Application Report
SLAA116A–December 2000–Revised March 2018

Using PWM Timer_B as a DAC

Mike Mitchell

MSP430 Applications

ABSTRACT

This application report describes how to simultaneously create a sine wave, a ramp, and a DC level with pulse-width modulated (PWM) signals from Timer_B on the MSP430™ ultra-low-power family of microcontrollers. PWM signals are often used to create analog signals in embedded applications. This report shows how to create both AC signals and DC levels with PWM outputs. For examples that implement the methods described in this report, see Dual-Output 8-Bit PWM DAC Using Low-Memory MSP430™ MCUs and Voice Band Audio Playback Using a PWM DAC.

Contents

List of Figures

Trademarks

MSP430 is a trademark of Texas Instruments.
All other trademarks are the property of their respective owners.

1 Introduction

Many embedded microcontroller applications require generation of analog signals. Sometimes an integrated or stand-alone digital-to-analog converter (DAC) is used for this purpose. However, PWM signals can often be used for generating the required analog signals. PWM signals can be used to create both DC and AC analog signals. The report below discusses using a PWM timer as a DAC and shows an example of simultaneously creating a sinusoid, a ramp, and a DC level and adding the DC level and sine wave to produce an offset AC signal. This report uses the PWM timer Timer_B. Timer_A could also be used in a similar manner.

2 Theory of Operation

A PWM signal is a digital signal with fixed frequency and variable duty cycle. Figure 1 shows an example of a PWM signal. If the duty cycle of the PWM signal is varied with time, and the PWM signal is filtered, the output of the filter is an analog signal. Figure 2 shows the block diagram for a PWM DAC employing this technique. PWM DACs can be used to generate sinusoids, ramp waves, and dc levels. In fact, some speech processors from Texas Instruments use PWM signals to generate speech for their applications.

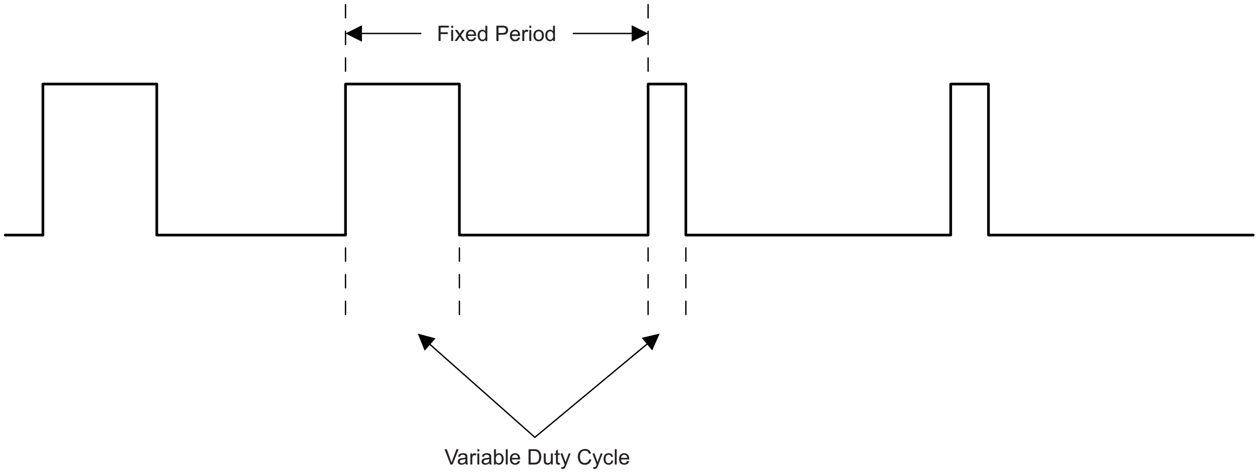

Figure 1. PWM Signal

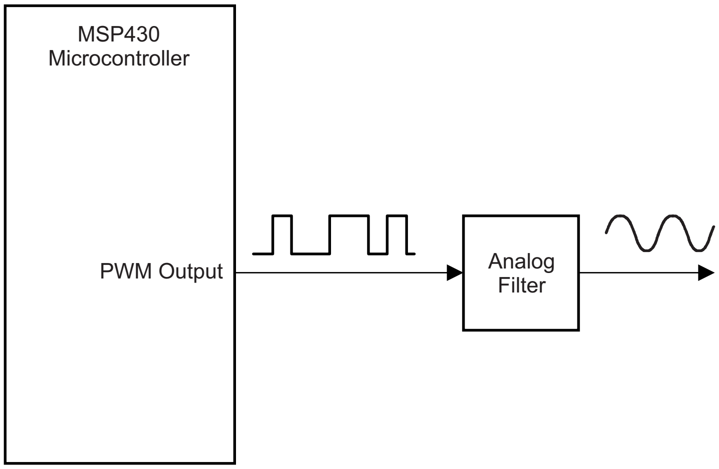

Figure 2. PWM DAC Block Diagram

2.1 Resolution

The resolution of a PWM DAC constructed with Timer_B is equivalent to the length of the counter, which is usually the value placed in CCR0. The LSB of the PWM DAC is one count and the resolution is the total number of counts:

$$R_{COUNTS} = L_{COUNTS} \tag{1}$$

where

- R_{COUNTS} = Resolution in counts
- L_{COUNTS} = Length of counter in counts

For example, this report implements an 8-bit DAC, so the length of the counter is 8 bits or 256 counts.

In more general terms, the resolution of a PWM DAC constructed with a PWM timer and a filter is equivalent to the resolution of the PWM signal used to create the DAC. The resolution of the PWM signal is then dependant on the length of the counter and the smallest duty-cycle change the PWM counter can make. The resolution is calculated by Equation 2.

$$R_{COUNTS} = \frac{L}{C}$$

where

- R_{COUNTS} = Resolution in counts
- L = Length of counter in counts
- C = Smallest duty-cycle change in counts (2)

Expressed as number of bits, the resolutions is calculated by Equation 3 or Equation 4.

$$R_{BITS} = \log_2\left(R_{COUNTS}\right) = \log_2 \frac{L}{C} = \frac{\ln\left(\frac{L}{C}\right)}{\ln(2)} \quad (3)$$

$$R_{BITS} = \log_2\left(R_{COUNTS}\right) = \frac{\ln\left(R_{COUNTS}\right)}{\ln(2)} \quad (4)$$

For example, if a PWM counter has a length of 512 counts and can vary the duty cycle by a minimum of 2 counts, the resolution in counts of the resulting PWM DAC is calculated by Equation 5.

$$R_{COUNTS} = \frac{L}{C} = \frac{512}{2} = 256 \quad (5)$$

And the resolution in bits is calculated by Equation 6.

$$R_{BITS} = \log_2(256) = \frac{\ln(256)}{\ln(2)} = 8\text{bits} \quad (6)$$

2.2 Frequency

The frequency required for the PWM output signal is equivalent to the update rate of the DAC, because each change in PWM duty cycle is the equivalent of one DAC sample. The frequency required for the PWM timer depends on the required PWM signal frequency and the desired resolution, as shown in Equation 7.

$$f_{clock} = f_{PWM} \times 2^n$$

where

- f_{clock} = Required PWM timer frequency
- f_{PWM} = PWM signal frequency, which is the DAC update rate
- n = Desired resolution of the DAC in bits (7)

The examples in this report show an 8-bit PWM DAC and simultaneously generating a 250-Hz sine wave and a 125-Hz ramp. The desired sampling rate for this example is 8 kHz (32 samples for each sine wave cycle (16x oversampled), and 64 samples for each ramp cycle (32x oversampled). This results in a required PWM signal frequency of 8 kHz and a required PWM clock frequency of 2.048 MHz.

It is usually best for the PWM signal frequency to be much higher than the desired sine wave frequency or the desired bandwidth of signals to be produced. Generally, the higher the PWM frequency, the lower the order of filter required and the easier it is to build a suitable filter.

2.3 MSP430 Resources Used

See Dual-Output 8-Bit PWM DAC Using Low-Memory MSP430™ MCUs and Voice Band Audio Playback Using a PWM DAC and their accompanying example code for a more detailed view of the resources used on the MSP430 MCU in each application.

Typical resources include:

- 32768-Hz crystal oscillator

- On-chip digitally controlled oscillator (DCO)
- SMCLK and MCLK
- Timer_A used to calibrate the DCO (optional)
- Timer_B or Timer_A to generate PWM signals

2.4 Circuit Diagram and Signals

Figure 3 shows a typical circuit diagram for a PWM DAC. Figure 4 shows the AC signals, and Figure 5 shows the the DC value with its PWM signal.

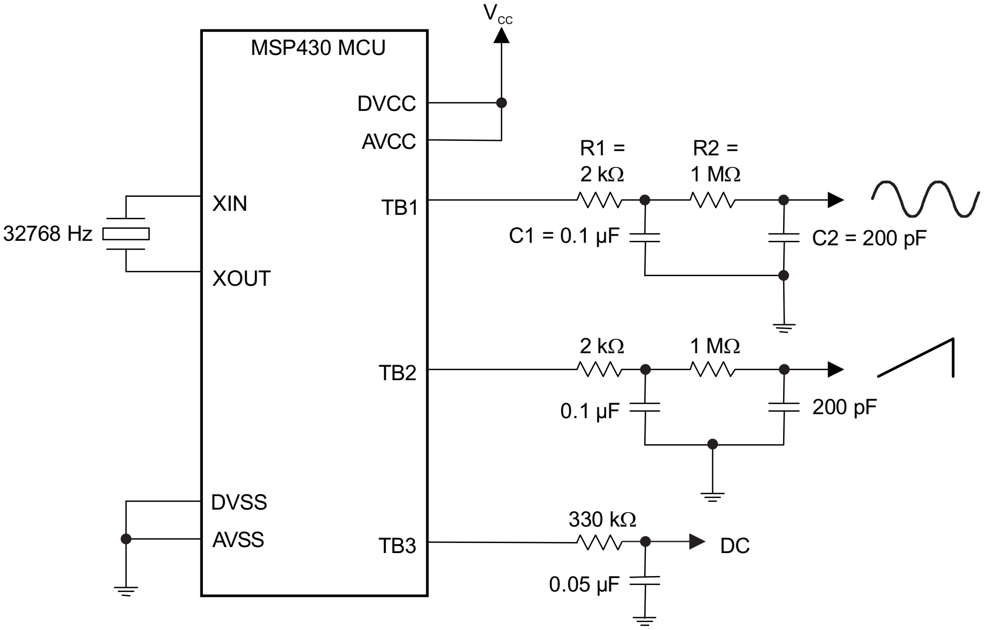

Figure 3. Typical Circuit Diagram

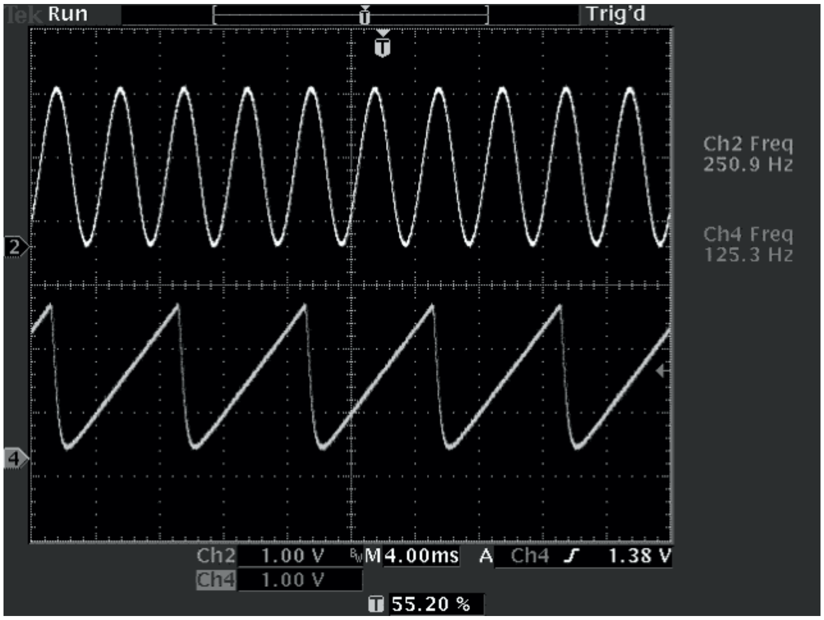

Figure 4. AC Signals

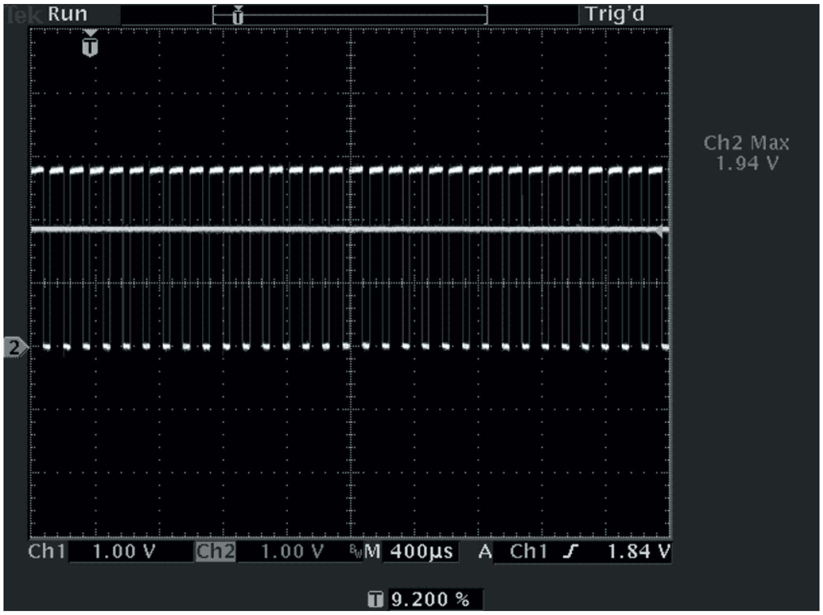

Figure 5. DC Signal

The sine wave produced by this example uses 32 samples per cycle. The sample values are contained in a table at the beginning of the program. A pointer is used to point to the next value of the sine table so that at the end of each PWM cycle, the new value of the sine wave is written to the capture/compare register of the PWM timer.

The ramp in this example does not require a table of data values. Rather, it is generated by incrementing the duty cycle each cycle of the PWM signal until the maximum is reached and then starting over at the minimum duty cycle. This gradual increase in PWM signal duty cycle results in a ramp voltage when it is filtered.

The DC level in this example is set by setting the value of the PWM signal duty cycle and never changing it. The DC level is directly proportional to the value of the duty cycle of the PWM signal. Because the duty cycle of the PWM signal on TB3 does not change, a DC value results when the signal is filtered by the RC network.

2.5 Filter Requirements

Figure 3 shows the reconstruction filters used for each signal in this example. The filter for the AC signals is a two-pole stacked-RC filter. This filter was chosen for its simplicity and lack of active components for low-power designs. This choice requires a higher sampling rate than would be required if the filter were a higher order. With this type of filter, TI recommends using at least 16x oversampling for the DAC.

The cutoff frequency of the filter can be calculated by Equation 8.

$$f_C = \frac{1}{2\pi RC}$$

where

- R1C1 = R2C2 = RC (8)

The filter gives better response when R2 is much greater than R1. Also, setting the cutoff frequency too close to the bandwidth edge causes significant attenuation. To reduce the amount of attenuation caused by the filter, set the cutoff frequency above the bandwidth edge, but much less than the frequency of the PWM signal.

The filter used for the DC value is used for charge storage rather than AC signal filtering. Therefore, a simple single-pole RC filter is used.

2.6 Adding the DC and AC Signals Together

The signals produced by the PWM DAC may be added together. The circuit in Figure 6 shows the DC offset of 2/3 V being added to the sine wave to produce an offset sine wave. In addition, the sine wave is attenuated by approximately 2/3 to allow it to be moved up or down on the DC offset without clipping the summing amplifier. This produces the 1-V sine wave on a DC offset of approximately 2 V in Figure 7. The PWM signal is shown in the background for reference.

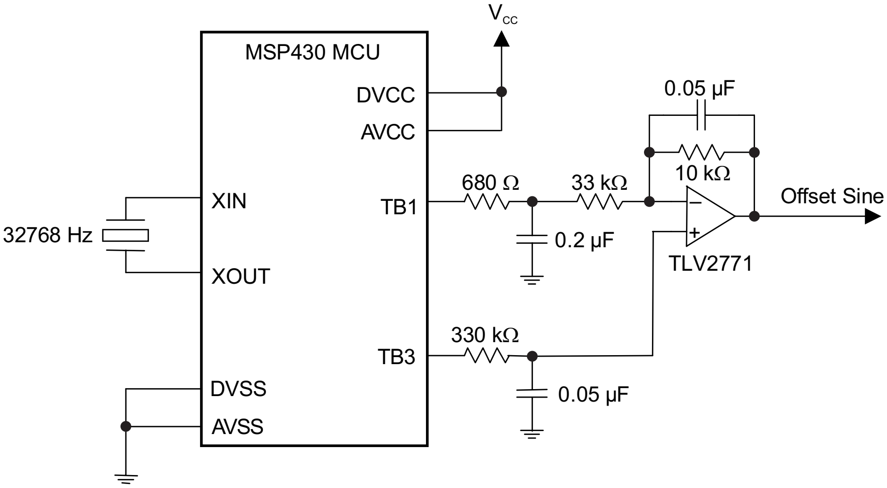

Figure 6. Summing Circuit

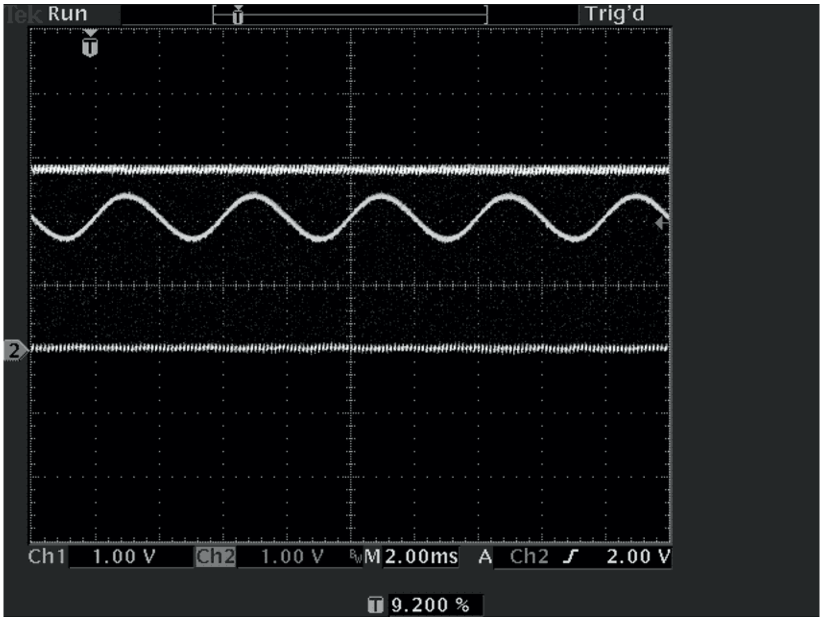

Figure 7. Offset Sine Wave

Adding the signals as shown in this example provides the flexibility to adjust the AC and DC signals separately and easily through the PWM outputs. For example, the sine wave may be kept constant, while the DC offset may be moved up or down by changing the PWM duty-cycle used to create the DC value. Note that the filter for the sine wave changed slightly. Because the summing amplifier was added to the circuit to add the offset to the sine wave, the filter required for the sine wave was integrated into the amplifier circuit. Other active filters and summing amplifier circuits could also be used to achieve the same results.

2.7 DCO Calibration

The DCO on MSP430 MCUs is an RC-type oscillator and exhibits RC-oscillator characteristics such as inaccurate frequency settings and drift. However, because the DCO is digitally controllable, it can be tuned to an accurate frequency using a stable known frequency source such as the 32768-Hz crystal oscillator. Some MSP430 MCUs have digital logic known as a frequency lock loop (FLL) to perform this function automatically, while other MSP430 MCUs do not. See the clock system chapter of the appropriate family user's guide for a more detailed discussion on the DCO and clock systems.

In devices that do not contain an FLL, the DCO must be calibrated with software and an external frequency source to produce a known stable frequency. TI provides code examples that implement a software FLL for applicable devices in MSP430Ware for MSP Microcontrollers. Additionally, Section 3.3.2 of UCS10 Guidance provides a detailed explanation and example of a software FLL routine. If using an MSP430x1xx device, Controlling the DCO Frequency of the MSP430x11x provides an specific implementation for MSP430x1xx devices.

An alternative to the DCO calibration routine would be to use an external crystal of the required PWM timer frequency. See the device-specific data sheet to determine the external crystal capabilities of your MSP430 MCU variant.

2.8 Software Examples

Examples that incorporate the PWM DAC methods described in this document can be found in Dual-Output 8-Bit PWM DAC Using Low-Memory MSP430™ MCUs and Voice Band Audio Playback Using a PWM DAC.

3 References

1. Controlling the DCO Frequency of the MSP430x11x
2. MSP430Ware for MSP Microcontrollers
3. UCS10 Guidance
4. Dual-Output 8-Bit PWM DAC Using Low-Memory MSP430™ MCUs
5. Voice Band Audio Playback Using a PWM DAC

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from December 8, 2000 to March 6, 2018	**Page**
• Updates throughout to refer to more recent software examples and to remove assembly example code	1

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated (‘TI”) technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, “TI Resources”) are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI’s provision of TI Resources does not expand or otherwise alter TI’s applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources 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.

TI RESOURCES ARE PROVIDED “AS IS” AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI’s standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2018, Texas Instruments Incorporated