Stellaris® LM4F120 LaunchPad Evaluation Board

User's Manual

EK-LM4F120XL-UM-01
SMPU289

Copyright

Texas Instruments
108 Wild Basin, Suite 350
Austin, TX 78746
http://www.ti.com/stellaris

Table of Contents

List of Figures

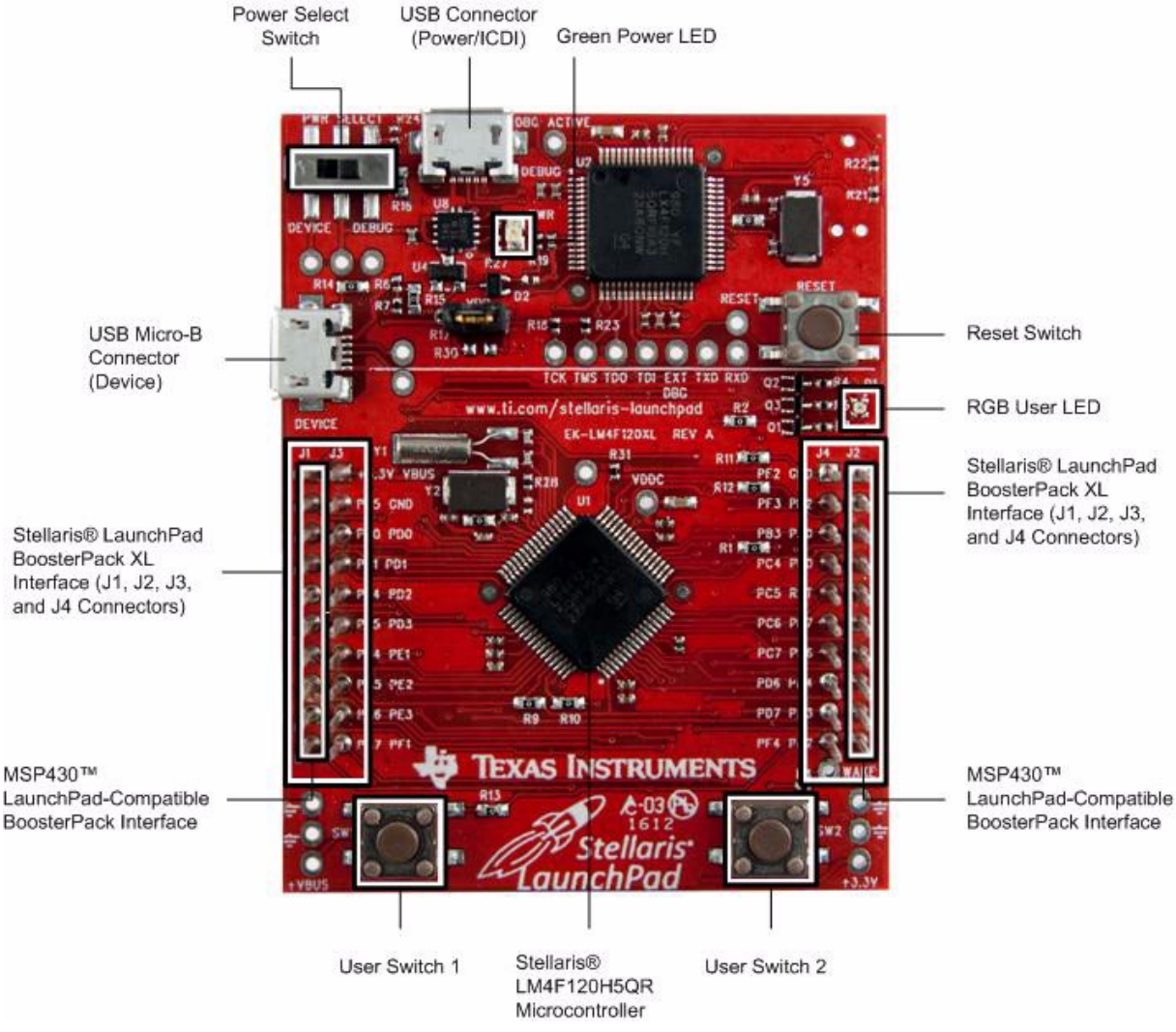
List of Tables

CHAPTER 1

Board Overview

The Stellaris® LM4F120 LaunchPad Evaluation Board (EK-LM4F120XL) is a low-cost evaluation platform for ARM® Cortex™-M4F-based microcontrollers. The Stellaris® LaunchPad's design highlights the LM4F120H5QR microcontroller's USB 2.0 Device interface and Hibernation module. The Stellaris® LaunchPad also features programmable user buttons and an RGB LED for custom applications. The stackable headers of the Stellaris® LM4F120 LaunchPad BoosterPack XL Interface demonstrate how easy it is to expand the functionality of the Stellaris® LaunchPad when interfacing to other peripherals with Stellaris® BoosterPacks and MSP430™ BoosterPacks. Figure 1-1 shows a photo of the Stellaris® LaunchPad.

Figure 1-1. Stellaris® LM4F120 LaunchPad Evaluation Board



Kit Contents

The Stellaris® LM4F120 LaunchPad Evaluation Kit comes with the following:

- Stellaris® LaunchPad Evaluation Board (EK-LM4F120XL)
- On-board Stellaris® In-Circuit Debug Interface (ICDI)
- USB Micro-B plug to USB-A plug cable
- *README First* document

Using the Stellaris® LaunchPad

The recommended steps for using the Stellaris® LM4F120 LaunchPad Evaluation Kit are:

1. **Follow the README First document included in the kit.** The README First document will help get the Stellaris® LaunchPad up and running in minutes. See the www.ti.com/stellaris-launchpad web site for additional information to get started.
2. **Experiment with LaunchPad BoosterPacks.** Stellaris® BoosterPacks and compatible MSP430™ BoosterPacks can be found at the www.ti.com/stellaris-launchpad web site.
3. **Take your first step toward developing an application with Project 0 using your preferred ARM tool-chain and the Stellaris Peripheral Driver Library.** Software applications are loaded using the on-board Stellaris® In-Circuit Debug Interface (ICDI). See Chapter 3, “Software Development” on page 20, for the programming procedure. The *StellarisWare Peripheral Driver Library Software Reference Manual* contains specific information on software structure and function. For more information on Project 0, go to the www.ti.com/stellaris-launchpad/project0 web site.
4. **Customize and integrate the hardware to suit an end application.** This user's manual is an important reference for understanding circuit operation and completing hardware modification.

Features

The Stellaris® LaunchPad includes the following features:

- Stellaris® LM4F120H5QR microcontroller
- USB Micro-B connector for USB Device
- RGB user LED
- 2 user switches (application/wake)
- Available I/O brought out to headers on a 0.1" grid
- On-board Stellaris® In-Circuit Debug Interface (ICDI)
- Switch-selectable power sources
  - ICDI
  - USB Device
- Reset switch
- Preloaded RGB quickstart application
- Supported by StellarisWare® software including the USB library and the peripheral driver library

- Stellaris® LM4F120 LaunchPad BoosterPack XL Interface which features stackable headers to expand the capabilities of the Stellaris® LaunchPad development platform
  - For a complete list of available BoosterPacks that can be used with the Stellaris® LaunchPad, see the www.ti.com/stellaris-launchpad web site.

BoosterPacks

Stellaris® LaunchPad provides an easy and inexpensive way to develop applications with the Stellaris® LM4F120H5QR microcontroller. Stellaris® BoosterPacks and MSP430™ BoosterPacks expand the available peripherals and potential applications of the Stellaris® LaunchPad. BoosterPacks can be used with the Stellaris® LaunchPad or just use the on-board LM4F120H5QR microcontroller as its processor. See"(Microcontroller, USB, Expansion, Buttons, and LED (Schematic on page 18)" on page 10 in Chapter 2 for more information.

Build your own BoosterPack and take advantage of Texas Instruments' web site to help promote it!

From sharing a new idea or project, to designing, manufacturing, and selling your own BoosterPack kit, TI offers a variety of avenues for you to reach potential customers with your solutions.

Specifications

Table 1-1 shows the specifications for the Stellaris® LaunchPad.

Table 1-1. EK-LM4F120XL Specifications

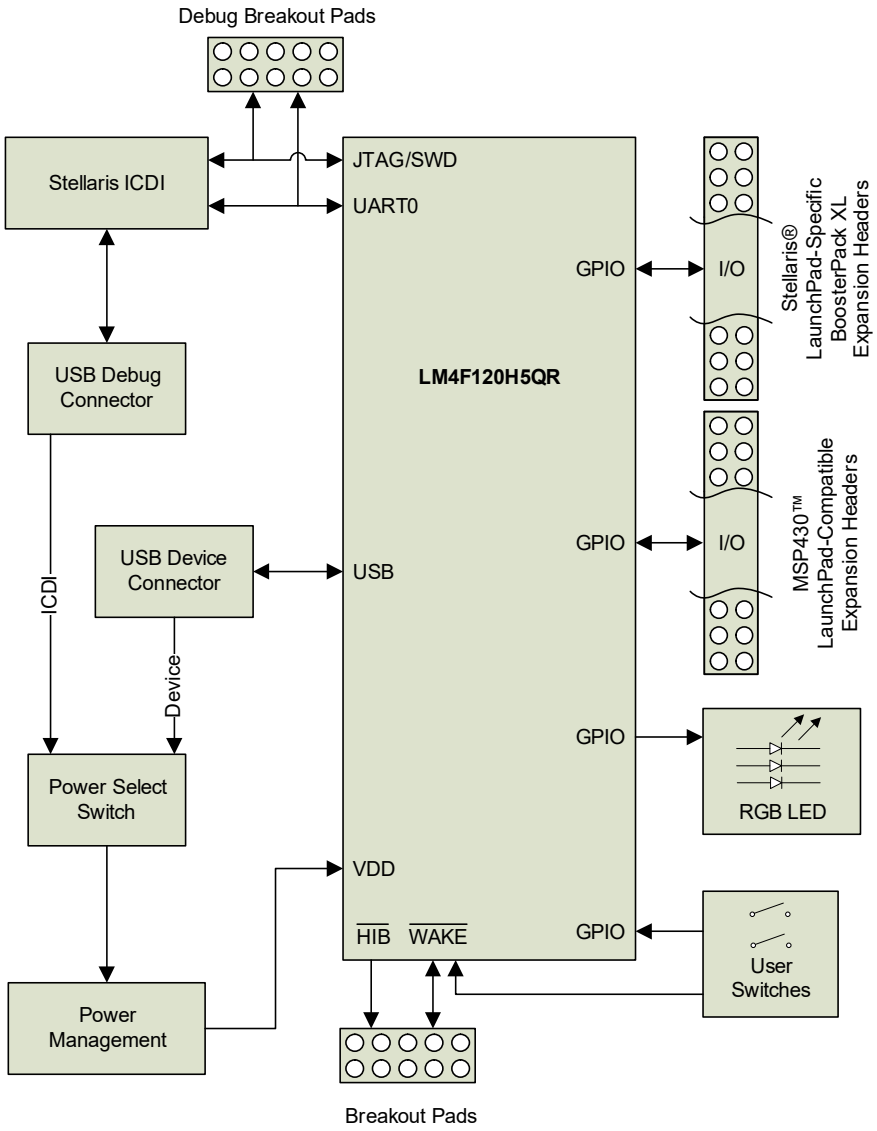
Parameter	Value
Board supply voltage	4.75–5.25 V_{DC} from one of the following sources: ■ Debugger (ICDI) USB Micro-B cable (connected to a PC) ■ USB Device Micro-B cable (connected to a PC)
Dimensions	2.0" x 2.25" x 0.425" (L x W x H)
Break-out power output	■ 3.3 V_{DC} (300 mA max) ■ 5.0 V_{DC} (depends on 3.3 V_{DC} usage, 23 mA - 323 mA)
RoHS status	Compliant

C H A P T E R 2

Hardware Description

The Stellaris® LaunchPad includes a Stellaris LM4F120H5QR microcontroller and an integrated Stellaris® In-Circuit Debug Interface (ICDI) as well as a range of useful peripheral features (see the block diagram in Figure 2-1). This chapter describes how these peripherals operate and interface to the microcontroller.

Figure 2-1. Stellaris® LaunchPad Evaluation Board Block Diagram



Functional Description

(Microcontroller, USB, Expansion, Buttons, and LED (Schematic on page 18)

Microcontroller

The Stellaris LM4F120H5QR is a 32-bit ARM® Cortex™-M4F-based microcontroller with 256-KB Flash memory, 32-KB SRAM, 80-MHz operation, USB Device, Hibernation module, and a wide range of other peripherals. See the LM4F120H5QR microcontroller data sheet (order number DS-LM4F120H5QR) for complete device details.

Most of the microcontroller signals are routed to 0.1" pitch headers. An internal multiplexer allows different peripheral functions to be assigned to each of these GPIO pads. When adding external circuitry, consider the additional load on the evaluation board's power rails.

The LM4F120H5QR microcontroller is factory-programmed with a quickstart demo program. The quickstart program resides in on-chip Flash memory and runs each time power is applied, unless the quickstart application has been replaced with a user program.

USB Device

The Stellaris® LaunchPad includes a USB Micro-B connector to allow for USB 2.0 Device operation. The signals shown in Table 2-1 are used for USB Device.

Table 2-1. USB Device Signals

GPIO Pin	Pin Function	USB Device
PD4	USB0DM	D-
PD5	USB0DP	D+

When connected as a USB Device, the evaluation board can be powered from either the Stellaris® ICDI or the USB Device connectors. The user can select the power source by moving the POWER SELECT switch (SW3) to the Device position. See the Power Management schematic on page 19

User Switches and RGB User LED

The Stellaris® LaunchPad comes with an RGB LED. This LED is used in the preloaded RGB quickstart application and can be configured for use in custom applications.

Two user buttons are included on the board. The user buttons are both used in the preloaded quickstart application to adjust the light spectrum of the RGB LED as well as go into and out of hibernation. The user buttons can be used for other purposes in the user's custom application.

The evaluation board also has a green power LED.

Table 2-2 shows how these features are connected to the pins on the microcontroller.

Table 2-2. User Switches and RGB LED Signals

GPIO Pin	Pin Function	Feature
PF4	GPIO	SW1
PF0	GPIO	SW2

Table 2-2. User Switches and RGB LED Signals (Continued)

GPIO Pin	Pin Function	Feature
PF1	GPIO	RGB LED (red)
PF2	GPIO	RGB LED (blue)
PF3	GPIO	RGB LED (green)

Headers and BoosterPacks

The two double rows of stackable headers are mapped to most of the GPIO pins of the LM4F120H5QR microcontroller. These rows are labeled as connectors J1, J2, J3, and J4. Connectors J3 and J4 are located 0.1 inches inside of the J1 and J2 connectors. All 40 header pins of the J1, J2, J3, and J4 connectors make up the Stellaris® LM4F120 LaunchPad BoosterPack XL Interface. Table 2-3, Table 2-4, Table 2-5, and Table 2-6 show how these header pins are connected to the microcontroller pins and which GPIO functions can be selected.

NOTE: To configure the device peripherals easily and intuitively using a graphical user interface (GUI), see the Stellaris® LM4F Pinmux Utility found at www.ti.com/tool/lm4f_pinmux. This easy-to-use interface makes setting up alternate functions for GPIOs simple and error-free.

Table 2-3. J1 Connector

J1 Pin	GPIO	Stellaris Pin	GPIOPCTL Register Setting							
			GPIOAMSEL	1	2	3	7	8	9	14
1.01	3.3 V									
1.02	PB5	57	AIN11	-	SSI2Fss	-	T1CCP1	CAN0Tx	-	-
1.03	PB0	45	-	U1Rx	-	-	T2CCP0	-	-	-
1.04	PB1	46	-	U1Tx	-	-	T2CCP1	-	-	-
1.05	PE4	59	AIN9	U5Rx	-	I2C2SCL	-	CAN0Rx	-	-
1.06	PE5	60	AIN8	U5Tx	-	I2C2SDA	-	CAN0Tx	-	-
1.07	PB4	58	AIN10	-	SSI2Clk	-	T1CCP0	CAN0Rx	-	-
1.08	PA5	22	-	-	SSI0Tx	-	-	-	-	-
1.09	PA6	23	-	-	-	I2C1SCL	-	-	-	-
1.10	PA7	24	-	-	-	I2C1SDA	-	-	-	-

Table 2-4. J2 Connector

J2 Pin	GPIO	Stellaris Pin	GPIOPCTL Register Setting							
			GPIOAMSEL	1	2	3	7	8	9	14
2.01	GND									
2.02	PB2	47	-	-	-	I2C0SCL	T3CCP0	-	-	-
2.03	PE0	9	AIN3	U7Rx	-	-	-	-	-	-
2.04	PF0	28	-	U1RTS	SSI1Rx	CAN0Rx	T0CCP0	NMI	C0o	-
2.05	RESET									
2.06[a]	PB7	4	-	-	SSI2Tx	-	T0CCP1	-	-	-
2.07[b]	PB6	1	-	-	SSI2Rx	-	T0CCP0	-	-	-

Table 2-4. J2 Connector (Continued)

J2 Pin	GPIO	Stellaris Pin	GPIOPCTL Register Setting							
			GPIOAMSEL	1	2	3	7	8	9	14
2.08	PA4	21	-	-	SSI0Rx	-	-	-	-	-
2.09	PA3	20	-	-	SSI0Fss	-	-	-	-	-
2.10	PA2	19	-	-	SSI0Clk	-	-	-	-	-

a. J2.06 (PB7) is also connected via 0-Ω resistor to J3.04 (PD1).
b. J2.07 (PB6) is also connected via 0-Ω resistor to J3.03 (PD0).

Table 2-5. J3 Connector

J3 Pin	GPIO	Stellaris Pin	GPIOPCTL Register Setting							
			GPIOAMSEL	1	2	3	7	8	9	14
3.01	5.0V									
3.02	GND									
3.03	PD0	61	AIN7	SSI3Clk	SSI1Clk	I2C3SCL	WT2CCP0	-	-	-
3.04	PD1	62	AIN6	SSI3Fss	SSI1Fss	I2C3SDA	WT2CCP1	-	-	-
3.05	PD2	63	AIN5	SSI3Rx	SSI1Rx	-	WT3CCP0	-	-	-
3.06	PD3	64	AIN4	SSI3Tx	SSI1Tx	-	WT3CCP1	-	-	-
3.07	PE1	8	AIN2	U7Tx	-	-	-	-	-	-
3.08	PE2	7	AIN1	-	-	-	-	-	-	-
3.09	PE3	6	AIN0	-	-	-	-	-	-	-
3.10[a]	PF1	29	-	U1CTS	SSI1Tx	-	T0CCP1	-	C1o	TRD1

a. Not recommended for BoosterPack use. This signal tied to on-board function via 0-Ω resistor.

Table 2-6. J4 Connector

J4 Pin	GPIO	Stellaris Pin	GPIOPCTL Register Setting							
			GPIOAMSEL	1	2	3	7	8	9	14
4.01[a]	PF2	30	-	-	SSI1Clk	-	T1CCP0	-	-	TRD0
4.02[a]	PF3	31	-	-	SSI1Fss	CAN0Tx	T1CCP1	-	-	TRCLK
4.03	PB3	48	-	-	-	I2C0SDA	T3CCP1	-	-	-
4.04	PC4	16	C1-	U4Rx	U1Rx	-	WT0CCP0	U1RTS	-	-
4.05	PC5	15	C1+	U4Tx	U1Tx	-	WT0CCP1	U1CTS	-	-
4.06	PC6	14	C0+	U3Rx	-	-	WT1CCP0	-	-	-
4.07	PC7	13	C0-	U3Tx	-	-	WT1CCP1	-	-	-
4.08	PD6	53	-	U2Rx	-	-	WT5CCP0	-	-	-
4.09	PD7	10	-	U2Tx	-	-	WT5CCP1	NMI	-	-
4.10[a]	PF4	5	-	-	-	-	T2CCP0	-	-	-

a. Not recommended for BoosterPack use. This signal tied to on-board function via 0-Ω resistor.

Connectors J1 and J2 of the Stellaris® LM4F120 LaunchPad BoosterPack XL Interface provide compatibility with MSP430™ LaunchPad BoosterPacks. Highlighted functions in Table 2-3, "J1 Connector" on page 11 and Table 2-4, "J2 Connector" on page 11 indicate configuration for compatibility with the MSP430 LaunchPad.

A complete list of Stellaris® BoosterPacks and Stellaris LaunchPad-compatible MSP430 BoosterPacks is available at www.ti.com/stellaris-launchpad.

Power Management (Schematic on page 19)

Power Supplies

The Stellaris® LaunchPad can be powered from one of two power sources:

- On-board Stellaris® In-Circuit Debug Interface (ICDI) USB cable (Debug, Default)
- USB Device cable (Device)

The POWER SELECT switch (SW3) is used to select one of the two power sources. Select only one source at a time.

Hibernate

The Stellaris® LaunchPad provides an external 32.768 kHz crystal (Y1) as the clock source for the LM4F120H5QR's Hibernation module clock source. The current draw while in Hibernate mode can be measured by making some minor adjustments to the Stellaris® LaunchPad. This is explained in more detail later in this section.

The conditions that can generate a wake signal to the Hibernate module on the Stellaris® LaunchPad are waking on a Real-time Clock (RTC) match and/or waking on assertion of the $\overline{\text{WAKE}}$ pin.[1] The second user switch (SW2) is connected to the $\overline{\text{WAKE}}$ pin on the microcontroller. The $\overline{\text{WAKE}}$ pin, as well as the V_{DD} and $\overline{\text{HIB}}$ pins, are easily accessible through breakout pads on the Stellaris® LaunchPad. See Appendix A, "Schematics" on page 22 for details.

There is no external battery source on the Stellaris® LaunchPad Hibernation module, which means the VDD3ON power control mechanism should be used. This mechanism uses internal switches to remove power from the Cortex-M4F processor as well as to most analog and digital functions while retaining I/O pin power.

To measure the Hibernation mode current or the Run mode current, the V_{DD} jumper that connects the 3.3 V pin and the MCU_PWR pin must be removed. See Appendix A, "Schematics" on page 17 for details on these pins and component locations. An ammeter should then be placed between the 3.3 V pin and the MCU_PWR pin to measure I_{DD} (or I_{HIB_VDD3ON}). The LM4F120H5QR microcontroller uses V_{DD} as its power source during $\overline{V}_{DD3ON}$ Hibernation mode, so I_{DD} is the Hibernation mode (VDD3ON mode) current. This measurement can also be taken during Run mode, which measures I_{DD} the microcontroller running current.

Clocking

The Stellaris® LaunchPad uses a 16.0-MHz crystal (Y2) to complete the LM4F120H5QR microcontroller's main internal clock circuit. An internal PLL, configured in software, multiples this clock to higher frequencies for core and peripheral timing.

The Hibernation module is clocked from an external 32.768 kHz crystal (Y1).

1. If the board does not turn on when you connect it to a power source, the microcontroller might be in Hibernate mode (depending on the programmed application). You must satisfy one of the programmed wake conditions and connect the power to bring the microcontroller out of Hibernate mode and turn on the board.

Reset

The $\overline{\text{RESET}}$ signal into the LM4F120H5QR microcontroller connects to the RESET switch and to the Stellaris® ICDI circuit for a debugger-controlled reset.

External reset is asserted (active low) under any of three conditions:

- Power-on reset (filtered by an R-C network)
- RESET switch held down
- By the Stellaris® ICDI circuit when instructed by the debugger (this capability is optional, and may not be supported by all debuggers)

Stellaris In-Circuit Debug Interface (ICDI) (Schematic on page 20)

Stellaris® In-Circuit Debug Interface (ICDI)

The Stellaris® LaunchPad evaluation board comes with an on-board Stellaris® In-Circuit Debug Interface (ICDI). The Stellaris® ICDI allows for the programming and debug of the LM4F120H5QR using LM Flash Programmer and/or any of the supported tool chains. Both JTAG and Serial Wire Debug (SWD) are supported.

Table 2-7 shows the pins used for JTAG and SWD. These signals are also mapped out to easily accessible breakout pads and headers on the board.

Table 2-7. Stellaris® In-Circuit Debug Interface (ICDI) Signals

GPIO Pin	Pin Function
PC0	TCK/SWCLK
PC1	TMS/SWDIO
PC2	TDI
PC3	TDO/SWO

Virtual COM Port

When plugged in to a PC, the device enumerates as a debugger and a virtual COM port. Table 2-8 shows the connections for the COM port to the pins on the microcontroller.

Table 2-8. Virtual COM Port Signals

GPIO Pin	Pin Function
PA0	U0RX
PA1	U0TX

CHAPTER 3

Software Development

This chapter provides general information on software development as well as instructions for Flash memory programming.

Software Description

The StellarisWare® software provided with the Stellaris® LaunchPad provides access to all of the peripheral devices supplied in the design. The Stellaris® Peripheral Driver Library is used to operate the on-chip peripherals as part of StellarisWare®.

StellarisWare® includes a set of example applications that use the StellarisWare® Peripheral Driver Library. These applications demonstrate the capabilities of the LM4F120H5QR microcontroller, as well as provide a starting point for the development of the final application for use on the Stellaris® LaunchPad evaluation board.

Source Code

The complete source code including the source code installation instructions are provided at www.ti.com/stellaris-launchpad. The source code and binary files are installed in the DriverLib tree.

Tool Options

The source code installation includes directories containing projects and/or makefiles for the following tool-chains:

- Keil ARM RealView® Microcontroller Development System
- IAR Embedded Workbench for ARM
- Sourcery CodeBench
- Texas Instruments' Code Composer Studio™ IDE

Download evaluation versions of these tools from www.ti.com/stellaris. Due to code size restrictions, the evaluation tools may not build all example programs. A full license is necessary to re-build or debug all examples.

Instructions on installing and using each of the evaluation tools can be found in the Quickstart guides (for example, Quickstart-Keil, Quickstart-IAR) which are available for download from the evaluation kit section of our web site at www.ti.com/stellaris.

For detailed information on using the tools, see the documentation included in the tool chain installation or visit the web site of the tools supplier.

Programming the Stellaris LaunchPad Evaluation Board

The Stellaris® LaunchPad software package includes pre-built binaries for each of the example applications. If you installed StellarisWare® to the default installation path of C:\StellarisWare, you can find the example applications in “C:\StellarisWare\boards\ek-lm4f120xl”. The on-board Stellaris ICDI is used with the Stellaris LM Flash Programmer tool to program applications on the Stellaris® LaunchPad.

Follow these steps to program example applications into the Stellaris® LaunchPad evaluation board using the Stellaris® ICDI:

1. Install LM Flash Programmer on a Windows PC.
2. Switch the POWER SELECT switch to the right for Debug mode.
3. Connect the USB-A cable plug to an available port on the PC and the Micro-B plug to the 'Debug' USB port on the board.
4. Verify that the POWER LED D4 on the board is lit.
5. Run LM Flash Programmer.
6. In the Configuration tab, use the Quick Set control to select the EK-LM4F120XL evaluation board.
7. Move to the Program tab and click the Browse button. Navigate to the example applications directory (the default location is "C:\StellarisWare\boards\ek-lm4f120xl\").
8. Each example application has its own directory. Navigate to the example directory that you want to load and then into the directory which contains the binary (*.bin) files. Select the binary file and click Open.
9. Set the "Erase Method" to "Erase Necessary Pages," check the "Verify After Program" box, and check "Reset MCU After Program".
10. Click the Program button to start the Erase, Download, and Verify process.

Program execution starts once the Verify process is complete.

APPENDIX A

Schematics

This section contains the schematics for the Stellaris® LaunchPad board.