

# Chapter 1

# Hardware

In this chapter, I will discuss the hardware used, which consists of the camera module, the reference temperature measurement module, the control and display unit, and finally, the model of the unit's construction. The implementation of the thermal camera is inspired by the Flir ETS320 thermal camera.

## 1.1 Camera Module

For the realization of the thermal camera, I chose the S319SPX camera module from SeekThermal. The module features a microbolometric sensor with a high resolution of 320x240 pixels and a high frame rate of up to 27 Hz[1]. The module communicates via a USB protocol. It is programmed using drivers available on the developer's site of SeekThermal. Programming, display modes, and other software specifications will be discussed in chapter 2.1. I selected the S319SPX module due to its optics, assuming that the factory optics would be capable of focusing on an object at a distance of 10 cm. The expected display area was approximately 32 mm x 43 mm with a resolution of 0.132 mm per pixel. After testing the module, it turned out that in the factory setting, the module could not display a sharp image of an object at a distance of 10 cm. Subsequent testing measured a minimum focal distance for a sharp image of approximately 1.5 m. The module is originally intended for imaging distant objects, see parameters HFOV, VFOV, identification, and detection distance in Table 1.1, and thus a lens attachment had to be used.

The used lens attachment, see Figure ??pic:ZnSeCocka], is made of ZnSe material[[a](#)]nd shortens the focal distance from the expected 100 mm to 38 mm. This reduction decreased the display area to 10 mm x 15 mm compared to the expected area of 32 mm x 43 mm. At the cost of a smaller scanned area, the resolution was improved to approximately 0.045 mm. A comparison of the module parameters with the inspiration camera can be seen in Table ??tab:SeekXFlir]. Figures ??pic:Seek] to ??pic:SeekSide] show the S319SPX module itself.

Table 1.1: Materials used for thermal camera lenses[1].

Parameter	Value
Sensor resolution	320 x 240 pixels
Lens type	24° fixed lens
Field of view (HFOV x VFOV)	24° x 18°
Detection distance	758 m
Identification distance	108 m
Spectral range	7.8 - 14 $\mu$ m
Temperature sensitivity (NETD)	typ. 65 mK
Frame rate	up to 27 Hz
Temperature measurement range	-40°C to 330°C
Communication interface	USB
Compatibility	Android, Windows, Linux



Figure 1.1: Added zoom lens.

## 1.2 Reference temperature probe

As a reference thermometer, I chose measurement using a thermocouple thermometer of type K. The thermocouple is evaluated by an analog-to-digital converter, the ADS1247 from Texas Instruments, with a resolution of 24 bits. As seen in Table 1.3, this is an ADC designed for sensor measurements, offering high resolution of the output word and high conversion speed. These two characteristics enable higher temperature accuracy compared to other specialized converters for thermocouples, as shown in Table 1.4.

The converter communicates via the SPI bus and utilizes two input signals and one output signal.

During implementation, the module went through two versions. Both versions are largely based on the application circuit from TI. The first version was intended to use the MAX6610 reference thermometer, which would provide a reference voltage of 2.56 V for the converter, as detailed in Appendix A.1. However, this setup had a major drawback: the resulting thermometer would only

Table 1.2: Comparison of S319SPX and Flir ETS320[1, 2]

Parametr	SeekThermal S319SPX	FLIR ETS320
IR resolution	320 x 240 px	320 x 240 px
FOV	24° x 18°	45° x 34°
Spectral responsivity	7,8 - 14 µm	7,5 - 13 µm
Temperature sensitivity (NETD)	typ. 65 mK	< 60 mK
Frame rate	up to 27 Hz	9 Hz
Measurement range	-40°C až 330°C	-20°C až 250°C
Lens	24° fixed focus	Fixed focus
Price (ex. VAT)	700 €	2850 €

Table 1.3: Specification of ADC ADS1247

Parameter	Value
Supply voltage	2,7 - 5,5 V
Resolution	24 bits
Data rate	up to 2 kSps
No. inputs	4
Internal voltage reference	2,048 V
Low noise PGA	48 nVRMS at gain of 128
Communication protocol	SPI

function properly if the temperature being measured was higher than the ambient temperature of the cold junction. Therefore, I opted for a simpler circuit using the internal voltage and temperature reference of the converter, as shown in Appendix A.2. This allowed me to utilize the full operating range of the thermocouple. The final module can be seen in Figure 1.5.

### 1.3 Control and display unit

For data processing, I chose the Raspberry Pi 4B with 4 GB of RAM. I opted for this route because it is a powerful and relatively compact device that is readily available. Additionally, I needed access to GPIO pins for connecting the reference thermometer, as discussed in Chapter 1.2. The microcomputer is cooled by a passive heatsink, reaching a maximum temperature of around 40°C during operation.

Connected to the microcomputer is a 4.3-inch display with a resolution of 800 pixels x 480 pixels and a capacitive touch layer from Waveshare [3]. I selected this display because it uses a DSI connector for connection, thereby not occupying the microcomputer's 40-pin connector or any of its built-in micro-HDMI ports. This arrangement keeps the connections more discreet.

The microcomputer also features a power button, allowing it to be put into a sleep mode and then powered back on. The resulting block diagram of the

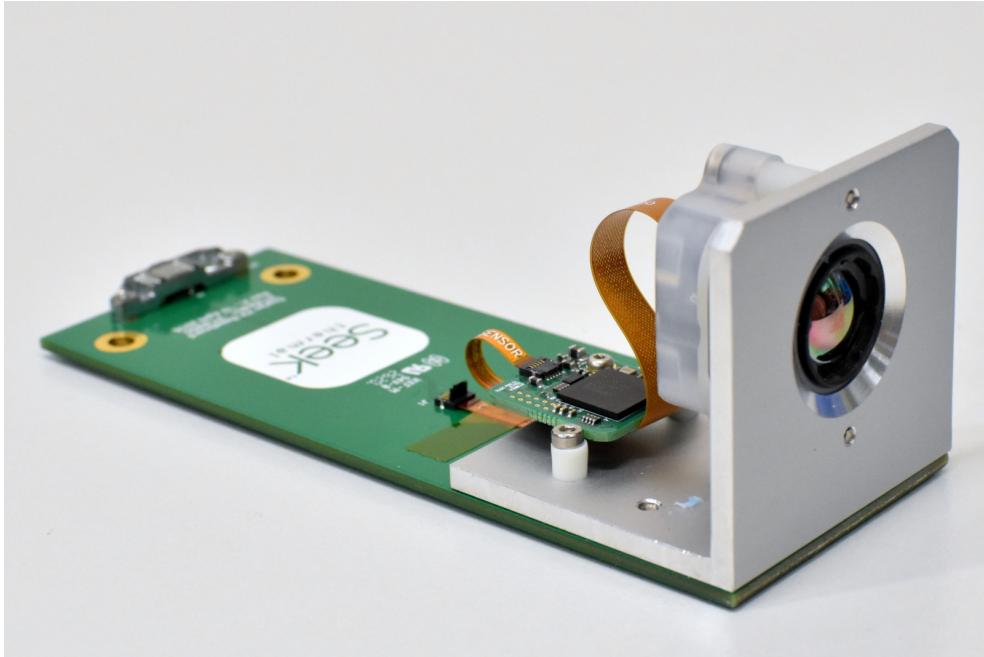


Figure 1.2: Camera module S319SPX.

connections between the individual modules can be seen in Figure 1.6.

## 1.4 Assembly

The entire structure is 3D printed in PETG plastic. I chose PETG plastic due to its higher glass transition temperature, between 75°C and 80°C [4]. The whole structure is designed so that you dont need to use any supports.

The frame of the unit consists of 10 pieces, joined together with screws. The display is tilted at a 30-degree angle to improve user viewing comfort.

The unit model can be seen in Figures 1.7, and the fully assembled print in Figure 1.8. To ensure the measuring space between the thermal camera and the object being monitored is as free as possible, an extension is attached to the distancing tunnel. This tunnel minimally intrudes into the field of view of the camera module itself, creating the impression of a freed up measuring space.

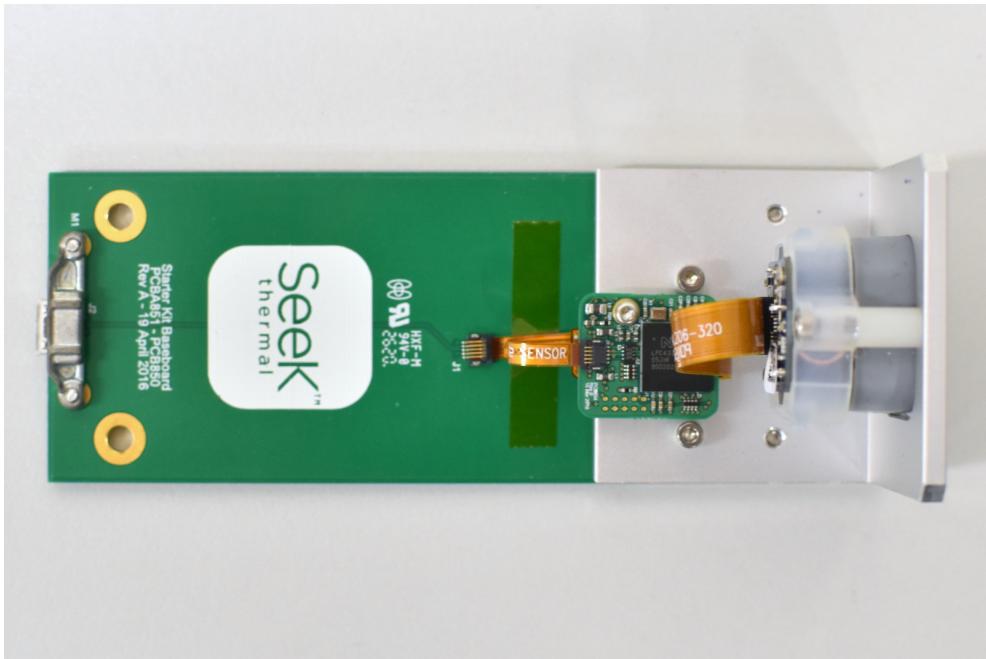


Figure 1.3: Top view of camera module S319SPX.

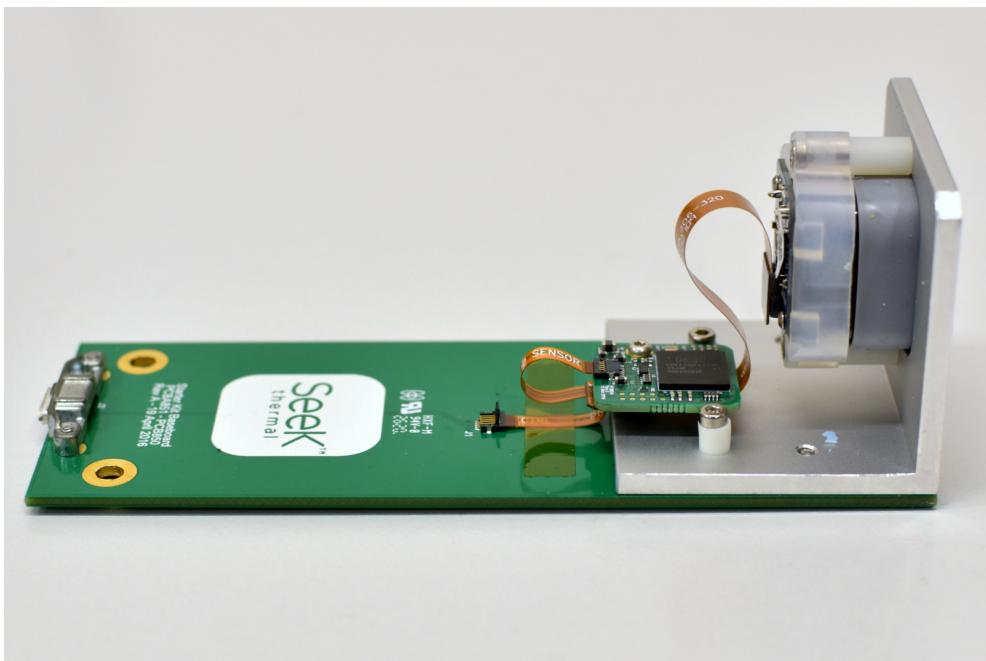


Figure 1.4: Side view of camera module S319SPX.

Table 1.4: Comparison of thermocouple interfaces

Interface	Positives	Negatives
ADS1247	high resolution (24-bit), low noise and drift, for precise measurement, high conversion rate of up to 2 kSps	relatively complicated implementation, requires external components
MAX6675	easy to implement, SPI comm protocol, cheap and available	only for type K thermocouples, lower resolution (12-bit), lower precision, slow conversion rate (200 ms)
MCP96L00	usable for more types of thermocouples, integrated CJC, I2C comm protocol	complicated configuration, could need more intensive calibration, slightly higher energy consumption

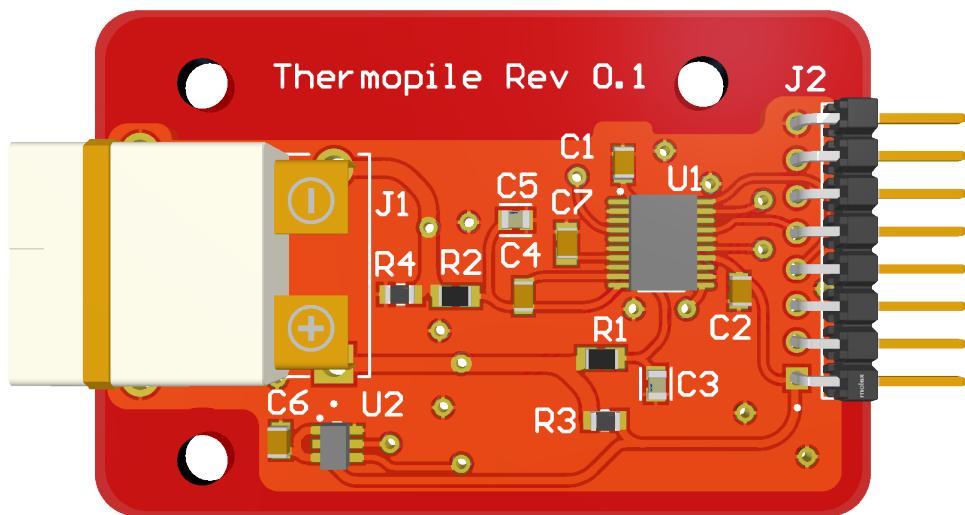


Figure 1.5: Reference temperature probe.

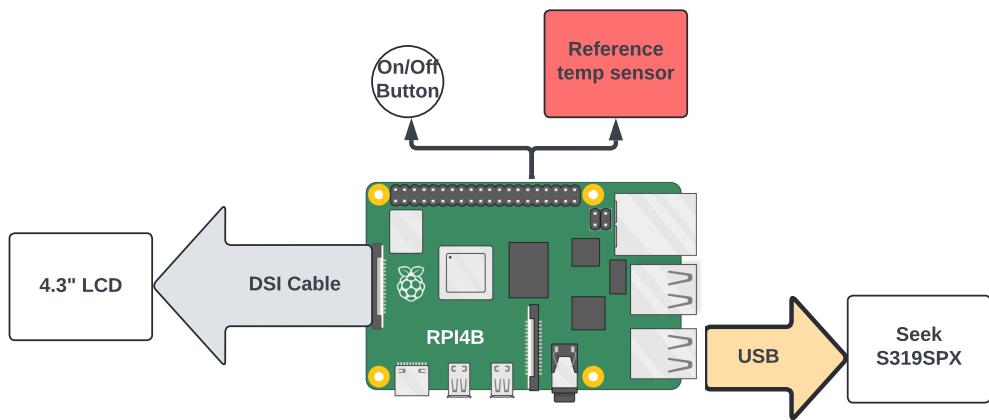


Figure 1.6: Connection schema for RPI4.

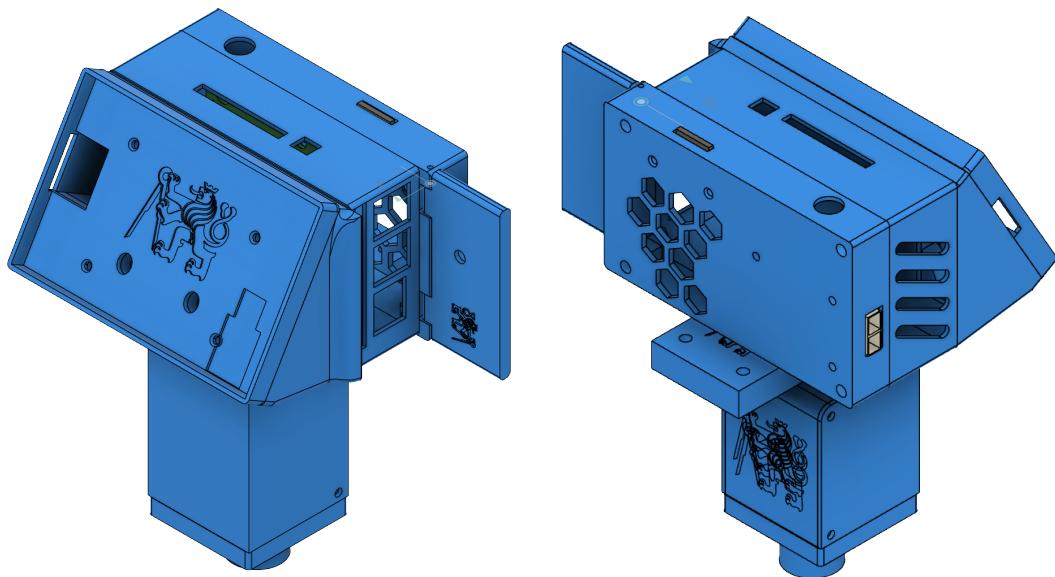


Figure 1.7: 3D model of thermal camera.

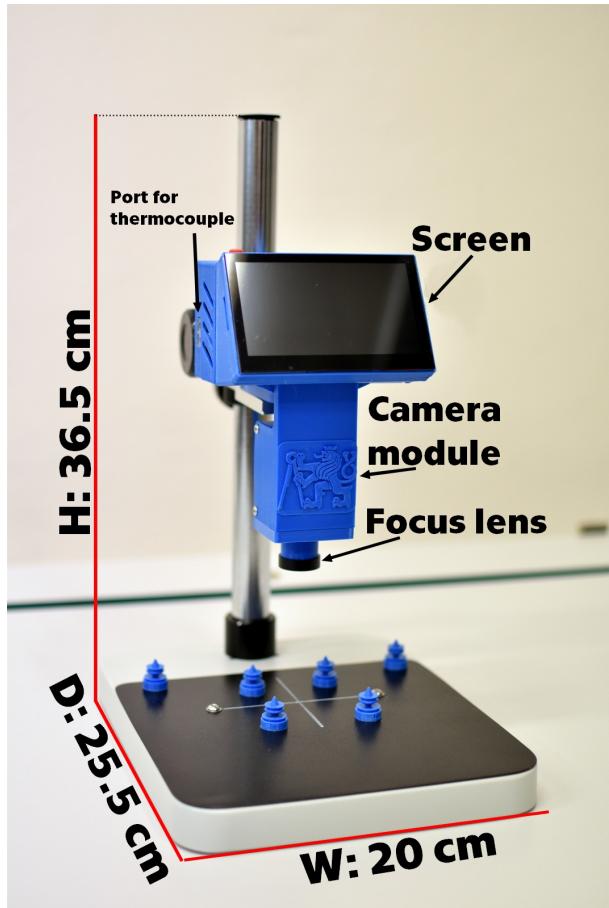


Figure 1.8: Resulting thermal camera with dims.



Figure 1.9: Inside ther thermal camera.

# Chapter 2

# Software

Since the thermal camera operates on the RPi4, I have decided that Python is the most suitable programming language for this purpose. In the following subchapters, I will discuss the software control of the camera module and then describe the resulting user application.

## 2.1 SeekThermal S319SPX

As I briefly mentioned in Chapter 1.1, the S319SPX camera module is connected to the display unit via USB and is programmed using SDKs from SeekThermal's developer portal. These controllers enable the utilization of the camera module in various applications such as machine learning, thermography, or detection monitoring, as shown in Figure 2.1.

Because the original SDKs are written in C/C++, I used the Python wrapper seekcamera-Python, maintained by the module manufacturer. The wrapper consists of the SeekCamera module, which maps functions from the original libraries into Python methods.

When programming the camera module, it's necessary to consider certain collision situations. For instance, when the camera module operates in display mode, the camera output provides color information for each pixel. In this mode, the module loses information about the measured temperature. If it's essential to know the measured temperature, the module must operate in thermography mode, where the module's output provides the measured temperature for each pixel. This mode fills the module's internal registers regarding minimum and maximum module temperatures, ambient temperature, and the so-called 'spot temperature'. Spot temperature refers to the average temperature of the central 36 pixels.

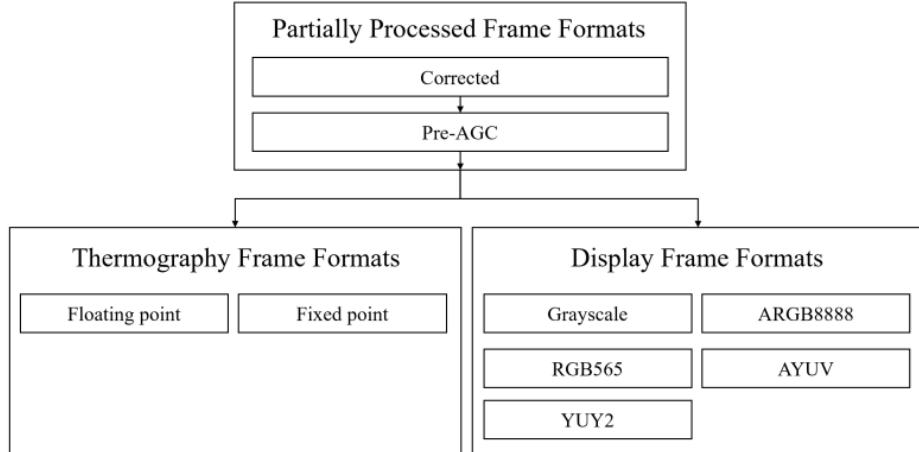


Figure 2.1: Image Processing Flowchart[[SeekSDKmanual](#)]

## 2.2 UI

The resulting application is created using the tkinter library, which is a basic library included in most Python installations. I chose this approach due to its simple structure and good documentation. As mentioned in subsection 2.1, I receive temperature information for each measured pixel from the camera module. These values need to be normalized to fit between 0 and 255. I then use these normalized values as indices in a color palette, creating the thermal image itself.

The application allows switching between various color palettes. Currently, it's possible to use any basic color palette from the OpenCV (CV2) library, as shown in Figure 2.2. During the programming of the color palette selection, I encountered a significant issue with the interaction between Tkinter and touch-screen input processing. If an element did not intersect with the mouse cursor, even if the cursor was invisible, the application did not register a click event (the application thought the element was not selected). The solution to this problem involved ensuring that the cursor always remained in the center of the screen unless the user moved it with the mouse. Therefore, I had to ensure that any popup windows intersected the center of the screen. This is not a visually ideal solution, but from my experiments, it proved to be the most reliable solution.

The application allows the user to save a snapshot either as a .PNG image or as raw data in a .CSV file. It also supports recording video in .AVI format. All recordings are saved to a folder on the desktop named 'Thermography'. This folder is organized into four subfolders, as shown in Figure 2.3. The 'logo' folder contains the used icons, the 'pictures' folder stores the thermal images, the 'RawData' folder contains raw data in .CSV format, and the 'videos' folder

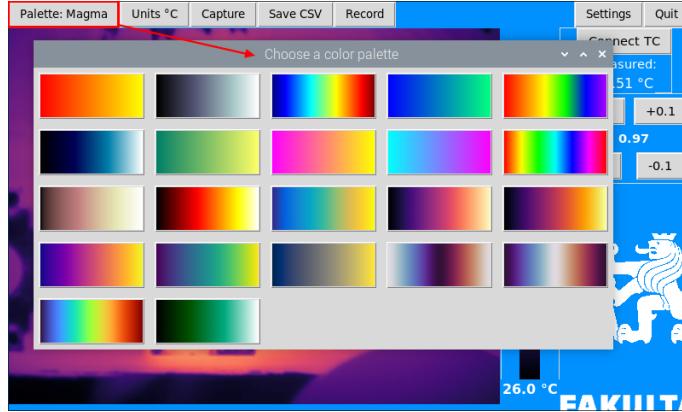


Figure 2.2: Available color palettes.

contains recorded videos.

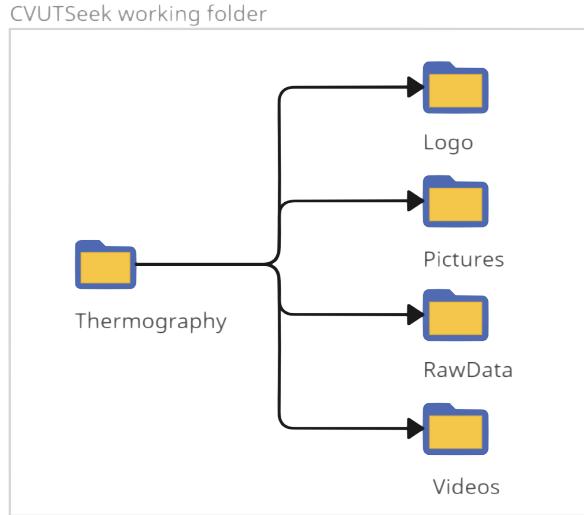


Figure 2.3: Diagram of the working directory of the thermal camera.

An important menu is the 'Settings' menu, as shown in Figure 2.4, where you can switch between fullscreen mode, toggle automatic scaling on/off, and manually adjust the image for larger monitors. When automatic scaling is turned off, a button for adjusting the scaling extremes appears in the settings, as seen in Figure 2.5. The scaling limit menu itself automatically opens when changing the scaling mode. The emissivity setting is located on the main screen.

In Figure 2.6, you can see the appearance of the application upon launch. Clicking the 'Connect TC' button initiates the reading of the reference ther-

mometer's thermocouple

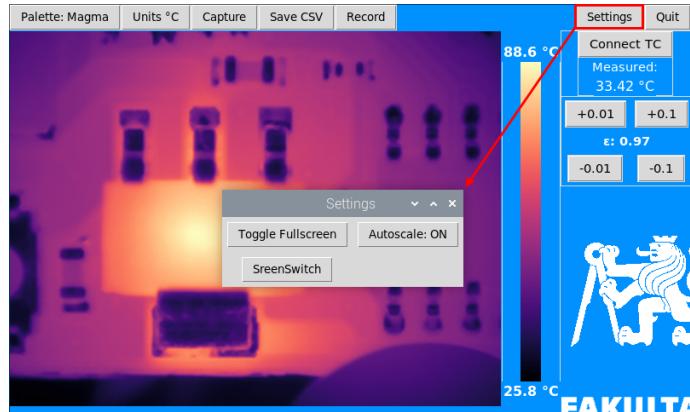


Figure 2.4: Setings menu.



Figure 2.5: Autoscale menu.

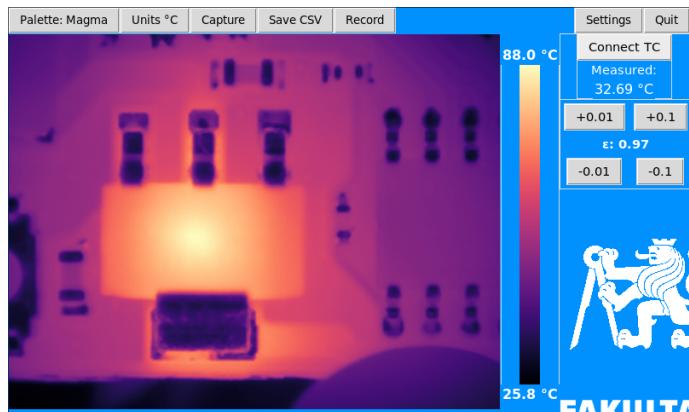


Figure 2.6: Default view of the UI.

# Chapter 3

# Testing

I conducted testing of the thermal camera's capabilities on the designed test PCB with three snapshots. Figure 3.1 shows the setup of the testing workstation.



Figure 3.1: Test workstation for performance evaluation.

## 3.1 Testing PCB

As previously mentioned, the test PCB consists of three snapshots. The first snapshot tests the resolution capability based on standard sizes of resistors commonly used in the industry. These components are listed in Table 3.1. The resistors are connected in parallel, as shown in Figure 3.2. I selected resistors based on the maximum load of size 0201, which is  $\frac{1}{20}$  W. At a fixed voltage of 15 V, the resistor value for 4.5 k $\Omega$  corresponds to the selected resistor size of

$4.7\text{ k}\Omega$ . This is a standard value readily available. To ensure uniform current distribution across all resistors, the remaining resistors are also  $4.7\text{ k}\Omega$ . Due to unavailability, I had to substitute one resistor of size 0201 with a value of  $6\text{ k}\Omega$ . Therefore, the total resistance of the test circuit is not  $940\text{ }\Omega$ , but  $980\text{ }\Omega$ .

Table 3.1: Resistor size table

Imperial	Size [thou]	Metric	Size [mm]
0201	h: 20 w: 10	0603	h: 0,6 w: 0,3
0402	h: 40 w: 20	1005	h: 0,6 w: 0,3
0603	h: 60 w: 30	1608	h: 1,6 w: 0,8
0805	h: 80 w: 50	2012	h: 2 w: 1,2
1206	h: 120 w: 60	3216	h: 3,2 w: 1,6

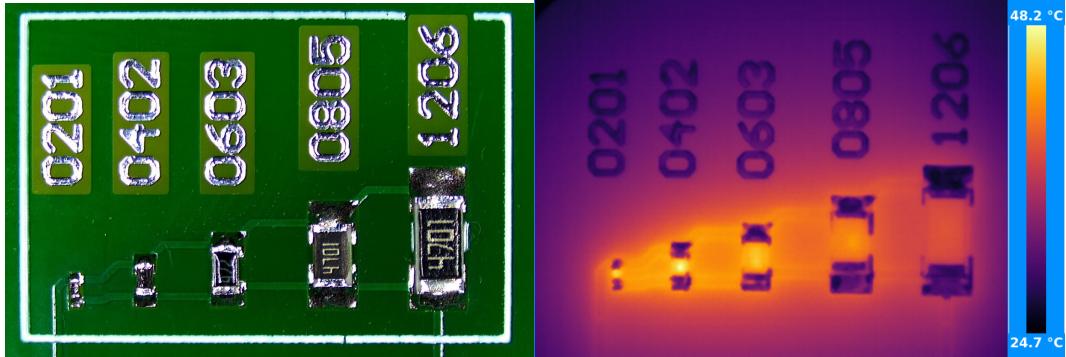


Figure 3.2: Resistor size recognition.

The second snapshot features a heating coil made of a copper trace with a width of  $0.254\text{ mm}$  and a total resistance of  $0.6\text{ }\Omega$ , as seen in Figure 3.3. From the image, a design flaw is apparent where the heating element lacks solder mask coverage. This causes issues with the emissivity of the tinned copper during measurements. In the thermal image, it appears as if the board is heating up while the tinned meander remains cool. This issue can be rectified by applying a solder mask layer.

### 3.2 Comparison with Optris XI 400

To assess the quality of the assembled thermal camera in displaying thermograms, I conducted testing using the Optris XI 400 thermal camera. This camera has a similar sensor resolution (382 pixels x 288 pixels) [5]. The test setup is shown in Figure 3.4. Compared to my constructed camera, the Optris thermal camera has adjustable focal length. For this test, to ensure the most similar conditions for comparison, I set the focal distance to the minimum possible,

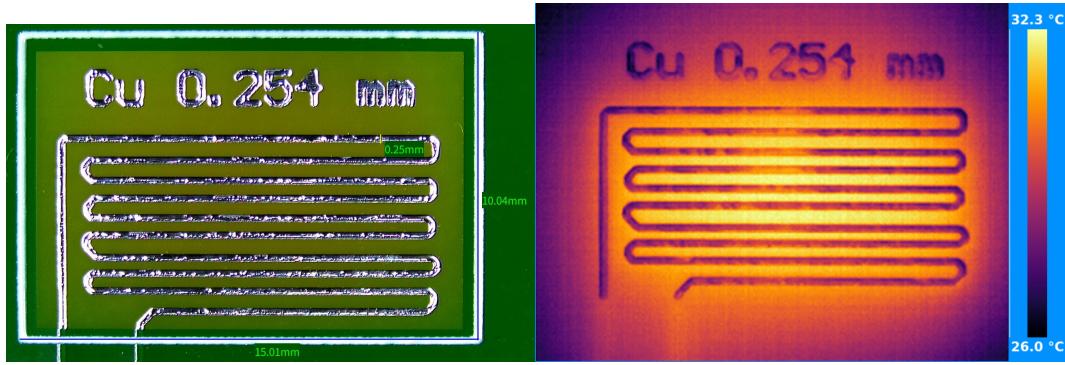


Figure 3.3: Heating coil test.

which is 80 mm. It's worth noting the larger required workspace due to the need for an external computer.

This highlights the advantage of an integrated computer system in the manufactured thermal camera. In the following images, you can see snapshots of the test PCB in various color palettes.



Figure 3.4: Test workstation for Optris XI 400.

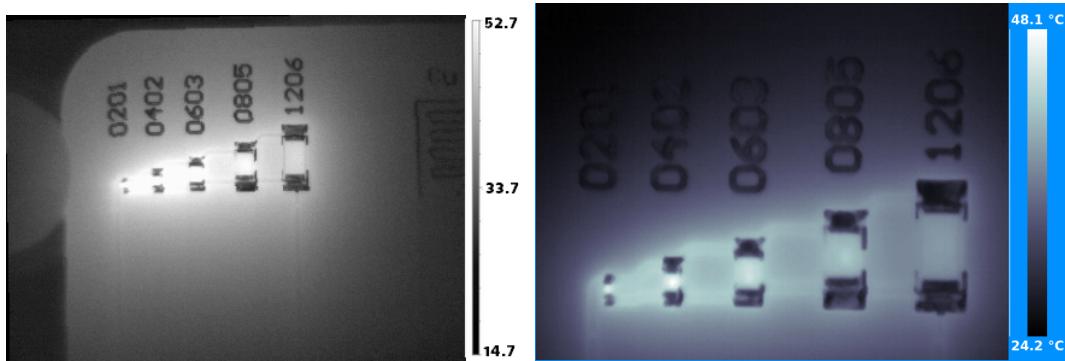


Figure 3.5: Comparison of component size thermograms: a) Optris b) custom-built thermal camera.

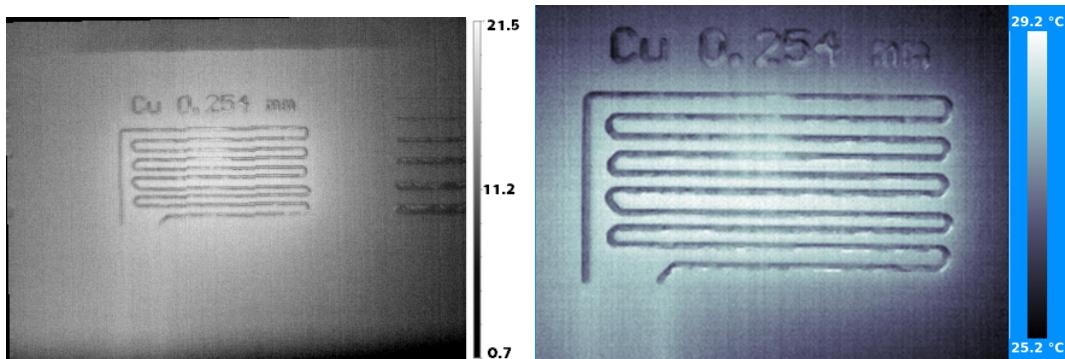


Figure 3.6: Comparison of heating element thermograms: a) Optris b) custom-built thermal camera.

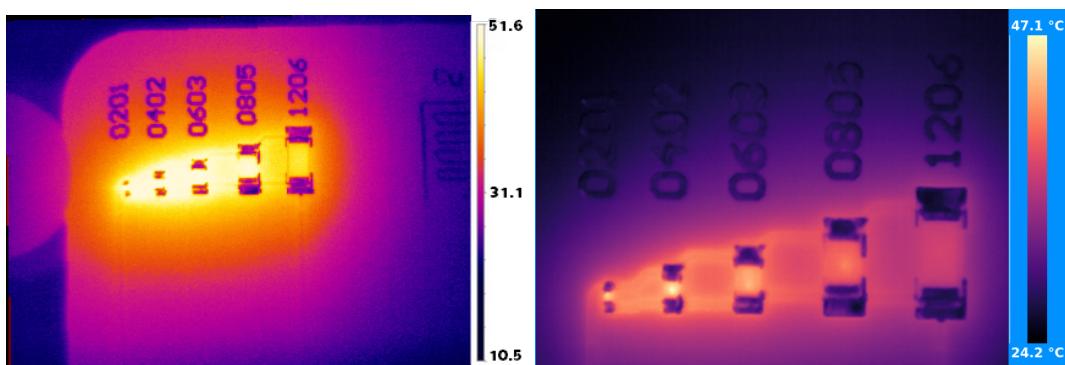


Figure 3.7: Comparison of component size thermograms: a) Optris b) custom-built thermal camera.

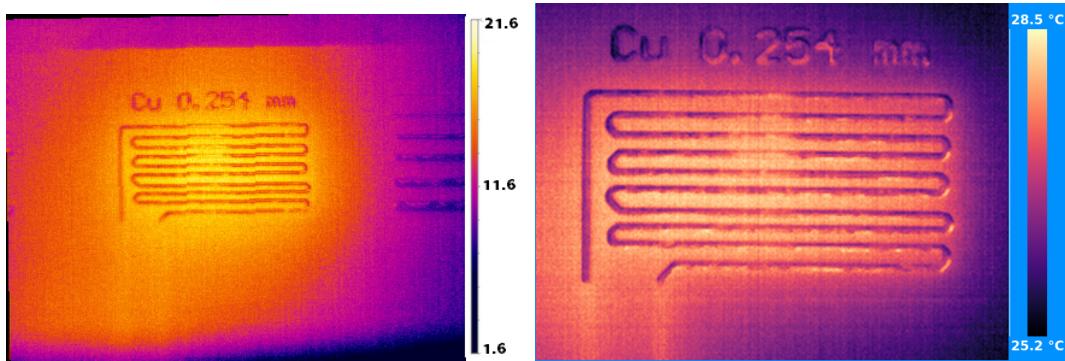


Figure 3.8: Comparison of heating element thermograms: a) Optris b) custom-built thermal camera.

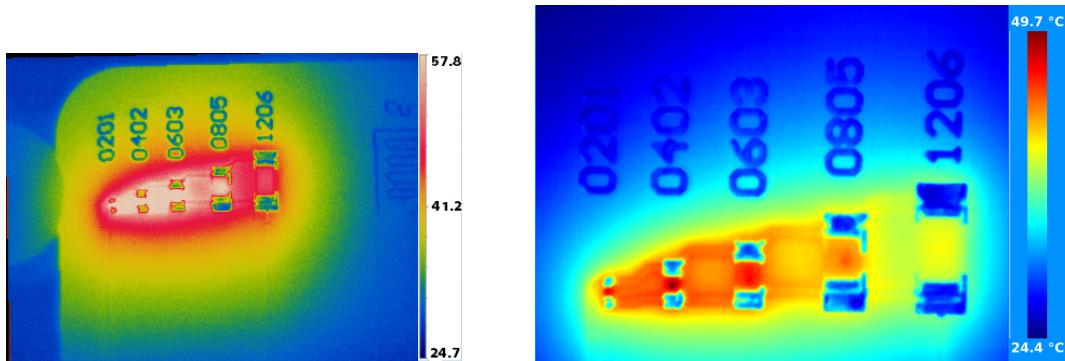


Figure 3.9: Comparison of component size thermograms: a) Optris b) custom-built thermal camera.

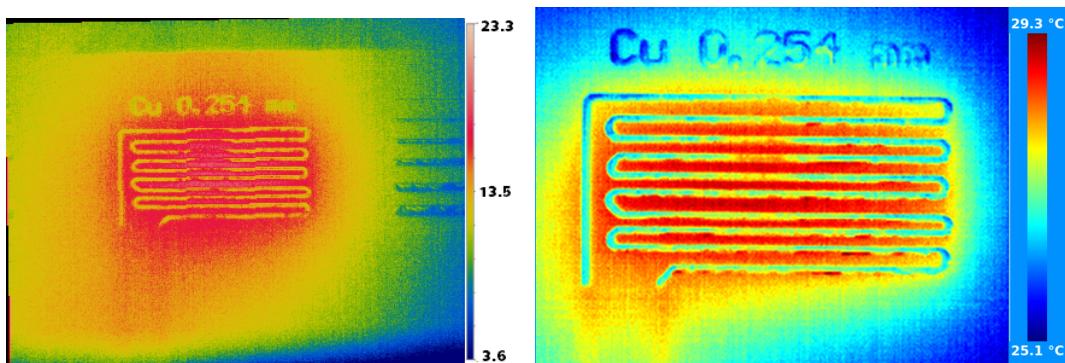


Figure 3.10: Comparison of heating element thermograms: a) Optris b) custom-built thermal camera.

# Chapter 4

## Results

From the testing, it can be seen that the thermal camera is suitable for inspecting PCBs populated with components as small as 0201. The higher refresh rate of the camera module allowed smooth scene rendering and captured rapid temperature changes. Tilting the screen by 30° greatly enhanced the ergonomics of the measurements.

The display application limits the measured temperature to a NETD of 100 mK. Therefore, the resulting thermal camera is suitable for basic PCB inspection. Due to its limited field of view, it is more suited for inspecting PCBs in mobile devices, which are continually shrinking in size.

In Table 4.1, I have summarized the costs for producing the thermal camera, excluding labor costs during the development phase.

Table 4.1: Final Product Cost

Item	Standard Price [\$]	Sale Price <sup>1</sup> [\$]
Camera module	700	134
Zoom lens	16	16
Stand	20	20
Stand base	20	20
Stand accessories	36	36
RPi4	60	60
Cooler	8	8
Display[6]	27	27
Frame casing (220g)	4	4
Reference thermometer	18	18
Fasteners	6	6
Case	20	20
Miscellaneous items	32	32
Total Cost (excluding VAT)	967	401

<sup>1</sup> Sale price at Arrow.com

The main idea for improvement is to redesign the user application. Currently, it utilizes a very basic graphical system, which limits the performance potential of the camera module. The module should be capable of capturing images at a frequency of 27 Hz, but the current display capability of the camera is only around 18 Hz. Therefore, further optimization, such as transitioning to platforms like PyQt or Kivy, which are designed for fast image processing, would benefit the application. Switching to a different graphical platform could also enhance performance at higher resolutions when connected to an external monitor. Additionally, adding measurement windows, which wasn't reliably achievable with Tkinter, could further enhance functionality.

Given that the measurement accuracy of the camera module is  $\pm 5$  °C or  $\pm 5\%$  over the range of 5 °C to 140 °C, or  $\pm 10\%$  over the range of 140 °C to 330 °C, the high precision of the ADS1247 converter used in the camera is not fully utilized. Hence, it would be advisable to consider the MCP9601 thermocouple interface, which offers sufficient resolution and accuracy even at higher conversion speeds.

Another enhancement for the thermal camera would be the addition of a battery power system. For the RPi, there are UPS modules commonly used for power outage protection, but they could also serve as a battery power solution for the thermal camera. During energy consumption measurements of the thermal camera, it was found to have a maximum power consumption of 12 W at 5 V and 2.4 A. Therefore, with a suitable power module, the thermal camera should be able to operate for approximately 2 hours on battery power.

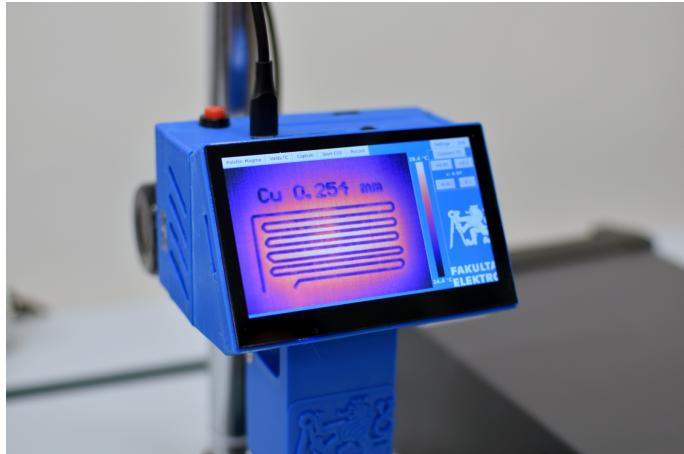


Figure 4.1: Thermal Camera in Operation During Testing.

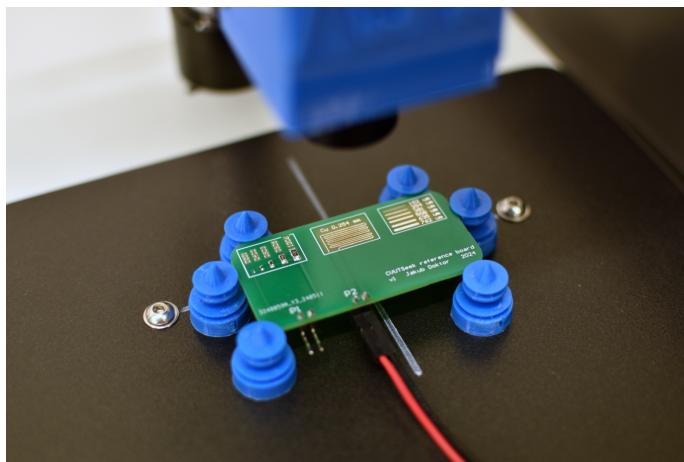


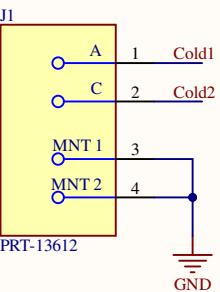
Figure 4.2: Mounting of the Test PCB.

# Bibliography

1. SEEKTHERMAL. *Mosaic core datasheet* [online]. 2021. Available also from: [https://www.thermal.com/uploads/1/0/1/3/101388544/mosaic\\_core\\_specification\\_sheet\\_2021v2-web.pdf](https://www.thermal.com/uploads/1/0/1/3/101388544/mosaic_core_specification_sheet_2021v2-web.pdf).
2. SYSTEMS, Flir. *Flir ETS320 datasheet* [online]. 2017. Available also from: <https://flir.netx.net/file/asset/3949/original/attachment>.
3. WAVESHARE. *43H-800480-IPS* [online]. [N.d.]. Available also from: <https://www.waveshare.com/wiki/43H-800480-IPS>.
4. COLÓN QUINTANA, José; SLATTERY, Lucinda; PINKHAM, Jon; KEATON, Joanna; LOPEZ-ANIDO, Roberto; SHARP, Keith. Effects of Fiber Orientation on the Coefficient of Thermal Expansion of Fiber-Filled Polymer Systems in Large Format Polymer Extrusion-Based Additive Manufacturing. *Materials*. 2022, vol. 15, p. 2764. Available from DOI: 10.3390/ma15082764.
5. *Operator's Manual fo Optris XI series* [online]. 2020. Available also from: <https://www.optris.com/en/download/xi-series-manual/?wpdmdl=160838&refresh=65e5a8e93998f1709549801>.
6. RPISHOP.CZ. *WaveShare 4.3 inch touchscreen* [online]. [N.d.]. Available also from: <https://rpishop.cz/lcd-oled-displeje/6295-wavehare-43-dsi-displej-ips-800-480-kapacitni-dotykovy-i2c.html>.

# **Appendix A**

## **Reference temperature probe**



Obrázek D.1. Iniciální návrh referenčního teploměru.

Title		
Size	Number	Revision
A4		
Date:	5.16.2024	Sheet of
File:	C:\Users\.\ThermopileBorad.SchDoc	Drawn By: Jakub Doktor

A

1

B

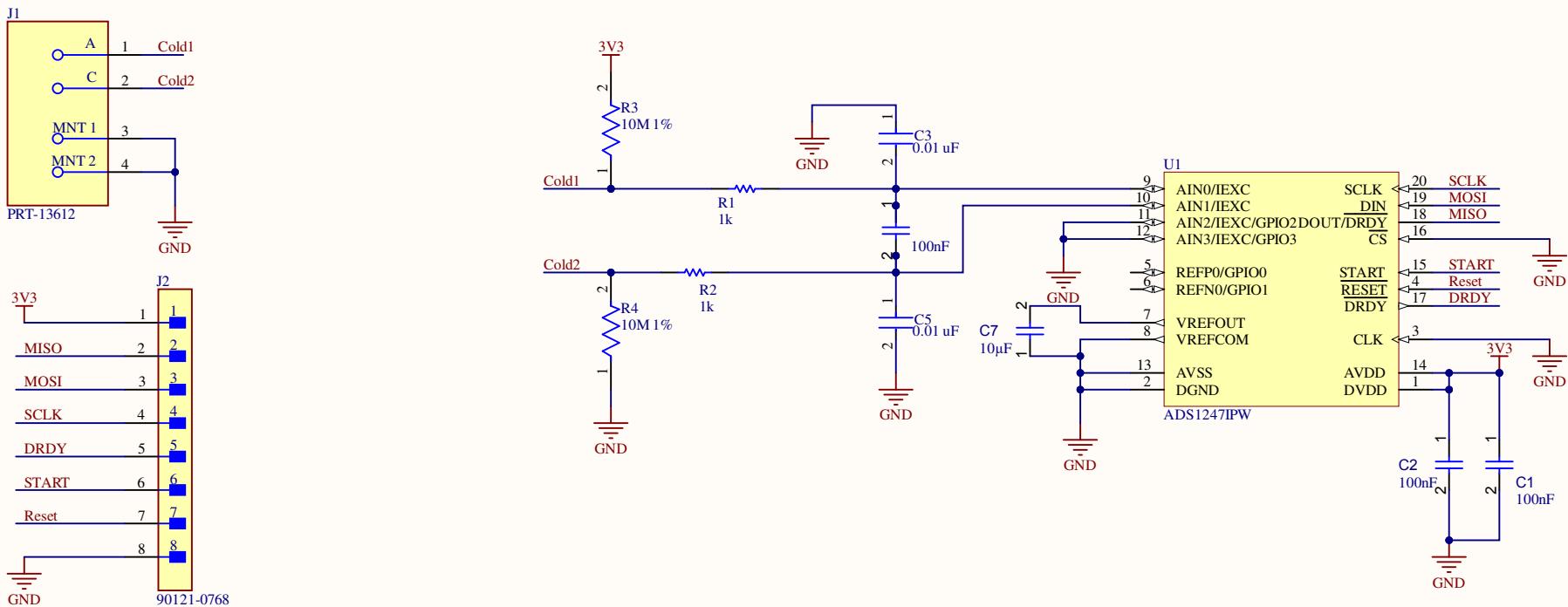
1

C

8

D

1



**Obrázek D.7.** Zjednodušené zapojení referenčního teploměru.

Title		
Thermocouple interface - simplified		
Size A4	Number	Revision
Date:	5.16.2024	Sheet of
File:	C:\Users\...\ThermopileBorad.SchDoc	Drawn By: Jakub Doktor