_PIN. Das ADC-Feld auf der Kalibrierseite zeigt einen frischen Rohwert ohne diesen Filter; warten Sie, bis er sich stabilisiert, bevor Sie einen Punkt festlegen. Samovar berücksichtigt bei der pH-Umrechnung die Temperatur nicht: Vergleichen Sie Ergebnisse und Kontrollproben bei gleicher Temperatur. Eine Aktualisierung der Dokumentation auf der Website aktualisiert nicht die Firmware auf Ihrem Gerät.

Kalibrierung und Pflege

Kalibrieren Sie das gesamte zusammengebaute Set mit zwei frischen Pufferlösungen von etwa pH 7 und pH 4. Geben Sie die Werte vom Etikett der Lösung für die Messtemperatur ein. Das Vorgehen mit den Schaltflächen ist im Abschnitt [pH-Kalibrierung](#) beschrieben.

Reinigen Sie die Elektrode nach Milch mit einem geeigneten Reiniger gegen Eiweiß- und Fettablagerungen gemäß den Herstellerangaben. Lagern Sie die Messspitze in der empfohlenen Aufbewahrungslösung für Elektroden, nicht trocken und nicht in destilliertem Wasser. Überschreiten Sie die zulässige Temperatur der Elektrode nicht. Prüfen Sie nach der Reinigung die Messwerte in einer frischen Portion Pufferlösung.

Referenzmaterial: [ADS1115](#), [ADuM1250](#), [Pflege der Elektrode](#), [Ursachen elektrischer Störungen bei der pH-Messung](#).

Zusätzliche Funktionen

Zusätzliche Funktionen lassen sich mit Lua und den Port-Erweiterungen PCF8575/PCF8591 über I2C umsetzen. Lua wird über die separate Build-Umgebung **Samovar_lua** in PlatformIO aktiviert, über das Kontrollkästchen „Lua verwenden“ («Использовать Lua») auf der Registerkarte „Hardware“ («Оборудование») des Konfigurators oder über die Direktive **#define USE_LUA** in **user_config_override.h**; die Erweiterungen werden an derselben Stelle mit den Direktiven **USE_EXPANDER** und **USE_ANALOG_EXPANDER** aktiviert (im Konfigurator die Felder „Adresse der PCF8575-Erweiterung“ («Адрес расширителя PCF8575») und „Adresse der PCF8591-Erweiterung“ («Адрес расширителя PCF8591»)).

Die Programmierung zusätzlicher Funktionen erfolgt auf Basis der [Skriptsprache Lua](#). Der Skripteditor ist auf der Seite Einstellungen – Editor verfügbar („Einstellungen“ («Настройки») – „Editor“ («Редактор»)).

An die Port-Erweiterung können Sie beliebige Aktoren oder Taster (Sensoren, die wie Taster arbeiten) anschließen und selbst festlegen, was zu einem bestimmten Zeitpunkt während des Betriebs von Samovar geschehen soll. Die Datei **init.lua** wird bei der Initialisierung von Lua ausgeführt. Während Samovar arbeitet, laufen das gemeinsame **script.lua** und das in den Einstellungen gewählte Modusskript etwa einmal pro Sekunde. Sie können sie im Dateieditor der Weboberfläche ändern.

Sie können Variablen im Speicher von Samovar ablegen, bis zu 10 Timer starten, den Betriebszustand von Samovar abfragen usw. Sie können also zusätzliche Geräte anschließen und sie während des Betriebs von Samovar automatisch steuern

Sie können prüfen, ob ein Sicherheitssensor ausgelöst hat, und in diesem Fall Samovar direkt aus dem Skript abschalten oder die Wasserzufuhr erhöhen bzw. stoppen. Die Szenarien können vielfältig sein. Sie hängen von der konkreten Ausrüstung ab.

Die Grundfunktionen von Samovar ändern sich dadurch nicht.

Skripte können einen analogen Eingang der Platine nutzen — **LUA_PIN** (GPIO34 beim ESP32 DevKit, GPIO4 beim ESP32-S3), der mit der Funktion `analogRead()` gelesen wird. **Samovar 7.00:** Im Modus [Käseherstellung](#) belegt der pH-Sensor diesen Eingang, und ein Zugriff aus einem Skript endet mit einem Fehler. Die Ausgänge für Relais, Pumpen und Taster lassen sich nicht aus einem Skript schalten — nur über die Port-Erweiterung oder die Standardfunktionen von Samovar.

Port-Monitor über Bluetooth

Die Samovar-Firmware nutzt das eingebaute Bluetooth des ESP32-Moduls nicht. Mit externen HC-05- oder HC-06-Modulen können Sie einen drahtlosen Monitor für die serielle Schnittstelle bauen: Sie übertragen einfach per Funk alles, was Samovar auf den Hauptdiagnoseport ausgibt (denselben, den Sie im Port-Monitor der Arduino IDE oder von PlatformIO sehen). Der Port arbeitet mit **115200 Baud**.

Verbinden Sie den RX-Eingang des Bluetooth-Moduls mit TX der seriellen Hauptschnittstelle (Pin TX0 der Platine), verbinden Sie die GND-Leitungen und stellen Sie das Modul auf 115200 Baud ein. Wählen Sie die Versorgungsspannung nach der Kennzeichnung Ihrer konkreten Modulplatine; die Signalleitungen des ESP32 arbeiten mit 3,3 V. Der TX-Ausgang des Moduls muss nicht an RX von Samovar angeschlossen werden, wenn Sie nur das Protokoll ansehen wollen. Lassen Sie UART2 mit den Pins RXD2/TXD2 (GPIO16/GPIO17 beim ESP32 DevKit, GPIO18/GPIO17 beim ESP32-S3) für den Leistungsregler frei – das ist ein anderer Port.

Die Ausgabe der Funktion **print()** von Lua-Skripten geht auf denselben Port. Der Ordner **Lua_UART** in den Projektquellen enthält ein Beispiel für einen separaten Arduino-Empfänger: Er empfängt eine Zeichenkette über UART, teilt sie an den Kommas in 8 Teile (je bis zu 20 Zeichen) und zeigt sie auf einem Display an. Schließen Sie das Modul nicht direkt an Netzstromkreise oder an die Ausgänge des Reglers an.