Expansion Board for Microchip PIC32 Starter Kits

PIC32-SSD1963 Multimedia Evaluation Kit



Table of Contents

	Revisions	Page 3
Chapter 1.	Introduction1.1Overview1.2Operational Requirements1.2.1Hardware1.2.2Software	4 4 5 5 6
Chapter 2.	Running the first application	8
Chapter 3.	TCP/IP demo program using wireless network	12
Chapter 4.	Interfacing Solomon SSD1963 display controller	20

Revisions

Chapter 1. Introduction

1.1 Overview

The PIC32-SSD1963 Multimedia Evaluation Kit (MMEVK) is a development board designed to work with our SSD1963 Evaluation Kit (SSD1963EVK)ⁱ together with any of the Microchip PIC32 Starter Kits. There are three variants of PIC32 Starter Kits at time of writing, that are the General Purpose Starter Kit (DM320001)ⁱⁱ, USB Starter Kit II (DM320003-2)ⁱⁱⁱ, and the Ethernet Starter Kit (DM320004)^{iv}.

Besides the interface for the SSD1963 Evaluation Kit, peripherals including a MP3 audio codec chip, WiFi module (footprint), and a microSD card socket are onboard for development of content-rich Graphical User Interface (GUI) applications.

Components on MMEVK are labeled in Figure 1-1.

- 1. Hirose FX10A-120S socket for Microchip PIC32 Starter Kits
- 2. VS1003B MP3 audio codec chip manufactured by VLSI Solution^v
- 3. Footprint for Microchip MRF24WB0MA/B WiFi modulesvi
- 4. SPI EEPROM for storage of web pages or non-volatile data such as touch screen calibration constants
- 5. MicroSD socket
- 6. USB to UART bridge CP2102 for program debug
- 7. 2x20 2.54mm receptacle for SSD1963 Evaluation Kit (SSD1963EVK R3B)
- 8. Buzzer with an amplifier circuit
- 9. 2x22 2.00mm expansion port (JP2k) for application modules



Figure 1-1. A board layout of PIC32-SSD1963 Multimedia Evaluation Kit (MMEVK)

1.2 Operational Requirements

The following hardware and software are required to use MMEVK.

1.2.1 Hardware

- Any of the Microchip PIC32 Starter Kits. It can be the PIC32 GP Starter Kit (DM320001), PIC32 USB Starter Kit II (DM320003-2), or PIC32 Ethernet Starter Kit (DM320004).
- A PC compatible system running Microsoft Windows XP SP2 or above
- Two USB ports on the PC, one for USB-UART bridge for program debug, and the other USB port for PIC32 Starter Kit.
- Two USB cables with a mini-USB type B at one end for our EVK and a USB type A at the other for PC.
- A microSD memory card of 2GB or smaller (optional).
- Microchip MRF24WB0MA/B WiFi modules (optional, and soldering is required).
- A wireless LAN router in your workplace (optional)
- SSD1963 Evaluation Kit (SSD1963EVK R3B). This is optional, but highly recommended.

Figure 1-2 illustrates various possible hardware combinations.





A PIC32 USB Starter Kit is stacked on MMEVK. The same socket can be used for PIC32 GP Starter Kit or PIC32 Ethernet Starter Kit. A wireless LAN module MRF24WB0MA has been soldered for internet applications.

A PIC32 GP Starter Kit is stacked on board. Wireless LAN module MRF24WB0MA soldered, with an application module for universal learning IR plugged in the 2x22 2.00mm receptacle (JP2k). The 2.00mm receptacle has been designed to bring out most of the critical I/O pins of the PIC32 MCU including SPI, I2C, PMP, UART, and ADC, to name few of those.



Figure 1-2. Examples of hardware combination

A PIC32 General purpose Starter Kit is stacked on board. This time, SSD1963 EVK is connected for applications requiring a graphical user interface up to 7" TFT with touch panel.

1.2.2 Software

All applications for this evaluation kit have been prepared under the following environment.

1. Microchip MPLAB Integrated Development Environment (IDE) version 8.63.

At time of writing, the latest software version is v8.73a. MPLAB can be downloaded from this hyperlink

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW0 07002

By following the installation procedure for default settings, MPLAB was installed under *C:\Program Files\Microchip\MPLAB IDE*.

2. MPLAB C Complier for PIC32 MCUs version 2.00.

The MPLAB C Compiler for PIC32 (C32) is a full-featured ANSI compliant C compiler for Microchip's PIC32 family of 32-bit microcontrollers. A free evaluation is available by downloading the Evaluation Edition. It has no code size limit and provides complete functionality for 60 days. The compiler is completely usable after 60 days other than certain optimization levels are disabled.

The hyperlink for downloading C32 is http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2615&dDocName=en532454.

By following the installation procedure for default settings, C32 was installed under *C:\Program Files\Microchip\mplabc32\v2.00*. All example programs from us are working on the Evaluation Edition before and after 60 days.

3. The firmware package available at Doc 03. This can be downloaded from the same web page you have got this user guide. The hyperlink of this package is <a href="http://www.techtoys.com.hk/PIC boards/PIC32STK%20SSD1963%20EVK/PIC32STK%20SSD1963%20EVK/PIC32STK%20SSD1963%20EVK/PIC32STK%20SSD1963%20EVK/PIC32STK%20SSD1963%20EVK/PIC32STK%20SSD1963%20EVK/PIC32STK%20SSD1963%20EVK%20R1

The origin of the firmware package is the Microchip Application Libraries version v2011-07-14. The main page of the libraries is found from this web page. http://www.microchip.com/stellent/idcplg?ldcService=SS_GET_PAGE&nodeId=2680&dDocName=en547784

The Microchip Application Libraries includes full source code for the following software libraries: USB, Graphics, Memory Disk Drive, TCP/IP Stack, mTouch Capacitive Touch Library, and Smart Card Library. This is a big library collection of 639MB for not only PIC32 MCUs, but also other MCU series of Microchip. Install as usual by following the default installation path. A new folder under the path C:\Microchip Solutions v2011-07-14 was installed. We have not used all of the projects but only few of those for our hardware.

Individual versions of the Microchip Application Libraries are shown on Figure 1-3 as a reference. We will be using the USB Framework, Graphics Library, Memory Disk Drive, and the TCP/IP Stack.

Microchip Application Librarie	s download i	ncludes the	following:		
Library	Current Version	PIC16F (8-bit)	PIC18F (8-bit)	PIC24/dsPIC (16-bit)	PIC32 (32-bit)
USB Framework	2.9a		x	X	x
Graphics Library	3.01			x	x
Memory Disk Drive (MDD)	1.3.2		x	x	x
TCP/IP Stack	5.36.2		x	x	x
<u>mTouch Capacitive Touch</u> Library	1.31	x	x	x	x
Smart Card Library	1.02.2		x	x	x
<u>MiWi™ Development</u> Environment	4.2.2		x	x	x
Accessory Framework for Android™	1.0.1			x	x

Figure 1-3. Microchip Application Libraries versions

4. Virtual COM Port (VCP) driver for CP2102 USB-to-UART Bridge from Silab's web site at this hyperlink:

http://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx



Figure 1-4. VCP Driver Kit for CP2102

After installing the VCP driver, connect a USB cable to J1U port which is the mini-USB port located close to the UART DEBUG white label on board. No external power is required because the board is drawing power from the USB cable. At this time, LED D1U will glow with a virtual com port emulated. Take a look at *Device Manager* in *Control Panel* to make sure a new COM PORT is emulated successfully under *Ports (COM & LPT)*.

Chapter 2. Running the first application

The first application is a simple demo to blink a LED at RD2 (LED3) with a key press at RD6 (SW1) that is on board of each PIC32 Starter Kit. Debug messages are displayed on two channels: Windows HyperTerminal via CP2102 and the *Output Window* of MPLAB.

The source code of this program is located in the Firmware folder under .. $MCHP_2011_07_14$ Port_IO Demo.

Choose any of the PIC32 Starter Kits for this project. The project names have demystified which PIC32 Starter Kit to use with "GP" standing for general purpose, "ETH" for Ethernet, and "USB" for USB as shown on Figure 2-1.

MPLAB IDE v8.63	
File Edit View Project Debugger Programmer	Tools Configure Window Help
D 📽 🖬 X 🐂 😫 🚳 A 🗯 🚚	III, 2
Checksum: 0xf7d8390e	
🗖 Untitled Workspace 📃 🗖 🔀	Open Project
	Look in: 🔁 Port_IO Demo 🛛 🔽 🕐 🚥
	Dipletts - PIC32 ETH Starter Kit
	Cobjects - PIC32 GP Starter Kit
	port_jo - PIC32 ETH Starter Kit
	PIC32 General Purpose Starter Kit
	File name: port_io - PIC32 ETH Starter Kit PIC32 USB Starter Kit
	Files of type: MPLAB IDE Project Files (".mcp) Cancel
	Jump to: D:\Projects\SSD1963 Eva\MPLAB\GUIv2(🗸

Figure 2-1. Project names of Port_IO demo

Stack the corresponding PIC32 starter kit on MMEVK to finish the hardware. No external 5V is required because the USB power for PIC32 starter kit is enough for this demo. The setup is shown on Figure 2-2 below showing a PIC32 USB SK for this example.



Figure 2-2. Hardware for Port_IO Demo.

Launch Windows HyperTerminal, set the baud rate to 19200bps with no parity with no hardware handshake (8-N-1). Make sure the correct emulated COM PORT has been selected. In my case, it is COM7 emulated from the VCP driver indicated in *Windows Device Manager* as shown on Figure 2-3 below. The procedure for VPC driver installation has been described under section 1.2.2 part 4 of this guide.

🚚 Device Manager	
File Action View Help	
Computer Computer Computer Computer Display adapters Compatible printers Social of COM 11 Display Communications Port (COM1) Display Display adapters Social of COM 2010 Bio Later Diridge (COM7) Display Display Social of COM 2010 Bio Later Diridge (COM7) Display Social of COM 2010 Bio Later Diridge (COM7) Display Social of COM 2010 Bio Later Diridge (COM7) Display Social of COM 2010 Bio Later Diridge (COM7) Display Social of COM 2010 Bio Later Diridge (COM7) Display Display Social of COM 2010 Bio Later Diridge (COM7) Display D	

Figure 2-3. VCP emulated by CP2102 USB-UART bridge

There are two options to compile the program, that are Release and Debug versions. Let's try the Debug version first.



Figure 2-4. Snapshot of MPLAB debugging Port_IO Demo

Under Debugger -> Select Tool -> PIC32 Starter Kit Under Project -> Build All

Run the program by *F9* or click on the *Run icon* as shown on Figure 2-5. A warning dialog box will prompt you for erase and re-program memories. Click *Yes.*

port_io - PIC32 ETH Starter Kit - MPLAB IDE v8.63	Click Run icon to erase and re-program, answer Yes on the dialog box.
	₩₩₩₩₩₩₩₩₩₩
Debug ♥ ☎ ☞ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩	MPLAB PIC32 Starter Kit
port_io - PIC32 ETH St 🔳 🗖 🗙 🗖 D:\Projects\PIC32STK	WARNING There have been changes made to one or more of the programmable memories.
• • • • • • • • • • • • • • • • • • •	Do you want to erase and re-program these memories before proceeding?
Ind // Enable	change notice enable discrete nins and meak nullums

Figure 2-5. Erase and re-program to run the Port_IO Demo

Observe the *Output Window* (Figure 2-6) for a welcome message. This message is exported from the native DEBUG PORT of PIC32 Starter Kit with the functions DBINIT() and DBPRINT("Welcome to the ...").



Figure 2-6. Debug message output from the native debug port of PIC32 start kit.

Keywords: SSD1963, Microchip PIC32, PIC32 Starter Kits, VS1003B audio codec, Microchip PIC32 Plays MP3, WiFi, 4.3" TFT GUI, 5" TFT GUI, 7" TFT GUI

Browsing the source code downward there are debug messages exported via the UART DEBUG port (via CP2102) as following.

putrsUART("Welcome to the PIC32 PORT input/output example. \r\n"); putrsUART("The build date and time is ... (" __DATE__ "," __TIME__ ") \r\n"); putrsUART("Press SW1 to toggle LED1 and Beep, press SW2 to toggle LED2\r\n");

The result is a message from HyperTerminal (Figure 2-7) as below.



Figure 2-7. Debug message output from CP2102 UART DEBUG port (J1U)

Now, press SW1, observe the debug message "Switch SW1 has been pressed", "Switch SW1 has been released". LED3 at RD2 will blink after the "key pressed" message. At the same time, observe the HyperTerminal for the same debug message.

Finally, swap to Release mode and re-program the board for standalone application. No *MPLAB Output Window* message is available but the HyperTerminal messages remain. This is very important for our next project to know the IP address assigned by wireless LAN router from this method.

Now, you have successfully finished the first demo with debug messages on Windows HyperTerminal.

Keywords: SSD1963, Microchip PIC32, PIC32 Starter Kits, VS1003B audio codec, Microchip PIC32 Plays MP3, WiFi, 4.3" TFT GUI, 5" TFT GUI, 7" TFT GUI

Chapter 3. TCP/IP demo program using wireless network

A demo application to use the wireless LAN module is included under the location ...\Firmware\MCHP_2011_07_14\TCPIP\Demo App. PIC32 GP Starter Kit and PIC32 USB Starter kit are supported with project names:

C32-PIC32_GP_SK__MRF24WB_MMVEK.mcp C32-PIC32_USB_SK__MRF24WB_MMVEK.mcp

PIC32 Ethernet Starter Kit is not supported for this demo yet.

This application has been modified from the original TCPIP demo application in Microchip Solutions v2011-07-14\TCPIP folder for hardware profile fitting our own evaluation kit. The TCPIP stack version is v5.36, with Microchip MRF24WB0MA onboard. In the project folder there is a readme.txt describing what we have done to port the original TCPIP demo to our hardware.

Procedures here describe what to do:

1. Solder a Microchip MRF24WB0MA/B on MMEVK board. Some solder flux will help this procedure. Please make sure no short circuit is made.



Figure 3-1. Solder MRF24WB0MA/B on MMEVK

- 2. Stack the PIC32 Starter kit of your choice on board. It is a PIC32 USB STK in this example. Connect the USB cable to the debug port of PIC32 USB STK.
- Connect the second USB cable to the CP2102 UART Debug (J1U) for messages from the emulated COM PORT. No external 5V is required. Figure 3-2 illustrates the setup.



Figure 3-2. Hardware setup for wireless LAN demo

- 4. By the time a USB cable is connected to the UART DEBUG port at J1U, an emulated COM PORT will be available under Windows Control Panel → Device Manager → Ports (COM & LPT) provided that the Silabs VCP has been installed. Launch HyperTerminal, set COM PORT to the emulated PORT number of your PC. In my case, it is a COM7. Use the baud rate 19200bps, 8-N-1 with Flow control to *None*.
- 5. Launch MPLAB and open the project C32-PIC32_USB_SK_MRF24WB_MMVEK.mcp or C32-PIC32_GP_SK_MRF24WB_MMVEK.mcp.

Under Debugger \rightarrow Select Tool \rightarrow PIC32 Starter Kit Under Project \rightarrow Build All Finally click *Run* to program the starter kit. From HyperTerminal, there will be a message as shown on Figure 3-3.



Figure 3-3. Debug message from UART DEBUG at J1U

The result above indicates a successful connection between the MCU on PIC32 starter kit with the WiFi module. It is a good sign if you can get this screen. *The last message indicates a connection failure, but why?*

Keywords: SSD1963, Microchip PIC32, PIC32 Starter Kits, VS1003B audio codec, Microchip PIC32 Plays MP3, WiFi, 4.3" TFT GUI, 5" TFT GUI, 7" TFT GUI

It is because the security configuration defined under WF_Config.h file has been set to *WF_SECURITY_OPEN*. See dump screen in Figure 3-4 below for details. This is the default configuration of the original Microchip example.

D:\)	WF_Config.h	
85 86 87	<pre>/* To DISABLE the serial console applicati /* To ENABLE the serial console applicatio /*</pre>	on: Comment out '#define WF_CONSOLE' */
00 89 90 91	<pre>//#define WF_CONSOLE_IFCFGUTIL /* needed //#define WF_CONSOLE_IFCFGUTIL /* needed</pre>	for console demo */
92 93 94	/* /* Default settings for Connection Manageme /*	*/ mt */
95	#define MY_DEFAULT_SSID_NAME	"linksys"
96 97 98	#define MY_DEFAULT_NETWORK_TYPE	WF_INFRASTRUCTURE /* WF_INFRASTRUCTURE or WF_ADH
99 100	#define MY_DEFAULT_SCAN_TYPE	WF_ACTIVE_SCAN /* WF_ACTIVE_SCAN or WF_PASSIV
101 102		<pre>{1,6,11} /* Desired channel list</pre>
103	#define MY DEFAULT LIST RETRY COUNT ADHOC	(3) /* Number of times to
104 105	#define MY_DEFAULT_LIST_RETRY_COUNT_INFRAST	RUCTURE (WF_RETRY_FOREVER) /* Number of times to
106 107 108 109 110 111	#define MY_DEFAULT_EVENT_NOTIFICATION_LIST	(WF_NOTIFY_CONNECTION_ATTEMPT_SUCCESSFUL WF_NOTIFY_CONNECTION_ATTEMPT_FAILED WF_NOTIFY_CONNECTION_TEMPORARILY_LOST WF_NOTIFY_CONNECTION_PERMANENTLY_LOST WF_NOTIFY_CONNECTION_REESTABLISHED)
112 113	#define MY_DEFAULT_PS_POLL	WF_DISABLED /* WF_DISABLED or WF_ENABLED
114	#define MY_DEFAULT_WIFI_SECURITY_MODE	WF_SECURITY_OPEN
		>

 $Figure \ 3-4. \ WF_Config.h \ header \ file \ for \ wireless \ LAN \ configurations$

Keywords: SSD1963, Microchip PIC32, PIC32 Starter Kits, VS1003B audio codec, Microchip PIC32 Plays MP3, WiFi, 4.3" TFT GUI, 5" TFT GUI, 7" TFT GUI

6. We need to set our wireless router for no security to suit the firmware. Else, we will have to change the firmware to suit the router. It is more easy to do the first in this demo. It is a Linksys WRT54GC router in our office. The Linksys, along with many other popular router brands, uses a built-in webserver on the router to administer the network for both wired and wireless configurations. Please consult the documentation that came with your router for further information on configuration and setup. To gain access to this web page, there is a default Linksys web admin URL. Input the *User name* and *Password* (Figure 3-5). By default, the user name is left blank, and the password is admin. In our case, the *User name* is *admin* with a dedicated password. Again, please remember what you or your network administrator has done for these fields.

🔗 www.TechToys.com.hk - Windows Internet Explor
OO • E http://192.168.1.1/
File Edit View Favorites Tools Help
🚖 Favorites 🛛 🚖 🌄 Suggested Sites 👻 🙋 Free Hotmail
Connect to 192.168.1.1
The server 192.168.1.1 at WRT54GC requires a username and password.
Warning: This server is requesting that your username and password be sent in an insecure manner (basic authentication without a secure connection). Enter the correct <i>User name</i> and <i>password</i> . By default the user name is left blank with
User name:
Password:
Remember my password

Now the Setup page will be available. Go to the Basic Wireless Settings tab that shows the SSID. In our case, the Wireless Network Name (SSID) is *linksys* (case sensitive).



Figure 3-6. Wireless basic settings

This explains why it is *linksys* at the definition under the WF_Config.h header, instead of the default SSID MicrochipDemoAP in the original setting.

#define MY_DEFAULT_SSID_NAME "linksys" //New setting under WF_Config.h

I just don't want to change the setting of our router. So, the next option is to change the firmware to suit the SSID name of WRT54GC router. This is why the original SSID 'MicrochipDemoAP" has been changed to "linksys". You will have to change this SSID as well if you don't want to change the router's SSID, because you may be using the same router for internet. In such case, because make sure the SSID defined under the file WF_Config.h is changed to suit your router.

Next, go to the *Wireless Security* tab, under the *Security Mode*, set it to *Disabled*. Finally, *Save Settings*. Make sure this setting will be restored to the original state if you don't want other devices around your workplace to share the wireless network, especially if your network provider charges by connection time.

LINKSYS [®] A Division of Cisco Systems, Inc.		Firmva	re Version: 1.00.7
		Compact Wireless-G Broadband Router	WRT54GC
Wireless	Setup Wireless	Security Access Applications & Administration	Status
	Basic Wireless Settings	Wireless security Wireless MAC Filter Advanced Wireless	Settings
Wireless Security		Help	
	Security Mode:	WPA Personal	
WPA	Encryption:	WEP WPA Personal	
	Passphrase:	WPA2 Personal WPA2 Mixed Mode	
	Key Renewal:	3600 seconds	l
			ISCO SYSTEMS
		Save Settings Cancel Changes	ومتاللتسيناللت

Figure 3-7. Disable Security Mode for demo purpose

Now, everything is set. Reboot the PIC32 start kit by removing reconnect the USB cable for PIC32 Starter Kit. This time, a new IP address was successfully assigned to my hardware of value 192.168.1.16 as shown on Figure 3-8.



Figure 3-8. Successful wireless LAN connection with new IP address

Verify by using command console, type in *ping 192.168.1.16*. The following dump screen indicates a successful wireless connection. The IP address 192.168.1.16 has been assigned by the Linksys router to our own wireless network device!



Figure 3-9. Verification of a successful WiFi connection with "ping"

Finally, launch the utility Microchip MPFS Generator from All Programs \rightarrow Microchip \rightarrow TCPIP Stack v5.36.2 \rightarrow MPFS2 (see Figure3-10). This utility was installed with Microchip Application Libraries.

(w-)	ilezilla 🕅	•	m TCPIP Stack v5.31	•	
Microsoft Office Wor	i Microchip	۶	Readme for MAL v2010-10-19		🛅 Browse Demo Board Files
MPLAB IDE V8.63	m VMware	•	Serial Boot Loader for PIC32MX	•	🚞 Browse Microchip Solutions Folder
10 END 10 E V0.03	microCode Studio	۰, F	🗃 PICkit 3		Google PowerMeter Information
Protel 99 SE	iAR Systems	ы	Android Accessory Framework	C	Hash Table Calculator
	🛅 Texas Instruments	- •	m Graphics Library v3.01	•	MPFS2
Penpower JR.	🛅 Altium	•	m MCHPFSUSB v2.9a		Releace Notes for v5.36.2
	m Mentor Graphics	×	m MDD File System v1.3.2	•	SNMP mib2bib converter readme
All Programs 👂	im Mentor Graphics SDD	•	miwi(TM) DE v4.2.2	•	🔏 TCPIP Configuration Wizard
	Cocklight V1.7	×	mTouchCap Library v1.31	•	TCPIP Discoverer
	m APAN Software	•	🛅 Smart Card Library v1.02.2	•	😰 TCPIP Stack Help
🯄 start 🛛 🗐 🗉	ANEOS Label Designer for Windows	•	m TCPIP Stack v5.36.2	Þ	📩 TCPIP Stack Help - PDF

Figure 3-10. Launch Microchip MPFS Generator

From Source Settings, browse to the bin file at ..\Demo App\MPFSImg2.bin. Leave the Upload Settings default, and click Upload.

Source	e Settings	ICT UTOT		
Jourc	Ctart Mith.		O Webpage Directory	Pre-Built MPES Image
	Start with		O webpage birectory	Of the built with 5 things
1.	Source Di	rectory:		
	C:\Microchip	Solutions\TCP	IP DemoApp\MPFSImg2.bin	Browse
Uploa	d Settings			
	Upload In	nage To:		
ł.	http://	admin@MCHPF	80ARD/ (==> to modify ==>)) Settings
		_		and the second
			Upload	Date JUNE,16 2011 Version MPES 2.2
-				Version MITS 2.2
4	Browse For	BIN File		
	Look in:	📄 Demo Ap	op	 Image: Image: Ima
	à	MPLAB Y		
	My Recent	Obi-C32-	PIC32 GP SK MRF24WB	
	Documents	Obj-C32-	PIC32_USB_SK_MRF24WB	
	100	Precompil	led Hex	
		Construction SSLKeys		
	Desktop	WebPage		
		MPFSImg	2	
	My Documents			
	My Documents			
	23			
	My Computer			
		File name:	MPESImp2.hip	Open
	My Network		In a sungeroux	
	Discor	Filos of huppy	W DOTAL	Capital

Figure 3-11. Make use of Microchip MPFS Generator utility to program the EEPROM 25LC256 to store a web page

The message box on Figure 3-12 shows the pre-built MPFS image has been uploaded to the EEPROM (25LC256) onboard. From now on, the board is accessible at the mchpboard host name or at the board's IP address. When accessed in a web browser, a real-time update of the board's controls is displayed.

i) The MPFS2 image	ge upload was successfully completed.	
Uploading MPFS2 image: 3	31400 bytes	

Figure 3-12. MPFS2 bin file uploaded successfully

As a final test, try clicking on the LEDs on the web page, and press buttons on PIC32 Starter Kit to observe the real-time changes.

		CP/IP Stack Demo Applica
Overview	Welcome!	
)ynamic Variables		LEDS: (click to toggle)
Form Processing	Stack Version: V5.36 Build Date: Jul 25 2011 00:08:00	Buttons:
Authentication	This site demonstrates the power, flexibility,	A A A A
Cookies	and scalability of an 8, 16, or 32-bit embedded web server. Everything you see is powered by a Microchip BIC microcontroller	
File Uploads	running the Microchip TCP/IP Stack.	
Contraction of the Contraction o		
Send E-mail	On the right you'll see the current status of th example, click the LEDs to toggle the lights on	e demo board. For a quick the board. Press the push
Send E-mail Dynamic DNS	On the right you'll see the current status of th example, click the LEDs to toggle the lights on buttons (except MCLR!) or turn the potentiome update immediately. This examples uses AJAX (e demo board. For a quick the board. Press the push eter and you'll see the status techniques to provide real-tin
Send E-mail Dynamic DNS Network	On the right you'll see the current status of th example, click the LEDs to toggle the lights on buttons (except MCLRI) or turn the potentione update immediately. This examples uses AJAX I feedback.	e demo board. For a quick the board. Press the push ater and you'll see the status techniques to provide real-tin
Send E-mail Dynamic DNS Network Configuration	On the right you'll see the current status of th example, click the LEDs to toggle the lights on buttons (except MCLR!) or turn the potentiome update immediately. This examples uses AJAX I feedback. This site is provided as a tutorial for the variou convertingledimet.	e demo board. For a quick the board. Press the push ster and you'll see the status techniques to provide real-tin is features of the HTTP web

Figure 3-13. Real-time update of the board's controls is displayed

Keywords: SSD1963, Microchip PIC32, PIC32 Starter Kits, VS1003B audio codec, Microchip PIC32 Plays MP3, WiFi, 4.3" TFT GUI, 5" TFT GUI, 7" TFT GUI

Chapter 4. Interfacing Solomon SSD1963 display controller

SSD1963 is a display controller of 1215k bytes frame buffer to support up to 864x480x24bit graphics content. We have a demo kit SSD1963 EVK R3B with all necessary circuits to facilitate testing the chip for online purchase. A demonstration program is described in this chapter to access SSD1963 in 16-bit 8080 addressing mode. No complex functions, no GUI, no touch panel, but the basic IO command for addressing SSD1963 is provided here. The PIC32 MCU is to generate the required control signal (CS#, DC, RD#, WR#, and D[15:0]) to display a single pixel on one of the 4.3" to 7" display panels. The low level driver developed in this section lays the foundation for more complex applications such as displaying jpeg pictures from a microSD card, Windows-style GUI with icons, animation, etc.

The hardware configuration is illustrated in Figure 4-1below.



Figure 4-1. Hardware setup for interfacing the SSD1963 Evaluation Kit

The source code of this project is located under the folder Firmware $MCHP_2011_07_14$ GraphicsPrimitives LLD.

Again, projects for all three MCU variants are available. Because it is a PIC32 Ethernet Starter kit in this example, we will select PIC32 ETH SK Primitives SSD1963 LLD.mcp

Name 🔺	Size	Туре
🛅 Objects - PIC32 ETH SK Primitive SSD1963 LLD		File Folder
🚞 Objects - PIC32 GP SK Primitive SSD1963 LLD		File Folder
🛅 Objects - PIC32 USB SK Primitive SSD1963 LLD		File Folder
🗒 MainDemo	4 KB	C File
🔤 MainDemo.d	1 KB	D File
MainDemo.o	13 KB	O File
TIC32 ETH SK Primitives SSD1963 LLD	2 KB	Microchip MPLAB.Pr
N PIC32 ETH SK Primitives SSD1963 LLD	45 KB	Microchip MPLAB.W
PIC32 ETH SK Primitives SSD1963 LLD.mcs	4 KB	MCS File
Reprint PIC32 GP SK Primitives SSD1963 LLD	2 KB	Microchip MPLAB.Pr
Reprint PIC32 GP SK Primitives SSD1963 LLD	45 KB	Microchip MPLAB.W
PIC32 GP SK Primitives SSD1963 LLD.mcs	4 KB	MCS File
Reprint PIC32 USB SK Primitives SSD1963 LLD	2 KB	Microchip MPLAB.Pr
Reprint PIC32 USB SK Primitives SSD1963 LLD	45 KB	Microchip MPLAB.W
PIC32 LISB SK Primitives SSD1963 LLD.mcs	4 KB	MCS File

User's Guide Revision 1.0 www.TechToys.com.hk TechToys Company All Rights Reserved

Keywords: SSD1963, Microchip PIC32, PIC32 Starter Kits, VS1003B audio codec, Microchip PIC32 Plays MP3, WiFi, 4.3" TFT GUI, 5" TFT GUI, 7" TFT GUI

There are only three steps to use this demo as follows:

- (1) Define the hardware platform under system.h file and comment out //#define PIC32MX_EVK_RD4 #define PIC32_STARTER_KIT
- (2) Define the TFT panel under *TFT.h*, select only one of the TFT panels available from us. In this example, the 5" TFT panel was connected.

//#define DISPLAY_PANEL TY430TFT480272
#define DISPLAY_PANEL TY500TFT800480
//#define DISPLAY_PANEL TY700TFT800480
//#define DISPLAY_PANEL YOUR_PANEL

(3) Finally, rebuild and program.

Online debug can be performed with this example. A breakpoint set at the delay function after running PutPixel(x,y) in a for-loop with color set to BRIGHTBLUE. Result of running the program is illustrated in this Figure 4-3.



Figure 4-3. A breakpoint at PutPixel(x,y) with blue color

End Notes

ⁱ Evaluation kit for Solomon SSD1963 Display Controller, hyperlink: <u>http://www.techtoys.com.hk/Displays/SSD1963EvalRev3B/SSD1963%20Eval%20Board%20Rev3B.htm</u>

ⁱⁱ Microchip PIC32 Starter Kit, hyperlink: <u>http://www.microchip.com/stellent/idcplg?ldcService=SS_GET_PAGE&nodeId=2615&dDocName=en532453</u>

ⁱⁱⁱ Microchip PIC32 USB Starter Kit, hyperlink: <u>http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2615&dDocName=en535536</u>

^{iv} Microchip PIC32 Ethernet Starter Kit, hyperlink <u>http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2615&dDocName=en545713</u>

v VLSI Solution http://www.vlsi.fi/

vi Microchip MRF24WB0MA/B WiFi modules http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2885¶m=en547232