

Politechnika Wrocławska
Wydział Informatyki i Telekomunikacji
Urządzenia Peryferyjne Laboratoria

Sprawozdanie z zadania nr.1
Akwizycja Sygnałów
13.10.2025

Autorzy:
Daniel Gościński 280878
Łukasz Duda 280916

Prowadzący:
dr inż. Marek Piasecki

Spis treści

1. Cel ćwiczenia	3
2. Zawartość pliku zip	3
3. Przebieg ćwiczenia	3
4. Opis działania programów	5
4.1. Program main.py (część Pythona)	5
4.2. Program measurement (część C#)	6
5. Wnioski	6
6. Podsumowanie	6
7. Skan raportu z zajęć	7

1. Cel ćwiczenia

Celem ćwiczenia było zapoznanie się z mechanizmem akwizycji sygnałów oraz sposobem w jaki można zbierać sygnały w celu wykorzystania danych w kodzie tworzonego programu. Do wykonania były zadania, których wynikiem końcowym był poprawny odczyt danych oraz ich interpretacja przez utworzony program.

2. Zawartość pliku zip

W skompresowanym pliku o rozszerzeniu .zip znajdują się następujące pliki:

- Folder measurment: znajduje się w nim program utworzony w C#, zajmujący się odczytem sygnału (wyjaśnienie w sekcji 4)
- Plik main.py: program utworzony w Pythonie, odczytujący dane zawierające się w pliku **sensor_data.csv**, zamieniając je w wykres w programie okienkowym
- seonsor_data.csv: plik csv utworzony poprzez program measurment na podstawie odczytu z fizycznego generatora sygnału
- sprawozdanie.pdf: dokument zawierający sprawozdanie z zajęć

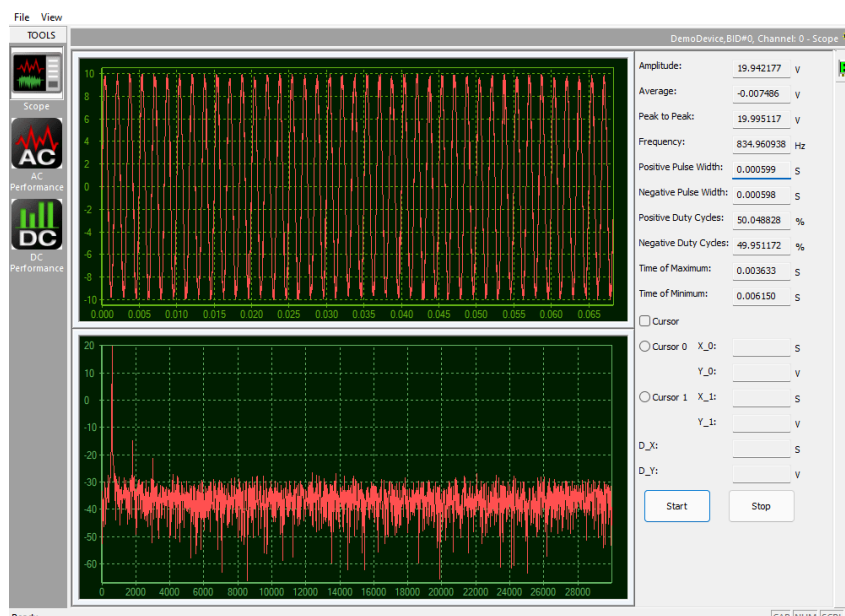
3. Przebieg ćwiczenia

Przed przystąpieniem do zajęć w celach lepszego zrozumienia problemu przygotowano prymitywną wersję finalnego programu w Pythonie, która zapisywała do pliku .csv wyniki wyliczonej na podstawie wybranej funkcji generującej sinusoidę, a następnie odczytywała ten plik i generowała stosowny wykres z możliwością interakcji z nią. Z tym programem, oraz nabytą wiedzą teoretyczną o problemie przystąpiono do zajęć laboratoryjnych.

Po rozpoczęciu zajęć przystąpiono do zapoznania się z fizycznym sprzętem leżącym na stanowisku. Po odpowiednich wyszukiwaniach ustalono, że stanowisko składało się z:

- Generators funkcji ORTEL G-5001
- Bloku rozdzielczego ADAM-3937 firmy ADVANTECH
- Modułu USB-4702 firmy ADVANTECH

W założeniu taka konfiguracja pozwalała na wygenerowanie sygnału sinusoidalnego, kwadratowego lub trójkątnego o wybranej amplitudzie i częstotliwości, a sygnał ten w postaci cyfrowej można było odczytać na komputerze posiadającym odpowiednie sterowniki. W internecie udało się zlokalizować program firmy ADVANTECH, pozwalający na odczyt sygnału generowanego przez generator.



Zdjęcie 1: Zdjęcie programu wyświetlającego odczyt dla testowego urządzenia

Podłączone urządzenie generujące sygnał poprawnie przesyłało rezultaty przez port USB do komputera, co pozwalało na wyświetlenie ich w formie wykresu. Oznaczało to, że komputer poprawnie widział, i interpretował dane przesyłane z generatora.

Z uzyskaną wiedzą doinstalowano pliki przy pomocy instalatora **XNavi** które umożliwiają na komunikację z systemem DAQ z poziomu kodu. Zmodyfikowano program napisany w Pythonie o komunikację bazując na przykładowych plikach dostarczonych przez firmę ADVANTECH. Wykonane czynności wzorowane na przykładowych plikach nie pozwalały na poprawne uruchomienie programu, w wyniku czego rozpoczęto działania diagnostyczne.

Diagnostykę rozpoczęto od prób uruchomienia plików źródłowych przykładowych działań na systemie DAQ napisanych w języku Python. Zauważono, że nawet programy dostarczone ze sterownikami od firmy ADVANTECH nie działają poprawnie, wywołując błędy z pobraniem bibliotek. Spróbowano uruchomienia programu napisanego w innym języku programowania. Programy utworzone w języku C# działały poprawnie, odpowiednio odczytując dane i wypisując je w konsoli. W wyniku działań diagnostycznych

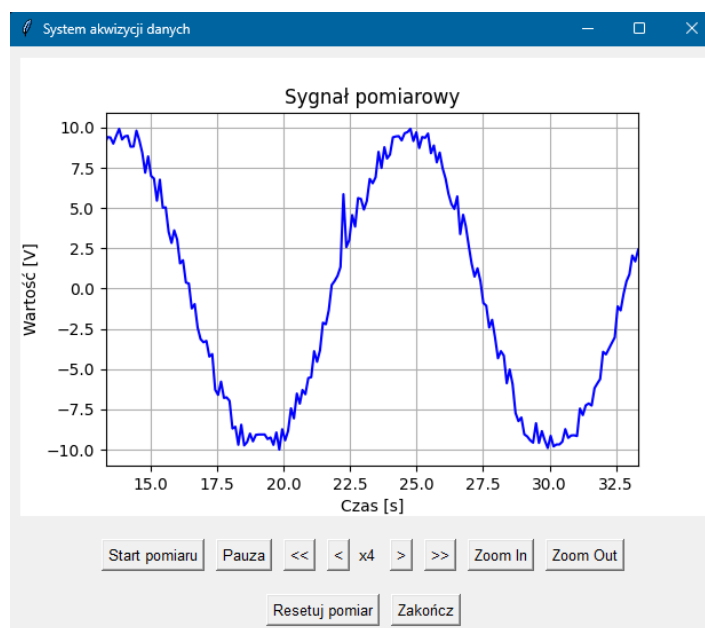
założono brak możliwości utworzenia działającego w pełni programu w Pythonie z powodu niekompletnych bibliotek.

W wyniku diagnostyki dokonana została zmiana strategii, która została zaimplementowana do zadań po zakończeniu się zajęć laboratoryjnych. Wcześniej przygotowany program w Pythonie został zmodyfikowany, aby tylko odczytywał dane z pliku .csv, a następnie zamieniał go w odpowiednio interaktywny wykres. Utworzono nowy program w C#, który zbierał dane generowane przez generator sygnału bazując na przykładowych plikach testowych. Ponieważ wiadomo, że pliki testowe działały poprawnie z podłączonym fizycznym urządzeniem, program testowano na modelu "DEMODEVICE", co pozawalało na generowanie wyników bez fizycznego sprzętu. Oba programy działają prawidłowo, wykonując swoje zadane funkcje, w wyniku czego wykonano poprawnie narzucone zadanie.

4. Opis działania programów

4.1. Program main.py (część Pythona)

Program po uruchomieniu (zakładając posiadane wszystkie używane biblioteki) odczyta dane z pliku **sensor_data.csv** (ważne jest aby plik znajdował się w tym samym folderze co program) a następnie wyświetli jego graf w oknie programu. Program umożliwia również na powiększanie, jak i oddalanie programu.



Zdjęcie 2: Zdjęcie programu napisanego w Pythonie

4.2. Program measurement (część C#)

Program po uruchomieniu wypisze zebrane dane o sygnale do pliku **sensor_data.csv**. Należy zaznaczyć, że aktualna forma programu posiada podłączone DEMODEVICE, ze względu na wykonywanie zadania w momencie braku dostępu do urządzenia fizycznego. Wiadomo jednak, że plik testowy poprawnie odczytywał dane po podłączeniu fizycznego urządzenia, więc zmiana danych urządzenia na **USB-4702** (zakładając, że urządzenie jest podpięte do komputera) powinno poprawnie odczytywać dane generatora sygnału.

5. Wnioski

Najważniejsza wiedza nabyta podczas tych laboratoriów, to bycie świadomym, że nawet producent nie zawsze dostarcza odpowiednio działające oprogramowanie do swojego produktu, w wyniku czego należy posiadać wystarczające zdolności adaptacji do przedstawionej sytuacji, aby móc obejść problem. Nabyta została również wiedza z zakresu działania systemów DAQ, ścieżki jaką przebywa sygnał od punktu mierzonego aż do komputera.

6. Podsumowanie

W wyniku wykonanego zadania zapoznano się z systemem akwizycji danych, sposobem podłączenia tego systemu do komputera oraz metodyki wykorzystania systemu w programie. Podczas wykonywanego zadania natrafiono na przeszkodę, z którą poradzono sobie poprzez zmianę języka programowania wykorzystanego do odczytywania danych systemu DAQ.

7. Skan raportu z zajęć

Grupa D4

Łukasz Duda 280916

Daniel Górciński 280878

Temat 9: Akwizycja sygnałów

- Wejście do sali
- Fizyczne zapoznanie się ze sprzętem
- Dostępny sprzęt na stanowisku
 - └ Generator funkcji ORTEL G-500A
 - └ Blok rozdzielczy ADAM-393F
 - └ Moduł USB-4702
- Znalaziono program do odczytania wartości sygnału. Poprawnie odczytano wartość sygnału oraz zaobserwowano zmiany w sygnale przy zmianie ustawień generatora sygnału
- Na stronie wsparcia znaleziono instalator „XNavi” zawierający wymagane oprogramowanie, sterowniki oraz przykładowe programy napisane w języku Python.
- Przykładowe programy nie działają. Analiza przyczyny problemu pokazała, że powodem są napełnione niekompletne biblioteki. Nawarta dokumentacja nie zawiera odpowiednich informacji wymaganych do pomyślnego uruchomienia programu
- W innym języku programowania (C#) przykłady dostarczone przez producenta działają poprawnie
- Zmodyfikowano wcześniej przygotowany własny program w języku Python aby umożliwić wizualizację pomiarów z już istniejącego pliku z rozszerzeniem .csv

13.X 9. 19:55
