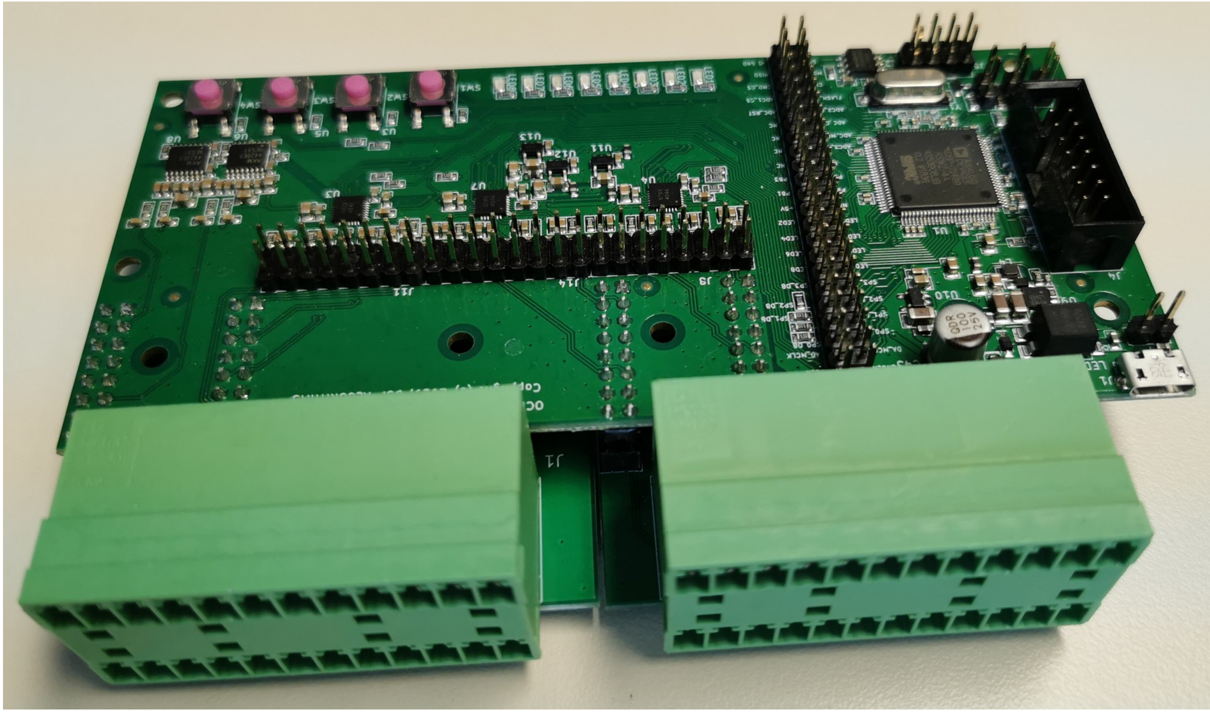

OCEAN-ADSP21489-1204HC Active Noise And Vibration Control

Document Version 1.0
November 9, 2023

TABLE OF CONTENTS

1 Introduction

This document describes the software features of the active noise and vibration control application on the OCEAN-ADSP21489-1204HC real-time processor. The hardware is based on the Analog Devices ADSP-21489 SHARC floating-point Digital Signal Processor (DSP). The software implements a multichannel feed-forward control system based on the active noise and vibration control theory. Both hardware and software are developed, maintained, and marketed by DSP Algorithms.

The OCEAN-ADSP21489-1204HC hardware includes all the necessary components that efficiently implement a complete stand-alone multi-channel digital control application. The hardware includes an Analog Devices ADSP-21489 SHARC processor, 12 analog input channels, 4 analog output channels, and simple user interface consisting in 4 buttons and 8 Light Emitting Diodes (LED). Full control (through for instance a web browser) is also possible using an external microcontroller which communicates with the SHARC DSP through SPI or UART. The hardware is described in a separate document and can be downloaded from https://www.dspalgorithms.com.

The application programs are stored in the on-board non-volatile flash memory. On powering the hardware, the processor copies the software from non-volatile memory to internal RAM and starts executing the program instructions. The real-time application reads the 12 analog audio inputs, processes the collected samples through the ANVC algorithm, and plays the control signals to the 4 analog outputs.

The analog inputs and outputs on the OCEAN-ADSP21489-1204HC are provided through industry standard 0.1" headers. Those headers can be used to offer the inputs and outputs on a variety of standard connectors, such as XLR, RCA, jacks, or terminal blocks, as needed by the end user requirements to interface to the transducers at hand.

The software supports both active noise control and active vibration control without any modification. The only difference is in the type of transducers connected to the hardware. For active noise control microphones are used as sensors and loudspeakers are used as actuators. In an active vibration control system, accelerometers (or similar devices) are used as sensors and shakers are used as actuators.

2 Active Noise & Vibration Control Theory

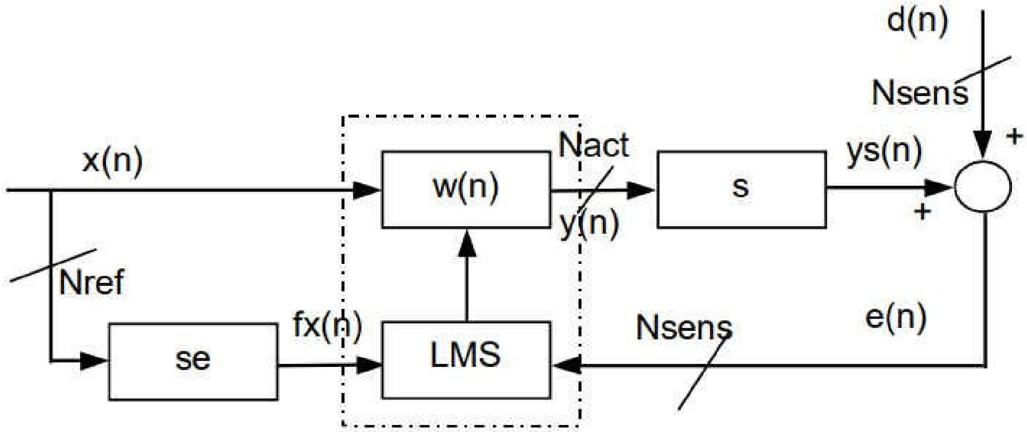

Figure 1: Block diagram of the active noise and vibration control system

Active noise control is the art of fighting noise with opposite phase noise. When the opposite phase noise is combined with the original (primary) noise, the result is almost silence. The problem to solve to realize such a system is to calculate that opposite phase noise and generate it using a loudspeaker. The same theory applies to vibration control by measuring the undesired vibrations and generating opposite phase vibrations at the measurement points. Active noise control systems therefore use loudspeakers to generate the opposite phase noise and microphones to measure the undesired noise. Active vibration control systems use shakers to generate the opposite phase vibrations and accelerometers to measure the unwanted vibrations. In general, the opposite phase signal is played to an actuator of one sort or another, and sensors are used to measure the unwanted signals to be canceled or reduced.

ANVC systems use adaptive digital signal processing to reduce noise at certain places in an enclosure (car cabin, airplane fuselage, room, etc), or in some cases in the whole space of interest. The block diagram of a multichannel ANVC system is shown in Figure 1. *Nref* sensors are used to pickup multiple clean reference signals of the original (primary) noise **x(n)**. Another *Nsens* error sensors are placed at several points where noise/vibration need to be minimized. The signals measured by the Nsens error sensors **e(n)** are used by the adaptive control system (labeled LMS in Figure 1) to adjust a set of adaptive filters **W(n)** in such a way that the generated *Nact* control signals **y(n)** minimize the output of the error sensors **e(n)**.

Each error sensor output $\mathbf{e_i(n)}$ is the sum of the primary noise **d(n)** (desired to be canceled or reduced) at that sensor and the contribution of all *Nact* control signals at the position of that sensor. The contribution of the control signals at the error sensor is the result of filtering the *Nact* control signals **y(n)** through the acoustic/structural transfer function **S** between the control actuators and the error sensor. Because of that filtering through the transfer function **S**, the adaptive LMS algorithms observes the contribution of the control signals delayed and distorted. To compensate for this effect, the LMS algorithm used to update the matrix of control filters **W(n)** needs to use a filtered version of the reference signals, depicted as **fx(n)** in Figure 1.

Ideally, **fx(n)** should be obtained by filtering the reference signals **x(n)** through the transfer function **S**. Therefore, to obtain **fx(n)** the *Nsens* x *Nact* matrix of transfer functions between each actuator and each error sensor must be measured before the real-time control operation can be performed. This measured transfer function is depicted as **se** in Figure 1.

From the above, it may be clear that an ANVC system needs to be implemented in two phases. The first phase being the identification phase, where the *Nsens* x *Nact* matrix of transfer functions **S** between each actuator and each error sensor is measured and stored to be used in updating the adaptive control filters **W(n)**. The second phase is the real-time control phase during which the system generates the opposite phase signals with the purpose of minimizing the error sensors output.

3 Active Noise & Vibration Control Firmware

The ANVC firmware is installed on the OCEAN-ADSP21489-1204HC flash memory and the application boots by default as soon as the hardware is powered on. This application requires the OCEAN-ADSP21489-1204HC main PCB only. The user is responsible for connecting the right sensors and actuators to the PCB. For active noise control applications, one of the standard OCEAN microphone array PCB, OCEAN-04-MIC-LEFT/RIGHT for instance, can also be used. The microphones PCB plug directly into the OCEAN-ADSP21489-1204HC Analog Input Header where the microphones receive proper low noise biasing.

The default software application expects up to 2 reference signals, up to 2 error sensors, and generates up to 2 opposite phase signals to be sent to two different actuators. It should be mentioned however that not all signals are necessary. The application will also work properly if only one reference, one sensor, and one actuator are connected and the remaining inputs and outputs left floating.

The two reference signals must be connected to the OCEAN-ADSP21489-1204HC at input IN1 and IN2. The error signals collected by the error sensors must be connected to the OCEAN-ADSP21489-1204HC at inputs IN3 and IN4. The following table and Figure 2 summarize the input connections for ANVC application.

Input	IN1	IN2	IN3	IN4	IN5	IN6	IN7	IN8	IN9	IN10	IN11	IN12
signal	REF1	REF2	SEN1	SEN2	NC	NC	NC	NC	NC	NC	NC	NC

At the output side, the generated opposite phase control signals are sent to OUT1 and OUT2 respectively. Those two outputs should be connected to two powered loudspeakers (noise control) or two powered shakers (vibration control). The individual error signals after applying the control algorithm are sent to OUT3 and OUT4 and can be used to evaluate the achieved noise/vibration reduction. The following table and Figure 2 summarize the output connections for the ANVC application.

Output	OUT1	OUT2	OUT3	OUT4
signal	ACTU1	ACTU2	SEN1	SEN2

Once the hardware is powered on, the code is copied from the flash memory to the SHARC DSP RAM and the application starts running. At start-up, the application enters STAND-BY mode (see Section 4). The user must press `SW1` to start the identification phase. In this phase, the application sends white noise to each of the actuators in turn and measures the transfer function between each error sensor and each actuator. Those transfer functions are needed for the adaptive control algorithm.

Once the identification process is complete (which may take a few seconds, depends on the number and position of sensors and actuators), the real-time control process starts. During this process, the software reads the reference and error signals, updates the adaptive controllers, and outputs the calculated opposite phase signals to OUT1 and OUT2. When all signals are connected properly and sensors and actuators are positioned properly, the error sensors at OUT3 and OUT should decrease in amplitude compared to their initial levels.

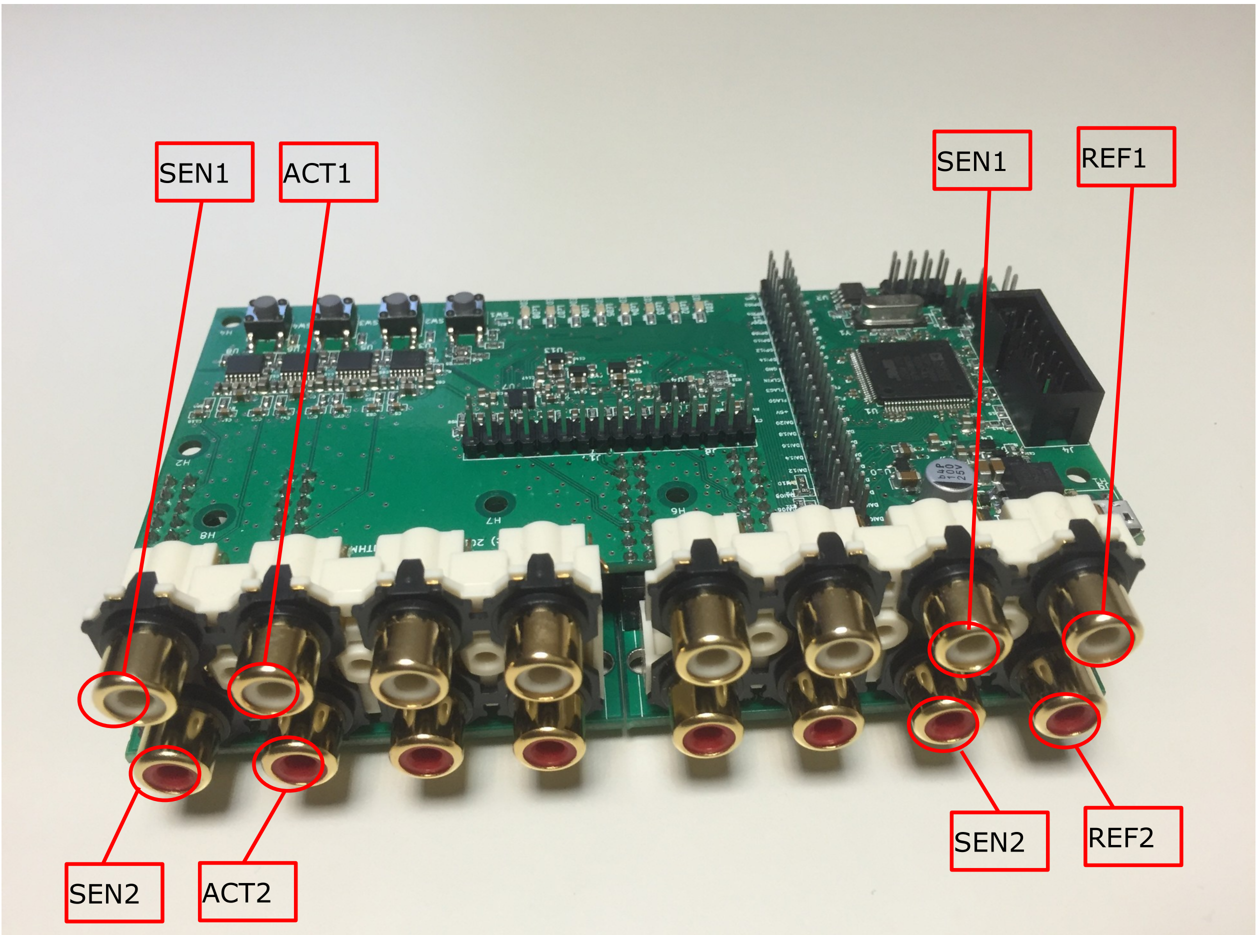

Figure 2: Signals inputs and outputs for the ANVC application.

4 ANVC Applications User Interface

The user interface of the ANVC applications on the OCEAN-ADSP21489-1204HC makes use of the on-board LEDs and push-buttons. It is important to mention that the user interface described below is designed to allow a fully-functioning stand-alone application. An OEM application based on this OCEAN-ADSP21489-1204HC may or may not need user interface. This section describes only user interface implemented using the OCEAN-ADSP21489-1204HC LEDs and buttons.

The ANVC application uses the four push-buttons and eight LEDs to implement a few capabilities of the Application Programming Interface (API) of the underlying ANVC library. The function of each button and LED are described below.

`SW1` button is used to start and stop the real-time processing. The application always starts in STAND-BY mode. In this mode the software just blinks `LED1` and `LED2` simultaneously at the rate of 0.25 Hz, no real-time processing is performed. Pressing `SW1` while in STAND-BY will reset the ANVC algorithm and start the real-time processing. During real-time operation, `LED1` and `LED2` switch ON and OFF alternatively at the rate of 1 Hz.

`SW2` and `SW3` button are not used in this application.

`SW4` button is used to Enable or Disable the ANVC algorithm on all sensors and actuators. Pressing this button will toggle the state of the ANVC and `LED3`. The application starts with the ANVC enabled by default. If all signal connections and levels are correct, you should observe the output of the error sensors decrease as a result of the control effort. When ANVC is enabled `LED3` will turn ON. Pushing `SW4` button when `LED3` is ON will disable ANVC and `LED3` will turn OFF and you should observe the error sensors output level increase immediately to their original level before any control effort. Pushing `SW4` button while `LED3` is OFF enables the ANVC, `LED3` will turn ON, and you should observe the effect of the control algorithm.

The applications also checks the level of all input signals before processing every sample. If any of the input signals exceeds the analog to digital converter maximum analog input level, a visual indication is given by temporarily turning ON one of the LEDs. `LED7` is used to indicate that at least one of the error sensors inputs is too high. `LED8` is used to indicate that at least one of the reference inputs is too high.

The table below summarizes the ANVC application user interface.

push-button	Function	Visual Indicators
`SW1`	Toggles between STAND-BY and RUN modes	`LED1,LED2`
`SW2`	NOT used	
`SW3`	NOT used	
`SW4`	Toggles ANVC ON or OFF	`LED3`
	At least one error sensor signal is too high	`LED7`
	At least one reference signal is too high	`LED8`

Remarks:

- The ANVC algorithm will not work properly if either the reference inputs or error sensors levels are too high. This is because the signals are then saturated and clipped when converted to digital samples causing severe non-linearity. Reduce the levels until both `LED7` and `LED8` turn OFF completely, reset the application by power cycling the hardware, and try again.

- It is essential to perform the identification step while all sensors and actuators are fixed in their place. Should one of the sensors or actuators move during real-time processing phase, the identification step must be repeated (by power cycling the hardware).

- For best performance, the control actuators should be placed as close as possible to the error sensors.

- To achieve any noise/vibration reduction, the ANVC filters must be causal (the system can generate the opposite phase control signal on-time to be combined with the primary noise/vibration). This is achieved only if the total processing delay (including analog to digital and digital to analog conversion) is shorter than the noise/vibration propagation delay from the source to the error sensors positions. Therefore, it is essential to carefully choose the positions of the error sensors, control actuators, and reference sensors to achieve this causality constraint.

5 Expected Results

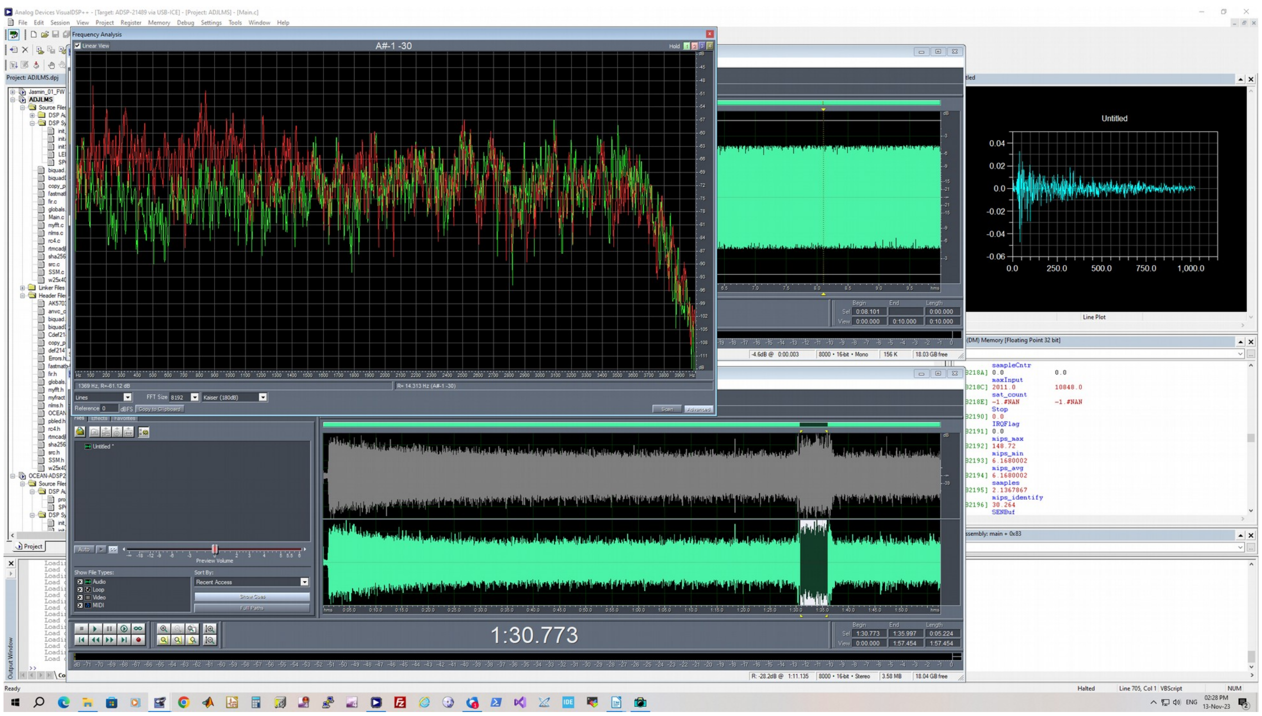

Figure 3: Noise reduction of white noise with 2 reference signals, two error sensors and two actuators.

Figure 3 shows the result of an experiment performed to test the ANVC firmware installed on the OCEAN-ADSP21489-1204HC hardware. As the firmware runs at sampling rate 8 kHz, a white noise sampled at 8 kHz was played from a PC sound card to two large powered loudspeakers capable of reproducing low frequencies without distortion. The primary signals from the PC were connected to OCEAN hardware IN1 and IN2. The primary loudspeakers playing the white noise from the PC sound card were placed about 3 meters from the two error microphones. The error microphones were connected to IN3 and IN4 of the OCEAN hardware. Two large powered loudspeakers were connected to the OCEAN hardware OUT1 and OUT2 as control actuators. The two control actuators were placed 25 cm behind the two error microphones. It is essential to place the primary sources far from the error sensors so that the causality constraint is maintained. Figure 4 depicts the experiment setup.

After powering the OCEAN hardware, loudspeakers, and microphones, SW1 on the OCEAN hardware must be pressed to start the identification phase. During this phase the firmware sends white noise to each of the control actuators to estimate the secondary transfer functions between each actuator and each error microphone. Once the identification phase is done, the firmware switches directly to the noise cancellation phase. As the noise cancellation phase started, the white noise was played to the primary loudspeakers from the PC sound card, and the two error sensor signals at OUT3 and OUT4 were recorded on the PC to visually observe the noise reduction. The time progress of the error sensors is shown at the bottom of Error: Reference source not found which clearly shows the amplitude of the noise at both sensors decreasing with time, as the control algorithm converges to a better solution. The frequency components of one of the error microphones is also shown when the algorithm is switched off (red color) and on (green color) of the recording. This shows clearly that the ANVC algorithm could reduce the noise significantly at low frequencies. It can be also observed that the noise reduction decreases as the frequency increases. This is a known limitation of the ANVC principle due to the fact that the audio wave length decreases as the frequency increases which makes ANC useful only below around 1kHz.

The adaptive controller coefficients are also shown in the top right corner of Figure 3. The shape of the filter coefficients and the amount of noise reduction achieved depend on the primary and secondary acoustic transfer functions (P and S in Figure 4).

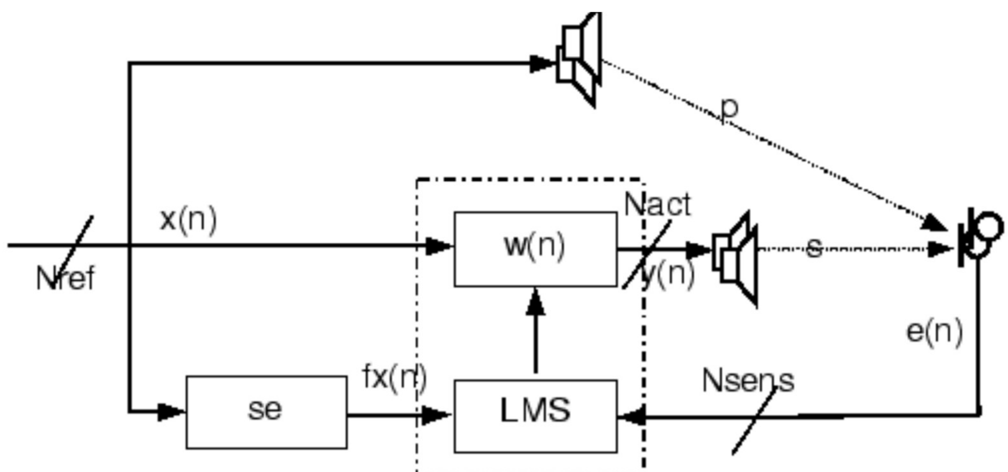

Figure 4: Active Noise Control using 2 primary sources, two error sensors, and two secondary sources.

6 Firmware Update

To program a new binary file onto the OCEAN-ADSP21489 board, do the following

1. Connect an emulator to the OCEAN-ADSP21489 JTAG header.

2. Connect the USB side of the emulator to a PC running VisualDSP++ version 5.1.2 (latest version)

3. Start VisualDSP++ on the PC and connect to the target (ADSP-21489 via USB-ICE or HP-USB-ICE, etc).

4. Start the flash programmer utility (select **Tools -> Flash Programmer ...**)

5. On the Driver tab of the flash programmer plugin, select the SERIAL FLASH Driver File from your installation directory, it should be some thing like the following
C:\Program Files (x86)\Analog Devices\VisualDSP 5.1.2\214xx\Examples\ADSP-21489 EZ-Board\Flash Programmer\Serial\21489EzFlashDriver_Serial.dxe
Then press "Load Driver" button.

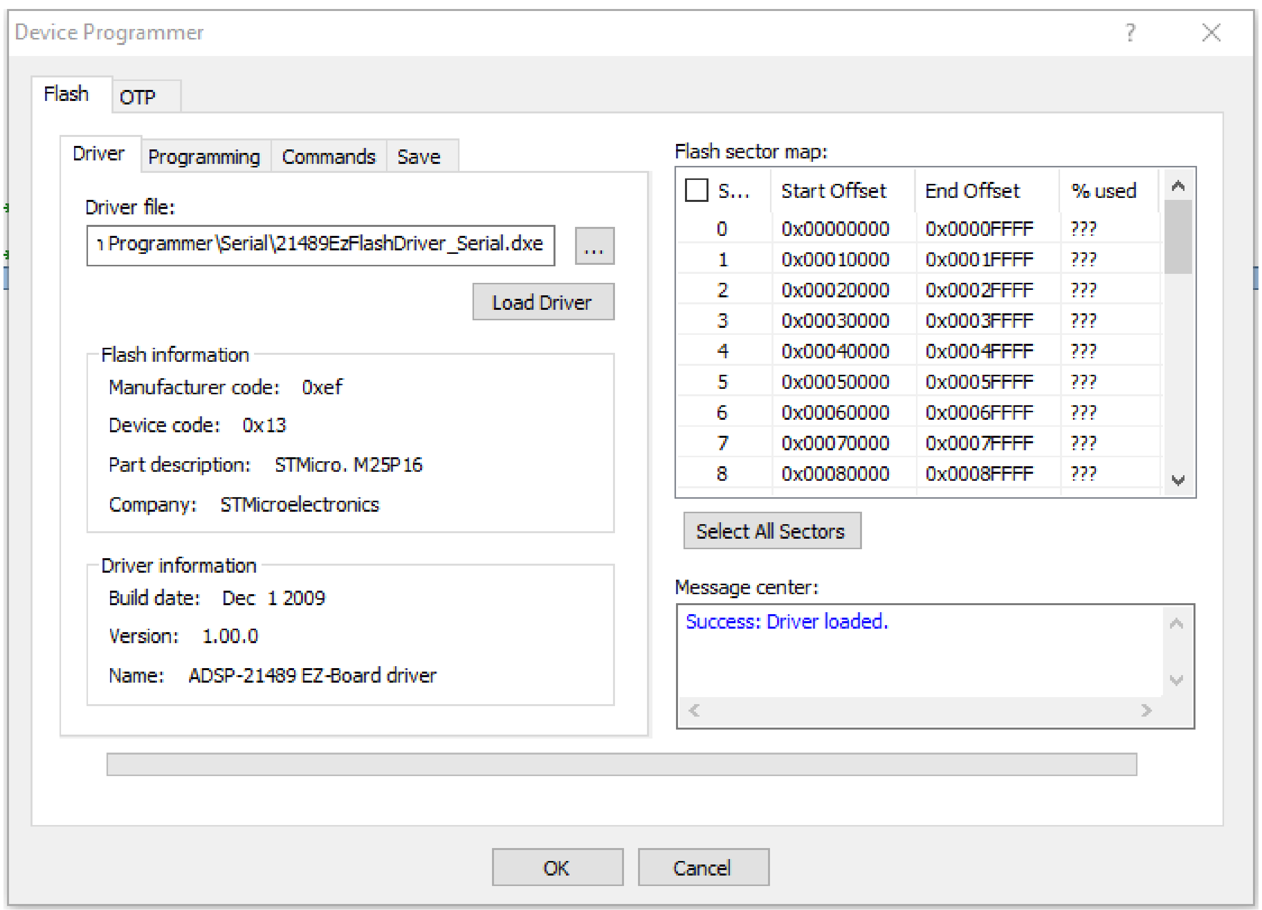

Figure 5: Flash Programmer Driver tab

6. After the driver is loaded, go to the Programming tab and make sure the settings are EXACTLY as in the Flash Programmer Programming tab (Figure 6).

Double-check that
- Programming erase options : *Erase affected SELECTED*
- File Format : *Binary SELECTED*
- Offset : *0x00000*
- Data File : *select the anvc.ldr file that you received for this update.*
- Check Verify while programming

Then press the "Program" button. This will write anvc.ldr to address 0x0000 0000 on the flash memory chip.

7. Close the flash programmer plugin, close VisualDSP++, disconnect the emulator, and power cycle the OCEAN-ADSP21489 board. The newly programmed application should boot.

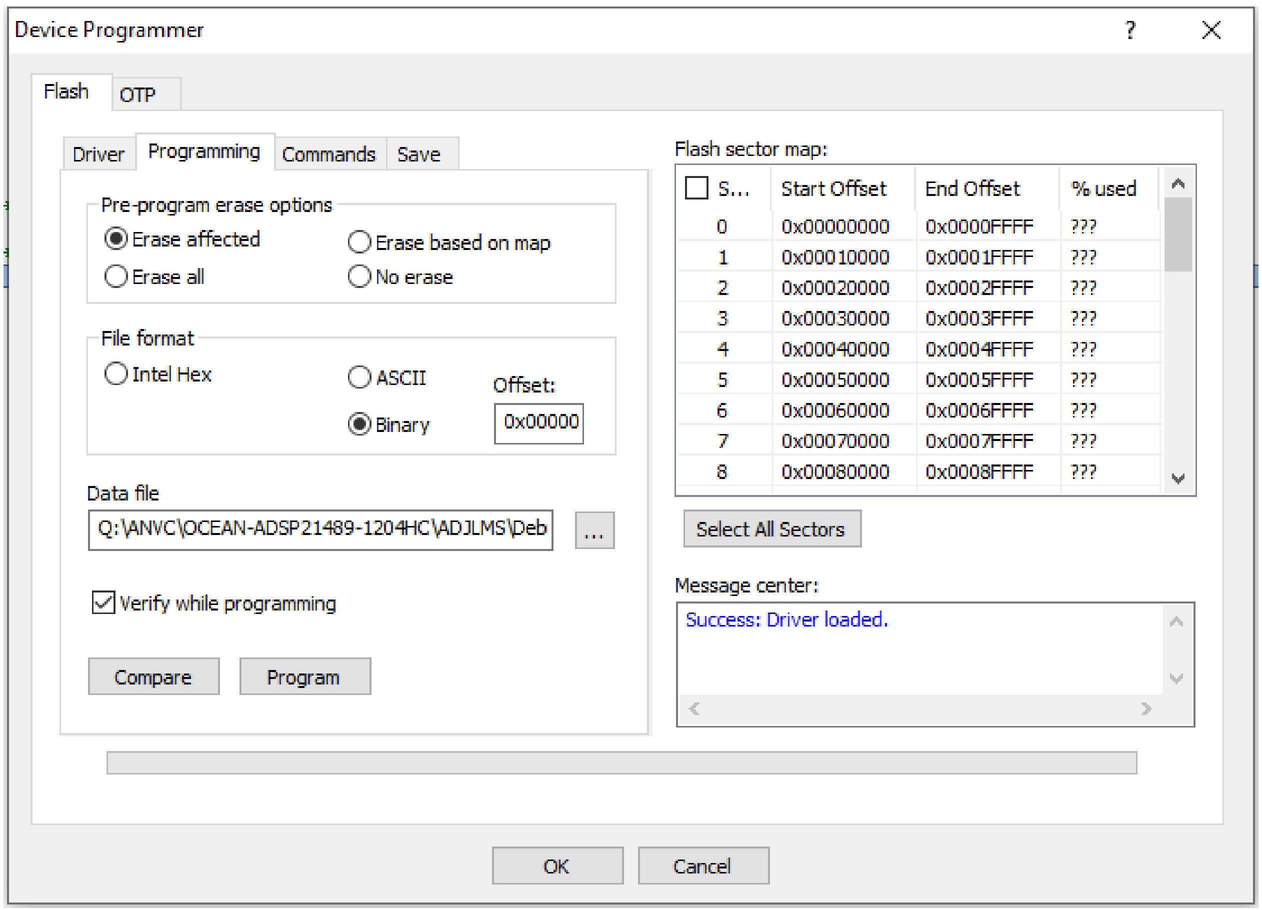

Figure 6: Flash Programmer Programming tab