

Instrukcja używania dystrybutorów cieczy Poseidon

Kacper Kamiński, Krzysztof Podoba, Mateusz Szczęśniak, EiT

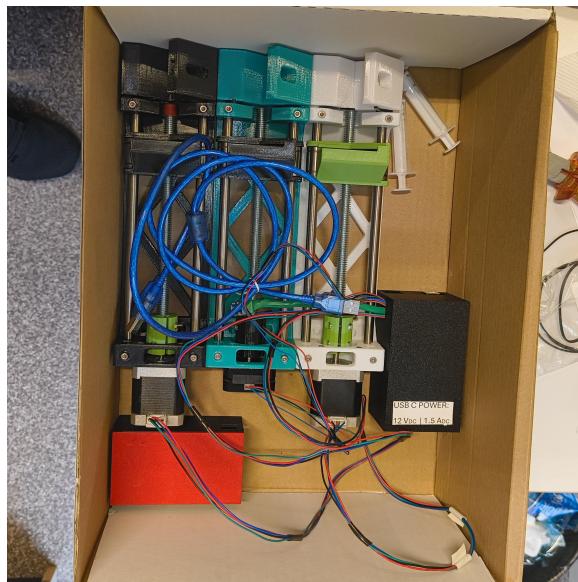
29.05.2025

Spis treści

1 Wstęp	2
2 Opis zestawu	2
2.1 Pompki	2
2.2 Kontroler	3
2.3 Przewód zasilający oraz komunikacyjny	3
2.4 Zatrzaski	4
3 Podłączenie pompek oraz uruchomienie programu	4
3.1 Zależności potrzebne do uruchomienia systemu	4
3.2 Pobranie repozytorium oraz przejście do odpowiedniego katalogu	4
4 Używanie pompek	4
4.1 Podłączenie systemu pompek do komputera	5
4.2 Kонтролювання помпек	7
5 Kalibracja oraz zmiany parametrów	7
6 Pomiary dokładności, zawartość do zmiany lub innego punktu	8
7 Możliwe błędy	8
8 Zmiana elementów oraz modeli	8
9 Bibliografia - dostęp z dnia 06/08/2025	9

1 Wstęp

Dystrybutory cieczy zostały zbudowane w oparciu o projekt *open-source* autorstwa grupy Pachter Lab, udostępnionego na platformie *GitHub* ([link](#)). Część modeli zmodyfikowano w celu dopasowania ich do posiadanych już elementów. Można je znaleźć w repozytorium praktyk ([link](#)).

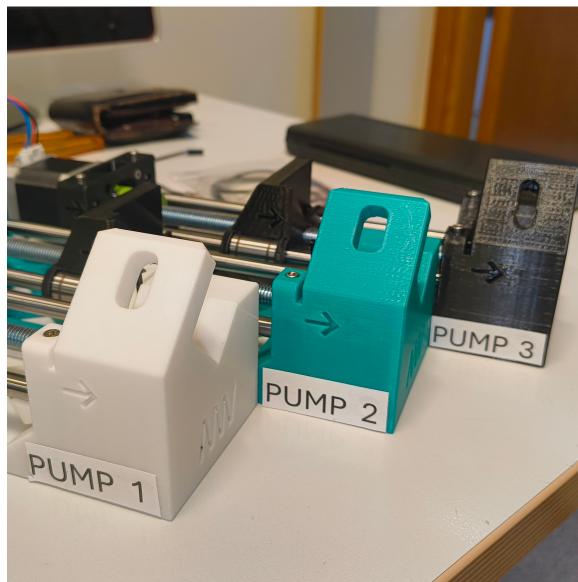


Rysunek 1: Zestaw pompek

2 Opis zestawu

2.1 Pompki

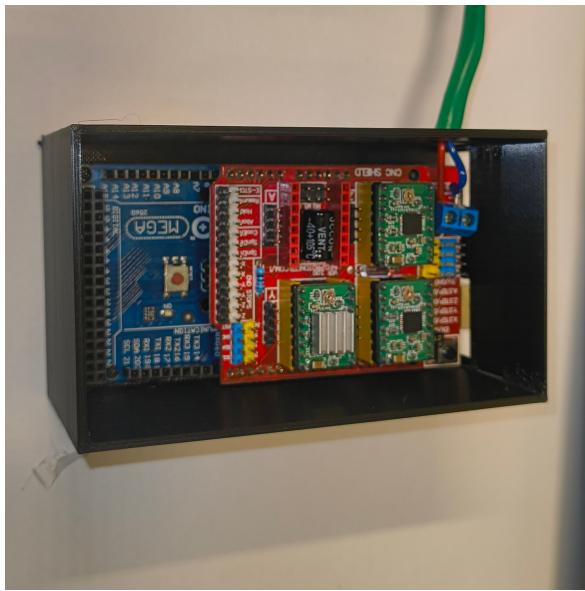
Zestaw jest zbudowany z trzech pompek podpisanych korespondującymi numerami - "Pump 1", "Pump 2", "Pump 3". Każda z nich posiada zatrzaski, które służą do utrzymania strzykawki w miejscu. Przy odpowiednim wygięciu powinny utrzymać stałą pozycję strzykawki z cieczą.



Rysunek 2: Pompki z podpisanymi numerami: biała: "PUMP 1", niebieska: "PUMP 2", czarna: "PUMP 3"

2.2 Kontroler

Mikrokontroler znajdujący się w pudełku zapewnia sterowanie dystrybutorami - jest to *Arduino Mega* ([link](#)) połączony z *CNC Shield* oraz sterownikami silników krokowych (*Nema17*) Pololu A4988 ([link](#)).



Rysunek 3: Mikrokontroler ze sterownikami silników krokowych z odpiętymi przewodami



Rysunek 4: Zamknięty pojemnik z mikrokontrolerem

2.3 Przewód zasilający oraz komunikacyjny

W zestawie nie ma ładowarki. Wymagane jest zapewnienie takiej, która zapewnia wystarczającą moc by sterować pompkami. Przykładowo może być to $45W - 12V_{DC}$ i $1.5A_{DC}$. Ładowarka z funkcją Quick-Charge 3.0 i wzwyż (np. 45W Baseus) spełniająca te założenia.

Przewód komunikacyjny to USB B (komputerowe) - USB B (drukarkowe), jest zapewniony w zestawie.



Rysunek 5: Przykład przewodu zasilającego wraz z ładowarką 45W



Rysunek 6: Przewód komunikacyjny USB

2.4 Zatrzaski

W pudełku o czerwonym wieczku znajdują się dodatkowe zatrzaski, które można użyć w przypadku zniszczenia poprzednio zamontowanych.

3 Podłączenie pompek oraz uruchomienie programu

Aby uruchomić program sterujący należy posiadać zainstalowany język programowania *Python* (poprawne zachowanie testowane na wersji 3.13.3). Mikrokontroler Arduino Mega wymaga środowiska Arduino IDE oraz kompilatora języka *C*, które pozwala na wgranie kodu potrzebnego do obsługi sygnałów sterujących silnikami. Trzeba go zainstalować tylko, jeżeli chce się ingerować w kod sterujący silnikami, w przeciwnym razie nie trzeba go instalować - kontroler został już zaprogramowany.

3.1 Zależności potrzebne do uruchomienia systemu

Należy posiadać następujące dependencje - aby zainstalować je można wykonać następujące polecenia:

```
1 pip3 install --upgrade -pip
2 pip3 install pyserial
3 pip3 install opencv-python
4
5 # jeżeli środowisko linux to również należy wykonac to:
6 sudo apt-get install libatlas-base-dev
7 sudo apt-get install libjasper-dev
8 sudo apt-get install libqtgui4
9 sudo apt-get install libqt4-qt3support
10 sudo apt-get install libqt4-test
```

3.2 Pobranie repozytorium oraz przejście do odpowiedniego katalogu

W celu pobrania repozytorium praktyk należy wykonać polecenie:

```
11 git clone https://github.com/mszczesniak02/eit_internship.git
```

następnie przejść do folderu `./eit_internship/Poseidon_pump/code/SOFTWARE/`.

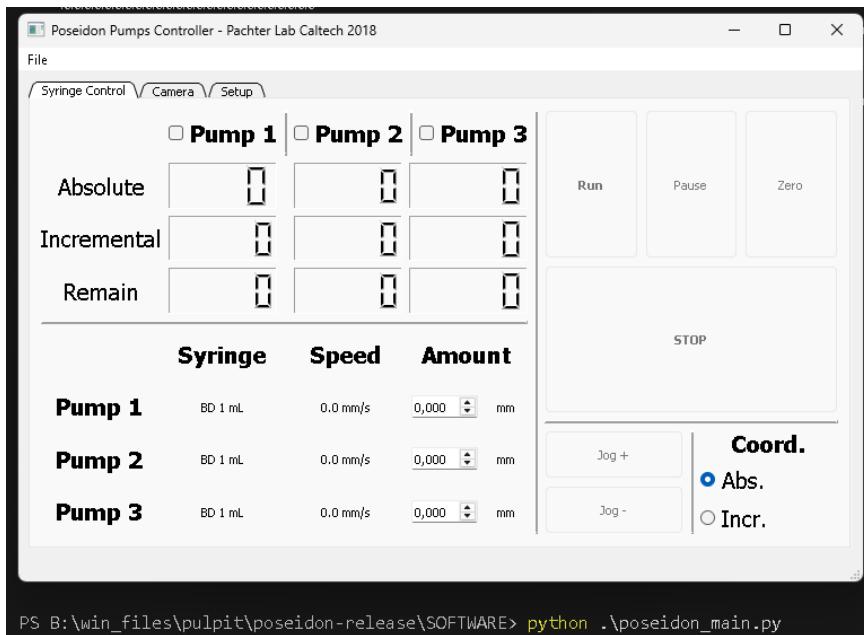
4 Używanie pompek

Aby korzystać z pompek należy zapewnić strzykawki, które mieszą się w prowadnicy. Zatrzaski powinny utrzymać strzykawki w miejscu. Aby wydobyć oczekiwana ilość płynu należy zmierzyć odległość na strzykawce, następnie ustawić tą ilość w programie.

Uwaga nr 1! Należy zapewnić 0.1 mm nadmiarowej (buforowej) odległości, by silniki pokonały siły trzymające strzykawkę w miejscu.

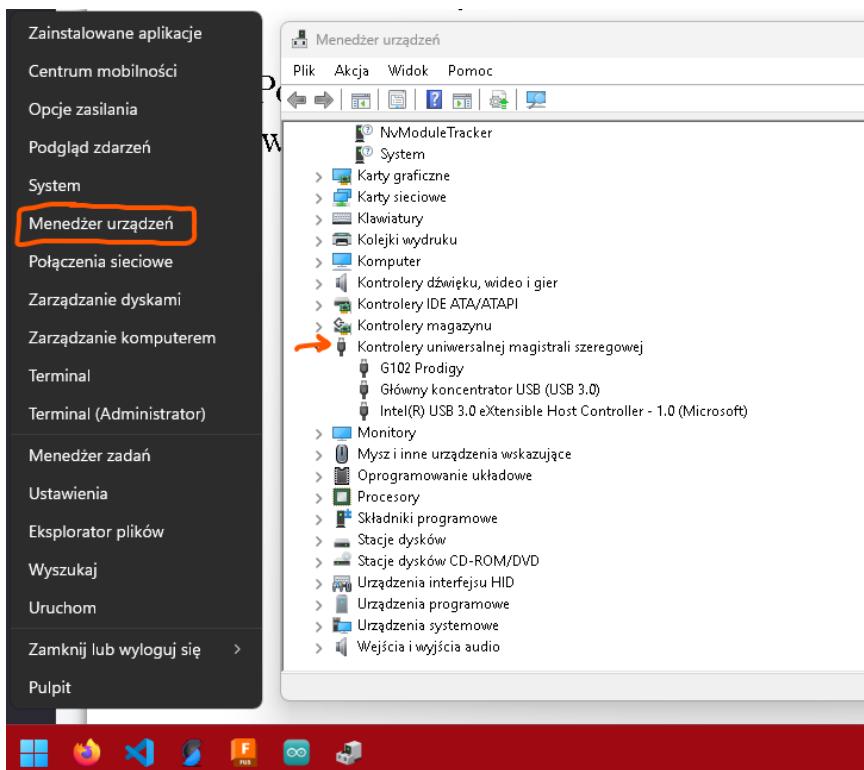
Uwaga nr 2! Należy pamiętać, że przesuwak nie dochodzi do końca - nie pozwalając na pełne opróżnienie strzykawki, stąd zalecanie jest zapewnienie większej ilości płynu.

4.1 Podłączenie systemu pompek do komputera



Rysunek 7: Uruchomienie systemu z poziomu terminala z widoczną komendą poniżej

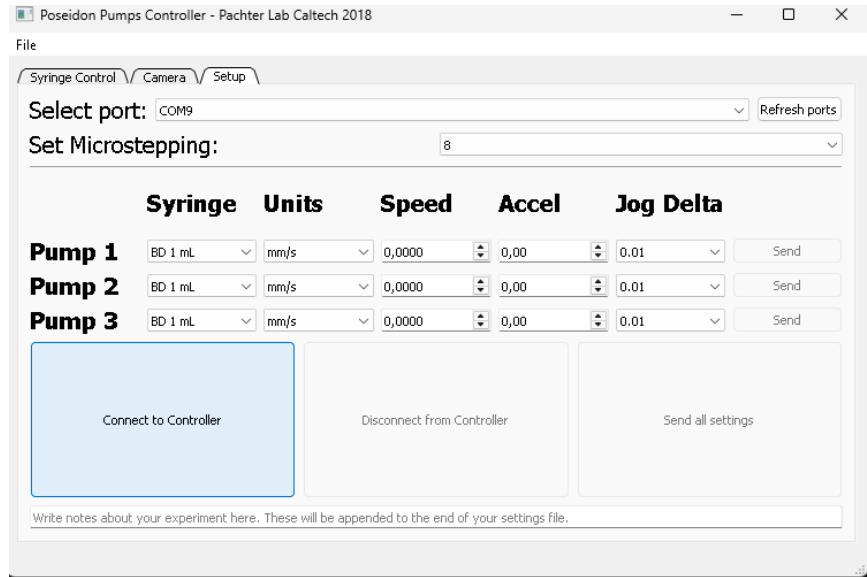
Po włączeniu oprogramowania oraz podłączenia kontrolera i zasilania, należy się z nim połączyć - po pierwsze sprawdzić na jaki port szeregowy został podłączony: Na systemie Windows / Linux



Rysunek 8: Prawy przycisk myszy na ikonkę Windows → Menadżer urządzeń → Magistrala szeregowa (Powinno się pojawić Arduino MEGA (COMx))

W celu sprawdzenia numeru portu w systemie Linux:

```
12 $ dmesg | grep tty
```

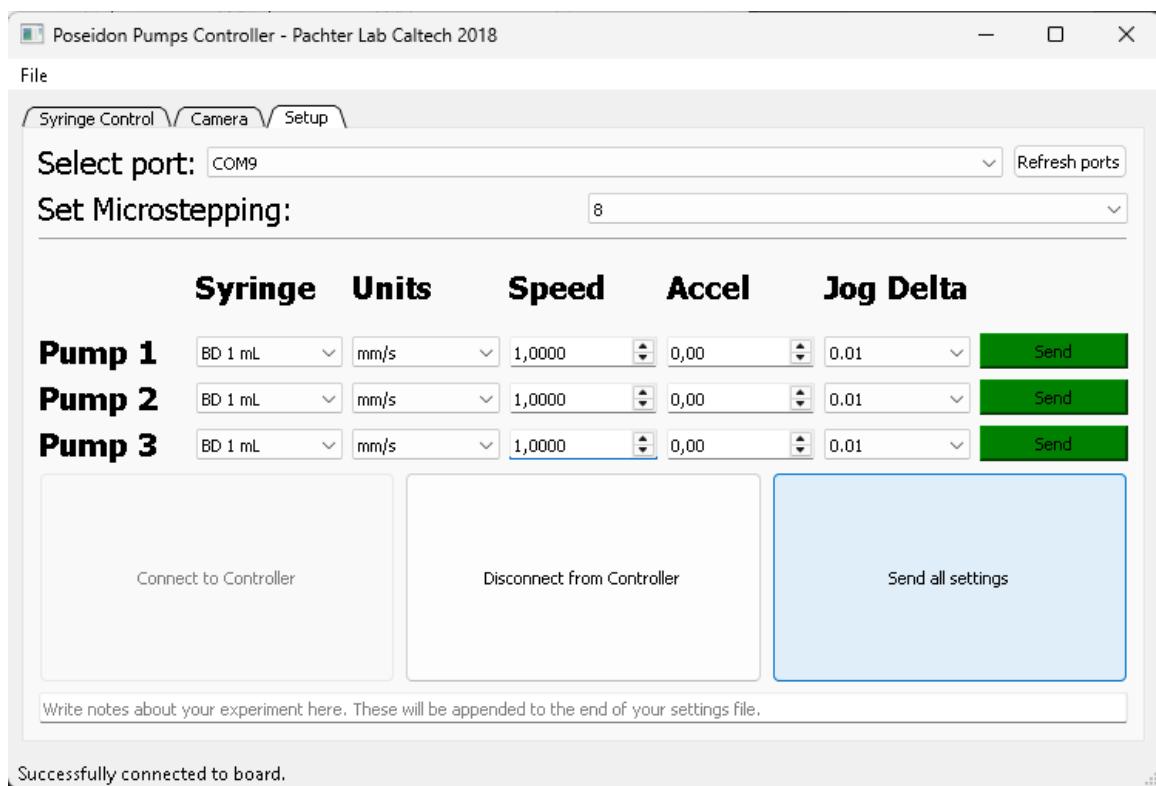


Rysunek 9: Połączenie się z kontrolerem

W przypadku braku wyświetlenia portu na liście należy odświeżyć ją przyciskiem "Refresh ports".

Aby dystrybutory działały poprawnie, mikrokrok należy ustawić na wartość **8**, jeżeli ten krok nie zostanie wykonany, kroki będą nieprecyzyjne.

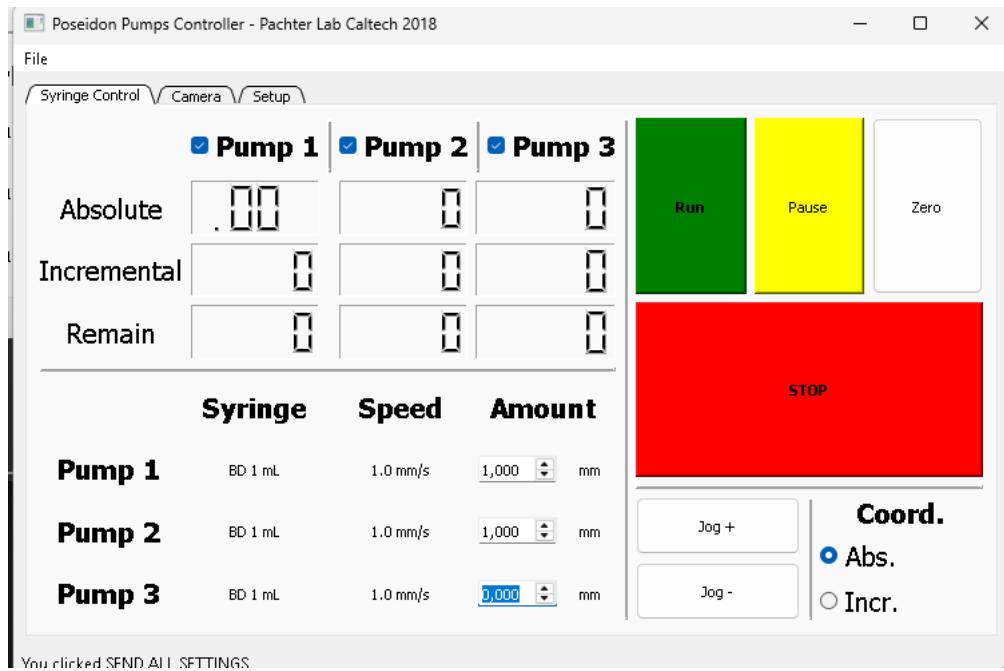
Następnie należy nacisnąć "Connect to Controller", po odczekaniu chwili można przejść do ustawiania prędkości (*Speed*) wpisując pożądane wartości (wpisanie za dużej wartości nie spowoduje zniszczenia sprzętu - ograniczenie prądowe, z tego powodu również nie zadziała funkcja zmiennego przyspieszenia). Zmiany należy zatwierdzić klikając "Send all settings".



Rysunek 10: Ustawienie prędkości oraz wysłanie ustawień

4.2 Kontrolowanie pompek

Następnie ostatecznie należy przejść do zakładki *"Syringe Control"*.



Rysunek 11: Ustawienia zakładki *"Syringe Control"*

Ostatnim krokiem jest zaznaczenie pompek, które chcemy aby się ruszały, wpisać *"Amount"* o jaki chcemy, aby pompki się przemieściły, oraz uruchomić je klikając zielony przycisk *"Run"*.

W przypadku chęci zastopowania w ciągu działania pompek - wcisnąć przycisk *"STOP"*.

5 Kalibracja oraz zmiany parametrów

Możliwa jest dostrojenie układu poprzez zmianę współczynnika kalibrującego *"correction_parameter"* znajdującego się w pliku *"poseidon_main.py"*:

```
1218 def convert_displacement(self, displacement, units, syringe_area, microsteps):
1219     length = units.split("/") [0]
1220     time = units.split("/") [1]
1221     inp_displacement = displacement
1222     # convert length first
1223     if length == "mm":
1224         displacement = self.mm2steps(displacement, microsteps)
1225     elif length == "mL":
1226         displacement = self.mL2steps(displacement, syringe_area, microsteps)
1227     elif length == " L ":
1228         displacement = self.uL2steps(displacement, syringe_area, microsteps)
1229     print('-----')
1230     print("INPUT DISPLACEMENT: " + str(inp_displacement) + ' ' + length)
1231     print("OUTPUT DISPLACEMENT: " + str(displacement) + ' steps')
1232     print('\n#####\n')
1233     # zmiana parametru kalibrujacego
1234     correction_parameter = 0.9311019628300378
1235     displacement = displacement / correction_parameter
1236
1237     return displacement
```

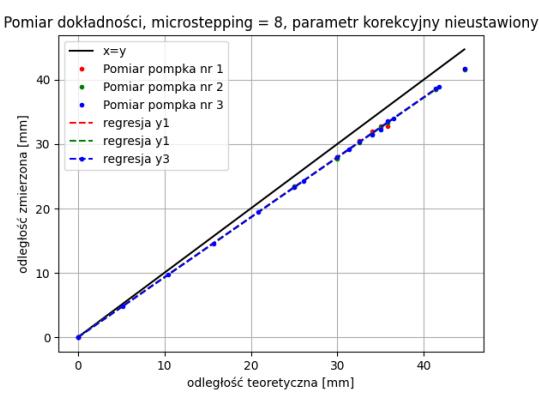
Fragment pliku *"poseidon_main.py"*.

Parametr powinien zostać dobrany polegając na zmierzeniu testowych odległości precyzyjnym narzędziem np. suwmiarką. Taką operację wykonano oraz opisano w punkcie **6 Pomiary dokładności, zawartość do zmiany lub innego punktu**. Należy pamiętać o tym, że elementy użyte w zbudowanych pompkach nie są idealne - krzywy pręt gwintowany, elementy wydrukowane na Voron 2, różne silniki.

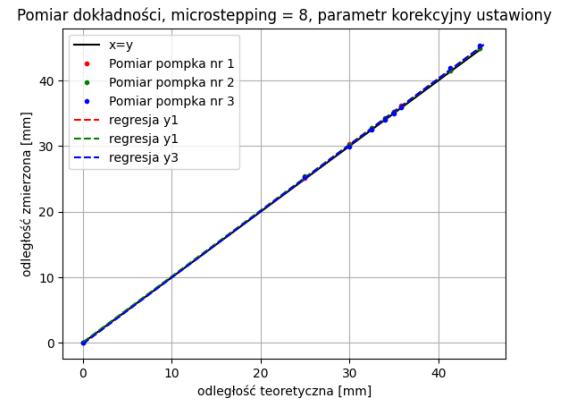
Badana dokładność / zbieżność zależy od dokładności urządzenia pomiarowego, od jego ułożenia na prowadnicach (każdy pomiar może odbiegać od początkowego pomiaru). Dokładność zmierzona przy pomocy suwmiarki (mierząc przez nas) pozwala uzyskać dokładność do $200 \mu m$ (na tyle oceniamy dokładność naszych pomiarów).

6 Pomiary dokładności, zawartość do zmiany lub innego punktu

Pomiarów dokonano na trzech pompkach jednocześnie, w celu równomiernego rozłożenia prądu. Zmierzono odległości suwmiarką, starając się robić to za każdym razem w tym samym miejscu. Krok przejścia silników został ustalony tak, by przestawał zróżnicowane wartości - od $0.1mm$, do $20mm$. Wykonano dopasowanie liniowe oraz obliczono nachylenie, względem którego dokonano korekty. Wyniki przedstawiono poniżej.



Rysunek 12: Pomiary bez współczynnika kalibrującego



Rysunek 13: Pomiary z współczynnikiem kalibrującym

7 Możliwe błędy

W przypadku losowego wyłączenia się silników bądź ich samoistnego działania, należy sprawdzić okablowanie w pudełku kontrolera - jeżeli przewody tworzą zwarcie, należy je zaizolować. Może nastąpić sytuacja, w której przewody działającej pompki znajdą się na "górze" mikroprocesora układu sterującego bez radiatorka - wtedy zmienia się działanie sterownika, co może powodować niepoprawne działanie pompek, bądź uszkodzenie (spalenie) okablowania spowodowanego wysokimi temperaturami układu - aby zapobiec temu, należy wstawić przewody do prowadnic na wieczku pudełka kontrolera tak, by nie dotykały "góry" mikroprocesora.

W przypadku chęci uruchomienia tylko dwóch bądź jednej, można zwiększyć ich prędkość, pamiętając jednak, że ustawione ograniczenie prądowe wynosi $300 mA$.

8 Zmiana elementów oraz modeli

Ustawianie prędkości działania pompek przy ograniczeniu prądowym nie pozwoli na stosowanie wysokich wartości tempa przesuwania prowadnicy - jeżeli wymagana jest większa prędkość, należy zmienić ograniczenie prądowe.

W przypadku chęci zmiany ograniczenia prądowego należy postępować zgodnie z poradnikiem ([link](#)). Obecny układ nie posiada radiatorów, które pozwoląby na dostarczenie większego prądu bez uszkodzenia sterowników silników. Przy zmianie ograniczenia prądowego należy zapewnić lepsze odprowadzenie cieplne.

Uwaga! Przy wypinaniu konektorów sterowników należy wyłączyć zasilanie płytki (z ładowarki). Silniki krokowe stanowią "load" dla sterownika, bez nich układ nie dostarcza mocy do zewnętrznego elementu, przegrzewając się błyskawicznie, co powodować jego spalenie.

Części używanych zestawów znajdują się w repozytorium w folderze ./eit_internship/Poseidon_pump/models/. Modele są przygotowane w programie **Autodesk Fusion**. Opisy modeli są przedstawione w pliku tekstowym "opis_modeli.txt" w tym samym folderze.

9 Bibliografia - dostęp z dnia 06/08/2025

- oryginalny projekt pumpki <https://github.com/pachterlab/poseidon>
- repozytorium praktykowe https://github.com/mszczesniak02/eit_internship
- Arduino Mega - strona producenta <https://docs.arduino.cc/hardware/mega-2560/>
- zmiana ograniczenia prądowego sterownika silnika https://reprap.org/wiki/StepStick#Adjusting_and_testing_the_current