Crystal Chamber

Documentation

Version 1.2.J

2022 - 2025

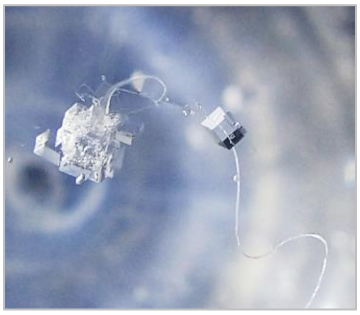
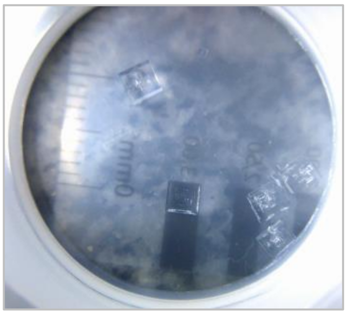
Crystal Chamber Project, a thermodynamic control chamber controlled by Arduino for nucleation and crystal growth from saturated saline solutions by evaporation method.

Designed and built by Vinícius Coelho (Voelho)

Main sketch version: Crystal Chamber J – 17/06/2025
Thermostat sketch version: Dismissed.

SUMMARY

mrvoelho@gmail.com

1. COMPONENTS

Item	Qtd
Arduino UNO	1
400-point breadboard	1
12V 10A Power supply	1
XL4015 step-down module	2
Fan + coolers	2
Peltier element 12705	1
SG90 microservo 0 - 180°	2
SG90 continuous microservo*	1

Item	Qtd
DHT22 sensor	3
Mini fan 4 x 4 cm 5v 0,14A	1
Relay module	3
MicroSD module	1
128 x 64 OLED display	1
Buzzer	1
Buttons, cables and resistors	-

* SG90 servo adapted for continuous rotation.

2. LIBRARIES

Default operation

Arduino.h		Basic library
DHT.h	1.4.6 Adafruit	DHT reading
PinButton.h	Multibutton 1.2.0 Poelstra	Button input control
SD.h		SD card logging
SPI.h		Serial communication (for debug)
U8x8lib.h	U8g2 - 2.34.22 Oliver	Display with low memory demand
VarSpeedServo.h	Version 2 – 2013 Philip van Allen	Servo control

Alternative

HX711.h	HX711 0.7.5 Bodgan	Load cell interaction for scale usage

3. PINS

Default operation

Pin	Connection	pinMode
2	DHT 2	-
3	RPM for ext. fan	-
4	TEC	output
5	DHT 1	-
6	DHT 3	-
7	Buzzer	-
8	SD - CS	-
9	Servo 1	-

Pin	Connection	pinMode
10	Inner fan	output
11	SD – MOSI	-
12	SD – MISO	-
13	SD – SCK	-
A0	Voltmeter	output
A1	BT Up	input
A2	BT Mid	input
A3	Servo 2 - Thermostat	-

4. ALGORITHM

This section describes the operation of the control algorithm in the execution sequence of the programs in a comprehensive manner. More details are documented in the Sketch itself.

4.1. Operation states

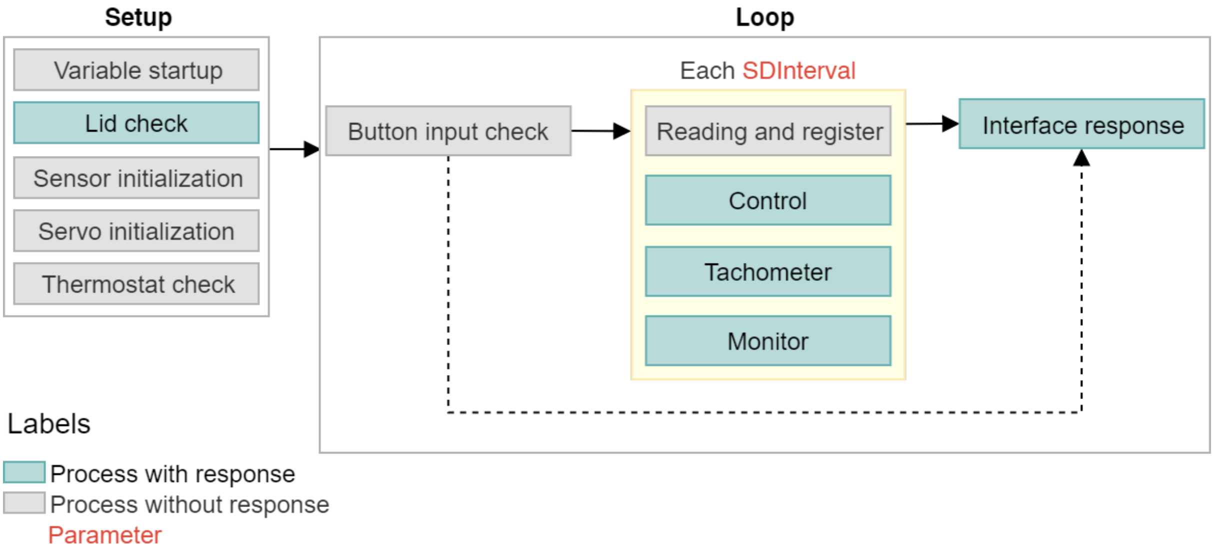
There are 5 operation modes, for different exposure conditions or adjustments

Mode	OPE	Indicator	Description
Maintenance	0	MAN	Mode where the peripherals are turned off for the physical operation of the chamber in case of maintenance, observation, or adjustments. Useful for monitoring crystallization conditions, as access to the humidity chamber is required.
T control	1	T:	First stage of the operating mode, adjusting the temperature range for later humidity adjustment.
U control	2	U:	Second stage of the operating mode, allows humidity control through drying, but only within the established temperature range.
Temperature only	3	T_	Controls only the temperature within the determined range. Useful for sensor calibration
Ventilation	4	__	Ventilation mode does not activate the thermal or drying control. It is useful for sensor calibration or exposing the solution to a temperature close to ambient.
Humidity only	5	_U	Controls only the humidity within a specified range. It varies depending on the behavior of the DHT sensors at different temperature ranges. Useful for sensor calibration.
Clear		CL	Option to ignore an error state and continue the operation (Forces *MoSt = 0* without resetting)

4.2. Default exposition

The system operates by checking the operating conditions at each *SDInterval*, adjusting the controls and performing safety checks. The initialization assumes operational autonomy in the event of interruptions or power outages, so several verification processes are given execution priority.

Diagram 1 – Algorithm overview



4.2.1. Setup

The thermal control is initialized as on.

The lid verification also prioritizes operational safety, ensuring that the fan on the hot sink of the TEC (the Peltier element) is working properly to prevent overheating. If there is a reading on the tachometer, then the box is closed, and the external lid is connected to the chamber. The operating modes with thermal control require the lid to be connected, while the calibration mode requires the lid to be open, to prevent overheating from the internal fan of the humidity chamber.

The servo controlling the humidity moves for a visual check of its operation, as it is crucial for humidity control.

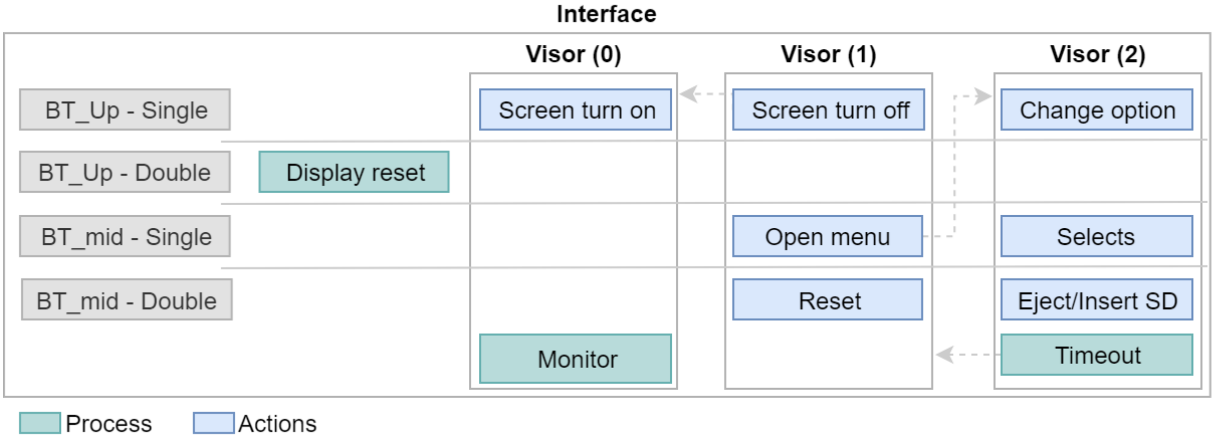
4.2.2. Button operation

The interaction of the buttons is checked every Loop cycle. If there is activity, the response will depend on the display status (variable *Visor*). A double-click on the *BT_Mid* button during *Visor 1* calls the function *retomaOp*, to force a reset on the readings and the *Monitor* state.

A double-click on the *BT_Up* button invokes a display reset, as sometimes the connection to the Arduino can be corrupted by loose wires or some movement in the LCD display terminals.

The variable *intInterval* controls the timeout period of the menu, returning to the operation screen.

Diagram 2 – Interface actions.

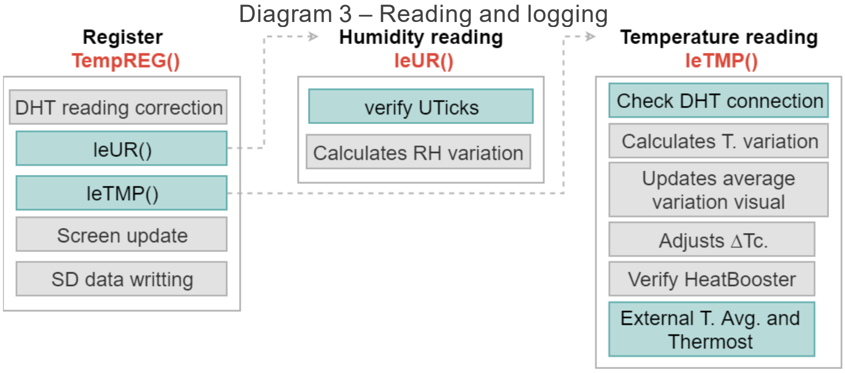


4.2.3. Reading and Logging

Temperature and humidity readings are performed by the function *TempReg()*. At this stage, the calibration equations for the DHT sensors are applied, and the data is displayed if *Visor = 1*. No reading or logging is performed in maintenance state (Topic 4.1).

If the SD card is connected (*vSD = true*), it is checked if it is time to register the information (*SDticks = 0*); if not, the skipping of the interval is recorded. The delay prevents the writing of multiple data entries within a relatively short interval (every *SDInterval*), which would produce a long file with unnecessary information.

Diagram 3 – Reading and logging



Humidity reading (leUR function)

The humidity reading is also delayed (*Uticks*) to reduce the influence of momentary fluctuations caused by small temperature variations or operational conditions.

Temperature reading (leTMP function)

Faulty wire connections or loss of connection with the DHT sensor in the exposure environment (indicated to the monitor with *MoSt = 7*) are verified. Instant readings are also performed, along with the updating of moving averages and the visual indicator of variation (item 6.1).

This function also calls the adjusts of thermostat correction factor. Sometimes, the calibration curve is not yet aligned with the operating conditions, and deviations from the desired temperature (*Talvo*) may require adjustment. In this case, *DT* (temperature difference between external and internal environment) is compensated with a value *DTc*, equal to the last adjustment value plus the difference between the stabilized temperature and *Talvo*. This adjustment is made every *varInterval* according to the actual condition (if *DTc < 0* and the temperature is rising; if *DTc > 0* and the temperature is falling or when the temperature is stable).

The variation in *DTc* is reduced to 1/3 of the original value if the HeatBooster (Item 4.3.1) is on (*HB = true*), to prevent the temporary power boost from being quickly undone by the adjustment. In a scenario where the ambient temperature is considerably close to the target temperature, the voltage adjustment sets the system to minimum power, thus reducing the *DTc* value to zero. This avoids an impossible attempt to further reduce the cooling power (which would impact future stages of operation) or at least the booster value, if active.

If the HeatBooster is on (*HB = true*), it's verified whether the external temperature has stopped increasing during the last two *varInterval* intervals. If the moving average has remained constant or decreased, it is assumed that the HeatBooster can be turned off.

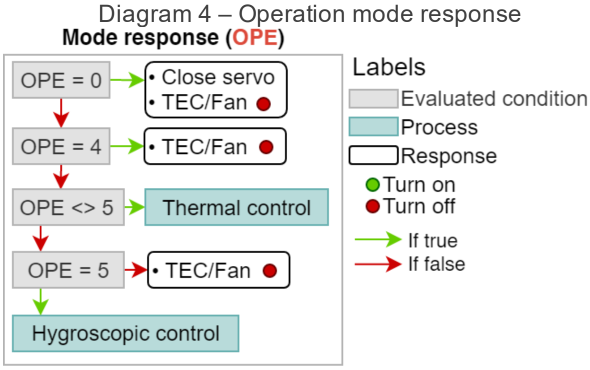
4.2.4. Control

Default operation

The response of the control system depends on the current operating mode, variable *OPE* (Item 5.1), with priority given to maintenance mode, which is also used to interrupt operation in case of any safety condition violation.

Safety conditions are checked by the *Monitor()* function, aiming to prevent scenarios that could be unfavorable to evaporation or crystal growth stability (e.g. very low relative humidity).

Diagram 4 – Operation mode response



Thermal control

Given the dependence of relative humidity on temperature, the aim is to stabilize the exposure within the target range before starting the humidity control. A moving average (*T.méd*) is used as a reference for monitoring to minimize the influence of sensor fluctuations.

The events marked in the diagram are illustrated in the temperature scenario charts with the possible behaviors of the average temperature. Given the assumption of constant temperature adjustment, only a critical error signaled by the *Monitor()* will deactivate the humidity control mode once it has been activated.

Diagram 5 – Thermal control

Temperature control

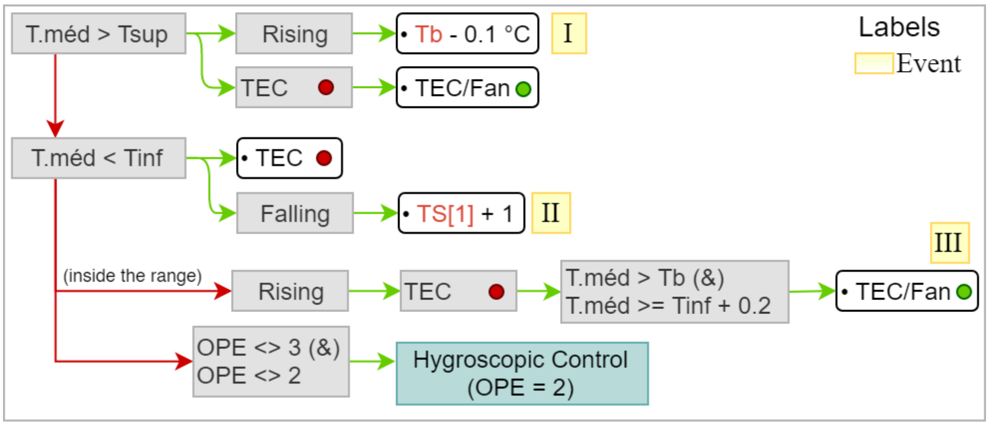
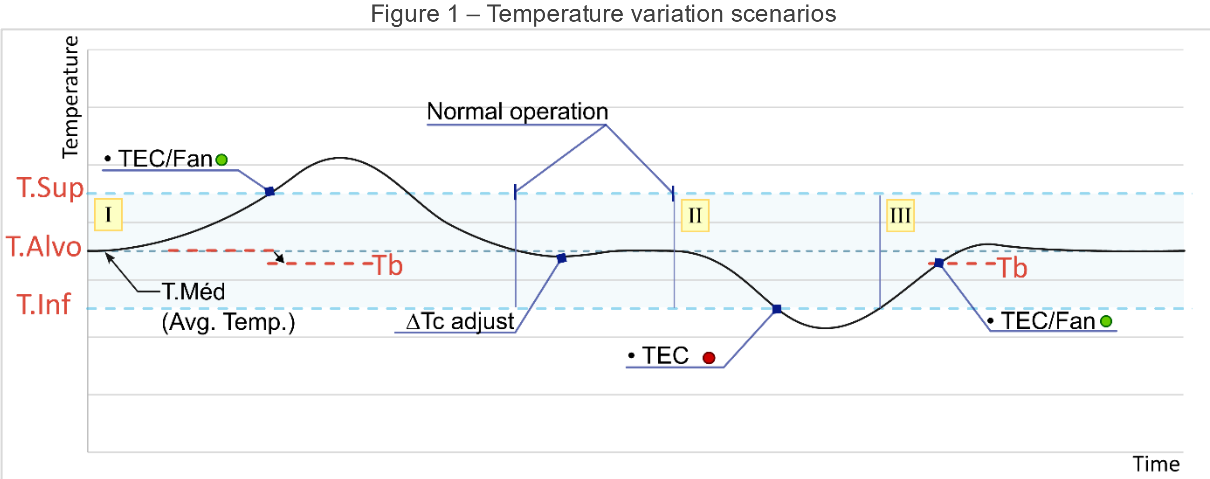


Figure 1 – Temperature variation scenarios

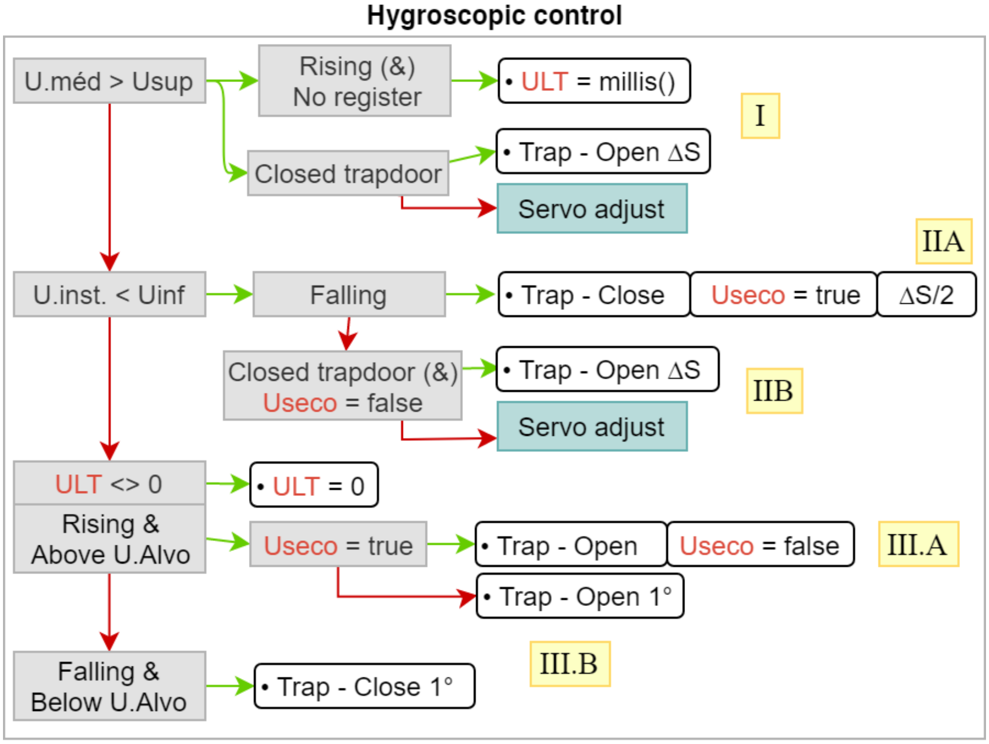


Hygroscopic control

The hygroscopic control regime seeks to adjust the servo's opening to bring the chamber's relative humidity closer to the target value (*Ualvo*). The equilibrium value of the exposure depends on the evaporation of the saline solutions and the adsorption of silica gel. The first is a function of the exposure area, the amount of solution and its concentration, and the relative humidity itself, while the second also depends on the silica gel's saturation level - interrelated and complex parameters, also dependent on temperature. Therefore, equilibrium is dynamically achieved by observing how a certain trapdoor opening alters the chamber's humidity and adjusting it to maintain the expected levels.

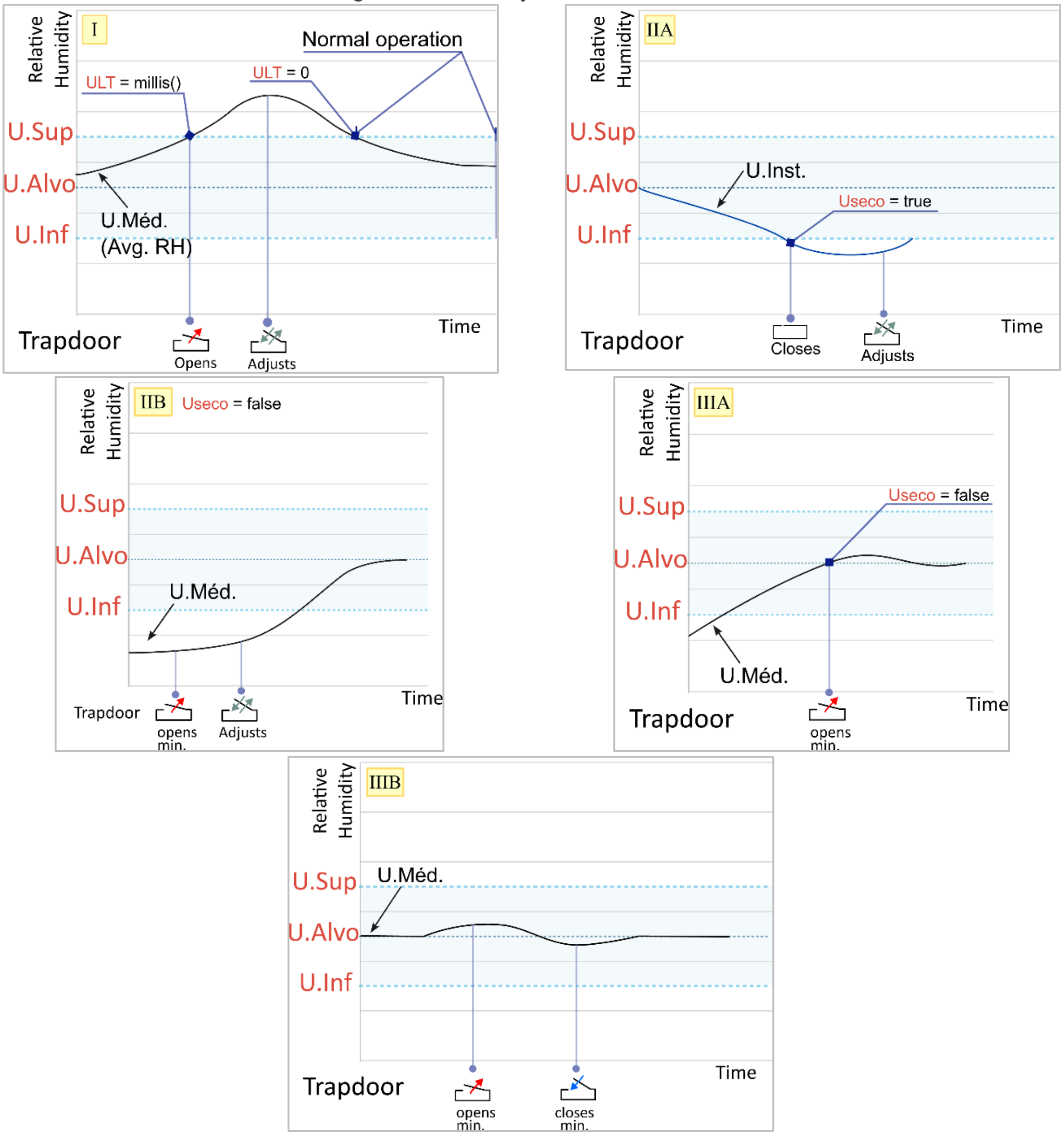
In summary, the control is based on the adjustment of drying intensity. Increases in relative humidity are due to the evaporation of the solution, mainly when the *trapdoor* is at the minimum position.

Diagram 6 – Hygroscopic control



ΔS = servo variation in degrees

Figura 2 – Humidity variation scenarios



4.2.5. Monitoring

Safety and stability checks are carried out through the *Monitor()* function every *SDInterval* if the screen is on, and every cycle of the *Loop* function if the screen is off. Any violations of the verified situations are reported as errors via the *MoSt* variable (acronym for Monitor Status).

If an alert is triggered, the display is turned on and an alert tone is emitted by the buzzer.

MoSt	Situation
0	System in maintenance, readings are interrupted, and components are deactivated, allowing for the removal or modification of components without corrupting the readings or operation
1	System returning from maintenance mode, tolerance is allowed for some temperature-related checks, such as the one that triggers *MoSt = 6*
2	Obsolete in version 1.2.J
3	Humidity is above the upper limit for more than two hours. Possible saturation of the silica gel to a level where it no longer adsorbs sufficiently, signaling the need for replacement.
4	More than 4 occurrences of average temperature below *Tinf*. Possible situation of excessive cooling power.
5	External fan RPM lower than the threshold for 1 minute. Possible failure in the connections or the fan, which could cause the system to overheat or lose efficiency. This is a critical situation that triggers maintenance.
6	Internal average temperature 2 °C above *Tsup* and rising. Possible component failure, critical situation that triggers maintenance.
7	The temperature reading is equal to zero. Possible loss of connection with the sensor or its failure. May trigger maintenance if associated with the condition *MoSt = 4*.

4.2.6. Operational notes

retomaOp

This is the function that resets the operation, clearing most control parameters.
It's called after a maintenance state or by manual reset through double-click on *BT_Mid*

In modes 4 and 5, the inner fan is only turned on if the lid is absent

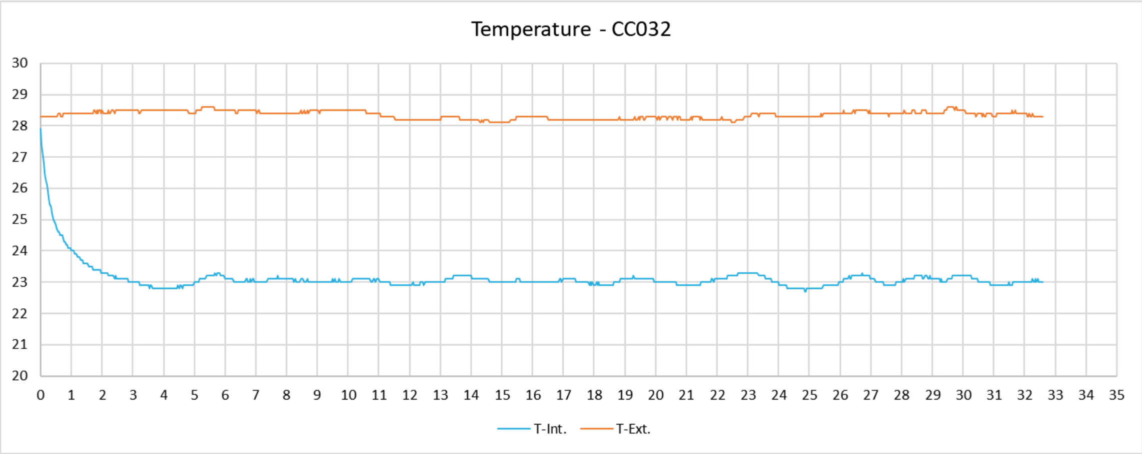
The SD card performs a buffer flush every *SDticks* cycles to ensure that data is written and not lost in case of a power failure or other interruption.
The interface with the card is only completed with the **operaSD** function, which is responsible for finalizing and reinitializing the card with the system, allowing for temporary removal during operation.

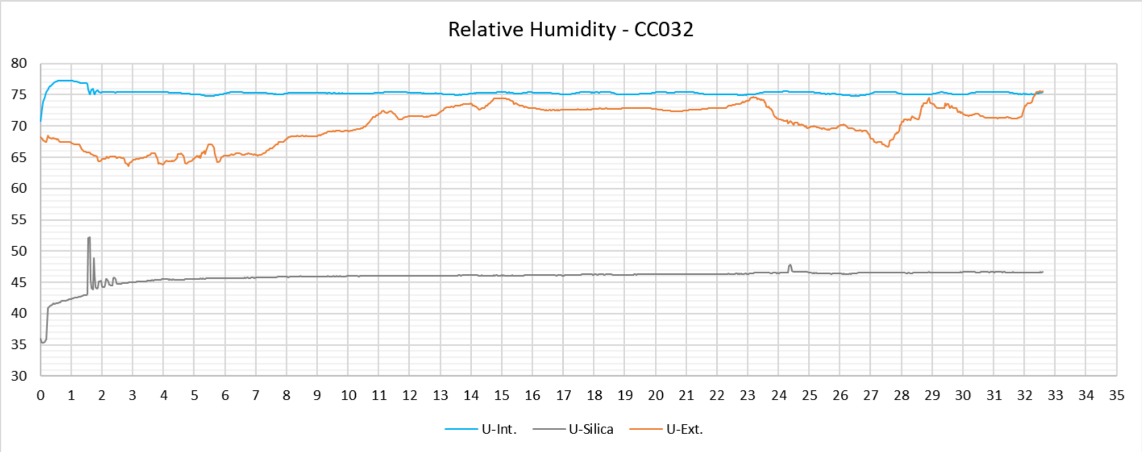
The servo adjustment function (**ajServo**) includes a damping system to prevent abrupt humidity variations that could compromise crystal formation, also ensuring that the variation is not too small when far from the target range, otherwise the exposure would take too long.
The limit values are relative to the difference between the average humidity and the target range during the last recorded check every *Uticks* cycle. By default, the control ranges are: above 5%, between 2% and 5%, and below 2%, with the maximum allowed variations being: 1%, 0.5%, and 0.2%, respectively.

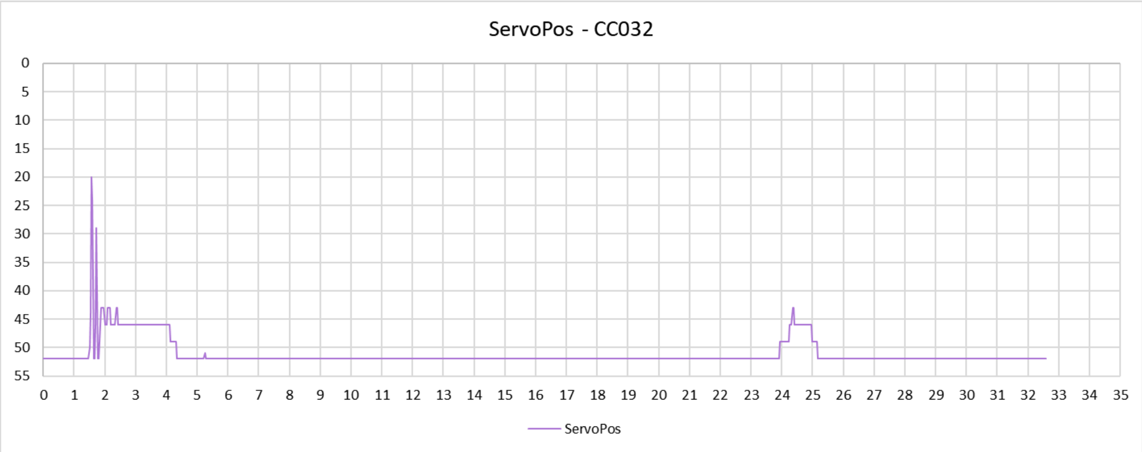
4.2.7. Typical behavior

The charts below show the typical behavior of chamber conditions. Exposure was set to *Talvo* of 23,0 °C and *Ualvo* of 75.4%. After reaching equilibrium, the internal temperature oscillated by ± 0,3°C while relative humidity remained around 75.2%. It's interesting to observe that the most expressive variations occur smoothly after the equilibrium point, ensuring that the exposure environment does not undergo thermodynamic shocks that could compromise crystal formation.

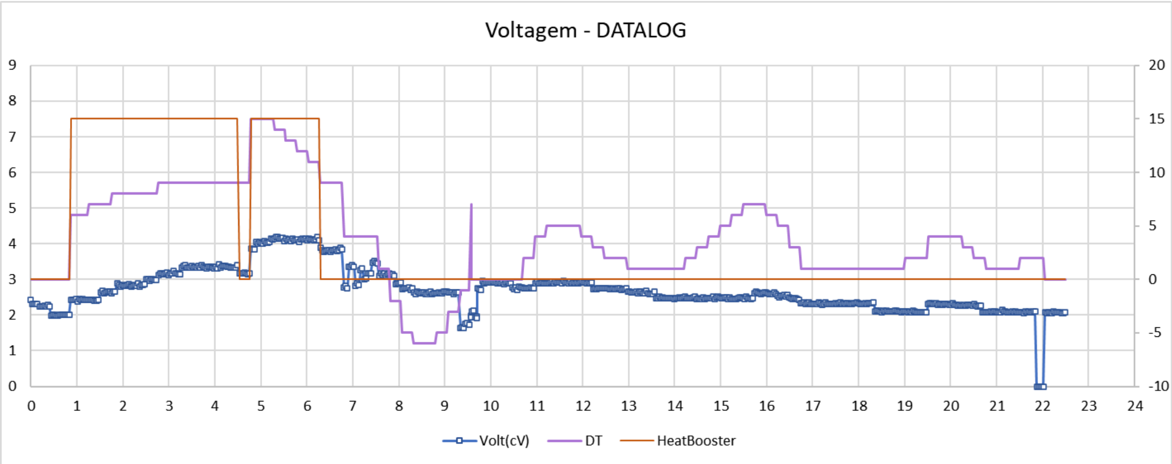
The relative humidity evolution of the silica gel container is also presented, which is useful to determine when the desiccant is saturated. With the trapdoor closed, there is minimal air transfer, leading to a gradual increase in the humidity. When open, the response is rapid, justifying the constant need to adjust the opening angle.







With version (1.2.J) there are new thermostat data, now represented by the TEC voltage supply, the *DT* variable value, and the HeatBooster activation period.



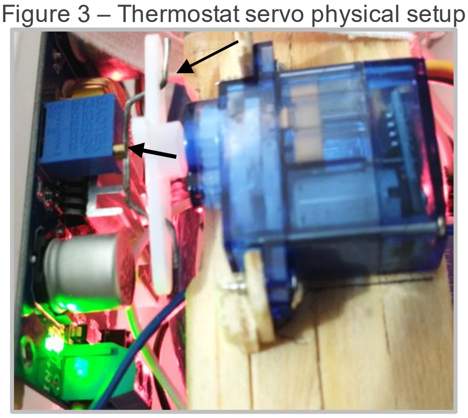
4.3. Thermostat adjustment

Observation

❖ Starting with version 1.2.J, the thermostat is controlled by a continuous rotation servo. For details on operation with a conventional servo (0° to 180°), see version 1.1.G.

The cooling power is controlled by a servo (*thermostat*) physically connected to the power supply that feeds the TEC. The servo's angle determines the output voltage of the power supply, which in turn determines the cooling power associated with a specific temperature difference between the external environment and the interior of the chamber (*DT*).

Figure 3 – Thermostat servo physical setup

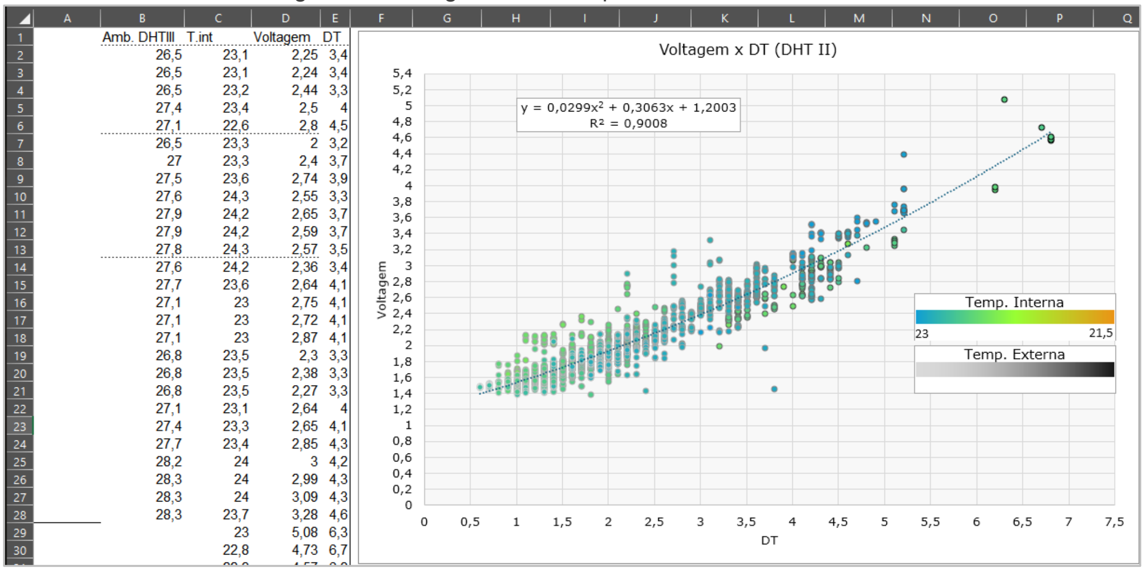


It is important to note that the physical arrangement of the chamber, especially its insulation capacity, is the primary factor related to the system's thermal performance and its ability to respond to variations in ambient temperature. Consequently, the response intensity and sensitivity to temperature variations (operation principle of HeatBooster) are also directly dependent on the arrangement.

Hence, a period of observation and data collection is required to correlate the TEC supply voltage with the system's internal equilibrium temperature.

The equilibrium values of the internal temperature should be observed according to the circuit's supply voltage, collecting the ambient temperature, the internal temperature, and the applied voltage. This way, a graph can be drawn that relates the voltage to the temperature difference (between the external and internal environments) and obtain a regression curve - voltage as function of (DT) - in quadratic regression using software such as Microsoft Excel (Figure 4).

Figure 4 – Voltage versus Temperature difference chart



Voltage detection is performed by reading a simple voltage divider from one of the analog pins. The divider is configured to map the voltage range from 0 to 12 V to 0 to 5 V, using the arrangement shown in Item 6.2.

Next, it is necessary to configure the continuous rotation servo. A standard servo (e.g. SG90) can be converted to continuous rotation, and there are several tutorials available online, each with its own advantages and disadvantages. The solution adopted in this arrangement was to position the servo at a 90° position and shorten the gear shaft, removing its connection to the potentiometer. The main advantage of this approach is reversibility; simply replacing the cut shaft with a whole one restores the original functionality. However, the greatest risk is potential instability, since the potentiometer remains free and possible physical movement can move it from its resting position.

Unlike conventional servos, in which position is controlled, continuous rotation servos have their speed controlled. Therefore, with the servo connected to the voltage supply's adjustment screw, it is necessary to know how long to keep it rotating at a given speed and in which direction to reach the target voltage.

First, one must know the servo's angular velocity for each value of the *write()* function, keeping in mind that the speed varies depending on the load to which the servo is exposed. Therefore, if possible, this calibration should be performed with the servo already connected to the voltage regulator.

I recommend recording videos of the servo making at least five revolutions at each speed of interest (for example, between positions 87 and 97). Then, calculate the average speed of each revolution using the video timestamps as a reference. See the table below for an example.

Position	Angular speed	Variation
88 (counterclockwise)	116,6 °/s	0,5 v/s
96 (clockwise)	147 °/s	0,61 v/s
Power supply variation	1,224 v/360°	
Mean value	0,55 v/s	

Using the voltage variation as a function of the servo's activation time, the algorithm is adjusted to calculate the difference between the target voltage and the current system voltage and activate the servo for the appropriate duration in the accordingly direction. It is advisable to perform the adjustment in a certain number of attempts to compensate for errors in the voltage adjustment (variation in the actual angular velocity relative to the calculated average, changes in the power supply variation per degree, output fluctuations due to the load on the TEC, etc.).

Observations

- ❖ Establish a minimum and maximum voltage to prevent attempts to adjust the system from exceeding the operational or physical limits of the voltage source.
- ❖ The stability of the internal temperature depends on the thermal energy balance of the chamber (with particular emphasis on the thermal conductivity of the arrangement). Data collection at various voltages can generate a characteristic curve that helps determine if the desired working range is achievable. Similarly, if there are modifications to the chamber structure, a new curve should be calculated.

4.3.1. Heatbooster

The implementation of the continuous rotation system brought great versatility to the chamber's operation over long periods and when exposed to significant seasonal variations. However, it was observed that the internal thermal balance became more sensitive to heating events with a delayed response, which is detrimental to operation, given the momentary reduction in relative humidity.

The slow response comes from the algorithm's approach of regulating the thermostat position to the equilibrium point with the **current** external temperature, which is rising. Therefore, if the external temperature rises faster than the system can respond, the internal temperature also rises. To compensate for this effect, the "heatbooster" algorithm was developed, which consists of temporarily increasing the cooling power to bring the final temperature forward by increasing the *DTc* variable. The increase value and the temperature rate that triggers the use of the heatbooster should be determined experimentally by observing the external and internal temperature records after a few sessions.

Because we are applying a temperature variation greater than the current one recorded by the algorithm, it is necessary to reduce the adjustment capacity during the heatbooster's operation. A value of 1/3 of the nominal setting was quite satisfactory, but other values can be adopted depending on the conditions – see the *TermoAjuste()* function.

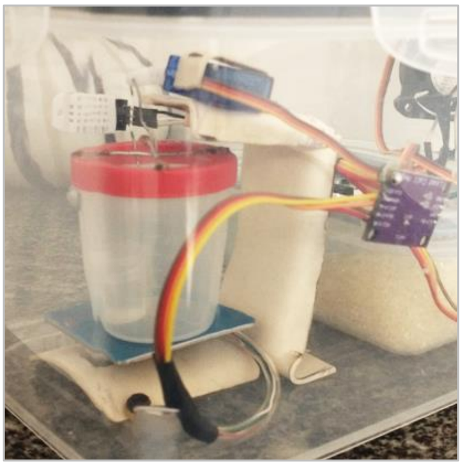
4.4. Operation with a scale

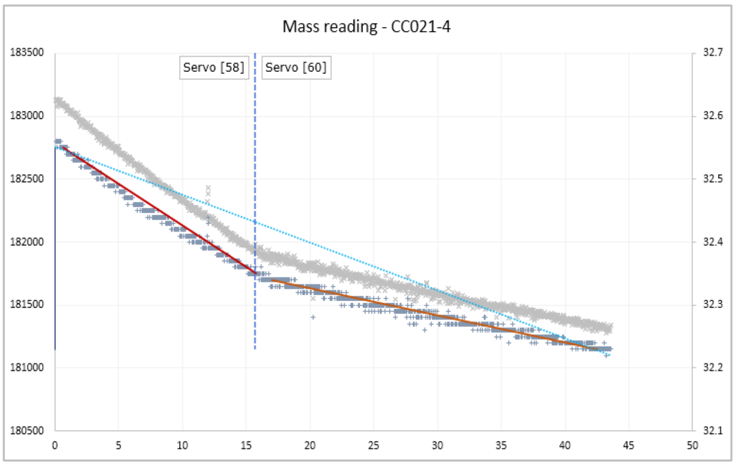
An alternative arrangement for the chamber consists in using a HX711 module with a Load Cell to measure the mass variation of the saline solution directly, observing the real evaporation rate in function of the exposure conditions.

Experience shows that to reduce the error associated with the Load Cell, it’s necessary to calibrate the sensor before each reading, avoiding deviations and error accumulation over time. Thus, it’s necessary to remove the solution container from the scale plate, turn it on, calibrate the scale, and only then place the container back and perform the weighting. The adopted answer to this was to utilize a servo to lift and lower the container during these stages.

This variation uses a specific sketch and requires almost all the Arduino available pins. Furthermore, there are requirements on the servo support, Load

Cell support structure, and the suspension element for the container which are not part of this documentation. Below is illustrated the tested arrangement and a typical result curve of mass reading over time.





5. VARIABLES AND PARAMETERS

The system prioritizes the adoption of integer arithmetic logic during the calculations and verifications, to save memory and optimize the Arduino UNO processing. Floating-point calculations are used only at specific times. Default variable values are highlighted in bold in the description below.

5.1. Operation

Type	Name	Value	Description
unsigned char	intLim	**0**	Registers the amount of menu entries
	OPE	0 – Maintenance **1** – Temperature control 2 – Humidity control 3 – Only temperature 4 – Calibration 5 – Only humidity	Control operation state. Has requirements with the lid (*tampa*)
	Visor	**0** – Off 1 – Operating 2 – Menu	Controls display operation mode
	SDticks	**0**	Delays SD card writing
	Uticks	**0**	Delays humidity control response
char	intSel	**0**	Operates the menu selector
bool	MAN	True/**False**	Checks whether the system has come from a maintenance state to reset
	vSD	True/**False**	Checks whether the SD card is inserted

5.2. Intervals

Type	Name	Value	Description
unsigned long	SDLT	**0**	Time of last cycle
	TECon	**0**	Time when TEC was turned on
	ULT	**0**	Time when humidity monitoring started
	RPMLT	**0**	Time the fan showed RPM below the limit
	varLT	**0**	Time the average variation was recorded
	intLT	**0**	Time of last interface action
	termLT	**0**	Time the thermostat was last changed
	buzzerLT	**0**	Time the buzzer alert was last activated
	tempResLT	**0**	Time of the last thermostat reset due to high temperature
const unsigned long	SDInterval	**20000**	Operation cycle interval (milliseconds)
	varInterval	**900000**	Recording of the average variation (15 minutes)
	intInterval	**5500**	Timeout for menu action and return to the standard screen
	termInterval	**180000**	Interval for thermostat functions
	buzzerInterval	**600000**	Interval to repeat the buzzer alerts

5.3. Moving Averages

Type	Name	Value	Description
unsigned char	pTin	**0**	Circular buffer position index of internal temperature
	pUin	**0**	Circular buffer position index of internal humidity
	pTex	**0**	Circular buffer position index of external temperature
	vTin	**0**	Filling of the internal temperature buffer
	vUin	**0**	Filling of the internal humidity buffer
	vTex	**0**	Filling of the external temperature buffer
int	mTin[6]	**0**	Circular buffer for internal temperature average
	mUin[6]	**0**	Circular buffer for internal humidity average
	mTex[6]	**0**	Circular buffer for external temperature average

5.4. Monitoring

Type	Name	Value	Description
bool	intW	True/**False**	Keep display on while interacting with the menu
	OPT	True/**False**	TEC status monitor (on or off)
	tampa	**True**/False	If the lid is attached
	Useco	True/**False**	Registers if an excessive dry event happened
	HB	True/**False**	Registers the “Heatbooster” status
int	rpm	**5000**	RPM reading for the external fan, starts high to bypass the initial verification
	US[3]		Humidity status 0 – Variation (1: rising -1: falling) 1 – Trapdoor servo position 2 – Servo angle variation
	varM		Variation multiplier for the 15-minute status
	varV	**9000**	15-Minute indicator - temperature
	varVE	**9000**	15-Minute indicator – external temperature
	varU	**9000**	15-Minute indicator - humidity
	conT[3]		Monitoring of temperature values 0 – Previous temperature, for calculating variation 1 – Moving average of 5 readings 2 – Temperature 15 minutes ago
	conU[4]		Monitoring of humidity values 0 – Previous humidity, for calculating variation 1 – Moving average of 5 readings 2 – Module of the last variation 3 – Humidity 15 minutes ago
	conE[3]		Monitoring of external temperature values 0 – Temperature 15 minutes ago 1 – Moving average of 5 readings 2 – Reference temperature for HeatBooster
	adcRead	**0**	ADC reading of TEC voltage
	Vtec	**0**	Moving average of TEC voltage
	Valvo	**300**	TEC voltage target, initialized at 3.0 volts
char	TS[2]		Temperature status 0 – variation (1: rising -1: falling) 1 – Occurrences below *Tinf*
	MoSt	**1**	*Monitor* status 0 – Normal 1 – Maintenance return 2 – Low cooling power 3 – Low drying capacity 4 – High cooling power 5 – Fan failure 6 – Temp. above *Tsup* 7 – DHT failure
unsigned int	termSec	**0**	Thermostat servo moving time
volatile unsigned long	counter	**0**	Counter for reading the RPM of the external fan

5.5. Operation parameters

Type	Name	Value	Description
const int	Tsup		Upper limit of temperature range
	Talvo		Target temperature
	Tinf		Lower limit of temperature range
	Usup		Upper limite of humidity range
	Ualvo		Target relative humidity
	Uinf		Lower limit of humidity range
int	Tb	**Talvo**	Tolerance for TEC restart when returning from a shutdown below *Tinf*
	dUmax	**10**	Initial maximum humidity variation
	DTc	**0**	∆T compensation on thermal control
float	DT	**0**	External temperature difference relative to the target
const unsigned char	sClose	**52**	Humidity servo maximum closed angle
	sOpen	**0**	Humidity servo maximum opened angle

6. SCHEMATICS

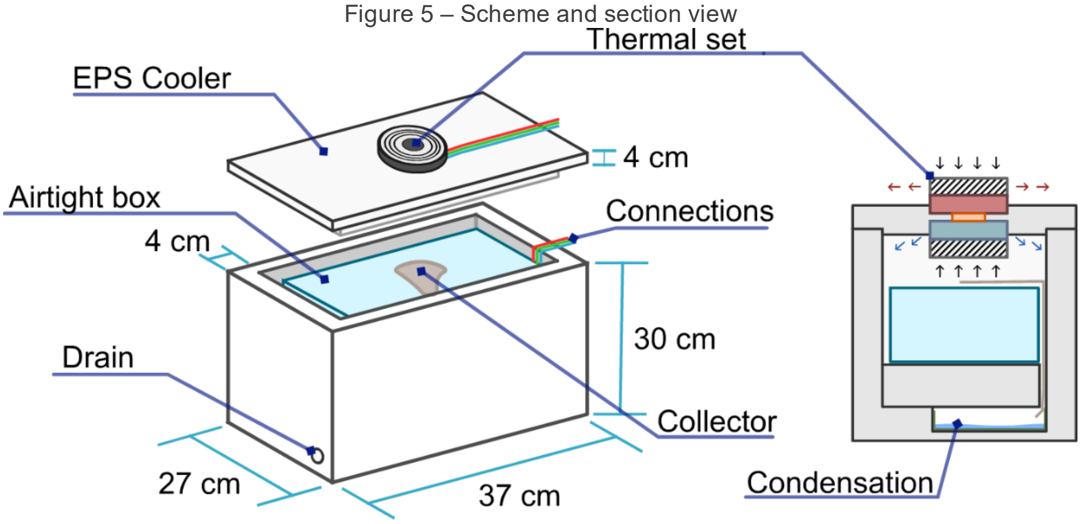
6.1. Structure

The chamber's hygroscopic insulation is achieved by using a airtight polypropylene container. The wiring passes through a hole in the side of the box sealed with silicone. For the dimensions of this project, a 4-liter container with locks on the lid was used. The thermal insulation comes from the EPS cooler box (12 liters). The walls were originally 2 cm thick and were reinforced by another 2 cm by gluing new plates to improve the insulation.

The thermal set is fixed in the center of the lid. A computer fan/heatsink is fixed to each side of the Peltier element. Thermal paste is used on all interfaces. It is possible for the internal temperature to reach the dew point on the cold heatsink, causing moisture condensation inside the chamber. Therefore, care must be taken with the arrangement of the internal heatsink to prevent water runoff from reaching the electronic parts of the fan and causing damage. In addition, dripping on the hermetic container is annoying and can interfere with operation. To prevent water accumulation, a fabric is used as a collector, directed along the side of the cooler to a reservoir at the bottom, where the water will be deposited by capillarity. A small hole in the front of the chamber acts as a drain for convenience, and it is advisable to cover it, since the volume of condensation water is usually small for short time periods.

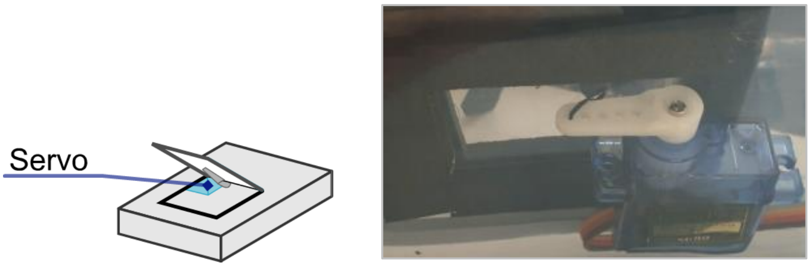
The DHT sensors are positioned: 1) inside the silica gel container 2) inside the airtight container 3) on the outside of the chamber.

Figure 5 – Scheme and section view



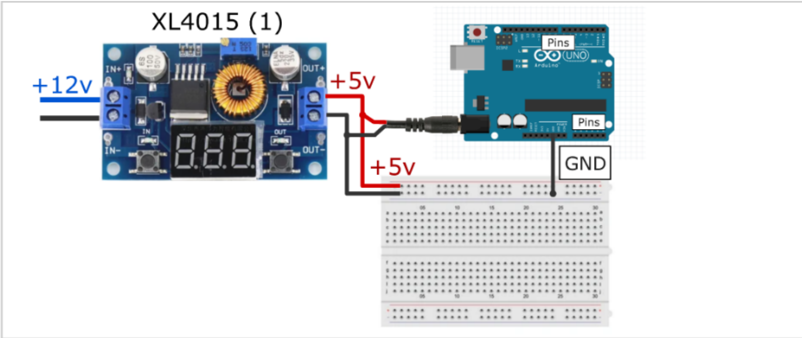
The desiccant trapdoor was also made using an airtight container, with a cut in the lid and positioning of the servo on the inside, attaching its arm to the lid so that in addition to pushing it open, it can pull it to close.

Figure 6 – Desiccant trapdoor scheme

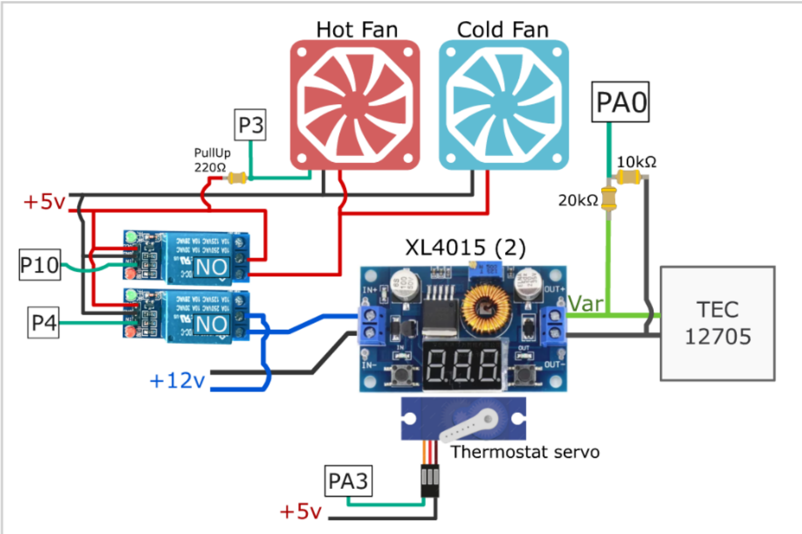


6.2. Connections

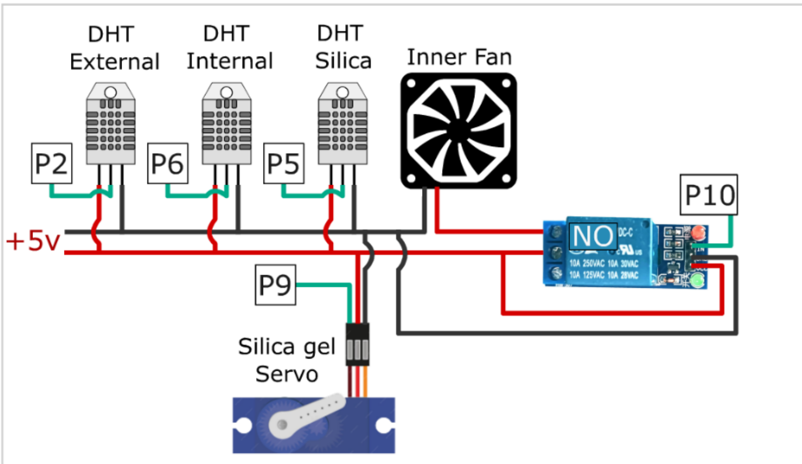
Power supply



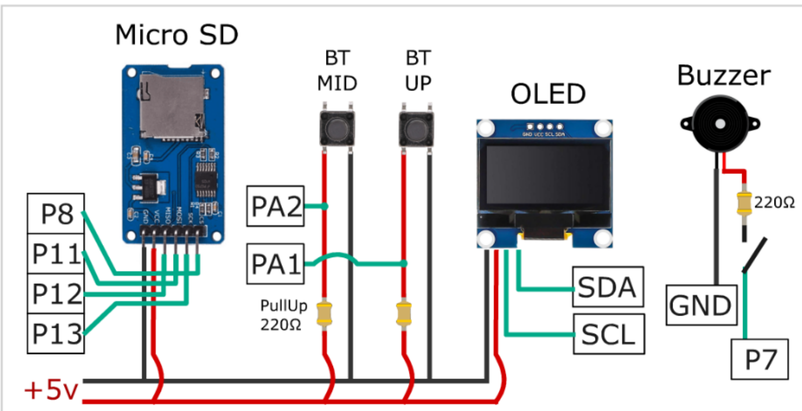
Temperature



Humidity



Interface



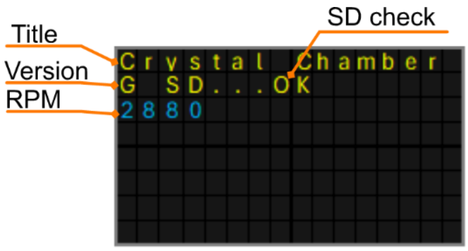
6.3. Display

Observation

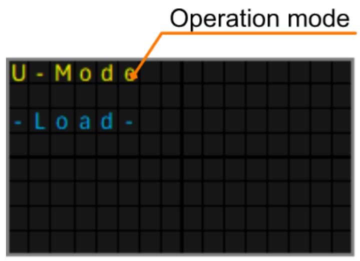
❖ Starting with version 1.2.J, the display was updated to show Thermostat and Heatbooster info that are not presented in this section. The below images follow the basic configuration from version 1.1.G

The sketch uses a 128 x 64 pixel display and the U8x8 library, where each character has 8 pixels resulting in a 16 x 8 grid to information arrangement with the following variations:

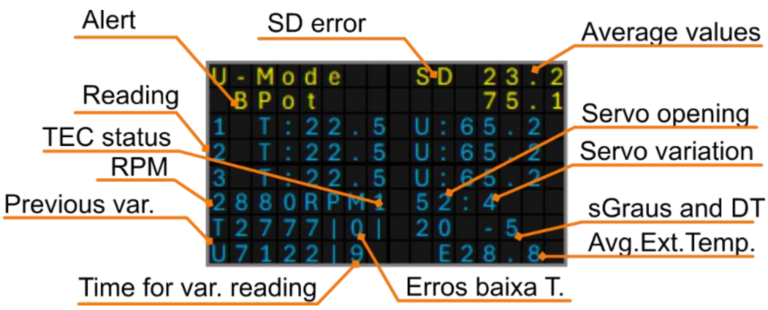
Initial screen



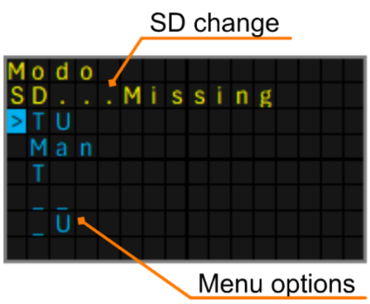
Restart



Usual operation

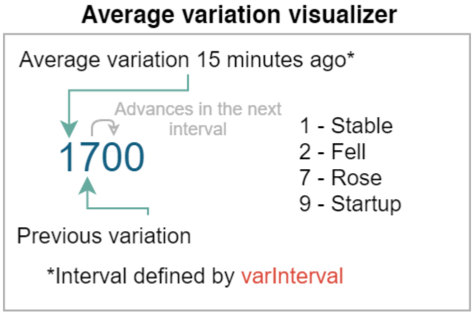


Control menu



The average variation is periodically verified and displayed in a coded form, allowing for the monitoring of recent history:

Average variation visualizer



6.4. Imagest

6.4.1. Basic version (1.1.G)

Figure 7 – External overview

Figure 8 – Side view

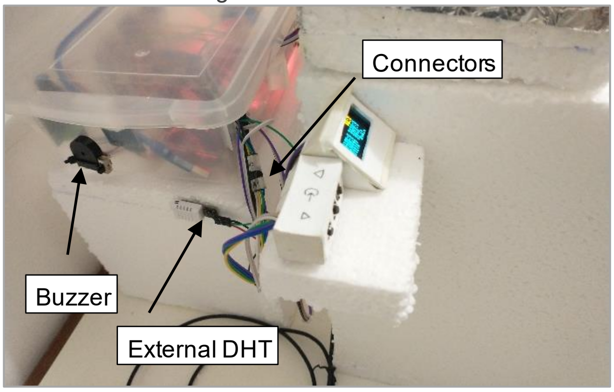


Figure 9 – Top view

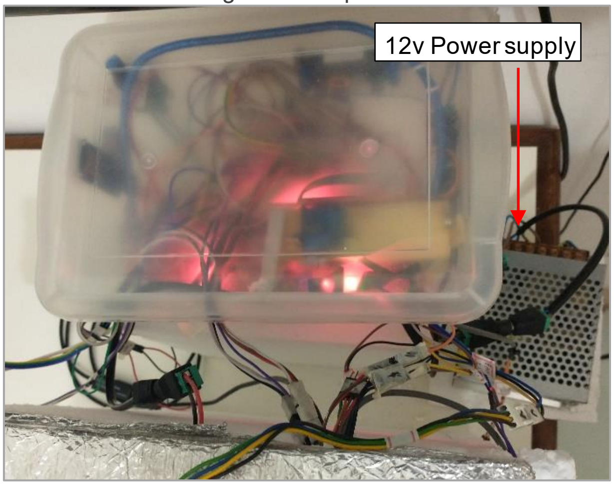


Figure 10 – Electronics box

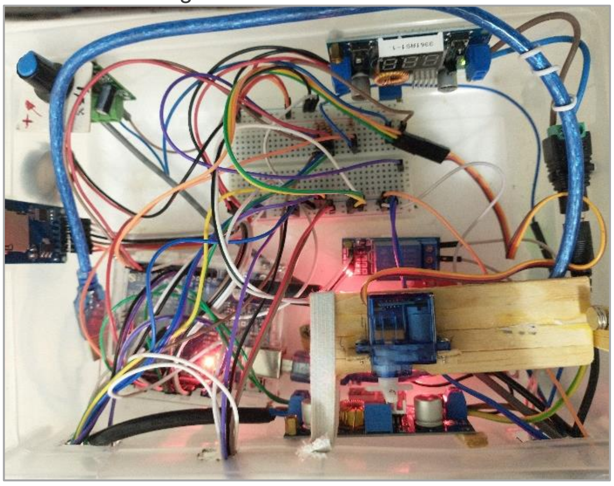


Figure 11 – Internal view

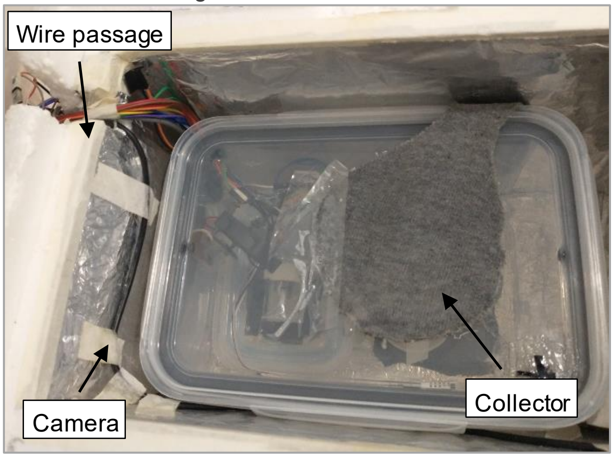


Figure 12 – Inside, silica gel and fan

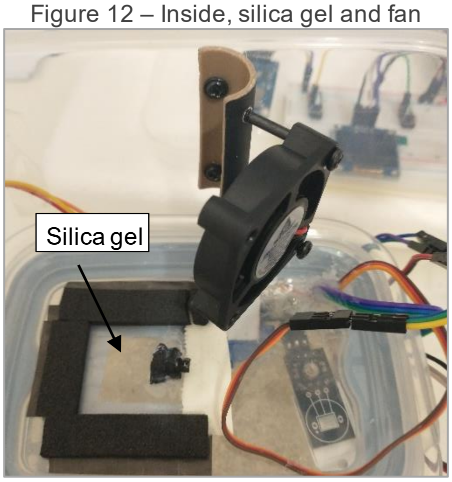


Figure 13 – Exposure environment

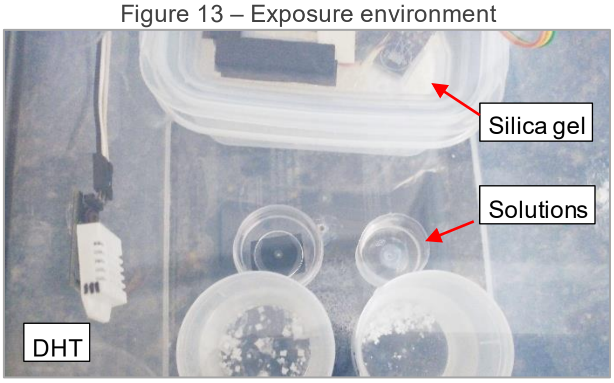
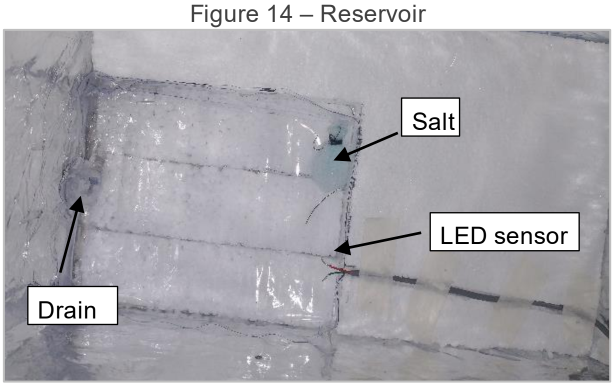
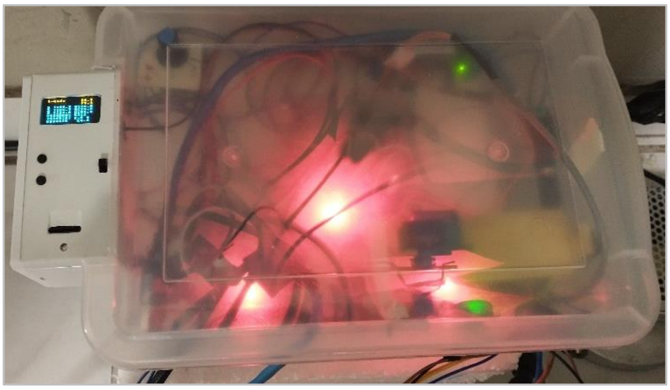
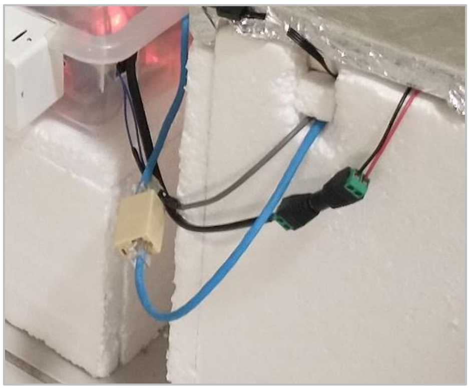
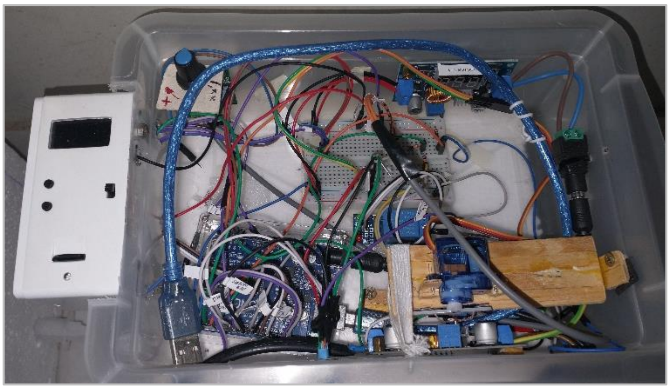
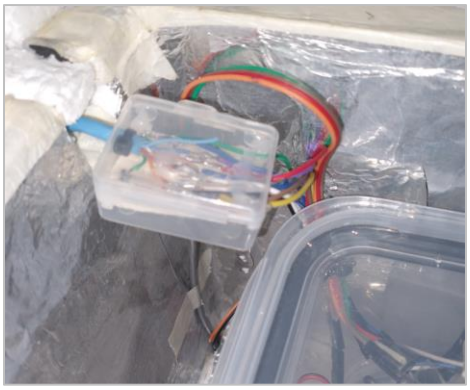


Figure 14 – Reservoir



6.4.2. Updates (1.2.J)

With version 1.2.J, an interface panel was developed and attached to the electronic box, unifying all external (buzzer, DHT) and interface components (SD, buttons, and display). Connections between the chamber and control box are now made using network cables and RJ45 connectors, facilitating handling and mitigating poor connection issues typical of jumper cables. The top fan was also replaced with a quieter model, using the same mounting scheme.


7. UPDATES

In relation to the previous version (1.1.G)

- ❖ Adoption of continuous rotation servo for the thermostat, eliminating the need for position-voltage calibration and providing greater flexibility to environmental variations.
- ❖ Added a voltage divider circuit to read the TEC supply voltage.
- ❖ Readjusted wiring, pin connections, and control algorithms (operational conditions, settings, and temperature and voltage monitoring).
- ❖ Implemented the "HeatBooster" algorithm.