Sprawozdanie

Identyfikacja wartości współczynnika przewodności cieplnej oraz pochłaniania ciepła płytki aluminiowej

Mateusz Krupnik, Wojciech Koźlak

# Opis teoretyczny zagadnienia

W rozważaniach związanych z temperatura i wymianą ciepła występująca takie zjawiska jak przewodzenie ciepła, konwekcja ciepła oraz promieniowanie. W tym rozdziale zostaną przybliżone opisy tych zjawisk oraz ich modele matematyczne. Dodatkowo opisany zostanie moduł do prowadzenia optymalizacji powierzchni odpowiedzi planu eksperymentu.

## Opis zjawisk fizycznych

Opis przewodzenia, konwekcji, promieniowania

## Modele zjawisk fizycznych

Modele przewodzenia, konwekcji i promieniowania

## Ograniczenia stosowania modeli

Jakieś ograniczenia trzeba wymyślić

## Moduł optymalizacji powierzchni odpowiedzi

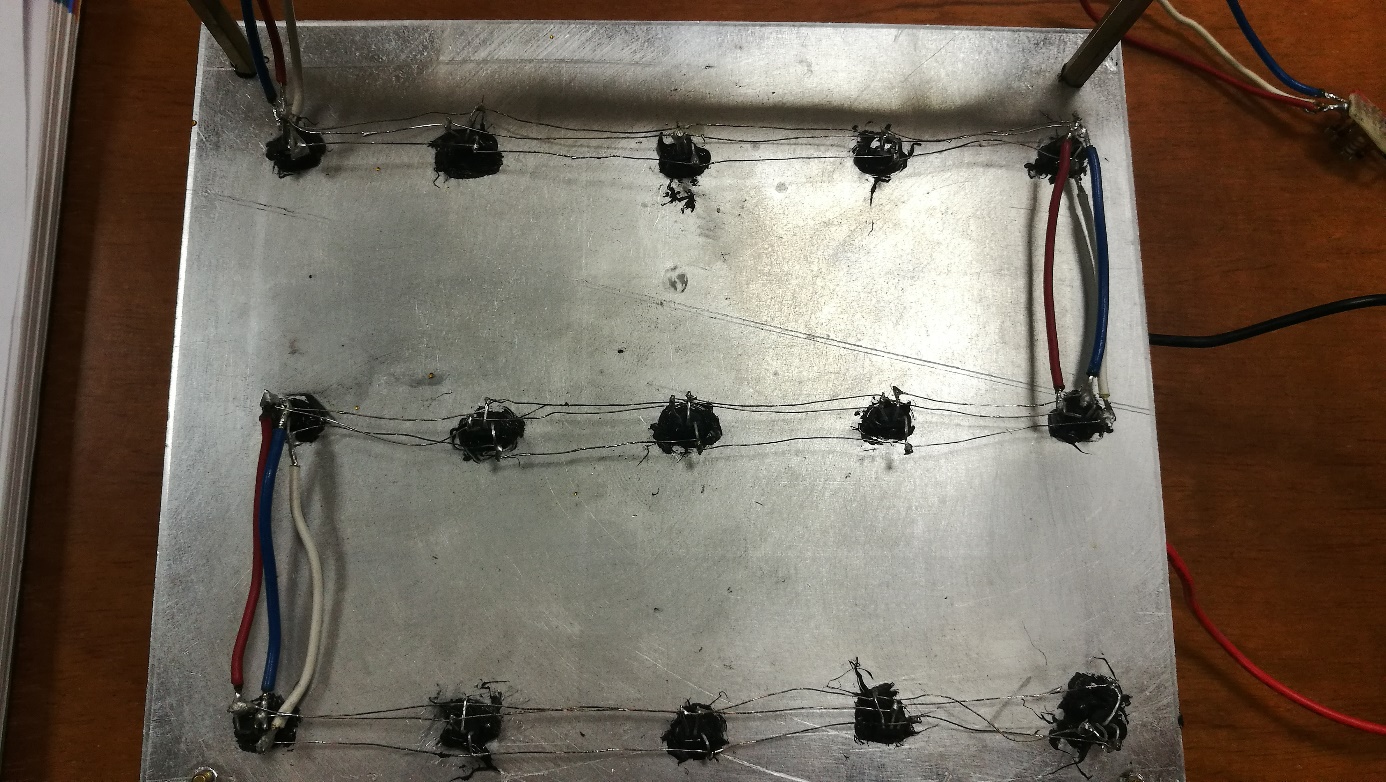
Opis modułu planowania eksperymentu i optymalizacji

# Eksperyment pomiarowy

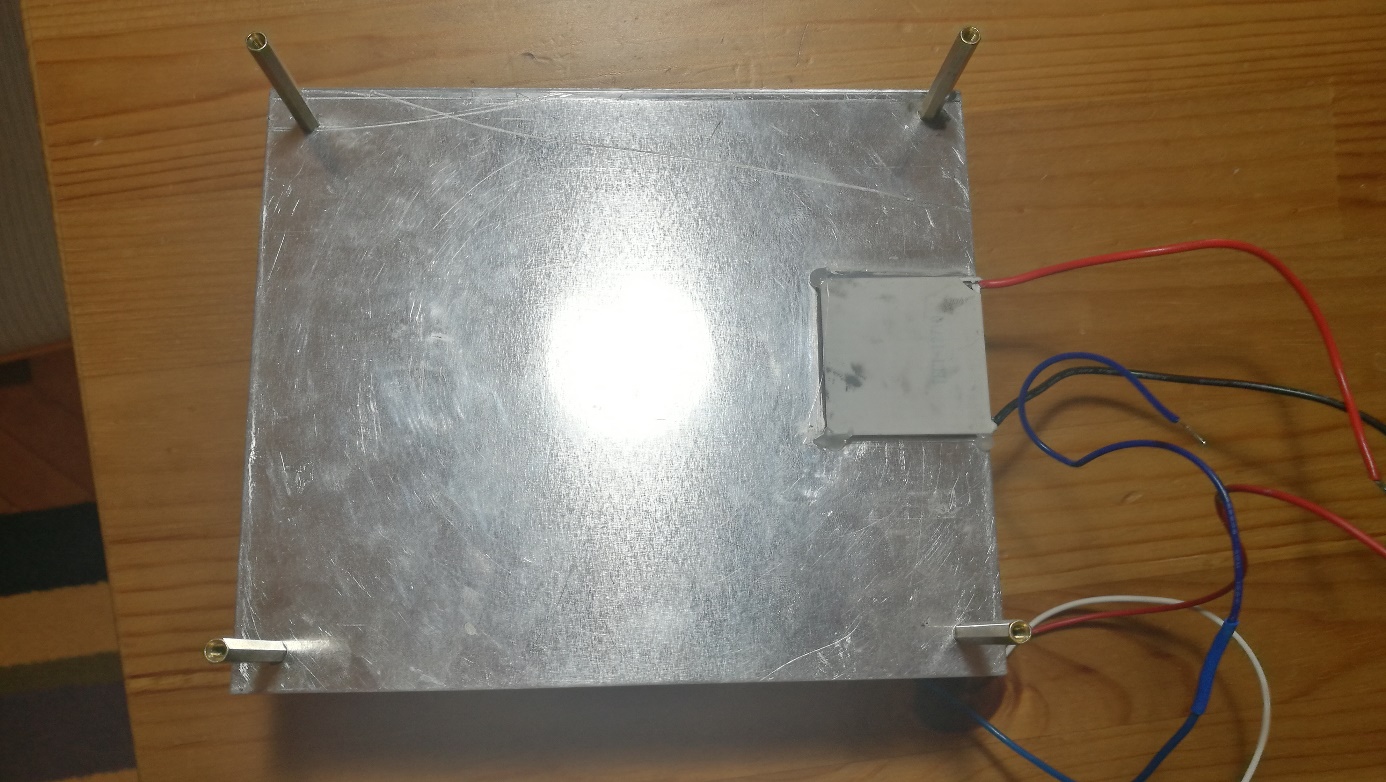
W ramach projektu przeprowadzony został eksperyment pomiarowy. Zostały przeprowadzone dwa pomiary temperatur na badanej płytce aluminiowej. Pierwszy pomiar zapewnił dane do identyfikacji parametrów związanych z wymianą ciepła płytki, drugi posłużył walidacji uzyskanego modelu i wyznaczonych parametrów.

## Opis eksperymentu oraz obiektu

Eksperyment polegał na zebraniu pomiarów temperatury z płytki aluminiowej nagrzewanej za pomocą modułu Peltiera 12710. Temperatura była mierzona w 15 punktach pomiarowych na płytce aluminiowej a także mierzona była temperatura otoczenia i temperatura zimnej strony modułu. Aby określić moc cieplną modułu potrzebna była znajomość różnicy temperatury zimnej i ciepłej strony modułu a także napięcie zasilania. Wspominana płytka aluminiowa o wymiarach 180x150x3 mm widoczna jest na poniższych zdjęciach wraz z rozmieszczeniem modułu grzewczego oraz czujników.

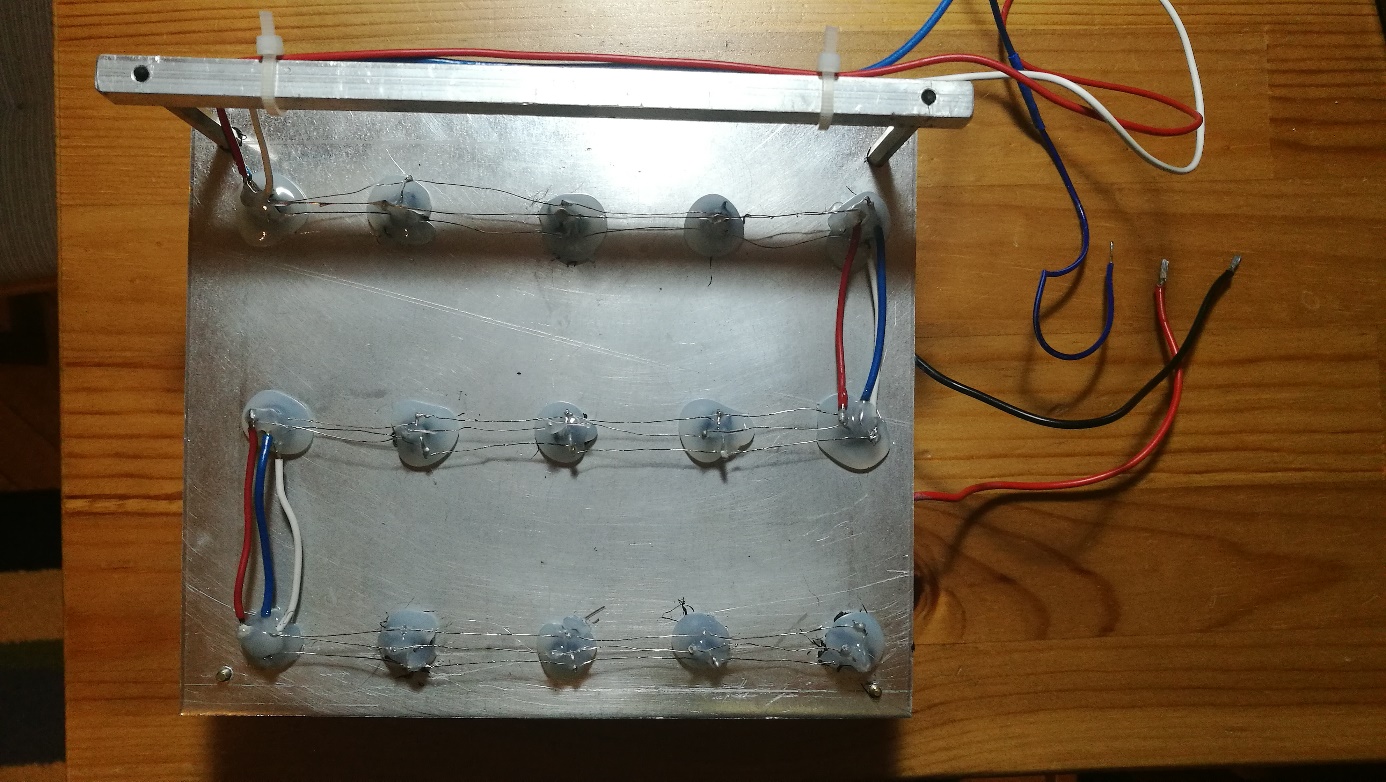


Rysunek 1. Badany obiekt - widok z góry



Rysunek 2. Badany obiekt - widok z dołu

Po przeprowadzeniu wstępnych pomiarów zaobserwowany znaczący wpływ otoczenia na mierzone przez czujniki wartości temperatury dlatego zdecydowano się je odizolować za pomocą kleju na gorąco. Ten sposób izolacji pomimo poprawy wskazań temperatur wprowadza pewne rozbieżności z modelem zastosowanym w środowisku Ansys. Wynikają one między innymi ze zmianą powierzchni wymiany ciepła. Na kolejnym rysunku widoczny jest obiekt po zastosowaniu izolacji.



Rysunek 3. Badany obiekt po zaizolowaniu czujników

Eksperyment właściwy prowadzony był w laboratorium katedry Systemów Energetycznych i Urządzeń Ochrony Środowiska. Jako zasilanie zastosowany został dostępny tam zasilacz z regulacją napięcia i natężenia prądu. Mikrokontroler został zasilony za pomocą interfejsu USB z komputera, który zbierał dane za pomocą przygotowanych skryptów. Wykonane zostały dwa pomiary, z czego jeden posłużył do wyznaczenia szukanych parametrów a drugi do ich walidacji.

## Stworzone oprogramowanie oraz tor pomiarowy

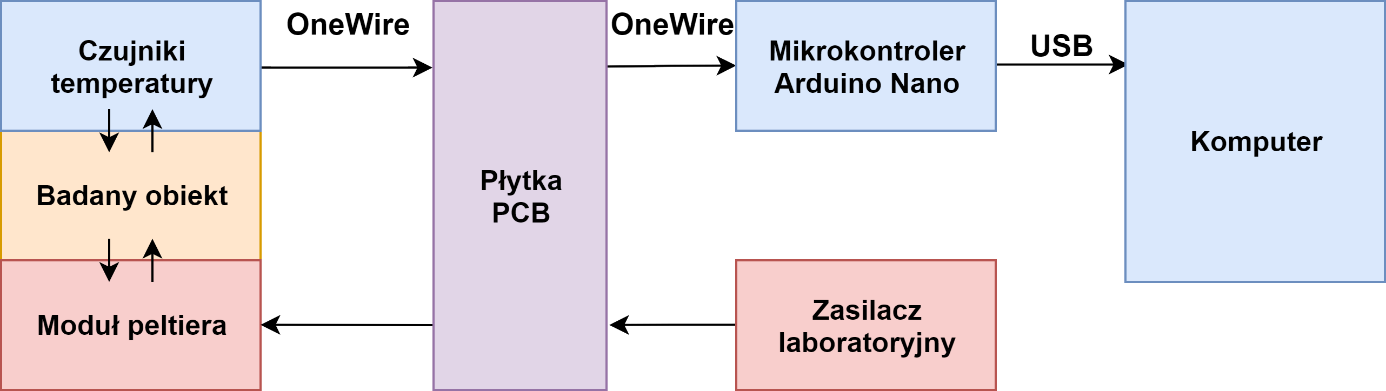
W ramach projektu wykonana została prosta płytka służąca do połączenia komputera, mikrokontrolera, czujników i modułu Peltiera. Ze względu na znaczną liczbę czujników zastosowano połączenie z rezystorem podciągającym i linią zasilającą. Nie wskazane jest stosowanie połączenia dwuprzewodowego z linią pasożytniczą, ponieważ pojemność tej linii powoduje powolny zanik zbocz sygnałowych przez co próbkowanie jest powolne. Przeprowadzono dodatkowo test szybkości próbkowania (dla 17 czujników) w zależności od zastosowanej rozdzielczości 9-12 bit. Wyniki przedstawiono w poniższej tabeli.

Tabela 1. Zależność rozdzielczości oraz okresu próbkowania od wybranej liczby bitów

|  |  |  |
| --- | --- | --- |
| Liczba bitów | Rozdzielczość [°C] | Okres próbkowania [s] |
| 9 | **0,5** | **0,3** |
| 10 | **0,25** | **0,3** |
| 11 | **0,125** | **0,5** |
| 12 | **0,0675** | **0,75** |

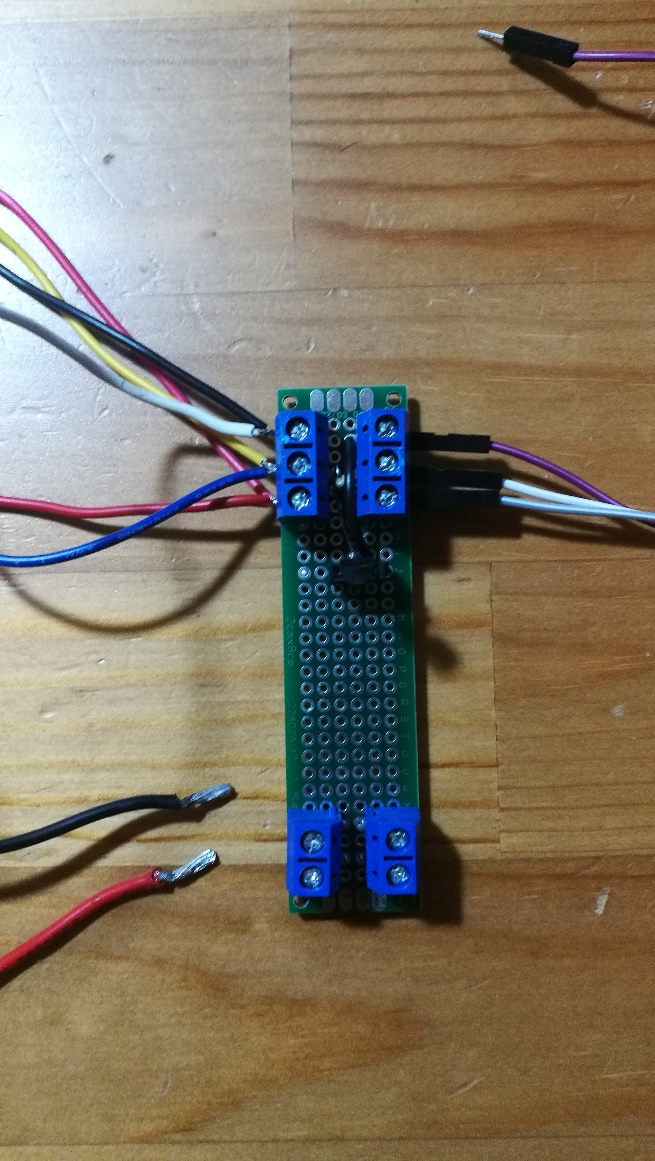
Jak można zauważyć różnica pomiędzy 9 i 10 bitami w okresie próbkowania jest nie widoczna. Okres próbkowania dla 12-to bitowej rozdzielczości jest zadowalający uwzględniając natura zjawiska przewodzenia ciepła, które nie charakteryzują się dużą dynamiką, a także nie wielką moc źródła ciepła.

Na kolejnych rysunkach przedstawiony został tor pomiarowy oraz schemat połączeń.

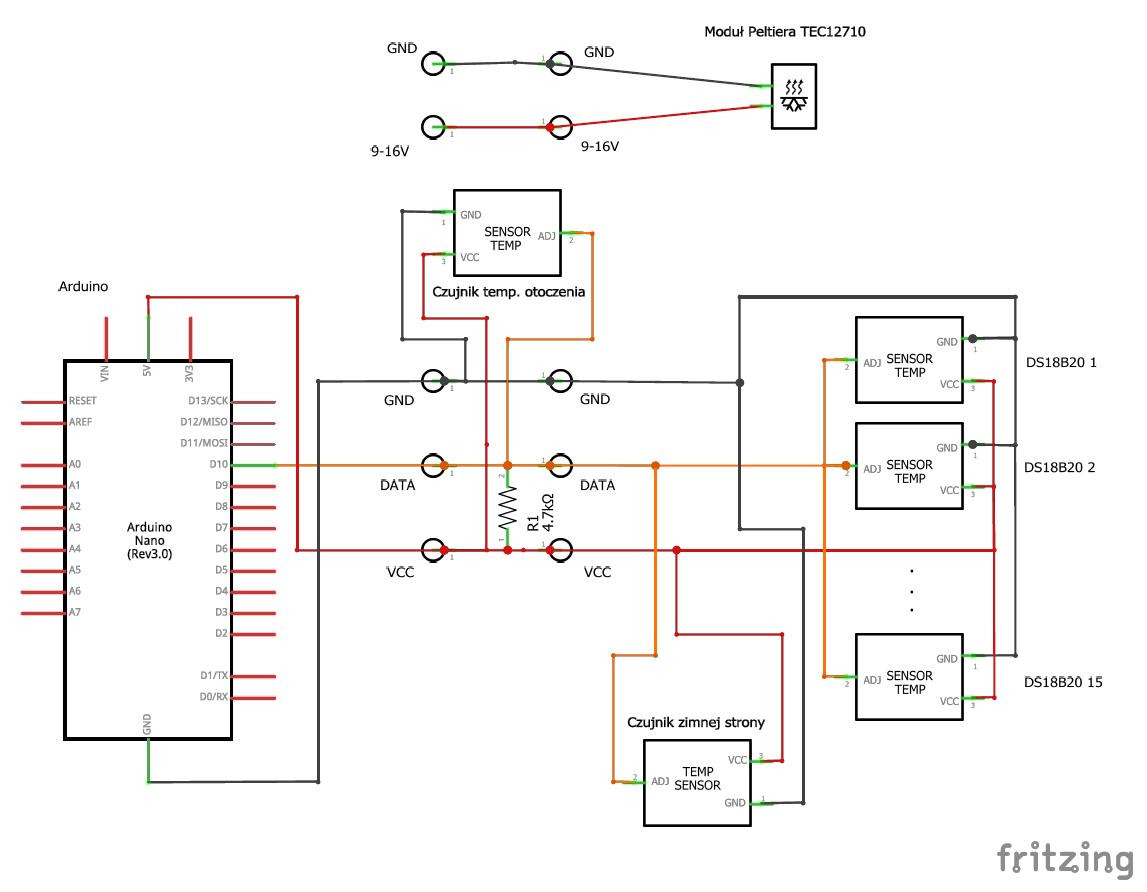


Rysunek 4. Tor pomiarowy

Komunikacja z czujnikami jak już zostało wspomniane odbywa się za pomocą linii danych magistrali *OneWire*, do której obsługi zastosowano dostępne biblioteki. W przypadku tego typu magistrali ważna jest jej długość ze względu na pojemność linii i spadki napięć, dlatego zdecydowano się na podłączenie trój przewodowe. Wykonana płytka lutownicza zawiera przyłącze listwowe o rastrze 2,54 mm na czujnik temperatury otoczenia. Dodatkowo znajduje się na niej rezystor podciągający linie danych do linii zasilania oraz terminale przyłączeniowe. Schemat połączeń został wykonany w darmowym programie *Fritzing*.



Rysunek 5. Płytka lutownicza



Rysunek 6. Schemat połączeń toru pomiarowego

Kod programów zostanie załączony w postaci załączników. Zostały napisane 2 programy w środowisku ArduinoIDE oraz skrypty w Pythonie.

Pierwszy program napisany w celu wgrania go w pamięci mikrokontrolera służył do wypisania adresów czujników dostępny na magistrali w monitorze portu szeregowego URAT. W celu zapisania tych adresów powstał skrypt w Pythonie. Po wgraniu skryptu DS18B20\_find\_adresses.ino i połączeniu układu adresy można wyświetlić w monitorze UART lub zapisać do pliku *csv* za pomocą skryptu read\_adresses.py. W skrypcie należy ustawić odpowiednią prędkość *baud\_rate* portu oraz jego nazwę. Program do wgrania w pamięć mikrokontrolera korzysta z ogólnodostępnej biblioteki, która również zostanie załączona. Adresy w kolejności 1-15, czujnik temperatury otoczenia i czujnik temperatury zimnej strony modułu przedstawione są tabeli 2.

Tabela 2. Adresy czujników

|  |  |
| --- | --- |
| Czujnik | Adres |
| 1 | 0x28, 0xC9, 0x6D, 0x5D, 0x4, 0x0, 0x0, 0x31, |
| 2 | 0x28, 0xAD, 0xC2, 0x8D, 0x3, 0x0, 0x0, 0x9, |
| 3 | 0x28, 0xD5, 0x95, 0x5D, 0x4, 0x0, 0x0, 0x8F, |
| 4 | 0x28, 0x9A, 0x30, 0x5D, 0x4, 0x0, 0x0, 0x3F, |
| 5 | 0x28, 0xBA, 0x1F, 0x5D, 0x4, 0x0, 0x0, 0x1E, |
| 6 | 0x28, 0xD4, 0x1C, 0x5D, 0x4, 0x0, 0x0, 0x80, |
| 7 | 0x28, 0x5, 0x86, 0x5D, 0x4, 0x0, 0x0, 0x79, |
| 8 | 0x28, 0x12, 0x51, 0x5D, 0x4, 0x0, 0x0, 0xA8, |
| 9 | 0x28, 0x2B, 0x1D, 0x5D, 0x4, 0x0, 0x0, 0x1B, |
| 19 | 0x28, 0x4E, 0x28, 0x5D, 0x4, 0x0, 0x0, 0x65, |
| 11 | 0x28, 0xCB, 0x5B, 0x5D, 0x4, 0x0, 0x0, 0x47, |
| 12 | 0x28, 0x81, 0x72, 0x5D, 0x4, 0x0, 0x0, 0xF6, |
| 13 | 0x28, 0xB1, 0x8D, 0x5D, 0x4, 0x0, 0x0, 0xD2, |
| 14 | 0x28, 0x93, 0x5B, 0x5D, 0x4, 0x0, 0x0, 0xC8, |
| 15 | 0x28, 0x10, 0x5D, 0x5D, 0x4, 0x0, 0x0, 0xE7, |
| Otoczenie | 0x28, 0xFF, 0x5A, 0xFC, 0x78, 0x18, 0x1, 0xD, |
| Zimna strona | 0x28, 0xFF, 0xFC, 0xC1, 0x78, 0x18, 0x1, 0x77 |

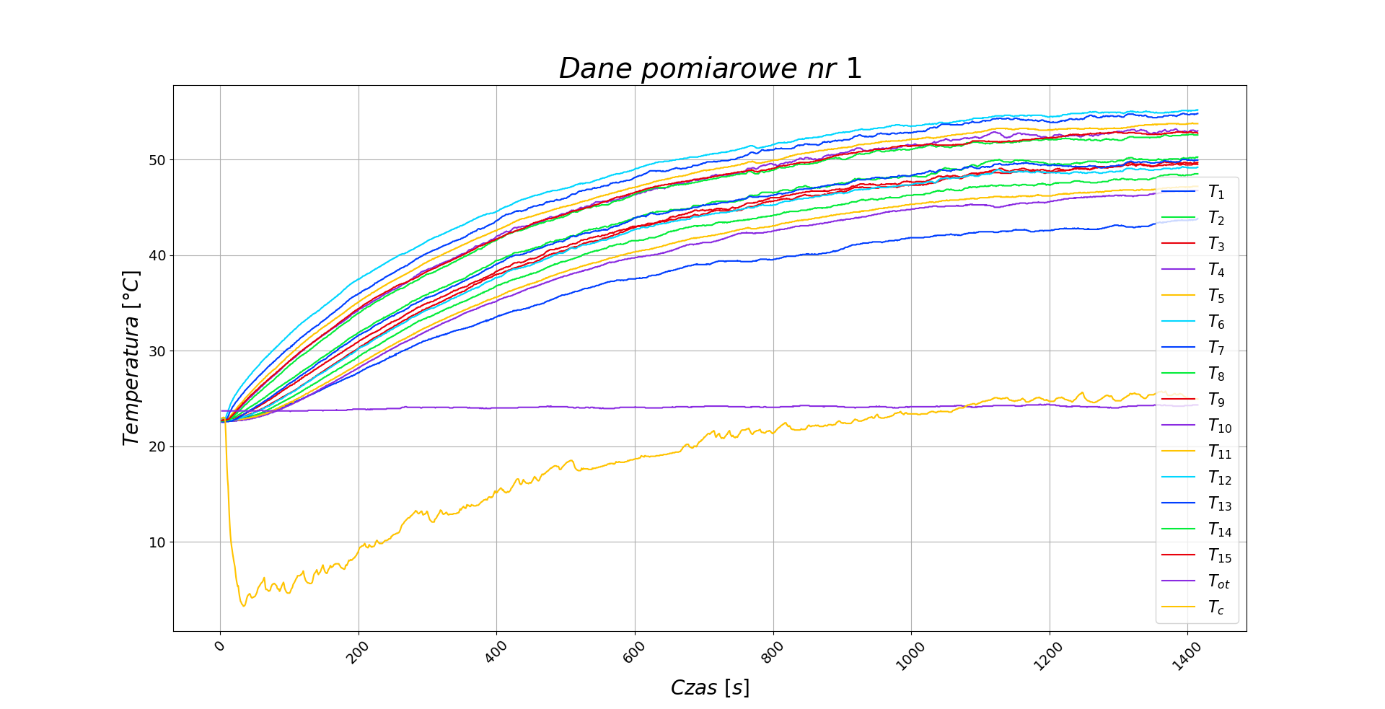
Drugi program służy do odczytu temperatury i wypisywania temperatury w monitorze portu szeregowego. Została w nim zdefiniowana tablica adresów, numer wejścia cyfrowego (*D10*) pod który podłączona jest linia danych czujnika. Dodatkowo można wybrać rozdzielczość czujnika definiując zmienną quality. Prędkość portu szeregowego jest ustawiona na *115200* baud. Wypisywanie pomiarów dobywa się w formacie *csv.* Do odczytywania danych można wykorzystać skrypt w Pythonie lub na przykład program Telemetry Viewer. Napisany skrypt w Pythonie read\_temperature.py pozwala na zapisywanie i jednoczesne wykreślanie pomiarów. Wymaga on dodatkowych bibliotek jak matplotlib, NumPy oraz Pandas. Program nazywa pliki zgodnie z czasem rozpoczęcia pomiarów, z dokładnością do sekundy co zapobiega nadpisaniu pliku pomiarowego.

Po złożeniu całego układu i wgraniu programu do pamięci mikrokontrolera pomiar można rozpocząć wywołując w terminalu skrypt z poprawnie podanymi parametrami w pliku.

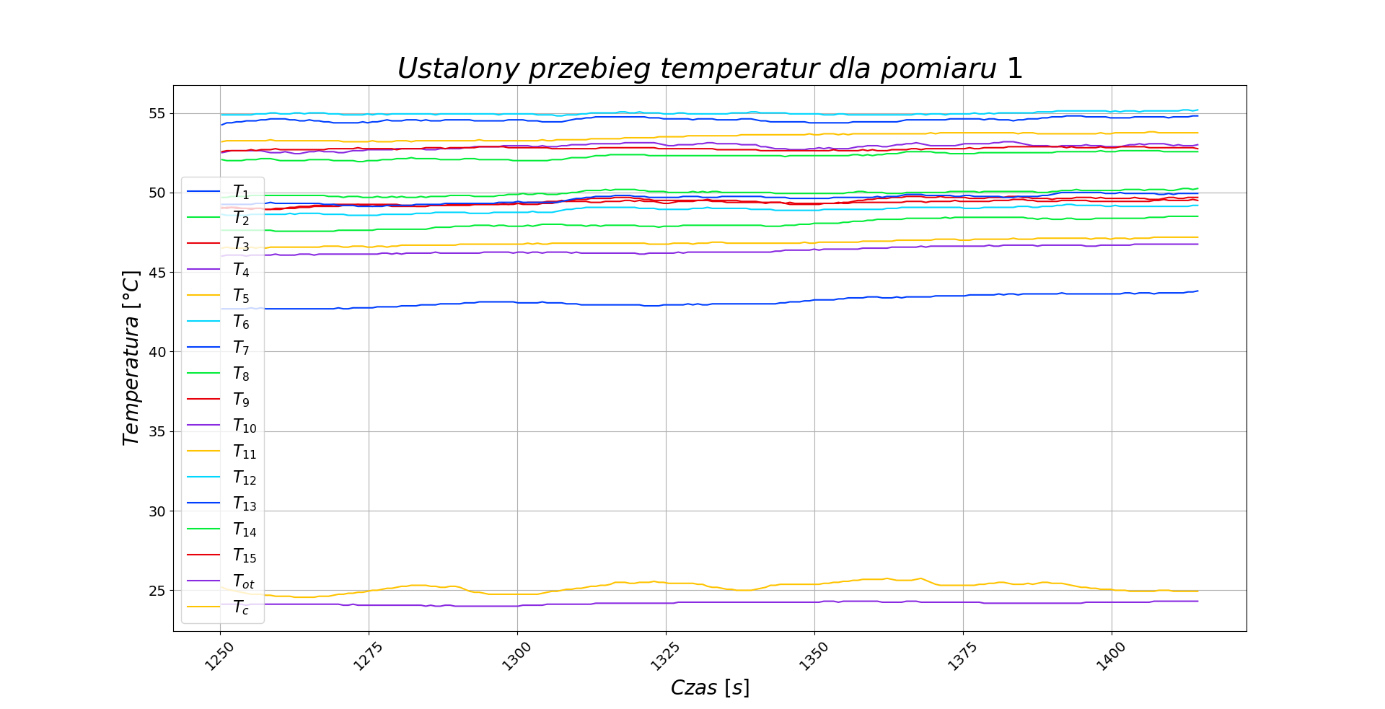
## Wyniki pomiarów

W ramach projektu dokonano dwóch pomiarów, które zostaną poddane analizie. Wyniki pomiarów zostaną przedstawione poniżej, a pliki pomiarowe zostaną załączone w załącznikach.

W przypadku pierwszego pomiaru, przeprowadzono go z wykorzystaniem mniejszego napięcia zasilania w celu osiągnięcia odpowiedniej temperatury i uzyskania ustalonego (stacjonarnego) stanu w celu wykorzystania go do symulacji pozwalającej na wyznaczenie parametrów. Na kolejnych rysunkach przedstawione zostaną dane pomiarowe z pierwszego pomiaru, ich stan ustalony, różnica temperatur modułu grzewczego a także podane zostaną wyznaczone parametry statystyczne i zapisane warunki pomiaru.

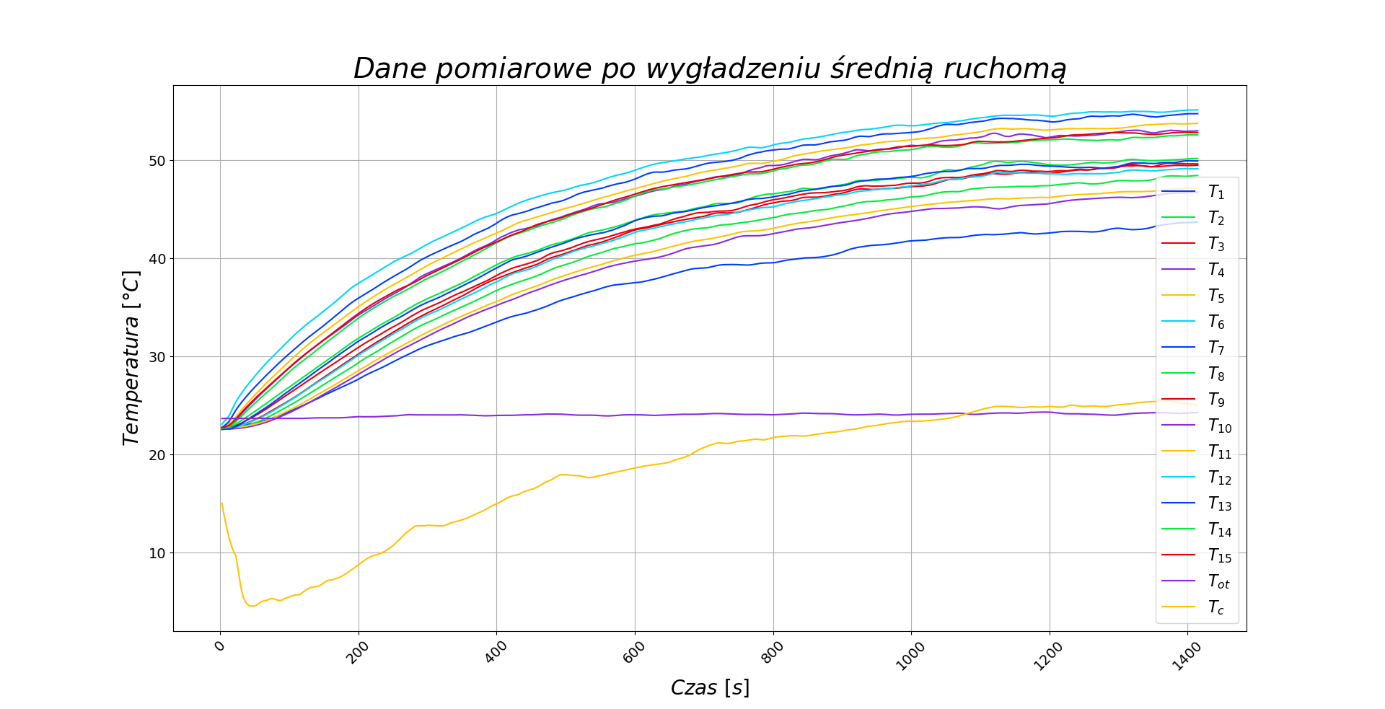


Rysunek . Wykres danych pomiarowych z pierwszego pomiaru

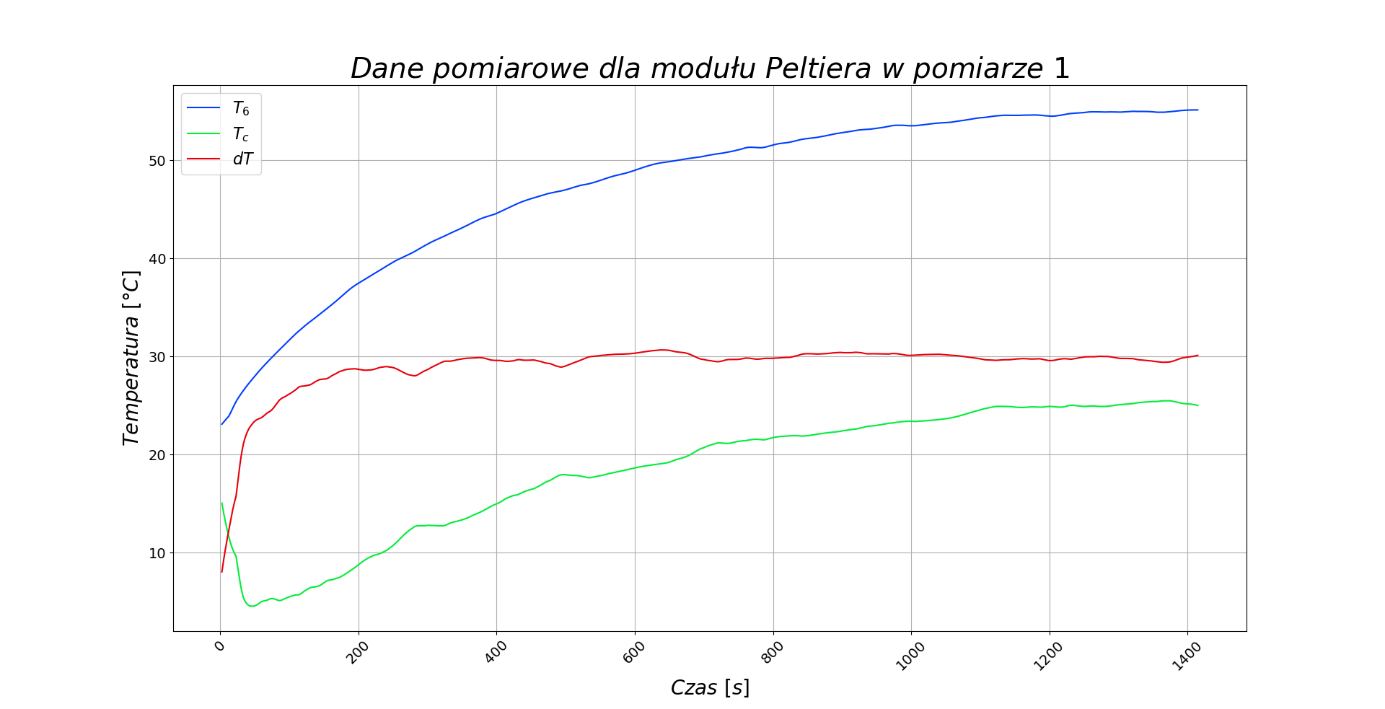


Rysunek . Stan ustalony badanego obiektu

Jak widać w stanie ustalony temperatury poszczególnych punktów pomiarowych nie zmieniały się, dlatego przyjęto, że ten stan obiektu można zasymulować modułem Steady State Thermal. Zostały odczytane średnie wartości temperatur a następnie wprowadzone zostały one do symulacji. Na kolejnych wykresach przedstawiony zostanie wygładzony przebieg danych pomiarowych a także wykres pokazujące stała różnice temperatury pomiędzy zimną i ciepłą stroną modułu, co wskazuje na stały strumień ciepła przepływający do płytki.



Rysunek . Dane pomiarowe poddane wygładzaniu przebiegu za pomocą średniej ruchomej

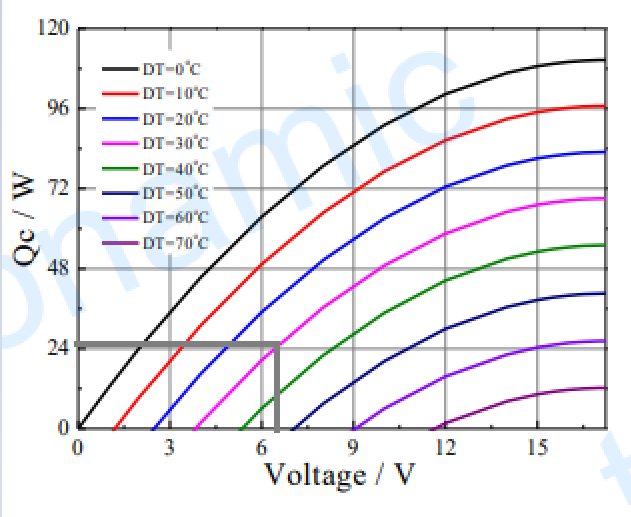


Rysunek . Wykres przebiegu różnicy temperatury na module grzewczym

W celu określenia pewnych parametrów dokonano uśrednienia odpowiednich przebiegów po ich stabilizacji za pomocą skryptu. Wyniki przedstawione zostaną w tabeli 3. Moc cieplna została wyznaczona z charakterystyki modułu.

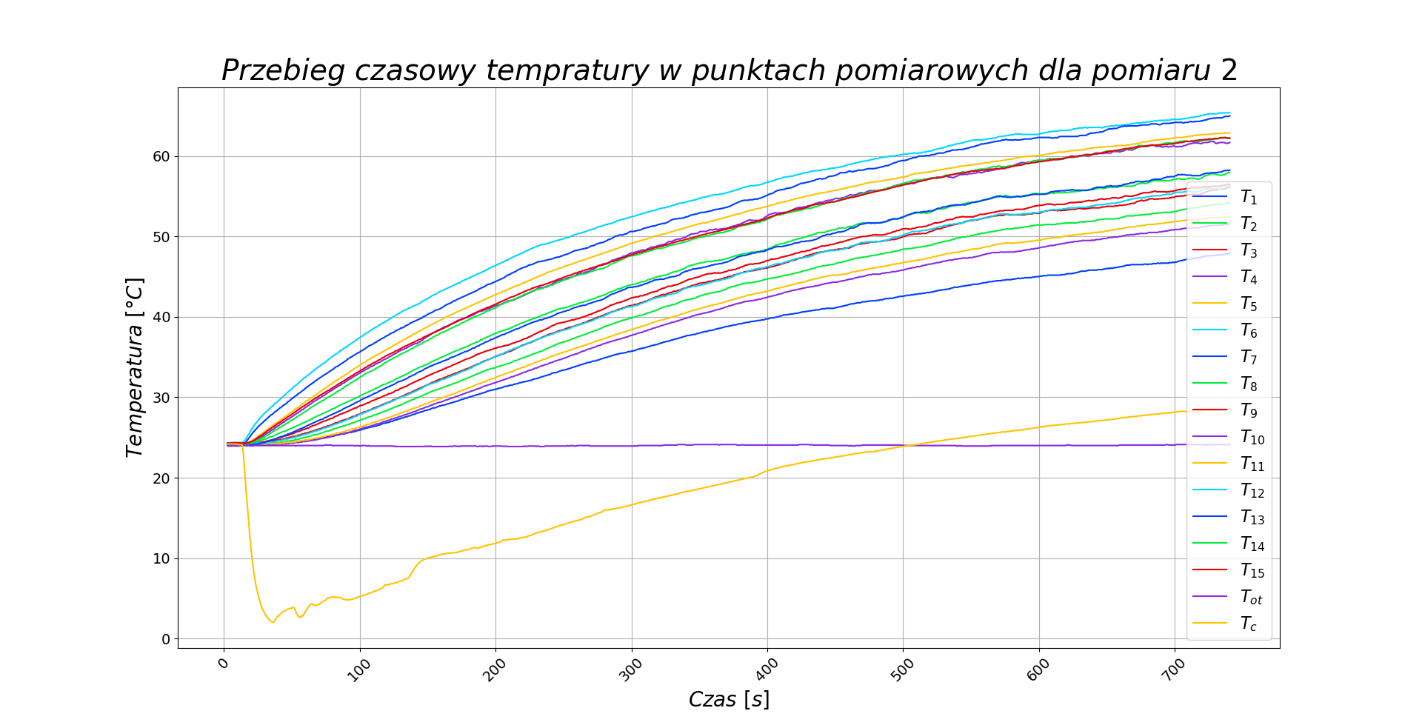
Tabela . Parametry pierwszego pomiaru

|  |  |
| --- | --- |
| Parametr | Wartość |
| Temperatura otoczenia (termometr lab.) [°C] | 22 |
| Napięcie zasilania [V] | 6,2 |
| Natężenie prądu [A] | 3 (2,6 od 700 [s]) |
| Średnia różnica temp. modułu dla ustalonej pracy [°C] | 29,92 |
| Średnia temperatura otoczenia podczas pomiaru [°C] | 24,05 |
| Średnia temperatura płytki w chwili startu podczas pomiaru [°C] | 22,67 |
| Moc cieplną modułu dla stanu ustalonego [W] | 25 |

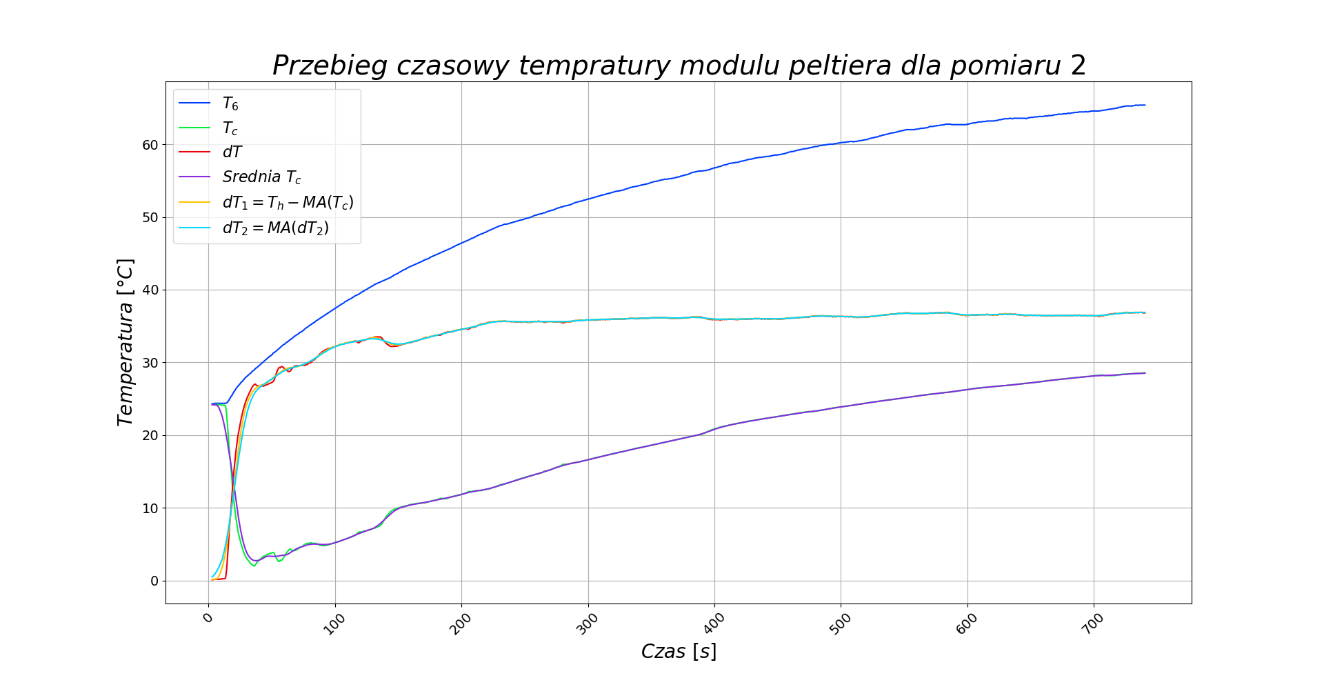


Rysunek . Moc cieplna dla pierwszego pomiaru

Drugi pomiar został przeprowadzony dla większego napięcia w celu uzyskania dynamiki obiektu. Walidacja odbędzie się na podstawie tego pomiaru i porównane zostaną wyniki symulacji i uzyskane rzeczywiste wyniki. Widoczny na poniższym rysunku jest fakt, iż nie został uzyskany stan stacjonarny jak w przypadku pierwszego pomiaru. Podobnie jak poprzednio przedstawione zostaną wykresy danych a także tabela parametrów pomiarów.



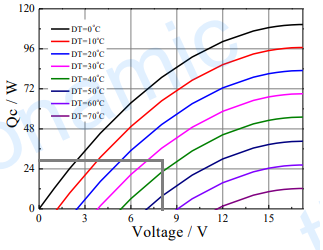
Rysunek . Dane pomiarowe dla pomiaru drugiego



Rysunek . Wykres przebiegu różnicy temperatury modułu

Tabela . Parametry drugiego pomiaru

|  |  |
| --- | --- |
| Parametr | Wartość |
| Temperatura otoczenia (termometr lab.) [°C] | 22 |
| Napięcie zasilania [V] | 8 |
| Natężenie prądu [A] | 4 ( -100 [s])  3,9 (100 – 140 )  3,8 (140 – 200)  …  3,4 (440 – 530)  3,35 (530 – 750)  3,3 (750 - ) |
| Średnia różnica temp. modułu dla ustalonej pracy [°C] | 36,34 |
| Średnia temperatura otoczenia podczas pomiaru [°C] | 24,0 |
| Średnia temperatura płytki w chwili startu podczas pomiaru [°C] | 24,21 |
| Moc cieplną modułu dla stanu ustalonego [W] | 30 |



Rysunek . Moc dla drugiego pomiaru

# Model numeryczny i jego analiza

W tym rozdziale przedstawione zostały: model geometryczny, siatka elementów skończonych i analiza ich jakości, proces identyfikacji parametrów oraz walidacja uzyskanych parametrów.

## Model geometryczny, siatka elementów skończonych

Zdjęcie modelu geometrycznego, jego wymiary, zdjęcie siatki i jej jakość

## Model numeryczny

Ustawienia w symulacji, warunki brzegowe itd

## Identyfikacja parametrów

Ustawienia w module optymalizacji i wyniki dla 1 pomiaru

## Walidacja poprawności modelu z wyznaczonymi parametrami

Wyniki modelu dla paramentów z 3.3 i 2 pomiaru temperatury, porównanie

# Wnioski

Wnioski