WICED Studio

WICED™ APOLLO Wireless Audio

Associated Part Family: BT CYW2070x

Doc. No.: 002-19004 Rev. *A

Cypress Semiconductor

198 Champion Court

San Jose, CA 95134-1709

www.cypress.com

Contents

About This Document **4**

Purpose and Scope 4
Audience 4
Acronyms and Abbreviations 4
IoT Resources and Technical Support 4

1 Apollo Wireless Audio Distribution System **5**

1.1 Audio Speaker Configurations 5
1.2 Stereo/2.1 5
1.3 5.1/7.1 Surround Sound 6
1.4 UPnP Audio Rebroadcast 6
1.5 Equipment Needed 6
1.6 Building the Firmware 7
1.6.1 Build Prerequisites (Pre 5.x only) 7
NOTE: for Wiced SDK 5.0 and later, Apollo Firmware is automatically copied during build. 7
1.6.1.1 For SDK 3.6.x/3.7.x 7
1.6.1.2 For SDK 4.x 7
1.6.1.3 For SDK 5.x 8
1.6.2 Platform Build strings 9
1.6.3 Platform Build string variants 9
1.7 Configuration 10
1.7.1 Multi-Zone - Bluetooth Re-broadcaster (> 2 devices) 10
1.7.1.1 BT Source configuration 10
1.7.1.2 BT Stereo Sink Configuration(s) 11
1.7.2 Multi-Zone - Analog Re-broadcaster 12
1.7.2.1 Analog Source configuration 12
1.7.2.2 Analog Stereo Sink Configuration 13
1.7.3 Multi-Speaker 13
1.7.3.1 Sink Configuration 14
1.7.4 UPnP Audio Rebroadcast 14
1.7.4.1 Source Configuration 14
1.7.4.2 Sink Configuration 14
1.7.4.3 Using BubbleUPnP with UPnP Audio Renderer Re-broadcaster 14
1.7.4.4 Installing and running BubbleUPnP on an Android-based phone or tablet 15
1.7.5 5.1 Surround sound 20
1.7.5.1 SONY PS4 Bluray Player Test Configuration 21
1.7.5.2 Bluray 5.1 Test Content 23
1.7.5.3 5.1 Source Configuration (WAE2/3 boards only) 24
1.7.5.4 5.1 Sink Configuration 25
1.7.6 5.1 Surround Sound Sink/Speaker Configuration 26
1.7.6.1 Front Left Speaker 26
1.7.6.2 Front Right Speaker 26
1.7.6.3 Front Center Speaker (Dialogue channel) 26
1.7.6.4 Rear (Surround) Left Speaker 26
1.7.6.5 Rear (Surround) Right Speaker 26
1.7.6.6 Low Frequency Effects (LFE1) - Subwoofer Speaker 26
1.8 Available Commands 27
1.9 Extra Debugging 29
1.9.1 Build Options 29
1.10 OLED Display 29

2 iOS Apollo Audio Configuration Application **31**

2.1 Setup 31

2.2 System configuration via the iOS Apollo application 32

3 Roadmap 36

Document Revision History 37

Worldwide Sales and Design Support 38

Products 38
PSoC® Solutions 38
Cypress Developer Community 38
Technical Support 38

About This Document

Purpose and Scope

This document provides instructions to use the WICED Apollo wireless audio distribution system.

Note: This document applies to **WICED SDK 3.6.x** or higher.

Audience

This document is for software developers who are using the WICED Development System to create Wireless audio distribution systems using CYW94390x product family (e.g. CYW943907WAE3, BCM943909WCD1_3, BCM943907WAE_1, BCM943907WAE2_1, BCM943907WCD2, etc).

Acronyms and Abbreviations

In most cases, acronyms and abbreviations are defined on first use.

For a comprehensive list of acronyms and other terms used in Cypress documents, go to www.cypress.com/glossary.

IoT Resources and Technical Support

Cypress provides a wealth of data at www.cypress.com/internet-things-iot to help you to select the right IoT device for your design, and quickly and effectively integrate the device into your design. Cypress provides customer access to a wide range of information, including technical documentation, schematic diagrams, product bill of materials, PCB layout information, and software updates. Customers can acquire technical documentation and software from the Cypress Support Community website (community.cypress.com/).

1 Apollo Wireless Audio Distribution System

Apollo is the software stack for implementing a whole-home multichannel Wireless Audio distribution system. Central to the design of Apollo is the concept of a sender (transmitter) or "source" device and one or more receiver or "sink devices". The system allows a transmitter/sender or "source" to transmit music via Wi-Fi (over an 802.11 network) to one or more receivers (“sink”, Wireless speakers) allowing for scalable speaker configurations from 2 speakers (stereo) up to 5.1 (6 speakers) and later 7.1 surround sound. Core features in the Apollo wireless audio streaming system:

- 802.1AS Precision Time Protocol and Grand Master Clock (GMC) supporting 802.11v
- Reliable Multicast Streaming protocol for one or more speaker devices
- Audio Packet Loss Concealment
- Low latency

The following configurations below are supported:

Input Source	Source Device	Sink Device(s)	Configuration	Availability
File (WAV)	Gigabyte BRIX + BCM4356	2 x BCM943907WAE_1	Stereo	Contact sales
File (WAV)	Gigabyte BRIX + BCM4356	6 x BCM943907WAE_1	5.1 Surround	Contact sales
Bluetooth A2DP	BCM943907WAE_1, BCM943907WAE2_1	BCM943907WAE_1	Stereo	WICED 3.5.2
Bluetooth A2DP	BCM943907WAE_1, BCM943907WAE2_1	2 x BCM943907WAE_1	Multi-zone (Stereo)	WICED 3.5.2
Analog Stereo	BCM943907WAE_1/BCM943909WCD1_3	BCM943907WAE_1	Stereo	WICED 3.5.2
Analog Stereo	BCM943907WAE_1/BCM943909WCD1_3	2 x BCM943907WAE_1	Multi-zone (Stereo)	WICED 3.5.2
SPDIF (Optical)/BT	BCM943907WAE_2	6 x BCM943907WAE_1	Stereo, 5.1 Surround	WICED 3.6
SPDIF/BT	CYW943907WAE3	6 x CYW943907WAE3	5.1 Surround Only 2 needed for BT	WICED 5.0
Analog/BT	BCM943907WCD1, BCM943907WCD2	2 x BCM943907WCDxx	2 for WLAN rebroadcast	WICED 5.0

1.1 Audio Speaker Configurations

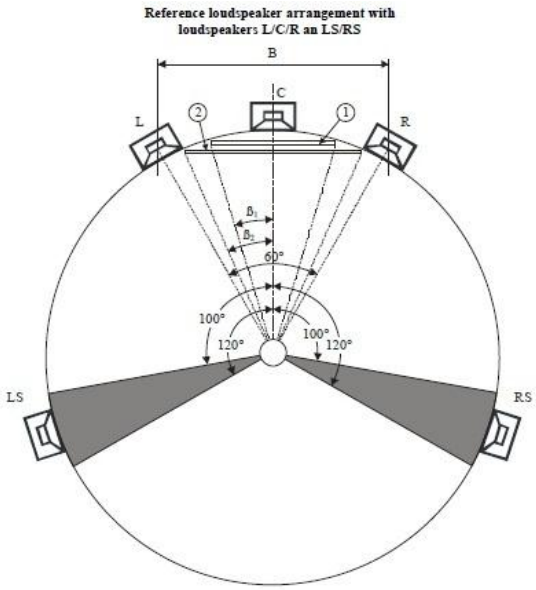
Apollo also scales and addresses the growth needs of the user as more speakers are added to the system - Stereo (2), 2.1 (3) (L/R/Bass), 6 (5.1), and larger systems may be built. Apollo supports up to 16 different speaker/channel configurations all of which may be adapted to the user’s entertainment requirements.

1.2 Stereo/2.1

The most common use case which Apollo addresses are two independent speakers (stereo) where (rather than having one receiver driving 2 speakers) each receiver renders one of the speaker pairs - a Left (L) or Right (R) channel independently. This system may be further enhanced to support a Center (C) configuration.

1.3 5.1/7.1 Surround Sound

The true benefit of Apollo comes from support for more than 2 speakers; the most relevant use case is "Surround sound" or a 5.1 system and larger. The 5.1 channel sound system has been specified in Recommendation ITU-R BS.775. The system is widely used as a part of digital broadcasting services. It enhances the directional stability of the frontal sound image and the sensation of spatial reality (ambience). The reference loudspeaker arrangement is shown below in which each loudspeaker is set at the same height as a listener's ears.



Screen 1 HDTV - Reference distance = 3 $H(2\beta_1 = 33°)$
Screen 2 = 2 $H(2\beta_2 = 48°)$
H: height of screen
B: loudspeaker base width

Loudspeaker	Horizontal angle from centre (degrees)	Height (meters)	Inclination (degrees)
C	0	1.2	0
L, R	30	1.2	0
LS, RS	100 ... 120	≥ 1.2	0 ... 15 down

1.4 UPnP Audio Rebroadcast

Apollo also supports receiving a UPnP stream from a Mobile device and rebroadcasting to multiple downstream peers. This application feature allows for extension of the rebroadcast for OTT Audio

1.5 Equipment Needed

- 2 Cypress WICED platforms of the above configurations (minimum for multi-zone Stereo)
- iPhone or Android phone supporting Bluetooth
- 2 Speakers (Amplified) or Headphones supporting standard 3.5mm Audio plug
- Power connectors
- WICED Debug board

1.6 Building the Firmware

Follow the steps below to build the firmware image for the platforms.Note that special care must be taken when using older versions of software and build/configuration steps are slightly different.

1.6.1 Build Prerequisites (Pre 5.x only)

Prior to WICED SDK 5.0, one must first install Apollo RMC/AVB WLAN firmware (for all 4390x platforms) (for all releases)

```
cd WICED-SDK-xxx/WICED-SDK/resources/firmware/43909
cp 43909B0-apollo.bin 43909B0.bin
```

NOTE: for Wiced SDK 5.0 and later, Apollo Firmware is automatically copied during build. Note also that you must do a clean and then rebuild whenever the firmware changes.

1.6.1.1 For SDK 3.6.x/3.7.x

Edit `WICED-SDK/apps/demo/apollo/wifi_config_dct.h` to specify SSID and channel; the defaults below are recommended for first time use:

```
/* This is the soft AP available for normal operation */
#define SOFT_AP_SSID "apollo"
#define SOFT_AP_CHANNEL 132
/* This is the default AP the device will connect to (as a client)*/
#define CLIENT_AP_SSID "apollo"
#define CLIENT_AP_BSS_TYPE WICED_BSS_TYPE_ADHOC
#define CLIENT_AP_SECURITY WICED_SECURITY_OPEN
#define CLIENT_AP_CHANNEL 132
#define CLIENT_AP_BAND WICED_802_11_BAND_5GHZ
```

1.6.1.2 For SDK 4.x

Edit `WICED-SDK/apps/demo/apollo/apollo_dct.h` to specify SSID and channel; the defaults below are recommended for first time use:

```
/* This is the default RMC for the device */
#define APOLLO_RMC_SSID       "apollo"
#define APOLLO_RMC_PASSPHRASE "abcd1234"
#define APOLLO_RMC_BSS_TYPE   WICED_BSS_TYPE_ADHOC
#define APOLLO_RMC_SECURITY   WICED_SECURITY_OPEN
#define APOLLO_RMC_CHANNEL    149
#define APOLLO_RMC_BAND       WICED_802_11_BAND_5GHZ
```

1.6.1.3 For SDK 5.x

Configuration is the same as SDK 4.x, however, SDK 5.0 supports either Ad-HOC or Infrastructure mode (where Infrastructure mode is the default). To revert to IBSS mode, edit the file `WICED-SDK/apps/demo/apollo/apollo_dct` and change the define below:

```
/* Default operating mode is Infrastructure mode */
#define APOLLO_RMC_BSS_TYPE    WICED_BSS_TYPE_INFRASTRUCTURE

/* This is for Ad-hoc mode (not recommended for new designs) */
/* #define APOLLO_RMC_BSS_TYPE    WICED_BSS_TYPE_ADHOC */
```

Note: supported 5Ghz DFS channels are: 132, 108

1.6.2 Platform Build strings

In the WICED IDE, create the following three build targets. Build and install on two systems (not necessarily the same Hardware platform needs to be employed – for example you may use a WAE3 board as source and WCD1/2 as Sink)

Wireless Audio Edition 1 (WAE_1)

```
demo.apollo-BCM943907WAE_1 download run
```

Wireless Audio Edition 2 (WAE2_1)

```
demo.apollo-BCM943907WAE2_1 download run
```

Wireless Connectivity Device 1 (WCD1_3)

```
demo.apollo-BCM943909WCD1_3 download run
```

Wireless Connectivity Device 2 (WCD2 - SPIL)

```
demo.apollo-BCM943907WCD2 download run
```

Wireless Connectivity Device 2 (WCD1 - Murata)

```
demo.apollo-BCM943907WCD1 download run
```

Wireless Audio Edition 3 (WAE3)

```
demo.apollo-CYW943907WAE3 download run
```

NOTE: WCD1 platforms use AKM audio output by default. You can change this with the "`config ad_tx`" option (to use for example the Wolfson codec). For example “config ad_tx 2”

NOTE: SDK 3.6.x and beyond no longer take a Platform chip version.

1.6.3 Platform Build string variants

Wireless Audio Edition 3 (WAE3) – with Dolby Digital Support (SPDIF TX Only)

```
demo.apollo-CYW943907WAE3 USE_UDC=1 download run
```

Wireless Audio Edition 3 (WAE3) – with UPnP AV Rendering support

```
demo.apollo-CYW943907WAE3 USE_UPNPAV=1 download run
```

Wireless Audio Edition 3 (WAE3) – with OLED Display Support (may be added to any build option)

```
demo.apollo-CYW943907WAE3 USE_AUDIO_DISPLAY=1download run
```

1.7 Configuration

Start by designating one system as the source device and the other as the sink device. Connect the speakers and power.

Now (with the USB serial port attached to your system for console), issue the below commands on each device:

1.7.1 Multi-Zone - Bluetooth Re-broadcaster (> 2 devices)

This system configuration will allow for a BT Audio connection to the device to be rebroadcasted (over Wi-Fi) to other receiver devices.

1.7.1.1 BT Source configuration

Wiced-SDK-3.5.x

```
> config apollo_role source
> config src_t bt
> config network_name <name>
> config network_channel <channel>
> config mac <mac>
> config bt_name <name>
> config pll 1
> config save
```

Wiced-SDK-3.7.x

```
> config apollo_role source
> config src_t bt
> config pll 1
> dct_wifi_ap_list 0 channel <channel>
> dct_wifi_ap_list 0 ssid <name>
> dct_bt_name <name>
> dct_wifi_mac <mac>
> dct_bt_mac <mac>
> config save
> reboot
```

Wiced-SDK-4.0 and later

```
> config apollo_role source
> config src_t bt
> config pll 1
> config rmc_chan <channel>
> config rmc_ssid <name>
> dct_bt_name <name>
> dct_wifi_mac <mac>
```

```
> dct_bt_mac <mac>
> config save
> reboot
```

1.7.1.2 BT Stereo Sink Configuration(s)

Wiced-SDK-3.5.x

```
> config apollo_role sink
> config network_name <name>
> config network_channel <channel>
> config speaker_channel FL FR
> config mac <mac>
> config clock 1
> config pll 1
> config save
> reboot
```

Wiced-SDK-3.7.x

```
> config apollo_role sink
> config pll 1
> config clock 1
> config speaker_channel FL FR
> dct_wifi_ap_list 0 channel <channel>
> dct_wifi_ap_list 0 ssid <name>
> dct_wifi_mac <mac>
> config save
> reboot
```

Wiced-SDK-4.0 and later

```
> config apollo_role sink
> config pll 1
> config clock 1
> config speaker_channel FL FR
> config rmc_chan <channel>
> config rmc_ssid <name>
> dct_wifi_mac <mac>
> config save
> reboot
```

NOTE: <name>, <channel> must be the same on source/sink (e.g. “Apollo-BT”, and “149”)

NOTE: the mac address <mac> should be unique on each board (you can also add enter 1 digit to change the last byte) and it is not guaranteed that every system has a MAC address programmed. For example “config mac 5a” will change the last byte of the existing MAC address.

Finally reboot each board after “config save”. Use your iPhone/Android phone and discover the device under your Bluetooth settings menu. Connect and play music. You will hear the music on both the source and sink devices.

1.7.2 Multi-Zone - Analog Re-broadcaster

This configuration allows for an analog device to be connected to the analog input port and rebroadcast (over Wi-Fi) to one or more downstream speakers. In this configuration, older analog equipment with a 3.5mm audio output jack will interface to the system.

1.7.2.1 Analog Source configuration

Wiced-SDK-3.5.x

```
> config apollo_role source
> config src_t capture
> config network_name <name>
> config network_channel <channel>
> config mac <mac>
> config save
> reboot
```

Wiced-SDK-3.7.x

```
> config apollo_role source
> config src_t capture
> config pll 1
> config save
> dct_wifi_ap_list 0 channel <channel>
> dct_wifi_ap_list 0 ssid <name>
> dct_wifi_mac <mac>
> reboot
```

Wiced-SDK-4.0 and later

```
> config apollo_role source
> config src_t capture
> config pll 1
> config rmc_chan <channel>
> config rmc_ssid <name>
> config save
> dct_wifi_mac <mac>
> reboot
```

1.7.2.2 Analog Stereo Sink Configuration

Wiced-SDK-3.5.x

```
> config apollo_role sink
> config network_name <name>
> config network_channel <channel>
> config speaker_channel FL FR
> config mac <mac>
> config clock 1
> config pll 1
> config save
> reboot
```

Wiced-SDK-3.7.x

```
> config apollo_role sink
> config speaker_channel FL FR
> config clock 1
> config pll 1
> dct_wifi_ap_list 0 channel <channel>
> dct_wifi_ap_list 0 ssid <name>
> dct_wifi_mac <mac>
> config save
> reboot
```

Wiced-SDK-4.0 and later

```
> config apollo_role sink
> config speaker_channel FL FR
> config clock 1
> config pll 1
> config rmc_chan <channel>
> config rmc_ssid <name>
> dct_wifi_mac <mac>
> config save
> reboot
```

Connect the audio input to the source device. As before, reboot the system and start playback.

1.7.3 Multi-Speaker

To support multiple speakers, each sink device is designated a channel which it would play. For a stereo system you would need 1 set of speakers on the source (playing stereo (FL/FR)) and a speaker attached to 2 more sink devices (one playing left – FL, the other playing right – FR).

1.7.3.1 Sink Configuration

```
> config speaker_channel FL FR
> config save
```

1.7.4 UPnP Audio Rebroadcast

For rebroadcasting Audio content from UPnP, the below configuration is required.

- In the configuration detailed below, the Apollo source/broadcaster is both a Soft AP and a STA
- The STA joins an home router / AP / gateway and is used to run an UPnP Audio Renderer (with the ability to pull in and play audio streams over HTTP)
- The Soft AP is used by the Apollo streaming engine to multicast uncompressed PCM samples to Apollo sinks

1.7.4.1 Source Configuration

Wiced-SDK-5.0 and later

```
> config rmc_chan 149
> config rmc_ssid Apollo-UPnPAV
> config save
> dct_wifi_mac aa:bb:cc:dd:ee:fd
# Apollo Source needs to associate with an upstream AP (home router/gateway and the likes)
> dct_wifi_ap_list 0 ssid <SSID_of_Upstream_AP>
> dct_wifi_ap_list 0 bss_type infra
> dct_wifi_ap_list 0 sec <Security_Type_on_Upstream_AP>
> dct_wifi_ap_list 0 key <Key_Or_Passphrase>
> reboot
```

1.7.4.2 Sink Configuration

Wiced-SDK-5.0 and later

```
> config apollo_role sink
> config speaker_channel FL FR
> config rmc_chan 149
> config rmc_ssid Apollo-UPnPAV
> config save
> dct_wifi_mac aa:bb:cc:dd:ee:fa
> reboot
```

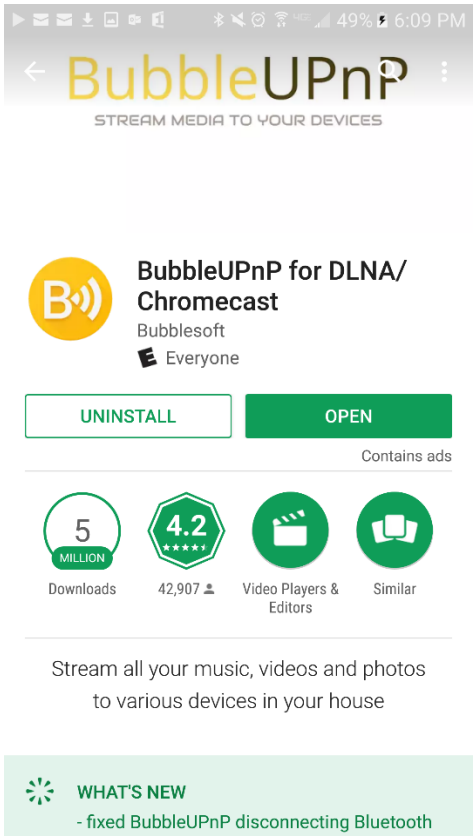
1.7.4.3 Using BubbleUPnP with UPnP Audio Renderer Re-broadcaster

BubbleUPnP is a UPnP Media Controller and an UPnP Media Server app for Android. Other UPnP Media controllers may be supported, but the below outlines a sane starting point. Using UPnP protocols, BubbleUPnP can find and recognize UPnP Media Renderers on a home network (this includes UPnP Audio Renderers). Since BubbleUPnP also doubles as a UPnP Media Server, it's possible to initiate

playback of audio media content on the phone/tablet and get it to play/render on our UPnP Audio Renderer and Re-broadcaster.

1.7.4.4 Installing and running BubbleUPnP on an Android-based phone or tablet

Launch the "Play Store" app on your Android phone or tablet and look for the BubbleUPnP application from Bubblesoft.

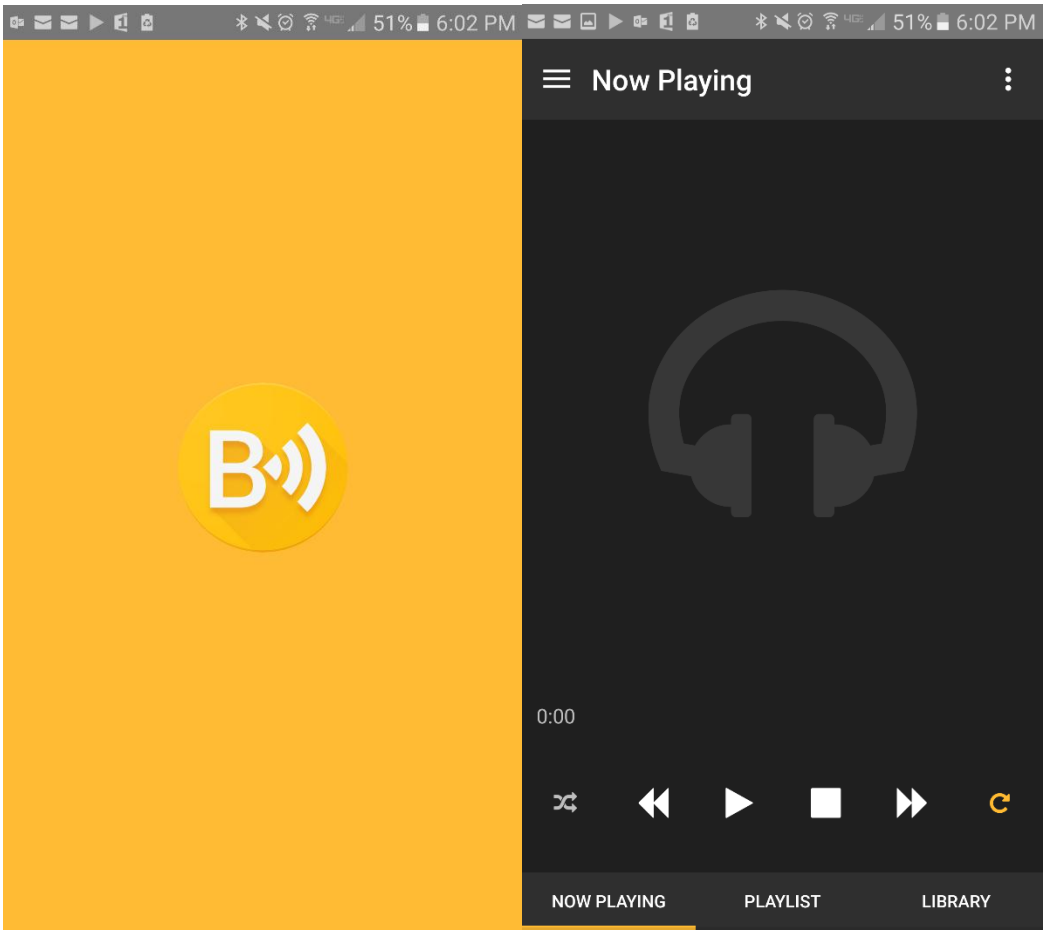


Next proceed with installation (the free version of BubbleUPnP is sufficient as it's fully functional);you may want to load up some audio content to your phone/tablet: MP3 and WAV (currently M4A/FLAC is not supported).

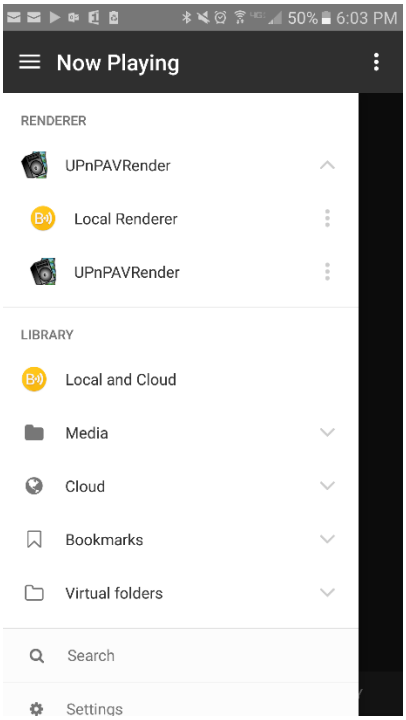
Associate the phone/tablet to the same upstream AP used by the Apollo Re-broadcaster

(see step above: "> dct_wifi_ap_list 0 ssid <SSID_of_Upstream_AP>)

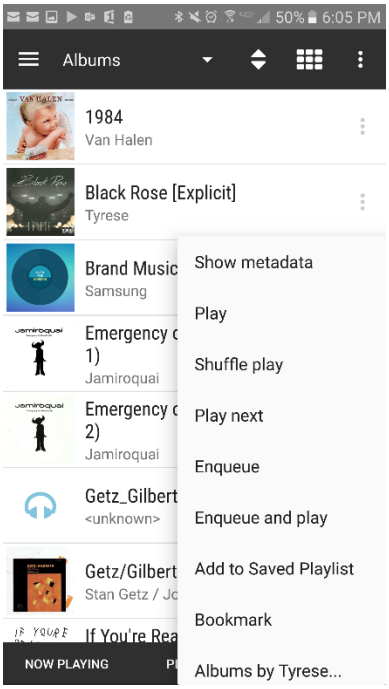
We're now ready to launch BubbleUPnP (that's the BubbleUPnP splash screen and "NOW PLAYING" tab is shown as below).



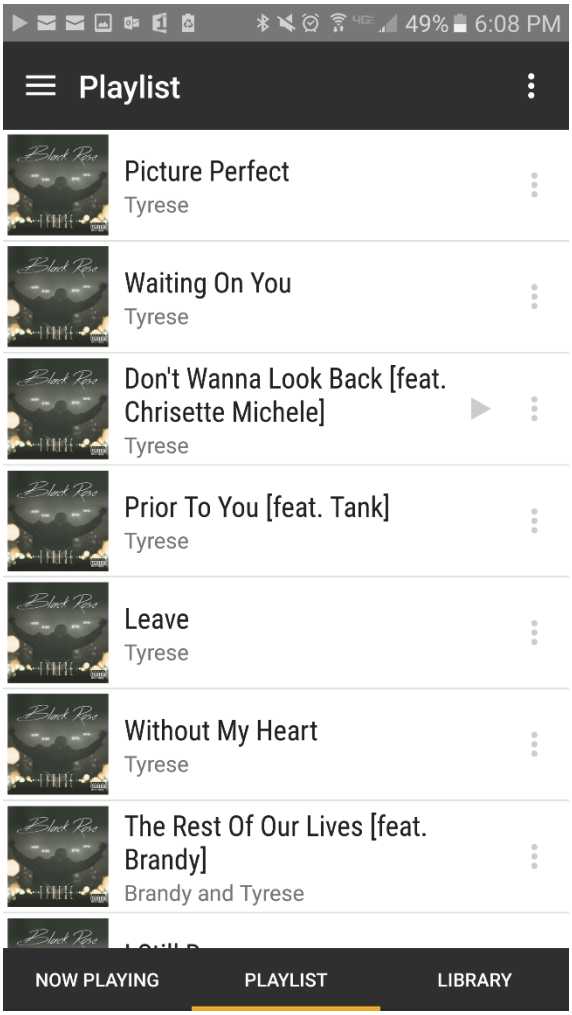
Next verify whether or not BubbleUPnP is able to find and recognize the Apollo Re-broadcaster as a Renderer. Tap/hit the area left of the "Now Playing" label to reveal the list of available Renderers. The "Local Renderer" refers to the phone/tablet itself (using whatever multimedia playback app available). A "UPnPAVRender" Renderer should be visible; please select it:



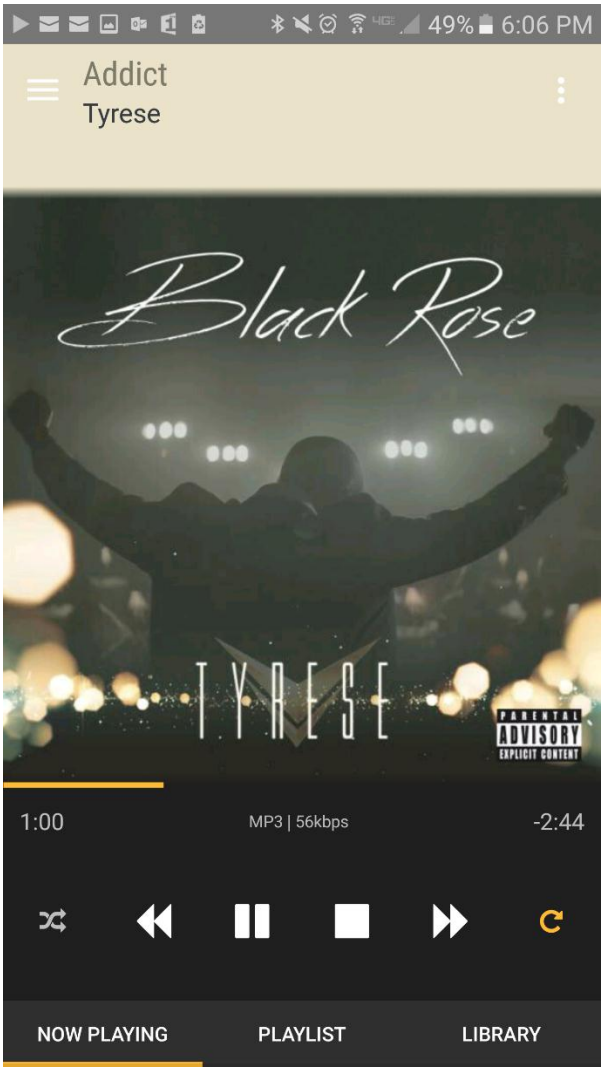
Go to the "LIBRARY" tab to initiate media playback (depending on what's available on your phone/tablet)



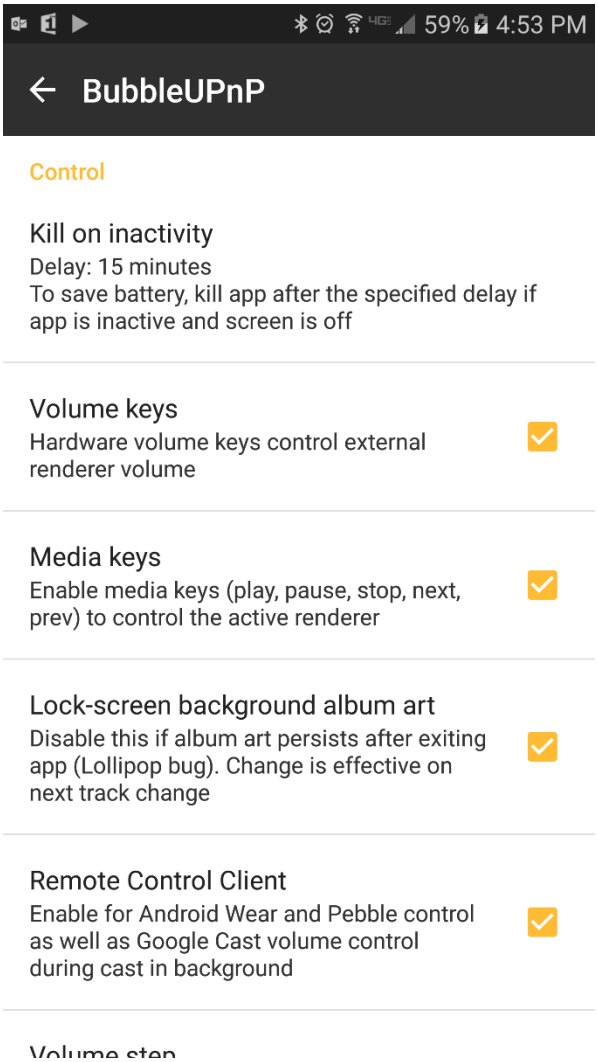
Tap/hit "Play" or "Enqueue and play" to add the selected Album or Songs to the playback list (the app will switch over to the "PLAYLIST" tab):



The "NOW PLAYING" tab will show the currently playing stream; music should be rendered on the Apollo source and all Apollo sinks



NOTE: Regarding playback volume control, make sure BubbleUPnP is configured to use the phone Volume Up/Volume Down keys (down in the "Settings/Control" menu/submenu)



1.7.5 5.1 Surround sound

The BCM943907WAE2/WAE3 platforms support a SPDIF Input port, this will allow you to connect the source device to a SPDIF transmitter and decode Dolby Digital real-time. To build for the WAE2 & 3 platforms, you need to include the Dolby Digital decoder. In addition to copying the RMC firmware (as above), you need to build the firmware for the source device differently. See instructions and build commands below:

With Dolby™ Digital Decoder

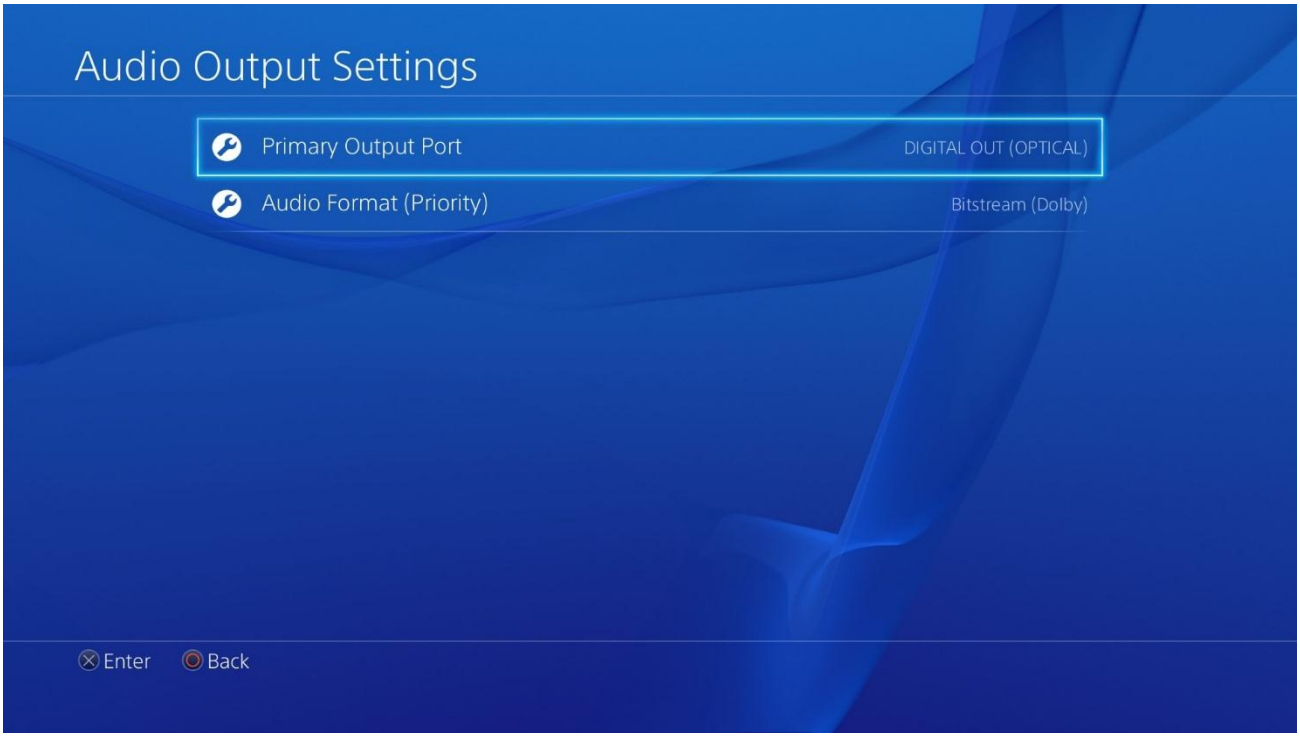
```
demo.apollo-BCM943907WAE2_1 USE_UDC=1 download run
```

```
demo.apollo-CYW943907WAE3 USE_UDC=1 download run
```

Build and FLASH your WAE2/3 platform and connect the SPDIF port to your source device.

1.7.5.1 SONY PS4 Bluray Player Test Configuration

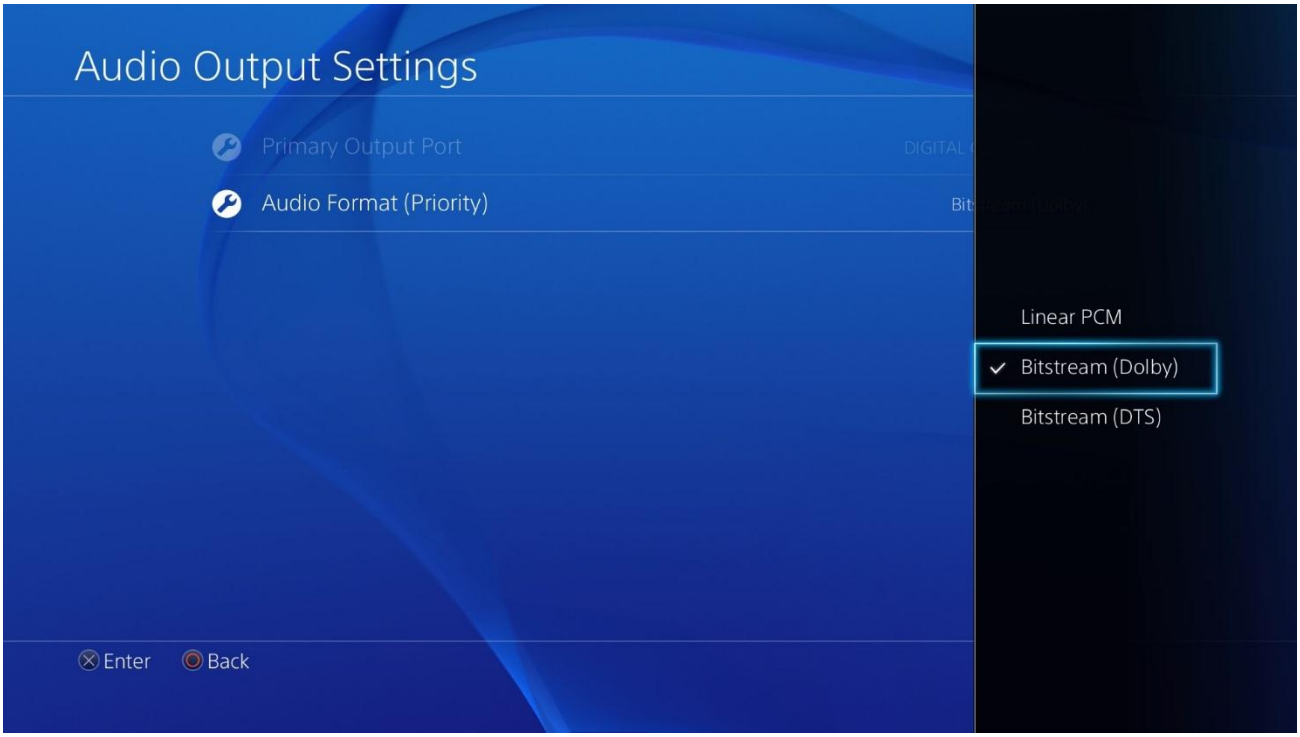
To configure the PS4 to use the optical digital output port, access the PS4 settings menu. Select Settings->[Sound and Screen]->[Audio Output Settings]



Select "Primary Output Port" and choose the "Digital Out (Optical)" option.

After you select the digital output option, you will be presented with a screen to select the audio formats supported by the Apollo Broadcast device. Make sure that "Dolby Digital 5.1ch" is selected and that "DTS 5.1ch" and "AAC" are not selected and press OK.

Select the "Audio Format (Priority)" option and select "Bitstream (Dolby)"



There is one other audio setting that may need to be changed. Once the movie has started playing, press the "Options" button on the PS4 controller to bring up the options menu on the bottom of the screen. Navigate to the "Settings" tab and press X. Make sure that the "Audio Format" is set to "Bitstream (Direct)"

1.7.5.2 Bluray 5.1 Test Content

The content below serves as a good starter for 5.1 testing.

Disney WOW: World of Wonder [Blu-ray] multi format

This title good for demos and customers

https://www.amazon.com/Disney-WOW-World-Wonder-Blu-ray/dp/B0045ASBLG?ie=UTF8&refRID=10G4JK7M1AQF4FM6MPVQ&ref_=pd_bxgy_74_img_2

3D Music Album - Demo & Audio Calibration Disc (3D Blu Ray) [Blu-ray]

This title is good for development and system setup, speaker localization, audio quality tests

https://www.amazon.com/3D-Music-Album-Calibration-Blu-ray/dp/B008P76W66?ie=UTF8&psc=1&redirect=true&ref_=oh_aui_detailpage_o02_s00

1.7.5.3 5.1 Source Configuration (WAE2/3 boards only)

Configure the WAE2/WAE3 platform as below

Wiced-SDK-3.7.x

```
> config apollo_role source
> config src_t capture
> config ad_rx 2
> config clock 1
> config pll 1
> dct_wifi_ap_list 0 channel <channel>
> dct_wifi_ap_list 0 ssid <name>
> dct_wifi_mac <mac>
> config save
> reboot
```

Wiced-SDK-4.0 and later

```
> config apollo_role source
> config src_t capture
> config ad_rx 2
> config clock 1
> config pll 1
> config rmc_chan <channel>
> config rmc_ssid <name>
> dct_wifi_mac <mac>
> config save
```

1.7.5.4 5.1 Sink Configuration

NOTE: you will specify each speaker (<speaker>) for each device below in the following sections below. The configuration for the device is the same on all speakers except for the speaker channel to be rendered.

Wiced-SDK-3.7.x

```
> config apollo_role sink
> config speaker_channel <speaker>
> config clock 1
> config pll 1
> dct_wifi_ap_list 0 channel <channel>
> dct_wifi_ap_list 0 ssid <name>
> dct_wifi_mac <mac>
> config save
```

Wiced-SDK-4.0 and later

```
> config apollo_role sink
> config speaker_channel <speaker>
> config clock 1
> config pll 1
> config rmc_chan <channel>
> config rmc_ssid <name>
> dct_wifi_mac <name>
> config save
```

1.7.6 5.1 Surround Sound Sink/Speaker Configuration

To configure for a 5.1 setup, each sink device will be configured to play one channel, the setup is as below, where <speaker> is one of FL, FR, FC, LS, RS, LFE1

1.7.6.1 Front Left Speaker

```
> config speaker_channel FL
> config save
```

1.7.6.2 Front Right Speaker

```
> config speaker_channel FL
> config save
```

1.7.6.3 Front Center Speaker (Dialogue channel)

```
> config speaker_channel FC
> config save
```

1.7.6.4 Rear (Surround) Left Speaker

```
> config speaker_channel LS
> config save
```

1.7.6.5 Rear (Surround) Right Speaker

```
> config speaker_channel RS
> config save
```

1.7.6.6 Low Frequency Effects (LFE1) - Subwoofer Speaker

```
> config speaker_channel LFE1
> config save
```

NOTE: With BRIX platforms, the above configuration should be used with all sink devices (43907 based).

1.8 Available Commands

Use the “config help” command for details on console commands. Note that changing some of the configuration settings below may render the system inoperable and require reconfiguration (as above).

```
> config help
Config commands:
    config                                  : output current config
    config <?|help>                         : show this list
    config auto_start <0|off|1|on>      : 0 = auto start off, 1 = auto start on
           auto       <0|off|1|on>
    config buffering_ms <xxx>               : xxx = milliseconds
           buff_ms <xxx>                    :       (range:0 <= xx <= 1000)
    config clock <0|disable|1|enable>    : 0 = disable AS clock, 1 = enable
    config pll    <0|disable|1|enable>    : 0 = disable audio PLL tuning, 1 =
enable
    config pll_tuning <0|disable|1|enable>
    config mac_addr <xx:xx:xx:xx:xx:xx> : xx:xx:xx:xx:xx:xx = new MAC address
    config mac <xx:xx:xx:xx:xx:xx>          : Shortcut:
    config mac <xx>                         :   enter 1 octet to change last octet
    config network_channel <xxx>            : xxx = channel
           net_chan <xxx>                   :  (1-11,36,40,44,48,52,56,60,64,
                                            :   100,104,108,112,116,120,124,128,
                                            :   132,136,140,149,153,157,161,165)
    config network_name <ssid_name>      : name of AP (max 32 characters)
           net_name <ssid_name>
           ssid <ssid_name>
    config network_passphrase <pass>     : passkey/password (max 64 characters)
           net_pass <pass>
           pass <pass>
    config network_security <type>       : security type is one of:
           net_sec    <type>                                                  :
open,none,ibss,wep,wepshared,wpatkip,wpaaes,
                                        :
wpa2tkip,wpa2aes,wpa2mix,wpsopen,wpsaes
    config speaker_name <name>           : speaker name (max 36 characters)
           spkr_name <name>
    config  speaker_channel  <ch>  [ch]...  :  channel  mix  -  all  will  be  OR'ed
together
           spkr_chan      <ch>      [ch]...                                   :
FL,FR,FC,LFE1,BL,BR,FLC,FRC,BC,LFE2,
```

```
                                        :
SIL,SIR,TPFL,TPFR,TPFC,TPC,TPBL,TPBR,
                                        :    TPSIL,TPSIR,TPBC,BTFC,BTFL,BTFR
    config threshold_ms <xx>            : xx = milliseconds
           thresh_ms <xx>               :      (range:0 <= xx <= 1000)
    config volume <xx>                  : xx = volume level
           vol <xx>                     :      (range:0 <= xx <= 100)
    config payload_size <size_in_bytes> : from 64 to 1432 bytes
    config fec_order     <string>         : "pre" = before audio, "post" = after
audio
    config fec_length   <packet_count>  : from 0 to 16
    config fec          <packet_count>
    config source_type   <string>          : "bt" = BT A2DP audio, "capture" =
using local ADC
    config src_t        <string>
    config audio_device_rx <device_X>    : enter X as in WICED_AUDIO_X, X starts
at 0 for all WICED platforms
    config ad_rx           <device_X>
    config audio_device_tx <device_X>    : enter X as in WICED_AUDIO_X, X starts
at 0 for all WICED platforms
    config ad_tx           <device_X>
    config apollo_role <source | sink>  : Configure as a source or a sink
    config role <source | sink>
    config clientaddr <IP address>      : Client IP address for sender to use
    config addr <IP address>
    config rtp_port <port number>       : RTP port number
    config port <port number>
    config log_level <level>              : Set the default application logging
level
    config log <level>
    config is_configured <0|no|1|yes>   : Set to 0 to force BT GATT configuration
    config is_conf       <0|no|1|yes>
    config bt_name <name>               : Set the bluetooth device name
    config bt_dev <name>
    config bt_mac <xx:xx:xx:xx:xx:xx>   : xx:xx:xx:xx:xx:xx = new Bluetooth MAC address
    config bt_mac <xx>                  :   enter 1 octet to change last octet
    config bt_class <xx:xx:xx>          : xx:xx:xx = new Bluetooth device class
    config save                         : save data to flash NOTE: Changes not
                                        :   automatically saved to flash!
>
```

1.9 Extra Debugging

Use the log command to get additional information on the source or sink. To make the setting persistent across reboot, use the “config loglevel” command. Valid values are 0 (no messages) to 9 (full debug messages). NOTE: when increasing the log-level for debug, audio artifacts may be heard due to too many UART interrupts during playback. This will be addressed in future releases. A log level of 4 is a useful debug level to start.

```
> log 9
Setting new log level to 10 (0 - off, 9 - max debug)
>
```

Sink Configuration

```
> config log_level 4
> config save
```

1.9.1 Build Options

Several build options are available from the command line/Eclipse as below:

Include Wireless LAN Commands (WL)

Note: this can only be used with BT/Analog rebroadcast, for SPDIF inclusion of Dolby Digital decoder exceeds memory capacity

```
demo.apollo-BCM943907WAE2_1 CONSOLE_INCLUDE_WL=1 download run
```

Verbose Build String

```
demo.apollo-BCM943907WAE2_1 VERBOSE=1 download run
```

1.10 OLED Display

The OLED display can be enabled to show AVRCP, Network, Wi-Fi and Battery state. In SDK 3.5.x/3.6.x the enable of OLED Display support is handled in the application Makefile

`apps/demo/apollo/apollo.mk`

```
# uncomment below to include display
GLOBAL_DEFINES += USE_AUDIO_DISPLAY
$(NAME)_COMPONENTS += audio/display
```

In SDK 4.x and later, this is enabled by adding the define to the build string:

USE_AUDIO_DISPLAY=1

So, for example, to build for WAE2_1 with Dolby Digital and OLED display support, the Eclipse build string becomes:

```
demo.apollo-BCM943907WAE2_1 USE_UDC=1 USE_AUDIO_DISPLAY=1 download run
```

2 iOS Apollo Audio Configuration Application

The iOS Apollo Configuration App demonstrates how to configure Wi-Fi and speaker settings of WICED Audio based speakers running the Apollo Wireless Audio stack vthrough Bluetooth Low Energy (BLE) on an iOS device.

2.1 Setup

- BCM43907WAE_1 or BCM43907WAE_2 WICED Audio boards. Up to 6 can be configured at a time. Follow the build instructions above.
- Install WICED-SDK-3.5.2 or later on your computer. NOTE: The Apollo iOS application is distributed as a patch available upon request from Cypress.
- Apple Mac
  - ☐ Clone and build the "Apollo Audio Config" source in Xcode from the SDK release 3.5.2 or later plus patch located in apps\demo\apollo\peerapps
- iPhone or iPad
  - ☐ Connect the iPad/iPhone to the Mac PC and install the iOS application
- WiFi Access point with internet connection.

Once the system starts, the application will come up as a remote speaker in its default configuration and you should see console output below

```
Config Info: * = dirty

  Apollo app DCT:

     is configured: yes

       apollo role: sink

      speaker name: Apollo

           channel: (0x00000003) FL FR

         buffering: 50 ms

         threshold: 40 ms

        auto_start: on

      clock enable: enable

 PLL tuning enable: enable

            volume: 50

       source type: bt

      payload size: 1432
```

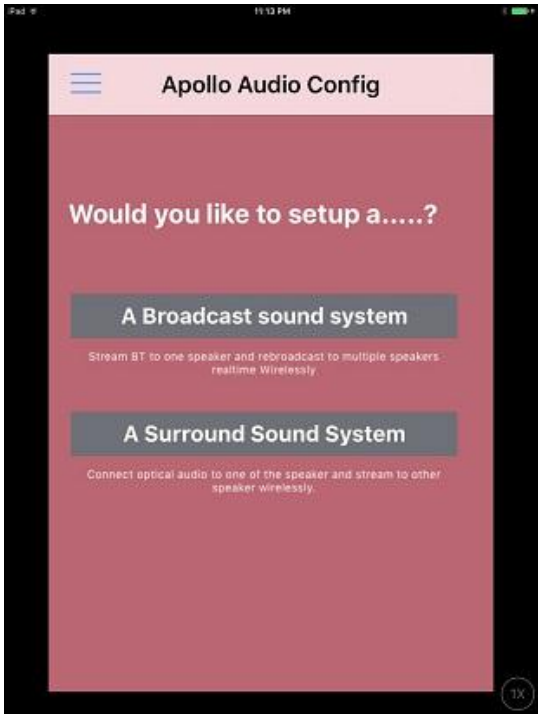
2.2 System configuration via the iOS Apollo application

- Step 1: Press the "BACK" button on the BCM43907WAE_1 or BCM43907WAE_2 WICED Audio boards. The Apollo audio application starts up, resets the device configuration and then launches a BLE GATT server advertising the Apollo service. You should see the console output below:

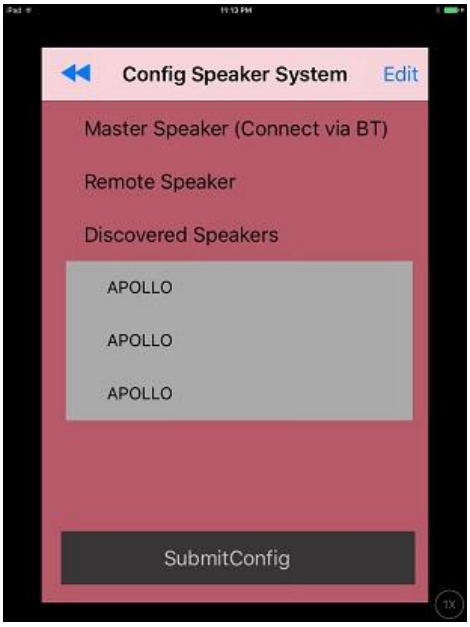
```
0000 00:00:02.217 *** Waiting for GATT configuration... ***

wiced_bt_ble_set_advertisement_data 0
```

- Step 2: On the iPhone or iPad, turn BT ON. Open the installed Apollo application whose icon appears as below:

- Step 3: The application's initial home screen will come up. Note: If the screen does not comes up there are no Speaker devices present running in GATT configuration mode. Separately, if no devices are found, confirm that BT is enabled on your iPhone/iPad.

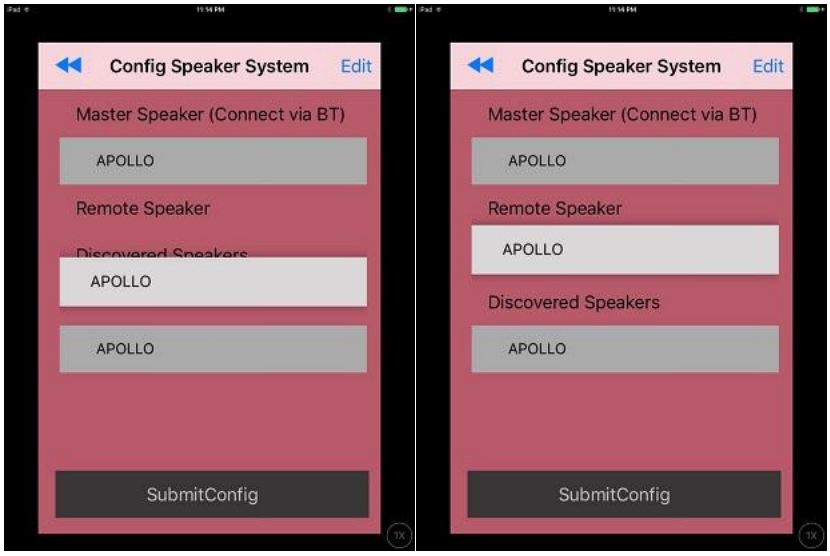


- Step 4: The iOS app only configures the Broadcast sound system in the first release. Press the button titled "A Broadcast sound system". This brings up the screen as below. Note: Pressing "A Surround Sound System" will not do anything.



- Step 5: In the above screen, the app lists all the discovered WICED devices which are running Apollo firmware in configuration mode. If you do not see a device or only one device being listed in the "Discovered Speakers" Section please go back to Step 1 to insure all devices are in the configuration state.

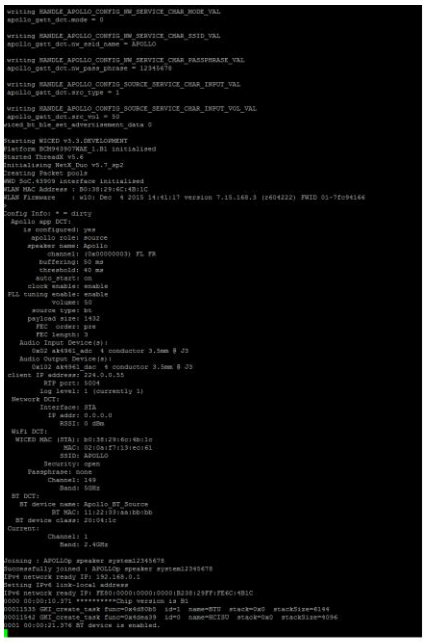
- Step 6: With at least two devices listed under the Discovered Speakers section, drag and drop each of these devices to the master and remote speaker section respectively. Tap and hold on the device in the Discovered Speakers area and then start dragging to a destination source device. Make sure the Master Speaker section and the Remote Speaker section have at least one device. The following screens below show the drag and drop association in action:

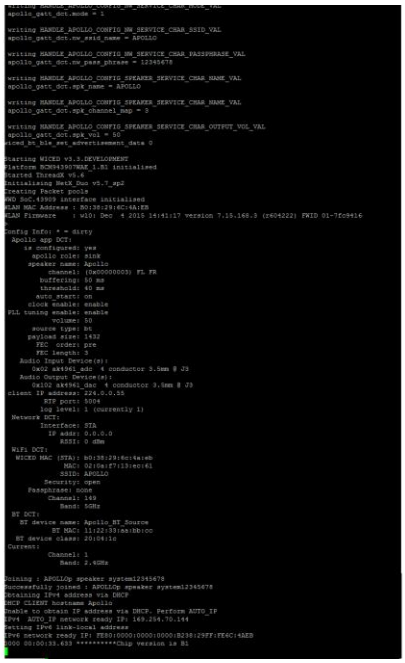


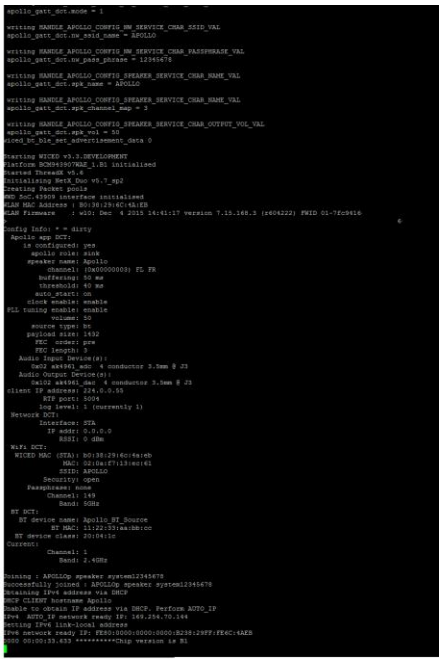
- Step 7: Tap on the Edit button in the top right corner, this brings up the soft keyboard to enter the Network Name in the Master Speaker Section. Once entered tap the enter key in the soft keyboard of iPad to close the keyboard. Now similarly enter speaker names for the speakers in the Remote Section by tapping on each of them (which pops up the soft keyboard). The following screens shows all of this in action:

- Step 8: Once done tap the "Done" button in the top right corner to signal the application that you are Done with the Editing of the devices.

- Step 9 : Tap the "SubmitConfig" button and you should see the WICED devices console print following where one of the device is being configured as a source device (Designated as “Master” in the app) and the all other devices are being configured as sink devices (remote speakers). Please verify that the master device starts a SoftAP with the network name you provided in the above screen and also verify the sink devices do the same. You should also verify that the sink devices connect to the source and properly obtain their IP addresses.







3 Roadmap

Future support for USB Speaker and rebroadcasting while WAE3 is connected to a USB Host.

Document Revision History

Document Title: WICED™ APOLLO Wireless Audio

Document Number:002-19004

Revision	ECN	Issue Date	Description of Change
**		12/18/2015	**WICED-APOLLO-1.0** : Initial release
		12/20/2016	**WICED-APOLLO-1.1** : Updated configuration commands for 3.x/4.x
*A		03/31/2017	Converted to Cypress template format

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP| PSoC 6

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709