

# Multimeter GUI

Generated by Doxygen 1.8.6

Sat May 7 2016 20:57:47



# Contents

<b>1</b>	<b>Main Page</b>	<b>1</b>
<b>2</b>	<b>multimeterGUI</b>	<b>3</b>
<b>3</b>	<b>Todo List</b>	<b>5</b>
<b>4</b>	<b>Namespace Index</b>	<b>7</b>
4.1	Namespace List . . . . .	7
<b>5</b>	<b>Hierarchical Index</b>	<b>9</b>
5.1	Class Hierarchy . . . . .	9
<b>6</b>	<b>Class Index</b>	<b>11</b>
6.1	Class List . . . . .	11
<b>7</b>	<b>File Index</b>	<b>13</b>
7.1	File List . . . . .	13
<b>8</b>	<b>Namespace Documentation</b>	<b>15</b>
8.1	Ui Namespace Reference . . . . .	15
<b>9</b>	<b>Class Documentation</b>	<b>17</b>
9.1	Flags Struct Reference . . . . .	17
9.1.1	Detailed Description . . . . .	18
9.1.2	Member Function Documentation . . . . .	18
9.1.2.1	operator!= . . . . .	18
9.1.2.2	operator== . . . . .	18
9.1.3	Member Data Documentation . . . . .	18
9.1.3.1	A . . . . .	18
9.1.3.2	Auto . . . . .	18
9.1.3.3	Bat . . . . .	18
9.1.3.4	Beep . . . . .	18
9.1.3.5	checksum . . . . .	18
9.1.3.6	continuity . . . . .	18

9.1.3.7	dash	18
9.1.3.8	dBm	18
9.1.3.9	Diode	18
9.1.3.10	DP	18
9.1.3.11	F	18
9.1.3.12	hFE	18
9.1.3.13	Hold	18
9.1.3.14	Hz	18
9.1.3.15	k	19
9.1.3.16	M	19
9.1.3.17	m	19
9.1.3.18	MAX	19
9.1.3.19	MIN	19
9.1.3.20	n	19
9.1.3.21	Ohms	19
9.1.3.22	percent	19
9.1.3.23	REL	19
9.1.3.24	RS232	19
9.1.3.25	s	19
9.1.3.26	tilde	19
9.1.3.27	u	19
9.1.3.28	V	19
9.2	LCD Class Reference	19
9.2.1	Detailed Description	20
9.2.2	Constructor & Destructor Documentation	20
9.2.2.1	LCD	20
9.2.3	Member Function Documentation	21
9.2.3.1	paintEvent	21
9.2.4	Member Data Documentation	21
9.2.4.1	data	21
9.2.4.2	lbl	21
9.3	MainWindow Class Reference	21
9.3.1	Detailed Description	23
9.3.2	Constructor & Destructor Documentation	23
9.3.2.1	MainWindow	23
9.3.2.2	~MainWindow	24
9.3.3	Member Function Documentation	24
9.3.3.1	addData	24
9.3.3.2	on_connectButton_clicked	25
9.3.3.3	on_disconnectButton_clicked	25

9.3.3.4	<a href="#">resetData</a>	26
9.3.4	<a href="#">Member Data Documentation</a>	26
9.3.4.1	<a href="#">counter</a>	26
9.3.4.2	<a href="#">graph</a>	26
9.3.4.3	<a href="#">label</a>	26
9.3.4.4	<a href="#">lcd</a>	26
9.3.4.5	<a href="#">maxData</a>	27
9.3.4.6	<a href="#">minData</a>	27
9.3.4.7	<a href="#">newData</a>	27
9.3.4.8	<a href="#">portPtr</a>	27
9.3.4.9	<a href="#">rawdata</a>	27
9.3.4.10	<a href="#">rData</a>	27
9.3.4.11	<a href="#">scene</a>	27
9.3.4.12	<a href="#">storeData</a>	27
9.3.4.13	<a href="#">timeMark</a>	27
9.3.4.14	<a href="#">timeRunning</a>	27
9.3.4.15	<a href="#">tmp</a>	27
9.3.4.16	<a href="#">ui</a>	27
9.4	<a href="#">Ui::MainWindow Class Reference</a>	27
9.5	<a href="#">plotGraph Class Reference</a>	28
9.5.1	<a href="#">Detailed Description</a>	30
9.5.2	<a href="#">Constructor &amp; Destructor Documentation</a>	30
9.5.2.1	<a href="#">plotGraph</a>	30
9.5.2.2	<a href="#">plotGraph</a>	30
9.5.3	<a href="#">Member Function Documentation</a>	30
9.5.3.1	<a href="#">boundingRect</a>	30
9.5.3.2	<a href="#">labelXaxis</a>	30
9.5.3.3	<a href="#">labelYaxis</a>	31
9.5.3.4	<a href="#">linkData</a>	31
9.5.3.5	<a href="#">paint</a>	32
9.5.3.6	<a href="#">paintAxis</a>	33
9.5.3.7	<a href="#">plotData</a>	34
9.5.3.8	<a href="#">real2Coord</a>	34
9.5.3.9	<a href="#">setScene</a>	35
9.5.3.10	<a href="#">setUnit</a>	35
9.5.3.11	<a href="#">setXaxis</a>	35
9.5.3.12	<a href="#">setXsticks</a>	35
9.5.3.13	<a href="#">setYaxis</a>	36
9.5.3.14	<a href="#">setYsticks</a>	36
9.5.4	<a href="#">Member Data Documentation</a>	36

9.5.4.1	<a href="#">axisMargin</a>	36
9.5.4.2	<a href="#">bRect</a>	36
9.5.4.3	<a href="#">data</a>	36
9.5.4.4	<a href="#">nx</a>	36
9.5.4.5	<a href="#">ny</a>	36
9.5.4.6	<a href="#">origin</a>	36
9.5.4.7	<a href="#">rightX</a>	36
9.5.4.8	<a href="#">scene</a>	36
9.5.4.9	<a href="#">unit</a>	37
9.5.4.10	<a href="#">upperY</a>	37
9.5.4.11	<a href="#">xmax</a>	37
9.5.4.12	<a href="#">xmin</a>	37
9.5.4.13	<a href="#">ymax</a>	37
9.5.4.14	<a href="#">ymin</a>	37
9.6	<a href="#">qt_meta_stringdata_LCD_t Struct Reference</a>	37
9.6.1	<a href="#">Member Data Documentation</a>	37
9.6.1.1	<a href="#">data</a>	37
9.6.1.2	<a href="#">stringdata0</a>	37
9.7	<a href="#">qt_meta_stringdata_MainWindow_t Struct Reference</a>	37
9.7.1	<a href="#">Member Data Documentation</a>	37
9.7.1.1	<a href="#">data</a>	37
9.7.1.2	<a href="#">stringdata0</a>	37
9.8	<a href="#">qt_meta_stringdata_plotGraph_t Struct Reference</a>	38
9.8.1	<a href="#">Member Data Documentation</a>	38
9.8.1.1	<a href="#">data</a>	38
9.8.1.2	<a href="#">stringdata0</a>	38
9.9	<a href="#">qt_meta_stringdata_RS22812_t Struct Reference</a>	38
9.9.1	<a href="#">Member Data Documentation</a>	38
9.9.1.1	<a href="#">data</a>	38
9.9.1.2	<a href="#">stringdata0</a>	38
9.10	<a href="#">qt_meta_stringdata_SerialPort_t Struct Reference</a>	38
9.10.1	<a href="#">Member Data Documentation</a>	38
9.10.1.1	<a href="#">data</a>	38
9.10.1.2	<a href="#">stringdata0</a>	38
9.11	<a href="#">RS22812 Class Reference</a>	39
9.11.1	<a href="#">Detailed Description</a>	40
9.11.2	<a href="#">Constructor &amp; Destructor Documentation</a>	41
9.11.2.1	<a href="#">RS22812</a>	41
9.11.3	<a href="#">Member Function Documentation</a>	41
9.11.3.1	<a href="#">byte2Digit</a>	41

9.11.3.2	getDigitString	42
9.11.3.3	getFlags	42
9.11.3.4	getMode	43
9.11.3.5	getVal	43
9.11.3.6	modeChanged	43
9.11.3.7	newData	44
9.11.3.8	newMode	44
9.11.3.9	newValue	44
9.11.3.10	resetFlags	45
9.11.4	Member Data Documentation	45
9.11.4.1	digits	45
9.11.4.2	flags	45
9.11.4.3	mode	45
9.11.4.4	oldflags	45
9.12	SerialPort Class Reference	46
9.12.1	Detailed Description	47
9.12.2	Constructor & Destructor Documentation	47
9.12.2.1	SerialPort	47
9.12.2.2	~SerialPort	47
9.12.3	Member Function Documentation	48
9.12.3.1	closePort	48
9.12.3.2	listPorts	48
9.12.3.3	openPort	49
9.12.3.4	readPort	50
9.12.3.5	ready	50
9.12.3.6	readyRead	51
9.12.4	Member Data Documentation	51
9.12.4.1	activePort	51
9.12.4.2	BAUDRATE	51
9.12.4.3	buffer	51
9.12.4.4	DATABITS	51
9.12.4.5	isOpen	51
9.12.4.6	MODE	51
9.12.4.7	PARITY	51
9.12.4.8	ports	51
9.12.4.9	readConnect	51
9.12.4.10	STOPBITS	52
9.13	Ui_MainWindow Class Reference	52
9.13.1	Member Function Documentation	53
9.13.1.1	retranslateUi	53

9.13.1.2	setupUi	53
9.13.2	Member Data Documentation	53
9.13.2.1	centralWidget	53
9.13.2.2	comboBoxPort	53
9.13.2.3	connectButton	53
9.13.2.4	disconnectButton	53
9.13.2.5	graphPlot	53
9.13.2.6	gridLayout	53
9.13.2.7	horizontalLayout	53
9.13.2.8	labelPort	54
9.13.2.9	mainToolBar	54
9.13.2.10	menuBar	54
9.13.2.11	statusBar	54
9.13.2.12	toolBar	54
9.13.2.13	verticalLayout	54
9.13.2.14	verticalLayout_2	54
<b>10</b>	<b>File Documentation</b>	<b>55</b>
10.1	build-multimeterGUI-Desktop-Debug/moc_lcd.cpp File Reference	55
10.1.1	Macro Definition Documentation	56
10.1.1.1	QT_MOC_LITERAL	56
10.1.2	Variable Documentation	56
10.1.2.1	qt_meta_data_LCD	56
10.1.2.2	qt_meta_stringdata_LCD	56
10.2	build-multimeterGUI-Desktop-Debug/moc_mainwindow.cpp File Reference	56
10.2.1	Macro Definition Documentation	57
10.2.1.1	QT_MOC_LITERAL	57
10.2.2	Variable Documentation	57
10.2.2.1	qt_meta_data_MainWindow	57
10.2.2.2	qt_meta_stringdata_MainWindow	57
10.3	build-multimeterGUI-Desktop-Debug/moc_plotgraph.cpp File Reference	58
10.3.1	Macro Definition Documentation	58
10.3.1.1	QT_MOC_LITERAL	58
10.3.2	Variable Documentation	59
10.3.2.1	qt_meta_data_plotGraph	59
10.3.2.2	qt_meta_stringdata_plotGraph	59
10.4	build-multimeterGUI-Desktop-Debug/moc_rs22812.cpp File Reference	59
10.4.1	Macro Definition Documentation	60
10.4.1.1	QT_MOC_LITERAL	60
10.4.2	Variable Documentation	60



10.4.2.1	qt_meta_data_RS22812	60
10.4.2.2	qt_meta_stringdata_RS22812	61
10.5	build-multimeterGUI-Desktop-Debug/moc_serialport.cpp File Reference	61
10.5.1	Macro Definition Documentation	61
10.5.1.1	QT_MOC_LITERAL	61
10.5.2	Variable Documentation	62
10.5.2.1	qt_meta_data_SerialPort	62
10.5.2.2	qt_meta_stringdata_SerialPort	62
10.6	build-multimeterGUI-Desktop-Debug/ui_mainwindow.h File Reference	63
10.7	lcd.cpp File Reference	63
10.8	lcd.h File Reference	64
10.9	main.cpp File Reference	65
10.9.1	Function Documentation	65
10.9.1.1	main	65
10.10	mainwindow.cpp File Reference	65
10.11	mainwindow.h File Reference	66
10.12	plotgraph.cpp File Reference	67
10.13	plotgraph.h File Reference	67
10.14	README.md File Reference	68
10.15	rs22812.cpp File Reference	68
10.16	rs22812.h File Reference	68
10.17	serialport.cpp File Reference	69
10.18	serialport.h File Reference	69
<b>Index</b>		<b>71</b>



# Chapter 1

## Main Page

Multimeter GUI GUI for the RS-232 mode of the Radio Shack 22-812. Copyright (C) 2016 FJ Salguero

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

### Author

F.J. Salguero.



## Chapter 2

# multimeterGUI

GUI for the Radio Shack 22-812 multimeter when used in RS232 mode.

This is a QT GUI to interface the RS 22-812. It has been developed with QT5 in OpenSuse Leap.

This code is "as-is". No warranty or responsibility is assumed by the author. You can use your code for any non-commercial purposes, and modify it as you wish with the only condition of licensing it with the same conditions. If you find it useful, make any change or have any suggestion, I will appreciate your input.



## Chapter 3

### Todo List

**Member `main` (int argc, char \*argv[])**

Add display controls.

**Member `MainWindow::MainWindow` (QWidget \*parent=0)**

: Temporary. It has to be set automatically.





## Chapter 4

# Namespace Index

### 4.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Ui .....	15
----------	----



## Chapter 5

# Hierarchical Index

### 5.1 Class Hierarchy

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

Flags . . . . .	17
QGraphicsItem	
plotGraph . . . . .	28
QLabel	
LCD . . . . .	19
QMainWindow	
MainWindow . . . . .	21
QObject	
RS22812 . . . . .	39
SerialPort . . . . .	46
qt_meta_stringdata_LCD_t . . . . .	37
qt_meta_stringdata_MainWindow_t . . . . .	37
qt_meta_stringdata_plotGraph_t . . . . .	38
qt_meta_stringdata_RS22812_t . . . . .	38
qt_meta_stringdata_SerialPort_t . . . . .	38
Ui_MainWindow . . . . .	52
Ui::MainWindow . . . . .	27



## Chapter 6

# Class Index

### 6.1 Class List

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

<a href="#">Flags</a>	Definition of custom data type . . . . .	17
<a href="#">LCD</a>	Displays the numerical value read . . . . .	19
<a href="#">MainWindow</a>	The <a href="#">MainWindow</a> class . . . . .	21
<a href="#">Ui::MainWindow</a>	. . . . .	27
<a href="#">plotGraph</a>	The <a href="#">plotGraph</a> class . . . . .	28
<a href="#">qt_meta_stringdata_LCD_t</a>	. . . . .	37
<a href="#">qt_meta_stringdata_MainWindow_t</a>	. . . . .	37
<a href="#">qt_meta_stringdata_plotGraph_t</a>	. . . . .	38
<a href="#">qt_meta_stringdata_RS22812_t</a>	. . . . .	38
<a href="#">qt_meta_stringdata_SerialPort_t</a>	. . . . .	38
<a href="#">RS22812</a>	Decoding of the data sent by the Radio Shack 22-812 . . . . .	39
<a href="#">SerialPort</a>	Class to manage the communication with a serial port . . . . .	46
<a href="#">Ui_MainWindow</a>	. . . . .	52



## Chapter 7

# File Index

### 7.1 File List

Here is a list of all files with brief descriptions:

<a href="#">lcd.cpp</a>	63
<a href="#">lcd.h</a>	64
<a href="#">main.cpp</a>	65
<a href="#">mainwindow.cpp</a>	65
<a href="#">mainwindow.h</a>	66
<a href="#">plotgraph.cpp</a>	67
<a href="#">plotgraph.h</a>	67
<a href="#">rs22812.cpp</a>	68
<a href="#">rs22812.h</a>	68
<a href="#">serialport.cpp</a>	69
<a href="#">serialport.h</a>	69
build-multimeterGUI-Desktop-Debug/ <a href="#">moc_lcd.cpp</a>	55
build-multimeterGUI-Desktop-Debug/ <a href="#">moc_mainwindow.cpp</a>	56
build-multimeterGUI-Desktop-Debug/ <a href="#">moc_plotgraph.cpp</a>	58
build-multimeterGUI-Desktop-Debug/ <a href="#">moc_rs22812.cpp</a>	59
build-multimeterGUI-Desktop-Debug/ <a href="#">moc_serialport.cpp</a>	61
build-multimeterGUI-Desktop-Debug/ <a href="#">ui_mainwindow.h</a>	63





## Chapter 8

# Namespace Documentation

### 8.1 Ui Namespace Reference

#### Classes

- class [MainWindow](#)



## Chapter 9

# Class Documentation

### 9.1 Flags Struct Reference

Definition of custom data type.

```
#include <rs22812.h>
```

#### Public Member Functions

- bool `operator==` (const `Flags` &f2) const  
*operator == Equality operator for `Flags` struct.*
- bool `operator!=` (const `Flags` &f2)

#### Public Attributes

- bool `Hz`
- bool `Ohms`
- bool `k`
- bool `M`
- bool `F`
- bool `A`
- bool `V`
- bool `m`
- bool `u`
- bool `n`
- bool `dBm`
- bool `s`
- bool `percent`
- bool `hFE`
- bool `REL`
- bool `MIN`
- bool `Beep`
- bool `Diode`
- bool `continuity`
- bool `Bat`
- bool `Hold`
- bool `dash`
- bool `tilde`
- bool `RS232`

- bool [Auto](#)
- bool [MAX](#)
- int [DP](#)
- bool [checksum](#)

### 9.1.1 Detailed Description

Definition of custom data type.

### 9.1.2 Member Function Documentation

9.1.2.1 bool `Flags::operator!=( const Flags & f2 )` [`inline`]

9.1.2.2 bool `Flags::operator==( const Flags & f2 ) const` [`inline`]

`operator ==` Equality operator for [Flags](#) struct.

Parameters

<i>f1</i>	
<i>f2</i>	

Returns

### 9.1.3 Member Data Documentation

9.1.3.1 bool `Flags::A`

9.1.3.2 bool `Flags::Auto`

9.1.3.3 bool `Flags::Bat`

9.1.3.4 bool `Flags::Beep`

9.1.3.5 bool `Flags::checksum`

9.1.3.6 bool `Flags::continuity`

9.1.3.7 bool `Flags::dash`

9.1.3.8 bool `Flags::dBm`

9.1.3.9 bool `Flags::Diode`

9.1.3.10 int `Flags::DP`

9.1.3.11 bool `Flags::F`

9.1.3.12 bool `Flags::hFE`

9.1.3.13 bool `Flags::Hold`

9.1.3.14 bool `Flags::Hz`

9.1.3.15 bool Flags::k

9.1.3.16 bool Flags::M

9.1.3.17 bool Flags::m

9.1.3.18 bool Flags::MAX

9.1.3.19 bool Flags::MIN

9.1.3.20 bool Flags::n

9.1.3.21 bool Flags::Ohms

9.1.3.22 bool Flags::percent

9.1.3.23 bool Flags::REL

9.1.3.24 bool Flags::RS232

9.1.3.25 bool Flags::s

9.1.3.26 bool Flags::tilde

9.1.3.27 bool Flags::u

9.1.3.28 bool Flags::V

The documentation for this struct was generated from the following file:

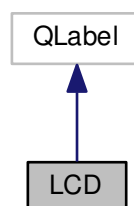
- [rs22812.h](#)

## 9.2 LCD Class Reference

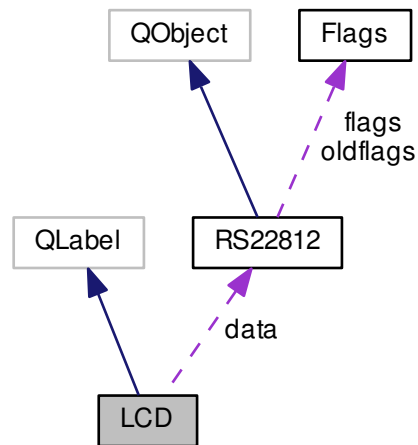
The [LCD](#) class displays the numerical value read.

```
#include <lcd.h>
```

Inheritance diagram for LCD:



Collaboration diagram for LCD:



## Public Member Functions

- **LCD** (const [RS22812](#) \**data*, QWidget \*parent=0)  
*Constructor.*

## Protected Member Functions

- void **paintEvent** (QPaintEvent \*event)  
*Paint event handler.*

## Private Attributes

- const QVector< QString > **lbl** ={"Auto","RS232","Hold","Rel","MAX","MIN","hFE","dBm","Cont","Diode","%","S"}
- const [RS22812](#) \* *data*

### 9.2.1 Detailed Description

The **LCD** class displays the numerical value read.

This class will show a representation of the multimeter's display showing the same values that are shown in the multimeter.

### 9.2.2 Constructor & Destructor Documentation

9.2.2.1 **LCD::LCD** ( const [RS22812](#) \* *data*, QWidget \* *parent* = 0 ) [explicit]

Constructor.

## Parameters

<i>data</i>	
<i>parent</i>	

## 9.2.3 Member Function Documentation

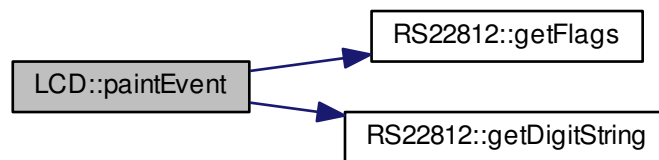
9.2.3.1 void LCD::paintEvent ( QPaintEvent \* *event* ) [protected]

Paint event handler.

## Parameters

<i>event</i>	It redraws the <a href="#">LCD</a> widget every time there is an update.
--------------	--

Here is the call graph for this function:



## 9.2.4 Member Data Documentation

## 9.2.4.1 const RS22812\* LCD::data [private]

## 9.2.4.2 const QVector&lt;QString&gt; LCD::lbl ={"Auto","RS232","Hold","Rel","MAX","MIN","hFE","dBm","Cont","Diode","%","S"} [private]

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

- [lcd.h](#)
- [lcd.cpp](#)

## 9.3 MainWindow Class Reference

The [MainWindow](#) class.

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





## Private Attributes

- [Ui::MainWindow](#) \* [ui](#)
- [SerialPort](#) \* [portPtr](#)
- [RS22812](#) \* [rawdata](#)
- [LCD](#) \* [lcd](#)
- [QLabel](#) \* [label](#)
- [plotGraph](#) \* [graph](#)
- [QGraphicsScene](#) \* [scene](#)
- [QVector< QPair< qint64, qreal > >](#) [storeData](#)
- [qint32](#) [counter](#) =0
- [qreal](#) [minData](#) =99999999
- [qreal](#) [maxData](#) =-99999999
- [QMetaObject::Connection](#) [newData](#)
- [QMetaObject::Connection](#) [rData](#)
- [QElapsedTimer](#) \* [timeMark](#)
- [bool](#) [timeRunning](#)
- [QVector< QPair< qreal, qreal > >](#) [tmp](#)

### 9.3.1 Detailed Description

The [MainWindow](#) class.

### 9.3.2 Constructor & Destructor Documentation

#### 9.3.2.1 [MainWindow::MainWindow](#) ( [QWidget](#) \* *parent* = 0 ) [[explicit](#)]

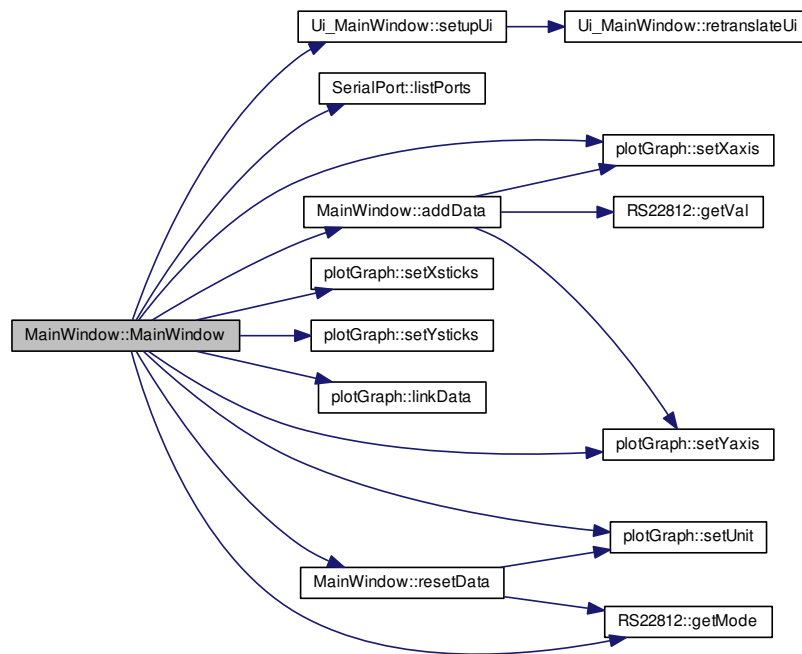
Constructor.

Parameters

<i>parent</i>	The main window constructor will, in addition to create the corresponding subwidgets, populate the list of available ports and connect signals with slots.
---------------	--

**Todo** : Temporary. It has to be set automatically.

Here is the call graph for this function:



### 9.3.2.2 MainWindow::~~MainWindow ( )

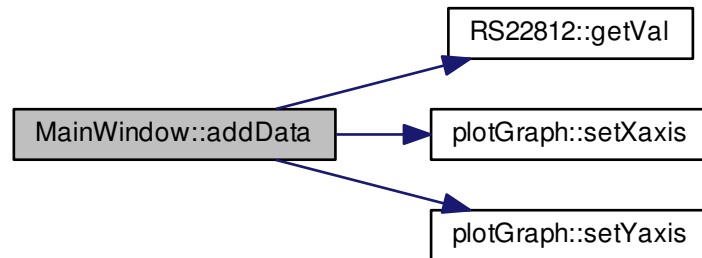
Destructor.

## 9.3.3 Member Function Documentation

### 9.3.3.1 void MainWindow::addData ( ) [private],[slot]

Add new data to the data set. This method will be called when new data has been read from the serial port, it will add the new value to the stored set of pairs (time,value) and update the graph.

Here is the call graph for this function:



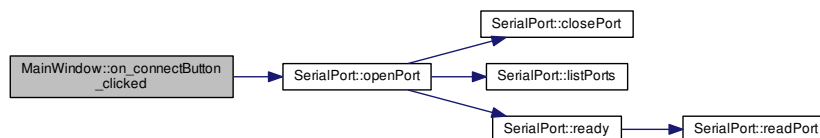
Here is the caller graph for this function:



#### 9.3.3.2 void MainWindow::on\_connectButton\_clicked ( ) [private],[slot]

Opens the selected port.

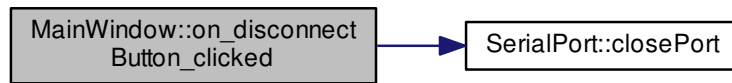
Here is the call graph for this function:



#### 9.3.3.3 void MainWindow::on\_disconnectButton\_clicked ( ) [private],[slot]

Disconnect from the current port.

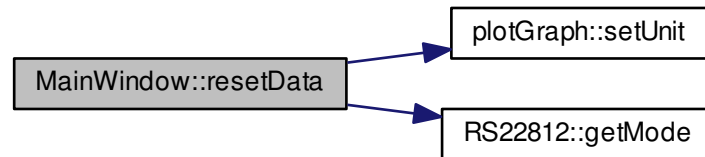
Here is the call graph for this function:



#### 9.3.3.4 void MainWindow::resetData ( ) [private],[slot]

Resets the data stored in memory. This method is called when the multimeter's mode changes, clearing all the data stored in memory.

Here is the call graph for this function:



Here is the caller graph for this function:



### 9.3.4 Member Data Documentation

9.3.4.1 qint32 MainWindow::counter =0 [private]

9.3.4.2 plotGraph\* MainWindow::graph [private]

9.3.4.3 QLabel\* MainWindow::label [private]

9.3.4.4 LCD\* MainWindow::lcd [private]

- 9.3.4.5 `qreal MainWindow::maxData = -99999999` [private]
- 9.3.4.6 `qreal MainWindow::minData = 99999999` [private]
- 9.3.4.7 `QMetaObject::Connection MainWindow::newData` [private]
- 9.3.4.8 `SerialPort* MainWindow::portPtr` [private]
- 9.3.4.9 `RS22812* MainWindow::rawdata` [private]
- 9.3.4.10 `QMetaObject::Connection MainWindow::rData` [private]
- 9.3.4.11 `QGraphicsScene* MainWindow::scene` [private]
- 9.3.4.12 `QVector<QPair<qint64,qreal>> MainWindow::storeData` [private]
- 9.3.4.13 `QElapsedTimer* MainWindow::timeMark` [private]
- 9.3.4.14 `bool MainWindow::timeRunning` [private]
- 9.3.4.15 `QVector<QPair<qreal,qreal>> MainWindow::tmp` [private]
- 9.3.4.16 `Ui::MainWindow* MainWindow::ui` [private]

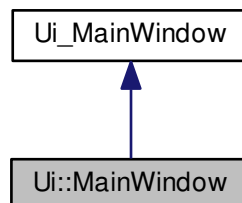
The documentation for this class was generated from the following files:

- [mainwindow.h](#)
- [mainwindow.cpp](#)

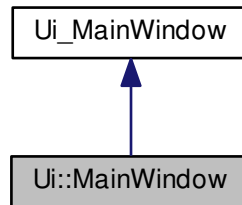
## 9.4 Ui::MainWindow Class Reference

```
#include <ui_mainwindow.h>
```

Inheritance diagram for Ui::MainWindow:



Collaboration diagram for Ui::MainWindow:



### Additional Inherited Members

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

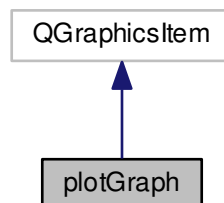
- [build-multimeterGUI-Desktop-Debug/ui\\_mainwindow.h](#)

## 9.5 plotGraph Class Reference

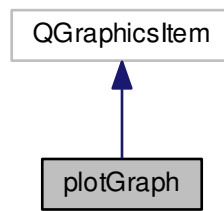
The [plotGraph](#) class.

```
#include <plotgraph.h>
```

Inheritance diagram for `plotGraph`:



Collaboration diagram for plotGraph:



## Public Member Functions

- `plotGraph ()`  
*plotGraph. Default constructor.*
- `plotGraph (QGraphicsScene *_scene)`  
*plotGraph. Constructor.*
- `void paint (QPainter *painter, const QStyleOptionGraphicsItem *option, QWidget *widget)`  
*Override of the paint method of QGraphicsScene.*
- `QRectF boundingRect () const`  
*Returns the bounding rectangle of the graph.*
- `void setXaxis (qint64 minVal, qint64 maxVal)`  
*Sets the maximum and minimum values of the x axis.*
- `void setYaxis (qreal minVal, qreal maxVal)`  
*Sets the maximum and minimum values of the y axis.*
- `void setXsticks (int nSticks)`  
*Sets the number of sticks in the x axis.*
- `void setYsticks (int nSticks)`  
*Sets the number of sticks in the y axis.*
- `void linkData (const QVector< QPair< qint64, qreal > > *dat)`  
*Links the data being plotted with the data vector that is being acquired from the port.*
- `void setUnit (int U)`  
*Selects the appropriate unit depending on the multimeter setting.*
- `void setScene (QGraphicsScene *scene)`

## Private Member Functions

- `void paintAxis (QPainter *painter, const QStyleOptionGraphicsItem *option, QWidget *widget)`  
*Calculates the scale and limits of the axis and draws it.*
- `void plotData (QPainter *painter, const QStyleOptionGraphicsItem *option, QWidget *widget)`  
*Draws the data on the widget.*
- `void labelXaxis (QPainter *painter, QPoint &p1, QPoint &p2)`  
*Adds labels to the X axis.*
- `void labelYaxis (QPainter *painter, QPoint &p1, QPoint &p2)`  
*Adds labels to the Y axis.*
- `QPoint real2Coord (const QPair< qreal, qreal > dpoint)`  
*Transforms reading coordinates to widget coordinates.*

## Private Attributes

- QRect [bRect](#) =QRect(0,0,0,0)
- QPoint [origin](#)
- QPoint [rightX](#)
- QPoint [upperY](#)
- const int [axisMargin](#) =40
- qreal [xmin](#) =0
- qreal [xmax](#) =1
- qreal [ymin](#) =0
- qreal [ymax](#) =1
- int [nx](#) =2
- int [ny](#) =2
- const QVector< QPair< qint64, qreal > > \* [data](#) =NULL
- QString [unit](#)
- QGraphicsScene \* [scene](#)

### 9.5.1 Detailed Description

The [plotGraph](#) class.

This class graphs the values read from the multimeter versus the time.

### 9.5.2 Constructor & Destructor Documentation

#### 9.5.2.1 [plotGraph::plotGraph \( \)](#) `[inline]`

[plotGraph](#). Default constructor.

#### 9.5.2.2 [plotGraph::plotGraph \( QGraphicsScene \\* \\_scene \)](#) `[inline]`

[plotGraph](#). Constructor.

Parameters

<a href="#">_scene</a>	
------------------------	--

### 9.5.3 Member Function Documentation

#### 9.5.3.1 [QRectF plotGraph::boundingRect \( \)](#) `const`

Returns the bounding rectangle of the graph.

Returns

So far, it returns a fake value. Need to implement.

#### 9.5.3.2 [void plotGraph::labelXaxis \( QPainter \\* painter, QPoint & p1, QPoint & p2 \)](#) `[private]`

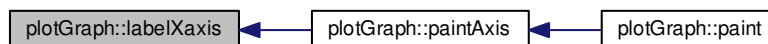
Adds labels to the X axis.



## Parameters

<i>painter</i>	
<i>p1</i>	
<i>p2</i>	

Here is the caller graph for this function:



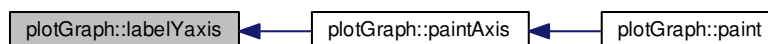
### 9.5.3.3 void plotGraph::labelYaxis ( QPainter \* *painter*, QPoint & *p1*, QPoint & *p2* ) [private]

Adds labels to the Y axis.

## Parameters

<i>painter</i>	
<i>p1</i>	
<i>p2</i>	

Here is the caller graph for this function:



### 9.5.3.4 void plotGraph::linkData ( const QVector< QPair< qint64, qreal > > \* *dat* ) [inline]

Links the data being plotted with the data vector that is being acquired from the port.

## Parameters

<i>dat</i>	
------------	--

Here is the caller graph for this function:



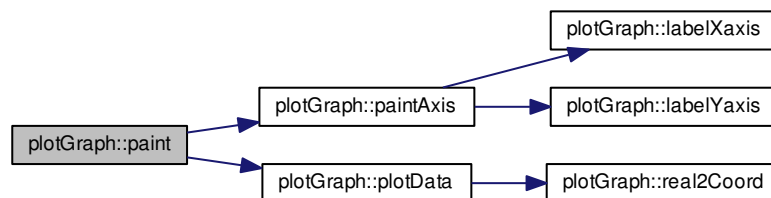
9.5.3.5 void plotGraph::paint ( QPainter \* *painter*, const QStyleOptionGraphicsItem \* *option*, QWidget \* *widget* )

Override of the paint method of QGraphicsScene.

## Parameters

<i>painter</i>	
<i>option</i>	
<i>widget</i>	

Here is the call graph for this function:



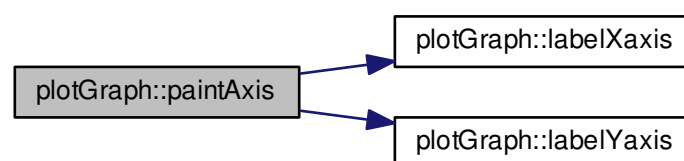
9.5.3.6 void `plotGraph::paintAxis` ( `QPainter * painter`, `const QStyleOptionGraphicsItem * option`, `QWidget * widget` )  
[private]

Calculates the scale and limits of the axis and draws it.

## Parameters

<i>painter</i>	
<i>option</i>	
<i>widget</i>	

Here is the call graph for this function:



Here is the caller graph for this function:



9.5.3.7 void plotGraph::plotData ( QPainter \* *painter*, const QStyleOptionGraphicsItem \* *option*, QWidget \* *widget* )  
[private]

Draws the data on the widget.

Parameters

<i>painter</i>	
<i>option</i>	
<i>widget</i>	

Here is the call graph for this function:



Here is the caller graph for this function:



9.5.3.8 QPoint plotGraph::real2Coord ( const QPair< qreal, qreal > *dpoint* ) [private]

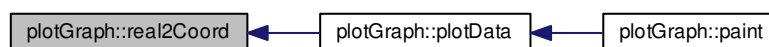
Transforms reading coordinates to widget coordinates.

Parameters

<i>dpoint</i>	
---------------	--

Returns

Here is the caller graph for this function:



9.5.3.9 void plotGraph::setScene ( QGraphicsScene \* *scene* )

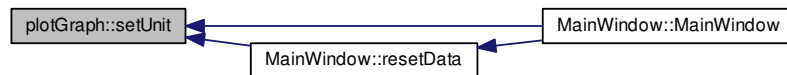
9.5.3.10 void plotGraph::setUnit ( int *U* )

Selects the appropriate unit depending on the multimeter setting.

Parameters

<i>U</i>	
----------	--

Here is the caller graph for this function:



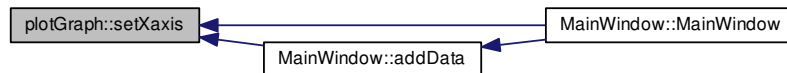
9.5.3.11 void plotGraph::setXaxis ( qint64 *minVal*, qint64 *maxVal* )

Sets the maximum and minimum values of the x axis.

Parameters

<i>minVal</i>	
<i>maxVal</i>	

Here is the caller graph for this function:



9.5.3.12 void plotGraph::setXsticks ( int *nSticks* ) [inline]

Sets the number of sticks in the x axis.

Parameters

<i>nSticks</i>	
----------------	--

Here is the caller graph for this function:



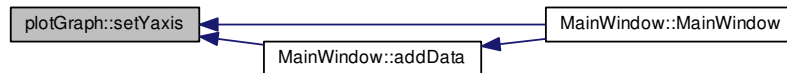
#### 9.5.3.13 void plotGraph::setYaxis ( qreal *minVal*, qreal *maxVal* )

Sets the maximum and minimum values of the y axis.

Parameters

<i>minVal</i>	
<i>maxVal</i>	

Here is the caller graph for this function:



#### 9.5.3.14 void plotGraph::setYsticks ( int *nSticks* ) [inline]

Sets the number of sticks in the y axis.

Parameters

<i>nSticks</i>	
----------------	--

Here is the caller graph for this function:



### 9.5.4 Member Data Documentation

9.5.4.1 const int plotGraph::axisMargin =40 [private]

9.5.4.2 QRect plotGraph::bRect =QRect(0,0,0,0) [private]

9.5.4.3 const QVector<QPair<qint64,qreal> >\* plotGraph::data =NULL [private]

9.5.4.4 int plotGraph::nx =2 [private]

9.5.4.5 int plotGraph::ny =2 [private]

9.5.4.6 QPoint plotGraph::origin [private]

9.5.4.7 QPoint plotGraph::rightX [private]

9.5.4.8 QGraphicsScene\* plotGraph::scene [private]

9.5.4.9 QString plotGraph::unit [private]

9.5.4.10 QPoint plotGraph::upperY [private]

9.5.4.11 qreal plotGraph::xmax =1 [private]

9.5.4.12 qreal plotGraph::xmin =0 [private]

9.5.4.13 qreal plotGraph::ymax =1 [private]

9.5.4.14 qreal plotGraph::ymin =0 [private]

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

- [plotgraph.h](#)
- [plotgraph.cpp](#)

## 9.6 qt\_meta\_stringdata\_LCD\_t Struct Reference

### Public Attributes

- QByteArrayData [data](#) [1]
- char [stringdata0](#) [4]

### 9.6.1 Member Data Documentation

9.6.1.1 QByteArrayData qt\_meta\_stringdata\_LCD\_t::data[1]

9.6.1.2 char qt\_meta\_stringdata\_LCD\_t::stringdata0[4]

The documentation for this struct was generated from the following file:

- build-multimeterGUI-Desktop-Debug/[moc\\_lcd.cpp](#)

## 9.7 qt\_meta\_stringdata\_MainWindow\_t Struct Reference

### Public Attributes

- QByteArrayData [data](#) [6]
- char [stringdata0](#) [83]

### 9.7.1 Member Data Documentation

9.7.1.1 QByteArrayData qt\_meta\_stringdata\_MainWindow\_t::data[6]

9.7.1.2 char qt\_meta\_stringdata\_MainWindow\_t::stringdata0[83]

The documentation for this struct was generated from the following file:

- build-multimeterGUI-Desktop-Debug/[moc\\_mainwindow.cpp](#)

## 9.8 qt\_meta\_stringdata\_plotGraph\_t Struct Reference

### Public Attributes

- QByteArrayData [data](#) [1]
- char [stringdata0](#) [10]

### 9.8.1 Member Data Documentation

9.8.1.1 QByteArrayData qt\_meta\_stringdata\_plotGraph\_t::data[1]

9.8.1.2 char qt\_meta\_stringdata\_plotGraph\_t::stringdata0[10]

The documentation for this struct was generated from the following file:

- build-multimeterGUI-Desktop-Debug/[moc\\_plotgraph.cpp](#)

## 9.9 qt\_meta\_stringdata\_RS22812\_t Struct Reference

### Public Attributes

- QByteArrayData [data](#) [6]
- char [stringdata0](#) [39]

### 9.9.1 Member Data Documentation

9.9.1.1 QByteArrayData qt\_meta\_stringdata\_RS22812\_t::data[6]

9.9.1.2 char qt\_meta\_stringdata\_RS22812\_t::stringdata0[39]

The documentation for this struct was generated from the following file:

- build-multimeterGUI-Desktop-Debug/[moc\\_rs22812.cpp](#)

## 9.10 qt\_meta\_stringdata\_SerialPort\_t Struct Reference

### Public Attributes

- QByteArrayData [data](#) [5]
- char [stringdata0](#) [35]

### 9.10.1 Member Data Documentation

9.10.1.1 QByteArrayData qt\_meta\_stringdata\_SerialPort\_t::data[5]

9.10.1.2 char qt\_meta\_stringdata\_SerialPort\_t::stringdata0[35]

The documentation for this struct was generated from the following file:

- build-multimeterGUI-Desktop-Debug/[moc\\_serialport.cpp](#)

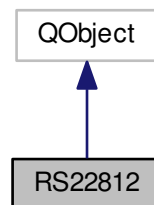


## 9.11 RS22812 Class Reference

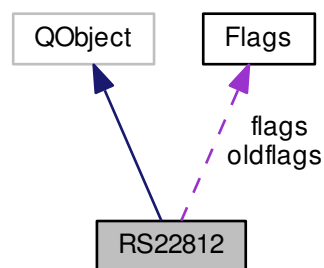
Decoding of the data sent by the Radio Shack 22-812.

```
#include <rs22812.h>
```

Inheritance diagram for RS22812:



Collaboration diagram for RS22812:



### Public Slots

- void [newValue](#) (const QByteArray &data)  
*Reads a new packet and reformats the data.*

### Signals

- void [newMode](#) ()
- void [newData](#) ()

### Public Member Functions

- [RS22812](#) (QObject \*parent=0)  
*Constructor.*

- float `getVal ()` const  
*Returns the numeric value of the reading.*
- `Flags getFlags ()` const  
*Gets the current flags structure.*
- `QString getDigitString ()` const  
*getDigitString It returns the multimeter reading in string format.*
- `uint getMode ()` const  
*getMode*

## Private Member Functions

- bool `modeChanged ()`  
*Checks the flags structure to see whether the read mode changed.*
- `QString byte2Digit (uchar byte)`  
*Translates the RS 22-812 byte value of the LCD mapping into a digit.*
- void `resetFlags (Flags &f)`  
*Sets all the flags to false.*

## Private Attributes

- `uint mode`
- `Flags flags`
- `Flags oldflags`
- `QString digits`

### 9.11.1 Detailed Description

Decoding of the data sent by the Radio Shack 22-812.

Part of the information was obtained from [http://sigrok.org/wiki/RadioShack\\_22-812](http://sigrok.org/wiki/RadioShack_22-812) and <https://code.google.com/archive/p/rs22812/>

RS 22-812 sends 9-bytes packets. Each packet is a mapping of the LCD of the screen plus some extra information.

	Bit							
Byte	7	6	5	4	3	2	1	0
0	----- Mode -----							
1	Hz	Ohms	K	M	F	A	V	m
2	u	n	dBm	s	%	hFE	REL	MIN
3	4D	4C	4G	4B	DP3	4E	4F	4A
4	3D	3C	3G	3B	DP2	3E	3F	3A
5	2D	2C	2G	2B	DP1	2E	2F	2A
6	1D	1C	1G	1B	MAX	1E	1F	1A
7	Beep	Diode	Bat	Hold	-	~	RS232	Auto
8	----- Checksum -----							

The LED mapping is:

```
| --A-- |
|       |
F       B
|       |
| --G-- |
|       |
E       C
|       |
| --D-- |
```

So, the equivalence between int value and digit are:

215 : "0", 80 : "1", 181 : "2", 241 : "3", 114 : "4",  
 227 : "5", 231 : "6", 81 : "7", 247 : "8", 243 : "9", 39 : "F",  
 55 : "P", 167 : "E", 135 : "C", 134 : "L", 118 : "H", 6 : "I",  
 102 : "h", 36 : "r", 166 : "t", 100 : "n", 32 : "-", 0 : " "

And the possible modes are:

.	.	.
0=DC V	1=AC V	2=DC uA
3=DC mA	4=DC A	5=AC uA
6=AC mA	7=AC A	8=OHM
9=CAP	10=HZ	11=NET HZ
12=AMP HZ	13=DUTY	14=NET DUTY
15=AMP DUTY	16=WIDTH	17=NET WIDTH
18=AMP WIDTH	19=DIODE	20=CONT
21=HFE	22=LOGIC	23=DBM
24=EF	25=TEMP	.

## 9.11.2 Constructor & Destructor Documentation

### 9.11.2.1 RS22812::RS22812 ( QObject \* *parent* = 0 ) [explicit]

Constructor.

Parameters

<i>parent</i>	It sets the mode to 0 and resets all the flags.
---------------	---

Here is the call graph for this function:



## 9.11.3 Member Function Documentation

### 9.11.3.1 QString RS22812::byte2Digit ( uchar *byte* ) [private]

Translates the RS 22-812 byte value of the [LCD](#) mapping into a digit.

Parameters

<i>byte</i>	RS 22-812 byte value
-------------	----------------------

**Returns**

String with the equivalent digit.

Here is the caller graph for this function:

**9.11.3.2** `QString RS22812::getDigitString ( ) const [inline]`

`getDigitString` It returns the multimeter reading in string format.

**Returns**

Here is the caller graph for this function:

**9.11.3.3** `Flags RS22812::getFlags ( ) const [inline]`

Gets the current flags structure.

**Returns**

Here is the caller graph for this function:



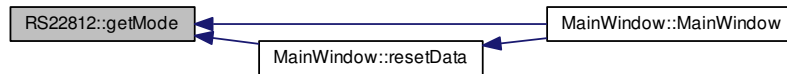
#### 9.11.3.4 uint RS22812::getMode ( ) const [inline]

getMode

Returns

It returns the mode on which the multimeter is working.

Here is the caller graph for this function:

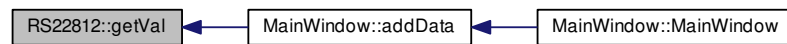


#### 9.11.3.5 float RS22812::getVal ( ) const

Returns the numeric value of the reading.

Returns

Here is the caller graph for this function:



#### 9.11.3.6 bool RS22812::modeChanged ( ) [private]

Checks the flags structure to see whether the read mode changed.

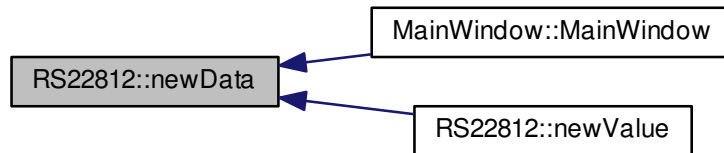
Returns

Here is the caller graph for this function:



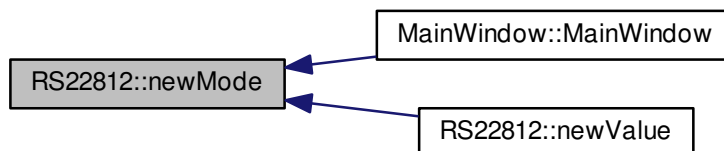
#### 9.11.3.7 void RS22812::newData ( ) [signal]

Here is the caller graph for this function:



#### 9.11.3.8 void RS22812::newMode ( ) [signal]

Here is the caller graph for this function:



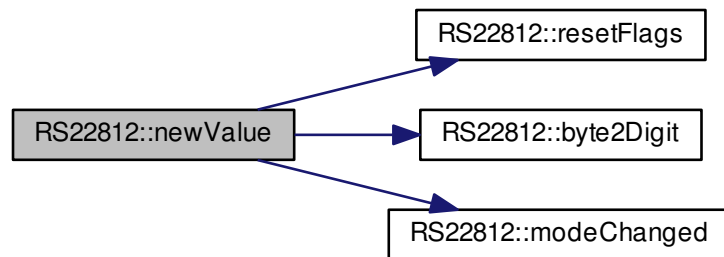
#### 9.11.3.9 void RS22812::newValue ( const QByteArray & data ) [slot]

Reads a new packet and reformats the data.

Parameters

<i>data</i>	
-------------	--

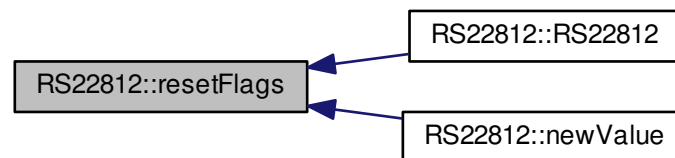
Here is the call graph for this function:



#### 9.11.3.10 void RS22812::resetFlags ( Flags & f ) [private]

Sets all the flags to false.

Here is the caller graph for this function:



### 9.11.4 Member Data Documentation

#### 9.11.4.1 QString RS22812::digits [private]

#### 9.11.4.2 Flags RS22812::flags [private]

#### 9.11.4.3 uint RS22812::mode [private]

#### 9.11.4.4 Flags RS22812::oldflags [private]

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

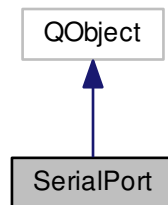
- [rs22812.h](#)
- build-multimeterGUI-Desktop-Debug/[moc\\_rs22812.cpp](#)
- [rs22812.cpp](#)

## 9.12 SerialPort Class Reference

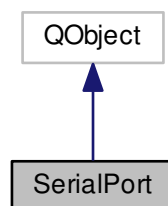
Class to manage the communication with a serial port.

```
#include <serialport.h>
```

Inheritance diagram for SerialPort:



Collaboration diagram for SerialPort:



### Public Slots

- void `ready` ()  
*Re-emits the readyRead signal.*

### Signals

- void `readyRead` (QByteArray `buffer`)

### Public Member Functions

- `SerialPort` (QObject \*parent=0)  
*Constructor.*
- `~SerialPort` ()  
*Destructor.*



- bool [openPort](#) (const QString portName)  
*Opens a serial port for reading. If port name is empty, it does nothing. If port name is not available. Does nothing and raises a warning.*
- bool [closePort](#) ()  
*Closes the port if it is open and disconnects the readConnect signal.*
- QList< QSerialPortInfo > [listPorts](#) ()  
*Obtains and return the list of available ports.*

### Private Member Functions

- void [readPort](#) ()  
*Reads the available data.*

### Private Attributes

- QSerialPortInfo \* [ports](#)
- QSerialPort \* [activePort](#)
- bool [isOpen](#)
- QByteArray [buffer](#)
- QMetaObject::Connection [readConnect](#)
- const QSerialPort::OpenMode [MODE](#) =QSerialPort::ReadOnly

### Static Private Attributes

- static const QSerialPort::BaudRate [BAUDRATE](#) =QSerialPort::Baud4800
- static const QSerialPort::DataBits [DATABITS](#) =QSerialPort::Data8
- static const QSerialPort::StopBits [STOPBITS](#) =QSerialPort::OneStop
- static const QSerialPort::Parity [PARITY](#) =QSerialPort::NoParity

#### 9.12.1 Detailed Description

Class to manage the communication with a serial port.

RS 22-812 sends 9bytes long packets with the codified information. This class is meant to read those packets and send it to the [RS22812](#) class to store and interpret.

#### 9.12.2 Constructor & Destructor Documentation

##### 9.12.2.1 SerialPort::SerialPort ( QObject \* *parent* = 0 ) [explicit]

Constructor.

Parameters

<i>parent</i>	
---------------	--

##### 9.12.2.2 SerialPort::~SerialPort ( )

Destructor.

It closes the port if it is open.

Here is the call graph for this function:



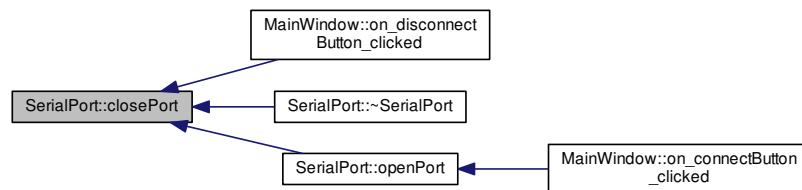
### 9.12.3 Member Function Documentation

#### 9.12.3.1 `bool SerialPort::closePort ( )`

Closes the port if it is open and disconnects the readConnect signal.

**Returns**

Here is the caller graph for this function:



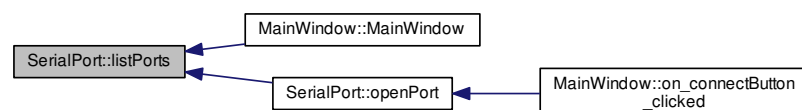
#### 9.12.3.2 `QList< QSerialPortInfo > SerialPort::listPorts ( )`

Obtains and return the list of available ports.

**Returns**

`QList<QSerialPortInfo>`

Here is the caller graph for this function:



#### 9.12.3.3 `bool SerialPort::openPort ( const QString portName )`

Opens a serial port for reading. If port name is empty, it does nothing. If port name is not available. Does nothing and raises a warning.

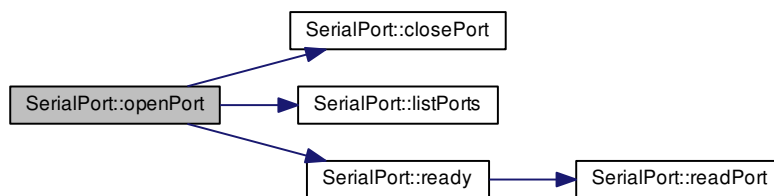
## Parameters

<i>portName</i>	Name of the port as returned from QSerialPortInfo.
-----------------	--

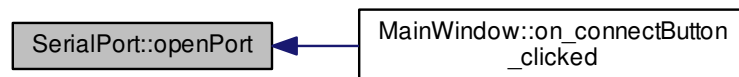
## Returns

If successful, returns 1. Else it returns 0;

Here is the call graph for this function:



Here is the caller graph for this function:



#### 9.12.3.4 void SerialPort::readPort ( ) [private]

Reads the available data.

Here is the caller graph for this function:



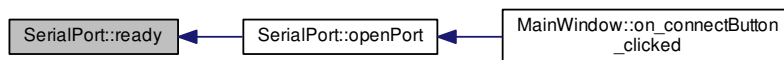
#### 9.12.3.5 void SerialPort::ready ( ) [slot]

Re-emits the readyRead signal.

Here is the call graph for this function:



Here is the caller graph for this function:



9.12.3.6 `void SerialPort::readyRead ( QByteArray buffer )` [signal]

Here is the caller graph for this function:



## 9.12.4 Member Data Documentation

9.12.4.1 `QSerialPort* SerialPort::activePort` [private]

9.12.4.2 `const QSerialPort::BaudRate SerialPort::BAUDRATE =QSerialPort::Baud4800` [static],[private]

9.12.4.3 `QByteArray SerialPort::buffer` [private]

9.12.4.4 `const QSerialPort::DataBits SerialPort::DATABITS =QSerialPort::Data8` [static],[private]

9.12.4.5 `bool SerialPort::isOpen` [private]

9.12.4.6 `const QSerialPort::OpenMode SerialPort::MODE =QSerialPort::ReadOnly` [private]

9.12.4.7 `const QSerialPort::Parity SerialPort::PARITY =QSerialPort::NoParity` [static],[private]

9.12.4.8 `QSerialPortInfo* SerialPort::ports` [private]

9.12.4.9 `QMetaObject::Connection SerialPort::readConnect` [private]

9.12.4.10 `const QSerialPort::StopBits SerialPort::STOPBITS =QSerialPort::OneStop` `[static],[private]`

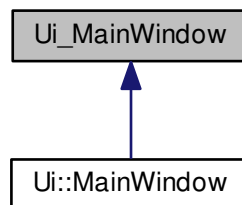
The documentation for this class was generated from the following files:

- [serialport.h](#)
- build-multimeterGUI-Desktop-Debug/[moc\\_serialport.cpp](#)
- [serialport.cpp](#)

## 9.13 Ui\_MainWindow Class Reference

```
#include <ui_mainwindow.h>
```

Inheritance diagram for Ui\_MainWindow:



### Public Member Functions

- void [setupUi](#) (QMainWindow \*[MainWindow](#))
- void [retranslateUi](#) (QMainWindow \*[MainWindow](#))

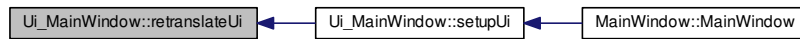
### Public Attributes

- QWidget \* [centralWidget](#)
- QGridLayout \* [gridLayout](#)
- QHBoxLayout \* [horizontalLayout](#)
- QVBoxLayout \* [verticalLayout\\_2](#)
- QPushButton \* [connectButton](#)
- QPushButton \* [disconnectButton](#)
- QVBoxLayout \* [verticalLayout](#)
- QLabel \* [labelPort](#)
- QComboBox \* [comboBoxPort](#)
- QGraphicsView \* [graphPlot](#)
- QMenuBar \* [menuBar](#)
- QToolBar \* [mainToolBar](#)
- QStatusBar \* [statusBar](#)
- QToolBar \* [toolBar](#)

### 9.13.1 Member Function Documentation

#### 9.13.1.1 void Ui\_MainWindow::retranslateUi ( QMainWindow \* *MainWindow* ) [inline]

Here is the caller graph for this function:



#### 9.13.1.2 void Ui\_MainWindow::setupUi ( QMainWindow \* *MainWindow* ) [inline]

Here is the call graph for this function:



Here is the caller graph for this function:



### 9.13.2 Member Data Documentation

#### 9.13.2.1 QWidget\* Ui\_MainWindow::centralWidget

#### 9.13.2.2 QComboBox\* Ui\_MainWindow::comboBoxPort

#### 9.13.2.3 QPushButton\* Ui\_MainWindow::connectButton

#### 9.13.2.4 QPushButton\* Ui\_MainWindow::disconnectButton

#### 9.13.2.5 QGraphicsView\* Ui\_MainWindow::graphPlot

#### 9.13.2.6 QGridLayout\* Ui\_MainWindow::gridLayout

#### 9.13.2.7 QHBoxLayout\* Ui\_MainWindow::horizontalLayout

9.13.2.8 `QLabel*` `Ui_MainWindow::labelPort`

9.13.2.9 `QToolBar*` `Ui_MainWindow::mainToolBar`

9.13.2.10 `QMenuBar*` `Ui_MainWindow::menuBar`

9.13.2.11 `QStatusBar*` `Ui_MainWindow::statusBar`

9.13.2.12 `QToolBar*` `Ui_MainWindow::toolBar`

9.13.2.13 `QVBoxLayout*` `Ui_MainWindow::verticalLayout`

9.13.2.14 `QVBoxLayout*` `Ui_MainWindow::verticalLayout_2`

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

- `build-multimeterGUI-Desktop-Debug/ui_mainwindow.h`

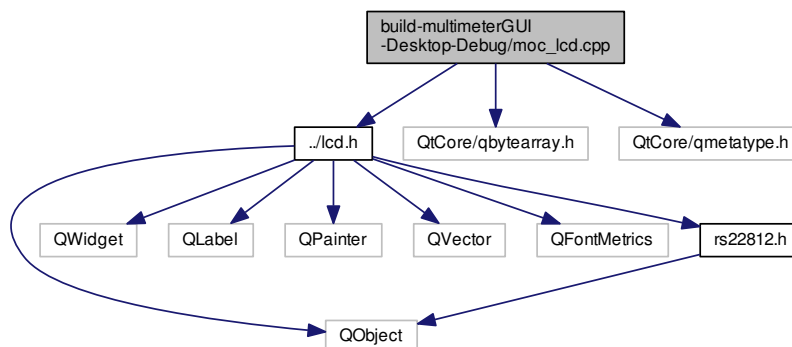


## Chapter 10

# File Documentation

### 10.1 build-multimeterGUI-Desktop-Debug/moc\_lcd.cpp File Reference

```
#include "../lcd.h"  
#include <QtCore/qbytearray.h>  
#include <QtCore/qmetatype.h>  
Include dependency graph for moc_lcd.cpp:
```



#### Classes

- struct [qt\\_meta\\_stringdata\\_LCD\\_t](#)

#### Macros

- `#define` [QT\\_MOC\\_LITERAL](#)(idx, ofs, len)

#### Variables

- static const [qt\\_meta\\_stringdata\\_LCD\\_t](#) [qt\\_meta\\_stringdata\\_LCD](#)
- static const uint [qt\\_meta\\_data\\_LCD](#) []

## 10.1.1 Macro Definition Documentation

### 10.1.1.1 `#define QT_MOC_LITERAL( idx, ofs, len )`

**Value:**

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_LCD_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

## 10.1.2 Variable Documentation

### 10.1.2.1 `const uint qt_meta_data_LCD[] [static]`

**Initial value:**

```
= {
    7,
    0,
    0,    0,
    0,    0,
    0,    0,
    0,    0,
    0,    0,
    0,    0,
    0,
    0
}
```

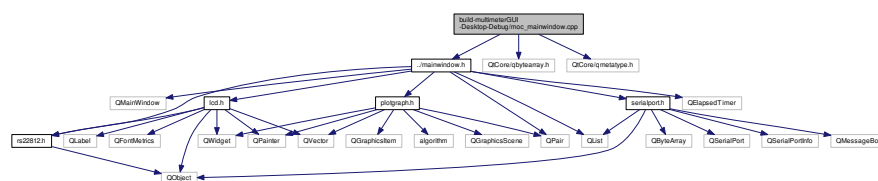
### 10.1.2.2 `const qt_meta_stringdata_LCD_t qt_meta_stringdata_LCD [static]`

**Initial value:**

```
= {
    {
        QT_MOC_LITERAL(0, 0, 3)
    },
    "LCD"
}
```

## 10.2 build-multimeterGUI-Desktop-Debug/moc\_mainwindow.cpp File Reference

```
#include "../mainwindow.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
Include dependency graph for moc_mainwindow.cpp:
```



## Classes

- struct [qt\\_meta\\_stringdata\\_MainWindow\\_t](#)

## Macros

- #define [QT\\_MOC\\_LITERAL](#)(idx, ofs, len)

## Variables

- static const [qt\\_meta\\_stringdata\\_MainWindow\\_t](#) [qt\\_meta\\_stringdata\\_MainWindow](#)
- static const uint [qt\\_meta\\_data\\_MainWindow](#) []

### 10.2.1 Macro Definition Documentation

#### 10.2.1.1 #define QT\_MOC\_LITERAL( idx, ofs, len )

##### Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_MainWindow_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

### 10.2.2 Variable Documentation

#### 10.2.2.1 const uint qt\_meta\_data\_MainWindow[] [static]

##### Initial value:

```
= {

    7,
    0,
    0,    0,
    4,   14,
    0,    0,
    0,    0,
    0,    0,
    0,
    0,
    0,

    1,    0,   34,    2, 0x08 ,
    3,    0,   35,    2, 0x08 ,
    4,    0,   36,    2, 0x08 ,
    5,    0,   37,    2, 0x08 ,

    QMetaType::Void,
    QMetaType::Void,
    QMetaType::Void,
    QMetaType::Void,
    0
}
```

#### 10.2.2.2 const qt\_meta\_stringdata\_MainWindow\_t qt\_meta\_stringdata\_MainWindow [static]

##### Initial value:

```

= {
{
    QT_MOC_LITERAL(0, 0, 10),
    QT_MOC_LITERAL(1, 11, 24),
    QT_MOC_LITERAL(2, 36, 0),
    QT_MOC_LITERAL(3, 37, 27),
    QT_MOC_LITERAL(4, 65, 7),
    QT_MOC_LITERAL(5, 73, 9)
},
    "MainWindow\0on_connectButton_clicked\0"
    "\0on_disconnectButton_clicked\0addData\0"
    "resetData"
}

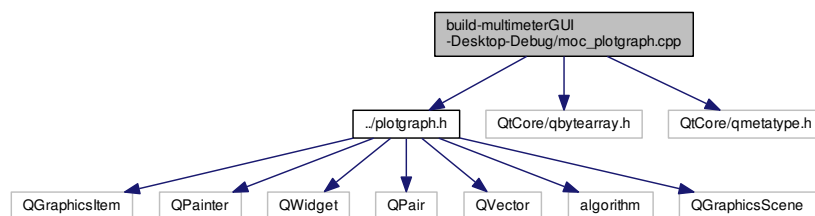
```

## 10.3 build-multimeterGUI-Desktop-Debug/moc\_plotgraph.cpp File Reference

```

#include "../plotgraph.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
Include dependency graph for moc_plotgraph.cpp:

```



### Classes

- struct [qt\\_meta\\_stringdata\\_plotGraph\\_t](#)

### Macros

- #define [QT\\_MOC\\_LITERAL](#)(idx, ofs, len)

### Variables

- static const [qt\\_meta\\_stringdata\\_plotGraph\\_t](#) [qt\\_meta\\_stringdata\\_plotGraph](#)
- static const uint [qt\\_meta\\_data\\_plotGraph](#) []

### 10.3.1 Macro Definition Documentation

#### 10.3.1.1 #define QT\_MOC\_LITERAL( idx, ofs, len )

##### Value:

```

Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt\_meta\_stringdata\_plotGraph\_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )

```

### 10.3.2 Variable Documentation

#### 10.3.2.1 `const uint qt_meta_data_plotGraph[]` `[static]`

**Initial value:**

```
= {
    7,
    0,
    0,    0,
    0,    0,
    0,    0,
    0,    0,
    0,    0,
    4,
    0,

    0
}
```

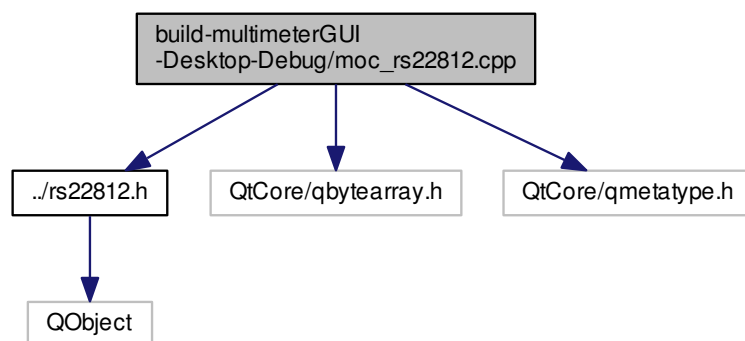
#### 10.3.2.2 `const qt_meta_stringdata_plotGraph_t qt_meta_stringdata_plotGraph` `[static]`

**Initial value:**

```
= {
    {
        QT_MOC_LITERAL(0, 0, 9)
    },
    "plotGraph"
}
```

## 10.4 build-multimeterGUI-Desktop-Debug/moc\_rs22812.cpp File Reference

```
#include "../rs22812.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
Include dependency graph for moc_rs22812.cpp:
```



## Classes

- struct [qt\\_meta\\_stringdata\\_RS22812\\_t](#)

## Macros

- #define [QT\\_MOC\\_LITERAL](#)(idx, ofs, len)

## Variables

- static const  
[qt\\_meta\\_stringdata\\_RS22812\\_t](#) qt\_meta\_stringdata\_RS22812
- static const uint [qt\\_meta\\_data\\_RS22812](#) []

### 10.4.1 Macro Definition Documentation

#### 10.4.1.1 #define QT\_MOC\_LITERAL( *idx*, *ofs*, *len* )

##### Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt\_meta\_stringdata\_RS22812\_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

### 10.4.2 Variable Documentation

#### 10.4.2.1 const uint qt\_meta\_data\_RS22812[] [static]

##### Initial value:

```
= {

    7,
    0,
    0,    0,
    3,    14,
    0,    0,
    0,    0,
    0,    0,
    0,    0,
    0,
    2,

    1,    0,    29,    2, 0x06 ,
    3,    0,    30,    2, 0x06 ,

    4,    1,    31,    2, 0x0a ,

    QMetaType::Void,
    QMetaType::Void,

    QMetaType::Void, QMetaType::QByteArray,    5,

    0
}
```

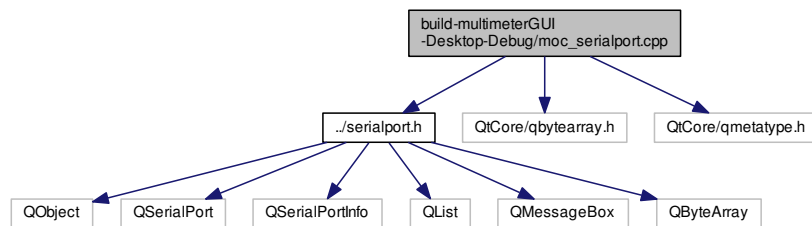
10.4.2.2 `const qt_meta_stringdata_RS22812_t qt_meta_stringdata_RS22812 [static]`**Initial value:**

```
= {
{
QT_MOC_LITERAL(0, 0, 7),
QT_MOC_LITERAL(1, 8, 7),
QT_MOC_LITERAL(2, 16, 0),
QT_MOC_LITERAL(3, 17, 7),
QT_MOC_LITERAL(4, 25, 8),
QT_MOC_LITERAL(5, 34, 4)

},
"RS22812\0newMode\0\0newData\0newValue\0"
"data"
}
```

## 10.5 build-multimeterGUI-Desktop-Debug/moc\_serialport.cpp File Reference

```
#include "../serialport.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
Include dependency graph for moc_serialport.cpp:
```

**Classes**

- struct [qt\\_meta\\_stringdata\\_SerialPort\\_t](#)

**Macros**

- `#define` [QT\\_MOC\\_LITERAL](#)(idx, ofs, len)

**Variables**

- static const [qt\\_meta\\_stringdata\\_SerialPort\\_t qt\\_meta\\_stringdata\\_SerialPort](#)
- static const uint [qt\\_meta\\_data\\_SerialPort \[\]](#)

## 10.5.1 Macro Definition Documentation

10.5.1.1 `#define QT_MOC_LITERAL( idx, ofs, len )`**Value:**

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_SerialPort_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

## 10.5.2 Variable Documentation

### 10.5.2.1 `const uint qt_meta_data_SerialPort[]` [static]

**Initial value:**

```
= {
    7,
    0,
    0,    0,
    2,    14,
    0,    0,
    0,    0,
    0,    0,
    0,
    1,

    1,    1,    24,    2, 0x06 ,

    4,    0,    27,    2, 0x0a ,

    QMetaType::Void, QMetaType::QByteArray,    3,

    QMetaType::Void,

    0
}
```

### 10.5.2.2 `const qt_meta_stringdata_SerialPort_t qt_meta_stringdata_SerialPort` [static]

**Initial value:**

```
= {
    {
        QT_MOC_LITERAL(0, 0, 10),
        QT_MOC_LITERAL(1, 11, 9),
        QT_MOC_LITERAL(2, 21, 0),
        QT_MOC_LITERAL(3, 22, 6),
        QT_MOC_LITERAL(4, 29, 5)
    },
    "SerialPort\0readyRead\0\0buffer\0ready"
}
```



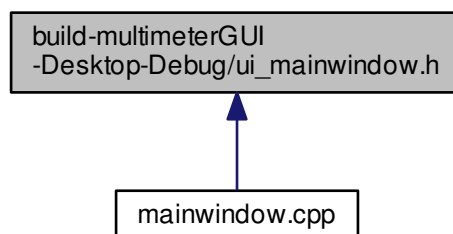
## 10.6 build-multimeterGUI-Desktop-Debug/ui\_mainwindow.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QGraphicsView>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QMainWindow>
#include <QtWidgets/QMenuBar>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QStatusBar>
#include <QtWidgets/QToolBar>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
```

Include dependency graph for ui\_mainwindow.h:



This graph shows which files directly or indirectly include this file:



### Classes

- class [Ui\\_MainWindow](#)
- class [Ui::MainWindow](#)

### Namespaces

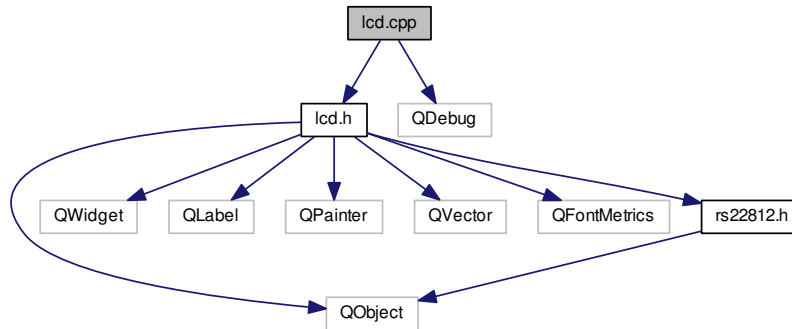
- [Ui](#)

## 10.7 lcd.cpp File Reference

```
#include "lcd.h"
```

```
#include <QDebug>
```

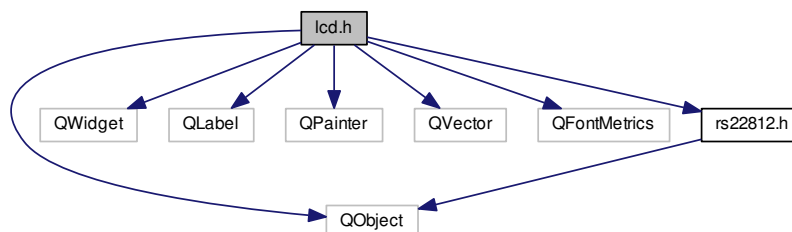
Include dependency graph for lcd.cpp:



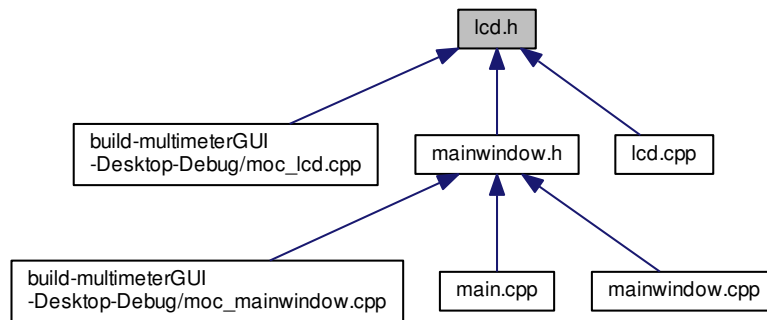
## 10.8 lcd.h File Reference

```
#include <QObject>
#include <QWidget>
#include <QLabel>
#include <QPainter>
#include <QVector>
#include <QFontMetrics>
#include "rs22812.h"
```

Include dependency graph for lcd.h:



This graph shows which files directly or indirectly include this file:



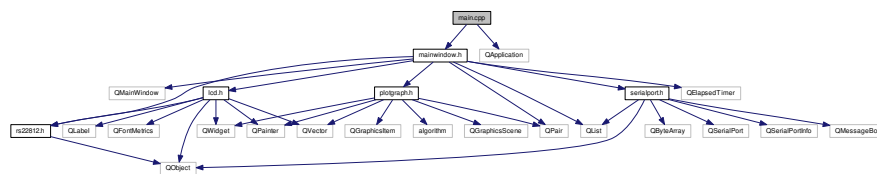
## Classes

- class [LCD](#)

The [LCD](#) class displays the numerical value read.

## 10.9 main.cpp File Reference

```
#include "mainwindow.h"
#include <QApplication>
Include dependency graph for main.cpp:
```



## Functions

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

### 10.9.1 Function Documentation

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

**Todo** Add display controls.

## 10.10 mainwindow.cpp File Reference

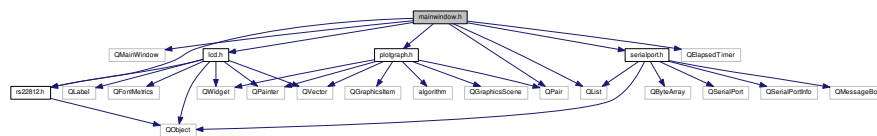
```
#include "mainwindow.h"
```

```
#include "ui_mainwindow.h"
#include <QDebug>
Include dependency graph for mainwindow.cpp:
```

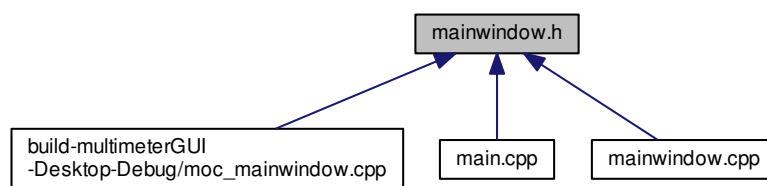


## 10.11 mainwindow.h File Reference

```
#include <QMainWindow>
#include <QList>
#include "serialport.h"
#include "rs22812.h"
#include "lcd.h"
#include "plotgraph.h"
#include <QPair>
#include <QElapsedTimer>
Include dependency graph for mainwindow.h:
```



This graph shows which files directly or indirectly include this file:



## Classes

- class [MainWindow](#)

The *MainWindow* class.

## Namespaces

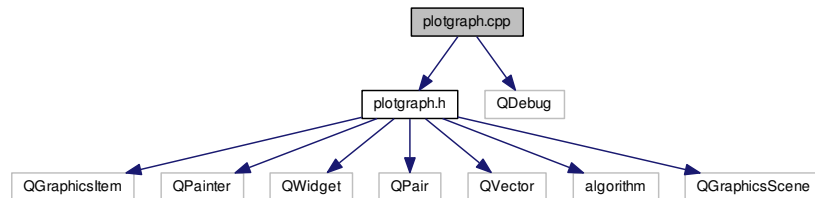
- [Ui](#)

## 10.12 plotgraph.cpp File Reference

```
#include "plotgraph.h"
```

```
#include <QDebug>
```

Include dependency graph for plotgraph.cpp:



## 10.13 plotgraph.h File Reference

```
#include <QGraphicsItem>
```

```
#include <QPainter>
```

```
#include <QWidget>
```

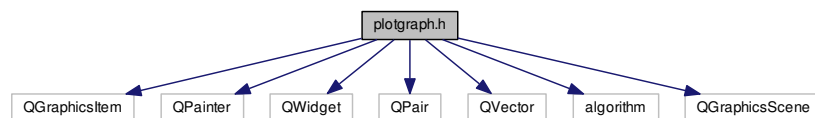
```
#include <QPair>
```

```
#include <QVector>
```

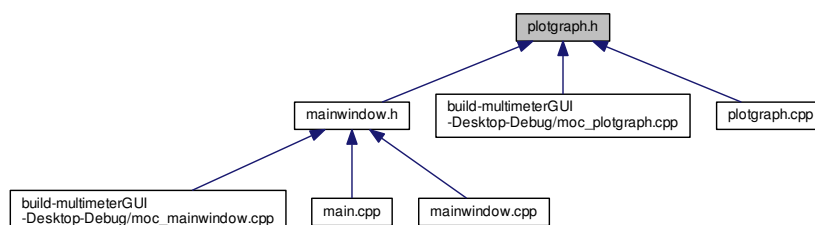
```
#include <algorithm>
```

```
#include <QGraphicsScene>
```

Include dependency graph for plotgraph.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [plotGraph](#)

The [plotGraph](#) class.

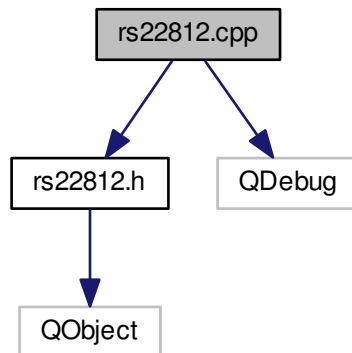
## 10.14 README.md File Reference

## 10.15 rs22812.cpp File Reference

```
#include "rs22812.h"
```

```
#include <QDebug>
```

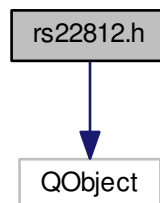
Include dependency graph for rs22812.cpp:



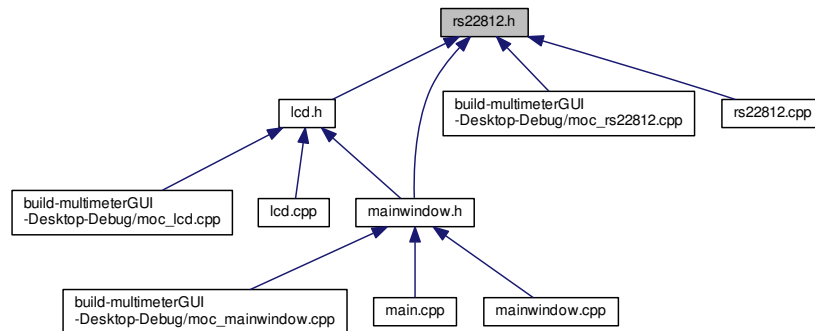
## 10.16 rs22812.h File Reference

```
#include <QObject>
```

Include dependency graph for rs22812.h:



This graph shows which files directly or indirectly include this file:



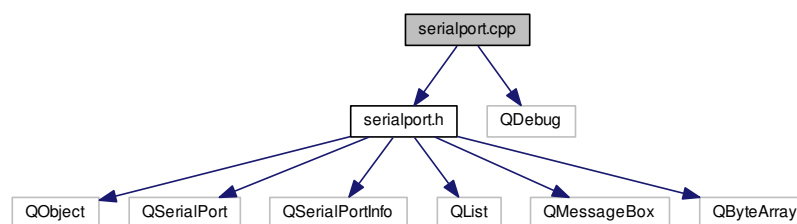
## Classes

- struct [Flags](#)  
*Definition of custom data type.*
- class [RS22812](#)  
*Decoding of the data sent by the Radio Shack 22-812.*

## 10.17 serialport.cpp File Reference

```
#include "serialport.h"
#include <QDebug>
```

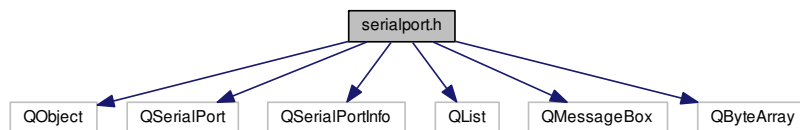
Include dependency graph for serialport.cpp:



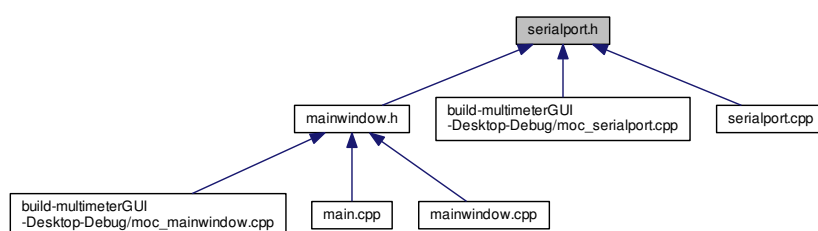
## 10.18 serialport.h File Reference

```
#include <QObject>
#include <QSerialPort>
#include <QSerialPortInfo>
#include <QList>
#include <QMessageBox>
#include <QByteArray>
```

Include dependency graph for serialport.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [SerialPort](#)

*Class to manage the communication with a serial port.*



# Index

~MainWindow  
    MainWindow, [24](#)  
~SerialPort  
    SerialPort, [47](#)

A  
    Flags, [18](#)  
activePort  
    SerialPort, [51](#)  
addData  
    MainWindow, [24](#)  
Auto  
    Flags, [18](#)  
axisMargin  
    plotGraph, [36](#)

BAUDRATE  
    SerialPort, [51](#)  
bRect  
    plotGraph, [36](#)  
Bat  
    Flags, [18](#)  
Beep  
    Flags, [18](#)  
boundingRect  
    plotGraph, [30](#)  
buffer  
    SerialPort, [51](#)  
build-multimeterGUI-Desktop-Debug/moc\_lcd.cpp, [55](#)  
build-multimeterGUI-Desktop-Debug/moc\_mainwindow.-  
    cpp, [56](#)  
build-multimeterGUI-Desktop-Debug/moc\_plotgraph.-  
    cpp, [58](#)  
build-multimeterGUI-Desktop-Debug/moc\_rs22812.cpp,  
    [59](#)  
build-multimeterGUI-Desktop-Debug/moc\_serialport.-  
    cpp, [61](#)  
build-multimeterGUI-Desktop-Debug/ui\_mainwindow.h,  
    [63](#)  
byte2Digit  
    RS22812, [41](#)  
  
centralWidget  
    Ui\_MainWindow, [53](#)  
checksum  
    Flags, [18](#)  
closePort  
    SerialPort, [48](#)  
comboBoxPort  
    Ui\_MainWindow, [53](#)

connectButton  
    Ui\_MainWindow, [53](#)  
continuity  
    Flags, [18](#)  
counter  
    MainWindow, [26](#)

DATABITS  
    SerialPort, [51](#)  
dBm  
    Flags, [18](#)  
DP  
    Flags, [18](#)  
dash  
    Flags, [18](#)  
data  
    LCD, [21](#)  
    plotGraph, [36](#)  
    qt\_meta\_stringdata\_LCD\_t, [37](#)  
    qt\_meta\_stringdata\_MainWindow\_t, [37](#)  
    qt\_meta\_stringdata\_plotGraph\_t, [38](#)  
    qt\_meta\_stringdata\_RS22812\_t, [38](#)  
    qt\_meta\_stringdata\_SerialPort\_t, [38](#)

digits  
    RS22812, [45](#)  
Diode  
    Flags, [18](#)  
disconnectButton  
    Ui\_MainWindow, [53](#)

F  
    Flags, [18](#)  
Flags, [17](#)  
    A, [18](#)  
    Auto, [18](#)  
    Bat, [18](#)  
    Beep, [18](#)  
    checksum, [18](#)  
    continuity, [18](#)  
    dBm, [18](#)  
    DP, [18](#)  
    dash, [18](#)  
    Diode, [18](#)  
    F, [18](#)  
    hFE, [18](#)  
    Hold, [18](#)  
    Hz, [18](#)  
    k, [18](#)  
    M, [19](#)  
    m, [19](#)

- MAX, 19
- MIN, 19
- n, 19
- Ohms, 19
- operator==, 18
- percent, 19
- REL, 19
- RS232, 19
- s, 19
- tilde, 19
- u, 19
- V, 19
- flags
  - RS22812, 45
- getDigitString
  - RS22812, 42
- getFlags
  - RS22812, 42
- getMode
  - RS22812, 42
- getVal
  - RS22812, 43
- graph
  - MainWindow, 26
- graphPlot
  - Ui\_MainWindow, 53
- gridLayout
  - Ui\_MainWindow, 53
- hFE
  - Flags, 18
- Hold
  - Flags, 18
- horizontalLayout
  - Ui\_MainWindow, 53
- Hz
  - Flags, 18
- isOpen
  - SerialPort, 51
- k
  - Flags, 18
- LCD, 19
  - data, 21
  - LCD, 20
  - lbl, 21
  - LCD, 20
  - paintEvent, 21
- label
  - MainWindow, 26
- labelPort
  - Ui\_MainWindow, 53
- labelXaxis
  - plotGraph, 30
- labelYaxis
  - plotGraph, 31
- lbl
  - LCD, 21
- lcd
  - MainWindow, 26
- lcd.cpp, 63
- lcd.h, 64
- linkData
  - plotGraph, 31
- listPorts
  - SerialPort, 48
- M
  - Flags, 19
- m
  - Flags, 19
- MAX
  - Flags, 19
- MIN
  - Flags, 19
- MODE
  - SerialPort, 51
- main
  - main.cpp, 65
- main.cpp, 65
  - main, 65
- mainToolBar
  - Ui\_MainWindow, 54
- MainWindow, 21
  - ~MainWindow, 24
  - addData, 24
  - counter, 26
  - graph, 26
  - label, 26
  - lcd, 26
  - MainWindow, 23
  - MainWindow, 23
  - maxData, 26
  - minData, 27
  - newData, 27
  - on\_connectButton\_clicked, 25
  - on\_disconnectButton\_clicked, 25
  - portPtr, 27
  - rData, 27
  - rawdata, 27
  - resetData, 26
  - scene, 27
  - storeData, 27
  - timeMark, 27
  - timeRunning, 27
  - tmp, 27
  - ui, 27
- mainwindow.cpp, 65
- mainwindow.h, 66
- maxData
  - MainWindow, 26
- menuBar
  - Ui\_MainWindow, 54
- minData
  - MainWindow, 27

- moc\_lcd.cpp
    - QT\_MOC\_LITERAL, 56
    - qt\_meta\_data\_LCD, 56
    - qt\_meta\_stringdata\_LCD, 56
  - moc\_mainwindow.cpp
    - QT\_MOC\_LITERAL, 57
    - qt\_meta\_data\_MainWindow, 57
    - qt\_meta\_stringdata\_MainWindow, 57
  - moc\_plotgraph.cpp
    - QT\_MOC\_LITERAL, 58
    - qt\_meta\_data\_plotGraph, 59
    - qt\_meta\_stringdata\_plotGraph, 59
  - moc\_rs22812.cpp
    - QT\_MOC\_LITERAL, 60
    - qt\_meta\_data\_RS22812, 60
    - qt\_meta\_stringdata\_RS22812, 60
  - moc\_serialport.cpp
    - QT\_MOC\_LITERAL, 61
    - qt\_meta\_data\_SerialPort, 62
    - qt\_meta\_stringdata\_SerialPort, 62
  - mode
    - RS22812, 45
  - modeChanged
    - RS22812, 43
- n
  - Flags, 19
- newData
  - MainWindow, 27
  - RS22812, 43
- newMode
  - RS22812, 44
- newValue
  - RS22812, 44
- nx
  - plotGraph, 36
- ny
  - plotGraph, 36
- Ohms
  - Flags, 19
- oldflags
  - RS22812, 45
- on\_connectButton\_clicked
  - MainWindow, 25
- on\_disconnectButton\_clicked
  - MainWindow, 25
- openPort
  - SerialPort, 48
- operator==
  - Flags, 18
- origin
  - plotGraph, 36
- PARITY
  - SerialPort, 51
- paint
  - plotGraph, 31
- paintAxis
  - plotGraph, 33
- paintEvent
  - LCD, 21
- percent
  - Flags, 19
- plotData
  - plotGraph, 34
- plotGraph, 28
  - axisMargin, 36
  - bRect, 36
  - boundingRect, 30
  - data, 36
  - labelXaxis, 30
  - labelYaxis, 31
  - linkData, 31
  - nx, 36
  - ny, 36
  - origin, 36
  - paint, 31
  - paintAxis, 33
  - plotData, 34
  - plotGraph, 30
  - plotGraph, 30
  - real2Coord, 34
  - rightX, 36
  - scene, 36
  - setScene, 34
  - setUnit, 35
  - setXaxis, 35
  - setXsticks, 35
  - setYaxis, 36
  - setYsticks, 36
  - unit, 36
  - upperY, 37
  - xmax, 37
  - xmin, 37
  - ymax, 37
  - ymin, 37
- plotgraph.cpp, 67
- plotgraph.h, 67
- portPtr
  - MainWindow, 27
- ports
  - SerialPort, 51
- QT\_MOC\_LITERAL
  - moc\_lcd.cpp, 56
  - moc\_mainwindow.cpp, 57
  - moc\_plotgraph.cpp, 58
  - moc\_rs22812.cpp, 60
  - moc\_serialport.cpp, 61
- qt\_meta\_data\_LCD
  - moc\_lcd.cpp, 56
- qt\_meta\_data\_MainWindow
  - moc\_mainwindow.cpp, 57
- qt\_meta\_data\_RS22812
  - moc\_rs22812.cpp, 60
- qt\_meta\_data\_SerialPort
  - moc\_serialport.cpp, 62

- qt\_meta\_data\_plotGraph
  - moc\_plotgraph.cpp, 59
- qt\_meta\_stringdata\_LCD
  - moc\_lcd.cpp, 56
- qt\_meta\_stringdata\_LCD\_t, 37
  - data, 37
  - stringdata0, 37
- qt\_meta\_stringdata\_MainWindow
  - moc\_mainwindow.cpp, 57
- qt\_meta\_stringdata\_MainWindow\_t, 37
  - data, 37
  - stringdata0, 37
- qt\_meta\_stringdata\_RS22812
  - moc\_rs22812.cpp, 60
- qt\_meta\_stringdata\_RS22812\_t, 38
  - data, 38
  - stringdata0, 38
- qt\_meta\_stringdata\_SerialPort
  - moc\_serialport.cpp, 62
- qt\_meta\_stringdata\_SerialPort\_t, 38
  - data, 38
  - stringdata0, 38
- qt\_meta\_stringdata\_plotGraph
  - moc\_plotgraph.cpp, 59
- qt\_meta\_stringdata\_plotGraph\_t, 38
  - data, 38
  - stringdata0, 38
- rData
  - MainWindow, 27
- README.md, 68
- REL
  - Flags, 19
- RS22812, 39
  - byte2Digit, 41
  - digits, 45
  - flags, 45
  - getDigitString, 42
  - getFlags, 42
  - getMode, 42
  - getVal, 43
  - mode, 45
  - modeChanged, 43
  - newData, 43
  - newMode, 44
  - newValue, 44
  - oldflags, 45
  - RS22812, 41
  - resetFlags, 45
  - RS22812, 41
- RS232
  - Flags, 19
- rawdata
  - MainWindow, 27
- readConnect
  - SerialPort, 51
- readPort
  - SerialPort, 50
- ready
  - SerialPort, 50
- readyRead
  - SerialPort, 51
- real2Coord
  - plotGraph, 34
- resetData
  - MainWindow, 26
- resetFlags
  - RS22812, 45
- retranslateUi
  - Ui\_MainWindow, 53
- rightX
  - plotGraph, 36
- rs22812.cpp, 68
- rs22812.h, 68
- s
  - Flags, 19
- STOPBITS
  - SerialPort, 51
- scene
  - MainWindow, 27
  - plotGraph, 36
- SerialPort, 46
  - ~SerialPort, 47
  - activePort, 51
  - BAUDRATE, 51
  - buffer, 51
  - closePort, 48
  - DATABITS, 51
  - isOpen, 51
  - listPorts, 48
  - MODE, 51
  - openPort, 48
  - PARITY, 51
  - ports, 51
  - readConnect, 51
  - readPort, 50
  - ready, 50
  - readyRead, 51
  - STOPBITS, 51
  - SerialPort, 47
  - SerialPort, 47
- serialport.cpp, 69
- serialport.h, 69
- setScene
  - plotGraph, 34
- setUnit
  - plotGraph, 35
- setXaxis
  - plotGraph, 35
- setXsticks
  - plotGraph, 35
- setYaxis
  - plotGraph, 36
- setYsticks
  - plotGraph, 36
- setUpUi
  - Ui\_MainWindow, 53

- statusBar
  - Ui\_MainWindow, [54](#)
- storeData
  - MainWindow, [27](#)
- stringdata0
  - qt\_meta\_stringdata\_LCD\_t, [37](#)
  - qt\_meta\_stringdata\_MainWindow\_t, [37](#)
  - qt\_meta\_stringdata\_plotGraph\_t, [38](#)
  - qt\_meta\_stringdata\_RS22812\_t, [38](#)
  - qt\_meta\_stringdata\_SerialPort\_t, [38](#)
- tilde
  - Flags, [19](#)
- timeMark
  - MainWindow, [27](#)
- timeRunning
  - MainWindow, [27](#)
- tmp
  - MainWindow, [27](#)
- toolBar
  - Ui\_MainWindow, [54](#)
- u
  - Flags, [19](#)
- Ui, [15](#)
- ui
  - MainWindow, [27](#)
- Ui::MainWindow, [27](#)
- Ui\_MainWindow, [52](#)
  - centralWidget, [53](#)
  - comboBoxPort, [53](#)
  - connectButton, [53](#)
  - disconnectButton, [53](#)
  - graphPlot, [53](#)
  - gridLayout, [53](#)
  - horizontalLayout, [53](#)
  - labelPort, [53](#)
  - mainToolBar, [54](#)
  - menuBar, [54](#)
  - retranslateUi, [53](#)
  - setupUi, [53](#)
  - statusBar, [54](#)
  - toolBar, [54](#)
  - verticalLayout, [54](#)
  - verticalLayout\_2, [54](#)
- unit
  - plotGraph, [36](#)
- upperY
  - plotGraph, [37](#)
- V
  - Flags, [19](#)
- verticalLayout
  - Ui\_MainWindow, [54](#)
- verticalLayout\_2
  - Ui\_MainWindow, [54](#)
- xmax
  - plotGraph, [37](#)
- xmin
  - plotGraph, [37](#)
- ymax
  - plotGraph, [37](#)
- ymin
  - plotGraph, [37](#)