ABSTRACT
This application report provides the necessary information to use the temperature sensor of the CC112X and CC120X families. The temperature sensor is based on a proportional to absolute temperature (PTAT) current from a bandgap cell fed to a resistor to generate a PTAT voltage. It is possible to read out the temperature information either as an analog voltage on a general-purpose input/output (GPIO) pin or using the on-chip analog-to-digital converter (ADC) to convert the voltage to a digital readout. This document is divided into two parts: the first part covers the analog readout and the second covers the digital readout.

Contents
1 Analog Readout .............................................................................................................. 2
2 Digital Readout ............................................................................................................. 6
3 References .................................................................................................................... 11

List of Figures
1 GPIO1 Voltage vs Temperature .......................................................................................... 3
2 Temperature Error Due to Different Temperature Coefficients After Single-Point Calibration .................................................................................................................... 4
3 Typical GPIO1 Measurements vs Supply Voltage ............................................................... 5
4 Register Values vs. Temperature ...................................................................................... 7

List of Tables
1 Register Settings for Temperature Sensor ........................................................................ 2
2 Typical Temperature Sensor Parameters ......................................................................... 2
3 Register Settings for Digital Readout ............................................................................. 6
4 Typical Parameters for Register Readout .......................................................................... 7
1 Analog Readout

1.1 Operation

The temperature sensor is activated using the register settings of Table 1, which makes the GBIAS output a single-ended voltage measurement on GPIO1.

<table>
<thead>
<tr>
<th>Register</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOCFG1</td>
<td>0x80</td>
</tr>
<tr>
<td>ATEST</td>
<td>0x2A</td>
</tr>
<tr>
<td>ATEST_MODE</td>
<td>0x0C</td>
</tr>
<tr>
<td>GBIAS1</td>
<td>0x07</td>
</tr>
</tbody>
</table>

Setting IOCFG1 to 0x80 configures the GPIO1 pad into analog mode (digital GPIO input and output is disabled). The remaining registers set up the ATEST (analog test) module to output the temperature value as a PTAT voltage on the GPIO1.

1.2 Temperature Sensor Parameters

<table>
<thead>
<tr>
<th>General Information</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature sensor fitted from</td>
<td>-40 to +85</td>
<td>°C</td>
</tr>
<tr>
<td>Effect of supply voltage deviance</td>
<td>1.17</td>
<td>mV/VDD-V</td>
</tr>
<tr>
<td>Effect of supply voltage deviance</td>
<td>0.44</td>
<td>°C/VDD-V</td>
</tr>
</tbody>
</table>

Changes in the supply voltage affect the voltage of the GPIO pin, and the supply voltage must be stable in order to get accurate temperature sensor readings.

<table>
<thead>
<tr>
<th>Technical Information</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>VDD - 2 V</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Typical output voltage @ 0 °C</td>
<td>727.42</td>
<td>mV</td>
</tr>
<tr>
<td>Typical output voltage @ 25 °C</td>
<td>793.73</td>
<td>mV</td>
</tr>
<tr>
<td>Temperature coefficient</td>
<td>2.6598</td>
<td>mV/°C</td>
</tr>
<tr>
<td>VDD - 3 V</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Typical output voltage @ 0 °C</td>
<td>728.55</td>
<td>mV</td>
</tr>
<tr>
<td>Typical output voltage @ 25 °C</td>
<td>794.78</td>
<td>mV</td>
</tr>
<tr>
<td>Temperature coefficient</td>
<td>2.6733</td>
<td>mV/°C</td>
</tr>
<tr>
<td>VDD - 3.6 V</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Typical output voltage @ 0 °C</td>
<td>730.62</td>
<td>mV</td>
</tr>
<tr>
<td>Typical output voltage @ 25 °C</td>
<td>796.94</td>
<td>mV</td>
</tr>
<tr>
<td>Temperature coefficient</td>
<td>2.6773</td>
<td>mV/°C</td>
</tr>
</tbody>
</table>
1.3 Calibration

As seen in Figure 1, the CC112X/CC120X temperature sensor voltage is highly linear, but for some devices there is an offset in the GPIO1 voltage from the typical (average) value that could potentially give an error of up to ±10°C in the temperature reading. In order to ensure accurate temperature sensor measurements, the sensor must be calibrated. There are two simple approaches depending on the required accuracy level: single- and two-point calibration.

![Figure 1. GPIO1 Voltage vs Temperature](image)

1.4 Single-Point Calibration

This is a simple and fast approach that can be applied for applications targeting approximately ±1°C accuracy within a limited temperature range around the temperature used for the single-point calibration, or approximately ±2°C accuracy across the -40°C to +85°C temperature range.

1.4.1 Performing Single-Point Calibration

The calibration should be performed at the center of the temperature range in which the device will operate. A given temperature, \( T \), will be given as:

\[
T = T_{\text{CALIBRATION}} + \frac{(V_{\text{MEASURED}} - V_{\text{CALIBRATION}})}{t_c}
\]  

- \( T_{\text{CALIBRATION}} \) is the temperature when the calibration is performed
- \( t_c \) is the temperature coefficient for the given supply voltage (see the typical temperature parameters in Table 2)
- \( V_{\text{MEASURED}} \) is the voltage of the GPIO1 pin at a given temperature
- \( V_{\text{CALIBRATION}} \) is the GPIO1 voltage at the calibration temperature

Performing a single-point calibration removes the error caused by the device-specific voltage offset seen in Figure 1. The temperature reading accuracy is then limited by the accuracy of the individual temperature coefficients as the typical temperature coefficient is used in Equation 1.
Figure 2 shows the maximum error in the temperature reading when using the lowest and highest temperature coefficients out of 30 devices from different processing corners.

- Approximately ±2°C accuracy is possible across the -40°C to +85°C temperature range with single-point calibration and using the typical temperature coefficient in Table 2.
- Approximately ±1°C accuracy is possible across the temperature range defined by $T_{\text{CALIBRATION}} \pm 25°C$ with single-point calibration and using the typical temperature coefficient in Table 2.

Figure 2. Temperature Error Due to Different Temperature Coefficients After Single-Point Calibration

1.4.2 Single-Point Calibration Example

A CC112X/CC120X device is operated using at 3 V supply voltage. The temperature coefficient is typically 2.673 mV/°C and for each degree Celsius increase in temperature the GPIO1 voltage increases by 2.673 mV.

The device is calibrated at room temperature (25°C), and the GPIO1 voltage is measured to be 793.0 mV. After changing the temperature, the GPIO1 voltage is measured to be 830.0 mV. This corresponds to a temperature $T$ of:

$$T = 25°C + \frac{(830 \text{ mV} - 793 \text{ mV})}{2.673 \text{ mV/°C}}$$
$$T = 25°C + 13.84°C = 38.84°C$$

(2)

1.5 Two-Point Calibration

If the application requires better accuracy than given by the single-point calibration, a two-point calibration must be used to correct for chip-to-chip variations in the temperature coefficients. As the sensor is highly linear, a two-point calibration will ensure high accuracy across the full temperature range of the chip.

1.5.1 Performing Two-Point Calibration

Choose two calibration temperatures more than 10°C apart, called T0 and T1, and set the reference voltage ($V_{\text{DD}}$) to what it will be in the final product.

NOTE: Changes in the voltage supply will influence the temperature sensor output.

Measure the output from the GPIO1 pin (V0 and V1) at the corresponding temperatures.

The temperature coefficient has a typical value of 2.673 mV/°C. The exact coefficient ($t_c$) for a given device is calculated as:

$$t_c = \frac{V_1 - V_0}{T_1 - T_0}$$

(3)
Using the exact coefficient, the measured voltage of the GPIO1 pin ($V_{\text{MEASURED}}$), the temperature ($T_0$) and the GPIO1 voltage ($V_0$) of the first calibration, the temperature, $T$, can be found as:

$$T = T_0 + \frac{(V_{\text{MEASURED}} - V_0)}{t_c}$$  

(4)

### 1.5.2 Two-Point Calibration Example

A CC112X/CC120X device is operated using a 3 V supply voltage, and will have a typical temperature coefficient of 2.673 mV/°C.

The device is calibrated at two temperatures: 0°C and 25°C ($T_0$ and $T_1$). The respective GPIO1 voltages are measured to be 743.379 mV and 808.312 mV ($V_0$ and $V_1$). The exact temperature coefficient $t_c$ is given as:

$$t_c = \frac{808.312 \text{ mV} - 743.379 \text{ mV}}{25^\circ \text{C} - 0^\circ \text{C}} = 2.5973 \text{ mV/}^\circ \text{C}$$

(5)

At a given temperature $T$, the GPIO1 voltage is measured to be 921.465 mV. This corresponds to:

$$T = 0^\circ \text{C} + \frac{(921.465 \text{ mV} - 743.379 \text{ mV})}{2.5973 \text{ mV/}^\circ \text{C}} = 68.57^\circ \text{C}$$

(6)

**NOTE:** Single-point calibration at 25°C, using the typical $t_c$ of 2.673 mV/°C, would in this case give a temperature reading of 67.33°C, which would have an error of 1.24°C.

### 1.6 Change in Supply Voltage ($V_{\text{DD}}$)

As seen in Figure 3, the voltage measured on the GPIO1 pin depends on the supply voltage. Changing the supply voltage affects the measured voltage on the GPIO1 pin by typically 1.17 mV/V. This means that if the supply voltage is decreased by 1 V, the voltage measured at the GPIO1-pin is typically 1.17 mV lower.

![Figure 3. Typical GPIO1 Measurements vs Supply Voltage](image-url)
2 Digital Readout

2.1 Operation

The digital readout uses the IFADC in the receive chain to convert the analog temperature sensor voltage to a digital value, which is read from the CHFILT register.

In order to get a digital readout, the following must be done:

• The chip must be in receive mode (RX)
• The DC filter must be disabled and the IF frequency set to zero. Otherwise, the DC information is filtered out.
• The IFAMP must be off; this is done with the chip set in debug mode.

Table 3 lists the registers used to activate digital readout for CC112x and CC120x, respectively.

<table>
<thead>
<tr>
<th>Table 3. Register Settings for Digital Readout</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Register</strong></td>
</tr>
<tr>
<td>DCFILT_CFG</td>
</tr>
<tr>
<td>MDMCFG1</td>
</tr>
<tr>
<td>CHAN_BW</td>
</tr>
<tr>
<td>FREQ_IF_CFG</td>
</tr>
<tr>
<td>ATEST</td>
</tr>
<tr>
<td>ATEST_MODE</td>
</tr>
<tr>
<td>GBIAS1</td>
</tr>
<tr>
<td>PA_IFAMP_TEST</td>
</tr>
</tbody>
</table>

| **Register** | **Value** | **Comment** |
| DCFILT_CFG | 0x40 | Turn off DC filtering |
| MDMCFG1 | 0x47 | Single ADC on, I channel |
| MDMCFG0 | 0x85 | Bypass channel filter |
| CHAN_BW | 0x81 | Lowest decimation factor |
| IF_MIX_CFG | 0x00 | Zero IF |
| ATEST | 0x2A | Temp sensor on |
| ATEST_MODE | 0x07 | Temp sensor on |
| GBIAS1 | 0x07 | Temp sensor on |
| PA_IFAMP_TEST | 0x01 | Route voltage into the ADC |
Figure 4 shows the digital readout is not linear as a function of temperature. A second order equation was chosen to best fit the measured data:

\[
\text{Digital readout} = t_{c1} \times T^2 + t_{c2} \times T + t_{c3}
\]  

(7)

The analog temperature sensor voltage depends on the supply voltage (V_{DD}) and the digital readout will, therefore, have a supply voltage dependency. Table 4 gives typical digital readout values @0°C and 25°C as well as temperature coefficients for different supply voltages.

Table 4. Typical Parameters for Register Readout

<table>
<thead>
<tr>
<th>Technical Information</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>VDD - 2 V</strong></td>
<td></td>
</tr>
<tr>
<td>Typical readout data @ 0°C</td>
<td>100</td>
</tr>
<tr>
<td>Typical readout data @ 25°C</td>
<td>21047</td>
</tr>
<tr>
<td>Temperature coefficient (t_{c1})</td>
<td>-3.72</td>
</tr>
<tr>
<td>Temperature Coefficient (t_{c2})</td>
<td>957.65</td>
</tr>
<tr>
<td>Temperature Coefficient (t_{c3})</td>
<td>385.21</td>
</tr>
<tr>
<td><strong>VDD - 3 V</strong></td>
<td></td>
</tr>
<tr>
<td>Typical readout data @ 0°C</td>
<td>-2059</td>
</tr>
<tr>
<td>Typical readout data @ 25°C</td>
<td>19874</td>
</tr>
<tr>
<td>Temperature coefficient (t_{c1})</td>
<td>-3.32</td>
</tr>
<tr>
<td>Temperature Coefficient (t_{c2})</td>
<td>992.1</td>
</tr>
<tr>
<td>Temperature Coefficient (t_{c3})</td>
<td>-2629.91</td>
</tr>
<tr>
<td><strong>VDD - 3.6 V</strong></td>
<td></td>
</tr>
<tr>
<td>Typical readout data @ 0°C</td>
<td>-3402</td>
</tr>
<tr>
<td>Typical readout data @ 25°C</td>
<td>19133</td>
</tr>
<tr>
<td>Temperature coefficient (t_{c1})</td>
<td>-3.29</td>
</tr>
<tr>
<td>Temperature Coefficient (t_{c2})</td>
<td>1010.87</td>
</tr>
<tr>
<td>Temperature Coefficient (t_{c3})</td>
<td>-3945.8</td>
</tr>
</tbody>
</table>

Figure 4. Register Values vs. Temperature
2.2 Calibration

To eliminate the offset discussed in Section 1, it is recommended to execute a single-point calibration before using the temperature sensor. The temperature, $T$, can then be estimated by:

$$T = t_{c2} + \frac{4^* t_{c1}^* (t_{c3} - D_{M E A S U R E D})}{2^* t_{c1}} - t_{c2} + \frac{4^* t_{c1}^* (t_{c3} - D_{C A L I B R A T I O N})}{2^* t_{c1}} + T_{C A L I B R A T I O N}$$

(8)

• $D_{M E A S U R E D}$ is the readout data at the given temperature
• $D_{C A L I B R A T I O N}$ is the readout data at the calibration temperature
• $T_{C A L I B R A T I O N}$ is the calibration temperature
• $t_{c1}, t_{c2}, t_{c3}$ are the temperature coefficients for the given $V_{DD}$ presented in Table 4

The following accuracy can be achieved using single-point calibration:

• Approximately ±5°C accuracy is possible over a 90°C range centered at $T_{C A L I B R A T I O N}$ when the single-point calibration is used with the coefficients given in Table 4.
• Approximately ±2°C accuracy is given over the range defined by $T_{C A L I B R A T I O N} \pm 10°C$ by the single-point calibration with the coefficients shown in Table 4.

2.3 Code Implementation

The function $tempRead$ shown in Example 2, Example 3 and Example 4 shows how to read the temperature on a CC112x. The same code can be used with CC120x by changing the registers shown in Table 3.

After temperature is read the chip must be reset and all radio configurations have to be redone like shown in Example 1.

Example 1. Radio Must be Reset After Temperature Reading

```c
//Read temperature, chip must be reset after this!
temp = tempRead();

//Reset chip
trxSpiCmdStrobe(CC112X_SRES);

// Write radio registers
registerConfig();
```
In Figure 2, a function for reading temperature and returning it as a Celsius value is shown. The specific registers are set and the chip is put in debug mode to be able to turn off the IFAMP and read the digital value from the CHFILT register.

**Example 2. tempRead Function – Write Relevant Registers and Put Chip in RX**

```c
static int8 tempRead(void) {
    //Variables
    uint8 RegValue = 0;
    uint8 marcStatus;
    uint8 writeByte;
    uint32 ADCValue_I = 0;
    int8 celsius = 0;

    //String to put radio in debug mode
    uint8 txBuffer[18] = {0x0F,0x28,0x02,0x90,0x42,0x1B,0x7E,0x1F,0xFE,0xCD,0x06,0x1B,0x0E,0xA1,0x0E,0xA4,0x00,0x3F};

    //Constants for temperature calculation
    float a = -3.3;
    float b = 992;
    float c = -2629.9;

    //Register settings specific for temp readout.
    writeByte = 0x40;
    cc112xSpiWriteReg( CC112X_DCFILT_CFG, &writeByte, 1); //Tempsens settings, bit 6 high
    writeByte = 0x47;
    cc112xSpiWriteReg( CC112X_MDMCFG1, &writeByte, 1); //Tempsens settings, single ADC, I channel
    writeByte = 0x81;
    cc112xSpiWriteReg( CC112X_CHAN_BW, &writeByte, 1); //Tempsens settings, bit 7 high Bypass ch filt.
    writeByte = 0x00;
    cc112xSpiWriteReg( CC112X_FREQ_IF_CFG, &writeByte, 1); //Tempsens settings, 0-IF
    writeByte = 0x2A;
    cc112xSpiWriteReg( CC112X_FRESH_IF_CFG, &writeByte, 1); //Tempsens settings
    writeByte = 0x07;
    cc112xSpiWriteReg( CC112X_ATEST, &writeByte, 1); //Tempsens settings
    writeByte = 0x07;
    cc112xSpiWriteReg( CC112X_ATEST_MODE, &writeByte, 1); //Tempsens settings
    writeByte = 0x01;
    cc112xSpiWriteReg( CC112X_GBIA, &writeByte, 1); //Tempsens settings
    writeByte = 0x01;
    cc112xSpiWriteReg( CC112X_PA_IFAMP_TEST, &writeByte, 1); //Tempsens settings

    //Set chip in RX
    trxSpiCmdStrobe(CC112X_SRX);

    //Read marcstate and wait until chip is in RX
    do {
        cc112xSpiReadReg(CC112X_MARCSTATE, &marcStatus, 1);
    } while (marcStatus != 0x6D);

    //Read temp from CHFILT register
    cc112xSpiReadReg(CC112X_CHFILT, &RegValue, 1);

    //Calculate temperature
    ADCValue_I = RegValue;
    celsius = (int8)((a *ADCValue_I + b) / c);

    return celsius;
}
```
The radio must be put in debug mode to be able to turn the IFAMP off and read the digital value from the CHFILT register.

**Example 3. tempRead Function – Put the Radio in Debug Mode**

```c
//#### Set radio in debug mode ####
// Write debug init to tx fifo
cc112xSpiWriteTxFifo(txBuffer,sizeof(txBuffer));
// Run code from FIFO
writeByte=0x01;
cc112xSpiWriteReg( CC112X_BIST, &writeByte, 1);   // Strobe IDLE
trxSpiCmdStrobe(CC112X_SIDLE);
// Set IF AMP in PD
writeByte=0x1F;
cc112xSpiWriteReg( CC112X_WOR_EVENT0_LSB, &writeByte, 1);   // Strobe SXOFF to copy command over
trxSpiCmdStrobe(CC112X_SXOFF);
//#### Radio in Debug Mode ####
```

With the radio in debug mode, wait until the data in CHFILT is valid and read the data. Then, the value is converted to Celsius and returned.

**Example 4. tempRead Function – Wait Until Data From CHFILT is Valid, Read Data and Convert to Celsius**

```c
//Wait until channel filter data is valid
do {
cc112xSpiReadReg(CC112X_CHFILT_I2, &RegValue, 1);
} while (!(RegValue&0x08));

//Read ADC value from CHFILT_I registers
cc112xSpiReadReg(CC112X_CHFILT_I2, &RegValue, 1);
ADCValue_I = ((uint32)RegValue) << 16;
cc112xSpiReadReg(CC112X_CHFILT_I1, &RegValue, 1);
ADCValue_I |= ((uint32)RegValue) << 8) & 0x0000FF00;
cc112xSpiReadReg(CC112X_CHFILT_I0, &RegValue, 1);
ADCValue_I |= (uint32)(RegValue) & 0x000000FF;

//Convert ADV value to celsius
celsius = (int) ( (-b+sqrt(pow(b,2)-(4*a*(c-ADCValue_I)) ) ) / (2*a));

//Return degrees celsius
return celsius;
```
3 References

1. *High Performance RF Transceiver for Narrowband Systems Data Sheet (SWRS112)*
2. *High Performance Low Power RF Transceiver Data Sheet (SWRS111)*
3. *Ultra-High Performance RF Narrowband Transceiver Data Sheet (SWRS120)*
# Revision History

## Changes from B Revision (September 2013) to C Revision

<table>
<thead>
<tr>
<th>Changes</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Removed text from the third bullet in Section 2.1.</td>
<td>6</td>
</tr>
<tr>
<td>• Added 'for CC112x and CC120x, respectively' to the sentence prior to Table 3 in Section 2.1.</td>
<td>6</td>
</tr>
<tr>
<td>• Changed information in Table 3.</td>
<td>6</td>
</tr>
<tr>
<td>• Updated information in Table 4.</td>
<td>7</td>
</tr>
<tr>
<td>• Updated information for Equation 8.</td>
<td>8</td>
</tr>
<tr>
<td>• Added information to Section 2.2.</td>
<td>8</td>
</tr>
<tr>
<td>• Added new Section 2.3 to the document.</td>
<td>8</td>
</tr>
<tr>
<td>• Removed Appendix A.</td>
<td>11</td>
</tr>
</tbody>
</table>

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

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license 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 significant portions 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. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have not been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

### Products

- **Audio**
  - www.ti.com/audio
- **Amplifiers**
  - amplifier.ti.com
- **Data Converters**
  - dataconverter.ti.com
- **DLP® Products**
  - www.dlp.com
- **DSP**
  - dsp.ti.com
- **Clocks and Timers**
  - www.ti.com/clocks
- **Interface**
  - interface.ti.com
- **Logic**
  - logic.ti.com
- **Power Mgmt**
  - power.ti.com
- **Microcontrollers**
  - microcontroller.ti.com
- **RFID**
  - www.ti-rfid.com
- **OMAP Applications Processors**
  - www.ti.com/omap
- **Wireless Connectivity**
  - www.ti.com/wirelessconnectivity

### Applications

- **Automotive and Transportation**
  - www.ti.com/automotive
- **Communications and Telecom**
  - www.ti.com/communications
- **Computers and Peripherals**
  - www.ti.com/computers
- **Consumer Electronics**
  - www.ti.com/consumer-apps
- **Energy and Lighting**
  - www.ti.com/energy
- **Industrial**
  - www.ti.com/industrial
- **Medical**
  - www.ti.com/medical
- **Security**
  - www.ti.com/security
- **Space, Avionics and Defense**
  - www.ti.com/space-avionics-defense
- **Video and Imaging**
  - www.ti.com/video

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

Copyright © 2014, Texas Instruments Incorporated