

САМОВАР

Техническая документация

Прошивка 7.00

Актуальная версия: <https://www.samovar-tool.ru/doc/device/>

Собрано 17.09.2026

Содержание

Служебный раздел. Прочитать в первую очередь:)

С чего начать

- Сборка и прошивка через PlatformIO

- Монтаж ESP32

Минимальная конфигурация

- Зуммер

- Датчики атмосферного давления

- Датчики температуры

- Используемые регуляторы напряжения

 - Обзор регуляторов

 - Как подключить регулятор к Самовару

 - ТЭН

 - Использование разгонного ТЭНа

- Контактор. УЗО. Аварийное отключение

- Охлаждение

 - Использование мембранного насоса

 - Использование центробежного насоса

 - Без насоса

- Датчик уровня флегмы

- Драйвер. Шаговый двигатель.

- Распределение по фракциям

- Аварийная кнопка. Дополнительные аварийные датчики

Рекомендованная конфигурация

- LCD дисплей, энкодер, кнопка

- Как правильно подключить релейный модуль

Расширенная конфигурация

- Измеритель

- I2CStepper

- Датчик кислотности pH и его подключение

Дополнительные возможности

- Монитор порта по Bluetooth

Служебный раздел. Прочесть в первую очередь:)

Служебная информация для тех, кто помогает дополнять и исправлять руководство по сборке.

Данная информация предназначена для тех, кто участвует в создании данного руководства.

Адрес для отправки материалов и взаимодействия по данному Руководству. device@samovar-tool.ru

Просьба, отправлять текст в форматах .doc или .txt, изображения в .jpg. Текст отдельно, а фото отдельно. Можно все объединить в архивный файл.

Руководство создано на основе общей справки. При обнаружении неработающей ссылки её нужно исправить или заменить ссылкой на актуальный раздел.

- Если некий раздел находится "не в том месте", это не является проблемой - перемещается он по дереву элементарно. Так что пока не заостряем внимания.

Что предлагается:

Структура руководства может уточняться по мере развития проекта.

Каждый раздел должен будет содержать максимальные сведения относящиеся к конкретному оборудованию:

Фото, ссылку где купить, оригинальное название, чтобы найти когда ссылка умрет, как подключать, особенности, возможные замены, переделка и пр.

Крайне желательно ссылку на пост обсуждения девайса на форуме - вам найти свои посты будет не трудно, а кому-то нужно будет весь форум перечитать, и не единожды... .

Т.е. один раздел под конкретный девайс, если раздел получается громоздкий - легко сделать подраздел. (Например как про [регуляторы](#).)

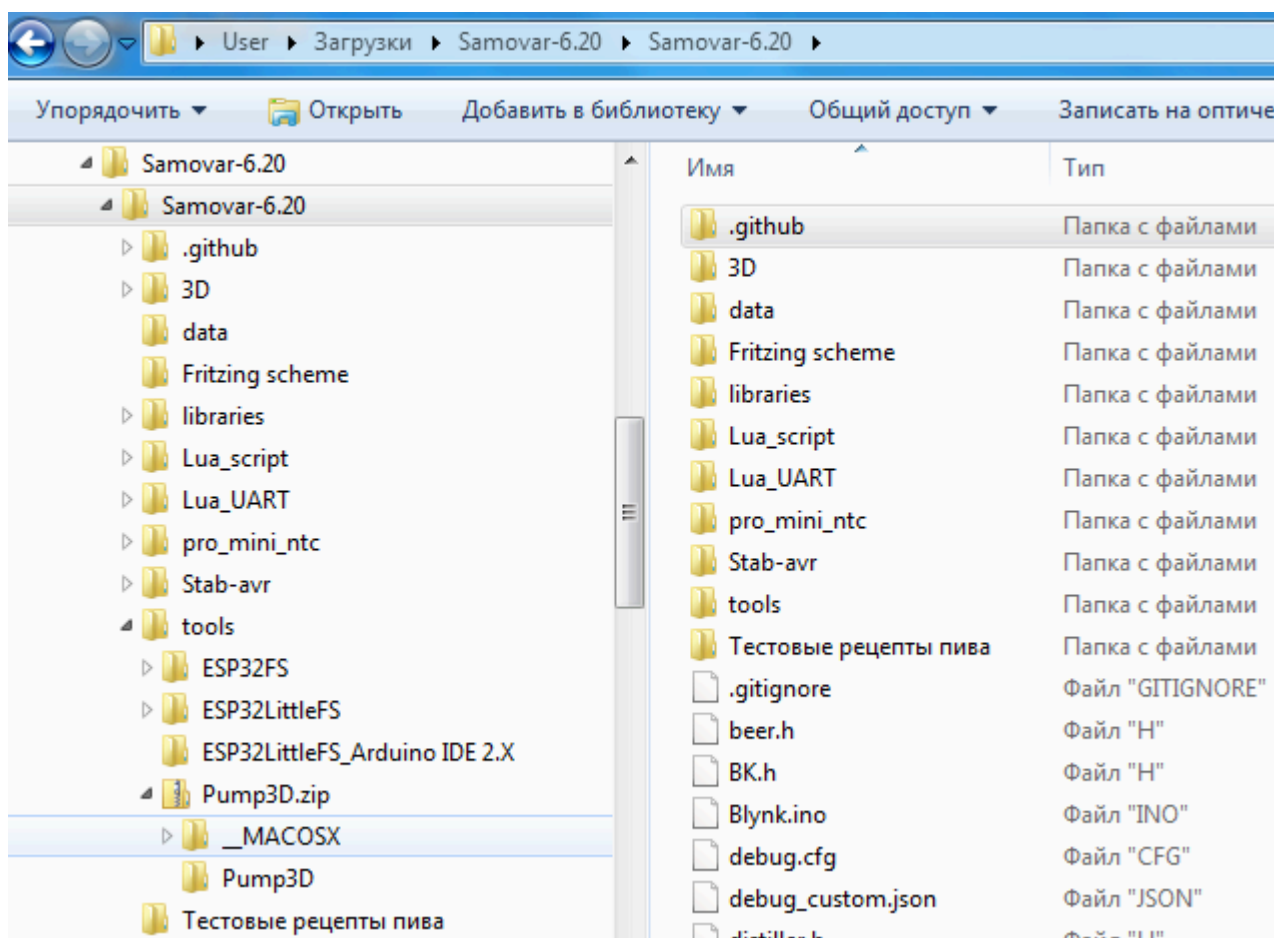
Более о конкретном девайсе никаких дополнительных сведений в других разделах. Упоминаний - сколько угодно. Сделаем ссылку на конкретный раздел и все. Иначе можно элементарно запутаться при составлении.

Могут быть исключения - например датчик давления в кубе. Начало описания реализации на MPX5010D при подключении конкретно к Самовару, а продолжение в разделе Измерителя от dranek.

Авторство. Если кто пожелает, ссылка на автора материала будет в конце раздела.

С чего начать

Для начала скачать [релиз](#), с последней версией Самовара, распаковать на компьютере и ознакомиться с его содержимым.



Что находится в каталогах:

Примечание

Начните с файла **README.md** в корне проекта: в нём перечислены поддерживаемые платы, ограничения выводов и порядок установки.

- **.github** - программный каталог.
- **3D** - модели для 3D печати. Столы разбора по фракциям, воронки и пр.
- **data_raw** — исходники web-интерфейса. Каталог **data** — готовый образ файловой системы устройства; он целиком пересобирается командой **python3 tools/build_web_assets.py**, поэтому вручную его не правят. Файлы **program_*.txt** содержат программы по умолчанию.
- **Fritzing scheme** - схемы по сборке.
- **libraries** - программный каталог (библиотеки).
- **Lua_script** - описание скриптового языка Lua и различные, весьма полезные, скрипты.
- **Lua_UART** - как изготовить выносной автономный монитор COM порта.

- [pro_mini_ntc](#) - как изготовить приставку для Самовара (Измеритель) может работать как самостоятельное устройство в качестве измерительного устройства.
- [Stab-avr](#) - версия стабилизатора мощности из готовых элементов, которым может управлять Samovar.
- **tools** — конфигуратор (samovar_configurator.py), скрипты сборки web-интерфейса и автоматические проверки проекта.
- doc - описание подсистем прошивки (файлы Markdown).
- **partitions** и **ESP32-S3** — таблицы разделов флеш-памяти для ESP32 и ESP32-S3; их использует PlatformIO.
- Stab-avr, Lua_UART, pro_mini_ntc - отдельные скетчи для Arduino, в сборку Самовара не входят.
- Тестовые рецепты пива - несколько готовых рецептов пива для загрузки в Самовар.

В какой минимальной конфигурации можно собрать Самовар?

Для первого запуска достаточно платы ESP32 и USB-питания. Прошивку можно собрать как [B Arduino IDE](#), так и в PlatformIO. Настройки оборудования находятся в **Samovar_ini.h**; пользовательские переопределения, которые нужно сохранять при обновлении, — в **user_config_override.h**. Готовые бинарные файлы в релиз не входят.

Подробные инструкции: [сборка в Arduino IDE](#) и [сборка и прошивка через PlatformIO](#).

Примечание:

Датчики и исполнительные устройства, наличие которых Самовар не контролирует, по причине отсутствия обратной связи:

- *Контактор/ включение РМ (реле №1)*
- *Реле дополнительного нагревателя/ Разгон РМ (реле №4)*
- *Биппер*
- *Клапан открытия воды (реле №3)*
- *Датчик уровня флегмы*
- *Шаговый двигатель перистальтического насоса*
- *Сервопривод стола разбора*
- *Насос охлаждения*
- *Датчик атмосферного давления*
- *Датчик давления в кубе.*
- *Мешалка (реле №2)*

Датчики и исполнительные устройства, наличие которых Самовар контролирует, если:

- *Дтчики температуры - если датчик присутствовал в начале процесса. Сообщения, в случае его отсутствия, будут приходить регулярно.*
- *Датчик потока воды – если использование разрешено.*

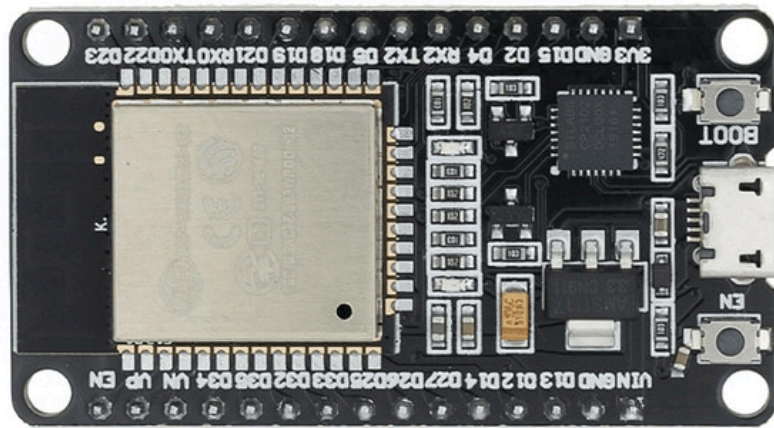
- Регулятор нагрева – если включена проверка использования регулятора.

Затем [настройте подключение к Wi-Fi](#) и откройте web-интерфейс Самовара. На классическом ESP32 DevKit команда «Включить/Выключить нагрев» переключает GPIO2, который на многих платах связан со встроенным синим светодиодом. На ESP32-S3 наличие такого светодиода зависит от конкретной платы.

Познавательная статья о [ESP32](#)

Пояснение.

ESP32 - это чип, который устанавливается на плату. Дополнительно на плате присутствуют преобразователь напряжения, модуль Wi-Fi, блютуз и может быть много еще чего. Здесь понятие ESP32 - это плата с установленным чипом.



ESP-32 30P

Модификаций плат существует множество.

Основная плата проекта — **ESP32 DevKit V1 30 pin**, окружение PlatformIO **Samovar**. Также поддерживаются **LILYGO T-Relay 4** (то же окружение **Samovar**, плата выбирается в конфигураторе) и **ESP32-S3 DevKitC-1**, окружение **Samovar_s3**. У LILYGO и ESP32-S3 другая распиновка (см. `Samovar_pin.h`); печатная плата Samovar 1.3.2 для 30-контактного DevKit к ним не подходит.

ESP32 - это чип

DevKit V1 - версия

30 pin - количество контактов (ног, штырьков) на плате

CH340C - это чип адаптера передачи данных шины USB

Type-C - формфактор разъема USB

Где купить.

[Алиэкспресс](#) (Искать как: Умная плата ESP32, флэш-накопитель USB CH340C WiFi + Bluetooth, сверхнизкое энергопотребление)

[Озон](#) (Искать как: ESP32 WROOM-32 Плата для разработки WiFi Bluetooth-совместимая плата для разработки IOT)

Если сомневаетесь в правильности выбора модели ESP32 - посоветуйтесь на форуме.

Теперь нужно подключить Самовар к оборудованию.

Сначала нужно определиться каким образом будет [монтироваться ESP32](#).

Затем определиться с вариантами компоновки устройств. Это будет моноблок или несколько отдельных блоков, связанных между собой.

И наконец начинать сборку от [минимальной конфигурации](#), постепенно добавляя необходимые компоненты, до решения максимально поставленных задач.

Вопросы и ответы. FAQ

*** Можно ли использовать плату ESP32 DevKit V1 38 pin с 38 контактами?**

*Да можно, но установить ее на печатную плату разработанную для **ESP32 30 pin** не удастся.*

Установка PlatformIO и первая прошивка Самовара

Samovar поддерживает два полноценных способа установки: Arduino IDE и конфигуратор на основе PlatformIO. На этой странице описан PlatformIO. Для Arduino IDE используйте раздел [«Подготовка ESP32 к загрузке»](#).

Конфигуратор позволяет выбрать плату и оборудование, сохранить настройки, прошить ESP32, отдельно загрузить LittleFS и открыть монитор порта. PlatformIO — программа, которая собирает исходный код и записывает его в плату.

Windows 10/11 (только 64-разрядная): скачайте [последний выпуск Samovar](#), распакуйте архив в простой путь, например **C:\Samovar**, подключите ESP32 USB-кабелем с передачей данных и дважды щёлкните **flash_windows.bat**. Батник проверит Python и PlatformIO, при необходимости установит их и откроет окно настройки.

macOS и Linux: откройте Terminal в распакованной папке проекта и выполните:

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

Скрипт проверит Python с Tkinter и PlatformIO, при необходимости установит их и откроет тот же конфигуратор.

Если macOS сообщает Operation not permitted или getcwd: cannot access parent directories:

перенесите папку проекта из Downloads в домашнюю папку и запустите скрипт там. Либо разрешите Terminal доступ к папке «Загрузки» в «Системные настройки → Конфиденциальность и безопасность → Файлы и папки», затем полностью перезапустите Terminal.

В macOS для автоматической установки Python и Tkinter нужен Homebrew. В Linux поддерживаются apt, dnf и pacman; система может запросить пароль sudo — пароль администратора компьютера.

Работа в конфигураторе: выберите ESP32 DevKit, LILYGO или ESP32-S3, задайте оборудование и при необходимости исходные данные Wi-Fi. Остальные параметры разложены по вкладкам «Основные», «Температуры», «Регулятор», «БК», «НБК», «Датчики», «Насосы», «Оборудование», «Шаговый двигатель», «Ректификация», «Сеть» — это те же директивы, что раньше правили в Samovar_ini.h. Нажмите **«Сохранить настройки»**, затем **«Прошить»**. Настройки сохраняются в файл **user_config_override.h** рядом с проектом, при обновлении исходников он не затирается. Журнал сборки и прошивки находится в нижней части окна.

Для новой платы отдельно нажмите **«Загрузить LittleFS»**. LittleFS — область памяти с веб-интерфейсом и пользовательскими файлами. Повторная загрузка может заменить программы, Lua-файлы и журналы, поэтому конфигуратор запрашивает подтверждение. Перед повторной загрузкой сохраните нужные пользовательские файлы через файловый редактор Самовара.

Кнопка **«Монитор порта»** открывает отдельное окно диагностики (скорость 115200). Если прошивка долго стоит на Connecting..., удерживайте BOOT, коротко нажмите RESET/EN и отпустите BOOT после начала записи.

Обновление по Wi-Fi. Последующие прошивки можно ставить без кабеля. Нажмите **«Обновить»** рядом с полем «Порт или адрес»: кроме COM-портов в список попадают устройства Samovar, найденные в сети, и адрес, который устройство сообщило после кнопки **«Получить IP»** в мониторе порта. Адрес можно ввести и вручную (например 192.168.1.37 или samovar.local). Выберите устройство вместо COM-порта и нажмите те же «Прошить» или «Загрузить LittleFS». Условия: прошивка на устройстве собрана с включённым **«Разрешить обновление по Wi-Fi»** (вкладка «Сеть»), компьютер и Самовар в одной сети, брандмауэр разрешает python.exe входящие подключения. Полная очистка флеша, монитор порта и **«Перезагрузить ESP»** работают только по USB.

Кнопка **«Редактор файлов»** открывает файлы устройства (программы, Lua-скрипты, страницы web-интерфейса) прямо в конфигураторе, **«Открыть в браузере»** — web-интерфейс Самовара.

Ручной способ через PlatformIO:

```
pio run -e Samovar -t upload — ESP32 DevKit или LILYGO;
```

```
pio run -e Samovar_s3 -t upload — ESP32-S3;
```

pio run -e <окружение> -t uploadfs — загрузить LittleFS;

pio run -e <окружение> -t monitor — открыть монитор порта.

Готовые окружения с другим регулятором: **Samovar_rmvk** (PMB-K), **Samovar_sem** (SEM_AVR), **Samovar_no_power** (без регулятора с UART); **Samovar_lua** — с Lua, **Samovar_alarm_button** — с аварийной кнопкой. Пользовательские параметры для ручной сборки задаются в **user_config_override.h** по образцу **user_config_override.example.h**.

После первой прошивки перейдите к [подключению Самовара к Wi-Fi](#). Если данные сети уже заданы в конфигураторе и в памяти платы ещё нет другой сети, прошивка попытается подключиться к ним автоматически.

Монтаж ESP32

К ESP32 подключается много периферийного оборудования. Поэтому ESP32 нужно разместить и организовать подключения.

Важно. Приведённая ниже распиновка и печатная плата версии 1.3.2 относятся к ESP32 DevKit V1 30 pin. Для LILYGO T-Relay 4 (окружение **Samovar**, плата выбирается в конфигураторе) и ESP32-S3 (окружение **Samovar_s3**) используйте распиновку из [Samovar_pin.h](https://samovar-pin.h); механически плата 1.3.2 к ним не подходит.

Назначение в проекте ← Разъем на плате	Обозн.	Не вывода	ESP-WROOM-32 DevKitC v1	Не 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]

Ограничения выводов ESP32. Выводы GPIO34–39 — только входы, внутренних подтяжек у них нет: для аварийной кнопки (D35), универсальной кнопки (D39) и датчика потока (D36) обязателен внешний резистор 10 кОм к 3,3 В, иначе будут ложные срабатывания и уход в аварию сразу после включения. На печатной плате эти резисторы уже предусмотрены. Провода к датчику потока ведите отдельно от силовых кабелей ТЭНа. Вывод D15 (реле №2, мешалка) при сбросе аппаратно подтянут — реле на нём кратковременно включается при каждой загрузке; используйте модуль с активным низким уровнем, полярность реле задаётся в настройках Самовара (поля реле 1–4). Подтяжки шины I2C (дисплей, датчик давления, I2CStepper) — 4,7–10 кОм.

Использование печатной платы.

Для использования ESP32_30pin с Самоваром разработана специальная печатная плата. На неё устанавливается:

- непосредственно ESP32,
- преобразователь питания с 12 на 5 вольт,
- драйвер шагового двигателя,
- звуковой бипер,

- конденсаторы и резисторы нужно установить все. Рекомендованные номиналы:

Конденсаторы:

C1, C2 - Электролитические конденсаторы 500мфХ25 вольт (или 1000Х25)

C3, C4 - Электролитические конденсаторы 500мфХ16 вольт (или 1000х16)

Примечание

В рекомендациях приведены номиналы с большим запасом по напряжению.

При запасе по напряжению, даже не очень качественный электролит будет служить дольше и надежнее.

Но если есть желание и уверенность в комплектующих, то можно:

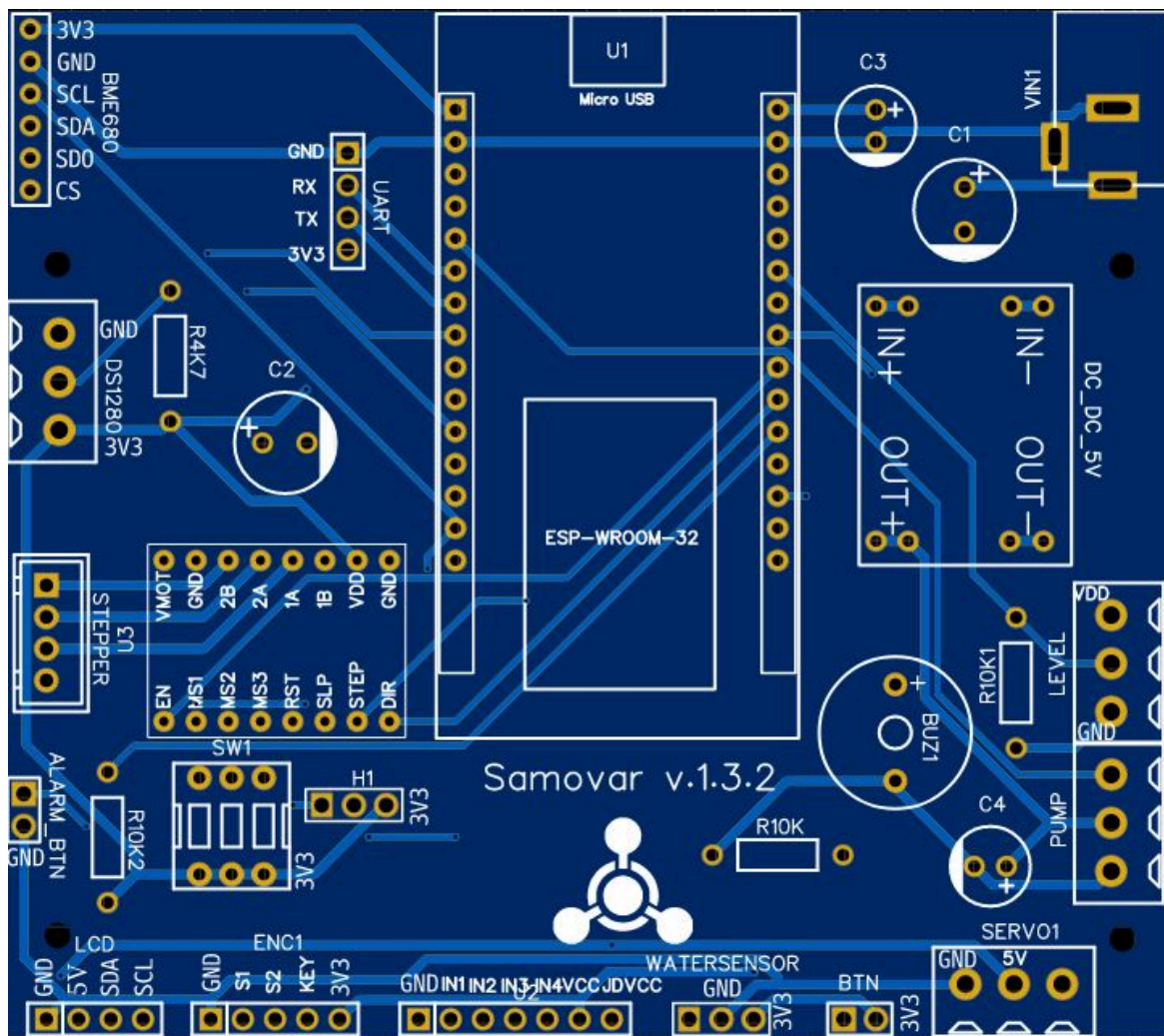
C1, C2 поставить на напряжение 16 вольт

C3, C4 - на 6,3 вольта.

Резисторы: R1-R3 = 10 кОм, R4=4.7 кОм . Все 0.125 ватт.

- разъемы подключения. (прим. Часть разъёмов можно не устанавливать, а распаять непосредственно проводами.)

Рекомендуется установить сразу на плату все элементы: бипер, датчик атмосферного давления, резисторы и конденсаторы, гребенку для драйвера шагового, переключатели и разъемы. Даже если, в данный момент, не планируется использование некоторого оборудования. В дальнейшем это существенно облегчит расширение возможностей Самовара.



Места установки и элементы:

- [ESP-WROOM-32](#) место установки [ESP32](#). Рекомендуется устанавливать на разъем типа гребенка (мама)
- [BME680](#) Место для подключения датчика атмосферного давления (BMP180, BMP085, BMP280, BME280, BME680).
- [DS18B20](#) подключение датчиков температуры. Справа (на плате) - резистор подтяжки по питанию 4,7 кОм для датчиков.
- [STEPPER](#) - подключение шагового двигателя (ШД) NEMA17 перистальтического насоса отбора. Справа место установки драйвера ШД.
- [SW1](#) и [H1](#) - делитель шагов ШД и дополнительное управление драйвера ШД
- [ALARM_BTN](#) Подключение аварийной кнопки и аварийных датчиков.
- [LCD](#) - подключение дисплея LCD2004 (любой другой не поддерживается)
- [ENC1](#) - разъем для энкодера с кнопкой.
- [IN1IN2IN3IN4](#) подключение модуля реле.

- WATERSENSOR - разъем датчика потока охлаждения
- [BTN](#) - многофункциональная кнопка
- [SERVO1](#) - подключение сервопривода стола разбора
- [PUMP](#) - управление насосом охлаждения
- [LEVEL](#) - подключение датчика уровня флегмы
- [BUZ1](#) - активный зуммер 3 вольт
- DC_DC_5V - преобразователь напряжения 12-5 вольт.
- VIN1 - разъем подключения 12 вольт
- C1, C2 - Электролитические конденсаторы 500мфX25 вольт (или 1000X25)
- C3, C4 - Электролитические конденсаторы 500мфX16 вольт (или 1000x16)
- Все резисторы на 10кОм, кроме одного резистора на 4,7кОм

Где взять плату?

- Спросить в теме поддержки. Возможно, у тех кто уже собрал, есть лишняя и могут поделиться.
- Заказать самостоятельно.

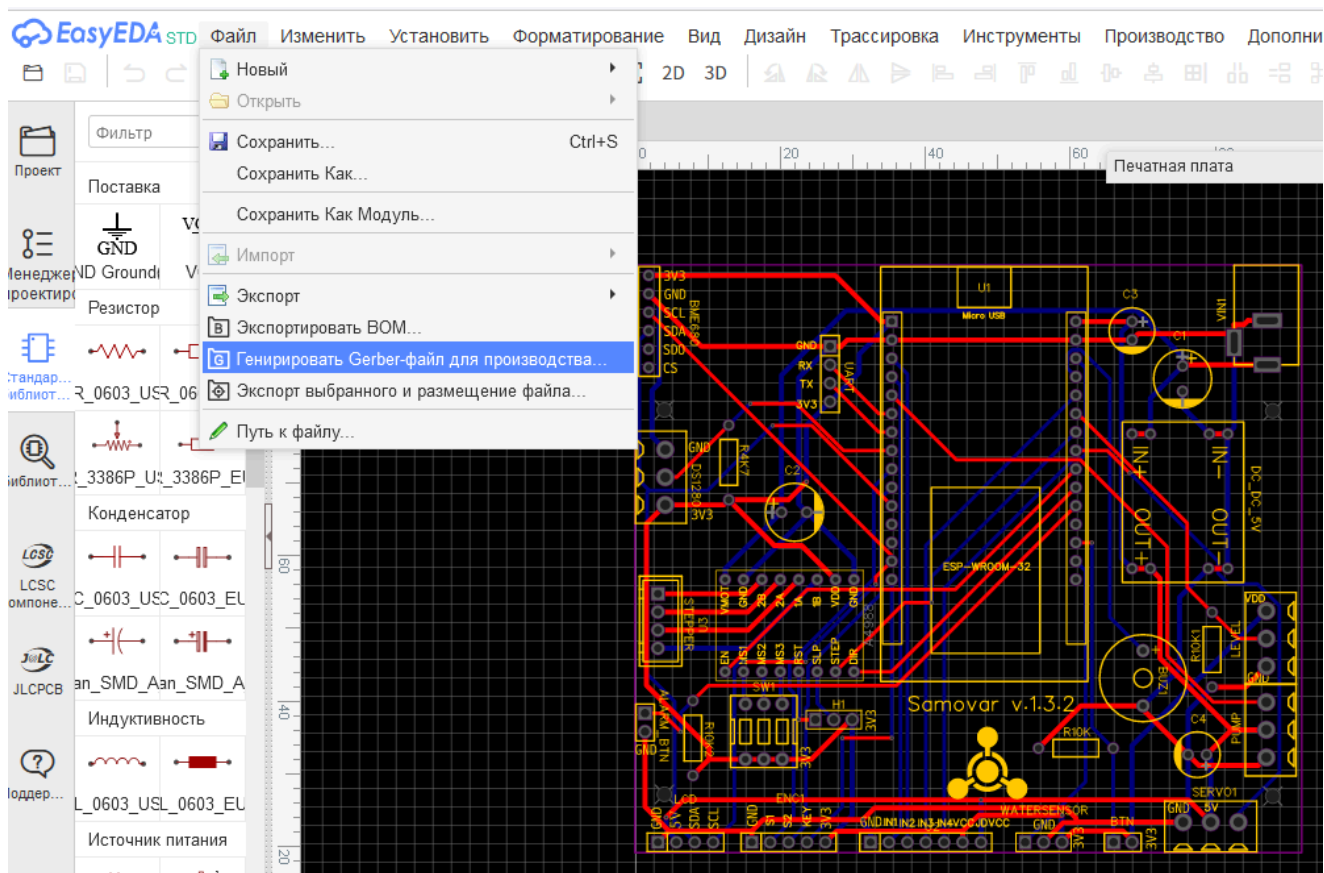
Разработанная плата находится на сервисе easyeda.com. В данный момент заказать изготовление через это сервис не получится. Но пользоваться сервисом можно.

Для производства нам нужен файлы в формате Gerber, представляющий собой описание проекта печатной платы для изготовления.

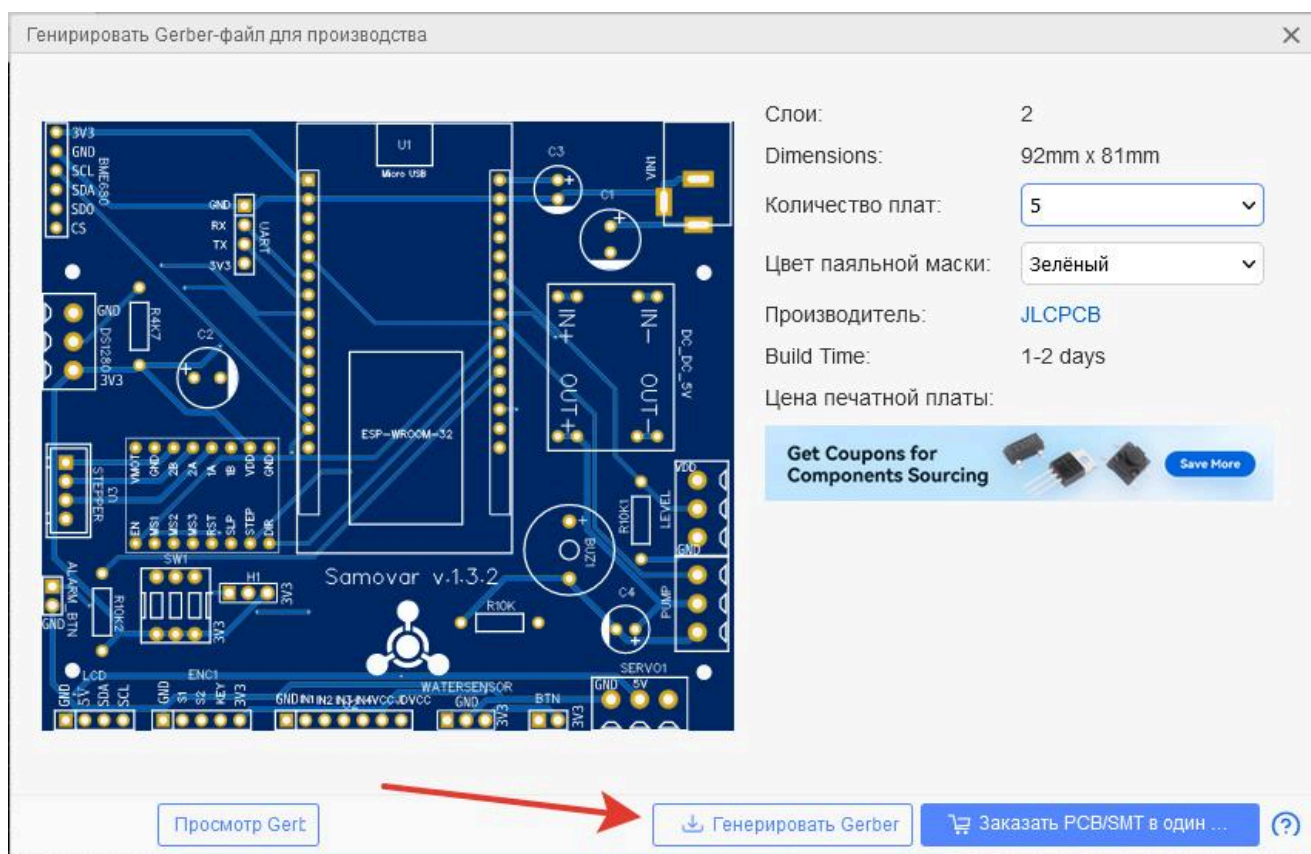
Архив Gerber для производства печатной платы версии 1.3.2, по состоянию на сентябрь 2024 г., можно скачать [по ссылке](#) с нашего сервера.

Или:

Идем через меню: Файл>Генерировать Gerber-файл для производства...



Нажимаем, после запроса на проверку, выбираем: "Нет, генерировать Gerber-файл", Появится окно



Можно изменить:

- количество плат - кратно 5 (меньше 5 заказать нельзя).
- цвет паяльной маски - это цвет платы. Выбирайте зеленый или синий. За другие цвета могут брать дополнительную плату.

Нажимаем генерировать Gerber. Созданный файл в виде zip-архива сохранится на ваш компьютер.

Теперь нужно заказать изготовление.

Сервисов, которые производят и самое главное - отправят, вполне достаточно. Например nextpcb.com инструкция есть на сайте или в интернете, оплата через Алиэкспресс.

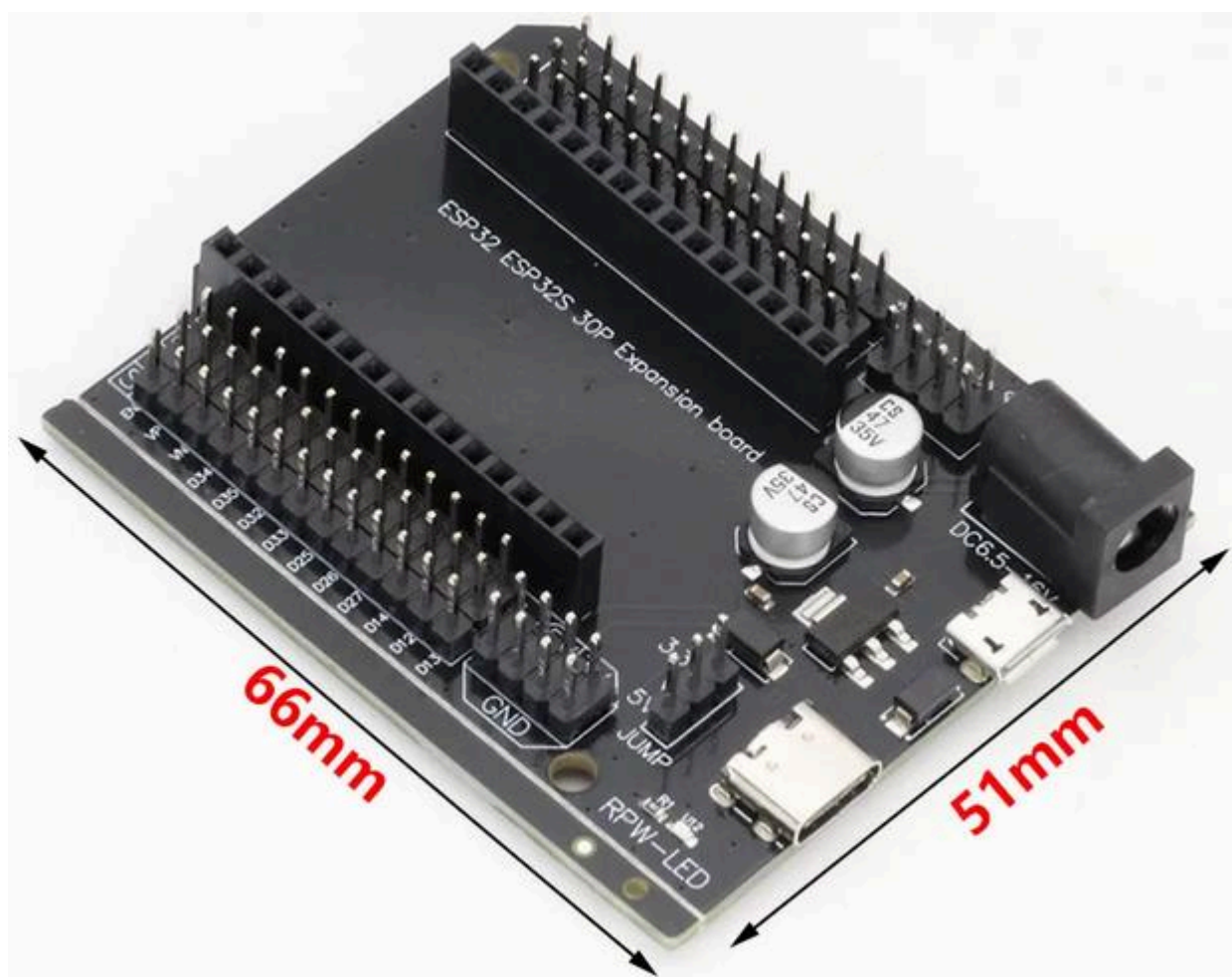
Или сразу идем на Алиэкспресс и забиваем в поиск "изготовление печатных плат на заказ". Появятся предложения продавцов. Выбираем по отзывам.

Принцип заказа один - вы отправляете продавцу полностью архив Gerber (если имя на кириллице - переименуйте на латиницу). Он определяет стоимость и сообщает вам. Устраивает, оплачиваете - через примерно месяц печатные платы у вас.

Стоимость 5-ти плат с доставкой, приблизительно 15-20 у.е.

Использование макетной платы (шилд)

Для ESP32 выпускается специальная плата ([Макетная плата расширения ESP32 Shield 30Pin](#)) позволяющая проводить быстрое макетирование различных устройств. Выглядит так:



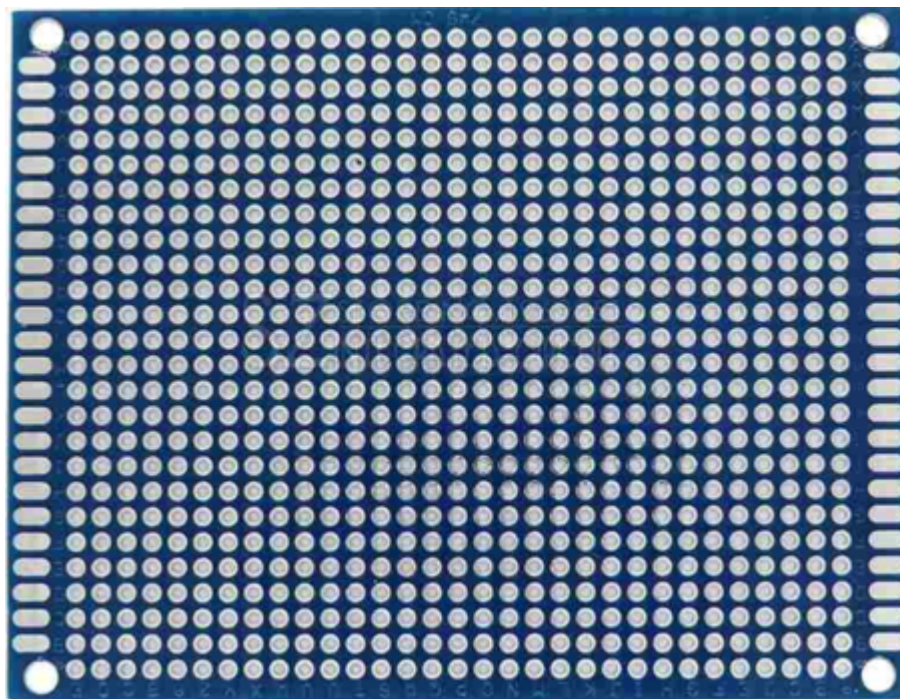
На шилде уже есть преобразователь питания 16...6,5-5v и разъемы USB - питание 5 вольт. ESP32 вставляется в гребенку-мама, а к штырькам (гребенка-папа) подключается периферия. Весь монтаж придется делать проводами, резисторы и часть конденсаторов распаять либо на самом шилде, либо на разъемах корпуса.

Использование шилда неплохое решение, к тому же весьма не дорогое. Но при окончательной сборке крайне желательно все провода припаять, а не оставлять на разъемных соединениях, иначе отказы будут следовать один за другим.

Шилды выпускаются и для плат ESP32_38 pin., что позволяет использовать ESP32 и с 38 выводами.

Использование макетной платы для пайки.

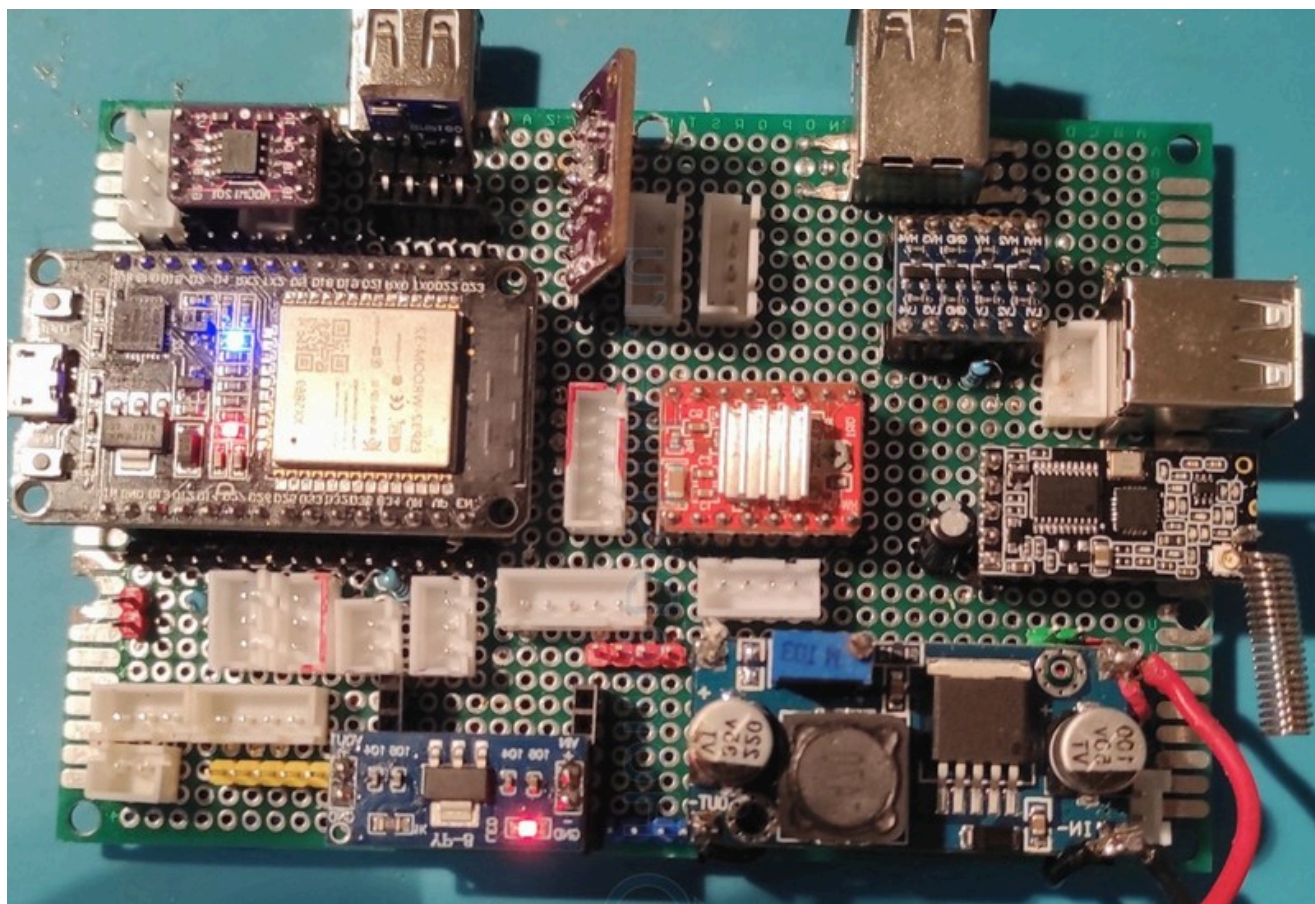
Можно использовать печатную макетную плату предназначенную для пайки элементов. (Прототип печатной платы, макетная плата, универсальная плата)



На такой плате можно разместить все элементы и даже останется место для будущих модернизаций. Но для работы с такой платой нужно иметь определенный навык. Хочется предупредить - собрать на такой плате Самовар - это не совсем просто.

Но тем не менее, проект может быть реализован подобным способом.

Пример сборки Самовара на подобной макетной плате:

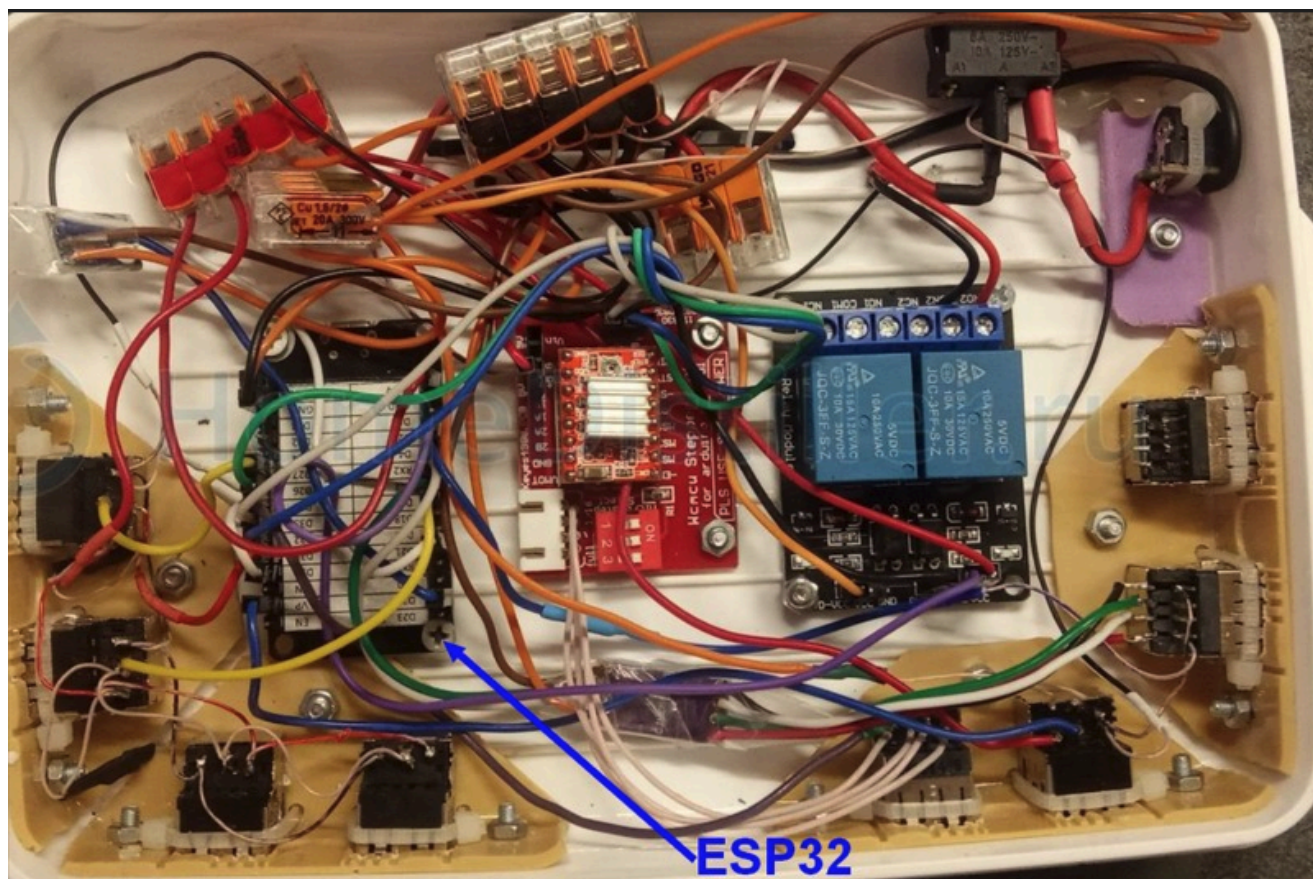


Больше фото [в теме на форуме](#).

Сборка без платы

Ну и последний вариант - без всяких печатных плат. Навесной монтаж. На принтере печатается бирка с обозначением выводов, наклеивается снизу на ESP32, которая переворачивается вверх ногами и прикручивается к корпусу на стойках (для охлаждения). Монтаж проводников на выводы ESP32 осуществляется посредством разъемов типа Dupont.

Пример реализации:



[Посмотреть на форуме](#)

Рекомендации:

- Наличие печатной платы весьма существенно облегчит сборку и последующую работу по настройке. Но самое главное избавит от большей части проблем связанных с возможными ошибками и плохими соединениями между ESP32 и периферией.
- Если есть желание начать сборку, а специальной печатной платой никто не поделился и самому заказывать изготовление нет желания - использование шилда наиболее приемлемый вариант.

Минимальная конфигурация

Минимальная конфигурация ниже описывает наблюдение без автоматического управления силовой частью.

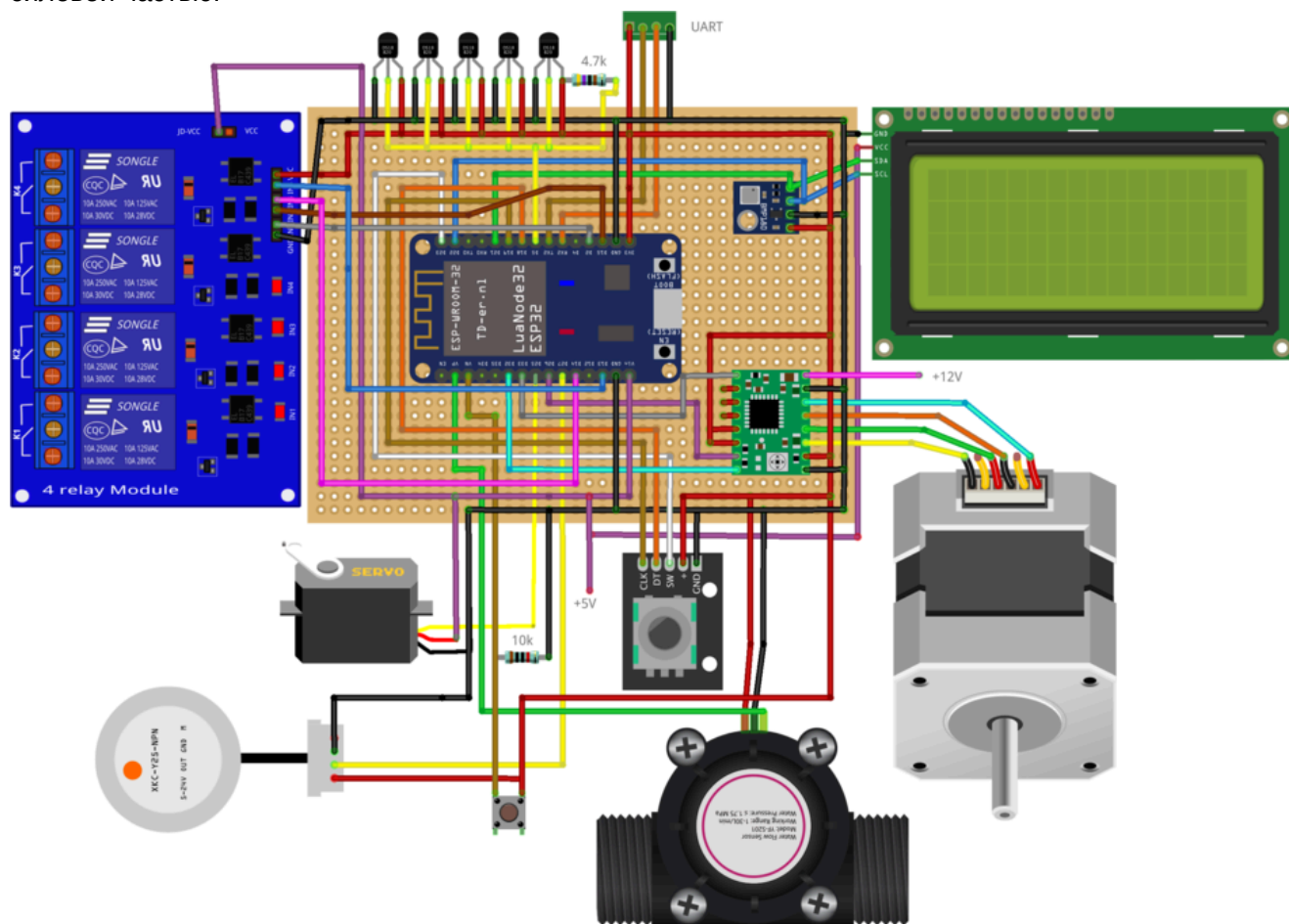


Схема в большем разрешении находится в архиве с релизом. Каталог Fritzing scheme.

Подразумевается, что при монтаже ESP32 на печатную плату были также установлены [датчик атмосферного давления](#) и [зуммер](#).

Далее нужно подключить [датчики температуры](#). А поскольку, обычно, компьютер находится в одном месте, а колонна в другом, то подать питание ESP32 можно с любой зарядки для телефона. Вот тут уже Самовар превратится в вашего наблюдателя. Он будет выводить на монитор и присылать на ваш смартфон сообщения: разгон завершен, включите воду, стабилизация завершенна, температуры превышены, пауза и пр. И при этом вы видите все температуры. В случае прорыва пара выскакивает сообщение, есть графики, можно анализировать поведение колонны и пр.

Но стабильность и безопасность обеспечиваете вы сами. Управлять он пока ничем не может. Только информирует.

Использование Самовара только с датчиками температуры и питанием от зарядки можно назвать скорее тестовым. Позволяющим убедиться в стабильности работы ESP32 и сети WiFi. Дальнейшая сборка потребует размещения платы Самовара в корпусе и использование более серьезного источника питания. Если планируется использование насоса охлаждения, понадобится один блок питания 12 вольт 80-100ватт.

Можно использовать два блока питания: 40-50 ватт для питания платы Самовара и шагового двигателя и 60-70 для насоса охлаждения.

Следующие этапы:

- Стабилизированный [регулятор напряжения](#), позволит практически полностью доверить Самовару управление нагревом. Разгон, стабилизация, поддержание температуры и отключение нагрева по окончании процесса.
- [Аварийное отключение](#) крайне желательно сделать сразу после подключения регулятора.
- [Организация охлаждения](#) полностью автоматизирует поддержание заданной температуры на выходе, открывает воду в начале и закрывает конце работы.
- [Датчик уровня флегмы](#) контролирует захлеб колонны, в связке с регулятором напряжения Самовар автоматически уменьшит нагрев и не допустит выброса жидкости через ТСА. Использование этого датчика позволяет автоматически поддерживать предзахлебный режим в колонне на протяжении всего периода отбора.
- [Насос отбора](#)
- [Распределение по фракциям](#)
- _

Зуммер

Служит для оповещения о событиях: сигнал — серия из пяти коротких гудков. Включается и отключается в web-интерфейсе: «Настройки» → вкладка «Основные», флажок **«Использовать зуммер для нотификации»** (при выключенном флажке зуммер отключён полностью). Там же флажок **«Включать зуммер при смене программы»**.

Подключается прямо к выводу ESP32 (на печатной плате — разъём BUZ1): D12 на ESP32 DevKit, GPIO0 на LILYGO T-Relay 4, GPIO39 на ESP32-S3.



Активный зуммер DC 3V (красная этикетка); 5V (синяя этикетка); 12V (черная этикетка)

Используется активный зуммер. Пассивный работать не будет. При установке необходимо соблюдать полярность. Обозначение должно быть на корпусе или - длинная нога это плюс.

Если есть в наличии какой-то зуммер как определить активный он или нет. Выставляем на источнике питания 3 вольта, подключаем минус, плюсом несколько раз прикасаемся к другому контакту. Если никаких звуков нет, поднимаем напряжение до 5 вольт.

Если слышен звук, писк - активный, если слышны только щелчки при касания плюса - пассивный. Тоже самое для 12 вольт.

Рекомендуется использовать зуммер на напряжение 3 вольта, но подключали и на 5 вольт. Вполне приличная громкость.

Регулировка громкости зуммера не предусмотрена. Можно попробовать поставить резистор по питанию, а можно просто слегка заклеить отверстие скотчем.

Датчики атмосферного давления

Samovar 7.00: тип датчика выбирается на вкладке «Датчики» конфигуратора; ручное изменение `Samovar_ini.h` требуется только при нестандартной сборке.

Примечание.

Самовар работает и без этого датчика, но лучше установить, тем более что стоимость датчика весьма скромная.

В проекте можно использовать датчики BMP180, BMP280, BME280, BME680. Некоторые из этих датчиков могут измерять не только атмосферное давление и температуру, но и влажность и даже качество воздуха.

Самовар использует только данные об атмосферном давлении и поэтому никакой разницы какой из датчиков будет использован - нет.

Датчик подключается к шине I2C вместе с дисплеем: SDA D21 и SCL D22 на ESP32 DevKit, GPIO33/GPIO32 на LILYGO T-Relay 4, GPIO8/GPIO9 на ESP32-S3. Показания переводятся в мм рт. ст. и используются для поправки температуры кипения.

На печатной плате предусмотрена возможность распайки датчика на 6 pin из которых только 4 имеют подключения.

Поэтому, если не использовать из имеющихся в наличии, а приобретать, то лучше взять простой и надежный BMP180. Более подробно прочесть об этом датчике можно [в обзоре](#).

Никаких дополнительных настроек датчик не требует.

Внимание

*Поскольку датчики работают по шине I2C некоторые датчики **BMP280** могут иметь другой адрес. В таком случае надо в файле `Samovar_ini.h` раскомментировать строку `#define USE_BMP280_ALT`*

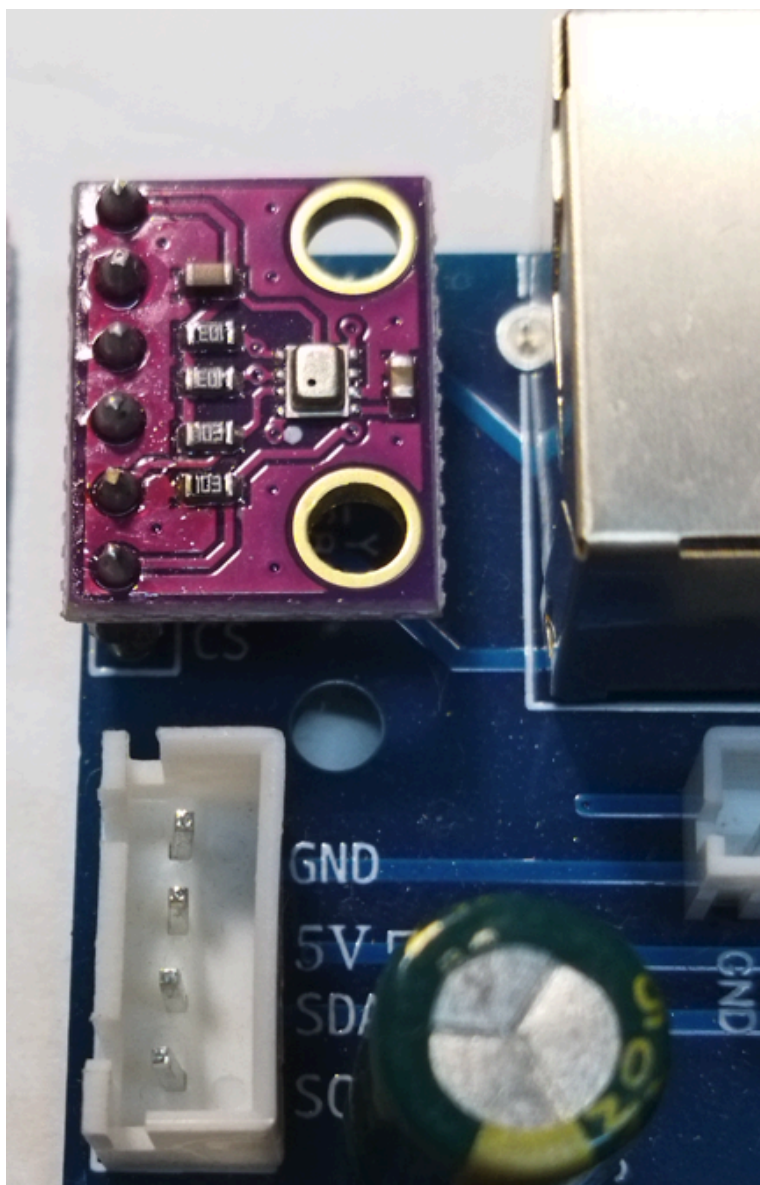
Т. е. должно выглядеть так:

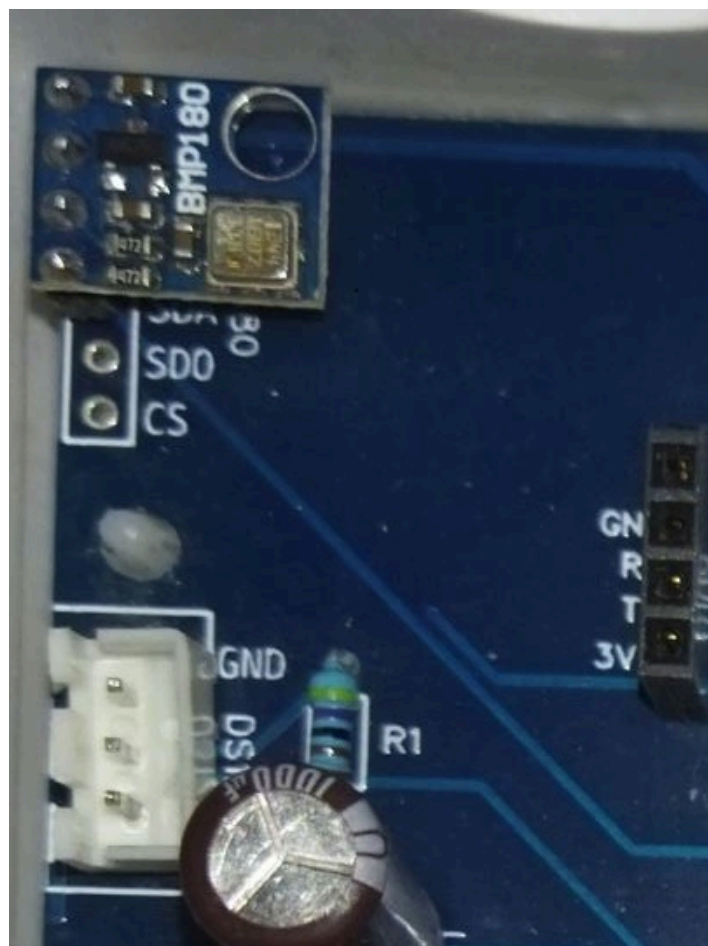
...

```
//#define USE_BMP280 //использовать датчик давления BMP280  
#define USE_BMP280_ALT //использовать датчик давления BMP280 с  
альтернативным адресом.
```

*В штатной конфигурации Samovar 7.00 включён BME680. В конфигураторе датчик выбирается на вкладке «Датчики» в поле «Атмосферное давление»: «BMP180/BMP085», «BMP280», «BMP280, альтернативный адрес», «BME280», «BME680» или «Не использовать». При сборке в Arduino IDE оставьте активной ровно одну соответствующую директиву в **Samovar_ini.h**; для BMP280 с альтернативным адресом — `USE_BMP280_ALT`.*

Распаять плату с датчиком лучше непосредственно на саму печатную плату.





Датчики температуры

Самовар использует цифровые датчики **DS18B20**. Подключайте их по трёхпроводной схеме: питание 3,3 В, GND и общая линия данных с подтягивающим резистором 4,7 кОм к 3,3 В.

Линия данных: GPIO5 на ESP32 DevKit, GPIO12 на ESP32-S3 и GPIO27 на LILYGO T-Relay 4. Все датчики подключаются параллельно к одной линии; прошивка различает их по уникальным адресам.

Поддерживаются пять назначений: температура пара, царги, охлаждающей воды или флегмы, куба и ТСА. После монтажа назначьте найденные адреса соответствующим датчикам в настройках Самовара («Настройки» → вкладка «Температура», поля «Адрес датчика пара перед дефлегматором», «Адрес датчика в царге», «Адрес датчика воды», «Адрес датчика в кубе», «Адрес датчика в ТСА») и проверьте показания до запуска нагрева. Датчики опрашиваются с разрешением 12 бит. Если датчик был подключён в начале процесса и пропал, Самовар сообщает об этом регулярно.

Для длинных линий используйте качественный кабель, надёжные соединения и отдельное питание датчиков. Паразитное питание по двум проводам не рекомендуется.

Используемые регуляторы напряжения

Samovar 7.00: тип регулятора выбирается на вкладке «Регулятор» конфигуратора: KVIC, PMB-K, SEM_AVR или «Не использовать».

ВНИМАНИЕ!

Все стабилизаторы, подключаемые к Самовару, должны быть физически развязаны по напряжению. Это могут быть реле, оптопары, изоляторы шин данных типа ADUM и пр. Ни в коем случае не подключайте стабилизаторы напрямую, иначе напряжение питающей сети попадет на вас и в Самовар. Самое лучшее что произойдет в этом случае - это сгорит Самовар.

Самовар поддерживает несколько вариантов. В Arduino IDE выбор выполняется директивами в **Samovar_ini.h**. В PlatformIO можно выбрать готовое окружение: **Samovar** — KVIC, **Samovar_rmvk** — PMB-K, **Samovar_sem** — SEM_AVR, **Samovar_no_power** — сборка без UART-регулятора. Одновременно выбирается только один вариант.

Стабилизаторы напряжения имеющие управление по протоколу UART

Алгоритм для регулятора с UART.

Включение: Самовар включает реле №1 и ждёт время **SAMOVAR_USE_POWER_START_TIME** (по умолчанию 2000 мс, в конфигураторе — «Задержка запуска регулятора, мс» на вкладке «Регулятор»; для SEM_AVR перед этим выдерживается ещё 3 с на старт регулятора), после чего передаёт регулятору рабочую команду. Если регулятор не отвечает, нагрев отключается с сообщением «Таймаут команды регулятора». Реле №4 управляется отдельно алгоритмом выбранного режима и не обязано включаться одновременно.

Отключение: реле №1 и №4 выключаются сразу. Через 700 мс регулятору передаётся команда SLEEP, ещё через 200 мс завершается последовательность остановки.

1. Регулятор [KVIC](#)

Полностью совместим с Самоваром. Шаг регулировки на энкодере 0,5 вольт. Стабилизация 0,1 вольт. Имеет возможность аварийного отключения от независимого датчика.

2. Регулятор [PMB-K](#) (с Самоваром работают только версии № 5 и 6)

Все модели PMB-K по умолчанию имеют развязку по питанию на управляющих контактах и переделка для работы с Самоваром не требуется.

шаг регулировки на регуляторе и с Самовара - 1 вольт. Стабилизация 1 вольт. Версия № 6 Имеет возможность аварийного отключения от независимого датчика. Самовар полностью, через PMB-K, управляет разгоном, стабилизацией и аварийным отключением нагрева.

Внимание! PMB-K работает только с целыми значениями, то имеет шаг регулировки 1 вольт, что имеет свою особенность при работе Самовара в [предзахлебном](#) режиме.

3. Регулятор [SEM AVR](#)

Совместим с Самоваром. Имеет особенность: в интерфейсе Самовара подаваемая в нагрузку мощность задается и отображается не в вольтах (как в двух предыдущих регуляторах), а в ваттах.

Для корректной работы с Самоваром нужно использовать скетч Stab-avr.ino, который идет в комплекте с Самоваром и находится в каталоге ...Samovar\Stab-avr

Стабилизаторы напряжения не имеющие управления из вне по протоколам.

Например:

[PM2, PM2M, PM2-Pro](#) (далее просто PM) .

Все модели PM по умолчанию имеют развязку по питанию на управляющих контактах, переделка для работы с Самоваром не требуется.

Управление данными стабилизаторами сводится к двум параметрам:

- Включить/Выключить сам стабилизатор.
- Запустить/ Остановить режим "Разгон"

Функция реализована на двух реле, которые своими контактами управляют регулятором. При команде Самовара "Включить нагрев",

контакты реле №1 включают PM. Реле №4 включает режим "Разгон" только тогда, когда этого требует алгоритм выбранного режима.

Реле №4 разомкнет контакты и PM отключит режим разгона. По окончании перегона Самовар контактами реле №1 отключит PM.

Примечание:

- Все перечисленные регуляторы работают как самостоятельные единицы и могут быть использованы для работы с Самоваром без подключения по протоколу UART по принципу и алгоритму работы регулятора PM.

На Самоваре и регуляторе напряжения разные показания мощности. Как исправить?

По большому счету абсолютная мощность не так и важна, однако добиться одинаковых показаний отображаемой мощности на регуляторе и Самоваре возможно и по своей сути метод только один.

Все регуляторы, в том числе и Самовар не измеряют мощность. Они ее вычисляют как калькуляторы. По двум величинам: текущему подаваемому на нагреватель напряжению и введенному значению сопротивления этого нагревателя.

Введенная величина сопротивления постоянна, а напряжение меняется. Вся настройка сводится к подбору величины сопротивления нагревателя.

Сопротивление можно взять расчетное: напряжение в квадрате деленное на мощность. Напряжение берется из шильдика на ТЭНе. Пример для ТЭНа 3 кВт 220 вольт $220 \cdot 220 / 3000 = 16,1333333333$ Ом.
Или измерить мультиметром.

Для регуляторов **РМВк** и Простой из китайских модулей (**Stab-avr**).

Внимание! Если не будем реально греть куб - отключаем нагреватель.

- В **РМВк** можно ввести значения мощности и напряжения, сопротивление будет вычислено автоматически. Или ввести сопротивление и мощность будет вычислена. [Видео как настроить](#). Смотрим, делаем и не забываем перевести регулятор в режим отображения в ваттах.

- В **Stab-avr**, выставляем потенциометром фактическое напряжение сети в данный момент времени и вводим значение мощности вашего нагревателя.

Внимание! Мощность ТЭНа для Stab-avr нужно вводить из расчета напряжения сети 230 вольт. Если производитель нагревателя указал в паспортных данных мощность для 220 вольт, то нужно пересчитать на напряжение 230 вольт.

Запускаем Самовар — нагрев. В режиме Разгон настраивать что-либо не имеет смысла — напряжение будет все время меняться, мощность соответственно тоже. Даем с Самовара команду — установить какое либо конкретное напряжение на регуляторе, (допустим 170 вольт для РМВк. Или 2000 ватт для Stab-avr). Регуляторы перейдут в режим стабилизации. Сравниваем отображаемые значения мощности в Самоваре и на регуляторе. Теперь есть два пути — либо менять значение сопротивления в Самоваре (WEB-интерфейс. ГЛАВНАЯ - Настройки - Основные - Вкладка Прочие), что гораздо удобнее. Либо менять введенное значение мощности в регуляторе. Задача добиться максимально близких значений на регуляторе и в Самоваре.

Регулятор **Kvic** .

Не имеет показаний мощности. Поэтому настраивается только Самовар на ту мощность которую хотите лицезреть...

Это первый вариант

Второй — это использовать связку: отдельный [вольтамперметр](#) — регулятор — Самовар.



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

Это удобно, если периодически используются разные ТЭНы и(или) коммутируются нагревательные элементы самой ТЭНовой сборки. Вольтамперметр реально измеряет потребляемый ток и выдает на дисплей фактическую мощность. Поэтому если нужно чтобы Самовар показывал эту мощность, то просто оперативно меняем значение сопротивления нагревателя в Самоваре.

Дополнительно смотрите:

- Помощь в выборе: [обзор регуляторов](#)
- Помимо управления регуляторами Самовар может [управлять дополнительным разгонным ТЭНом](#)
- Как подключить регулятор к Самовару без проводов [по радио-каналу](#)

Обзор регуляторов

Обзор поможет определиться с выбором регулятора для работы с Самоваром.

Все представленные ниже регуляторы являются стабилизаторами выходного напряжения. На выходе выдают стабилизированное напряжение вне зависимости от скачков питающей сети. Не являются повышающими. Стабилизация происходит только при условии, что выходное напряжение ниже входного. Могут работать как самостоятельные устройства.

Внимание.

В данном обзоре, положительные и отрицательные функции регуляторов рассматриваются исключительно в аспекте работы с Самоваром.

Регулятор от KVIC

Представляет из себя стабилизатор на базе микроконтроллера Arduino. Можно собрать самостоятельно при условии, что присутствует элементная база и имеются навыки сборки и/или ремонта электронных устройств. Ручное управление - энкодер. Отображение: ЖК дисплей 0,96 дюйма. Можно приобрести готовое устройство. Форм-фактор, комплектация, гарантия и прочие условия обговариваются с изготовителем индивидуально.

Исходный код написан на языке ассемблер, открыт и [доступен](#) для самостоятельного изменения.

Плюсы:

- Быстрый старт, стабилизация напряжения на выходе. До 0,1 вольта
- Ручное управление энкодером с кнопкой. Шаг регулировки 0,5 вольта
- Удобочитаемые показания на дисплее.
- Сигнализация стабилизации напряжения.
- Возможность ручной предустановки напряжения стабилизации, к которому будет переход после выполнения команды «Стоп разгон».
- Управление по UART или I2C
- Ремонтопригодность и возможность обновления программы по мере выхода новых версий. Возможность установки альтернативной прошивки со скоростью работы на порту 9600 бод (по умолчанию 38400 бод)
- Управление разгонным ТЭНом через отдельное силовое реле.
- Разъем для внешней команды «Стоп разгон»

- Разъем для внешней команды «Авария»
- Плавная подача напряжения на нагрузку при старте и плавное снятие при отключении.

Минусы:

- **Все элементы регулятора** находятся под высоким напряжением.
- Для замены программы, возможно нужно будет извлекать микроконтроллер из регулятора.
- Предустановленное напряжение стабилизации не сохраняется в памяти устройства при выключении или перезагрузке.
- Для ремонта и прошивки нужно обладать специальными навыками.

Купить «под заказ» или собрать самостоятельно: [тема на форуме поддержки](#).

Регулятор РМВ-к



Представляет из себя стабилизатор, собранный в корпусе 2DIN, с креплением на дин-рейку. Ручное управление и настройка — кнопками. Отображение информации: трехзначный индикатор. Самостоятельная сборка невозможна.

Стабилизатор имеется в свободной продаже 6 версиях и в 5 комплектациях. (на момент написания обзора).

Внимание! Для работы с Самоваром по UART подходят только версии 5 и 6.

Плюсы:

- Срок службы изделия — 10 лет.
- Гарантия 6 месяцев
- Стабильность удержания напряжения 1 Вольт. Шаг регулировки 1 вольт
- Память на предустановленные режимы 10 значений
- Сохранение в памяти сопротивления ТЭНа для расчета мощности
- Калибровка вольтметра
- Возможность переключения на отображение показаний в ваттах.
- Управление с Самовара по UART (только версии 5 и 6).
- Разъем для внешней команды «Авария» (присутствует на версии 3 и выше).
- Разъем для внешней команды «Стоп разгон» (присутствует на версии 4 и выше).

Минусы:

- В случае ремонта нужно обладать специальными навыками и оборудованием.
- **Все элементы регулятора** находятся под высоким напряжением.
- Обновление микропрограммы невозможно
- Если регулятор используется как самостоятельная единица, для использования функций «Стоп разгон» и «Авария» нужен дополнительный источник питания (достаточно батарейки типа "Крона")
- Сложное ручное программирование и введение значений в память.
- Особенности ручного управления (например, чтобы понизить напряжение нужно сначала поднять его на 1 вольт, только потом понизить).

Купить регулятор можно у разработчика через [тему на форуме поддержки](#) (только регулятор, без симистора и пр.)

Приобрести полностью собранный в бокс, готовый к подключению ТЭНа и работе, в [интернет-магазине](#) (там же можно купить набор из комплектующих для самостоятельной сборки).

Регулятор Stab-avr

Простой стабилизированный регулятор мощности ТЭНа из китайских модулей

Представляет из себя стабилизатор на базе микроконтроллера Arduino. Управление кнопками. Отображение: ЖК дисплей 0,96 дюйма. Можно

собрать самостоятельно, обладая минимальными навыками работы с паяльником. Нет в продаже.

Исходный код написан на языке C++, открыт и может быть изменен самостоятельно с помощью Arduino IDE.

Плюсы:

- Простая сборка из готовых модулей.
- Полная ремонтпригодность.
- Управление с Самовара по UART
- Возможность обновления программы, открытый исходный код.
- Все элементы регулятора находятся под безопасным напряжением, кроме силовых.
- Возможно подключение к Самовару без развязки по напряжению (нужно согласовать напряжение на Arduino NANO и ESP32).
- Внешний разъем «Стоп разгон»
- Внешний разъем «Авария»
- Калибровка вольтметра
- Возможность предустановки и запись в память значения выходной мощности, к которому будет переход после выполнения команды «Стоп разгон».
- Номинальная мощность ТЭНа сохраняется в памяти.

Минусы:

- Относительно долгий старт оригинальной прошивки при работе с Самоваром. Рекомендуется использовать модифицированную прошивку, идущую в комплекте со скетчем Самовара.

Как собрать самостоятельно: [тема на форуме поддержки](#).

Регулятор РМ (PM2M, PM2M-Pro)



Представляет из себя стабилизатор, собранный в корпусе 3DIN с креплением на дин-рейку. Ручное управление и настройка — кнопками. Отображение информации: трехзначный индикатор. Самостоятельная сборка невозможна.

Стабилизатор имеется в свободной продаже у различных производителей. Соответственно, комплектация и гарантии у всех разные. Можно приобрести, как отдельно стабилизатор (без симистора и пр.), так и полностью комплектное устройство готовое к работе.

Плюсы:

- Стабильность поддержания заданного напряжения: плюс-минус 1 Вольт.
- Шаг регулировки 1 вольт
- Индикация потребляемой нагрузкой мощности: от 0 до 9,99 kW.
- Память на 10 предварительных установок напряжения.
- Встроенный таймер напряжение-время: от 0 до 999 минут.
- Не нуждается в дополнительном источнике питания при использовании функций «Стоп разгон» и «Авария»

Минусы:

- Отсутствует канал управления.
- Сложное программирование и введение значений в память.

Выводы.

Для использования с Самоваром рекомендуются первые три регулятора. При условии, что используется управление по UART, Самовар будет управлять регулятором полностью: от включения разгона, до изменения напряжения в любой момент времени и отключения нагрева.

Возможность аварийного отключения от стороннего устройства может быть использована как опция. Например, при использовании Самовара и ПВК в части отключения нагрева в случае превышения давления в пароводяной рубашке. В качестве датчика можно применить электроконтактный манометр.

Регулятор **KVIC**. Удобное ручное управление, шаг регулировки 0,5 вольт позволяет весьма точно вывести колонну на режим предзахлеба и удерживать её в этом состоянии длительное время. Но собрать его под силу только опытному электронщику.

Регулятор **PMB-к**.

Стабильно работающее устройство при довольно большом выборе конфигураций. Надежный, насколько известно, рекламаций практически нет. При подключении к Самовару по UART удобен и прост в управлении. Можно приобрести полностью готовый к работе бокс в комплекте с датчиками «Стоп Разгон» и «Авария». Датчики могут быть использованы при самостоятельной работе регулятора и представляют из себя биметаллический термостат KSD9700. При подключении к Самовару датчик «Стоп Разгон» не используется, т.к разъем задействуется для линии обмена данными - соответственно Самовар управляет режимом "Стоп Разгон". Датчик «Авария» можно использовать как самостоятельную опцию. Для использования в Самоваре датчики не подходят, но могут быть использованы как комплектующие при создании дополнительной [линии защиты](#).

Не уступает ему и регулятор **Stab-avr**. Возможность самостоятельной сборки и ремонта делает его весьма привлекательным. Особенность регулятора в том, что задание мощности отображается в ваттах (а не в вольтах, как у регуляторов, рассмотренных выше).

В шаг регулировки 1/200 от номинала ТЭНа. В силу особенностей установки мощности, фактическое значение может незначительно отличаться от заданного Самоваром (например, 1803 Вт вместо 1800). Что абсолютно не критично.

Все достоинства регуляторов **PM** (PM2M, PM2M-Pro) перечеркиваются одним большим недостатком. Это отсутствием связи по UART. И в силу этого невозможностью устанавливать напряжение на регуляторе «из вне». Исключительно ручное задание напряжения стабилизации. При работе с РМ Самовар может только включить/отключить регулятор и включить/отключить Разгон. Потому как других вариантов управления от сторонних устройств, данный регулятор не предусматривает.

Рекомендации.

Если у вас есть хоть небольшой опыт работы с паяльником имеет смысл собрать регулятор **Stab-avr**.

Если есть большие сомнения в собственных силах или нет времени, имеет смысл приобрести регулятор **PMB-к** в сборе.

Как подключить регулятор к Самовару

Подключение с помощью проводов.

Регуляторы Kvic и PMBк

В регуляторе KVIC для развязки по напряжению используется ADUM1201. Самовар работает с ним на 38400 бод, а при включённом **KVIC_USE_9600** — на 9600 бод.

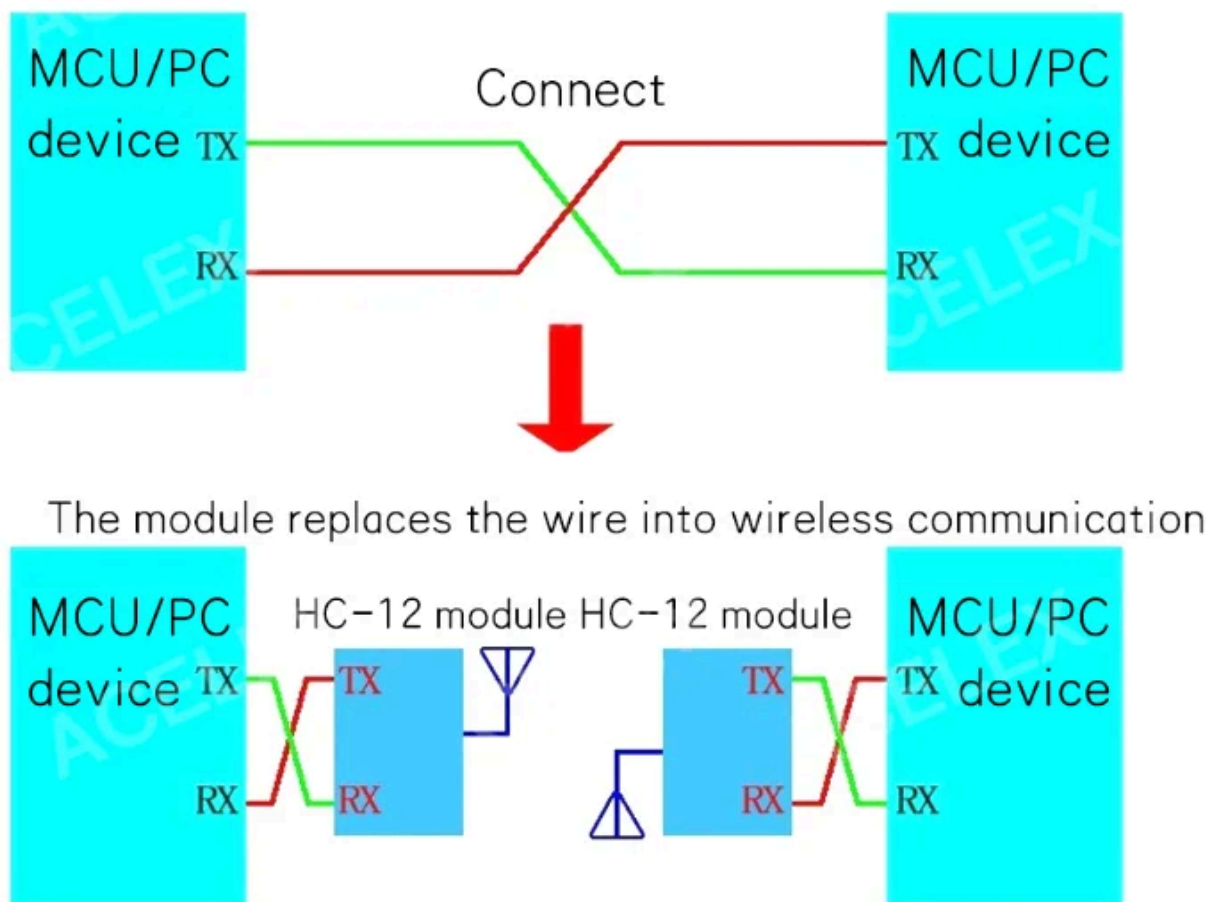
В регуляторе PMBк для развязки используются оптопары PC817. Скорость на порту 9600 бод.

Принцип соединения этих регуляторов с Самоваром одинаков. Для работы необходимо чтобы Rx одного устройства соединялся с Tx другого.

Со стороны Самовара используется UART2 (на печатной плате — разъём UART): RX D16 и TX D17 на ESP32 DevKit, RX GPIO25 и TX GPIO4 на LILYGO T-Relay 4, RX GPIO18 и TX GPIO17 на ESP32-S3. Регулятор SEM_AVR (Stab-avr) тоже подключается к этому порту, скорость 9600 бод.

Как подключить регулятор к Самовару без проводов

Самовар с регулятором, управляемым по UART (com-порт) можно соединить без провода с помощью двух RF module (радиочастотный модуль) HC-12. Сделать это очень просто, достаточно их соединить по схеме:



При этом адум не нужен, так как модули рассчитаны на работу с напряжением от 3.1 в до 5 в, т. е. их можно подключить и к ESP32 (Самовар) и к Arduino Nano.

Производитель заявляет, что дальность передачи информации этого модуля до 1 км (на открытой местности и с внешней антенной), для нашей задачи более чем достаточно. Модуль идет по умолчанию с настройкой скорости 9600bps, если вы используете регулятор STAB-AVR или RMVK с управлением по UART, просто подключите один модуль к Самовару, другой к регулятору в соответствии с приведенной схемой, и все должно заработать. Если же используется регулятор от Kvic, то необходимо настроить оба модуля. Сделать это не сложно, но потребуются их подключить к компьютеру, чтобы изменить скорость, по которой модули обмениваются друг с другом информацией.

Оба модуля HC-12 должны быть настроены на ту же скорость, что Самовар и регулятор. Для PMB-K и SEM_AVR используется 9600 бод. Для KVIC используйте 38400 бод либо 9600 бод, если в сборке включён **KVIC_USE_9600**.

У модуля много разных опций, но нас интересует одна: изменение скорости порта модуля. Для того, чтобы модуль перевести в режим настройки, нужно контакт SET замкнуть с GND. После этого подключить к нему питание и подключить к компьютеру через переходник UART-USB (можно для этого использовать Arduino Nano). Если выбрана скорость 38400 бод, отправьте в определившийся COM-порт команду **AT+b38400** и дождитесь ответа **OK+b38400**. При скорости 9600 бод менять заводскую настройку HC-12 не требуется.

Отключаем SET от GND, отключаем питание, подключаем модули к Самовару и регулятору, все должно работать.

Внимание!

Регулятор Kvic не имеет гальванической развязки с сетевым напряжением, даже антенна HC-12 будет находиться под напряжением сети! Поэтому модуль должен быть убран в корпус регулятора или иным образом надежно заизолирован.

ТЭН

Можно использовать любые нагревательные элементы которым может управлять используемый стабилизатор напряжения. Здесь рассмотрим наиболее стандартную модель ТЭНа.

ТЭН расшифровывается как "трубчатый электронагреватель".

Соответственно, сколько трубок, столько и нагревателей в сборке. Если взять общую мощность ТЭНа и разделить на количество трубок, получим мощность одной трубки. Все трубки по мощности одинаковы.



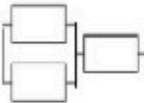
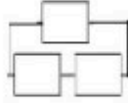
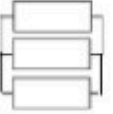
Как правило выбирают модель мощностью 3кВт. Соответственно каждая трубка будет иметь мощность 1кВт. Для 1,5" колонн это хороший выбор, поскольку для работы в предзахлебном режиме колонне высотой 1,5 метра потребуется 1,2-1,4 кВт энергии. Остается запас по мощности. Поэтому можно применить комбинированное подключение:

- Две трубки подключить через регулятор, это будет 2 кВт. При ректификации регулятор будет поддерживать напряжение на выходе в пределах 170 -180 вольт. Вполне достаточно, даже при "просадке" питающей сети.
- Оставшуюся трубку подключить через твердотельное реле, а управление этим реле будет через контакты реле №4. Это будет разгонный нагреватель.

Для колонн 2", если использовать подобную схему коммутации, ТЭНа 3кВт будет недостаточно. А вот 4,5 кВт будет самый раз.

Коммутируя соединения трубок можно использовать ТЭНы и гораздо большей мощности, при этом они будут работать в более щадящем режиме. Разумеется нужно соблюдать принцип разумной достаточности.

Ниже приведена таблица возможных соединений нагревателей в сборке.

	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

Пояснение к таблице. На примере ТЭНа 4,5 кВт

В левом столбце выбираем мощность - 4,5 kw. В верхней строке под номером, схема соединения нагревателей.

Например под номером 1 последовательное соединение всех трех нагревателей. На пересечении строки и столбца - мощность которую выдаст сборка если на нее подать 220 вольт = 0,5кВт или всего 500 ватт.

Соединение №6 позволяет, получить на 2-х нагревателях 3кВт и один нагреватель не задействуется - соответственно его можно использовать как разгонный.

Использование разгонного ТЭНа

Самовар может управлять дополнительным разгонным ТЭНом в режимах ["Ректификация"](#), ["Дистилляция"](#) и ["Пиво"](#)

Для использования функции нужно к нормально разомкнутым контактам реле №4 подключить либо контактор, либо твердотельное реле с током нагрузки достаточным для управления вторым нагревателем.

Работа разгонного ТЭНа в режиме "Ректификация"

Для использования функции дополнительного разгонного ТЭНа в скетче ничего менять не нужно. Эта функция включена по умолчанию.

Пороговые температуры, о которых идёт речь ниже, задаются в конфигураторе на вкладке «Температуры» («Температура перехода из разгона, °C», «Температура открытия охлаждения, °C», «Порог разгонного ТЭНа, °C»); при сборке в Arduino IDE — теми же директивами в [Samovar_ini.h](#).

Логика работы аналогична работе [РМ](#) в режиме "Разгон":

При команде "Включить нагрев", помимо включения Самоваром основного ТЭНа, реле №4 включит контактор или твердотельное реле, которые в свою очередь подадут напряжение на разгонный ТЭН. По достижении температуры пара значения указанного в настройках (по умолчанию 39°):

//Значение температуры датчика пара, при котором колонна перейдет из режима разгона в рабочий режим

```
#define CHANGE_POWER_MODE_STEAM_TEMP 39
```

Реле №4 разомкнет контакты и отключит разгонный нагреватель.

При регуляторе с управлением по UART (KVIC, PMB-K, SEM_AVR) есть ещё два условия: реле №4 включено только пока идёт статус «Разгон» и температура в кубе не выше

OPEN_VALVE_TANK_TEMP (по умолчанию 77°, «Температура открытия охлаждения»). Как только куб прогрелся выше этого значения, разгонный ТЭН отключается, даже если пар ещё не достиг 39°. Если датчик куба не подключён, разгонным ТЭНом Самовар не управляет. Без регулятора с UART реле №4 включается вместе с реле №1 при старте нагрева и выключается по завершении разгона.

Если команда "Включить нагрев" не подана, или температура датчика выше указанного в скетче значения - разгон не включится.

Работа в режиме «Дистилляция». Разгонный выход включается при старте нагрева. После начала кипения он выключается, если в настройках («Настройки» → вкладка «Пиво») снят флажок **«Использовать разгонный тэн при кипении»**; при установленном флажке остаётся включённым до штатной остановки нагрева.

Работа разгонного ТЭНа в режиме "Пиво"

Подключение ТЭНА аналогично указанному выше.

Но алгоритм работы - иной

За работу разгонного ТЭНа при затирке отвечает параметр «Порог разгонного ТЭНа, °C» (вкладка «Температуры» конфигуратора), в [Samovar_ini.h](#) — строка:

```
#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

Включено по умолчанию, 4 - значение в градусах С°. Работает только с регулятором, управляемым по UART. Без такого регулятора реле №4 включается вместе с основным нагревом всякий раз, когда температура ниже заданной, и выключается вместе с ним.

При включении Самовар дает команду регулятору подать на основной ТЭН полную мощность. Если температура в кубе ниже заданной в строке программы затирания на 4°, то Самовар включит реле № 4, которое в свою очередь даст команду включить разгонный ТЭН. Если разница меньше 4° - разгонный ТЭН не включится и нагрев будет происходить только за счет основного нагревателя.

Когда пройдена очередная температурная пауза, или оператор дает команду - "Следующая программа". Самовар сравнивает температуру в кубе и заданную в следующей строке программы. Если разница больше 4° - задействуется разгонный ТЭН.

Если команда "Включить нагрев" не подана или дельта меньше установленной - разгон не включается.

При кипячении сусли разгонный ТЭН включается, если установлен флажок «Использовать разгонный тэн при кипении», иначе выключается.

Samovar 7.00: в режиме «Сыроварение» реле №4 занято клапаном слива рассола. Подключайте к нему либо клапан для сыра, либо разгонный ТЭН для остальных режимов, но не оба устройства одновременно. Отдельной логики разгонного ТЭНа в режимах «БК», «НБК» и «Су-вид» нет: с регулятором по UART реле №4 в них не включается.

Контактор. УЗО. Аварийное отключение

Дополнительная защита при нештатных ситуациях.

Рекомендуется к реализации.

Предлагаемая защита работает независимо и отключает оборудование при критически нештатных ситуациях.

Основную опасность представляет собой прорыв спиртовых паров. Если по какой-то причине случится:

- Пробой симистора.
- Зависание или некорректная работа регулятора напряжения.
- Сбой автоматики
- Выход из строя БП 12 вольт.
- Отсутствие охлаждения.

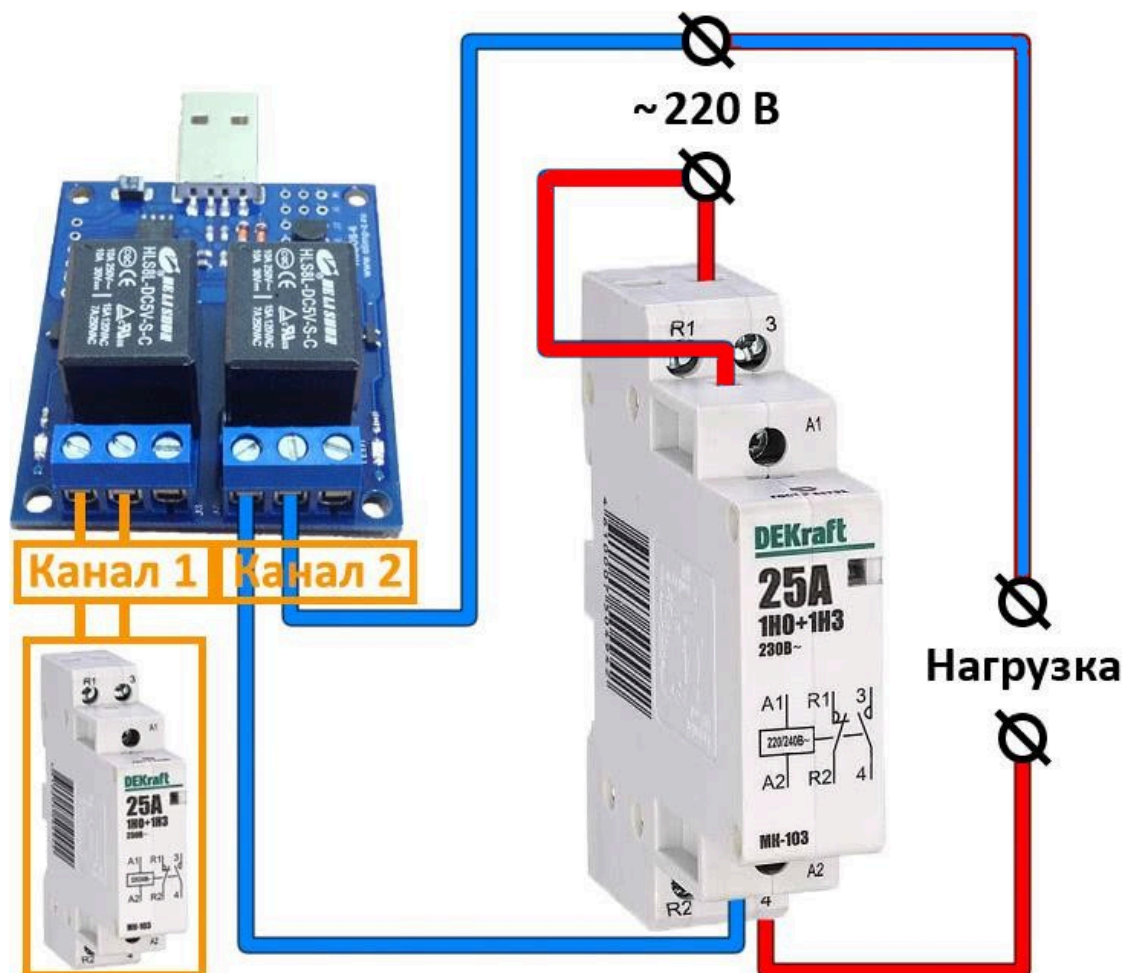
Необходимо отключить нагрев и прекратить выброс паров. Точка контроля

- ТСА.

Вариант №1

При условии, что совместно с Самоваром используется контактор.

Подключение контактора к [релейному модулю](#)



Потребуется биметаллический термостат KSD9700 НЗ - нормально замкнутый (eng. NC - normally closed) на температуру срабатывания 55-70°C. Желательно с изолированным корпусом: керамический или пластиковый.

Как реализовать:

Минусовое напряжение с блока питания 12 вольт подается на плату Самовара не на прямую, а через KSD9700, который закреплен на ТСА. Удобнее это организовать через двух контактный разъем на корпусе с Самоваром и провод нужной длины.

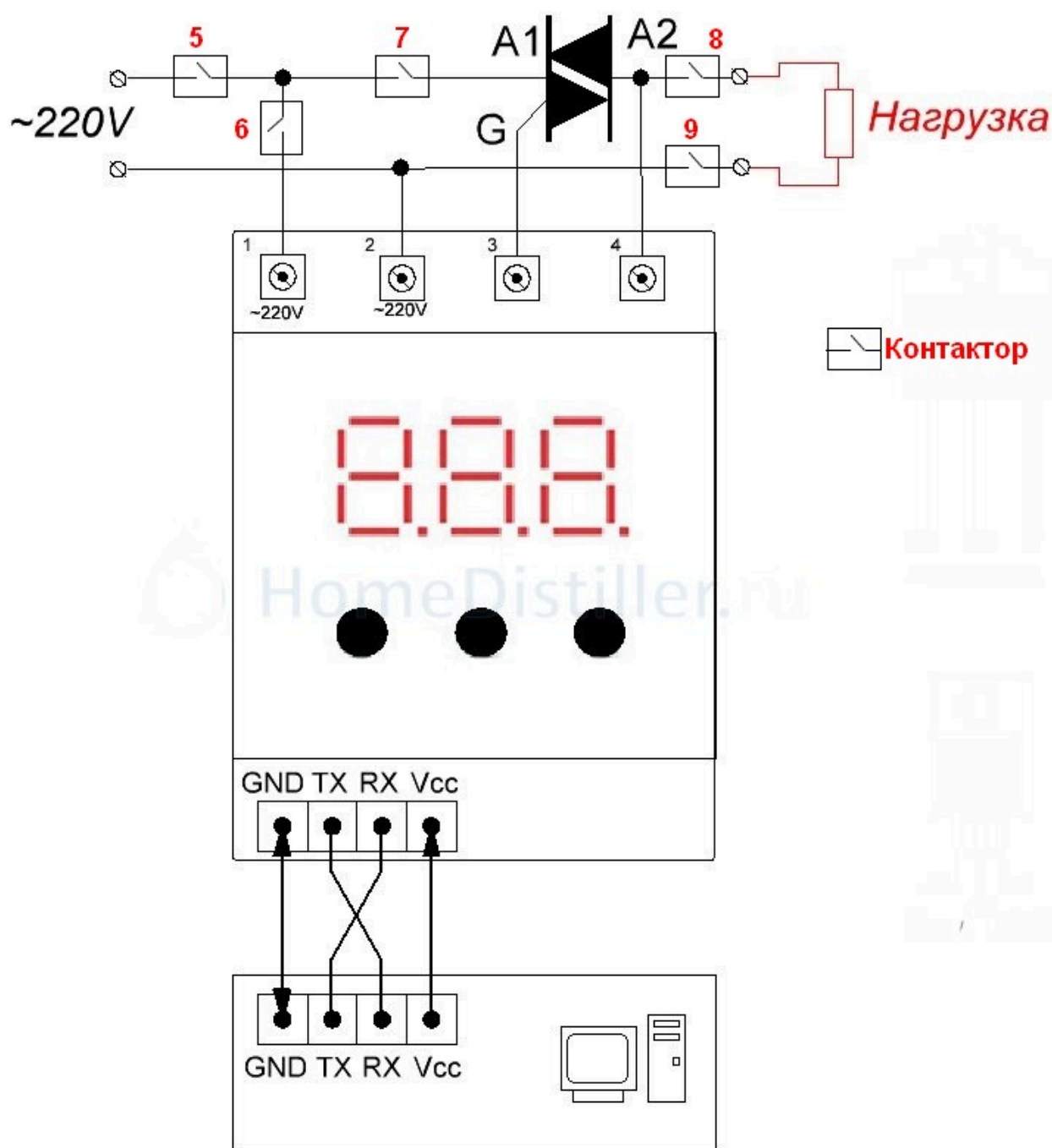
Как работает.

Если возникнет аварийная ситуация с которой Самовар не может справиться (обрыв связи с регулятором, пробой симистора и пр.) пар начнет прорыв через ТСА. Термостат нагреется и разорвет цепь. Самовар отключится, управляющее реле отключит контактор, а он в свою очередь, отключит нагрев. Выброс пара прекратится. Если используется клапан открытия воды - он закроется.

По прошествии некоторого времени ТСА остынет, термостат замкнет контакты, Самовар включится. Но все остальные устройства не включатся без вашей команды.

Разумеется нужно сначала разобраться почему произошло отключение. Ну и конечно, при подключении Самовара к колонне, если забыть подключить разъём, Самовар не включится - потому как нет напряжения, что равносильно выходу из строя блока питания.

Схема возможных мест установки контактора



Точка №5. Контактор включается и подает напряжение на регулятор и одновременно на симистор, который по команде с регулятора подаст напряжение в нагрузку.

Включение контактора происходит по команде с Самовара «Включить нагрев».

Соответственно нужно какое-то время чтобы регулятор стартовал, а затем получил команду с Самовара «Разгон» и подал напряжение на нагрузку.

Эта задержка задаётся параметром

SAMOVAR_USE_POWER_START_TIME (по умолчанию 2000 мс; в конфигураторе — «Задержка запуска регулятора, мс» на вкладке «Регулятор»), для SEM_AVR к ней добавляются ещё 3 с. Если регулятор за это время не ответил, Самовар отключает нагрев с сообщением «Таймаут команды регулятора».

Рекомендуется взвесить все ЗА и ПРОТИВ, прежде чем устанавливать контактор в данной точке.

Точка №6. Категорически не рекомендуется. Контактор в данной точке по сути, отключает регулятор и все. Сам регулятор потребляет ничтожно мало и можно поставить маломощный выключатель, если нужно отключать именно регулятор.

При этом подключении, если зависает регулятор, пробьётся симистор и прочее, ТЭН будет продолжать нагрев.

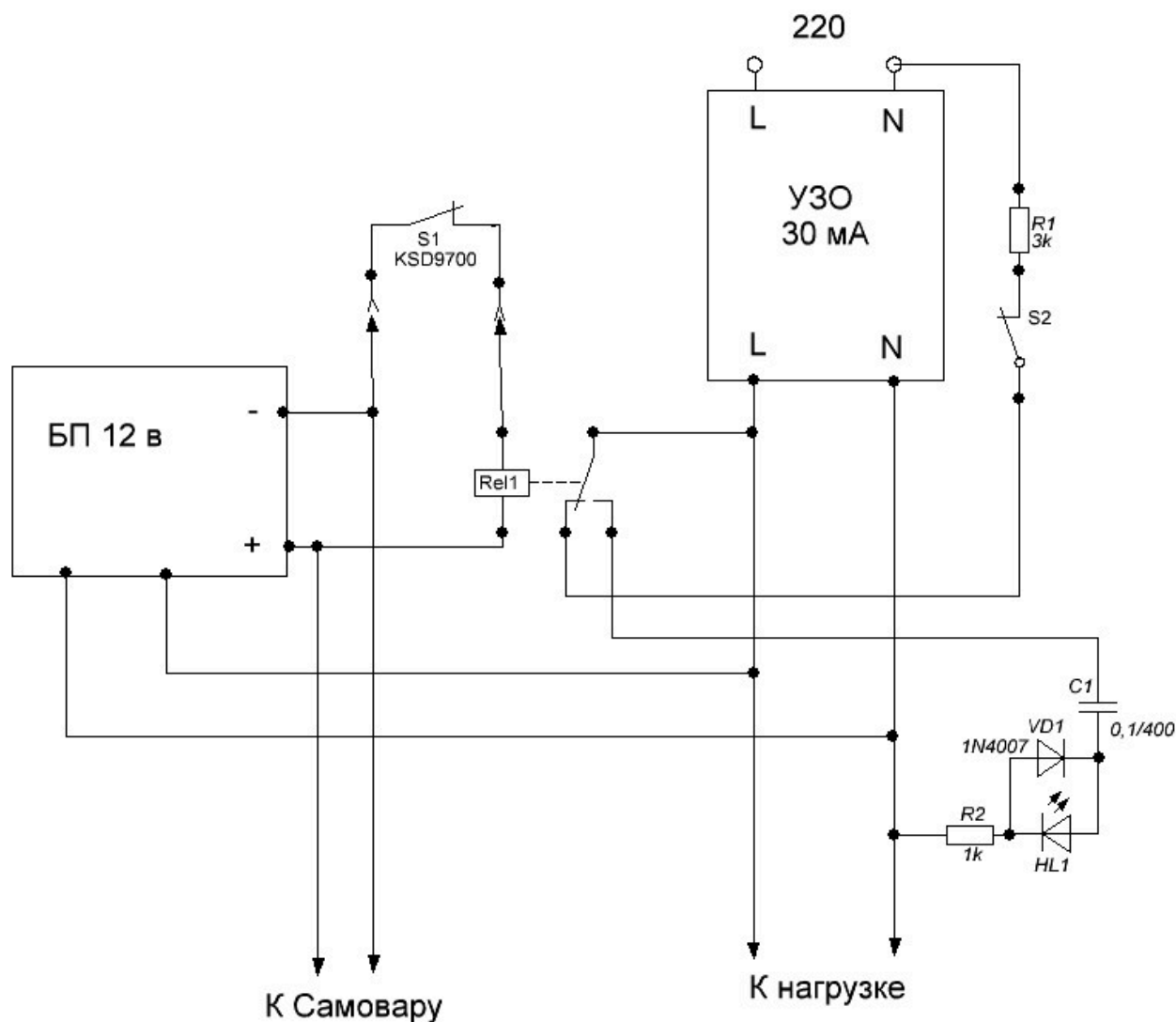
Точки №7 и №8 . При подаче общего питания включается регулятор. Стартует – столько времени, сколько ему нужно. Можно настраивать, задавать напряжения и пр. Но в нагрузку ничего не идет, потому как контактор разрывает силовую цепь на симистор. И вот только когда с Самовара дать команду «Включить нагрев», только тогда контактор замкнется и напряжение через симистор пойдет в нагрузку.

Точка №9 - рекомендуется для установки. При установке в этом месте контактора и при условии, что симистор закрыт - ТЭН полностью обесточен.

Вариант №2

Вместо контактора можно использовать УЗО, которое еще и добавит безопасности в плане утечки напряжения на корпус установки.

Датчик S1 (нормально замкнутый KSD9700) устанавливается на ТСА.



Как работает.

Для включения, в том числе и Самовара, нужно нажать и удерживать нормально замкнутую кнопку S2. Включение производится непосредственно самим УЗО. После подачи напряжения начнет работать БП 12 вольт и подаст напряжение на катушку реле Rel1. Контакты (НО) реле замкнутся и загорится светодиод HL1. Кнопку S1 можно отпустить. Сторож готов к работе. Если светодиод не горит и отпустить кнопку – сработает УЗО.

Рассмотрим возможные ситуации:

- Оператор забыл подключить датчик S1 – реле не включится – УЗО сработает
- Пробой симистора – пар через ТСА - датчик S1 разомкнет цепь – реле обесточится – сработает УЗО

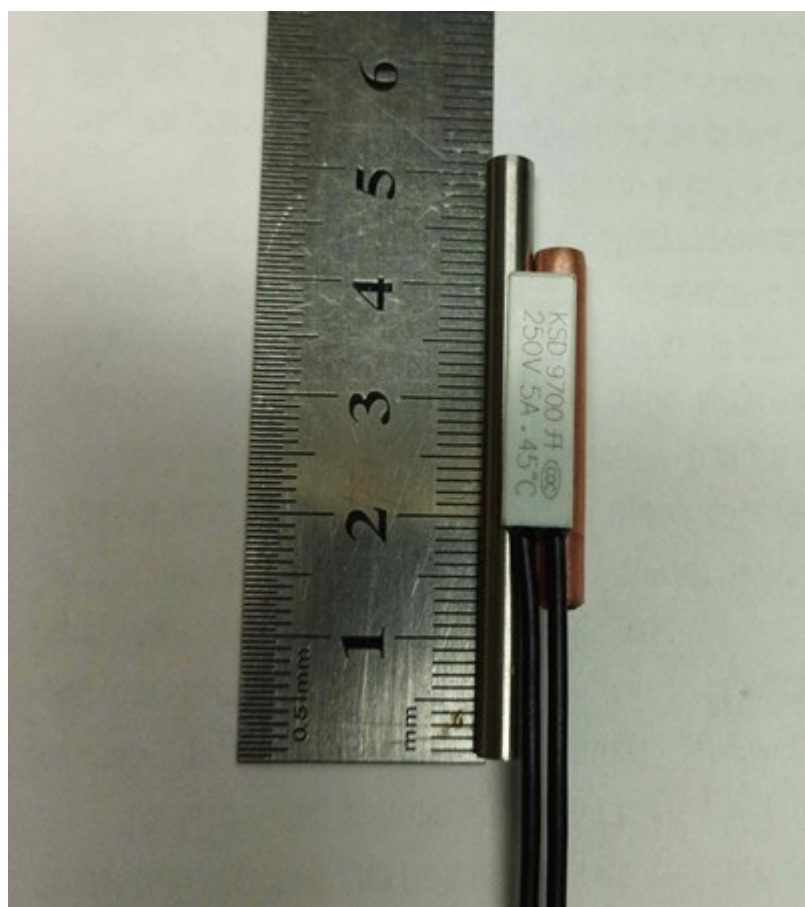
- Не достаточное охлаждение - пар через ТСА - датчик S1 разомкнет цепь – реле обесточится – сработает УЗО
- Вышел из строя БП12 вольт - реле обесточится – сработает УЗО
- Зависла автоматика и/или регулятор - пар через ТСА - датчик S1 разомкнет цепь – реле обесточится – сработает УЗО.
- Кратковременно отключилась электроэнергия – После подачи электроэнергии сработает УЗО.

Пример исполнения узла контроля

Узел линии контроля ТСА.

Материалы: трубка нержавеющая диаметром 6 мм., трубка медная диаметром 6,1 мм (при условии что датчик 18B20 без гильзы, если с гильзой то трубка не обязательна), датчик KSD9700, трубка термоусадочная, трубки силиконовые.

Трубку нержавеющую располагаем рядом с медной (или рядом с гильзой датчика температуры) . Длину трубки подбираем с таким расчетом чтобы нержавеющая была на пару сантиметров длиннее. Сверху укладываем датчик KSD9700 и обжимаем всю конструкцию термоусадкой. В медную трубку вставляем датчик 18B20, который прописан в Самоваре как ТСА. Один конец медной трубки, заливаем любым герметиком, на всякий случай, чтобы исключить попадание жидкости





При ректификации подсоединяем нержавеющей трубку в разрыв трубки от ТСА дефлегматора к емкости сброса конденсата паров . По опыту лучше в сантиметрах 30 от ТСА.

При перегоне браги и работе бражной колонны нержавеющей трубку присоединяем непосредственно к носику выхода продукта. Т.е продукт течет через трубку далее в приемную емкость. При этом, дополнительно, видим температуру продукта на выходе как температуру ТСА. Учитывая возможность в режиме реального времени регулировки температуры воды на выходе из холодильника, можно настроить насос на оптимальную производительность.



Можно не использовать медную трубку вообще, а прижать датчик без гильзы непосредственно к нержавеющей трубке, но придется озаботиться герметизацией датчика, к тому же площадь теплового контакта для KSD9700 уменьшится.

Для корректной работы, температуру отключения Самоваром нагрева по температуре датчика ТСА («Аварийная температура ТСА, °С» на вкладке «Температуры» конфигулятора; в [Samovar ini.h](#) — строка)

//Максимальное значение температуры в ТСА, при котором выключится питание

#define MAX_ACP_TEMP 75

нужно выставить на 5-10° ниже температуры срабатывания KSD9700. Значение по умолчанию 75° выше порога термостата 55-70°, поэтому его нужно уменьшить.

Тогда при прорыве пара сначала отработает Самовар и даст команду на отключение, а если она не будет исполнена - сработает контактор или УЗО.

Охлаждение

Samovar 7.00: температура открытия охлаждения и параметры насоса задаются на вкладках «Температуры» и «Насосы» конфигуратора.

В режиме ректификации и дистилляции Самовар поддерживает заданную температуру воды охлаждения.

Для организации охлаждения Самовар использует:

1. Выход реле №3: GPIO14 на ESP32 DevKit и GPIO41 на ESP32-S3, которое управляет клапаном подачи воды из магистрали. Или любым другим оборудованием, которое должно включаться по команде "Откройте подачу воды".
2. Выход управления насосом: GPIO4 на ESP32 DevKit и GPIO11 на ESP32-S3; на нём работает ШИМ-модулятор, обеспечивая автоматическое изменение скорости насоса подачи воды. Также данный выход может управлять клапаном увеличения потока воды, при условии что насос не используется.

Алгоритм работы (на примере Ректификация, при использовании оригинального скетча).

При достижении температуры куба **OPEN_VALVE_TANK_TEMP** (по умолчанию 77 °C) открывается клапан и запускается насос. Первые десять циклов регулирования насос получает стартовый уровень **PWM_START_VALUE** (по умолчанию 40%), затем регулятор поддерживает заданную температуру воды, не опускаясь ниже **PWM_LOW_VALUE**.

Настройки охлаждения делаются в двух местах:

1. В конфигураторе (вкладки «Температуры» и «Насосы») или в файле **Samovar_ini.h** — это одни и те же параметры, конфигуратор записывает их в этот файл. Строки:

*//Значение температуры датчика в кубе, при котором откроется клапан
подачи воды и включится насос подачи воды*

#define OPEN_VALVE_TANK_TEMP 77

При температуре куба 77° будет дана команда на открытие клапана подачи воды, одновременно будет подан управляющий сигнал ШИМ на включение насоса и послано сообщение оператору о необходимости открыть воду. Сообщение посылается в любом случае, даже если не используется ни насос, ни клапан.

#define DELTA_T_CLOSE_VALVE 20

//Разность между заданной Т охлаждения воды и Т воды, до которой будет работать насос после завершения процесса перегонки в режимах дистилляции и ректификации. Значение в градусах С°, можно менять. Отключать крайне не желательно. (подробнее см. [здесь](#))

//Нижнее значение (в процентах), для работы ШИМ-регулятора для насоса. Необходимо подобрать такое значение, которое обеспечивает поток воды
#define PWM_LOW_VALUE 10

Как известно двигателю для старта нужно напряжение больше, чем для поддержания работы. В данной строке выставляется минимально возможное значение в % при котором двигатель может поддерживать работу насоса подачи воды и при этом не останавливаться.

//Нижнее значение (в процентах), с которого начнет работать ШИМ-регулятор для насоса. Необходимо подобрать такое значение, которое гарантированно обеспечивает запуск мотора

#define PWM_START_VALUE 40

Значение в % от номинального напряжения, при котором двигатель насоса гарантированно стартует.

Т.е если ваш насос (при условии, что он уже установлен в системе и в нем есть жидкость) уверенно начинает стартовать при подаче от 4 вольт, а при меньшем не стартует, то при его номинальном напряжении 12 вольт, это составляет 30% от номинала. Значит значение 40 можно заменить на 30.

2. В настройках на вкладке Температура - уставка датчика воды. Эту температуру будет поддерживать насос в процессе.

После остановки нагрева охлаждение выключается, когда вода остыла до уставки минус **DELTA_T_CLOSE_VALVE** и прошло не менее трёх минут. Второй вариант — ранее нагретый куб остыл ниже 70 °С. Для режима с датчиком ТСА дополнительно проверяется его охлаждение.

В режиме **Бражной колонны** доступно автоматическое управление охлаждением после включения кнопки **«Автомат»** в web-интерфейсе. В режимах **Пиво** и **Сыроварение** насос и клапан включаются на этапах охлаждения и перекачки согласно программе (см. [Режим Сыроварение](#)). Клапан используется только в тех режимах и этапах, где прошивка запрашивает охлаждение.

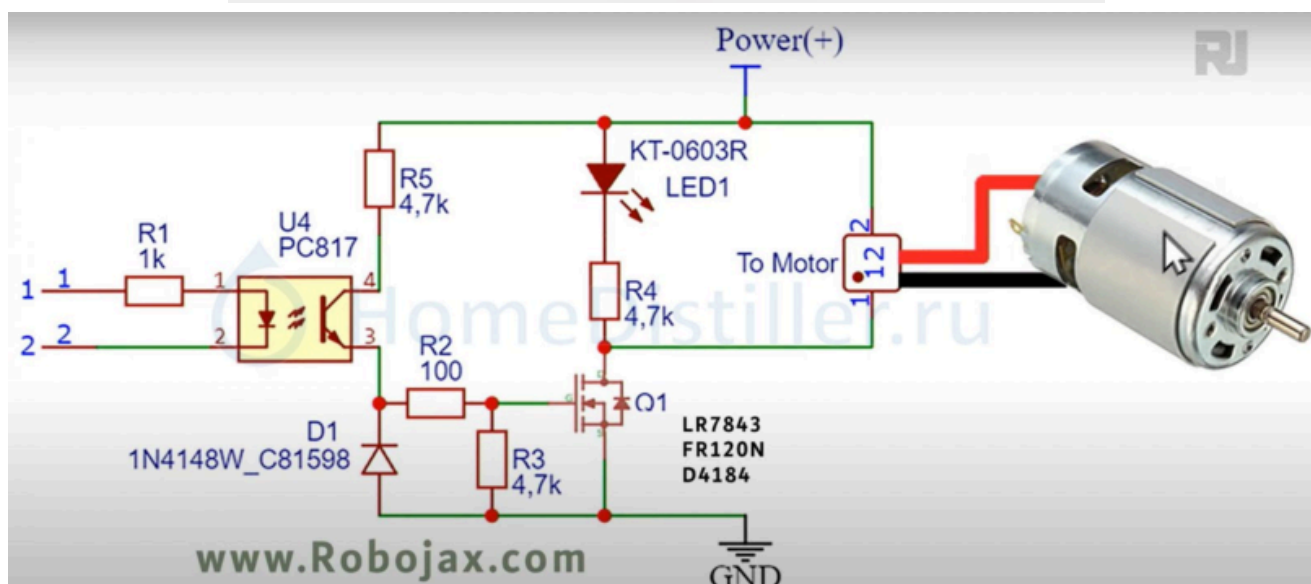
Реализация:

Для управления насосом (или клапаном потока) нужен посредник: твердотельное реле или силовая плата.

Внимание.

Некоторые твердотельные реле могут не срабатывать если напрямую подключены к ESP32 - не хватает напряжения и тока.

Рекомендуется использовать плату с полевым транзистором (мосфет) [LR7843](#)



Данный мосфет управляет двигателем "по минусу"
Хотя утверждается что мосфет почти не греется, лучше расположить плату, в корпусе, в зоне движения воздуха.

Далее: [мембранный насос](#), [центробежный насос](#) и [охлаждение без насоса](#)

Использование мембранного насоса

Насос не рекомендуется подключать напрямую к напорной магистрали. Крайне желательно установить промежуточную емкость с поплавковым клапаном.

Конструкция поплавкового клапана проста и надежна. Работает годами без проблем. Для гарантии от протечки можно предусмотреть сбросовую аварийную магистраль.

Мембранные насосы, в основном, являются самовсасывающими, но тем не менее желательно организовать подачу воды снизу емкости, через штуцер, чтобы насос был постоянно подпитан водой.

Рекомендованная модель насоса: питание 12 вольт, мощность до 60 ватт, производительность 2-3,5 л/мин.

Примечание

При ректификации на колонне 2" расход воды составляет примерно 40-50 л/час, при дистилляции, с использованием ТЭНа на 3кВт, расход выше.

В основном используют вот такие насосы.



И хотя его мощность избыточна, подобные модели хорошо себя зарекомендовали. Обратите внимание на винт и крышку слева. Это устройство автоматического отключения от питания при превышении давления. Может быть регулируемое и нет. На некоторых моделях может отсутствовать.

Если планируется использовать подобный насос, то блок питания Самовара на 12 вольт должен быть соответствующий. Например 12 вольт 100 ватт.

Насос шумный, имейте ввиду при размещении. *

Если использовать насос планируется только на ректификации, можно использовать насос поскромнее. Малые размеры позволяют закрепить его непосредственно на емкости или на колонне. Соответственно шум от такого насоса тоже меньше.



Рабочее напряжение: 12 В постоянного тока,
Рабочий ток: 0,5-0,7 А (мощность должна быть более 6 Вт)
Расход: 1,5-2 л/мин (или около того),
Самое большое всасывание: 78,7 дюйм (2 м)

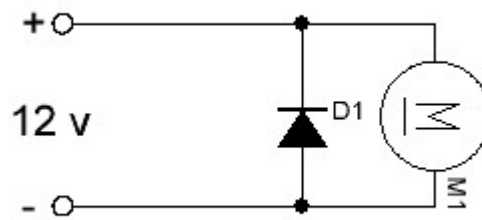
В принципе можно использовать любой мембранный насос на 12 вольт характеристики которого вас устраивают.

* Существует способ снизить шум и сделать работу насоса более комфортной. Можно изменить частоту ШИМ. По умолчанию она задана в файле **pumppwm.h**; чтобы не править этот файл, добавьте свою строку в **user_config_override.h**:

```
#define PUMP_PWM_FREQ 15
```

15 это частота ШИМ насоса охлаждения в герцах, заданная по умолчанию. Можно выставить до 450. В конфигураторе этого параметра нет.

Также нужно добавить шунтирование индуктивной нагрузки. Звучит страшно , но делается просто. Нужно как можно ближе к двигателю установить диод.



Диод любой, на ток не менее тока потребления двигателя. Хорошо себя зарекомендовали быстрые диоды Шотки, например **1N5844** или подобный.
Немного подробнее - [пост в теме](#)

Использование центробежного насоса

Насосы бывают - мембранные, плунжерные, шестеренчатые, центробежные и прочие. Используемые для наших целей можно разделить на 2 типа. Через которые, при выключенном состоянии, жидкость не проходит ни при каких условиях и соответственно через которые жидкость может пройти самотеком.

Ко второму типу относятся различные центробежные насосы с крыльчаткой.

Как определить? Нужно подуть во входной патрубок, если воздух проходит свободно — это он и есть.

По поводу применения данного типа насосов.

Какие факторы будут оказывать значимое влияние при использовании.

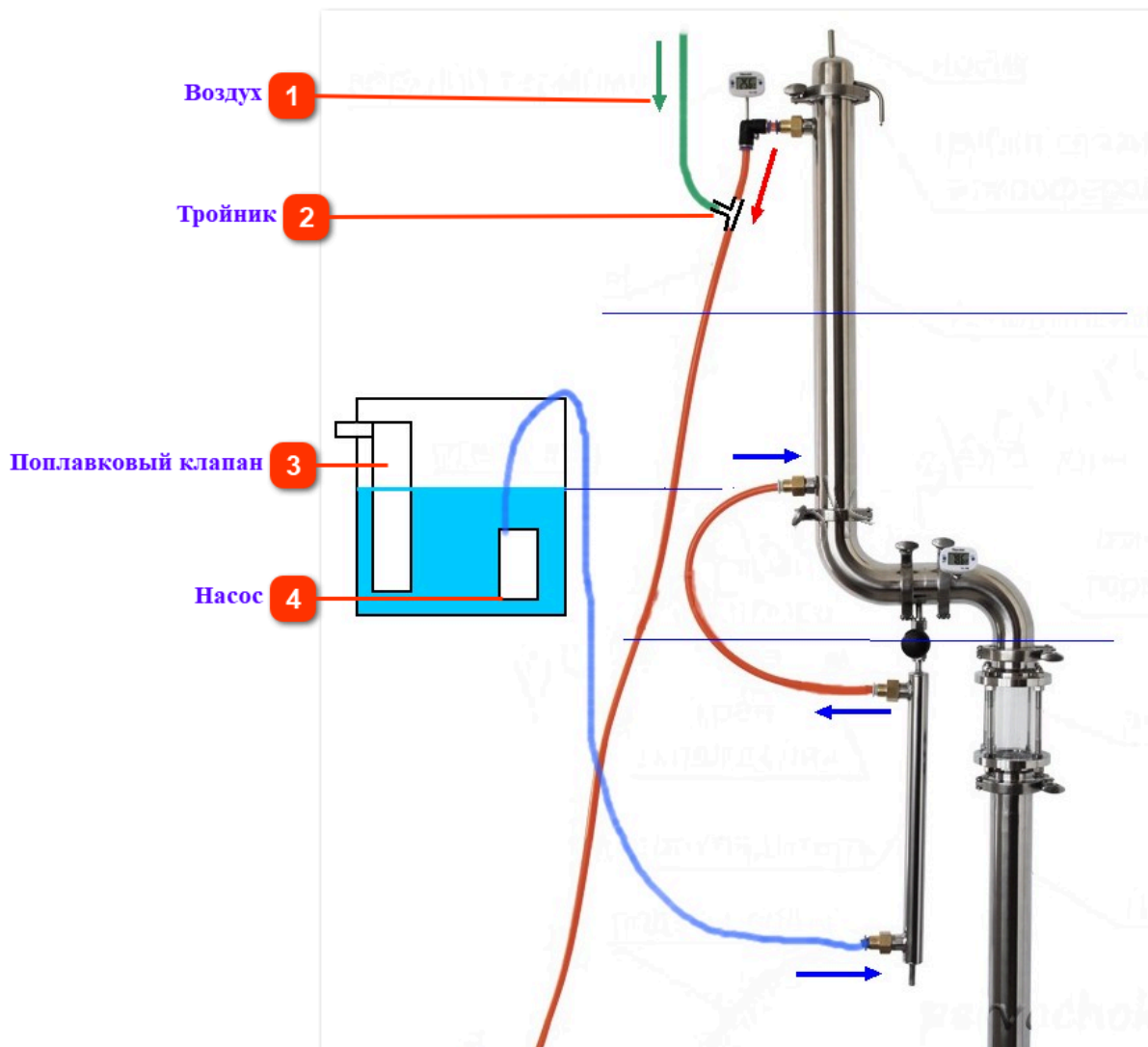
- Первый фактор - производительность. Если изготовитель указал что насос дает 600 л/час и поднимает воду на высоту 5 метров, это не значит что насос выкачает 600 литров на высоту 5 метров за час. Либо 600 литров на уровне забора воды, либо 5 метров в высоту и сколько выдавит (если выдавит вообще).
- Второй фактор - ШИМ. Он не только определяет объем который прокачает насос, но и высоту, на которую может быть поднят столб воды.
- Третий фактор – паразитный. Т.е не контролируемый «самотек», который может иметь место в системе.

Рассмотрим организацию охлаждения с таким типом насоса, на конкретном варианте, который используется нашим коллегой и трудится уже более года под управлением Самовара.

Дельта температуры составляет 1-2 градуса, при температуре поддержания на выходе 60°C.

Емкость (3-5 литров, лишь бы поместилась арматура и насос), с поплавковым клапаном и погружным аквариумным насосом поднята на высоту дефлегматора.

Желательно, чтобы уровень воды в емкости, который будет держать поплавок, находился на уровне нижнего патрубка дефлегматора. Не критично выше/ниже на 10-15 см.



1

Воздух

Через эту трубку будет поступать воздух, чтобы исключить неконтролируемый "самотек". По сути это ТСА для воды

2

Тройник

Обеспечивает "разрыв водяной струи"

3

Поплавковый клапан

Держит постоянный уровень воды в емкости.

4

Насос

При использовании **кожухотрубного дефлегматора**.

Подача насосом осуществляется на нижний вход доохладителя, с выхода доохладителя – на нижний вход дефлегматора. Выход горячей воды с верхнего выхода дефлегматора.

На выходе из дефлегматора нужно организовать установку датчика температуры и сделать «разрыв струи».

Для этого, на трубку сброса, установить тройник, трубку от него поднять вверх. Главное, чтобы тройник находился выше уровня жидкости емкости с насосом.

Что достигается данной организацией.

Насосу не нужно понимать высоко воду, он начинает обеспечивать именно расход воды, а не перемалывает крыльчаткой воду внутри себя, стремясь создать давление, чтобы поднять воду вверх.

ШИМ реально управляет расходом воды.

«Разрыв струи» полностью избавляет от неконтролируемого самотека.

Примечание:

«Разрыв струи» - это очень важно. На тестировании, без разрыва струи, с выключенным насосом, (т.е. охлаждение только за счет самотека через насос), при подаваемой мощности 1800 ватт, температура воды на выходе не поднималась выше 40°.

При использовании в качестве дефлегматора **холодильника Димрота**.

Подача насосом осуществляется на нижний вход доохладителя, с выхода доохладителя – на верхний вход дефлегматора. Выход горячей воды с нижнего выхода дефлегматора.

Т.е. поток воды идет навстречу пару.

На нижнем выходе из дефлегматора нужно организовать установку датчика температуры и желательно трубку сброса горячей воды, сначала поднять к верхнему входу дефлегматора, а потом вниз – на сброс. После того как трубка сброса пойдет вниз тоже сделать «разрыв струи». Для этого, на трубку сброса, установить тройник, трубку "ТСА для воды" от него поднять вертикально вверх. Проследить, чтобы тройник находился не ниже уровня жидкости емкости с насосом.

Проблемы имеющие место быть при использовании центробежных насосов практически не касаются владельцев мембранных насосов.

Единственная проблема, которая имела место – это очень холодная вода и капризный двигатель. При выставленных 15% мощности насос останавливался, а при 20% температура на выходе была слишком низкой. Кстати, это может иметь место, если использовать насос, рассчитанный на большую производительность.

Выход из положения – сразу на выходе из насоса сделать обратный сброс. На выходном патрубке насоса ставится тройник и делается

фиксированный сброс через жиклер (калиброванное отверстие) в ту же емкость. Таким образом можно поднять минимальное значение мощности, при этом реально на охлаждение пойдет меньше охлаждающей жидкости. Принцип контролируемого сброса можно применять на любых насосах.

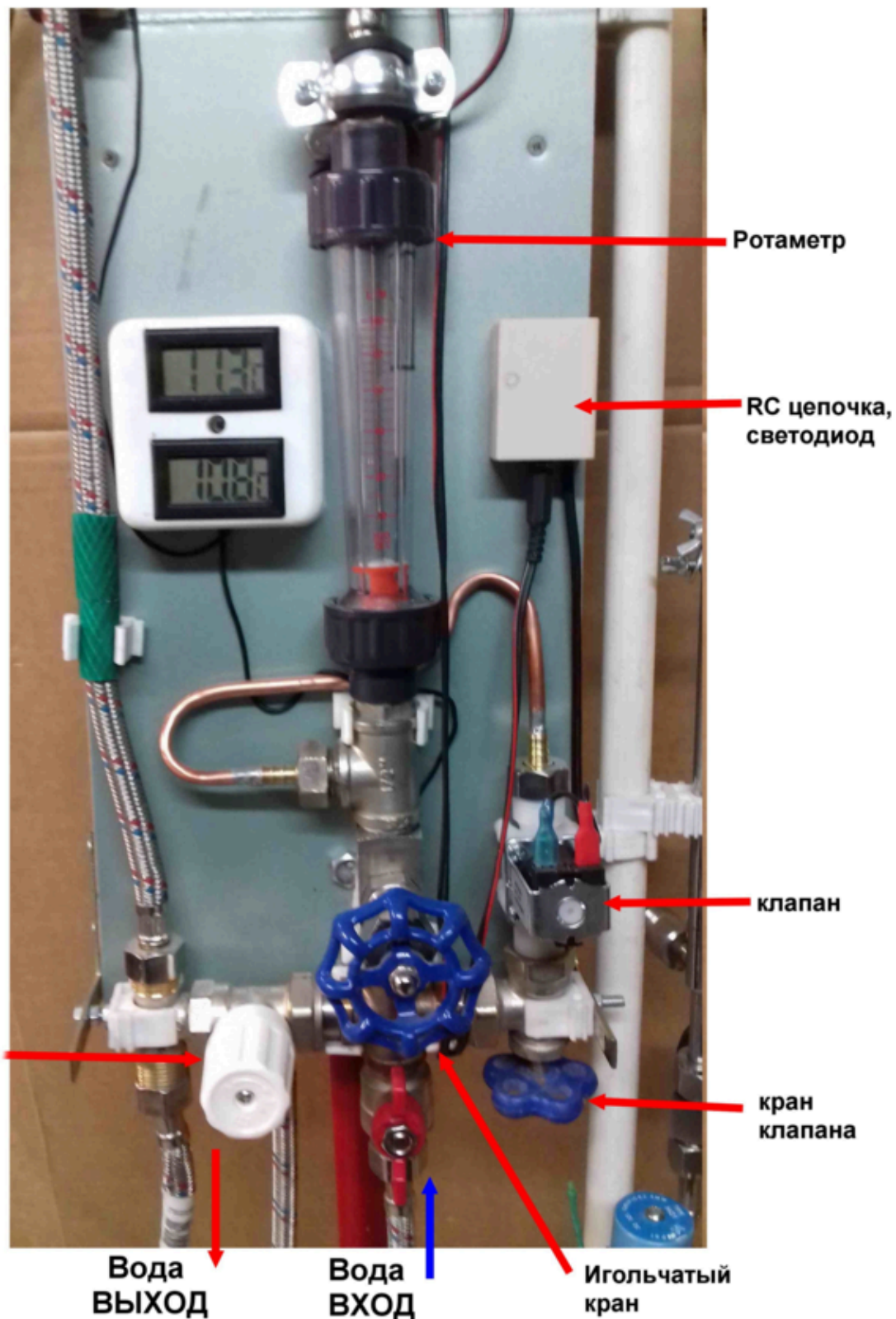
Без насоса

Samovar 7.00: насос и клапан воды включаются на вкладке «Насосы» конфигуратора; приведённые ниже директивы нужны только при ручной настройке.

Полуавтоматический режим

Суть этого метода в том что это скорее сторож, который работает и весьма успешно.

Вода подается из магистрали (или напорной емкости, размещенной выше колонны).
Вода приходит на игольчатый кран и далее для настройки и контроля устанавливается ротаметр.
Вокруг игольчатого крана делается обходная магистраль с еще одним краном и клапаном.



Настраивается так:

Подается вода. Игольчатый кран закрывается полностью. На клапан принудительно подается напряжение, клапан открывается и краном клапана выставляется расход, например 10 литров в час. Напряжение снимается- клапан закрывается.

Клапан обходной магистрали подключается вместо насоса к Самовару — к выходу управления насосом (GPIO4 на ESP32 DevKit, GPIO11 на ESP32-S3) через транзисторный ключ или реле, как описано на странице [Охлаждение](#).

В конфигураторе на вкладке «Насосы» снимите флажок **«Управлять насосом воды или мешалкой»** и в поле **«Управление клапаном воды»** укажите **LOW** (или HIGH, если ваш ключ включается высоким уровнем). Конфигуратор запишет это в файл `Samovar_ini.h`, где те же строки можно править вручную:

```
//#define USE_WATER_PUMP //использовать управление ШИМ для  
насоса для воды  
нужно закомментировать
```

а строку:

```
#define USE_WATER_VALVE LOW //использовать управление  
клапаном для регулировки напора воды  
нужно раскомментировать. Пока включён USE_WATER_PUMP, директива USE_WATER_VALVE не  
действует — в прошивке они взаимоисключающие.
```

Затем запускается колонна и игольчатым краном выставляется расход воды, чтобы на выходе из дефлегматора получить температуру, допустим 50°. Выводим колонну на рабочий режим, слегка регулируя расход игольчатым краном. Все - колонна охлаждается.

Если температура воды выйдет за уставку датчика воды, то откроется клапан на обходной магистрали и дополнительные 10 литров в час пойдут на охлаждение, как только температура снизится - клапан закроется.

Клапан открывается, когда температура воды поднимется выше уставки на 1 °С, и закрывается, когда опустится ниже уставки на 1 °С. Так работают режимы ректификации и дистилляции. В режиме Пиво этот выход не используется — открывается только клапан реле №3; в режиме Сыроварение обходной клапан открыт, пока температура продукта выше заданной более чем на 1 °С.

В таком режиме система может работать довольно-таки продолжительное время. Но оставлять без присмотра на долго не стоит и нужно периодически регулировать поток игольчатым краном.

Датчик уровня флегмы

Samovar 7.00: включение и полярность датчика задаются на вкладке «Датчики» конфигуратора и сохраняются в user_config_override.h.

Бесконтактный датчик уровня жидкости ХКС-Y25-V выполнен на ёмкостном чувствительном элементе, который реагирует на изменение диэлектрической проницаемости. Это свойство позволяет сигнализатору уровня обнаруживать жидкость через различные не металлические препятствия: пластик, стекло, керамика и т.д. Датчик жидкости даже зафиксировывает прикосновение вашей руки. Максимальная толщина стенки препятствия и соответственно дистанция чувствительности сенсора составляет 12 мм. С металлическими преградами датчик работать не будет.

Подробнее можно прочесть на ресурсе: [Бесконтактный датчик уровня жидкости ХКС-Y25-V](#)

Фактически Самовар может работать с любым датчиком (лазерный, индукционный, контактный и пр.) который может работать от 3.3 вольта (или 5 вольт, при условии использования преобразователя логических уровней.) и выдавать, при срабатывании, логическую единицу или логический ноль.

Выбор какой датчик использовать можно сделать в файле Samovar_ini.h

Использовать датчик уровня флегмы, который при срабатывании имеет на выходе логическую единицу.

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

Использовать датчик уровня флегмы, который при срабатывании имеет на выходе логический ноль

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

(Далее все примеры на основании использования датчика ХКС-Y25-V P-N-P)

Выбор режима выходного сигнала.

Некоторые модели датчиков имеют 4 вывода и имеют встроенную схему инвертирования выходного сигнала, которая управляется состоянием контакта MODE.

Штатный режим

Контакт MODE подключен к плюсу питания или не подключен вовсе: сигнал OUT не инвертируется.

- Если уровень жидкости в зоне видимости датчика: светодиод горит, на выходе датчика высокий уровень.
- Если уровень жидкости вне зоны видимости датчика: светодиод не горит, на выходе датчика низкий уровень.

Инверсный режим

Контакт MODE подключен к земле: сигнал OUT инвертируется.

- Если уровень воды в зоне видимости датчика: светодиод горит, на выходе датчика низкий уровень.
- Если уровень воды вне зоны видимости датчика: светодиод не горит, на выходе датчика высокий уровень.

Куда ставится, как крепить,

Датчик ставится на диоптр, так как через металл он не работает.

Этап 1 - временно-постоянный.

Понадобится:

- пластиковая карта
- два канцелярских зажима
- датчик
- маркер
- ступенчатое сверло
- нож



Берем карту, зажимом крепим к стойкам диопра (фото не очень удачное), отмечаем где находится вторая стойка.

Обрезаем и ступенчатым сверлом сверлим отверстие, немного меньше требуемого. Ножом подрезаем чтобы датчик вставлялся плотно.



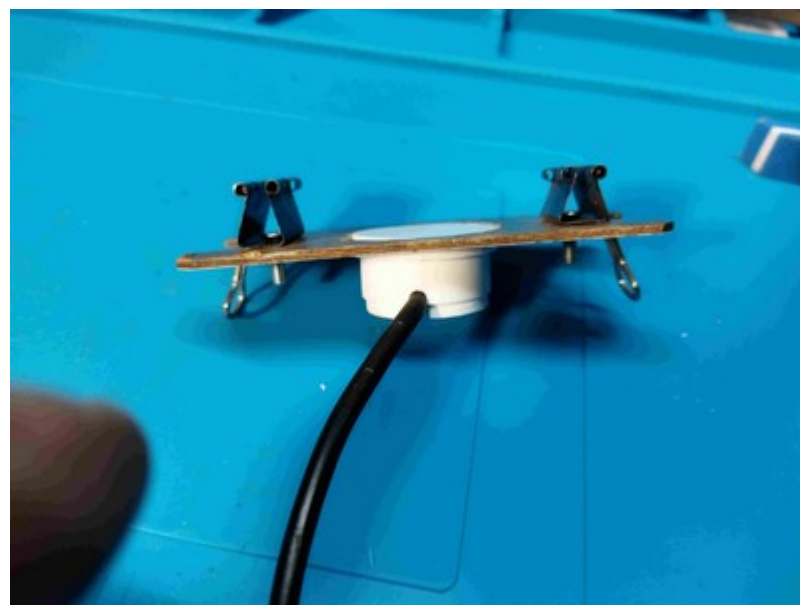
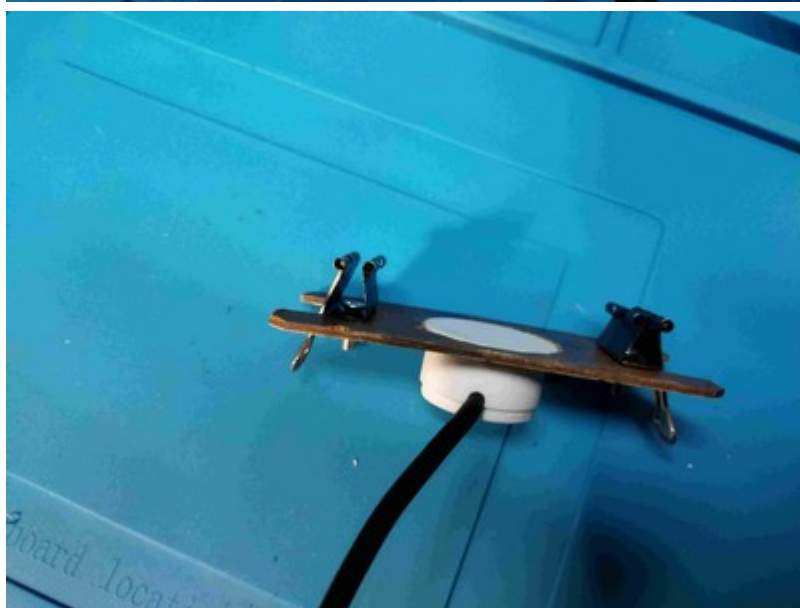
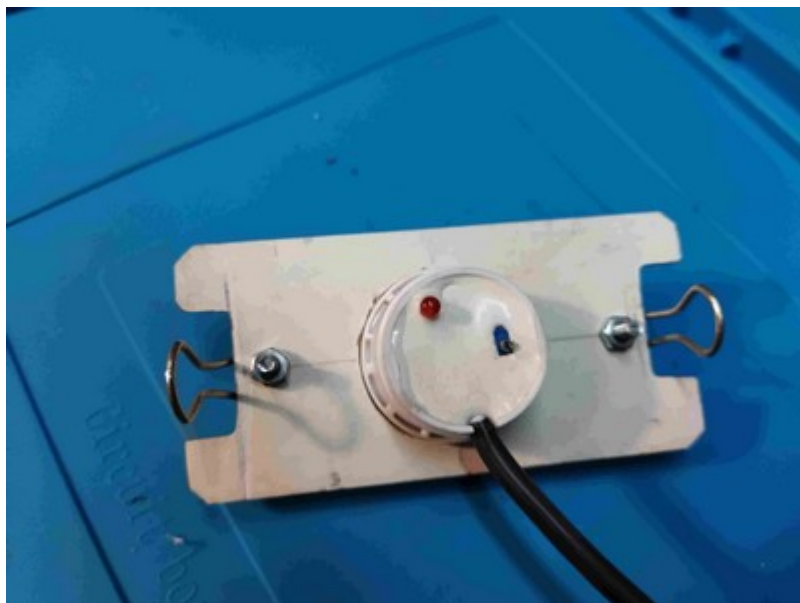
Рисуем + (плюс) и - (минус), это регулировка чувствительности, снимаем с датчика крышку, вставляем в отверстие и с помощью двух зажимов крепим на диоптр.



Готово! Крышку не ставим - придется крутить чтобы поймать срабатывание.

Постоянное.

Берется твердый пластик, также делается отверстие, но теперь зажимы сверлятся и крепятся на винты М3 "потай".



Снимаем одну пружину, также крепим на диоптр.



Что это дает. Первое - это жесткость конструкции, Второе - это возможность закрепить как можно ниже и даже под наклоном.

Датчик емкостной. Регулировать отверткой с длинной изолированной ручкой, лучше если вся отвертка пластиковая или ее основная часть.

Подключается датчик к разъему печатной платы LEVEL. Проверка проста - подключили, поднесли руку, светодиод сработал.

Срабатыванием датчика уровня флегмы Самовар считается период равный 3 секундам, в течение которого постоянно горит светодиод датчика. Если светодиод мигает с частотой менее 3-х секунд это не считается срабатыванием.

Если Самовар определил что датчик сработал, будет дана команда регулятору понизить напряжение: на 1 В для KVIC, на 2 В для PMBK, на 3 % от текущей мощности для SEM_AVR. Следующее снижение возможно не раньше чем через 40 секунд — процесс инерционный. Во время программы «Предзахлёб» срабатывание датчика — штатное событие, сообщение о захлёбе не отправляется.

После разгона, когда колонна выйдет на режим стабилизации и диоптр нагреется, нужно выставить порог срабатывания.

Для начала уменьшаем чувствительность датчика, выводим колонну на режим, начинаем отбирать головы. Отверткой вращаем винт

подстроечника до срабатывания датчика и немного возвращаем назад. Можно при отборе голов, поднятием мощности, захлебнуть колонну, чтобы убедиться в корректности срабатывания датчика и сделать более тонкую настройку. На качество конечного продукта этот эпизод не повлияет.

Чем тоньше настройка, тем раньше датчик отреагирует на появление флегмы в диоптре, но при этом возможны ложные срабатывания. Если датчик снимался или был длительный простой, корректировку нужно повторять.

Возможные проблемы.

- Датчик не подключен, а Самовар сообщает, что датчик сработал.

При настройке скетча, можно в конфигураторе на вкладке «Датчики» снять флажок «Использовать датчик уровня флегмы» или в файле *Samovar_ini.h* закомментировать строку

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

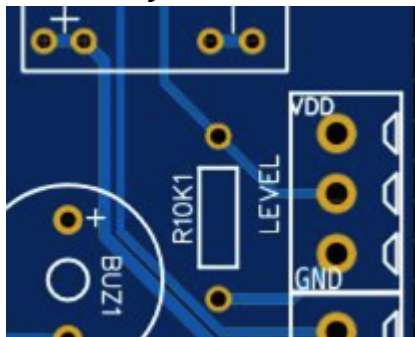
Или другой способ:

В файле *Samovar_ini.h* закомментировать строку

```
//#define IGNORE_HEAD_LEVEL_SENSOR_SETTING //Если датчик флегмы  
используется, то игнорировать команду на отключение датчика в  
WEB-интерфейсе. Тогда , в интерфейсе, на вкладке Настройки появится  
галочка, которой можно включить или отключить использование датчика  
флегмы.
```

- Датчик подключен, не срабатывает, а Самовар сообщает, что датчик сработал.

При использовании датчика, резистор R1-10k (находится рядом с разъемом LEVEL) должен быть обязательно установлен.



Если датчик при срабатывании выдаёт логическую единицу, вход уровня должен быть подтянут к GND: GPIO27 на ESP32 DevKit или GPIO19 на

ESP32-S3. На штатной печатной плате DevKit эта подтяжка уже предусмотрена.

Если датчик при срабатывании выдаёт логический ноль, вход уровня должен быть подтянут к +3,3 В: GPIO27 на ESP32 DevKit или GPIO19 на ESP32-S3.

Драйвер. Шаговый двигатель.

Samovar 7.00: направление и плавный разгон двигателя задаются на вкладке «Шаговый двигатель» конфигуратора.

В проекте используются драйвера шагового двигателя (далее ШД) A4988 или DRV8825.

Какой драйвер выбрать? Прочтите эту статью, чтобы иметь представление о предмете, в конце приведены основные различия этих драйверов.

Драйвер желательно устанавливать также как и ESP32 - на гребенку. Перед настройкой драйвера ESP32 должна быть установлена на плату и в нее должен быть загружен скетч Самовара.

Питание и ШД не подключаем.

Берем драйвер, ориентируясь по обозначениям выводов на плате драйвера и печатной плате, вставляем в гребенку, .

Проверяем прозвонкой мультиметром. На контактах GND должна быть масса, на VMOT +12 вольт от разъема питания VIN, на VDD +3,3 от первого контакта ESP32.

Подаем питание 12 вольт на разъем VIN. Можно еще раз проверить факт наличия напряжений.

На печатной плате, для драйвера предусмотрены переключатели:

- H1 - соединяете все три контакта
- SW1 - если есть переключатели, то все в положение ВЫКЛ (OFF) . Если нет переключателей - перемычки пока не ставите.

Теперь нужно настроить драйвер.

Двигатель не подключаем.

Драйвер штука нежная. Есть даже рекомендации настраивать с отключением питания.

Т.е. чуть-чуть крутнули подстроечный резистор, подали питание, замеряли напряжение, сняли питание - опять крутнули . И так все время. Пока не установите нужное.

Есть и другой, более быстрый способ.

Отвертка с изолированной ручкой. Один щуп мультиметра подключаем в любом месте на печатной плате к массе (GND), вторым либо аккуратно прикасаемся к подстроечнику и крутим отверткой ползунок, либо вешаем

на щуп "крокодил", цепляем его на жало отвертки и крутим отверткой через крокодил.

Методику настройки смотрим по ссылкам.

[Как настроить драйвер шагового двигателя DRV8825](#)

[Как настроить драйвер шагового двигателя A4988](#)

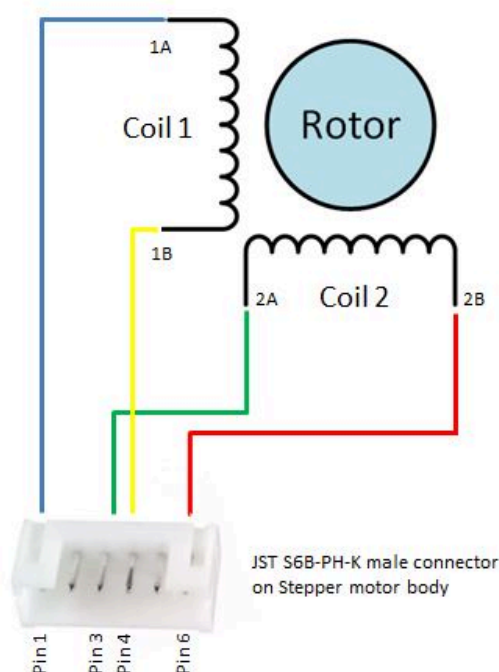
Подключение шагового двигателя.

В проекте используется биполярный ШД NEMA17. Сначала проверяем соответствие ШД проекту.

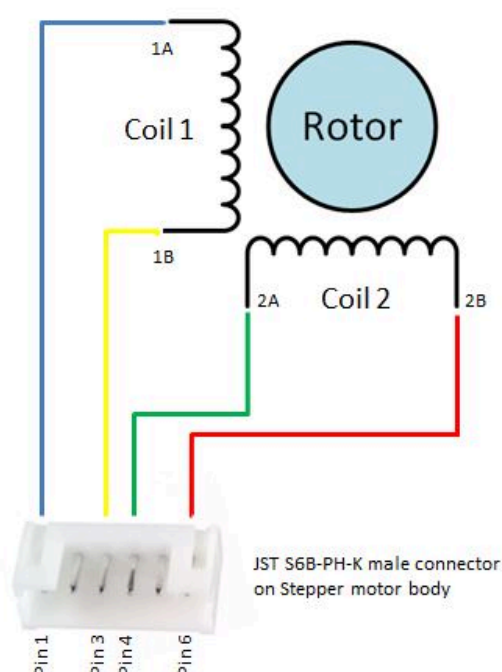
Биполярные ШД имеют две независимые обмотки. Двигатели разных производителей могут иметь разное расположение выводов.

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

Crossover



Straight



Подключаем к ШД разъем с проводами, который был в комплекте, берем любой провод и прозвонкой мультиметром ищем ему пару. Это одна обмотка. Помечаем оба провода. Проверяем вторую пару. Прозваниваемся - хорошо.

Теперь берем любой провод с одной пары и любой провод со второй. Прозваниваем. Они не должны звониться. Если есть прозвон - это не

биполярный двигатель.

Теперь подключаем ШД к разъему на печатной плате. По рисунку выше 1А-1В это одна обмотка, 2А-2В это другая. "А" это начало обмотки "В" это конец. Можно, конечно, подключить четко по обозначениям, но это не обязательно.

Главное подключить одну обмотку к контактам 1А-1В драйвера, а вторую к 2А-2В. Не важно какую куда, и не важно начало это обмотки или конец. Подключили. Подаем питание. Двигатель не крутится. Открываем web-интерфейс и даем команду на отбор.

Вал двигателя должен начать вращаться.

Двигатель гудит и не крутится; Двигатель крутится рывками

- Проверьте правильность и наличие подключения двигателя к драйверу. Именно к драйверу, а не только к печатной плате.
- Проверьте что на контактах драйвера RST,SLP присутствует +3,3 вольта.
- В web-интерфейсе на странице Настройки, вкладка «Насос», нажмите «Калибровка насоса». Если в поле «Скорость шагового двигателя» установлено значение 8000 (или подобное), измените на 800

([пост с форума](#): ...была подобная ситуация при настройке Самовара. Отличие драйвер ТМС2208. Проблема решилась подбором деления шага, насос стабильно заработал только на 1/8, на других режимах, либо гудел, либо работал рывками с дикими пропусками (хотя на другом устройстве этот драйвер и насос работали на всех делителях отличие было только в управляющих сигналах 5в а не 3,3в). Возможно это как-то связано с частотой следования импульсов STEP, тогда попробовать подобрать в окне «Калибровка шагового двигателя» параметр «скорость шагового двигателя» (допустим установить 4000 а не 8000). А после калибровать на 100ml.

Не забываем выбор режимов работы ШД осуществлять при полном отключении питания печатной платы Самовара.)

Если вал вращается не в ту сторону которую нужно, на подключении к плате, у любой из обмоток меняем провода местами.

Примечание.

В файле Samovar_ini.h присутствует строка, которая позволят изменить вращение программно:

```
//#define STEPPER_REVERSE //изменить направление вращения шагового
```

двигателя

В конфигураторе это флажок «Обратное направление шагового двигателя» на вкладке «Шаговый двигатель», там же — «Плавный разгон шагового двигателя» (USE_STEPPER_ACCELERATION). Но удобнее один раз перебросить провода, чем при каждом обновлении следить за параметром.

Погоняете двигатель на разных скоростях через программу отбора. Выставьте отбор 70-150 мл/час. Возьмитесь одной рукой за корпус двигателя а второй попытайтесь остановить вал, если вал легко остановился, нужно вернуться к настройкам драйвера и немного поднять напряжение. Буквально на 0,02-0,04 вольта.

Снова проверьте насколько легко можно остановить вал.

Приклейте на чип драйвера радиатор идущий в комплекте.

Теперь можно ставить головку насоса. Предварительно смажьте ролики и трубку нейтральной силиконовой смазкой. Смазку нужно будет проводить периодически (раз в 5-7 перегонов), это увеличит срок службы роликов, трубки, да и двигателю будет легче.

Погоняйте с насосом, посмотрите нет ли остановок, не слышно ли перегрузки двигателя, насколько греется двигатель и радиатор чипа драйвера...

Теперь можно перейти к тонкой настройке, если в этом есть необходимость.

Немного теории.

В отличие от традиционных двигателей постоянного и переменного тока, шаговые двигатели движутся дискретными шагами или приращениями, отсюда и название «шаговый».

Каждый шаг шагового двигателя соответствует точному угловому вращению. Обычно углы шага шаговых двигателей составляют 1,8° или 0,9°, что соответствует полному обороту, разделенному на 200 или 400 шагов соответственно.

Стандартные NEMA17 имеют угол поворота 1,8° за 1 шаг (200 шагов на полный оборот)

Двигатель может работать в нескольких режимах - полношаговый и микрошаговый.

Полношаговый режим это когда двигатель за 1 шаг делает поворот вала на 1,8°

При полношаговом режиме двигатель имеет самый высокий крутящий момент среди всех режимов.

Микрошаг- это деление стандартного шага на несколько микрошагов..
Разделить стандартный шаг можно на 2, на 4; 8;16 и 32 части
Настроить деление шага можно переключателями SW1

MS1, MS2, MS3 – контакты выбора разрешения микрошагов. С их помощью мы можем выбрать один из пяти режимов:

Для драйвера **A4988**

MS1	MS2	MS2	Шаги
откл	откл	откл	Полный
вкл	откл	откл	1/2
откл	вкл	откл	1/4
вкл	вкл	откл	1/8
вкл	вкл	вкл	1/16

Для драйвера **DRV8825**

MS1	MS2	MS2	Шаги
откл	откл	откл	Полный
вкл	откл	откл	1/2
откл	вкл	откл	1/4
вкл	вкл	откл	1/8
откл	откл	вкл	1/16
вкл	вкл	вкл	1/32

Пояснение.

Допустим выбран режим 1/4. При этом за 1 микрошаг вал двигателя повернется на $1,8^\circ/4 = 0,45^\circ$ что будет соответствовать $200 \cdot 4 = 800$ (или если считать иначе $360^\circ/0,45^\circ = 800$) микрошагам на 1 полный оборот вала.

При микрошаговом режиме двигатель имеет пониженный крутящий момент, но повышенную плавность и точность (что не особо важно для проекта)

Шаговые двигатели в полношаговом режиме создают более высокие уровни вибрации и шума.

Микрошаг существенно снижает вибрацию и шум, что приводит к более плавной и тихой работе двигателя.

В общем посмотрите и послушайте как работает ваш ШД. Можно попробовать различные режимы и сравнить.

Внимание.

Выбор режимов работы ШД осуществляется при полном отключении питания печатной платы Самовара.

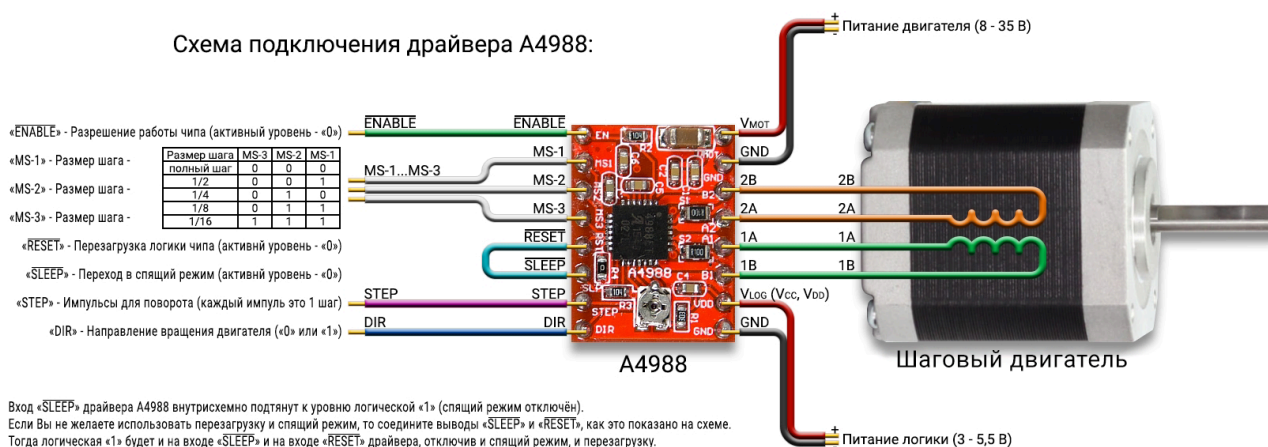
После того как определитесь с режимом работы ШД, поставите dir-переключатели в нужное положение (или окончательно распаяете перемычки) нужно будет сделать калибровку насоса. Калибровку нужно делать каждый раз, если вы решите сменить режим работы двигателя. На точность отбора также влияет регулярность смазки насоса. Чем реже смазываете - тем меньше фактическая скорость отбора.

Различия между драйверами двигателя DRV8825 и A4988 (популярно, без специфичных особенностей)

- Различное расположение потенциометра, регулирующего предельный ток.
- Драйвер на DRV8825 поддерживает 1/32-шаг в микрошаговом режиме.
- Драйвер A4988 максимум 1/16-шага.
- Драйвер DRV8825 обеспечивает ток до 1,5А на фазу без радиатора,
- Драйвер A4988 обеспечивает ток 1А на фазу без радиатора.
- Драйвер A4988 дешевле драйвера DRV8825.

Справочная информация.

Схема подключения драйвера A4988:



Распределение по фракциям

Samovar 7.00: поправки 11 положений сервопривода задаются в поле «Поправки сервопривода (11 чисел)» на вкладке «Оборудование» конфигуратора; он записывает их в массив servoDelta файла Samovar_ini.h.

Самовар может разделить весь перегон на 11 частей, согласно составленной вами программы отбора.

Примечание.

Если распределение на части не нужно или вы сами будете менять емкости, использование данного узла не является обязательным. Самовар будет работать и без него.

Для изготовления понадобится сервопривод используемый в проектах Ардуино.

Сервопривод [SG90](#)

(Rc Mini Micro 9g 1,6 кг сервопривод SG90 180° Для RC 250 450 для Arduino DIY)



Внимание!

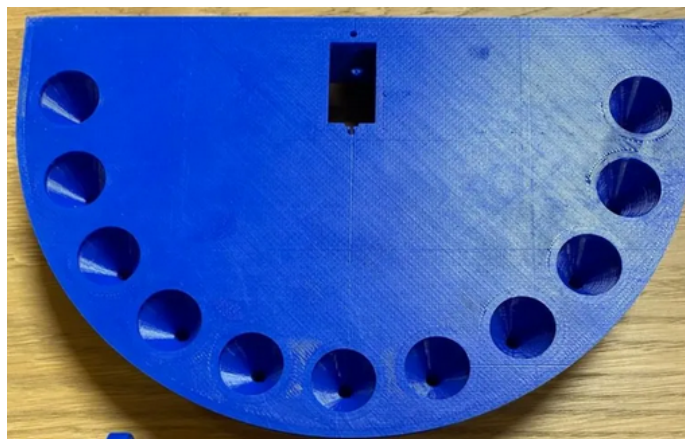
В Самоваре используется **сервопривод с углом поворота 180°**, любые другие будут работать не корректно (сервоприводы могут быть с углом поворота 90°, 180°, 360° и полного вращения).

Внимательно выбирайте привод перед приобретением. Если есть сомнения, лучше отказаться и приобрести у другого продавца с явно заявленным углом поворота.

Распиновка проводов: красный — +5 В, коричневый — GND, жёлтый или оранжевый — сигнал управления. Сигнальный вход подключается к GPIO25 на ESP32 DevKit либо к GPIO10 на ESP32-S3.

Для проверки угла поворота сервопривода можно воспользоваться специальным [скетчем](#). Скетч помещается в один каталог с файлами Самовара (... Documents\Arduino\...) и устанавливается в ESP32 вместо Самовара. Как установить скетч смотрите в [Руководстве пользователя](#). После установки скетча в ESP32 сервопривод должен начать двигаться от начального положения до крайнего и обратно, в режиме цикла. После проверки сервопривода скетч Самовара нужно будет вновь загрузить в ESP32.

Для сервопривода SG90 разработана модель стола разбора для печати на 3D принтере. Файлы модели находятся в каталоге 3D который входит в каждый архив с релизом Самовара. Это файлы **Samovar_plate.stl** и **Samovar_horn.stl**. Печатать нужно пластиком стойким к ацетону и спирту , например PLA.



SG90 — один из самых дешевых и популярных сервоприводов. У некоторых пользователей возникали проблемы с его стабильной работой. Это выражалось в том, что привод с трудом доходил до нужной позиции или в какой то момент коромысло начинало "дрожать". Как показывает опыт это связано с неудачным расположением трубки подвода продукта, её излишним размером, жесткостью или наоборот гибкостью. Заменить SG90 можно на:

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



Этот привод дороже предыдущего, но он и мощнее (соответственно потребляет больше энергии), может быть как с пластиковым редуктором, так и с металлическим (для наших целей не имеет значения), имеет в поставке больше аксессуаров.

В Самоваре используется **сервопривод с углом поворота 180°**, любые другие будут работать не корректно.

Распиновка проводов: красный — +5 В, коричневый, серый или чёрный — GND, жёлтый или оранжевый — сигнал управления. Сигнальный вход подключается к GPIO25 на ESP32 DevKit либо к GPIO10 на ESP32-S3.

Для этих сервоприводов тоже разработана модель стола разбора для печати на 3D принтере. Это файлы **plate_996_servo.stl** и **horn_996_servo.stl**. Файлы модели находятся в каталоге 3D, который находится в архиве с релизом Самовара. Печатать пластиком стойким к ацетону и спирту.

Примеры исполнения и доработки стола разбора:



Посмотреть на форуме: [пост 1](#), [пост 2](#)

Корректировка позиционирования

Каждый сервопривод имеет свои погрешности и индивидуально настраивается на позиции стола. Если стол разбора будет напечатан или изготовлен по шаблону, с одинаковыми расстояниями между воронками, нужно будет произвести корректировку углов поворота.

Углы поворота сервопривода настраиваются в файле `Samovar_ini.h` (см. [Загрузка и обновление скетча](#) и [Настройка скетча](#)), это строки:

```
//Корректировка для угла поворота сервопривода.  
int8_t servoDelta[11] = {0, -2, -3, -4, -3, -2, 0, 0, 0, 0, -2};
```

Значения в фигурных скобках и есть корректировка угла по каждой ПОЗИЦИИ. Всего 11. Отсчет приемных воронок начинается с нулевой позиции. Значения корректуры указываются в градусах. Каждая позиция корректируется индивидуально. От корректировки соседних позиций не зависит.

Для корректировки составляем тестовую программу отбора, для режима "Ректификация"

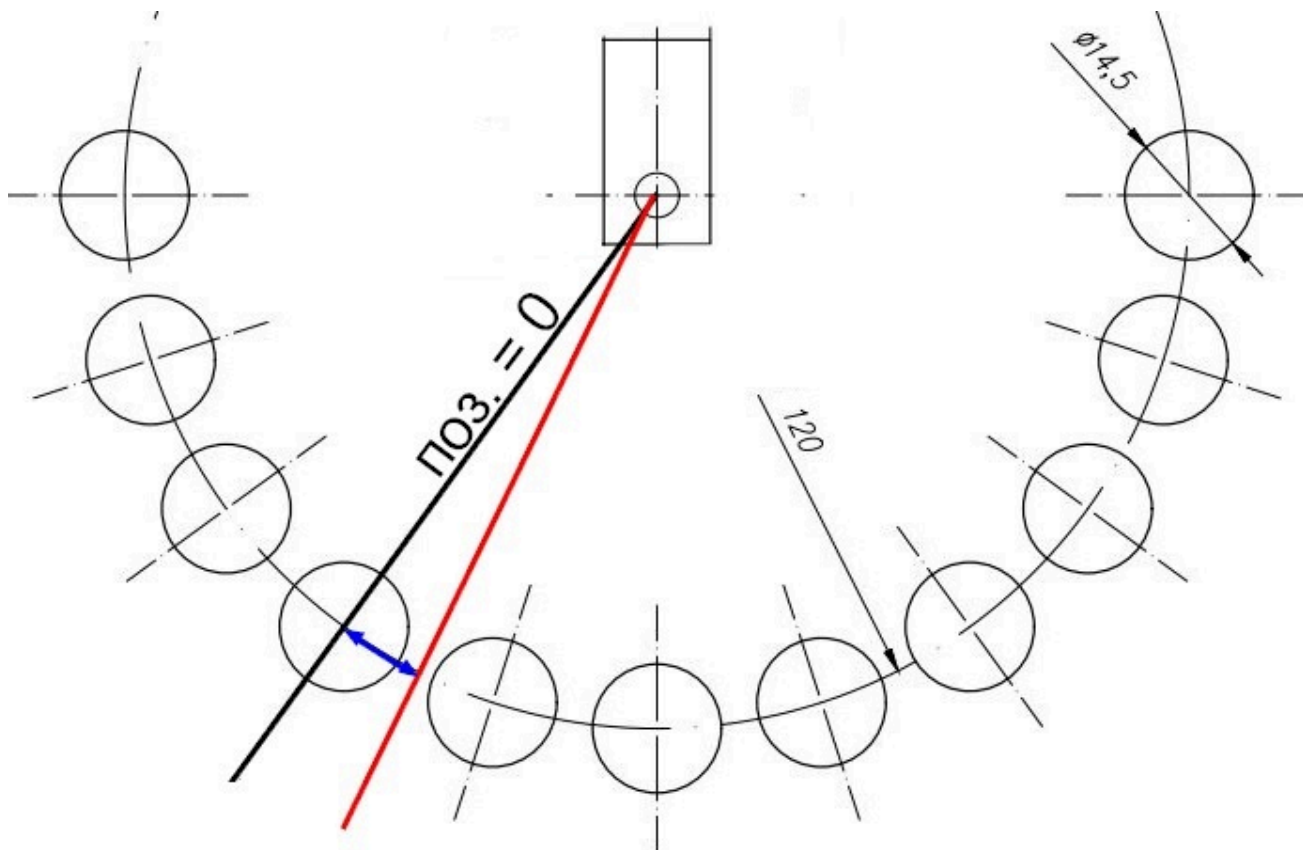
```
H;450;0.10;0;0.00;45  
H;450;0.10;1;0.00;45  
H;450;0.10;2;0.00;45  
H;450;0.10;3;0.00;45  
H;450;0.10;4;0.00;45  
H;450;0.10;5;0.00;45  
H;450;0.10;6;0.00;45  
H;450;0.10;7;0.00;45  
H;450;0.10;8;0.00;45  
H;450;0.10;9;0.00;45  
H;450;0.10;10;0.00;45
```

Нужно скопировать этот блок текста и вставить в [текстовое поле программы](#) отбора на вкладке "Программа" главной страницы web-интерфейса Самовара и сохранить.

Коромысло на сервопривод можно не устанавливать или устанавливаем, но так чтобы его можно было легко снять. Подключаем сервопривод, включаем Самовар. При старте Самовара сервопривод станет в стартовую позицию. Устанавливаем коромысло так чтобы оно смотрело на нулевую воронку.

Даем команду с Самовара "включить нагрев" (нагреватель должен быть отключен) и затем даем команду "Начать отбор". Коромысло останется на нулевой воронке.

Команда "Следующая программа" передвинет коромысло на следующую позицию - это воронка №1. Если положение коромысла совпало с условной линией проходящей через центр оси вращения сервопривода и центр приемной воронки, хорошо. Это ноль. Корректировка не нужна. Снова даем команду "Следующая программа", коромысло передвинется на следующую позицию. Если коромысло не дошло до нужного положения или проскочило - нужна корректировка. Рассмотрим на примере воронки №3. (не забываем - нумерация воронок начинается с нуля). Для данной воронки корректирующее значение в оригинальном скетче, это - 4 (минус 4)



Предположим коромысло "проскочило" и остановилось на красной линии. Нужно замерить длину дуги (синяя стрелка) между нулем и фактическим положением.

Допустим оно равно 10 мм.

Теперь нужно узнать какая длина дуги будет при повороте на 1 градус. Находим в сети [калькулятор](#) . Вводите значение радиуса (в данном случае 120), значение для угла ставим = 1.

Получаем, что при повороте на 1 ° длина дуги будет = 2 мм. У нас коромысло ушло на 10 мм от нужной позиции.

$10/2=5$ градусов.

Коромысло проскочило и его нужно вернуть назад, это будет отрицательное значение. В скетче уже стоит - 4 (минус 4)

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

Таким образом корректировка угла поворота для воронки №3, а это позиция №4, равна минус 9°

Это верно для конкретного сервопривода и стола, с воронками расположенным по окружности, радиусом 120 мм.

Подобным образом прогоняем все позиции коромысла и корректируем значения в фигурных скобках строки

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

После корректировки нужно вновь загрузить скетч в Самовар, установить тестовую программу отбора и проверить перемещение коромысла по позициям.

Самостоятельное изготовления стола разбора

Стол разбора с фиксированным расположением воронок. Пост [на форуме](#). Там же можно скачать шаблон для изготовления.

Стол разбора, почти не требующий корректировки значений угла поворота сервопривода.



Суть изготовления в том, что центра для сверления отверстий, размечаются по фактическому положению коромысла.

Инструкция по изготовлению [на форуме](#).

Воронки



- Купить:

[Мини-воронка кухонная металлическая со съёмным фильтром, 5 шт.](#)

[Маленькие пластиковые масляные воронки SUEF 2/5/10/20 шт., жидкие масляные воронки.](#)

[Мини-воронка кухонная из нержавеющей стали со съёмным фильтром, 5 шт](#)

-

- Изготовить на 3D принтере - файл воронки находится в каталоге 3D.

- Использовать [колпачки от флаконов](#) или медицинские [шприцы](#). Шприцы менее 5мл не рекомендуются - может быть переполнение, лучше 10 или 20 мл.

Если приемные воронки переполняются, самое простое - приемные емкости поставить как можно ниже, а стол поднять повыше. Столб жидкости в трубке от воронки до емкости придаст дополнительную скорость потоку.

Устранение возможных проблем.

При условии что сервопривод исправен

- В момент работы сервопривода Самовар перезагружается - возможно не хватает мощности преобразователя 12-5 вольт или у двигателя сервопривода излишнее потребление. Что можно сделать:

- Как временная мера. Увеличить емкость конденсатора C4. Можно и C3 тоже.

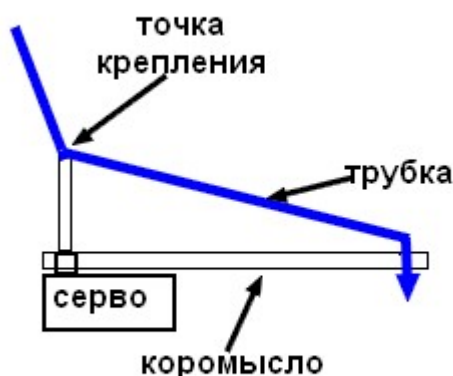
- Добавить второй преобразователь 12-5 вольт. Минусы объединить, а +5 подать на привод (можно запитать все потребители на 5 вольт от этого преобразователя)

- Коромысло "дрожит" над воронкой, даже без надетой трубки - плохая горизонтальная жесткость коромысла.

- Коромысло "раскачивается" над воронкой с надетой трубкой подвода жидкости.

- Излишне жестка трубка. Она постоянно "оттягивает" коромысло назад. Сервопривод все время пытается вернуть его на позицию.

- Трубка "болтается" над столом, она тянет коромысло то в одну, то в другую сторону. Как вариант, дополнительно закрепить трубку по центру оси вращения редуктора сервопривода.



Аварийная кнопка. Дополнительные аварийные датчики

Проверено по Samovar 7.00.

Кнопка или датчик подключается между аварийным входом и GND и срабатывает по низкому уровню: GPIO35 на ESP32 DevKit либо GPIO48 на ESP32-S3.

При срабатывании Самовар отключает нагрев, подачу воды, отбор и мешалку и блокирует повторный запуск до перезагрузки. Несколько нормально разомкнутых кнопок и датчиков можно соединить параллельно. Для ESP32 DevKit используйте окружение PlatformIO

Samovar_alarm_button либо добавьте **#define USE_ALARM_BTN** в **user_config_override.h**. В окружении **Samovar_s3** аварийный вход GPIO48 уже включён.

Датчик разлития спирта. [Начало](#). [Реализация](#).

На ESP32 DevKit аварийный вход — GPIO35, на ESP32-S3 — GPIO48. В обоих вариантах срабатывание происходит при замыкании входа на GND.

Для GPIO35 классического ESP32 обязательна внешняя подтяжка резистором 10 кОм к +3,3 В: у выводов GPIO34–39 нет внутренней подтяжки. Для GPIO48 ESP32-S3 прошивка включает внутреннюю подтяжку.

На печатной плате Samovar 1.3.2 разъём аварийной кнопки обозначен ALRM_BT (в старых материалах также встречается имя ALARM_BTN). Он соединён с GPIO35 и должен иметь подтягивающий резистор 10 кОм к +3,3 В.

Вот ссылки на посты: [ссылка1](#), и [ais77](#) [ссылка2](#), [ссылка3](#).

Рекомендованная конфигурация

Рекомендованная конфигурация добавляет к минимальной сборке исполнительные устройства, необходимые для автоматической работы:

- регулятор мощности и контактор для управления нагревом;
- клапан подачи воды и, при необходимости, насос охлаждения;
- шаговый двигатель и драйвер для отбора;
- датчики температуры куба, пара, царги и воды;
- датчик атмосферного давления;
- датчик уровня флегмы или другие используемые аварийные датчики.

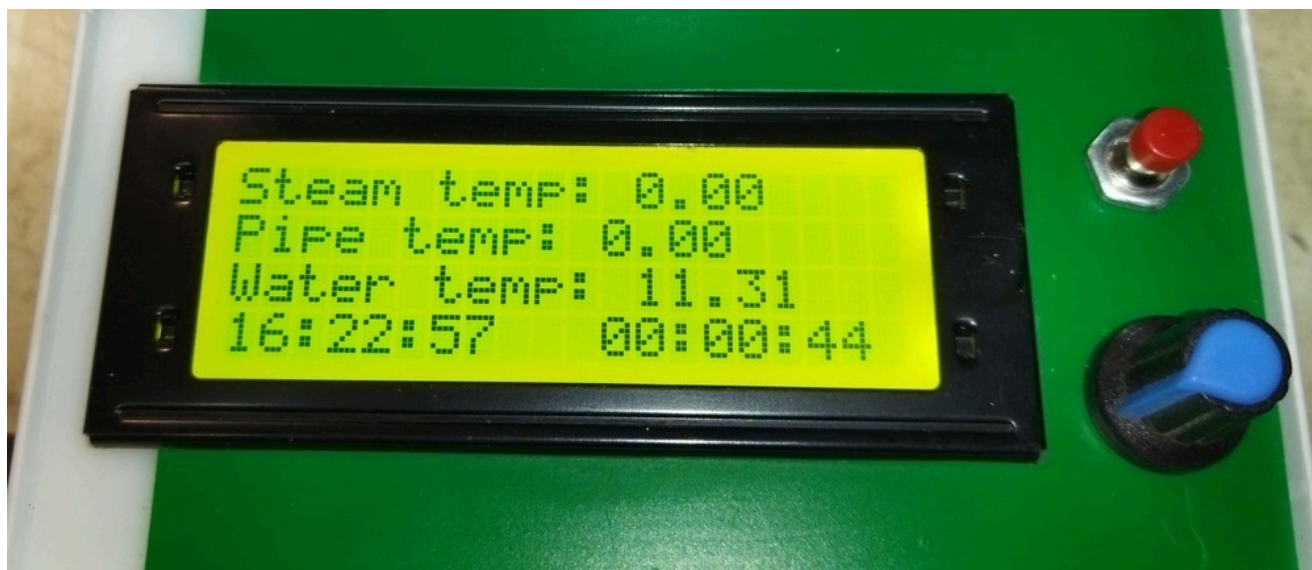
LCD-дисплей, энкодер и отдельная кнопка удобны для локального управления, но не обязательны: основные настройки и управление доступны через web-интерфейс.

LCD дисплей, энкодер, кнопка

Samovar 7.00: кнопка управления включается на вкладке «Оборудование» конфигуратора.

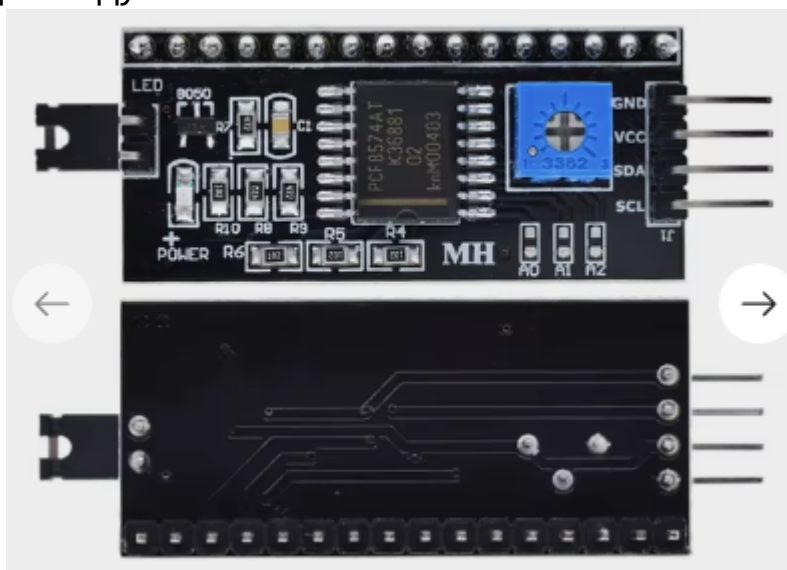
Внимание.

Самовар работает без дисплея, энкодера и универсальной кнопки. Вся информация доступна в web-интерфейсе, а при включённой интеграции — также в Blynk. Дисплей удобен рядом с установкой; энкодер без дисплея обычно не нужен, отдельная кнопка может быть полезна.



В проекте используется четырех-строчный дисплей: 20X4 LCD2004 с модулем I2C. Без модуля дисплей подключить к Самовару не удастся.

Вот так выглядит модуль I2C



А вот так дисплей 20x4 с распаянным модулем I2C.



Можно приобрести дисплей с уже припаянным модулем, или отдельно в комплекте и припаять самостоятельно.

Подсветка дисплея может быть синей, зеленой, желтой и белой (влияет на стоимость). Если подсветка не нужна, мешает, то ее , после настройки дисплея, можно отключить сняв джампер LED на модуле I2C. Текст будет читаем.

В основном дисплеи поддерживают только латинский шрифт, но есть и с поддержкой кириллицы. Самовар использует латиницу.

Подключение.

Дисплей подключается к плате Самовара 4-мя проводами от модуля I2C согласно распиновке модуля и платы.

Как правило при первом подключении загорается подсветка, а текста нет.

Покрутите подстроечный резистор на модуле I2C. Текст проявится.

Выставьте наибольшую контрастность.

На этом настройка дисплея закончена.

Примечание.

Если на дисплее часть строк или символов отображается не корректно пропаяйте тщательно еще раз соединения модуля и дисплея. Удалите флюс.

Энкодер

В проекте используется энкодер ЕС11 с "обвесом". Выглядит вот так.



Подключается к плате Самовара 5-ю проводами согласно распиновке. В основном управляет "пролистыванием" строк на дисплее LCD2004. Если энкодер прокручивает строки на дисплее не в ту сторону, поменяйте местами провода на контактах S1 и S2.

Кнопка.

Обработка Самоваром нажатия кнопки зависит от выбранного режима работы (см. [LCD дисплей и универсальная кнопка](#)).

Подойдёт нормально разомкнутая тактовая кнопка. Подключите её между входом и +3,3 В: GPIO39 на ESP32 DevKit или GPIO47 на ESP32-S3. Входу нужна внешняя подтяжка резистором 10 кОм к GND.

Если вы не используете кнопку, то на ESP32 DevKit резистор не нужен, но обязательно нужно в конфигураторе на вкладке «Оборудование» снять флажок «Использовать кнопку управления» или в файле `Samovar_ini.h` закомментировать использование кнопки:

```
//#define USE_BTN
```

```
//использовать кнопку
```

На ESP32-S3 вход кнопки GPIO47 включён всегда, независимо от этой настройки, поэтому резистор 10 кОм к GND нужен и без кнопки — иначе возможны ложные нажатия.

Внимание!

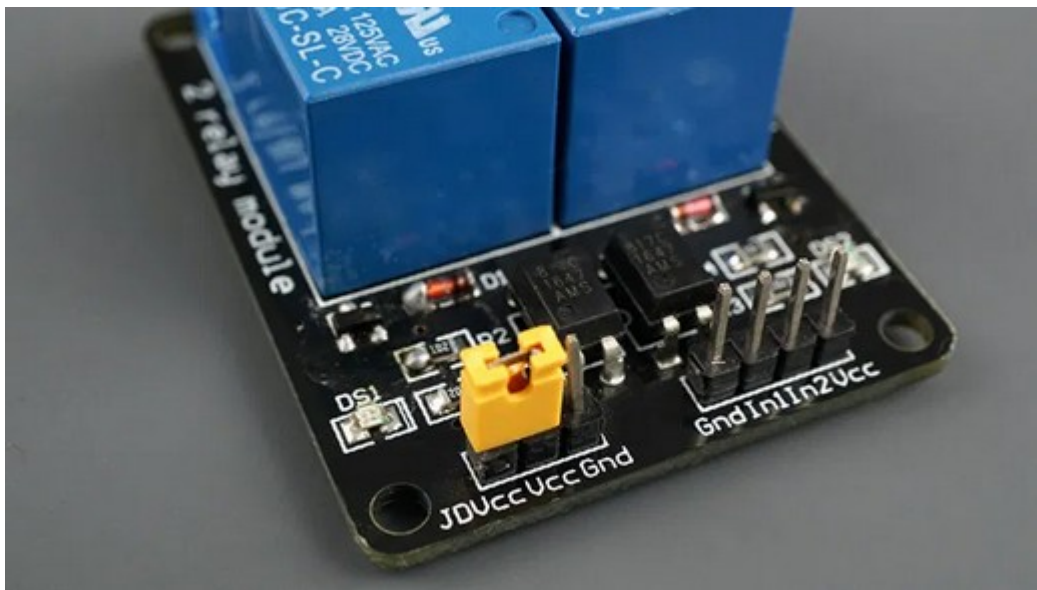
При старте удержание отдельной кнопки около двух секунд запускает точку доступа Wi-Fi. Удержание кнопки энкодера около двух секунд удаляет только сохранённые настройки Wi-Fi. Остальные настройки оборудования не сбрасываются

Как правильно подключить релейный модуль

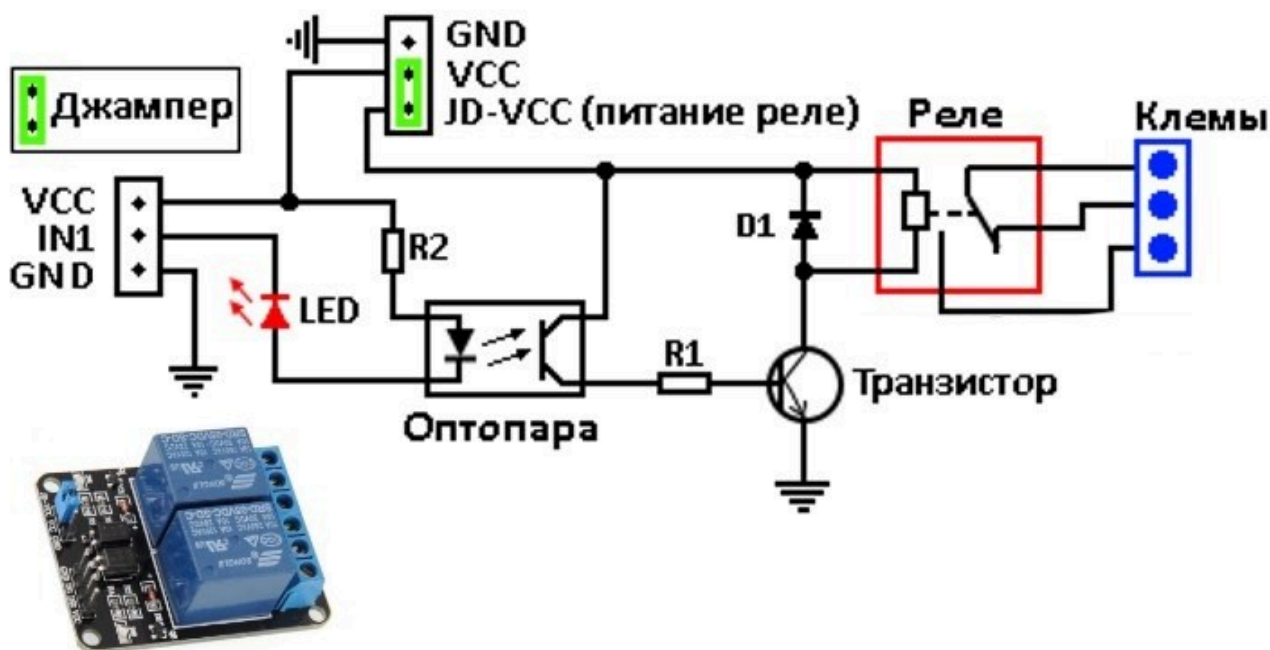
К ESP32 можно подключать модули реле на 5 вольт или на 12. На модуле обязательно должна присутствовать перемычка (джампер) JDVCC - VCC

Внимание:

Если перемычка - джампер JDVCC-VCC отсутствует, значит данный модуль не имеет гальванической развязки по питанию. Использование таких модулей не рекомендуется.



Принципиальная схема на примере одного канала:



Для подключения к ESP32:

Верхний по схеме разъем:

ОБЯЗАТЕЛЬНО! Удалить джампер!

- На контакт GND, подать минус от источника питания для реле (-5 или -12 в, в зависимости от используемых реле).
- Контакт VCC не используется (т.е. никуда не подключается).
- На контакт JDVCC подать плюс от источника питания для реле (+5 или +12 в).

Второй разъем:

- VCC подключить к +3,3 вольта от ESP32
- IN1, IN2 подключить к нужному выходу ESP32. Для полной гальванической развязки вывод GND этого разъема к выводу GND ESP32 можно не подключать.

Таким образом весь модуль, в том числе и реле будут запитаны от отдельного источника питания (5 или 12 вольт), а с Самовара будет приходить только 3,3 вольта: на резистор R2, светодиод оптопары и сигнальный светодиод.

Если на контакте IN1 будет логический ноль, то реле включится, если логическая единица (+3,3 вольта), то выключится

Примечание.

В основном, исполнение модулей реле: Low Level Trigger (LLT - Низкий уровень переключателя, по умолчанию включено). Это значит, что для включения реле на входе управления реле (IN1) должен быть логический "0". При подаче логической "1" на вход (IN1) реле отключается. Такие реле используются, когда микроконтроллер управляет большим кол-вом устройств для того, чтобы снизить нагрев и потребление тока микроконтроллером. А также в системах, где реле при подаче питания или "по умолчанию" должно быть включено.

Чтобы управлять модулем не логическим "0", а логической "1" :

- В WEB-интерфейсе Самовара, на вкладке [Настройки – Прочие](#), выбрать нужный уровень срабатывания модуля.

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

Поля **rele1...rele4** задают уровень, которым реле **включается**. Для описанного модуля с включением логическим нулём установите уровень **LOW**.

Внимание. На ESP32 DevKit реле №2 (мешалка) подключено к GPIO15 — это служебный вывод, который при сбросе платы аппаратно подтянут к высокому уровню. Модуль с включением логической единицей на этом канале кратковременно включится при каждой загрузке Самовара. Для реле №2 используйте модуль с включением низким уровнем (LOW).

Расширенная конфигурация

Расширенная конфигурация строится на рекомендованной и добавляет оборудование под конкретные режимы и сценарии.

Можно подключить [LCD-дисплей с энкодером](#), [сервопривод распределения по фракциям](#), один или два модуля [I2CStepper](#) для мешалки и насоса, [аварийную кнопку](#), дополнительные датчики, [внешний измеритель](#) и оборудование [НБК](#). Подключайте только те узлы, которые включены в выбранном окружении сборки и настроены в **Samovar_ini.h** или **user_config_override.h** (проще всего — через конфигуратор, который запускается из **flash_windows.bat** или **flash_macos_linux.sh**).

Оборудование для сыроварения

Samovar 7.00: для режима [Сыроварение](#) используются штатные выводы платы, отдельных разъёмов нет:

Датчик pH — аналоговый модуль PH-4502C. Его выход подключается к аналоговому входу LUA_PIN: GPIO34 на ESP32 DevKit, GPIO4 на ESP32-S3. Этот же вход занимает датчик давления MPX5010DP и функция analogRead() скриптов Lua, поэтому одновременно с датчиком pH они работать не могут. Показания переводятся в pH по калибровке, см. [Калибровка pH](#).

Мешалка — реле №2 (то же, что мешалка в режиме Пиво) или мешалка модуля I2CStepper по адресу 1; выбор задаётся в строке программы.

Сливной клапан — реле №4. В других режимах этот выход управляет разгонным ТЭНом, поэтому к нему подключают либо клапан, либо ТЭН.

Охлаждение — насос или клапан воды, как в остальных режимах; без USE_WATER_PUMP или USE_WATER_VALVE этап охлаждения в программе сыроварения не пройдёт проверку.

Измеритель

Измеритель от dranek — отдельное внешнее устройство для дополнительных измерений. Для базовой работы Самовара он не требуется и в состав прошивки Samovar не входит: Самовар его показания не читает.

Исходники измерителя лежат в репозитории Самовара в каталоге **pro_mini_ntc** (скетч для ESP8266, папки «Схема и плата» с принципиальной схемой и Gerber-файлами и «Справочная информация» с описанием). В сборку прошивки Самовара этот каталог не входит.

[Описание, схема и обсуждение измерителя на форуме](#). При сборке используйте схему из исходного сообщения автора.

I2CStepper

I2CStepper — отдельный модуль на Arduino Nano со своим шаговым двигателем, четырьмя реле, внешним датчиком, LCD 16×2 и энкодером. Текущая прошивка использует протокол I2C v3.

Модуль может быть использован как самостоятельное устройство.

Управление возможно как отдельно с помощью меню на LCD экране 16x2 и энкодера, так и при подключении к Самовару.

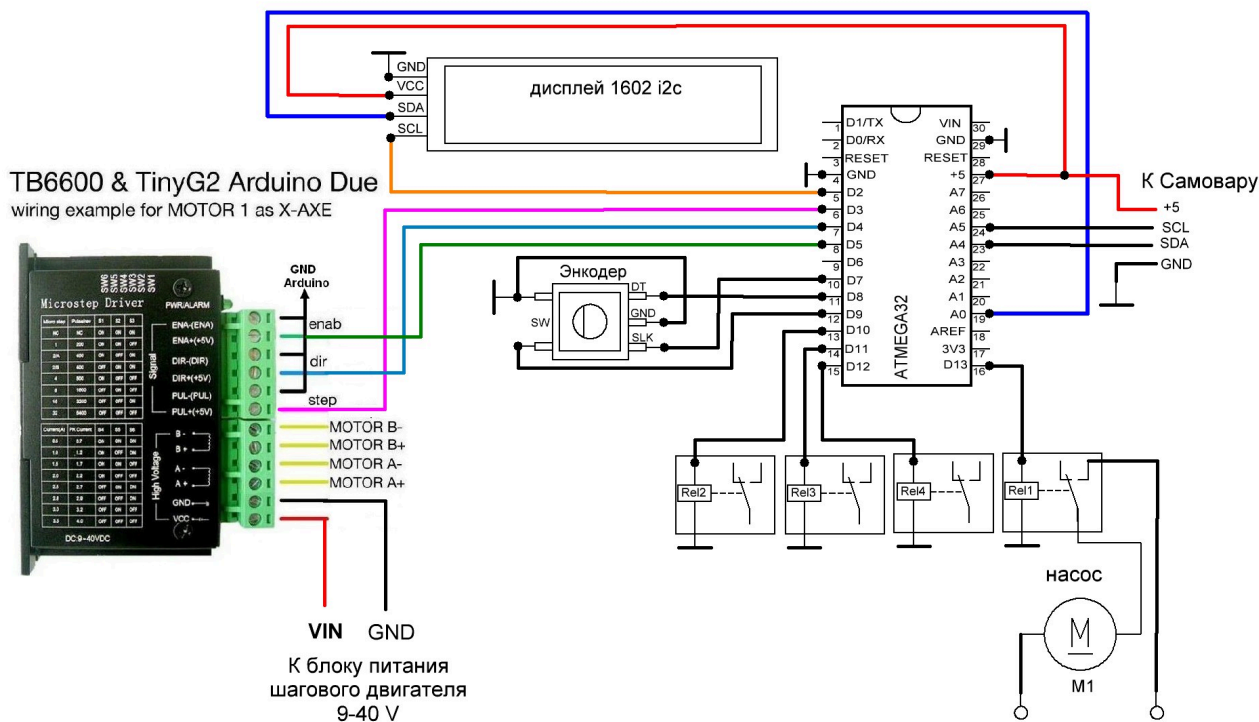
Расширитель подключается к Самовару по шине I2C четырьмя проводами (как LCD экран). Самовар понимает, что расширитель подключен и может им управлять.

Samovar поддерживает до десяти плат с адресами 1–10: нечётный адрес — мешалка, чётный — насос/наполнение. В работе используются платы с наименьшим адресом каждого типа. Мешалка работает по скорости, времени работы и паузе; насос — по мл/ч; режим наполнения — по заданным объёму и скорости.

Если устройство сообщает поддержку реле, первое реле переключается со страницы I2CStepper, все четыре — из Lua. Назначение реле задаёт пользователь: прошивка Самовара его не определяет.

Модули ищутся на шине при загрузке Самовара. Если модуль найден, на страницах режимов Дистилляция, Бражная колонна и Пиво появляется вкладка «**I2CStepper**», которая открывает страницу ручного управления платой: скорость, число шагов, запуск и остановка, реле 1. Настройки самой платы (адрес, режим, обороты, время работы и пауза мешалки, подача насоса, объём и скорость наполнения, шагов на 1 мл) правятся на вкладке Настройки → «I2CStepper» и хранятся в памяти Nano. Калибровка внешнего насоса запускается кнопкой «Калибровка внешнего насоса» на странице I2CStepper; результат записывается в поле «Шагов/мл» платы.

Схема соединений



Распиновка подключения периферии:

// Пины для I2C Master - подключение экрана и потенциально других устройств

// SDA_PIN 0 //A0

// SCL_PIN 2 //D2

// Пины для I2C Slave - связь с Samovar

// SDA_PIN A4

// SCL_PIN A5

// Пины для шагового двигателя

#define STEPPER_STEP 3

#define STEPPER_DIR 4

#define STEPPER_EN 5

//Пины для реле

#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

// Пины для Encoder

#define ENC_CLK 7 //S2

```
#define ENC_DT 8 //S1  
#define ENC_SW 9 //KEY
```

Совет.

Если планируется использовать расширитель как отдельное устройство, в отдельном корпусе, имеет смысл подключать его к Самовару через USB кабель A-B (компьютер - принтер). Непосредственно от дисплея Самовара берется питание для модуля +/- 5 вольт и два сигнала шины I2C. На корпусе Самовара монтируется разъем USB-B.

На корпусе расширителя монтируется разъем USB-A.

Таким образом, при подключении к Самовару Arduino Nano одновременно подключается и к питанию и к шине I2C. В случае использования как самостоятельное устройство, вместо Самовара используется отдельный блок питания, например зарядка от телефона.

При использовании встроенного блока питания на выход БП ставится разъем USB-B, и для подачи питания используется тот же кабель USB A-B, что исключает случайную подачу +5 вольт на Самовар.

Периферия.

Блок реле используется на напряжение 5 вольт, предпочтительно с управлением через оптопары. Можно использовать и 12-ти вольтовые реле с [подключением](#) по принципу как в Самоваре

Насос можно использовать любой. Двигатель постоянного напряжения или переменного одно- или трех- фазный. Потому как пуск и остановка двигателя насоса осуществляются через реле и далее через любой силовой коммутатор: контактор или твердотельное реле. Для небольших объемов можно присмотреться к насосу с магнитным приводом 15RM. Нержавейка, тихий, с хорошей производительностью.

Шаговый двигатель для мешалки или НБК можно использовать любой биполярный, при условии, что у него хватит мощности и крутящего момента. Рекомендуется Nema 23 с соответствующим драйвером.

Актуальная прошивка Arduino Nano и описание протокола v3 находятся в [репозитории I2CStepper](#). После перехода на протокол v3 проверьте адрес платы, режим и калибровку «Шагов/мл»: при смене версии структура EEPROM инициализируется заново.

Текущие режимы:

— Mixer: мешалка, обороты в минуту, время работы, пауза, реверс после паузы и плавный старт.

— Pump: непрерывная подача в мл/ч с необязательной паузой.

— Fill: дозированный налив заданного объёма в мл.

Внешний датчик может останавливать насос или ставить его на паузу. При зависании I2C сам модуль останавливает шаговик и выключает реле; после восстановления требуется подтверждение энкодером.

Внимание.

Функция Lua **set_stepper_target(spд, direction, target)** управляет только насосом I2CStepper (рабочей платой с чётным адресом): скорость, направление и число шагов. Если насос не отвечает на шине, функция возвращает 0; штатным шаговиком Самовара она не управляет.

--set_stepper_target(spд, direction, target) //spд - скорость в шагах в секунду,
direction - прямое или обратное направление,
target - количество шагов

При spд = 0 шаговый двигатель будет остановлен.

Пример

set_stepper_target(950, 0, 1000)

Пример задаёт скорость 950, направление 0 и перемещение на 1000 шагов.

Датчик кислотности pH и его подключение

В режиме Сыроварение датчик pH позволяет дождаться заданной кислотности на этапе «Ожидание pH». Измерительный комплект состоит из погружного электрода и платы PH-4502C. Электрод подключается к разъёму BNC платы; её аналоговый выход PO передаёт напряжение в Самовар. Не подключайте сам электрод непосредственно к ESP32 или ADS1115: для него нужна плата обработки сигнала.

Электрод должен подходить для измеряемого продукта и температуры. Датчик, рассчитанный на воду, может загрязняться белком и жиром в молоке; пригодность к непрерывному погружению определяется паспортом электрода. Калибровка не устраняет загрязнение или износ.

Подключение через ADS1115

ADS1115 — внешний АЦП, преобразователь напряжения в число. Для подключения нужны модуль ADS1115, два резистора 10 кОм с допуском 1% и конденсатор 100 нФ. Номиналы относятся к выходу платы PH-4502C, а не к кабелю электрода.

- Подайте на PH-4502C питание 5 В, на ADS1115 — 3,3 В. Соедините земли GND обеих плат и Самовара. Перед подключением проверьте маркировку выводов своей платы: у разных модулей порядок контактов отличается.
- Соедините PO платы PH-4502C с AIN0 ADS1115 через первый резистор 10 кОм. Между AIN0 и GND установите второй резистор 10 кОм. Конденсатор 100 нФ подключите параллельно второму резистору. Делитель уменьшает напряжение вдвое: 5 В на PO превращаются в 2,5 В на AIN0. Выход DO для измерения pH не используется.
- Подключите SDA и SCL ADS1115 к одноимённым линиям I2C Самовара — общей двухпроводной линии связи с модулями. Используйте контакты своей платы ESP32, указанные в её схеме; номера GPIO зависят от сборки. Подтяжки SDA/SCL должны быть к 3,3 В.
- Соедините ADDR с GND: адрес ADS1115 будет 0x48. Если этот адрес уже занят другим устройством, выберите свободный адрес ADS1115 и укажите его в настройке сборки. Два устройства с одинаковым адресом на одной линии работать не должны.
- Включите настройку сборки [USE ADS1115](#) со значением адреса, например [#define USE ADS1115 0x48](#), и загрузите прошивку. pH автоматически читается с AIN0. LUA_PIN при этом остаётся свободным. Если ADS1115 недоступен, запуск сыроварения и сохранение калибровки pH блокируются.

Проверьте мультиметром напряжение между AIN0 и GND перед подключением к ESP32. Диапазон измерения ADS1115 в прошивке — $\pm 4,096$ В, но он не разрешает подавать на вход напряжение выше питания микросхемы. После изменения схемы или перехода на ADS1115 выполните калибровку заново.

Подключение без ADS1115

Если USE_ADS1115 выключен, выход того же делителя подключается к LUA_PIN вместо AIN0. PH-4502C питается от 5 В; его GND соединяется с GND Самовара. На вход ESP32 нельзя подавать 5 В напрямую. LUA_PIN также используется датчиком давления MPX5010DP: подключайте к нему только одно устройство. Во время сыроварения этот вход занят измерением pH и недоступен для чтения из Lua.

Если показания скачут при включении оборудования

Сначала сравните показания в отдельном пластиковом стакане с буфером и в баке, включая насос, мешалку и нагрев по одному. Связь скачков с включением оборудования указывает на возможные электрические помехи. Медленный уход показаний в буфере при выключенном оборудовании требует проверки электрода и PH-4502C; развязка не исправит их неисправность.

Для устранения помех через общее питание и землю можно разделить измерительную часть:

электрод → PH-4502C → делитель 1:2 → ADS1115 → ADuM1250 → ESP32. ADuM1250 — модуль электрической развязки I2C: передаёт данные, не соединяя электрически две стороны.

Питайте PH-4502C, ADS1115 и измерительную сторону ADuM1250 от отдельного USB-пауэрбанка с выходом 5 В и поддержкой малой нагрузки. ADS1115 в этой схеме также получает 5 В; делитель на его входе остаётся. Во время измерения пауэрбанк не должен заряжаться или питать другие части Самовара.

Сторону ADuM1250, обращённую к ESP32, питайте от 3,3 В Самовара. Подключите SDA/SCL ADS1115 к стороне 5 В, SDA/SCL Самовара — к стороне 3,3 В. Подтягивающие резисторы каждой стороны должны соединяться с питанием именно этой стороны.

Не соединяйте земли двух сторон. Если соединить их проводом, USB-кабелем или через другое оборудование, развязка будет нарушена. Остальные устройства I2C Самовара остаются со стороны ESP32.

Расположите PH-4502C и ADS1115 рядом, отдельно от регулятора нагрева. Провод ПО сделайте коротким, кабель электрода проложите подальше от силовых проводов. Адрес ADS1115 и настройка USE_ADS1115 остаются прежними.

Это вариант подключения против электрических помех, а не гарантия точности электрода. После калибровки повторите проверку с включением оборудования; схема не должна давать воспроизводимый сдвиг pH при переключениях.

Программный фильтр и ограничения

В прошивке с медианным фильтром показание pH рассчитывается по пяти последовательным измерениям с интервалом в секунду. Медиана — среднее по порядку число после сортировки: ряд 5,42; 5,41; 3,10; 5,43; 5,42 даёт 5,42. Фильтр убирает короткие выбросы, но не исправляет постоянную ошибку и медленный уход показаний. При плавном изменении задержка составляет около двух секунд.

pH становится доступен после заполнения всех пяти отсчётов. После ошибки чтения, недопустимого показания, перерыва между измерениями больше пяти секунд, изменения коэффициентов калибровки или сброса процесса окно заполняется заново. Старый результат не выдаётся за новое исправное измерение. На этапе «Ожидание pH» сохраняется подтверждение достижения порога в течение 30 секунд.

Фильтр применяется к обоим подключениям — через ADS1115 и через LUA_PIN. Поле ADC на странице калибровки показывает свежий исходный отсчёт, без этого фильтра; дождитесь его стабилизации перед фиксацией точки. Самовар не учитывает температуру в пересчёте pH: сравнивайте результаты и контрольные пробы при одинаковой температуре. Обновление документации сайта не обновляет прошивку на вашем устройстве.

Калибровка и уход

Откалибруйте весь собранный комплект по двум свежим буферным растворам около pH 7 и pH 4. Вводите значения с этикетки раствора для температуры измерения. Порядок работы с кнопками описан в разделе [Калибровка pH](#).

После молока очищайте электрод совместимым средством для белковых и жировых загрязнений по инструкции производителя. Храните чувствительный конец в рекомендованном растворе для хранения электродов, не сухим и не в дистиллированной воде. Не превышайте допустимую температуру электрода. После очистки проверьте показания в свежей порции буфера.

Справочные материалы: [ADS1115](#), [ADuM1250](#), [уход за электродами](#), [причины электрических помех при измерении pH](#).

Дополнительные возможности

Дополнительные функции можно реализовать через Lua и расширители PCF8575/PCF8591 по I2C. Lua включается отдельным окружением сборки **Samovar_lua** в PlatformIO, флажком «Использовать Lua» на вкладке «Оборудование» конфигулятора либо директивой **#define USE_LUA** в **user_config_override.h**; расширители включаются там же директивами **USE_EXPANDER** и **USE_ANALOG_EXPANDER** (в конфигураторе — поля «Адрес расширителя PCF8575» и «Адрес расширителя PCF8591»).

Программирование дополнительных функций реализовано на основе [скриптового языка](#) программирования [Lua](#). Редактор скриптов доступен на странице Настроек – Редактор.

На расширитель портов можно вешать любые исполнительные устройства или кнопки (датчики, которые работают как кнопки), и самому определить, что должно происходить в тот или иной момент при работе Самовара. Файл **init.lua** выполняется при инициализации Lua. Во время работы примерно раз в секунду запускаются общий **script.lua** и режимный скрипт, выбранный в настройках. Изменять их можно через файловый редактор web-интерфейса.

Можно сохранять переменные в памяти Самовара, запускать до 10 таймеров, получать статус работы Самовара и т. д. То есть, можно подключать дополнительное оборудование и автоматически управлять им в процессе работы Самовара

Можно проверить, что не сработал датчик безопасности, а если сработал, то прямо из скрипта можно выключить Самовар, увеличить или остановить подачу воды. Сценариев может быть много. Определяются они конкретным оборудованием.

При этом базовый функционал Самовара не меняется.

Скриптам доступен один аналоговый вход платы — **LUA_PIN** (GPIO34 на ESP32 DevKit, GPIO4 на ESP32-S3), его читает функция **analogRead()**. **Samovar 7.00**: в режиме [Сыроварение](#) этот вход занят датчиком pH, и обращение к нему из скрипта завершится ошибкой. Выходы реле, насоса и кнопок из скрипта переключать нельзя — только через расширитель портов или штатные функции Самовара.

Монитор порта по Bluetooth

Встроенный Bluetooth модуля ESP32 прошивка Самовара не использует. Удалённый монитор последовательного порта можно собрать на внешних модулях HC-05 или HC-06: они просто передают по радиоканалу то, что Самовар выводит в основной диагностический порт (тот же, что виден в мониторе порта Arduino IDE или PlatformIO). Порт работает на скорости **115200 бод**. Подключите вход RX Bluetooth-модуля к TX основного последовательного порта (вывод TX0 платы), объедините GND и настройте модуль на 115200 бод. Напряжение питания выбирайте по маркировке конкретной платы модуля; сигнальные линии ESP32 имеют уровень 3,3 В. Выход TX модуля к RX Самовара для одного только просмотра журнала не нужен. UART2 с выводами RXD2/TXD2 (GPIO16/GPIO17 на ESP32 DevKit, GPIO18/GPIO17 на ESP32-S3) оставьте регулятору мощности — это другой порт.

В этот же порт попадает вывод функции **print()** скриптов Lua. В каталоге **Lua_UART** исходников проекта есть пример отдельного приёмника на Arduino: он принимает по UART строку, разбивает её по запятым на 8 частей (до 20 символов каждая) и выводит их на дисплей. Не подключайте модуль непосредственно к сетевым цепям или к выходам регулятора.