2026 ASEE ANNUAL CONFERENCE June 21–24, 2026 | Charlotte Convention Center, NC
Host Sponsor UNC Charlotte
Paper ID #52987

BYOE: Thermal Contact Resistance Measurements Using a Guarded Heat Flow Meter

Prof. Blake Everett Johnson, University of Illinois at Urbana - Champaign

Dr. Blake Everett Johnson is a Teaching Associate Professor and instructional laboratory manager in the Department of Mechanical Science and Engineering at the University of Illinois Urbana-Champaign. His research interests include experimental fluid mechanics, measurement science, engineering education, engineering leadership, and professional identity development.

Parth Sandip Patil, University of Illinois at Urbana - Champaign

I am a Mechanical Engineer with experience in thermal sciences, semiconductor fabrication, and product development. I received my M.S. in Mechanical Engineering from the University of Illinois Urbana-Champaign in 2025 and my B.Tech. in Mechanical Engineering, with a minor in Industrial and Management Engineering, from the Indian Institute of Technology Kanpur in 2023. My interests include thermal management, simulation, and experimental validation.

BYOE: Thermal Contact Resistance Measurement using a lab-scale Guarded Heat Flow Meter Experimental Setup

Abstract

Wherever two solids meet, such as heat sinks on chips, bolted joints in engines, battery pack interfaces, and exchanger plates, thermal contact resistance (TCR) frequently dominates the thermal budget, as actual surfaces only touch at minuscule asperities with trapped gaps of air or filler between them. Its value is susceptible to material combinations, oxidation, cleanliness, flatness, hardness, clamping pressure, and the presence or thickness of thermal interface materials. Handbook values or purely analytical models may be orders of magnitude incorrect for a particular assembly due to the nonlinear interactions between all these elements. Therefore, under the precise conditions a design would encounter, lab measurement is the most dependable method of obtaining a usable TCR.

This Bring Your Own Experiment (BYOE) paper describes a modular, lab-scale implementation of the ASTM E1530 guarded heat-flow meter (GHFM) method. The modular architecture uses swappable components so instructors can change samples, Thermal Interface Materials (TIMs), and meter bar materials and lengths, while preserving the E1530 boundary conditions and quasi-1D guarded geometry. The students can simply expand experiments by sweeping pressure and different TIM and sample selections. This setup was built for an undergraduate heat transfer laboratory course.

The experimental setup measures one-dimensional heat transfer through a specimen under steady-state conditions using Fourier's law. A silicon wafer test sample is mounted in a stack between a heat source, two meter bars, and a heat sink. Thermistors are placed at specific locations in the meter bars to measure the thermal gradient in the meters bars. A surrounding guard region is held at nearly the same temperature as the meter path, suppressing lateral losses so the conduction is effectively 1D. With heat flux q known, the bulk of the specimen and the interface thermal resistance follow from $R=\Delta T/q$. To correlate the dependence of the TCR on clamping force, this setup includes an integrated load cell into the stack to measure and control the normal force at the interface. Students vary clamping pressure quantitatively, measure the TCR as a function of clamping pressure, and interpret results through both thermal and mechanical lenses (surface conformity, micro-contact area).

The experiment targets three learning outcomes: (i) translating the heat equation into experimental evidence through analysis of $T(x)$ profiles; (ii) understanding how assembly variables govern TCR, as well as when a TIM meaningfully alters performance; and (iii) practicing experimental judgment (steady-state detection, uncertainty budgeting, replication). Because the setup construction is transparent and inexpensive, the exercise avoids black-box measurement and foregrounds the craft that determines credibility.

This BYOE paper is written with a goal of allowing students to examine steady-state gradients, reproduce assembly procedures, assess the significance of measurement uncertainty, and provide data that directly informs design margins and safety considerations, while also validating models in a controlled environment.

Introduction:

In engineering laboratory courses, there is often a struggle to balance the need for predictable, structured instructional activities with the desire to cultivate genuine experimental reasoning. While traditional "recipe-style" laboratories can efficiently reinforce core concepts, their prescriptive nature may limit opportunities for students to engage in the iterative, judgment-laden aspects of experimental work that characterize professional engineering practice. When procedures are fixed and outcomes are known in advance, students may complete experiments successfully without developing competencies such as selecting appropriate measurement devices, recognizing sources of uncertainty, diagnosing unexpected behavior in equipment, or refining a protocol based on early observations. By providing students with laboratory experiences in which the experimental outcome is not predetermined and procedural choices matter, they gain the opportunity to transfer theoretical knowledge into applied settings, foster ownership of the investigative process, and develop engineering judgment.

The instructional team has developed a laboratory exercise that gives students a hands-on feel for thermal contact resistance and how heat really moves through solid–solid assemblies, not just neat textbook blocks. With the Guarded Heat Flow Meter (GHFM) meter bars, the system reaches a steady state and then allows the determination of temperature gradients as direct evidence of conduction and uses them to estimate heat flow through the stack. From there, students separate effects at interfaces and see why contact resistance often matters more than bulk heat conduction through materials. By changing clamping force with a load cell, they also relate solid mechanics concepts to heat transfer by observing a drop in thermal resistance as contact improves. Just as important, they learn the often-messy, practical side of experimentation, including difficulties with alignment, surface preparation, consistent thermal interface material (TIM) application, repeatable assembly, and data acquisition and logging, all while building good habits around reproducibility and uncertainty.

Concept

Thermal conduction is the most straightforward mechanism for heat transfer in solids. If one region of a material is warmer than another, energy naturally flows from the hot region to the cold one. At the scale used in heat transfer, Fourier's law captures this behavior by linking heat flux to the temperature gradient, with thermal conductivity describing how easily the material carries heat [1]. Under common engineering assumptions such as steady operation, constant properties, and no internal heat generation, the governing equation simplifies, and the temperature often changes nearly linearly along the direction of heat flow. This is also why the thermal resistance viewpoint is so convenient. For a given temperature difference, the heat rate depends on the material and geometry, where thicker paths resist heat flow, while larger area and higher conductivity make it easier for heat to pass.

Thermal contact resistance appears when heat must cross an interface between two solids. Surfaces that look smooth at the macroscopic scale are still rough under magnification. When two parts are pressed together, they touch only at scattered asperity spots, while the rest of the interface contains microscopic gaps that are often filled with air or other low-conductivity media. The interface, therefore, behaves like an additional series resistance and can cause a measurable temperature drop at the contact. This resistance is not a fixed material property. It depends on how the joint is assembled, including contact pressure, surface finish and flatness, cleanliness and oxidation, material compliance, and the presence and thickness of a thermal interface material [2,3].

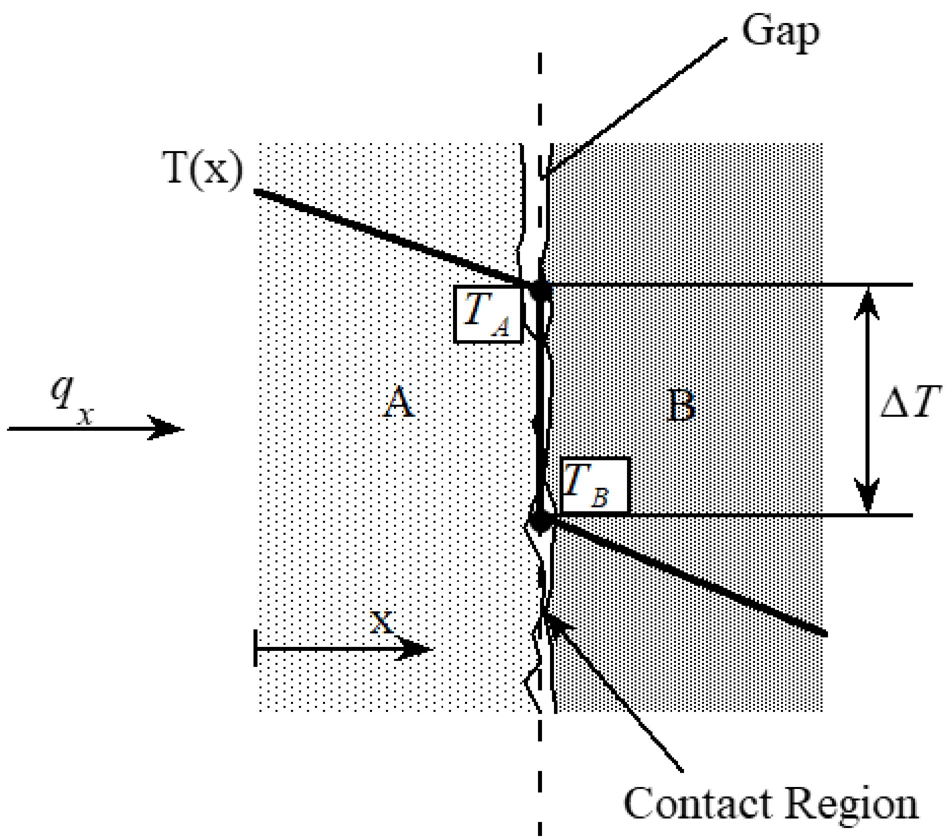


Figure 1. Detail of surface roughness effect on contact resistance

The contact resistance for a unit area of the interface is defined as the ratio of the temperature difference across the interface (driving potential) to the heat transfer rate as shown is Figure 1:

$$R_{t.c} = \frac{(\Delta T)_{interface}}{q_x A} = \frac{T_A(x_{interface}) - T_B(x_{interface})}{q_x A} \quad (1)$$

In this lab, we study the effect of normal stress by adding a load cell in series with the GHFM and adjusting a clamp to apply different compressive loads. Since the silicon wafer sits between two meter bars, there are two interfaces to account for. The thermal resistor model for this experiment is shown in Figure 2. We are primarily focusing on the region near the interface, so we have considered the two silicon-aluminum interfaces and the silicon bulk resistance. We will assume the interface between silicon and aluminum is identical on both the front and back sides of the wafer sample. For simplicity, we assume that the other contacts in the system are perfect and have no contact resistance.

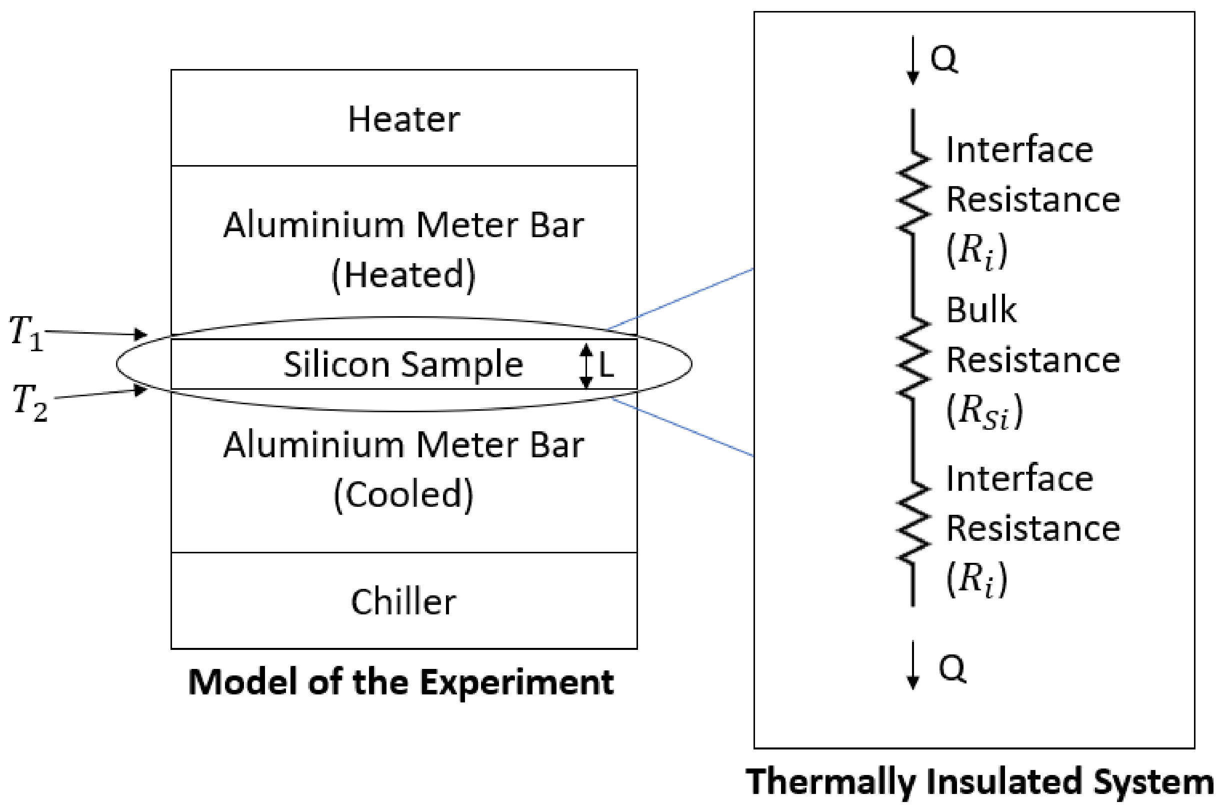


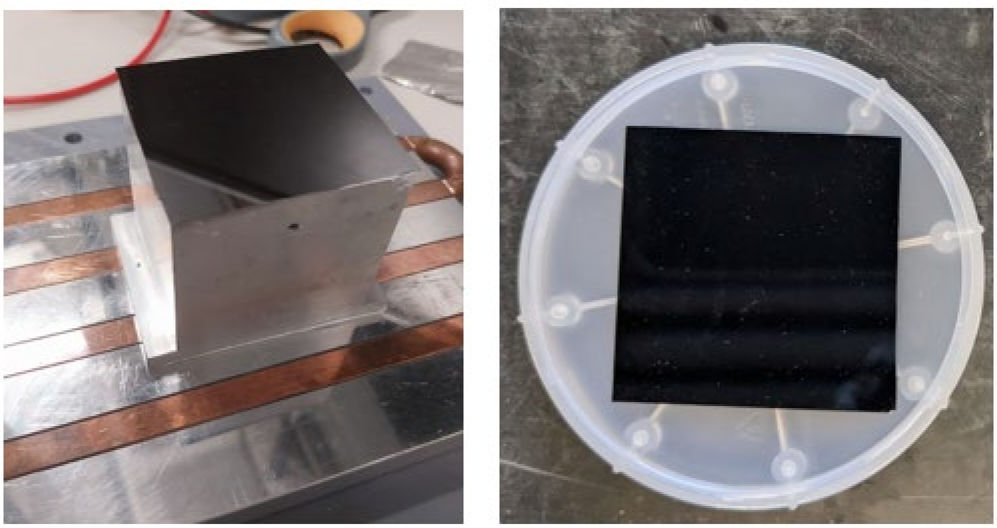
Figure 2. GHFM thermal-resistance network showing bulk layer resistances in series with aluminum–silicon interface resistances along the 1D heat-flow path.

Design of Experiment – First Iteration

The first version of the setup was built to demonstrate the core GHFM physics in a simple, classroom-ready form using readily available parts. It uses a vertical conduction stack powered by an external supply, with performance driven mainly by good insulation and by allowing the system to reach a steady state. A copper heater block with four cartridge heaters (heat source) sits above two aluminum meter bars, with a silicon wafer clamped between them [4]. The properties of the silicon sample are given in Table 1. It has an area of 50.8 mm x 50.8 mm and a thickness of 0.4 mm.

TABLE 1: Properties of Silicon Wafer, used as the sample for the experiment. (As given by the vendor).

Property	**Value**
Diameter	100 mm
Thickness	0.4 mm
Orientation	<1 0 0>
Doping Type	P/B
Grade	Prime
Resistivity	1 – 30 Ωcm
Polish	SSP
TTV	< 15 µm



a b

Figure 3. a. Silicon sample used for measurement, **b.** Close-up image of the sample.

The heaters are connected so that one pair of two resistors is connected in series, the other in series, and both pairs are then connected in parallel, as shown in Figure 4. Each cartridge heater is ≈ 47 Ω, and this arrangement will maintain the net resistance at ≈ 47 Ω. We chose four resistors to cover the entire length of the copper and aluminum meter bars, which have an area of 50.8 mm x 50.8 mm, allowing us to approximate one-dimensional conduction into the aluminum meter bar from the copper block. A single, localized heater would behave more like a line heat source, leading to lateral spreading and non-uniform heat flux, resulting in curved isotherms and multidimensional heat flow. The thickness of the meter bar is 50.8 mm. The CAD drawings of the components is attached in the Appendix along with the Bill of Materials (BOM).

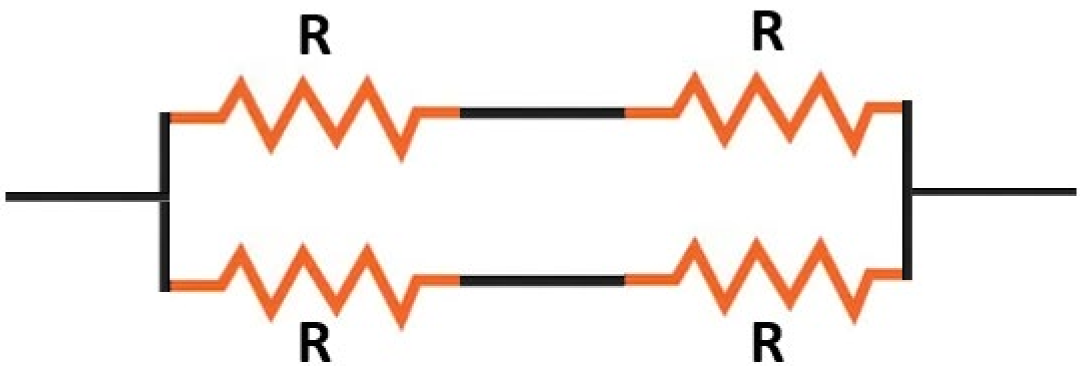


Figure 4. Cartridge heater connection for the GHFM setup.

A very thin layer of thermal paste is applied at the interfaces to maintain consistency while still allowing the contact resistance to be measured. We need to be careful not to apply too much TIM, as it can introduce thermal resistance of its own if the thickness becomes significant. Thermistors embedded in the meter bars record temperatures along the length so students can clearly see the steady axial gradients form on both sides of the interface and use that data for analysis within a single lab period. Figure 6 shows the different components of the first experimental setup in our lab.

The thermistors are mounted through 2 mm-diameter blind holes in the aluminum meter blocks. The thermally sensitive tips of the thermistors have a diameter of about 2 mm and are located at the central axis of the thermally sensitive electrical resistors at four locations in the meter bars. The positions of the hole centers are at 5 mm, 21 mm, 36 mm, and 49.5 mm. Note that the final hole is drilled extremely close to the edge of the block. The thermistor in that hole should be located nearest the silicon wafer for both meter blocks, as shown in Figure 5.

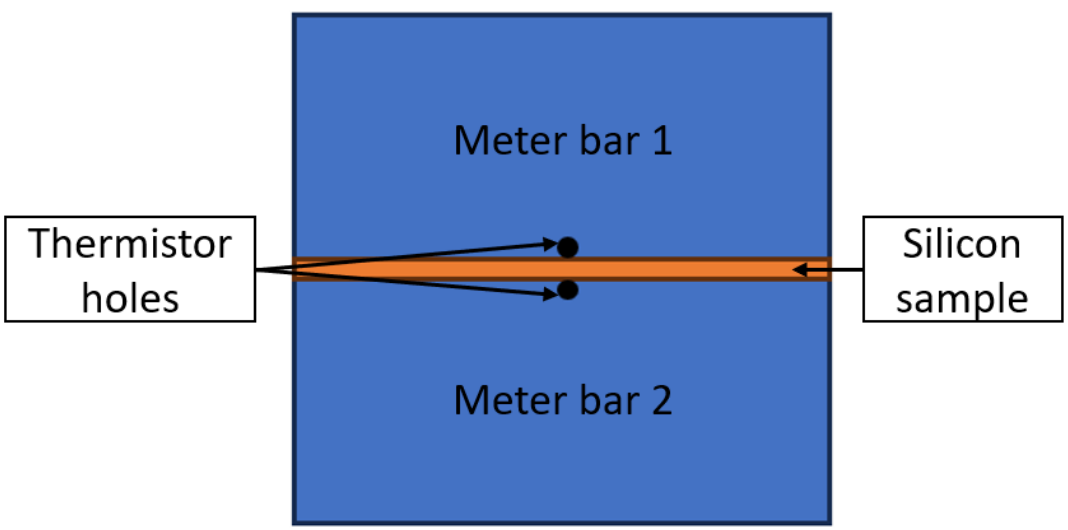


Figure 5. Placement of thermistors for measurement of temperatures close to the interfaces.

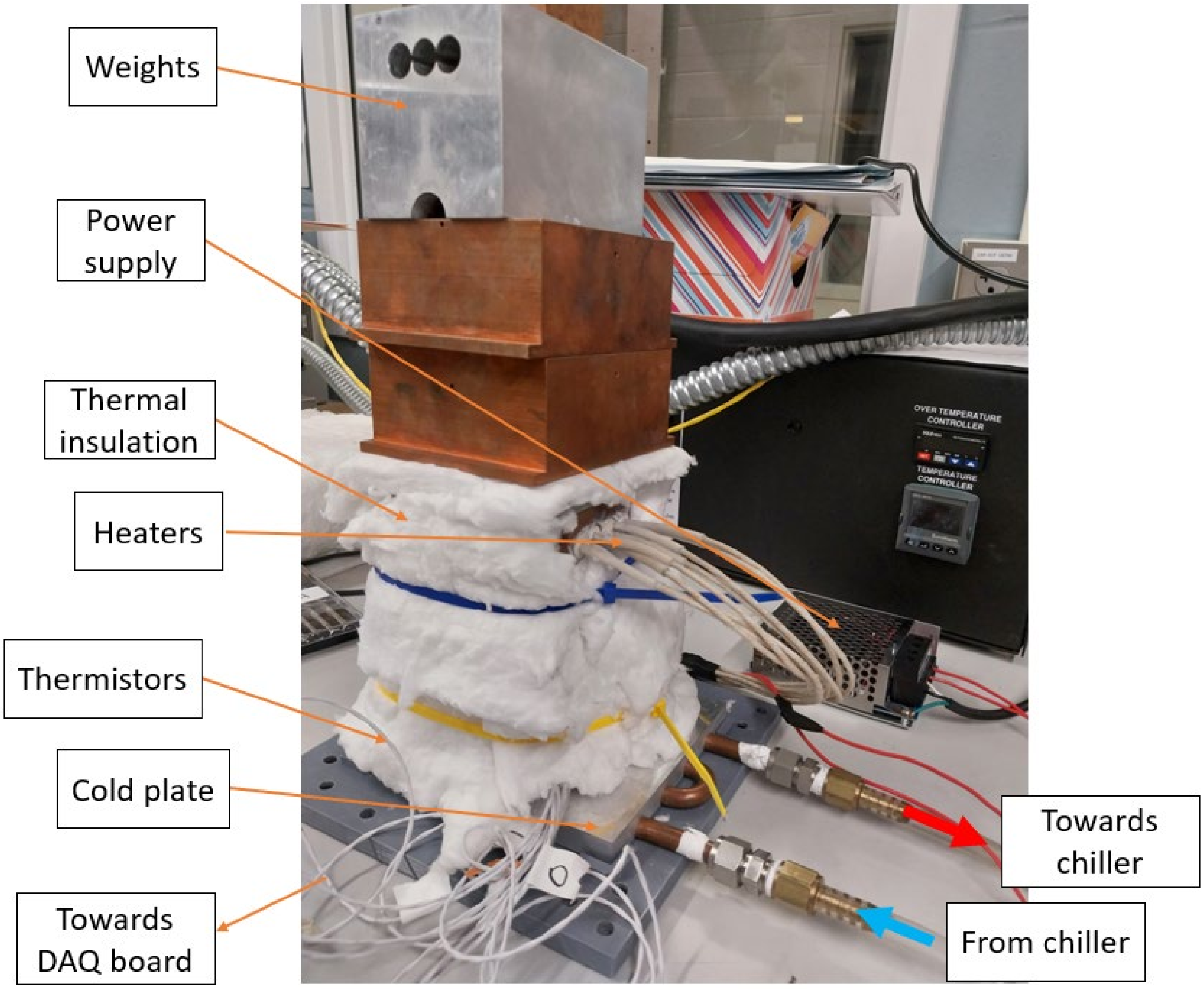


Figure 6. First Iteration of the GHFM Experimental Setup

In this setup, compressive loading was implemented in a quick-to-run format using known external weights, which can be varied to easily measure the pressure dependence of the contact resistance, allowing new users to explore without the risk of mechanical failure. To approximate one-dimensional conduction, the stack was wrapped in fiberglass insulation and secured with zip ties, with explicit attention to eliminating air gaps that would introduce convection and lateral heat loss. The cold boundary condition (heat sink) was provided by a cold plate with embedded tubing, connected to an external chiller that continually removed heat from the bottom of the stack to maintain a stable sink temperature. The fluid entering the cold plate was maintained at a constant 10 °C temperature. The resistance of the thermistors was measured using a Diligent USB-TEMP series data acquisition device, as shown in Figure 7a, and the data were recorded on a laboratory computer running InstaCal software. A total of eight 44008 series Negative Temperature Coefficient (NTC) thermistors with a resistance of 30000 Ω at 25 °C were connected to the DAQ [5], four on each side of the board. The thermistors are split between the meter bars, with four mounted on each of the upper and lower meter bars, as shown in Figure 7b.

NTC thermistors exhibit a decrease in resistance with increasing temperature, and InstaCal captures this change. A Python code was developed to interface with InstaCal. It is a software that connects to the DAQ system and converts the captured resistance readings into temperature values for each thermistor using each thermistor's Steinhart Coefficients. It takes the temperature readings every 9 seconds and stores the data in a CSV file. The correlation used in this conversion is as follows:

$$T = \frac{1}{A + B\ln(R) + C[\ln(R)]^3}, \tag{2}$$

where R is the resistance at a particular Temperature T in Kelvin, and A, B, C are the Steinhart Coefficients.

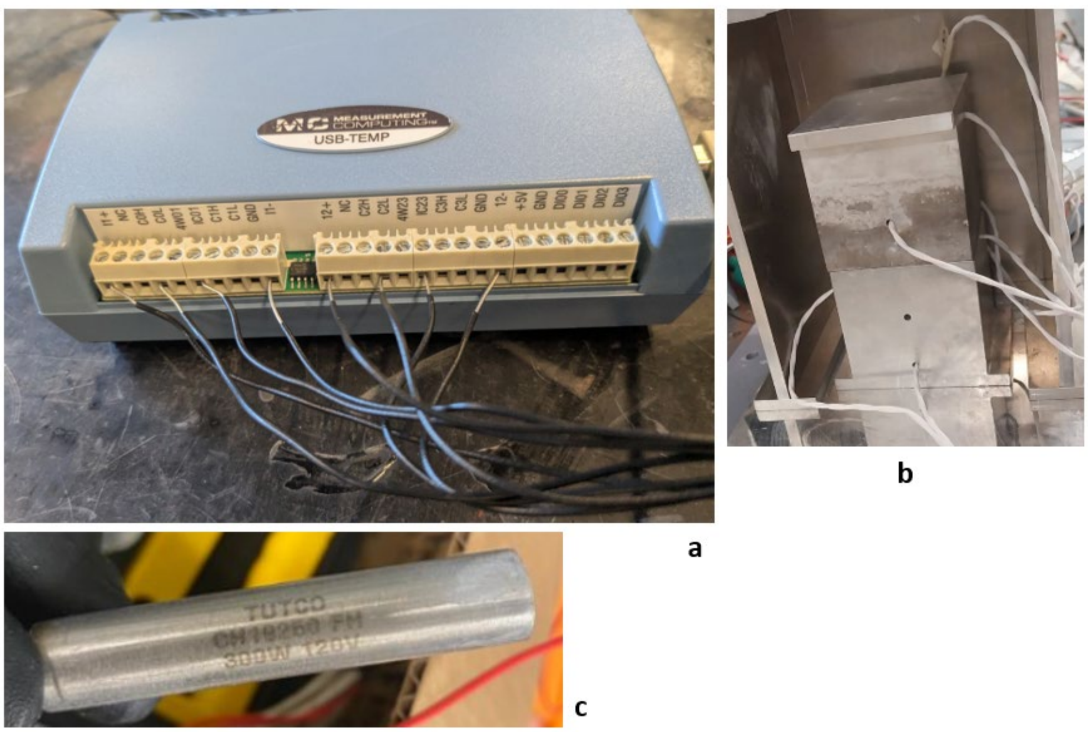


Figure 7. a. The Digilent USB-TEMP data acquisition device, **b.** Four thermistors on each meter bar, **c.** Tutco cartridge heater used as a heat source.

The total power applied to the four cartridge heaters is controlled through a variable autotransformer or "variac." To determine the power delivered to the heaters, we used the following relation:

$$P = \frac{V^2}{R}, \tag{3}$$

where V is the magnitude of the AC power voltage output from the variac, and R is the net resistance of the four heater resistors.

Improvements on the experimental setup

The final iteration was developed to make the experiment more repeatable, more lab-grade, and easier to duplicate at other institutions by reducing sensitivity to alignment and inconsistent loading. Mechanically, the biggest upgrade is the addition of an integrated loading path with a load cell and a handwheel-driven mechanism, and a rigid T-slotted aluminum frame for mechanical rigidity, so a normal force can be applied and held at a chosen value [6].

The team originally planned to use a load cell to measure the applied force in real time. They had purchased the MCL-02-1KN miniature compression load cell and the PSDS handheld load cell display from Load Cell Systems, as shown in Figures 8a and 8b. However, due to delays in receiving the load cells and display meters, a quicker solution was required, so a kitchen weighing scale was used to estimate the applied load for this experimental setup. The team plans to integrate the load cell system in a future version of the setup.

a b

Figure 8. a. Miniature compression load cell, **b.** Handheld display meter.

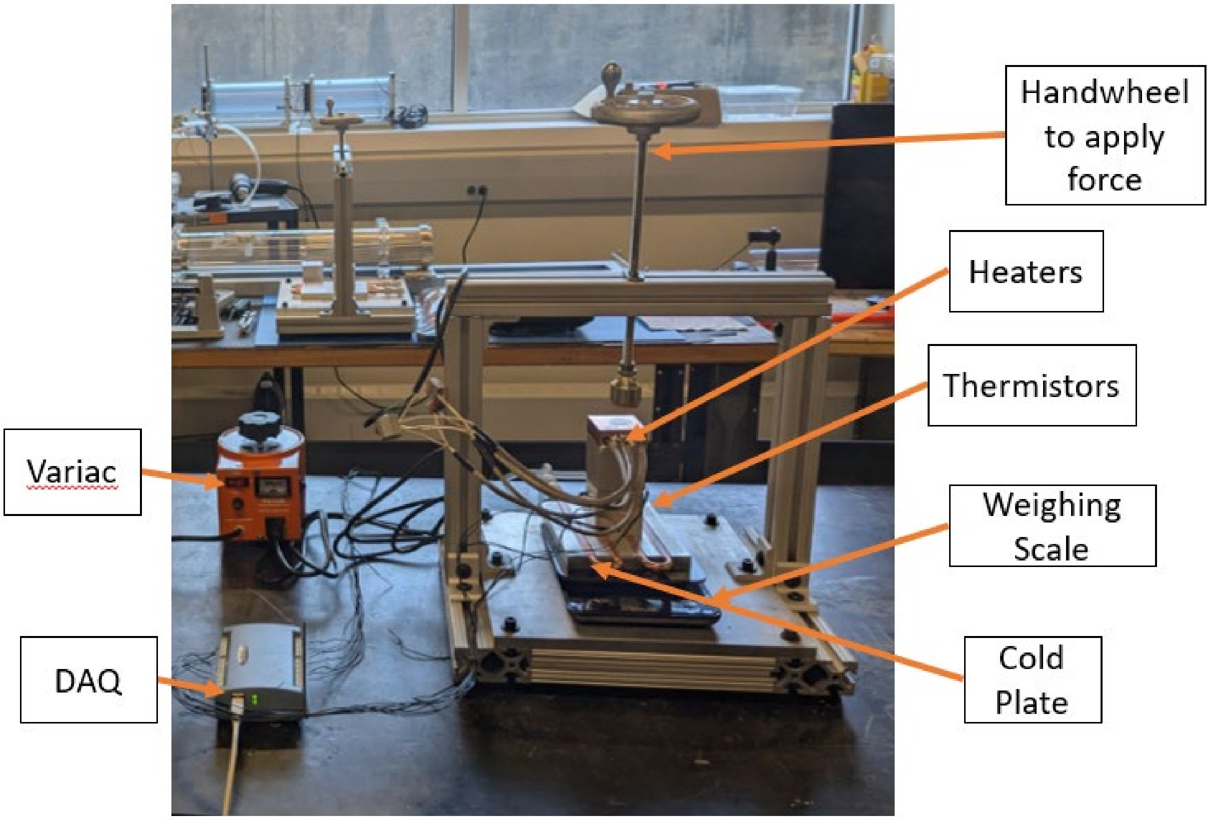


Figure 9. The modified GHFM setup.

Students apply pressure by turning the handwheel, and the load is recorded directly from the weighing scale, allowing it to be converted to contact pressure for pressure sweeps. The weighing scale is rated as a maximum load of 5 kg. The weighing scale is kept beneath the cold plate to measure the load applied by the handwheel assembly. The entire GHFM setup weighs just over 2 kg, as shown in Figure 10a. Therefore, to avoid overloading and damaging the scale, no more than a 29.4 N clamping force should be applied. Loads of 4.45 N, 13.35 N, or 22.25 N are advised for this investigation.

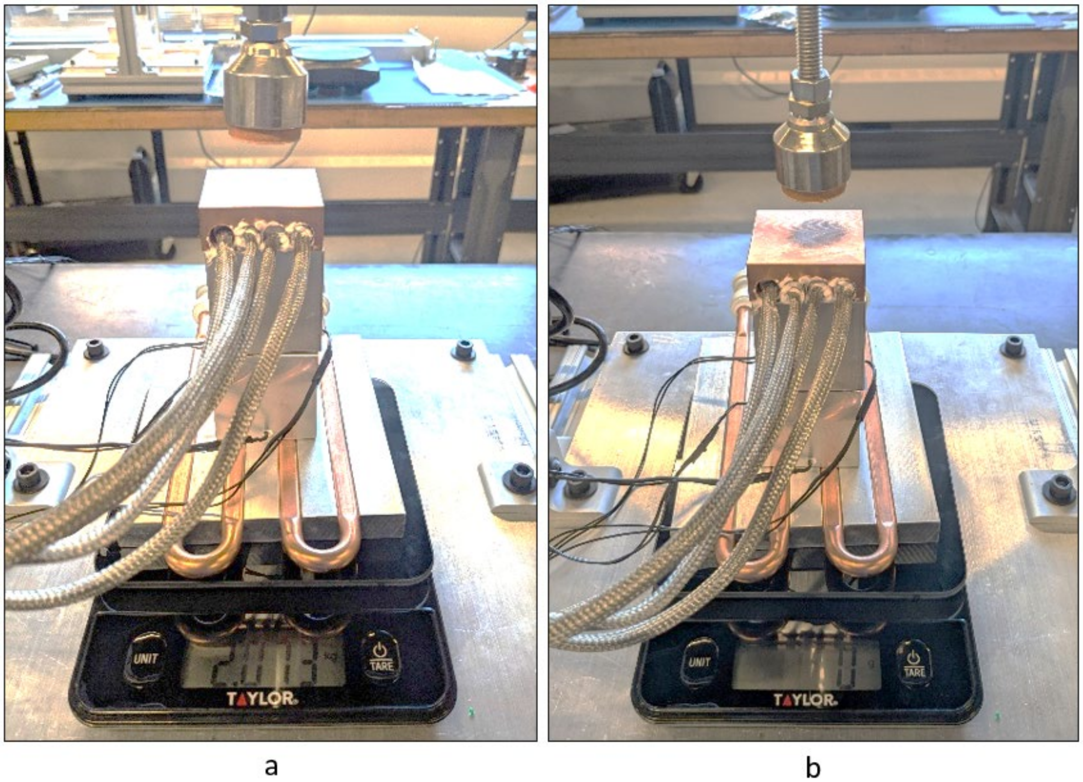


Figure 10. Taring of the scale to accurately determine the compressive load applied to the GHFM. **a.** Scale is not tared. **b.** Scale is tared.

The dimensions of the aluminum meter bars, the copper heater bar and the silicon sample were kept the same. The remaining components (thermal paste, thermistors, insulation, InstaCal & Python code) and the chiller's temperature were also kept the same.

The fixture uses a rigid T-slotted aluminum extrusion frame to create a compact gantry over the GHFM stack. A double-rail base, having two parallel extrusions per side, supports the aluminum base plate and provides a wide, stable footprint. Two uprights bolt to the base with gusseted corner brackets, and a doubled top cross-member ties them together to stiffen the frame in the loading direction. The load train is centered on this top member: a handwheel turns a vertical threaded rod that passes through the cross-member, pushing down through the load cell and compression hardware into the stack. The open-frame design keeps the stack easy to access for assembly, thermistor wiring, and chiller tubing, and the modular T-slot hardware lets us adjust heights, shift parts around, or swap plates without rebuilding the structure.

To perform the deflection analysis on the fixture, we treated the top horizontal member as a beam under a central point load and estimated its elastic deflection using standard beam-bending relations (equation 4) with the aluminum modulus (about 69.9 GPa) and the member's second moment of area based on its cross-section. Using a target maximum interface load of around 22.25 N and the beam-clamped-at-two-ends correlation with center load application, the maximum deflection of the beam is ≈ 0.27 μm.

$$y_{max} = \frac{-FL^3}{192EI} \quad (4)$$

This small deflection confirms that the frame remains effectively rigid during loading, helping maintain consistent contact conditions and minimizing measurement error from mechanical bending.

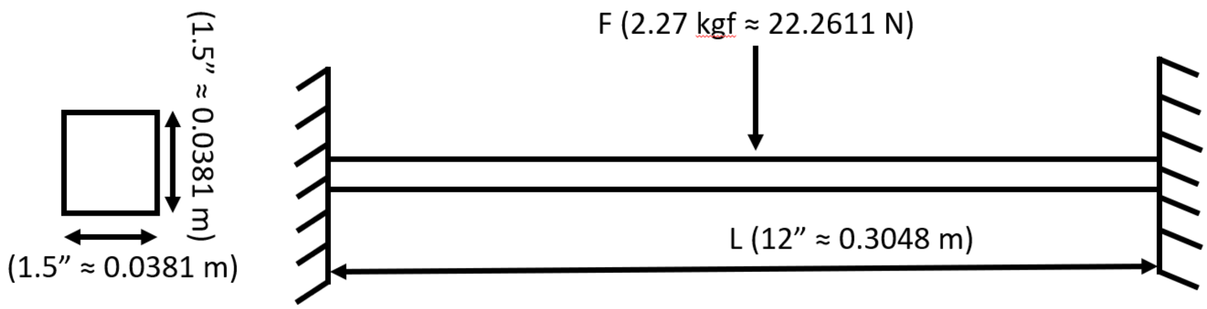


Figure 11. Beam idealization for frame deflection analysis. The top cross-member is modelled as a fixed–fixed beam with a central maximum load to estimate worst-case elastic deflection.

Conceptually, the final iteration makes the thermal–mechanical connection explicit: increasing normal stress increases real contact area by flattening asperities, thereby reducing the interfacial contribution to the overall resistance.

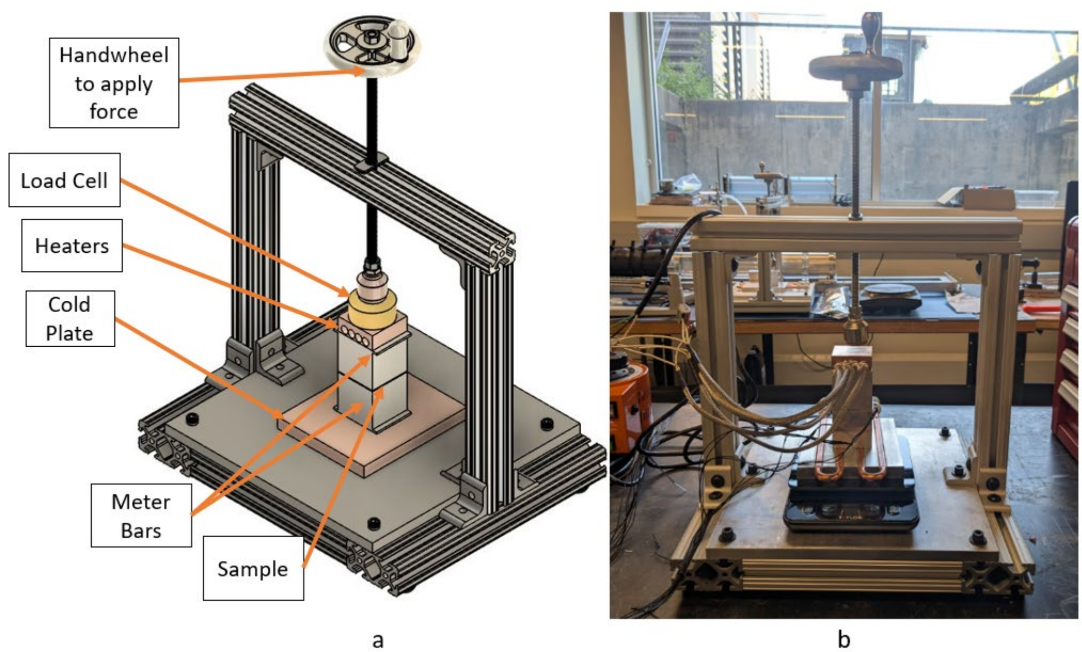


Figure 12. A. Isometric CAD model of the new GHFM setup (with load cell). **B.** Actual modified experimental setup (with weighing scale).

Procedure

- Confirm all components are present. Place the two meter bars and the sample on the bench and verify the thermistor-hole spacing from a chosen reference side (heater or cold plate).
- Start the chiller and let the temperature reach a steady state for 15 to 20 minutes.
- Wipe the contact parts of the cold plate, meter bars, sample, heater block, and heaters with acetone if needed. Clean the silicon using acetone → IPA → distilled water → IPA →nitrogen dry and handle only with tweezers.
- Add a thin layer on every contact surface (cold plate/bar, bar/sample, sample/bar, bar/copper block), plus thermistor beads and heater surfaces.
- Assemble the stack (Cold plate →lower meter bar →silicon sample →upper meter bar →Heater block). Keep everything aligned and orient the meter bars so the holes closest to the surface face the sample.
- Insert the T0 – T3 greased thermistors beads in the upper bar and T4 – T7 beads in the lower bar.
- Grease and insert the cartridge heaters into the holes of the heater block and wrap the stack with glass wool (including the top of the copper block). Avoid air pockets, and secure with zip ties without shifting alignment or pulling thermistors out.
- Place the load cell on top, connect to the display, and gently turn the handwheel to reach the target force.
- Plug thermistors into the USB-TEMP DAQ, start InstaCal/Python logging (InstaCal instructions are in Appendix C), and do a quick sanity check: temperatures should be lowest near the cold plate and increase toward the heaters.
- Set the variac/voltage regulator to produce 60 V or less. Once the output stabilizes, start the Python logging and monitor the live temperature plot. Let the system run until it reaches steady state (typically 35–40 minutes).
- When steady state is reached, stop the Python script. Turn the variac down to zero and switch off the heaters' power. Keep the chiller running to cool down the setup.
- Once temperatures have dropped close to room temperature, reapply the desired load/pressure and repeat the run for the next condition (e.g., a different pressure, TIM, or sample).

Data Analysis & Measurement

Before handing the final GHFM experiment to the students, we performed a trial run with many different voltage values to confirm that the wiring, insulation, loading, and data-acquisition workflow behaved as expected and that the stack could reliably reach steady conditions. Once logging begins, thermistor temperatures are read from the CSV file and plotted against iteration number (since the logger samples at a fixed interval, iterations are proportional to time). Plotting all channels on the same axes serves as a quick sanity check: each trace should rise smoothly and then level off, with no sudden drops (loose wiring), oscillations (unstable power input), or other nonphysical behavior. Here, we show the data for the 30 V output from the variac, which yields a power of 19.03 W.

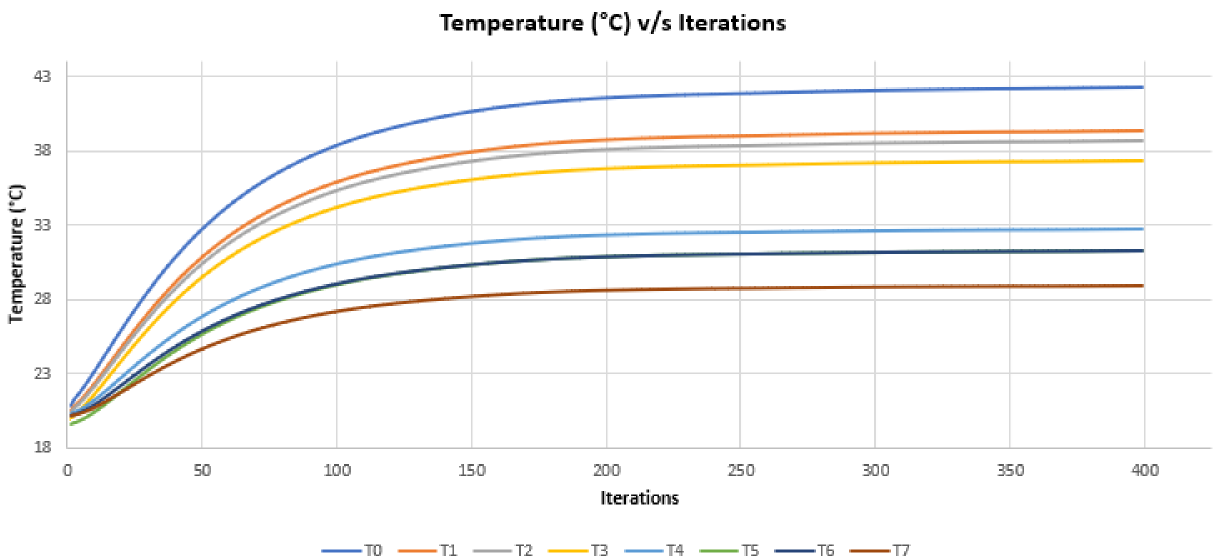


Figure 13. Temperature response of all thermistors (T0–T7) during a GHFM trial run (30 V) plotted versus data-logging iterations.

To identify when the steady state begins, we first use the practical check, where the curves become essentially flat, and then support that decision with a simple first-order transient fit using the following equation:

$$T(t) = T_{\infty} + (T_0 - T_{\infty})e^{-t/\tau} \tag{5}$$

where, T_{∞} is the average steady (final) temperature for that thermistor, T_0 is the initial temperature when heating begins and τ is the effective time constant. Treating the warm-up as a single dominant thermal mode is reasonable here because the heater block and meter bars are highly conductive solids, and the assembly is heavily insulated, so the transient is mainly governed by the stack's effective thermal capacitance and the overall resistance from heater to chiller. Consistent with this, the measured temperature responses are smooth, monotonic, and asymptotic, without overshoot or multi-stage kinks that would demand a higher-order model. Rather than fitting every channel, we apply the exponential fit to a representative set of thermistors near the heater, near the chiller, and close to the specimen/interface region, since these locations bracket the thermal path and are most likely to reveal the slowest settling behavior.

So, once we get the slope and the data values from the thermistors by solving equation 5, as shown in Figures 14 and 15. The slope is the negative inverse of the time constant; five times the time constant means the system has reached steady state. (> 99.32%)

We then adopt a conservative steady-state criterion based on the largest settling time among the key thermistors and use it as the starting point for extracting meter-bar gradients and performing resistance and thermal-conductivity calculations. This approach keeps the procedure simple for students while providing a repeatable basis for deciding when the system has truly stabilized.

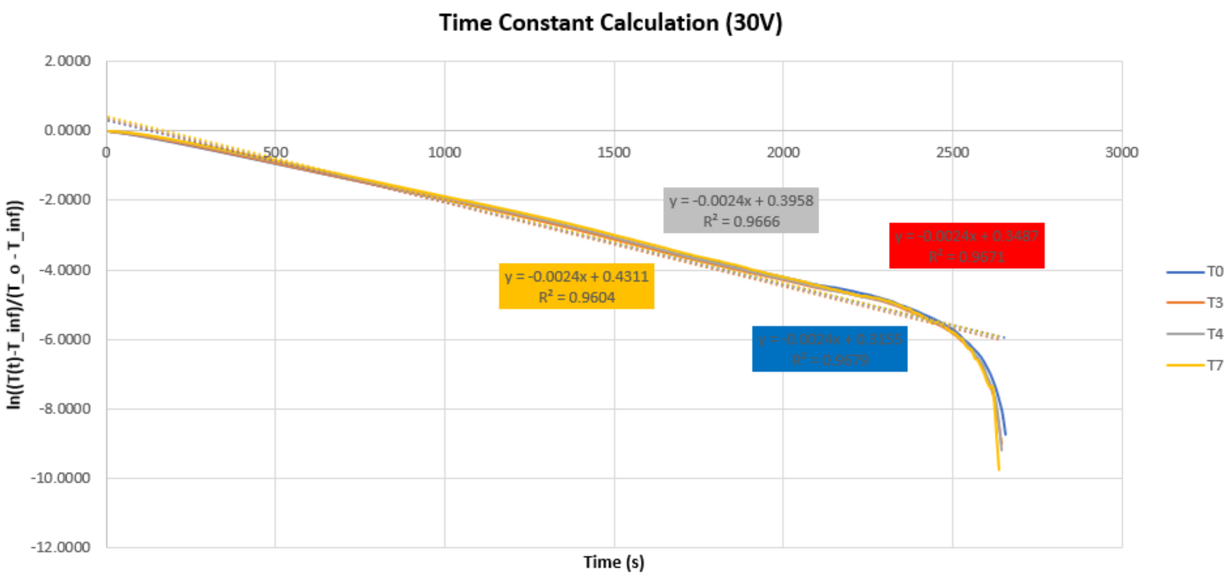


Figure 14. Time-constant estimation for a 30 V heating trial using a first-order exponential model.

Run	Thermistor	T_inf	T_o	T_o - T_inf	Slope (-1/tau)	Tau (sec)	5*Tau (sec)	R^2	Notes
1 (19.03 W)	T0	42.02	20.78	-21.24	-0.0024	416.67	2083.33	0.968	Closest to heater
	T3	37.11	19.91	-17.20	-0.0024	416.67	2083.33	0.967	Closest to chiller
	T4	32.63	20.22	-12.41	-0.0024	416.67	2083.33	0.967	Immediately on top of sample
	T7	28.77	20.15	-8.62	-0.0024	416.67	2083.33	0.960	Immeidiately to the bottom of sample

Figure 15. Summary of first-order time-constant calculations for the 30 V trial. For selected thermistors (T0, T3, T4, and T7), the table lists T_{∞}, T_0, fitted slope, resulting τ, the conservative settling time 5τ, and fit quality (R^2), along with sensor locations.

Once the steady-state start time is identified, we use only the data after that point to construct the temperature vs. distance profiles along the two meter bars, as illustrated in Figure 16. The linear fits to these steady state gradients provide the core quantities needed for our resistance analysis using equations 6 and 7, which are the Ohm's law analogue for heat conduction and the parallel wall resistance for conduction, respectively:

$$\Delta T = QR \tag{6}$$

$$R_{cond} = \frac{L}{KA} \tag{7}$$

Students can first compute the effective thermal conductivity of the aluminum meter bars and then estimate the thermal conductivity of the silicon specimen under an initial simplifying assumption that the interfaces contribute negligible resistance.

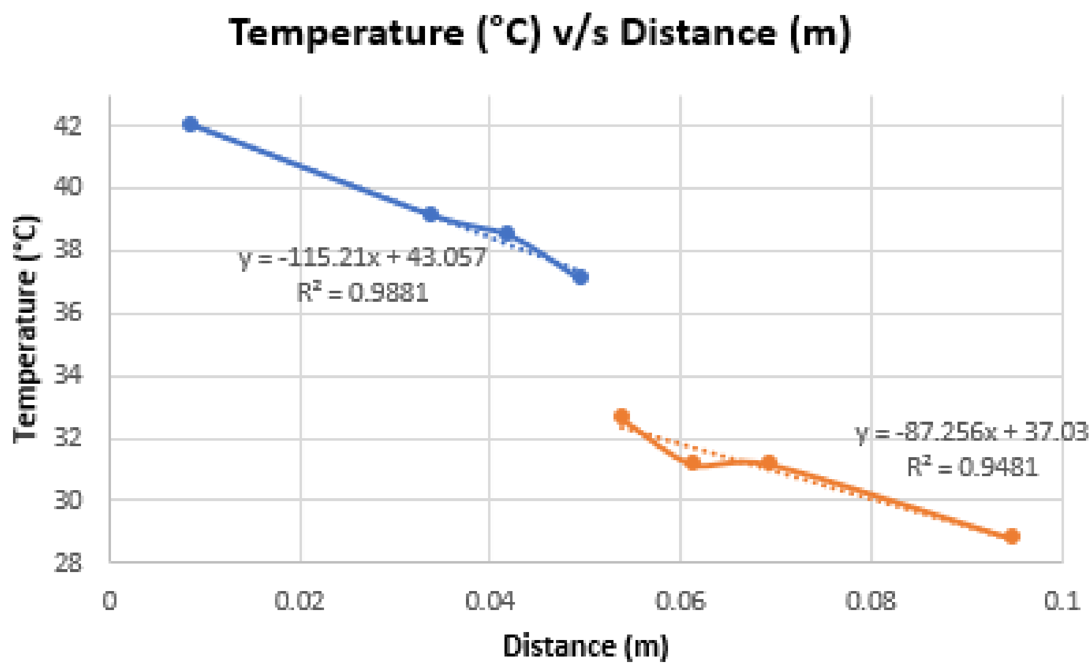


Figure 16. Steady-state temperature–distance profiles for the upper and lower meter bars with linear fits used to extract bar gradients and heat flux.

In that first-pass calculation, the silicon conductivity is obtained from the temperature drop between the two thermistors closest to the sample, on either side; as expected, the resulting value typically deviates from literature values because ignoring thermal contact resistance incorrectly attributes some of the interfacial temperature drop to the silicon. Once students move beyond the ideal interface assumption, they instead treat the silicon conductivity as known (from literature) and use the resistance-network picture (Figure 2) to back out the thermal contact resistance from the measured interface temperature drop and the heat rate. With this workflow established, students can independently explore how contact resistance and heat-flow assumptions change as they sweep applied pressure, add or remove insulation (to test 1D conduction versus heat losses), vary heater power with the variac, and compare runs with and without TIM.

Conclusion

In conclusion, this GHFM activity provides a practical, repeatable way for students to move from ideal conduction theory to the realities of real assemblies, where interfaces and contact quality can dominate thermal performance. Using a guarded, quasi-1D heat-flow path with instrumented meter bars, students extract steady-state gradients as direct evidence of conduction and then separate bulk resistance from interfacial effects to quantify thermal contact resistance.

The modular design and integrated load cell make the experiment easy to extend, and students can systematically sweep clamping pressure, compare TIM choices, and evaluate how assembly decisions change measured resistance, while also reinforcing good experimental habits such as steady-state detection, repeatable assembly, and uncertainty awareness.

Ultimately, the setup is intended to be transparent and adaptable, giving instructors a standards-aligned pathway to teach conduction and contact resistance using data directly relevant to design margins and real thermal hardware.

Acknowledgments

We thank Ms. Asena Gelisli at the University of Illinois Urbana-Champaign for developing the initial Python code used to read temperature data from InstaCal and Dr. Youngmun Lee for modifying it to be used in Jupyter Notebook, and all the graduate teaching assistants of the course, who helped run the lab. Financial support from the Department of Mechanical Science and Engineering at the University of Illinois Urbana-Champaign is gratefully acknowledged.

References

[1] Bergman, T. L., Lavine, A. S., Bergman, T. L., Incropera, F. P., and DeWitt, D. P., Fundamentals of Heat and Mass Transfer, 7th edition, Wiley, New York, 2011.

[2] D. Fustinoni, L. Vitali, P. Gramazio, and A. Niro, “Accurate contact resistance characterization for thermal conductivity measurement with the heat flow meter method,” J. Phys.: Conf. Ser., vol. 1868, Art. no. 012005, 2021.

[3] K. Yao, Y. Liu, Y. Zhou, X. Wang, Q. Wang, Y. Ma, and C. Yao, “Influence of thermal contact resistance on thermal conductivity measurement with a high-temperature guarded hot plate apparatus,” Int. J. Thermophys., vol. 41, no. 18, Jan. 2020.

[4] ASTM E1530-19, Standard Test Method for Evaluating the Resistance to Thermal Transmission by the Guarded Heat Flow Meter Technique, ASTM International, West Conshohocken, PA, USA, 2019.

[5] DwyerOmega, “Thermistor Elements — 44000 Series,” product specification sheet (PDF), accessed Jan. 20, 2026.

[6] Elkholy, A., & Kempers, R. (2022). An accurate steady-state approach forcharacterizing the thermal conductivity of Additively manufactured polymer composites. Case Studies in Thermal Engineering, 31, 101829.

Appendix

A. Bill of Materials (BOM):

This BOM is prepared for a single GHFM experimental setup.

Material Description	Quantity	Cost/unit	Total Cost	Company
Precision Thermistor Elements 44008	8	$29.74	$237.92	Dwyer Omega
Fiberglass Insulation Sheet	1	$25.16	$25.16	McMaster-Carr
ATS-TCP-1001 Cold Plates	1	$118.05	$118.05	Digikey
VEVOR Auto Variable Voltage Transformer 500 VA, 3.84 A	1	$57.17	$57.17	Amazon
Food Scale	1	$19.99	$19.99	Target
Super Lube Heat Sink Thermally Conductive Compound 98003	1	$14.50	$14.50	Grainger
CH43671 Circular Cartridge Heaters	4	$28.42	$113.68	Tutco
High Strength Steel Threaded Rod, Zinc Yellow Chromate Plated, 3/8”-16 Thread Size, 1 Foot Long	1	$14.11	$14.11	McMaster-Carr
Aluminum Hub Hand Wheel, 4-spoke flat wheel with stationary handle, 4” Dia	1	$52.22	$52.22	McMaster-Carr
Vibration Damping Swivel Levelling Mount with brown cushion, 3/8”-16 Threaded hole	1	$23.04	$23.04	Mcmaster-Carr
MCC USB-TEMP & TC Series DAQ Device	1	$419.00	$419.00	Digilent
Screw-on round bumper, 1.5” High Single Rail, 1.5” OD	4	$5.60	$22.40	McMaster-Carr
6061 Aluminum Bar, 1/2” x 12” Wide, 1 ft Length	1	$68.91	$68.91	McMaster-Carr
6061 Aluminum Bar, 2” Thick, 2” Wide, 1 ft Length	1	$44.23	$44.23	McMaster-Carr
110 Copper Bar, 3/4” Thick, 2” Wide, 1/2 ft Length	1	$77.58	$77.58	McMaster-Carr
Single 4-Slot Rail, Silver, 1.5” High x 1.5” Wide, Solid, 15 in Length	2	$20.25	$40.50	McMaster-Carr
Single 4-Slot Rail, Silver, 1.5” High x 1.5” Wide, Solid, 1 ft Length	1	$16.20	$16.20	McMaster-Carr
Double Six Slot Rail, Silver, 3” High x 1-1/2” Wide, Solid, 9 in Length	2	$19.35	$38.70	McMaster-Carr
Double Six Slot Rail, Silver, 3” High x 1-1/2” Wide, Solid, 1 ft Length	2	$25.80	$51.60	McMaster-Carr
Silver corner bracket, 1.5” long for 1.5” high rail T-slotted framing	12	$8.80	$105.60	McMaster-Carr

Steel Low-Profile Narrow Base Weld Nuts, 3/8"-16 Thread Size	1 Pack	$8.20	$8.20	McMaster-Carr
18-8 Stainless Steel Socket Head Screw, 6-32 Thread Size, 2" long, fully threaded	1 Pack	$4.96	$4.96	McMaster-Carr
Black-Oxide Alloy Steel Socket Head Screw, 5/16"-18 Thread Size, 1" Long	1 Pack	$15.30	$15.30	McMaster-Carr
Black-Oxide Alloy Steel Socket Head Screw, 5/16"-18 Thread Size, 1-1/4" Long	1 Pack	$16.42	$16.42	McMaster-Carr
Zinc-Plated Medium-Strength Steel Hex Nut, SAE Grade 5, 3/8"-16 Thread Size	1 Pack	$9.48	$9.48	McMaster-Carr
End-Feed Nut and 5/16"-18 Button Head, 1.5" Rail, Black		$3.56		McMaster-Carr
P/B Type, Prime <1 0 0> Silicon Wafer, 400 µm thick, Resistivity: 1 – 30, SSP finish, TTV < 15 um, 100 mm dia.	1 Wafer	$15.00	$15.00	Ultrasil LLC

B. CAD Drawing of the design

The aluminum meter bars and copper blocks were designed and fabricated in-house. The T-slotted frame and weld nuts were modified to match the frame design of the modified experimental setup.

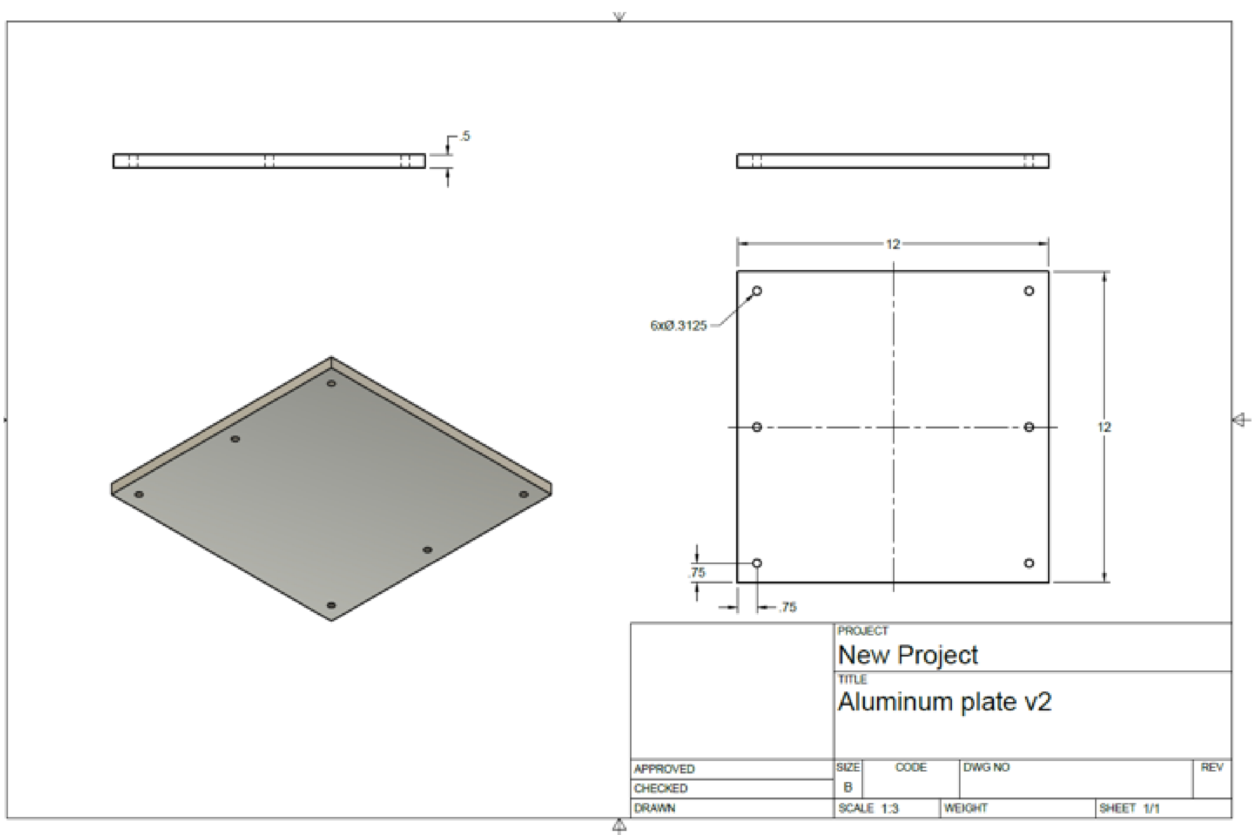


Figure B.1: Aluminum base plate providing a rigid mounting surface and fastening points for the GHFM assembly.

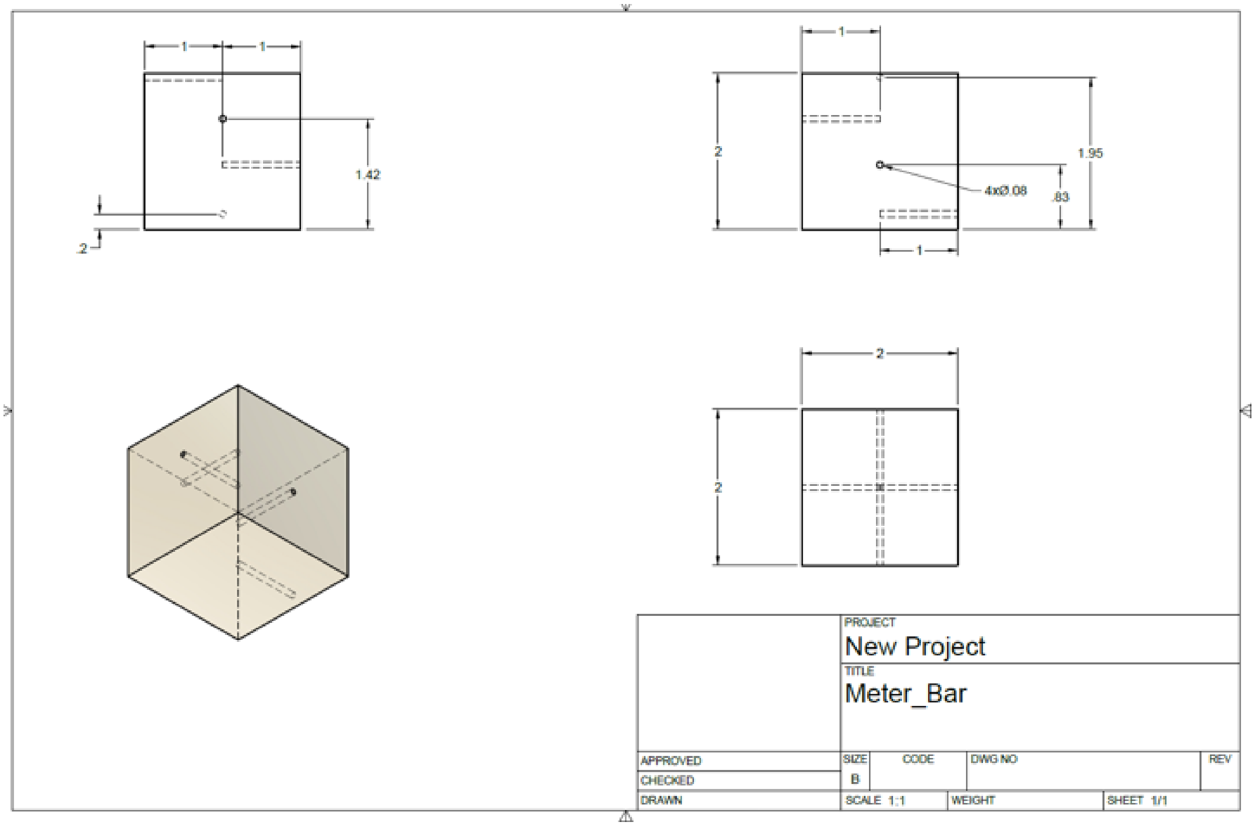


Figure B.2: GHFM meter bar with machined thermistor ports for measuring the 1-D temperature gradients.

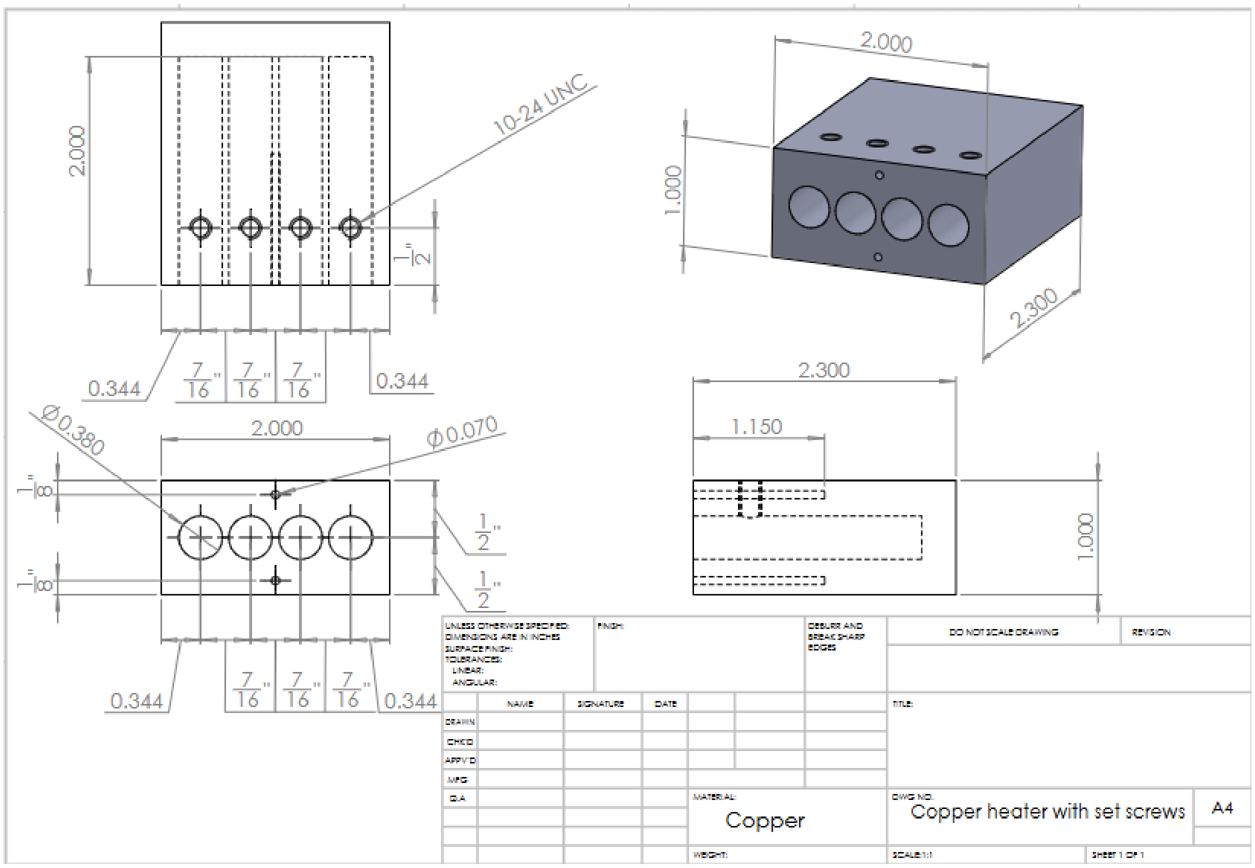


Figure B.3: Copper heater block housing cartridge heaters for controlled heat input.

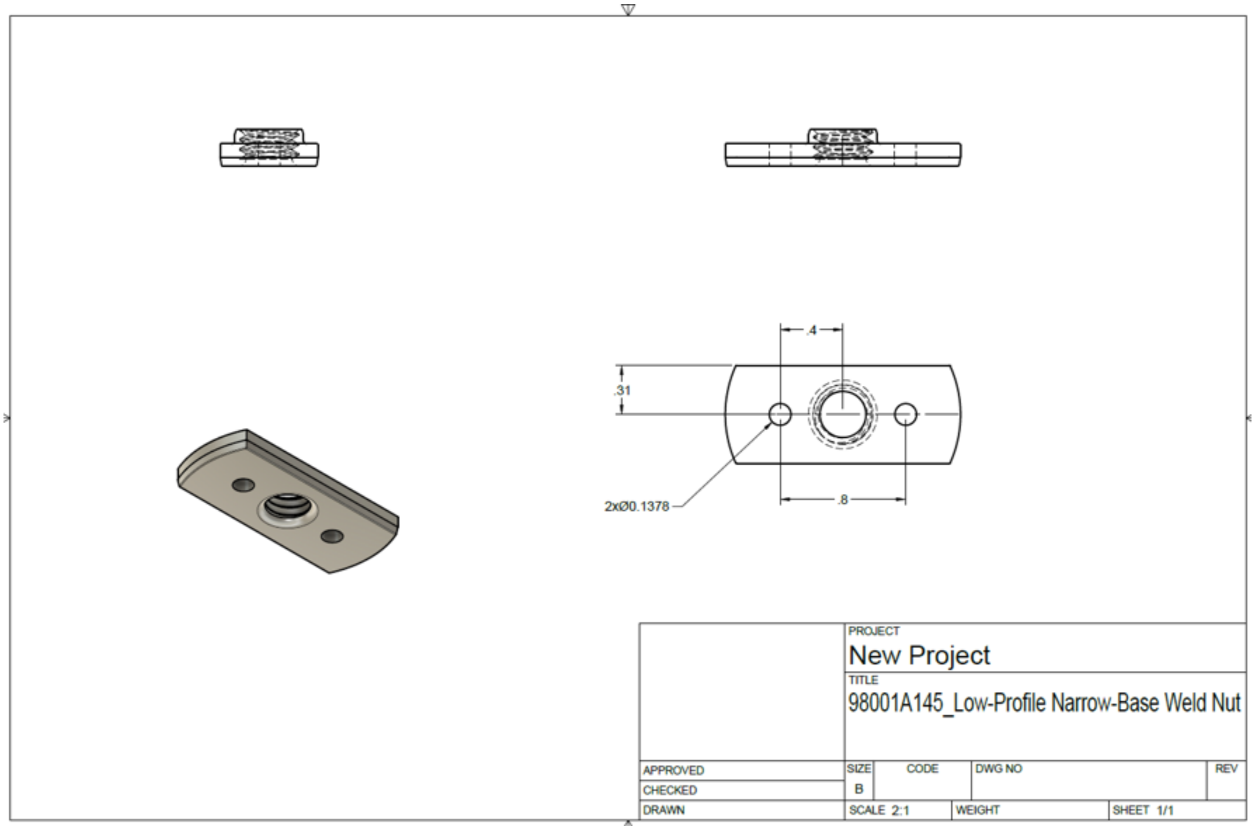


Figure B.4: Weld Nut used for locking the transverse motion of the threaded rod used for applying controlled pressure.

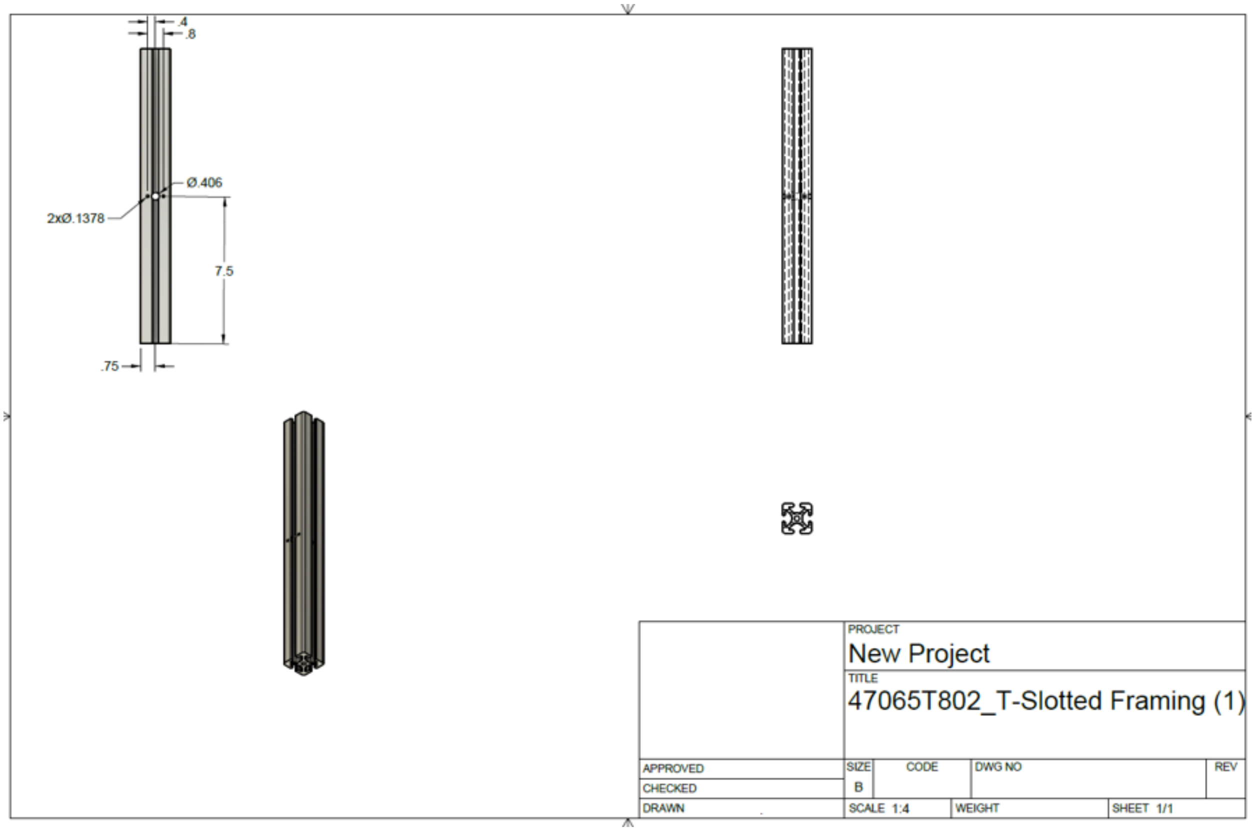


Figure B.5: T-slotted frame used as the top part of the GHFM setup frame.

C. Software Installation Guide & Python Code

DAQ (Digilent USB-TC or USB-Temp) software is only available on Windows OS. You should install InstaCal and Python Jupyter Notebook, along with the necessary libraries, to run and record data correctly.

1. *Program Installation:*
   a. *InstaCal*
   InstaCal is free software that is compatible with Digilent DAQ.
   Here is the link for download:
   https://digilent.com/reference/software/instacal/start?srsltid=AfmBOorMdOWYu3YSn8uu3RMWu-2l35_iVEW-Td8h1iNG55Mf6lduschR
   Once you open InstaCal, you can connect the Digilent DAQ.

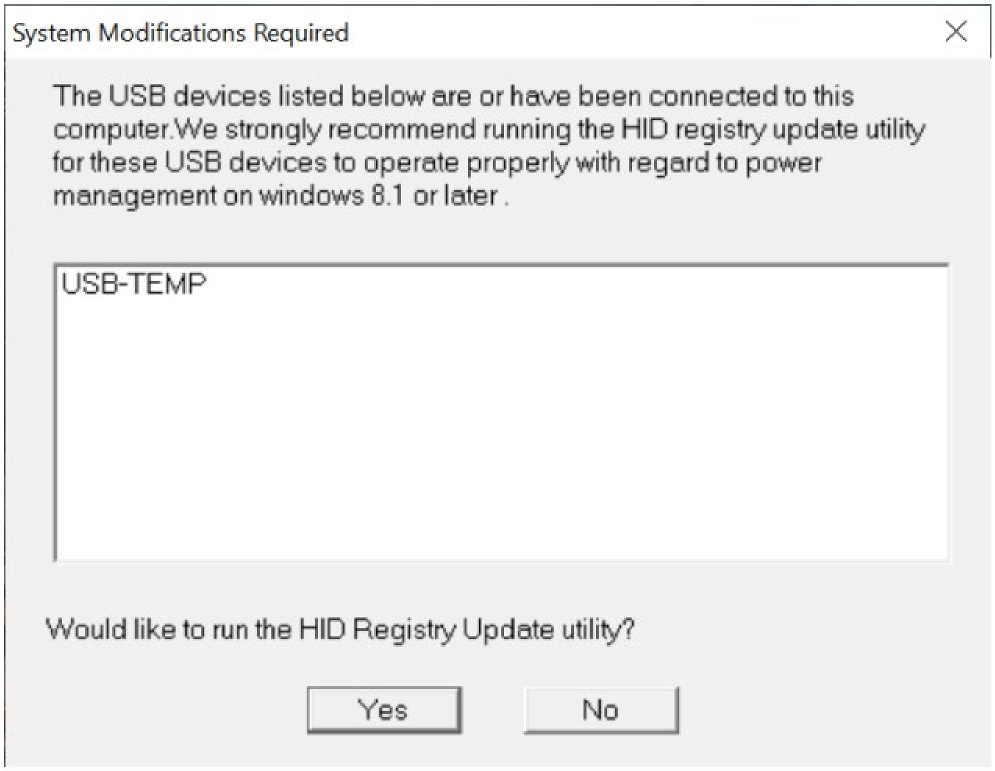


Figure C.1

Then you will see the image above. Click "Yes" and update HID.

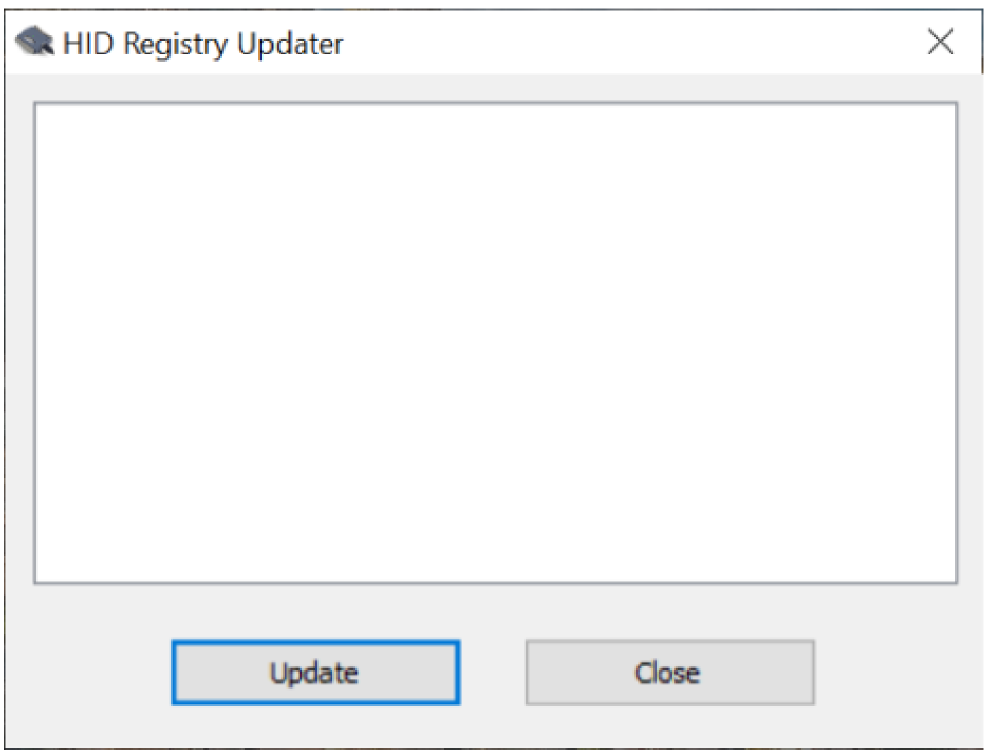


Figure C.2

Usually, it is recommended to click “Yes” for update, but ME320 computers do not allow us to access as Admin. Thus, we cannot install anything. Therefore, we recommend that you click “No” when prompted by the HID Registry Update utility. In general, it will work for the next procedure as below. Then, you will see which type of device will be connected to the computer. Now, for the thermistor, we will use USB-TEMP and click “OK”.

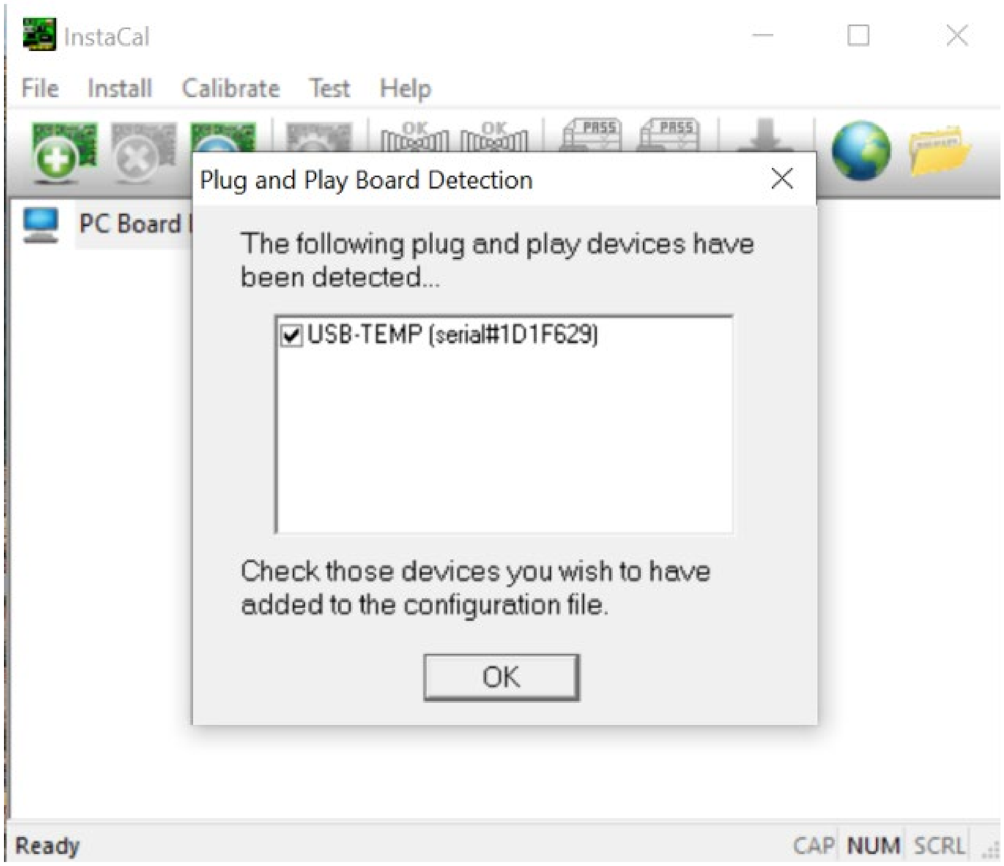


Figure C.3

Once you see your connected DAQ (USB-TEMP), you can double-click it and check the board configuration as shown below. To use a thermistor, check that all channels you want to use are properly configured for the thermistor and the connection type. Here, we use a 2-wire sensor.

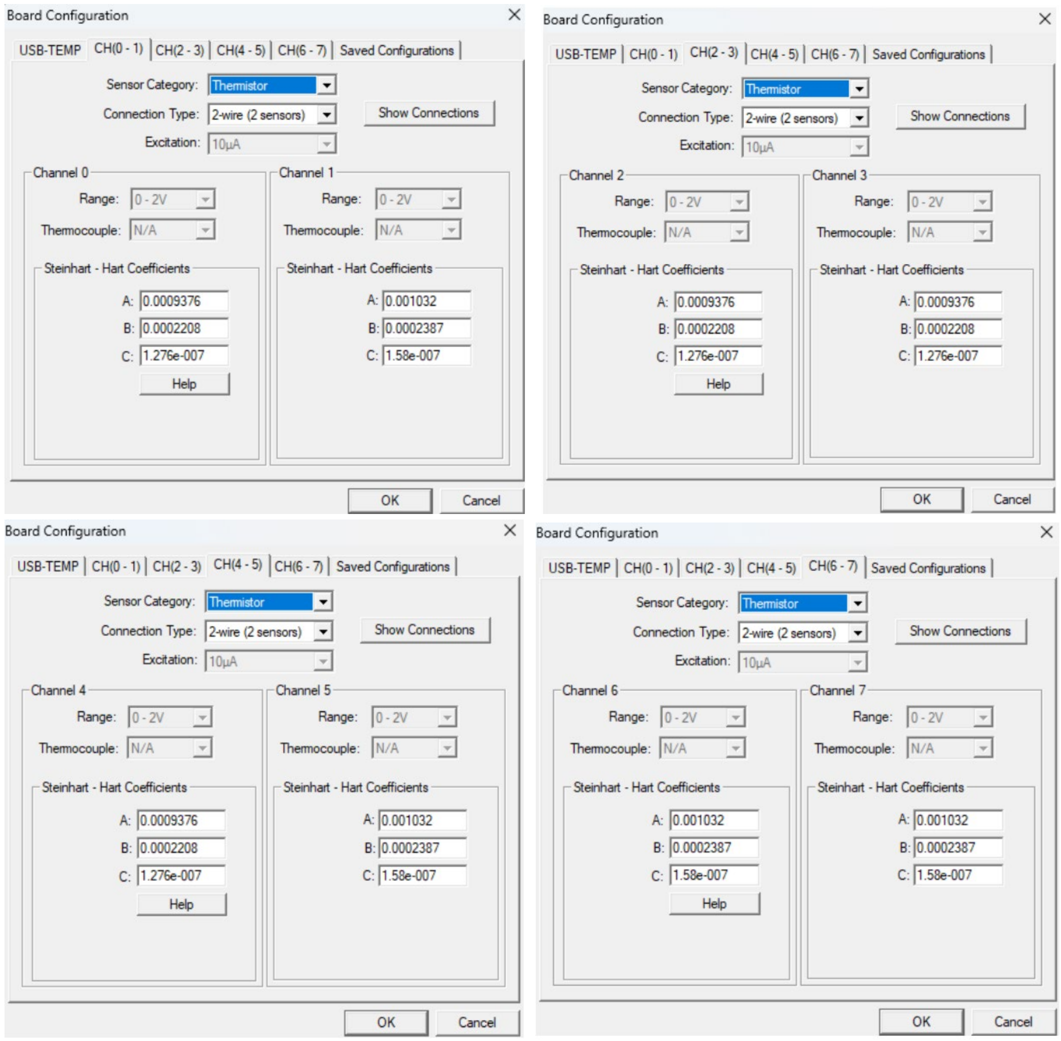


Figure C.4

To check if your DAQ and probes measure the temperature correctly, you can use a board test. If there is connection issue, then you will see wrong value such as -9999.000000. Here is an example. I connected 7 channels with no connection of CH5. CH4 shows -9999 meaning the wire is broken or has an issue.

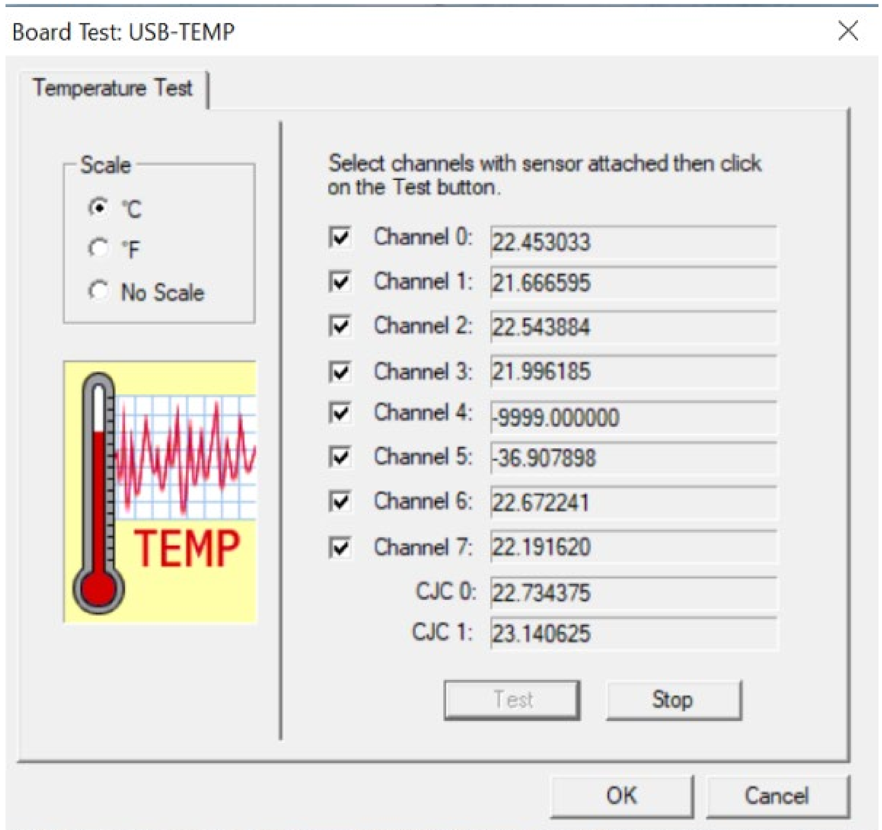


Figure C.5

b. *Python*

Python is required to run InstaCal and record the temperature data.
Here is the link for download: https://www.python.org/downloads/

c. *STEP 1: Jupyter Notebook*

Open CMD or PowerWell as Admin, and install Jupyter Notebook typing:
pip install notebook

d. *STEP 2. Libraries for MCC*

Now, you should have a library for MCC and here is the code to upgrade your pip.
Please type this code in CMD.
pip install --upgrade pip
Then, you can install the library for MCC
pip install mcculw
Now, you can use mcculw library and check the InstaCal

e. *STEP 3. Library for ipympl*

Now, for animation of plotting, please type this code in CMD.
pip install ipympl

f. *STEP 4. Running Jupyter Notebook*

Then, run the software typing in CMD as below,
jupyter notebook

or
python -m notebook

2. *Data Acquisition Code*
The code below directly measures the temperature of assigned channels. You should check the proper board number of the connected DAQ via InstaCal. Also, check the proper channels' connection.
    a. Simply, you can run the code below, but you should write something by yourself in the “CHECK POINT” lines. You will put your experiment name.
    b. Also, you should check “Config” lines. The board number should be the same as the DAQ InstaCal status. Check channels and interval_sec. Read the comments very carefully to use this code.

```
######### DAQ Reading from InstaCal and Temperature Logging with Python
#########
### Author: Asena Gelisli
### Convertered to Jupyter Notebook and modified by Youngmun
###
### This code is developed for ME 320 Heat Transfer Lab instrumentation
### to record temperature readings from 8 thermistors on the Guarded Heat Flow Meter
(GHFM) setup
### using the MCC USB-TEMP DAQ device configured via InstaCal.
###
### ⚠ WARNING:
### Do NOT open the output CSV file in Excel while the script is running.
### Excel locks the file and will cause the script to crash during write attempts.
### → To monitor the data safely during logging:
###    • Use Notepad++ or VSCode (non-blocking viewers)
###    • OR view a copied version of the CSV file

# --- Jupyter Notebook version of DAQ logger ---

from mcculw import ul
from mcculw.enums import TempScale
import matplotlib.pyplot as plt
import matplotlib.animation as animation
from datetime import datetime
import time, csv, os
```

```
# Activate interactive plots inside Jupyter
%matplotlib widget

# --- CHECK POINT ---
EXPERIMENT_NAME = "Testing local" #"Top_block_test1"
timestamp = datetime.now().strftime("%Y%m%d_%H%M%S")

# --- Output Files ---
output_file = f"{EXPERIMENT_NAME}_temp_log_{timestamp}.csv"
plot_file = f"{EXPERIMENT_NAME}_plot_live.png"

# --- Config ---
board_num = 0  # This should be the same in InstaCal
channels = list(range(8)) # If you use all channels, then range(8)
interval_sec = 9 # This means recording of temperature is done for every "interval_sec".
You can change it.
total_duration_sec = None
tick_spacing = 10

# --- Storage ---
time_data, elapsed_time_data = [], []
temp_data = [[] for _ in channels]

# --- Setup CSV ---
if not os.path.exists(output_file):
    with open(output_file, 'w', newline='') as f:
        writer = csv.writer(f)
        header = ['Time', 'Elapsed_sec'] + [f'CH{i}_Temp(C)' for i in channels]
        writer.writerow(header)

# --- Matplotlib Setup ---
fig, ax = plt.subplots(figsize=(10, 5))
lines = []
color_list = plt.rcParams['axes.prop_cycle'].by_key()['color']

for i, ch in enumerate(channels):
    (line,) = ax.plot([], [], label=f'T{i}', color=color_list[i % len(color_list)])
    lines.append(line)

ax.set_xlabel("Experiment Time (sec)")
```

```
ax.set_ylabel("Temperature (°C)")
ax.set_title(f"Real-Time Temperature Monitoring — {EXPERIMENT_NAME}")
ax.legend(loc='upper right')
plt.tight_layout()

# --- Main Loop ---
start_time = time.time()

def update_plot(frame):
    current_time = time.time()
    elapsed_time = round(current_time - start_time, 2)

    if total_duration_sec and elapsed_time > total_duration_sec:
        print("Logging complete.")
        ani.event_source.stop()
        return

    timestamp_now = datetime.now().strftime("%H:%M:%S")
    time_data.append(timestamp_now)
    elapsed_time_data.append(elapsed_time)
    temps = []

    for ch_index, ch in enumerate(channels):
        try:
            temp = ul.t_in(board_num, ch, TempScale.CELSIUS)
        except Exception as e:
            print(f"Error reading CH{ch}: {e}")
            temp = float('nan')
        temps.append(temp)
        temp_data[ch_index].append(temp)

    # Write to CSV
    with open(output_file, 'a', newline='') as f:
        writer = csv.writer(f)
        writer.writerow([timestamp_now, elapsed_time] + temps)

    # Update plot
    for i, line in enumerate(lines):
        line.set_data(elapsed_time_data, temp_data[i])
```

```
    ax.relim()
    ax.autoscale_view()
    ax.set_xlim(left=0, right=elapsed_time + 10)

    # Update ticks
    if len(elapsed_time_data) > 1:
        tick_indices = range(0, len(elapsed_time_data), tick_spacing)
        tick_values = [elapsed_time_data[i] for i in tick_indices]
        ax.set_xticks(tick_values)
        ax.set_xticklabels([str(t) for t in tick_values], rotation=45, ha='right')

    ax.tick_params(axis='x', labelsize=9, pad=6)
    fig.autofmt_xdate(rotation=45)
    plt.subplots_adjust(bottom=0.25)

    fig.canvas.draw_idle()
    plt.savefig(plot_file, dpi=80)

# Start animation
ani = animation.FuncAnimation(fig, update_plot, interval=interval_sec * 1000)
plt.show()

# --- Manual stop function ---
def stop_logging():
    print(">>> Logging stopped by user.")
    ani.event_source.stop()

# When you want to stop recording and plotting, please run this cell
stop_logging()
```

Then, you will see the saved file as below:

Figure C.6