

L.A.M.P - DevKit 8000

Genereret af Doxygen 1.8.11

## Indhold

<b>1 Modul-indeks</b>	<b>2</b>
1.1 Moduler . . . . .	2
<b>2 Namespace-indeks</b>	<b>2</b>
2.1 Oversigt over namespaces . . . . .	2
<b>3 Hierarkisk indeks</b>	<b>2</b>
3.1 Klassehierarki . . . . .	2
<b>4 Indeks over datastrukturer</b>	<b>3</b>
4.1 Datastrukturer . . . . .	3
<b>5 Fil-indeks</b>	<b>4</b>
5.1 Filoversigt . . . . .	4
<b>6 Modul-dokumentation</b>	<b>4</b>
6.1 Length constant . . . . .	4
6.1.1 Detaljeret beskrivelse . . . . .	5
6.1.2 #Define-dokumentation . . . . .	5
<b>7 Namespace-dokumentation</b>	<b>5</b>
7.1 Ui namespace-reference . . . . .	5
<b>8 Datastruktur-dokumentation</b>	<b>5</b>
8.1 E3PJR Klasse-reference . . . . .	5
8.1.1 Detaljeret beskrivelse . . . . .	6
8.1.2 Dokumentation af konstruktører og destruktører . . . . .	7
8.1.3 Dokumentation af medlemsfunktioner . . . . .	7
8.1.4 Felt-dokumentation . . . . .	7
8.2 E3PJR Klasse-reference . . . . .	7
8.2.1 Detaljeret beskrivelse . . . . .	10
8.2.2 Dokumentation af medlemsfunktioner . . . . .	10
8.2.3 Felt-dokumentation . . . . .	13

8.3	Light Klasse-reference . . . . .	16
8.3.1	Detaljeret beskrivelse . . . . .	17
8.3.2	Dokumentation af konstruktører og destruktører . . . . .	17
8.4	MainDisplay Klasse-reference . . . . .	17
8.4.1	Detaljeret beskrivelse . . . . .	20
8.4.2	Dokumentation af konstruktører og destruktører . . . . .	20
8.4.3	Dokumentation af medlemsfunktioner . . . . .	20
8.4.4	Felt-dokumentation . . . . .	21
8.5	Planner Klasse-reference . . . . .	22
8.5.1	Detaljeret beskrivelse . . . . .	22
8.5.2	Dokumentation af konstruktører og destruktører . . . . .	23
8.6	PlannerDialog Klasse-reference . . . . .	23
8.6.1	Detaljeret beskrivelse . . . . .	24
8.6.2	Dokumentation af konstruktører og destruktører . . . . .	25
8.6.3	Dokumentation af medlemsfunktioner . . . . .	25
8.6.4	Felt-dokumentation . . . . .	25
8.7	QDialog Klasse-reference . . . . .	25
8.8	QTabWidget Klasse-reference . . . . .	26
8.9	QVirtualKeyboard Klasse-reference . . . . .	27
8.9.1	Detaljeret beskrivelse . . . . .	30
8.9.2	Dokumentation af konstruktører og destruktører . . . . .	30
8.9.3	Dokumentation af medlemsfunktioner . . . . .	30
8.9.4	Felt-dokumentation . . . . .	31
8.10	QWidget Klasse-reference . . . . .	34
8.11	SPIapi Klasse-reference . . . . .	35
8.11.1	Detaljeret beskrivelse . . . . .	35
8.11.2	Dokumentation af konstruktører og destruktører . . . . .	36
8.11.3	Dokumentation af medlemsfunktioner . . . . .	36
8.12	SpiTestProgram Klasse-reference . . . . .	36
8.12.1	Detaljeret beskrivelse . . . . .	37

8.12.2	Dokumentation af konstruktører og destruktører . . . . .	38
8.12.3	Dokumentation af medlemsfunktioner . . . . .	38
8.12.4	Felt-dokumentation . . . . .	38
8.13	SpiTestProgram Klasse-reference . . . . .	39
8.13.1	Detaljeret beskrivelse . . . . .	41
8.13.2	Dokumentation af medlemsfunktioner . . . . .	41
8.13.3	Felt-dokumentation . . . . .	44
8.14	Ui_E3PJR Klasse-reference . . . . .	46
8.14.1	Detaljeret beskrivelse . . . . .	49
8.14.2	Dokumentation af medlemsfunktioner . . . . .	49
8.14.3	Felt-dokumentation . . . . .	52
8.15	Ui_SpiTestProgram Klasse-reference . . . . .	55
8.15.1	Detaljeret beskrivelse . . . . .	57
8.15.2	Dokumentation af medlemsfunktioner . . . . .	57
8.15.3	Felt-dokumentation . . . . .	60
<b>9</b>	<b>Fil-dokumentation</b>	<b>63</b>
9.1	e3pjr.h filreference . . . . .	63
9.2	hotplug_psoc_spi_device.c filreference . . . . .	63
9.2.1	Funktions-dokumentation . . . . .	64
9.2.2	Variabel-dokumentation . . . . .	66
9.3	light.h filreference . . . . .	67
9.4	maindisplay.h filreference . . . . .	67
9.4.1	Detaljeret beskrivelse . . . . .	68
9.5	planner.h filreference . . . . .	68
9.6	plannerdialog.h filreference . . . . .	68
9.6.1	Detaljeret beskrivelse . . . . .	69
9.7	psoc_spi_cdrv.c filreference . . . . .	69
9.7.1	#Define-dokumentation . . . . .	70
9.7.2	Funktions-dokumentation . . . . .	71
9.7.3	Variabel-dokumentation . . . . .	77

9.8	<a href="#">psoc_spi_dev.c filreference</a>	78
9.8.1	<a href="#">#Define-dokumentation</a>	79
9.8.2	<a href="#">Funktions-dokumentation</a>	79
9.8.3	<a href="#">Variabel-dokumentation</a>	82
9.9	<a href="#">psoc_spi_dev.h filreference</a>	83
9.9.1	<a href="#">Funktions-dokumentation</a>	83
9.10	<a href="#">QVirtualKeyboard.h filreference</a>	86
9.11	<a href="#">spiapi.h filreference</a>	87
9.12	<a href="#">spiapi.h filreference</a>	88
9.12.1	<a href="#">#Define-dokumentation</a>	88
9.13	<a href="#">spitestprogram.h filreference</a>	89
9.14	<a href="#">ui_e3pr.h filreference</a>	89
9.15	<a href="#">ui_sptestprogram.h filreference</a>	90

## 1 Modul-indeks

### 1.1 Moduler

Her er en liste over alle moduler:

<b>Length constant</b>	<b>4</b>
------------------------	----------

## 2 Namespace-indeks

### 2.1 Oversigt over namespaces

Her er en liste over alle namespaces med korte beskrivelser:

<b>Ui</b>	<b>??</b>
-----------	-----------

## 3 Hierarkisk indeks

### 3.1 Klassehierarki

Denne nedarvningsliste er sorteret næsten - men ikke nødvendigvis helt - alfabetisk:

<b>QDialog</b>	<b>25</b>
----------------	-----------

PlannerDialog	23
QTabWidget	26
E3PJR	5
QWidget	34
Light	16
MainDisplay	17
Planner	22
QVirtualKeyboard	27
SpiTestProgram	36
SPlapi	35
Ui_E3PJR	46
E3PJR	7
Ui_SpiTestProgram	55
SpiTestProgram	39

## 4 Indeks over datastrukturer

### 4.1 Datastrukturer

Her er datastrukturerne med korte beskrivelser:

E3PJR	??
E3PJR	??
Light	??
MainDisplay	??
Planner	??
PlannerDialog	??
QDialog	??
QTabWidget	??
QVirtualKeyboard	??
QWidget	??
SPlapi	??
SpiTestProgram	??
SpiTestProgram	??

<a href="#">Ui_E3PJR</a>	??
<a href="#">Ui_SpiTestProgram</a>	??

## 5 Fil-indeks

### 5.1 Filoversigt

Her er en liste over alle filer med korte beskrivelser:

<a href="#">e3pjr.h</a>	??
<a href="#">hotplug_psoc_spi_device.c</a>	??
<a href="#">light.h</a>	??
<a href="#">maindisplay.h</a> Handles all UI-related in maindisplay including all tabs	??
<a href="#">planner.h</a>	??
<a href="#">plannerdialog.h</a> Handles all UI-related in plannerdialog	??
<a href="#">psoc_spi_cdrv.c</a>	??
<a href="#">psoc_spi_dev.c</a>	??
<a href="#">psoc_spi_dev.h</a>	??
<a href="#">QVirtualKeyboard.h</a>	??
<a href="#">Semesterprojekt3/spiapi.h</a>	??
<a href="#">SpiTestProgram/spiapi.h</a>	??
<a href="#">spitestprogram.h</a>	??
<a href="#">ui_e3pjr.h</a>	??
<a href="#">ui_spitestprogram.h</a>	??

## 6 Modul-dokumentation

### 6.1 Length constant

#Defines

- #define [MAXLEN](#) 5

### 6.1.1 Detaljeret beskrivelse

### 6.1.2 #Define-dokumentation

#### 6.1.2.1 #define MAXLEN 5

```
#include <spiapi.h>
```

Set length of buffer

Defineret på linje 24 i filen Semesterprojekt3/spiapi.h.

## 7 Namespace-dokumentation

### 7.1 Ui namespace-reference

Datastrukturer

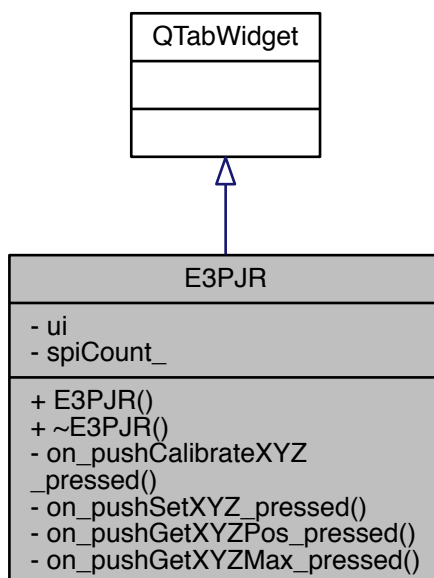
- class [E3PJR](#)
- class [SpiTestProgram](#)

## 8 Datastruktur-dokumentation

### 8.1 E3PJR Klasse-reference

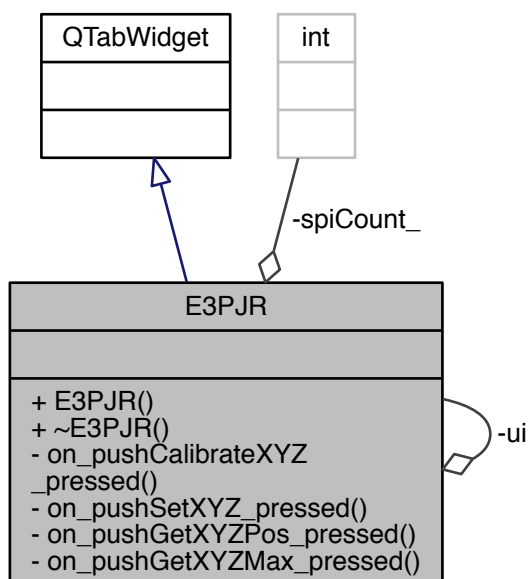
```
#include <e3pjr.h>
```

Stamtræ for E3PJR:





Samarbejdsdiagram for E3PJR:



#### Offentlige metoder

- `E3PJR (QWidget *parent=0)`
- `~E3PJR ()`

#### Private slots

- `void on_pushCalibrateXYZ_pressed ()`
- `void on_pushSetXYZ_pressed ()`
- `void on_pushGetXYZPos_pressed ()`
- `void on_pushGetXYZMax_pressed ()`

#### Private attributter

- `Ui::E3PJR * ui`
- `int spiCount_`

#### 8.1.1 Detaljeret beskrivelse

Defineret på linje 15 i filen `e3pjr.h`.

### 8.1.2 Dokumentation af konstruktører og destruktører

8.1.2.1 **E3PJR**( *QWidget* \* *parent* = 0 ) [explicit]

8.1.2.2 **~E3PJR**( )

### 8.1.3 Dokumentation af medlemsfunktioner

8.1.3.1 void on\_pushCalibrateXYZ\_pressed( ) [private],[slot]

8.1.3.2 void on\_pushGetXYZMax\_pressed( ) [private],[slot]

8.1.3.3 void on\_pushGetXYZPos\_pressed( ) [private],[slot]

8.1.3.4 void on\_pushSetXYZ\_pressed( ) [private],[slot]

### 8.1.4 Felt-dokumentation

8.1.4.1 int spiCount\_ [private]

Defineret på linje 38 i filen e3pjr.h.

8.1.4.2 **Ui::E3PJR**\* ui [private]

Defineret på linje 37 i filen e3pjr.h.

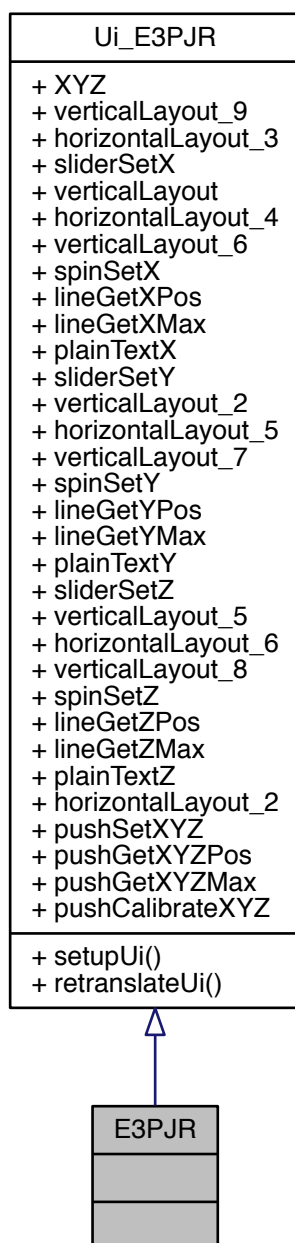
Dokumentationen for denne klasse blev genereret ud fra filen:

- [e3pjr.h](#)

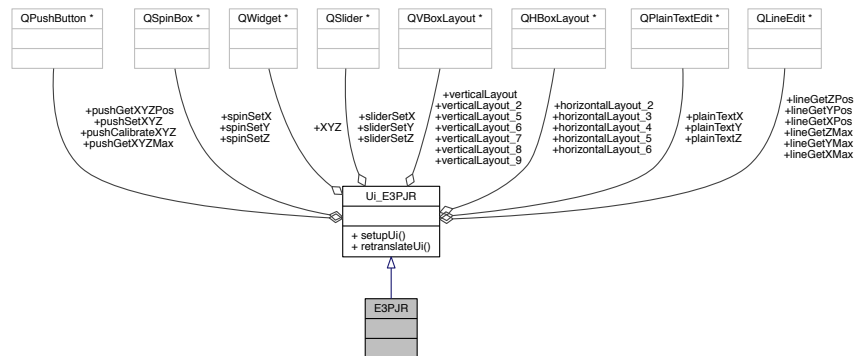
## 8.2 E3PJR Klasse-reference

```
#include <ui_e3pjr.h>
```

Stamtræ for E3PJR:



Samarbejdsdiagram for E3PJR:



### Offentlige metoder

- void [setupUi](#) (QTabWidget \*E3PJR)
- void [retranslateUi](#) (QTabWidget \*E3PJR)

### Datafelter

- QWidget \* [XYZ](#)
- QVBoxLayout \* [verticalLayout\\_9](#)
- QHBoxLayout \* [horizontalLayout\\_3](#)
- QSlider \* [sliderSetX](#)
- QVBoxLayout \* [verticalLayout](#)
- QHBoxLayout \* [horizontalLayout\\_4](#)
- QVBoxLayout \* [verticalLayout\\_6](#)
- QSpinBox \* [spinSetX](#)
- QLineEdit \* [lineGetXPos](#)
- QLineEdit \* [lineGetXMax](#)
- QPlainTextEdit \* [plainTextX](#)
- QSlider \* [sliderSetY](#)
- QVBoxLayout \* [verticalLayout\\_2](#)
- QHBoxLayout \* [horizontalLayout\\_5](#)
- QVBoxLayout \* [verticalLayout\\_7](#)
- QSpinBox \* [spinSetY](#)
- QLineEdit \* [lineGetYPos](#)
- QLineEdit \* [lineGetYMax](#)
- QPlainTextEdit \* [plainTextY](#)
- QSlider \* [sliderSetZ](#)
- QVBoxLayout \* [verticalLayout\\_5](#)
- QHBoxLayout \* [horizontalLayout\\_6](#)
- QVBoxLayout \* [verticalLayout\\_8](#)
- QSpinBox \* [spinSetZ](#)
- QLineEdit \* [lineGetZPos](#)
- QLineEdit \* [lineGetZMax](#)
- QPlainTextEdit \* [plainTextZ](#)
- QHBoxLayout \* [horizontalLayout\\_2](#)
- QPushButton \* [pushSetXYZ](#)
- QPushButton \* [pushGetXYZPos](#)
- QPushButton \* [pushGetXYZMax](#)
- QPushButton \* [pushCalibrateXYZ](#)

## 8.2.1 Detaljeret beskrivelse

Defineret på linje 300 i filen ui\_e3pjr.h.

## 8.2.2 Dokumentation af medlemsfunktioner

### 8.2.2.1 void retranslateUi ( QTabWidget \* *E3PJR* ) [inline],[inherited]

Defineret på linje 281 i filen ui\_e3pjr.h.

Refereret til af Ui\_E3PJR::setupUi().

```

282     {
283         E3PJR->setWindowTitle(QApplication::translate("E3PJR", "TabWidget", 0, QApplication::UnicodeUTF8));
284         lineGetXPos->setPlaceholderText(QApplication::translate("E3PJR", "xPos", 0,
QApplication::UnicodeUTF8));
285         lineGetXMax->setPlaceholderText(QApplication::translate("E3PJR", "xMax", 0,
QApplication::UnicodeUTF8));
286         lineGetYPos->setPlaceholderText(QApplication::translate("E3PJR", "yPos", 0,
QApplication::UnicodeUTF8));
287         lineGetYMax->setPlaceholderText(QApplication::translate("E3PJR", "yMax", 0,
QApplication::UnicodeUTF8));
288         lineGetZPos->setPlaceholderText(QApplication::translate("E3PJR", "zPos", 0,
QApplication::UnicodeUTF8));
289         lineGetZMax->setPlaceholderText(QApplication::translate("E3PJR", "zMax", 0,
QApplication::UnicodeUTF8));
290         pushSetXYZ->setText(QApplication::translate("E3PJR", "SetXYZPos", 0,
QApplication::UnicodeUTF8));
291         pushGetXYZPos->setText(QApplication::translate("E3PJR", "GetXYZPos", 0,
QApplication::UnicodeUTF8));
292         pushGetXYZMax->setText(QApplication::translate("E3PJR", "GetXYZMax", 0,
QApplication::UnicodeUTF8));
293         pushCalibrateXYZ->setText(QApplication::translate("E3PJR", "CalibrateXYZ", 0,
QApplication::UnicodeUTF8));
294         E3PJR->setTabText(E3PJR->indexOf(XYZ), QApplication::translate("E3PJR", "XYZ", 0,
QApplication::UnicodeUTF8));
295     } // retranslateUi

```

Her er kalder-grafen for denne funktion:



### 8.2.2.2 void setupUi ( QTabWidget \* *E3PJR* ) [inline],[inherited]

Defineret på linje 65 i filen ui\_e3pjr.h.

Indeholder referencer til Ui\_E3PJR::retranslateUi().

```

66     {
67         if (E3PJR->objectName().isEmpty())
68             E3PJR->setObjectName(QString::fromUtf8("E3PJR"));
69         E3PJR->resize(480, 278);
70         XYZ = new QWidget();
71         XYZ->setObjectName(QString::fromUtf8("XYZ"));
72         verticalLayout_9 = new QVBoxLayout(XYZ);
73         verticalLayout_9->setObjectName(QString::fromUtf8("verticalLayout_9"));
74         horizontalLayout_3 = new QHBoxLayout();
75         horizontalLayout_3->setObjectName(QString::fromUtf8("horizontalLayout_3"));
76         sliderSetX = new QSlider(XYZ);
77         sliderSetX->setObjectName(QString::fromUtf8("sliderSetX"));
78         sliderSetX->setMaximum(255);
79         sliderSetX->setSingleStep(5);
80         sliderSetX->setPageStep(15);
81         sliderSetX->setOrientation(Qt::Vertical);
82
83         horizontalLayout_3->addWidget(sliderSetX);
84
85         verticalLayout = new QVBoxLayout();
86         verticalLayout->setObjectName(QString::fromUtf8("verticalLayout"));
87         horizontalLayout_4 = new QHBoxLayout();
88         horizontalLayout_4->setObjectName(QString::fromUtf8("horizontalLayout_4"));
89         verticalLayout_6 = new QVBoxLayout();
90         verticalLayout_6->setObjectName(QString::fromUtf8("verticalLayout_6"));
91         spinSetX = new QSpinBox(XYZ);
92         spinSetX->setObjectName(QString::fromUtf8("spinSetX"));
93         spinSetX->setAlignment(Qt::AlignCenter);
94         spinSetX->setMaximum(255);
95         spinSetX->setSingleStep(5);
96
97         verticalLayout_6->addWidget(spinSetX);
98
99         lineGetXPos = new QLineEdit(XYZ);
100         lineGetXPos->setObjectName(QString::fromUtf8("lineGetXPos"));
101         lineGetXPos->setAlignment(Qt::AlignCenter);
102
103         verticalLayout_6->addWidget(lineGetXPos);
104
105         lineGetXMax = new QLineEdit(XYZ);
106         lineGetXMax->setObjectName(QString::fromUtf8("lineGetXMax"));
107         lineGetXMax->setAlignment(Qt::AlignCenter);
108
109         verticalLayout_6->addWidget(lineGetXMax);
110
111
112         horizontalLayout_4->addLayout(verticalLayout_6);
113
114
115         verticalLayout->addLayout(horizontalLayout_4);
116
117         plainTextX = new QPlainTextEdit(XYZ);
118         plainTextX->setObjectName(QString::fromUtf8("plainTextX"));
119         plainTextX->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
120         plainTextX->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
121         plainTextX->setUndoRedoEnabled(false);
122         plainTextX->setReadOnly(true);
123         plainTextX->setOverwriteMode(false);
124
125         verticalLayout->addWidget(plainTextX);
126
127
128         horizontalLayout_3->addLayout(verticalLayout);
129
130         sliderSetY = new QSlider(XYZ);
131         sliderSetY->setObjectName(QString::fromUtf8("sliderSetY"));
132         sliderSetY->setMaximum(255);
133         sliderSetY->setSingleStep(5);
134         sliderSetY->setPageStep(15);
135         sliderSetY->setOrientation(Qt::Vertical);
136
137         horizontalLayout_3->addWidget(sliderSetY);
138
139         verticalLayout_2 = new QVBoxLayout();
140         verticalLayout_2->setObjectName(QString::fromUtf8("verticalLayout_2"));
141         horizontalLayout_5 = new QHBoxLayout();
142         horizontalLayout_5->setObjectName(QString::fromUtf8("horizontalLayout_5"));
143         verticalLayout_7 = new QVBoxLayout();
144         verticalLayout_7->setObjectName(QString::fromUtf8("verticalLayout_7"));
145         spinSetY = new QSpinBox(XYZ);
146         spinSetY->setObjectName(QString::fromUtf8("spinSetY"));
147         spinSetY->setAlignment(Qt::AlignCenter);
148         spinSetY->setMaximum(255);
149         spinSetY->setSingleStep(5);
150
151         verticalLayout_7->addWidget(spinSetY);
152

```

```
153     lineGetYPos = new QLineEdit(XYZ);
154     lineGetYPos->setObjectName(QString::fromUtf8("lineGetYPos"));
155     lineGetYPos->setAlignment(Qt::AlignCenter);
156
157     verticalLayout_7->addWidget(lineGetYPos);
158
159     lineGetYMax = new QLineEdit(XYZ);
160     lineGetYMax->setObjectName(QString::fromUtf8("lineGetYMax"));
161     lineGetYMax->setAlignment(Qt::AlignCenter);
162
163     verticalLayout_7->addWidget(lineGetYMax);
164
165
166     horizontalLayout_5->addLayout(verticalLayout_7);
167
168
169     verticalLayout_2->addLayout(horizontalLayout_5);
170
171     plainTextY = new QPlainTextEdit(XYZ);
172     plainTextY->setObjectName(QString::fromUtf8("plainTextY"));
173     plainTextY->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
174     plainTextY->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
175     plainTextY->setUndoRedoEnabled(false);
176     plainTextY->setReadOnly(true);
177
178     verticalLayout_2->addWidget(plainTextY);
179
180
181     horizontalLayout_3->addLayout(verticalLayout_2);
182
183     sliderSetZ = new QSlider(XYZ);
184     sliderSetZ->setObjectName(QString::fromUtf8("sliderSetZ"));
185     sliderSetZ->setMaximum(255);
186     sliderSetZ->setSingleStep(5);
187     sliderSetZ->setPageStep(15);
188     sliderSetZ->setOrientation(Qt::Vertical);
189
190     horizontalLayout_3->addWidget(sliderSetZ);
191
192     verticalLayout_5 = new QVBoxLayout();
193     verticalLayout_5->setObjectName(QString::fromUtf8("verticalLayout_5"));
194     horizontalLayout_6 = new QHBoxLayout();
195     horizontalLayout_6->setObjectName(QString::fromUtf8("horizontalLayout_6"));
196     verticalLayout_8 = new QVBoxLayout();
197     verticalLayout_8->setObjectName(QString::fromUtf8("verticalLayout_8"));
198     spinSetZ = new QSpinBox(XYZ);
199     spinSetZ->setObjectName(QString::fromUtf8("spinSetZ"));
200     spinSetZ->setAlignment(Qt::AlignCenter);
201     spinSetZ->setMaximum(255);
202     spinSetZ->setSingleStep(5);
203
204     verticalLayout_8->addWidget(spinSetZ);
205
206     lineGetZPos = new QLineEdit(XYZ);
207     lineGetZPos->setObjectName(QString::fromUtf8("lineGetZPos"));
208     lineGetZPos->setAlignment(Qt::AlignCenter);
209
210     verticalLayout_8->addWidget(lineGetZPos);
211
212     lineGetZMax = new QLineEdit(XYZ);
213     lineGetZMax->setObjectName(QString::fromUtf8("lineGetZMax"));
214     lineGetZMax->setAlignment(Qt::AlignCenter);
215
216     verticalLayout_8->addWidget(lineGetZMax);
217
218
219     horizontalLayout_6->addLayout(verticalLayout_8);
220
221
222     verticalLayout_5->addLayout(horizontalLayout_6);
223
224     plainTextZ = new QPlainTextEdit(XYZ);
225     plainTextZ->setObjectName(QString::fromUtf8("plainTextZ"));
226     plainTextZ->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
227     plainTextZ->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
228     plainTextZ->setUndoRedoEnabled(false);
229     plainTextZ->setReadOnly(true);
230
231     verticalLayout_5->addWidget(plainTextZ);
232
233
234     horizontalLayout_3->addLayout(verticalLayout_5);
235
236
237     verticalLayout_9->addLayout(horizontalLayout_3);
238
239     horizontalLayout_2 = new QHBoxLayout();
```

```

240     horizontalLayout_2->setObjectName(QString::fromUtf8("horizontalLayout_2"));
241     pushSetXYZ = new QPushButton(XYZ);
242     pushSetXYZ->setObjectName(QString::fromUtf8("pushSetXYZ"));
243
244     horizontalLayout_2->addWidget(pushSetXYZ);
245
246     pushGetXYZPos = new QPushButton(XYZ);
247     pushGetXYZPos->setObjectName(QString::fromUtf8("pushGetXYZPos"));
248
249     horizontalLayout_2->addWidget(pushGetXYZPos);
250
251     pushGetXYZMax = new QPushButton(XYZ);
252     pushGetXYZMax->setObjectName(QString::fromUtf8("pushGetXYZMax"));
253
254     horizontalLayout_2->addWidget(pushGetXYZMax);
255
256     pushCalibrateXYZ = new QPushButton(XYZ);
257     pushCalibrateXYZ->setObjectName(QString::fromUtf8("pushCalibrateXYZ"));
258
259     horizontalLayout_2->addWidget(pushCalibrateXYZ);
260
261
262     verticalLayout_9->addLayout(horizontalLayout_2);
263
264     E3PJR->addTab(XYZ, QString());
265     QWidget::setTabOrder(spinSetY, spinSetZ);
266
267     retranslateUi(E3PJR);
268     QObject::connect(sliderSetY, SIGNAL(valueChanged(int)),
spinSetY, SLOT(setValue(int)));
269     QObject::connect(spinSetY, SIGNAL(valueChanged(int)), sliderSetY, SLOT(setValue(
int)));
270     QObject::connect(sliderSetZ, SIGNAL(valueChanged(int)),
spinSetZ, SLOT(setValue(int)));
271     QObject::connect(spinSetZ, SIGNAL(valueChanged(int)), sliderSetZ, SLOT(setValue(
int)));
272     QObject::connect(sliderSetX, SIGNAL(valueChanged(int)),
spinSetX, SLOT(setValue(int)));
273     QObject::connect(spinSetX, SIGNAL(valueChanged(int)), sliderSetX, SLOT(setValue(
int)));
274
275     E3PJR->setCurrentIndex(0);
276
277
278     QMetaObject::connectSlotsByName(E3PJR);
279 } // setupUi

```

Her er kald-grafen for denne funktion:



## 8.2.3 Felt-dokumentation

### 8.2.3.1 QHBoxLayout\* horizontalLayout\_2 [inherited]

Defineret på linje 59 i filen ui\_e3pjr.h.

### 8.2.3.2 QHBoxLayout\* horizontalLayout\_3 [inherited]

Defineret på linje 34 i filen ui\_e3pjr.h.



**8.2.3.3 QHBoxLayout\* horizontalLayout\_4** [inherited]

Defineret på linje 37 i filen ui\_e3pjr.h.

**8.2.3.4 QHBoxLayout\* horizontalLayout\_5** [inherited]

Defineret på linje 45 i filen ui\_e3pjr.h.

**8.2.3.5 QHBoxLayout\* horizontalLayout\_6** [inherited]

Defineret på linje 53 i filen ui\_e3pjr.h.

**8.2.3.6 QLineEdit\* lineGetXMax** [inherited]

Defineret på linje 41 i filen ui\_e3pjr.h.

**8.2.3.7 QLineEdit\* lineGetXPos** [inherited]

Defineret på linje 40 i filen ui\_e3pjr.h.

**8.2.3.8 QLineEdit\* lineGetYMax** [inherited]

Defineret på linje 49 i filen ui\_e3pjr.h.

**8.2.3.9 QLineEdit\* lineGetYPos** [inherited]

Defineret på linje 48 i filen ui\_e3pjr.h.

**8.2.3.10 QLineEdit\* lineGetZMax** [inherited]

Defineret på linje 57 i filen ui\_e3pjr.h.

**8.2.3.11 QLineEdit\* lineGetZPos** [inherited]

Defineret på linje 56 i filen ui\_e3pjr.h.

**8.2.3.12 QPlainTextEdit\* plainTextX** [inherited]

Defineret på linje 42 i filen ui\_e3pjr.h.

**8.2.3.13 QPlainTextEdit\* plainTextY** [inherited]

Defineret på linje 50 i filen ui\_e3pjr.h.

**8.2.3.14 QPlainTextEdit\* plainTextZ** [inherited]

Defineret på linje 58 i filen ui\_e3pjr.h.

**8.2.3.15 QPushButton\* pushCalibrateXYZ** [inherited]

Defineret på linje 63 i filen ui\_e3pjr.h.

**8.2.3.16 QPushButton\* pushGetXYZMax** [inherited]

Defineret på linje 62 i filen ui\_e3pjr.h.

**8.2.3.17 QPushButton\* pushGetXYZPos** [inherited]

Defineret på linje 61 i filen ui\_e3pjr.h.

**8.2.3.18 QPushButton\* pushSetXYZ** [inherited]

Defineret på linje 60 i filen ui\_e3pjr.h.

**8.2.3.19 QSlider\* sliderSetX** [inherited]

Defineret på linje 35 i filen ui\_e3pjr.h.

**8.2.3.20 QSlider\* sliderSetY** [inherited]

Defineret på linje 43 i filen ui\_e3pjr.h.

**8.2.3.21 QSlider\* sliderSetZ** [inherited]

Defineret på linje 51 i filen ui\_e3pjr.h.

**8.2.3.22 QSpinBox\* spinSetX** [inherited]

Defineret på linje 39 i filen ui\_e3pjr.h.

**8.2.3.23 QSpinBox\* spinSetY** [inherited]

Defineret på linje 47 i filen ui\_e3pjr.h.

**8.2.3.24 QSpinBox\* spinSetZ** [inherited]

Defineret på linje 55 i filen ui\_e3pjr.h.

**8.2.3.25 QVBoxLayout\* verticalLayout** [inherited]

Defineret på linje 36 i filen ui\_e3pjr.h.

**8.2.3.26 QVBoxLayout\* verticalLayout\_2** [inherited]

Defineret på linje 44 i filen ui\_e3pjr.h.

**8.2.3.27** `QVBoxLayout* verticalLayout_5` `[inherited]`

Defineret på linje 52 i filen `ui_e3pjr.h`.

**8.2.3.28** `QVBoxLayout* verticalLayout_6` `[inherited]`

Defineret på linje 38 i filen `ui_e3pjr.h`.

**8.2.3.29** `QVBoxLayout* verticalLayout_7` `[inherited]`

Defineret på linje 46 i filen `ui_e3pjr.h`.

**8.2.3.30** `QVBoxLayout* verticalLayout_8` `[inherited]`

Defineret på linje 54 i filen `ui_e3pjr.h`.

**8.2.3.31** `QVBoxLayout* verticalLayout_9` `[inherited]`

Defineret på linje 33 i filen `ui_e3pjr.h`.

**8.2.3.32** `QWidget* XYZ` `[inherited]`

Defineret på linje 32 i filen `ui_e3pjr.h`.

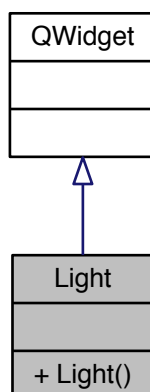
Dokumentationen for denne klasse blev genereret ud fra filen:

- [ui\\_e3pjr.h](#)

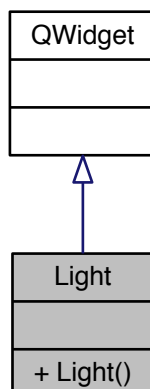
### 8.3 Light Klasse-reference

```
#include <light.h>
```

Stamtræ for Light:



Samarbejdsdiagram for Light:



Offentlige metoder

- [Light \(QWidget \\*parent=0\)](#)

#### 8.3.1 Detaljeret beskrivelse

Defineret på linje 7 i filen `light.h`.

#### 8.3.2 Dokumentation af konstruktører og destruktører

##### 8.3.2.1 `Light ( QWidget * parent = 0 ) [explicit]`

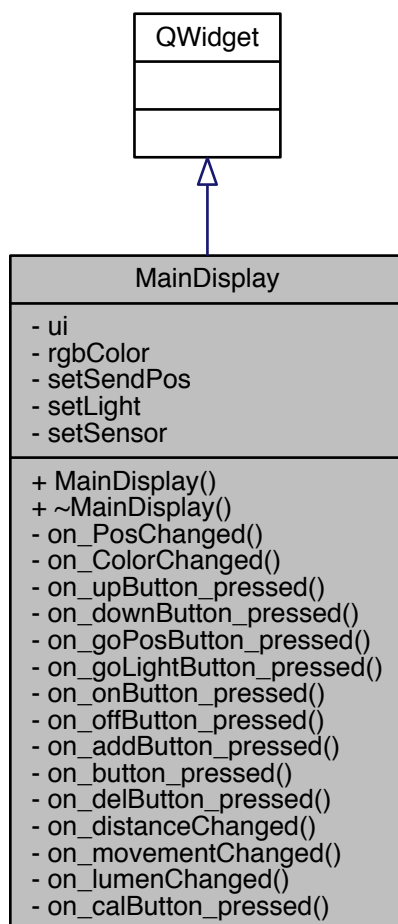
Dokumentationen for denne klasse blev genereret ud fra filen:

- [light.h](#)

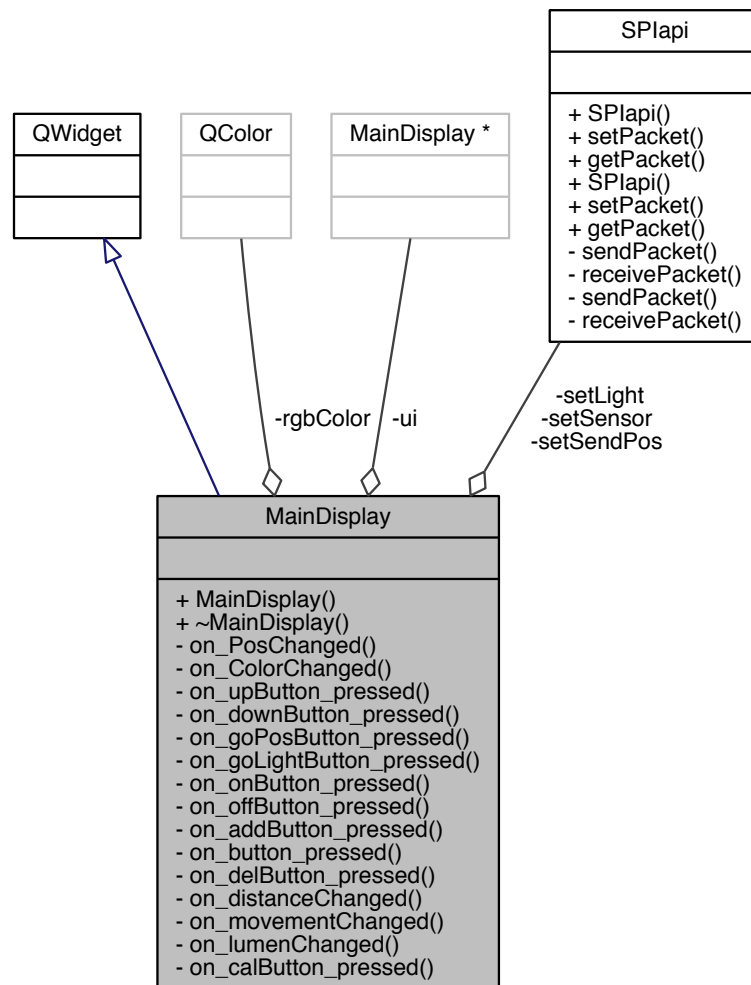
## 8.4 MainDisplay Klasse-reference

```
#include <maindisplay.h>
```

Stamtræ for MainDisplay:



Samarbejdsdiagram for MainDisplay:



#### Offentlige metoder

- [MainDisplay \(QWidget \\*parent=0\)](#)
- [~MainDisplay \(\)](#)

#### Private slots

- void [on\\_PosChanged \(\)](#)
- void [on\\_ColorChanged \(\)](#)
- void [on\\_upButton\\_pressed \(\)](#)
- void [on\\_downButton\\_pressed \(\)](#)
- void [on\\_goPosButton\\_pressed \(\)](#)
- void [on\\_goLightButton\\_pressed \(\)](#)
- void [on\\_onButton\\_pressed \(\)](#)
- void [on\\_offButton\\_pressed \(\)](#)

- void [on\\_addButton\\_pressed](#) ()
- void [on\\_button\\_pressed](#) ()
- void [on\\_delButton\\_pressed](#) ()
- void [on\\_distanceChanged](#) ()
- void [on\\_movementChanged](#) ()
- void [on\\_lumenChanged](#) ()
- void [on\\_calButton\\_pressed](#) ()

#### Private attributter

- Ui::MainDisplay \* [ui](#)
- QColor [rgbColor](#)
- [SPLapi](#) [setSendPos](#)
- [SPLapi](#) [setLight](#)
- [SPLapi](#) [setSensor](#)

#### 8.4.1 Detaljeret beskrivelse

##### Forfatter

Victor Busk ([201409557@post.au.dk](mailto:201409557@post.au.dk))

Defineret på linje 19 i filen maindisplay.h.

#### 8.4.2 Dokumentation af konstruktører og destruktører

8.4.2.1 **MainDisplay** ( **QWidget** \* *parent* = 0 ) [explicit]

8.4.2.2 **~MainDisplay** ( )

#### 8.4.3 Dokumentation af medlemsfunktioner

8.4.3.1 void **on\_addButton\_pressed** ( ) [private],[slot]

8.4.3.2 void **on\_button\_pressed** ( ) [private],[slot]

8.4.3.3 void **on\_calButton\_pressed** ( ) [private],[slot]

8.4.3.4 void **on\_ColorChanged** ( ) [private],[slot]

8.4.3.5 void **on\_delButton\_pressed** ( ) [private],[slot]

8.4.3.6 void **on\_distanceChanged** ( ) [private],[slot]

8.4.3.7 void **on\_downButton\_pressed** ( ) [private],[slot]

8.4.3.8 void **on\_goLightButton\_pressed** ( ) [private],[slot]

8.4.3.9 void on\_goPosButton\_pressed ( ) [private],[slot]

8.4.3.10 void on\_lumenChanged ( ) [private],[slot]

8.4.3.11 void on\_movementChanged ( ) [private],[slot]

8.4.3.12 void on\_offButton\_pressed ( ) [private],[slot]

8.4.3.13 void on\_onButton\_pressed ( ) [private],[slot]

8.4.3.14 void on\_PosChanged ( ) [private],[slot]

8.4.3.15 void on\_upButton\_pressed ( ) [private],[slot]

#### 8.4.4 Felt-dokumentation

8.4.4.1 QColor rgbColor [private]

Defineret på linje 46 i filen maindisplay.h.

8.4.4.2 SPLapi setLight [private]

Defineret på linje 48 i filen maindisplay.h.

8.4.4.3 SPLapi setSendPos [private]

Defineret på linje 47 i filen maindisplay.h.

8.4.4.4 SPLapi setSensor [private]

Defineret på linje 49 i filen maindisplay.h.

8.4.4.5 Ui::MainDisplay\* ui [private]

Defineret på linje 45 i filen maindisplay.h.

Dokumentationen for denne klasse blev genereret ud fra filen:

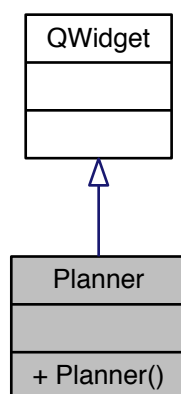
- [maindisplay.h](#)



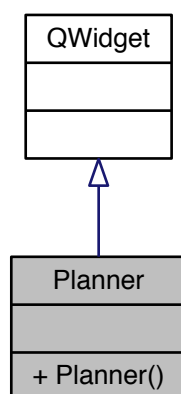
## 8.5 Planner Klasse-reference

```
#include <planner.h>
```

Stamtræ for Planner:



Samarbejdsdiagram for Planner:



Offentlige metoder

- [Planner](#) ([QWidget](#) \*parent=0)

### 8.5.1 Detaljeret beskrivelse

Defineret på linje 6 i filen `planner.h`.

## 8.5.2 Dokumentation af konstruktører og destruktører

8.5.2.1 Planner ( QWidget \* *parent* = 0 ) [explicit]

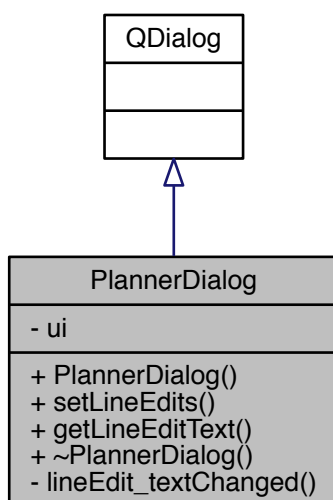
Dokumentationen for denne klasse blev genereret ud fra filen:

- [planner.h](#)

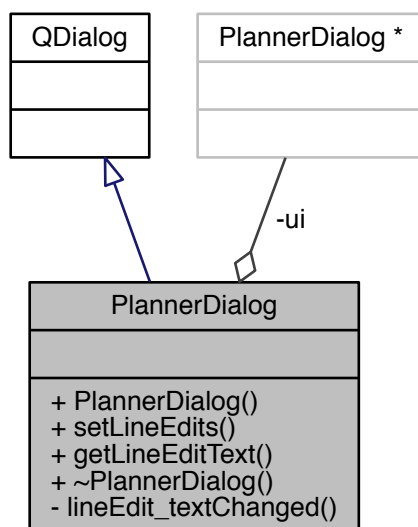
## 8.6 PlannerDialog Klasse-reference

```
#include <plannerdialog.h>
```

Stamtræ for PlannerDialog:



Samarbejdsdiagram for PlannerDialog:



#### Offentlige metoder

- [PlannerDialog](#) ([QWidget](#) \*parent=0)
- void [setLineEdits](#) (QString plan)
- QString [getLineEditText](#) () const
- [~PlannerDialog](#) ()

#### Private slots

- void [lineEdit\\_textChanged](#) ()

#### Private attributter

- Ui::PlannerDialog \* [ui](#)

#### 8.6.1 Detaljeret beskrivelse

##### Forfatter

Victor Busk ([201409557@post.au.dk](mailto:201409557@post.au.dk))

Defineret på linje 18 i filen plannerdialog.h.

### 8.6.2 Dokumentation af konstruktører og destruktører

8.6.2.1 **PlannerDialog** ( **QWidget** \* *parent* = 0 ) [explicit]

8.6.2.2 **~PlannerDialog** ( )

### 8.6.3 Dokumentation af medlemsfunktioner

8.6.3.1 **QString** getLineEditText ( ) const

8.6.3.2 void lineEdit\_textChanged ( ) [private],[slot]

8.6.3.3 void setLineEdits ( **QString** *plan* )

### 8.6.4 Felt-dokumentation

8.6.4.1 **Ui::PlannerDialog\*** ui [private]

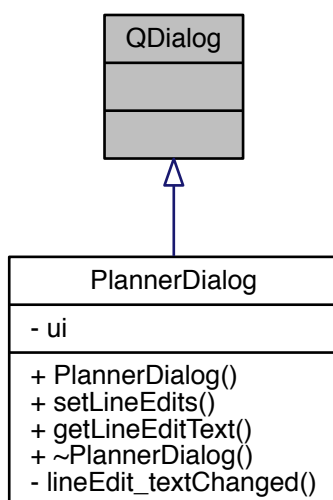
Defineret på linje 34 i filen plannerdialog.h.

Dokumentationen for denne klasse blev genereret ud fra filen:

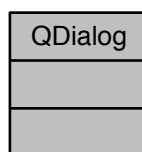
- [plannerdialog.h](#)

## 8.7 QDialog Klasse-reference

Stamtræ for QDialog:



Samarbejdsdiagram for QDialog:

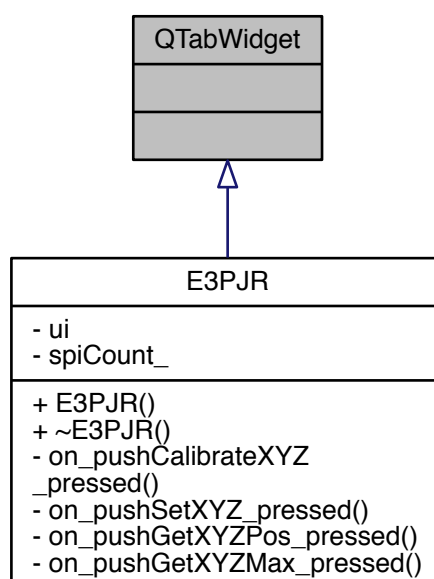


Dokumentationen for denne klasse blev genereret ud fra filen:

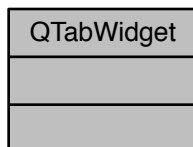
- [plannerdialog.h](#)

## 8.8 QTabWidget Klasse-reference

Stamtræ for QTabWidget:



Samarbejdsdiagram for QTabWidget:



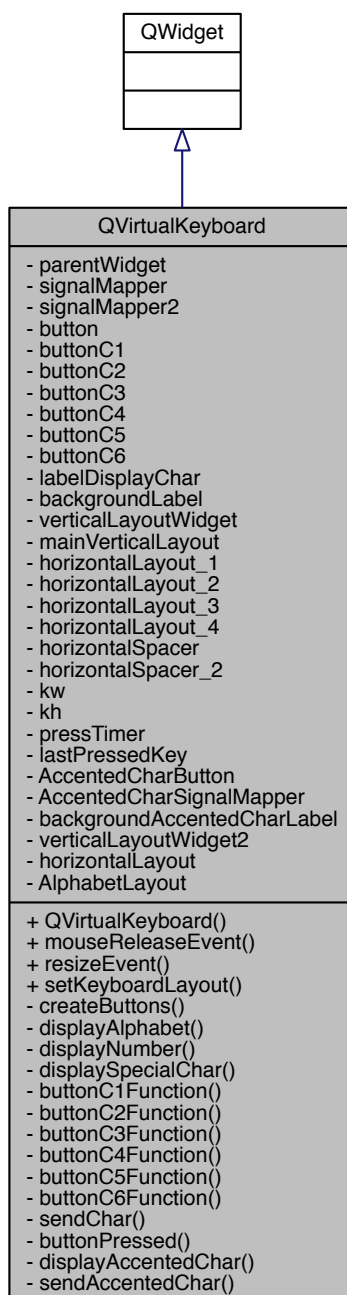
Dokumentationen for denne klasse blev genereret ud fra filen:

- [e3pjr.h](#)

## 8.9 QVirtualKeyboard Klasse-reference

```
#include <QVirtualKeyboard.h>
```

Stamtræ for QVirtualKeyboard:



Samarbejdsdiagram for QVirtualKeyboard:



### Offentlige metoder

- [QVirtualKeyboard](#) (QWidget \*parent=0)
- void [mouseReleaseEvent](#) (QMouseEvent \*event)
- void [resizeEvent](#) (QResizeEvent \*event)
- void [setKeyboardLayout](#) (int layout)

### Private slots

- void [createButtons](#) (void)
- void [displayAlphabet](#) (void)
- void [displayNumber](#) (void)
- void [displaySpecialChar](#) (void)
- void [buttonC1Function](#) (void)
- void [buttonC2Function](#) (void)
- void [buttonC3Function](#) (void)
- void [buttonC4Function](#) (void)
- void [buttonC5Function](#) (void)
- void [buttonC6Function](#) (void)
- void [sendChar](#) (int indexOfCharToSend)
- void [buttonPressed](#) (int indexOfCharToSend)
- void [displayAccentedChar](#) (void)
- void [sendAccentedChar](#) (int indexOfCharToSend)

### Private attributter

- QWidget \* [parentWidget](#)
- QSignalMapper \* [signalMapper](#)
- QSignalMapper \* [signalMapper2](#)
- QVector< QPushButton \* > [button](#)
- QPushButton \* [buttonC1](#)
- QPushButton \* [buttonC2](#)
- QPushButton \* [buttonC3](#)
- QPushButton \* [buttonC4](#)
- QPushButton \* [buttonC5](#)
- QPushButton \* [buttonC6](#)
- QPushButton \* [labelDisplayChar](#)
- QLabel \* [backgroundLabel](#)
- QWidget \* [verticalLayoutWidget](#)



- QVBoxLayout \* [mainVerticalLayout](#)
- QHBoxLayout \* [horizontalLayout\\_1](#)
- QHBoxLayout \* [horizontalLayout\\_2](#)
- QHBoxLayout \* [horizontalLayout\\_3](#)
- QHBoxLayout \* [horizontalLayout\\_4](#)
- QSpacerItem \* [horizontalSpacer](#)
- QSpacerItem \* [horizontalSpacer\\_2](#)
- int [kw](#)
- int [kh](#)
- QTimer \* [pressTimer](#)
- int [lastPressedKey](#)
- QVector< QPushButton \* > [AccentedCharButton](#)
- QSignalMapper \* [AccentedCharSignalMapper](#)
- QLabel \* [backgroundAccentedCharLabel](#)
- QWidget \* [verticalLayoutWidget2](#)
- QHBoxLayout \* [horizontalLayout](#)
- int [AlphabetLayout](#)

### 8.9.1 Detaljeret beskrivelse

Defineret på linje 28 i filen QVirtualKeyboard.h.

### 8.9.2 Dokumentation af konstruktører og destruktører

#### 8.9.2.1 QVirtualKeyboard ( QWidget \* *parent* = 0 )

### 8.9.3 Dokumentation af medlemsfunktioner

8.9.3.1 void buttonC1Function ( void ) [private],[slot]

8.9.3.2 void buttonC2Function ( void ) [private],[slot]

8.9.3.3 void buttonC3Function ( void ) [private],[slot]

8.9.3.4 void buttonC4Function ( void ) [private],[slot]

8.9.3.5 void buttonC5Function ( void ) [private],[slot]

8.9.3.6 void buttonC6Function ( void ) [private],[slot]

8.9.3.7 void buttonPressed ( int *indexOfCharToSend* ) [private],[slot]

8.9.3.8 void createButtons ( void ) [private],[slot]

8.9.3.9 void displayAccentedChar ( void ) [private],[slot]

8.9.3.10 void displayAlphabet ( void ) [private],[slot]

8.9.3.11 void displayNumber ( void ) [private],[slot]

8.9.3.12 `void displaySpecialChar ( void ) [private],[slot]`

8.9.3.13 `void mouseReleaseEvent ( QMouseEvent * event )`

8.9.3.14 `void resizeEvent ( QResizeEvent * event )`

8.9.3.15 `void sendAccentedChar ( int indexOfCharToSend ) [private],[slot]`

8.9.3.16 `void sendChar ( int indexOfCharToSend ) [private],[slot]`

8.9.3.17 `void setKeyboardLayout ( int layout )`

#### 8.9.4 Felt-dokumentation

8.9.4.1 `QVector<QPushButton*> AccentedCharButton [private]`

Defineret på linje 99 i filen QVirtualKeyboard.h.

8.9.4.2 `QSignalMapper* AccentedCharSignalMapper [private]`

Defineret på linje 100 i filen QVirtualKeyboard.h.

8.9.4.3 `int AlphabetLayout [private]`

Defineret på linje 107 i filen QVirtualKeyboard.h.

8.9.4.4 `QLabel* backgroundAccentedCharLabel [private]`

Defineret på linje 102 i filen QVirtualKeyboard.h.

8.9.4.5 `QLabel* backgroundLabel [private]`

Defineret på linje 81 i filen QVirtualKeyboard.h.

8.9.4.6 `QVector<QPushButton*> button [private]`

Defineret på linje 70 i filen QVirtualKeyboard.h.

8.9.4.7 `QPushButton* buttonC1 [private]`

Defineret på linje 72 i filen QVirtualKeyboard.h.

8.9.4.8 `QPushButton* buttonC2 [private]`

Defineret på linje 73 i filen QVirtualKeyboard.h.

8.9.4.9 `QPushButton* buttonC3 [private]`

Defineret på linje 74 i filen QVirtualKeyboard.h.

**8.9.4.10** `QPushButton* buttonC4` `[private]`

Defineret på linje 75 i filen `QVirtualKeyboard.h`.

**8.9.4.11** `QPushButton* buttonC5` `[private]`

Defineret på linje 76 i filen `QVirtualKeyboard.h`.

**8.9.4.12** `QPushButton* buttonC6` `[private]`

Defineret på linje 77 i filen `QVirtualKeyboard.h`.

**8.9.4.13** `QHBoxLayout* horizontalLayout` `[private]`

Defineret på linje 105 i filen `QVirtualKeyboard.h`.

**8.9.4.14** `QHBoxLayout* horizontalLayout_1` `[private]`

Defineret på linje 85 i filen `QVirtualKeyboard.h`.

**8.9.4.15** `QHBoxLayout* horizontalLayout_2` `[private]`

Defineret på linje 86 i filen `QVirtualKeyboard.h`.

**8.9.4.16** `QHBoxLayout* horizontalLayout_3` `[private]`

Defineret på linje 87 i filen `QVirtualKeyboard.h`.

**8.9.4.17** `QHBoxLayout* horizontalLayout_4` `[private]`

Defineret på linje 88 i filen `QVirtualKeyboard.h`.

**8.9.4.18** `QSpacerItem* horizontalSpacer` `[private]`

Defineret på linje 90 i filen `QVirtualKeyboard.h`.

**8.9.4.19** `QSpacerItem* horizontalSpacer_2` `[private]`

Defineret på linje 91 i filen `QVirtualKeyboard.h`.

**8.9.4.20** `int kh` `[private]`

Defineret på linje 94 i filen `QVirtualKeyboard.h`.

**8.9.4.21** `int kw` `[private]`

Defineret på linje 93 i filen `QVirtualKeyboard.h`.

**8.9.4.22 QPushButton\* labelDisplayChar** [private]

Defineret på linje 79 i filen QVirtualKeyboard.h.

**8.9.4.23 int lastPressedKey** [private]

Defineret på linje 98 i filen QVirtualKeyboard.h.

**8.9.4.24 QVBoxLayout\* mainVerticalLayout** [private]

Defineret på linje 84 i filen QVirtualKeyboard.h.

**8.9.4.25 QWidget\* parentWidget** [private]

Defineret på linje 64 i filen QVirtualKeyboard.h.

**8.9.4.26 QTimer\* pressTimer** [private]

Defineret på linje 97 i filen QVirtualKeyboard.h.

**8.9.4.27 QSignalMapper\* signalMapper** [private]

Defineret på linje 67 i filen QVirtualKeyboard.h.

**8.9.4.28 QSignalMapper\* signalMapper2** [private]

Defineret på linje 68 i filen QVirtualKeyboard.h.

**8.9.4.29 QWidget\* verticalLayoutWidget** [private]

Defineret på linje 83 i filen QVirtualKeyboard.h.

**8.9.4.30 QWidget\* verticalLayoutWidget2** [private]

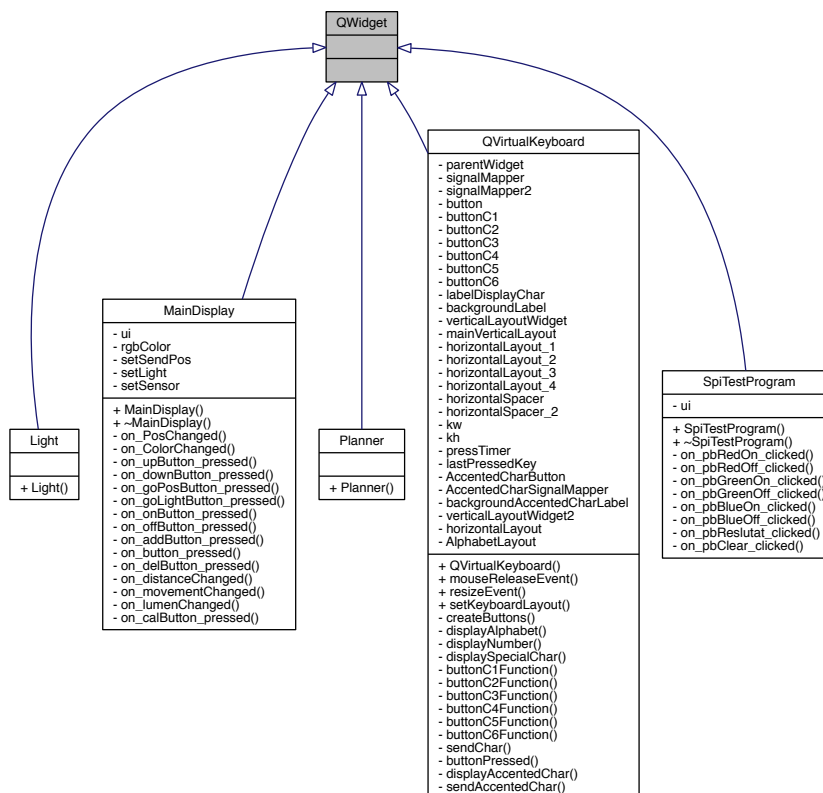
Defineret på linje 104 i filen QVirtualKeyboard.h.

Dokumentationen for denne klasse blev genereret ud fra filen:

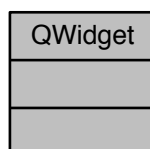
- [QVirtualKeyboard.h](#)

## 8.10 QWidget Klasse-reference

Stamtræ for QWidget:



Samarbejdsdiagram for QWidget:



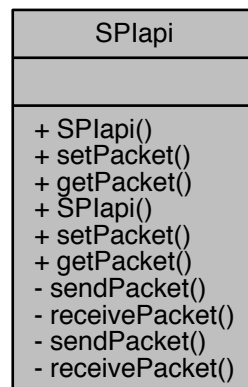
Dokumentationen for denne klasse blev genereret ud fra filen:

- [spitestprogram.h](#)

## 8.11 SPLapi Klasse-reference

```
#include <spiapi.h>
```

Samarbejdsdiagram for SPLapi:



### Offentlige metoder

- `SPLapi ()`
- `int setPacket (const unsigned char *cmd, unsigned char *value) const`
- `int getPacket (unsigned char *cmd, unsigned int *value)`
- `SPLapi ()`
- `int setPacket (unsigned char *cmd, unsigned char *value) const`
- `int getPacket (unsigned char *cmd, unsigned int *value)`

### Private metoder

- `int sendPacket (const unsigned char *cmd, unsigned char *data) const`
- `int receivePacket (unsigned char *cmd, unsigned int *data)`
- `int sendPacket (unsigned char *cmd, unsigned char *data) const`
- `int receivePacket (unsigned char *cmd, unsigned int *data)`

### 8.11.1 Detaljeret beskrivelse

#### Forfatter

Victor Busk ([201409557@post.au.dk](mailto:201409557@post.au.dk))

Defineret på linje 27 i filen Semesterprojekt3/spiapi.h.

### 8.11.2 Dokumentation af konstruktører og destruktører

#### 8.11.2.1 SPlapi ( )

#### 8.11.2.2 SPlapi ( )

### 8.11.3 Dokumentation af medlemsfunktioner

#### 8.11.3.1 int getPacket ( unsigned char \* *cmd*, unsigned int \* *value* )

#### 8.11.3.2 int getPacket ( unsigned char \* *cmd*, unsigned int \* *value* )

#### 8.11.3.3 int receivePacket ( unsigned char \* *cmd*, unsigned int \* *data* ) [private]

#### 8.11.3.4 int receivePacket ( unsigned char \* *cmd*, unsigned int \* *data* ) [private]

#### 8.11.3.5 int sendPacket ( unsigned char \* *cmd*, unsigned char \* *data* ) const [private]

#### 8.11.3.6 int sendPacket ( const unsigned char \* *cmd*, unsigned char \* *data* ) const [private]

#### 8.11.3.7 int setPacket ( unsigned char \* *cmd*, unsigned char \* *value* ) const

#### 8.11.3.8 int setPacket ( const unsigned char \* *cmd*, unsigned char \* *value* ) const

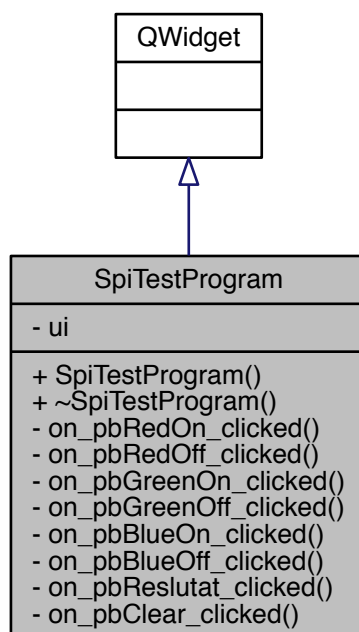
Dokumentationen for denne klasse blev genereret ud fra filen:

- [Semesterprojekt3/spiapi.h](#)

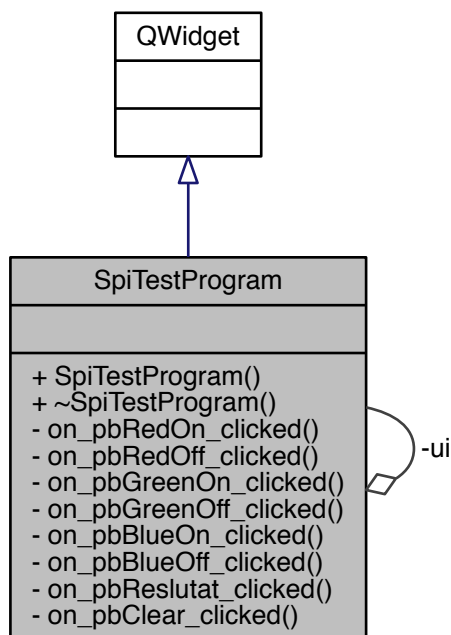
## 8.12 SpiTestProgram Klasse-reference

```
#include <spitestprogram.h>
```

Stamtræ for SpiTestProgram:



Samarbejdsdiagram for SpiTestProgram:



#### Offentlige metoder

- [SpiTestProgram \(QWidget \\*parent=0\)](#)
- [~SpiTestProgram \(\)](#)

#### Private slots

- void [on\\_pbRedOn\\_clicked \(\)](#)
- void [on\\_pbRedOff\\_clicked \(\)](#)
- void [on\\_pbGreenOn\\_clicked \(\)](#)
- void [on\\_pbGreenOff\\_clicked \(\)](#)
- void [on\\_pbBlueOn\\_clicked \(\)](#)
- void [on\\_pbBlueOff\\_clicked \(\)](#)
- void [on\\_pbReslutat\\_clicked \(\)](#)
- void [on\\_pbClear\\_clicked \(\)](#)

#### Private attributter

- [Ui::SpiTestProgram \\* ui](#)

#### 8.12.1 Detaljeret beskrivelse

Defineret på linje 12 i filen spitestprogram.h.



## 8.12.2 Dokumentation af konstruktører og destruktører

8.12.2.1 `SpiTestProgram ( QWidget * parent = 0 )` `[explicit]`

8.12.2.2 `~SpiTestProgram ( )`

## 8.12.3 Dokumentation af medlemsfunktioner

8.12.3.1 `void on_pbBlueOff_clicked ( )` `[private]`, `[slot]`

8.12.3.2 `void on_pbBlueOn_clicked ( )` `[private]`, `[slot]`

8.12.3.3 `void on_pbClear_clicked ( )` `[private]`, `[slot]`

8.12.3.4 `void on_pbGreenOff_clicked ( )` `[private]`, `[slot]`

8.12.3.5 `void on_pbGreenOn_clicked ( )` `[private]`, `[slot]`

8.12.3.6 `void on_pbRedOff_clicked ( )` `[private]`, `[slot]`

8.12.3.7 `void on_pbRedOn_clicked ( )` `[private]`, `[slot]`

8.12.3.8 `void on_pbReslutat_clicked ( )` `[private]`, `[slot]`

## 8.12.4 Felt-dokumentation

8.12.4.1 `Ui::SpiTestProgram* ui` `[private]`

Defineret på linje 31 i filen `spitestprogram.h`.

Dokumentationen for denne klasse blev genereret ud fra filen:

- [spitestprogram.h](#)

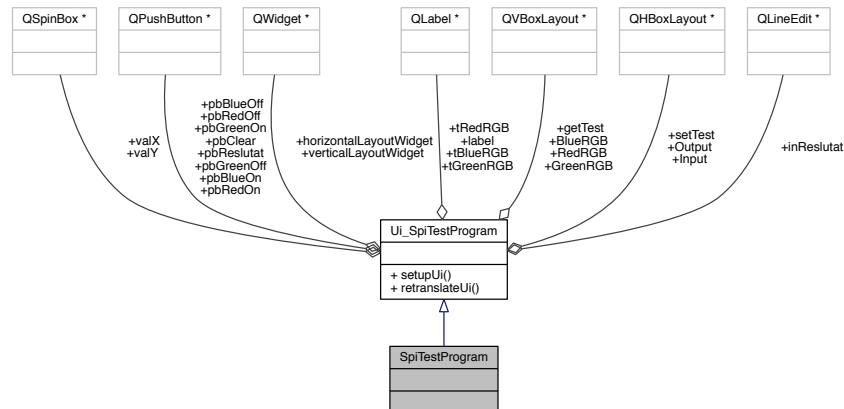
## 8.13 SpiTestProgram Klasse-reference

```
#include <ui_spitestprogram.h>
```

Stamtræ for SpiTestProgram:



Samarbejdsdiagram for SpiTestProgram:



#### Offentlige metoder

- void `setupUi` (`QWidget *SpiTestProgram`)
- void `retranslateUi` (`QWidget *SpiTestProgram`)

#### Datafelter

- `QWidget * horizontalLayoutWidget`
- `QHBoxLayout * setTest`
- `QVBoxLayout * RedRGB`
- `QLabel * tRedRGB`
- `QPushButton * pbRedOn`
- `QPushButton * pbRedOff`
- `QVBoxLayout * GreenRGB`
- `QLabel * tGreenRGB`
- `QPushButton * pbGreenOn`
- `QPushButton * pbGreenOff`
- `QVBoxLayout * BlueRGB`
- `QLabel * tBlueRGB`
- `QPushButton * pbBlueOn`
- `QPushButton * pbBlueOff`
- `QWidget * verticalLayoutWidget`
- `QVBoxLayout * getTest`
- `QHBoxLayout * Input`
- `QSpinBox * valX`
- `QLabel * label`
- `QSpinBox * valY`
- `QPushButton * pbReslutat`
- `QHBoxLayout * Output`
- `QLineEdit * inReslutat`
- `QPushButton * pbClear`

## 8.13.1 Detaljeret beskrivelse

Defineret på linje 234 i filen ui\_spitestprogram.h.

## 8.13.2 Dokumentation af medlemsfunktioner

8.13.2.1 void retranslateUi ( QWidget \* *SpiTestProgram* ) [inline],[inherited]

Defineret på linje 214 i filen ui\_spitestprogram.h.

Refereret til af Ui\_SpiTestProgram::setupUi().

```

215     {
216         tRedRGB->setText(QApplication::translate("SpiTestProgram", "Red RGB", 0,
217         QApplication::UnicodeUTF8));
218         pbRedOn->setText(QApplication::translate("SpiTestProgram", "On", 0,
219         QApplication::UnicodeUTF8));
220         pbRedOff->setText(QApplication::translate("SpiTestProgram", "Off", 0,
221         QApplication::UnicodeUTF8));
222         tGreenRGB->setText(QApplication::translate("SpiTestProgram", "Green RGB", 0,
223         QApplication::UnicodeUTF8));
224         pbGreenOn->setText(QApplication::translate("SpiTestProgram", "On", 0,
225         QApplication::UnicodeUTF8));
226         pbGreenOff->setText(QApplication::translate("SpiTestProgram", "Off", 0,
227         QApplication::UnicodeUTF8));
228         tBlueRGB->setText(QApplication::translate("SpiTestProgram", "Blue RGB", 0,
229         QApplication::UnicodeUTF8));
230         pbBlueOn->setText(QApplication::translate("SpiTestProgram", "On", 0,
231         QApplication::UnicodeUTF8));
232         pbBlueOff->setText(QApplication::translate("SpiTestProgram", "Off", 0,
233         QApplication::UnicodeUTF8));
234         label->setText(QApplication::translate("SpiTestProgram", "x", 0, QApplication::UnicodeUTF8));
235         pbReslutat->setText(QApplication::translate("SpiTestProgram", "Resultat", 0,
236         QApplication::UnicodeUTF8));
237         pbClear->setText(QApplication::translate("SpiTestProgram", "Clear", 0,
238         QApplication::UnicodeUTF8));
239         Q_UNUSED(SpiTestProgram);
240     } // retranslateUi

```

Her er kalder-grafen for denne funktion:

8.13.2.2 void setupUi ( QWidget \* *SpiTestProgram* ) [inline],[inherited]

Defineret på linje 55 i filen ui\_spitestprogram.h.

Indeholder referencer til Ui\_SpiTestProgram::retranslateUi().

```

56     {
57         if (SpiTestProgram->objectName().isEmpty())
58             SpiTestProgram->setObjectName(QString::fromUtf8("SpiTestProgram"));
59         SpiTestProgram->resize(480, 272);
60         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
61         sizePolicy.setHorizontalStretch(0);
62         sizePolicy.setVerticalStretch(0);
63         sizePolicy.setHeightForWidth(SpiTestProgram->sizePolicy().hasHeightForWidth());
64         SpiTestProgram->setSizePolicy(sizePolicy);
65         SpiTestProgram->setMinimumSize(QSize(480, 272));
66         SpiTestProgram->setMaximumSize(QSize(480, 272));
67         horizontalLayoutWidget = new QWidget(SpiTestProgram);
68         horizontalLayoutWidget->setObjectName(QString::fromUtf8("horizontalLayoutWidget"));
69         horizontalLayoutWidget->setGeometry(QRect(9, 9, 461, 131));
70         setTest = new QHBoxLayout(horizontalLayoutWidget);
71         setTest->setSpacing(6);
72         setTest->setContentsMargins(11, 11, 11, 11);
73         setTest->setObjectName(QString::fromUtf8("setTest"));
74         setTest->setContentsMargins(0, 0, 0, 0);
75         RedRGB = new QVBoxLayout();
76         RedRGB->setSpacing(6);
77         RedRGB->setObjectName(QString::fromUtf8("RedRGB"));
78         tRedRGB = new QLabel(horizontalLayoutWidget);
79         tRedRGB->setObjectName(QString::fromUtf8("tRedRGB"));
80         tRedRGB->setMaximumSize(QSize(272, 30));
81         tRedRGB->setAlignment(Qt::AlignCenter);
82
83         RedRGB->addWidget(tRedRGB);
84
85         pbRedOn = new QPushButton(horizontalLayoutWidget);
86         pbRedOn->setObjectName(QString::fromUtf8("pbRedOn"));
87         pbRedOn->setMaximumSize(QSize(160, 30));
88
89         RedRGB->addWidget(pbRedOn);
90
91         pbRedOff = new QPushButton(horizontalLayoutWidget);
92         pbRedOff->setObjectName(QString::fromUtf8("pbRedOff"));
93         pbRedOff->setMaximumSize(QSize(160, 30));
94
95         RedRGB->addWidget(pbRedOff);
96
97
98         setTest->addLayout(RedRGB);
99
100         GreenRGB = new QVBoxLayout();
101         GreenRGB->setSpacing(6);
102         GreenRGB->setObjectName(QString::fromUtf8("GreenRGB"));
103         tGreenRGB = new QLabel(horizontalLayoutWidget);
104         tGreenRGB->setObjectName(QString::fromUtf8("tGreenRGB"));
105         tGreenRGB->setMaximumSize(QSize(160, 30));
106         tGreenRGB->setAlignment(Qt::AlignCenter);
107
108         GreenRGB->addWidget(tGreenRGB);
109
110         pbGreenOn = new QPushButton(horizontalLayoutWidget);
111         pbGreenOn->setObjectName(QString::fromUtf8("pbGreenOn"));
112         pbGreenOn->setMaximumSize(QSize(160, 30));
113
114         GreenRGB->addWidget(pbGreenOn);
115
116         pbGreenOff = new QPushButton(horizontalLayoutWidget);
117         pbGreenOff->setObjectName(QString::fromUtf8("pbGreenOff"));
118         pbGreenOff->setMaximumSize(QSize(160, 30));
119
120         GreenRGB->addWidget(pbGreenOff);
121
122
123         setTest->addLayout(GreenRGB);
124
125         BlueRGB = new QVBoxLayout();
126         BlueRGB->setSpacing(6);
127         BlueRGB->setObjectName(QString::fromUtf8("BlueRGB"));
128         tBlueRGB = new QLabel(horizontalLayoutWidget);
129         tBlueRGB->setObjectName(QString::fromUtf8("tBlueRGB"));
130         tBlueRGB->setMaximumSize(QSize(160, 30));
131         tBlueRGB->setAlignment(Qt::AlignCenter);
132
133         BlueRGB->addWidget(tBlueRGB);
134
135         pbBlueOn = new QPushButton(horizontalLayoutWidget);
136         pbBlueOn->setObjectName(QString::fromUtf8("pbBlueOn"));
137         pbBlueOn->setMaximumSize(QSize(160, 30));
138
139         BlueRGB->addWidget(pbBlueOn);
140
141         pbBlueOff = new QPushButton(horizontalLayoutWidget);

```

```

142     pbBlueOff->setObjectName(QString::fromUtf8("pbBlueOff"));
143     pbBlueOff->setMaximumSize(QSize(160, 30));
144
145     BlueRGB->addWidget(pbBlueOff);
146
147
148     setTest->addLayout(BlueRGB);
149
150     verticalLayoutWidget = new QWidget(SpiTestProgram);
151     verticalLayoutWidget->setObjectName(QString::fromUtf8("verticalLayoutWidget"));
152     verticalLayoutWidget->setGeometry(QRect(10, 150, 461, 111));
153     getTest = new QVBoxLayout(verticalLayoutWidget);
154     getTest->setSpacing(6);
155     getTest->setContentsMargins(11, 11, 11, 11);
156     getTest->setObjectName(QString::fromUtf8("getTest"));
157     getTest->setContentsMargins(0, 0, 0, 0);
158     Input = new QHBoxLayout();
159     Input->setSpacing(6);
160     Input->setObjectName(QString::fromUtf8("Input"));
161     valX = new QSpinBox(verticalLayoutWidget);
162     valX->setObjectName(QString::fromUtf8("valX"));
163     valX->setMaximumSize(QSize(160, 30));
164
165     Input->addWidget(valX);
166
167     label = new QLabel(verticalLayoutWidget);
168     label->setObjectName(QString::fromUtf8("label"));
169     label->setMaximumSize(QSize(20, 30));
170     label->setAlignment(Qt::AlignCenter);
171
172     Input->addWidget(label);
173
174     valY = new QSpinBox(verticalLayoutWidget);
175     valY->setObjectName(QString::fromUtf8("valY"));
176     valY->setMaximumSize(QSize(160, 30));
177
178     Input->addWidget(valY);
179
180     pbReslutat = new QPushButton(verticalLayoutWidget);
181     pbReslutat->setObjectName(QString::fromUtf8("pbReslutat"));
182     pbReslutat->setMaximumSize(QSize(160, 30));
183
184     Input->addWidget(pbReslutat);
185
186
187     getTest->addLayout(Input);
188
189     Output = new QHBoxLayout();
190     Output->setSpacing(6);
191     Output->setObjectName(QString::fromUtf8("Output"));
192     inReslutat = new QLineEdit(verticalLayoutWidget);
193     inReslutat->setObjectName(QString::fromUtf8("inReslutat"));
194     inReslutat->setMaximumSize(QSize(340, 30));
195     inReslutat->setAlignment(Qt::AlignCenter);
196
197     Output->addWidget(inReslutat);
198
199     pbClear = new QPushButton(verticalLayoutWidget);
200     pbClear->setObjectName(QString::fromUtf8("pbClear"));
201     pbClear->setMaximumSize(QSize(160, 30));
202
203     Output->addWidget(pbClear);
204
205
206     getTest->addLayout(Output);
207
208
209     retranslateUi(SpiTestProgram);
210
211     QMetaObject::connectSlotsByName(SpiTestProgram);
212 } // setupUi

```

Her er kald-grafen for denne funktion:



### 8.13.3 Felt-dokumentation

#### 8.13.3.1 `QVBoxLayout* BlueRGB` [inherited]

Defineret på linje 40 i filen `ui_spitestprogram.h`.

#### 8.13.3.2 `QVBoxLayout* getTest` [inherited]

Defineret på linje 45 i filen `ui_spitestprogram.h`.

#### 8.13.3.3 `QVBoxLayout* GreenRGB` [inherited]

Defineret på linje 36 i filen `ui_spitestprogram.h`.

#### 8.13.3.4 `QWidget* horizontalLayoutWidget` [inherited]

Defineret på linje 30 i filen `ui_spitestprogram.h`.

#### 8.13.3.5 `QHBoxLayout* Input` [inherited]

Defineret på linje 46 i filen `ui_spitestprogram.h`.

#### 8.13.3.6 `QLineEdit* inReslutat` [inherited]

Defineret på linje 52 i filen `ui_spitestprogram.h`.

#### 8.13.3.7 `QLabel* label` [inherited]

Defineret på linje 48 i filen `ui_spitestprogram.h`.

#### 8.13.3.8 `QHBoxLayout* Output` [inherited]

Defineret på linje 51 i filen `ui_spitestprogram.h`.

#### 8.13.3.9 `QPushButton* pbBlueOff` [inherited]

Defineret på linje 43 i filen `ui_spitestprogram.h`.

**8.13.3.10 QPushButton\* pbBlueOn** [inherited]

Defineret på linje 42 i filen ui\_spitestprogram.h.

**8.13.3.11 QPushButton\* pbClear** [inherited]

Defineret på linje 53 i filen ui\_spitestprogram.h.

**8.13.3.12 QPushButton\* pbGreenOff** [inherited]

Defineret på linje 39 i filen ui\_spitestprogram.h.

**8.13.3.13 QPushButton\* pbGreenOn** [inherited]

Defineret på linje 38 i filen ui\_spitestprogram.h.

**8.13.3.14 QPushButton\* pbRedOff** [inherited]

Defineret på linje 35 i filen ui\_spitestprogram.h.

**8.13.3.15 QPushButton\* pbRedOn** [inherited]

Defineret på linje 34 i filen ui\_spitestprogram.h.

**8.13.3.16 QPushButton\* pbReslutat** [inherited]

Defineret på linje 50 i filen ui\_spitestprogram.h.

**8.13.3.17 QVBoxLayout\* RedRGB** [inherited]

Defineret på linje 32 i filen ui\_spitestprogram.h.

**8.13.3.18 QHBoxLayout\* setTest** [inherited]

Defineret på linje 31 i filen ui\_spitestprogram.h.

**8.13.3.19 QLabel\* tBlueRGB** [inherited]

Defineret på linje 41 i filen ui\_spitestprogram.h.

**8.13.3.20 QLabel\* tGreenRGB** [inherited]

Defineret på linje 37 i filen ui\_spitestprogram.h.

**8.13.3.21 QLabel\* tRedRGB** [inherited]

Defineret på linje 33 i filen ui\_spitestprogram.h.



#### 8.13.3.22 `QSpinBox* valX` `[inherited]`

Defineret på linje 47 i filen `ui_spitestprogram.h`.

#### 8.13.3.23 `QSpinBox* valY` `[inherited]`

Defineret på linje 49 i filen `ui_spitestprogram.h`.

#### 8.13.3.24 `QWidget* verticalLayoutWidget` `[inherited]`

Defineret på linje 44 i filen `ui_spitestprogram.h`.

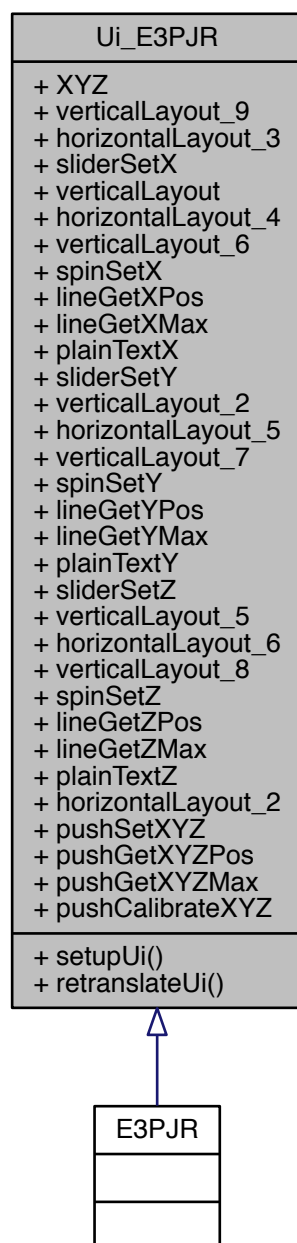
Dokumentationen for denne klasse blev genereret ud fra filen:

- [ui\\_spitestprogram.h](#)

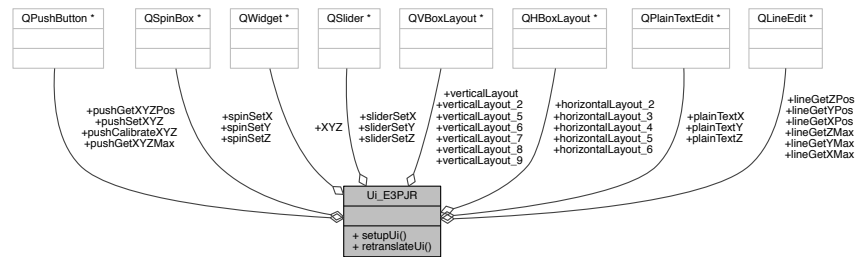
### 8.14 `Ui_E3PJR` Klasse-reference

```
#include <ui_e3pjr.h>
```

Stamtræ for Ui\_E3PJR:



Samarbejdsdiagram for Ui\_E3PJR:



#### Offentlige metoder

- void `setupUi` (QTabWidget \*E3PJR)
- void `retranslateUi` (QTabWidget \*E3PJR)

#### Datafelter

- QWidget \* `XYZ`
- QVBoxLayout \* `verticalLayout_9`
- QHBoxLayout \* `horizontalLayout_3`
- QSlider \* `sliderSetX`
- QVBoxLayout \* `verticalLayout`
- QHBoxLayout \* `horizontalLayout_4`
- QVBoxLayout \* `verticalLayout_6`
- QSpinBox \* `spinSetX`
- QLineEdit \* `lineGetXPos`
- QLineEdit \* `lineGetXMax`
- QPlainTextEdit \* `plainTextX`
- QSlider \* `sliderSetY`
- QVBoxLayout \* `verticalLayout_2`
- QHBoxLayout \* `horizontalLayout_5`
- QVBoxLayout \* `verticalLayout_7`
- QSpinBox \* `spinSetY`
- QLineEdit \* `lineGetYPos`
- QLineEdit \* `lineGetYMax`
- QPlainTextEdit \* `plainTextY`
- QSlider \* `sliderSetZ`
- QVBoxLayout \* `verticalLayout_5`
- QHBoxLayout \* `horizontalLayout_6`
- QVBoxLayout \* `verticalLayout_8`
- QSpinBox \* `spinSetZ`
- QLineEdit \* `lineGetZPos`
- QLineEdit \* `lineGetZMax`
- QPlainTextEdit \* `plainTextZ`
- QHBoxLayout \* `horizontalLayout_2`
- QPushButton \* `pushSetXYZ`
- QPushButton \* `pushGetXYPos`
- QPushButton \* `pushGetXZMax`
- QPushButton \* `pushCalibrateXYZ`

## 8.14.1 Detaljeret beskrivelse

Defineret på linje 29 i filen ui\_e3pjr.h.

## 8.14.2 Dokumentation af medlemsfunktioner

8.14.2.1 void retranslateUi ( QTabWidget \* *E3PJR* ) [inline]

Defineret på linje 281 i filen ui\_e3pjr.h.

Refereret til af setupUi().

```

282     {
283         E3PJR->setWindowTitle(QApplication::translate("E3PJR", "TabWidget", 0, QApplication::UnicodeUTF8));
284         lineGetXPos->setPlaceholderText(QApplication::translate("E3PJR", "xPos", 0,
QApplication::UnicodeUTF8));
285         lineGetXMax->setPlaceholderText(QApplication::translate("E3PJR", "xMax", 0,
QApplication::UnicodeUTF8));
286         lineGetYPos->setPlaceholderText(QApplication::translate("E3PJR", "yPos", 0,
QApplication::UnicodeUTF8));
287         lineGetYMax->setPlaceholderText(QApplication::translate("E3PJR", "yMax", 0,
QApplication::UnicodeUTF8));
288         lineGetZPos->setPlaceholderText(QApplication::translate("E3PJR", "zPos", 0,
QApplication::UnicodeUTF8));
289         lineGetZMax->setPlaceholderText(QApplication::translate("E3PJR", "zMax", 0,
QApplication::UnicodeUTF8));
290         pushSetXYZ->setText(QApplication::translate("E3PJR", "SetXYZPos", 0,
QApplication::UnicodeUTF8));
291         pushGetXYZPos->setText(QApplication::translate("E3PJR", "GetXYZPos", 0,
QApplication::UnicodeUTF8));
292         pushGetXYZMax->setText(QApplication::translate("E3PJR", "GetXYZMax", 0,
QApplication::UnicodeUTF8));
293         pushCalibrateXYZ->setText(QApplication::translate("E3PJR", "CalibrateXYZ", 0,
QApplication::UnicodeUTF8));
294         E3PJR->setTabText(E3PJR->indexOf(XYZ), QApplication::translate("E3PJR", "XYZ", 0,
QApplication::UnicodeUTF8));
295     } // retranslateUi

```

Her er kalder-grafen for denne funktion:

8.14.2.2 void setupUi ( QTabWidget \* *E3PJR* ) [inline]

Defineret på linje 65 i filen ui\_e3pjr.h.

Indeholder referencer til retranslateUi().

```

66     {
67         if (E3PJR->objectName().isEmpty())
68             E3PJR->setObjectName(QString::fromUtf8("E3PJR"));
69         E3PJR->resize(480, 278);
70         XYZ = new QWidget();
71         XYZ->setObjectName(QString::fromUtf8("XYZ"));
72         verticalLayout_9 = new QVBoxLayout(XYZ);
73         verticalLayout_9->setObjectName(QString::fromUtf8("verticalLayout_9"));
74         horizontalLayout_3 = new QHBoxLayout();
75         horizontalLayout_3->setObjectName(QString::fromUtf8("horizontalLayout_3"));
76         sliderSetX = new QSlider(XYZ);
77         sliderSetX->setObjectName(QString::fromUtf8("sliderSetX"));
78         sliderSetX->setMaximum(255);
79         sliderSetX->setSingleStep(5);
80         sliderSetX->setPageStep(15);
81         sliderSetX->setOrientation(Qt::Vertical);
82
83         horizontalLayout_3->addWidget(sliderSetX);
84
85         verticalLayout = new QVBoxLayout();
86         verticalLayout->setObjectName(QString::fromUtf8("verticalLayout"));
87         horizontalLayout_4 = new QHBoxLayout();
88         horizontalLayout_4->setObjectName(QString::fromUtf8("horizontalLayout_4"));
89         verticalLayout_6 = new QVBoxLayout();
90         verticalLayout_6->setObjectName(QString::fromUtf8("verticalLayout_6"));
91         spinSetX = new QSpinBox(XYZ);
92         spinSetX->setObjectName(QString::fromUtf8("spinSetX"));
93         spinSetX->setAlignment(Qt::AlignCenter);
94         spinSetX->setMaximum(255);
95         spinSetX->setSingleStep(5);
96
97         verticalLayout_6->addWidget(spinSetX);
98
99         lineGetXPos = new QLineEdit(XYZ);
100         lineGetXPos->setObjectName(QString::fromUtf8("lineGetXPos"));
101         lineGetXPos->setAlignment(Qt::AlignCenter);
102
103         verticalLayout_6->addWidget(lineGetXPos);
104
105         lineGetXMax = new QLineEdit(XYZ);
106         lineGetXMax->setObjectName(QString::fromUtf8("lineGetXMax"));
107         lineGetXMax->setAlignment(Qt::AlignCenter);
108
109         verticalLayout_6->addWidget(lineGetXMax);
110
111
112         horizontalLayout_4->addLayout(verticalLayout_6);
113
114
115         verticalLayout->addLayout(horizontalLayout_4);
116
117         plainTextX = new QPlainTextEdit(XYZ);
118         plainTextX->setObjectName(QString::fromUtf8("plainTextX"));
119         plainTextX->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
120         plainTextX->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
121         plainTextX->setUndoRedoEnabled(false);
122         plainTextX->setReadOnly(true);
123         plainTextX->setOverwriteMode(false);
124
125         verticalLayout->addWidget(plainTextX);
126
127
128         horizontalLayout_3->addLayout(verticalLayout);
129
130         sliderSetY = new QSlider(XYZ);
131         sliderSetY->setObjectName(QString::fromUtf8("sliderSetY"));
132         sliderSetY->setMaximum(255);
133         sliderSetY->setSingleStep(5);
134         sliderSetY->setPageStep(15);
135         sliderSetY->setOrientation(Qt::Vertical);
136
137         horizontalLayout_3->addWidget(sliderSetY);
138
139         verticalLayout_2 = new QVBoxLayout();
140         verticalLayout_2->setObjectName(QString::fromUtf8("verticalLayout_2"));
141         horizontalLayout_5 = new QHBoxLayout();
142         horizontalLayout_5->setObjectName(QString::fromUtf8("horizontalLayout_5"));
143         verticalLayout_7 = new QVBoxLayout();
144         verticalLayout_7->setObjectName(QString::fromUtf8("verticalLayout_7"));
145         spinSetY = new QSpinBox(XYZ);
146         spinSetY->setObjectName(QString::fromUtf8("spinSetY"));
147         spinSetY->setAlignment(Qt::AlignCenter);
148         spinSetY->setMaximum(255);
149         spinSetY->setSingleStep(5);
150
151         verticalLayout_7->addWidget(spinSetY);
152

```

```
153     lineGetYPos = new QLineEdit(XYZ);
154     lineGetYPos->setObjectName(QString::fromUtf8("lineGetYPos"));
155     lineGetYPos->setAlignment(Qt::AlignCenter);
156
157     verticalLayout_7->addWidget(lineGetYPos);
158
159     lineGetYMax = new QLineEdit(XYZ);
160     lineGetYMax->setObjectName(QString::fromUtf8("lineGetYMax"));
161     lineGetYMax->setAlignment(Qt::AlignCenter);
162
163     verticalLayout_7->addWidget(lineGetYMax);
164
165
166     horizontalLayout_5->addLayout(verticalLayout_7);
167
168
169     verticalLayout_2->addLayout(horizontalLayout_5);
170
171     plainTextY = new QPlainTextEdit(XYZ);
172     plainTextY->setObjectName(QString::fromUtf8("plainTextY"));
173     plainTextY->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
174     plainTextY->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
175     plainTextY->setUndoRedoEnabled(false);
176     plainTextY->setReadOnly(true);
177
178     verticalLayout_2->addWidget(plainTextY);
179
180
181     horizontalLayout_3->addLayout(verticalLayout_2);
182
183     sliderSetZ = new QSlider(XYZ);
184     sliderSetZ->setObjectName(QString::fromUtf8("sliderSetZ"));
185     sliderSetZ->setMaximum(255);
186     sliderSetZ->setSingleStep(5);
187     sliderSetZ->setPageStep(15);
188     sliderSetZ->setOrientation(Qt::Vertical);
189
190     horizontalLayout_3->addWidget(sliderSetZ);
191
192     verticalLayout_5 = new QVBoxLayout();
193     verticalLayout_5->setObjectName(QString::fromUtf8("verticalLayout_5"));
194     horizontalLayout_6 = new QHBoxLayout();
195     horizontalLayout_6->setObjectName(QString::fromUtf8("horizontalLayout_6"));
196     verticalLayout_8 = new QVBoxLayout();
197     verticalLayout_8->setObjectName(QString::fromUtf8("verticalLayout_8"));
198     spinSetZ = new QSpinBox(XYZ);
199     spinSetZ->setObjectName(QString::fromUtf8("spinSetZ"));
200     spinSetZ->setAlignment(Qt::AlignCenter);
201     spinSetZ->setMaximum(255);
202     spinSetZ->setSingleStep(5);
203
204     verticalLayout_8->addWidget(spinSetZ);
205
206     lineGetZPos = new QLineEdit(XYZ);
207     lineGetZPos->setObjectName(QString::fromUtf8("lineGetZPos"));
208     lineGetZPos->setAlignment(Qt::AlignCenter);
209
210     verticalLayout_8->addWidget(lineGetZPos);
211
212     lineGetZMax = new QLineEdit(XYZ);
213     lineGetZMax->setObjectName(QString::fromUtf8("lineGetZMax"));
214     lineGetZMax->setAlignment(Qt::AlignCenter);
215
216     verticalLayout_8->addWidget(lineGetZMax);
217
218
219     horizontalLayout_6->addLayout(verticalLayout_8);
220
221
222     verticalLayout_5->addLayout(horizontalLayout_6);
223
224     plainTextZ = new QPlainTextEdit(XYZ);
225     plainTextZ->setObjectName(QString::fromUtf8("plainTextZ"));
226     plainTextZ->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
227     plainTextZ->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
228     plainTextZ->setUndoRedoEnabled(false);
229     plainTextZ->setReadOnly(true);
230
231     verticalLayout_5->addWidget(plainTextZ);
232
233
234     horizontalLayout_3->addLayout(verticalLayout_5);
235
236
237     verticalLayout_9->addLayout(horizontalLayout_3);
238
239     horizontalLayout_2 = new QHBoxLayout();
```

```

240     horizontalLayout_2->setObjectName(QString::fromUtf8("horizontalLayout_2"));
241     pushSetXYZ = new QPushButton(XYZ);
242     pushSetXYZ->setObjectName(QString::fromUtf8("pushSetXYZ"));
243
244     horizontalLayout_2->addWidget(pushSetXYZ);
245
246     pushGetXYZPos = new QPushButton(XYZ);
247     pushGetXYZPos->setObjectName(QString::fromUtf8("pushGetXYZPos"));
248
249     horizontalLayout_2->addWidget(pushGetXYZPos);
250
251     pushGetXYZMax = new QPushButton(XYZ);
252     pushGetXYZMax->setObjectName(QString::fromUtf8("pushGetXYZMax"));
253
254     horizontalLayout_2->addWidget(pushGetXYZMax);
255
256     pushCalibrateXYZ = new QPushButton(XYZ);
257     pushCalibrateXYZ->setObjectName(QString::fromUtf8("pushCalibrateXYZ"));
258
259     horizontalLayout_2->addWidget(pushCalibrateXYZ);
260
261
262     verticalLayout_9->addLayout(horizontalLayout_2);
263
264     E3PJR->addTab(XYZ, QString());
265     QWidget::setTabOrder(spinSetY, spinSetZ);
266
267     retranslateUi(E3PJR);
268     QObject::connect(sliderSetY, SIGNAL(valueChanged(int)),
spinSetY, SLOT(setValue(int)));
269     QObject::connect(spinSetY, SIGNAL(valueChanged(int)), sliderSetY, SLOT(setValue(
int)));
270     QObject::connect(sliderSetZ, SIGNAL(valueChanged(int)),
spinSetZ, SLOT(setValue(int)));
271     QObject::connect(spinSetZ, SIGNAL(valueChanged(int)), sliderSetZ, SLOT(setValue(
int)));
272     QObject::connect(sliderSetX, SIGNAL(valueChanged(int)),
spinSetX, SLOT(setValue(int)));
273     QObject::connect(spinSetX, SIGNAL(valueChanged(int)), sliderSetX, SLOT(setValue(
int)));
274
275     E3PJR->setCurrentIndex(0);
276
277
278     QMetaObject::connectSlotsByName(E3PJR);
279 } // setupUi

```

Her er kald-grafen for denne funktion:



### 8.14.3 Felt-dokumentation

#### 8.14.3.1 QHBoxLayout\* horizontalLayout\_2

Defineret på linje 59 i filen ui\_e3pjr.h.

#### 8.14.3.2 QHBoxLayout\* horizontalLayout\_3

Defineret på linje 34 i filen ui\_e3pjr.h.

**8.14.3.3 QHBoxLayout\* horizontalLayout\_4**

Defineret på linje 37 i filen ui\_e3pjr.h.

**8.14.3.4 QHBoxLayout\* horizontalLayout\_5**

Defineret på linje 45 i filen ui\_e3pjr.h.

**8.14.3.5 QHBoxLayout\* horizontalLayout\_6**

Defineret på linje 53 i filen ui\_e3pjr.h.

**8.14.3.6 QLineEdit\* lineGetXMax**

Defineret på linje 41 i filen ui\_e3pjr.h.

**8.14.3.7 QLineEdit\* lineGetXPos**

Defineret på linje 40 i filen ui\_e3pjr.h.

**8.14.3.8 QLineEdit\* lineGetYMax**

Defineret på linje 49 i filen ui\_e3pjr.h.

**8.14.3.9 QLineEdit\* lineGetYPos**

Defineret på linje 48 i filen ui\_e3pjr.h.

**8.14.3.10 QLineEdit\* lineGetZMax**

Defineret på linje 57 i filen ui\_e3pjr.h.

**8.14.3.11 QLineEdit\* lineGetZPos**

Defineret på linje 56 i filen ui\_e3pjr.h.

**8.14.3.12 QPlainTextEdit\* plainTextX**

Defineret på linje 42 i filen ui\_e3pjr.h.

**8.14.3.13 QPlainTextEdit\* plainTextY**

Defineret på linje 50 i filen ui\_e3pjr.h.

**8.14.3.14 QPlainTextEdit\* plainTextZ**

Defineret på linje 58 i filen ui\_e3pjr.h.



**8.14.3.15 QPushButton\* pushCalibrateXYZ**

Defineret på linje 63 i filen ui\_e3pjr.h.

**8.14.3.16 QPushButton\* pushGetXYZMax**

Defineret på linje 62 i filen ui\_e3pjr.h.

**8.14.3.17 QPushButton\* pushGetXYZPos**

Defineret på linje 61 i filen ui\_e3pjr.h.

**8.14.3.18 QPushButton\* pushSetXYZ**

Defineret på linje 60 i filen ui\_e3pjr.h.

**8.14.3.19 QSlider\* sliderSetX**

Defineret på linje 35 i filen ui\_e3pjr.h.

**8.14.3.20 QSlider\* sliderSetY**

Defineret på linje 43 i filen ui\_e3pjr.h.

**8.14.3.21 QSlider\* sliderSetZ**

Defineret på linje 51 i filen ui\_e3pjr.h.

**8.14.3.22 QSpinBox\* spinSetX**

Defineret på linje 39 i filen ui\_e3pjr.h.

**8.14.3.23 QSpinBox\* spinSetY**

Defineret på linje 47 i filen ui\_e3pjr.h.

**8.14.3.24 QSpinBox\* spinSetZ**

Defineret på linje 55 i filen ui\_e3pjr.h.

**8.14.3.25 QVBoxLayout\* verticalLayout**

Defineret på linje 36 i filen ui\_e3pjr.h.

**8.14.3.26 QVBoxLayout\* verticalLayout\_2**

Defineret på linje 44 i filen ui\_e3pjr.h.

**8.14.3.27 QVBoxLayout\* verticalLayout\_5**

Defineret på linje 52 i filen ui\_e3pjr.h.

**8.14.3.28 QVBoxLayout\* verticalLayout\_6**

Defineret på linje 38 i filen ui\_e3pjr.h.

**8.14.3.29 QVBoxLayout\* verticalLayout\_7**

Defineret på linje 46 i filen ui\_e3pjr.h.

**8.14.3.30 QVBoxLayout\* verticalLayout\_8**

Defineret på linje 54 i filen ui\_e3pjr.h.

**8.14.3.31 QVBoxLayout\* verticalLayout\_9**

Defineret på linje 33 i filen ui\_e3pjr.h.

**8.14.3.32 QWidget\* XYZ**

Defineret på linje 32 i filen ui\_e3pjr.h.

Dokumentationen for denne klasse blev genereret ud fra filen:

- [ui\\_e3pjr.h](#)

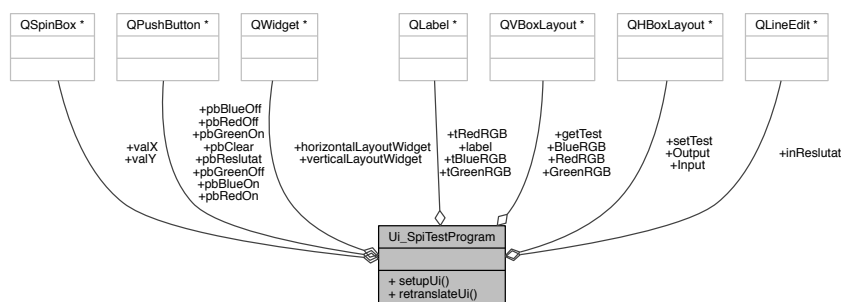
**8.15 Ui\_SpiTestProgram Klasse-reference**

```
#include <ui_spitestprogram.h>
```

Stamtræ for Ui\_SpiTestProgram:



Samarbejdsdiagram for Ui\_SpiTestProgram:



## Offentlige metoder

- void `setUpUi` (`QWidget *SpiTestProgram`)
- void `retranslateUi` (`QWidget *SpiTestProgram`)

## Datafelter

- `QWidget * horizontalLayoutWidget`
- `QHBoxLayout * setTest`
- `QVBoxLayout * RedRGB`
- `QLabel * tRedRGB`
- `QPushButton * pbRedOn`
- `QPushButton * pbRedOff`
- `QVBoxLayout * GreenRGB`
- `QLabel * tGreenRGB`
- `QPushButton * pbGreenOn`
- `QPushButton * pbGreenOff`
- `QVBoxLayout * BlueRGB`
- `QLabel * tBlueRGB`
- `QPushButton * pbBlueOn`
- `QPushButton * pbBlueOff`
- `QWidget * verticalLayoutWidget`
- `QVBoxLayout * getTest`
- `QHBoxLayout * Input`
- `QSpinBox * valX`
- `QLabel * label`
- `QSpinBox * valY`
- `QPushButton * pbReslutat`
- `QHBoxLayout * Output`
- `QLineEdit * inReslutat`
- `QPushButton * pbClear`

## 8.15.1 Detaljeret beskrivelse

Defineret på linje 27 i filen `ui_spitestprogram.h`.

## 8.15.2 Dokumentation af medlemsfunktioner

8.15.2.1 `void retranslateUi ( QWidget * SpiTestProgram )` `[inline]`

Defineret på linje 214 i filen `ui_spitestprogram.h`.

Refereret til af `setUpUi()`.

```

215     {
216         tRedRGB->setText(QApplication::translate("SpiTestProgram", "Red RGB", 0,
217         QApplication::UnicodeUTF8));
218         pbRedOn->setText(QApplication::translate("SpiTestProgram", "On", 0,
219         QApplication::UnicodeUTF8));
220         pbRedOff->setText(QApplication::translate("SpiTestProgram", "Off", 0,
221         QApplication::UnicodeUTF8));
222         tGreenRGB->setText(QApplication::translate("SpiTestProgram", "Green RGB", 0,
223         QApplication::UnicodeUTF8));
224         pbGreenOn->setText(QApplication::translate("SpiTestProgram", "On", 0,
225         QApplication::UnicodeUTF8));
226         pbGreenOff->setText(QApplication::translate("SpiTestProgram", "Off", 0,
227         QApplication::UnicodeUTF8));
228         tBlueRGB->setText(QApplication::translate("SpiTestProgram", "Blue RGB", 0,
229         QApplication::UnicodeUTF8));
230         pbBlueOn->setText(QApplication::translate("SpiTestProgram", "On", 0,
231         QApplication::UnicodeUTF8));
232         pbBlueOff->setText(QApplication::translate("SpiTestProgram", "Off", 0,
233         QApplication::UnicodeUTF8));
234         label->setText(QApplication::translate("SpiTestProgram", "x", 0, QApplication::UnicodeUTF8));
235         pbReslutat->setText(QApplication::translate("SpiTestProgram", "Resultat", 0,
236         QApplication::UnicodeUTF8));
237         pbClear->setText(QApplication::translate("SpiTestProgram", "Clear", 0,
238         QApplication::UnicodeUTF8));
239         Q_UNUSED(SpiTestProgram);
240     } // retranslateUi

```

Her er kalder-grafen for denne funktion:



#### 8.15.2.2 void setupUi ( QWidget \* SpiTestProgram ) [inline]

Defineret på linje 55 i filen ui\_spitestprogram.h.

Indeholder referencer til retranslateUi().

```

56     {
57         if (SpiTestProgram->objectName().isEmpty())
58             SpiTestProgram->setObjectName(QString::fromUtf8("SpiTestProgram"));
59         SpiTestProgram->resize(480, 272);
60         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
61         sizePolicy.setHorizontalStretch(0);
62         sizePolicy.setVerticalStretch(0);
63         sizePolicy.setHeightForWidth(SpiTestProgram->sizePolicy().hasHeightForWidth());
64         SpiTestProgram->setSizePolicy(sizePolicy);
65         SpiTestProgram->setMinimumSize(QSize(480, 272));
66         SpiTestProgram->setMaximumSize(QSize(480, 272));
67         horizontalLayoutWidget = new QWidget(SpiTestProgram);
68         horizontalLayoutWidget->setObjectName(QString::fromUtf8("horizontalLayoutWidget"));
69         horizontalLayoutWidget->setGeometry(QRect(9, 9, 461, 131));
70         setTest = new QHBoxLayout(horizontalLayoutWidget);
71         setTest->setSpacing(6);
72         setTest->setContentsMargins(11, 11, 11, 11);
73         setTest->setObjectName(QString::fromUtf8("setTest"));
74         setTest->setContentsMargins(0, 0, 0, 0);
75         RedRGB = new QVBoxLayout();
76         RedRGB->setSpacing(6);
77         RedRGB->setObjectName(QString::fromUtf8("RedRGB"));
78         tRedRGB = new QLabel(horizontalLayoutWidget);
79         tRedRGB->setObjectName(QString::fromUtf8("tRedRGB"));
80         tRedRGB->setMaximumSize(QSize(272, 30));
81         tRedRGB->setAlignment(Qt::AlignCenter);
82     }

```

```

83     RedRGB->addWidget (tRedRGB);
84
85     pbRedOn = new QPushButton(horizontalLayoutWidget);
86     pbRedOn->setObjectName (QString::fromUtf8("pbRedOn"));
87     pbRedOn->setMaximumSize (QSize(160, 30));
88
89     RedRGB->addWidget (pbRedOn);
90
91     pbRedOff = new QPushButton(horizontalLayoutWidget);
92     pbRedOff->setObjectName (QString::fromUtf8("pbRedOff"));
93     pbRedOff->setMaximumSize (QSize(160, 30));
94
95     RedRGB->addWidget (pbRedOff);
96
97
98     setTest->addLayout (RedRGB);
99
100    GreenRGB = new QVBoxLayout();
101    GreenRGB->setSpacing(6);
102    GreenRGB->setObjectName (QString::fromUtf8("GreenRGB"));
103    tGreenRGB = new QLabel(horizontalLayoutWidget);
104    tGreenRGB->setObjectName (QString::fromUtf8("tGreenRGB"));
105    tGreenRGB->setMaximumSize (QSize(160, 30));
106    tGreenRGB->setAlignment (Qt::AlignCenter);
107
108    GreenRGB->addWidget (tGreenRGB);
109
110    pbGreenOn = new QPushButton(horizontalLayoutWidget);
111    pbGreenOn->setObjectName (QString::fromUtf8("pbGreenOn"));
112    pbGreenOn->setMaximumSize (QSize(160, 30));
113
114    GreenRGB->addWidget (pbGreenOn);
115
116    pbGreenOff = new QPushButton(horizontalLayoutWidget);
117    pbGreenOff->setObjectName (QString::fromUtf8("pbGreenOff"));
118    pbGreenOff->setMaximumSize (QSize(160, 30));
119
120    GreenRGB->addWidget (pbGreenOff);
121
122
123    setTest->addLayout (GreenRGB);
124
125    BlueRGB = new QVBoxLayout();
126    BlueRGB->setSpacing(6);
127    BlueRGB->setObjectName (QString::fromUtf8("BlueRGB"));
128    tBlueRGB = new QLabel(horizontalLayoutWidget);
129    tBlueRGB->setObjectName (QString::fromUtf8("tBlueRGB"));
130    tBlueRGB->setMaximumSize (QSize(160, 30));
131    tBlueRGB->setAlignment (Qt::AlignCenter);
132
133    BlueRGB->addWidget (tBlueRGB);
134
135    pbBlueOn = new QPushButton(horizontalLayoutWidget);
136    pbBlueOn->setObjectName (QString::fromUtf8("pbBlueOn"));
137    pbBlueOn->setMaximumSize (QSize(160, 30));
138
139    BlueRGB->addWidget (pbBlueOn);
140
141    pbBlueOff = new QPushButton(horizontalLayoutWidget);
142    pbBlueOff->setObjectName (QString::fromUtf8("pbBlueOff"));
143    pbBlueOff->setMaximumSize (QSize(160, 30));
144
145    BlueRGB->addWidget (pbBlueOff);
146
147
148    setTest->addLayout (BlueRGB);
149
150    verticalLayoutWidget = new QWidget (SpiTestProgram);
151    verticalLayoutWidget->setObjectName (QString::fromUtf8("verticalLayoutWidget"));
152    verticalLayoutWidget->setGeometry (QRect(10, 150, 461, 111));
153    getTest = new QVBoxLayout (verticalLayoutWidget);
154    getTest->setSpacing(6);
155    getTest->setContentsMargins(11, 11, 11, 11);
156    getTest->setObjectName (QString::fromUtf8("getTest"));
157    getTest->setContentsMargins(0, 0, 0, 0);
158    Input = new QHBoxLayout();
159    Input->setSpacing(6);
160    Input->setObjectName (QString::fromUtf8("Input"));
161    valX = new QSpinBox(verticalLayoutWidget);
162    valX->setObjectName (QString::fromUtf8("valX"));
163    valX->setMaximumSize (QSize(160, 30));
164
165    Input->addWidget (valX);
166
167    label = new QLabel(verticalLayoutWidget);
168    label->setObjectName (QString::fromUtf8("label"));
169    label->setMaximumSize (QSize(20, 30));

```

```

170     label->setAlignment (Qt::AlignCenter);
171
172     Input->addWidget (label);
173
174     valY = new QSpinBox (verticalLayoutWidget);
175     valY->setObjectName (QString::fromUtf8 ("valY"));
176     valY->setMaximumSize (QSize (160, 30));
177
178     Input->addWidget (valY);
179
180     pbReslutat = new QPushButton (verticalLayoutWidget);
181     pbReslutat->setObjectName (QString::fromUtf8 ("pbReslutat"));
182     pbReslutat->setMaximumSize (QSize (160, 30));
183
184     Input->addWidget (pbReslutat);
185
186
187     getTest->addLayout (Input);
188
189     Output = new QHBoxLayout ();
190     Output->setSpacing (6);
191     Output->setObjectName (QString::fromUtf8 ("Output"));
192     inReslutat = new QLineEdit (verticalLayoutWidget);
193     inReslutat->setObjectName (QString::fromUtf8 ("inReslutat"));
194     inReslutat->setMaximumSize (QSize (340, 30));
195     inReslutat->setAlignment (Qt::AlignCenter);
196
197     Output->addWidget (inReslutat);
198
199     pbClear = new QPushButton (verticalLayoutWidget);
200     pbClear->setObjectName (QString::fromUtf8 ("pbClear"));
201     pbClear->setMaximumSize (QSize (160, 30));
202
203     Output->addWidget (pbClear);
204
205
206     getTest->addLayout (Output);
207
208
209     retranslateUi (SpiTestProgram);
210
211     QMetaObject::connectSlotsByName (SpiTestProgram);
212 } // setupUi

```

Her er kald-grafen for denne funktion:



### 8.15.3 Felt-dokumentation

#### 8.15.3.1 QVBoxLayout\* BlueRGB

Defineret på linje 40 i filen ui\_spitestprogram.h.

#### 8.15.3.2 QVBoxLayout\* getTest

Defineret på linje 45 i filen ui\_spitestprogram.h.

#### 8.15.3.3 QVBoxLayout\* GreenRGB

Defineret på linje 36 i filen ui\_spitestprogram.h.

**8.15.3.4 QWidget\* horizontalLayoutWidget**

Defineret på linje 30 i filen ui\_spitestprogram.h.

**8.15.3.5 QHBoxLayout\* Input**

Defineret på linje 46 i filen ui\_spitestprogram.h.

**8.15.3.6 QLineEdit\* inReslutat**

Defineret på linje 52 i filen ui\_spitestprogram.h.

**8.15.3.7 QLabel\* label**

Defineret på linje 48 i filen ui\_spitestprogram.h.

**8.15.3.8 QHBoxLayout\* Output**

Defineret på linje 51 i filen ui\_spitestprogram.h.

**8.15.3.9 QPushButton\* pbBlueOff**

Defineret på linje 43 i filen ui\_spitestprogram.h.

**8.15.3.10 QPushButton\* pbBlueOn**

Defineret på linje 42 i filen ui\_spitestprogram.h.

**8.15.3.11 QPushButton\* pbClear**

Defineret på linje 53 i filen ui\_spitestprogram.h.

**8.15.3.12 QPushButton\* pbGreenOff**

Defineret på linje 39 i filen ui\_spitestprogram.h.

**8.15.3.13 QPushButton\* pbGreenOn**

Defineret på linje 38 i filen ui\_spitestprogram.h.

**8.15.3.14 QPushButton\* pbRedOff**

Defineret på linje 35 i filen ui\_spitestprogram.h.

**8.15.3.15 QPushButton\* pbRedOn**

Defineret på linje 34 i filen ui\_spitestprogram.h.



**8.15.3.16 QPushButton\* pbReslutat**

Defineret på linje 50 i filen ui\_spitestprogram.h.

**8.15.3.17 QVBoxLayout\* RedRGB**

Defineret på linje 32 i filen ui\_spitestprogram.h.

**8.15.3.18 QHBoxLayout\* setTest**

Defineret på linje 31 i filen ui\_spitestprogram.h.

**8.15.3.19 QLabel\* tBlueRGB**

Defineret på linje 41 i filen ui\_spitestprogram.h.

**8.15.3.20 QLabel\* tGreenRGB**

Defineret på linje 37 i filen ui\_spitestprogram.h.

**8.15.3.21 QLabel\* tRedRGB**

Defineret på linje 33 i filen ui\_spitestprogram.h.

**8.15.3.22 QSpinBox\* valX**

Defineret på linje 47 i filen ui\_spitestprogram.h.

**8.15.3.23 QSpinBox\* valY**

Defineret på linje 49 i filen ui\_spitestprogram.h.

**8.15.3.24 QWidget\* verticalLayoutWidget**

Defineret på linje 44 i filen ui\_spitestprogram.h.

Dokumentationen for denne klasse blev genereret ud fra filen:

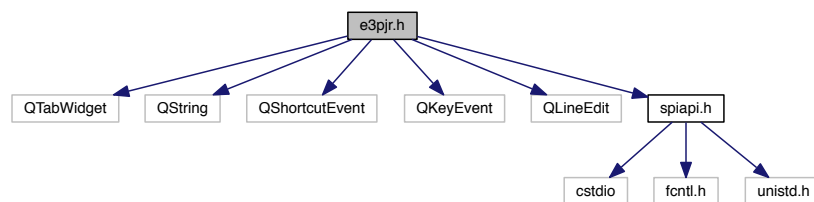
- [ui\\_spitestprogram.h](#)

## 9 Fil-dokumentation

### 9.1 e3pjr.h filreference

```
#include <QTabWidget>
#include <QString>
#include <QShortcutEvent>
#include <QKeyEvent>
#include <QLineEdit>
#include "spiapi.h"
```

Inklusions-afhængighedsgraf for e3pjr.h:



#### Datastrukturer

- class [E3PJR](#)

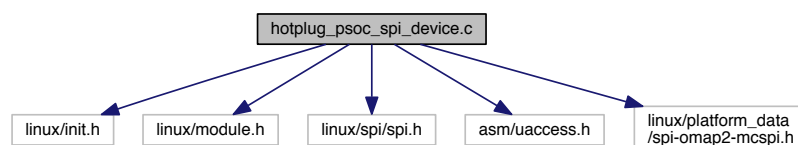
#### Namespaces

- [Ui](#)

### 9.2 hotplug\_psoc\_spi\_device.c filreference

```
#include <linux/init.h>
#include <linux/module.h>
#include <linux/spi/spi.h>
#include <asm/uaccess.h>
#include <linux/platform_data/spi-omap2-mcspi.h>
```

Inklusions-afhængighedsgraf for hotplug\_psoc\_spi\_device.c:



## Funktioner

- `MODULE_AUTHOR` ("Jeppe Stærk")
- `MODULE_LICENSE` ("Dual BSD/GPL")
- `static int hotplug_spi_init (void)`
- `static void hotplug_spi_exit (void)`
- `module_init (hotplug_spi_init)`
- `module_exit (hotplug_spi_exit)`

## Variable

- `static struct spi_device * slave_spi_device`
- `static struct omap2_mcspi_device_config mcspi_config`
- `static struct spi_board_info slave_spi_board_info`

### 9.2.1 Funktions-dokumentation

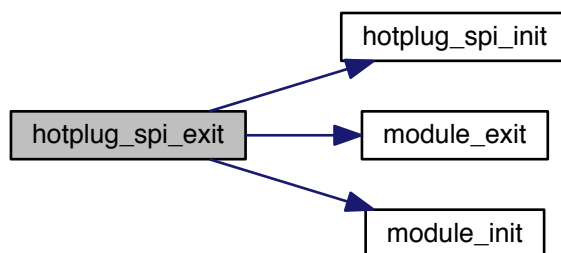
#### 9.2.1.1 `static void hotplug_spi_exit ( void ) [static]`

Defineret på linje 67 i filen `hotplug_psoc_spi_device.c`.

Indeholder referencer til `hotplug_spi_init()`, `module_exit()`, `module_init()`, `slave_spi_board_info` og `slave_spi_device`.

```
68 {  
69     printk(KERN_ALERT "Removing SPI Device: %s, bus: %i, chip-sel: %i\n",  
70         slave_spi_board_info.modalias, slave_spi_board_info.bus_num,  
71         slave_spi_board_info.chip_select);  
72     spi_unregister_device(slave_spi_device);  
72 }
```

Her er kald-grafen for denne funktion:



## 9.2.1.2 static int hotplug\_spi\_init( void ) [static]

Defineret på linje 32 i filen hotplug\_psoc\_spi\_device.c.

Indeholder referencer til slave\_spi\_board\_info og slave\_spi\_device.

Refereret til af hotplug\_spi\_exit().

```

33 {
34     int bus_num;
35     struct spi_master *slaves_spi_master;
36
37     printk(KERN_ALERT "Adding SPI Device: %s, bus: %i, chip-sel: %i\n",
38            slave_spi_board_info.modalias, slave_spi_board_info.bus_num,
39            slave_spi_board_info.chip_select);
40
41     /* Add the slave SPI device to the SPI bus
42      *
43      * These methods are used to hot-plug spi devices.
44      * SPI devices are by nature NOT hot-pluggable, as
45      * they cannot be probed for functionality etc. SPI
46      * devices are normally cold-plugged during boot, that
47      * is, they are added in the board description file:
48      * /arch/arm/mach-omap2/board-devkit8000.c
49      * Using this method we actually doing "hot" cold-plugging
50      * adding devices using a kernel module.
51      * Note that it is crucial that driver and device uses
52      * the same name alias. If not, the device and driver
53      * will not be paired and the probe method in the driver
54      * not be called.
55      */
56     bus_num = slave_spi_board_info.bus_num;
57     slaves_spi_master = spi_busnum_to_master(bus_num);
58     slave_spi_device = spi_new_device(slaves_spi_master,
59                                     &slave_spi_board_info);
60     if(slave_spi_device < 0) {
61         printk(KERN_ALERT "Unsuccesful creating a new device\n");
62         return -1;
63     }
64     return 0;
65 }

```

Her er kalder-grafen for denne funktion:



## 9.2.1.3 MODULE\_AUTHOR ( "Jeppe Stærk" )

## 9.2.1.4 module\_exit( hotplug\_spi\_exit )

Refereret til af hotplug\_spi\_exit().

Her er kalder-grafen for denne funktion:



#### 9.2.1.5 module\_init ( hotplug\_spi\_init )

Refereret til af `hotplug_spi_exit()`.

Her er kalder-grafen for denne funktion:



#### 9.2.1.6 MODULE\_LICENSE ( "Dual BSD/GPL" )

### 9.2.2 Variabel-dokumentation

#### 9.2.2.1 struct omap2\_mcspi\_device\_config mcspi\_config [static]

**Startværdi:**

```
= {
    .turbo_mode    = 1,
}
```

Defineret på linje 16 i filen `hotplug_psoc_spi_device.c`.

#### 9.2.2.2 struct spi\_board\_info slave\_spi\_board\_info [static]

**Startværdi:**

```
= {
    .modalias      = "psoc4",
    .bus_num       = 1,
    .chip_select    = 0,
    .max_speed_hz   = 100000,
    .controller_data = &mcspi_config,
    .mode          = SPI_MODE_3,
}
```

Defineret på linje 23 i filen `hotplug_psoc_spi_device.c`.

Refereret til af `hotplug_spi_exit()` og `hotplug_spi_init()`.

### 9.2.2.3 struct spi\_device\* slave\_spi\_device [static]

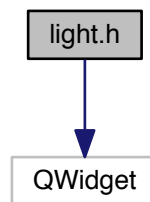
Defineret på linje 11 i filen hotplug\_psoc\_spi\_device.c.

Refereret til af hotplug\_spi\_exit() og hotplug\_spi\_init().

## 9.3 light.h filreference

```
#include <QWidget>
```

Inklusions-afhængighedsgraf for light.h:



### Datastrukturer

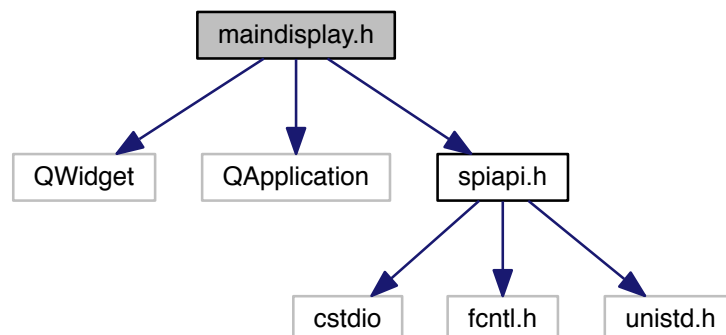
- class [Light](#)

## 9.4 maindisplay.h filreference

Handles all UI-related in maindisplay including all tabs.

```
#include <QWidget>
#include <QApplication>
#include "spiapi.h"
```

Inklusions-afhængighedsgraf for maindisplay.h:



#### Datastrukturer

- class [MainDisplay](#)

#### Namespaces

- [Ui](#)

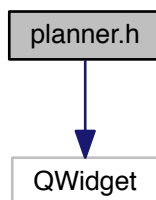
#### 9.4.1 Detaljeret beskrivelse

Handles all UI-related in maindisplay including all tabs.

### 9.5 planner.h filreference

```
#include <QWidget>
```

Inklusions-afhængighedsgraf for planner.h:



#### Datastrukturer

- class [Planner](#)

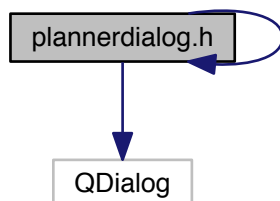
### 9.6 plannerdialog.h filreference

Handles all UI-related in plannerdialog.

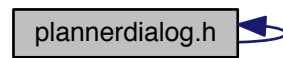
```
#include <QDialog>
```

```
#include "plannerdialog.h"
```

Inklusions-afhængighedsgraf for plannerdialog.h:



Denne graf viser, hvilke filer der direkte eller indirekte inkluderer denne fil:



#### Datastrukturer

- class [PlannerDialog](#)

#### Namespaces

- [Ui](#)

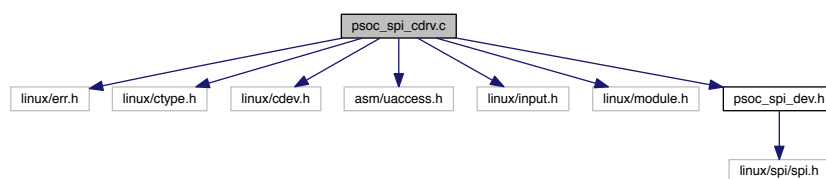
#### 9.6.1 Detaljeret beskrivelse

Handles all UI-related in plannerdialog.

## 9.7 psoc\_spi\_cdrv.c filreference

```
#include <linux/err.h>
#include <linux/ctype.h>
#include <linux/cdev.h>
#include <asm/uaccess.h>
#include <linux/input.h>
#include <linux/module.h>
#include "psoc_spi_dev.h"
```

Inklusions-afhængighedsgraf for psoc\_spi\_cdrv.c:



#### #Defines

- #define [psoc4\\_MAJOR](#) 64
- #define [psoc4\\_MINOR](#) 0
- #define [psoc4\\_DEVICE](#) 1
- #define [MAXLEN](#) 16
- #define [MODULE\\_DEBUG](#) 1
- #define [ERRGOTO](#)(label, ...)



## Funktioner

- `MODULE_AUTHOR` ("Jeppe Stærk")
- `MODULE_LICENSE` ("Dual BSD/GPL")
- static int `__init psoc4_cdrv_init` (void)
- static void `__exit psoc4_cdrv_exit` (void)
- int `psoc4_cdrv_open` (struct inode \*inode, struct file \*filep)
- int `psoc4_cdrv_release` (struct inode \*inode, struct file \*filep)
- ssize\_t `psoc4_cdrv_write` (struct file \*filep, const char \_\_user \*ubuf, size\_t count, loff\_t \*f\_pos)
- ssize\_t `psoc4_cdrv_read` (struct file \*filep, char \_\_user \*ubuf, size\_t count, loff\_t \*f\_pos)
- `module_init` (psoc4\_cdrv\_init)
- `module_exit` (psoc4\_cdrv\_exit)

## Variable

- static struct cdev `psoc4Dev`
- struct file\_operations `psoc4_Fops`
- static int `devno`
- static struct spi\_device \* `psoc4_spi_device` = NULL

### 9.7.1 #Define-dokumentation

#### 9.7.1.1 #define ERRGOTO( label, ... )

#### Værdi:

```
{
    printk (__VA_ARGS__);
    goto label;
} while(0)
```

Defineret på linje 28 i filen `psoc_spi_cdrv.c`.

Refereret til af `psoc4_cdrv_init()`.

#### 9.7.1.2 #define MAXLEN 16

Defineret på linje 15 i filen `psoc_spi_cdrv.c`.

Refereret til af `psoc4_cdrv_write()`.

#### 9.7.1.3 #define MODULE\_DEBUG 1

Defineret på linje 17 i filen `psoc_spi_cdrv.c`.

Refereret til af `psoc4_cdrv_read()` og `psoc4_cdrv_write()`.

#### 9.7.1.4 #define psoc4\_DEVICE 1

Defineret på linje 14 i filen `psoc_spi_cdrv.c`.

Refereret til af `psoc4_cdrv_exit()`, `psoc4_cdrv_init()`, `psoc4_cdrv_open()` og `psoc4_cdrv_release()`.

#### 9.7.1.5 #define psoc4\_MAJOR 64

Defineret på linje 12 i filen psoc\_spi\_cdrv.c.

Refereret til af psoc4\_cdrv\_init().

#### 9.7.1.6 #define psoc4\_MINOR 0

Defineret på linje 13 i filen psoc\_spi\_cdrv.c.

Refereret til af psoc4\_cdrv\_init().

### 9.7.2 Funktions-dokumentation

#### 9.7.2.1 MODULE\_AUTHOR ( "Jeppe Stærk" )

#### 9.7.2.2 module\_exit ( psoc4\_cdrv\_exit )

Refereret til af psoc4\_cdrv\_read().

Her er kalder-grafen for denne funktion:



#### 9.7.2.3 module\_init ( psoc4\_cdrv\_init )

Refereret til af psoc4\_cdrv\_read().

Her er kalder-grafen for denne funktion:



#### 9.7.2.4 MODULE\_LICENSE ( "Dual BSD/GPL" )

#### 9.7.2.5 static void \_\_exit psoc4\_cdrv\_exit ( void ) [static]

Defineret på linje 74 i filen psoc\_spi\_cdrv.c.

Indeholder referencer til devno, psoc4\_DEVICE, psoc4\_spi\_exit() og psoc4Dev.

Refereret til af psoc4\_cdrv\_read().

```

75 {
76     printk("psoc4 driver Exit\n");
77     cdev_del(&psoc4Dev);
78
79     unregister_chrdev_region(devno, psoc4_DEVICE);
80
81     psoc4_spi_exit();
82 }
```

Her er kald-grafen for denne funktion:



Her er kalder-grafen for denne funktion:



#### 9.7.2.6 static int \_\_init psoc4\_cdrv\_init ( void ) [static]

Defineret på linje 38 i filen psoc\_spi\_cdrv.c.

Indeholder referencer til devno, ERRGOTO, psoc4\_DEVICE, psoc4\_Fops, psoc4\_MAJOR, psoc4\_MINOR, psoc4\_spi\_exit(), psoc4\_spi\_init() og psoc4Dev.

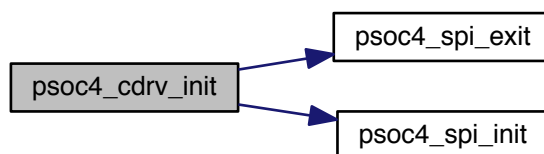
Refereret til af psoc4\_cdrv\_read().

```

39 {
40     int err;
41
42     printk("psoc4 driver initializing\n");
43
44     /* Register SPI Driver */
45     err=psoc4_spi_init();
46     if(err)
47         ERRGOTO(error, "Failed SPI Initialization\n");
48
49     /* Allocate chrdev region */
50     devno = MKDEV(psoc4_MAJOR, psoc4_MINOR);
51     err = register_chrdev_region(devno, psoc4_DEVICE, "psoc4");
52     if(err)
53         ERRGOTO(err_spi_init, "Failed registering char region (%d,%d) +%d, error %d\n",
54                 psoc4_MAJOR, psoc4_MINOR, psoc4_DEVICE, err);
55
56     /* Register Char Device */
57     cdev_init(&psoc4Dev, &psoc4_Fops);
58     err = cdev_add(&psoc4Dev, devno, psoc4_DEVICE);
59     if (err)
60         ERRGOTO(err_register, "Error %d adding psoc4 device\n", err);
61
62     return 0;
63
64 err_register:
65     unregister_chrdev_region(devno, psoc4_DEVICE);
66
67 err_spi_init:
68     psoc4_spi_exit();
69
70 error:
71     return err;
72 }

```

Her er kald-grafen for denne funktion:



Her er kalder-grafen for denne funktion:



#### 9.7.2.7 int psoc4\_cdrv\_open ( struct inode \* inode, struct file \* filep )

Defineret på linje 84 i filen psoc\_spi\_cdrv.c.

Indeholder referencer til psoc4\_DEVICE, psoc4\_get\_device() og psoc4\_spi\_device.

Refereret til af psoc4\_cdrv\_read().

```

85 {
86     int major = imajor(inode);
87     int minor = iminor(inode);
88
89     printk("cdrv_open: Opening psoc4 Device [major], [minor]: %i, %i\n", major, minor);
90
91     /* Check if minor number is within range */
92     if (minor > psoc4_DEVICE-1)
93     {
94         printk("Minor no out of range (0-%i): %i\n", psoc4_DEVICE, minor);
95         return -ENODEV;
96     }
97
98     /* Check if a psoc4 device is registered */
99     if (!(psoc4_spi_device=psoc4_get_device()))
100         return -ENODEV;
101
102     return 0;
103 }

```

Her er kald-grafen for denne funktion:



Her er kalder-grafen for denne funktion:



#### 9.7.2.8 ssize\_t psoc4\_cdrv\_read ( struct file \* filep, char \_\_user \* ubuf, size\_t count, loff\_t \* f\_pos )

Defineret på linje 158 i filen psoc\_spi\_cdrv.c.

Indeholder referencer til MODULE\_DEBUG, module\_exit(), module\_init(), psoc4\_cdrv\_exit(), psoc4\_cdrv\_init(), psoc4\_cdrv\_open(), psoc4\_cdrv\_release(), psoc4\_cdrv\_write(), psoc4\_Fops, psoc4\_spi\_device og psoc4\_spi\_read().

```

159 {
160     int minor, rxLen;
161     u16 rxData;
162     char rxBuffer[5];
163
164     minor = iminor(filep->f_inode);
165
166     if(MODULE_DEBUG)
167         printk(KERN_ALERT "cdrv_read: Reading from psoc4 [Minor] %i \n", minor);
168
169     psoc4_spi_read(psoc4_spi_device, &rxData);
170

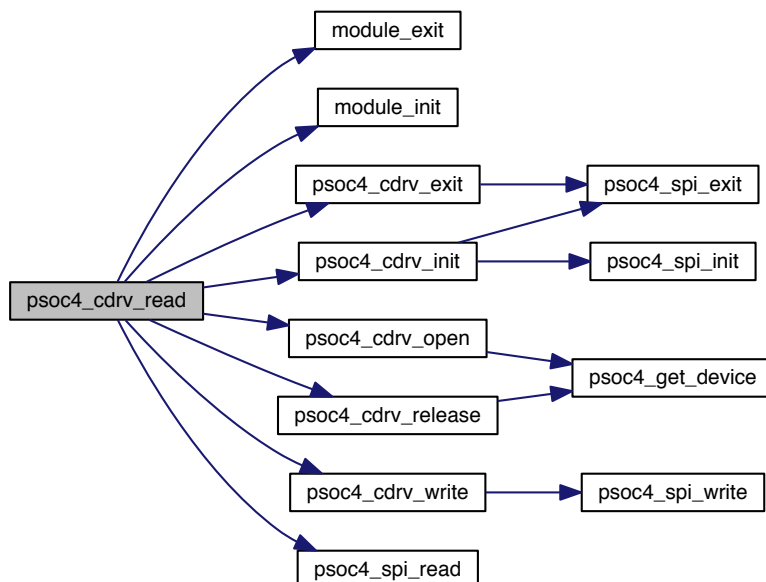
```

```

171     if(MODULE_DEBUG)
172         printk(KERN_DEBUG "cdrv_read: Reading from psoc4 result: 0x%02x\n", rxData);
173
174     // Laver en int om til en string
175     /* Convert to string and copy to user space */
176     // len = snprintf(resultBuf, sizeof resultBuf, "%d\n", result);
177     /* Convert integer to string limited to "count" size. Returns
178     * length excluding NULL termination */
179     rxLen = sprintf(rxBuffer, "%hu", rxData);
180     rxLen++;
181
182     if(MODULE_DEBUG)
183         printk(KERN_DEBUG "cdrv_read: Convert from psoc4 result: %s \n", rxBuffer);
184
185     /* Copy data to user space */
186     if(copy_to_user(ubuf, rxBuffer, rxLen))
187         return -EFAULT;
188
189     /* Move fileptr */
190     *f_pos += rxLen;
191
192     return rxLen;
193 }

```

Her er kald-grafen for denne funktion:



#### 9.7.2.9 int psoc4\_cdrv\_release ( struct inode \* inode, struct file \* filep )

Defineret på linje 105 i filen psoc\_spi\_cdrv.c.

Indeholder referencer til psoc4\_DEVICE, psoc4\_get\_device() og psoc4\_spi\_device.

Refereret til af psoc4\_cdrv\_read().

```

106 {
107     int major = imajor(inode);
108     int minor = iminor(inode);
109
110     printk("cdrv_release: Closing psoc4 Device [major], [minor]: %i, %i\n", major, minor);

```

```

111
112     if ((minor > psoc4_DEVICE-1) || !(psoc4_spi_device=
psoc4_get_device()))
113         return -ENODEV;
114
115     return 0;
116 }

```

Her er kald-grafen for denne funktion:



Her er kalder-grafen for denne funktion:



#### 9.7.2.10 ssize\_t psoc4\_cdrv\_write ( struct file \* filep, const char \_\_user \* ubuf, size\_t count, loff\_t \* f\_pos )

Defineret på linje 119 i filen psoc\_spi\_cdrv.c.

Indeholder referencer til MAXLEN, MODULE\_DEBUG, psoc4\_spi\_device og psoc4\_spi\_write().

Refereret til af psoc4\_cdrv\_read().

```

120 {
121     int err, minor, txLen;
122     char txBuffer[MAXLEN];
123     u16 txData;
124
125     minor = iminor(filep->f_inode);
126
127     printk(KERN_ALERT "cdrv_write: Writing to psoc4 [Minor] %i \n", minor);
128
129     /* Limit copy length to MAXLEN allocated andCopy from user */
130     txLen = count < MAXLEN ? count : MAXLEN;
131     err = copy_from_user(txBuffer, ubuf, txLen);
132     if(err) {return -err;}
133
134     /* Pad null termination to string */
135     txBuffer[txLen] = '\0';
136
137     if(MODULE_DEBUG)
138         printk("cdrv_write: string from user: %s lenth: %i\n", txBuffer, txLen);
139
140     /* Convert sting to int */
141     sscanf(txBuffer, "%hu", &txData);
142
143     if(MODULE_DEBUG)

```

```

144     printk("cdrv_write: data from user: 0x%x\n", txData);
145
146     psoc4_spi_write(psoc4_spi_device, txData);
147
148     /* Legacy file ptr f_pos. Used to support
149      * random access but in char drv we dont!
150      * Move it the length actually written
151      * for compability */
152     *f_pos += txLen;
153
154     /* return length actually written */
155     return txLen;
156 }

```

Her er kald-grafen for denne funktion:



Her er kalder-grafen for denne funktion:



### 9.7.3 Variabel-dokumentation

#### 9.7.3.1 int devno [static]

Defineret på linje 22 i filen psoc\_spi\_cdrv.c.

Refereret til af psoc4\_cdrv\_exit() og psoc4\_cdrv\_init().

#### 9.7.3.2 struct file\_operations psoc4\_Fops

**Startværdi:**

```

=
{
    .owner    = THIS_MODULE,
    .open     = psoc4_cdrv_open,
    .release  = psoc4_cdrv_release,
    .write    = psoc4_cdrv_write,
    .read     = psoc4_cdrv_read,
}

```

Defineret på linje 21 i filen psoc\_spi\_cdrv.c.

Refereret til af psoc4\_cdrv\_init() og psoc4\_cdrv\_read().



### 9.7.3.3 struct spi\_device\* psoc4\_spi\_device = NULL [static]

Defineret på linje 25 i filen psoc\_spi\_cdrv.c.

Refereret til af psoc4\_cdrv\_open(), psoc4\_cdrv\_read(), psoc4\_cdrv\_release() og psoc4\_cdrv\_write().

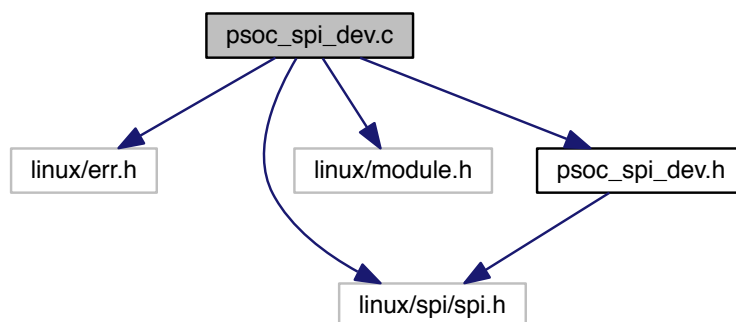
### 9.7.3.4 struct cdev psoc4Dev [static]

Defineret på linje 20 i filen psoc\_spi\_cdrv.c.

Refereret til af psoc4\_cdrv\_exit() og psoc4\_cdrv\_init().

## 9.8 psoc\_spi\_dev.c filreference

```
#include <linux/err.h>
#include <linux/spi/spi.h>
#include <linux/module.h>
#include "psoc_spi_dev.h"
Inklusions-afhængighedsgraf for psoc_spi_dev.c:
```



### #Defines

- #define [MODULE\\_DEBUG](#) 1

### Funktioner

- [MODULE\\_AUTHOR](#) ("Jeppe Stærk")
- [MODULE\\_LICENSE](#) ("Dual BSD/GPL")
- struct spi\_device \* [psoc4\\_get\\_device](#) (void)
- int [psoc4\\_spi\\_read](#) (struct spi\_device \*spi, u16 \*rxData)
- int [psoc4\\_spi\\_write](#) (struct spi\_device \*spi, u16 txData)
- static int [psoc4\\_spi\\_probe](#) (struct spi\_device \*spi)
- static int [psoc4\\_remove](#) (struct spi\_device \*spi)
- int [psoc4\\_spi\\_init](#) (void)
- void [psoc4\\_spi\\_exit](#) (void)

## Variable

- static struct spi\_device \* `psoc4_spi_device` = NULL
- static struct spi\_driver `psoc4_spi_driver`

## 9.8.1 #Define-dokumentation

## 9.8.1.1 #define MODULE\_DEBUG 1

Defineret på linje 9 i filen `psoc_spi_dev.c`.

Refereret til af `psoc4_spi_read()` og `psoc4_spi_write()`.

## 9.8.2 Funktions-dokumentation

## 9.8.2.1 MODULE\_AUTHOR ( "Jeppe Stærk" )

## 9.8.2.2 MODULE\_LICENSE ( "Dual BSD/GPL" )

## 9.8.2.3 struct spi\_device\* psoc4\_get\_device ( void )

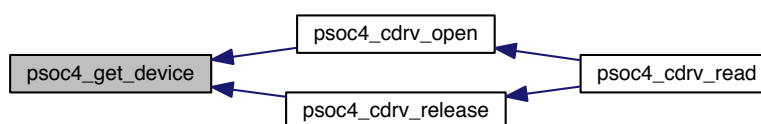
Defineret på linje 16 i filen `psoc_spi_dev.c`.

Indeholder referencer til `psoc4_spi_device`.

Refereret til af `psoc4_cdrv_open()` og `psoc4_cdrv_release()`.

```
16 {
17     return psoc4_spi_device;
18 }
```

Her er kalder-grafen for denne funktion:



## 9.8.2.4 static int psoc4\_remove ( struct spi\_device \* spi ) [static]

Defineret på linje 117 i filen `psoc_spi_dev.c`.

Indeholder referencer til `psoc4_spi_device`.

```
118 {
119     psoc4_spi_device = 0;
120
121     printk (KERN_ALERT "dev_remove: Removing SPI device %s on chip select %i\n",
122            spi->modalias, spi->chip_select);
123
124     return 0;
125 }
```

### 9.8.2.5 void psoc4\_spi\_exit ( void )

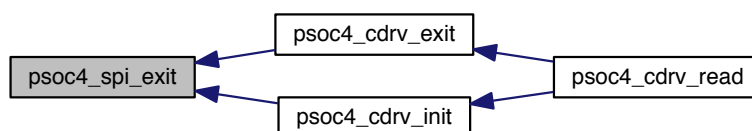
Defineret på linje 165 i filen psoc\_spi\_dev.c.

Indeholder referencer til psoc4\_spi\_driver.

Refereret til af psoc4\_cdrv\_exit() og psoc4\_cdrv\_init().

```
166 {  
167     spi_unregister_driver(&psoc4_spi_driver);  
168 }
```

Her er kalder-grafen for denne funktion:



### 9.8.2.6 int psoc4\_spi\_init ( void )

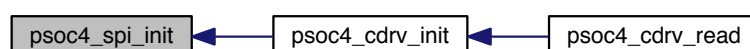
Defineret på linje 148 i filen psoc\_spi\_dev.c.

Indeholder referencer til psoc4\_spi\_driver.

Refereret til af psoc4\_cdrv\_init().

```
149 {  
150     int err;  
151  
152     err = spi_register_driver(&psoc4_spi_driver);  
153  
154     if(err<0)  
155         printk (KERN_ALERT "Error %d registering the psoc4 SPI driver\n", err);  
156  
157     return err;  
158 }
```

Her er kalder-grafen for denne funktion:



## 9.8.2.7 static int psoc4\_spi\_probe ( struct spi\_device \* spi ) [static]

Defineret på linje 97 i filen psoc\_spi\_dev.c.

Indeholder referencer til psoc4\_spi\_device.

```

98 {
99     int err = 0;
100     printk(KERN_DEBUG "dev_probe: New SPI device: %s using chip select: %i\n",
101             spi->modalias, spi->chip_select);
102
103     spi->bits_per_word = 16;
104
105     spi_setup(spi);
106
107     /* In this case we assume just one device */
108     psoc4_spi_device = spi;
109
110     return err;
111 }

```

## 9.8.2.8 int psoc4\_spi\_read ( struct spi\_device \* spi, u16 \* rxData )

Defineret på linje 25 i filen psoc\_spi\_dev.c.

Indeholder referencer til MODULE\_DEBUG.

Refereret til af psoc4\_cdrv\_read().

```

26 {
27     struct spi_transfer t[1];
28     struct spi_message m;
29     u16 rxBuffer = 0;
30
31     /* Check for valid spi device */
32     if(!spi)
33         return -ENODEV;
34
35     /* Init Message */
36     memset(t, 0, sizeof(t));
37     spi_message_init(&m);
38     m.spi = spi;
39
40     t[0].delay_usecs = 60;
41     t[0].tx_buf = NULL;
42     t[0].rx_buf = &rxBuffer;
43     t[0].len = 2;
44     spi_message_add_tail(&t[0], &m);
45
46     /* Transmit SPI Data (blocking) */
47     spi_sync(m.spi, &m);
48
49     if(MODULE_DEBUG)
50         printk(KERN_DEBUG "dev_read: Read data 0x%x\n", rxBuffer);
51
52     *rxData = rxBuffer;
53     return 0;
54 }

```

Her er kalder-grafen for denne funktion:



### 9.8.2.9 int psoc4\_spi\_write ( struct spi\_device \* spi, u16 txData )

Defineret på linje 61 i filen psoc\_spi\_dev.c.

Indeholder referencer til MODULE\_DEBUG.

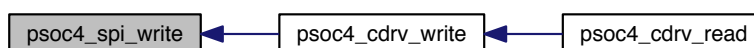
Refereret til af psoc4\_cdrv\_write().

```

62 {
63     struct spi_transfer t[1];
64     struct spi_message m;
65
66     /* Check for valid spi device */
67     if(!spi)
68         return -ENODEV;
69
70     /* Init Message */
71     memset(&t, 0, sizeof(t));
72     spi_message_init(&m);
73     m.spi = spi;
74
75     if(MODULE_DEBUG)
76         printk(KERN_DEBUG "dev_write: Write data 0x%x\n", txData);
77
78     /* Configure tx/rx buffers */
79     t[0].tx_buf = &txData;
80     t[0].rx_buf = NULL;
81     t[0].len = 2;
82     t[0].delay_usecs = 60;
83     spi_message_add_tail(&t[0], &m);
84
85     /* Transmit SPI Data (blocking) */
86     spi_sync(m.spi, &m);
87
88     return 0;
89 }

```

Her er kalder-grafen for denne funktion:



## 9.8.3 Variabel-dokumentation

### 9.8.3.1 struct spi\_device\* psoc4\_spi\_device = NULL [static]

Defineret på linje 13 i filen psoc\_spi\_dev.c.

Refereret til af psoc4\_get\_device(), psoc4\_remove() og psoc4\_spi\_probe().

### 9.8.3.2 struct spi\_driver psoc4\_spi\_driver [static]

**Startværdi:**

```

= {
    .driver = {
        .name = "psoc4",
        .bus = &spi_bus_type,
        .owner = THIS_MODULE,
    },
    .probe = psoc4_spi_probe,
    .remove = psoc4_remove,
}

```

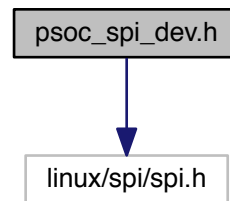
Defineret på linje 133 i filen psoc\_spi\_dev.c.

Refereret til af psoc4\_spi\_exit() og psoc4\_spi\_init().

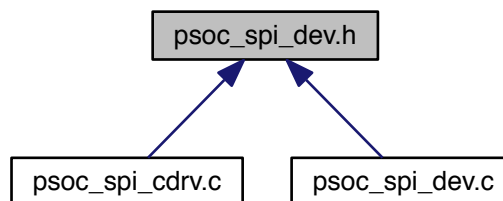
## 9.9 psoc\_spi\_dev.h filreference

```
#include <linux/spi/spi.h>
```

Inklusions-afhængighedsgraf for psoc\_spi\_dev.h:



Denne graf viser, hvilke filer der direkte eller indirekte inkluderer denne fil:



### Funktioner

- struct spi\_device \* [psoc4\\_get\\_device](#) (void)
- int [psoc4\\_spi\\_read](#) (struct spi\_device \*spi, u16 \*rxData)
- int [psoc4\\_spi\\_write](#) (struct spi\_device \*spi, u16 txDate)
- int [psoc4\\_spi\\_init](#) (void)
- void [psoc4\\_spi\\_exit](#) (void)

### 9.9.1 Funktions-dokumentation

#### 9.9.1.1 struct spi\_device\* psoc4\_get\_device ( void )

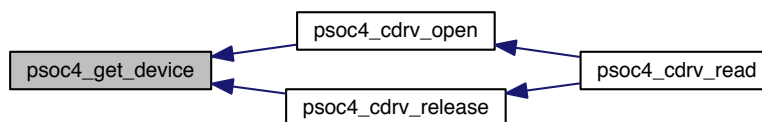
Defineret på linje 16 i filen psoc\_spi\_dev.c.

Indeholder referencer til psoc4\_spi\_device.

Refereret til af psoc4\_cdrv\_open() og psoc4\_cdrv\_release().

```
16 {  
17     return psoc4_spi_device;  
18 }
```

Her er kalder-grafen for denne funktion:



#### 9.9.1.2 void psoc4\_spi\_exit ( void )

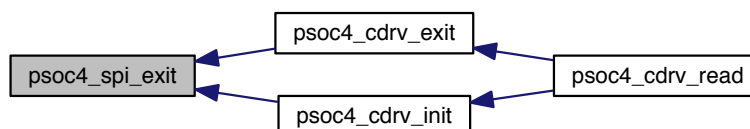
Defineret på linje 165 i filen `psoc_spi_dev.c`.

Indeholder referencer til `psoc4_spi_driver`.

Refereret til af `psoc4_cdrv_exit()` og `psoc4_cdrv_init()`.

```
166 {  
167     spi_unregister_driver(&psoc4_spi_driver);  
168 }
```

Her er kalder-grafen for denne funktion:



#### 9.9.1.3 int psoc4\_spi\_init ( void )

Defineret på linje 148 i filen `psoc_spi_dev.c`.

Indeholder referencer til `psoc4_spi_driver`.

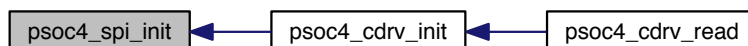
Refereret til af `psoc4_cdrv_init()`.

```

149 {
150     int err;
151
152     err = spi_register_driver(&psoc4_spi_driver);
153
154     if(err<0)
155         printk (KERN_ALERT "Error %d registering the psoc4 SPI driver\n", err);
156
157     return err;
158 }

```

Her er kalder-grafen for denne funktion:



#### 9.9.1.4 int psoc4\_spi\_read ( struct spi\_device \* spi, u16 \* rxData )

Defineret på linje 25 i filen psoc\_spi\_dev.c.

Indeholder referencer til MODULE\_DEBUG.

Refereret til af psoc4\_cdrv\_read().

```

26 {
27     struct spi_transfer t[1];
28     struct spi_message m;
29     u16 rxBuffer = 0;
30
31     /* Check for valid spi device */
32     if(!spi)
33         return -ENODEV;
34
35     /* Init Message */
36     memset(t, 0, sizeof(t));
37     spi_message_init(&m);
38     m.spi = spi;
39
40     t[0].delay_usecs = 60;
41     t[0].tx_buf = NULL;
42     t[0].rx_buf = &rxBuffer;
43     t[0].len = 2;
44     spi_message_add_tail(&t[0], &m);
45
46     /* Transmit SPI Data (blocking) */
47     spi_sync(m.spi, &m);
48
49     if(MODULE_DEBUG)
50         printk(KERN_DEBUG "dev_read: Read data 0x%x\n", rxBuffer);
51
52     *rxData = rxBuffer;
53     return 0;
54 }

```

Her er kalder-grafen for denne funktion:





### 9.9.1.5 int psoc4\_spi\_write ( struct spi\_device \* spi, u16 txData )

Defineret på linje 61 i filen psoc\_spi\_dev.c.

Indeholder referencer til MODULE\_DEBUG.

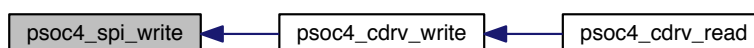
Refereret til af psoc4\_cdrv\_write().

```

62 {
63     struct spi_transfer t[1];
64     struct spi_message m;
65
66     /* Check for valid spi device */
67     if (!spi)
68         return -ENODEV;
69
70     /* Init Message */
71     memset(&t, 0, sizeof(t));
72     spi_message_init(&m);
73     m.spi = spi;
74
75     if(MODULE_DEBUG)
76         printk(KERN_DEBUG "dev_write: Write data 0x%x\n", txData);
77
78     /* Configure tx/rx buffers */
79     t[0].tx_buf = &txData;
80     t[0].rx_buf = NULL;
81     t[0].len = 2;
82     t[0].delay_usecs = 60;
83     spi_message_add_tail(&t[0], &m);
84
85     /* Transmit SPI Data (blocking) */
86     spi_sync(m.spi, &m);
87
88     return 0;
89 }

```

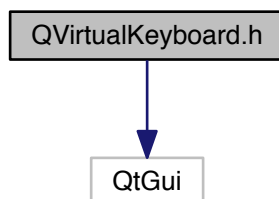
Her er kalder-grafen for denne funktion:



## 9.10 QVirtualKeyboard.h filreference

```
#include <QtGui>
```

Inklusions-afhængighedsgraf for QVirtualKeyboard.h:



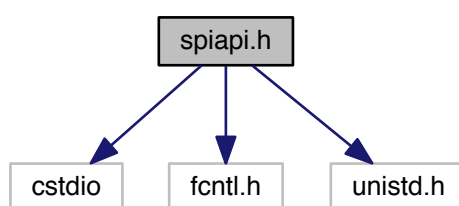
## Datastrukturer

- class [QVirtualKeyboard](#)

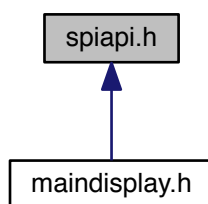
## 9.11 spiapi.h filreference

```
#include <cstdio>
#include <fcntl.h>
#include <unistd.h>
```

Inklusions-afhængighedsgraf for Semesterprojekt3/spiapi.h:



Denne graf viser, hvilke filer der direkte eller indirekte inkluderer denne fil:



## Datastrukturer

- class [SPiapi](#)

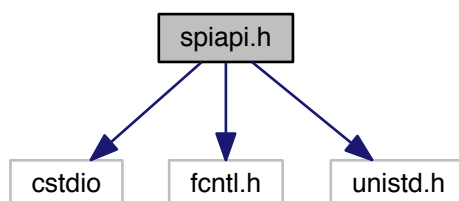
## #Defines

- #define [MAXLEN](#) 5

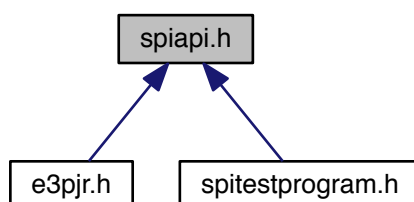
## 9.12 spiapi.h filreference

```
#include <stdio>
#include <fcntl.h>
#include <unistd.h>
```

Inklusions-afhængighedsgraf for SpiTestProgram/spiapi.h:



Denne graf viser, hvilke filer der direkte eller indirekte inkluderer denne fil:



### Datastrukturer

- class [SPlapi](#)

### #Defines

- #define [MAXLEN 5](#)

#### 9.12.1 #Define-dokumentation

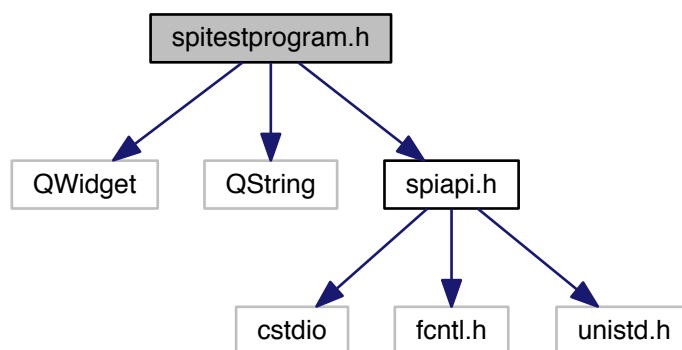
##### 9.12.1.1 #define MAXLEN 5

Defineret på linje 8 i filen SpiTestProgram/spiapi.h.

### 9.13 spitestprogram.h filreference

```
#include <QWidget>
#include <QString>
#include "spiapi.h"
```

Inklusions-afhængighedsgraf for spitestprogram.h:



#### Datastrukturer

- class [SpiTestProgram](#)

#### Namespaces

- [Ui](#)

### 9.14 ui\_e3pjr.h filreference

```
#include <QtCore/QVariant>
#include <QtGui/QAction>
#include <QtGui/QApplication>
#include <QtGui/QButtonGroup>
#include <QtGui/QHBoxLayout>
#include <QtGui/QHeaderView>
#include <QtGui/QLineEdit>
#include <QtGui/QPlainTextEdit>
#include <QtGui/QPushButton>
#include <QtGui/QSlider>
#include <QtGui/QSpinBox>
#include <QtGui/QTabWidget>
#include <QtGui/QVBoxLayout>
#include <QtGui/QWidget>
```

Inklusions-afhængighedsgraf for ui\_e3pjr.h:



## Datastrukturer

- class [Ui\\_E3PJR](#)
- class [E3PJR](#)

## Namespaces

- [Ui](#)

## 9.15 ui\_spitestprogram.h filreference

```
#include <QtCore/QVariant>
#include <QtGui/QAction>
#include <QtGui/QApplication>
#include <QtGui/QButtonGroup>
#include <QtGui/QHBoxLayout>
#include <QtGui/QHeaderView>
#include <QtGui/QLabel>
#include <QtGui/QLineEdit>
#include <QtGui/QPushButton>
#include <QtGui/QSpinBox>
#include <QtGui/QVBoxLayout>
#include <QtGui/QWidget>
```

Inklusions-afhængighedsgraf for ui\_spitestprogram.h:



## Datastrukturer

- class [Ui\\_SpiTestProgram](#)
- class [SpiTestProgram](#)

## Namespaces

- [Ui](#)