TVP5147 Evaluation Module

User's Guide

Literature Number: SLEU103
September 2008

Contents

List of Figures

List of Tables

Read This First

About This Manual

This manual is intended to be a complete reference on how to use the TVP5147EVM. The manual describes how to configure and use the EVM.

How to Use This Manual

This document contains the following chapters:

- Chapter 1 - Functional Description
- Chapter 2 - Setup and Configuration
- Chapter 3 - System Operation
- Chapter 4 - Troubleshooting
- Chapter 5 - Schematics and Board Layout

FCC Warning

This equipment is intended for use in a laboratory test environment only. It generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to subpart J of part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment in other environments may cause interference with radio communications, in which case the user at his own expense will be required to take whatever measures may be required to correct this interference.

Functional Description

The TVP5147EVM evaluation module is a printed circuit board designed for demonstration of the TVP5147 video decoder. The board is designed with a 120-pin connector with all video clocks and data, which allows a connection to multiple back ends. The EVM is shipped with a professional encoder module. The board is designed to provide ease of use, while allowing full evaluation of the video decoder.

Topic **Page**

1.1 Description Overview

The TVP5147EVM uses the PC parallel port to emulate the I^2C bus, which provides communication with the TVP5147 video decoder and the video encoder. The WinVCC application software that communicates with the devices via I^2C is provided.

The analog video inputs supported by the TVP5147EVM include composite video, S-video, component video (YPbPr). More detail about the video inputs are discussed in the Section 1.2.3. In general, the video decoder converts the analog video input signal into digital component data. This data and the associated clocks from the video decoder are sent to the video backend. The video back end then converts the digital data back into analog video. The analog video outputs supported by our EVM include composite video, S-video, and component video. These are all output simultaneously

To experiment with the programmable features of the TVP5147 video decoder and the video encoder, the parallel port of the TVP5147EVM is connected to the parallel port of a PC. A Windows compatible application provides the user interface for performing register-level and high-level control of the TVP5147 video decoder and the video encoder. This application is called WinVCC (Windows Video Control Center).

1.2 Board-Level Description

The TVP5147EVM consists of the TVP5147EVM module (see Figure 1-1) and the Encoder EVM module (see Figure 1-2). A 4-row 120-pin connector connects the boards. The block diagram of the EVM set is shown in Figure 1-3.

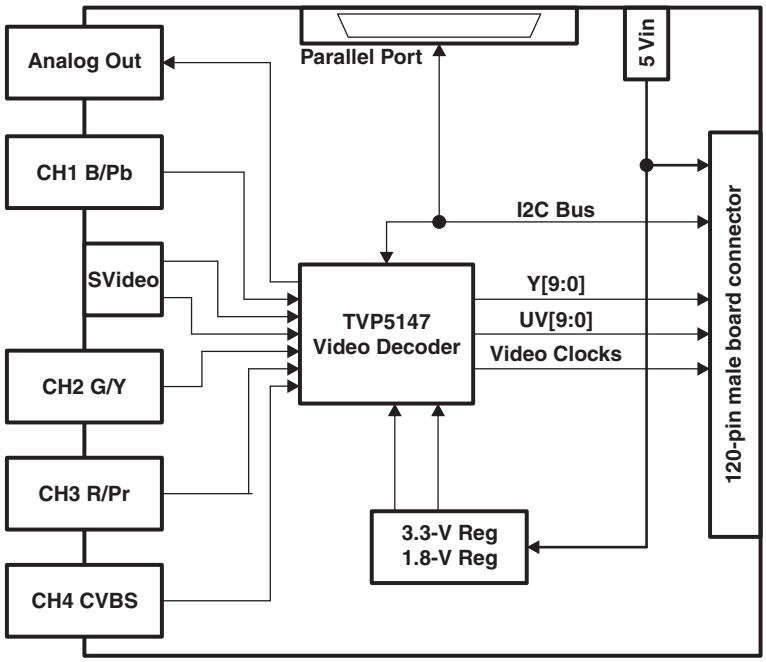

Figure 1-1. TVP5147 Module

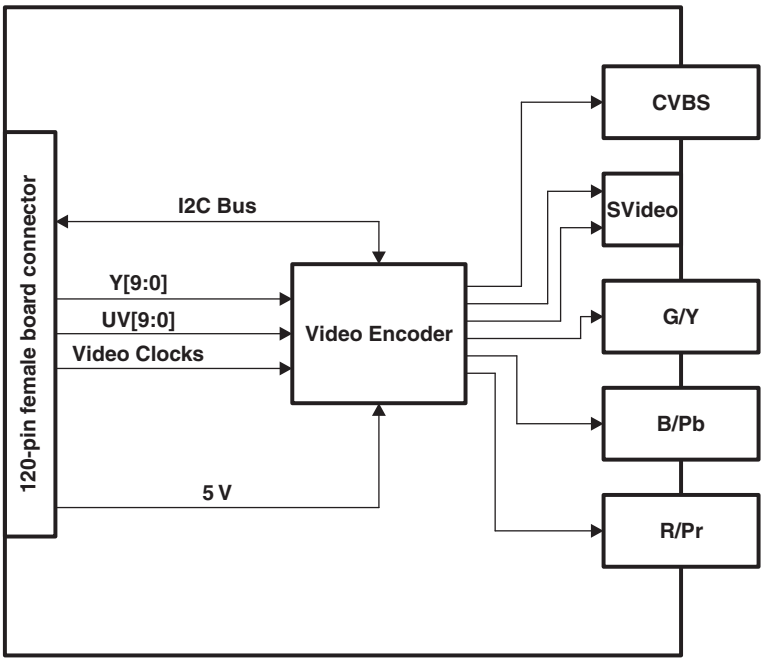

Figure 1-2. Encoder Module

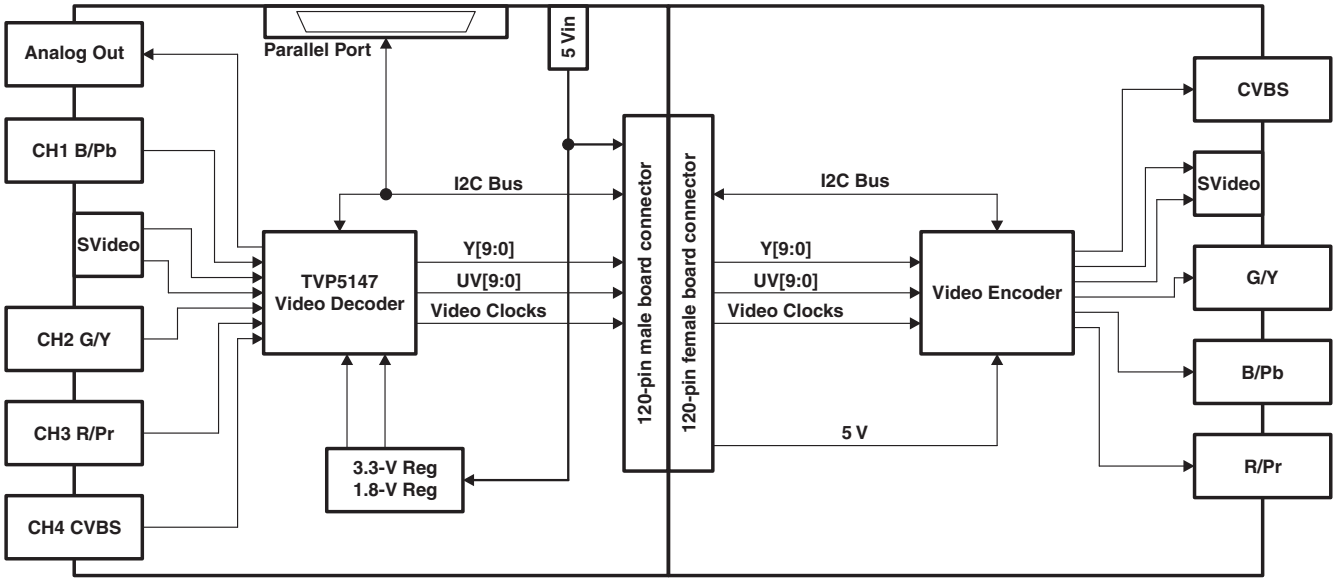

Figure 1-3. TVP5147EVM

1.2.1 Testpoints and Jumpers

The TVP5147EVM was designed with test-points and jumpers to help in evaluation and troubleshooting. There are test-points for SCL, SDA, 3.3 V, and 1.8 V. All digital video data are brought out to a dual row header which allows easy hookup to test equipment. The I^2C address selection is made with a shunt jumper, which is only read after a reset or at power up. The default address is 0xB8. If the address needs to be changed, the TVP5147 must receive a reset.

1.2.2 Decoder to Encoder Connection

The TVP5147EVM uses a four row 120-pin connector to share common signals and the 5-V power supply between the boards. This connection allows multiple backends to be connected to the TVP5147EVM. The EVM package is shipped with an encoder module. This connector shares all digital video bits (Y[9:0] and C[9:0]), all video clocks (VS, HS, GLCO, AVID, FID, and DATACLK), 5 V, ground, I^2C bus (SCL and SDA), and reset.

1.2.3 Video Input Description

The TVP5147 has four analog channels that accept up to 10 video inputs that can be configured with a combination of YPbPr, CVBS, or S-video video inputs. There are many different combinations that can be configured; however, the TVP5147EVM only has four BNC connectors and one S-video jack. The four BNC connectors allow the user to input four composite video inputs, or one composite and one component input. Each input has an anti-alias filter that can be in-circuit (default), or bypassed by jumpers (JP1, JP2, JP7-JP10). To select the filter, the shunts need to short positions 1-3 and 2-4. To bypass the filter, the shunts need to be moved to short positions 1-2 and 3-4 as shown in Figure 1-4. The boards are shipped with the filter in circuit.

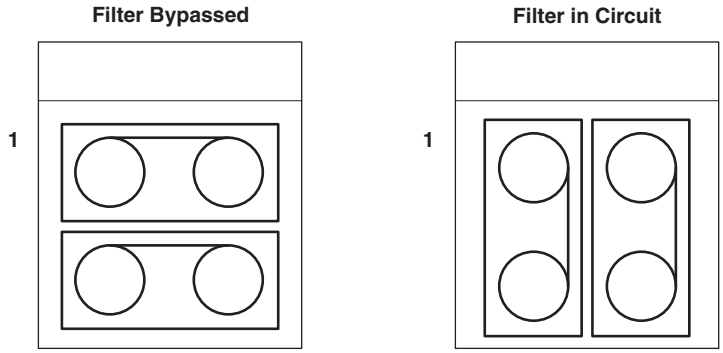

Figure 1-4. Filter Selection

1.2.4 Video Output Description

The TVP5147 digital outputs are routed to the 120-pin edge connector along with all video clocks on the TVP5147 module. The encoder module connects to the 120-pin connector and is capable of receiving digital video with or with out embedded syncs. The analog outputs of the encoder module are composite, S-video, and component. All of these outputs come out of the encoder module simultaneously for the user's convenience.

Active CVBS or Y output is available on the Analog Out connector of the Decoder board.

1.3 System-Level Description

A system-level block diagram incorporating the TVP5147 is shown in Figure 1-5. Typical test equipment is also shown. The primary features of this configuration are listed below.

- Power is provided by a single 5-V power supply provided with the EVM and is shared between both modules via the 120-pin connector.
- Supported analog inputs include composite video, S-video, and component video.
- Re-encoded composite video and S-video are output via the encoder module.
- Component (YPbPr) video is output via the encoder module.
- I^2C bus initialization of the video devices via a PC parallel port
- TVP5147 video decoder performance parameters may be measured with a video analyzer.

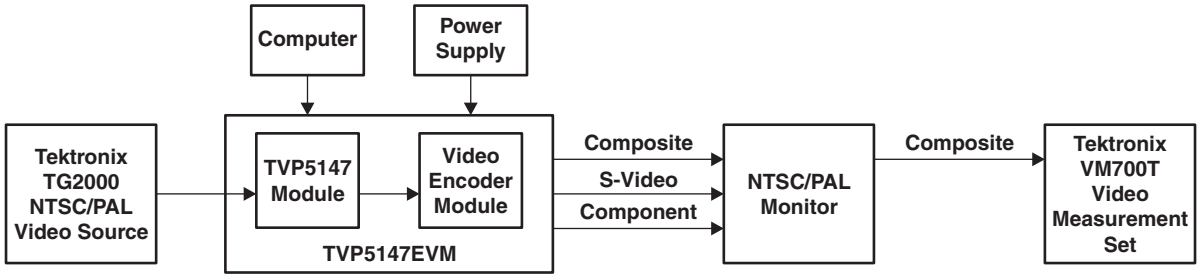

Figure 1-5. TVP5147EVM System Level Block Diagram

Setup and Configuration

The following summarizes the steps for setup and operation of the EVM. Follow the steps in the order shown.

- Software Installation (Section 2.1) or Updating the Software (Section 2.1.1)
- Hardware Configuration (Section 2.2)

Topic **Page**

2.1 Software Installation

All necessary software for the TVP5147EVM is provided on enclosed CD-ROM. Both the EVM software and the device files must be installed on the PC emulating the I^2C bus via the parallel port.

1. Insert the CD-ROM into the computer that will emulate the I^2C bus via the parallel port.
2. Run SETUP.EXE file to install WinVCC and documentation.
3. Click Next at all prompts and finally click Finish to complete the installation process.
4. A message may report that the DriverLINX PORT I/O Driver must be installed (if it was not previously installed). This driver must be installed for WinVCC to run. To install this driver, run Port95NT.exe, which is located in the root directory of the installation CD-ROM.
5. Go to Start->Programs->TVP5147EVM Software to start WinVCC and to browse documentation.
6. At the root directory of the installation CD is a zip file named Application_Reports.zip, which contains several application documents.

2.1.1 Software Update

Visit our web site at www.ti.com for the latest version of the EVM software and the device software.

2.2 Hardware Configuration

Figure 1-3 shows the TVP5147EVM layout and indicates the location of the power supply jack and the appropriate connectors. All connectors are labeled according to their function. To prepare the EVM for evaluation, connect the following:

- TVP5147 module to encoder module
- Parallel port cable from a PC to the TVP5147EVM
- Analog video in
- Analog video out
- Power supply

The system comes with the anti-alias filters connected. To bypass the filters, set the filters as shown in Section 1.2.3.

The I^2C address LSB can be set with JP4. The default setting for this jumper is for the shunt to short pins 2-3, which selects 0xB8. This is connected to TVP5147 pin 37, which is read at power up. If the jumper is moved to positions 1-2 and the board is reset, the I^2C address is 0xBA. When changing the address, exit WinVCC, restart WinVCC and select the new slave address.

System Operation

This chapter describes the system operation.

Topic **Page**

3.1 Power Up

After Software Installation (Section 2.1) and Hardware Configuration (Section 2.2) have been completed, the TVP5147EVM may be powered up.

3.1.1 Starting WinVCC

The WinVCC (Windows Video Control Center) application program must have been previously installed on the PC. Run WinVCC from the Windows Start Menu:

Start->Programs->TVP5147EVM Software->WinVCC

3.1.2 WinVCC Configuration Dialog Box

The WinVCC Configuration dialog box, as shown in Figure 3-1, should now be visible. All settings from this dialog box are stored in the Windows registry and are restored the next time the program is started. After initial installation, VID_DEC will be set to TVP5147 and VID_ENC will be set to 7194 encoder module.

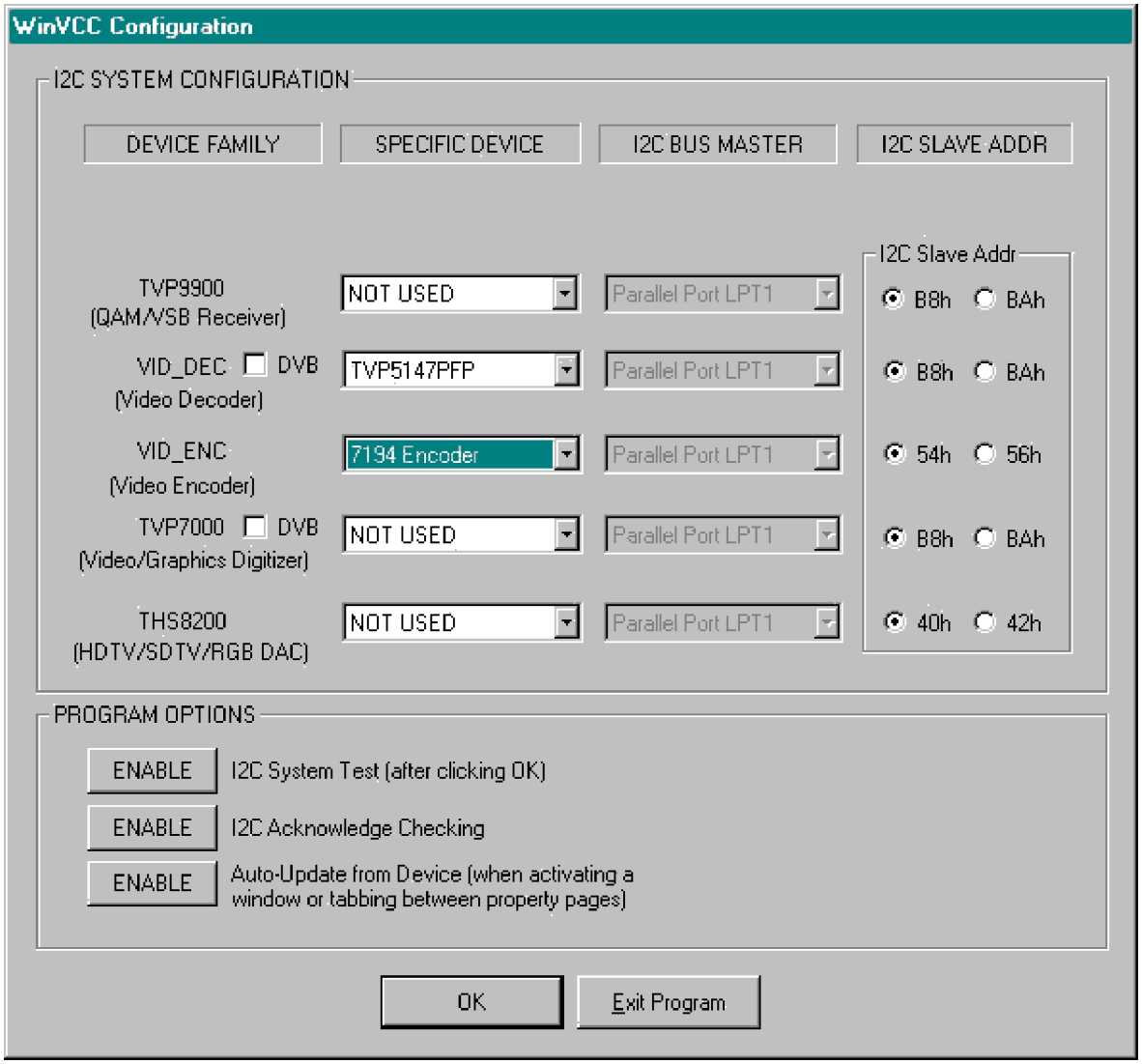

Figure 3-1. WinVCC I²C Address Configuration

The I²C slave addresses for each device must match the I²C slave address selected by jumper on the TVP5147EVM. These jumpers are set by the factory to use B8h for the video decoder and 54h for the encoder.

It is also important to select the correct specific devices. The TVP5147 and 7194 encoder module must be selected for this EVM.

The Program Options should all be enabled. Disabling these options should only be required when debugging a problem with the I²C bus itself.

Clicking OK begins I²C communication with the selected devices.

3.1.3 I²C System Test

The I²C system test of selected registers runs immediately after closing the WinVCC Configuration dialog box with OK (unless the I²C system test program options button was disabled).

If the I²C system test passes, no message is displayed. If the test failed, a dialog box similar to Figure 3-2 is displayed. See Section 4.1 on resolving I²C communication problems.

The I²C system test can be run at anytime by clicking Run System I²C Test in the Tools menu.

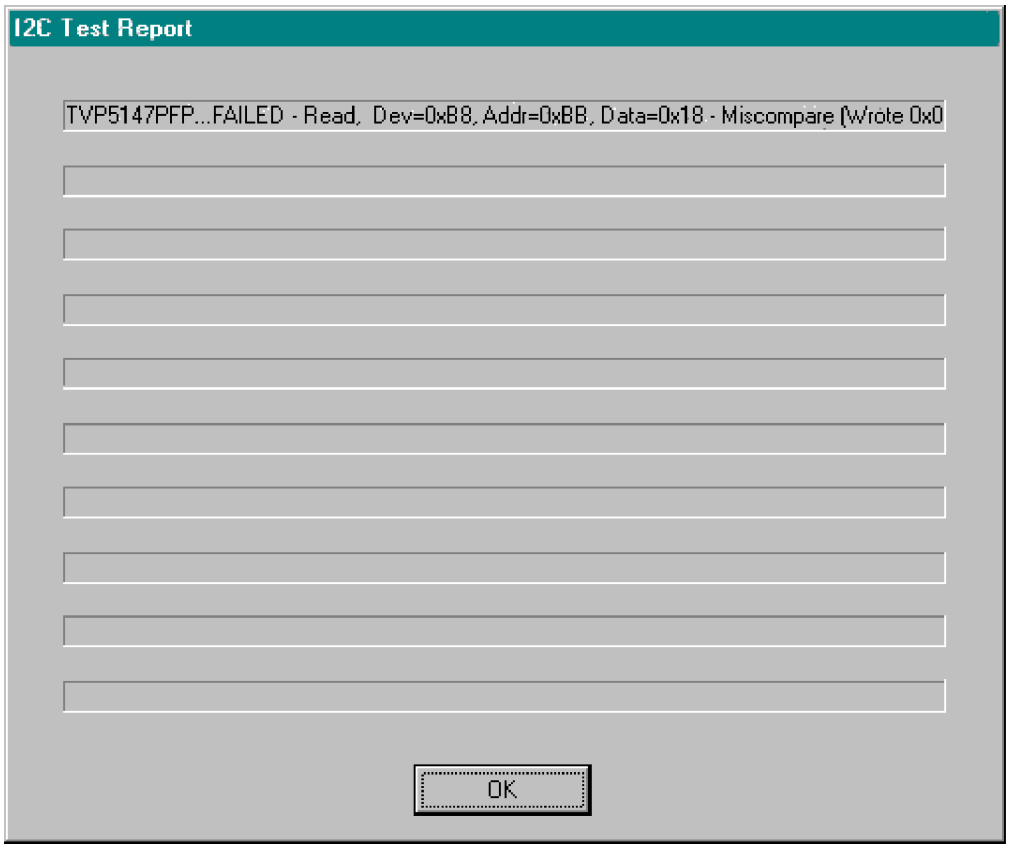

Figure 3-2. I²C System Failure

3.1.4 Real-Time Polling

Real-time polling provides polling functions that execute in the background continuously, when enabled via the Real-Time Polling dialog. There are two polling functions. The function that applies to the TVP5147 is VIDEO-STANDARD AUTO-SWITCH POLLING.

When TVP5147 detects a change in the input video standard, it automatically switches to operation in the detected standard (which includes all necessary I^2C register initialization) for proper decoding of the input video. To enable auto-switch on the TVP5147, the Set Video Standard register must be set to auto-switch mode (Reg 0x02 = 0x00).

If the WinVCC auto-switch polling function is enabled, the detected video standard status from the TVP5147 is polled until a change in the input video standard (or in the TVP5147 sampling mode) is detected. When a change is detected, the video encoder is reprogrammed as needed for the detected standard. Using this feature, the video source can change its video standard and the system will display video using the new standard without user intervention.

To enable auto-switch polling (recommended), the video-standard auto-switch polling function must be enabled in the Real-Time Polling dialog as shown in Figure 3-3. The Real-Time Polling dialog can also be accessed once WinVCC is up and running by clicking Real-Time Polling in the Tools menu.

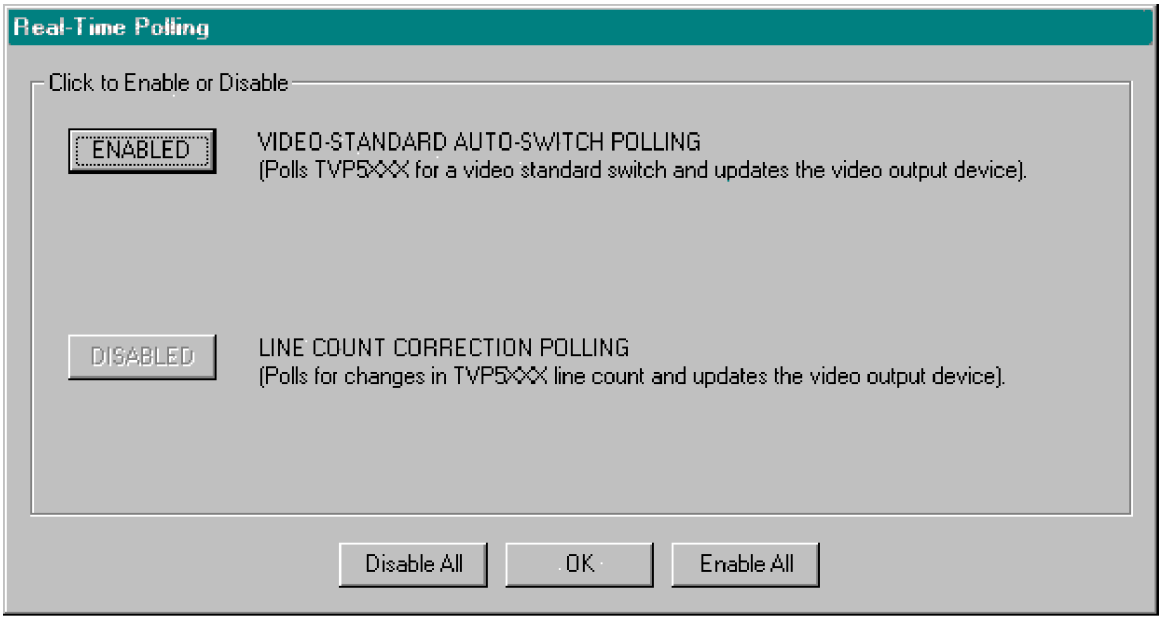

Figure 3-3. Real-Time Polling Dialog Box

3.2 Main Menu

After closing the Real-Time Polling dialog, the main menu is displayed as shown in Figure 3-4. The menus, which are used to operate WinVCC, are File, Edit, Tools, Window, and Help. The File menu's only function is Exit, which terminates the program. Table 3-1 is a summary of the main menu contents.

Figure 3-4. WinVCC Main Menu

Table 3-1. Main Menu Summary

Menu	Contents
File	Exit
Edit	Register Map Editor
	Memory Map
	Property Sheets

Table 3-1. Main Menu Summary (continued)

Menu	Contents
Tools	System Initialization
	Real-time Polling
	TV Tuner Control
	Multiple-Byte I^2C Transfers
	Set I^2C Bit Rate
	Run System I^2C Test
	Run Continuous I^2C Test
	Read VBI FIFO
	Capture Live VBI Data
Window	Initially empty. Used to jump between multiple open windows.
Help	About WinVCC

3.2.1 System Initialization

Clicking System Initialization in the Tools menu displays the dialog shown in Figure 3-5. This provides the means for initializing the video decoder and/or video encoder for a particular video mode. The details of the initialization are contained in the command file (with a CMD file extension).

The command file is opened using the Browse... button. Once the command file is opened, a text list displays descriptions of the individual data sets contained within the command file.

Click once on the desired data set description to select it. Click the Program Device(s) Using Selected Dataset button to run the selected data set, which loads the devices via the I^2C bus. When the device initialization has completed, the status indicator reads Ready.

Note: If Ready does not display, then the devices are not initialized and the I^2C bus is not communicating. See Chapter 4: *Troubleshooting* for a solution.

Click the Close button to close the dialog box. Each time the System Initialization dialog is closed, the initialization file pathname and the data set selection number are saved in the Windows registry to allow these settings to be retained for the next time WinVCC runs.

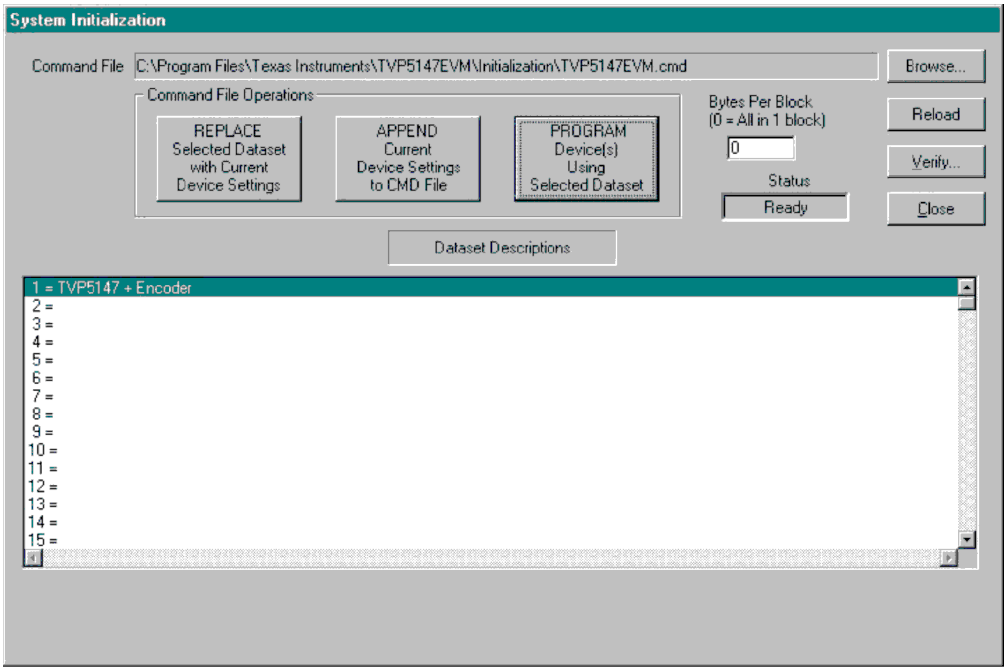

Figure 3-5. System Initialization

3.2.2 Adding a Custom Data Set

After programming the EVM via the System Initialization tool using the factory-supplied command file, device register settings can be customized to suit the application. This can be done using a Graphical User Interface (Edit > Property Sheets) or by editing individual registers (Edit > Register Map). The customized register settings can then be saved by appending them to the currently selected command file as follows:

1. Reopen the System Initialization dialog via the Tools menu.
2. Click the Append Current Device Settings to Command File button. A dialog requesting a description of the new data set is displayed
3. Optionally, click the drop-down box and select one of the existing descriptions.
4. Modify the description text or type another description.
5. Click OK. All nondefault register values from the decoder and encoder are appended to the current command file as an additional data set.

Now, select the custom data set and send it with a press of the Program... button.

Note: 1.The command file (.CMD) must be saved as plain text.

2. Do not double click on these .cmd files in Windows Explorer. Use right click > Edit to open a .cmd file.

3.2.3 Command Files

The command file is a text file that can be generated using any common editor; however, it must be saved as plain text. Command files are especially useful for quickly switching between the various system configurations. These .CMD files are unrelated to the typical Windows .CMD files.

A default command file has been installed from the CD-ROM. This command file should contain most of the desired setups. This command file is located at:

C:\Program Files\Texas Instruments\TVP5147EVM\Initialization\TVP5147EVM.cmd

A command file can contain up to 250 data sets. A data set is a set of register settings to initialize the video decoder and/or video encoder for a particular video mode. Each data set includes a description that is displayed in one row of the dataset descriptions list. The register settings may be located in the command file itself and/or may be stored in separate include file(s) (with an INC file extension) and be included into the command file using the INCLUDE statement.

3.2.3.1 Example Command File

An example of one data set within a command file is shown in Figure 3-6. Each command file may contain individual write to register (WR_REG) commands or include these commands from a separate INC file or both. The purpose of the INCLUDE feature is to avoid repeating a long list of register settings many times in a command file and having to maintain that list.

```
BEGIN_DATASET
DATASET_NAME,"TVP5147 Auto-switch, CVBS CH4A, 656 Out"
// Initialize video encoder using an include file
INCLUDE, EncoderNTSC656_RTC.INC
// Program TVP5147 registers
WR_REG,VID_DEC,0x01,0x00,0x0C // Input: CVBS CH4
WR_REG,VID_DEC,0x01,0x02,0x00 // Auto-switch mode
WR_REG,VID_DEC,0x01,0x04,0x3F // Auto-switch MASK
WR_REG,VID_DEC,0x01,0x06,0x40 // Pedestal off
WR_REG,VID_DEC,0x01,0x31,0x05 // RTC mode
WR_REG,VID_DEC,0x01,0x33,0x40 // ITU 601 (extended range)
WR_REG,VID_DEC,0x01,0x34,0x11 // OUTPUTS ACTIVE
WR_REG,VID_DEC,0x01,0x35,0xFA // GLCO and FID ACTIVE
WR_REG,VID_DEC,0x01,0x36,0xAF // HS and VS ACTIVE
END_DATASET
```

Figure 3-6. Command File Example

3.2.3.2 Command File Syntax

1. The comment indicator is the double-slash //.
2. The command file is not case-sensitive and ignores all white-space characters.
3. All numbers can be entered as hexadecimal (beginning with 0x) or as decimal.
4. Every data set in a command file begins with BEGIN_DATASET and ends with END_DATASET. The maximum number of datasets is 250.
5. The dataset text description is entered between double quotes using the DATASET_NAME command. The enclosed text can be up to 128 characters in length. This text is displayed in the System Initialization dialog when the command file is opened.
6. The INCLUDE command inserts the contents of an include file (with an INC file extension) in-line in place of the INCLUDE command. Therefore, the include file should not contain the BEGIN_DATASET, END_DATASET, and DATASET_NAME commands.

   Note: All included files must be located in the same directory as the command (CMD) file.

7. The write to register command is written as follows:

   WR_REG, <DeviceFamily>, <Number of data bytes (N)>, <sub-address>, <Data1>,<Data2>...<DataN>

   or

   WR_REG,<Literal slave address>,<Number of data bytes (N)>,<sub-address>, <Data1>,<Data2>...<DataN>

   The valid device family mnemonics for this EVM are:

   VID_DEC for the video decoders

   VID_ENC for the video encoders

   WinVCC translates the device family mnemonic to the slave address that was selected on the WinVCC

Configuration dialog upon program startup. This eliminates having to edit command files if the alternate slave address must be used.

If the literal slave address method is used, the slave address entered will be used directly. This method is normally used for programming the video encoder.

8. A delay may be inserted between commands using the WAIT command, which is written as follows:
   WAIT,<# milliseconds>

3.2.4 Register Editing

The next sections describe the four available modes of register editing: Register Map Editor, Encoder Module Editor, Memory Editor, and Property Sheets. Each of these functions can be selected from the Edit menu.

3.2.4.1 Register Map Editor

The register map editor, as shown in Figure 3-7, allows the display and editing of the entire used register space of the device within a simple scrolling text box. To open this, click on Edit Register Map in the Edit menu and click on the device type to edit.

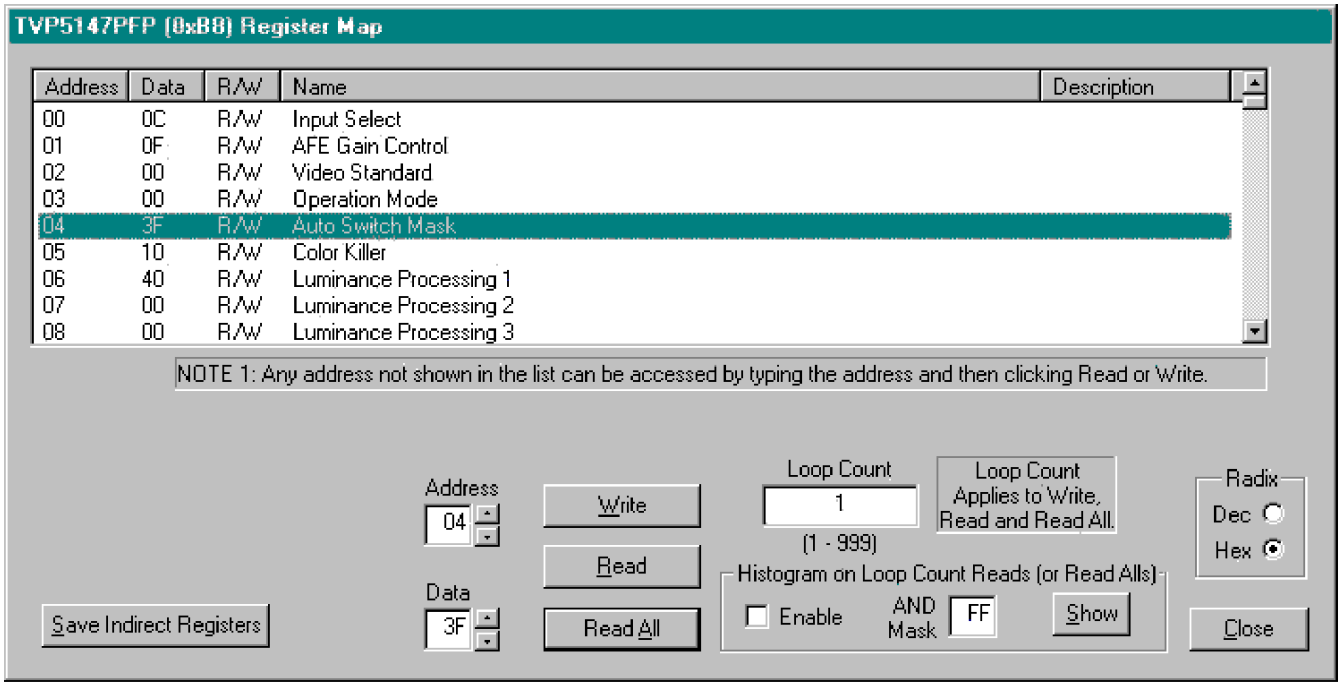

Figure 3-7. Register Map Editor

Table 3-2. Register Map Editor Controls

Control	Definition
Register Window	Scrolling text box that displays the address and data for the I^2C registers that are defined for the device.
Address Edit Box	This contains the I^2C sub-address that will be accessed using the Write and Read buttons. Clicking on a row selects an address, which is then displayed in the address edit box. **NOTE:** After clicking on a row, the Data Edit box contains the data that was in the register window. The device has not yet been read. The address up/down arrows are used to jump to the next/previous sub-address that is defined for the device. If an address is not defined for the device, it can still be accessed by typing the sub-address in the Address Edit box.
Data Edit Box	This contains the data which will be written to or was read from the I^2C sub-address. The data up/down arrows increment/decrement the data value by 1.

Table 3-2. Register Map Editor Controls (continued)

Control	Definition
Write Button	Writes the byte in the Data Edit box to the address in the Address Edit box. The I^2C register is written to whether or not the data is different from the last time the register was read.
Read Button	Reads the data from the address in the Address Edit box into the Data Edit box and the register window.
Read All Button	Reads all defined readable registers from the device and updates the register window.
Hex Button	Converts all values in the register window and address and data edit boxes to hexadecimal.
Dec Button	Converts all values in the register window and address and data edit boxes to decimal.
Close Button	Closes the dialog. **NOTE:** Multiple edit register map windows can be open at the same time (one for each device). Use the Window menu to navigate.
Loop Count	Causes subsequent write or read all operations to be performed N times. N is entered as a decimal number from 1 to 999.
Save Indirect Registers Button	Allows a range of indirect register space(device memory) to be saved to a text file.

3.2.4.2 Encoder Module Editor

The encoder module editor, as shown in Figure 3-8, allows the display and editing of the video encoder registers. This editor works like the register map editor.

To open this, click on Edit Register Map in the Edit menu and click on 7194 Encoder Module.

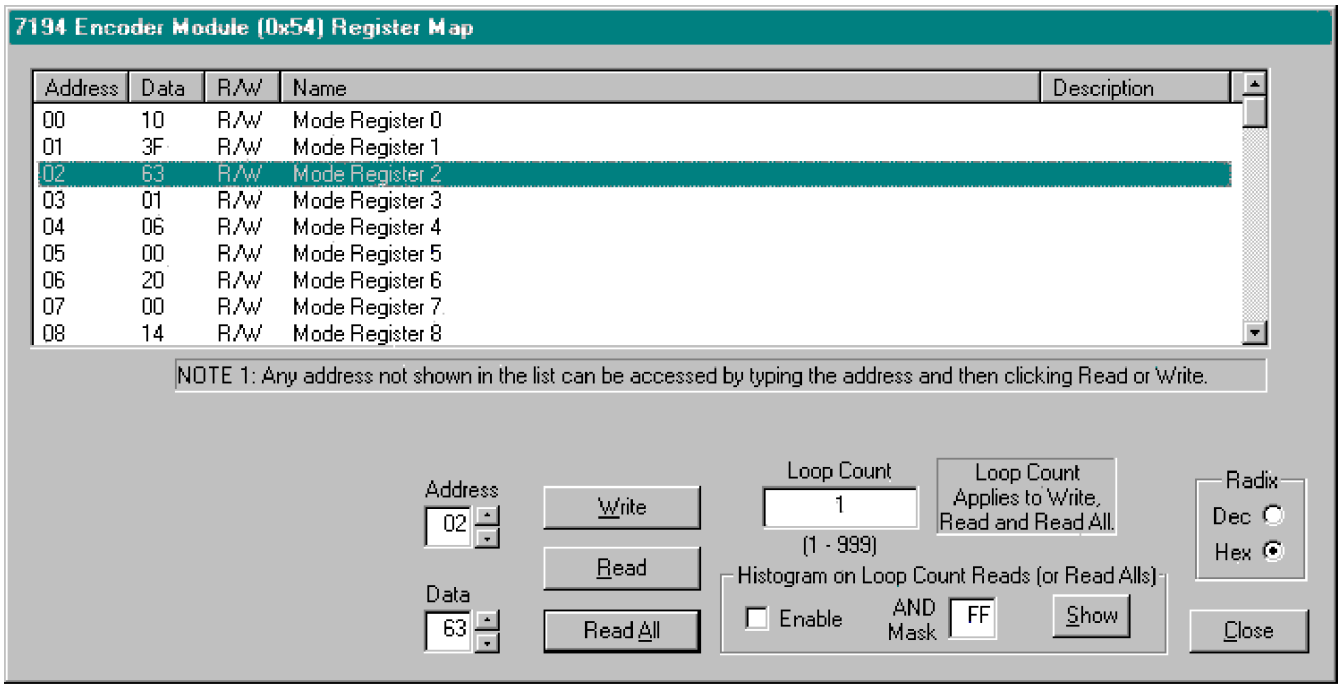

Figure 3-8. Encoder Module Editor

3.2.4.3 Memory Editor

The memory editor, as shown in Fig 3-9 allows display and editing of the indirect register space (memory map).

To open this, click on the "Edit > Edit Memory > TVP5147" menu item. The operation of the memory editor is described in Table 3-3.

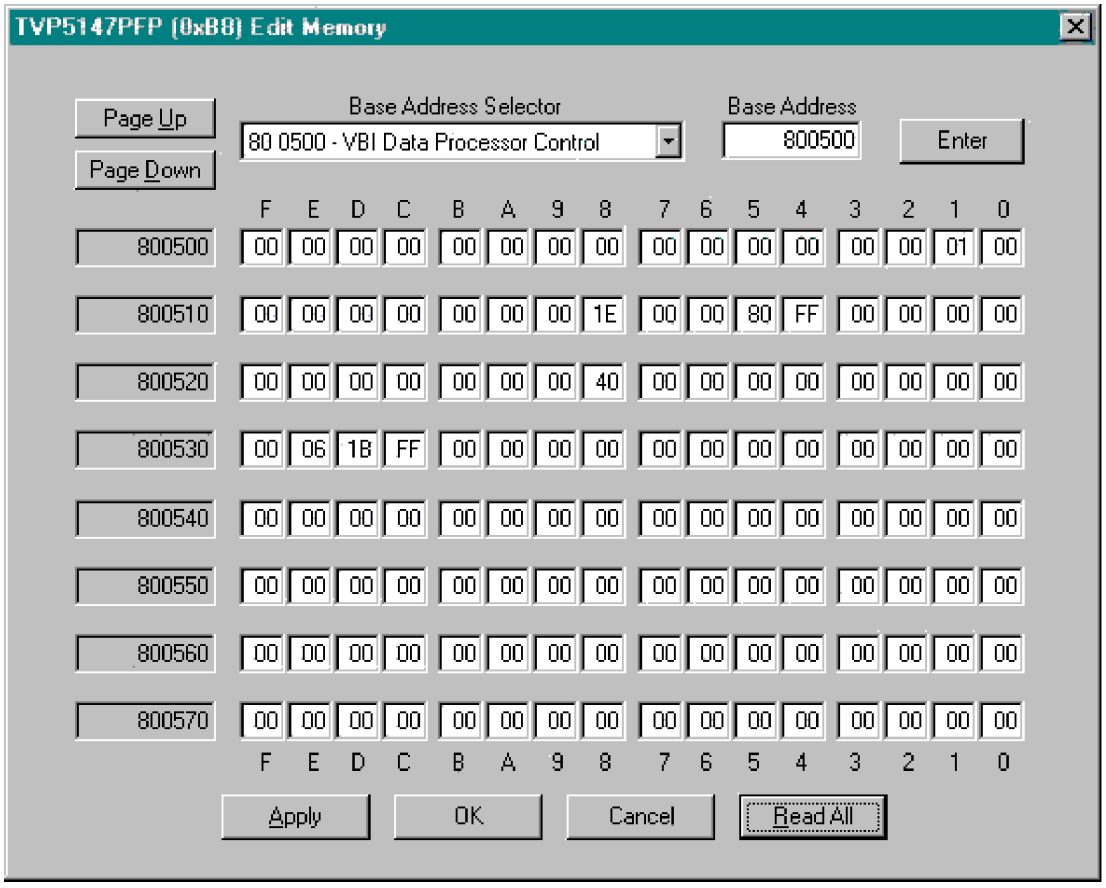

Figure 3-9. Memory Map Editor

Table 3-3. Memory Editor Controls

Control	Definition
Base Address Selector	Indirect addresses are 24 bits in length. The base address selector drop-down box allows quick selection of the base address for various functional blocks within the device. When a base address is selected it gets copied to the Base Address edit box and the list of addresses in the left-most column gets updated.
Base Address Edit Box	This contains the first indirect address that will be read or written to. Any address can be typed into this box, but it will be truncated to a multiple of 16 (10h) when the enter button is clicked.
Enter Button	This takes the address in the Base Address Edit Box and updates the list of addresses in the left-most column. At this point the memory has not been read.
Page Up	Decreases the base address by 128 (80h). Reads from device memory starting at the new base address and fills the 128 data edit boxes.
Page Down	Increases the base address by 128 (80h). Reads from device memory starting at the new base address and fills the 128 data edit boxes.
Data Edit Boxes	The memory data is displayed in a little endian format with the least significant byte (the first byte read) in the far right column. Values must be entered as hexadecimal values from 0 to FF.
Apply Button	Writes the entire 128 bytes shown in the data edit boxes to the device memory. Locations that were edited will get updated with the new values in the device.
OK Button	This is the same as the Apply button except that it also closes the dialog box.
Cancel Button	This closes the dialog box without applying edits to device memory.
Read All	Reads from device memory starting at the current base address and fills the 128 data edit boxes. Note that Read All is done automatically when you activate a different window and then re-activate the Edit Memory window.

3.2.5 Property Sheets

The property sheets represent the register data using a graphical user interface. The data is organized by function. Tabs across the top are used to navigate between pages.

To open this, click on Edit Property Sheets in the Edit menu and select the device type to edit.

When the property sheet function is started or whenever tabbing between pages, all readable registers in the device are read from hardware to initialize the dialog pages. Values on the page are changed by manipulating the various dialog controls as described in Table 3-4.

There are OK, Cancel, and Apply buttons at the bottom of each property page. These are explained in Table 3-5.

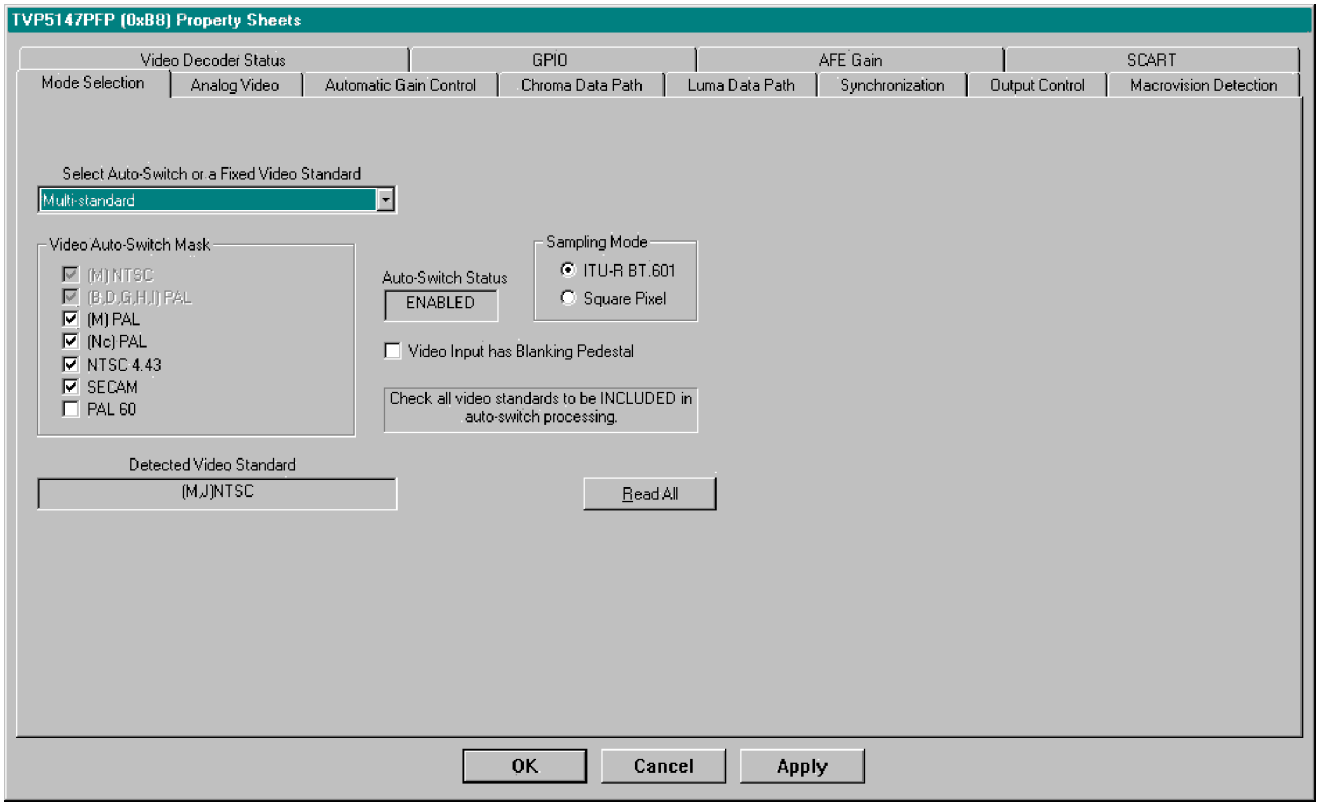

Figure 3-10. Property Sheets

Table 3-4. Property Sheet Controls

Type of Control	Function/How to Set a Value	When is Hardware Updated?
Read-Only Edit Box	Read status information	N/A
Check Box	Toggle a single bit	After Apply
Drop-Down List	Select from a text list	After Apply
Edit Box	Type a number	After Apply
Edit Box with Up/Down Arrows	Use up/down arrows or type a number	Up/Down Arrows: Immediately Type a number: After Apply
Slider	Slide a lever	Immediately
Pushbutton	Initiate an action	Immediately

Table 3-5. Property Sheet Button Controls

Button	Definition
OK	Writes to all writeable registers whose data has changed. A register is flagged as changed if the value to be written is different from the value last read from that address. Closes the dialog.
Cancel	Causes all changes made to the property page since the last Apply to be discarded. Changes made to dialog controls with 'immediate hardware update' are not discarded, since they have already been changed in hardware. Does not write to hardware. Closes the dialog.
Apply	Writes to all writeable registers whose data has changed. A register is flagged as changed if the value to be written is different from the value last read from that address. Does not close the dialog.

3.2.5.1 Reading the Register Map

The property sheets were designed so that the data displayed is always current. Certain actions cause the entire register map to be read from the device and to update the property sheets. This happens when:

- Property sheets are initially opened.
- When tabbing from one page to another.
- When Read All is clicked.
- When making the Property Sheets window the active window (by clicking on it).
- When making a Register Map Editor window the active window (by clicking on it).

3.2.5.2 Auto-Update From Device

The last two items in the previous list are referred to as the Auto-Update feature. Auto-Update can be disabled by setting its program option button to DISABLED. This button is located on the initial dialog box (WinVCC configuration).

With Auto-Update enabled (default), the user can open both the Property Sheets and the Register Map Editor at the same time. Changes made to the Property Sheets (and applied) are updated in the register map window as soon as the Register Map window is clicked on. It also works the other way; changes made in the Register Map Editor are updated in the Property Sheets as soon as the Property Sheets window is clicked on.

Troubleshooting

This chapter discusses ways to troubleshoot the TVP5147EVM.

Topic **Page**

4.1 Troubleshooting Guide

See Table 4-1 for available solutions to problems with the TVP5147EVM hardware or the WinVCC software.

Table 4-1. Troubleshooting

Symptom	Cause	Solution
At startup, the error message Cannot find DLL file DLPORTIO.DLL is displayed.	The parallel port driver supplied with the EVM has not been installed.	Run Port95NT.EXE on the CD ROM to install the driver.
Blank screen	Wrong video input is selected.	Go to Edit->Property Sheets->TVP5147, Analog Input page, select the correct video input(s) and click Apply. (The composite video input 4A is selected by default.)
	Source is connected to the wrong input connector.	Connect source to the correct input connector.
	YCbCr outputs or clock output is disabled.	Go to Edit->Property Sheets->TVP5147, Output page, check the Enable YCbCr Outputs and Enable Clock Outputs check boxes and click Apply.
No color	Wrong mode selected for color subcarrier genlock output.	Go to Edit->Property Sheets->TVP5147, Synchronization page, set the Fsc control format to RTC and click Apply.
	GLCO pin is not set to output the GLCO signal.	Go to Edit->Property Sheets->TVP5147, GPIO page, set the drop down box labeled GLCO/I2CA Terminal to Genlock Output, and click Apply.
Screen colors are only magenta and green.	Wrong YCbCr output format.	Go to Edit->Property Sheets->TVP5147, Output page, set the YCbCr output format to 10-bit 4:2:2, 2X pixel rate, ITU-R BT.656 embedded syncs mode and click Apply.
Video standard auto-switch does not work on the video decoder side.	Auto-switch masks are not set correctly.	Go to Edit->Property Sheets->TVP5147, Mode Selection page, check all standards to be included in auto-switch processing and click Apply.
	Video decoder is not in auto-switch mode.	Go to Edit->Property Sheets->TVP5147, Mode Selection page, set the drop-down box to Multi-Standard and click Apply.
Video standard auto-switch does not work on the video encoder side.	Auto-switch polling is not enabled.	Click Real-Time Polling in the Tools menu. Click Enable All and OK.
No I^2C communication	I^2C slave address is wrong.	Close and restart WinVCC. Choose the alternate slave address in the WinVCC Configuration dialog.
	Device is in power-down mode.	Press the reset button on the TVP5147EVM.
	EVM is not powered on.	The power supply must be plugged into a 100 V - 240 V/47-Hz - 63-Hz power source and the cord must be plugged into the power connector on the EVM.
	Parallel cable is not connected from the PC parallel port to the EVM DB25 connector.	Connect cable.
	Wrong type of parallel cable.	Some parallel cables are not wired straight through pin-for-pin. Use the parallel cable supplied with the EVM.
	"Undock Computer" was clicked on laptop in a docking station.	Install laptop in a docking station and restart the PC.
	Still no I^2C communication	The PC may not be capable of operating in the required parallel port mode. This is true of some laptop computers. Use a different computer, preferably a desktop PC.
	PC parallel port mode is not set correctly	Reboot PC, enter BIOS setup program, set parallel port LPT1 mode (Addr 378h) to ECP mode or bidirectional mode (sometimes called PS/2 mode or byte mode). If already set to one of these two modes, switch to the other setting (see Section 4.2.1).

When WinVCC is started and the WinVCC Configuration dialog box is closed with OK, the I^2C system test is performed (unless the I^2C System Test program options button was disabled).

If the I^2C System Test fails, a message similar to Figure 4-1 is displayed. This example reports that a read from TVP5147 failed, using slave address 0xB8, sub-address 0xBB. The data read was 0x18. The expected data was 0x01.

After noting which device had a problem, click OK to continue. Next, the Corrective Action dialog box (see Figure 4-2) is displayed to help fix the problem.

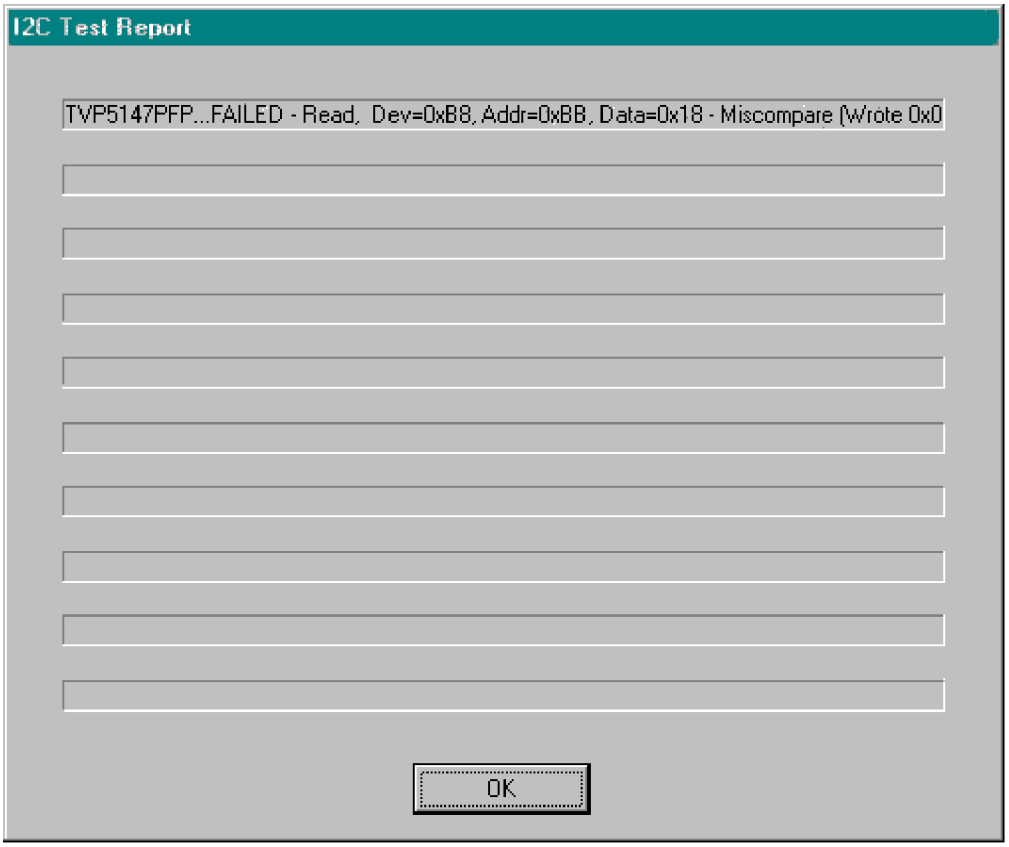

Figure 4-1. I^2C System Failure Dialog Box

4.2 Corrective Action Dialogs

After closing the I²C system test report dialog box, the dialog in Figure 4-2 is displayed.

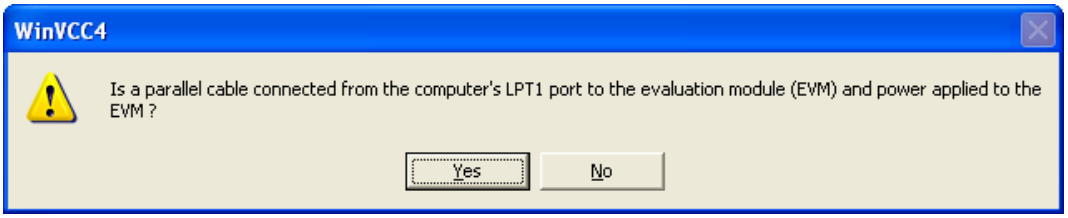

Figure 4-2. Corrective Action Dialog Box

1. If the cable is NOT connected from the PC parallel port to the TVP5147EVM or if the EVM power is not on:
   a. Click No.
   b. The dialog shown in Figure 4-3 provides instructions on how to correct the problem.

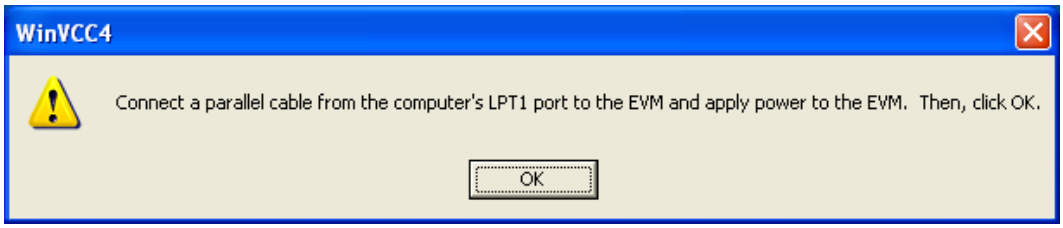

Figure 4-3. Corrective Action Required

   c. Correct the problem.
   d. Click OK to continue. The real-time polling dialog is displayed (see Section 3.2.3).
2. If the cable IS connected from the PC parallel port to the TVP5147EVM AND the EVM power IS on:
   a. Click Yes.
   b. The dialog shown in Figure 4-4 is displayed because the I2C communication problem does not have an obvious cause. Click OK to continue.
   c. The Real-Time Polling dialog is displayed. Click OK to close it and return to the main menu.
   d. Click the "File > Exit" menu item to close the program.
   e. See Section 4.1 for additional trouble shooting instructions.

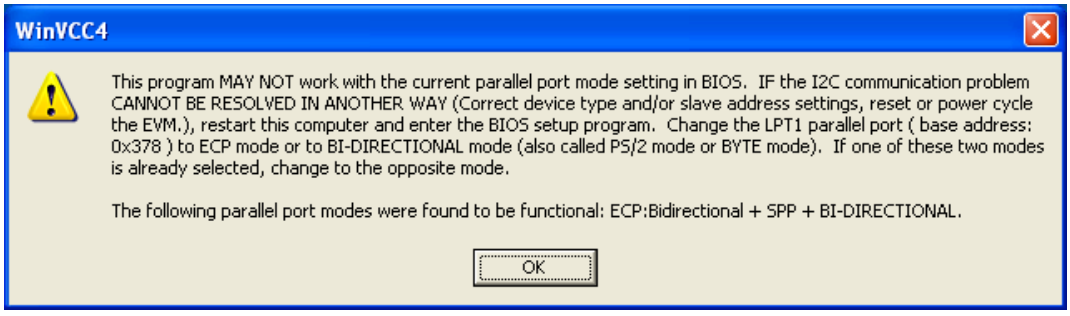

Figure 4-4. Corrective Action Required

4.2.1 Setting the PC Parallel Port Mode

Note: Run the PC BIOS setup program only if the I²C communication problem CANNOT BE RESOLVED IN ANOTHER WAY. (Correct slave address settings, reset or power cycle the EVM, and/or check that the device type selected was TVP5147).

1. Restart the PC.
2. During the boot process, enter the BIOS setup program by pressing the required key (usually the initial text screen indicates which key to press).
3. Find the place where the parallel port settings are made.
4. Set the parallel port LPT1 at address 378h to ECP mode or bidirectional mode (sometimes called PS/2 mode or byte mode). If one of these two modes is already selected, change to the opposite mode.
5. Exit and save changes.

4.2.2 General I^2C Error Report

The error report shown in Figure 4-5 is displayed when an I^2C error occurs at any time other than after the I^2C system test. In this example, there was an acknowledge error at slave address 0x54 (the video encoder module). The error occurred on Read Cycle Phase 1 on the device (slave) address byte.

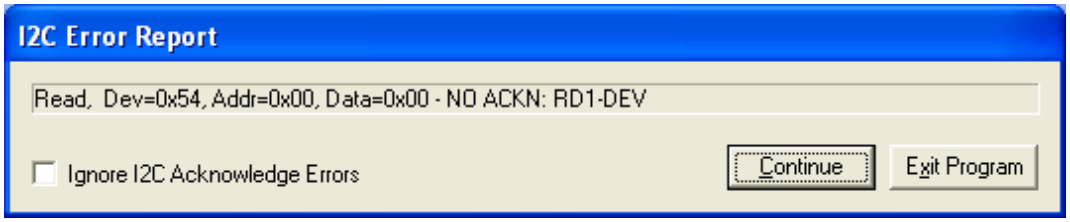

Figure 4-5. I^2C Error

Schematics and Board Layout

This chapter contains the TVP5147EVM schematics.

5.1 TVP5147EVM Schematics

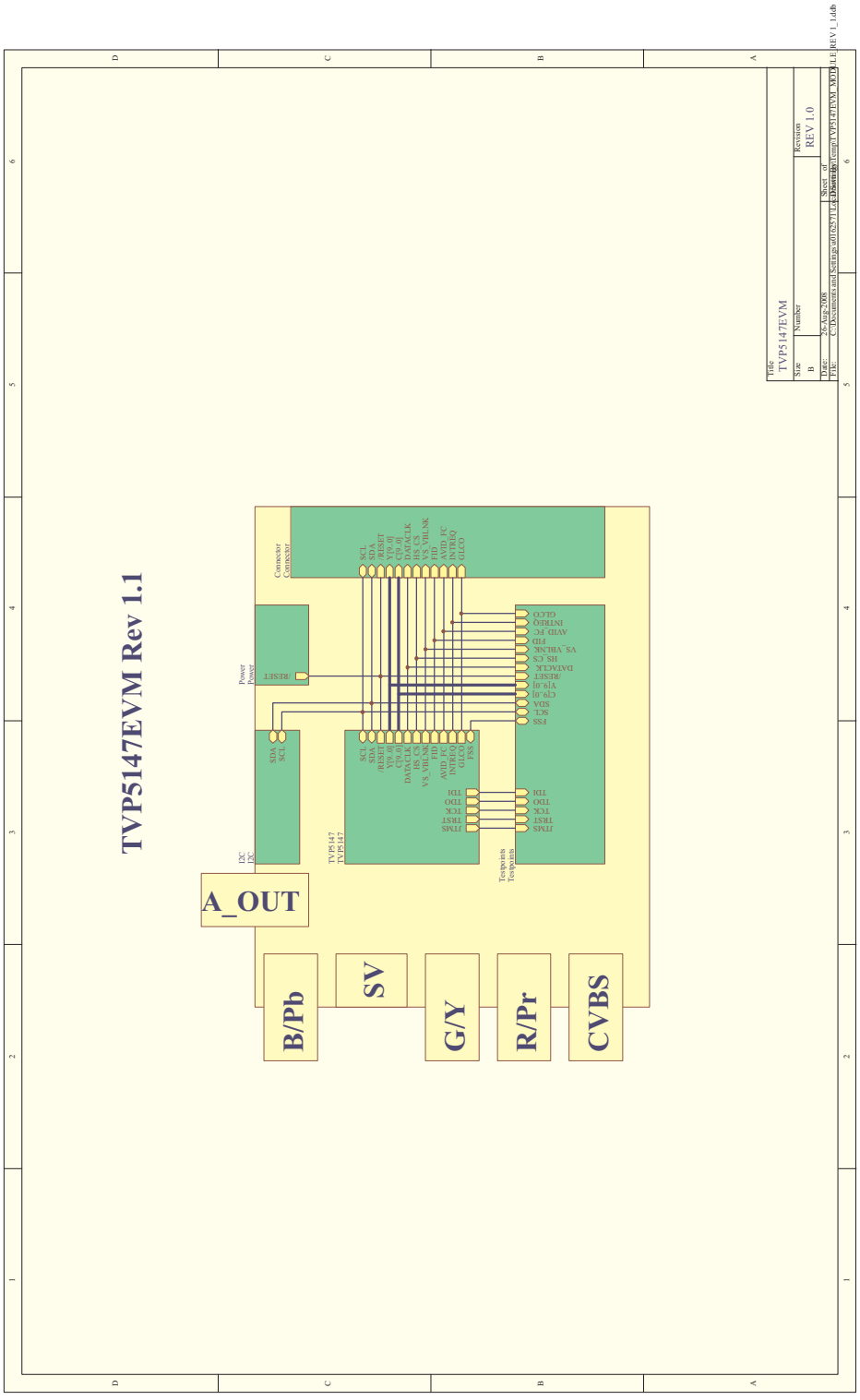

Figure 5-1. TVP5147EVM Schematics (1 of 6)

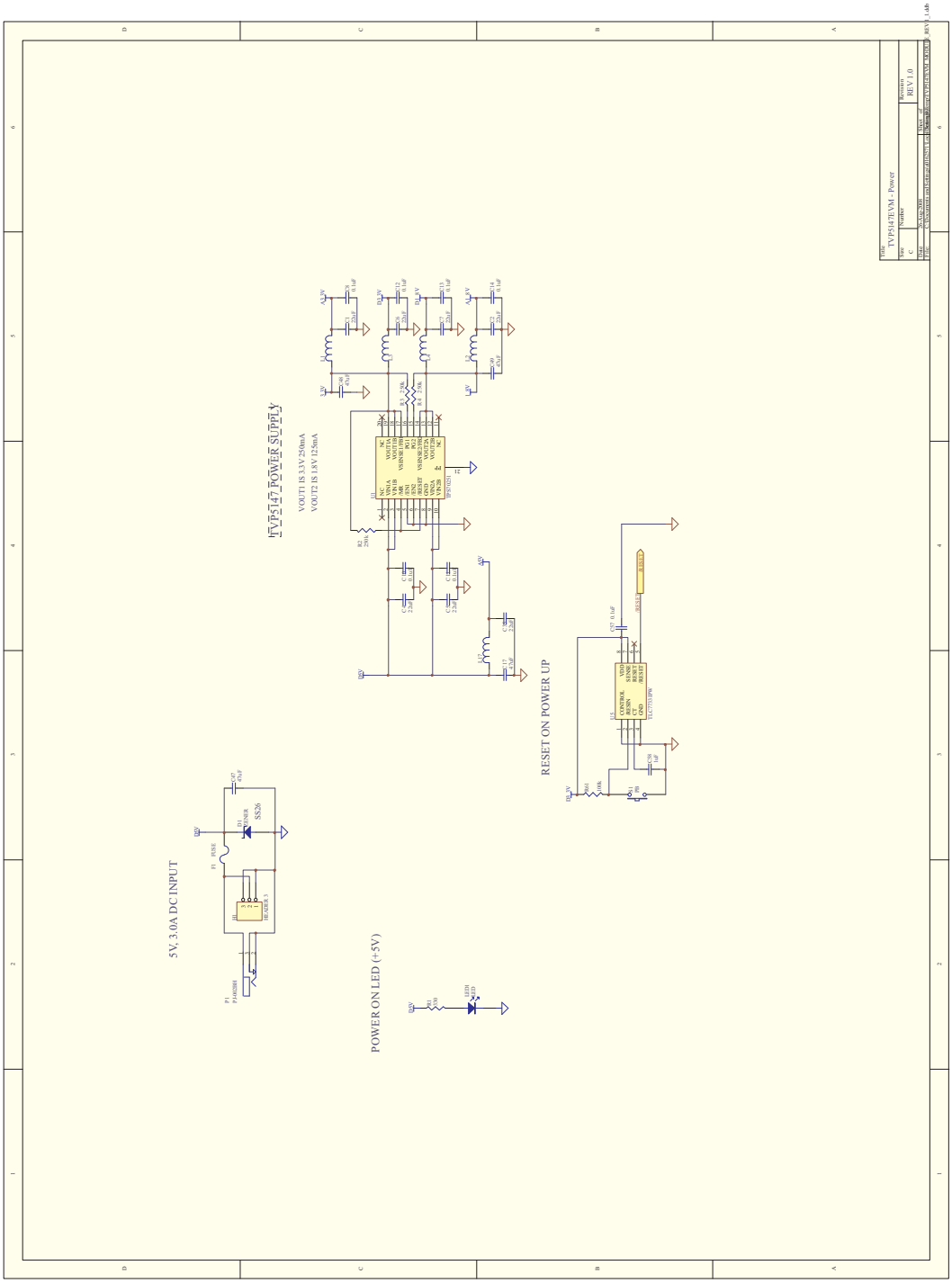

Figure 5-2. TVP5147EVM Schematics (2 of 6)

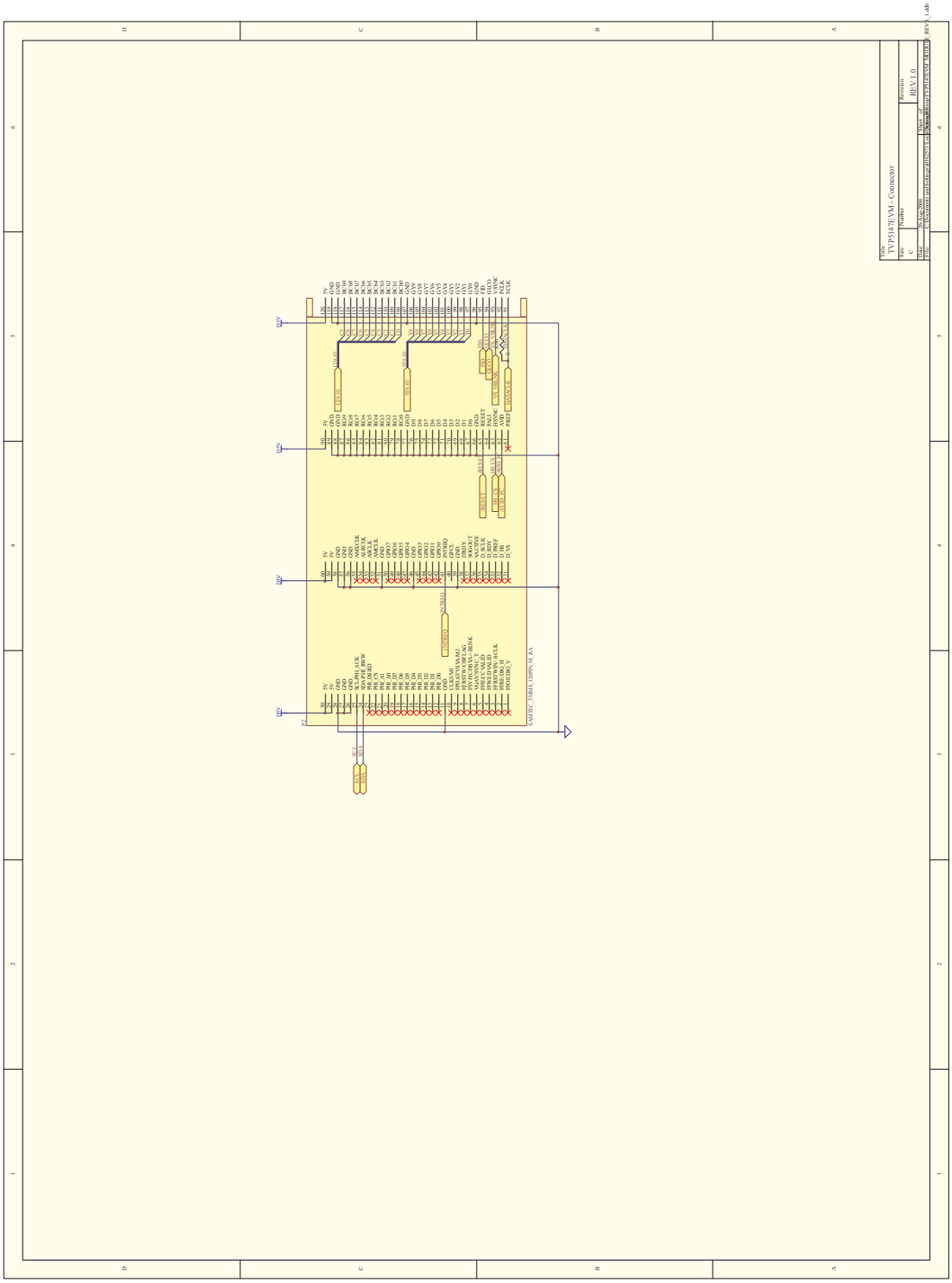

Figure 5-3. TVP5147EVM Schematics (3 of 6)

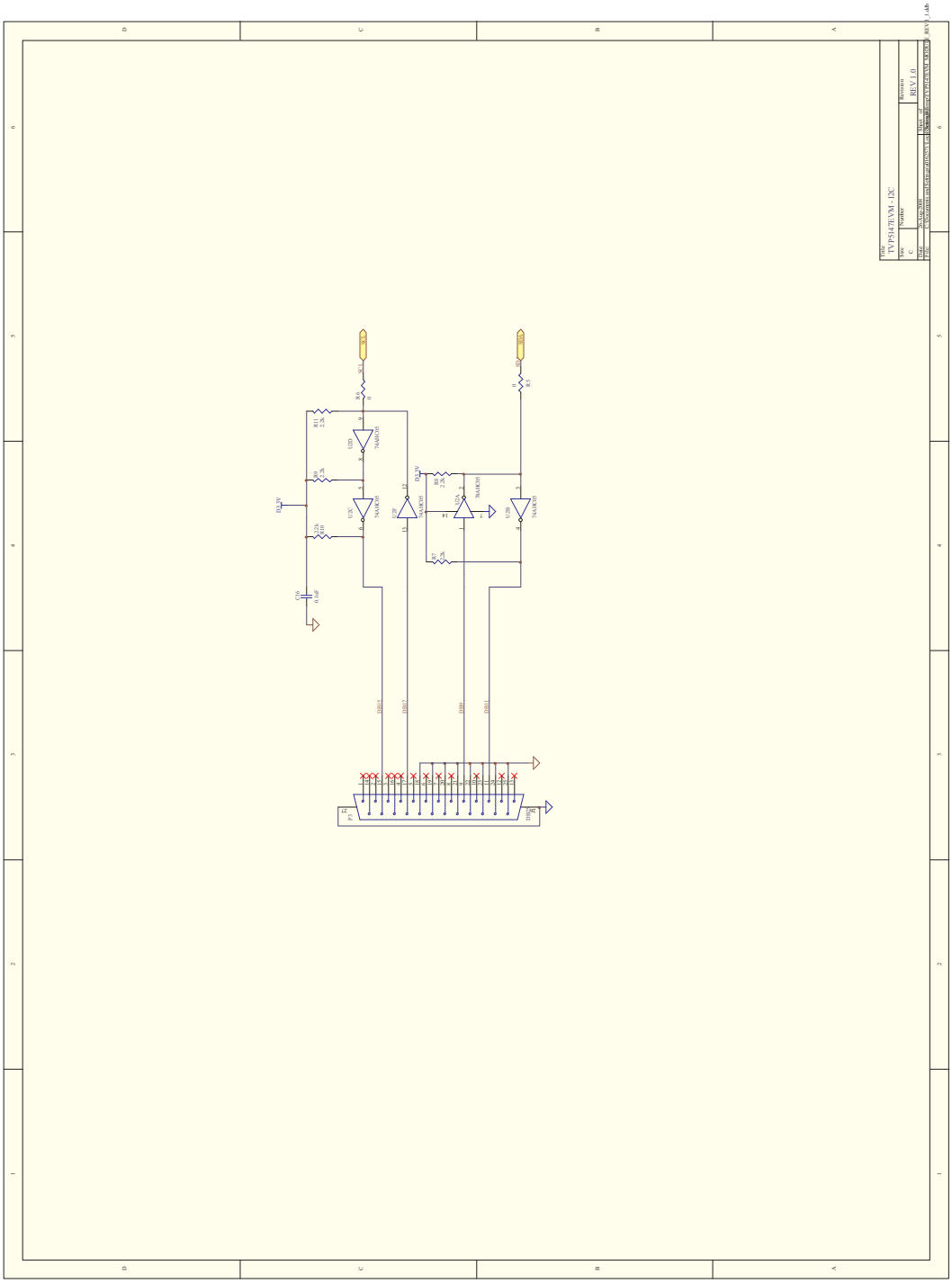

Figure 5-4. TVP5147EVM Schematics (4 of 6)

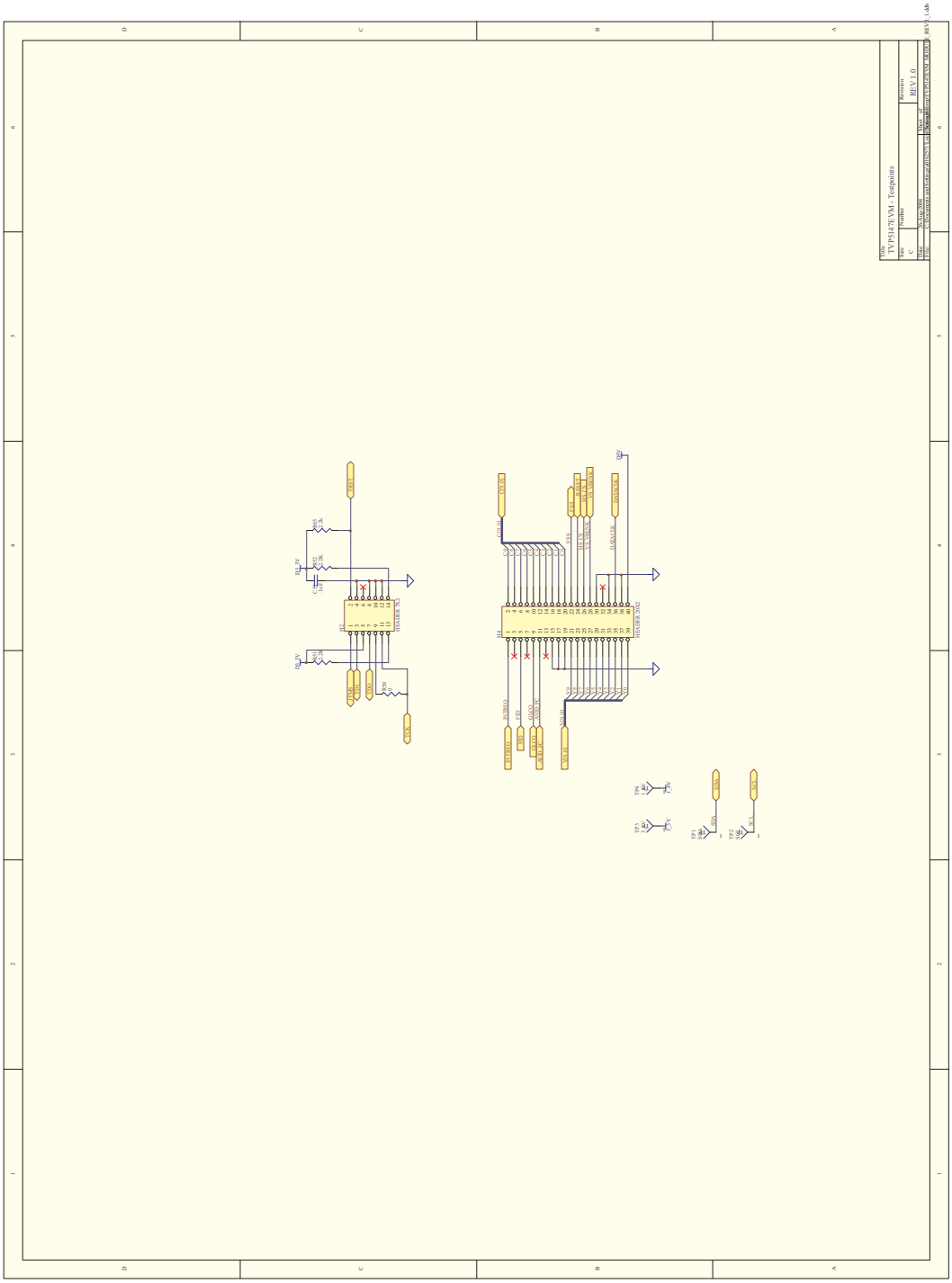

Figure 5-5. TVP5147EVM Schematics (5 of 6)

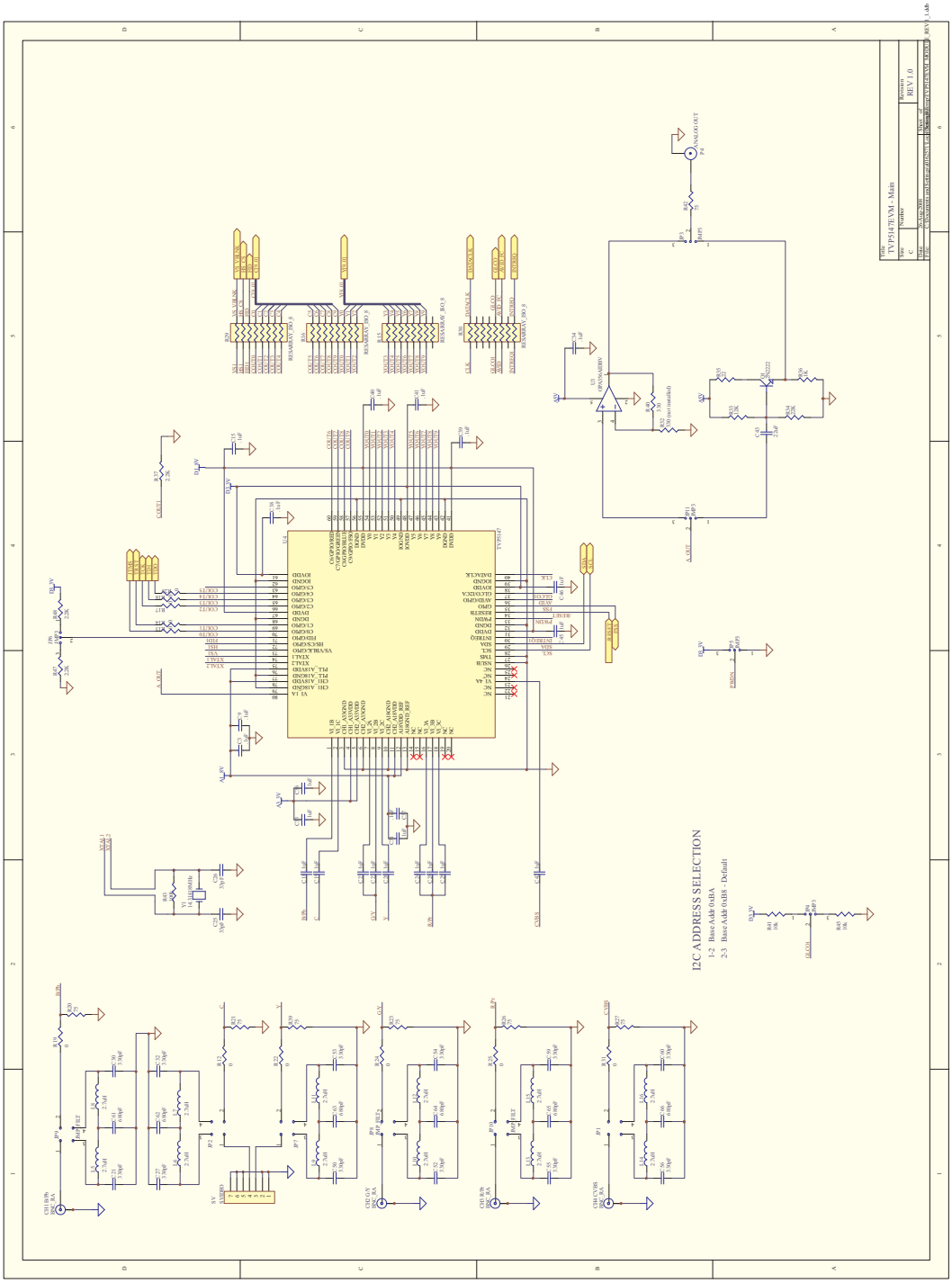

Figure 5-6. TVP5147EVM Schematics (6 of 6)

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
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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