

# WL18xx TI Bluetooth Stack HDSM Demo App

[Return to WL18xx AM335x TI Bluetooth Stack](#)

## Contents

### Demo Overview

- Instructions for platform audio routing
- Running the Bluetooth Code

### Demo Application

- Headset Profile (HSP)
  - HDSM Client (HDSM\_AG)
  - HDSM Server (HDSM\_HS)

## Demo Overview

 **Note:** : **This instructions can be used to run this demo on the AM335x Platform.**


This demo allows users to evaluate TI's WL18xx device by using the [AM335x platform \(http://www.ti.com/tool/tmdxevm3358\)](http://www.ti.com/tool/tmdxevm3358). The Headset profile allows the user to demonstrate the use of Headset profile on embedded device. The Headset profile is used to connect a headset or speakerphone with a mobile device or used to connect a Audio gateway with headset device to provide basic control and voice connections. The Headset profile supports two roles, Headset and Audio Gateway. This document demonstrates how to use both roles of the profile.

It is recommended that the user visits the kit setup [Getting Started Guide for AM335x \(http://processors.wiki.ti.com/index.php/WL18xx\\_AM335x\\_TI\\_Bluetooth\\_Stack\)](http://processors.wiki.ti.com/index.php/WL18xx_AM335x_TI_Bluetooth_Stack) pages before trying the application described on this page.

## Instructions for platform audio routing

The am335x platform can route the Audio to the Microphone and Speakers Jacks on the development board. The audio can be routed only on development boards with the main board version 1.5 or later, Doughter board 1.2. Two audio devices should be added to the platform configuration- AM335xEVM and WILINK8BT. For reconfiguring the platform kernel, the user should download the SDK version 3.14.43 for the AM335x. After downloading the kernel and compilation under the Linux folder: `~/ti-processor-sdk-linux-am335x-evm-01.00.00.03/`. For the kernel download and compilation instruction follow the link: [TI Linux Kernel Users Guid \(http://processors.wiki.ti.com/index.php/Linux\\_Kernel\\_Users\\_Guide\)](http://processors.wiki.ti.com/index.php/Linux_Kernel_Users_Guide) For the BluetopiaPM environment follow the link: [Bluetopia PM Build Process \(http://processors.wiki.ti.com/index.php/TI\\_Bluetooth\\_Stack\\_for\\_WL18xx\\_-\\_Build\\_Process\)](http://processors.wiki.ti.com/index.php/TI_Bluetooth_Stack_for_WL18xx_-_Build_Process)

When the setup is ready go to the kernel folder: `~/ti-processor-sdk-linux-am335x-evm-01.00.00.03/board-support/linux-3.14.43+gitAUTONIC+875c69b2c3-g875c89b`. Then copy the following patch to the kernel folder and type: "git apply 0001-New-Feature-BT-PCM-integration-on-the-AM335x-EVM.PATCH".


 **Note:** : Currently all Sitara boards 1.5 and later needs the following patch to route the Audio in HFRM Demo App:

Description	Patch type	Support for board	Link to patch
HFP Audio route in the Sitara Patch	Bug fix	All AM335x variants	<patch> ( <a href="https://gforge.ti.com/gf/download/user/10583/7930/0001-New-Feature-BT-PCM-integration-on-the-AM335x-EVM.PATCH">https://gforge.ti.com/gf/download/user/10583/7930/0001-New-Feature-BT-PCM-integration-on-the-AM335x-EVM.PATCH</a> )

The patch requires recompiling the zImage:

```
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- zImage
cp arch/arm/boot/zImage [target-rootfs]/boot/
```

After the new configurations done, the devices, AM335xEVM and WILINK8BT, should appear in `/proc/asound/` folder on the platform's file system. The BluetopiaPM application version should fit the Bluetooth device, cc256x or wl18xx, for the audio to work.

 **Note:** When using a CC256x controller you will need to change the init script in order to enable AVPR. You can use the following command in the AM335x device terminal

```
cp -f /lib/firmware/TIinit_6.7.16_avpr_on.bts /lib/firmware/TIinit_6.7.16.bts
```

## Running the Bluetooth Code

Once the am335x is initialized on the terminal, follow the instructions;  
Enter to the Sample Application folder, for this example "cd BluetopiaPM/bin"

```
root@am335x-evm:~# cd BluetopiaPM/bin/
root@am335x-evm:~/BluetopiaPM/bin#
```

# Demo Application

This section provides a description of how to use the demo application to connect smart phone over Bluetooth HSP profile, the same for the second smart phone. Bluetooth HSP is a simple Client-Server connection process with one side, the client, operating in the Audio-Gateway role and the other, the server, operating in the Headset role. We will setup the boards one as a Headset server and the second one as Audio gateway Client.

## Headset Profile (HSP)

### HDSM Client (HDSM\_AG)

To verify HDSM\_AG client role functionality, first run the "LinuxHDSM\_AG" executable, initialize, and power on the local device if necessary.

Start the Server and the Demo application by typing `./SS1BTPM & ./LinuxHDSM_AG`

```
root@am335x-evm:~/BluetopiaPM/bin# ./SS1BTPM & ./LinuxHDSM_AG
[1] 2054
```

Now the help screen is displayed

```
*****
* Command Options: 1) Initialize *
* 2) Cleanup *
* 3) QueryDebugZoneMask *
* 4) SetDebugZoneMask *
* 5) ShutdownService *
* 6) RegisterEventCallback, *
* 7) UnRegisterEventCallback, *
* 8) QueryDevicePower *
* 9) SetDevicePower *
* 10)QueryLocalDeviceProperties *
* 11)SetLocalDeviceName *
* 12)SetLocalClassOfDevice *
* 13)SetDiscoverable *
* 14)SetConnectable *
* 15)SetPairable *
* 16)StartDeviceDiscovery *
* 17)StopDeviceDiscovery *
* 18)QueryRemoteDeviceList *
* 19)QueryRemoteDeviceProperties *
* 20)AddRemoteDevice *
* 21)DeleteRemoteDevice *
* 22)DeleteRemoteDevices *
* 23)PairWithRemoteDevice *
* 24)CancelPairWithRemoteDevice *
* 25)UnPairRemoteDevice *
* 26)QueryRemoteDeviceServices *
* 27)RegisterAuthentication *
* 28)UnRegisterAuthentication *
* 29)PINCodeResponse *
* 30)PassKeyResponse *
* 31)UserConfirmationResponse *
* 32)ChangeSimplePairingParameters *
* 33)ConnectDevice *
* 34)DisconnectDevice *
* 35)ConnectionRequestResponse *
* 36)ManageAudio *
* 37)SetSpeakerGain *
* 38)SetMicrophoneGain *
* 39)QueryHeadsetServices *
* 40)ChangeIncomingConnectionFlags *
* 41)RingIndication *
* 42)HDSRegisterEventCallback *
* 43)HDSUnRegisterEventCallback *
* 44)HDSRegisterDataCallback *
* 45)HDSUnRegisterDataCallback *
* 46)EnableBluetoothDebug *
* Help, Quit. *
*****
```

- Initialize the profile by typing "1 1" or "Initialize 1".
- Powering up the Bluetooth device by typing "9 1" or "SetDevicePower 1", if the device is already powered up you'll receive an error that the device is already powered up

```
Headset>Initialize 1
BTPM_Initialize() Success: 0.
DEVM_RegisterEventCallback() Success: 5.
Headset>SetDevicePower 1

echo 117 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio117/direction
echo 0 > /sys/class/gpio/gpio117/value
echo 1 > /sys/class/gpio/gpio117/value
BT COMM PORT (/dev/ttyO1): 1
Changing HCI baud rate to 3000000
Status: Executing BTS Script /lib/firmware/TIIInit_11.8.32.bts.
Status: BTS Script successfully executed.
DEVM_PowerOnDevice() Success: 0.
Headset>
```

```
Device Powered On.  
Headset>
```

### c) Register HDS Event and Data Callbacks.

```
Usage: HDSRegisterEventCallback [(0 = Non-Control Callback, 1 = Control Callback)].
```

```
Headset>HDSRegisterEventCallback 1  
HDSM_Register_Event_Callback() Success: 1.  
Headset>
```

```
Headset>HDSRegisterDataCallback  
HDSM_Register_Data_Event_Callback() Success: 2.  
Headset>
```

### d) Register Authentication

```
Headset>RegisterAuthentication  
DEVN_RegisterAuthentication() Success: 5.  
Headset>
```


e) In order to make connection, we need to check the RFCOMM ID, use QueryRemoteDeviceServices function in order to discover SDP.

```
Usage: QueryRemoteDeviceServices [BD_ADDR] [Force Update] [Bytes to Query (specified if Force is 0)].
```

```
Headset>QueryRemoteDeviceServices 84DD209C58DD 1  
Attempting Query Remote Device 84DD209C58DD For Services.  
DEVN_QueryRemoteDeviceServices() Success: 0, Total Number Service Bytes: 0.  
Headset>  
Remote Device Properties Changed.  
BD_ADDR: 84DD209C58DD  
Device Flags: 0x800001CD  
Connect State: TRUE  
Headset>  
Remote Device 84DD209C58DD Services Status: BR/EDR, SUCCESS.  
Headset>  
Remote Device Properties Changed.  
BD_ADDR: 84DD209C58DD  
Device Flags: 0x800001CD  
Serv. Known : TRUE  
Headset>  
Remote Device Properties Changed.  
BD_ADDR: 84DD209C58DD  
Device Flags: 0x80000185  
Connect State: FALSE  
Headset>
```

```
Headset>QueryRemoteDeviceServices 84DD209C58DD 0 2000  
Attempting Query Remote Device 84DD209C58DD For Services.  
DEVN_QueryRemoteDeviceServices() Success: 813, Total Number Service Bytes: 813.  
Returned Service Data (813 Bytes):...
```

```
Service Record: 2:  
Attribute ID 0x0000  
Type: Unsigned Int = 0x00010002  
Attribute ID 0x0001  
Type: Data Element Sequence  
Type: UUID_16 = 0x1108  
Type: UUID_16 = 0x1203  
Attribute ID 0x0002  
Type: Unsigned Int = 0x00000005  
Attribute ID 0x0004  
Type: Data Element Sequence  
Type: Data Element Sequence  
Type: UUID_16 = 0x0100  
Type: Data Element Sequence  
Type: UUID_16 = 0x0003  
Type: Unsigned Int = 0x02  
Attribute ID 0x0006  
Type: Data Element Sequence  
Type: Unsigned Int = 0x656E  
Type: Unsigned Int = 0x006A  
Type: Unsigned Int = 0x0100  
Attribute ID 0x0009  
Type: Data Element Sequence  
Type: Data Element Sequence  
Type: UUID_16 = 0x1108  
Type: Unsigned Int = 0x0102  
Attribute ID 0x0100  
Type: Text String = Headset  
Attribute ID 0x0302  
Type: Boolean = TRUE
```

 **Note:** : The RFCOMM we are searching according to the spec is "UUID\_16 = 0x0003", that means the port id is "Unsigned Int = 0x02"

### f) Connect to the Headset

```
Usage: Connect [BD_ADDR] [Port Number (optional)] [Connection Flags (optional - 0 = No Flags, 1 = Authentication, 2 = Encryption, 3 = Authentication/Encryption)].
```

```
Headset>ConnectDevice 84DD209C58DD
Service Name: Headset.
RFCOMM Port Number: 0x02.
HDS Profile Version: 0x0102.
HDSM_Connect_Remote_Device: Function Successful.
Headset>
Remote Device Properties Changed.
BD_ADDR: 84DD209C58DD
Device Flags: 0x80001CD
Connect State: TRUE
Headset>
Remote Device Properties Changed.
BD_ADDR: 84DD209C58DD
Device Flags: 0x80001DD
Encrypt State: TRUE
Headset>
hetHDSConnectionStatus, BD_ADDR: 84DD209C58DD, Type: sctAudioGateway, Status: 0.
Headset>
hetHDSConnected, BD_ADDR: 84DD209C58DD, Type: sctAudioGateway.
Headset>
hetHDSConnected, BD_ADDR: 84DD209C58DD, Type: sctAudioGateway.
Headset>
```

g) Sending indications - Ring indication.

```
Usage: RingIndication [BD_ADDR].
```

```
Headset>RingIndication 84DD209C58DD
HDSM_Ring_Indication: Function Successful.
Headset>
```

At the Head Set side (When using another AM335x as a Head Set device) you will see

```
hetHDSRingIndication, BD_ADDR: D03972507A42.
Headset>
```

h) In order to change Speaker gain, use "SetSpeakerGain" Command

```
Usage: SetSpeakerGain [BD_ADDR] [0 <= SpeakerGain <= 15].
```

```
SetSpeakerGain D03972507A42 5
HDS_Set_Remote_Speaker_Gain: Function Successful.
Headset>
```

At the Head Set side (When using another AM335x as a Head Set device) you will see

```
hetHDSSpeakerGainIndication, BD_ADDR: D03972507A42, Type: sctHeadset, Speaker Gain: 5.
Headset>
```

i) In order to change Microphone gain, use "SetMicrophoneGain" Command

```
Usage: SetMicrophoneGain [BD_ADDR] [0 <= MicrophoneGain <= 15].
```

```
SetMicrophoneGain D03972507A42 5
HDSM_Set_Remote_Microphone_Gain(): Function Successful.
Headset>
```

At the Head Set side (When using another AM335x as a Head Set device) you will see

```
hetHDSMicrophoneGainIndication, BD_ADDR: D03972507A42, Type: sctHeadset, Microphone Gain: 5.
Headset>
```

j)

```
Usage: ManageAudio [BD_ADDR] [Release = 0, Setup = 1] [InBandRinging (0 = No, 1 = In Band Ringing) only valid if Setup = 1].
```

```
Headset>ManageAudio 84DD209C58DD 1 0
HDSM_Setup_Audio_Connection: Function Successful.
Headset>
hetHDSAudioConnected, BD_ADDR: 84DD209C58DD, Type: sctAudioGateway.
Headset>
hetHDSAudioConnected, BD_ADDR: 84DD209C58DD, Type: sctAudioGateway.
Headset>
```

At the Head Set side (When using another AM335x as a Head Set device) you will see

```
hetHDSAudioConnected, BD_ADDR: D03972507A42, Type: sctHeadset.
Headset>
```

```
hetHDSAudioConnected, BD_ADDR: D03972507A42, Type: sctHeadset.  
Headset>
```

k)

```
Headset>ManageAudio 84DD209C58DD 0  
HDSM_Release_Audio_Connection: Function Successful.  
Headset>
```

At the Head Set side (When using another AM335x as a Head Set device) you will see

```
hetHDSAudioDisconnected, BD_ADDR: D03972507A42, Type: sctHeadset  
Headset>  
hetHDSAudioDisconnected, BD_ADDR: D03972507A42, Type: sctHeadset  
Headset>
```

l) When You finish using the Head Set Device, Use "DisconnectDevice" Command

```
Usage: Disconnect [BD_ADDR].
```

```
Headset>DisconnectDevice 84DD209C58DD  
  
hetHDSDisconnected, BD_ADDR: 84DD209C58DD, Type: sctAudioGateway, Reason: 0.  
Headset>  
hetHDSDisconnected, BD_ADDR: 84DD209C58DD, Type: sctAudioGateway, Reason: 0.  
Headset>HDSM_Disconnect_Device: Function Successful.  
Headset>  
Remote Device Properties Changed.  
BD_ADDR: 84DD209C58DD  
Device Flags: 0x80000185  
Connect State: FALSE  
Encrypt State: FALSE  
Headset>
```

## HDSM Server (HDSM\_HS)

To verify HDSM\_HS server role functionality, first run the "LinuxHDSM\_HS" executable, initialize, and power on the local device if necessary.

Start the Server and the Demo application by typing ". /SS1BTPM & ./LinuxHDSM\_HS"

```
root@am335x-evm:~/BluetopiaPM/bin# ./SS1BTPM & ./LinuxHDSM_HS  
[1] 1674
```

Now the help screen is displayed

```
*****  
* Command Options: 1) Initialize *  
* 2) Cleanup *  
* 3) QueryDebugZoneMask *  
* 4) SetDebugZoneMask *  
* 5) ShutdownService *  
* 6) RegisterEventCallback, *  
* 7) UnRegisterEventCallback, *  
* 8) QueryDevicePower *  
* 9) SetDevicePower *  
* 10)QueryLocalDeviceProperties *  
* 11)SetLocalDeviceName *  
* 12)SetLocalClassOfDevice *  
* 13)SetDiscoverable *  
* 14)SetConnectable *  
* 15)SetPairable *  
* 16)StartDeviceDiscovery *  
* 17)StopDeviceDiscovery *  
* 18)QueryRemoteDeviceList *  
* 19)QueryRemoteDeviceProperties *  
* 20)AddRemoteDevice *  
* 21)DeleteRemoteDevice *  
* 22)DeleteRemoteDevices *  
* 23)PairWithRemoteDevice *  
* 24)CancelPairWithRemoteDevice *  
* 25)UnPairRemoteDevice *  
* 26)QueryRemoteDeviceServices *  
* 27)RegisterAuthentication *  
* 28)UnRegisterAuthentication *  
* 29)PINCodeResponse *  
* 30)PassKeyResponse *  
* 31)UserConfirmationResponse *  
* 32)ChangeSimplePairingParameters *  
* 33)ConnectDevice *  
* 34)DisconnectDevice *  
* 35)ConnectionRequestResponse *  
* 36)ManageAudio *  
* 37)SetSpeakerGain *  
* 38)SetMicrophoneGain *  
* 39)QueryHeadsetServices *  
* 40)ChangeIncomingConnectionFlags *  
* 41)SendButtonPress *
```

```

*          42)HDSRegisterEventCallback          *
*          43)HDSUnRegisterEventCallback        *
*          44)HDSRegisterDataCallback          *
*          45)HDSUnRegisterDataCallback        *
*          46)EnableBluetoothDebug            *
*          Help, Quit.                          *
*****

```

- a) Initialize the profile by typing "1 1" or "Initialize 1".
- b) Powering up the Bluetooth device by typing "9 1" or "SetDevicePower 1", if the device is already powered up you'll receive an error that the device is already powered up

```

Headset>Initialize 1
BTM_Inititalize() Success: 0.
DEVM_RegisterEventCallback() Success: 5.
Headset>SetDevicePower 1
echo 117 > /sys/class/gpio/export
sh: line 0: echo: write error: Device or resource busy
echo out > /sys/class/gpio/gpio117/direction
echo 0 > /sys/class/gpio/gpio117/value
echo 1 > /sys/class/gpio/gpio117/value
BT COMM PORT (/dev/ttyS1): 1
Changing HCI baud rate to 3000000
Status: Executing BTS Script /lib/firmware/TIInit_12.10.28.bts.
Status: BTS Script successfully executed.
DEVM_PowerOnDevice() Success: 0.
Headset>
Device Powered On.
Headset>

```

- c) Register HDS Event and Data Callbacks.

```

Usage: HDSRegisterEventCallback [(0 = Non-Control Callback, 1 = Control Callback)].

```

```

Headset>HDSRegisterEventCallback 1
HDSM_Register_Event_Callback() Success: 1.
Headset>

```

```

Headset>HDSRegisterDataCallback
HDSM_Register_Data_Event_Callback() Success: 2.
Headset>

```

- d) Register Authentication

```

Headset>RegisterAuthentication
DEVM_RegisterAuthentication() Success: 5.
Headset>

```

- e) In order to make connection, we need to check the RFCOMM ID, use QueryRemoteDeviceServices function in order to discover SDP.

```

Usage: QueryRemoteDeviceServices [BD_ADDR] [Force Update] [Bytes to Query (specified if Force is 0)].

```

```

Headset>QueryRemoteDeviceServices 84DD209C58DD 1
Attempting Query Remote Device 84DD209C58DD For Services.
DEVM_QueryRemoteDeviceServices() Success: 0, Total Number Service Bytes: 0.
Headset>
Remote Device Properties Changed.
BD_ADDR:      84DD209C58DD
Device Flags: 0x80001CD
Connect State: TRUE
Headset>
Remote Device 84DD209C58DD Services Status: BR/EDR, SUCCESS.
Headset>
Remote Device Properties Changed.
BD_ADDR:      84DD209C58DD
Device Flags: 0x80001CD
Serv. Known : TRUE
Headset>
Remote Device Properties Changed.
BD_ADDR:      84DD209C58DD
Device Flags: 0x8000185
Connect State: FALSE
Headset>

```

```


Headset>QueryRemoteDeviceServices 84DD209C58DD 0 2000
Attempting Query Remote Device 84DD209C58DD For Services.
DEVM_QueryRemoteDeviceServices() Success: 813, Total Number Service Bytes: 813.
Returned Service Data (813 Bytes):...
Service Record: 1:
  Attribute ID 0x0000
    Type: Unsigned Int = 0x00010001
  Attribute ID 0x0001
    Type: Data Element Sequence
      Type: UUID_16 = 0x1112
      Type: UUID_16 = 0x1203
  Attribute ID 0x0002
    Type: Unsigned Int = 0x00000004

```

```

Attribute ID 0x0004
  Type: Data Element Sequence
    Type: Data Element Sequence
      Type: UUID_16 = 0x0100
    Type: Data Element Sequence
      Type: UUID_16 = 0x0003
      Type: Unsigned Int = 0x04
Attribute ID 0x0006
  Type: Data Element Sequence
    Type: Unsigned Int = 0x656E
    Type: Unsigned Int = 0x006A
    Type: Unsigned Int = 0x0100
Attribute ID 0x0009
  Type: Data Element Sequence
    Type: Data Element Sequence
      Type: UUID_16 = 0x1108
      Type: Unsigned Int = 0x0102
Attribute ID 0x0100
  Type: Text String = Headset - AG

```

 **Note** : The RFCOMM we are searching according to the spec is "UUID\_16 = 0x0003", that means the port id is "Unsigned Int = 0x04"

#### f) Connect to the Audio Gateway

Usage: Connect [BD\_ADDR] [Port Number (optional)] [Connection Flags (optional - 0 = No Flags, 1 = Authentication, 2 = Encryption, 3 = Authentication/Encryption)].

```

Headset>ConnectDevice 84DD209C58DD
Service Name: Headset - AG.
RFCOMM Port Number: 0x04.
HDS Profile Version: 0x0102.
HDSM_Connect_Remote_Device: Function Successful.
Headset>
Remote Device Properties Changed.
BD_ADDR:      84DD209C58DD
Device Flags: 0x800001CD
Connect State: TRUE
Headset>
Remote Device Properties Changed.
BD_ADDR:      84DD209C58DD
Device Flags: 0x800001DD
Encrypt State: TRUE
Headset>
hetHDSConnectionStatus, BD_ADDR: 84DD209C58DD, Type: sctHeadset, Status: 0.
Headset>
hetHDSConnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset.
Headset>
hetHDSConnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset.
Headset>

```

#### g) In order to Answer a call or Disconnect active call use "SendButtonPress" Command

Usage: SendButtonPress [BD\_ADDR].

#### Answer Call

```

Headset>SendButtonPress 84DD209C58DD
HDSM_Send_Button_Press: Function Successful.
Headset>
Remote Device Properties Changed.
BD_ADDR:      84DD209C58DD
COD:          0x1C0424
Headset>
hetHDSAudioConnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset.
Headset>
hetHDSAudioConnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset.
Headset>

```

At the Audio Gateway side (When using another AM335x as a Audio Gateway device) you will see

```

hetHDSAudioConnected, BD_ADDR: D03972507A42, Type: sctAudioGateway.
Headset>
hetHDSAudioConnected, BD_ADDR: D03972507A42, Type: sctAudioGateway.
Headset>

```

#### Disconnect Call

```

Headset>SendButtonPress 84DD209C58DD
HDSM_Send_Button_Press: Function Successful.
Headset>
hetHDSAudioDisconnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset
Headset>
hetHDSAudioDisconnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset
Headset>

```

At the Audio Gateway side (When using another AM335x as a Audio Gateway device) you will see

```

hetHDSAudioDisconnected, BD_ADDR: D03972507A42, Type: sctAudioGateway
Headset>
hetHDSAudioDisconnected, BD_ADDR: D03972507A42, Type: sctAudioGateway
Headset>

```

h) In order to change Speaker gain or Microphone gain you can use the following command "SetSpeakerGain" "SetMicrophoneGain"

```
Usage: SetSpeakerGain [BD_ADDR] [0 <= SpeakerGain <= 15].
```

```
Usage: SetMicrophoneGain [BD_ADDR] [0 <= MicrophoneGain <= 15].
```

i) You can Setup or release your audio connection, by using "ManageAudio" command

```
Usage: ManageAudio [BD_ADDR] [Release = 0, Setup = 1].
```

j) When You finish using the Audio Gateway Device, Use "DisconnectDevice" Command

```
Usage: Disconnect [BD_ADDR].
```

```
Headset>DisconnectDevice 84DD209C58DD
```

```
HeadsetDisconnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset, Reason: 0.
Headset>
HeadsetDisconnected, BD_ADDR: 84DD209C58DD, Type: sctHeadset, Reason: 0.
Headset>HDSM_Disconnect_Device: Function Successful.
Headset>
Remote Device Properties Changed.
BD_ADDR: 84DD209C58DD
Device Flags: 0x80000185
Connect State: FALSE
Encrypt State: FALSE
Headset>
```

<pre> {{ 1. switchcategory:MultiCore=   ■ For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum   ■ For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum Please post only comments related to the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>	<pre> Keystone=   ■ For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum   ■ For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum Please post only comments related to the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>	<pre> C2000=For technical support on the C2000 please post your questions on The C2000 Forum. Please post only comments about the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>	<pre> DaVinci=For technical support on DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>	<pre> MSP430=For technical support on MSP430 please post your questions on The MSP430 Forum. Please post only comments about the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>	<pre> OMAP35x=For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>	<pre> OMAPL1=For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>	<pre> MAVRK=For technical support on MAVRK please post your questions on The MAVRK Toolbox Forum. Please post only comments about the article <b>WL18xx TI Bluetooth Stack HDSM Demo App</b> here.         </pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Links



[Amplifiers & Linear Audio](#)  
[Broadband RF/IF & Digital Radio](#)  
[Clocks & Timers](#)  
[Data Converters](#)

[DLP & MEMS High-Reliability Interface](#)  
[Logic](#)  
[Power Management](#)

[Processors](#)

- [ARM Processors](#)
- [Digital Signal Processors \(DSP\)](#)
- [Microcontrollers \(MCU\)](#)
- [OMAP Applications Processors](#)

[Switches & Multiplexers](#)  
[Temperature Sensors & Control ICs](#)  
[Wireless Connectivity](#)

Retrieved from "https://processors.wiki.ti.com/index.php?title=WL18xx\_TI\_Bluetooth\_Stack\_HDSM\_Demo\_App&oldid=209314"

This page was last edited on 4 November 2015, at 12:24.

Content is available under [Creative Commons Attribution-ShareAlike](#) unless otherwise noted.