

**Nume:**Odajiu Tudor

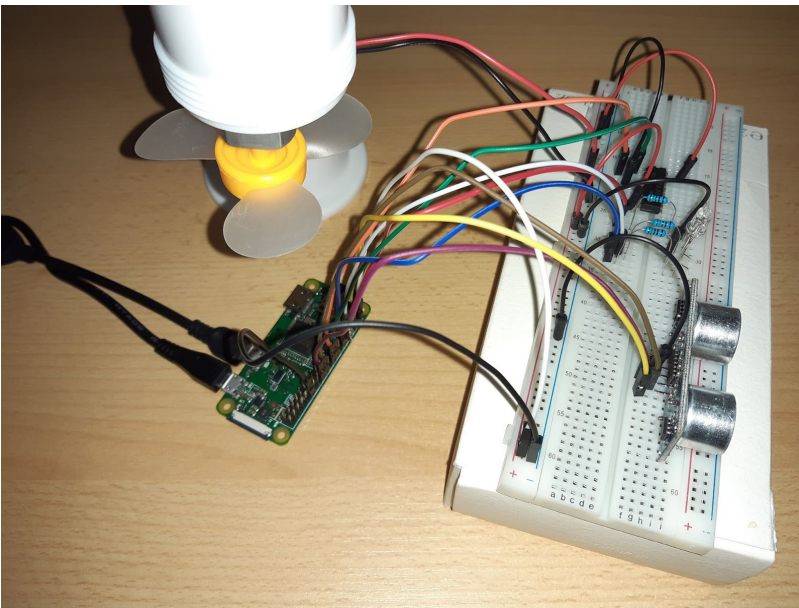
**Mail:** tudor.odajiu.liis@gmail.com



**În echipă cu:** Postolachi Vasile, grupa 1308 A























**Title:** Monitoring and self-regulating the processor temperature

Elevator pitch: The temperature will be monitored with an rgb led, and the regulation will be done with a fan which can be turned on manually at any time or automatically when it reaches a certain temperature threshold.



**Things used in this project**

## Hardware components

	Raspberry Pi Zero	× 1	 
	Breadboard (generic)	× 1	 
	Ultrasonic Sensor - HC-SR04 (Generic)	× 1	
	RGB Diffused Common Anode	× 1	
	Through Hole Resistor, 150 ohm	× 3	
	DC Motor, Miniature	× 1	
	Texas Instruments Dual H-Bridge motor drivers L293D	× 1	 
	Male/Female Jumper Wires	× 10	 
	Male/Male Jumper Wires	× 5	 

## Story

If we use more than one application at the same time, which causes the cpu to overheat, problems with processing speed may occur... or worse, the board may be damaged.

### How it works:

The solution is to use a fan that will cool the board

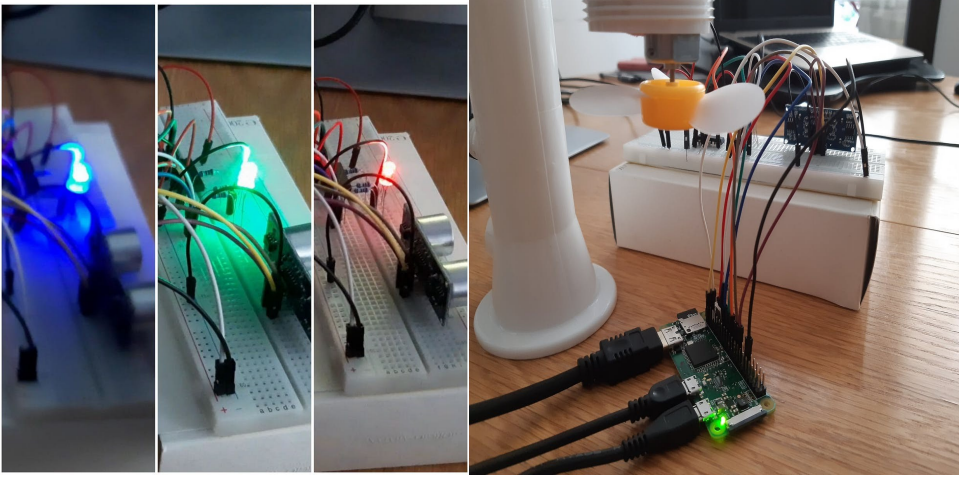
The processor temperature is constantly checked.

If the temperature is below 38 degrees, the LED is blue.

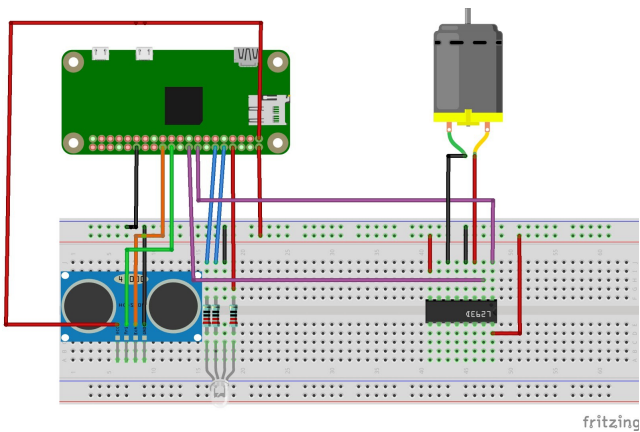
If the temperature rises, and exceeds 38 degrees, but is lower than 45 degrees, the LED is green.

Thus, we can see that the board is more demanding, and cooling is needed. In this case we can bring our hand close to the sensor to turn on the fan. We can be sure that the board is in good condition when the LED will be blue.

If the temperature is higher than 45 degrees, the LED is red and the fan turns on automatically.



## Schematics



Code: [https://github.com/tudor-odajiu/Proiect-SM/blob/master/project\\_SM.py](https://github.com/tudor-odajiu/Proiect-SM/blob/master/project_SM.py)

Hackster: <https://www.hackster.io/348621/monitoring-and-self-regulating-processor-temperature-6230c3>

youtube: [https://www.youtube.com/watch?v=QGc2gkVCAts&fbclid=IwAR374VtncmguuiD\\_nGowLyNP4bgl6euOZE5WcXsQ6kD1-ac\\_PkDpcdYkJ24](https://www.youtube.com/watch?v=QGc2gkVCAts&fbclid=IwAR374VtncmguuiD_nGowLyNP4bgl6euOZE5WcXsQ6kD1-ac_PkDpcdYkJ24)

[https://www.youtube.com/watch?v=QGc2gkVCAts&fbclid=IwAR374VtncmguuiD\\_nGowLyNP4bgl6euOZE5WcXsQ6kD1-ac\\_PkDpcdYkJ24](https://www.youtube.com/watch?v=QGc2gkVCAts&fbclid=IwAR374VtncmguuiD_nGowLyNP4bgl6euOZE5WcXsQ6kD1-ac_PkDpcdYkJ24)

**Contribuția fiecărui membru:**

**Postolachi Vasile**

- implementare cod pentru detectia miscarilor , utilizand senzorul ultrasonic

- implementare cod pentru semnalizari luminoase a starilor procesorului in dependenta de temperatura , prin intermediul LED-ului RGB
- creare schematica a conectarii senzorului si a LED -ului la placa
- prezentare video a proiectului
- completare in hackster a componentelor utilizate si crearea schematica a proiectului final

### **Odajiu Tudor**

- implementare cod pentru monitorizarea temperaturii
- implementare cod pentru reglarea motorului la anumite temperaturi
- creare schematica a conectarii motorului la placa
- editare in hackster a modului de functionare a proiectului
- editare cod in hackster si salvare cod pe github

/\*\*\*\*\*/