Hy=	Set hysteresis	0.0 to 5.0°C or 0.0 to 10.0°F
hy2=	Set hysteresis for second temp probe	0.0 to 25.0°C or 0.0 to 50.0°F
Tc=	Set temperature correction	-5.0 to 5.0°C or -10.0 to 10.0°F
tc2=	Set temperature correction for second temp probe	-5.0 to 5.0°C or -10.0 to 10.0°F
dI=	Device ID for Fine Offset 433MHz firmware	0 to 15
SA=	Setpoint alarm	0 = off, -40 to 40°C or -80 to 80°F
SP=	Set setpoint	-40 to 140°C or -40 to 250°F
St=	Set current profile step	0 to 8
Dh=	Set current profile duration	0 to 999 hours
Cd=	Set cooling delay	0 to 60 minutes
Hd=	Set heating delay	0 to 60 minutes
rP=	Ramping	0 = off, 1 = on
Pb2=	Enable second temp probe for use in thermostat control	0 = off, 1 = on
Rn=	Set run mode	Pr0 to Pr5 and th

Hysteresis, is the allowable temperature range around the setpoint where the thermostat will not change state. For example, if temperature is greater than setpoint + hysteresis AND the time passed since last cooling cycle is greater than cooling delay, then cooling relay will be engaged. Once the temperature reaches setpoint again, cooling relay will be disengaged.

Hysteresis 2, is the allowable temperature range around the setpoint for temp probe 2, if it is enabled (Pb=1). For example, if temperature 2 is less than *SP* - *hy2* cooling relay will cut out even if *SP* - *hy* has not been reached for temperature (1). Also, cooling will not be allowed again, until temperature 2 exceeds *SP* - 0.5 * *hy2* (that is, it has regained at least half the hysteresis).

Temperature correction, will be added to the read temperature, this allows the user to calibrate temperature reading. It is best to calibrate around your working point. That means for fermentation, it is better to calibrate at room temperature against a reference thermometer than using ice water.

Temperature correction 2, same as tc but for secondary temp probe.

Device ID Only available for the FO433MHz firmware. The Fine Offset protocol allovs for 16 different ID:s. That means more than one STC-1000+ or Fine Offset tempsensor can be used at the same time, provided they use different ID's.

Setpoint alarm, if setpoint alarm is greater than 0.0, then the alarm will sound once temperature differs from *SP* by more than *SA* degrees (this can be useful to warn against malfunctions, such as fridge door not closed or probe not attached to carboy). If *SA* is less than 0.0, then the alarm will sound if the temperature does **NOT** differ by more than (-) *SA* degrees (this could be used as an indication that wort has finally reached pitching temp). If *SA* is set to 0.0, the alarm will be disabled. If the alarm is tripped, then the buzzer will sound and the display will flash between temperature display and showing "SA", it will not however disengage the outputs and the unit will continue to work as normal. Please note, that care needs to be taken when running a profile (especially when not using ramping or with steep ramps) to allow for a sufficiently large margin, or the alarm could be tripped when setpoint changes.

Setpoint, well... The desired temperature to keep. The way STC-1000+ firmware works, setpoint is *always* the value the thermostat strives towards, even when running a profile. What the profile does is simply setting the setpoint at given times.

Current profile step and **current profile duration**, allows 'jumping' in the profile. Step and duration are updated automatically when running the profile, but can also be set manually at any time. Note that profile step and profile duration are the variables directly used to keep track of progress in a profile. Little or no validation is made of what values are entered. It is up to the user to know what he/she is doing by changing these values. Changing these values will not take effect until next point in profile is calculated, which could be as much as one hour. Every hour, current duration, *dh* (and if next step is reached, also current step, *St*) is updated with new value(s). That means in case of a power outage, STC-1000+ will pick up (to within the hour) from where it left off. Current profile step and current profile duration are only available in the menu when a profile is currently running.

Cooling and **heating delay** is the minimum 'off time' for each relay, to spare the compressor and relays from short cycling. If the the temperature is too high or too low, but the delay has not yet been met, the corresponding LED (heating/cooling) will blink, indicating that the controller is waiting to for the delay to pass before it will start heating or cooling. When the controller is powered on, the initial delay (for both heating and cooling) will **always** be approximately 1 minute, regardless of the settings. That is because even if your system could tolerate no heating or cooling delays during normal control (i.e. *cd* and/or *hd* set to zero), it would be undesirable for the relay to rapidly turn on and off in the event of a power outage causing mains power to fluctuate. Both cooling and heating delays are loaded when either cooling/heating relays switched off. So, for instance if you set cooling delay to 60 minutes and setpoint is reached, turning cooling relay off, it will be approximately one hour until cooling relay will be allowed to switch on again, even if you change your mind and change the setting in EEPROM (i.e. it will not affect the current cycle).

The delay can be used to prevent oscillation (hunting). For example, setting an appropriately long heating delay can prevent the heater coming on if the cooling cycle causes an undershoot that would otherwise cause heater to run. What is 'appropriate' depends on your setup.

Run mode, selecting *Pr0* to *Pr5* will start the corresponding profile running from step 0, duration 0. Selecting *th* will switch to thermostat mode, the last setpoint from the previously running profile will be retained as the current setpoint when switching from a profile to thermostat mode.

Thermostat mode

When mode is set to thermostat, setpoint, *SP*, will not change and the controller will aim to keep the temperature to within the range of $SP \pm hy$. Much like how the normal STC-1000 firmware works.

The thermostat control runs approximately once every second.

Running profiles

By entering the 'rn' submenu under settings and selecting a profile, the current duration, *dh*, and current step, *St*, is reset to zero and the initial setpoint for that profile, *SP0*, is loaded into *SP*. Even when running a profile, *SP* will always be the value the controller aims to keep. The profile simple updates *SP* during its course. When a profile is running the 'Set' LED on the display will be lit as an indication.

From the instant the profile is started a timer will be running, and every time that timer indicates that one hour has passed, current duration, *dh*, will be incremented. If and only if, it has reached the current step duration, *dhx*, current duration will be reset to zero and the current step, *St*, will be incremented and the next setpoint in the profile will be loaded into *SP*. Note that all this only happens on one hour marks after the profile is started.

So, what will happen if the profile data is updated while the profile is running? Well, if that point has not been reached the data will be used. For example profile is running step 3 (with the first step being step 0). Then *SP3* has already been loaded into *SP*, so changing *SP0 - SP3* will not have any effect on the current run. However, the duration *dh3* is still being evaluated every hour against the current duration, so changing it will have effect.

Changing the current duration, *dh*, and current step, *St*, will also have effect, but the change will not be immediate, only on the next one hour mark will these new values be used in the calculation. You will need to know what you are doing when changing these values manually, but correctly used, it could come in handy.

Changing the setpoint, *SP*, when running a profile, will have immediate effect (as it is used by thermostat control), but it will be overwritten by profile when it reaches a new step.

Once the profile reaches the final setpoint, *SP9*, or a duration of zero hours, it will switch over to thermostat mode and maintain the last known setpoint indefinitely.

Finally, to stop a running profile, simply switch to thermostat mode.

Ramping

The essence of ramping is to interpolate between the setpoints in a profile. This allows temperature changes to occur gradually instead of in steps.

Unfortunately, due to hardware limitations, true ramping (or true interpolation), is not feasible. So instead, an approximative approach is used.

Each step is divided into (at most) 64 substeps and on each substep, setpoint is updated by linear interpolation. The substeps only occur on one hour marks, so if the duration of the step is less than 64 hours, not all substeps will be used, if the duration is greater than 64 hours, setpoint will not be updated on every one hour mark, for example if duration is 192 hours (that is 8 days), setpoint will be updated every third hour).

Note, that in order to keep a constant temperature with ramping enabled, an extra setpoint with the same value will be needed (STC-1000+ will attempt to ramp between all setpoints, but if the setpoints are the same, then the setpoint will remain constant during the step).

You can think of the ramping as being true, even if this approximation is being used, the only caveat is, if you need a long ramp (over several days or weeks) and require it to be smoother. Then you may need to split it over several steps.

Another tip would be to try to design your profiles with ramping in mind, if possible (that is include the extra setpoints when keeping constant temperature is desired), even if you will not use ramping. That way, the profiles will work as expected even if ramping is enabled.

Power button is also the "back out button".
The S button is the set button and gets you into the relevant settings.
The up and down navigate the menu.