



Stellaris® LM3S811 Evaluation Kit for CodeSourcery's Sourcery G++

This Quickstart shows you how to install the Sourcery G++ tool, compile the StellarisWare™ package, and flash/debug an application using GDB. The LM3S811 Evaluation Board (EVB) can function as either a complete evaluation target, or as a debugger interface to any external Stellaris® device. The included USB cable is all that is needed to provide power and communication to the host PC.

LM3S811 Evaluation Board

Requirements

- You have a PC, with a USB interface, running Microsoft® Windows 2000, XP, or 2003
- You have the Stellaris® LM3S811 Evaluation Kit Documentation and Software CD
- You have installed a command-line environment that includes `make` and the other associated utilities described in the *Stellaris® Peripheral Driver Library User's Guide*. The UnxUtils package is included on the Luminary Micro CD.



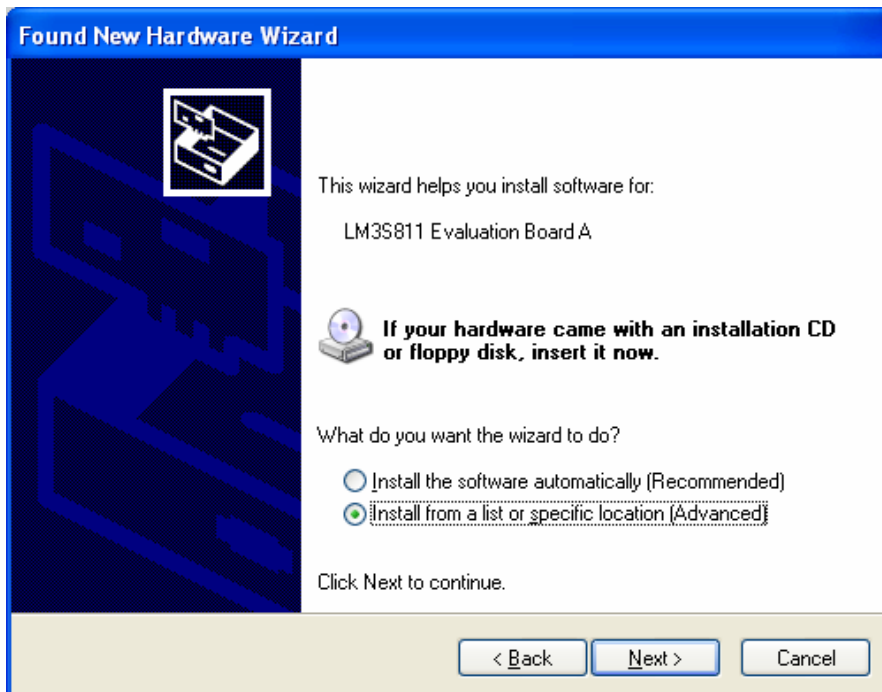
CAUTION: There is a known electrical issue with the FT2232 device that is used in the on-board In Circuit Debug Interface (ICDI). Some USB hubs can cause the device to misbehave, with symptoms ranging from failed enumeration to corrupt data transfers. If you experience trouble when using the on-board ICDI, try connecting the USB cable directly to one of the USB ports on your PC or laptop.

The LM3S811 Evaluation Board is configured for immediate use. To power the EVB, use the USB cable supplied in the kit. Connect the mini-b (smaller) end of the USB cable to the connector labeled “USB” on the EVB. Connect the other end (Type A) to a free USB port on your host PC. The USB is capable of sourcing up to 500 mA for each attached device, which is sufficient for the evaluation board. If connecting the board through a USB hub, it must be a powered hub.

When you plug in the EVB for the first time, Windows starts the Found New Hardware Wizard and asks if Windows can connect to Windows Update to search for software. Select “No, not this time” and then click Next.



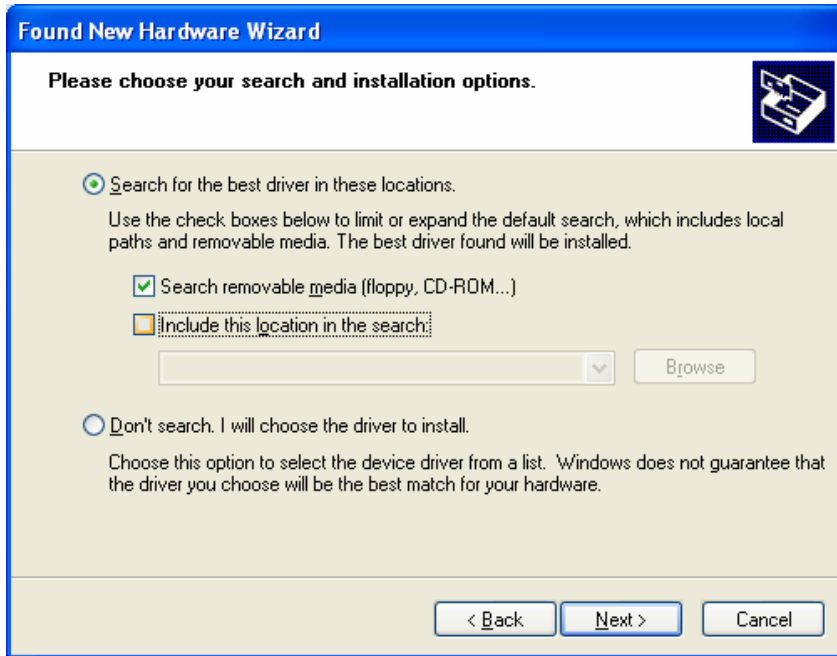
Next, the Found New Hardware Wizard asks you from where to install the software. Select “Install from a list or specific location (Advanced)” and click Next.



EKC-LM3S811 QUICKSTART



Make sure the LM3S811 Evaluation Kit Documentation and Software CD that came with the Evaluation Kit is in your CD-ROM drive. Select “Search for the best driver in these locations,” and check the “Search removable media (floppy, CD-ROM...)” option. Click Next.

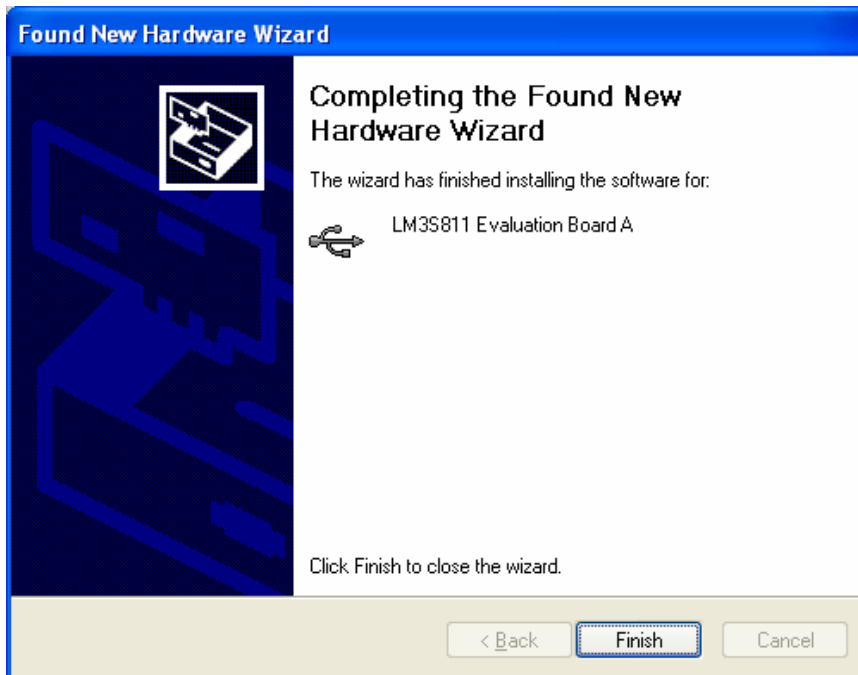


A warning pops up during the Hardware Installation like the one below; click Continue Anyway.





Windows finishes installing the drivers for “LM3S811 Evaluation Board A.” When the driver install is finished, the Found New Hardware Wizard window appears like the one below. Click Finish to close the dialog box.



You have just installed the drivers for “LM3S811 Evaluation Board A.” The USB device built into the EVB is a composite USB device. After you click Finish, a new Found New Hardware Wizard appears asking to install drivers for another device. This is for the “LM3S811 Evaluation Board B” part of the composite USB device. Follow the same instructions as above to install the drivers for this device.

The Found New Hardware Wizard appears one last time. This is to install the drivers for the “LM3S811 Virtual COM Port.” Again, follow the same instructions to install the drivers for this device.

Now all of the hardware drivers for the LM3S811 Evaluation Board have been installed. These drivers give the debugger access to the JTAG interface and the host PC access to the Virtual COM Port.

Quickstart Application

The LM3S811 Evaluation Board comes preprogrammed with a quickstart application. Once you have powered the board, this application runs automatically. You have probably already noticed this running as you installed the drivers. A Luminary Micro and CodeSourcery splash screen appear on the OLED display for a few seconds before the application begins.

The quickstart application is a game in which you navigate a ship through an endless tunnel. Use the potentiometer (POT) to move the ship up and down, and the user pushbutton (USER)

EKC-LM3S811 QUICKSTART



to fire a missile to destroy obstacles in the tunnel. Score accumulates for survival and destroying obstacles. The game lasts for only one ship; the score displays at the end of the game.

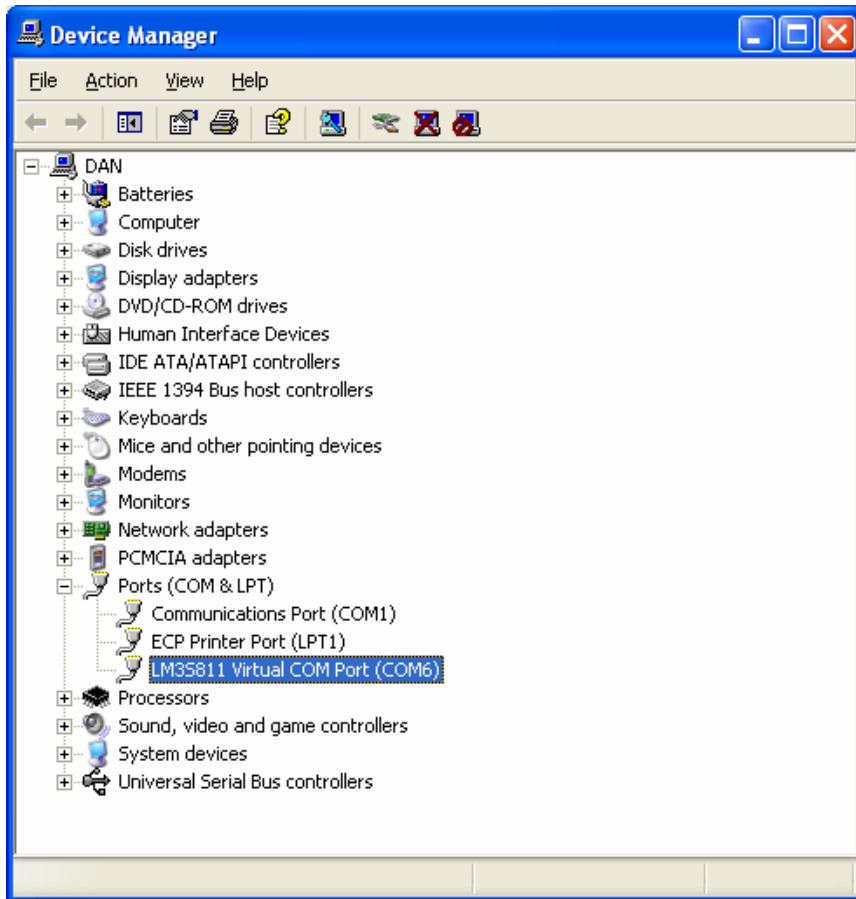
Since the OLED display on the evaluation board has burn-in characteristics similar to a CRT, the application also contains a screen saver. The screen saver only becomes active if two minutes have passed without the user pushbutton being pressed while waiting to start the game (that is, the screen saver never appears during game play). An implementation of the Game of Life is run with a field of random data as the seed value.

After two minutes of running the screen saver, the display turns off and the user LED blinks. Exit either mode of screen saver (Game of Life or blank display) by pressing the user pushbutton (USER). Press the button again to start the game.

While the game is being played, a running tally of the score is output through UART0 of the LM3S811 microcontroller. UART0 is connected to the FTDI's second serial channel. This serial channel is available to Windows as a Virtual COM port. To view the score, open up a terminal application such as HyperTerminal. Connect using COM#, where # is the number Windows has assigned the Virtual COM port. Set the serial connection to a baud rate of 115200, 8 data bits, no parity, 1 stop bit, and no flow control.

To determine which COM# Windows has assigned to the Virtual COM port on the LM3S811 microcontroller, follow these steps:

- 1) From the Start Menu, select Control Panel, then double-click the System icon.
- 2) Select the Hardware tab.
- 3) Click Device Manager.
- 4) Click the + symbol to expand the Ports (COM & LPT) group.
- 5) "LM3S811 Virtual COM Port (COM#)" is listed as shown in the figure below. This COM# is the device you connect to using your terminal application. In this example, the COM port is COM6.



CodeSourcery Sourcery G++

This quickstart shows you how to install the Sourcery G++ tools, and how to build and run an example application on the Luminary Micro Stellaris® LM3S811 Evaluation Board.

NOTE: The GDB stub included with Sourcery G++ is a 30-day time-limited evaluation version. For long-term use, you must purchase a license from CodeSourcery.

Step 1: Install UnxUtils (if necessary)

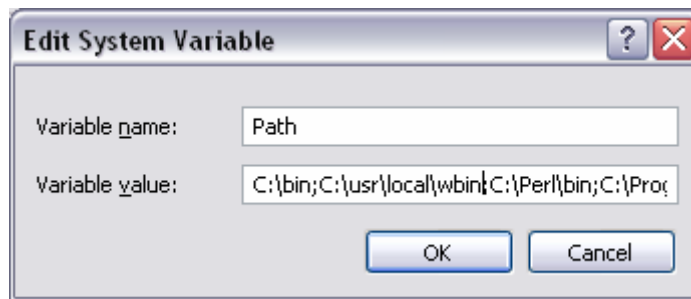
If your PC requires a command-line interface, you may install the UnxUtils package found on the LM3S811 Evaluation Kit Documentation and Software CD.

1. On the LM3S811 Evaluation Kit Documentation and Software CD, navigate to the *Tools* tab and click the *UnxUtils* link in the center of the page.

NOTE: If you are navigating the CD using Windows Explorer (or a similar application), go to the “Tools/UnxUtils” directory.



2. Download the .zip file to your PC and extract the contents to your “C:\” drive. You now have a “bin” and “usr” directory under “C:/”.
3. Make sure that you have “C:/usr/local/wbin” and “C:/bin” in your path. To do this, right-click on My Computer and select Properties. Select the *Advanced* tab and click the Environment Variables button. If you scroll down in the *System variables* box, you will see the path variable. Double-click on *Path* and add “C:/user/local/wbin;C:/bin” to the *beginning* of the string.



4. Open a command-prompt (shell) window by clicking on the `sh.exe` file located in “C:/bin.” Type “which make” to verify that Windows is looking at the correct version of `make`. You should see this:

```
MYPC# which make
c:/usr/local/wbin/make
```

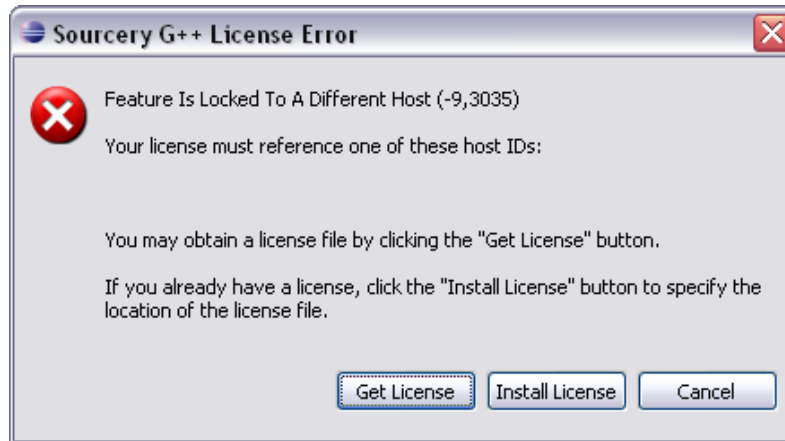
NOTE: It is a good idea to create a shortcut to `sh.exe` and place it somewhere convenient (like your desktop).

Step 2: Install Sourcery G++

1. Insert the LM3S811 Evaluation Kit Documentation and Software CD into the CD-ROM drive of your PC. Navigate to the *Tools* section of the CD and click on the CodeSourcery logo to run the installer. Depending on your web browser, you may be able to run the installer directly from the CD or you may have to download it to your PC first.

NOTE: If you are installing manually, click on the `sourceryg++-4.1-xx-arm-stellaris-eabi.exe` file in the “Tools/CodeSourcery” directory of the CD.

2. Follow the instructions detailed in the installer until it indicates that the installation is complete.
3. Once installation is complete, you will be required to register with CodeSourcery to obtain an evaluation license. To do so, start the Sourcery G++ IDE. You will be prompted with the Sourcery G++ License Error window.



4. Click Get License and follow the instructions on the web page to get the license.

Step 3: Install the StellarisWare™ Package

1. Navigate to the *Tools* tab on the LM3S811 Evaluation Kit Documentation and Software CD.

NOTE: If you are navigating the CD using Windows Explorer (or a similar application), go to the Tools/StellarisWare directory.

2. Click on the *StellarisWare* link and download the zip file to your PC.
3. Extract the contents of the .zip file to your hard drive. Again, the contents can be extracted to any convenient location on your hard drive since the compiler is called from the command line.
4. Click on *Stellaris® Peripheral Driver User's Guide* to download the documentation for the software.

Step 4: Building StellarisWare™ with GCC

For all examples, the UnxUtils package is used as the command-line interface. The Sourcery G++ IDE is not discussed in this document.

1. Using your command-line utility (shell), navigate to the StellarisWare directory. For this example, assume that StellarisWare™ was extracted to C:/StellarisWare.

```
MYPC# cd C:/StellarisWare
```

2. Type “make” to build the StellarisWare™ components and software examples:

```
MYPC# make
```

3. You see the compiler going through the build process for each file and build the output files (*.axf and *.bin).



Step 5: Running GDB

The GDB stub is installed with the CodeSourcery tools and is located in the “CodeSourcery/Sourcery G++/bin” directory (which is installed to “C:/Program Files” by default).

In this example, the “hello” example from “StellarisWare/ev-lm3s811/hello” is loaded into flash.

1. To run GDB, open a shell window and navigate to the “ev-lm3s811/hello/gcc” directory of StellarisWare™. Again, assume that StellarisWare™ is located at “C:/StellarisWare.”

```
MYPC# cd C:/StellarisWare/ev-lm3s811/hello/gcc
```

2. Now, type “arm-stellaris-eabi-gdb hello.axf” (this assumes that you have already built the examples as described in Step 4). After pressing Enter, you see:

```
MYPC# arm-stellaris-eabi-gdb hello.axf
GNU gdb (Sourcery G++ 4.1-xx) 6.6.50.200601124-cvs
Copyright (C) 2006 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and
you are welcome to change it and/or distribute copies of it under
certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show warranty" for
details.
This GDB was configured as "--host=i686-mingw32 --target=arm-
stellaris-eabi".
```

```
(gdb)
```

3. GDB is now running. To connect to the board, type “target extended-remote | arm-stellaris-eabi-sprite armusb:?speed=2 <target>”, where <target> is replaced with the Stellaris® device on the board. For this example, <target> is replaced with lm3s811. You see the following after pressing Enter:

```
(gdb) target extended-remote | arm-stellaris-eabi-sprite armusb:?speed=2
lm3s811
Remote debugging using | arm-stellaris-eabi-sprite armusb:?speed=2 lm3s811
```

4. To download hello.axf to the flash, simply type “load”. You see something similar to this after pressing Enter:

```
(gdb) load
Loading section .text, size 0xc10 lma 0x0
Start address 0x51d, load size 3088
Transfer rate: xxxxx bits/sec, xxx bytes/write.
```

5. Now that the image has been loaded into flash, the application can be restarted by typing “run”. When GDB asks if you would like to restart the application, type “y”.



After pressing enter, GDB restarts the application and stops at the ResetISR.

```
(gdb) run
The program being debugged has been started already.
Start it from the beginning? (y or n) y

Starting program: C:\StellarisWare\examples\gcc\hello.axf
Peripherals in SoC have been reset
Processor was reset: PC/SP loaded from 0x0.
Stopped at entry point

Program received signal SIGTRAP, Trace/breakpoint trap.
0x0000051c in ResetISR()
```

6. Type “continue” to execute the application. You see the words “Hello World!” on the LCD panel once the application begins executing.

```
(gdb) continue
Continuing.

Program exited normally.
```

Step 6: Simplifying GDB Commands

It is possible (and easy) to simplify the commands that you send GDB by defining custom commands in a .gdbinit file and placing it in your \$HOME directory. To find out where \$HOME is, type “echo \$HOME” in a shell window. For Windows XP users, that is “C:/Documents and Settings/<user name>” by default.

An example of a command to include in your .gdbinit file is “connect”. This command replaces the complex “target extended-remote | arm-stellaris-eabi-sprite -s 2 -f lmi.dll stdio” command string described in Step 5, item 3. As an example, after starting GDB, typing “connect” would have the same effect as typing “target extended-remote | arm-stellaris-eabi-sprite -s 2 -f lmi.dll stdio”.

Conclusion

You have now installed GCC, built the StellarisWare™ components, and flashed the device using GDB. To learn more about using GDB commands, type “help”. This displays a list of help topics, which can be expanded. For example, typing “help data” lists the commands available for doing data operations.



References

The following references are included on the Stellaris® LM3S811 Evaluation Kit Documentation and Software CD and are also available for download at www.luminarymicro.com:

- *Stellaris® LM3S811 Evaluation Kit User's Manual*
- StellarisWare™ Software, Order Number SW-LM3S
- *Stellaris® Peripheral Driver Library User's Guide, Order Number SW-DRL-UG*
- *Stellaris® LM3S811 Microcontroller Data Sheet*

In addition, the following website may be useful:

- CodeSourcery website at <http://www.codesourcery.com>

Support Information

For support on Luminary Micro products, contact:

support@luminarymicro.com

+1-512-279-8800, ext. 3

Copyright © 2007-2009 Luminary Micro, Inc. All rights reserved. Stellaris is a registered trademark and the Luminary Micro logo is a trademark of Luminary Micro, Inc. or its subsidiaries in the United States and other countries. ARM and Thumb are registered trademarks, and Cortex is a trademark of ARM Limited. Other names and brands may be claimed as the property of others.

Luminary Micro, Inc.
108 Wild Basin, Suite 350
Austin, TX 78746
Main: +1-512-279-8800
Fax: +1-512-279-8879
<http://www.luminarymicro.com>
support@luminarymicro.com



IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products

Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
RF/IF and ZigBee® Solutions	www.ti.com/lprf

Applications

Audio	www.ti.com/audio
Automotive	www.ti.com/automotive
Broadband	www.ti.com/broadband
Digital Control	www.ti.com/digitalcontrol
Medical	www.ti.com/medical
Military	www.ti.com/military
Optical Networking	www.ti.com/opticalnetwork
Security	www.ti.com/security
Telephony	www.ti.com/telephony
Video & Imaging	www.ti.com/video
Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2009, Texas Instruments Incorporated