

Level Meter Module

REV A

Publication Date: 2013/9/19
XMOS © 2013, All Rights Reserved.



Table of Contents

1	Overview	3
1.1	Features	3
1.2	Memory requirements	3
1.3	Resource requirements	3
2	Hardware Requirements	4
3	API	5
4	API	6
5	Programming Guide	7
5.1	Includes	7
5.2	Programming	7
6	Example Applications	8
6.1	app_display_spectrum	8
6.1.1	Getting Started	8
6.2	app_display_spectrum_from_adc	8

1 Overview

IN THIS CHAPTER

- ▶ Features
 - ▶ Memory requirements
 - ▶ Resource requirements
-

The level meter module is used to create a level meter display of a data array on LCD. The rendered image is stored in SDRAM.

1.1 Features

- ▶ Non-blocking SDRAM management.
- ▶ Real time rendering.
- ▶ Color selection for the display.
- ▶ No real time constraints on the application.

1.2 Memory requirements

Resource	Usage
Stack	bytes
Program	bytes

1.3 Resource requirements

Resource	Usage
Channels	
Timers	
Clocks	
Threads	

2 Hardware Requirements

This module may be evaluated using the SliceKit Modular Development Platform, available from digikey. Required board SKUs are:

- ▶ XP-SKC-L2 (SliceKit U16 Core Board) plus XTAG2
- ▶ XA-SK-SCR480 (which includes a 480x272 color touch screen)
- ▶ XA-SK-SDRAM

To build a project including the `module_display_controller` the following modules are required:

- ▶ `module_display_controller`
- ▶ `module_sdram` in `sc_sdram_burst` which handles the SDRAM
- ▶ `module_lcd` in `sc_lcd` which handles the LCD

The section below details the configuration defines and the APIs used in the application.

3 API

The color palette to be used for the level meter display can be configured via the header `level_meter_conf.h`. The defines are:

LEVEL_METER_NCOLORS

This defines the number of color used for the level meter display.

level_meter_colors

This array gives the colors used. The colors can be picked from the list given in `level_meter.h`.

4 API

- ▶ `level_meter.xc`
- ▶ `level_meter.h`

The `level_meter` API is: .. doxygenfunction:: level_meter

5 Programming Guide

IN THIS CHAPTER

- Includes
 - Programming
-

This section provides information on how to create an application using level_meter API.

5.1 Includes

The application needs to include level_meter.h and the configuration file level_meter_conf.h. The color palette for the level meter display is defined in the configuration file.

5.2 Programming

The level meter module uses the APIs of display controller module. A simple application function that uses level_meter API is given below.

```
void app(c_dc, data, N)
{
    unsigned frBuf;

    // Create frame buffer
    frBuf = display_controller_register_image(c_dc, LCD_ROW_WORDS, LCD_HEIGHT
        ↵ );

    // Render level meter display frame and commit
    level_meter(c_dc, frBuf, data, N);
    display_controller_frame_buffer_commit(c_dc, frBuf);
}
```

c_dc is the channel connecting display controller. data is the array of unsigned data values to be displayed. N is the number of data values.

6 Example Applications

IN THIS CHAPTER

- ▶ `app_display_spectrum`
 - ▶ `app_display_spectrum_from_adc`
-

This tutorial describes the demo applications that uses the level meter module. ??? describes the required hardware setup to run the demos.

6.1 `app_display_spectrum`

This application uses display controller and other modules to create a level-meter kind of spectral display on an LCD for a simulated signal. This application demonstrates real-time rendering and display of spectrum by taking short-time fourier transform.

6.1.1 Getting Started

1. Connect XA-SK-SDRAM Slice Card to the XP-SKC-U16 Slicekit Core board using the connector marked with SQUARE.
2. Connect XA-SK-SCR480 Slice Card with LCD to the XP-SKC-U16 Slicekit Core board using the connector marked with DIAMOND.
3. Select `app_display_spectrum`. Build the project and run.

The spectra of segments of mixed signal of two simulated chirp waveforms are displayed on LCD.

6.2 `app_display_spectrum_from_adc`

This application uses `module_usb_tile_support` along with display controller and other modules to create a level-meter kind of spectral display on an LCD for an analog audio input. This application showcases the use of multichannel ADC in an xCORE-USB series XMOS device to sample the analog input and real-time rendering and display of spectrum by taking short-time fourier transform.

1. Connect XA-SK-SDRAM Slice Card to the XP-SKC-U16 Slicekit Core board using the connector marked with SQUARE.
2. Connect XA-SK-SCR480 Slice Card with LCD to the XP-SKC-U16 Slicekit Core board using the connector marked with DIAMOND.
3. Connect XA-SK-MIXED SIGNAL Slice Card to the XP-SKC-U16 Slicekit Core board using the connector marked with A.

4. Give the two channels of audio input from a PC or a mobile to pins 1 and 2 of J2 on the mixed signal slice card using a suitable cable. The ground is connected to pin 4 of J3.
5. Select `app_display_spectrum_from_adc`. Build the project and run.
6. Play an audio in the PC or mobile.

The spectra of segments of mixed signal of two audio channels are displayed on LCD.



Copyright © 2013, All Rights Reserved.

Xmos Ltd. is the owner or licensee of this design, code, or Information (collectively, the "Information") and is providing it to you "AS IS" with no warranty of any kind, express or implied and shall have no liability in relation to its use. Xmos Ltd. makes no representation that the Information, or any particular implementation thereof, is or will be free from any claims of infringement and again, shall have no liability in relation to any such claims.

XMOS and the XMOS logo are registered trademarks of Xmos Ltd. in the United Kingdom and other countries, and may not be used without written permission. All other trademarks are property of their respective owners. Where those designations appear in this book, and XMOS was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.