Application Report
SPNA101–December 2006

TMS470 Expansion Bus Module Example

Keith Engler and John Mangino *TMS470 Applications*

ABSTRACT

This document supplies information for using the Expansion Bus Module (EBM) on the TMS470R1B1M to interface to external memory and peripherals. The module supports the multiplexing of the input/output (I/O) functions and the expansion bus interface. When the I/O functions are not used, the EBM can be used to interface 8- or 16-bit memories.

Contents

List of Figures

Disclaimer

This document is not intended to replace the TMS470R1B1M data manual nor is it in any way a device specification, and all values in this document cannot be ensured. In case of any discrepancy between this document and the TMS470R1B1M data manual, the TMS470R1B1M data manual is correct.

1 Overview

The TMS470R1B1M is well suited to interfacing to external memory. The Cypress CY62148DV30 SRAM (8 × 512K) directly maps to the TMS470R1B1M.

Following is a discussion of how much extended memory can be accessed directly with the TMS470R1B1M using the Expansion Bus Module (EBM). Each fetch of data from the extended memory requires a minimum of three clock cycles; this is shown on page 49 of the TMS470R1B1M data manual (TI literature number SPNS109). (If the memory is slow and longer wait states are required, each fetch may require more clock cycles). In the application described in this document, 30 GIO lines are used for the EBM interface.

These are the general EBM features (device specific):

- Multiplexing of I/O signals to an expansion memory interface or a peripheral interface
- Supports 8- and 16-bit expansion bus memory interface mappings
- Supports mapping of the following expansion bus signals:
  - Up to 30-bit address bus (EBADDR[29:0]) for 8-bit data bus
  - Up to 22-bit address bus (EBADDR[21:0]) for 16-bit data bus
  - 8- or 16-bit data bus (EBDATA[7:0] or EBDATA[15:0])
  - Up to two write strobes
  - Up to four memory chip selects
  - One output enable
  - One external hold signal for interfacing to slow memories
  - Up to eight DMA request lines

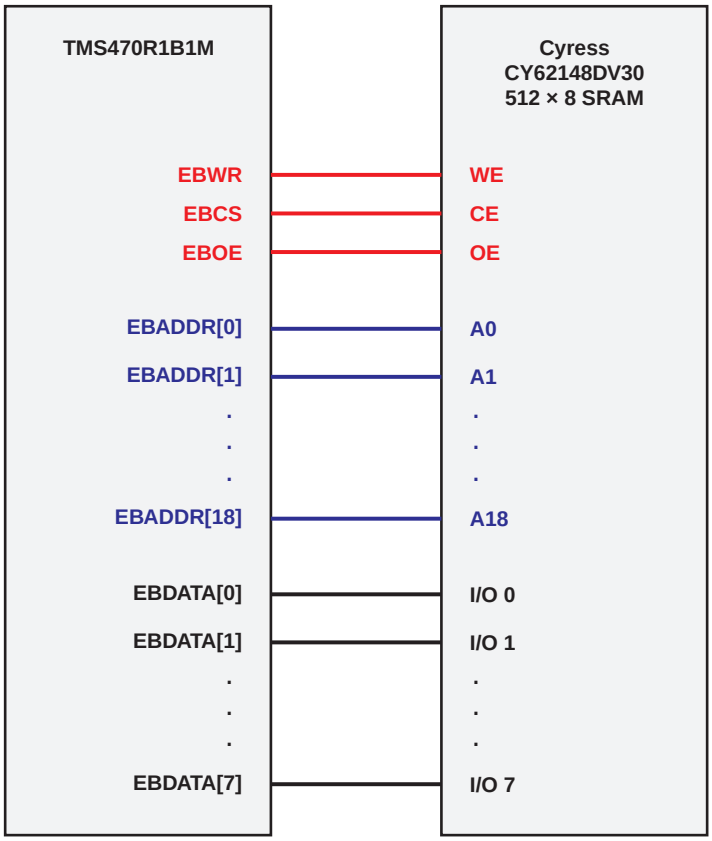
2 TMS470 EBM Interface to Cypress CY62148DV30

2.1 TMS470R1B1M EBM Pin map to the Cypress CY62148DV30

The following table shows the pin mapping from the TMS470R1B1M to the Cypress memory.

TMS470R1B1M EBM Address		Cypress Memory Address	
Pin Description	**Pin Number**	**Pin Description**	**32-Pin SOIC Pin Number**
EBADDR[0]	42	A0	12
EBADDR[1]	39	A1	11
EBADDR[2]	35	A2	10
EBADDR[3]	30	A3	9
EBADDR[4]	27	A4	8
EBADDR[5]	23	A5	7
EBADDR[6]	80	A6	6
EBADDR[7]	82	A7	5
EBADDR[8]	89	A8	27
EBADDR[9]	90	A9	26
EBADDR[10]	93	A10	23
EBADDR[11]	96	A11	25
EBADDR[12]	99	A12	4
EBADDR[13]	100	A13	28
EBADDR[14]	20	A14	3
EBADDR[15]	10	A15	31
EBADDR[16]	8	A16	2
EBADDR[17]	6	A17	1
EBADDR[18]	3	A18	30
TMS470R1B1M EBM Data		**Cypress Memory I/O Data**	
Pin Description	**Pin Number**	**Pin Description**	**32-Pin SOIC Pin Number**
EBDATA[0]	44	I/O 0	13
EBDATA[1]	47	I/O 1	14
EBDATA[2]	58	I/O 2	15
EBDATA[3]	61	I/O 3	17
EBDATA[4]	64	I/O 4	18
EBDATA[5]	67	I/O 5	19
EBDATA[6]	70	I/O 6	20
EBDATA[7]	77	I/O 7	21
TMS470R1B1M EBM Control		**Cypress Memory I/O Control**	
Pin Description	**Pin Number**	**Pin Description**	**32-Pin SOIC Pin Number**
EBWR[0]	128	WE	29
EBCS[5]	120	CE	22
EBOE[0]	135	OE	24

2.2 Connectivity Block Diagram



3 Code Examples

3.1 Main Routine Setup

```
// Added to the main routine to setup the EBM registers for Writing
  EBMCR1=0x00; // EBM Control register 1 set for 8 bit Data
  EBRWCR=0x7F; // EBM Read/Write Control Register
               // 7   = EBHOLD = 0   = The HOLD not mapped to an external device
               // 6:3 = EBCS   = 1111 = Chip Selects mapped to the Mux Output
               // 2:1 = EBWR   = 11  = The Write Enable bits mapped to the Mux Output
               // 0   = EBOE   = 1   = The Output Enable bit mapped to the Mux Output

  //The next four Registers Map Addresses [29:0] to the Mux Output
  EBACR1=0x3F; // EBM Address Control Register [5:0]
               // 7:6 = EBWR   = 00    = The Write Enables not mapped to the Mux Output
               // 5:0 = EBADDR = 11111 = The Address Lines are mapped to the Mux Output
  EBADCR=0xFF; // EBM Address/Data Control Register
               // Address Lines D13:D6 are Mapped to the Mux Output
  EBACR2=0x1F; // EBM Address Control Register [21:14]
               // 7:0 = EBADDR[21:14] = 0x1F = The address Lines mapped to Mux Output
  EBACR3=0x00; // EBM Address Control Register [29:22]
               // 7:0 = EBADDR[29:22] = The address Lines mapped to the Mux Output

  //The next Register Maps the 8 bit Data to the Mux Output
  EBDCR=0xFF; // EBM Data Control Register D7:D0
              // Data Lines D7:D0 are Mapped to the Mux Output
```

3.2 Memory Map Setup

```
// added to the low level init file with the memory mapping code
// activate Expansion bus at 0x00500000 set size to 512KB

  MCBAHR2 = 0x0050; //EBM RAM base addr at 0x00500000
  MCBALR2 = 0x0050; // Size of 512KB

//Bits 7:4 control the wait states therefore only 0xF wait states are available.
//Choose only one of the following for generating internal Wait States.
//  SMCR5 = 0x0004; //8-bit data width/External/Big Endian/1 wait states
//  SMCR5 = 0x0014; //8-bit data width/External/Big Endian/1 wait states
//  SMCR5 = 0x0024; //8-bit data width/External/Big Endian/2 wait states
//  SMCR5 = 0x0034; //8-bit data width/External/Big Endian/3 wait states
//  SMCR5 = 0x0044; //8-bit data width/External/Big Endian/4 wait states
```

4 EBM Timing Examples

The SYSCLK is set to 30 MHz for this example.

```
// The following code sets the SYSCLK to 30 MHz with a 7.5MHz Crystal
// and the PLL Enabled
PCR = CLKDIV_6;            // ICLK = SYSCLK/6
PCR |= PENABLE;            // enable peripherals
GCR = ZPLL_CLK_DIV_PRE_2;  // On B1M Development board 7.5 Mhz crystal * 8 / 2 = 30 Mhz
```

The SYSCLK is routed to CLKOUT.

```
// R34 on the B1M EVM may not be connected
// Therefore SYSCLK will have to be monitored at the resistor pad.
// The following code routes the SYSCLK to the CLKOUT
CLKCNTL = CLKSR_SYSCLK + CLKDIR + CLKDOUT + LPM_RUN;  // Send SYSCLK to CLKOUT Pin
dummy = CLKCNTL;  //Dummy Write required when changing CLKCNTL
__no_operation();
```

4.1 Screen Shot With One Internal Wait State

There is always a minimum of one wait state. Therefore, bits 7:4 of SMCR5 give the same results when set to 0 or 1.

In Figure 1, the yellow line is the SYSCLK running at 30 MHz, the blue line is the output enable ($\overline{OE}$), and the violet line is the chip enable ($\overline{CS}$). The $\overline{CS}$ occurs 33.2 ns before $\overline{OE}$, regardless of the number of wait states.

```
SMCR5 = 0x0004; //8-bit data width/External/Big Endian/1 wait states
SMCR5 = 0x0014; //8-bit data width/External/Big Endian/1 wait states
```

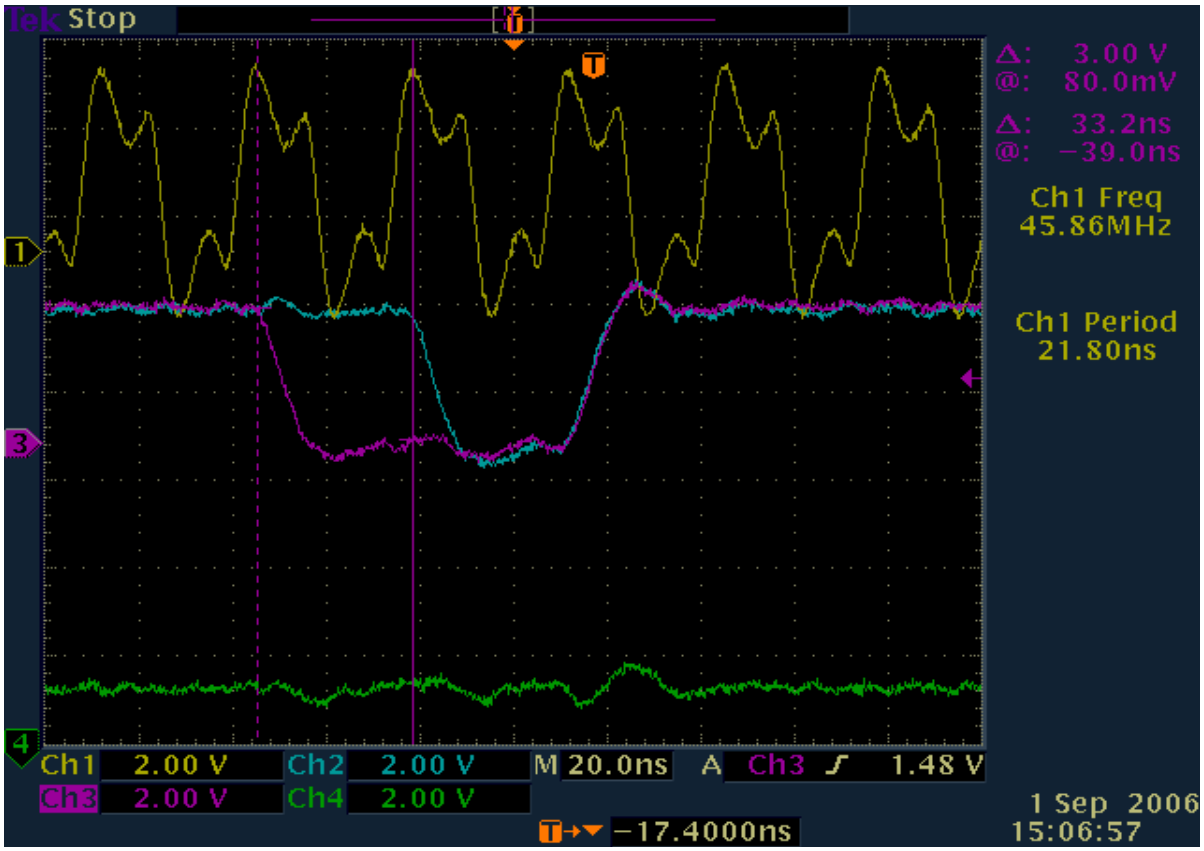


Figure 1. One Internal Wait State

4.2 Screen Shot With Two Internal Wait States

In Figure 2, the yellow line is the SYSCLK running at 30 MHz, the blue line is the output enable ($\overline{OE}$), and the violet line is the chip enable ($\overline{CS}$). The $\overline{CS}$ occurs 33.2 ns before $\overline{OE}$, regardless of the number of wait states.

```
SMCR5 = 0x0024; //8-bit data width/External/Big Endian/2 wait states
```

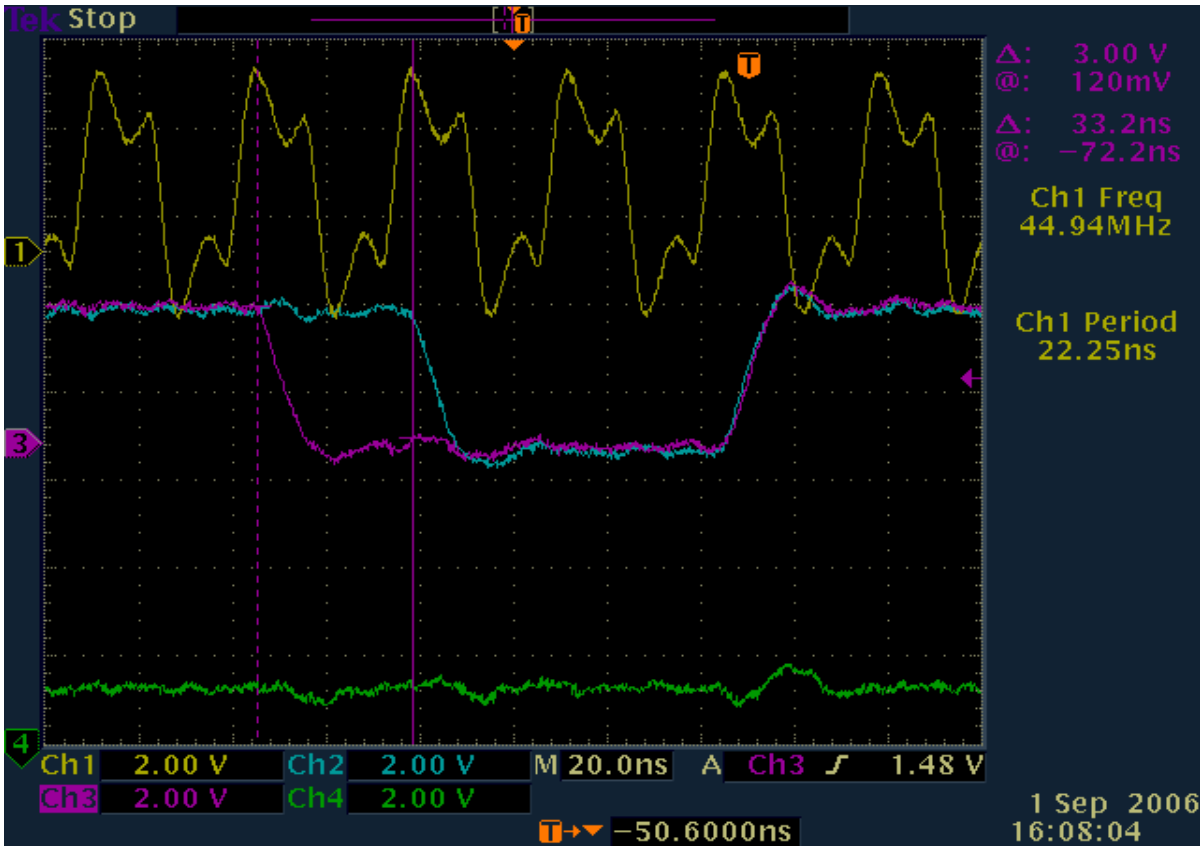


Figure 2. Two Internal Wait States

4.3 Screen Shot With Three Internal Wait States

In Figure 3, the yellow line is the SYSCLK running at 30 MHz, the blue line is the output enable ($\overline{OE}$), and the violet line is the chip enable ($\overline{CS}$). The $\overline{CS}$ occurs 33.2 ns before $\overline{OE}$, regardless of the number of wait states.

```
SMCR5 = 0x0034; //8-bit data width/External/Big Endian/3 wait states
```

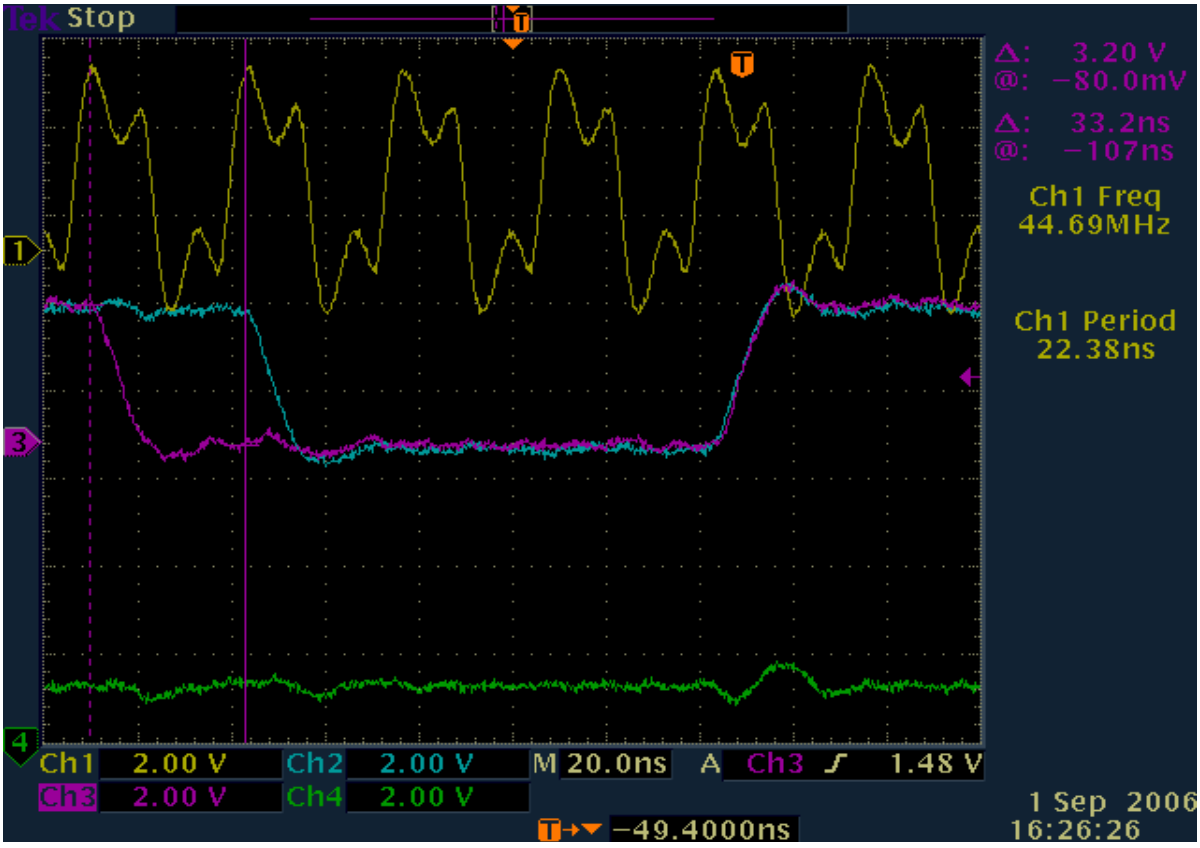


Figure 3. Three Internal Wait States

4.4 Screen Shot With Four Internal Wait States

In Figure 4, the yellow line is the SYSCLK running at 30 MHz, the blue line is the output enable ($\overline{OE}$), and the violet line is the chip enable ($\overline{CS}$). The $\overline{CS}$ occurs 33.2 ns before $\overline{OE}$, regardless of the number of wait states.

```
SMCR5 = 0x0044; //8-bit data width/External/Big Endian/4 wait states
```

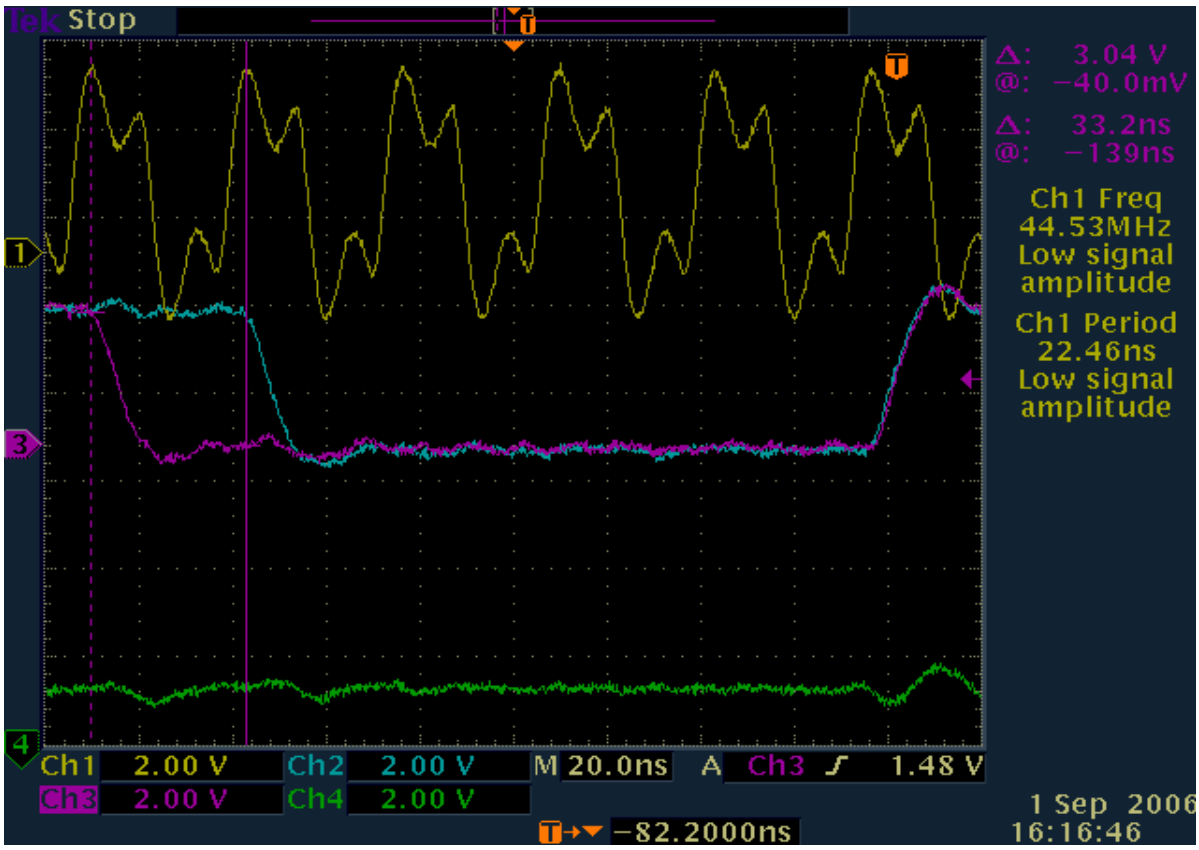


Figure 4. Four Internal Wait States

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.

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
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
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
Low Power Wireless	www.ti.com/lpw	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