

# MEMS Microphone Direct PDM Input via I2S to a C5515 EVM With Software Decimation

### ABSTRACT

This application report helps system designers demonstrate and understand the technique of directly feeding a MEMS mic output PDM stream into the Inter-IC Sound (I2S) lines of a C5515 EVM. This document is intended for audiences familiar with Digital Signal Processing, decimation and filtering techniques.

#### Contents

1	Overview	2
2	Introduction	2
3	Stages of Decimation	4
4	C5515 EVM Modifications	6
5	Running the Demonstration	7

### List of Figures

1	C5515 Audio Decimation System Overview	2
2	Overview of the Decimation Process	3
3	Examples of Various I2S Clock Frequencies Derived From the DSP Clock	4
4	1 kHz Tone in Baseband is Difficult to Observe, But Its Present at About -30 dB Magnitude	4
5	CIC Output With Significant Noise Spectrum Removed	5
6	FIR1 Output Spectrum	5
7	FIR2 Output Spectrum	5
8	Close-Up of the Board Modifications and Attachment of Jumper Wires	6
9	Connecting the MEMS Microphone to the C5515 EVM	7
10	CSL Projects to Import Into the CCS Workspace	7
11	Importing the Main Decimation Project Into the CCS Workspace	8
12	Building the CCS Project	8
13	Connect Target	9
14	Loading the target_test.out File on the C5515 Core	9
15	Adding a Watch Expression in CCS to Determine the Location of the Audio Buffers	10
16	Audacity Tone Generation Tool	11
17	Executing the Program in CCS	11
18	Halting the Program in CCS	11
19	Memory Browser Window Showing 16-Bit Signed Audio Samples	12
20	Saving the Captured Audio Samples to a .dat File	12
21	Saving Data From the Memory Browser to a .dat File	13
22	Format to Use to Save the Audio Samples in dig-mic-decimation_test_output.dat	13
23	1KHz Sine Wave Output Captured From the PDM Microphone and Decimated on the C5515 DSP	14
24	Applying a High-Pass Filter to the Audio Sample in Audacity to Remove the DC Offset	14
Compose	er Studio is a trademark of Texas Instruments.	

Code Composer Studio is a trademark of Texas Instruments. Audacity is a trademark of Dominic Mazzoni. Matlab is a registered trademark of The MathWorks, Inc. All other trademarks are the property of their respective owners.



### 25 Spectrum Showing a Peak of Approximately 1KHz Signal ..... 15

### 1 Overview

A typical application that uses a MicroElectrical-Mechanical System (MEMS) microphone in its design requires a codec such as the TLV320AIC3204 in order to interface the system processor to the MEMS microphone. The AIC3204 has an I2S interface that provides the down-sampled audio stream utilizing a clock source from the codec. Alternatively, digital microphones with an I2S output can be used, but they tend to consume more power, which requires a larger PCB area and cost.

However, another technique to input a MEMS mic PDM stream into the system processor is to bypass a codec altogether and directly injecting the PDM stream to the processor's I2S lines.

This application report provides a significant advantage where the overall system design could eliminate the cost of a codec, thus, cost savings in the bill of materials (BOM). The power footprint of PDM mics are lower compared to their inter-IC sound (I2S) counterparts. It also demonstrates the steps on how to take a MEMS microphone direct PDM mono input via I2S to a C5515 EVM. The decimation of the captured audio is done in software (see Figure 1).

Note that this document illustrates this technique for evaluation purposes only. It is not intended for a production system without first understanding and evaluating the effects of clock jitter on the audio quality.



The source code discussed in this document can be downloaded from the following URL: https://git.ti.com/apps/c55x-digital-mic-decimation.

Figure 1. C5515 Audio Decimation System Overview

# 2 Introduction

2

Figure 2 shows an overview of the decimation process in the application note's demonstration. Separate DMA channels for "left" and "right" transfer into separate 32-bit buffers. The "Bit picking" block pulls correct bits for the "left" or "right" buffer for input to the Cascaded Integrated Comb (CIC) filter.

16 fractional bits are maintained in the data path to ensure staying below the noise floor of mic.

The input PDM stream from the microphone is 1.024 Mbits/sec divided into two channels. The I2S hardware assembles the input data into 32-bit samples at a rate of 16KHz for each channel. The data is accumulated in a DMA ping-pong buffers. Every 20 ms a PDM stream snapshot contains 640, 32-bit words in each channel (total of 1280, 32-bit words every 200 ms). The "bit picking" block interleaves the data from two channels into 1 stream that is presented to the CIC filter.

**NOTE:** "left" and "right" do not indicate positioning as in stereo microphones. This is a mono setup with 1 microphone. The terms are used to label the DMA buffers only.

The output from the CIC filter is then put through two finite impulse response (FIR) filters to complete the decimation process.



Figure 2. Overview of the Decimation Process

Output sampling rates of the various stages of decimation:

- Output from bit picking = 1 MHz
- Output from CIC = 64 KHz
- Output from FIR1 = 32 KHz
- Output from FIR2 = 16 KHz

Note that the above system supports mono audio single channel is supported since I2S uses single clock edge (rising or falling) to latch data. The typical oversampling ration is 64. For 16 KHz baseband sampling rate:

2Sn\_CLK = 64 × 16KHZ = 1.024 MHz

The I2Sn\_CLK is derived from the system clock, so this poses a limited I2S bit clock divider options. This can have implications for other peripherals in the system that would take the clock from the system clock.

Example:

- Fs = 16 kHz
- I2Sn\_CLK = 1.024 MHz (64xFs)

Derived system clock frequencies:

- 2.048 MHz (/2)
- 4.096 MHz (/4)
- 8.192 MHz (/8)
- 16.384 MHz (/16)
- 32.768 MHz (/32)
- 65.536 MHz (/64)

TEXAS INSTRUMENTS

Introduction



# Figure 3. Examples of Various I2S Clock Frequencies Derived From the DSP Clock

# 2.1 These are the Tasks of the Decimation Block

The output of the "bit-picking" block is the input to the decimation block.

- Decimation: reduce sampling frequency (1.024 Mbps) to baseband Fs (16Kx32 M samples per seconds).
- Shape the signal and remove quantization noise
- Applying anti-aliasing filter; analog anti-aliasing filter rolls off gradually since ADC sampling frequency much higher than baseband Fs/2. Provide additional necessary aliasing rejection.

### 3 Stages of Decimation

4

Figure 4 through Figure 7 depict the sampling spectrum after each stage of the filtering processes.

# 3.1 Digital Mic Output Magnitude Spectrum



# Figure 4. 1 kHz Tone in Baseband is Difficult to Observe, But Its Present at About -30 dB Magnitude



# 3.2 Cascaded Integrated Comb (CIC) Output Magnitude Spectrum



Figure 5. CIC Output With Significant Noise Spectrum Removed

# 3.3 Finite Impulse Response (FIR1) Output Magnitude Spectrum



Figure 6. FIR1 Output Spectrum

15 taps, symmetric coefficients - such as, linear phase - S16Q15 coefficients

# 3.4 Finite Impulse Response (FIR2) Output Magnitude Spectrum





Copyright © 2016, Texas Instruments Incorporated



### C5515 EVM Modifications

www.ti.com

At this point, a 0dB digital gain can be applied along with a high-pass filter for DC offset removal. 58 taps, symmetric coefficients - such as, linear phase. S16Q15 coefficients.

### 4 C5515 EVM Modifications

This application report is based on the use of a C5515EVM Rev B evaluation board. The board schematics are available at http://support.spectrumdigital.com/boards/evm5515/revb/.

The C5515 EVM hardware modifications to be made are:

- Attach wires to the NO-POP pad of R248 that is for I2S0\_CLK
- Remove R240 from the EVM and attach a wire to the pad that connects to I2S0\_RX (see Figure 8)
- R240 I2S0\_RX
- R248 I2S0\_CLK
  - **NOTE:** These pads are shared by the multimedia card/secure data memory card (MMC/SD), so removing R240 affects the SD card functionality.



Figure 8. Close-Up of the Board Modifications and Attachment of Jumper Wires



Figure 9 details connecting the SPM1423HM4H-B MEMS microphone directly to the C5515 EVM.



Figure 9. Connecting the MEMS Microphone to the C5515 EVM

# 5 Running the Demonstration

- 1. Download or clone the source code package from https://git.ti.com/apps/c55x-digital-mic-decimation.
- If not already done, install the C55x Chip Support Library (CSL) from http://softwaredl.ti.com/dsps/dsps\_public\_sw/dsps\_swops\_houston/C55X/latest/index\_FDS.html. This document assumes that the CSL is installed at C:\ti\c55\_lp\c55\_csl\_3.06.
- 3. Launch Code Composer Studio<sup>™</sup> with a new workspace and import the CSL projects *C55XXCSL\_LP* and *atafs\_bios\_drv\_lib*.

Import CCS Eclipse Projec	ts	<u> </u>
Select CCS Projects to Im Select a directory to search	p <b>ort</b> for existing CCS E <mark>c</mark> lipse projects.	
Select search-directory:	C:\ti\c55_lp\c55_csl_3.06	Browse
Select archive file:		Browse
Discovered projects:		-
V atafs_bios_drv_li	ib [C:\ti\c55_lp\c55_csl_3.06\ccs_v6.x_exam +	Select All
CSL_DAT_Examp	ole_Out [C:\ti\c55_lp\c55_csl_3.06\ccs_v6.x	Deselect All
CSL_DMA_IntcE	xample [C:\ti\c55_lp\c55_csl_3.06\ccs_v6.x_ PongExample [C:\ti\c55_lp\c55_csl_3.06\cc	Refresh

Figure 10. CSL Projects to Import Into the CCS Workspace

- **NOTE:** Ensure that the correct macro for the C5515 device is enabled in *C:\ti\c55\_lp\c55\_csl\_3.06\inc\csl\_general.h* prior to compiling the project.
- 4. Import the target\_test project from the software package downloaded in step 1 into the workspace.

Select CCS Projects to Im	port	-
Select a directory to search	for existing CCS Eclipse projects	
Select search-directory:	:55x-digital-mic-decimation	Browse
Select archive file:		Browse
Discovered projects:		
📝 🚞 target_test [C:\	Users\	Select All
		Deselect All
		Defeasels

## Figure 11. Importing the Main Decimation Project Into the CCS Workspace

5. Right click on the target\_test project and build.



### Figure 12. Building the CCS Project



www.ti.cor	n	Running the Demonstration
6.	Create a target configuration for the C5515 EVM and launch the session with core. For more information on setting up CCS (if not familiar), see the http://processors.wiki.ti.com/index.php/CCSv6_Getting_Started_Guide.	a GEL file loaded on the
🎄 Debug		🙀 V 🗖
⊿ 😳 C55	5_evm.ccxml [Code Composer Studio - Device Debugging]	

Figure 13. Connect Target

-

Connect Target

Ctrl+Alt+C

9

🔎 Spectrum Digital DSK-EVM-eZdsp onboard USB Emulator\_0/C55xx (Disconnected : Unknowr

7. Once the core is connected, load target\_test.out onto the core by going to Run → Load → Load Program then navigating to the .out file.

<ul> <li>atafs_bios_drv_lib</li> <li>SSXXCSL_LP</li> <li>target_test</li> </ul>				1 182
Debug target_test.out			Browse	Browse project
			ОК	Cancel
Select a program				
0	OK	Cancel		

Figure 14. Loading the target\_test.out File on the C5515 Core



### Running the Demonstration

8. Prior to running the program, create a watch expression for digGainOutFrame in the watch expression window. digGainOutFrame is the buffer that holds the final output of the decimation stages. The objective of having this expression in the watch window is to ascertain the memory location to grab the audio sample output from the test program. As seen in Figure 15, the location in this case is DATA 0x1246A. This location may vary, so ensure that the address is known before running the program.

	digGainOutFrame	<b>Fil</b>			
			Breakpoint (Code	Composer Studio)	►
			Open Declaration		F3
)	ut circular bui		Cut		Ctrl+X
			Сору		Ctrl+C
al	ReadBufLeft[off		Paste		Ctrl+V
			Use Spaces for Ta	b	
			Declarations		+
			References		+
			Search Text		+
		⇒]	Run to Line		Ctrl+R
		Ð.	Move to Line		
		<b>x+y</b> ⁼?	Add Watch Expres	sion	
			Preferences		
×)= Varia	bles 🙀 Expression	ns 🛛	1919 Registers		
Expressio	on		Туре	Value	Address
a 🏉 d	igGainOutFrame		short[320]	0x01246A@DATA	0x01246A@DAT
Þ 💋	99] [0 99]				
⊳ 💪	] [100 199]				
▶ 🛃	[200 299]				
	= [300 319]				
- \Upsilon A	aa new expression				

Figure 15. Adding a Watch Expression in CCS to Determine the Location of the Audio Buffers

9. In order to demonstrate the capture and decimation of the audio, an audio signal of known frequency needs to be played in front of the microphone. Audacity<sup>™</sup> provides a quick way to generate a tone and is useful in analyzing the audio in later steps. For this demo, a Sine wave of 1KHz is used.

Audacity File Edit View	Transport	Tracks Generate	Effect	Analyze Help				
	)))	(M) (M)	•		7-54-51-48-45	-42-3 Click t	to Start Monit	oring 1 - 18 - P P P
- 1.0	949	1,0	2.0	3.0 4.0	5.0		6.0	7.0
× Audio Track ▼ Mono, 44100Hz 32-bit float Mute Solo T+ LR	1.0 0.5· 0.0- -0.5· -1.0	_		Tone Waveform: Frequency (Hz): Amplitude (0-1): Duration: Manage Preview	Sine 1000 0.25 0.0 h 10 m v	03 s▼ OK	• • •	ncel

Figure 16. Audacity Tone Generation Tool

- 10. Play the audio tone through the speakers in front of the microphone.
- 11. Hit the Resume button or F8 to run the program while the tone is playing.



Figure 17. Executing the Program in CCS

12. Let the program run for a few seconds and then halt it.



Figure 18. Halting the Program in CCS

13. Open a memory browser window and observe the captured audio data at the memory location discussed in step 8.

Running the Demonstration

U menory				-	ain		*   L2 L
DATA	▼ 0x124	6A					
DATA:0x1246	ia <memory< th=""><th>y Rendering</th><th>1&gt; 🛛 🗌</th><th></th><th></th><th></th><th></th></memory<>	y Rendering	1> 🛛 🗌				
16-Bit Signe	d Int	-					
0x01246A	digGainO	OutFrame					
0x01246A	-11395	-8740	-7035	-6550	-7435	-9470	-12435
0x012471	-15725	-19005	-21680	-23370	-23825	-23020	-20955
0x012478	-17970	-14625	-11360	-8685	-7025	-6540	-7360
0x01247F	-9375	-12210	-15555	-18820	-21445	-23180	-23700
0x012486	-22820	-20750	-17855	-14600	-11270	-8580	-6975
0x01248D	-6495	-7340	-9355	-12250	-15550	-18730	-21435
0x012494	-23140	-23600	-22760	-20670	-17805	-14450	-11285
0x01249B	-8630	-6950	-6555	-7415	-9455	-12370	-15710
0x0124A2	-18890	-21625	-23280	-23725	-22825	-20800	-17910
0x0124A9	-14575	-11360	-8620	-6995	-6610	-7435	-9470
0x0124B0	-12390	-15690	-18985	-21690	-23370	-23845	-22985
0x0124B7	-20905	-17980	-14705	-11445	-8835	-7180	-6740
0x0124BE	-7655	-9600	-12540	-15910	-19165	-21890	-23515
0x0124C5	-24020	-23190	-21185	-18300	-14990	-11810	-9175
0x0124CC	-7505	-7060	-7960	-10005	-12910	-16225	-19500
0x0124D3	-22190	-23870	-24305	-23495	-21455	-18550	-15260
0x0124DA	-11975	-9320	-7610	-7170	-7990	-10025	-12910
0x0124E1	-16220	-19530	-22145	-23800	-24230	-23320	-21255
0x0124E8	-18315	-15020	-11790	-9055	-7405	-7000	-7810
0x0124EF	-9850	-12810	-16085	-19325	-22015	-23645	-24060
0x0124F6	-23235	-21170	-18250	-14930	-11655	-8965	-7215
0x0124FD	-6800	-7685	-9730	-12665	-15990	-19210	-21940
0x012504	-23615	-24060	-23215	-21120	-18250	-14950	-11680
0x01250B	-9045	-7330	-6860	-7710	-9725	-12695	-16035

# Figure 19. Memory Browser Window Showing 16-Bit Signed Audio Samples

14. Save the audio samples into a .dat file by right-clicking on the memory browser window as in Figure 20. Save the data at \c55x-digital-mic-decimation\test\_data\output\dig-mic-decimation\_test\_output.dat, file type set as TI Data.

📋 Memory I	Browser 🛛	🧟 🝷 🌆 👻 🦚 1	- 🖗
DATA	▼ 0x1246	A	
DATA:0x1246	a <memory< td=""><td>Rendering 1&gt; 🛛</td><td></td></memory<>	Rendering 1> 🛛	
16-Bit Signe	d Int	•	
0x01246A	digGainOu	tFrame	
0x01246A	-11395		9470
0x012471	-15725	Add Watchpoint (C/C++)	2302
0x012478	-17970	Find and Replace Ctrl+F	5540
0x01247F	-9375		2318
0x012486	-22820	Configure	3580
0x01248D	-6495		L873
0x012494	-23140	Copy To Clipboard	1445
0x01249B	-8630	с. т.н.	L237
0x0124A2	-18890	Copy To Memory	2080
0x0124A9	-14575	Load Memory	7435
0x0124B0	-12390	Caus Manager	2384
0x0124B7	-20905 🦉	Save Memory	7180
0x0124BE	-7655	Fill Memory	2189
0x0124C5	-24020	Brint	1181
0x0124CC	-7505	Plint	1622
0x0124D3	-22190	Reference Buffer	1855
0x0124DA	-11975	Tiler Display	1002
0x0124E1	-16220	The Display	2332
0x0124E8	-18315 🌧	) Refresh	7000
0x0124EF	-9850	/ Nellesii	2364
0x0124F6	-23235	-21170 -18250 -14930 -11655 -	-8965

Figure 20. Saving the Captured Audio Samples to a .dat File



ave men Select a fil	ory e to save the memory data
ile:	decimation\test_data\output\dig-mic-decimation_test_output.dat
File Type:	TI Data 👻

## Figure 21. Saving Data From the Memory Browser to a .dat File

15. Save the data in the format as shown in Figure 22 and click Finish.

ormat: 16-Bit	Signed Int	•			
Farget	0.12464				_
start Address:	UXI246A				_
viemory Page:	DATA				
ength:					
Specify the	e number of m	emory words to	o read:		
320					
O Specify the	e data block di	mension in nur	nber of memo	ory words:	
Number o	f Rows:	Numbe	of Columns:		
		12		100 million (100 m	

### Figure 22. Format to Use to Save the Audio Samples in dig-mic-decimation\_test\_output.dat

- 16. At this point, the data is ready to be processed by the Matlab® script for .wav file conversion.
- 17. Open \c55x-digital-mic-decimation\matlab\_scripts\dig\_mic\_decimation\_ConvertCCSbuf2wav.m in Matlab.
- 18. Hit the Run button and a .wav file will be created: \c55x-digital-micdecimation\matlab\_scripts\wavoutput\_dig-mic-decimation\_test.wav.
- 19. Open *wavoutput\_dig-mic-decimation\_test.wav* in Audacity or any other audio software for analysis. As observed in Figure 23, an output with a 1KHz Sine wave would be observed.



**NOTE:** The sine wave has some DC offset. A high-pass filter (HPF) can be implemented in the decimation code to remove this offset. Alternatively, the audio can be scrubbed by using a HPF in Audacity as discussed in step 20.

🔒 wavoutpu	ıt_dig-mic-decin	ation_test							Agent and Annual		- Name and Address	-	-		_	_
File Edit	View Transpor	t Tracks Genera	te Effect Anal	yze Help												
	(F)			201	R -57 -54	-51 -48 -45 -42 -3 (	Click to Start Mo	nitoring 1 -18 -1	-12 -9 -6 -3 0	€ 10 L -57 -1	54 -51 -48 -45 -	12 - 39 - 36 - 33 - 30	-27 -24 -21 -18 -	15-12-9-6-3		
	~	00	1	> ↔ * 3	む 🖦 🖷	夏季で	○	P P P	<u>k</u>   • - •	) <u>.</u> [[M	IME 🔻	Microsoft So	und Map 👻 2 (Si	tereo) Rei 👻 🎝	Speakers / Head	lphon 🔻
- 0.0010	0.000	0.0010	0.0020	0.0030	0.0040	0.0050	0.0060	0.0070	0.0080	0.0090	0.0100	0.0110	0.0120	0.0130	0.0140	0.0150
X wavoutpu Mono, 16000 32-bit float Mute S	t <b>v</b> 1.0 Hz 0.5-															
	+ 0.0- R -0.5-	$\sim$		$ \land $	$ \land $	$\sim$	5		$\sim$		5			$\sim$	$\sim$	<u></u>

Figure 23. 1KHz Sine Wave Output Captured From the PDM Microphone and Decimated on the C5515 DSP

20. An HPF can be used in Audacity by going to Effect → High Pass Filter and choosing the suitable HPF parameters as seen in Figure 24.



Figure 24. Applying a High-Pass Filter to the Audio Sample in Audacity to Remove the DC Offset



21. The audio spectrum can be plotted in Audacity to better analyze the signal Analyze → Plot Spectrum. As seen in Figure 25, the audio spectrum indicates a peak of a signal of approximately 1KHz.



# Figure 25. Spectrum Showing a Peak of Approximately 1KHz Signal

22. Further filtering techniques can be implemented as desired to refine the audio quality.

### **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		Applications					
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive				
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications				
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers				
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps				
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy				
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial				
Interface	interface.ti.com	Medical	www.ti.com/medical				
Logic	logic.ti.com	Security	www.ti.com/security				
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense				
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video				
RFID	www.ti-rfid.com						
OMAP Applications Processors	www.ti.com/omap	TI E2E Community	e2e.ti.com				
Wireless Connectivity	www.ti.com/wirelessconnectivity						

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265 Copyright © 2016, Texas Instruments Incorporated