Analog Engineer's Circuit

Negative Voltage Margining and Scaling Circuit With Voltage Output Smart DAC

Smart DAC

Katlynne Jones

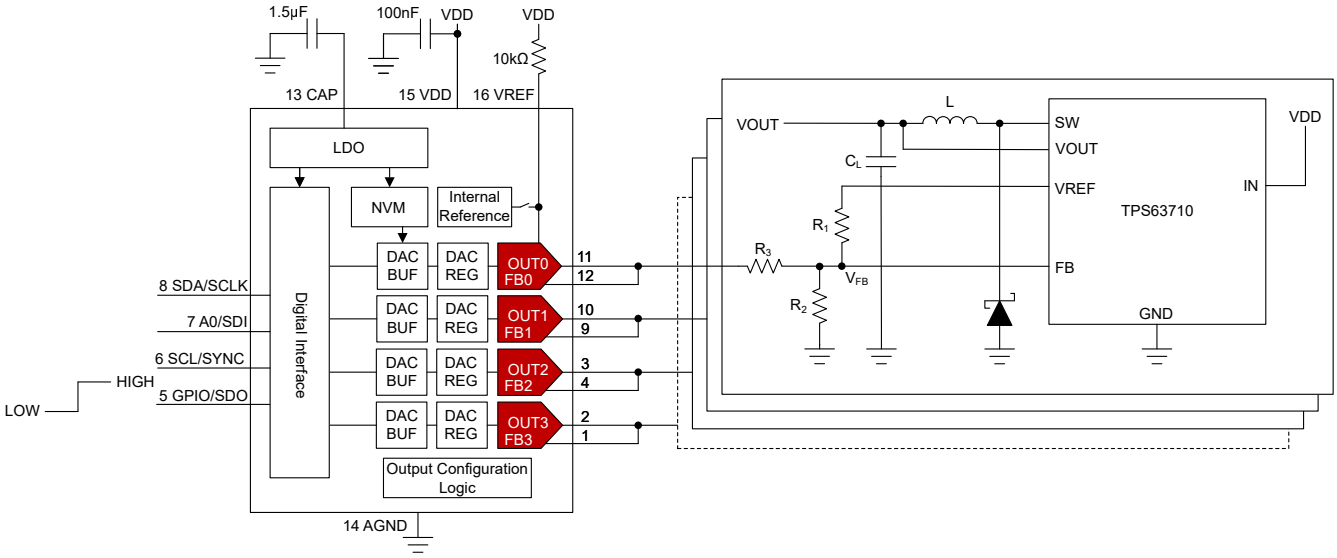
Design Objective

Key Input Parameter	Key Output Signal	Recommended Device
SPI or I^2C communication to control DAC voltage output	0-V to 1.2-V programmable current sink, –3.45 V ±26% DC/DC output	DAC63204W, DAC53204W, DAC63004W, DAC53004W, DAC63204, DAC53204, DAC43204, TPS63710

Objective: Margin a DC/DC output to ±26% the nominal value.

Design Description

This circuit uses a four-channel buffered voltage output DAC to margin an inverting step-down DC/DC converter. A voltage margining circuit is used to trim, scale, or test the output of a power converter. Adjustable power supplies, such as low dropout regulators (LDOs), DC/DC converters, or switch-mode power supplies (SMPS) provide a feedback (FB) pin that is used, along with a resistive voltage divider, to control the desired output. A precision smart DAC, such as the DAC63204W or DAC53204W (DACx3204W), provides linear control of the power supply output by using a series resistor to inject current into the voltage divider. The DACx3204 have a general-purpose input (GPI) pin that allows the DAC output to be toggled between a high and low voltage output. This allows the DC/DC to be toggled within ±26% of the nominal output value. All register settings are saved using the integrated non-volatile memory (NVM), enabling the device to be used without runtime software, even after a power cycle or reset. This circuit can be used in applications such as LIDAR, virtual reality headsets, and OLED TVs.



Design Notes

1. The *DACx3204W 12-Bit and 10-Bit, Quad Voltage and Current Output Smart DACs With Auto-Detected I2C, SPI, or PMBus® Interface in DSBGA Package* data sheet recommends using a 100-nF decoupling capacitor for the VDD pin and a 1.5-µF or greater bypass capacitor for the CAP pin. The CAP pin is connected to the internal LDO. Place these capacitors close to the device pins.
2. Connect a 100-nF capacitor from the VREF pin to GND if the external reference is used. Ramp up the external reference after VDD. Connect a pullup resistor from the VREF pin to VDD if the external reference is not used. This example uses the internal reference and the VREF pin is pulled up to VDD with a 10-kΩ resistor.
3. The output voltage (V_{OUT}) of the TPS63710 when the DAC63204W is not connected or the current through the series resistor, R_3, is 0 A set by resistors R_1 and R_2. The TPS63710 uses an internal –700-mV reference voltage (V_{FB}) at the FB pin to determine V_{OUT}. The current through R_3 is 0 A when the DAC63204W output voltage (V_{DAC}) equals V_{FB}. The DAC63204W cannot output a negative voltage, so this design assumes that the DAC63204W output is always positive, and the TPS63710 is at the nominal output voltage when the DAC63204W is at midscale.
4. Choose R_3 so that V_{DAC} **>** –0.3 V when the DAC63204W is set to power-down mode. When the DAC63204W is configured in 10 kΩ to GND power-down mode, the 10-kΩ resistance creates a resistor divider with R_3. R_3 is chosen to be 200 kΩ in this example, making V_{DAC} equal to –0.038 V when in 10 kΩ to GND power down. Do not use Hi-z power-down mode when a negative voltage is connected to V_{DAC}.
5. Choose the current through R_2 (I_{R2}) so that the bias current into the FB pin of the TPS63710 is negligible. R_2 is calculated using:

$$R_2 = \frac{V_{FB}}{I_{R2}}$$

I_{R2} is chosen to be 5.2 µA. The TPS63701 has an internal gain factor of 1/0.9 which makes the effective V_{FB} –778 mV. R_2 is calculated to be:

$$R_2 = \frac{|-778\ \text{mV}|}{5.2\ \mu\text{A}} = 150\ \text{k}\Omega$$

6. The nominal TPS63710 V_{OUT} is chosen to be –3.45 V when V_{DAC} is at midscale, or 0.91 V. The current sourced from the DAC63204W output is calculated by:

$$I_{DAC} = \frac{V_{DAC} - V_{FB}}{R_3}$$

$$I_{DAC} = \frac{910\ \text{mV} + 778\ \text{mV}}{200\ \text{k}\Omega} = 8.44\ \mu\text{A}$$

R_1 can be calculated to achieve the desired nominal V_{OUT} using:

$$R_1 = \frac{V_{FB} - V_{OUT}}{I_{R2} - I_{DAC}}$$

$$R_1 = \frac{-0.778\ \text{V} + 3.45\ \text{V}}{5.2\ \mu\text{A} + 8.44\ \mu\text{A}} = 196\ \text{k}\Omega$$

7. The DAC63204W sinks or sources additional current through R_1 by adjusting V_{DAC} to achieve the desired margin. V_{DAC} is calculated by:

$$V_{DAC} = \left(I_{R2} - \frac{V_{OUT} - V_{FB}}{R_1}\right) \times R_3 + V_{FB}$$

$V_{DAC,MAX}$ and $V_{DAC,MIN}$ are configured to margin V_{OUT} by 26%. V_{OUT} low is –4.34 V and V_{OUT} high is –2.55 V

$$V_{DAC,\,MAX} = \left(-5.2\ \mu\text{A} - \frac{-4.34\ \text{V} + 0.778\ \text{V}}{196\ \text{k}\Omega}\right) \times 200\ \text{k}\Omega - 0.778\ \text{V} = 1.82\ \text{V}$$

$$V_{DAC,\,MIN} = \left(-5.2\ \mu\text{A} - \frac{-2.55\ \text{V} + 0.778\ \text{V}}{196\ \text{k}\Omega}\right) \times 200\ \text{k}\Omega - 0.778\ \text{V} = 0\ \text{V}$$

8. The DAC codes for $V_{DAC,MAX}$ and $V_{DAC,MIN}$ are stored in the DAC-X-MARGIN-HIGH and DAC-X-MARGIN-LOW registers. $V_{DAC,NOM}$ is stored in the DAC-X-DATA register. The codes programmed to these registers, in decimal, is calculated using:

$$\text{DAC\_CODE} = \frac{V_{DAC} \times 2^{12}}{V_{REF}}$$

This design uses the internal 1.21-V reference with a gain of ×1.5 giving a full-scale voltage of 1.82 V. The equations become:

$$\text{DAC\_MARGIN\_HIGH} = \frac{1.82\ \text{V} \times 2^{12}}{1.82\ \text{V}} = 4096\text{d}$$

$$\text{DAC\_DATA} = \frac{0.91\ \text{V} \times 2^{12}}{1.82\ \text{V}} = 2048\text{d}$$

$$\text{DAC\_MARGIN\_LOW} = \frac{0\ \text{V} \times 2^{12}}{1.82\ \text{V}} = 0\text{d}$$

The maximum output code for a 12-bit device is 4095 so the $V_{DAC,MAX}$ becomes 1.819 V.

9. The TPS63710 requires that $V_{IN} \geq |V_{OUT}| / 0.7$. The max V_{OUT} for this application is –4.34 V, so the minimum V_{IN} is 6.2 V. 10 V is used in this design.
10. Using a 1.21-V reference with a ×1.5 gain and the 12-bit DAC63204W, the LSB size, or step size between each code, is about 443 µV. Using the lowest reference voltage possible decreases the LSB size and thus maximizes the resolution of $V_{DAC,MAX}$ and $V_{DAC,MIN}$.
11. The DAC63204W has a programmable slew-rate feature. The programmable slew is configured with the CODE-STEP-X and SLEW-RATE-X fields in the DAC-X-FUNC-CONFIG register. The programmable slew is only available when toggling between two values stored in the DAC-X-MARGIN-HIGH and DAC-X-MARGIN-LOW registers.

    CODE-STEP-X defines the number of LSB steps used to transition from the starting code to the final output code. SLEW-RATE-X defines the time-period for each code step. The slew time is calculated by:

$$t_{SLEW} = \text{SLEW\_RATE} \times \text{CEILING}\left(\frac{\text{MARGIN\_HIGH\_CODE} - \text{MARGIN\_LOW\_CODE}}{\text{CODE\_STEP}} + 1\right)$$

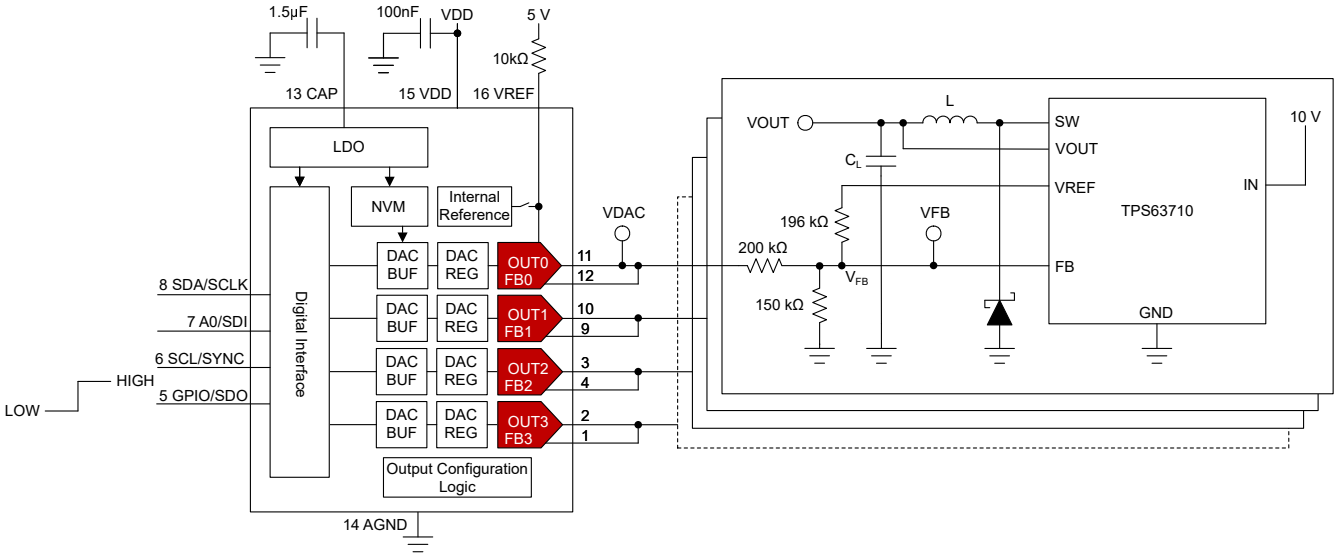
    This application uses a margin high code of 4095, margin low code of 0, SLEW-RATE of 1282 µs/LSB and a CODE-STEP of 1 LSB to achieve a 5.25-s slew time:

$$t_{SLEW} = 1282\ \left(\mu\text{s/LSB}\right) \times \text{CEILING}\left(\frac{4095 - 0}{1\ \text{LSB}} + 1\right) = 5.25\ \text{s}$$

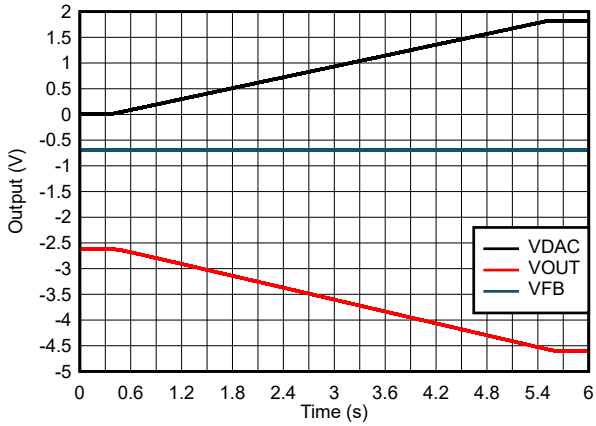
12. The GPIO pin can be configured as a digital input to switch the DAC63204W output between the margin high and margin low outputs with a programmable slew. The GPI-EN bit in the GPIO-CONFIG register enables the GPIO pin as an input. The GPI-CH-SEL field selects which channels are controlled by the GPI. The GPI-CONFIG field selects the GPI function. Write 0b1010 to the GPI-CONFIG field to configure the GPIO pin to trigger margin-high or margin-low functions.
    - A high on GPI sets the DAC output to $V_{DAC,MAX}$ and the TPS63710 V_{OUT} to low, or –4.34 V. A low on GPI sets the DAC output to $V_{DAC,MIN}$ and the TPS63710 V_{OUT} to high, or –2.55 V.
13. The DAC63204W can be programmed with the initial register settings described in the Register Settings section using I^2C or SPI. Save the initial register settings in the NVM by writing a 1 to the NVM-PROG field of the COMMON-TRIGGER register. After programming the NVM, the device loads all registers with the values stored in the NVM after a reset or a power cycle.

Design Results

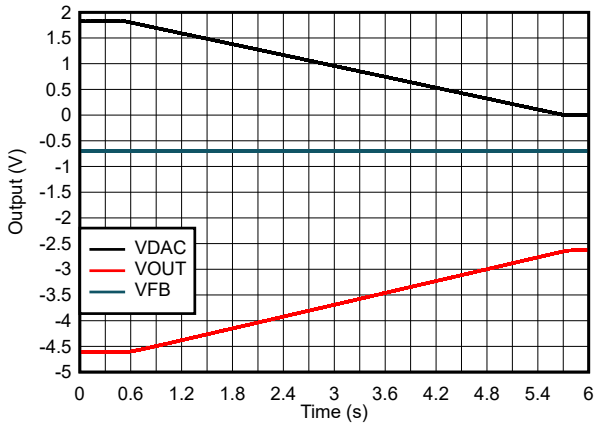
This schematic is used for the following design results of the DAC63204W. The V_{DAC}, V_{OUT}, and V_{FB} signals are measured on an oscilloscope at the test points marked on the schematic.



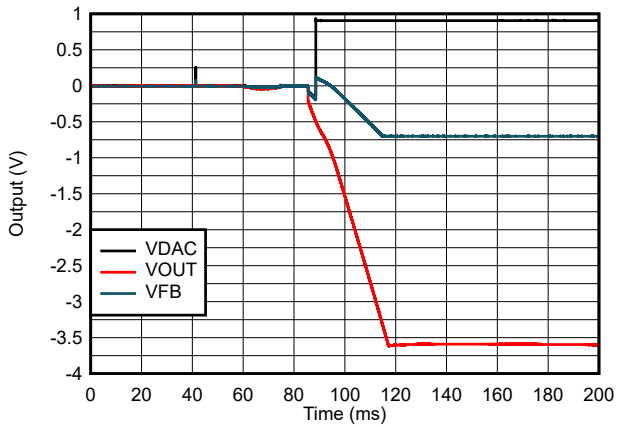
This plot shows the low-to-high transition of the DAC63204W output with the 5.25-s slew configured using the settings discussed in the Design Notes. The V_{DAC} output voltage slews from 0 V to 1.82 V which causes the TPS63710 V_{OUT} voltage to slew from –2.55 V to –4.34 V.



This plot shows the low-to-high transition of the DAC63204W output with the 5.25-s slew. The V_{DAC} output voltage slews from 1.82 V to 0 V which causes the TPS63710 V_{OUT} voltage to slew from –4.34 V to –2.55 V.



This plot shows the start-up behavior of the circuit. The 10-V TPS63710 supply and 5-V DAC63204W supply are switched on at the same time. The V_{DAC} output starts up to the nominal voltage of 910 mV. The TPS63710 V_{OUT} ramps to the nominal output of –3.45 V as the V_{FB} reference voltage starts up.



Register Settings

The following table shows an example register map for this application. The values given here are for the design choices made in the Design Notes section.

Register Settings for DAC63204W

Register Address	Register Name	Setting	Description
0x1F	COMMON-CONFIG	0x1249	[15] 0b0: Write 0b0 to set the window-comparator output to a non-latching output
			[14] 0b0: Device not locked
			[13] 0b0: Fault-dump read enable at address 0x00
			[12] 0b1: Enables the internal reference
			[11:10] 0b00: Powers up VOUT3
			[9] 0b1: Powers down IOUT3
			[8:7] 0b00: Powers up VOUT2
			[6] 0b1: Powers down IOUT2
			[5:4] 0b00: Powers up VOUT1
			[3] 0b1: Powers down IOUT1
			[2:1] 0b00: Powers up VOUT0
			[0] 0b1: Powers down IOUT0
0x24	GPIO-CONFIG	0x01F5	[15] 0b0: Glitch filter disabled for GP input
			[14] 0b0: Don't care
			[13] 0b0: Disable output mode for GPIO pin
			[12:9] 0b0000: Selects the STATUS function setting mapped to GPIO as output
			[8:5] 0b1111: Enables GPI function on all channels
			[4:1] 0b1010: GP input configured to trigger margin high or low
			[0] 0b1: Enables input mode for GPIO pin

Register Settings for DAC63204W (continued)

Register Address	Register Name	Setting	Description
0x20	COMMON-TRIGGER	0x0002	[15:12] 0b0000: Write 0b0101 to unlock the device
			[11:8] 0b0000: Write 0b1010 to trigger a POR reset
			[7] 0b0: LDAC is not triggered
			[6] 0b0: DAC clear is not triggered
			[5] 0b0: Don't care
			[4] 0b0: Fault-dump is not triggered
			[3] 0b0: PROTECT function not triggered
			[2] 0b0: Fault-dump read not triggered
			[1] 0b1: Write 0b1 to store applicable register settings to the NVM
			[0] 0b0: NVM reload not triggered. Write 0b1 to reload applicable registers with existing NVM settings
0x03, 0x09, 0x0F, 0x15	DAC-X-VOUT-CMP-CONFIG	0x0800	[15:13] 0b000: Don't care
			[12:10] 0b010: Selects internal reference with ×1.5 gain
			[9:5] 0x00: Don't care
			[4] 0b0: Set OUTx pins as push-pull in comparator mode
			[3] 0b0: Comparator output consumed internally
			[2] 0b0: FBx input has high-impedance in comparator mode
			[1] 0b0: Comparator output not inverted
			[0] 0b0: Disable comparator mode
0x06, 0x0C, 0x12, 0x18	DAC-X-FUNC-CONFIG	0x000D	[15] 0b0: DAC-X clear mode set to zero-scale
			[14] 0b0: DAC-X output updates immediately after a write command
			[13] 0b0: Do not update DAC-X with broadcast command
			[12:11] 0b00: Phase set to 0°
			[10:8] 0b000: Selects sine wave mode
			[7] 0b0: Enable linear slew
			[6:4] 0b000: Selects 8 LSB CODE-STEP
			[3:0] 0xD: Selects 8 µs/step SLEW-RATE
0x01, 0x07, 0x0D, 0x13	DAC-X-MARGIN-HIGH	0xFFF0	[15:4] 0xFFF: 12-bit margin high code
			[3:0] 0x0: Don't care
0x02, 0x08, 0x0E, 0x14	DAC-X-MARGIN-LOW	0x0000	[15:4] 0x000: 12-bit margin low code
			[3:0] 0x0: Don't care

Pseudocode Example

The following shows a pseudocode sequence to program the initial register values to the NVM of the DAC63204W. The values given here are for the design choices made in the Design Notes section.

Pseudocode Example for DAC63204W

```
1:  //SYNTAX: WRITE <REGISTER NAME (Hex code)>, <MSB DATA>, <LSB DATA>
2:  //Set gain setting to 1.5x internal reference (1.8 V) (repeat for all channels)
3:  WRITE DAC-0-VOUT-CMP-CONFIG(0x3), 0x08, 0x00
4:  //Power-up voltage output on all channels and enable the internal reference
5:  WRITE COMMON-CONFIG(0x1F),0x12, 0x49
6:  //Configure GPI for Margin-High, Low trigger for all channels
7:  WRITE GPIO-CONFIG(0x24), 0x01, 0xF5
8:  //Set slew rate and code step (repeat for all channels)
9:  //CODE_STEP: 1 LSB, SLEW_RATE: 1282 µs/step
10: WRITE DAC-0-FUNC-CONFIG(0x06), 0x00, 0x0D
11: //Write nominal DAC code (repeat for all channels)
12: //For a 1.8-V output range, the 12-bit hex code for 0.9 V is 0x800. With 16-bit left alignment,
13: this becomes 0x8000
14: WRITE DAC-0-DATA(0x19), 0x80, 0x00
15: //Write DAC margin high code (repeat for all channels)
16: //For a 1.8-V output range, the 12-bit hex code for 1.8 V is 0xFFF. With 16-bit left alignment,
17: this becomes 0xFFF0
18: WRITE DAC-0-MARGIN-HIGH(0x01), 0xFF, 0xF0
19: //Write DAC margin low code (repeat for all channels)
20: //The 12-bit hex code for 0 V is 0x000. With 16-bit left alignment, this becomes 0x0000
21: WRITE DAC-0-MARGIN-LOW(0x02), 0x00, 0x00
21: //Save settings to NVM
22: WRITE COMMON-TRIGGER(0x20), 0x00, 0x02
```

Design Featured Devices

Device	Key Features	Link
DAC63204W	4-channel, 12-bit, VOUT and IOUT smart DAC with I^2C, SPI, and Hi-Z out during power off in DSBGA package	DAC63204W
DAC53204W	4-channel, 10-bit, VOUT and IOUT smart DAC with I^2C, SPI, and Hi-Z out during power off in DSBGA package	DAC53204W
DAC63004W	4-channel, 12-bit, VOUT and IOUT smart DAC with I^2C, SPI, and Hi-Z out during power off in DSBGA package	DAC63004W
DAC53004W	4-channel, 10-bit, VOUT and IOUT smart DAC with I^2C, SPI, and Hi-Z out during power off in DSBGA package	DAC53004W
TPS63710	Low Noise, 1 A Synchronous Inverting Buck Converter in 3x3 WSON Package	TPS63710

Find other possible devices using the Parametric search tool.

Design References

See *Analog Engineer's Circuit Cookbooks* for TI's comprehensive circuit library.

Additional Resources

- Texas Instruments, *DAC63004WCSP-Evaluation Module*
- Texas Instruments, *DAC63004WCSP-EVM User's Guide*
- Texas Instruments, Precision Labs - DACs
- Texas Instruments, *TPS63710EVM-811*

For direct support from TI Engineers, use the E2E community:

e2e.ti.com

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

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