

Politechnika Wrocławska  
AiR ARR  
Projekt zespołowy

# SENSGLOVE

*Autorzy:*  
*Beata Berajter*  
*Dawid Brząkała*  
*Dorota Gidel*  
*Katarzyna Wądrzyk*  
*Ada Weiss*  
*Małgorzata Witka-Jeżewska*

Prowadzący: dr inż. Krzysztof Arent

20 czerwca 2017

# Spis treści

<b>1</b>	<b>Podsumowanie prac dla poszczególnych komponentów</b>	<b>2</b>
1.1	Rękawiczka sensoryczna . . . . .	2
1.1.1	Wykonanie rękawiczki . . . . .	2
1.1.2	Wprowadzone zmiany . . . . .	2
1.2	Interfejs sprzętowy . . . . .	3
1.2.1	Schemat układu . . . . .	3
1.2.2	Lista elementów . . . . .	5
1.2.3	Połączenie z płytą STM32F3Discovery . . . . .	6
1.3	Kod programu . . . . .	6
1.4	Program do akwizycji danych . . . . .	7
1.5	Opis klas programu . . . . .	7
1.5.1	Opis działania programu . . . . .	7
1.5.2	Format danych . . . . .	7
1.6	Baza danych . . . . .	7
1.6.1	Wprowadzone zmiany . . . . .	8
1.6.2	Opis działania skryptu . . . . .	10
1.7	Wizualizacja danych . . . . .	10
1.7.1	Ogólny diagram klas . . . . .	10
1.7.2	Diagram klas zakładek . . . . .	11
1.7.3	Interfejs użytkownika . . . . .	12
<b>2</b>	<b>Integracja komponentów</b>	<b>13</b>
<b>3</b>	<b>Testowanie całości systemu i rozpowszechnianie</b>	<b>14</b>
<b>4</b>	<b>Podsumowanie</b>	<b>14</b>

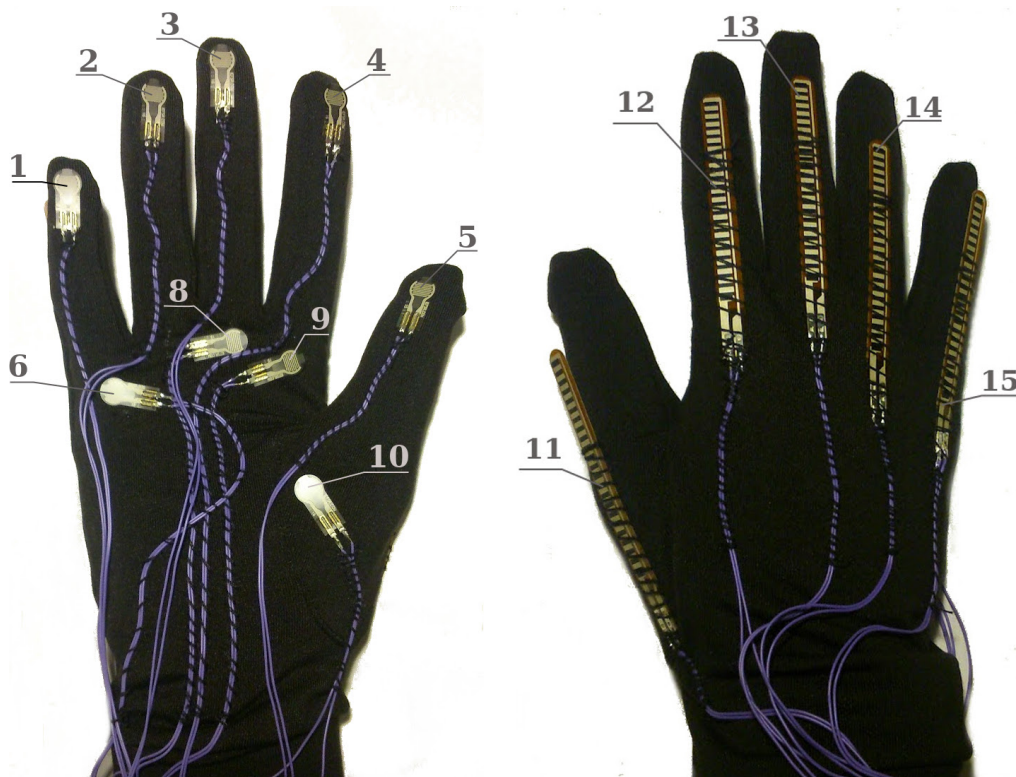
# 1 Podsumowanie prac dla poszczególnych komponentów

## 1.1 Rękawiczka sensoryczna

Osoby przydzielone do zadania: Dorota Gidel, Katarzyna Wądrzyk.

### 1.1.1 Wykonanie rękawiczki

Na rękawiczce wykonanej z poliestru przymocowano 9 czujników nacisku oraz 5 czujników ugięcia. Umieszczenie czujników zostało przedstawione na rysunku 1. Z powodu zniszczenia jednego czujnika nacisku został on odłączony od rękawiczki. Istnieje jednak możliwość ponownego przymocowania go.



Rysunek 1: Umieszczenie i numeracja czujników na rękawiczce

Proces wykonania rękawiczki polegał na przylutowaniu przewodów do czujników, następnie przyklejeniu punktowo klejem czujników w odpowiednich miejscach na rękawiczce oraz przyszycie przewodów do rękawiczki, w przypadku czujników zgięcia przyszyto również część czujnika by mógł się poruszać.

Przyszycie przewodów pozwala na zachowanie porządku i sprzyja ruchom osoby używającej rękawiczkę a przytwierdzenie klejem zapewnia trwałość. W celu zapobiegania uszkodzenia czujników (przede wszystkim haczeniu) uszyte zostały "kapturki" na palce. Dzięki nim sensory nacisku jak i także ugięcia są zabezpieczone przed uszkodzeniami.

Na końcach przewodów odprowadzających sygnały z czujników zamontowane zostały żeńskie końce, które umożliwiają proste podłączenie ich do płytki wzmacniającej. Przymocowane czujniki numerowane są w kolejności przedstawionej na rysunku 1.

### 1.1.2 Wprowadzone zmiany

Planowane było kupienie rękawiczki w rozmiarze S, ze względu na skład grupy, jednakże ostatecznie wybrano rozmiar M. Rękawiczka nadal dobrze dopasowuje się na dłoniach małych ale dodatkowo może z niej

korzystać większe grono osób. Dzięki czemu do tworzenia bazy danych nie będzie potrzeby specjalnie wybierania i szukania osób o danym rozmiarze dłoni.

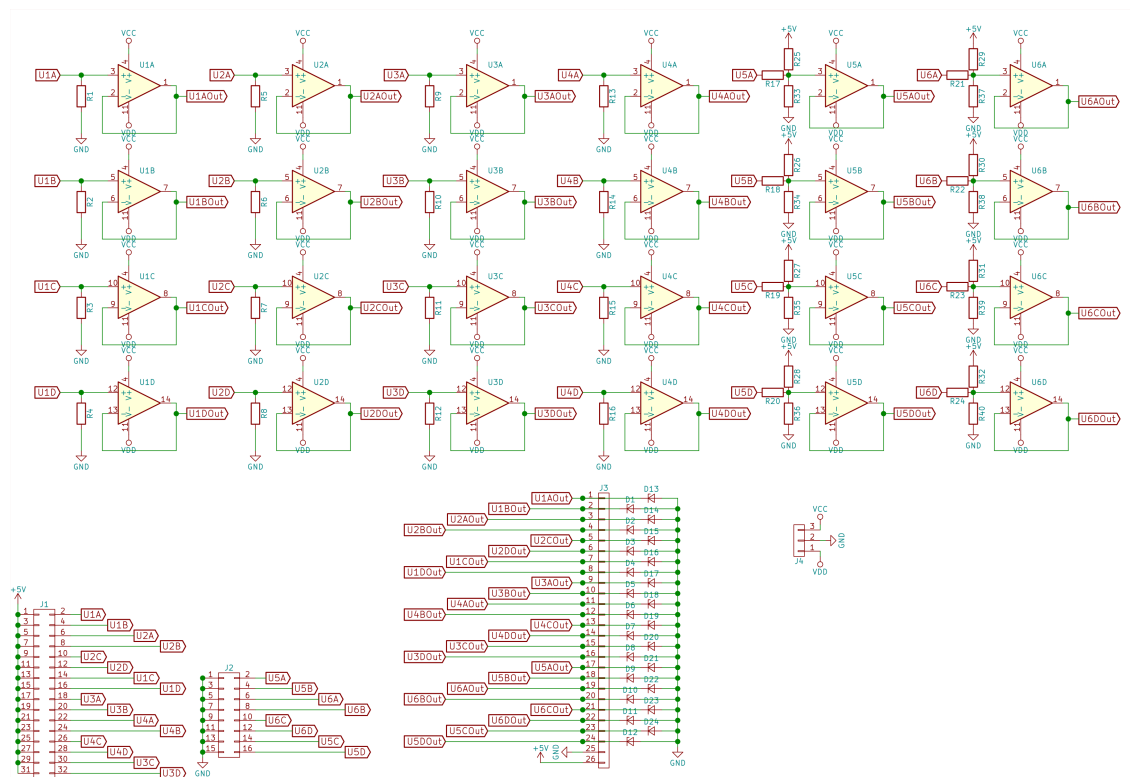
Zaletą większego rozmiaru jest też większa powierzchnia do mocowania czujników. Dużo łatwiej jest rozmieścić wszystkie sensory, po ich przymocowaniu nie ograniczają one ruchów w tak dużym stopniu jakby miało to miejsce przy rękawiczce o rozmiarze S.

Ze względu uszkodzenia się czujników, rękawiczka posiada 9 czujników nacisku. Jak widać na zdjęciu 1 czujnik 7 został pominięty, a czujnik 8 lekko przesunięty w lewą stronę by to zrekompensować. Jednakże dalej istnieje możliwość przymocowania brakującego czujnika 7.

## 1.2 Interfejs sprzętowy

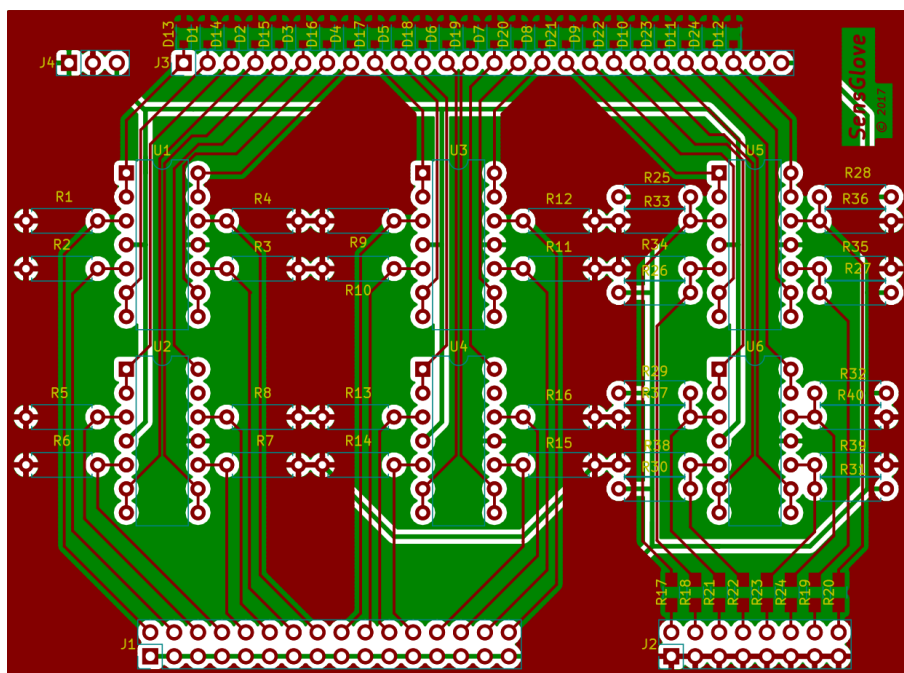
Osoby przydzielone do zadania: Beata Berajter i Dawid Brząkała.

### 1.2.1 Schemat układu

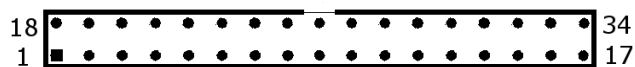


Rysunek 2: schemat układu

Projekt płytki pcb i wyprowadzenie pinów układu przedstawiono na rysunku 3.

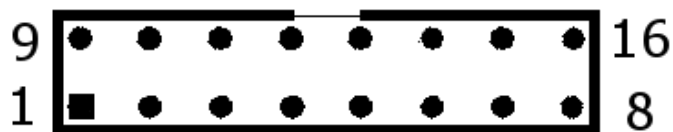


Rysunek 3: Schemat płytki pcb



Rysunek 4: J1 - pinout

- 1 - 16 zasilanie czujników - +5v
- 17 - 27 złącza czujników nacisku 1 - 10
- 28 - 33 złącza czujników ugięcia 1 - 6
- 2, 34 NC



Rysunek 5: J2 - pinout

- 1 - 7 GND
- 8 - 16 złącza czujników biosygnatów 1 - 8



Rysunek 6: J3 - pinout

1 - 10	sygnał czujników nacisku 1 - 10
11 - 16	sygnał czujników ugięcia 1 - 6
17 - 24	sygnał czujników biosygnałów 1 - 8
25	+5V
26	GND



Rysunek 7: J4 - pinout

1	-9V
2	GND
3	+9V

### 1.2.2 Lista elementów

U1-U6	TLC084CN	THT
R1-R10	10kOhm	THT
R11-R16	33kOhm	THT
R17-R24	3.3kOhm	SMD
R25-R32	3.3kOhm	THT
R33-R40	2.2kOhm	THT
D1-D24	3.3V Zenera	SMD
J1	IDC-34 żeńskie	THT
J2	IDC-16 żeńskie	THT
J3	listwa kołkowa 1x26 2.56mm	THT
J4	listwa kołkowa 1x3 2.56mm	THT

### 1.2.3 Połączenie z płytką STM32F3Discovery

J1	STM32Discovery
1	PB12
2	PB14
3	PB15
4	PD8
5	PD9
6	PD10
7	PD11
8	PD12
9	PD13
10	PD14
11	PB0
12	PB1
13	PE7
14	PE8
15	PE9
16	PE10
17	PA0
18	PA3
19	PA2
20	PF4
21	PA4
22	PA5
23	PA6
24	PA7
25	GND
26	5V

### 1.3 Kod programu

```
//struktura danych do zapisu odczytu sygnałów
typedef union
{
    struct
    {
        uint16_t forceSensors[10];
        uint16_t flexSensors[6];
        uint16_t biosignals[8];
        uint16_t checksum;
    }measure;
    uint16_t measures[25];
} data_t;
data_t data;

...

//konfiguracja DMA
HAL_ADC_Start_DMA(&hadc1, &data.measure.biosignals[0], 4);
HAL_ADC_Start_DMA(&hadc2, &data.measure.biosignals[4], 4);
HAL_ADC_Start_DMA(&hadc3, &data.measure.flexSensors[0], 6);
HAL_ADC_Start_DMA(&hadc4, &data.measure.forceSensors[0], 10);
```

```
data.measure.checksum = 0;
for(int i = 0; i < 24; i++)
{
    data.measure.checksum+=data.measures[i];
}
printf("X%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x%4.4x\r\n",
data.measures[0], data.measures[1], data.measures[2], data.measures[3], data.measures[4],
data.measures[5], data.measures[6], data.measures[7], data.measures[8], data.measures[9],
data.measures[10], data.measures[11], data.measures[12], data.measures[13], data.measures[14],
data.measures[15], data.measures[16], data.measures[17], data.measures[18], data.measures[19],
data.measures[20], data.measures[21], data.measures[22], data.measures[23], data.measures[24]);
```

## 1.4 Program do akwizycji danych

Osoby przedzielone do zadania: Ada Weiss, Małgorzata Witka-Jezewska

## 1.5 Opis klas programu

Dokładny opis klas i poszczególnych parametrów tworzony jest za pomocą programu Doxygen.

### 1.5.1 Opis działania programu

Interfejs graficzny programu został napisany z wykorzystaniem klas biblioteki Qt5 i został zaprezentowany na rys. 10.

Zasada działania została przedstawiona na rys. 9. Pierwszym krokiem jest wybór portu USB i wciśnięcie przycisku start, co uruchamia konfigurację wybranego portu USB. W przypadku braku podłączonego urządzenia na danym porcie wyświetlany jest komunikat przedstawiony na rys. 11. Kliknięcie "Yes" umożliwia ponowny wybór, a "No" kończy pracę programu.

Dane są odczytywane z portu USB w oddzielnym wątku `Read_thread`, dziedziczącego po `QThread`. Zapewnia to odpowiednio szybką pracę programu. W tym samym wątku sprawdzana jest poprawność otrzymanej wiadomości (obliczanie sumy kontrolnej, sprawdzanie długości ciągu znaków) oraz wysyłanie danych do Socketu, który również został zaprogramowany z wykorzystaniem bibliotek Qt5 (klasa `Socket`).

W przypadku aktywowania przycisku "Miejsce pomiaru" jest timer (na 2s) i dane te są zapisywane w wektorze 2000x23, z którego to jest możliwość zapisu danych do pliku w przypadku akceptacji pomiaru po jego ukończeniu. Widok okna w trakcie pomiaru przedstawia 12.

Akceptacja pomiaru wymaga również podania imienia użytkownika oraz wyboru ruchu który jest wykonywany. Program wywołuje po zapisie do pliku skrypt który umieszcza plik w bazie danych.

Można również dany pomiar zignorować, np. w przypadku stwierdzenia, że został on błędnie wykonany.

### 1.5.2 Format danych

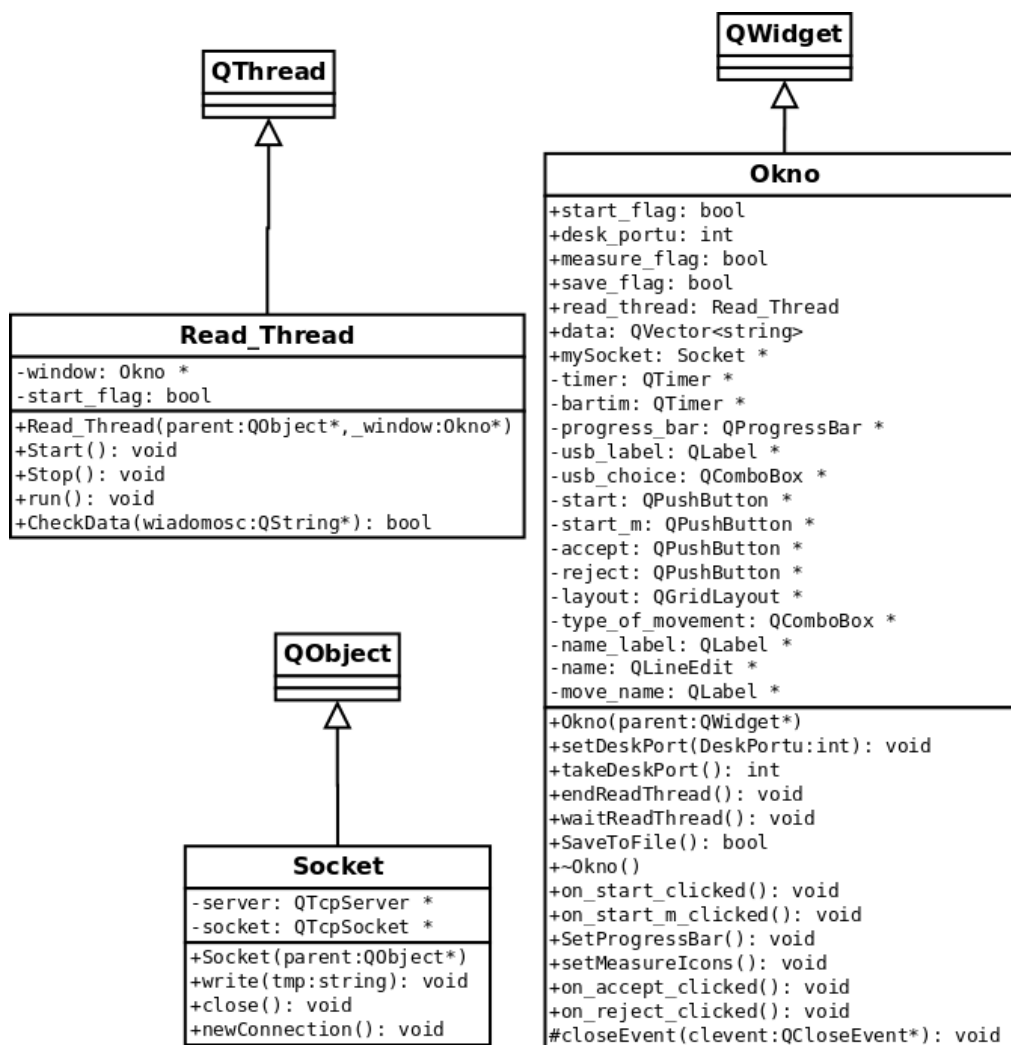
Ramka danych:

- znak X,
- 23 sygnały z czujników w postaci 4 znaków w systemie szesnastkowym,
- suma kontrolna w postaci dwóch znaków w systemie szesnastkowym,
- znak końca linii \r \n;

## 1.6 Baza danych

Osoby przydzielone do zadania: Beata Berajter, Dorota Gidel.

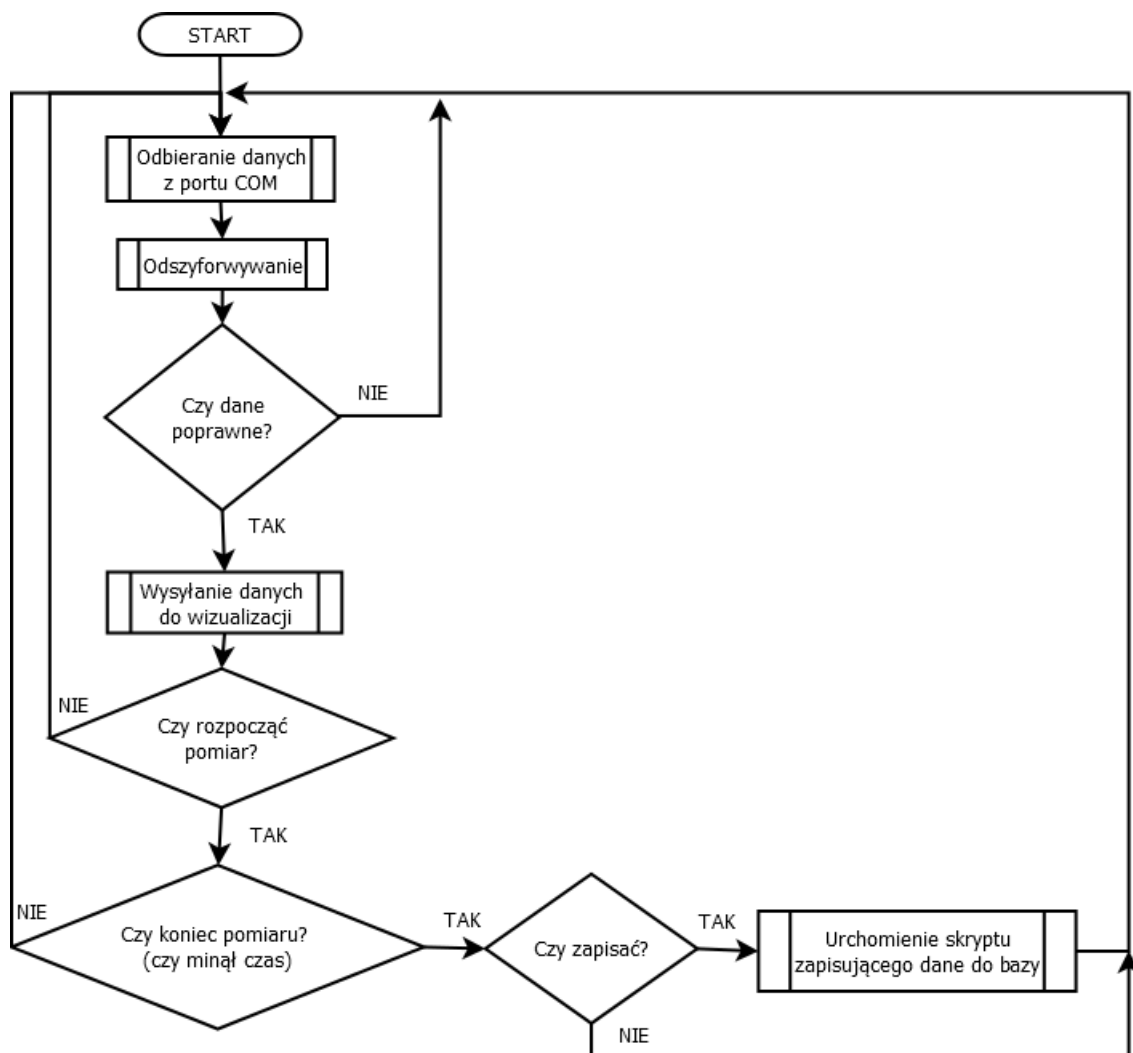




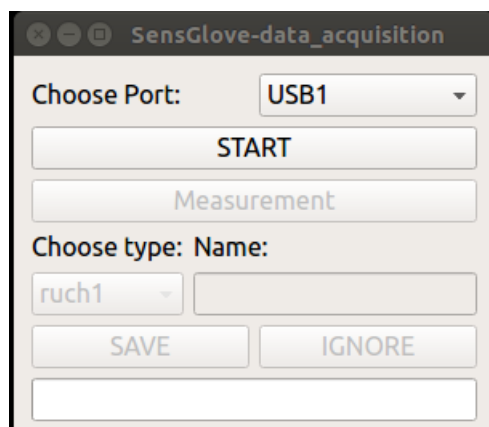
Rysunek 8: Diagram klas programu

### 1.6.1 Wprowadzone zmiany

Zdecydowano się na zmianę struktury bazy danych. Zamiast umieszczania trzech osobnych plików zawierających dane odpowiednio z czujników nacisku, zgięcia i elektrod, tworzony będzie tylko jeden plik zawierający wszystkie te dane w takiej kolejności. Plik taki zawierać będzie 2000 linii i 23 kolumny odczytów. Dodatkowo, nazwa pliku nie będzie zawierać daty gdyż można ją uzyskać z daty tworzenia pliku. Nowy schemat struktury bazy danych przedstawiono na rysunku 13.

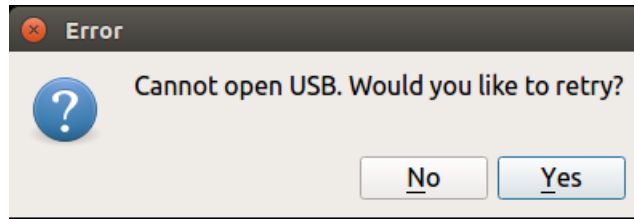


Rysunek 9: Schemat ideowy działania programu

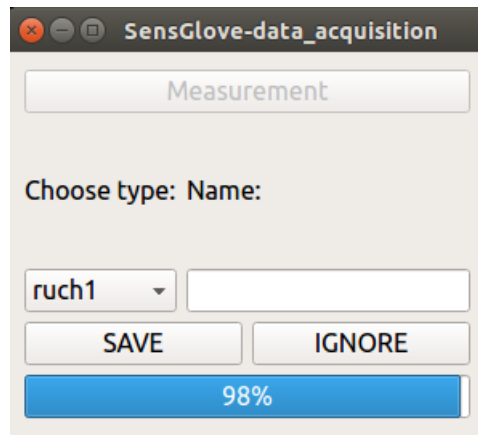


Rysunek 10: Okno główne programu

BAZA DANYCH						
Dorota_Gidel			Beata_Berajter		Imie_Nazwisko	
Ruch_A			9		.	
Ruch_B			Ruch_C		.	
.			.		.	
.			.		.	



Rysunek 11: Komunikat o błędnym otwarciu USB.



Rysunek 12: Okno główne programu w trakcie pomiaru

### 1.6.2 Opis działania skryptu

Stworzony skrypt do umieszczania w odpowiednim miejscu plików z odczytami napisany jest w powłoce bash. Przykładowe wywołanie skryptu wygląda następująco:

```
$ bash przeniesPlik sciezka/do/pliku/z/danymi Imie NazwaRuchu
```

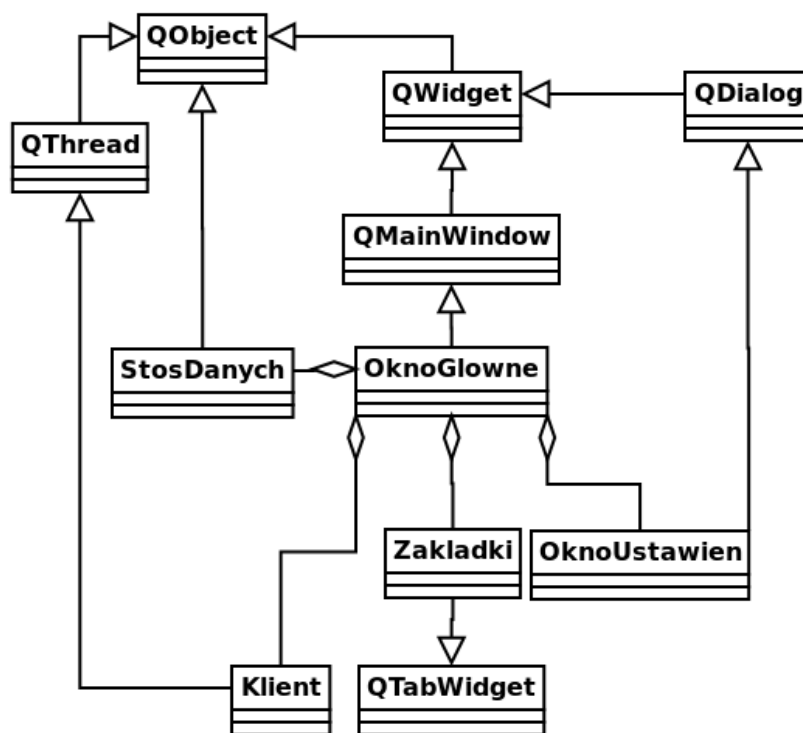
Skrypt przenosi podany plik (*sciezka/do/pliku/z/danymi*) do głównego folderu bazy danych który zdefiniowany jest jako katalog o nazwie "baza\_danych" w katalogu domowym. Dzięki podanym argumentom umieszcza plik w odpowiednim miejscu. Numeracja pliku po kolei jest zapewniona przez prowadzenie spisu w osobnym pliku tekstowym należącym do bazy danych. Dla bezpieczeństwa sprawdzane jest też czy w wywołaniu skryptu została podana wystarczająca liczba argumentów.

## 1.7 Wizualizacja danych

Osoby przydzielone do zadania: Dorota Gidel, Katarzyna Wądrzyk.

### 1.7.1 Ogólny diagram klas

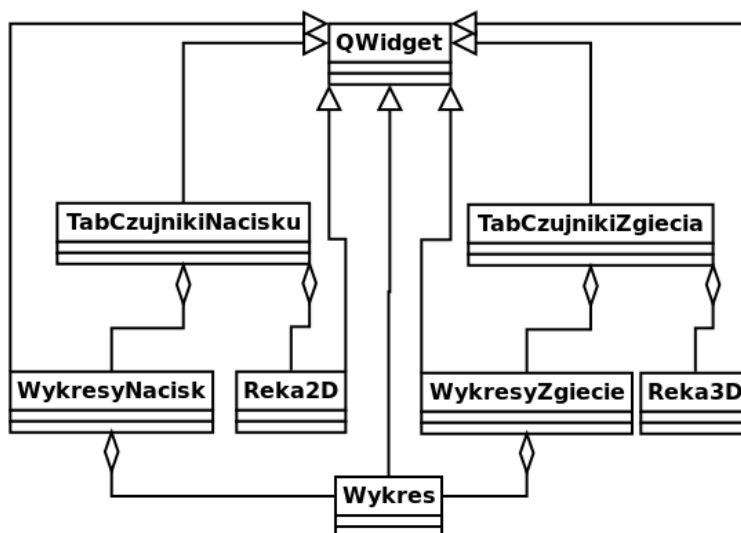
Na rysunku 14 zaprezentowano ogólny diagram klas w programie wizualizującym dane z czujników.



Rysunek 14: Ogólny diagram klas - wizualizacja

### 1.7.2 Diagram klas zakładek

Na obrazku 15 zaprezentowano ogólny diagram klas zakładek zawierających wykresy i prezentacje odpowiedzi z czujników nacisku oraz zgięcia.



Rysunek 15: Diagram klas zakładek - wizualizacja

### 1.7.3 Interfejs użytkownika

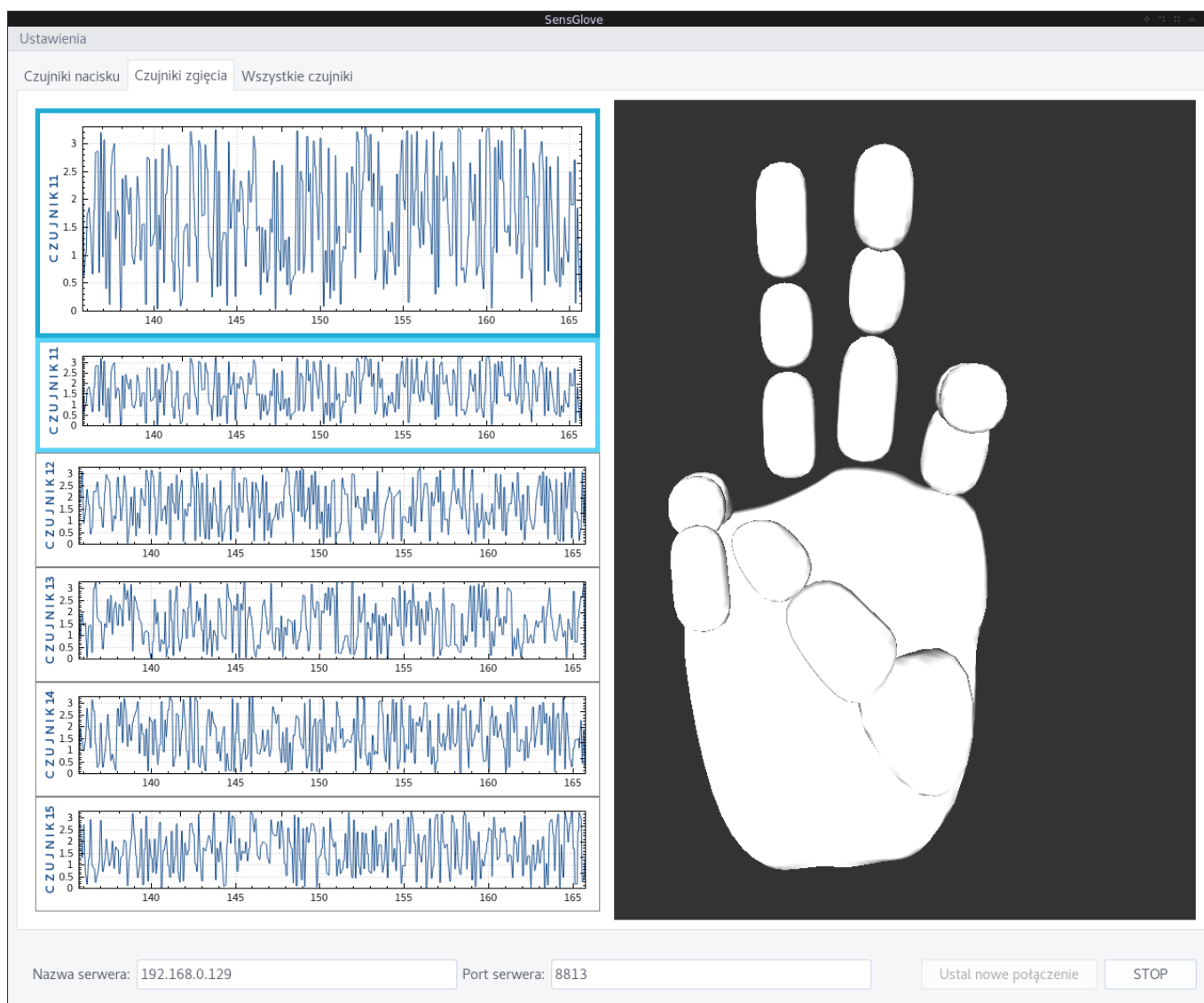
Użytkownik ma możliwość połączenia się z portem nadającym dane przez przyciśnięcie przycisku "Ustal nowe połączenie". W trakcie odbierania danych istnieje możliwość zatrzymania wykresów i wizualizacji dłoni w danym momencie poprzez przyciśnięcie przycisku "Stop". W lewym górnym rogu użytkownik ma dostęp do ustawień w których obecnie może zmienić zakres czasu prezentowanego na wykresach. Domyślnie wartość ta wynosi 30 s.

Rysunek 16 przedstawia pierwszą zakładkę aplikacji wizualizującej dane - czujniki nacisku. Po lewej stronie umieszczone są wykresy. Pierwszy największy wykres, można zmieniać przyciskając odpowiednio wykres który chcemy najbardziej śledzić. Wybrany wykres zaznaczony jest przez kolorową obwódkę. Po prawej stronie znajduje się widget zawierający rękę 2D prezentującą przez kolory odczytywane w danym momencie wartości napięcia.



Rysunek 16: Diagram klas zakładek - wizualizacja

Rysunek 17 przedstawia drugą zakładkę aplikacji wizualizującej dane - czujniki zgięcia. Po lewej stronie tak samo jak poprzednio znajdują się wykresy odczytywanego napięcia. Po prawej stronie znajduje się widget zawierający rękę 3D której palce zginają się odpowiednio do otrzymywanych wartości z czujników.



Rysunek 17: Diagram klas zakładek - wizualizacja

## 2 Integracja komponentów

Po przeprowadzeniu ewaluacji poszczególnych komponentów zostały one połączone ze sobą w celu integracji.

1. Rękawiczka założona na rękę pozwala na możliwie swobodne poruszanie palcami.
2. Podczas wykonywania ruchów czujniki oraz przewody nie haczą się o nic dzięki dołączonym kapturkom oraz dodatkowej bezpalcowej osłonie.
3. Sygnały z czujników prowadzone przez przewody są bezpośrednio podłączone do styków na wzmacniaczu dzięki ich żeńskim końcówkom.
4. Na przedramieniu osoby badanej umieszczono sensory biosygnalów w sposób pokazany w instrukcji "Stanowsiko badawcze do akwizycji biosygnalów" autorstwa A. Wołczowskiego i J.S. Witkowskiego.
5. Płytkę ze wzmacniaczami połączoną z płytką deweloperską Discovery, zasilaniem bateryjnym, wzmacniaczami biosygnalów oraz czujnikami nacisku oraz ugięcia.

6. Dane z płytki Discovery płynnie przesyłane są poprzez przewód USB do komputera z włączonym programem do akwizycji danych.
7. Poprzez obsługę programu do akwizycji danych rozpoczynany jest pomiar poprzez przyciśnięcie przycisku "Start".
8. We włączonym programie od wizualizacji w odpowiednich zakładkach na bieżąco można obserwować przebieg sygnałów z czujników: nacisku, ugięcia i biosygnałów.
9. Przy chwytaniu przedmiotów czujniki ugięcia działają sprawnie, o czym świadczy ugięcie palców na wirtualnej dłoni oraz wykresy z odczytami wartości rezystancji z czujników.
10. Po nałożeniu kapturków sensory nacisku na wewnętrznej części dłoni sprawnie odczytują siłę nacisku. Świadczy o tym zmiana koloru odpowiednich pól na wizualizowanej dłoni.
11. Przy wybraniu odpowiednich zakładek widać na poszczególnych wykresach jak ruch wpływa na sygnały z czujników: ugięcia, nacisku i biosygnałów.
12. Przy wciśnięciu przycisku "Measurement" w programie do akwizycji danych pojawia się pasek ładowania odmierzający 2 s. Po tym czasie pojawia się możliwość zapisu lub zignorowania danych.
13. Po wybraniu ruchu i wpisaniu przykładowej nazwy użytkownika oraz przyciśnięciu przycisku "Save".
14. Wartości z pomiarów zostały zapisane do odpowiedniego folderu.

Aplikacja odbiera pomyślnie dane z programu do akwizycji oraz wizualizuje zinterpretowane pomiary na wykresach i modelach dłoni.

### 3 Testowanie całości systemu i rozpowszechnianie

Po połączeniu wszystkich komponentów projektu przeprowadzono testy. Polegały one na wykonywaniu ruchów w rękawiczce sensorycznej i obserwowanie efektów w programie wizualizującym oraz sprawdzenie poprawnego zapisu pliku do bazy danych. Testy pokazały, że sygnały przesyłane są odbierane poprawnie, co pokazują zdjęcia i film umieszczony na stronie internetowej <http://sensglove.happyrobotics.com> w odpowiednich zakładkach. Na stronie został również zamieszczony link do strony GitHub <https://github.com/Sensglove/SG>, na której zamieszczony jest kod oprogramowania i dokumentacja.

### 4 Podsumowanie