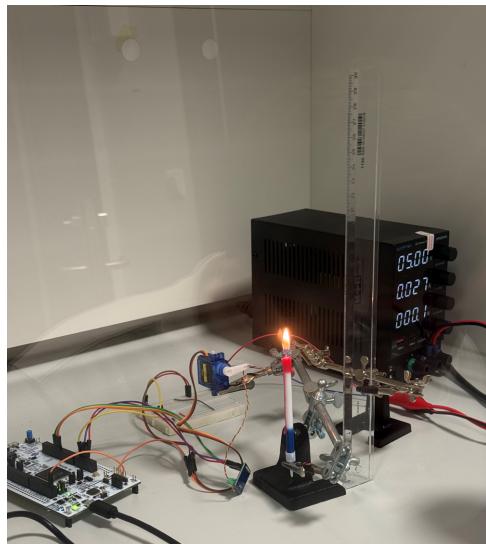


# ELE201 Semester Project Report - Candle Flame Temperature Mapping

Tobias Slethei  
Even Børshheim  
Tobias Haavik Hjelle  
Mathias Johannessen-Bech

November 9, 2025



**Figure 1:** Experimental setup showing the STM32F767ZI Nucleo board, MAX31855 thermocouple converter, servo mechanism, and candle flame measurement apparatus.

## Contents

<b>1 Abstract</b>	<b>2</b>
<b>2 Methodology</b>	<b>2</b>
<b>3 Challenges</b>	<b>4</b>
<b>4 Conclusion</b>	<b>5</b>
<b>5 AI Declaration</b>	<b>7</b>
<b>6 References</b>	<b>7</b>
<b>7 Appendix</b>	<b>7</b>

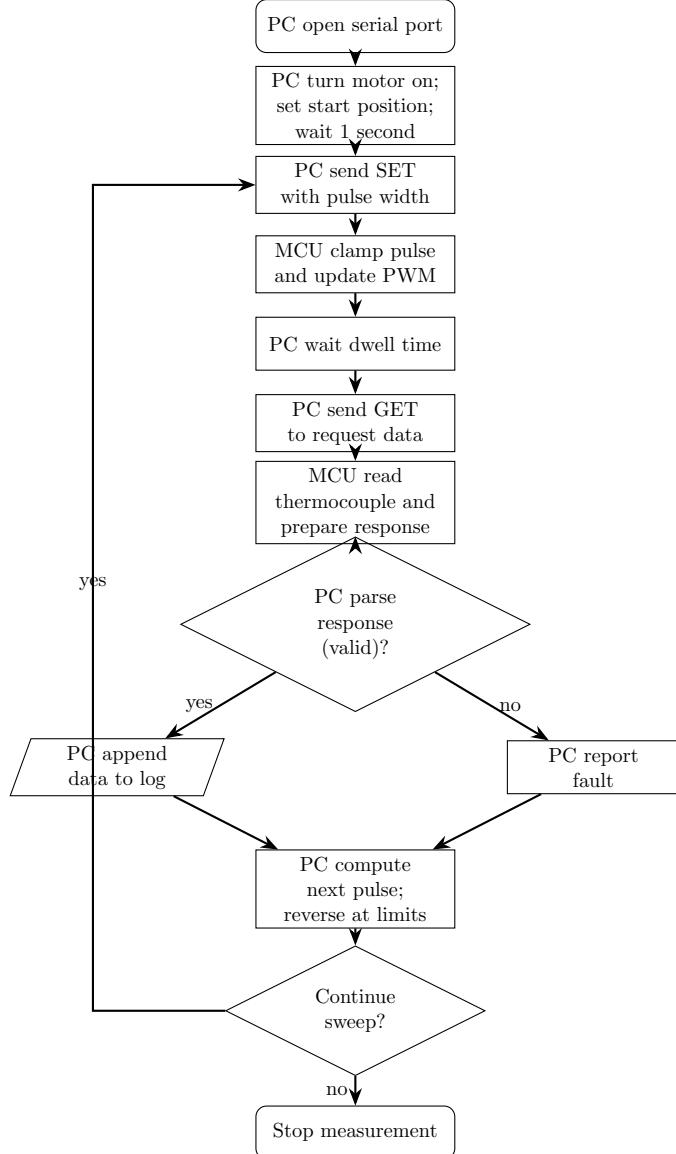
## 1 Abstract

We studied how the temperature inside a candle flame changes while the candle burns down. A K-type thermocouple mounted on a hobby servo swept across the flame so that every probe position could be linked to a temperature sample. The STM32 Nucleo-144 development board (shown in Figure 1) generated the PWM signal for the servo, read the MAX31855 thermocouple converter over SPI, and shared each measurement through a small UART command set. A Python program collected the data, converted pulse widths to spatial coordinates, and produced heatmaps that show how the flame narrows and cools as the wick collapses.

## 2 Methodology

The experiment was designed around a clear division of responsibilities between firmware and host processing. On the microcontroller we configured timer TIM4 to generate a 20 ms PWM period (50 Hz) with a controllable pulse width between 500 microseconds and 2500 microseconds. This covered the mechanical travel of the servo while leaving safety margins on both ends. SPI4 operated as an 8-bit receive-only master with a prescaler of 32 so that the serial clock remained below the 5 MHz limit set by the MAX31855 converter. A dedicated GPIO line drove chip select. During each sampling cycle the firmware collected the thermocouple temperature, the internal cold junction estimate, and the converter fault bits. These values, together with the current pulse command and a millisecond timestamp, were returned on request through a line-based UART command called `GET`. Companion commands `SET <pulse>` and `MOTOR ON/OFF` allowed the host to move the servo while the firmware ensured that all requested pulse widths stayed inside the safe window.

The host program opened the serial connection, enabled the servo, and applied a saw-tooth scan between 1300 microseconds and 1700 microseconds. After each change of pulse width the program paused 100 milliseconds to allow the motor to reach its target position and ensure the thermocouple had time to reach thermal equilibrium. Every recorded row was written to comma-separated storage with its timestamp, measured temperature, servo position, and internal converter temperature. The interaction between the MCU and PC is visualized in Figure 2.



**Figure 2:** PC-MCU interaction during flame temperature measurement sweep. The loop implements a sawtooth scan pattern with direction reversal at configured pulse width limits.

The analysis stage then transformed each pulse width into an angle by linearly relating the commanded pulse to the servo travel limits:

$$\theta(p) = \theta_{\min} + (\theta_{\max} - \theta_{\min}) \frac{p - p_{\min}}{p_{\max} - p_{\min}} \quad (1)$$

Here we used  $\theta_{\min} = -90^\circ$ ,  $\theta_{\max} = 90^\circ$ ,  $p_{\min} = 2500$  microseconds, and  $p_{\max} = 500$  microseconds based on the mechanical calibration performed before the measurements. We measured that the servo arm  $r = 16$  mm. The horizontal displacement of the probe followed

$$x(\theta) = r \sin(\theta), \quad (2)$$

while the instantaneous vertical position combined the radial projection and the natural regression of the candle with an offset that anchors the first sample at the initial wick height:

$$y(\theta, t) = r \cos(\theta) + v \cdot t - r. \quad (3)$$

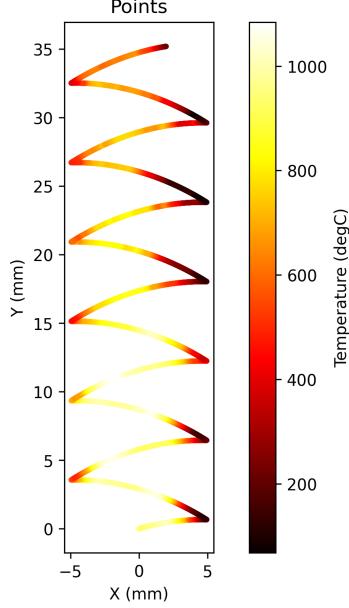
with  $v = 40$  mm/(10 · 60 s) estimated from video of the burn. Subtracting the servo radius  $r$  ensures that  $y = 0$  corresponds to the first measurement taken at the starting wick height. Using the kinematic model defined in Equations (1), (2), and (3), we converted all pulse width commands to Cartesian coordinates  $(x, y)$  for spatial analysis. We assumed that the single wick produced a symmetric flame, so each sample was mirrored as  $(x, y, T) \mapsto (-x, y, T)$  to increase spatial coverage before the temperatures were interpolated on a regular grid.

### 3 Challenges

The changing shape of the flame was the most significant challenge. As the wick bent and shortened, the hottest region drifted laterally and vertically, so it was difficult to capture a single, repeatable temperature field. We expected the communication layer to be demanding, yet the simple UART command set made it reliable once the bounds were enforced in firmware. Conversely, we assumed that probe alignment would be straightforward, but minor mechanical offsets caused noticeable asymmetry and required several recalibrations. Another unexpected difficulty was settling time: when the dwell delay was too short, the recorded temperatures lagged behind the actual flame temperature because the thermocouple junction could not equilibrate quickly enough.

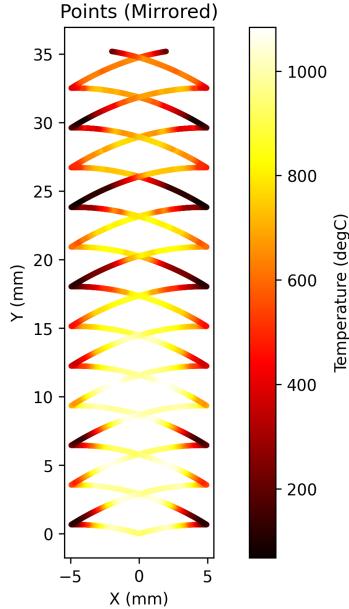
## 4 Conclusion

We first examine the raw spatial samples shown in Figure 3. The plot follows the servo arc and shows how the measurement plane moves as the candle shortens; early samples cluster near the initial wick height, while later samples shift downward with time.



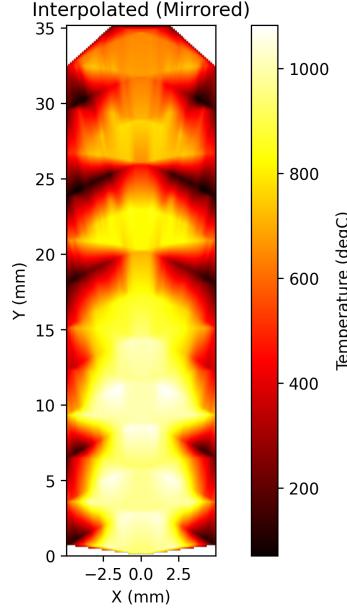
**Figure 3:** Raw temperature measurements showing the curved servo path and vertical drift due to candle consumption over time.

To increase spatial coverage, we then mirror the data about the centerline as seen in Figure 4. This highlights small alignment errors while the hottest measurements remain close to the axis.



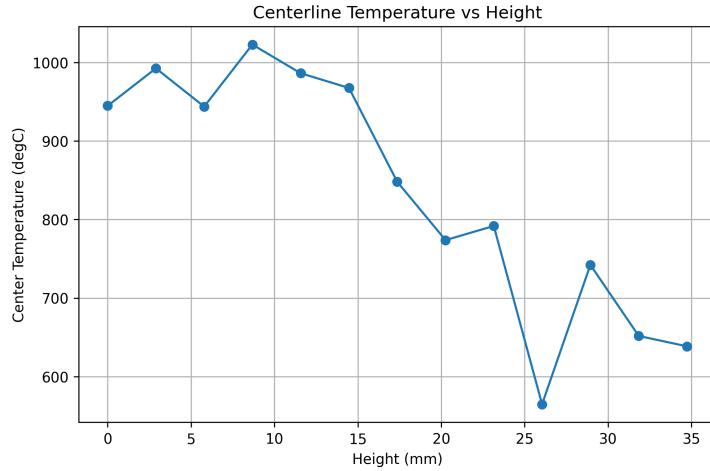
**Figure 4:** Mirrored dataset showing improved spatial coverage and revealing asymmetry due to mechanical alignment errors.

Interpolating the scattered measurements onto a regular grid produces the temperature field shown in Figure 5. The peak region is centered near the wick at the start and becomes thinner and slightly tilted as the burn progresses, consistent with visual observations of the flame leaning.



**Figure 5:** Interpolated temperature field revealing the hottest core near the wick with characteristic thinning and tilting as the flame evolves.

Finally, Figure 6 presents the centerline profile that summarises how the hottest core changes over height. Ideally, this would decrease smoothly, but the bending and shortening of the wick introduce deviations from a simple trend.



**Figure 6:** Centerline temperature profile showing the variation of peak temperature with height above the initial wick position.

## **5 AI Declaration**

Artificial intelligence tools were used as a supplementary resource throughout the project. They assisted primarily with language editing, document structure, and generating preliminary code templates or conceptual outlines during firmware development.

## **6 References**

MAX31855 Thermocouple-to-Digital Converter Datasheet: <https://www.analog.com/media/en/technical-documentation/data-sheets/max31855.pdf>

## **7 Appendix**

Code repository: [https://github.com/tobiasslethei/flame\\_logger](https://github.com/tobiasslethei/flame_logger)