

Code\_generator

Generated by Doxygen 1.8.12



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Packages . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List . . . . .	3
<b>3</b>	<b>Namespace Documentation</b>	<b>5</b>
3.1	Package common . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.2	Package microcontroller . . . . .	5
3.2.1	Detailed Description . . . . .	6
3.3	Package xmlParser . . . . .	6
3.3.1	Detailed Description . . . . .	6
<b>4</b>	<b>Class Documentation</b>	<b>7</b>
4.1	common.ErrorCode Enum Reference . . . . .	7
4.1.1	Detailed Description . . . . .	7
4.1.2	Member Data Documentation . . . . .	7
4.1.2.1	NO_ERROR . . . . .	7
4.2	common.Features Class Reference . . . . .	8
4.2.1	Detailed Description . . . . .	8
4.2.2	Member Data Documentation . . . . .	8
4.2.2.1	DEBUG . . . . .	8

4.2.2.2	DEBUG_STR	8
4.2.2.3	VERBOSE	8
4.2.2.4	VERBOSE_STR	9
4.3	microcontroller.Microcontroller Class Reference	9
4.3.1	Detailed Description	9
4.3.2	Constructor & Destructor Documentation	9
4.3.2.1	Microcontroller()	9
4.3.3	Member Function Documentation	10
4.3.3.1	getPin()	10
4.3.3.2	getUc_manufacturer()	10
4.3.3.3	getUc_model()	10
4.3.3.4	getUc_pinNum()	11
4.3.3.5	processDocument()	11
4.4	microcontroller.Pin Class Reference	11
4.4.1	Detailed Description	12
4.4.2	Constructor & Destructor Documentation	13
4.4.2.1	Pin()	13
4.4.3	Member Function Documentation	13
4.4.3.1	getAdc()	13
4.4.3.2	getClock()	14
4.4.3.3	getFeat_adc()	14
4.4.3.4	getFeat_clock()	14
4.4.3.5	getFeat_i2c()	14
4.4.3.6	getFeat_int()	15
4.4.3.7	getFeat_reset()	15
4.4.3.8	getFeat_spi()	15
4.4.3.9	getFeat_timer()	15
4.4.3.10	getFeat_uart()	16

4.4.3.11	<a href="#">getFunc_gnd()</a>	16
4.4.3.12	<a href="#">getFunc_gpio()</a>	16
4.4.3.13	<a href="#">getFunc_vcc()</a>	16
4.4.3.14	<a href="#">getI2c()</a>	17
4.4.3.15	<a href="#">getInt()</a>	17
4.4.3.16	<a href="#">getName()</a>	17
4.4.3.17	<a href="#">getNumber()</a>	17
4.4.3.18	<a href="#">getPort()</a>	18
4.4.3.19	<a href="#">getReset()</a>	18
4.4.3.20	<a href="#">getSpi()</a>	18
4.4.3.21	<a href="#">getTimer()</a>	18
4.4.3.22	<a href="#">getUart()</a>	19
4.4.3.23	<a href="#">isValid()</a>	19
4.4.3.24	<a href="#">setAdc()</a>	19
4.4.3.25	<a href="#">setClock()</a>	19
4.4.3.26	<a href="#">setFeat_adc()</a>	20
4.4.3.27	<a href="#">setFeat_clock()</a>	20
4.4.3.28	<a href="#">setFeat_i2c()</a>	20
4.4.3.29	<a href="#">setFeat_int()</a>	20
4.4.3.30	<a href="#">setFeat_reset()</a>	21
4.4.3.31	<a href="#">setFeat_spi()</a>	21
4.4.3.32	<a href="#">setFeat_timer()</a>	21
4.4.3.33	<a href="#">setFeat_uart()</a>	21
4.4.3.34	<a href="#">setFunc_gnd()</a>	22
4.4.3.35	<a href="#">setFunc_gpio()</a>	22
4.4.3.36	<a href="#">setFunc_vcc()</a>	22
4.4.3.37	<a href="#">setI2c()</a>	22
4.4.3.38	<a href="#">setInt()</a>	23

4.4.3.39	setName()	23
4.4.3.40	setNumber()	23
4.4.3.41	setPort()	23
4.4.3.42	setReset()	24
4.4.3.43	setSpi()	24
4.4.3.44	setTimer()	24
4.4.3.45	setUart()	24
4.4.4	Member Data Documentation	25
4.4.4.1	DEF_FEATURE	25
4.4.4.2	DEF_FEATURE_AV	25
4.4.4.3	DEF_FUNCTION	25
4.4.4.4	DEF_NAME	25
4.4.4.5	DEF_NUMBER	25
4.4.4.6	DEF_PORT	25
4.4.4.7	DISABLE	25
4.4.4.8	ENABLE	26
4.5	xmlParser.TestMain Class Reference	26
4.5.1	Detailed Description	26
4.5.2	Member Function Documentation	26
4.5.2.1	main()	26
4.6	xmlParser.XmlOpener Class Reference	27
4.6.1	Detailed Description	27
4.6.2	Constructor & Destructor Documentation	27
4.6.2.1	XmlOpener()	27
4.6.3	Member Function Documentation	27
4.6.3.1	getParsedDoc()	27
4.6.3.2	OpenFile()	27

# Chapter 1

## Namespace Index

### 1.1 Packages

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

<a href="#">common</a>	5
<a href="#">microcontroller</a>	5
<a href="#">xmlParser</a>	6





## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">common.ErrorCode</a>	7
<a href="#">common.Features</a>	8
<a href="#">microcontroller.Microcontroller</a>	9
<a href="#">microcontroller.Pin</a>	11
<a href="#">xmlParser.TestMain</a>	26
<a href="#">xmlParser.XmlOpener</a>	27



## 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 microcontroller

#### Classes

- class [Microcontroller](#)
- class [Pin](#)

### 3.2.1 Detailed Description

[Microcontroller](#) related classes

Author

Miguel Diaz

Version

0.1

## 3.3 Package xmlParser

### Classes

- class [TestMain](#)
- class [XmlOpener](#)

### 3.3.1 Detailed Description

XML parser for microcontroller information and project settings

Author

Miguel Diaz

Version

0.1

## Chapter 4

# Class Documentation

### 4.1 common.ErrorCode Enum Reference

#### Public Attributes

- [NO\\_ERROR](#)

#### 4.1.1 Detailed Description

Error codes enum

#### Author

Miguel Diaz

#### Version

0.1

#### 4.1.2 Member Data Documentation

##### 4.1.2.1 NO\_ERROR

`common.ErrorCode.NO_ERROR`

No error message

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

- `src/common/ErrorCode.java`

## 4.2 common.Features Class Reference

### Static Public Attributes

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

### 4.2.1 Detailed Description

Class that includes all project features

#### Author

Miguel Diaz

#### Version

0.1

### 4.2.2 Member Data Documentation

#### 4.2.2.1 `DEBUG`

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

Enables debug functions

#### 4.2.2.2 `DEBUG_STR`

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

Debug messages indicator on system console

#### 4.2.2.3 `VERBOSE`

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

Enables console messages

#### 4.2.2.4 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

## 4.3 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](#) ()

### 4.3.1 Detailed Description

[Microcontroller](#) related methods

#### Author

Miguel Diaz

#### Version

0.1

### 4.3.2 Constructor & Destructor Documentation

#### 4.3.2.1 Microcontroller()

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

#### Constructor

**Parameters**

<i>ucDoc</i>	Document obtained from XML file
--------------	---------------------------------

### 4.3.3 Member Function Documentation

#### 4.3.3.1 `getPin()`

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

Get a pin's characteristics

**Parameters**

<i>pinNum</i>	Number of pin
---------------	---------------

**Returns**

[Pin's characteristics](#)

#### 4.3.3.2 `getUc_manufacturer()`

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

Get the microcontroller's manufacturer

**Returns**

[Microcontroller's manufacturer](#)

#### 4.3.3.3 `getUc_model()`

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

Get the microcontroller's model

**Returns**

[Microcontroller's model](#)



#### 4.3.3.4 getUc\_pinNum()

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

Get the microcontroller's pins number

##### Returns

Number of pins

#### 4.3.3.5 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

## 4.4 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](#) ()

### Static Public Attributes

- static final boolean [ENABLE](#) = true
- static final boolean [DISABLE](#) = false
- static final boolean [DEF\\_FUNCTION](#) = DEF\_BOOLEAN
- static final boolean [DEF\\_FEATURE\\_AV](#) = DEF\_BOOLEAN
- static final String [DEF\\_FEATURE](#) = DEF\_STRING
- static final String [DEF\\_NAME](#) = DEF\_STRING
- static final int [DEF\\_NUMBER](#) = DEF\_INT
- static final String [DEF\\_PORT](#) = DEF\_STRING

#### 4.4.1 Detailed Description

Basic pin object.

- [Pin](#) necessary characteristics:
  - Name
  - Number

- [Pin](#) optional characteristics:

- Port

- [Pin](#) main functions:

- VCC

- GND

- GPIO

- [Pin](#) features:

- Interruption

- ADC

- UART

- I2C

- SPI

- Clock

- Reset

**Author**

Miguel Diaz

**Version**

0.1

## 4.4.2 Constructor & Destructor Documentation

### 4.4.2.1 [Pin\(\)](#)

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

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

## 4.4.3 Member Function Documentation

### 4.4.3.1 [getAdc\(\)](#)

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

Get the pin's ADC name

**Returns**

[Pin](#)'s ADC

#### 4.4.3.2 `getClock()`

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

Get the pin's clock name

##### Returns

[Pin's clock](#)

#### 4.4.3.3 `getFeat_adc()`

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

See if the pin has an ADC

##### Returns

Feature availability

#### 4.4.3.4 `getFeat_clock()`

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

See if the pin supports a clock

##### Returns

Feature availability

#### 4.4.3.5 `getFeat_i2c()`

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

See if the pin has I2C

##### Returns

Feature availability

#### 4.4.3.6 getFeat\_int()

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

See if the pin has an interruption

##### Returns

Feature availability

#### 4.4.3.7 getFeat\_reset()

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

See if the pin has a reset feature

##### Returns

Feature availability

#### 4.4.3.8 getFeat\_spi()

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

See if the pin has SPI

##### Returns

Feature availability

#### 4.4.3.9 getFeat\_timer()

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

See if the pin supports a timer

##### Returns

Feature availability

#### 4.4.3.10 `getFeat_uart()`

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

See if the pin has a UART

##### Returns

Feature availability

#### 4.4.3.11 `getFunc_gnd()`

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

See if the pin is GND

##### Returns

Function availability

#### 4.4.3.12 `getFunc_gpio()`

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

See if the pin is GPIO

##### Returns

Function availability

#### 4.4.3.13 `getFunc_vcc()`

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

See if the pin is Vcc

##### Returns

Function availability

#### 4.4.3.14 `getI2c()`

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

Get the pin's I2C name

Returns

Pin's I2C

#### 4.4.3.15 `getInt()`

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

Get the pin's interruption name

Returns

Pin's interruption

#### 4.4.3.16 `getName()`

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

Get the pin's name

Returns

Pin's name

#### 4.4.3.17 `getNumber()`

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

Get the pin's number

Returns

Pin's number

#### 4.4.3.18 `getPort()`

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

Get the pin's port

##### Returns

[Pin's port](#)

#### 4.4.3.19 `getReset()`

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

Get the pin's reset name

##### Returns

[Pin's reset](#)

#### 4.4.3.20 `getSpi()`

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

Get the pin's SPI name

##### Returns

[Pin's SPI](#)

#### 4.4.3.21 `getTimer()`

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

Get the pin's timer name

##### Returns

[Pin's timer](#)



#### 4.4.3.22 getUart()

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

Get the pin's UART name

Returns

Pin's UART

#### 4.4.3.23 isValid()

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

Check if the pin is correctly initialized

Returns

true if the pin is correctly initialized

#### 4.4.3.24 setAdc()

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

Set the pin's ADC

Parameters

<i>feature</i>	Pin's ADC
----------------	-----------

#### 4.4.3.25 setClock()

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

Set the pin's clock

Parameters

<i>feature</i>	Pin's clock
----------------	-------------

#### 4.4.3.26 setFeat\_adc()

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

Set the pin's ADC feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.27 setFeat\_clock()

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

Set the pin's Clock feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.28 setFeat\_i2c()

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

Set the pin's I2C feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.29 setFeat\_int()

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

Set the pin's interruption feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.30 setFeat\_reset()

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

Set the pin's reset feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.31 setFeat\_spi()

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

Set the pin's SPI feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.32 setFeat\_timer()

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

Set the pin's timer feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.33 setFeat\_uart()

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

Set the pin's UART feature

##### Parameters

<i>featState</i>	Feature availability
------------------	----------------------

#### 4.4.3.34 setFunc\_gnd()

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

Set the pin to GND status

##### Parameters

<i>funcState</i>	Function availability
------------------	-----------------------

#### 4.4.3.35 setFunc\_gpio()

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

Set the pin to GPIO status

##### Parameters

<i>funcState</i>	Function availability
------------------	-----------------------

#### 4.4.3.36 setFunc\_vcc()

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

Set the pin to Vcc status

##### Parameters

<i>funcState</i>	Function availability
------------------	-----------------------

#### 4.4.3.37 setI2c()

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

Set the pin's I2C

##### Parameters

<i>feature</i>	Pin's I2C
----------------	-----------

#### 4.4.3.38 setInt()

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

Set the pin's interruption

##### Parameters

<i>feature</i>	Pin's interruption
----------------	--------------------

#### 4.4.3.39 setName()

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

Set the pin's name

##### Parameters

<i>pinName</i>	Pin's name
----------------	------------

#### 4.4.3.40 setNumber()

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

Set the pin's number

##### Parameters

<i>pinNum</i>	Pin's number
---------------	--------------

#### 4.4.3.41 setPort()

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

Set the pin's port

##### Parameters

<i>pinPort</i>	Pin's port
----------------	------------

#### 4.4.3.42 setReset()

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

Set the pin's reset

##### Parameters

<i>feature</i>	<a href="#">Pin's reset</a>
----------------	-----------------------------

#### 4.4.3.43 setSpi()

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

Set the pin's SPI

##### Parameters

<i>feature</i>	<a href="#">Pin's SPI</a>
----------------	---------------------------

#### 4.4.3.44 setTimer()

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

Set the pin's timer

##### Parameters

<i>feature</i>	<a href="#">Pin's timer</a>
----------------	-----------------------------

#### 4.4.3.45 setUart()

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

Set the pin's UART

##### Parameters

<i>feature</i>	<a href="#">Pin's UART</a>
----------------	----------------------------

#### 4.4.4 Member Data Documentation

##### 4.4.4.1 DEF\_FEATURE

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

Default value for pin's feature as not available

##### 4.4.4.2 DEF\_FEATURE\_AV

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

Default value for pin's feature availability as not available

##### 4.4.4.3 DEF\_FUNCTION

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

Default value for pin's function as not enabled

##### 4.4.4.4 DEF\_NAME

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

Default value for pin's name

##### 4.4.4.5 DEF\_NUMBER

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

Default value for pin's number

##### 4.4.4.6 DEF\_PORT

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

Default value for pin's port

##### 4.4.4.7 DISABLE

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

Disable value for features and functions

#### 4.4.4.8 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

## 4.5 xmlParser.TestMain Class Reference

### Static Public Member Functions

- static void [main](#) (String[] openOption)

#### 4.5.1 Detailed Description

Dummy main class for testing the other classes

#### Author

Miguel Diaz

#### 4.5.2 Member Function Documentation

##### 4.5.2.1 main()

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

Main without GUI

#### Parameters

<i>openOption</i>	Options include:
-------------------	---------------------

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

- src/xmlParser/TestMain.java



## 4.6 xmlParser.XmlOpener Class Reference

### Public Member Functions

- [XmlOpener](#) ()
- [ErrorCode OpenFile](#) (String fileName)
- Document [getParsedDoc](#) ()

### 4.6.1 Detailed Description

Open and process XML files

#### Author

H112943

#### Version

0.1

### 4.6.2 Constructor & Destructor Documentation

#### 4.6.2.1 XmlOpener()

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

Constructor

### 4.6.3 Member Function Documentation

#### 4.6.3.1 getParsedDoc()

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

Get the parsed document AFTER opening the file

#### Returns

Parsed document

#### 4.6.3.2 OpenFile()

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

Open the XML file

**Parameters**

<i>fileName</i>	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

- common, [5](#)
- common.ErrorCode, [7](#)
- common.Features, [8](#)
- common::ErrorCode
  - NO\_ERROR, [7](#)
- common::Features
  - DEBUG\_STR, [8](#)
  - DEBUG, [8](#)
  - VERBOSE\_STR, [8](#)
  - VERBOSE, [8](#)
- DEBUG\_STR
  - common::Features, [8](#)
- DEBUG
  - common::Features, [8](#)
- DEF\_FEATURE\_AV
  - microcontroller::Pin, [25](#)
- DEF\_FEATURE
  - microcontroller::Pin, [25](#)
- DEF\_FUNCTION
  - microcontroller::Pin, [25](#)
- DEF\_NAME
  - microcontroller::Pin, [25](#)
- DEF\_NUMBER
  - microcontroller::Pin, [25](#)
- DEF\_PORT
  - microcontroller::Pin, [25](#)
- DISABLE
  - microcontroller::Pin, [25](#)
- ENABLE
  - microcontroller::Pin, [25](#)
- getAdc
  - microcontroller::Pin, [13](#)
- getClock
  - microcontroller::Pin, [13](#)
- getFeat\_adc
  - microcontroller::Pin, [14](#)
- getFeat\_clock
  - microcontroller::Pin, [14](#)
- getFeat\_i2c
  - microcontroller::Pin, [14](#)
- getFeat\_int
  - microcontroller::Pin, [14](#)
- getFeat\_reset
  - microcontroller::Pin, [15](#)
- getFeat\_spi
  - microcontroller::Pin, [15](#)
- getFeat\_timer
  - microcontroller::Pin, [15](#)
- getFeat\_uart
  - microcontroller::Pin, [15](#)
- getFunc\_gnd
  - microcontroller::Pin, [16](#)
- getFunc\_gpio
  - microcontroller::Pin, [16](#)
- getFunc\_vcc
  - microcontroller::Pin, [16](#)
- getI2c
  - microcontroller::Pin, [16](#)
- getInt
  - microcontroller::Pin, [17](#)
- getName
  - microcontroller::Pin, [17](#)
- getNumber
  - microcontroller::Pin, [17](#)
- getParsedDoc
  - xmlParser::XmlOpener, [27](#)
- getPin
  - microcontroller::Microcontroller, [10](#)
- getPort
  - microcontroller::Pin, [17](#)
- getReset
  - microcontroller::Pin, [18](#)
- getSpi
  - microcontroller::Pin, [18](#)
- getTimer
  - microcontroller::Pin, [18](#)
- getUart
  - microcontroller::Pin, [18](#)
- getUc\_manufacturer
  - microcontroller::Microcontroller, [10](#)
- getUc\_model
  - microcontroller::Microcontroller, [10](#)
- getUc\_pinNum
  - microcontroller::Microcontroller, [10](#)
- isValid
  - microcontroller::Pin, [19](#)
- main

- xmlParser::TestMain, 26
- Microcontroller
  - microcontroller::Microcontroller, 9
- microcontroller, 5
- microcontroller.Microcontroller, 9
- microcontroller.Pin, 11
- microcontroller::Microcontroller
  - getPin, 10
  - getUc\_manufacturer, 10
  - getUc\_model, 10
  - getUc\_pinNum, 10
  - Microcontroller, 9
  - processDocument, 11
- microcontroller::Pin
  - DEF\_FEATURE\_AV, 25
  - DEF\_FEATURE, 25
  - DEF\_FUNCTION, 25
  - DEF\_NAME, 25
  - DEF\_NUMBER, 25
  - DEF\_PORT, 25
  - DISABLE, 25
  - ENABLE, 25
  - getAdc, 13
  - getClock, 13
  - getFeat\_adc, 14
  - getFeat\_clock, 14
  - getFeat\_i2c, 14
  - getFeat\_int, 14
  - getFeat\_reset, 15
  - getFeat\_spi, 15
  - getFeat\_timer, 15
  - getFeat\_uart, 15
  - getFunc\_gnd, 16
  - getFunc\_gpio, 16
  - getFunc\_vcc, 16
  - getI2c, 16
  - getInt, 17
  - getName, 17
  - getNumber, 17
  - getPort, 17
  - getReset, 18
  - getSpi, 18
  - getTimer, 18
  - getUart, 18
  - isValid, 19
  - Pin, 13
  - setAdc, 19
  - setClock, 19
  - setFeat\_adc, 19
  - setFeat\_clock, 20
  - setFeat\_i2c, 20
  - setFeat\_int, 20
  - setFeat\_reset, 21
  - setFeat\_spi, 21
  - setFeat\_timer, 21
  - setFeat\_uart, 21
  - setFunc\_gnd, 22
  - setFunc\_gpio, 22
  - setFunc\_vcc, 22
  - setI2c, 22
  - setInt, 23
  - setName, 23
  - setNumber, 23
  - setPort, 23
  - setReset, 24
  - setSpi, 24
  - setTimer, 24
  - setUart, 24
- NO\_ERROR
  - common::ErrorCode, 7
- OpenFile
  - xmlParser::XmlOpener, 27
- Pin
  - microcontroller::Pin, 13
- processDocument
  - microcontroller::Microcontroller, 11
- setAdc
  - microcontroller::Pin, 19
- setClock
  - microcontroller::Pin, 19
- setFeat\_adc
  - microcontroller::Pin, 19
- setFeat\_clock
  - microcontroller::Pin, 20
- setFeat\_i2c
  - microcontroller::Pin, 20
- setFeat\_int
  - microcontroller::Pin, 20
- setFeat\_reset
  - microcontroller::Pin, 21
- setFeat\_spi
  - microcontroller::Pin, 21
- setFeat\_timer
  - microcontroller::Pin, 21
- setFeat\_uart
  - microcontroller::Pin, 21
- setFunc\_gnd
  - microcontroller::Pin, 22
- setFunc\_gpio
  - microcontroller::Pin, 22
- setFunc\_vcc
  - microcontroller::Pin, 22
- setI2c
  - microcontroller::Pin, 22
- setInt

- microcontroller::Pin, [23](#)
- setName
  - microcontroller::Pin, [23](#)
- setNumber
  - microcontroller::Pin, [23](#)
- setPort
  - microcontroller::Pin, [23](#)
- setReset
  - microcontroller::Pin, [24](#)
- setSpi
  - microcontroller::Pin, [24](#)
- setTimer
  - microcontroller::Pin, [24](#)
- setUart
  - microcontroller::Pin, [24](#)
- VERBOSE\_STR
  - common::Features, [8](#)
- VERBOSE
  - common::Features, [8](#)
- XmlOpener
  - xmlParser::XmlOpener, [27](#)
- xmlParser, [6](#)
- xmlParser.TestMain, [26](#)
- xmlParser.XmlOpener, [27](#)
- xmlParser::TestMain
  - main, [26](#)
- xmlParser::XmlOpener
  - getParsedDoc, [27](#)
  - OpenFile, [27](#)
  - XmlOpener, [27](#)