ezLCD3xx Python Module Documentation

Release 1.0

Ken Segler

March 27, 2013

CONTENTS

Contents:

CHAPTER

ONE

PTYHON EZLCD3XX FUNCTIONS

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

requires pySerial

class `ezLCD3xx.`**`ezLCD`**(*interface*)

`Button`(*ID*, *x*, *y*, *width*, *heigth*, *options*, *align*, *radius*, *theme*, *stringID*)
button [ID][x][y][width][height][options][align][radius][theme][stringID]

`Dial`(*ID*, *x*, *y*, *radius*, *option*, *resolution*, *value*, *maxx*, *theme*)
dial [ID][x][y][radius][option][resolution][value][max][theme]

`Font`(*fontnumber*)
selectFont(fontnumber) fontnumber in firmware

`FontW`(*fontnumber*, *name*)
sets widget font

`Line`(*x*, *y*)
draw a horizontal line from current position to x, y

`Picture`(*name*)
display bitmap at current coordinates gif,bmp,jpg

`PictureXY`(*x*, *y*, *name*)
display bitmap at x y coordinates gif, bmp, jpg

`Slider`(*ID*, *x*, *y*, *width*, *heigth*, *options*, *rrange*, *resolution*, *value*, *theme*)
makes a slider

`String`(*stringNumber*, *string*)
sets string value for buttons ect.

`Verbose`(*state*)
turn verbose on and off . use off in this module

`WaitForCR`()

`ameter`(*ID*, *x*, *y*, *width*, *height*, *options*, *value*, *min*, *max*, *theme*, *stringID*, *type*)
ameter [ID][x][y][width][height][options][value][min][max][theme][stringID][type]

`box`(*width*, *height*, *fill*)
draw a box of ‘width’ and ‘height’ from current coordinates fill =1 will draw a filled box

circle (*width*, *radius*, *fill*)
: draw a circle with radius in pixels fill =1 will draw a filled circle

closeSerial ()

cls ()
: clear screen to black

clsColor (*color*)
: clear screen to color

color (*color*)
: set color to use for next commands see ezLCD3xx manual for colors

colorId (*ID*, *R*, *G*, *B*)
: set colorid to rgb vaules

getWstate (*ID*)
: return state of widget button presses ect.

getWvalue (*ID*)

getXmax ()
: return the width of display in pixels

getYmax ()
: return the height of the display in pixels

lightBrightness (*brightness*)
: Sets backlight brightness 0-100

lineType (*type*)
: Options: 0 = solid, 1= dotted (1 pixel spacing between dots), 2 = dashed (2 pixel spacing between dashes)

lineWidth (*width*)
: The LINEWIDTH command allows you to draw either a thin line (width = 1) or a thick line (width =3). Only [width] = 1 or 3 are available.

openSerial ()

ping ()
: sends a status check -> ezLCD responds with pong

plot ()
: plot a point at current position in current color

plotXY (*x*, *y*)
: plot pixel at x, y in current color

printChar (*character*)

printString (*string*)
: print string at current coordinates in current color and font

printStringXY (*x*, *y*, *string*)
: print string at x y coordinates in current color and font

setWvalue (*ID*, *value*)

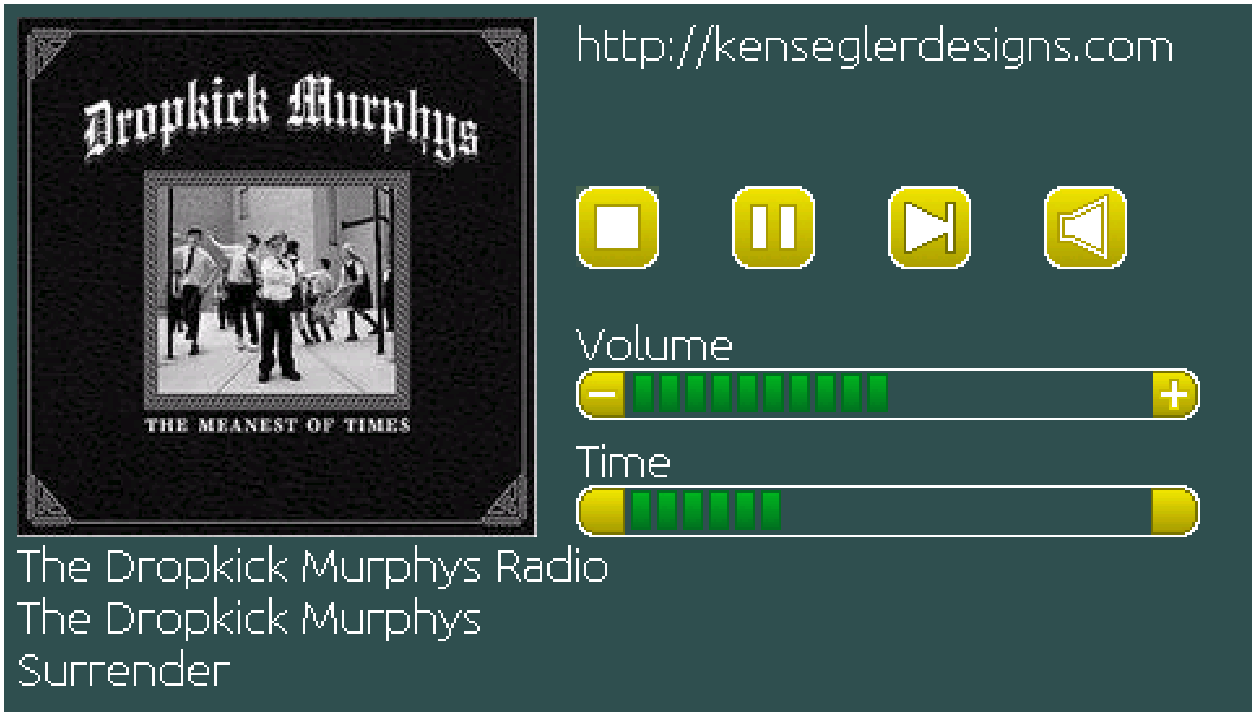
touchZone (*ID*, *x*, *y*, *width*, *heigth*, *options*)
: best widget here

xy (*x*, *y*)
: setXY(x,y) sets coordinates to x, y for commands like print and picture ect.

CHAPTER

TWO

SAMPLE PANDORA PLAYER



The Dropkick Murphys Radio
The Dropkick Murphys
Surrender

```
#!/usr/bin/env python

# Python Serial library for ezLCD3xx
# http://www.ezlcd.com/
#
# You need the pySerial Library by Chris Liechti
# http://pyserial.wiki.sourceforge.net/pySerial
#

BLACK = 0
GRAY  = 1
SILVER= 2
WHITE = 3
RED = 4
MAROON = 5
YELLOW = 6
OLIVE = 7
LIME = 8
GREEN = 9
```

```
AQUA = 10
TEAL = 11
BLUE = 12
NAVY = 13
FUCHISA = 14
PURPLE = 15
ON = 1
OFF = 0


def showStations(temp):
    LCD.clsColor(155)
    sl=open(’/tmp/station.lst’)
    lines = 0
    LCD.PictureXY(220, 230, ’up.gif’)
    LCD.PictureXY(280, 230, ’down.gif’)
    LCD.PictureXY(340, 230, ’play.gif’)
    LCD.Font(’0’)
    LCD.color(’white’)
    LCD.printString(’ ---------- Station List ----------’)
    x=15
    y=20
    LCD.xy(x, y)
    with  open(’/tmp/station.lst’) as sf:
        temp = sf.read().splitlines()
    while True:
        if lines < 12:
            LCD.printString(temp[lines])
            y += 16
            LCD.xy(x, y)
            lines +=1
    sl.close()

def updateVolume( level):
    LCD.PictureXY(240, 141,’vol’+str(level)+’.gif’)

# END SerLCD Class Definition ----------------------------------------

# Start Test Program ------------------------------------------------
import commands
import os
import re
import time as zeit
import sys
sys.path.append("/home/codeman/workspace/ezLCD3xx")
from ezLCD3xx import *

ips = commands.getoutput("/sbin/ifconfig | grep -i \"inet\" | grep -iv \"inet6\" | " +
                         "awk {’print $2’} | sed -ne ’s/addr\:/ /p’")
fifoName = ’/tmp/ezLCD’
pbFifo = ’/tmp/pianobar’
LCD = ezLCD(’/dev/ttyACM0’)
LCD.openSerial()
LCD.Verbose(’off’);
LCD.lightBrightness(100)
LCD.clsColor(155)
LCD.Font(’0’)
LCD.touchZone(1, 217, 67, 38, 38, 1) #stop
LCD.touchZone(2, 277, 67, 38, 38, 1) #play/pause
LCD.touchZone(3, 337, 67, 38, 38, 1) #next
LCD.touchZone(4, 214, 136, 32, 32, 1) #vol -
LCD.touchZone(5, 435, 136, 32, 32, 1) #vol +
LCD.PictureXY(220, 70, ’stop.gif’)
LCD.PictureXY(280, 70, ’pause.gif’)
LCD.PictureXY(340, 70, ’next.gif’)
LCD.PictureXY(400, 70, ’son.gif’)
LCD.color(’white’)
LCD.xy(220, 120)
LCD.printString(’Volume’)
LCD.PictureXY(220, 140, ’volume.gif’)
LCD.color(’white’)
LCD.xy(220, 165)
LCD.printString(’Time’)
LCD.printStringXY(220, 5, ’http://kenseglerdesigns.com’)
LCD.PictureXY(220, 185, ’progfrm.gif’)
time = commands.getoutput("/bin/date")
playPause = 1
LCD.Font(’space’)
x=250
pbf = open(pbFifo,’w’)
volume = 50
updateVolume(volume)
while 1:
        stop = LCD.getWstate(1)
        play = LCD.getWstate(2)
        next = LCD.getWstate(3)
        volDn = LCD.getWstate(4)
        volUp = LCD.getWstate(5)
        if     stop == ’0004\r’:
            showStations(1)
        if     play == ’0004\r’:
            if playPause:
                LCD.PictureXY(280, 70, ’play.gif’)
                playPause = 0
            else:
                LCD.PictureXY(280, 70, ’pause.gif’)
                playPause =1
            pbf.write(’p’)
            pbf.flush()
        if     next == ’0004\r’:
            pbf.write(’n’)
            pbf.flush()
        if     volUp == ’0004\r’:
            volume +=1
            if volume > 100:
                volume =100
            updateVolume(volume)
            pbf.write(’)’)
            pbf.flush()
        if     volDn == ’0004\r’:
            volume -=1
            if volume < 0:
                volume = 0
            updateVolume(volume)
            pbf.write(’(’)
            pbf.flush()
        if  os.path.isfile(fifoName):
            zeit.sleep(0.50)
            fifo = open(fifoName, ’r’)
            station = fifo.readline()
            artist = fifo.readline()
            song = fifo.readline()
            fifo.close()
            os.remove(fifoName)
            LCD.color(’155’)
            LCD.xy(0, 205)
            LCD.Box(480, 62, 1)
            LCD.color(’white’)
            LCD.Font(’0’)
            LCD.printStringXY(5, 205,  station )
            LCD.printStringXY(5, 225, artist )
            LCD.printStringXY(5, 245, song )
            LCD.PictureXY(220, 70, ’stop.gif’)
            LCD.PictureXY(280, 70, ’pause.gif’)
            LCD.PictureXY(340, 70, ’next.gif’)
            LCD.color(’white’)
            LCD.xy(220, 120)
            LCD.printString(’Volume’)
            LCD.PictureXY(220, 140, ’volume.gif’)
            LCD.color(’white’)
            LCD.xy(220, 165)
            LCD.printString(’Time’)
            updateVolume(volume)
            x=220
            LCD.PictureXY(x, 185, ’progfrm.gif’)
            x +=21
            LCD.PictureXY(x, 187, ’prog.gif’)
            x +=10
            LCD.PictureXY(x, 187, ’prog.gif’)
            x +=10
            LCD.PictureXY(x, 187, ’prog.gif’)
            x +=10
            LCD.PictureXY(x, 187, ’prog.gif’)
            x +=10
            LCD.PictureXY(x, 187, ’prog.gif’)
            x +=10
            LCD.PictureXY(x, 187, ’prog.gif’)
            LCD.PictureXY(5, 5, ’temp.jpg’)
        else:
            station =’ ’
            artist = ’ ’
            song = ’ ’

LCD.closeSerial()
# End Test Program --------------------------------------
```

CHAPTER

THREE

PIANOBAR EVENT_COMMAND

```
#!/usr/bin/env python

import os
import re
import time as zeit
import sys
import urllib2
import tempfile
import shutil
import PIL
from PIL import Image

fifoName = ’/tmp/ezLCD’
pandora_icon = os.path.join(os.path.abspath(os.path.dirname(__file__)),
                            "pandora-icon-cc-by-sa-rossr.png")

command = sys.argv[1]

params = {}

for s in sys.stdin.readlines():
        param, value = s.split("=", 1)
        params[param.strip()] = value.strip()


info = {}
info["song"] = params["title"]
info["artist"] = params["artist"]
info["album"] = params["album"]
info["stationCount"] = params["stationCount"]
info["stationName"] = params["stationName"]
info["coverArt"] = params["coverArt"]
stations = []
growl_image = pandora_icon
for i in range(0, int(params["stationCount"])):
        stations.append(params["station"+str(i)])
        info["stations"] = stations
print (’Command ->’)
print (command)
if command == "usergetstations":
        print (info["stations"])
        f = open(’/tmp/station.lst’,’w’)
        f.write("\n".join(info[’stations’]))
        f.close()
if command == "songstart":

        cover_art_url = info["coverArt"].strip()
    #if len(cover_art_url) > 0:
        (imgfp, growl_image) = tempfile.mkstemp(prefix="pianobar",suffix=".jpg")

        try:
            urlfp = urllib2.urlopen(cover_art_url)
            os.write(imgfp, urlfp.read())
        except ValueError, e:
            os.unlink(growl_image)
            growl_image = pandora_icon
        except urllib2.URLError, e:
            os.unlink(growl_image)
            growl_image = pandora_icon
        finally:
            os.close(imgfp)
            basewidth = 200
            img = Image.open(growl_image)
            wpercent = (basewidth/float(img.size[0]))
            hsize = int((float(img.size[1])*float(wpercent)))
            img = img.resize((basewidth,hsize), PIL.Image.ANTIALIAS)
            img.save(growl_image)

            shutil.copy(growl_image, "/media/codeman/EZLCD-304/EZUSER/IMAGES/temp.jpg")
            fifo = open(fifoName, 'w')
            fifo.write(info["stationName"] + '\n' + info["artist"] + '\n' + info["song"])
            fifo.close()


#/media/codeman/EZLCD-304/EZUSER/IMAGES/
```

CHAPTER

FOUR

PIANOBAR CONFIG FILE

```
# This is an example configuration file for pianobar. You may remove the # from
# lines you need and copy/move this file to ~/.config/pianobar/config
# See manpage for a description of the config keys
#
# User
user = email@email.com
password = password

# Proxy (for those who are not living in the USA)
#control_proxy = http://127.0.0.1:9090/
tls_fingerprint = 2D0AFDAFA16F4B5C0A43F3CB1D4752F9535507C0
# Keybindings
act_help = ?
act_songlove = +
act_songban = -
act_stationaddmusic = a
act_stationcreate = c
act_stationdelete = d
act_songexplain = e
act_stationaddbygenre = g
act_songinfo = i
act_addshared = j
act_songmove = m
act_songnext = n
act_songpause = p
act_quit = q
act_stationrename = r
act_stationchange = s
act_songtired = t
act_upcoming = u
act_stationselectquickmix = x
act_voldown = (
act_volup = )

# Misc
# mp3, mp3-hifi or aacplus
#audio_format = mp3-hifi
#autostart_station = 123456
event_command = /home/codeman/Python/ezLCD3xx/pianobar.py
fifo = /tmp/pianobar
#sort = quickmix_10_name_az
#love_icon = [+]
#ban_icon = [-]
```

```
#volume = 0

# Format strings
format_nowplaying_song = "%t" by "%a" on "%l"%r%@%s
format_nowplaying_station = Station "%n" (%i)
format_list_song = %i) %a - %t%r
```

CHAPTER

FIVE

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*

PYTHON MODULE INDEX

INDEX