TI 15.4-Stack Linux® Gateway Example Application

Contents

List of Figures

List of Tables

Trademarks

LaunchPad, BoosterPack, SmartRF are trademarks of Texas Instruments.
UBUNTU is a registered trademark of Canonical.
Linux is a registered trademark of Linus Torvalds.
Windows is a trademark of Microsoft Corporation.
Wi-Fi is a trademark of Wi-Fi Alliance.
All other trademarks are the property of their respective owners.

1 Introduction

This quick start guide provides instructions for developers to begin using the out-of-box TI 15.4-Stack Linux® SDK Example Application.

2 Background

The TI 15.4-Stack Linux SDK example applications helps developers create ultra-low power, very long-range, star-topology network solutions. The TI 15.4-Stack Linux SDK includes the Collector and Gateway Example Applications (in addition to others, see the TI 15.4-Stack Embedded Developer's Guide for more details). The Linux Collector Example Application interfaces with the CC13x0 running the MAC CoProcessor (CoP) using a UART. The Collector Example Application builds a full-function device which performs the functions of a network coordinator (starting a network and permitting devices to join that network), and also provides an application to monitor and collect sensor data from one or more sensor devices. In addition, the Gateway Example Application provides a socket server interface to the Linux Gateway Application.

The Linux Gateway Application, implemented within the NodeJs framework, connects as a client to the socket server created by the Linux Collector. In addition, the Linux Gateway Application establishes a local web server, to which the user can connect through a web browser, to monitor and control the network devices. The Collector and Gateway Example Applications, which provides Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 to the IP Bridge, is a great starting point to create Internet of Things (IoT) applications with the TI 15.4-Stack.

NOTE: In the following sections, the project names for CC1310 and CC1350 platforms are referred to as CC13x0. Replace *x* with either *1* or *5* depending on the wireless MCU being used.

3 Supported Hardware Combinations

The out-of-box Linux Collector and Gateway Example Applications can be run using either of two host hardware combinations.

Hardware Combination 1

The application runs on BeagleBone Black (BBB). It uses BBB and CC13x0LP running the MAC CoP. BBB runs the Collector and Gateway Applications while the CC13x0 runs the MAC CoP Application.

- Pros
  - Once set up, only a web browser is required.
  - No Linux machine is required right away.
- Cons
  - Additional steps are required to set up the BBB.

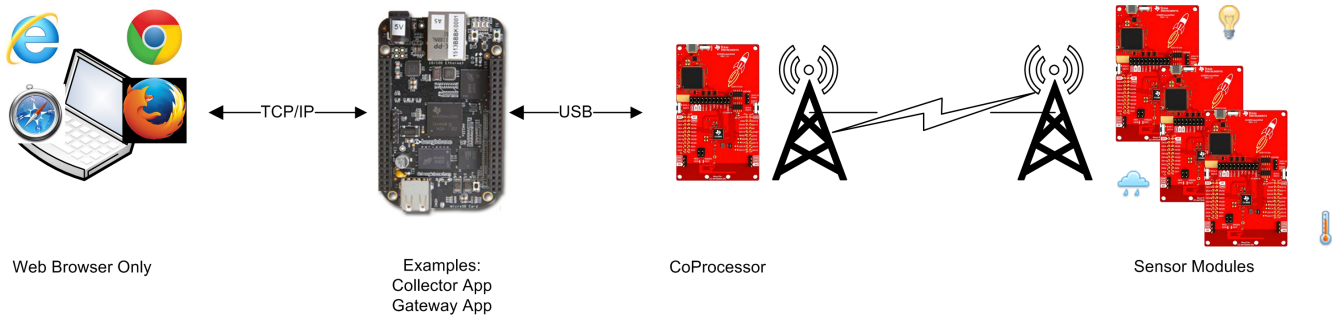

Figure 1. Hardware Combination 1 Overview

Hardware Combination 2

The Collector and Gateway Example Applications run on the Linux machine running UBUNTU®. The x86 machine interfaces to the CC13x0 LaunchPad running the MAC CoP Application.

- Pros
  - Easy to set up
  - No BBB is required; instead, all example applications run on the Linux development host machine
- Cons
  - Requires a Linux machine
  - Does not demonstrate the scalability of the example application
- Important Linux host requirements
  - Prebuilt Linux binaries assume: x86_64 machine running UBUNTU 14.04 64-bit LTS

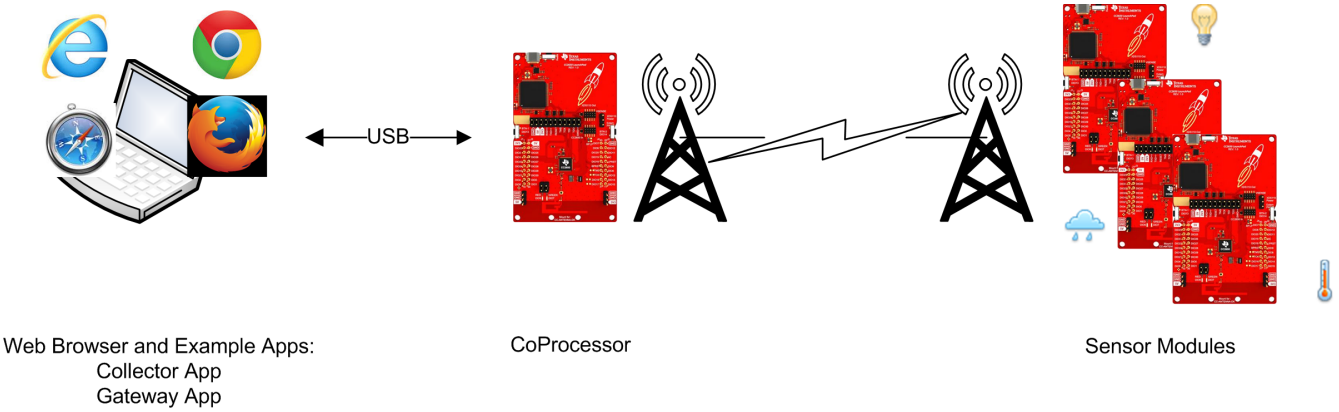

Figure 2. Hardware Combination 2 Overview

Common to Hardware Combination 1 and 2

The network devices – at least two of the CC13x0 LaunchPad™ (LP) development kits. LP1 runs the prebuilt MAC CoP Application, and LP2 (to n) acts as sensor nodes running the prebuilt Sensor Application.

4 Required Hardware

Embedded Devices (common):

- Two CC1310 LPs or two CC1350 LPs
- One LCD BoosterPack™ – optional (http://www.ti.com/tool/430boost-sharp96)

Hardware Combination 1 – BBB host machine (running the prebuilt applications directly on the BBB):

- BBB (https://beagleboard.org/black)
- 8-GB SD card (the TI processor SDK image requires at least 8-GB of space)
- A means to configure and set up the BBB SD card (Windows™ or Linux machine)
- A PC to host and run the web browser used to view the example application
- In some configurations, a standard Wi-Fi™ router may be required (see Figure 6 for details)

Hardware Combination 2 – Linux host machine (running the prebuilt applications on a Linux x86 development host):

- UBUNTU 14.04 LTS 64-bit Linux machine

5 Programming the CC13x0 LaunchPads

To run the example application users must first program one CC13x0 LP with the MAC CoP hex file and the other LP(s) with the Sensor Example Application hex file. There are two ways to program the CC13x0 with the desired hex files. In this quick start guide, the Flash Programmer 2 tool running on a Windows machine is used. Developers can also use the Serial Flash Programmer tool, described in the TI 15.4-Stack Developers Guide at /doc under the TI 15.4-Stack Linux installation directory, to program the desired hex image onto the CC13x0LP.

NOTE: It is easy to confuse the sensor and CoP devices. Be sure to label the devices as they are programmed.

To program the LPs, follow these steps.

1. Download and install the SimpleLink CC13x0 SDK at http://www.ti.com/tool/SIMPLELINK-CC13X0-SDK.
2. Download and install the SmartRF™ Flash Programmer 2 at http://www.ti.com/tool/flash-programmer.
3. Program CC13x0 LP 1 – this device runs the CoP example application.
   (a) Label this device collector. The LCD BoosterPack is not supported in the CoP application.
   (b) From a Windows PC, use the SmartRF Flash Programmer 2 to program a CC1310 LP or CC1350 LP with coprocessor_cc1310lp.hex or coprocessor_cc1350lp.hex.
   - For the CC1310 LP the hex file is at C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\rtos\CC1310_LAUNCHXL\ti154stack\hexfiles
   - For the CC1350 LP the hex file is at C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\rtos\CC1350_LAUNCHXL\ti154stack\hexfiles
4. Program CC13x0 LP 2 – this device runs the Sensor Example Application.
   (a) Label this device sensor. Optional: connect the LCD BoosterPack to this LP.
   (b) From a Windows PC, use the SmartRF Flash Programmer 2 to program a CC1310 LP or CC1350 LP with sensor_default.hex.
   - For the CC1310 LP the hex file is at C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\rtos\CC1310_LAUNCHXL\ti154stack\hexfiles\default
   - For the CC1350 LP the hex file is at C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\rtos\CC1350_LAUNCHXL\ti154stack\hexfiles\default

NOTE: Important – the default hex files are built for 915-MHz band operation. To rebuild the hex files for other bands (for example, 868 MHz ETSI band) see the following:

- The Embedded Developer's Guide or Linux Quick Start Guide
- The Linux SDK Developer's Guide, specifically the Example Collector Application configuration section, to change the Linux example application.

Troubleshooting – if the devices (sensor or CoP) get mixed up, use the Flash Programmer 2 tool to verify the flash content.

- Uncheck the ERASE option, uncheck the PROGRAM option, and only enable the VERIFY option along with the read-back feature, to double-check or double-verify the flash operation.

6 Software Setup

Depending on the hardware setup combination used, follow the steps in Section 6.1 or Section 6.2 respectively, to get the desired software setup to run the applications as described in Section 8.

6.1 Hardware Combination 1

This section describes the steps required when running the application using hardware combination 1.

6.1.1 BBB SD Card Image

Program the SD card with the processor SDK image using the following steps:

1. Download the prebuilt TI processor SDK SD card image am335x-evm-linux-03.01.00.06.img.zip from http://software-dl.ti.com/processor-sdk-linux/esd/AM335X/latest/index_FDS.html
2. Follow the instructions on the wiki page to program the microSD memory card.

- Using a Windows machine: http://processors.wiki.ti.com/index.php/Processor_SDK_Linux_Creating_a_SD_Card_with_Windows
- Using a Linux machine: http://processors.wiki.ti.com/index.php/Processor_SDK_Linux_create_SD_card_script

6.1.2 Booting the BBB

Boot the BBB from the SD card using the following steps:

1. Disconnect power and unplug the USB cable from the BeagleBoard.
2. Insert the SD card into the BBB (see Figure 3).
3. Press (and hold) the Boot Switch. **Important:** The boot switch is detected only at initial power on.
4. Provide power to the BBB (1.5 A, 5 V).
5. Wait a few seconds, then release the Boot Switch. In about 5 to 15 seconds the LEDs begin to blink.

NOTE: The first boot from a freshly formatted SD card takes about 1 to 2 minutes longer, during this extended time the BBB Linux distribution performs some one-time-only steps.

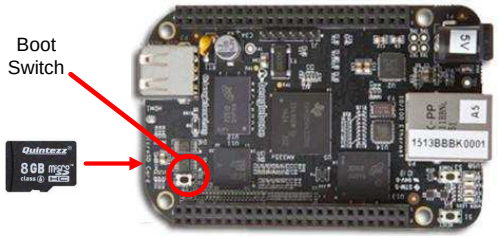

Figure 3. Boot BBB from SD Card

6.1.3 Determining the BBB Network Address

There are four possible network permutations: (USB or Network cable) × (Windows or Linux) = 4. Choose the combination that applies to your environment. All four combinations are shown as follows:

- **Connect the BBB to the Linux Host using USB** – use the TCP/IP over the USB cable (also known as Linux USB Gadget Interface). The USB BBB configuration always uses IP address 192.168.7.2 for the BBB (this address is hard coded in the TI processor SDK SD card image).

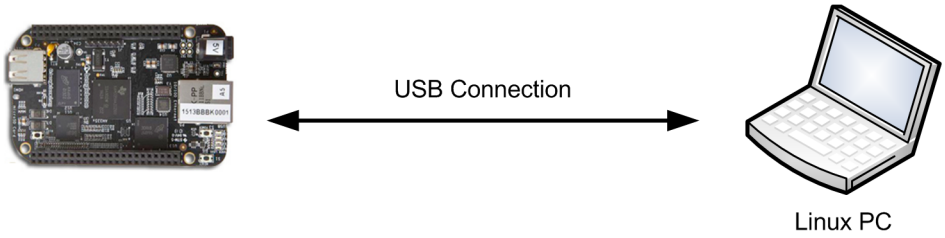

Figure 4. BBB to Linux Host Connection

- **Connect the BBB to the Windows Host using USB** – this method is currently unsupported (due to Windows drivers).

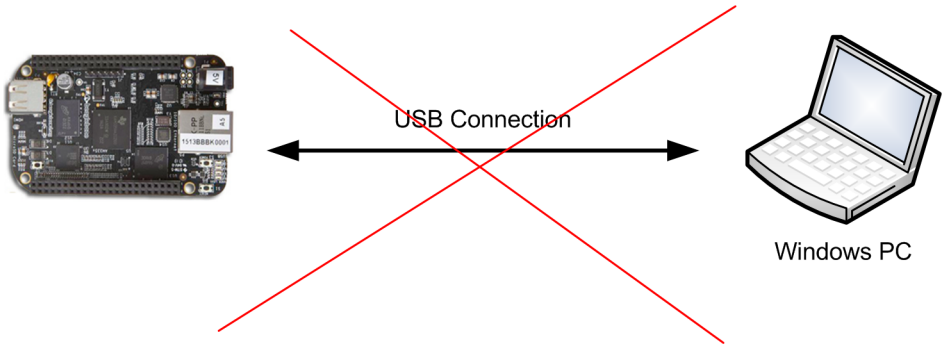

Figure 5. BBB to Windows Host Connection

- **Connect the BBB with a Network Router and Using a Linux or Windows Host** – this method is supported.

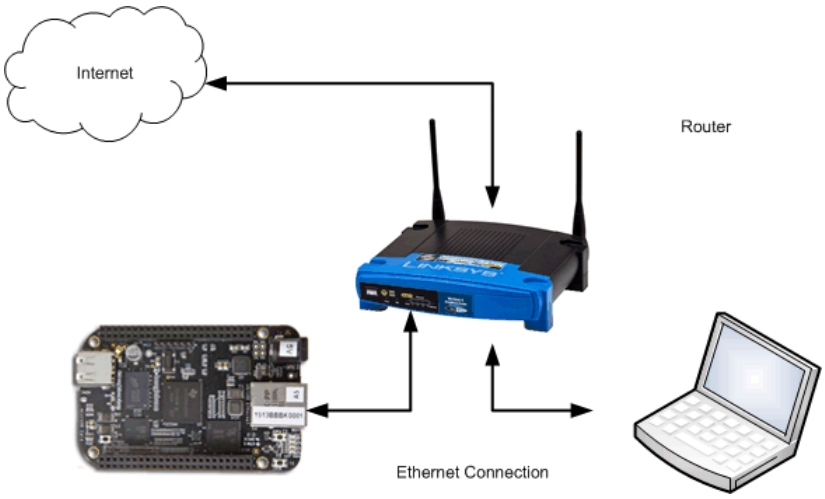

Figure 6. BBB to Linux or Windows Host Connection Using Network Router

In this configuration, the IP address of the BBB can be determined in two ways.

– Method 1: Use the FTDI cable to connect through the serial header on the BBB, and use ifconfig to determine the IP address allocated to the BBB.
– Method 2: Most routers include a built-in web server to configure the device (see Table 1).

1. Connect the BBB to the router.
2. Boot the BBB.
3. Find the DHCP Client page, to determine the IP address of the BBB. Some examples follow. The generic name for this feature is the DHCP Client Table.

NOTE: Troubleshooting – the DHCP IP address is often determined by the order in which the devices boot. If the user's laptop booted first, it may receive address: xx.xx.xx.100. The BBB boots second, and it receives the address: xx.xx.xx.101 – however, on the next use, or if another device is attached (for example a cell phone or tablet) the resulting boot order may change, and therefore the IP address might change.

Table 1. Commercial Routers

Brand	Example Link
LinkSys	http://www.linksys.com/us/support-article?articleNum=139502
NetGear	http://documentation.netgear.com/fvs336g/enu/202-10257-01/FVS336G_RM-11-07.html
Belkin	http://www.belkin.com/pyramid/AdvancedInfo/F5D8235-4/Advance/reserveIP.htm

6.1.4 Copying the Prebuilt Files to the BBB Using the Secure Copy

Copy the bbb_prebuilt.tar.gz file to the BBB using the secure copy (SCP), note "xx" in the SDK name is the final release installer number.

Linux prebuilt files are at ${HOME}/ti/simplelink/ti15.4stack_linux_64_02_00_01_xx/prebuilt

On Linux, the command is (recall that ${SDK_ROOT} is the Linux SDK installation directory, substitute the appropriate address for ${BBB_IP_ADDRESS}):

```
bash$ cd ${SDK_ROOT}/prebuilt
bash$ scp bbb_prebuilt.tar.gz root@${BBB_IP_ADDRESS}:~/.
```

In the previous command, the final: ~/. is a short-hand notation for the home directory of the user. In this specific case the directory is /home/root

6.1.5 Logging into the BBB Using the Secure Shell (Get a Shell Prompt)

1. Login to the BBB using the secure shell (SSH): bash$ ssh root@${BBB_IP_ADDRESS}
   This command connects to the BBB and provides the root (#) prompt, specifically: root@am335x-em#
2. Unpack the TAR file.
   root@am335x-evm# cd ${HOME} (where the tar file is located)
   root@am335x-evm# tar xf bbb_prebuilt.tar.gz
3. The prebuilt binaries are in the prebuilt directory, specifically: /home/root/prebuilt

6.2 Hardware Combination 2

This section describes the steps required when running the application using hardware combination 2.

6.2.1 Linux Host Software Setup

1. Download the TI-15.4 Stack Linux SDK Installer at http://www.ti.com/tool/SIMPLELINK-CC13X0-SDK to the x86 machine running UBUNTU OS (64-bit, Version 14.04 LTS).
2. Install the TI 15.4-Stack Linux SDK by going to the directory to which the file was downloaded. Note, the default TI 15.4-Stack install directory is ${HOME}/ti/simplelink/ti15.4stack_linux_64_02_00_01_xx

3. Execute the following commands as a normal user (do not perform these steps as the root user). Note xx is the build sequence number and can be ignored.

```
bash$ cd ${where_the_run_file_is_located}
bash$ chmod +x ti15.4stack_linux_x64_02_00_00_xx.run
bash$ ./ti15.4stack_linux_x64_02_00_00_xx.run
```

4. If needed do bash$ sudo apt-get update. Note the font for the line bash$ sudo apt-get update must appear like all the bash$ lines used in this section.
5. Install the package: build essentials.

```
bash$ sudo apt-get install build-essential
```

6. Install the package: NodeJS.

```
bash$ sudo apt-get install nodejs
```

7. Type in a user name which must be a member of the group dialout.

```
bash$ sudo adduser $USER dialout
```

8. Optional (can be done later): to cross compile for BBB, install the TI processor-SDK-Linux-AM335x from http://www.ti.com/tool/PROCESSOR-SDK-AM335X. The TI-AM335x-Linux SDK contains all of the cross compilation tools, headers, libraries, and other required files for cross compiling to the BBB. See the TI 15.4-Stack Linux Developer's guide for more details.

NOTE: The prebuilt Linux applications are in the ${SDK_ROOT}/prebuilt directory.

6.2.2 Linux Host Prebuilt Directory

Troubleshooting note: all of these steps are performed as a normal user. Do not use the root user to perform these steps.

1. Change to the SDK Installation Directory.

```
bash$ cd ${SDK_ROOT}
```

2. Change to the Linux Prebuilt Directory.

```
bash$ cd prebuilt
```

7 Running the Application

7.1 Connecting the CoProcessor LaunchPad

Plug the CC13x0 running the CoP into the x86 machine or the BBB.

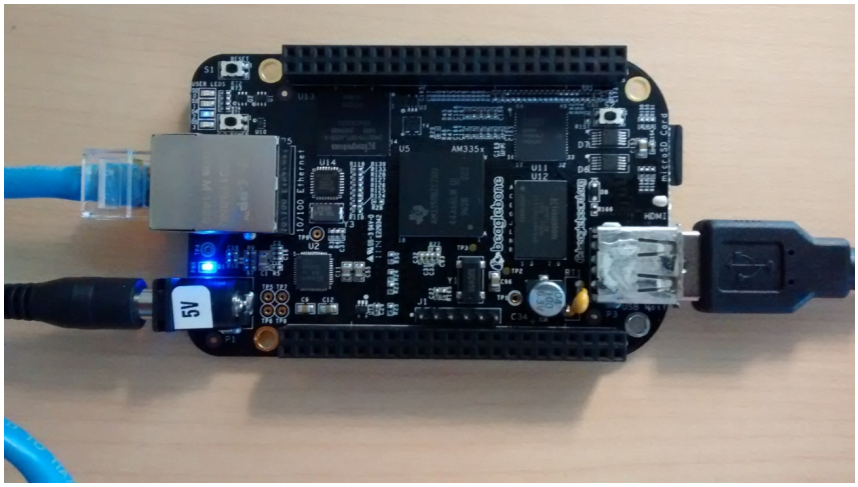

Figure 7. CoProcessor LaunchPad Connected to BBB

7.1.1 CoProcessor LaunchPad Troubleshooting

- Check for the /dev/ttyACM0 device using this command:

```
bash$ ls –l /dev/ttyACM*
```

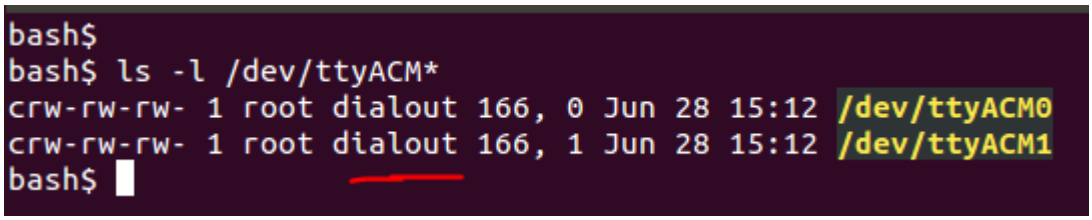

Figure 8. /dev/ttyACM0 Device Check

The LP presents as two USB serial ports named: /dev/ttyACM<somenumber> Typically these are ACM0 and ACM1, and they are members of the group dialout. The Collector Example Application uses a configuration file located at: ${PREBUILT}/bin/collector.cfg

The default configuration file assumes the LP is: /dev/ttyACM0

Sometimes the simplest solution is to edit or change the ${PREBUILT}/bin/collector.cfg file.

- Multiple /dev/ttyACM Devices (Reason 1)

  In some cases, an application may have crashed, or the LP USB cable was unplugged and plugged back in. This occurrence can cause the application to hold on to a reference to an existing /dev/ttyACMx device name. When the USB device was removed and reinserted, the old name was still in use. Linux then uses the next available number (for example, /dev/ttyACM2 or /dev/ttyACM3 … and so on). Another example is closing your laptop lid or suspending your Virtual Machine and later resuming. The easiest solution is to reboot the machine (the other method is to kill various process through kill).

- Multiple /dev/ttyACM Devices (Reason 2)

  For the BBB (hardware combination 1) the LP is typically the only device that is present, and therefore, the LP generally appears as /dev/ttyACM0 and /dev/ttyACM1. For the Linux x86 (hardware combination 2) it is common to have multiple devices plugged into your Linux development host machine, and these other devices may also present a /dev/ttyACM<somenumber>. For example the BBB, when connected through USB to the user's Linux machine, can be configured to present an /dev/ttyACM<somenumber>.

  If required, a simple solution is to edit the ${PREBUILT}/bin/collector.cfg file (see Figure 9).

```
[uart-cfg]
        ;; Launchpads use USB and show up as: /dev/ttyACM0 and ACM1
        ;; Solutions using an FTDI or Prolific cable use /dev/ttyUSB0 or USB1
        ;; Hard serial ports are: /dev/ttyS0 to ttyS9
        ;devname = /dev/ttyUSB1
        devname = /dev/ttyACM0
        baudrate = 115200
        ; we use the default flags
        flag = default
```

Figure 9. UART Configuration

- Wrong Permissions

  For the BBB (hardware combination 1), the application generally runs as root, therefore there is never a permission problem. For the Linux x86 (hardware combination 2), the application normally runs as a normal user, so it is easy to overlook the adduser step.

```
bash$ sudo adduser $USER dialout
```

  Remember that after adding the dialout group, the change does not propagate to other existing shell windows automatically. The easiest solution is to reboot the machine and log in again.

7.2 Starting the Application – Start the Network

The PREBUILT directory contains a simple shell script called run_demo.sh

```
cd ${PREBUILT}        (change to the prebuilt directory)
bash ./run_demo.sh     (run the demo)
```

BBB (hardware combination 1): the shell script prints the URL for your browser.

1. Select the appropriate URL.
2. Cut and paste the URL into your browser (see the following troubleshooting for more details).

The Linux x86 (hardware combination 2): the script launches your browser automatically.

7.2.1 Network Troubleshooting

- Error Messages

  The application should start the applications in the background, and eventually return to the bash prompt. During this process the software prints the software version of the CoP Application. If this action does not occur, or if there are other error messages printed, review the troubleshooting steps in Section 7.1.1.

- Which URL to use?

  The BBB (hardware combination 1) prints two different URL addresses. The 192.168.7.2 URL is specific to the Linux only USB interface. The other URL is for the wired interface, and should be the ${BBB_IP_ADDRESS} previously used.

- Old network information?

  Clear nonvolatile memory to start a network fresh. To do this delete nv-simulation.bin from: ${prebuilt}/bin

- Wrong network configuration

  You can change network configurations such as PAN ID, Beacon Order, Channel etc. in collector.cfg found in: ${prebuillt}/bin/collector.cfg.

7.3 The Gateway Application Web Page – Open Network for New Device Joins

Figure 10 shows the example web page. Initially, the example application starts with no devices present, the network is closed, and it will not accept new devices.

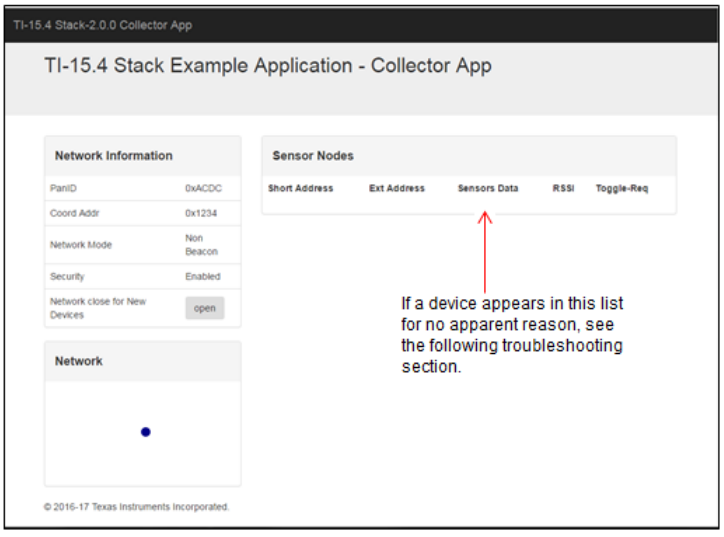

Figure 10. TI-15.4 MAC Gateway Application Web-Application Served by the Local Web Server After Network Startup

7.3.1 Gateway Application Web Page Troubleshooting

- About the Port Number

BBB (hardware combination 1): The default web server service is on port 80. The BBB is running two demos; demo 1 is on the default service port (port 80), and this is a demo for the processor SDK package.

Both hardware combination 1 and 2: The TI 15.4-Stack Linux SDK example gateway application uses port 1310, therefore be sure to use :1310 in the URL used to connect to the gateway. For example, the URLs would be as follows in Table 2.

Table 2. IP Addresses for Connecting to the Web-Server

IP Address	Hardware Combination
http://192.168.7.2:1310	Hardware combination 1 – BeagleBone using USB interface
http://${BBB_IP_ADDRESS}:1310	Hardware combination 1 – BeagleBone using wired interface
http://localhost:1310	Hardware combination 2 – Linux only
http://${LINUX_IP_ADDRESS}:1310	Hardware combination 2 – Linux only

- Device List and Restoring Previous State

The first time the application runs there are no devices in the Device list. If the Linux example application is run a second time, the Collector Application restores to the previous state by reading the file: ${PREBUILT}/bin/nv-simulation.bin. Therefore, sensor devices may initially appear in the list of devices, even though it is not present or powered off.

- To Reset the Devices (Linux Collector Application and Sensor Application)

To reset the Linux Collector Application, remove the nv-simulation.bin file and restart the demo application. The same restore operation applies to the example embedded Sensor Applications. If they are reset or power-cycled they resume their previous state. To reset the embedded Sensor Application on the sensor LP:

1. Press and hold both BTN1 and BTN2.
2. Press and release the RESET button.
3. Release both BTN1 and BTN2.

7.4 *Joining the Devices to the Network*

At start up, the Collector Example Application initially has the network closed; therefore sensor devices cannot join. To open the network, click the Open button on the web browser. Within a few seconds (time depends on the polling interval and other configuration settings) the sensor joins the network. When the device joins the network, the red LED turns on. If the sensor LP has an LCD module, it indicates the current state on the LCD (see Figure 11). More details can be found in the embedded documentation.

- State 1 = Not joined
- State 3 = Joined
- State 4 = Restored
- State 5 = Orphan condition

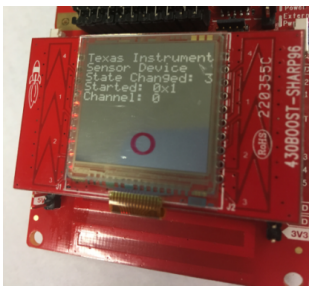

Figure 11. Sensor LaunchPad State Change LCD

7.5 Data Communication

After the new device appears, initially only the short and extended addresses appear. The data fields are displayed as --- (hyphens) indicating no data.

Sensor Data Reports

After about 1-2 minutes data appears on the screen (the exact interval is configured in the Collector application using a #define value), see the TI 15.4-Stack Embedded Developer's Guide or the Linux Example Collector source code for more details. After this time, the sensor nodes periodically report the sensor data.

Actuation

Clicking the Toggle LED button sends a message to the sensor module to toggle the LED. A slight delay in toggle operation may occur (a few seconds) on the desired sensor LP. This delay is because the sensor nodes are in sleep mode, and only wake up periodically to receive the command buffered on the collector.

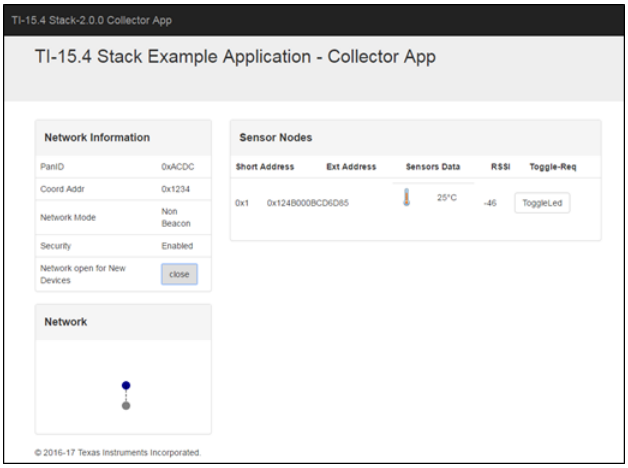

Figure 12. TI 15.4-Stack Gateway Application Web Application After Device Joins Network

NOTE: To run the example from the source, refer to section 4.2.1 in the TI 15.4-Stack Linux Developer's Guide.

8 Next Steps

For additional information, see the TI 15.4-Stack Linux Developer's Guide installed with the TI 15.4-Stack Linux SDK. For details on the stack operation, setting up the packet sniffer, and more, see TI 15.4-Stack Developer's Guide installed with the TI 15.4-Stack Windows Installer. Developers can also refer to the other documents included with the TI 15.4-Stack SDK installation.

Useful links

- Find answers to your questions and common issues, post your questions, and answer questions from other developers at the TI e2e forums. http://e2e.ti.com/support/wireless_connectivity/proprietary_sub_1_ghz_simpliciti/
- TI 15.4-Stack Wiki page

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Original (July) to A Revision **Page**

- Changed text from x86_64 machine to Linux machine. 3
- Changed x86 to x86_64. 3
- Updated file location from C:\ti\simplelink\ti-15.4-stack-sdk_2_00_00_xx\examples\hexfiles to C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\os\CC1310_LAUNCHXL\154stack\hexfiles and C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\os\CC1350_LAUNCHXL\154stack\hexfiles. 5
- Updated file location from C:\ti\simplelink\ti-15.4-stack-sdk_2_00_00_xx\examples\hexfiles to C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\os\CC1310_LAUNCHXL\154stack\hexfiles\default and C:\ti\simplelink_cc13x0_sdk_1_00_00_00\examples\os\CC1350_LAUNCHXL\154stack\hexfiles\default. 5
- Updated from am335x-evm-linux-02.00.02.11.img.zip to am335x-evm-linux-03.01.00.06.img.zip. 6
- Updated location from ${HOME}/ti/simplelink/ti15.4stack_linux_64_02_00_00_xx/prebuilt to ${HOME}/ti/simplelink/ti15.4stack_linux_64_02_00_01_xx/prebuilt 8
- Updated directory from ${HOME}/ti/simplelink/ti15.4stack_linux_64_02_00_00_xx to ${HOME}/ti/simplelink/ti15.4stack_linux_64_02_00_01_xx. 8
- Added content to Network Troubleshooting (Section 7.2.1) 11
- Added note to Section 7.5. 13
- Added TI wiki page link. 13

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ('TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265