

# Spis treści

1. Wstęp . . . . .	2
2. Komunikacja z urządzeniami pomiarowymi na platformie Linux . . . . .	4
2.1. Programowane urządzenia pomiarowe . . . . .	4
2.2. Komunikacja . . . . .	4
2.3. SCPI — standard komend do komunikacji z urządzeniami . . . . .	4
3. Program do sterowania pomiarami charakterystyk laserów półprzewodnikowych z wykorzystaniem sprzętu Thorlabs . . . . .	6
3.1. Wstęp . . . . .	6
3.2. Krótki opis najważniejszych klas . . . . .	6
3.3. Wersja skryptowa programu . . . . .	8
3.4. Wersja okienkowa programu do pomiarów . . . . .	9
4. Lasery półprzewodnikowe . . . . .	11
4.1. Teoria . . . . .	11
4.1.1. Prąd progowy . . . . .	11
4.1.2. Sprawność . . . . .	12
4.1.3. Wpływ temperatury chłodnicy lasera na jego parametry . . . . .	12
4.1.4. Funkcja Fermiego . . . . .	13
5. Opis eksperymentu . . . . .	15
5.1. Układ pomiarowy . . . . .	15
5.1.1. Przebieg pomiarów . . . . .	15
5.2. Laser 635 nm . . . . .	16
5.3. Laser VCSEL 850 nm . . . . .	19
5.4. Omówienie wyników . . . . .	24
5.5. Laser krawędziowy 850 nm . . . . .	25
5.6. Laser VCSEL 980 nm . . . . .	26
5.7. Porównanie laserów . . . . .	27
Bibliografia . . . . .	30

# 1. Wstęp

Niniejsza praca dotyczy zakresu inżynierii oprogramowania sprzętu pomiarowego w celu wykorzystania go w badaniu charakterystyk laserów półprzewodnikowych w laboratorium fotoniki Politechniki Łódzkiej.

Głównym celem pracy jest przedstawienie wykorzystania systemu Linux oraz oprogramowania open source w badaniach naukowych na przykładzie stworzenia interfejsu pomiarowego w laboratorium fononiki do badania charakterystyk laserów półprzewodnikowych.

W ostatnich latach obserwuje się gwałtowny rozwój wykorzystania oprogramowania open source w codziennej pracy naukowej. Coraz większą popularność zdobywa język Python. Od dawna podstawowym systemem operacyjnym używanym przez naukowców są różne odmiany systemu Unix. Jest to spowodowane dostępnością wielu narzędzi (C, Python, Gnuplot) których naturalnym środowiskiem jest środowisko Linux, ułatwiającym pracę naukową. Inną zaletą środowiska Unix jest możliwość korzystania z linii poleceń, która ułatwia wiele zadań. Szukając informacji o wykorzystaniu języka Python do komunikacji ze sprzętem pomiarowym można zauważyć pewną lukę, którą moja praca ma cel wypełnić. Korzystając z strony oraz dokumentacji firmy Thorlabs, której sprzęt jest używany w laboratorium fononiki, należy zauważyć brak programu do komunikacji ze sprzętem na platformie Linux. Dostępne są jedynie wysokopoziomowe API do systemu Windows oraz możliwość użycia LabVIEW. Minusów środowiska Windows nie sposób wymienić w kilku zdaniach. Program LabVIEW jest programem płatnym. Rozwiązaniem wszystkich problemów jest użycie środowiska Linux, gdzie wszystko jest plikiem, także sprzęt połączony przez usb z komputerem, dzięki czemu możemy się z nim komunikować używając standardu komend SCPI przez wykorzystanie wywołań systemowych. Dzięki temu mamy możliwość dostępu do wszystkich możliwych funkcji sprzętu pomiarowego bez ponoszenia kosztów. Umożliwia nam to sterowanie sprzętu za pomocą komputera oraz wizualizację i analizę danych w sposób, jaki potrzebujemy. A wszystko to dzięki połączeniu możliwości środowiska Linux oraz języka Python

Głównymi celami mojej pracy jest przedstawienie wykorzystania oprogramowania open source takiego jak Python, C/C++ oraz systemu Linux do stworzenia stanowiska pomiarowego w celu badania laserów półprzewodnikowych. Korzystając z tych technologii mam zamiar stworzyć interfejs pomiarowy na platformę Ubuntu w laboratorium fotoniki.

Dzięki mojej pracy możliwe będzie wykonywanie w szybki sposób charakterystyk laserów półprzewodnikowych. Charakterystyki te dają nam ważne informacje o laserze, dzięki nim możliwe jest określenie prądu progowego dla laserów krawędziowych, określenie ich sprawności. Za pomocą mojego stanowiska pomiarowego możliwe także będzie badanie ile uzyskuje się mocy z lasera przy danej mocy aplikowanej.

WYDZIAŁ FIZYKI TECHNICZNEJ, INFORMATYKI I MATEMATYKI  
STOSOWANEJ

Instytut Fizyki PŁ,

~~Praca jest podzielona na dwie części: jedna składa się z opisu przygotowania ekspe-~~  
rymentu, komunikacji oraz sterowaniem urządzeniami laboratoryjnymi za pomocą  
programu napisanego w języku Python. Druga część pracy opisuje badanie laserów  
półprzewodnikowych na podstawie danych uzyskanych za pomocą programu przed-  
stawionego w pierwszej części programu. Do wykreślenia charakterystyk wyjścio-  
wych oraz wyznaczenie sprawności badanych laserów używam skryptów napisanych  
w języku Python.

## 2. Komunikacja z urządzeniami pomiarowymi na platformie Linux

### 2.1. Programowane urządzenia pomiarowe

Przez programowane urządzenia pomiarowe rozumiemy sprzęt mogący dokonywać pomiarów wielkości elektrycznych i nieelektrycznych, który wyposażony jest w interfejs umożliwiający sterowanie nimi przy pomocy komputera. Przykładami takich urządzeń, którymi zajmuje się w swojej pracy są:

- Zasilacza diód laserowych firmy Thorlabs model LDC4005.
- Miernik mocy firmy Thorlabs model PM100.

Z wyżej wymienionymi urządzeniami możliwa jest fizyczna komunikacja za pomocą interfejsu USB przy pomocy standardu komend SCPI, który zostanie opisany w dalszej części rozdziału.

### 2.2. Komunikacja

W systemach Unix z którego dziedziczy system Linux, wszystko jest plikiem. Linuksowy sterownik znakowy (ang. char driver) pozwala na reprezentowanie urządzenia za pomocą specjalnych plików wirtualnych, które znajdują się w przestrzeni użytkownika w katalogu `/dev/ <nazwa>`. Obsługa tych plików możliwa jest za pomocą wywołań systemowych (ang. system call), które stanowią API za pomocą którego użytkownicy może sterować sprzętem. Podstawowe wywołaniami systemowymi pozwalające na sterowanie sprzętem to:

- `open` — służy do połączenia z urządzeniem, zwraca deskryptor pliku.
- `write` — funkcja służąca do wysyłania komend do urządzenia.
- `read` — funkcja służąca do odczytywania buffora urządzenia.
- `close` — funkcja zamykająca połączenie.

Funckje te mają swoją implementację w języku C w bibliotece `<fcntl.h>`, oraz w języku Python w bibliotece `os`.

### 2.3. SCPI — standard komend do komunikacji z urządzeniami

SCPI (ang. Standard Commands for Programmable Instruments) jest tekstowym interfejsem ASCII do programowanych urządzeń pomiarowych mający na celu standaryzację poleceń używanych w systemach pomiarowych. Zdefiniowany został 1990 roku, wedle specyfikacji IEEE 488.2. (Institute of Electrical and Electronics Engineers, międzynarodowa organizacja stowarzyszeń inżynierów elektryków i elektroników. Dzięki temu możliwa jest obsługa tych urządzeń przy wykorzystaniu komputera. Polecenia SCPI są to ciągi tekstowe ASCII, które są wysyłane do urządzenia.

Polecenia są serią jednego lub więcej słów, przy czym wiele z nich używa dodatkowych parametrów. Odpowiedzi do zapytania polecenia są zazwyczaj ciągami ASCII. W przypadku danych masowych mogą być używane także formaty binarne.

Cechą poleceń wspólnych (ang. common) jest ich implementacja przez każde urządzenie. Czyli na przykład to samo polecenie będzie działać na każdym oscyloskopie bez względu na producenta. Można wyróżnić dwie grupy poleceń:

- Polecenia dla każdego urządzenia pomiarowego nie zależnie od jego przeznaczenia. Takimi komendami są m.in.
  - `*idn?` — odczytuje identyfikator urządzenia.
  - `*rst` — powoduje przywrócenie ustawień początkowych urządzenia.
  - `*cls` — powoduje wyzerowanie informacji o błędach.
  - `*opc?` — (ang. operation complete) zapytanie o zakończenie wykonania poprzedzających poleceń.  
W odpowiedzi na zapytanie po zakończeniu wykonywania poprzedzających poleceń urządzenie prześle wartość 1.
  - `*wai` — (ang. wait) oczekiwanie na zakończenie wykonania poprzedzających poleceń.
- Polecenia charakterystyczne dla danego urządzenia pomiarowego zgodnie z jego przeznaczeniem. Przykładowe polecenie które będzie działać na każdym zasilaczu korzystającym z standardu SCPI:
  - Służące do ustawienie wartości prądu na 0.01 A  
`SOURce : CURRent : LEVel : AMPLitude 0.01`

Fizyczne łącze komunikacyjne nie jest zdefiniowane przez SCPI. Stworzony standard IEEE-488 był dla GPIB, ale może być również używany z interfejsem RS-232, Ethernet, USB, VXIbus.

### 3. Program do sterowania pomiarami charakterystyk laserów półprzewodnikowych z wykorzystaniem sprzętu Thorlabs

#### 3.1. Wstęp

W ramach pracy inżynierskiej zostały stworzone programy do sterowania pomiarami charakterystyk laserów. Program został napisany w dwóch wersjach: skryptowej oraz okienkowej. Podstawą działania programów są następujące klasy:

- `device.py` — główna klasa, zawiera funkcje: do sprawdzania dostępnych urządzeń, zwraca instancje danego urządzenia, co umożliwia korzystanie z jego funkcji.
- `IIODevice.py` — klasa do operacji wejścia-wyjścia na programowalnych urządzeniach pomiarowych.
- `LDC4005.py` — klasa zawierająca funkcje od obsługi zasilacza diód laserowych Thorlabs 4005.
- `PM100.py` — klasa zawierająca funkcje do obsługi detektora mocy Thorlabs PM100.

#### 3.2. Krótki opis najważniejszych klas

W tym podrozdziale przedstawię kody programu do sterowania pomiarami charakterystyk laserów półprzewodnikowych. W zaprezentowanych kodach przedstawiam tylko najważniejsze funkcje, aby ułatwić czytelność (pełny kod jest dołączony do pracy).

Pierwszy listing przedstawia klasę do operacji wejścia-wyjścia na urządzeniach programowalnych. Jest to podstawowa klasa, która następnie używana jest w klasach do sterowania zasilaczem diod laserowych - `LDC4005.py` (listing 2) oraz miernika mocy - `PM100.py` (listing 3). Ostatni kod (listing 4) pokazuje przykładowy skrypt w języku Python 3, który wykonuje pomiar charakterystyki lasera.

```
1 import os
2
3
4 class IODevice:
5     def __init__(self, path_to_device):
6         self.device = path_to_device
7         self.file_descriptor = os.open(path_to_device, os.O_RDWR | os.O_NOCTTY)
8
9     def write(self, command):
10        os.write(self.file_descriptor, command.encode())
11
12    def read(self, length=4000):
13        return os.read(self.file_descriptor, length)
14
15    def close(self):
16        os.close(self.file_descriptor)
17
18    def __del__(self):
19        self.close()
```

```
1 from IODevice import IODevice
2
3 class PM100:
4     def __init__(self, path_to_usb_path):
5         self.device = IODevice(path_to_usb_path)
6
7     def set_wavelength_in_nm(self, value):
8         self.device.write("CORRection:WAVelength " + str(value))
9
10    def get_power(self):
11        self.device.write("INITiate")
12        self.device.write("MEASure:POWEr")
13        self.device.write("FETCh?")
14        value = self.device.read(100)
15        return float(value)
```

```
1 from IODevice import IODevice
2
3
4 class LDC4005:
5     def __init__(self, path_to_usb_path):
6         self.device = IODevice(path_to_usb_path)
7
8     def on(self):
9         self.device.write("OUTPut ON")
10
11    def off(self):
12        self.device.write("OUTPut OFF")
13
14    def set_ld_current_in_amper(self, value):
15        self.device.write("SOURce:CURRent:LEVel:AMPLitude %s" % value)
16
17    def ld_current_reading(self):
18        self.device.write("INITiate")
19        self.device.write("MEASure:CURRent")
20        self.device.write("FETCh:CURRent?")
21        value = self.device.read(100)
22        return float(value)
```

```
23
24     def ld_voltage_reading(self):
25         self.device.write("INITiate")
26         self.device.write("MEASure:VOLTage")
27         self.device.write("FETCh:VOLTage?")
28         value = self.device.read(100)
29         return float(value)
```

Skrypt przedstawiony na listingu 4 w liniach 1-3 importuje potrzebne moduły, ważne, aby moduły LDC4005.py oraz PM100.py znajdowały się w tym samym folderze co measure.py. W linii 6 deklarujemy tablicę 20 elementów o wartościach prądu od 0 do 0.02 A. Następnie w liniach 8-10 deklarujemy tablice do przechowywania danych, które zostaną zmierzone. W liniach 13-14 tworzy instancje klasy dla zasilacza i miernika, zakładamy, że port /dev/usbTMC0 odpowiada zasilaczowi ldc4005, a port /dev/usbTMC1 odpowiada miernikowi mocy PM100. Następnie w linii 20 ustawiany prąd na zasilaczu, w 21 czytamy prąd z zasilacza, w 22 czytamy napięcie, a w 23 czytamy moc wyjściową na mierniku mocy. Ostatni etapem jest zapisanie danych do pliku tekstowego w linii 25-26.

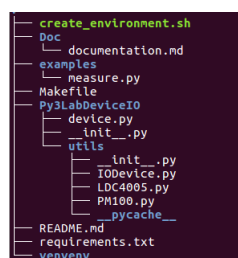
```
1 import numpy as np
2 from LDC4005 import LDC4005
3 from PM100 import PM100
4
5
6 set_current = np.linspace(0, 0.02, 20)
7
8 measure_current = np.zeros(len(set_current))
9 measure_voltage = np.zeros(len(set_current))
10 measure_power = np.zeros(len(set_current))
11
12 # usbtmc0 odpowiadawia ldc, a usbtmc1 pm100
13 ldc = LDC4005("/dev/usbTMC0")
14 pm100 = PM100("/dev/usbTMC1")
15
16 pm100.set_wavelength_in_nm(635)
17 ldc.on()
18
19 for i in range(0, len(set_current)):
20     ldc.set_ld_current_in_amber(set_current[i])
21     measure_current[i] = ldc.ld_current_reading()
22     measure_voltage[i] = ldc.ld_voltage_reading()
23     measure_power[i] = pm100.get_power()
24
25 np.savetxt("dane.txt", np.c_[measure_current, measure_voltage, measure_power]
26 , fmt='%1.16f', header="current [A] \t voltage [V] \t power [W]")
```

### 3.3. Wersja skryptowa programu

Jedną z możliwości przeprowadzania pomiarów jest wykorzystanie skryptu (wraz z innymi klasami dołączony jest do pracy). Struktura programu pokazana jest na rysunku 3.1. Aby zainstalować wszystkie niezbędne biblioteki należy w konsoli wywołać polecenie **make**.

```
1 student@ubuntu:~$ make
```





Rysunek 3.1. Teoria.

Następnie uruchamiamy wirtualne środowisko (linia 1), które posiada wszystkie potrzebne biblioteki, następnie przechodzimy do katalogu `examples`, gdzie znajduje się plik `measure.py`, który należy użyć do przeprowadzenia pomiarów:

```
1 student@ubuntu:~$ source venvenv/bin/activate
2 student@ubuntu:~$ cd examples/
```

Do przeprowadzenia pomiarów wywołujemy skrypt `measure.py` z parametrami:

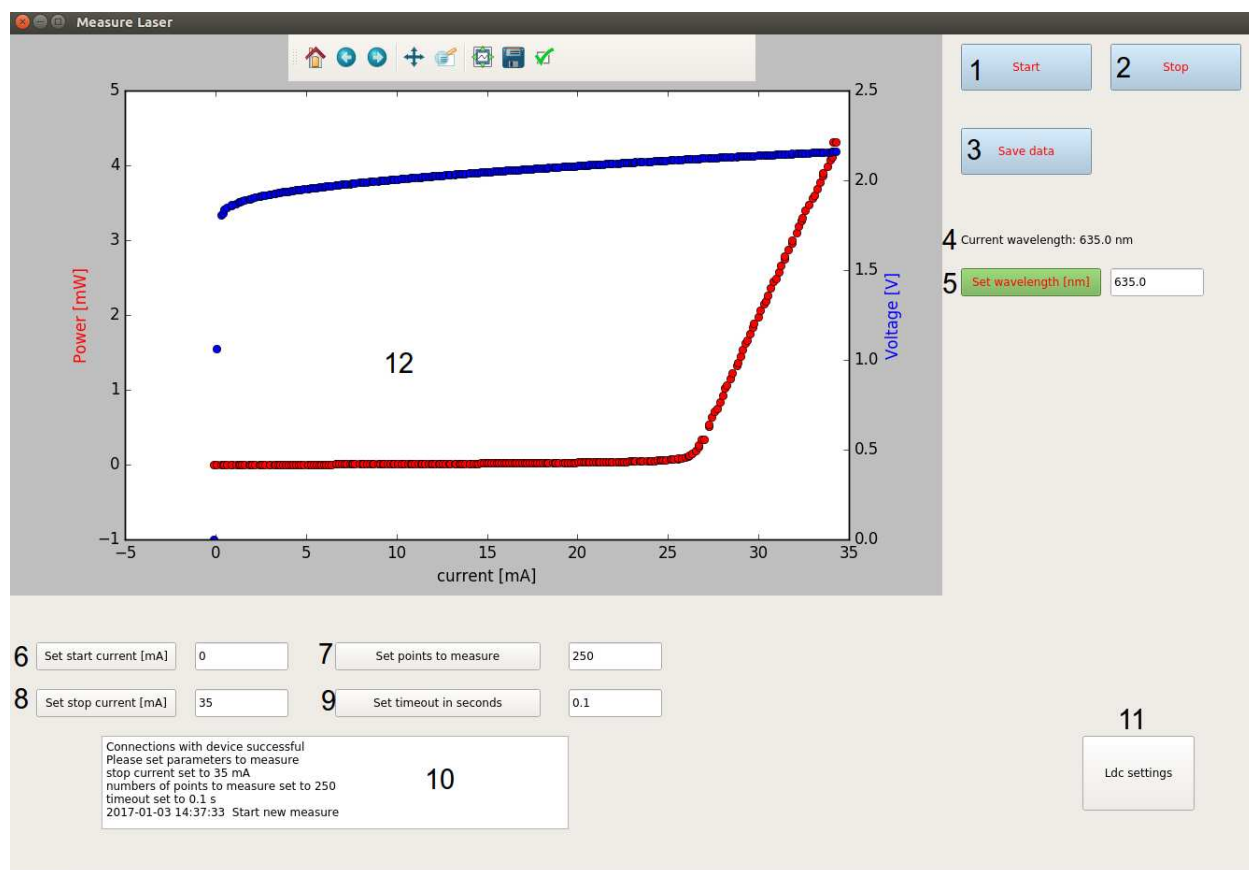
- nr: liczba punktów do pomiaru
- sc: początkowa wartość prądu w mA, od którego należy rozpocząć pomiar
- ec: wartość prądu w mA do którego przeprowadza się pomiar.
- fn: nazwa wynikowa pliku z danymi, która będzie zapisana w katalogu `output`

```
1 student@ubuntu:~$ python3 measure.py -nr 150 -sc 0 -ec 20
2 -fn dane
```

### 3.4. Wersja okienkowa programu do pomiarów

Na rysunku 3.4 przedstawiony jest okienkowy program do wykonywania charakterystyk. Program napisany jest w języku Python z wykorzystaniem bibliotek PyQt5, matplotlib

- Przycisk „Start” [1] służący do rozpoczęcia pomiarów. Po jego wciśnięciu nastąpi wykonanie charakterystyki lasera na podstawie ustawionych parametrów.
- Przycisk „Stop” [2] służący do zatrzymania pomiarów. Po jego wciśnięciu nastąpi wyłączenie zasilacza.
- Przycisk „Save data” [3] służący do zapisania zebranych danych. Po wciśnięciu należy wybrać ścieżkę. Zapis dokonywany jest w formacie txt.
- Wyświetlacz długości fali wybranej na detektorze [4]. Długość fali wyświetlana jest w nanometrach.
- Przycisk służący do zmiany długości fali na detektorze [5]. Wartość należy wprowadzić w nanometrach i zatwierdzić.
- Przycisk „Set start current” [6] — służy do wybrania prądu, od jakiego ma się zacząć pomiar w mA.
- Przycisk „Set points to measure” [7] — służy do wybrania ilości punktów do charakterystyki.
- Przycisk „Set stop current” [8] — służy do wybrania granicy prądu, do jakiego ma się odbyć pomiar w mA.



Rysunek 3.2. 1 — rozpoczęcie pomiarów, 2 — zatrzymanie pomiarów, 3 — zapisanie danych pomiarowych, 4 — pokazuje długość fali detektora, 5 — zmiana długości fali detektora 6 — ustawia prąd początkowy do pomiarów, 7 —

- Przycisk „Set timeout in seconds” [9] — służy do ustawienie długości pauzy między zadaniem prądu do zasilacza, a wykonaniem pomiaru.
- Okienko informacyjne [10] — wyświetla informacje o pomiarze.
- Przycisk „Ldc settings” [11] — ustawia najważniejsze parametry zasilacza diod takie jako wartość maksymalna prądu.

## 4. Lasery półprzewodnikowe

### 4.1. Teoria

#### 4.1.1. Prąd progowy

Wśród laserów półprzewodnikowych możemy wyróżnić lasery krawędziowe oraz lasery o emisji powierzchniowej z pionową wnęką rezonansową tzw. Lasery VCSEL (ang. Vertical Cavity Surface Emitting Laser) będące obiektem moich badań. Aby scharakteryzować lasery, można wykonać ich charakterystyki, które przedstawiają, jak zmienia się moc wyjściowa oraz napięcie lasera w funkcji zadanego prądu.

Ważnym parametrem laserów półprzewodnikowych jest prąd progowy (z ang. threshold current) który określa wartość prądu, przy którym zaczyna zachodzić akcja laserowa, czyli rośnie gwałtownie natężenie promieniowania i maleje szerokość linii emisyjnej. W celu wyznaczenia prądu progowego należy sporządzić wykres zależności mocy wyjściowej lasera od prądu zasilającego. Następnie dla prądu gdzie zaczyna się akcja laserowa dla odcinka liniowego należy metodą najmniejszych kwadratów przy użyciu wielomianu pierwszego stopnia znaleźć parametry krzywej. Dla wyznaczonej krzywej należy znaleźć miejsce zerowe, które będzie wyznaczonym prądem progowym.

$$P = a \cdot I + b \quad (4.1)$$

$$I_{th} = -\frac{b}{a} \quad (4.2)$$

$$\Delta I_{th} = \left| \frac{\partial I_{th}}{\partial a} \right| \cdot \Delta a + \left| \frac{\partial I_{th}}{\partial b} \right| \cdot \Delta b \quad (4.3)$$

$$\Delta I_{th} = \left| -\frac{b}{a^2} \right| \cdot \Delta a + \left| -\frac{1}{a} \right| \cdot \Delta b \quad (4.4)$$

Dla laserów krawędziowych prąd progowy rośnie wraz z temperaturą, co może być scharakteryzowany za pomocą parametru  $T_0$  wyrażonego w kelwinach. Dla laserów krawędziowych zależności prądu progowego  $I_{th}$  od temperatury  $T$  wyrażamy w postaci równania:

$$I_{th} = I_0 \exp\left(\frac{T}{T_0}\right) \quad (4.5)$$

Wartości parametrów  $I_0$  oraz  $T_0$  możemy wyznaczyć na podstawie charakterystyk emisyjnych lasera w różnych temperaturach  $T$ .

Przez zlogarytmowanie wartości prądu oraz podstawienie otrzymujemy:

$$\ln(I_{th}) = \frac{T}{T_0} + \ln(I_0) \quad (4.6)$$

Mając wartości prądu progowego w danej temperaturze można do nich dopasować funkcje liniową w postaci:

$$y = a \cdot T + b \quad (4.7)$$

Gdzie:

$$y = \ln(I_{th}) \quad (4.8)$$

$$a = \frac{1}{T_0} \quad (4.9)$$

$$b = \ln(I_0) \quad (4.10)$$

Na tej podstawie możemy znaleźć poszukiwane parametry  $I_0$  oraz  $T_0$ :

$$I_0 = e^b \quad (4.11)$$

$$T_0 = \frac{1}{a} \quad (4.12)$$

Korzystając z różniczki zupełnej można obliczyć wartości błędów wyznaczonych wartości:

$$\Delta I_0 = \left| \frac{\partial I_0}{\partial b} \right| \cdot \Delta b = |e^b| \cdot \Delta b \quad (4.13)$$

$$\Delta T_0 = \left| \frac{\partial T_0}{\partial a} \right| \cdot \Delta a = \left| -\frac{1}{a^2} \right| \cdot \Delta a \quad (4.14)$$

Dla laserów VCSEL nie można zastawać powyższej zależności, ponieważ zależności prądu progowego od temperatury charakteryzuje się pewnym minimalnym prądem progowym, gdzie w niższych i wyższych temperaturach wartość prądu progowego jest większa.

#### 4.1.2. Sprawność

Innym ważnym parametrem, którym możemy scharakteryzować lasery półprzewodnikowe jest ich sprawność. Można wyróżnić następujące sprawności:

- Sprawność różniczkowa (ang. slope efficiency) — jest zdefiniowana jako nachylenie krzywej uzyskanej przez wykreślenie zależności mocy wyjściowej z lasera versus energii dostarczonej do lasera w postaci natężenie prądu  $I$  lub moc dostarczonej  $P$ . Moc dostarczoną definiujemy jako:

$$P = U \cdot I \quad (4.15)$$

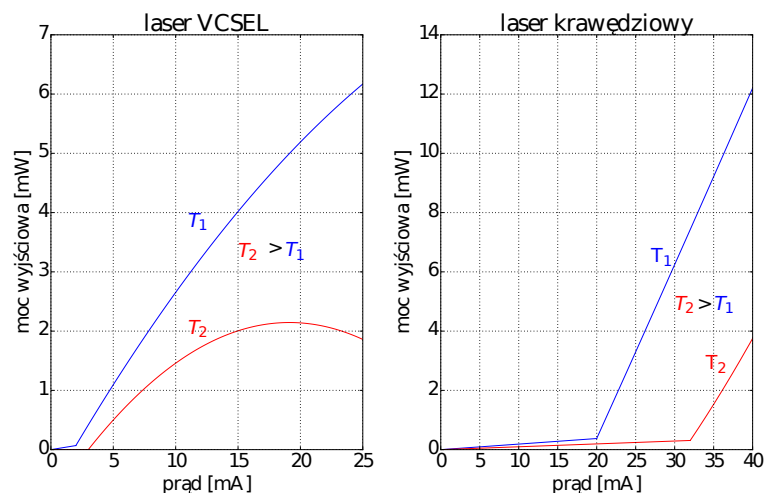
gdzie:  $U$  — napięcie na laserze.

- Sprawność całkowita (ang. Wall-plug-efficiency) — jest zdefiniowana jako stosunek mocy wyjściowej do całkowitej mocy wejściowej lasera.

#### 4.1.3. Wpływ temperatury chłodnicy lasera na jego parametry

Wraz ze wzrostem temperatury wartość prądu progowego  $I_{th}$  rośnie, natomiast sprawność różniczkowa  $\eta$  maleje. Jest to spowodowane przez:

- W wyższych temperaturach funkcja Fermiego-Diraca, która opisuje prawdopodobieństwo zajmowania stanów energetycznych staje się bardziej "rozmarzana". Przez co obrządzenie poziomów energetycznych jest bliższe powłoce przewodzenia dla elektronów oraz bliższe powłocy walencyjnej dla dziur. Dzięki temu możliwość wzmocnienia promieniowania lasera na długości fali emitowanej jest zredukowane.



Rysunek 4.1. Teoria.

- Dodatkowo w podwójnym aktywnym regionie heterostruktury, rozkład energii elektronów i dziur w wyższych temperaturach jest przesunięty dalej od krawędzi pasma, przez co zwiększa się prawdopodobieństwo pobytu ładunków w aktywnym regionie, co powoduje obniżenie sprawności  $\eta$
- Wraz ze wzrostem temperatury rośnie współczynnik Auger, obniżając wydajność i czas życia ładunku podczas przejścia promienistego, co zwiększa wartość prądu progowego  $I_{th}$ .

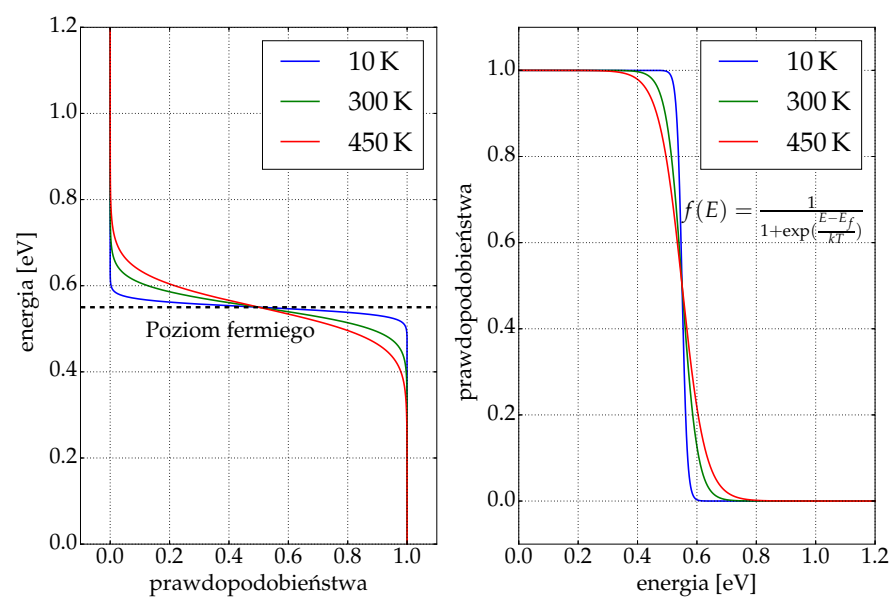
Sprawność różniczkowa zarówno dla laserów VCSEL i krawędziowych zwiększa się wraz z spadkiem temperatury chłodnicy lasera. Jest to spowodowane wyostreniem (ang. sharpening) się widma wzmocnienia na skutek zwężania się funkcji rozkładu Fermiego.

#### 4.1.4. Funkcja Fermiego

Na wygląd charakterystyk laserów półprzewodnikowych duży wpływ ma funkcja rozkładu Fermiego-Diraca, który opisuje własności półprzewodników jest ona zależna od temperatury.

$$f(E) = \frac{1}{e^{(E-E_F)/kT} + 1} \quad (4.16)$$

Poziom fermiego reprezentuje średnią pracę którą należy wykonać aby usunąć elektron z materiału.



Rysunek 4.2. Teoria.

## 5. Opis eksperymentu

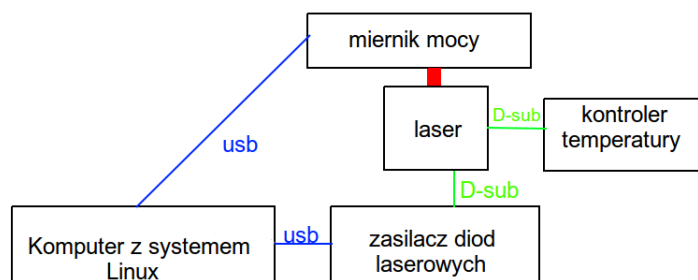
### 5.1. Układ pomiarowy

Układ pomiarowy składał się z:

- Komputera z systemem Linux (Ubuntu) — wymagany jest, aby na komputerze zainstalowany był język Python wraz z bibliotekami: matplotlib, numpy, PyQt5. Do sterowania sprzętem wymagane są uprawnienia administratora..
- Zasilacza diód laserowych firmy Thorlabs model LDC4005 [1] — zapewnia stabilne zasilanie prądowe laserów prądem do 5 A. Możliwe jest zasilanie ciągłe i impulsywne. Posiada interfejs SCPI [2], umożliwiający sterowanie za pomocą komputera przez USB.
- Miernik mocy firmy Thorlabs model PM100 [3] — stworzony do mierzenia mocy wyjściowej z lasera. Pozwala operować na długościach fali od 400 nm do 1100 nm. Posiada interfejs SCPI [3], umożliwiający sterowanie za pomocą komputera przez USB.
- Kontroler temperatury diód laserowych firmy Thorlabs — precyzyjny kontroler temperatury pozwalający na zmiany temperatury chłodniczy lasera podczas operowania prądami do 2 A.

#### 5.1.1. Przebieg pomiarów

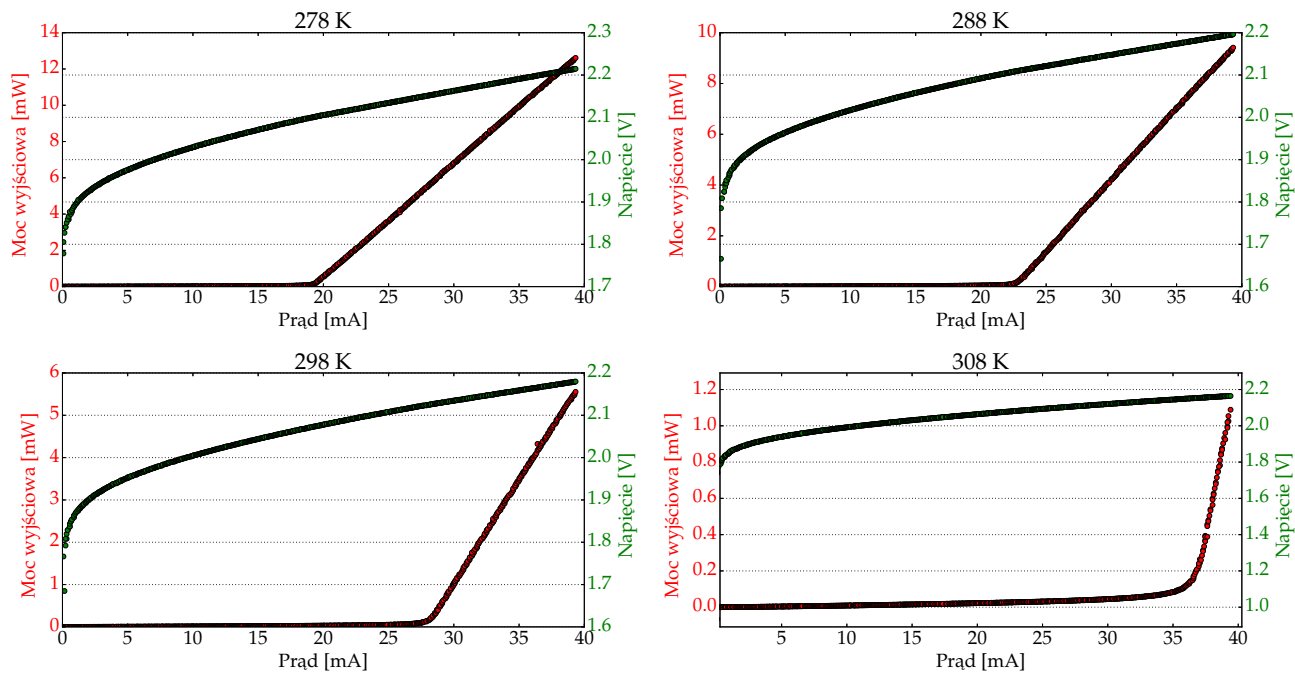
Laser był umieszczony w mocowaniu diód laserowych połączonych z zasilaczem diód laserowych oraz kontrolerem temperatury. Na wyjściu lasera umieszczony był miernik mocy. Komunikacja z zasilaczem oraz miernikiem odbywała się za pomocą standardu komend SCPI przez połączenie USB. Temperatura była zmieniana manualnie na kontrolerze temperatury. Charakterystyki wyjściowe (czyli wartości prądu zasilania, napięcia na laserze oraz mocy wyjściowej) mierzone były pod falą ciągłą prądu. Wyniki zapisywane były w pliku tekstowym.



Rysunek 5.1. Schemat układu pomiarowego.

Tabela 5.1. Wyznaczone wartości prądu progowego  $I_{th}$  w różnych temperaturach  $T$  dla lasera krawędziowego 635 nm.

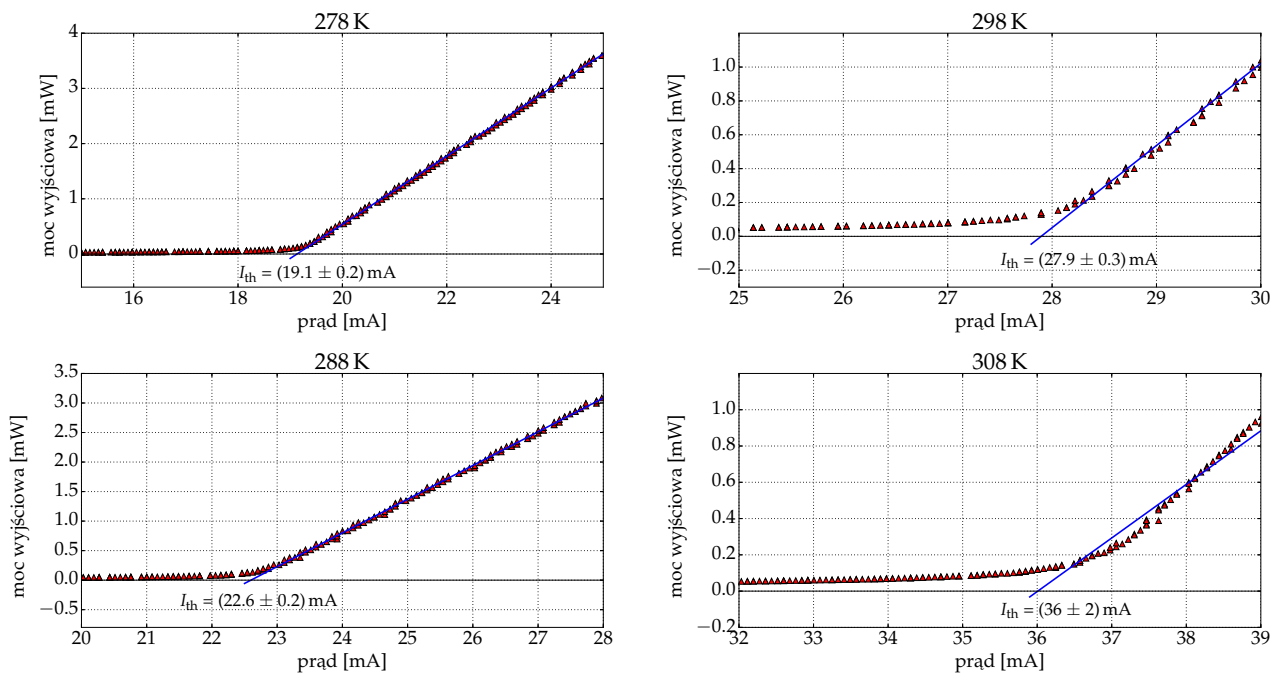
$T$ [K]	278	283	288	293	298	303	308
$I_{th}$ [mA]	$19.1 \pm 0.2$	$20.7 \pm 0.2$	$22.6 \pm 0.2$	$25.0 \pm 0.2$	$27.9 \pm 0.3$	$31.4 \pm 0.5$	$36 \pm 2$



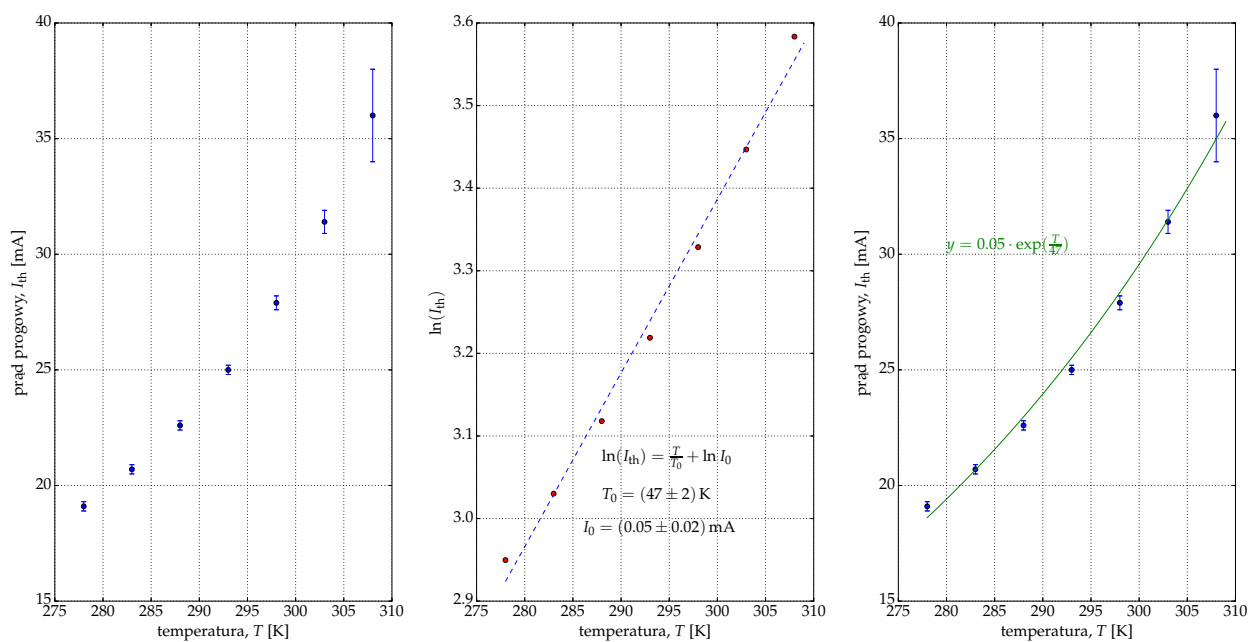
Rysunek 5.2. Wykres napięcia i mocy od prądu dla lasera krawędziowego 635 nm.

## 5.2. Laser 635 nm

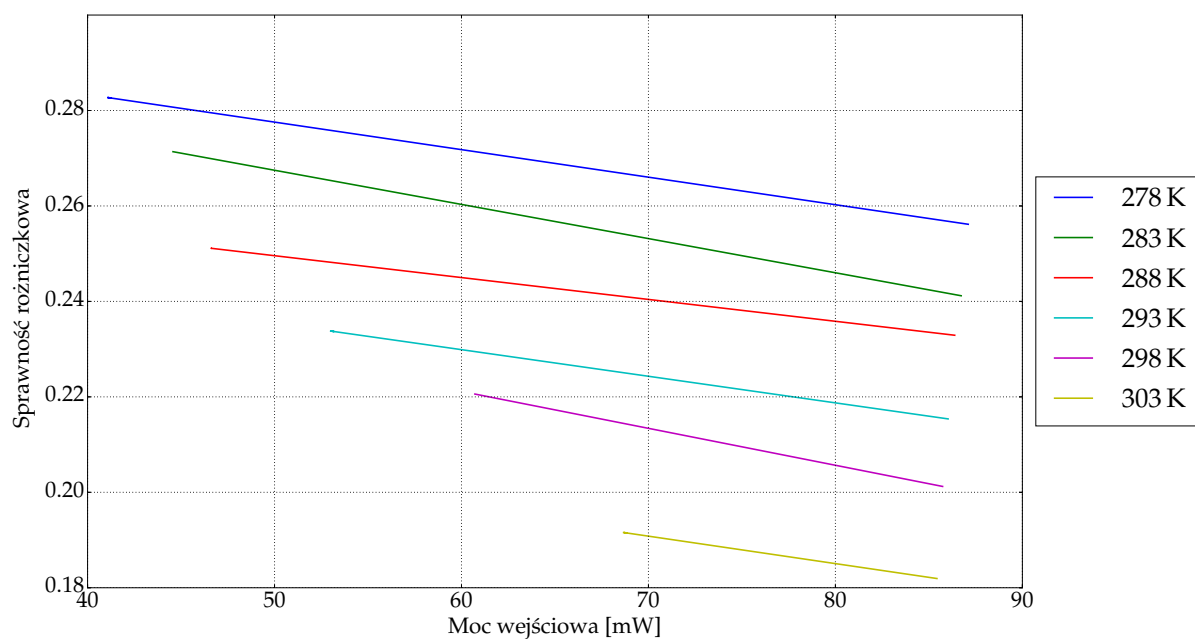




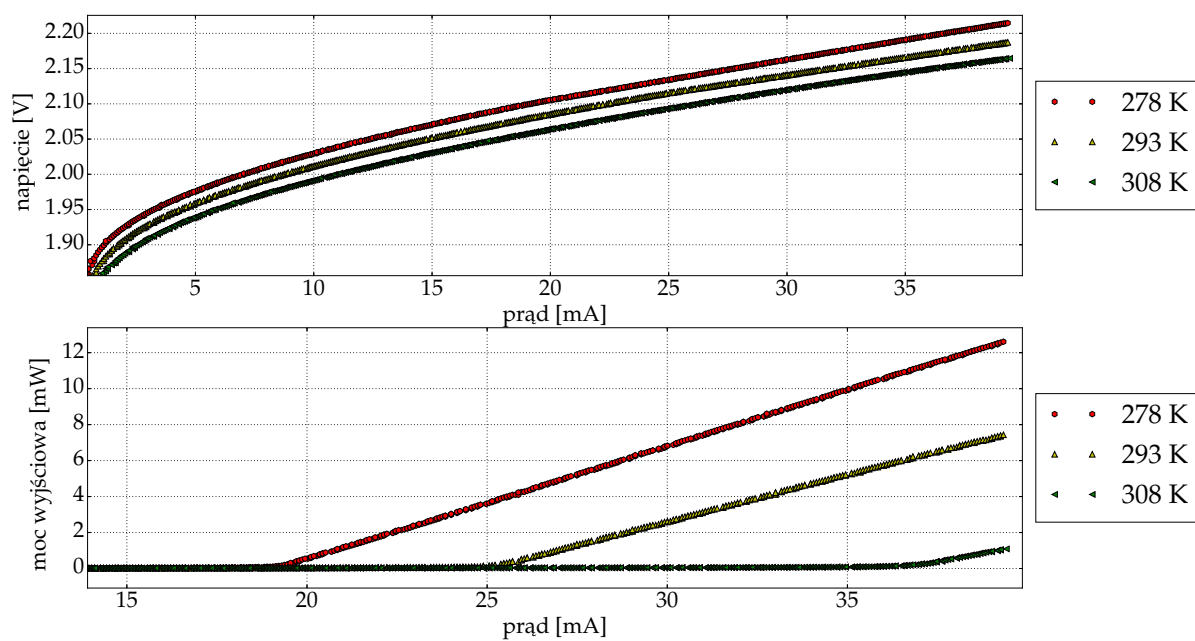
Rysunek 5.3. Wykres prądu progowego dla lasera krawędziowego 635 nm.



Rysunek 5.4. Wykres prądu progowego z dopasowanymi wartościami  $I_0$  i  $T_0$  dla lasera krawędziowego 635 nm.



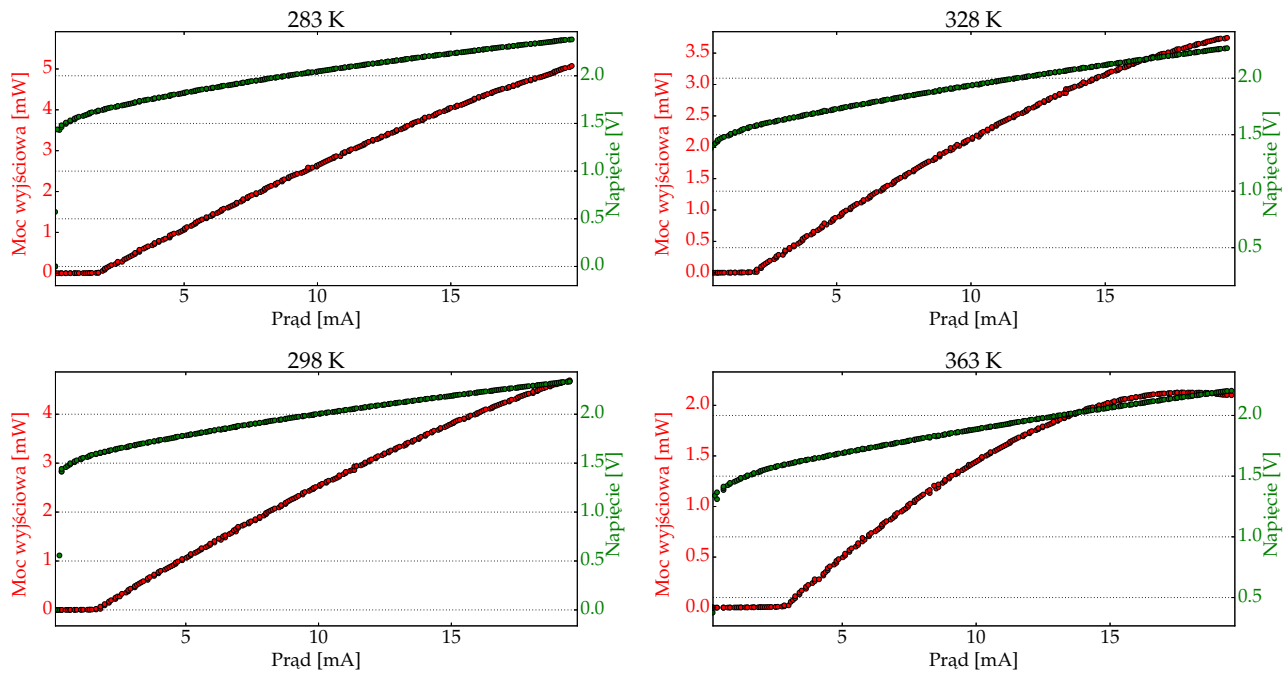
Rysunek 5.5. Wykres sprawności w funkcji mocy wejściowej dla lasera krawędziowego 635 nm.



Rysunek 5.6. Wykres napięcia na laserze oraz mocy w funkcji prądu dla lasera krawędziowego 635 nm.

Tabela 5.2. Wyznaczone wartości prądu progowego  $I_{th}$  w różnych temperaturach  $T$  dla lasera VCSEL 850 nm.

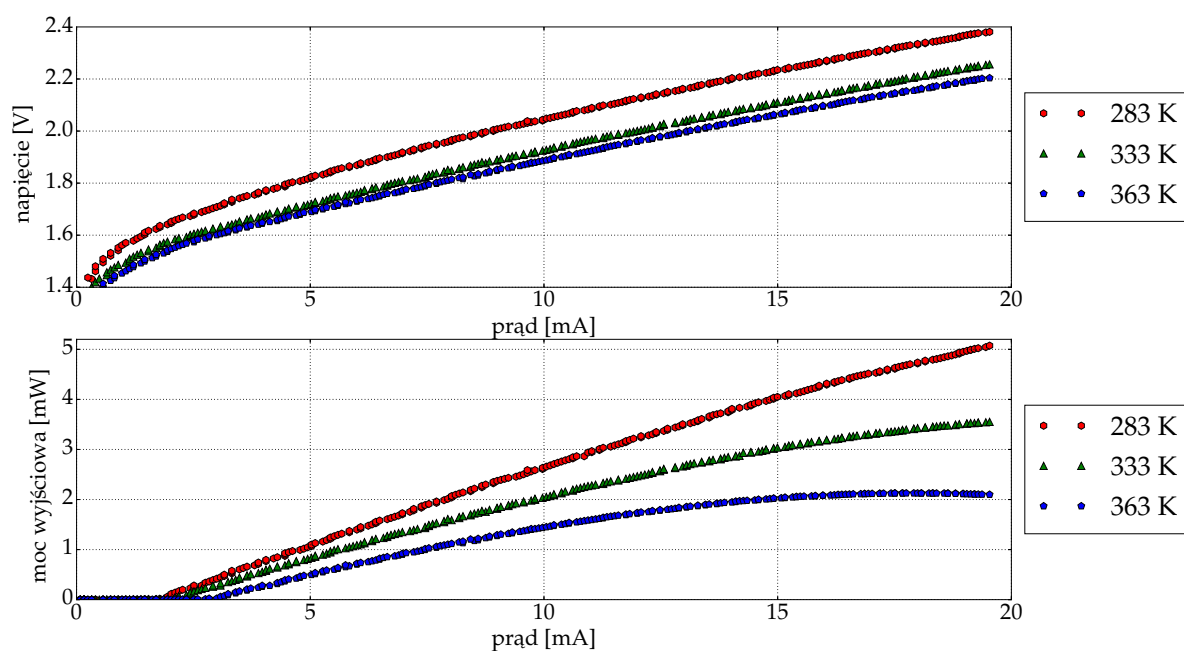
$T$ [K]	$I_{th}$ [mA]	$T$ [K]	$I_{th}$ [mA]	$T$ [K]	$I_{th}$ [mA]
283	$1.70 \pm 0.03$	288	$1.67 \pm 0.03$	293	$1.60 \pm 0.03$
298	$1.55 \pm 0.04$	303	$1.59 \pm 0.03$	308	$1.63 \pm 0.03$
313	$1.65 \pm 0.03$	318	$1.68 \pm 0.04$	323	$1.73 \pm 0.04$
328	$1.83 \pm 0.04$	333	$1.89 \pm 0.04$	338	$2.01 \pm 0.04$
343	$2.14 \pm 0.04$	348	$2.24 \pm 0.05$	353	$2.38 \pm 0.05$
358	$2.57 \pm 0.05$	363	$2.74 \pm 0.07$		



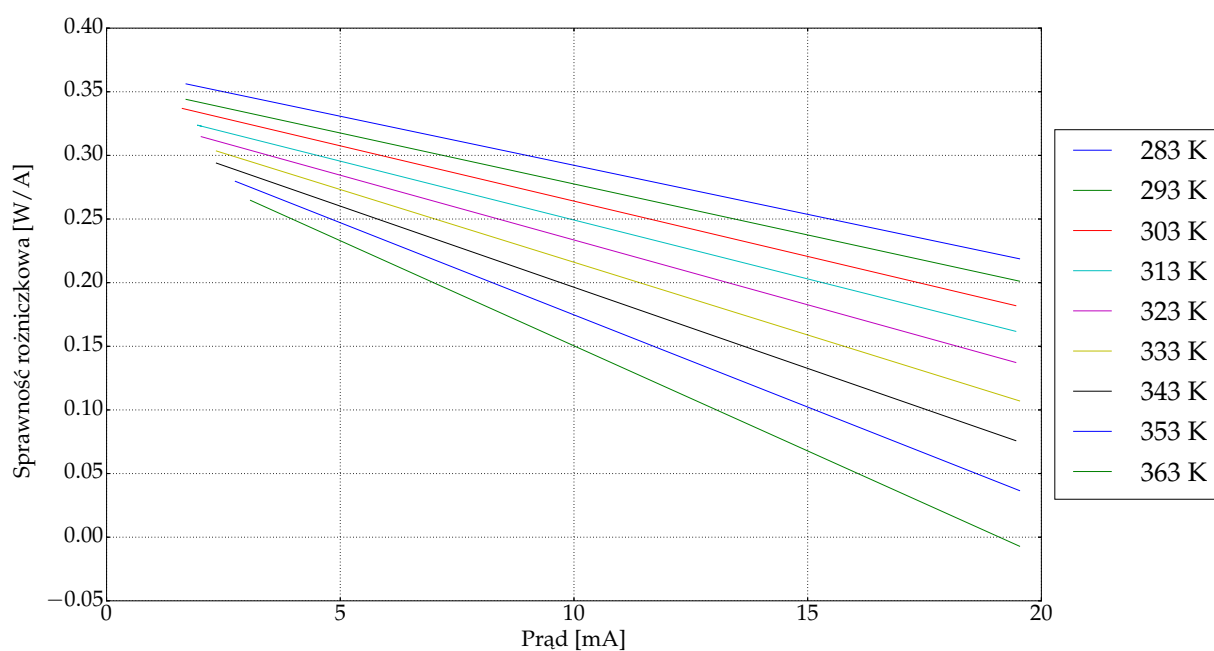
Rysunek 5.7. Sprawność VCSEL 850.

### 5.3. Laser VCSEL 850 nm

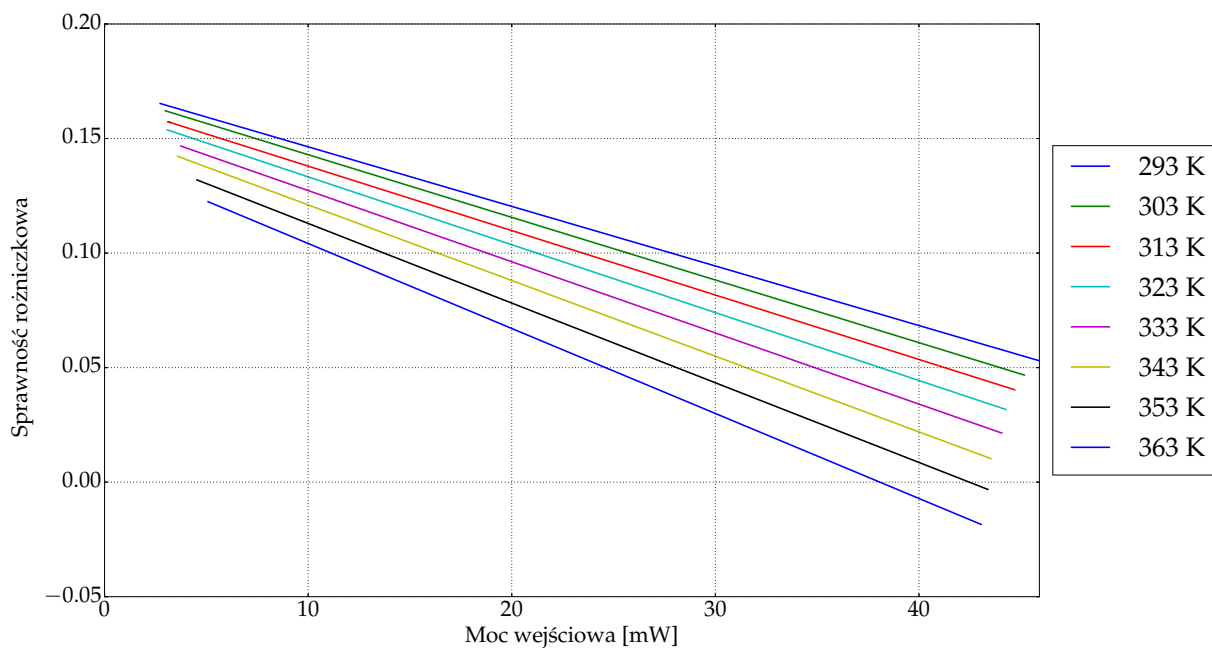
Rysunki 5.7 – 5.13 przedstawiają wykresy dla lasera VCSEL 850 nm.



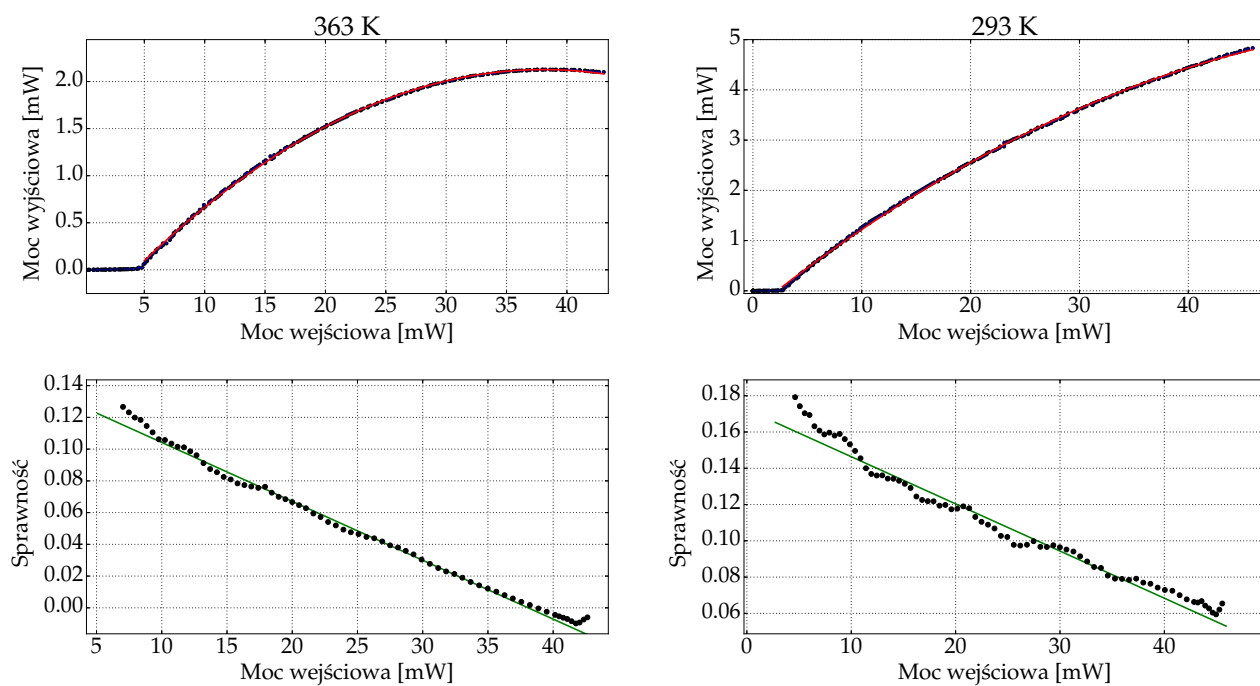
Rysunek 5.8. Wykres napięcia na laserze i mocy wyjściowej w funkcji prądu od temperatury chłodnicy dla lasera VCSEL 850 nm.



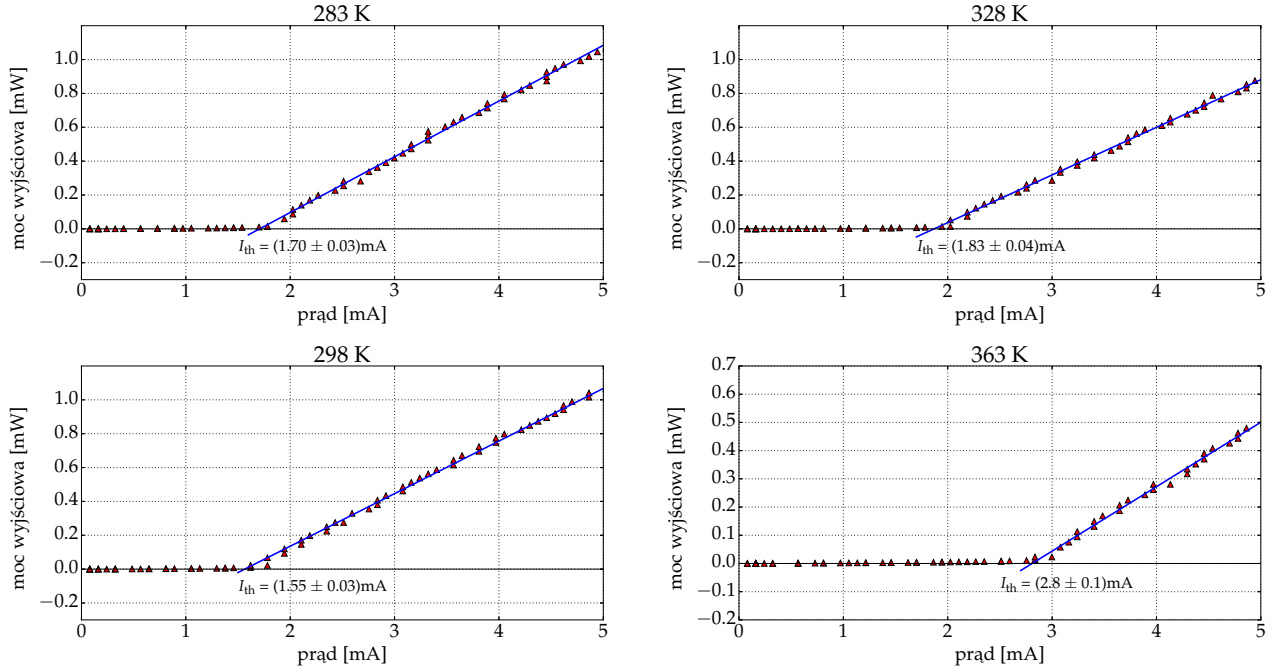
Rysunek 5.9. Sprawność VCSEL 850 w funkcji prądu.



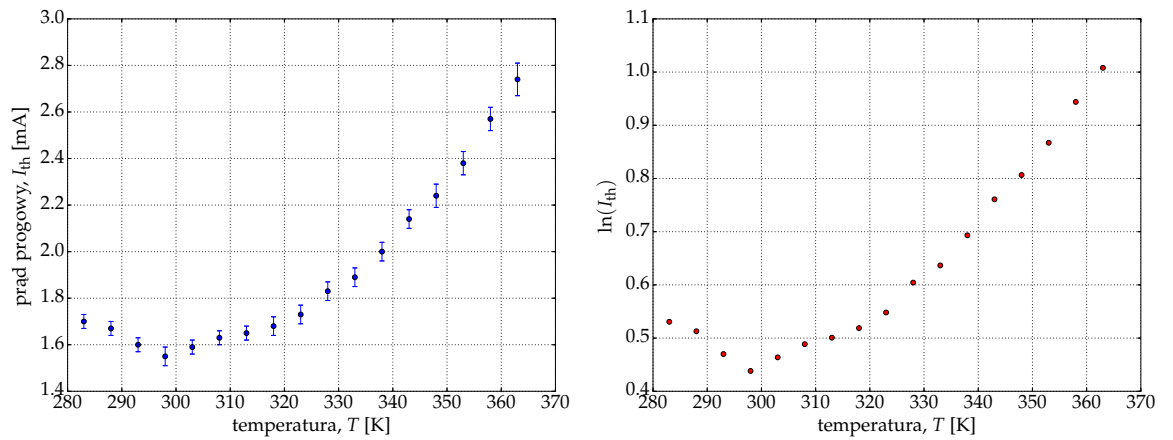
Rysunek 5.10. Sprawność VCSEL 850 w funkcji mocy wejściowej.



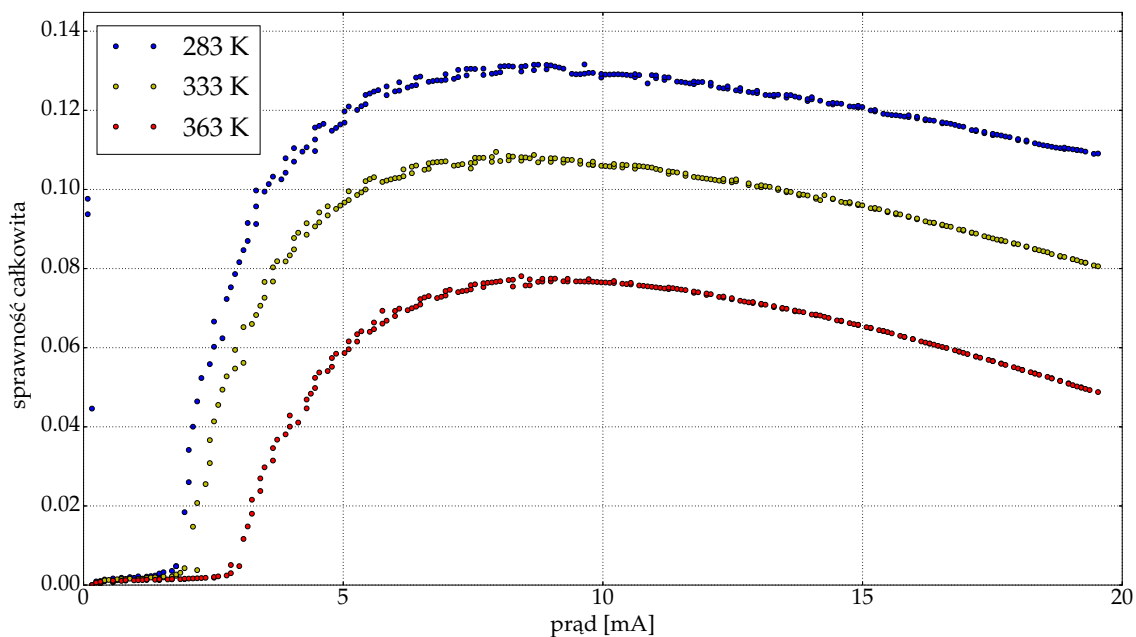
Rysunek 5.11. Sprawność VCSEL 850 dla temperatury 293 K i 363 K.



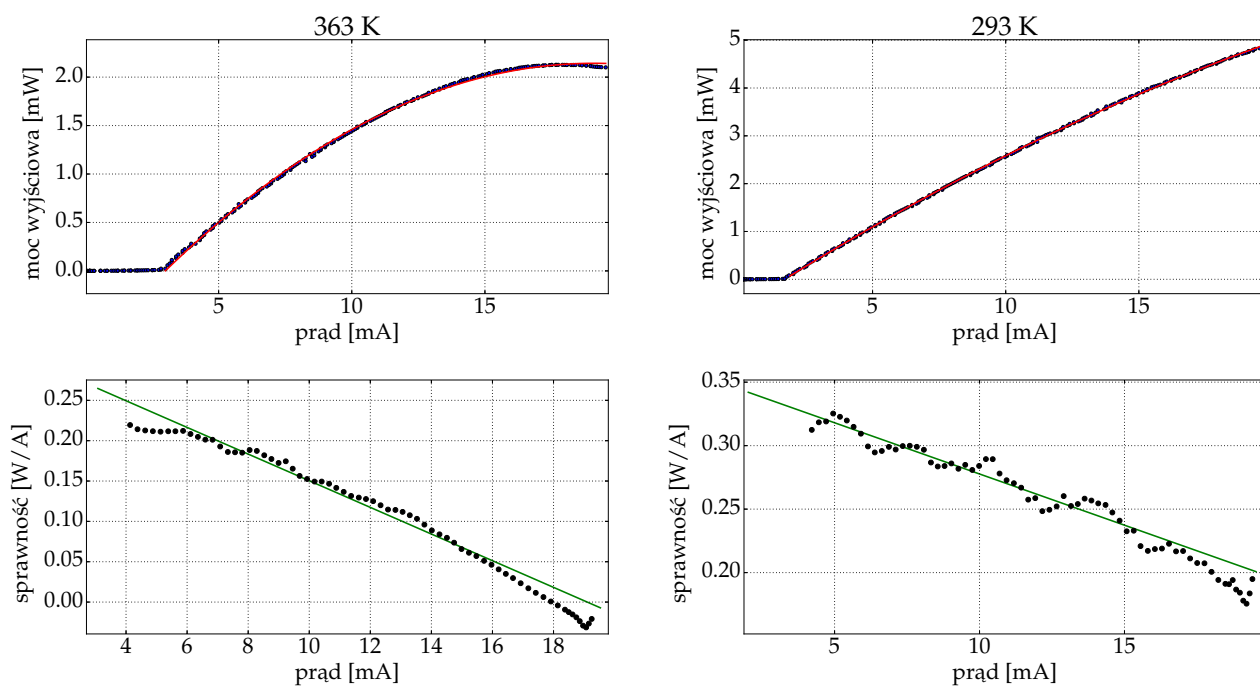
Rysunek 5.12. Wykres prądu progowego od temperatury z wyznaczonymi progami prądu dla lasera VCSEL 850 nm.



Rysunek 5.13. Wykres prądu progowego od temperatury dla lasera VCSEL 850 nm.



Rysunek 5.14. Sprawność całkowita dla lasera VCSEL 850 nm w różnych temperaturach.



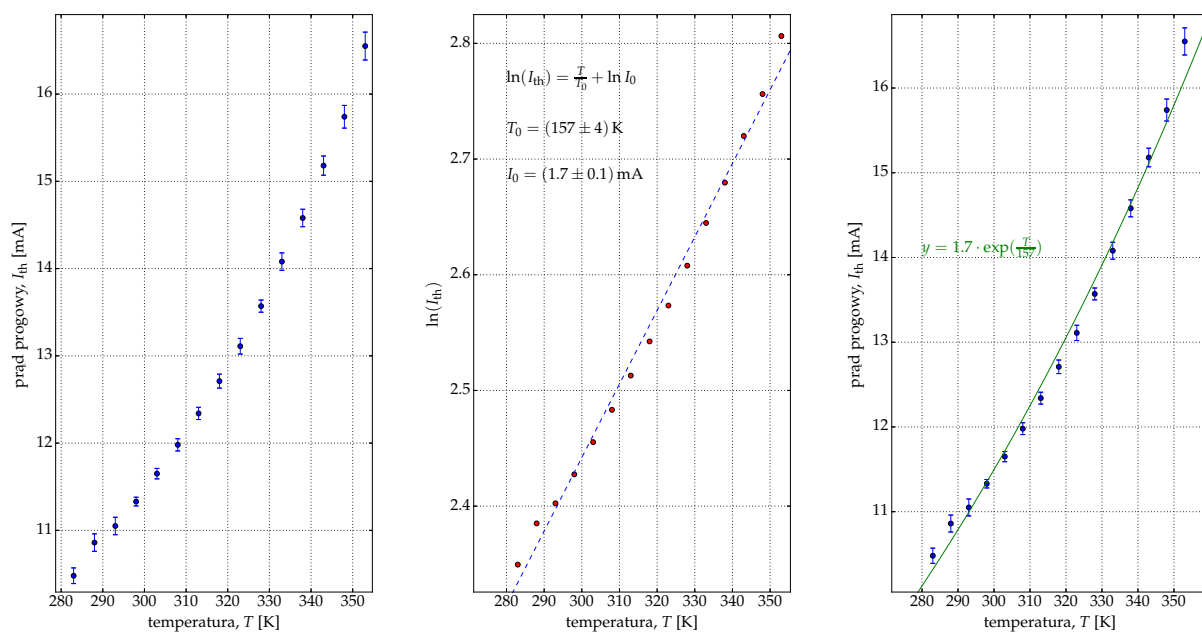
Rysunek 5.15. Sprawność różniczkowa dla lasera VCSEL 850 nm w różnych temperaturach.

#### 5.4. Omówienie wyników

Dzięki analizie sporządzonych wykresów można dojść do następujących konkluzji:

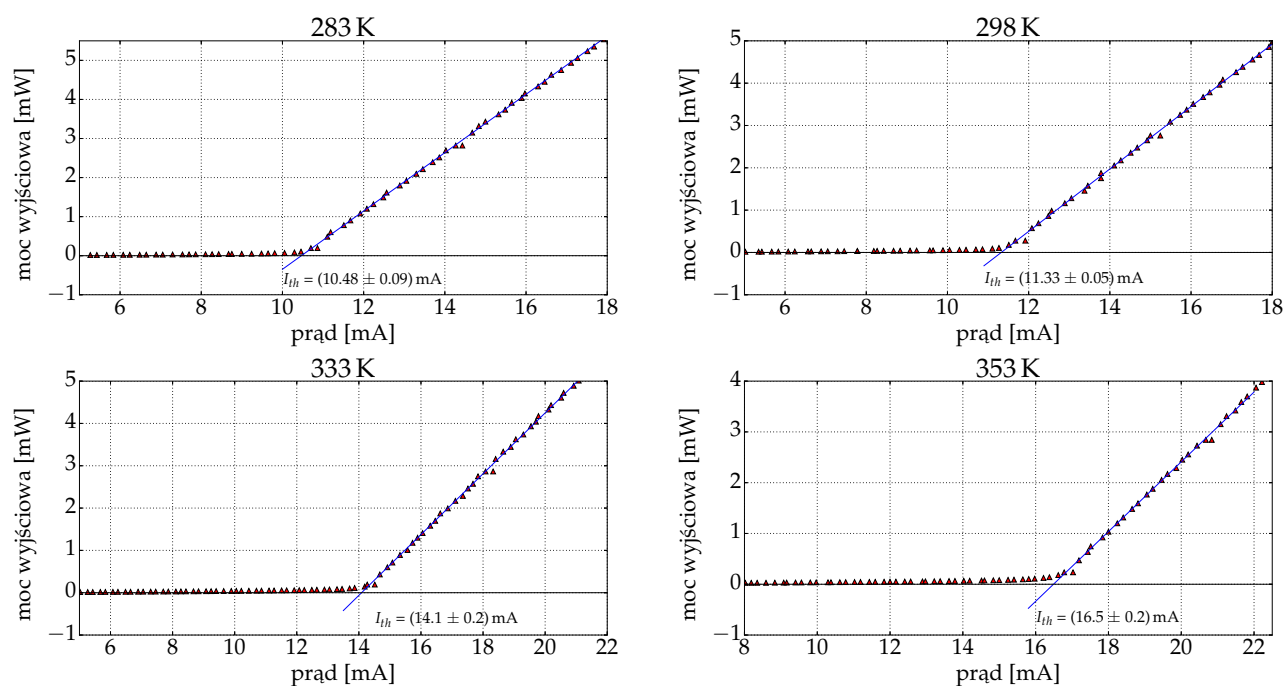
- Analizując wykres napięcia na laserze od prądu wejściowego przedstawiony na rysunku 5.7 oraz 5.8 można zauważyć, że wraz ze wzrostem temperatury na chłodnicy maleje opór lasera. Także, wraz z wyższą temperaturą chłodnicy maleje moc wyjściowa lasera.
- Wykres na rysunku 5.9 przedstawia sprawność różniczkowa lasera w funkcji prądu wejściowego od temperatury na chłodnicy, jak wynika z wykresu im wyższa temperatura tym sprawność lasera mniejsza.
- Wykres na rysunku 5.10 przedstawia sprawność różniczkowa lasera w funkcji mocy wejściowej od temperatury na chłodnicy, jak wynika z wykresu im wyższa temperatura tym sprawność lasera mniejsza.
- Wykres na rysunku 5.11 przedstawia w górnej części zależności mocy wyjściowej od mocy wejściowej.
- Wykres na rysunku 5.13 pokazuje zależności prądu progowego od temperatury. Jak widzimy przy temperaturach (280-300) K wraz ze wzrostem temperatury maleje wartość prądu progowego, natomiast dla temperatur  $> 300$  K im wyższa temperatura to zwiększa się wartość prądu progowego.
- Wykres na rysunku 5.14 przedstawia sprawność całkowitą lasera dla trzech temperatur: 283 K, 333 K, 363 K. Analizując ten wykres dochodzę do wniosku, że im wyższa temperatura tym sprawność mniejsza.
- Wykres na rysunku 5.15 przedstawia sprawności różniczkowe dla dwóch temperatur. W górnej części przedstawiona jest charakterystyka wyjściowa z dopasowaną funkcją w postaci wielomianu stopnia drugiego. Pochodna tej funkcji jest sprawnością różniczkową. Na dolnym wykresie przedstawiona jest sprawność w postaci prostej będącej pochodną dopasowanej funkcji. Natomiast czarne kropki przedstawiają sprawność powstałą w wyniku obliczenia pochylenia 10 punktów przefiltrowanych co 3 punkty.





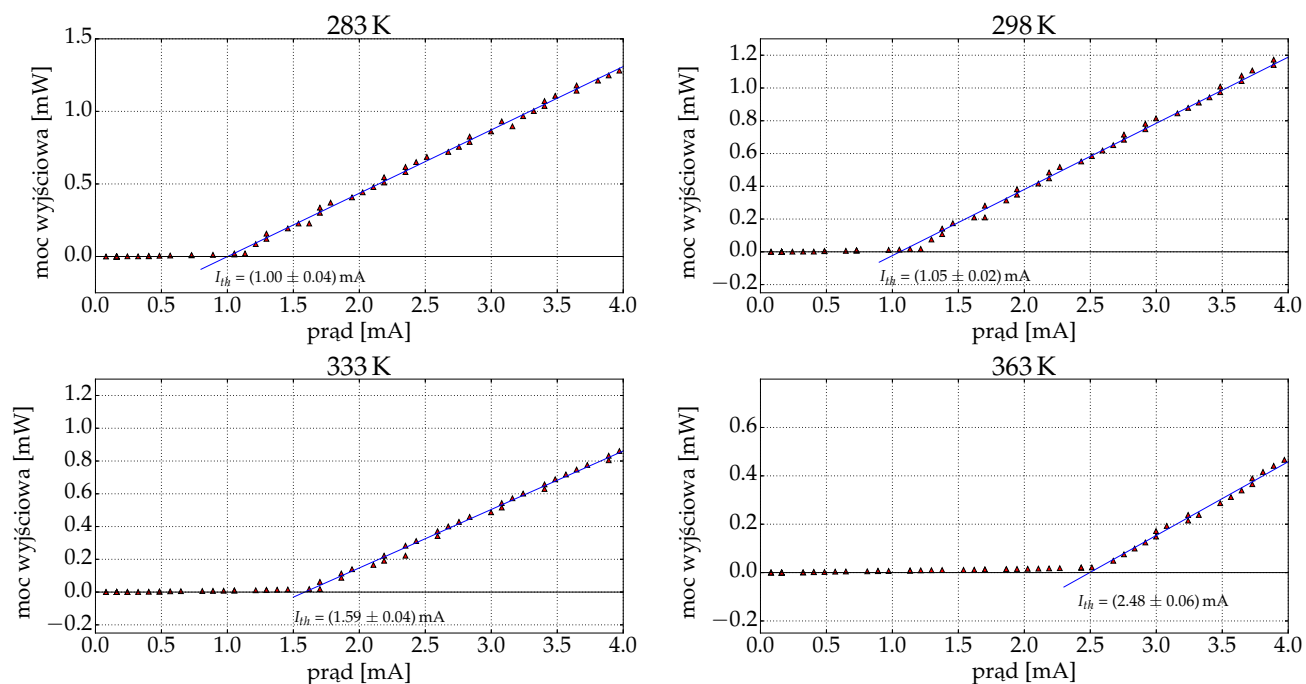
Rysunek 5.16. Wykres napięcia i mocy od prądu dla lasera krawędziowego 850 nm.

## 5.5. Laser krawędziowy 850 nm



Rysunek 5.17. Wykres ilustrujący wyznaczanie prądu progowego dla lasera krawędziowego 850 nm.

## 5.6. Laser VCSEL 980 nm

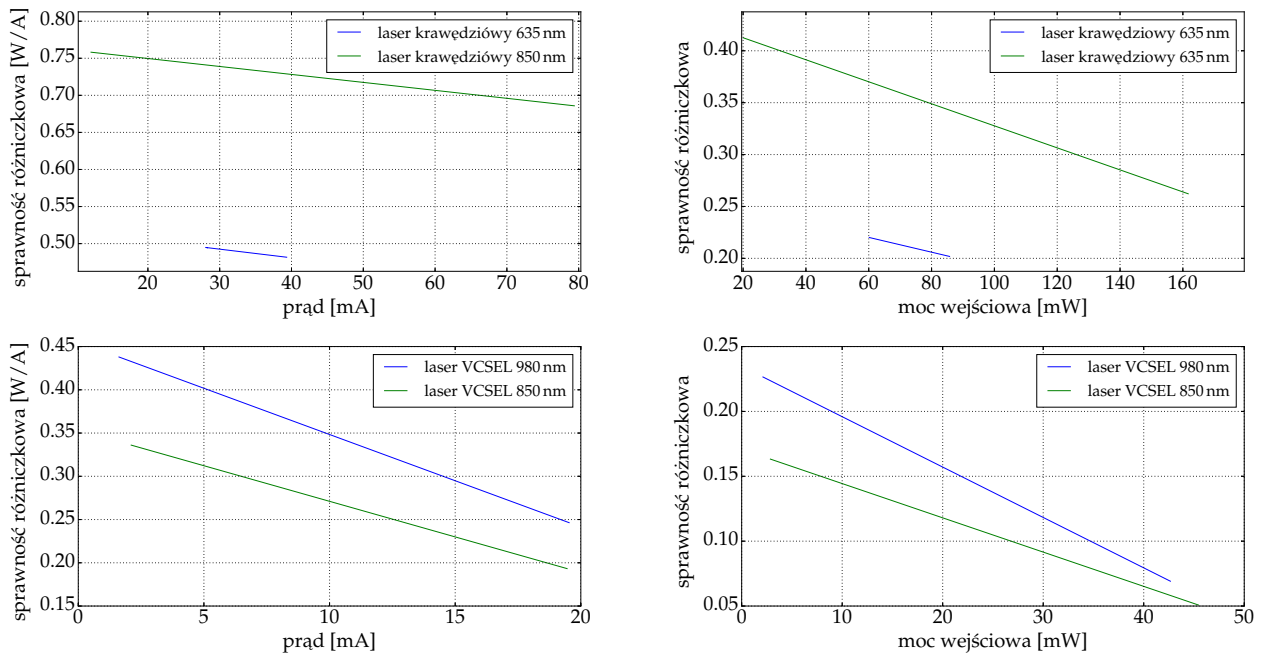


Rysunek 5.18. Wykres ilustrujący wyznaczanie prądu progów dla lasera VCSEL 980 nm.

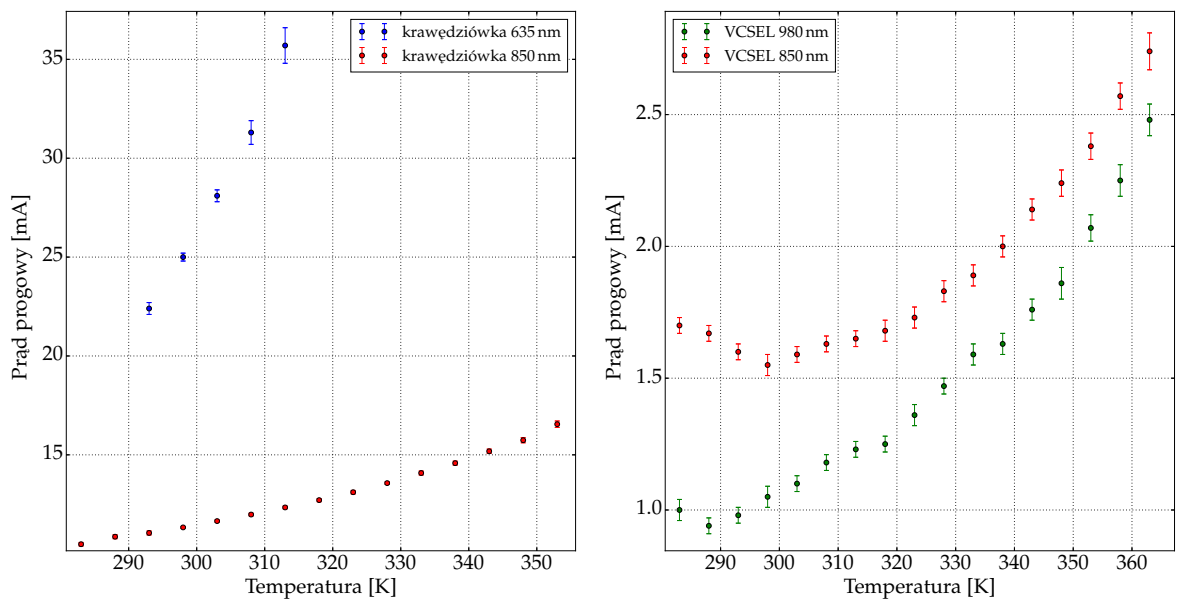
## 5.7. Porównanie laserów

Analizując pomiary dla 4 laserów które przeprowadziłem, można wyciągnąć następujące wnioski:

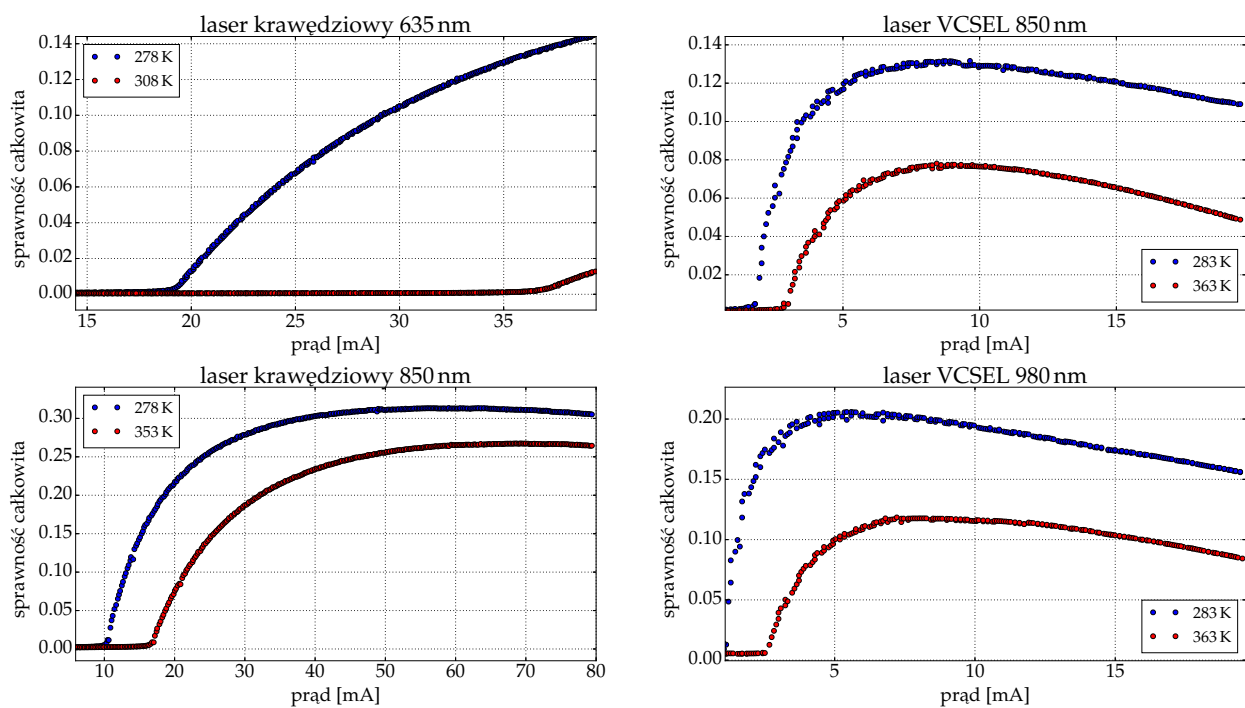
- Sprawność różniczkowa laserów krawędziowych w funkcji zarówno prądu i mocy wejściowej jest większa.
- Prąd progowy dla laserów krawędziowych jest większy od prądu progowego dla laserów VCSEL.



Rysunek 5.19. Wykres sprawności różniczkowej w funkcji prądu oraz mocy wejściowej.



Rysunek 5.20. Wykres prądu progowego od temperatury.



Rysunek 5.21. Wykres sprawności całkowitej w funkcji prądu.

# Bibliografia

- [1] Thorlabs Manual: LDC4000 Series Operation Manual, 2016
- [2] Thorlabs Manual: Series 4000 SCPI Programmer's Reference Manual, 2015
- [3] Thorlabs Manual: Operation Manual Thorlabs Instrumentation Optical Power and Energy Meter PM100 USB, 2011
- [4] E. Bressert: Scipy and NymPy, O'Reilly 2013
- [5] A. Devert: matplotlib Plotting Cookbook Packt Publishing Ltd. 2014