

serial-port-assistant

Generated by Doxygen 1.11.0



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 MainWindow Class Reference	7
4.1.1 Constructor & Destructor Documentation	7
4.1.1.1 MainWindow()	7
4.1.1.2 ~MainWindow()	7
4.2 SerialConnection Class Reference	8
4.2.1 Constructor & Destructor Documentation	8
4.2.1.1 SerialConnection()	8
4.2.1.2 ~SerialConnection()	8
4.2.2 Member Function Documentation	9
4.2.2.1 closeConnection()	9
4.2.2.2 dataReceived	9
4.2.2.3 dataWritten	9
4.2.2.4 readData()	9
4.2.2.5 writeData()	9
4.2.2.6 writeFile()	9
4.3 SerialMonitor Class Reference	10
4.3.1 Constructor & Destructor Documentation	11
4.3.1.1 SerialMonitor()	11
4.3.2 Member Data Documentation	11
4.3.2.1 BaudRateList	11
4.3.2.2 ClearButton	11
4.3.2.3 DataReceivedTextBox	11
4.3.2.4 DataToSendTextBox	11
4.3.2.5 DateTimeLabel	11
4.3.2.6 EstablishConnectionButton	11
4.3.2.7 LeftLayout	11
4.3.2.8 MainLayout	11
4.3.2.9 MiddleLayout	12
4.3.2.10 ReceiveAsFileButton	12
4.3.2.11 ReceiveAsTextButton	12
4.3.2.12 ReceiveControllLayout	12
4.3.2.13 RefreshSerialListButton	12
4.3.2.14 RightLayout	12

---

4.3.2.15 SendControlLayout . . . . .	12
4.3.2.16 SendDataButton . . . . .	12
4.3.2.17 SendFileButton . . . . .	12
4.3.2.18 SerialPortList . . . . .	12
4.3.2.19 TerminateConnectionButton . . . . .	12
<b>5 File Documentation</b>	<b>13</b>
5.1 src/include/MainWindow.h File Reference . . . . .	13
5.1.1 Detailed Description . . . . .	13
5.2 MainWindow.h . . . . .	14
5.3 src/include/SerialConnection.h File Reference . . . . .	14
5.3.1 Detailed Description . . . . .	15
5.3.2 Enumeration Type Documentation . . . . .	15
5.3.2.1 SerialConnectionState . . . . .	15
5.4 SerialConnection.h . . . . .	15
5.5 src/include/SerialMonitor.h File Reference . . . . .	16
5.5.1 Detailed Description . . . . .	16
5.6 SerialMonitor.h . . . . .	16
5.7 src/main.cpp File Reference . . . . .	17
5.7.1 Function Documentation . . . . .	17
5.7.1.1 main() . . . . .	17
5.8 src/MainWindow.cpp File Reference . . . . .	17
5.8.1 Detailed Description . . . . .	17
5.9 src/SerialConnection.cpp File Reference . . . . .	18
5.9.1 Macro Definition Documentation . . . . .	18
5.9.1.1 SERIAL_UNABLE_TO_WRITE . . . . .	18
5.10 src/SerialMonitor.cpp File Reference . . . . .	18
<b>Index</b>	<b>19</b>

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

QMainWindow	
MainWindow . . . . .	<a href="#">7</a>
QObject	
SerialConnection . . . . .	<a href="#">8</a>
QWidget	
SerialMonitor . . . . .	<a href="#">10</a>



## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">MainWindow</a>	7
<a href="#">SerialConnection</a>	8
<a href="#">SerialMonitor</a>	10





# Chapter 3

## File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

src/ <a href="#">main.cpp</a> . . . . .	17
src/ <a href="#">MainWindow.cpp</a> Implementations of <a href="#">MainWindow.h</a> . . . . .	17
src/ <a href="#">SerialConnection.cpp</a> . . . . .	18
src/ <a href="#">SerialMonitor.cpp</a> . . . . .	18
src/include/ <a href="#">MainWindow.h</a> This class is responsible for connecting the UI and the serial connection together . . . . .	13
src/include/ <a href="#">SerialConnection.h</a> This class is responsible for the serial connection . . . . .	14
src/include/ <a href="#">SerialMonitor.h</a> This class is all about the user interface . . . . .	16



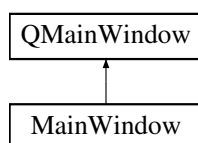
## Chapter 4

# Class Documentation

### 4.1 MainWindow Class Reference

```
#include <MainWindow.h>
```

Inheritance diagram for MainWindow:



#### Public Member Functions

- [MainWindow](#) (QWidget \*parent=nullptr)
- [~MainWindow](#) () override

#### 4.1.1 Constructor & Destructor Documentation

##### 4.1.1.1 MainWindow()

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

##### 4.1.1.2 ~MainWindow()

```
MainWindow::~MainWindow () [override]
```

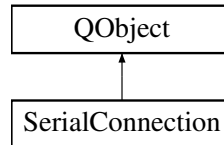
The documentation for this class was generated from the following files:

- [src/include/MainWindow.h](#)
- [src/MainWindow.cpp](#)

## 4.2 SerialConnection Class Reference

```
#include <SerialConnection.h>
```

Inheritance diagram for SerialConnection:



### Signals

- void [dataWritten](#) ()
- void [dataReceived](#) ()

### Public Member Functions

- [SerialConnection](#) (const QString &SerialPortName, QSerialPort::BaudRate BaudRate=QSerialPort::↔ Baud9600)
- [~SerialConnection](#) () override
- [SerialConnectionState writeData](#) (const QByteArray &data) const
- [SerialConnectionState writeFile](#) (const QString &NewFileName) const
- [SerialConnectionState closeConnection](#) () const
- QByteArray [readData](#) () const

### 4.2.1 Constructor & Destructor Documentation

#### 4.2.1.1 SerialConnection()

```
SerialConnection::SerialConnection (
    const QString & SerialPortName,
    QSerialPort::BaudRate BaudRate = QSerialPort::Baud9600) [explicit]
```

#### Parameters

<i>SerialPortName</i>	the name of the serial port to open, which is shown in the ComboBox widget in user interface and chosed by the user.
<i>BaudRate</i>	as the baudrate for the serial connection , chosed by the user in user interface , default at 9600

#### Returns

a [SerialConnection](#) object

#### 4.2.1.2 ~SerialConnection()

```
SerialConnection::~~SerialConnection () [override]
```

## 4.2.2 Member Function Documentation

### 4.2.2.1 closeConnection()

```
SerialConnectionState SerialConnection::closeConnection () const [nodiscard]
```

#### Returns

the Error Code. SerialPort is closed only when last operation is finished(which is nothing to read and nothing to write)

### 4.2.2.2 dataReceived

```
void SerialConnection::dataReceived () [signal]
```

### 4.2.2.3 dataWritten

```
void SerialConnection::dataWritten () [signal]
```

### 4.2.2.4 readData()

```
QByteArray SerialConnection::readData () const [nodiscard]
```

#### Returns

data readed from serialport. these data will be passed to [MainWindow](#),user is responsible to decide whether to save it as file or show in the TextEdit widget.

### 4.2.2.5 writeData()

```
SerialConnectionState SerialConnection::writeData (  
    const QByteArray & DataToWrite) const [nodiscard]
```

#### Parameters

<i>DataToWrite</i>	as the data to be written, which is inputted in the TextEdit widget in user interface
--------------------	---

#### Returns

the Error Code of the operation,used to show warning MessageBox when write operation failed

### 4.2.2.6 writeFile()

```
SerialConnectionState SerialConnection::writeFile (  
    const QString & NameOfFileToWrite) const [nodiscard]
```

**Parameters**

<i>NameOfFileToWrite</i>	is the filename of the file to send.
--------------------------	--------------------------------------

**Returns**

The error code of the operation,used to show warning MessageBox when write operation failed

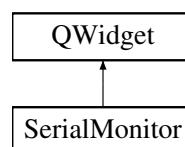
The documentation for this class was generated from the following files:

- src/include/[SerialConnection.h](#)
- src/[SerialConnection.cpp](#)

## 4.3 SerialMonitor Class Reference

```
#include <SerialMonitor.h>
```

Inheritance diagram for SerialMonitor:

**Public Member Functions**

- [SerialMonitor](#) (QWidget \*parent=nullptr)

**Public Attributes**

- QHBoxLayout \* [MainLayout](#)
- QVBoxLayout \* [LeftLayout](#)
- QVBoxLayout \* [MiddleLayout](#)
- QVBoxLayout \* [RightLayout](#)
- QHBoxLayout \* [ReceiveControlLayout](#)
- QHBoxLayout \* [SendControlLayout](#)
- QTextEdit \* [DataToSendTextBox](#)
- QTextEdit \* [DataReceivedTextBox](#)
- QPushButton \* [ClearButton](#)
- QPushButton \* [SendDataButton](#)
- QPushButton \* [SendFileButton](#)
- QPushButton \* [RefreshSerialListButton](#)
- QPushButton \* [EstablishConnectionButton](#)
- QPushButton \* [TerminateConnectionButton](#)
- QPushButton \* [ReceiveAsTextButton](#)
- QPushButton \* [ReceiveAsFileButton](#)
- QComboBox \* [BaudRateList](#)
- QComboBox \* [SerialPortList](#)
- QLabel \* [DateTimeLabel](#)

## 4.3.1 Constructor & Destructor Documentation

### 4.3.1.1 SerialMonitor()

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

## 4.3.2 Member Data Documentation

### 4.3.2.1 BaudRateList

```
QComboBox* SerialMonitor::BaudRateList
```

### 4.3.2.2 ClearButton

```
QPushButton* SerialMonitor::ClearButton
```

### 4.3.2.3 DataReceivedTextBox

```
QTextEdit* SerialMonitor::DataReceivedTextBox
```

### 4.3.2.4 DataToSendTextBox

```
QTextEdit* SerialMonitor::DataToSendTextBox
```

### 4.3.2.5 DateTimeLabel

```
QLabel* SerialMonitor::DateTimeLabel
```

### 4.3.2.6 EstablishConnectionButton

```
QPushButton* SerialMonitor::EstablishConnectionButton
```

### 4.3.2.7 LeftLayout

```
QVBoxLayout* SerialMonitor::LeftLayout
```

### 4.3.2.8 MainLayout

```
QHBoxLayout* SerialMonitor::MainLayout
```

#### 4.3.2.9 MiddleLayout

`QVBoxLayout* SerialMonitor::MiddleLayout`

#### 4.3.2.10 ReceiveAsFileButton

`QPushButton* SerialMonitor::ReceiveAsFileButton`

#### 4.3.2.11 ReceiveAsTextButton

`QPushButton* SerialMonitor::ReceiveAsTextButton`

#### 4.3.2.12 ReceiveControlLayout

`QHBoxLayout* SerialMonitor::ReceiveControlLayout`

#### 4.3.2.13 RefreshSerialListButton

`QPushButton* SerialMonitor::RefreshSerialListButton`

#### 4.3.2.14 RightLayout

`QVBoxLayout* SerialMonitor::RightLayout`

#### 4.3.2.15 SendControlLayout

`QHBoxLayout* SerialMonitor::SendControlLayout`

#### 4.3.2.16 SendDataButton

`QPushButton* SerialMonitor::SendDataButton`

#### 4.3.2.17 SendFileButton

`QPushButton* SerialMonitor::SendFileButton`

#### 4.3.2.18 SerialPortList

`QComboBox* SerialMonitor::SerialPortList`

#### 4.3.2.19 TerminateConnectionButton

`QPushButton* SerialMonitor::TerminateConnectionButton`

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

- [src/include/SerialMonitor.h](#)
- [src/SerialMonitor.cpp](#)



# Chapter 5

## File Documentation

### 5.1 src/include/MainWindow.h File Reference

This class is responsible for connecting the UI and the serial connection together.

```
#include "SerialConnection.h"
#include "SerialMonitor.h"
#include <QMainWindow>
#include <QStandardItemModel>
#include <vector>
```

#### Classes

- class [MainWindow](#)

#### 5.1.1 Detailed Description

This class is responsible for connecting the UI and the serial connection together.

#### Note

including call the functions in [SerialConnection](#) when buttons are clicked, insert BaudRate into BaudRateList, pdate SerialPortList constantly, and show the data received from serial port in the TextEdit widget, or save it as file.

## 5.2 MainWindow.h

[Go to the documentation of this file.](#)

```

00001
00011 #ifndef MAINWINDOW_H
00012 #define MAINWINDOW_H
00013
00014 #include "SerialConnection.h"
00015 #include "SerialMonitor.h"
00016 #include <QMainWindow>
00017 #include <QStandardItemModel>
00018 #include <vector>
00019
00020 class MainWindow : public QMainWindow {
00021     Q_OBJECT
00022
00023 public:
00024     explicit MainWindow(QWidget *parent = nullptr);
00025     ~MainWindow() override;
00026
00027 private:
00028     SerialMonitor *UI = nullptr;
00029     SerialConnection *UiHandledSerialConnection = nullptr;
00030     QStandardItemModel *SerialPortModel = nullptr;
00031     QStandardItemModel *BaudRateModel = nullptr;
00032     std::vector<QSerialPort::BaudRate> BaudRateList = {QSerialPort::Baud9600, QSerialPort::Baud19200,
00033                                                         QSerialPort::Baud38400, QSerialPort::Baud57600,
00034                                                         QSerialPort::Baud115200};
00035
00036 private slots:
00037     void connectSlots();
00038     void createConnection();
00039     void terminateConnection();
00040     void sendText();
00041     void sendFile();
00042     void receiveText();
00043     void receiveFile();
00044     void initBaudRate();
00045     void clearAll();
00046     void getAvaliableSerialPorts();
00047     void dataWrittenHint();
00048 };
00049
00050 #endif // MAINWINDOW_H

```

## 5.3 src/include/SerialConnection.h File Reference

This class is responsible for the serial connection.

```

#include <QSerialPort>
#include <QObject>

```

### Classes

- class [SerialConnection](#)

### Enumerations

- enum class [SerialConnectionState](#) {  
[FileNotFound](#) , [FileReadError](#) , [LastSerialOperationNotCompleted](#) , [NoError](#) ,  
[NothingToBeReaded](#) , [SerialPortNotOpened](#) }

### 5.3.1 Detailed Description

This class is responsible for the serial connection.

#### Note

This class is a wrapper of QSerialPort, including sending data, sending file, receiving data, also there's a Errorcode class named SerialConnectionState ,which can be passed to [MainWindow](#) to do some specific operations, such as pop up a window to warn users that serialport is not opened.

### 5.3.2 Enumeration Type Documentation

#### 5.3.2.1 SerialConnectionState

```
enum class SerialConnectionState [strong]
```

#### Enumerator

FileNotFound	
FileReadError	
LastSerialOperationNotCompleted	
NoError	
NothingToBeReaded	
SerialPortNotOpened	

## 5.4 SerialConnection.h

[Go to the documentation of this file.](#)

```
00001
00009 #ifndef SERIAL_CONNECTION_H
00010 #define SERIAL_CONNECTION_H
00011 #include <QSerialPort>
00012 #include <QObject>
00013
00014 enum class SerialConnectionState {
00015     FileNotFound,
00016     FileReadError,
00017     LastSerialOperationNotCompleted,
00018     NoError,
00019     NothingToBeReaded,
00020     SerialPortNotOpened
00021 };
00022
00023 class SerialConnection : public QObject{
00024     Q_OBJECT
00025 public:
00026     explicit SerialConnection(const QString& SerialPortName, QSerialPort::BaudRate
00027         BaudRate=QSerialPort::Baud9600);
00028     ~SerialConnection() override;
00029     [[nodiscard]] SerialConnectionState writeData(const QByteArray &data) const;
00030     [[nodiscard]] SerialConnectionState writeFile(const QString& NewFileName) const;
00031     [[nodiscard]] SerialConnectionState closeConnection() const;
00032     [[nodiscard]] QByteArray readData() const;
00033
00034     signals:
00035     void dataWritten();
00036     void dataReceived();
00037
00038 private:
00039     QSerialPort *SerialPort{};
00040     QSerialPort::BaudRate CurrentConnectedSerialPortBaudRate;
00041     QString CurrentConnectedSerialPortName;
00042
00043 private slots:
00044     void onBytesWritten();
00045 };
00046 #endif //SERIAL_CONNECTION_H
```

## 5.5 src/include/SerialMonitor.h File Reference

This class is all about the user interface.

```
#include <QApplication>
#include <QCheckBox>
#include <QComboBox>
#include <QDateTime>
#include <QGridLayout>
#include <QHBoxLayout>
#include <QLabel>
#include <QPushButton>
#include <QTextEdit>
#include <QTimer>
#include <QVBoxLayout>
#include <QWidget>
```

### Classes

- class [SerialMonitor](#)

### 5.5.1 Detailed Description

This class is all about the user interface.

#### Note

This class is nothing but some widgets, including buttons, textedit, combobox, etc. this class is NOT responsible for integration between UI and [SerialConnection](#), which is [MainWindow](#)'s job.

## 5.6 SerialMonitor.h

[Go to the documentation of this file.](#)

```
00001
00011 #ifndef UIWINDOW_H
00012 #define UIWINDOW_H
00013
00014 #include <QApplication>
00015 #include <QCheckBox>
00016 #include <QComboBox>
00017 #include <QDateTime>
00018 #include <QGridLayout>
00019 #include <QHBoxLayout>
00020 #include <QLabel>
00021 #include <QPushButton>
00022 #include <QTextEdit>
00023 #include <QTimer>
00024 #include <QVBoxLayout>
00025 #include <QWidget>
00026
00027 class SerialMonitor : public QWidget {
00028     public:
00029         explicit SerialMonitor(QWidget *parent = nullptr);
00030
00031     public:
00032         QHBoxLayout      *MainLayout;
00033         QVBoxLayout      *LeftLayout;
00034         QVBoxLayout      *MiddleLayout;
00035         QVBoxLayout      *RightLayout;
00036         QHBoxLayout      *ReceiveControlLayout;
```

```

00037     QHBoxLayout      *SendControllLayout;
00038     QTextEdit         *DataToSendTextBox;
00039     QTextEdit         *DataReceivedTextBox;
00040     QPushButton       *ClearButton;
00041     QPushButton       *SendDataButton;
00042     QPushButton       *SendFileButton;
00043     QPushButton       *RefreshSerialListButton;
00044     QPushButton       *EstablishConnectionButton;
00045     QPushButton       *TerminateConnectionButton;
00046     QPushButton       *ReceiveAsTextButton;
00047     QPushButton       *ReceiveAsFileButton;
00048     QComboBox         *BaudRateList;
00049     QComboBox         *SerialPortList;
00050     QLabel            *DateTimeLabel;
00051 };
00052
00053 #endif // UIWINDOW_H

```

## 5.7 src/main.cpp File Reference

```
#include "include/MainWindow.h"
```

### Functions

- int [main](#) (int argc, char \*argv[])

### 5.7.1 Function Documentation

#### 5.7.1.1 main()

```

int main (
    int argc,
    char * argv[])

```

## 5.8 src/MainWindow.cpp File Reference

Implementations of [MainWindow.h](#).

```

#include "include/MainWindow.h"
#include "include/SerialConnection.h"
#include <QFileDialog>
#include <QMessageBox>
#include <QSerialPortInfo>
#include <QStandardItemModel>
#include <QTimer>

```

### 5.8.1 Detailed Description

Implementations of [MainWindow.h](#).

#### Note

This file is about the integration of UI and [SerialConnection](#), which means when you click a button, the [SerialConnection](#) or the UI will react to the event.

## 5.9 src/SerialConnection.cpp File Reference

```
#include "include/SerialConnection.h"  
#include <QFile>  
#include <QMessageBox>
```

### Macros

- #define [SERIAL\\_UNABLE\\_TO\\_WRITE](#) -1

### 5.9.1 Macro Definition Documentation

#### 5.9.1.1 SERIAL\_UNABLE\_TO\_WRITE

```
#define SERIAL_UNABLE_TO_WRITE -1
```

## 5.10 src/SerialMonitor.cpp File Reference

```
#include "include/SerialMonitor.h"
```

# Index

- ~MainWindow
  - MainWindow, [7](#)
- ~SerialConnection
  - SerialConnection, [8](#)
- BaudRateList
  - SerialMonitor, [11](#)
- ClearButton
  - SerialMonitor, [11](#)
- closeConnection
  - SerialConnection, [9](#)
- dataReceived
  - SerialConnection, [9](#)
- DataReceivedTextBox
  - SerialMonitor, [11](#)
- DataToSendTextBox
  - SerialMonitor, [11](#)
- dataWritten
  - SerialConnection, [9](#)
- DateTimeLabel
  - SerialMonitor, [11](#)
- EstablishConnectionButton
  - SerialMonitor, [11](#)
- FileNotFound
  - SerialConnection.h, [15](#)
- FileReadError
  - SerialConnection.h, [15](#)
- LastSerialOperationNotCompleted
  - SerialConnection.h, [15](#)
- LeftLayout
  - SerialMonitor, [11](#)
- main
  - main.cpp, [17](#)
- main.cpp
  - main, [17](#)
- MainLayout
  - SerialMonitor, [11](#)
- MainWindow, [7](#)
  - ~MainWindow, [7](#)
  - MainWindow, [7](#)
- MiddleLayout
  - SerialMonitor, [11](#)
- NoError
  - SerialConnection.h, [15](#)
- NothingToBeReaded
  - SerialConnection.h, [15](#)
- readData
  - SerialConnection, [9](#)
- ReceiveAsFileButton
  - SerialMonitor, [12](#)
- ReceiveAsTextButton
  - SerialMonitor, [12](#)
- ReceiveControlLayout
  - SerialMonitor, [12](#)
- RefreshSerialListButton
  - SerialMonitor, [12](#)
- RightLayout
  - SerialMonitor, [12](#)
- SendControlLayout
  - SerialMonitor, [12](#)
- SendDataButton
  - SerialMonitor, [12](#)
- SendFileButton
  - SerialMonitor, [12](#)
- SERIAL\_UNABLE\_TO\_WRITE
  - SerialConnection.cpp, [18](#)
- SerialConnection, [8](#)
  - ~SerialConnection, [8](#)
  - closeConnection, [9](#)
  - dataReceived, [9](#)
  - dataWritten, [9](#)
  - readData, [9](#)
  - SerialConnection, [8](#)
  - writeData, [9](#)
  - writeFile, [9](#)
- SerialConnection.cpp
  - SERIAL\_UNABLE\_TO\_WRITE, [18](#)
- SerialConnection.h
  - FileNotFound, [15](#)
  - FileReadError, [15](#)
  - LastSerialOperationNotCompleted, [15](#)
  - NoError, [15](#)
  - NothingToBeReaded, [15](#)
  - SerialConnectionState, [15](#)
  - SerialPortNotOpened, [15](#)
- SerialConnectionState
  - SerialConnection.h, [15](#)
- SerialMonitor, [10](#)
  - BaudRateList, [11](#)
  - ClearButton, [11](#)
  - DataReceivedTextBox, [11](#)
  - DataToSendTextBox, [11](#)

- DateTimeLabel, [11](#)
- EstablishConnectionButton, [11](#)
- LeftLayout, [11](#)
- MainLayout, [11](#)
- MiddleLayout, [11](#)
- ReceiveAsFileButton, [12](#)
- ReceiveAsTextButton, [12](#)
- ReceiveControlLayout, [12](#)
- RefreshSerialListButton, [12](#)
- RightLayout, [12](#)
- SendControlLayout, [12](#)
- SendDataButton, [12](#)
- SendFileButton, [12](#)
- SerialMonitor, [11](#)
- SerialPortList, [12](#)
- TerminateConnectionButton, [12](#)
- SerialPortList
  - SerialMonitor, [12](#)
- SerialPortNotOpened
  - SerialConnection.h, [15](#)
- src/include/MainWindow.h, [13](#), [14](#)
- src/include/SerialConnection.h, [14](#), [15](#)
- src/include/SerialMonitor.h, [16](#)
- src/main.cpp, [17](#)
- src/MainWindow.cpp, [17](#)
- src/SerialConnection.cpp, [18](#)
- src/SerialMonitor.cpp, [18](#)
- TerminateConnectionButton
  - SerialMonitor, [12](#)
- writeData
  - SerialConnection, [9](#)
- writeFile
  - SerialConnection, [9](#)