

ezLCD Python Module

1.02

Generated by Doxygen 1.8.3.1

Tue Jul 16 2013 12:04:23

Contents

1	Module Index	1
1.1	Modules	1
2	Namespace Index	3
2.1	Namespace List	3
3	Hierarchical Index	5
3.1	Class Hierarchy	5
4	Class Index	7
4.1	Class List	7
5	Module Documentation	9
5.1	Commands	9
5.1.1	Detailed Description	10
5.1.2	Function Documentation	10
5.1.2.1	backlight	10
5.1.2.2	cfgio	10
5.1.2.3	io	10
5.1.2.4	ping	10
5.1.2.5	play	10
5.1.2.6	reset	11
5.1.2.7	run	11
5.1.2.8	snapshot	11
5.1.2.9	verbose	11
5.1.2.10	wquiet	11
5.1.2.11	xmax	11
5.1.2.12	ymax	12
5.2	Primitive Drawing Commands	13
5.2.1	Detailed Description	13

5.2.2	Function Documentation	13
5.2.2.1	arc	13
5.2.2.2	box	13
5.2.2.3	circle	14
5.2.2.4	clipArea	14
5.2.2.5	clipEnable	14
5.2.2.6	cls	14
5.2.2.7	color	14
5.2.2.8	colorId	14
5.2.2.9	line	15
5.2.2.10	lineType	15
5.2.2.11	lineWidth	15
5.2.2.12	pie	15
5.2.2.13	plot	15
5.2.2.14	xy	15
5.3	Widgets	17
5.3.1	Detailed Description	17
5.3.2	Function Documentation	17
5.3.2.1	ameter	17
5.3.2.2	ameter_color	18
5.3.2.3	button	18
5.3.2.4	choice	18
5.3.2.5	dial	19
5.3.2.6	dmeter	19
5.3.2.7	fontw	19
5.3.2.8	groupBox	20
5.3.2.9	progressBar	20
5.3.2.10	radioButton	20
5.3.2.11	slider	21
5.3.2.12	staticText	21
5.3.2.13	string	21
5.3.2.14	theme	22
5.3.2.15	touchZone	22
5.3.2.16	wstack	22
5.3.2.17	wstate	22
5.3.2.18	wvalue	23
5.4	Bitmaps and Fonts	24

5.4.1	Detailed Description	24
5.4.2	Function Documentation	24
5.4.2.1	font	24
5.4.2.2	fonto	24
5.4.2.3	picture	24
5.4.2.4	printChar	25
5.4.2.5	printString	25
6	Namespace Documentation	27
6.1	ezLCD3xx Namespace Reference	27
6.1.1	Detailed Description	30
6.1.2	Function Documentation	30
6.1.2.1	__init__	30
6.1.2.2	closeSerial	30
6.1.2.3	WaitForCR	30
7	Class Documentation	31
7.1	ezLCD3xx.ezLCD Class Reference	31
Index		31

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

Commands	9
Primitive Drawing Commands	13
Widgets	17
Bitmaps and Fonts	24

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

ezLCD3xx	27
------------------------------------	--------------------

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

object	
ezLCD3xx.ezLCD	31

Chapter 4

Class Index

4.1 Class List

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

ezLCD3xx.ezLCD	31
--	----

Chapter 5

Module Documentation

5.1 Commands

Functions

- def [ezLCD3xx.verbose](#)
The Verbose command will turn on or off more verbose errors.
- def [ezLCD3xx.xmax](#)
The xmax command will return the max x of current display.
- def [ezLCD3xx.ymax](#)
The ymax command will return the max y of current display.
- def [ezLCD3xx.ping](#)
the ping command
- def [ezLCD3xx.backlight](#)
The backlight command will set backlight brightness and timeout.
- def [ezLCD3xx.wquiet](#)
The wquiet command disables the touch event data being sent to the console port.
- def [ezLCD3xx.cfgio](#)
The cfgio command will configure io pins.
- def [ezLCD3xx.io](#)
The io command use to set and clear io pins.
- def [ezLCD3xx.play](#)
The play command will play a macro stored on the drive of the [ezLCD](#).
- def [ezLCD3xx.run](#)
The run command will run a macro stored on the drive of the [ezLCD](#).
- def [ezLCD3xx.reset](#)
The reset command will reset the [ezLCD](#) and run startup.ezm same as power up.
- def [ezLCD3xx.snapshot](#)
The snapshot command will write a copy of the current display to the flash drive as a bmp.

5.1.1 Detailed Description

5.1.2 Function Documentation

5.1.2.1 `def ezLCD3xx.backlight (self, brightness, timeout=None, level=None)`

The backlight command will set backlight brightness and timeout.

Parameters

<i>brightness</i>	1
<i>timeout</i>	2
<i>level</i>	3

5.1.2.2 `def ezLCD3xx.cfgio (self, pin, function)`

The cfgio command will configure io pins.

Parameters

<i>pin</i>	
<i>function</i>	

5.1.2.3 `def ezLCD3xx.io (self, pin, level)`

The io command use to set and clear io pins.

Parameters

<i>pin</i>	
<i>level</i>	

5.1.2.4 `def ezLCD3xx.ping (self)`

the ping command

Returns

0

5.1.2.5 `def ezLCD3xx.play (self, filename)`

The play command will play a macro stored on the drive of the [ezLCD](#).

Parameters

<i>macro</i>	filename
--------------	----------

5.1.2.6 `def ezLCD3xx.reset (self)`

The reset command will reset the `ezLCD` and run `startup.ezm` same as power up.

5.1.2.7 `def ezLCD3xx.run (self, filename)`

The run command will run a macro stored on the drive of the `ezLCD`.

Parameters

<i>macro</i>	filename
--------------	----------

5.1.2.8 `def ezLCD3xx.snapshot (self, x, y, w, h, filename)`

The snapshot command will write a copy of the current display to the flash drive as a bmp.

Parameters

<i>x</i>	
<i>y</i>	
<i>w</i>	
<i>h</i>	
<i>filename.bmp</i>	

5.1.2.9 `def ezLCD3xx.verbose (self, state)`

The Verbose command will turn on or off more verbose errors.

Parameters

<i>state</i>	0=off 1=on
--------------	------------

5.1.2.10 `def ezLCD3xx.wquiet (self, state)`

The wquiet command disables the touch event data being sent to the console port.

Parameters

<i>state</i>	0=off 1=on
--------------	------------

5.1.2.11 `def ezLCD3xx.xmax (self)`

The xmax command will return the max x of current display.

Returns

x-horizontal resolution in pixels starting from 0

5.1.2.12 `def ezLCD3xx.ymax (self)`

The ymax command will return the max y of current display.

Returns

y-vertical resolution in pixels starting from 0

5.2 Primitive Drawing Commands

Functions

- def [ezLCD3xx.cls](#)
The cls command will clear the screen to black if no color is given.
- def [ezLCD3xx.color](#)
The color command see [ezLCD3xx](#) manual for colors.
- def [ezLCD3xx.colorId](#)
The colorId command.
- def [ezLCD3xx.xy](#)
The xy command will set or return the x y coordinates.
- def [ezLCD3xx.plot](#)
The plot command will set a pixel to current color and if used x y.
- def [ezLCD3xx.lineType](#)
The lineType Command will set the line type for the line command.
- def [ezLCD3xx.lineWidth](#)
The lineWidth Command will set the line width for the line command.
- def [ezLCD3xx.line](#)
The line command will draw a line from current xy to line(x,y)
- def [ezLCD3xx.box](#)
The box command will draw a box starting from the current xy in width and height with option for filled.
- def [ezLCD3xx.circle](#)
- def [ezLCD3xx.pie](#)
- def [ezLCD3xx.arc](#)
- def [ezLCD3xx.clipArea](#)
- def [ezLCD3xx.clipEnable](#)

5.2.1 Detailed Description

5.2.2 Function Documentation

5.2.2.1 def [ezLCD3xx.arc](#) (*self*, *radius*, *start*, *end*, *fill* = 0)

The ARC command draws an arc at current XY position. Replace {R} with the desired radius of the arc, in pixels. *fill* =1 will draw a filled circle

5.2.2.2 def [ezLCD3xx.box](#) (*self*, *width*, *height*, *fill* = 0)

The box command will draw a box starting from the current xy in width and height with option for filled.

Parameters

<i>width</i>	width of box in pixels
<i>height</i>	height of box in pixels
<i>fill</i>	1=filled box 0=outline only *optional defaults to outline

5.2.2.3 `def ezLCD3xx.circle (self, width, radius, fill = 0)`

draw a circle with radius in pixels
fill =1 will draw a filled circle

5.2.2.4 `def ezLCD3xx.clipArea (self, left, top, right, bottom)`

Parameters

<i>left</i>	
<i>top</i>	
<i>right</i>	
<i>bottom</i>	

5.2.2.5 `def ezLCD3xx.clipEnable (self, enable)`

Parameters

<i>enable</i>	0=off 1=on
---------------	------------

5.2.2.6 `def ezLCD3xx.cls (self, Color = None)`

The cls command will clear the screen to black if no color is given.

Parameters

<i>Color</i>	color to clear screen to
--------------	--------------------------

5.2.2.7 `def ezLCD3xx.color (self, color = None)`

The color command see [ezLCD3xx](#) manual for colors.

Parameters

<i>color</i>	number
--------------	--------

Returns

color as a tuple

5.2.2.8 `def ezLCD3xx.colorId (self, ID, R = None, G = None, B = None)`

The colorId command.

Parameters

<i>R</i>	
<i>G</i>	
<i>B</i>	

Returns

color as a tuple

5.2.2.9 `def ezLCD3xx.line (self, x, y)`

The line command will draw a line from current xy to line(x,y)

Parameters

<i>x</i>	
<i>y</i>	

5.2.2.10 `def ezLCD3xx.lineType (self, option)`

The lineType Command will set the line type for the line command.

Parameters

<i>option</i>	0 = solid, 1= dotted (1 pixel spacing between dots), 2 = dashed (2 pixel spacing between dashes)
---------------	--

5.2.2.11 `def ezLCD3xx.lineWidth (self, width)`

The lineWidth Command will set the line width for the line command.

Parameters

<i>width</i>	thin line (width = 1) or a thick line (width =3). Only [width] = 1 or 3 are available.
--------------	--

5.2.2.12 `def ezLCD3xx.pie (self, radius, start, end)`

The PIE command draws a section of a circle (pie slice) at current xy position.

5.2.2.13 `def ezLCD3xx.plot (self, x=None, y=None)`

The plot command will set a pixel to current color and if used x y.

Parameters

<i>x</i>	optional
<i>y</i>	optional

5.2.2.14 `def ezLCD3xx.xy (self, x=None, y=None)`

The xy command will set or return the x y coordinates.

Parameters

x	optional
y	optional

5.3 Widgets

Functions

- def `ezLCD3xx.ameter`
The ameter widget.
- def `ezLCD3xx.ameter_color`
The ameter_color command.
- def `ezLCD3xx.dmeter`
The dmeter widget.
- def `ezLCD3xx.button`
The button command.
- def `ezLCD3xx.choice`
The choice widget allows you to print a string and display buttons for the user to choose a response.
- def `ezLCD3xx.groupBox`
The groupBox widget.
- def `ezLCD3xx.radioButton`
The radioButton widget.
- def `ezLCD3xx.staticText`
The staticText widget.
- def `ezLCD3xx.slider`
The slider command.
- def `ezLCD3xx.progressBar`
The progressBar command.
- def `ezLCD3xx.touchZone`
The touchZone command.
- def `ezLCD3xx.dial`
The dial command.
- def `ezLCD3xx.theme`
The theme command sets the colors for widgets.
- def `ezLCD3xx.fontw`
The fontW command will set the font for widget.
- def `ezLCD3xx.string`
The string command will set or return a internal string.
- def `ezLCD3xx.wstack`
The wstack command will return the stack of widgets pressed.
- def `ezLCD3xx.wvalue`
The wvalue command will set or return a value to or from a widget.
- def `ezLCD3xx.wstate`
The wstate command.

5.3.1 Detailed Description

5.3.2 Function Documentation

5.3.2.1 `def ezLCD3xx.ameter (self, ID, x, y, width, height, options, value, minV, maxV, theme, stringID, meterType = 0)`

The ameter widget.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	
<i>value</i>	
<i>minV</i>	
<i>maxV</i>	
<i>theme</i>	
<i>stringID</i>	
<i>meter- Type@verbatim</i>	ameter [ID][x][y][width][height][options][value][minV][maxV][theme][stringID][type]

5.3.2.2 `def ezLCD3xx.ameter_color (self, ID, color1, color2, color3, color4, color5, color6)`

The ameter_color command.

Parameters

<i>ID</i>	
<i>color1</i>	
<i>color2</i>	
<i>color3</i>	
<i>color4</i>	
<i>color5</i>	
<i>color6</i>	

5.3.2.3 `def ezLCD3xx.button (self, ID, x, y, width, height, options, align, radius, theme, stringID)`

The button command.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	
<i>align</i>	
<i>radius</i>	
<i>theme</i>	
<i>stringID</i>	

5.3.2.4 `def ezLCD3xx.choice (self, string, theme, string1=None, string2=None, string3=None)`

The choice widget allows you to print a string and display buttons for the user to choose a response.

Parameters

<i>string</i>	the text about the buttons
<i>theme</i>	the theme ID
<i>string1</i>	string for left button *optional defaults to YES
<i>string2</i>	string for center button *optional defaults to NO
<i>string3</i>	string for right button *optional defaults to CANCEL

Returns

1=left button
 0=center button
 -1=right button

5.3.2.5 `def ezLCD3xx.dial (self, ID, x, y, radius, option, resolution, value, maxx, theme)`

The dial command.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>radius</i>	
<i>option</i>	
<i>resolution</i>	
<i>value</i>	
<i>maxx</i>	
<i>theme@verbatim</i>	dial [ID][x][y][radius][option][resolution][value][max][theme]

5.3.2.6 `def ezLCD3xx.dmeter (self, ID, x, y, width, height, options, value, digits, dp, theme)`

The dmeter widget.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	
<i>value</i>	
<i>digits</i>	
<i>dp</i>	
<i>theme</i>	

5.3.2.7 `def ezLCD3xx.fontw (self, fontnumber, name)`

The fontW command will set the font for widget.

Parameters

<i>fontnumber</i>	number of the font
<i>name</i>	filename of font '0' and '1' are internal fonts

5.3.2.8 `def ezLCD3xx.groupBox (self, ID, x, y, width, height, options, theme, stringID)`

The groupBox widget.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	
<i>theme</i>	
<i>stringID</i>	

5.3.2.9 `def ezLCD3xx.progressBar (self, ID, x, y, width, height, options, value, mmax, theme, stringID)`

The progressBar command.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	
<i>value</i>	
<i>mmax</i>	
<i>theme</i>	
<i>stringID</i>	

5.3.2.10 `def ezLCD3xx.radioButton (self, ID, x, y, width, height, options, theme, stringID)`

The radioButton widget.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	Options: 1=draw , 2=disabled, 3=checked, 4=first, 5=first and checked.
<i>theme</i>	
<i>stringID</i>	

5.3.2.11 `def ezLCD3xx.slider(self, ID, x, y, width, height, options, rrange, resolution, value, theme)`

The slider command.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	
<i>rrange</i>	
<i>resolution</i>	
<i>value</i>	
<i>theme</i>	

5.3.2.12 `def ezLCD3xx.staticText(self, ID, x, y, width, height, options, theme, stringID)`

The staticText widget.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	Options: 1=left, 2=disabled , 3=right , 4=center, 5=left framed, 6=disabled framed, 7=right framed, 8=center framed , 9=redraw text.
<i>theme</i>	
<i>stringID</i>	

5.3.2.13 `def ezLCD3xx.string(self, stringNumber, string = None)`

The string command will set or return a internal string.

Parameters

<i>stringNumber</i>	number of string to set or return
<i>string</i>	string to set optional internal strings are used for text on buttons and other widgets Strings are defined as 128 characters. There are 64 strings (0 to 63). String 61-63 are used by the CHOICE command. String 64 is temp location. String 65 is the product string String 66 is the firmware string

```
5.3.2.14 def ezLCD3xx.theme ( self, ID, EmbossDkColor, EmbossLtColor, TextColor0, TextColor1, TextColorDisabled, Color0,
                             Color1, ColorDisabled, CommonBkColor, Fontw )
```

The theme command sets the colors for widgets.

Parameters

<i>ID</i>	Theme ID
<i>EmbossDkColor</i>	Dark color for 3d effect
<i>EmbossLtColor</i>	Light color for 3d effect
<i>TextColor0</i>	
<i>TextColor1</i>	
<i>TextColorDisabled</i>	
<i>Color0</i>	
<i>Color1</i>	
<i>ColorDisabled</i>	
<i>CommonBkColor</i>	
<i>Fontw</i>	widget font for theme

```
5.3.2.15 def ezLCD3xx.touchZone ( self, ID, x, y, width, height, options )
```

The touchZone command.

Parameters

<i>ID</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>options</i>	

```
5.3.2.16 def ezLCD3xx.wstack ( self, option )
```

The wstack command will return the stack of widgets pressed.

Parameters

<i>option</i>	0=FIFO 1=LIFO 2=CLEAR
---------------	-----------------------

```
5.3.2.17 def ezLCD3xx.wstate ( self, ID )
```

The wstate command.

Parameters

<i>ID</i>	
-----------	--

5.3.2.18 `def ezLCD3xx.wvalue (self, ID, value = None)`

The wvalue command will set or return a value to or from a widget.

Parameters

<i>ID</i>	
<i>value</i>	

5.4 Bitmaps and Fonts

Functions

- def `ezLCD3xx.picture`
The picture command will display a bitmap in bmp, jpg, gif formats with optional coordinates.
- def `ezLCD3xx.font`
The font command will set current font to use.
- def `ezLCD3xx.fonto`
The FONTO command will change the orientation or direction the text prints.
- def `ezLCD3xx.printChar`
- def `ezLCD3xx.printString`
print string in current color and font and optional coordinates

5.4.1 Detailed Description

5.4.2 Function Documentation

5.4.2.1 `def ezLCD3xx.font (self, font)`

The font command will set current font to use.

Parameters

<i>font</i>	font name '0' and '1' are internal fonts
-------------	---

5.4.2.2 `def ezLCD3xx.fonto (self, orientation)`

The FONTO command will change the orientation or direction the text prints.

Parameters

<i>orientation</i>	0=0 1=90 2=180 3=270
--------------------	----------------------

5.4.2.3 `def ezLCD3xx.picture (self, image, x=None, y=None)`

The picture command will display a bitmap in bmp, jpg, gif formats with optional coordinates.

Parameters

<i>image</i>	filename of image 'logo.gif'
<i>x</i>	x coordinates
<i>y</i>	y coordinates
	x y are optional and if not supplied will display image at current xy

5.4.2.4 `def ezLCD3xx.printChar (self, character)`

add here

5.4.2.5 `def ezLCD3xx.printString (self, string, x=None, y=None)`

print string in current color and font and optional coordinates

Parameters

<i>string</i>	string to print
<i>x</i>	x coordinates
<i>y</i>	y coordinates
	x y are optional and if not supplied will print string at current xy

Chapter 6

Namespace Documentation

6.1 ezLCD3xx Namespace Reference

Classes

- class [ezLCD](#)

Functions

- def [__init__](#)
ezLCD object
- def **openSerial**
- def [closeSerial](#)
- def [WaitForCR](#)
This is a internal use function.
- def [verbose](#)
The Verbose command will turn on or off more verbose errors.
- def [xmax](#)
The xmax command will return the max x of current display.
- def [ymax](#)
The ymax command will return the max y of current display.
- def [ping](#)
the ping command
- def [backlight](#)
The backlight command will set backlight brightness and timeout.
- def [wquiet](#)
The wquiet command disables the touch event data being sent to the console port.
- def [cfgio](#)
The cfgio command will configure io pins.
- def [io](#)
The io command use to set and clear io pins.
- def [play](#)
The play command will play a macro stored on the drive of the [ezLCD](#).
- def [run](#)

- The run command will run a macro stored on the drive of the [ezLCD](#).*

 - def [reset](#)

The reset command will reset the [ezLCD](#) and run startup.ezm same as power up.
 - def [snapshot](#)

The snapshot command will write a copy of the current display to the flash drive as a bmp.
 - def [cls](#)

The cls command will clear the screen to black if no color is given.
 - def [color](#)

The color command see [ezLCD3xx](#) manual for colors.
 - def [colorId](#)

The colorId command.
 - def [xy](#)

The xy command will set or return the x y coordinates.
 - def [plot](#)

The plot command will set a pixel to current color and if used x y.
 - def [lineType](#)

The lineType Command will set the line type for the line command.
 - def [lineWidth](#)

The lineWidth Command will set the line width for the line command.
 - def [line](#)

The line command will draw a line from current xy to line(x,y)
 - def [box](#)

The box command will draw a box starting from the current xy in width and height with option for filled.
 - def [circle](#)
 - def [pie](#)
 - def [arc](#)
 - def [clipArea](#)
 - def [clipEnable](#)
 - def [ameter](#)

The ameter widget.
 - def [ameter_color](#)

The ameter_color command.
 - def [dmeter](#)

The dmeter widget.
 - def [button](#)

The button command.
 - def [choice](#)

The choice widget allows you to print a string and display buttons for the user to choose a response.
 - def [groupBox](#)

The groupBox widget.
 - def [radioButton](#)

The radioButton widget.
 - def [staticText](#)

The staticText widget.
 - def [slider](#)

The slider command.
 - def [progressBar](#)

- The progressBar command.*
- def [touchZone](#)
The touchZone command.
- def [dial](#)
The dial command.
- def [theme](#)
The theme command sets the colors for widgets.
- def [fontw](#)
The fontW command will set the font for widget.
- def [string](#)
The string command will set or return a internal string.
- def [wstack](#)
The wstack command will return the stack of widgets pressed.
- def [wvalue](#)
The wvalue command will set or return a value to or from a widget.
- def [wstate](#)
The wstate command.
- def [picture](#)
The picture command will display a bitmap in bmp, jpg, gif formats with optional coordinates.
- def [font](#)
The font command will set current font to use.
- def [fonto](#)
The FONTO command will change the orientation or direction the text prints.
- def [printChar](#)
- def [printString](#)
print string in current color and font and optional coordinates

Variables

- int **BLACK** = 0
- int **GRAY** = 1
- int **SILVER** = 2
- int **WHITE** = 3
- int **RED** = 4
- int **MAROON** = 5
- int **YELLOW** = 6
- int **OLIVE** = 7
- int **LIME** = 8
- int **GREEN** = 9
- int **AQUA** = 10
- int **TEAL** = 11
- int **BLUE** = 12
- int **NAVY** = 13
- int **FUCHISA** = 14
- int **PURPLE** = 15
- int **FILLED** = 1
- int **ON** = 1
- int **OFF** = 0

- int **FIFO** = 0
- int **LIFO** = 1
- int **CLEAR** = 2
- [interface](#)
 - open serial port*
- **ser**
- **sio**

6.1.1 Detailed Description

Python Module for earthlcd.com ezLCD 3xx line of displays
<http://earthlcd.com>

(c)2013 ken segler
ken@earthlcd.com
requires pySerial <http://pyserial.sourceforge.net/>

6.1.2 Function Documentation

6.1.2.1 `def ezLCD3xx.__init__(self, interface)`

[ezLCD](#) object

6.1.2.2 `def ezLCD3xx.closeSerial (self)`

close

6.1.2.3 `def ezLCD3xx.WaitForCR (self)`

This is a internal use function.

Chapter 7

Class Documentation

7.1 ezLCD3xx.ezLCD Class Reference

Inheritance diagram for ezLCD3xx.ezLCD:

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

- C:/Users/Segler/Documents/GitHub/ezLCD3xxPython/module/ezLCD3xx.py

Index

- `__init__`
 - `ezLCD3xx`, [30](#)
- `ameter`
 - Widgets, [17](#)
- `ameter_color`
 - Widgets, [18](#)
- `arc`
 - Primitive Drawing Commands, [13](#)
- `backlight`
 - Commands, [10](#)
- Bitmaps and Fonts, [24](#)
 - `font`, [24](#)
 - `fonto`, [24](#)
 - `picture`, [24](#)
 - `printChar`, [24](#)
 - `printString`, [25](#)
- `box`
 - Primitive Drawing Commands, [13](#)
- `button`
 - Widgets, [18](#)
- `cfgio`
 - Commands, [10](#)
- `choice`
 - Widgets, [18](#)
- `circle`
 - Primitive Drawing Commands, [13](#)
- `clipArea`
 - Primitive Drawing Commands, [14](#)
- `clipEnable`
 - Primitive Drawing Commands, [14](#)
- `closeSerial`
 - `ezLCD3xx`, [30](#)
- `cls`
 - Primitive Drawing Commands, [14](#)
- `color`
 - Primitive Drawing Commands, [14](#)
- `colorId`
 - Primitive Drawing Commands, [14](#)
- Commands, [9](#)
 - `backlight`, [10](#)
 - `cfgio`, [10](#)
 - `io`, [10](#)
 - `ping`, [10](#)
 - `play`, [10](#)
 - `reset`, [10](#)
 - `run`, [11](#)
 - `snapshot`, [11](#)
 - `verbose`, [11](#)
 - `wquiet`, [11](#)
 - `xmax`, [11](#)
 - `ymax`, [11](#)
- `dial`
 - Widgets, [19](#)
- `dmeter`
 - Widgets, [19](#)
- `ezLCD3xx`, [27](#)
 - `__init__`, [30](#)
 - `closeSerial`, [30](#)
 - `WaitForCR`, [30](#)
 - `ezLCD3xx.ezLCD`, [31](#)
- `font`
 - Bitmaps and Fonts, [24](#)
- `fonto`
 - Bitmaps and Fonts, [24](#)
- `fontw`
 - Widgets, [19](#)
- `groupBox`
 - Widgets, [20](#)
- `io`
 - Commands, [10](#)
- `line`
 - Primitive Drawing Commands, [15](#)
- `lineType`
 - Primitive Drawing Commands, [15](#)
- `lineWidth`
 - Primitive Drawing Commands, [15](#)
- `picture`
 - Bitmaps and Fonts, [24](#)
- `pie`
 - Primitive Drawing Commands, [15](#)
- `ping`
 - Commands, [10](#)
- `play`

- Commands, 10
- plot
 - Primitive Drawing Commands, 15
- Primitive Drawing Commands, 13
 - arc, 13
 - box, 13
 - circle, 13
 - clipArea, 14
 - clipEnable, 14
 - cls, 14
 - color, 14
 - colorId, 14
 - line, 15
 - lineType, 15
 - lineWidth, 15
 - pie, 15
 - plot, 15
 - xy, 15
- printChar
 - Bitmaps and Fonts, 24
- printString
 - Bitmaps and Fonts, 25
- progressBar
 - Widgets, 20
- radioButton
 - Widgets, 20
- reset
 - Commands, 10
- run
 - Commands, 11
- slider
 - Widgets, 21
- snapshot
 - Commands, 11
- staticText
 - Widgets, 21
- string
 - Widgets, 21
- theme
 - Widgets, 21
- touchZone
 - Widgets, 22
- verbose
 - Commands, 11
- WaitForCR
 - ezLCD3xx, 30
- Widgets, 17
 - ameter, 17
 - ameter_color, 18
 - button, 18
 - choice, 18
 - dial, 19
 - dmeter, 19
 - fontw, 19
 - groupBox, 20
 - progressBar, 20
 - radioButton, 20
 - slider, 21
 - staticText, 21
 - string, 21
 - theme, 21
 - touchZone, 22
 - wstack, 22
 - wstate, 22
 - wvalue, 22
- wquiet
 - Commands, 11
- wstack
 - Widgets, 22
- wstate
 - Widgets, 22
- wvalue
 - Widgets, 22
- xmax
 - Commands, 11
- xy
 - Primitive Drawing Commands, 15
- ymax
 - Commands, 11