

My Project

Generated by Doxygen 1.9.6

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 MainWindow Class Reference	7
4.1.1 Detailed Description	8
4.1.2 Constructor & Destructor Documentation	8
4.1.2.1 MainWindow()	8
4.1.3 Member Function Documentation	8
4.1.3.1 handlePortConnected	8
4.1.3.2 on_actionInformacje_o_programie_triggered	9
4.1.3.3 on_actionKonfiguracja_polaczenia_triggered	9
4.1.3.4 on_actionPolacz_ze_skonfigurowanym_portem_triggered	9
4.1.3.5 on_actionZapisz_dane_do_pliku_CSV_triggered	9
4.1.3.6 on_pushButton_clicked	9
4.1.3.7 readFromPort	10
4.2 Pomoc Class Reference	10
4.2.1 Detailed Description	10
4.2.2 Constructor & Destructor Documentation	10
4.2.2.1 Pomoc()	10
4.2.3 Member Function Documentation	11
4.2.3.1 calculateCRC8()	11
4.2.3.2 portConnected	12
4.3 wyswietlanie Class Reference	12
4.3.1 Detailed Description	13
4.3.2 Constructor & Destructor Documentation	13
4.3.2.1 wyswietlanie()	13
4.3.3 Member Function Documentation	13
4.3.3.1 ustaw_wykres()	13
5 File Documentation	15
5.1 konfiguracja.h	15
5.2 mainwindow.h	15
5.3 wyswietlanie.h	16

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

QDialog	
Pomoc	10
wyswietlanie	12
QMainWindow	
MainWindow	7

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

MainWindow		
	Klasa MainWindow reprezentuje główne okno aplikacji	7
Pomoc		
	Klasa Pomoc reprezentuje okno pomocnicze	10
wyswietlanie		
	Klasa dialogowa odpowiedzialna za wyświetlanie danych	12

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

konfiguracja.h	??
mainwindow.h	??
wyswietlanie.h	??

Chapter 4

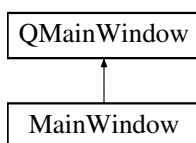
Class Documentation

4.1 MainWindow Class Reference

Klasa [MainWindow](#) reprezentuje główne okno aplikacji.

```
#include <mainwindow.h>
```

Inheritance diagram for MainWindow:



Public Slots

- void [on_actionInformacje_o_programie_triggered](#) ()
Slot wywoływany po wciśnięciu akcji "Informacje o programie".
- void [on_actionKonfiguracja_polaczenia_triggered](#) ()
Slot wywoływany po wciśnięciu akcji "Konfiguracja połączenia".
- void [handlePortConnected](#) (QString portName)
Slot wywoływany po połączeniu z portem.
- void [on_actionPolacz_ze_skonfigurowanym_portem_triggered](#) ()
Slot wywoływany po wciśnięciu akcji "Połącz ze skonfigurowanym portem".
- void [readFromPort](#) ()
Slot wywoływany po otrzymaniu danych z portu.
- void [on_pushButton_clicked](#) ()
Slot wywoływany po wciśnięciu przycisku.
- void [on_lineEdit_editingFinished](#) ()
Slot wywoływany po zakończeniu edycji pola tekstowego.
- void [on_actionRoz_cz_triggered](#) ()
Slot wywoływany po wciśnięciu akcji "Rozłącz".
- void [on_actionZapisz_dane_do_pliku_CSV_triggered](#) ()
Slot wywoływany po wciśnięciu akcji "Zapisz dane do pliku CSV".
- void [on_actionInstrukcja_u_ytkowania_triggered](#) ()
Slot wywoływany po wciśnięciu akcji "Instrukcja użytkownika".
- void [on_actionWyswietlanie_triggered](#) ()
Slot wywoływany po wciśnięciu akcji "Wyswietlanie".

Public Member Functions

- [MainWindow](#) (QWidget *parent=nullptr)
Konstruktor klasy [MainWindow](#).
- [~MainWindow](#) ()
Destruktor klasy [MainWindow](#).

Public Attributes

- int **flag1** =1
- int **flag2** =1

4.1.1 Detailed Description

Klasa [MainWindow](#) reprezentuje główne okno aplikacji.

Obsługuje różne interakcje z interfejsem użytkownika oraz komunikację z urządzeniami poprzez port szeregowy.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 MainWindow()

```
MainWindow::MainWindow (
    QWidget * parent = nullptr )
```

Konstruktor klasy [MainWindow](#).

Parameters

<i>parent</i>	Wskaźnik na obiekt rodzica (domyślnie: nullptr)
<i>parent</i>	Wskaźnik na rodzica obiektu.

4.1.3 Member Function Documentation

4.1.3.1 handlePortConnected

```
void MainWindow::handlePortConnected (
    QString portName ) [slot]
```

Slot wywoływany po połączeniu z portem.

Obsługa podłączenia portu. Ustawia nazwę portu dla obiektu device.

Parameters

<i>portName</i>	Nazwa połączonego portu
<i>portName</i>	Nazwa portu.

4.1.3.2 on_actionInformacje_o_programie_triggered

```
void MainWindow::on_actionInformacje_o_programie_triggered ( ) [slot]
```

Slot wywoływany po wciśnięciu akcji "Informacje o programie".

Obsługa akcji Informacje o programie. Wyświetla okno dialogowe z informacjami o programie.

4.1.3.3 on_actionKonfiguracja_polaczenia_triggered

```
void MainWindow::on_actionKonfiguracja_polaczenia_triggered ( ) [slot]
```

Slot wywoływany po wciśnięciu akcji "Konfiguracja połączenia".

Obsługa akcji Konfiguracja połączenia. Wyświetla okno pomocy.

4.1.3.4 on_actionPolacz_ze_skonfigurowanym_portem_triggered

```
void MainWindow::on_actionPolacz_ze_skonfigurowanym_portem_triggered ( ) [slot]
```

Slot wywoływany po wciśnięciu akcji "Połącz ze skonfigurowanym portem".

Slot obsługujący akcję połączenia ze skonfigurowanym portem. Sprawdza zawartość pliku "devices.txt" i próbuje nawiązać połączenie z każdym urządzeniem z listy.

4.1.3.5 on_actionZapisz_dane_do_pliku_CSV_triggered

```
void MainWindow::on_actionZapisz_dane_do_pliku_CSV_triggered ( ) [slot]
```

Slot wywoływany po wciśnięciu akcji "Zapisz dane do pliku CSV".

Zapisuje dane do pliku CSV.

4.1.3.6 on_pushButton_clicked

```
void MainWindow::on_pushButton_clicked ( ) [slot]
```

Slot wywoływany po wciśnięciu przycisku.

Obsługa kliknięcia przycisku. Ustawia wykres na podstawie wybranych opcji z okna wyświetlania. Ustawia zakres osi Y na wykresie. Ustawia flagi flag1 i flag2 w zależności od stanu checkboxów w oknie wyświetlania.

4.1.3.7 readFromPort

```
void MainWindow::readFromPort ( ) [slot]
```

Slot wywoływany po otrzymaniu danych z portu.

Slot obsługujący odczyt danych z portu szeregowego. Aktualizuje wykresy w oknie wyświetlania i przetwarza odebrane dane.

The documentation for this class was generated from the following files:

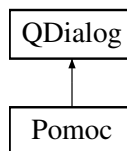
- mainwindow.h
- mainwindow.cpp

4.2 Pomoc Class Reference

Klasa [Pomoc](#) reprezentuje okno pomocnicze.

```
#include <konfiguracja.h>
```

Inheritance diagram for Pomoc:



Signals

- void [portConnected](#) (QString portName)
Sygnał emitowany po nawiązaniu połączenia z portem.

Public Member Functions

- [Pomoc](#) (QWidget *parent=nullptr)
Konstruktor klasy [Pomoc](#).
- uint8_t [calculateCRC8](#) (float value1, float value2, int value3, int value4)
Funkcja oblicza sumę kontrolną CRC8 na podstawie podanych wartości.
- ~[Pomoc](#) ()
Destruktor klasy [Pomoc](#).

4.2.1 Detailed Description

Klasa [Pomoc](#) reprezentuje okno pomocnicze.

Pozwala na interakcję z użytkownikiem w celu konfiguracji połączenia i wysyłania danych do urządzenia.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 Pomoc()

```
Pomoc::Pomoc (
    QWidget * parent = nullptr ) [explicit]
```

Konstruktor klasy [Pomoc](#).

Parameters

<i>parent</i>	Wskaźnik na obiekt rodzica (domyślnie: nullptr)
<i>parent</i>	Wskaźnik do rodzica okna.

4.2.3 Member Function Documentation

4.2.3.1 calculateCRC8()

```
uint8_t Pomoc::calculateCRC8 (
    float value1,
    float value2,
    int value3,
    int value4 )
```

Funkcja oblicza sumę kontrolną CRC8 na podstawie podanych wartości.

Oblicza sumę kontrolną CRC-8 dla wartości przekazanych jako argumenty.

Parameters

<i>value1</i>	Wartość 1
<i>value2</i>	Wartość 2
<i>value3</i>	Wartość 3
<i>value4</i>	Wartość 4

Returns

Suma kontrolna CRC8

Parameters

<i>value1</i>	Pierwsza wartość do obliczenia sumy kontrolnej.
<i>value2</i>	Druga wartość do obliczenia sumy kontrolnej.
<i>value3</i>	Trzecia wartość do obliczenia sumy kontrolnej.
<i>value4</i>	Czwarta wartość do obliczenia sumy kontrolnej.

Returns

Suma kontrolna CRC-8 obliczona dla wartości.

4.2.3.2 portConnected

```
void Pomoc::portConnected (
    QString portName ) [signal]
```

Sygnał emitowany po nawiązaniu połączenia z portem.

Parameters

<i>portName</i>	Nazwa połączonego portu
-----------------	-------------------------

The documentation for this class was generated from the following files:

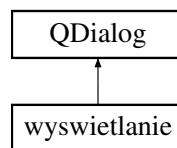
- konfiguracja.h
- konfiguracja.cpp

4.3 wyświetlanie Class Reference

Klasa dialogowa odpowiedzialna za wyświetlanie danych.

```
#include <wyświetlanie.h>
```

Inheritance diagram for wyświetlanie:



Public Member Functions

- [wyświetlanie](#) (QWidget *parent=nullptr)
Konstruktor klasy wyświetlanie.
- [~wyświetlanie](#) ()
Destruktor klasy wyświetlanie.
- void [ustaw_wykres](#) ()
Metoda ustawiająca wykres.

Public Attributes

- bool **Ctem1** =true
- bool **Ctem2** =true
- bool **Ctacho** =true
- bool **Cbelka** =true
- int **zakresX** =50
- int **zakresY** =50
- Qt::GlobalColor **kolor1**
- Qt::GlobalColor **kolor2**
- Qt::PenStyle **linia1**
- Qt::PenStyle **linia2**

4.3.1 Detailed Description

Klasa dialogowa odpowiedzialna za wyświetlanie danych.

4.3.2 Constructor & Destructor Documentation

4.3.2.1 wyświetlanie()

```
wyswietlanie::wyswietlanie (
    QWidget * parent = nullptr ) [explicit]
```

Konstruktor klasy wyswietlanie.

Parameters

<i>parent</i>	Wskaźnik na rodzica obiektu.
<i>parent</i>	Wskaźnik na obiekt rodzica.

4.3.3 Member Function Documentation

4.3.3.1 ustaw_wykres()

```
void wyswietlanie::ustaw_wykres ( )
```

Metoda ustawiająca wykres.

Ustawia wykres na podstawie wybranych opcji. Ustala kolory i style linii dla serii danych na wykresie.

The documentation for this class was generated from the following files:

- wyswietlanie.h
- wyswietlanie.cpp

Chapter 5

File Documentation

5.1 konfiguracja.h

```
00001 #ifndef KONFIGURACJA_H
00002 #define KONFIGURACJA_H
00003
00004 #include <QDialog>
00005 #include <QSerialPort>
00006
00007 namespace Ui {
00008 class Pomoc;
00009 }
00010
00016 class Pomoc : public QDialog
00017 {
00018     Q_OBJECT
00019
00020 public:
00025     explicit Pomoc(QWidget *parent = nullptr);
00026
00035     uint8_t calculateCRC8(float value1, float value2, int value3, int value4);
00036
00040     ~Pomoc();
00041
00042 private slots:
00046     void on_ROZLACZ_clicked();
00047
00051     void on_POLACZ_clicked();
00052
00056     void on_SZUKAJ_clicked();
00057
00061     void on_pushButton_clicked();
00062
00066     void readFromPort();
00067
00072     void sendMessageToDevice(QString message);
00073
00074 private:
00075     Ui::Pomoc *ui2;
00076     void addToLogs(QString message);
00077     void readData(QString receivedString);
00078     Pomoc *oknoPomocy;
00079     QSerialPort *device;
00080
00081 signals:
00086     void portConnected(QString portName);
00087 };
00088
00089 #endif // KONFIGURACJA_H
```

5.2 mainwindow.h

```
00001
00002 #ifndef MAINWINDOW_H
00003 #define MAINWINDOW_H
00004
00005 #include <QMainWindow>
```

```

00006 #include <QSerialPort>
00007 #include <QObject>
00008 #include "konfiguracja.h"
00009 #include "ui_mainwindow.h"
00010 #include "wyswietlanie.h"
00011 #include "qcgaugewidget.h"
00012 #include "qcustomplot.h"
00013 #include <QtCharts>
00014 #include <QtCharts/QChartView>
00015 #include <QtCharts/QBarSet>
00016 #include <QtCharts/QBarSeries>
00017 #include <QtCharts/QValueAxis>
00018 #include <QtCharts/QCategoryAxis>
00019
00020
00021
00022 QT_BEGIN_NAMESPACE
00023 namespace Ui { class MainWindow; }
00024 QT_END_NAMESPACE
00025
00031 class MainWindow : public QMainWindow
00032 {
00033     Q_OBJECT
00034
00035 public:
00040     MainWindow(QWidget *parent = nullptr);
00041
00045     ~MainWindow();
00046
00047 public slots:
00051     void on_actionInformacje_o_programie_triggered();
00052
00056     void on_actionKonfiguracja_polaczenia_triggered();
00057
00062     void handlePortConnected(QString portName);
00063
00067     void on_actionPolacz_ze_skonfigurowanym_portem_triggered();
00068
00072     void readFromPort();
00073
00077     void on_pushButton_clicked();
00081     void on_lineEdit_editingFinished();
00082
00086     void on_actionRoz_cz_triggered();
00087
00091     void on_actionZapisz_dane_do_pliku_CSV_triggered();
00092
00096     void on_actionInstrukcja_u_ytkowania_triggered();
00097
00101     void on_actionWyswietlanie_triggered();
00102
00103
00104
00105 public:
00106     int flag1=1;
00107     int flag2=1;
00108
00109
00110 private:
00111     Ui::MainWindow *ui;
00112     Pomoc *oknoPomocy;
00113     wyswietlanie *oknoWyswietlania;
00114     QSerialPort *device;
00115     void readData(QString receivedString);
00116     QCPBars *gauge;
00117     QcGaugeWidget * mSpeedGauge;
00118     QcNeedleItem *mSpeedNeedle;
00119     QBarSet *set1;
00120     QBarSeries *series;
00121     QChart *chart;
00122     QChartView *chartView;
00123
00124     QCPAxis *axis2;
00125
00126
00127
00128 };
00129
00130 #endif // MAINWINDOW_H

```

5.3 wyswietlanie.h

```

00001 #ifndef WYSWIETLANIE_H

```

```
00002 #define WYSWIETLANIE_H
00003
00004 #include <QDialog>
00005
00006 namespace Ui {
00007 class wyswietlanie;
00008 }
00009
00013 class wyswietlanie : public QDialog
00014 {
00015     Q_OBJECT
00016
00017 public:
00022     explicit wyswietlanie(QWidget *parent = nullptr);
00023
00027     ~wyswietlanie();
00028
00029     bool Ctem1=true; // Zmienna oznaczająca, czy tem1 jest zaznaczone
00030     bool Ctem2=true; // Zmienna oznaczająca, czy tem2 jest zaznaczone
00031     bool Ctacho=true; // Zmienna oznaczająca, czy tacho jest zaznaczone
00032     bool Cbelka=true; // Zmienna oznaczająca, czy belka jest zaznaczona
00033     int zakresX=50; // Zakres osi X na wykresie
00034     int zakresY=50; // Zakres osi Y na wykresie
00035     Qt::GlobalColor kolor1; // Kolor dla danych tem1
00036     Qt::GlobalColor kolor2; // Kolor dla danych tem2
00037     Qt::PenStyle linia1; // Styl linii dla danych tem1
00038     Qt::PenStyle linia2; // Styl linii dla danych tem2
00039
00043     void ustaw_wykres();
00044
00045 private slots:
00049     void on_pushButton_clicked();
00050
00051
00052
00056     void on_checkBoxTemp2_clicked();
00057
00058
00062     void on_checkBoxBelka_clicked();
00063
00068     void on_horizontalSlider_osY_valueChanged(int value);
00069
00073     void on_checkBoxTacho_stateChanged(int arg1);
00074
00078     void on_checkBoxTemp1_stateChanged(int arg1);
00079
00084     void on_horizontalSlider_osX_valueChanged(int value);
00085
00086     void on_horizontalSlider_osX_2_valueChanged(int value);
00087
00088 private:
00089     Ui::wyswietlanie *ui;
00090     wyswietlanie *oknoWyswietlania;
00091 };
00092
00093 #endif // WYSWIETLANIE_H
```

