

# Terminus K3

Development of an Affordable and Portable Guarded Hot Plate Apparatus for On-Site Thermal Conductivity Measurements of Fabrics in Forensic Investigations



# Development of an Affordable and Portable Guarded Hot Plate Apparatus for On-Site Thermal Conductivity Measurements of Fabrics in Forensic Investigations

O.C.H ten Haven (4456610)<sup>1</sup>  
Supervised by dr.ir. A.J. Loeve<sup>2</sup>

<sup>1</sup>MSc programme Mechanical Engineering, Track Bio Mechanical Design, Delft university of Technology, 2628CD Delft, The Netherlands

<sup>2</sup>Forensic Biomechanical Engineering, Department of Biomechanical Engineering, Faculty of Mechanical Engineering, Delft university of Technology, 2628CD Delft, The Netherlands

---

## KEYWORDS

---

Post-Mortem  
Interval  
Guarded Hot  
Plate  
Thermal  
Conductivity  
Measurement

---

## ABSTRACT

---

Accurately determining the early Post-Mortem Interval (PMI), the period shortly after death, is critical for reconstructing the timeline of suspected crimes. The PHOEBE model, a new thermodynamic finite-difference model that simulates body cooling, significantly improves PMI estimations but is highly sensitive to the thermal conductivity of the deceased's clothing. This study introduces the Therminus-K3 prototype, an affordable and portable Guarded Hot Plate apparatus, designed specifically to support the PHOEBE model by providing rapid, non-destructive thermal conductivity measurements of clothing garments. The Therminus-K3 features a compact Hot Stage with resistive PCB heaters, a Cold Stage with thermoelectric coolers, and a Compression Stage that replicates real-world fabric compression conditions. This design allows for rapid attainment of steady-state conditions and thermal conductivity measurements within 25 minutes, essential for time-sensitive forensic investigations. Preliminary results suggest that the Therminus-K3 prototype measures with a precision of less than  $\pm 1\%$  and a measurement uncertainty below  $\pm 1.5\%$ , but further testing is required to confirm its overall accuracy. With a material cost of just €404.39, the Therminus-K3 offers forensic practitioners a more affordable option to accurately measure the thermal conductivity of clothing, thereby enhancing the accuracy of PMI estimations. Initial tests suggest that integrating the Therminus-K3 with the PHOEBE model could improve PMI estimations by over 30 minutes. Its ongoing development holds significant potential to further impact and improve the determination of time of death in forensic scenarios in a more efficient and reliable manner.

---

## 1. Introduction

Accurately determining the Post-Mortem Interval (PMI), the time elapsed since death, is crucial in forensic investigations, especially in cases of non-natural deaths. A precise PMI is key to pinpointing the time of death, facilitating the collection of time-sensitive evidence such as security camera footage and eyewitness accounts. It also plays a significant role in verifying alibis and piecing together events leading to the death, thereby enhancing the investigation's efficiency and reliability. An accurate PMI adds objective evidence to a case, reducing reliance on subjective personal testimonies. Thus, accurate PMI estimation is vital for thorough and effective forensic investigations [1-3].

Establishing an accurate PMI during the early phase, spanning 3 to 72 hours post-mortem, is crucial for investigations into the cause of death as most legal cases are examined in this period [2]. There are two groups of methodologies for estimating the PMI in the early phase. The first group includes techniques that rely on sampling tissue or bodily fluids, followed by meticulous laboratory examinations to quantify the physiological changes that have occurred since death. The second group, on the other hand, focuses on assessing optical, mechanical, or thermal changes in human tissue that do not require any sample extraction or laboratory examinations, allowing for immediate measurements at the crime scene [4, 5].

Of these techniques, the change in body temperature is most frequently measured, with Henssge's nomogram method being considered the golden standard for early thermometric PMI estimations. This method correlates the deceased's rectal core temperature, ambient temperature, and body weight to the PMI, based on the assumption of a double exponential decay of the post-mortem rectal temperature. To enhance the precision of this method and adapt it to various scenarios, a correction factor is utilized. This factor takes into account additional variables such as body weight, clothing, and environmental conditions, with the aim of improving the accuracy of the PMI estimation by tailoring it to the specifics of each individual case [6, 7].

While Henssge's nomogram method is praised for its simplicity and ease of use, it is susceptible to personal error due to the subjective choice of the correction factor, which necessitates experience and expertise. Additionally, the invasive procedure required to measure the rectal core temperature carries the risk of contaminating the mortal remains, potentially compromising crucial forensic evidence. Moreover, another notable downside is its substantial uncertainty in PMI estimations, with variations ranging broadly from  $\pm 3$  to  $\pm 7$  hours. Henssge's method does also not account for variables such as posture, body physique or changing cooling conditions, which can all play crucial roles in body cooling and therefore PMI estimation [5-7].

In response to these limitations, a new method for PMI determination, utilizing a numerical thermodynamic finite-difference model has been developed by Wilk et al. This model, known as the PHOEBE model, addresses previously mentioned limitations by incorporating variables such as body posture, stature, and body composition, as well as environmental factors like contact surface, submersion in water, and clothing coverage. The method has shown promising results, with an average deviation of no more than  $\pm 38$  minutes from their respective true PMI ranging from 5 to 50 hours, making it a significantly more precise tool for forensic investigations. The PHOEBE model requires 41 specific input parameters, such as body dimensions, parameters related to body composition, environmental contact parameters, and a variety of environmental conditions [5, 8]. This extensive list of input parameters not only enhances the precision of the model's PMI estimations but also ensures that each estimation is meticulously tailored to the specific circumstances of each individual case. Crucially, the selection of these parameters was guided by their measurability or identifiability at the crime scene. This selection of parameters enables the model to closely replicate the real-life conditions surrounding the death, which significantly enhances the model's accuracy in providing a reliable PMI estimation.

Recognizing the importance of accurately measuring these input parameters, Project Therminus was initiated. This collaborative effort between the Netherlands Forensic Institute (NFI), Amsterdam Medical Centre (AMC), and Delft University of Technology, focuses on developing new methods to precisely measure the necessary input parameters for the PHOEBE model at the crime scene. This project aims to provide more accurate and reliable parameter input data for the PHOEBE model, leading to more precise and trustworthy time of death estimations.

The thermal conductivity coefficient of the deceased's clothing is the most critical factor in the PHOEBE model, as it has a big effect on the accuracy of the PMI estimation. Even a minor uncertainty as small as 0.01 W/mK, has the potential to skew the PMI by nearly three hours in the most extreme scenario [5]. Unfortunately, it is not feasible to determine thermal conductivity solely from the fabric composition listed on the garment labels. The fabric's intrinsic properties, such as thickness, weaving pattern, yarn spacing, and multilayer sequence, together with external factors like moisture content and compression, significantly impact its thermal conductivity [9]. Therefore, obtaining on-site measurements that accurately reflect the clothing's thermal conductivity at the time of death is essential for enhancing the accuracy of PMI estimations.

However, the tools currently available for measuring thermal conductivity are not suited for on-site use due to their size, complexity, and cost [10]. This highlights the need for a portable, user-friendly, and affordable device capable of

measuring the thermal conductivity of clothing garments found on deceased individuals at the crime scene. Initial steps towards developing such a device have already been taken. These preliminary efforts include a literature review conducted by the current author, aimed at identifying the most suitable measurement technique for on-site use, along with the creation of the Therminus-K1 and K2 prototypes by multiple student groups [10-12]. This paper focuses on the subsequent development of the Therminus-K3 prototype, with the goal of addressing the shortcomings of its predecessors and enhancing the functionality, accuracy, and user-friendliness for on-site measurement in forensic investigations.

## 2. Theoretical Background

The measurement principle of the Therminus-K3 prototype is based on the Guarded Hot Plate (GHP) method. This technique is considered an industry benchmark due to its absolute measurement nature, precision, and repeatability [13]. It is furthermore capable of measuring the thermal conductivity of heterogeneous materials like fabrics, in which heat transfer takes place through each of the distinctive heat transfer modes: conduction, convection and radiation [14, 15]. Consequently, the GHP method effectively measures the apparent thermal conductivity, an important aspect for accurately evaluating fabric behaviour under varied real-world conditions where multiple heat transfer modes often occur simultaneously [16].

In the GHP method, thermal conductivity is determined by applying the one-dimensional form of Fourier's law of heat conduction. This approach involves creating a controlled environment where heat transfer occurs primarily in one direction, thereby simplifying the complex heat transfer process to a one-dimensional model. By ensuring steady-state conditions, where the temperature distribution within the material remains constant over time, Fourier's Law can be applied to accurately measure thermal conductivity. In this context, Fourier's law correlates the rate of heat flow per unit area to the temperature gradient, allowing for the derivation of thermal conductivity as follows:

$$k = \frac{\dot{Q}L}{A\Delta T} \quad [1]$$

Where  $k$  represents the apparent thermal conductivity parallel to the one-dimensional heat flow, measured in W/mK,  $\dot{Q}$  denotes the one-dimensional heat flow in watts (W), indicating the amount of heat passing through the material per unit of time.  $L$  is the thickness of the material, measured in meters (m), and  $A$  is the cross-sectional area through which the heat is transferred, measured in square meters (m<sup>2</sup>). Lastly,  $\Delta T$  refers to the temperature difference across the material, measured in Kelvin (K) [17].

Figure 1 illustrates the general arrangement of a GHP apparatus. The apparatus establishes a

Requirement:	Target:	Reason:
Accuracy error	< 3 %	Accuracy greatly affects PMI estimation and results should be admissible in court [5, 8].
Precision error	< 1 %	Precision greatly affects PMI estimation and results should be admissible in court [5, 8].
Measurement Range	0.020 - 0.700 [W/mk]	K-values above 0.700 W/mk hardly affect the accuracy of the PMI estimation [8].
Variable Pressure	0 – 10000 [Pa]	Target range covers average pressures encountered in supine and sitting positions [18].
Sample thickness range	0.44 – 28 [mm]	Target range covers 99.73% of all clothing [12].
Sample dimensions	Non-destructive	Clothing garments should ideally remain undamaged to preserve them as evidence [10].
Measurement time	< 30 minutes	To minimize investigative interference.
Recalibration Period	≥ 12 months	Low maintenance requirement to ensure reliable and uninterrupted long-term operation.

**Table 1. Key Design Requirements for the Therminus-K3 Prototype**

temperature gradient across the test sample utilizing a Hot-Plate and Cold-Plate Assembly. The guard heaters, maintained at the same temperature as the main heater, are designed to minimize lateral heat loss, while the insulated back heater prevents backwards heat flow. This setup ensures a unidirectional heat flow through the sample, which is a key factor for obtaining precise thermal measurements. Thermal conductivity of the test sample is then calculated by measuring the heat input ( $\dot{Q}_m$ ) to the main heater, the temperature difference ( $\Delta T$ ) across the sample, the sample's thickness ( $L$ ), and the cross-sectional area of the main heater ( $A$ ), as defined in Equation 1 [14, 19].

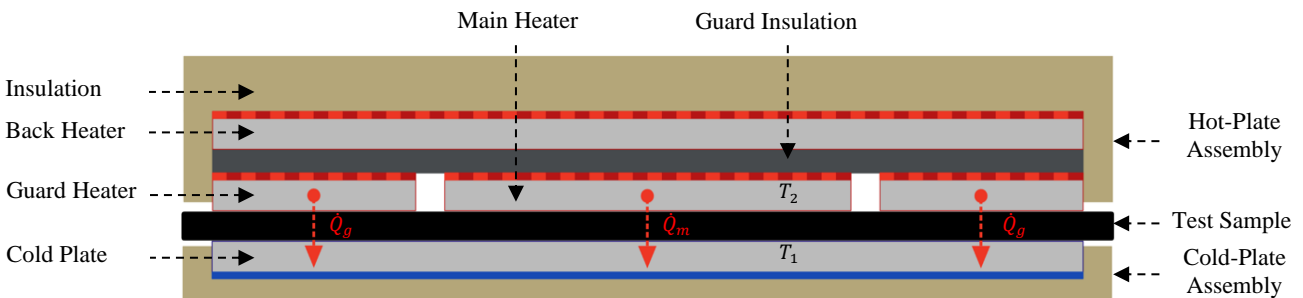
### 3. Design Requirements

The primary objective in designing the Therminus-K3 prototype emerged from a comprehensive literature study, highlighting the need for an modified GHP method [11]. This new approach aimed to combine the best attributes of both the Heat Flow Meter (HFM) method and traditional GHP methods. The developed prototype was to combine the portability, non-destructive nature, and rapid measurement capabilities typically associated with HFM methods with the unmatched accuracy of absolute GHP methods. This integration is critical for developing a tool that is not only efficient and user-friendly but also maintains the precision necessary for forensic investigations.

To ensure the prototype adheres to technical standards, an in-depth examination of existing standards for both HFM and GHP methods was conducted. This review focused on identifying the key parameters and specifications that the Therminus-K3 must meet to align with industry benchmarks. Incorporating the requirements from these standards was expected to ensure the reliability and accuracy of the prototype, thereby securing its usability in forensic applications.

In establishing the design criteria for the Therminus-K3 prototype, discussions were held with key stakeholders. These stakeholders included the Dutch Police (Department of Forensic Investigation), the Netherlands Forensic Institute (Department of Crime Scene Investigation), the Amsterdam Medical Centre (PHOEBE development team), and TU Delft (Department of Biomedical Engineering) [10]. Their insights and requirements significantly contributed to shaping the design requirements of the prototype, guaranteeing compliance with both technical specifications and the practical demands of forensic investigations.

Understanding the factors influencing thermal conductivity measurements was important in the design and construction of the Therminus-K3 prototype. Based on a previously conducted systematic review, the most significant factors



**Figure 1: General Arrangement of a Single Sample Guarded Hot Plate Apparatus**

Illustrates the arrangement of the back, guard, and main heater relative to the test sample and cold plate.  $\dot{Q}_m$  indicates unidimensional heat flow from main heater to cold plate. The temperature gradient over the sample is calculated using  $T_2$  for the hot plate temperature and  $T_1$  for the cold plate temperature.



affecting thermal conductivity measurements of textiles were identified [9]. These include fabric thickness, sample compression, and moisture content. Each factor substantially impacts thermal conductivity measurements, necessitating their careful consideration in the prototype's development. Consequently, the design and development of the Therminus-K3 prototype incorporated strategies to address these factors, leading to the establishment of the following requirements:

- **Thickness Measurement:** The prototype must be equipped with a precise and standardized mechanism for measuring fabric thickness, ensuring consistent repeatability and accuracy.
- **Compression Mechanism:** The prototype must include a mechanism for controlled sample compression, to simulate the fabric's compression condition as it was at the time of death.
- **Moisture Resilience:** Recognizing the diverse conditions in which clothing can be found at crime scenes, the device should be able to handle samples with varying moisture levels.

The key requirements for the Therminus-K3 prototype, based on stakeholder consultations, technical standards, and measurement factors, are presented in Table 1. These outline the critical attributes necessary for its effectiveness in forensic applications. For the complete list of requirements and more in-depth information on these specifications, please see Annex 1.

#### 4. Therminus-K3 Design-Methodology

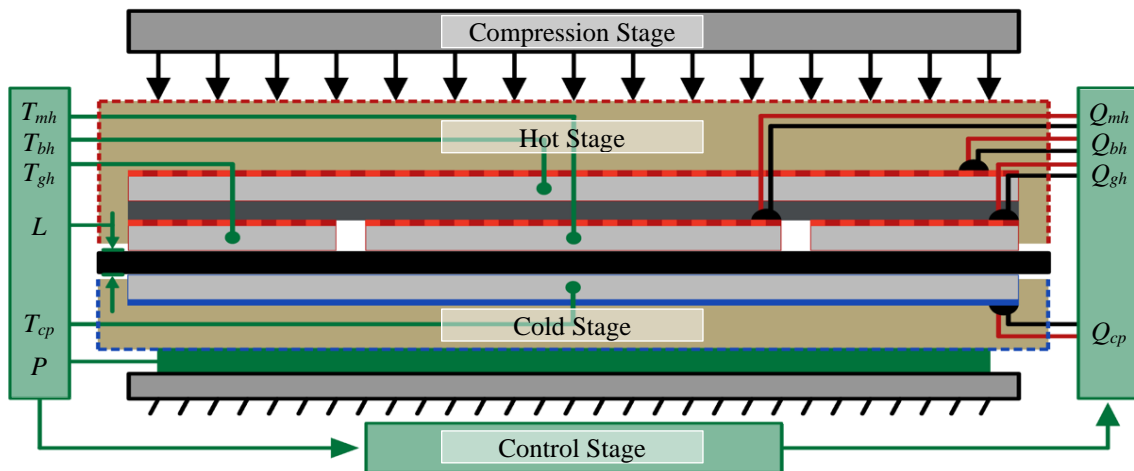
The design methodology for the Therminus-K3 prototype began with the establishment of the system architecture based on an in-depth analysis of the technical standards and the comprehensive list of requirements [18]. This architecture was then segmented into distinct subsystems, each designed

to fulfil a specific role within the overall system. The design process for each subsystem commenced with exploring potential design solutions, and a morphological chart was utilized to systematically evaluate and select the best design solutions. The chosen designs were then refined and modelled in SolidWorks [20], with thermal and static simulations supporting their development. Following this, prototypes for each subsystem were built and underwent real-world testing, enabling practical assessment and iterative enhancements. The final phase in the design methodology of the Therminus-K3 prototype involved the integration of all subsystems, ensuring they functioned cohesively within the overall design. This step involved a thorough evaluation, calibration, and necessary adjustments, to verify the prototype's effectiveness and reliability.

Maintaining consistency and building upon proven effectiveness, this third iteration of the Therminus device incorporated several design elements from its predecessors. Notably, the use of resistive heating for the heaters, Negative Temperature Coefficient (NTC) thermistors as temperature sensors, Peltier modules for cold plate cooling, and an Arduino Mega for central control were retained. These choices, as substantiated in a previous study [10], demonstrated their efficacy and were therefore carried over in the current design, ensuring the prototype benefits from established performance while integrating new advancements.

#### 5. Therminus-K3 Design

The Therminus-K3 prototype's system architecture, as shown in Figure 2, is composed of four integral subsystems. The Hot Stage is designed to generate unidirectional heat flow through the test sample, while the Cold Stage establishes the necessary temperature gradient to facilitate this heat flow. The Compression Stage replicates real-world compression conditions, and the Control Stage serves a dual purpose: it coordinates all other



**Figure 2: Therminus-K3 System Architecture**

System architecture is based on the general arrangement of a single sample guarded hot plate apparatus.  $T_{mh}$ ,  $T_{bh}$ ,  $T_{gh}$  represent the temperature of the main heater, back heater, guard heater in Kelvin, respectively.  $T_{cp}$  indicates the cold plate's temperature.  $L$  denotes the sample thickness in meters, while  $P$  signifies the applied pressure in Pascal.  $Q_{mh}$ ,  $Q_{bh}$ ,  $Q_{gh}$ , are the heat input to the main heater, back heater, and guard heater respectively.  $Q_{cp}$  represents the power to the cold plate.

subsystems while also gathering and processing the data required for calculating the thermal conductivity.

## 5.1 Hot Stage Design

### 5.1.1 Heater Design

The Hot Stage of the Thermanus-K3 prototype was designed to create a unidirectional heat flow through the test sample using three strategically dimensioned resistive heaters: a main heater, a guard heater surrounding the main heater, and a back heater positioned behind the main heater. These heaters were constructed from Printed Circuit Boards (PCBs) due to their affordability and proven performance in similar applications such as 3D printer heatbeds. PCB heaters operate by passing electrical current through an intricate copper trace pattern, generating heat through resistive heating. The flat and stackable nature of PCBs allows for a compact Hot Stage design, aligning with the portability objectives of the Thermanus-K3 prototype.

The previous prototype utilized commercially available heaters, with the main heater rated at 50W, the guard heater comprising four 10W heaters, and a 20W back heater. This configuration proved effective, heating up and reaching steady-state unidimensional heat flow within an acceptable time

frame [10]. Consequently, these power ratings served as a baseline for the design of the new PCB heaters.

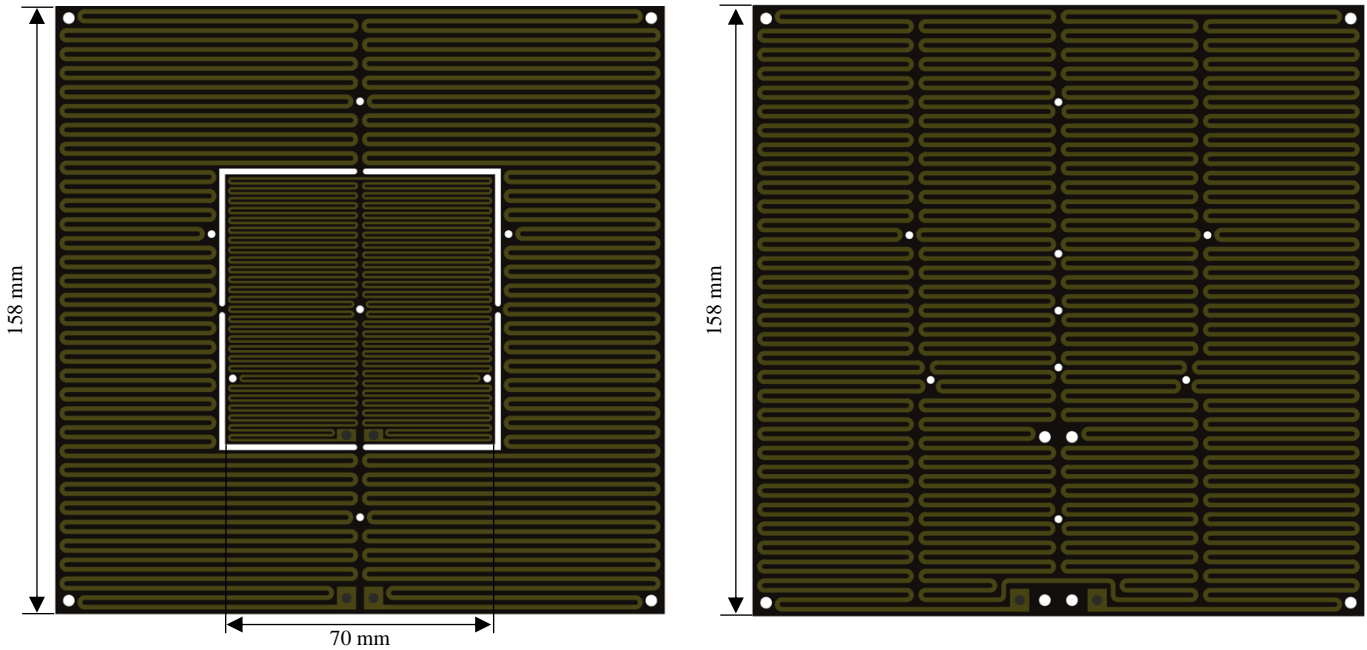
To design the PCB heaters, the required trace length for each heater was estimated using the power rating in combination with the trace resistance equation [21]:

$$R = \rho \cdot \frac{L}{T \cdot W} \cdot [1 + \alpha (T_{trace} - 25)] \quad [2]$$

Here,  $R$  represents the trace resistance in ohms ( $\Omega$ ),  $\rho$  is the resistivity parameter of copper at  $1.7E-5 \Omega \cdot \text{mm}$ ,  $L$  is the trace length in millimetres (mm),  $T$  is the trace thickness set at a standard  $35\mu\text{m}$  [22],  $W$  is the trace width in millimetres (mm),  $\alpha$  is copper's resistivity temperature coefficient at  $3.9E-3 1/^\circ\text{C}$ , and  $T_{trace}$  is the trace temperature in  $^\circ\text{C}$ . The equation can be rearranged to solve for trace length as follows:

$$L = \frac{R \cdot T \cdot W}{\rho \cdot [1 + \alpha (T_{trace} - 25)]} \quad [3]$$

The trace resistance is calculated from the power rating of each heater and the fixed operating voltage of 12V using Ohm's law. The trace temperature is set to  $50^\circ\text{C}$ , representing the maximum desired temperature increase of the heaters, thus leaving the trace width as the adjustable design variable.



Heater:	Main Heater	Guard Heater	Back Heater	Units
Estimated Trace Length	3782	8442	16833	[mm]
Chosen Trace Width	0.70	1.25	1.25	[mm]
Realized Dimensions	70 x 70	158 x 158	158 x 158	[mm]
Realized Trace Length	$\approx 3673$	$\approx 7642$	$\approx 9624$	[mm]
Realized Resistance	$\approx 2.5$	$\approx 2.9 \Omega$	$\approx 3.7 \Omega$	$[\Omega]$
Realized Power Rating	$\approx 58$	$\approx 50 \text{ W}$	$\approx 40 \text{ W}$	[W]

**Figure 3: PCB Heater Designs with Visible Copper Traces**

**Left:** Back view of combined main and guard heater PCB. **Right:** Back view back heater PCB. Corner holes are for mounting, while other holes indicate temperature sensor locations. See Annex 3 for more details. Power ratings from the Thermanus-K2 prototype guided trace length estimations using Equation 3. Final heater dimensions are based on trade-off between trace width, length, and heater dimensions. The back heater's realized power rating was increased due to space constraints limiting trace length extension.

To achieve unidimensional heat flow, it is crucial to size the guard and main heaters proportionally. Previous simulations have shown that a heater area ratio of 1:2:1 for guard-main-guard heaters maintains unidimensional heat flow across the desired sample thickness range [10]. Therefore, this ratio was also implemented in the new Therminus-K3 design. Furthermore, to thermally isolate the main heater from the guard heater, a gap was incorporated between the two. To comply with technical standards, the area of this gap must be kept below 5% of the main heater area [15]. Based on these design requirements and the estimated trace length, the dimensions and trace patterns of the heaters were designed. This design process, guided by the PCB manufacturer's design rules, also emphasized on minimizing the overall size of the heaters to ensure compactness, resulting in a more portable prototype suitable for on-site use.

This design process led to the PCB heater designs shown in Figure 3. These designs combine the main and guard heaters into one single PCB, not only reducing costs but also simplifying the overall design and the assembly process. This integration is achieved through the use of small bridges that link the main heater to the guard heater. These bridges are designed in accordance with technical standards [15] to minimize heat transfer between the two heaters.

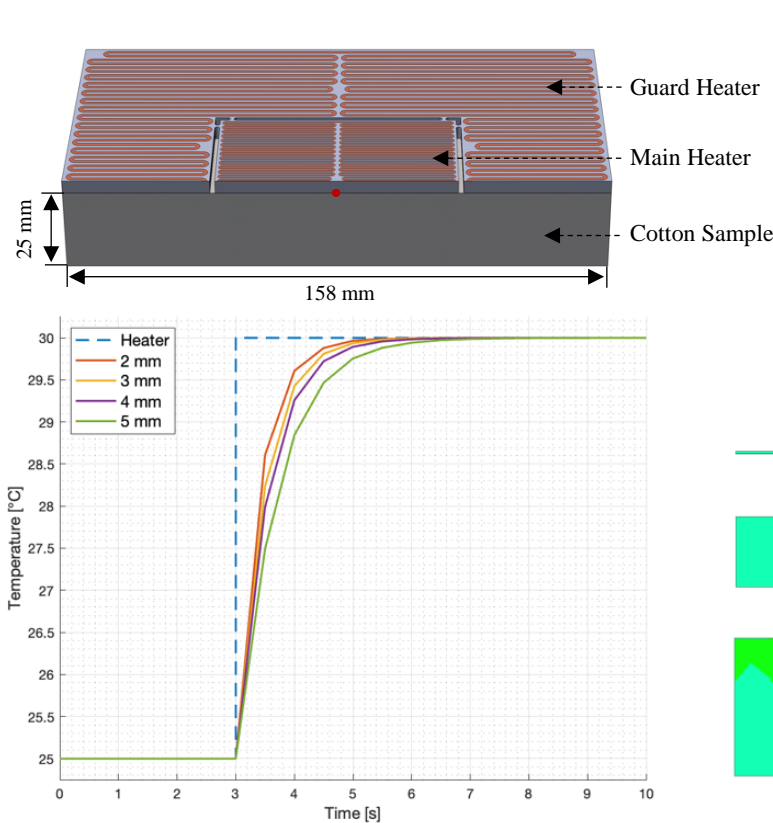
Furthermore, strategically positioned holes within the PCB heaters have been incorporated to facilitate

unobstructed wire routing for the temperature sensors monitoring the surface temperature. Additionally, gap imbalance sensors are placed along both sides of the gap in accordance with the technical standard, at a distance of one-fourth the side length from each corner [15], to monitor the temperature variations across the gap.

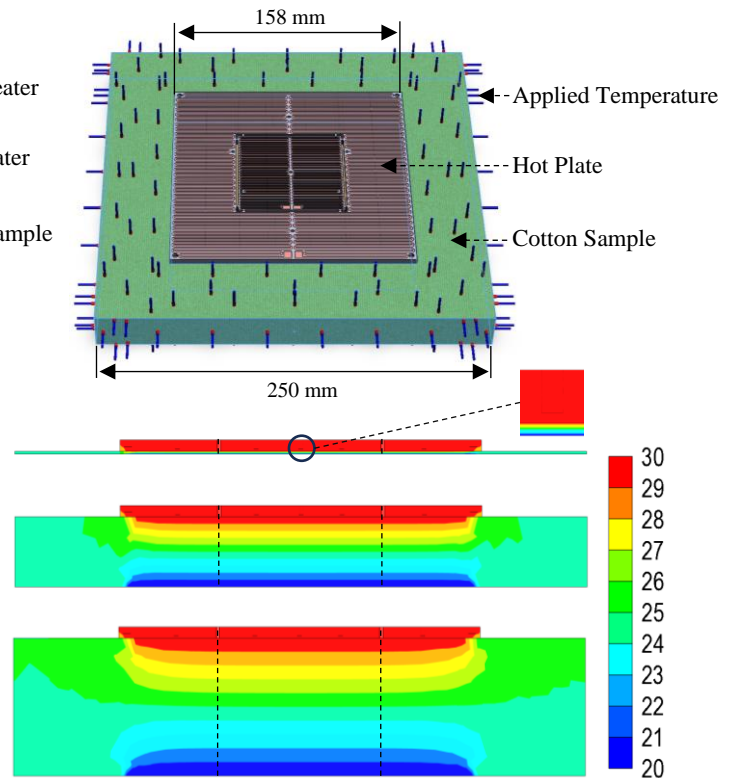
### 5.1.2 Surface Plates Design

To uniformly distribute the heat generated by the PCB-heaters, surface plates with a high thermal conductivity were added to each heater. This uniform distribution is critical in maintaining a consistent temperature across the contact surface with the test sample. Aluminium was selected as the material for these plates due to its affordability and good thermal conductivity. Between these surface plates and the heaters, thermal paste was applied to ensure maximum heat transfer. Additionally, these plates were designed to incorporate temperature sensors, vital for monitoring the temperature differential over the test sample. It is important that the integration of these sensors does not disrupt the thermal uniformity of the plates. As a result, designing the surface plates, including their thickness and the placement of temperature sensors, demanded careful consideration to maintain both consistent heat distribution and precise temperature measurements.

Achieving this balance involved determining the optimal thickness for the plates to house the thermistors while maintaining efficient thermal



**Figure 4: Surface Plate Temperature Response Simulations**  
**Top:** Cross-sectional view of the simulation model: 4 mm aluminum surface plate + 25 mm cotton sample, red dot indicates the location of the temperature probe. **Bottom:** Temperature response at probe location of surface plates with varying thickness ranging from 2 to 5 mm upon heat input from the heater.



**Figure 5: Side Guard Heater Necessity Simulations**  
**Top:** Non-destructive simulation model with applied temperature of 25 °C to cotton sample. **Bottom:** Steady-state thermal simulations for fabric samples of 1 mm, 25 mm, and 50 mm thicknesses, each with a thermal conductivity of 0.04 W/mK. Horizontal temperature lines indicate a steady state temperature distribution.

response. Although thicker plates promote thermal uniformity, these can negatively impact response time. To address this challenge, the design process involved SolidWorks [20] modelling of the main heater on the surface plates and conducting a series of thermal transient simulations. These simulations assessed the thermal response of plates with varying thicknesses, ranging from 2 mm to 5 mm. This range was chosen considering the diameter of the selected NTC thermistors, which was 1.2 mm [23]. Figure 4 illustrates the results, demonstrating that plates with thicknesses between 2 and 4 mm exhibited similar thermal responses, while the 5 mm plates exhibited a notably slower response. Consequently, 4 mm thick plates were chosen offering ample space for embedding the NTC thermistors while maintaining minimal lag in thermal response.

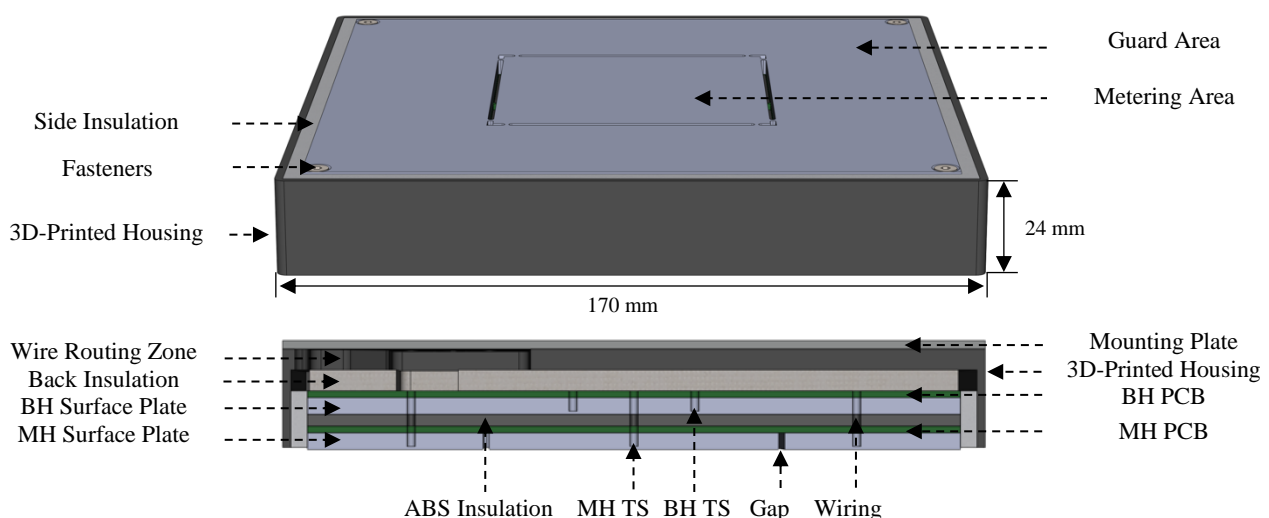
The next step involved determining the optimal depth at which to mount the thermistors. These needed to be positioned close enough to the surface of the surface plates to accurately measure the surface temperature while also maintaining sufficient distance to prevent any heat disturbances on the surface. First, an evaluation was performed to determine the depth at which the thermistors could still precisely measure surface temperature. Additionally, SolidWorks thermal steady-state simulations were conducted to examine the thermal uniformity across the surface plate to ensure that the thermistors would not affect the thermal homogeneity. Further details can be found in Annex 4. The conclusion drawn was that positioning the thermistors at a depth of 0.75 mm from the measurement surface had negligible impact on the thermal uniformity of the surface plate, while still allowing for the accurate surface temperature measurements with errors smaller than the minimum detectable temperature change of the temperature sensors. Therefore, a depth of 0.75 mm was chosen as the mounting depth. To improve measurement accuracy and minimize thermal resistance between the thermistor and the surface plates, they were attached using thermal paste.

### 5.1.3 Side Guard Heater Evaluation

For thicker samples, there is a significant risk of heat escaping from the sides. Therefore, an evaluation was conducted to determine if side guard heaters were necessary to mitigate this heat loss. This evaluation involved conducting SolidWorks simulations for fabric samples up to 50 mm thick. The inclusion of side guard heaters would impose constraints on the measurement process, necessitating a precise match of the sample's dimensions with the surface plates. However, if side guard heaters were determined to be unnecessary, the apparatus could facilitate non-destructive measurements with samples larger than the measurement surface. To explore this possibility, simulations considering non-destructive scenarios were conducted. All simulations were based on a worst-case scenario where every exposed edge of the fabric sample, not in contact with the Hot or Cold stages, was cooled to ambient room temperature, thus ensuring maximum potential heat loss. Further details of these simulations can be found in Annex 5. The simulation results, as depicted in Figure 5, suggested that additional guard heaters are not required, since steady-state conditions were successfully achieved in all simulations. In real-world applications, it is unlikely that the edges of a fabric sample would be cooled to such an extreme degree since the primary mechanism affecting the edges would be convection rather than direct cooling. This effect of natural convection around the edges of the sample would be less severe than the simulated worst-case scenario, thus supporting the decision to forego the installation of side guard heaters.

### 5.1.4 Final Hot Stage Design

The final Hot Stage design of the Therminus-K3 prototype, as shown in Figure 6, features a 3D-printed enclosure selected for its affordability and customizable design. This enclosure encases the heating stack, which consists of the PCB heaters and their corresponding surface plates separated by an ABS insulation layer. This layer enables a



**Figure 6: Hot Stage Design of Therminus-K3 Prototype**  
 BH: Back heater, MH: Main Heater, TS: Temperature sensor



measured thermal response from the back heater, while still providing some insulation. The whole stack is insulated using Spaceloft aerogel [24], effectively reducing lateral heat loss.

## 5.2 Cold Stage Design

The Cold Stage of the Terminus-K3 prototype is responsible for creating a consistent temperature gradient across the test sample. Its primary function is to maintain a constant temperature across the Cold Stage's surface plate to ensure a one-dimensional heat flow through the test sample under the hot stage's metering area.

### 5.2.1 Cooling Stack Design

In the design of the Cold Stage, cooling stacks employing Peltier modules are utilized to cool and maintain a low constant temperature across the cold surface plate. The choice of Peltier modules, also known as thermoelectric coolers, was driven by their distinct advantages over traditional, often water-cooled [25], systems. Their compact size, efficient cooling capabilities, and ease of integration significantly reduce the overall size and complexity of the Cold Stage, thereby enhancing the prototype's portability. Additionally, Peltier modules are capable of rapid temperature regulation, allowing for quick stabilization of the

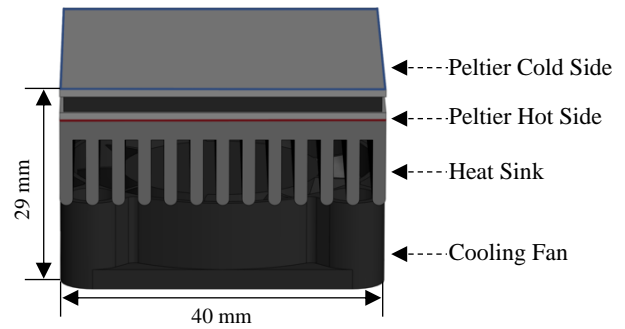


Figure 7: Peltier Cooling Stack Configuration

desired temperature, which is crucial for efficient on-site measurements.

Each cooling stack, as depicted in Figure 7 is composed of a TEC1-12706 Peltier module [26], paired with a heatsink and a fan. The Peltier module operates based on the Peltier effect, where the passage of an electric current causes one side of the module to absorb heat and the other to dissipate it. The heatsink, in conjunction with the fan aid in dissipating the heat from the hot side of the Peltier module, ensuring that the cold side remains at a steady, low temperature. The temperature of the cold stage is finely regulated by adjusting the power supplied to the Peltier modules, enabling precise temperature control.

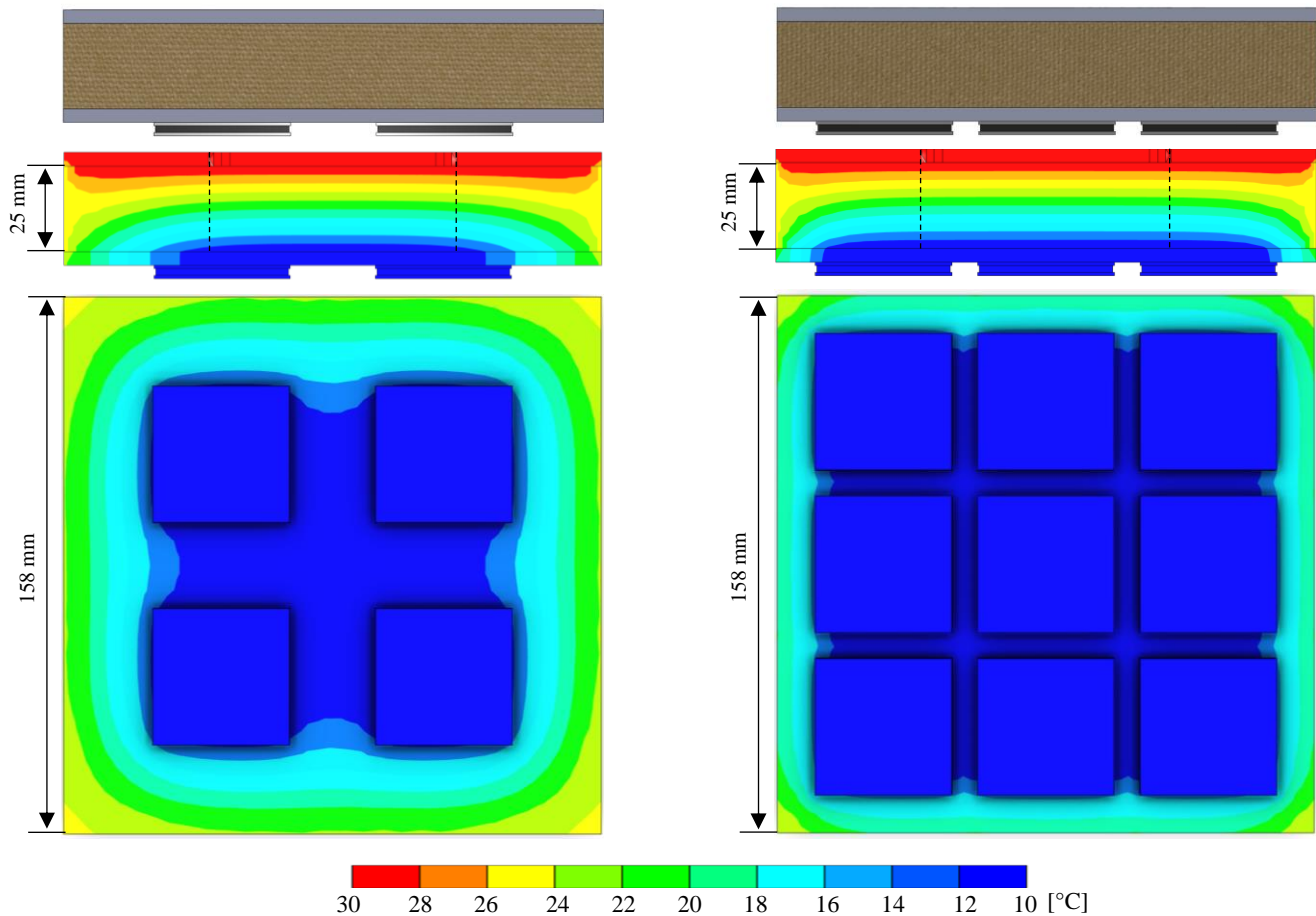


Figure 8: Comparative Thermal Simulations of Cooling Stack Arrangements

**Top:** 3D simulation model containing: Hot Stage surface plate, 25 mm fabric sample, Cold Stage surface plate and Peltier modules. **Middle:** Cross-sectional temperature distribution to analyze steady-state condition over the metering area between the dotted lines. **Bottom:** Temperature uniformity on the Cold Stage surface plate.

### 5.2.2 Cooling Stack Configuration

To determine the optimal number and arrangement of these cooling stacks, SolidWorks Thermal steady-state simulations were conducted. The simulation setup involved the Hot Stage main heater surface plate, a 25 mm cotton fabric sample, the cold stage surface plate, and various configurations of Peltier modules. The main heater was set to maintain a temperature of 30 °C, while the cold side of the Peltier modules were set to cool down to 10 °C. The open sides of the textile sample were cooled to room temperature to simulate the worst-case side cooling scenario.

Initial simulations with a 2x2 array of Peltier coolers revealed insufficient cooling to establish steady-state conditions across the metering section. This finding led to the exploration of a 3x3 cooler arrangement, as illustrated in Figure 8. This configuration proved successful in achieving the desired steady-state conditions, uniformly cooling the majority of the cold surface plate.

### 5.2.3 Final Cold Stage Design

The final design of the Cold Stage, as depicted in Figure 9, incorporates the 3x3 array of Peltier cooling stacks to ensure steady-state heat transfer across the measuring area. In this setup, the Peltier modules are organized in three parallel circuits, with each circuit comprising three in series-connected Peltier modules. Given the use of a 12V DC power supply, each module operates effectively at 4V. Each module then draws 1.5A according to the TEC1-12706 specifications [26]. In this configuration, the modules can establish a maximum temperature difference of about 45 °C between their hot and cold sides, provided that the heat on the hot side is dissipated. This temperature differential is more than sufficient for the Therminus-K3's requirements.

For temperature measurement consistency, the thermistors in the Cold Stage's surface plate are installed at the same depth as those in the Hot Stage. This ensures uniform temperature readings across both stages. The underside of the surface plate is

insulated with Styrofoam, reducing heat transfer to the cold plate from the hot sides of the Peltier elements. The design of the 3D-printed housing incorporates a ventilation mesh to improve airflow over the heatsinks. Additionally, the mounting plate is designed with air vents below each cooling fan to facilitate effective air circulation. The orientation of the cooling fans was evaluated to determine their most effective configuration. Tests compared a push configuration, where fans force air up through the heatsinks from beneath the Cold Stage, against a pull configuration, drawing air from the sides downwards. The push setup proved more effective, lowering the cold stage temperature by an additional degree. The complete testing methodology and results are detailed in Annex 6.

### 5.3 Compression Stage Design

The Compression Stage in the Therminus-K3 prototype is essential for applying controlled, uniform pressure to textile samples, accurately reflecting any changes in thermal conductivity that occur due to compression, as observed in real-world conditions.

#### 5.3.1 Compression System Design

In designing the Compression Stage of the Therminus-K3 prototype, an extensive concept generation process, detailed in Annex 7, was carried out. This led to the development of a system resembling a mechanical vertical vice, as depicted in Figure 10. This design uses a leadscrew (C), enabling the Hot Stage (D), mounted on linear guide rails (H), to move up and down above the Cold Stage (E). This movement is controlled by turning the handwheel (F), which engages the leadscrew with an M10 thread. As the handwheel is turned, the Hot Stage moves vertically, exerting a force onto the Cold Stage. This force is measured using a force sensor (I), and when this force data is combined with the dimensions of the Hot Stage, the pressure exerted on the sample can be calculated. With each full rotation of the leadscrew, the Hot Stage advances 1.5 mm vertically. Given that such a degree of movement is relatively coarse for a compression system, a geared input ratio of 1:0.5 is

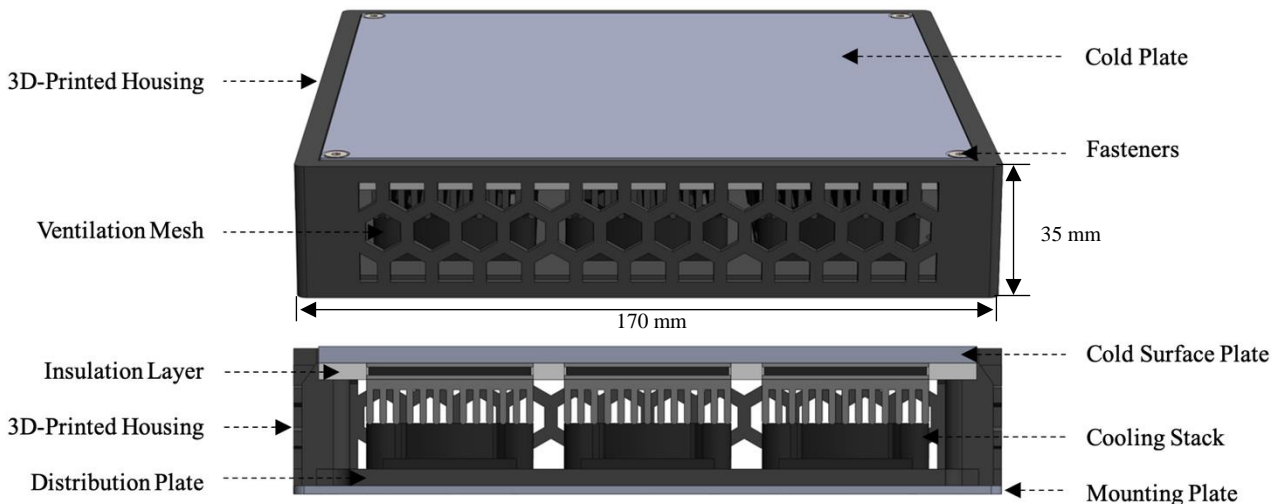
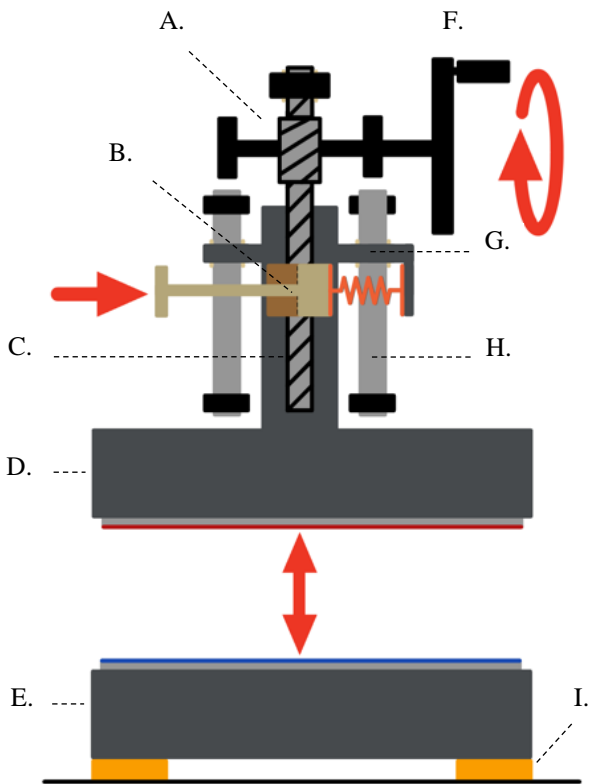


Figure 9: Cold Stage Design of Therminus-K3 Prototype





**Figure 10: Schematic of Compression Stage Concept**

A. Geared Transmission, B. Quick Release Mechanism, C. Leadscrew, D. Hot Stage, E. Cold Stage, F. Handwheel G. Linear Sleeve Bearing, H. Linear Guide Rod, I. Force Sensor

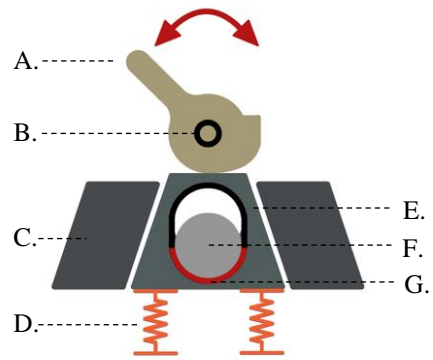
used. This modification allows for more refined control, enabling accurate application of force on the textile samples. This design allows for precise pressure control, essential for mimicking varied crime scene compression conditions.

### 5.3.2 Pressure Measuring System

To measure the force applied during compression, the design integrates strain gauge load cells. Four SC134 full bridge low profile load cells [27] are positioned at the corners of the Cold Stage's mounting plate for accurate force detection. A single HX711 Load Cell Amplifier board [28] compiles the readings into one force reading. This setup not only facilitates precise real-time force feedback but also allows for fine-tuning the compression force to align with specific forensic investigative scenarios. Calibration of the load cells was performed to guarantee that the force measurements are consistent and reliable, see chapter 6.4.

### 5.3.3 Quick Release Mechanism

A key usability feature of the Compression Stage is the incorporation of a quick-release mechanism, detailed as component B in Figure 10. The concept generation process for designing this feature can be found in Annex 8. The chosen concept, as illustrated in Figure 11 operates on a bistable switch principle, where the toggle switch (A) actuates the mechanism, allowing the Actuator Block (E) to swiftly engage with or disengage from the Leadscrew (F). This mechanism enables rapid and effortless engagement and disengagement with the



**Figure 11: Schematic of Quick Release Mechanism Concept**

A. Toggle Switch, B. Swivel Point, C. Linear Guide, D. Spring, E. Actuator Block, F. Leadscrew G. Half a nut to engage with the thread's leadscrew.

leadscrew, allowing for the fast loading and unloading of samples.

### 5.3.4 Thickness Measurement System

Integrated within the Compression Stage, the Therminus-K3 prototype features a thickness measurement system utilizing a linear slide potentiometer for effective and cost-efficient sample thickness measurements. This system operates by measuring the variable resistance of the slide potentiometer, which is directly related to the slider's position. The slider is connected to the movable Hot Stage, allowing it to effectively measure the distance between the Hot and Cold Stage's surface plates, which corresponds to the sample thickness during measurements. Notably, the potentiometer is only linear in its central range, leading to a compression stage design that avoids the potentiometer's ends. Through a calibration process, as detailed in chapter 6.2, these resistance values are mapped to corresponding thickness measurements, enabling precise and reliable assessment of sample thickness.

### 4.3.5 Final Compression Stage Design

The Compression Stage of the Therminus-K3 prototype, shown in Figure 12, combines cost-efficiency with custom-designed components. Its frame is built with off-the-shelf 20x20 mm aluminium extrusions (K), ensuring a robust and affordable structure. Complementing the frame are custom 3D-printed parts made from black PLA and laser-cut 4 mm steel sheet parts to reinforce critical areas. The design also features chrome-hardened steel rods (C) with polymer sleeve bearings ensuring smooth operation over its entire 50 mm range. Furthermore, static SolidWorks simulations, detailed in Annex 9, identified the need for a steel cover plate (F) on the Hot Stage (G) to evenly distribute load and prevent deformation of the 3D-printed Hot Stage housing. Additional simulations demonstrated that the 3 mm steel reinforcement plates for the gear housing (B) and linear carriage (E) can withstand a 300 N load, equivalent to about 12.000 Pa on the sample, thereby confirming their strength and minimal deformation under load. Additionally, a set screw (J) was added that allows for adjustability of the Hot Stage position so that the surface plate of the Hot Stage can be levelled with

the surface plate of the Cold Stage, reducing contact resistance, and enhancing measurement accuracy.

#### 5.4 Control Stage Design

The Control Stage of the Therminus-K3 prototype is responsible for key operations such as temperature measurements, thickness assessments, pressure calculations, and the regulation of the heaters in the Hot Stage and the Peltier modules in the Cold Stage. It also enables the display of the measurement data through an intuitive user interface module and supports either data storage or transfer to a computer for comprehensive analysis and thermal conductivity calculations. At the heart of this system is the Arduino Mega 2560 [29], chosen for its versatility, functionality, and affordability.

##### 5.4.1 Thermal Regulation System

The Therminus-K3 prototype's thermal regulation system is designed to independently control the temperatures of the main, guard, and back heaters, ensuring they operate within a tight margin of  $\pm 0.2^\circ\text{C}$  of each other, while also regulating the temperature of the cold stage. To achieve this precision while maintaining overall affordability, the system employs a self-developed PID (Proportional, Integral, Derivative) controller. This PID controller operates using Pulse Width Modulation (PWM), a method that effectively adjusts the duration of power pulses to control the amount of energy supplied. It modulates this power

through a fast-switching N-channel MOSFET (Metal-Oxide-Semiconductor Field - Effect Transistor). The MOSFET acts like a high-speed switch that turns the 12V 360W DC power supply on and off at a rapid rate. The PID controller continuously monitors the temperature of the Hot and Cold stage and tunes the on-off cycle of the corresponding MOSFETs accordingly, effectively controlling power delivery and therefore enabling precise temperature regulation. This approach negates the need for more complex, bulky hardware solutions such as relays, streamlining the system while maintaining high accuracy in temperature regulation.

Figure 13 showcases the PWM-controlled MOSFET circuit diagram used for precise power regulation. In this setup, the gate of the N-channel MOSFET is connected to the Arduino through a  $1\text{k}\Omega$  resistor, which limits the current to the gate, while a  $10\text{k}\Omega$  pull-down resistor ensures that the MOSFET remains off when not being actively driven by the PWM signal. The load, representing the heaters, fans, or Peltier elements, is connected in series with the MOSFET to the 12V supply, allowing for precise power modulation.

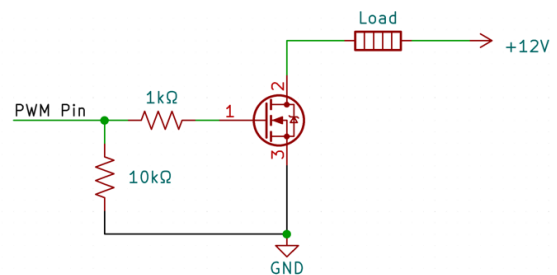


Figure 13: PWM-Controlled MOSFET Circuit Diagram  
1. Gate, 2. Drain, 3. Source.

##### 5.4.2 Measurement System

The temperature measurement system of the Therminus-K3 prototype employs 12 Murata  $10\text{ k}\Omega$  Negative Temperature Coefficient (NTC) thermistors [23]. In this system, each NTC thermistor is connected alongside a  $10\text{ k}\Omega$  resistor in a voltage divider circuit. As the thermistor's resistance changes with temperature, it alters the voltage output of the divider. To enhance the accuracy and stability of these measurements, a 5V Shunt Voltage Reference [30] was incorporated to create a stable voltage with a precision of  $\pm 0.1\%$ . The Arduino Mega then reads and processes these output voltage signals, converting them into temperature readings. This conversion was calibrated, as detailed in 6.1, to ensure the accuracy of the temperature measurements.

Similar to the temperature measurement system, the linear slide potentiometer used for thickness measurement operates based on variable resistance. It directly generates an output voltage proportional to its resistance. This output voltage, accurately stabilized by the 5V Shunt Voltage Reference, is read by an analog read pin on the Arduino Mega and converted into a thickness reading, enabling precise

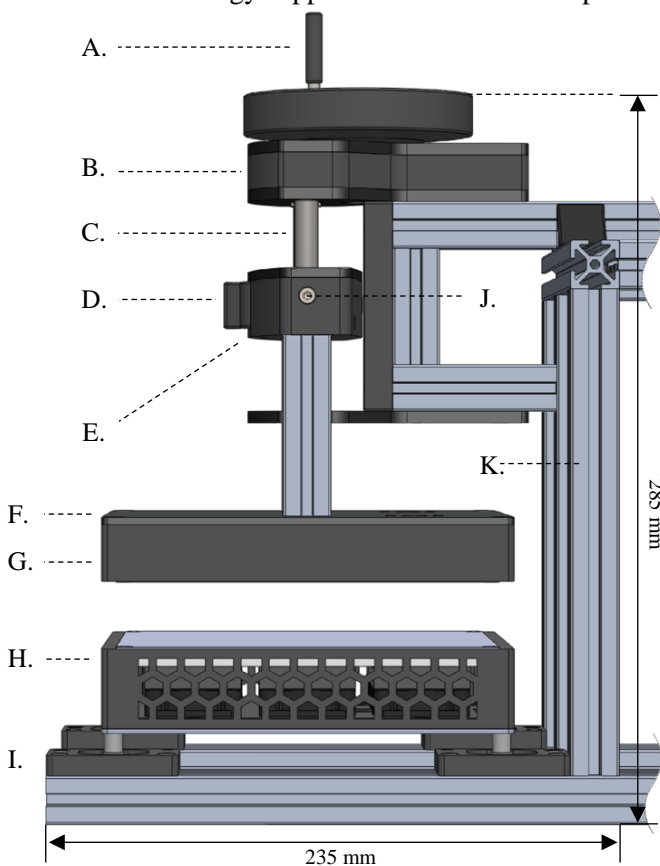
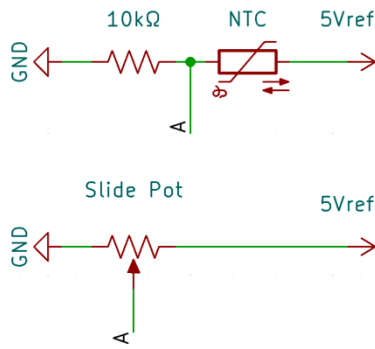


Figure 12: Compression Stage Design Therminus-K3 Concept  
A. Handwheel, B. Steel Reinforced Gear Housing, C. Chrome-hardened Steel Linear Rods, D. Quick Release Mechanism, E. Linear Carriage, F. Hot Stage Cover Plate, G. Hot Stage, H. Cold Stage, I. Loadcell Housing, J. Level Set Screw, K. Structural Framework.



**Figure 14: Temperature and Thickness Sensor Circuits Diagram**

**Top:** Single NTC thermistor-based temperature sensor circuit.  
**Bottom:** Slide potentiometer with variable resistance for thickness measurement A. Indicates analog read pin of Arduino Mega.

thickness measurements crucial for the device's functionality.

Figure 14 illustrates the respective wiring diagrams for both the NTC thermistor and the slide potentiometer circuits.

To accurately determine the power delivered to the main heater, an INA226 Power module was utilized [31]. This module incorporates an INA226 Bi-Directional Current and Power Monitor Chip, featuring an I<sup>2</sup>C Compatible Interface [32] which is capable of continuously measuring both current and voltage. Current measurements are made through a shunt resistor in series with the heater, while voltage is measured directly across the heater's power supply lines. The module's accuracy typically has an error margin of 0.02% to a maximum of 0.1% when properly calibrated. The power to the heater is calculated by multiplying the current with the voltage across the heater. For detailed information on the INA226 module's settings and power calculation method, please refer to Annex 10. The INA226's robust and independent measurement capabilities significantly reduce the computational load on the Arduino Mega. This advantage is further amplified by its internal averaging function, which calculates the average over a set number of readings, enhancing the reliability and stability of the measurements. Its communication with the Arduino is seamlessly managed via the I<sup>2</sup>C interface, utilizing serial data and clock pins for synchronized data transmission. This integration ensures a high level of precision in power management, crucial for the accurate functioning of the Therminus-K3 prototype.

As for the force measurements, they are directly derived from the HX711 Load Cell Amplifier which combines the signals from each load cell into a single reading. This load cell amplifier is connected to two digital pins of the Arduino, enabling precise measurement of the sample compression force when properly calibrated. Further details of this calibration can be found in chapter 6.4.

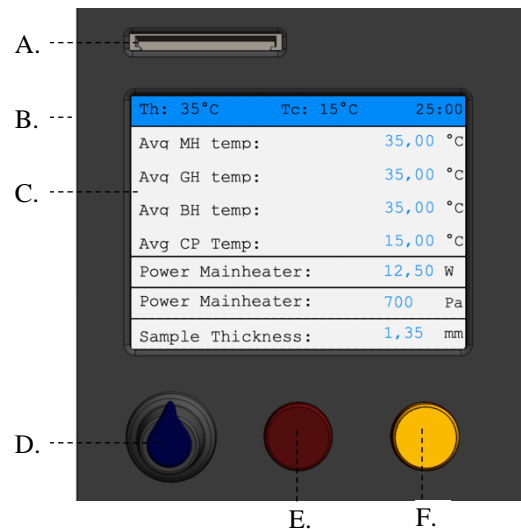
#### 5.4.3 User Interface Module

The Therminus-K3 prototype's user interface module is shown in Figure 15 and was designed

with simplicity in mind to be accessible for users without technical expertise. Central to this interface is a 2.8" ILI9431 LCD screen [33] (C), displaying all essential information clearly. Adjusting the temperatures for the Cold and Hot Stages is done using a rotary encoder button (D) that doubles as an enter key, a press confirms each of the temperature settings successively. Once the Cold Stage temperature is set, the interface is ready to take pressure and thickness measurements. Taring these sensors with the push buttons (E & F) is an important step, as it zeroes out any existing data, ensuring that subsequent readings are accurate for reliable thermal conductivity calculations. A final press of the encoder button starts the measurement process, streamlining the transition from setup to execution. One tare button (E) also acts as a cancel function, allowing users to stop an ongoing measurement and adjust settings as needed. All these components are encased in a custom 3D-printed housing (B), which ensures that the LCD-screen and buttons are securely mounted and can be easily integrated into the prototype's design.

#### 5.4.4 Arduino Mega PCB Shield

To effectively integrate and manage the various functionalities of the Therminus-K3 prototype, a custom-designed Arduino Mega PCB shield was developed. The layout of the PCB shield is illustrated in Figure 16, while a comprehensive circuit diagram of the PCB is provided in Annex 11. This shield directly attaches to the Arduino Mega, which eliminates the need for a breadboard, simplifying the overall design and the assembly process. The PCB shield features screw terminal connectors for the heaters, Peltier modules, and fans (A) as well as for the power supply input (I). JST XH crimp connectors [34] are used for all the sensors (F, G, H) and button connections (E). This choice of connector simplifies the process of debugging and replacing components. The PCB furthermore features an octal bus transceiver (D) to convert the 5V logic of the Arduino Mega to the



**Figure 15: User Interface Module**  
**A.** SD-Card Reader, **B.** 3D-Printed Housing, **C.** 2,8" LCD Screen, **D.** Rotary Encoder Button, **E.** Pressure Sensor Tare/Cancel Button, **F.** Thickness Sensor Tare Button.

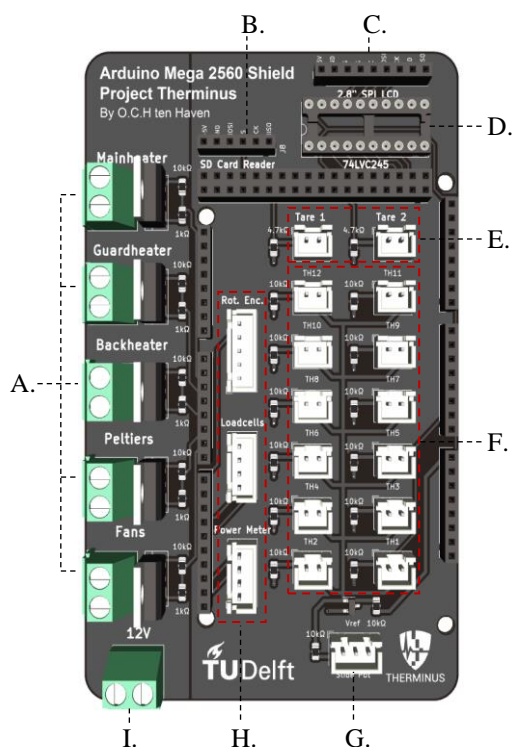


required 3.3V of the LCD-Screen [35]. This shield not only enhances the prototype's operational efficiency but also contributes to its overall reliability and ease of maintenance. The accompanying Arduino script, which outlines the PID controller's logic and the rest of the prototype's functionality, is available in the Supplementary Materials.

#### 5.4.5 Data Post Processing

The Thermanus-K3 prototype is configured to log the essential measurement data, instead of performing direct thermal conductivity calculations. This data, including applied pressure, sample thickness, temperatures of the Hot and Cold Stages, and the heat input to the Main Heater, can be stored on an SD card via the User Interface Module's SD card reader, or transmitted directly to a computer in real-time through a USB. This approach of logging data instead of converting it directly to the thermal conductivity is vital in maintaining transparency and accuracy, both of which are essential in forensic investigations.

For thermal conductivity analysis, a custom postprocessing MATLAB [36] script is used, which can be found in the Supplementary Materials. It determines when steady state conditions are reached and subsequently calculates the thermal conductivity using the logged data. The calculations are based on the one-dimensional form of Fourier's Law of heat conduction, as outlined in the Theoretical Background.



**Figure 16: Custom Arduino Mega PCB Shield**

A. PWM Power Outputs for Heaters, Peltier Modules and Fans, B. SD-Card Reader Connector, C. LCD-Screen Connector, D. Dip 20 Socket for Octal Bus Transceiver, E. Push button Connectors, F. Thermistor Connectors, G. Slide potentiometer Connector, H. Rotary encoder, HX711 Loadcell Amplifier and INA226 Power Meter Module connector, I. Power Supply Input.

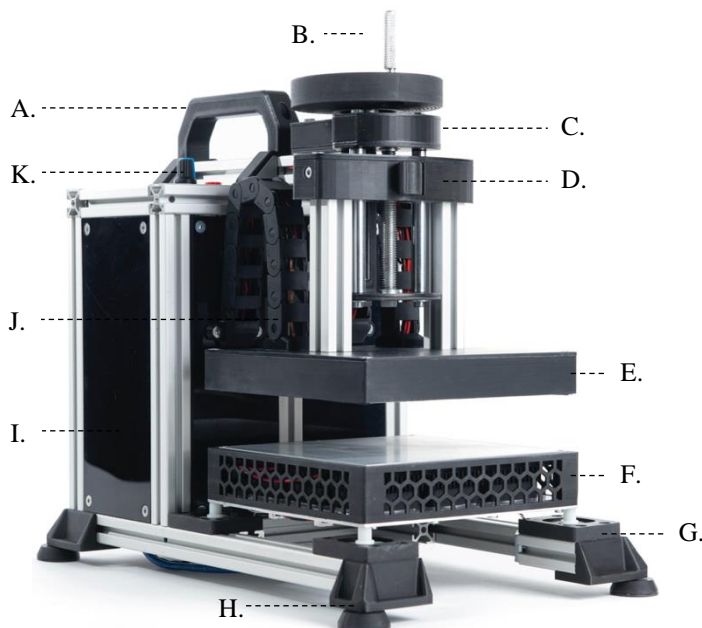
## 6. Thermanus-K3 Prototype and Calibration

The culmination of the design phase led to the successful fabrication of the Thermanus-K3 prototype, showcased in Figure 17. The prototype features a frame made from aluminium extrusions, providing structural stability while also making up the Electronics Housing (I). Within this housing, the Arduino Mega, the Custom PCB shield, the User Interface Module (K), and the power supply are securely enclosed by removable 4 mm thick PMMA panels, ensuring both durability and easy access. The design furthermore includes cable drag chains (J) to neatly organize and integrate power lines and temperature sensor wires from the Hot Stage (E) to the Electronics Housing. Additionally, a handle (A) and adjustable feet (H) were added for portability and usability in various field conditions. In the electrical design, measures were taken to optimize measurement quality. The surface plates of both the Cold and Hot Stages were grounded to reduce noise, and all sensor wires were twisted to minimize electromagnetic interference and crosstalk. The Thermanus-K3 prototype, measuring 330 x 200 x 310 mm and weighing 7.2 kg, was built at a material cost of €404.39. Additional photographs and the full Bill of Materials are detailed in Annex 12 and 13.

To ensure the Thermanus-K3 prototype operates accurately and reliably, a thorough calibration and validation process was conducted for all subsystems and sensors, as described in the following sections.

### 6.1 NTC Thermistors Calibration

The 12 NTC thermistors were calibrated by immersing them in a stirred ice water bath alongside two reference thermistors connected to a Greisinger G1700 thermometer [37]. Warm water was

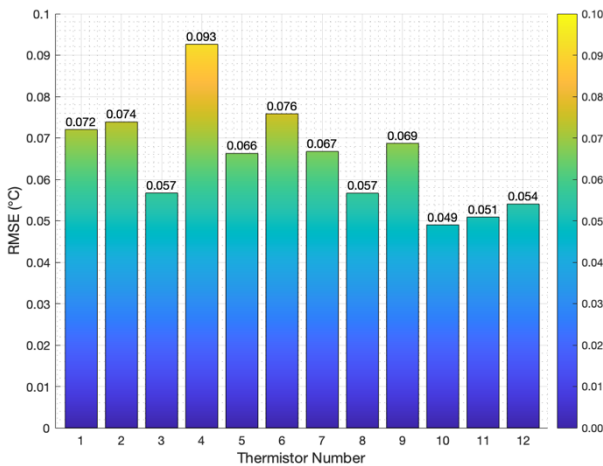


**Figure 17: Thermanus-K3 Prototype**

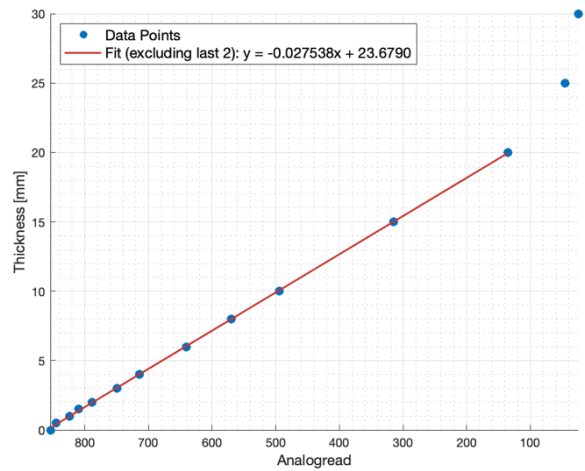
A. Handle, B. Handwheel, C. Gear Housing, D. Linear Carriage Including Quick Release Mechanism, E. Hot Stage, F. Cold Stage, G. Loadcell Housing, H. Adjustable Feet, I. Electronics Housing, J. Cable Drag Chain, K. User Interface module.  
Photo credit: A.J. Loeve.

gradually added in 2-degree increments ranging from 2 to 50 degrees Celsius. At each temperature increment, the reference temperature was recorded concurrently with the voltage outputs of each thermistor. An average of four consecutive voltage readings, taken within a second, was used to represent the voltage signal at each temperature point. The reference temperature was determined by averaging the readings from the two thermistors of the reference thermometer. To convert the voltage measurements from the thermistors into temperature values, a fourth-order polynomial fit was applied to the voltage-temperature data, as it demonstrated to have the most accurate fit among different orders. A more conventional Steinhart-Hart fitting model [38] was also considered, but it showed negligible difference in accuracy compared to the polynomial fit. The polynomial model was preferred for its simpler integration, eliminating the need to first convert voltage readings into resistance values.

The calibration was validated by repeating the procedure, confirming the precision of the polynomial fit. Figure 18 illustrates the Root Mean Square (RMS) error for each of the 12 thermistors, calculated over the entire temperature range and compared against the temperatures measured by the reference thermometer. The overall average RMS error was 0.056°C, suggesting high accuracy in the temperature measurements. Additionally, Figure 19 illustrates the relative measurement consistency across the thermistors. This colour matrix represents the pairwise RMS differences between the temperature readings of each thermistor pair. Most thermistor pairings show close agreement, affirming the uniformity of the calibration. This detailed error mapping strengthens confidence in the thermistors' performance. The resolution of the temperature measurements was calculated using the minimum detectable voltage reading and the polynomial fits, leading to a resolution of 0.087 °C. Annex 14 contains the full calibration procedure and the associated data.



**Figure 18: Calibration Precision of NTC Thermistors**  
RMS Error of each thermistor relative to reference temperature measurements, calculated 20 data points corresponding to the temperature increments.

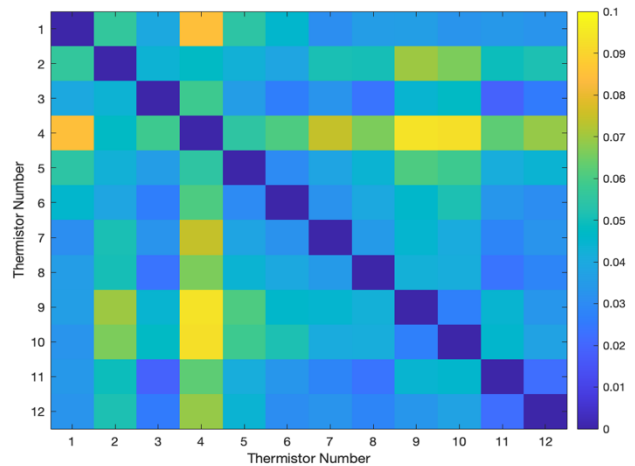


**Figure 20: Thickness Sensor Calibration**

The curve illustrates the linear potentiometer's response to known thicknesses. The linear regression fit excludes the last two data points due to their non-linear response.

## 6.2 Thickness Measurement Calibration

To calibrate the linear potentiometer used for thickness measurements, Cary Le Locle Suisse gauge blocks [39] with a precision of  $\pm 0.15 \mu\text{m}$  [40] and thicknesses ranging from 0.50 mm to 50.00 mm were measured. The calibration process began with the surface plates touching to establish a 0.00 mm baseline. Subsequently, the gauge blocks were placed between the surface plates, and the potentiometer's output was incrementally measured at each specific gauge block thickness. This process resulted in a calibration curve as depicted in Figure 20. A linear regression fit was then applied to the linear section of the data, enabling the conversion of the potentiometer's variable resistance readings into thickness measurements. The accuracy of this calibration was verified by re-measuring the thickness of the gauge blocks, yielding an RMS error of just 0.0132 mm. Furthermore, the sensor's resolution, determined using the linear regression fit and the minimum differential in analog read values, was determined to be 0.0275 mm. For more comprehensive calibration data, please see Annex 15.



**Figure 19: Inter-Thermistor Measurement Consistency**  
RMS difference matrix between thermistor pairs in °C.

### 6.3 Power Sensor Calibration

The calibration of the INA226 Power meter module focused exclusively on the current sensor, as the voltage sensor's inherent accuracy as claimed by the manufacturer eliminated the need for its calibration [32]. To calibrate the current sensor, a Fluke 115 True RMS Digital Multimeter [41], capable of measuring the average current of signals, including block signals, with an accuracy of  $\pm 1\%$  from the true value, was connected in series with the heater and the INA226 module. The power to the main heater was then incrementally increased by adjusting the duty cycle of the PWM signal that controls the main heater circuit. At each increment, current readings were simultaneously recorded from both the Fluke Multimeter and the INA226 Module. A linear regression was performed on the current data, allowing for the adjustment of the INA226 current readings. To verify the accuracy and reliability of this calibration, the procedure was repeated. The INA226 module demonstrated a resolution of 0.1mA, and the RMS error between the calibrated INA226 readings and the reference readings, based on 28 current readings, was 0.424mA. For a comprehensive overview and detailed data of this calibration process, see Annex 16.

### 6.4 Load cell Calibration

The SC134 load cells were calibrated by determining a scaling factor for converting the raw output of the HX711 Load Cell Amplifier into kilograms. Initially, the load cells were zeroed to establish a baseline, which accounted for the weight of the compression stage itself, ensuring no inherent bias or unloaded weight affected the measurements. Subsequently, a known weight was placed on the load cells, and the output from the load cell amplifier, minus the tare value, was recorded. The scaling factor was then calculated as the ratio of this known weight to the net amplifier's output. It was assumed that the relationship between the load cell output and the applied weight is linear over the entire measurement range, since the manufacture specified a full-scale linearity of  $\pm 0.05\%$  [27]. To

validate the accuracy of the calibration, various weights were measured using both the Cold Stage with the calibrated load cells and a Kern EMB5.2K5 scale [42] for comparison. The validation measurements yielded a weight measurement accuracy of less than 0.2%. For more details on this calibration, refer to Annex 17.

## 7. Therminus-K3 Performance Analysis

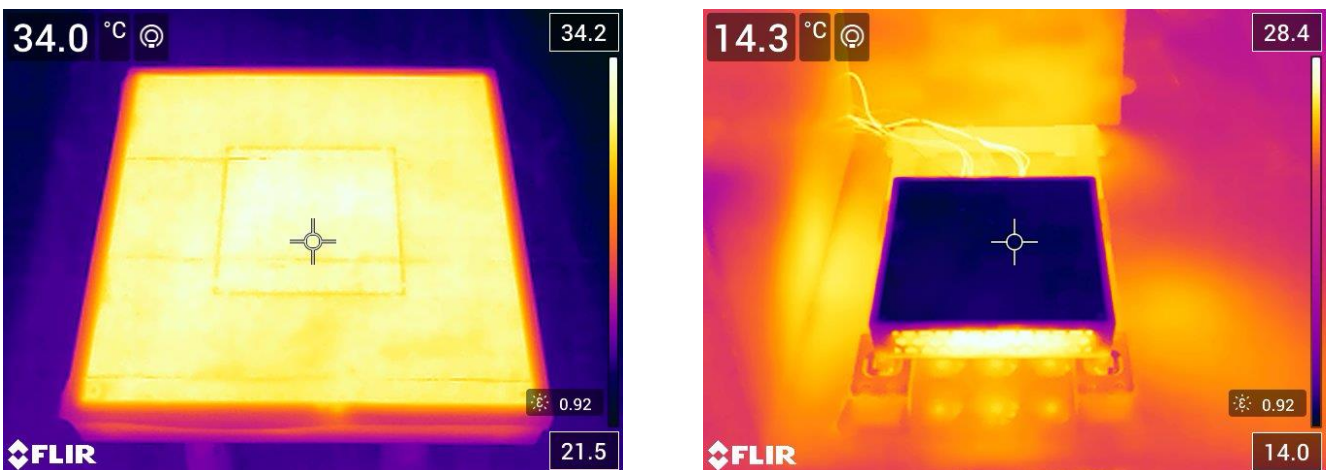
Prior to the measurement performance analysis of the Therminus-K3 prototype, certain key features were verified. The thermal uniformity of the surface plates of the Hot and Cold Stage and the accuracy of the thermistors in measuring the surface temperatures were checked. These initial checks were conducted to ensure the reliability of the performance tests that followed.

### 7.1 Thermal Uniformity Verification

To verify the thermal uniformity of the Therminus-K3 prototype's surface plates, a Flir E75 Advanced Thermal Camera [43] was utilized. The camera was used to take thermal images of both the Hot and Cold Stage's surface plates, effectively pinpointing and visually emphasizing any temperature variations, thereby enabling the assessment of the plates' uniform thermal distribution. To counteract the thermal reflective nature of the bare aluminium plates, painters' tape was applied to their surfaces prior to taking the thermal images, ensuring a non-reflective, dull finish. The resulting images, as shown in Figure 21, displayed no noticeable hot or cold spots, suggesting an even distribution of temperature across the surface plates. This outcome affirmed the effective thermal uniformity of the plates, critical for the accurate functioning of the prototype. For a more detailed description of the setup and methodology, see Annex 18.

### 7.2 Surface Temperature Measurement Verification

To check if the embedded thermistors in the Therminus-K3 prototype accurately measure the surface temperature of the surface plates, a comparison test was conducted. This test involved activating the cold stage and comparing the



**Figure 21: Therminus-K3 Thermal Homogeneity Assessment**

**Left:** Infrared thermal image of the Hot Stage surface plate demonstrating uniform heat distribution.

**Right:** Infrared thermal image of the Cold Stage surface plate illustrating consistent cold temperature distribution.



Material:	L: [mm]	P: [Pa]	$\Delta T$ : [°C]	$T_m$ : [°C]	$t_{ss}$ : [s]	k: [W/mk]	STD:	Precision: %	Accuracy %
Silicone	1.76	712	10.48	29.77	336.71	0.2437	0.001072	0.44	1.54%
Cotton	0.52	707	12.02	28.99	400.71	0.0450	0.0003237	0.72	-

**Table 2: Average Measurement Results**

**L:** Sample Thickness, **P:** Applied pressure,  **$\Delta T$ :** Temperature difference across the sample,  **$T_m$ :** Mean measurement temperature,  **$t_{ss}$ :** time to steady state heat transfer, **k:** Thermal Conductivity, **STD:** Standard Deviation. Data of independent measurements can be found in Annex 21.

temperature readings from the embedded thermistors with those from two reference thermistors attached to its surface plate. This comparison provided a clear evaluation of the thermistors' performance in reflecting the actual surface temperatures. The results demonstrated a strong agreement between the two sets of thermistors, with an RMS error of only 0.09 °C. This level of error is expected, given the limitations of the test setup: the thermistors were taped on the surface plate without any thermal grease and their round shape provided only a small contact point, leading to some expected discrepancy. Further details about this test and its methodology can be found in Annex 19.

### 7.3 Performance Measurements

To evaluate the performance of the Therminus-K3 prototype, two different materials were measured: a 1.76 mm thick 100% silicone rubber sheet [44] and a coat made of 0.52 mm thick 100% cotton fabric [45]. Silicone was selected as the reference material for its homogeneous nature and known thermal conductivity of 0.24 W/mK [46, 47]. The cotton fabric, representing a relevant material for practical applications, was assessed non-destructively to preserve the coat's integrity. This approach ensured that the cotton measurements reflected the prototype's application in actual forensic settings. Between each successive set of measurements, the material was removed and then repositioned on the apparatus. This step ensured each reading was independent of the previous one in order to guarantee the reliability of the test results. The full

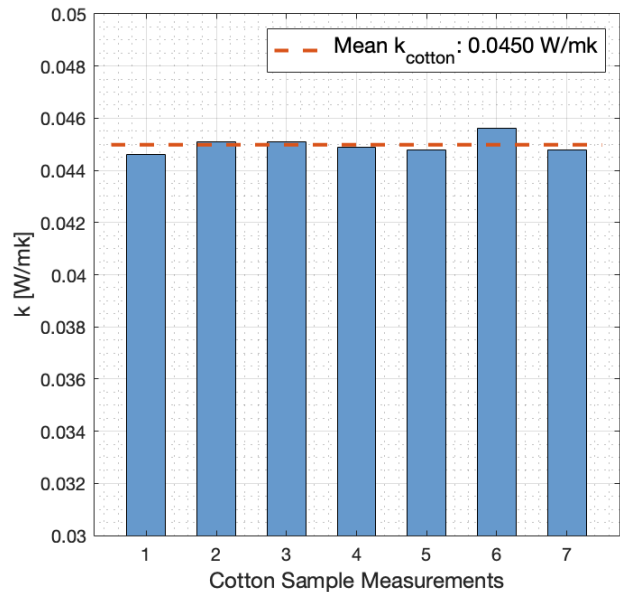
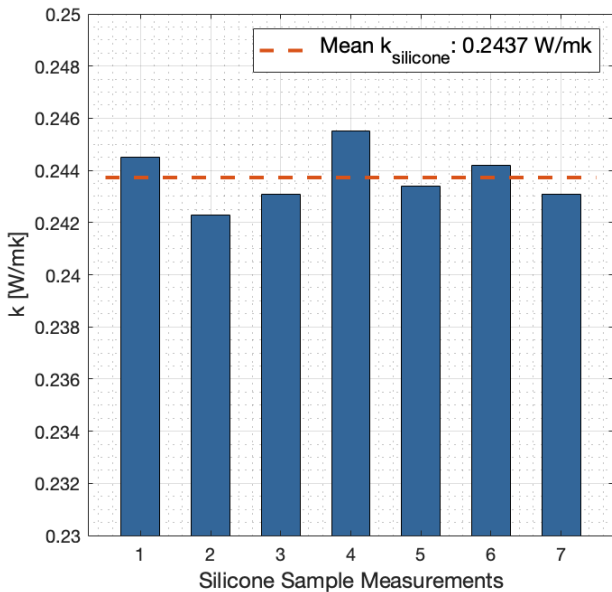
measurement procedure of the Therminus-K3 prototype can be found in Annex 20. The results of the independent tests are depicted in Figure 22. The average of the measurements was used to calculate the standard deviations, which were then expressed as a percentage of repeatability, indicating the precision of the prototype, see Table 2.

### 7.4 Measurement Uncertainty

The measurement uncertainty of the Therminus-K3 prototype, denoted as  $u_k$ , quantifies the confidence in the accuracy of its thermal conductivity measurements. This uncertainty is the cumulative effect of the uncertainties associated with each of the input parameters used in the calculation of thermal conductivity as per Equation 3. The overall uncertainty is calculated using the law of error propagation, which employs the RMS approach to combine individual uncertainties, as described in Equation 4, following[48]. This method effectively sums up the uncertainty of the input parameters to provide a single value that indicates the overall trustworthiness of the measurement.

$$u_k = k \cdot \left[ \left( \frac{u_V}{V} \right)^2 + \left( \frac{u_I}{I} \right)^2 + \left( \frac{u_{LS}}{LS} \right)^2 + \left( \frac{u_L}{L} \right)^2 + \left( \frac{u_W}{W} \right)^2 + \left( \frac{u_{\Delta T}}{\Delta T} \right)^2 \right]^{\frac{1}{2}} \quad [4]$$

In this equation,  $u_V$ ,  $u_I$ ,  $u_{LS}$ ,  $u_L$ ,  $u_W$ , and  $u_{\Delta T}$  are the uncertainties related to voltage, current, sample thickness, measurement area (length and width), material thickness, and temperature difference,



**Figure 22: Therminus-K3 Thermal Conductivity measurement Results**  
See Annex 19 for full measurement data.

Source $X_i$	Units	Measurement Estimate $x$	Uncertainty $u(x)$	Relative Uncertainty $u(x)/x$	Uncertainty contributed %
Voltage	[V]	12.00	$\pm 5.000E-03$	4.167E-04	0.08
Current	[A]	0.1845	$\pm 4.000E-04$	2.168E-03	2.21
Thickness	[m]	0.001	$\pm 1.375E-05$	1.375E-02	88.82
Length	[m]	0.07018	$\pm 5.000E-06$	7.125E-05	0.002
Width	[m]	0.07016	$\pm 5.000E-06$	7.127E-05	0.002
Temperature	[°C]	10.00	$\pm 4.350E-02$	4.350E-03	8.89

Property	Units	Measurement Estimate	Uncertainty	Uncertainty in %
Thermal Conductivity	[W/mK]	0.0450	$\pm 6.560E-04$	$\pm 1.46$
Thermal Resistance	[K/W]	0.0222	$\pm 1.085E-04$	$\pm 0.49$

Table 3: Measurement Uncertainty of the Therminus-K3 Prototype Based on Estimated Measurements

respectively. The corresponding measurement estimate values for these parameters are denoted by V, I, Ls, L, W, and  $\Delta T$ . Each of the uncertainties was calculated as half the corresponding measurement resolution, a common way for estimating measurement uncertainty [49]. This approach is particularly justified in this case, as the resolution was typically coarser than the precision of the measurements. The expected measurement uncertainty was determined to be 1.46%, as detailed in Table 3. The precision of the sample thickness measurement was identified as the largest contributing factor. Consequently, the concept of measuring the thermal resistance instead of the thermal conductivity was investigated as an alternative metric. Unlike thermal conductivity, thermal resistance, as expressed in Equation 5, is not dependent on the specimen's thickness and can therefore be a more accurate parameter in cases where sample thickness measurement precision is the limiting factor. The measurement uncertainty associated with thermal resistance was calculated to be only 0.49%. For a detailed breakdown of each uncertainty component and the methodology used in this analysis, see Annex 22.

$$R = \frac{A\Delta T}{\dot{Q}} \quad [5]$$

### 7.5 Therminus-K3's Impact on PMI Estimation

To assess the potential impact of the Therminus-K3 prototype on PMI estimations, body cooling simulations were conducted utilizing the PHOEBE model. The results are illustrated in Figure 23, which presents two body cooling curves over time. Each curve is based on a distinct value of thermal conductivity for cotton clothing: a commonly used standard value ( $k = 0.04 \text{ W/mk}$  [50]) and a more precise value as measured with the Therminus-K3 device ( $k = 0.0451 \text{ W/mk}$ ). The PMI points on these curves represent the estimated time required for the body to cool off to that specific temperature based on the respective  $k$ -value of the clothing. This comparison highlights how minor differences in thermal conductivity can significantly affect PMI estimations. Notably, the simulations show that the discrepancy in PMI estimations varies, ranging from about 10 minutes for a PMI of approximately 7:30 hours (450 minutes) to as much as 34 minutes for a PMI of around 22:54 hours (1374 minutes). This variation suggests that the variance in PMI

estimations increases with longer PMI durations, underscoring the importance of accurate thermal conductivity measurements in forensic investigations. Further details about the simulations can be found in Annex 23.

## 8. Discussion

This study focused on the development and evaluation of the Therminus-K3 prototype, an innovative device designed for on-site thermal conductivity measurements of clothing fabrics in forensic settings. Aiming to enhance the accuracy of PMI estimations, the Therminus-K3 employs the industry-standard GHP method and distinguishes itself as a portable yet affordable solution. Unlike traditional GHP apparatuses, which can have long measurement durations of several hours [25], this prototype's compact Hot Stage with PCB heaters and Cold Stage with Peltier cooling stacks achieve steady-state conditions rapidly, within just 6 minutes. Consequently, it enables thermal

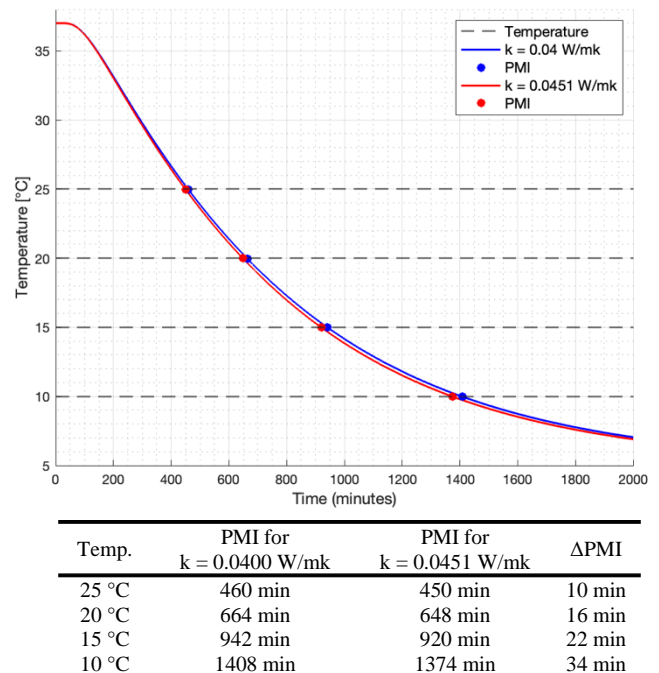


Figure 23: Effect of Measured  $k$ -Value on PMI Estimations  
Graph shows the decline in rectal core temperature as calculated by the PHOEBE model, using both a standard thermal conductivity value for cotton and a measured value from the Therminus-K3 prototype.

Table displays the impact of the measured  $k$ -value at various hypothetical PMI's.

conductivity measurements in under 25 minutes, which is significantly faster than typical, offering a distinct advantage in time-sensitive forensic contexts. This efficiency, along with the Compression Stage's ability to replicate real-world compression conditions and the possibility to make non-destructive measurements, underscores the prototype's suitability for forensic applications.

In addition to its speed, the Therminus-K3 has demonstrated promising results in its initial evaluations, achieving a measurement uncertainty in thermal conductivity below  $\pm 1.5\%$  and a precision below  $\pm 1\%$ . Building on these promising results, preliminary PMI simulations indicate that the enhanced thermal conductivity measurements from the Therminus-K3 could lead to more accurate PMI estimations. These simulations reveal a potential improvement of up to 34 minutes in PMI accuracy, highlighting the prototype's significance in forensic contexts.

The Therminus-K3's compact design, featuring adjustable feet and a carrying handle, makes it a portable device suitable for various field scenarios. Its affordability, with a material cost of just €404.39, makes it more accessible, while the simplistic user interface contributes to straightforward operation. These features make the Therminus-K3 an effective tool for improving PMI estimations in forensic investigations, striking a balance between efficiency, ease of use, and cost-effectiveness.

Furthermore, for a detailed comparison of the Therminus-K3 prototype against its original design requirements, please refer to Annex 24. This section provides an in-depth overview of how the prototype's performance aligns with the set objectives.

### Limitations

The Therminus-K3 prototype also faces certain limitations. Its measurement uncertainty is constrained by the thickness measurements precision due to its limiting resolution. This limitation highlights the potential benefit of considering thermal resistance as an alternative metric. Thermal resistance calculations, which do not require the measurement of sample thickness, have a lower measurement uncertainty, estimated at only  $\pm 0.49\%$ . However, shifting to thermal resistance would necessitate adapting the PHOEBE model to utilize this property instead.

An alternative solution to enhance the measurement uncertainty is to address the limited resolution of the thickness measurement. The resolution could be improved by transitioning from the 10-bit Arduino Mega to a microprocessor with a higher bitrate. A viable alternative would be a custom PCB with an integrated ESP32 board [51], boasting a 16-bit resolution, designed to fit the existing PCB shield. This upgrade would not only significantly improve the measurement resolutions but also enhance the performance of the PID controller, since it also boasts a higher bit rate for the PWM pins.

A notable limitation in the Therminus-K3 prototype is the linear slide potentiometer's range in the thickness measurement system. Initially designed to measure fabric samples up to 50 mm, it was found that the potentiometer's linear range is only about 25 mm. This limitation restricts the maximum measurable sample thickness to 25 mm. Future versions of the prototype should consider using digital readout systems or higher quality slide potentiometers with a wider linear range to overcome this constraint and enhance measurement capabilities.

The Therminus-K3 prototype's cooling capacity is limited by the small size of its heatsinks, which leads to inadequate heat dissipation. As a result, there is a minor temperature drift observed in the Cold Stage during measurements. While this drift has the potential to negatively influence measurement accuracy, preliminary testing has not shown any significant adverse effects. Implementing larger heatsinks or reducing the temperature differential across the sample could effectively mitigate these issues and enhance the prototype's performance and measurement reliability.

During a 25-minute measurement period, a notable decrease of approximately 3% in compression force was observed, as detailed in Annex 25. This reduction is likely due to stress relaxation in the 3D-printed PLA parts used in the prototype's construction. This observation indicates a potential need for a structural redesign where the load bearing components are replaced by materials with better mechanical properties. Such a change could enhance the prototype's durability and ensure more consistent performance.

Another limitation is the surface emissivity of the bare aluminium surface plates used in the Therminus-K3 prototype. According to GHP technical standards, the hemispherical emissivity of these plates should be greater than 0.8 to maximize heat transfer to the measurement sample. Initially, the use of high emissive paints was considered, but this approach was found to excessively influence the surface temperature readings, as detailed in Annex 26. Therefore, the next iteration of the prototype should explore anodizing the surface plates to increase the emissivity without affecting temperature measurement accuracy.

The most significant limitation in this study was the lack of a precise reference standard for verifying the accuracy of the Therminus-K3 prototype. The used thermal conductivity value provided by the silicone manufacturer lacks precision, and the accuracy of the cotton measurements, while within the expected range for plain knitted cotton fabrics (0.035 W/mk to 0.073 W/mk [52-54]), could hence not be truly verified the way it was desired. A comparative assessment with another GHP apparatus is necessary for a definitive evaluation of its accuracy.

Thermal conductivity measurements with the Therminus-K3 prototype were performed under the inherent pressure exerted by the weight of the

device's Hot Stage, leading to a standard pressure of approximately 710 Pa. This setup standardizes thickness measurements but could affect the thermal conductivity readings of highly compressible samples due to excessive compression. Such standard sample pressure is a common aspect in similar devices [48, 55] and complicates the direct comparison of thermal conductivity values obtained from different studies or devices. To mitigate this effect the Hot Stage can be adjusted to a specific height, which would allow for the preservation of the sample's natural state without any compression. While this adjustment might alter the thickness readings, it helps ensure that the sample's compression does not influence the thermal conductivity measurements. Consequently, further investigation is needed to understand the extent of this impact on thermal conductivity measurements and its effect on PMI estimation.

### Recommendations

Given these considerations, further testing of the Therminus-K3 prototype is recommended. Future tests should aim to confirm the prototype's accuracy by measuring reference standards with known thermal properties and by comparing it with other GHP machines. Additionally, tests examining the effects of replicating fabric compression as observed at crime scenes on PMI estimation should be conducted. It is furthermore essential to conduct tests with deceased human subjects to ascertain how effectively the prototype enhances the accuracy of PMI estimations made with the PHOEBE model in real forensic cases. This approach will provide a more realistic assessment of the prototype's applicability and effectiveness in actual forensic investigations.

Another area requiring attention is the prototype's PID controller, which regulates multiple heaters through interdependent PID variables. This control challenge is not fully addressed in this study so improving the PID controller could lead to more precise temperature regulation, reducing overshoots and thus accelerating the attainment of steady-state conditions. Moreover, replacing the existing power supply with a battery pack is advised to enhance the prototype's portability and on-site usability. Implementing these recommendations will not only help to further validate the prototype's effectiveness but also pinpoint areas for improvement, ensuring the Therminus-K3's suitability for practical forensic applications.

### 9. Conclusion

The Therminus-K3 prototype represents an important development in the field of forensic science, particularly in the context of thermal conductivity measurements for PMI estimation. This innovative prototype employs the GHP technique in a portable, yet cost-effective format, specifically designed for on-site thermal conductivity measurements of clothing garments. Combining precision, speed, and practicality, the Therminus-K3 holds the potential to be a valuable tool for forensic investigations. Preliminary

evaluations indicate its ability to perform thermal conductivity measurements with a precision of  $\pm 1\%$  and a measurement uncertainty of less than  $\pm 1.5\%$ . Additionally, these assessments indicate the promising potential to enhance PMI estimations of the PHOEBE by more than 30 minutes. It furthermore demonstrated the ability to reach steady state conditions within 6 minutes and complete measurements in 25 minutes, aligning well with the demands of time-sensitive forensic scenarios. Its non-destructive measurement capability safeguards the integrity of potential evidence, a crucial aspect in forensic investigations, ensuring that accurate data collection does not compromise evidence quality.

In summary, the Therminus-K3 prototype is a notable innovation in GHP and forensic technology, providing a practical and reliable aid for refining PMI estimations. Its ongoing development holds significant potential to further impact and improve the efficient and reliable determination of the time of death.

### Acknowledgements

I would like to express my sincere gratitude to dr. ir. A.J. Loeve for his invaluable help and supportive advice throughout this project. Additionally, I am particularly thankful to ir. J.A. Brenkman, whose expertise in electronic design and help with troubleshooting was essential to the success of the prototype.

### Bibliography

1. Henßge, C. and B. Madea, *Estimation of the time since death in the early post-mortem period*. Forensic Science International, 2004. **144**(2): p. 167-175.
2. Shrestha, R., T. Kanchan, and K. Krishan, *Methods Of Estimation Of Time Since Death*. 2019.
3. Cockle, D.L. and L.S. Bell, *Human decomposition and the reliability of a 'Universal' model for post mortem interval estimations*. Forensic Science International, 2015. **253**: p. 136.e1-136.e9.
4. Madea, B., *Methods for determining time of death*. Forensic Science, Medicine, and Pathology, 2016. **12**(4): p. 451-485.
5. Wilk, L., et al., *Reconstructing the time since death using noninvasive thermometry and numerical analysis*. Science Advances, 2020. **6**: p. eaba4243.
6. Henssge, C., *Death time estimation in case work. I. The rectal temperature time of death nomogram*. Forensic Science International, 1988. **38**(3): p. 209-236.
7. Henssge, C., *Rectal temperature time of death nomogram: Dependence of corrective factors on the body weight under stronger thermic insulation conditions*. Forensic Science International, 1992. **54**(1): p. 51-66.
8. Dessing, O., *Surface Testing Methods to Define Thermal Properties of Surfaces at Crime Scenes*. 2018, Delft University of Technology
9. Kaanen, C.I., *A Systematic Review on Factors Affecting the Thermal Conductivity Measurement of Clothing in a Forensic Context*. 2021.

10. Kaanen, C.I., *Terminus-K2: A Thermal Conductivity Measurement Device for Textile Layers at a Crime Scene*. 2022.
11. Haven, O.C.H.t., *Evaluating Thermal Measurement Techniques for Textile Analysis in Forensic Investigations*. 2023.
12. Ivo Sebastiaan Best, N.S.N., Sebastiaan Thomas Njio, Jason Morgan van Schijndel, Fabian Matthias Verhage, *Measuring thermal conductivity of textiles for time of death estimation*. 2019.
13. *Insulation Materials in Context of Sustainability*. 2016, Rijeka: IntechOpen.
14. *Thermal performance of building materials and products — Determination of thermal resistance by means of guarded hot plate and heat flow meter methods — Products of high and medium thermal resistance*, B.S. Insitution, Editor. 2001.
15. *Standard Test Method for Steady-State Heat Flux Measurements and Thermal Transmission Properties by Means of the Guarded-Hot-Plate Apparatus*, A. International, Editor. 2019.
16. Zhang, X., et al., *Personal thermal management by thermally conductive composites: A review*. Composites Communications, 2021. **23**: p. 100595.
17. Moran, M.J.t., *Principles of engineering thermodynamics*. 7th ed. 2012, Hoboken, N.J: Wiley.
18. Haven, O.C.H.t., *Terminus K3 Supplementary Material*. 2023.
19. *Standard Practice for Using a Guarded-Hot-Plate Apparatus or Thin-Heater Apparatus in the Single-Sided Mode*, A. International, Editor. 2016.
20. SolidWorks, C., *SolidWorks 3D CAD 2023*. Dassault Systèmes: Waltham, MA.
21. Brooks, D., *PCB Design Guide to Via and Trace Currents and Temperatures*. 2021: ARTECH HOUSE.
22. *JLC PCB*. [cited 2023; Available from: <https://jlcpcb.com>].
23. *NXFT15XV103FA2B*. 2018, Murata Manufacturing Co.: Japan. <https://www.murata.com/en-us/products/productdetail?partno=NXFT15XV103FA2B150>.
24. *Spaceloft: Flexible high performance insulation for building envelopes & equipment*. 2016, Aspen Aerogels. <https://www.aerogel.com/wp-content/uploads/2021/06/Spaceloft-Datasheet-English.pdf>.
25. Czichos, H., Saito, Tetsuya, Smith, L *Springer handbook of materials measurement methods*. 2008.
26. *TEC1-12706 Thermoelectric Cooler*. Hebei I.T. Co. Ltd. <https://peltiermodules.com/peltier.datasheet/TEC1-12706.pdf>.
27. Sencorcon, *SC134 Full Bridge Micro Load Cell*. 2018, Sensor and Control Co.,Ltd
28. *JB4-HX711 Junction Board With Load Cell Amplifier*. 2022, Sensorcon.
29. *Arduino Mega 2560 Rev3*. 2023, Arduino.CC. <https://store.arduino.cc/products/arduino-mega-2560-rev3>.
30. *LM4040-N/-Q1 Precision Micropower Shunt Voltage Reference*. 2016, Texas Instruments.
31. *INA226 Voltage and Current Module, Power Meter Module*. 2019, MJKDZ. <http://www.mjkdz.com>.
32. *INA226 High-Side or Low-Side Measurement, Bi-Directional Current and Power Monitor with I2C Compatible Interface*. 2015, Texas Instruments. [https://www.ti.com/lit/ds/symlink/ina226.pdf?ts=1695267473725&ref\\_url=https%253A%252F%252Fwww.google.com%252F](https://www.ti.com/lit/ds/symlink/ina226.pdf?ts=1695267473725&ref_url=https%253A%252F%252Fwww.google.com%252F).
33. *ILI9341-240x320 SPI TFT-LCD Screen*. MLMLH: Shenzhen.
34. J.S.T. Mfg. Co, L., *XH Connector*: 2024.
35. *SN74LVC245A Octal Bus Transceiver*. 2015, Texas Instruments. <https://www.ti.com/lit/ds/symlink/sn74lvc245a.pdf>.
36. MathWorks, *MATLAB*. 2023. <https://www.mathworks.com/products/matlab.html>.
37. *Greisinger G1700 Thermometer*. GHM GROUP - Greisinger: Germany. [https://www.greisinger.de/files/upload/en/produkte/bda/G1700\\_EN.pdf](https://www.greisinger.de/files/upload/en/produkte/bda/G1700_EN.pdf).
38. Chen, C., *Evaluation of resistance-temperature calibration equations for NTC thermistors*. Measurement, 2009. **42**(7): p. 1103-1111.
39. *Gauge Blocks: 630.880*, in *Qualite: Extra 1991*, Carry Le Locle Suisse: Switzerland.
40. *Tolerances in  $\mu\text{m}$  of CARY gauge-blocks*. Cary Le Locle Swiss: Switzerland.
41. *Fluke 114, 115, 116 and 117 Digital Multimeters*. 2006, Fluke Corporation.
42. *School Balance EMB5.2K5*. Kern & Sohn GmbH. <https://www.kern-sohn.com/shop/en/products/laboratory-balances/EMB-5-2K5/>.
43. *FLIR E75 Advanced Thermal Camera*. 2017, TELEDYNE FLIR.
44. *Sweet & Salt 100% Silicone Baking Sheet for Rolled Cake*. Casa International NV: Belgium. <https://ch.casashops.com/de/produkte/sweet-salt-fuer-gerollten-kuchen-594867/594867/>.
45. *100% Cotton Blue Work Jacket*. KLM Kleding: Netherlands.
46. *General Purpose Silicone Sheet: Food Grade Quality Silicone, 60 Shore A*. 2015, Jehbco Manufacturing Pty Ltd. <https://jehbco.com.au/catalogues/SiliconeSheet.pdf>.
47. *High Temperature Silicone Sheeting Datasheet*, in *Product Code: RC0323-HT*. 2004, The Rubber Company Limited. <https://therubbercompany.com/wp-content/uploads/2018/05/RC0323-HT-High-Temperature-Silicone-Sheeting-MDS.pdf>.
48. Reddy, K.S. and S. Jayachandran, *Investigations on design and construction of a square guarded hot plate (SGHP) apparatus for thermal conductivity measurement of insulation materials*. International Journal of Thermal Sciences, 2017. **120**: p. 136-147.
49. Hogan, R., *How to Calculate Resolution Uncertainty: The Complete Step-by-Step Guide*. 2016.
50. ToolBox, T.E., *Solids, Liquids and Gases - Thermal Conductivities*, in *Thermal conductivity coefficients for insulation materials, aluminum, asphalt, brass,*



*copper, steel, gases and more.* 2003.  
[https://www.engineeringtoolbox.com/thermal-conductivity-d\\_429.html](https://www.engineeringtoolbox.com/thermal-conductivity-d_429.html).

51. Espressif, S., *ESP32 Series Datasheet*. 2021.
52. Oglakcioglu, N. and A. Marmarali, *Thermal comfort properties of some knitted structures*. *Fibres & Textiles in Eastern Europe*, 2007. **15**: p. 94-96.
53. Hoque, M., et al., *Effect of Yarn Count, Weave Structure, Construction on Thermal Properties and Air Permeability of Fabrics*. *European Journal of Engineering and Technology Research*, 2022. **7**: p. 87-91.
54. Majumdar, A., S. Mukhopadhyay, and R. Yadav, *Thermal properties of knitted fabrics made from cotton and regenerated bamboo cellulosic fibres*. *International Journal of Thermal Sciences*, 2010. **49**(10): p. 2042-2048.
55. Johra, H., *Description of the Guarded Hot Plate Method for Thermal Conductivity Measurement with the EP500*. 2019, Department of Civil Engineering, Aalborg University: Aalborg.



# Therminus K3

Supplementary Material



## Table of Contents:

<i>Annex 1: Therminus-K3 Design Requirements</i> .....	3
<i>Annex 2: Overview of Surface Pressure Measurements</i> .....	4
<i>Annex 3: PCB Heater Designs</i> .....	5
<i>Annex 4 A: Thermistor Dept – Temperature Gradient Calculations</i> .....	6
<i>Annex 4 B: Thermistor Dept – Thermal Uniformity Simulation</i> .....	8
<i>Annex 5: Side Guard Heater Simulations</i> .....	9
<i>Annex 6: Cold Stage Fan Configuration Test</i> .....	11
<i>Annex 7.A: Concept Generation Compression Stage</i> .....	13
<i>Annex 7.B: Morphological Chart Compression Stage</i> .....	14
<i>Annex 8: Concept Generation Quick Release Mechanism</i> .....	15
<i>Annex 9A: Compression Stage - Hot Stage Cover Plate Simulations</i> .....	16
<i>Annex 9B: Compression Stage - Linear Carriage Simulations</i> .....	18
<i>Annex 10: INA226 Power Module Settings</i> .....	22
<i>Annex 11: Arduino Mega PCB Shield Circuit Diagram</i> .....	24
<i>Annex 12: Therminus-K3 Prototype Photos</i> .....	30
<i>Annex 13: Therminus-K3 Prototype Build of Materials</i> .....	37
<i>Annex 14A: Thermistor Calibration Setup</i> .....	39
<i>Annex 14B: Thermistor Calibration Results</i> .....	41
<i>Annex 14C: Thermistor Validation Results</i> .....	45
<i>Annex 15A: Thickness Sensor Calibration</i> .....	47
<i>Annex 15B: Thickness Sensor Validation</i> .....	48
<i>Annex 16A: INA226 Current Sensor Calibration Setup</i> .....	49
<i>Annex 16B: INA226 Current Sensor Calibration Results</i> .....	50
<i>Annex 16C: INA226 Current Sensor Validation Results</i> .....	51
<i>Annex 17: Load Cell Calibration and Validation</i> .....	52
<i>Annex 18: Thermal Uniformity Verification Surface Plates</i> .....	53
<i>Annex 19: Thermistor Dept Verification</i> .....	55
<i>Annex 20: Therminus-K3 Measurement Procedure</i> .....	57
<i>Annex 21: Performance Validation Measurements</i> .....	58
<i>Annex 22: Therminus-K3 Measurement Uncertainty:</i> .....	60
<i>Annex 23: PHOEBE Model Simulations</i> .....	62
<i>Annex 24: Therminus-K3 Design Criteria Evaluation</i> .....	66

<i>Annex 25: Compression Force Measurement Drift</i> .....	67
<i>Annex 26: Paint Layer – Temperature Gradient Calculations</i> .....	67
<i>Annex 27: Therminus-K3 Postprocessing Results</i> .....	69
<i>Annex 28: Therminus-K3 Postprocessing MATLAB Script</i> .....	71
<i>Annex 29: Therminus-K3 Arduino Code</i> .....	73
<i>Annex 30: Therminus-K3 Technical Drawings</i> .....	86
<i>Bibliography</i> .....	96

## Annex 1: Therminus-K3 Design Requirements.

Design criteria for a thermal conductivity measurement apparatus were established through discussions with key stakeholders, including the Dutch Police, Netherlands Forensic Institute, Amsterdam Medical Centre, and TU Delft. The resulting design requirements are categorized into two groups: Performance requirements and General Requirements, Table 1, and Table 2, respectively. Additionally, requirements from technical standards, primarily focusing on the hot plate assembly, are summarized in Table 3[1-3].

Performance Requirements:		
Requirement:	Target:	Reason:
Accuracy error	< 3 %	Accuracy greatly affects PMI estimation and results should be admissible in court [4, 5]
Precision error	< 1 %	Precision greatly affects PMI estimation and results should be admissible in court [4, 5]
Measurement Range	0.020 - 0.700 [W/mk]	K-values above 0.700 W/mk hardly affect the accuracy of the PMI estimation [4]
Variable Pressure	0 – 10000 [Pa]	Target range covers average pressures encountered in supine and sitting positions *
Sample thickness range	0.44 – 28 [mm]	Target range covers 99.73% of all clothing [6]
Sample dimensions	Non-destructive	Clothing garments should ideally remain undamaged to preserve them as evidence [7]
Setup time	< 5 minutes	To minimize investigative interference
Measurement time	< 30 minutes	To minimize investigative interference
Lifetime recalibration	without ≥ 12 months	Low maintenance requirement to ensure reliable and uninterrupted long-term operation

**Table 1. Performance Requirements**

\* Pressure range determined from literature findings as presented in Annex 2

General Requirements:		
Requirement:	Target:	Reason:
Shielding	Measurement should be shielded from wind, water, and temperature fluctuations	Environmental factors should not influence the measurement
Decontamination	Resistant to alcohol-based cleaning up to 5 times a day	To prevent cross-contamination between crime scenes
Portability	Volume: 330 x 330 x 330 [mm] [4] Weight: < 8 [kg] [4]	To facilitate device transportation to a crime scene
Durability	The enclosed device must pass rough handling and repetitive shock tests: NEN-EN-IEC 60068-2-27:2008 [8]	Considering the device's likely exposure to demanding conditions during hectic crime scene investigations
Simplicity	The device should have a user-friendly and intuitive interface	To ensure efficient operation by investigators with varying levels of technical expertise
Power Source	The device should operate using either battery power or a car battery connection	To offer versatile power options for ease of use in different settings
Reporting Protocol	The device should provide both raw measurement data files and PDF reports summarizing measurements and outcomes	To ensure comprehensive data collection and facilitate clear documentation of measurements and results for investigative purposes
Cost	≤ € 1.945	To maintain budget constraints and ensure affordability

**Table 2. General Requirements**

Technical Standard Requirements:	
Requirement:	Explanation:
Flatness of Surface Plates	The surface plates should exhibit a maximum deviation from flatness that does not exceed 0.025% of their linear dimensions
Material for Surface Plates	Materials are selected that are dimensionally and chemically stable and suitably strong to withstand warpage and distortion when a clamping force is applied. Materials shall also have a high thermal conductivity to increase thermal homogeneity and response.
Surface Emissivity of Surface Plates	The surfaces that are in contact with the sample shall be treated to maintain a total hemispherical emittance greater than 0.8.
Gap Dimensions	The area of the gap separating the main heater from the guard heaters shall not be more than 5% of the metered section
Heat Flow Minimization	Any connections between the metered section and the primary guard shall be designed to minimize heat flow across the gap
Heater Temperature Difference	Average temperature difference between the measurement plate and the guard plates shall not exceed 0.2K
Heater Uniformity	The heaters in the metered section and guard sections shall be designed to create a uniform temperature over the surface plates
Temperature Sensors Measurement Area	The measurement area shall at least contain two temperature sensors
Temperature Imbalance Sensors Gap	Temperature imbalance sensors shall be placed at a strategic distance of one-fourth the side length from each corner for optimal detection of temperature variations across the gap
Temperature Control Mechanism	Main, guard, and back heater shall have independent temperature control which may be achieved by adjusting the voltage or current, or both
Power Measurement	The power to the metered section is determined with a wattmeter or from voltage and current measurement across the heater in the metered section

**Table 3. Summary of Key Requirements from Technical Standards**

BS 12667 - 01: Determination of Thermal Resistance by Means of Guarded-Hot-Plate and Heat Flow meter methods [1]  
 ASTM C177 – 19: Standard Test Method for Steady-State Heat Flux Measurements and Thermal Transmission Properties by Means of the Guarded-Hot-Plate Apparatus [2]  
 ASTM C1044 – 16: Standard Practice for Using a Guarded-Hot-Plate Apparatus or Thin-Heater Apparatus in Single- Sided Mode [3]

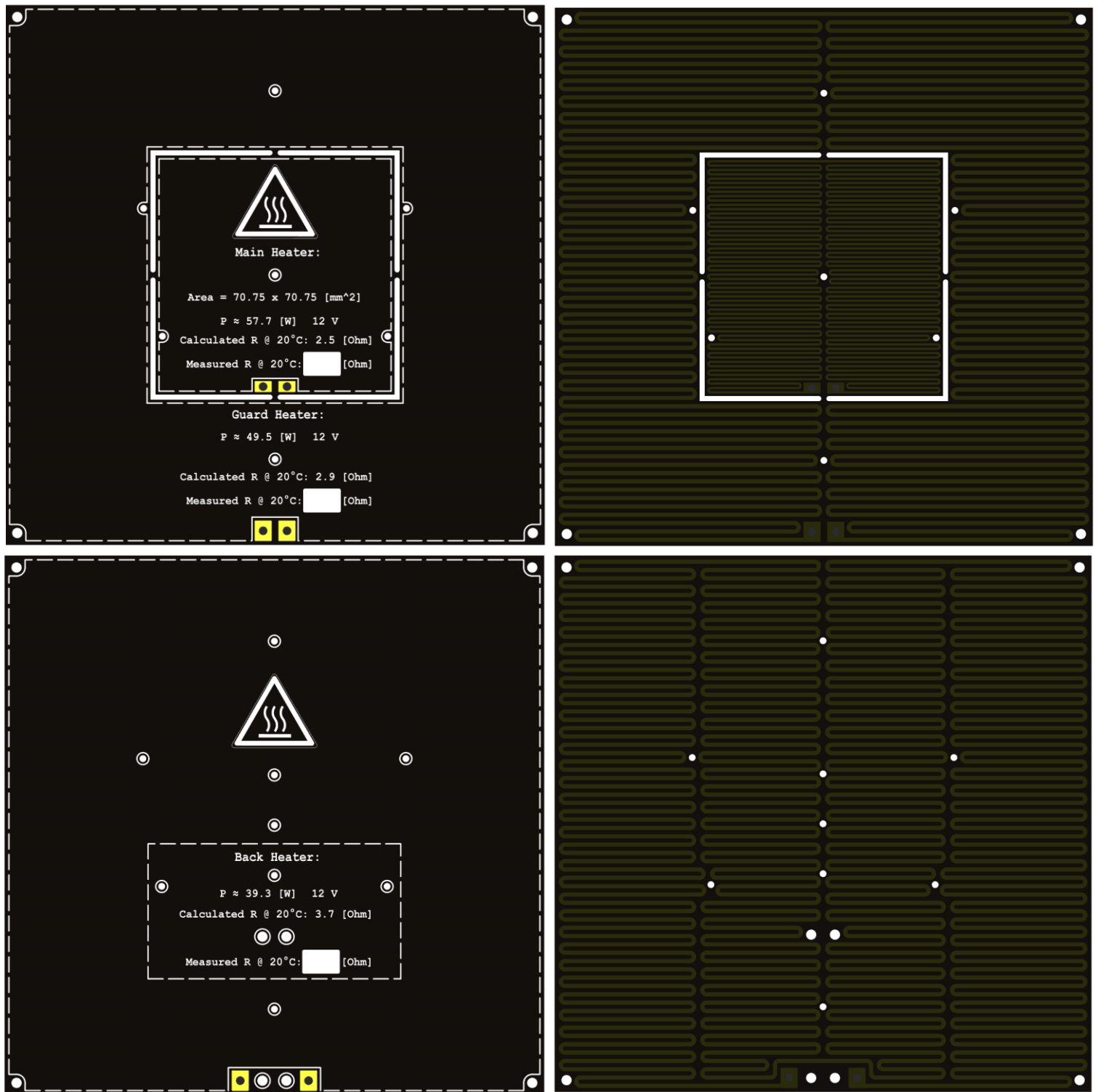
## Annex 2: Overview of Surface Pressure Measurements

Pressure Measurements Human Body					
Posture	Measurement Region	Surface material	Average Pressure [Pa]	Maximum Pressure [Pa]	Source
Sitting	Buttocks	Rigid Plane Surface	≈ 6343	≈ 8480	Swearingen et al. 1962. [9]
Supine	Entire body	Mattress	≈ 2400	--	Rajala et al. 2006. [10]
		Floor	≈ 4900	≈ 6000	
Supine under different inclinations	Sacrum	Mattress	--	≈ 7032 - 8373	Bush et al. 2015. [11]
	Shoulder		--	≈ 3817 - 4906	
	Buttocks		--	≈ 7305 - 8537	
Supine	Sacrum	Mattress	≈ 6510	≈ 13018	Everton et al. 2014. [12]
		Floor	≈ 9067	≈ 28531	
Sitting	Buttocks	Office Chair	≈ 7190	≈ 23080	Yuan et al. 2020 [13]
		Bamboo Chair	≈ 7770	≈ 26260	
		Wooden Chair	≈ 8330	≈ 28910	
	Back	Office Chair	≈ 3430	≈ 10500	
		Bamboo Chair	≈ 2420	≈ 6660	
		Wooden Chair	≈ 2530	≈ 6940	

**Table 4. Summary of Human Body Surface Pressure Measurement from Literature**

The values in this table have been converted to Pascals (Pa) for standardization across different measurement units. Additionally, these values are rounded to enhance readability and ease of comparison.

## Annex 3: PCB Heater Designs



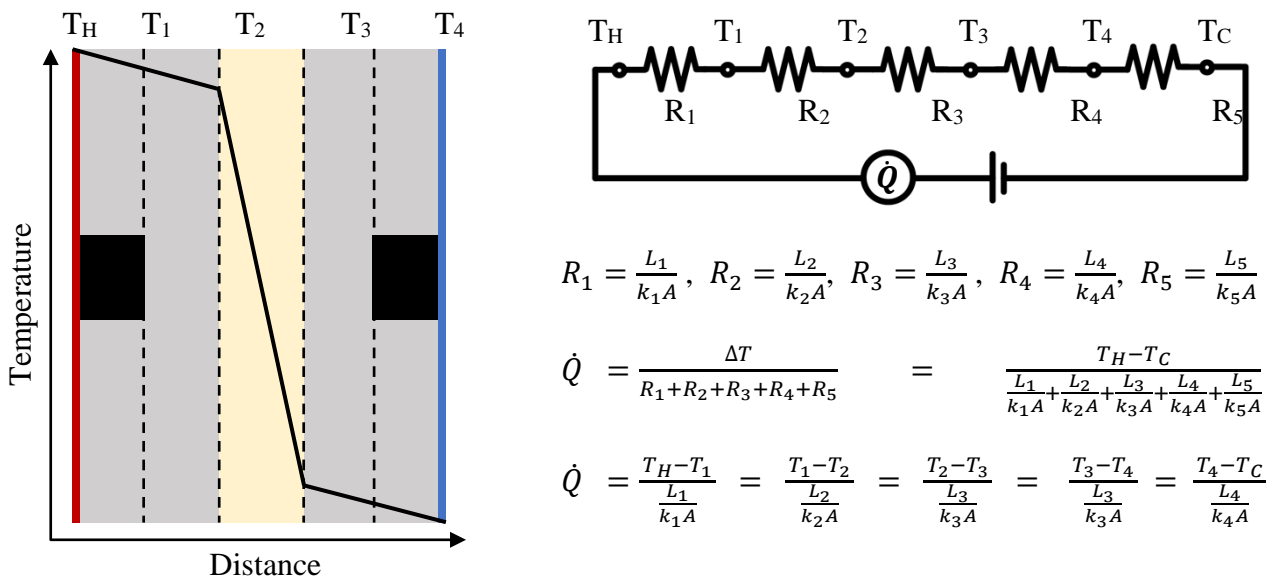
**Figure 1: PCB-Heater Designs with Visible Copper Traces**

**Top:** Main heater PCB featuring integrated guard heater, displayed in both front and rear orientations. **Bottom:** PCB heater design back heater, front and back. Each PCB measures 158 mm by 158 mm with a thickness of 1.6 mm. Mounting holes are located at the corners, with other holes specifying thermistor locations. Power is supplied via 12V connections, indicated by yellow pads. The PCBs are constructed from FR-4 material, consist of 2 layers, and have an outer copper layer with a thickness of 35 $\mu$ m.



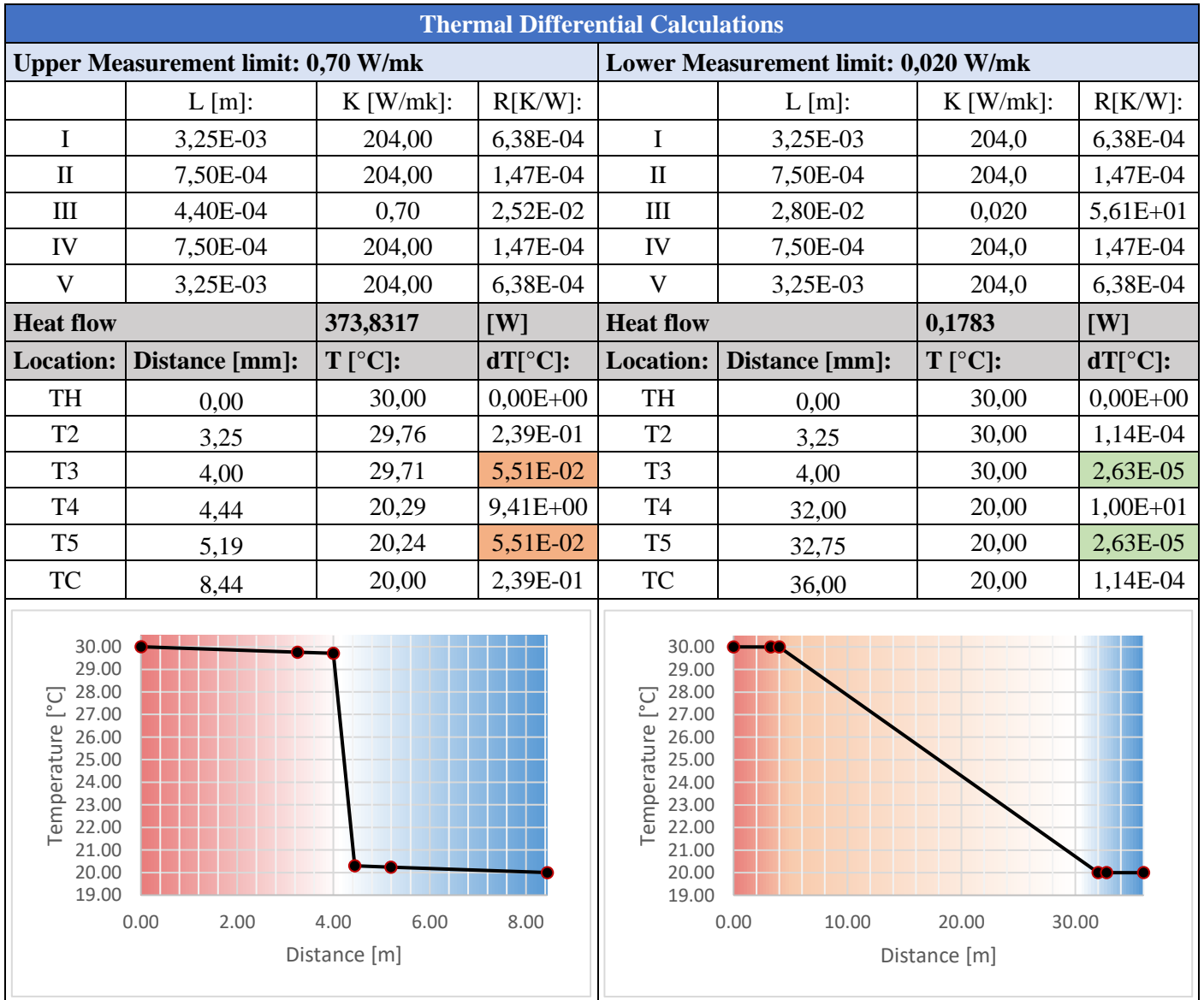
## Annex 4 A: Thermistor Dept – Temperature Gradient Calculations

To check at what dept the imbedded thermistors in the surface plates could still accurately measure the surface temperature a thermal circuit was set up using electrical analogies, where temperatures are represented as voltages and thermal resistances  $R_i$  are analogous to electrical resistances. The thermistor depth was iteratively adjusted to find a depth at which the temperature differential was acceptably small. In this analogy, the flow of heat ( $\dot{Q}$ ) is analogous to electrical current, and it is driven by the temperature difference ( $\Delta T$ ) similar to a voltage difference driving a current. The thermal resistance of each section is given by the formula  $R = \frac{L}{kA}$ , where  $L$  is the length of the section,  $k$  is the thermal conductivity of the material, and  $A$  is the cross-sectional area through which heat is conducted. The total heat flow is then calculated by the temperature difference divided by the sum of the thermal resistances, like Ohm's Law ( $V=I \cdot R$ ) for electrical circuits. With the total known heat flow the temperature difference across each section can be determined. The temperature difference between  $T_1$  and  $T_2$  and similarly between  $T_2$  and  $T_4$ , which represent the temperature differential between the thermistor and the surface temperature, should be smaller than the measurement resolution of the embedded thermistors to facilitate accurate surface temperature readings of the thermistors. Figure 2 details the electrical circuit analogy and table 5 shows the actual temperature differences between each section calculated for each of the thermal conductivity measurement limits. In these calculations  $\Delta T$  is set to 10 K and  $A$  is set to 0,024964 m<sup>2</sup>, representing the temperature differential between the plates and the surface area of the plates of the Therminus-K3 prototype respectively.



**Figure 2: Thermal Circuit using Electrical Analogy**

**Left:** Temperature distribution in a surface plate-sample-surface plate assembly with embedded thermistors (black rectangles), **Right:** Electrical circuit analogy used to calculate total heat flow and temperature gradients over each segment.



**Table 5. Temperature Differential Calculations for Thermistors at a dept of 0.75 mm**

**Left:** Temperature differential calculations for upper thermal conductivity measurement limit, thickness assumed to be minimum at 0.44 mm.

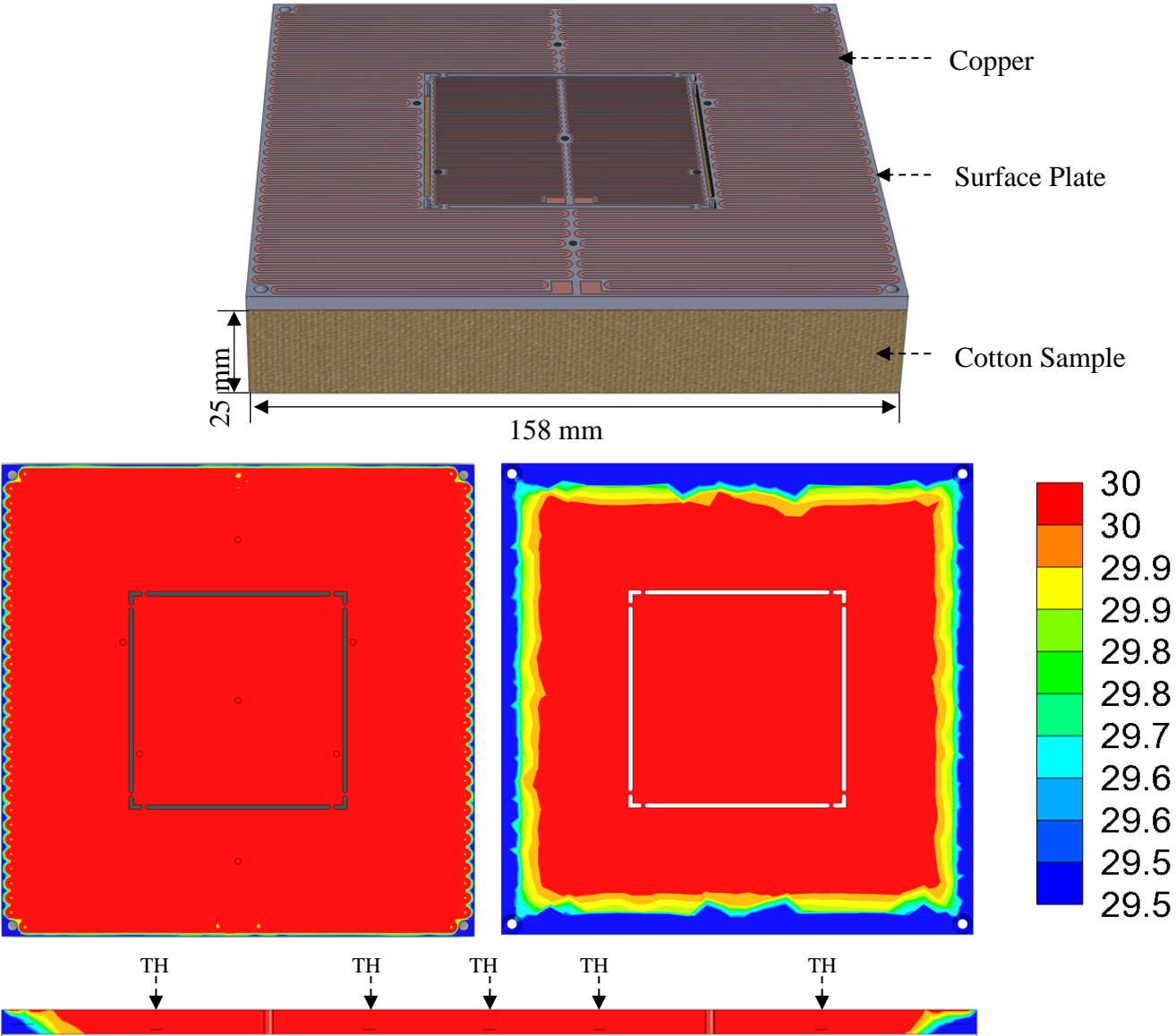
**Right:** Temperature differential calculations for lower thermal conductivity limit, sample thickness assumed to be maximum at 28 mm

Graphs display temperature gradient across total assembly length.

The temperature difference between the surface and the depth of 0.75 mm, where the thermistors are embedded, is maintained below 0.051 °C. This value is within the measurement resolution of the thermistors as specified in Annex 14, confirming that a depth of 0.75 mm is suitable for accurate surface temperature readings.

### Annex 4 B: Thermistor Dept – Thermal Uniformity Simulation

SolidWorks [14] simulations were conducted, as depicted in Figure 3, to verify if thermistors placed 0.75 mm below the surface affects the temperature distribution on the surface that is in contact with the sample. The findings revealed no thermal disturbances in the metering area, indicating that a depth of 0.75 mm for the thermistors is effective and does not interfere with the surface temperature measurements.

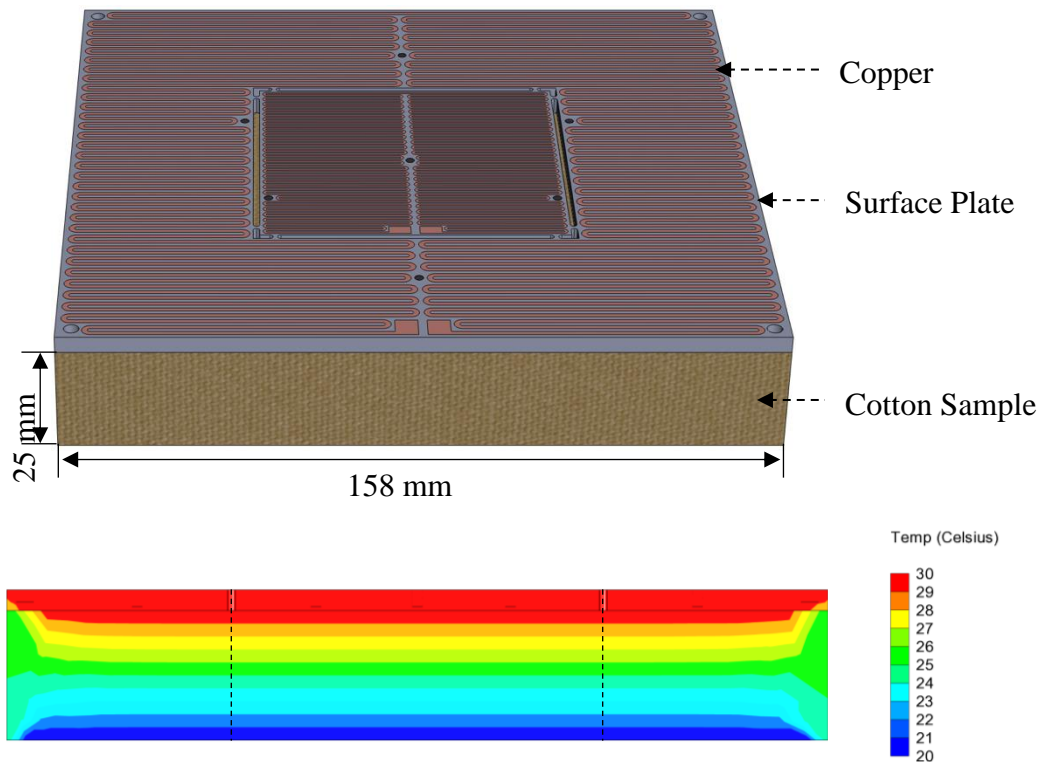


**Figure 3: Surface Plater Thermal Uniformity Simulation**

**Top:** Simulation model featuring the 4 mm thick aluminum surface plate (thermal conductivity: 130 W/mK) housing the 7 thermistors mounted at a dept of 0.75 mm (thermal conductivity: 1.8 W/mK as specified in [15]), and a 25 mm thick cotton sample (thermal conductivity: 0.04 W/mK). **Middle:** Thermal simulation result with applied temperature of 30 °C to the copper traces, 25 °C to the edges of the cotton sample and 20°C to the bottom of the cotton sample. **Bottom:** Section cut surface plate where **TH** indicate thermistor locations in surface plate. The color scale is fine-tuned to highlight temperature variations between 29.5 °C and 30 °C, emphasizing any deviations in thermal uniformity.

## Annex 5: Side Guard Heater Simulations

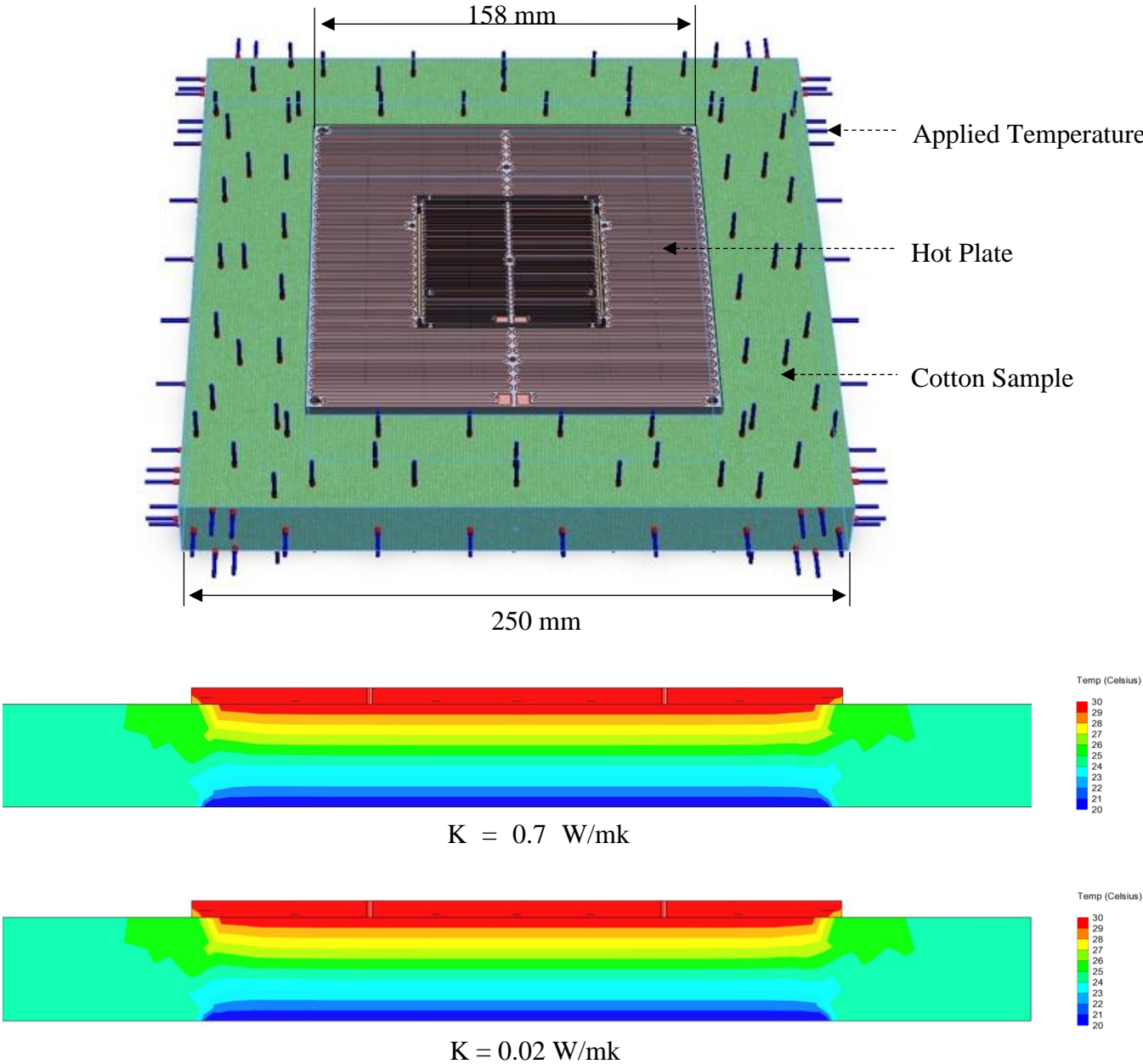
Simulations were performed to determine the need for side guard heaters. Both non-destructive tests, as referenced in the main text, and destructive scenarios, where the sample matches the surface plate dimensions, were simulated. The destructive simulation, as shown in Figure 4, also demonstrate a uniform heat transfer across the metering area, evidenced by consistent horizontal temperature profiles, indicating effective steady-state conditions.



**Figure 4: Side Guard Heater Necessity Simulations with Dimensionally Matched Sample**

**Top:** Simulation model featuring the 4 mm thick aluminum surface plate (thermal conductivity: 130 W/mK) housing the 7 thermistors mounted at a dept of 0.75 mm (thermal conductivity: 1.8 W/mK as specified in [15]), and a 25 mm thick cotton sample (thermal conductivity: 0.04 W/mK). **Bottom:** Section cut of thermal simulation result with applied temperature of 30 °C to the copper traces, 25 °C to the edges of the cotton sample and 20°C to the bottom of the cotton sample. Dotted lines indicate measuring area under main heater.

The impact of the sample's thermal conductivity was also evaluated. The non-destructive simulations of the main text were repeated for a 25 mm thick sample, using 0.7 W/mk to represent the upper limit and 0.02 W/mk for the lower limit of thermal conductivity. Figure 5 illustrates that no significant differences emerge between the two, both establish a steady-state heat flow beneath the metering area. Thus, the results indicate that side heaters are not necessary.



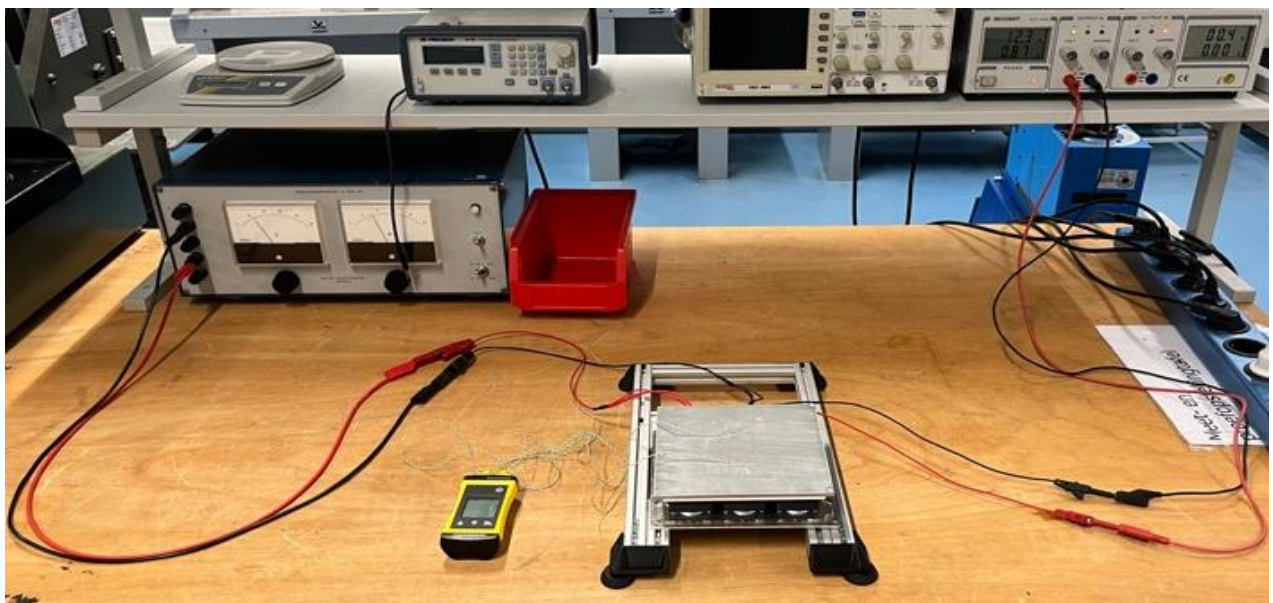
**Figure 5: Side Guard Heater Necessity Simulations**

**Top:** Simulation model featuring the 4 mm thick aluminium surface plate (thermal conductivity: 130 W/mK) and a 250 x250 x 25 mm cotton sample with applied temperature of 30 °C to the copper traces, 25 °C to the open edges of the cotton sample and 20°C to the bottom of the cotton sample. **Middle:** simulation result of a sample with a thermal conductivity of 0.7 W/mK. **Bottom:** simulation result of a sample with a thermal conductivity of 0.02 W/mK.



## Annex 6: Cold Stage Fan Configuration Test

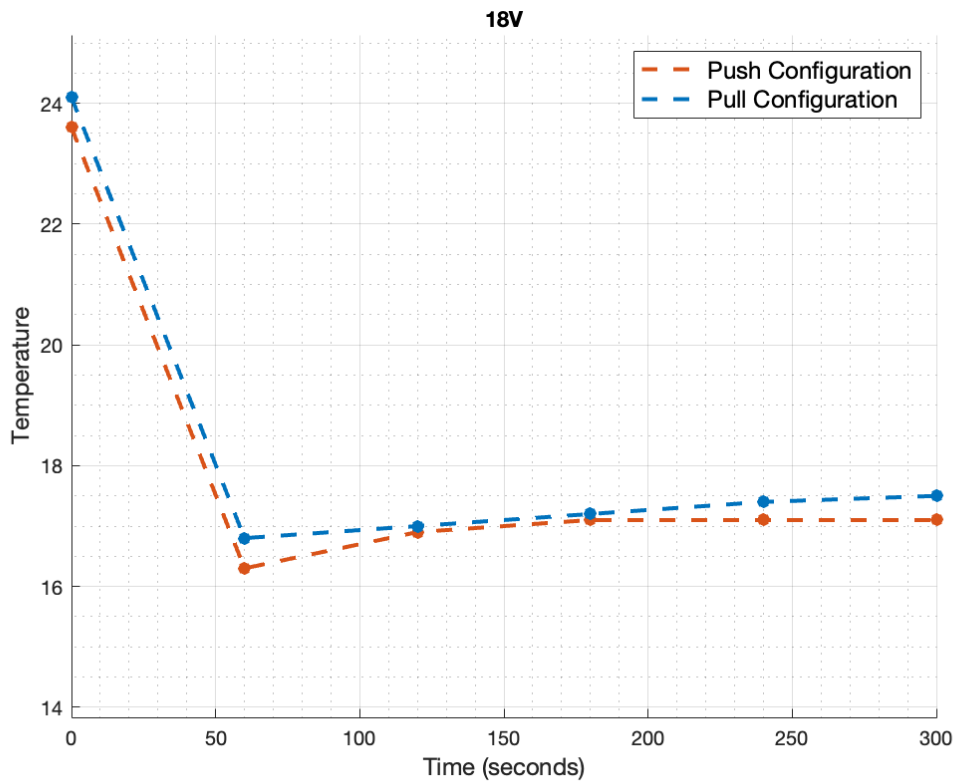
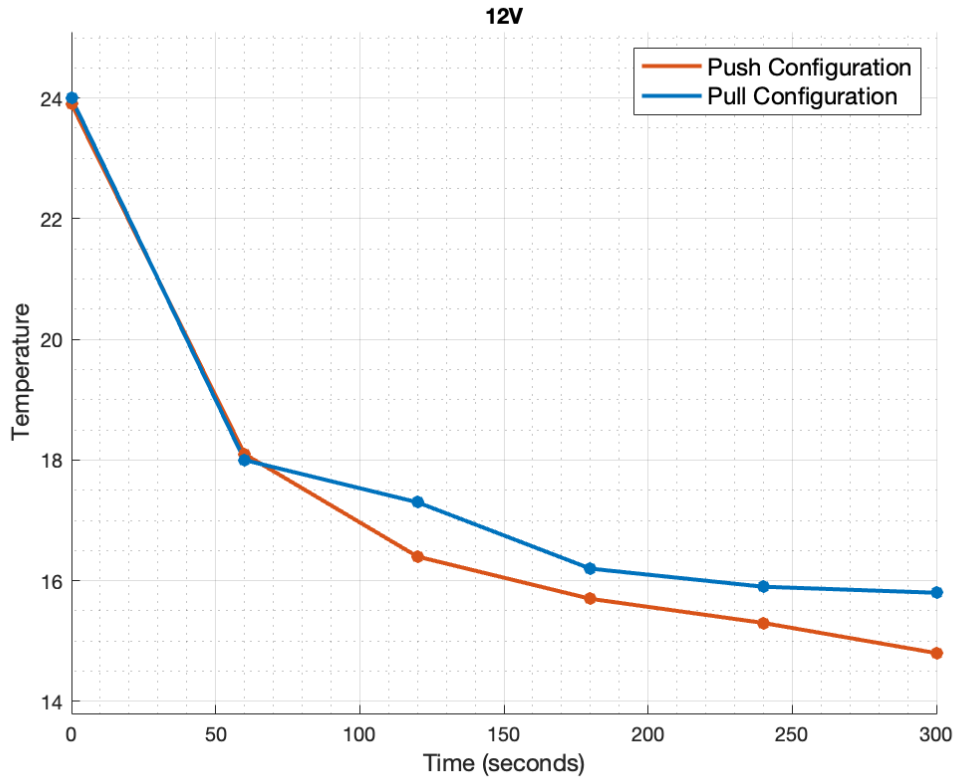
To assess the most effective orientation for the cooling fans on the Peltier stacks, experiments were conducted comparing push and pull configurations. In the push configuration, fans push air upwards through the heatsinks from below the Cold Stage, while the pull configuration draws air downward from the sides of the Cold Stage. In the experiments the 3x3 array of Peltier modules were connected to a variable power supply, while the array of cooling fans, set in one of the configurations, was connected to another variable power supply, as shown in Figure 6. The temperature of the Cold Stage's surface plate was logged using two thermistors connected to the plate by tape. The thermistors were linked to a Greisinger G1700 thermometer [16] and the temperature was recorder every 60 seconds over 5 minutes. The process was repeated for both fan configurations and at two different Peltier voltages, 12V and 18V.



**Figure 6: Cold Stage Fan Configuration Test Setup**

Showing, from left to right, the variable power supply powering the 3x3 array of Peltier modules, the Greisinger thermometer, the Cold Stage, the variable power supply powering the 3x3 array of cooling fans.

The findings are shown in Figure 7. For both voltage settings, the push configuration outperformed in cooling the Cold Stage, leading to its selection for the final concept. Peltier modules should theoretically cool more effectively at an operational voltage of 18V [17]. However, it's noteworthy that in the tests, the 18V setup provided less cooling compared to the 12V setup. This suggests that heat dissipation from the Peltier modules was inadequate at the higher voltage.



**Figure 7: Cold Stage Fan Configuration Test Results**  
**Top: 12V Configuration, bottom: 18V Configuration.**

## Annex 7.A: Concept Generation Compression Stage

Compression Stage Concepts		
Gas Strut Concept	Dead Weight Concept	Constant Force Spring Concept
Rack and Pinion Concept	Linear Spindle Concept	Geared Spindle Concept
Quick Release Geared Spindle Concept	Motorized Spindle Concept	

Table 5. Compression Stage Concept Generation

## Annex 7.B: Morphological Chart Compression Stage

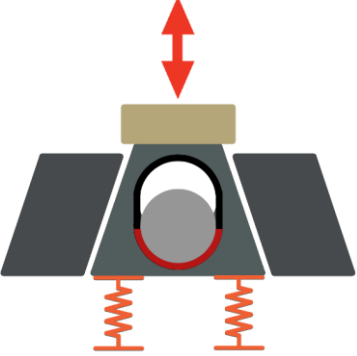
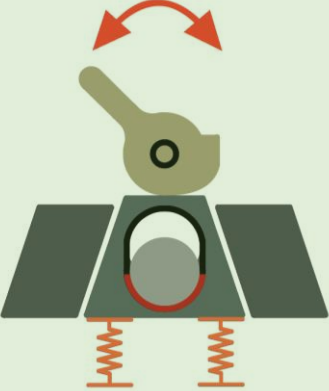
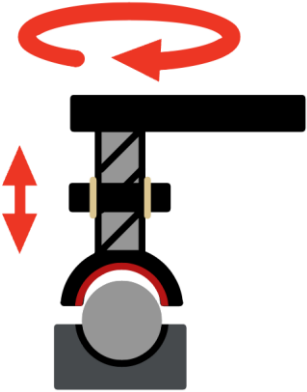
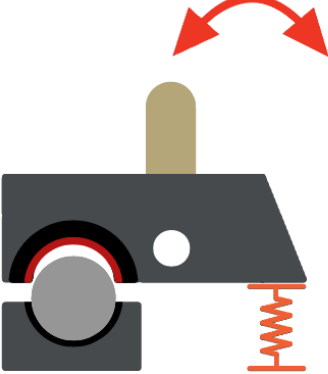
Morphological Chart Compression Stage						
Sub-functions	Sub-solutions:					
Linear motion	Linear guide with linear ball bearings	Linear guide with sleeve bearings	Linear Slide with linear ball bearings	Linear Slide with sleeve bearings		
Pressure mechanism	Dead weight	Constant force spring	Gas strut	Manual or motorized vice/screw press	Hydraulic press	Rack and pinion
Drive system	Manual Spindle	Motorized Spindle	Lever actuated	Gear driven Spindle		
Sensors	Strain gage load cell	Capacitive load cell	Spring-Loaded Mechanical gauge	Hydraulic Pressure Sensors		

	= Optimal solution		= Suboptimal solution
	= Viable solution		= Infeasible solution

*Table 6. Compression Stage Concept Generation*

The chosen concept, named the 'Quick Release Geared Spindle Concept', was selected from eight distinct concepts presented in Table 5. This table was formulated based on a variety of sub-solutions from the morphological chart as detailed in Table 6. The key feature of this concept is its capacity to provide variable sample compression, which sets it apart from the majority of other solutions that are limited to applying a single, pre-determined level of pressure. The ability to adjust the compression level is particularly advantageous for mimicking the varying pressure conditions that might be found at crime scenes. This adaptability is crucial in ensuring the accuracy and reliability of measurements in forensic contexts. Moreover, the concept's quick-release mechanism adds a practical advantage, allowing for the rapid and easy loading and unloading of samples, enhancing the overall efficiency of the apparatus.

**Annex 8: Concept Generation Quick Release Mechanism**

Quick Release Mechanism Concept Generation	
Push Button Mechanism	Bi-Stable Switch Mechanism
	
Twist Lock Mechanism	Spring Lever Mechanism
	

*Table 7. Quick Release Mechanism Concept Generation*

The Bi-Stable Switch Mechanism was chosen as the optimal solution for the quick release mechanism. It was chosen from four different concepts shown in Table 7. Its standout feature lies in its dual-state functionality, allowing it to seamlessly switch between engagement and disengagement with the lead screw. This inherent stability in both positions eliminates the need for constant operation, making it user-friendly and ideal for the efficient loading and unloading of samples.

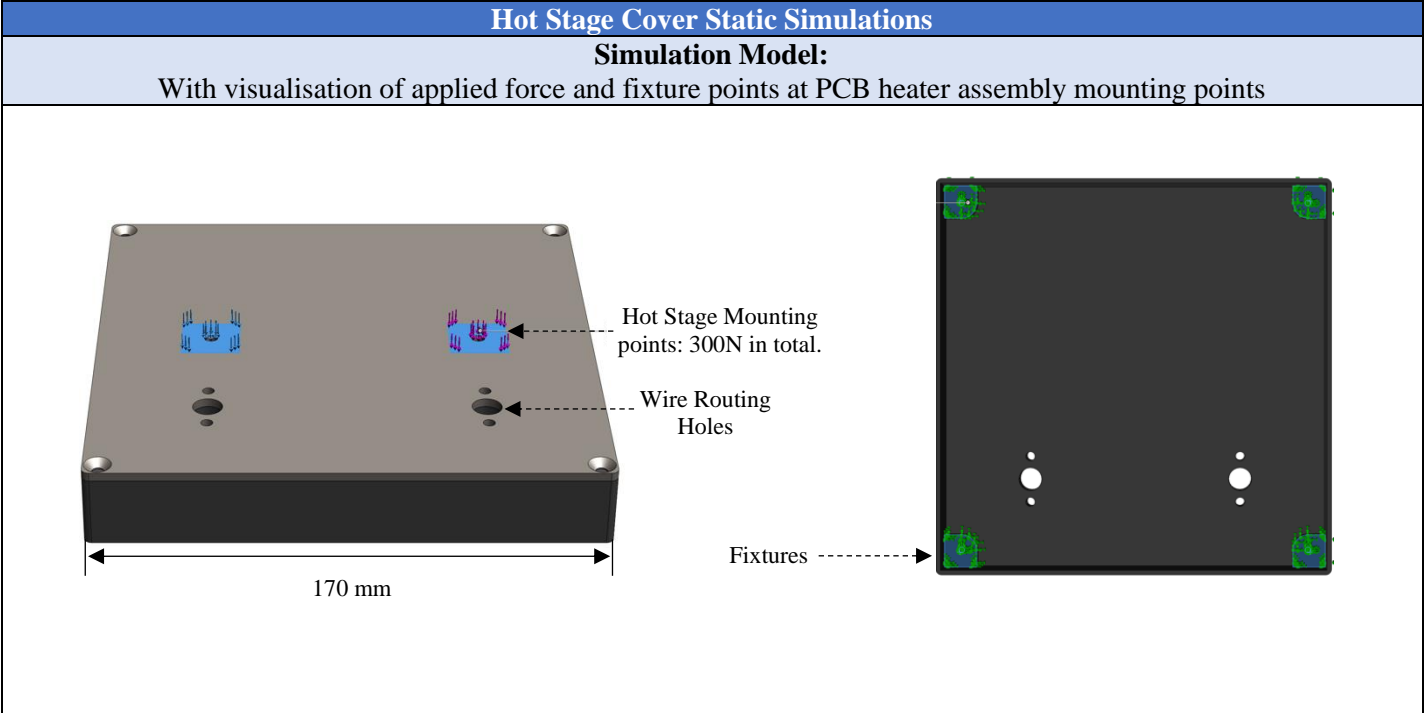


## Annex 9A: Compression Stage - Hot Stage Cover Plate Simulations

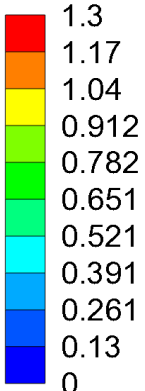
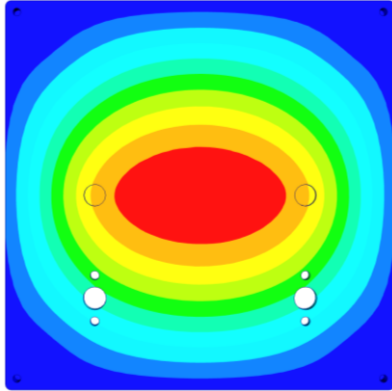
Since the cover of the Hot Stage would be 3D printed from PLA, a static force simulation was conducted using SolidWorks to assess the need for a reinforcement plate to minimize excessive deformation under load. The 3D-printed housing was modelled in SolidWorks with and without a 3 mm carbon steel cover plate. Additionally, a simulation was performed to assess the impact of a cut-out in the PLA cover, which would function as a wire routing zone. A total force of 300 N was applied to the mounting points, resembling a pressure of approximately 12,000 Pa across the surface plate with dimensions of 158 x 158 mm. 12,000 Pa exceeds the maximum required sample pressure as determined in Annex 2. The material properties as presented in table 8 were used for the simulations, the PLA properties were derived from the Ultimaker PLA datasheet [18]. The simulation results, shown in Table 9, highlight the necessity of a steel reinforcement plate to prevent excessive deformation. Additionally, it was found that the inclusion of a wire routing zone does not significantly affect the overall deformation.

Solidworks Material Properties:	
Ultimaker PLA[18]:	
Elastic modulus:	3000 MPa
Poison's ratio:	0.35 [19]
Mass Density	1240 kg/m <sup>3</sup>
Yield Strength:	59 Mpa
Plain Carbon Steel [14]:	
Elastic modulus:	205 GPa
Poison's ratio:	0.29
Mass Density	7858 kg/m <sup>3</sup>
Yield Strength:	283 MPa

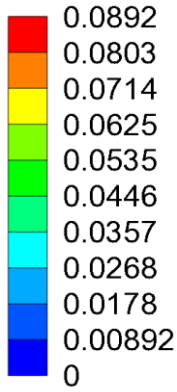
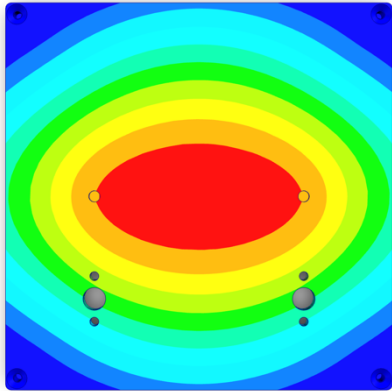
Table 8. Solidworks Simulation Material Properties



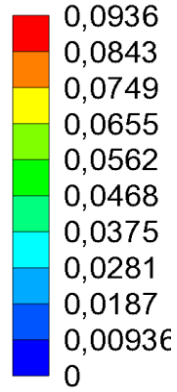
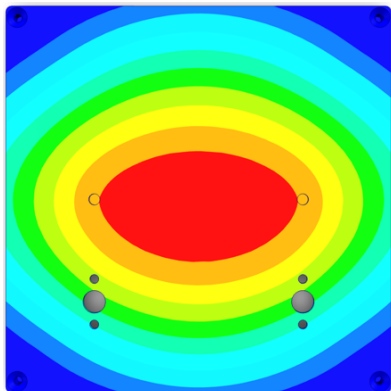
**PLA Hot Stage Cover Deformation [mm]:**  
Simulation result show excessive deformation



**PLA Hot Stage Cover + 3 mm Steel Reinforcement Plate Deformation [mm]:**  
Steel reinforcement plate eliminates the excessive deformation



**PLA Hot Stage Cover with Wire Routing Zone + 3mm Steel Reinforcement Plate Deformation [mm]:**  
Wire routing zone for thermistor and power wires has hardly any influence on the deformation



*Table 9. Hot Stage Cover Static Simulations*

## Annex 9B: Compression Stage - Linear Carriage Simulations

To ensure the Compression Stage's design was strong enough for the intended loads, static force simulations were conducted using Solidworks. A free body diagram was created to determine the load directions, as shown in Figure 8. Simulations were then performed on both the steel cover plate of the linear carriage, the 3D-printed linear carriage itself and the linear guide rail assembly. For these simulations, material properties of carbon steel and PLA were used as specified in Table 8. The results for the steel cover plate, presented in Table 10, indicate it is sufficiently strong, exhibiting minimal deformation and stress. The 3D-printed linear carriage went through an iterative design process to guarantee its strength. The final design as presented in Table 11 proved to be sufficiently strong, displaying acceptable deformations and stresses under load, thereby validating the iterative design approach. Lastly, the linear guide rail assembly, as shown in Table 12, exhibited minimal deformation under load, confirming the overall strength of the design.

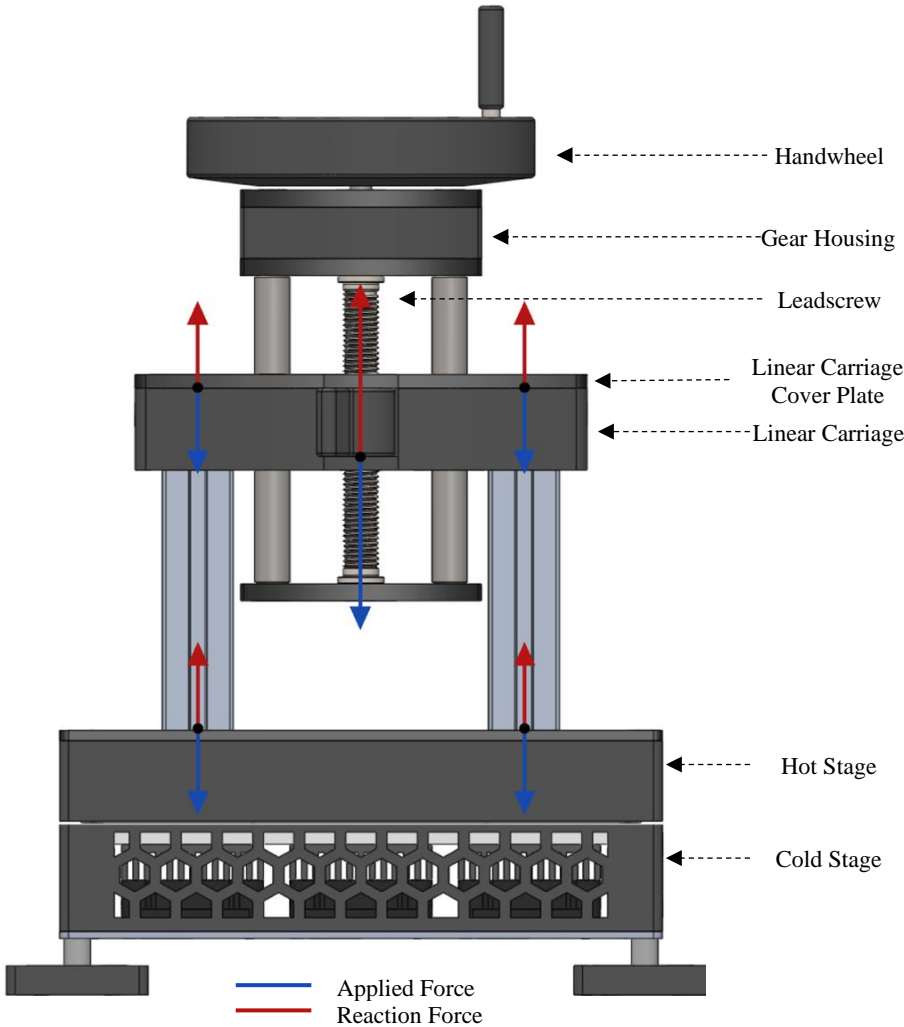
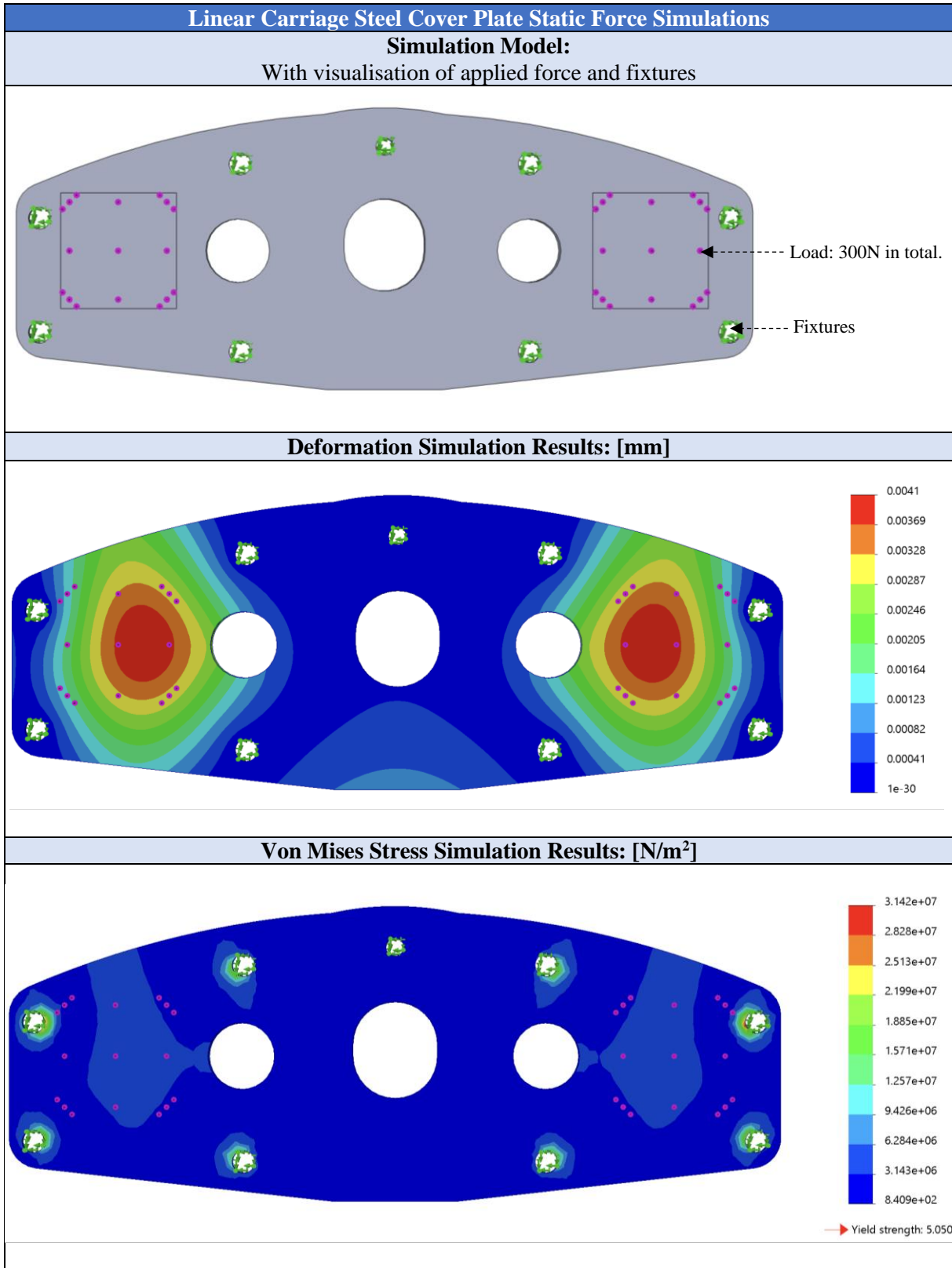
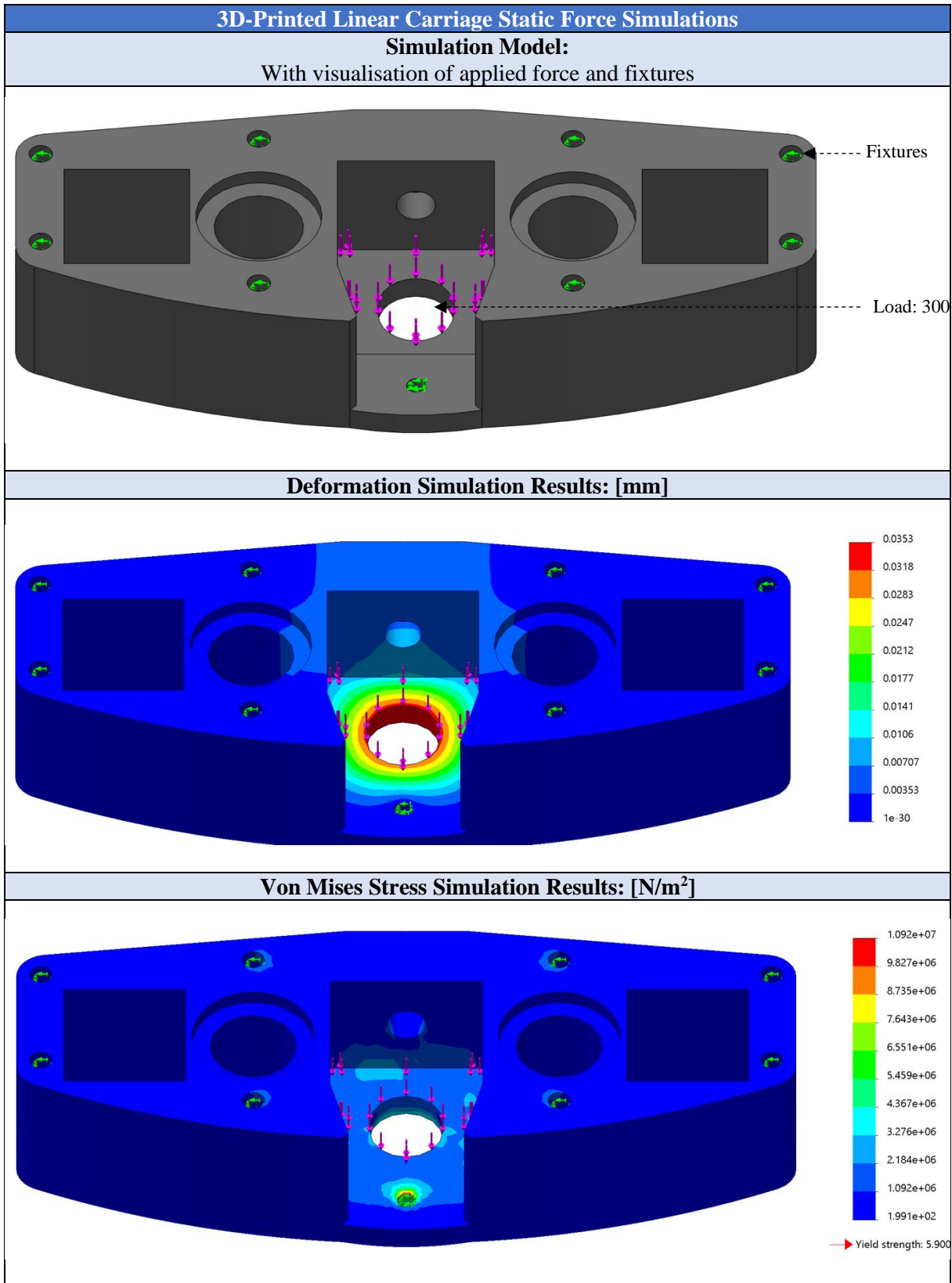


Figure 8: Free Body Diagram Compression Stage



*Table 10. Compression Stage - Linear Carriage Reinforcement Plate Static Simulations Results*



*Table 11. Compression Stage - Linear Carriage Static Simulations Results*



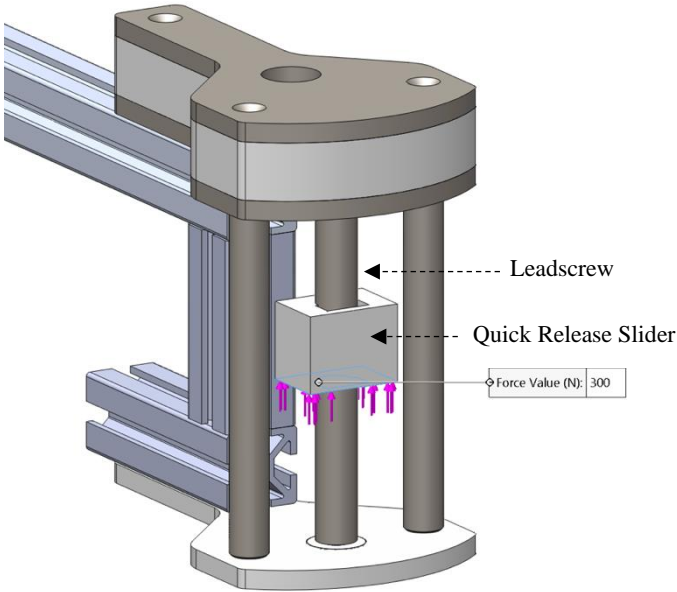
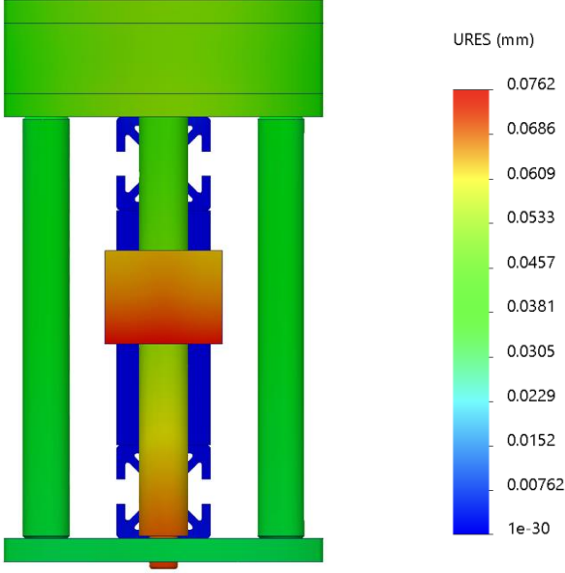
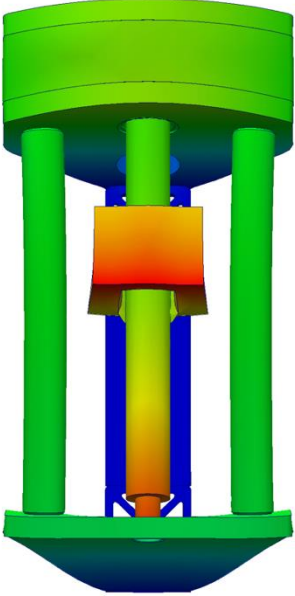
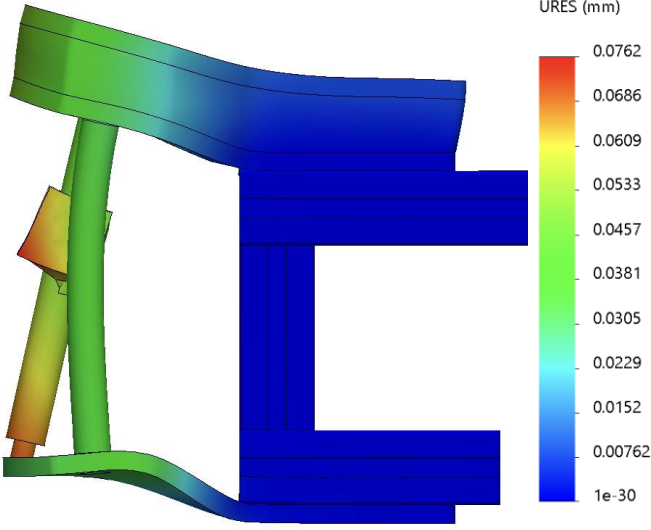
Compression Stage Linear Guide Rail Assembly Static Force Simulations	
<p><b>Simplified Simulation Model:</b> With visualisation of the applied force on the quick-release slider that engages with the leadscrew</p>	<p><b>Deformation Simulation Results:</b> Aluminium extrusions were fixed in the simulations (Front view)</p>
	
<p><b>Exaggerated deformation simulation results:</b> (Front view)</p>	<p><b>Exaggerated deformation simulation results:</b> (Side view)</p>
	

Table 12. Compression Stage - Linear Guide Rail Assembly Simulation Results

## Annex 10: INA226 Power Module Settings

The power delivered to the main heater is a block signal due to the PWM control, therefore the INA226's sampling rate and internal averaging function were set to accurately measure the mean power of this block signal. The module's analog-to-digital conversion time was set to 204  $\mu\text{s}$ , resulting in a sample rate  $f_s$  of 4901Hz. This rate is ten times the frequency of the power's block signal, which coincides with the standard 490Hz frequency of the Arduino's PWM control signal. This significantly higher sampling rate is important for accurately capturing the block signal's details and measuring the average current accurately. A simulated power signal, consisting of a block voltage signal and block current signal, with a duty cycle of 30% is shown in Figure 9, with the according sampling rate of 4901 Hz. Figure 10 shows the delivered power where the horizontal line indicates the average delivered power. Figure 11 shows how this average delivered power can simply be calculated by multiplying the current's block signal with the maximum voltage. This shows how the power is calculated in the Therminus-K3 prototype. Moreover, to get a more stable current reading the internal averaging function of the INA226 was configured to average 256 samples. This ensures that the module can accurately calculate the mean current of the block signal created by the PWM control, while also maintaining an adequate response time to changes in the signal.

$$\bullet f_s = \frac{1}{\text{Conversion Time}(s)} = \frac{1}{204 \cdot 10^{-6}} = 4901(\text{Hz}) \gg 490(\text{Hz})$$

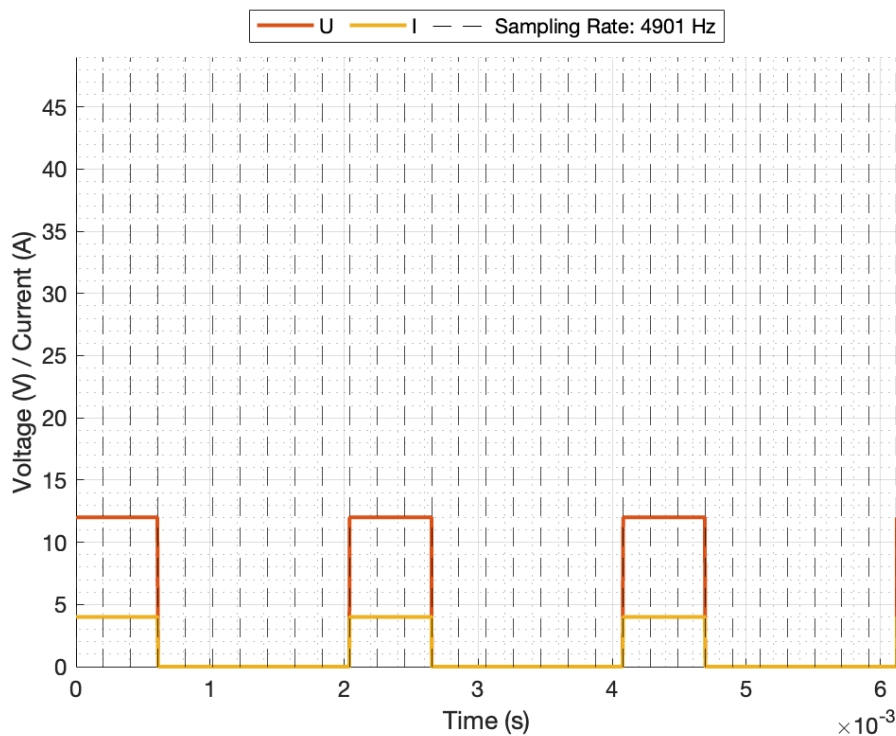
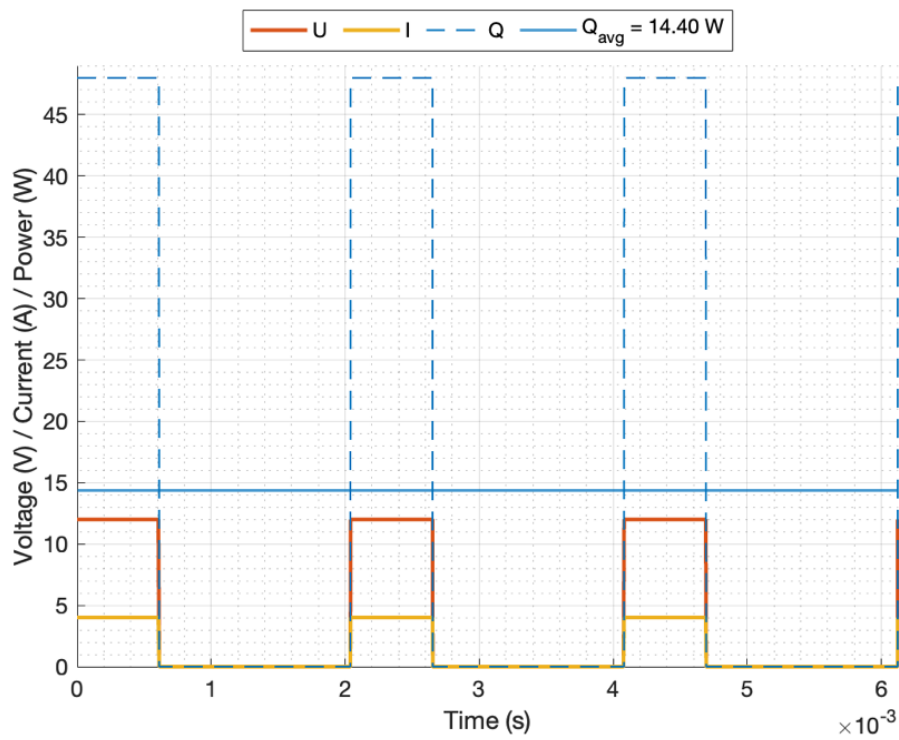
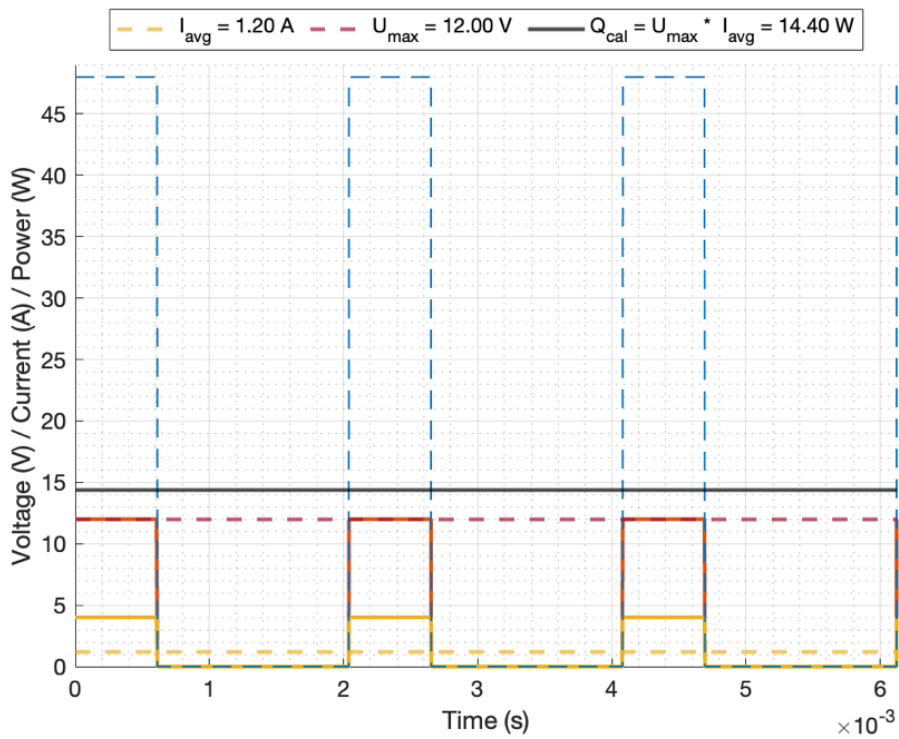


Figure 9: Simulated Power Signal with INA226's Sample Rate

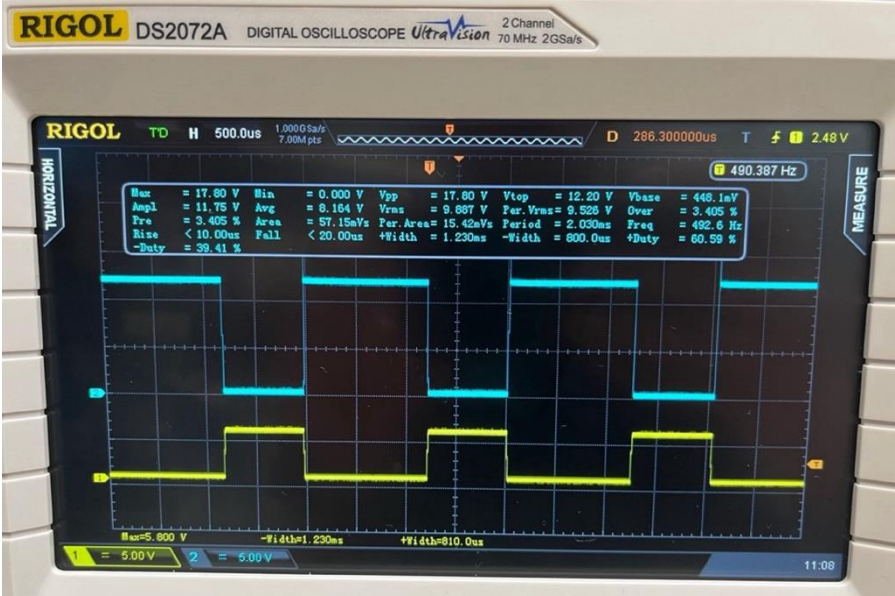


**Figure 10: Simulated Power Signal**  
 Where  $Q_{avg}$  is the average of the striped power block signal  $Q$ .



**Figure 11: Simulated Power Signal Calculated**  
 Where  $Q_{cal}$  is the average delivered power calculated as:  $Q_{cal} = U_{max} * I_{avg}$ .

The voltage block signals were verified using a RIGOL DS2072A Digital Oscilloscope [20]. Figure 12 displays both the 5V signal controlling the MOSFET and the 12V signal powering the heater. Both signals present clear block waveforms, confirming the correct functioning of the MOSFET and the heater but also affirming the suitability of the power calculation method as mentioned before.



*Figure 12: Oscilloscope Measurement of Voltage Signals  
 Yellow: 5V signal controlling the MOSFET.  
 Blue: 12V signal to the main heater.*

**Annex 11: Arduino Mega PCB Shield Circuit Diagram**

The custom Arduino Mega PCB shield is depicted in Figures 13 and 14. In Figure 13, illustrated the PCB designed to function as an Arduino Mega shield with the arrangement of components and JST-style connectors. This configuration allows for a direct connection to the Arduino Mega without the need for extra wires or a breadboard. Additionally, Figure 13 furthermore displays the trace routing, created following JLC PCB manufacturing guidelines [21] using KiCad design software[22].

Figure 14 provides both front and back views of the PCB, featuring power planes and ground planes. The front side includes a 12V power plane that supplies power to the heaters, Peltier modules, and fans. Power planes are employed to minimize heat build-up under full load, as the combined loads can draw a current of approximately 18A, as shown in Table 13. Therefore, a 12V, 360W, 30A power supply will suffice. The remaining components are powered through the 5V power plane or the 5V reference voltage. On the backside, there is a large ground plane directly connected to the power supply's ground. It's important to note that the Arduino Mega is powered via its own barrel connector but without a ground connection. This setup prevents the system from being grounded through the Arduino and the PCB Shield, which could potentially create unwanted ground loops.

Maximum Current Draw		
Load:	Maximum Power Draw	Maximum Current Draw at 12V
Main Heater	≈ 58 W	≈ 4.8 A
Guard Heater	≈ 50 W	≈ 4.2 A
Back Heater	≈ 40 W	≈ 3.3 A
Peltier Modules	≈ 54 W	≈ 4.5 A
Fans	≈ 18 W	≈ 1.2 A
<b>Total Maximum Current Draw:</b>		<b>≈ 18 A</b>

Table 13. Maximum Theoretical Current Draw

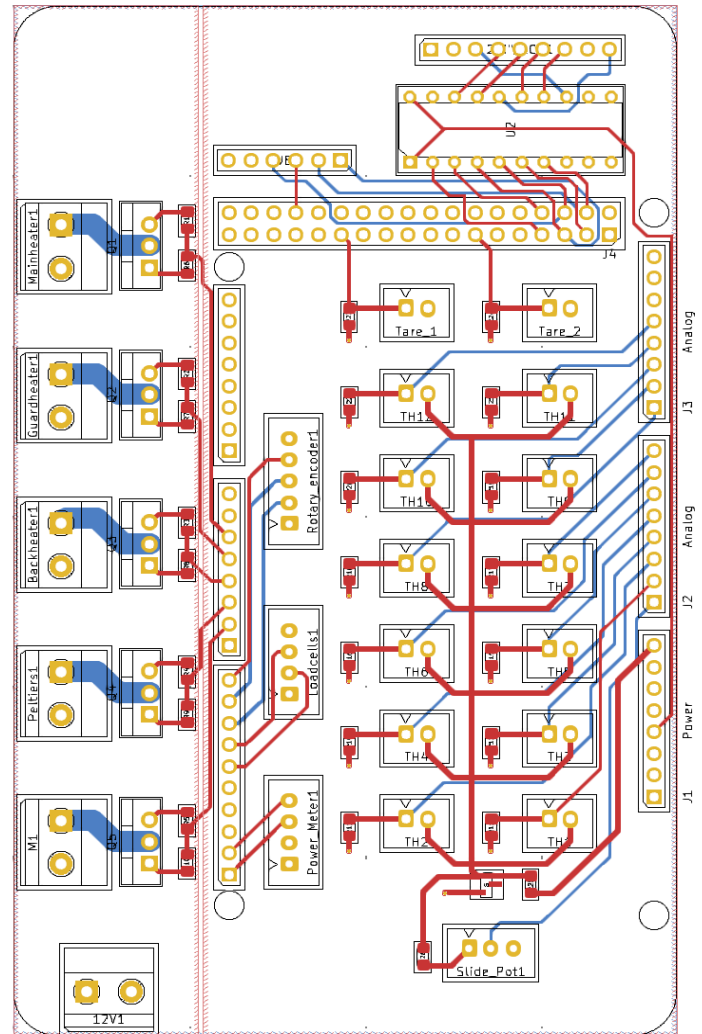
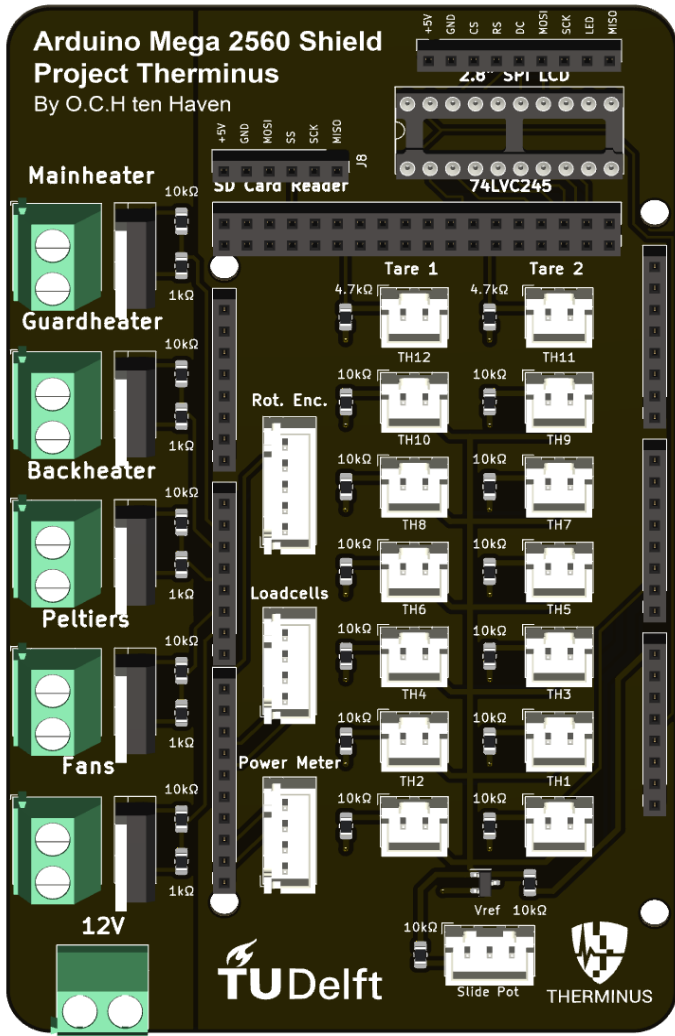
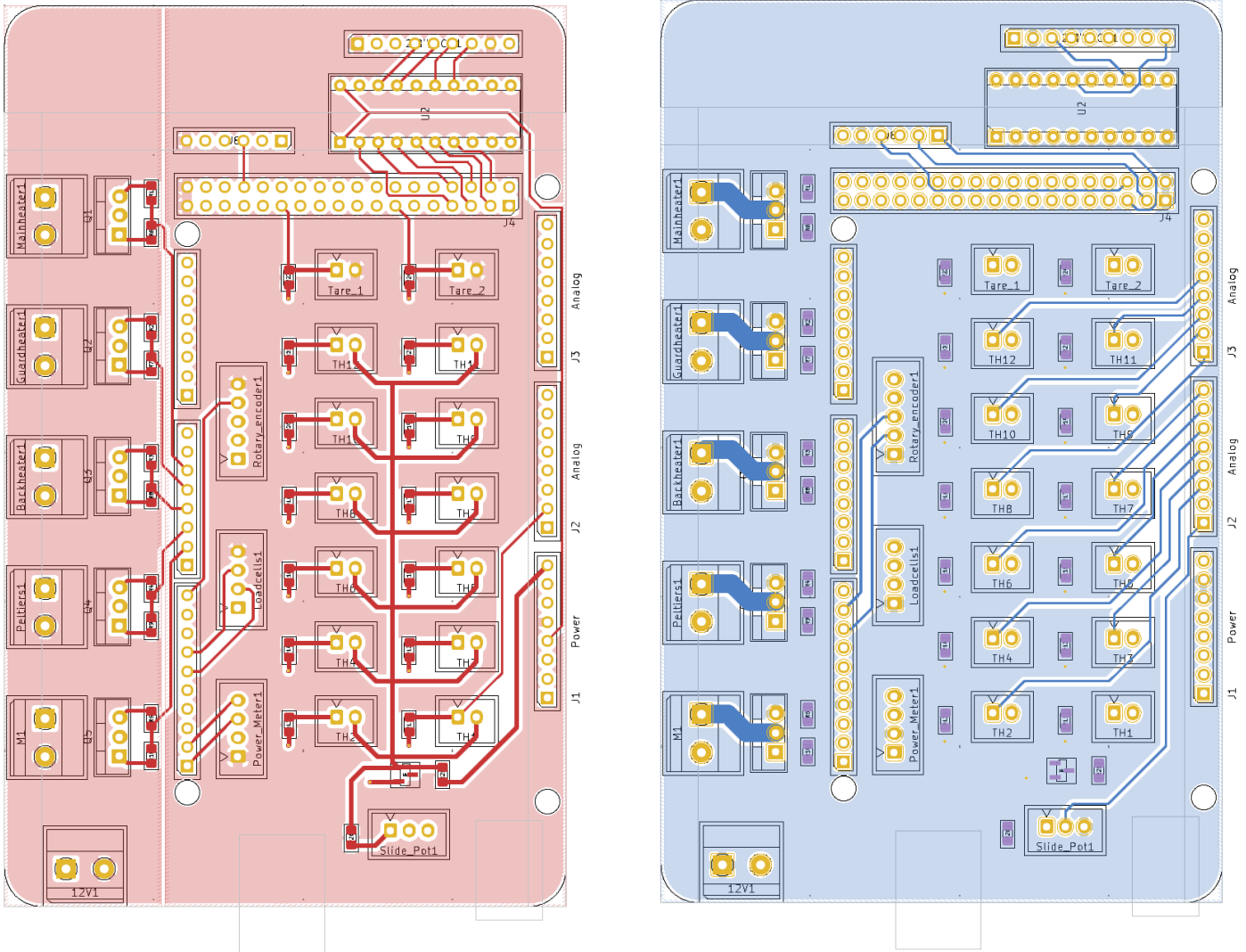


Figure 13: Custom Arduino Mega PCB Shield Layout





**Figure 14: Dual-Sided PCB Shield Layout**

**Left:** Front Side, featuring a 12V power plane on the left and a 5V power plane on the right of the PCB.

**Right:** Back side of the PCB with the ground plane. The ground traces from the screw terminals to the MOSFETs are widened to reduce heating, especially important since they may carry up to 4.5A in maximum load conditions.

The corresponding circuit diagrams of the custom PCB shield are shown in Figures 15, 16, and 17. The outputs of the Arduino Mega and all the subsystems are labelled so that you can easily identify which Arduino pin connects to which component. The 12V External Power Supply screw terminal as depicted in Figure 16 functions as the input power connection, to which all the +12V and GND connections of the PID Controllers schematic of Figure 15 are connected. The rest of the power and ground connections come straight from the +5V Arduino and GND pins or the +5VA reference voltage connection. In the PID Controllers diagram of Figure 15, the heaters, Peltier modules, and fans are connected via screw terminals rather than the symbols used to represent them. This allows for the lead wires of these components to be quickly and securely connected or disconnected, enabling ease of assembly and maintenance, the screw terminals are visible Figure 13 and 14. The Tare Button Circuits, shown in Figure 16 use a 4.7k $\Omega$  pull-down resistor. These resistors are connected to the ground, ensuring that when the button is not pressed, the input pin is pulled to 0V, preventing floating inputs which can result in unpredictable behaviour.

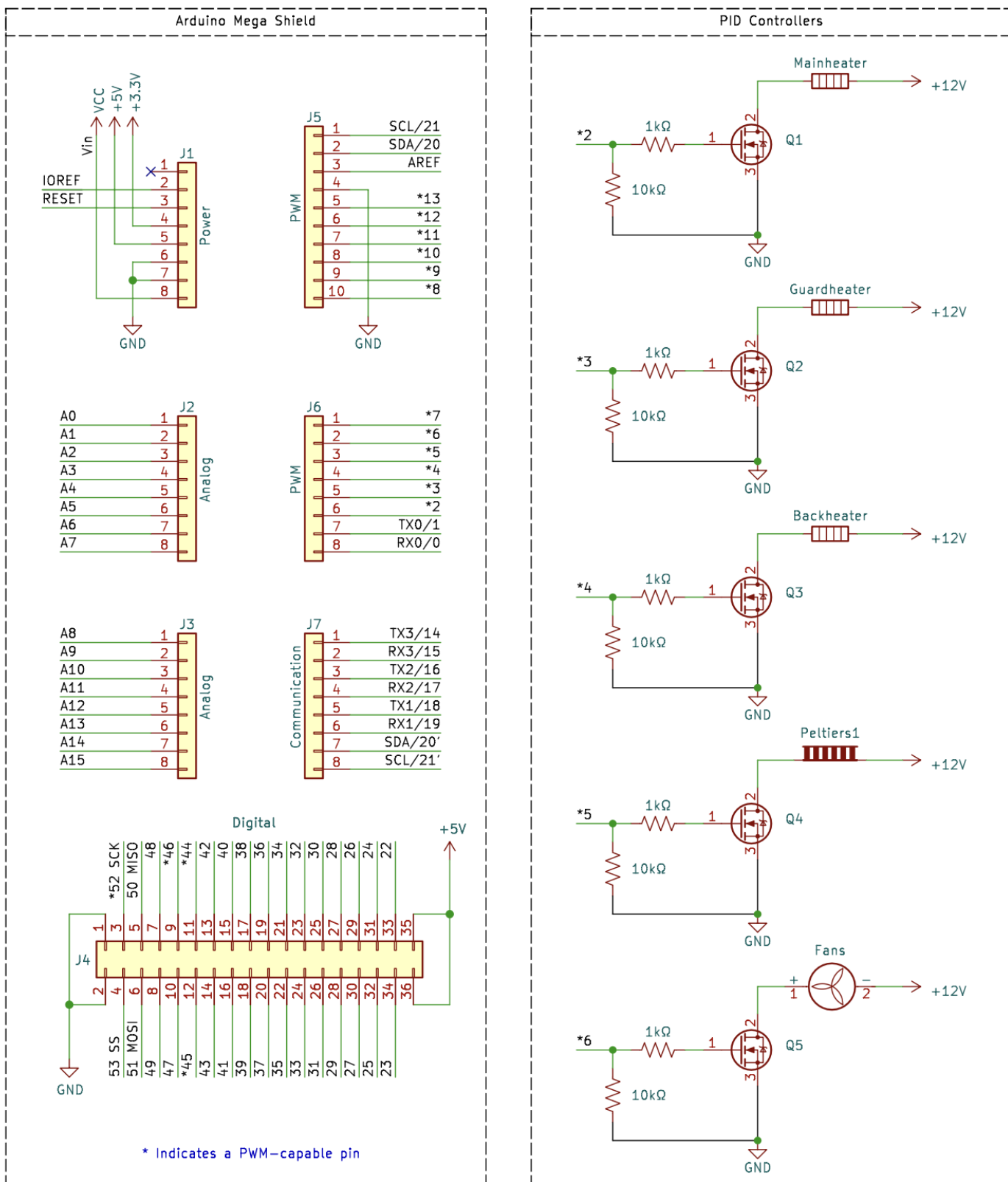


Figure 15: PCB Shield Circuit Diagram Part 1

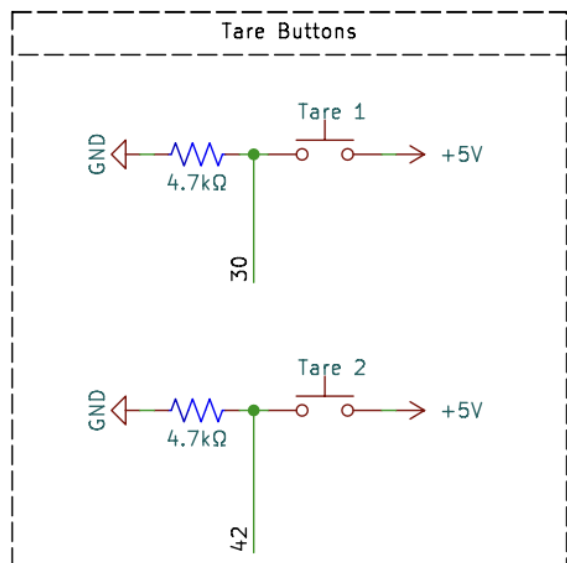
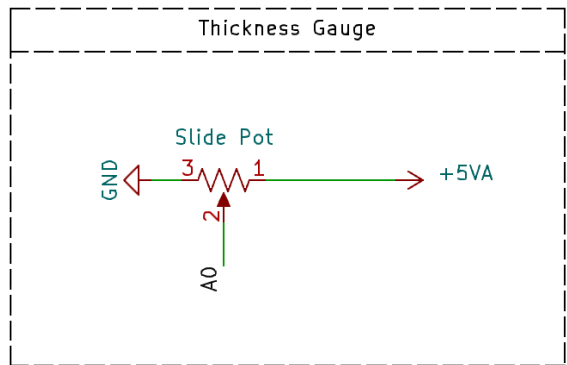
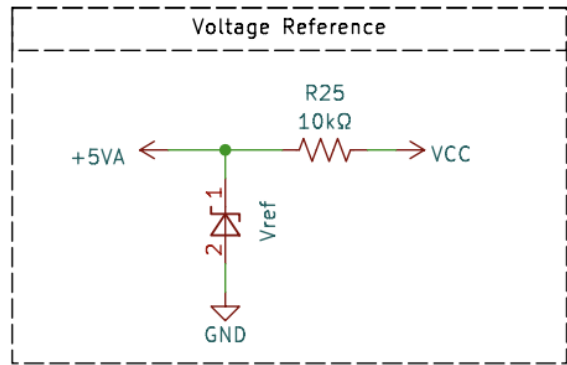
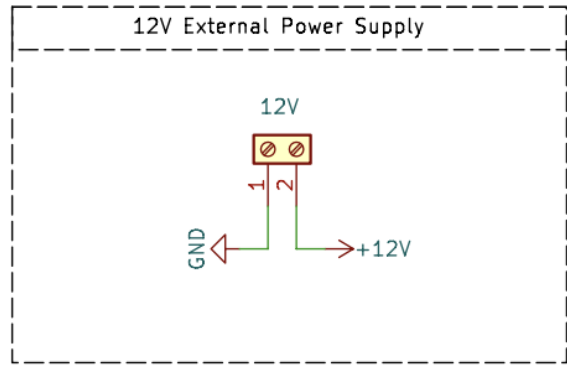
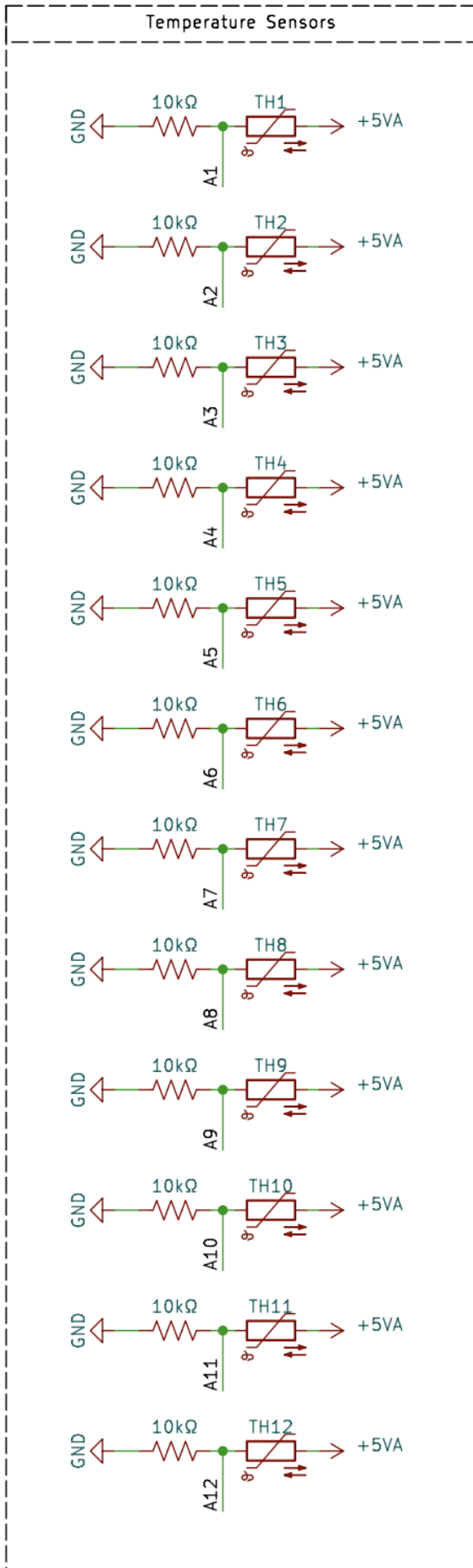


Figure 16: PCB Shield Circuit Diagram Part 2

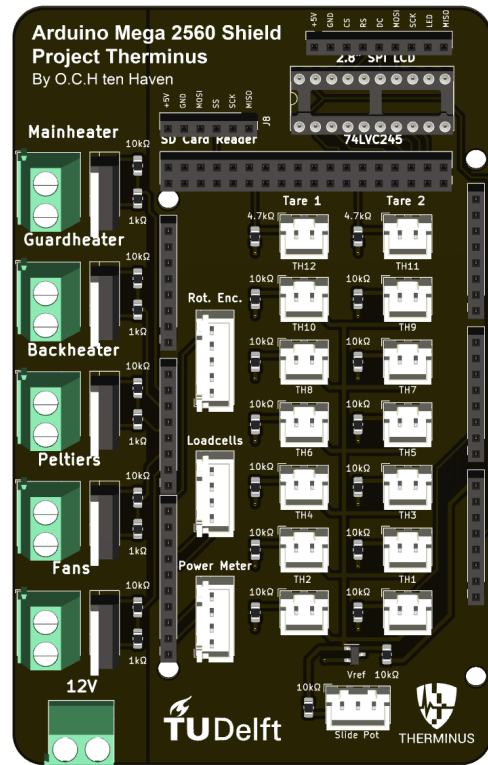
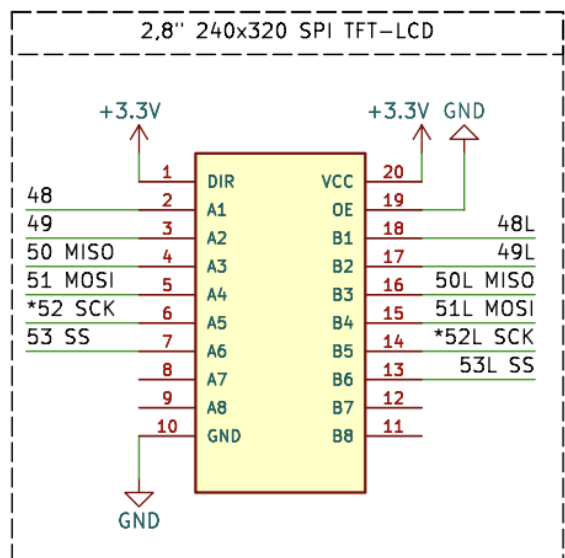
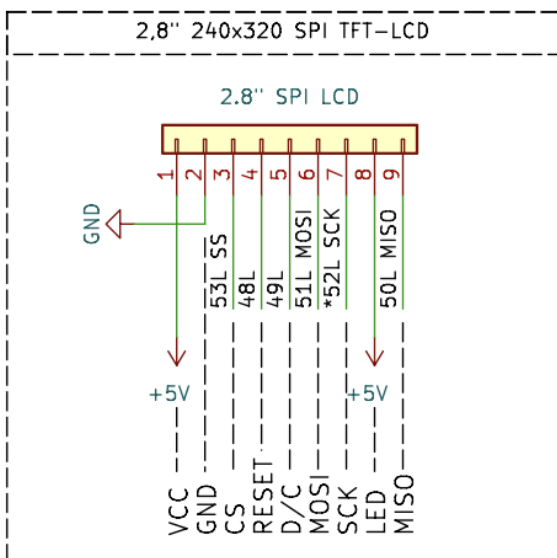
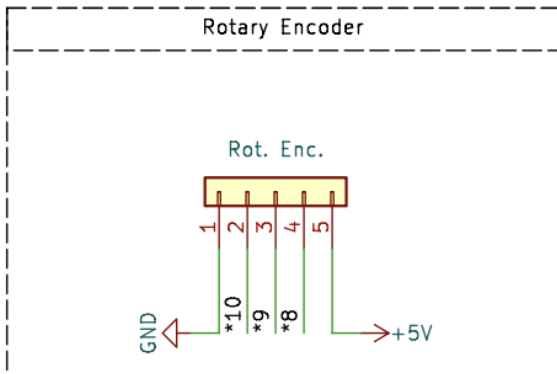
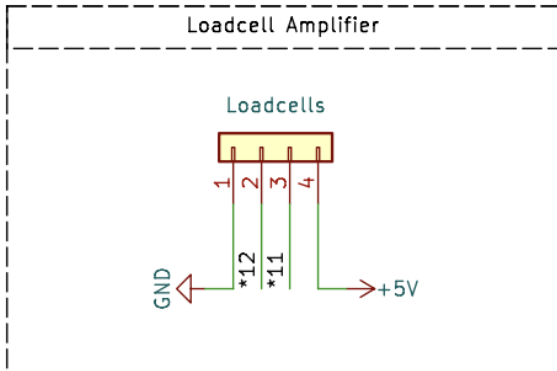
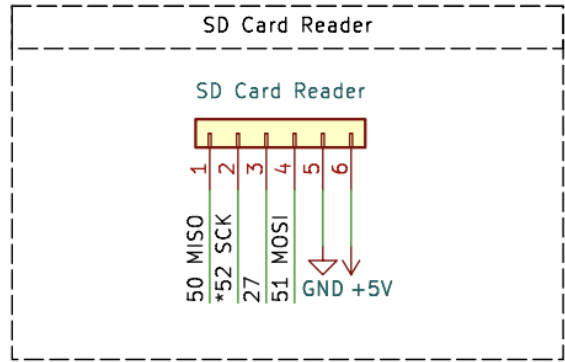
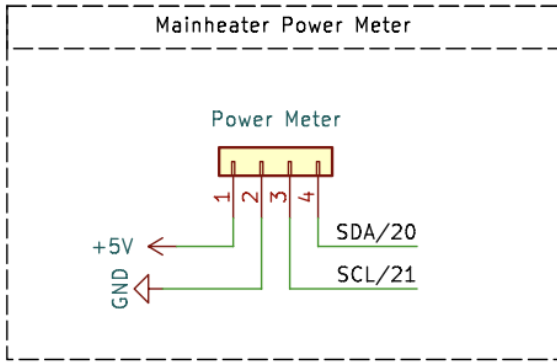


Figure 17: PCB Shield Circuit Diagram Part 3

**Annex 12: Therminus-K3 Prototype Photos**

**Therminus-K3 Prototype Photos by A.J. Loeve**

**Front View**



**Left side View**



*Table 14.1 Therminus-K3 Prototype pictures*



**Right Side View**

Showing air vent in side panel of electronics housing for cooling purposes.



**Back View**

Displaying a power socket for a C13 power cable with an on/off switch located at the lower right corner of the electronics housing.



*Table 14.2 Therminus-K3 Prototype pictures*

### Detailed View of User Interface Module

Showcasing the rotary encoder functioning as the temperature adjuster and enter key, accompanied by two pushbuttons functioning as the tare buttons for the thickness measurement and pressure measurements. The red buttons also function as cancel buttons.



### Detailed View of Compression Stage Assembly

Featuring a handwheel with a knurled handle, mounted with a bearing for smooth operation of the Compression Stage. The quick-release mechanism is switched to the left, indicating that it is engaged with the leadscrew.

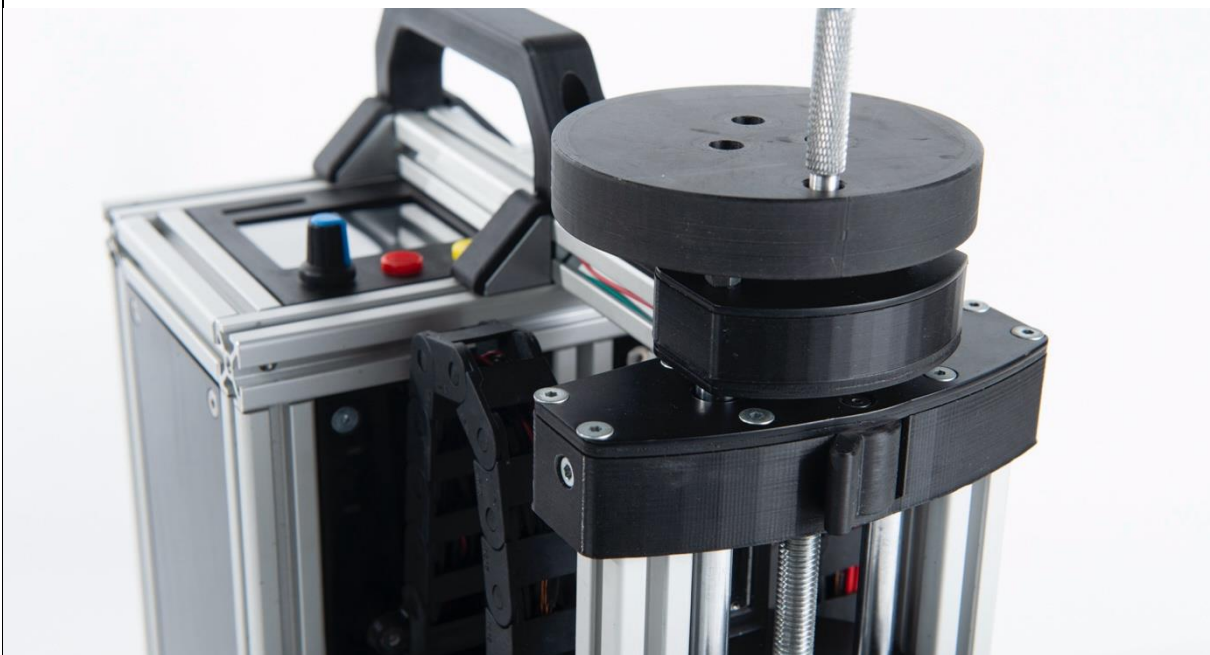


Table 14.3 *Therminus-K3 Prototype pictures*

### Adjustable Feet

The feet can be raised or lowered by simply turning them.



### Main Heater PCB & Flexible ABS Insulation Layer

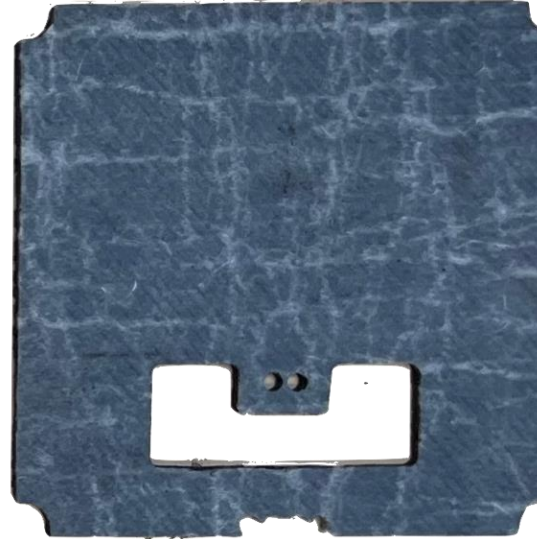
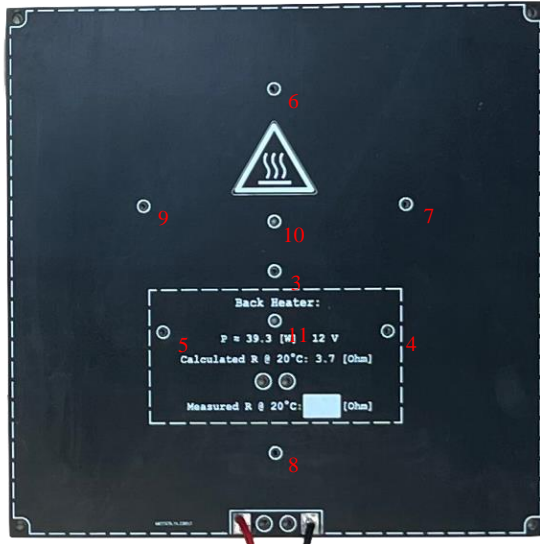
The ABS insulation sheet features die-cut holes for routing the thermistor wires and power cables to the heaters.



Table 14.4 Therminus-K3 Prototype pictures

### Back Heater PCB & Aerogel Insulation Layer

The red numbers denote thermistor identifiers corresponding to the Arduino code detailed in Annex 28. The Spaceloft Aerogel [23] insulation layer incorporates wire routing zones and was cut with a CO2 laser.



### Side View of Hot Stage Stack Assembly

Featuring from top to bottom: Hot Stage Cover, Back Heater PCB, Back Heater Surface Plate, ABS Insulation Layer, Main Heater PCB, and Hot Stage Surface Plate. Black electrical wire grounds both surface plates to minimize measurement noise.

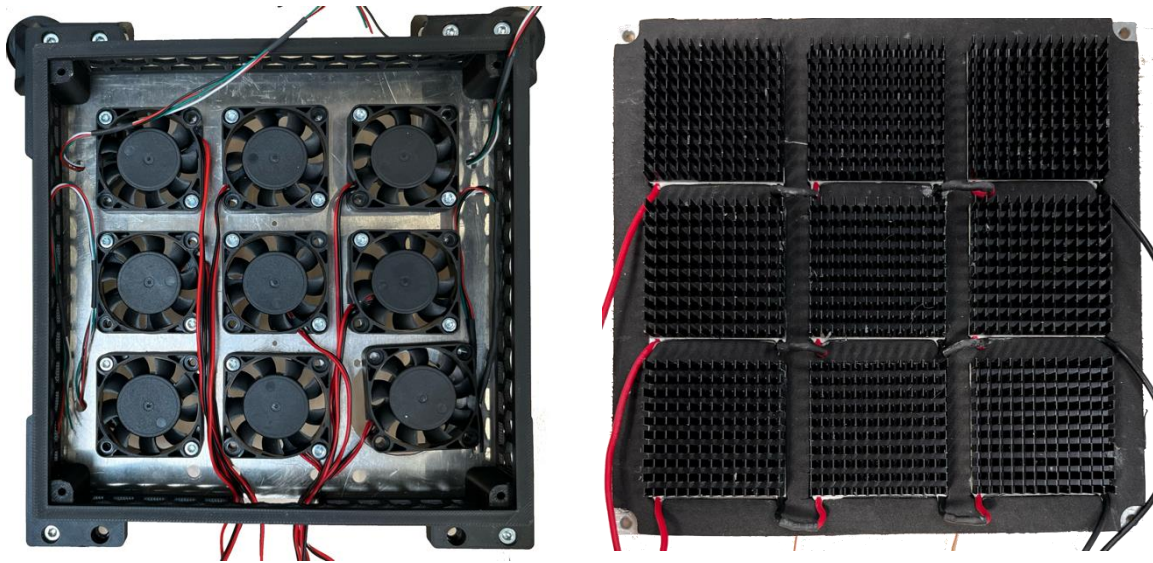


Table 14.5 *Therminus-K3 Prototype pictures*



### Disassembled Cold Stage

**Left:** 3 x 3 array of cooling fans all connected in parallel  
**Right:** 3 x 3 array of Peltier modules and accompanying heatsinks mounted on the Cold Stage's surface plate. Each row is connected in series, and these series are then connected in parallel.



### Custom Arduino Mega PCB shield

Showcasing all the soldered-on components.

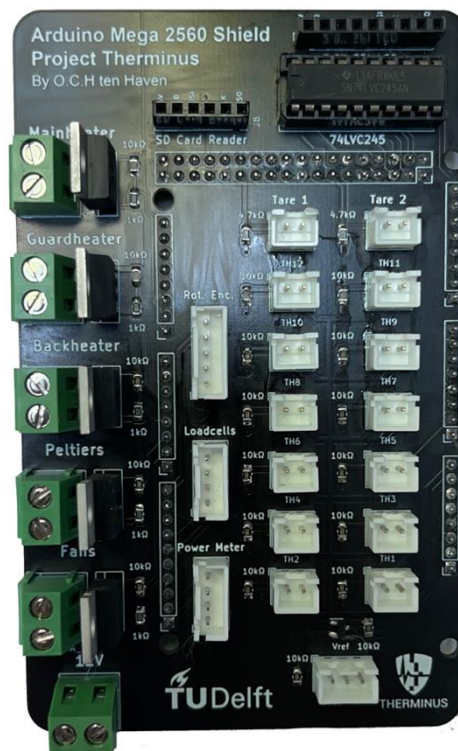


Table 14.6 Thermanus-K3 Prototype pictures




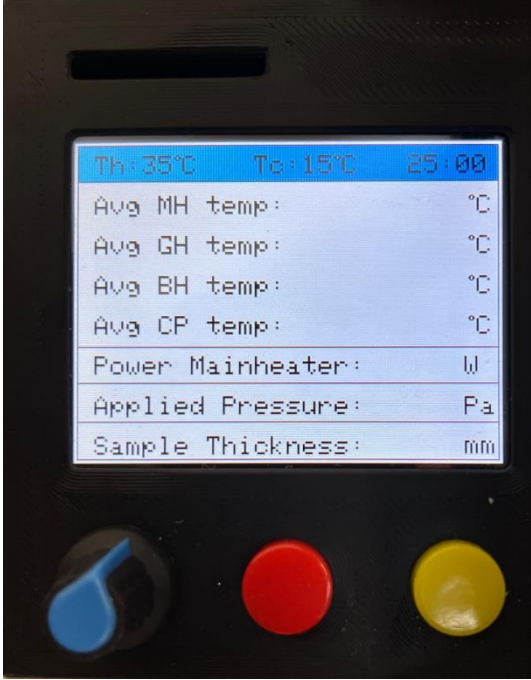
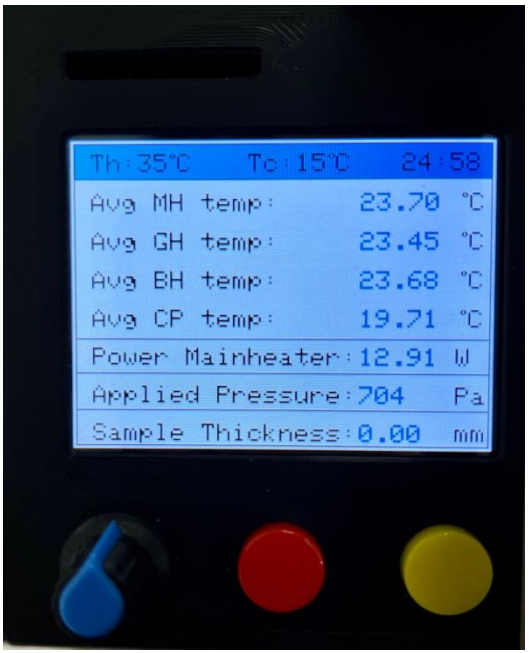

User Interface Module Pictures	
<p><b>Start Screen of the Thermanus-K3 Prototype</b> This 3-second start screen, enables the initialization of all sensors in the background.</p>	<p><b>Main screen</b> The rotary encoder lets you set the temperature of the Hot and Cold Stage, confirming each setting by a press</p>
	
<p><b>Average Temperature Measurement Screen</b> Once the temperatures are set, the pressure and thickness sensors can be zeroed using the red and yellow buttons. Pressing enter again initiates the measurement</p>	<p><b>Individual Temperature Measurement Screen</b> Pressing the rotary encoder again during a measurement displays individual temperatures, while the red button serves as a cancel button stopping the measurement</p>
	

Table 15. User Interface Explanation

## Annex 13: Thermanus-K3 Prototype Build of Materials

Thermanus-K3 BOM:				
Type:	Quantity:	Price:	Total:	Link:
Power Supply: 360W, 12V, 30A	1	€ 22,99	€ 22,99	<a href="https://www.amazon.nl/dp/B08885TXGF?ref=ppx_yo2ov_dt_b_product_details&amp;th=1">https://www.amazon.nl/dp/B08885TXGF?ref=ppx_yo2ov_dt_b_product_details&amp;th=1</a>
Power Socket Switch	1	€ 12,19	€ 12,19	<a href="https://www.amazon.nl/dp/B07PROTP4W?psc=1&amp;ref=ppx_yo2ov_dt_b_product_details">https://www.amazon.nl/dp/B07PROTP4W?psc=1&amp;ref=ppx_yo2ov_dt_b_product_details</a>
Cable Drag Chain	1	€ 12,99	€ 12,99	<a href="https://www.amazon.nl/dp/B08BC58V6L?psc=1&amp;ref=ppx_yo2ov_dt_b_product_details">https://www.amazon.nl/dp/B08BC58V6L?psc=1&amp;ref=ppx_yo2ov_dt_b_product_details</a>
Peltier Elements	9	€ 4,00	€ 36,00	<a href="https://www.tinytronics.nl/shop/nl/mechanica-en-actuatoren/onderdelen/koelementen/peltier-element-60w-tec1-12706">https://www.tinytronics.nl/shop/nl/mechanica-en-actuatoren/onderdelen/koelementen/peltier-element-60w-tec1-12706</a>
Heatsinks	9	€ 2,95	€ 26,55	<a href="https://www.123-3d.nl/123-3D-NEMA17-Heatsink-40-x-40-x-11-mm-i8005-t14804.html">https://www.123-3d.nl/123-3D-NEMA17-Heatsink-40-x-40-x-11-mm-i8005-t14804.html</a>
Fans	9	€ 3,95	€ 35,55	<a href="https://www.123-3d.nl/123-3D-Ventilator-12V-40x40x10mm-axiaal-i76-t15055.html">https://www.123-3d.nl/123-3D-Ventilator-12V-40x40x10mm-axiaal-i76-t15055.html</a>
60 mm Slide Pot	1	€ 3,45	€ 3,45	<a href="https://www.tinytronics.nl/shop/en/components/resistors/potentiometers/10k-slide-potmeter-2-channel-75mm">https://www.tinytronics.nl/shop/en/components/resistors/potentiometers/10k-slide-potmeter-2-channel-75mm</a>
Current Sensor	1	€ 8,00	€ 8,00	<a href="https://www.tinytronics.nl/shop/en/sensors/current-voltage/dfrobot-gravity-i2c-digital-wattmeter-current-and-voltage-sensor-ina219">https://www.tinytronics.nl/shop/en/sensors/current-voltage/dfrobot-gravity-i2c-digital-wattmeter-current-and-voltage-sensor-ina219</a>
Thermistors	12	€ 0,57	€ 6,84	<a href="https://nl.rs-online.com/web/p/thermistor-ics/1811348?gb=s">https://nl.rs-online.com/web/p/thermistor-ics/1811348?gb=s</a>
Full Bridge Load Cell	4	€ 2,50	€ 10,00	<a href="https://www.alibaba.com/product-detail/Full-bridge-4-wires-SC134-flat_60821896052.html?spm=a2756.order-detail-ta-ta-b.0.0.340a2fc20x3m7C">https://www.alibaba.com/product-detail/Full-bridge-4-wires-SC134-flat_60821896052.html?spm=a2756.order-detail-ta-ta-b.0.0.340a2fc20x3m7C</a>
HX711 weight sensor load cell amplifier	1	€ 9,00	€ 9,00	<a href="https://www.alibaba.com/product-detail/5kg-10kg-Scale-Arduino-Load-Cell_1600684745217.html?spm=a2756.order-detail-ta-ta-b.0.0.340a2fc20x3m7C">https://www.alibaba.com/product-detail/5kg-10kg-Scale-Arduino-Load-Cell_1600684745217.html?spm=a2756.order-detail-ta-ta-b.0.0.340a2fc20x3m7C</a>
PCB Heaters	2	€ 2,55	€ 5,10	<a href="https://jlcpcb.com/user-center/orders/">https://jlcpcb.com/user-center/orders/</a>
Rotary Encoder	1	€ 1,50	€ 1,50	<a href="https://www.tinytronics.nl/shop/en/switches/manual-switches/rotary-encoders/rotary-encoder-module">https://www.tinytronics.nl/shop/en/switches/manual-switches/rotary-encoders/rotary-encoder-module</a>
Push Buttons	2	€ 0,40	€ 0,80	<a href="https://www.tinytronics.nl/shop/index.php?route=product/search&amp;search=Tactile%20Pushbutton">https://www.tinytronics.nl/shop/index.php?route=product/search&amp;search=Tactile%20Pushbutton</a>
LCD Screen	1	€ 12,89	€ 12,89	<a href="https://www.amazon.nl/dp/B07RJOJCCS?psc=1&amp;ref=ppx_yo2ov_dt_b_product_details">https://www.amazon.nl/dp/B07RJOJCCS?psc=1&amp;ref=ppx_yo2ov_dt_b_product_details</a>
SD Card Reader	1	€ 2,50	€ 2,50	<a href="https://www.tinytronics.nl/shop/en/data-storage/modules/sd-card-adapter-module-3.3v-5v">https://www.tinytronics.nl/shop/en/data-storage/modules/sd-card-adapter-module-3.3v-5v</a>
JST-ZHR 5p Cable	4	€ 1,00	€ 4,00	<a href="https://www.tinytronics.nl/shop/en/cables-and-connectors/cables-and-adapters/jst-compatible/jst-zhr-5p-to-dupont-female-compatible-cable-15cm">https://www.tinytronics.nl/shop/en/cables-and-connectors/cables-and-adapters/jst-compatible/jst-zhr-5p-to-dupont-female-compatible-cable-15cm</a>
<b>Arduino Mega PCB Shield</b>	1	€ 36,56	€ 36,56	-
Gear M1 15T	1	€ 2,72	€ 2,72	<a href="https://www.dts-products.nl/tandwiel-moduul-1-z-15.html">https://www.dts-products.nl/tandwiel-moduul-1-z-15.html</a>
Gear M1 30T	1	€ 3,75	€ 3,75	<a href="https://www.dts-products.nl/tandwiel-moduul-1-z-30.html">https://www.dts-products.nl/tandwiel-moduul-1-z-30.html</a>

Hardened Liner Rod	1	€ 7,71	€ 7,71	<a href="https://www.123-3d.nl/123-3D-Lineaire-as-staaf-gehard-en-geslepen-met-chroomlaag-10x350-mm-i1779-t13816.html">https://www.123-3d.nl/123-3D-Lineaire-as-staaf-gehard-en-geslepen-met-chroomlaag-10x350-mm-i1779-t13816.html</a>
Arduino Mega	1	€ 42,00	€ 42,00	<a href="https://store.arduino.cc/products/arduino-mega-2560-rev3">https://store.arduino.cc/products/arduino-mega-2560-rev3</a>
Electrical wiring	≈ 2 m	€ 3,27	€ 3,27	<a href="https://nl.farnell.com/staubli/60-7180-22/wire-25m-0-5mm2-tinned-copper/dp/135264">https://nl.farnell.com/staubli/60-7180-22/wire-25m-0-5mm2-tinned-copper/dp/135264</a>
Aluminium Extrusion 20 x 20	≈ 3 m	€ 26,17	€ 26,17	<a href="https://www.tech-specialist.com/nl/type-isb/aluminium-profielen-isb/sleuf-5-isb/aluminum-profiel-isb-20x20n-o-1007465">https://www.tech-specialist.com/nl/type-isb/aluminium-profielen-isb/sleuf-5-isb/aluminum-profiel-isb-20x20n-o-1007465</a>
Aluminium Plate 4mm	0,09 m <sup>2</sup>	€ 7,50	€ 7,50	<a href="https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop">https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop</a>
Aluminium Plate 3mm	0,03 m <sup>2</sup>	€ 1,87	€ 1,87	<a href="https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop">https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop</a>
Steel Plate 2 mm	0,06 m <sup>2</sup>	€ 2,50	€ 2,50	<a href="https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop">https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop</a>
PMMA Sheets	0,15 m <sup>2</sup>	€ 10,46	€ 10,46	<a href="https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop">https://www.tudelft.nl/io/over-io/labs-en-faciliteiten/pmb/materialenshop</a>
Ball Bearing	2	€ 0,50	€ 1,00	<a href="https://www.123-3d.nl/123-3D-Kogellager-met-flens-F624ZZ-1-stuk-i1437-t13131.html">https://www.123-3d.nl/123-3D-Kogellager-met-flens-F624ZZ-1-stuk-i1437-t13131.html</a>
Bush Bearing	4	€ 0,09	€ 0,36	<a href="https://www.verpas.nl/lagers/glijlager-kunststof-pa-kraag/lagerbus-pa-10-x-12-x-16-x-12-x-2/">https://www.verpas.nl/lagers/glijlager-kunststof-pa-kraag/lagerbus-pa-10-x-12-x-16-x-12-x-2/</a>
3D Printed Parts	≈ 600 gr	€ 12,17	€ 12,17	<a href="https://www.123-3d.nl/123-3D-Filament-zwart-1-75-mm-PLA-1-1-kg-Jupiter-serie-i9729-t7316.html">https://www.123-3d.nl/123-3D-Filament-zwart-1-75-mm-PLA-1-1-kg-Jupiter-serie-i9729-t7316.html</a>
Fasteners	-	€ 15,00	€ 15,00	<a href="https://www.fabory.com/nl">https://www.fabory.com/nl</a>
Angle Connector 20 x 20	6	€ 3,50	€ 21,00	<a href="https://www.tech-specialist.com/nl/type-isb/profielverbinders-isb/rechte-hoekverbinders-isb/hoekverbinders-isb/rechte-hoekverbinders-isb-hoekverbinder-set-sleuf-5-20x20-1007699">https://www.tech-specialist.com/nl/type-isb/profielverbinders-isb/rechte-hoekverbinders-isb/hoekverbinders-isb/rechte-hoekverbinders-isb-hoekverbinder-set-sleuf-5-20x20-1007699</a>
<b>Total</b>			<b>€ 404,39</b>	

Table 16. Therminus-K3 Prototype Build of Materials

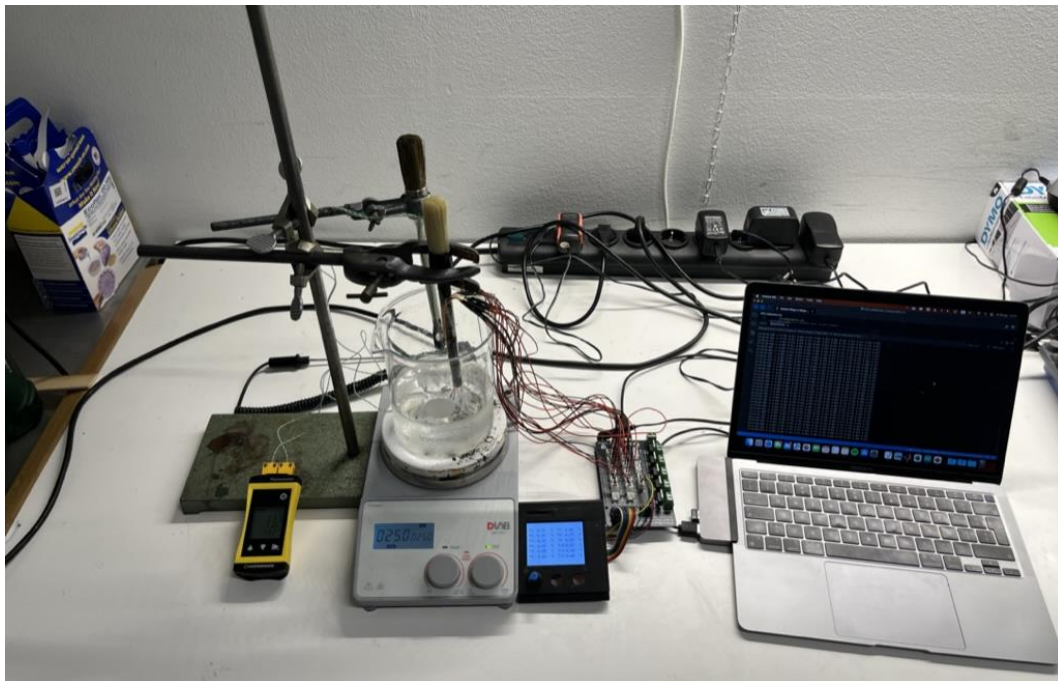
Arduino Mega PCB Shield				
Type:	Quantity:	Price:	Total:	Link:
Custom PCB Arduino Shield	1	€ 1,48	€ 1,48	<a href="https://jlcpcb.com/user-center/orders/">https://jlcpcb.com/user-center/orders/</a>
10kΩ Resistor	19	€ 0,47	€ 8,97	<a href="https://nl.rs-online.com/p/surface-mount-resistors/2430652">https://nl.rs-online.com/p/surface-mount-resistors/2430652</a>
4.7kΩ Resistor	2	€ 0,31	€ 0,63	<a href="https://nl.rs-online.com/p/surface-mount-resistors/0565932">https://nl.rs-online.com/p/surface-mount-resistors/0565932</a>
1kΩ Resistor	5	€ 0,26	€ 1,29	<a href="https://nl.rs-online.com/p/surface-mount-resistors/0565964">https://nl.rs-online.com/p/surface-mount-resistors/0565964</a>
5V Voltage Reference	1	€ 1,81	€ 1,81	<a href="https://nl.rs-online.com/p/voltage-references/8096352">https://nl.rs-online.com/p/voltage-references/8096352</a>
JST XH 2 Header	14	€ 0,13	€ 1,88	<a href="https://nl.rs-online.com/web/p/pcb-headers/8201554">https://nl.rs-online.com/web/p/pcb-headers/8201554</a>

JST XH 2 Female Housing	14	€ 0,09	€ 1,23	<a href="https://nl.rs-online.com/web/p/wire-housings-plugs/8201611">https://nl.rs-online.com/web/p/wire-housings-plugs/8201611</a>
JST XH 3 Header	1	€ 0,18	€ 0,18	<a href="https://nl.rs-online.com/web/p/pcb-headers/8201557?gb=s">https://nl.rs-online.com/web/p/pcb-headers/8201557?gb=s</a>
JST XH 3 Female Housing	1	€ 0,10	€ 0,10	<a href="https://nl.rs-online.com/web/p/wire-housings-plugs/8201614?gb=s">https://nl.rs-online.com/web/p/wire-housings-plugs/8201614?gb=s</a>
JST XH 4 Header	2	€ 0,20	€ 0,40	<a href="https://nl.rs-online.com/web/p/pcb-headers/8201551?gb=s">https://nl.rs-online.com/web/p/pcb-headers/8201551?gb=s</a>
JST XH 4 Female Housing	2	€ 0,10	€ 0,20	<a href="https://nl.rs-online.com/web/p/wire-housings-plugs/8201618?gb=s">https://nl.rs-online.com/web/p/wire-housings-plugs/8201618?gb=s</a>
JST XH 5 Header	1	€ 0,25	€ 0,25	<a href="https://nl.rs-online.com/web/p/pcb-headers/8201560?gb=s">https://nl.rs-online.com/web/p/pcb-headers/8201560?gb=s</a>
JST XH 5 Female Housing	1	€ 0,12	€ 0,12	<a href="https://nl.rs-online.com/web/p/wire-housings-plugs/8201627?gb=s">https://nl.rs-online.com/web/p/wire-housings-plugs/8201627?gb=s</a>
XH Contact	44	€ 0,05	€ 1,98	<a href="https://nl.rs-online.com/p/crimp-contacts/8201529">https://nl.rs-online.com/p/crimp-contacts/8201529</a>
MOSFETs	5	€ 1,13	€ 5,63	<a href="https://nl.rs-online.com/p/mosfets/1300923">https://nl.rs-online.com/p/mosfets/1300923</a>
Octal bus transceiver	1	€ 0,75	€ 0,75	<a href="https://www.tinytronics.nl/shop/en/components/ics-and-microcontroller-chips/ics/sn74lvc245a-octal-bus-transceiver-20-pin-dip">https://www.tinytronics.nl/shop/en/components/ics-and-microcontroller-chips/ics/sn74lvc245a-octal-bus-transceiver-20-pin-dip</a>
Through Hole DIP Socket	1	€ 1,56	€ 1,56	<a href="https://nl.rs-online.com/web/p/dil-sockets/6742482?gb=s">https://nl.rs-online.com/web/p/dil-sockets/6742482?gb=s</a>
2 Contact Screw Terminal	6	€ 1,08	€ 6,50	<a href="https://nl.rs-online.com/web/p/pcb-terminal-blocks/1930564?gb=s">https://nl.rs-online.com/web/p/pcb-terminal-blocks/1930564?gb=s</a>
40 Pins Header	3	€ 0,40	€ 1,20	<a href="https://www.tinytronics.nl/shop/en/cables-and-connectors/connectors/pin-headers/male/40-pins-header-male">https://www.tinytronics.nl/shop/en/cables-and-connectors/connectors/pin-headers/male/40-pins-header-male</a>
9 pin female Header	1	€ 0,24	€ 0,24	<a href="https://www.tinytronics.nl/shop/en/cables-and-connectors/connectors/pin-headers/female/9-pins-header-female">https://www.tinytronics.nl/shop/en/cables-and-connectors/connectors/pin-headers/female/9-pins-header-female</a>
6 pin female Header	1	€ 0,18	€ 0,18	<a href="https://www.tinytronics.nl/shop/en/cables-and-connectors/connectors/pin-headers/female/6-pins-header-female">https://www.tinytronics.nl/shop/en/cables-and-connectors/connectors/pin-headers/female/6-pins-header-female</a>
<b>Total</b>			<b>€ 36,56</b>	

*Table 17. Custom Arduino Mega PCB Shield*

## Annex 14A: Thermistor Calibration Setup

The 12 Murata 10k $\Omega$  NTC thermistors [15] were calibrated with the setup as shown in Figure 18. They were immersed in an iced water bath placed on a magnetic stirrer, along with two reference thermistors connected to a Greisinger G1700 thermometer [16]. To calibrate, warm water was slowly added to the bath, increasing the temperature in 2-degree increments from 2 to 50 degrees Celsius. At each increment, the reference temperature and the voltage outputs from each thermistor were recorded. The voltage at each temperature point was determined by averaging four consecutive voltage readings, taken within one second. The reference temperature was calculated by averaging the readings from the two reference thermistors. Following this, the voltage readings were linked to the reference temperatures through mathematical modelling in MATLAB [24]. To achieve this, various polynomial functions were fitted to the data. Among these, the conventional Steinhart-Hart [25] fitting model and a 4th-order polynomial fit were found to yield the best results, accurately correlating the thermistor readings with the known temperatures. The detailed calibration results, including the models and their effectiveness, are presented in Annex 14B. The procedure was repeated to validate the calibration, validation results are presented in Annex 14C.



**Figure 18: Thermistor Calibration Setup.**

*Displaying, from left to right: Greisinger G1700 thermometer for reference measurements, a magnetic stirrer topped with an iced water bath for consistent temperature control, an LCD screen module for real-time data display, the Arduino Mega coupled with a Custom PCB shield for control and data processing, and a computer for data acquisition.*



## Annex 14B: Thermistor Calibration Results

### 4<sup>th</sup> Order polynomial fit between voltage and temperature:

Figure 19 presents the calibration results of all 12 NTC thermistors, illustrating the relationship between the voltage output from each thermistor and its corresponding temperature reading. This relationship is quantified using a fourth-order polynomial fit, which can be expressed in the following form:

$$T(V_{out}) = A \cdot V_{out}^3 + B \cdot V_{out}^2 + C \cdot V_{out} + D$$

In this equation,  $T(V_{out})$  represents the temperature corresponding to a given voltage output  $V_{out}$ , and A, B, C, and D are the coefficients determined through the polynomial fitting process, as presented in Table 18. This fit enables the conversion of voltage readings from the thermistors directly into temperature measurements.

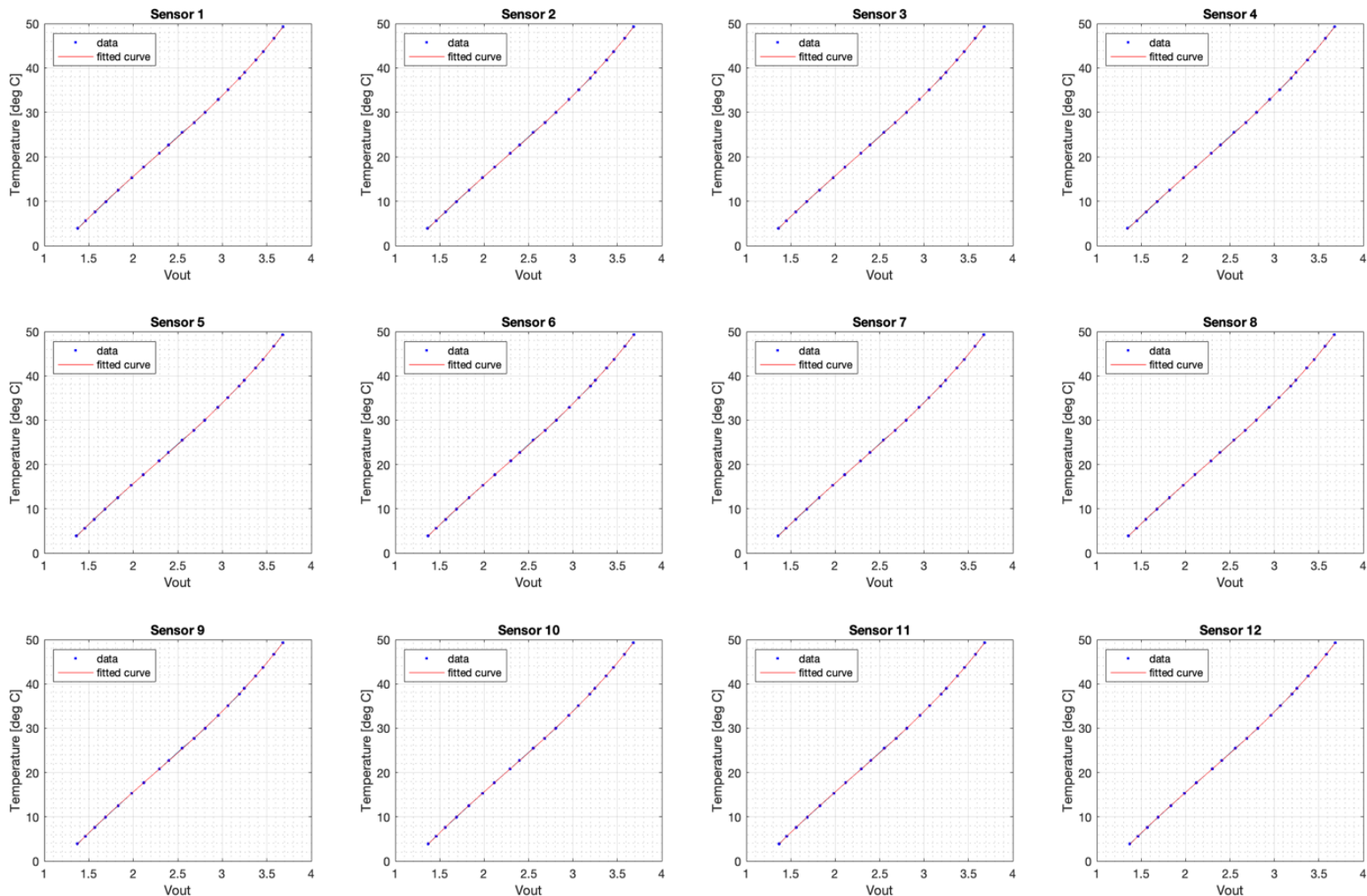


Figure 19: 4<sup>th</sup> Order Polynomial Fit Between Voltage and Temperature

4 <sup>th</sup> Order polynomial Thermistor Calibration Coefficients					
Sensor	A	B	C	D	Resolution °C
1	-34.918	38.163	-9.1814	1.3646	0.0871
2	-32.731	35.726	-8.3117	1.2642	0.0873
3	-32.761	35.963	-8.4200	1.2801	0.0873
4	-30.744	33.540	-7.4808	1.1628	0.0876
5	-33.120	36.301	-8.5292	1.2906	0.0872
6	-32.970	35.977	-8.4041	1.2742	0.0872
7	-32.799	36.047	-8.4438	1.2820	0.0873
8	-32.513	35.702	-8.2998	1.2642	0.0875
9	-33.909	37.035	-8.7768	1.3190	0.0873
10	-33.830	37.016	-8.7865	1.3219	0.0873
11	-32.793	35.863	-8.3856	1.2763	0.0873
12	-33.258	36.279	-8.5301	1.2926	0.0873

Table 18. 4<sup>th</sup> Order polynomial Calibration Coefficients and Measurement Resolution

To get an idea about the accuracy of the fit, the Root Mean Square Error (RMSE) of each thermistor's readings compared to its own fit was calculated according to the following formula:

$$RMSE_i = \sqrt{\frac{1}{N} \sum_{k=1}^N (T_{measured,k} - T_{fit,k})^2}$$

- $RMSE_i$  is the RMSE for the  $i$ -th thermistor.
- $N$  is the total number of data points (temperature readings) for the thermistor.
- $T_{measured,k}$  is the actual temperature at the  $k$ -th data point.
- $T_{fit,k}$  is the temperature predicted by the polynomial fit for the  $k$ -th data point.

The RMSE of each thermistor is depicted in Figure 20, with an average RMS error of 0.065 °C.

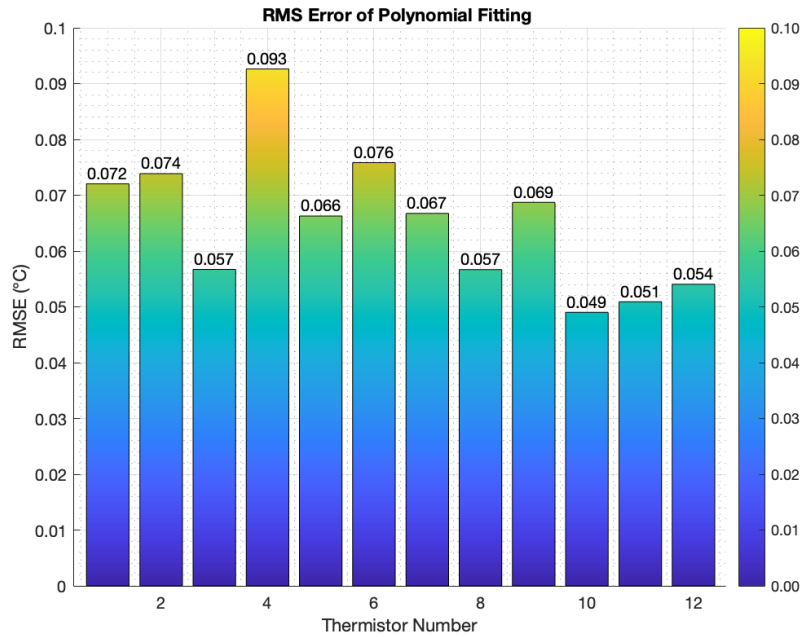


Figure 20: RMSE Between Temperature Readings and 4<sup>th</sup> Order Polynomial Fit

The measurement resolution of each fit was also determined using the Analog to Digital Converter (ADC) resolution of the 10-bit Arduino Mega:

$$ADC_{Res} = \frac{V_{ref}}{steps} = \frac{4.995}{2^{10} - 1} = 0.0049 \frac{V}{step}$$

The measurement resolution was then determined by calculating the change in temperature corresponding to the smallest change in voltage, one ADC step, for every polynomial fit. The measurement resolution was observed to range between 0.0871 and 0.0876 °C across the different polynomial fits, as shown in Table 18. Consequently, the resolution is estimated to be 0.0876 °C, as this value represents the highest, and thus limiting, resolution among the observed resolutions.

### Steinhart-Hart fit between Resistance and temperature:

The Steinhart-Hart fit is a more conventional fit that relates the variable resistance of the thermistor with the temperature. So, in order to use this method, the voltage readings first have to be converted to a resistance value, which was done using the voltage divider rule, which relates the output voltage to the ratio of the thermistor's resistance to the total resistance in the circuit:

$$R_{NTC} = R \cdot \left( \frac{V_{Ref}}{V_{out}} - 1 \right)$$

- $R_{NTC}$  represents the thermistor's resistance.
- $R$  is the known reference resistance in the voltage divider circuit.
- $V_{Ref}$  is the reference voltage supplied to the circuit.
- $V_{out}$  is the output voltage measured across the thermistor.

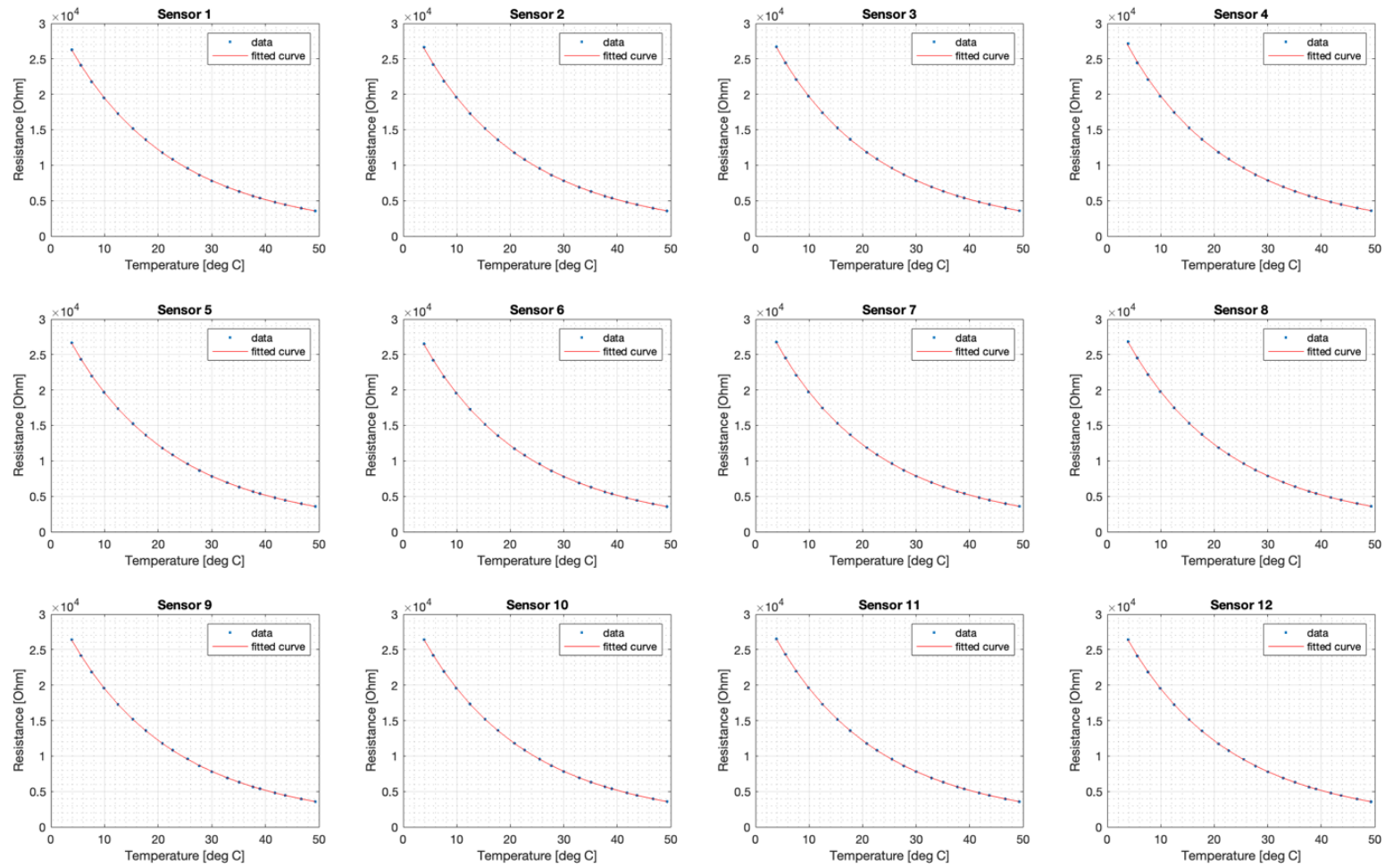
Then the resistance can be fitted to the temperature readings using the Steinhart-Hart equation:

$$\frac{1}{T} = A + B \cdot \ln(R_{NTC}) + C \cdot \ln(R_{NTC})^3$$

In this equation,  $T$  represents the temperature corresponding to a given thermistor resistance  $R_{NTC}$  and A, B, and D are the Steinhart-Hart coefficients determined through the calibration process. The Steinhart-Hart coefficients and the corresponding measurement resolutions are presented in Table 19, with 0.0881 °C being the limiting resolution. Figure 21 depicts the temperature vs resistance graphs of each thermistor.

Steinhart-Hart Thermistor Calibration Coefficients				
Sensor	A	B	C	Resolution °C
1	1.21050e-03	2.22950e-04	1.23890e-07	8.81e-02
2	1.11640e-03	2.38790e-04	5.79940e-08	8.78e-02
3	1.10830e-03	2.39930e-04	5.30680e-08	8.78e-02
4	1.05560e-03	2.48790e-04	1.61570e-08	8.77e-02
5	1.13230e-03	2.36040e-04	6.86490e-08	8.79e-02
6	1.13220e-03	2.36330e-04	6.72470e-08	8.78e-02
7	1.11480e-03	2.38780e-04	5.74630e-08	8.79e-02
8	1.10640e-03	2.39950e-04	5.38770e-08	8.80e-02
9	1.16050e-03	2.31130e-04	9.13360e-08	8.81e-02
10	1.15130e-03	2.32680e-04	8.45090e-08	8.81e-02
11	1.10370e-03	2.40820e-04	5.02630e-08	8.78e-02
12	1.12170e-03	2.37860e-04	6.31540e-08	8.79e-02

Table 19. Steinhart-Hart Coefficients



**Figure 21: Steinhart-Hart Fit Between Resistance and Temperature**

Similar as before, the RMSE between the thermistor readings and the Steinhart-Hart fit was calculated, yielding an average RMSE of 0.063 °C. Although the Steinhart-Hart fitting demonstrates a slightly lower RMSE, indicating potentially higher accuracy, its resolution is marginally higher compared to the polynomial fit. However, the difference in performance between the two fitting methods is negligible. Thus, the decision was made to use the polynomial fit due to its simpler implementation in the Arduino code. Additionally, the conversion from voltage to resistance required for the Steinhart-Hart model could introduce a slight error, further justifying the preference for the polynomial approach. The RMSE values for each of the thermistors for the Steinhart-hart fitting are shown in Figure 22.

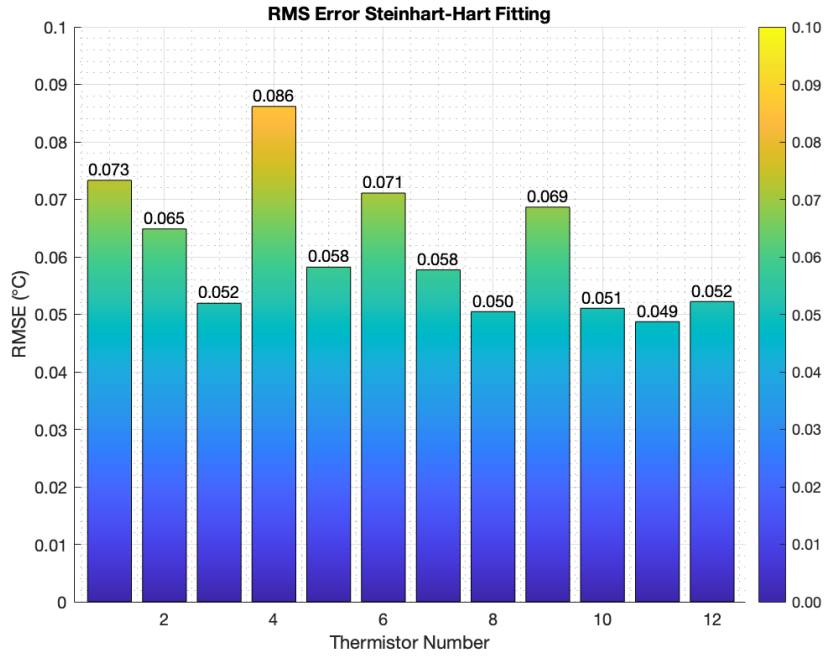


Figure 22: RMSE between temperature readings and Steinhart-Hart Fit

## Annex 14C: Thermistor Validation Results

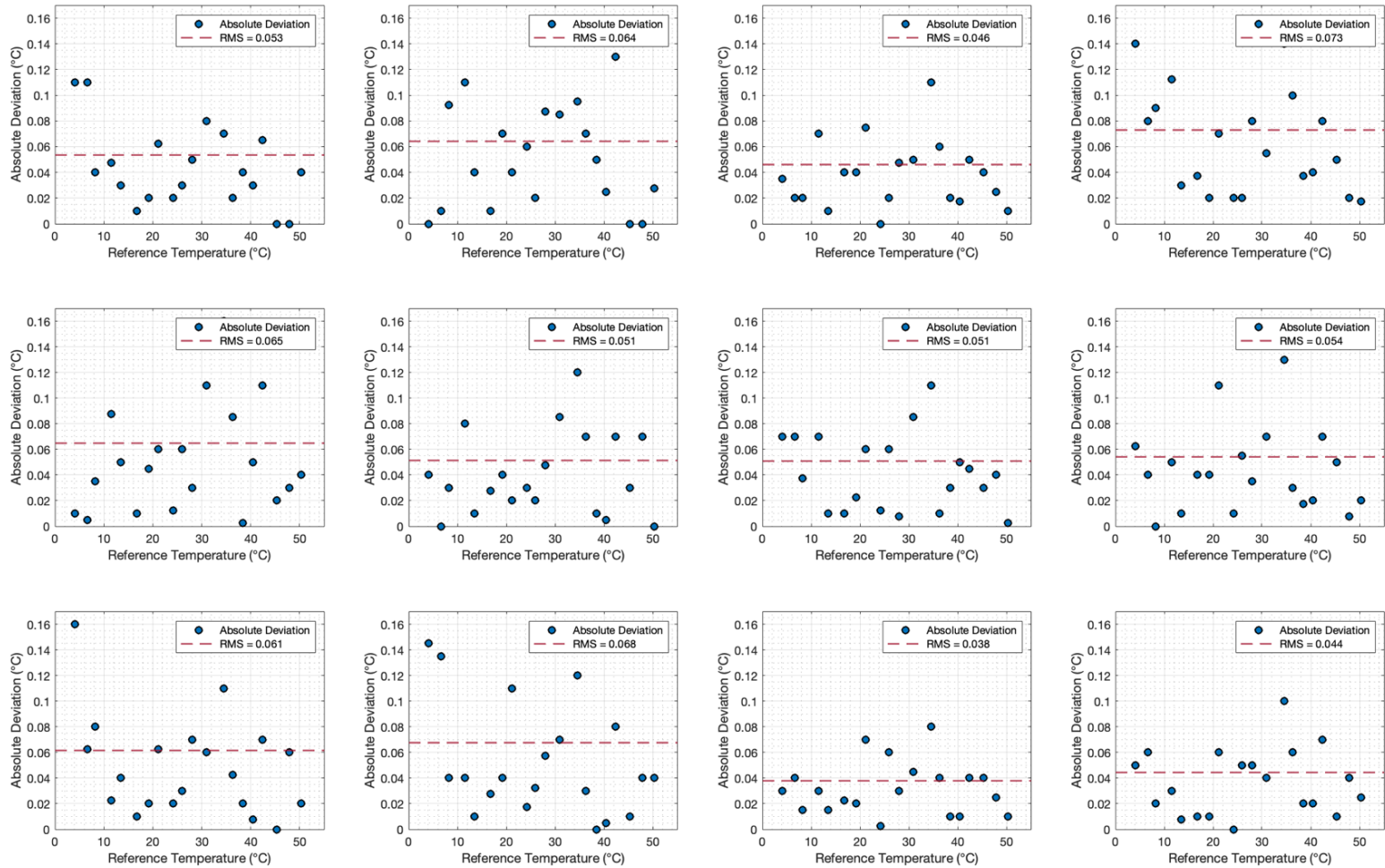
The calibration's effectiveness was verified by repeating the calibration process. This time, the temperature readings from the thermistors, derived from the 4th order polynomial fit, were directly compared against the readings from two reference thermistors. The outcomes of this validation are displayed for each thermistor in Figure 23**Figure**, highlighting the absolute deviation of each reading from the reference temperature. The RMSE for every thermistor was computed across the entire temperature range relative to the reference temperatures, resulting in an average RMSE of 0.056 °C and a maximum RMSE of 0.073 °C.

To assess the measurement consistency among thermistors, the pairwise RMSE was computed for each thermistor pair, according to:

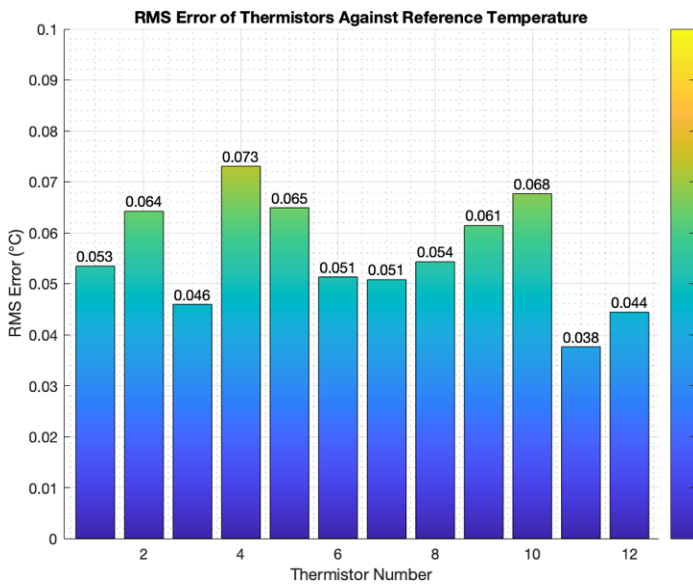
$$RMSE_{i,j} = \sqrt{\frac{1}{N} \sum_{k=1}^N (T_{i,k} - T_{j,k})^2}$$

- $RMSE_{i,j}$  is the RMSE between thermistor  $i$  and thermistor  $j$ .
- $N$  is the total number of data points (temperature readings) for the thermistor.
- $T_{i,k}$  is the temperature reading from thermistor  $i$  at the  $k$ -th data point.
- $T_{j,k}$  is the temperature reading from thermistor  $j$  at the  $k$ -th data point.

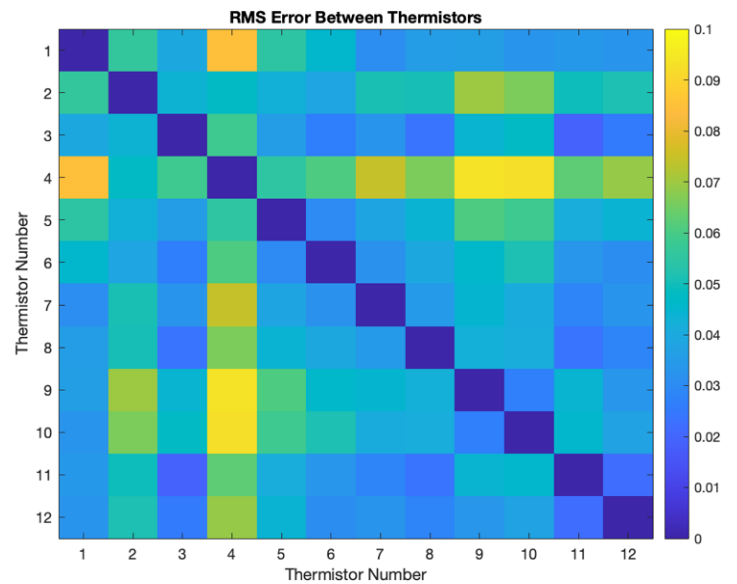
Figure 23 illustrates the RMSE results against the reference temperatures, while Figure 25 displays the pairwise RMSE findings.



**Figure 23: Thermistor Readings Deviation and RMS Error Compared to Reference Temperature**



**Figure 24: RMSE Thermistor Validation**  
 Average RMS = 0.056 °C.  
 Maximum RMS = 0.073 °C.

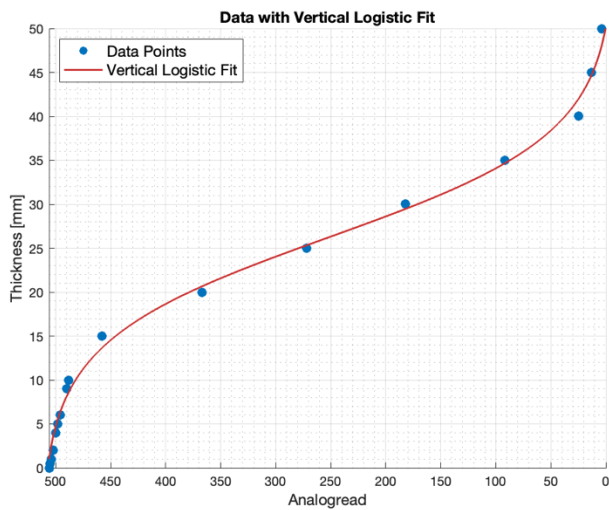


**Figure 25: Pairwise RMSE Thermistor Validation**  
 Average Pairwise RMS = 0.041 °C.  
 Maximum Pairwise RMSE = 0.094 °C.



## Annex 15A: Thickness Sensor Calibration

The thickness measurements were calibrated by measuring Cary Le Locle Suisse gauge blocks [26] with a precision of  $\pm 0.15 \mu\text{m}$  [27] and thicknesses ranging from 0.50 mm to 50.00 mm. Figure 25 shows the initial calibration measurements covering 50 mm of the 60 mm range of the slide potentiometer. The variable resistance was clearly not linear, so a logistic function was fitted to the readings using MATLAB. To get an idea about the accuracy of the logistic fit the RMSE was calculated comparing the measurements with the logistic fit. This resulted in an unacceptable RMSE of 1.15 mm. Therefore, the Terminus-K3's thickness measurement range was modified by extending the aluminium extrusions connecting the Hot Stage to the Linear Carriage. This modification ensured that the thickness measurement range largely fell within the linear region of the slide potentiometer. Subsequently, new calibration measurements were conducted, and the results are depicted in Figure 24. The majority of the data points displayed a linear behaviour, allowing for a linear regression fit to be applied to these points. As a result, the new RMSE was reduced to only 0.055 mm, confirming the accuracy of the linear fit.

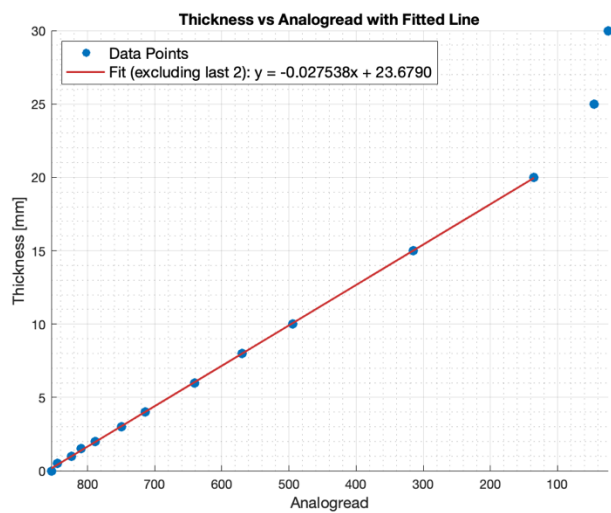


$$L(x) = a - \frac{b}{1 + e^{-k(x-x_0)}}$$

- $a = 513.2077$
- $b = 521.9314$
- $k = 0.16964$
- $x_0 = 26.2202$

RMSE to Logistic Fit = 1.15 mm

Figure 25: Thickness Measurements with Logistic Fit



$$L(x) = ax + b$$

- $a = -0.027538$
- $b = 23.6790$

RMSE to Linear Fit = 0.055 mm

Figure 25: Thickness Measurements with Linear Fit

## Annex 15B: Thickness Sensor Validation

To validate the linear regression fit from the calibration, new measurements were taken of gauge blocks ranging between 0 and 20 mm. Figure 26 illustrates the absolute deviation between these measurements and the actual gauge block thicknesses, along with the calculated RMSE over the entire range. The RMSE was found to be 0.0132 mm, affirming the precision of the thickness measurement system. Following the method used in Annex 14, the measurement resolution was determined by assessing the thickness change corresponding to the smallest detectable Analogread value change, set at 1. This analysis yielded a measurement resolution of 0.028.

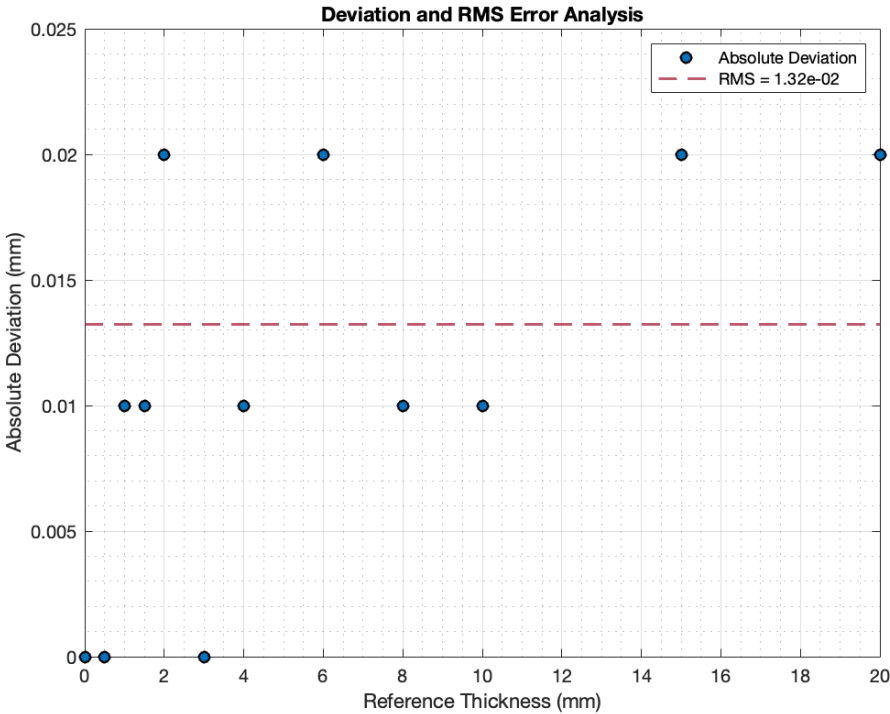
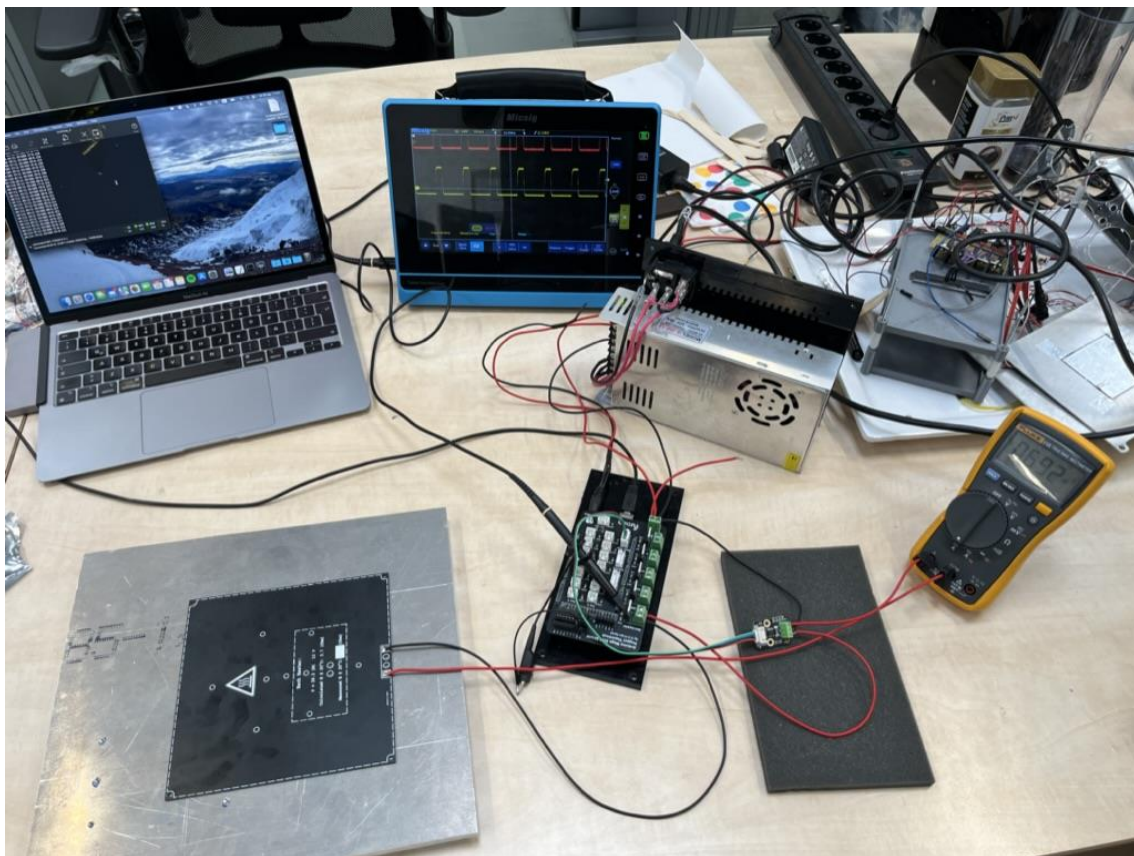


Figure 26: Thickness Readings Deviation and RMS Error compared to Gauge Block Thicknesses

## Annex 16A: INA226 Current Sensor Calibration Setup

The current sensor of the INA226 module was calibrated by incrementally increasing the power to the main heater PCB and comparing the current measurement of the INA226 with and Fluke 115 True RMS digital Multimeter [28]. The power was increased by increasing the PWM output signal controlling the MOSFET circuit of the main heater on custom Arduino PCB shield. To calibrate the current readings of the INA226 module, a linear regression fit was applied between its current readings and those from the Fluke Multimeter using MATLAB. Figure 29 shows the calibration set up. The Fluke Multimeter was connected in series with the INA226 Power module and the main heater PCB. The main heater PCB was placed on a thick aluminium plate to dissipate the generated heat. The setup was powered by the 12V power supply intended for use in the final prototype, maintaining consistency with the prototype's final design and components. The INA226's readings were recorded using a laptop utilizing a specialized Arduino library for interfacing with the INA226 module [29]. Additionally, a Micsig TO202A digital oscilloscope [30] was used to monitor the proper functioning of the PWM signal powering the heater.

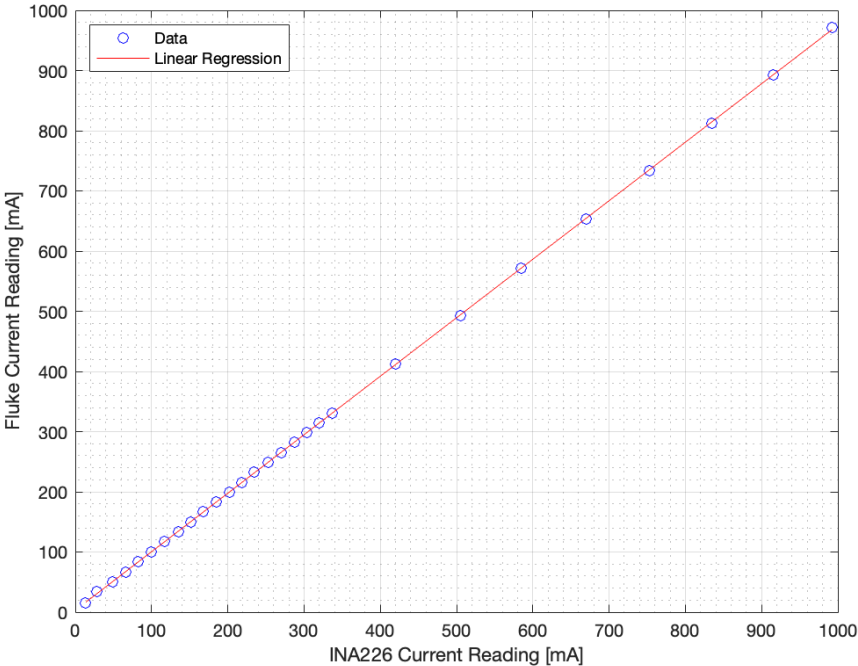


**Figure 29: INA226 Current Sensor Calibration Setup**

**Top:** (from left to right) Laptop for Data Logging, Micsig TO202A Digital Oscilloscope, 12V Power supply  
**Bottom:** (from left to right) Main heater PCB on Aluminium Heat Dissipation Plate, Arduino Mega + Custom PCB Shield, INA226 Power module, Fluke 115 True RMS Digital Multimeter.

# Annex 16B: INA226 Current Sensor Calibration Results

Figure 30 illustrates the results of the calibration measurements of the INA226 current sensor, compared to the readings of the Fluke 115 True RMS digital Multimeter. The data demonstrates a strong linear correlation, suggesting that with a correct linear regression fit, the INA226 module is capable of accurately measuring current. The linear regression fit, as determined by MATLAB, shows a RMSE of only 1.1586 mA.



$$I(x) = ax + b$$

- $a = 0.97214$
- $b = 3.1859$

RMSE to Linear Fit = 1.1586 mA

**Figure 30: INA226 Current Readings vs Fluke 115 Current Readings and Linear Regression Fit**  
*Red line indicates linear regression fit between the current readings.*

### Annex 16C: INA226 Current Sensor Validation Results

A series of validation measurements similar to those described in Annex 14A were carried out for the INA226 current sensor to verify the accuracy of its calibration. The sensor's calibrated readings were again compared with those from the Fluke Multimeter, with the absolute deviations displayed in Figure 31. The tests resulted in a RMSE of 0.424 mA, indicating the sensor's accuracy in measuring current. The maximum deviation between the INA226 sensor and the Fluke Multimeter readings was just 0.783 mA, while the measurement resolution of the INA226 current sensor is 0.1mA.

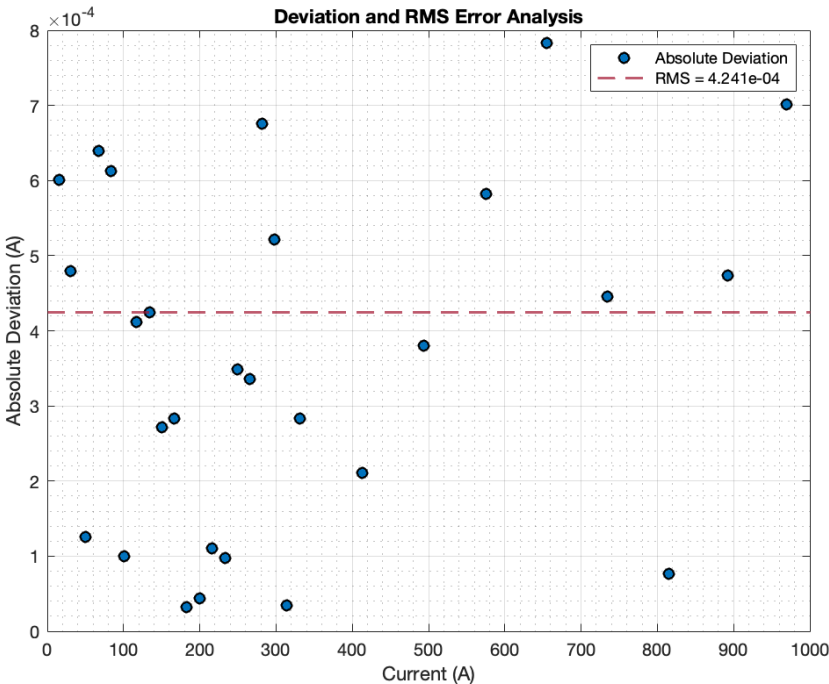


Figure 31: INA226 Current Readings Deviation and RMS Error compared to Fluke 115 True RMS Digital Multimeter



## Annex 17: Load Cell Calibration and Validation

The calibration of the HX711 Loadcell Amplifier [31], which combines the readings from the SC134 Full Bridge Micro Loadcells [32], was performed using the HX711\_ADC Arduino Library [33]. This library not only facilitated the calibration process but also served as the method for communicating with the HX711 module. The process started with zeroing the scale to account for the unloaded weight of the Cold Stage. Then, a known weight was placed on the Cold Stage's surface plate and its weight was entered into the Arduino script [33]. The script from the library then calculated the calibration factor assuming a linear relationship, using the formula:

$$\text{Calibration value} = \frac{\text{Raw HX711 Data at Known Weight}}{\text{Known weight [g]}}$$

The script generated a calibration factor of 51.76. To verify the accuracy of this calibration, a series of weight measurements were conducted using known weights. The outcomes of these tests are detailed in Table 20. Before conducting these tests, the known weights were validated for accuracy using a Kern EMB5.2K5 scale. The validation process demonstrated a high level of precision, with an error margin of less than 0.2%.

Load Cell Validation Measurements			
Weight:	Kern EMB5.2K5	Compression Stage:	Accuracy:
500g	500 g	500 g	0.00 %
2 x 500g	1000 g	1001 g	0.10 %
3 x 500g	1500 g	1502 g	0.13 %
1 x 2000g	2000 g	2001 g	0.05 %
1 x 2000g + 1 x1000g	3000 g	3005 g	0.17 %

Table 20. HX711 Validation Measurements Compared to Kern EMB5.2K5 Scale

## Annex 18: Thermal Uniformity Verification Surface Plates

The tests aimed to verify the thermal uniformity of the surface plates of the Hot and Cold Stage were conducted using the setup depicted in Figure 32. This setup consisted of a tripod holding the Flir E75 Advanced Thermal Camera [34]. To mitigate the thermal reflective properties of the bare aluminium plates, painters' tape was applied to their surfaces before capturing thermal images. To minimize environmental thermal interference during the image and video capture process, cardboard panels were placed around the background of both the Cold and Hot stage. Thermal images were taken to confirm the thermal uniformity of the surface plates, and thermal videos were recorded to track how quickly thermal uniformity was achieved. Table 21 displays still images extracted from the video recordings of both the Hot and Cold stages at 10 seconds and 120 seconds. At 10 seconds, the temperature distribution across the surface plates is not uniform. However, by 120 seconds, the Cold Stage's surface plate already demonstrates thermal uniformity, whereas the Hot Stage's surface plate still needs more time to achieve complete thermal uniformity. This is evident from the slightly higher temperature observed in the main heater area of the Hot Stage surface plate. But this temperature difference disappears over time, as demonstrated in the thermal images presented in the paper [35].



*Figure 32: Thermal Uniformity Verification Test Setup*

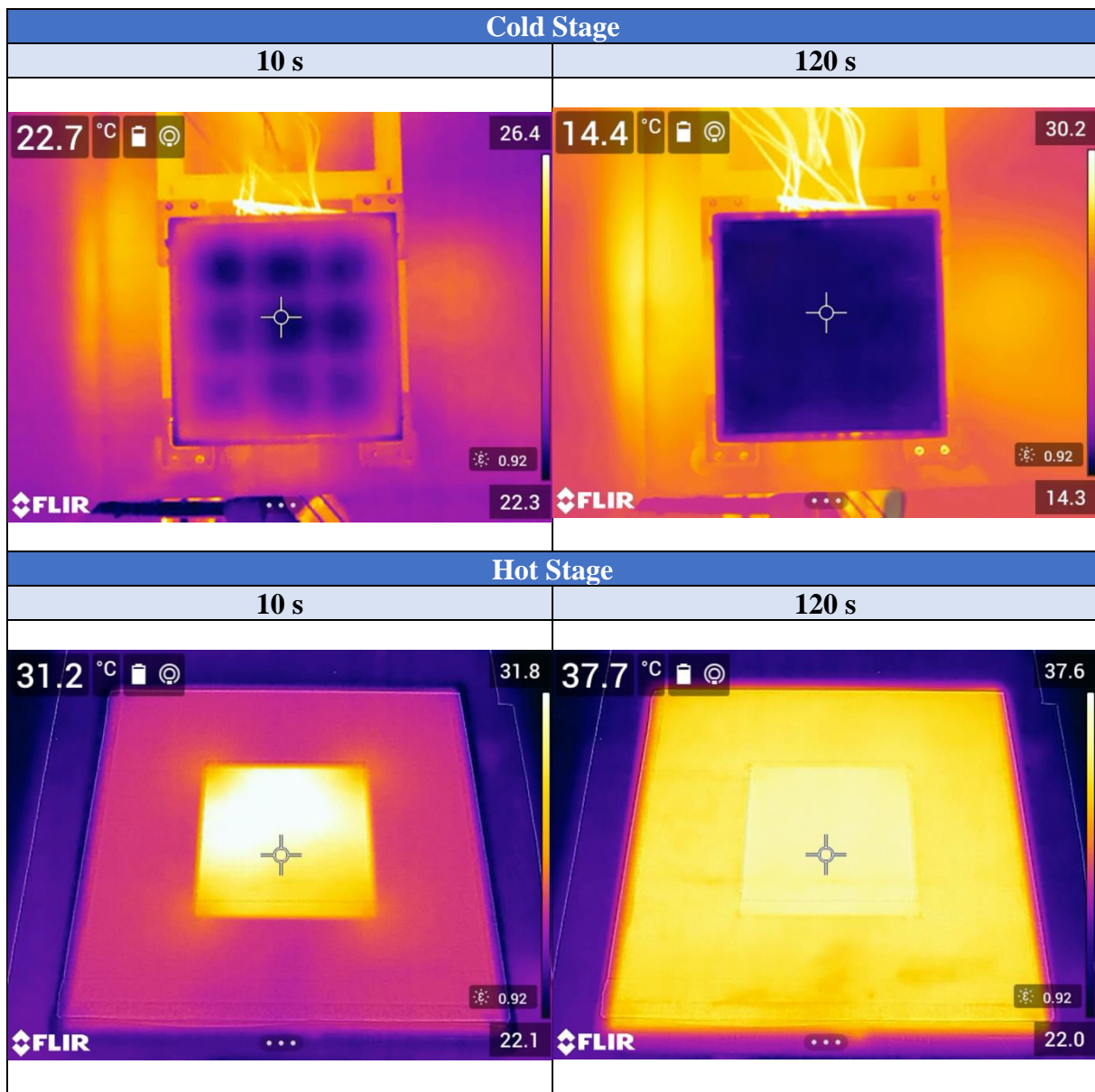


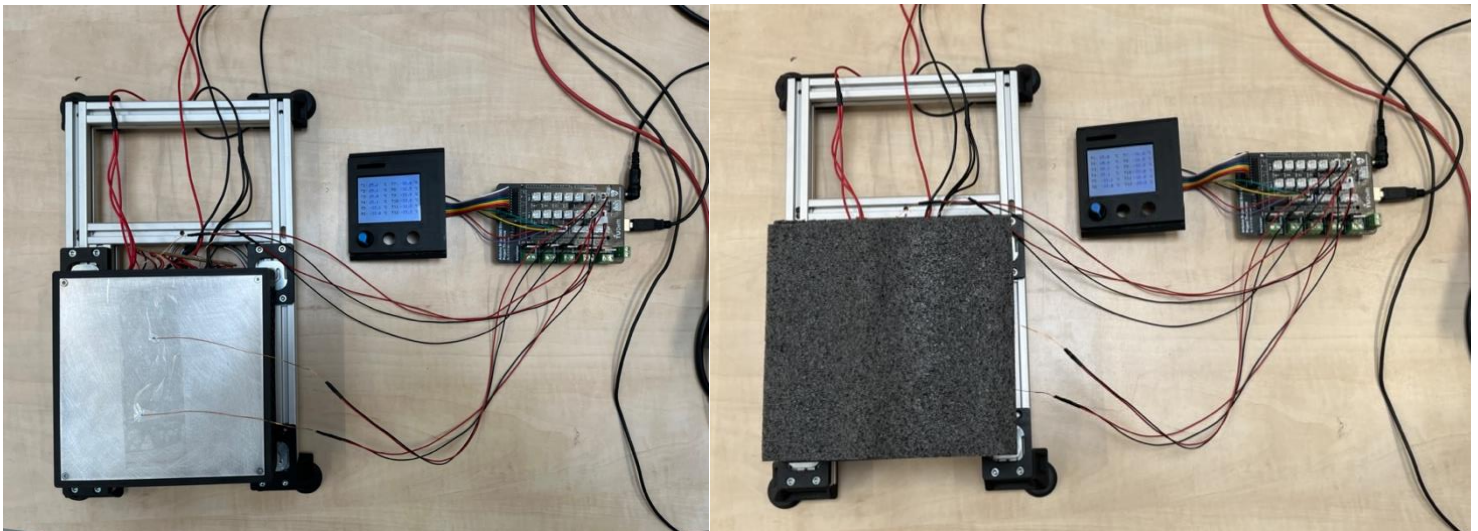
Table 21. Thermal Uniformity Evaluation Results for the Hot and Cold Stage

## Annex 19: Thermistor Dept Verification

To assess the accuracy of the embedded thermistors in measuring the surface temperature of the plates, a comparative test was conducted. The test configuration, as illustrated in Figure 33, included the Cold Stage with its two embedded thermistors, two reference thermistors taped to the Cold Stage's surface plate, an Arduino Mega equipped with the custom PCB shield for data processing, and a foam cover plate to minimize heat transfer to the reference thermistors from the surrounding environment.

During the experiment, the cold plate was activated, and temperature readings from all four thermistors were recorded at 60-second intervals for a total duration of 300 seconds. Each measurement cycle involved the rapid acquisition of four temperature readings within one second, which were subsequently averaged to obtain a single temperature reading. The averaged results are presented in Table 22 and Figure 34.

The obtained results indicated a strong agreement between the two sets of thermistors. The RMSE was calculated between the averages of the embedded thermistors and the surface thermistors over the entire data range, yielding an RMSE value of 0.0928 °C. This low RMSE value suggests a close correspondence between the embedded thermistors and the surface thermistors, confirming their accuracy in measuring the surface temperature of the plates.



**Figure 33: Thermistor Dept Verification Test Setup**

**Left:** Test set up featuring taped on thermistors.

**Right:** Test setup in operation, equipped with a foam insulation cover to minimize external influences.

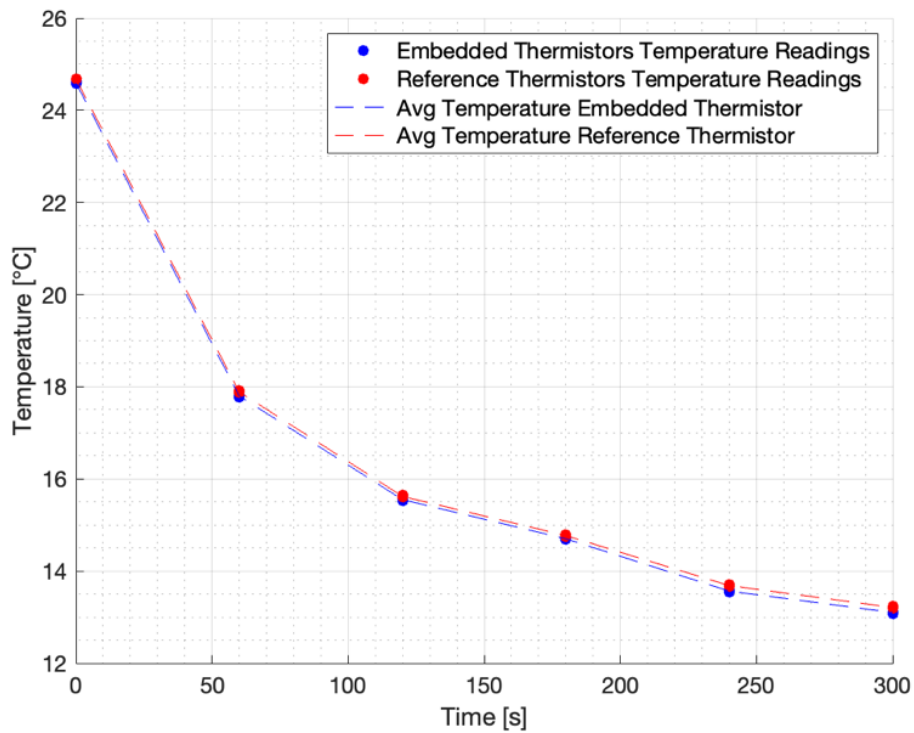
Thermistor Location Validation							
Time[s]	T1 [ °C]	T2 [ °C]	Tavg [ °C]	Tr1 [ °C]	Tr2 [ °C]	T <sub>r</sub> Avg [ °C]	Abs. Diff. [ °C]
0	24,62	24,57	24,59	24,68	24,68	24,68	0,09
60	17,78	17,82	17,80	17,87	17,91	17,89	0,09
120	15,58	15,52	15,55	15,60	15,64	15,62	0,07
180	14,68	14,72	14,70	14,75	14,79	14,77	0,07
240	13,55	13,58	13,57	13,67	13,70	13,69	0,12
300	13,08	13,12	13,10	13,19	13,23	13,21	0,11
<b>RMSE of embedded thermistors Tavg compared to reference readings T<sub>r</sub>avg [ °C]:</b>							0,0928

**Table 22. Thermistor Dept Verification Test Results**

*T1 and T2 are the average readings of 4 successive temperature readings of the embedded thermistors, Tavg is the average of the T1 and T2 readings.*

*Tr1 and Tr2 are the average readings of 4 successive temperature readings of the surface thermistors, T<sub>r</sub> Avg is the average of th the Tr1 and Tr2 readings.*

*Abs.Diff. is the absolute difference between the average embedded temperature readings and the average reference temperature readings.*



**Figure 34: Comparative Temperature Analysis between Embedded Thermistor Readings and Surface Thermistors Readings**

*For each time increment, four averaged temperatures are presented, but the data points are so closely spaced that they are hardly distinguishable.*



## Annex 20: Therminus-K3 Measurement Procedure

To take thermal conductivity measurements with the Therminus-K3 prototype, please follow this measurement procedure:

1. **Setup:** Position the Therminus-K3 Prototype on a stable surface. Adjust the adjustable feet as necessary to ensure stability.
2. **Preparation:** Use the quick release mechanism to disengage the linear carriage from the leadscrew. Lower the Hot Stage onto the Cold Stage.
3. **Power Connection:** Connect the C19 Power cable to a 220V AC outlet.
4. **Data Acquisition:** Connect the USB cable connected to the Arduino Mega to your laptop. Use a data acquisition program like CoolTerm [36] to read and store serial port data from the Arduino Mega. Ensure your chosen data acquisition program is running before starting a measurement on the Therminus-K3 prototype.
5. **Powering On:** Turn on the machine using the switch located on the back.
6. **Initialization:** Wait for the start-up screen to disappear.
7. **Temperature Setting - Hot Stage:** Set the Hot Stage temperature, indicated by 'Th' Confirm the setting by pressing the rotary encoder.
8. **Temperature Setting - Cold Stage:** Set the Cold Stage temperature, indicated by 'Tc' Confirm the setting by pressing the rotary encoder.
9. **Thickness Calibration:** Press the red pushbutton to tare the thickness measurement. Wait until the 'Sample Thickness' reading shows 0.00 mm.
10. **Sample Placement:** Raise the Hot Stage and engage the leadscrew by activating the quick release mechanism. Place the sample over the Cold Stage, smoothing out any wrinkles.
11. **Pressure Calibration:** Press the yellow pushbutton to zero the pressure measurement. Wait until the 'Sample Pressure' reading shows 0 Pa.
12. **Measurement Initiation:** Disengage the leadscrew and lower the Hot Stage onto the sample for measurement under the standard deadweight pressure. Alternatively, if a specific pressure is required, re-engage the leadscrew after placing the sample. Carefully adjust the handwheel in a clockwise direction to apply the desired pressure. Monitor and verify the applied pressure on the user interface screen.
13. **Start Measurement:** Begin the measurement by pressing the rotary encoder.
14. **Measurement Interruption:** If necessary, cancel the measurement by pressing the red button. You can now restart a measurement by repeating steps 7 to 13.
15. **Shutdown:** Before turning off the Therminus-K3 prototype with the rear switch, ensure no measurement is in progress. Either cancel the ongoing measurement or wait until it has finished.

## Annex 21: Performance Validation Measurements

Performance validation measurements were conducted with two distinct materials: a silicone rubber sheet and a 100% cotton fabric coat. The silicone rubber sheet, sourced from a baking sheet [37], was cut to match the dimensions of the measurement area. In contrast, the cotton coat was measured non-destructively, aligning with the prototype's design requirements. Table 23 provides detailed characteristics of each sample and depicts the measurement methodology for both materials. Each material was measured 7 times, following the complete measurement procedure as outlined in Annex 20. The thermal conductivity was then calculated using the custom post MATLAB processing script as detailed in Annex 28. Finally, the standard deviation of the resulting thermal conductivity measurement values of each material was calculated using the following formula:

$$\text{Standard Deviation (SD)} = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$$

In this formula,  $x_i$  represents each individual measurement,  $\bar{x}$  is the mean of the measurements, and  $n$  is the number of measurements (7 in this case). The denominator  $n-1$  (instead of  $n$ ) is used as part of Bessel's correction [38], an adjustment to provide an unbiased estimate of the population standard deviation when dealing with small sample sizes. With the calculated standard deviation and the mean of the 7 measurements, the percentile precision of the measurement was then determined as follows:

$$\text{Precision \%} = \left( \frac{SD}{\text{Mean}} \right) \cdot 100$$

The results for the silicon rubber sheet measurements are shown in Table 24, with the corresponding standard deviation, mean and percentile precision. Table 25 shows the results for the cotton fabric measurements.

Performance Validation Measurement Details	
Destructive Measurement	Silicon Rubber Sample Details [37]
	
Non-destructive Measurement	Cotton Sample Details [39]
	

Table 23. Performance Validation Measurements and Sample Details

Silicon Rubber Measurement Results							
Material:	Thickness: [mm]	Pressure: [Pa]	dT: [ °C]	Tmean: [ °C]	Time to SS [s]	Measurement Duration [min]	k: [W/mk]
100% Silicone Rubber Sheet	1,76	718	10,57	29,71	388,00	25	0,2445
	1,76	714	10,47	29,77	344,00	25	0,2423
	1,76	715	10,49	29,75	388,00	25	0,2431
	1,76	705	10,34	29,83	390,00	25	0,2455
	1,76	707	10,43	29,79	369,50	25	0,2434
	1,76	702	10,40	29,80	190,00	25	0,2442
	1,76	721	10,69	29,75	287,50	25	0,2431
<b>Mean:</b>	<b>1,76</b>	<b>712</b>	<b>10,48</b>	<b>29,77</b>	<b>336,71</b>	<b>25</b>	<b>0,2437</b>
<b>STD:</b>							<b>0,001072</b>
<b>Precision %</b>							<b>0,44</b>

Table 24. Performance Validation Measurements Results Silicon Rubber

Cotton Measurement Results							
Material:	Thickness: [mm]	Pressure: [Pa]	dT: [ °C]	Tmean: [ °C]	Time to SS [s]	Measurement Duration [min]	k: [W/mk]
100% Cotton Fabric	0,52	708	12,20	28,90	325,50	25	0,0446
	0,52	708	12,06	28,97	456,00	25	0,0451
	0,52	702	12,19	28,91	462,00	25	0,0451
	0,52	705	12,15	28,92	421,50	25	0,0449
	0,52	701	11,94	29,03	437,50	25	0,0448
	0,52	713	11,42	29,29	371,00	25	0,0456
	0,52	711	12,20	28,90	331,50	25	0,0448
<b>Mean:</b>	<b>0,52</b>	<b>707</b>	<b>12,02</b>	<b>28,99</b>	<b>400,71</b>	<b>25</b>	<b>0,0450</b>
<b>STD:</b>							<b>0,0003237</b>
<b>Precision %</b>							<b>0,72</b>

Table 25. Performance Validation Measurements Results 100% Cotton Fabric

## Annex 22: Therminus-K3 Measurement Uncertainty:

To calculate the measurement uncertainty  $u_k$  of the Therminus-K3 prototype the law of error propagation was used, which is described as:

$$u_k = k \cdot \left[ \left(\frac{u_V}{V}\right)^2 + \left(\frac{u_I}{I}\right)^2 + \left(\frac{u_{LS}}{LS}\right)^2 + \left(\frac{u_L}{L}\right)^2 + \left(\frac{u_W}{W}\right)^2 + \left(\frac{u_{\Delta T}}{\Delta T}\right)^2 \right]^{\frac{1}{2}}$$

Where  $u_V$ ,  $u_I$ ,  $u_{LS}$ ,  $u_L$ ,  $u_W$ , and  $u_{\Delta T}$  are the specific measurement uncertainties related to voltage, current, sample thickness, measurement area (length and width), material thickness, and temperature difference, respectively. The corresponding measurement estimate values for these parameters are denoted by V, I, Ls, L, W, and  $\Delta T$ . The squared terms are known as the measurement variances, which are calculated by squaring the relative uncertainty, which is the ratio between the specific measurement uncertainty and its corresponding estimated measurement value. By summing these variances, the formula accounts for the combined effect of uncertainties from all measurements. The square root of this sum then gives the overall relative uncertainty in the calculation of thermal conductivity. This approach is based on the assumption that the individual uncertainties are independent, meaning the error in one measurement does not affect the errors in others. The specific measurement uncertainties are based on the maximum measurement resolution, or the maximum deviation found during the calibration of each sensor, depending on which is the limiting factor:

$$\begin{aligned} u_V &= \text{Resolution} \div 2 = 0.01 \div 2 = 0.005 \text{ V} \\ u_I &= \text{Max Calibration Deviation} \div 2 = 0.0008 \div 2 = 0.0004 \text{ A} \\ u_{LS} &= \text{Resolution} \div 2 = 0.0275 \div 2 = 0.01375 \text{ mm} \\ u_L &= \text{Resolution} \div 2 = 0.01 \div 2 = 0.005 \text{ mm} \\ u_W &= \text{Resolution} \div 2 = 0.01 \div 2 = 0.005 \text{ mm} \\ u_{\Delta T} &= \text{Resolution} \div 2 = 0.087 \div 2 = 0.0435 \text{ [}^\circ\text{C]} \end{aligned}$$

To understand how much each measurement's uncertainty contributed to the total measurement uncertainty, the percentile contribution of each uncertainty was calculated as well. This is done by dividing the individual variance by the sum of all variances and multiplying by 100%. Finally, the measurement uncertainty of the Therminus-K3 prototype can be converted to a percentile uncertainty by dividing it by the estimated thermal conductivity value and multiplying it by 100%. The results of these calculations are summarized in Table 26. The

variance contributed by the thickness measurement uncertainty was the largest, accounting for 88.8%, which significantly influenced the total uncertainty. Therefore, the calculations were repeated, but this time for the thermal resistance, which does not depend on the thickness measurement. The thermal resistance (R) is calculated as follows:

$$R = \frac{A\Delta T}{\dot{Q}}$$

The approach for the subsequent calculations remains unchanged. As indicated in Table 26, utilizing thermal resistance instead of thermal conductivity significantly lowers the measurement uncertainty.

Measurement Uncertainty Thermal Conductivity					
Source $X_i$	Units	Measurement Estimate $x$	Uncertainty $u(x)$	Variance $(u(x)/x)^2$	Contributed Variance %
Voltage	[V]	12.00	$\pm 5.000E-03$	1.7361e-07	0.08
Current	[A]	0.1845	$\pm 4.000E-04$	4.7003e-06	2.21
Thickness	[m]	0.001	$\pm 1.375E-05$	1.8906e-04	88.82
Length	[m]	0.07018	$\pm 5.000E-06$	5.0788e-09	0.002
Width	[m]	0.07016	$\pm 5.000E-06$	5.0759e-09	0.002
Temperature	[°C]	10.00	$\pm 4.350E-02$	1.8922e-05	8.89
Property	Units	Measurement Estimate	Uncertainty	Uncertainty in %	
Thermal Conductivity k	[W/mK]	<b>0.0450</b>	$\pm 6,560E-04$	$\pm 1,46$	
Thermal Resistance R	[K/W]	<b>0.0222</b>	$\pm 1,085E-04$	$\pm 0,49$	

Table 26. Measurement Uncertainty Analysis for a thermal conductivity of 0.0450 W/mk and a Thermal resistance of 0.0222 K/W



## Annex 23: PHOEBE Model Simulations

To assess the impact of the Terminus-K3 prototype on PMI estimations, body cooling simulations were conducted using the PHOEBE model. The simulations compared a standard thermal conductivity value for cotton ( $k = 0.04 \text{ W/mK}$ ) with a more precise value obtained using the Terminus-K3 prototype ( $k = 0.0451 \text{ W/mK}$ ). The aim was to demonstrate how a minor variation in fabric thermal conductivity can influence the body cooling process, thereby highlighting the sensitivity of the PMI estimation process to changes in fabric thermal conductivity. In practical forensic scenarios, the PHOEBE model would use skin thermometry values as input parameters to reconstruct PMI. However, these simulations focused on how the body cooling curves are influenced by the input thermal conductivity values, so the simulation method was set in 'Forward' mode in which it calculates the cooling curves for a set period of time. That period was set to 48h, with a time step size of 2 minutes. The standard body dimension settings remained unchanged, while the weight was set to 85Kg, and the environment and floor temperature was set to  $5^{\circ}\text{C}$ . The body was simulated to wear trousers, a shirt and a jumper/jacket and it was simulated as it was laying on the floor with its back touching the ground. Figure 35- 39 outline the PHOEBE model settings, and Figure 40 displays the enlarged cooling curves. These temperature curves demonstrate the cooling of the rectal core temperature, allowing for PMI estimation by correlating a specific core temperature to a corresponding time value, as is done with the horizontal temperature lines in Figure 40.

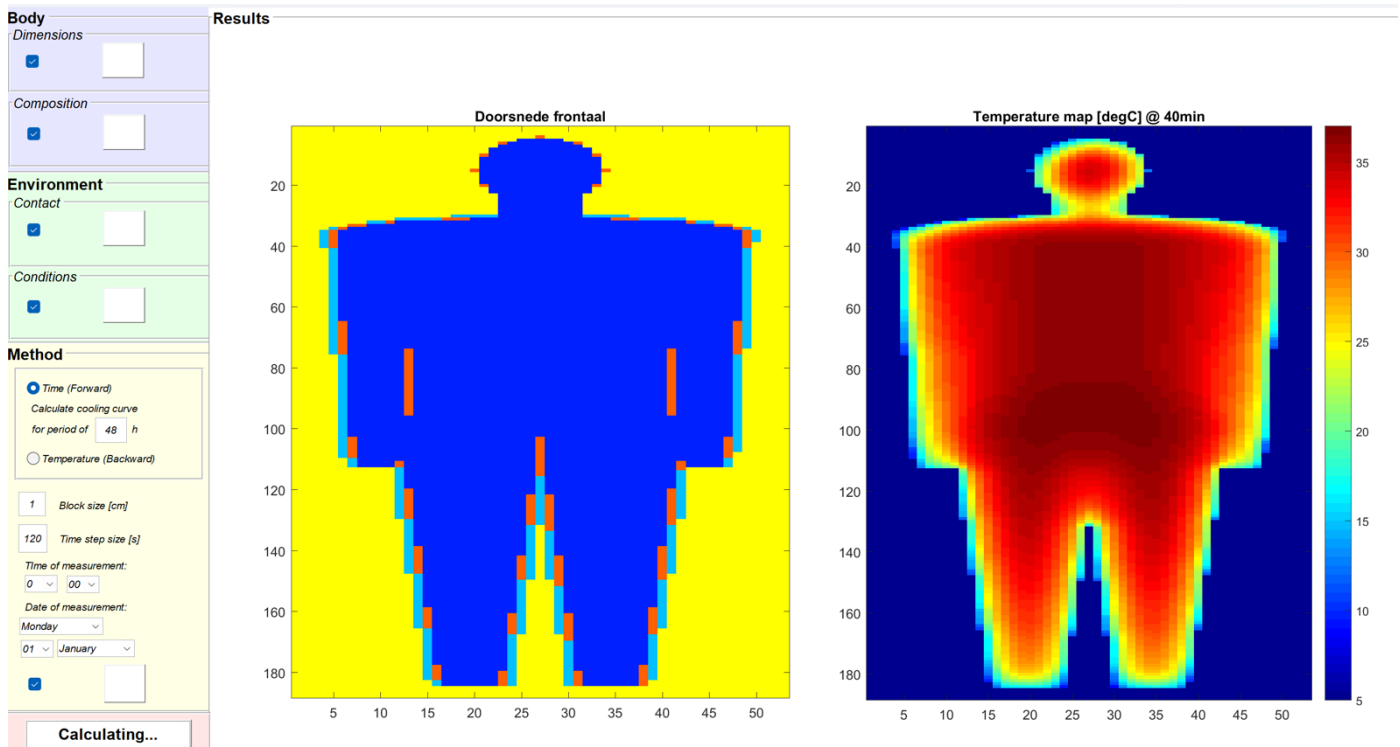


Figure 35: PHOEBE Main Screen Settings

### Body Dimensions in cm

-Total body length-	180	-Total body width-	45	-Total body depth-	15
-Head length-	22	-Head width-	14	-Head depth-	17
- Neck length-	9	-Neck circumference-	31	-Breast protrusion-	0
-Torso width-	30	-Torso circumference-	90	-Stomach protrusion-	0
-Arm length-	77	-Shoulder circumference-	30	-Wrist circumference-	16
-Leg length-	88	-Thigh circumference-	53	-Ankle circumference-	22

**Figure 36: PHOEBE Model Body Dimensions Settings**

### Body Composition

Weight	85	Body thermal conductivity [W/(mK)]	0.55
Body fat percentage	15	Fat thermal conductivity [W/(mK)]	0.2
Amount of blood loss	0	Clothing thermal conductivity [W/(mK)]	0.04
Duration of blood loss	--	Thickness of clothing [cm]	0.2
Emissivity	0.96	Clothes specific heat [J/cm3]	4
		Floor/walls thermal conductivity [W/(mK)]	3
		Floor specific heat [J/cm3]	3.68

**Figure 37: PHOEBE Model Body Composition Settings**

### Environmental Contact

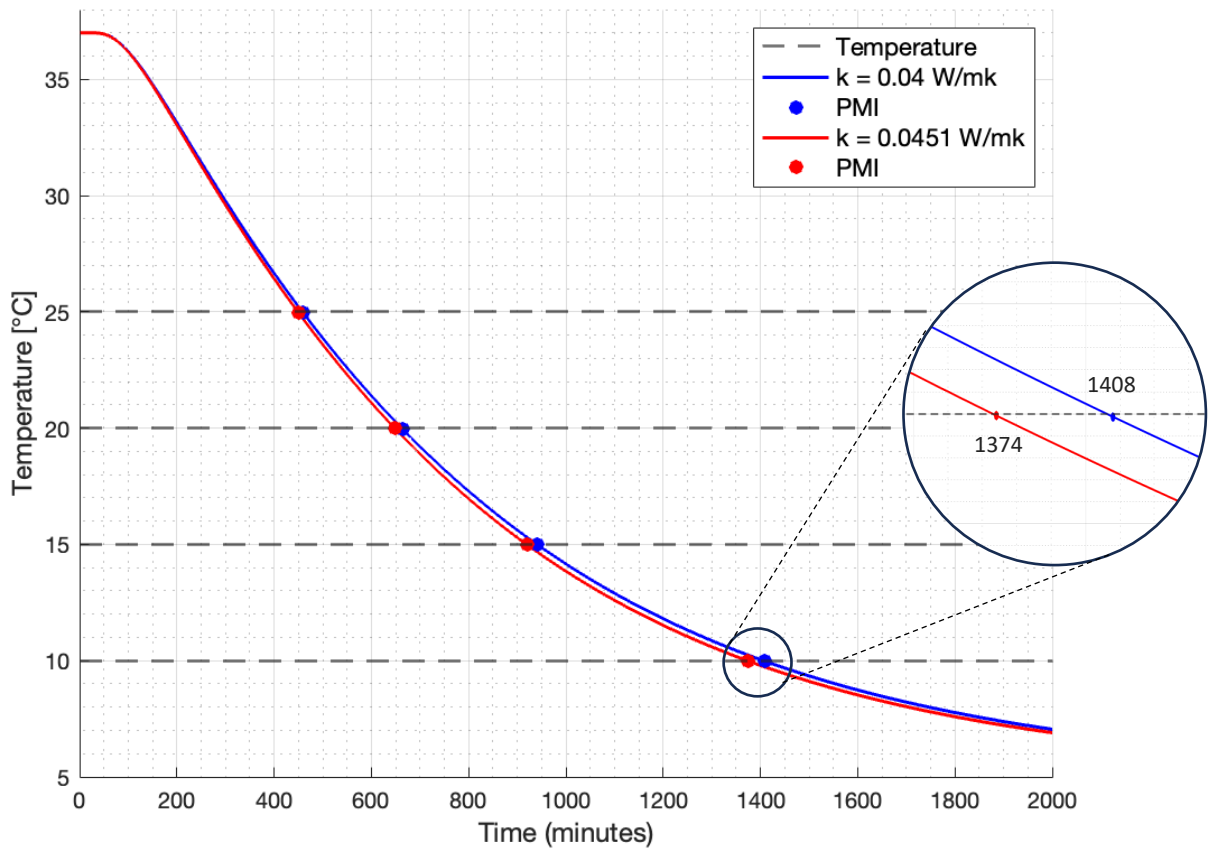
<p style="text-align: center;"><i>Clothing</i></p> <p><input type="checkbox"/> Naked</p> <p><input checked="" type="checkbox"/> Trousers</p> <p><input type="checkbox"/> Shorts</p> <p><input checked="" type="checkbox"/> Shirt</p> <p><input checked="" type="checkbox"/> Jumper/Jacket</p> <p><input type="checkbox"/> Covered head <small>(helmet, hat...)</small></p> <p><input type="checkbox"/> Covered face <small>(Facial hair, mask...)</small></p>	<p style="text-align: center;"><i>Contact with floor/walls</i></p> <p><input checked="" type="checkbox"/> Back</p> <p><input type="checkbox"/> Front</p> <p><input type="checkbox"/> Left</p> <p><input type="checkbox"/> Right</p> <p><input type="checkbox"/> Feet</p> <p><input type="checkbox"/> Head</p>	<p style="text-align: center;"><i>Contact with water</i></p> <p><input type="checkbox"/> Front</p> <p><input type="checkbox"/> Back</p> <p><input type="checkbox"/> Whole body</p>	<p style="text-align: center;"><i>Exposed to radiation</i></p> <p><input type="checkbox"/> Back</p> <p><input type="checkbox"/> Front</p> <p><input type="checkbox"/> Left</p> <p><input type="checkbox"/> Right</p> <p style="text-align: center; margin-top: 20px;"><b>Not possible yet</b></p>
---	---	--	---

**Figure 38: PHOEBE Model Environmental Contact Settings**

### Environmental Conditions

<p><i>Initial body temperature [degC]</i> <input style="width: 50px;" type="text" value="37"/></p> <p><i>Mean ambient temperature (AC)</i> <input style="width: 50px;" type="text" value="5"/></p> <p><i>Deviation from mean (AC)</i> <input style="width: 50px;" type="text" value="0"/></p> <p><i>Period of temperature variation (AC)</i> <input style="width: 50px;" type="text" value="0"/></p> <p><i>Phase of temperature variation (AC)</i> <input style="width: 100px;" type="text" value="Increasing"/></p> <p><input type="checkbox"/> <i>Shirt removed after some time?</i></p> <p><i>Hours until shirt removal:</i> <input style="width: 50px;" type="text" value="--"/></p>	<p><i>Temperature jump</i> <input style="width: 50px;" type="text" value="--"/></p> <p><i>Time of temperature jump</i> <input style="width: 50px;" type="text" value="--"/></p> <p><input type="checkbox"/> <i>AMC morgue?</i></p> <p><i>Water temperature [degC]</i> <input style="width: 50px;" type="text" value="--"/></p> <p><i>Floor temperature [degC]</i> <input style="width: 50px;" type="text" value="5"/></p> <p><i>Wind velocity [m/s]</i> <input style="width: 50px;" type="text" value="0"/></p> <p><i>Water velocity [m/s]</i> <input style="width: 50px;" type="text" value="0"/></p>
--	--

**Figure 39: PHOEBE Model Environmental Conditions Settings**



**Figure 40: PHOEBE Simulation Rectal Core Cooling Curves for Different Clothing Thermal Conductivity Values**

Intersection points of the cooling curves with the horizontal temperature lines estimate imaginary PMIs. Difference in those PMIs show the potential improvement in PMI estimation through the use of the Thermins-K3 prototype, results are summarized in Table 27.

PMI Estimations			
Temp.	PMI for $k = 0.0400$ W/mk	PMI for $k = 0.0451$ W/mk	$\Delta$ PMI
25 °C	460 min	450 min	10 min
20 °C	664 min	648 min	16 min
15 °C	942 min	920 min	22 min
10 °C	1408 min	1374 min	34 min

**Table 27. Potential PMI Estimation Improvements**

## Annex 24: Therminus-K3 Design Criteria Evaluation

The evaluation of the Therminus-K3 prototype involved assessing its conformity to the requirements as detailed in Annex 1. Table 28 provides an evaluation of the prototype's performance requirements, while Table 29 focuses on its compliance with the general requirements. In these tables, the targets are color-coded: green indicates a high degree of alignment with the specific targets, while red signifies areas where the prototype's performance did not fully meet the set objectives.

Performance Requirements:		
Requirement:	Target:	Achieved:
Accuracy error	< 3 %	Preliminary tests suggest a thermal conductivity accuracy of approximately 1.5%, but further testing is required to determine the true accuracy
Precision error	< 1 %	All the preliminary test showed a measurement precision below 1%
Measurement Range	0.020 - 0.700 [W/mk]	Measurement range was not put to the test in the preliminary tests, but the simulation results in Annex 5 and independent sensor and heating capabilities suggest that achieving the target measurement range should not be an issue
Variable Pressure	0 – 10000 [Pa]	Pressures over 10.000 Pa could be achieved
Sample thickness range	0.44 – 28 [mm]	Due to the non-linearity of the slide potentiometer, accurate thickness measurements were limited to 20 mm
Sample dimensions	Non-destructive	Therminus-K3 can make non-destructive measurements
Setup time	< 5 minutes	Setup time of prototype is well below 5 minutes due to its lightweight, portable design and simplistic user interface
Measurement time	< 30 minutes	All preliminary measurements took only 25 minutes
Lifetime recalibration without	≥ 12 months	The Therminus-K3 prototype uses the absolute GHP method, which basically eliminates the need for frequent recalibration. However, the thermistors may experience calibration drift due to aging, but the manufacturer indicates excellent long-term stability, thus it is assumed that the recalibration period exceeds 12 months

Table 28. Performance Requirements Evaluation

General Requirements:		
Requirement:	Target:	Achieved:
Shielding	Measurement should be shielded from wind, water, and temperature fluctuations	The side guard heater simulations in Annex 5 suggest that the prototype can achieve steady state conditions under the metering area in the worst conditions, but this is not verified yet.
Decontamination	Resistant to alcohol-based cleaning up to 5 times a day	The surface plates are made from durable aluminium which is resistant to alcohol-base cleaning solutions
Portability	Volume: 330 x 330 x 330 [mm] Weight: < 8 [kg]	Therminus-K3 meets both the volume and weight requirements
Durability	The enclosed device must pass rough handling and repetitive shock tests: NEN-EN-IEC 60068-2-27:2008	The Therminus-K3 incorporates a sturdy aluminium frame and shield the electronics well but no rough handling test were preformed
Simplicity	The device should have a user-friendly and intuitive interface	The prototype boasts a simplistic user interface, but its real user-friendliness and intuitiveness still needs to be determined through user tests.
Power Source	The device should operate using either battery power or a car battery connection	The device currently still works on a 12V power supply, but it is designed so that it can easily be swapped out for a 12V battery pack
Reporting Protocol	The device should provide both raw measurement data files and PDF reports summarizing measurements and outcomes	The prototype only provides raw measurement data, and the conversion of this raw data into a thermal conductivity value requires another computer running MATLAB
Cost	≤ € 1.945	With a material cost below the € 405 the expected production cost and commercial price will be well below the target price

Table 29. General Requirements Evaluation



## Annex 25: Compression Force Measurement Drift

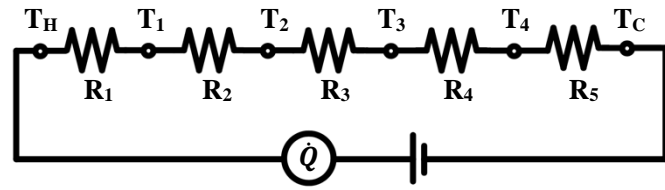
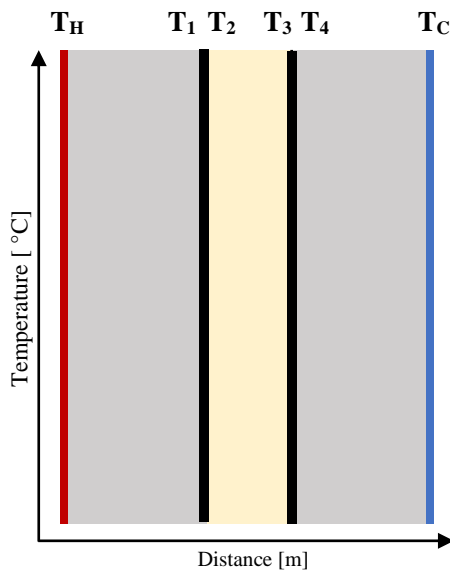
A test was conducted to evaluate the decrease in pressure as observed in the compression stage readings. The handwheel was adjusted to achieve an initial pressure at around 75% of the maximum pressure of 10.000 Pa. The pressure was then monitored at 5-minute intervals for a total duration of 25 minutes. The results, presented in Table 28, indicate that the pressure decreased by no more than 3% during a complete thermal conductivity measurement duration of 25 minutes.

Pressure Decrease Test Results		
Time: [min]	Pressure: [Pa]	% Decrease
0	7544	0
5	7414	1,72
10	7387	0,36
15	7353	0,46
20	7343	0,14
25	7330	0,18
<b>Total Decrease in Pressure:</b>		<b>2,84%</b>

Table 30. Pressure Decrease Test Results

## Annex 26: Paint Layer – Temperature Gradient Calculations

To increase the emissivity of the surface plates, technical standards suggest using a high-emissive paint. However, this paint layer could have a negative effect on the accuracy of the surface temperature readings from the embedded thermistors. Similar to Annex 4, the temperature differential over the paint layers can be calculated using an electrical circuit analogy. This temperature differential should be smaller than the measurement resolution of the thermistors to ensure accurate surface temperature readings. Figure 27 details the electrical circuit analogy, and table 29 shows the actual temperature differences between each section calculated for the thermal conductivity measurement limits from the requirement, as well as an additional expected thermal conductivity value corresponding to the approximate thermal conductivity of clothing. In these calculations,  $\Delta T$  is set to 10 K, and A is set to 0.024964 m<sup>2</sup>, representing the temperature differential between the plates and the surface area of the plates of the Terminus-K3 prototype, respectively. Additionally, the paint layer thickness is set to 0.1 mm with a thermal conductivity of 0.254 W/mK, according to [40].



$$R_1 = \frac{L_1}{k_1 A}, R_2 = \frac{L_2}{k_2 A}, R_3 = \frac{L_3}{k_3 A}, R_4 = \frac{L_4}{k_4 A}, R_5 = \frac{L_5}{k_5 A}$$

$$\dot{Q} = \frac{\Delta T}{R_1 + R_2 + R_3 + R_4 + R_5} = \frac{T_H - T_C}{\frac{L_1}{k_1 A} + \frac{L_2}{k_2 A} + \frac{L_3}{k_3 A} + \frac{L_4}{k_4 A} + \frac{L_5}{k_5 A}}$$

$$\dot{Q} = \frac{T_H - T_1}{\frac{L_1}{k_1 A}} = \frac{T_1 - T_2}{\frac{L_2}{k_2 A}} = \frac{T_2 - T_3}{\frac{L_3}{k_3 A}} = \frac{T_3 - T_4}{\frac{L_4}{k_4 A}} = \frac{T_4 - T_C}{\frac{L_5}{k_5 A}}$$

Figure 27: Thermal Circuit Paint Layer Influence using Electrical Analogy

Left: Temperature distribution in a surface plate-paint layer-sample-paint layer-surface plate assembly.

Right: Electrical circuit analogy used to calculate total heat flow and temperature gradients over each segment.

Thermal Differential Calculations Paint Layer:									
	Upper limit: 0,70 W/mK			Expected values: 0,04 W/mk			Lower limit: 0,020 W/mk		
Location:	L [m]:	K [W/mk]:	R[K/W]:	L [m]:	K [W/mk]:	R[K/W]:	L [m]:	K [W/mk]:	R[K/W]:
I	4,00E-03	204	7,85E-04	4,00E-03	204	7,85E-04	4,00E-03	204	7,85E-04
II	1,00E-04	0,254	1,58E-02	1,00E-04	0,254	1,58E-02	1,00E-04	0,254	1,58E-02
III	4,40E-04	0,022	5,10E+01	1,00E-03	0,05	8,01E-01	2,80E-02	0,022	5,10E+01
IV	1,00E-04	0,254	1,58E-02	1,00E-04	0,254	1,58E-02	1,00E-04	0,254	1,58E-02
V	4,00E-03	204	7,85E-04	4,00E-03	204	7,85E-04	4,00E-03	204	7,85E-04
Heat flow		233,79	[W]	Heat flow	12,21	[W]	Heat flow	0,20	[W]
Location:	Distance:	T [°C]:	dT[°C]:	Distance	T [°C]:	dT[°C]:	Distance	T [°C]:	dT[°C]:
TH	0,000	30,000	0,00E+00	0,000	30,000	0,00E+00	0,000	30,000	0,00E+00
T2	4,000	29,865	1,35E-01	4,000	29,991	9,41E-03	4,000	30,000	1,54E-04
T3	4,100	27,160	2,71E+00	4,100	29,802	1,89E-01	4,100	29,997	3,09E-03
T4	4,540	22,840	4,32E+00	5,100	20,198	9,60E+00	32,100	20,003	9,99E+00
T5	4,640	20,135	2,71E+00	5,200	20,009	1,89E-01	32,200	20,000	3,09E-03
TC	8,640	20,000	1,35E-01	9,200	20,000	9,41E-03	36,200	20,000	1,54E-04

Table 31. Temperature Differential Calculations for Thermistors at a dept of 0.75

Left: Temperature differential calculations for upper thermal conductivity measurement limit, thickness assumed to be minimum at 0.44 mm. Middle: Temperature differential calculations for estimated thermal conductivity, sample thickness assumed to be 1 mm Right: Temperature differential calculations for lower thermal conductivity limit, sample thickness assumed to be maximum at 28 mm

Only at the lower thermal conductivity level is the temperature differential over the paint layer smaller than the measurement resolution of the embedded thermistors, which is 0.087 °C. At an estimated clothing thermal conductivity of 0.04 W/mK, the temperature drop over the paint layer is more than twice as large as the thermistors' measurement resolution, which would result in significant errors in the thermal conductivity measurement. Therefore, the decision was made not to apply paint to the surface plates.

## Annex 27: Therminus-K3 Postprocessing Results

The custom post-processing MATLAB as presented in Annex 28 script for thermal conductivity calculation generates three graphs: one for raw and filtered temperature data (Figure 42), another illustrating power delivery to the main heater (Figure 43), and a third depicting the calculated thermal conductivity over the entire measurement duration (Figure 44). Additionally, the script also prints the most important measurement data, exemplified in Table 30 for a silicone rubber measurement.

--- Metadata for Silicone_Rubber_7.xlsx ---		--- Thermal Conductivity (k) in Steady State ---	
Measurement Duration:	25 min	Average k:	0.2431 W/(mK)
Tenvironment:	22.96°C	Maximum k:	0.2502 W/(mK)
Thot (Set):	35°C	Minimum k:	0.2325 W/(mK)
Tcold (Set):	15°C	--- Other Parameters in Steady State ---	
Applied Weight:	721 Pa	Average dT:	10.69°C
Sample Thickness:	1.79 mm	Average T:	29.65°C
		Average Q:	7.15 W
		Time to steady state:	287.50 s

Table 32. Measurement Data Printed by Postprocessing Script

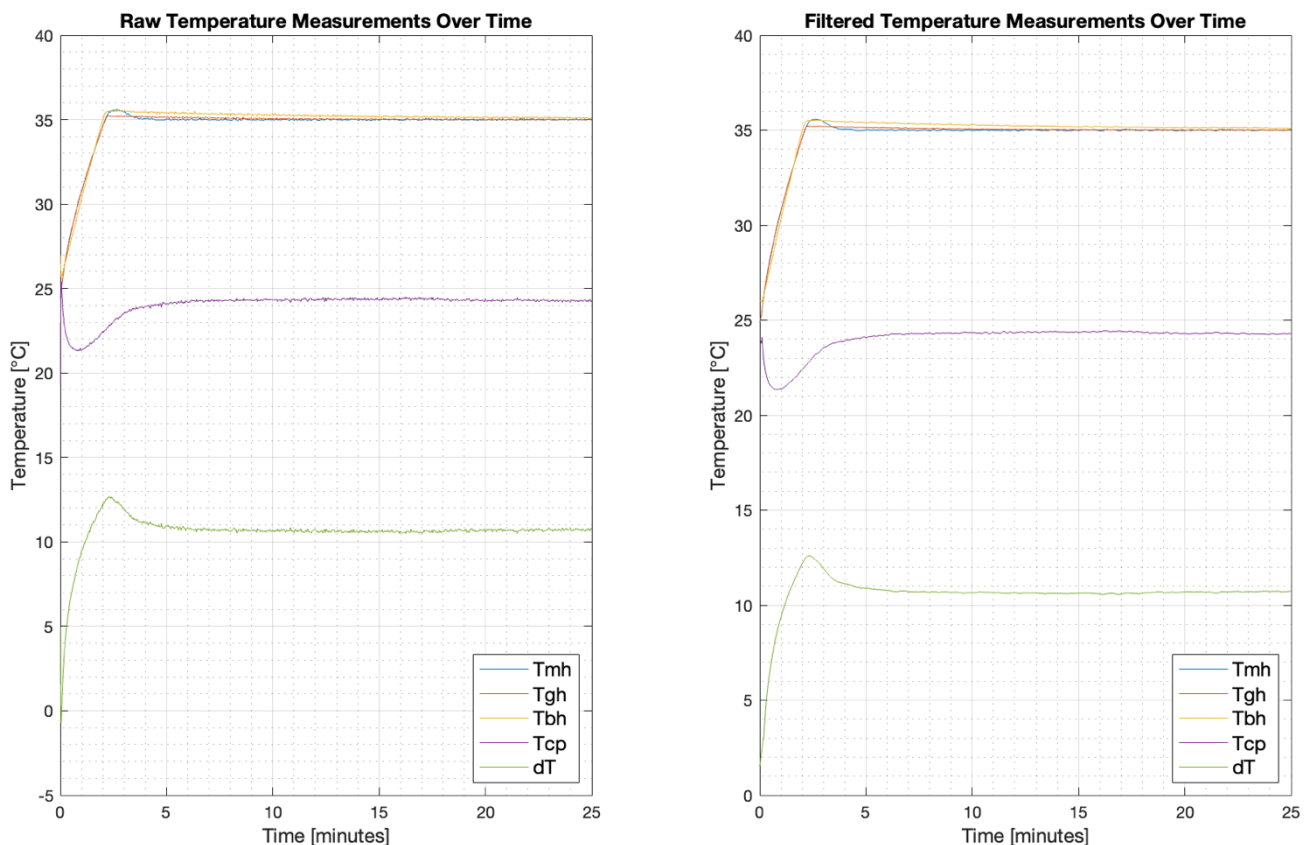
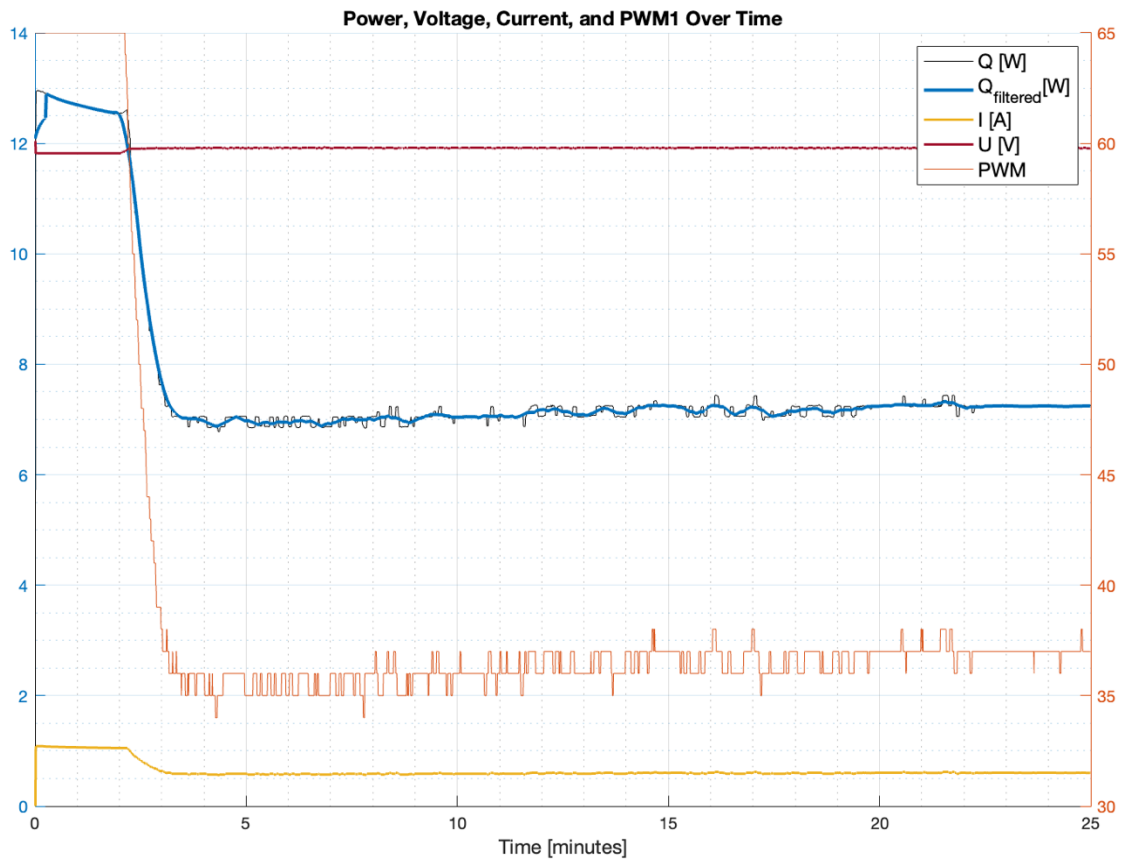
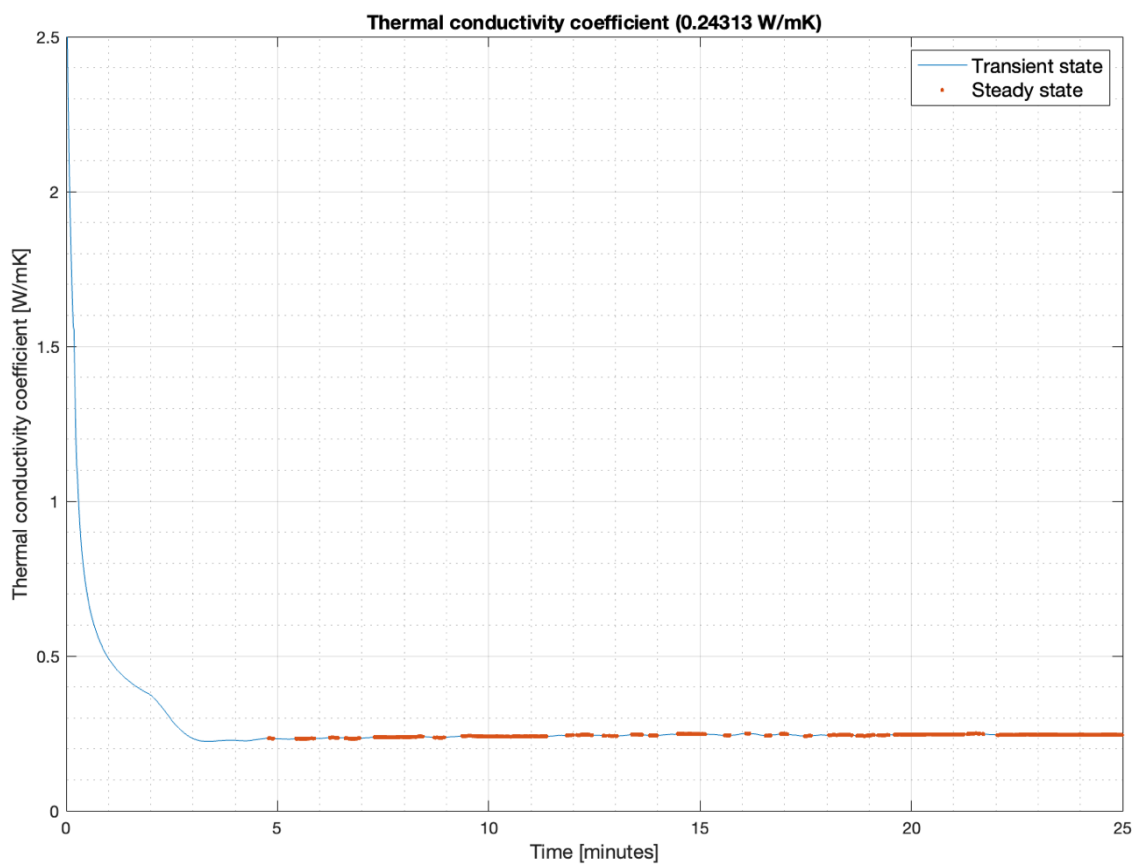


Figure 42: Temperature Data Thermal Conductivity Measurement



**Figure 43: Power Data Thermal Conductivity Measurement**



**Figure 44: Thermal Conductivity Measurement Graph**

## Annex 28: Thermanus-K3 Postprocessing MATLAB Script

```
clc;
clear;
close all;

%Define Constants and Read out data:
A = 70.18 * 70.16 / 1E6;

% Specify the Excel file
file = ['Silicone_Rubber_7.xlsx'];

%% Extract Metadata
[~, ~, raw_metadata] = xlsread(file, 'A1:B8');

% Extract metadata values
measurementDuration = str2double(raw_metadata{3, 2});
Thot_set = str2double(raw_metadata{5, 2});
Tcold_set = str2double(raw_metadata{6, 2});
AppliedWeight = raw_metadata{7, 2};
L = str2double(raw_metadata{8, 2});

%% Extract Data
% Read Data
D = readmatrix(file);
ts = D(:, 1);           % Time in [s]
t = ts/60;             % Time in [m]
Tmh = D(:, 14);        % Main heater temperature in [°C]
Tgh = D(:, 15);        % Guard heater temperature in [°C]
Tbh = D(:, 16);        % Back heater temperature in [°C]
Tcp = D(:, 17);        % Cold plate temperature in [°C]
Tenv = nanmean(D(:, 13)); % Environment temperature in [°C]
dT = Tmh - Tcp;        % Temperature difference in [°C]
PWM = D(:, 18);        % PWM signal to main heater
Q = (D(:, 22) / 1000); % Power in [Watt]
U = D(:, 23);          % Voltage in [V]
I = D(:, 24)/1000000;  % Current in [A]

%% Filter data:
F_Tmh = movmean(Tmh, [10,10], 'Endpoints', 'shrink');
F_Tgh = movmean(Tgh, [10,10], 'Endpoints', 'shrink');
F_Tbh = movmean(Tbh, [10,10], 'Endpoints', 'shrink');
F_Tcp = movmean(Tcp, [10,10], 'Endpoints', 'shrink');
F_dT = movmean(dT, [20,20], 'Endpoints', 'shrink');
F_Q = movmean(Q, [30,30], 'Endpoints', 'shrink');

% Plot Hot plate temperatures:
figure();
% Plot Unfiltered Data
subplot(1,2,1);
plot(t, Tmh, t, Tgh, t, Tbh, t, Tcp, t, dT);
grid on;
grid minor;
legend('Tmh', 'Tgh', 'Tbh', 'Tcp', 'dT', 'Location', 'southeast');
xlabel('Time [minutes]');
ylabel('Temperature');
title('Raw Temperature Measurements Over Time');

% Plot Filtered Data
subplot(1,2,2);
plot(t, F_Tmh, t, F_Tgh, t, F_Tbh, t, F_Tcp, t, F_dT);
grid on;
grid minor;
legend('Tmh', 'Tgh', 'Tbh', 'Tcp', 'dT', 'Location', 'southeast');
xlabel('Time [minutes]');
ylabel('Temperature');
title('Filtered Temperature Measurements Over Time');

% Calculate rate of change of F_dT and F_Q:
dF_dT = movmean(diff(F_dT), [30,30]);
dF_Q = movmean(diff(F_Q), [30,30]);

% Calculate temperature difference between mainheater and guardheaters:
Diff_Mh_Gh = F_Tmh - F_Tgh;
Diff_Mh_Bh = F_Tmh - F_Tbh;

% Find steady state conditions:
SS = find( (abs(dF_dT)<0.001) & (abs(dF_Q)<0.001) & (abs(Diff_Mh_Gh(2:end))<0.5) & ...
          (abs(Diff_Mh_Bh(2:end))<0.5));

%% Calculate Thermal Conductivity with Averaging Model:
```



```

k = (F_Q.* (L/1000))./(A.*(F_DT));
k_SS = k(SS);
t_SS = t(SS);

% DETERMINE K, Q AND dT IN STEADY STATE AND TIME TO REACH THAT
k_SS_mean = mean(k_SS); % aveage k in steady state
k_SS_max = max(k_SS); % maximum k in steady state
k_SS_min = min(k_SS); % minimum k in steady state
k_SS_std = std(k_SS); % std of k in steady state
dT_SS = mean(F_DT(SS)); % average dT in steady state
Q_SS = mean(F_Q(SS)); % average Q in steady state
t_SSs = ts(SS);
t_to_SS = t_SSs(1); % time to steady state [s]
Tavg = mean((Tmh(SS) + Tcp(SS)) / 2); % average temperature between hot can cold plate

%% Plot Thermal Conductivity:
figure();
plot(t,k);
hold on
if k_SS == 0;
else
plot(t_SS,k_SS,'.');
end
title("Thermal conductivity coefficient (" + k_SS_mean + " W/mK)")
xlabel('Time [minutes]');
ylabel('Thermal conductivity coefficient [W/mK]');
grid on; grid minor
ylim([0 2.5]);
legend('Transient state','Steady state');

% Plot P, U and I:
figure;
yyaxis left;
hold on
plot(t, Q, '-k')
plot(t, F_Q, '-', 'Color', [0 0.4470 0.7410], 'LineWidth', 2)
plot(t, I, '-', 'Color', [0.9290 0.6940 0.1250], 'LineWidth', 1.5);
plot(t, U, '-', 'Color', [0.6350 0.0780 0.1840], 'LineWidth', 1.5); % Added Voltage plot with a distinct
color
yyaxis right;
plot(t,PWM)
grid on;
grid minor;
legend('Q [W]', 'Q_{filtered}[W]', 'I [A]', 'U [V]', 'PWM', 'Location', 'northeast'); % Updated legend
xlabel('Time [minutes]');
title('Power, Voltage, Current, and PWM Over Time');

%% Print results:
fprintf('\n--- Metadata for %s ---\n', file);
fprintf('Measurement Duration: \t\t%s min\n', num2str(measurementDuration));
fprintf('Tenvironment: \t\t\t%.2f°C\n', Tenv);
fprintf('Thot (Set): \t\t\t%.2f°C\n', num2str(Thot_set));
fprintf('Tcold (Set): \t\t\t%.2f°C\n', num2str(Tcold_set));
fprintf('Applied Weight: \t\t\tPa\n', num2str(AppliedWeight));

fprintf('Sample Thickness: \t\t\tmm\n', num2str(L));
fprintf('\n');
fprintf('--- Thermal Conductivity (k) in Steady State ---\n');
fprintf('Average k: \t\t\t%.4f W/(m·K)\n', k_SS_mean);
fprintf('Maximum k: \t\t\t%.4f W/(m·K)\n', k_SS_max);
fprintf('Minimum k: \t\t\t%.4f W/(m·K)\n', k_SS_min);
fprintf('Standard Deviation (k): \t\t%.4f W/(m·K)\n', k_SS_std);

fprintf('\n');
fprintf('--- Other Parameters in Steady State ---\n');
fprintf('Average dT: \t\t\t%.2f°C\n', dT_SS);
fprintf('Average T: \t\t\t%.2f°C\n', Tavg);
fprintf('Average Q: \t\t\t%.2f W\n', Q_SS);
fprintf('Time to steady state: \t\t%.2f s\n', t_to_SS);

```

## Annex 29: Therminus-K3 Arduino Code

```
-----  
//  
-----  
// Libraries:  
-----  
#include <Adafruit_GFX.h> // Adafruit GFX library: Adafruit Industries, License: BSD.  
#include <Adafruit_ILI9341.h> // Adafruit_ILI9341 library: Adafruit Industries, License:  
BSD.  
#include "UI.h" // Custom User Interface Library.  
#include "SD.h" // SD-Card Reader Libraries (Standard Arduino SD library.  
#include "Pressure_Sensor.h" // Custom Pressure Sensor Library.  
#include "Thickness_Sensor.h" // Custom Thickness Sensor Library.  
#include "Power_Sensor.h" // Custom Power Sensor Library.  
#define ENCODER_DO_NOT_USE_INTERRUPTS // Rotary Encoder  
#include <Encoder.h> // Rotary Encoder Library by Paul Stoffregen. Source:  
https://github.com/PaulStoffregen/Encoder  
#include "PID_Script.h" // Custom Library for PID controller.  
#include "VtoT_Script.h" // Custom Library for voltage to temperature conversion.  
  
-----  
//  
-----  
// Define Pins:  
-----  
-----  
// Thermistor pins:  
#define pinT01 A1  
#define pinT02 A2  
#define pinT03 A3  
#define pinT04 A4  
#define pinT05 A5  
#define pinT06 A6  
#define pinT07 A7  
#define pinT08 A8  
#define pinT09 A9  
#define pinT10 A10  
#define pinT11 A11  
#define pinT12 A12  
int pins[] = { pinT01, pinT02, pinT03, pinT04, pinT05, pinT06, pinT07, pinT08, pinT09, pinT10, pinT11,  
pinT12 };  
  
// PWM pins:  
#define PWM_MH 2  
#define PWM_GH 3  
#define PWM_BH 4  
#define PWM_PT 5  
#define PWM_FS 6  
  
// Rotary Encoder Pins  
#define ROT_EN_KEY 8  
#define ROT_EN_S2 9  
#define ROT_EN_S1 10  
  
Encoder myEnc(ROT_EN_S1, ROT_EN_S2);  
// Define push button pins:  
#define PB_1 30  
#define PB_2 42  
  
// LCD Pins:  
#define TFT_RST 48  
#define TFT_DC 49  
#define TFT_CS 53  
Adafruit_ILI9341 tft(TFT_CS, TFT_DC, TFT_RST);  
  
-----  
//  
-----  
// Define Variables & Constants:  
-----  
-----  
  
// Initial set temperatures:  
float Thot = 35;  
float Tcold = 15;  
  
// Constants:  
const byte calType = 1; // NTC Thermistor: 1 = polynomial fit, 2 =Steinhart-hart fit  
const int intPinMax = 1023; // Max value analog read pin  
const double Vref = 4.995; // Physical reference voltage
```

```

// Timing variables:
unsigned long startTime = 0;
int elapsedMinutes = 0, elapsedSeconds = 0;
unsigned long lastMeasurementTime = 0;
const long measurementInterval = 500; // Measurement Interval set to 500 ms
const unsigned long MeasurementDuration = 25UL * 60 * 1000; // Measurement Duration set to 25 minutes

// Measurement Variables:
float Tmainheat, Tguard, Tbh, Tpel, Tenv;
float pressure, weight, TaredSampleThickness;

// Encoder variables:
int stepsPerDegree = 4, encoderSteps = 0, oldPosition = 0;

// Menu variables:
int lastButtonState = HIGH;
int lastPb1State = HIGH;
int bgColor = tft.color565(0, 166, 214);

// Operation modes:
enum OperationModes { SET_TEMPERATURE, READINGS, MEASUREMENT };
OperationModes currentMode = SET_TEMPERATURE;
bool isMeasuring = false, ShowSingleTempScreen = false, ShowAvgTempScreen = false, SelectTemp = true;

-----
// Define filter size, thermistor classes and PID classes:
-----

// Median Filter set up for temperature measurements: window size 5
MedianFilter<float> medianFilter01(5);
MedianFilter<float> medianFilter02(5);
MedianFilter<float> medianFilter03(5);
MedianFilter<float> medianFilter04(5);
MedianFilter<float> medianFilter05(5);
MedianFilter<float> medianFilter06(5);
MedianFilter<float> medianFilter07(5);
MedianFilter<float> medianFilter08(5);
MedianFilter<float> medianFilter09(5);
MedianFilter<float> medianFilter10(5);
MedianFilter<float> medianFilter11(5);
MedianFilter<float> medianFilter12(25); //Environment temperature signal has more noise, so gets bigger
window
MedianFilter<float> medianFilterMh(5);
MedianFilter<float> medianFilterGh(5);
MedianFilter<float> medianFilterBh(5);
MedianFilter<float> medianFilterPt(5);

// Temperature Measurement Coefficients Polynomial Fit
double calCoefsT01[4] = {-34.918, 38.163, -9.1814, 1.3646};
double calCoefsT02[4] = {-32.731, 35.726, -8.3117, 1.2642};
double calCoefsT03[4] = {-32.761, 35.963, -8.42, 1.2801};
double calCoefsT04[4] = {-30.744, 33.54, -7.4808, 1.1628};
double calCoefsT05[4] = {-33.12, 36.301, -8.5292, 1.2906};
double calCoefsT06[4] = {-32.97, 35.977, -8.4041, 1.2742};
double calCoefsT07[4] = {-32.799, 36.047, -8.4438, 1.282};
double calCoefsT08[4] = {-32.513, 35.702, -8.2998, 1.2642};
double calCoefsT09[4] = {-33.909, 37.035, -8.7768, 1.319};
double calCoefsT10[4] = {-33.83, 37.016, -8.7865, 1.3219};
double calCoefsT11[4] = {-32.793, 35.863, -8.3856, 1.2763};
double calCoefsT12[4] = {-33.258, 36.279, -8.5301, 1.2926};

// Number of calibration coefficients provided
byte calCoefsNR = sizeof(calCoefsT01) / sizeof(calCoefsT01[0]);

//Make a class that handles the temperature measurement for each sensor:
VtoT v2t01(pinT01, intPinMax, Vref, calType, calCoefsNR, calCoefsT01);
VtoT v2t02(pinT02, intPinMax, Vref, calType, calCoefsNR, calCoefsT02);
VtoT v2t03(pinT03, intPinMax, Vref, calType, calCoefsNR, calCoefsT03);
VtoT v2t04(pinT04, intPinMax, Vref, calType, calCoefsNR, calCoefsT04);
VtoT v2t05(pinT05, intPinMax, Vref, calType, calCoefsNR, calCoefsT05);
VtoT v2t06(pinT06, intPinMax, Vref, calType, calCoefsNR, calCoefsT06);
VtoT v2t07(pinT07, intPinMax, Vref, calType, calCoefsNR, calCoefsT07);
VtoT v2t08(pinT08, intPinMax, Vref, calType, calCoefsNR, calCoefsT08);
VtoT v2t09(pinT09, intPinMax, Vref, calType, calCoefsNR, calCoefsT09);
VtoT v2t10(pinT10, intPinMax, Vref, calType, calCoefsNR, calCoefsT10);
VtoT v2t11(pinT11, intPinMax, Vref, calType, calCoefsNR, calCoefsT11);
VtoT v2t12(pinT12, intPinMax, Vref, calType, calCoefsNR, calCoefsT12);

// PID Coefficients:
double Kp_mh = 15, Ki_mh = 0.8, Kd_mh = 0, PWMmaxMH = 65;
double Kp_gh = 507.5, Ki_gh = 1.70, Kd_gh = 0, PWMmaxGH = 255;

```

```

double Kp_bh = 450, Ki_bh = 0.9, Kd_bh = 0, PWMmaxBH = 255;
double Kp_pt = 500, Ki_pt = 5, Kd_pt = 10, PWMmaxPT = 255;

// PID classes:
PID pidMainHeater(Tmainheat, Thot, Kp_mh, Ki_mh, Kd_mh, PWMmaxMH);
PID pidGuardHeater(Tguard, Thot, Kp_gh, Ki_gh, Kd_gh, PWMmaxGH);
PID pidBackHeater(Tbh, Thot, Kp_bh, Ki_bh, Kd_bh, PWMmaxBH);
PID pidPeltier(Tpel, Tcold, Kp_pt, Ki_pt, Kd_pt, PWMmaxPT);

//-----
----
// Void setup:
//-----
----

void setup() {
  Serial.begin(115200);
  Serial.println("Therminus K3 Startup");

  // Reset PID Controller:
  pidMainHeater.Reset(Tmainheat);
  pidGuardHeater.Reset(Tguard);
  pidBackHeater.Reset(Tbh);
  pidPeltier.Reset(Tpel);

  // Power Meter Initialization
  Wire.begin();
  INA.setMaxCurrentShunt(2, 0.002);
  INA.setBusVoltageConversionTime(1);
  INA.setShuntVoltageConversionTime(1);
  INA.setAverage(6);

  // LCD Screen Initialization
  tft.begin();
  tft.setRotation(1);
  StartText();

  // Initialize Pins
  pinMode(pinTG, INPUT);
  pinMode(PB_1, INPUT);
  pinMode(PB_2, INPUT);
  pinMode(ROT_EN_KEY, INPUT);
  // Loadcell Amplifier Initialization

  LoadCell.begin();
  LoadCell.start(stabilizingtime);
  LoadCell.setCalFactor(calibrationValue);

  // Mainscreen Initialization
  delay(500);
  tft.fillScreen(ILI9341_WHITE);
  Background();
  MainScreen();
  Serial.println("Therminus K3 Initialized");
}

//-----
----
// Void loop:
//-----
----

void loop() {
  int currentButtonState = digitalRead(ROT_EN_KEY);
  int currentPB1State = digitalRead(PB_1);

  switch (currentMode) {
    case SET_TEMPERATURE:
      HandleEncoder();
      // Check for button transition from not pressed to pressed
      if (currentButtonState == LOW && lastButtonState == HIGH) {
        if (SelectTemp) {
          SelectTemp = false;
        } else {
          currentMode = READINGS;
          SelectTemp = true;
        }
      }
      break;
    case READINGS:
      Scale();
      ThicknessGauge();
      // Check for button transition from not pressed to pressed

```

```

        if (currentButtonState == LOW && lastButtonState == HIGH) {
            currentMode = MEASUREMENT;
            isMeasuring = true;
            Serial.println("Measurement Data: ");
            Serial.println();
            SerialPrintMeasurementData();
            measurementStartTime = millis();
            lastMeasurementTime = measurementStartTime - 500;
        }
        break;
    case MEASUREMENT:
        tft.drawRect(2, 2, 316, 236, ILI9341_BLACK);
        if (isMeasuring && (millis() - measurementStartTime) <= MeasurementDuration) {
            // Check if 500 ms have passed since the last measurement.
            if (millis() - lastMeasurementTime >= measurementInterval) {
                RunMeasurement(pins);
                lastMeasurementTime += measurementInterval;
            }
            DisplayTimer();
            // Check for button press:
            if (currentButtonState == LOW && lastButtonState == HIGH) {
                ShowSingleTempScreen();
                ShowSingleTempScreen = !ShowSingleTempScreen;
                if (!ShowSingleTempScreen) {
                    ShowAvgTempScreen = true;
                }
            }
            // Check for button 1 press to cancel the measurement
            if (currentPB1State == HIGH && lastPB1State == LOW) {
                tft.drawRect(2, 2, 316, 236, ILI9341_RED);
                isMeasuring = false; // Stop measuring
                analogWrite(PWM_FS, 0);
                analogWrite(PWM_MH, 0);
                analogWrite(PWM_GH, 0);
                analogWrite(PWM_BH, 0);
                analogWrite(PWM_PT, 0);
                currentMode = SET_TEMPERATURE; // Go back to SET_TEMPERATURE mode
            }
            lastPB1State = currentPB1State;
        } else if (isMeasuring && (millis() - measurementStartTime) > MeasurementDuration) {
            isMeasuring = false;
            analogWrite(PWM_FS, 0);
            analogWrite(PWM_MH, 0);
            analogWrite(PWM_GH, 0);
            analogWrite(PWM_BH, 0);
            analogWrite(PWM_PT, 0);
            currentMode = SET_TEMPERATURE;
        }
        break;
    }
    lastButtonState = currentButtonState;
}

//-----
// Functions:
//-----

void HandleEncoder() {
    long newPos = myEnc.read();
    long deltaPos = newPos - oldPosition;
    encoderSteps += deltaPos;
    UpdateCurrentTemperatures();

    if (encoderSteps >= stepsPerDegree) {
        if (SelectTemp) {
            Thot += 1.0;
            pidMainHeater.SetSetpoint(Thot, Tmainheat);
            pidGuardHeater.SetSetpoint(Thot, Tguard);
            pidBackHeater.SetSetpoint(Thot, Tbh);
        } else {
            Tcold += 1.0;
            pidPeltier.SetSetpoint(Tcold, Tpel);
        }
        encoderSteps -= stepsPerDegree;
        SetTemperature();
    } else if (encoderSteps <= -stepsPerDegree) {
        if (SelectTemp) {
            Thot -= 1.0;
            pidMainHeater.SetSetpoint(Thot, Tmainheat);
            pidGuardHeater.SetSetpoint(Thot, Tguard);
            pidBackHeater.SetSetpoint(Thot, Tbh);
        }
    }
}

```



```

    } else {
        Tcold -= 1.0;
        pidPeltier.SetSetpoint(Tcold, Tpel);
    }
    encoderSteps += stepsPerDegree;
    SetTemperature();
}
oldPosition = newPos;
}

void RunMeasurement(int pins[]) {

    // Turn on fans:
    analogWrite(PWM_FS, 255);

    // Calculate and Display Measurement & Control Interval:
    float Time = (millis() - measurementStartTime) / 1000.0;
    Serial.print(Time,1);
    Serial.print(" | ");
    Serial.print(" ");

    if (ShowAvgTempScreen) {
        tft.fillRect(3, 30, 314, 123, ILI9341_WHITE);
        AvgTempScreen();
        ShowAvgTempScreen = false;
    }
    // Measure Temperatures:
    float temperatures[12];
    temperatures[0] = medianFilter01.AddValue(v2t01.CalcT());
    temperatures[1] = medianFilter02.AddValue(v2t02.CalcT());
    temperatures[2] = medianFilter03.AddValue(v2t03.CalcT());
    temperatures[3] = medianFilter04.AddValue(v2t04.CalcT());
    temperatures[4] = medianFilter05.AddValue(v2t05.CalcT());
    temperatures[5] = medianFilter06.AddValue(v2t06.CalcT());
    temperatures[6] = medianFilter07.AddValue(v2t07.CalcT());
    temperatures[7] = medianFilter08.AddValue(v2t08.CalcT());
    temperatures[8] = medianFilter09.AddValue(v2t09.CalcT());
    temperatures[9] = medianFilter10.AddValue(v2t10.CalcT());
    temperatures[10] = medianFilter11.AddValue(v2t11.CalcT());
    temperatures[11] = medianFilter12.AddValue(v2t12.CalcT());

    // Calculate Temperature Averages:
    Tpel = medianFilterPt.AddValue((temperatures[0] + temperatures[1]) / 2);
    Tmainheat = medianFilterMh.AddValue((temperatures[2] + temperatures[3] + temperatures[4]) / 3);
    Tguard = medianFilterGh.AddValue((temperatures[5]+temperatures[6]+temperatures[7]+ temperatures[8])

    Tbh = medianFilterBh.AddValue((temperatures[9] + temperatures[10]) / 2);
    Tenv = temperatures[11];

    // Serial Print Temperatures:
    for (int i = 0; i < 12; i++) {
        char temperature[6];
        dtostrf(temperatures[i], 4, 2, temperature);
        Serial.print(temperature);
        Serial.print(" | ");
    }
    Serial.print(" ");
    Serial.print(Tmainheat);
    Serial.print(" | ");
    Serial.print(Tguard);
    Serial.print(" | ");
    Serial.print(Tbh);
    Serial.print(" | ");
    Serial.print(Tpel);
    Serial.print(" | ");

    // Display Temperature readings on TFT display:
    tft.setTextSize(2);
    tft.setTextColor(bgColor, ILI9341_WHITE);

    if (ShowSingleTempScreen) {
        // Display individual temperature readings on the TFT and Serial Monitor
        for (int i = 0; i < 12; i++) {
            char temperature[6];
            dtostrf(temperatures[i], 4, 2, temperature);
            int col = i / 6;
            int row = i % 6;
            int x = 60 + col * 160;
            int y = 33 + row * 20;
            tft.setCursor(x, y);
            tft.setTextColor(bgColor, ILI9341_WHITE);
            tft.print(temperature);
        }
    }
}

```

```

} else {
    // Display average values on the TFT
    tft.setCursor(220, 40);
    tft.println(Tmainheat, 2);
    tft.setCursor(220, 70);
    tft.println(Tguard, 2);
    tft.setCursor(220, 100);
    tft.println(Tbh, 2);
    tft.setCursor(220, 130);
    tft.println(Tpel, 2);
}

// Get the PID outputs:
int outputMH = pidMainHeater.CalcPWM(Tmainheat);
int outputGH = pidGuardHeater.CalcPWM(Tguard);
int outputBH = pidBackHeater.CalcPWM(Tbh);
int outputPT = pidPeltier.CalcPWM(Tpel, true);

// Write to the PWM pins:
analogWrite(PWM_MH, outputMH);
analogWrite(PWM_GH, outputGH);
analogWrite(PWM_BH, outputBH);
analogWrite(PWM_PT, outputPT);

// Print PWM outputs:
Serial.print(" ");
Serial.print(outputMH);
Serial.print("| ");
Serial.print(outputGH);
Serial.print("| ");
Serial.print(outputBH);
Serial.print("| ");
Serial.print(outputPT);
Serial.print("| ");

// Delay for settling and synchronization:
delay(55);

// Power Measurements:
Power_Sensor();
}

void UpdateCurrentTemperatures() {

    // Measure Temperatures:
    float temperatures[12];
    temperatures[0] = medianFilter01.AddValue(v2t01.CalcT());
    temperatures[1] = medianFilter02.AddValue(v2t02.CalcT());
    temperatures[2] = medianFilter03.AddValue(v2t03.CalcT());
    temperatures[3] = medianFilter04.AddValue(v2t04.CalcT());
    temperatures[4] = medianFilter05.AddValue(v2t05.CalcT());
    temperatures[5] = medianFilter06.AddValue(v2t06.CalcT());
    temperatures[6] = medianFilter07.AddValue(v2t07.CalcT());
    temperatures[7] = medianFilter08.AddValue(v2t08.CalcT());
    temperatures[8] = medianFilter09.AddValue(v2t09.CalcT());
    temperatures[9] = medianFilter10.AddValue(v2t10.CalcT());
    temperatures[10] = medianFilter11.AddValue(v2t11.CalcT());
    temperatures[11] = medianFilter12.AddValue(v2t12.CalcT());

    // Calculate Temperature Averages:
    Tpel = (temperatures[0] + temperatures[1]) / 2;
    Tmainheat = (temperatures[2] + temperatures[3] + temperatures[4]) / 3;
    Tguard = (temperatures[5] + temperatures[6] + temperatures[7] + temperatures[8]) / 4;
    Tbh = (temperatures[9] + temperatures[10]) / 2;
    Tenv = temperatures[11];
}

void SerialPrintMeasurementData() {

    Serial.print("Measurement Duration:| ");
    Serial.println(float(MeasurementDuration / 1000) / 60);
    Serial.print("Tenvironment:| ");
    Serial.println(medianFilter11.AddValue(v2t12.CalcT()));
    Serial.print("Thot:| ");
    Serial.println(Thot);
    Serial.print("Tcold:| ");
    Serial.println(Tcold);
    Serial.print("Applied Weight:| ");
    Serial.print(pressure, 0);
    Serial.print("| ");
    Serial.println(weight, 0);
    Serial.print("Sample Thickness:| ");

```

```

Serial.println(TaredSampleThickness, 2);
Serial.println("Time   |T1  | T2  | T3  | T4  | T5  | T6  | T7  | T8  | T9  | T10 |
                T11 | T12 |    Tmh | Tgh  | Tbh  | Tcp  |    PWM1|PWM2|PWM3|PWM4|
                Watt | V   | I   ");
}

//-----
//-----
// PID_script.h
//-----
//-----

#ifndef PID_Script_h
#define PID_Script_h
#include <Arduino.h>

class PID {
public:
    PID(double Tmeas, double setpoint, double Kp, double Ki, double Kd, double PWMmax);
    //void SetIntegrationRange(double range);
    void Reset(double Tmeas);
    void SetSetpoint(double setpoint, double Tmeas);
    double CalcPWM(double Tmeas, bool inverse = false);
private:
    double _setpoint;
    double _Kp;
    double _Ki;
    double _Kd;
    double _PWMmax;
    //double _IntegrationRange;
    double _lastTmeas;
    double _integ;
    double _deriv;
    double _PWM;
    double _err;
    double _lastTime;
};

#endif

//-----
//-----
// PID_script.cpp
//-----
//-----

#include "PID_Script.h"

PID::PID(double Tmeas, double setpoint, double Kp, double Ki, double Kd, double PWMmax) {
    _setpoint = setpoint;
    _Kp = Kp;
    _Ki = Ki;
    _Kd = Kd;
    _PWMmax = PWMmax;
    Reset(Tmeas);
}

void PID::Reset(double Tmeas) {
    _lastTmeas = Tmeas; // Initialize with the current temperature
    _integ = 0.0;
    _deriv = 0.0;
    _PWM = 0.0;
    _err = 0.0;
    _lastTime = millis() / 1000.0; // Convert to seconds
}

void PID::SetSetpoint(double setpoint, double Tmeas) {
    _setpoint = setpoint;
    Reset(Tmeas);
}

double PID::CalcPWM(double Tmeas, bool inverse) {
    double currentTime = millis() / 1000.0;
    double deltaTime = currentTime - _lastTime;
    // Determine the error based on the mode
    if (inverse) {
        _err = Tmeas - _setpoint; // Inverse mode for cooling
    } else {
        _err = _setpoint - Tmeas; // Regular mode for heating
    }

    // Proportional term
    double Pout = _Kp * _err;

```

```

// Integral term with anti-windup
if (!((_PWM >= _PWMmax && _err > 0) || (_PWM <= 0 && _err < 0))) {
    _integ += _Ki * _err * deltaTime;
    _integ = constrain(_integ, -_PWMmax, _PWMmax);
}

// Derivative term
double derivative = (Tmeas - _lastTmeas) / deltaTime;
double Dout = _Kd * derivative;

// Calculate total output
double PWM = Pout + _integ - Dout;
_PWM = constrain(PWM, 0, _PWMmax); // Constrain PWM within bounds
_lastTime = currentTime;
_lastTmeas = Tmeas;
return _PWM;
}

//-----
// Power_Sensor.h
//-----

//Power Meter Libraries:
#include <Wire.h>
#include "INA226.h" // INA226 library by Rob Tillaart. Source: https://github.com/RobTillaart/INA226

// Power meter pins:
INA226 INA(0x40);

// Medianfilter for current measurement:
MedianFilter<float> medianFilterI(10);

// Linear regression coefficients:
const float slope = 0.97214;;
const float intercept = 3.1859;

// Power Sensor Settings:
void Power_Sensor(){
    // Power Measurements:
    float FilteredCurrent = medianFilterI.AddValue(INA.getCurrent_mA());
    float MainheaterCurrent = FilteredCurrent * slope + intercept;
    float MainheaterVoltage = INA.getBusVoltage();
    float MainheaterPower_mW = MainheaterCurrent * MainheaterVoltage;
    float MainheaterPower_W = MainheaterPower_mW/1000;
    tft.setCursor(220, 160);
    tft.print(MainheaterPower_W, 2);
    tft.print(" ");
    Serial.print(" ");
    Serial.print(MainheaterPower_mW, 2);
    Serial.print(" | ");
    Serial.print(MainheaterVoltage, 2);
    Serial.print(" | ");
    Serial.println(MainheaterCurrent, 3);
}

//-----
// Pressure_Sensor.h
//-----

#include <HX711_ADC.h> // HX711_ADC library by Olav Kallhovd. Source:
https://github.com/olkal/HX711\_ADC
#include <MedianFilterLib.h> // MedianFilterLib by Luis Llamas. Source:
https://github.com/luisllamasbinaburo/Arduino-MedianFilter
MedianFilter<float> medianFilterPS(20);

// Loadcell Amplifier pins:
#define HX711_Dout 11
#define HX711_SCK 12
HX711_ADC LoadCell(HX711_Dout, HX711_SCK);

// Tare Buttons pin:
#define PB_1 30

// Loadcell Amplifier Variables:
const float calibrationValue = 51.92;
unsigned long stabilizingtime = 2000;
const float area = 0.024964; // Area in square meters

```

```

const float g = 9.81; // gravitational acceleration in m/s^2

// Define Extern variable for pressure:
extern float pressure;
extern float weight;
void Scale() {
    static boolean newDataReady = 0;

    // Check for new data/start next conversion:
    if (LoadCell.update()) newDataReady = true;
    if (newDataReady) {
        weight = medianFilterPS.AddValue(LoadCell.getData());
        float force = (weight/1000)*g;
        pressure = force / area; // Pressure in Pascals (Pa)
        tft.setTextColor(bgColor, ILI9341_WHITE);
        tft.setCursor(220, 190);
        tft.print(pressure,0);
        //tft.print(weight,0);
        tft.print(" ");
        newDataReady = 0;
    }

    if (digitalRead(PB_1) == HIGH) {
        LoadCell.tare();
        tft.setCursor(220, 190);
        tft.print(" ");
        delay(200);
    }
}

//-----
// Thickness_Sensor.h
//-----

#define pinTG A0 // Thickness Gauge pin:
#define PB_2 42 // Tare Buttons pin:

// Define Extern variables:
extern const double Vref;
extern float TaredSampleThickness;
// Thickness Gauge variables:
float tareOffset = 0.0;
const float P1 = -0.0275, P2 = 23.6790;
// MedianFilter Window Size:
MedianFilter<float> filter(20);

void ThicknessGauge() {
    float Raw_Data = analogRead(pinTG);
    float Fil_Data = filter.AddValue(Raw_Data);

    float SampleThickness = P1 * Raw_Data + P2;
    float AveragedSampleThickness = P1 * Fil_Data + P2;

    if (digitalRead(PB_2) == HIGH) {
        tareOffset = AveragedSampleThickness;
        delay(100);
    }

    TaredSampleThickness = AveragedSampleThickness - tareOffset;
    tft.setTextColor(bgColor, ILI9341_WHITE);
    tft.setCursor(220, 220);
    tft.print(TaredSampleThickness, 2);
    tft.print(" ");
}

//-----
// UI.h
//-----

// User Interface library:
#ifndef UI_h
#define UI_h
#include <Adafruit_ILI9341.h>
#include <Fonts/FreeSansBold24pt7b.h>
#include <Fonts/FreeSans24pt7b.h>
#include <Fonts/FreeSans9pt7b.h>
#include <Fonts/FreeSans12pt7b.h>

// Access variables from main script:

```

```

extern Adafruit_ILI9341 tft;
extern int bgColor;
extern float Tcold;
extern float Thot;
extern const unsigned long MeasurementDuration;
unsigned long measurementStartTime = 0;

// 'TU_Logo', 24x24px
const unsigned char epd_bitmap_TU_Logo [] PROGMEM = {
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfd, 0xff, 0xff, 0xf3, 0xff, 0xff, 0xc7, 0xff,
  0xfc, 0x1f, 0xff, 0xe0, 0x3f, 0xff, 0x80, 0x7f, 0xfe, 0x00, 0xff, 0xf8, 0x01, 0xff, 0xf0, 0x63,
  0xff, 0xe0, 0xc3, 0xff, 0xc1, 0xc1, 0xf7, 0xc3, 0x80, 0xcf, 0x87, 0x00, 0x0f, 0x8f, 0x04, 0x1f,
  0x9f, 0x0c, 0x1f, 0xbf, 0x18, 0x3f, 0xbf, 0xf8, 0x7f, 0xff, 0xf8, 0xff, 0xff, 0xf1, 0xff, 0xff,
  0xe3, 0xff, 0xff, 0xcf, 0xff, 0xff, 0xff, 0xff
};

void Background() {
  tft.fillRect(2, 2, 316, 27, bgColor);
  tft.drawRect(2, 2, 316, 236, ILI9341_BLACK);
  tft.drawRect(2, 29, 316, 125, ILI9341_BLACK);
  tft.drawRect(2, 29, 158, 125, ILI9341_BLACK);
  tft.drawRect(2, 180, 316, 30, ILI9341_BLACK);
}

void MainScreen() {
  tft.setFont(NULL);
  tft.setTextSize(2);

  // Print Temperature Settings:
  tft.setTextColor(ILI9341_BLACK, bgColor);
  tft.setCursor(15, 10);
  tft.print("Th:");
  tft.print(Thot, 0);
  tft.setTextSize(1);
  tft.print(char(247));
  tft.setTextSize(2);
  tft.print("C");
  tft.setCursor(135, 10);
  tft.print("Tc:");
  tft.print(Tcold, 0);
  tft.setTextSize(1);
  tft.print(char(247));
  tft.setTextSize(2);
  tft.print("C");

  // Print Measurement Duration:
  tft.setCursor(250, 10);
  int TotalSeconds = MeasurementDuration / 1000;
  int MeasurementMinutes = TotalSeconds / 60;
  int MeasurementSeconds = TotalSeconds % 60;

  if (MeasurementMinutes < 10) {
    tft.print("0");
  }

  tft.print(MeasurementMinutes);
  tft.print(":");

  if (MeasurementSeconds < 10) {
    tft.print("0");
  }

  tft.print(MeasurementSeconds);

  // Print Average Temperature Readings:
  tft.fillRect(3, 30, 314, 123, ILI9341_WHITE);
  tft.setTextColor(ILI9341_BLACK, ILI9341_WHITE);
  tft.setCursor(15, 40);
  tft.print("Avg MH temp:");
  tft.setTextSize(1);
  tft.setCursor(295, 40);
  tft.print(char(247));
  tft.setTextSize(2);
  tft.print("C");
  tft.setCursor(15, 70);
  tft.print("Avg GH temp:");
  tft.setTextSize(1);
  tft.setCursor(295, 70);
  tft.print(char(247));
  tft.setTextSize(2);
  tft.print("C");
  tft.setCursor(15, 100);
  tft.print("Avg BH temp:");
}

```



```

tft.setTextSize(1);
tft.setCursor(295, 100);
tft.print(char(247));
tft.setTextSize(2);
tft.print("C");
tft.setCursor(15, 130);
tft.print("Avg CP temp:");
tft.setTextSize(1);
tft.setCursor(295, 130);
tft.print(char(247));
tft.setTextSize(2);
tft.print("C");

// Print Power Reading:
tft.setCursor(15, 160);
tft.print("Power Mainheater:");
tft.setCursor(220, 160);
tft.print("      W");

// Print Loadcell Readings
tft.setCursor(15, 190);
tft.print("Applied Pressure:");
tft.setCursor(220, 190);
tft.print("      Pa");

// Print Thickness Readings:
tft.setCursor(15, 220);
tft.print("Sample Thickness:");
tft.setCursor(220, 219);
tft.print("      mm");
}

void DisplayTimer() {
  unsigned long ElapsedTime = millis() - measurementStartTime;
  int TotalSeconds = MeasurementDuration / 1000;
  int TotalElapsedSeconds = ElapsedTime / 1000;
  int RemainingSecondsTotal = TotalSeconds - TotalElapsedSeconds;

  // Convert the remaining time in seconds to minutes and seconds format
  int remainingMinutes = RemainingSecondsTotal / 60;
  int remainingSeconds = RemainingSecondsTotal % 60;

  // Position for lower left corner and set properties
  tft.setCursor(250, 10);
  tft.setTextSize(2);
  tft.setTextColor(ILI9341_BLACK, bgColor);

  if (remainingMinutes < 10) {
    tft.print("0");
  }
  tft.print(remainingMinutes);
  tft.print(":");

  if (remainingSeconds < 10) {
    tft.print("0");
  }
  tft.print(remainingSeconds);
}

void SingleTempScreen(){
  tft.fillRect(3, 30, 314, 123, ILI9341_WHITE);
  tft.setTextColor(ILI9341_BLACK, ILI9341_WHITE);

  // Print Individual Temperature Readings:
  for (int i = 0; i < 12; i++) {
    int col = i / 6;
    int row = i % 6;
    int x = 15 + col * 160;
    int y = 33 + row * 20;
    String sensorLabel = "T" + String(i + 1) + ":";
    if (i < 9) sensorLabel += " ";
    tft.setCursor(x, y);
    tft.setTextSize(2);
    tft.print(sensorLabel);
    tft.setCursor(x + 120, y);
    tft.setTextSize(1);
    tft.print(char(247));
    tft.setTextSize(2);
    tft.print("C");
  }
}

void AvgTempScreen() {

```

```

tft.fillRect(3, 30, 314, 123, ILI9341_WHITE);
tft.setTextColor(ILI9341_BLACK, ILI9341_WHITE);

// Print Average Temperature Readings:
tft.setCursor(15, 40);
tft.print("Avg MH temp:");
tft.setTextSize(1);
tft.setCursor(295, 40);
tft.print(char(247));
tft.setTextSize(2);
tft.print("C");
tft.setCursor(15, 70);
tft.print("Avg GH temp:");
tft.setTextSize(1);
tft.setCursor(295, 70);
tft.print(char(247));
tft.setTextSize(2);
tft.print("C");
tft.setCursor(15, 100);
tft.print("Avg BH temp:");
tft.setTextSize(1);
tft.setCursor(295, 100);
tft.print(char(247));
tft.setTextSize(2);
tft.print("C");
tft.setCursor(15, 130);
tft.print("Avg CP temp:");
tft.setTextSize(1);
tft.setCursor(295, 130);
tft.print(char(247));
tft.setTextSize(2);
tft.print("C");
}

void SetTemperature() {
  tft.setTextColor(ILI9341_BLACK, bgColor);
  tft.setCursor(51, 10);
  tft.print(Thot, 0);
  tft.setCursor(170, 10);
  if (Tcold < 10) {tft.print("0");}
  tft.print(Tcold, 0);
}

unsigned long StartText() {
  String text1_1 = "T";
  String text1_2 = "U";
  String text1_3 = "Delft";
  String text2 = "Project Therminus";
  String text3 = "By O.C.H ten Haven";
  tft.fillRect(ILI9341_WHITE);
  tft.drawBitmap(90, 58, epd_bitmap_TU_Logo, 24, 24, ILI9341_WHITE, ILI9341_BLACK);
  tft.setTextColor(ILI9341_BLACK);
  tft.setCursor(80, 110);
  tft.setFont(&FreeSansBold24pt7b);
  tft.print(text1_1);
  tft.setTextColor(bgColor);
  tft.print(text1_2);
  tft.setFont(&FreeSans24pt7b);
  tft.setTextColor(ILI9341_BLACK);
  tft.print(text1_3);
  int x1, y1, x2, y2;
  int w1, h1, w2, h2;
  tft.setFont(&FreeSans12pt7b);
  tft.getTextBounds(text2, 0, 0, &x1, &y1, &w1, &h1);
  tft.setCursor((tft.width() - w1) / 2, 145);
  tft.println(text2);
  tft.setFont(&FreeSans9pt7b);
  tft.getTextBounds(text3, 0, 0, &x2, &y2, &w2, &h2);
  tft.setCursor((tft.width() - w2) / 2, 220);
  tft.setTextColor(ILI9341_BLACK);
  tft.println(text3);
}
#endif

//-----
// VtoT_Script.h
//-----

#ifndef VtoT_Script_h
#define VtoT_Script_h
#include <Arduino.h>

```

```

class VtoT {
public:
    VtoT(int pin, int intPinMax, double Vmax, byte calType, byte calCoefsNR, double *calCoefs);
    // Method to calculate and return the actual voltage based on the analog pin reading
    double CalcV();

    // Method to calculate and return the temperature based on the actual voltage and calibration data
    double CalcT();
private:
    int _pin;           // Pin to read the voltage from
    int _PinMax;        // Maximum possible pin signal value
    double _Vmax;       // Maximum possible voltage value
    double _Vtrue;      // Calculated true voltage value
    byte _calType;      // Type of calibration to use: 1 for polynomial, 2 for Steinhart-Hart
    byte _calCoefsNR;   // Number of calibration coefficients in the array
    double *_calCoefs; // Array of calibration coefficients for n-th order polynomial: [a b c ...] -->
    y=a+bx+cx^2 + ...
};

#endif

//-----
// VtoT_Script.cpp
//-----

#include <Arduino.h>
#include "VtoT_Script.h"
// Constructor: Initializes a new VtoT object with the given pin information and calibration data
VtoT::VtoT(int pin, int intPinMax, double Vmax, byte calType, byte calCoefsNR, double *calCoefs){
    pinMode(pin, INPUT);
    _pin = pin;
    _PinMax = intPinMax;
    _Vmax = Vmax;
    _calType = calType;
    _calCoefsNR = calCoefsNR;
    _calCoefs = calCoefs;
}
// Function to calculate the true voltage based on the analog pin value
double VtoT::CalcV(){
    int _PinVal = analogRead(_pin);
    _Vtrue = _Vmax * ((float)_PinVal/(float)_PinMax);
    return _Vtrue;
}
// Function to calculate the temperature based on the true voltage and calibration data
double VtoT::CalcT(){
    double _T = 0;
    double _V = CalcV();

    if (_calType == 1) { // Polynomial calibration
        // Loop through the calibration coefficients to calculate the temperature using a polynomial
        // expression
        for (byte i=0; i<_calCoefsNR; i++) {
            _T = _T + _calCoefs[i]*pow(_V, i); // Accumulate the temperature value based on the polynomial
            // term
        }
    } else if (_calType == 2) { // Steinhart-Hart calibration
        double R_ref = 10000.0; // Reference resistance in ohms (10k ohms)
        double R_NTC = R_ref * (_Vmax / _V - 1); // Calculate the NTC thermistor resistance
        // Apply the Steinhart-Hart equation
        double logR_NTC = log(R_NTC);
        _T = 1.0 / (_calCoefs[0] + _calCoefs[1] * logR_NTC + _calCoefs[2] * pow(logR_NTC, 3));

        // Convert temperature from Kelvin to Celsius
        _T = _T - 273.15;
    }

    return _T; // Return the calculated temperature
}
}

```

# Annex 30: Therminus-K3 Technical Drawings

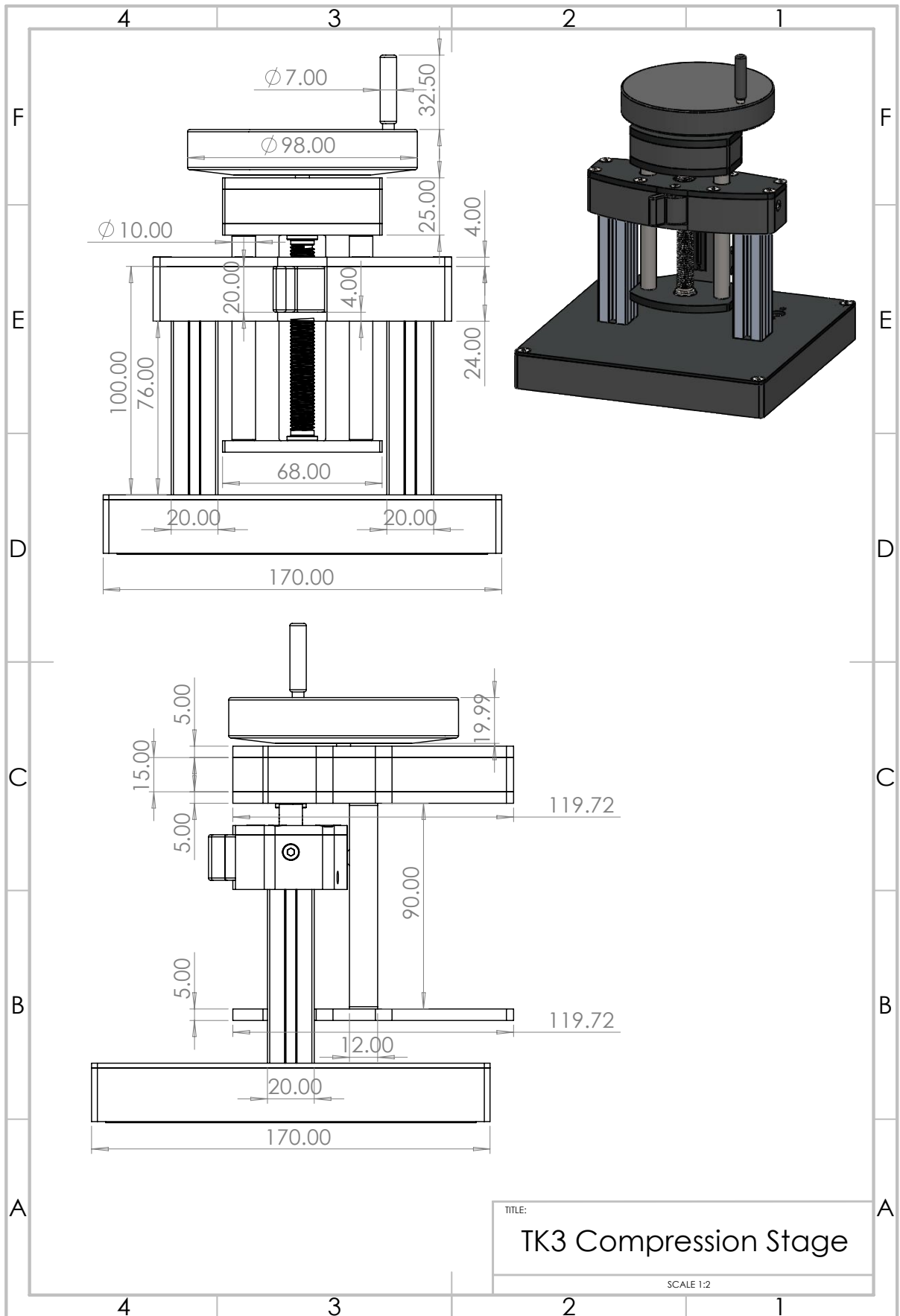


Figure 45: Technical Drawing of Compression Stage

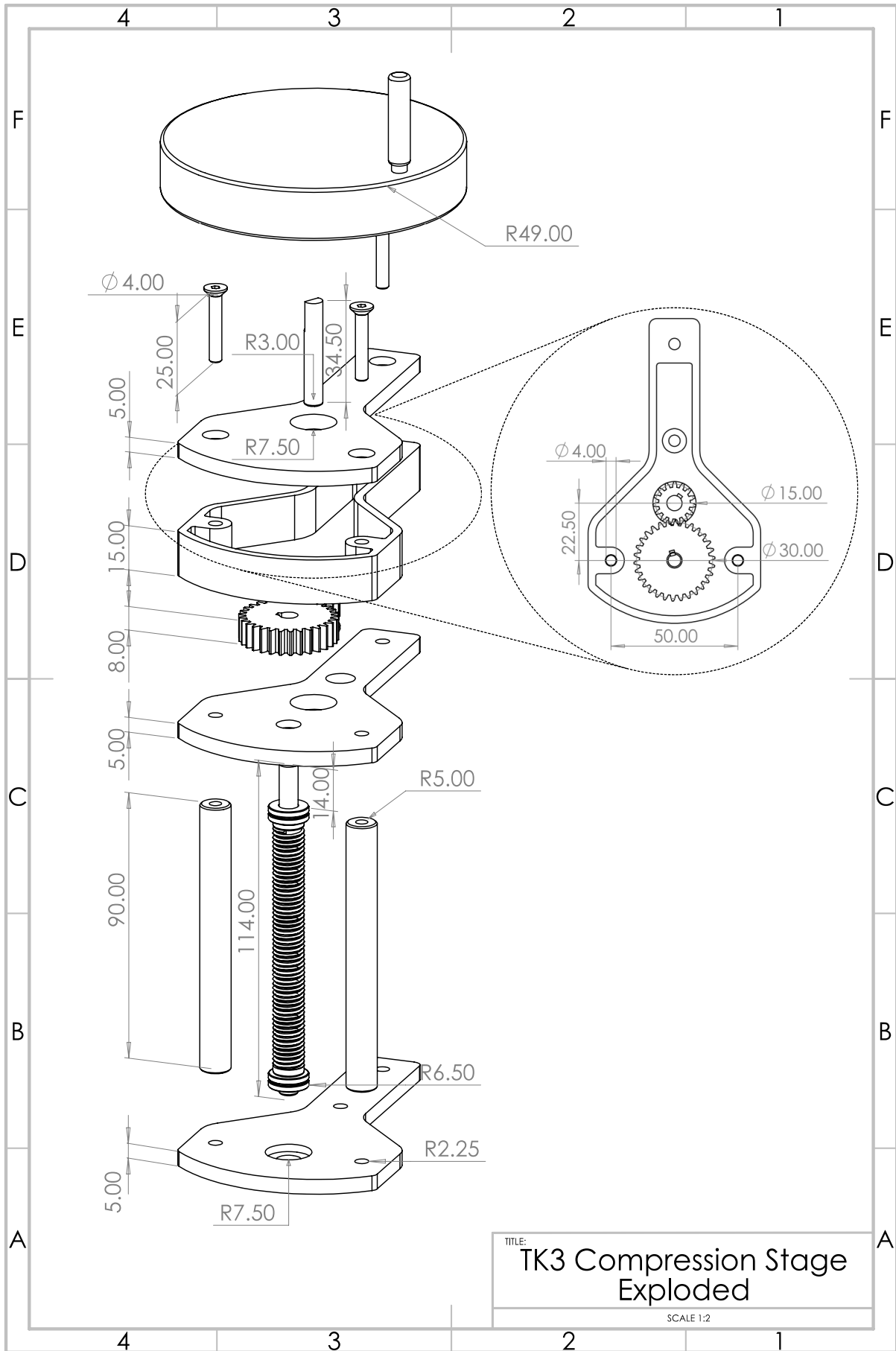


Figure 46: Exploded View Technical Drawing of Compression Stage

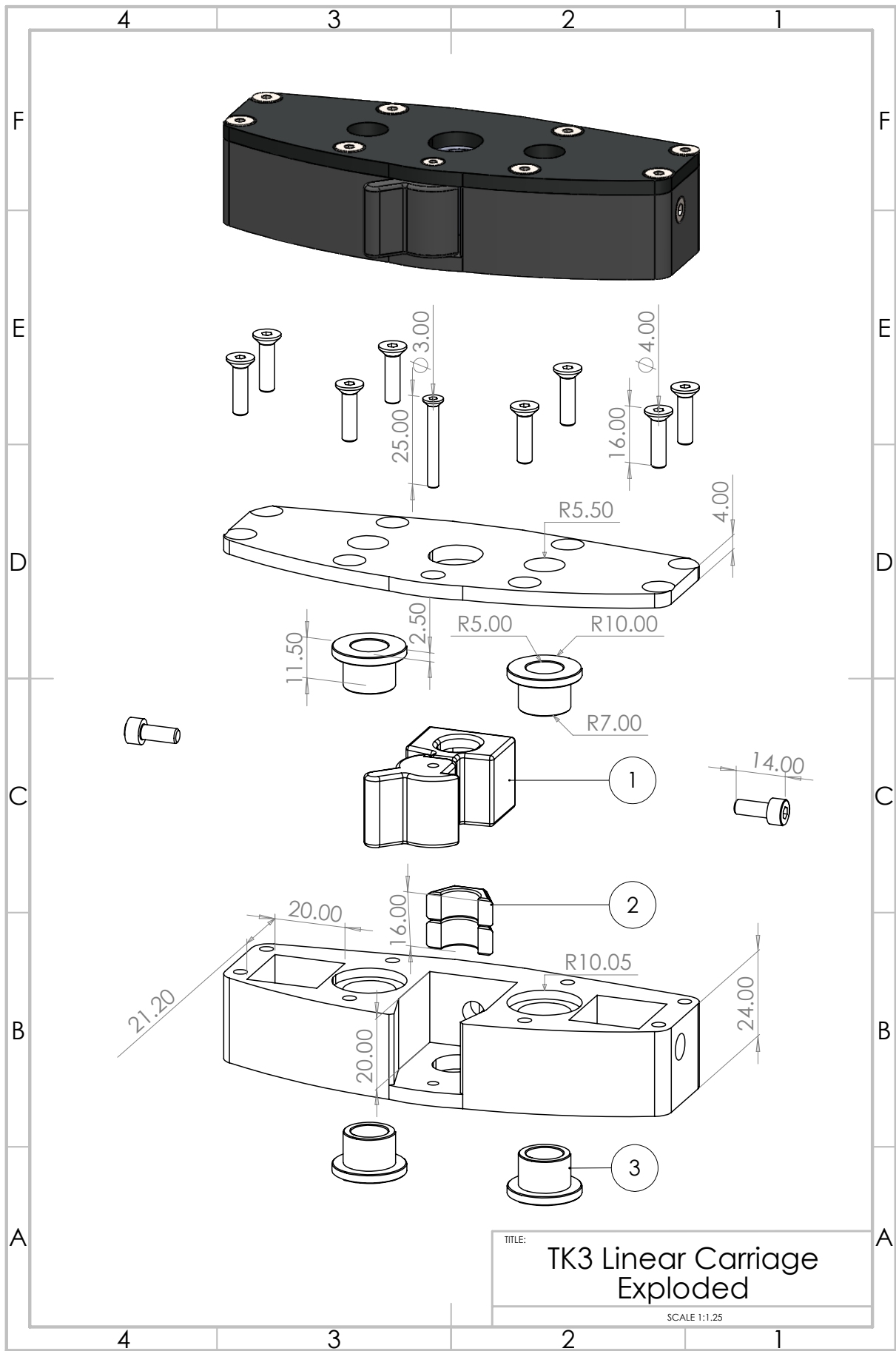
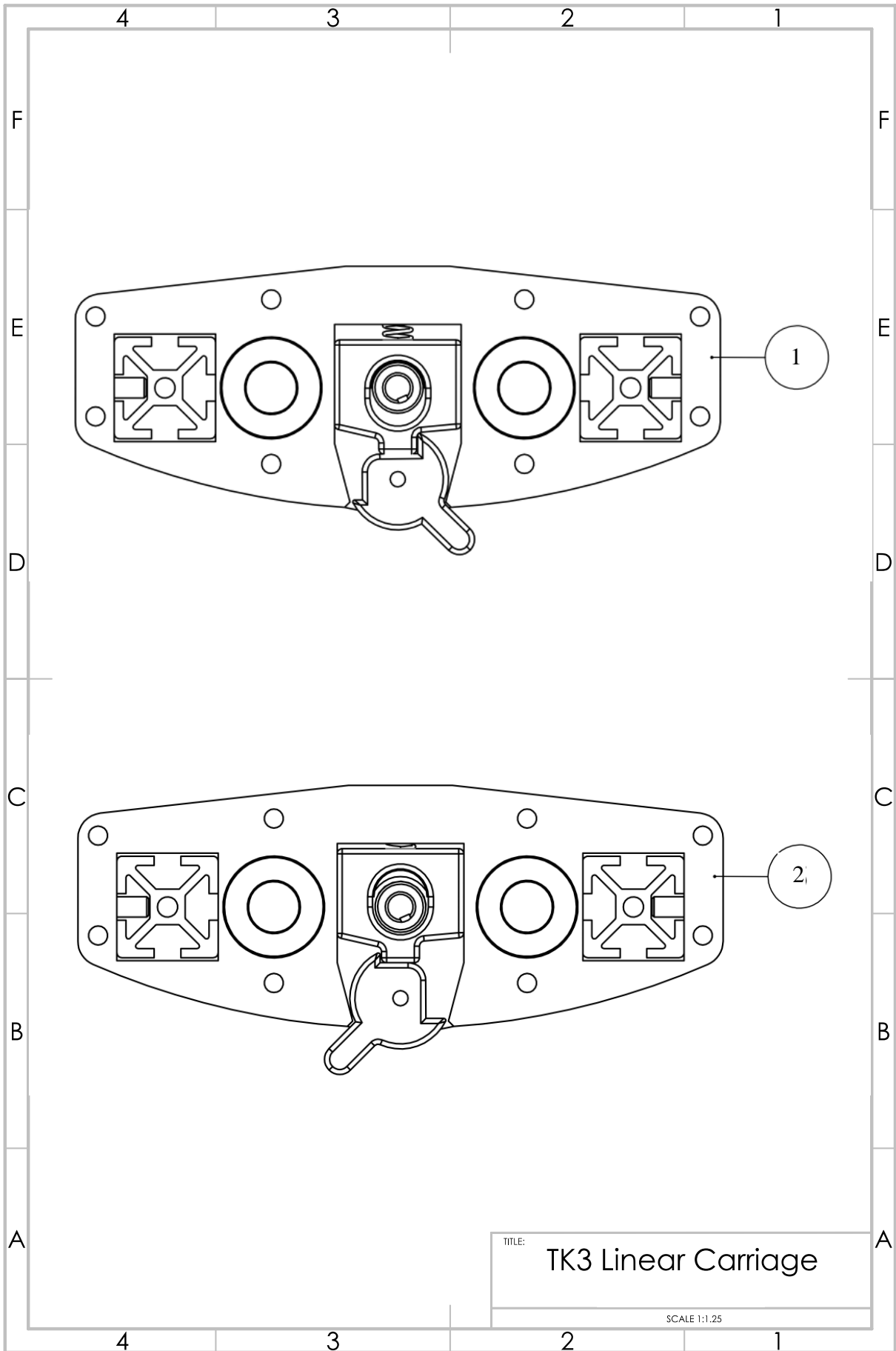


Figure 47: Exploded View Technical Drawing of Linear Carriage:  
 1. Quick release mechanism. 2. Dual M10 half-nuts to engage with the linear rod. 3. Sleeve bearings.





**Figure 48: Top View of Linear Carriage**

*1. Quick release mechanism is off, linear rod is engaged with the linear carriage. 2. Quick release mechanism is switched on, linear rod is disengaged, thus linear carriage is free to move.*

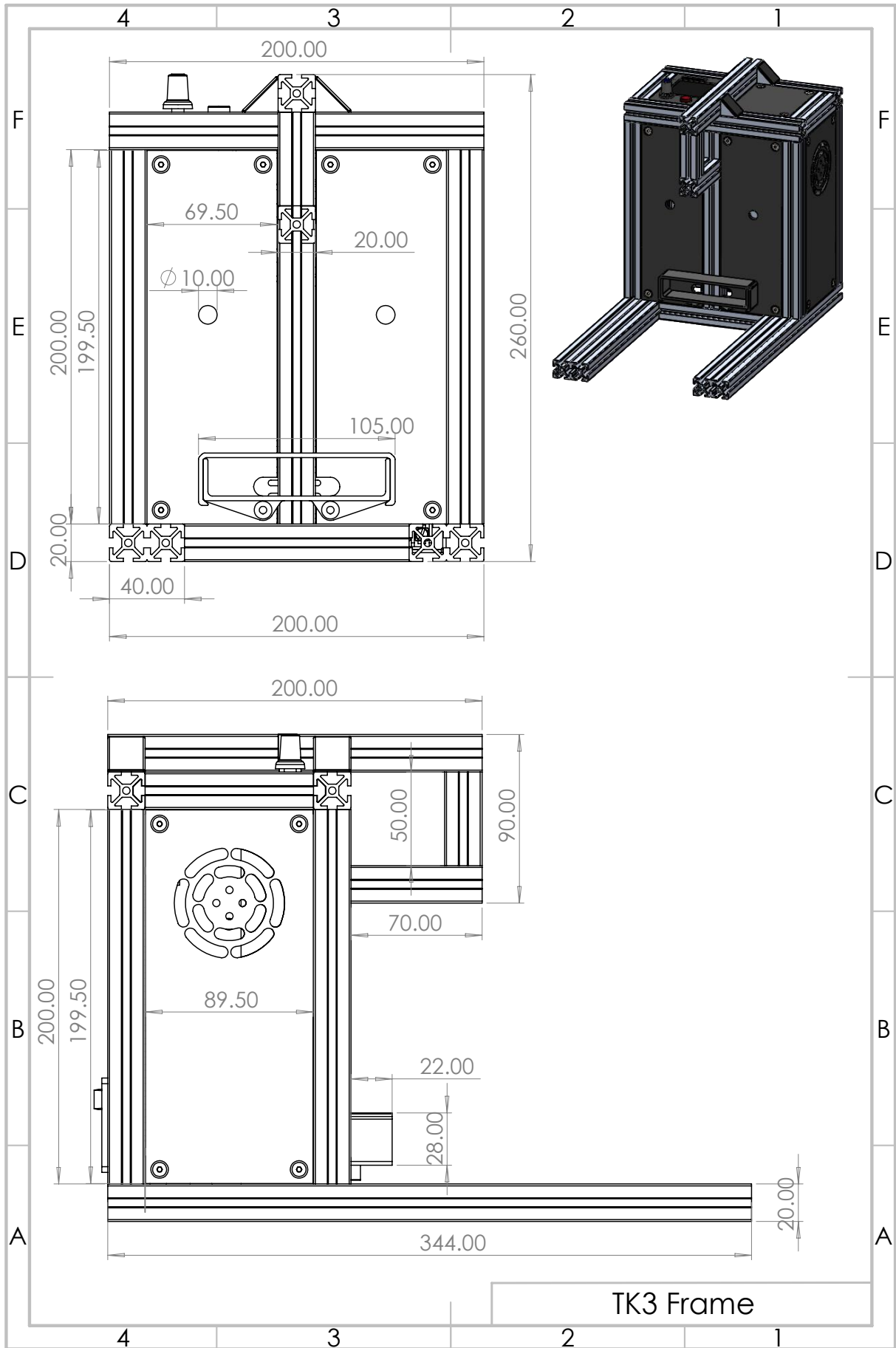


Figure 49: Technical Drawing of Aluminium Frame Assembly



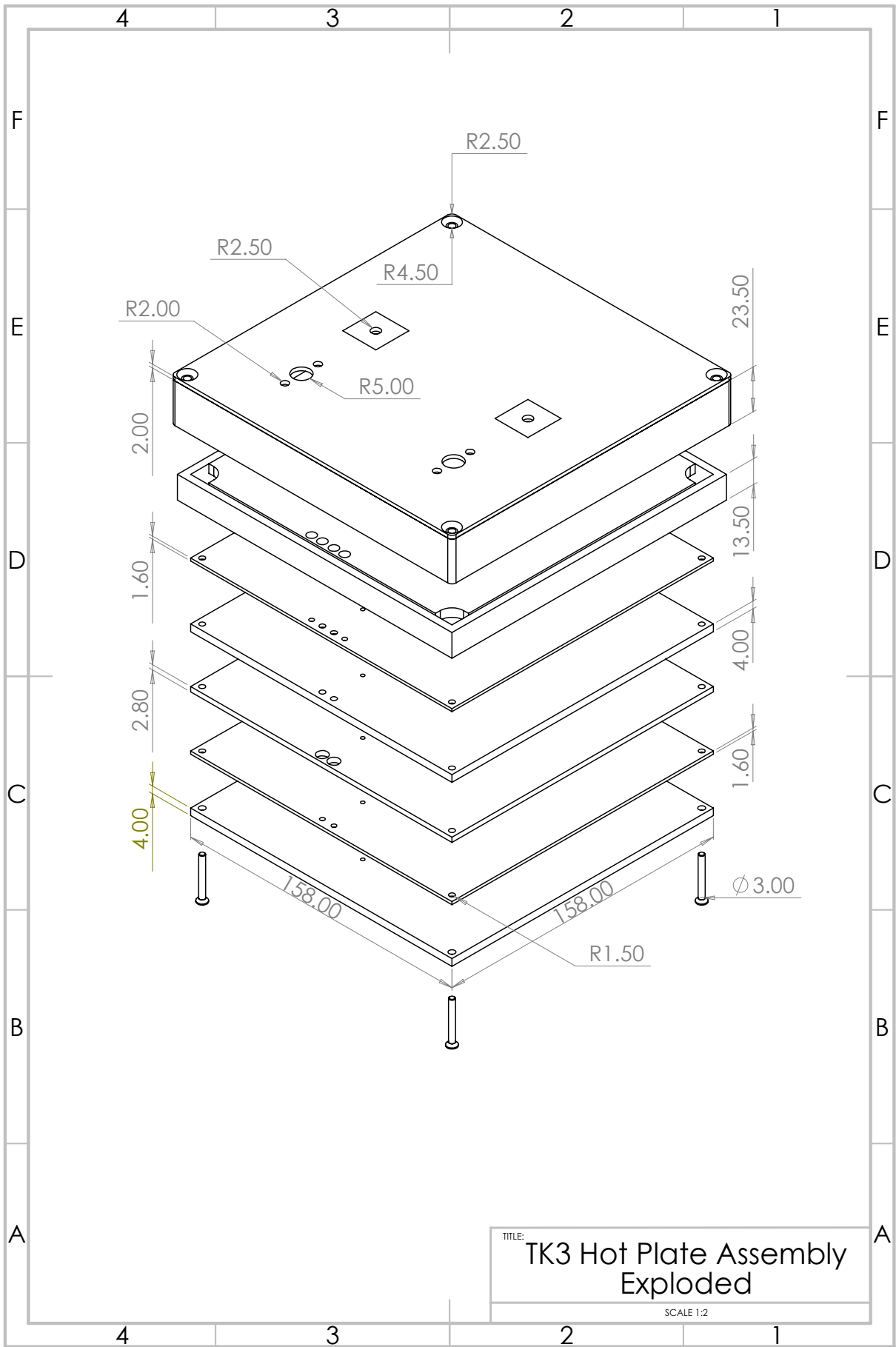


Figure 51: Technical Drawing of Hot Stage Assembly Exploded View

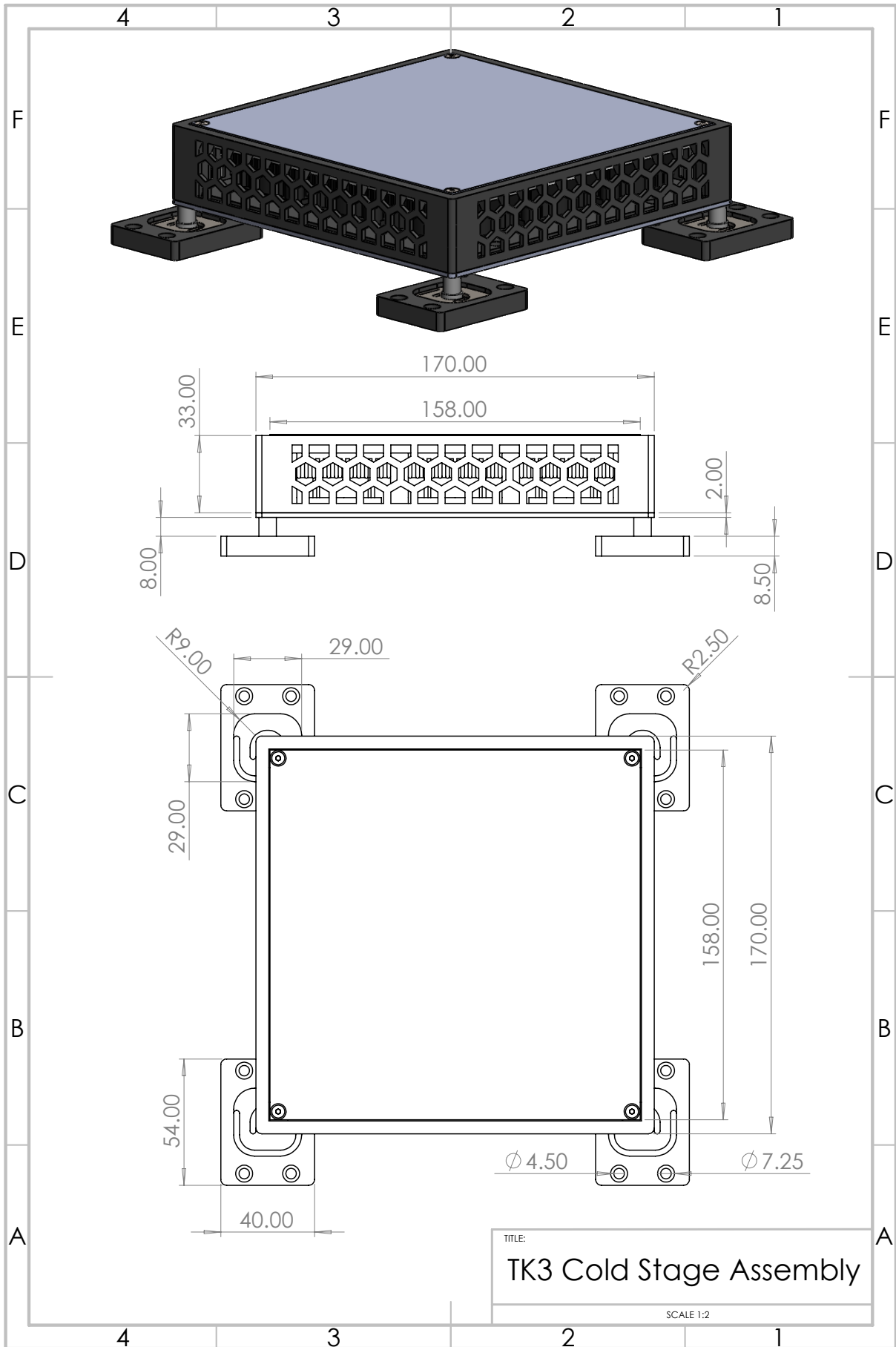


Figure 52: Technical Drawing of Cold Stage Assembly

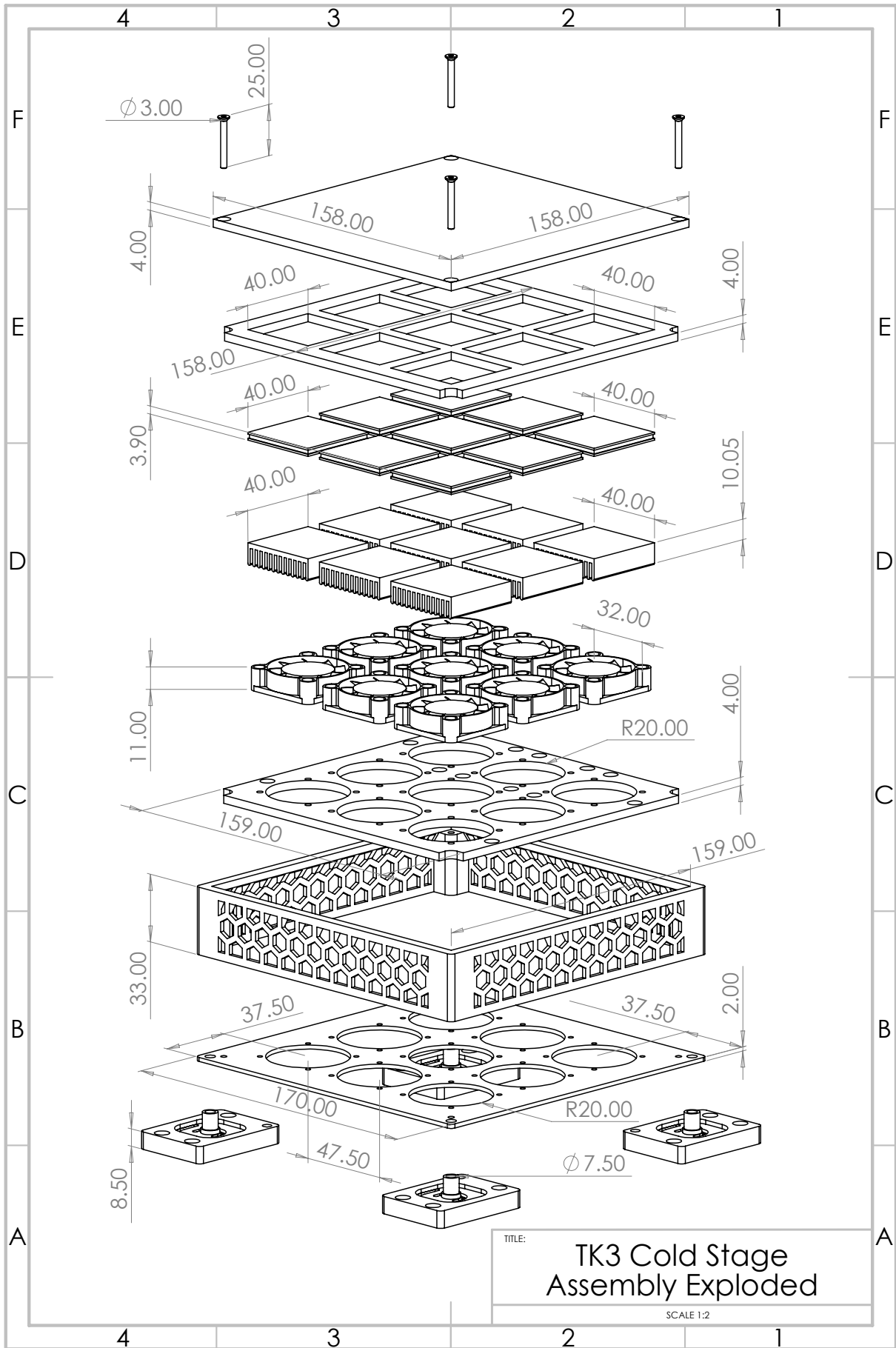


Figure 53: Technical Drawing of Cold Stage Assembly Exploded View



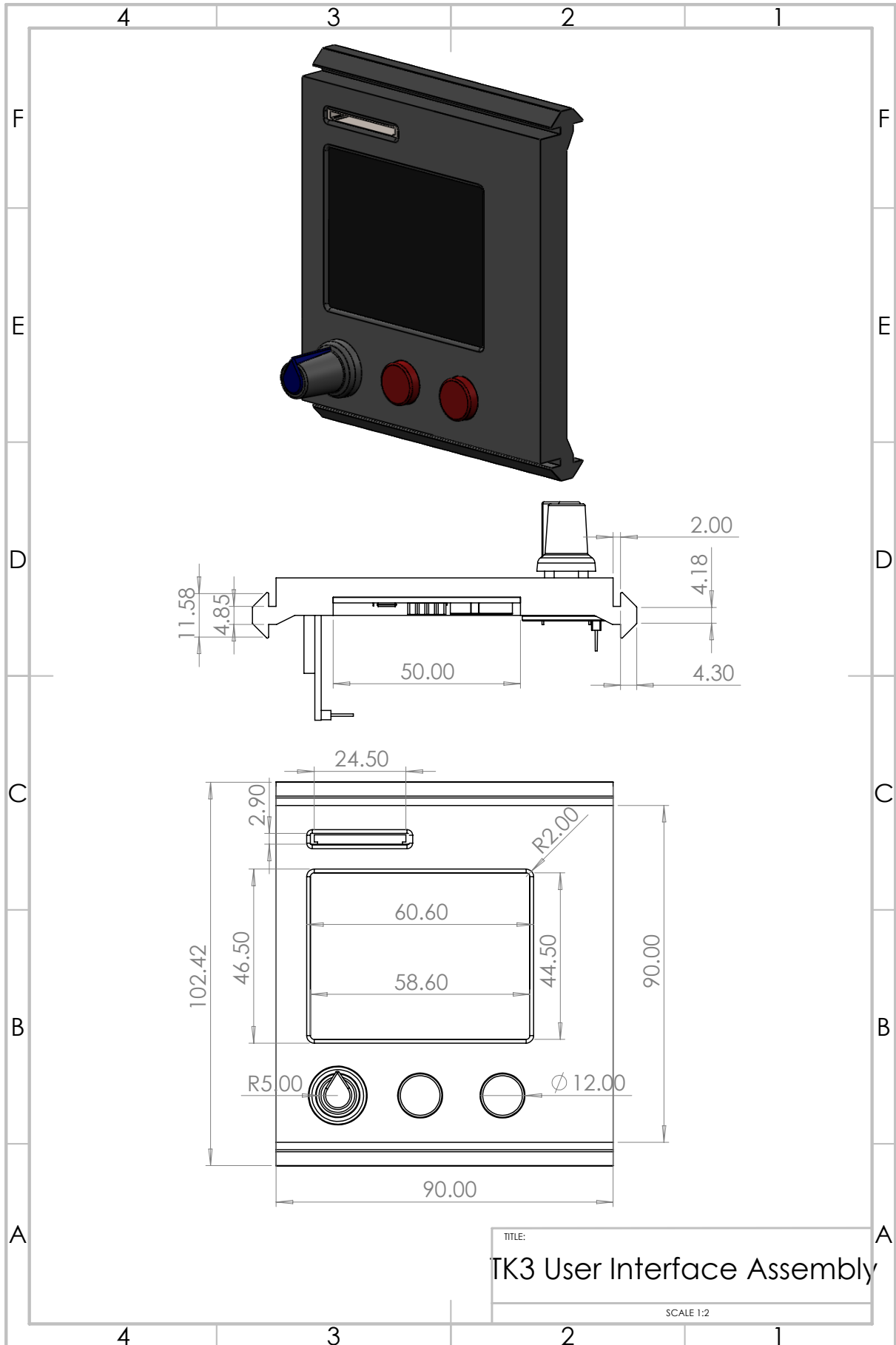


Figure 54: Technical Drawing of User Interface Assembly

## Bibliography

1. *Thermal performance of building materials and products — Determination of thermal resistance by means of guarded hot plate and heat flow meter methods — Products of high and medium thermal resistance*, B.S. Insitution, Editor. 2001.
2. *Standard Test Method for Steady-State Heat Flux Measurements and Thermal Transmission Properties by Means of the Guarded-Hot-Plate Apparatus*, A. International, Editor. 2019.
3. *Standard Practice for Using a Guarded-Hot-Plate Apparatus or Thin-Heater Apparatus in the Single-Sided Mode*, A. International, Editor. 2016.
4. Dessing, O., *Surface Testing Methods to Define Thermal Properties of Surfaces at Crime Scenes*. 2018, Delft University of Technology
5. Wilk, L., et al., *Reconstructing the time since death using noninvasive thermometry and numerical analysis*. Science Advances, 2020. **6**: p. eaba4243.
6. Ivo Sebastiaan Best, N.S.N., Sebastiaan Thomas Njio, Jason Morgan van Schijndel, Fabian Matthias Verhage, *Measuring thermal conductivity of textiles for time of death estimation*. 2019.
7. Kaanen, C.I., *Therminus-K2: A Thermal Conductivity Measurement Device for Textile Layers at a Crime Scene*. 2022.
8. *NEN-EN-IEC 60068-2-31 Environmental testing - Part 2-31: Tests - Test Ec: Rough handling shocks, primarily for equipment-type specimens*, IEC, Editor. 2008.
9. Swearingen, J.J., C.D. Wheelwright, and J.D. Garner, *An analysis of sitting areas and pressures of man*. 1962.
10. Rajala, S. and J. Leikkala, *Pressure mapping system for physiological measurements*. 18th IMEKO World Congress 2006: Metrology for a Sustainable Development, 2006. **3**.
11. Bush, T., et al., *A Comparison of Pressure Mapping Between Two Pressure-Reducing Methods for the Sacral Region*. Journal of wound, ostomy, and continence nursing: official publication of The Wound, Ostomy and Continence Nurses Society / WOCN, 2015.
12. C. Everton, S.B., W. Brito, P. Collé, A. P. Franco, S. Lutjeber, K. Nodeland, S. Rième, M. Siddika, J. Webb, S. Angmörterh, *An experimental study to compare the interface pressure and experience of healthy participants when lying still for 20 minutes in a supine position on two different imaging surfaces*. 2014.
13. Yuan, F., et al., *A novel bamboo sheet chair and its influence on sitting comfort*. PeerJ, 2020. **8**: p. e9476.
14. SolidWorks, C., *SolidWorks 3D CAD 2023*. Dassault Systèmes: Waltham, MA.
15. *NXFT15XV103FA2B Thermistor*. 2018, Murata Manufacturing Co.: Japan. <https://www.murata.com/en-us/products/productdetail?partno=NXFT15XV103FA2B150>.
16. *Greisinger G1700 Thermometer*. GHM GROUP - Greisinger: Germany. [https://www.greisinger.de/files/upload/en/produkte/bda/G1700\\_EN.pdf](https://www.greisinger.de/files/upload/en/produkte/bda/G1700_EN.pdf).
17. *TEC1-12706 Thermoelectric Cooler*. Hebei I.T. Co. Ltd. <https://peltiermodules.com/peltier.datasheet/TEC1-12706.pdf>.
18. *Ultimaker PLA Technical Data Sheet*. 2022, Ultimaker Netherlands.
19. Aldosari, F., et al., *Finite Element Analysis of PolyLactic Acid (PLA) under Tensile and Compressive Loading*. Journal of Physics: Conference Series, 2023. **2468**: p. 012094.

20. *Rigol DS2072A Digital Oscilloscope in UltraVision 2 Channel, 70 MHz 2GSa/s*. Rigol Technologies: Portland.
21. *JLC PCB*. [cited 2023; Available from: <https://jlcpcb.com>].
22. *KiCad EDA Software*. 2023. <https://www.kicad.org/>.
23. *Spaceloft: Flexible high performance insulation for building envelopes & equipment*. 2016, Aspen Aerogels. <https://www.aerogel.com/wp-content/uploads/2021/06/Spaceloft-Datasheet-English.pdf>.
24. MathWorks, *MATLAB*. 2023. <https://www.mathworks.com/products/matlab.html>.
25. Chen, C., *Evaluation of resistance–temperature calibration equations for NTC thermistors*. Measurement, 2009. **42**(7): p. 1103-1111.
26. *Gauge Blocks: 630.880*, in *Qualite: Extra 1991*, Carry Le Locle Suisse: Switzerland.
27. *Tolerances in  $\mu\text{m}$  of CARY gauge-blocks*. Cary Le Locle Swiss: Switzerland.
28. *Fluke 114, 115, 116 and 117 Digital Multimeters*. 2006, Fluke Corporation.
29. Tillaart, R., *INA226 Arduino Library*. 2023. <https://github.com/RobTillaart/INA226>.
30. *Micsig TO202A Oscilloscope*. 2023.
31. *JB4-HX711 Junction Board With Load Cell Amplifier*. 2022, Sensorcon.
32. Sencorcon, *SC134 Full Bridge Micro Load Cell*. 2018, Sensor and Control Co.,Ltd
33. Kalør, O., *HX711\_ADC Arduino Library*. 2023. [https://github.com/olkal/HX711\\_ADC](https://github.com/olkal/HX711_ADC).
34. *FLIR E75 Advanced Thermal Camera*. 2017, TELEDYNE FLIR.
35. Haven, O.C.H.t., *Development of an Affordable and Portable Guarded Hot Plate Apparatus for On-Site Thermal Conductivity Measurements of Fabrics in Forensic Investigations*. 2024.
36. Meier, R., *CoolTerm*. 2023. <https://freeware.the-meiers.org>.
37. *Sweet & Salt 100% Silicone Baking Sheet for Rolled Cake*. Casa International NV: Belgium. <https://ch.casashops.com/de/produkte/sweet-salt-fuer-gerollten-kuchen-594867/594867/>.
38. Unknown, *Bessel's Correction: Why Use N-1 For Variance/Standard Deviation?* 2023. <https://www.statisticshowto.com/bessels-correction/>.
39. *100% Cotton Blue Work Jacket*. KLM Kleding: Netherlands.
40. Panas, A.J., W. Stryczniewicz, and R. Szczepaniak, *Investigation of thermophysical properties of thin-layered paint*. Thermochemica Acta, 2018. **662**: p. 100-107.