



WICED Studio



WICED™ APOLLO Wireless Audio

Associated Part Family: CYW43907

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

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

Contents

About This Document.....	3
Purpose and Scope	3
Audience	3
Acronyms and Abbreviations	3
IoT Resources and Technical Support	3
1 Apollo Wireless Audio Distribution System.....	4
1.1 Audio Speaker Configurations	4
1.2 Stereo/2.1	4
1.3 UPnP Audio Rebroadcast.....	5
1.4 Equipment Needed.....	5
1.5 Building the Firmware.....	6
1.5.1.1 For SDK 6.0.....	6
1.5.2 Platform Build strings.....	7
1.5.3 Platform Build string variants	7
1.6 Configuration	8
1.6.1 Multi-Zone - Bluetooth Re-broadcaster (> 2 devices)	8
1.6.1.1 BT Source configuration	8
1.6.1.2 BT Stereo Sink Configuration(s).....	8
1.6.2 Multi-Zone - Analog Re-broadcaster	9
1.6.2.1 Analog Source configuration	9
1.6.2.2 Analog Stereo Sink Configuration	10
1.6.3 Multi-Speaker	10
1.6.3.1 Sink Configuration	10
1.6.4 UPnP Audio Rebroadcast.....	10
1.6.4.1 Source Configuration.....	10
1.6.4.2 Sink Configuration	11
1.6.4.3 Using BubbleUPnP with UPnP Audio Renderer Re-broadcaster	11
1.6.4.4 Installing and running BubbleUPnP on an Android-based phone or tablet	11
1.7 Available Commands.....	17
1.8 Extra Debugging.....	19
1.8.1 Build Options	20
1.9 OLED Display	20
2 iOS Apollo Audio Configuration Application	21
2.1 Setup	21
2.2 System configuration via the iOS Apollo application.....	21
Document Revision History	27
Worldwide Sales and Design Support.....	28
Products	28
PSoC® Solutions.....	28
Cypress Developer Community	28
Technical Support.....	28

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 6.0** 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. CYW943907WAE4 and BCM943907WCD2).

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). 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
Bluetooth A2DP	CYW943907WAE4	CYW943907WAE4	Stereo
Bluetooth A2DP	CYW943907WAE4	2 x CYW943907WAE4	Multi-zone (Stereo)
Analog Stereo	CYW943907WAE4	CYW943907WAE4	Stereo
Analog Stereo	CYW943907WAE4	2 x CYW943907WAE4	Multi-zone (Stereo)
SPDIF (Optical)/BT	CYW943907WAE4	2 x CYW943907WAE4	Stereo
SPDIF/BT	CYW943907WAE4	2 x CYW943907WAE4	Only 2 needed for BT
Analog/BT	BCM943907WCD2	2 x BCM943907WCD2	2 for WLAN rebroadcast

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), and 6 (5.1) may be built.

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 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

1.4 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

1.5 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.

1.5.1.1 For SDK 6.0

SDK 6.0 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

1.5.2 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 WAE4 board as source and WCD2 as Sink)

Wireless Connectivity Device 2 (WCD2 - SPIL)

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

Wireless Audio Edition 4 (WAE4)

```
demo.apollo-CYW943907WAE4 download run
```

1.5.3 Platform Build string variants

Wireless Audio Edition 4 (WAE4) – with UPnP AV Rendering support

```
demo.apollo-CYW943907WAE4 USE_UPNPAV=1 download run
```

Wireless Audio Edition 4 (WAE4) – with OLED Display Support (may be added to any build option)

```
demo.apollo-CYW943907WAE4 USE_AUDIO_DISPLAY=1download run
```

1.6 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:

1.6.1 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.

1.6.1.1 BT Source configuration

Wiced-SDK-6.0 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
```

1.6.1.2 BT Stereo Sink Configuration(s)

Wiced-SDK-6.0 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.

1.6.2 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.

1.6.2.1 Analog Source configuration

Wiced-SDK-6.0 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
```

1.6.2.2 Analog Stereo Sink Configuration

Wiced-SDK-6.0 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.

1.6.3 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).

1.6.3.1 Sink Configuration

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

1.6.4 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

1.6.4.1 Source Configuration

Wiced-SDK-6.0 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
```

1.6.4.2 Sink Configuration

Wiced-SDK-6.0 and later

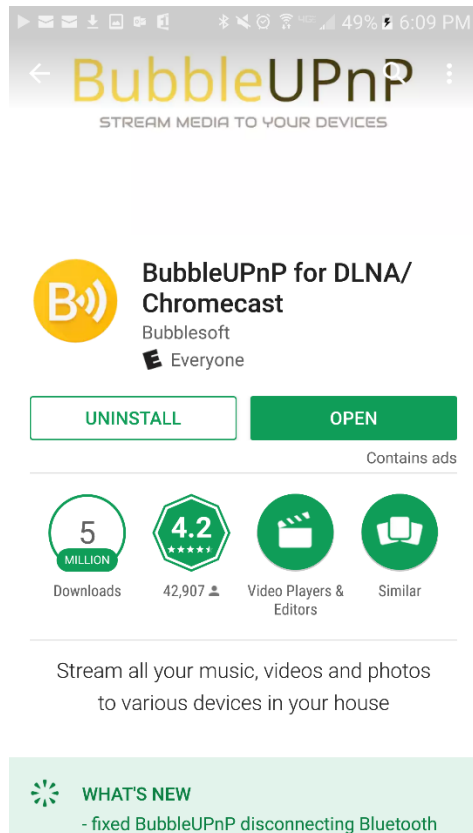
```
> config apollo_role sink
> config speaker_channel FL FR
> config rmc_chan 149
> config rmc_ssid Apollo-UPnP AV
> config save
> dct_wifi_mac aa:bb:cc:dd:ee:fa
> reboot
```

1.6.4.3 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.

1.6.4.4 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.

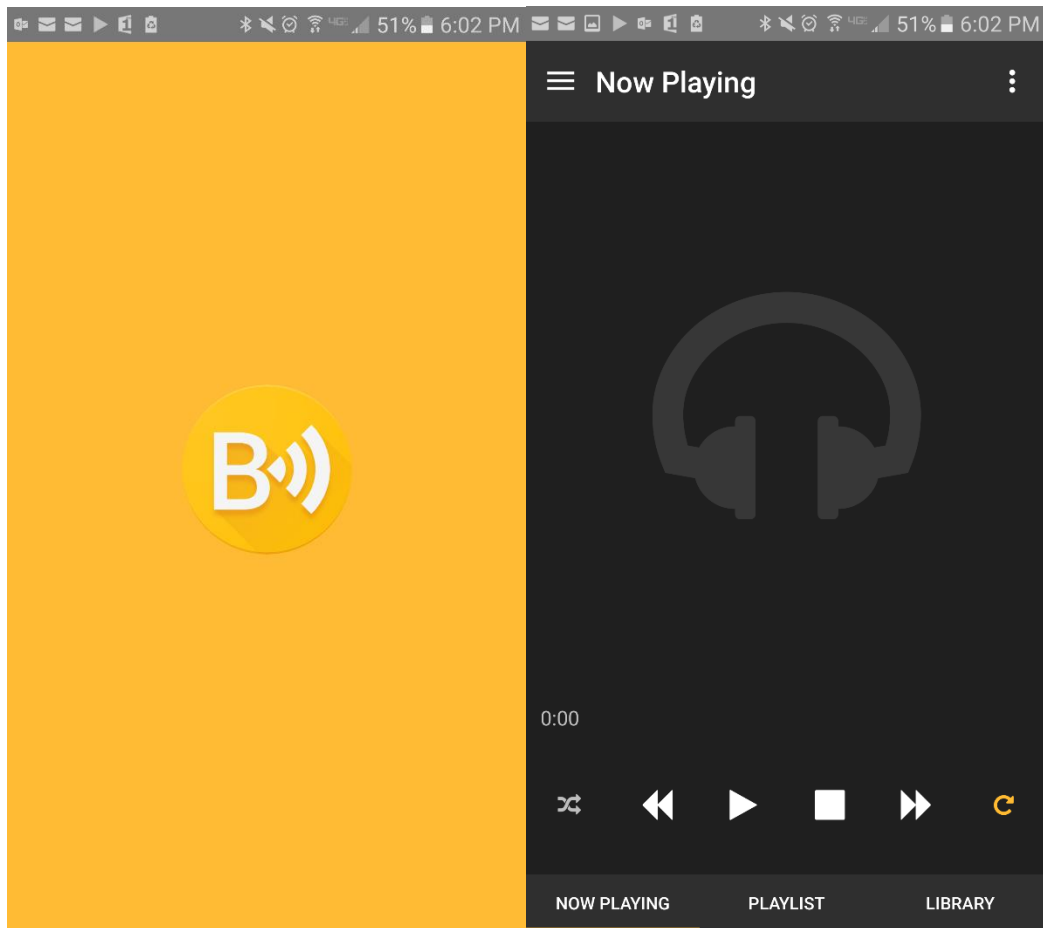


Next proceed with installation (the free version of BubbleUPnP is sufficient as it's fully functional); you may want to load up some audio content to your phone/tablet: MP3 and WAV (currently M4A/FLAC is not supported).

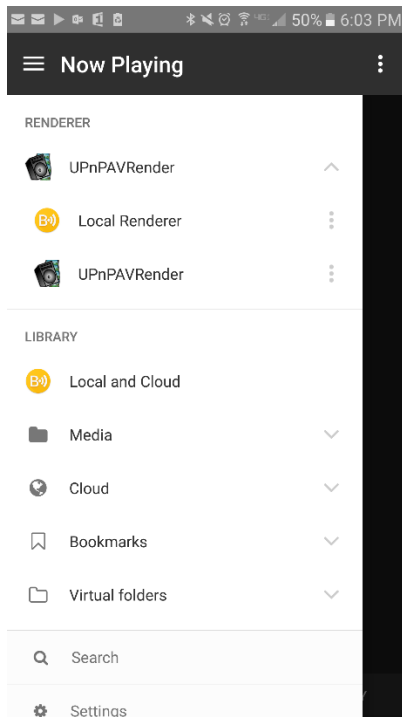
Associate the phone/tablet to the same upstream AP used by the Apollo Re-broadcaster

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

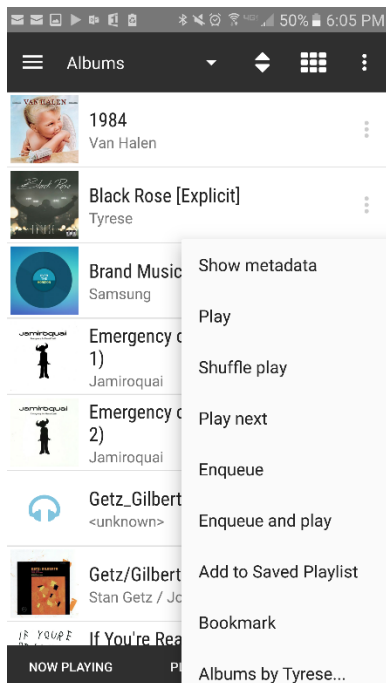
We're now ready to launch BubbleUPnP (that's the BubbleUPnP splash screen and "NOW PLAYING" tab is shown as below).



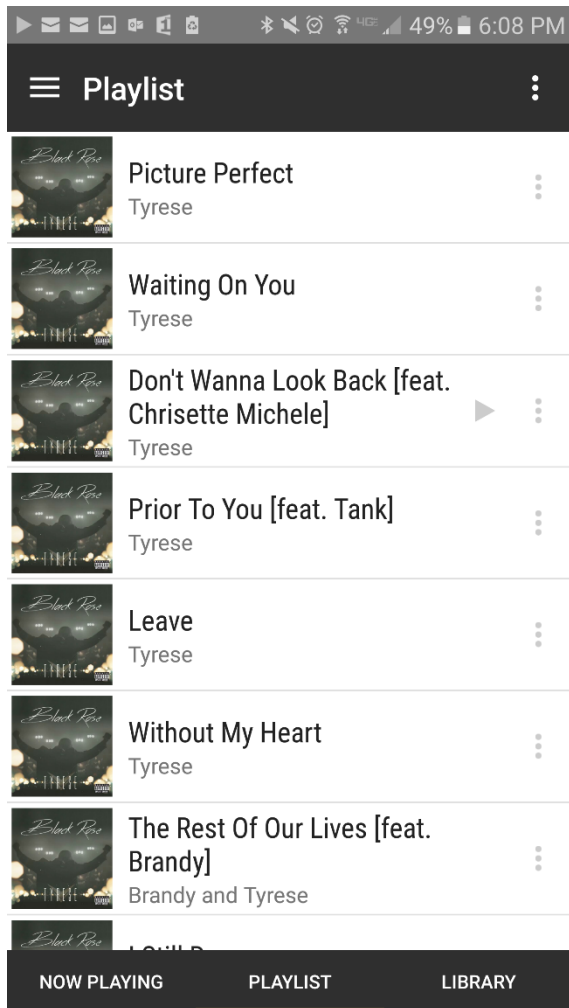
Next verify whether or not BubbleUPnP is able to find and recognize the Apollo Re-broadcaster as a Renderer. Tap/hit the area left of the "Now Playing" label to reveal the list of available Renderers. The "Local Renderer" refers to the phone/tablet itself (using whatever multimedia playback app available).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



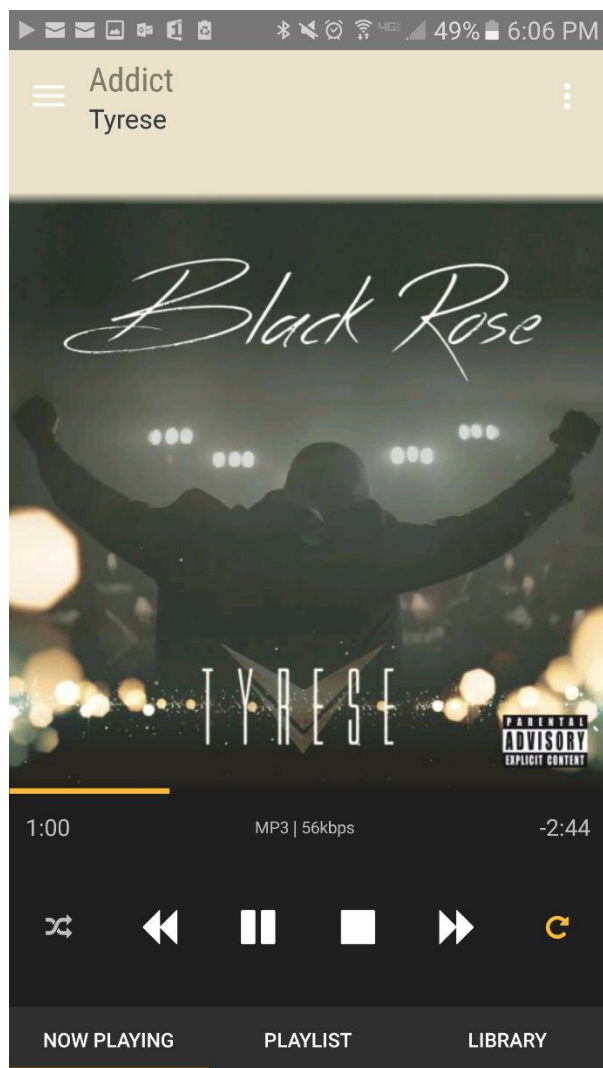
Go to the "LIBRARY" tab to initiate media playback (depending on what's available on your phone/tablet)



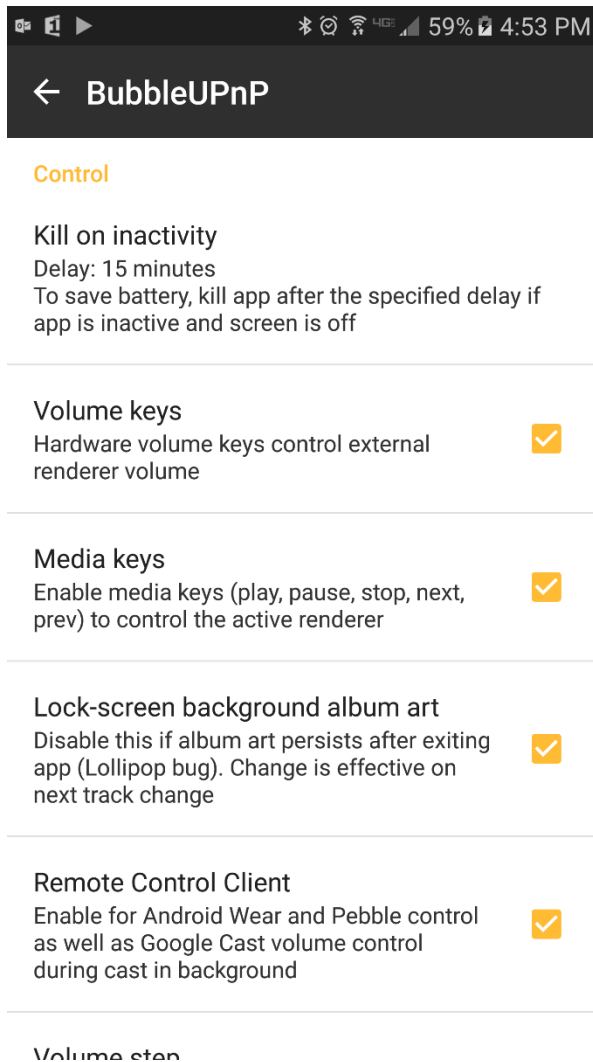
Tap/hit "Play" or "Enqueue and play" to add the selected Album or Songs to the playback list (the app will switch over to the "PLAYLIST" tab):



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 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:

<code>config</code>	: output current config
<code>config <? help></code>	: show this list
<code>config auto_start <0 off 1 on></code>	: 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,wpa2tkip,wpa2aes,
                                     :
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!
>

```

1.8 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
```

1.8.1 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

```
demo.apollo-CYW943907WAE4 CONSOLE_INCLUDE_WL=1 download run
```

Verbose Build String

```
demo.apollo-CYW943907WAE2 VERBOSE=1 download run
```

1.9 OLED Display

The OLED display can be enabled to show AVRCP, Network, Wi-Fi and Battery state. In SDK 6.0 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 6.0 and later, this is enabled by adding the define to the build string:

USE_AUDIO_DISPLAY=1

So, for example, to build for WAE4 with OLED display support, the Eclipse build string becomes:

```
demo.apollo-CYW943907WAE4 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 through Bluetooth Low Energy (BLE) on an iOS device.

2.1 Setup

- CYW43907WAE4 WICED Audio boards. Up to 6 can be configured at a time. Follow the build instructions above.
- Install WICED-SDK-6.0 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 6.0 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 CYW43907WAE4 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:

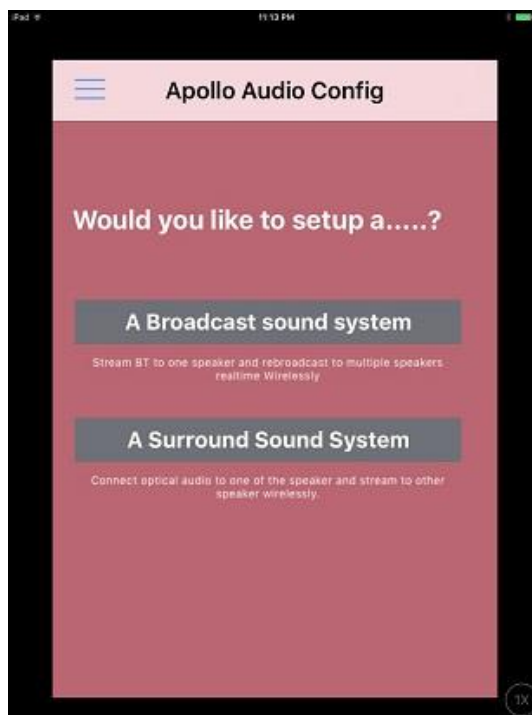
```
0000 00:00:02.217 *** Waiting for GATT configuration... ***
```

```
wiced_bt_ble_set_advertisement_data 0
```

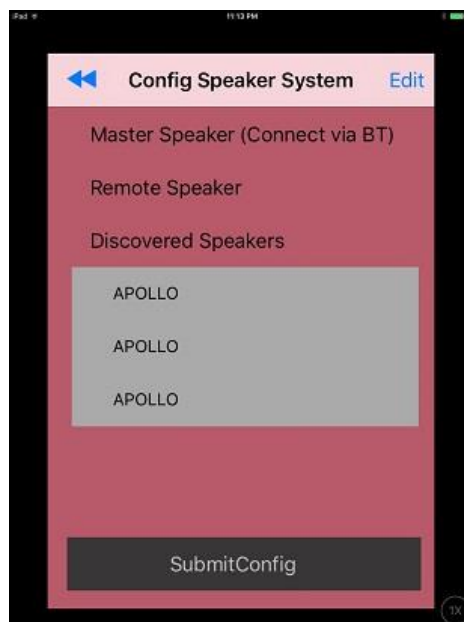
- Step 2: On the iPhone or iPad, turn BT ON. Open the installed Apollo application whose icon appears as below:



- Step 3: The application's initial home screen will come up. Note: If the screen does not come 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.

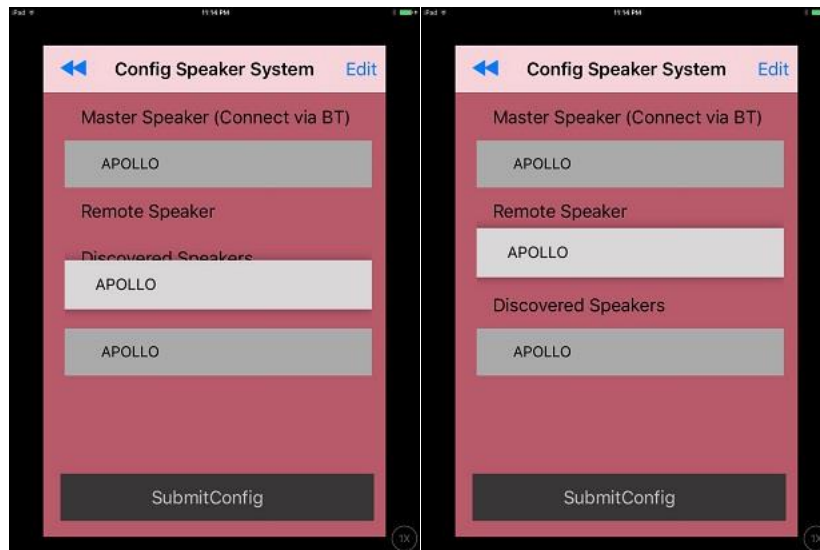


- Step 4: The iOS app only configures the Broadcast sound system in the first release. Press the button titled "A Broadcast sound system". This brings up the screen as below. Note: Pressing "A Surround Sound System" will not do anything.

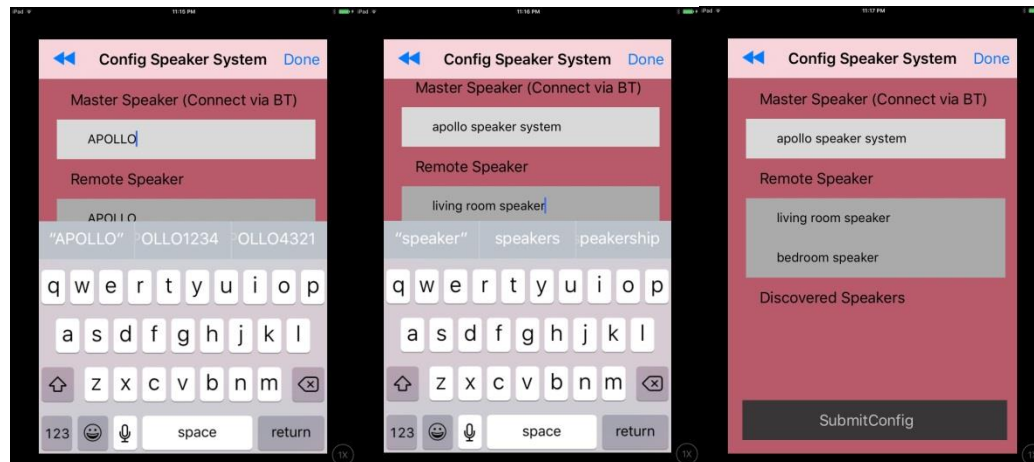


- Step 5: In the above screen, the app lists all the discovered WICED devices which are running Apollo firmware in configuration mode. If you do not see a device or only one device being listed in the "Discovered Speakers" Section please go back to Step 1 to insure all devices are in the configuration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

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

©Cypress Semiconductor Corporation, 20xx-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.