OMAP5910 NTSC or VGA Output

Thanh Tran, Ph.D. *DSP/EEE Catalog, OMAP Applications*

ABSTRACT

The OMAP™5910 is a true system-on-a-chip device, which consists of an ARM925T microprocessor unit (MPU) and TMS320C55x digital signal processor (DSP) cores. The device has many important peripherals necessary for gluelessly interfacing with external devices in multimedia communication systems. One of the ports available is a general-purpose, 16-bit parallel bus called EMIFS that is typically being used for communicating with devices such as flash, ethernet controller, and others. This application report describes an external video system design using a 2-D graphics accelerator that interfaces with the EMIFS bus and provides a super video graphics array (SVGA), video graphics array (VGA), phase alternate line (PAL) or National TV Standards Committee (NTSC) video output.

Contents

List of Figures

List of Tables

1 Introduction

This application report describes an implementation of the 2-D graphics controller on the 16-bit parallel asynchronous bus of the OMAP5910 [1] [2]. The video system design consists of an external graphics accelerator and associated circuits to provide direct support for CRT up to 800x600 60Hz 64K colors, using embedded RAMDAC and direct support for NTSC/PAL TV output using embedded RAMDAC. The accelerators include a 2-D engine with bit block transfers or BitBLTs (write BLT, move BLT, solid fill, pattern fill, transparent write BLT transparent move BLT, read BLT, color expansion, and move BLT with color expansion) [3], and hardware cursor.

1.1 Supported Display Modes

- 4/8/15/16 bit per pixel (bpp) color depths
- Up to 64 shades of gray on monochrome passive LCD panels using frame rate modulation (FRM) and dithering
- Up to 32K/64K colors in 15/16 bpp modes on color passive LCD panels using dithering
- Up to 64K colors on TFT/D-TFD, CRT and TV

1.1.1 Example resolution: 320× 240 16bpp, 640× 240 16bpp, 640× 480 16bpp, 800× 600 16bpp

2 Design Description

Figure 1 shows the video architecture of the design and how the interface between an external graphics accelerator and the EMIFS port of the OMAP5910 is done. The system consists of three main parts: parallel bus interface, graphics accelerator, and TV video/VGA output.

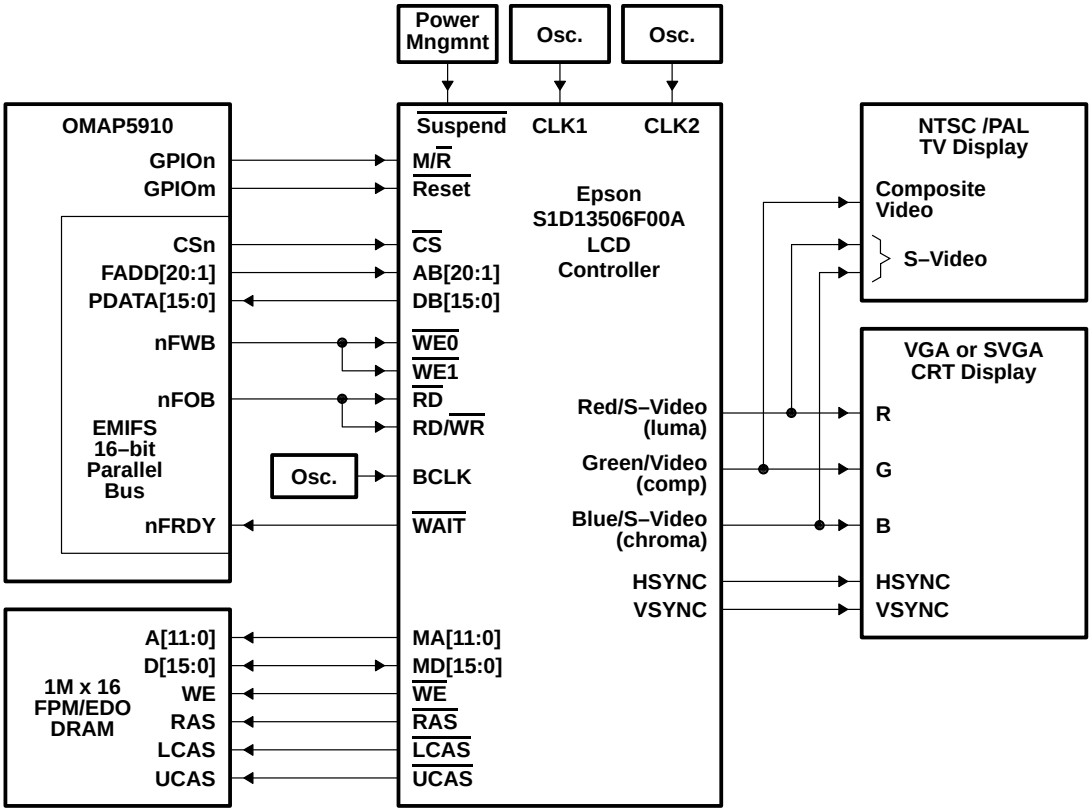


Figure 1. OMAP5910 Video Architecture

2.1 Parallel Bus Interface (EMIFS)

The EMIFS is a general-purpose, 16-bit data 24-bit address bus that is designed to interface and handle all transactions to flash memory, read-only memory (ROM), asynchronous memories, and synchronous burst flash. In addition to interfacing with different memory devices, this bus can be configured to work with other hardware accelerators such as 2D graphics controllers, full-frame-rate MPEG-2 decoders, and other 16-bit devices.

The interface can drive up to four devices by assignment to one of the four chip-selects. Each chip-select has a corresponding register, to specify the protocol used for the associated external device.

Table 1 outlines the signals associated with the EMIFS port. For more detailed technical information and discussion of this port, refer to the OMAP5910 Data Manual [1], which can be downloaded from the OMAP web site, www.omap.com

Table 1. EMIFS Signal List

Signal Name	I/O	Description
FLASH.RDY	I	Ready/busy signal from device
$\overline{\text{FLASH.WP}}$	O	Active-low write protection
FLASH.CLK	I/O	Clock signal for flash device
$\overline{\text{FLASH.RP}}$	O	Active-low flash power-down/reset
$\overline{\text{FLASH.CS0}}$	O	First active-low chip-select
$\overline{\text{FLASH.CS1}}$	O	Second active-low chip-select
$\overline{\text{FLASH.CS2}}$	O	Third active low chip-select
$\overline{\text{FLASH.CS3}}$	O	Fourth active low chip-select
$\overline{\text{FLASH.OE}}$	O	Active-low output enable
$\overline{\text{FLASH.WE}}$	O	Active-low write enable
$\overline{\text{FLASH.ADV}}$	O	Active-low address valid
FLASH.D[15:0]	I/O	16-bit flash data bus
FLASH.A[24:1]	O	24-bit flash address bus
$\overline{\text{FLASH.BE}}$	O	Active-low byte enable

2.2 2-D Graphics Accelerator on EMIFS

The external 2-D accelerator being utilized is the Epson S1D13506 color liquid crystal display/cathode ray tube/television (LCD/CRT/TV) controller [3]. This device can be configured to operate in different bus interface modes for supporting reduced instruction set computing (RISC), million instructions per second (MIPS) and other processors. In this design, the bus is configured in a generic mode, to allow direct connection to the EMIFS port of the OMAP.

Figure 2 shows the video accelerator device architecture. In this design, only the CRT/TV path is being used, since the design is intended to drive a stand-alone flat panel with analog RGB inputs, computer monitor with analog RGB inputs, or TV with composite or S-Video input.

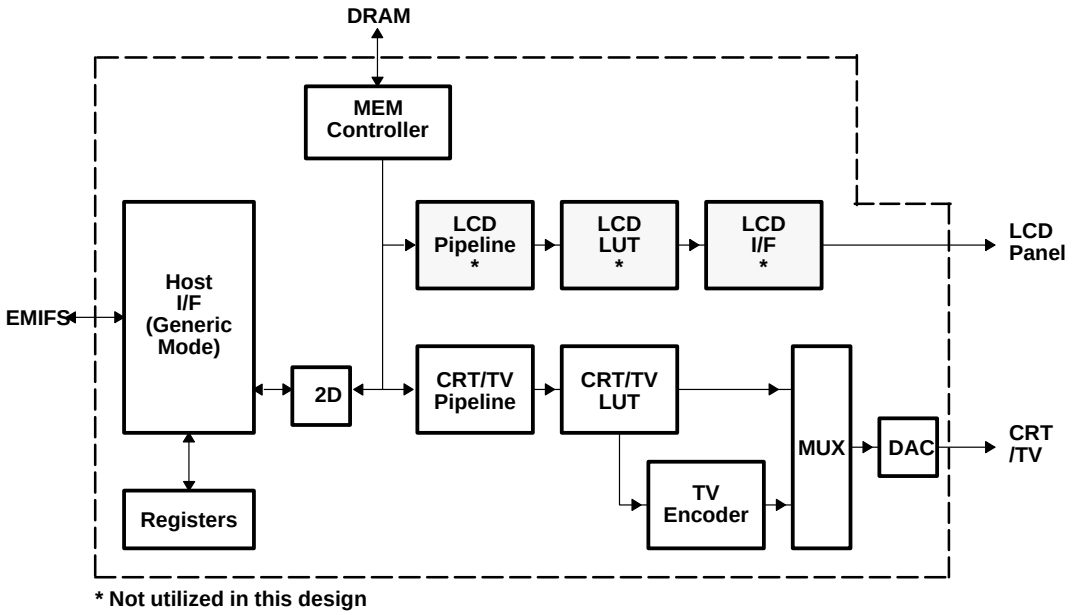


Figure 2. S1D13506 Block Diagram

2.2.1 Host Interface Block

The host interface port consists of a 16-bit data bus, 21-bit address bus, and control signals. Table 2 shows how the S1D13506 host interface is connected to the EMIFS port of the OMAP.

Table 2. Host Interface Connection to EMIFS

Host Interface (2D)	EMIFS (OMAP)	Description
DB[15:0]	FLASH.D[15:0]	16-bit bus connected to the OMAP Flash data bus
AB[0]	—	Connected to V_{DD} for Generic mode
AB[20:1]	FLASH.D[20:1]	The upper 20 bits of the host interface addr bus, connected directly to the OMAP addr bus
—	FLASH.D[24:21]	Not used by the host interface
$\overline{CS}$	$\overline{FLASH.CS2}$	Chip-select connected to Flash chip-select 2
M/$\overline{R}$	GPIO1	Memory or register access connected to the GPIO1 of the OMAP
$\overline{BS}$	—	Connected to V_{DD} for generic mode
$\overline{RD}$	$\overline{FLASH.OE}$	Input read command for lower data byte connected to the Flash output enable
RD/$\overline{WR}$	$\overline{FLASH.OE}$	In generic mode, this input reads the command for upper data byte and is connected to the Flash output enable
$\overline{WE0}$	$\overline{FLASH.WE}$	Input write enable command for lower data byte
$\overline{WE1}$	$\overline{FLASH.WE}$	Input write enable command for upper data byte
$\overline{RESET}$	GPIO2	Device reset connected to GPIO2, allowing OMAP to control the accelerator

Table 2. Host Interface Connection to EMIFS (Continued)

Host Interface (2D)	EMIFS (OMAP)	Description
BUSCLK	—	Driven by external 80-MHz oscillator, with BUSCLK divide by 2 option (40 MHz operation)
$\overline{\text{WAIT}}$	FLASH.RDY	OMAP flash ready is not supported.

In this design, the S1D13506 graphics accelerator bus interface is configured in the generic mode, which requires some signals on the graphics memory interface to be pulled up appropriately. Table 3 shows a summary of the power-on-reset options selected for this design.

Table 3. Power-On-Reset Options

Mode	10K Resistor Pullup
Generic	Pins 37 and 39 (MD1 and MD2)
Little Endian	Pin 43 (MD4)
2MB DRAM Frame Buffer	Pin 47 (MD6)
Bus Clock Divided by 2	Pin 40 (MD12)

2.2.2 2-D Accelerator Block

The 2-D block within the graphics accelerator includes a hardware cursor for fast cursor update, a double buffering scheme for smooth animation and instantaneous screen update, and hardware BitBLTs for moving and copying rectangles of bits between the main and display memories. The integrated BitBLTs are Write BLT, Move BLT, Solid Fill, Pattern Fill, Transparent Fill, Transparent Write BLT, Transparent Move BLT, Read BLT, Color Expansion, and Move BLT with Color Expansion. The BitBLT engine design is based on a 16-bit architecture that supports rectangular and linear addressing modes for source and destination. This 2-D block also has a dedicated BitBLT IO access space, which allows for running multi-tasking applications, such as simultaneous BitBLT and CPU read/write operations.

2.2.2.1 BitBLT Operations

Table 4 shows a summary of the BitBLT operations. For the definition of the BitBLT registers, see the component specifications [3] .

Table 4. Summary of BitBLT Operations

BitBLT Name	Function
Write BitBLT	Provides two 16-bit operand ROP (raster operation) functions
Move BitBLT	Provides two 16-bit operand ROP functions and supports both positive and negative directions
Read BitBLT	Supports bit block transfers from the display buffer to the host, no ROP function applied
Solid Fill	Fills the specified BitBLT area with a solid color, as defined in the foreground color register. In the 8-bpp mode, only the lower byte of the foreground color is used.
Pattern Fill	Fills the specified BitBLT area with an 8-pixel by 8-line pattern in the full color defined in the off-screen display buffer. The pattern data has to be stored in a contiguous address.

Table 4. Summary of BitBLT Operations (Continued)

BitBLT Name	Function
Transparent Pattern Fill	Fills the specified BitBLT area with an 8 x 8 pattern in the full color defined in the off-screen display buffer
Transparent Write BitBLT	Supports bit block transfers from the host to display buffer
Transparent Move BitBLT	Supports bit block transfers from display buffer to display buffer in a positive direction only
Color Expansion BitBLT	Expands the host's monochrome data to 8- or 16-bpp color format
Move BitBLT with Color Expansion	Move BitBLT with color expansion

2.2.3 TV/CRT Look-Up-Table (LUT)

The LUT architecture consists of four modes, monochrome, 4 bit-per pixel, 8 bit-per-pixel, and 15/16 bit-per-pixel. Except for the 15/16 bit-per-pixel, all other modes use the LUT to create proper colors, or a gray scale for the monochrome case that are controlled by the pixel data stored in the image buffer. For the 15/16 bit-per-pixel case, the LUT is bypassed and the color data is directly mapped to the color depths showing in Figure 6. Figure 3 through Figure 6 show the LUT architectures for all modes, and how the image buffer is being arranged for the direct mode or 15/16 bit-per-pixel mode.

In the monochrome mode, shown in Figure 3, only the green outputs are being used to send out different gray scale data to the integrated digital–to–analog converter (DAC). The 4 bit-per-pixel from the image buffer selects which gray level to display from the LUT.

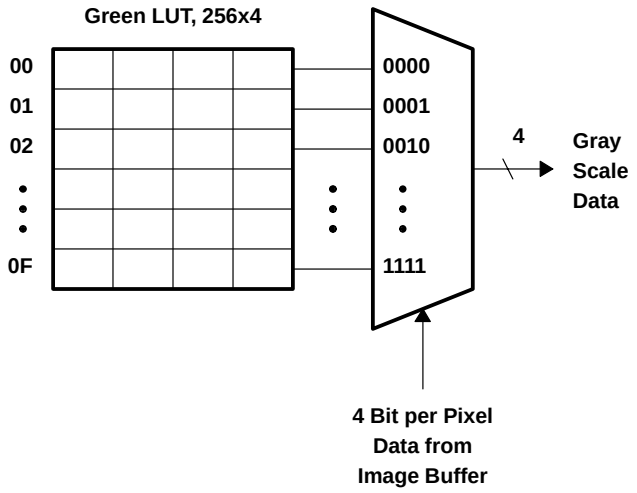


Figure 3. 4-Bit-Per-Pixel Monochrome Mode Data Output Path

In the 4 bit-per-pixel color mode, shown in Figure 4, the 4-bit-per-pixel data from the image buffer selects which colors to display from the three LUTs (red, green, and blue). The LUT only consists of 16 levels for each color, and the outputs to the integrated DAC are always 4 bits per displayed color.

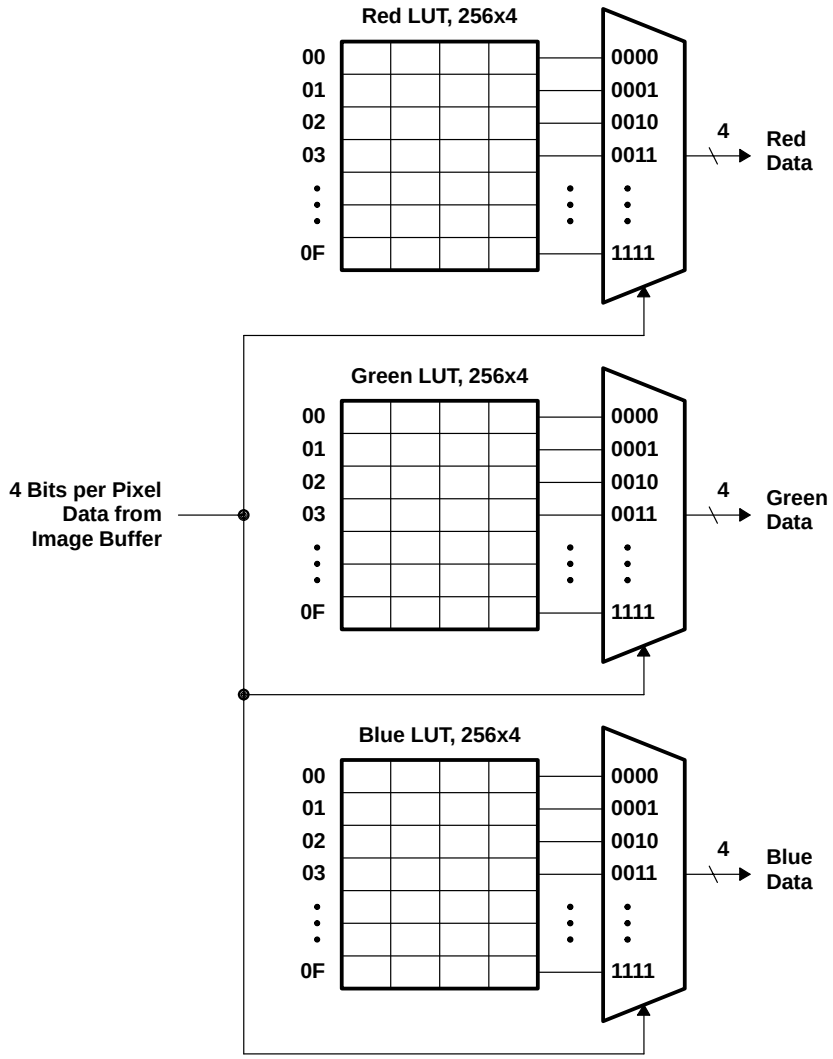


Figure 4. 4-Bit-Per-Pixel Color Mode Data Output Path

In the 8-bit-per-pixel color mode, shown in Figure 5, the 8-bit-per-pixel data from the image buffer selects which colors to display from the three LUTs. Each of the LUT consists of 256 colors, and the outputs to the integrated DAC are always 4 bits per displayed color.

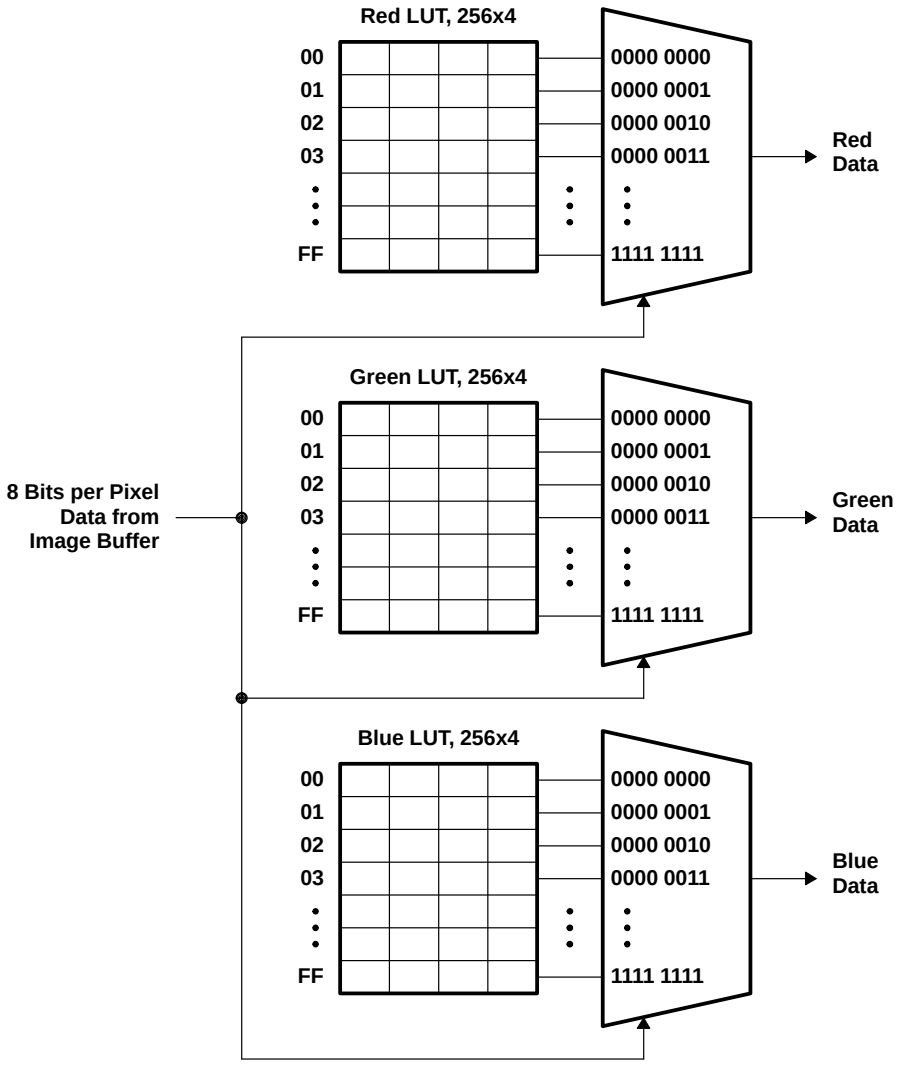


Figure 5. 8-Bit-Per-Pixel Color Mode Data Output Path

In the 15/16 bit-per-pixel color modes, shown in Figure 6, the LUT is bypassed, and the colors are directly mapped to the selected color depths. In this case, the display memory are arranged, as shown in Figure 6, for the little-endian system, where the host addresses the display buffer directly to display the 15- or 16-bit color depth on the selected pixel. For example, in the 15-bit color mode, the pixel P_0 is displaying the RGB data addressed by byte 0 and byte 1 from the host.

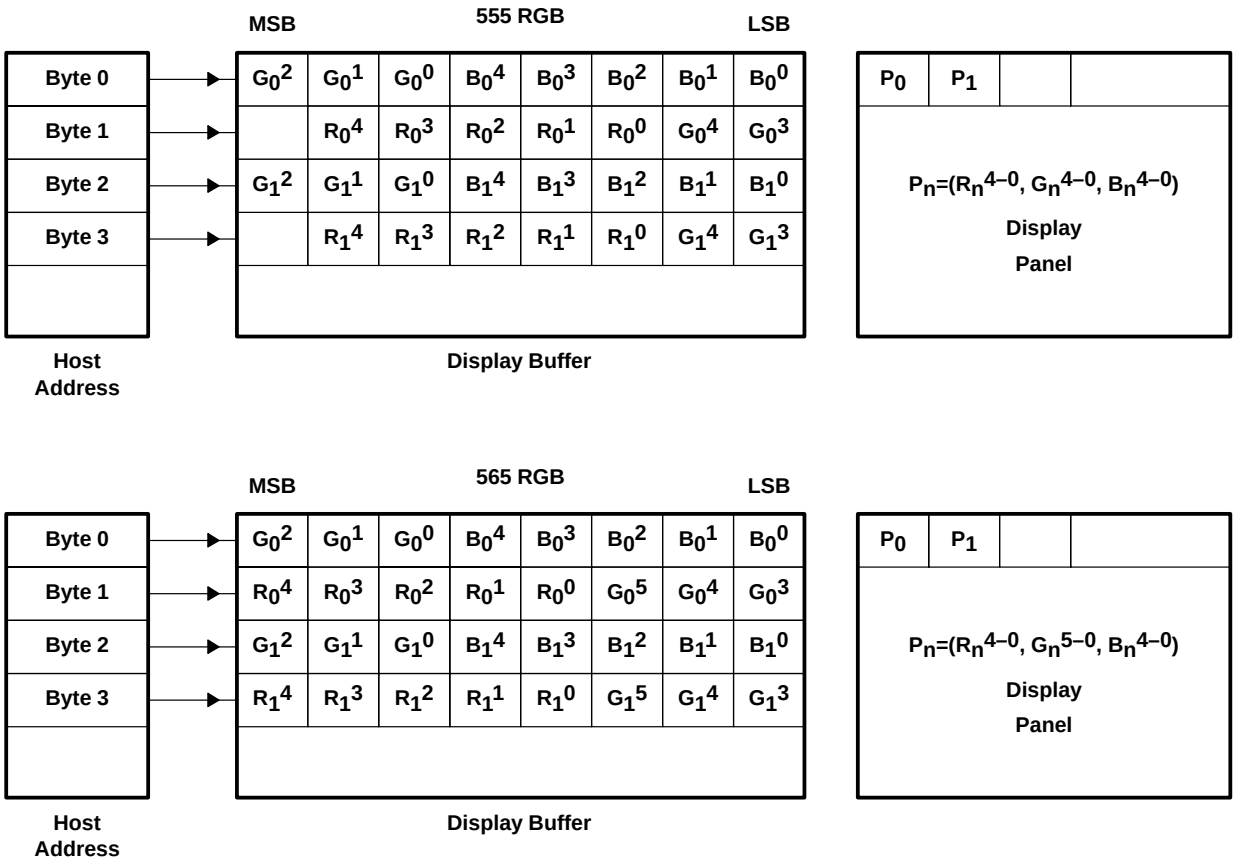


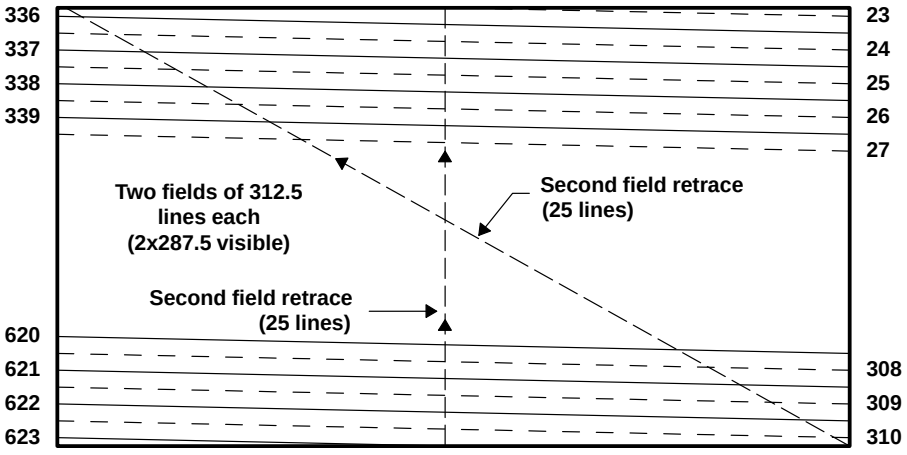
Figure 6. 15/16 Bit-Per-Pixel Format Memory Organization

2.2.4 TV Encoder

As shown in the Figure 2 block diagram, the video output can be configured as CRT or TV mode. For CRT, the DAC data input comes from the LUT or the image buffer directly, as described in section 2.2.3. For the TV mode, the DAC data input comes from the TV encoder, which can be configured to output either NTSC (14.31818 MHz input to CLKI2 or pin 71) or PAL (17.734475 MHz to CLKI2 or pin 71) video format. In this design, only NTSC mode is included. The video data output from the 2-D graphics block is compatible with a noninterlaced display, such as computer monitor. To display the data on an interlaced TV display, it is necessary to do the non-interlaced-to-interlaced conversion.

2.2.4.1 Noninterlaced to Interlaced Conversion Overview

Figure 7 shows an example of how interlaced scanning works for the television monitor. Each TV frame consists of two fields, and each field comprises half the scanning lines. The second field is delayed half the frame time from the first field. For NTSC, there are 780 pixels in one horizontal line, 525 horizontal lines total, and 59.94-Hz vertical refresh rate.



Schematic representation of interlaced scanning (625 lines)

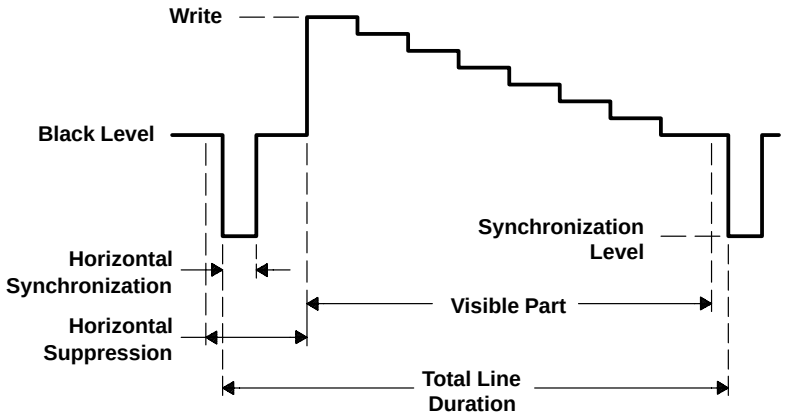


Figure 7. Interlaced Scanning for Television

There are two common techniques [4] to convert the non-interlaced to interfaced format, throwing away every other line in each non-interlaced frame, and using three scan lines of non-interlaced frame to generate one interlaced scan line. The first technique is simple but the video artifacts, such as flicker, are not acceptable; therefore, the second method is preferred. The disadvantage of the second method, where the conversion requires three non-interlaced scan lines to generate one interlaced scan line, is that this technique requires line stores to store the video data for processing as well as digital filters. This method greatly reduces the flicker and makes this the preferred implementation. Figure 8 demonstrates the first method, where each interlaced field is generated by throwing away every other line of the noninterlaced frame. Figure 9 demonstrates the second conversion method, where each interlaced field is generated by taking an average of three lines.

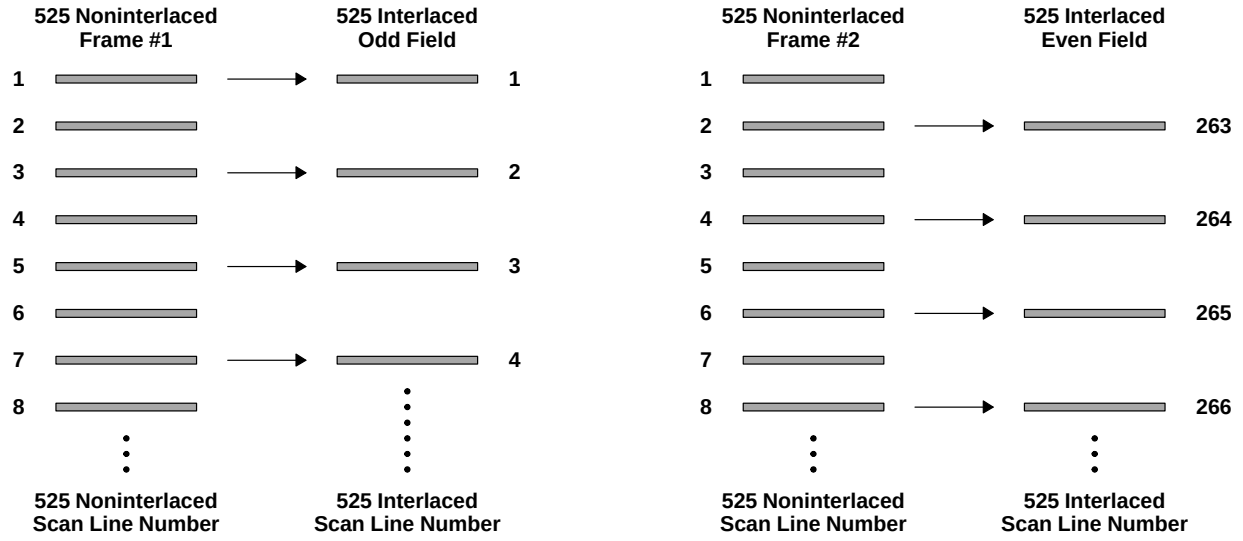


Figure 8. Non-Interlaced-to-Interlaced Conversion (Throw-Away Lines)

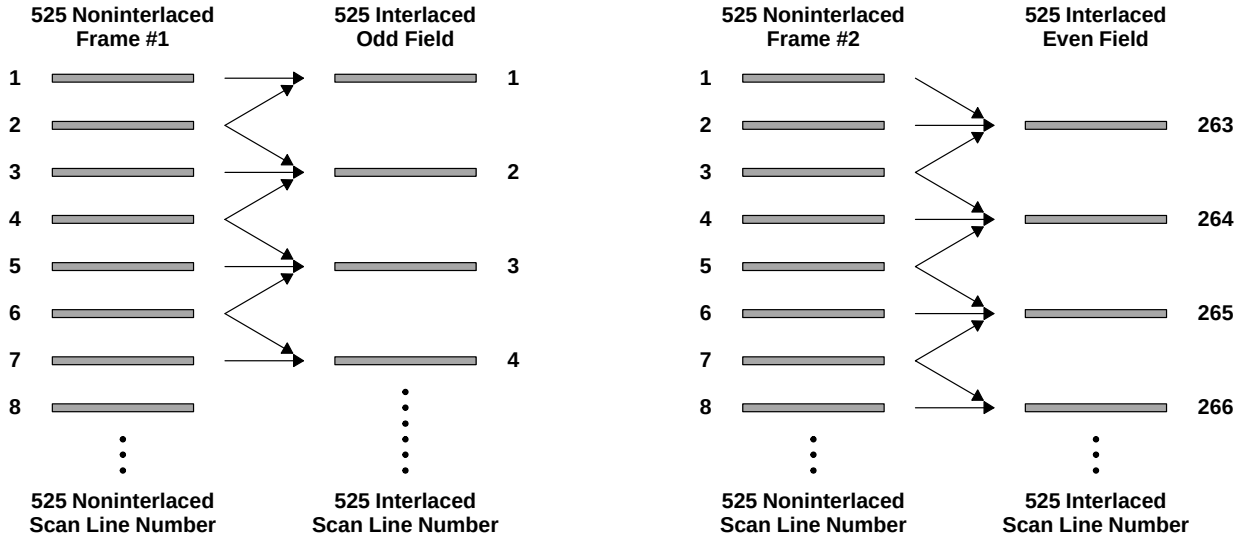


Figure 9. Non-Interlaced-to-Interlaced Conversion (Lines Averaging)

2.2.4.2 NTSC TV Encoder Filters

The S1D13506 TV encoder design is based on averaging three non-interlaced lines to create one interlaced line. Using this technique requires proper digital filters to reduce image distortions such as cross-luminance, cross-chrominance, and flicker. The distortions are typically generated by sharp chrominance and luminance transitions, which are necessary for high-resolution computer graphics. To optimize the quality on the TV output, it is preferable to turn on the filters listed in Table 5.

Table 5. TV Encoder Filters

Filter	Register	Definition
Chrominance Filter	REG[05Bh], bit 5 = “1” enable	Reduces cross-luminance distortion by limiting the bandwidth of the chrominance signal. This filter has the most effect on the composite video output.
Luminance Filter	REG[05Bh], bit 4 = “1” enable	Reduces cross-chrominance distortion by limiting the bandwidth of the luminance signal. This filter has the most effect on the composite video output.
Anti-Flicker Filter	REG[1FCh], bits [2:0] = ”110” flicker filter enable	Reduces flicker on TV display caused by displaying only half the refresh rate of the original non-interlaced images.

2.3 Video System Design

Figure 10 shows a complete video system design that includes external components to interface with the OMAP5910. External filter circuitries and low-noise board layout techniques are required and are very important for designers to achieve an acceptable video performance. Here are some of the recommended techniques for low noise video system design:

- Separate the power supply voltage for the video DAC from the system power supply voltage. The best way is to use a high power supply rejection ratio (PSRR) linear voltage regulator such as TI TPS76701QPWPR.
- Place the 100-nF capacitors as close to the power pins of the video device as possible, and distribute the number of the capacitors evenly around the part. Place one 10- μF bulk capacitor at each side of the device as close to the 100 nF as possible.
- As shown in Figure 10, all data bus, address bus, and clock signals should have 22-Ω series termination resistors. These termination resistors should be placed as close to the pins as possible.
- Use 100-Ω ferrite beads at 100 MHz for the video output Pi filters. The Pi structure provides two-pole roll-off where the first corner frequency is $\frac{1}{2\pi RC}$, where *R* is the device output resistance, and *C* is the external filter capacitance at the output pin (C100, 102, 104, 106, or 108). The second corner frequency is also $\frac{1}{2\pi RC}$, where *R* is the ferrite bead impedance, and *C* is the capacitance connected from the VGA connector to ground (C101, 103, 105, 107 or 109).
- Keep all the high-speed signal traces as short as possible. The video outputs (red, green, blue, hsync, and vsync) should have the same widths and lengths.

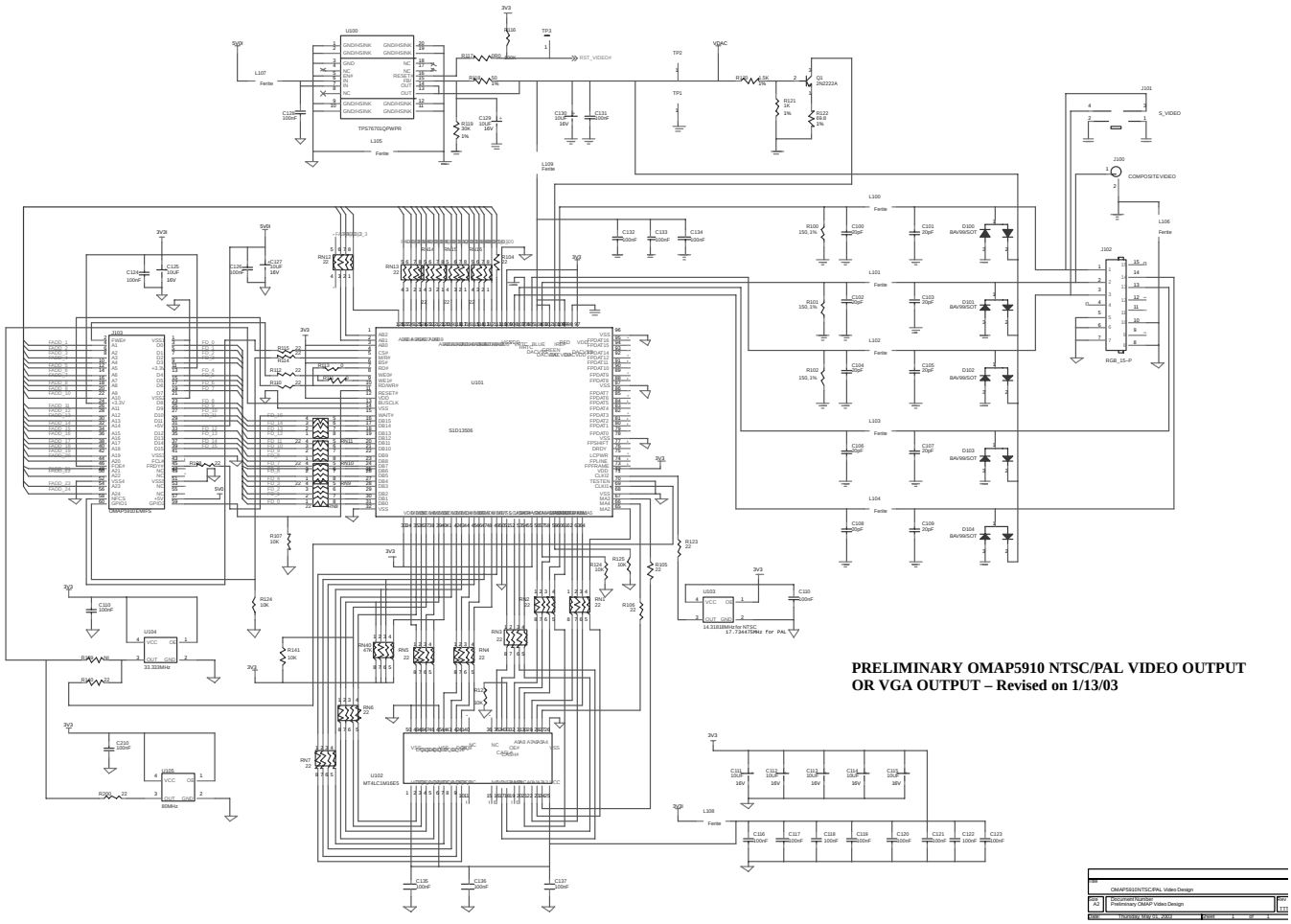


Figure 10. OMAP5910 Video System Design

2.4 Software

Refer [6] for programming notes and examples for initializing the S1D13506 Color LCD/CRT/TV Controller. The reference source codes are available on the Epson web site, www.eea.epson.com. Currently, there are device drivers available for many popular embedded operating systems and detailed information can be found on the same web site.

Figure 11 demonstrates the OMAP5910 on the Innovator Development platform [5] interfacing with the S1D13506 graphics controller, to drive an external VGA monitor.

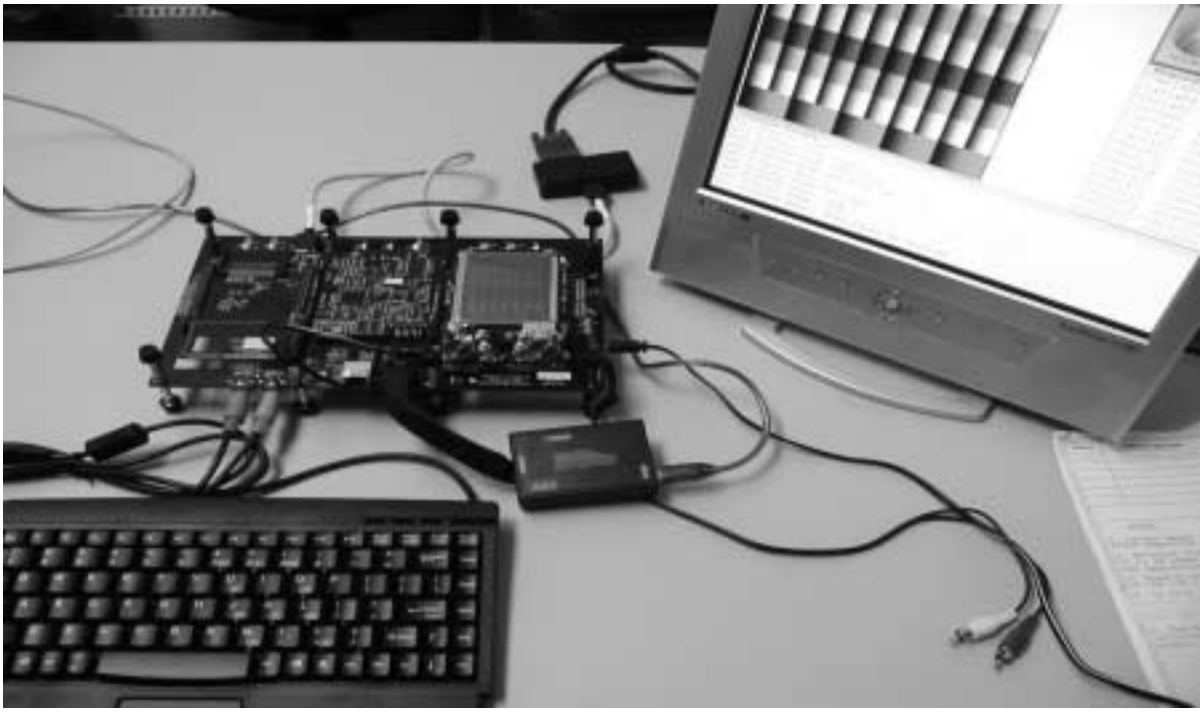


Figure 11. OMAP5910 Video Demonstration on Innovator

3 Conclusion

This application report provides an example for OMAP5910 system designers to add a VGA or NTSC video output feature to their product definition. It is intended to provide important information to successfully design the video hardware and to interface with the EMIFS bus of the OMAP5910. While the document only discusses the NTSC video, it is possible to make simple a modification (changing the video clock frequency) to provide a PAL video output as well.See [3] for more detailed information.

One limitation to this design is that only one output can be enabled at a time. For example, designers can only configure the video controller to operate in one of the two modes, VGA up to 800 × 600 resolution or NTSC video, not both simultaneously.

4 References

1. *OMAP5910 Dual-Core Processor Data Manual* (SPRS197).
2. *OMAP5910 Dual-Core Processor Technical Reference Manual* (SPRU602).
3. Epson Research and Development. *S1D13506 Color LCD/CRT/TV Controller Technical Manual*. www.eea.epson.com
4. Jack, Keith. *Video Demystified*. San Diego: Hightext, 1996.
5. Productivity Systems, Inc. *Innovator Development Kit for the Texas Instruments OMAP Platform*, www.prodsys.com
6. Epson Research and Development. *S1D13506 Programming Notes and Examples*. www.eea.epson.com
7. Poynton, Charles A. *A Technical Introduction to Digital Video*. Toronto: Wiley, 1996.

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 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.

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.

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265