User manual

03-05-2017

Version 1.0

Company Information

Copyright and third-party information as required

Document Revisions

Date	Version Number	Document Changes
11/03/2017	1.0	First draft

Table of Contents

1Safety & Precaution........ 5

1SmartPID SPC1000 overview........ 6

1.1Scope and Purpose........ 6

1.2SmartPID overview........ 6

2SmartPID smart thermostat application........ 7

1.1Single PID control - heating........ 8

1.2Single PID control - heating and cooling (thermostatic)........ 9

1.3Dual ON/OFF control – cooling........ 9

3 Smart thermostat application SW........ 10

3.1Basic control principle........ 10

3.2Smart Thermostat application principle........ 12

4Menu navigation........ 14

4.1Main Menu........ 14

4.2Status Mode........ 15

4.3Configuration........ 15

4.3.1HW setup........ 16

4.3.2Unit Parameter........ 18

4.3.3Process parameter........ 19

4.3.4PID auto tuning........ 21

4.4Ramp/Soak........ 22

4.5Run mode........ 24

5Process Execution........ 25

6Appendix........ 27

6.1Zeigler-Nichols PID tune........ 27

6.2Data Logging structure........ 28

1 Safety & Precaution

Ensure that the product is always used within the specifications

Do not use product close to flammable and explosive gas otherwise injury from explosion may occur

Never disassemble, modify, or touch any of the internal part to avoid electric shock or malfunctions

Do not use the relay over their life cycle and do not exceed the rated load of the outputs

Do not touch the terminals at least while power is being supplied. Doing so may occasionally result in injury due to electric shock.

Do not allow pieces of metal, wire clippings, or fine metallic shaving or filings from installation to enter the product.

Do not allow water or any liquid enter the product. Enclosure is not water prof

The board is sold as a DIY standalone component and people buying should take care of connecting and integrating with their own system. The manula connection diagram and short explanations but minimum expertise in electric circuit is needed.

The board is powered by **High Voltage 220/110V** so you must be very careful and all connections are at your own risk. If you are not familiar with electricity and power please ask a technician to help you. I'm not responsible for any damage or risk you can create

1 SmartPID SPC1000 overview

1.1 Scope and Purpose

The purpose of this document is to describe in detail the user application software called **smart thermostat app**. The document provide the support to configure the application SW and run all different use cases

For the HW installation and basic configuration (including wifi) please refer to proper manuals.

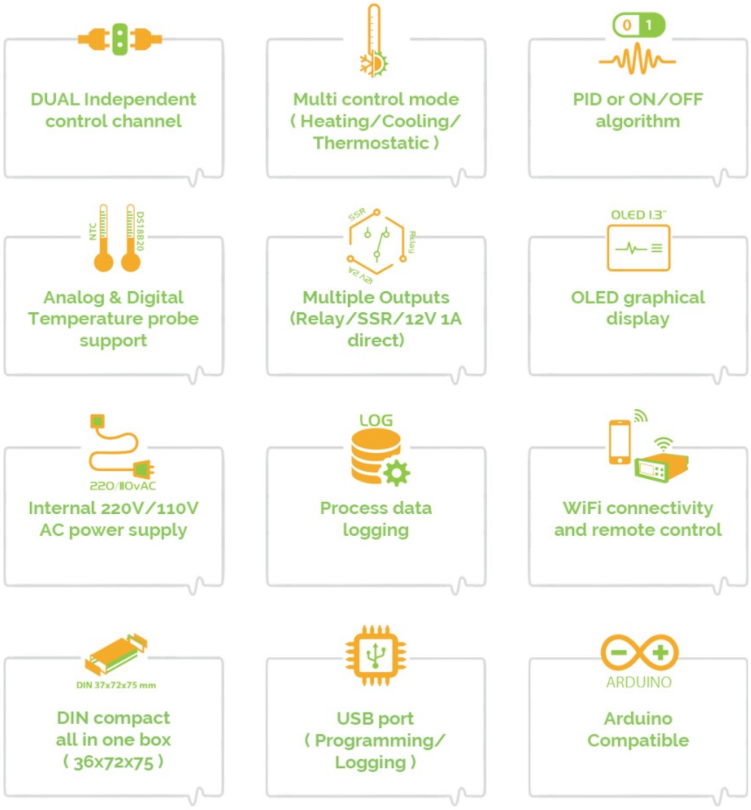
1.2 SmartPID overview

The smartPID controller has been designed to replace low cost simple thermoregulator with a smart controller that can be programmed and adapted to any process.

Using the Same DIN enclosure SmartPID microcontroller could replace almost plug an play the standard thermostat in any possible application

- process enhancement and full automation (heating, cooling,thermostatic)
- flexible programming (single set point or profile with ramp/soak definition)
- Higher accuracy (PID control)
- Wide variety of HW (5 different outputs)
- Remote control via dedicated smartphone app or web

In term of technical characteristic SmartPID support


DUAL Independent control channel
Multi control mode (Heating/Cooling/ Thermostatic)
0 1
PID or ON/OFF algorithm
NTC
DS18B20
Analog & Digital Temperature probe support
SSR
Relay
12V
Multiple Outputs (Relay/SSR/12V 1A direct)
OLED 1.3"
OLED graphical display
220/110vAC
Internal 220V/110V AC power supply
LOG
Process data logging
WiFi connectivity and remote control
DIN 37x72x75 mm
DIN compact all in one box (36x72x75)
USB port (Programming/ Logging)
ARDUINO
Arduino Compatible

- Dual Independent channels
- PID control and ON/OFF control with hysteresis
- 2x Relays output
- 1 SSR Output
- 2 x 12V 2A power output
- 2 Digital temperature probe input OR NTC input (configurable)
- OLED graphic display
- On Board EEPROM for data logging and parameter configuration/recipe store
- Process data logging
- SW update and loading via USB
- WiFi connection to remote server
- Buzzer events indication
- 4 push button interaction
- 220V power supply

The SmartPID controller is complemented by the **dedicated smartphone app** that allow the user to remote control the brewing process

For details on how to pair the smartPID with the app and configure wifi and thingspeak service please refer to installation and integration manual

2 SmartPID smart thermostat application

Via HW configuration SmartPID can be adapted to manage in a very flexible way different set up and solutions. The HW configuration allows to assign different processes/logical channels to different physical HW resources. This provide great flexibility to manage different heating source (gas or electric) and different cooling method (traditional fridge, TEC cooling systems)

3 main control mode are supported

a) **heating**: controller mange an heating systems to reach set point temperature and keep it stable

b) **cooling**: controller manage a cooling system to to reach set point temperature and keep it stable

c) **thermostatic**: controller activate either a cooling and heating systems to keep temparture at certain set point value regardless external temperature value

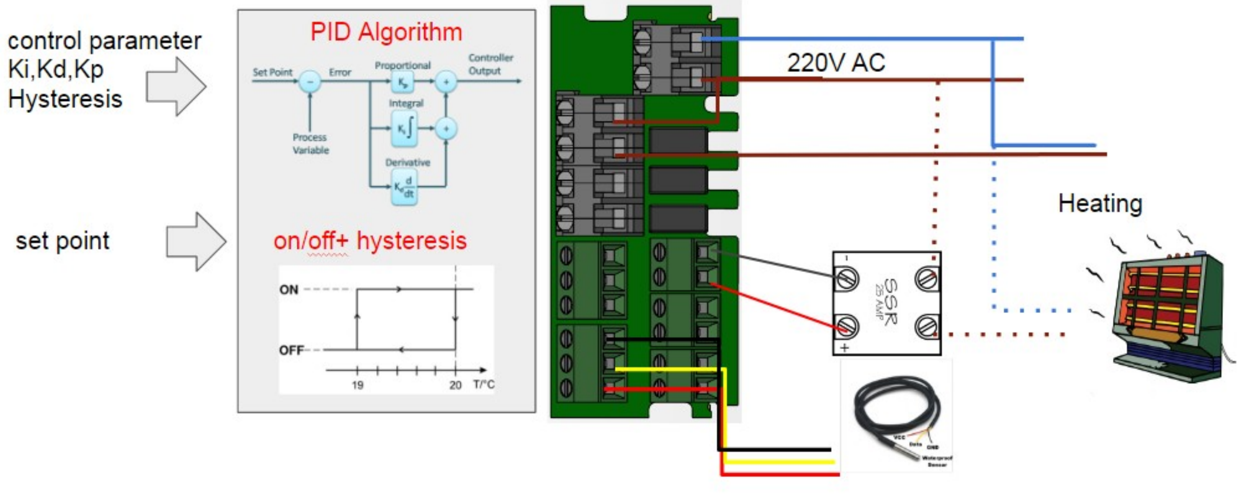
Below some example to illustrate the flexibility and multiple configurations possible

1.1 Single PID control - heating

SmartPID reads temperature from 1 probe and drive the heating element in order to reach the setup temperature

In the example DC1 out is connected to a solid state relay to drive power heating element and the heating power is controlled by PID-PWM algorithm 0-100%

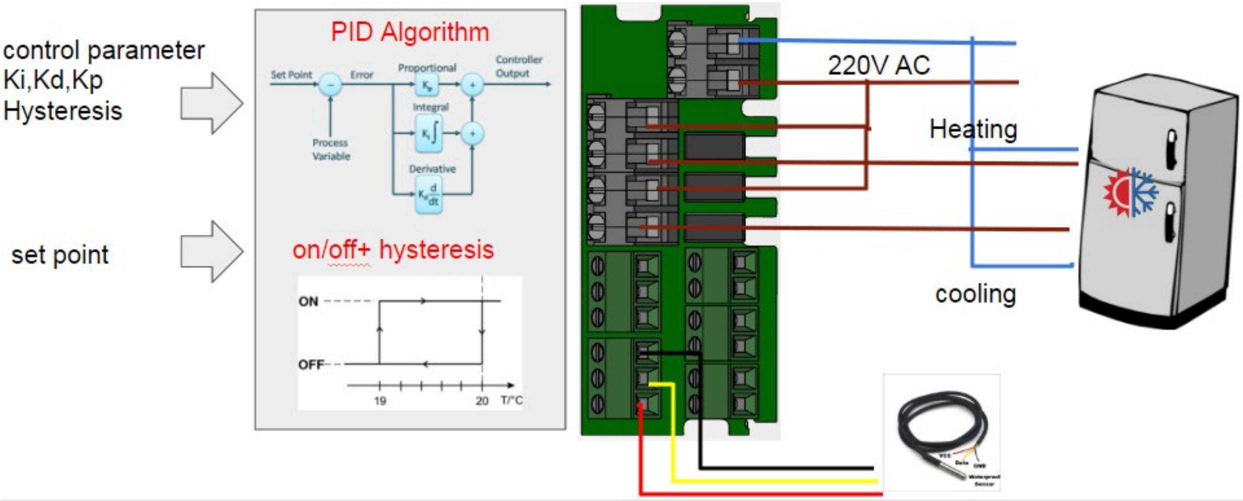
for load ap to 10A you can use one of the internal mechanical relay



1.2 Single PID control - heating and cooling (thermostatic)

SmartPID reads temperature from 1 probe and drive the heating element OR cooling element in order to get the setup temp with a thermostatic control

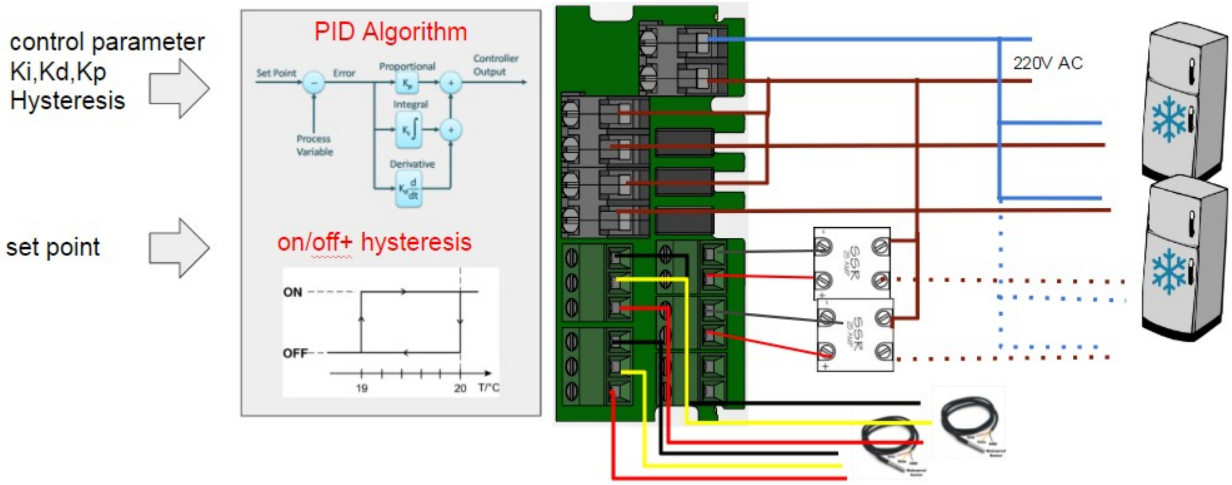
In the example 220V AC relay are used to drive heating element and cooling element (compressor)



1.3 Dual ON/OFF control – cooling

SmartPID reads temperature from 2 probes and drive 2 fridges independently in order to get the setup temperature

in the example 2x SSR are used to drive fridge compressors

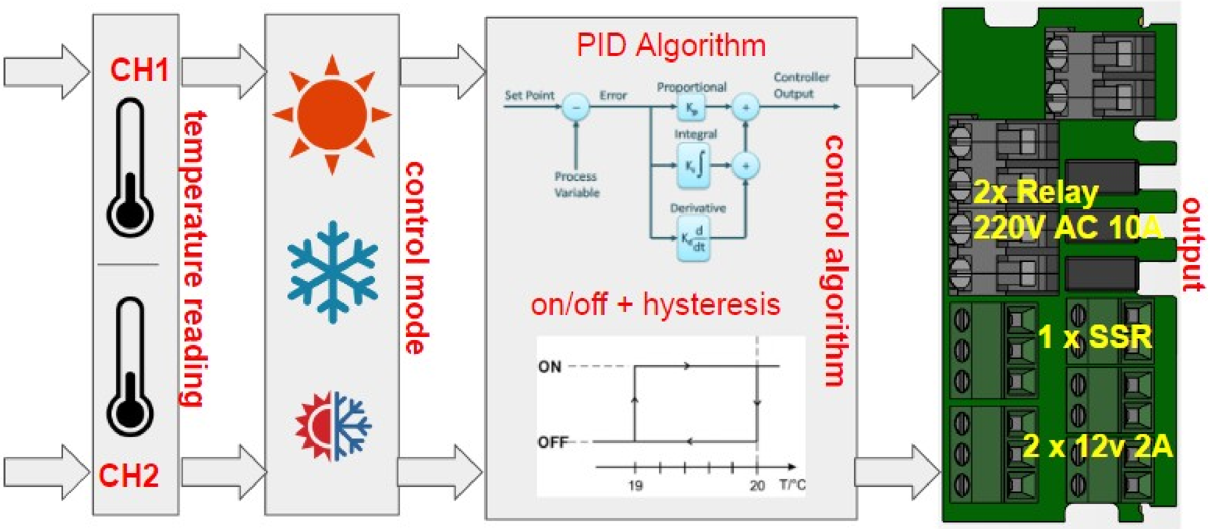


3 Smart thermostat application SW

In this chapter the application SW installed and running on the controller is described in order to provide to end user the full understanding about how the smart thermostat can be used and configured

3.1 Basic control principle

The core of this application is to implement a temperature control logic either to drive the heating element (electric or gas fired) in an heating process OR to drive a cooling system (typically a fridge) in a cooling process.



The main variable is the setpoint temperature selected by push buttons and this temperature is the input for the controlling algorithm that can be

1. **PID algorithm + PWM** control [electric heating element]

Set point is compared with current temperature and the Proportional Integrative Derivative calculation are performed.

Ki,Kp,Kd must be configured or an auto tuning process is be started during set up phase (optional)

PID output drive a PWM power control block in charge to vary the duty cycle of PWM signals from 0 to 100%. The PWM output will drive the load via SSR. In this way a fine control of electric heating element can be performed.

Two other relevant parameter can be configure and impact PID behavior

n) Window Size --> Period of the PWM signal

o) Sample Time --> PID sample time

2. **ON/OFF algorithm** [gas heating and fridge cooling]

In this case set point is compared with current temperature and depending if the value is over or under the target the output is put in ON or OFF state driving either the mechanical relay or SSR.

In order to avoid oscillations near the set point an hysteresis / dead zone should be implemented. The range of hysteresis should be configurable.

The ON/OFF control will topically be applied directly to heating systems gas fired or to fridge compressor

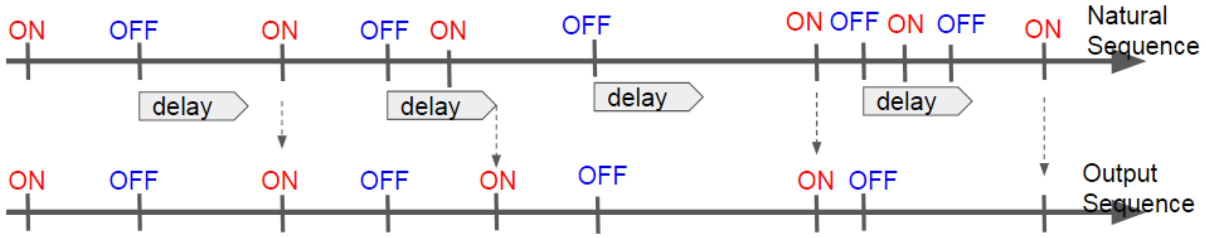
Fridge delay start

In order to avoid fridge compressor motor damage with frequent ON/OFF cycle a it's possible to insert a **fridge delay start**.

In the process parameter menu you can configure delay start parameter independent for CH1 and CH2 with a range value 0 - 240s (default value 0). When the delay is 0 the logic is not applied

When a transition ON → OFF occurs a timer is started that inhibit the ON transition of the fridge. In case of ON transition during delay window the output will be ON when the timers expires. In case of multiple transition during the delay window only the last transition is relevant, if is ON the output will be ON when timers expires if is OFF the output will be ON at the next natural ON.

Below some example on how the logic works



During the delay window the cooling mode icon on the OLED will blink indicating the suspension of action

3.2 Smart Thermostat application principle

The smart thermostat implement a complete temperature control logic with 3 control mode and provide also capability to define temperature profile over time.

Main functionalities are

1. **Status mode**

In this status smartPID acts like a monitor and no automatic action is performed over the 5 outputs. It's anyway possible to activate one of the outputs via menu with “soft switch” function

A timer is also started in this mode

2. **RUN mode standard**

this is the core mode of the thermostat and according to configuration (heating, cooling , thermostatic) the control process is started an temperature is sampled and compared to the set point in order to take proper action (PID or ON/OFF) on the configured output (relay, SSR, DC)

Once temperature reach the set point a timer and count down are started

In case of configuration of dual heating(cooling) mode both channels are controlled independently

3. **RUN mode advanced**

In this it should be possible to manage temperature profile by setting proper steps of time and temperature and control ramp slope and soak period. The profile can be associated to each of the two channels independently and can be used in any control mode (heating,cooling,thermostatic)

It's possible to configure the ramp/soak profile adding/editing/viewing the temperature and time parameters in a dedicate menu. Profiles are stored in the internal memory and are addressed by an index In run mode the proper profile index number is selected by the user for each channel. All normal editing operations on profile are available

For each step the profile is defined by a couple for Temperature and Time value and between each step a ramp time is defined and SP is increased linearly between the two set point with single increment of 1c / 40f. If ramp time =0 the SP1 is increased instantly.

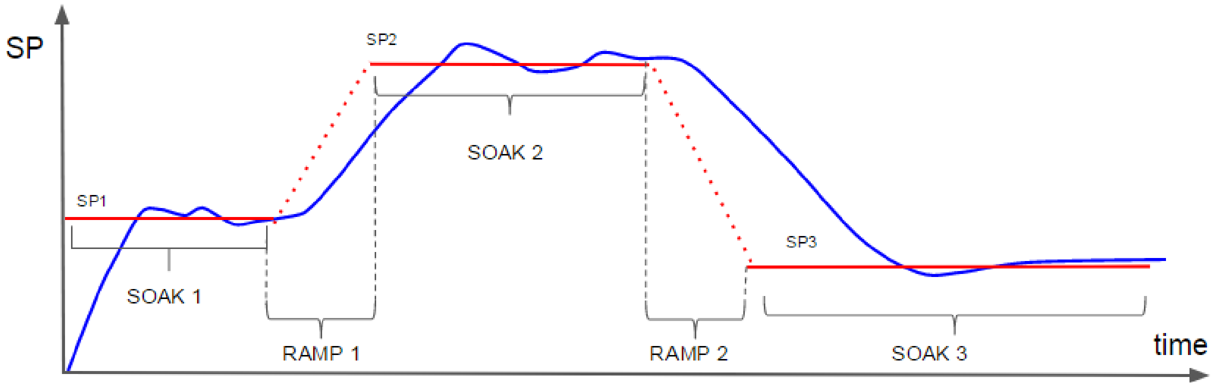
There are than two possible algorithm execution

Ramp/soak = STATIC

Soak time start when ramp time finish independently from temperature. Each soak period is predefined and is independently from the current temperature and set point.

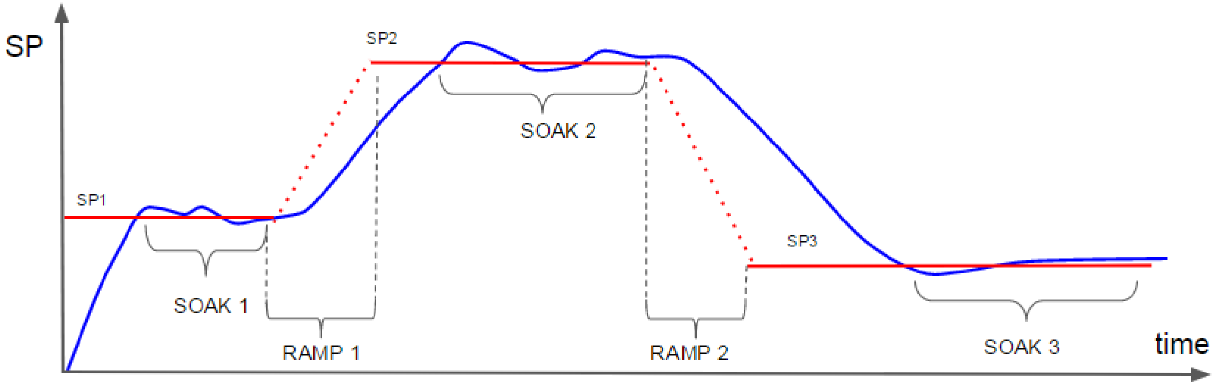
After each soak period a new set point is defined and a RAMP time defines the time needed to reach the new set point. SP is moved linearly from SPx to SPx+1 with increment of 1c/40f

If the next step hast soak time=0 (or is the eight steps) the SP is moved to that value and is kept constant until the process abort by user



Ramp/soak = DYNAMIC

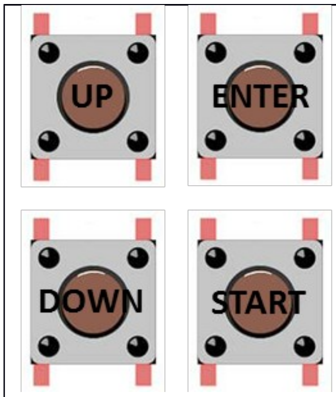
Soak time start when temperature reach the set point. When the temperature reached the SPx the soak period is started. After each soak period a new set point is defined and a RAMP time defines the time needed to reach the new set point. SP is moved linearly from SPx to SPx+1 with increment of 1c/40f. If the next step hast soak time=0 (or is the eight steps) the SP is moved to that value and is kept constant until the process abort by user.



4 Menu navigation

In following chapter the complete menu structure and navigation is reported, navigation trough menu and selection is performed via tactile switch on the front panel

Each button get its own functions and meaning in the proper context



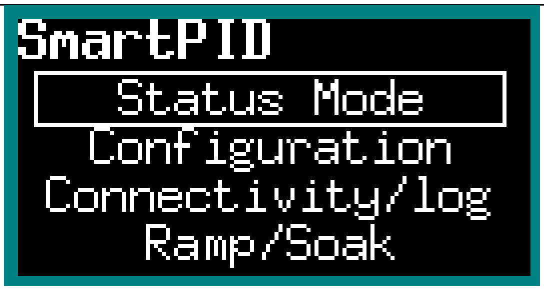
a) UP/DOWN
   a. Scroll in configuration menu
   b. Increase decrease temperature value
   c. Scroll in value in configuration menu

a) ENTER
   d. Select/enter a specific menu
   e. Select/confirm a specific value
   f. Confirm action upon prompt request

b) Start/Stop
   g. Start process
   h. Stop Process

4.1 Main Menu

After the boot smartPID enter in a standby mode with all outputs off and the user can select the main menu items



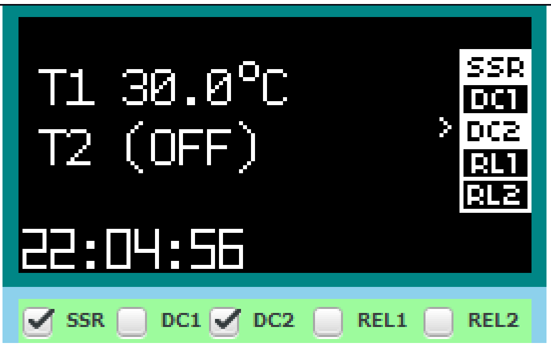
SmartPID
Status Mode
Configuration
Connectivity/log
Ramp/Soak

The top level menu corresponds to the main SmartPID functions

1. Status Mode --> it's a simple status where the temperature of the two channels is reported and user can activate / deactivate manually all the outputs (soft switch function)
2. Configuration Menu--> under this menu all the specific smartPID and process parameter configuration is performed
3. Connectivity and Log--> in this menu the user can configure all the wifi parameters, see the connection status, configure the remote server for the data logging. This functions are performed in cooperation with the smartphone app [See installation and configuration manual for details]

4. Ramp/Soak --> this menu allow the user to enter/edit/view the temperature profile with ramp/soak temperature and time definition
5. Run Mode--> trough this menu user start the process either standard or advanced

4.2 Status Mode

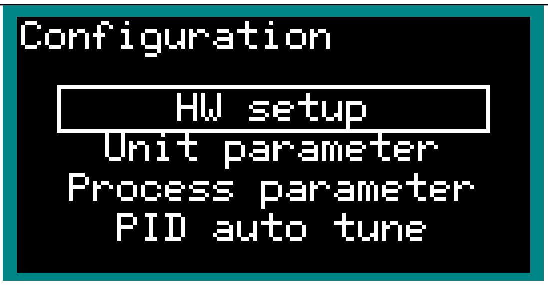


In the status mode the temperature of the T1 and T2 ports is reported as well as the time from the boot of the application. If no temperature probe is connected an error is reported

Moving with UP/DOWN button user can select any output and pressing SET the output can be activated and deactivated. On the OLED the relative label changes the color

To exit the status mode long press S/S button

4.3 Configuration

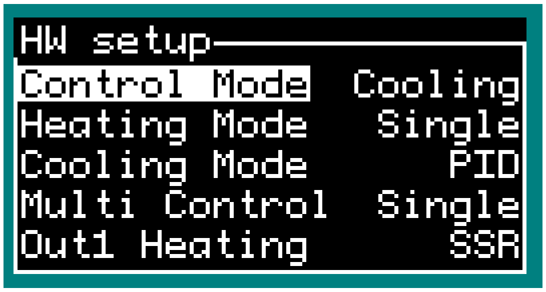
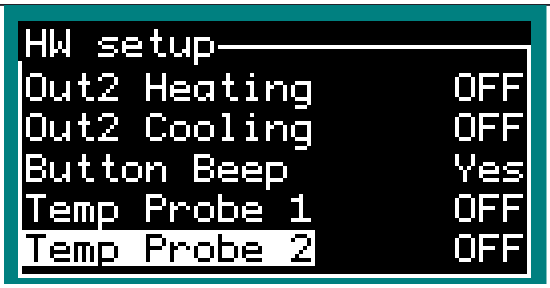
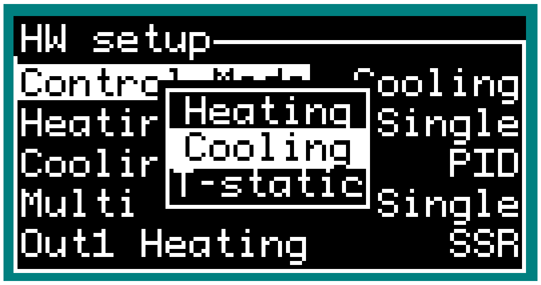
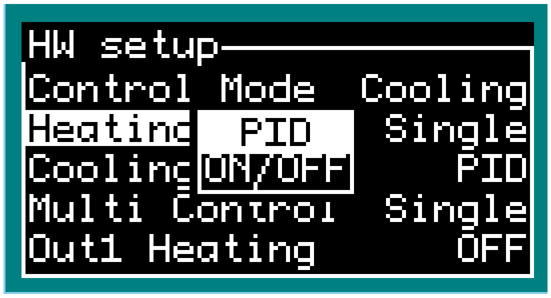
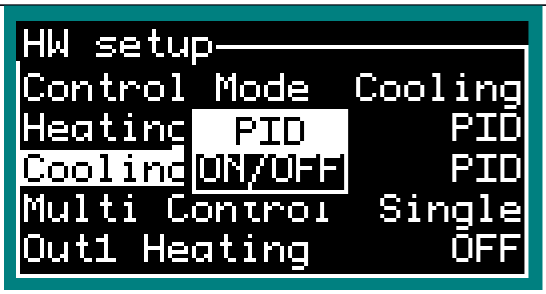


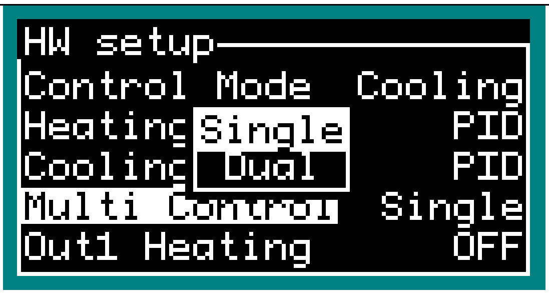
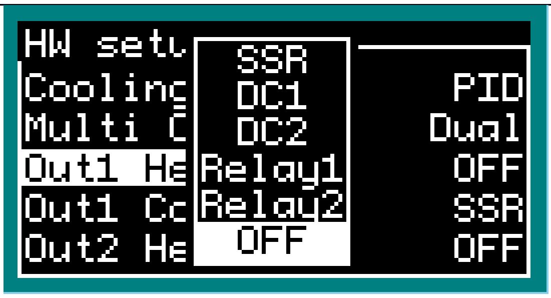
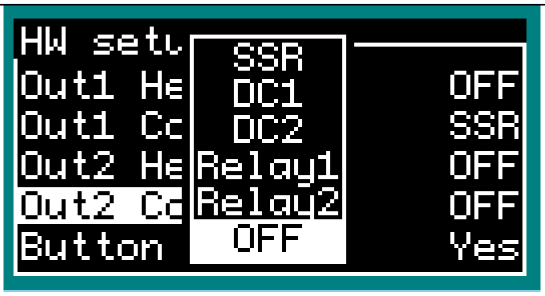
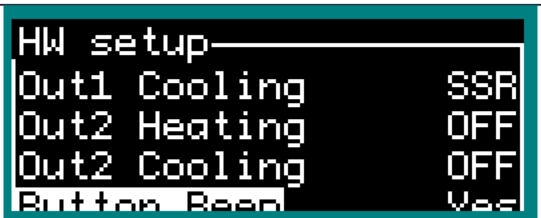
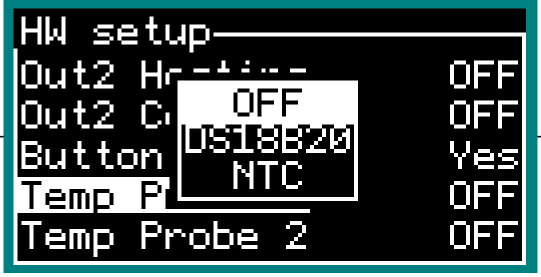
This menu is dedicated to all the configuration operations that typically are done once during the initialization / setup

1. HW setup --> the HW resources are configured and assigned to I/O process
2. Unit Parameter--> the main parameter that control the PID process behavior are configured in this menu
3. Process parameter--> smart thermostat process specific parameter are configured in this menu
4. PID auto tune--> this is a special section to configure and run the auto tune process to help end user to calculate critical parameters Kp,Ki,Kd that regulate PID behavior

4.3.1 HW setup

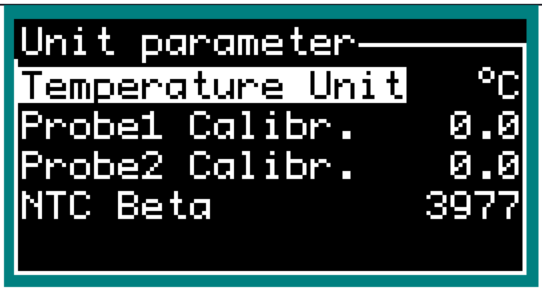
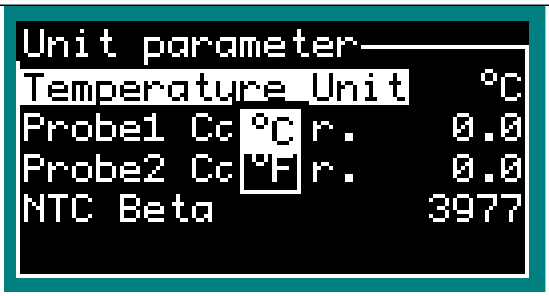
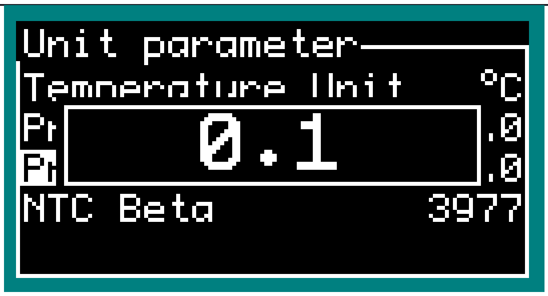
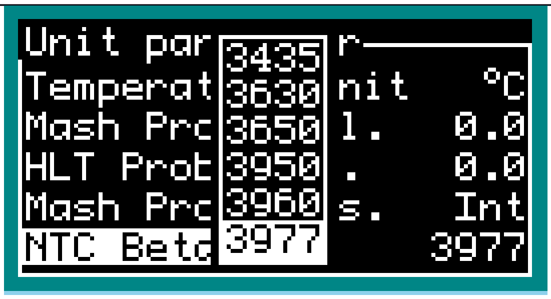
This section allow to configure physical resources and assign properly to process/channel

 HW setup Control Mode Cooling Heating Mode Single Cooling Mode PID Multi Control Single Out1 Heating SSR	 HW setup Out2 Heating OFF Out2 Cooling OFF Button Beep Yes Temp Probe 1 OFF Temp Probe 2 OFF
Control Mode User can select the control mode between -Heating -Cooling -Thermostatic	 HW setup Contro Cooling Heatir Heating Single Coolir Cooling PID Multi T-static Single Out1 Heating SSR
Heating mode User can select the control algorithm for the heating channel -PID -ON/OFF	 HW setup Control Mode Cooling Heatinc PID Single Coolinc ON/OFF PID Multi Control Single Out1 Heating OFF
Cooling Mode User can select the control algorithm for the cooling channel -PID -ON/OFF	 HW setup Control Mode Cooling Heatinc PID PID Coolinc ON/OFF PID Multi Control Single Out1 Heating OFF

Multi control Through this menu is possible to select if the thermostat will run a single control process or two process in parallel	 HW setup Control Mode Cooling Heating Single PID Cooling Dual PID Multi Control Single Out1 Heating OFF
OUT 1 (heating and cooling) through this menu is possible to assign physical output to the control channel 1 (either heating or cooling) Note that in order to avoid conflict the resources already assigned can't be selected	 HW setu SSR Cooling DC1 PID Multi C DC2 Dual Out1 He Relay1 OFF Out1 Cc Relay2 SSR Out2 He OFF OFF
OUT 2 (heating and cooling) through this menu is possible to assign physical output to the control channel 2 (either heating or cooling) if multi control = dual is selected Note that in order to avoid conflict the resources already assigned can't be selected	 HW setu SSR Out1 He DC1 OFF Out1 Cc DC2 SSR Out2 He Relay1 OFF Out2 Cc Relay2 OFF Button OFF Yes
Button Beep this flag enable or disable the sound upon button pressing	 HW setup Out1 Cooling SSR Out2 Heating OFF Out2 Cooling OFF Button Beep Yes
Temp probe 1 & 2 The proper temperature sensor is selected and assigned to channel 1 and 2 Sensor type can be different between channels	 HW setup Out2 H OFF OFF Out2 C DS18B20 OFF Button NTC Yes Temp P OFF Temp Probe 2 OFF

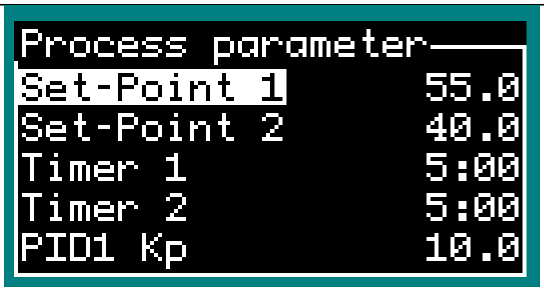
4.3.2 Unit Parameter

This section allow the user to configure global parameter that regulates the overall SmartpPID behavior

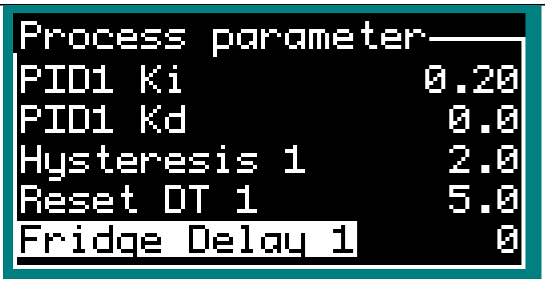
 Unit parameter Temperature Unit °C Probe1 Calibr. 0.0 Probe2 Calibr. 0.0 NTC Beta 3977	
Temperature unit This menu allow the user to select the proper measurement unit -Celsisus -Fahrenheit Same configuration should be reported on the smartphone app	 Unit parameter Temperature Unit °C Probe1 Cc °C r. 0.0 Probe2 Cc °F r. 0.0 NTC Beta 3977
Probe Calibration (1 & 2) It's possible to add an offset of +- 5c to the mash temperature probe reading to compensate any measurement error	 Unit parameter Temperature Unit °C Pr 0.1 .0 Pr .0 NTC Beta 3977
NTC beta User can select the Beta coefficient for NTC sensor , most common value are reported Beta will affect the temperature reading and so the precision, try to find the proper one according to spec of your NTC	 Unit par 3435 r Temperat 3630 nit °C Mash Prc 3650 l. 0.0 HLT Prot 3950 . 0.0 Mash Prc 3960 s. Int NTC Betc 3977 3977

4.3.3 Process parameter

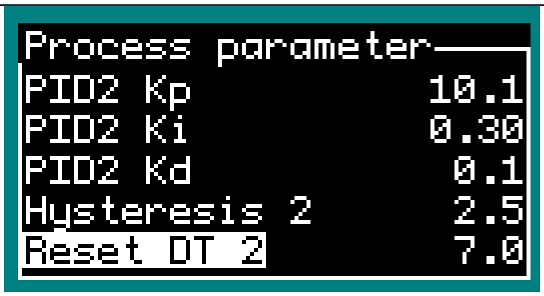
This section allow the user to configure all the parameter that are specific for the smart thermostat process, each of them influence the overall process execution



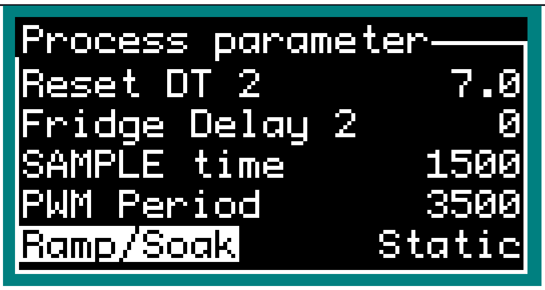
```
Process parameter
Set-Point 1         55.0
Set-Point 2         40.0
Timer 1             5:00
Timer 2             5:00
PID1 Kp             10.0
```



```
Process parameter
PID1 Ki             0.20
PID1 Kd              0.0
Hysteresis 1         2.0
Reset DT 1           5.0
Fridge Delay 1         0
```



```
Process parameter
PID2 Kp             10.1
PID2 Ki             0.30
PID2 Kd              0.1
Hysteresis 2         2.5
Reset DT 2           7.0
```



```
Process parameter
Reset DT 2           7.0
Fridge Delay 2         0
SAMPLE time         1500
PWM Period          3500
Ramp/Soak         Static
```

Below table reports for each parameter a short description and the range of valid value

Parameter Type	**Parameter**	**Description**	**Range**	**default**
Default parameter	Set-Point 1	Default set point for channel 1	-55 +120 c	55c
	Set-Point 1	Default set point for channel 2	-55 +120 c	40c
	Timer 1	Define default count down timer for channel 1		5min
	Timer 2	Define default count down timer for channel 2		5min
Process Parameter	Kp/Ki/Kd For CH1 and CH2	Define the proper constant value for the PID algorithm	Kp x—y Ki x—y	

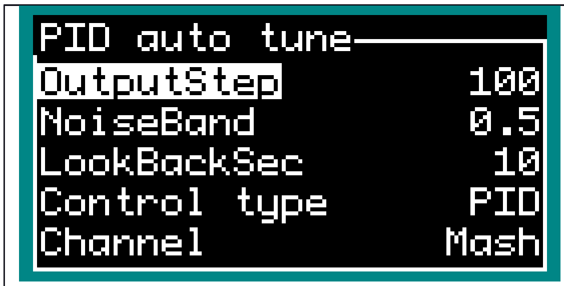
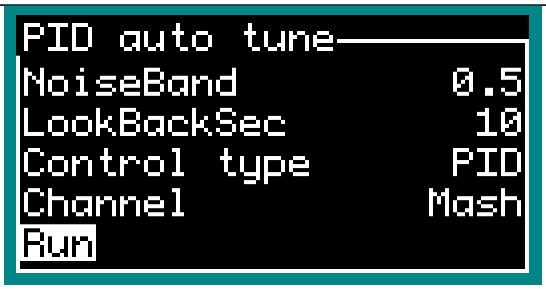
			Kd x—y	
	SAMPLE time	Define the sample time of temperature in ms	1000ms 4000ms	1500ms
	PWM perido	Define the PWM period/window size for PWM control in ms	500ms 7000ms	3500ms
	Hysteresis For CH1 and CH2	Define the hysteresis in ON/OFF control in C/F	0c 5c	2c
	Reset DT 1 & 2	Define temperature variation from set point to reset the count down timer	2 - 20c	7c
	Fridge delay 1 & 2	Define the duration of the fridge delay start 0 means no delay logic	0 - 240s	0
	Ramp/soak	Define if the control logic is dynamic or static (see proper manual section)	Dynamic Static	Static

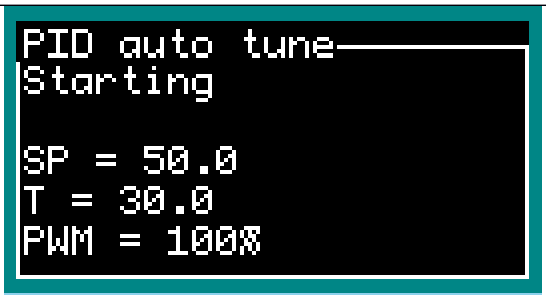
4.3.4 PID auto tuning

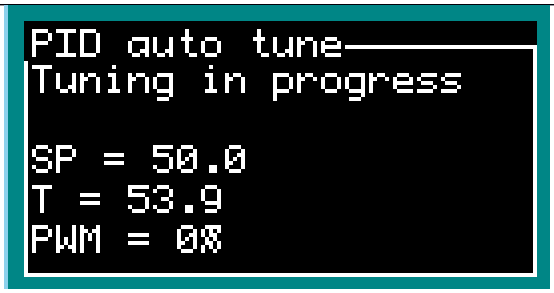
This is a special advanced menu that allow to start a specific SW algorithm that try to estimate the best value of KpKiKd constant considering the specific setup

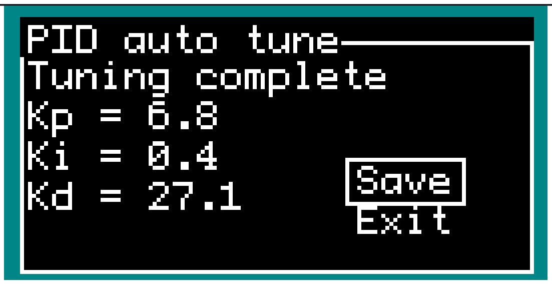
In this section is not reported the complete theory behind this algorithm but just how to configure and start . There are few resources on the internet that explain the Ziegler-Nichols method implemented, as resource starting point refers to Wikipedia page

https://en.wikipedia.org/wiki/PID_controller

 PID auto tune OutputStep 100 NoiseBand 0.5 LookBackSec 10 Control type PID Channel Mash	 PID auto tune NoiseBand 0.5 LookBackSec 10 Control type PID Channel Mash Run

PID auto tune RUN
Whene the autotune is in run mode the PID try first to reach the set point using the default parameters
	
	PID auto tune
	Starting
	SP = 50.0
	T = 30.0
	PWM = 100%

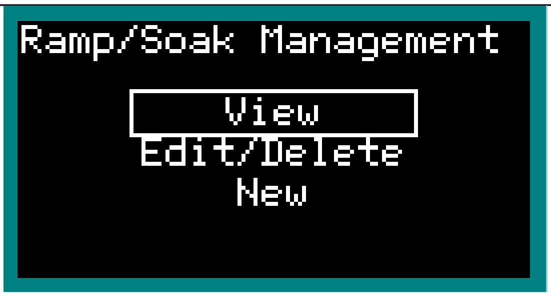
PID auto tune progress
Once the set point is reached the auto tune starts some step cycle with OutputStep increase and decrease calcualting the time to reach SP+-NoiseBand temperature value and evalaute system reaction
	
	PID auto tune
	Tuning in progress
	SP = 50.0
	T = 53.9
	PWM = 0%

PID auto tune complete If the alghoritm converge (it can take also one hour) the Kp,Ki,Kd calcualted are reported and user can save as default value Notice that the auto tuning process is per channel	 PID auto tune Tuning complete Kp = 6.8 Ki = 0.4 Kd = 27.1 Save Exit

4.4 Ramp/Soak

The ramp/soak feature is based on on a concept of **"profile"** that defines the input parameter to the controller

Through this section is possible to:



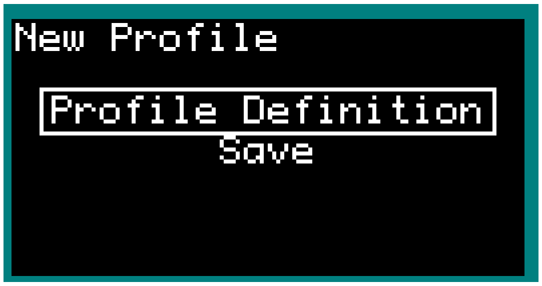
VIEW→ load profile from local EEPROM selecting index number and allow to display the full set of parameter in each sub-menu

EDIT /DELETE→ load profile from local EEPROM selecting index number and allow to change/modify/delete all parameter in each sub menu. At the end of the modification user can save on the same index (replace) or with a new index (clone).

NEW → allow user to define a new profile from scratch and add all parameters divided in two submenu. For parameter list see the doc spec chatper 3.3.3

At the end of the profile definition the entire profile can be saved in the EEPROM with a proper index

When a new profile should be inserted a profile definition menu is presented



Profile definition

For each point of the profile user should define:

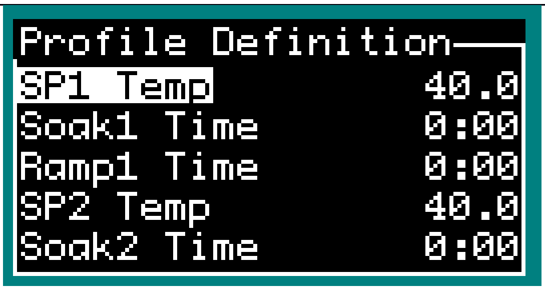
SP X that is the starting set point temperature

Ramp Time Y that is the time to move set point from X to X+1

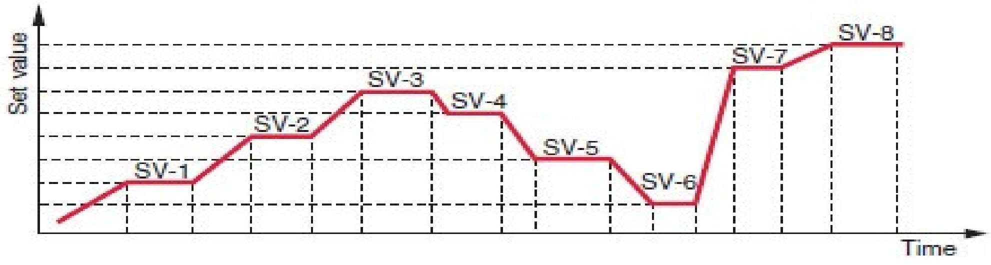
Soak time Z that is the time the temperature is kept constant

If the ramp time = 0 the set point is moved instantly (step)

If the next step has soak time=0 (or is the eight steps) the SP is moved to that value and is kept constant until the process abort by user



```
Profile Definition
SP1 Temp        40.0
Soak1 Time      0:00
Ramp1 Time      0:00
SP2 Temp        40.0
Soak2 Time      0:00
```



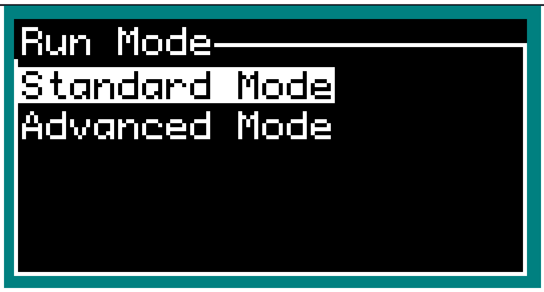
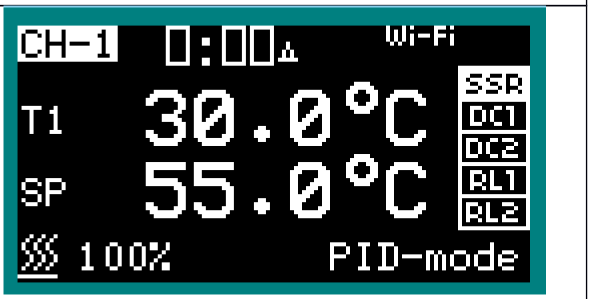
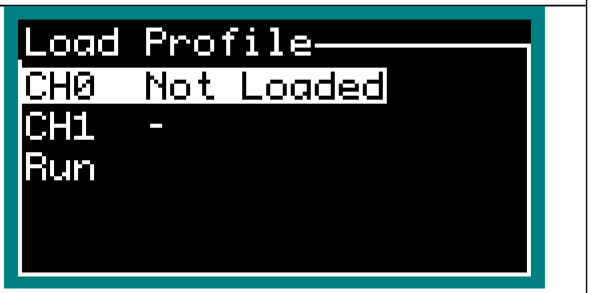
4.5 Run mode

Entering the run mode allow the user to start the control process

Two possible run mode are possible

Standard Mode--> simple temperature controller according to control mode configured

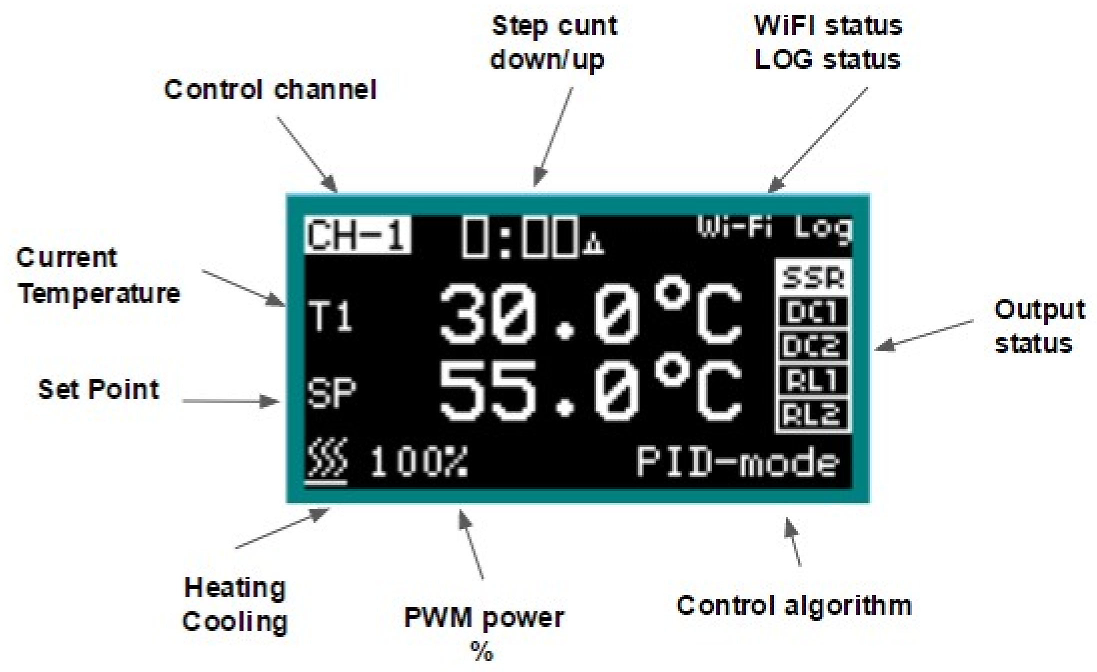
Advanced Mode -→set point temperature is controlled according profile (ramp/soak) definition

 Run Mode Standard Mode Advanced Mode	
Standard Mode The system enter the run mode and starts controlling the the CH1 and CH2 See "process execution" chapter for details of the information reported on the display	 CH-1 0:00 Wi-Fi T1 30.0°C SP 55.0°C SSR DC1 DC2 RL1 RL2 100% PID-mode
Advanced Mode Before entering the advanced mode user is requested to load a profile from the memory or define a new one. Profile can be different for channel 1 and 2	 Load Profile CH0 Not Loaded CH1 - Run

5 Process Execution

Once the controller has started the run mode either standard or advanced the temperature is sampled and the control logi is applied according to parameter configuration (PID or ON/OFF).

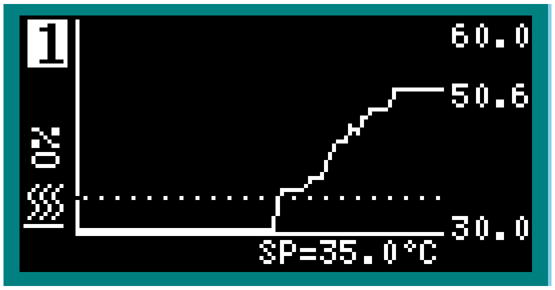
The main screen in run mode report following information



via the push button is possible to

- Switch to the graphic mode --> press SET
- Switch between count UP and count DOWN --> press S/S
- Modify the set point temperature --> UP/DOWN buttons
- Exit the process --> long press S/S button

The graphic mode reports the real time temperature and the set point, graph is automatically updated and scalded

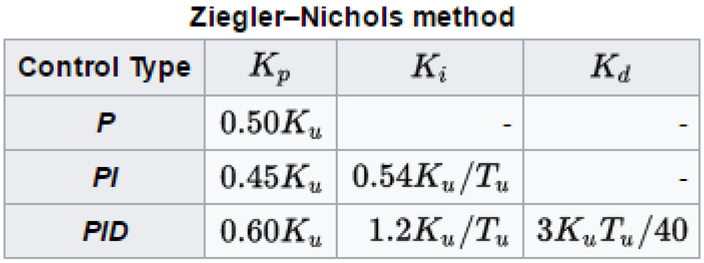


Pressing SET button it’s possible to move to the channel 2 screen (if dual mode has been configured)

6 Appendix

6.1 Zeigler-Nichols PID tune

Another heuristic tuning method is formally known as the Ziegler–Nichols method, introduced by John G. Ziegler and Nathaniel B. Nichols in the 1940s. As in the method above, the Ki and Kd gains are first set to zero. The proportional gain is increased until it reaches the ultimate Ku, a which the output of the loop starts to oscillate. Tu and the oscillation period are used to set the gains as shown:



Ziegler–Nichols method

Control Type	K_p	K_i	K_d
P	$0.50K_u$	-	-
PI	$0.45K_u$	$0.54K_u/T_u$	-
PID	$0.60K_u$	$1.2K_u/T_u$	$3K_uT_u/40$

These gains apply to the ideal, parallel form of the PID controller. When applied to the standard PI form, the integral and derivative time parameters Ti and Td are only dependent on the oscillation period Tu

6.2 Data Logging structure

The main process parameter are logged either on the EEPROM or pushed to the external server via wifi (currently using the thingspeak service)

The information stored are the following for each channel

a) time stamp (format Epoch, ISO 8601, MySQL)
b) channel number (CH1 or CH2)
c) control mode (heating or cooling)
d) heating or cooling mode (PID or ON-OFF)
e) set point temperature
f) current temperature
g) PWM percentage
h) PID direction (cooling/heating or ON/OFF)