# TI Bluetooth Stack for WL18xx - Getting Started Guide

TI Bluetooth Stack - Demo Guide

## Contents

## Introduction

The following guide explains how to get started with TI Bluetooth Protocol Stack on a host device running Linux. It provides a basic description of the package, and walks through the download and installation procedures. Finally, it shows how to run a sample application. For users who wish to modify any sample applications or the build environment, please visit the **TI Bluetooth Stack - BluetopiaPM Build Process** page for further information. This does not have to be done to run the out of box AM335x Wilink8 Demo applications as the BluetopiaPM installation comes with pre-built binaries for Wilink8.

## Overview

### Bluetopia

Bluetopia is a low-level Bluetooth Protocol Stack. Bluetopia is designed to be small in code space and RAM so that it can be ran on small embedded platforms where memory is limited. The Bluetopia stack is limited in that only one application has access to the Bluetooth Controller at a time. This can make development difficult and complicated if multiple profiles are using the Bluetooth Controller concurrently.

### Bluetopia Platform Manager

Bluetopia Platform Manager (Bluetopia PM) is a service that allows multiple applications to access the Bluetooth link concurrently. This allows developers to create multiple independent applications that each use the Bluetooth link. Take for example a Head Set running Linux that uses the Hands-Free Profile and also supports A2DP Sink audio playback. Using Bluetopia a development team would have to write one application that handles both Hands-Free Events and A2DP Events. With Bluetopia PM the team could write two independent applications: one to handle Hands-Free Events and one to handle A2DP Events.

Bluetopia PM also provides features that simplify the management of device connections:

1. It stores pairing and link key information for each device connection in persistent memory
2. It automatically caches discovered device information such as the device name and class of the device
3. It automatically caches remote device service discovery information for quick access to a device's services without having to retrieve them again from the remote device

A convenient API is provided to access cached device information.

TI Bluetooth Stack (based on Bluetopia) is provided with the WL18xx and CC256x devices. It is composed of Bluetooth protocol stack, Bluetooth profiles stack and Platform Manager.

# Requirements

### Hardware

- 1x AM335x Evaluation Module (http://www.ti.com/tool/tmdxevm3358) or AM437x Evaluation Module (http://www.ti.com/tool/tmdsevm437x)
- For Wilink8: WL1837 (http://www.ti.com/tool/wl1837modcom8i?keyMatch=wilink%208&tisearch=Search-EN-Everything) or WL1835 (http://www.ti.com/tool/wl1835modcom8b?keyMatch=wl1835&tisearch=Search-EN)

- For CC256x: CC2564MODNEM (http://www.ti.com/tool/cc2564modnem) or CC2564MODNEM (http://www.ti.com/product/cc2564moda/description) or CC2564C QFN (http://www.ti.com/tool/cc256xcqfn-em)
- 1x RS-232 console wire to access CLI interface on the platform from PC
- Bluetooth remote devices (such as a phone or another EVM) that should support the above profiles, preferably with the opposite role for some cases, for demonstration purposes

**Note:** When using a CC256x controller you need to perform the following Hardware Modifications in order to make it compatible to the Sitara platforms.

- CC256X Configuration (http://processors.wiki.ti.com/index.php/SDMMC_Adapter_Configuration#CC256X_Configuration)

Note: Hardware requirements may depend on the application, specific demos will have detailed hardware configurations.

### Software

- AM335x Linux SDK (http://www.ti.com/tool/PROCESSOR-SDK-AM335X) or image for flashing onto an SD card
- Wilink 8 Bluetooth Add-On (http://www.ti.com/tool/TI-BT-4-2-STACK-LINUX-ADDON)
- TI EVM (AM335x, AM437x, AM57x) Bluetooth setup script can be downloaded from the following <zip file>
- PC Terminal software (e.g. Putty or TeraTerm).
- Most recently updated BTS script from WL18xx-BT-SP (http://www.ti.com/tool/WL18xx-BT-SP) (see note below).

**Note:** Download the latest WL18xx Bluetooth Firmware Script (BTS script) from: WL18xx-BT-SP (http://www.ti.com/tool/WL18xx-BT-SP). Read the license agreement and follow the installation instructions in the previous link. The script will be automatically be downloaded to the WL18xx device whenever the stack power up the device.
**Note:** When using a CC256x controller you will need to copy the init scripts. Follow the instructions in CC256x Initscripts (http://processors.wiki.ti.com/index.php/TI_Bluetooth_Stack_for_WL18xx_-_Build_Process#CC256x_Init-scripts)

# Getting Started

### Step 1. Installing Bluetopia Platform Manager

TI Bluetooth Stack (based on Bluetopia) is provided with the WL18xx or CC256x devices. It is composed of Bluetooth protocol stack, Bluetooth profiles stack and Platform Manager. The Bluetooth Add-On Packages and can be re-built and used with any Linux and WL18xx or CC256x platform but for convenience for those using an AM335x standard SDK the following pre-built packages are available for download. The stack is not part of platform SDK but is available as an add-on which can be downloaded from the Software Requirements (http://processors.wiki.ti.com/index.php/TI_Bluetooth_Stack_for_WL18xx_-_Getting_Started_Guide#Software) section.

The package is downloaded as tarball and should be extracted (in a Linux machine).

```
$  AM335xBluetopiaLinuxProduction-4.2.1.0.1.0-linux-installer.tar.gz
```

The tarball includes 2 Linux installers: for 32b and 64b release. Please use the right one according to your working station type and execute it (the following demonstrates the 64b case):

```
$ chmod +x ./AM335xBluetopiaLinuxProduction-4.2.1.0.1.0-linux-x64-installer.run
$ ./AM335xBluetopiaLinuxProduction-4.2.1.0.1.0-linux-x64-installer.run
```

When the installer completes, the installation directory should include Bluetopia Platform Manager (BluetopiaPM) Binaries, Sources and Documentation.

- Binaries - includes pre-compiled sample applications, BluetopiaPM server daemon and BluetopiaPM Client libraries.
- Sources - platform specific files (to enable recompiling for different platforms) and sample applications code.
- Documentation - License docs, users guides and API specifications.
- The installation directory can be updated during the installer execution. The default is

```
~/AM335xBluetopiaLinuxProduction-4.2.1.0.1.0/
```

- For more details on the distribution content, please refer to the Bluetopia Platform Manager Architecture section.

### Step 2. SD card creation

Once the SDK is installed and compiled and the Bluetooth Add-on is installed, the SD card needs to be created. This can be achieved by using a script to create the SD card (http://processors.wiki.ti.com/index.php/Processor_SDK_Linux_create_SD_card_script).

### Build and export the Bluetopia PM

Once the SD card is ready, You will need to build the Add-on and copy the files to your SD-card. Source code for each sample can be found in the [install-dir]/BluetopiaPM/sample directory. To modify and re-build the samples follow the instructions in Build the Bluetopia PM (http://processors.wiki.ti.com/index.php/TI_Bluetooth_Stack_for_WL18xx_-_Build_Process)

**Note:** If using the WL183x there are pre-built applications that are ready to run out of the box.

Adding the Bluetopia files to the SD card requires copying over 4 main components: the Sample Applications, Gstreamer Plugin, VNET Plugin, and Init-scripts. Follow the instructions below in order to install the Bluetopia Stack into the imaged SD card.

#### Step 3. Sample Applications

Copy the BluetopiaPM server (SS1BTPM) and the provided Sample applications that were built

```
sudo mkdir [target-root]/home/root/BluetopiaPM
sudo cp -rf <INSTALLATION_DIR>/BluetopiaPM/bin [target-root]/home/root/BluetopiaPM/
```

#### Gstreamer Plugin

BluetopiaPM include a GStreamer Plugin that should be copied to the target rootfs, The GStreamer is being used in **AUD Demo** application

```
sudo cp <INSTALLATION_DIR>/BluetopiaPM/lib/libgstss1bluetooth.so [target-root]/usr/lib/gstreamer-1.0/
```

**Note:** For 4.2 releases please see the release notes as this feature not included in the release. If needed, you can obtain the plugin from the 4.0 release available from the 4.0 Download Page (http://www.ti.com/tool/TI-BT-STACK-LINUX-ADDON)

#### VNET Plugin

BluetopiaPM include a VNET Plugin that should be copied to the target rootfs, The VNET is being used in **PAN Demo** application

```
sudo mkdir -p [target-root]/home/root/BluetopiaPM/Bluetopia/VNET/driver
sudo cp -rf <INSTALLATION_DIR>/BluetopiaPM/Bluetopia/VNET/driver/VNETInst [target-root]/home/root/BluetopiaPM/Bluetopia/VNET/driver/
sudo cp -rf <INSTALLATION_DIR>/BluetopiaPM/Bluetopia/VNET/*.ko [target-root]/home/root/BluetopiaPM/Bluetopia/VNET/
```

#### Voice Over BLE Python Script

The **VoLE Demo** application utilizes a python script in order to decode, format and analyze the audio data.

```
sudo cp <INSTALLATION_DIR>/BluetopiaPM/sample/LinuxVoLE/audio_frame_linux.py [target-root]/home/root/BluetopiaPM/bin
```

#### Bluetooth Hardware configuration script

The bt-en.sh script, downloaded from the Software Requirements (http://processors.wiki.ti.com/index.php/TI_Bluetooth_Stack_for_WL18xx_-_Getting_Started_Guide#Software%7C) section which creates a custom configuration file based on the board you're using.

**Note:** If using a Beaglebone see the page on patching the device tree (http://processors.wiki.ti.com/index.php/WL18xx_Platform_Integration_Guide#Appendix_A:_Device_Tree_file_-_patching_and_updating) and **skip** the bt-en.sh setup script steps below.

First unzip the file Downloaded file and then copy it to the target platform:

```
sudo cp Downloads/bt-en.sh [target-root]/home/root/BluetopiaPM/
```

Navigate to the correct directory on the target platform:

```
 cd /home/root/BluetopiaPM/
```

Update the scripts permissions:

```
chmod +x bt-en.sh
```

Then run the script:

```
./bt-en.sh
```

When using a custom platform, you'll need to create a similar script with your platform specific settings. As an example, you'll have to update the following variables:

- **tty**

and

- **gpio**

Note, this includes the:

```
 echo gpio
```

statement at the start of the script

Ex:

1. Create a new script file:

```
vi bt-en.sh
```

2. Copy the following into the empty file:

```
echo 16 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio16/direction
echo 1 > /sys/class/gpio/gpio16/value
echo "Done enabling BT"

gpio="nshutdown_gpio=16"
tty="tty=/dev/ttyS3"
flow="flow_cntrl=1"
baud_rate="baud_rate=3000000"
mkdir /home/root/tibt
echo $gpio > /home/root/tibt/config
echo $tty >> /home/root/tibt/config
echo $flow >> /home/root/tibt/config
echo $baud_rate >> /home/root/tibt/config
```

3. Update the **gpio** and **tty** variables to match your platform, in this example we're using GPIO 16 and tty S3.

4. Update the scripts permissions:

```
chmod +x bt-en.sh
```

5. Run the script:

```
./bt-en.sh
```

### Step 4. Installing updated Firmware Service Pack (Init-scripts)

Before the stack can be used make sure to have an updated Bluetooth Script (BTS) on your target platform. The latest BTS file and installation instructions can be found in WL18xx-BT-SP (http://www.ti.com/tool/WL18xx-BT-SP) and can be copied directly to the SD card's firmware directory as shown below:

```
sudo cp -rf TIInit_X.Y.Z.bts [target-root]/lib/firmware/
```

In order to add support for **AVPR**, **BLE** or **BR/EDR** for the CC256x devices the CC256x Service Pack User's Guide (http://processors.wiki.ti.com/index.php/CC256x_Service_Pack_User%27s_Guide) should be followed.

## Step 5. Terminal Connection

Once the code is copied. Plug in the RS-232 0 port to a pc via NULL modem and then look at the Device manager for Communications Port (COM x) under Ports (COM & LPT). Attach a terminal program like PuTTY to the serial port (COM x) for the board, x means which COM is open for Communications Port in Device Manager. The serial parameter to use is 115200 Baud rate. Once connected, Turn on the device Power button and you should see the am335x getting initialized on the terminal.

```
Arago Project http://arago-project.org am335x-evm /dev/ttyS0
Arago 2015.05 am335x-evm /dev/ttyS0
am335x-evm login:
```

When the initialization is over, type root in order to use the file system.

**Note:** The serial port may differ for different platforms. E.g. AM437x will be **/dev/ttyS1**

```
am335x-evm login: root
root@am335x-evm:~#
```

### Step 6. Running a Sample Application

Now that you have installed Platform Manager to your host platform and copied relevant binaries (for this example at least SS1BTPM and LinuxSPPM should be copied) and init-scripts to your target, let's demonstrate running a Platform Manager application. The following should be done on the target (using USB or Ethernet connection):

```
# cd /home/root/BluetopiaPM/bin
# ./SS1BTPM &
# ./LinuxSPPM
```

You should now see LinuxSPPM's menu displayed in the terminal. Now let's try to initialize the Bluetopia PM Server in LinuxSPPM and turn on the Bluetooth Controller.

```
SPPM>Initialize 1
SPPM>SetDevicePower 1
```

If all went well you should see a "Device Powered On" message in the terminal. Congratulations, you have just run your first Platform Manager application! For more detailed information on using the LinuxSPPM Demo and for other Demo Guides, refer to Bluetopia Platform Manager Demo Guide. Next we'll go over the basics of Bluetopia Platform Manager's sample applications,

### Bluetopia PM Samples

Samples are provided with the Bluetopia Platform Manager distribution to demonstrate how to use the Bluetopia PM modules. Each sample provides a command-line interface which can be used to send commands to the PM Server. The sample application commands encapsulate Bluetopia API commands and demonstrate how to use common API functions. The samples also display events and event data when an event is received from the PM Server.

The samples are provided to help the end user get started with application development in PM. The samples are not ready-to-use production applications, as an application like this depends on the project-specific requirements. However, the samples can be used as a starting point for application development. We'll now go over re-building a sample which will help with getting started with application development in PM.

For more information on how to use a specific sample, also referred to as a demo, refer to the Bluetopia Platform Manager Demo Guide.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| {{<br>1. switchcategory:MultiCore=<br>- For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum<br>- For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum<br>Please post only comments related to the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here. | Keystone=<br>- For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum<br>- For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum<br>Please post only comments related to the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here. | C2000=*For technical support on the C2000 please post your questions on The C2000 Forum. Please post only comments about the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here.* | DaVinci=*For technical support on DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here.* | MSP430=*For technical support on MSP430 please post your questions on The MSP430 Forum. Please post only comments about the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here.* | OMAP35x=*For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here.* | OMAPL1=*For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here.* | MAVRK=*For technical support on MAVRK please post your questions on The MAVRK Toolbox Forum. Please post only comments about the article **TI Bluetooth Stack for WL18xx - Getting Started Guide** here.* | *For technical s please post yo questions at http://e2e.ti.co Please post on comments abo article **TI Bluet Stack for WL1 Getting Starte Guide** here.*<br>}} |

## 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=TI_Bluetooth_Stack_for_WL18xx_-_Getting_Started_Guide&oldid=226535"

**This page was last edited on 9 April 2017, at 19:41.**