TN120

Audio Precision

☐ 2700 Series
☑ APx555
☑ APx585 Series
☑ APx525 Series
☑ APx515

SMARTPHONE AUDIO TEST APP 2.1
FOR ANDROID™

by Steve Peterson

Overview

The connections for testing a simple analog audio device are straightforward: connect a generator to the inputs and an analyzer to the outputs.

But a smartphone or tablet is quite different. Audio inputs include acoustic (microphones), analog (mic jack), *Bluetooth™* hands-free microphone, and speech received from the mobile network. Audio outputs include speaker, earpiece, headphone jack, *Bluetooth* hands-free headset or speaker phone, *Bluetooth* speaker for music, and speech transmitted to the mobile network. *Bluetooth* audio hands-free testing is particularly difficult because it usually requires a phone call in progress through the wireless network, so isolating the *Bluetooth* link from the rest of the signal chain is impossible.

What's more, the device's operating system (OS) often routes audio through a "media player" that processes the audio and may mask the true performance of the audio hardware (such as level and tone controls). These features can be difficult or impossible to defeat.

We've created a software app, called "AP Smartphone Audio Test", that runs on a smartphone or tablet with Android versions 4, 5, and 6. It routes audio signals within the smartphone for playback and record and enables testing *Bluetooth* hands-free voice quality without a phone call through the wireless network.

For input signal path testing, use the APx instrument to provide audio test signals to the device inputs while the Smartphone app records linear WAV files. The files are then downloaded to the PC for analysis by the APx software.

Android is a trademark of Google, Inc.
The *Bluetooth* word mark and logos are registered trademarks owned by Bluetooth SIG, Inc., and any use of such marks by Audio Precision is under license. Other trademarks and trade names are those of their respective owners.

For output signal path testing, play audio WAV files provided. The Smartphone app plays the test signals through the device's outputs for real-time analysis by the APx instrument.
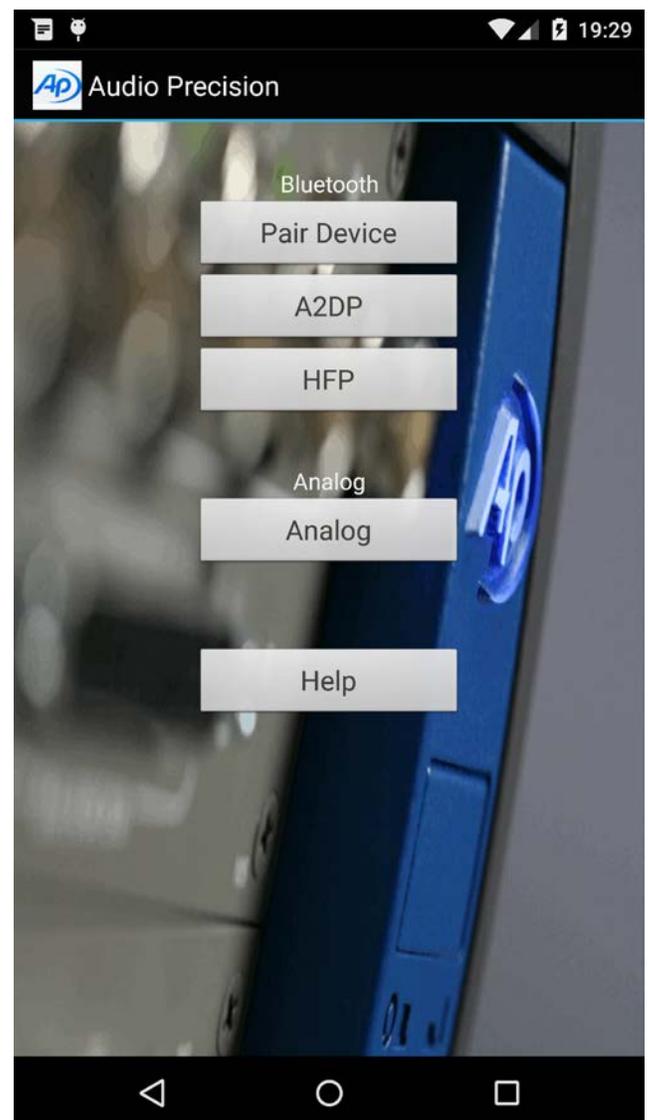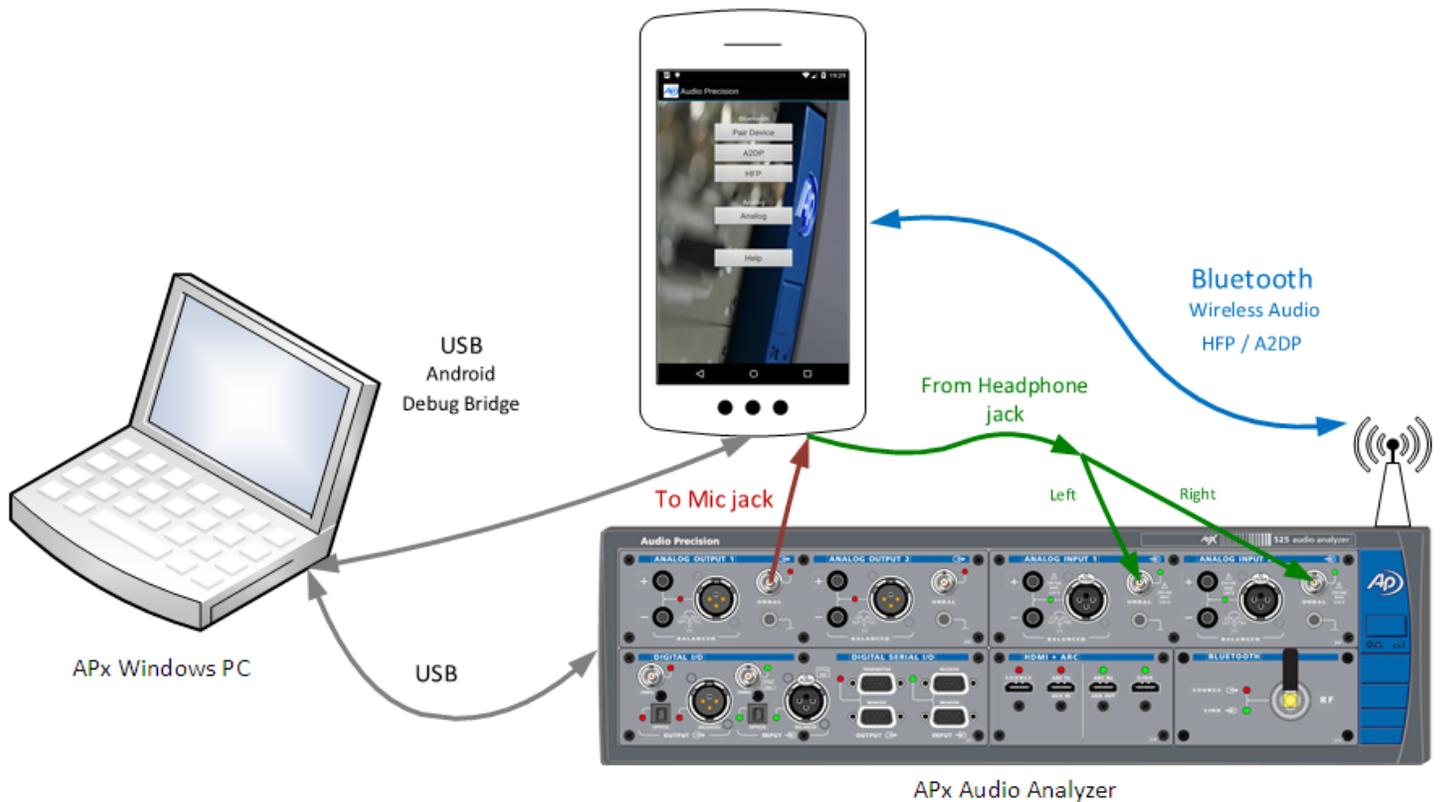
Figure 1. Android smartphone test system with the Smartphone Audio Test app and APx525 audio analyzer with Bluetooth audio interface.

We've created a series of APx500 measurement projects that test the analog and *Bluetooth* audio input and output paths, and audio test files for playback on the device.

This technote will describe how to install the Technote software on the PC and the smartphone, smartphone audio signal paths, how to use the app on the smartphone, overview the APx projects provided for smartphone testing, and present typical smartphone audio test results for an Android 6 smartphone.

What you need

You will need an Audio Precision APx analyzer. If you want to test the DUT's *Bluetooth* I/O, your analyzer must have the Audio Precision *Bluetooth* I/O option installed.

Note that although the APx515 analyzer can perform analog testing on smartphone and tablet devices, it cannot be fitted with the *Bluetooth* option module in order to test Bluetooth HFP and A2DP audio performance.

You'll need a Windows PC running Audio Precision's APx500 software version 4.2.1 or later.

Depending upon your requirements, for acoustic testing you may need a measurement microphone, a loudspeaker and amplifier, an acoustic chamber, or a testing manikin such as a head and torso simulator with an ear or mouth simulator.

You'll also need the Audio Precision software package that includes the Smartphone Audio Test App and support files. This package can be downloaded from the Audio Precision Web site, discussed next. You will need to obtain separately the Software Developer's Kit (SDK) for Android and the USB driver for your smartphone device.

Installing the Smartphone Audio Test app package on the PC

Go to the Audio Precision Web site at http://ap.com and navigate to **TN120 Smartphone Audio Test App 2.1**. You will see download links for

- TN120 Smartphone Audio Test App 2.1.pdf (this document)
- TN120 Smartphone Audio Test App Package 2.1.zip

Download the TN120 Smartphone Audio Test App Package.zip file and extract the files in the archive. You will find these files:

- SmartphoneAppSetup2.1.exe
- TN120 Smartphone Audio App.pdf (this document)
- ReadMe.txt

Run **SmartphoneAppv2.1Setup.exe**. This program will install APx project files and audio wav files on your PC at a folder location you may specify. Install only to folders that have full permissions enabled. It will also install the application "Smartphone App Device Installer2.1.exe".

Note: During installation on the PC, Smartphone-AppSetup.exe inspects all APx project files (*.approjx) found in the install folder and its subfolders, and will make changes to filename variables in the project files. We recommend that only project files related to Smartphone Audio Test are saved in the install folder and its subfolders.

Installing a USB device driver

Each smartphone or tablet device requires specific device driver software to be installed on the PC for USB communications with the device. Device drivers can be found at

http://developer.android.com/sdk/win-usb.html.

or

http://developer.android.com/tools/extras/oem-usb.html

or from the Web site of the device manufacturer. Follow the instructions provided with the driver to install it on your PC.

Installing the SDK for Android

Three component files of the Software Developers Kit (SDK) for Android are required for the Smartphone Audio Test App to function properly. These files comprise the Android Debug Bridge:

- adb.exe
- AdbWinApi.dll
- AdbWinUsbApi.dll

The Android Debug Bridge (adb) is a command line tool that lets APX communicate with your connected Android device.

If you already have these files installed on your PC from a previous SDK installation, you can skip this step. If not, go to the Android Developer web site to obtain the stand-alone SDK tools:

http://developer.android.com/sdk/installing/index.html?pkg=tools

Follow the instructions to install the SDK on your PC. This is a large package and takes some time to install. The three files will be found in the <sdk>/platform-tools/ folder.

For more information about the Android Debug Bridge go to: http://developer.android.com/tools/help/adb.html

Setting the Windows Environment Variable for adb.exe

The **SmartPhoneAppInstaller.exe** software that was installed as part of the AP Smartphone Audio Test App package must know where to find the file "adb.exe". It achieves this by using a Windows environment variable that points to the file location.
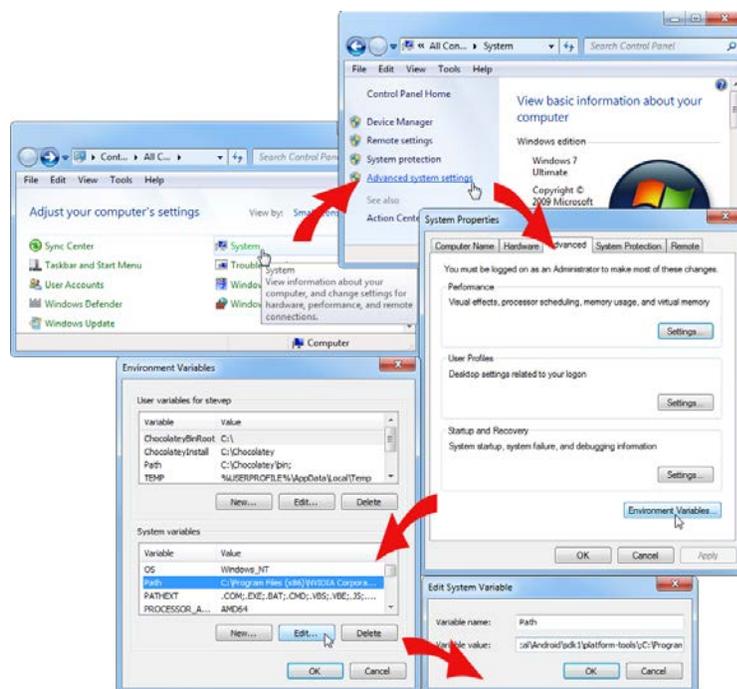

Figure 2. Setting the Windows Environment Variable

In Microsoft Windows, go to **Computer > System Properties > Advanced System Settings** and select the **Advanced** tab. Click the **Environment Variables** button. In the Environment Variables dialog, scroll through the variables in the System variables list until you find the Path variable. Select the **Path** variable and click **Edit...**

The **Path** variable is typically a very long string. Place the cursor at the right end of the string and type a semicolon, as shown here within parentheses (;). Do not type the parentheses into the Path variable. After the semicolon, enter the path to the folder where the SDK for Android installer has placed the file **adb.exe**. This path will depend upon the destination folder chosen when the SDK archive was extracted. A typical path might be

C:\Users*user*\AppData\Local\Android\sdk\platform-tools\

Be sure to end the path string with a final backslash.

Preparing the Android device

Before you can install the Smartphone Audio Test app and components on your device, you must first prepare by making several Android settings on the device itself.

Enabling USB debugging and Unknown sources

Go to **Settings > Developer options > USB debugging** and set the checkbox to turn on *Debug mode when USB is connected*.

> In some devices, the manufacturer has hidden "Developer options." In that case, go to **Settings > About phone** and touch **Build number** seven times in quick succession to reveal Developer options.

Then proceed to **Settings > Security > Unknown sources** and set the checkbox to *Allow installation of apps from sources other than the Play Store*.

Language & Input

Go to Settings > Language & input and select English (United States).

When using the on-screen Google keyboard, select the English (US) keyboard.

> When entering recording filenames, enter only alpha-numeric and punctuation characters. Do not enter special characters or emoticons.

Uploading files to the device

Connect an APx analyzer to the PC and turn it on. APx500 (version 4.2 or later) must already be installed and run at least once to instantiate the APX USB driver.

Verify that you have installed the USB device driver for your Android device on the PC. Connect the smartphone or tablet device to the PC using a USB cable.

On the PC, connect your APX audio analyzer via USB and turn it on, then open Audio Precision's **SmartPhoneAppInstaller.exe**. A shortcut has been provided on the Windows Start Menu under **Audio Precision > APx500 Utilities > Smartphone App 2.1**.

> Note: Installing this version of the Smartphone app on your Android device will uninstall the former version if it was installed on the device. No previously recorded audio files will be affected.
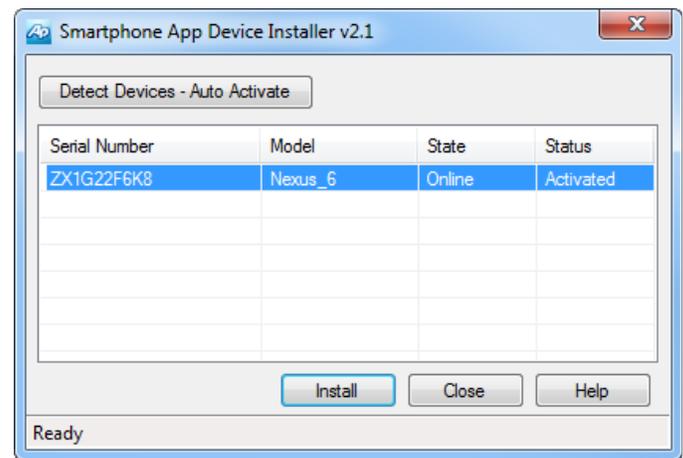

Figure 3. Smartphone app device installer

In the Smartphone App Device Installer window that opens, click **Detect Devices – Auto Activate**. All USB-connected devices running under the Android OS will be listed. Choose the device under test and click **Install**.

If this is the first time you have uploaded to this device via USB from this PC, the device may display a message requiring the authorization of USB debugging for the specific connected PC. On the device, you may have to authorize the PC. Then **Close** the Smartphone App Device Installer on the PC and restart it.

Select the device and click Install. The Smartphone App Device Installer first uploads and installs **OI File Manager** version 2.0.5 on the Android device. On the device screen, you will be asked to accept the End User License Agreement for OI File Manager. Accept the agreement before continuing.

Next, the installer uploads and installs the AP Smartphone Audio Test app, starts the app, and displays the End User License Agreement screen. Accept the agreement with the OK button. You may cancel the agreement and accept later.

The installer automatically activates the app and waits for you to select OK. Activation is only required one time per Android device.

> Note: Activation requires an active connection to your APX analyzer. APx software must be installed and the analyzer must be connected and turned on.

The installer creates a folder and subfolders on the device for use by the Smartphone App, and copies audio test files into these folders. The path on the device is: /sdcard/AudioPrecision/…
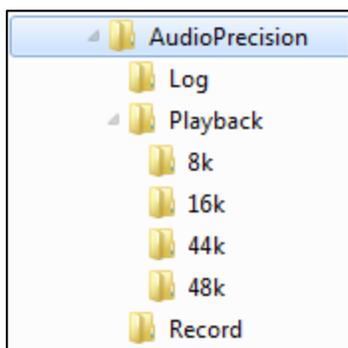

Figure 4. Folders created on the device.

The Playback folders are populated with audio test files in the WAV format, separated into folders by sample rate. These playback files should not be deleted.

The Record folder contains empty files for recording. These are very short (44 byte) placeholder files. These empty files are the same file names used by the APx projects provided with this Technote. This simplifies file name entry on the Android device if you decide to perform audio tests with the manual APx projects. These record files may be deleted if desired.

Uninstalling software from the Android device

The software apps and audio files installed on the Android device may be uninstalled manually.

Uninstall Smartphone Audio Test

The Smartphone Audio Test app may be uninstalled from the device with the Apps selection in the Android Settings menu.

- Select the DOWNLOADED column.
- Select the AP icon.
- Select the UNINSTALL button.
- Select OK when the confirmation window appears.

Uninstall Audio Files

The /sdcard/AudioPrecision/ folder may be deleted manually with OI File Manager. This will delete all recorded files as well.

- Start OI File Manager and browse to /sdcard.
- Hold your finger on the AudioPrecision folder icon until the top line of OI File Manager window displays the tool bar with the trash can icon.
- Select the trash can icon.
- The confirmation window will appear with the message "Do you really want to delete AudioPrecision?" Select OK in the confirmation window.

Uninstall OI File Manager

OI File Manager may be uninstalled from the device with the Apps selection in the Android Settings menu.

- Select the DOWNLOADED column.
- Select the OI File Manager icon.
- Select the UNINSTALL button.
- Select OK when the confirmation window appears.

Using the Smartphone Audio Test app

Launch the app on your Android device by touching the AP icon in Apps.

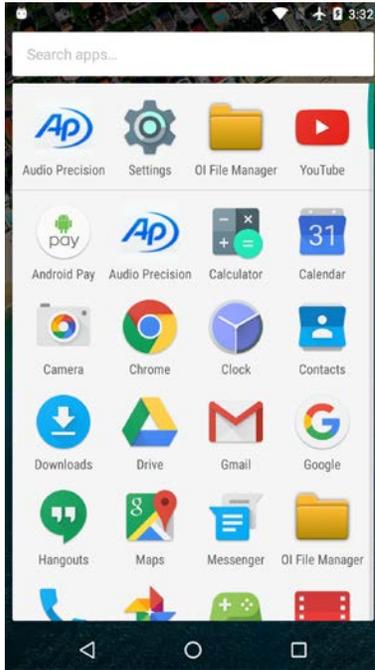OI File Manager is also installed. Use it to manage waveform files recorded during audio tests.

Figure 5. AP Smartphone Audio Test and OI File Manager icons in the Android Apps menu.

Smartphone Audio Signal Paths

The Smartphone Audio Test app supports analog (electric and acoustic) and *Bluetooth* input and output signal paths. Analog signal paths depend upon hardware connections (jack insertion) and app software settings.

Testing Device Outputs with Playback

The Smartphone Audio Test app plays audio test files from the device through the selected output while the APx500 instrument measures the audio in real time.

Testing Device Inputs with Record

For analog, acoustic, and *Bluetooth* device inputs, the Smartphone Audio Test App records the audio to a file on the device while the APx analyzer outputs analog or *Bluetooth* streaming audio. The APx project files included in the AP Smartphone Audio Test App package use adb.exe to copy the recorded audio files from the Android file system to the PC for file analysis. Optionally, you can browse the connected device from the PC with a USB connection (if an MTP connection is enabled on the device) and manually copy the recorded audio files to the PC for subsequent analysis.

Analog Audio Signal Paths

These are the analog signal paths available with the Smartphone Audio Test app. An asterisk (*) in the caption indicates the acoustic domain.

Definition of Terms

- **Stereo Headphone Output Jack**, the 3.5 mm "mini" audio jack that provides mono or stereo analog audio output to headphones, earbuds or as a line output for amplification. Note that this same jack provides the electrical microphone input when presented with the proper connector.

- **Loudspeaker Output**, the small, built-in loudspeaker(s) for audio playback, typically on the rear or sides of the device.

- **Ear Output**, a small, built-in loudspeaker for telephone communication, located near the ear when the device is held in the typical orientation for a telephone call.

- **Acoustic Microphone Input**, the built-in microphone(s) for telephone communication.

- **Electrical Microphone Input**, a contact available within the 3.5 mm audio jack for connection to an external microphone, typically part of a communication headset. See Stereo Headphone Output Jack, above.

These signal paths may be configured with combinations of settings and headphone plugs described in Figure 44 and Figure 45.
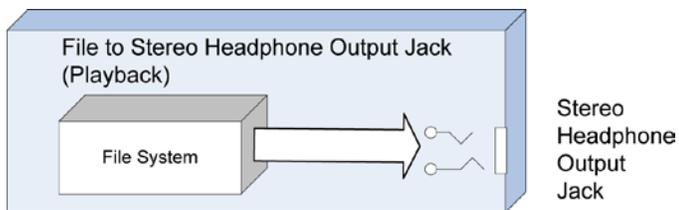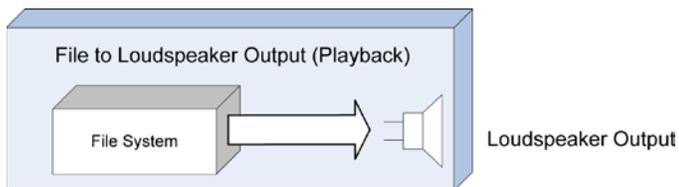
Figure 6. Playback: File > Headphone Output Jack
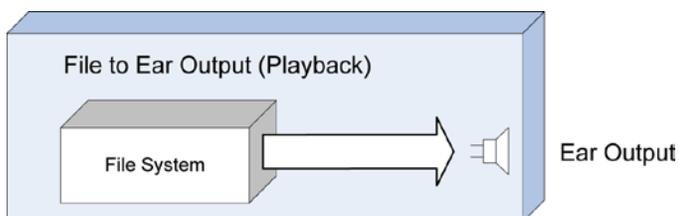

Figure 7. Playback: File > Speaker Output *

Figure 8. Playback: File > Ear Output *

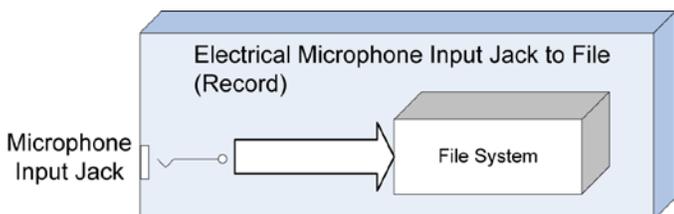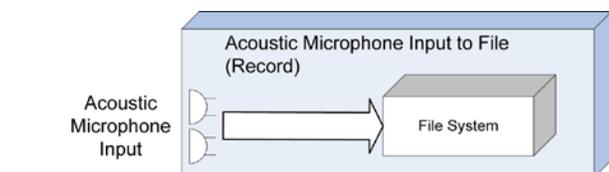
Figure 9. Record: Microphone Input jack > File
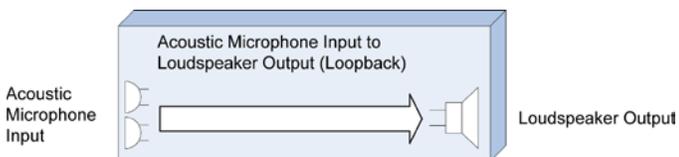

Figure 10. Record: Microphone Input * > File

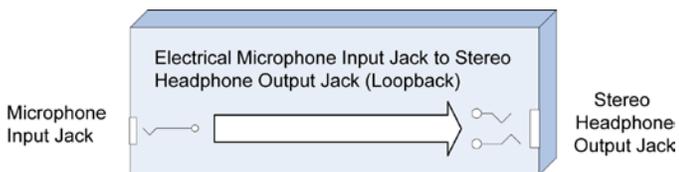
Figure 11. Loopback: Microphone Input * > Speaker *
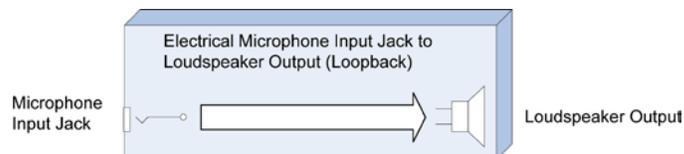

Figure 12. Loopback: Microphone Input Jack

Figure 13. Loopback: Microphone Input Jack > Speaker *

Headphone Plug Connections

Smartphone Audio Jack and Plug

Smartphones use one 3.5 mm "mini" jack to accomplish mono headphone (earbud), stereo headphone (earbuds) and microphone connections and operational modes. The jack provides four conductors in a TRRS (tip-ring-ring-sleeve) configuration, typically designated as shown in Figure 14. When a plug is inserted, the smartphone senses the impedance between the conductors to determine operational mode and connection.

The 3.5 mm TRRS plug has four conductors, typically designated as shown in Figure 14. A smartphone device will sense the connection of mono or stereo headphones, and the availability of a microphone signal, and connect accordingly.

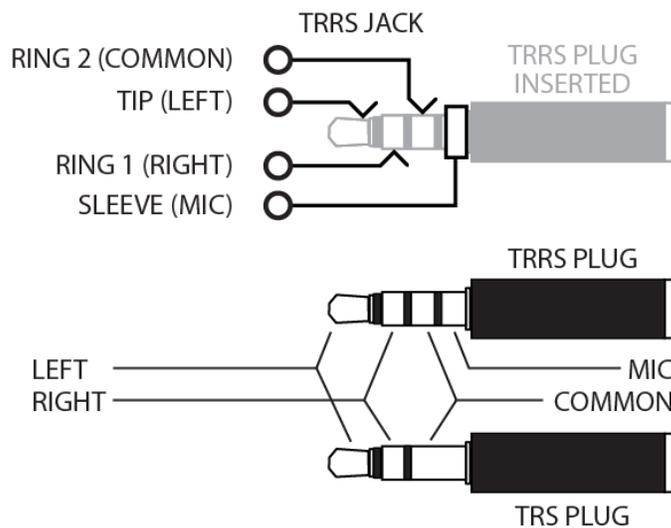Figure 14. Typical audio jack and plug signal assignments.

For more information, see the Table of Smartphone App Analog Settings beginning on page 29.

Microphone Input Connections

The smartphone analog microphone input circuit is designed to sense jack insertion and to determine whether or not a microphone is connected. The circuit provides +2.5 VDC for external microphone bias, and uses this to de-

tect microphone presence. The use of a TRS plug, or a direct connection to the APx low impedance analog generator output, will short-circuit this voltage. The microphone sense circuitry will detect this short, turn off the electrical microphone input, and route the internal (acoustic) smartphone microphone(s) to the smartphone microphone circuits.

To avoid this when using an APx, the microphone connection to the APx analog generator unbalanced output requires an RC filter, shown in Figure 15. Insert a 10 µF series coupling capacitor and a 2.2 kΩ resistor shunt to ground (at the smartphone jack side of the capacitor) to simulate a normal microphone DC impedance. The result is an AC coupled low-frequency high-pass filter. The cutoff frequency depends on the electrical characteristics of the microphone sense circuit, typically −0.125 dB @ 100 Hz.
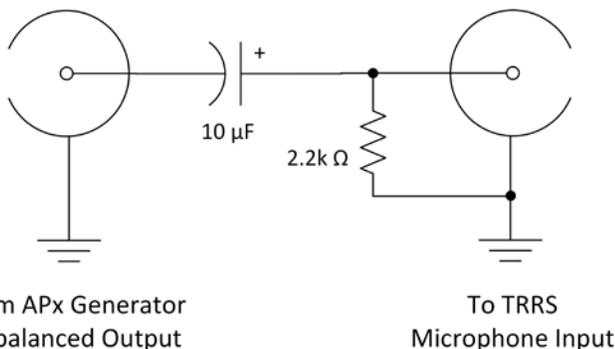

Figure 15. Microphone input coupling circuit.

Electrical microphone input circuits require very low audio signal levels, typically 5 mV rms to 100 mV rms. These low level audio signals are subject to interference caused by the USB connection to the PC. This will increase noise and distortion measurements and prevent accurate measurement of the microphone input circuit.

To avoid this analog noise, disconnect the USB during recording, or connect the Android device to a USB port on an unused USB hub. Don't connect the device to a port on the same USB hub connected to the APx audio analyzer.

Headphone Output Loading

Headphone output power and distortion at full volume should be tested with normal resistive loads that emulate earphone speakers, such as 32 Ω per channel.

The Analog Menu

The Analog menu displays controls and settings for testing analog audio streams through the electrical TRRS jack and through the acoustic microphone, ear piece, and speaker (depending on SETTINGS).
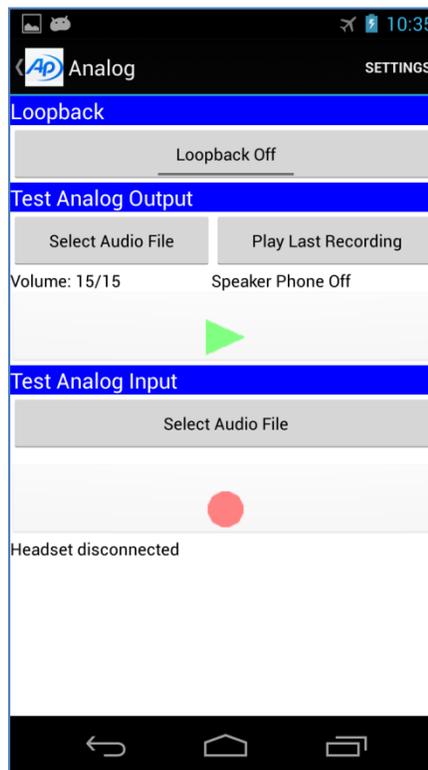
Figure 16. Analog menu, Loopback in the Off state

Loopback Button

Touch the **Loopback** button to toggle between **On** and **Off**. This function routes audio input from acoustic microphones or the microphone input jack to audio outputs (ear, speaker, and/or headphone jack). The specific inputs and outputs depend on Settings selections.

NOTE: Acoustic feedback between the speaker and microphone will cause high volume noise from the speaker if Loopback is enabled while Speakerphone is enabled and a jack is NOT inserted.

Loopback cannot be selected if a file is being played or recorded with the Analog menu. If **Loopback** is **On** (see Figure 24), then playback and recording are disabled in the Analog menu until Loopback is turned Off.

Test Analog Output

The Select Audio File Button

Use **Select Audio File** to select an audio file from the device SD Card memory. This invokes OI File Manager to select a file for playback from a list of existing files. The selected filename will appear on the **Play/Stop** ▶️🟦 button. For best audio performance select a file encoded at the base sample rate for the device, typically 44.1 kHz or 48 kHz.

Manually (or remotely via adb.exe) set the device audio volume level to maximum for accurate output measurements with the test signals provided in the Audio Precision Technote 120 download.

> You may need to override the Android volume limiter in order to set the volume to maximum with a TRRS plug inserted into the headset jack.

The Play/Stop Button ▶️🟦

Use the **Play/Stop** ▶️🟦 button to start and stop streaming the selected file to the selected audio connection (Ear, Speaker, or Headphone Jack). The file will continue to play until stopped or until you exit this menu.

The Play Last Recording Button

Use **Play Last Recording** to play the last recorded file.

Test Analog Input

The Select Audio File Button

Use **Select Audio File** to select an audio file for recording to the device SD Card memory. The selected filename will appear on the **Record/Stop** 🔴🟥 button. This invokes OI File Manager to select a file from a list of existing files or enter a new filename for recording. Name the file with a .wav extension.

Use the APx audio analyzer microphone measurement projects provided in Technote 120 to determine the maximum input level for the microphone jack.

The Record/Stop Button 🔴🟥

The **Record/Stop** 🔴🟥 button starts and stops recording to the selected filename indicated on the button. When recording starts this automatically increments the filename numeric value if the **Record Settings Increment Filename** checkbox is checked in Settings.

If the file exists (length greater than 44 bytes) a **Confirm Record** menu will display "Do you really want to record over the selected file?" Answer **Yes** to proceed or **No** to abort recording.

Status Activity Display and Log

The Smartphone App displays a log of status activity in the white area at the bottom of each Play or Record screen. If you touch and hold in the status area, you will be given the option to clear the status information or to save the information as a text file in the default Log folder (\sdcard\AudioPrecision\Log\).

Status Log file naming

For Analog status information, saved Log text file names begin with "ANALOG.", followed by a date and time string, such as "ANALOG.20151126094410.txt".

SETTINGS Menu

Use the SETTINGS button (top right of the screen) to display the Analog Settings menu in order to set recording, playback, and loopback options for analog (electrical and acoustic) signal routing.
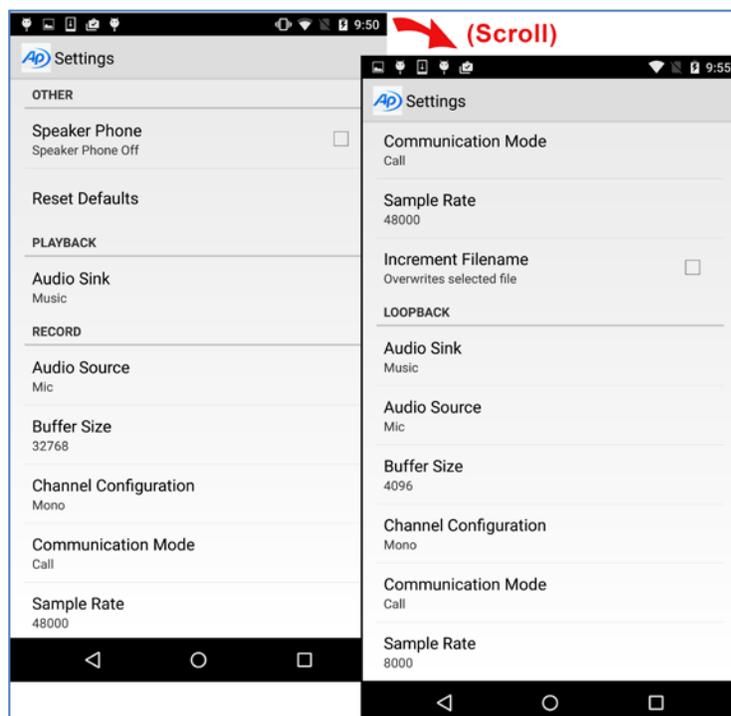

Figure 17. Analog SETTINGS menu

Select the back button to return to the Analog menu.

Other

Reset Defaults resets Analog settings to default values.

Speaker Phone

Select the Speaker Phone checkbox to toggle Speaker Phone On or Off. The electrical audio signal path within the device depends on Jack insertion and Speaker phone On/Off.

The chart below shows which electrical inputs and outputs will be enabled for recording, playback, and loopback when Playback Audio Sink is set to Voice Call and Record Audio Source is set to Mic.

	Speakerphone ON		Speaker phone OFF	
	Jack Inserted	Jack NOT Inserted	Jack Inserted	Jack NOT Inserted
Phone Microphone		●	●	●
Headset Microphone	●			
Phone Speaker	●	●		
Phone Earpiece				●
Headset Earpiece			●	

Figure 18. Analog signal routing depends on jack insertion and Speakerphone mode.

Playback—Audio Sink

Select a sink path for playback of an audio file through the device to an output (ear, speaker, or headphone jack).
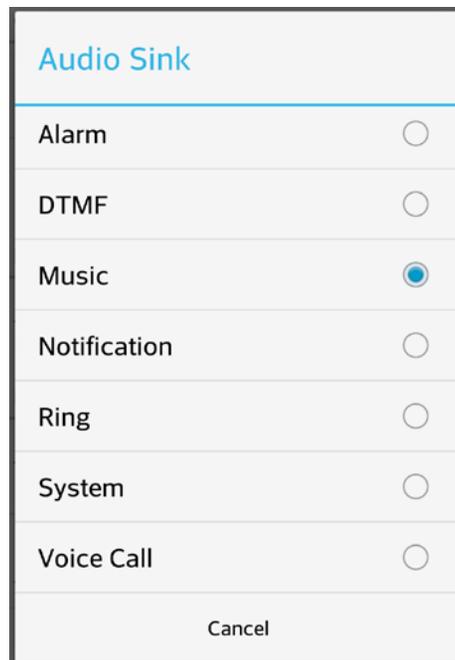

Figure 19. Analog Playback Audio Sink settings

Audio Sink Selections

- Alarm
- DTMF
- Music (the default)
- Notification
- Ring
- System
- Voice Call

The Android operating system for mobile devices exposes many output paths for audio. Most functions are hardware, firmware, and software dependent, and vary with Android versions. These selections provide the most flexibility for testing audio with your particular device. Of the possible selections available, System, Voice Call, and Music settings usually work well. Refer to the Android SDK documentation for more information about these signal paths.

Audio Sink selections may override Speaker Phone behavior. For example, Music will route audio to the speaker regardless of Speaker Phone setting, but Voice Call will not. Behavior depends on the device design.

Record—Audio Source

Select an audio source path for recording to a file through the device from an input (device acoustic microphone, or microphone jack).
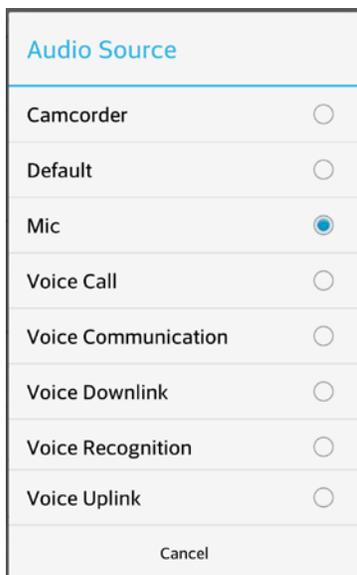

Figure 20. Analog Record Audio Source settings

Audio Source Selections

- Camcorder
- Default
- Mic (the default)
- Voice Call
- Voice Communication
- Voice Downlink
- Voice Recognition
- Voice Uplink

The Android operating system for mobile devices exposes many input paths for audio. Most functions are hardware, firmware, and software dependent, and vary with Android versions. These selections provide the most flexibility for testing audio with your particular device. Jack insertion and Speaker Phone settings will affect how these signal paths behave. Of the possible selections available, Camcorder, Default, Mic, and Voice Call usually work well. Refer to the Android SDK for more information about these signal paths.

Record—Buffer Size

Selects a recording buffer size with a slider and + or − buttons. The range of buffer size is 512 bytes to 32768 bytes in 512 byte increments.
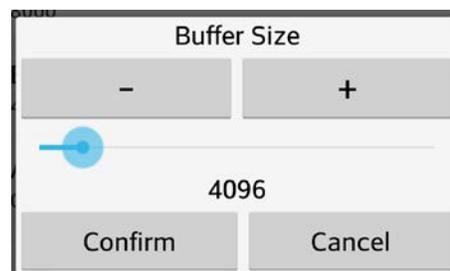
The default is 4096 bytes.

Figure 21. Analog Record Buffer Size setting

Select Confirm or Cancel to continue. The Confirm button will be disabled if the buffer size selected is below the minimum permitted buffer size for the combination of Channel Configuration, Communication mode, and Sample rate. Increase the buffer size until the Confirm button is no longer disabled.

Buffer size will affect the quality of audio recordings. Higher buffer sizes may be required for higher sample rates. A small buffer size may result in audio gaps in recording (dropouts).

Record—Channel Configuration

Select Mono (the default) or Stereo.

Figure 22. Analog Record Channel Configuration

Devices may have the capability of recording in Stereo when the Communication Mode setting is Camcorder, in which case the noise canceling microphone (near the ear position of a mobile phone) may be used as a separate input independent of the main microphone (near the mouth position of a mobile phone). However, even if this feature exists in a device it may not be accessible to this application if the device software has been customized by the OEM vendor.

Record—Communication Mode

Communication Mode selections:

- Call (the default)
- Communication
- Normal
- Ringtone

"Call" is preferred. Audio bandwidth and other behaviors may change with different settings.
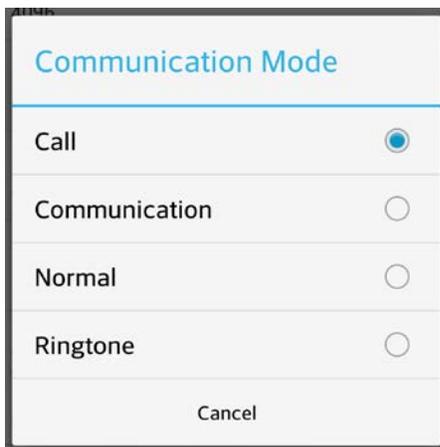
Figure 23. Analog Record Communication Mode settings

Record Sample Rate

Select a sample rate for recording. Not all available sample rates are usable, depending on the capabilities of the device. The selected sample rate will be indicated above the Record/Stop button.

Sample Rate selections:

- 48000
- 44100
- 32000
- 22050
- 16000
- 11025
- 8000 (the default)
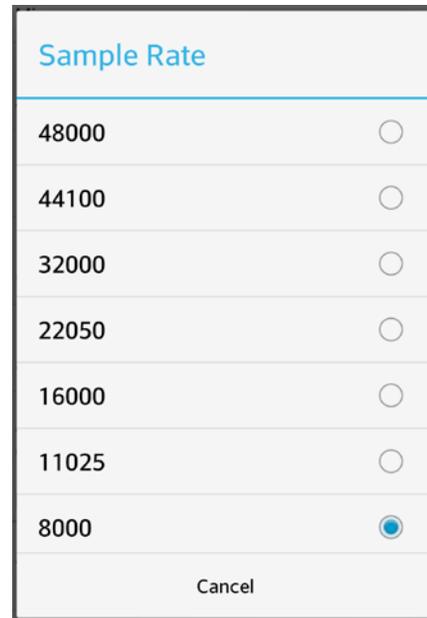
Figure 24. Record Sample Rate settings

Increment Filename

If selected (checked), the recording filename will be automatically incremented with a suffix each time recording is started. If myfile.wav is selected for recording, the filename will be myfile.0000.wav and the numeric part of the filename will be incremented the next time the record button is selected.

If not selected (un-checked), the recording filename will not be changed and subsequent recordings will overwrite the selected file.
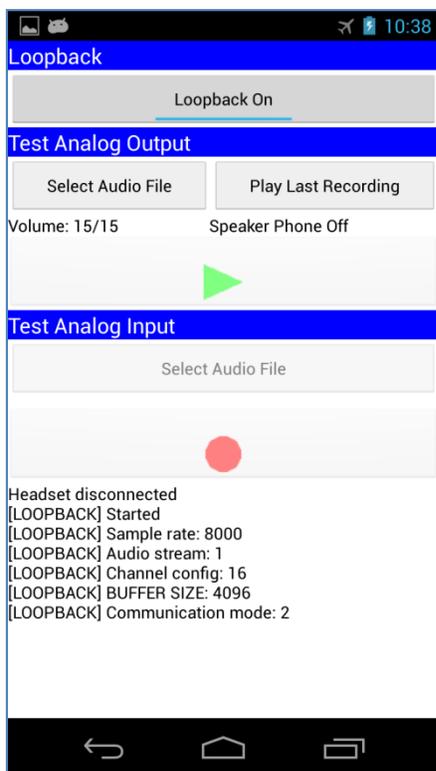
Figure 25. Loopback On

Figure 26. Loopback Audio Sink settings

Loopback—Audio Sink

Select a path for streaming audio to an output (ear, speaker, or headphone jack) from an Audio Source.

Audio Sink selections:

- Alarm
- DTMF
- Music (the default)
- Notification
- Ring
- System
- Voice Call

Music and Voice Call work best for most Loopback signal paths. See the Loopback signal paths described in Figure 45 for recommended settings.

Loopback—Audio Source

Select an audio source path for streaming audio from an input (acoustic microphone or microphone jack).

Audio Source Selections:

- Camcorder
- Default
- Mic (the default)
- Voice Call
- Voice Communication
- Voice Downlink
- Voice Recognition
- Voice Uplink

The Android operating system for mobile devices exposes many input paths for audio. Most functions are hardware, firmware, and software dependent, and vary with Android versions. These selections provide the most flexibility for testing audio with your particular device. Jack insertion and Speaker Phone settings will affect how these signal paths behave. Of the possible selections available, Camcorder, Default, Mic, and Voice Call usually work well. Refer to the

Android SDK help for more information about these signal paths.
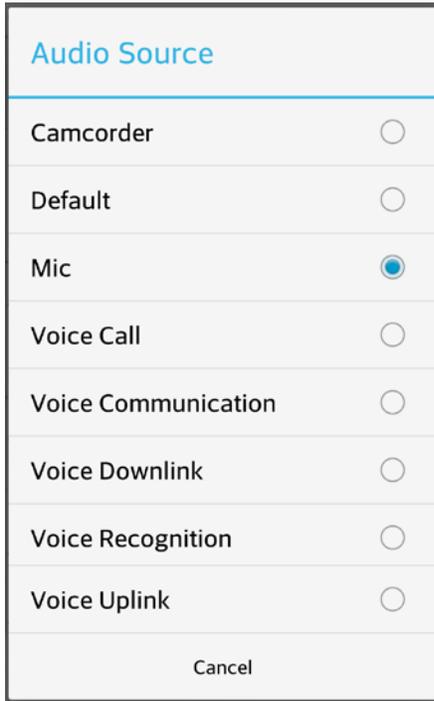
Figure 27. Loopback Audio Source settings

Loopback—Buffer Size

Selects a minimum buffer size for audio streaming with a slider and + or − buttons. The range of buffer size is 512 bytes to 32768 bytes in 512 byte increments.

The default is 4096 bytes.
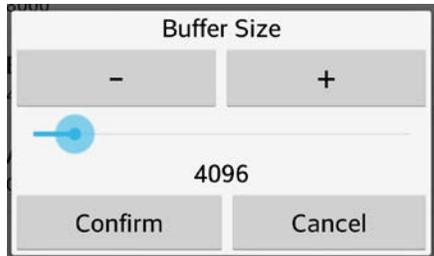
Figure 28. Loopback Buffer Size settings

Select Confirm or Cancel to continue. The Confirm button will be disabled if the buffer size selected is below the minimum permitted buffer size for the combination of Channel Configuration, Communication mode, and Sample rate. Increase the buffer size until the Confirm button is no longer disabled.

Buffer size will affect the quality of audio streaming. Higher buffer sizes may be required for higher sample rates. A small buffer size may result in audio gaps (dropouts).

Latency will increase with larger buffer sizes.

Loopback—Channel Configuration

Select Mono (the default) or Stereo.

Figure 29. Loopback Channel Configuration setting

Loopback—Communication Mode

Communication Mode selections:

- Call (the default)
- Communication
- Normal
- Ringtone

"Call" is preferred. Audio bandwidth and other behaviors may change with different settings.
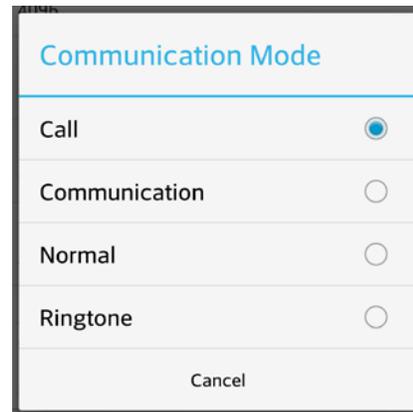
Figure 30. Loopback Communications Mode settings

Loopback—Sample Rate

Select a sample rate for input streaming. Not all available sample rates are usable, depending on the capabilities of the device. The selected sample rate will be indicated above the Record/Stop button.

Sample Rate selections:

- 48000
- 44100
- 32000
- 22050
- 16000
- 11025
- 8000 (the default).
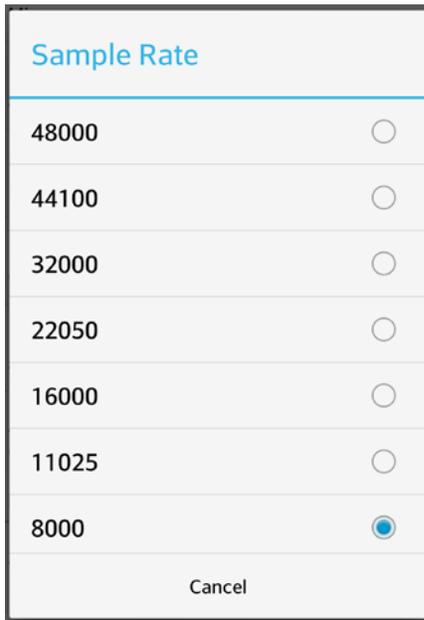
Figure 31. Loopback Sample Rate settings

Typical Smartphone Analog Audio Test Results

The test results shown below are typical for high performance Android smartphones.

Headphone Jack Output

Smartphones provide a TRRS jack for stereo headphones and external monaural electret microphone. The APX audio analyzer measures the stereo headphone outputs during file playback by the app. These results were obtained with the APx project **Analog Headphone Jack, Automated.approjx**. Measurements were performed with test signals encoded at 16-bit 44.1 kHz and 24-bit 48 kHz, with volume set to maximum and outputs terminated with 32 ohm loads to simulate a typical headphone load.

Performance at 24-bit 48 kHz did not differ from 16-bit 44.1 kHz other than frequency response. The graphs below were obtained with 16-bit 44.1 kHz test signals.

Single Tone Sinewave Measurements

Playback Test Signal:	997 Hz sinewave @ 0 dBFS
Measurement BW:	20−20 kHz

Level	252 mV
Output Power	1.985 mW
THD+N	0.0113 %
THD	0.0019 %
Noise Ratio	0.0077 %
Dynamic Range	84.2 dB
Crosstalk @ 10 kHz	−40.5 dB

FFT Spectrum

THD+N Ratio
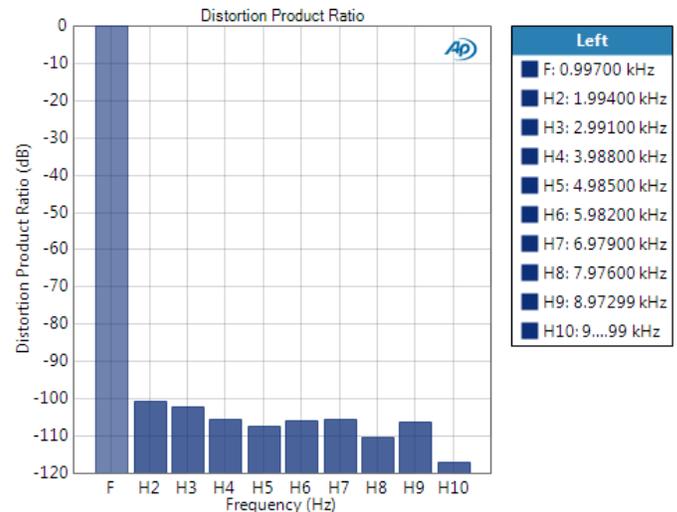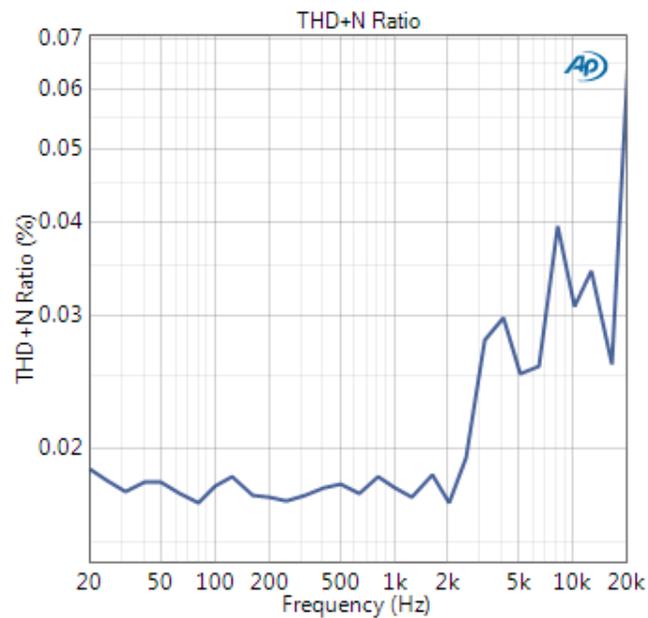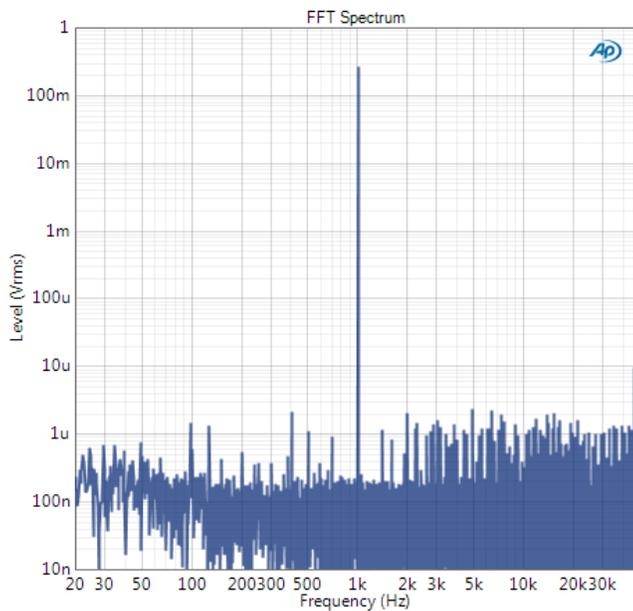
Stepped Frequency Sweep

Playback Test Signal: 3rd octave swept sinewave @ −20 dBFS level

Measurement BW: DC−45 kHz

Relative Level (1.00000 kHz)

THD Ratio
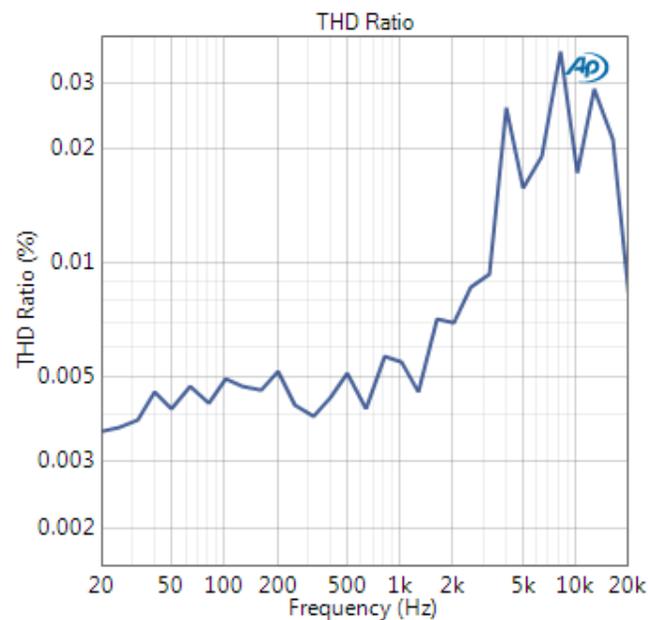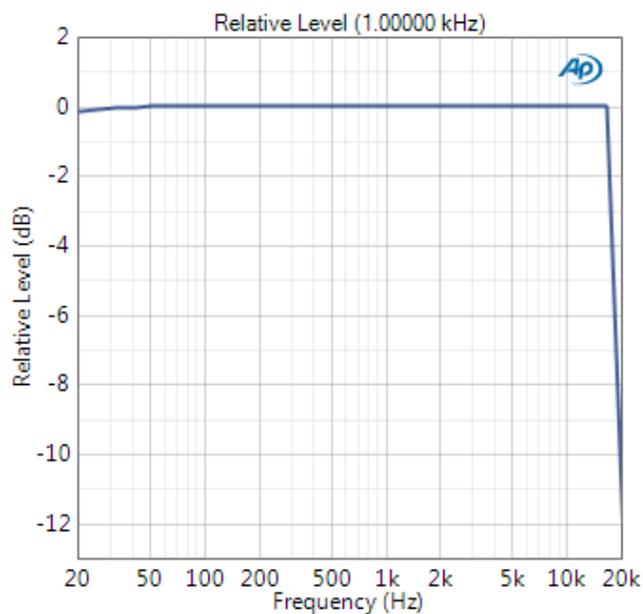
Measurement Recorder Stepped Level

This test displays output gain linearity with a 400 Hz sinewave as the audio level changes versus time. Audio level at the output is measured with a 400 Hz bandpass filter while the playback file decrements level from 0 dBFS to −90 dBFS in 5 dB steps at 3 second intervals. 0 dBrA represents full scale output at 0 dBFS audio level with volume at maximum.

The measured level at each step should correspond to a 5 dB change in the playback level and should align with the vertical scale markings on the graph. In this headphone output the level compresses when playback level is below −75 dBFS.

Bandpass Level dBrA
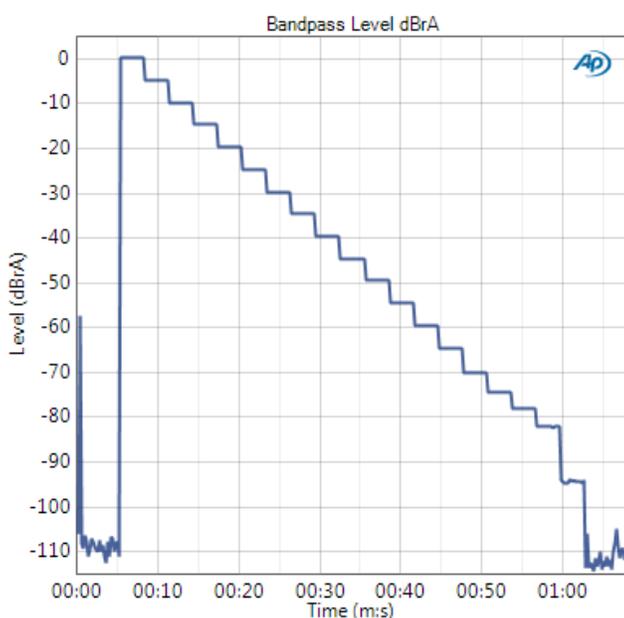
Multitone Measurements

Playback Test Signal: 0 dBFS Multitone with 31 tones from 20 Hz to 20 kHz.

The Multitone Analyzer tests total distortion (harmonic + intermodulation) and frequency response with a worst-case test signal.

TD+N Ratio (Total Distortion + Noise)	0.02%
Frequency Response Deviation	± 0.1 dB
Signal-to-Noise Ratio	77.5 dB
Frequency Shift	−0.772 ppm

FFT Spectrum
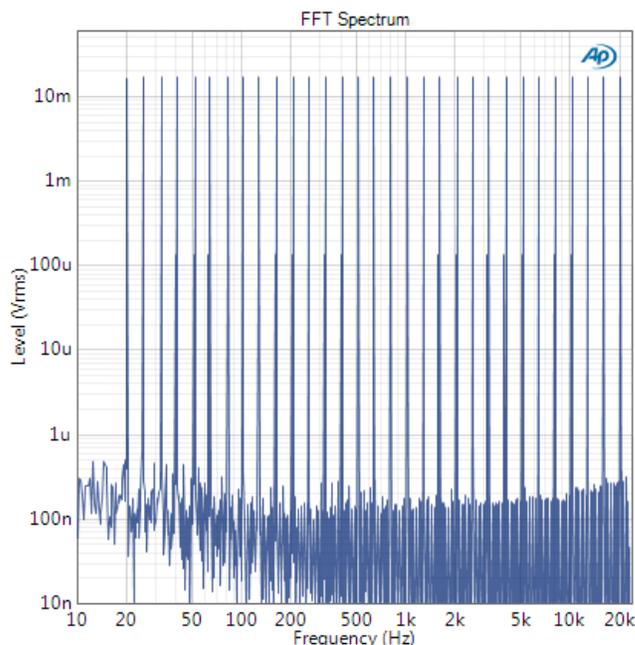
Microphone Jack Input

Audio tests provided in the APx project file **Analog Microphone Jack Wideband and Narrowband, Automate.approjx** record microphone input through the TRRS jack, transfers recorded waveform files to the PC, and then analyzes the files.

Single Tone Sinewave Measurements

Input Test Signal:	1 kHz sinewave @ −1 dBFS
	0 dBFS = 55 mV RMS
Measurement BW:	20–24 kHz

Level	−1.003 dBFS
THD+N	0.2438 %
THD	−64.8 dB
Noise Ratio	−55.821 dB
Dynamic Range AES17	57.21 dB

Distortion Product Ratio

FFT Spectrum

Relative Level (1.00000 kHz)

THD+N Ratio

THD Ratio
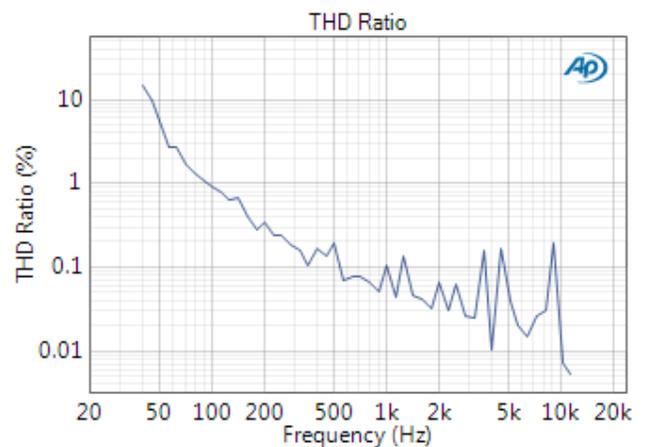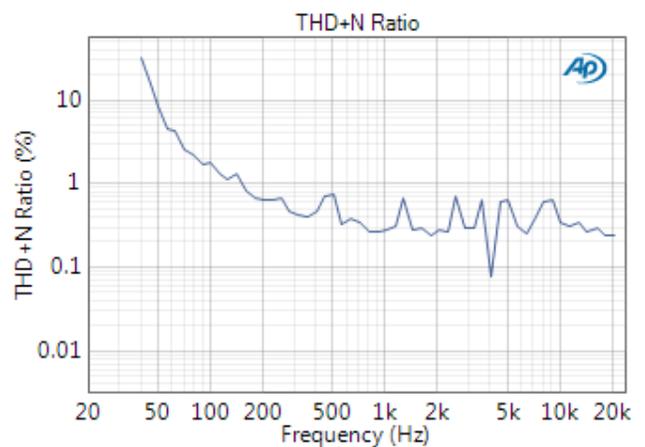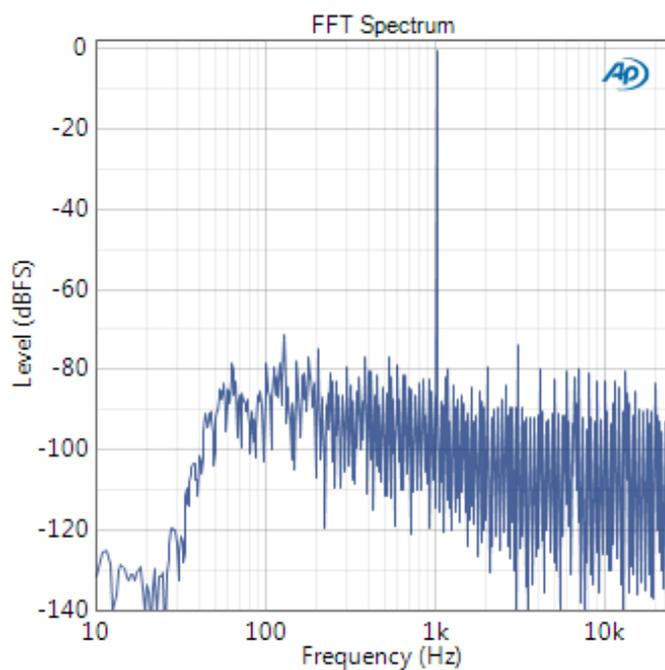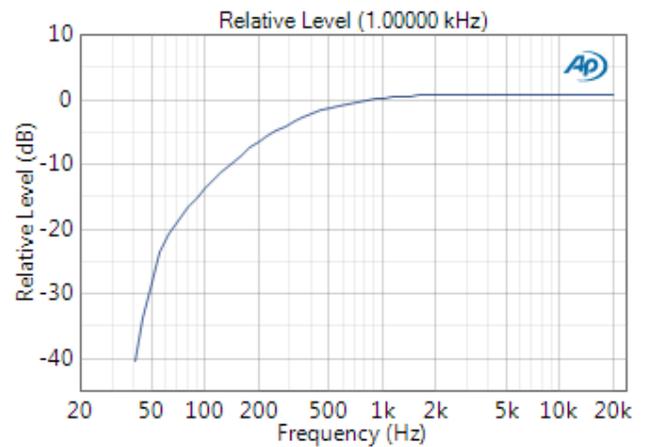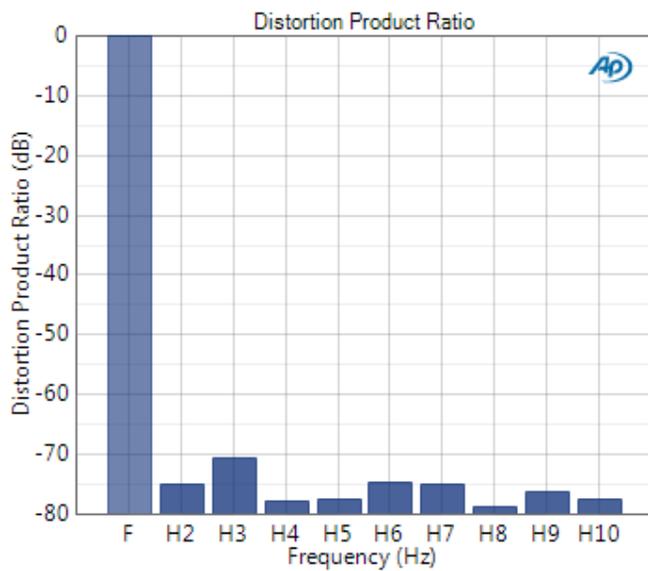
Stepped Frequency Sweep

Input Test Signal: 20–20 kHz 6th octave swept sinewave at −6 dBFS input level

Measurement BW: DC–24 kHz

Relative Level measures frequency response. The low frequency response shown here is the result of the RC input coupling in the microphone input circuit. See Figure 15 for an explanation.

Multitone Measurements

Input Test Signal: 0 dBFS Multitone with 31 tones from 20 Hz to 20 kHz.

The Multitone Analyzer tests total distortion (harmonic plus intermodulation) and frequency response with a worst-case test signal.

|	
TD+N Ratio (Total Distortion + Noise)	0.578%
Signal-to-Noise Ratio	45.287 dB
Frequency Shift	0.329 ppm

Bluetooth: Pair Device Menu

For *Bluetooth* testing, the APx audio analyzer must have the *Bluetooth* hardware option installed and must be configured for *Bluetooth* profile set **A2DP Sink, HFP Hands-Free, AVRCP Controller**.

This opens the smartphone's Android *Bluetooth* screen. If an Audio Precision APX analyzer has been properly configured for *Bluetooth* it will appear in the Paired devices or Available devices lists.

Figure 32 shows an APx analyzer already paired and connected as a hands-free device. It is not connected as a media device capable of accepting an A2DP music stream from the smartphone.
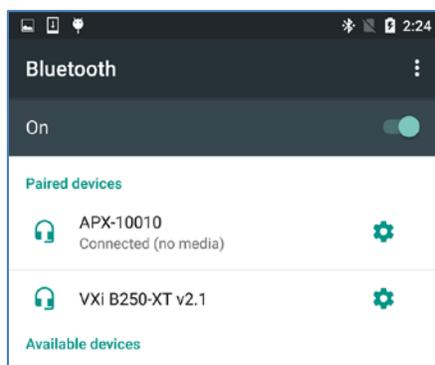
Figure 32. Android device *Bluetooth* menu

When configured with the correct *Bluetooth* profile the analyzer assumes the roles of a *Bluetooth* device, such as a media receiver for the A2DP profile, or a hands-free device used during a phone call. The APx project files provided with this Technote have been setup to accomplish this for you.

Bluetooth: A2DP menu

The A2DP menu button displays the *Bluetooth* Advanced Audio Distribution Profile test menu.
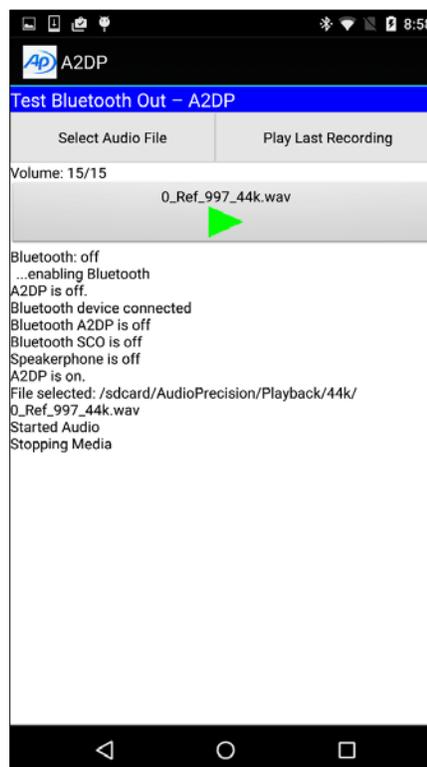
Figure 33. A2DP Menu

Test *Bluetooth* Out — A2DP

When this menu first appears it displays the Android device's *Bluetooth* menu (Figure 32) in order to manually select a *Bluetooth* device for pairing and connecting. See *Bluetooth*: Pair Device Menu above. Select the APx analyzer to pair and connect to it. Then exit the *Bluetooth* menu to automatically return to the A2DP menu.

Select Audio File Button

Select Audio File invokes OI File Manager to select a file for playback from a list of existing files installed on the device in AudioPrecision\Playback\..., as shown in Figure 34Figure 33. Select a file encoded at the base sample rate for the A2DP stream, typically 44.1 kHz. Devices may also support A2DP streaming at 16 kHz, 32 kHz and 48 kHz sample rates.

The selected filename will appear on the **Play/Stop** button, as shown in Figure 33.

Manually set the device audio volume level to maximum for accurate gain measurements with the test signals provided.
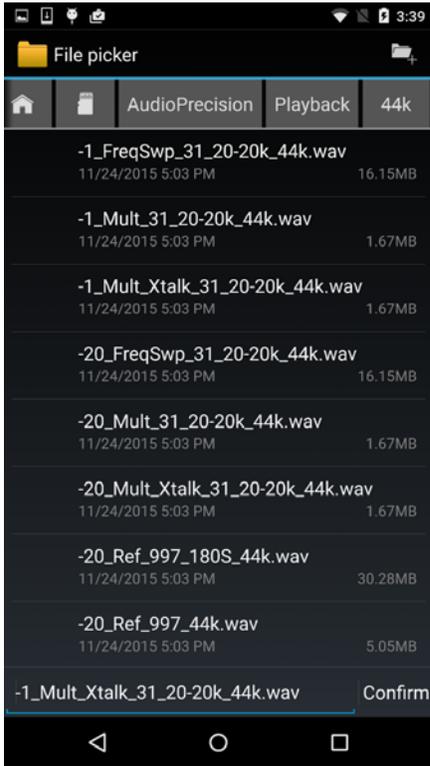
Figure 34. OI File Manager Playback

The Play/Stop Button ▶■

The **Play/Stop** ▶■ button starts and stops streaming of the selected file. The audio stream is transmitted through the device *Bluetooth* A2DP connection to an Audio Precision APx audio analyzer. The APx analyzer receives the audio stream and measures it in real-time. The file will continue to play until you stop or exit the A2DP menu.

Play Last Recording Button

The **Play Last Recording** button plays the last file that was recorded with the HFP or Analog test menus. If no recording has been made since startup then playback will not start.

Status Activity Display and Log

The Smartphone App displays a log of status activity in the white area at the bottom of each Play or Record screen. If you touch and hold in the status area, you will be given the option to clear the status information or to save the information as a text file in the default Log folder (\sdcard\AudioPrecision\Log\).

Status Log file naming

For *Bluetooth* A2DP status information, saved Log text file names begin with "A2DP.", followed by a date and time string, such as "A2DP.201511261350.txt".

Bluetooth: HFP menu

A smartphone with *Bluetooth* hands-free capability operates as an HFP (Hands Free Profile) Audio Gateway (AG) and the APx audio analyzer operates as the HFP Hands Free device (HF). Audio streaming from the AG to the HF is a downlink. Audio streaming from the HF to the AG is an uplink. Downlink and uplink operate simultaneously during a hands free phone call. The APx project files use this terminology to describe the type of test.
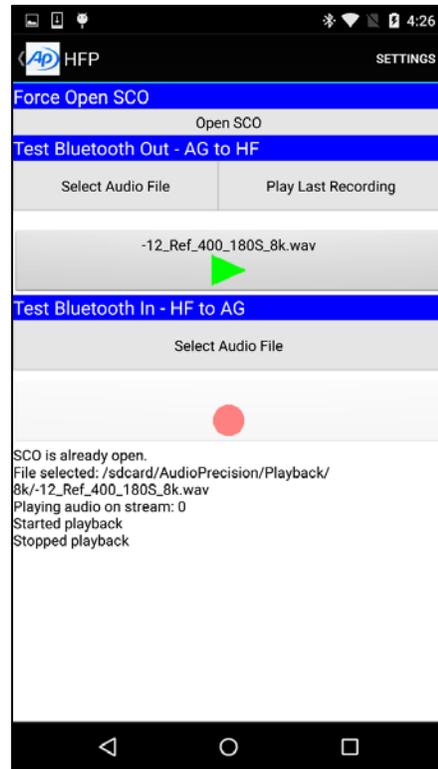
Figure 35. HFP menu

The HFP menu button displays the *Bluetooth* Hands Free Profile test menu. This menu sets up a hands-free connection to the APx audio analyzer (or other device) as though the device were in a phone call, for the purpose of testing the audio quality of the *Bluetooth* link.

When this menu first appears it displays the Android device's *Bluetooth* menu (Figure 32) in order to manually select a *Bluetooth* device for pairing and connecting. See *Bluetooth*: Pair Device above. Select the APx analyzer to

pair and connect to it. Then exit the *Bluetooth* menu to automatically return to the HFP menu.

HFP Volume

HFP audio volume is not controlled by the phone, but by the hands-free device, typically implemented with volume up and down buttons on the device. When testing audio quality of the HFP link with a smartphone device, the role of the hands-free device is performed by the APx audio analyzer's Bluetooth interface. In this role, the APx Bluetooth interface is capable of sending AT commands to emulate the volume buttons of a typical hands-free device.

When recording the uplink audio stream with the HFP Audio Gateway functionality of a smartphone, the APx Bluetooth interface must send the `AT+VGM` command with a volume number argument, such as **AT+VGM=15**. The value 15 sets the microphone gain to maximum in order to measure audio level accurately. The project files in this technote use sequence steps to send AT commands, as shown in Figure 36 and Figure 37.
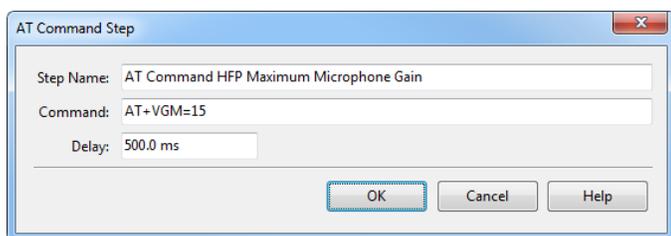
Figure 36 An AT Command Step within a test sequence sends an AT command to set the microphone gain to maximum for uplink recording.

When playing back an audio stream from the HFP Audio Gateway to the APx Bluetooth interface, the APx Bluetooth interface must send the `AT+VGS` command with a volume number argument, such as **AT+VGS=15**. The value 15 sets the speaker gain to maximum in order to measure audio level accurately.
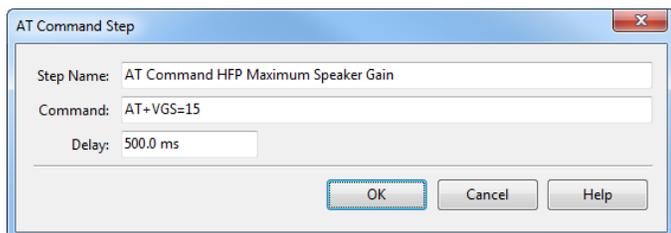
Figure 37 An AT Command Step sends an AT command to set the speaker gain to maximum for downlink playback.

Force Open SCO

If a hands-free device (HF) is paired and connected (such as the APx analyzer), a SCO (Synchronous Connection-Oriented link) is opened automatically.

Open SCO is for debug purposes. Use the **Open SCO** button to initiate a request from the device to open a SCO with the APx audio analyzer. During normal operation this is not necessary because the APx projects provided will automatically open a SCO.

Test *Bluetooth* Out—AG to HF

The Select Audio File Button
Select Audio File invokes OI File Manager to select a file for playback from a list of existing files installed on the device in AudioPrecision\Playback\..., see Figure 38.

.
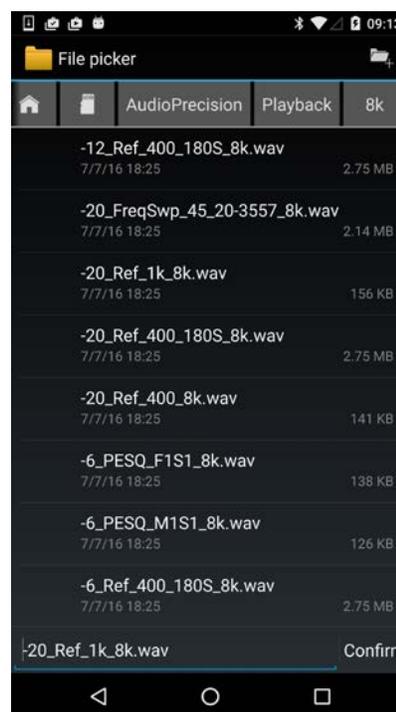
Figure 38 OI File Manager playback file selection.

The selected filename will appear on the **Play/Stop** ▶■ button.

The Play/Stop Button ▶■

Use the **Play/Stop** ▶■ button (green right arrow symbol) to start and stop streaming the selected file through the device *Bluetooth* HFP connection to the Audio Precision APx

audio analyzer for analysis. The file will continue to play until stopped or until you exit this menu.

The Play Last Recording Button

Use **Play Last Recording** to play the last file recorded with the *Bluetooth* HFP menu.

Test *Bluetooth* In — HF to AG

The Select Audio File Button

Select Audio File invokes OI File Manager to specify a file for recording. You can choose from a list of placeholder files installed on the device in AudioPrecision\Record\ or enter a new filename for recording, in the form *myfile*.wav. See Figure 39.

The selected filename will appear on the **Record/Stop** ⏺◼ button. The recorded file format will be monaural (single channel), 16-bit linear PCM. The audio sample rate will be the selected sample rate from the Record Settings, either 8 kHz (for standard narrow band CVSD codec) or 16 kHz (for wide band mSBC codec).
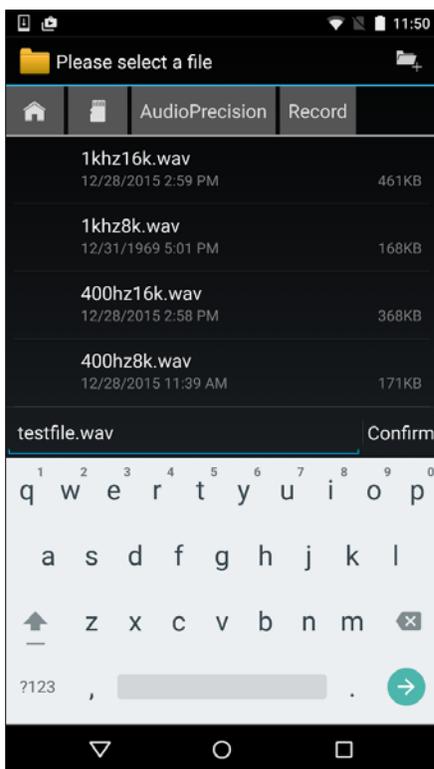
Figure 39. OI File Manager Record file selection.

The Record/Stop Button ⏺◼

The **Record/Stop** ⏺◼ button starts or stops recording to the selected filename indicated on the button face. When

recording starts this automatically increments the filename numeric value if the **Increment Filename** checkbox is set in the record SETTINGS menu.

If the file exists (length greater than 44 bytes) a **Confirm Record** menu will display "Do you really want to record over the selected file?" Answer **Yes** to proceed or **No** to abort recording.

If a SCO is not open a "SCO disconnected" menu will display "*Bluetooth* SCO is not Open! The HF device must be paired and connected." Select **OK** to abort recording.
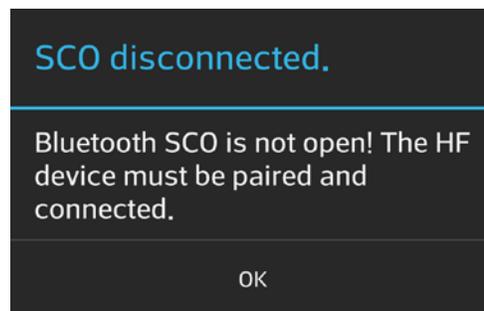
Figure 40. SCO disconnected warning

Status Activity Display and Log

The Smartphone App displays a log of status activity in the white area at the bottom of each Play or Record screen. If you touch and hold in the status area, you will be given the option to clear the status information or to save the information as a text file in the default Log folder (\sdcard\AudioPrecision\Log\).

Status Log file naming

For *Bluetooth* HFP status information, saved Log text file names begin with "HFP.", followed by a date and time string, such as "HFP.201511261410.txt".

SETTINGS menu

Use SETTINGS (top right of the HFP screen) to display the HFP Record Settings menu in order to set recording options for the input signal from the HF device (the APx audio analyzer *Bluetooth* interface).
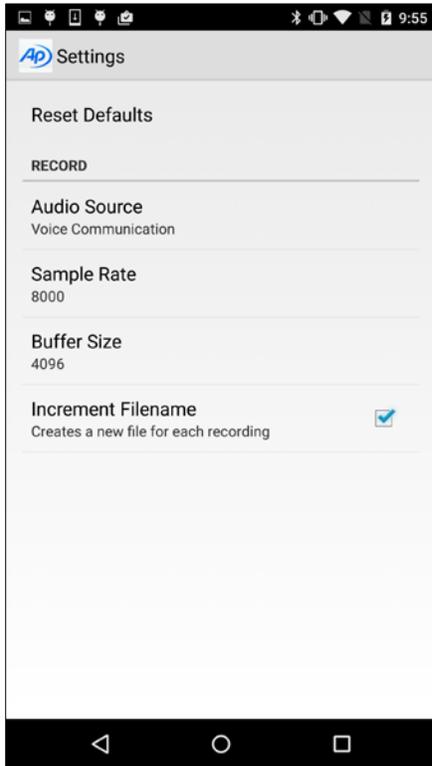
Figure 41. HFP Settings menu

Select the back button to return to the HFP menu.

Reset Defaults

Resets HFP recording settings to default values.

Audio Source

The default is Voice Communication.

- Camcorder
- Default
- Mic
- Voice Call
- Voice Communication (default)
- Voice Downlink
- Voice Recognition
- Voice Uplink

The Android operating system exposes many input paths for audio. Most functions are hardware, firmware, and software dependent, and vary with Android versions. These selections provide the most flexibility for testing audio with your

particular device. Of the possible selections available, Default and Voice Call usually work well. Refer to the Android SDK for more information about these signal paths.
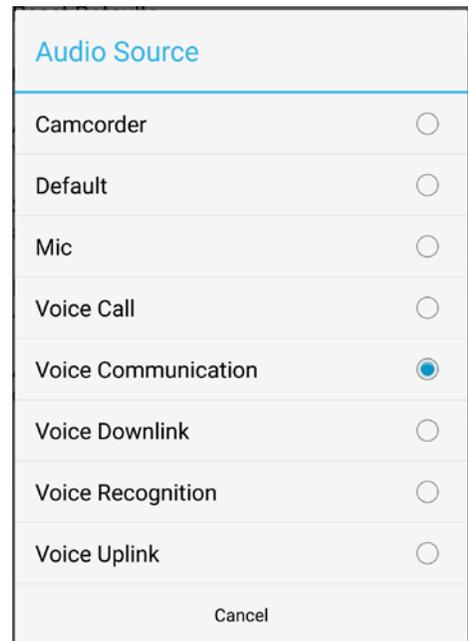
Figure 42. HFP Record Audio Source setting

Sample Rate

The default is 8 kHz.

Touch a radio button to make a selection. Select 8 kHz for standard HFP CVSD codec configurations. Select 16 kHz for HFP mSBC codec configurations. Cancel to make no change.

Figure 43. HFP sample rate setting

Buffer Size

Touch Buffer Size to display a control.

Select a recording buffer size with a slider and + or − buttons. The range of buffer size is 512 bytes to 32768 bytes in 512 byte increments. The default is 4096 bytes. Select Confirm to keep the setting or Cancel to make no change.
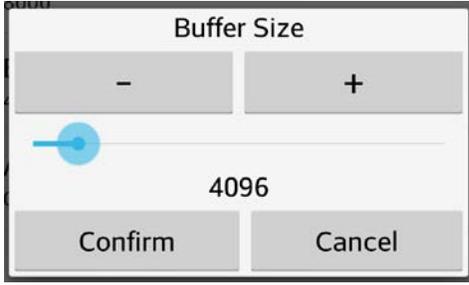
Figure 44. HFP Record Buffer Size setting

Increment Filename

If selected (checked), the recording filename will be auto-matically incremented with a suffix each time recording is started. If myfile.wav is selected for recording, the filename will be myfile.0000.wav and the numeric part of the file-name will be incremented the next time the record button is selected.

If not selected (un-checked), the recording filename will not be changed and subsequent recordings will overwrite the selected file.

Typical Smartphone Bluetooth Audio Test Results

Bluetooth A2DP Streaming

Audio tests provided in the APx project file *Bluetooth* A2DP, Automated.approjx play waveform files and measure audio received at the APx *Bluetooth* Interface.

Measurement Recorder Dropouts

The *Bluetooth* wireless link is subject to interference that may corrupt audio measurements. The Measurement Re-corder plots measurements over a 30 second time interval in order to confirm the link has no significant interference and that clock recovery results in stable sample rate and frequency scaling.

These plots show typical performance with no significant RF interference.
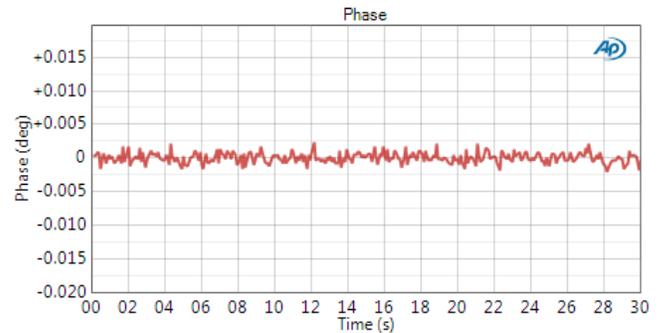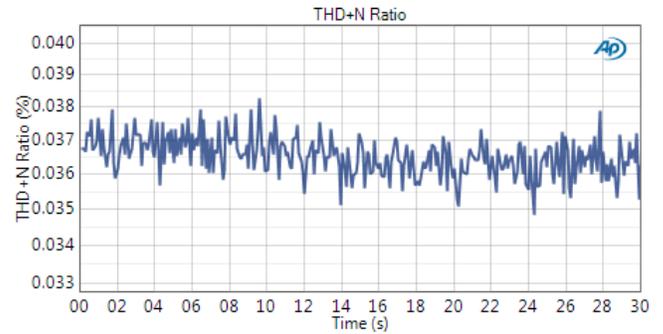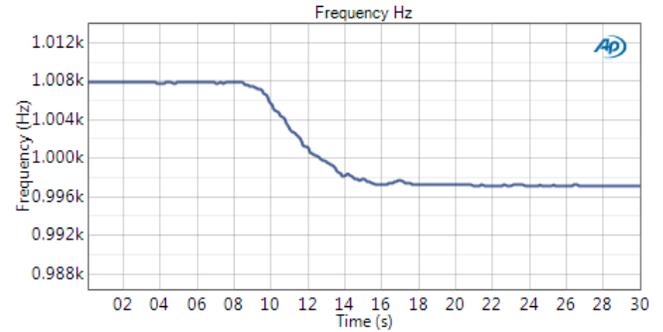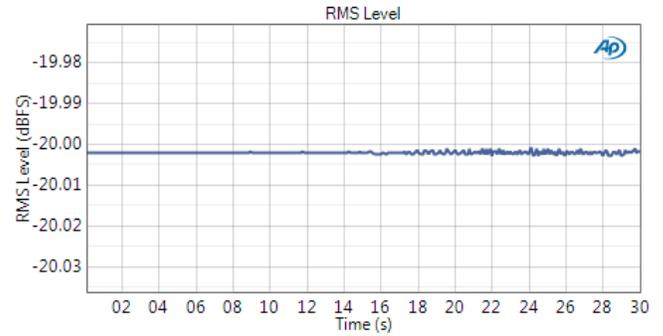
Bluetooth A2DP—Spectrum Analysis

Spectrum analysis provides insight about audio problems in the A2DP link or the SBC codec.
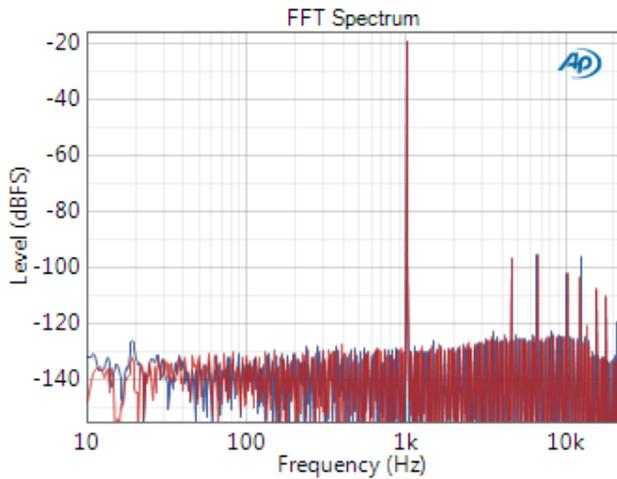
Bluetooth A2DP—Stepped Frequency Sweep

Stepped Frequency Sweep provides measurements over the 20 to 20 kHz audio band.
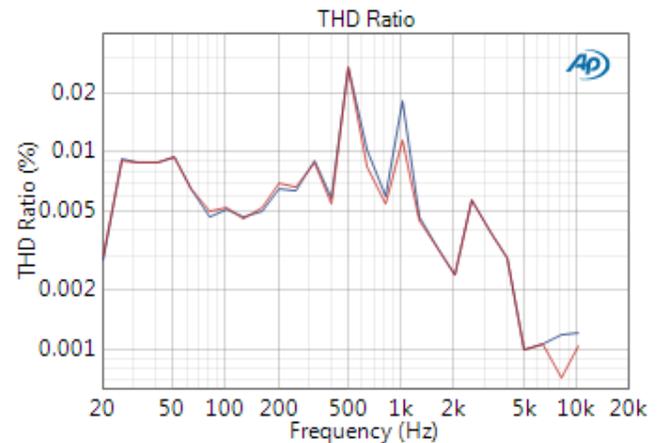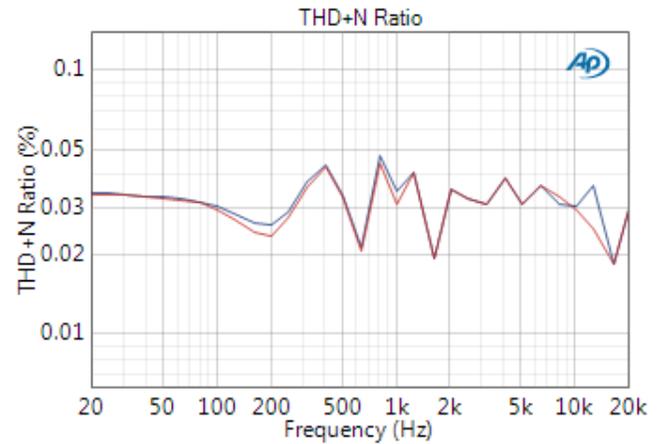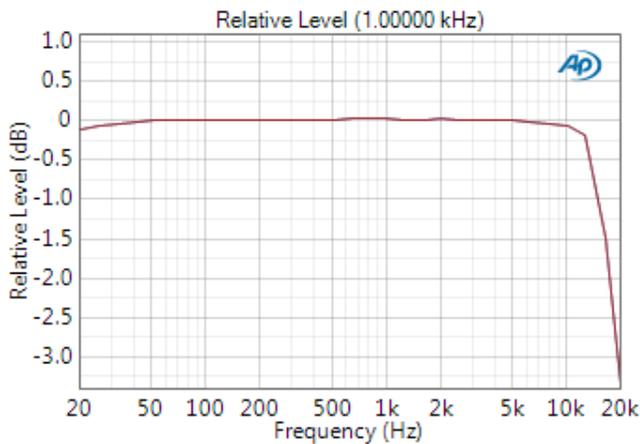
Bluetooth A2DP—Stepped Level Sweep

A2DP audio level is measured while the playback file decrements level from 0 dBFS to −90 dBFS in 5 dB steps at 3 second intervals. The steps represent gain linearity.
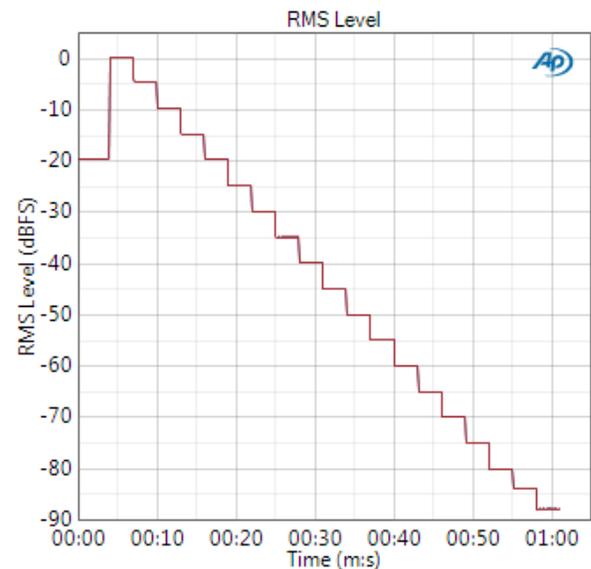
Bluetooth HFP Downlink

Bluetooth HFP downlink emulates a phone call in progress in order to test the quality of the connection from the phone to the hands-free device (the APx analyzer). All smartphones support *Bluetooth* 4 kHz narrowband voice with the CVSD codec. Many new phones also support wide-band 8 kHz HD Voice capability with the *Bluetooth* mSBC codec. Both codecs can be tested with the APx *Bluetooth* interface.

Wideband HFP may be tested with the ***Bluetooth*** **HFP Wide-band Downlink, Automated.approjx** project. A similar project file is provided for narrowband voice.

Measurement Recorder—*Bluetooth* RF Interference Detection

The *Bluetooth* HFP wireless link has poor error recovery and is subject to RF interference that will corrupt audio measurements. We use the Measurement Recorder to plot measurements over a time interval in order to determine if the RF test environment is likely to cause interference and corrupt the audio stream. Audio measurements quality may degrade if RF interference is high or the *Bluetooth* radio performs poorly.

THD+N distortion is very sensitive to audio dropouts and may be used to reveal the effects of poor *Bluetooth* radio performance. The Measurement Recorder detects dropouts as brief bursts of high distortion, shown in the graph below.
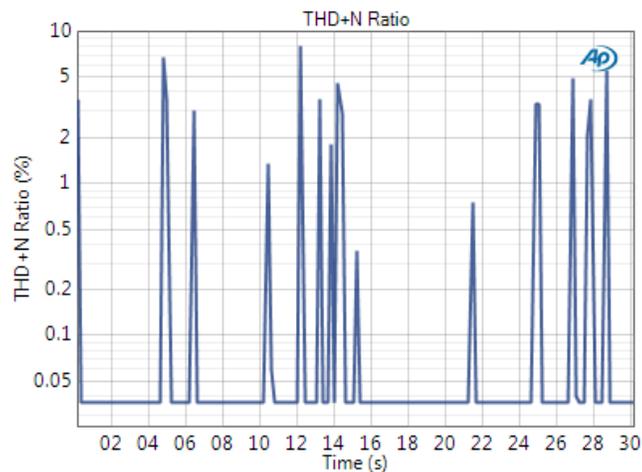
Measurement Recorder does not mask audio glitches because it disables the normal APx measurement settling algorithm. Other APx measurements enable settling and may be immune to brief glitches in the audio stream.

Bluetooth HFP Downlink—Sinewave Measurements

Playback Test Signal:	16 kS/s sinewave @ −20 dBFS
Measurement BW:	0–8 kHz
Level @ 1 kHz	−20.032 dBFS
THD+N @ 1 kHz	0.0452 %
THD @ 1 kHz	0.0452 %
SINAD @ 400 Hz	61.507 dB

Bluetooth HFP Downlink—Stepped Frequency Sweep

Stepped Frequency Sweep provides measurements over the 20 to 8 kHz HD audio band. Brief *Bluetooth* audio glitches may affect the measurements. This is evident in the high distortion at 80 Hz and 4 kHz in the THD+N Ratio graph below.

THD Ratio
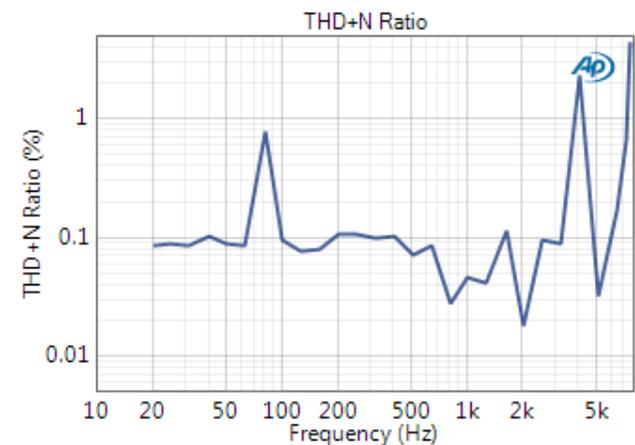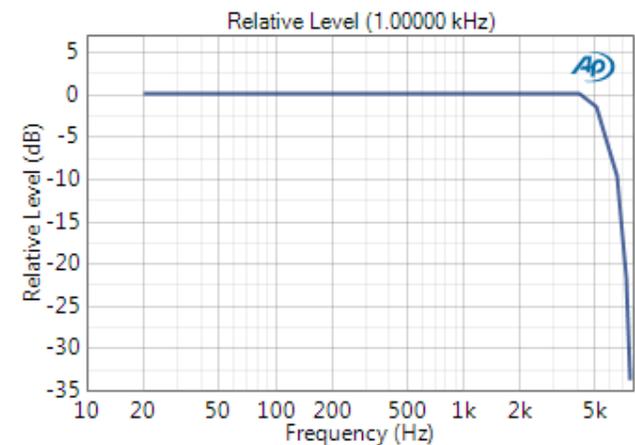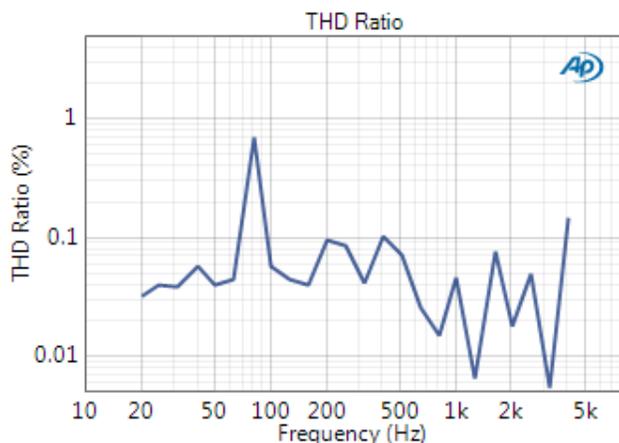
Bluetooth HFP Downlink—Stepped Level Sweep

This test displays level as the playback level of a 400 Hz sinewave changes versus time. Audio level is measured while the playback file decrements level from 0 dBFS to −90 dBFS in 5 dB steps at 3 second intervals. The steps represent gain linearity. In this example level compresses below −70 dBFS due to the mSBC codec.

RMS Level
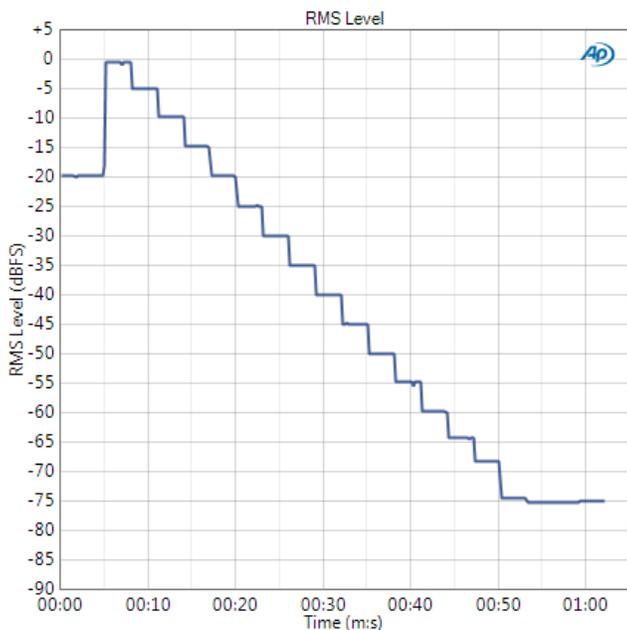
Bluetooth HFP Downlink—Voice Quality Assessment

APx audio analyzers equipped with the *Bluetooth* interface and PESQ or POLQA options can measure the Mean Opinion Score (MOS) of *Bluetooth* HFP voice transmission quality. The stimulus signal must be a properly designed voice

signal. This Technote provides waveforms with male and female voices for both PESQ and POLQA.

Voice quality can be degraded by high levels of *Bluetooth* RF interference, evident in the MOS vs Time graph as low MOS score glitches. This particular test had one interference glitch that resulted in a MOS score of 3.61. A subsequent retest had no interference, resulting in a 4.72 MOS score.

MOS vs Time

Acquired Waveform

Reference Waveform
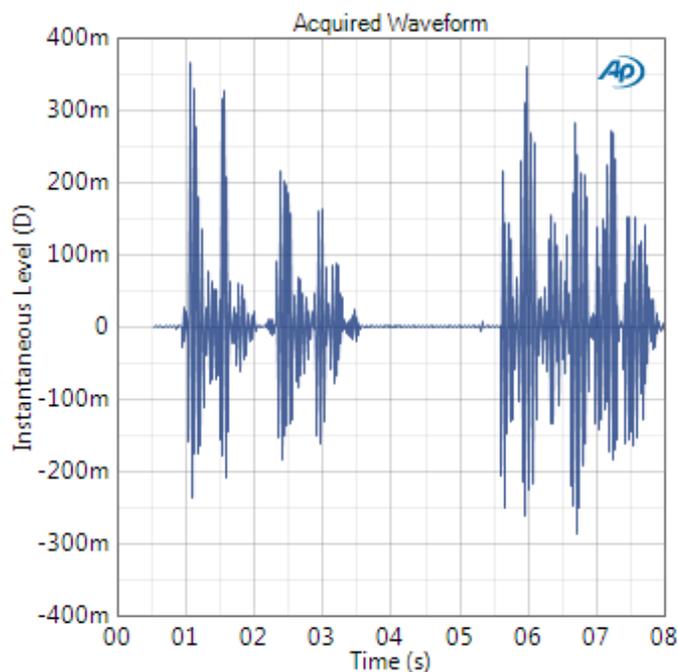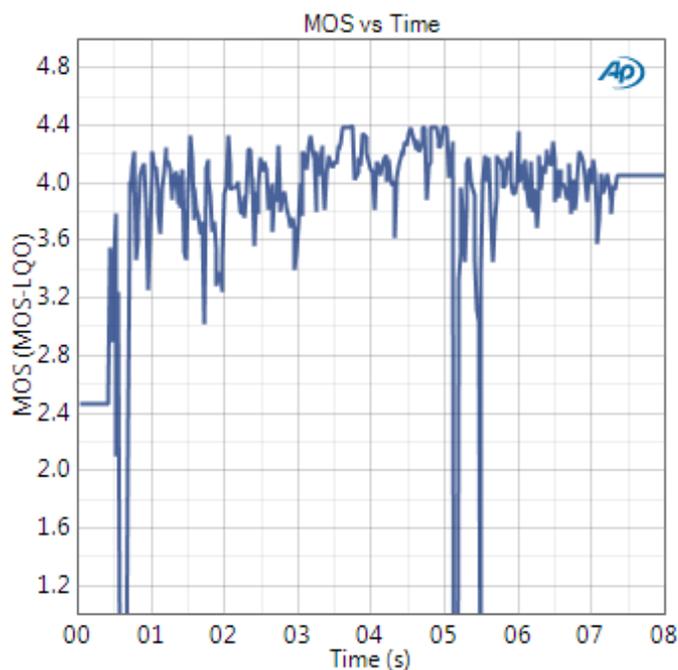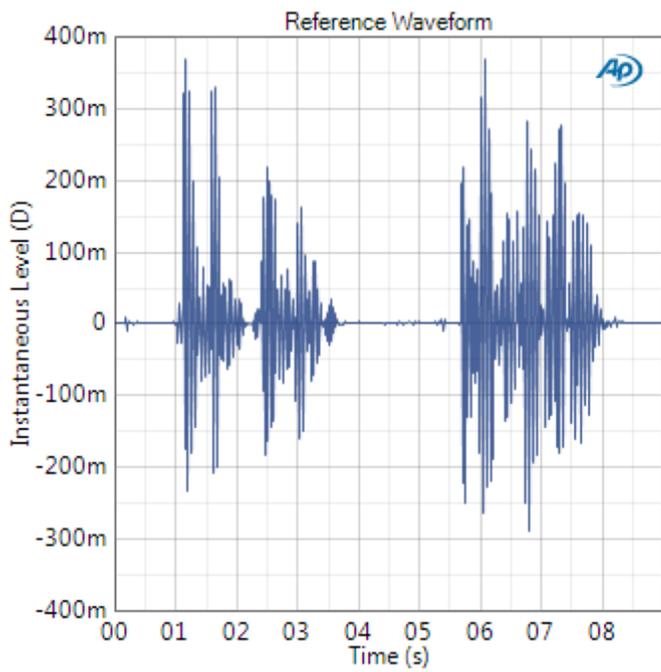
How to Setup Analog Signal Paths

Figure 45 and Figure 46 show settings for analog signal paths you might want to test in your smartphone. Recommended settings are marked with an asterisk (*) character.

TRRS = Tip Ring Sleeve jack with AC blocking cap and resistive pull-down resistor.
TRS = Stereo Tip Ring Sleeve jack with no mic sleeve (mic contact in the device is grounded on the sleeve of the jack).

Audio Path	Mode	Jack Type	Speaker Phone	Playback Audio Sink	Record Audio Source	Record Communication Mode
File to Headphone Output Jack	Playback	TRS or TRRS	Off	DTMF, Music *, System, Voice Call	NA	NA
File to Speaker Output	Playback	None	Off	Alarm, DTMF, Music *, Notification, Ring, System	NA	NA
		None	On	Voice Call *	NA	NA
File to Ear Output	Playback	None	Off	Voice Call *	NA	NA
Acoustic Microphone Input to File	Record	None or TRS	Off	NA	Camcorder, Default, Mic *, Voice Communication, Voice Recognition	Any
Electrical Microphone Input to File	Record	TRRS	Off	NA	Default *, Mic *, Voice Communication, Voice Recognition	Any

Figure 45. Table of Settings for Analog Audio Playback and Record Signal Paths

Audio Path	Notes	Mode	Jack Type	Speaker Phone	Loopback Audio Sink	Loopback Audio Source	Loopback Communication Mode
Acoustic Microphone Input to Headphone Output Jack		Loopback	TRS	Off	Music *, System, Voice Call	Camcorder, Default, Mic *, Voice Communication, Voice Recognition	Communication, Normal *, Ringtone
Acoustic Microphone Input to Speaker Output	Beware of acoustic feedback from speaker to mic	Loopback	None	On	All Settings	Camcorder, Default, Mic *, Voice Communication, Voice Recognition	Communication, Normal *, Ringtone
Electrical Microphone Input Jack to Headphone Output Jack		Loopback	TRRS	Off	DTMF, Music *, System, Voice Call	Default, Mic *, Voice Communication, Voice Recognition	Call, Communication, Normal *, Ringtone
Electrical Microphone Input Jack to Speaker Output		Loopback	TRRS	On or Off	Alarm, Notification, Ring	Default, Mic, Voice Recognition	Normal, Ringtone
				On	Voice Call *	Default, Mic *, Voice Recognition	Normal *, Ringtone

Figure 46. Table of Settings for Analog Audio Loopback Signal Paths

Test Automation with adb.exe Commands

The smartphone app is fully programmable through a USB cable using adb.exe, a component of the Android SDK. The adb.exe utility software provides the communication path between the device and the PC. Commands may be sent from the command line utility in Windows or from APx project measurement sequence steps. Each of these commands performs the same functions as buttons and menus visible in the Smartphone Audio Test user interface.

To see how these may be used, open the "automated" APx projects and inspect the sequence steps for each measurement in the sequence navigator. These sequence steps use the **Run External Program Step** method to invoke adb.exe with command line arguments. The argument for the adb.exe command is provided in the "Arguments:" parameter. The example illustrated below transfers a file recorded on the device to a folder on your Windows PC using an **adb pull** command.

This requires proper entry of the adb.exe path in the Windows PATH environment variable as noted on page 3.
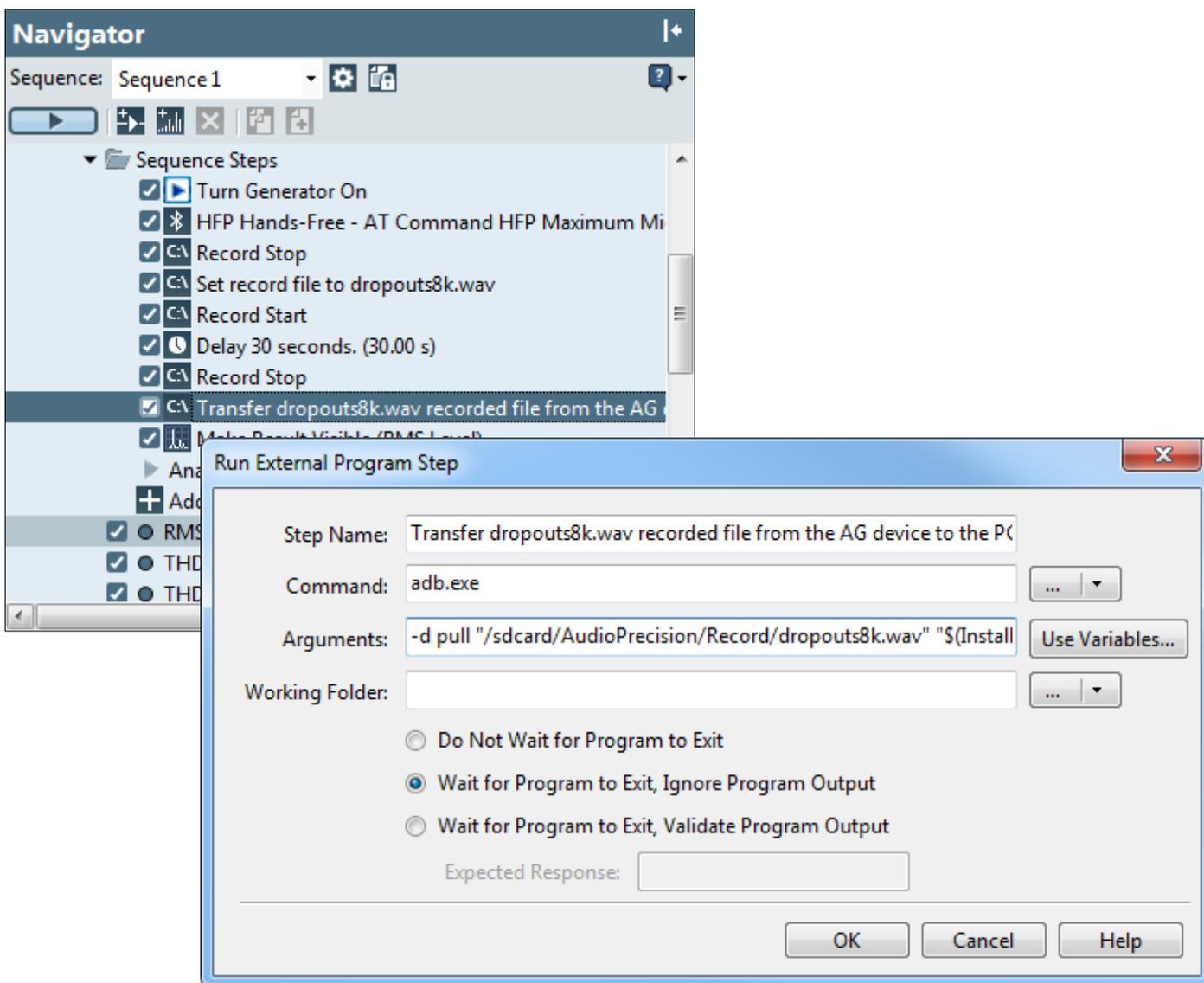
Figure 47. Invoking adb.exe from a Program sequence step.

adb Remote Commands

The Remote Command Table below lists remote commands that may be sent to the smartphone app from a PC. The commands are adb.exe invocations with command line arguments. The Android Debug Bridge connects through the Android device's USB driver to the Android OS. USB Debugging must be enabled on the smartphone.

Response Codes

Android Debug Bridge sends commands to the Smartphone app and passes responses to the Windows operating system as null, single line, and multi-line strings.

A table entry of **NO RESPONSE** indicates a Null response. For example, the Android OS does not respond to certain adb commands, such as the Back button command:

```
adb shell input keyevent 4
```

A table entry of **DEFAULT** indicates the response to most commands, shown in Figure 47. The response substring "com.ap" indicates the Android App is the source of the response.

A Failure response code may indicate the command was not recognized. This will occur if the command was received when the app is in a state that does not support the command, such as receiving the command to show the A2DP screen when the Main screen is not displayed. It may also be caused by an error in a command string.

Response String	Status
`Broadcasting: Intent { act=com.ap.CONTROL (has extras) }` `Broadcast completed: result=0`	App not active
`Broadcasting: Intent { act=com.ap.CONTROL (has extras) }` `Broadcast completed: result=0, data="Command not found"`	Failure
`Broadcasting: Intent { act=com.ap.CONTROL (has extras) }` `Broadcast completed: result=-1`	Failure
`Broadcasting: Intent { act=com.ap.CONTROL (has extras) }` `Broadcast completed: result=1`	Success
`Broadcasting: Intent { act=com.ap.CONTROL (has extras) }` `Broadcast completed: result=4096`	Success

Figure 48. DEFAULT Response Codes

Remote Command Table

Commands are sorted by **Action**. Columns **Main**, **Analog**, **HFP**, and **A2DP** indicate app screens that must be displayed for the command to be valid. A **SUCCESS** response will occur only if a command is sent when a relevant screen is displayed.

Action	Main	Analog	HFP	A2DP	adb Command	adb Response
Close the application	●	●	●	●	`adb shell am force-stop com.ap`	**NO RESPONSE**
Deactivate the application					`adb shell pm clear com.ap`	**Success** if installed. **FAILED** if not installed.
Disable Increment Filename mode		●	●		`adb shell am broadcast -a com.ap.CONTROL -e increment_filename false`	**DEFAULT**
Enable Increment Filename mode		●	●		`adb shell am broadcast -a com.ap.CONTROL -e increment_filename true`	**DEFAULT**
Get loopback buffer size		●			`adb shell am broadcast -a com.ap.CONTROL -e get loopback_buffer`	**DEFAULT**
Get loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e get loopback_rate`	**DEFAULT**
Get record buffer size		●	●		`adb shell am broadcast -a com.ap.CONTROL -e get record_buffer`	**DEFAULT**
Get record sample rate		●	●		`adb shell am broadcast -a com.ap.CONTROL -e get record_rate`	**DEFAULT**

Install the application					`adb install <filename>`
					
					Filenames with space characters must be quoted. Example:
					
					`adb install " C:\Program Files (x86)\Audio Precision\APx500 Utilities\Smartphone App 2.1\AP_Smartphone_Audio_Test_v2.1.apk"`
						If not already installed:
						`3197 KB/s (491110 bytes in 0.150s)`
						`        pkg: /data/local/tmp/`
						`ap_smartphone_audio_test_v2.1.apk`
						`Success`
						
						If already installed:
						`2397 KB/s (491110 bytes in 0.200s)`
						`        pkg: /data/local/tmp/`
						`ap_smartphone_audio_test_v2.1.apk`
						`Failure [INSTALL_FAILED_ALREADY_EXISTS]`

Loopback in mono		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_channel stereo`	**DEFAULT**
Loopback in stereo		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_channel mono`	**DEFAULT**
Open SCO		●	●		`adb shell am broadcast -a com.ap.CONTROL -e sco open`	**DEFAULT**
Open the A2DP screen	●				`adb shell am broadcast -a com.ap.CONTROL -e activity a2dp` The Main screen must be displayed. Opens the device *Bluetooth* settings screen, then continues to run in the background while displaying the *Bluetooth* screen. Exit the *Bluetooth* screen with the Back button command.	**DEFAULT**
Open the Analog screen	●				`adb shell am broadcast -a com.ap.CONTROL -e activity analog` The Main screen must be displayed.	**DEFAULT**

Action	Main	Analog	HFP	A2DP	adb Command	adb Response
Open the HFP screen	●				`adb shell am broadcast -a com.ap.CONTROL -e activity hfp` The Main screen must be displayed. Opens the device *Bluetooth* settings screen, then continues to run in the background while displaying the *Bluetooth* screen. Exit the *Bluetooth* screen with the Back button command.	**DEFAULT**
Open the Main screen	●				`adb shell am start -n com.ap/com.ap.activities.MainActivity` Start the application with this command, then open a screen (A2DP, Analog, or HFP). Displays the Activate screen if the application is not already activated.	Starting: Intent { cmp=com.ap/.activities.MainActivity }
Play last recording		●	●	●	`adb shell am broadcast -a com.ap.CONTROL -e play last`	**DEFAULT**
Record in mono		●			`adb shell am broadcast -a com.ap.CONTROL -e record_channel mono`	**DEFAULT**
Record in stereo		●			`adb shell am broadcast -a com.ap.CONTROL -e record_channel stereo`	**DEFAULT**
Restore default settings		●	●		`adb shell am broadcast -a com.ap.CONTROL -e settings default`	**DEFAULT**
Select the Back button	●	●	●	●	`adb shell input keyevent 4`	**NO RESPONSE**
Set loopback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_sink alarm`	**DEFAULT**
Set loopback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_sink dtmf`	**DEFAULT**
Set loopback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_sink music`	**DEFAULT**
Set loopback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_sink notification`	**DEFAULT**
Set loopback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_sink ring`	**DEFAULT**
Set loopback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_sink system`	**DEFAULT**
Set loopback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_sink voice_call`	**DEFAULT**
Set loopback audio source		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_source default`	**DEFAULT**
Set loopback audio source		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_source mic`	**DEFAULT**
Set loopback audio source		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_source voice_call`	**DEFAULT**
Set loopback audio source		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_source voice_communication`	**DEFAULT**
Set loopback audio source		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_source voice_downlink`	**DEFAULT**
Set loopback audio source		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_source voice_recognition`	**DEFAULT**
Set loopback audio source		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_source voice_uplink`	**DEFAULT**
Set loopback buffer size		●			`adb shell am broadcast -a com.ap.CONTROL --ei loopback_buffer 768`	**DEFAULT**
Set loopback communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_mode call`	**DEFAULT**
Set loopback communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_mode communication`	**DEFAULT**
Set loopback communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_mode normal`	**DEFAULT**

Action	Main	Analog	HFP	A2DP	adb Command	adb Response
Set loopback communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_mode ringtone`	**DEFAULT**
Set loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_rate 11025`	**DEFAULT**
Set loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_rate 16000`	**DEFAULT**
Set loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_rate 22050`	**DEFAULT**
Set loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_rate 32000`	**DEFAULT**
Set loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_rate 44100`	**DEFAULT**
Set loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_rate 48000`	**DEFAULT**
Set loopback sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback_rate 8000`	**DEFAULT**
Set playback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e record_sink alarm`	**DEFAULT**
Set playback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e record_sink dtmf`	**DEFAULT**
Set playback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e record_sink music`	**DEFAULT**
Set playback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e record_sink notification`	**DEFAULT**
Set playback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e record_sink ring`	**DEFAULT**
Set playback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e record_sink system`	**DEFAULT**
Set playback audio sink		●			`adb shell am broadcast -a com.ap.CONTROL -e record_sink voice_call`	**DEFAULT**
Set playback filename		●	●	●	`adb shell am broadcast -a com.ap.CONTROL -e setPlayFile /sdcard/AudioPrecision/Record/testfile.wav` File paths and names with space characters must be enclosed in quotes.	**DEFAULT**
Set record audio source		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_source default`	**DEFAULT**
Set record audio source		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_source mic`	**DEFAULT**
Set record audio source		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_source voice_call`	**DEFAULT**
Set record audio source		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_source voice_communication`	**DEFAULT**
Set record audio source		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_source voice_downlink`	**DEFAULT**
Set record audio source		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_source voice_recognition`	**DEFAULT**
Set record audio source		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_source voice_uplink`	**DEFAULT**
Set record buffer size		●	●		`adb shell am broadcast -a com.ap.CONTROL --ei record_buffer 768`	**DEFAULT**
Set record communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e record_mode call`	**DEFAULT**
Set record communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e record_mode communication`	**DEFAULT**
Set record communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e record_mode normal`	**DEFAULT**
Set record communication mode		●			`adb shell am broadcast -a com.ap.CONTROL -e record_mode ringtone`	**DEFAULT**

Action	Main	Analog	HFP	A2DP	adb Command	adb Response
Set record filename		●	●		`adb shell am broadcast -a com.ap.CONTROL -e setRecordFile /sdcard/AudioPrecision/Record/testfile.wav` File paths and names with space characters must be enclosed in quotes.	**DEFAULT**
Set record sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e record_rate 11025`	**DEFAULT**
Set record sample rate		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_rate 16000`	**DEFAULT**
Set record sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e record_rate 22050`	**DEFAULT**
Set record sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e record_rate 32000`	**DEFAULT**
Set record sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e record_rate 44100`	**DEFAULT**
Set record sample rate		●			`adb shell am broadcast -a com.ap.CONTROL -e record_rate 48000`	**DEFAULT**
Set record sample rate		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record_rate 8000`	**DEFAULT**
Start loopback		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback start`	**DEFAULT**
Start playback		●	●	●	`adb shell am broadcast -a com.ap.CONTROL -e play start`	**DEFAULT**
Start record		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record start`	**DEFAULT**
Stop loopback		●			`adb shell am broadcast -a com.ap.CONTROL -e loopback stop`	**DEFAULT**
Stop playback		●	●	●	`adb shell am broadcast -a com.ap.CONTROL -e play stop`	**DEFAULT**
Stop record		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record stop`	**DEFAULT**
Toggle playback		●	●	●	`adb shell am broadcast -a com.ap.CONTROL -e play toggle`	**DEFAULT**
Toggle record		●	●		`adb shell am broadcast -a com.ap.CONTROL -e record toggle`	**DEFAULT**
Toggle speaker phone		●			`adb shell am broadcast -a com.ap.CONTROL -e speaker toggle`	**DEFAULT**
Volume Down (decrement 1 click)		●		●	`adb shell am broadcast -a com.ap.CONTROL -e volume down`	**DEFAULT**
Volume Maximum		●		●	`adb shell am broadcast -a com.ap.CONTROL -e volume max`	**DEFAULT**
Volume Minimum		●		●	`adb shell am broadcast -a com.ap.CONTROL -e volume min`	**DEFAULT**
Volume Up (increment 1 click)		●		●	`adb shell am broadcast -a com.ap.CONTROL -e volume up`	**DEFAULT**

Smartphone Testing with APx Projects

This technote provides APx v4.2 project files for testing analog and *Bluetooth* signal paths in the Android device. The project files are installed on your PC as part of the AP Smartphone Audio Test App package, typically in a location as shown below. We recommend using the automated projects.

Manual projects are not automated with adb. These projects provide operator prompts to help you synchronize your testing with manual operation of the Smartphone app menus. Follow the prompts regarding waveform file Play/Stop and Record/Stop to assure synchronization with the APX measurements.

Android provides a USB connection to a Windows PC using Media Transfer Protocol (MTP). When files are written to the Android file system they should be visible in Windows File Explorer when the Android device is connected with USB to the Windows PC. However, because of a well-documented bug in Android, names of new files recorded on the Android device do not refresh the file list as seen within Windows File Explorer. When this happens the Android device must be restarted. Windows File Explorer may then be used access those files and copy them to the Windows PC.

C:\Users*username*\Documents\
    Audio Precision\
        Smartphone App\
            APX Analog Project Files\
                Analog Headphone Jack, Automated.approjx
                Analog Headphone Jack, Manual.approjx
                Analog Jack Input to Output Loopback, Automated.approjx
                Analog Microphone Jack, Automated.approjx
                Analog Microphone Jack, Manual.approjx
                Analog Microphone Jack, Loopback Find Max Input, Automated.approjx
            APX Bluetooth A2DP Project Files\
                Bluetooth A2DP, Automated.approjx
                Bluetooth A2DP, Manual.approjx
            APX Bluetooth HFP Project Files\
                Bluetooth HFP Narrowband Downlink, Automated.approjx
                Bluetooth HFP Narrowband Downlink, Manual.approjx
                Bluetooth HFP Narrowband Uplink, Automated.approjx
                Bluetooth HFP Narrowband Uplink, Manual.approjx
                Bluetooth HFP Wideband Downlink, Automated.approjx
                Bluetooth HFP Wideband Downlink, Manual.approjx
                Bluetooth HFP Wideband Uplink, Automated.approjx
                Bluetooth HFP Wideband Uplink, Manual.approjx

The chart below describes each of the projects.

APx Project Files (adb AUTOMATED)	Description
Analog Headphone Jack, Automated.approjx	This project measures headphone jack output performance of a device playing audio test files encoded at 16-bit 44.1 kHz and 24-bit 48 kHz.
	It requires adb.exe to automate the AP Smartphone Audio Test application on the device.
	Measurements: Level, THD+N, THD, Noise, Distortion Product Ratio, Dynamic Range, Frequency Response, THD+N & THD versus Frequency, THD+N & THD & Level versus Level, Crosstalk, Multitone Measurements (Total Distortion, Total Level, Noise, SNR, Frequency Response, Crosstalk, Inter-channel Phase, DC Offset, FFT Spectrum).

APx Project Files (adb AUTOMATED)	Description
Analog Jack Input to Output Loopback, Automated.approjx	This project measures the analog loopback path audio performance between the microphone jack input and the headphone jack output. It requires adb.exe to automate the AP Smartphone Audio Test application on the Android device. Measurements: Reference Level for 1% THD+N distortion at the output (microphone maximum input level at clipping), Stepped Level Sweep, Level and Gain, THD+N, Continuous Sweep, Frequency, Signal Analyzer (1 kHz sine wave spectra), Measurement Recorder, Multitone Analyzer, Stepped Frequency Sweep.
Analog Microphone Jack, Loopback Find Max Input, Automated.approjx	This project provides measurement information to determine the maximum input to a microphone input jack on a device, in order to set proper microphone input levels for subsequent audio quality tests. This project technique assumes the maximum microphone input signal level causes clipping at the input and not at the headphone jack output. It requires adb.exe to automate the AP Smartphone Audio Test application on the device. Measurements: Level and Gain, Stepped Level Sweep, THD+N, Signal Analyzer, Multitone Analyzer.
Analog Microphone Jack, Wideband and Narrowband, Automated.approjx	This project measures microphone jack performance of a device by stimulating the input with an analog signal and analyzing recorded files. Tests are performed at 8 kHz and 48 kHz recorded sample rates. The maximum microphone input level that causes 1% THD+N distortion must be known and entered as the dBrG reference level. Run the "Analog Microphone Jack, Loopback Find Max Input, Automated.approjx" project to determine this dBrG level. It requires adb.exe to automate the AP Smartphone Audio Test application on the device. Measurements: Level, THD+N, THD, Noise, Distortion Product Ratio, Frequency, Stepped Frequency Sweep (Frequency Response, THD+N & THD versus Frequency), Dynamic Range, Multitone Measurements (Total Distortion, Total Level, Noise, SNR, Frequency Response, DC Offset, FFT Spectrum).
Bluetooth A2DP, Automated.approjx 44.1 kHz & 48 kHz Sample Rates	This project measures audio quality of the *Bluetooth* A2DP downlink from the mobile device (A2DP source) to the APx audio analyzer (A2DP Sink). The APx audio analyzer receives audio test signals played on the device and analyzes them. It requires adb.exe to automate the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, Frequency and Phase versus time), Level, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, THD+N & THD versus Frequency), THD+N & THD & Level versus Level.

APx Project Files (adb AUTOMATED)	Description
Bluetooth HFP Narrowband Downlink, Automated.approjx 8 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP downlink from the Android device (the Audio Gateway, or AG) to the APx audio analyzer (HF device) using the narrowband CVSD codec operating at 8 kHz sample rate. The APx audio analyzer receives audio test signals played on the device and analyzes them. It requires adb.exe to automate the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).
Bluetooth HFP Wideband Downlink, Automated.approjx 16 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP mSBC (wideband voice) downlink from the mobile device (the Audio Gateway, or AG) to the APx audio analyzer (HF device). The APx audio analyzer receives audio test signals played on the device and analyzes them. The APx analyzer must be configured with the wideband *Bluetooth* option. Test signals are coded at 16 kHz sample rate. It requires adb.exe to automate the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).
Bluetooth HFP Narrowband Uplink, Automated.approjx 8 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP uplink from the APx audio analyzer (HF device) to the Android device (the Audio Gateway, or AG) using the narrowband CVSD codec operating at 8 kHz sample rate. The APx audio analyzer downloads the recorded files to the PC and analyzes them. It requires adb.exe to automate the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).

APx Project Files (adb AUTOMATED)	Description
Bluetooth HFP Wideband Uplink, Automated.approjx 16 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP uplink from the APx audio analyzer (HF device) to the Android device (the Audio Gateway, or AG) using the mSBC (wideband voice) codec operating at 16 kHz sample rate. The APx audio analyzer downloads the recorded files to the PC and analyzes them. It requires adb.exe to automate the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).

APx Project Files (MANUAL DOWNLOAD)	Description
Analog Headphone Jack, Manual.approjx	This project measures headphone jack output performance of a device playing audio test files. It requires manual use of the AP Smartphone Audio Test application on the device. Measurements: Level, THD+N, THD, Noise, Distortion Product Ratio, Dynamic Range, Signal Analyzer, Stepped Frequency Sweep (Frequency Response, THD+N & THD versus Frequency), THD+N & THD & Level versus Level, Crosstalk, Multitone Measurements (Total Distortion, Total Level, Noise, SNR, Frequency Response, Crosstalk, Inter-channel Phase, DC Offset, FFT Spectrum).
Analog Microphone Jack, Narrowband, Manual.approjx	This project measures microphone jack performance of a mobile device by stimulating the input with an analog signal and analyzing files recorded at 8 kHz sample rate. It requires manual use of the AP Smartphone Audio Test application on the device. The project will prompt you to manually copy the recorded files from the device folder "/sdcard/AudioPrecision/Record/" to the default PC folder "C:\Users*username*\Documents\Audio Precision\Smartphone App\Recorded\". The maximum microphone input level that causes 1% THD+N distortion must be known and entered as the dBrG reference level. Run the "Analog Microphone Jack, Loopback Find Max Input, Automated.approjx" project to determine this dBrG level. Measurements: Level, THD+N, THD, Noise, Distortion Product Ratio, Dynamic Range, Stepped Frequency Sweep (Frequency Response, THD+N & THD versus Frequency), THD+N & THD & Level versus Level, Signal Analyzer (Spectrum and Waveform), Multitone Measurements (Total Distortion, Total Level, Noise, SNR, Frequency Response, DC Offset, FFT Spectrum).

APx Project Files (MANUAL DOWNLOAD)	Description
Bluetooth A2DP, Manual.approjx 44.1 kHz & 48 kHz Sample Rates	This project measures audio quality of the *Bluetooth* A2DP downlink from the mobile device (A2DP source) to the APx audio analyzer (A2DP Sink). The APx audio analyzer receives audio test signals played on the device and analyzes them. It requires manual use of the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, Frequency and Phase versus time), Level, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, THD+N & THD versus Frequency), THD+N & THD & Level versus Level.
Bluetooth HFP Narrowband Downlink, Manual.approjx 8 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP downlink from the Android device (the Audio Gateway, or AG) to the APx audio analyzer (HF device) using the narrowband CVSD codec operating at 8 kHz sample rate. The APx audio analyzer receives audio test signals played on the device and analyzes them. It requires manual use of the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).
Bluetooth HFP Wideband Downlink, Manual.approjx 16 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP mSBC (wideband voice) downlink from the mobile device (the Audio Gateway, or AG) to the APx audio analyzer (HF device). The APx audio analyzer receives audio test signals played on the device and analyzes them. The APx analyzer must be configured with the wideband *Bluetooth* option. Test signals are coded at 16 kHz sample rate. It requires manual use of the AP Smartphone Audio Test application on the device. Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).

APx Project Files (MANUAL DOWNLOAD)	Description
Bluetooth HFP Narrowband Uplink, Manual.approjx 8 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP uplink from the APx audio analyzer (HF device) to the Android device (the Audio Gateway, or AG) using the narrowband CVSD codec operating at 8 kHz sample rate. The APx audio analyzer downloads the recorded files to the PC and analyzes them. It requires manual use of the AP Smartphone Audio Test application on the device. The project will prompt you to manually copy the recorded files from the device folder "/sdcard/AudioPrecision/Record/" to the PC folder "C:\Users*username*\Documents\Audio Precision\Smartphone App\Recorded\". Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).
Bluetooth HFP Wideband Uplink, Manual.approjx 16 kHz Sample Rate	This project measures audio quality of the *Bluetooth* HFP uplink from the APx audio analyzer (HF device) to the Android device (the Audio Gateway, or AG) using the mSBC (wideband voice) codec operating at 16 kHz sample rate. The APx audio analyzer downloads the recorded files to the PC and analyzes them. It requires manual use of the AP Smartphone Audio Test application on the device. The project will prompt you to manually copy the recorded files from the device folder "/sdcard/AudioPrecision/Record/" to the PC folder "C:\Users*username*\Documents\Audio Precision\Smartphone App\Recorded\". Measurements: Measurement Recorder Dropouts (Level, THD+N, THD, SINAD, and Frequency versus time), Level, SINAD, THD+N, THD, Noise, Distortion Product Ratio, Signal Analyzer (Spectrum and Waveform), Stepped Frequency Sweep (Frequency Response, SINAD, THD+N, and THD versus Frequency, THD+N & THD & Level versus Level, PESQ and POLQA scores (Female Voice and Male Voice MOS, PESQ, POLQA, MOS vs. Time, Acquired Waveform and Reference Waveform).

Smartphone Audio Test Signals

Audio test signals for downlink (playback) testing on a device are provided at four sample rates: 8 kHz, 16 kHz, 44.1 kHz, and 48 kHz. Files are linear PCM encoded 16 bit or 24 bit resolution. The manual APx project files provided in Technote 120 provide prompts to manually play these files on the device with the AP Smartphone Audio Test application.

Description	Level dBFS	Duration Seconds	Test File
HFP standard bandwidth downlink tests at 8 kHz sample rate (16 bit, Mono).			
1 kHz sine wave	−20	10	−20_Ref_1k_8k.wav
400 Hz sine wave	−6	180	−6_Ref_400_180S_8k.wav
	−12	180	−12_Ref_400_180S_8k.wav
	−20	9	−20_Ref_400_8k.wav
	−20	180	−20_Ref_400_180S_8k.wav
Sine wave 45 tone logarithmic stepped frequency sweep beginning with a 5 second 1 kHz sine wave pilot tone, then 3 second third octave interval steps from 20 Hz to 317 Hz, followed by sixth octave interval steps to 3557 Hz. For stepped frequency sweep measurements.	−20	140	−20_FreqSwp_45_20−3557_8k.wav
Stepped level sinewave sweep beginning with a 5 second interval −20 dBFS 1 kHz sine wave pilot tone, followed by 3 second intervals of 400 Hz sine waves at decreasing amplitudes from 0 dBFS to −90 dBFS, followed by 7 seconds of silence. For gain linearity measurements.	Variable	69	LevelSwp_8k.wav
Female voice speaking three phrases for MOS measurements with PESQ.	0	8.858	0_PESQ_F1S1_8k.wav
Male voice speaking three phrases for MOS measurements with PESQ.	0	8.057	0_PESQ_M1S1_8k.wav
Female voice speaking three phrases for MOS measurements with PESQ.	−6	8.858	−6_PESQ_F1S1_8k.wav
Male voice speaking three phrases for MOS measurements with PESQ.	−6	8.057	−6_PESQ_M1S1_8k.wav
Female voice speaking two phrases for MOS measurements with POLQA.	0	9.000	0_POLQA_F1S4_8k.wav
Male voice speaking two phrases for MOS measurements with POLQA.	0	9.000	0_POLQA_M1S2_8k.wav

Description	Level dBFS	Duration Seconds	Test File
HFP high bandwidth downlink tests at 16 kHz sample rate (16 bit, Mono).			
1 kHz sine wave	0	30	0_Ref_1k_16k.wav
	−6	30	−6_Ref_1k_16k.wav
	−20	30	−20_Ref_1k_16k.wav
400 Hz sine wave	0	180	0_Ref_400_180S_16k.wav
	−6	180	−6_Ref_400_180S_16k.wav
	−12	180	−12_Ref_400_180S_16k.wav
	−20	29.998	−20_Ref_400_16k.wav
	−20	180	−20_Ref_400_180S_16k.wav
Sine wave 28 tone logarithmic stepped frequency sweep beginning with a 5 second 997 Hz sine wave pilot tone, then 3 second third octave interval steps from 20 Hz to 6300 Hz, followed by sixth octave interval steps to 7500 Hz. For stepped frequency sweep measurements.	0	89	0_FreqSwp_28_20−7500_16k.wav
	−20	89	−20_FreqSwp_28_20−7500_16k.wav
Stepped level sinewave sweep beginning with a 5 second interval −20 dBFS 1 kHz sine wave pilot tone, followed by 3 second intervals of 400 Hz sine waves at decreasing amplitudes from 0 dBFS to −90 dBFS, followed by 7 seconds of silence. For gain linearity measurements.	Variable	69	LevelSwp16k.wav
Female voice speaking three phrases for MOS measurements with PESQ.	0	8.858	0_PESQ_F1S1_16k.wav
Male voice speaking three phrases for MOS measurements with PESQ.	0	8.056	0_PESQ_M1S1_16k.wav
Female voice speaking three phrases for MOS measurements with PESQ.	−6	8.858	−6_PESQ_F1S1_16k.wav
Male voice speaking three phrases for MOS measurements with PESQ.	−6	8.056	−6_PESQ_M1S1_16k.wav
Female voice speaking two phrases for MOS measurements with POLQA.	0	8.000	0_POLQA_F1S4_16k.wav
Male voice speaking two phrases for MOS measurements with POLQA.	0	8.000	0_POLQA_M1S2_16k.wav

Description	Level dBFS	Duration Seconds	Test File
Analog and A2DP tests at 44.1 kHz sample rate (16 bit).			
0 dBFS 10 kHz sine wave in Left channel. Silence in Right channel.	0	30	0_Crosstalk_Left_Only_10kHz_44k.wav
0 dBFS 10 kHz sine wave in Right channel. Silence in Left channel.	0	30	0_Crosstalk_Right_Only_10kHz_44k.wav
997 Hz sine wave	0	30	0_Ref_997_44k.wav
	−20	30	−20_Ref_997_44k.wav
	−20	180	−20_Ref_997_180S_44k.wav
Sine wave 31 tone logarithmic stepped frequency sweep beginning with a 3 second 997 Hz sine wave pilot tone, then 3 second third octave interval steps from 20 Hz to 20 kHz. For stepped frequency sweep measurements.	0	96	0_FreqSwp_31_20−20k_44k.wav
	−1	96	−1_FreqSwp_31_20−20k_44k.wav
	−3	96	−3_FreqSwp_31_20−20k_44k.wav
	−6	96	−6_FreqSwp_31_20−20k_44k.wav
	−20	96	−20_FreqSwp_31_20−20k_44k.wav
Sine wave 61 tone logarithmic stepped frequency sweep beginning with a 5 second 997 Hz sine wave pilot tone, then 3 second 1/6 octave interval steps from 20 Hz to 20 kHz. For stepped frequency sweep measurements.	−20	188	−20_Slow_Freqswp_61_20−20k_44k.wav
997 Hz sine wave, 7 repetitions of 2 seconds at −60 dBFS and 2 seconds at 0 dBFS, followed by 2 seconds at −60 dBFS. For Dynamic Range measurements with APx.	Variable	30	DNR_1kHz_44k.wav
Stepped level sinewave sweep beginning with a 5 second interval −20 dBFS 1 kHz sine wave pilot tone, followed by 3 second intervals of 400 Hz sine waves at decreasing amplitudes from 0 dBFS to −90 dBFS, followed by 6 seconds of silence. For gain linearity measurements.	Variable	68	LevelSwp_44k.wav
A stereo Multitone signal with 31 identical sine waves in Left and Right channels at third octave frequencies from 20 Hz to 20 kHz, a crest factor of 3.604, no phase shifts, and length of 43904 samples. The file is an integer number of repetitions.	0	9.956	0_Mult_31_20−20k_44k.wav
	−1	9.956	−1_Mult_31_20−20k_44k.wav
	−3	9.956	−3_Mult_31_20−20k_44k.wav
	−6	9.956	−6_Mult_31_20−20k_44k.wav
	−20	9.956	−20_Mult_31_20−20k_44k.wav

Description	Level dBFS	Duration Seconds	Test File
Analog and A2DP tests at 44.1 kHz sample rate (16 bit)			
A stereo 31 tone multitone signal with 25 different sine waves in Left and Right at third octave frequencies. Left Channel: 20 Hz to 20.1496 kHz, 3.649 crest factor Right Channel: 20 Hz to 19.8502 kHz, 3.733 crest factor Left & Right Channels: randomized phase shifts to minimize crest factor, length of 43904 samples. The file is an integer number of repetitions. Provides additional simultaneous L to R and R to L crosstalk measurements from 80 Hz to 20 kHz.	0	9.956	0_Mult_Xtalk_31_20−20k_44k.wav
	−1	9.956	−1_Mult_Xtalk_31_20−20k_44k.wav
	−3	9.956	−3_Mult_Xtalk_31_20−20k_44k.wav
	−6	9.956	−6_Mult_Xtalk_31_20−20k_44k.wav
	−20	9.956	−20_Mult_Xtalk_31_20−20k_44k.wav

Description	Level dBFS	Duration Seconds	Test File
Analog and A2DP tests at 48 kHz sample rate (16 bit)			
0 dBFS 10 kHz sine wave in Left channel. Silence in Right channel.	0	30	0_Crosstalk_Left_Only_10kHz_48k.wav
0 dBFS 10 kHz sine wave in Right channel. Silence in Left channel.	0	30	0_Crosstalk_Right_Only_10kHz_48k.wav
997 Hz sine wave	0	30	0_Ref_997_48k.wav
	−20	30	−20_Ref_997_48k.wav
	−20	180	−20_Ref_997_180S_48k.wav
Sine wave 31 tone logarithmic stepped frequency sweep beginning with a 3 second 997 Hz sine wave pilot tone, then 3 second third octave interval steps from 20 Hz to 20 kHz. For stepped frequency sweep measurements.	0	96	0_FreqSwp_31_20−20k_48k.wav
	−1	96	−1_FreqSwp_31_20−20k_48k.wav
	−3	96	−3_FreqSwp_31_20−20k_48k.wav
	−6	96	−6_FreqSwp_31_20−20k_48k.wav
	−20	96	−20_FreqSwp_31_20−20k_48k.wav
Stereo sine wave 61 tone logarithmic stepped frequency sweep beginning with a 5 second 997 Hz sine wave pilot tone, then 3 second 1/6 octave interval steps from 20 Hz to 20 kHz. For stepped frequency sweep measurements.	−20	188	−20_Slow_Freqswp_61_20−20k_48k.wav
997 Hz sine wave, 7 repetitions of 2 seconds at −60 dBFS and 2 seconds at 0 dBFS, followed by 2 seconds at −60 dBFS. For Dynamic Range measurements with APx.	Variable	30	DNR_1kHz_48k.wav

Description	Level dBFS	Duration Seconds	Test File
Analog and A2DP tests at 48 kHz sample rate (16 bit)			
Stepped level sinewave sweep beginning with a 5 second interval −20 dBFS 1 kHz sine wave pilot tone, followed by 3 second intervals of 400 Hz sine waves at decreasing amplitudes from 0 dBFS to −90 dBFS, followed by 6 seconds of silence. For gain linearity measurements.	Variable	64	LevelSwp_48k.wav
A stereo Multitone signal with 31 identical sine waves in Left and Right channels at third octave frequencies from 20 Hz to 20 kHz, a crest factor of 3.821, phase shifts to minimize crest factor, and length of 96000 samples. The file is an integer number of repetitions.	0	10.000	0_Mult_31_20−20k_48k.wav
	−1	10.000	−1_Mult_31_20−20k_48k.wav
	−3	10.000	−3_Mult_31_20−20k_48k.wav
	−6	10.000	−6_Mult_31_20−20k_48k.wav
	−20	10.000	−20_Mult_31_20−20k_48k.wav
A stereo 31 tone multitone signal with 25 different sine waves in Left and Right at third octave frequencies. Left Channel: 20 Hz to 20.150 kHz, 3.943 crest factor. Right Channel: 20 Hz to 19.850 kHz, 3.849 crest factor. Left & Right Channels: randomized phase shifts to minimize crest factor, length of 96000 samples. The file is an integer number of repetitions. Provides additional simultaneous L to R and R to L crosstalk measurements from 101.000 Hz to 20.150 kHz.	0	10.000	0_Mult_Xtalk_31_20−20k_48k.wav
	−1	10.000	−1_Mult_Xtalk_31_20−20k_48k.wav
	−3	10.000	−3_Mult_Xtalk_31_20−20k_48k.wav
	−6	10.000	−6_Mult_Xtalk_31_20−20k_48k.wav
	−20	10.000	−20_Mult_Xtalk_31_20−20k_48k.wav

Description	Level dBFS	Duration Seconds	Test File
High Resolution Audio Performance Analog tests at 48 kHz sample rate (24 bit, Stereo).			
0 dBFS 10 kHz sine wave in Left channel. Silence in Right channel. Stereo	0	30	0_Crosstalk_Left_Only_10kHz_24bit_48k.wav
0 dBFS 10 kHz sine wave in Right channel. Silence in Left channel. Stereo	0	30	0_Crosstalk_Right_Only_10kHz_24bit_48k.wav
Stereo 997 Hz sine wave	0	30	0_Ref_997_24bit_48k.wav
	−20	30	−20_Ref_997_24bit_48k.wav
	−20	180	−20_Ref_997_180S_24bit_48k.wav
Stereo sine wave 31 tone logarithmic stepped frequency sweep beginning with a 3 second 997 Hz sine wave pilot tone, then 3 second third octave interval steps from 20 Hz to 20 kHz. For stepped frequency sweep measurements.	0	96	0_FreqSwp_31_20−20k_24bit_48k.wav
	−1	96	−1_FreqSwp_31_20−20k_24bit_48k.wav
	−3	96	−3_FreqSwp_31_20−20k_24bit_48k.wav
	−6	96	−6_FreqSwp_31_20−20k_24bit_48k.wav
	−20	96	−20_FreqSwp_31_20−20k_24bit_48k.wav
Stereo sine wave 61 tone logarithmic stepped frequency sweep beginning with a 5 second 997 Hz sine wave pilot tone, then 3 second 1/6 octave interval steps from 20 Hz to 20 kHz. For stepped frequency sweep measurements.	0	188	0_Slow_Freqswp_61_20−20k_24bit_48k.wav
	−20	188	−20_Slow_Freqswp_61_20−20k_24bit_48k.wav
Stereo 997 Hz sine wave, 7 repetitions of 2 seconds at −60 dBFS and 2 seconds at 0 dBFS, followed by 2 seconds at −60 dBFS. For Dynamic Range measurements with APx.	Variable	30	DNR_1kHz_24bit_48k.wav
Stereo stepped level sinewave sweep beginning with a 5 second interval −20 dBFS 1 kHz sine wave pilot tone, followed by 3 second intervals of 400 Hz sine waves at decreasing 5 dB amplitude steps from 0 dBFS to −90 dBFS, followed by 6 seconds of silence. For gain linearity measurements.	Variable	64	LevelSwp_24bit_48k.wav

Description	Level dBFS	Duration Seconds	Test File
High Resolution Audio Performance Analog tests at 48 kHz sample rate (24 bit).			
A stereo Multitone signal with 31 identical sine waves in Left and Right channels at third octave frequencies from 20 Hz to 20 kHz, a crest factor of 3.821, phase shifts to minimize crest factor, and length of 96000 samples. The file is an integer number of repetitions.	0	10.000	0_Mult_31_20–20k_24bit_48k.wav
	−1	10.000	−1_Mult_31_20–20k_24bit_48k.wav
	−3	10.000	−3_Mult_31_20–20k_24bit_48k.wav
	−6	10.000	−6_Mult_31_20–20k_24bit_48k.wav
	−20	10.000	−20_Mult_31_20–20k_24bit_48k.wav
A stereo 31 tone multitone signal with 25 different sine waves in Left and Right at third octave frequencies. Left Channel: 20 Hz to 20.150 kHz, 3.943 crest factor Right Channel: 20 Hz to 19.850 kHz, 3.849 crest factor Left & Right Channels: randomized phase shifts to minimize crest factor, length of 96000 samples. The file is an integer number of repetitions. Provides additional simultaneous L to R and R to L crosstalk measurements from 101.000 Hz to 20.150 kHz.	0	10.000	0_Mult_Xtalk_31_20–20k_24bit_48k.wav
	−1	10.000	−1_Mult_Xtalk_31_20–20k_24bit_48k.wav
	−3	10.000	−3_Mult_Xtalk_31_20–20k_24bit_48k.wav
	−6	10.000	−6_Mult_Xtalk_31_20–20k_24bit_48k.wav
	−20	10.000	−20_Mult_Xtalk_31_20–20k_24bit_48k.wav