AN10815

SWIM: NXP's basic graphics library for LPC products

Rev. 3 — 15 June 2011 **Application note**

ocument information

Info	Content
Keywords	SWIM, Graphics Library for LPC178x, LPC18xx, LPC24xx and LPC32x0, LCD, TFT, STN, IRD, Phytec, Embedded Artists, Keil MDK, IAR EWARM, Rowley Crossworks.
Abstract	This document describes example projects created to demonstrate the LPC SWIM Graphics library. Toolchains used are Keil MDK, IAR EWARM and Rowley Crossworks. The development platforms were from Embedded Artists, NXP, and Phytec.

Revision history

Rev	Date	Description
3	20110615	Added support for EA LPC1788 OEM board.
2	20110501	Added support for Hitex LPC1850.
1	20090501	Initial revision.

Contact information

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

1. Introduction

Simple Window Interface Manager (SWIM) is a basic graphics library developed for the NXP LPC products. It can be also be used with LPC controllers that do not have a dedicated LCD interface. The SWIM graphics library allows developers to quickly and easily implement a system with basic graphics support.

Project examples for IAR EWARM, KEIL MDK, and Rowley Crossworks toolchains are provided as part of the package. These projects are directed at the following target platforms:

- LPC3250 from Phytec: http://www.phytec.com
- LPC2478 from EA: http://www.embeddedartists.com
- IRD Platform from NXP: http://ics.nxp.com/support/boards/ird/
- LPC1850 from Hitex: http://www.hitex.com/index.php?id=3212
- LPC1788 from EA http://www.embeddedartists.com

The example projects demonstrate how to use the library and will help users get familiar with the library calls quickly and efficiently.

2. SWIM features

The following section describes the main features and functionality of the SWIM library. In addition, the library code is well commented and an associated API document for the software calls - SWIM v1.0.pdf – is also available. Some of the main SWIM functions are briefly summarized in the following subsections.

2.1 Graphic primitives

- swim_put_pixel – places a pixel of specified color at a specified location on the LCD
- swim_put_line – draws a colored line for positions x to y
- swim_put_diamond – draws a diamond shape of specified color and position
- swim_put_box – put box at specified location. Pen color for edges and fill color for center
- swim_set_pen_color – sets the pen color
- swim_clear_screen – fills the draw area of the display with the selected color

2.2 Image support

- swim_put_image – puts a raw image into a window
- swim_put_scale_image – puts and scales a raw image into a window
- swim_put_invert_image – puts a raw image into a window inverted

2.3 Font support

- Helvetica 10-point proportional font
- 8x16 proportional font
- 8x8 proportional font
- Fixed 5x7 proportional font
- Windows FreeSystem 14x16 Font

3. IRD platform

The Industrial Reference Design (IRD) v2.0 is a platform targeted at RTOS based embedded systems. Designed around a flexible Core and Base printed circuit board (PCB) concept, it features many of the system functions and wired communications protocols found in today's embedded applications. The examples provided with this SWIM library are developed for use with the LPC2478 core board module.

Fig 1. IRD platform

3.1 Description of IRD board setup

The IRD 2.0 platform should come preassembled with the LPC2478 core board installed and the Toshiba LCD module attached. If not, refer to the IRD User's Manual to assemble them properly. There are no jumpers to configure that affect the example code for this platform.

The IRD 2.0 2478 board has either 8 MB, 16 MB, or 32 MB external SDRAM using a 64 Mbit, 128 Mbit, or 256 Mbit x 32 SDRAM device in U6 respectively:

- MT48LC2M32B2 (64 Mbit SDRAM, 32-bit databus) from Micron (default)
- MT48LC4M32B2 (128 Mbit SDRAM, 32-bit databus) from Micron (option)
- MT48LC8M32B2 (256 Mbit SDRAM, 32-bit databus) from Micron (option)

The supplied example is setup for the 64 Mbit (8 MB) Micron SDRAM configuration.

For other versions of the core board it will be necessary to modify the SDRAM initialization code in the "ex_sdram.c" file. This SDRAM memory will be used as the LCD frame buffer memory for this example. The SDRAM memory resides at address:

0xA000 0000 - 0xA07F FFFFF using DYNCS0 (8 MB, 2Mx32)

Using the 240x320 16bpp RGB1:5:5:5 mode, the frame buffer uses 150 kB of SDRAM starting at address 0xA0000000.

The parameters for the Toshiba LCD panel are configured in lcd_params.c; this panel is configured for operation in 1:5:5:5 mode.

3.2 Rowley project description

3.2.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

3.2.2 Required hardware

IRD 2.0 2478 Evaluation board w/Toshiba LTA057A347F 5.7" 320x240 LCD module.

3.2.3 Required software

Rowley CrossStudio for ARM v1.5 or newer.

Rowley CrossConnect for ARM or other supported debugger.

3.2.4 Usage

1. Start Rowley CrossStudio for ARM and open the example solution file.

   Select File->Open Solution->

   Open the following solution: LPC2478_SWIM_Example.hzp
2. Build the solution.

   Build->Build Solution
3. Attach your CrossConnect debugger to the IRD 2.0 board and PC, then connect to it.

   Targets->Connect USB CrossConnect for ARM
4. Download the program into flash.

   Debug->Start Debugging (F5)
5. Run the program!

   Debug->Go (F5)

3.3 IAR EWARM project description

3.3.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

3.3.2 Required hardware

IRD 2.0 2478 Evaluation board w/Toshiba LTA057A347F 5.7" 320x240 LCD module

3.3.3 Required software

IAR Embedded Workbench for ARM (EWARM) v5.x or newer

3.3.4 Usage

1. Start IAR Embedded Workbench for ARM and open the example workspace space.

   File->Open->Workspace...

   Open the following workspace: 2478_swim_example.eww
2. Build the project.

   Project->Rebuild All
3. Attach your JLink JTAG unit to the EA-2478 board and PC.
4. Download the program into flash.

   Project->Download and Debug (Ctrl+D)
5. Run the program!

   Debug->Go (F5)

3.4 Keil MDK project description

3.4.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

3.4.2 Required hardware

IRD 2.0 2478 Evaluation board w/Toshiba LTA057A347F 5.7" 320x240 LCD module.

3.4.3 Required software

Keil uVision v3.x or newer. Code should compile on evaluation version.

3.4.4 Usage

1. Start Keil uVision3 for ARM and open the example project file.

   Project->Open Project...

   Open the following project: ea_lcd.Uv2
2. Build the project.

   Project->Build Target
3. Attach your ULink2 JTAG unit to the IRD 2.0 board and PC.
4. Download the program into flash.

   Debug->Start/Stop Debug Session (Ctrl+F5)
5. Run the program!

   Debug->Run (F5)

4. LPC2478 EA board

Embedded Artists' LPC2478 OEM Board (mounted on the QVGA OEM Base Board with touch panel) lets you get up-and-running quickly with NXP's ARM7TDMI LPC24xx microcontroller series.

The OEM board has SODIMM format and is only 66x48 mm. All processor signals are available on the 200-pin connector for easy expansion. The board can be used in OEM applications, as well as for educational purposes, experiments, and prototype projects.

Fig 2. EA LPC2478 platform

4.1 Description of EA LPC2478 board setup

The EA-2478 board should come preassembled with the LPC2478 core board installed and the Truly LCD module attached. If not, refer to the EA-2478 User's Manual to assemble them properly.

The jumpers can be left in the default position when shipped from the factory. The jumpers on this board are not labeled with the standard "J1" labels but only descriptive labels. The ones that matter for the example projects are:

- Int/Ext display – set to internal display
- Enable LCD – installed
- 16 bit / 24 bit RGB data – set to 16 bit
- Backlight shutdown – removed
- Enable JTAG – installed

The EA-2478 board has either 32 MB or 16 MB of external SDRAM using a x16 or a x32 SDRAM device in U9 or U13 respectively:

- K4S561632H-UC75 (256 Mbit SDRAM, 16-bit databus) from Samsung (LPC2478-16 OEM Board)
- K4M563233G-HN75 (256 Mbit Mobile SDRAM, 32-bit databus) from Samsung (v1.0 of LPC2478-32 OEM Board)
- MT48LC8M32B2B5-7 (256 Mbit SDRAM, 32-bit databus) from Micron (v1.1a of LPC2478-32 OEM Board)

This example is setup for the x32 Micron SDRAM configuration, v1.1a of the LPC2478-32 core board. For other versions of the core board it may be necessary to modify the SDRAM initialization code in the "ex_sdram.c" file. The SDRAM memory will be used as the LCD frame buffer memory for this example. The SDRAM memory resides at address:

- 0xA0000000 - 0xA1FFFFFF using DYNCS0 (32 MB, 8Mx32)

In 240x320 16bpp RGB1:5:5:5 mode, the frame buffer uses 150 kB starting at:

- 0xA0000000 within the SDRAM.

The Truly LCD module on the EA-2478 board contains a complete LCD controller module with its own frame buffer as well as a hardware touchscreen controller. The example presented disables the LCD controller and frame buffer on the Truly module and places the LCD module into a dumb RGB1:5:5:5 mode. The SPI interface to the module is used to program the module into this mode. The parameters for the Truly LCD panel are configured in lcd_params.c.

4.2 Rowley project description

4.2.1 Description

Draws color bars on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

4.2.2 Required hardware

EA-2478 Evaluation board w/Truly LCD module in RGB1:5:5:5 mode

Rowley CrossConnect for ARM or other supported debugger

4.2.3 Required software

Rowley Crossworks for ARM v1.5 or newer

4.2.4 Usage

1. Start Rowley CrossStudio for ARM and open the example solution file.

   Select File->Open Solution

   Open the following solution: EAC2478_SWIM_Example.hzp
2. Build the solution.

   Build->Build Solution
3. Attach your CrossConnect debugger to the EA-2478 board and PC, then connect to it.

   Targets->Connect USB CrossConnect for ARM
4. Download the program into flash.

   Debug->Start Debugging (F5)
5. Run the program!

   Debug->Go (F5)

4.3 IAR EWARM project description

4.3.1 Description

Draws color bars on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

4.3.2 Required hardware

EA-2478 Evaluation board w/Truly LCD module

4.3.3 Required software

IAR Embedded Workbench for ARM (EWARM) v5.x or newer

4.3.4 Usage

1. Start IAR Embedded Workbench for ARM and open the example workspace space.

   File->Open->Workspace...

   Open the following workspace: ea2478_swim_example.eww
2. Build the project

   Project->Rebuild All
3. Attach JLink JTAG unit to the EA-2478 board and PC
4. Download the program into flash

   Project->Download and Debug (Ctrl+D)
5. Run the program!

   Debug->Go (F5)

4.4 Keil MDK project description

4.4.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

4.4.2 Required hardware

EA-2478 Evaluation board w/Truly LCD module.

4.4.3 Required software

Keil uVision v3.x or newer. Tested with Evaluation version of MDK.

4.4.4 Usage

1. Start Keil uVision3 for ARM and open the example project file.

   Project->Open Project...

   Open the following project: ea_lcd.Uv2
2. Build the project.

   Project->Build Target
3. Attach ULink2 JTAG unit to the EA-2478 board and PC.
4. Download the program into flash.

   Debug->Start/Stop Debug Session (Ctrl+F5)
5. Run the program!

   Debug->Run (F5)

5. LPC3250 Phytec

The LPC3250 Phytec system is made up from three boards: the phyCORE-ARM9/LPC3250 System on Module (PCM-040), the phyCORE-ARM9/LPC3250 Carrier Board (PCM-967), and a optional add-on LCD - Hitachi 3.5" QVGA TFT-LCD with integrated touch on adapter board (KLCD-011). The complete system is shown in Fig 3.

Fig 3. Phytec LPC3250 Platform

The phyCORE-LPC3250 module is populated with the NXP LPC3250. State-of-the-art power management, Vector Floating Point Unit (VFP), and rich peripherals such as USB OTG, Ethernet, and integrated LCD controller make this device the ideal candidate for embedded applications requiring high performance and low power consumption. The on-board MMU supports major operating systems, including Linux and Windows Embedded CE. Other chip-level features include 7 UARTs, SPI, I2C, a real-time clock with a separate power domain, and NAND Flash and DDR memory controllers. These features make the devices particularly suitable for automotive and industrial control applications as well as medical systems.

5.1 Description of Phytec board setup

The LPC3250 Phytec is shipped with a preinstalled bootloader called the stage 1 loader (S1L). It is assumed that this is present on the board. The S1L first initializes the board with the code from the phy3250_startup_entry.s and phy3250_startup.c files before starting the monitor program. Without this initialization the colorbar example will not run.

Be sure to configure your system defines in phy3250_board.h to match your hardware revisions. Failure to properly set these defines to the correct value may prevent the code from working correctly.

The possible define values are below:

- PHY3250_CARRIERBOARD_1305_X
- PHY3250_MODULEBOARD_1304_X
- PHY3250_LCD_1307_X

LCD modules have a board number of 1307.x, where x = 0 or 1. The change between .0 and .1 LCD module is detailed below:

LCD module revision differences:

.0 initial board

.1 Active HIGH backlight signal, different wiring

Because of differences in the connector wiring of ‘.0’ and ’.1’ LCD modules, the ‘.0’ LCD modules should only be used with ‘.0’ carrier boards, while ‘.1’ LCD modules should only be used with ’.2’ or greater carrier boards.

Carrier boards have a revision number of 1305.x, where x = 0 to 3. The changes between the boards module are shown below:

Carrier board revision differences:

.0/.1 Initial board

.2/.3 USB peripheral VBUS routed to USB_VBUS signal

.2/.3 GPI4 used to VBUS detection for USB peripheral

.2/.3 USB_ADR/SW used to control USB_ host power

Module boards have a revision number of 1304.x, where x = 0 to 1. The changes between the module boards module are shown below:

Module board revision differences:

.0 Initial board

.1 USB ISP1301 I2C address changed from 0x2C to 0x2D

For correct jumper settings please refer to the PHYTEC phyCORE-LPC3250 System on Module and Carrier Board Hardware Manual.

The phyCORE-LPC3250 board comes preconfigured with 64 MB of 133 MHz SDR SDRAM configured for 32-bit access using two 16-bit wide RAM chips at U10 and U11.

The LPC3250 is capable of addressing a single RAM bank located at memory address 0x8000 0000 and extending to 0x9FFF FFFF via the /DYCS0 signal.

The LCD is a Hitachi TX09D71VM1CCA. In 240x320 16bpp RGB565 mode the frame buffer uses 150kB starting at: 0x8000 0000 within the SDRAM. The parameters for the Hitachi LCD panel are configured in lpc_lcd_params.c

5.2 Keil MDK project description

5.2.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:6:5 mode.

5.2.2 Required hardware

LPC3250 Phytec board with LCD.

5.2.3 Required software

Keil uVision v3.x or newer. Tested with Evaluation version of MDK.

5.2.4 Usage

The code is setup to execute from IRAM (Debug version) of the LPC3250.

1. Start Keil uVision3 for ARM and open the example project file

   Project->Open Project...
2. Build the project

   Project->Build Target
3. Attach ULink2 JTAG unit to the Phytec board and PC
4. Download the program

   Debug->Start/Stop Debug Session (Ctrl+F5)
5. Run the program!

   Debug->Run (F5)

For instructions on how to load the code into NAND Flash (Release version) please refer to the documents in the LPC3250 Common Driver Library package which can be downloaded at www.nxp.com/microcontrollers.

Keil also provides a NAND Flash bootloader which can be used to download the code in to NAND flash. Refer to the Keil documentation/examples on how to save and execute the code from NAND flash.

5.3 IAR EWARM project description

5.3.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:6:5 mode.

5.3.2 Required hardware

LPC3250 Phytec board with LCD.

5.3.3 Required software

The project was developed using IAR Embedded Workbench ARM v5.x or newer. The IAR Evaluation version will also work.

5.3.4 Usage

The code is setup to execute from IRAM (Debug version) of the LPC3250.

1. Start EWARM and open the example project file.

   Project->Open Project.

2. Build the project.

   Project->Build Target

3. Attach JLINK JTAG unit to the Phytec board and PC.
4. Download the program.

   Debug->Start/Stop Debug Session (Ctrl+F5)

5. Run the program!

   Debug->Run (F5)

For instructions on how to load the code into NAND Flash (Release version) please refer to the documents in the LPC3250 Common Driver Library package which can be downloaded at www.nxp.com/microcontrollers.

IAR also provides a NAND Flash bootloader which can be used to download the code in to NAND flash. Refer to the IAR documentation/examples on how to save and execute the code from NAND flash.

5.4 Rowley project description

Currently Rowley Crosswork does not support the NXP LPC3250 microcontroller.

6. LPC1850 Hitex & LogicPD LCD panel

The LPC1850 evaluation board is USB-powered, but can also be driven by external power supply or via power-over-Ethernet. It is equipped with 64 Mbit SDRAM, 32 MB parallel flash, 512 kB SRAM and a serial EEPROM. For debugging a JTAG as well as a 20-pin Cortex debug connector with ETM is available. All channels (USB1 and USB2) as well as Ethernet are provided.

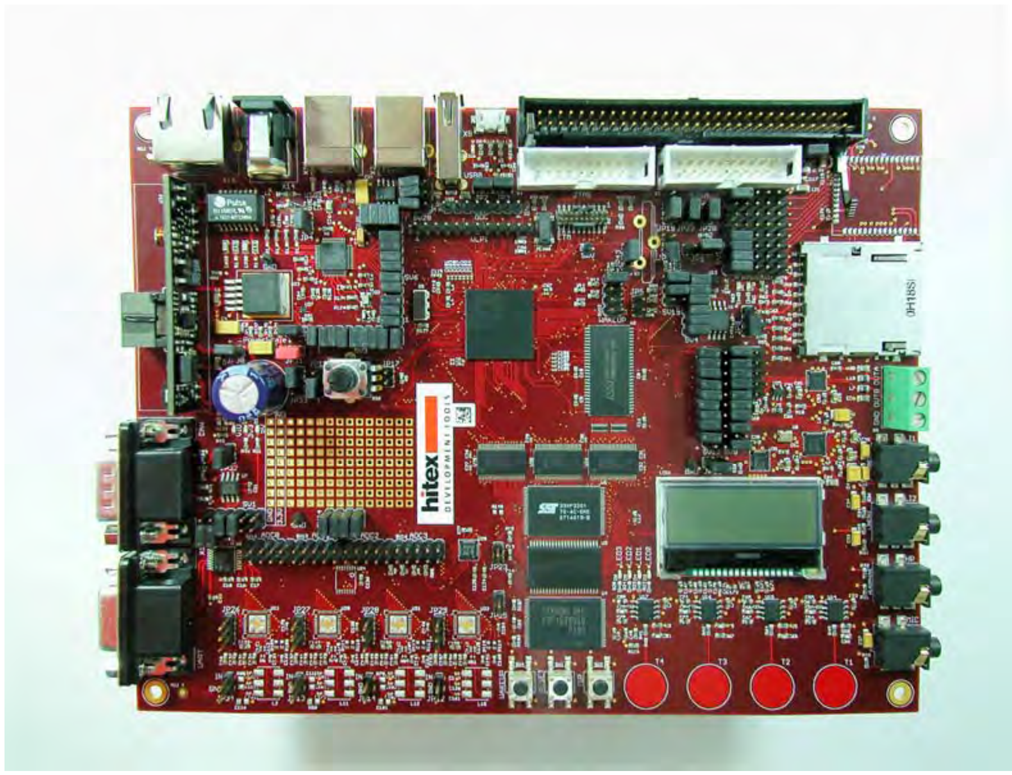

Fig 4. Hitex LPC1850 platform

6.1 Description of Hitex LPC1850 board and LogicPD panel setup

The supplied example software is designed to run on a Hitex LPC1850 evaluation board that is connected to a LogicPD LCD panel (model number LCD-6.4-VGA-10R-A). This module contains a VGA (640 x 480) TFT Sharp LCD (LQ64D343).

The LCD ribbon cable should be attached to connector labeled “LCD” on the LPC1850 board. The jumpers can be left in the default position when shipped from the factory.

The Hitex LPC1850 board has 8 MB of external SDRAM using a 1 Mbit x 16bit x 4 bank SDRAM device. This SDRAM memory will be used as the LCD frame buffer memory for this example. The SDRAM memory resides at address:

0x2800 0000 - 0x287F FFFF using DYNCS0

Using the 640x480 16bpp RGB1:5:5:5 mode, the frame buffer uses 614 kB of SDRAM starting at address 0x28000000.

The parameters for the LogicPD LCD panel are configured in lcd_params.c; this panel is configured for operation in 1:5:5:5 mode.

6.2 Keil MDK project description

6.2.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

6.2.2 Required hardware

LPC1850 Hitex Board and PD Logic LCD module (model number LCD-6.4-VGA-10R-A).

6.2.3 Required software

Keil uVision v4.x or newer. Tested on evaluation version 4.20.

6.2.4 Usage

1. Start Keil uVision4 for ARM and open the example project file.

   Project->Open Project...

   Open the following project: LPC1850_SWIM_Example.uvproj
2. Build the project.

   Project->Build Target
3. Attach ULINK2 or ULINKPro JTAG unit to the LPC1850 board and PC.
4. Download the program.

   Debug->Start/Stop Debug Session (Ctrl+F5)
5. Run the program!

   Debug->Run (F5)

Note: If the example is to be loaded into external flash memory then contents of the tools directory should be copied to the following location:

C:\Keil\ARM\Flash

7. EA LPC1788 Setup

The Embedded Artists LPC1788 OEM Board is a small, reusable DIMM module that mounts to the EA OEM Base Board. The OEM Base Board has many I/O connectors including J26, a 50 pin connector for an external LCD. Several external LCDs can be plugged in. The SWIM library is designed to work with a Truly 3" LCD module (EA QVGA TFT LCD v2.3) and QVGA display adapter. The whole setup can be powered either by a USB connector J32 or a DC barrel connector U18.

7.1 Description of the Embedded Artists board and Truly panel setup

The supplied example software is designed to run on the four-piece setup from EA including the LPC1788 DIMM and Truly QVGA display. Before assembling everything, a few jumpers must be checked on the QVGA LCD Adapter. The **CFG1** through **CFG4** jumpers need to be set up like so:

CFG1 H
CFG2 L
CFG3 L
CFG4 L

Also, there is a jumper labeled **PWM / SHDN**. This should be in the **SHDN** position.

Once the jumpers on the adapter have been set up properly, the whole system can be connected together. The QVGA LCD LCD v2.3 should be connected to the QVGA Display Adapter, which is connected to the EA OEM Base Board using the 50-pin ribbon cable. On the EA OEM Base Board, an LPC1788 OEM Module DIMM should be inserted (red).

The display used in this setup is the Truly G240320LTSW, a portrait-mode QVGA module. Although this is a smart module capable of managing its own raster refreshes, in this application it is reconfigured to display 16-bit RGB data (1:5:5:5 mode) directly from the LPC1788. This configuration happens in lcd_driver.c in lcd_hw_init_Truly() which initializes the SSP interface to communicate with the display and lcd_display_init_Truly() which writes the registers in the LCD controller to configure 1:5:5:5 mode.

The SDRAM is present on the EA OEM Module DIMM and is a single chip ISSI IS42S32800D. This SDRAM implements a 32-bit wide 32 megabyte memory area, of which 150 KB is reserved for the LCD framebuffer.

7.2 Keil MDK project description

7.2.1 Description

Draws color bars and text on the LCD using the SWIM library with LCD in RGB1:5:5:5 mode.

7.2.2 Required hardware

Embedded Artists LPC1788 OEM board
Embedded Artists OEM Base Board rev. A
Embedded Artists QVGA 320x240 LCD module v2.3
Embedded Artists QVGA Display Adapter rev. A (with jumper settings)

7.2.3 Required software

Keil uVision v4.x or newer or IAR 6.0 or newer. Tested on Keil evaluation version 4.20 and IAR v6.20.

7.2.4 Keil Usage

1. Start Keil uVision4 for ARM and open the example project file.

   Project->Open Project...

   Open the following project: EA_LPC1788\swim.uvproj
2. Build the project.

   Project->Build Target
3. Attach a ULINK2 or ULINKPro debugger to the LPC1788 board and PC.
4. Download the program.

   Debug->Start/Stop Debug Session (Ctrl+F5)
5. Run the program!

   Debug->Run (F5)

7.2.5 IAR Usage

1. Start Keil uVision4 for ARM and open the example project file.

   Project->Open Project...

   Open the following project: EA_LPC1788\swim.eww
2. Build the project.

   Project->Make
3. Attach a J-Link debugger to the LPC1788 board and PC.
4. Download the program into flash and run it.

   First select the Flash Debug target in the project window

   Then choose Project->Download and Debug (Ctrl+D)

8. Miscellaneous

8.1 RGB setting

The SWIM RGB format will need to be adapted depending on the format supported by the LCD and the manner in which the LCD is connected to the LPC controller.

For the Phytec board, the LCD uses RGB565 while the Hitex, EA and IRD examples use RGB555.

In the file “lpc_colors.h” set the RGB for the particular format as shown below:

For Phytec LPC3250 Board:

```
#define COLORS_DEF 16     /* 16-bit 565 color mode */
```

For EA LPC2478 and IRD boards:

```
#define COLORS_DEF 15     /* 15-bit 555 color mode */
```

The SWIM library will can also be used to manipulate and display Bitmaps.

8.2 Demo output

The snapshot in Fig 5 shows the resulting LCD panel look and feel when the code is run on the different platforms. The image in this case is taken from the EA LPC2478 Board.

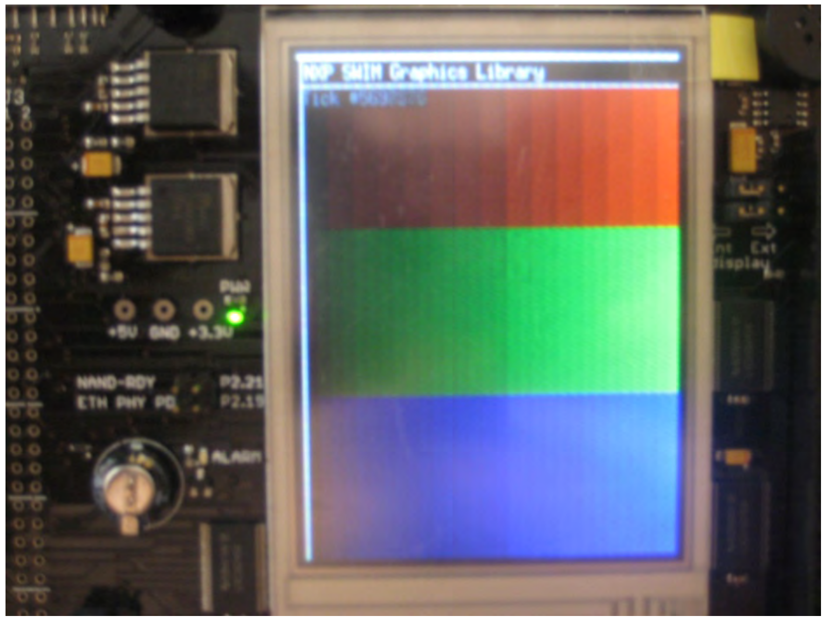

Fig 5. EA LPC2478 platform

9. Legal information

9.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

9.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from national authorities.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

9.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.

10. Contents

Date of release: 15 June 2011
Document identifier: AN10815