AND9109/D

BelaSigna® R262 Getting Started Guide

ON Semiconductor®

http://onsemi.com

APPLICATION NOTE

Introduction

This application note guides you through the process of evaluating the performance of BelaSigna R262 and discusses the typical steps required to integrate BelaSigna R262 into a new prototype or design.

The intended audience is system architects and engineers designing products that capture speech using standard electret or MEMS microphones, which can benefit from advanced noise reduction and improved voice clarity.

Overview

This application note will first discuss the BelaSigna R262 advanced noise reduction algorithm in terms of functionality and performance. Next, the main points to consider as part of the design−in process are discussed, and four reference designs are presented. A schematic and a list of design notes are provided for each reference design.

The Design Validation section discusses using the BelaSigna R262 Tweaker software to adjust the noise reduction settings and other key parameters, generate customized EEPROM images, and calibrate signal levels.

Advanced Speech and Noise Management System

The BelaSigna R262 advanced noise reduction algorithm is an adaptive solution that identifies and enhances human speech. Many two−microphone algorithms depend on beamforming and rely on the speech originating from a specific direction. As a result, the performance of these algorithms can vary greatly depending on the location of the speech and the noise relative to the microphones, and typically will distort the speech as well as the noise. Other algorithms depend on a speech−facing microphone and a noise−facing microphone and can require a lot of algorithm tuning for reasonable performance.

BelaSigna R262 is different as it is a true 360° self−adaptive multi−beamformer that distorts neither speech nor noise. It applies attenuation or gain based on intelligent classification of signals, resulting in a signal where the noise is attenuated, and the speech is enhanced. The attenuated noise is left as natural as possible so other downstream processing is still effective.

The algorithm works with two forward−facing microphones spaced 10 to 30 mm apart (typically), and no tuning or matching of microphones is required. The speech source can come from any direction (normally in front of the microphones), as long as it is within the “working sphere” of the current algorithm mode. The algorithm mode is set based on the required talking distance (distance between the mouth reference point and the microphones), with maximum noise reduction performance decreasing as talk distance increases. BelaSigna R262 can be set to perform maximum noise reduction with a talk distance of 5 to 10 cm, mild noise reduction with improved voice pickup at up to 5 m away or anywhere in between.

Since the solution is Signal−to−Noise Ratio (SNR) independent, it operates optimally in both low SNR as in high SNR environments. At the same time it performs de−reverberation of the received signals. The solution is ideal for electronic communication devices where microphones capture speech and might be used in noisy environments. The solution allows for flexible microphone positioning and arbitrary placement of the device (e.g. handheld position) in the actual environment.

An overview of the fundamental components of the algorithm is presented in Figure 1.

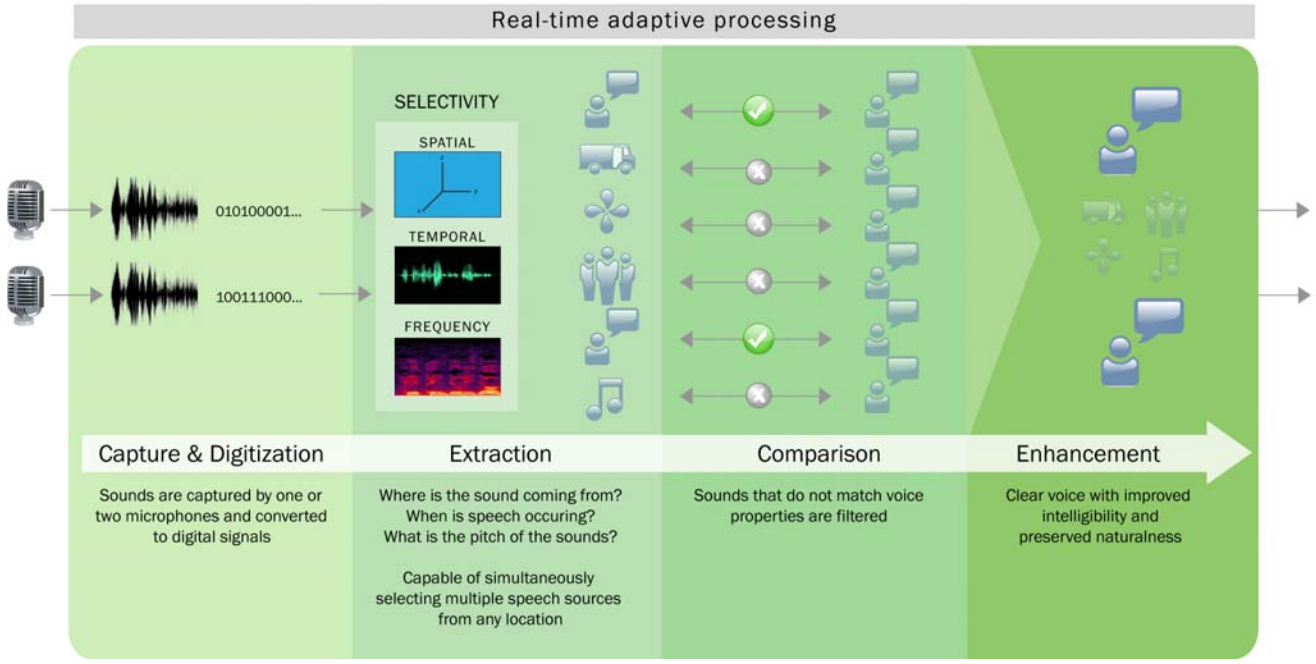

Figure 1. BelaSigna R262 Advanced Speech and Noise Management

Algorithm Modes and Performance

BelaSigna R262 offers three core algorithm modes: close−talk, near−talk, and far−talk.

Close−talk mode aggressively filters noise and manages gain to pick up speech within 5 to 10 cm from the microphone array, effectively attenuating both speech and noise that is farther away. This mode is suitable for mobile handsets and radios where the speech source is very close to the microphones and the noise level can be quite high; up to 90 dB SPL. Noise reduction performance in this mode ranges from 20 to 30 dB SNR Improvement (SNR−I), depending on the type of noise.

Near−talk mode features excellent noise reduction in devices where the speech source is located slightly further away from the microphones; typically 50 to 100 cm. This mode is suitable for handsfree devices and performs well in noise levels up to 70 dB SPL. Expected noise reduction performance in this mode is 10 to 20 dB SNR−I, depending on the type of noise.

Far−talk mode amplifies and enhances speech from sources up to 5 m from the microphone array while removing mild noise; up to 50 dB SPL. This mode is suitable for recording lectures or processing conference calls on speakerphone and will provide 7 to 15 dB SNR−I, depending on the type of noise.

In addition to these three core modes (Close, Near and Far), BelaSigna R262 features a mixing capability, which allows the selection of any two of these three main modes combined through a mixing ratio to create an intermediate mode, providing unlimited flexibility. This allows you to fine tune the algorithm based on the microphone−to−user distance on the target device. And with its dual output, BelaSigna R262 can be configured to provide any two core modes, or any core mode plus an intermediate mode simultaneously on its two outputs.

A selection of performance metrics are shown in Table 1 for some input signal−to−noise ratios and noise types. Signal−to−Noise Ratio Improvement (SNR−I) was measured according to the ITU−T G.160 standard, with BelaSigna R262 operating on reference hardware, in conjunction with common omni−directional microphones (ECMs) positioned 10 mm away from each other. To verify that subjective quality was maintained through the noise management process, a Perceptual Enhancement of Speech Quality (PESQ) measurement was also taken for each condition. PESQ improvement correlates to MOS improvement, the widely accepted subjective standard in voice quality.

Table 1. ALGORITHM PERFORMANCE

Parameter	Test Conditions	Value	Unit
CLOSE−TALK MODE (5 cm to 10 cm talking distance)			
SNR Improvement (SNR−I)	White noise, 6 dB SNR	30.0	dB
	Babble noise, 9 dB SNR	20.6	dB
	Pink noise, 12 dB SNR	19.9	dB
PESQ Improvement	White noise, 6 dB SNR	0.8	–
	Babble noise, 9 dB SNR	0.3	–
	Pink noise, 12 dB SNR	0.4	–
NEAR−TALK MODE (50 cm to 100 cm talking distance)			
SNR Improvement (SNR−I)	White noise, 6 dB SNR	19.6	dB
	Babble noise, 9 dB SNR	10.6	dB
	Pink noise, 12 dB SNR	14.5	dB
PESQ Improvement	White noise, 6 dB SNR	0.5	–
	Babble noise, 9 dB SNR	0.4	–
	Pink noise, 12 dB SNR	0.4	–
FAR−TALK MODE (2.5 m to 5 m talking distance)			
SNR Improvement (SNR−I)	White noise, 6 dB SNR	14.7	dB
	Babble noise, 9 dB SNR	6.9	dB
	Pink noise, 12 dB SNR	11.1	dB
PESQ Improvement	White noise, 6 dB SNR	0.7	–
	Babble noise, 9 dB SNR	0.4	–
	Pink noise, 12 dB SNR	0.5	–

Pick−Up Distance

BelaSigna R262 is a fully adaptive system that will automatically adjust its signal processing algorithm depending on the input conditions. As the noise level increases, the user will naturally speak louder or move closer to the microphones due to the Lombard Effect. As a result, the optimal talking distance from the microphones will vary, depending on the noise conditions.

Figure 2, Figure 3, and Figure 4 show the optimal talking distance for the three core algorithm modes as a function of the input signal−to−noise ratio. When designing BelaSigna R262 into a device, it is important to assess the targeted use case of the product and to consequently select an appropriate algorithm mode. It is also important to realize that the three modes discussed here are just examples of what can be obtained with BelaSigna R262. Adjustment of the talking distance and the noise reduction aggressiveness are just two of the performance parameters that can be easily controlled.

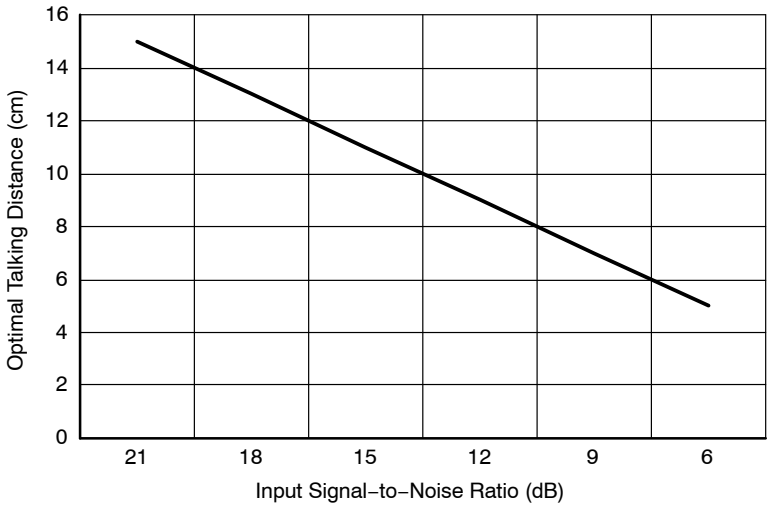

Figure 2. Talk Distance vs. Input SNR, Close−Talk Mode

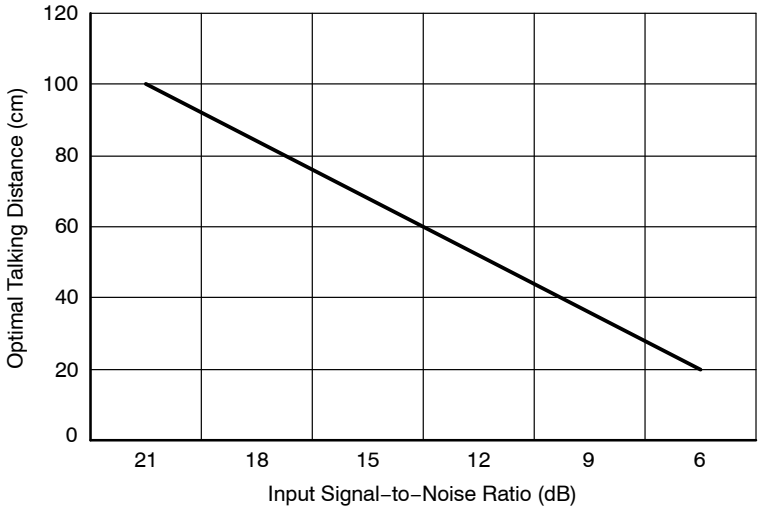

Figure 3. Talk Distance vs. Input SNR, Near−Talk Mode

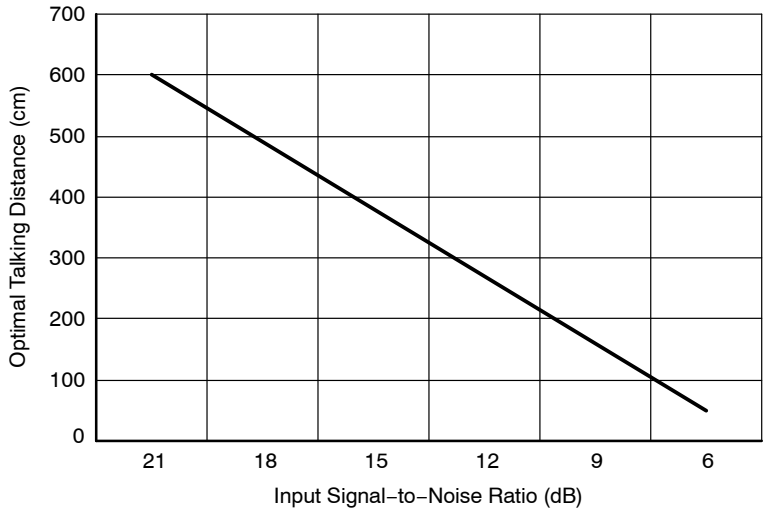

Figure 4. Talk Distance vs. Input SNR, Far−Talk Mode

The Design−In Process

This section will guide you through the design−in process, including how to choose a reference design, and what to consider when designing your own BelaSigna R262−based product.

Reference Designs

ON Semiconductor provides four slightly different reference designs to integrate BelaSigna R262 into a product or prototype. Which reference design you choose depends on your particular application. There are some fundamental questions that can be answered to make the selection of a reference design simple. The characteristics of the four reference designs are summarized in Table 3.

Schematics for these four reference designs are provided in subsequent sections. For recommendations on PCB layout, refer to the BelaSigna R262 datasheet, AND9111/D − BelaSigna R262 Analog Demonstrator User's Guide, AND9112/D − BelaSigna R262 Prototyping Module User's Guide and the BelaSigna R262 Routing Options presentation available from ON Semiconductor.

Note that these reference designs are provided for your convenience. There are many other suitable designs to integrate BelaSigna R262 into your product. If you require assistance contact your local ON Semiconductor support representative.

Selecting a Package

BelaSigna R262 comes in two different packages: a 26−ball and a 30−ball WLCSP. Both packages are 2.223 x 2.388 x 0.724 millimeters (5.3 mm^2) but the 26−ball package allows for PCB layouts with wider trace and space requirements, providing an opportunity to further reduce design−in cost while trading off features. Depending on your application, this trade−off might make sense.

The 26−ball package allows for 5−mil trace/space PCBs (3−mil trace/space is required to route all 26 balls, but this is rarely required for a given application) but is missing the following pins (compared to the 30−ball package):

- SWAP_CHAN
- NRESET
- A_OUT0
- AI1/LOUT1

The 30−ball device has no pin restrictions, but requires 3 or 2−mil trace/space PCB technology.

The package you select largely depends on your output requirements. If you need to connect the analog output of BelaSigna R262 to a fully differential input or if you require two different, simultaneously processed outputs (i.e. both close−talk and near−talk being output on separate channels at the same time) you must select the 30−ball version of BelaSigna R262, and thus reference design 4. If you only require one single−ended analog output, or intend on using the digital microphone (DMIC) output only, you can safely select the 26−ball version and benefit from simpler PCB routing and a reduced solution cost. Your choice of reference design 1, 2, or 3 (all of which use the 26−ball device) then depends on other factors, which are discussed in subsequent sections.

For additional information on the functionality of the pins missing on the 26−ball device refer to the BelaSigna R262 datasheet.

Audio Outputs

BelaSigna R262 has the ability to output two processed channels simultaneously, each with their own mode of operation. For example, it is possible to output a close−talk on one channel while simultaneously outputting a near− or far−talk on the other channel. This is especially useful in applications that require two separate modes, but may not have any means of communicating with BelaSigna R262. This dual−output feature allows your application to select between two completely different use cases, simply by using the other output channel (analog or DMIC); no mode switching command or algorithm re−configuration is required.

If your application requires this dual output functionality, you must use reference design 2, 3, or 4. Reference designs 2 and 3 will give you this functionality on the DMIC interface only, while reference design 4 provides dual single−ended analog outputs, as well as the ability to use the stereo DMIC interface (with a minor configuration change to select external clocking).

Low−Power Standby

If your application has a low−power requirement and will put BelaSigna R262 into deep sleep mode and requires the microphones to also be powered off, you must choose reference design 4.

Reference designs 1, 2, and 3 all leave the microphones powered, even when BelaSigna R262 is sleeping.

Standard Hardware Configurations

BelaSigna R262 was designed to be configured using resistive dividers on four different input pins. There are four inputs which facilitate configuration of the system using voltage dividers. Reference designs 1, 2, and 4 are completely configured using voltage dividers connected to the configuration pins. For reference, the recommended resistor values for the typical voltage divider between 0 V and VREG (1.0 V) shown in Figure 5 are summarized in Table 2.

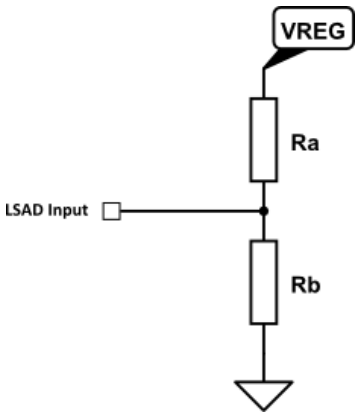

Figure 5. LSAD Voltage Divider

Table 2. VOLTAGE DIVIDER RANGES AND PRESETS

Ra	Rb	Voltage Range	Preset
10 kΩ	DNI	0.65 – 1.00 V	0–2
75 kΩ	100 kΩ	0.50 – 0.63 V	3
100 kΩ	75 kΩ	0.36 – 0.49 V	4
100 kΩ	39 kΩ	0.22 – 0.35 V	5
100 kΩ	16 kΩ	0.08 – 0.21 V	6
DNI	10 kΩ	0.00 – 0.07 V	7

Configuring BelaSigna R262 for your application using voltage dividers greatly simplifies the integration process. Refer to reference designs 1, 2, and 4 as well as the BelaSigna R262 datasheet and the BelaSigna R262 Communications & Configuration Guide for details on what standard configurations are possible through hardware configuration only.

Custom Configurations

It is possible to configure all parameters of the device through software, provided BelaSigna R262 is connected to an I^2C master mode device. Parameters such as input routing, output attenuation, mode switching, and even custom algorithm tunings can all be configured over I^2C. This additional configuration over I^2C can be combined with any of the four reference designs.

Similarly, it is possible to load a custom application from an external EEPROM (I^2C or SPI) that configures all aspects of the system. A custom application can augment or replace the core functionality provided by the ROM application, providing customized access to GPIO signals or the ability to enable and use the PCM port, for example.

Some examples of configuration options that will require either an external EEPROM or configuration from an I^2C master are:

- Custom input stage routing and configuration
- Custom preamplifier gains
- A VMIC setting of anything other than VDDA (2.0 V)
- Custom noise reduction “core” tunings other than the standard close/near/far−talk
- A noise reduction mixing ratio that is not one of the following values:
  - 0%
  - 20%
  - 40%
  - 60%
  - 80%
  - 100%
- Custom equalization (default is no equalization)
- A non−standard clocking configuration (refer to the BelaSigna R262 datasheet for standard values)
- Custom analog or digital output attenuation. The default is 0 dB analog attenuation and −6 dB digital attenuation.

- Any non−standard DMIC output channel routing (refer to the BelaSigna R262 datasheet for standard values)

For additional details on how to customize the configuration of BelaSigna R262, refer to the BelaSigna R262 Communications & Configuration Guide. For details on custom applications, contact your local ON Semiconductor support representative.

Reference Design Checklist

The following details are common to all reference designs and should be validated on your final schematic:

1. The DC removal capacitors between the microphones and BelaSigna R262 should be 10 nF.
2. The pins VSSD (A5) and VSSRCVR (E9) as well as the negative terminal of the filter capacitors on VBAT and VDDD should all be connected to digital ground.
3. The pin VSSA (F4), the negative terminal of the filtering capacitors on VREG and VDDA, the resistive dividers, the negative terminal of the microphones as well as all unused analog inputs should be connected to analog ground.
4. DEBUG_RX (A1) should be connected to VDDO and DEBUG_TX (B2) should be left floating.
5. The charge pump capacitor between CAP0 (G7) and CAP1 (F8) should be 100 nF.
6. The filter capacitors on all power supplies (VBAT, VDDD, VREG, VDDO, and VDDA) should be 1 μF.
7. Since the output driver is not used, VBATRCVR is connected to VDDA and does not require a separate capacitor (VDDA has its own filter capacitor)

Reference Designs

This section provides detailed information on the four BelaSigna R262 reference designs provided by ON Semiconductor. The main characteristics of each reference design are summarized in Table 3.

Table 3. REFERENCE DESIGN CHARACTERISTICS

Parameter	Reference Design			
Reference Design Number	1	2	3	4
Reference Design Name	Single−ended analog output with internal clocking	Standard DMIC output	Single−ended analog or DMIC output with custom configuration	Full feature set
Package	26−ball	26−ball	26−ball	30−ball
OPN	BR262W26A103E1G	BR262W26A103E1G	BR262W26A103E1G	BR262W30A103E1G
Analog Outputs	One single−ended	Unused	One single−ended	Dual single−ended or mono differential
DMIC Interface	Not available	Available	Available (choose DMIC or analog)	Not available
Configuration	LSADs	LSADs	I^2C or EEPROM	LSADs
Clocking	Internal	External (from DMIC host)	Defined by custom application	Internal
Microphone Bias	VDDA (always on)	VDDA (always on)	VDDA (always on)	VMIC (software controlled)
Reset Pin	Not available	Not available	Not available	Available
PCB Trace/Space	5 mils	5 mils	5 mils	3 or 2 mils
Comments	Additional configuration can be obtained by using an external I^2C host	Additional configuration can be obtained by using an external I^2C host	Custom application downloaded either by I^2C host, or loaded from an externally connected EEPROM (I^2C or SPI)	Additional configuration can be obtained by using an external I^2C host

Reference Design 1 – Single−Ended Analog Output with Internal Clocking

Reference design 1 is the simplest way to deploy BelaSigna R262 in an application that does not use the DMIC interface, and only requires one single−ended analog output. It uses the internal oscillator, avoiding the need for an external clock, and is configured using voltage dividers on the four LSAD pins. The microphones are biased by VDDA, a 2.0 V regulated power supply generated by BelaSigna R262. This reference design uses the 26−ball package, allowing 5−mil PCB routing. A schematic is shown in Figure 6.

The values for the resistive dividers for this reference design are described in Table 4.

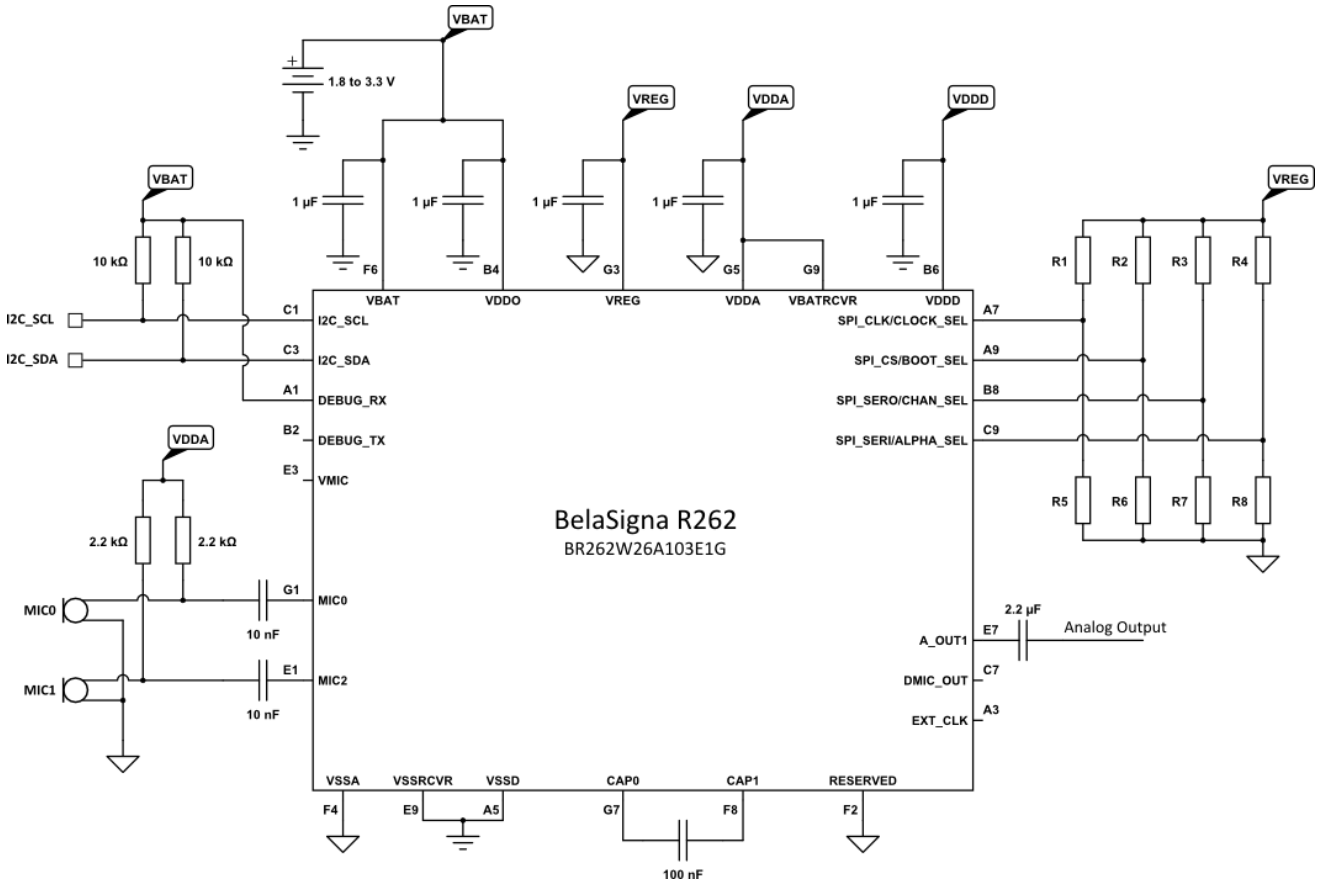

Figure 6. Schematic for Reference Design 1

Table 4. REFERENCE DESIGN 1 CLOCK_SEL AND CHAN_SEL RESISTIVE DIVIDER VALUES

CLOCK_SEL		CHAN_SEL	
R1	**R5**	**R3**	**R7**
Do not install (DNI)	10 kΩ	Do not install (DNI)	10 kΩ
This reference design uses the internal oscillator and is always configured with CLOCK_SEL set to preset 7.		This reference design uses a single−ended analog output, but allows the selection of any intermediate noise reduction tuning by choosing preset 7 to provide the mixed output (the result of BOOT_SEL and ALPHA_SEL) on Channel 1 (A_OUT1).	

The noise reduction tuning that will be output on A_OUT1 is determined by the combination of the BOOT_SEL and ALPHA_SEL voltage dividers. The BOOT_SEL setting sets the start and end points or “working range” of the noise reduction core algorithm and the ALPHA_SEL setting determines the location of the final tuning within the working range (the “mixed” output). The possible combinations of BOOT_SEL and ALPHA_SEL are summarized in Appendix A in Table 9 – *BOOT_SEL and ALPHA_SEL Resistive Divider Values for Intermediate Noise Reduction Tunings.*

Reference design 1 allows you to select between eleven different configurations (or “tunings”) of the noise reduction algorithm based on the distance from the microphones to the speech source(s). There are three core modes: close−talk, near−talk and far−talk, and there are four intermediate tunings between each of these core modes based on the ALPHA_SEL settings of 20%, 40%, 60%, and 80%. When ALPHA_SEL is set to 0% or 100% the resulting tuning is equivalent to the start or end point respectively, and as a result is identical to one of the three core modes.

Refer to Table 9 in Appendix A for the resistive divider settings for BOOT_SEL and ALPHA_SEL that result in the eleven different configurations, as well as details on the characteristics of the intermediate tunings.

Design Notes for Reference Design 1

The following notes apply to reference design 1:

1. This reference design uses VDDA (2.0 V) for the microphone bias. You can safely change this to use VREG (1.0 V), provided your microphones can operate with a 1.0 V bias. Alternatively, you can also use the VMIC pin, which is software controlled (and defaults to VDDA). Using VMIC

has the added advantage that the microphone bias will be shut down when BelaSigna R262 goes to sleep, saving power. However, using the VMIC pin requires 3−mil trace/space on your PCB in order to successfully route the I^2C data line (I2C_SDA) in addition to the VMIC bias pin.

2. The analog output (A_OUT1) has an impedance of approximately 3 kΩ. You cannot use this pin to directly drive a speaker or headphone. If you wish to drive a speaker or headphone an audio amplifier is required.
3. The power supply of BelaSigna R262 must be between 1.8 V and 3.3 V.
4. Internal clocking is recommended if you only require an analog output (the DMIC output requires an external clock). To configure BelaSigna R262 for internal clocking, the CLOCK_SEL pin can either be grounded, or configured to use preset 7, as described in Table 4.
5. It is recommended that, at a minimum, the I^2C port be routed to a set of test points with 10 kΩ pull−up resistors to VBAT

Reference Design 2 – Standard DMIC Output

Reference design 2 is intended to be used when BelaSigna R262 must communicate with an external codec using the DMIC interface. In this case BelaSigna R262 is clocked externally by the DMIC clock (provided by the codec). Configuration is performed using voltage dividers on the four LSAD pins. The microphones are biased by VDDA, a 2.0 V regulated power supply generated by BelaSigna R262. This reference design uses the 26−ball package, allowing 5−mil PCB routing. A schematic is shown in Figure 7.

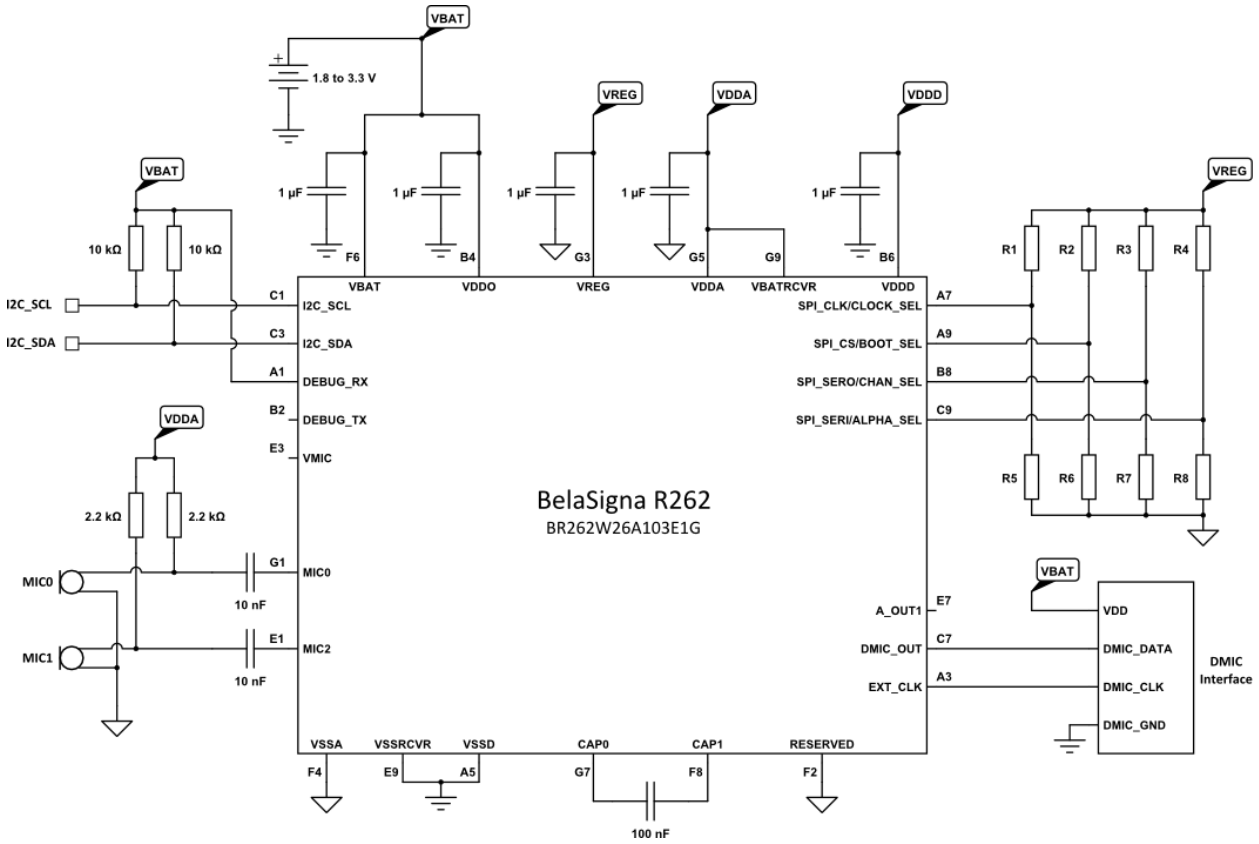

Figure 7. Schematic for Reference Design 2

The DMIC clock frequency provided by the external codec dictates the voltage divider values on the CLOCK_SEL pin. Choose the values of R1 and R5 based on the entry in Table 5 that matches the DMIC clock frequency provided by the codec.

Since this reference design interfaces to a codec with a stereo DMIC interface, CHAN_SEL is configured as described in Table 6 to provide two processed outputs; one on each channel of the DMIC interface. The noise reduction tunings that will be output on the two DMIC channels are determined by the combination of the BOOT_SEL and ALPHA_SEL voltage dividers. The “start” point of the noise reduction tuning range will be output on Channel 0 (the DMIC left channel) and is governed by the LSAD value read on the BOOT_SEL pin. The mixed/intermediate noise reduction tuning will be output on Channel 1 (the DMIC right channel), and depends on a combination of the LSAD values read on both the BOOT_SEL and ALPHA_SEL pins. The

possible combinations of BOOT_SEL and ALPHA_SEL are summarized in Appendix A in Table 9 – *BOOT_SEL and ALPHA_SEL Resistive Divider Values for Intermediate Noise Reduction Tunings*.

Using the values provided in Table 9 you have a choice of two different tunings for Channel 0 (close−talk or near−talk) since Channel 0 will always contain the “start” of the noise reduction tuning range. For a given start point on Channel 0 there are then six possible tuning selections for Channel 1. Refer to Table 9 in Appendix A for all of the possible combinations.

Table 5. REFERENCE DESIGN 2 CLOCK_SEL RESISTIVE DIVIDER VALUES

CLOCK_SEL		(Preset Number) Clock Frequency
R1	R5	
10 kΩ	Do not install (DNI)	(0−2) 2.048 MHz
75 kΩ	100 kΩ	(3) 2.4 MHz
100 kΩ	75 kΩ	(4) 2.8 MHz
100 kΩ	39 kΩ	(5) 3.072 MHz

Table 6. REFERENCE DESIGN 2 CHAN_SEL RESISTIVE DIVIDER VALUES

CHAN_SEL		Preset	Noise Reduction Outputs	Channel 0	Channel 1
R3	R7				
75 kΩ	100 kΩ	3	Dual	Start of range (as per BOOT_SEL)	Mixed Output (as per BOOT_SEL and ALPHA_SEL)

Design Notes for Reference Design 2

The following notes apply to reference design 2:

1. This reference design uses VDDA (2.0 V) for the microphone bias. You can safely change this to use VREG (1.0 V), provided your microphones can operate with a 1.0 V bias. Alternatively, you can also use the VMIC pin, which is software controlled (and defaults to VDDA). Using VMIC has the added advantage that the microphone bias will be shut down when BelaSigna R262 goes to sleep, saving power. However, using the VMIC pin requires 3−mil trace/space on your PCB in order to successfully route the I^2C data line (I2C_SDA) in addition to the VMIC bias pin.
2. The DMIC output sends a stereo pulse−density−modulated (PDM) signal with output Channel 0 of BelaSigna R262 on the left DMIC channel, and output Channel 1 of BelaSigna R262 on the right DMIC channel.
3. The power supply of BelaSigna R262 must be between 1.8 V and 3.3 V.
4. It is recommended that, at a minimum, the I^2C port be routed to a set of test points with 10 kΩ pull−up resistors to VBAT.

Reference Design 3 – Single−Ended Analog or DMIC Output with Custom Configuration

Reference design 3 is intended to be used in situations when configuration of the chip beyond what is possible in hardware is required, or perhaps it is just desired to have full control over all aspects of the device via an external I^2C master. If you choose this reference design, you must either connect an I^2C master device to load a custom application to configure BelaSigna R262, or connect an EEPROM containing a custom application to be loaded automatically at boot time by BelaSigna R262. Both I^2C and SPI EEPROMs are supported so either one may be used, but a 64 kilobit I^2C EEPROM is recommended for this particular reference design (CAT24C64).

The microphones are biased by VDDA, a 2.0 V regulated power supply generated by BelaSigna R262. This reference design uses the 26−ball package, allowing 5−mil PCB routing but as a consequence only offers one single−ended analog output (A_OUT1). It is also possible to use the DMIC interface with this reference design, but if you want to maintain 5−mil PCB routing you must choose either the analog output or the DMIC output as routing both will require a 3−mil trace/space PCB. More information is provided in the design notes for this reference design. A schematic is shown in Figure 8.

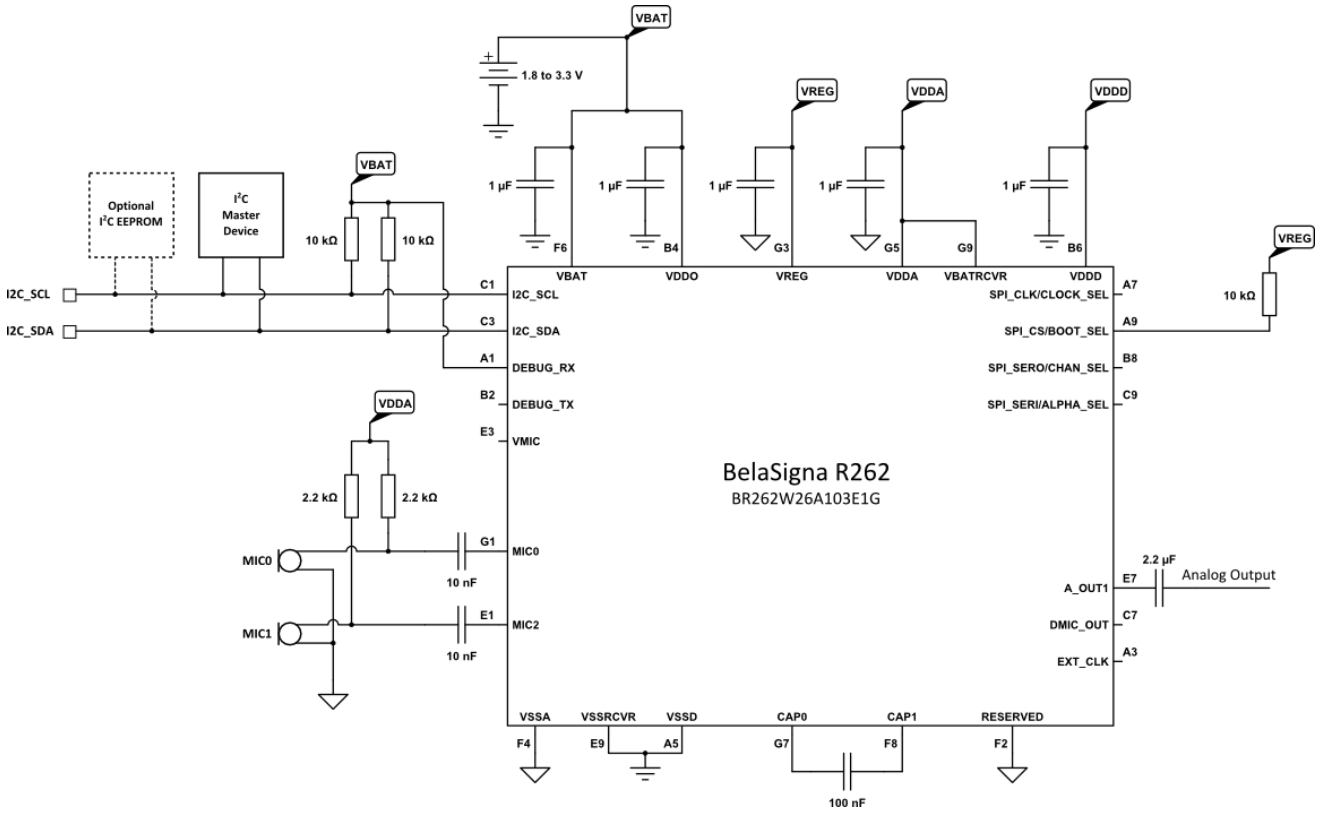

Figure 8. Schematic for Reference Design 3

The four LSAD resistive dividers do not need to be configured in this reference design, except for the BOOT_SEL pin which needs to be set for external boot mode by connecting a 10 kΩ pull−up resistor to VREG. In this mode BelaSigna R262 will boot on its internal clock and look for an externally connected EEPROM on both the I^{2}C and SPI ports. If no EEPROM is found, it will wait for an external I^{2}C master to connect and load a custom application onto the device. It is the custom application being loaded that will determine whether BelaSigna R262 will continue to run on its internal clock, or whether it will switch to an external clock at an expected clock frequency.

For additional details on the configuration of the algorithm, please refer to BelaSigna R262 Communications & Configuration Guide. For details on custom applications, contact your local ON Semiconductor support representative.

Design Notes for Reference Design 3

The following notes apply to reference design 3:

1. This reference design uses VDDA (2.0 V) for the microphone bias. You can safely change this to use VREG (1.0 V), provided your microphones can operate with a 1.0 V bias. Alternatively, you can also use the VMIC pin, which is software controlled (and defaults to VDDA). Using VMIC has the added advantage that the microphone bias will be shut down when BelaSigna R262 goes to sleep, saving power. However, using the VMIC pin requires 3−mil trace/space on your PCB in order to successfully route the I^{2}C data line (I2C_SDA) in addition to the VMIC bias pin.
2. This reference design only offers one single−ended analog output (A_OUT1). It is also possible to use the DMIC interface with this reference design, but if you want to maintain 5−mil PCB routing you must choose either the analog output or the DMIC output as routing both will require a 3−mil trace/space PCB. If you intend on using the DMIC output, you will need to switch to external clocking (see design note 6).
3. Two analog outputs are not possible on the 26−ball device. If you require two analog outputs or one differential analog output, you must use reference design 4, which uses the 30−ball package.
4. The analog output (A_OUT1) has an impedance of approximately 3 kΩ. You cannot use this pin to directly drive a speaker or headphone. If you wish to drive a speaker or headphone an audio amplifier is required.
5. The power supply of BelaSigna R262 must be between 1.8 V and 3.3 V.
6. Internal clocking is recommended if you only require an analog output (the DMIC output requires an external clock). If you intend on routing and using the DMIC output instead, you must provide a valid external clock on the EXT_CLK (A3) pin and the custom application

loaded onto the device must configure the clocking appropriately as part of the device configuration. Refer to the BelaSigna R262 Communications & Configuration Guide for details.

7. It is recommended that, at a minimum, the I^2C port be routed to a set of test points with 10 kΩ pull−up resistors to VBAT.

Reference Design 4 – Full Feature Set

Reference design 4 is the only reference design using the 30−ball package of BelaSigna R262. With the 30−ball package any design scenario is possible. This reference design uses the analog outputs but you can also use the DMIC output; see the design notes for details.

The microphones on this reference design are biased by the VMIC (E3) pin. This is the ideal situation as the voltage on this pin is software controlled and can be set to any one of VREG (1.0 V), VDDA (2.0 V), VSSA (analog ground) or High−Z (high−impedance). The default setting is VDDA, and this pin is automatically powered off when BelaSigna R262 is put to sleep.

This reference design is configured with CLOCK_SEL set to preset 7 (R1 = DNI, R5 = 10 kΩ) and therefore uses the internal oscillator, avoiding the need for an external clock. An external clock could be used if desired with a minor configuration change. Refer to the BelaSigna R262 datasheet and the BelaSigna R262 Communications & Configuration Guide for details.

With this reference design it is possible to output two processed channels simultaneously on two single−ended analog outputs, or configure the device for a single processed channel output as a mono, differential analog signal. This is accomplished by configuring the CHAN_SEL voltage divider using one of the configurations in Table 7.

The algorithm outputs are configured using a combination of the BOOT_SEL and ALPHA_SEL configuration pins, as described in Table 9 – *BOOT_SEL and ALPHA_SEL Resistive Divider Values for Intermediate Noise Reduction Tunings*, found in Appendix A. Refer to the design notes for more details on the possible algorithm output combinations.

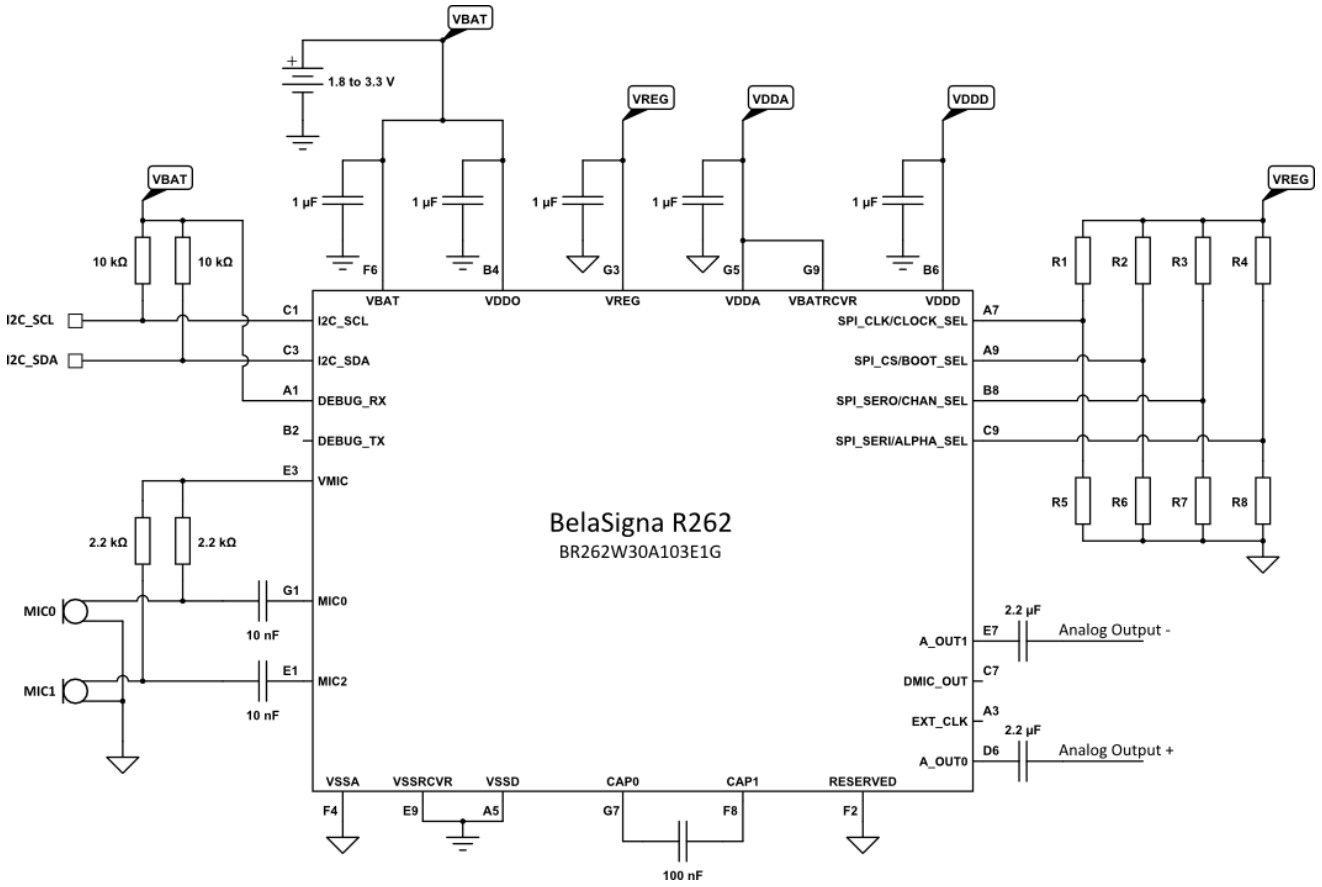

Figure 9. Schematic for Reference Design 4

Table 7. REFERENCE DESIGN 4 CHAN_SEL RESISTIVE DIVIDER VALUES

CHAN_SEL		Preset	Noise Reduction Outputs	Channel 0	Channel 1
R3	R7				
100 kΩ	39 kΩ	5	Single (Mono differential)	Mixed Output (as per BOOT_SEL and ALPHA_SEL)	N/A
75 kΩ	100 kΩ	3	Dual (Stereo, single−ended)	Start of range (as per BOOT_SEL)	Mixed Output (as per BOOT_SEL and ALPHA_SEL)

Design Notes for Reference Design 4

The following notes apply to reference design 4:

1. The analog outputs (A_OUT0 and A_OUT1) have an impedance of approximately 3 kΩ. You cannot use these pins to directly drive a speaker or headphone. If you wish to drive a speaker or headphone an audio amplifier is required.
2. The power supply of BelaSigna R262 must be between 1.8 V and 3.3 V.
3. Internal clocking is recommended if you only require analog outputs (the DMIC output requires an external clock). If you intend to use the DMIC output instead, you must provide a valid external clock on the EXT_CLK (A3) pin and the clocking settings for the device must be configured appropriately using the CLOCK_SEL voltage divider. Refer to the BelaSigna R262 datasheet and the BelaSigna R262 Communications & Configuration Guide for details.
4. Other CHAN_SEL configurations are valid for this reference design, but the two presented in Table 7 are the most practical and useful selections.
5. When CHAN_SEL is configured for preset 3, two different and simultaneous processed outputs will be provided to the output channels. Use the BOOT_SEL resistive divider to configure the “start” of range tuning on output Channel 0, and the BOOT_SEL and ALPHA_SEL resistive dividers combined will determine the mixed tuning on output Channel 1.
6. When CHAN_SEL is configured for preset 5, a single processed output will be available on output Channel 0, and the combined values of the BOOT_SEL and ALPHA_SEL resistive dividers will determine the characteristics of the mixed/intermediate tuning available on the output.
7. It is recommended that, at a minimum, the I^2C port be routed to a set of test points with 10 kΩ pull-up resistors to VBAT.

Design Validation

This section discusses various methods to verify that your BelaSigna R262−based design is functioning correctly.

Verifying Operation

If you chose one of the reference designs that are configured entirely through hardware (reference designs 1, 2, and 4), the chip should output processed audio when it is powered on and this should be easy to verify. If your design depends on a custom application being loaded from EEPROM, again the device should output audio when power is applied, after the device has bootloaded from the external EEPROM provided you have obtained a valid custom application from ON Semiconductor and loaded this application onto the EEPROM.

If your design involves an I^2C master device that will send commands or a custom application to BelaSigna R262, you can easily verify operation by issuing the **Get Application Status** command. This will return two status bytes indicating the current state of BelaSigna R262. The specific format of these two status bytes is discussed in detail in the BelaSigna R262 Communications & Configuration Guide.

ON Semiconductor provides a utility called the BelaSigna R26x Tweaker that communicates with a connected BelaSigna R262 chip over I^2C, allowing you to verify settings, change parameters, and generate EEPROM images using one of a set of provided custom applications as a base application. The BelaSigna R26x Tweaker runs on Microsoft Windows® XP, Windows Vista, or Windows 7. You will also need a USB port and either an ON Semiconductor Communication Accelerator Adaptor (CAA), or an Aardvark I^2C/SPI Host Adaptor from Total Phase (Note 1) to communicate with BelaSigna R262 using this software.

The remainder of this section will discuss various aspects of the BelaSigna R26x Tweaker, but keep in mind that if you are using hardware−based configuration of BelaSigna R262, this utility is not strictly required.

If you are using the BelaSigna R26x Tweaker software, the first thing you will do is connect to the BelaSigna R262 device. This is done by executing the software, choosing a valid communications configuration, and clicking the **Connect** button. Provided you have a valid I^2C connection to the chip, you should see the results of the **Get Application Status** command as shown in Figure 10; although the value of the status word will depend on how BelaSigna R262 has been integrated into your particular design.

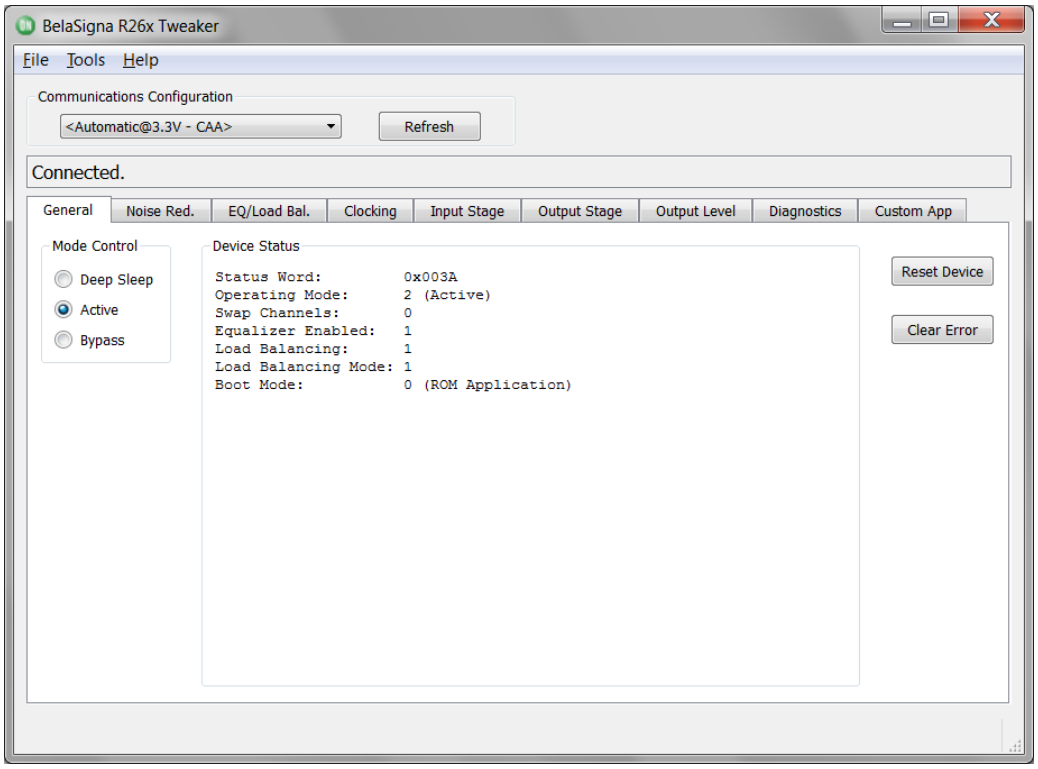

Figure 10. BelaSigna R26x Tweaker General Tab

1. http://www.totalphase.com/products/aardvark_i2cspi/

BelaSigna R262 Operating Modes

Before using the BelaSigna R26x Tweaker, it is useful to briefly discuss the different operating modes of BelaSigna R262. There are four main operating modes that you need to be aware of: Standby, Deep Sleep, Active, and Bypass. Deep Sleep mode and Standby mode are essentially the same from the perspective of the Tweaker software because when the chip is sleeping, any I^2C traffic will put the chip into Standby mode. Refer to the BelaSigna R262 Communications & Configuration Guide for more information on Deep Sleep and Standby modes.

The only two useful modes from the perspective of the Tweaker software are Active and Bypass mode. It is very important to note that both of these modes maintain a completely separate list of system settings stored in what is known as a context table. As a result, when you change the value of a particular setting in Bypass mode using the Tweaker, this setting is changed for Bypass mode only. If this is intended to be a global change for all modes, you need to switch the chip to Active mode and make the same setting change.

Configuring the BelaSigna R262 Noise Reduction Algorithm

The noise reduction algorithm inside BelaSigna R262 is quite flexible in terms of tuning and routing. It is possible to execute the algorithm in single or dual mode, the latter giving two completely independent noise−reduced outputs. Configuring the algorithm for dual mode is like having two conventional noise reduction algorithms executing in parallel with completely separate tunings. Examining all of the different algorithm configuration combinations can be slightly overwhelming, so this section will attempt to clarify what all of these options mean boiling this down to a fairly succinct set of configuration combinations. Refer to the BelaSigna R262 Communications & Configuration Guide for a more detailed explanation of all of the possible algorithm configuration settings.

In its most basic form, the advanced noise reduction algorithm running in BelaSigna R262 is an algorithm that takes a set of parameters. A particular set of parameters tuned for a specific use case (a mobile phone, for instance) is called a tuning. BelaSigna R262 has three core tunings built into its ROM: close−talk, near−talk, and far−talk. The algorithm also can accept any “custom” parameter set tuned for a specific use case. But what makes BelaSigna R262 so flexible is the fact that it essentially has the ability to run two noise reduction algorithms in parallel, both of which can be configured with a unique set of parameters. In addition to this, BelaSigna R262 is able to take two sets of parameters, along with a mixing ratio, and generate a third, interpolated parameter set or tuning. Taking the two specified parameter sets as endpoints along a continuum, the final interpolated tuning is somewhere along this continuum between the start and end points, as defined by the mixing ratio. Because the end points can be any one of the three core parameter sets, or a completely custom parameter set, the possibilities for the interpolated parameter set are limitless.

A block diagram describing the fundamental parts of the noise reduction algorithm is shown in Figure 11. You can see from this block diagram that you can configure the parameter sets for the two parallel noise reduction channels; and there is also a fundamental processing mode that can be re−configured by loading a custom “common” parameter set. Currently, the only supported fundamental processing mode is “fully adaptive” and changing fundamental algorithm modes is not supported.

Figure 11 also clearly shows how, through the use of the mixer, there are three possible processed outputs that can be routed to the two output channels, depending on how the chip is configured.

To better visualize how parameter set interpolation works, the three core tunings have been plotted on an axis of talking distance versus noise reduction performance (SNR improvement) in Figure 12. This graph is meant for conceptual purposes only as it is not to scale (and the true interpolation lines may not be entirely linear). There are other characteristics that can be measured for a given parameter set; talk distance and SNR improvement are just two parameters used to quantify a given tuning. There are three dashed lines on the graph shown in Figure 12 representing the tuning continuums available using the three core parameter sets (close−, near−, and far−talk) as the endpoints for the mixing process. The mixing ratio then defines where on this line the final, mixed parameter set will fall in terms of talk distance and noise reduction performance (SNR improvement). Three points (A, B, and C) have been plotted on these lines along with their respective mixer start, mixer end, and mixing ratio values.

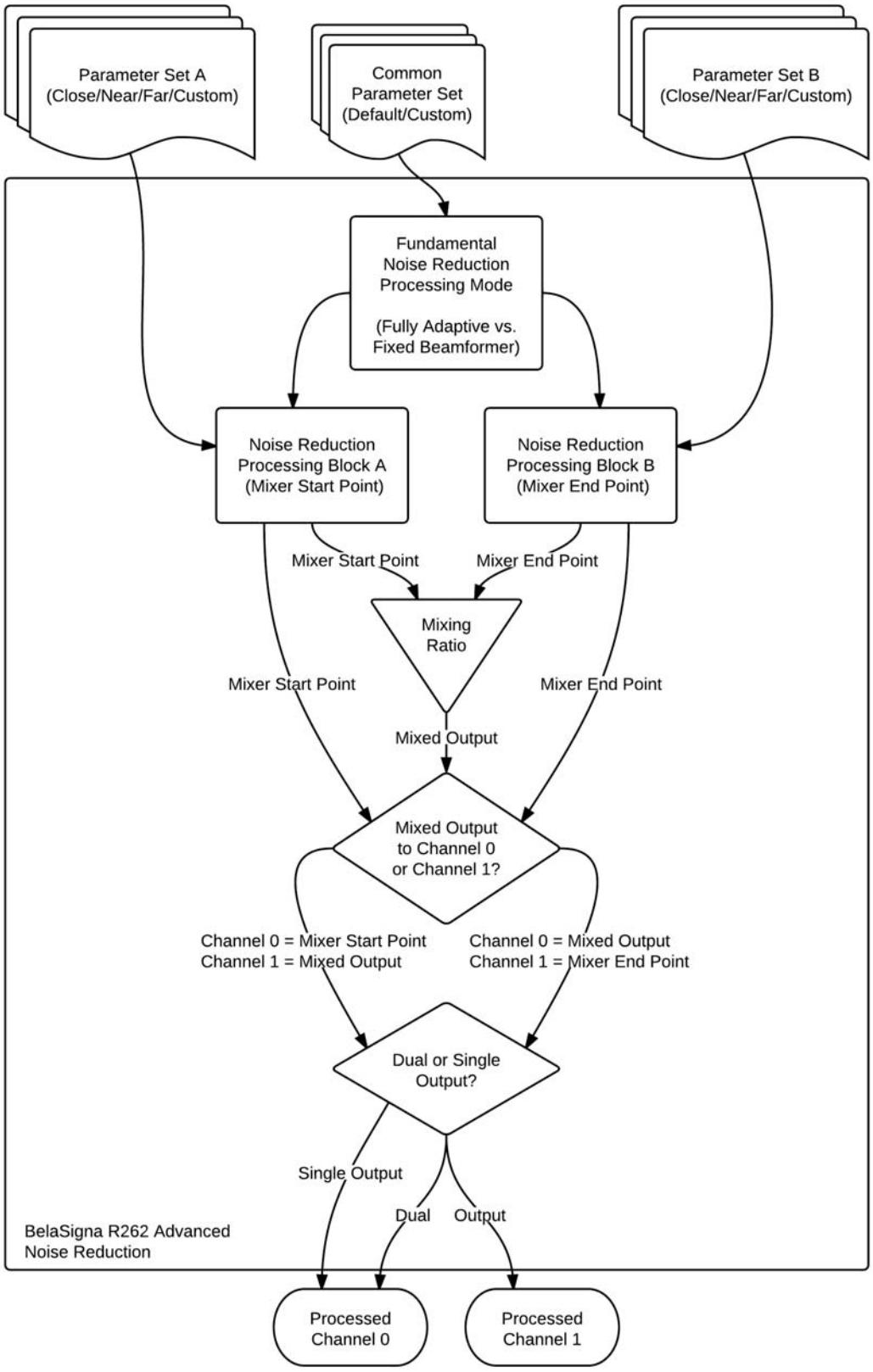

Figure 11. BelaSigna R262 Noise Reduction Block Diagram

Parameter mixing is entirely optional. It is possible to run the algorithm with just a single processed output that uses one of the core modes. How you choose to configure the noise reduction algorithm depends on the requirements of your specific application.

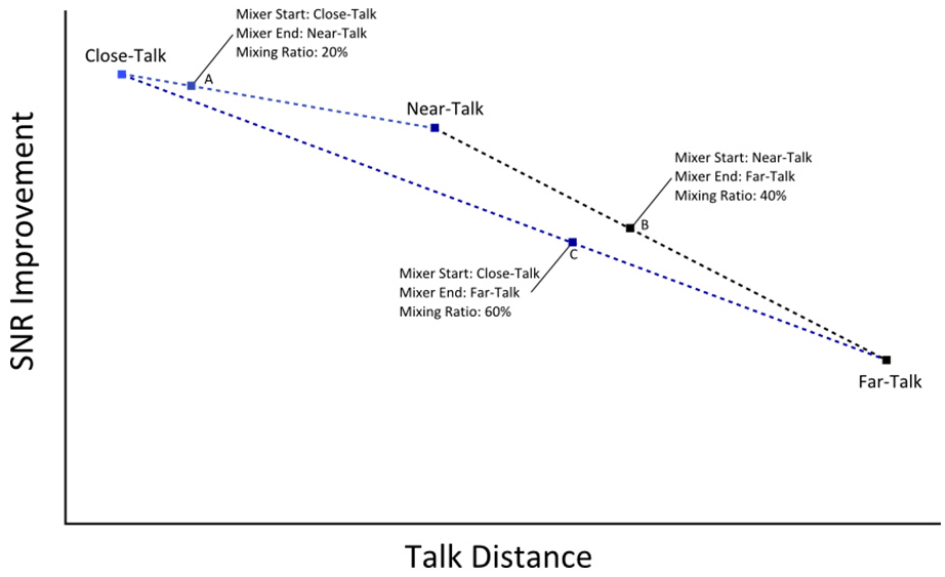

Figure 12. Using the Mixer for Interpolated Tunings

The noise reduction parameter sets defining the start and end points for the mixer do not need to be one of the three core modes. It is also possible to specify a completely custom tuning for one or both of the start and end points, provided you are willing to add an EEPROM to your design or have the ability to load parameters over I^2C from an external master device. This can effectively transform the ranges of possible tunings as shown in Figure 13, making the tuning possibilities truly limitless.

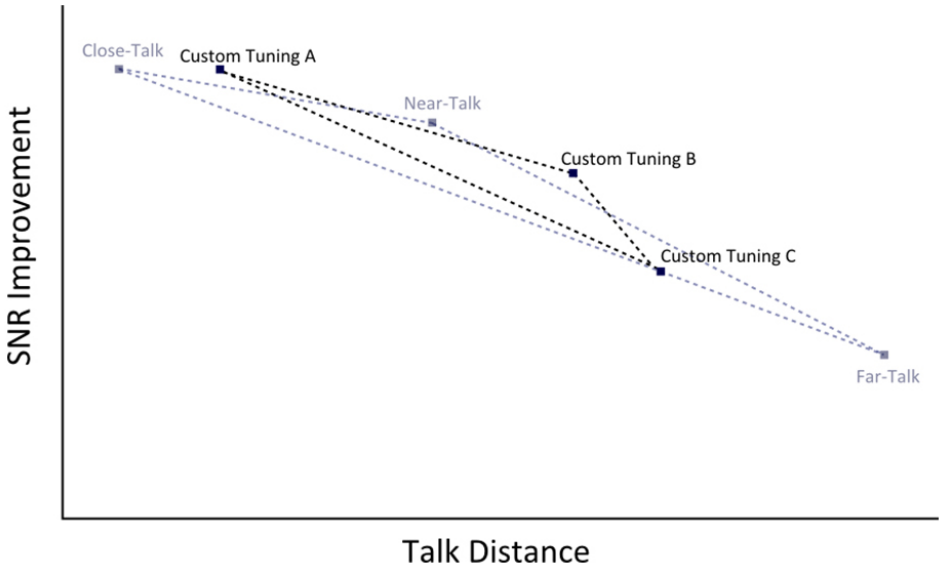

Figure 13. The Mixer and Custom Parameter Sets

Configuring Noise Reduction Using the BelaSigna R26x Tweaker

The **Noise Reduction** tab in the Tweaker software is shown in Figure 14. While this dialog is fairly dense, it delivers an incredible amount of control into a concise and easy−to−use interface. The first section of controls inside the **Noise Reduction Configuration** group box allows the main noise reduction processing mode to be set. There are two fundamental modes: Fully Adaptive and Fixed Beamformer. There is also provision for saving and loading custom “common” parameter sets, effectively altering the behavior of the fundamental processing mode. Currently, the only supported mode is **Fully Adaptive**, with the **Default (ROM−based, Adaptive)** parameter set. Other processing modes and common parameter sets are not supported in Tweaker revision 3.3.6.

The remainder of the controls grouped as **Channel Parameter Set Configuration** is where the configuration of the channel−specific noise reduction tunings, mixed channel configuration, and noise reduction channel routing take place. These settings correlate to the noise reduction block diagram shown in Figure 11. The Channel 0 and Channel 1 parameter sets allow the mixer start and end point to be set. The three core modes are available in the drop−down menu, and any valid custom parameter sets present in the Tweaker `tunings` folder are presented as well. Clicking one of the **Save...** buttons allows the current parameter set to be read from chip memory and saved to a file.

The **Mixed Channel Parameter Set** slider and text entry field allow you to set the mixing ratio. You can use the slider, or type a hexadecimal value directly into the text field. Keep in mind that the only valid mixing ratio settings available when configuring the device purely via LSADs are 0%, 20%, 40%, 60%, 80%, and 100%. Using the page−up and page−down keys, or clicking beside the slider thumb will increment the mixing ratio slider by 20% increments.

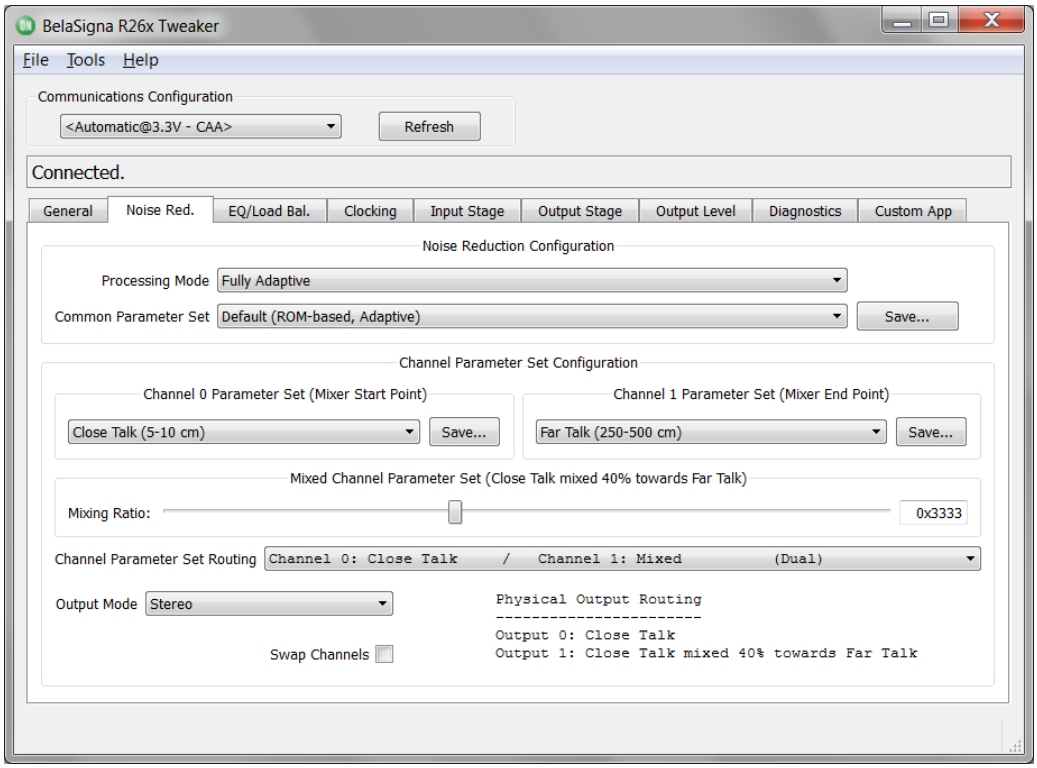

Figure 14. BelaSigna R26x Tweaker Noise Reduction Tab

The **Channel Parameter Set Routing** drop−down control actually configures two aspects of the noise reduction block. It allows you to configure dual versus single output, at the same time specifying which channel's parameter set is replaced with the mixed output. The options presented in this drop−down list follow the selected values for the mixer start and end point, making things much more intuitive.

For convenience, the **Output Mode**, and **Swap Channels** controls are also presented on the noise reduction tab, allowing you to quickly configure the output stage routing as it relates to the two channels output by the noise reduction block. Additionally, perhaps the most useful part of the noise reduction tab is the text in the bottom right corner entitled **Physical Output Routing**. This takes into account all of the noise reduction settings and relevant output stage routing settings, and shows you at a glance exactly what audio signals will be present on the two physical outputs of the chip.

Generating Custom Application EEPROM Images with the BelaSigna R26x Tweaker

The main function of the BelaSigna R26x Tweaker application is to configure and generate custom EEPROM images for BelaSigna R261 and BelaSigna R262 devices. While BelaSigna R262 is intended to be extremely flexible, allowing simple hardware−only configuration for most scenarios, there are some cases where an external EEPROM containing a custom application may be beneficial; for example, if you would like to use specific input stage settings to match atypical microphones, use a completely custom noise reduction tuning, or perhaps take advantage of the PCM interface.

Custom applications must be obtained from ON Semiconductor. There are a group of custom applications distributed with the Tweaker software included in the `custom_apps/BR262` folder which should meet the needs of most custom scenarios. A description of the custom applications provided with the Tweaker is presented in Table 8.

Table 8. BELASIGNA R262 CUSTOM APPLICATIONS

Custom Application	Description
BR262_Custom_App_GPIO_Mode_Swap	When SWAP_CHAN (pin C5) is momentarily pulled low, the current mode is toggled between Active and Bypass mode This custom application is configured for external clocking at 2.048 MHz
BR262_Custom_App_GPIO_Mode_Swap_Internal_Clock	When SWAP_CHAN (pin C5) is momentarily pulled low, the current mode is toggled between Active and Bypass mode This custom application is configured to use the internal clock
BR262_Custom_App_Standard	When SWAP_CHAN (pin C5) is momentarily pulled low, the output channels are swapped This custom application is configured for external clocking at 2.048 MHz
BR262_Custom_App_Standard_Internal_Clock	When SWAP_CHAN (pin C5) is momentarily pulled low, the output channels are swapped This custom application is configured to use the internal clock

To configure a custom application, one must be resident in the RAM of the BelaSigna R262 device. If the device has not already booted from EEPROM loading a custom application to RAM, you must first download one using the Tweaker. With the Tweaker connected to a chip, select the **Tools > Download File...** menu option, and choose the desired custom application to download to the device. You can download either the EEPROM image (`.eeprom`) if an I^2C or SPI EEPROM is currently attached to the BelaSigna R262 device, or the `.abs` file (an executable that is downloaded to RAM only). Either choice is valid and will allow you to proceed to the next step: configuring and exporting a new customized EEPROM image. Also note that instead of using the **Tools > Download File...** menu, you can simply drag and drop an `.eeprom` or `.abs` file onto the Tweaker user interface to download a file to the device.

When BelaSigna R262 is executing a custom application, the **General** tab indicates this by showing the custom application ID and version number as shown in Figure 15. In this case, the Tweaker indicates that a custom application with an ID of 1 and a version of 0x2002 (2.0.2) has been loaded into the device by an I^2C master (an external host or the Tweaker itself).

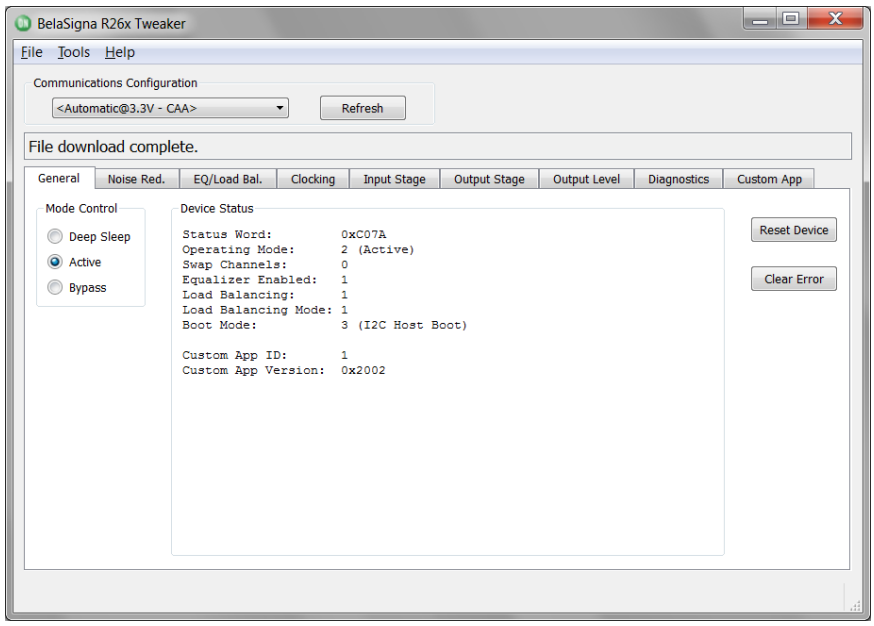

Figure 15. BelaSigna R26x Tweaker Custom Application Version

Once you have confirmed BelaSigna R262 is executing a custom application by checking the ID and version on the **General** tab, you should confirm that the Tweaker can successfully detect exactly which custom application is executing by switching to the **Custom App** tab. This is a good sanity check since there could be many versions of the same custom application. Also, ON Semiconductor may issue updates to existing custom applications, or release additional custom applications providing new functionality.

When you switch to the **Custom App** tab, the Tweaker will attempt to match the ID and version (along with some other key information) to one of the existing custom applications in the `custom_apps/BR262` folder. If no match is found you will be shown a message indicating that the connected device either does not contain a custom application (in which case you neglected to download one as described earlier), or that the detected custom application does not match any of the applications in the `custom_apps/BR262` folder. In this case, it is possible you are using an updated or newly−released custom application that is not in the `custom_apps/BR262` folder by default. If you have received an updated custom application from ON Semiconductor and it is not in the `custom_apps/BR262` folder, copy the custom application files you received into this folder, and then click **Refresh** while on the **Custom App** tab in the Tweaker to restart the custom application detection process.

If a match is found, the matching file name along with a dump of all of the context tables and relevant memory locations on the chip will be shown in the text box on the **Custom App** tab as shown in Figure 16. You will also be allowed to export the current chip state to a new custom application, also generating an EEPROM image if desired by clicking on the **Export Custom App**... button.

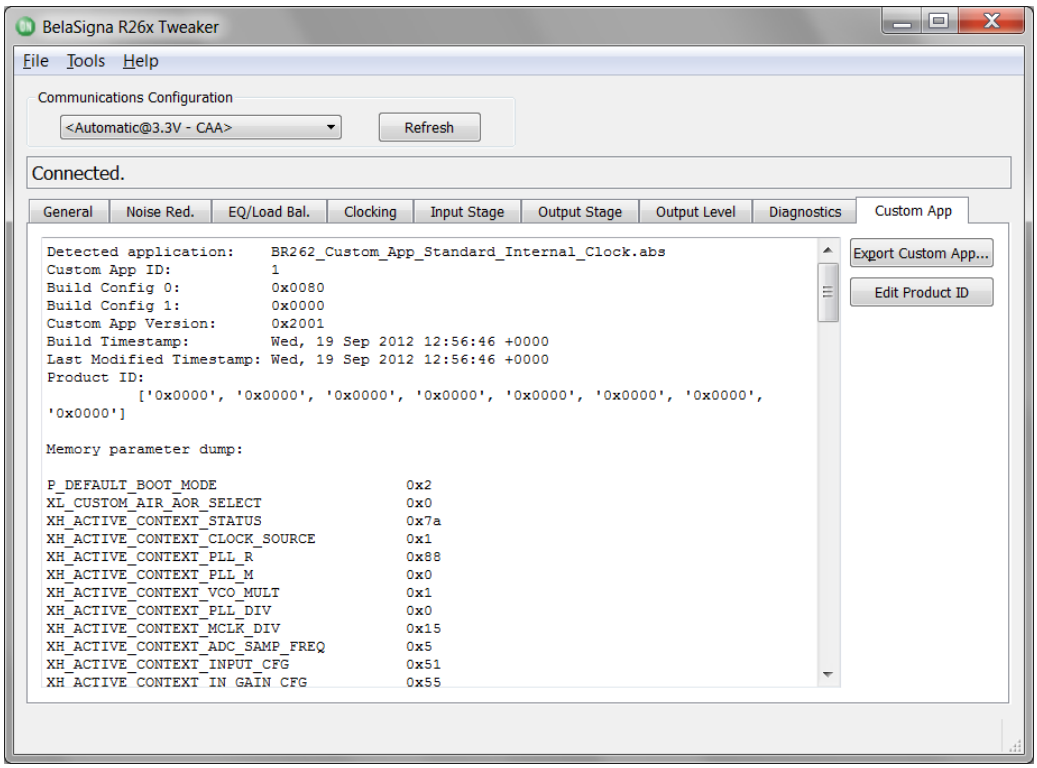

Figure 16. BelaSigna R26x Tweaker Custom Application Tab

Once you have confirmed that a valid custom application recognized by the Tweaker is executing on the device, you are free to make any changes you like on any of the other tabs. Keep in mind both Bypass mode and Active mode have a separate group of settings stored in their own context table. Thus, any change you make while in Active mode will also need to be made while in Bypass mode if you intend on that particular setting to be a global change. Once you are finished making changes to the system settings, and are happy with the results, then go back to the **Custom App** tab to export the current chip state to a new custom application.

Before exporting the new application, you may want to adjust the Product ID. The Product ID is a series of bytes stored in the custom application that can be used to identify a specific version of your product. You can change its value by clicking on the **Edit Product ID** button from the **Custom App** tab, and typing in any ASCII characters you like into the dialog box provided. This will update the Product ID in RAM, and the customized value will also be exported when you generate a new custom application or EEPROM image.

To export a custom application, click on the **Export Custom App...** button from the **Custom App** tab. You will be presented with the dialog shown in Figure 17.

Click the **Browse...** button to choose a directory and filename for your new custom application. By default, an EEPROM image is also created. You can also choose to download this EEPROM image immediately to the device's attached EEPROM, and whether or not you want to open a file explorer window to view the exported files. Once this export operation is complete, you can use the resulting EEPROM image to program your customized application into other BelaSigna R262 devices.

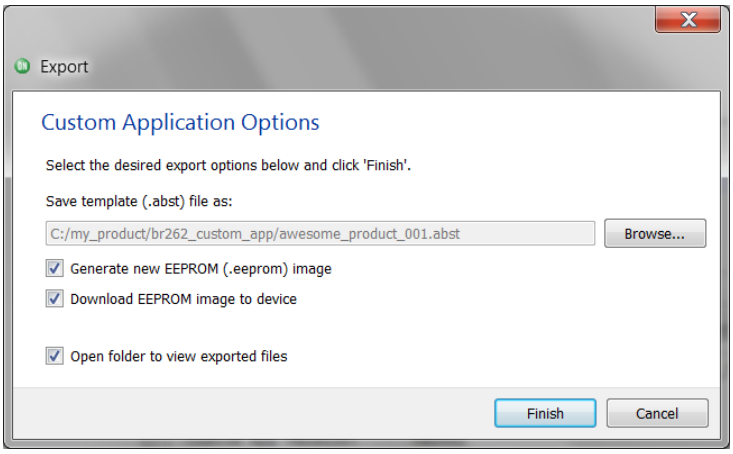

Figure 17. BelaSigna R26x Tweaker Exporting a Custom Application

Calibrating Signal Levels

Another useful step when designing a new product involving BelaSigna R262 is to validate the signal levels into and out of the chip and to characterize the gain through the system. The **Input Stage** and **Diagnostics** tabs allow you to determine the best input level based on your specific microphones, while the **Output Stage**, **Output Level**, and **Diagnostics** tabs in the Tweaker can be used to quickly determine the downstream system gain.

To validate the input levels:

1. First, ensure you have a valid connection to the chip.
2. Next, go to the **Diagnostics** tab and ensure that the **Mode Selection** drop−down is set to **Bypass Mode**, and that the **Realtime Monitor** check box is checked.
3. This puts the chip into Bypass mode and shows you a real−time graph of the input signals. You can now speak into the device and monitor the input levels, particularly the **Maximum** values for both channels.

Now, holding the device as you would in normal operation, subject it to the loudest speech levels you would expect to see under typical operating conditions. You should see that the FIFO (i.e. the digital representation) energy for both channels does not approach the maximum level (0 dBFS) as shown in Figure 18. Ideally you will want to keep the maximum level below approximately −3 dBFS. If you find your signal levels to be extremely low (yelling into the microphones results in little to no maximum FIFO energy) you may have a bad microphone, or the preamplifier gain on the input stage is too low for your microphone. The input stage is pre−configured with 24 dB of preamplifier gain for a typical microphone sensitivity of approximately −42 dB (where 0 dB = 1 V/Pa, @ 1 kHz). If you have a microphone that is less sensitive than this (unlikely) you might need to increase the preamplifier gains on the **Input Stage** tab. A more likely scenario is that you are using more sensitive microphones and as a result see maximum FIFO energies that saturate the input stage (i.e., reach 0 dBFS). If this is the case, you can reduce the preamplifier gains on the **Input Stage** tab. A good rule of thumb is that if your microphone is 20 dB more sensitive than the standard value of −42 dB (in other words your microphone has a sensitivity of −22 dB), then you should decrease your preamplifier gain by approximately the same amount.

Any change to the input stage settings will require a custom application stored in an external EEPROM, or an external I^2C master to configure BelaSigna R262 as these settings are not exposed for configuration via LSADs. For this reason it is best to use standard −42 dB microphones, if possible.

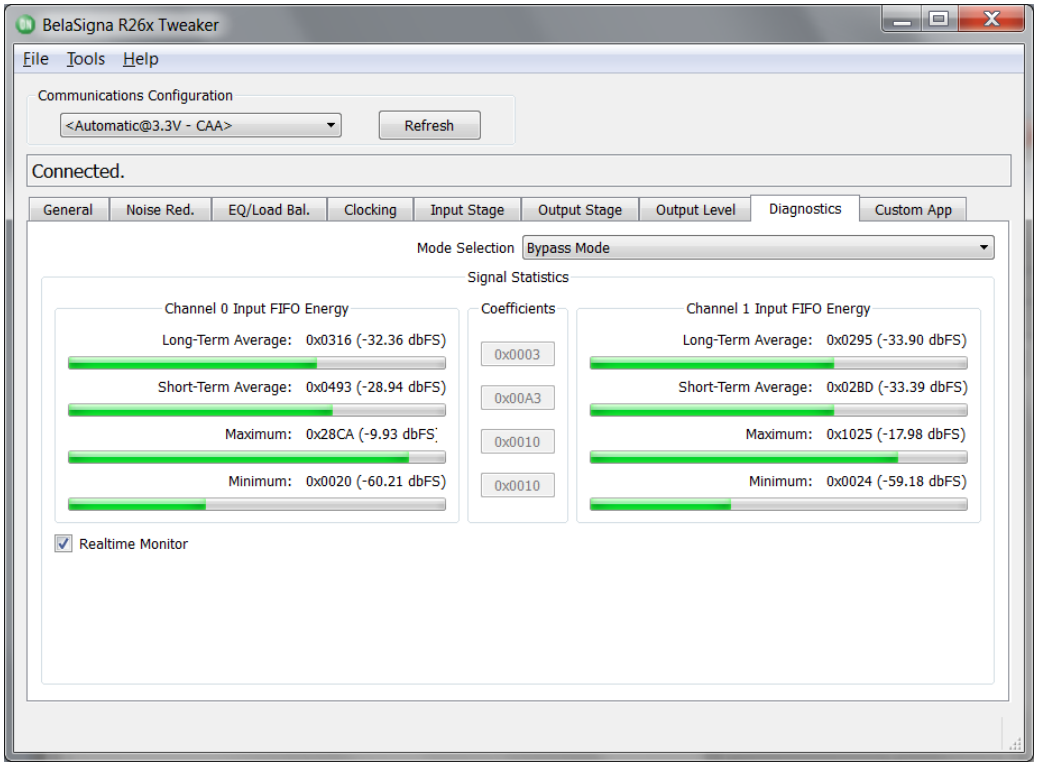

Figure 18. BelaSigna R26x Tweaker Realtime Input Monitoring

In many situations, determining the system gain after the output of BelaSigna R262 is also useful and can be accomplished by:

1. First, ensure you have a valid connection to the chip.
2. Next, go to the **Diagnostics** tab and switch the chip into **Bypass/Diagnostic Mode: Output 1 kHz Tone** mode using the **Mode Selection** combo box. This places the chip into Bypass mode, and plays a 1 kHz sine wave of a known magnitude (6 dB below full scale) on both output channels. It is important to switch the chip into Bypass/Diagnostic mode first, as all of the other settings depend on the current chip mode (there are a separate group of settings for each mode: Active, Bypass, Standby, etc.).
3. Once in Bypass/Diagnostic mode, go to the **Output Stage** tab, ensure the **Output Mode** is set to **Stereo** and make a note of the **Analog Attenuation** setting for each channel. The default value is 0 dB.
4. Next, go to the **Output Level** tab and make a note of the **Total Gain** for each channel as a result of the output shift and master volume settings. The default value is −6 dB.

You can now determine the gain or attenuation applied after the output of BelaSigna R262 by making an audio recording on your particular device. For instance, if your design involves interfacing the analog output(s) of BelaSigna R262 to the microphone input of a baseband chip in a tablet or mobile phone, recording a WAV file on the device will allow you to calculate the gain or attenuation being applied by the baseband chip. The total gain or attenuation is calculated differently depending on whether you are using the analog output(s) or the DMIC output on BelaSigna R262. The two calculations are shown in Equation 1 and Equation 2.

Calculating External System Gain (Analog Outputs)

$$\text{External System Gain} = \text{Recorded Sine Level dBFS (pk\_pk)} + 6\text{ dB} + (\text{BelaSigna R262 Analog Attenuation}) - (\text{BelaSigna R262 Total Output Stage Gain}) \quad \text{(eq. 1)}$$

Calculating External System Gain (DMIC Output)

$$\text{External System Gain} = \text{Recorded Sine Level dBFS (pk\_pk)} + 6\text{ dB} - (\text{BelaSigna R262 Total Output Stage Gain}) \quad \text{(eq. 2)}$$

As an example, let's assume we are using the analog outputs and a recording of the diagnostic sine wave resulted in a level of −10 dBFS (pk−pk); and let's also assume the default values for analog attenuation (0 dB) and total output stage gain (−6 dB). Using Equation 1 we obtain an external system gain of (−10) + 6 + (0) – (−6), or 2 dB of external gain. Since we expected a sine wave at −12 dBFS (a sine wave at 6 dB below full scale, plus −6 dB output stage gain) and we observed a sine wave of −10 dBFS, this validates our calculation of 2 dB of system gain.

The calculation is similar when using the DMIC output, but the BelaSigna R262 analog attenuation is not included as this setting only applies to the analog outputs.

Using these equations, it is easy to see how much gain or attenuation is being applied after the output of BelaSigna R262, allowing you to adjust system settings to maximize dynamic range and avoid saturation.

Appendix A

Table 9. BOOT_SEL and ALPHA_SEL Resistive Divider Values for Intermediate Noise Reduction Tunings

BOOT_SEL			ALPHA_SEL			Mixed Output (Talk Distance)
R2	R6	Description	R4	R8	Description	
100 kΩ	39 kΩ	Preset: 5 Start of Range: Close−Talk (5 cm) End of Range: Near−Talk (100 cm)	10 kΩ	Do not install (DNI)	Preset: 0−2 Alpha = 0% (Start of range)	Close−Talk (5 cm to 10 cm)
			75 kΩ	100 kΩ	Preset: 3 Alpha = 20%	Close−Talk 20% towards Near−Talk (15 cm to 30 cm)
			100 kΩ	75 kΩ	Preset: 4 Alpha = 40%	Close−Talk 40% towards Near−Talk (25 cm to 50 cm)
			100 kΩ	39 kΩ	Preset: 5 Alpha = 60%	Close−Talk 40% towards Near−Talk (30 cm to 65 cm)
			100 kΩ	16 kΩ	Preset: 6 Alpha = 80%	Close−Talk 40% towards Near−Talk (40 cm to 80 cm)
			Do not install (DNI)	10 kΩ	Preset: 7 Alpha = 100% (End of range)	Near−Talk (50 cm to 100 cm)
75 kΩ	100 kΩ	Preset: 3 Start of Range: Near−Talk (50 cm) End of Range: Far−Talk (500 cm)	10 kΩ	Do not install (DNI)	Preset: 0−2 Alpha = 0% (Start of range)	Near−Talk (50 cm to 100 cm)
			75 kΩ	100 kΩ	Preset: 3 Alpha = 20%	Near−Talk 20% towards Far−Talk (100 cm to 200 cm)
			100 kΩ	75 kΩ	Preset: 4 Alpha = 40%	Near−Talk 40% towards Far−Talk (125 cm to 250 cm)
			100 kΩ	39 kΩ	Preset: 5 Alpha = 60%	Near−Talk 60% towards Far−Talk (150 cm to 350 cm)
			100 kΩ	16 kΩ	Preset: 6 Alpha = 80%	Near−Talk 80% towards Far−Talk (200 cm to 400 cm)
			Do not install (DNI)	10 kΩ	Preset: 7 Alpha = 100% (End of range)	Far−Talk (250 cm to 500 cm)

Note that preset 4 (R2 = 100 kΩ, R6 = 75 kΩ) is also a valid selection for the BOOT_SEL resistive divider in Table 9. This configuration is not listed because the two existing ranges covered by preset 5 (close−talk to near−talk) and preset 3 (near−talk to far−talk) when combined cover the same total range, with more granularity in adjustment via the ALPHA_SEL setting.

If the full range from close−talk to far−talk is desired in your application, you are free to configure the BOOT_SEL pin to use preset 4.

Company or Product Inquiries

For more information about ON Semiconductor products or services visit our Web site at http://onsemi.com .

Technical Contact Information

For technical support, email: dsp.support@onsemi.com

BELASIGNA is a registered trademark of Semiconductor Components Industries, LLC.
Windows is a registered trademark of Microsoft Corporation.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
Literature Distribution Center for ON Semiconductor
P.O. Box 5163, Denver, Colorado 80217 USA
Phone: 303−675−2175 or 800−344−3860 Toll Free USA/Canada
Fax: 303−675−2176 or 800−344−3867 Toll Free USA/Canada
Email: orderlit@onsemi.com

N. American Technical Support: 800−282−9855 Toll Free USA/Canada
Europe, Middle East and Africa Technical Support:
Phone: 421 33 790 2910
Japan Customer Focus Center
Phone: 81−3−5817−1050

ON Semiconductor Website: **www.onsemi.com**

Order Literature: http://www.onsemi.com/orderlit

For additional information, please contact your local Sales Representative