

# ArduinoSerialConfigurationMenuLibrary

Generated by Doxygen 1.7.6.1

Sun Jul 1 2012 15:00:44

## Contents

<b>1</b>	<b><a href="#">Class Index</a></b>	<b>1</b>
1.1	<a href="#">Class List</a>	1
<b>2</b>	<b><a href="#">Class Documentation</a></b>	<b>1</b>
2.1	<a href="#">SerialConfigMenu Class Reference</a>	1
2.1.1	<a href="#">Constructor &amp; Destructor Documentation</a>	2
2.1.2	<a href="#">Member Function Documentation</a>	2

## 1 Class Index

### 1.1 Class List

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

<a href="#">SerialConfigMenu</a>	1
----------------------------------	---

## 2 Class Documentation

### 2.1 SerialConfigMenu Class Reference

#### Public Member Functions

- [SerialConfigMenu](#) ()  
*Constructor.*
- void [begin](#) ()  
*Begin variables: default value for serial speed: 115200.*
- void [begin](#) (long speed)  
*Begin using custom speed settings for the serial port.*
- void [show](#) ()  
*Show menu empty menu.*
- void [show](#) (char name)  
*Show menu: specify a menu name: d -> default, e -> empty.*
- void [getLine](#) (uint8\_t stringIndex)  
*Get one line of text from the program memory and put it into the buffer.*
- void [printLine](#) (uint8\_t line)  
*Print one line of text on the terminal window.*
- void [goHome](#) ()  
*Place the cursor in the beginning of the serial window.*
- void [clearScreen](#) ()

*Clear the serial port window.*

- char `readChar` ()

*Read a char from the serial port.*

- char `readCharBlocking` ()

*Read a char from the serial port (blocking)*

- void `readLine` ()

*Read a line from the serial port (read until 'max length' reached or ENTER pressed)  
TODO this does not work correctly yet.*

- void `printLine` ()

*Print a line of text.*

### 2.1.1 Constructor & Destructor Documentation

#### 2.1.1.1 SerialConfigMenu::SerialConfigMenu ( )

Constructor.

### 2.1.2 Member Function Documentation

#### 2.1.2.1 void SerialConfigMenu::begin ( )

Begin variables: default value for serial speed: 115200.

#### 2.1.2.2 void SerialConfigMenu::begin ( long *speed* )

Begin using custom speed settings for the serial port.

#### 2.1.2.3 void SerialConfigMenu::clearScreen ( )

Clear the serial port window.

#### 2.1.2.4 void SerialConfigMenu::getLine ( uint8\_t *stringIndex* )

Get one line of text from the program memory and put it into the buffer.

#### 2.1.2.5 void SerialConfigMenu::goHome ( )

Place the cursor in the beginning of the serial window.

#### 2.1.2.6 void SerialConfigMenu::printLine ( uint8\_t *line* )

Print one line of text on the terminal window.

#### 2.1.2.7 void SerialConfigMenu::printLine ( )

Print a line of text.

#### 2.1.2.8 char SerialConfigMenu::readChar ( )

Read a char from the serial port.

#### 2.1.2.9 char SerialConfigMenu::readCharBlocking ( )

Read a char from the serial port (blocking)

#### 2.1.2.10 void SerialConfigMenu::readLine ( )

Read a line from the serial port (read until 'max length' reached or ENTER pressed)  
TODO this does not work correctly yet.

#### 2.1.2.11 void SerialConfigMenu::show ( )

Show menu empty menu.

#### 2.1.2.12 void SerialConfigMenu::show ( char *name* )

Show menu: specify a menu name: d -> default, e -> empty.