

SAMOVAR

用户手册

固件 7.00

最新版本: <https://www.samovar-tool.ru/doc/zh/manual/>
生成于 01.10.2026

目录

从哪里开始

SAMOVAR 综合系统组装指南

为上传准备 ESP32

首次上传草图

连接 Wi-Fi

检查安装是否正确

草图设置*

更新草图

远程控制

移动应用与网站

使用的调压器

使用辅助加热管

冷却

紧急按钮。附加紧急传感器

LCD 显示屏和万能按钮

网页界面。主页

设置：基本

温度

泵

啤酒

奶酪

NBK

其他

保存设置

"附加"选项卡

系统消息

云端日志和图表

本地日志和图表

Samovar 工作模式

初次蒸馏模式

蒸馏程序

醪塔

连续醪塔

精馏模式

精馏模式的取酒程序

编辑取酒程序

程序计算器

酒头取酒

酒心取酒

预液泛

酒尾取酒

暂停

啤酒模式

糖化程序

搅拌器、泵、温度

洗糟水

发酵程序

"附加"选项卡

加载 BeerXML 配方

网站和应用中的啤酒酿造记录

低温慢煮

奶酪制作模式

奶酪制作程序

网站奶酪配方目录

pH 校准

手动模式下的工作

手动设置取酒速度

手动设置调压器电压

监控模式下的工作

功能扩展

I2CStepper

添加物投料器（啤酒和奶酪）

高压灭菌锅控制

关于 Lua 脚本语言

脚本示例

问题解答。FAQ

链接。实用建议

版本历史

从哪里开始

Samovar 的目标是把各种现成的设备组合起来，并在统一的理念下让控制它们尽可能简单。

Samovar 的核心理念：通过自动化，把用户从酿造、蒸馏、精馏等过程中的日常控制操作中解放出来。

安全是 Samovar 的第一要务。任何超出允许范围的情况都会被 Samovar 立即停止。同时，得益于开源代码 以及可以使用 lua 脚本，Samovar 足够灵活，几乎可以针对任何任务调整参数。

用什么最小配置就能搭建 Samovar？

只需一块 ESP32 微控制器，并往里面加载程序（草图（sketch））。然后让 ESP32 连接到您的 WiFi 网络，电脑上就会出现 Samovar 界面。ESP 通过 USB 接口供电。但它还无法显示任何数据。当然，它也无法控制设备。

现在需要把 Samovar 连接到设备上。

参见 [SAMOVAR 软硬件综合系统组装指南](#)（在新窗口中打开）

首先连接温度传感器。由于电脑通常在一个地方，而精馏塔在另一个地方，可以用任意手机充电器给 ESP 供电。

这时 Samovar 已经成为您的顾问。它会在显示器上显示并向智能手机发送消息：加速加热完成、打开冷却水、稳定完成、温度超限、暂停等等。同时您还能看到所有需要的温度。如果蒸汽冲出，会出现消息；还有图表，可以分析精馏塔的工作情况，等等。但稳定性和安全仍要靠您自己。Samovar 只提供建议，旋钮和“龙头”还得您亲自去拧。时间一长就会觉得累，需要实现一些自动化。

以初学者的标准设备为例。

一根 3 kW 的加热管和一个带旋钮的普通可控硅调压器……

我们来实现加速加热的自动化。在加热管组件中，留 2 根加热管接在调压器上，另一根接到 16 A 固态继电器（SSR）上。把继电器的控制端接到相应的引脚（第四路继电器通道，RELE_CHANNEL4 — D13）*。

通电。在调压器上只设置精馏塔正常工作所需的电压，比如 200 伏，然后向 Samovar 发出“开启加热”（«Включить нагрев»）命令，除了调压器之外，它还会通过 SSR 以全功率接通辅助加热管。当蒸汽到达精馏塔、塔顶传感器检测到蒸汽时，Samovar 会关闭辅助加热管，只保留调压器。

这样您就不必守着等蒸馏釜沸腾后自己去调低电压了。Samovar 会替您完成。

但同样，为了稳定精馏塔而不停地拧调压器旋钮也不是什么乐事……尤其是精馏要持续 20 个小时左右。

因此，为了节省精力、保护健康，最好首先购置一台 [稳压调压器](#)（RMV-k、KVIC、RM），或者自己动手制作一台 SEM_AVR 调压器。

这样就开启了以前无法实现的功能：

1. 根据运行的程序自动控制功率，包括让精馏塔一直保持在预液泛状态**、按蒸馏釜温度关闭加热，以及最重要的——在蒸汽冲出或冷却水温度超限时自动关闭加热！
2. 当然，还有稳定的加热。

就这样，借助 Samovar 一步一步地逐渐连接其他设备，几乎可以实现整个过程的自动化。每增加一台新设备，都会带来新的功能。

请注意，连接 SSR 或稳压调压器大概已经是手机充电器能承受的极限了。再往后发展就需要更可靠的电源。

* ESP32 输出 3.3 伏电压，电流有限，因此并非所有固态继电器 都能 直接 “开箱即用”。

** 在使用液位传感器和某些类型调压器的情况下。

SAMOVAR 综合系统组装指南

Samovar 软硬件综合系统的文档由 两个独立部分组成:

- 编程、设置和使用指南（也就是您现在正在阅读的这一部分），以及
- Samovar 综合系统自行组装指南。

下面列出组装指南的各个章节，但由于该指南仍在补充中，部分章节的链接可能已经过时。
建议直接查看 [目录](#)，它位于 [Samovar 综合系统自行组装指南](#) 中。

- [从哪里开始](#)
- [安装 ESP32](#)
- [最小配置](#)
- [蜂鸣器](#)
- [大气压力传感器](#)
- [温度传感器](#)
- [所用的调压器](#)
- [调压器概述](#)
- [如何以无线方式将调压器连接到 Samovar](#)
- [加热管](#)
- [使用辅助加热管](#)
- [接触器。漏电保护器。紧急断电](#)
- [冷却](#)
- [使用隔膜泵](#)
- [使用离心泵](#)
- [不用水泵](#)
- [回流液位传感器](#)
- [驱动器。步进电机。](#)
- [按馏分分配](#)
- [紧急按钮。附加紧急传感器](#)
- [推荐配置](#)
- [LCD 显示屏、编码器、按钮](#)
- [如何正确连接继电器模块](#)
- [扩展配置](#)
- [计量器](#)
- [I2CStepper](#)
- [附加功能](#)
- [NBK（连续醪塔）](#)
- [使用 PlatformIO 编译和烧录](#)

为上传准备 ESP32

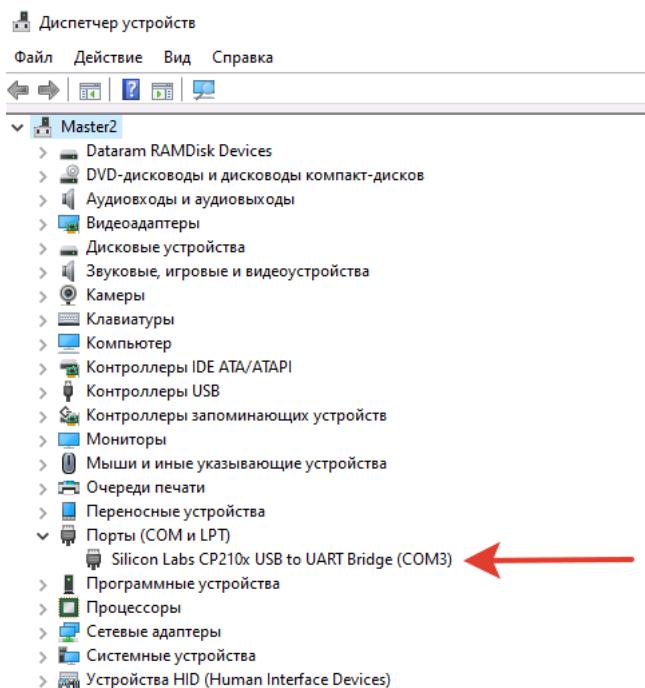
Samovar 7.00: 下面使用 Arduino IDE 的说明仍然是有效的安装方法。也可以同样使用配置工具：Windows 上用 flash_windows.bat（仅限 64 位 Windows 10/11），macOS/Linux 上用 flash_macos_linux.sh。配置工具会自行安装 PlatformIO 和所需版本的 ESP32 核心，不需要 Arduino IDE。参见[使用 PlatformIO 编译和烧录](#)。无论哪种方式，都需要 USB 控制器驱动（见下文）。

如果您有 ESP32，现在就可以上传草图并测试 Samovar 的工作。

首先，只需把 ESP32 连接到电脑的 USB 端口。

ESP32 开发板带有板载 USB 和 **CH340/CH341** 或 **CP210x** 控制器（商品页面上通常会注明）

打开“此电脑 > 属性 > 设备管理器 - 端口 (COM 和 LPT)”（«Этот компьютер > Свойства > Диспетчер устройств - Порты (COM и LPT)»）查看，应该出现一个新端口。



如果电脑检测到了新硬件但没有安装驱动，新端口就不会出现。需要下载您的 ESP32 开发板上控制器的驱动并手动安装。

下载驱动 [CH340/CH341](#)（Windows、Mac、Linux）

从我们的网站下载 驱动 [210/341](#)，仅限 Windows。

要把草图上传到 ESP32，需要安装 Arduino IDE。它用作草图编辑器和草图编译器，也用于把编译好的程序上传到 ESP32。。

下面及后续章节介绍一种首次把 Samovar 草图安装到 ESP32 的方法，可以保证成功。

下载 并安装 Arduino IDE [1.8.x 版](#)。

Legacy IDE (1.8.X)

Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Arduino IDE 1.x documentation](#) for installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer

Windows ZIP file

Windows app Win 8.1 or 10

Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

Mac OS X 10.10 or newer

Release Notes

Checksums (sha512)

1

Скачать установочный файл

2

Скачать zip-архив

下载安装程序

用于安装到操作系统中的安装程序

1

2

下载 zip 压缩包

无需安装到操作系统即可运行

无论选择哪种方式，网站都会请您感谢开发者，是否感谢由您决定。

如果只想下载程序，点击“JUST DOWNLOAD”按钮。随后会打开另一个窗口，邀请您订阅新闻，再次点击“JUST DOWNLOAD”按钮。

文件会被下载。如果选择的是 zip 压缩包，把它解压到电脑上的任意位置（目录路径中最好不要有西里尔字母）。如果选择安装程序，建议不要直接装到默认目录，而是在目录名中加上版本号。这样以后使用更方便。

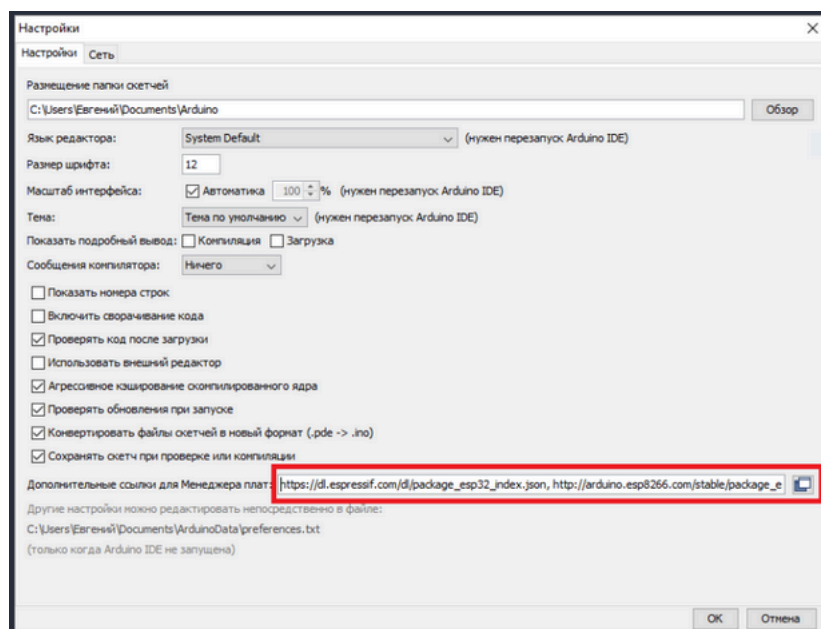
例如。

C:\Program Files(x86)\Arduino1_1_18;

C:\Program Files(x86)\Arduino1_1_16

安装 Arduino IDE 后，需要让它支持 ESP32，默认情况下它不支持。

- 进入解压或安装的文件夹，运行文件 `arduino.exe`
- 在 Arduino IDE 中打开“首选项”（«Настройки»）窗口。为此，点击“文件”>“首选项”（«Файл»>«Настройки»）。
- 把链接 https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json 粘贴到“附加开发板管理器网址”（«Дополнительные ссылки для менеджера плат»）字段中，如下面的截图所示。
-



注意：

如果该字段中已有其他开发板描述的链接，可以用逗号分隔多个链接。像这样：

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json,

http://arduino.esp8266.com/stable/package_esp8266com_index.json

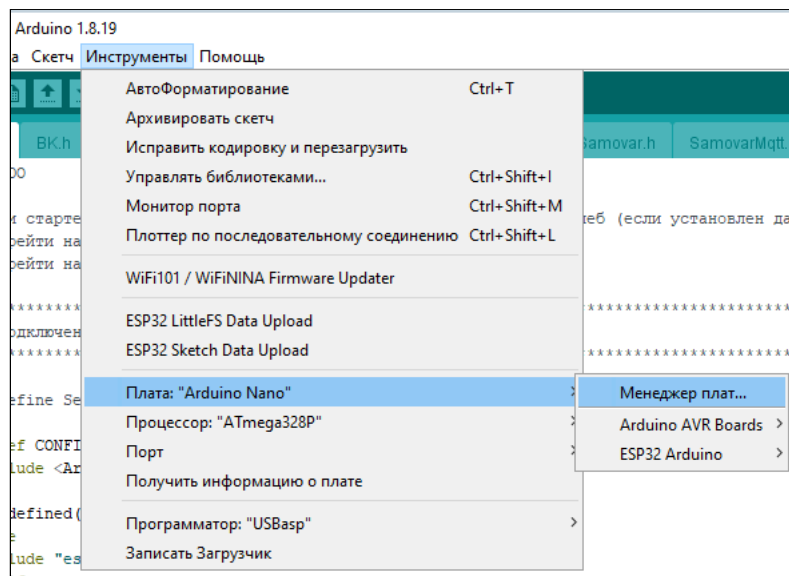
注意！

如果该字段中有链接 https://dl.espressif.com/dl/package_esp32_index.json，必须删除它。

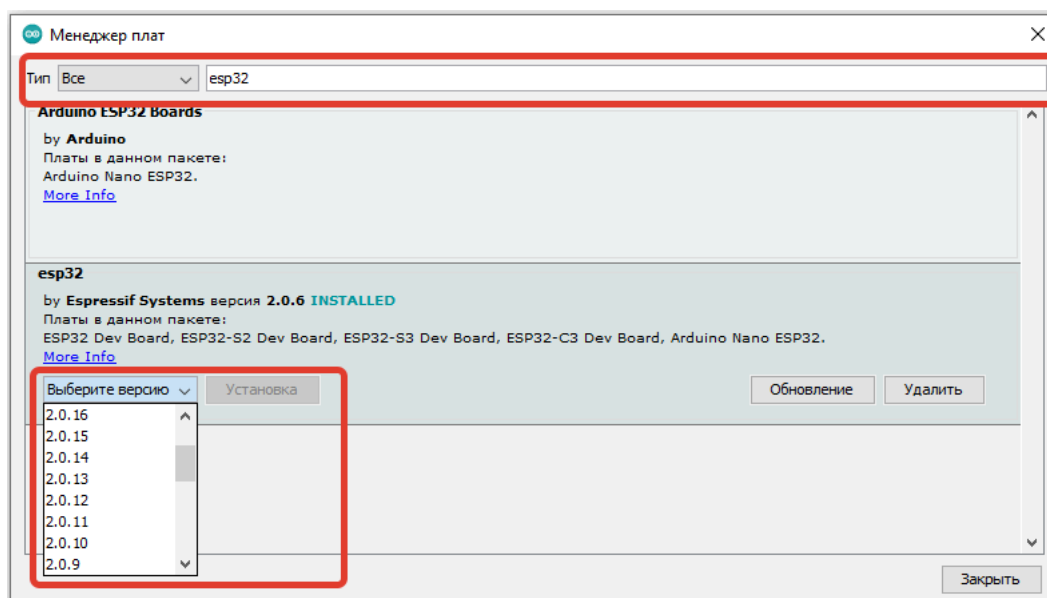
- 然后点击“OK”。

接下来，在 Arduino IDE 中打开“开发板管理器”（«Менеджер плат»）。

为此，点击“工具”>“开发板”>“开发板管理器...”（«Инструменты»>«Плата»>«Менеджер плат...»）。



打开一个新窗口。在搜索框中输入“**esp32**”。会找到 ESP32 的扩展包（或 另一个常用名称：SDK）：“**ESP32 by Espressif Systems**”。



在下拉菜单中选择 2.0.17 版，点击——**安装**（«**Установка**»），等待安装完成

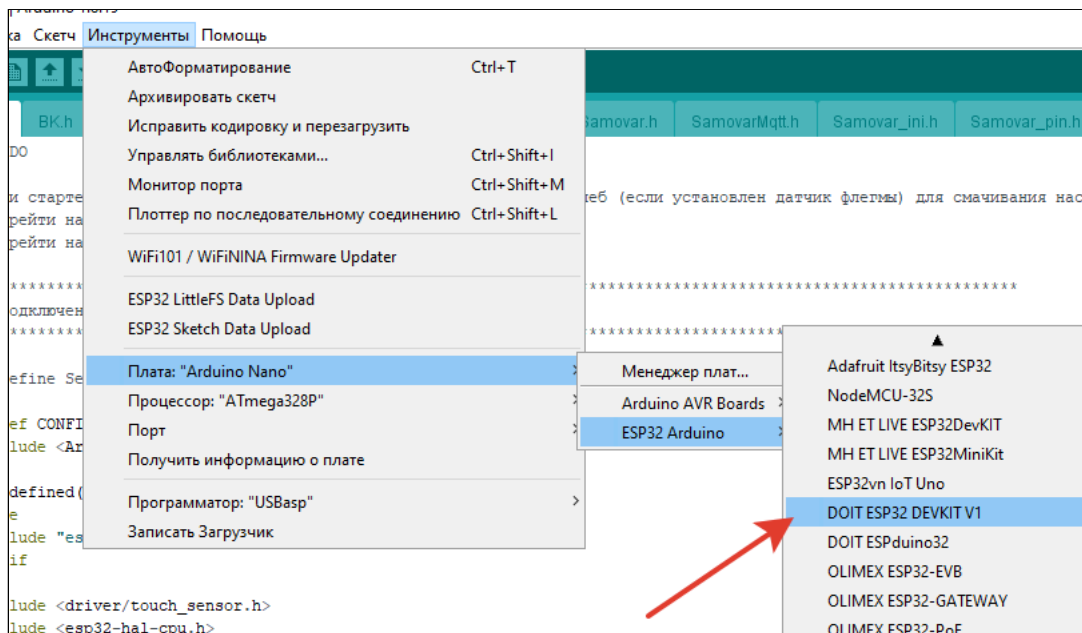
注意：

如果已经安装了扩展包但版本不同，请删除它并安装 2.0.17

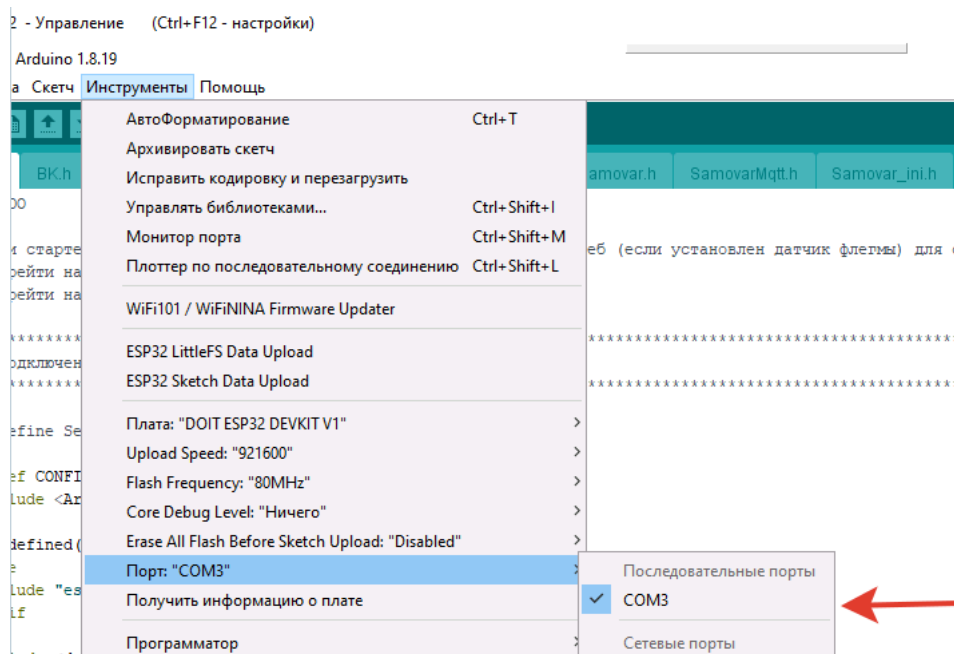
PlatformIO 配置工具使用的也正是 2.0.17 版。草图在 3.x 核心下也能编译，但其中 WiFi 更新（USE_UPDATE_OTA）会被自动禁用，而且需要手动把 Samovar 分区表添加到核心中（参见[草图更新](#)）。

重启 Arduino IDE，然后在菜单“工具”>“开发板 > ESP32 Arduino”（«Инструменты» > «Плата > ESP32 Arduino»）中选择所需的开发板。

在我们的例子中是“DOIT ESP32 DEVKIT V1”。



检查“工具 > 端口 > 串口”（«Инструменты > Порт > Последовательные порты»）中选中的是开发板所连接的端口。

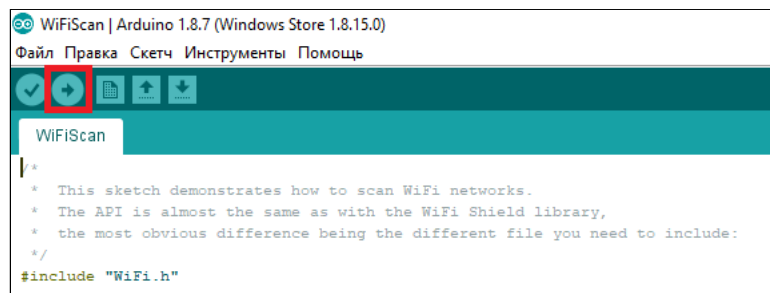


现在需要检查一切是否正确。

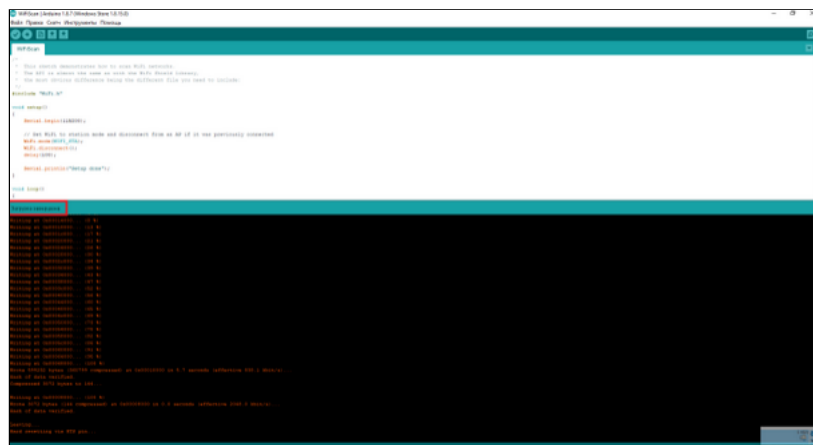
在 Arduino IDE 中打开示例草图“WiFi Scan”。

为此，点击“文件” > “示例” > （子菜单：DOIT ESP32 DEVKIT V1 的示例） > “WiFi” > “WiFi Scan”（«Файл» > «Примеры» > ... > «WiFi» > «WiFi Scan»），草图会在编辑器窗口中打开。

点击 Arduino IDE 中的“上传”（«Загрузка»）按钮。稍等片刻，代码会被编译并上传到 ESP32 开发板。上传按钮的图标如下：



草图成功上传到 ESP32 后，应该看到消息“上传成功” («Загрузка завершена»)



在 Arduino IDE 中打开“工具” > “串口监视器” («Инструменты» > «Монитор порта»)，并把波特率设为 115200。把 ESP32 重新连接到 USB 端口。这样就能在串口监视器中看到 ESP32 开发板周围的所有 Wi-Fi 网络。ESP32 的设置和检查到此完成。

现在可以进行[Samovar 草图到 ESP32 的首次上传](#)

如果 ESP32 以前用于其他项目，或者 Samovar 草图上传不正常，需要擦除 ESP32 模块的闪存

在 Arduino IDE 中

Arduino IDE 有“Erase Flash”选项。擦除闪存的步骤：

1. 在“工具”→“开发板”菜单 («Инструменты» → «Доска») 中选择合适的 ESP32 开发板。
2. 在“工具”→“端口”菜单中选择正确的 COM 端口。
3. 进入“工具”→“Erase Flash”菜单。在下拉菜单中选择“All Flash Contents” («Все содержимое флэш-памяти»)。
4. 点击“Erase Flash”选项 («Стереть флэш»)。进度显示在 IDE 底部的输出控制台中。完成后会出现确认消息。

使用 ESPTool

ESPTool 是基于 Python 的开源工具。擦除 ESP32 模块闪存的步骤：

1. 在电脑上打开终端窗口，按住 ESP32 模块上的 BOOT 按钮。
2. 在终端中输入命令，把 COMx 替换为 ESP32 的实际 COM 端口：
python -m esptool --port COMx erase_flash.
3. 按 Enter。
4. 终端中开始擦除后，就可以松开 BOOT 按钮。几秒钟后，ESP32 的闪存会被完全擦除。

使用配置工具

运行 flash_windows.bat (flash_macos_linux.sh)，选择开发板的 COM 端口，点击“完全擦除闪存” («Полностью очистить флеш») 按钮。确认后，固件、LittleFS 和所有已保存的设置都会被删除。该按钮仅在 USB 连接时有效。

重要：如果在消息“Connecting...”之后屏幕上不断出现新的一串点，说明 ESP32 模块没有处于引导/烧录模式。这时请重复上述过程，并在执行时按住 BOOT 按钮

首次上传草图

Samovar 7.00: 首次启动时，如果还没有保存的设置，Samovar 7.00 会自行创建带默认值的配置。无需手动恢复 Samovar_default_settings.txt。

Samovar 7.00: 支持通过 Arduino IDE 进行首次安装。另一种方法是在配置工具中选择开发板和参数，点击“烧录”（«Прошить»），对于新开发板，再单独点击“上传 LittleFS”（«Загрузить LittleFS»）。

首次把 Samovar 草图上传到 ESP32 之前，需要准备好电脑和开发板，详见 [“为上传准备 ESP32”](#)

安装 Arduino IDE 时，程序会在“文档”中新建一个名为 **Arduino** 的目录。默认路径为 C:\User\Documents\Arduino

需要准备好这个目录。

安装时，Arduino IDE 可能会在 ...Documents\Arduino 中放入子目录，其中包括名为 libraries 和 tools 的目录。需要移走这些目录。

注意，这很重要！

要正确安装 Samovar 草图，必须使用每个 Samovar 版本压缩包中附带的库和工具。使用早期版本或其他来源的库，会导致 Samovar 草图上传到 ESP32 时出错。

下面的建议可以帮助您在开始时避免因粗心等原因出错。以后您可以形成自己的方法。

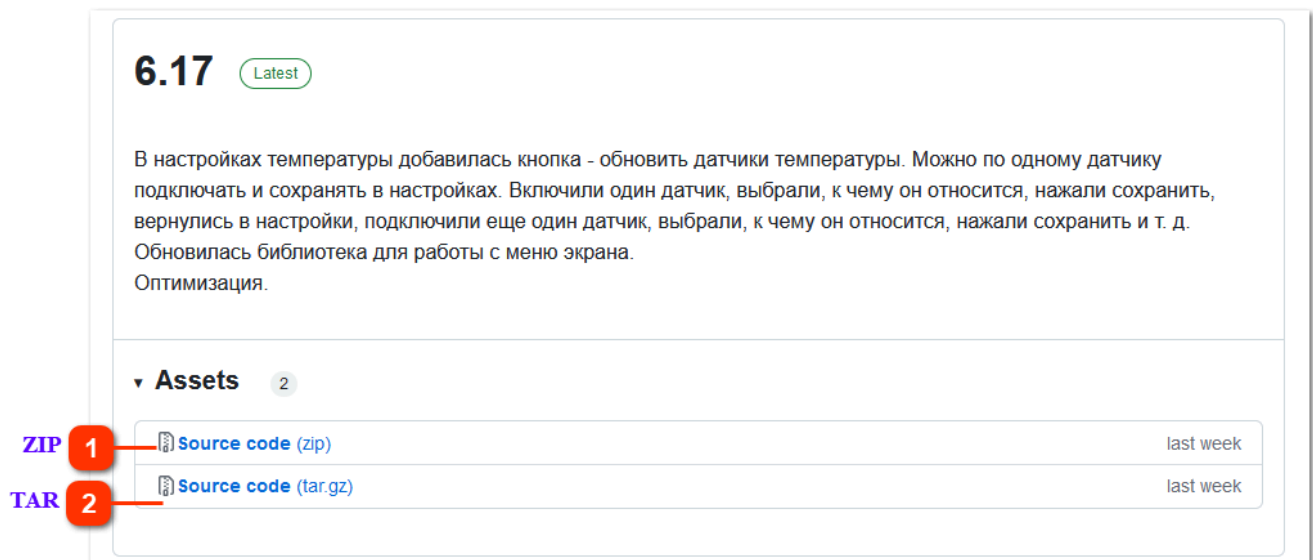
在 ...Documents\Arduino 中新建子目录 OLD，把 Arduino IDE 安装时创建的 libraries 和 tools 目录拖入（或剪切后粘贴到）OLD 目录。

如果操作正确，C:\User\Documents\Arduino\ 中不应再有名为 libraries 和 tools 的目录。其他任何目录名都可以。

现在通过这个[链接](#)下载最新固件的压缩包。

注意！

建议从上面的链接下载压缩包。那是完成并经过测试的版本。该资源上还有[Master 版本](#)。这是正在开发中的未来版本。它可能工作不正常，甚至完全无法工作。



1 ZIP
下载 ZIP 格式的压缩包

2 TAR
下载 TAR 格式的压缩包（RAR、GZ）

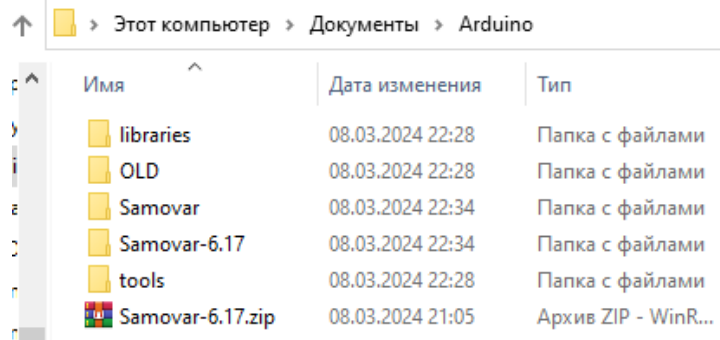
通常，压缩包会保存到浏览器的“下载”目录。

下载的压缩包名为：Samovar-X.XX，其中 X.XX 对应版本号（**下面所有示例都使用 Samovar-7.00 版**）。

把下载的压缩包复制（或剪切后粘贴）到目录 C:\User\Documents\Arduino\（您的路径可能不同）

- 解压缩包。会出现目录 Samovar-7.00。
- 进入该目录——里面还有一个目录 Samovar-7.00。把它重命名为 Samovar。
- 把这个 Samovar 目录拖动（或剪切 - 粘贴）到上一级，即 C:\User\Documents\Arduino\
- 进入 Samovar 目录。里面有目录和文件。把 libraries 和 tools 两个目录拖动到上一级，即 C:\User\Documents\Arduino\

最终，...Documents\Arduino\ 中应形成如下目录结构：



- libraries——库
- OLD——旧文件
- Samovar——Samovar 草图
- Samovar-7.00——空目录（不要删除，后面还会用到）
- tools ——工具
- Samovar-7.00.zip——原始草图的压缩包。

编译草图时，Arduino IDE 会使用以下目录：Samovar、libraries 和 tools。

现在草图已准备好上传到 ESP32。

- 把 ESP32 连接到 USB 端口。
- 启动 Arduino IDE。
- 在 Arduino IDE 中打开文件 **Samovar.ino**（位于目录 ...\Documents\Arduino\Samovar\）。
- 按照准备 ESP32 时 [上传标准示例](#) 的同样方法，把草图上传到 ESP32。



点击箭头开始把版本上传到 Samovar

说明。

- 编译代码可能需要很长时间——不用担心，请耐心等待。
- 上传监视器中红色高亮的行可以忽略——这是正常的。如果发生严重错误，上传会停止，并出现带有错误消息的橙色条。

注意！

某些 ESP32 开发板的 仿制品不按 **BOOT** 按钮就无法把草图上传到 ESP。

使用 **BOOT** 按钮上传有两种方法。

- 如果在 ESP 通电前按住 **BOOT** 按钮，通电后继续按住 3 秒再松开，就可以烧录 ESP，并且之前所做的所有设置都会保留，特别是 WiFi 连接设置。
- 如果直接通电并发出“上传”命令，在草图编译完成、开始上传到 ESP32 之后，Arduino IDE 控制台 在几行红色文字后会显示消息：

"Connecting" 之后开始出现点。

这时按下 ESP32 开发板上的 **BOOT** 按钮，点不再出现时松开。

按 **BOOT** 只是让开发板进入引导模式，本身不会清除 Wi-Fi 设置。

安装网页界面。

注意！

从 6.0 版开始，所用的文件系统改变了。更换文件系统是为了优化 Samovar 的速度。

安装草图后，必须在本地安装网页界面。

不要断开 ESP32 与电脑的连接，在“工具”菜单中选择：

- ESP32 LittleFS Data Upload - 用于 6.0 及更高版本
- ESP32 Sketch Data Upload - 用于低于 6.0 的版本。

再次点击 Arduino IDE 中的“上传”按钮，等待安装完成。在某些 ESP32 版本上，上传界面时也需要使用 **BOOT** 按钮（见上文）。

注意！重要！

最近出现了一些工作完全正常的 ESP32 开发板 仿制品，但如果在“工具”菜单 > 开发板 > ESP32 Arduino > **DOIT ESP32 DEVKIT V1** 中选择，网页界面无法安装。如果选择 **ESP32 Dev Module** 开发板，界面可以正常上传。

注意！重要！

从 6.0 版开始，只要 Samovar 连接了互联网，界面就会自动下载并安装，无需您参与。

如果需要回退到旧版本的草图（适用于 6.0 及更高版本），界面也会自行回退到对应的版本。

如果没有互联网连接，则需要为每个版本自行安装界面，也就是在本地安装。

固件会把服务器上的界面版本（web.samovar-tool.ru 上其自身版本的目录）与本地文件 version.txt 进行比较，只有不同时才下载文件。这时串口监视器会输出 WEB interface version 和 Local interface version 两行。如果 LittleFS 的剩余空间少于 64 KB，更新会被取消，并显示消息“网页界面更新已取消：磁盘空间不足”（«Обновление веб-интерфейса отменено: мало места на диске»）。如果文件系统未能挂载（新开发板、已擦除的闪存），固件会自行将其格式化，显示消息“无法连接到文件系统，正在格式化...”（«Не удалось подключиться к файловой системе, форматируем...»）。

补充。

但事实证明，情况有所变化，这并非对所有人都有效。

从 6.20 版开始，文件 Samovar_ini.h 中增加了一个参数：

```
##define NOT_USE_INTERFACE_UPDATE //启动时不使用界面更新
```

首次上传草图时，需要取消它的注释：

```
#define NOT_USE_INTERFACE_UPDATE //启动时不使用界面更新
```

这样界面就不会通过互联网更新，只能手动更新。以后在设置草图时，可以把这一行恢复原样（注释掉），看看通过互联网更新是否正常。

提示

建议通过 USB 在本地进行界面的首次安装。原因是，如果启用了通过互联网更新界面的选项，下载和更新界面可能需要很长时间，在此期间界面不会显示。

警告。

本地上传 LittleFS 支持 Arduino IDE 1.8.x（使用经典上传工具）和 Arduino IDE 2.x（使用 ESP32 LittleFS Data Upload 插件）。

补充。

在 ESP32 by Espressif Systems 开发板包 3.0.0 及更高版本中，可以用 Arduino IDE 2 的插件在本地安装界面。说明见[这里](#)。插件本身（文件 arduino-littlefs-upload-1.1.7.vsix）在 Samovar 压缩包的工具\ESP32LittleFS_Arduino IDE 2.X 目录中：把它复制到目录 ~\.arduino\IDE\plugins\（Windows 上为 C:\Users\<name>\.arduino\IDE\plugins\），重启 Arduino IDE，按 Ctrl+Shift+P 并选择“Upload LittleFS to Pico/ESP8266/ESP32”。

说明。

在某些 ESP32 版本上，使用 ESP32 LittleFS Data Upload 烧录 6.0 及更高版本时，界面仍然无法安装。原因至今是个谜。遇到这种情况，请交替多次尝试，分别用 ESP32 LittleFS Data Upload 和 ESP32 Sketch Data Upload 烧录 ESP32。每次烧录后检查界面是否出现。

上传草图和网页界面后，断电重启 Samovar，然后 [把 Samovar 连接到您的 Wi-Fi 网络。](#)

Samovar 连接到您的 Wi-Fi 网络后，在浏览器的地址栏中输入分配给它的 IP 地址或名称 **http://samovar.local**，打开 Samovar 界面。

推荐的默认值无需手动加载。

Samovar 7.00: 如果还没有保存的配置，固件会自行创建并保存默认设置。tools 目录中的文件 **Samovar_default_settings.txt** 可用于手动恢复，但首次启动时不需要。

要手动恢复之前保存的配置，打开“设置 → 其他”（«Настройки → Прочие»），点击“上传设置文件”（«Загрузить файл настроек»），选择文件，点击“保存”（«Сохранить»）并重启 Samovar。可以用同一选项卡上的“下载设置文件”（«Скачать файл настроек»）按钮保存自己的配置。

说明。

如果草图或 Samovar 界面上传出错，需要[擦除](#) ESP32 模块的闪存，然后重新上传。

Samovar 的初始设置完成——可以根据自己的设备[设置文件 Samovar_ini.h](#)了。

连接 Wi-Fi

如果在配置器中填写了 SSID 和密码（“网络”（«Сеть»）部分的“Wi-Fi SSID”（«SSID Wi-Fi»）和“Wi-Fi 密码”（«Пароль Wi-Fi»）字段），并且开发板内存中还没有保存其他网络，Samovar 会先自动尝试连接这个网络。此时串口监视器中显示 **WiFi: initial credentials loaded from local configuration** 一行。如果之前已保存过网络，则使用已保存的数据。

如果连接失败，ESP32 会创建 **Samovar** 接入点。用手机或电脑连接它。如果门户页面没有自动打开，在浏览器中打开 <http://192.168.4.1>，选择家里的网络并输入密码。同一页面上有 **blynk token** 字段 —— 可以直接在这里输入 Samovar 手机应用中的设备令牌（32 个字符），它会保存到 Samovar 的设置中。也可以留空，稍后在“设置”（«Настройки»）页面输入令牌。

输入数据的时间为 **6 分钟**。如果没有完成设置，设备将继续作为独立的 Samovar 接入点运行，密码为 **SamApp123**；界面地址为 <http://192.168.4.1>。

连接成功后，可以在显示屏上、串口监视器中（**IP address:** 一行）、通过配置器“串口监视器”（«Монитор порта»）窗口中的“**获取 IP**”（«Получить IP»）按钮、在手机应用中或在路由器的客户端列表中查看 IP 地址。在同一局域网内设备的浏览器中打开这个 IP 地址。也可以用 <http://samovar.local> 这个名称代替地址（mDNS；并非所有网络和设备都支持）。

要清除已保存的网络，在启动时按住编码器按钮至少两秒。短按不应清除 Wi-Fi。其他清除方式：

通电后立即按住 ESP32 开发板上的 **BOOT** 按钮（GPIO0）至少两秒 —— 串口监视器中显示 **GPIO0 held low: erasing WiFi settings**；

在 LCD 上选择 **Reset WiFi** 菜单项；

在网页界面中打开“设置 → 其他”（«Настройки → Прочие»），点击“**重置 Wi-Fi 设置**”（«Сбросить настройки WiFi»）按钮。

清除后，Samovar 重启，并再次开启 Samovar 接入点。

检查安装是否正确

这项检查确认三件事：固件已启动，ESP32 已连接网络，网页界面可以打开。

1. 打开串口监视器

- . 用可传输数据的 USB 线把 ESP32 连接到电脑。
- . 在配置器中按“**串口监视器**”（«**Монитор порта**»）：日志会在单独的窗口中打开。在 Arduino IDE 中打开 **Tools** → **Serial Monitor**。在 PlatformIO 中打开 **PROJECT TASKS** → **所选环境** → **General** → **Monitor**。
- . 把速率设为 **115200** 波特——这是与开发板通信的速率。
- . 按 ESP32 上的 RESET/EN，或者短暂拔出再插上 USB。

2. 查看连接结果

连接家庭网络时，会出现 **Connected to ...** 这一行，随后是 **IP address: ...**。记下冒号后面的地址。如果无法连接已保存的网络，Samovar 会开启 Samovar 热点及设置门户，持续 6 分钟（见[连接 Wi-Fi](#)）。

如果 Samovar 以自身热点方式启动，会出现 **Started as WiFi AP** 以及地址 **192.168.4.1**。连接到 Samovar 网络并打开这个地址。

如果同样的启动信息反复出现，说明 ESP32 在不断重启。这不是正常工作状态：请保存监视器中从第一行启动信息到重启的全部文本。

3. 打开网页界面

- . 电脑或手机必须与 Samovar 连接在同一个家庭网络中，或者直接连接到 Samovar 热点。
 - . 在浏览器地址栏中输入 **http://** 和 IP 地址，例如 **http://192.168.2.100**。必须使用 HTTP，不带字母 s。
 - . 应打开 Samovar 页面，显示版本、温度和**设置**（«**Настройки**»）按钮。按“设置”，确认页面能够打开。
- 如果打开的不是界面，而是路由器或其他设备的页面，说明 IP 地址不对。请重新查找地址：在串口监视器中、在显示屏上、在配置器“串口监视器”窗口中用“**获取 IP**”（«**Получить IP**»）按钮，或者在路由器的客户端列表中。在同一网络中也可以打开 **http://samovar.local**。
- 网页界面由 Samovar 自己根据已安装的固件版本从服务器更新；在串口监视器中对应 **WEB interface version** 和 **Local interface version** 这两行。如果没有网络连接，会出现 **WEB interface update skipped: no WiFi connection**。首次检查时，不要关闭更新机制，也不要手动替换单个文件。

草图设置*

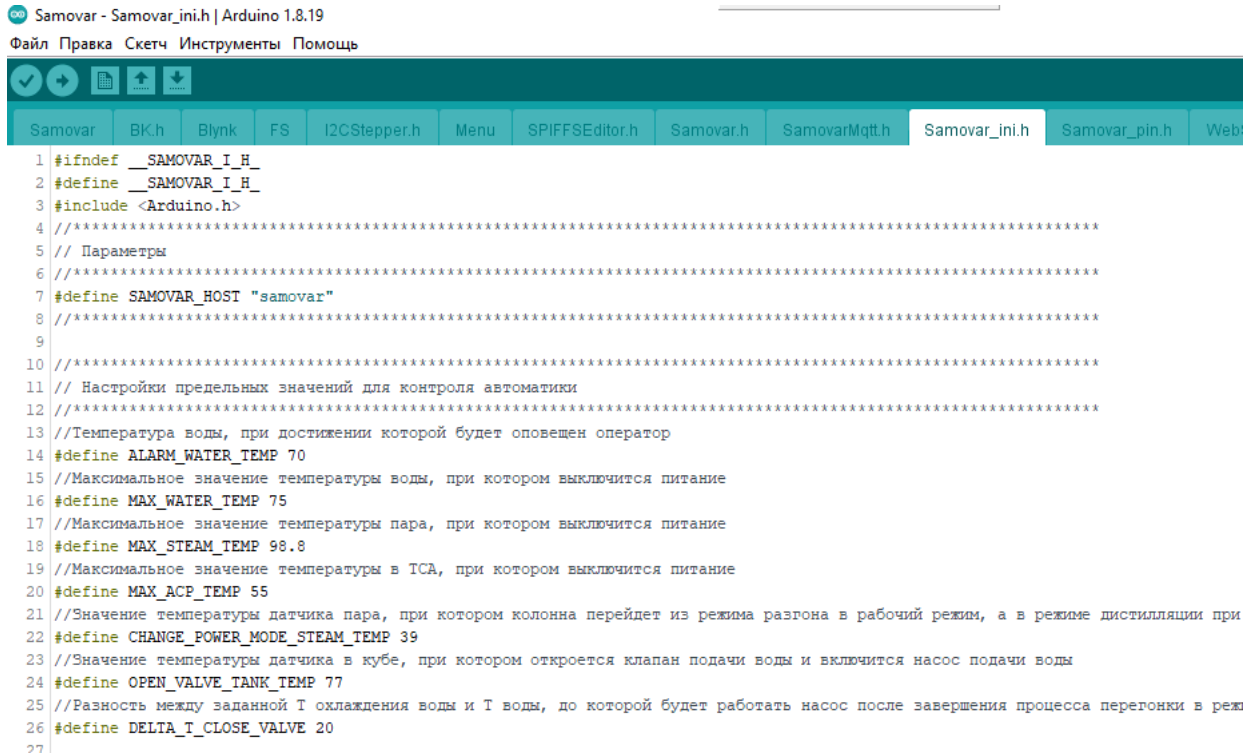
Samovar 7.00: 参数既可以按下文介绍的方法在 Arduino IDE 中设置，也可以通过配置器设置。配置器把所选的值保存在 user_config_override.h 中，不改动主文件 Samovar_ini.h。固件会自动包含这个文件，其中的值优先于 Samovar_ini.h；也可以按 user_config_override.example.h 模板手工创建它。在配置器中，参数按分区排列（“分区”（«Раздел»）列表）：基本（«Основные»）、温度（«Температуры»）、泵（«Насосы»）、NBK（«НБК»）、调节器（«Регулятор»）、设备（«Оборудование»）、精馏（«Ректификация»）、BK（«БК»）、网络（«Сеть»）、传感器（«Датчики»）、步进电机（«Шаговый двигатель»）。

要让草图适配您的设备配置，请在 Arduino IDE 中打开它，并修改 Samovar_ini.h 文件

在 Arduino IDE 中打开 **Samovar.ino** 文件，它位于文件夹 ...Documents\Arduino\Samovar\ 中。

这样，...\Documents\Arduino\Samovar\ 文件夹中的所有文件都会加载到 Arduino IDE 中

在 Arduino IDE 顶部一行找到 Samovar_ini.h 选项卡，开始编辑。



```
1 #ifndef __SAMOVAR_I_H_
2 #define __SAMOVAR_I_H_
3 #include <Arduino.h>
4 //*****
5 // Параметры
6 //*****
7 #define SAMOVAR_HOST "samovar"
8 //*****
9
10 //*****
11 // Настройки предельных значений для контроля автоматики
12 //*****
13 // Температура воды, при достижении которой будет оповещен оператор
14 #define ALARM_WATER_TEMP 70
15 // Максимальное значение температуры воды, при котором выключится питание
16 #define MAX_WATER_TEMP 75
17 // Максимальное значение температуры пара, при котором выключится питание
18 #define MAX_STEAM_TEMP 98.8
19 // Максимальное значение температуры в ТСА, при котором выключится питание
20 #define MAX_ACP_TEMP 55
21 // Значение температуры датчика пара, при котором колонна перейдет из режима разгона в рабочий режим, а в режиме дистилляции при
22 #define CHANGE_POWER_MODE_STEAM_TEMP 39
23 // Значение температуры датчика в кубе, при котором откроется клапан подачи воды и включится насос подачи воды
24 #define OPEN_VALVE_TANK_TEMP 77
25 // Разность между заданной Т охлаждения воды и Т воды, до которой будет работать насос после завершения процесса перегонки в режи
26 #define DELTA_T_CLOSE_VALVE 20
27
```

要让某行代码不被编译、从而不被 Samovar 执行，请在该行前加上 // 字符（注释掉）。

反之，要让某行生效，请删除 // 字符（取消注释）

建议只排除与您没有的设备相关的行，而不要把您认为用不到的参数一股脑全部关掉。

为方便起见，文件中的代码分成若干块。

#define SAMOVAR_HOST "samovar"

Samovar 在网络中的名称。设备可通过 http://samovar.local (mDNS) 访问；通过 WiFi 更新时，Arduino IDE 和配置器的网络端口中也显示这个名称。在配置器中：分区“基本”（«Основные»），“设备网络名称”（«Имя устройства в сети»）。

代码块

//选择所用的 ESP32 开发板

//*****

##define BOARD DEVKIT

##define BOARD LILYGO

##define BOARD ESP32S3

//*****

这一块可以跳过。固件会根据 Arduino IDE 中选择的开发板（DOIT ESP32 DEVKIT V1 / ESP32 Dev Module - DEVKIT, ESP32S3 Dev Module - ESP32S3）或配置器中选择的开发板自动确定开发板类型。如果什么都没识别出来，则使用 DEVKIT。

从这一行开始，可以修改数值并启用/禁用代码行：

//向操作者报警的水温

#define ALARM_WATER_TEMP 70

这里的 70 是温度，单位为摄氏度。达到该温度时，会发出冷却液温度（下文简称 CT）达到临界值的消息。如果使用了蜂鸣器，并且在界面中启用了事件提醒，Samovar 还会发出声音报警。在温度降下来之前，报警会一直持续。如果希望在其他温度报警，请修改数字 70。

//切断电源的最高水温

#define MAX_WATER_TEMP 75

CT 达到 75° 时，Samovar 发出关闭加热的命令。

//切断电源的最高蒸汽温度

#define MAX_STEAM_TEMP 98.8

蒸汽传感器温度达到 98.8° 时，Samovar 发出关闭加热的命令。

//切断电源的 TCA 最高温度

#define MAX_ACP_TEMP 75

TCA 传感器温度达到 75° 时，Samovar 发出关闭加热的命令。（另见[TCA 传感器设定值](#)）

//蒸汽传感器达到此温度时，塔从升温模式切换到工作模式；在蒸馏模式下，如装有蒸汽传感器，则以此判断开始沸腾

#define CHANGE_POWER_MODE_STEAM_TEMP 39

蒸汽传感器温度（或塔节上部传感器的温度）达到 39° 时，Samovar 向调节器发出“停止升温”命令，并在调节器上设置取酒程序第一行中指定的电压。在蒸馏模式下，Samovar 也用这一蒸汽传感器温度判断开始沸腾。在配置器中：“温度”（«Температуры»），“退出升温的温度，°C”（«Температура перехода из разгона, °C»）。

//蒸馏釜传感器达到此温度时，打开供水阀门并启动供水泵

#define OPEN_VALVE_TANK_TEMP 77

蒸馏釜温度达到 77° 时，发出打开供水阀门的命令，同时发送启动泵的 PWM 控制信号，并向操作者发送需要开水的消息。

无论是否使用泵或阀门、水是否已经打开，这条消息都会发送。

#define DELTA_T_CLOSE_VALVE 20

//设定的冷却水温度与实际水温之差：在蒸馏和精馏模式下，蒸馏过程结束后，泵会一直运转，直到水温降到这个差值。数值单位为 C°，可以修改。强烈不建议禁用。（详见[这里](#)）

//泵的 PWM 调节下限值（百分比）。请选择能保证水流的数值

#define PWM_LOW_VALUE 10

众所周知，电机启动所需的电压比维持运转所需的电压高。这一行设置供水泵电机能够持续运转而不停转的最小值（百分比）。

//泵的 PWM 调节开始工作的下限值（百分比）。请选择能保证电机启动的数值

#define PWM_START_VALUE 40

能保证泵电机启动的电压，以额定电压的 % 表示。

也就是说，如果您的泵（已装入系统且其中有液体）在施加 6 伏及以上电压时能可靠启动，低于此电压则不启动，那么在额定电压 12 伏时，这相当于额定值的 50%。因此应把数字 40 改为 50。

//舵机转角修正。

int8_t servoDelta[11] = {0, -2, -3, -4, -3, -2, 0, 0, 0, 0, -2};

修正取酒转盘舵机的转角，以便精确定位。单位为度。每个位置单独修正，与相邻位置的修正无关。

例如，如果摇臂“越过”了 2 号位置，我们设置了负 10°，那么修正 3 号位置时不必考虑这个值（-10）。

//目标温度与当前温度之差（用于啤酒和低温慢煮模式），在差值大于此值时以全功率加热（升温模式）。

#define HEAT_DELTA 1

用于啤酒和低温慢煮模式。数值单位为摄氏度。该参数针对主加热器。含义是：只要实际温度与糖化程序行中设定温度之差大于 1°，就以全功率加热。一旦差值小于或等于 #define HEAT_DELTA 中指定的值，就让调节器减小加热，以便平稳达到设定温度和/或不明显超过它。

如果加热器功率较大，请增大该值，以免“冲过”目标温度。

//如果目标温度与当前温度之差（用于啤酒和低温慢煮模式）大于设定值，在使用 UART 控制的调节器时，4 号继电器会额外接通辅助加热管；差值较小时，辅助加热管关闭

#define ACCELERATION_HEATER_DELTA 4

用于啤酒和低温慢煮模式。数值单位为摄氏度。该参数只针对附加加热器。

如果蒸馏釜温度低于目标温度（此处为低 4°），附加加热器接通。

//沸腾温度（用于啤酒模式）

#define BOILING_TEMP 98.9

用于啤酒和低温慢煮模式。数值单位为摄氏度。该值需手动设置，告诉 Samovar 在您的设备上蒸馏釜中真正开始沸腾的温度。该值[影响](#)主加热器和附加加热器的工作算法

//默认的蒸馏结束温度

#define DEFAULT_DIST_TEMP 99.9

用于蒸馏模式。如果蒸馏程序没有设定其他值，蒸馏釜达到此温度时结束运行。在配置器中：“温度”（«Температуры»），“蒸馏结束温度，°C”（«Температура завершения дистилляции, °C»）。

Samovar 子系统的使用设置

#define SAMOVAR_USE_BLYNK

//在项目中使用 Blynk

如果不使用 Blynk，必须用 // 注释掉这一行。

#define BLYNK_SAMOVAR_TOOL "samovar-tool.ru" //使用免费的 Blynk 服务器 samovar-tool.ru，而不是 Blynk 云

如果您使用付费的 Blynk 服务器，而不是开发者提供的服务器，必须注释掉这一行。

`##define USE_MQTT` //把读数发送到您自己的 MQTT 代理

把日志保存到云端不需要这个参数：只要在设置中填写了设备令牌并启用了 SAMOVAR_USE_BLYNK, Samovar 的日志和消息就会自动发送到 samovar-tool.ru 服务器。请在 www.samovar-tool.ru 网站注册，并在编辑个人资料时填入设备令牌；之后即可在网站上通过菜单“用户 - 日志”查看日志。详见章节 - [云端日志和图表](#)。

USE_MQTT 是一项独立的可选功能，用于把简化的读数行发送到任意 MQTT 代理（不使用 TLS，运行过程中每 4 秒一次，空闲时每 5 秒一次）。Samovar_ini.h 中没有这一行：该参数在配置器中启用（分区“网络”（«Сеть»）：“MQTT 服务器”（«Сервер MQTT»）、“MQTT 端口”（«Порт MQTT»）、“MQTT 用户”（«Пользователь MQTT»）、“MQTT 密码”（«Пароль MQTT»）、“MQTT 主题”（«Топик MQTT»）），或在 user_config_override.h 中启用，此时必须设置 MQTT_SERVER、MQTT_PORT、MQTT_USER、MQTT_PASSWORD 和 MQTT_TOPIC；缺少它们时，带 USE_MQTT 的编译会报错中止。

选择功率调节器

在 Samovar_ini.h 文件中，用以下普通代码行进行选择：

`##define SAMOVAR_USE_POWER`

`##define KVIC_USE_9600`

`##define SAMOVAR_USE_RMVK`

`##define SAMOVAR_USE_SEM_AVR`

使用 Arduino IDE 编译时，这些行应保持如下状态：

KVIC：什么都不用改。原始文件中 **SAMOVAR_USE_POWER** 一行已经启用，RMVK 和 SEM_AVR 两行已被注释掉。

RMVK：只取消 **SAMOVAR_USE_RMVK** 一行的注释。固件会自动启用所需的通用调节器支持。

SEM_AVR：只取消 **SAMOVAR_USE_SEM_AVR** 一行的注释。固件会自动启用 RMVK 协议支持和通用调节器支持。

不使用 UART 控制的调节器：注释掉 **SAMOVAR_USE_POWER**、**SAMOVAR_USE_RMVK** 和 **SAMOVAR_USE_SEM_AVR**。

KVIC_USE_9600 参数只适用于 KVIC：调节器 UART 速率为 9600 时取消其注释。如果调节器使用 38400，则保持该行被注释。

使用 PlatformIO 编译时，不需要手动修改这些行：选择现成的环境 **Samovar**、**Samovar_no_power**、**Samovar_rmvk** 或 **Samovar_sem** 即可（另有益于 ESP32-S3 开发板的 **Samovar_s3**，以及 **Samovar_lua** 和 **Samovar_alarm_button**）。配置器会根据“功率调节器”（«Регулятор мощности»）组（不使用（«Не использовать»）/ KVIC / RMVK / SEM_AVR）和开发板自动选择环境。详见说明[使用 PlatformIO 编译和烧录](#)。

设备的详细介绍见章节[所用的调压器](#)。

NBK 参数

`##define NBK_MULT_PAUSE_OVERFLOW 2` // 液泛后暂停的时长，以惯性时间的倍数计

`##define NBK_PUMP_LIMIT 30` // “优化”时发酵泵的最大流量，l/h

`##define NBK_WORK_PRESSURE_RATIO 0.5f` // 工作压力上限 = 液泛压力的比例

`##define NBK_PRESSURE_MARGIN 5` // 学习上限时，比液泛时刻压力低的毫米汞柱数

`##define USE_NBK_DELTA_PRESSURE` // 根据酒糟排放口的压力修正酒糟温度

`##define USE_NBK_END_BY_STEAM_RISE` // 蒸汽温度比最低值升高 NBK_END_STEAM_RISE 时结束 NBK 运行

`##define NBK_END_STEAM_RISE 5.0f` // 用于 USE_NBK_END_BY_STEAM_RISE 的蒸汽温度升高阈值（°C）

只在连续醪塔模式下需要；在配置器中：分区“NBK”（«НБК»）。如果不使用 NBK，保持原样即可。详见章节[NBK](#)。

`##define SAMOVAR_USE_POWER_START_TIME 2000` //向调节器发送升温命令前的延时。保证接触器“冷”接通功率电路。

用于避免接触器主触点产生电弧。发出“开启加热”（«Включить нагрев»）命令后，1 号继电器动作并接通接触器，接触器再接通加热器的功率电路。经过以毫秒设定的时间后，调节器才输出电压。

关闭时顺序相反：先由调节器切断电压，延时后接触器再断开。

（参见[负载接通/断开算法](#)）

`##define LCD_RESET_PERIOD_MS 240000UL` //LCD 显示屏定期重新初始化的周期（ms），4 分钟

防止 I2C 显示屏“卡死”。无需修改。在配置器中：“设备”（«Оборудование»），“显示屏复位周期，ms”（«Период сброса дисплея, ms»）。

`##define USE_WATERSENSOR` //使用冷却水流量传感器。启用后，如果在设定时间内没有水流，Samovar 会关闭塔的加热

流量传感器基于霍尔传感器原理工作。有冷却液流动时，Samovar 会收到周期性的脉冲。如果没有脉冲或脉冲太少，Samovar 会关闭加热并发出报警。

`##define WF_CALIBRATION 98` //流量传感器系数 $F=98 \cdot Q(L/min)$ ；其他传感器请设置相应的值。

流量传感器修正。让 1 升水流过传感器，会收到 5880 个脉冲。实际上，不同水流速度下每升的脉冲数不同。可以通过修改数值 98 进行修正。

`##define WATER_FLOW_MIN_PULSES 7` //每个周期的脉冲数不少于此阈值时，才认为确实有水流

关水时滤除流量传感器的零星脉冲。如果 Samovar 在没有水流时认为有水流，请增大该值；如果水流较弱时它关闭了加热，请减小该值。在配置器中：“泵”（«Насосы»），“最少流量脉冲数”（«Минимум импульсов потока»）。

`##define USE_HEAD_LEVEL_SENSOR` //使用塔头中的回流液位传感器（P-N-P 型传感器）。启用后，传感器触发时功率降低 $1 \cdot PWR_FACTOR$ （KVIC：1 V，RMVK：2 V；SEM_AVR：当前值的 3%）

使用触发时输出逻辑 1 的回流液位传感器。

`##define IGNORE_HEAD_LEVEL_SENSOR_SETTING` //如果使用回流传感器，则忽略 WEB 界面中禁用传感器的命令。（如果注释掉这一行，界面的“设置”（«Настройки»）选项卡上会出现一个复选框，可用它启用或禁用回流传感器。）

示例。

如果这一行

`##define USE_HEAD_LEVEL_SENSOR` // 使用回流液位传感器.....被注释掉，则不使用传感器，下一行处于什么状态（注释或未注释）都无所谓。

如果 **`##define USE_HEAD_LEVEL_SENSOR`** 这一行未被注释，则要考虑下一行的状态：

#define IGNORE_HEAD_LEVEL_SENSOR_SETTING - 如果它未被注释，WEB 界面中就没有控制回流传感器的复选框。因此无法从 WEB 界面禁用传感器。

如果 **#define IGNORE_HEAD_LEVEL_SENSOR_SETTING** 被注释掉，则可以从 WEB 界面快速 [启用和关闭 \(第 5 项\)](#) 回流液位传感器的使用。

//define WHLS_HIGH_PULL //使用 N-P-N 型液位传感器

使用触发时输出逻辑 0 的回流液位传感器。

#define USE_WATER_PUMP //对水泵使用 PWM 控制，和/或在啤酒模式下用于接通搅拌器

如果不使用自动维持 CT 的泵，也没有在啤酒模式下连接搅拌器，必须注释掉这一行。在配置器中：“泵”（«Насосы»），“控制水泵或搅拌器”（«Управлять насосом воды или мешалкой»）。

//define USE_WATER_VALVE LOW //用阀门控制来调节水压

使用旁通阀时，必须取消这一行的注释。**阀门接在泵的位置上。**它按恒温器原理工作：温度高于设定值时阀门打开，低于设定值时关闭。详见[这里](#)。

如果旁通管路上使用常闭阀门，必须把 LOW 改为 HIGH。

如果 CT 由手动或其他第三方设备调节，必须注释掉这两行。

//define USE_ALARM_BTN //使用紧急按钮。

按下它会停止供水并切断电源。可以安装漏水传感器（允许并联任意数量带“干接点”的装置），也可以使用普通按钮（它也可以与传感器并联）

详见章节[紧急按钮_附加紧急传感器](#)

#define USE_BODY_TEMP_AUTOSSET //对酒头之后第一个酒心或预液泛取酒程序自动修正酒心温度；如果预液泛程序位于倒数第二个酒心或预液泛取酒程序之前，也对其修正

在标准的 Samovar 7.00 版本中，该功能已启用，且无法注释掉：固件会强制启用 USE_BODY_TEMP_AUTOSSET (Samovar.h)。在配置器中位于“精馏”（«Ректификация»）分区（“酒心温度自动修正”（«Автокоррекция температуры тела»）），升温上限 BODY_TEMP_AUTOSSET_MAX_RISE 也在那里（见下文）。

在以下条件下，取酒不会暂停：

对于**酒心（«Тело»）**行：酒心行紧接在酒头行之后，且其后有两行酒心或预液泛行（任意组合）。

对于**预液泛（«Предзахлеб»）**行：预液泛行位于取酒程序中的任意位置，且其后有两行酒心或预液泛行（任意组合）

//选择所用的压力传感器，只保留一行，其余行必须注释掉。

//define USE_BMP180 //使用 BMP180/BMP085 压力传感器

//define USE_BMP280 //使用 BMP280 压力传感器

//define USE_BMP280_ALT //使用备用地址的 BMP280 压力传感器

//define USE_BME280 //使用 BME280 压力传感器

#define USE_BME680 //使用 BME680 压力传感器

选择所用的大气压传感器，只保留一行，其余行必须注释掉。如果不使用大气压传感器，可以注释掉所有行，但这样 Samovar 就不会根据大气压修正沸腾温度。

#define USE_BTN //使用按钮

使用 Samovar 上的万能按钮。它接在 ESP32 的 D39 引脚上。按钮很方便，但如果不用，最好禁用。

#define USE_UPDATE_OTA //使用 ArduinoOTA 通过 WiFi 更新。便于调试；运行时会给 ESP32 增加额外负担

如果用串口有线连接更新 ESP32 草图，可以注释掉这一行。这样就无法通过 WiFi 更新。使用 ESP32 核心 3.x 版本编译时，固件会自动禁用该参数。在配置器中：“网络”（«Сеть»），“允许通过 Wi-Fi 更新”（«Разрешить обновление по Wi-Fi»）。

//define USE_EXPANDER 0x20 //使用 PCF8575 端口扩展器

//define USE_ANALOG_EXPANDER 0x48 //使用 PCF8591 模拟端口扩展器

//define USE_LUA //使用 Lua 编写控制 Samovar 和端口扩展器的脚本

启用端口扩展器和 Lua 脚本语言。详见章节[功能扩展](#)

//选择塔内所用压力传感器的类型。只能使用一种。

//define USE_PRESSURE_XGZ 32 //使用 XGZP6897D 压力传感器（或同类产品）。数值为传感器参数表中的 K 参数

//define USE_PRESSURE_1WIRE {0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x55} //使用 1Wire 压力传感器

//define USE_PRESSURE_MPX //使用 MPX5010D 压力传感器

塔内压力传感器（不要与上面的大气压传感器混淆）。NBK 和监测塔内压力需要它。只取消一行的注释；如果取消了两行或三行，则使用 MPX5010D。没有传感器时，三行都保持注释。在配置器中：“塔内压力传感器”（«Датчик давления в колонне»）组。

//define USE_STEPPER_ACCELERATION //步进电机平滑启动和制动

启用取酒泵的平滑加速，停止时也会平滑减速。

//define STEPPER_REVERSE //反转步进电机的旋转方向

如果需要取酒泵电机（Nema17）向另一方向旋转，必须取消这一行的注释。

//define I2CStepperStepMl 16000 //I2CStepper 每 ml 的步数

仅在使用外部 I2CStepper 模块时需要。设置取酒泵的初始校准值（每毫升步数）。在配置器中：“步进电机”（«Шаговый двигатель»），“I2CStepper 每 ml 步数”（«Шагов на мл для I2CStepper»）。

```
//define COLUMN_WETTING //如果装有回流液位传感器，精馏开始时让塔进入预液泛，以润湿填料
//define WETTING_POWER 220 //润湿填料时调节器的电压/功率（用于 SEM_AVR）
```

精馏前润湿填料。只在使用回流液位传感器（USE_HEAD_LEVEL_SENSOR）时有效：升温结束后，Samovar 在调节器上保持程序第一行的电压（未设定时为 220 V；SEM_AVR 使用 WETTING_POWER），必要时逐步降到 80%，并等待回流液位传感器触发，最长 20 分钟。消息中显示：“开始润湿塔填料”（«Начало смачивания насадки колонны»）和“塔填料已成功润湿！”（«Насадка колонны успешно смочена!»）。在配置器中：“精馏”（«Ректификация»），“精馏前润湿填料”（«Смачивание насадки перед ректификацией»）和“填料润湿功率”（«Мощность смачивания насадки»）。

```
//define NOT_USE_INTERFACE_UPDATE //启动时不更新界面
```

Samovar 会比较本地安装的界面版本与远程可用的版本，如果不一致就进行更新。

这种方式可以让您在 Samovar 中保留自己开发的文件。本地更新会覆盖所有内容，而通过互联网更新只更新界面，脚本和设置会保留。

安装新版本时无需上传界面，它会自动更新。

但事实证明，情况有所变化，这一功能并非对所有人都有效。

如果取消该参数的注释，界面将不会通过互联网更新。

```
//define KVIC_DEBUG //把与 KVIC 调节器通信的调试信息输出到串口监视器
```

只在排查与调节器的通信故障时需要。在配置器中：“调节器”（«Регулятор»），“KVIC 调试信息”（«Отладочные сообщения KVIC»）。

精馏算法参数

```
#define PAUSE_RESUME_HYSTERESIS_DELTA 0.07f //按蒸汽/塔节温度进入暂停的阈值与退出阈值之间的回差（°C）
```

```
#define PROGRAM_ROW_STOP_PAUSE_LIMIT 3 //当前程序行内的停止暂停次数，达到后自动降低该行的基础速度
```

```
#define PROGRAM_ROW_STOP_PAUSE_SPEED_CUT_PCT 10 //PROGRAM_ROW_STOP_PAUSE_LIMIT 触发时，该行基础速度降低的百分比
```

```
#define PROGRAM_DONE_AUTO_POWEROFF_MIN 30 //保持“程序执行完毕”（«Выполнение программы завершено»）状态的时间（分钟）（泵已停止，加热继续），之后自动关闭。0 = 立即关闭
```

```
#define BODY_TEMP_AUTOSET_MAX_RISE 0.3f //一个程序行内酒心温度自动升高的上限（°C）
```

默认值适用于常见的塔，无必要请勿修改。在配置器中：分区“精馏”（«Ректификация»）。

BK 模式参数

```
#define BK_STEAM_SETPOINT_MIN 30 //BK 分凝器蒸汽温度设定值的下限（程序行第 5 个字段），°C；0 - 手动控制水
```

```
#define BK_STEAM_SETPOINT_MAX 100 //BK 分凝器蒸汽温度设定值的上限，°C
```

```
#define BK_WATER_ADJUST_PERIOD_MS 60000 //BK 水量调节的步进周期，ms
```

```
#define BK_WATER_DEADBAND 0.2f //BK 蒸汽设定值附近的死区（°C）
```

```
#define BK_WATER_PWM_STEP 30 //BK 水量 PWM 调节的步长（满量程 1023）
```

馏塔模式下设定值的范围和自动调水的步长。在配置器中：分区“BK”（«БК»）。

编辑完文件后，必须重新把草图上传到 ESP32。为此，点击 Arduino IDE 顶部工具栏上的箭头，等待上传完成。



您的 Samovar 已可以投入使用。

备注。

今后安装新版本时，不必每次都编辑 Samovar_ini.h。

通过 Arduino IDE 更新时，请比较新旧两个 Samovar_ini.h，只把您自己的数值转移过去。不要用旧文件整体替换新文件：新版本可能有新的参数，缺少它们配置就不完整。配置器会把用户的改动单独保存在 user_config_override.h 中。

最好直接编辑新文件，而不是把新代码中的行添加到旧文件中，以免出现意外错误（参见“更新草图”）。

比较和编辑时，可以使用能够对比两个文件、直观显示差异的专用程序。

注意。

建议从 [此链接](#) 下载新版 Samovar 的压缩包。这是稳定且经过测试的版本。

* 面向开发者。草图各部分的说明和框图 见 github.com

1. 用户交互（网页和 LCD）

2. 工艺程序的执行

3. [系统状态和模式管理](#)
4. [传感器数据采集](#)
5. [硬件控制 \(执行机构\)](#)
6. [安全监控和报警](#)
7. [配置的保存](#)
8. [网络和外部通信](#)

其他文件和代码行的说明：

文件 *Samovar_pin.h*

Samovar 7.00: 该文件中已没有 TARGET_WATER_TEMP 常量。在所有模式下（包括 NBK），泵的 PWM 调节所维持的水温都取自[设置 - 温度 \(«Настройки - Температура»\)](#) 选项卡上的“水温传感器设定值” («Уставка датчика воды») 字段；该字段为空时，固件使用 48°。过程结束后，当出水温度比设定值低 DELTA_T_CLOSE_VALVE（默认 20°，见上文）时，Samovar 停止泵，发出关闭水阀的命令，并发出需要关水的提醒。

文件 *pumppwm.h*

`define PUMP_PWM_FREQ 15`

冷却泵的最低 PWM 频率，单位为赫兹

更新草图

Samovar 7.00: 支持通过 Arduino IDE 更新。另一种同样可行的方式是运行新版本的配置器并点击“烧录”（«Прошить»）；LittleFS 用单独的“上传 LittleFS”（«Загрузить LittleFS»）按钮上传，它会替换网页界面文件（LittleFS 中的用户文件可能因此丢失）。配置器把您的设置值保存在草图旁边的 `user_config_override.h` 文件中——把它复制到新版本的目录，设置就会保留。也可以通过 Wi-Fi 烧录：如果在“网络”（«Сеть»）部分启用了“允许通过 Wi-Fi 更新”（«Разрешить обновление по Wi-Fi»）（USE_UPDATE_OTA），“刷新”（«Обновить»）按钮会在网络中找到 Samovar。

随着新版本的发布，就需要更新。新版本可能附带新的或修改过的 库和工具。因此，为避免出错，建议每次都使用具体版本归档 中附带的库和工具。

注意！重要！

从 6.0 版开始，只要 Samovar 连接了互联网，界面就会自动更新为与该草图版本对应的版本；也就是说，如果需要回退到旧版草图，界面也会回退到相应版本。如果没有互联网连接，界面需要自己在本地安装。

更新草图

在[首次上传草图](#)一节中，我们在 `...Documents\Arduino` 中建立了一定的文件和目录结构。下面在原有结构的基础上继续。

按以下顺序整理 `... Documents\Arduino\` 目录：

- 把 `libraries` 和 `tools` 目录拖动（或剪切粘贴）到 Samovar 目录中。
- 把 Samovar 目录移到一个空的 Samovar-7.00 目录中。
- 把 Samovar-7.00 目录移到 OLD 目录中。
- 把装有原版草图的 Samovar-7.00.zip 归档移到 OLD 目录中。

这样 `... Documents\Arduino\` 目录中只剩下 OLD 目录。

OLD 里面则有：Samovar-7.00 归档，以及包含 Samovar 子目录的 Samovar-7.00 目录。

这种存放方式可以避免更新时出错，便于新版本沿用以前的设置，必要时还能快速回退到上一个或更早的版本。

接下来按首次安装时的步骤进行：把新版本“解压”到 `... Documents\Arduino\` 中，直到得到原来的[目录结构](#)。相应地，归档和从归档中解出的目录名称中会带有新的版本号。

但现在不要马上把草图上传到 ESP；先在 `Samovar_ini.h` 文件中完成必要的设置。有两种方法：

1. 正确的方法。

为方便起见，需要使用专门的程序，它能比较两个不同的文件，直观地显示它们之间的差异。

推荐使用 [WinMerge](#)——（Windows 下的比较与合并工具，既能比较文件也能比较文件夹，以直观的文本形式显示差异。该程序简单易用，并有俄文界面。）

或者使用功能更强、但也更难掌握的 程序 **Notepad++**，并安装 **Compare** 插件。

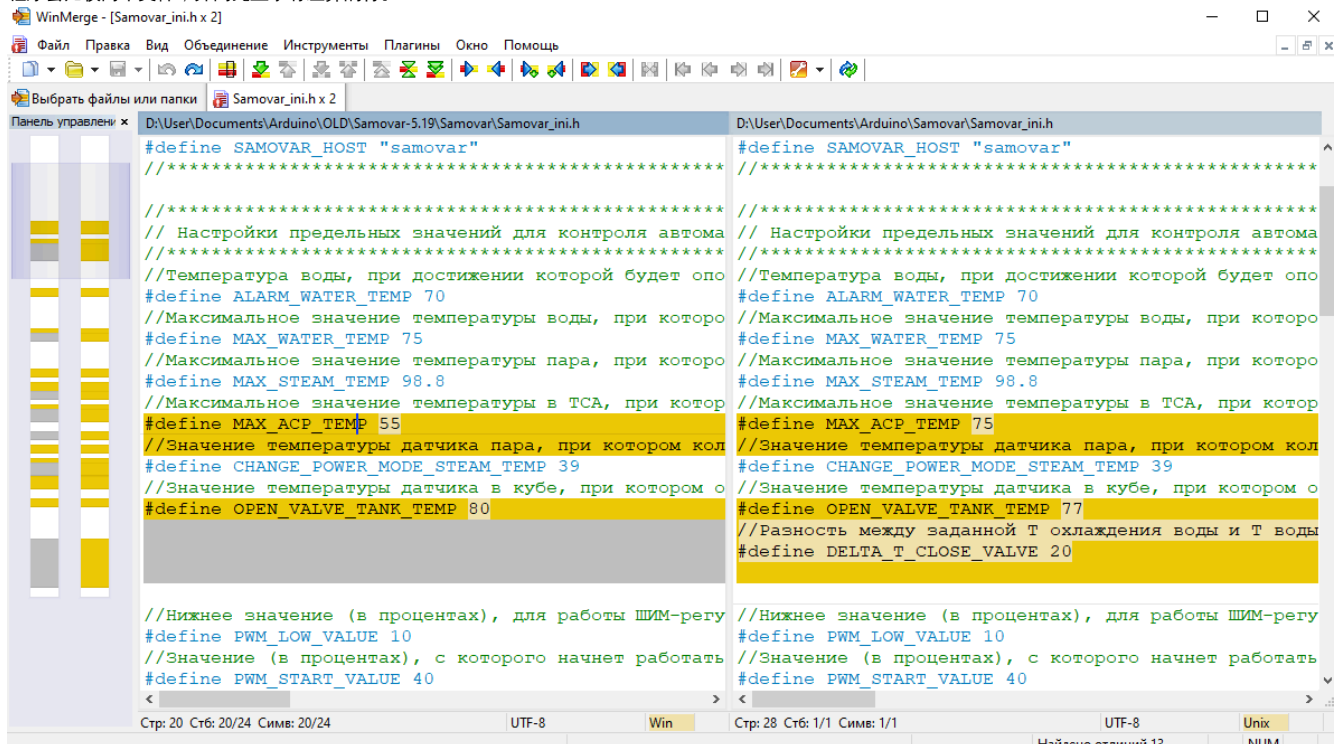
以下以 WinMerge 程序为例。

比较时需要选择两个不同的 `Samovar_ini.h` 文件

先选择 OLD 目录中的文件，我们刚把上一版 Samovar 移到那里。它位于 `... \Documents\Arduino\OLD\Samovar-7.00\Samovar\`

第二个文件取自新版 Samovar 的目录。它位于 `... \Documents\Arduino\Samovar\`

程序会比较两个文件，并高亮显示有差异的行。



差异会以较浅的颜色高亮显示。可以手动编辑右侧的数值，也可以把光标放在左侧，执行命令“复制到右侧”（«Копировать направо»）。这样逐行把设置文件调整为适合您的设备。全部修改完成后，点击“保存”图标。关键是不要搞混：左边是旧的设置文件，右边是 Samovar 将要使用的文件。

2. 快速的方法

通过 Arduino IDE 更新时，比较新旧两个版本的 Samovar_ini.h，只把自己的设置值搬过去。不要用旧的 Samovar_ini.h 覆盖新的：即使是小版本也可能包含新的必需参数。

如果新版本带来了 Samovar 的新功能，必须编辑 Samovar_ini.h 文件，否则新功能将不可用，Samovar 也无法正常工作。

如果设置是在配置器中完成的，则不需要编辑 Samovar_ini.h：您的设置值保存在 user_config_override.h 中，它会覆盖 Samovar_ini.h。只需把这个文件搬到新版本中即可。

设置完成。可以把草图上传到 ESP。由于这是对已安装草图的更新，不必重新设置 Wi-Fi 连接，也不需要加载默认设置。

更新时，可以不再把 ESP 连接到 USB 口，而是“无线”更新固件

“回退”到旧版本。

新版本中有些地方您不满意，或者工作不正常，想回到以前的版本。采用这种旧版本存放方式，回退非常简单。

像更新时那样整理 ... \Documents\Arduino 目录：

- 把 libraries 和 tools 目录拖动（或剪切粘贴）到 Samovar 目录中。
- 把 Samovar 目录移到一个空的 Samovar-7.00 目录中。
- 把 Samovar-7.00 目录移到 OLD 目录中
- 把装有原版草图的 Samovar-7.00.zip 归档移到 OLD 目录中

然后：

- 进入 OLD 目录，选择以前的版本，进入其中的 Samovar 目录：... \Documents\Arduino\OLD\Samovar-6.16\

- 把整个 Samovar 目录 复制（建议复制而不是拖动）到 ... \Documents\Arduino\ 中

从 Samovar 目录中把 libraries 和 tools 两个目录拖动（或剪切粘贴）到上一级，即 ... \Documents\Arduino\ 中。

就这样。可以把草图上传到 ESP 了。

以后更新到新版本时，只需清空 ... \Documents\Arduino 目录，删除 Samovar、libraries 和 tools。不必把它们移到 OLD，因为它们已经在那里，我们用的是副本。

更新草图时已经可以 使用 Arduino IDE 2.x.x 版本。但这些版本没有“ESP32 LittleFS Data Upload”菜单项，没有插件就无法在本地更新界面。因此，如果 Samovar 无法访问互联网，网页界面将不会更新。

补充。 在 Arduino IDE 2.x.x 版本中，已经可以通过插件在本地安装界面。说明见[这里](#)。插件 (arduino-littlefs-upload-1.1.7.vsix) 在 Samovar 归档的 tools\ESP32LittleFS_Arduino IDE 2.X 目录中，见[首次上传草图](#)。

可以在电脑上同时安装多个 Arduino IDE 2.x.x 和 Arduino IDE 1.x.x 版本，它们互不影响。因此可以继续把 Arduino IDE 2.x 作为主要 IDE，只在需要把 LittleFS 文件系统（包括界面）本地上传到 ESP32 时才使用 Arduino IDE 1.x。

使用不同版本的 Arduino IDE 时，需要把它们安装到不同的目录中。为此只需在目录名中加上版本号。

例如。

C:\Program Files(x86)\Arduino1_1_18;

C:\Program Files\Arduino IDE_2_3_2;

C:\Program Files\Arduino IDE_2_3_1

这样就可以在每个版本的 Arduino IDE 中使用不同版本的 Espressif Systems ESP32 SDK。

但要注意，您使用的每个 Arduino IDE 版本，都需要单独为 ESP32 进行配置。

注意！

如果使用 3.0.0 或更高 版本的 ESP32 扩展 “ESP32 by Espressif Systems”，固件会自行禁用 Wi-Fi（OTA）更新：USE_UPDATE_OTA 参数在此内核中不起作用，只能通过 USB 上传。此外，还需要更改所用的开发板类型并选择正确的分区。3.x 内核中没有 Samovar 分区表，需要手动添加（说明在 Samovar.ino 文件开头）：

• 把草图目录中的 partitions\samovar.csv 文件复制到 ... \Arduino15\packages\esp32\hardware\esp32\3.x.x\tools\partitions\（在 Windows 上，Arduino15 目录位于 C:\Users\<name>\AppData\Local\）；

• 在 ... \esp32\3.x.x\boards.txt 文件中添加三行：

esp32.menu.PartitionScheme.samovar=Samovar

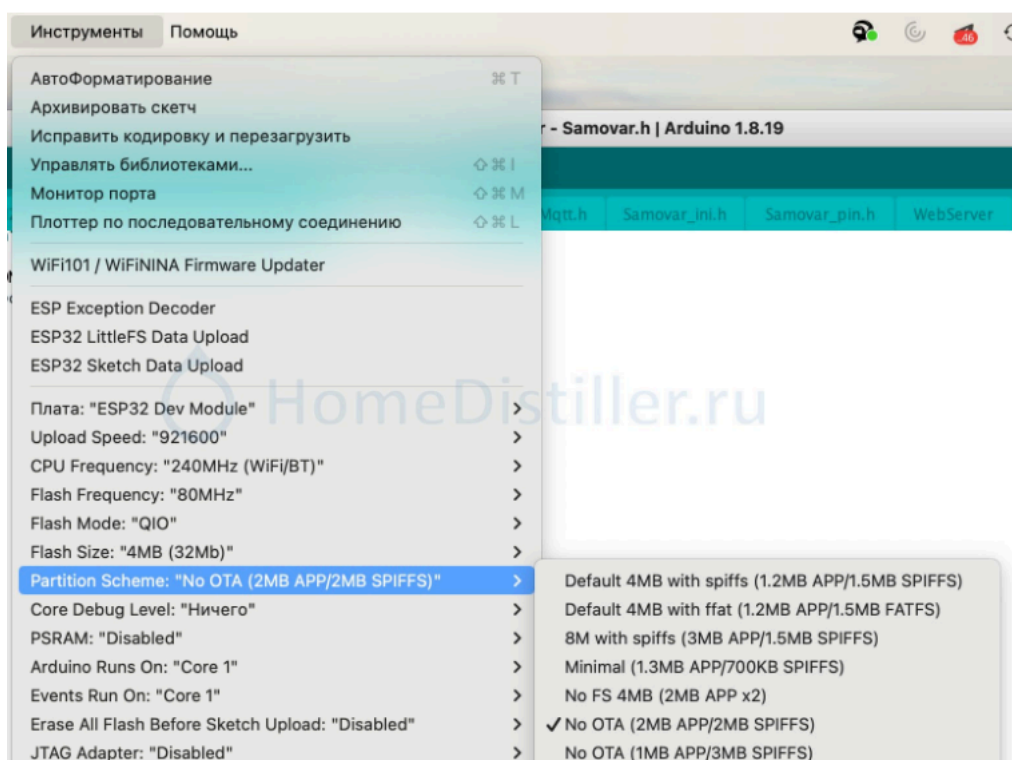
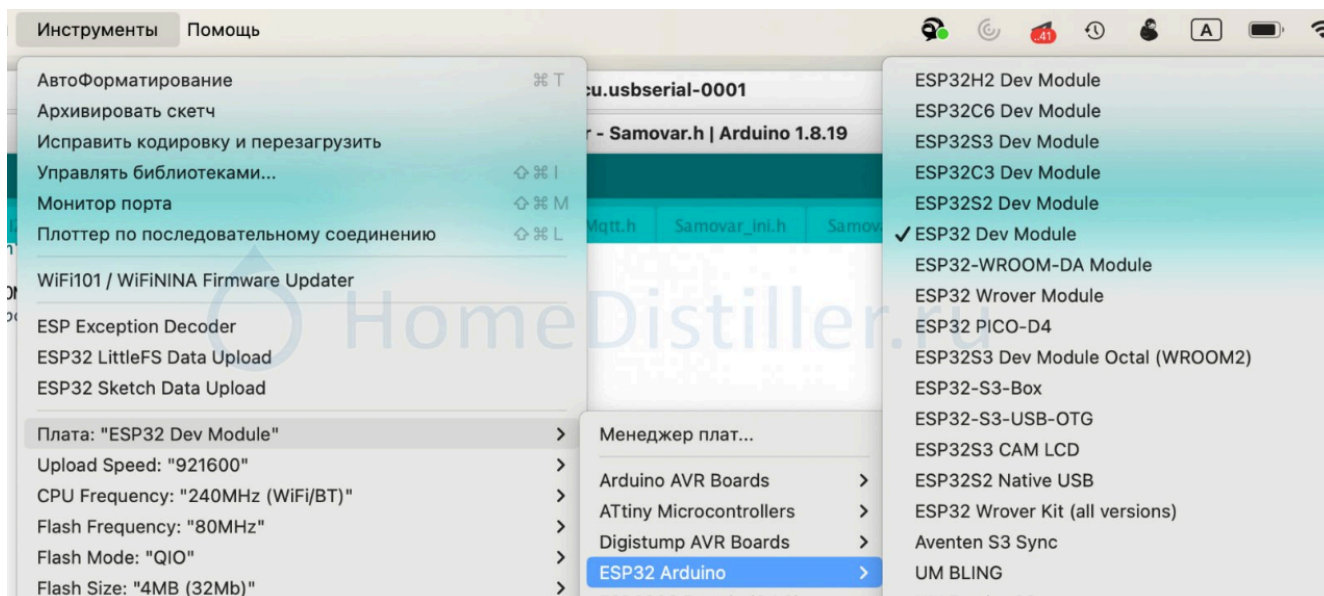
esp32.menu.PartitionScheme.samovar.build.partitions=samovar

esp32.menu.PartitionScheme.samovar.upload.maximum_size=1507328

• 重启 Arduino IDE，选择“Tools”→“Partition Scheme”→“Samovar”。

PlatformIO 配置器使用 2.0.17 内核和同一个分区表（partitions/samovar.csv）；无需添加任何内容。

[论坛上的帖子](#)。



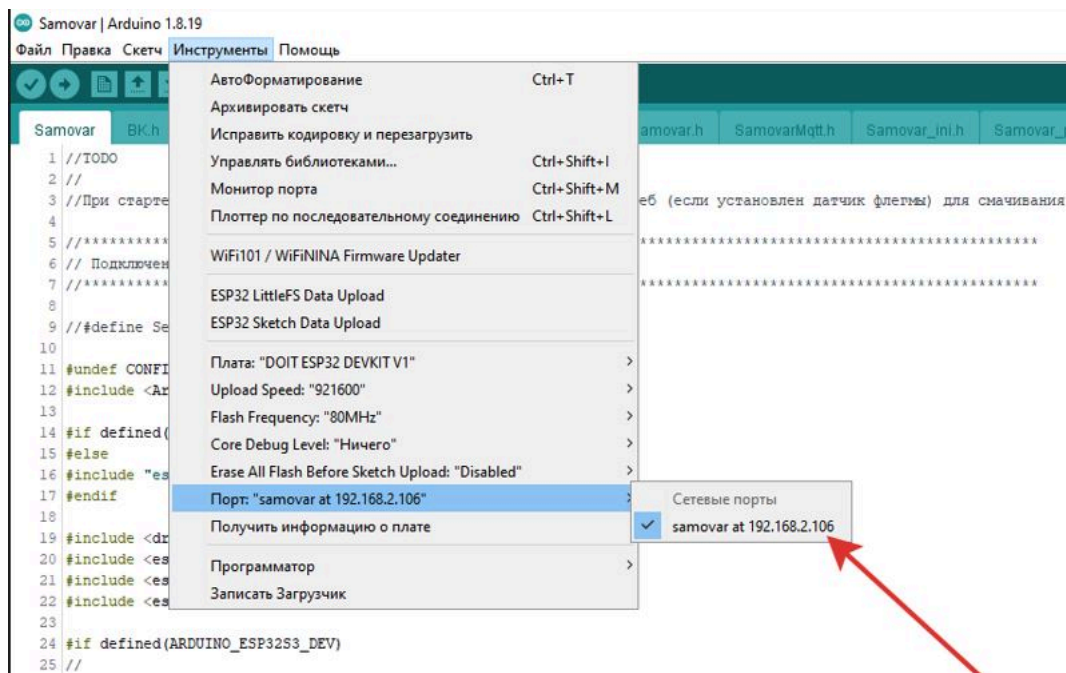
问题与解答. FAQ

我到处都找不到如何“无线”更新固件。请指点.....

第一次无论如何都需要通过数据线 烧录，并把 Samovar 连接到您的 Wi-Fi 网络。Samovar_ini.h 中必须启用 USE_UPDATE_OTA 参数（默认启用；在配置器中为“允许通过 Wi-Fi 更新”（«Разрешить обновление по Wi-Fi»）），ESP32 内核为 2.0.17。

重启 Samovar 并重新启动 Arduino IDE。之后，Samovar 的 IP 地址会出现在 Arduino IDE 的网络端口 中（名为 samovar 的网络端口）。

选择该 IP 地址代替 USB (COM) 端口，更新就会以无线方式进行。



以后，如果用 USB 线把 Samovar 连接到电脑，端口中会有两个选项：“无线”更新或通过 COM 端口更新。选择哪个通道，草图就通过哪个通道发送。

配置器的工作方式相同：点击“刷新”（«Обновить»），端口列表中会同时出现 COM 端口和在网络中找到的 Samovar；选择网络端口后点击“烧录”（«Прошить»）。“上传 LittleFS”（«Загрузить LittleFS»）按钮同样可以通过网络使用。

远程控制

Samovar 网页界面设计用于可信的局域网。固件中的控制页面没有内置用户名和密码验证。

不要把本地 Samovar 网页界面公开到互联网上。 任何人拿到这样的地址，都能打开界面并向设备发送命令。

如需从其他网络访问，请使用到家庭网络的安全连接，例如在路由器上或单独服务器上搭建 VPN。VPN 是一种加密连接，连接后手机或电脑就像在家里一样工作。具体设置取决于路由器型号，不属于 Samovar 固件的功能。

如需通过互联网进行监控、接收通知和控制，而不开放网页界面，请使用 samovar-tool.ru 服务器：Samovar 会自行连接到该服务器，通过它可以使用 [适用于 Android 和 iOS 的 Samovar 移动应用以及网站上的控制页面](#)、Samovar 的推送通知，以及 [云端日志和图表](#)。单个应用可以凭令牌（token）使用，无需在网站上注册。如有多台 Samovar，可在网站菜单“用户 → 设备”中设置共享列表和名称。每台设备的令牌在其设置的“[其他](#)”选项卡中输入，切勿公开。

远程启动前，请确认设备组装安全，并且附近有人能够立即切断电源和原料供应。远程界面不能取代硬件保护措施和对设备的现场看管。

移动应用与网站

Samovar 会把读数、程序状态、过程日志和消息发送到 samovar-tool.ru。可以通过互联网在 Android 移动应用、同款 iOS 应用以及网站的“Samovar Control”页面上查看并控制过程。应用和网站页面的界面完全相同，因此下面使用网站截图进行说明。

为此无需向互联网开放任何端口：连接由 Samovar 自行建立，见[远程控制](#)。

设备令牌

令牌 (token) 是由 32 个拉丁字母、数字、“-”或“_”字符组成的密钥，服务器靠它识别您的 Samovar。每台设备都有自己的令牌。固件、应用、控制页面和云端日志都会用到它。

第一台设备。首次启动移动应用时选择“创建账户”，输入邮箱和密码。系统会创建一个 Blynk 账户，把 Samovar 项目复制到其中，令牌会显示在“设置”界面上。如果已有 Blynk 账户，点击“登录并获取令牌”，或手动输入已知的令牌。只有一台设备时无需在网站注册。

在哪里输入。在 Samovar 网页界面中打开“设置” («Настройки») ，进入“[其他](#)” («[Прочие](#)») 选项卡，把令牌粘贴到“Blynk 令牌” («Токен Blynk») 字段中，然后点击“保存” («Сохранить») 。之后断电重启 Samovar：与服务器的连接在启动时建立。也可以在首次连接 Wi-Fi 时，在[Samovar 接入点页面](#)的“blynk token”字段中输入令牌。

不要把令牌告诉任何人。知道令牌的人都能查看读数并控制您的 Samovar。

网站

在网站 www.samovar-tool.ru 上注册并登录。打开菜单“用户 → 设备”。注册时网站会为您创建一个名为“设备”的设备：奶酪、啤酒和蒸馏日志都关联在它上面。这是一台拥有自己令牌的完整设备。要把 Samovar 连接到它，在其卡片上点击“显示令牌”，输入当前的网站密码，然后把显示出的令牌写入 ESP32 设置中。

如果在 Blynk 中已经有 Samovar 项目，点击“导入设备”，输入名称、Blynk 令牌和当前的网站密码。“创建设备”按钮会再创建一个 Samovar 项目，并只显示一次它的新令牌。把该令牌写入对应 ESP32 的设置中。列表中任何设备的令牌之后都可以用“显示令牌”按钮查看。

设备数量有限制。默认每个用户允许一台设备，注册时创建的“设备”也计算在内。页面顶部会显示已用数量和允许数量：“已用设备：1 / 1”。达到上限后，“导入设备”和“创建设备”按钮将不再显示；如果 Blynk 账户中的 Samovar 数量多于空余名额，账户迁移会被拒绝。要连接更多 Samovar，请联系网站管理员：管理委员会提高上限。降低上限时，已添加的设备不会被删除。

名称同时保存在网站和 Blynk 中。在同一页面上可以重命名设备，并为“[控制](#)”页面和“日志”页面选择 Samovar，见[云端日志与图表](#)。

应用中的多台设备

在应用设置中输入网站的邮箱和密码，点击“连接网站设备”。网站和 Blynk 的邮箱与密码必须一致。如果不一致，请先按下面的说明迁移账户。应用会加载设备列表。只有一台设备时会自动选中；有多台时，设置中会出现名称列表。

Android、iOS 和网站各自独立保存所选设备。例如，Android 可以控制家里的 Samovar，iOS 控制车库里的 Samovar，网站控制第三台设备。在一处更改选择不会切换其他地方。在网站上做了更改后，请在应用中点击“刷新设备列表”。

公共列表中所有设备的推送通知都会送达。点击通知时，应用会自动切换到发送该通知的 Samovar。仅收到通知本身不会改变当前选择。在单设备模式下，推送在登录或注册 Blynk 之后才会生效；只有令牌而没有邮箱和密码，无法为手机注册通知。

如果 Blynk 邮箱不同

在“用户 → 设备”页面上点击“迁移 Blynk 账户”。输入旧的 Blynk 邮箱、旧的 Blynk 密码和当前的网站密码。网站会核对输入的数据，并把符合条件的项目转移到使用网站邮箱和密码的账户中。

注意。只迁移名为 Samovar（不区分大小写）、只有一台设备、并带有“主页” («Основное») 、“控制” («Управление») 、“图表” («Графики») 选项卡的项目。名称不同、选项卡组合不同或包含多台设备的项目不会被迁移。迁移成功后，旧 Blynk 账户连同其中所有其他项目都会被删除。如果没有找到 Samovar 项目，旧账户同样会被删除。迁移前没有预览列表，迁移成功后也无法撤销。

迁移后的 Samovar 令牌保持不变，因此无需修改 ESP32 设置。

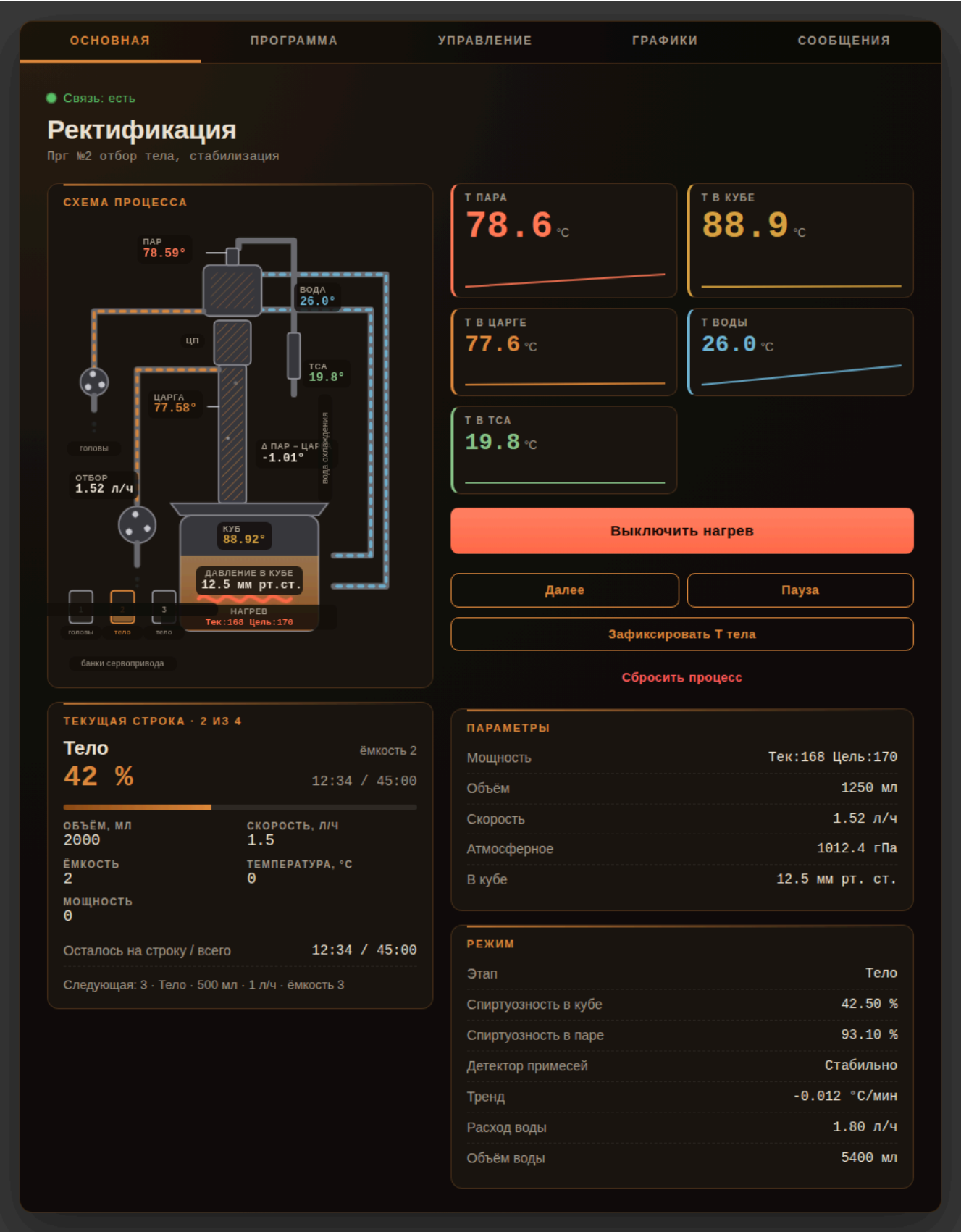
如果网站在迁移过程中报错，不要立即重试。先用网站邮箱和密码登录 Blynk，再重新打开“设备”页面：可能 Blynk 中的迁移已经完成，只是网站上的列表还没有更新。

迁移后请使用网站和 Blynk 共用的邮箱和密码。要同时修改两处的密码，在设备页面上点击“修改共用密码”。修改密码后，请重新登录网站和应用。

浏览器标签页打开期间，控制页面每 3 秒刷新一次数据。报警会显示在弹出窗口中，直到窗口被关闭。手机推送通知只能通过移动应用接收。

界面

“主页”。连接状态、模式和状态行，带当前温度的过程示意图（与网页界面相同：按模式、啤酒酿造顺序以及冷却水上方的第二台泵显示），带迷你图表的传感器卡片，“开启加热/关闭加热”“下一步”“暂停”“锁定酒心 T”和“重置流程”按钮，当前程序行、参数和模式。



网站上的“主页”界面；应用中完全相同

“程序”。当前模式的程序行及各字段说明，正在执行的行会高亮显示。这里的程序只能查看：修改需在 Samovar 网页界面中进行。



“控制”。调压器电压、取酒速度、锁定酒心温度、I2CStepper 搅拌器和泵（参数、启动和停止、继电器 R1-R4、泵校准）、执行一行 Lua 代码以及系统信息：Samovar 的 IP、固件版本、模式、时间。

● Связь: есть

Ректификация

Прг №2 отбор тела, стабилизация

МОЩНОСТЬ (НАПРЯЖЕНИЕ)

Сейчас: Тек:168 Цель:170

Напряжение, В

Установить

СКОРОСТЬ ОТБОРА

Сейчас: 1.52 л/ч

Скорость, л/ч

Установить

Зафиксировать Т тела

МЕШАЛКА I2CSTEPPER

Статус: работает · Скорость: 120 шаг/с

RPM

60

Работа, сек

30

Пауза, сек

10

☐ Реверс после паузы

☒ Плавный старт

☐ Датчик активен уровнем 1

☐ Стоп по внешнему датчику

Применить

Работает — остановить

Сохранить

R1: выкл

R2: выкл

R3: выкл

R4: выкл

НАСОС / НАПОЛНЕНИЕ I2CSTEPPER

Статус: работает, калибровка · Осталось: 320 мл · Скорость: 400 шаг/с

☐ Прокачка

☒ Наполнение

Прокачка, мл/час

1000

Пауза насоса, сек

5

Наполнение, мл

500

Скорость наполнения, мл/час

2000

Шагов на 1 мл

150

☒ Плавный старт

Применить

Работает — остановить

Сохранить

Калибровка старт

100 мл стоп

R1: вкл

R2: выкл

R3: вкл

R4: выкл

ВЫПОЛНИТЬ LUA-СТРОКУ

Команда Lua

Выполнить

СИСТЕМНАЯ ИНФОРМАЦИЯ

IP Самовара	192.168.1.55
Прошивка	7.00
Режим	Ректификация
Время системы	12:02:12
Сервер	www.samovar-tool.ru

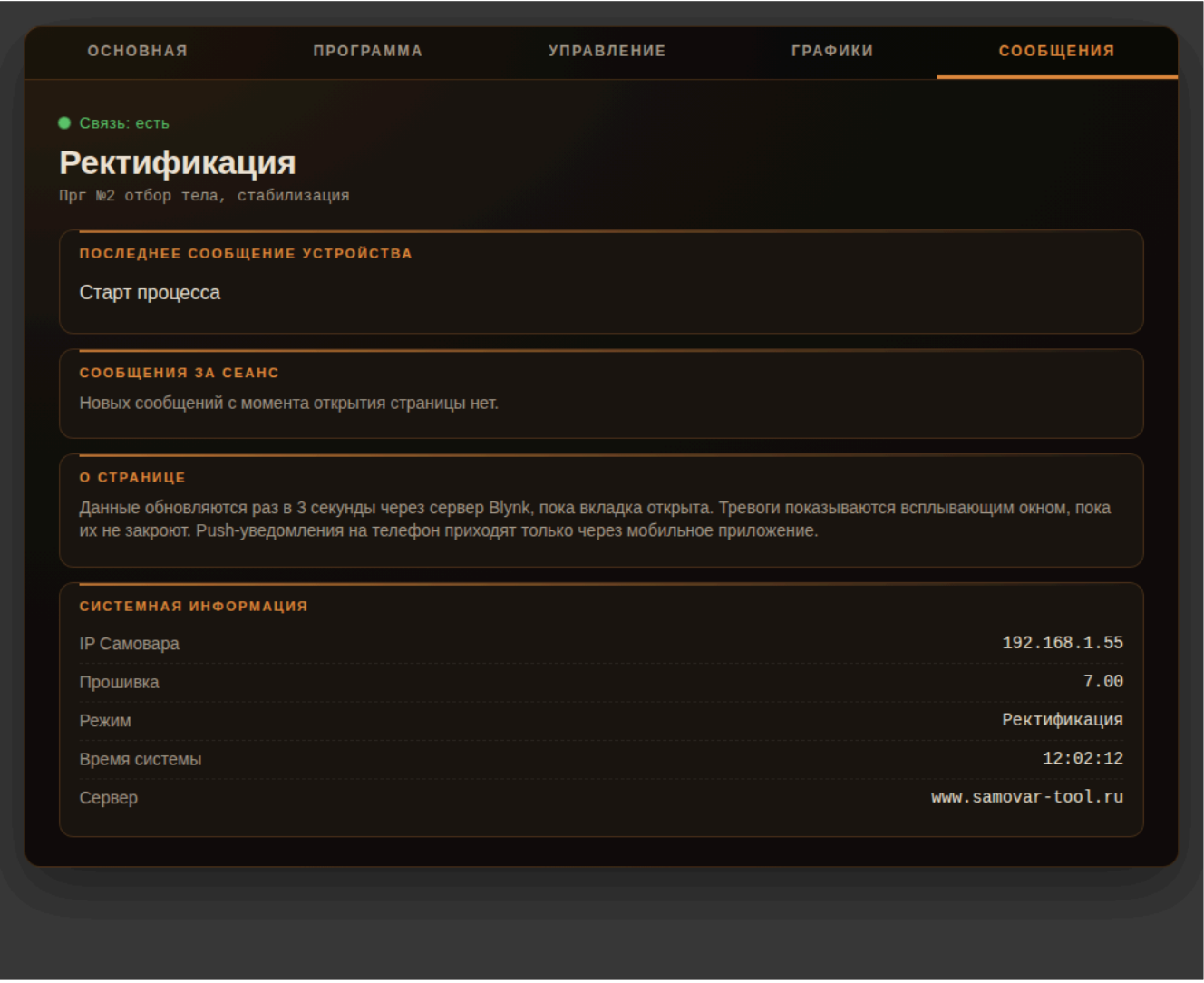
“控制”界面

“图表”。来自服务器的当前会话历史：分为“温度”“压力”“取酒与冷却水”“酒精”“功率”几组，时间范围从 1 分钟到整个会话，各条曲线可用图表下方的按钮单独打开或关闭。



“图表”界面

网站上的“消息”，应用中的“设置”。在网站上，第五个选项卡显示所选设备的最新消息以及本次会话中收到的消息。在应用中，这个位置是“设置”界面：账户、单个令牌或设备选择，以及通知设置；应用以推送通知的形式显示消息。



网站上的“消息”选项卡

工作原理

Samovar 自行把读数发送到服务器：温度、压力、电压和取酒速度每 5 秒发送一次，其余数据（状态、程序行号、程序、加热和暂停状态）在变化时发送。来自应用和网站的命令会立即执行，不需要确认。两条命令之间请间隔 1-2 秒。

Samovar 的消息显示在“消息”选项卡中，并以推送通知的形式发送到应用；报警带有“报警！”（«Тревога!»）前缀，警告带有“警告！”（«Предупреждение!»）前缀。固件还通过同一台服务器传输过程日志和会话开始信息；这些数据保存在服务器上，可在网站上查看，见[云端日志与图表](#)。

注意。远程启动前，请确认设备已安全组装，并且附近有人能够立即切断电源和原料供应。远程界面不能代替硬件保护措施和对设备的现场监控。

使用的调压器

警告！

所有连接到 Samovar 的调压器都必须在电压上实现物理隔离。隔离器件可以是继电器、光耦、ADUM 等数据总线隔离器等。

绝对不要直接连接调压器，否则市电电压会传到您和 Samovar 身上。这种情况下最好的结果是 Samovar 烧毁，其他后果会严重得多。

Samovar 7.00: 调压器类型在配置器的“调压器”（«Регулятор»）选项卡中选择（“功率调节器”（«Регулятор мощности»）字段）：KVIC、RMV-K、SEM_AVR 或“不使用”（«Не использовать»）。配置器通过脚本 flash_windows.bat 或 flash_macos_linux.sh 启动，并会自行选择所需的编译环境；下文给出的 PlatformIO 环境名称供手动编译时使用。

Samovar 可配合以下类型和型号的调压器工作：

通过 UART 协议控制的调压器

注意！使用 UART 控制时负载接通/断开的算法：

接通。

收到“开启加热”（«Включить нагрев»）命令后，Samovar 接通 1 号继电器，由它接通接触器。然后等待以下行指定的时间 `#define SAMOVAR_USE_POWER_START_TIME`（位于文件 **Samovar_ini.h** 中，默认 2 秒；对于 SEM_AVR，固件还会再等 3 秒让调压器启动），然后命令调压器输出电压。这样可以保证接触器在负载电路中没有电压时闭合，从而避免打火，延长接触器寿命。在调压器输出电压的同时，4 号继电器接通，用于连接[辅助加热管](#)（在使用辅助加热管的模式下）。

断开。

收到“关闭加热”（«Отключить нагрев»）命令后，Samovar 命令调压器切断电压，等待执行完毕，0.2 秒后断开 4 号继电器和 1 号继电器，后者再断开接触器。如果调压器没有响应命令，继电器也会在超时后断开，消息中会出现报警“调压器命令超时”（«Таймаут команды регулятора»）。

1. 调压器 KVIC

在配置器中将“功率调节器”选为 KVIC（这是默认值）。在 PlatformIO 中对应 Samovar 环境。整个配置块的说明见[草图设置](#)一节。

与 Samovar 完全兼容。编码器调节步长 0.5 伏。稳压精度 0.1 伏。紧急情况下可由独立传感器关断。

2. 调压器 RMV-K（只有 5 号和 6 号版本能配合 Samovar 工作）

所有 RMV-K 型号的控制触点默认都带电源隔离，配合 Samovar 使用无需改装。

在配置器中将“功率调节器”选为 RMV-K。在 PlatformIO 中对应 Samovar_rmvk 环境（SAMOVAR_USE_RMVK 指令）。固件会自动启用通用调压器支持。

无论在调压器上还是由 Samovar 调节，步长都是 1 伏。稳压精度 1 伏。6 号版本在紧急情况下可由独立传感器关断。Samovar 通过 RMV-K 完全控制升温、稳压和紧急切断加热。

注意！由于 RMV-K 设定稳压电压时只接受整数，Samovar 在[预液泛](#)模式下工作时会有一些特殊之处。

3. 调压器 SEM_AVR

在配置器中将“功率调节器”选为 SEM_AVR。在 PlatformIO 中对应 Samovar_sem 环境（SAMOVAR_USE_SEM_AVR 指令）。固件会自动启用 RMV-K 协议支持和通用调压器支持。

与 Samovar 兼容。有一个特点：在 Samovar 界面中，输出到负载的功率不是以伏特（如前两种调压器）而是以瓦特设定和显示的。

为了与 Samovar 正常配合，请使用 Samovar 附带的 Stab-avr.ino 草图，它位于目录 ...Samovar\Stab-avr 中

不支持通过协议外部控制的调压器。

例如：

RM2, RM2M, RM2-Pro（以下简称 RM）。

所有 RM 型号的控制触点默认都带电源隔离，配合 Samovar 使用无需改装。

如果 Samovar 不通过 UART 控制调压器，请在配置器中将“功率调节器”选为“不使用”（«Не использовать»）。在 PlatformIO 中对应 Samovar_no_power 环境：SAMOVAR_USE_POWER、SAMOVAR_USE_RMVK 和 SAMOVAR_USE_SEM_AVR 指令均关闭。这种编译版本中没有 [NBK](#) 模式。

控制这类调压器归结为两个参数：

- 开启/关闭调压器本身。

- 启动/停止“升温”（«Разгон»）模式

这通过两个继电器实现，由它们的触点控制调压器。收到 Samovar 的“开启加热”（«Включить нагрев»）命令时，1 号继电器的触点开启 RM，同时 4 号继电器的触点发出开启“升温”模式的命令。

当蒸汽温度达到文件 Samovar_ini.h 中指定的值时：（此处为 39°）

//Vapor sensor temperature at which the column switches from boost mode to working mode

#define CHANGE_POWER_MODE_STEAM_TEMP 39

4 号继电器断开触点，RM 关闭升温模式。运行结束时，Samovar 通过 1 号继电器的触点关闭 RM。

注意。

如果关闭加热后再重新开启，而蒸汽传感器温度高于草图中指定的值，升温不会开启。

安全系统传感器触发时，Samovar 会完全关闭 RM。

说明：

- 上述所有调压器都可作为独立设备工作，也可以不经 UART 连接、按照 RM 调压器的原理和算法与 Samovar 配合使用。

- 选型参考：[调压器概览](#)

- 除了控制调压器，Samovar 还能[控制一根附加的辅助加热管](#)

问题与解答。FAQ

Samovar 和调压器显示的功率不一致。如何解决？

总的来说，绝对功率并不那么重要，但让调压器和 Samovar 上显示的功率一致是可以做到的，而且基本上只有一种方法。

包括 Samovar 在内，没有一个调压器是在测量功率。它们像计算器一样，根据两个值计算功率：当前加在加热器上的电压和输入的加热器电阻。

输入的电阻是恒定的，而电压会变化。整个调整过程就是选取加热器电阻值。

可以使用计算电阻：电压的平方除以功率。电压取加热管铭牌上的值。以 3 kW 220 V 加热管为例：

$220^2 / 3000 = 16.133333333333 \text{ Ohm}$ 。

也可以用万用表测量。

适用于 RMV_k 调压器以及用中国模块组装的简易调压器 (Stab-avr)。

注意！ 如果并不打算真正加热蒸馏釜，请断开加热器。

- 在 RMV_k 中可以输入功率和电压值，电阻会自动计算；或者输入电阻，由它计算功率。[设置方法视频](#)。看完照做，别忘了把调压器切换到瓦特显示模式。

- 在 Stab-avr 中，用电位器设定当前的实际市电电压，并输入您的加热器功率。

注意！ 对于 Stab-avr，加热管功率必须按 230 伏市电电压输入。如果加热器厂家在说明书中给出的是 220 伏下的功率，请换算到 230 伏。

启动 Samovar 加热。在“升温”模式下调整没有意义 — 电压一直在变，功率也一样。从 Samovar 发出命令，让调压器设定某个具体电压（例如 RMV_k 设 170 伏，Stab-avr 设 2000 瓦）。调压器转入稳压模式。

比较 Samovar 和调压器上显示的功率值。

接下来有两种方法 — 要么修改 Samovar 中的电阻值（网页界面：“设置” («Настройка»)，“其他” («Прочие») 选项卡，“用于计算功率的加热管电阻” («Сопротивление тэна для расчета мощности») 字段），这要方便得多；要么修改调压器中输入的功率值。目标是让调压器和 Samovar 上的数值尽可能接近。

调压器 Kvic。

它没有功率显示。因此只调整 Samovar，使其显示您想看到的功率……

这是第一种方法

第二种是使用这样的链路：独立的电压电流表 — 调压器 — Samovar。



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

★★★★★ 4,8 222 отзыва 345 продано

[Описание](#) [Характеристики](#)

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

Если时常更换不同的加热管，和/或在加热管组中切换加热管，这种方法很方便。

电压电流表真正测量消耗的电流，并在显示屏上显示实际功率。所以如果希望 Samovar 显示这个功率，只需在运行中随时修改 Samovar 里的加热器电阻值即可。

使用辅助加热管

Samovar 可以在“[精馏](#)”、“[蒸馏](#)”和“[啤酒](#)”模式下控制一根附加的辅助加热管。辅助加热管接在 4 号继电器上；接线方法见[组装手册](#)。

网页界面中没有单独的辅助加热管按钮：Samovar 根据温度自行开关它。阈值在配置器的“温度”（«Температуры»）选项卡中设置（“退出升温温度，°C”（«Температура перехода из разгона, °C»）、“开启冷却温度，°C”（«Температура открытия охлаждения, °C»）、“辅助加热管阈值，°C”（«Порог разгонного ТЭНа, °C»））；在 Arduino IDE 中编译时，相同的行位于 [Samovar_ini.h](#) 中。

“精馏”（«Ректификация»）模式下辅助加热管的工作

使用附加辅助加热管功能无需修改草图中的任何内容。该功能默认启用。

其逻辑与[RM](#)在“升温”（«Разгон»）模式下的逻辑类似：

收到“开启加热”（«Включить нагрев»）命令时（来自网页界面、移动应用或[万能按钮](#)），Samovar 除了开启主加热管外，还通过 4 号继电器接通接触器或固态继电器，由后者给辅助加热管供电。此时状态行显示“升温”（«Разгон»）。当蒸汽温度达到设置中的值（默认 39°）时：

```
//Steam sensor temperature at which the column switches from boost mode to working mode
#define CHANGE_POWER_MODE_STEAM_TEMP 39
```

4 号继电器断开触点，关闭辅助加热器。Samovar 显示消息“升温结束。稳定/全回流运行。”（«Разгон завершён. Стабилизация/работа на себя.»）

使用 UART 控制的调压器时，如果釜温升到开启冷却温度（默认 77°）以上，即使蒸汽温度尚未达到 39°，辅助加热管也会提前关闭。如果没有连接釜温传感器，Samovar 不控制辅助加热管。

如果没有发出“开启加热”（«Включить нагрев»）命令，或传感器温度高于草图中设定的值，升温不会开始。

“蒸馏”（«Дистилляция»）模式下辅助加热管的工作

蒸馏开始时，辅助加热管与主加热管一同开启。在[蒸馏程序](#)第一次转到下一行时关闭。如果在设置中（“设置”（«Настройки»）→“啤酒”（«Пиво»）选项卡）取消勾选“[沸腾时使用辅助加热管](#)”（«Использовать разгонный тэн при кипении»）复选框，辅助加热管会更早关闭 — 一旦 Samovar 检测到开始沸腾即关闭。

“啤酒”（«Пиво»）模式下辅助加热管的工作

加热管的连接方法与上文相同。

但算法不同。

糖化期间，辅助加热管由“辅助加热管阈值，°C”（«Порог разгонного ТЭНа, °C»）参数控制（配置器的“温度”（«Температуры»）选项卡）；在文件 [Samovar_ini.h](#) 中对应的行是：

```
#define ACCELERATION_HEATER_DELTA 4
//If the difference between the target temperature and the current temperature (for beer and sous-vide modes) is greater than the set value,
then with a UART-controlled regulator relay No. 4 additionally switches on the boost heating element; with a smaller difference, the boost
heating element switches off
```

默认启用；4 是以 C° 为单位的数值。 仅在使用 UART 控制的调压器时有效。

开启后，Samovar 命令调压器向主加热管输出全功率。如果釜温比[糖化程序](#)该行设定的温度低 4°，Samovar 接通 4 号继电器，由它命令辅助加热管开启。如果温差小于 4°，辅助加热管不开启，只由主加热器加热。

当前一个温度停留阶段结束，或操作者发出“下一个程序”（«Следующая программа»）命令时，Samovar 把釜温与下一程序行设定的温度比较。如果温差大于 4°，就使用辅助加热管。

如果没有发出“开启加热”（«Включить нагрев»）命令，或温差小于设定值，辅助加热管不开启。

麦汁沸腾时，如果勾选了“沸腾时使用辅助加热管”（«Использовать разгонный тэн при кипении»）复选框，辅助加热管开启，否则关闭。PID 控制器自整定期间不使用辅助加热管。

Samovar 7.00: 在“[奶酪制作](#)”（«Сыроварение»）模式下，4 号继电器被盐水排放阀占用，因此辅助加热管和该阀门不能同时连接。在“BK”（«БК»）、“NBK”（«НБК»）和“低温慢煮”（«Су-вид»）模式下不使用辅助加热管。

冷却

工作原理（以精馏模式和原版草图为例）。

达到冷却启动条件（蒸馏釜传感器温度达到 77°）时，发出打开供水阀（3 号继电器）的命令，同时发送 PWM 控制信号启动水泵。最初十秒水泵以启动功率运行（默认 40%），之后 PWM 调节器调整转速，使分凝器出口的冷却液（以下简称冷却液）温度保持在设定值，且不低于最低功率（默认 10%）。随着塔升温、冷却液温度升高，水泵会加大流量。升温结束时，水泵会再次以启动功率运行几秒钟以加大水流，之后调节器继续工作。

如果接有 TCA 传感器，且其温度高于 39° 并高于水温，水泵会加大出力：调节器把水温视为比设定值高 3°。

冷却在两处设置：

1. 在配置器的“温度”（«Температуры»）和“水泵”（«Насосы»）选项卡中，或在 Arduino IDE 中编译时，在[草图](#)的以下各行中：

```
//Значение температуры датчика в кубе, при котором откроется клапан подачи воды и включится насос подачи воды
#define OPEN_VALVE_TANK_TEMP 77
```

在配置器中为“冷却开启温度，°C”（«Температура открытия охлаждения, °C»）。蒸馏釜温度达到 77° 时，发出打开供水阀的命令，发送 PWM 控制信号启动水泵，并伴随声音信号向操作者发送消息“请打开供水！”（«Откройте подачу воды!»）。无论是否使用水泵或阀门，该消息都会发送。如果接有 TCA 传感器，冷却会更早开启——在 TCA 温度升到 70° 时。

```
#define DELTA_T_CLOSE_VALVE 20
```

在蒸馏和精馏模式下，蒸馏过程结束后，水泵继续运行，直到水温比设定的冷却水温度低这个差值为止。单位为摄氏度 °C，可以修改。强烈不建议关闭此功能。（详见[这里](#)）。在配置器中为“冷却关闭温度余量，°C”（«Запас температуры выключения охлаждения, °C»）。

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

在配置器中为“水泵最低功率，%”（«Минимальная мощность насоса, %»）。众所周知，电机启动所需的电压高于维持运转所需的电压。此行设置电机能维持供水泵不停转的最低值，单位为 %。

```
//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
```

在配置器中为“水泵启动功率，%”（«Стартовая мощность насоса, %»）。该值以额定电压的 % 表示，在此值下水泵电机保证能够启动。也就是说，如果您的水泵（假设已装入系统并且充满液体）在 4 伏供电时能可靠启动，低于此电压则无法启动，那么对于 12 伏的额定电压，这就是额定值的 30%。因此可以把 40 改为 30。

2. 在设置的“[温度](#)”（«Температура»）选项卡中——水温传感器的设定值。过程中水泵会保持这个温度。

关闭加热后，满足以下任一条件时，Samovar 关闭水泵并关闭供水阀：水温已降到比水温传感器设定值低 20°，且关闭加热后已过至少三分钟；或者曾加热到 77° 以上的蒸馏釜已冷却到 70° 以下。在精馏模式下接有 TCA 传感器时，还要求 TCA 冷却到 65°。操作者会收到消息“请关闭供水！”（«Закройте подачу воды!»）。

Samovar 7.00: 以前只有在水冷却后才会关水。夏天进水温度较高时，水泵可能一直运行，直到操作者干预；现在蒸馏釜冷却后也会关水。

水路保护。

如果加热开启时水温传感器没有响应，Samovar 会紧急关闭加热。

水温达到 65° 及以上时，Samovar 发出声音信号并显示消息“水温达到临界值！”（«Критическая температура воды!»）；达到 70° 及以上时还会降低调节器功率；达到 75° 及以上时紧急关闭加热。紧急关闭时阀门关闭，水泵停止。

如果在“[其他](#)”（«Прочие»）选项卡中勾选了“使用水流监控”（«Использовать контроль потока воды»），并安装了流量传感器，那么在冷却开启且 20 秒没有水流时，Samovar 会紧急关闭加热，并显示消息“紧急关闭！供水中断。”（«Аварийное отключение! Прекращена подача воды.»）

在**蒸馏**模式下，冷却的工作方式与精馏相同，只是关闭时不检查 TCA 的冷却情况。

在**醪塔**模式下，阀门同样方式打开，水泵功率由 BK 页面上的“水泵速度”（«Скорость насоса»）滑块设置，或根据程序行中的蒸汽温度设定值自动设置，见[链接](#)。

在**NBK**模式下，冷却只在加热开启时打开；与精馏相同，当水温比水温传感器设定值低 DELTA_T_CLOSE_VALVE（20°）或蒸馏釜冷却后关闭。

在**啤酒**模式下，阀门和水泵在“冷却”（«Охлаждение»）程序行中开启，需要冷却麦汁时也在“发酵”（«Брожение»）行中开启；此时水泵以全功率运行。在[糖化程序](#)中设置了麦汁转移的行，同一个水泵输出会启动麦汁转移泵。

在**奶酪制作**模式下，阀门和水泵用于冷却阶段，见[奶酪制作模式](#)。

在**低温慢煮**模式下不使用冷却。

具体实现见[组装指南](#)。

紧急按钮。附加紧急传感器

可以给 Samovar 接一个按钮或通过触点闭合工作的传感器（例如漏水传感器）。所用输入引脚取决于开发板：ESP32 DEVKIT 为 GPIO35，ESP32-S3 为 GPIO48。LILYGO 开发板不支持紧急按钮。

当输入端出现低电平（对地短接）时，Samovar 执行紧急停机：关闭加热并让调压器进入休眠模式，关闭阀门并停止供水，停止取酒泵、搅拌器和 I2CStepper 加料泵，蜂鸣器报警，并发送消息“紧急停机：紧急按钮被按下”（«Аварийное отключение: нажата аварийная кнопка»）。按钮和多个传感器可以并联连接，唯一的要求是传感器须适用于 3.3V。只能通过重启退出紧急模式。如果通电时按钮处于按下状态（传感器闭合），Samovar 启动后会立即进入紧急模式。

注意。ESP32 DEVKIT 的 GPIO35 输入没有内部上拉，因此必须在输入端与 3.3V 之间接一个外部电阻（10 kOhm）。ESP32-S3 使用内部上拉，无需外部电阻。

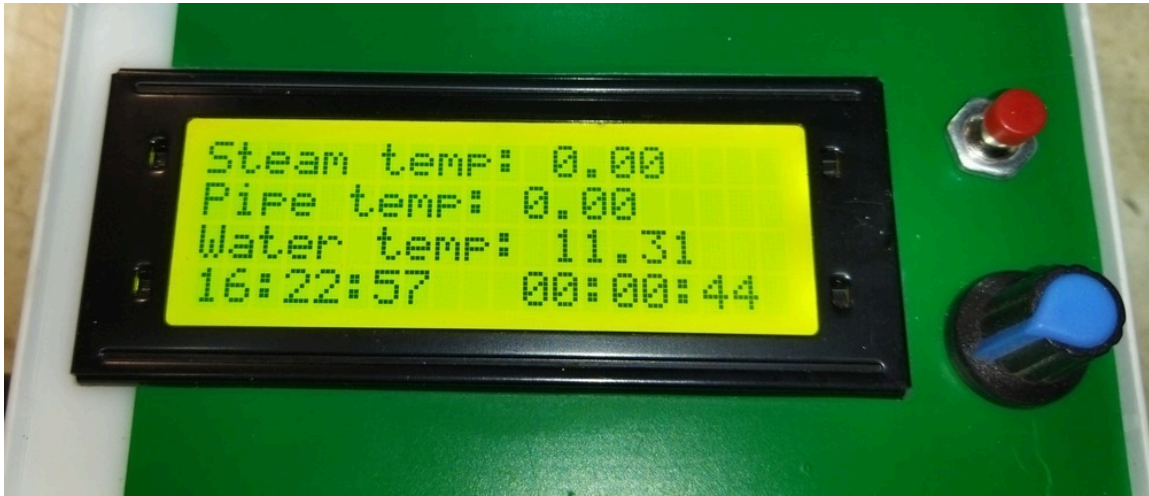
按钮默认处于禁用状态。**Samovar 7.00:** 在配置器的“设备”（«Оборудование»）选项卡中启用按钮（“使用紧急按钮”（«Использовать аварийную кнопку»）复选框）；PlatformIO 中为此准备了现成的环境 **Samovar_alarm_button**。如果手动配置，需要在文件 **Samovar_ini.h** 中将

```
##define USE_ALARM_BTN  
替换为  
#define USE_ALARM_BTN
```

更多细节见 DIY 组装[手册](#)。

LCD 显示屏和万能按钮

推荐配置：四行显示屏、带按钮的编码器，以及一个单独的常开触点按钮。连接方法见[组装指南](#)。没有显示屏、编码器和按钮，Samovar 也能工作——所有控制都可以在网页界面中完成。



万能按钮。

按下按钮的作用取决于 Samovar 的工作模式。区分短按和按住约两秒两种操作。

精馏模式下：

- 如果加热关闭——按下按钮开启加热；当蒸汽温度低于[设定值](#)时，执行快速升温（Boost）阶段
- 如果加热已开启——第一次按下相当于“开始取酒”（«Начать отбор»）命令
- 取酒开始后，按下按钮可暂停和恢复取酒泵。如果正在执行程序的“暂停”行，按下按钮将转到下一行。
- 程序运行时按住按钮——转到程序的下一行，与网页界面中的“下一步”（«Далее»）按钮相同。
- 校准取酒泵时，短按结束校准。

蒸馏模式下：

- 如果加热关闭——开启加热
- 如果加热已开启——短按转到蒸馏程序的下一行，按住则结束蒸馏并关闭加热

醪塔和 NBK 模式下：

- 如果加热关闭——开启加热
- 如果加热已开启——结束过程，关闭加热

啤酒模式下：

- 如果 Samovar 没有在执行程序，则启动程序
- 如果程序正在运行，按下按钮转到程序的下一行。

奶酪制作模式下——与啤酒模式相同；按住则结束程序。

低温慢煮和 Lua 模式下按钮不起作用。

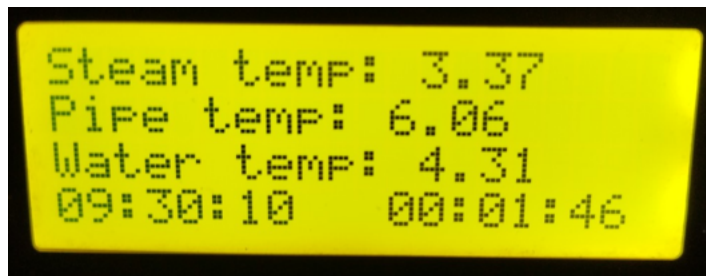
接通电源时：

- 如果按住 2 秒——可以把 Samovar 当作独立的[接入点](#)进行连接。
- 如果接通电源时按住编码器按钮 2 秒——已保存的 WiFi 设置将被删除。

显示屏。

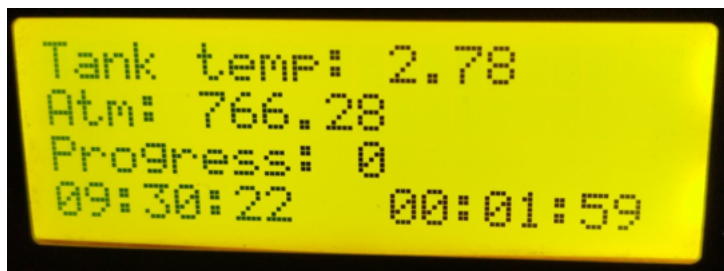
接通电源后，显示屏显示连接 WiFi 和 samovar-tool.ru 服务器的过程。之后在加热关闭时，显示屏轮流显示三个画面。转动编码器切换画面。

第一个画面显示：



Steam T——蒸汽温度
Pipe T——塔身温度
Water T——水温
当前时间和 Samovar 运行时间

第二个画面



Tank T——蒸馏釜温度
Atm——大气压
Progress——当前取酒程序行的完成进度，单位为百分比

第三个画面



Get power——加热（ON 为开启，OFF 为关闭）。通过此项关闭加热会停止当前过程。
Setup——设置
IP——Samovar 在本地网络中的地址

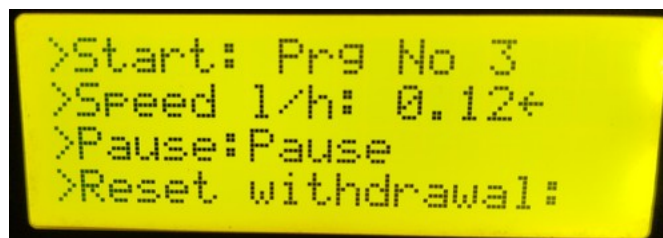
参数前有 (>) 符号的，转动编码器时该项执行操作或改变其值
参数前有 (^) 符号的，表示它是一个数值。

/Setup 和 Prog> 项打开子菜单，<BACK 返回。

要修改参数，请按下编码器，该项右侧会出现箭头 (<)。每按一次，箭头下移一行。转动编码器可改变该项的值或打开子菜单。按住编码器按钮转动时，数值按 10 倍步长改变。

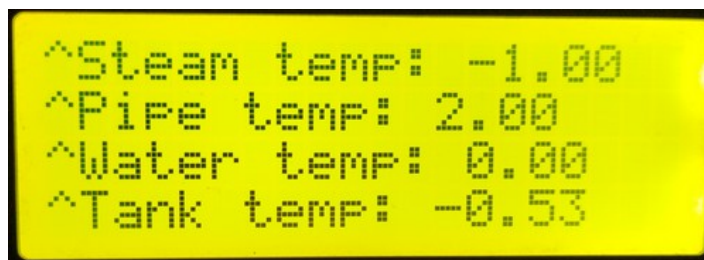
箭头位于最后一行时，再按一次箭头消失，此时转动编码器可切换到下一个画面。再按一次，箭头出现在第一行。

开启加热后，会增加第四个画面——当前状态。



Start——正在执行的程序编号 -> 转到下一个程序（精馏模式下）
Speed l/h——取酒速度 -> 以 0.01 l/h 增减
Pause——暂停/恢复（标签变为 Continue）
Reset withdrawal——重置过程：程序停止，加热关闭。

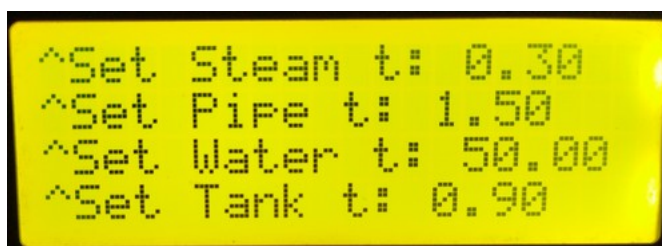
Setup 项子菜单，第一个画面



Steam T——蒸汽温度传感器修正值
Pipe T——塔身温度传感器修正值
Water T——水温传感器修正值
Tank T——蒸馏釜温度传感器修正值

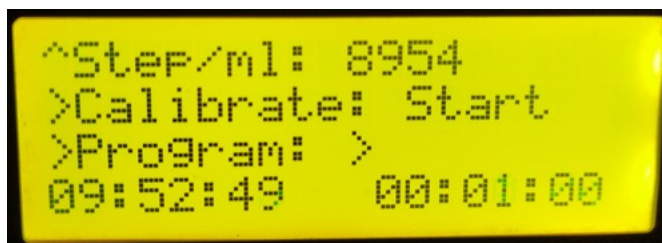
这些值可以修改。按下编码器，出现箭头。箭头所在行可以修改：转动编码器使数值增减 0.01。通过 <BACK 项退出子菜单时，修改会保存到设置中，之后显示在 Samovar 网页界面的“[温度](#)”([«Температура»](#)) 选项卡上。如果保存失败，Samovar 会显示消息“菜单设置未保存”([«Настройки меню не сохранены»](#))。

Setup 项子菜单，第二个画面



Set Steam T——蒸汽温度传感器设定值
Set Pipe T——塔身温度传感器设定值
Set Water T——水温传感器设定值
Set Tank T——蒸馏釜温度传感器设定值

Setup 项子菜单，第三个画面



Step/ml——取 1 毫升所需的步进电机步数
Calibrate——[蠕动泵校准](#)——泵启动，用编码器调节步进电机转速。达到 100 ml 时按下编码器结束校准，Samovar 计算并保存每 1 ml 的步数。
Prog——进入取酒程序子菜单

Setup 项子菜单，第四个画面

<BACK——返回初始画面，并保存已修改的设置

Prog 项子菜单



第一个画面显示取酒程序的第一行。不能编辑
Type——行类型，以字符代码显示 (72 - H, 66 - B, 84 - T)
Volume——体积
Speed——取酒速度，单位为升
Capacity——容器编号。

第二个画面



Reset WiFi——重置 WiFi 设置，Samovar 将重启

<BACK——返回 Setup 子菜单

注意！

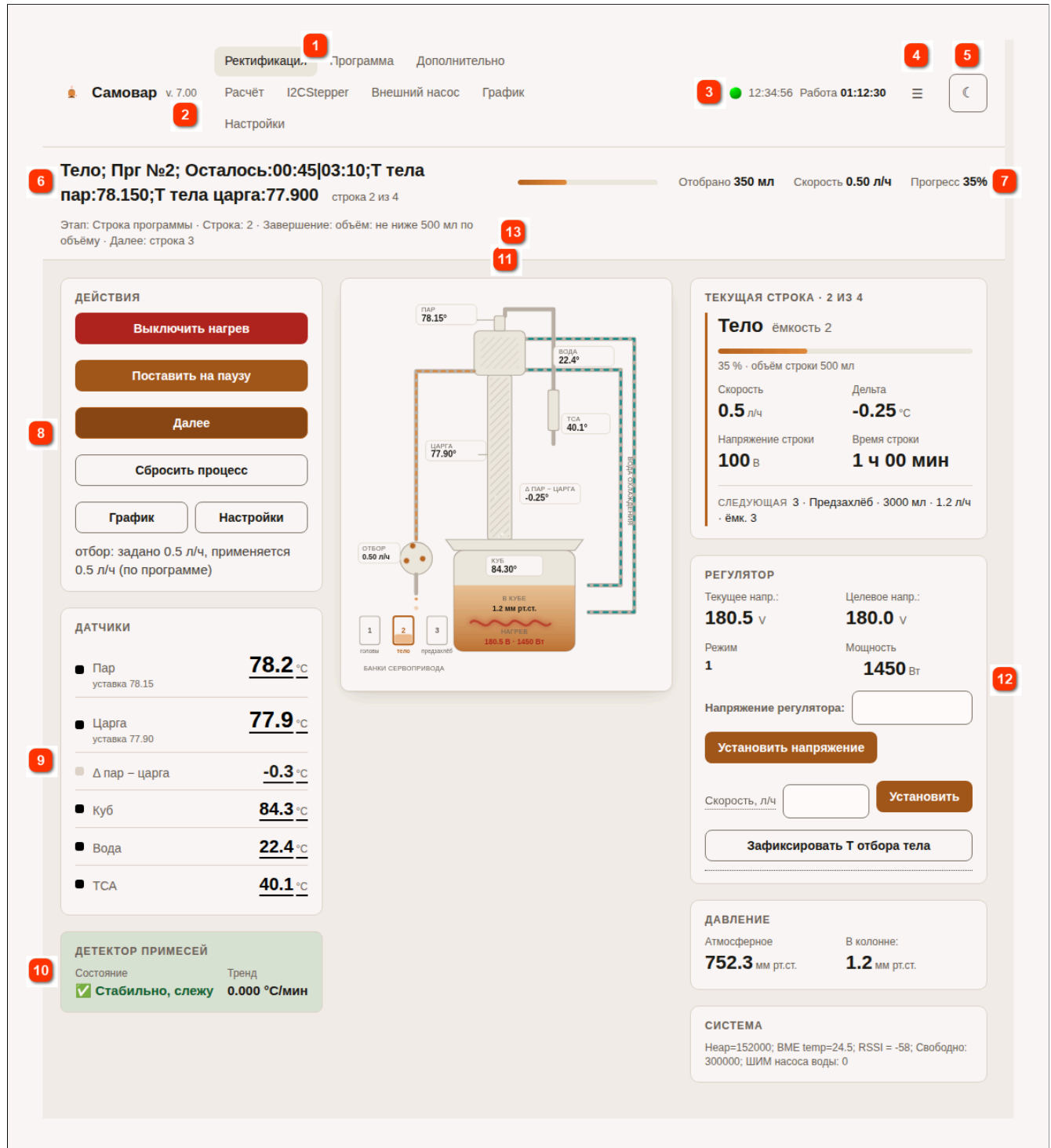
启动时按住按钮或编码器不会重置已保存的参数：按钮启动接入点，编码器只删除 WiFi 设置。

网页界面：主页

Samovar 7.00: 主页已重新设计。读数分组显示在“操作”（«Действия»）、“传感器”（«Датчики»）、“杂质检测器”（«Детектор примесей»）、“当前行”（«Текущая строка»）、“调压器”（«Регулятор»）、“压力”（«Давление»）和“系统”（«Система»）卡片中，中间是显示当前温度的装置示意图。取酒速度和“记录酒心取酒温度”（«Зафиксировать Т отбора тела»）按钮已从“附加”（«Дополнительно»）选项卡移到“调压器”卡片。新增了消息历史（≡ 按钮）和深色主题（C 按钮）。如果显示横幅“配置加载出错”（«Конфигурация загружена с ошибкой»），不要开启加热：先检查传感器分配以及消息中提到的设置。

通电后，Samovar 打开时加热处于关闭状态。页面显示传感器读数，但过程尚未启动。下面介绍“精馏”（«Ректификация»）模式的主页；其他模式下卡片和按钮的组成有所不同，见[Samovar 的运行模式](#)。

页眉和读数



“精馏”（«Ректификация»）模式主页

1

运行模式

第一个选项卡以当前模式命名。模式在设置 → 基本中更改。

2

版本

已安装固件的版本。

- 3 连接和时间**
Samovar 连接指示灯（绿色——数据正在到达，红色——连接中断）、当前时间和开机以来的运行时间。
- 4 消息历史**
打开保存在本浏览器中的消息，带“清除历史”（«Очистить историю»）和“关闭”（«Закрыть»）按钮。
- 5 主题**
在浅色和深色主题之间切换页面。
- 6 状态行**
Samovar 的当前状态：正在执行的行的类型和编号、剩余时间、已记录的酒心温度、暂停、升温、稳定。旁边是行号和进度条。
- 7 取酒**
已取体积、实际取酒速度和当前行的进度。
- 8 命令按钮**
“开启/关闭加热”（«Включить/Выключить нагрев»）、“暂停/继续”（«Поставить на паузу/Продолжить»）、“开始取酒/下一步”（«Начать отбор/Далее»）、“重置过程”（«Сбросить процесс»），以及指向“图表”（«График»）和“设置”（«Настройки»）的链接。
- 9 温度**
蒸汽、塔节、蒸馏釜、水和 TCA 传感器的读数。颜色在设置 → 温度中设置。
- 10 杂质检测器**
杂质穿透检测器的状态和蒸汽温度趋势。在设置中启用检测器后才显示此卡片。
- 11 装置示意图**
蒸馏釜、塔、分凝器、TCA、取酒单元和接酒瓶，并标出当前温度、压力和加热。当前行的接酒瓶高亮显示。
- 12 右栏**
“当前行”（«Текущая строка»）、“调压器”（«Регулятор»）、“压力”（«Давление»）和“系统”（«Система»）卡片——见下一张截图。
- 13 详细说明**
阶段、完成条件、等待及如何继续、下一行。只有发送新阶段数据的固件才有此区块。

显示的内容

版本和模式——固件版本显示在页眉中“Samovar”（«Самовар»）字样旁边。页眉的第一个选项卡以当前模式命名（精馏时为“精馏”（«Ректификация»））。模式在[设置 → 基本](#)中选择：精馏（«Ректификация»）、蒸馏（«Дистилляция»）、啤酒（«Пиво»）、醪塔（«Бражная колонна»）、连续醪塔（«Непрерывная бражная колонна»）、奶酪制作（«Сыроварение»）。

连接和时间——绿色指示灯表示页面正在接收 Samovar 的数据。如果连续几次请求都没有响应，指示灯变红，并出现消息“连接中断！”（«Обрыв связи!»）。旁边是当前时间和“运行时间”（«Работа»）——Samovar 开机以来的时间。

状态行——当前状态：“已关闭”（«Выключено»）、“塔升温”（«Разгон колонны»）、“升温完成。稳定/全回流运行”（«Разгон завершен. Стабилизация/Работа на себя»）、“稳定完成/全回流运行”（«Стабилизация завершена/Работа на себя»）、正在执行的程序行的类型和编号（例如“酒心；程序 №2”（«Тело; Прг №2»））、“暂停”（«Пауза»）、“程序执行完毕”（«Выполнение программы завершено»）。取酒期间，状态中会加上“剩余：”（«Осталось:»）——到本行结束和整个程序结束的预计时间；记录酒心温度后，还会显示“酒心蒸汽温度”（«Т тела пар»）和“酒心塔节温度”（«Т тела царга»）。旁边显示行号（“第 2 行，共 4 行”（«строка 2 из 4»））、本行进度条和取酒汇总：“已取”（«Отобрано»）、“速度”（«Скорость»）、“进度”（«Прогресс»）。

详细说明——在发送新数据的固件上，状态下方显示阶段、当前行、其完成条件和目标、等待原因及如何继续，以及下一行。如果显示“详细说明不可用”（«Подробное объяснение недоступно»），说明固件没有发送新数据；常规读数和状态仍然可用。

关键数值——仅用于窄屏（智能手机）的卡片：“Δ 蒸汽 – 塔节”（«Δ пар – царга»）、“蒸汽与设定值之差”（«Пар к уставке»）（蒸汽温度与已记录的酒心取酒温度的偏差）、“蒸馏釜”（«Куб»）以及简要的杂质检测器状态。宽屏上这些数值显示在“传感器”和“杂质检测器”卡片中。

传感器——温度“蒸汽”（«Пар»）、“塔节”（«Царга»）、“Δ 蒸汽 – 塔节”（«Δ пар – царга»）、“蒸馏釜”（«Куб»）、“水”（«Вода»）、“TCA”（«TCA»）。每个传感器的颜色在[设置 → 温度](#)中设置。记录酒心温度后，蒸汽和塔节旁会显示设定值。读数为零或明显错误时，必须在启动前到设置 → 温度中排除。

杂质检测器——如果在设置中启用了检测器，就会显示这张卡片。它显示状态（“稳定，监视中”（«Стабильно, слежу»）、“温度上升：降低速度”（«Рост температуры: снижаю скорость»）、“穿透（暂停）”（«ПРОСКОК (ПАУЗА)»），酒头和酒尾阶段为“观察”（«наблюдение»），行启动或继续后为“行启动后暂停”（«Пауза после старта строки»）、“等待蒸汽稳定”（«Жду стабилизации пара»）、“正在积累历史”（«Набираю историю»）以及蒸汽温度趋势（°C/min）。在酒头和酒尾行上，检测器只做观察：不降低速度，也不暂停取酒。详见[取酒心](#)。

装置示意图——蒸馏釜、塔、分凝器、TCA、取酒单元和接酒瓶。示意图上标注当前温度、蒸馏釜内或启动时的压力，以及加热电压和功率。运行时，冷却水流、加热、取酒泵和产品液滴以动画显示；当前程序行接酒的瓶子会高亮。暂停时，蒸馏釜上出现“暂停”（«ПАУЗА»）字样。如果在[泵](#)选项卡上启用了“在巴氏段上方使用 I2C 泵”（«Использовать I2C-насос над ЦП»），并且启动时找到了泵板，塔的上方会画出巴氏段（«ЦП»），其左侧是带单独酒头取酒管路的第二个泵；泵工作时以动画显示。

右栏：当前行、调压器、压力、系统

ТЕКУЩАЯ СТРОКА · 2 ИЗ 4

Тело ёмкость 2

35 % · объём строки 500 мл

Скорость

Дельта

0.5 л/ч

-0.25 °C

Напряжение строки

Время строки

100 В

1 ч 00 мин

СЛЕДУЮЩАЯ 3 · Предзахлёб · 3000 мл · 1.2 л/ч · ёмк. 3

РЕГУЛЯТОР

Текущее напр.:

Целевое напр.:

180.5 V

180.0 V

Режим

Мощность

1

1450 Вт

Напряжение регулятора:

Установить напряжение

Скорость, л/ч

Установить

Зафиксировать T отбора тела

ДАВЛЕНИЕ

Атмосферное

В колонне:

752.3 мм рт.ст.

1.2 мм рт.ст.

СИСТЕМА

Неар=152000; BME temp=24.5; RSSI = -58; Свободно: 300000; ШИМ насоса воды: 0

“当前行”（«Текущая строка»）、“调压器”（«Регулятор»）、“压力”（«Давление»）、“系统”（«Система»）卡片

- 1 **当前行**
行类型、容器、进度、体积、速度、转换温度、电压和本行时间、下一行。
- 2 **调压器**
当前和目标电压（功率）、调压器模式和以瓦为单位的功率。
- 3 **调压器电压**
手动设置电压（功率）：输入数值后点击“设置电压”（«Установить напряжение»）。
- 4 **速度，升/时**
手动修改当前行的取酒速度。输入数值后点击“设置”（«Установить»）。
- 5 **记录酒心取酒温度**
把当前蒸汽和塔节温度作为酒心取酒温度。
- 6 **压力**
大气压力，以及启动时的压力（“启动时”（«В начале»））或蒸馏釜内的压力（“塔内”（«В колонне»）），后者需连接压力传感器。
- 7 **系统**
空闲内存、压力传感器温度、Wi-Fi 信号强度、文件系统剩余空间、水泵的 PWM 和 I2C 泵的状态。

当前行——行类型（酒头、酒心、预液泛、暂停、酒尾、Lua）、容器编号、进度条、本行体积、速度、转换温度、本行电压（功率）、本行预计时间以及下一行的简要说明。程序本身在“程序”（«Программа»）选项卡上编辑，见[精馏模式的取酒程序](#)。

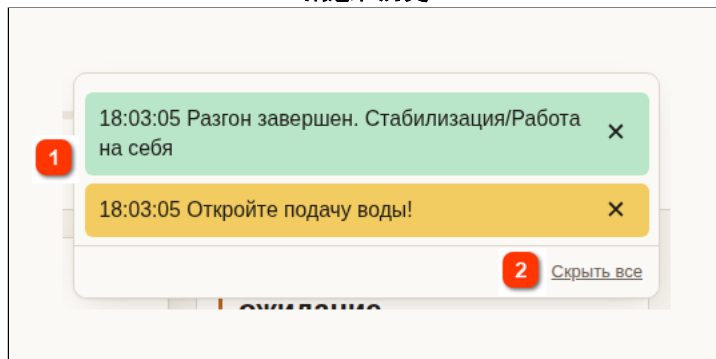
操作——按钮下方可能显示当前生效命令的说明：设备、设定值、实际执行值，以及命令来源——程序、手动输入、自动化或保护功能。

调压器——当前电压和目标电压（如果调压器按功率工作，则为功率）、调压器模式和以瓦为单位的功率。可以在“调压器电压”（«Напряжение регулятора»）字段中手动输入电压，然后点击“设置电压”（«Установить напряжение»），见[手动设置调压器电压](#)。在“速度，升/时”（«Скорость, л/ч»）字段中，可以用“设置”（«Установить»）按钮修改当前行的取酒速度，见[手动设置取酒速度](#)。“记录酒心取酒温度”（«Зафиксировать Т отбора тела»）按钮把当前蒸汽和塔节温度记录为酒心取酒温度。如果固件中不使用调压器，该卡片隐藏。

压力——BME/BMP 传感器测得的大气压力以及第二个数值：“启动时”（«В начале»）——启动瞬间的压力；如果连接了蒸馏釜压力传感器，则为“塔内”（«В колонне»）。

系统——Heap（控制器空闲内存）、BME temp（压力传感器温度）、RSSI（Wi-Fi 信号强度）、“空闲”（«Свободно»）（文件系统剩余空间），连接相应设备时还显示水泵的 PWM 和外部 I2C 泵的状态。

消息和历史



系统消息

1 系统消息

新通知及其出现时间。点叉号关闭单条消息；紧急消息需确认后才能关闭。

2 全部隐藏

把所有消息从屏幕上移除。它们仍保留在历史中（≡ 按钮）。

新通知出现在页面右上角。每条消息点叉号关闭；紧急消息只有确认后才能关闭。“全部隐藏”（«Скрыть все»）按钮把所有消息从屏幕上移除。页眉中的 ≡ 按钮打开历史：保存在本浏览器中的消息，包括页面刷新前的消息（保留最近 500 条）。“清除历史”（«Очистить историю»）按钮把它们从浏览器中删除，“关闭”（«Закрыть»）收起历史。Samovar 的消息列表见[系统消息](#)页面。

主要按钮

开启加热 / 关闭加热（«Включить нагрев / Выключить нагрев»）——为所选模式启动或停止加热。关闭时需要确认。如果设置中没有分配蒸汽、水和蒸馏釜传感器，加热不会开启，屏幕上出现消息“传感器未分配”（«Датчики не назначены»）。

开始取酒 / 下一步（«Начать отбор / Далее»）——启动精馏程序。加热关闭时不执行该命令。如果塔仍在升温或稳定，页面会要求确认启动。取酒期间该按钮名为“下一步”，确认后让 Samovar 转到下一个程序行。

暂停 / 继续（«Поставить на паузу / Продолжить»）——停止和恢复取酒，见[暂停](#)。

重置过程（«Сбросить процесс»）——确认后，用该模式的标准命令结束当前过程并关闭加热。

图表（«График»）——打开当前的本地图表，见[本地日志和图表](#)。**设置**（«Настройки»）——打开 Samovar 的参数。

选项卡

设置：基本

Samovar 7.00: 杂质检测器由一个“使用检测器”（«Использовать детектор»）开关启用，只在精馏中工作。在酒头阶段它只做观察；温度临界上升时暂停取酒。平稳降低速度由另一个“温度上升时降低取酒速度”（«Снижать скорость отбора при росте температуры»）开关控制。

在此选项卡中选择运行模式和通用限值。修改后，点击页面底部的“**保存**”（«Сохранить»）。切换模式无需重启 ESP32，但只能在加热关闭时进行：过程运行期间，Samovar 不接受表单，并在“运行模式”（«Режим работы»）字段处显示错误。

Error: I2CStepper недоступен: HTTP 404 — File not found ✕

Настройки

Основные Температура Насос Пиво Сыр НБК Прочие I2CStepper

Режим работы **Ректификация**

Ректификация, дистилляция и БК до температуры в кубе: 0 °C

Время для контроля завершения процесса дистилляции: 0 мин

- ☐ Использовать корректировку давления
- ☐ Использовать датчик уровня флегмы при ректификации
- ☐ Не показывать схему процесса

⚙️ Параметры оборудования (колонны)

Диаметр: 1.5 дюймов

Высота насадки: 0.50 метров

Плотность насадки: 80 % (0-100)

Детектор проскока примесей (ректификация)

- ☒ Использовать детектор
 - ☐ Снижать скорость отбора при посто температуры
- Шаг снижения скорости: 0 % (0 = по умолчанию 5, максимум 20)

Давление, при котором включится авария: 0 мм. Рт. ст.

Нулевая точка МРХ: 0

Отсчётов МРХ на 1 мм рт. ст.: 0

Настройки зуммера

- ☐ Использовать зуммер для нотификации
- ☐ Использовать зуммер для нотификации в браузере
- ☐ Включать зуммер при смене программы

Самотестирование

Внимание: включится подача воды (насос или клапан), запустится перистальтический насос отбора, сервопривод пройдёт 5 положений ёмкостей.

Сохранить

На главную

Редактор

设置 → 基本 («Настройки → Основные»)

1

运行模式 («Режим работы»)

模式选择：“精馏” («Ректификация»)、“蒸馏” («Дистилляция»)、“啤酒” («Пиво»)、“醪塔” («Бражная колонна»)、“连续醪塔” («Непрерывная бражная колонна»)、“奶酪制作” («Сыроварение») 。只能在加热关闭时更改。

2

过程结束温度 («Температура окончания процесса»)

蒸馏、БК 以及精馏程序结束后的取酒在此蒸馏釜温度下结束。

- 3 蒸馏结束判定时间 («Время для контроля завершения дистилляции»)**
蒸馏釜温度不再上升的分钟数，达到后蒸馏或 ВК 结束。0 表示不监控。
- 4 压力修正 («Корректировка давления»)**
按大气压修正传感器读数。
- 5 回流液位传感器 («Датчик уровня флегмы»)**
只在带回流液位传感器的编译版本中起作用。
- 6 塔参数 («Параметры колонны»)**
直径、填料高度和填料密度——供程序计算器使用。
- 7 使用检测器 («Использовать детектор»)**
杂质穿透检测器的总开关。
- 8 降低取酒速度 («Снижать скорость отбора»)**
蒸汽温度上升时平稳降低速度；只与检测器配合工作。
- 9 降速步长 («Шаг снижения скорости»)**
单次速度变化的百分比 (0 = 5%，最大 20%)。
- 10 报警压力 («Давление аварии»)**
蒸馏釜内超压 (mmHg)，达到时 Samovar 进入报警状态。0 表示关闭。
- 11 蜂鸣器 («Зуммер»)**
设备上的声音、浏览器中的声音、程序行切换时的提示音。
- 12 自检 («Самотестирование»)**
检查供水、取酒泵和舵机。只能在加热关闭时进行。
- 13 保存 («Сохранить»)**
写入所有选项卡的设置。旁边是“返回主页” («На главную») (不保存离开) 和“编辑器” («Редактор») (文件管理器)。

模式与过程结束

运行模式 («Режим работы») ——“精馏” («Ректификация»)、“蒸馏” («Дистилляция»)、“啤酒” («Пиво»)、“醪塔” («Бражная колонна»)、“连续醪塔” («Непрерывная бражная колонна») 或“奶酪制作” («Сыроварение»)。“低温慢煮” («Су-вид») 和 Lua 两项在列表中隐藏，无法通过列表选择。如果您的固件中没有编译某个模式，Samovar 会在保存时提示。

精馏、蒸馏和 ВК 的结束釜温 («Ректификация, дистилляция и ВК до температуры в кубе») ——正常结束温度。在蒸馏和 ВК 中，过程在此温度结束；在精馏中，所有程序行执行完后，Samovar 会继续取酒，直到蒸馏釜达到此温度。如果关闭了压力修正，在高海拔地区沸点较低，设得过的限值可能永远达不到。

蒸馏结束判定时间 («Время для контроля завершения процесса дистилляции») ——蒸馏釜温度 (高于 90°C) 需要保持几乎不变 (上升不足 0.1°C) 多少分钟后才结束。适用于蒸馏和 ВК。0 表示关闭此判据，最大值为 255。

使用压力修正 («Использовать корректировку давления») ——按大气压修正蒸汽、塔身、蒸馏釜和 TCA 传感器的读数 (相对 760 每偏差 1 mmHg 修正 0.037°C)。水温传感器不做修正。

精馏时使用回流液位传感器 («Использовать датчик уровня флегмы при ректификации») ——如果编译版本中有此传感器，则把它纳入精馏算法。标准编译版本在编译时禁用了回流液位传感器，此开关不起作用。

设备 (塔) 参数

直径 («Диаметр»)、**填料高度** («Высота насадки») 和 **填料密度** («Плотность насадки») 描述您的塔。它们供 [程序计算器](#) (“计算” («Расчёт») 选项卡) 使用，并写入过程日志。

杂质穿透检测器 (精馏)

使用检测器 («Использовать детектор») ——杂质检测器的总开关。检测器在精馏程序的“预液泛” («Предзахлёб»)、“酒心” («Тело») 和“酒尾” («Хвосты») 行中跟踪蒸汽温度的变化趋势；温度危险上升时暂停取酒，温度回落自动恢复。检测器状态显示在 [主页面的](#)“杂质检测器” («Детектор примесей») 区块中。

温度上升时降低取酒速度 («Снижать скорость отбора при росте температуры») ——只与检测器配合工作。出现温度上升警告时，取酒速度逐级降低，直到趋势平稳。温度临界上升时的暂停与此开关无关，总会执行。

降速步长 («Шаг снижения скорости») ——单次速度变化的百分比。0 表示默认的 5%，最大为 20%。

触发报警的压力 («Давление, при котором включится авария») ——接有压力传感器时，蒸馏釜内超压的上限，单位 mmHg。达到此压力时，Samovar 进入报警模式并关闭加热。0 表示关闭监控。数值设错可能导致误报警，或在应当停机时不停机。如果固件中启用了 MPX5010DP 传感器，下方会出现它的两个校准字段：“MPX 零点” («Нулевая точка MPX») 和“每 1 mmHg 的 MPX 计数” («Отсчётов MPX на 1 мм рт.ст.»)。

蜂鸣器与自检

蜂鸣器参数分别启用设备上的声音、浏览器中的声音以及程序行切换时的提示音。第一个参数关闭时，蜂鸣器完全禁用。

“**自检**” («Самотестирование») 按钮用于检查执行机构：打开供水（水泵启动或阀门打开），启动取酒蠕动泵，舵机依次走过 5 个接酒容器位置。启动前需要确认操作。测试持续约 15 秒，以消息“自检完成。” («Самотестирование закончено.») 结束。加热开启或过程运行时，自检不会启动。按下按钮前，请断开所有危险负载，并把软管引入安全的容器中。

页面底部有三个按钮：“**保存**”写入所有设置选项卡并返回主页面；“**返回主页**” («На главную») 不保存直接离开；“**编辑器**” («Редактор») 打开固件文件管理器。如果表单有未保存的修改，离开前 Samovar 会请求确认。如果某个字段有错误，则不会保存：会打开该字段所在的选项卡，并显示消息“设置未保存。请检查字段：...” («Настройки не сохранены. Проверьте поля: ...»)。

温度

选择和设置温度传感器

Samovar 使用 DS18B20 数字传感器。每个传感器都有自己唯一的地址，即标识符。为了让 Samovar 从正确的传感器读取数据，必须在“传感器地址” («Адрес датчика») 下拉菜单中选择它。安装位置共有五个：分凝器前的蒸汽、塔节、水、蒸馏釜和 TCA（通大气管）。Samovar 在总线上最多可找到六个传感器。

警告。

如果设置中指定了某个传感器，但在加热开启时它的数据不再到达，Samovar 会在主屏幕上定期显示“蒸汽温度传感器错误！” («Ошибка датчика температуры пара!») 之类的消息，无论该传感器是用于过程控制还是仅用于显示。加热关闭时不显示这些消息。

[重新扫描](#)传感器只能在没有运行任何过程（加热关闭）时进行。

Адрес датчика пара перед дефлегматором

Корректировка датчика пара перед дефлегматором: 0

Вылет за уставку датчика пара перед дефлегматором: 0

Задержка отбора при выходе Т пара за уставку: 0 сек

设置 → 温度：分凝器前的蒸汽传感器 («Настройки → Температура: датчик пара перед дефлегматором»)

- 1 传感器地址 («Адрес датчика»)**
从下拉菜单中选择所需传感器的地址。列表其中包括扫描时找到的传感器和 NONE 项（此位置未安装传感器）。
- 2 修正 («Корректировка»)**
可以修正传感器读数，正值和负值都可以设置。修正值加到传感器的原始读数上。如果在“基本” («Основные») 选项卡上启用了气压修正，除水传感器外的所有传感器还会额外加上大气压修正。
- 3 超出设定值 («Вылет за уставку»)**
在“精馏” («Ректификация») 模式中执行预液泛和酒心程序行时使用。
Samovar 完成取酒头并转到取酒心程序行时，会记住蒸汽温度和塔节温度。如果传感器的实际温度偏离记录值超过设定值，Samovar 会暂停取酒，显示“按蒸汽温度暂停” («Пауза по Т пара») 消息，并等待温度回到设定范围内。值为 0 时关闭对该传感器的监控。
- 4 取酒延迟 («Задержка отбора»)**
温度超出设定值后，取酒泵暂停的时间（秒）。这段时间过后，Samovar 检查温度：如果已回到范围内，取酒继续并显示“自动暂停后继续取酒” («Продолжаем отбор после автоматической паузы») 消息；如果没有，延迟重新开始计时。

塔节传感器的设置与蒸汽传感器的设置相同。

注意。

随时可以用[主页](#)上“调节器” («Регулятор») 卡片中的“记录酒心取酒温度” («Зафиксировать Т отбора тела») 按钮设置新的蒸汽温度和塔节温度值。

水传感器的设置与前面的类似，但有一个重要区别。

Адрес датчика воды - ▾

Корректировка датчика воды: 0 °C

Уставка датчика воды: 0

Задержка срабатывания при выходе Т воды за уставку: 0 сек

设置 → 温度：水传感器 («Настройки → Температура: датчик воды»)

- 1** 水传感器设定值 («Уставка датчика воды»)
 水传感器设定值字段填写冷却供水泵需要保持的实际值。在 Samovar 运行期间随时可以更改。如果保留 0，Samovar 使用 48°C。如果用旁通阀代替水泵，超过设定值 1°C 时阀门打开，温度降到比设定值低 1°C 时阀门关闭。关闭加热后，当水温比设定值低 20°C（不早于 3 分钟后）或蒸馏釜冷却后，供水关闭。

- 2** 触发延迟 («Задержка срабатывания»)
 当前版本中不使用。延迟 = 0

蒸馏釜传感器的设置与前面的类似，但也有自己的特点。

Адрес датчика в кубе - ▾

Корректировка датчика в кубе: 0 °C

Вылет за уставку датчика в кубе: 0 °C

Задержка срабатывания при выходе Т в кубе за уставку: 0 сек

设置 → 温度：釜内传感器 («Настройки → Температура: датчик в кубе»)

蒸馏釜传感器用于按温度结束过程（“[精馏、蒸馏和 ВК 进行到釜内温度](#)” («Ректификация, дистилляция и ВК до температуры в кубе»)），用于在蒸馏和 ВК（[蒸馏](#)）中监测温度平台，以及在啤酒、低温慢煮和奶酪制作模式中保持温度。在啤酒和奶酪制作模式中，**釜内传感器超出设定值**（«Вылет за уставку датчика в кубе»）字段设置程序行的温度容差：0.3° 到 3°；小于 0.3° 的值（包括 0）按 0.3° 处理，见[糖化程序](#)和[奶酪制作程序](#)。如果在程序行中选择了其他传感器，其超出设定值的作用方式相同；对于蒸汽和塔节传感器，这就是精馏中取酒心时使用的同一个值。**触发延迟** («Задержка срабатывания») 字段在当前版本中不使用：该值会被保存，但没有任何模式读取它。

注意。

“温度” («Температура») 选项卡上的设置对所有模式都相同。读数修正值和传感器地址不会随工作模式一起切换。

Адрес датчика в ТСА - ▾

Корректировка датчика в ТСА: 0 °C

Уставка датчика в ТСА: 0

Задержка срабатывания при выходе Т в ТСА за уставку: 0 сек

设置 → 温度：TCA 传感器 («Настройки → Температура: датчик в ТСА»)

TCA 传感器设定值 («Уставка датчика в ТСА»)

1

默认设定值为 43°C；如果保留 0，Samovar 使用 43°C。

在精馏模式中，TCA 过热阈值等于设定值，但不低于 45°C。在蒸馏和 NBK 模式中直接使用设定值，没有 45°C 的下限。

加热开启且 TCA 温度高于该阈值时，Samovar 每次超限只显示一次 TCA 温度过高警告。TCA 冷却到比阈值低 2°C 后，才允许再次警告。

超过同一阈值时，如果 TCA 比水热且冷却泵已打开，冷却泵会加大供水量。

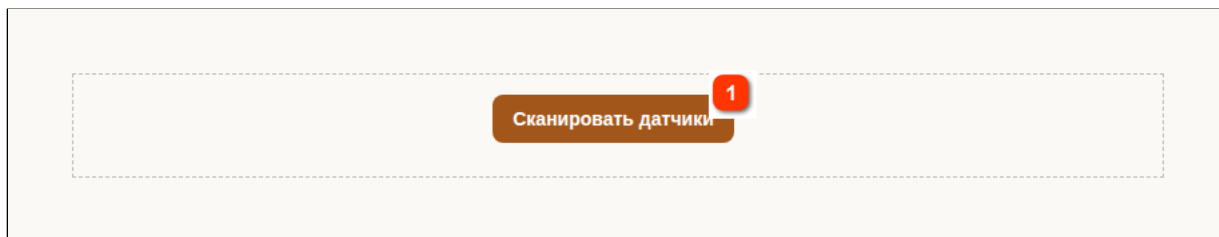
紧急阈值与设定值无关：当 TCA 温度达到或超过 MAX_ACP_TEMP（默认 75°C）时，Samovar 关闭加热。

例如，设定值为 47°C 时，超过 47°C 会触发警告，并在满足上述条件时加大水泵供水；冷却到 45°C 以下后可再次警告；达到 75°C 及以上时紧急关闭加热。

2

触发延迟 («Задержка срабатывания»)

当前版本中不使用。



1

扫描传感器 («Сканировать датчики»)

点击按钮后，Samovar 重新轮询总线，确定哪些传感器存在、哪些不存在，然后设置页面重新加载，显示最新的地址列表。

温度传感器可以“热插拔”，无需重启 Samovar。但扫描只在加热关闭时进行：如果过程正在运行，Samovar 会回复“传感器扫描被拒绝：过程正在运行。” («Сканирование датчиков отклонено: процесс активен.»)

可以逐个连接传感器并保存到设置中。连接一个传感器，点击“扫描传感器” («Сканировать датчики»)，选择它的位置，点击“保存” («Сохранить»)，返回设置，连接下一个传感器，扫描，选择它的位置，点击“保存”，依此类推。

关键是每添加一个传感器并指定位置后都要保存设置。

要保存所选设置，请点击“保存”；要不保存离开，请点击“返回主页” («На главную»)。如果表单中有未保存的更改，Samovar 会在离开页面前要求确认。更改传感器地址后，它之前的读数会被清除并重新采集。

警告。

每个安装位置的传感器地址必须唯一。选择时请确保传感器地址不重复。这是 Samovar 正常工作所必需的。不过，不禁止把同一个标识符同时用于多个地址。

设定值、超出设定值、温度修正等设置不会随传感器标识符一起转移，它们属于具体的安装位置。

注意。

如果过程开始时就没有某个温度传感器（即使设置中指定了它的地址），这不成问题。如果传感器在过程中掉线，Samovar 会报告。

泵

蠕动泵电机设置。

泵只需设置一个参数：取酒 1 ml 所需的步进电机步数。

Количество шагов шагового двигателя: 100 мл.

☐ Использовать I2C-насос над ЦП

Скорость отбора над ЦП: 0 л/ч

Калибровка насоса

设置 → 泵 («Настройки → Насос»)

1

步进电机步数

每取酒 1 ml 的电机步数。可手动填写，也可由校准结果自动填入。

2

泵校准

打开“步进电机校准” («Калибровка шагового двигателя») 窗口。

该参数可以手动输入，也可以通过校准泵获得：取酒 100 ml，Samovar 统计取出 100 ml 时电机走过的步数，除以 100，结果显示在“步进电机步数：_____ /ml.” («Количество шагов шагового двигателя: _____ /мл.») 一行中。

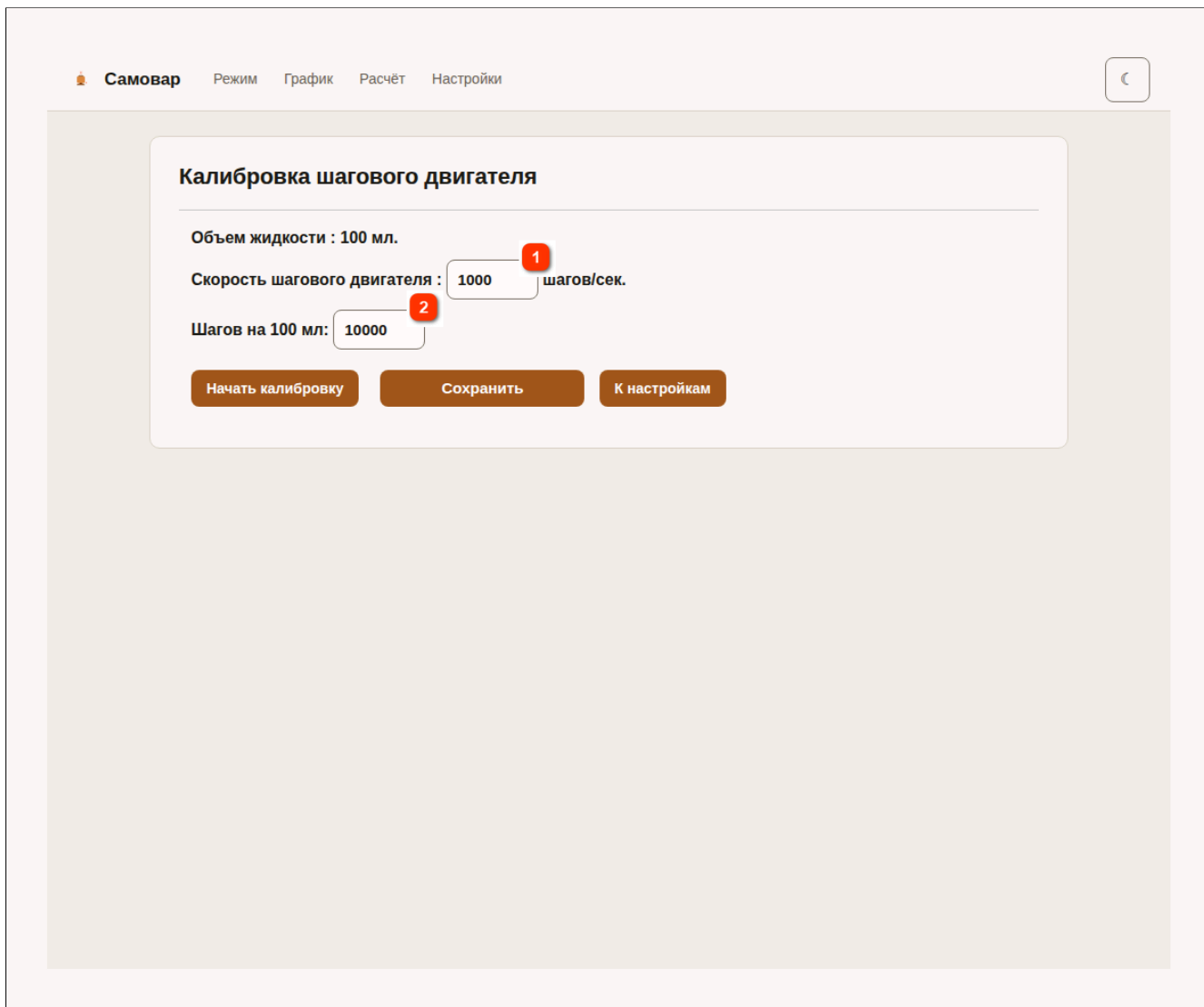
Samovar 7.00: 如果通过 I2C 总线给 Samovar 连接了外部泵 ([I2CStepper](#) 模块)，该选项卡上会出现第二个区块：“在巴氏杀菌段上方使用 I2C 泵” («Использовать I2C-насос над ЦП») 开关及其流量 (l/h)。没有 I2C 泵时，该区块隐藏。I2C 泵每毫升的步数保存在该模块自身的存储器中 (“设置” («Настройки») → I2CStepper, “步数/ml” («Шагов/мл») 字段)，并通过 I2CStepper 页面上的“外部泵校准” («Калибровка внешнего насоса») 按钮校准。

泵校准

可以在实际蒸馏过程中校准，但必须在程序的“开始取酒” («Начать отбор») 命令之前进行。

出现“稳定完成，塔运行稳定。” («Стабилизация завершена, колонна работает стабильно.») 消息后，等待约 30-40 分钟，将泵的出口引入量杯，然后点击“泵校准” («Калибровка насоса») 按钮。

将打开如下窗口：“步进电机校准” («Калибровка шагового двигателя»)



步进电机校准 («Калибровка шагового двигателя»)

1

电机转速

NEMA17 电机的最佳转速，单位为步/秒（1 至 8000）。

2

每 100 ml 步数

校准后得到的取酒 100 ml 所需步数。可以手动修改该值并点击“保存”（«Сохранить»）（数值必须是 100 的倍数）。“返回设置”（«К настройкам»）按钮返回设置页面。

点击“开始校准”（«Начать калибровку»）。泵开始运转，按钮文字变为“记录 100 ml”（«Зафиксировать 100 мл»）。取满 100 ml 后，点击“记录 100 ml”按钮。Samovar 计算并记住取酒 1 ml 所需的步数。结果立即保存到设置中，所得数值显示在“每 100 ml 步数”（«Шагов на 100 мл»）字段中。校准期间，主页面的状态行显示“校准”（«Калибровка»）。校准结束后，Samovar 回到“全回流运行”（«Работа на себя»）状态。

之后即可用“开始取酒”命令启动取酒程序，按常规方式继续蒸馏。

注意。

取酒程序运行期间不能进行校准。

也可以不进行蒸馏而单独校准。

向泵的进口供应 100 ml 以上的液体。建议使用酒精、粗酒或以前蒸馏留下的酒头。

注意。

取酒 1 ml 所需的步数会在保存设置时写入 samovarbackup.txt 文件，但恢复设置时不会加载，也不会被覆盖。这是有意为之。

如需手动恢复以前的数值，请在 samovarbackup.txt 文件中找到“StepperStepMl”参数，冒号后引号内的数字就是步数。将其填入“设置”（«Настройки»）->“泵”（«Насос»）选项卡上的“步进电机步数：_____ /ml.”（«Количество шагов шагового двигателя: _____ /мл.»）字段（第一张截图）。

提示。

如果装有编码器，校准期间可以用编码器调节步进电机转速；达到 100 ml 时，按下编码器即可结束校准。

重要。

强烈建议定期用中性硅脂润滑泵管和滚轮。润滑不仅影响泵零件的寿命，也影响取酒精度。

啤酒

此选项卡中的数值用于啤酒模式。例外：“沸腾后 BK 功率”（«Мощность БК после закипания»）字段属于**醪塔**模式，“煮沸时使用辅助加热管”（«Использовать разгонный тэн при кипении»）复选框在蒸馏模式下也有效。

设置 → 啤酒

1

糖化顺序

糖化设备布局，而不是配方。“All-in-one / BIAB”——按蒸馏釜传感器（糖化醪）加热，搅拌器按程序设定工作。“HERMS”——按水温传感器（蒸馏釜）进行 PID 控制，泵持续让糖化醪循环通过盘管，蒸馏釜传感器监测糖化醪。“RIMS”——按塔节传感器（回流管路）进行 PID 控制，泵持续运行；在真正的 RIMS 上无流动加热是危险的。

2

加热用 PID 调节器系数

Kp、Ki、Kd。通过自动标定（“自动整定”（«Автотюнинг»）程序行）或手动确定。

3

保温模式下的调压器电压/功率

在温度停留阶段，调压器施加到加热器上的最大电压（功率）。

4

加热和煮沸模式下的调压器电压/功率

加热到设定温度时以及在“煮沸”（«Кипячение»）行上的电压（功率）。

5

沸腾后 BK 功率

蒸馏釜沸腾后醪塔模式的工作功率。

6

煮沸时使用辅助加热管

不勾选时，在“煮沸”行上不开启辅助加热管。在蒸馏模式下也有效。

PID 调节器系数

可以对 PID 调节器系数进行自动标定。
为此，在蒸馏釜中加水并运行以下程序：

Программа затирки:

Варочный порядок из настроек: All-in-one / VIAВ. ПИД по кубу (затор), мешалка как в универсальной программе.

№	Тип программы	Темп	Время	Мешалка	Датчик темп.	
01	Пауза	50	1	0^0^0^0^0	куб (затор)	+ -
02	Автотюнинг	50	0	0^0^0^0^0	куб (затор)	+ -

► Описание программы затирки:

► Описание программы затирки в текстовом формате для сохранения в облаке:

Сохранить программу

Загрузить программу

Установить программу

Очистить программу

Загрузить рецепт из XML

PID 自动标定程序：一行“暂停”（«Пауза»）和一行温度相同的“自动整定”（«Автотюнинг»）

Samovar 加热到指定温度后开始自动标定。标定完成后，Samovar 把新系数写入设置，立即开始使用，并转到下一程序行。无需重启 Samovar。如果系数未能保存，会出现消息“PID 自动整定未保存”（«PID autotune не сохранён»）。标定过程可能需要几个小时，但控温精度应该会提高。也可以在设置中手动调整 PID 调节器系数。

警告。 没有专门知识时，不要随意修改 PID 调节器系数。

保温模式下的调压器电压

如果你认为温度停留阶段保温不正确，可以修改保温模式下的调压器电压。

Samovar 在温度停留阶段通过 PWM 保持温度。

“保温模式下的调压器电压/功率”（«Напр./мощность регулятора в режиме поддержания темп.»）字段中的值（例如 100 伏）是保温模式下可以施加到加热器上的最大电压。

也就是说，在保温模式（温度停留）下，施加到加热器上的电压不超过 100 伏。施加这个电压的频率由 Samovar 的逻辑决定。

实际示例：

如果在寒冷的房间里使用一个没有保温的大糖化锅，而主加热器功率又小，那么在温度停留阶段糖化醪温度可能会下降。这时可能出现以下情况：Samovar 不得不持续给加热器施加 100 伏电压，但即便如此，糖化醪温度仍在下降。

当温度下降超过 Samovar_ini.h 文件中下面这一行设定的值（单位 C°）时：

```
#define HEAT_DELTA 1
```

为了回到设定温度，Samovar 开始通过调压器向负载施加“加热和煮沸模式下的调压器电压/功率”（«Напр./мощность регулятора в режиме нагрева и кипения»）字段中的电压（加热）。

为避免这种情况，请提高保温模式下的调压器电压，例如提高到 120 伏。

再举一个例子。

糖化锅中的温度起初超过（冲过）糖化程序行中设定的停留温度。如果使用的加热器相对锅的容量功率过大，就可能出现这种情况。这时保温模式下的调压器电压保持 100 不变，而把下面这一行的值：

```
#define HEAT_DELTA 1
```

增大几个单位。

主加热器和附加加热器的算法说明见[搅拌机、泵、温度](#)一节

加热和煮沸模式下的调压器电压

可以为使用“煮沸”（«Кипячение»）程序行的情况设置较低的调压器电压。酿造某些啤酒风格时，或在小糖化锅上使用大功率加热管时需要这样做。

注意。

此行设定的值也用于加热到停留温度的阶段（糖化醪温度比设定值低 HEAT_DELTA 以上时），因此可以为整个啤酒模式设定并快速修改主加热器的功率。

沸腾后 BK 功率

[醪塔](#)模式下蒸馏釜沸腾后加热器的工作功率。如果 BK 程序第一行设定了自己的功率，则使用该功率。低于固件所用最小值的数值不被接受。

煮沸时使用辅助加热管

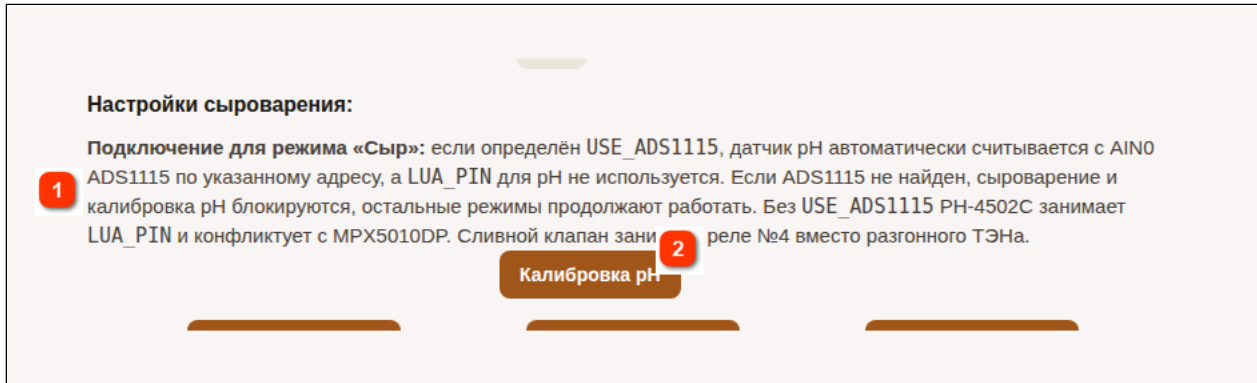
可以在使用“煮沸”程序行时禁止使用辅助加热管。在酿造过程的其他程序行中，辅助加热管仍按设置工作。该复选框在蒸馏模式下也有效：如果不勾选，蒸馏釜一开始沸腾就立即关闭辅助加热管。

返回 [“啤酒”模式](#)

奶酪

[奶酪制作](#) 模式的设置。

设置 («Настройки») → 奶酪 («Сыр») 选项卡中只有接线提示和 pH 传感器校准的链接。奶酪制作程序本身在该模式的主页面上设置 (见 [奶酪制作程序](#))。



设置 → 奶酪 («Настройки → Сыр»)

1

“奶酪”模式的接线

提示奶酪制作使用哪些输入端和继电器，以及它们与哪些功能冲突。

2

pH 校准

打开用两种缓冲溶液校准 pH 传感器的页面。如果表中有未保存的更改，Samovar 会要求确认。

接线。 pH 传感器通过 ADS1115 连接，或连接到 LUA_PIN。电路图和引脚分配见技术章节 [pH 传感器及其连接方法](#)。排液阀连接到 4 号继电器；不要同时把阀门和辅助加热管接到这个继电器上。

警告。 使用 LUA_PIN 时需要进行电压匹配；PH-4502C 和 MPX5010DP 不能同时接到这个输入端。使用 ADS1115 时，LUA_PIN 输入端保持空闲。

4 号继电器的触发电平（低电平或高电平闭合）在 [其他](#) («Прочие») 选项卡的“继电器 4” («Реле 4») 字段中设置。排液阀遵循此设置。

奶酪制作中的冷却使用与啤酒模式相同的输出：冷却泵（标准版本）或阀门。此选项卡上没有单独的冷却设置。

pH 校准。 该按钮打开 [pH 校准](#) 页面：使用两种缓冲溶液（默认 7.00 和 4.00），Samovar 计算系数 k 和偏移量 d，并通过“保存校准” («Сохранить калибровку») 按钮保存。pH 读数在 0–14 范围内视为有效；如果传感器数据 5 秒内没有更新，该数值会被标记为过时。

注意。 pH 校准系数不属于设置表单，不会写入 samovarbackup.txt 文件 (见 [保存设置](#))。从文件恢复设置后无需重新进行 pH 校准，但也无法通过该文件把校准转移到另一个控制器上。

[连续醪塔 \(NBK\)](#) 模式的设置。

“设置 → NBK” («Настройки → НБК») 选项卡包含连续醪塔自动控制的参数：塔对变化的响应速度、自动控制以多大的步长调整功率和发酵醪进料量，以及判断液泛和结束的阈值。默认值适用于大多数塔；在最初几次运行、了解自己的塔的表现之后再修改。

Настройки НБК:

Инерция: 0

Дельта: 0

Шаг мощности: 0

Шаг подачи: 0

Т пара: 0

Давление захлёба: 0

Т завершения (барда): 0

8 ☐ Переключать поток сервоприводом при захлёбе

设置 → NBK («Настройки → НБК»)

- 1 **惯性 («Инерция»)**
功率或进料量改变后，塔达到新的稳定状态所需的时间，单位为秒。默认 180。自动控制的所有暂停（调整后的等待、液泛后的恢复）都以它为基准计算。值为 1 或更小时，NBK 拒绝启动。对于浓稠的谷物发酵醪和带预热的塔，请调大此值。
- 2 **偏差 («Дельта»)**
废醪（酒糟）温度相对基准值 T_n (°C) 的允许偏差，在此范围内不调整功率和进料量。默认 0.5。偏差超过三个偏差值时视为发酵醪溢流——进料量降低一个步长。
- 3 **功率步长 («Шаг мощности»)**
自动控制每次调整时功率改变多少瓦。默认 100。在“优化” («Оптимизация») 行中，每次迭代功率增加一个步长；在“工作” («Работа») 行中，液泛后功率降低步长的十分之一。
- 4 **进料步长 («Шаг подачи»)**
自动控制每次调整时发酵醪进料量改变多少升/小时。默认 0.5。在“优化”行中为整个步长，在“工作”行中为步长的十分之一，溢流时降低一个整步长。主页面上的手动调节进料按钮使用同样的步长。
- 5 **蒸汽温度 («Т пара»)**
蒸汽温度基准值，°C。默认 81；允许范围为 80 到 97。蒸汽温度达到 75°C 时认为升温 (H 行) 结束。界面中的提示称此值为上限，但自动控制把它用作工作蒸汽温度的下限。
- 6 **液泛压力 («Давление захлёба»)**
蒸馏釜压力传感器的阈值 (mmHg)，达到时记录液泛。默认 40。工作压力上限为该值的一半 (40 mm 时为 20 mm)；运行过程中，每次液泛后自动控制会自行降低上限。
- 7 **结束温度 (酒糟) («Т завершения (барда)»)**
蒸净酒糟的基准温度，°C。默认 98.5。在第一次“优化”迭代时，会根据传感器读数自动向上修正 (最高 102°C)。
- 8 **液泛时用舵机切换流向 («Переключать поток сервоприводом при захлёбе»)**
液泛时舵机转到位置 1——脏流。位置 0——净流——只有在完整的恢复暂停结束后、转入“工作”行时以及 NBK 结束时才恢复。默认关闭。

除本选项卡外，NBK 还需要在“其他” («Прочие») 选项卡中填写电网电压和加热管电阻：Samovar 用它们计算最大加热功率。液泛压力阈值只有在蒸馏釜上接有压力传感器时才起作用。

NBK 程序 (“升温” («Разгон»)、“手动调整” («Ручная настройка»)、“优化” («Оптимизация»)、“工作” («Работа») 各行) 在该模式的主页面上设置，见 [连续醪塔 \(NBK\)](#)。

Токен Blynk: 0

URL видео камеры: 0

Часовой пояс: 0

☐ Использовать проверку подключения регулятора

Сопротивление тэна для расчета мощности: 10.00000 Ом

Напряжение сети: 230.00 В

☐ Использовать контроль потока воды

Цвет для температур:

- Температура пара
- Температура датчика в царге
- Температура воды
- Температура датчика в кубе
- Температура датчика в ТСА

[цвета можно подобрать тут](#)

Настройки ур... для реле

Реле 1 LOW

Реле 2 LOW

Реле 3 LOW

Реле 4 LOW

Скачать файл настроек

Сбросить настройки WiFi

Загрузить файл настроек

Перезагрузить Самовар

设置 («Настройки») → 其他 («Прочие»)

- 1 **Blynk 令牌 («Токен Blynk»)**
在 samovar-tool.ru 注册后邮件中的令牌。日志和消息也通过这个令牌发送到云端。
- 2 **摄像头视频 URL («URL видео камеры»)**
Blynk 应用中视频组件使用的视频流地址。
- 3 **时区 («Часовой пояс»)**
相对格林尼治的时差，整数小时。
- 4 **使用调压器连接检查 («Использовать проверку подключения регулятора»)**
如果调压器不应答请求，Samovar 会关闭加热。
- 5 **用于计算功率的加热管电阻 («Сопротивление тэна для расчета мощности»)**
欧姆。使用带反馈的调压器时用于计算功率。
- 6 **电网电压 («Напряжение сети»)**
伏。与加热管电阻一起决定加热器的最大功率 (NBK 模式)。

- 7 使用水流量监控 («Использовать контроль потока воды»)**
考虑冷却水流量传感器的读数。
- 8 温度颜色 («Цвета для температур»)**
每个传感器在主页、“温度” («Температура») 选项卡和本地图表中的颜色。
- 9 继电器电平设置 («Настройки уровней для реле»)**
四个继电器各自的有效电平：Low Level Trigger 模块选 LOW，High Level Trigger 模块选 HIGH。
- 10 下载设置文件 («Скачать файл настроек»)**
把所有设置保存到电脑上的 samovarbackup.txt 文件中。
- 11 重置 WiFi 设置 («Сбросить настройки WiFi»)**
忘记 Wi-Fi 网络并重启。
- 12 上传设置文件 («Загрузить файл настроек»)**
把之前保存的文件中的设置填入表单。要应用这些设置，请点击“保存” («Сохранить»)。
- 13 重启 Samovar («Перезагрузить Самовар»)**
不断电重启。

Samovar 7.00: 选项卡中已没有“日志数据记录” («Запись данных в лог») 字段；日志以固定的 4 秒间隔写入。消息和日志经 samovar-tool.ru 服务器发送到 samovar-tool.ru 网站，见[云端日志和图表](#)。

Blynk 令牌 («Токен Blynk»)

用于 samovar-tool.ru 服务器的设备令牌。注册后由 Samovar 手机应用显示，见[手机应用和网站](#)。令牌也可以在 [Wi-Fi 连接](#) 门户中输入。过程日志和消息也通过这个令牌发送到云端。

摄像头视频 URL («URL видео камеры»)

如果你能通过 http 或 rtp stream 提供视频流的摄像头，可以填写摄像头的访问 URL。视频将在 Blynk 应用中可用（有一个限制：视频只能在局域网内观看）。

时区 («Часовой пояс»)

你所在的时区：相对格林尼治的时差，整数小时。

使用调压器连接检查 («Использовать проверку подключения регулятора»)

用于带反馈的调压器。Samovar 定期向调压器发送请求。如果连续几次请求都没有应答，Samovar 显示消息“与调压器无连接！” («Нет связи с регулятором!»)，然后停止过程，并显示消息“紧急关断！与加热调压器无连接。” («Аварийное отключение! Нет связи с регулятором нагрева.»)

用于计算功率的加热管电阻 («Сопротивление тэна для расчета мощности»)

加热管的电阻值。使用带反馈的调压器时用于计算功率。当前功率值显示在主页上。

电网电压 («Напряжение сети»)

供电电网电压。它与加热管电阻一起决定加热器的最大功率；用于 [NBK](#)。

使用水流量监控 («Использовать контроль потока воды»)

使用冷却水流量传感器。如果传感器已连接但未勾选此复选框，监控供水时不考虑它的读数。

温度颜色 («Цвета для температур»)

为各个温度传感器设置显示颜色。

为某个传感器设置颜色。该传感器的读数将以所选颜色显示在：

“主页” («Главная») 选项卡、“温度” («Температура») 选项卡和本地图表中；云端图表的颜色设置目前不会改变。

继电器电平设置 («Настройки уровней для реле»)

继电器模块大多是低电平触发型 (Low Level Trigger, LLT；默认启用)。这意味着要接通继电器，必须给继电器控制输入端 (IN1) 加逻辑“0”。给输入端 (IN1) 加逻辑“1”时，继电器断开。

选择某个继电器模块接通时的电平。对于普通的 Low Level Trigger 模块，选择 LOW。启动后 Samovar 先输出相反的断开电平；所选的有效电平只在发出接通继电器的命令时才施加。

按钮 («Кнопки»)

下载设置文件 («Скачать файл настроек»)

设置以文本文件 samovarbackup.txt 的形式保存到电脑上。文件可以重命名（例如改成模式名称）。

重置 WiFi 设置 («Сбросить настройки WiFi»)

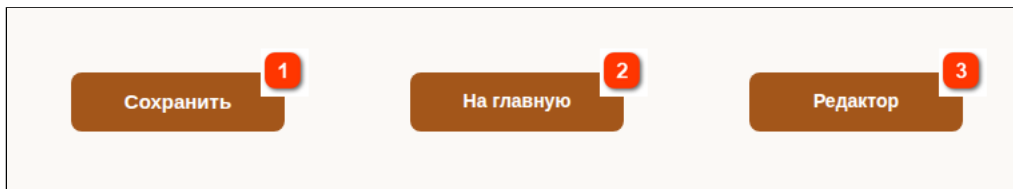
忘记 Samovar 当前连接的 Wi-Fi 网络。确认后 Samovar 重启，并开启用于 [Wi-Fi 连接](#) 的接入点。

上传设置文件 («Загрузить файл настроек»)

从之前保存的文件恢复设置。数值被填入本页的字段，然后出现提示“设置已载入表单，点击“保存”以应用。” («Настройки загружены в форму, нажмите «Сохранить», чтобы применить.»)。数值无效的字段会列在错误消息中。

重启 Samovar («Перезагрузить Самовар»)

在不切断电源的情况下重启 Samovar。



设置页面底部的按钮

1 保存 («Сохранить»)
保存页面所有选项卡的设置。

2 返回主页 («На главную»)
返回主页。

3 编辑器 («Редактор»)
打开 Samovar 文件编辑器。

保存 («Сохранить»)

保存本页的设置。保存后打开主页。如果某个字段的值超出允许范围，会出现消息“设置未保存。请检查字段：...” («Настройки не сохранены. Проверьте поля: ...»)。

返回主页 («На главную»)

返回主页。如果有未保存的更改，Samovar 会要求确认。

编辑器 («Редактор»)

切换到 Samovar 文件编辑器 (/edit 页面)。在这里可以删除旧日志 data_old.csv 或其他文件，并查看文件系统的内容。

保存设置

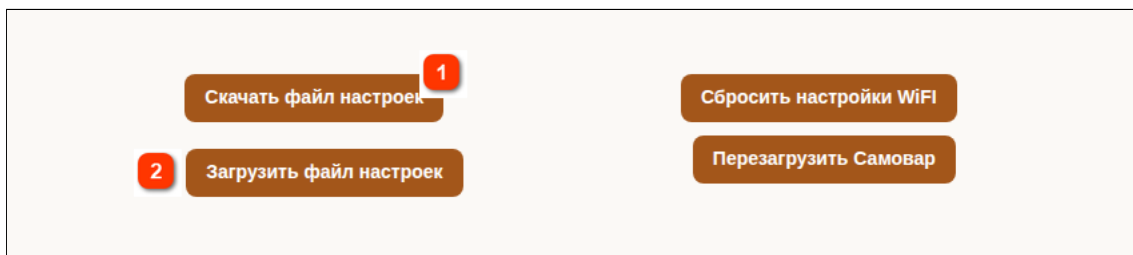
Samovar 7.00: “下载设置文件” («Скачать файл настроек») 和“上传设置文件” («Загрузить файл настроек») 按钮位于设置 («Настройки») → 其他 («Прочие») 。文件包含所有设置选项卡的全部字段，因此下表只是旧字段集的示例，不包括新参数。加载文件后，数值只是填入表单：要应用它们，请点击“保存” («Сохранить») 。

Samovar 可以把所有设置保存到文本文件中，之后再恢复。如果您使用几套温度传感器，这样就不必每次都输入它们的地址，很方便。点击“下载设置文件”后，浏览器会创建文件 `samovarbackup.txt`，并保存到浏览器的下载文件夹中。

保存的文件可以重新命名（按原料、模式等），需要时再加载。

要恢复设置，请点击“上传设置文件”并选择文件。Samovar 会检查它是否为 JSON 格式的设置文件，把数值填入表单字段，并显示“设置已载入表单，点击“保存”以应用” («Настройки загружены в форму, нажмите «Сохранить», чтобы применить») 消息。如果文件中有超出允许范围的值，消息会列出出错的字段；请在保存前改正。载入的值只有点击“保存”后才会生效。

这个文件中还有一些信息，可以单独复制粘贴到 Samovar 中，而无需加载之前的全部设置。例如 Blynk 令牌 或 摄像头 URL。



设置 → 其他：保存和加载设置文件 («Настройки → Прочие: сохранение и загрузка файла настроек»)

1 下载设置文件 («Скачать файл настроек») 创建包含设置表单全部字段的 `samovarbackup.txt` 文件，并保存到浏览器的下载文件夹中。

2 上传设置文件 («Загрузить файл настроек») 打开文件选择窗口。文件中的数值会填入表单；要应用它们，请点击“保存” («Сохранить») 。StepperStepMI 字段不会从文件加载。

警告。

在某些情况下，添加新传感器时，传感器可能会改变之前分配给它们的安装位置。这与传感器标识符以及 Samovar 排列传感器的逻辑有关。

`samovarbackup.txt` 文件 内容如下：

```
{ "mode": "0", "DistTemp": "98.000", "usepressure": true, "useautospeed": true, "autospeed": "2", "useautopowerdown": false, "UseBuzzer": true, "UseBBuzzer": true, "ChangeProgramBuzzer": false, "SteamAddr": "0", "DeltaSteamTemp": "-2.000", "SetSteamTemp": "0.300", "SteamDelay": "20", "PipeAddr": "2", "DeltaPipeTemp": "0.000", "SetPipeTemp": "2.000", "PipeDelay": "20", "WaterAddr": "3", "DeltaWaterTemp": "0.000", "SetWaterTemp": "0.000", "WaterDelay": "20", "TankAddr": "4", "DeltaTankTemp": "0.000", "SetTankTemp": "0.000", "TankDelay": "20", "ACPAddr": "1", "DeltaACPTemp": "0.000", "SetACPTemp": "0.000", "ACPDelay": "20", "StepperStepMI": "8954", "Kp": "150.000", "Ki": "1.400", "Kd": "1.400", "StbVoltage": "100.00", "blynkauth": "9u0XzN*****R1e7xHYp", "videourl": "", "TimeZone": "3", "CheckPower": false, "HeaterR": "35.000", "LogPeriod": "3", "SteamColor": "#2c03fc", "PipeColor": "#f90126", "WaterColor": "#08c7f7", "TankC
```

参数之间用逗号分隔，每个参数及其值都用引号括起来，并用冒号连接。

例如： "DistTemp": "98.000",

把保存的设置文件加载到 Samovar 时请注意：除 StepperStepMI 外，文件中的所有值都会替换当前值。如果在保存和加载之间更改过设置（例如校准了外部 I2C 泵，即 StepperStepMI2C 字段），加载后旧值会被填入表单。为避免这种情况，请从文件中删除这些参数及其值。

警告。

从 5.20 版开始，“取 1 ml 所需的电机步数” («Количество шагов двигателя необходимых для отбора 1 мл.») 值 (StepperStepMI) 包含在 `samovarbackup.txt` 文件中，但恢复设置时不会加载到 Samovar 中。恢复设置时已有的值不会被覆盖。可以手动恢复它。

下表列出参数名称及说明。数值作为示例取自上面的 `samovarbackup.txt` 文件。

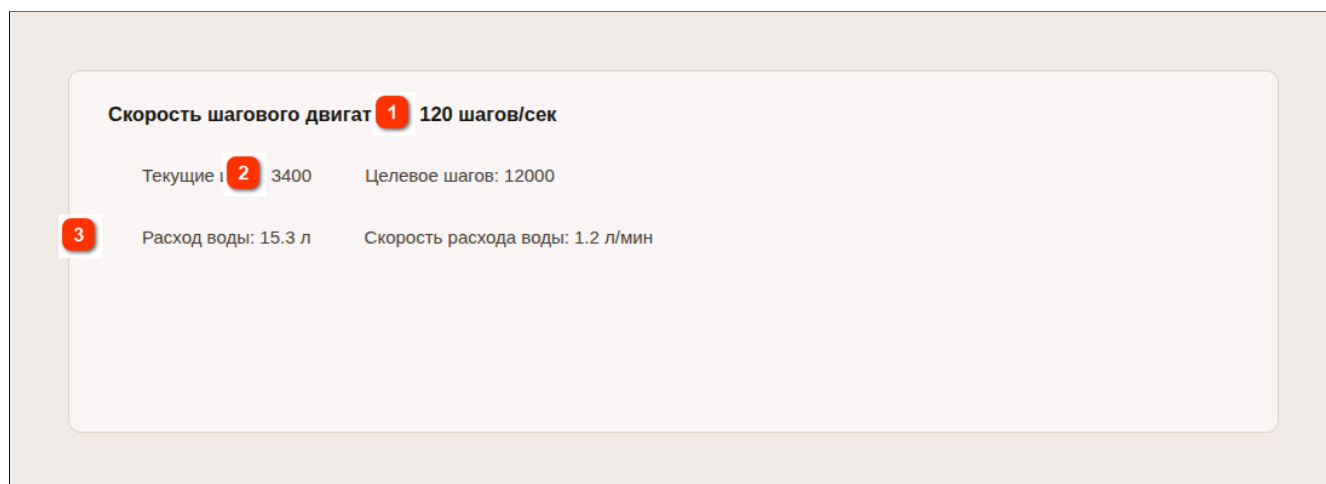
"mode"	0	运行模式
DistTemp	98.000	精馏和蒸馏进行到釜内温度 («Ректификация и дистилляция до температуры в кубе»)
usepressure	true	使用气压修正 («Использовать корректировку давления»)
useautospeed	true	使用水泵速度修正 («Использовать корректировку скорости насоса»)
autospeed	2	修正百分比 («Процент корректировки»)
useautopowerdown	false	使用供电功率修正 («Использовать корректировку подаваемой мощности»)
UseBuzzer	true	使用蜂鸣器提示 («Использовать зуммер для нотификации»)

UseBBuzzer	true	在浏览器中使用蜂鸣器提示 («Использовать зуммер для нотификации в браузере»)
ChangeProgramBuzzer	false	切换程序时蜂鸣器响 («Включать зуммер при смене программы»)
SteamAddr	0	分凝器前蒸汽传感器地址 («Адрес датчика пара перед дефлегматором»)
DeltaSteamTemp	-2.000	分凝器前蒸汽传感器修正 («Корректировка датчика пара перед дефлегматором»)
SetSteamTemp	0.300	分凝器前蒸汽传感器设定值 («Уставка датчика пара перед дефлегматором»)
SteamDelay	20	蒸汽温度超出设定值时的取酒延迟 (秒) («Задержка отбора при выходе Т пара за уставку (сек)»)
PipeAddr	2	塔节传感器地址 («Адрес датчика в царге»)
DeltaPipeTemp	0.000	传感器修正 («Корректировка датчика»)
SetPipeTemp	2.000	传感器设定值 («Уставка датчика»)
PipeDelay	20	温度超出设定值时的取酒延迟 (秒) («Задержка отбора при выходе Т за уставку (сек)»)
WaterAddr	3	水传感器地址 («Адрес датчика воды»)
DeltaWaterTemp	0.000	传感器修正 («Корректировка датчика»)
SetWaterTemp	0.000	传感器设定值 («Уставка датчика»)
WaterDelay	20	温度超出设定值时的取酒延迟 (秒) («Задержка отбора при выходе Т за уставку (сек)»)
TankAddr	4	蒸馏釜传感器地址 («Адрес датчика в кубе»)
DeltaTankTemp	0.000	传感器修正 («Корректировка датчика»)
SetTankTemp	0.000	传感器设定值 («Уставка датчика»)
TankDelay	20	温度超出设定值时的取酒延迟 (秒) («Задержка отбора при выходе Т за уставку (сек)»)
ACPAddr	1	TCA 传感器地址 («Адрес датчика TCA»)
DeltaACPTemp	0.000	传感器修正 («Корректировка датчика»)
SetACPTemp	0.000	传感器设定值 («Уставка датчика»)
ACPDelay	20	温度超出设定值时的取酒延迟 (秒) («Задержка отбора при выходе Т за уставку (сек)»)
StepperStepMl	8954	取 1 ml 所需的电机步数 («Кол-во шагов двигателя необходимых для отбора 1 мл.»)
Kp	150.000	加热 PID 调节器系数 («Коэффициент для PID регулятора нагрева»)
Ki	1.400	加热 PID 调节器系数 («Коэффициент для PID регулятора нагрева»)
Kd	1.400	加热 PID 调节器系数 («Коэффициент для PID регулятора нагрева»)
StbVoltage	100.00	保温模式下调压器电压 (V) («Напр. регулятора в режиме поддержания темп (V)»)
blynkauth	9u0Xz....7xHYp	Blynk 令牌 («Токен Blynk»)
videourl		摄像头 URL («URL видеокamеры»)
TimeZone	3	时区 («Временная зона»)
CheckPower	false	使用调压器连接检查 («Использовать проверку подключения регулятора»)
HeaterR	35.000	用于计算功率的加热管电阻 («Сопротивление тэна для расчета мощности»)
LogPeriod	3	数据写入日志的间隔 (秒) («Периодичность записи данных в лог (сек)»)
SteamColor	#2c03fc	蒸汽传感器温度颜色 («Цвет температуры датчика пара»)
PipeColor	#f90126	塔节传感器温度颜色 («Цвет температуры датчика царги»)
WaterColor	#08c7f7	水传感器温度颜色 («Цвет температуры датчика воды»)
TankColor	#338a05	蒸馏釜传感器温度颜色 («Цвет температуры датчика куба»)
ACPColor	#000000	TCA 传感器温度颜色 («Цвет температуры датчика TCA»)
rele1	1	继电器 1 电平 («Уровень реле 1»)
rele2	0	继电器 2 电平 («Уровень реле 2»)
rele3	0	继电器 3 电平 («Уровень реле 3»)
rele4	0	继电器 4 电平 («Уровень реле 4»)

注意。

如果没有已保存的配置，Samovar 7.00 会自行创建标准设置。固件源码 tools 文件夹中的 Samovar_default_settings.txt 文件可以像普通设置文件一样加载，但首次刷写后不需要这样做。

“附加” («Дополнительно») 选项卡



精馏模式中的“附加” («Дополнительно») 选项卡

- 1 步进电机速度**
执行当前程序行时每秒的步数。
- 2 步数**
当前步数是当前行中已经走过的步数；目标步数是该行结束前需要走的步数。
- 3 水流量**
流量传感器给出的总升数和流速（升/分）。仅在连接了传感器时才显示此区域。

Samovar 7.00: 带“设置” («Установить») 按钮的“速度，升/时” («Скорость, л/ч») 字段、“记录酒心取酒温度” («Зафиксировать T отбора тела») 按钮和系统参数行已移到“精馏” («Ректификация») 选项卡（主选项卡）。它们的说明保留在本页。

速度，升/时 («精馏»选项卡)

手动输入蠕动泵的取酒速度。它的优先级高于程序。前提是程序已经在运行，并且点击了“设置”按钮后才生效。按设定速度取酒，直到下一行开始。

步进电机速度

执行当前程序时步进电机每秒走多少步。

步数

当前步数：当前执行的程序中已经走过的步数。

目标步数：当前运行的程序结束前还剩多少步。

水流量

冷却液流量数据：总量（升）和流速（升/分）。仅在连接了流量传感器时显示。

注意。

请注意，流量传感器的读数误差很大，并且与流速密切相关。所以总的来说，它只能用来检查是否有水，以及非常粗略地估计流量。可以在 Samovar_ini.h 文件中的 `#define WF_CALIBRATION 98` 这一行进行校准。

系统参数 («精馏»选项卡)

显示在主页的“系统” («Система») 行中。已增加冷却泵状态信息。

Heap - 可用动态内存量，单位为字节

BME temp - 压力传感器芯片上的温度。实际上就是安装 ESP32 的机壳内的温度

RSSI - 接收到的 WiFi 信号功率水平指标。

空闲 («Свободно») - Samovar 文件系统上的可用空间，单位为字节。过程日志写在这里。

水泵 PWM («ШИМ насоса воды») - 冷却泵当前的 PWM 占空比（如果水泵由 Samovar 控制）。

注意。

RSSI 值以 dBm 为单位。网络正常工作可接受的最小值：RSSI = -85 dBm。重要！RSSI 数值越大，即负得越少，信号越强（例如 -48 大于 -78）。

记录酒心取酒温度 («精馏»选项卡)

设置新的蒸汽温度和塔节温度。

说明。

Samovar 在取酒头结束、“酒心” («Тело») 或“预液泛” («Предзахлеб») 程序开始的时刻记录蒸汽温度和塔节温度。此时通常会提高功率以改善分离效果，并取后酒头。功率提高后，温度可能超出[温度](#)选项卡上设置的设定值。这时 Samovar 会暂停取酒，等待温度回到记录值。点击该按钮后，当前的塔节温度和蒸汽温度成为主温度，取酒随之恢复。Samovar 会显示“新的酒心温度：蒸汽 = ...，塔节 = ...” («Новые Т тела: пар = ..., цагра = ...») 消息确认。

系统消息

运行过程中，Samovar 的警告显示在网页界面中。

在界面的任何页面上，消息都显示在右上角的面板中。每条消息都可以用叉号隐藏；“全部隐藏”（«Скрыть все»）按钮隐藏整个面板。隐藏紧急消息前，Samovar 会要求确认。页眉中的 ≡ 按钮打开消息历史；历史保存在浏览器中，“清除历史”（«Очистить историю»）按钮将其删除。

同样的消息也会发送到[手机应用和网站](#)（推送通知和“消息”选项卡）以及[云端](#)，可在日志的“消息”页面查看。

Samovar 7.00: 暂停或等待可能被记录为两条服务消息：开始和结束。在本地界面和网站的图表中，以及“控制”（«Управление»）页面上，这一对消息会被解析为原因、结果和持续时间。在普通消息列表中可能会看到服务标头 @P1。如果只收到其中一条，这就是没有持续时间的不完整事件。如果连接中断或 Samovar 重启，部分历史可能会缺失。

The screenshot displays the Samovar 7.00 web interface. At the top, there's a navigation bar with tabs: Ректификация, Программа, and Дополнительно. Below it, a status bar shows 'Самовар v. 7.00' and various system parameters like 'Расчёт', 'I2CStepper', 'Внешний насос', and 'График'. A green dot indicates '12:34:56 Работа 01:1 3'. A hamburger menu icon is on the right.

The main content area is titled 'Тело; Отбор тела' with a sub-header 'строка —'. Below this, a note states: 'Подробное объяснение недоступно: данные этапа отсутствуют или некорректны.'.

On the left, there's a 'ДЕЙСТВИЯ' (Actions) section with buttons: 'Выключить нагрев', 'Поставить на паузу', 'Далее', 'Сбросить процесс', 'График', and 'Настройки'.

Below the actions is a 'ДАТЧИКИ' (Sensors) section listing: Пар (78.2 °C), Царга (77.9 °C), Δ пар – царга (-0.3 °C), Куб (84.3 °C), Вода (22.4 °C), and ТСА (40.1 °C). There's also a 'ДЕТЕКТОР ПРИМЕСЕЙ' (Impurity Detector) section showing 'Состояние' as 'Стабильно, слежу' and 'Тренд' as '0.000 °C/мин'.

In the center is a schematic diagram of the distillation process with labels for 'ПАР 78.15°', 'ВОДА 22.4°', 'ЦАРГА 77.90°', 'КУБ 84.30°', and 'ТСА 40.1°'. It also shows 'Δ ПАР – ЦАРГА -0.25°' and 'ОТБОР 600.00 л/ч'.

On the right, there's a 'РЕГУЛЯТОР' (Regulator) section with 'Текущее напр.: 180.5 V' and 'Целевое напр.: 180.0 V'. It also shows 'Режим 1' and 'Мощность 1450 Вт'. Below this is a 'ДАВЛЕНИЕ' (Pressure) section with 'Атмосферное 752.3 мм рт.ст.' and 'В колонне: 1.2 мм рт.ст.'. At the bottom right is a 'СИСТЕМА' (System) section with parameters like 'Неар=152000', 'BME temp=24.5', 'RSSI = -58', and 'Свободно: 300000'.

Overlaid on the right side is a message notification panel. It contains three messages: '18:02:51 Переход к строке программы №2' (green), '18:02:51 Пауза по Т царги' (yellow), and '18:02:51 Сработал датчик захлёба!' (red). Each message has a close button (X). A 'Скрыть все' (Hide all) button is at the bottom right of the panel.

主页上的消息面板

1 消息 («Сообщения»)
消息列表，背景颜色取决于严重程度。右侧的叉号隐藏单条消息。

2 全部隐藏 («Скрыть все»)
隐藏面板。有新消息到达时面板会重新出现。

消息历史 («История сообщений»)

打开浏览器收到的所有消息列表，带有“清除历史”（«Очистить историю»）和“关闭”（«Заккрыть»）按钮。

消息

根据严重程度，消息的背景颜色不同：紧急、警告、通知。在手机应用中，紧急消息带有“报警！”前缀，警告带有“警告！”前缀。

有些警告不需要立即处理：

例如：“打开供水！”（«Откройте подачу воды!»）

当釜内温度达到草图中设置的值时，无论冷却水阀开着还是关着，都会以警告形式出现。

如果警告反复出现，请检查水阀是否真的打开、流量是否足够。如果用水泵保持温度，请检查水泵是否已连接。检查水流量传感器的接线。

另一些警告则相反，需要操作者立即干预。

例如：“水温达到临界！”（«Критическая температура воды!»）

这表示出口处的冷却液温度正在接近草图中设置的极限值（默认 70°，提前 5° 出现该消息）。如果操作者没有反应，Samovar 会降低加热功率；如果温度继续升到 75°，则关闭加热。

报警声音

紧急锁定加热和连接中断时，浏览器会发出报警声。如果浏览器阻止了播放，会出现提示“浏览器阻止了报警声音。点击页面以允许声音。”（«Браузер заблокировал звук тревоги. Нажмите на страницу, чтобы разрешить звук.»）

以下是 Samovar 的消息列表：

液泛传感器触发！（«Сработал датчик захлёба!»）

- 每当回流液位传感器触发且该事件被实际计时出现。Samovar 只计入持续 3 秒及以上的触发（WHLS_ALARM_TIME）。

从 XX.X 降低电压（功率）（«Напряжение (Мощность) снижаем с XX,X»）

- 液泛传感器触发后，从当前电压开始降低。

- 启用“[使用供电功率修正](#)”（«Использовать корректировку подаваемой мощности»）选项时，超出塔节或蒸汽设定值。

水温达到临界！（«Критическая температура воды!»）

- 水温距 [#define ALARM_WATER_TEMP 70](#) 这一行中的值已不足 5°。蜂鸣器响起，取酒暂停 30 秒。

水温达到临界！供水故障。从 XXX 降低电压（«Критическая температура воды! Ошибка подачи воды. Напряжение снижаем с XXX»）

- 如果对上一条消息没有反应，电压降低 5 伏，以防塔过热。在蒸馏和醪塔模式中，文字为“水温达到临界！从 XXX 降低……”（«Критическая температура воды! Понижаем ... с XXX»）。

紧急关闭！超过水（蒸汽、TCA）最高温度（«Аварийное отключение! Превышена максимальная температура Воды (Пара, TCA)»）

- 水温超过了 [#define MAX_WATER_TEMP 75](#) 这一行中的值，蒸汽为 MAX_STEAM_TEMP 98.8，TCA 为 MAX_ACP_TEMP 75。加热关闭，水阀关闭。

紧急关闭！供水中断。（«Аварийное отключение! Прекращена подача воды.»）

- 使用了流量传感器，但没有收到它的数据。

蒸馏器加热已开启（«Включен нагрев дистиллятора»）

- 蒸馏。发出了“开启加热”（«Включить нагрев»）命令。

您的设备不支持水泵控制（«Управление насосом не поддерживается вашим оборудованием»）

- 醪塔。固件中未启用冷却泵的 PWM 控制。

醪塔加热已开启（«Включен нагрев бражной колонны»）

- 醪塔。发出了“开启加热”命令。

醪塔运行结束（«Работа бражной колонны завершена»）

- 醪塔。达到设定的釜内最高温度时关闭加热。

- 发出了“关闭加热”（«Выключить нагрев»）命令。

糖化程序已完成（«Программа затираания завершена»）

- 啤酒模式。[啤酒酿造程序](#)的所有行均已执行。加热关闭。

冷却水阀已打开！（«Открыт клапан воды охлаждения!»）

- 啤酒模式。“冷却”（«Охлаждение»）行开始。

- 啤酒模式。“发酵”（«Брожение»）行。醪液温度超过釜内设定值，超出量达到超出设定值时。

冷却水阀已关闭！（«Закрыт клапан воды охлаждения!»）

- 啤酒模式。“冷却”行结束。

- 啤酒模式。“发酵”行。醪液温度回到设定范围内。

已达到投麦温度！（«Достигнута температура засыпи солода!»）

- 啤酒模式。已达到投麦温度，等待中。

已达到 XX° 温度休止。等待 N 分钟。（«Достигнута температурная пауза XX°. Ждем N минут.»）

- 啤酒模式。“暂停”（«Пауза»）行开始计时。

保温暂停：温度低于目标（«Пауза выдержки: температура ниже цели»）

- 啤酒模式。“暂停”行。温度比设定值低 3° 或更多，计时停止，加热继续工作。低得较少时没有消息，计时变慢。

保温中止：温度超出范围（«Выдержка приостановлена: температура вне полосы»）

- 奶酪制作模式。“保温”（«Выдержка»）阶段。温度低 3° 或更多，或高出容差；保温时间不计入。

保温继续（«Выдержка продолжена»）

- 啤酒和奶酪制作模式。温度已恢复，保温继续计时。

请投放啤酒花！（«Засыпьте хмель!»）

- 啤酒模式。“煮沸”（«Кипячение»）程序行结束前 30 秒。

转到程序第.....行（«Переход к строке программы №»）

- 转到下一个程序行。在啤酒模式中还会加上“加热到投麦温度：XX°”（«Нагрев до температуры засыпи солода: XX°»）。

釜内已无酒精（«В кубе не осталось спирта»）

- 蒸馏、醪塔。条件：釜内温度高于 90 度且加热开启。在设置中“蒸馏结束监测时间”（«Время для контроля завершения процесса дистилляции»）字段设定的时间内（默认 60 分钟，0 表示不检查），釜内温度变化不超过 0.1 度。

蒸馏已完成。总时间：N 分钟。（«Дистилляция завершена. Общее время: N мин.»）

- 达到设定的[釜内最高温度](#)时关闭加热。

- 加热被关闭。

- 釜内已无酒精。

按塔节（或蒸汽）温度暂停（«Пауза по Т царги (или пара)»）

- 塔节（或蒸汽）传感器温度超出了[设定值](#)。

程序执行完毕。（«Выполнение программы завершено.»）

- Samovar 内存中的整个程序已执行完。

- 加热被关闭。

达到釜内最高温度限值。程序结束。（«Лимит максимальной температуры куба. Программа завершена.»）

- 精馏。达到设定的[釜内最高温度](#)时关闭加热。

加速加热结束。稳定/全回流运行。（«Разгон завершён. Стабилизация/работа на себя.»）

- 蒸汽传感器温度达到 #define CHANGE_POWER_MODE_STEAM_TEMP 39 这一行中的值。

稳定完成，塔运行稳定。（«Стабилизация завершена, колонна работает стабильно.»）

- 条件：加热开启。蒸汽传感器已越过加速加热关闭点。六分钟内蒸汽温度变化不超过 0.1 度。

打开供水！（«Откройте подачу воды!»）

- 精馏。釜内温度达到 #define OPEN_VALVE_TANK_TEMP 77 这一行给出的值。

- 啤酒。“冷却”行开始。

- 啤酒。“发酵”行。醪液温度超过釜内设定值，超出量达到超出设定值时。

关闭供水！（«Закройте подачу воды!»）

- 精馏、蒸馏、醪塔。加热已关闭，水温比[水温设定值](#)低了草图中设定的量。

釜内开始沸腾！（«Началось кипение в кубе!»）

- 条件：加热开启；蒸汽温度升到 39° 以上，或者釜内温度停止上升而冷却水已变热。在蒸馏模式中，消息里还会加上计算出的釜内酒精度。

与调压器无连接！（«Нет связи с регулятором!»）

- 已启用[调压器连接检查](#)，而调压器连续几次没有应答。如果连接没有恢复，则显示“紧急关闭！与加热调压器无连接。”（«Аварийное отключение! Нет связи с регулятором нагрева.»）

紧急关闭！加热器控制错误。（«Аварийное отключение! Ошибка управления нагревателем.»）

- 调压器报告的实际功率连续几次与设定功率相差 20% 以上。

蒸汽（塔节、水、釜内、TCA）温度传感器错误！（«Ошибка датчика температуры пара (царги, воды, куба, в TCA)!»）

- 没有温度传感器的数据。

存储空间即将用完！总计：xxxxx，已用：xxxxxx（«Заканчивается память! Всего: xxxxx, использовано: xxxxxx»）

- 文件系统中已没有存放日志的空间。可用空间少于 16 KB 时，Samovar 会自行删除旧日志 data_old.csv。请通过[编辑器](#)删除不需要的文件。

紧急关闭：紧急按钮被按下（«Аварийное отключение: нажата аварийная кнопка»）

- [紧急按钮被按下或紧急传感器触发](#)。

加热被紧急保护锁定，需要重启（«Нагрев заблокирован аварийной защитой, требуется перезагрузка»）

- 紧急关闭后，在 Samovar 重启之前加热不会开启。锁定期间浏览器会发出报警声。

发现未完成的会话，模式 N，程序 M 号；加热未自动恢复。（«Обнаружена незавершённая сессия, режим N, программа №M; нагрев НЕ возобновлён автоматически.»）

- Samovar 在过程中重启了。必须手动开启加热；如果距重启不到 30 分钟，云端日志会接续上一次会话。

连接中断！（«Обрыв связи!»）

- 浏览器消息：Samovar 连续三次轮询都没有应答。页面变暗并被锁定，直到连接恢复；时钟行显示最后一次收到数据的时间。

配置加载出错：.....（«Конфигурация загружена с ошибкой: ...»）

- 启动时读取设置出错，Samovar 使用默认值工作。原因显示在主页读数下方的横条中。

云端日志和图表

在云端查看信息。

要让 Samovar 把日志发送到云端，请获取设备令牌，并在[设置 → 其他 \(«Настройки → Прочие»\)](#) 选项卡的“Blynk 令牌” («Токен Blynk») 字段中输入。要在您的个人资料中查看该日志，请在网站 www.samovar-tool.ru 的“用户 → 设备”菜单中添加设备。只要设定了令牌并且与服务器保持连接，Samovar 在写入本地日志时就会把数据和消息发送到云端，可以在网站 www.samovar-tool.ru 的“日志”页面 (samovar-tool.ru/logi/) 上查看。

如果网站管理员提高了设备数量上限（默认为一台，见[移动应用和网站](#)），一个网站用户可以拥有多台 Samovar。“日志”页面显示在“用户 → 设备”菜单中所选设备的过程。要打开另一台 Samovar 的存档，请先在那里选中它。

Samovar 7.00: 发送云端日志不再有单独的设置。只要设定了令牌并且有互联网连接，日志行就会每 4 秒自动发送一次。如果 Samovar 在过程中重启，并在 30 分钟内再次开启加热，记录会在同一会话中继续。

在图表上，新的暂停显示为带有原因、结果和持续时间的区间。如果部分数据未收到或设备时间被更改，可能只剩一个单独的标记而不是区间。网站更新后，旧日志中不会出现这类区间。

云端日志列表如下所示：

Список логов					
Дата логa	Детализация	График	Скачать лог	Сообщения	Редактировать
16.09.2022 10:20:59	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог
22.09.2022 09:09:22	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог
23.09.2022 08:04:33	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог
23.09.2022 12:55:09	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог
24.09.2022 09:11:44	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог
26.09.2022 09:40:13	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог
05.10.2022 07:50:03	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог
05.10.2022 12:39:39	Детализация	График	Сохранить лог	Сообщения	Удалить Объединить лог Расшарить лог

[详情](#)——取酒程序和附加信息（“过程备忘”）
图表——生成过程图表。

保存日志——日志文件将下载到您的电脑。文件名为 data.csv

消息——该过程中出现的所有消息。

删除——删除该日志。不可撤销。

合并日志——将该日志与上一个日志合并。不可撤销。

共享日志——允许所有拥有该日志链接的人查看。可以撤销。

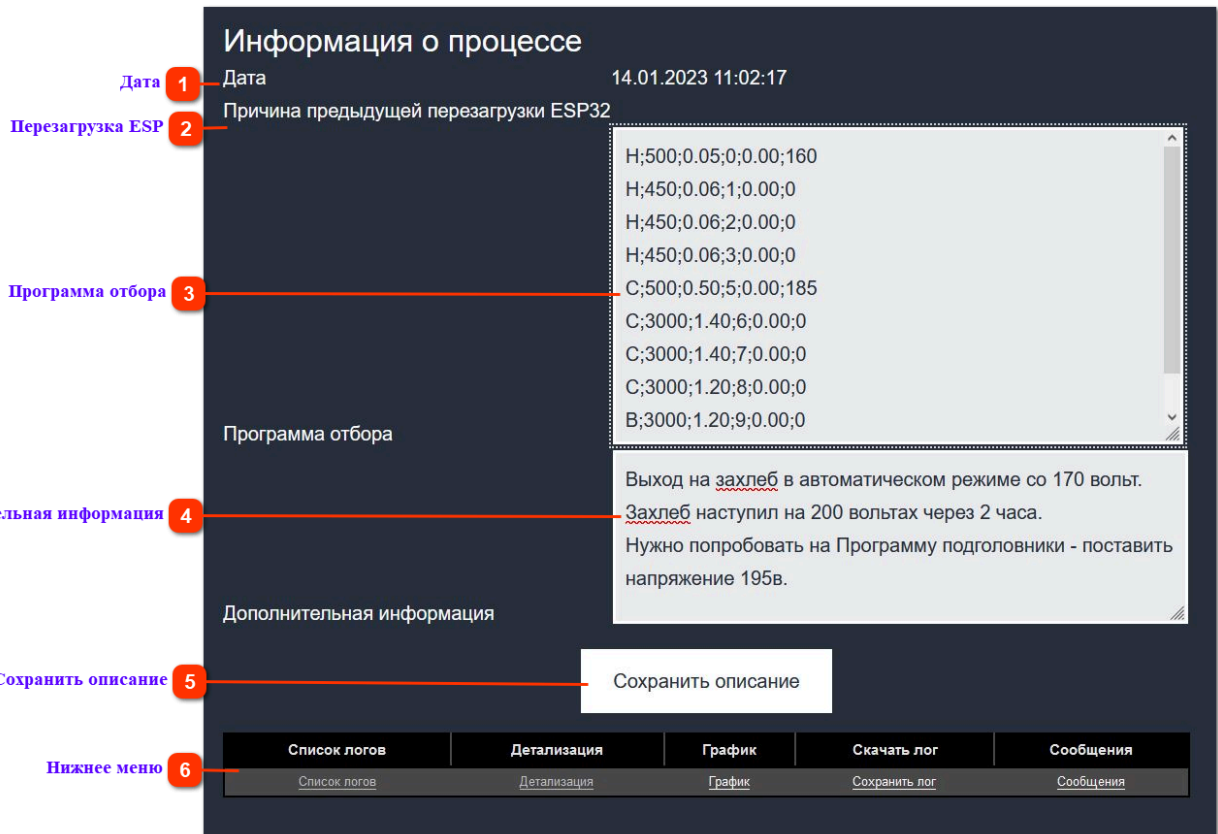
日志和图表只有您自己可以访问，但您可以与其他参与者共享。

共享日志的方法：
点击“共享日志”。对“将日志设为所有人可见？”的提问回答“是”。
“共享日志”标签将变为“将日志设为私有”。
打开日志（例如在“详情”页），从浏览器地址栏复制链接，并以任意方式发送给您想共享的人。
所有拥有该链接的人都能查看图表并下载日志文件 data.csv

禁止查看：点击“将日志设为私有”，标签将变为“共享日志”。其他人将无法再访问该日志。

详情

点击“详情”会打开一个新窗口。



1. 日期

过程开始的日期和时间。开始时间按开始记录本地日志的[条件](#)记录。

2. ESP32 上次重启的原因

这里可能显示 Samovar 停止工作的原因。

3. 取酒程序

文本格式的取酒程序。可以复制，并粘贴到“程序”（«Программа»）选项卡的文本框中用于其他过程。不可编辑。

4. 附加信息

文本格式的取酒程序说明。自由文本。可以编辑。

5. 保存说明

编辑取酒程序说明后点击此按钮保存修改。

6. 底部菜单

日志列表——返回上一个窗口，即所有日志的列表

详情——就是当前这个窗口

图表——查看该过程的图表

消息——该过程中出现的所有消息。

图表

与本地图表相比，云端图表增加了以下数据：

- 取酒速度
- 水流量
- 电压/功率。

- 蒸馏釜中的酒精度
- 取出酒液的酒精度（该数据只对“蒸馏”（«Дистилляция»）模式有意义）

开始绘图的所有条件和视图控制方式都与[本地图表](#)相同。数据存储在云端，图表图像由您电脑上的浏览器生成。数据越少、网速越快、电脑性能越强，图表生成得越快。

图表示例：



借助图表上的附加数据，该示例展示了调压器在预液泛模式下的工作情况。

消息

如果错过了某条消息，可以在这里查看历史记录。例如：加热为什么“突然”关闭了。

示例：
消息窗口片段：取酒头，以及塔进入预液泛模式。

Список сообщений		
Дата сообщения	Сообщение	Критичность сообщения
14.01.2023 11:02:18	Программа: старт строки №1; отбор в ёмкость 0	2
14.01.2023 21:02:32	Программа: старт строки №2; отбор в ёмкость 1	2
15.01.2023 03:28:12	Программа: старт строки №3; отбор в ёмкость 2	2
15.01.2023 07:34:41	Программа: старт строки №4; отбор в ёмкость 3	2
15.01.2023 07:34:54	Новые Т тела: пар = 75.00; чарга = 77.06	1
15.01.2023 07:34:54	Программа: старт строки №5; отбор в ёмкость 5	2
15.01.2023 08:35:14	Программа: старт строки №6; отбор в ёмкость 6	2
15.01.2023 08:41:50	Напряжение снижаем с 200.00	0
15.01.2023 08:42:35	Напряжение снижаем с 198.00	0
15.01.2023 08:43:21	Напряжение снижаем с 196.00	0
15.01.2023 08:44:06	Напряжение снижаем с 194.00	0
15.01.2023 08:44:51	Напряжение снижаем с 192.00	0
15.01.2023 08:45:37	Напряжение снижаем с 190.00	0
15.01.2023 08:46:00	Напряжение снижаем с 188.00	0

问题与解答。FAQ

日志在云端保存多久？
时长不足 30 分钟的日志在创建一天后删除。其余日志无限期保存。

“啤酒”（«Пиво»）模式下的酿造会自动出现在网站个人中心中的酿造记录中吗？
会。在“啤酒”模式下开始的会话会自动在网站“酿酒师工作台”的酿造记录中创建一条记录，所用数据与本页面收到的数据相同（糖化步骤、带有“投放啤酒花！”（«Засыпьте хмель!»）消息的煮沸、冷却）；设备不会额外发送任何内容。详情见页面[网站和应用中的啤酒酿造记录](#)。

本地日志和图表

本地查看

Samovar 记录温度日志并保存在自身内存中。每种模式下，日志在满足特定条件时开始记录：

精馏——执行“开始取酒”（«Начать отбор»）命令时。
蒸馏——执行“开启加热”（«Включить нагрев»）命令时。

啤酒——酿造程序第一行开始时。

醪塔（BK）——执行“开启加热”命令时。

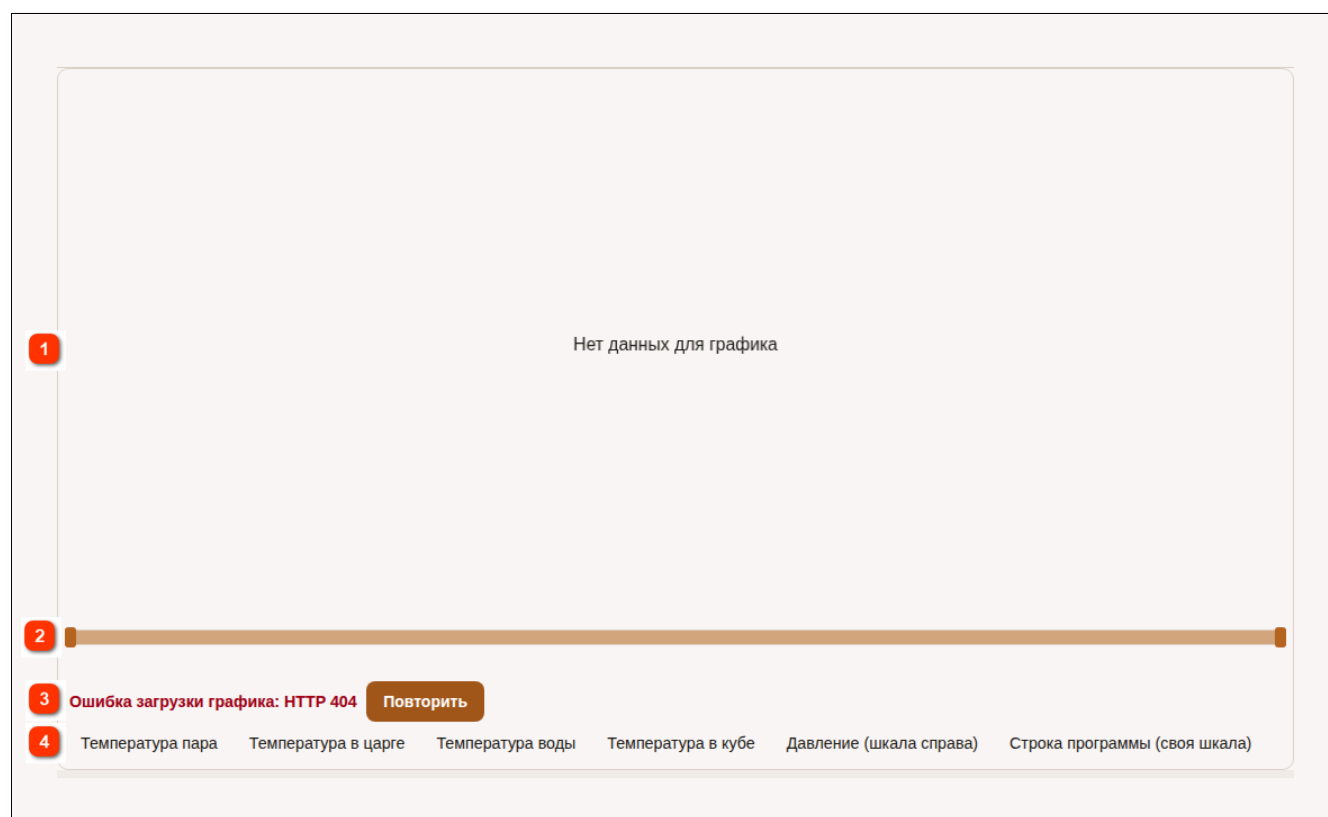
NBK 和奶酪制作——程序启动时。

所有模式下，日志在执行“关闭加热”（«Отключить нагрев»）或“重置过程”（«Сбросить процесс»）命令时结束

如果在一次运行中关闭了加热，之后又恢复了过程，则每次都会按上述条件重新开始记录日志。

Samovar 7.00: 日志间隔不可配置——每 4 秒写入一行。“将数据写入日志”（«Запись данных в лог»）字段已从[设置 - 其他](#)选项卡中删除。如果剩余可用内存少于 16 KB，Samovar 会自动删除旧日志 data_old.csv。

本地图表查看窗口如下所示：



本地图表

- 1 图表**
左侧是温度刻度，右侧是压力刻度。温度刻度的下限取所有传感器中的最小值。
- 2 区段滑块**
左右滑块设定显示的区段。双击滑条显示整个图表。
- 3 状态行**
已加载的点数和操作提示。
- 4 图例**
点击曲线可隐藏或显示它。

图表由固件绘制，不使用外部库。左侧是温度刻度，右侧是压力刻度；程序行号作为单独的曲线显示。图表下方是状态行：“已加载点数：N。图表下方的滑块用于选择区段，双击滑条显示整个图表，点击图例可开关曲线，悬停显示数值。”（«Загружено точек: N. Ползунки под графиком задают участок, двойной щелчок по полосе - весь график, легенда - вкл/выкл, наведение - значения.»）

Samovar 7.00: 图表打开期间，数值曲线旁可能出现暂停和等待的标记。这些是页面在内存中保存的当前运行的单独消息，不是 data.csv 或 data_old.csv 中的行。如果同一原因的开始和结束都已收到，则显示时间区间、持续时间和结果；如果只收到其中一部分，则是不完整的事件，不会凭空推算持续时间。漏收消息或 Samovar 重启可能导致历史记录不完整或被清空。

只有当时间可靠时，标记才会放到时间轴上。如果没有时间或 Samovar 的时钟出现明显跳变，原因和持续时间的信息保留在状态行中，但图表上不绘制区间。较早的程序行标记和暂停事件仍作为单独的观测点，而不是区间。

图例

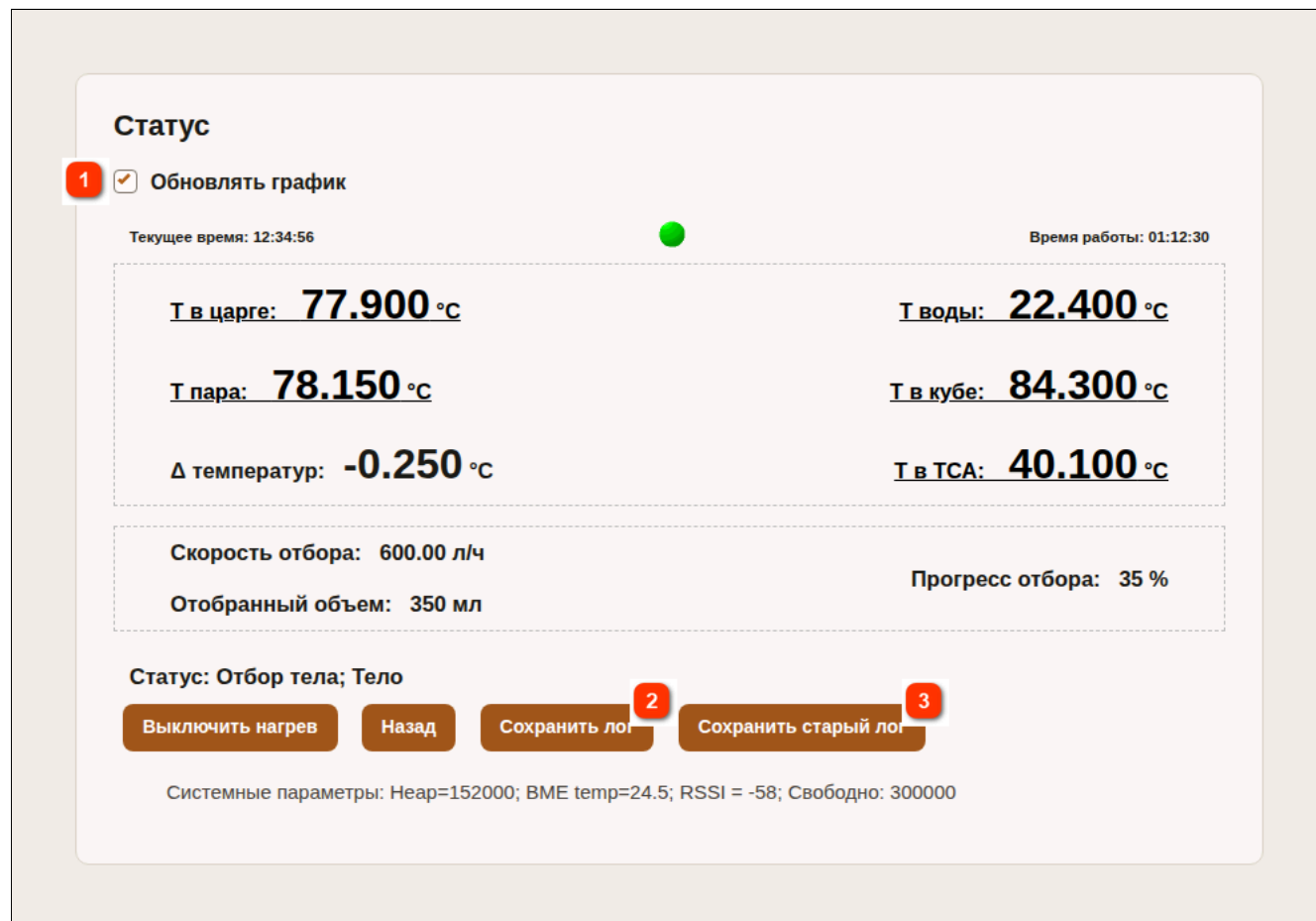
点击图例中的条目即可关闭任意曲线。例如，点击“压力”（«Давление»），压力曲线就不再显示。再次点击则重新显示。在主页面上隐藏的传感器在这里同样隐藏。

图表区段

图表下方有一个带两个滑块的滑条。拖动左右滑块，可以选择分析所需的任意细节程度。单击滑条会平移所选区段，双击则使图表恢复原始视图。滑块也可以用键盘方向键移动。

将光标悬停在图表上时，会出现一条竖线，并显示光标位置处所有曲线的精确数值。

查看期间，图表会自动补充新的数据点，频率不超过每 15 秒一次；也可以在下方窗口中取消勾选“更新图表”（«Обновлять график»）复选框，关闭自动更新。



图表下方的“状态”（«Статус»）窗口

- 1 **更新图表（«Обновлять график»）**
取消勾选后，图表不再补充新的数据点。
- 2 **保存日志（«Сохранить лог»）**
下载当前日志 data.csv。
- 3 **保存旧日志（«Сохранить старый лог»）**
下载上一个日志 data_old.csv。

关闭图表更新

关闭更新后，查看和分析图表更方便。日志仍会继续写入文件。重新勾选复选框后，期间发生的所有事件都会显示在图表上。

注意！

图表在您的电脑上生成；如果日志很大，可能需要相当长的时间。如果 Samovar 正忙，会自动重试下载日志。

同一页面上（第二个窗口中）有一个“保存日志”（«Сохранить лог»）按钮。点击后，过程结束时当前日志会保存到您电脑上浏览器的“下载”文件夹中。文件名为 data.csv。

如果新日志已经开始记录，仍可点击“保存旧日志”（«Сохранить старый лог»）按钮下载上一个日志。文件名为 data_old.csv

即使只是打开了 Samovar 而没有开始加热或取酒，也有两个日志可以保存：最新的（data.csv）和上一个（data_old.csv）。

当过程按条件启动时，新日志会取代旧日志。

示例。

如果在一次运行中加热被关闭了五次，过程又被恢复，并且每次都满足了日志开始条件，那么在本地只能查看根据最后一个日志数据生成的图表。上一个日志只能作为数据文件 data_old.csv 下载。前三个日志在本地已无法获取。

但在[云端](#)，五个日志都会保存，可以查看五张图表。此外，在云端还可以把它们合并为一个日志，并生成一张图表。单独的或合并后的日志都可以下载到电脑上。

之后可以在电脑上用 chartlocal.htm 页面查看图表，该页面位于 Samovar 文件夹的 tools 中

Samovar 工作模式

1

Режим работы

Ректификация

Ректификация, дистилляция и БК до температуры в кубе:

0

°C

Время для контроля завершения процесса дистилляции:

0

мин

☐ Использовать корректировку давления

☐ Использовать датчик уровня флегмы при ректификации

☐ Не показывать схему процесса

Параметры оборудования (колонны)

Диаметр:

1.5

дюймов

Высота насадки:

0.50

метров

Плотность насадки:

80 % (0-100)

Детектор проскока примесей (ректификация)

☐ Использовать детектор

Давление, при котором включится авария:

0

мм. Рт. ст.

Нулевая точка МРХ:

0

Отсчётов МРХ на 1 мм рт. ст.:

0

Настройки зуммера

☐ Использовать зуммер для нотификации

☐ Использовать зуммер для нотификации в браузере

☐ Включать зуммер при смене программы

Самотестирование

Внимание: включится подача воды (насос или клапан), запустится перистальтический насос отбора, сервопривод пройдёт 5 положений ёмкостей.

在“设置 → 基本” («Настройки → Основные») 选项卡上选择工作模式

1

工作模式 («Режим работы»)

模式列表。低温慢煮和 Lua 模式在列表中是隐藏的。

在网页界面中可以选择以下模式之一：

[精馏](#) («Ректификация»)

[蒸馏](#) («Дистилляция»)

[啤酒](#) («Пиво»)

[醪塔 \(БК\)](#) («Бражная колонна (БК)»)

[连续醪塔 \(NBK\)](#) («Непрерывная бражная колонна (NBK)»)

[奶酪制作](#) («Сыроварение»)

列表中的 [低温慢煮](#) («Су-вид») 和 [Lua 模式](#) («Lua-режим») 两项是隐藏的，不能通过普通的模式切换开启。NBK 模式和 Lua 模式并非每个固件版本都有：

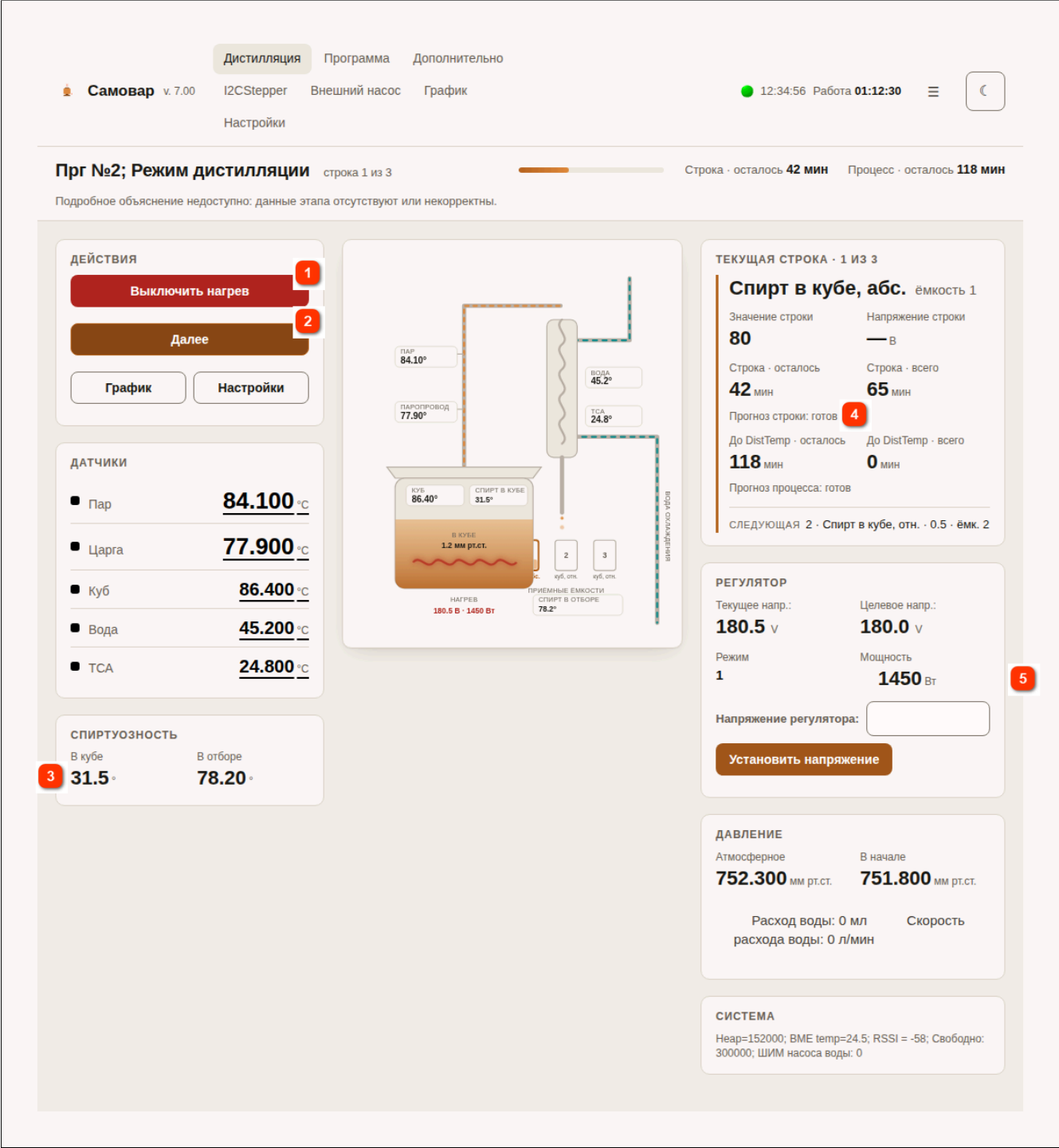
如果设置中保存的模式在当前版本中不可用，它会显示在列表中，但无法再次选择。

Samovar 7.00: 新增了奶酪制作模式。切换模式不再需要重启 ESP32。

如何切换模式

- . 如果有流程正在运行，请正常停止，并等待加热和水泵关闭。加热开启时，新模式的设置不会被保存。
 - . 打开 **设置** → **基本** («Настройки → Основные»)。在 **工作模式** («Режим работы») 字段中选择所需的模式。
 - . 保存之前，连接新模式所需的传感器和设备。
 - . 点击 **保存** («Сохранить»)。Samovar 会关闭日志，取消待执行的命令，无需重启即可切换模式，浏览器会转到所选模式的主页面。
 - . 打开 **设置** → **温度** («Настройки → Температура») ，点击刷新传感器列表，为每个传感器指定位置。再次点击“保存”。
 - . 检查所选模式的程序，并在开启加热前运行 [自检](#)。
- 已保存的程序和设置按模式分开存放。如果使用不同的传感器组合，请在 [其他](#) («Прочие») 选项卡上为每种配置保存一份设置备份文件。

初次蒸馏模式



“蒸馏”（«Дистилляция»）模式主窗口

- 1 打开加热（«Включить нагрев»）
启动和停止蒸馏。加热打开时，按钮名称变为“关闭加热”（«Выключить нагрев»）。
- 2 下一步（«Далее»）
强制转到蒸馏程序的下一行。只在加热打开时可用。
- 3 酒精度（«Спиртуозность»）
计算得出的蒸馏釜内和取酒的酒精度。蒸馏釜开始沸腾前显示 100%。
- 4 时间预测（«Прогноз времени»）
距当前行条件满足和距过程结束还剩多少分钟。每 30 秒重新计算一次。
- 5 调压器（«Регулятор»）
调压器的当前电压（功率）和目标电压（功率），以及手动设置字段。见[手动设置调压器电压](#)。

警告！

只监控三个温度传感器：**水、蒸馏釜和 TCA**

如果在“设置”（«Настройки»）→“温度”（«Температура»）选项卡上连接并分配了温度传感器，界面会显示所有温度传感器的读数。

这样可以把多余的传感器作为信息传感器，按自己的需要使用。例如：观察进水温度等。

警告。

沸腾开始前酒精度显示为 100% - 这是为了之后能确定正确的实际酒精度。

收到“打开加热”（«Включить нагрев»）命令后，Samovar 让调压器切换到“升温”（«Разгон»）模式，并显示消息“蒸馏器加热已打开”（«Включен нагрев дистиллятора»）。可以在“调压器电压”（«Напряжение регулятора»）字段中输入数值来改变功率。设定的功率一直保持到转入下一条取酒程序行。如果连接了**辅助加热管**，它与加热一起打开，并在第一次转入下一程序行时关闭。如果在“设置”（«Настройки»）→“主要”（«Основные»）选项卡上取消勾选“沸腾时使用辅助加热管”（«Использовать разгонный тэн при кипении»），辅助加热管会更早关闭 - 一旦 Samovar 检测到蒸馏釜开始沸腾就关闭。“下一步”（«Далее»）按钮让 Samovar 强制转到**蒸馏程序**的下一行，不等当前行的条件满足。只在加热打开时有效。

在初次蒸馏模式下，Samovar 显示并监控三个传感器的温度：

水。可以通过修改“设置”（«Настройки»）-**温度**（«Температура»）选项卡上水温传感器的设定值来调节水温（前提是使用了供水泵）。如果 TCA 中的传感器温度超过其设定值并且高于水温，水泵会暂时加大水流量。

蒸馏釜。当蒸馏釜温度达到以下设定值时：

1. 在文件 [Samovar.ini.h](#) 中的这一行：

```
#define OPEN_VALVE_TANK_TEMP 77
```

将发出打开水阀并启动冷却泵的命令。

蒸馏釜开始沸腾时，Samovar 确定酒精度并显示消息“蒸馏釜已开始沸腾！酒精度 XX.X”（«Началось кипение в кубе! Спиртуозность XX.X»）。

出现以下任一情况时，蒸馏结束（加热随之关闭）：

- 蒸馏釜温度超过设置中设定的**蒸馏结束温度**。
 - 蒸馏釜中已没有酒精（判断依据：在**设置**（«Настройки»）选项卡上“蒸馏结束监控时间”（«Время для контроля завершения процесса дистилляции»）字段设定的时间内，蒸馏釜温度变化不到 0.1°，且蒸馏釜温度高于 90°）。Samovar 报告“蒸馏釜中已没有酒精”（«В кубе не осталось спирта»）。设为 0 则关闭此项检查。
- 结束时显示消息“蒸馏已完成。总时间：N min。”（«Дистилляция завершена. Общее время: N мин.»）。

加热关闭后，出现以下任一情况时，供水阀门关闭，水泵停止：

- 水温已降到比水温传感器设定值低 20°，但不早于加热关闭后 3 分钟；
- 在此过程中曾经很热的蒸馏釜已冷却到 70° 以下。

Samovar 7.00: 以前只按第一个条件关水。如果水温设定值低于进水温度加 20°，冷却泵会一直运行，直到操作者手动关闭。现在蒸馏釜冷却后也会关水。

负责温差的那一行位于文件 [Samovar.ini.h](#)

```
#define DELTA_T_CLOSE_VALVE 20
```

// 设定的冷却水温度 T 与水温 T 之差：在蒸馏和精馏模式下，蒸馏过程结束后水泵一直运行到水温降至该值为止

TCA。如果蒸汽冲出，Samovar 会关闭加热。

建议制作一个简单的监控单元，在蒸馏时与**附加保护**一起使用。这时 TCA 温度传感器显示从取酒口流出的产品温度，从而可以把冷却水流量调到最佳。

水温接近临界值时，Samovar 发出声音信号，显示消息“水温达到临界！”（«Критическая температура воды!»），并将调压器电压降低 5 个单位；如果水温或 TCA 温度继续上升，则紧急关闭加热。

如果在“设置”（«Настройки»）选项卡上勾选了**使用气压修正**（«Использовать корректировку давления»），Samovar 会根据酒精-水混合物沸点与气压的关系修正蒸馏釜温度。

时间预测。在“当前行”（«Текущая строка»）区块中，Samovar 显示距当前程序行条件满足还剩多少分钟，以及距过程结束（达到设定的蒸馏结束温度）还剩多少分钟。预测每 30 秒重新计算一次；蒸馏釜开始沸腾前显示“等待实际沸腾”（«ожидание фактического кипения»），沸腾后的最初几分钟显示“正在积累数据”（«накопление данных»）。如果所有程序行都已执行完，则不再预测行时间。

记录日志，将信息发送到 Blynk 和云端，并绘制图表。安全组处于工作状态。

问题与解答。FAQ

从 6.0 版起，蒸馏模式增加了取酒程序，这是否意味着取酒必须使用伺服驱动器？

如果不需要把产品分装到不同的容器中，就不必连接伺服驱动器。这时就是普通蒸馏，与蒸馏程序中写了什么无关。

蒸馏 程序

蒸馏模式的程序。它的作用是根据温度或酒精含量更换取酒容器和/或改变电压。仅此而已。

注意！

请注意，更换容器的逻辑与精馏模式不同。精馏模式下，转到指定了该容器的程序行时才切换到该容器；蒸馏模式下，当前程序行所监控的事件发生时，就切换到另一个容器。

与精馏模式还有一个区别：精馏模式下程序行执行完毕就意味着精馏结束，而蒸馏模式下过程会一直继续，直到操作者停止它，或者触发边界条件停止：蒸馏釜温度超过设置中设定的蒸馏**结束温度**，或者蒸馏釜中**已没有酒精**。

蒸馏的启动和结束算法没有改变，所以如果您不打算按容器分配，有没有程序都无所谓。

酒精含量既在蒸馏釜中测定，也在蒸汽中测定（对蒸馏来说，蒸汽中的含量等同于取酒中的含量）。计算纯粹是数学上的，在实际条件下可能与酒精计的读数不同。两个值都显示在界面中。

如果不需要按容器分配，可以不接舵机。这时无论蒸馏程序里写了什么，都是普通的蒸馏。也可以手动更换容器。

№	Тип программы	Значение	№ емкости	Напряжение
01	Спирт в кубе абс	80	3	0
02	Спирт в кубе отн.	0.50	2	0
03	Спирт в кубе отн.	0.30	3	0

Описание программы дистилляции в текстовом формате для сохранения в облаке:

Программа дистилляции:

Установить программу

Очистить программу

蒸馏模式的“程序”选项卡（«Вкладка «Программа» режима Дистилляция»）

1

编号（«Номер»）

程序行编号。

2

程序类型（«Тип программы»）

T（温度，«Температура»）——按蒸馏釜温度蒸馏。

程序行示例：

T;85;1;150 - 蒸馏釜温度达到 85° 时，舵机转到 1 号位置，调压器把电压设为 150 伏，并转到下一个取酒程序行。

S（蒸馏釜酒精，相对值，«Спирт в кубе отн.»）——按蒸馏釜中酒精含量的相对变化蒸馏。即当前酒精含量与该程序行开始时酒精含量之比。例如，需要在蒸馏釜酒精含量比开始取酒时下降 50% 时切断取酒。该值是大于 0 且小于 1 的数。

程序行示例：

S;0.5;1;0 - 蒸馏釜酒精含量下降 50% 时，舵机转到 1 号位置，调压器保持上一行指定的电压，并转到下一个取酒程序行。

Samovar 记住转换时刻的蒸馏釜酒度。

注意：

要在达到蒸馏釜酒精含量的 1/4 时切换到下一个容器，并且比如改变加热功率，请在程序中再加一行：

S;0.5;2;200 - 这样，当蒸馏釜内容物剩余部分中的酒精含量下降 50% 时，舵机转到 2 号位置，调压器把电压设为 200 伏，并转到下一个取酒程序行。Samovar 再次记住转换时刻的蒸馏釜酒度。

A（蒸馏釜酒精，绝对值，«Спирт в кубе абс»）——按蒸馏釜中酒精含量的绝对变化蒸馏。计算基于蒸馏釜中的绝对酒精含量。

程序行示例：

A;15;1;0 - 蒸馏釜酒精含量降到 15% 时，转到下一个程序行。

P（取酒酒精，绝对值，«Спирт в отборе абс»）——按蒸汽中酒精含量的绝对变化蒸馏与 A 模式相同，但依据蒸汽中（实际上是取酒中）计算出的酒精含量。

R（取酒酒精，相对值，«Спирт в отборе отн.»）——按蒸汽中酒精含量的相对变化蒸馏与 S 模式相同，但依据蒸汽中计算出的酒精含量。

- 3 值 («Значение»)**
取决于程序类型：温度，或酒精含量的绝对值或相对值。
- 4 容器编号 («Номер емкости»)**
取酒容器编号 (0..10)。该行条件触发时，舵机转到这个位置。
- 5 电压 («Напряжение»)**
调压器的电压 (对于 SEM_AVR 是以瓦为单位的功率；列标题随调压器类型变化)。零表示忽略。在该行条件触发时应用。
- 6 设置程序 («Установить программу»)**
检查并保存程序。“清除程序” («Очистить программу») 按钮删除所有行。

文本形式的程序行：Type;Value;Container;Voltage。程序既可以在表格中编辑，也可以在表格下方的文本框中编辑；编辑后点击“设置程序” («Установить программу») 。程序运行时不能修改。

除了温度行和酒精含量行，还有 **Lua** 类型：该行由 Samovar 上文件中的 Lua 脚本控制。选择此类型时会打开“Lua 阶段” («Lua-этап») 窗口 (.lua 文件、以秒为单位的超时、脚本参数) ；文本形式的这种行形如 L;timeout;0;file.lua^parameters。详见[关于 Lua 脚本语言](#)。

程序中可以使用蒸馏釜中的酒精含量，也可以使用取酒中的酒精含量，由取酒行的类型决定。

注意：

每次转到下一个程序行时，Samovar 都会记住当前的酒精含量值，并在数学表达式中把它作为 100%。

程序检查。 点击“设置程序”时，Samovar 会检查各行，出错时显示消息。有效值：T——温度 0 到 150；A 和 P——酒精含量 0 到 100；S 和 R——严格大于 0 且小于 1 的小数 (例如 0.50) ；容器编号 0 到 10；电压 (功率) ——整数，可以为负。第一个电压不为零的行必须设定绝对值 (大于 40 V 或 400 W) ，否则 Samovar 无法计算增量的起点。

“默认”加载的程序 (图形视图见上面的截图)

A;80.00;1;0

S;0.50;2;0

S;0.30;3;0

执行过程如下：

发出“开启加热” («Включить нагрев») 命令后，Samovar 报告：

“蒸馏器加热已开启” («Включен нагрев дистиллятора»)

“转到程序行 №1” («Переход к строке программы №1»)

当蒸馏釜温度达到文件 Samovar_ini.h

中设定的值 / 打开供水阀并开启供水泵时的蒸馏釜传感器温度

#define OPEN_VALVE_TANK_TEMP 77

时，会发出打开供水阀的命令，同时发送 PWM 控制信号以开启水泵。

请注意， 使用冷却泵时，Samovar 会把水温保持在“[温度](#)” («Температура») 选项卡中作为水温传感器设定值的温度。如果需要水泵“全功率”运行，请降低设定值。

蒸馏釜开始沸腾时，Samovar 测定酒精含量并显示消息：

“蒸馏釜开始沸腾！酒精含量 XX.X” («Началось кипение в кубе! Спиртуозность XX.X»)

然后转到第一个程序行，其含义是：

- 如果蒸馏釜中的绝对酒精含量低于 80%，把舵机转到 1 号位置。

由于蒸馏釜中的酒精含量显然低于 80%，于是转去执行 2 号程序行，并显示消息：

“转到程序行 №2”

因此，从开始沸腾起，取酒流入 1 号容器，并执行 2 号程序行。

接下来：

- 当蒸馏釜酒精含量达到 1 号行开始执行时的 50% 时，把舵机转到 2 号位置并转到下一行。

相应地，Samovar 转去执行 3 号程序行并显示消息

“转到程序行 №3”

现在取酒流入 2 号容器。

也就是说：酒精含量下降了 50%，容器更换了，并转到了下一个程序行。

接下来：

转到 3 号程序行时，Samovar 测定蒸馏釜酒精含量，并把该值作为 100%。

当蒸馏釜酒精含量达到 3 号行开始时的 30% 时，显示消息

“程序执行完毕，继续取酒” («Выполнение программ закончилось, продолжение отбора»)

说明。

假设转到 3 号行时蒸馏釜酒精含量为 6%。

$6/100 \times 30$ (程序行中设定的值) = 1.8%。

也就是说, 当蒸馏釜中剩下 1.8% 的酒精时, Samovar 报告所有程序行已执行完毕, 但过程会继续。

粗酒会继续流入 3 号容器, 直到出现任一关闭加热的条件, 即:

- 蒸馏釜温度超过[设置中设定的](#)蒸馏结束温度;

- 蒸馏釜中[酒精耗尽](#)

然后 Samovar 关闭加热, 并在[水温下降 20° 或蒸馏釜冷却后](#), 停止水泵并关闭水阀。

注意: 主窗口中的“下一步” («Далее») 按钮让 Samovar 不等条件满足就转到下一行。当前行的容器和电压的应用方式与条件触发时相同。

取酒程序示例。

示例 1

带酒尾分离的相对取酒程序:

为了[准确](#)测定蒸馏釜酒精含量, 在程序开头放两行:

T;85;0;150

A;80;0;220

第一行: 蒸馏釜温度达到 85° 时, Samovar 把电压降到 150 伏, 慢慢把蒸馏釜加热到沸腾。如果需要相对于蒸馏釜中的总酒精含量准确地按容器分配取酒, 就必须这样做, 以便准确测定蒸馏釜酒精含量。

蒸馏釜开始沸腾时, Samovar 测定蒸馏釜酒精含量, 记住它, 并转到下一行:

A;80.00;0;220

其含义是: 如果蒸馏釜中的绝对酒精含量低于 80%, 把电压设为 220 伏并转到下一行。由于蒸馏釜酒精含量无论如何都明显低于 80%, 所以会转到下一行。

程序后面的行需要根据实际取酒的目的和任务来编写,

S;0.50;1;0

这一行的含义是: 取酒, 直到[蒸馏釜中的酒度](#)降到转到该行时记住的值的 50%, 把舵机转到 1 号位置, 并转到下一个程序行。

S;0.30;2;0

取酒, 直到[蒸馏釜中的酒度](#)降到转到该行时记住的值的 30%, 把舵机转到 2 号位置, 并转到下一个程序行。

P;20.00;3;0

取酒, 直到 [取酒中的酒度](#)降到 20%, 然后把舵机转到 3 号位置。

因此, 这个程序

T;85;0;150

A;80;0;220

S;0.50;1;0

S;0.30;2;0

P;20.00;3;0

的执行过程如下:

发出“开启加热”命令后, Samovar 让调压器进入“升温” («Разгон»)。蒸馏釜温度达到 85° 后, 把电压降到 150 伏, 等待蒸馏釜沸腾。蒸馏釜沸腾时, 测定蒸馏釜中的绝对酒精含量 (在数学表达式中把该值作为 100%) 并记住它; 由于蒸馏釜的绝对酒精含量显然低于 80%, 于是在加热管上设定 220 伏。

开始向 0 号容器取粗酒。

当蒸馏釜中的酒精含量为记住值的 50% 时, 把舵机转到 1 号位置, 再次记住新的蒸馏釜酒精含量 (把该值作为 100%), 并向 1 号容器取粗酒, 直到蒸馏釜酒精含量降到[新](#)记住值的 30%。

条件满足时, 把舵机转到 2 号位置, 向 2 号容器取粗酒, 直到[取酒中的酒度](#)降到 20%。然后把舵机转到 3 号位置, 向 3 号容器取粗酒, 直到出现以下任一条件:

- 蒸馏釜温度超过[设置中设定的](#)蒸馏结束温度;

- 蒸馏釜中[酒精耗尽](#)

然后关闭加热, 并在[水温下降 20° 或蒸馏釜冷却后](#), 停止水泵并关闭水阀。

示例 2

如果不需要非常精确地分离馏出物, 又想节省时间, 但仍想按蒸馏釜酒精含量把馏出物分成两部分, 可以使用以下程序:

P;60.00;0;0

S;0.50;1;0

执行过程如下：

升温后，粗酒开始流入 0 号容器。不会降低功率。当取酒中的酒度降到 60% 以下时（大约在开始后 3-5 分钟），Samovar 记住新的蒸馏釜酒精含量，并继续向 0 号容器取酒。

当蒸馏釜中的酒精含量降到记住值的 50% 时，把舵机转到 1 号位置，并向 1 号容器取酒，直到出现某个蒸馏结束条件。当然，这个程序还可以补充其他条件。

电压/功率的设定逻辑与精馏模式相同。为 0 时当前值不变；绝对值小于 40 伏（400 瓦）时，加到当前值上（小于 0 时则减去）；否则应用设定的值。

在**醪塔**模式的程序中使用同样的 T、S、A、P、R 行类型，那里的行有第五个字段——蒸汽温度。

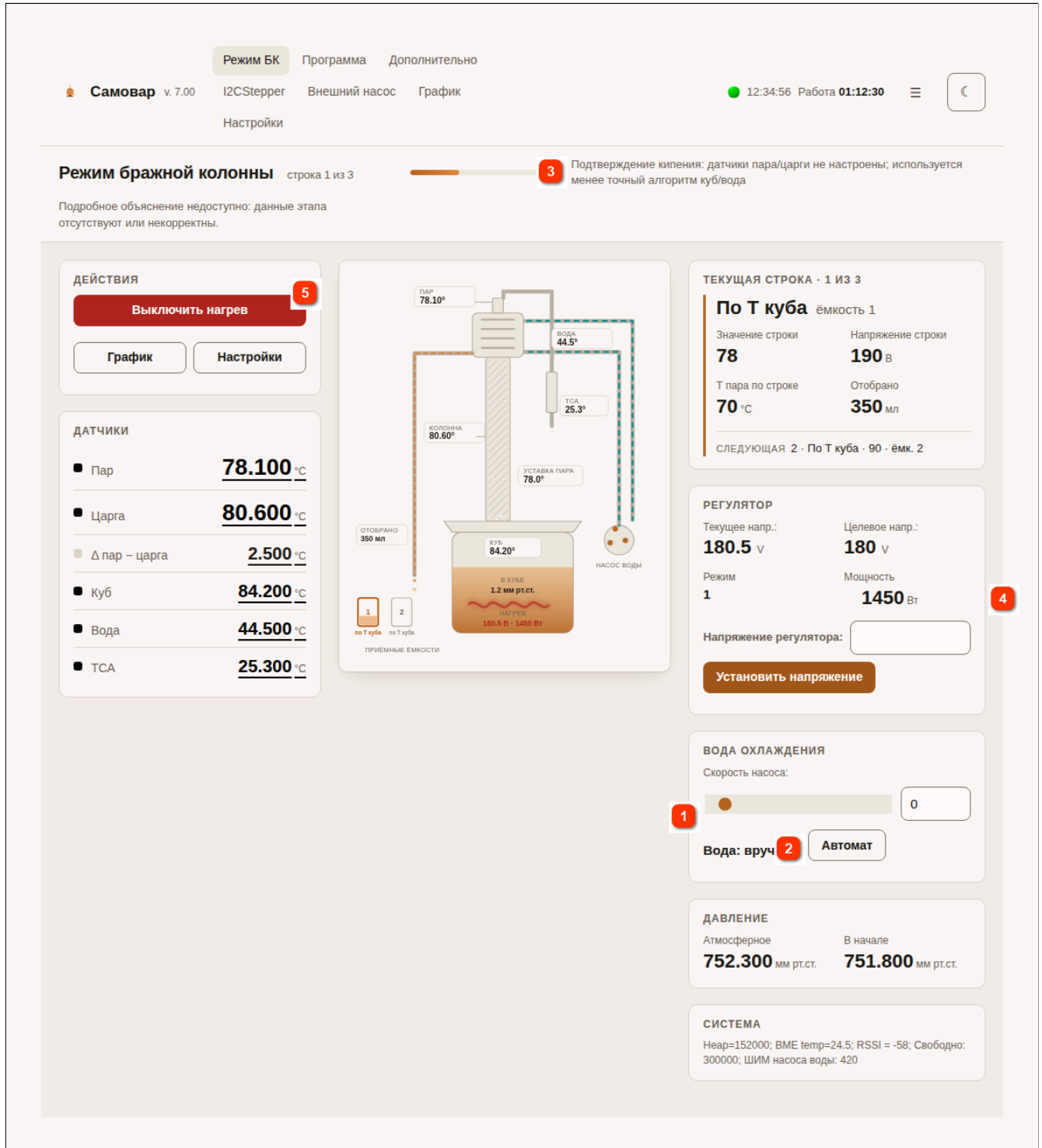
醪塔 (BK)

醪塔 (BK) 是一种装有直流式分凝器和一段短塔节的精馏塔，塔节内填充粗大或非常稀疏的填料。实践证明，它适合制取既经过净化、又保留原料风味和香气的馏出液。

产品采用“分凝器后蒸汽取酒”方式。控制塔的运行归结为两个参数：加热功率和分凝器冷却。取酒速度通过冷却泵的转速调节。

Samovar 7.00: BK 模式现在有取酒程序。程序规定：取酒持续到釜温（或酒精度）达到多少、接入哪个容器、使用什么电压、保持什么蒸汽温度。Samovar 会通过改变分凝器的供水量自行保持蒸汽温度。蒸汽温度为 0 的程序行与旧版本一样，冷却由人工控制。

运行模式主窗口



醪塔 (BK) 模式主窗口

1

冷却泵

滑块用于调节冷却液流量。

功率通过 PWM 调制改变。

既可以拖动转速刻度上的滑块控制，也可以直接在字段中输入数值。

滑块最左端为泵的最低转速（PWM_LOW_VALUE，默认 100 个单位）。

滑块在最右端时，泵直接获得 12 伏电压，不经调制。输入字段的值 = 1023。

因此为简便起见粗略估算，10 个单位 = 1% 功率。任何手动更改都会关闭蒸汽温度自动控制，直到该行结束。

- 2

“自动”（«Автомат»）

按当前程序行的蒸汽温度把冷却交还给 Samovar 控制。只有在加热开启且该行设置了蒸汽温度时，按钮才可见。
- 3

沸腾确认

Samovar 用哪个传感器确认了釜内沸腾：蒸汽和塔节是准确的标志；釜、水或它们的组合准确性较低。沸腾前显示“等待确认”（«ожидание подтверждения»）。
- 4

调压器

调压器控制。见[手动设置调压器电压](#)。
- 5

开启加热

启动和停止塔的运行。加热开启时，按钮名称变为“关闭加热”（«Выключить нагрев»）。

其余数据与“[精馏](#)”（«Ректификация»）模式主窗口中的相同

启动与升温

启动前请指定釜温传感器：没有它 Samovar 会拒绝启动，并显示消息“BK 未启动：釜温传感器未指定或无响应”（«БК не запущена: датчик куба не назначен или не отвечает»）。如果程序中有蒸汽温度大于 0 的行，还需要蒸汽传感器（“BK 未启动：程序需要蒸汽传感器”（«БК не запущена: программа требует датчик пара»））。

开启加热后，Samovar 显示消息“醪塔加热已开启”（«Включен нагрев бражной колонны»），并向调压器发送“升温”（«Разгон»）命令。

当釜温达到 77°C 时

(文件 [Samovar.ini.h](#) 中的这一行：

```
#define OPEN_VALVE_TANK_TEMP 77 //Boiler sensor temperature at which the water supply valve opens
) , Samovar 打开供水阀，并按“泵速”（«Скорость насоса»）滑块设定的速度开启冷却泵。泵的最低转速由以下行设定
//Lower value (in percent) for the PWM regulator of the pump. Choose a value that ensures water flow
#define PWM_LOW_VALUE 10
```

注意。
建议一开始就把冷却开到最大。

升温一直持续到 Samovar 确认釜内沸腾为止。沸腾的标志：蒸汽或塔节温度高于 39°C，或者釜温停止上升而水温继续上升。确认沸腾所用的传感器显示在主窗口的“沸腾确认”（«Подтверждение кипения»）行中。

釜内沸腾后，Samovar 设置预先配置的 BK 工作功率。默认值：调压器为 45 V，SEM_AVR 为 200 W。可以在[设置](#)页面的“沸腾后 BK 功率”（«Мощность БК после закипания»）字段中修改。如果程序第一行指定了电压，它会叠加在工作功率之上（小于 40 V / 400 W 时作为增量，否则作为绝对值）。随后 Samovar 转到程序第一行。

程序

Программа БК:

№	Тип программы	Значение	№ емкости	Напряжение	Т пара, °C	
01	Температура	78	1	19	70	+ -
02	Температура	90	2	0	78	+ -
03	Температура	93	2	0	0	+ -

► Описание программы БК в текстовом формате для сохранения в облаке:

► Программа БК:

Установить программу

Очистить программу

醪塔（BK）模式的“程序”（«Программа»）选项卡

- 1

程序类型

转到下一行的条件：釜温（T），釜内或取出液的酒精度，绝对值或相对值（A、S、P、R）。各类型见[蒸馏程序](#)页面。
- 2

电压

满足该行条件时施加的调压器电压（功率）。0 — 不改变。第一行的电压在沸腾开始后立即施加，叠加在 BK 工作功率之上。

蒸汽温度，°C



Samovar 在该行通过向分凝器供水所保持的蒸汽温度：0 — 人工控制供水，30..100 — 自动控制。其他数值不被接受（“程序错误：蒸汽温度：0 或 30..100”（«Ошибка программы: Т пара: 0 или 30..100»））。

程序行：Type;Value;Container;Voltage;Vapor T（类型;数值;容器;电压;蒸汽温度）。前四个字段以及行类型 T、S、A、P、R、L（Lua 阶段：.lua 文件、以秒为单位的超时和参数在“Lua 阶段”（«Lua-этап»）窗口中设置）与[蒸馏程序](#)中的相同：满足该行条件时，舵机转到指定容器，调压器接收指定电压，Samovar 转到下一行。第五个字段是该行要维持的蒸汽温度：0 — 人工控制供水，30..100 — Samovar 自行调节分凝器供水量。

程序示例：

T;78;1;190;70

T;90;2;0;78

T;93;2;0;0

沸腾开始后，Samovar 保持 70°C 的蒸汽温度（取酒头）。釜温升到 78°C 时，舵机转到 1 号容器，调压器得到 190 V，蒸汽设定值变为 78°C。釜温 90°C 时 — 转到 2 号容器，电压不变，冷却转为人工控制。釜温 93°C 时程序行结束，取酒一直持续到满足结束条件。

如果程序中没有任何行，沸腾开始后 Samovar 按以前的方式工作：由操作者手动控制功率和冷却。

分凝器冷却

自动冷却时，Samovar 每分钟把蒸汽温度与该行设定值比较一次。如果蒸汽温度比设定值高出 0.2°C 以上，供水量增加一级（1023 中的 30 个 PWM 单位）；如果偏低，则减少供水，但不会低于泵的最低值，水温已高于 65°C 时也不会减少。状态显示为标签“水：自动 78.0 °C”（«Вода: авто 78,0 °C»）或“水：手动”（«Вода: вручную»）。

只要移动“泵速”（«Скорость насоса»）滑块（或在旁边的字段中输入数值），自动控制就会关闭直到该行结束，Samovar 报告“分凝器供水：手动控制”（«Вода дефлегматора: ручное управление»）。可以用“自动”（«Автомат»）按钮重新开启自动控制；只有当前行设置了蒸汽温度时，该按钮才可见。

操作者按照本次运行的工艺卡设定所需的加热功率，并逐渐减少冷却水供应，直到冷凝器取酒口开始滴酒。

— 酒头取酒开始。

随后在取酒心时，同样通过调节加热和冷却来达到所需的取酒速度。

在此模式下 Samovar 不考虑大气压。

结束与安全

安全系统处于工作状态。

如果发生蒸汽穿透，且 TCA 传感器通过[控制单元](#)连接到取酒口，加热将被关闭。

当水温接近临界值时，Samovar 发出蜂鸣，显示“水温达到临界！”（«Критическая температура воды!»），并将调压器电压降低 5 个单位。

满足以下任一条件时，Samovar 关闭加热并显示消息“蒸馏运行结束”（«Работа бражной колонны завершена»）：

— 达到[设定的釜温](#)；

— 釜内已无酒精：釜温高于 90°C，并且在“蒸馏结束检测时间”（«Время для контроля завершения процесса дистилляции»）字段设定的时间内上升不到 0.1°（0 — 不检测）。

关闭加热后，停水方式与[蒸馏模式](#)相同：水温降到比设定值低 20° 时，或釜冷却下来时。

运行时记录日志并绘制图表。

连续醪塔

NBK 是连续醪塔：发酵醪由泵持续送入，Samovar 控制功率、进料量以及各阶段之间的切换。下面介绍该模式以及在 Samovar 网页界面中的操作顺序。
不要一上来就打开加热。先在不带功率负载的情况下检查泵、传感器、冷却和程序。

模式如何工作

NBK 程序始终由四行组成——按顺序执行的阶段：

预热 (H) («Прогрев»)。Samovar 按 H 行打开功率和进料，等待蒸汽温度达到 75 °C，然后自动转入“调整”。

调整 (S) («Настройка»)。没有时间限制的手动阶段：Samovar 按 S 行设置功率和进料量，之后由操作者自行调整。要进入下一阶段，使用“下一步” («Далее») 按钮。

优化 (O) («Оптимизация»)。自动寻找工作点：Samovar 逐级改变进料量和功率，每次改变后等待“惯性” («Инерция») 时间，并记住找到的最佳值 **Mo** («Мо») (功率) 和 **Po** («По») (进料量)。此阶段需要塔头液位传感器，并在编译固件时启用。

工作 (W) («Работа»)。Samovar 保持找到的工作点，并根据酒糟温度、蒸汽温度和压力逐步修正进料量。此阶段禁止手动控制功率和进料量。

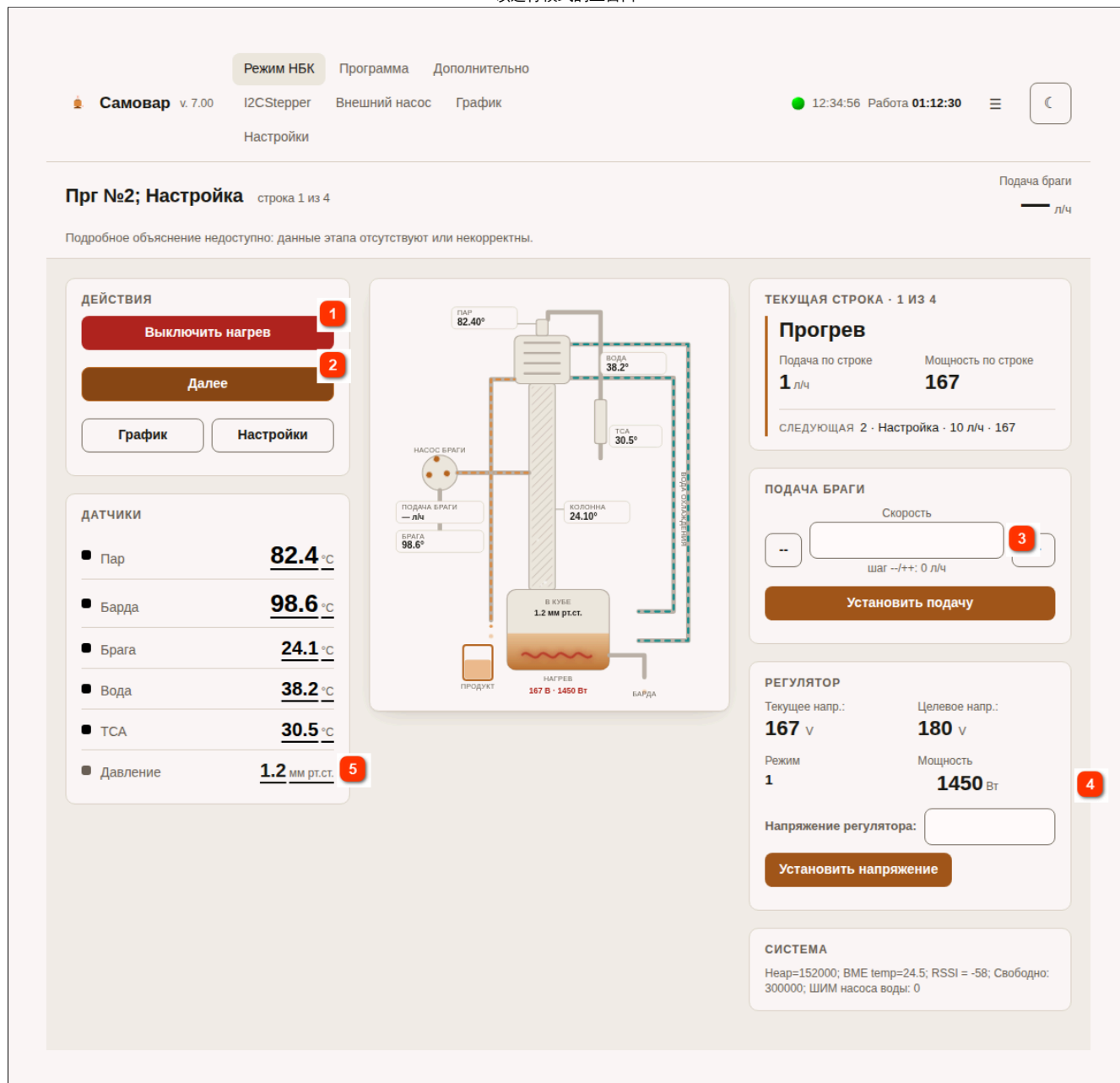
当前阶段显示在主页面的状态中，例如“程序第 2 行；调整” («Прг №2; Настройка»)。在除“调整”以外的任何阶段，只要蒸汽温度高于 98 °C，Samovar 就认为发酵醪已经用完，并以消息“发酵醪已用完。NBK 程序结束。” («Кончилась брага. Программа NBK завершена.») 结束程序。结束时显示统计：处理了多少升发酵醪、平均速度和本次运行时间。

液泛。液泛由塔头液位传感器 (DZ) 判断，或由酒糟排放口压力 (DD) 超过“液泛压力” («Давление захлёба») 阈值判断。在“调整”阶段，Samovar 把进料量降到 1/3、功率降到 1/2 并发出提示；在“优化”阶段，找到最佳值之前液泛会停止过程，找到之后则转入“工作”；在“工作”阶段，Samovar 以降低的功率和进料量暂停 (两个“惯性”时间)，然后以降低后的 Mo 和 Po 继续。如果在设置中启用了“液泛时用舵机切换流向” («Переключать поток сервоприводом при захлёбе») ，液泛期间舵机把流向转到位置 1 (脏产品)，暂停结束后再转回位置 0。

冷却。当蒸馏釜 (酒糟) 温度达到 77 °C 时，阀门和冷却泵打开，一直工作到关闭加热为止；当水温冷却到比水温传感器设定值 (设置 - 温度) 低 20° 或蒸馏釜冷却后关闭。如果 TCA 传感器升到 60 °C 或水温升到 75 °C 并持续 60 秒，Samovar 以消息“冷却不足！停机。” («Недостаточное охлаждение! Останов.») 停止过程。

示意图。主窗口中央是塔的图示，带实时读数：发酵醪泵及当前进料量 (L/h)，发酵醪、蒸汽、水和 TCA 的温度，蒸馏釜压力，产品和酒糟排放。加热开启且正在进料时泵转子转动，加热时加热管高亮，运行时显示产品瓶和液滴，暂停时示意图上出现暂停图标。应用和网站上也是同样的示意图 ([移动应用与网站](#))。

该运行模式的主窗口



NBK 模式在“调整” («Настройка») 阶段的主窗口

1

打开加热

从“预热”阶段启动 NBK 程序。运行期间该按钮名为“关闭加热”（«Выключить нагрев»），用于正常停止过程。

2

下一步

转入下一阶段：从“调整”转入“优化”，跳过“优化”（在最初 30 秒内），从安全待机恢复“工作”。

3

速度

发酵醪进料量，L/h。输入数值后点击“设定进料”（«Установить подачу»）。“-”和“+”按钮把进料量改变设置中的一个“进料步长”。只在加热开启时的“预热”和“调整”阶段可用。

4

调节器

当前电压和目标电压（功率）。手动设置在“预热”和“调整”阶段有效；在“优化”和“工作”阶段被禁止。

5

压力

DD 传感器测得的酒精排放口压力。超过设置中的“液泛压力”阈值即视为液泛。

1. 检查模式是否可用

- 固件必须带功率调节器控制功能编译。对于 PlatformIO，合适的环境有 **Samovar (KVIC)**、**Samovar_rmvk**、**Samovar_sem** 或正确配置的 **Samovar_s3**。在 **Samovar_no_power** 编译版本中 NBK 无法启动：Samovar 会回复“NBK 启动被拒绝：此编译版本中没有功率调节器。”（«Запуск НБК отклонён: регулятор мощности недоступен в этой сборке.»）
- 在主页面点击“**设置**”（«Настройки»）。在“**基本**”（«Основные»）选项卡的“**运行模式**”（«Режим работы»）字段中选择“**连续醪塔**”（«Непрерывная бражная колонна»）。
- 点击页面底部的“**保存**”（«Сохранить»）。保存后打开主页面：应当出现“**NBK 模式**”（«Режим НБК»）页面。

2. 需要连接什么

可控的功率调节器，在编译固件时选定。用手动命令在安全负载上测试它。

用于输送发酵醪的**外部 I2CStepper 泵**。它必须显示在 NBK 页面上并能接受命令。连接方法见[I2CStepper](#)一节。

蒸汽传感器是“预热”“优化”和“工作”阶段必需的。**酒精传感器**在 Samovar 中指定为蒸馏釜传感器，是“优化”和“工作”阶段必需的。指定为塔身传感器的传感器在页面上显示为“**发酵醪**”（«Брага»）。

冷却：水必须能顺畅流过冷凝器，阀门和冷却泵必须按正确方向工作。送入发酵醪之前先检查它们。

液泛保护：塔头液位传感器（DZ）和/或酒精排放口的压力传感器（DD）。热启动前请实际检查传感器能否触发。如果固件中启用了 DZ，Samovar 就认为它可用，无法发现传感器本身没有接上；因此没有警告并不能证明保护有效。如果完全没有传感器，启动时 Samovar 会警告：“注意：没有任何液泛传感器（DZ 已关闭，DD 无响应）。液泛保护不起作用。”（«Внимание: нет ни одного датчика захлёба (DZ выключен, DD не отвечает). Защита от захлёба не работает.»）自动“优化”要求在编译时启用塔头液位传感器。

第一次热启动前，打开 NBK 页面，确认温度不为零、没有明显错误，有 DD 时能显示其压力，并且 I2C 泵的状态为“已连接”（«подключён»）。

3. 校准发酵醪进料泵

校准是告诉 Samovar 输送 1 毫升需要多少步电机步数。不校准就无法正确设定以升/小时计的速度。

- 打开 **I2CStepper** 页面（主页面上的按钮）；如果有多块板，在“设备”（«Устройство»）列表中选择泵，然后点击“**外部泵校准**”（«Калибровка внешнего насоса»）。
- 把进液软管放入水中，出液口对准量杯。
- 设置一个较小且稳定的电机速度，点击“**开始校准**”（«Начать калибровку»）。
- 当量杯中正好有 100 mL 时，点击“**记录 100 mL**”（«Зафиксировать 100 мл»）。
- Samovar 会计算每毫升步数并写入板的存储器。在“**设置 → I2CStepper**”选项卡中，确认“**步数/mL**”（«Шагов/мл»）字段大于零。更换驱动器细分、电机、泵头或软管后，请重新校准。

4. 填写 NBK 设置

打开[设置 → NBK](#)。第一次启动可以保留出厂值，之后再根据自己的塔细调：

惯性——**180 s**。这是改变功率或进料量后的等待时间。如果您的塔达到新的稳定状态需要更长时间，请填入实际测得的时间。

偏差——**0.5 °C**。酒精温度相对基准温度的允许偏差。

功率步长——**100**。寻找工况时，自动控制每次把内部计算功率改变多少瓦。

进料步长——**0.5 L/h**。“优化”阶段以及按“-”/“+”按钮时进料量改变这么多；在“工作”阶段，常规修正为该步长的十分之一。

蒸汽温度——**81 °C**。算法使用的蒸汽温度下限。固件允许的值：高于 80 且不高 97 °C。

液泛压力——**40 mm Hg**。DD 的触发阈值。请根据自己的塔实际发生液泛时的数据细调，而不要照搬别人设备的数值。

结束温度（酒精）——**98.5 °C**。蒸净酒精的基准温度。在“优化”的第一次迭代中，Samovar 可能根据传感器把它调高，但不超过 102 °C。

然后打开“**其他**”（«Прочие»）选项卡，正确填写**加热管电阻**和**电网电压**。这两个值用于计算最大功率；如果其中一个为空或有误，启动会被拒绝，并显示消息“NBK 启动被拒绝：设置不正确 - ...”（«Запуск НБК отклонён: некорректная настройка - ...»）。点击“**保存**”。

5. 准备四行程序

Программа НБК

№	Тип программы	С ¹ ть	Напряжение
01	Прогрев	1	167
02	Настройка	10	167
03	Оптимизация	0	0
04	Работа	0	0

► Программа НБК:

► Как работает автоматика и что влияет на НБК:

► Описание программы НБК в текстовом формате для сохранения в облаке:

Установить программу

Очистить программу

НБК模式的“程序”（«Программа»）选项卡：四行固定程序

1

速度

该阶段的发酵醪进料量，L/h。“预热”和“调整”必须大于零；“优化”和“工作”中零表示“采用当前值或找到的值”。

2

电压 / 功率

该阶段的加热功率：KVIC 和 RMV-K 为伏，SEM_AVR 为瓦。“预热”中零表示计算出的最大功率；“调整”必须为非零值。

- 打开“NBK 模式”页面，然后打开“程序”（«Программа»）选项卡。程序必须正好有四个阶段：**预热** → **调整** → **优化** → **工作**。文本形式为 $H;rate;power;S;...;O;...;W;...$ 各行；任何其他的行组合或顺序都不被接受。
- 在“**预热 (H)**”行中设置非零的进料量。如果功率为零，Samovar 使用计算出的最大功率。蒸汽温度达到 75 °C 时预热自动结束。
- 在“**调整 (S)**”行中务必设置非零的进料速度和功率。先从您的 NBK 和调节器制造商推荐的安全值开始。
- 在“**优化 (O)**”和“**工作 (W)**”行中，零表示使用“调整”或“优化”阶段得到的值。如果跳过自动优化，请事先在 W 行中设置非零的速度和功率。
- 点击“**设定程序**”（«Установить программу»）。确认四行中的数值保留下来，并且没有出现格式错误消息。
- 功率单位取决于调节器：SEM_AVR 以瓦为单位，KVIC 和 RMV-K 以伏为单位；选项卡上的列标题会相应变化。所有行中的速度都以升/小时为单位。不要未经核对自己的加热管和调节器就照搬别人的功率值。

6. 开始预热

- 给蒸汽发生器加水，确认酒糟能够排出，把泵的进液口放入准备好的发酵醪中，并打开冷却水通路。检查软管没有被压扁。
- 在“NBK 模式”选项卡上再次检查温度、压力和泵的状态。
- 点击“**打开加热**”（«Включить нагрев»）。Samovar 会提示“启动 NBK 程序。预热”（«Запуск программы НБК. Прогрев»），状态变为“预热”。Samovar 按 H 行打开功率和进料。
- 不要离开设备。检查液体确实在流动、没有泄漏、酒糟正常排出、冷却正常工作。蒸汽传感器达到 75 °C 时，Samovar 会自动转入“调整”阶段。

7. 进行手动调整

- 转入“**调整**”后，至少等待一个“**惯性**”时间，让温度和出酒稳定下来。
- 每次只改变一个参数。在“调节器电压”（«Напряжение регулятора»）字段中输入功率，点击“**设定电压**”（«Установить напряжение»）（SEM_AVR 为“**设定功率**”（«Установить мощность»））；在“发酵醪进料”（«Подача браги»）区块的“**速度**”（«Скорость»）字段中输入速度，点击“**设定进料**”（«Установить подачу»）。-- 和 ++ 按钮把进料量改变一个“**进料步长**”。
- 每次改变后，同样至少等待一个“**惯性**”时间。观察蒸汽温度、酒糟温度、压力、酒糟是否稳定排出以及有无液泛。
- 发生液泛时，Samovar 会提示液泛来源，并把实际进料量降到三分之一、功率降到一半。不要马上把参数调回去；等液泛迹象消失，并找出原因。
- 如果在“调整”阶段蒸汽温度在 100 °C 或以上保持 60 秒，Samovar 会停止过程：蒸汽发生器可能已经干烧。
- 工况稳定后，打开“**附加**”（«Дополнительно»）选项卡，点击“**设为最佳值**”（«Установить как оптимальные»）。然后回到主选项卡，点击“**下一步**”。

Скорость: — л/ч Напряжение: 167 V

1

Установить как оптимальные

Расход воды: 0 мл

Скорость расхода воды: 0 л/мин

NBK 模式的“附加” («Дополнительно») 选项卡

1

设为最佳值

把当前功率和实际进料量记为 M_0 和 P_0 。只在“调整”和“工作”阶段有效。

8. 完成优化

进入“优化”后有 30 秒时间，可以用“下一步”按钮跳过它。如果什么都不按，且塔头液位传感器已启用，Samovar 就从 O 行的功率和进料量（如果为零，则从当前值）开始自动寻找工作点。

如果酒精温度和蒸汽温度不低于设定限值，Samovar 把当前值记为最佳值，并把进料量增加一个“进料步长”。

如果温度低于限值，Samovar 把进料量减少一个步长，并把功率增加一个“功率步长”。每次改变后等待“惯性”时间。

找到最佳值后，在发生液泛、达到工作压力上限或其他搜索限制（进料上限 30 L/h、功率上限）时，自动控制转入“工作”。消息“优化完成...” («Оптимизация завершена...») 会给出 M_0 ——找到的功率和 P_0 ——找到的进料量；转入“工作”之前，Samovar 以降低的功率和进料量暂停两个“惯性”时间。

如果固件中没有启用塔头液位传感器，自动优化不会启动。Samovar 会提示“无法优化 - 没有液泛传感器...” («Оптимизация невозможна - отсутствует датчик захлэба...»)）。因此，W 行的非零功率和速度必须在点击“打开加热”之前设置并保存好。当 Samovar 转入“优化”并显示该消息时，点击“下一步”。过程运行期间不能修改程序。如果没有事先准备好 W 行，请关闭加热，等待停止，修改程序后重新启动。不要让设备一直等待：10 分钟后 Samovar 会进入安全待机，关闭加热和进料。

9. 监控“工作”阶段

在“工作”阶段，禁止手动设置功率和进料量：由算法控制（消息“在 NBK 的优化和工作阶段不能手动控制。” («Ручное управление недоступно на Оптимизации и в Работе НБК.») ）。要改变工作点，先正常停止过程，修改程序或设置，然后重新启动。

如果酒精温度或蒸汽温度降到设定限值以下，Samovar 会逐步减少进料量。如果酒精温度骤降（发酵醪溢流），进料量会立即减少一个完整的“进料步长”。

如果压力一直接近工作上限，进料量会减少。如果酒精温度持续高于基准温度，进料量可能逐步增加，但不会超过找到的最佳值。

发生液泛时，Samovar 暂时降低进料量和加热，暂停一段时间，然后以降低后的最佳值继续。暂停结束前不要干预，除非出现危险迹象。

“附加”选项卡上的“设为最佳值”按钮在“工作”阶段同样有效：它用当前功率和实际进料量替换 M_0 和 P_0 。

如果蒸汽温度升到 98 °C 以上，Samovar 认为发酵醪已用完，并正常结束程序。

10. 如何停止过程以及收到消息时怎么做

正常停止时，点击运行期间名为“关闭加热” («Выключить нагрев») 的按钮，并确认操作。等待页面上进料量归零、加热关闭。Samovar 会提示“NBK 运行结束” («Работа НБК завершена») 。

消息“安全待机” («Безопасное ожидание») 表示 Samovar 已经停止了泵和加热。先排除提示的原因。在“工作”阶段，用“下一步”按钮恢复；在“预热”“调整”或“优化”阶段，该按钮会结束本次运行，之后需要重新启动。

收到冷却不足、压力传感器故障 (“超过 60 秒没有读数” («Нет показаний более 60 секунд»)) 或泵、调节器命令未确认等消息时，在排除原因之前不要尝试继续过程。

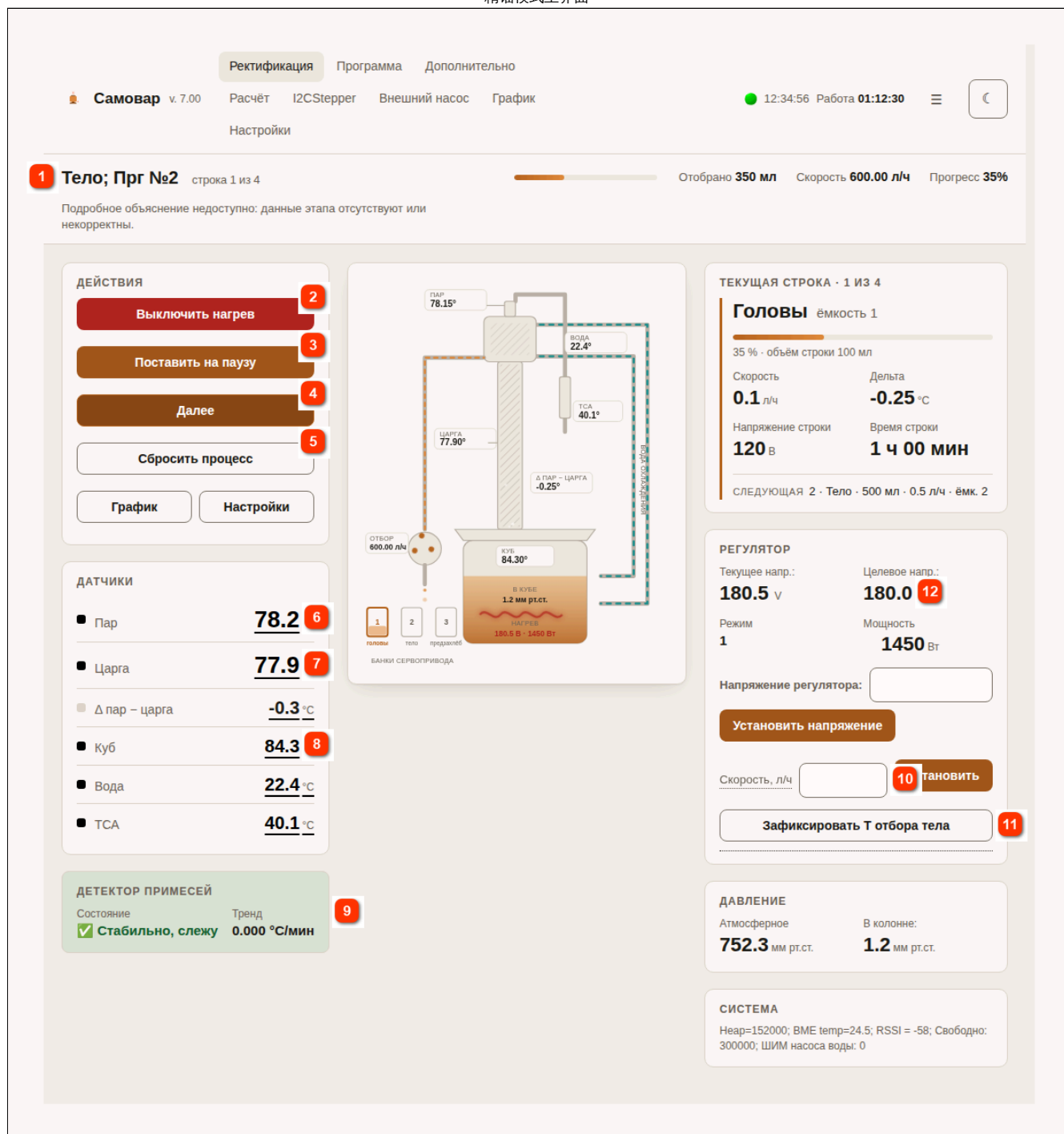
结束后，等待温度下降、冷却正常关闭。之后才能关水、拆开热的接头或给设备断电。

每次启动前的快速检查

- . 已选择 NBK 模式，设置已保存。
- . I2CStepper 泵已连接、已校准，泵送方向正确。
- . 蒸汽和酒精传感器显示的温度合理；DZ/DD 已检查。
- . 冷却已打开并正常工作；酒精排放通畅。
- . 程序有 H、S、O、W 四行；H 有进料量，S 有进料量和功率。
- . 加热管电阻和电网电压填写正确。
- . 在进入稳定的“工作”阶段之前，操作者一直守在设备旁。

精馏模式

精馏模式主界面



精馏模式主界面，“精馏”（«Ректификация»）选项卡

- 1 **状态（«Статус»）**
Samovar 的当前状态：正在执行的程序行的类型和编号、暂停、记录的酒心温度。
- 2 **打开加热（«Включить нагрев»）**
以升温模式打开功率调压器。加热打开期间，按钮变为“关闭加热”（«Выключить нагрев»）。
- 3 **暂停（«Поставить на паузу»）**
停止取酒泵。要继续，请按“继续”（«Продолжить»）。
- 4 **开始取酒（«Начать отбор»）**
启动取酒程序。取酒期间该按钮名为“下一步”（«Далее»），用于让 Samovar 转到下一程序行。
- 5 **重置过程（«Сбросить процесс»）**
确认后停止取酒并关闭加热。

蒸汽温度（«Температура пара»）

- 6** 分凝器前的蒸汽传感器。用于判断升温结束和精馏塔稳定。
- 7** **塔身温度 («Температура в царге»)**
位于塔高 2/3 处的传感器。
- 8** **蒸馏釜温度 («Температура в кубе»)**
用于打开水阀 (77°C)，并在达到设定限值时结束程序。
- 9** **杂质检测器 («Детектор примесей»)**
杂质穿透检测器的状态：观察、降速或暂停。
- 10** **速度, L/h («Скорость, л/ч»)**
手动更改当前行的取酒速度。输入数值并按“设置” («Установить»)。
- 11** **记录酒心取酒温度 («Зафиксировать Т отбора тела»)**
将当前蒸汽温度和塔身温度设为酒心取酒温度。
- 12** **调压器 («Регулятор»)**
调压器的设定电压 (功率) 和实际电压 (功率)。

Samovar 7.00: 取酒速度和“记录酒心取酒温度” («Зафиксировать Т отбора тела») 按钮已从“附加” («Дополнительно») 选项卡移到主界面的“调压器” («Регулятор») 卡片中。新增了 [杂质穿透检测器](#)。执行完最后一条程序行后，Samovar 不会立即关闭加热，而是让精馏塔以“全回流运行”模式保持 30 分钟。

开机或重启后，Samovar 处于“关闭” («Выключено») 状态。

在此状态下，它显示传感器读数，但不控制任何东西。如果在网页界面中按“打开加热” («Включить нагрев») 按钮，Samovar 会以升温模式打开功率调压器 (如果已连接)，并等待精馏塔预热。如果设置中没有分配蒸汽、水和蒸馏釜传感器，加热不会打开，屏幕上出现消息“传感器未分配” («Датчики не назначены»)

也可以用 [万能按钮](#) 启动升温。

在蒸馏釜温度达到 77°C 之前，[辅助加热管](#) (如果已连接) 与主加热管一起工作。

当蒸馏釜温度达到 77°C 时

[Samovar_ini.h](#) 文件中的这一行：

```
#define OPEN_VALVE_TANK_TEMP 77 //打开供水阀门时的蒸馏釜传感器温度
```

Samovar 接通 3 号继电器 (假定其上确实连接了打开供水的阀门)，发出提示音并显示消息：“请打开供水！” («Откройте подачу воды!»)。无论如何都会出现这条通知 - 即使没有阀门或已手动打开。启动时冷却泵以启动功率运行几秒钟，然后 PWM 调节器调整转速，以保持设定的水温，

[Samovar_ini.h](#) 文件中的这一行：

```
//水泵 PWM 调节器的下限值 (百分比)。请选择能保证有水流
```

```
#define PWM_LOW_VALUE 10
```

当蒸汽温度达到 39 度时

[Samovar_ini.h](#) 文件中的这一行：

```
#define CHANGE_POWER_MODE_STEAM_TEMP 39 //精馏塔从升温模式切换到工作模式时的蒸汽传感器温度
```

Samovar 关闭升温模式，命令调压器设置取酒程序第一行中指定的电压 (功率)，并切换到精馏塔稳定模式。通知“升温完成。稳定/全回流运行。” («Разгон завершён. Стабилизация/работа на себя.») 发送到智能手机 (如果已配置)，并显示在主界面上。

同时，它用冷却泵保持冷却水温度。蒸馏釜开始沸腾时，Samovar 估算并显示釜内液体的酒精度。

Samovar 等待蒸汽温度在六分钟内的波动不超过 0.1 度。此后即认为稳定完成。网页界面中显示消息“稳定完成，精馏塔运行稳定。” («Стабилизация завершена, колонна работает стабильно.»)。六分钟计数只考虑蒸汽传感器；读数变化超过 0.1° 时，计数重新开始。

这一逻辑在 [alarm.h](#) 文件中实现

是否开始取酒由操作者决定。如果决定开始，请在网页界面、移动应用中按“开始取酒” («Начать отбор») 按钮，或使用 [万能按钮](#)。如果精馏塔尚未稳定，网页界面会要求确认开始。开始前，Samovar 会检查程序：水泵必须已校准，第一行必须设置绝对电压 (功率)，每条取酒行都必须设置体积或转换温度；否则取酒不会开始，原因会在消息中显示。

命令发出后，[取酒程序](#) 启动，蠕动泵按程序中设定的速度开始工作。分配单元的摇臂移到取酒程序中指定的位置。

在**酒头取酒**程序的所有行执行完之前，Samovar：

- 取出程序中指定的体积，
- 按程序规定在各取酒容器之间切换液流
- 可以按程序改变取酒速度

同时，它忽略塔身温度和蒸汽温度的变化。也就是说，它只是按程序取酒。在酒头阶段，杂质检测器只做观察。

与酒头取酒相关的程序行执行完后，Samovar 转入**酒心取酒**

酒心取酒程序第一行开始时，Samovar 记下蒸汽温度和塔身温度，此后以这些记录的温度为准。如果酒心行设置了自己的电压 (功率)，温度不会立即记录，而是在新功率下蒸汽温度稳定之后再记录。

注：

在酒心取酒程序行运行期间，可以随时按主界面“调压器” («Регулятор») 卡片中的“记录酒心取酒温度” («Зафиксировать Т отбора тела») 按钮来修正这些温度。Samovar 随即用当前传感器读数覆盖之前记录的酒心取酒温度。

注：

此外，在任何程序行运行期间，都可以随时在主界面“调压器”卡片的“速度, L/h” («Скорость, л/ч») 字段中输入所需的取酒速度 (升/小时)，然后按“设置” («Установить») 按钮来改变取酒速度。新速度只对当前程序行有效。

如果在设置中启用了“使用气压修正”（«Использовать корректировку давления»）选项，Samovar 会借助气压传感器把所有温度传感器的读数换算到 760 mmHg 气压下。这样，记录的酒心取酒温度就不受运行期间大气压变化的影响：气压下降时实际沸点降低，但换算后的数值保持不变，不会出现误暂停。

酒心取酒程序行一直工作，直到收集到取酒程序中指定的体积。

如果在酒心取酒期间蒸汽或塔身传感器的温度超出设置中规定的范围，Samovar 会暂停取酒。这就是所谓的“启停”（Start-Stop）模式。详见[酒心取酒](#)页面。

- 如果在设置中启用了杂质检测器和“温度上升时降低取酒速度”（«Снижать скорость отбора при росте температуры»）参数，那么蒸汽温度上升时，Samovar 会按设置中的步长降低取酒速度，温度严重上升时则暂停取酒。

- 如果在同一行内因温度连续三次暂停取酒，Samovar 会把该行的基础速度降低 10%。

酒心取酒程序行执行完后，如果程序中有**酒尾取酒**行，Samovar 转入这些行。最后一条程序行结束时，Samovar 停止取酒泵，把摇臂移到位置 0，并保持加热 — 精馏塔全回流运行 30 分钟（消息“程序执行完毕。全回流运行（30 min 后自动关闭）。”（«Выполнение программы завершено. Работа на себя (30 мин до автоотключения).»））。在此期间，可以按“下一步”（«Далее»）立即结束过程，也可以向程序中添加行。30 分钟后，Samovar 关闭加热，等待精馏塔冷却并关闭供水。

如果在任何程序行运行期间按“下一步”按钮，Samovar 在确认后转到下一行。程序运行期间也可以按“暂停”（«Поставить на паузу»）按钮，或再次使用 [万能按钮](#)，Samovar 会暂停水泵。要继续取酒，需要按“继续”（«Продолжить»）按钮，程序随即继续执行。“重置过程”（«Сбросить процесс»）按钮在确认后停止取酒并关闭加热。

如果蒸馏釜温度达到设置中“精馏、蒸馏和 BK 直至蒸馏釜温度”（«Ректификация, дистилляция и БК до температуры в кубе»）字段设定的值，Samovar 会结束程序并关闭加热，显示消息“达到蒸馏釜最高温度限值。程序已结束。”（«Лимит максимальной температуры куба. Программа завершена.»）。

问题与解答。FAQ

如何判断精馏塔已经稳定？

Samovar 自己判断精馏塔是否稳定。升温结束后，如果蒸汽传感器读数在 6 分钟内变化不超过 0.1°，即认为精馏塔已稳定。

能否在精馏塔稳定之前开始取酒？

可以。Samovar 不禁止这样做，但网页界面会要求确认开始。

Samovar 会发消息提示该开始取产品了吗？它有酒质传感器吗？

不会。是否开始酒心取酒由您自己决定。[这里](#)用通俗的语言介绍了 Samovar 在精馏模式下的工作方式。

精馏模式的取酒程序

Samovar 7.00: [程序计算器](#)现在可以计算设备参数和推荐速度，并带有“应用推荐”（«Применить рекомендации»）按钮；设置程序前请检查功率单位：伏还是瓦。“程序”（«Программа»）选项卡中增加了“清除程序”（«Очистить программу»）按钮。设置程序时会进行检查：第一行必须设定绝对电压（功率），每个取酒行都必须设定体积或转换温度。

“程序”（«Программа»）选项卡

Samovar 开机或重启后，“程序”（«Программа»）选项卡显示最后一次设置的取酒程序；Samovar 会把它保存在内存中。新设备上加载的是一个示例（三行 H-B-H，每行 450 ml），[需要根据自己的情况进行调整](#)。

取酒程序主窗口包含以下项目：

精馏模式下的“程序”（«Программа»）选项卡

- 1 类型（«Тип»）**
该行的取酒模式：酒头（«Головы»）、酒心（«Тело»）、预液泛（«Предзахлеб»）、酒尾（«Хвосты»）、暂停（«Пауза»）。
- 2 出量（«Выход»）**
取酒体积，单位毫升。对于“暂停”行，为以秒为单位的暂停时间。
- 3 速度（«Скорость»）**
取酒速度，升/小时。
- 4 容器（«Ёмкость»）**
接取的容器编号（0-10）。0 号位置为原点位置。
- 5 电压（«Напр.»）**
调压器电压（对于 SEM_AVR 为以瓦为单位的功率）。0 表示不改变；较小的数或负数表示相对当前值的偏移。
- 6 时间（«Время»）**
该行的计算执行时间。
- 7 添加 / 删除行**
加号在下方添加一个空行；减号删除当前行。
- 8 合计（«Итого»）**
程序的总执行时间（不含暂停）。
- 9 取酒程序说明（«Описание программы отбора»）**
用于自由文本的可折叠字段，开始取酒时发送到云端。
- 10 取酒程序（«Программа отбора»）**
以文本格式显示程序的可折叠字段。与表格关联。
- 11 计算程序（«Рассчитать программу»）**
进入程序计算器。

设置程序（«Установить программу»）

- 12

检查程序并写入 Samovar 内存。
- 13

清除程序 («Очистить программу»)

确认后删除所有程序行。

各列说明

No. - 程序行号。添加或删除行时自动编号。正在执行的行号显示在“精馏” («Ректификация») 主选项卡的状态行中。

类型 («Тип») - 下拉菜单, 选择取酒模式名称: [酒头 \(«Головы»\)](#)、[酒心 \(«Тело»\)](#)、[预液泛 \(«Предзахлеб»\)](#)、[酒尾 \(«Хвосты»\)](#)、[暂停 \(«Пауза»\)](#)、[Lua](#)。根据所选类型, 采用不同的逻辑。表格中的行按类型着色。

出量 («Выход») - 以毫升为单位设定: 取出多少后转到下一行程序。对于“暂停”行, 此字段设定以秒为单位的暂停时间; 对于“Lua”行, 设定以秒为单位的阶段超时。

Lua - 由 Lua 脚本控制的行: 选择此类型会打开“Lua 阶段” («Lua-этап») 窗口, 在其中选择 Samovar 内存中的 .lua 文件、以秒为单位的超时以及脚本参数。这种行不设定速度、容器和电压, 第一行电压的要求也不适用于它。详见: [关于 Lua 脚本语言](#)。

速度 («Скорость») - 蠕动泵的取酒速度。以升/小时为单位设定。

容器 («Ёмкость») - 接取产品的容器编号。从零开始编号 (0-10)。也就是说, 舵机摆臂的“原点”位置就是 0 号容器。

电压 («Напр.») - 需要在[调压器](#)上设定的电压, 单位为伏。零表示不改变当前值; 较小的数或负数表示相对当前值的偏移 (见[编辑取酒程序](#))。

注意。 对于 SEM_AVR 调压器, 数值以瓦为单位设定, 该列名称为“功率” («Мощн.»)。

时间 («Время») - 执行某一行程序所需的时间 (体积除以速度)。(自 5.25 版起)

表格中没有显示行的另一个字段——**转换温度** (文本格式中的第五个值)。它只能在“取酒程序” («Программа отбора») 文本框中设定; 编辑表格时该值会保留。为 0 时不使用。如果设定 0 到 20 之间的数, 当蒸汽温度超过该行开始时的蒸汽温度加上这个数时, 该行结束。如果设定 20 或更大的数, 当蒸汽温度超过该值时, 该行结束。转到下一行取决于哪个条件 (体积或温度) 先满足。

程序

取酒所依据的表格形式的程序。

取酒程序逐行执行, 执行完最后一行后结束。可以添加和删除行。点击行末的加号 (+) 会添加下一个空行; 点击减号 (-) 会删除当前行。程序设置之后, 操作无法撤销。

所需参数在行与列的交叉处设定。

每一行描述 Samovar 在某一时刻应执行的操作。一行的运行时间由以下之一决定: 泵以设定速度需要输送的液体体积, 或转换温度, 或明确设定的时间 (对于“暂停”行)。表格与“取酒程序”文本框直接关联。

当前正在执行的行以深红色斜体突出显示, 已执行过的行为斜体。在编辑表格单元格时, 用 Samovar 数据更新表格会被推迟; 标题旁的“数据已更新” («Данные обновились») 图标会提示这一点。

注意。

Samovar 接受执行的程序最多 20 行。行数更多的程序不会被设置 (提示“行数过多” («слишком много строк»))。

总时间

“合计” («Итого») 字段是整个取酒程序的执行时间 (不含“暂停”行)。

取酒程序说明

自由文本 (最多 250 个字符)。关于此取酒程序的任何信息。

▼ Описание программы отбора в текстовом формате для сохранения в облаке:

Описание программы отбора:

Произвольный текст.

“取酒程序说明” («Описание программы отбора») 字段

可折叠的文本框。可以输入与某个取酒程序相关的信息。说明通过“设置程序” («Установить программу») 命令与程序一起保存在 Samovar 中, 并在用“开始取酒” («Начать отбор») 命令开始会话时一次性发送到云端。之后只能在云端编辑该文本。详见: [云端日志和图表](#)。

取酒程序

取酒所依据的文本格式的程序。

▼ Программа отбора:

Текст программы отбора:

```
H;100;0.1;1;0;120  
B;500;0.5;2;0;100  
C;3000;1.2;3;0;0  
T;300;1;4;0;100
```

- **Тип отбора**

H - головы (Heads)

B - тело (Body)

P - пауза (Pause)

T - хвост (Tail)

- **Объем отбора (мл)**

- **Скорость отбора (л/ч)**

- **Номер емкости для отбора**

- **Температура отбора**

Если значение Температуры равно 0 - не используется.

Если значение Т указано и больше 20°C, то при Т пара больше, чем указанная температура, произойдет переход на следующую строку программы.

Если значение Т указано больше 0 и меньше 20°C, то при Т пара больше, чем сумма Т пара, которая была запомнена при переходе на эту строку программы и Т в программе, произойдет переход на следующую строку программы.

- **Напряжение отбора**

Пример программы отбора:

```
H;100;0.1;1;0.00;120  
P;120;0.00;0;0.00;140  
B;5000;0.75;2;0.00;90
```

Текстовый формат "Программы отбора" ("Программа отбора")

Скрываемый текстовый блок. Программа отбора в текстовом формате: шесть значений, разделенных запятой: тип; объем (мл); скорость (л/ч); номер емкости; температура; напряжение (мощность). В текстовом блоке ниже каждого значения дано объяснение и пример программы. Текст можно скопировать и сохранить как обычный текстовый файл на компьютере. Он связан с таблицей. Данные будут скопированы в облако при выполнении команды "Начать отбор". В облаке нельзя редактировать текст.

Вычисление программы («Вычислить программу»)

Открыть отдельную страницу, перейти к [калькулятору программы](#), рассчитать объем и процентное содержание в зависимости от объема и процента. Если в таблице есть несохраненные изменения, Samovar потребует подтверждения перехода.

Настройка программы

Скопировать программу в Samovar. После нажатия будет проверена программа, и она будет установлена в память Samovar для последующего выполнения. Если программа не прошла проверку (ошибка формата, более 20 строк, первая строка не содержит абсолютное напряжение, отбор не имеет заданного объема или заданной температуры), то не будет установлено, причина будет указана в сообщении вместе с номером строки.

Удаление программы

После подтверждения, удалить все программы из памяти Samovar.

编辑取酒程序

如何创建和修改取酒程序。

创建取酒程序有几种方法：

直接在“程序”（«Программа»）选项卡的表格中创建；

在“计算器”（«Калькулятор расчета»）中创建；

在“Calculation AS-3.xlsx”（«Расчет AC-3.xlsx»）文件中创建，该文件位于 Samovar 草稿（sketch）文件夹的根目录。

第一种方法。直接编辑程序：根据自己的需要在表格中添加/删除行，并设置所需参数。有些参数可以从下拉菜单中选择，有些需要自己填写，并遵守一定的条件。

当前正在执行的程序行以深红色斜体突出显示，已执行的行为斜体。

Программа отбора:

Рассчитать программу

№	Тип	Выход	Скорость	Ёмкость	Напр	Время	
01	Головы	450	0.07	0	160	06:25	+ -
02	Головы	150	0.1	1	165	01:29	+ -
03	Предзахлеб	300	0.5	2	170	00:36	+ -
04	Предзахлеб	3000	1.2	3	0	02:30	+ -
05	Предзахлеб	3000	1.2	4	0	02:30	+ -
06	Тело	3000	1	5	5	03:00	+ -
07	Хвосты	1000	0.9	6	7	01:06	+ -

Итого: 17:38

► Описание программы отбора в текстовом формате для сохранения в облаке:

► Программа отбора:

Установить программу

Очистить программу

取酒过程中的程序表格：正在执行第 3 行

第一行最好直接在行内编辑：

类型（«Тип»）- 从下拉菜单中选择“**酒头**”（«Головы»）

出量（«Выход»）- 设定取酒量，单位毫升。

速度（«Скорость»）- 升/小时

容器（«Ёмкость»）- 接取所用容器的编号。注意编号从 0（零）号位置开始

电压（«Напр.»）- 设定您的**调压器**实际使用的值。这是快速升温结束后 Samovar 命令调压器设定的电压。对于 SEM_AVR 调压器，该列名称为“功率”（«Мощн.»），以瓦为单位设定。

时间（«Время»）- 自动计算。在该列底部累加为总计。

(+) 加号 - 在下方添加一行。

(-) 减号 - 删除当前行。

第二行及以后各行。

可以编辑预设程序中的其余各行，和/或按 (-) 删除它们，按 (+) 在下方添加一行并填入数值。

编辑方法与第一行相同，但在电压设置中还可以使用其他条件：

可以明确设定所需数值，例如 160。

或者：

- 如果电压设为 0（零），则不改变调压器：保持当前实际施加的电压。

例如：在“**预泼洒**”（«Предзахлеб»）模式中，上一行设定了 175 伏，但到转入设为 0.00 的行时，Samovar 已把电压升到 180 伏。新的取酒行将正好施加 180 伏。

- 如果设定的值不超过 40 伏，则把该电压加到当前电压上。

例如：如果上一行设定了 160 伏（或者 Samovar 自己把电压升到了 160 伏），那么执行设为 5 伏的行时，Samovar 会在实际电压上加 5 伏（160+5）。

- 如果设定负值，则从当前电压中减去该电压。

例如：如果上一行设定了 175 伏（或者 Samovar 自己把电压升到了 175 伏），那么执行设为 -5（负 5）伏的行时，Samovar 会从实际电压中减去 5 伏（175-5）。

对于 SEM_AVR 调压器（瓦），修正值与绝对值的分界为 400 W。

这些电压设定条件适用于所有取酒模式，包括“暂停”（«Пауза»）行。
必须满足的条件：程序第一行的电压必须设为实际值（大于 40 伏或 400 瓦）。否则程序不会被设置，也不会启动，提示中会说明原因：如果不这样，快速升温后精馏塔会一直保持全功率。

在表格中编辑程序时，程序会以文本格式同步显示在“取酒程序”（«Программа отбора»）字段中

▼ Программа отбора:

Текст программы отбора:

H;450;0.07;0;0;160
H;150;0.1;1;0;165
C;300;0.5;2;0;170
C;3000;1.2;3;0;0
C;3000;1.2;4;0;0
P;3000;1.5;0.5

• Тип отбора

H - головы (Heads)
B - тело (Body)
P - пауза (Pause)
T - хвост (Tail)

• Объем отбора (мл)

• Скорость отбора (л/ч)

• Номер емкости для отбора

• Температура отбора

Если значение Температуры равно 0 - не используется.
Если значение Т указано и больше 20°C, то при Т пара больше, чем указанная температура, произойдет переход на следующую строку программы.
Если значение Т указано больше 0 и меньше 20°C, то при Т пара больше, чем сумма Т пара, которая была запомнена при переходе на эту строку программы и Т в программе, произойдет переход на следующую строку программы.

• Напряжение отбора

Пример программы отбора:

H;100;0.1;1;0.00;120
P;120;0.00;0;0.00;140
B;5000;0.75;2;0.00;90

同一程序显示在“取酒程序”（«Программа отбора»）文本框中
该字段与表格相互关联。表格中的修改会立即显示在该字段中；同样，如果修改该字段中的值，表格中的值也会随之改变。
行的第五个值——转换温度——不在表格中显示，只能在该字段中设定（见[精馏模式的取酒程序](#)）。编辑表格时它不会丢失。

这样就可以把程序复制到文本文件中保存，反过来也可以通过复制粘贴把自己的任何程序（包括端端的程序）粘贴到该字段中。
可以粘贴以前某次蒸馏使用过的程序，必要时在表格中快速调整。这个功能让您不必每次都重新输入程序。

取酒程序创建好之后，必须在同一选项卡上按“设置程序”（«Установить программу»），把它设置到 Samovar 中。Samovar 会检查程序，如果发现错误，会显示带行号的提示；此时程序不会被修改。

每次调整程序后，如果希望 Samovar 接受修改，都要发出“设置程序”命令。

注意！
收到“开始取酒”（«Начать отбор»）命令后，Samovar 开始执行程序的第一行。如果在执行过程中修改了当前正在执行的程序行并发出“设置程序”命令，不会有任何变化。
Samovar 会继续按该行开始时设定的体积、速度、容器和电压执行当前程序行。

也就是说，如果操作者在 Samovar 已经开始取酒后决定修改程序，修改后的取酒程序只有尚未开始执行的部分会被 Samovar 接受执行。

提示。
如果想通过改变当前正在执行的过程来修改程序，有几种方法：
- 在正在执行的行之后紧接着添加一个空行，在新行中设定所需数值（考虑已经接取的体积），发出“设置程序”命令，然后在主选项卡上按“下一步”（«Далее»）按钮（取酒期间“开始取酒”按钮就叫这个名字）。Samovar 会要求确认转到下一行。
- 切换到[手动控制](#)。这时 Samovar 会继续按程序执行取酒量和容器部分。取酒速度可以在主选项卡“调压器”（«Регулятор»）卡片的“速度，л/ч”（«Скорость, л/ч»）字段中修改，电压在同一卡片的电压字段中修改。修改在转到下一行之前有效。

Samovar 7.00 - 用户手册, 01.10.2026

102 / 179

程序计算器

Samovar 7.00: 程序计算器现在包含设备参数、液泛功率、工作功率和速度的计算、“应用建议” («Применить рекомендации») 按钮以及 V/W 单位检查。将结果转入程序之前，请检查所选的单位。

“程序” («Программа») 选项卡上有一个“计算程序” («Рассчитать программу») 键；点击它即可进入程序计算器页面，其中有一个根据粗酒体积和酒精度计算程序的示例。

该计算器用于根据各馏分的百分比快速编写取酒程序，并立即为即将进行的运行设置程序。

Samovar v. 7.00

Расчет программы отбора **Фрукты** ¹ диаметр колонны **2 дюйма** ²

ПАРАМЕТРЫ ОБОРУДОВАНИЯ

Диаметр выбран выше (1.5 / 2 / 3 дюйма). Высота: **1 м** | Насадка: **80%**

Мощность ТЭНа для расчета: **3000** ³ Вт

Захлёб: **171 В** Рабочая: **148 В** Макс.отбор: **500 мл/ч**

Головы: **0.330 л/ч** | Тело: **0.600...1.000 л/ч** | Тарелок: **39.2**

Стратегия "Фрукты": Плёночный режим (низкое ФЧ) для сохранения аромата.

ПРИМЕНИТЬ РЕКОМЕНДАЦИИ

Программа в вольтах, как текущий регулятор.

Объем спирта-сырца **12.25** ⁴ л | Содержание АС в с-с **34** ⁵ % | Выход спирта **94** ⁶ %
 Объем голов **8** ⁷ % | Объем хвостов **5** ⁸ % | Кубовый остаток **6.84** ⁹ %

Информация о параметрах программы отбора

Итоги программы отбора

Головы

От АС Объем
8% **354 мл**
 Время
2 ч 54 мин

По строкам: **100%** — распределено полностью

Тело

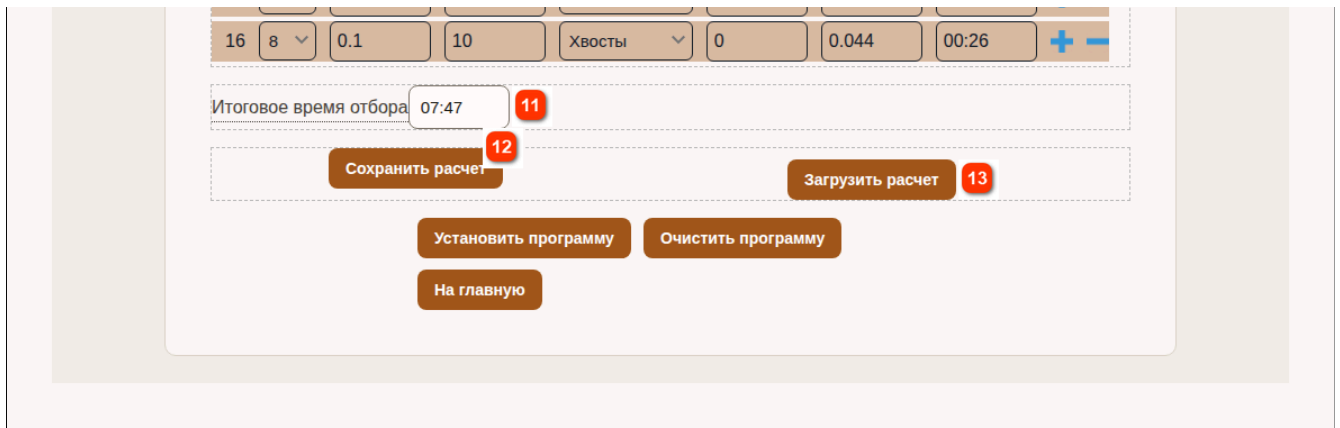
От АС Объем
87% **3855 мл**
 Время
4 ч 02 мин

По строкам В+С: **100%** — распределено полностью

Хвосты От АС: **5%** Объем: **88 мл** Время: **0 ч 39 мин** По строкам: **20%** — информационно

Итого по программе Отбор: **4297 мл** Время: **7 ч 47 мин** Паузы: **0 ч 11 мин**

№	Ёмкость	Скорость	Процент	Тип	Напр	Выход	Время	
01	0	0.07	30	Головы	135	0.106	01:30	+ -
02	0	0.1	20	Головы	0	0.071	00:42	+ -
03	1	0.2	20	Головы	0	0.071	00:21	+ -
04	2	0.3	20	Головы	0	0.071	00:14	+ -
05	3	0.4	10	Головы	140	0.035	00:05	+ -
06	0	300	0	Пауза	145	300	00:05	+ -
07	4	1	10	Тело	140	0.385	00:23	+ -
08	4	0.9	40	Тело	0	1.542	01:42	+ -
09	0	200	0	Пауза	173	200	00:03	+ -
10	0	200	0	Пауза	170	200	00:03	+ -
11	5	1.1	40	Предзахлеб	3	1.542	01:24	+ -
12	6	0.9	5	Предзахлеб	0	0.193	00:12	+ -
13	6	0.7	3	Предзахлеб	0	0.116	00:09	+ -
14	6	0.5	2	Тело	140	0.077	00:09	+ -
15	7	0.2	10	Хвосты	140	0.044	00:13	+ -



取酒程序计算器

- 1 取酒程序计算 («Расчет программы отбора»)**
原料类型：水果、谷物或糖。加载相应的模板。
- 2 塔径 («Диаметр колонны»)**
1.5、2 或 3 英寸，用于计算设备参数。
- 3 用于计算的加热管功率 («Мощность ТЭНа для расчета»)**
取自设置（加热管电阻和电网电压）。用于计算液泛功率和工作功率。
- 4 粗酒体积 («Объем спирта-сырца»)**
粗酒体积，单位为升。
- 5 粗酒中纯酒精含量 («Содержание АС в с-с»)**
粗酒中纯酒精的含量，单位为 %。
- 6 出酒度 («Выход спирта»)**
塔出口处预计的酒精度，单位为 %。
- 7 酒头体积 («Объем голов»)**
预计的酒头产出，单位为 %。
- 8 酒尾体积 («Объем хвостов»)**
预计的酒尾产出，单位为 %。
- 9 釜残液 («Кубовый остаток»)**
自动计算：蒸馏后蒸馏釜中的残液，单位为 l。
- 10 计算表 («Таблица расчёта»)**
程序行：容器、速度、馏分百分比、类型、电压（功率）以及计算出的产出和时间。
- 11 总取酒时间 («Итоговое время отбора»)**
按设定参数进行精馏所需的时间。
- 12 保存计算 («Сохранить расчет»)**
下载包含程序行的 programbackup.txt 文件。
- 13 加载计算 («Загрузить расчет»)**
将之前保存的计算文件加载到计算器中。

Samovar 文件系统中（与网页界面放在一起）存有三个模板文件：

program_shugar.txt,
program_grain.txt,
program_fruit.txt.

它们包含糖、谷物和水果的取酒程序计算模板，作为基础计算加载。草图的 tools 文件夹中也有同样的文件。模板中每一行的功率同时以伏特和瓦特写出（以 3480 W / 230 V 的参考加热管为准）；计算器根据调压器类型选择使用哪一列。

在“取酒程序计算” («Расчет программы отбора») 下拉菜单中选择原料类型（水果、谷物、糖），并在旁边选择“塔径” («Диаметр колонны»)（1.5、2 或 3 英寸）。

设备参数

此区块显示“设置” («Настройки») 选项卡“设备（塔）参数” («Параметры оборудования (колонны)») 一节中设定的填料高度和填料密度（“直径” («Диаметр»)、“填料高度” («Высота насадки»)、“填料密度” («Плотность насадки»)），以及“用于计算的加热管功率” («Мощность ТЭНа для расчета») 字段，该值根据设置中的加热管电阻和电网电压算出。计算器据此计算液泛功率、工作功率、最大取酒量 (ml/h)、酒头和酒心的推荐速度以及理论塔板数。

“应用建议” («Применить рекомендации») 按钮将计算出的速度和功率填入程序各行。如果推荐功率大于您的加热管功率，建议不会被应用，原因会在消息中显示。按钮下方有提示，说明所加载模板中的功率以何种单位（伏特或瓦特）表示。

计算数据

计算时需要输入您的数据：“粗酒体积”（«Объем спирта-сырца»）（l）、“粗酒中纯酒精含量”（«Содержание АС в с-с»）（%），并设定塔出口处预期的酒精度——“出酒度”（«Выход спирта»）（%）。还可以为您的粗酒设定酒头和酒尾的百分比（“酒头体积”（«Объем голов»）、“酒尾体积”（«Объем хвостов»），取决于所用原料的类型）。

“釜残液”（«Кубовый остаток»）、每行的“产出”（«Выход»）、“时间”（«Время»）和“总取酒时间”（«Итоговое время отбора»）均自动计算。表格下方显示各馏分的合计：酒头、酒心和酒尾各占多少纯酒精、体积和时间。

计算表的列为：序号（«№»）、容器（«Ёмкость»）、速度（«Скорость»）（l/h；对于“暂停”（«Пауза»）行为暂停时间，单位秒）、百分比（«Процент»）（该类型馏分所占体积份额）、类型（«Тип»）、电压（«Напр.»）（或功率（«Мощн.»））、产出（«Выход»）、时间（«Время»）。用加号和减号添加和删除行，与“程序”（«Программа»）选项卡相同。

将鼠标悬停在字段名称上时，会出现提示，说明如何正确填写或在什么条件下使用。例如（“取酒程序参数说明”（«Информация о параметрах программы отбора»）提示）：

处理糖发酵醪的粗酒时

塔高取最大，填料装填紧密

取酒：酒头用膜状模式，酒心用最大分离模式（预液泛）

酒头 10%-15%，酒尾 5%

处理水果粗酒时

塔高一米，填料装填疏松

以膜状模式开始取酒

取完酒头后提高分离度，但不要达到预液泛

酒头 8%，酒尾 5%

出酒度 94%

处理谷物粗酒时

塔高一米，填料装填疏松

以膜状模式开始取酒

取完酒头后提高分离度，但不要达到预液泛

酒头 6-10%，酒尾 5-7%

出酒度 94%

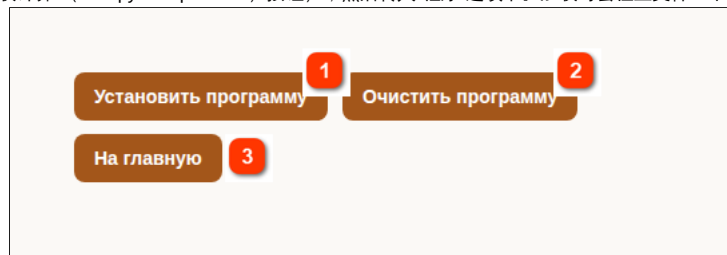
这些计算结果可以直接使用，加载到 Samovar 中执行。为此，请执行“设置程序”（«Установить программу»）命令。发送之前，计算器会检查程序和功率单位：如果与调压器不匹配，会出现消息“程序的功率单位与调压器不匹配。请点击‘应用建议’。”（«Единицы мощности программы не совпадают с регулятором. Нажмите «Применить рекомендации.»»）。

出现消息“程序已保存。”（«Программа сохранена.»）后，程序将显示在主页面的“程序”（«Программа»）选项卡上，可以立即开始运行。

您可以在计算器中按需调整程序，再次执行“设置程序”命令，然后开始运行。

计算结果可以保存到本地文件夹。执行“保存计算”（«Сохранить расчет»）命令后，名为 programbackup.txt 的文件会保存到浏览器的“下载”文件夹。程序行写入文件时功率同时以伏特和瓦特表示，因此保存前必须设定加热管功率。

将保存的计算加载到计算器中（“加载计算”（«Загрузить расчет»）按钮），然后转入“程序”选项卡。加载时会检查文件：不超过 20 行，格式正确。



计算器按钮

- 1 设置程序（«Установить программу»）**
检查计算结果并将程序写入 Samovar。
- 2 清除程序（«Очистить программу»）**
确认后从 Samovar 内存中删除程序。
- 3 返回主页（«На главную»）**
不设置程序，离开计算器。

“清除程序”（«Очистить программу»）按钮在确认后从 Samovar 内存中删除程序。如需不保存也不设置程序就离开计算器，请点击“返回主页”（«На главную»）。

说明。

在计算器中完成的计算，既可以立即转入 Samovar 的“程序”选项卡，也可以保存为计算文件。之前完成的计算只能通过计算器加载到 Samovar 中执行。

警告。

请注意，计算器用于在运行前即时计算取酒程序。重新加载页面后，计算结果不会保留。保存的文件只存储程序行：粗酒体积、酒精度以及酒头和酒尾的百分比不会写入其中，因此加载计算时数据字段保持“默认”值，计算器会立即按这些值重新计算程序。要让保存的计算重新符合实际，请在数据字段中输入您自己的数值。

如需预先计算并随后保存，建议使用文件“Расчет AC-3.xlsx” (“AS-3 计算”) ——一个用于生成程序的 Excel 文件。输入取酒参数（粗酒体积、酒精度、酒头量、酒尾量），即可得到 Samovar 的取酒程序。随时可以从文件中复制现成的程序，粘贴到“程序”选项卡的文本框中，执行“设置程序”命令，即可开始取酒。该文件随每个版本一起下载，解压后位于 Samovar 文件夹的根目录。

酒头取酒

要运行酒头取酒程序，可在“程序” («Программа») 选项卡的下拉菜单中选择“酒头” («Головы») 模式，也可以在文本框中手动编写一行。行首写拉丁字母 H (Heads, 酒头)。

取酒程序行大致如下：

H;250;0.10;1;0.00;120

所有数值之间必须用分号分隔。

第一个参数 (H) 是取酒类型，这里是 H (酒头)。

第二个参数 (250) 与取酒类型之间用分号隔开，它规定转到下一行之前需要取出多少液体。本例中为 250 毫升。

第三个参数 (0.10) 位于分号之后，是 Samovar 取出指定体积时的速度；示例第一行中为每小时 100 毫升，即每小时 0.1 升。

第四个参数 (1) 规定液体收集到哪个容器，示例中为 1 号容器。

第五个参数 (0.00) ——转换温度。0 —— 不使用，该行按体积结束。0 到 20 之间的值 —— 当蒸汽温度超过该行开始时的蒸汽温度加上此值时，该行结束。20 及以上的值 —— 当蒸汽温度超过此值时，该行结束。体积和温度两个条件以先满足者为准。两者至少要设置一个，否则程序不会被接受。此字段不显示在“程序” («Программа») 选项卡的表格中，只能在文本框中设置。

第六个参数 (120) 规定 Samovar 在执行这一行时给调压器设定的电压，即 120 伏 (对于 SEM_AVR 调压器则为功率，单位瓦)。程序第一行必须是实际值：大于 40 伏 (400 瓦)。其余各行中，0 —— 不改变电压，不大于 40 的数 —— 加到当前值上，负数 —— 从当前值中减去。

酒头取酒算法中，Samovar 不考虑设定值，也不考虑塔节温度传感器与蒸汽温度传感器之间的温差。即使某个阶段温差变为负值，Samovar 也会继续取酒。酒头行开始时，之前记录的酒心取酒温度会被清除。在酒头阶段，[杂质检测器](#)只做观察：显示温度趋势，但不改变速度，也不暂停 —— 酒头阶段蒸汽温度上升是正常过程。其他所有参数 (回流液位传感器、TCA 温度、水流量等) 都由 Samovar 监控和处理。回流液位传感器触发时，Samovar 发出提示音，显示消息“液泛传感器已触发！” («Сработал датчик захлёба!»)，并把电压降低一档 (见[预液泛](#))。

如果在设置中启用了“使用巴氏段上方的 I2C 泵” («Использовать I2C-насос над ЦП») 选项，并且 Samovar 启动时检测到了这种泵，那么在酒头行中它将成为主取酒泵，并按该行设定的速度运行。

注意：当前正在执行的程序行的取酒速度和调压器电压可随时在[手动模式](#)中修改 —— 位于主选项卡的“调压器” («Регулятор») 卡片中。修改在下一行开始之前有效。

酒心取酒

Samovar 7.00: 启用杂质检测器后，出现警告时会降低取酒速度，进入临界状态时会暂停过程。手动恢复后，大约一分钟内暂不执行自动响应；反复的停止暂停可能会降低当前行的基础速度。

当 Samovar 开始执行酒心取酒程序时，会记住蒸汽传感器和塔节传感器的温度。此后 Samovar 在运行中以这些温度为依据。记录的温度显示在主选项卡的状态行中，消息中会出现“新的酒心温度：蒸汽 = ...，塔节 = ...”（«Новые Т тела: пар = ..., цапра = ...»）。

注意：某些情况下需要重新记录温度。例如，在程序运行时提高了加热管的电压，塔节温度会随之变化，可能超出设定值，从而使得取酒暂停。为避免这种情况，可随时在主选项卡“**精馏**”（«Ректификация»）的“**调压器**”（«Регулятор»）卡片中点击“记录酒心取酒温度”（«Зафиксировать Т отбора тела»）按钮，之后将使用塔节和蒸汽的当前温度。该按钮仅在酒心（«Тело»）、预液泛（«Предзахлеб»）和暂停（«Пауза»）行上有效。

警告！

转到新的酒心或预液泛行时，Samovar 的处理方式如下：

- 如果该行设定了自己的电压（不为 0），之前的温度被清除，在新功率下蒸汽温度稳定后记录新温度，但最迟不超过 10 分钟。在此之前不执行设定值暂停；

- 如果该行电压为 0，温度保持不变；如果还没有记录温度（酒头之后或“暂停”行之后的第一个酒心行），则立即记录，与手动点击“记录酒心取酒温度”按钮相同。

如果在酒心取酒期间（酒心或预液泛程序行）蒸汽和/或塔节传感器的温度超出设置中指定的设定值，Samovar 会暂停取酒。

设定值在“设置”（«Настройки»）→“温度”（«Температура»）选项卡中设置：“分凝器前蒸汽传感器超出设定值”（«Вылет за уставку датчика пара перед дефлегматором»）和“塔身 2/3 处塔节传感器超出设定值”（«Вылет за уставку датчика в царге на 2/3 колонны»）（°C），以及“蒸汽温度超出设定值时的取酒延迟”（«Задержка отбора при выходе Т пара за уставку»）和“塔节温度超出设定值时的取酒延迟”（«Задержка отбора при выходе Т в царге за уставку»）（秒）。超出值为 0 时，该传感器的控制被关闭。

当传感器温度升高到记录温度加设定值时，Samovar 停止泵，发出提示音，并显示消息“蒸汽温度暂停”（«Пауза по Т пара»）或“塔节温度暂停”（«Пауза по Т царги»）。经过设定的延迟时间且温度降到阈值以下（留有 0.07° 余量，以免暂停在边界上反复切换）后，取酒恢复（消息“自动暂停后继续取酒”

（«Продолжаем отбор после автоматической паузы»））。如果在同一行内累计出现三次这样的暂停（来自传感器或检测器），Samovar 会将该行的基础速度降低 10% 并发出通知。

可以取消温度超出设定值时的取酒暂停：Samovar 不暂停，而是记录新的酒心温度。

满足以下条件时取酒不会暂停：

对于酒心或预液泛行。

它是酒头行之后的第一个酒心或预液泛行（中间的“暂停”行不计）。

例如：

- . 酒头（«Головы»）
- . 酒心（«Тело»）
- . 预液泛（«Предзахлеб»）
- . 酒心
- . 酒尾（«Хвосты»）
- . 等等。

执行第 2 行时，Samovar 不会暂停取酒，而是记录新的蒸汽和塔节温度。执行第 3 行和第 4 行时，取酒会暂停。

对于预液泛行

如果预液泛行位于取酒程序的任意位置，并且其后跟着两行任意组合的酒心或预液泛行（其中第一行也可以是“暂停”行）。

例如：

- . 酒头
- . 酒心
- . 暂停（«Пауза»）
- . 预液泛
- . 预液泛
- . 酒心

如果在执行第 2 行（酒头之后的第一个酒心行）和第 4 行（其后还有两行酒心类行的预液泛行）时温度超出设定值，Samovar 不会暂停取酒，而是记住新的蒸汽和塔节温度。执行第 5 行和第 6 行时，如果温度超出设定值，取酒会暂停。

温度的自动上调是有限度的：在一行内，Samovar 相对于该行第一次记录的蒸汽温度最多上调 0.3°。超过这个限度后，超出设定值会引起正常暂停。

该功能在 Samovar_ini.h 文件中启用。对应的行为

```
#define USE_BODY_TEMP_AUTOSSET
```

//对酒头之后的第一个酒心或预液泛取酒程序，以及位于倒数第二个酒心或预液泛取酒程序之前的预液泛程序，使用酒心温度自动修正

标准版本中该功能已启用。可以在 user_config_override.h 文件中关闭它，上调限度由 Samovar_ini.h 中的 #define BODY_TEMP_AUTOSSET_MAX_RISE 0.3f 一行设定。

- 如果在设置中启用了“使用气压修正”（«Использовать корректировку давления»）选项，Samovar 会借助压力传感器把温度传感器读数换算到 760 mm Hg 气压下。这样记录的酒心取酒温度就不受蒸馏过程中大气压变化的影响：气压下降时，实际沸点降低，但换算后的数值保持不变，不会出现误暂停。

杂质穿透检测器

在设置中通过“使用检测器”（«Использовать детектор»）复选框启用（不勾选时检测器完全不工作）。在酒心和预液泛行上，检测器监测温度上升速度（°C/min）——使用蒸汽传感器，当蒸馏釜温度高于 92.5° 时使用塔节传感器。检测器会在平稳阶段测量自身的温度噪声，然后自行确定触发阈值。检测器状态显示在主选项卡的卡片中：“稳定”（«стабильно»）、“上升”（«рост»）（或“正在降速”（«снижаю скорость»））、“穿透”（«проскок»）。

- 当上升速度高于阈值（警告）时，如果勾选了“温度上升时降低取酒速度”（«Снижать скорость отбора при росте температуры»）复选框，Samovar 会按“降速步长”（«Шаг снижения скорости»）（默认 5%，最大 20%）逐步降低取酒速度，但不低于该行基础速度的 70%。上升停止后，速度以同样的步长逐步恢复到基础速度。不勾选此复选框时，检测器只报告温度上升。

- 在酒头之后的第一个酒心行上，检测器警告不会降低速度，而是记录新的酒心温度（同样在 0.3° 范围内）。

- 出现临界上升（高于阈值 2.5 倍，连续两次测量）时，Samovar 暂停取酒，发出提示音，并显示消息“检测器：临界趋势！取酒暂停”（«Детектор: Критический тренд! Пауза отбора»）。经过传感器延迟时间且上升停止后，取酒自动恢复（消息“检测器：暂停后继续取酒”（«Детектор: Продолжаем отбор после

паузы»))。

- 在一行开始后的前 30 秒内，以及手动点击“继续” («Продолжить») 后的一分钟内，检测器不介入。在酒头之后的第一个酒心行上，它会等待蒸汽温度稳定（最长 10 分钟，状态为“等待蒸汽稳定” («Жду стабилизации пара»))。

在酒头阶段检测器只做观察；在酒尾阶段它使用高 1.2 倍的阈值工作。

注意：

在酒心取酒模式下，回流液位传感器同样工作；如果有液泛危险，Samovar 会根据传感器的指令，每 40 秒把加热管电压降低一个调压器档位，直到传感器不再发出指令（见[预液泛](#)）。降低后的电压一直作为工作电压，直到操作者干预，或执行到设定了自己电压的程序行。

酒心取酒程序的各行一直执行，直到收集到取酒程序中指定的体积或达到转换温度（该行第五个值），然后 Samovar 转到下一行。

程序最后一行结束后，Samovar 停止取酒，把取酒单元的摇臂移到位置 0，让精馏塔以“全回流运行” («работа на себя») 模式保持 30 分钟，然后关闭加热，等塔冷却后关闭水。如果提前达到设置中设定的蒸馏釜最高温度，Samovar 结束程序并立即关闭加热。

注意：当前正在执行的程序行的取酒速度和调压器电压可随时在[手动模式](#)中修改——位于主选项卡的“调压器”卡片中。手动设定的速度成为检测器的基础速度。

预液泛

“预液泛” («Предзахлеб») 程序

自动把精馏塔保持在预液泛状态。此模式的工作方式与**酒心取酒**模式相同，但增加了把精馏塔保持在预液泛状态的功能。Samovar 会自行升高和降低调压器上的电压。也可以在自动调节之外手动调节电压，手动调节优先。

此模式需要回流液位传感器，并且要在设置中启用“精馏时使用回流液位传感器” («Использовать датчик уровня флегмы при ректификации») 选项。

工作原理。

警告。所有调压器的时间间隔相同，但功率调整量不同。每种调压器型号都使用一个称为“调节步长”的常数。

自动降压为 2 个步长，升压为 1 个步长。

“预液泛”程序行开始时，Samovar 设定该行中指定的电压并等待 4 分钟。如果在此期间回流液位传感器没有触发，就把电压升高 1 个步长。然后再等待 4 分钟，如果传感器仍未触发，再次升高电压，依此类推。

传感器触发时，Samovar 记住当前电压，并进行第一次降压，降低 2 个步长（消息“电压从 ... 降低” («Напряжение снижаем с ...») ）。然后等待 40 秒，期间忽略传感器触发；之后如果信号仍然存在，再次降压并再等待 40 秒。降压一直持续到回流传感器不再触发为止。

传感器最后一次触发 48 秒后，Samovar 把电压升高到之前记住的传感器首次触发时的电压减去所用调压器的 2 个调节步长（消息“电压升高到 ...” («Напряжение повышаем до ...») ）。然后等待 4 分钟。如果在此期间传感器仍未触发，再把电压升高 1 个步长，如此反复，直到传感器触发。

循环开始重复。

每个新循环中，液位传感器首次触发时的电压都会在 Samovar 内存中被覆盖。传感器停止触发后，电压正是回到这个值减去 2 个调压器步长。经过几次循环后，精馏塔进入稳定且几乎恒定的预液泛状态。

转到下一行时，如果下一行不是“预液泛”，计数器和记住的电压会被清除。如果“预液泛”行后面还是同类的行，循环不重新开始而是继续进行。

注意。

回流液位传感器的一次触发是指传感器 LED 持续亮 3 秒。如果 LED 以短于 3 秒的间隔闪烁，则不算触发。

注意。

Samovar 以事件发生时的实际电压值为准。因此在调整模式时可以手动改变电压，快速得到稳定的结果。

例如：肉眼看回流液面已经离开了视镜，但传感器 LED 仍然亮着，Samovar 继续降压。在我们调整传感器的同时，功率已经降了不少。没有必要等 Samovar 自动升高，可以在调压器上或通过 Samovar 手动升高。Samovar 的下一步将从实际电压开始。

要使用此模式，在下拉菜单中选择“预液泛”，文本框中的程序行大致如下：

C;2500;1.00;3;0.00;170

(C —— 保持预液泛模式；2500 —— 取酒体积，ml；1.00 —— 取酒速度，l/h；3 —— 取酒到 3 号容器；0.00 —— 转换温度，0 表示不使用；170 —— 加热管上设定的电压)

Kvic 调压器示例。调压器步长为 0.5 伏。

假设“预液泛”模式的取酒行设定为 170 伏

取酒程序执行到这一行时，Samovar 通过调压器给负载施加 170 伏电压并等待 4 分钟。如果回流液位传感器没有触发，就增加 0.5 伏并等待 4 分钟。如果仍未触发，再增加 0.5 伏，再等待 4 分钟。如此反复，直到传感器触发。

传感器触发时（假设在 174 伏），Samovar 记住这个电压并降低 1 伏。然后等待 40 秒，期间忽略传感器信号。如果仍在触发，再降低 1 伏，如此每隔 40 秒一次，直到传感器不再发出信号。

假设电压降到了 168 伏。那么在最后一次触发 48 秒后，Samovar 把电压升高到记录的液泛电压减 1 伏 (174-1=173)，再次等待 4 分钟。如果传感器没有触发，就增加 0.5 伏。

Samovar 配 RMVK 调压器。调压器步长为 1 伏

假设“预液泛”模式的取酒行设定为 170 伏

取酒程序执行到这一行时，Samovar 通过调压器给负载施加 170 伏电压并等待 4 分钟。如果回流液位传感器没有触发，就增加 1 伏并等待 4 分钟。如果仍未触发，再增加 1 伏，再等待 4 分钟。如此反复，直到传感器触发。

传感器触发时（假设在 174 伏），Samovar 记住这个电压并降低 2 伏。然后等待 40 秒，期间忽略传感器触发。如果仍在触发，再降低 2 伏，依此类推，直到传感器不再发出信号。

假设电压降到了 168 伏。那么在最后一次触发 48 秒后，Samovar 把电压升高到记录的液泛电压减 2 伏 (174-2=172)，再次等待 4 分钟。如果传感器没有触发，就增加 1 伏。

Samovar 配 SEM AVR 调压器。

与 RMVK 调压器类似，但这种调压器设定的是功率（瓦）而不是电压（伏），步长按当前功率的百分比计算：降低 3%，触发结束后升高到记住的功率减 3%，每 4 分钟的定时增加为 1%。

例如：当前功率为 1800 W。传感器触发时，Samovar 降低 54 W (3%)，定时增加时增加 18 W (1%)。

注意

由于功率的设定方式，实际值可能与 Samovar 设定的值略有不同（例如 1803 W 而不是 1800）。这完全没有影响。

如果需要修改回流液位传感器的响应时间，在这里设置：

在 alarm.h 文件中：

//如果已经响应过 —— 等待 40 秒，因为过程有惯性

alarm_h_min = millis() + 1000 * 40;

40 是回流液位传感器触发时再次降压之前的秒数。安装了回流液位传感器时，这在任何模式下都有效，因为它属于防止精馏塔液泛的安全功能：在“预液泛”行之外，传感器触发时 Samovar 还会发出提示音，并显示消息“液泛传感器已触发！” («Сработал датчик захлёба!»)。

在 Samovar_pin.h 中：

```
#define WHLS_ALARM_TIME 3 //回流液位报警触发前的秒数
```

由于回流液波动，传感器可能会闪烁；此参数规定液位传感器连续触发多少秒才算真正的触发。

仅适用于预液泛运行：

在 *Samovar.h* 中：

```
#define TIME_C 4 // 预液泛模式程序的等待时间，单位分钟
```

—— 这是回流液位传感器未触发时 Samovar 升高电压之前的时间。触发结束后恢复电压之前的停顿为这个时间的五分之一（48 秒）。

酒尾取酒

“酒尾馏分取酒” («Отбор хвостовых фракций») 程序

酒尾取酒。（不要与从酒尾中提取酒精混淆。）这是专门用于取出酒尾馏分的程序：芳香水，或脂肪酸含量较高的产品等。

要使用此模式，在程序行开头写拉丁字母“T” (Tail, 酒尾)。酒尾取酒程序行大致如下：

T;2500;1.00;5;0.00;176

所有功能与酒心取酒相同，只是不考虑塔节温度与蒸汽温度之差，也不考虑它们超出设定值。酒尾行开始时，之前记录的酒心取酒温度会被清除。

取酒不会因蒸汽和塔节传感器的设定值而暂停。[杂质检测器](#)如果在设置中启用，在酒尾阶段会继续收集历史数据并显示温度趋势，但不会降低取酒速度，也不会暂停取酒。

也就是说，Samovar 在固定电压下按设定速度把回流液取到指定容器中，直到收集到设定体积或达到转换温度（该行第五个值），并在[设置](#)中“精馏、蒸馏和 BK 直到釜温” («Ректификация, дистилляция и БК до температуры в кубе») 字段设定的蒸馏釜温度时结束取酒。

注意：

最后一个程序行结束后，Samovar 停止泵，将摇臂移到位置 0，并继续保持加热 30 分钟：精馏塔全回流运行。在此期间可以按“下一步” («Далее») 立即结束过程，或向程序中添加行。30 分钟后加热关闭。如果蒸馏釜温度提前达到设置中设定的上限，程序结束，加热立即关闭。

暂停

“暂停”程序

拉丁字母 P 表示暂停 (Pause)。取酒程序行可以是这样的：

P;120;0;0;0;172

含义是：等待 120 秒，把调压器电压设为 172 伏：

执行 P 类型的行时，Samovar 停止取酒泵，并等待该行设定的时间。时间以秒为单位，写在该行的第二个值中（在“程序”（«Программа»）选项卡的表格中是“输出”（«Выход»）列），必须大于零。其他参数（特别是蒸汽和塔节温度）都被忽略，只有第六个参数——电压除外。电压的规则与取酒行相同：0 —— 不改变，较小的数或负数 —— 相对当前值的增减。主选项卡的状态中会显示暂停和剩余时间。

注意：

在任何运行模式下，Samovar 都不会忽略为安全而监控的参数。

例如，需要大幅提高（或降低）调压器电压时，可以使用暂停。由于蒸馏釜和精馏塔的惯性相当大，这需要时间。等待多久，需要根据自己的经验来决定。

注意！

此程序行的特点是：它开始时，Samovar 会清除内存中之前记录的塔节温度和蒸汽温度。如果下一程序行是酒心或预液泛行，且其中未设定电压（0），则转到该行时记录转换时刻的当前温度。如果下一行设定了自己的电压，则稍后记录温度——在新功率下蒸汽温度稳定之后（不超过 10 分钟）。也就是说，这相当于操作者想设定新的“塔节”（«Царга»）和“蒸汽”（«Пар»）温度值，并在主选项卡的“调压器”（«Регулятор»）卡片中实际点击了“记录酒心取酒温度”（«Зафиксировать T отбора тела»）（见[酒心取酒](#)）。

问题与解答。常见问题

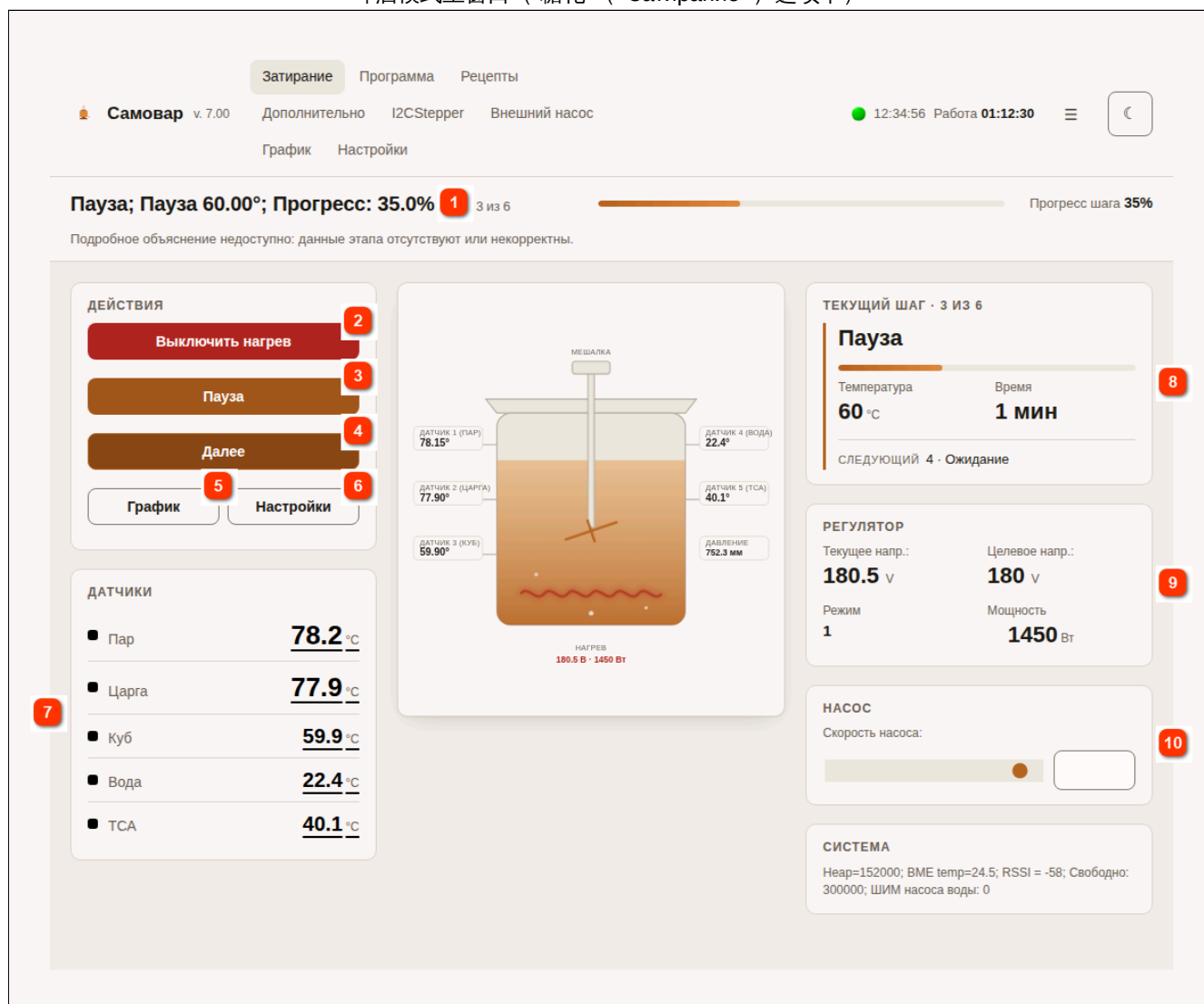
帮助中说暂停会清除塔节和蒸汽温度值。每次暂停后都要设定新值吗？

不需要。取酒程序的“暂停”行执行完毕后，Samovar 会自己设定新的酒心取酒温度。（蒸汽温度和塔节温度）

“暂停”（«Поставить на паузу»）命令只停止取酒泵和正在执行的程序行的计时器，记录的温度值不会被清除。执行“继续”（«Продолжить»）命令后，大约一分钟内杂质检测器不干预取酒。

啤酒模式

啤酒模式主窗口（“糖化”（«Затирание»）选项卡）



啤酒模式主窗口，“糖化”（«Затирание»）选项卡

- 1 **状态行（«Строка состояния»）**
当前程序行的类型和状态，步骤编号“N of M”（«N из M»）。下方是以百分比表示的步骤进度。
- 2 **打开加热（«Включить нагрев»）**
开始或停止执行糖化程序。
- 3 **暂停 / 继续（«Пауза / Продолжить»）**
手动暂停酿造：关闭加热、冷却、搅拌器和水泵，直到按下“继续”（«Продолжить»）。
- 4 **下一步（«Далее»）**
转到糖化程序的下一行。
- 5 **图表（«График»）**
打开本地图表。
- 6 **设置（«Настройки»）**
转到设置页面。
- 7 **传感器（«Датчики»）**
所有已分配传感器的读数。只有程序行中选定的传感器控制加热。
- 8 **当前步骤（«Текущий шаг»）**
行类型、温度、时间和下一程序行。
- 9 **调压器（«Регулятор»）**
当前电压和目标电压（功率）、调压器模式。啤酒模式下不使用手动输入电压。

警告！重要！

Samovar 酿造啤酒时的工作方式与精馏模式有很大不同。

Samovar 的啤酒模式在 **3 个地方** 设置：

1. 在 Samovar_ini.h 文件中的以下几行：

[#define BOILING_TEMP 98.9](#) - 您设备上的沸点。在“煮沸” («Кипячение») 行上，Samovar 将蒸馏釜加热到 $BOILING_TEMP + 5^{\circ}$ ，并在麦汁过热时用同一数值关闭加热。沸腾开始本身是根据蒸馏釜温度趋于稳定自动检测的。
[#define ACCELERATION_HEATER_DELTA 4](#) - 距设定温度还差多少度时关闭辅助加热管。
[#define HEAT_DELTA 1](#) - 距设定温度还差多少度时主加热器从升温切换为 PID 控制。

2. 在设置的 [啤酒选项卡](#) («Пиво») 上：酿造布局 (一体式 / HERMS / RIMS)、PID 调节器系数、保温模式以及加热和沸腾模式下的调压器电压 (功率)，以及沸腾时是否使用辅助加热管。

3. 在啤酒模式开启时的“程序” («Программа») 和“附加” («Дополнительно») 选项卡上。

注。

如果在“设置” («Настройки») → “温度” («Температура») 选项卡上连接并分配了温度传感器，界面会显示所有温度传感器的读数。

这是实现各种酿造布局所必需的

但是，只监控一个温度传感器，即糖化程序行中选定的那个。

这样可以把多余的传感器作为信息传感器，按自己的需要使用。例如：观察附加容器中洗糟水的温度等。

该行的温度取自糖化程序。达到温度的容差是“温度” («Температура») 选项卡上程序行所选传感器的“超出设定值” («вылет за уставку»)： 0.3° 到 3° ；低于 0.3° 的值 (包括 0) 按 0.3° 处理。

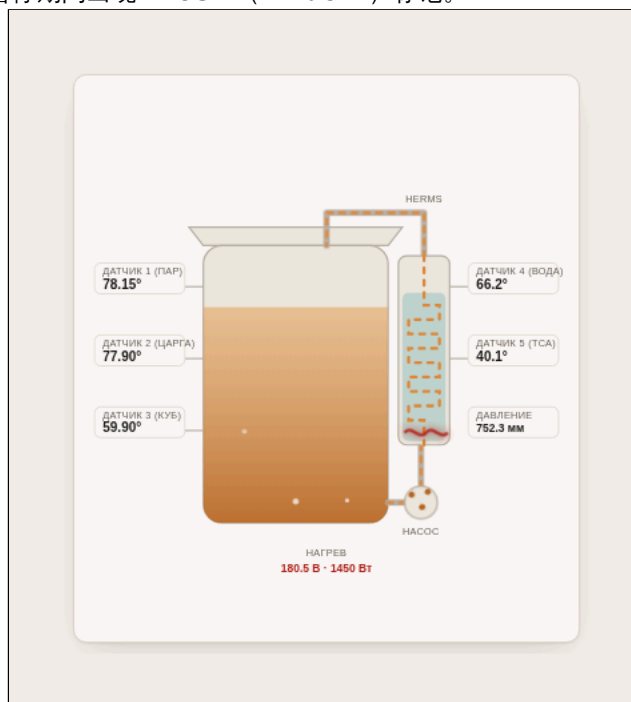
窗口中显示的内容

状态行 (顶部)：当前程序行的类型和状态，以及步骤编号“N of M” («N из M») 。下方是进度条和以百分比表示的 **步骤进度** («Прогресс шага»)：该行时间已过去的比例。只对“休止” («Пауза») 和“煮沸” («Кипячение») 行计算，并且只在达到温度之后 (煮沸行在检测到沸腾之后) 才计算。升温期间进度为 0%。

主要数值：蒸馏釜温度、当前步骤的设定值和时间、搅拌器状态。

传感器：所有已分配传感器 (蒸汽、塔节、蒸馏釜、水、TCA) 的读数。

示意图：图形取决于啤酒 («Пиво») 选项卡上选择的酿造布局。一体式 - 带搅拌器和加热管的锅；HERMS - 糖化锅、循环泵和带盘管的热水罐；RIMS - 糖化锅、水泵和带加热管的流通管。当前程序行的设备工作时，搅拌器或水泵显示动画；加热时加热管高亮显示；暂停期间出现“PAUSE” («ПАУЗА») 标记。



HERMS 酿造布局的工艺示意图：循环泵和带盘管的热水罐

当前步骤：行类型、温度、时间和下一程序行。

调压器：当前电压和目标电压 (功率)、调压器模式和功率。啤酒模式下不能手动输入电压 - Samovar 按自己的算法达到并保持设定温度 (见[加热器工作算法](#))。

水泵：麦汁输送泵（循环泵）的转速滑块，见[附加选项卡](#)。

状态行

示例：

麦芽已投入，按下了“下一步”（«Далее»）。

Samovar 会发送消息“转到程序行 №2；温度休止：68.00°，时间：60.00 min”（«Переход к строке программы №2; Температурная пауза: 68.00°, время: 60.00 мин»），并在状态行显示：“休止；休止 68.00°；升温中；当前 T：61.50°”（«Пауза; Пауза 68.00°; Разогрев; Текущая T: 61.50°»）。

也就是说，正在升温，程序时间不计。

温度达到 68°（准确地说 68° 减去容差，默认 0.3°）时，会收到消息“已达到温度休止 68.00°。等待 60.00 分钟。”

（«Достигнута температурная пауза 68.00°. Ждем 60.00 минут.»），状态行变为“休止；休止 68.00°；进度：1.7%”（«Пауза; Пауза 68.00°; Прогресс: 1.7%»）。“当前步骤”（«Текущий шаг»）卡片中的步骤时间显示的是程序规定的该行时长；各次休止之间的升温时间不计入其中。

其他状态：“升温至下麦芽温度”（«Разогрев до температуры засыпи солода»）、“等待下麦芽”（«Ожидание засыпи солода»）、“煮沸 - 加热”（«Кипячение - нагрев»）、“煮沸 60.00 min”（«Кипячение 60.00 мин»）、“冷却至 30.00°”（«Охлаждение до 30.00°»）、“等待。请按‘下一程序’”（«Ожидание. Нажмите 'Следующая программа'»）、“发酵；保持 T=26.00°”（«Ферментация; Поддерж. T=26.00°»）。如果酿造已暂停，状态行以“已暂停”（«На паузе»）开头。

程序行的计时在达到设定温度之前稍早一点就开始。详见[糖化程序](#)一节。

按钮

打开加热（«Включить нагрев»）/**关闭加热**（«Выключить нагрев»）：开始或停止执行已设置的程序。开始前，Samovar 会检查所有程序行；如果某行有错误，则取消启动，并给出带行号的消息。停止时收到消息“糖化程序已完成”（«Программа затираания завершена»）。

暂停（«Пауза»）/**继续**（«Продолжить»）：手动暂停酿造。关闭加热、冷却、搅拌器和水泵，不让程序行继续推进。暂停时间不计入该行的保温时间；按“继续”后，搅拌器的时间表从中断处接着执行。消息：“糖化已暂停。”（«Затирание поставлено на паузу.»）和“糖化已继续。”（«Затирание продолжено.»）。在“自动整定”（«Автотюнинг»）和“Lua 脚本”（«Lua Скрипт»）行上，暂停会推迟到最近的酿造行。

下一步（«Далее»）：转到糖化程序的下一行（需确认）。如果在“冷却”（«Охлаждение»）行上麦汁温度仍高于目标值时按“下一步”，Samovar 会要求在 10 秒内再按一次。

图表（«График»）：查看本地图表

设置（«Настройки»）：转到[设置](#)选项卡。

窗口顶部的选项卡：“糖化”（«Затирание»）（本窗口）、[“程序”（«Программа»）](#)、[“附加”（«Дополнительно»）](#)。如果连接了 [I2CStepper](#) 扩展板，会出现“I2CStepper”选项卡；如果有外部 I2C 水泵，则出现“外部水泵”（«Внешний насос»）选项卡。

糖化程序

用于啤酒麦芽糖化、谷物发酵醪糖化以及低温慢煮

程序保存在 Samovar 的内存中，重启后不会丢失。如果没有用户程序，则加载出厂示例。*

默认加载的程序如下：

Программа затирки:

Варочный порядок из настроек: All-in-one / BIAB. ПИД по кубу (затор), мешалка как в универсальной программе.

1	Тип программы	2	Темп	3	Время	Мешалка	4	Датчик темп.	5
01	Темп. закладки солода	45	0	0^0^0^0^0	Куб (затор)				+ -
02	Пауза	45	1	0^0^0^0^0	Куб (затор)				+ -
03	Пауза	60	1	0^0^0^0^0	Куб (затор)				+ -
04	Ожидание	0	0	0^0^0^0^0	Куб (затор)				+ -
05	Кипячение	0	1	0^0^0^0^0	Куб (затор)				+ -
06	Охлаждение	30	0	0^0^0^0^0	Куб (затор)				+ -

► Описание программы затирки:

► Описание программы затирки в текстовом формате для сохранения в облаке:

Сохранить программу 6

Загрузить программу 7

Установить программу 8

Очистить программу 9

Загрузить рецепт из XML 10

"程序" («Программа») 选项卡：出厂糖化程序

1 程序类型 («Тип программы»)
选择要执行的程序：休止 («Пауза») 、下麦芽温度 («Темп. закладки солода») 、煮沸 («Кипячение») 、冷却 («Охлаждение») 、等待 («Ожидание») 、发酵 («Брожение») 、自动整定 («Автотюнинг») 、Lua 脚本 («Lua Скрипт») 。

2 温度 («Темп.»)
该行的温度，单位 °C。用于休止、冷却、下麦芽温度、发酵和自动整定；煮沸和等待为 0。

3 时间 («Время»)
时间，单位分钟。只用于休止和煮沸；其他类型为 0。

4 搅拌器 («Мешалка»)
搅拌器和循环泵的工作模式：设备^搅拌器转速^水泵转速^运行时间，s^暂停，s。点击该字段会打开编辑窗口。

5 温度传感器 («Датчик темп.»)
在该行中控制加热或冷却的传感器：蒸馏釜 («Куб») 、水 («Вода») 、塔节 («Цагра») 、蒸汽 («Пар») 或 TCA («TCA») 。

6 保存程序 («Сохранить программу»)
将程序及其说明保存到 programbackup.txt 文件。

7 加载程序 («Загрузить программу»)
从文件把程序加载到编辑窗口（按“设置程序” («Установить программу») 后才写入 Samovar 内存）。

8 设置程序 («Установить программу»)
将程序写入 Samovar 内存。

9 清除程序 («Очистить программу»)
从 Samovar 内存中删除所有程序行。

从 XML 加载配方 («Загрузить рецепт из XML»)
转到 BeerXML 配方导入页面。

表格上方显示提示“设置中的酿造工序：...”（«Варочный порядок из настроек: ...»），即在设置的[啤酒选项卡](#)（«Пиво»）中选择的酿造设备布局。它不影响程序的执行，只用于导入 BeerXML 配方。

各列：

№ - 程序行号。总共最多可设置 20 行。

程序类型（«Тип программы»）- 选择要执行的程序

温度（«Темп.»）- 温度设定，单位 C°

时间（«Время»）- 时间，单位分钟。

搅拌器（«Мешалка»）- 用于设置[搅拌器和循环泵](#)工作模式的字段。点击该字段会打开一个窗口，在其中填写。

温度传感器（«Датчик темп.»）- 选择用哪个温度传感器来控制加热或冷却（用于实现不同的酿造工序）。可以从 5 个传感器中选择一个：“蒸馏釜（糖化醪）”（«Куб (затор)»）、“水（HERMS 热水罐）”（«Вода (бойлер HERMS)»）、“塔节（RIMS 回流）”（«Царга (обратка RIMS)»）、“蒸汽”（«Пар»）、“TCA”（«TCA»）。

行末的“+”和“-”按钮分别在当前行之后添加一行和删除当前行。

程序下方有一个可折叠的“糖化程序说明”（«Описание программы затирки»）字段，其中以文本格式显示程序。

Программа затирки:

Варочный порядок из настроек: All-in-one / BIAB. ПИД по кубу (затор), мешалка как в универсальной программе.

№	Тип программы	Темп	Время	Мешалка	Датчик темп.	
01	Темп. закладки солода	45	0	0^0^0^0^0	Куб (затор)	+ -
02	Пауза	45	1	0^0^0^0^0	Куб (затор)	+ -
03	Пауза	60	1	0^0^0^0^0	Куб (затор)	+ -
04	Ожидание	0	0	0^0^0^0^0	Куб (затор)	+ -
05	Кипячение	0	1	0^0^0^0^0	Куб (затор)	+ -
06	Охлаждение	30	0	0^0^0^0^0	Куб (затор)	+ -

▼ Описание программы затирки:

Текст программы затирки:

```
M;45;0;0^0^0^0^0;0
P;45;1;0^0^0^0^0;0
P;60;1;0^0^0^0^0;0
W;0;0;0^0^0^0^0;0
B;0;1;0^0^0^0^0;0
C;30;0;0^0^0^0^0;0
```

1

Параметры программы затирания: Тип программы;Температура;Время;Устройства;Датчик

M - malt application temp (температура закладки солода), P - pause (пауза при заданной температуре),

B - boil (кипячение и засыпка хмеля в конце), C - cool (охлаждение до заданной температуры),

W - wait (режим ожидания, ничего не делает до смены строки программы)

F - ferment (ферментирование; время 0 - держать температуру бесконечно, время в минутах - перейти к следующей строке по его истечении; ручная пауза выключает и нагрев, и охлаждение до продолжения).

L - Lua скрипт: все параметры строки должны быть нулевыми. Вход ждёт подтверждённый запуск Lua-job, выход ждёт его остановку; ошибка Lua останавливает варку.

Варочный порядок (All-in-one / HERMS / RIMS) задаётся в Настройках, вкладка «Пиво». Импорт BeerXML подставляет датчик и насос из этого порядка; TYPE шага (Infusion/Temperature/Decoction) берётся из рецепта.

Пример программы отбора:

```
M;45.00;0;1^20^0^1^1;0
```

```
P;60.00;1;1^20^0^1^1;0
```

```
W;0.00;0;1^20^0^1^1;0
```

```
B;0.00;1;1^20^0^1^1;0
```

```
C;30.00;0;1^20^0^1^1;0
```

Описание устройств: предпоследнее поле имеет вид 3^100^1200^30^10, где

3 — мешалка и насос; -100 — 100 об/мин со сменой направления после паузы; 1200 — скорость шагового I2C-насоса в мл/ч; 30 — работа, сек; 10 — пауза, сек.

Первое число — устройство: 1 — мешалка, 2 — насос, 3 — оба устройства.

Второе число: 0 — мешалка на реле 1 I2CStepper, другое значение — обороты шаговой мешалки. Отрицательное значение включает смену направления после каждой паузы.

Третье число: 0 — насос на реле 1 I2CStepper, больше 0 — скорость шагового I2C-насоса в мл/ч. Мешалку и насос нельзя одновременно назначить на реле 1. Четвёртое и пятое — время работы и паузы в секундах.

▼ Описание программы затирки в текстовом формате для сохранения в облаке:

Описание программы затирки:

Венское. 25 л. Затирание 64/67/72/78, кипячение 60 мин.

2

Сохранить программу

Загрузить программу

Установить программу

Очистить программу

Загрузить рецепт из XML

1

糖化程序文本 («Текст программы затирки»)

同一程序的文本格式：Type;Temperature;Time;Stirrer;Sensor (类型;温度;时间;搅拌器;传感器)。可以直接在这里编辑各行，表格会随之更新。

2

糖化程序说明 («Описание программы затирки»)

自由格式的说明 (最多 250 字节)。它与程序一起保存到文件中，并在酿造会话开始时发送到云端。

行格式：**Type;Temperature;Time;Stirrer;Sensor** (类型;温度;时间;搅拌器;传感器)，例如“P;60.00;1;1^30^0^2^3;0”。
搅拌器用五个以“^”分隔的数字设置 (见[搅拌器](#)、[水泵](#)、[温度](#))，传感器用 0 (蒸馏釜) 到 4 (TCA) 的数字设置。

行类型说明：

M (下麦芽温度) – “下麦芽温度” («Темп. закладки солода»)：加热到设定温度并保持，直到操作者切换到下一程序行。达到设定温度时，会收到需要投入麦芽的消息。该行的时间必须为 0。

P (休止) – “休止” («Пауза»)：加热到指定温度并保持设定的时间。时间到后，Samovar 切换到下一程序行。温度和时间都必须大于 0。

W (等待) – “等待” («Ожидание»)：什么都不做，等待操作者切换到下一程序行。加热和冷却关闭；搅拌器和水泵按该行的设定工作。温度和时间均为 0。

B (煮沸) – “煮沸” («Кипячение»)：按设定时间煮沸。时间结束前 30 秒，Samovar 发送需要投入啤酒花的消息。也就是说，如果需要多次投入啤酒花，就要设置多个煮沸行。也可以只用一行，但投啤酒花的时间就得自己掌握。温度必须为 0，时间必须大于 0。

C (冷却) – “冷却” («Охлаждение»)：冷却到设定温度。冷却水供水阀门打开，冷却泵启动 (如果使用)。时间必须为 0。

F (发酵) – “发酵” («Брожение»)：通过加热和冷却保持设定温度。时间为 0 - 无限期保持，直到手动切换行；时间大于 0 (单位分钟，最多 43200) - 时间到后转到下一行，见[发酵程序](#)。

A – “自动整定” («Автотюнинг»)：在设定温度下自动整定加热 PID 调节器的系数。完成后，Kp、Ki、Kd 系数保存到设置中 (啤酒选项卡)，Samovar 转到下一行。时间为 0，搅拌器关闭。

L – “Lua” («Lua»)：该阶段由 Samovar 上某个文件中的 Lua 脚本控制 (仅限支持 Lua 的固件，见[关于 Lua 脚本语言](#))。选择 L 类型时，会打开“Lua 阶段” («Lua-этап») 窗口：在其中选择脚本文件 (Samovar 内存中的任意 .lua 文件)，设置阶段超时时间 (秒，1–65535)，如有需要，还可设置脚本在 arg 表中收到的参数。这样的行不设置温度、设备和传感器。脚本必须调用 setNextProgram()；如果在超时前没来得及调用或因错误结束，酿造将停止。

程序行执行完后，Samovar 关闭加热并结束程序，显示消息“糖化程序已完成” («Программа затириания завершена»)。

每次转到新行时，都会收到带有该行参数的消息“转到程序行 №N” («Переход к строке программы №N»)。如果在设置 (主要选项卡) 中启用了“程序切换时打开蜂鸣器” («Включать зуммер при смене программы») ，蜂鸣器也会响。

下一个可折叠字段是自由文本格式的糖化程序说明 (最多 250 字节)。它与程序一起保存在备份文件中，并在酿造会话开始时发送到云端，例如：

“维也纳。25 L。糖化 64/67/72/78，煮沸 60 min，啤酒花 33 g，分别在 60、30 和 5 min 投入。”

以默认加载的初始程序为例进行说明。

注意。有些行类型忽略时间，有些忽略温度。填写有误的程序不会被接受：尝试设置时，会出现消息“程序错误。”

(«Ошибка программы.»)，并给出出错行的行号和原因 (例如“第 1 行：“休止”- 需要大于 0 的温度和时间” («Строка 1: «Пауза» - нужны температура и время больше 0»))，而在酿造开始时会出现带有出错行号的消息。

第 01 行。下麦芽温度。温度 45°。时间 0

向糖化锅中注水并打开加热。如果使用辅助加热管，它也会打开。

温度达到 41°C 时，辅助加热管关闭 (见[加热器工作算法](#))。

在“升温” («Разгон») 模式下，Samovar 会将蒸馏釜一直加热到 44°C。

随后“升温”模式关闭，加热由 PID 调节器控制，以便平稳地接近设定温度。

达到 45° 温度后 (准确地说是 44.7°，容差 0.3°，容差详见下文)，Samovar 显示消息“已达到下麦芽温度！”

(«Достигнута температура засыпи солода!»)，状态行出现“等待下麦芽” («Ожидание засыпи солода»)。

Samovar 将保持该温度并无限期等待。

投入麦芽后，糖化锅中的温度开始下降，Samovar 会加大加热以保持温度。

投入麦芽后，在[“啤酒” \(«Пиво»\)](#) 模式主窗口中按“下一步” («Далее») 。开始执行第 2 行

第 02 行。休止。温度：45°。时间 1

在 45° 保持 1 分钟，然后自动转到第 3 行。将发送消息“转到程序行 №3；温度休止：60.00°，时间：1.00 min”
(«Переход к строке программы №3; Температурная пауза: 60.00°, время: 1.00 мин»)

第 03 行。休止。温度：60°。时间 1

为了达到该温度，Samovar 向调压器发出“升温” («Разгон») 指令。在距目标温度还差几度时，辅助加热管关闭，主加热管继续以“升温”模式工作，之后由 PID 调节器控制，即加热器的工作原理与第 1 行相同

注意！

根据蒸馏釜容量的不同，加热最后零点几度可能需要很长时间。因此，当温度与设定值只差一个容差时，就开始计算该行的时间。容差就是设置-温度选项卡上该行传感器的“超出设定值” («Вылет за уставку») 字段：0.3° 到 3°；低于 0.3° 的值（包括 0）按 0.3° 计。新值在保存设置后立即生效，酿造过程中也是如此。对于第 3 行，容差为 0.3° 时，从 59.7° 开始计时，并收到消息“已达到温度休止 60.00°。等待 1.00 分钟。” («Достигнута температурная пауза 60.00°. Ждем 1.00 минут.»)。

如果休止期间温度降到容差以下（例如补水后），时间不会全额计算：温度差得越多，计时越慢。容差为 0.3°、温度为 58.35°（差 1.65°）时，一分钟只按半分钟计。差 3° 或更多时，计时停止，并收到消息“保温暂停：温度低于目标”

(«Пауза выдержки: температура ниже цели»); 恢复后收到“保温继续” («Выдержка продолжена») 。在此期间加热一直工作。温度超过设定值不会减慢计时。手动暂停（“暂停” («Пауза») 按钮）的时间也不计入保温时间。

执行时间结束后，Samovar 自动转到下一程序行

第 04 行。等待。温度 0.00。时间 0。

什么都不做，等待操作者的指令。不计时间。加热也不工作。如果该行设置了搅拌器和水泵，它们会工作。

第 05 行。煮沸。温度 0.00。时间 1。

按设定时间煮沸。该行不填温度，只计时间；不允许时间为零。Samovar 以 BOILING_TEMP + 5° 的设定值加热蒸馏釜（达到温度的算法与第 1 行相同；状态行显示“煮沸 - 加热” («Кипячение - нагрев»))。

Samovar 会自行判断沸腾开始：当该行传感器的温度不低于 98° 且约一分钟不再上升时。此时收到消息“已开始煮沸模式” («Начался режим кипячения») ，开始计时，加热切换为啤酒选项卡中“加热和沸腾模式下调压器的电压/功率”

(«Напр./мощность регулятора в режиме нагрева и кипения») 的值；只有在该处启用了“沸腾时使用辅助加热管”

(«Использовать разгонный тэн при кипении») 时，辅助加热管才在沸腾期间工作。如果 120 分钟内未检测到沸腾，酿造将停止，并提示检查蒸馏釜传感器、液体量和锅盖。

该行时间结束前 30 秒，Samovar 发送消息“投入啤酒花！” («Засыпьте хмель!») ，打开蜂鸣器并让步进电机走一步

(用于啤酒花投料器，见[添加投料器](#)) 。如果需要多次投入啤酒花，就要设置多个煮沸行：转到下一个“煮沸”

(«Кипячение») 行时，不再重新判断沸腾，立即开始计时。也可以只用一行，但投啤酒花的时间就得自己掌握。

您设备的沸点在 Samovar_ini.h 文件中手动设置，即这一行 `#define BOILING_TEMP 98.9` 。如果蒸馏釜温度超过 BOILING_TEMP + 5°，Samovar 会紧急关闭加热。

第 06 行。冷却。温度 30。时间 0

冷却到设定温度。不计时间。加热关闭，冷却水供水阀门打开，冷却泵以全功率启动（如果固件中启用了水泵控制）。当该行传感器的温度降到设定值时，阀门关闭，水泵停止，Samovar 转到下一行。如果 120 分钟内蒸馏釜未冷却下来，酿造将停止，并提示检查传感器、供水和阀门。

麦汁温度仍高于目标值时，若要用“下一步” («Далее») 按钮跳过冷却，需要在 10 秒内再按一次；Samovar 会用消息提醒这一点。

命令按钮位于窗口最下方：

保存程序 («Сохранить программу») - 糖化程序连同说明保存到浏览器“下载”文件夹中的 programbackup.txt 文件。

加载程序 («Загрузить программу») - 从文件加载糖化程序。既可以用“保存程序”按钮保存的文件，也可以用包含程序行的普通文本文件。

设置程序 («Установить программу») - 将程序写入 Samovar 内存以便执行。会出现消息“程序已保存。” («Программа сохранена.») 。

清除程序 («Очистить программу») - 从 Samovar 内存中删除所有程序行（需确认）。

从 XML 加载配方 («Загрузить рецепт из XML») - 转到[从 BeerXML 文件加载配方](#)

还有第二种获取配方的方法，不需要把文件保存到电脑：按“从 XML 加载配方”；在“啤酒” («Пиво») 模式下，打开的页面顶部会显示“网站上的啤酒配方” («Рецепты пива с сайта») 卡片，即 samovar-tool.ru 的配方目录，可以直接在浏览器中选择配方。详见[从 BeerXML 文件加载配方](#)页面。

运行期间，与其他模式一样，运行日志保存到内部存储和云端，并绘制图表。

*** 如何让啤酒程序默认加载您的配方？**

出厂示例程序在 `sensorinit.h` 文件的 `prepare_default_program_for_mode()` 函数中设置。不要依赖行号：不同版本中行号会变化。

即将酿造的程序请通过网页界面设置并保存。只有需要另一个出厂示例（没有用户程序时创建）时，才修改源代码。

修改出厂示例后，必须重新编译并上传固件。每次更新 Samovar 版本后，都要重新做一遍这项修改。

搅拌器、泵、温度。

搅拌器和 麦汁输送泵（循环泵）

注意。

Samovar 7.00: 在啤酒 («Пиво») 模式下，搅拌器和麦汁输送泵既可以由 Samovar 自身的输出端驱动，也可以通过扩展板 [I2CStepper](#) 驱动。如果程序行中设定了步进搅拌器或步进泵，但缺少所需的板子，加热将不会启动：程序启动时会显示说明原因的消息 («第 N 行 I2C 搅拌器不可用» («I2C-мешалка недоступна в строке N»))、«第 N 行 I2C 泵不可用» («I2C-насос недоступен в строке N»)) 或«第 N 行 I2C 泵继电器不可用» («I2C-реле насоса недоступно в строке N»)) 。

可以采用以下连接方式：

继电器搅拌器——接通 Samovar 的 2 号继电器（转速为 0），不需要扩展板；

步进电机搅拌器——使用奇数地址的 I2CStepper 板。设定转速和旋转方向；2 号继电器同样接通；

步进电机泵——使用偶数地址的 I2CStepper 板。设定流量，单位 ml/h；

无步进电机的泵（流量为 0）——接通 Samovar 的冷却泵输出；如果连接了 I2CStepper，板上的 1 号继电器也同时接通。只有固件在编译时启用了 PWM 泵 (USE_WATER_PUMP, 默认启用)，才有冷却泵输出；否则泵只能由 I2CStepper 的 1 号继电器驱动。

泵所用的 1 号继电器优先取自搅拌器板（如果该板有继电器），否则取自泵板。

点击搅拌器参数输入框会打开一个附加的选择菜单。

“编辑设备模式” («Редактирование режима устройства») 窗口（点击程序行的“搅拌器” («Мешалка») 字段）

- 1 **设备 («Устройство»)**
关闭、搅拌器、泵、搅拌器和泵。
- 2 **搅拌器转速 («Скорость мешалки»)**
0 表示使用 Samovar 2 号继电器上的搅拌器，否则为 I2CStepper 步进搅拌器的转速，单位转/分；负值表示每次暂停后改变方向。
- 3 **运行，秒 («Работа, сек»)**
运行时间，最大 65535 秒。
- 4 **暂停，秒 («Пауза, сек»)**
暂停时间；0 表示不暂停连续工作。
- 5 **确定 («Ок»)**
将数值写入程序行字段。

设备 («Устройство») ——选择在执行该程序行时具体运行什么：搅拌器、泵，或两者同时运行。也可以把两者都关闭 («关闭» («Выключено»)) 。

搅拌器：0 — ESP32 2 号继电器，否则为 I2C，转/分 («Мешалка: 0 — реле 2 ESP32, иначе I2C, об/мин») ——0 表示使用 2 号继电器上的搅拌器，否则为步进搅拌器的转速。正值为正向旋转；为负值时，搅拌器每次暂停后改变旋转方向。只有连接了步进搅拌器板才显示此字段；否则该字段被替换为“搅拌器：ESP32 2 号继电器” («Мешалка: реле 2 ESP32») 。

泵 («Насос») ——0 表示无步进电机的泵（冷却泵输出和/或 1 号继电器），大于 0 表示步进泵的流量，单位 ml/h。只

有连接了步进泵板才显示此字段；否则显示文字“泵：I2CStepper 1 号继电器” («Насос: реле 1 I2CStepper»)（如果连接了步进搅拌器板）或“泵：ESP32 输出” («Насос: выход ESP32»)。

运行，秒 («Работа, сек»)——运行时间。（*注意！从 6.22 版起时间以秒为单位设定*）

暂停，秒 («Пауза, сек»)——暂停时间。每个字段的最大值为 65535 秒（18 小时 12 分）。

点击“确定” («Ок») 时会检查数值：如果为未选中的设备设定了非零值，会显示消息“对已关闭的设备请填写 0。” («Для выключенного устройства укажите 0.»)。

搅拌器和泵的运行方式也可以在糖化程序的文本框中手动设定（见[糖化程序](#)）。该字段由五个数字组成，(^) 符号为数值分隔符：

第一个——设备类型：0 为关闭，1 为搅拌器，2 为泵，3 为搅拌器和泵同时运行；

第二个——搅拌器转速（-32768 到 32767）：0 表示使用 Samovar 2 号继电器上的搅拌器，正值为正向旋转，负值表示每次暂停后改变旋转方向。未选择搅拌器时为 0；

第三个——步进泵流量，单位 ml/h（0 到 65535）：0 表示无步进电机的泵（Samovar 冷却泵输出和/或 I2CStepper 1 号继电器）。未选择泵时为 0；

第四个——运行时间，单位秒（*注意！从 6.22 版起时间以秒为单位设定*）；

第五个——停止（暂停）时间，单位秒。

示例：

1^30^0^2^3——只有步进搅拌器，30 转/分：旋转 2 秒，暂停 3 秒，暂停后改变旋转方向。

1^0^0^20^40——只有 2 号继电器上的搅拌器：工作 20 秒，暂停 40 秒。

3^30^500^60^0——步进搅拌器 30 转/分，步进泵 500 ml/h，不暂停。

如果搅拌器/泵需要连续工作，请设定任意运行时间（最大 65535），并设 暂停 = 0，或将暂停字段留空。

手动暂停（主窗口上的“暂停” («Пауза») 按钮）会停止搅拌器和泵；点击“继续” («Продолжить») 后，运行计划从中断处继续。

对“搅拌器” («Мешалка») 字段的填写有正确性检查。Samovar 7.00: 允许的值为 0^0^0^0^0（设备关闭），或类型 1、2、3 且运行时间不为零；只有选择了搅拌器才设定转速，只有选择了泵才设定流量。不接受由四个数字组成的记录（早期版本的格式）。格式不正确的行不会被保存：点击“设置程序” («Установить программу») 时，会显示“程序错误。” («Ошибка программы.») 以及行号和原因。

搅拌器和泵工作时 Samovar 的输出

搅拌器。

搅拌器工作时，接在 ESP32 的 PIN D15 (LILYGO 板为 19 脚，ESP32-S3 为 42 脚) 上的 2 号继电器接通——无论是继电器搅拌器，还是与 I2CStepper 步进搅拌器一起工作时都是如此。

麦汁输送泵

如果固件编译时启用了 PWM 泵 (USE_WATER_PUMP)，那么按程序运行泵时，冷却泵输出会接通——无论是无步进电机的泵，还是与 I2CStepper 步进泵一起工作时都是如此。可以把 12 伏直流电机的泵接到该输出上，代替冷却泵。

有一个重要特点：运行时间和暂停时间同时作用于搅拌器和麦汁输送泵。也就是说，搅拌器暂停时，泵也不工作。

如需连接非 12 伏供电（例如 220 伏）的泵和搅拌器，请使用固态继电器等中间设备。

搅拌器和麦汁输送泵算法的说明：

如果程序中设定的设备类型为“泵” («Насос»)，搅拌器不会接通。

如果只使用泵，它会 按照程序行中规定的暂停工作。

如果只选择了搅拌器，泵不会接通。

“快速升温” («Разгон») 模式下的加热器算法。

这些设置可在以下文件中修改：Samovar_ini.h。

为了达到糖化程序中设定的温度，Samovar 向主加热器施加“加热和煮沸模式下调压器的电压/功率” («Напр./мощность регулятора в режиме нагрева и кипения») 字段中的电压（[设置 - 啤酒 \(«Настройки - Пиво»\)](#)）。如果使用了[辅助加热管](#)，它也会接通。

在距目标温度还有几度时，辅助加热管关闭。该值在以下行中设定：

[#define ACCELERATION_HEATER_DELTA 4](#) (4 为数值，单位 C°)

但主加热管继续以该电压工作，直到达到设定温度减去以下行中的数值：[#define HEAT_DELTA 1](#) (1 为数值，单位 C°)

接近设定值时，加热由 PID 调节器控制，电压受“保温模式下调压器的电压/功率” («Напр./мощность регулятора в режиме поддержания темп.») 字段限制。这种算法可以防止温度“冲过”设定值。

其他设置可以在设置的[“啤酒” \(«Пиво»\)](#) 选项卡中 进行。

说明。按 ACCELERATION_HEATER_DELTA 控制辅助加热管的功能适用于使用 UART 调压器的编译版本。在继电器版本中，只要与设定值的差大于 HEAT_DELTA，辅助加热管就与主加热管一起接通，转入 PID 时关闭。

请根据您的设备手动设定沸腾温度，在文件 Samovar_ini.h 的以下行中：[#define BOILING_TEMP 98.9](#)

Samovar 7.00: 沸腾开始不再以达到 BOILING_TEMP 来判断，而是自动检测——当蒸馏釜温度停止上升时（不低于 MIN_BOILING_TEMP 98°）。此时显示消息“煮沸模式已开始”（«Начался режим кипячения»），并开始计算“煮沸”（«Кипячение»）行的时间。如果 120 分钟内未检测到沸腾，酿造将中止，并显示消息“120 分钟内未能检测到沸腾开始...”（«Не удалось зафиксировать начало кипения за 120 минут...»）。

BOILING_TEMP 的值用途如下：

- 检测到沸腾之前，“煮沸”行以 BOILING_TEMP + 5° 为设定值，按上述算法加热蒸馏釜（使用“加热和煮沸模式”的电压，差值大于 ACCELERATION_HEATER_DELTA 时使用辅助加热管）；
- 检测到沸腾之后，加热器使用“加热和煮沸模式下调压器的电压/功率”（«Напр./мощность регулятора в режиме нагрева и кипения»）字段中的电压，只有勾选了“煮沸时使用辅助加热管”（«Использовать разгонный тэн при кипении»）复选框（[设置 - 啤酒 \(«Настройки - Пиво»\)](#)），辅助加热管才会接通；
- BOILING_TEMP + 5° 是任何程序行上蒸馏釜传感器的紧急阈值：“紧急停机！超过麦汁最高温度”（«Аварийное отключение! Превышена максимальная температура сусла»）。

冷却（«Охлаждение»）

程序行：“冷却”（«Охлаждение»）。不计时间。**Samovar 7.00:** 冷却水供水阀打开，冷却泵以全功率接通。达到“冷却”行中设定的温度后，供水阀关闭，泵也关闭，Samovar 转到下一行。

由于麦汁输送泵接在冷却泵的位置上，在执行“冷却”行期间必须在该位置接上冷却泵——否则整个冷却期间泵输出端都会通电。

如果 120 分钟内蒸馏釜未冷却到设定温度，酿造将中止，并显示消息“120 分钟内未能将蒸馏釜冷却到目标温度...”（«Не удалось охладить куб до целевой температуры за 120 минут...»）。可以用“下一步”（«Далее»）按钮跳过冷却——请在 10 秒内再按一次。

洗槽水

在温度停留期间准备洗槽水。

大多数啤酒酿造配方在完成所有停留并放出麦汁后，使用约 80°C 的洗槽水冲洗麦糟。

可能出现的问题：电网能提供的最大功率。如果同时加热蒸馏釜和洗槽水容器，电网可能承受不了负载。

解决办法是：在执行糖化程序的温度停留期间加热洗槽容器中的水。

这很容易实现。

把洗槽容器加热器的控制（运行许可）接到控制辅助加热管的 4 号继电器的常闭（NC）触点上。

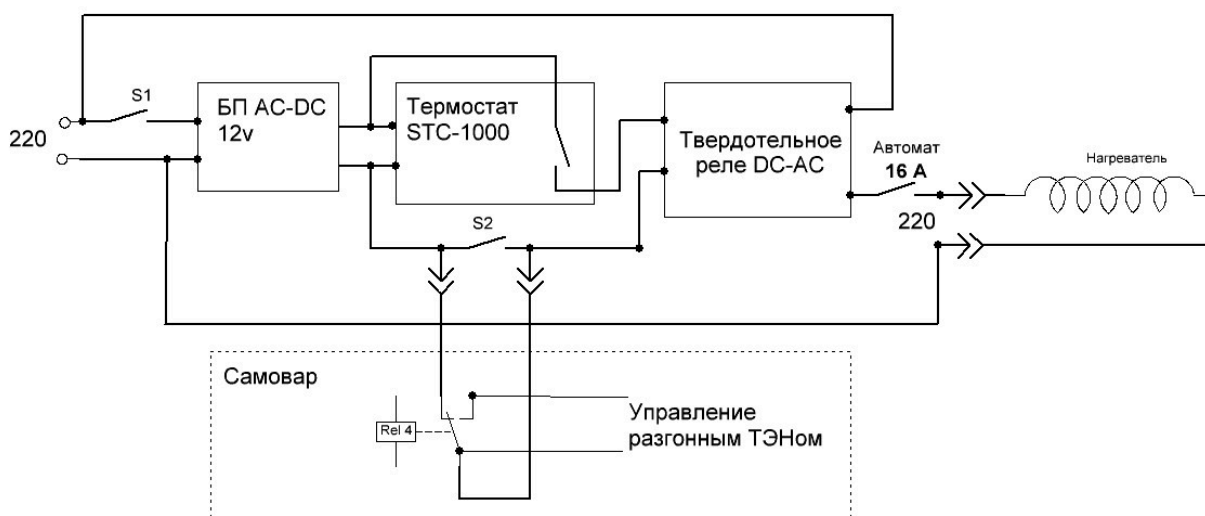
工作原理。

如果洗槽容器已加满水，即使 Samovar 没有开机，也可以打开加热，因为 4 号继电器的触点是闭合的。

当 Samovar 开始工作并发出运行糖化程序的命令后，控制蒸馏釜辅助加热管的 4 号继电器吸合，断开常闭触点，洗槽容器的加热停止。

达到设定温度后，4 号继电器释放，常闭触点闭合，洗槽容器继续加热。

对于确实使用辅助加热管的用户来说，也就是功率足够的情况，做起来很简单。4 号继电器切换触点时，要么接通蒸馏釜的辅助加热管，要么接通洗槽容器的加热器。



关于电路图的说明：

开关 S1 用于接通温控器；S2 短接接插件，使设备可以脱离 Samovar 独立工作。

16 A 断路器起保护作用，并用于开关加热器。

使用 AC-DC 220-12 伏模块和 12 伏供电的温控器 是为了提高安全性。也可以使用 220 伏供电的温控器和 AC-AC 固态继电器。

STC-1000 温控器有两种工作模式：加热和制冷，相应地有两个继电器。一个负责加热，另一个负责制冷，因此这个设备可以单独使用，不依赖 Samovar 就能保持发酵容器中的设定温度。

注意。

下文所述通过 ACCELERATION_HEATER_DELTA 参数控制 4 号继电器的方式，适用于带 UART 调压器的版本。在继电器版本（没有调压器）中，当糖化醪温度比设定值低超过 HEAT_DELTA 时，4 号继电器与主加热管一起接通，转入保温时断开。

对于不使用辅助加热管的用户。

接线图完全相同，但是.....

按照 Samovar 的逻辑，辅助加热管在达到目标温度前几度就会关闭，而按我们的接线方式，两个区间会重叠：

主加热管仍在全功率工作，而第二个加热管已经接通。

为避免这种情况，在 Samovar_ini.h 文件中，在这一行

#define ACCELERATION_HEATER_DELTA 4

把数值 4 改为 0（零），或者更好的是改为 -（负）0.3

这样得到如下工作算法：

在距设定停留温度 1°C 时（在 #define HEAT_DELTA 1 这一行中设置）

Samovar 转入 PID 控制：主加热管通过调压器获得的电压不超过“保温模式下调压器电压/功率”（«Напр./мощность регулятора в режиме поддержания темп.»）字段中的值（[“设置 - 啤酒”（«Настройки - Пиво»）](#)）。温度达到设定值时，主加热管 停止加热，4 号继电器也释放，常闭触点闭合，洗糟容器获得加热许可。如果 **#define ACCELERATION_HEATER_DELTA** 这一行设为 -0.3，只有当蒸馏釜温度高于目标 0.3°C 时继电器才断电；这样可以保证两个加热管不会同时接通，哪怕只有几秒钟。

补充。

如果温度停留的逼近几乎完美，上面所述在这一行中的设置

```
#define ACCELERATION_HEATER_DELTA
```

无论取 0 还是 -0.3，都可能无法完全起作用。中间容器里的水来不及加热，因为温度先会稍微超过目标 约 0.4，然后回落，等于目标值或比目标低 0.1-0.2°C。这时附加容器中的加热管根本不会接通。

建议采用另一种设置。

例如：

取以下数值：

保温模式下调压器电压/功率：150 伏

```
#define HEAT_DELTA 2
```

在这一行

```
#define ACCELERATION_HEATER_DELTA
```

 中填入数值 1.5

并把洗糟容器中加热管的功率限制在 1 kW

距目标温度还剩 2° 时，主加热器开始获得 150 伏电压。先是连续供电， 然后是脉冲供电。

距目标 1.5° 时，中间容器中的加热器以 1 kW 功率接通。

电网不会过载。第二个加热管有足够的 时间在温度停留期间加热洗糟水，并 保持温度

加热器的工作说明见[搅拌机、泵、温度](#)一节。

发酵程序

Samovar 可以保持发酵容器中的温度。

可以使用任意选定的传感器——推荐使用蒸馏釜温度传感器。

使用方法：在程序列表中选择“发酵”（«Брожение»）行，并设定温度。

时间可以保持为 0——这时 Samovar 会一直保持该温度，直到您手动停止程序或转到下一行。

如果设定了时间（单位为分钟，不超过 43200，即 30 天），时间到后 Samovar 会自动转到下一行。这样可以用几行“发酵”组成逐步降温的程序，例如：20° 下 4320 分钟（3 天），然后 18° 下 10080 分钟（7 天），最后一行温度 4°、时间 0——保持低温，直到您回来。“发酵”行的时间从切换到该行时开始计算，手动暂停的时间不计入。如果带时间的行是最后一行，时间结束后程序完成并断电——为避免这种情况，请将最后一行的时间保持为 0。

在文本框中，程序行以下列字符开头：

F (ferment) ——发酵，保持设定温度，直到程序行切换。

程序运行时不考虑时间，因此只能手动切换程序行或停止运行。

该行的时间必须为 0，设备（搅拌器、泵）必须关闭：0^0^0^0。否则程序不会被接受（参见[糖化程序](#)）。

Samovar 7.00: 设定值周围死区的宽度不在“温度”（«Температура»）选项卡中设置，而是在固件中固定：*BEER_TEMP_HYSTERESIS 0.3*（文件 beer.h）。发酵时冷却泵与阀门一起使用。

工作原理举例：

已知：需要保持温度 26°C，误差 +/- 0.3°（误差即该行传感器的“超出设定值”（«Вылет за уставку»），参见[糖化程序](#)）。

文本形式的程序行如下：F;26;0;0^0^0^0;

假设容器中的温度为 28°C。Samovar 关闭加热，打开水阀，并以全功率开启冷却泵。冷却一直进行，直到温度降到 26.3°。

如果温度再次升到 26.3° 以上，阀门和泵会再次开启。

在 25.7 到 26.3 的范围内系统处于空闲状态：加热关闭，阀门关闭。

当温度降到 25.7° 以下时，加热管通过 PID 控制器开启，最大电压受“保持温度模式下调压器的电压/功率”（«Напр./мощность регулятора в режиме поддержания темп.»）字段的值限制（[设置 - 啤酒](#)）。温度降得越低，加到加热管上的功率就越大。发酵时不开启辅助加热管。

主窗口上的“暂停”（«Пауза»）按钮会关闭加热、冷却、搅拌器和泵，直到您按下“继续”（«Продолжить»）。

运行过程中会记录日志，温度会发送到 Blynk，并绘制图表。

Samovar 7.00: “发酵”行也有保护功能：如果超过水温或 TCA 温度的最大值，如果蒸馏釜温度高于 BOILING_TEMP + 5°，或者没有水流（安装了流量传感器时），加热会被紧急关闭，并显示消息“紧急关闭！...”（«Аварийное отключение!...»）。

注意！

每隔 12-18 小时，加热可能会因与调压器的通信错误而关闭。这时请在“设置 - [其他](#)”选项卡中取消勾选“使用调压器连接检查”（«Использовать проверку подключения регулятора»）复选框。

说明。

Samovar 7.00：日志行的记录周期不再可调（固定为 4 秒）。“长时间发酵时增大日志记录间隔”的建议不适用于 7.00。

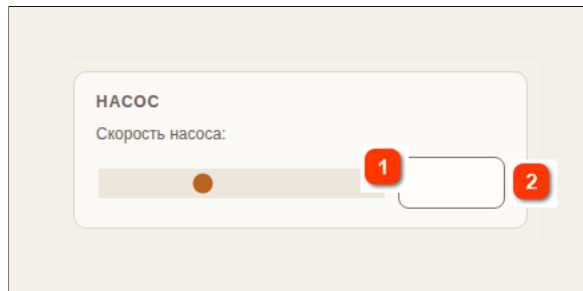
提示。

通常，接种酵母后的头 1-2 天需要密切关注糖化醪，以免发酵醪自身发热把酵母杀死，或者因强烈冷却而使发酵明显变慢。之后，当温度稳定下来（从图表上可以看出），可以只在天冷时的夜间或天热时的白天接上 Samovar，以平衡糖化醪的温度。

“附加”选项卡

Samovar 7.00: 水泵速度滑块已移到“糖化” («Затираение») 选项卡 ([啤酒](#) 模式主窗口的“水泵” («Насос») 面板)。“附加” («Дополнительно») 选项卡上只保留了 Lua 面板。

水泵速度



“糖化” («Затираение») 选项卡上的“水泵” («Насос») 面板

- 1 水泵速度**
PWM 滑块：左端为 PWM_LOW_VALUE (100)，右端为 1023 (12 V，不调制)。
- 2 输入字段**
以数字表示的同一个值，可以直接输入。400 是 PWM_START_VALUE 起始值。

这里可以控制接在冷却泵位置上的麦汁输送泵 (循环泵) 的速度，或者冷却泵本身的速度，前提是两种情况下水泵电机都使用 12 伏直流电。功率通过 PWM 调制改变。
可以用速度刻度上的滑块控制，也可以直接在字段中输入数值。
滑块在最右端时，水泵上加的是不经调制的 12 伏电压。此时输入字段的值为 1023。
因此粗略地说，10 个单位 = 1% 功率。
如果固件编译时没有启用 PWM 水泵支持 (USE_WATER_PUMP)，尝试更改速度时会显示“您的设备不支持水泵控制” («Управление насосом не поддерживается вашим оборудованием») 消息。

启动和运行条件：

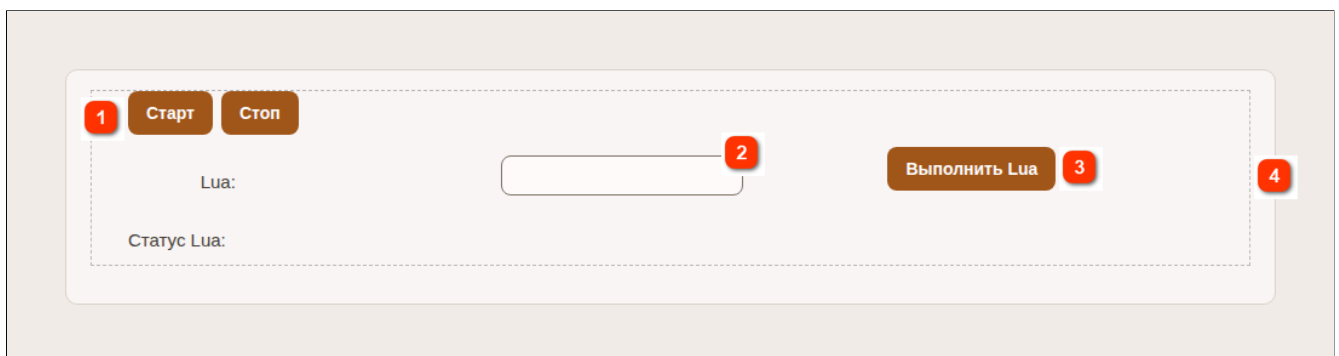
- 如果允许水泵运行，程序行开始执行时，水泵以 Samovar_ini.h 文件中这一行设置的功率启动
//水泵 PWM 调节器开始工作的值 (百分比)。请选择能保证电机启动的值
`#define PWM_START_VALUE 40`
(40% = 400 个单位)，并以此功率持续运行。如果移动滑块，Samovar 会记住该值，之后按新设定的功率工作，但仅限当前会话。Samovar 重启后，如果需要其他功率，必须重新设置。

警告！

按时间表循环运行 (程序行中的工作/暂停) 时，每次暂停后水泵都会以 PWM_START_VALUE 起始值重新启动；要恢复所需速度，必须再次移动滑块。如果此时水泵正在运行，滑块的值会立即生效。

- 滑块在最左端时，水泵不会停止，而是以 Samovar_ini.h 文件中这一行设置的最小功率运行
//水泵 PWM 调节器的下限值 (百分比)。请选择能保证水流动的值
`#define PWM_LOW_VALUE 10`
(10% = 100 个单位，即滑块的左端)。
- 在“冷却” («Охлаждение») 和“发酵” («Брожение») 行上，无论滑块位置如何，固件都以全功率 (1023) 开启冷却泵。

Lua 面板



“附加” («Дополнительно») 选项卡：Lua 面板

脚本按钮

- 1** Lua 脚本声明的按钮。如果脚本没有定义任何按钮，面板会隐藏。
- 2** **Lua**
用于输入一次性运行的一小段 Lua 代码的字段。
- 3** **执行 Lua («Выполнить Lua»)**
把字段中的代码发送去执行。
- 4** **Lua 状态 («Статус Lua»)**
脚本状态及其输出的数值。

“附加” («Дополнительно») 选项卡上有 Lua 脚本控制面板。只有当脚本定义了自己的按钮列表时，面板才会显示；列表中的按钮用于运行脚本函数。

“Lua 状态” («Статус Lua») 行显示脚本正在运行还是已停止，并在执行过程中显示信息和数值。要在啤酒模式中运行脚本，必须通过[编辑器](#)把它放进 beer.lua 文件。

可以在专用字段中输入一小段 Lua 代码，然后发出“执行 Lua” («Выполнить Lua») 命令，代码会被执行一次。

加载 BeerXML 配方

BeerXML 格式在业余和专业酿造中都有使用，便于通过互联网交换酿造数据。

BeerXML 是一种免费、定义完整的 XML 数据描述，用于交换酿造配方和其他酿造数据。

能把配方导入 Samovar，意味着您可以拿一份现成的啤酒配方，在设备合适的情况下，“点两下”就重现有经验酿酒师的成果。

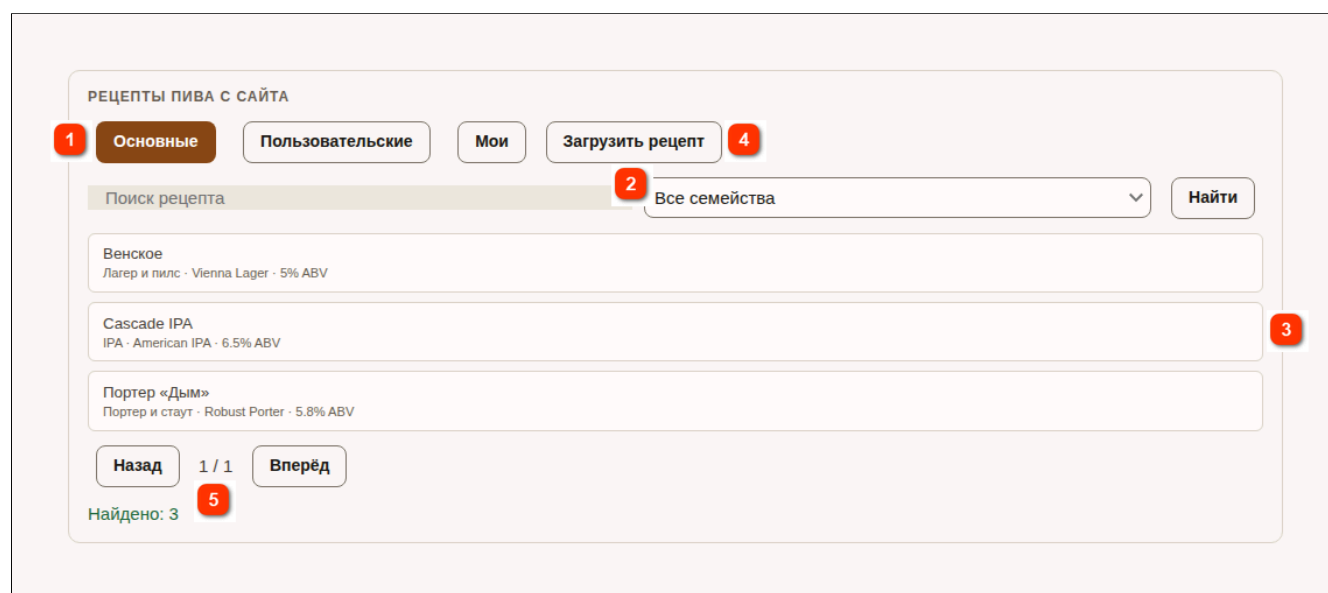
Samovar 7.00: 除 BeerXML 外，还支持 BrewMate 程序的配方文件（windows-1251 编码）。如果文件中有多个配方，会出现“文件中的配方：”（«Рецепт в файле:»）列表，供您选择需要的那个。

网站上的啤酒配方

在“啤酒”（«Пиво»）模式下，本页顶部显示“网站上的啤酒配方”（«Рецепты пива с сайта»）卡片：直接在网页界面中浏览 samovar-tool.ru 的配方目录，无需把文件保存到电脑。同一张卡片上还有用于加载电脑中文件的“加载配方”（«Загрузить рецепт»）按钮。

配方来源有三种：“主要”（«Основные»）（网站的通用目录）、“用户”（«Пользовательские»）（其他用户的配方）和“我的”（«Мои»）（您自己的配方；使用此列表需要在设备设置中填写 Blynk 令牌，否则会显示消息“设备中没有为“我的”列表设置 Blynk 令牌。”（«Для списка «Мои» в устройстве не задан токен Blynk.»））。可以按名称搜索、按啤酒类别筛选，配方列表分页显示。

在列表中选择一个配方，它会立即加载到同一页面的卡片下方，与加载文件时一样。配方名称下方显示“来源：...”（«Источник: ...»）；从电脑加载文件时这一行隐藏。



“啤酒”（«Пиво»）模式下配方加载页面顶部的“网站上的啤酒配方”（«Рецепты пива с сайта»）卡片

- 1 配方来源（«Источник рецептов»）**
“主要”（«Основные»）是网站的通用目录，“用户”（«Пользовательские»）是其他用户的配方，“我的”（«Мои»）是您自己的配方（需要在设备设置中填写 Blynk 令牌）。
- 2 搜索和筛选（«Поиск и фильтр»）**
按名称搜索，按啤酒类别筛选（IPA、“波特和世涛”（«Портер и стаут»）、“拉格和皮尔森”（«Легер и пилс»）等）。
- 3 配方列表（«Список рецептов»）**
点击配方，它会加载到同一页面的卡片下方，与加载文件时一样。
- 4 加载配方（«Загрузить рецепт»）**
从电脑加载 BeerXML 或 BrewMate 文件。
- 5 分页列表（«Постраничный список»）**
用“上一页”（«Назад»）/“下一页”（«Вперёд»）按钮翻页。

从电脑中的文件加载配方

下面以从网站 www.brewmate.ru

导入配方为例。在网站上选好感兴趣的配方后，把它保存到电脑上的某个位置，在“程序”（«Программа»）选项卡上点击“从 XML 加载配方”（«Загрузить рецепт из XML»）（或“配方”（«Рецепты»）选项卡的按钮），在打开的页面上，点击“网站上的啤酒配方”卡片中的“加载配方”。

文件管理器会打开。

进入文件夹，选择保存的文件进行加载。Samovar 会加载配方，并在卡片下方显示配方的主要数据。示例使用的是“维也纳”（«Венское»）啤酒配方。

Самовар

РЕЦЕПТЫ ПИВА С САЙТА

Основные

Пользовательские

Мои

Загрузить рецепт

1

Поиск рецепта

Все семейства

▼

Найти

Венское
Лагер и пилс - Vienna Lager - 5% ABV

Назад

1 / 1

Вперёд

Найдено: 1

Программа затирания

Описание рецепта:

Стиль: Vienna Lager

Сорт: Венское

Тип: All Grain

Пивовар: Самовар

Выход: 25 л.

Объем кипячения: 30 л.

Температура брожения: 18 °C

Время кипячения: 60 мин.

Время брожения: 14 дней

OG: 1.05

FG: 1.012

IBU: 24

ABV: 5 %

КПД: 75 %

Ингредиенты:

Тип	Название	Количество	Параметры
Солод	Венский солод	5 кг	4 °L, выход 80 %
Хмель	Saaz	30 г	Кипячение, 60 мин, α 3.5 %, пеллеты
Хмель	Saaz	20 г	Кипячение, 15 мин, α 3.5 %, пеллеты
Дрожжи	Saflager W-34/70	11.5 мл	Fermentis, сухие, атт. 82 %

Температурные паузы:

Название	Тип	Температура	Время	Вода
Мальтозная пауза	Infusion	64	40	15 л
Осахаривание	Temperature	72	20	
Мэш-аут	Temperature	78	10	

Варочный порядок из настроек: All-in-one / BIAB. ПИД по кубу (затор), мешалка как в универсальной программе.

Установить программу

На главную

Загрузка файла после загрузки рецепта (в «Венское» (Венское) рецепт)

1

Загрузка рецепта («Загрузить рецепт»)

«Веб-сайт пива» («Рецепты пива с сайта») кнопка: в браузере выбрать BeerXML или BrewMate файл.

2

Рецепт («Описание рецепта»)

Сорт, стиль, объем, время кипячения, температура брожения, OG/FG/IBU/ABV и другие данные из файла.

3

Сырье («Ингредиенты»)

Сырье, пиво (с временем), дрожжи и другие добавки.

4

Температурные паузы («Температурные паузы»)

Шаги в рецепте: название, тип, температура, время, количество.

Подсказка («Подсказки»)

- 5 设置中的酿造方式，以及 Infusion/Decoction 步骤所加“等待” («Ожидание») 行的说明。
- 6 **设置程序 («Установить программу»)**
把生成的程序写入 Samovar 的存储器。
- 7 **返回主页 («На главную»)**
返回主页，不保存程序。

注意。可以忽略“配方中没有” («Нет в рецепте») 消息，因为缺少这些数据 对糖化程序影响不大。文件中没有的说明行会隐藏；只有“品种” («Сорт»)、“发酵温度” («Температура брожения») 和“发酵时间” («Время брожения») 始终显示。

糖化程序根据配方生成：

- “下麦芽温度” («Темп. закладки солода») - 第一个糖化步骤的水温 (INFUSE_TEMP)；如果没有，取第一个步骤的温度，再没有则为 45°；
- “休止” («Пауза») - “温度休止” («Температурные паузы») 部分中的每个糖化步骤各一行。在 Infusion 步骤（第一个除外）之前和 Decoction 步骤前后，会加一行“等待” («Ожидание») ，用于补水或煮出。这些行中的搅拌器、泵和传感器按设置中的“酿造方式” («Варочный порядок») 填写 (All-in-one / BIAB、HERMS、RIMS，见[设置 - 啤酒](#))；当前方式和提示显示在休止列表下方；
- “煮沸” («Кипячение») - 配方的总煮沸时间，按啤酒花投放时间 (USE = Boil 或 First Wort) 拆分，这样每次投放前 Samovar 都会提醒“投放啤酒花！” («Засыпьте хмель!») ；
- “冷却” («Охлаждение») 和“发酵” («Брожение») ，发酵温度取自配方 (PRIMARY_TEMP) 。

程序总共不能超过 20 行。如果文件损坏或没有糖化步骤，会显示错误消息 (“配方文件已损坏或不是 BeerXML/BrewMate 格式：...” («Файл рецепта повреждён или не в формате BeerXML/BrewMate: ...»))、“配方中没有糖化步骤” («в рецепте нет шагов затирания») 。

接着点击“设置程序” («Установить программу») (会出现消息“程序已保存。” («Программа сохранена.»))，酿造程序就会出现在“程序” («Программа») 窗口中。“返回主页” («На главную») 按钮返回主页，不保存程序。

Программа затирки:

Варочный порядок из настроек: All-in-one / BIAB. ПИД по кубу (затор), мешалка как в универсальной программе.

№	Тип программы	Темп	Время	Мешалка	Датчик темп.	
01	Темп. закладки солода	70	0	1^60^0^2^	Куб (затор)	+ -
02	Пауза	64	40	1^60^0^2^	Куб (затор)	+ -
03	Пауза	72	20	1^60^0^2^	Куб (затор)	+ -
04	Пауза	78	10	1^60^0^2^	Куб (затор)	+ -
05	Кипячение	0	45	1^60^0^2^	Куб (затор)	+ -
06	Кипячение	0	15	1^60^0^2^	Куб (затор)	+ -
07	Охлаждение	18	0	0^0^0^0^0	Куб (затор)	+ -
08	Брожение	18	0	0^0^0^0^0	Куб (затор)	+ -

Описание программы затирки:

Описание программы затирки в текстовом формате для сохранения в облаке:

Сохранить программу

Загрузить программу

Установить программу

Очистить программу

Загрузить рецепт из XML

导入“维也纳” («Венское») 配方后的“程序” («Программа») 选项卡

- 1 **来自配方的程序**
下麦芽、64/72/78° 休止、煮沸 45 + 15 min (投放啤酒花)、冷却和 18° 发酵。任何一行都可以编辑。

可以按自己的想法和所用设备调整程序。糖化程序也会出现在文本框中。

Программа затирки:

Варочный порядок из настроек: All-in-one / BIAB. ПИД по кубу (затор), мешалка как в универсальной программе.

№	Тип программы	Темп	Время	Мешалка	Датчик темп.	
01	Темп. закладки солода	70	0	1^60^0^2^	Куб (затор)	+ -
02	Пауза	64	40	1^60^0^2^	Куб (затор)	+ -
03	Пауза	72	20	1^60^0^2^	Куб (затор)	+ -
04	Пауза	78	10	1^60^0^2^	Куб (затор)	+ -
05	Кипячение	0	45	1^60^0^2^	Куб (затор)	+ -
06	Кипячение	0	15	1^60^0^2^	Куб (затор)	+ -
07	Охлаждение	18	0	0^0^0^0^	Куб (затор)	+ -
08	Брожение	18	0	0^0^0^0^	Куб (затор)	+ -

▼ Описание программы затирки:

Текст программы затирки:

```
M;70;0;1^60^0^2^3;0
P;64;40;1^60^0^2^3;0
P;72;20;1^60^0^2^3;0
P;78;10;1^60^0^2^3;0
B;0;45;1^60^0^2^3;0
B;0;15;1^60^0^2^3;0
C;18;0;0^0^0^0^0;0
E;18;0;0^0^0^0^0;0
```

1

Параметры программы затирания: Тип программы;Температура;Время;Устройства;Датчик

M - malt application temp (температура закладки солода), P - pause (пауза при заданной температуре),

B - boil (кипячение и засыпка хмеля в конце), C - cool (охлаждение до заданной температуры),

W - wait (режим ожидания, ничего не делает до смены строки программы)

F - ferment (ферментирование; время 0 - держать температуру бесконечно, время в минутах - перейти к следующей строке по его истечении; ручная пауза выключает и нагрев, и охлаждение до продолжения).

L - Lua скрипт: все параметры строки должны быть нулевыми. Вход ждёт подтверждённый запуск Lua-job, выход ждёт его остановку; ошибка Lua останавливает варку.

Варочный порядок (All-in-one / HERMS / RIMS) задаётся в Настройках, вкладка «Пиво». Импорт BeerXML подставляет датчик и насос из этого порядка; TYPE шага (Infusion/Temperature/Decoction) берётся из рецепта.

Пример программы отбора:

```
M;45.00;0;1^20^0^1^1;0
P;60.00;1;1^20^0^1^1;0
W;0.00;0;1^20^0^1^1;0
B;0.00;1;1^20^0^1^1;0
C;30.00;0;1^20^0^1^1;0
```

Описание устройств: предпоследнее поле имеет вид 3^-100^1200^30^10, где

3 — мешалка и насос; -100 — 100 об/мин со сменой направления после паузы; 1200 — скорость шагового I2C-насоса в мл/ч; 30 — работа, сек; 10 — пауза, сек.

Первое число — устройство: 1 — мешалка, 2 — насос, 3 — оба устройства.

Второе число: 0 — мешалка на реле 1 I2CStepper, другое значение — обороты шаговой мешалки. Отрицательное значение включает смену направления после каждой паузы.

Третье число: 0 — насос на реле 1 I2CStepper, больше 0 — скорость шагового I2C-насоса в мл/ч. Мешалку и насос нельзя одновременно назначить на реле 1. Четвёртое и пятое — время работы и паузы в секундах.

▼ Описание программы затирки в текстовом формате для сохранения в облаке:

Описание программы затирки:

Венское. 25 л. Хмель Saaz на 60 и 15 мин.

2

Сохранить программу

4

Загрузить программу

Установить программу

3

Очистить программу

Загрузить рецепт из XML

同一程序的文本形式和程序说明

- 1 糖化程序文本**
根据配方生成的 Type;Temperature;Time;Stirrer;Sensor 行。
- 2 糖化程序说明 («Описание программы затирки»)**
程序备忘，最多 250 字节。
- 3 设置程序 («Установить программу»)**
编辑后把程序写入 Samovar 的存储器。
- 4 保存程序 («Сохранить программу»)**
把程序和说明保存到 programbackup.txt 文件。

在“糖化程序说明” («Описание программы затирки») 文本框中，可以输入程序特点的说明 作为备忘（最多 250 字节）；它会在酿造会话开始时发送到云端。

所有修改完成后，必须把程序设置到 Samovar 中。

为此，点击页面底部的“**设置程序**” («Установить программу»)，然后就可以开始糖化了。

使用“**保存程序**” («Сохранить программу») 命令，会把糖化程序及其说明保存为名为 programbackup.txt 的文本文件，放在浏览器的“下载”文件夹中。

网站和应用中的啤酒酿造记录

与其他模式一样，啤酒模式的酿造日志会上传到 samovar-tool.ru 云端，可在[云端日志和图表](#)页面查看。除了这份日志，网站和移动应用中还有单独的“酿造记录”：配方目录、带计划任务的酿造批次列表，以及每个批次的卡片（含计划和实际结果）。

啤酒配方目录

网站页面“[啤酒配方](#)”与[加载 BeerXML 配方](#)页面上的“网站上的啤酒配方”卡片是同一个目录，但筛选条件齐全：类别、风格类型、配方类型、来源、ABV（酒精度，%）、IBU（苦度）和 OG（麦汁原始比重）的范围，按名称搜索，以及按名称、酒精度、苦度、比重、点赞数或更新日期排序。

配方卡片：SRM 色标彩条（啤酒颜色——从麦秆色到像世涛那样的黑色）、OG/FG（发酵前后的比重）/ABV/IBU/SRM 目标值、谷物清单（麦芽和其他可发酵原料）、啤酒花及投放时间、酵母、糖化步骤、作者备注、来源和许可证。滑块可把配方按原始体积的 1/4 到 4× 缩放——这只改变屏幕上的显示：“下载 BeerXML”按钮给出的始终是未缩放的原始文件。

“开始记录”按钮根据该配方在个人专区的记录中创建一个条目。点赞图标可以标记您喜欢的配方。文件格式本身见网站页面“[BeerXML 规范](#)”。

Beer recipes

Search

Catalogue

Family

Style type

Recipe type

Source

ABV min

ABV max

IBU min

IBU max

Sort

Apply

Community

Porter and stout

All

All

All

Likes

3 recipes

Dry Stout

Портер и стаут · Dry Stout (Ale)

ABV 4.3% · IBU 32 · OG 1.044

Batch: 20 L

Likes: 12

By brewer_anna

Blonde Session Ale

Портер и стаут · Session Ale (Ale)

ABV 3.1% · IBU 18 · OG 1.032

Batch: 19 L

Likes: 4

By brewer_anna

Next

啤酒配方目录：按类别筛选，卡片上有 SRM 彩条（截图取自带英文标签的界面测试样板；在网站和应用的俄文版中，同样的元素以俄文标注）

啤酒酿造者专区

[“啤酒酿造者专区”](#)页面与 samovar-tool.ru 的其他页面使用同一个网站账户（相同的登录、相同的设备列表）。

页面顶部的**任务面板**显示进行中的批次需要做什么、什么时候做。批次进入“发酵”状态时，任务会根据配方自动生成；如果没有配方，则根据啤酒类别生成：

“比重检查”- 发酵开始后第 3 天和第 7 天；

“干投啤酒花”- 对每种“干投”（Dry Hop）的啤酒花，在第 5 天或配方规定的时间；

“双乙酰休止”- 10 天后，仅限拉格；

“冷冲击”- “包装”前 2 天；

“包装”- 第 14 天（拉格为第 21 天），如果配方没有规定自己的熟化时间；

“碳化检查”- 包装后 14 天；

“就绪”- 碳化检查后的第二天。

没有配方时（例如直接在设备上启动的批次），到包装的时间按类别取：艾尔 - 14 天，拉格 - 21 天（并带双乙酰休止），小麦啤酒 - 10 天，比利时啤酒 - 18 天。

批次列表分为“酿造中”、“发酵中”、“完成”三组以及归档（已取消和已合并的条目）。“新批次”按钮打开一个对话框：选择目录中的配方或只选啤酒类别、开始日期和批次体积。

点击列表中的条目会打开**批次卡片**，它包含三个选项卡：

“**酿造**”- 糖化步骤、煮沸和啤酒花投放的计划与实际结果，快速记录比重和温度测量；

“**发酵**”- 计划任务（见上文）和发酵期间的测量；

“**概要**”- OG/FG/ABV、发酵度（已发酵的糖所占百分比）、糖化效率、IBU、发酵天数、与以往批次的比较，以及品鉴和异味。

打印卡片和打印标签（名称、日期、酒精度）只能在网站上进行，应用中没有。

JournalMy recipesSettings and export

Journal

New brew

The board shows care tasks; the list below holds every brew.

1 overdue · 1 today · 0 this week

Lager batch · 9/13/2026, 11:00:00 AM · No recipe yet

Choose a recipe

Choose recipe

Choose a family

Specify family

Overdue

Session Ale batch · 9/10/2026, 11:00:00 AM · Gravity check

Mark donePostpone...

Today

IPA batch · 9/16/2026, 11:00:00 AM · Dry hop

Mark donePostpone...

This week

Search

Recipe

All recipes

Brewing

Session Ale batch

Manual · Brew day · 9/10/2026, 11:00:00 AM

Samovar log

Samovar · Brew day · 9/14/2026, 12:00:00 PM

Lager batch

Manual · Brew day · 9/13/2026, 11:00:00 AM

Fermenting

IPA batch

Manual · Fermenting · 9/5/2026, 11:00:00 AM

Done

Dry Stout batch

Manual · Done · 8/20/2026, 11:00:00 AM

Dry Stout batch (first try)

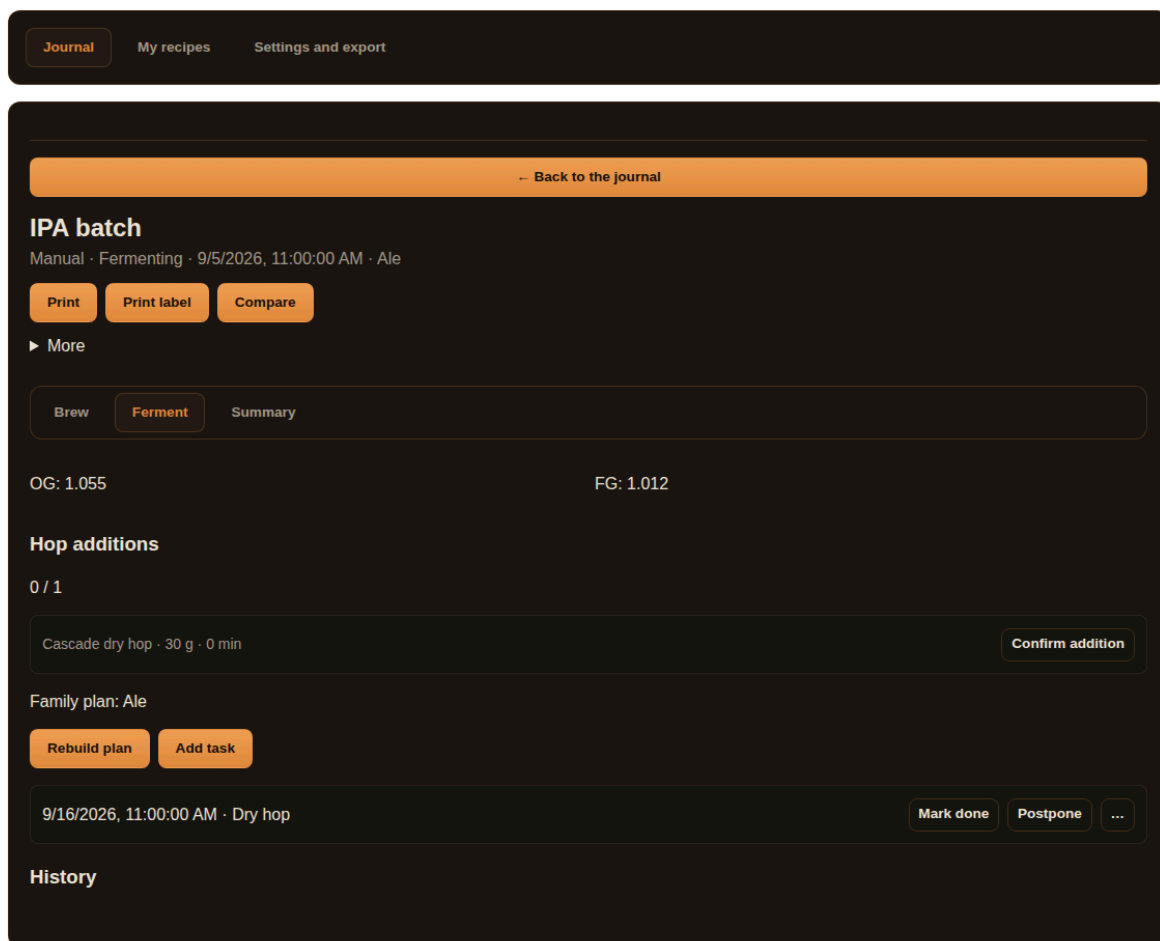
Manual · Done · 7/20/2026, 11:00:00 AM

► Archive

任务面板和按组划分的批次列表（同一测试样板的截图，英文标签）

Samovar 7.00 - 用户手册, 01.10.2026

138 / 179



批次卡片：带啤酒花和投放确认的发酵选项卡（截图中为英文标签）

自动条目。不必在网站上创建批次：如果在设备上以啤酒模式启动，记录中的条目会根据过程日志数据自动出现——也就是显示在[云端日志和图表](#)页面上的同一份数据。条目包括糖化步骤、煮沸、“投放啤酒花！”（«Засыпьте хмель!»）消息（作为啤酒花投放的确认）和冷却。设备不会为此额外发送任何内容——所有标注都由网站根据同一份日志数据完成。如果事先在网站上手动创建了批次，之后又在设备上启动，可以把手动条目和自动条目合并为一个。

应用中的啤酒

在移动应用（Android 和 iOS）中，啤酒酿造记录有单独的“酿酒”选项卡，分为两部分：“记录”（任务面板、列表和批次卡片——与网站个人专区中的相同）和“配方”（带筛选的目录和配方卡片，不能编辑）。“开始批次”对话框与网站上的相同：可搜索的配方、日期和批次体积（配方中的体积作为提示给出）。

关于任务或批次变更的推送通知，会直接在“酿酒”选项卡中打开相应批次的卡片。

与奶酪制作的区别

啤酒酿造者专区的结构与奶酪制作者专区相同，只是更简短：卡片选项卡中没有“奶酪块”（一批啤酒不会产出好几块奶酪），没有“护理”（奶酪熟成，有自己的翻面和洗皮计划），也没有单独的“库存”。任务面板、条目列表以及手动条目与自动条目的合并，在两种模式下工作方式相同。

另见：[加载 BeerXML 配方](#)、[云端日志和图表](#)。

低温慢煮

低温慢煮的原理

低温慢煮技术诞生于法国，一个非常重视烹饪艺术的国家。sous-vide 在法语中意为“真空之下”，直接点明了这种方法的原理。这项技术曾经在一段时间内属于少数人的专利：设备昂贵，并非每家餐厅都负担得起。它在乐于吸收一切新事物的分子料理中扎下了根。如今在家里也能做低温慢煮菜肴了。

低温慢煮技术分为两个阶段：

- 包装；
- 加热处理。

食材装入真空袋中，这一步使用专门的设备——真空封口机完成。然后把食材放入水浴中，在恒定的、相对较低的温度下长时间烹煮。与传统方法的主要区别在于温度受到严格控制，不同食材平均为 55-85 °C。

低温慢煮可以烹饪各种各样的食材：肉、鱼、海鲜、蔬菜和水果。食材能保持原有的质地和天然外观，天然风味得以释放和增强。即使是较硬、较干的肉，也会变得多汁，入口即化。蔬菜不会变成索然无味的泥，还能保留轻微的脆感。独特的低温慢煮技术能把普通水果变成精致的甜点。煮过的杏、李子或甜瓜吃起来像果酱软糖。

Samovar 中的低温慢煮

Samovar 7.00: 模式列表中没有单独的“低温慢煮” («Су-вид») 模式（也没有 Lua 模式；这两项在 [设置](#) 中是隐藏的）。低温慢煮请使用 [啤酒](#) («Пиво») 模式：“暂停” («Пауза») (P) 程序行把水加热到设定温度并保持设定的时间，同时水泵按同一行中的计划搅动水，使容器中的温度均匀。

设置方法：

- . 在 [设置](#) 中选择“啤酒” («Пиво») 工作模式并保存。
- . 在 [设置 - 啤酒](#) («Настройки - Пиво») 选项卡上，设置保温模式下调压器的电压或功率；Samovar 在保温期间会用它补充加热。
- . 把循环泵连接到麦汁泵的输出端（见 [搅拌器、水泵、温度](#)），并把它放入盛水的容器中。蒸馏釜传感器必须浸没在水中。
- . 在啤酒模式页面上，设置由一行“暂停” («Пауза») 组成的程序：温度为食材所需的温度，时间为以分钟计的保温时长，并在“搅拌器” («Мешалка») 字段中开启水泵。程序行的格式见 [糖化程序](#) 一节。
- . 放入装袋的食材，开启加热。水温升到设定温度后计时器开始计时；程序结束时，Samovar 关闭加热和水泵。

示例：

P;65;300;2^0^600^0;0

该程序按蒸馏釜传感器（最后一个字段为 0）把水加热到 65°，并保持 300 分钟（5 小时）。“搅拌器”字段 2^0^600^0：设备 2 为水泵，运行时间 600 秒，暂停 0——水泵不停运转，搅动水。如果需要水泵间歇运行，请以秒为单位指定暂停时间，例如 2^0^60^120。

如果保温期间温度降到容差以下（例如加水之后），计时器会变慢；如果低于设定值 3° 或更多，计时器会停止，直到温度恢复，见 [糖化程序](#)。

问题与解答。FAQ

低温慢煮时，罐里的传感器必须浸在水中吗？如果它够不到液面怎么办？也许有某种修正差值？

可以多加点水。

也可以试着找一个差值。它会非常粗略，而且只适用于特定的容器和水量。

最好再安排水循环来搅拌。

在家庭烹饪中，相差一两度并不要紧（但 如果传感器在水面以上，差别会远远超过 1-2°），

奶酪制作模式

Samovar 7.00: 奶酪制作模式在这个版本中首次出现。

在奶酪制作模式中，Samovar 按程序执行奶酪制作过程：以设定速度加热牛奶、保持温度、冷却、按时间表开启搅拌器、根据 pH 传感器等待达到所需酸度、打开排液阀，并在需要奶酪师动手的地方停下（加入发酵剂、切割凝乳、装入模具）。程序由若干阶段组成，每个阶段是一行。各行的说明见[奶酪制作程序](#)页面。

设备

加热。主加热管通过 1 号继电器和调压器开启，与啤酒模式相同：当设定值与温度之差大于 [HEAT_DELTA](#) 时全功率加热，之后由 PID 调节器保持温度。此模式不使用辅助加热管——4 号继电器被排液阀占用。

温度传感器。在每个带温度的行中，选择该阶段所依据的传感器：“蒸馏釜”（«Куб»）、“水”（«Вода»）、“塔节”（«Цапра»）、“蒸汽”（«Пар»）或“TCA”（«TCA»）（角色在[温度](#)选项卡上分配）。通常容器里有一个传感器就够了，即“蒸馏釜”角色。其他传感器的读数在主页上显示，仅供参考。

搅拌器。可以通过 2 号继电器（开/关），也可以通过带步进电机的 [I2CStepper](#) 板——这时可以设置速度和方向。搅拌器在每个程序行中单独设置。

PH-4502C pH 传感器。通过 ADS1115 连接，或经电压匹配后接到 LUA_PIN 模拟输入端。接线图、电气隔离方案和滤波器的工作原理见技术章节[pH 传感器及其连接方法](#)。使用前请进行 [pH 校准](#)。

排液阀。连接到 4 号继电器。在其他模式中，这个继电器上接的是辅助加热管——阀门和加热管只能接其中一个，不能同时接。阀门在“排放”（«Слив»）阶段打开，在转到下一阶段以及程序以任何方式结束时关闭。

冷却。在“冷却”（«Охлаждение»）阶段使用。冷却方案取决于固件编译配置，显示在状态行的“冷却”字段中：

水泵 ([#define USE_WATER_PUMP](#)) ——供水阀打开，冷却泵启动；温度高出目标值超过 1°C 时，水泵全功率运行，接近目标值时以最小功率 ([PWM_LOW_VALUE](#)) 运行；

两个阀门 ([#define USE_WATER_VALVE](#)) ——供水阀打开，[WATER_PUMP_PIN](#) 引脚上的第二个阀门也打开，以加大水流；

不可用——编译配置中既没有水泵也没有阀门；带“冷却”阶段的程序不会启动。

投料器（可选）。“计量投料”（«Дозирование»）阶段可以用 Samovar 的标准蠕动泵加入配料——体积和速度按水泵校准换算成步数（[水泵](#)）。

警告。

没有 ADS1115 时，奶酪制作期间 LUA_PIN 输入端被 pH 传感器占用：不要把 PH-4502C 和 MPX5010DP 同时接到这个输入端。pH 通过 ADS1115 连接时，这个输入端保持空闲。在奶酪制作模式中，4 号继电器控制排液阀。

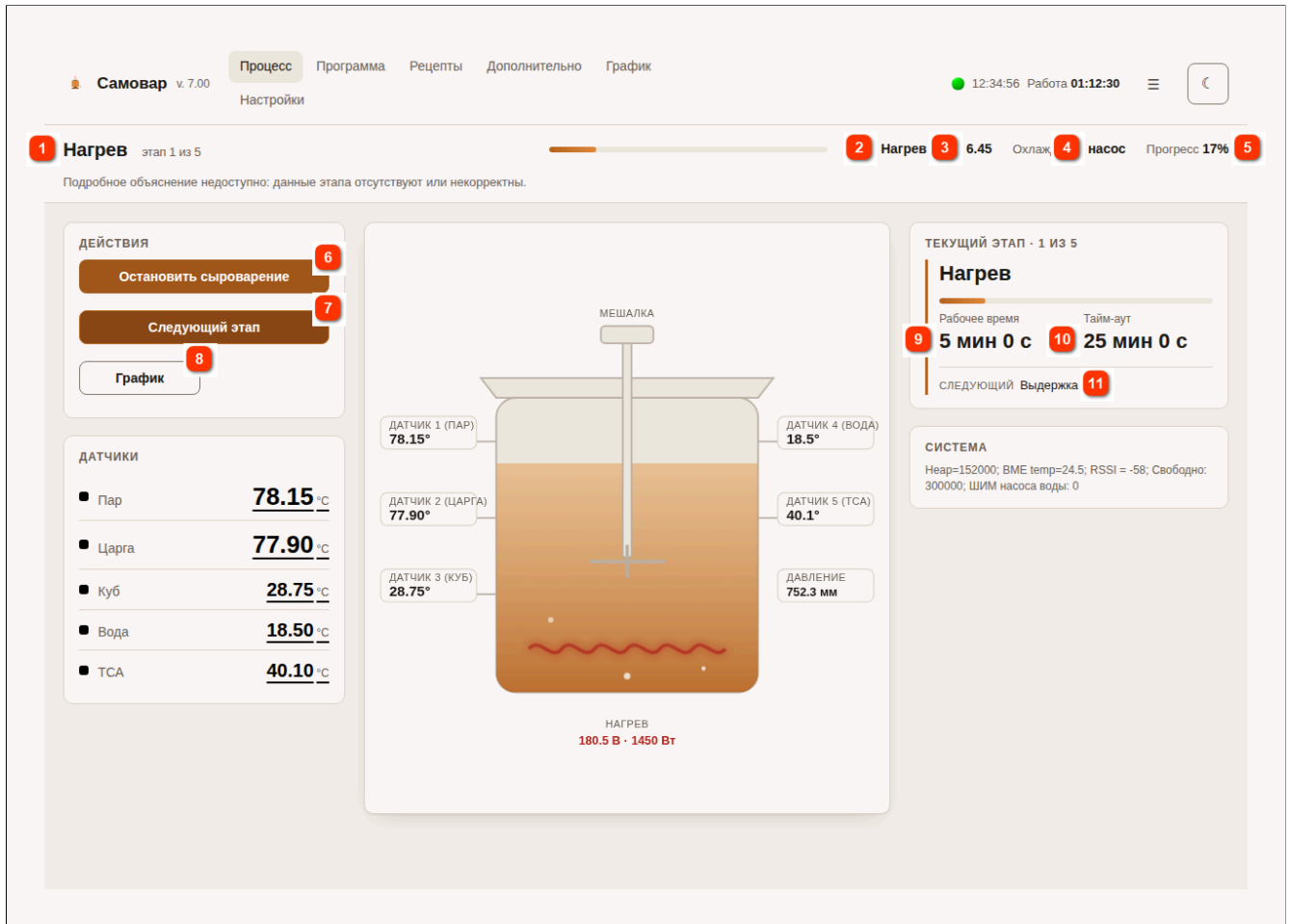
如何启用

打开[设置 → 基本](#)，在“运行模式”（«Режим работы»）字段中选择“奶酪制作”（«Сыроварение»）并保存设置。Samovar 的主页就变成奶酪制作页面。与该模式相关的设置集中在[设置 → 奶酪](#)选项卡上。

首次启用该模式时，会加载默认程序（加热到 63°C、保温、冷却到 34°C、加入发酵剂和凝乳酶、切割、排放和压榨）。请在“程序”（«Программа»）选项卡上换成自己的程序。

模式主页

页面由四个选项卡组成：“过程”（«Процесс»）、“程序”（«Программа»）、“配方”（«Рецепты»）和“附加”（«Дополнительно»）。“图表”（«График»）按钮打开[本地图表](#)。



奶酪制作模式主页，“过程”（«Процесс»）选项卡

- 状态**
当前阶段的名称及其编号/阶段总数。程序未运行时显示“等待”（«Ожидание»）。
- 阶段**
当前阶段的类型：加热（«Нагрев»）、保温（«Выдержка»）、冷却（«Охлаждение»）、搅拌（«Перемешивание»）、计量投料（«Дозирование»）、等待 pH（«Ожидание pH»）、手动操作（«Ручное действие»）、排放（«Слив»）或 Lua。
- pH**
PH-4502C 传感器给出的当前酸度。如果传感器未连接、未校准或数值超出 0-14，显示“无数据”（«нет данных»）。
- 冷却**
此固件编译配置中的冷却方案：“水泵”（«насос»）、“两个阀门”（«два клапана»）或“不可用”（«недоступно»）。
- 进度**
阶段开始以来已过去的时间占“超时，min”（«Тайм-аут, мин»）字段的比例（保温阶段相对于保温时间）。
- 开始/停止奶酪制作**
从第一个阶段开始运行程序。运行期间，该按钮名为“停止奶酪制作”（«Остановить сыроварение»），确认后结束程序。
- 下一阶段/确认**
转到下一阶段。在“手动操作”、“排放”和手动“计量投料”阶段，该按钮名为“确认”（«Подтвердить»）——不点击它，阶段不会结束。
- 图表**
转到本地温度图表。
- 运行时间**
阶段已运行的时间。对于保温阶段，温度低于容差范围的时间按部分计入，低 3°C 及以上的时间和高于容差的过热时间不计入。
- 超时**
距离阶段因超时被强制停止还剩多少时间。
- 当前阶段和下一阶段**
当前阶段卡片：名称、进度、运行时间、超时，以及程序中下一阶段的名称。

在手机屏幕上，显示的不是单独的卡片，而是“关键数据”（«Ключевое»）区：容器温度、阶段设定值、当前温度、运行时间和超时。

“传感器”（«Датчики»）区显示所有已连接的温度传感器，中间的示意图显示容器及其当前温度。底部是“系统”（«Система»）行：可用内存（Heap）、板子温度（BME temp）、Wi-Fi 信号强度（RSSI）、文件系统可用空间，以及使用水泵时的 PWM。

“程序”选项卡是阶段编辑器，见[奶酪制作程序](#)。

“配方”选项卡转到[网站奶酪配方目录](#)页面：samovar-tool.ru 配方目录、按配方预览程序并应用。

“附加”选项卡上有“pH 校准”（«Калибровка pH»）按钮（转到 [pH 校准](#) 页面）、关于 PH-4502C 和 4 号继电器接线的提示，以及执行 [Lua](#) 命令的字段和 Lua 按钮（如果在 btn_cheese_button1.lua 和 btn_cheese_button2.lua 文件中定义了它们）。

操作流程

- 分配温度传感器，必要时校准 pH 传感器和投料泵。
- 在“程序”选项卡上编写程序，然后点击“保存程序”（«Сохранить программу»）。
- 在“过程”选项卡上点击“开始奶酪制作”（«Начать сыроварение»）。Samovar 会检查程序：带温度的阶段必须指定传感器；使用 I2C 搅拌器时必须连接 I2CStepper 板；使用本地投料器时必须完成水泵校准；“冷却”阶段需要有可用的冷却方案；“等待 pH”（«Ожидание pH»）阶段需要已保存的 pH 校准；“Lua”阶段需要存在 /cheese.lua 文件。如果检查不通过，消息中会显示原因和行号，加热不会开启。
- 检查通过后，创建日志文件，开启加热，开始第一个阶段。每次进入一个阶段都会显示“第 N 行；阶段名称”（«Строка N; название этапа»）消息。
- “手动操作”（«Ручное действие»）、“排放”以及手动方式的“计量投料”阶段会等待奶酪师：完成操作后点击“确认”（«Подтвердить»）（此时“下一阶段”（«Следующий этап»）按钮就叫这个名字）。“下一阶段”按钮可以提前结束任何阶段。短按 [万能按钮](#) 效果相同，长按则结束奶酪制作。
- 最后一个阶段结束后，Samovar 关闭加热、冷却、搅拌器和投料器，关闭排液阀，并显示“奶酪制作程序已完成”（«Программа сыроварения завершена»）消息。状态行显示“等待”（«Ожидание»）。

切换阶段前，所有执行机构都会进入安全状态（关闭加热和冷却，停止搅拌器和投料器，关闭排液），然后再开启新阶段所需的设备。

停止、超时和错误

停止奶酪制作（«Остановить сыроварение»）——随时结束程序（需要确认）。结果与正常结束相同：所有输出关闭，排液阀关闭。

超时。每个阶段都有时间限制（“超时，min”（«Тайм-аут, мин»）字段；“保温”（«Выдержка»）阶段为“总超时，min”（«Общий тайм-аут, мин»））。如果阶段在这段时间内没有完成，奶酪制作会停止，并显示“第 N 行：奶酪程序操作超时”（«Строка N: Тайм-аут операции сырной программы»）消息（保温阶段为“保温超时”（«Тайм-аут выдержки»））。例外是“搅拌”（«Перемешивание»）：它的持续时间本身就是完成条件。

阶段错误也会停止程序，并显示“第 N 行：原因”消息：所选温度传感器没有数据、搅拌器或冷却错误、“等待 pH”阶段 pH 不可靠超过 10 秒、Lua 脚本错误。

安全系统的工作方式与啤酒模式相同：容器温度高于 [BOILING TEMP](#) + 5°C、冷却阶段超过水或 TCA 的极限温度、没有水流（如果启用了流量监控）时紧急关闭。过热和紧急情况见[紧急按钮](#)、[附加紧急传感器](#)页面。

奶酪制作过程中重启。Samovar 会记住模式和当前阶段编号。重启后加热不会恢复，消息中会出现“会话因重启中断：第 N/M 行”（«Сессия прервана перезагрузкой: строка N/M»）。如果快照中的程序未通过检查，它会被清除，也不会换成默认程序——请重新编写程序。

奶酪制作程序

程序在**奶酪制作**模式主页面的“程序” («Программа») 选项卡上设置。每一行是一个阶段；各阶段按顺序执行。一个程序最多可以有 20 个阶段。

奶酪制作模式的“程序”选项卡 («Вкладка «Программа» режима Сыроварение»)

- 1 阶段编号 («Номер этапа»)**
行的序号。各阶段从上到下执行。
- 2 阶段类型 («Тип этапа»)**
加热、保温、冷却、搅拌、计量投料、等待 pH、手动操作、排放或 Lua ("Нагрев, Выдержка, Охлаждение, Перемешивание, Дозирование, Ожидание pH, Ручное действие, Слив, Lua")。更改类型时，该行的字段重置为默认值。
- 3 数值 1 («Значение 1»)**
目标温度（计量投料时为体积，ml）。字段上方的标签随阶段类型变化。
- 4 数值 2 («Значение 2»)**
阶段超时，单位分钟；保温时为保温时间，搅拌时为持续时间。
- 5 数值 3 («Значение 3»)**
加热时为速度，°C/min；保温时为总超时；等待 pH 时为目标 pH；计量投料时为速度或组分。手动操作时这里是操作列表。
- 6 搅拌器 («Мешалка»)**
搅拌设备：NONE、RELAY (2 号继电器) 或 I2C (I2CStepper 板)。
- 7 RPM**
I2C 搅拌器的转速。对于 RELAY 和 NONE 为 0。旋转方向在旁边的列表中选择。
- 8 运行, s («Работа, с»)**
搅拌器运行时间，单位秒。0——搅拌器在整个阶段保持开启。
- 9 暂停, s («Пауза, с»)**
两次启动之间的暂停。设定了运行时间而暂停为 0——搅拌器只接通一次。
- 10 传感器 / 方式 («Датчик / Способ»)**
该阶段所依据的温度传感器（蒸馏釜、水、塔节、蒸汽、TCA；"Куб, Вода, Царга, Пар, TCA")。计量投料时为方式：“手动”或“本地步进电机” (“Вручную” or “Локальный шаговый двигатель”)。
- 11 添加行 («Добавить строку»)**
在当前行之后插入新阶段。
- 12 删除行 («Удалить строку»)**
删除该行。最后剩下的一行不能删除。
- 13 保存程序 («Сохранить программу»)**
检查所有行并把程序发送到 Samovar。旁边是“导出”到文件和从文件“导入” (“Экспорт”, "Импорт")。

更改阶段类型时，该行的字段会重置为该类型的默认值，不需要的字段会被隐藏。保存时会检查数值：温度 0 到 150°C，时间 0 到 1440 min。出错时显示提示“请检查第 N 阶段的字段” («Проверьте поля этапа N») ，这样的程序不会发送到 Samovar。

保存程序 («Сохранить программу») - 把程序发送到 Samovar。制作过程中，“过程” («Процесс») 选项卡使用的是最后保存的程序，而不是未保存的修改。

导出 («Экспорт») - 把程序下载到 cheese-program.txt 文件。

导入 («Импорт») - 从文本文件加载程序。文件的检查方式与编辑器中的行相同；出错时当前程序不变。

行格式

在文件和 Samovar 内存中，一行以六个用分号分隔的字段保存：

[Type;Value1;Value2;Value3;Stirrer;Value4](#)

"Value1-4"字段的含义取决于阶段类型（见下表）。"Stirrer"字段是用 [△](#) 分隔的四个数：设备^转速^运行时间，s^暂停时间，s。例如：

[H;63;90;1;1^0^0^0;0](#) - 以 1°C/min 加热到 63°C，超时 90 min，搅拌器通过继电器在整个阶段运行，使用"蒸馏釜"传感器；

[M;0;1;0;1^0^30^0;0](#) - 搅拌：搅拌器通过继电器接通一次，持续 30 s；

[W;0;15;1;0^0^0^0;0](#) - 手动操作"加入发酵剂"，等待确认，最多 15 min；

[N;32;120;6.4;0^0^0^0;0](#) - 保持 32°C，直到 pH 降到 6.4，最多 120 min。

阶段类型

除"搅拌"（«Перемешивание»）外的所有阶段中，"超时，min"（«Тайм-аут, мин»）字段是该阶段的最长时间：如果阶段在此时间内没有结束，奶酪制作停止。
"传感器"（«Датчик»）是该阶段所依据的温度传感器："蒸馏釜"（«Куб»）、"水"（«Вода»）、"塔节"（«Цагра»）、"蒸汽"（«Пар»）或"TCA"（«TCA»）。当传感器读数连续 10 秒保持在目标值 $\pm 0.3^\circ\text{C}$ 以内时，认为温度已达到。

H - 加热（«Нагрев»）。字段："目标，°C"（«Цель, °C»）、"超时，min"、"速度，°C/min"（«Скорость, °C/мин»）、"传感器"。加热设定值从起始温度按给定速度上升，直到达到目标。传感器达到目标时阶段结束。

P - 保温（«Выдержка»）。字段："目标，°C"、"保温，min"（«Выдержка, мин»）、"总超时，min"（«Общий тайм-аут, мин»）、"传感器"。Samovar 保持目标温度。温度与目标的差不超过容差时，保温时间全额计入。容差是"设置 - 温度"选项卡（«Настройки - Температура»）上该阶段传感器的"超出设定值"（«Вылет за уставку»）字段：0.3°C 到 3°C；低于 0.3°C 的值（包括 0）按 0.3°C 处理。温度低于目标超过容差时，时间部分计入——差得越多，计入越少；低 3°C 或更多，以及高出容差时，时间完全不计：会收到"保温暂停：温度超出范围"（«Выдержка приостановлена: температура вне полосы»）消息，温度恢复后收到"保温继续"（«Выдержка продолжена»）。累计达到设定时间时阶段结束。总超时从阶段开始时计算，不能小于保温时间；超时后过程停止，提示"保温超时"（«Тайм-аут выдержки»）。

C - 冷却（«Охлаждение»）。字段："目标，°C"、"超时，min"、"传感器"。加热关闭；温度高于目标时供应冷却水（高于目标超过 1°C 时全流量，更接近时最小流量）。传感器达到目标时阶段结束，随后关闭冷却。需要"泵"或"双阀"冷却方案。

M - 搅拌。字段："持续时间，min"（«Длительность, мин»）。需要搅拌器。持续时间到达时阶段结束；如果搅拌器设定了运行时间且没有暂停，则在运行该时间一次后立即结束。该阶段没有超时。

D - 计量投料（«Дозирование»）。字段："体积，ml"（«Объём, мл»）、"超时，min"、"方式"（«Способ»）以及取决于方式的第三个字段。

"手动"（«Вручную»）- 第三个字段为"组分"（«Компонент»）（列表与手动操作相同）。Samovar 等待奶酪制作者加入组分并按"确认"（«Подтвердить»）。

"本地步进电机"（«Локальный шаговый двигатель»）- 第三个字段为"速度，ml/min"（«Скорость, мл/мин»）。体积由 Samovar 的标准蠕动泵输送；体积和速度按泵的校准换算为步数，因此泵必须经过校准（[泵](#)）。泵输送完设定体积时阶段结束。计量装置的各种方案见[添加剂计量装置](#)页面。

N - 等待 pH（«Ожидание pH»）。字段："温度，°C"（«Температура, °C»）、"超时，min"、"目标 pH"（«Целевой pH»）、"传感器"。Samovar 保持温度并监视 pH 传感器。pH 连续 30 秒不高于目标时阶段结束。如果 pH 读数不可靠（无数据或超出 0-14）超过 10 秒，奶酪制作停止，提示"pH 连续 10 秒不可靠"（«pH недостоверен 10 секунд»）。该阶段需要已保存的[pH 校准](#)。

W - 手动操作（«Ручное действие»）。字段："超时，min"、"操作"（«Действие»）。Samovar 不做任何事（如果设定了搅拌器，搅拌器除外），等待按下"确认"。操作列表："加入发酵剂"（«Внести закваску»）、"加入酶"（«Внести фермент»）、"加入添加剂"（«Внести добавку»）、"切割凝乳"（«Разрезать сгусток»）、"装入模具"（«Переложить в формы»）、"开始或改变压制"（«Начать или изменить прессование»）、"翻转奶酪"（«Перевернуть сыр»）、"其他操作"（«Другое действие»）。操作名称在程序中起提示作用；在状态行中 Samovar 只显示"手动操作"。

S - 排放（«Слив»）。字段："超时，min"。排放阀（4 号继电器）打开，Samovar 等待按下"确认"。转到下一阶段时阀门关闭。此阶段不能设置搅拌器。

L - Lua。字段："超时，sec"（«Тайм-аут, сек»）。启动在"Lua 阶段"（«Lua-этап»）窗口中所选文件的脚本（Samovar 内存中的任意 .lua 文件；超时以及脚本在 arg 表中获得的参数也在那里设置，见 [Lua](#)）。要转到下一阶段，脚本必须调用 [setNextProgram\(\)](#)。如果脚本没有调用它就结束，或因错误结束，奶酪制作停止。此阶段不能设置搅拌器。

搅拌器

搅拌器在每一行中单独设置，只在该阶段运行。共五个字段：

搅拌器（«Мешалка»）- 设备："NONE"——没有搅拌器；"RELAY"——2 号继电器（开/关）；"I2C"——[I2CStepper](#) 板上的步进电机。选择"I2C"时必须连接该板，否则程序不会启动。

RPM - "I2C"的转速，大于零的整数。对于"RELAY"和"NONE"必须为 0。

方向（«Направление»）- "正向"（«Прямое»）或"反向"（«Обратное»），仅用于"I2C"。在文件中，反向写作负转速（例如 [2^-120^0^0](#)）。

运行，s（«Работа, с»）和**暂停**，s（«Пауза, с»）- 运行时间表。运行行为 0——搅拌器在整个阶段保持开启。运行大于 0、暂停为 0——搅拌器接通一次，持续给定时间后关闭。运行和暂停都大于 0——搅拌器交替运行和暂停，直到阶段结束。不允许只有暂停而没有运行。

转到下一阶段时，搅拌器停止，并按新行的设置重新启动。

网站奶酪配方目录

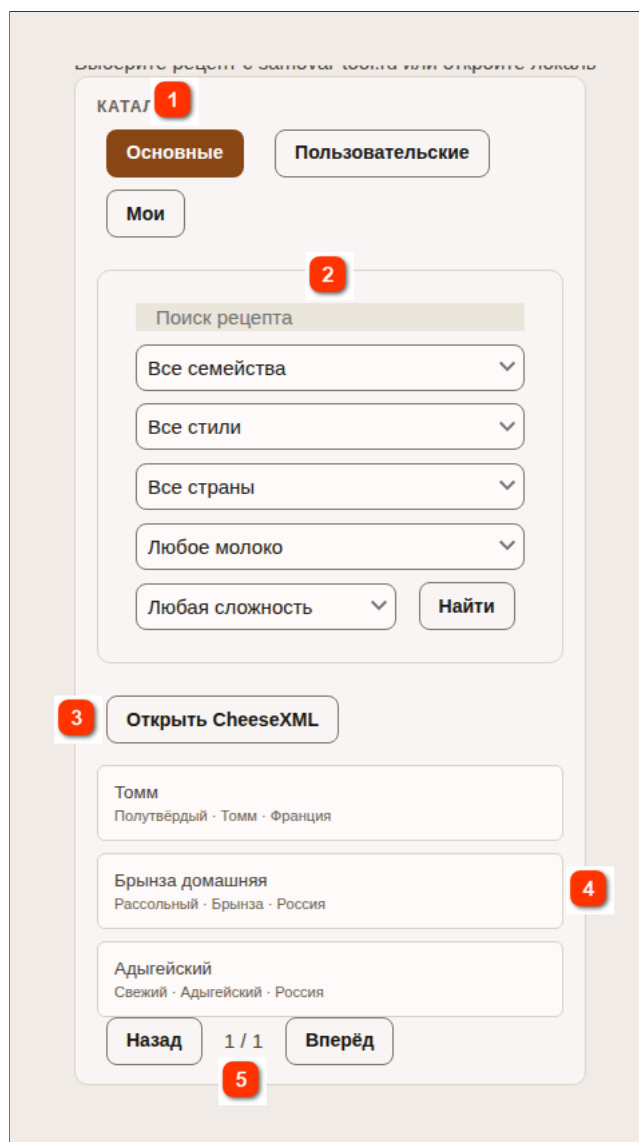
[奶酪制作模式](#)主页面上的“配方”（«Рецепты»）选项卡会打开此页面——samovar-tool.ru 的奶酪配方目录，可以预览，并能直接在浏览器中应用程序，无需把文件保存到电脑。

目录

配方来源有三种：“主要”（«Основные»）、“用户”（«Пользовательские»）和“我的”（«Мои»）（列表上方的按钮）。使用“我的”来源时，必须在设备设置中设定 Blynk 令牌，否则会显示消息“设备中未为‘我的’列表设定 Blynk 令牌。”（«Для списка «Мои» в устройстве не задан токен Blynk.»），筛选器也不可用：只能在已加载的配方中按名称搜索。

对于“主要”和“用户”来源，可以按名称搜索并使用筛选器：奶酪类别、风格、国家、奶的种类、难度。配方列表分页显示，用“上一页”（«Назад»）和“下一页”（«Вперёд»）按钮翻页。

搜索框旁边的“打开 CheeseXML”（«Открыть CheeseXML»）按钮从电脑上的本地 .xml 文件加载配方——效果与从目录中选择配方相同，只是预览面板中显示的来源为“本地 CheeseXML”（«Локальный CheeseXML»）。



“配方”（«Рецепты»）页面上的奶酪配方目录

- 1 配方来源**
“主要”（«Основные»）、“用户”（«Пользовательские»）或“我的”（«Мои»）（“我的”需要在设备设置中设定 Blynk 令牌）。
- 2 搜索和筛选器**
按名称搜索以及筛选器：奶酪类别、风格、国家、奶的种类、难度。“我的”来源不能使用筛选器。
- 3 打开 CheeseXML**
从电脑上的本地 .xml 文件加载配方，而不是从网站目录加载。
- 4 配方列表**
点击配方即可将其加载到程序预览面板。

分页列表

5 用“上一页” («Назад») / “下一页” («Вперёд») 按钮翻页。

程序预览

点击列表中的配方（或加载本地文件）后，它会在右侧面板中打开：名称、来源、阶段数。如果配方中某个阶段给出的是温度范围而不是固定值，“转换” («Преобразовать») 按钮上方会出现一个字段，用于从该范围中选择一个值——不选择就无法转换（“阶段...：请从范围中选择一个温度值。” («Этап «...»: выберите одно значение температуры из диапазона.») 。

“转换” («Преобразовать») 按钮根据配方阶段生成 Samovar 程序，并在只读文本框中显示，同时列出警告。例如：自动控制无法执行的阶段（手动添加发酵剂、切割凝乳等）会变为手动操作，并显示消息“阶段...不受自动控制支持，已转换为手动操作。” («Этап «...» не поддерживается автоматикой и преобразован в ручное действие.»)；成熟阶段不包含在程序中（“成熟阶段已从 Samovar 程序中排除。” («Стадии созревания исключены из программы Самовара.»)）；如果无法从配方中获取阶段超时，Samovar 会使用自己的超时值——对此也会显示警告。程序不能超过 30 个阶段——超过时会显示消息“程序超过 30 个阶段，无法应用。” («В программе больше 30 этапов; применение недоступно.»)，程序无法应用。

程序预览：从范围中选择温度、转换和警告

- 1 **配方**
名称、来源（目录、“我的配方” («Мой рецепт») 或“本地 CheeseXML” («Локальный CheeseXML»)）和阶段数。
- 2 **选择温度**
对于给出温度范围而非固定值的阶段，必须从范围中选择一个值。
- 3 **转换**
根据配方阶段生成 Samovar 程序。
- 4 **警告**
例如：哪些阶段变成了手动操作，哪里使用了默认超时。
- 5 **程序预览**
完成的程序行，只能查看。不超过 30 个阶段。

6**应用程序**

仅在“奶酪”（«Сыр»）模式下可用；将程序发送到设备，但不启动奶酪制作。

7**打开编辑器**

转到奶酪制作模式的“程序”（«Программа»）选项卡，可在那里修改并保存程序。

“应用程序”（«Применить программу»）按钮将完成的程序发送到 Samovar 内存——与奶酪制作模式“程序”（«Программа»）选项卡上的“设置程序”（«Установить программу»）相同，但不会启动奶酪制作（“设备已接受程序，未启动。”（«Программа принята устройством. Запуск не выполнялся.»））。只有当 Samovar 处于“奶酪”（«Сыр»）模式且转换没有阻塞性错误时，该按钮才可用；否则按钮下方会显示以下消息之一：“如需应用，请将 Samovar 切换到‘奶酪’模式。”（«Для применения переключите Самовар в режим «Сыр».»）或“该转换无法应用。”（«Преобразование нельзя применить.»）。

“打开编辑器”（«Открыть редактор»）链接通往奶酪制作模式的主页面（“程序”（«Программа»）选项卡），您可以在那里查看已应用的程序，进一步修改并按常规方式保存，见 [奶酪制作程序](#)。

pH 校准

PH-4502C pH 传感器输出电压，Samovar 在 [LUA_PIN](#) 输入端将其读取为 ADC 值 (0-4095)，并按公式 $pH = k \cdot ADC + d$ 换算为 pH。每支电极都有各自的系数 k 和 d，因此在第一次制作之前以及更换电极之后，必须用两种缓冲溶液校准传感器。未校准时，主页面上的 pH 栏显示“无数据” («нет данных») ，程序中的“等待 pH” («Ожидание pH») 阶段也不会开始。

校准页面可通过 [奶酪制作](#) («Сыроварение») 模式“附加” («Дополнительно») 选项卡上的“pH 校准” («Калибровка pH») 按钮打开，也可在 [设置 → 奶酪](#) («Настройки → Сыр») 选项卡上打开。传感器只在奶酪制作模式下轮询，在其他模式下页面显示“无数据”。

Самовар Режим График Рецепты Настройки

Калибровка pH

К сыроварению Настройки

Текущее значение **1** 6.45 pH - ADC: 1830 **2**

Промойте электрод, поместите его **3** в раствор и дождитесь стабильного ADC. Повторите для второго буфера.

pH первой точки: 7.00 **4** ADC: — **5** Зафиксировать **6**

pH второй точки: 4.00 **6** ADC: — Зафиксировать

Рассчитать **7**

Коэффициент k: -0.0035 **8**

Смещение d: 12.87 **9**

Сохранить калибровку **10**

Коэффициенты рассчитаны. Сохраните их. **11**

Без USE_ADS1115 PH-4502C подключается к LUA_PIN через согласование уровня до 3,3 В и конфликтует с MPX5010DP.

pH 传感器校准页面

- 1** 当前值 («Текущее значение») 用已保存的系数换算出的 pH。“无数据” («нет данных») 表示传感器未被轮询或结果超出 0-14。
- 2** ADC LUA_PIN 输入端的原始读数 (0-4095)，每秒更新一次。校准点从这里记录。
- 3** 第一点 pH («pH первой точки») 第一种缓冲溶液的 pH (默认 7.00)。
- 4** 点 ADC («ADC точки») 记录该点时保存的 ADC 值。
- 5** 记录 («Зафиксировать») 将当前 ADC 与输入的 pH 一起保存。需要传感器的新数据以及 0 到 14 之间的 pH。
- 6** 第二点 pH («pH второй точки») 第二种缓冲溶液的 pH (默认 4.00)。第二点的 ADC 必须与第一点不同。
- 7** 计算 («Рассчитать») 根据记录的两个点计算 k 和 d，并填入下方各栏。

系数 k («Коэффициент k»)

斜率：每一个 ADC 单位对应的 pH 变化。可手动输入。

8

9

偏移 d («Смещение d»)

偏移量：ADC = 0 时的 pH。可手动输入。

10

保存校准 («Сохранить калибровку»)

将 k 和 d 写入 Samovar 设置。允许范围为 -100 到 100。

11

消息 («Сообщение»)

上一次操作的结果：“第 1 点已记录” («Точка 1 зафиксирована»)、“系数已计算，请保存” («Коэффициенты рассчитаны. Сохраните их»)、“校准已保存” («Калибровка сохранена») 或错误文本。

校准步骤

- . 冲洗电极，将其放入第一种缓冲溶液，等待 ADC 值不再变化。读数每秒更新一次。
- . 在“第一点 pH” («pH первой точки») 栏中输入缓冲液的 pH (默认 7.00)，然后点击“记录” («Зафиксировать») 。旁边会显示保存的 ADC 值。如果没有传感器数据或 pH 超出 0-14，会出现消息“需要新的 ADC 和 0 到 14 之间的 pH” («Нужны свежий ADC и pH от 0 до 14») 。
- . 冲洗电极，将其放入第二种缓冲液 (默认 4.00)，用同样的方法记录第二点。两点的 ADC 值必须不同。
- . 点击“计算” («Рассчитать») 。系数由两点算出： $k \equiv (pH2 - pH1) / (ADC2 - ADC1)$ ， $d \equiv pH1 - k \cdot ADC1$ ，并填入“系数 k” («Коэффициент k») 和“偏移 d” («Смещение d») 栏。
- . 点击“保存校准” («Сохранить калибровку») 。系数写入 Samovar 设置，重启后仍然保留。在带中值滤波的固件中，系数改变后需要等待五次新的测量：之后“当前值” («Текущее значение») 一行将显示电极所在缓冲液的 pH。

如果已知系数，也可以手动输入并直接保存，无需记录测量点。k 和 d 的允许范围为 -100 到 100。首次校准之前，设置中为 k = 1、d = 1：换算结果超出 0-14，Samovar 认为 pH 不可信。

检查。保存后，将电极放入缓冲液，比较“当前值”一行与缓冲液的 pH。模式主页面和“等待 pH”阶段显示的也是同一个值。在带中值滤波的固件中，pH 由间隔一秒的五次连续测量确定。ADC 栏仍是用于校准的原始读数：记录每个点之前请等待它稳定。

注意。

将 PH-4502C 接到 LUA_PIN 时需要进行电平匹配；MPX5010DP 不得接在同一输入端。使用 ADS1115 时，LUA_PIN 输入端保持空闲。接线图和维护见 [pH 传感器及其连接](#) 一节。

手动模式下的工作

手动控制

为此可以只设置一行程序，例如：

H;65535;0.10;0;0;120

一行中的体积不能超过 65535 ml，并且第一行程序必须指定调压器的电压（功率），否则取酒不会开始。

发出“开启加热”（«Включить нагрев»）命令后，Samovar 让调压器开始快速升温。达到设定温度后，Samovar 打开冷却水，结束快速升温，转入精馏塔稳定阶段。按下“开始取酒”（«Начать отбор»）后，Samovar 转到上面这行程序并按它运行，直到以 100 ml/h 的速度取出虚拟的 65.5 升（ $65.535/0.1 = 655$ 小时）；在此期间可以控制取酒参数。可以修改[取酒速度](#)和[调压器电压](#)。

安全保护组、设定水温的保持以及“精馏、蒸馏和 BK 直到蒸馏釜温度”（«Ректификация, дистилляция и БК до температуры в кубе»）设置也会起作用。

不考虑温度差值以及温度计超出设定值的情况。在酒头行上，杂质检测器只做观察。分酒台的摆臂始终停在 0 号位置。

注意：

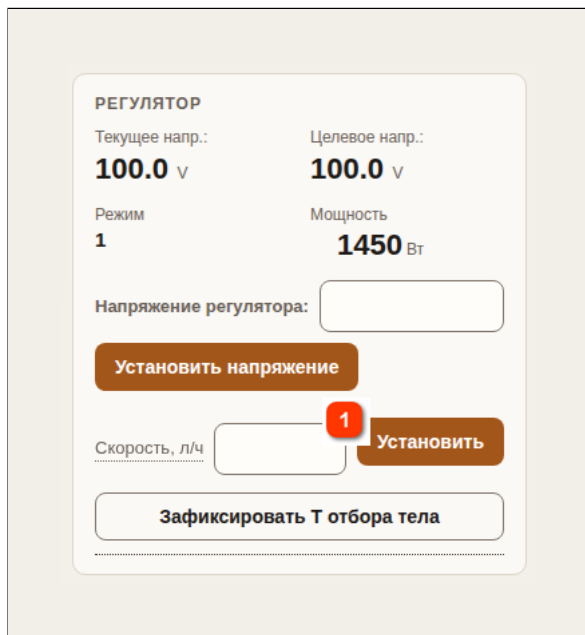
可以随时切换到程序自动执行。只需添加所需模式的程序行，执行“设置程序”（«Установить программу»）命令，然后在主窗口中按“下一步”（«Далее»）。

Samovar 将按照内置算法开始执行新程序。手动调整调压器电压和取酒速度仍然可用。

手动设置取酒速度

当前运行程序的取酒速度可以在“精馏” («Ректификация») 选项卡 ([主 页面](#)) 的“调节器” («Регулятор») 卡片中修改。

Samovar 7.00: 取酒速度字段已从“高级” («Дополнительно») 选项卡移到主屏幕的“调节器”卡片中，名称为“速度，升/小时” («Скорость, л/ч»)。



“精馏” («Ректификация») 选项卡上的“调节器” («Регулятор») 卡片

1

数值

输入以升/小时为单位的取酒速度。点击“设置” («Установить») 使其生效

程序运行期间可以随时修改取酒速度，次数不限。手动设置的取酒速度优先于程序中的速度，按下“设置” («Установить») 后立即生效，并一直有效到 Samovar 转入下一行程序为止。取酒程序未运行时，该命令不执行。

修改速度后，Samovar 会重新计算当前行的剩余时间；对杂质穿透检测器来说，新速度成为基准速度。

提示。

因此，如果您已经“找到”了合适的速度（例如蒸馏芳香原料时），可以修改下一行程序，使速度在切换时保持不变。别忘了点击“设置程序” («Установить программу»)。

手动设置调压器电压

在取酒程序运行且加热开启时，可以随时手动修改送到调压器的电压。操作位置在“[主页](#)”（«Главная»）选项卡的“调节器”（«Регулятор»）卡片中进行

РЕГУЛЯТОР

Текущее напр.: 100.0 V

Целевое напр.: 100.0 2

Режим 1

Мощность 1450 Вт

Напряжение регулятора:

Установить напряжение

Скорость, л/ч

Установить

Зафиксировать Т отбора тела

“精馏”（«Ректификация»）选项卡上的“调节器”（«Регулятор»）卡片

1

手动输入

输入以伏为单位的数值（Stab-avr 调压器以瓦为单位），然后点击“设置电压”（«Установить напряжение»）按钮。

2

目标电压（«Целевое напр.»）

调压器当前设定的电压（功率）：来自程序行或手动输入。

输入以伏为单位的数值（例如：150 或 171），然后点击“设置电压”（«Установить напряжение»）按钮。对于 Stab-avr 调压器，该字段名为“调压器功率”（«Мощность регулятора»），数值以瓦为单位输入，按钮为“设置功率”（«Установить мощность»）。手动设置的电压优先于程序中的电压，并一直有效到 Samovar 转入取酒程序的下一行为止

注意： 如果希望手动设置的电压在取酒程序的下一行中继续生效，请在该行程序的“电压”（«Напряжение»）字段中填 0（零）

注意： 取酒程序的下一行开始时，调压器会被设为该行程序中指定的电压，但输入框中仍显示操作员手动输入的数值。要了解调压器当前实际按哪个设定值工作，请比较两个数值：输入框中的数值和“调节器”卡片中的“目标电压”（«Целевое напр.»）。

警告： 在“预液泛”（«Предзахлеб»）模式下，手动修改电压会生效，但 Samovar 仍按精馏塔的 [工作算法](#) 运行（预液泛 模式）。

警告： 在 [NBK](#) 模式下，在“优化”（«Оптимизация»）和“工作”（«Работа»）阶段不执行手动修改电压，Samovar 会显示提示信息“NBK 优化和工作阶段不可手动控制”（«Ручное управление недоступно на Оптимизации и в Работе НБК»）。

监控模式下的工作

如果不发出“开启加热” («Включить нагрев») 命令，Samovar 会在网页界面中显示所有传感器的当前温度和大气压力。温度读数也会发送到 Blynk（每 5 秒一次），可在手机应用中查看。这样就可以监控任何设备的温度，无论是精馏塔、糖化醪（发酵）容器、壶式回流塔，还是不受自动控制的间接加热蒸馏器。

警告！

在此模式下**安全保护组不起作用**：不监控蒸汽、水、蒸馏釜以及附加输入端传感器的极限温度。不记录日志（既没有本地日志，也没有 samovar-tool.ru 服务器上的日志），也不绘制图表。

不建议使用此模式，过程安全完全由操作者负责！

功能扩展

通过 I2C 总线连接的附加模块和附加脚本，可以大大扩展 Samovar 的功能：

PCF8575 —— 带 16 个数字端口的扩展器。在固件中用 `#define USE_EXPANDER 0x20` 这一行启用（模块在总线上的地址）。

PCF8591 —— 带四路模拟输入和一路模拟输出的扩展器。用 `#define USE_ANALOG_EXPANDER 0x48` 这一行启用。

[I2CStepper](#) —— 带步进电机（用于搅拌机或计量泵）和四个继电器的开发板。Samovar 启动时自动识别。

附加功能的编程基于[脚本语言 Lua](#)实现。Lua 支持在编译固件时启用：在配置器中勾选“使用 Lua”（«Использовать Lua»）复选框（“硬件”（«Оборудование»）部分）；在 PlatformIO 中使用 **Samovar_lua** 环境；手动编译时在 `user_config_override.h` 文件中加入 `#define USE_LUA` 这一行。脚本编辑器通过设置页面上的“编辑器”（«Редактор»）按钮打开。

端口扩展器上可以连接任何执行机构或按钮（以按钮方式工作的传感器），并且可以自行规定 Samovar 运行过程中某一时刻应执行的动作。

Samovar 启动时运行初始化脚本，之后每秒运行另一个脚本。在第一个脚本中描述所需的设备，在第二个脚本中描述它们的行为。例如，可以检查温度或 Samovar 的当前状态，并/或开关水泵、打开或关闭阀门、控制附加加热管。

可以在 Samovar 的内存中保存变量，运行最多 10 个定时器，获取 Samovar 的运行状态，控制 I2CStepper 板的搅拌机、泵和继电器，以及发送 HTTP 请求。也就是说，可以连接附加设备，并在 Samovar 运行时对其进行自动控制。

可以检查安全传感器是否触发；如果已触发，可以直接从脚本关闭 Samovar，或者加大或停止供水。可能的场景很多，取决于具体设备。

Samovar 的基本功能不会改变。

I2CStepper

通过附加的 I2CStepper 板（带一台步进电机和四个继电器）扩展 Samovar 的功能。

该模块可以作为独立设备使用。既可以用 16x2 LCD 屏幕上的菜单和编码器单独控制，也可以连接到 Samovar 使用。

扩展器通过 I2C 总线用四根线连接到 Samovar（与 LCD 屏幕相同）。Samovar 能识别到扩展器已连接，并可以控制它。

Samovar 7.00: 固件使用第 3 版协议与扩展板通信。总线上最多可以有十块板，I2C 地址为 1 到 10：**奇数地址**（1、3、5...）的板是**搅拌器**，**偶数地址**（2、4、6...）的板是**泵**（连续泵送或灌装设定体积）。每块板会告诉 Samovar 自己能做什么：搅拌、泵送、灌装、继电器、外部传感器。Samovar 启动时搜索扩展板，之后在后台轮询；只要找到至少一块，每个模式的主页面上都会出现“I2CStepper”按钮，用于打开扩展板控制页面。曾在总线上响应过的板会一直保留在列表中，直到 Samovar 重启；如果与它的通信中断，它会被标记为不可用。

运行过程中（精馏程序、Lua、“啤酒”和“奶酪”模式），Samovar 使用一台**工作搅拌器**和一台**工作泵**——即找到的板中地址最小的那块。它们在过程开始时确定，直到过程结束都不会改变，即使运行中出现了地址更小的板也一样。

在“啤酒”模式下，第一个继电器用于开关麦汁转移泵（例如让麦汁在煮锅中循环）。其余继电器固件不使用——用途由您决定，可以从 Lua 中切换。继电器由工作搅拌器负责（如果它声明支持继电器），否则由工作泵负责。

扩展板本身的设置（模式、转速、搅拌器运行时间和暂停时间、泵流量、灌装体积和速度、每毫升步数）保存在 Nano 的存储器中，在**设置** → **I2CStepper** 选项卡（«Настройки → I2CStepper»）中编辑，见下文。“I2CStepper”页面只用于手动启动和停止。

“I2CStepper”页面

The screenshot displays the 'I2CStepper' control page. At the top, a dropdown menu (1) shows 'Насос, адрес 2'. Below it, a status section (2) indicates 'НАСОС, АДРЕС 2' is 'Состояние: Работает' (Working) with 'Сейчас: Скорость: 0 л/ч; осталось: 0 мл; направление: прямое.' (Now: Speed: 0 l/h; remaining: 0 ml; direction: forward). The control section (3) includes sliders for 'Скорость, л/ч' (Speed, l/h) and 'Направление' (Direction) set to 'прямое'. Below these are input fields for 'Объём, мл' (Volume, ml) and a text field for 'Пусто — качать без остановки; на ходу — докачать остаток.' (Empty — pump without stopping; on the way — top up the remainder). At the bottom, there are buttons for 'Применить скорость' (Apply speed) (4), 'Стоп' (Stop), 'Калибровка внешнего насоса' (External pump calibration), and four relay control buttons: 'Включить реле 1', 'Включить реле 2', 'Включить реле 3', and 'Включить реле 4'.

“I2CStepper”页面：已选择地址为 2 的泵板

1

设备 («Устройство»)

在总线上找到的扩展板列表；有两块或更多板时显示。搅拌器为奇数地址，泵为偶数地址。

2

扩展板状态

“运行中” («Работает») 或“已停止” («Остановлен»)；下方是当前泵速和剩余体积。

3

速度与体积

用于“启动” («Запустить») 按钮的泵速 (升/小时) 和体积 (毫升)。“停止” («Стоп») 使泵停止。

4

继电器 1

扩展板支持继电器时显示；切换扩展板的第一个继电器。

5

外部泵校准

仅用于泵板：打开所选板的“步/毫升” («Шагов/мл») 校准页面。

如果找到两块或更多扩展板，顶部会出现“设备”（«Устройство»）列表 — 用它选择要操作的板：“搅拌机，地址 N”（«Мешалка, адрес N»）或“泵，地址 N”（«Насос, адрес N»）。只有一块板时该列表隐藏。如果所选的板不可用，面板位置会显示消息“所选 I2CStepper 不可用”（«Выбранный I2CStepper недоступен»）— 不会自动换成另一块板，需要在列表中手动选择。

对于所选的板，页面显示其类型和地址、状态（“运行中”（«Работает»）或“已停止”（«Остановлен»））、以每秒步数表示的当前速度以及停止前剩余的步数。数据每两秒刷新一次。

控制。“速度”（«Скорость»）是以每秒步数表示的电机速度（1 到 18000），“目标，步”（«Цель, шагов»）是要走的步数。“启动”（«Запустить»）按钮把这两个值发送给扩展板，并按 Nano 存储器中保存的模式启动电机：搅拌机为旋转，泵为按给定步数灌装。对于泵，按体积计算步数更方便：把所需毫升数乘以 Nano 设置中的“步/毫升”（«Шагов/мл»）。“停止”（«Стоп»）使电机停止。如果泵未校准（“步/毫升”为零），或 Samovar 此时正在切换模式，则无法启动 — 此时会显示错误消息。

继电器。如果扩展板声明支持继电器，会显示“打开继电器 1”（«Включить реле 1»）/“关闭继电器 1”（«Выключить реле 1»）按钮 — 它切换扩展板的第一个继电器（在“啤酒”模式下是麦汁泵）。继电器 2-4 不能在此页面上切换，只能从 Lua 中切换。

外部泵校准。该按钮仅对支持灌装的泵板可用。它会打开所选板的校准页面：准备一个量具，设定电机速度，然后点击“开始校准”（«Начать калибровку»）— 泵开始泵送。正好收集到 100 mL 时，点击“记录 100 mL”（«Зафиксировать 100 мл»）：Samovar 计算出“步/毫升”，并把该值写入 Nano 存储器。更换驱动器微步设置、电机、泵头或管子后，请重新校准。

Samovar 设置

设置 → I2CStepper。“Nano I2CStepper 设置”（«Настройки Nano I2CStepper»）选项卡显示保存在扩展板自身存储器中的设置。“重新扫描 I2C 设备”

（«Пересканировать I2C-устройства»）按钮重新在总线上搜索扩展板（例如在连接新板之后）。如果有多块板，在“设备”列表中选择需要的那块。字段：

“新地址”（«Новый адрес»）— 扩展板在总线上的地址，1 到 10。奇数地址使板成为搅拌机，偶数地址使板成为泵；已被占用的地址不能选择。新地址在 Nano 重启后生效。

“模式”（«Режим»）— “搅拌机”（«Мешалка»，仅限奇数地址）、“泵”（«Насос»）或“灌装”（«Наполнение»，仅限偶数地址）。点击“启动”时以及收到来自 Lua 和应用的命令时，扩展板按此模式工作。

“步/毫升”— 泵的校准系数：用于把毫升和升/小时换算成电机步数。通过校准（见上文）或手动填写。该值为零时，泵无法启动。

“搅拌机转速”（«Обороты мешалки»）、“搅拌机运行时间，s”（«Работа мешалки, с»）、“搅拌机暂停时间，s”（«Пауза мешалки, с»）— 扩展板独立工作时的搅拌机参数；暂停时间为 0 时搅拌机连续运转。

“泵流量，mL/h”（«Подача насоса, мл/ч»）和“泵暂停时间，s”（«Пауза насоса, с»）— 用于“泵”模式（连续泵送）；“灌装体积，mL”（«Объём наполнения, мл»）和“灌装速度，mL/h”（«Скорость наполнения, мл/ч»）— 用于“灌装”模式。

“保存到 Nano”（«Сохранить в Nano»）按钮把这些字段写入扩展板的非易失性存储器；Samovar 和扩展板重启后数据都会保留。

设置 → 泵（«Настройки → Насос»）。如果找到 I2C 泵，该选项卡上会出现两个用于精馏的字段：

“在巴氏段上方使用 I2C 泵”（«Использовать I2C-насос над ЦП»）— 如果 Samovar 启动时检测到了该泵，它就参与精馏：取酒头时它成为主泵，取酒心和预液泛时它与本地泵协同工作。

“巴氏段上方取酒速度”（«Скорость отбора над ЦП»）— I2C 泵在酒心行 B 和预液泛行 C 上的恒定速度。在酒头行 H 上使用程序行本身的速度。

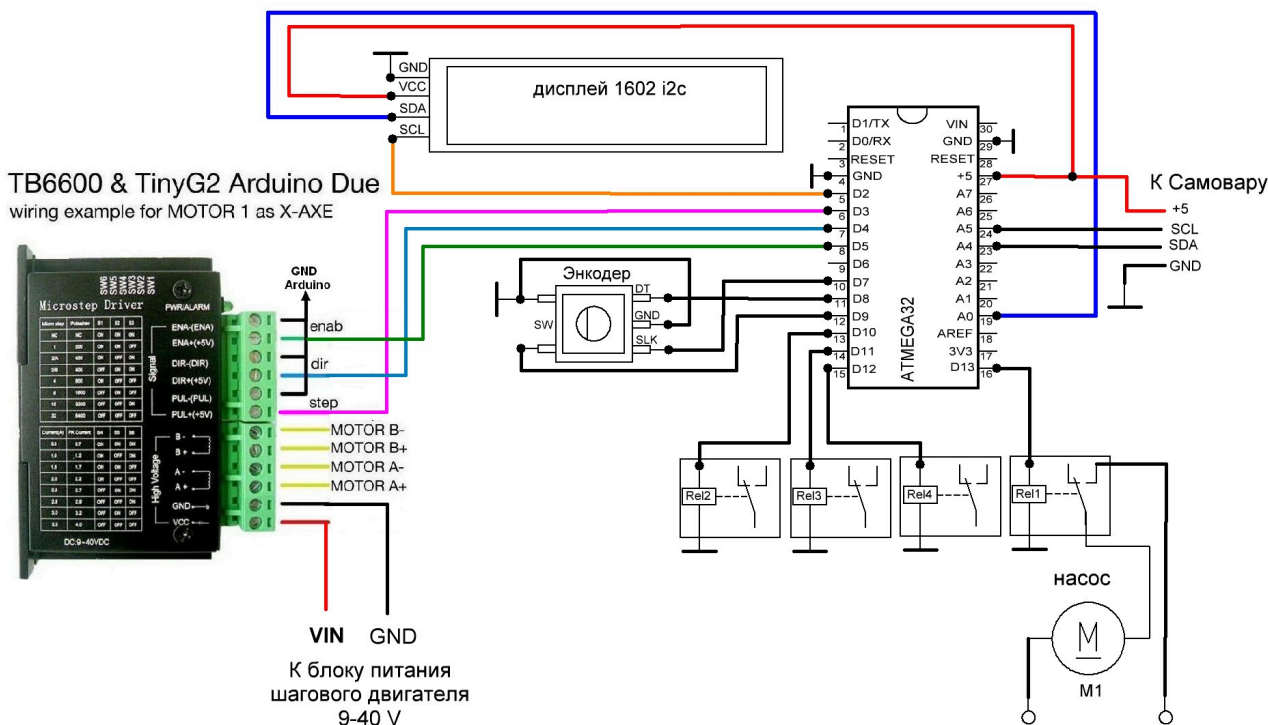
该泵以升/小时表示的速度按 Nano 存储器中的“步/毫升”换算成步数；Samovar 设置中已不再有单独的“I2C 泵步数”（«Количество шагов I2C насоса»）字段。

移动应用与网站

Samovar 应用和 samovar-tool.ru 网站上的 Samovar 控制页面每五秒接收一次扩展板列表及其状态（可用性、模式、速度、剩余量、继电器掩码），并可以启动和停止扩展板、切换继电器。Nano 设置和校准不能通过应用进行 — 只能通过 Samovar 网页界面。如果命令被拒绝，原因会以“Blynk V37 a=N: code”（«Blynk V37 a=N: код»）的形式出现在 Samovar 消息中。

电路图与接线

接线图



外设连接引脚：

// Pins for I2C Master - connecting the screen and potentially other devices

```
// SDA_PIN 0 //A0
// SCL_PIN 2 //D2

// Pins for I2C Slave - communication with Samovar
// SDA_PIN A4
// SCL_PIN A5

// Pins for the stepper motor
#define STEPPER_STEP 3
#define STEPPER_DIR 4
#define STEPPER_EN 5

//Pins for the relays
#define MIXER_PUMP_PIN 13 // RELE_PIN1
#define RELE_PIN2 10 // RELE_PIN2
#define RELE_PIN3 11 // RELE_PIN3
#define RELE_PIN4 12 // RELE_PIN4

// Pins for the encoder
#define ENC_CLK 7 //S2
#define ENC_DT 8 //S1
#define ENC_SW 9 //KEY
```

提示。

如果打算把扩展器做成装在独立外壳里的单独设备，用 USB A-B 线（电脑 - 打印机线）把它连接到 Samovar 是合理的选择。模块电源（+/- 5 伏）和两路 I2C 总线信号直接从 Samovar 的显示屏处引出。Samovar 外壳上安装 USB-B 插座。

扩展器外壳上安装 USB-A 插座。

这样，连接到 Samovar 时，Arduino Nano 同时获得电源和 I2C 总线。作为独立设备使用时，用单独的电源代替 Samovar，例如手机充电器。

使用内置电源时，在电源输出端装一个 USB-B 插座，用同一根 USB A-B 线供电，这样就排除了误把 +5 伏送入 Samovar 的可能。

外设。

继电器模块使用 5 伏，最好带光耦控制。也可以使用 12 伏继电器，接法与 Samovar 中相同。

可以使用任何泵：直流电机，或者交流单相、三相电机。泵电机的启停先经过继电器，再经过任意功率开关：接触器或固态继电器。小流量时可以考虑 15RM 磁力驱动泵。不锈钢材质，安静，流量好。

搅拌器或 NBK 可以使用任何双极步进电机，只要功率和扭矩足够即可。推荐使用 Nema 23 相匹配的驱动器。

控制步进电机和四个继电器的 Arduino Nano 最新草图（sketch）可以通过[此链接](#)下载。需要使用协议 v3 的版本；从协议 v2 升级时，扩展板存储器会重新初始化 — 请重新检查地址、模式和“步/毫升”。它的主要用途是作为扩展模块与 Samovar 配合使用。必须使用 Samovar 附带的库。

从 Lua 控制

当固件带 Lua 编译时，以下函数可用（见[关于 Lua 脚本语言](#)）。搅拌器函数作用于工作搅拌器，泵函数作用于工作泵（即地址最小的板，见上文）。所有启动函数在命令被扩展板接受时返回 1，否则返回 0 — 包括总线上没有所需扩展板的情况。

set_stepper_by_time(spd, direction, time) – 启动搅拌器：spd – 以每秒步数表示的速度，direction – 方向 0 或 1，time – 以秒为单位的运行时间。time = 0 时搅拌器连续旋转直到收到停止命令，spd = 0 时搅拌器停止。

set_stepper_target(spd, direction, target) – 以灌装模式启动泵：spd – 以每秒步数表示的速度（最高 18000），direction – 方向 0 或 1，target – 步数。spd = 0 或 target = 0 时泵停止。该函数不控制 Samovar 的标准步进电机。

get_stepper_status() – 泵还要走多少步（0 – 已停止或没有扩展板）。

i2cpump_start(rate, volume) – 以 rate L/h 的速度泵送 volume mL，按 Nano 存储器中的“步/毫升”换算成步数。如果泵板未连接、未校准或参数无效，返回 0。

i2cpump_stop() – 停止泵。

i2cpump_get_speed() – 以每秒步数表示的当前泵速。

i2cpump_get_target_ml() – 最近一次 i2cpump_start 命令设定的体积，mL。

i2cpump_get_remaining_ml() – 还剩多少要泵送，mL。

i2cpump_get_running() – 泵正在运行时为 1，否则为 0。

set_mixer_pump_target(Val) – 打开（1）或关闭（0）第一个继电器 – 麦汁泵。

get_mixer_pump_status() – 第一个继电器的状态：0 或 1；没有带继电器的扩展板时为 255。

set_i2c_rele_state(Num, Val) – 打开（1）或关闭（0）继电器 Num（1–4）。命令执行成功时返回 1。

get_i2c_rele_state(Num) – 继电器 Num（1–4）的状态：0 或 1；没有带继电器的扩展板时为 255。

check_i2c_device(Addr) – 检查地址为 Addr 的设备是否在 I2C 总线上响应；找到设备时返回其地址。

注意。继电器由工作搅拌器负责（如果它声明支持继电器）；否则由工作泵负责。改变状态的函数在 Lua 模拟模式下以及 Samovar 切换模式期间不会执行。

示例：set_stepper_by_time(800, 0, 0) – 搅拌器以每秒 800 步连续旋转；set_stepper_by_time(0, 0, 0) – 停止搅拌器。i2cpump_start(6, 250) – 以 6 L/h 泵送 250 mL；当 i2cpump_get_running() == 1 时，泵正在运行。

添加物投料器（啤酒和奶酪）

在“啤酒”（«Пиво»）和“奶酪制作”（«Сыроварение»）模式下，需要随程序进行添加各种添加物：啤酒中加啤酒花（通常每次煮沸加 2-4 次），奶酪中加发酵剂、凝乳酶、氯化钙、盐和香料。Samovar 可以自己完成投料，也可以在合适的时刻提醒您。下面介绍四种内置的投料器实现方式，从最简单到最灵活，说明每种需要什么、在程序中如何设置；随后概述世界各地酿酒爱好者使用的自制结构，以及如何把它们接入 Samovar。程序的一般说明见[糖化程序](#)和[奶酪制作程序](#)两节；扩展板见[J2CStepper](#)页面；脚本见[Lua](#)一节。

方式 1. 按 Samovar 信号手动添加



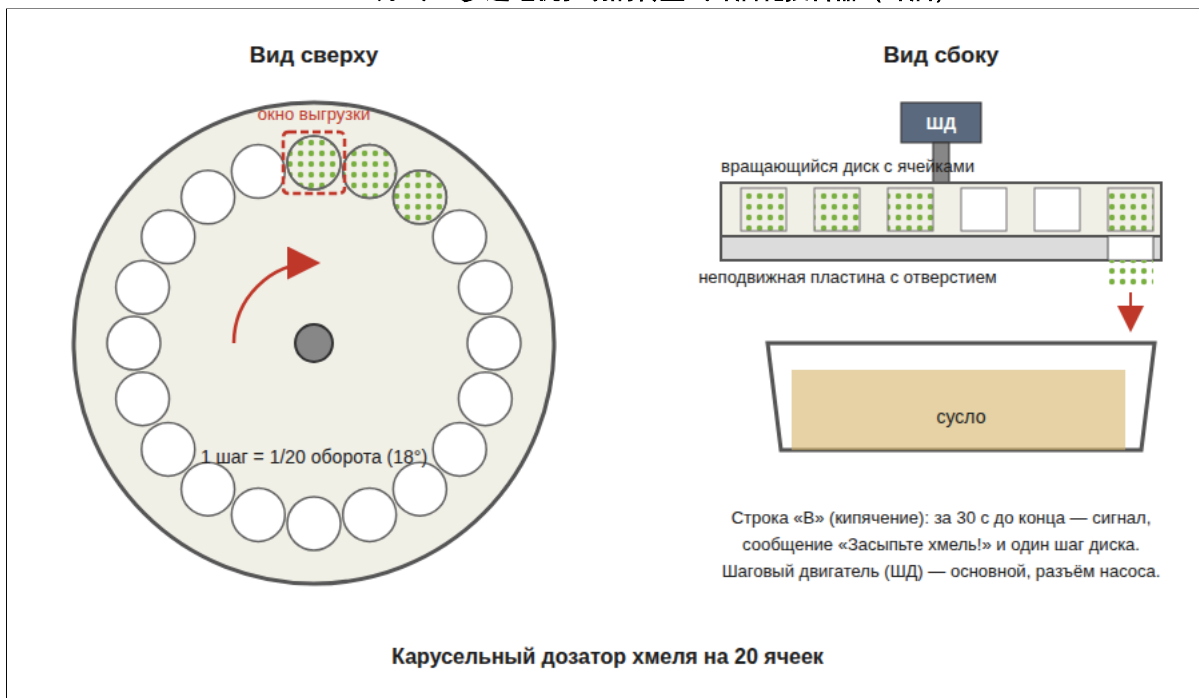
不需要任何硬件。到了合适的时刻，Samovar 会让蜂鸣器响起，在显示屏和网页界面上显示消息，并向移动应用和网站发送推送通知。添加物由您亲手加入。

啤酒。在每个“B”行（煮沸）结束前 30 秒，Samovar 发送消息“请加啤酒花！”（«Засыпьте хмель!»）。“B”行的数量等于投放啤酒花的次数：例如，煮沸 60 分钟、在结束前 60、30 和 5 分钟投放啤酒花，就写成 30、25 和 5 分钟三个“B”行。程序不等待确认，直接继续执行。

奶酪。使用“W”行（手动操作），操作选择“加入发酵剂”（«Внести закваску»）、“加入酶”（«Внести фермент»）或“加入添加物”（«Внести добавку»）：Samovar 暂停程序（如设置了搅拌机，搅拌机继续运转），等待在模式页面、应用或网站上按下“确认”（«Подтвердить»）按钮。方法为“手动”（«Вручную»）的“D”行（投料）作用相同，另外还会记录成分和体积。

优点：无需制作任何东西。缺点：人必须在附近；在啤酒中，投料的准确时间取决于您。

方式 2. 步进电机驱动的转盘式啤酒花投料器（啤酒）



这是“啤酒”模式内置的干料自动投料器。在发送“请加啤酒花！”消息的同时，Samovar 让主步进电机（在其他模式下驱动取酒泵的那台）精确转动 1/20 圈（18°）。电机轴上装有一个带格子的圆盘，圆盘下方是一块固定板，板上只在锅的上方开有一个孔。每转动一次，下一个格子就对准这个孔，格子内的物料落入麦汁中。

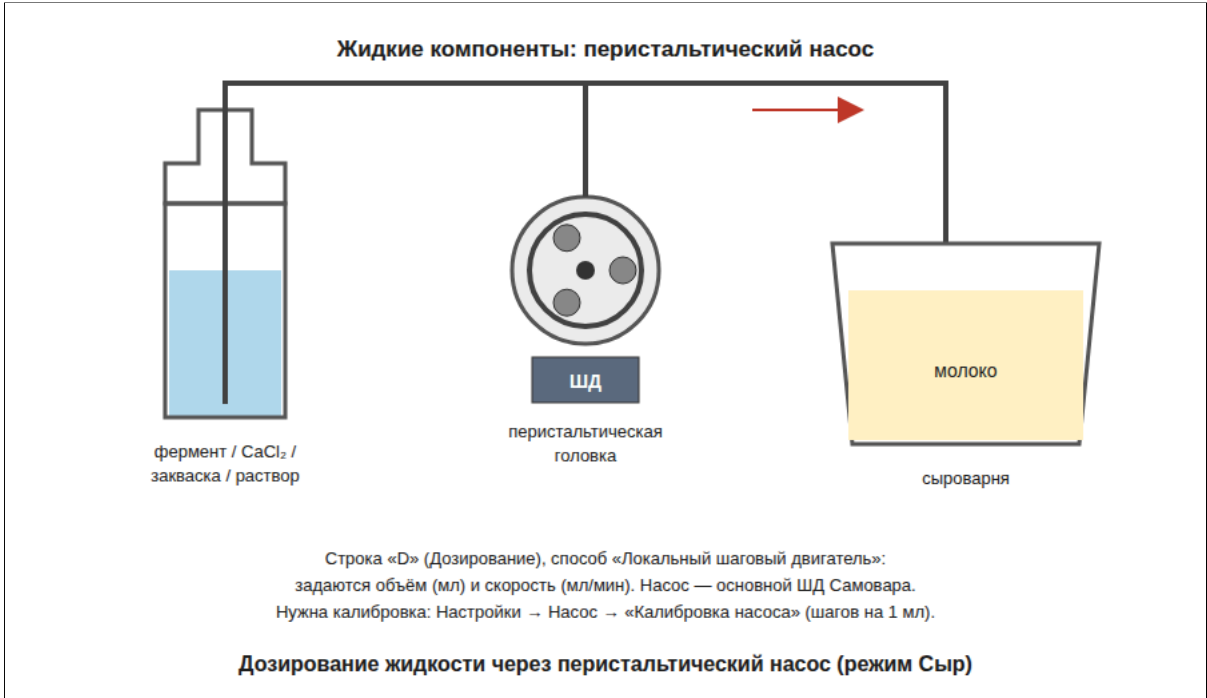
需要什么。一台 NEMA17 步进电机（接到 Samovar 的泵接口上，不需要单独的驱动器和设置），一个直径 150-200 mm、用食品级塑料或不锈钢制成、圆周上有 20 个格子（或更少，见下文）的圆盘，一块只开一个孔的底板，以及一个盖子，以免沸腾麦汁的蒸汽把啤酒花弄潮。圆盘用 3D 打印制作很方便。啤酒花按旋转顺序放入格子，从紧挨出料口的那一格开始。

在程序中如何设置。无需额外设置：每个“B”行就是一次投料。转动发生在行结束时，所以第一份（例如“60 分钟时的啤酒花”）在第一个“B”行结束时落下，这一行可以设得很短，例如 1 分钟。驱动器的微步模式必须为 1/16，与泵相同。

如果格子不是 20 个。每步的转角在固件的 HopStepperStep() 函数（文件 beer.h）中设定： $\text{TargetSteps} = 360 / 1.8 * 16 / 20$ ，其中 20 是整圈的格子数。对于 10 格圆盘，把 20 改为 10 并重新刷写 Samovar。也可以不用圆盘，而在水平轴上做一个带纵向隔仓的滚筒，计算方法相同。

优点：酿造全自动，除标准配置外不需要其他电子元件。缺点：只能投放干料；“奶酪”模式下不执行此步进（那里电机被泵占用）。

方式 3. 用于液体的蠕动泵（奶酪）



在“奶酪制作”模式下，Samovar 的主步进电机作为计量泵工作。方法为“本地步进电机”（«Локальный шаговый двигатель»）的“D”程序行（投料）按设定的速度（ml/min）输送设定体积（毫升）的液体：例如 1 分钟内输送 25 ml 氯化钙溶液，或 2 分钟内输送 50 ml 稀释的凝乳酶。输送结束后，程序自动转到下一行；如果在“超时，分钟”（«Тайм-аут, мин»）时间内未输送完设定体积，奶酪制作会停止并发出消息。

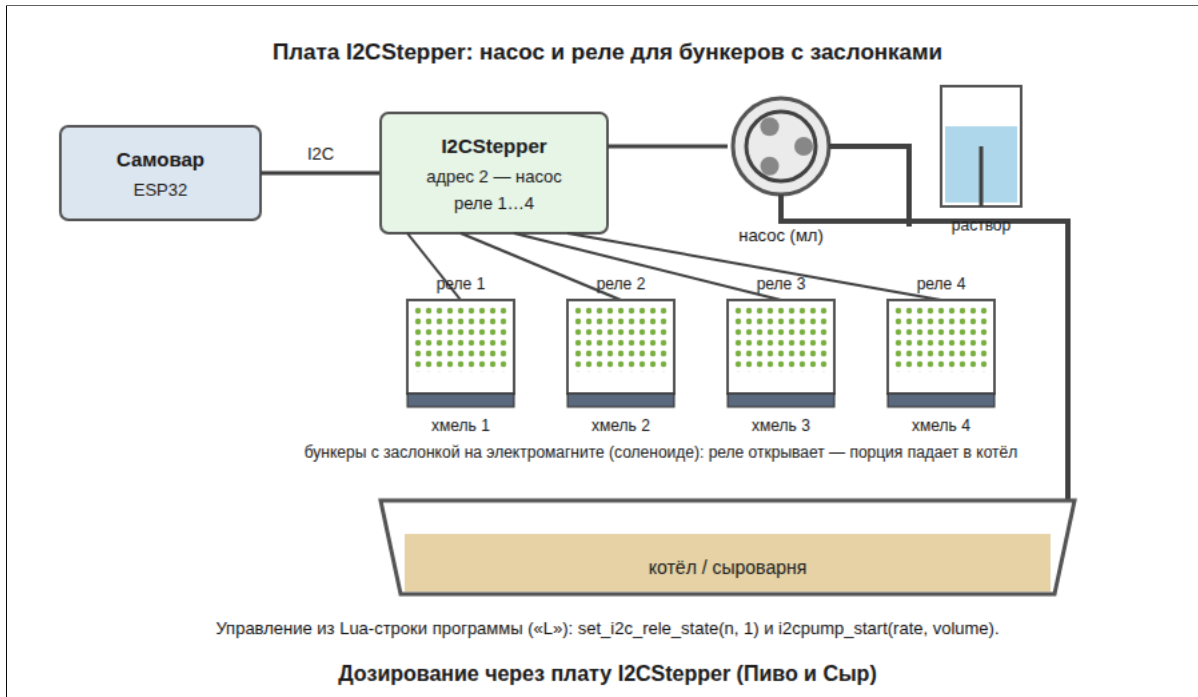
需要什么。一个适配 NEMA17 步进电机的蠕动泵头（与精馏取酒所用的相同）和食品级硅胶管。管子的进口端放入装溶液的容器，出口端引到奶酪锅中、高于牛奶液面。溶液要提前配好；发酵剂和凝乳酶用凉开水稀释，使体积足够大，以便泵能准确输送。

校准。体积按“泵步数”（«Количество шагов насоса»）值（每 1 ml 的步数）换算成步数。该值在“设置 → 泵”选项卡中用“泵校准”（«Калибровка насоса»）按钮设定（见[这里](#)）：泵把液体泵入量杯，您输入实际体积，Samovar 保存系数。校准时请使用与投料时相同的管子和相同的液体，并预先充满管子，否则第一份液体会用于充管。

多种成分。一台泵对应一种溶液。如果一次制作中需要加入两三种不同的溶液，可在“D”行之间插入“加入添加物”（«Внести добавку»）的“W”行，并把进液管移到另一个容器（先冲洗），或者使用 I2CStepper 板上的第二台泵（方式 4）。

优点：体积和时间准确，无需手工操作。缺点：只能投放液体；一台泵只能投一种成分。

方式 4. I2CStepper 板：由 Lua 脚本控制泵和继电器（啤酒和奶酪）



I2CStepper 板为 Samovar 增加第二台步进电机（地址 2 — 泵）和四个继电器。固件本身不会用它们投料，但所有功能都可以通过“L”程序行启动的 Lua 脚本调用：在该行中选择一个 .lua 文件并传给它参数（例如继电器编号和体积），这样一个脚本就能完成所有投料。脚本打开泵或继电器，等待完成，然后调用 `setNextProgram()` 转到下一行。

液体 — I2C 泵。函数 `i2cpump_start(rate, volume)` 以 `rate l/h` 的速度泵送 `volume ml`；每 1 ml 的步数取自“I2C 泵步数”（«Количество шагов I2C насоса»）字段（设置 → 泵），校准在“I2CStepper”页面上进行。示例脚本 `dose_pump.lua` 以 6 l/h 的速度加入该行第一个参数所给的体积（在程序中：文件 `dose_pump.lua`，参数 20）：

```
-- dose_pump.lua : 用 I2CStepper 泵加入 arg[1] ml 溶液
if i2cpump_start(6, tonumber(arg[1])) == 1 then
    while i2cpump_get_running() == 1 do delay(500) end
    sendMsg("Solution added", 2)
else
    sendMsg("I2CStepper pump is not responding", 1)
end
setNextProgram()
```

干料 — 继电器和带闸门的料斗。为每一份啤酒花（或盐、香料）在锅上方做一个小料斗，料斗底部由 12 V 电磁铁（螺线管）驱动的闸门封住。板上的继电器给电磁铁通电，闸门打开，这一份物料落入锅中。四个继电器，四份物料；每份在装料时预先称好。示例脚本 `dose_rele.lua` 把编号在该行参数中给出的继电器（在程序中：文件 `dose_rele.lua`，参数 3）上的料斗闸门打开一秒半：

```
-- dose_rele.lua : 从继电器 arg[1] 上的料斗投下一份
set_i2c_rele_state(tonumber(arg[1]), 1)
delay(1500)
set_i2c_rele_state(tonumber(arg[1]), 0)
sendMsg("Hops added", 2)
setNextProgram()
```

每次投料在程序中各占一个“L”行，文件相同，参数各自不同（继电器编号或体积）；该行的超时以秒为单位设定。

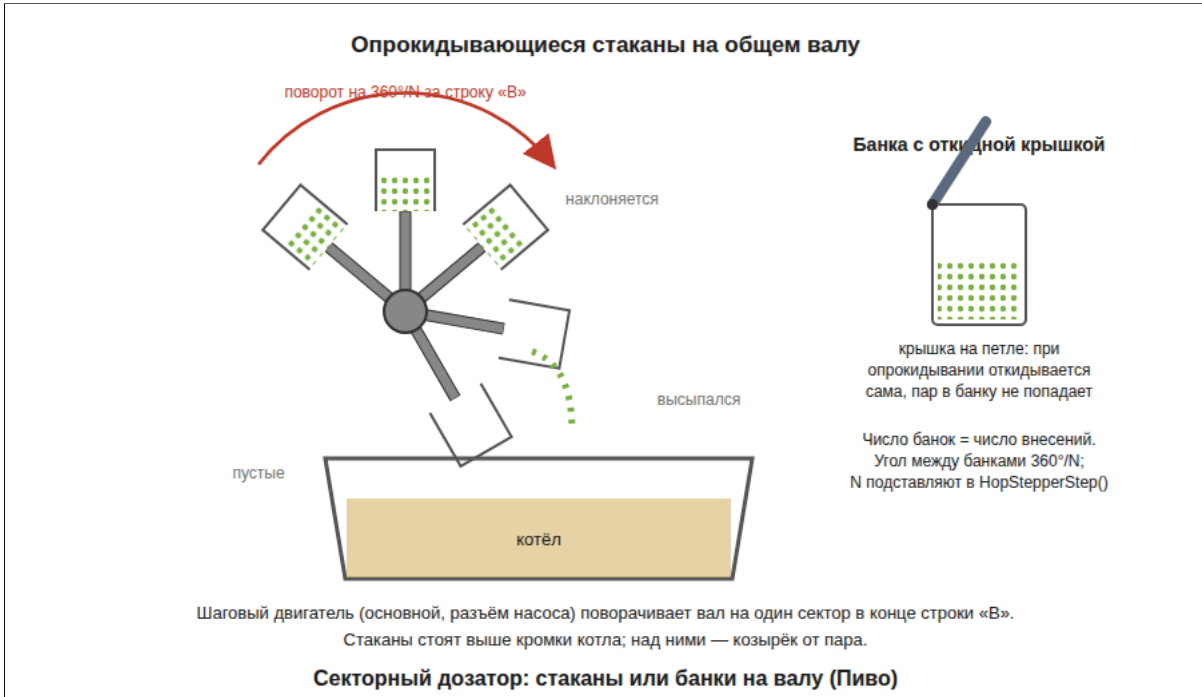
哪些继电器空闲。在“啤酒”模式下，板上第一个继电器被麦汁转移泵占用（见“I2CStepper”页面）；其余三个固件不使用。在“奶酪制作”模式下，固件不占用板上的继电器（搅拌器和排放阀接在 Samovar 自身的 2 号和 4 号继电器上）。可以用 `check_i2c_device(2)` 函数检查板是否响应。

“啤酒”模式的特点。“L”行是一个独立的程序行，在该行上不会自动维持沸腾，所以要把它设得很短（几秒钟）；下一个“B”行会重新检测沸腾 — 此时麦汁仍在沸腾，所以很快。如果只需要按时间表投放啤酒花，方式 2 更简单：既不需要扩展板，也不需要脚本。

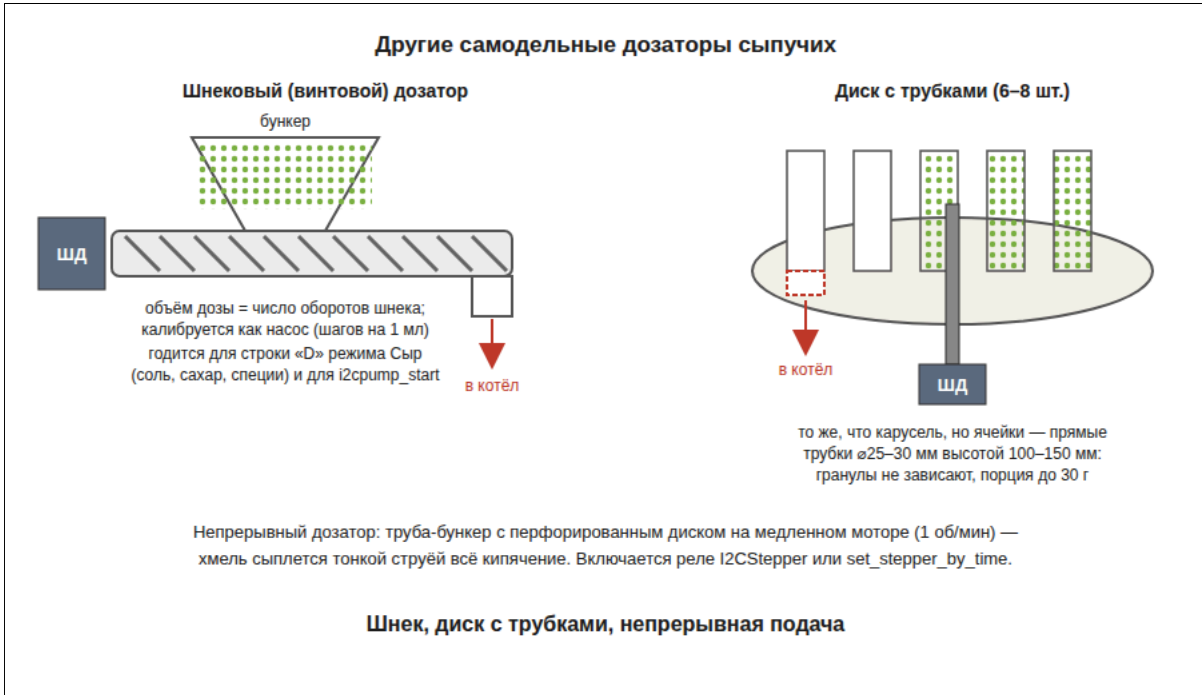
优点：成分数量不限，液体和干料均可，逻辑完全自由。缺点：需要 I2CStepper 板、支持 Lua 的固件和脚本。

方式 5. 自制结构：酿酒爱好者的其他做法

商用自动酿酒设备用不同方法解决同一个问题：Brewie 和 PicoBrew Zymatic 把啤酒花放在四个篮子里，在合适的时刻让麦汁流过下一个篮子；MiniBrew 在盖子上装有一个有六个圆筒的转盘；Grainfather 和 Braumeister 不投放啤酒花，只发送通知（与方式 1 相同）。在家庭作坊中（Homebrew Talk 论坛、CraftBeerPi 和 BrewTroller 项目、Brew Your Own 杂志文章），下面这些结构最常见。它们都可以用前面已介绍的某种方式接入 Samovar：在“B”行上让主步进电机走一步，奶酪中的“D”行，或者由 Lua 控制 I2CStepper 板的继电器或电机。



扇区式投料器：轴上的杯子或小罐。在步进电机的水平轴上按扇形、每隔 $360^\circ/N$ 牢固地安装 N 个杯子（塑料杯、截短的瓶子、带翻盖的小罐）。杯子位于圆周上半部时开口朝上；转到下一个扇区时，下一个杯子越过水平位置，其中的物料落入锅中。带翻盖的小罐比敞口杯更方便：倾倒之前，盖子保护啤酒花免受蒸汽影响，倾倒时盖子靠自重打开。优点是简单：不需要精确的格子，也不需要底板。用于 Samovar：电机接到泵接口上，在 `HopStepperStep()` 中用杯子数 N 代替 20（4 个杯子时每步 90° ），每个“B”行倾倒一个杯子。轴必须高于锅沿，并尽量偏离蒸汽柱；在杯子上方加一块挡板。



带管子的圆盘（“Lazy Susan”）。Homebrew Talk 上最常见的自制结构：直径 150 mm 的圆盘上装 6-8 根竖管（ $\varnothing 25\text{--}30$ mm，高 100-150 mm，可装约 30 g 颗粒），下方是只开一个孔的固定板；驱动用大扭矩舵机或廉价的 28BYJ-48 步进电机配 ULN2003 驱动器，为了精确停止，在凸轮上装微动开关或光电传感器。它与方式 2 中的转盘相同，只是用高的直管代替小格子。作者们的经验：颗粒会卡在锥形底的管子里，所以管壁必须笔直光滑；粉状和潮湿的啤酒花倒不出来，需要干燥的颗粒；锅上方必须有挡板或蒸汽出口，否则煮沸 20-30 分钟后颗粒会受潮结块。用于 Samovar 时按方式 2 使用，在 `HopStepperStep()` 中填入管子数；如有需要，可以把圆盘装到 `I2CStepper` 板的电机上，由 Lua 用 `set_stepper_by_time` 函数转动。

螺旋（蛟龙）投料器。一种来自鱼和猫自动喂食器的结构（见 Instructables 和 GitHub 上的项目）：水平管上方是料斗，管内是装在步进电机轴上的螺旋；投料量由转数决定。零件用 3D 打印，螺旋直径 20-30 mm。它可以从一个料斗按任意份量投放任何松散物料：盐、糖、香料、啤酒花颗粒。对 Samovar 来说，这种投料器就是同一台“泵”：在“奶酪”模式下接到泵接口上，用“泵校准”（«Калибровка насоса»）按钮校准（把物料倒入量杯，输入其体积 ml），并用方法为“本地步进电机”（«Локальный шаговый двигатель»）的“D”行投料。第二台相同的投料器可以挂在 `I2CStepper` 板上（地址 2），由 Lua 用 `i2cpump_start(rate, volume)` 函数启动。缺点：一个料斗只能装一种物料；3-4 个不同的啤酒花品种需要 3-4 个螺旋或一个转盘。

连续投放啤酒花。对于采用“连续投花”（在整个煮沸过程中以细流加入啤酒花）的酒款，Brew Your Own 介绍了一种投料器：一段 $\varnothing 100$ mm、长 60 cm 的管子，下端有一个由 1 rpm 电机带动旋转的多孔圆盘；颗粒平均以每 10 分钟约 8 g 的速度落下，圆盘上的销钉防止颗粒架桥。用于 Samovar 时，电机在整个“B”行期间由 `I2CStepper` 板的继电器（`set_i2c_rele_state`）接通，或由板上的步进电机（`set_stepper_by_time`）转动。

舵机或减速电机驱动的闸门。在 CraftBeerPi 项目（HopDropper 插件）中，带隔断的托盘由慢速 12 V 减速电机通过一对齿轮带动，控制器只在设定时间内供电；另一种流行的方案是把杯子装在舵机摆臂上，挂在锅壁上远离蒸汽的地方。舵机接到 Samovar 的 `SERVO_PIN` 输出（精馏中切换容器的同一个输出），由 Lua 用 `setServoAngle(angle)` 函数转动：例如 `setServoAngle(90)` 打开闸门，`delay(1500)`，`setServoAngle(0)` 关闭。大扭矩舵机请用单独的 5-6 V 电源供电，不

要从开发板取电。12 V 减速电机机构由 I2CStepper 板的继电器接通 1-3 秒（转到限位或转到带微动开关的凸轮所需的时间）。控制方法同方式 4。舵机摆臂上装杯子的示例脚本 tip_cup.lua（在程序中是使用此文件的“L”行）：

```
-- tip_cup.lua：倾倒舵机摆臂上的杯子
setServoAngle(150)
delay(1500)
setServoAngle(0)
sendMsg("Hops added", 2)
setNextProgram()
```

用麦汁冲洗的篮子。 Brewie/Zymatic 的原理：啤酒花放在锅外的网篮里，麦汁转移泵在合适的时刻把液流引过下一个篮子。在家里这是最复杂的方式（需要泵、接在 I2CStepper 继电器上的若干电磁阀和管路），但啤酒花一定不会受蒸汽影响，也不需要往锅上方倾倒任何东西。用于 Samovar 时，阀门由 Lua 用 set_i2c_rele_state 函数切换，泵由板上第一个继电器控制（在“啤酒”模式下它本来就控制麦汁转移泵）。

从别人的设计中总结的经验。

任何干料投料器的头号敌人都是蒸汽：在锅上方加挡板，把投料器移到一侧，最好在投料前一直用盖子盖住每一份物料（带翻盖的小罐、闸门）。

颗粒不容易通过收窄处和锥形部分：格子、管子和托盘要做成直壁、平底，出料口不小于格子的截面。

在装料时预先称好每一份，让自动装置只负责选择时刻；这样精度就不取决于机械结构。

酿造前先在测试模式下空载运行投料器，第一次带投料器酿造时人要守在旁边：“B”行无论如何都会发送通知，必要时可以手动加入。

创意来源：

[Lazy Susan 啤酒花投料器 \(Homebrew Talk\)](#)、[自动颗粒啤酒花投料器 \(Homebrew Talk\)](#)、[3D 打印的啤酒花转盘 \(Homebrew Talk\)](#)、[自动投放啤酒花 \(Homebrew Talk\)](#)、[CraftBeerPi 的 HopDropper \(GitHub\)](#)、[连续投花器 \(Brew Your Own\)](#)、[螺旋微量投料器 \(Instructables\)](#)、[MiniBrew Hop Carousel](#)、[自动酿酒设备概览 \(Brew Your Own\)](#)。

如何选择

啤酒，投 2-4 次啤酒花 — 方式 2（内置电机驱动的转盘），或方式 5 中它的简化版：轴上的杯子或小罐。投料器做好之前用方式 1：“B”行本来就会发出信号。

奶酪，发酵剂 / 凝乳酶 / CaCl_2 — 方式 3（“D”行，内置泵）。一台泵，每次制作投一种成分；其余用“W”行手工加入。盐和干香料：方式 5 中的螺旋投料器，接在同一个接口上。

自动投放多种液体和干料成分 — 方式 4（I2CStepper 和 Lua）。

通用建议：添加物要加盖存放（蒸汽和冷凝水是干料投料器的大敌），事先用水并在测试模式下检验投料器，液体体积不少于 10-20 ml，这样泵的误差就不明显。

高压灭菌锅控制

使用 Samovar 控制高压灭菌锅的指南。

工作温度。

高压灭菌锅的工作温度在 **110-120 度** 范围内。差别看起来不大，但对某些食品至关重要。蔬菜在 110 度就可以制作罐头。鱼需要 115 度，肉则需要整整 120 度。

实际上，Samovar 控制高压灭菌锅只做一件事：通过调压器控制加热，从而控制锅内温度。

可能遇到的问题是选择质量好的 DS18B20 传感器。

按规格书，这种传感器可工作到 125°C，但实际上远不是每个传感器在 120°C 下都能正常工作。可能出现这种情况：传感器在蒸馏和精馏中工作得很好，用在高压灭菌锅上却会失灵。

实用建议。

建议自己制作传感器：在知名商店里从评价好的供应商处买几颗 DS18B20 芯片。自己做出能耐受 120 度工作的优质传感器，几率比买成品高得多。成品传感器在高温下经常失灵。制作时，重要的是保证传感器与套管表面之间有导热介质。为此用中性耐热密封胶比较方便。不要使用酸性密封胶：它会腐蚀导线。也可以用热熔导热胶，但如果传感器出了问题，导热胶要难清除得多。

重要：放置传感器的高压灭菌锅套管内部也必须填充导热介质。用甘油最方便。

要控制高压灭菌锅，请切换到“啤酒”（«Пиво»）模式。

1. 检查：

设置（«Настройки»）/ 温度（«Температура»）/ “蒸馏釜传感器地址”（«Адрес датчика в кубе»）。确认这里设置的是正确传感器的地址。

2. 设置 / 啤酒（«Пиво»）。

把“酿造方式”（«Варочный порядок»）保留为 All-in-one / BIAB：在这种方式下，加热按蒸馏釜传感器调节。HERMS 和 RIMS 按水传感器和塔节传感器控制温度，不适用于高压灭菌锅。

“保温模式下调压器的电压/功率：___ v”（«Напр./мощность регулятора в режиме поддержания темп.: ___ v»）。这个参数需要调试，根据您的设备选取数值。它是关闭加速加热后，足以让高压灭菌锅温度继续上升的最小电压/功率。默认情况下，温度比程序设定值低 1 度时关闭加速加热（可在文件 Samovar_ini.h 中修改，即这一行 [#define HEAT_DELTA 1](#)）。在这最后一度里，高压灭菌锅应以此字段设置的电压/功率缓慢继续升温。开始时可以设为 100 伏。

3. 程序。

使用麦芽糖化程序。在这里设置温度和保持该温度的时间。

这些数值可能根据所制作的食品略有变化。经典做法是 120 度保持一小时。设置“暂停”（«ПАУЗА»）方式，温度 120 度，时间 60 分钟。

实践表明，如果不是用加速加热尽快升到 120 度，而是逐步升温，大约用半小时，例如按 105、110 和 115 度分“台阶”，每级保持 3 - 5 分钟，罐子损坏和食品溢出的几率会更低。

得到的程序如下：

```
P;105;4;0^0^0^0;0
```

```
P;110;4;0^0^0^0;0
```

```
P;115;4;0^0^0^0;0
```

```
P;120;60;0^0^0^0;0
```

行的最后一个字段是用于保持温度的传感器编号：0 为蒸馏釜传感器。设备字段 0^0^0^0 表示该行不使用搅拌器和水泵。

安全。

为保证安全，可以采用以下方法：

- 在高压灭菌锅的安全阀附近安装一个 55-70° 的双金属温控开关 KSD9700（NO - 常开）。

把温控开关的输出端与 Samovar 的紧急按钮并联（向 ALARM_BTN_PIN 输入逻辑 1：DEVKIT 开发板为 D35，ESP32-S3 为 D48；固件必须带紧急按钮编译，即 Samovar_alarm_button 环境或 [#define USE_ALARM_BTN](#)）。蒸汽泄放时，Samovar 会通知调压器关闭加热。

- 把压力表换成（或自行改装成）电接点压力表，设定最高压力 0.25 MPa，并用它的触点与 Samovar 的紧急按钮并联。

- 在安全阀上接一个通过接触器或漏电保护器提供[附加保护](#)的装置。如果蒸汽冲出，Samovar 会断电，加热也随之关闭。

设置完成，点击“开启加热”（«Включить нагрев»）按钮，享受自动运行过程直到结束。

过程结束时，Samovar 会通知调压器关闭加热。等高压灭菌锅冷却后 取出成品。

注意。

高压灭菌锅内的温度和压力直接相关。加热越强，压力越高。因此，可以根据工作压力计算和检查温度工况，反之亦然。

例如，要让高压灭菌锅达到 120 度，需要有表压（超过大气压的压力）。

网上有温度与压力的对照表。考虑到您所在地的修正，完全可以使用这些表。

关于 Lua 脚本语言。

Samovar 支持基于脚本编程语言 [Lua](#) 的自动化。Lua 支持在编译固件时启用：在配置器中勾选“使用 Lua”（«Использовать Lua»）复选框（位于“硬件”（«Оборудование»）部分），在 PlatformIO 中选择 **Samovar_lua** 环境，手动编译时在 `user_config_override.h` 文件中加入 `#define USE_LUA` 一行。脚本编辑器通过设置页面上的“编辑器”（«Редактор»）按钮打开（地址 <http://samovar.local/edit>）。不需要输入用户名和密码。

当前 Lua 版本为 5.4.4

Samovar 启动时会运行 `init.lua` 文件。它的作用是开启或关闭每秒一次的脚本执行循环，并初始化其他脚本要用到的变量。出厂的 `init.lua` 关闭了该循环：`setNumVariable("loop_lua_fl",0)`。

如果开启了循环运行脚本，每秒会运行两个脚本：先运行 `script.lua`，再运行与工作模式对应的脚本：`rectificat.lua`（精馏）、`dist.lua`（蒸馏）、`beer.lua`（啤酒）、`bk.lua`（醪塔）、`nbk.lua`（NBK）、`suvid.lua`（低温慢煮）、`cheese.lua`（奶酪制作）。在“Lua 模式”（«Lua-режим»）下（设置中模式列表里的一项，只在带 Lua 的固件中可见）没有模式脚本——全部模式逻辑都写在 `script.lua` 中。打开 <http://samovar.local/lua> 可以单次运行这一对脚本，打开 http://samovar.local/lua?script=file_name 可以运行单独的文件。

脚本在同一个任务中依次运行：一个脚本运行时，另一个等待。单次脚本运行不能超过 20 秒——超时后会被中止，并显示消息“Lua: chunk execution aborted by timeout”（«Lua: выполнение чанка прервано по таймауту»）。如果模式脚本连续 5 次出错结束，每秒循环会停止（`loop_lua_fl` 被重置为 0），并在控制台输出一条汇总消息；`script.lua` 连续出错 5 次后将不再运行，直到重新加载脚本，但循环不会停止。脚本错误会输出到浏览器控制台和串口监视器。

还可以在界面中添加运行文件的按钮。文件名为 `btn_<mode>_button<N>.lua`，其中 `mode` 是 `rect`、`dist`、`beer`、`bk`、`nbk`、`suvid` 或 `cheese`，`N` 是按钮编号。例如 `btn_rect_button1.lua`、`btn_beer_button2.lua`。按钮显示在其所属模式主页面的“附加”（«Дополнительно»）选项卡上的 Lua 区块中；那里还有用于执行一两行代码的“Lua.”（«Lua.»）输入框和“执行 Lua”（«Выполнить Lua»）按钮，以及“Lua 状态”（«Статус Lua»）——由脚本通过 `setLuaStatus` 函数设置的一行文字。

按钮名称在文件第一行用 `--|名称^` 形式的注释设置（例如 `--|Start^`）。如果没有这一行，按钮按找到文件的顺序命名为 `LUA1`、`LUA2` 等。

出厂时每个模式有两个按钮：“开始”（«Начать»）（`btn_*_button1.lua`——开启每秒循环）和“停止”（«Остановить»）（`btn_*_button2.lua`——关闭循环）。按钮的主要用途是启动循环运行脚本（例如设置一个变量的值，循环中运行的脚本随后可以读取它），或者开关某个执行设备（例如水泵）。

因此，按钮的使用架构如下（以蒸馏模式为例）：

- 主受控循环：每秒一次 `script.lua => dist.lua => script.lua => dist.lua =>...`（“受控”是指可以在任何脚本中通过全局变量以编程方式启动或停止它）
- `script.lua` 脚本对所有模式通用，可以在其中放置任何模式下都要运行的代码（精馏、啤酒、低温慢煮等）
- `dist.lua` 脚本只在蒸馏模式下使用，可以在其中放置每秒都要执行的全部逻辑，这些逻辑取决于脚本从 Samovar 全局变量中获取的参数，例如 `getNumVariable("TankTemp")`，或者取决于任何脚本保存在公共命名空间中的变量，例如 `getObject("StartTankFilling")`。也就是说，可以在这个脚本的代码中写入例如这样的条件：
`if (StartTankFilling == "true" and TankFillingPercent < 80) then StartPump() else StopPump() end`
脚本每秒都会检查是否有需要向蒸馏釜加料的标志；如果有，再检查是否已经加满 80%，以及是否应该停止之前启动的泵。
- 按钮脚本 `btn_dist_button1.lua`：按下按钮时，通过 `setObject("StartTankFilling", "true")` 设置需要向蒸馏釜加料的标志。一秒后的下一个周期，`dist.lua` 脚本会读取这个标志并开启泵（如果蒸馏釜还没有加满）。这样，所有脚本共享一个可通过 `setObject/getObject` 访问的公共命名空间，可以借此交换数据。
- 按钮脚本 `btn_dist_button2.lua`：按下按钮时，可以通过 `setNumVariable("loop_lua_fl", 0)` 停止主受控循环，例如在脚本出现问题时。

L 型程序行（Lua 阶段）。除 NBK 外，所有模式的程序（精馏、蒸馏、醪塔、啤酒、奶酪制作）中都有 L 型行——由 Lua 控制的阶段。在程序编辑器中选择这种类型时，会打开“Lua 阶段”（«Lua-этап»）窗口，在其中设置：

Lua 文件——Samovar 内存中任何扩展名为 `.lua` 的文件（列表取自文件编辑器）。可以通过网页界面的文件编辑器上传自己的脚本。

阶段超时，秒——1 到 65535 秒。如果脚本在这段时间内没有调用 `setNextProgram()`，Samovar 会显示消息“Lua 未在超时前完成操作”（«Lua не завершила операцию до тайм-аута»）并结束该模式。

参数——传给脚本的任意数量的字符串。在脚本中可以通过 `arg` 表访问：`arg[0]` 是文件名，`arg[1]`、`arg[2]`..... 依次是各个参数。空参数写作 `""`。

程序运行到 L 行时，Samovar 运行所选文件并等待 `setNextProgram()` 调用——即转到下一行。脚本出错（“Lua 出错结束”（«Lua завершилась с ошибкой»））或超时都会停止该模式。L 型行不设置其他字段（温度、速度、容器、设备）。在文本形式中，这种行以 `^` 字符分隔文件和参数来保存，例如精馏中的

`L;120;hops.lua^3^45;0;0;0。`

`dose.lua` 脚本示例：按第二个参数给定的时间开启编号为第一个参数的继电器（在程序中：文件 `dose.lua`，参数 3 和 1500）：

```
set_i2c_rele_state(tonumber(arg[1]), 1)
delay(tonumber(arg[2]))
set_i2c_rele_state(tonumber(arg[1]), 0)
setNextProgram()
```

如果运行过程中通过文件编辑器上传了某个 L 行正在运行的文件的新版本，Samovar 会重新读取脚本，无需重新启动程序。

Samovar 函数

在脚本中可以访问 Samovar 的内部变量（其中一部分不能修改，只能读取），并调用 Samovar 的函数。切换模式期间，改变 Samovar 状态的函数会以错误“mode switch blocks state changes”结束。

引脚编号取决于开发板。在这里和下文，请传入其数值：

ESP32 DEVKIT：RELE_CHANNEL1 – 2，RELE_CHANNEL2 – 15，RELE_CHANNEL3 – 14，RELE_CHANNEL4 – 13，WATER_PUMP_PIN – 4，LUA_PIN – 34，ALARM_BTN_PIN – 35。

ESP32-S3：RELE_CHANNEL1 – 2，RELE_CHANNEL2 – 42，RELE_CHANNEL3 – 41，RELE_CHANNEL4 – 40，WATER_PUMP_PIN – 11，LUA_PIN – 4，ALARM_BTN_PIN – 48。

LILYGO T-Relay：RELE_CHANNEL1 – 21，RELE_CHANNEL2 – 19，RELE_CHANNEL3 – 18，RELE_CHANNEL4 – 5，WATER_PUMP_PIN – 2，LUA_PIN – 12。

脚本中定义了常量 `INPUT`、`OUTPUT`、`LOW`、`HIGH` 以及命令结果代码 `ACTUATOR_COMMAND_ACCEPTED`、`ACTUATOR_COMMAND_PENDING`、`ACTUATOR_COMMAND_APPLIED`、`ACTUATOR_COMMAND_FAILED`。

`pinMode(pin, mode)` – 与 Arduino 的同名函数相同，将指定的输入/输出（引脚）设置为输入或输出模式。只有 `RELE_CHANNEL1`–`RELE_CHANNEL4` 和 `LUA_PIN` 端口可以更改模式。也就是说，如果要把 DEVKIT 开发板的 `RELE_CHANNEL4` 端口设为输入，调用 `pinMode(13, INPUT)`；设为输出，调用 `pinMode(13, OUTPUT)`。

`digitalWrite(pin, Value)` – 与 Arduino 的同名函数相同，向数字输出（引脚）输出 `HIGH` 或 `LOW`。可用的端口有 `RELE_CHANNEL1`–`RELE_CHANNEL4`、`WATER_PUMP_PIN`、`LUA_PIN`、`ALARM_BTN_PIN` 和 `BTN_PIN`，其他端口会被忽略。也就是说，如果要在 `RELE_CHANNEL4` 输出端设置高电平，调用 `digitalWrite(13, 1)`。对于 `WATER_PUMP_PIN` 上的 PWM 水泵，值 1 让水泵全速运转，0 让它停止。如果加热紧急保护已触发，则不会写入加热通道（`RELE_CHANNEL1`、`RELE_CHANNEL4`）。

`digitalRead(pin)` – 与 Arduino 的同名函数相同，读取指定输入的值——`HIGH` 或 `LOW`。也就是说，如果要读取 `RELE_CHANNEL4` 输入或输出上的值，调用 `digitalRead(13)`

analogRead() – 与 Arduino 的同名函数相同，读取模拟输入 LUA_PIN 的值（无参数）。施加到模拟输入上的电压会被转换为 0 到 4095 的数值。注意！向输入端施加高于 3.3 伏的电压可能损坏该端口甚至整个 ESP32。在奶酪制作模式下，LUA_PIN 被 pH 传感器占用：对它调用 pinMode 和 digitalWrite 会出错结束。

exp_pinMode(pin, mode) – 与 pinMode(pin, mode) 相同，但用于控制 PCF8575 端口扩展器；pin 是扩展器端口号（0–15），mode 为 INPUT、OUTPUT 或 INPUT_PULLUP。扩展器的所有端口都可以选为读取或输出

exp_digitalWrite(pin, Value) – 与 digitalWrite(pin, Value) 相同，但用于控制 PCF8575 端口扩展器；pin 是扩展器端口号（0–15），Value 为 1 或 0。扩展器的所有端口都可以写入

exp_digitalRead(pin) – 与 digitalRead(pin) 相同，但用于控制 PCF8575 端口扩展器；pin 是扩展器端口号（0–15）。扩展器的所有端口都可以读取

exp_analogWrite(Value) – 与 analogWrite(pin, Value) 相同，但用于控制 PCF8591 端口扩展器，Value 为 0 到 255 的数值。该扩展器有一个可用于输出的端口。

exp_analogRead(pin) – 与 analogRead() 相同，但用于控制 PCF8591 端口扩展器；pin 是扩展器输入编号（0–3），返回 0 到 255 的数值。

exp_* 函数只存在于用 USE_EXPANDER（PCF8575）和 USE_ANALOG_EXPANDER（PCF8591）编译的固件中。

delay(ms) – 与 Arduino 的同名函数相同，按参数给定的毫秒数暂停程序执行（1 秒等于 1000 毫秒）。不超过 1000。

millis() – 与 Arduino 的同名函数相同，返回 Samovar 启动以来经过的毫秒数

setTimer(Num, Sec) – 将编号为 Num 的定时器设置为 Sec 秒（最多 65535）。共有 10 个定时器，编号 1 到 10。也就是说，脚本最多可以同时使用 10 个定时器

getTimer(Num) – 获取编号为 Num 的定时器的秒数。如果定时器未设置或时间已到，函数返回 0

sendMsg(Msg, Level) – 如果 Level = -1，消息 Msg 输出到 COM 端口和浏览器控制台，便于调试。如果 Level 为 0、1 或 2，消息作为系统消息发送（0 – 报警，1 – 警告，2 – 通知）：发送到显示屏、网页界面、应用和网站。

setPower(Power) – 开启/关闭 Samovar。setPower(0) 关闭，setPower(1) 开启（与当前模式的加热按钮相同）

setCurrentPower(Value) – 在调压器上设置值 Value。如果使用带功率控制的调压器（SEM/AVR），Value 是要设置的功率（瓦），否则是电压，精度 0.1 V。该函数只存在于带调压器的固件中。

setBodyTemp() – 在精馏过程中设定酒心温度。只在精馏模式下有效。在其他模式下会在控制台输出无法设定酒心温度的消息

setMixer(Val) – 开启/关闭搅拌器。setMixer(0) – 关闭，setMixer(1) – 开启

openValve(Val) – 打开/关闭水阀。openValve(0) – 关闭，openValve(1) – 打开

setPumpPwm(Val) – 设置水泵转速，Val 为 0 到 1023。只存在于带 PWM 水泵的固件中（USE_WATER_PUMP）。

setMixer、openValve、setPumpPwm 和 setPumpPwm 返回命令结果代码（ACTUATOR_COMMAND_* 常量）。

setAlarm() – 进入报警模式。Samovar 会切断电源、关闭水阀并关闭水泵

setNextProgram() – 转到下一个程序。相当于在界面中按下“下一个程序”（«Следующая программа»）按钮。只在加热开启时有效。

setPauseWithdrawal(Val) – 暂停/恢复取酒。setPauseWithdrawal(0) – 恢复，setPauseWithdrawal(1) – 暂停

setCapacity(Num) – 将舵机转到编号为 Num 的容器（0 – 排放，1 及以上 – 容器）。

setServoAngle(Angle) – 将舵机转到 Angle 度（0–180）。如果固件有舵机输出（SERVO_PIN），返回 1，否则返回 0。用于自制添加剂投放器（见 [添加剂投放器](#)）；舵机与 setCapacity 共用，因此在精馏中转动舵机后必须重新设置容器。

getState() – 获取 Samovar 状态（一个数字），便于判断 Samovar 所处的状态：0 – 空闲；精馏：10 – 正在取酒，15 – 程序行之间的自动暂停，20 – 程序完成，30 – 取酒泵校准，50 – 加速升温，51 – 稳定，52 – 稳定完成；40 – 手动暂停（任何模式）；1000 – 正在蒸馏，2000 – 啤酒，3000 – 醪塔，4000 – NBK，5000 – 奶酪制作。

setNumVariable("Variable", Val) – 设置 Samovar 的内部数值变量。并非所有变量都可以设置值（列表见下文）。setNumVariable("loop_lua_fl", 1) 会开启每秒一次的脚本循环。注意！如果不了解 Samovar 的工作原理，最好不要使用这个函数，因为它可能扰乱 Samovar 的运行

setStrVariable("Variable", Val) – 设置 Samovar 的内部字符串变量。并非所有变量都可以设置值。setStrVariable("SamovarStatus", "测试 Samovar 状态")

getNumVariable("Variable") – 获取 Samovar 的内部数值变量。TankTemp = getNumVariable("TankTemp") 会把蒸馏釜温度值赋给脚本变量 TankTemp

getStrVariable("Variable") – 获取 Samovar 的内部字符串变量。

program_type = getStrVariable("program_type") – 会把当前运行程序的类型赋给脚本变量 program_type。

脚本变量只在脚本运行期间存在，下一次运行时不会保留它们的值。有时需要在一次脚本运行中记住某个变量的值，并在另一次运行中读取它。为此有两个函数——setObject("Object", Val) 和 getObject("Object")/getObject("Object", "NUMERIC")。设置的变量值保留到重启为止。不同的对象最多 32 个，超出时 setObject 会出错结束。

setLuaStatus(Status) - 在界面中显示脚本状态（“附加”（«Дополнительно»）选项卡上的“Lua 状态”（«Статус Lua»））。例如，跟踪脚本的执行：

```
i = getObject("cnt", "NUMERIC")
```

```
i = i + 1
```

```
setLuaStatus("计数器 cnt = "..i)
```

```
setObject("cnt", i)
```

```
setObject("Object", Val) – 在 Samovar 内存中记住值为 Val 的对象 Object。
```

```
getObject("Object") – 获取之前保存的对象 Object 的值。如果没有保存过这个对象，返回空字符串。如果把空字符串当作数字使用，它的值会是 nil——未定义。
```

为了方便处理数值、免去对 nil 的检查，可以带附加参数 "NUMERIC" 调用这个函数；这时如果对象尚未初始化，会返回 0。

setObject/getObject 的使用示例：

```
n = 156
```

```
setObject("MyObject", n)
```

```
print (n)
```

```
n = n + 5
```

```
print (n)
```

```
n = getObject("MyObject")
```

```
print (n)
```

运行这个脚本，Arduino 串口监视器中会输出

```
156
```

```
161
```

```
156
```

http_request(Url) – 向地址 Url 发送 GET 请求并返回响应正文（或字符串 "error"）。例如，可以把读数发送到自己的网络服务。也可以传入四个参数来发送

POST 请求：地址、方法、Content-Type 头和正文：

```
http_request("http://test.com:80/post?foo1=bar1", "POST", "Content-Type: text/text; charset=utf-8", "body")
```

将蒸馏釜温度发送到外部网络服务的脚本示例（请把地址和参数换成自己的）：

```
local char_to_hex = function(c)
```

```
    return string.format("%02X", string.byte(c))
```

```
end
```

```
local function urlencode(url)
```

```
    if url == nil then
```

```
        return
```

```
    end
```

```
    url = url:gsub("%n", "%0A")
```

```

url = url:gsub("[^%w ]", char_to_hex)
url = url:gsub(" ", "+")
return url
end

function SendToServer(text)
    local key = "my_secret_key" -- 你的服务访问密钥
    http_request("http://example.com/notify?key=" .. key .. "&text=" .. urlencode(text))
end

local text = "当前蒸馏釜温度 = " .. getNumVariable("TankTemp")
SendToServer(text)

```

I2CStepper 板（搅拌器、计量泵、继电器）的函数——set_stepper_by_time、set_stepper_target、get_stepper_status、i2cpump_start、i2cpump_stop、i2cpump_get_speed、i2cpump_get_target_ml、i2cpump_get_remaining_ml、i2cpump_get_running、set_mixer_pump_target、get_mixer_pump_status、set_i2c_rele_state、get_i2c_rele_state、check_I2C_device——见 [I2CStepper](#) 页面。

变量

脚本中默认可用的内部变量列表（它们的值在每次脚本运行前填入，通过赋值修改没有意义）：

bme_pressure – 当前大气压值
 capacity_num – 正在取酒的容器编号
 SamovarStatusInt – Samovar 运行状态（与 getState() 相同）
 ProgramNum – 当前程序编号
 ProgramLen – 程序的行数
 ActualVolumePerHour – 当前取酒速度，单位为升
 WthdrwlProgress – 当前取酒的进度
 PowerOn – 加热开关标志。0 – 关，1 – 开
 PauseOn – 暂停标志。0 – 否，1 – 是
 StepperMoving – 蠕动泵运行标志。0 – 未运行，1 – 运行中
 program_Pause – 正在执行暂停程序的标志。0 – 否，1 – 是
 program_Wait – 程序处于暂停的标志。0 – 否，1 – 是
 program_Wait_Type – 暂停原因——塔节温度或蒸汽温度超限。取值为俄文字符串 "(nap)"（蒸汽）或 "(qapra)"（塔节）
 WFlowMilliLitres、WftotalMilliLitres、WFlowRate – 流量传感器测得的上一个间隔的水量、总水量（ml）和冷却水流量（只在带流量传感器的固件中）
 WthdrwTimeAll – 剩余取酒时间
 WthdrwTime – 当前程序行的取酒时间
 WthdrwTimeAllS – 剩余取酒时间（字符串）
 WthdrwTimeS – 当前程序行的取酒时间（字符串）
 pump_started – 水泵运行状态。0 – 否，1 – 是
 heater_state – 糖化时的加热状态。0 – 未加热，1 – 加热中
 mixer_status – 搅拌器运行状态。0 – 未运行，1 – 运行中
 alarm_event – 报警按钮已触发的标志。0 – 否，1 – 是
 acceleration_heater – 辅助加热管开启标志。0 – 关，1 – 开
 valve_status – 供水阀门状态。0 – 关闭，1 – 打开
 program_type – 当前程序的类型（来自精馏和啤酒程序的说明）
 program_volume – 取酒量，单位 ml
 program_speed – 取酒速度，单位 l/h
 program_temp – 这部分馏出物的取酒温度。0 - 自动确定
 program_power – 这部分馏出物取酒时的电压/功率
 program_time – 执行这一程序行所需的时间
 program_capacity_num – 取酒用的容器编号
 SamSetup_Mode – Samovar 工作模式：0 – 精馏，1 – 蒸馏，2 – 啤酒，3 – 醪塔，4 – NBK，5 – 低温慢煮，6 – Lua 模式，7 – 奶酪制作
 test_num_val – 用于调试的数值变量
 test_str_val – 用于调试的字符串变量
 SteamTemp – 蒸汽传感器温度
 PipeTemp – 塔节传感器温度
 WaterTemp – 水温传感器温度
 TankTemp – 蒸馏釜传感器温度
 ACPTemp – TCA 传感器温度
 current_power_mode – 调压器当前工作模式
 target_power_volt – 调压器设定的工作电压
 wp_count

可通过 getNumVariable 读取的变量（只读）：WfpulseCount、pump_started、valve_status、SamSetup_Mode、Samovar_Mode、Samovar_CR_Mode（当前模式和正在切换到的模式，编号与 SamSetup_Mode 相同）、SteamTemp、PipeTemp、WaterTemp、TankTemp、ACPTemp、WftotalMilliLitres、WFlowRate、program_volume、program_speed、program_temp、program_power、program_time、program_capacity_num、capacity_num、target_power_volt、PowerOn、PauseOn、program_Wait、alcohol（按温度计算的蒸馏釜酒精度）、alcohol_s（按蒸汽计算的酒精度）、water_pump_speed（水泵转速 0-1023）、pressure_value（蒸馏釜压力）、YY、MM、DD、HH、MI、SS（年、月、日、时、分、秒）。
 可通过 getNumVariable 和 setNumVariable 访问的变量：acceleration_temp、boil_temp（记住的沸腾温度）、wp_count、test_num_val、loop_lua_fl、SetScriptOff、show_lua_script。只能通过 setNumVariable 设置：pmpKp、pmpKi、pmpKd——水泵的 PID 系数。
 字符串变量（getStrVariable/setStrVariable）：SamovarStatus——主页面上的状态文字（可读写），test_str_val（可读写），program_type（只读），Msg——读取时返回下一条未读系统消息或空字符串，写入时添加一条消息。

用于控制脚本的变量：

loop_lua_fl – 取值 0 或 1。0 – 不运行每秒脚本循环，1 – 运行。
 SetScriptOff – 取值 0 或 1。1 – 停止每秒循环（重置 loop_lua_fl）。程序结束时 Samovar 会自动将其设为 1。
 show_lua_script – 取值 0 或 1。0 – 不在控制台和串口监视器中显示正在运行的脚本，1 – 显示。可用于调试。同时还会显示为该脚本设置的所有变量的值。

例如，如果在 init.lua 脚本中调用下面这两个函数，每秒会运行两个脚本（script.lua 和由当前 Samovar 模式决定的脚本），并且每个脚本都会输出到串口监视器和浏览器控制台

```
setNumVariable("loop_lua_fl",1)
setNumVariable("show_lua_script",1)
```

脚本示例

下面是一个可以直接在网页界面编辑器中编写的脚本示例。它每秒运行一次，当蒸馏釜温度达到 70 度时，开始交替开关接在 PCF8575 扩展器第一个端口上的设备，并把相应信息写入 UART 和浏览器控制台

```
if TankTemp > 70 then
  if exp_digitalRead(1) == 0 then
    exp_digitalWrite(1,1)
    print("Включено")
  else
    exp_digitalWrite(1,0)
    print("Выключено")
  end
end
end
```

下面的脚本示例从 LUA_PIN 输入（DEVKIT 开发板为 34）上的模拟液位传感器读取数值，并控制水泵（例如向蒸馏釜泵入发酵醪）。信号在 1000 到 2000 之间时打开泵；信号为零时关闭泵。脚本直接在浏览器中编写，保存后即可执行。引脚 4 是 DEVKIT 开发板的 WATER_PUMP_PIN；其他开发板请换成自己的编号。

```
start_pump = getObject("start_pump", "NUMERIC") + 0 --从之前保存的对象中获取数值并转换为数字
sensor = analogRead() --读取 LUA_PIN 的模拟值（为 Lua 保留）
--sensor = 1500
if sensor >= 1000 and sensor <= 2000 and start_pump == 0 then
  setObject("start_pump", 1) --把值保存到 Samovar 内存中的对象里，以便在下一个脚本运行周期中使用
  digitalWrite(4,1) --在引脚 4 上输出高电平
  print("Start pump") --[写入 COM 端口。也可以使用 sendMsg("Msg", Level) 命令。如果 Level = -1，消息输出到 COM 端口和浏览器控制台，便于调试。如果 Level 为 0、1 或 2，消息作为系统消息发送]=]
  --sendMsg("Start pump",-1)
elseif sensor == 0 then
  setObject("start_pump", 0) --把值保存到 Samovar 内存中的对象里，以便在下一个脚本运行周期中使用
  if (start_pump == 1) then
    digitalWrite(4,0) --在引脚 4 上输出低电平
    print("Finish pump")
  end
end
end
```

随网页界面一起上传到 Samovar 的现成脚本：

rectificat.lua —— 根据液位传感器或水流量传感器用泵给蒸馏釜加料（体积和传感器阈值在文件开头设置）。

dist.lua —— 分三个容器的蒸馏：记住开始沸腾时的蒸馏釜温度，当温度升高设定值（或酒精度下降）时，把舵机切换到下一个容器，并可按需降低功率。

cheese.lua —— 奶酪制作程序 L 行的脚本模板。

btn_<mode>_button1.lua “开始”（«Начать»）和 btn_<mode>_button2.lua “停止”（«Остановить»）—— 为各模式打开和关闭每秒一次的循环。

更多示例见 [GitHub 上 Samovar 代码库](#) 的 Lua_script 文件夹：发酵温度控制（beer_mash.lua）、用按钮控制水泵转速、带浓缩的蒸馏（dist_degidrate.lua）、通过 HTTP 把读数发送到外部网络服务（http_request_telegram.lua）、借助压力传感器保持预液泛，以及 NBK 脚本。

问题解答。FAQ

我没有装蒸汽传感器的测温套管。怎么办？能不用这个传感器吗？

Samovar 工作需要塔节传感器和蒸汽传感器的读数。可以这样做：

第 1 种方法。在填料塔节上沿以下 15-20 cm 处安装一个传感器，把它当作蒸汽传感器。第二个、较低的传感器装在上面那个以下至少 30 cm 处；如果塔节较短，则装在塔高的 2/3 处。

第 2 种方法。只安装下面的传感器。在“温度” («Температура») 选项卡上把它选为塔节传感器，同时也把同一个传感器选为蒸汽传感器。为了看起来直观，把它作为蒸汽传感器的读数修正为比塔节传感器高 1-2 度，并把它的超出设定值设得比塔节传感器大。

塔高的 2/3 是什么意思？在哪里？

这里只指塔中装有填料部分的高度，上部或下部的空段不算。拿卷尺量出填料在塔中占的高度，除以 3。从上往下数两份就是 2/3，也就是从下往上数一份。这只是建议，您也可以根据自己的习惯或设备结构安装塔节温度传感器。

回流液位传感器应该装在哪里？

通常液位传感器直接装在视镜上，距视镜下沿 5-7 mm。传感器需要调整：塔进入稳定状态、视镜热透后，再设置触发阈值。

什么是设定值？

触发设定值是在调节器中设置的被控过程参数值，达到该值时调节器开始动作。换句话说：实际值超出设定值的允许“偏差”，在此范围内系统保持不动作。

塔已经装好，传感器都在保温层下面。怎么知道哪个传感器对应哪个地址？

这并不难。首先，在下拉菜单中为每个安装位置选择不同的传感器地址，随便选，但要互不相同。

然后看哪个传感器够得着。比如水传感器：用手捂热，或用吹风机稍微吹一下。看主页，其中一个传感器的温度开始上升，比如是“蒸汽”。到“温度” («Температура») 选项卡上，把水传感器和蒸汽传感器的地址对调。

下一个是 TCA 传感器，做法相同：把 TCA 与温度开始上升的那个对调。这样两个传感器就设置好了。

其余的在保温层下面够不着。没关系。把粗酒倒进蒸馏釜，准备开始运行。

准备好了吗？开启加速加热。首先升温的应该是釜内传感器，但您的“蒸汽”又在上升，那就对调地址，依此类推。下一个会是塔节传感器，最后是蒸汽传感器。所有传感器都各就各位，继续运行。

警告。选择地址时请确保不重复。Samovar 不检查这一点。每个安装位置的传感器地址必须唯一。

警告。设定值、超出设定值、温度等设置不会随传感器地址一起转移，它们属于具体的安装位置。

如何校准传感器并消除误差？

方法是在“修正” («Корректировка») 字段中输入数值。先说明一点：并不需要绝对精确的实际温度值。别人的蒸汽温度是 77，您的是 75，这关系不大。下面回到校准。

最准确的方法是在接近工作温度的条件下同时校准所有传感器。

可以把所有传感器放进沸水中，把它们的温度都设为 99。

但不是所有传感器都能放进水里。可以把它们放进一个空容器，再把容器放到水浴中，底部温度会接近 99。

更简单的是让传感器彼此对齐校准。

例如冬天把它们放到暖气片上盖好，等温度稳定后给它们设成同一个值。这样的办法可以想出很多。

如果传感器已经在保温层下面怎么办？可以把塔的所有部件摆在桌上，把空闲的传感器放进塔节里，用厚实的东西盖住，过一段时间后让它们彼此对齐校准。

啤酒糖化程序的说明中提到“不同的酿造方式”。那是什么？

啤酒生产中的酿造方式是指用于煮制麦汁的专用设备系统。不同的工艺任务对应不同的酿造方式。[这篇文章](#)详细介绍了家庭酿造方式的类型及其工作原理。

Samovar 能检测哪些设备和传感器是否存在？

由于没有反馈，Samovar 不检查以下传感器和执行机构是否存在：

- 接触器/调压器开启 (1 号继电器)
- 辅助加热管继电器/调压器加速 (4 号继电器)
- 蜂鸣器
- 供水阀 (3 号继电器)
- 回流液位传感器
- 蠕动泵步进电机
- 取酒转盘舵机

- 冷却泵
- 大气压传感器
- 釜内压力传感器。
- 搅拌器 (2 号继电器)

Samovar 会检查以下传感器和执行机构是否存在，条件如下：

- 所有温度传感器——如果过程开始时该传感器存在。缺失的消息会定期出现。
- 水流量传感器——如果启用了它。
- 加热调压器——如果启用了调压器使用检查。
- I2CStepper 板（搅拌器、水泵）：Samovar 启动时以及之后在后台轮询；如果通信中断，会显示消息，并隐藏“I2CStepper”选项卡和与该板相关的字段。

如何使用旁通控制阀

需要两个手阀、两个电磁阀和两个三通。

一个电磁阀为 NC（常闭），另一个可以是 NC 也可以是 NO（常开），最好用 NC。连接方式 如下：

进水管——第一个电磁阀，这是主阀，用于打开冷却供水。

接着是三通，两个手阀都接在三通上。其中一个（通常是针阀）作为主手阀，另一个作为辅助手阀。辅助手阀后面接另一个电磁阀，然后两路再通过一个三通汇合成一路，把水送到 Samovar。

也就是说，在主手阀旁边做一条旁路。

用主手阀调节水流量，使分凝器出口温度为 50°，比如 40 升/小时。用辅助手阀调到 10 l/h。在 Samovar_ini.h 文件中注释掉 `#define USE_WATER_PUMP` 这一行，并取消 `#define USE_WATER_VALVE` 的注释（水泵和阀门不能同时工作）。

工作原理：釜内温度达到设定阈值（默认 77°）时，Samovar 打开主阀，40 l/h 的水进入冷却系统。如果水压下降，温度开始上升，辅助阀打开，额外的 10 l/h 进入冷却系统。温度回到设定范围内时，辅助阀关闭。

链接。实用建议。

各种资源的链接，提供有关元器件及其设置的有用信息。

Samovar 6.26 及更早版本的帮助文档，PDF 和 DOC 格式。

[下载 PDF 格式的帮助用户文档](#)

[下载 .doc 格式的帮助用户文档](#)

注意。

PDF 文档只在发布对界面或 Samovar 工作方式有重大改动的版本时才更新，因此可能与网站上的文档有所不同。

[下载 Samovar](#) 发布版本。

[如何安装 Arduino IDE](#)

[关于使用 Arduino IDE 的更多信息](#)

[在 Arduino IDE 中安装和配置 ESP32 开发板](#)

[计量器](#) 以及 dranek 提供的电路图。

版本历史

7.00 — 2026 年 9 月 4 日

重大更新。主要变化：

新增“奶酪制作”（«Сыроварение»）模式：加热、保温、冷却，用 pH 传感器控制酸度（PH-4502C 接在 LUA_PIN 输入上，在“pH 校准”（«Калибровка pH»）页面校准），搅拌机，4 号继电器上的盐水排放阀，以及专用的程序行类型。设置位于“奶酪”（«Сыр»）选项卡。

精馏模式下的杂质穿透检测器：带气压修正地分析温度趋势，根据塔的噪声自行选择阈值，温度上升时可降低取酒速度，温度严重上升时暂停取酒并自行恢复；在酒头阶段只做观察，等待原因显示在界面中。

啤酒、BK 和 NBK 模式重大更新：All-in-one、HERMS 和 RIMS 酿造方案，改进的 BeerXML 导入，重写的 NBK 算法（预热、调整、优化、运行）；啤酒和奶酪程序中新增 L 行——由 Lua 控制的阶段。

除 NBK 外的所有程序都支持 L 行（Lua 阶段）：精馏、蒸馏、BK、啤酒、奶酪制作。脚本文件、以秒为单位的超时时间和参数（脚本中的 arg 表）在程序编辑器的“Lua 阶段”（«Lua-этап»）窗口中设置；通过编辑器上传新版本文件后，正在运行的脚本会重新读取。在 BK 中，“下一个程序”（«Следующая программа»）按钮现在可以使用。

新的网页界面：深色“铜”主题，带实时读数的工艺流程图（啤酒模式下按酿造方案显示，精馏模式下显示分凝器以及已启用的第二台泵），当前程序行，图表，各模式页面，“外部泵”（«Внешний насос»）页面和“I2CStepper”页面；页面以压缩形式传输，初始数据通过一次请求获取。

通过 TLS 连接 samovar-tool.ru 服务器，取代 MQTT：读数、命令、过程日志和消息都经由此通道传输；网站上的日志页面、网站上的控制页面以及带报警推送通知的 Android 和 iOS 手机应用都基于此通道。向自建 MQTT 代理发送数据作为可选的编译设置保留。

I2CStepper 板，协议第 2 版：两块板（搅拌器和泵），灌泵，按设定容量加注，“每毫升步数”校准，四个继电器，一个外部传感器。可从“I2CStepper”页面、程序、Lua、应用和网站进行控制。I2C 泵可用作巴氏杀菌段上方的酒头取酒泵。

Lua：单次运行时间限制（20 秒），反复出错后循环停止，新函数 setPumpPwm 和 setCapacity，I2CStepper 板函数，Lua 模式成为独立的运行模式。

状态快照：紧急重启后，如果在 30 分钟内重新启动过程，固件会恢复程序并继续同一个日志会话。

适用于 Windows、macOS 和 Linux 的配置工具：编译，通过 USB 或 Wi-Fi 烧录，设备文件编辑器，配置报告。支持 ESP32-S3。

重写了精馏和蒸馏程序的计算、程序和过程状态的保存以及设备错误处理。

6.27 — 2026 年 5 月 11 日

修复了计时器、网络超时、用万能按钮启动接入点、蒸馏结束、啤酒模式下的 UART 调压器控制、功率显示、PID 自整定以及冷却泵读数。

6.26 — 2026 年 1 月 1 日

更新了库，提高了稳定性，降低了内存占用。新增无需重启即可切换模式。更新到此版本前，建议通过“设置”（«Настройки»）→“其他”（«Прочие»）→“保存设置”（«Сохранить настройки»）保存设置。

6.25

- 加入了用户建议，更新了库，另有一些小改进。
- 修正了分区文件。同时修正了向 Telegram 发送消息的函数。

6.24.

- 修复了之前发现的错误。
- NBK 模式也已可以使用。
- 改进了网络处理。

6.23.

- 修正了网页界面。
- 修复了错误。
- 修正了啤酒程序中泵的工作。

6.22.

- 主要变化——更新了库。
- 在啤酒模式下，搅拌机运行时间现在以秒而不是分钟为单位设置。
- 此外，蒸馏模式下会显示预计完成时间。

6.20

- 修复了错误。
- 实现了 NBK 模式，基于以下用户的研究：[dranek](#)
- 修改了酿造啤酒时加热器和麦汁转移泵的工作方式。
- 由于 Blynk 中的通知已停止工作，新增了 Telegram 通知。

6.19.

- 新增 NBK 模式。已实现安全组的主要功能和冷却泵的手动控制。自动化目前计划通过 Lua 实现。
- Lua 中新增用于处理当前时间的数值变量。

YY、MM、DD、HH、MI、SS——分别为年、月、日、时、分、秒。

调用示例：

```
setLuaStatus(getNumVariable("HH"))
```

6.18.

- 新增 Lua 变量。
- 更新了 6.16 和 6.17 版本啤酒模式的界面——PID 自整定模式恢复了。
- 现在可以在界面中选择，在糖化程序中关闭搅拌机。
- 代码小幅优化

下次 Samovar 启动时界面会自动更新。

6.17.

- 更新了文件编辑器版本 ([在设置中](#)).
- 更新了屏幕菜单库。
- Samovar 和 [I2CStepper](#) 草图 (sketch) 做了小幅优化。
- 压缩包中现已包含最新版本的 Meter 以及来自以下用户的电路图：[dranek](#)，还有 SEM(Stab)_AVR 调压器外壳的 3D 模型
- 借助 Lua 脚本（目前阶段），啤酒模式可以实现任何[酿造方案](#)，完全自动化，并可使用 beerxml 或 BrewMate 格式的配方。对配方的支持仅限于构建糖化程序所需的最低程度。也就是说，配方中可能不会显示制备糖化醪所需的全部原料
- 功能更新之一——在温度设置中新增了刷新温度传感器的按钮。可以逐个连接传感器并保存到设置中。接上一个传感器，选择它所属的位置，点击保存，回到设置，接上另一个传感器，选择它所属的位置，点击保存，依此类推。
- 运行逻辑没有任何变化。

6.16.

- 现在可以启用取酒泵的平滑加速。不过它停止时也会平滑减速，在从一个程序行切换到另一个程序行时，这可能不利于过程的稳定。

要启用步进电机的加速/减速，请在 Samovar_ini.h 文件中将以下内容

```
##define USE_STEPPER_ACCELERATION //использовать плавный старт и торможение шагового двигателя
```

替换为

```
#define USE_STEPPER_ACCELERATION //использовать плавный старт и торможение шагового двигателя
```

- 新增了无需导线将调压器连接到 Samovar 的说明。该说明也可在以下位置查看：[网站](#).

- 更新了步进电机库，改进了与 I2CStepper 的交互。

发布了 I2CStepper 0.3 版，修复了错误。如果需要速度精确，请降低驱动器上的细分倍数（不超过 1/2）；要更精确，请设为 1。草图中细分默认设为 1/2。可在 I2CStepper.h 文件第 22 行调整，那里指定了分频值。

更新了界面。在“啤酒”模式下，程序行中的搅拌器参数在单独的窗口中编辑。

注意！ Samovar 和 I2CStepper 所用的库必须取自 Samovar 附带的库。

6.15

- 初始设置文件 tools/Samovar_default_settings.txt 中新增了默认值。编译后首次启动 Samovar 时，必须加载默认值。选项已经很多，很难一次全部设置正确，而默认值能让 Samovar 正常工作。之后可以按自己的需要进行调整。

- 修复了啤酒模式下用于投放啤酒花的步进电机的错误。

- Samovar_ini.h 文件中新增了一个选项，用于从网页界面控制是否使用回流传感器（默认启用）

```
#define IGNORE_HEAD_LEVEL_SENSOR_SETTING //如果使用回流传感器，则忽略网页界面中关闭该传感器的命令
```

(详见[此处](#))。

6.14

- 如果在 Samovar 通电前按住按钮（通电 2 秒后松开），Samovar 将以接入点模式启动。网络：Samovar，密码：SamApp123

- 新增对 Blynk 终端的支持——可以运行一小段 Lua 代码（几行；不建议用这种方式运行大型脚本）。如何在 Blynk 中添加终端——[见此处](#)

- 在 Samovar 中配置了 Lua 后，也可以在网页界面中运行几行 Lua——在所有模式的“附加”（«Дополнительно»）选项卡上可用

- Lua 现在可以控制水泵的 PID 系数——分别为 pmpKp、pmpKi、pmpKd。要设置系数，可以运行以下 Lua 代码
- ```
setNumVariable("pmpKp", 6)
```

当前值可以这样获取：

```
setLuaStatus("pmpKp"..getNumVariable("pmpKp"))
```

- 啤酒模式下新增用于自动投放啤酒花的步进电机控制。有待测试。

- 通过互联网更新网页界面的问题应该已经解决。

- 还修复了一些小错误。

#### 6.12

- 新增对三种压力传感器的支持：

- XGZP6897D（通过 I2C 连接），

- MPX5010D（模拟量，接 ESP32 的 34 号引脚）

- 通过 1Wire。通过 1Wire 连接的传感器地址固定为 {0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22}。使用哪种传感器



(或不用) 在 Samovar\_ini.h 中选择：

```
//定义所用压力传感器的类型。只能使用一种类型。如果取消注释了三个或两个传感器，则只使用 MPX5010D
//#define USE_PRESSURE_XGZ 32 //использовать датчик давления XGZP6897D (или аналогичный). Какое
использовать значение - смотреть параметр K в таблице:
```

<https://github.com/fanfanlatulipe26/XGZP6897D>

```
//#define USE_PRESSURE_1WIRE {0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22} //использовать
датчик давления по 1Wire
```

```
//#define USE_PRESSURE_MPX //использовать датчик давления MPX5010D
```

界面中还没有相关逻辑和显示，但可以在 Lua 脚本中使用：把数值输出到 Lua 状态，并在那里实现达到某些数值时 Samovar 应有的行为。此外，在 Samovar 设置中可以设置压力上限，达到该值时 Samovar 将转入紧急模式。如果设置中的压力值为 0，Samovar 将忽略压力传感器的读数。

将压力输出到 Lua 状态的示例：

```
pressure_value = getNumVariable("pressure_value") + 0
setLuaStatus("давление = "..pressure_value)
```

- 为了查看压力传感器信息，新增了向 Blynk 的输出。可以自行将其添加到 Blynk 界面中，使用虚拟引脚 22。
- 新增通过 1Wire 扩展板连接 1Wire 压力传感器的支持，并可在 Blynk 中查看。
- Samovar 显示屏上现在显示全部 5 个温度传感器的读数。

## 6.11

- 啤酒模式。
  - 除供水阀门外，还会开启冷却泵。
  - 新增冷却泵转速控制。
  - 新增对“搅拌器” («Мешалка») 字段格式是否正确的检查。如果格式不正确，会改为 0^0.00^0^0
- 精馏模式。在“设置” («Настройки») >“基本” («Основные») 中新增选项：是否使用回流液位传感器。
- 蒸馏模式。在“设置” («Настройки») >“基本” («Основные») 中新增选项：以分钟为单位的时间，若蒸馏釜传感器数值在此时间内不变，则停止蒸馏。为 0 时不考虑该时间。
- 当前加热管功率的字体加大了。

## 6.10

- 在啤酒模式下，程序行中新增选择用哪个温度传感器控制加热（用于实现各种酿造方案）。可在 5 个传感器中选择。
- 自检模式下泵的运行时间延长了
- 在啤酒和蒸馏模式下，程序和当前程序执行状态现在会保存到文本文件中，与精馏模式的做法相同。
- Lua 现在可以调节水泵的输出。新增函数 setPumpPwm(p)，p=0——泵关闭，p=1023——最大输出。
- 现在可以在设置中关闭水流传感器监测。
- 设置中新增选项——啤酒模式下煮沸时不降低功率。

**注意！** 可以使用 SDK 2.0.13。如果遇到稳定性问题，请回退到 2.0.9

## 6.9

扩展了通过 LUA 脚本控制步进电机的功能。

如果未连接 Arduino Nano 上的 I2CStepper 扩展板，调用 Lua 函数 set\_stepper\_target 将控制内置步进电机；否则通过扩展板控制。

控制内置步进电机时只考虑一个参数——spd，即每秒步数表示的速度；调用函数时仍需传入全部三个参数。

--set\_stepper\_target(spd, direction, target) //spd——速度，每秒步数；direction——正向或反向；target——步数

set\_stepper\_target(950, 0, 0)

spd = 0 时步进电机停止。

**注意！** 更新到此版本时必须使用 SDK 2.09 版

## 6.8

- 通过界面按钮停止 Lua 脚本的方式改变了。
- 新增 SetScriptOff 变量。在按钮脚本 btn\_button2.lua 中它被置为 1，之后还会再执行一个 Lua 脚本运行周期。
- 这样做是为了能够捕获脚本停止事件，并执行必要的操作，使脚本正确结束。

## 6.7

- 错误修复和小改进。
- 值得注意的是——6.6 版在蒸馏模式下无法检测沸腾开始。已修复。

## 6.6

- 在蒸馏模式下新增了切换到下一个程序的按钮。修复了与检测沸腾开始以及蒸馏模式下程序运行相关的错误。

## 6.5

- 草图现在可以改变取酒蠕动泵步进电机的旋转方向。为此，请在 Samovar\_ini.h 中取消注释以下行：  
//`#define STEPPER_REVERSE` //изменить направление вращения шагового двигателя
- 还修复了过早检测到沸腾开始的问题。

## 6.4

- 修复了蒸馏釜中酒精度的计算
- 同时修复了取酒酒精度的计算。
- 新增向云端上传酒精度；在网站日志中查看蒸馏和精馏图表时会显示酒精度。
- 安装更新时，请将 SDK 更新到最新版本。

## 6.3

- 改动涉及蒸馏模式——新增了蒸馏程序。它的作用是按温度或酒精度更换取酒容器，除此之外不做任何事。蒸馏的启动和结束算法没有改变，所以如果你不打算分容器接取，有没有程序对你来说都一样。我还加入了蒸汽中酒精含量的计算（对蒸馏来说相当于取酒中的酒精含量）。我只从数学上验证了计算，实际条件下可能与酒精计读数有出入。两个数值都显示在界面中。
- 程序中既可以使用蒸馏釜中的酒精度，也可以使用取酒的酒精度；由取酒行的类型决定。
- 如果不需要分容器接取，就不必连接舵机。此时无论蒸馏程序里写了什么，都只是普通蒸馏。

## 6.2

修复的错误：

- 在 SDK 2.0.5 上，Samovar 在精馏模式下可能会重启。
- 在精馏模式下，取酒程序写入 prg.csv 文件时不正确。代码优化。

## 6.1

- 改进——在确定沸腾温度和蒸馏釜初始酒精含量时加入延时。
- 关于蒸馏釜内容物酒度的数据 将出现 在消息和界面中，并发送到 Blynk。

## 6.0

主要改进：

- 以前很不方便：Samovar 的存储总是被覆盖，每次都得恢复自己的成果（如果有的话）——精馏程序、Lua 脚本。现在，在有互联网连接时，Samovar 草图版本变化后界面会自动更新，而不会覆盖 Samovar 存储中的其他文件，这样就可以在 Samovar 存储中保存每种模式的设置，并在切换模式时加载。如果是配置一块新的 ESP32，会下载 data 文件夹中的全部文件。手动安装（不通过互联网）的方式仍然保留，但所用的文件系统变了，所以手动上传时必须使用的不是 `ESP32 Sketch Data Upload`，而是 `ESP32 LittleFS Data Upload`。更换文件系统是为了优化 Samovar 的运行速度。
- 界面会针对具体的草图版本自动下载（从 6.0 开始），也就是说，如果需要回退到旧版草图，界面也会回退到相应版本。  
(即更新或回退时，网页界面会自行更新。草图和库需要手动更新)
- 改进了与 Lua 脚本的交互。脚本示例和 Lua 文档放在单独的文件夹 (Lua\_script) 中。
- 使用脚本语言可以利用 Samovar 的全部功能，并针对你的具体任务进行扩展。例如——按 Gabriel 法蒸馏或控制 NBK。
- 取酒程序的改动。引入了“蒸汽温度” («Температура пара») 数值（在后续版本中已停用）。
- 泵加大冷却时的 TCA 温度值移到了“温度” («Температура») 选项卡。
- 修复了发现的错误。

## 5.27

- 改动涉及啤酒模式下搅拌器和泵的工作。啤酒模式程序中新增了一个参数——搅拌器工作模板。
- 在啤酒模式下，开启搅拌器时，Samovar 会接通 2 号继电器和 SSD 继电器；如果连接了 Arduino Nano 上的扩展板（用于控制步进电机），还会向其发送命令，开启步进电机和另一个继电器。
- 在搅拌器工作模板中可以设置搅拌器的运行时间、旋转方向（仅限步进电机）和暂停。例如：搅拌 2 分钟，等待 1 分钟，然后反方向搅拌 2 分钟，等待 1 分钟。

### 5.27b (测试版)

- 用于测试 Lua 脚本的版本。详情[见论坛帖子](#)。

## 5.26

- 1. 在精馏过程中，新增保存精馏状态和取酒程序。取酒程序保存到 prg.csv 文件，状态保存到 state.csv 文件。这些文件可以通过设置中的编辑器查看（默认登录名和密码均为 admin），也可以在以下地址下载：YOUR\_IP/prg.csv 和 YOUR\_IP/state.csv。

- 在 Samovar\_ini.h 中新增了一个参数，用于设置调压器输出电压的延时，从接触器动作后开始计算。
- 此外还做了一些小改进、优化，并修复了发现的错误。

#### 5.25

- 新增对 I2CStepper 的支持功能。
- 精馏模式下程序界面的改动——编写程序时，现在会显示每一行的时间以及整个程序的总时间。

#### 5.24

界面改动。

#### 5.23

在精馏和啤酒模式下，状态中新增了正在执行的程序行的类型。

还修复了发现的错误。

实现了以下格式啤酒配方的加载：

1. [BeerXML](#)（在西方非常流行），主要的配方库在[这里](#)，但下载需要付费，还有[这里](#)以及许多其他地方。
2. [Brewmate](#)（在前苏联国家很流行）。

配方示例可通过链接查看。可以从网站下载配方，加载到 Samovar 中，它会根据这些配方构建糖化程序。这些网站上不仅有配方，还有配方计算器。任何配方都可以改编后加载到 Samovar 中。

这两种格式都非常灵活，很可能并未考虑到它们的所有特性。如果在使用配方时遇到问题，请通过电子邮件发送配方 [info@samovar-tool.ru](mailto:info@samovar-tool.ru)

#### 5.22.

主要变化之一——预液泛（«Предзахлеб»）和酒心（«Тело»）程序中自动修正酒心 T 的功能默认关闭。另外还做了一些与编写 Samovar 功能文档相关的小改动。

#### 5.21

浏览器中新增了 Samovar 消息队列。现在很难错过消息：它们会一直显示在页面上，直到点击消息本身或“清除”

（«ОЧИСТИТЬ»）按钮明确清除为止。

浏览器中新增了关键事件的声音提醒——收到关键消息时，浏览器会一直发出声音，直到消息被清除。可在设置中开启/关闭。现代浏览器出于对用户的考虑，默认阻止自动播放声音。有两种办法：点击页面上的任意元素，或者在浏览器设置中把 Samovar 的地址加入例外。

还实现了以下功能：

如果当前程序行是“预液泛”（«Предзахлеб»），且其后还有一行酒心取酒，则会自动修正酒心 T。手动修正仍然可用。

延长了自检期间冷却水泵的运行时间。

修复的错误：

从 Samovar 下载日志并不总能成功。已修复。

在自检模式下，取酒泵的步进电机可能不会开启。已修复。

#### 5.20

修复了发现的错误。

#### 5.19.

修复了发现的错误。

#### 5.18.

修复了 Samovar 在预液泛模式下运行的一个错误。该错误从 5.11 版起就存在。如果你使用 5.11-5.17 版并使用预液泛模式，必须更新。

#### 5.17.

优化了通过 UART 协议与调压器的协作，并修改了液泛传感器触发与水温超限同时发生时的电压复位逻辑。

#### 5.16.

支持 BeerXML 格式。（该格式非常灵活，可能没有考虑到格式的所有变体）

#### 5.15.

改进了自检。新增加载 BeerXML 格式的功能（目前仅能查看）。更改 Samovar 运行模式并保存设置后，Samovar 会重启以切换到所设模式。在蒸馏模式下，取酒一直进行到设置中指定的温度，或者直到蒸馏釜温度 10 分钟内不再变化。此外还有一些小的系统改进。

#### 5.14.

改进了对以下调压器的控制：**kvic**，优化了对以下调压器的控制：**SEM\_AVR**。

“设置” («Настройка») 页面新增了一个按钮——“自检” («Самотестирование»)：供水泵会开启 5 秒和/或水阀打开，取酒蠕动泵启动，舵机依次转过前 5 个位置后返回。

#### 5.13.

啤酒模式新增一种程序类型——“发酵” («Брожение»)。它用于把蒸馏釜中的温度保持在设定值的允许范围内。如果蒸馏釜温度低于设定温度减去设定偏差，加热器和辅助加热继电器开启；当蒸馏釜 T 达到设定值减去设定偏差时，加热关闭（与其他啤酒模式一样由 PID 控制器工作）。如果蒸馏釜 T 高于设定温度加上设定偏差，会触发一个继电器。根据你的设备配置，可以在它上面连接供水阀门、泵或其他执行机构。当蒸馏釜 T 降到设定值加上设定偏差时，继电器关闭。附加逻辑可以在 Lua 脚本中实现，必要时可使用端口扩展板来控制任何执行机构或传感器。

还修复了错误并更新了库。SDK 必须更新到 2.0.5 版。

在某些情况下 Samovar 不保存 WiFi 设置；通过更新库和更新 SDK 解决了这个问题。

控制 kvic 调压器（旧固件）时存在一些错误。这个问题也已解决。

新固件的 kvic 调压器暂时无法与 Samovar 配合工作。

如果有人使用“用中国模块制作的简易加热管功率稳定器” («Простой стабилизатор мощности ТЭНа из китайских модулей») ，要与新版 Samovar 正确配合，必须使用 Samovar 附带的 Stab-avr 文件夹中的草图对其进行更新。

#### 5.12. 小改动。代码优化。

#### 5.10.

更换了水果、谷物和糖原料的取酒程序（相应文件位于 tools 文件夹中，更新界面时也会加载到 Samovar 存储中）。此外，在取酒程序计算界面中现在可以选择原料类型（水果、谷物、糖），所需的取酒程序会立即加载。