Renesas RA Family

Audio Processing Frond for Renesas Voice UI Name

Introduction

This application note aims to provide all necessary details for the operation and configuration of the Audio Processing Frond application in the Renesas RA family of voice kits.

Contents

1. Introduction

The Audio Processing Frond (APF) is an ultra-lightweight library of voice processing algorithms designed to enhance the recognition of voice commands recognition and speech quality when noisy conditions are met.

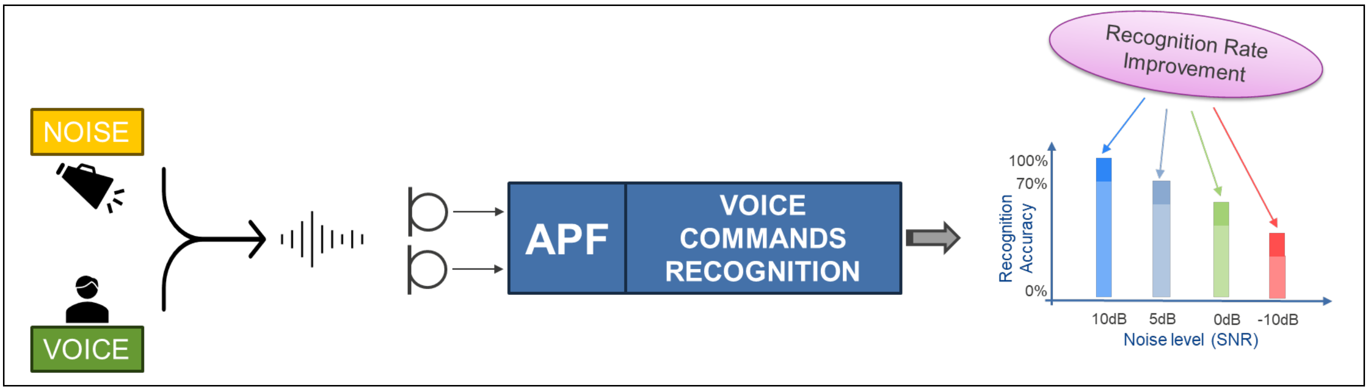


Figure 1. Recognition Rate improvement with APF

Table 1. Operating and Configuration Features

Operating Features:	Configuration Features:
Two microphones acoustic beamformer (BF) with configuration for endfire & broadside operation Four-band single channel noise suppression (NSUPP) Combined with a smaller size Dspotter level API for controllable operation and tuning Methodology to configure for other microphones' distance Operation as USB audio enumerated device	Activate/deactivate the complete AFE Select for NSUPP (1 mic) or BF/NSUPP (2 mic) operation Select between end-fire / broad-side BF configurations Configure noise level activation threshold Configure attack/decay and timeout thresholds

The implementation is focused on the ARM Cortex-M33 processor as integrated into the Renesas RA family of devices. All accompanying software projects are based on VOICE-RA6E1.

- CPU load = ~50MHz
- RAM = 180KB
- ROM = 14KB

Please visit "add weblink" for the simple example operating together with voice command recognition or contact the Reality AI CS team via rai-cs@dm.renesas.com to request additional support.

2. Description of Audio Processing Frond

The beamforming algorithms support both end-fire and broadside microphone configurations. The noise suppression algorithm supports a programmable target for the amount of stationary noise suppression.

The BF algorithm requires two identical microphones. Applications with two microphones can use the BF alone or the BF followed by the NSUPP. Single microphone applications can disable the BF and use the NSUPP stand-alone.

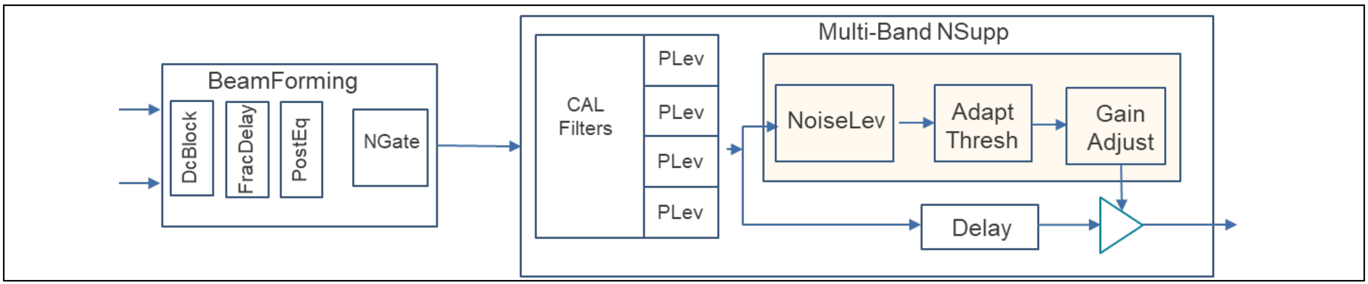


Figure 2. APF high-level drawing

2.1 Beamforming

This is a two-microphone static delay-sum beamformer. This means the location at which the maximum gain occurs and the location of the null are static with respect to the microphones. The algorithm does not adjust or track the sound source.

Omni-directional microphones should be used, with close to identical sensitivity characteristics. Performance is optimal when the microphone sensitivity is identical. Software-based calibration is also possible.

The beamforming algorithms support two different configurations: broadside (Fig. 2a) and end-fire (Fig.2b).

Broadside is most useful when the desired sound waves arrive perpendicular to the microphones. Sound waves arriving from the sides are attenuated.

End-fire configurations are most useful when the desired sound waves arrive parallel to the microphones. Sounds arriving from the back are attenuated.

Both configurations have certain microphone spacing/geometry limitations and frequency characteristics, which are described in further detail below.

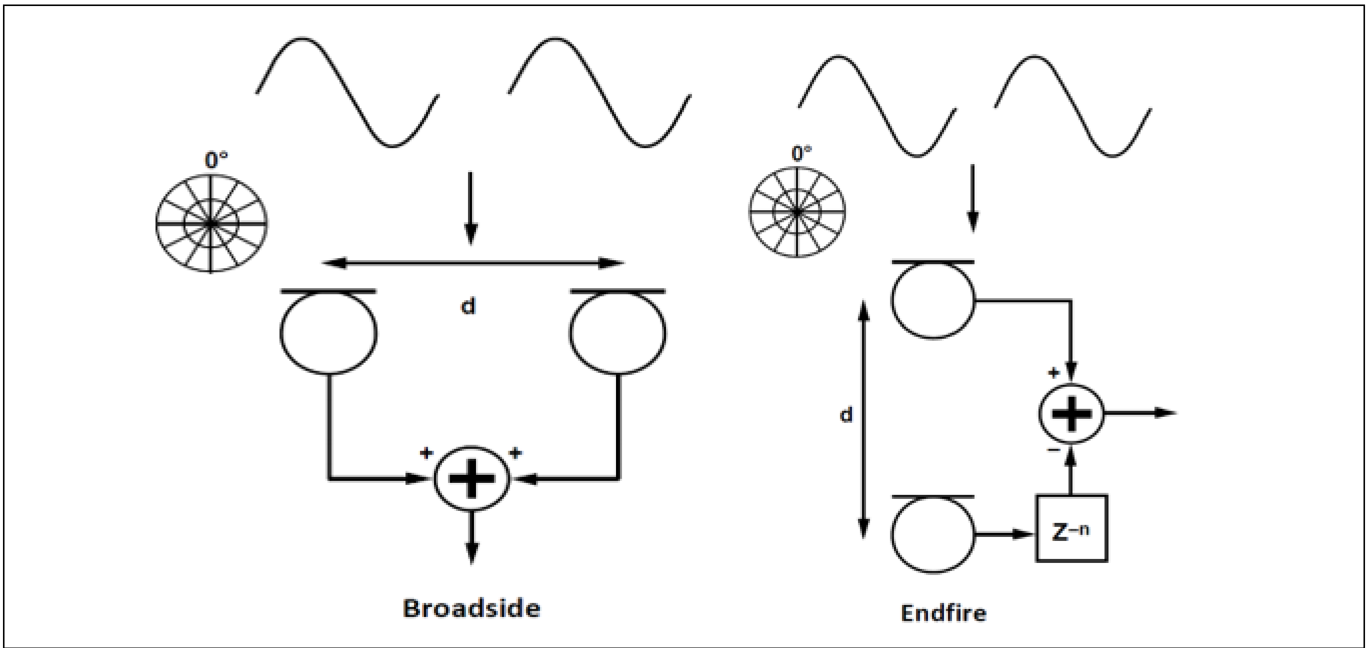


Figure 3. (a) Broadside and (b) EndFire microphones configurations

2.1.1 EndFire

An end-fire configuration is most useful when the desired sound is expected to arrive at 0 degrees (in front of) the microphones. The null, or location of maximum attenuation, is at 180 degrees or behind the microphones. This type of direction-sensitive pattern is known as a cardioid.

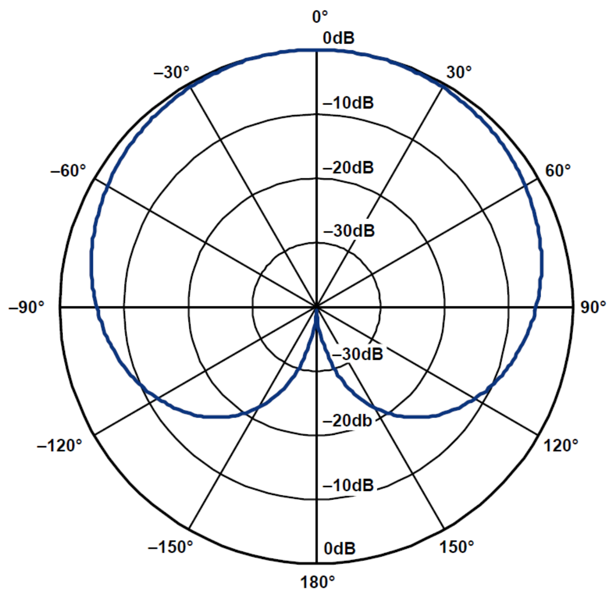


Figure 4. Response of a 2-microphone EndFire Cardioid Beamformer

2.1.2 Broadside

A broadside configuration is most useful when the desired sound source is perpendicular to the microphones, or “broadside”.

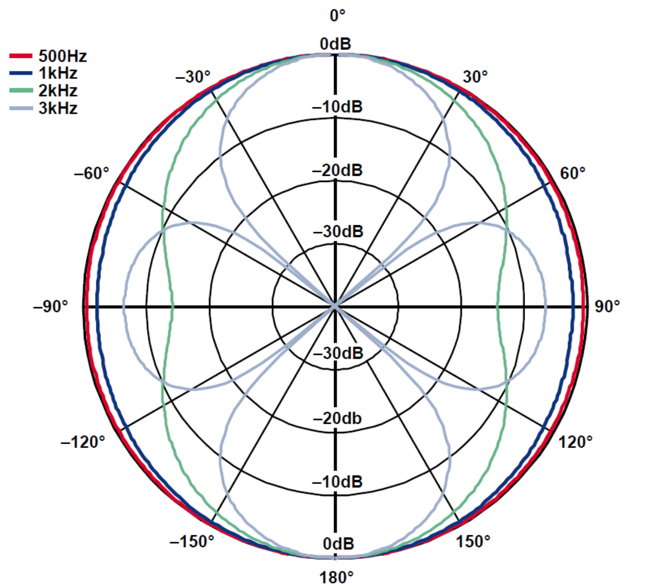


Figure 5. Response of a 2-microphone Broadside Beamformer

A broadside beamformer provides an SNR boost to sounds approaching the microphones at 0 degrees while attenuating sounds parallel (from the sides) of the microphones. It’s important to note that sounds behind the microphones (180 degrees) will not be attenuated; hence, broadside configurations are more useful in applications where sound is not expected from the back of the microphone array – such as a voice control panel mounted on the wall.

3. Noise Suppressor

NSUPP is a multi-band noise suppressor intended to suppress stationary noise in speech. It can also handle slow-changing cases of non-stationary noise.

A multi-band noise suppressor does noise suppression in several different frequency bands of the incoming speech signal.

Consider the speech signal in Figure 6 below:

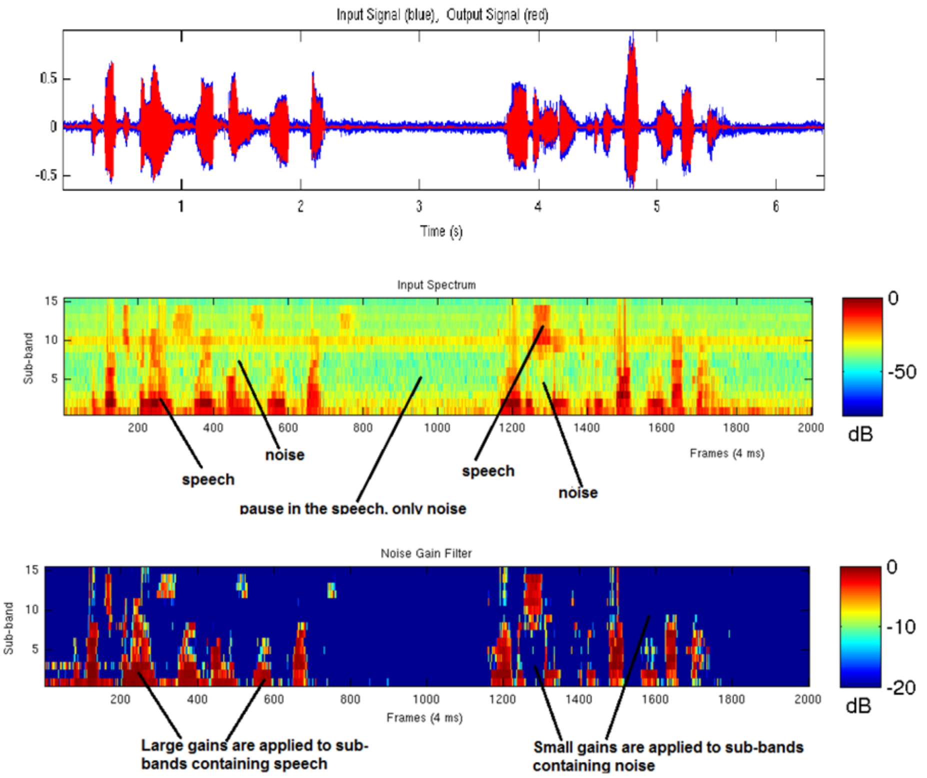


Figure 6. Speech Signal Containing Noise

The second window, in Figure 6, is a spectral view of the speech from the top window. Typically, the frequency components of speech are present in only a subset of the spectrum, whereas noise tends to be more broadband. By performing noise suppression in multiple bands, the noise can be isolated from the speech and removed, leaving the bands containing speech alone, hence improving the overall signal-to-noise ratio.

This implementation of multi-band noise suppression operates in the time domain and uses several filters to split the speech signal into different frequency bands.

In this time-domain approach, the signal is split into different bands using a multi-band filter construction. After that, noise suppression is carried out on the individual bands, after which the bands are summed back together, and the signal is reconstructed, as shown in Figure 7.

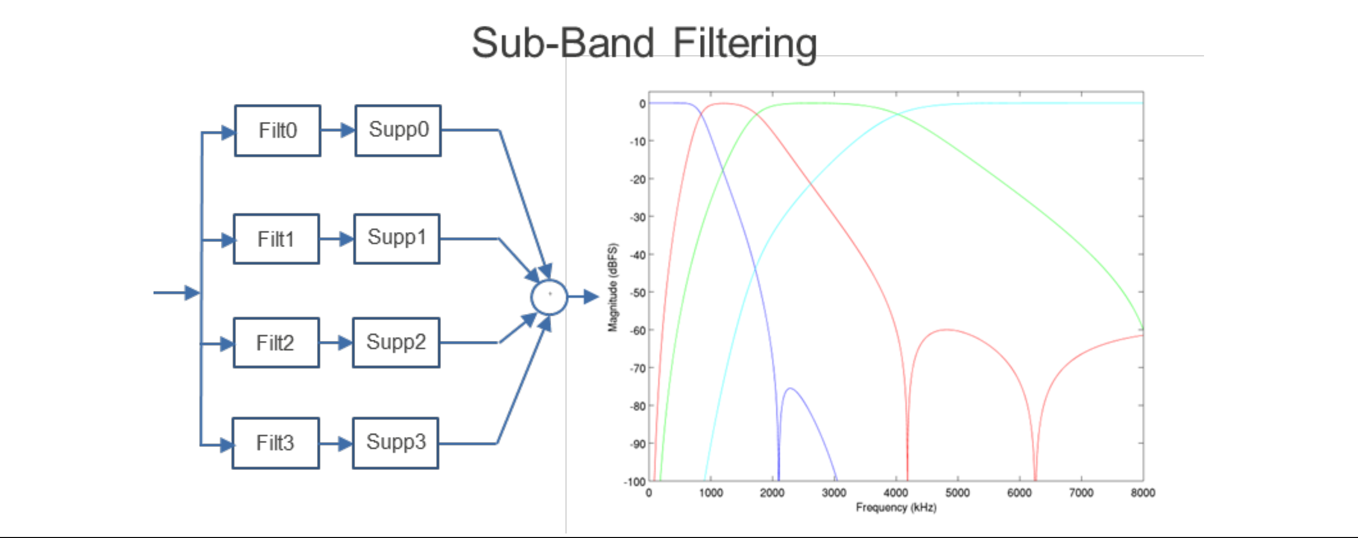


Figure 7. NSUPP sub-bands

Once the audio is separated into four sub-bands, power levels, and noise floor estimates are calculated for each sub-band. The power level estimate is a fast-updating average power calculation, similar to the RMS signal level. The noise floor estimate is a slow updating average of the minimum signal level measured during speech pauses over several seconds.

A threshold is then used to classify the signal as either speech or noise based on the instantaneous power level estimate. This threshold is dynamically updated, meaning that as the noise floor increases/decreases, so does this threshold. Finally, high-valued gains (near unity) are applied to speech, and appropriately low-valued gains are applied to the noise, as shown in Figure 8.

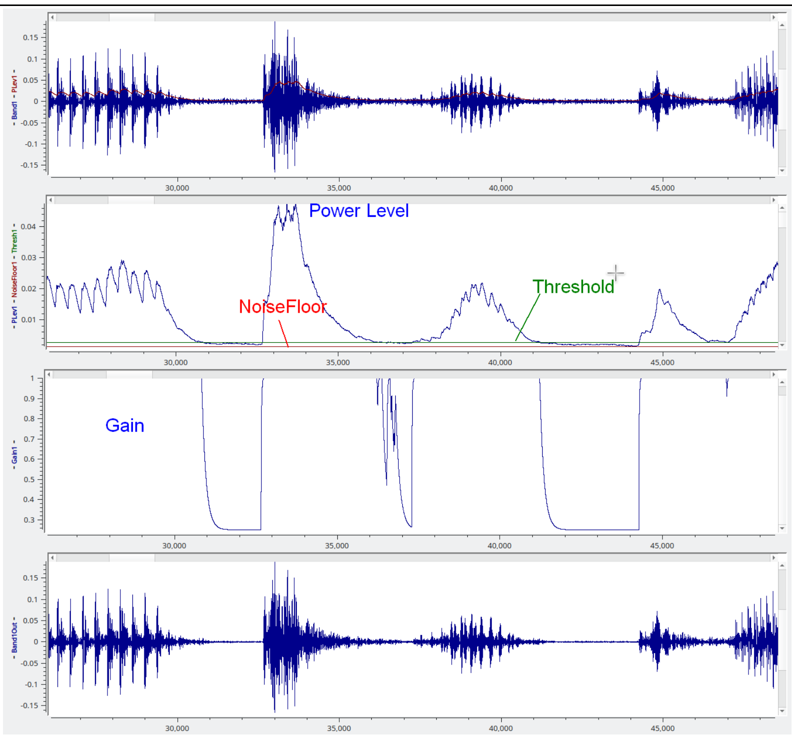


Figure 8. Classification of a sub-band as speech or noise

4. Configuration of Audio Processing Frond

The BFNSUPP library and example applications have several configuration options.

The complete AFE can be enabled/disabled at the application level.

4.1 Operating Mode

The BFNSUPP algorithms support a configuration API:

```
BfNSuppSetMode ((BfNSupp_t*)&bfNSuppData, int32_t mode);
```

Table 2. Operating Modes

Mode	Description
0	Bypass. Beamforming and Noise Suppression are disabled. The front microphone is passed “raw” from the input to the output
1	DC blocking only. DC blocking on the front microphone is passed to the output
2	DC blocking only. DC blocking on the back microphone is passed to the output
3	End-fire beamforming configuration with NSUPP (2 mic)
4	Broadside beamforming configuration with NSUPP (2 mic)
5	Beamforming is disabled. DC blocking and NSUPP (1 mic)

Note: BfNSuppSetMode() can only be used after BfNsupp is initialized. When not initialized, BfNSuppSetMode has no affect.

4.2 NSUPP Parameters

4.2.1 Programmable Gain Factor

This parameter modulates the amount of attenuation applied when the signal level is below the noise threshold. Larger values result in more attenuation, and smaller values less.

A typical range for "val" is between (0.5 min and 0.99 max)

```
extern void NsSetmaxGain(BfNSupp_t *p, double val)
```

4.2.2 Programmable Threshold Factor

This parameter modulates the threshold when classifying the signal as speech or noise. Smaller values result in more attenuation (both speech and noise), while larger values result in less attenuation.

A typical range for "val" is between (0.1 min and 0.5 max)

```
extern void NsSetNoiseThresh(BfNSupp_t *p, double val)
```

4.2.3 Programmable Noise Level Target

This parameter controls how gentle or aggressive the NSUPP attenuates noise.

Smaller values result in more aggressive attenuation, while large values are gentler. A value of 0 attenuates everything. A value of 0.99f provides 0dB of attenuation.

Typical range is between (0.06 min, 0.75 max).

```
extern void NSuppSetNoiseGainTrgt(BfNSupp_t *p, double val);
```

Beware that if NoiseGAinTrgt is set too small, the noise will be removed, and some unwanted attenuation of speech may occur.

4.2.4 Programmable Attack / Decay Time Constants

The attack parameter controls how fast attenuation is released to 0 dB when a sub-band contains the desired speech. The decay parameter controls how fast attenuation towards NoiseGainTrgt is applied when a sub-band contains noise.

```
extern void NSuppSetGainAttack(BfNSupp_t *p, double val);
extern void NSuppSetGainDecay(BfNSupp_t *p, double val);
```

A typical value for an attack is around 125 msec. A typical value for decay is around 15 msec.

4.2.5 Programmable Noise Floor Timeout

This parameter determines the length of time used to measure the minimum signal level for the noise floor. Typically, it's 0.5 to 1.5 seconds. A value too long will not respond fast enough to changing ambient noise conditions. A value too short will lead to rapid and annoying fluctuations in the amount of attenuation.

```
extern void NSuppSetnoiseCntFrame(BfNSupp_t *p, int32_t val);
```

4.2.6 Summary of Recommended Parameter Values

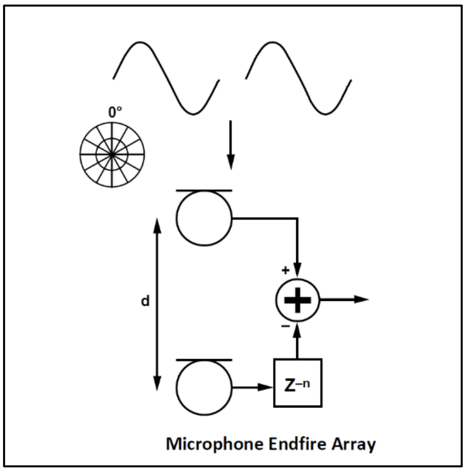
Table 3. Recommended Parameter Values

Parameter	Recommended Value
NsSetmaxGain	0.99
NsSetNoiseThresh	0.50
NSuppSetNoiseGainTrgt	0.25
NsuppSetGainAttack	0.125
NsuppSetGainDecay	0.016
NsuppSetnoiseCntFrame	16,000

4.3 End-fire MIC spacing and delay

With an end-fire configuration, the amount of delay must be chosen according to the microphone spacing. Also, spacing should be selected such that the location of the first null is above the desired audio content. The 21 mm (or below) is suitable for speech/music; up to 50 mm or slightly higher for voice.

The microphone spacing "d" is matched using a delay line of "n" samples, where n = d * Fs / c.



The location of the 1st null is C/(2*d)

d=10 - 50 mm

Fs=16 kHz

C=343 m/sec (speed of sound)

We consider 3 spacing choices:

n = 0.98 samples @ 21 mm (1st null at 8.2 kHz)

n = 1.45 samples @ 30 mm (1st null at 5.7 kHz)

n = 2.33 samples @ 50 mm (1st null at 3.4 kHz)

The software supports a programmable fractional delay element, where the amount of delay can be a fraction of a sample period.

End-fire configurations are high-pass, with a 6 dB/octave roll-off. The software supports a compensating post-filter used to equalize this roll-off, providing a flat frequency response. The default configuration is a suitable starting point for the typical voice command enhancements and noise reduction. Standards-compliant voice applications with more demanding performance may require fine-tuning these filters. Tuning guidelines are available in another application note upon request.

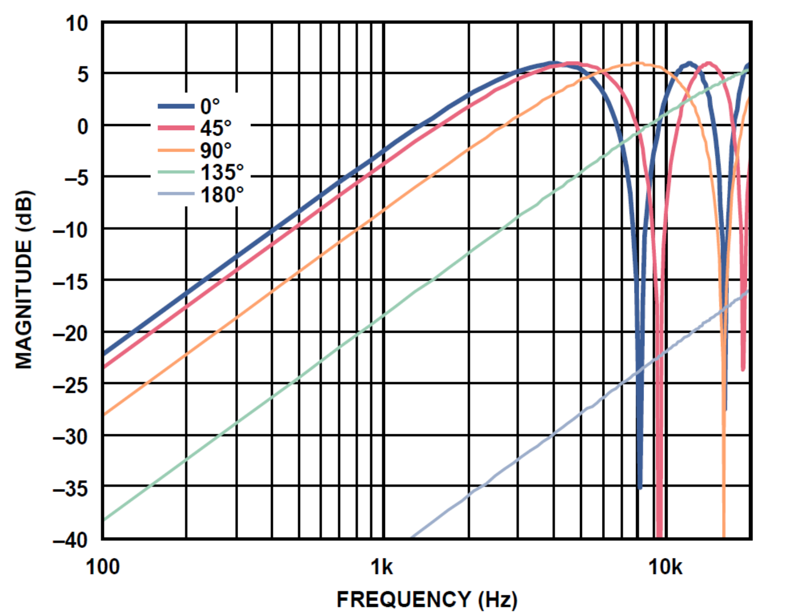


Figure 9. End-fire filters frequency response

4.3.1 Configuring Fractional Delay in the Algorithm

As described in the previous section, the required (fractional) delay is calculated according to the spacing between the microphones.

As calculated above, the RA6E1-VOICE-KIT with 50 mm spacing has:

n = 2.33 samples @ 50 mm (1st null at 3.4 kHz)

The delay is set as follows:

```
extern void FracDelaySetDelay(FracDelay_t *p, int intDelay, double fracDelay);
FracDelaySetDelay((FracDelay_t*)&bfNSuppData.beamForm.fracDelay, 2, 0.33);
```

Note: The library supports delay values up to n = 5. Higher values would typically not be used but can be made available upon request.

4.4 Broadside MIC spacing

A broadside configuration is most useful when the desired sound source is perpendicular to the microphones, or “broadside”.

A broadside beamformer provides an SNR boost to sounds approaching the microphones at 0 degrees while attenuating sounds parallel (from the sides) of the microphones. It’s important to note that sounds behind the microphones (180 degrees) will not be attenuated; hence, broadside configurations are more useful in applications where sound is not expected from the back of the microphone array – such as a voice control panel mounted on the wall.

Attenuation from a broadside beamformer is frequency dependent, where the maximum attenuation occurs when the sound arrives at +/- 90 degrees to the microphones.

Maximum attenuation occurs for one frequency in particular when the half wavelength of the incident sound matches the spacing between the microphones. However, all incident frequencies will receive the maximum attenuation possible at +/- 90 degrees – more so than any other location.

For example, with 75 mm spacing, maximum attenuation occurs at 2.3 kHz

(340 m/s) / (0.075mm*2) = 2.3 kHz (theoretically, attenuated 15 dB)

Every frequency is attenuated more at +/- 90 than elsewhere. However, low frequencies are not attenuated very much.

The most useful microphone spacing for rejecting voice and typical ambient noises is from 5 to 10 mm. Smaller distances reject very high frequencies, while larger distances are very low.

Table 4. Broadside Null Frequency for microphone distance

Distance (mm)	Null Frequency (kHz)
2.5	6.8
5	3.4
7.5	2.3
10	1.7
12.5	1.4
15	1.1

Please note that fractional delay is not used in a Broadside Beamforming configuration.

5. Application Examples

The following two application examples are offered to evaluate the APF functionality. It is recommended to contact the “Customers Success” team through rai-cs@renesas.com to receive the complete project bundle and follow the instructions in e2studio to import it and run it.

5.1 APF with Voice Commands Recognition

The first application example combines the functionality of Cyberon's DSpotter VCR enhanced with noise suppression from APF. Hence, it demonstrates the improvement in the quality of recognition accuracy when evaluated in noisy environments. The application example is offered as a project in e2studio that can be loaded directly into the voice kit. The on-board push button can be used to switch between the modes of Table 1.

Evaluation results in the audio lab have shown that the recognition rate gets superior enhancement as the noise level reaches the level of the voice. The results are expressed as "recognition-hits" / "number-of-repetitions"

Table 5. Improvement in recognition accuracy

	APF enabled (endfire)		APF enabled (broadside)		APF bypassed	
	female voice (~74dB-SPL)	male voice (~74dB-SPL)	female voice (~74dB-SPL)	male voice (~74dB-SPL)	female voice (~74dB-SPL)	male voice (~74dB-SPL)
68 dB-SPL	10 / 10	10 / 10	10 / 10	10 / 10	10 / 10	10 / 10
72 dB-SPL	5 / 10	7 / 10	6 / 10	8 / 10	0 / 10	0 / 10
74 dB-SPL	0 / 10	1 / 10	0 / 10	3 / 10	0 / 10	0 / 10

5.2 APF with USBx audio recording

The second application example is audio-related and combines APF with the USBx functionality that is offered in the Renesas FSP library for the RA family of devices. It demonstrates stereo audio recordings at 16KHz, while the processing of APF can be controlled per Table 1 configurations (see Appendix A for instructions to interface with a PC)

The Signal to Noise Ratio improvement (SNRi) for the results of has been estimated as the difference in the APF input and output signal when processing is enabled (see Fig.9).

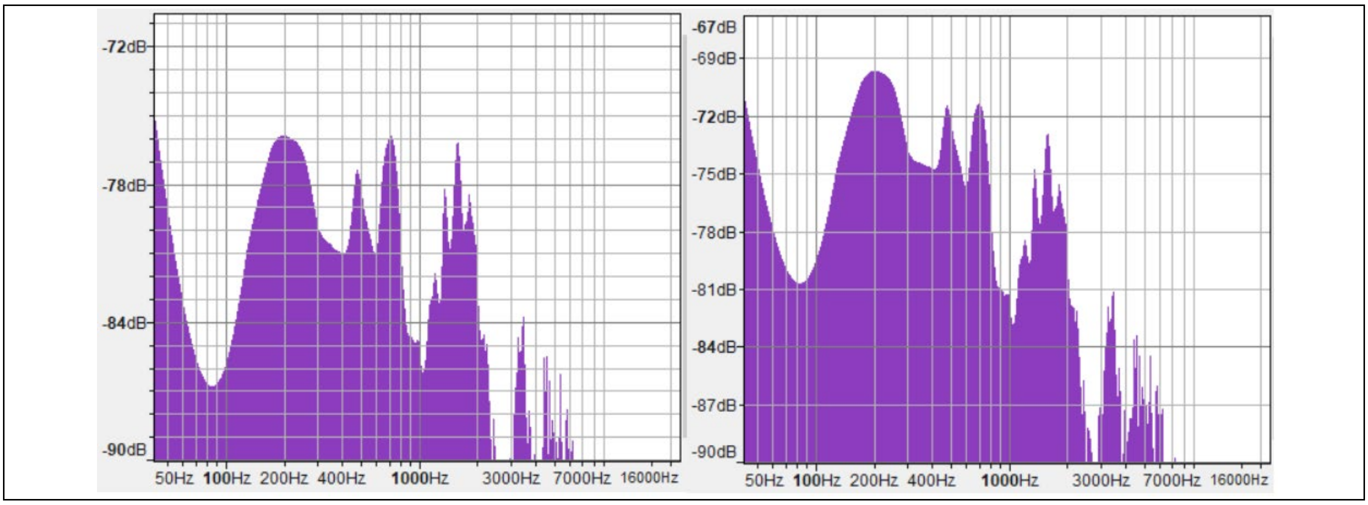


Figure 11. APF processing enabled (left), disabled (right)

Appendix A Windows PC setup for USB Audio

When using the USBx project, the example application expects the USB microphone device to be "open". This can be accomplished either by enabling the microphone through the Windows Sound "listen" feature or by opening an audio application of your choice; examples include Audacity, Teams, etc.

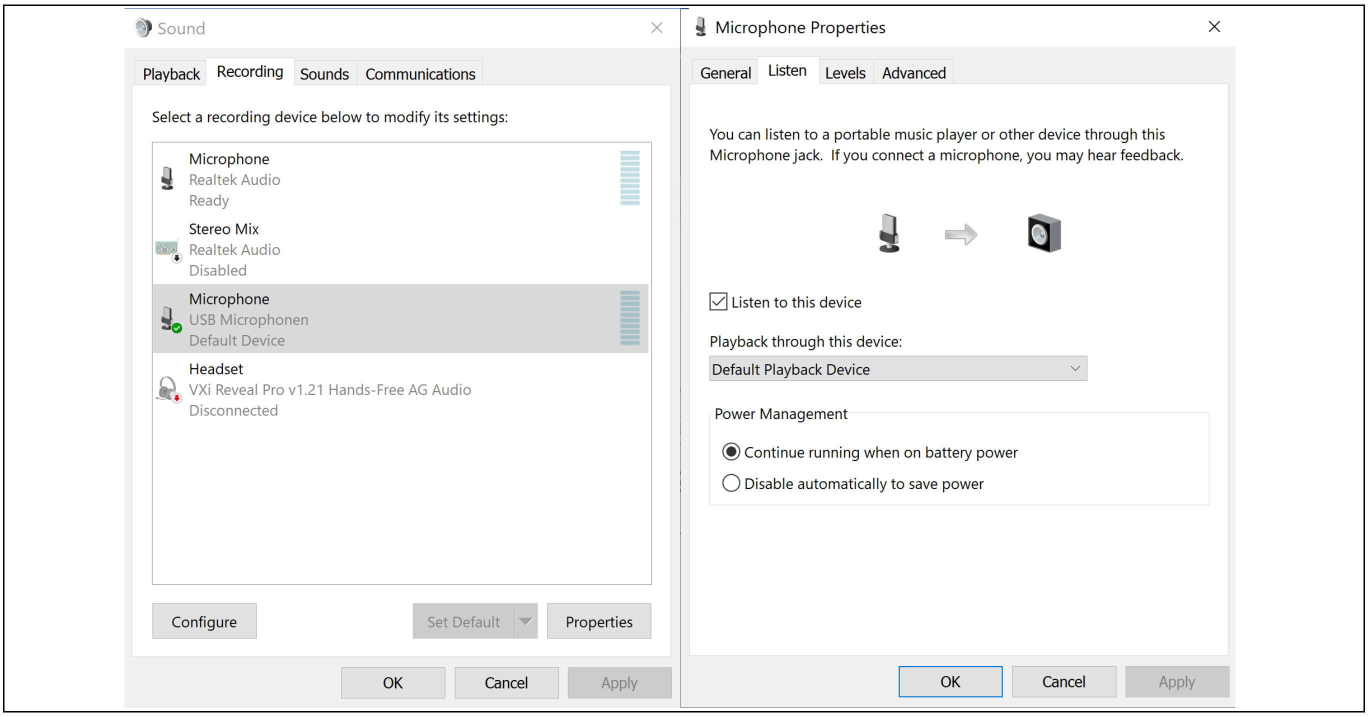


Figure 12. Windows configuration for USB audio recording

6. Terms and Definitions

APF	Audio Processing Frond
BF	BeamFormer
NSUPP	Noise Suppressor
VCR	Voice Commands Recognition

7. References

[1] R01DS0392EJ0120 RA6E1 Group Datasheet, v1.20, Renesas Electronics.

[2] R01UH0930EJ0120, RA6E1 Group Users Manual, v1.20, Renesas Electronics.

[3] VOICE-RA6E1 Engineering Manual, v1.0, Renesas Electronics.

Note: References are for the latest published version unless otherwise indicated.

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Nov.20.24	—	Initial release

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between VIL (Max.) and VIH (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between VIL (Max.) and VIH (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.