RemoTI™ Win32 SimpleConsole Application

By Stig Torud

Keywords

- *RemoTI™*
- *ZigBee RF4CE*
- *RTI PC dll*
- *RemoTI Network Processor*
- *Visual Studio C++ 2008 Express Edition*
- *CC2530*

1 Introduction

One of the powerful features of RemoTI is the uniform application programming interface across different platform configuration. RemoTI basically presents the same interface if you develop your application on the CC2530 only or if you develop your application on an MCU and utilize the C2530 as a network processor.

The RemoTI software release also includes a dll to facilitate development in a Win32 PC environment. This dll presents the exact interface as on the embedded device and can therefore be use to develop and test application code before your custom hardware arrives, develop tools etc. The Target Emulator included in the RemoTI release is written using this dll.

This application note describes a simple Win32 console application exercising the dll. The main objective of the application note is to guide users through project environment setup using Visual Studio C++ 2008 Express edition and to provide a very simple target node and remote controller node application to showcase how to initialize, pair and send/receive CERC commands.

The application is kept simple and will only highlight the very basic steps needed to create a Win32 console application. Among other things, no GUI software library is used and key presses on the command window are used to synchronize events. Please refer to the sample applications in the RemoTI software release for more complete RemoTI examples.

Table of Contents

2 Abbreviations

API	Application Programmer Interface
CERC	Consumer Electronics Remote Control
DLL	Dynamically Linked Library
EM	Evaluation module
EB	Evaluation Board
RTI	RemoTI Application Interface
RF4CE	Radio Frequency for Consumer Electronics
SDK	Software Development Kit
TE	Target Emulator
UI	User Interface

3 System Setup

The Win32 SimpleConsole application communicates with a CC2530 RemoTI network processor (RNP) using a UART connection. The software included with this application note includes a sample application configuring the RNP as a remote controller (RC) node and another sample application configuring the RNP as a target node.

The Target Emulator (TE) application included with the RemoTI release can be used with the RemoTI kit target board to enable discovery/pairing and display the messages sent from the SimpleConsole application configured as an RC node. See Figure 1. The TE is connected to UART1 and the SimpleConsole application is connected to UART2. Note that the RemoTI kit only includes one target board and you need two target boards to test this setup. The SmartRF05+CC2530EM can also be configured as a RNP.

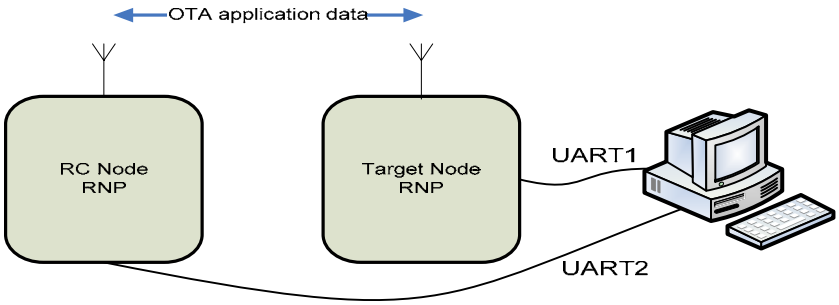


Figure 1. SimpleConsole Application configured as a RC Node

The basic RC included with the RemoTI kit can be used to trigger discovery/pairing and send CERC command to the SimpleConsole application configured as a target node. See Figure 2.

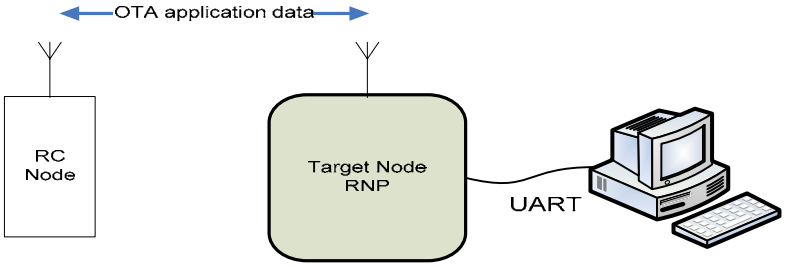


Figure 2. SimpleConsole Application Configured as a Target Node

4 Installation Requirements

4.1 Host Computer Requirements

The RTI dll was tested on a personal computer running Microsoft Windows XP Professional. The following are the minimum platform requirements for using the RTI dll.

- Visual C++ 2008 Express Edition
- Windows XP Service Pack 1
- Multiple USB ports to connect to target network processor
- RemoTI software release (www.ti.com/remoti)

4.2 Target System Requirements

The application developed using RTI dll connects to the RemoTI network processors via a UART interface. It is important that the RemoTI network processor is built with the same version of the software as the RTI dll.

- IAR EW 8051 Embedded Workbench (www.iar.com). The RemoTI software release list the IAR version requirement for the release.
- Depending on the application, an additional RemoTI target Board or SmartRF05+CC2530EM may be required.

5 Product Installation Requirements

5.1 Install RemoTI

The default installation folder for RemoTI 1.0 this is "C:\Texas Instruments\RemoTI-CC2530DK-1.0". This folder is referenced in the rest of this application note. If you install RemoTI in another folder, you must reference your specific install folder in project setup settings, configuration paths etc that are referenced later in this application note. Note that you also need to upgrade this path when installing future RemoTI releases.

5.2 Install Visual C++ 2008 Express Edition

Go to http://www.microsoft.com/express/product and follow the instruction to download and install Visual C++ 2008 Express Edition. Note that other tools, such as Visual C++ 2005 Express Edition and Visual Studio 2005/2008, will work but the configuration and setup instructions may be different from what is captured in this document.

6 Create and Configure the SimpleConsole Application

This section provides the steps necessary to configure a simple RemoTI application using the RTI dll. If you don't want to go through these steps, you can unzip the sample application included with this application note to C:\Documents and Settings\'user'\My Documents\Visual Studio 2008\Projects, and go to section 8 for instructions for how to execute the application.

6.1 Create a Win32 Console Application

1. From the Visual C++ Express Edition main menu choose “File – New Project”
   a. Select Win32 project type and Win32 Console Application template
   b. Enter name and path of the project. See Figure 3.

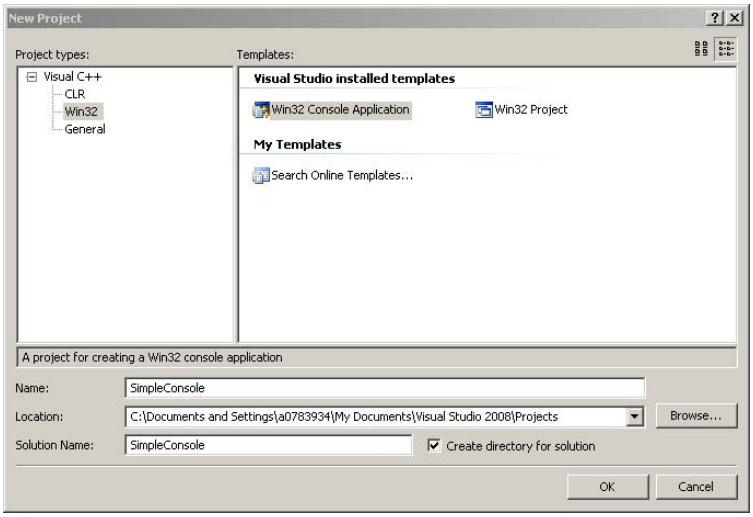


Figure 3. VC++ Express New Project Wizard

2. Click ‘Application Settings’ in the left hand column, and choose empty project in the application wizard. See Figure 4.

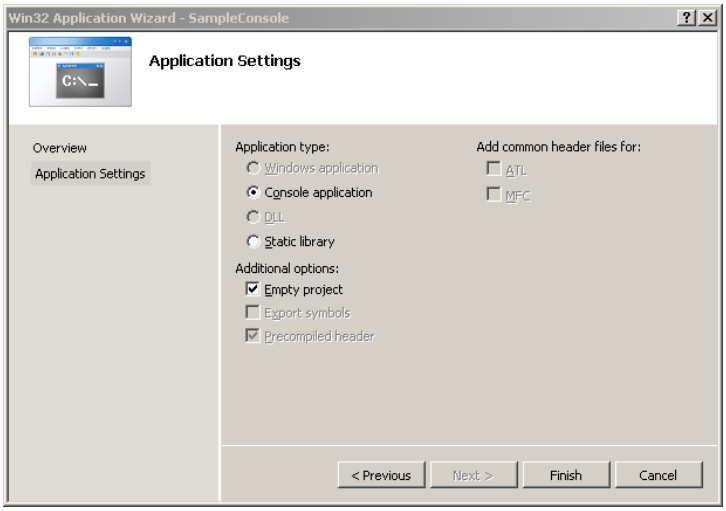


Figure 4. VC++ Express Console Application Setting

6.2 Configure Project settings

1. Right click on the “Source Files” folder in the solution explorer and select “Add – New Item”. Select “code” and “C++ File (.cpp)” in the wizard, name the file sample_main and click the add button. See Figure 5.

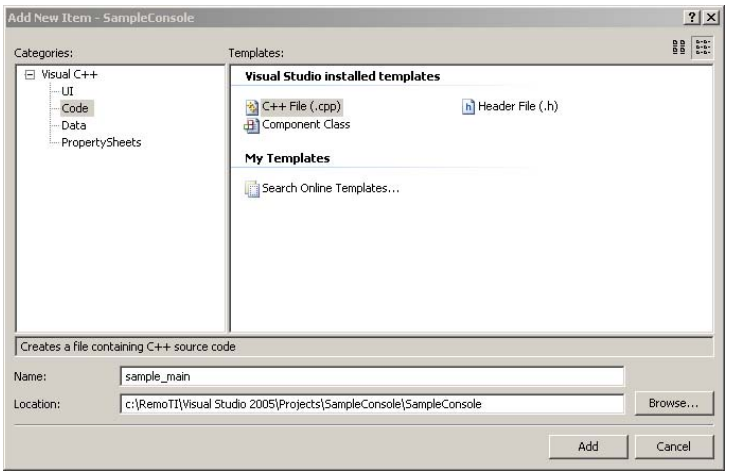


Figure 5. VC++ Express Add New Source File Wizard

2. Right-click on the project name (in bold) in the solution explorer and select properties.

   a. Go to C/C++ and the general category and add “C:\Texas Instruments\RemoTI-CC2530DK-1.0\Tools\Include” to Additional Include Directories. See Figure 6.

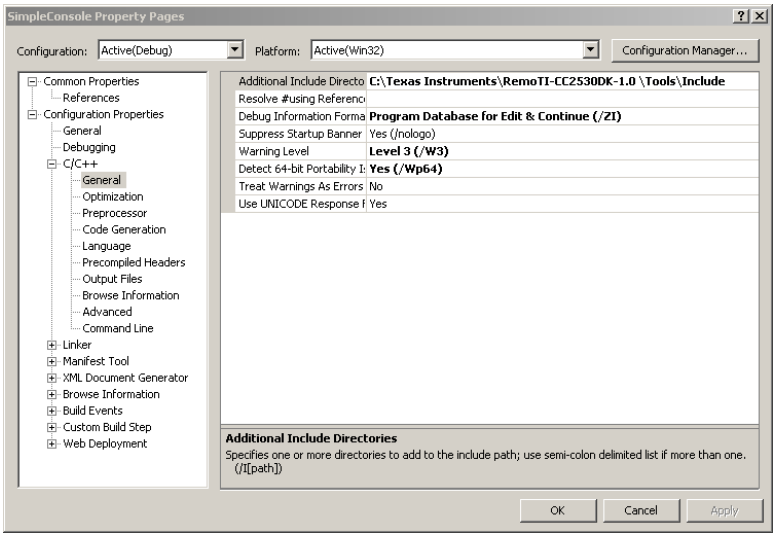


Figure 6. VC++ Express Include Directory Configuration

   b. Go to C/C++ and the Code Generation category and set Runtime Library to “Multi-threaded Debug (/MTd)”. See Figure 7.

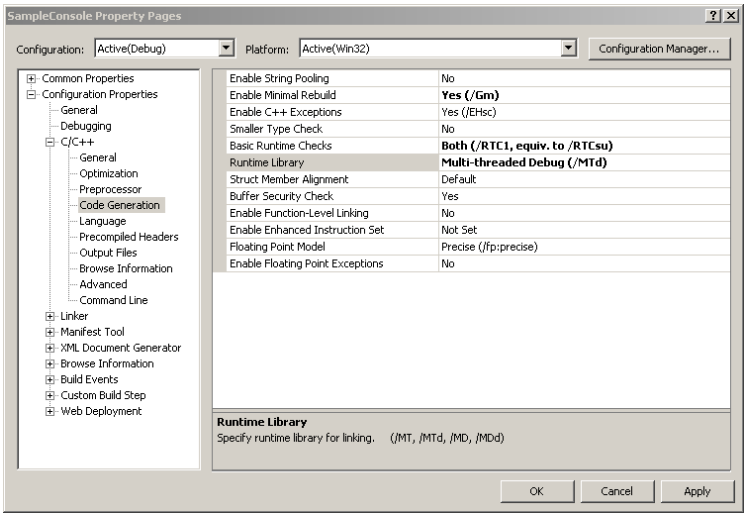


Figure 7. VC++ Express Runtime Library Configuration

c. Go to Linker and the General category and add “C:\Texas Instruments\RemoTI-CC2530DK-1.0\Tools” to Additional Library Directories. See Figure 8.

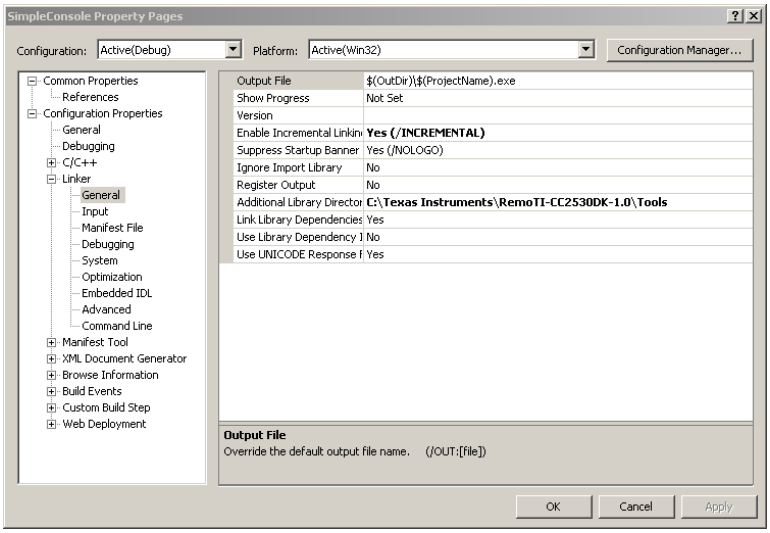


Figure 8. VC++ Express Additional Linker Directories Configuration

d. Go to Linker and the Input category and add “rtilibinit.lib rtilib.lib” (with a space between the two libraries” to Additional Dependencies. See Figure 9.

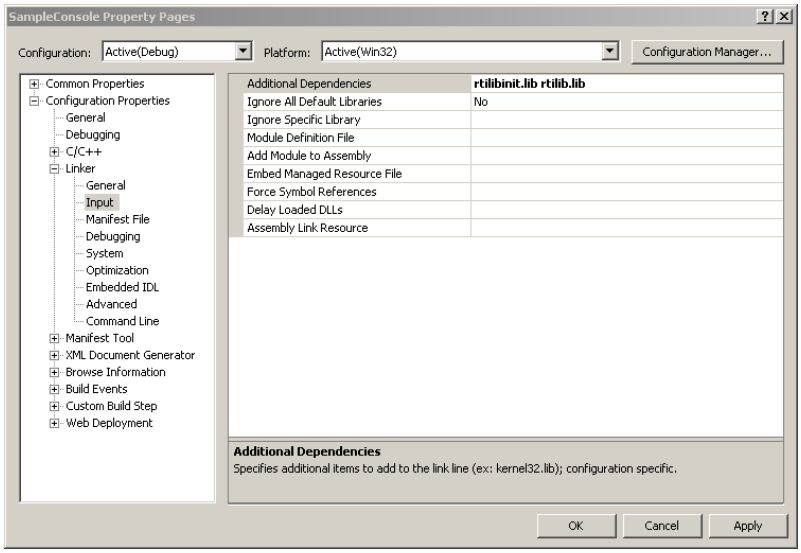


Figure 9. VC++ Express Additional Dependencies Configuration

e. Go to Debugging and the Environment category and add "PATH=%PATH%;c:\Texas Instruments\RemoTI-CC2530DK-1.0\Tools". See Figure 10.

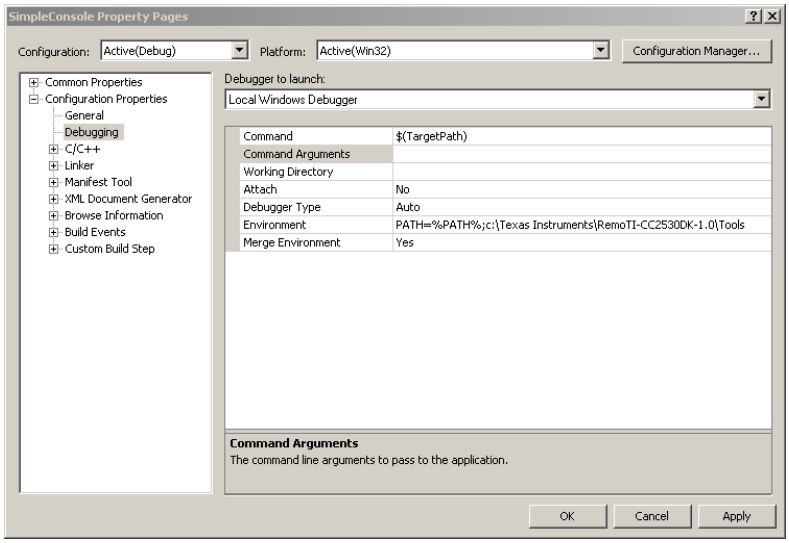


Figure 10. VC++ Express Environment Path Configuration

7 Application Code

We created the file sample_main.c during the configuration process in section 6.2 step 1. This file is still empty but we needed this file as a placeholder in order to complete all the project configurations. The sample code included with this application note contains two main files:

1. sample_main_rc.c. This file contains the application code for running the SimpleConsole application as a RC node.
2. sample_main_target.c. This file contains the application code for running the SimpleConsole application as a target node.

You can copy the content of either of the files above into sample_main.c, but a better approach is to delete the sample_main.c and include both files mentioned above. This way you can include/exclude either of the files for the desired functionality. You can include/exclude a file with a right-click on the file, choose properties and select 'yes' or 'no' in the General -> Exclude from build section as seen in Figure 11.

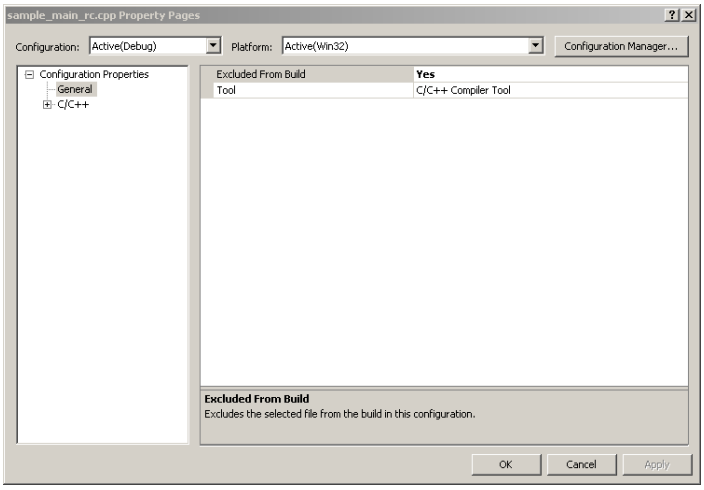


Figure 11. VC++ Express File Exclude from Build Configuration

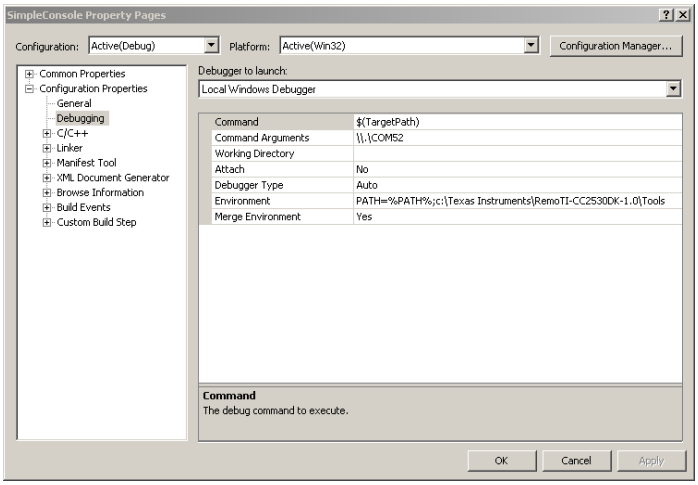
Open up sample_main_rc.cpp or sample_main_target.cpp and notice the RTI APIs called are exactly the same as if the code was developed for the CC2530 embedded IAR project. See [1] for more details. However, note that RTI_InitWin32Module() and RTI_CloseWin32Module() are special APIs included in rtilib.dll. These APIs will open and close the connection to the COM port used to communicate with the RNP.

8 Execute the SimpleConsole Application

8.1 SimpleConsole Application as RC Node

1. Build the network processor project in C:\Texas Instruments\RemoTI-CC2530DK-1.0\Projects\RemoTI\RNP and load the executable onto the target board. Note that you need to set the correct compiler option depending on the platform you want to use for the RNP for the SimpleConsole application.
2. Connect the mini USB (or RS232) cable between the PC and the target board and go to the PC device manger and note the specific COM port designated for this connection
3. Make sure that sample_main_rc.cpp is included and sample_main_target.cpp is excluded from the project

4. Right-click on the project name (in bold) in the solution explorer and select properties
    a. Go to the Configuration properties and debugging category and enter the COM port number from step 2 in the Command Arguments field. Note that port 1-4 is entered as COMX and port numbers >4 are entered as \\.\COMXX. See example below for COM46.



5. Hit the green arrow to start run the application and follow the instruction on the command line to pair and send a CERC command over the air. Note that you will need to configure the target emulator on a separate COM port and click the Allow Pair button to enable pairing. After pairing is complete, hit '0'-'9' to send the corresponding CERC command to the emulator and 'q' to quit the application. When pressing key '0'-'9', observe the corresponding key highlighted the target emulator.

8.2 SimpleConsole Application as Target Node

1. Build the network processor project in C:\Texas Instruments\RemoTI-CC2530DK-1.0\Projects\RemoTI\RNP and load the executable onto the target board.
2. Connect the mini USB cable between the PC and the target board and go to the PC device manger and note the specific COM port designated for this connection.
3. Make sure that sample_main_target.cpp is included and sample_main_rc.cpp is excluded from the project
4. Configure the COM port as described in step 4 of section 8.1.
5. Hit the green arrow to start run the application and follow the instruction on the command line to allow pairing and receive CERC command over the air from a RC. After pairing is complete, the CERC command codes will be displayed on the command line. The CERC commands will not be parsed since this is beyond the scope of this application note.

9 Conclusion

This application note has outlined the steps for correct project environment configuration using the RTI dll with Visual Studio C++ 2008 Express Edition. The sample code included with the application note also provides a very simple target node and remote controller node application. The uniform RTI application programmer interface across different platforms enables the Win32 environment for developing RemoTI application code and customer specific tools.

References

[1] RemoTI API, SWRA268

10 General Information

10.1 Document History

Revision	Date	Description/Changes
SWRA304	2009.09.03	Initial release.
SWRA304A	2009.09.09	Updated broken link to Microsoft VC++ 2008 Express Edition

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		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DLP® Products	www.dlp.com	Broadband	www.ti.com/broadband
DSP	dsp.ti.com	Digital Control	www.ti.com/digitalcontrol
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Military	www.ti.com/military
Logic	logic.ti.com	Optical Networking	www.ti.com/opticalnetwork
Power Mgmt	power.ti.com	Security	www.ti.com/security
Microcontrollers	microcontroller.ti.com	Telephony	www.ti.com/telephony
RFID	www.ti-rfid.com	Video & Imaging	www.ti.com/video
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265