Code_generator

Generated by Doxygen 1.8.12

Contents

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

Namespace Documentation

3.1 Package common

Classes

- enum ErrorCode
- class Features

3.1.1 Detailed Description

Common information that needs to be accessed across all the project

Author

Miguel Diaz

Version

0.1

3.2 Package configurator

Classes

- class PinConf

3.2.1 Detailed Description

Configuration classes

Author

Miguel Diaz

Version

0.1

3.3 Package framework

3.3.1 Detailed Description

Framework information

Author

H112943

Version

0.1

3.4 Package microcontroller

Classes

- class Microcontroller
- class Pin

3.4.1 Detailed Description

Microcontroller related classes

Author

Miguel Diaz

Version

0.1

3.5 Package xmlCreator

Classes

- class ConfXmlWriter

3.5.1 Detailed Description

Create configuration XML

Author

Miguel Diaz

Version

0.1

3.6 Package xmlParser

Classes

- class TestMain
- class XmlOpener

3.6.1 Detailed Description

XML parser for microcontroller information and project settings

Author

Miguel Diaz

Version

0.1

Chapter 4

Class Documentation

4.1 xmlCreator.ConfXmlWriter Class Reference

Public Member Functions

- ConfXmlWriter (int pins)
- void addPin (PinConf pin, int pinNum)
- void writeXml (String fileName)

4.1.1 Detailed Description

Write a XML file

Author

Miguel Diaz

Version

0.1

4.1.2 Constructor & Destructor Documentation

4.1.2.1 ConfXmlWriter()

```
xmlCreator.ConfXmlWriter.ConfXmlWriter (
            int pins )
```

Create configuration file

Parameters

pins	Total number of GPIO pins to save

4.1.3 Member Function Documentation

4.1.3.1 addPin()

```
void xmlCreator.ConfXmlWriter.addPin (
            PinConf pin,
            int pinNum )
```

Add a pin configuration to the file

Parameters

pin	Pin configuration
pinNum	Number of GPIO pin

4.1.3.2 writeXml()

```
void xmlCreator.ConfXmlWriter.writeXml (
            String fileName )
```

Write the XMl file

Parameters

fileName	Name of XML configuration file

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

- src/xmlCreator/ConfXmlWriter.java

4.2 common.ErrorCode Enum Reference

Public Attributes

- NO_ERROR

4.2.1 Detailed Description

Error codes enum

Author

Miguel Diaz

Version

0.1

4.2.2 Member Data Documentation

4.2.2.1 NO_ERROR

`common.ErrorCode.NO_ERROR`

No error message

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

- src/common/ErrorCode.java

4.3 common.Features Class Reference

Static Public Member Functions

- static void verbosePrint (String verboseMessage)
- static void debugPrint (String debugMessage)

Static Public Attributes

- static final boolean DEBUG = false
- static final boolean VERBOSE = true
- static final String VERBOSE_STR = "# "
- static final String DEBUG_STR = "#$ "

4.3.1 Detailed Description

Class that includes all project features

Author

Miguel Diaz

Version

0.1

4.3.2 Member Function Documentation

4.3.2.1 debugPrint()

```
static void common.Features.debugPrint (
            String debugMessage )  [static]
```

Print Debug message to console

Parameters

debugMessage	Message to display

4.3.2.2 verbosePrint()

```
static void common.Features.verbosePrint (
            String verboseMessage )  [static]
```

Print Verbose message to console

Parameters

verboseMessage	Message to display

4.3.3 Member Data Documentation

4.3.3.1 DEBUG

```
final boolean common.Features.DEBUG = false  [static]
```

Enables debug functions

4.3.3.2 DEBUG_STR

```
final String common.Features.DEBUG_STR = "#$ "  [static]
```

Debug messages indicator on system console

4.3.3.3 VERBOSE

```
final boolean common.Features.VERBOSE = true  [static]
```

Enables console messages

4.3.3.4 VERBOSE_STR

```
final String common.Features.VERBOSE_STR = "# "  [static]
```

Verbose messages indicator on system console

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

- src/common/Features.java

4.4 microcontroller.Microcontroller Class Reference

Public Member Functions

- Microcontroller (Document ucDoc)
- ErrorCode processDocument ()
- Pin getPin (int pinNum)
- String getUc_model ()
- String getUc_manufacturer ()
- int getUc_pinNum ()
- int getUc_gpioNum ()

4.4.1 Detailed Description

Microcontroller related methods

Author

Miguel Diaz

Version

0.1

4.4.2 Constructor & Destructor Documentation

4.4.2.1 Microcontroller()

```
microcontroller.Microcontroller.Microcontroller (
            Document ucDoc )
```

Constructor

Parameters

ucDoc	Document obtained from XML file

4.4.3 Member Function Documentation

4.4.3.1 getPin()

```
Pin microcontroller.Microcontroller.getPin (
            int pinNum )
```

Get a pin's characteristics

Parameters

pinNum	Number of pin

Returns

Pin's characteristics

4.4.3.2 getUc_gpioNum()

```
int microcontroller.Microcontroller.getUc_gpioNum ( )
```

Get the number of GPIOs in the microcontroller

Returns

Number of GPIOs

4.4.3.3 getUc_manufacturer()

```
String microcontroller.Microcontroller.getUc_manufacturer ( )
```

Get the microcontroller's manufacturer

Returns

Microcontroller's manufacturer

4.4.3.4 getUc_model()

```
String microcontroller.Microcontroller.getUc_model ( )
```

Get the microcontroller's model

Returns

Microcontroller's model

4.4.3.5 getUc_pinNum()

```
int microcontroller.Microcontroller.getUc_pinNum ( )
```

Get the microcontroller's pins number

Returns

Number of pins

4.4.3.6 processDocument()

```
ErrorCode microcontroller.Microcontroller.processDocument ( )
```

Process the document obtained from XML file

Returns

Error status

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

- src/microcontroller/Microcontroller.java

4.5 configurator.GPIO.Mode Enum Reference

Public Attributes

- **MODE_INPUT**
- **MODE_OUTPUT**
- **MODE_ALTERNATE_FUNCTION**
- **MODE_ANALOG**

4.5.1 Detailed Description

GPIO modes

Author

Miguel Diaz

Version

0.1

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

- src/configurator/GPIO/Mode.java

4.6 configurator.GPIO.OutType Enum Reference

Public Attributes

- **OTYPE_PUSH_PULL**
- **OTYPE_OPEN_DRAIN**
- **OTYPE_NOT_AVAILABLE**

4.6.1 Detailed Description

Pin's output type

Author

Miguel Diaz

Version

0.1

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

- src/configurator/GPIO/OutType.java

4.7 microcontroller.Pin Class Reference

Public Member Functions

- Pin ()
- void setFunc_vcc (boolean funcState)
- boolean getFunc_vcc ()
- void setFunc_gnd (boolean funcState)
- boolean getFunc_gnd ()
- void setFunc_gpio (boolean funcState)
- boolean getFunc_gpio ()
- void setFeat_int (boolean featState)
- boolean getFeat_int ()
- void setFeat_adc (boolean featState)
- boolean getFeat_adc ()
- void setFeat_uart (boolean featState)
- boolean getFeat_uart ()
- void setFeat_i2c (boolean featState)
- boolean getFeat_i2c ()
- void setFeat_spi (boolean featState)
- boolean getFeat_spi ()
- void setFeat_clock (boolean featState)
- boolean getFeat_clock ()
- void setFeat_timer (boolean featState)
- boolean getFeat_timer ()
- void setFeat_reset (boolean featState)
- boolean getFeat_reset ()
- void setInt (String feature)
- String getInt ()
- void setAdc (String feature)
- String getAdc ()
- void setUart (String feature)
- String getUart ()
- void setI2c (String feature)
- String getI2c ()
- void setSpi (String feature)
- String getSpi ()
- void setClock (String feature)
- String getClock ()
- void setReset (String feature)
- String getReset ()
- void setTimer (String feature)
- String getTimer ()
- void setName (String pinName)
- String getName ()
- void setNumber (int pinNum)
- int getNumber ()
- void setPort (String pinPort)
- String getPort ()
- boolean isValid ()

4.7.2 Constructor & Destructor Documentation

4.7.2.1 Pin()

```
microcontroller.Pin.Pin ( )
```

Initialize all pin's characteristics and features to their default values

4.7.3 Member Function Documentation

4.7.3.1 getAdc()

```
String microcontroller.Pin.getAdc ( )
```

Get the pin's ADC name

Returns

Pin's ADC

4.7.3.2 getClock()

```
String microcontroller.Pin.getClock ( )
```

Get the pin's clock name

Returns

Pin's clock

4.7.3.3 getFeat_adc()

```
boolean microcontroller.Pin.getFeat_adc ( )
```

See if the pin has an ADC

Returns

Feature availability

4.7.3.4 getFeat_clock()

```
boolean microcontroller.Pin.getFeat_clock ( )
```

See if the pin supports a clock

Returns

Feature availability

4.7.3.5 getFeat_i2c()

```
boolean microcontroller.Pin.getFeat_i2c ( )
```

See if the pin has I2C

Returns

Feature availability

4.7.3.6 getFeat_int()

```
boolean microcontroller.Pin.getFeat_int ( )
```

See if the pin has an interruption

Returns

Feature availability

4.7.3.7 getFeat_reset()

```
boolean microcontroller.Pin.getFeat_reset ( )
```

See if the pin has a reset feature

Returns

Feature availability

4.7.3.8 getFeat_spi()

```
boolean microcontroller.Pin.getFeat_spi ( )
```

See if the pin has SPI

Returns

Feature availability

4.7.3.9 getFeat_timer()

```
boolean microcontroller.Pin.getFeat_timer ( )
```

See if the pin supports a timer

Returns

Feature availability

4.7.3.10 getFeat_uart()

```
boolean microcontroller.Pin.getFeat_uart ( )
```

See if the pin has a UART

Returns

Feature availability

4.7.3.11 getFunc_gnd()

```
boolean microcontroller.Pin.getFunc_gnd ( )
```

See if the pin is GND

Returns

Function availability

4.7.3.12 getFunc_gpio()

```
boolean microcontroller.Pin.getFunc_gpio ( )
```

See if the pin is GPIO

Returns

Function availability

4.7.3.13 getFunc_vcc()

```
boolean microcontroller.Pin.getFunc_vcc ( )
```

See if the pin is Vcc

Returns

Function availability

4.7.3.14 getI2c()

```
String microcontroller.Pin.getI2c ( )
```

Get the pin's I2C name

Returns

Pin's I2C

4.7.3.15 getInt()

```
String microcontroller.Pin.getInt ( )
```

Get the pin's interruption name

Returns

Pin's interruption

4.7.3.16 getName()

```
String microcontroller.Pin.getName ( )
```

Get the pin's name

Returns

Pin's name

4.7.3.17 getNumber()

```
int microcontroller.Pin.getNumber ( )
```

Get the pin's number

Returns

Pin's number

4.7.3.18 getPort()

```
String microcontroller.Pin.getPort ( )
```

Get the pin's port

Returns

Pin's port

4.7.3.19 getReset()

```
String microcontroller.Pin.getReset ( )
```

Get the pin's reset name

Returns

Pin's reset

4.7.3.20 getSpi()

```
String microcontroller.Pin.getSpi ( )
```

Get the pin's SPI name

Returns

Pin's SPI

4.7.3.21 getTimer()

```
String microcontroller.Pin.getTimer ( )
```

Get the pin's timer name

Returns

Pin's timer

4.7.3.22 getUart()

```
String microcontroller.Pin.getUart ( )
```

Get the pin's UART name

Returns

Pin's UART

4.7.3.23 isValid()

```
boolean microcontroller.Pin.isValid ( )
```

Check if the pin is correctly initialized

Returns

True if the pin is correctly initialized

4.7.3.24 setAdc()

```
void microcontroller.Pin.setAdc (
            String feature )
```

Set the pin's ADC

Parameters

feature	Pin's ADC

4.7.3.25 setClock()

```
void microcontroller.Pin.setClock (
            String feature )
```

Set the pin's clock

Parameters

feature	Pin's clock

4.7.3.26 setFeat_adc()

```
void microcontroller.Pin.setFeat_adc (
            boolean featState )
```

Set the pin's ADC feature

Parameters

featState	Feature availability

4.7.3.27 setFeat_clock()

```
void microcontroller.Pin.setFeat_clock (
            boolean featState )
```

Set the pin's Clock feature

Parameters

featState	Feature availability

4.7.3.28 setFeat_i2c()

```
void microcontroller.Pin.setFeat_i2c (
            boolean featState )
```

Set the pin's I2C feature

Parameters

featState	Feature availability

4.7.3.29 setFeat_int()

```
void microcontroller.Pin.setFeat_int (
            boolean featState )
```

Set the pin's interruption feature

Parameters

featState	Feature availability

4.7.3.30 setFeat_reset()

```
void microcontroller.Pin.setFeat_reset (
            boolean featState )
```

Set the pin's reset feature

Parameters

featState	Feature availability

4.7.3.31 setFeat_spi()

```
void microcontroller.Pin.setFeat_spi (
            boolean featState )
```

Set the pin's SPI feature

Parameters

featState	Feature availability

4.7.3.32 setFeat_timer()

```
void microcontroller.Pin.setFeat_timer (
            boolean featState )
```

Set the pin's timer feature

Parameters

featState	Feature availability

4.7.3.33 setFeat_uart()

```
void microcontroller.Pin.setFeat_uart (
            boolean featState )
```

Set the pin's UART feature

Parameters

featState	Feature availability

4.7.3.34 setFunc_gnd()

```
void microcontroller.Pin.setFunc_gnd (
            boolean funcState )
```

Set the pin to GND status

Parameters

funcState	Function availability

4.7.3.35 setFunc_gpio()

```
void microcontroller.Pin.setFunc_gpio (
            boolean funcState )
```

Set the pin to GPIO status

Parameters

funcState	Function availability

4.7.3.36 setFunc_vcc()

```
void microcontroller.Pin.setFunc_vcc (
            boolean funcState )
```

Set the pin to Vcc status

Parameters

funcState	Function availability

4.7.3.37 setI2c()

```
void microcontroller.Pin.setI2c (
            String feature )
```

Set the pin's I2C

Parameters

feature	Pin's I2C

4.7.3.38 setInt()

```
void microcontroller.Pin.setInt (
            String feature )
```

Set the pin's interruption

Parameters

feature	Pin's interruption

4.7.3.39 setName()

```
void microcontroller.Pin.setName (
            String pinName )
```

Set the pin's name

Parameters

pinName	Pin's name

4.7.3.40 setNumber()

```
void microcontroller.Pin.setNumber (
            int pinNum )
```

Set the pin's number

Parameters

pinNum	Pin's number

4.7.3.41 setPort()

```
void microcontroller.Pin.setPort (
            String pinPort )
```

Set the pin's port

Parameters

pinPort	Pin's port

4.7.3.42 setReset()

```
void microcontroller.Pin.setReset (
            String feature )
```

Set the pin's reset

Parameters

feature	Pin's reset

4.7.3.43 setSpi()

```
void microcontroller.Pin.setSpi (
            String feature )
```

Set the pin's SPI

Parameters

feature	Pin's SPI

4.7.3.44 setTimer()

```
void microcontroller.Pin.setTimer (
            String feature )
```

Set the pin's timer

Parameters

feature	Pin's timer

4.7.3.45 setUart()

```
void microcontroller.Pin.setUart (
            String feature )
```

Set the pin's UART

Parameters

feature	Pin's UART

4.7.4 Member Data Documentation

4.7.4.1 DEF_FEATURE

```
final String microcontroller.Pin.DEF_FEATURE = DEF_STRING  [static]
```

Default value for pin's feature as not available

4.7.4.2 DEF_FEATURE_AV

```
final boolean microcontroller.Pin.DEF_FEATURE_AV = DEF_BOOLEAN  [static]
```

Default value for pin's feature availability as not available

4.7.4.3 DEF_FUNCTION

```
final boolean microcontroller.Pin.DEF_FUNCTION = DEF_BOOLEAN  [static]
```

Default value for pin's function as not enabled

4.7.4.4 DEF_NAME

```
final String microcontroller.Pin.DEF_NAME = DEF_STRING  [static]
```

Default value for pin's name

4.7.4.5 DEF_NUMBER

```
final int microcontroller.Pin.DEF_NUMBER = DEF_INT  [static]
```

Default value for pin's number

4.7.4.6 DEF_PORT

```
final String microcontroller.Pin.DEF_PORT = DEF_STRING  [static]
```

Default value for pin's port

4.7.4.7 DISABLE

```
final boolean microcontroller.Pin.DISABLE = false  [static]
```

Disable value for features and functions

4.7.4.8 ENABLE

```
final boolean microcontroller.Pin.ENABLE = true  [static]
```

Enable value for features and functions

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

- src/microcontroller/Pin.java

4.8 configurator.PinConf Class Reference

Public Member Functions

- PinConf (Pin gpioPin)
- boolean isValid ()
- String getPort ()
- String getPin ()
- Mode getMode ()
- void setMode (Mode mode)
- OutType getOutType ()
- void setOutType (OutType outType)
- Speed getSpeed ()
- void setSpeed (Speed speed)
- Pull getPull ()
- void setPull (Pull pull)
- boolean isAv_Adc ()
- boolean isAv_altFunc ()

Static Public Attributes

- static final Mode DF_MODE = Mode.MODE_INPUT
- static final Speed DF_SPEED = Speed.SPEED_FAST
- static final OutType DF_OUTTYPE = OutType.OTYPE_PUSH_PULL
- static final Pull DF_PULL = Pull.PULL_NOT_AVAILABLE

4.8.1 Detailed Description

GPIO pin configuration

Author

Miguel Diaz

Version

0.1

4.8.2 Constructor & Destructor Documentation

4.8.2.1 PinConf()

```
configurator.PinConf.PinConf (
            Pin gpioPin )
```

Constructor

Parameters

gpioPin	Pin information

4.8.3 Member Function Documentation

4.8.3.1 getMode()

```
Mode configurator.PinConf.getMode ( )
```

Get the pin's mode configuration

Returns

Mode

4.8.3.2 getOutType()

```
OutType configurator.PinConf.getOutType ( )
```

Get the pin's output configuration

Returns

Output configuration

4.8.3.3 getPin()

```
String configurator.PinConf.getPin ( )
```

Get the pin's number

Returns

Pin's number

4.8.3.4 getPort()

```
String configurator.PinConf.getPort ( )
```

Get the pin's port

Returns

Port

4.8.3.5 getPull()

```
Pull configurator.PinConf.getPull ( )
```

Get the pin's pull resistor configuration

Returns

Pull Resistor configuration

4.8.3.6 getSpeed()

```
Speed configurator.PinConf.getSpeed ( )
```

Get the pin's speed

Returns

Speed

4.8.3.7 isAv_Adc()

```
boolean configurator.PinConf.isAv_Adc ( )
```

Check availability of ADC

Returns

True if ADC is available

4.8.3.8 isAv_altFunc()

```
boolean configurator.PinConf.isAv_altFunc ( )
```

Check the availability of alternate function

Returns

True if alternate function is available

4.8.3.9 isValid()

```
boolean configurator.PinConf.isValid ( )
```

Check if the GPIO pin is valid

Returns

True if valid

4.8.3.10 setMode()

```
void configurator.PinConf.setMode (
            Mode mode )
```

Set the pin's mode configuration

Parameters

mode	Mode

4.8.3.11 setOutType()

```
void configurator.PinConf.setOutType (
            OutType outType )
```

Set the pin's output configuration

Parameters

outType	Output configuration

4.8.3.12 setPull()

```
void configurator.PinConf.setPull (
            Pull pull )
```

Set the pull resistor configuration

Parameters

pull	Resistor configuration

4.8.3.13 setSpeed()

```
void configurator.PinConf.setSpeed (
            Speed speed )
```

Set the pin's speed

Parameters

speed	Speed

4.8.4 Member Data Documentation

4.8.4.1 DF_MODE

```
final Mode configurator.PinConf.DF_MODE = Mode.MODE_INPUT  [static]
```

Default Pin mode

4.8.4.2 DF_OUTTYPE

```
final OutType configurator.PinConf.DF_OUTTYPE = OutType.OTYPE_PUSH_PULL  [static]
```

Default pin's output type

4.8.4.3 DF_PULL

```
final Pull configurator.PinConf.DF_PULL = Pull.PULL_NOT_AVAILABLE  [static]
```

Default pin'r pull resistor

4.8.4.4 DF_SPEED

```
final Speed configurator.PinConf.DF_SPEED = Speed.SPEED_FAST  [static]
```

Default pin's speed

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

- src/configurator/PinConf.java

4.9 configurator.GPIO.Pull Enum Reference

Public Attributes

- **PULL_UP**
- **PULL_DOWN**
- **PULL_NOT_AVAILABLE**

4.9.1 Detailed Description

Pin's pull resistor

Author

Miguel Diaz

Version

0.1

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

- src/configurator/GPIO/Pull.java

4.10 configurator.GPIO.Speed Enum Reference

Public Attributes

- **SPEED_FAST**
- **SPEED_MEDIUM**
- **SPEED_HIGH**
- **SPEED_NOT_AVAILABLE**

4.10.1 Detailed Description

Pin's speed

Author

Miguel Diaz

Version

0.1

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

- src/configurator/GPIO/Speed.java

4.11 xmlParser.TestMain Class Reference

Static Public Member Functions

- static void main (String[ ] openOption)

4.11.1 Detailed Description

Dummy main class for testing the other classes

Author

Miguel Diaz

4.11.2 Member Function Documentation

4.11.2.1 main()

```
static void xmlParser.TestMain.main (
            String [] openOption )  [static]
```

Main without GUI

Parameters

openOption	Options include:

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

- src/xmlParser/TestMain.java

4.12 xmlParser.XmlOpener Class Reference

Public Member Functions

- XmlOpener ()
- ErrorCode OpenFile (String fileName)
- Document getParsedDoc ()

4.12.1 Detailed Description

Open and process XML files

Author

H112943

Version

0.1

4.12.2 Constructor & Destructor Documentation

4.12.2.1 XmlOpener()

```
xmlParser.XmlOpener.XmlOpener ( )
```

Constructor

4.12.3 Member Function Documentation

4.12.3.1 getParsedDoc()

```
Document xmlParser.XmlOpener.getParsedDoc ( )
```

Get the parsed document AFTER opening the file

Returns

Parsed document

4.12.3.2 OpenFile()

```
ErrorCode xmlParser.XmlOpener.OpenFile (
            String fileName )
```

Open the XML file

Parameters

fileName	Complete path and name of XML file

Returns

Error code

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

- src/xmlParser/XmlOpener.java

Index