Keysight U3810A Advanced IoT Teaching Solution

U3813A/14A IoT System Design and Validation Fundamentals

Lab 4: Introduction to Zigbee Communications

Lab Sheet

KEYSIGHT TECHNOLOGIES

Notices

Copyright Notice

© Keysight Technologies 2020

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Keysight Technologies as governed by United States and international copyright laws.

Edition

Edition 1, June 2020

Printed in:

Printed in Malaysia

Published by:

Keysight Technologies

Bayan Lepas Free Industrial Zone, 11900 Penang, Malaysia

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

Declaration of Conformity

Declarations of Conformity for this product and for other Keysight products may be downloaded from the Web. Go to http://www.keysight.com/ go/conformity. You can then search by product number to find the latest Declaration of Conformity.

U.S. Government Rights

The Software is “commercial computer software,” as defined by Federal Acquisition Regulation (“FAR”) 2.101. Pursuant to FAR 12.212 and 27.405-3 and Department of Defense FAR Supplement (“DFARS”) 227.7202, the U.S. government acquires commercial computer software under the same terms by which the software is customarily provided to the public. Accordingly, Keysight provides the Software to U.S. government customers under its standard commercial license, which is embodied in its End User License Agreement (EULA), a copy of which can be found at http://www.keysight.com/ find/sweula. The license set forth in the EULA represents the exclusive authority by which the U.S. government may use, modify, distribute, or disclose the Software. The EULA and the license set forth therein, does not require or permit, among other things, that Keysight:

(1) Furnish technical information related to commercial computer software or commercial computer software documentation that is not customarily provided to the public; or (2) Relinquish to, or otherwise provide, the government rights in excess of these rights customarily provided to the public to use, modify, reproduce, release, perform, display, or disclose commercial computer software or commercial computer software documentation. No additional government requirements beyond those set forth in the EULA shall apply, except to the extent that those terms, rights, or licenses are explicitly required from all providers of commercial computer software pursuant to the FAR and the DFARS and are set forth specifically in writing else- where in the EULA. Keysight shall be under no obligation to update, revise or otherwise modify the Software. With respect to any technical data as defined by FAR 2.101, pursuant to FAR 12.211 and 27.404.2 and DFARS 227.7102, the U.S. government acquires no greater than Limited Rights as defined in FAR 27.401 or DFAR 227.7103-5 (c), as applicable in any technical data.

Warranty

THE MATERIAL CONTAINED IN THIS DOCUMENT IS PROVIDED “AS IS,” AND IS SUBJECT TO BEING CHANGED, WITHOUT NOTICE, IN FUTURE EDI- TIONS. FURTHER, TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, KEYSIGHT DISCLAIMS ALL WAR- RANTIES, EITHER EXPRESS OR IM- PLIED, WITH REGARD TO THIS MANU- AL AND ANY INFORMATION CON- TAINED HEREIN, INCLUD ING BUT NOT LIMITED TO THE IMPLIED WARRAN- TIES OF MERCHANTABILITY AND FIT- NESS FOR A PARTICULAR PURPOSE. KEYSIGHT SHALL NOT BE LIABLE FOR ERRORS OR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CON- NECTION WITH THE FURNISHING, USE, OR PERFORMANCE OF THIS DOCUMENT OR OF ANY INFORMATION CON- TAINED HEREIN. SHOULD KEYSIGHT AND THE USER HAVE A SEPARATE WRITTEN AGREEMENT WITH WARRANTY TERMS COVERING THE MATERIAL IN THIS DOCUMENT THAT CONFLICT WITH THESE TERMS, THE WARRANTY TERMS IN THE SEPARATE AGREEMENT SHALL CONTROL.

Safety Information

CAUTION

A CAUTION notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a CAUTION notice until the indicated conditions are fully understood and met.

WARNING

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

Table of Contents

Objective

In this lab session, you will learn how to interface the XBee3 module with analog temperature sensor. You will also learn how to communicate to the XBee3 transceiver using its AT command.

IMPORTANT

You must complete Lab 1 before you start this lab. There are basic operations that are covered in Lab 1 will not be covered in this lab. The instructions to install and set up the necessary USB drivers for serial and SSH connections with BeagleBone are also included in Lab 1.

Pre-Lab Setup Instructions

1. Prepare the required items as listed in the “Equipment and Accessories Required” list below.
2. Download the required software installers according to the “Software Required” list and install them on your Windows PC.

Equipment Required

1. Keysight U3810A IoT development kit with new BeagleBone Wireless CPU with XBee3 installed in its XB socket
2. Laptop or desktop PC running on Windows with internet access (Linux and MacOS may work but are not presently on Keysight’s list of supported platforms). The laptop or PC must have at least three USB ports available or you will require a USB hub.
3. Keysight 34465A 6½ Digital Multimeter

Accessories Required

1. 3x Micro-USB cables
2. 1x Analog temperature sensor (On-board analog temperature sensor accessory)
3. 2x SMA (f) to SMA(f) RF Coaxial Cable
4. 1x XB1 or XB2 Transceiver with XBee3 installed in its socket (On-board Xbee3 module accessory)
5. 1x Relay actuator (On-board relay actuator accessory)
6. Jumper wires
7. USB hub (Optional to support the use of more than three USB ports)

Software Required

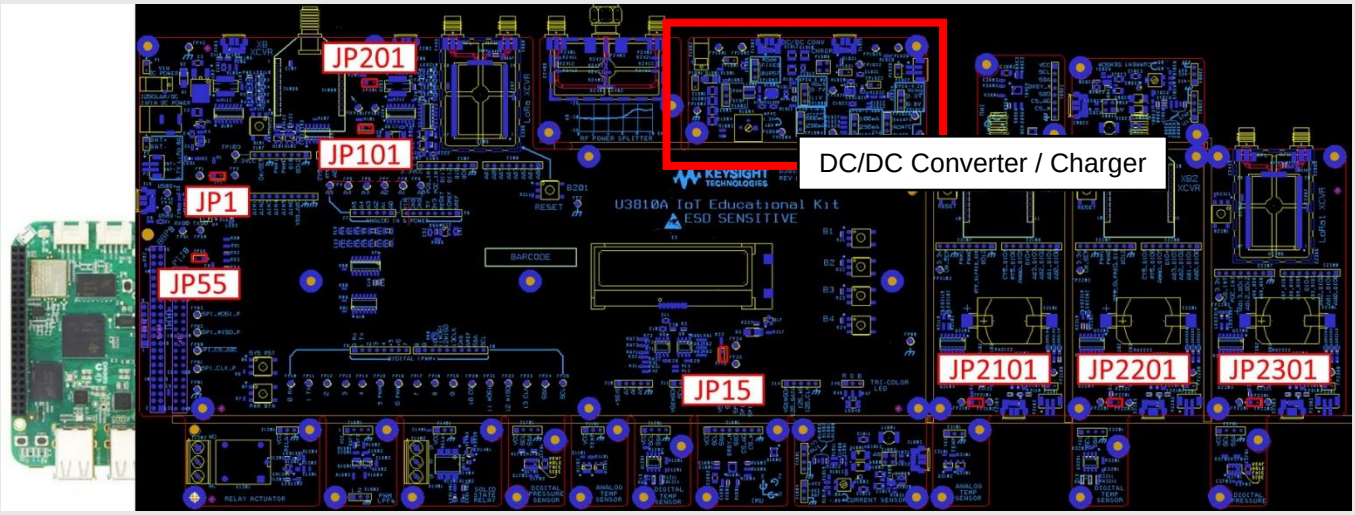
1. PuTTY (http://www.putty.org/)
2. WinSCP (https://winscp.net/eng/download.php)
3. Digi XCTU (https://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu)

Task 1 – Interface the Analog Temperature Sensor to the XBee3

NOTE

Before you begin the experiment, configure the Keysight U3810A IoT Development Kit as a "cape" on top of the BeagleBone CPU, and with the jumper configuration shown below:

Jumper	JP1	JP15	JP55	JP101	JP201	JP2101	JP2201	JP2301
Name	Input Current	Sensor Current	+5VSYS +5VRAW	XB Current	LoRa Current	XB1 Current	XB2 Current	LoRa1 Current
Position	In place	In place	Removed	In place	In place	In place	In place	In place



The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

WARNING Do not connect voltages greater than 3.3 V to GPIO pins as this may damage the BeagleBone CPU. These over-voltage sources include the VIN pin on the Arduino Shield and DC Power connectors, and +5VRAW and +5VSYS on interface connectors such as J10, JP55 and TP51.

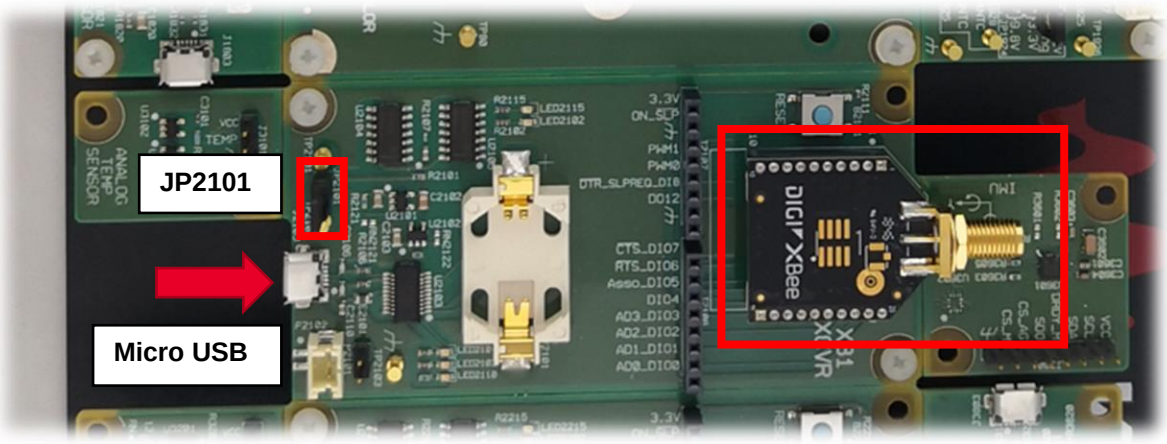
The U3810A on-board XB1 or XB2 Accessories are compact circuits that provide UART serial communication to XBee3 transceivers. It provides a 5V TTL logic interface and allows a straightforward interface to the microcontroller for embedded wireless development. With proper configuration, the XBee3 XB1 or XB2 can broadcast messages or send a point to point message to other XBee3 modules via Zigbee wireless connectivity.

In this task, you will connect the TMP36 analog temperature sensor to the XB1 or XB2 Transceiver and capture the measured temperature value using AT commands. Remember, the digitized value of the temperature sensor output voltage is represented in digital format (hex).

The AT commands are also known as Hayes commands, which were developed in the 1980's to control Hayes modems. They are commonly known as AT commands as each command starts with AT, which is the abbreviation of attention. The term AT in the command line is used to notify the host modem that a command will follow. For example, the command **ATID** obtains the personal area network ID of the modem. And AT commands are always preceded by **+++**

Perform the following procedure to interface the temperature sensor to the XBee3 XB1 or XB2 Transceiver module.

1. Connect the XB1 (or XB2) accessory to your PC using a micro USB cable. Assure proper orientation and seating of all 20 pins on the XBee3.

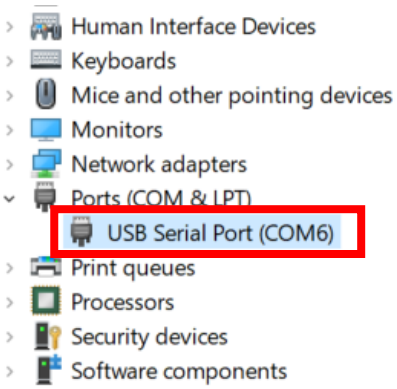


The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

NOTE

Ensure that jumper JP2101 (or JP2201) connected with black jumper.

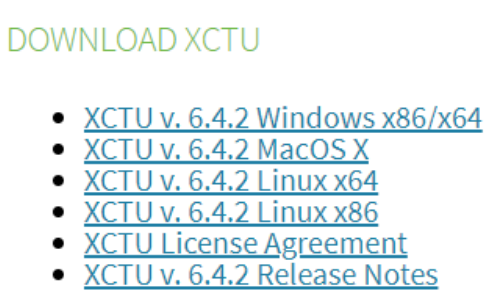
2. Open **Device Manager** on the development PC and identify the COM port of the XB1 or XB2 Transceiver board. You can disconnect the USB and reconnect it to identify the COM port in the Device Manager.



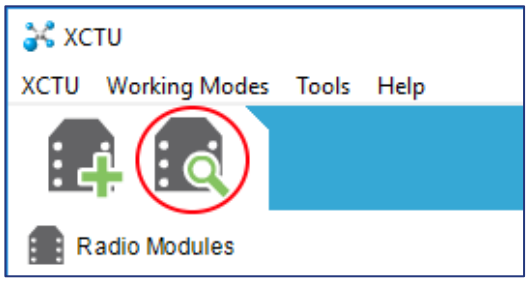
NOTE

There are many ways to open Device Manager. One way that works on all versions of Windows OS is to press the **Windows Key + R >** type **devmgmt.msc**. In the resulting pop-up window > hit **Enter**.

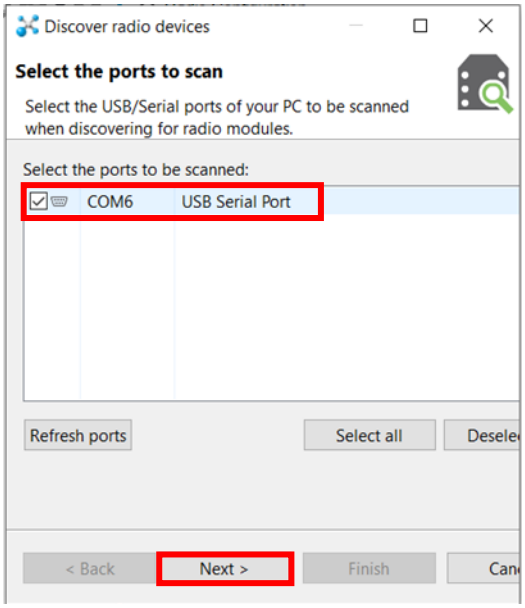
3. Download the XCTU software (version 6.3.2 or above) at https://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu. You can either register when prompted or click "No thanks, register later".



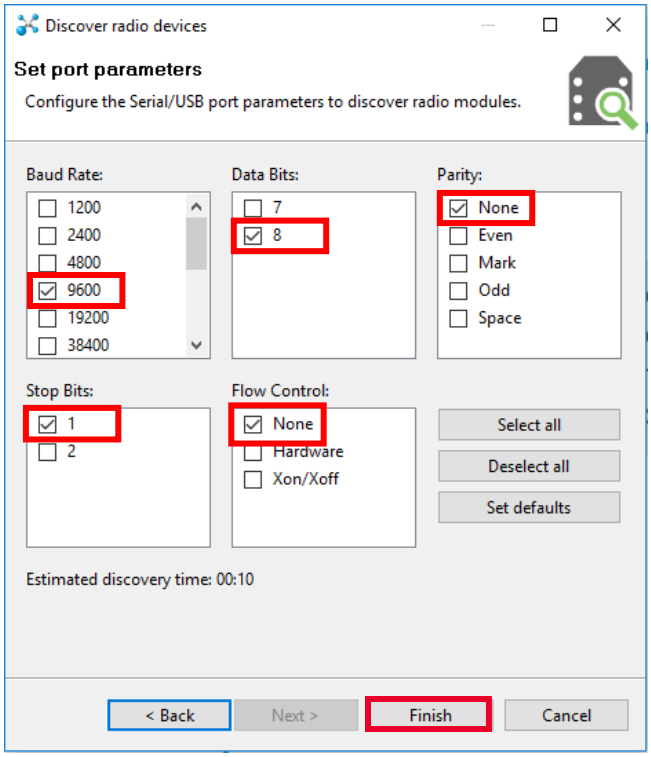
4. After launching XCTU, click the **Discover Devices** icon.



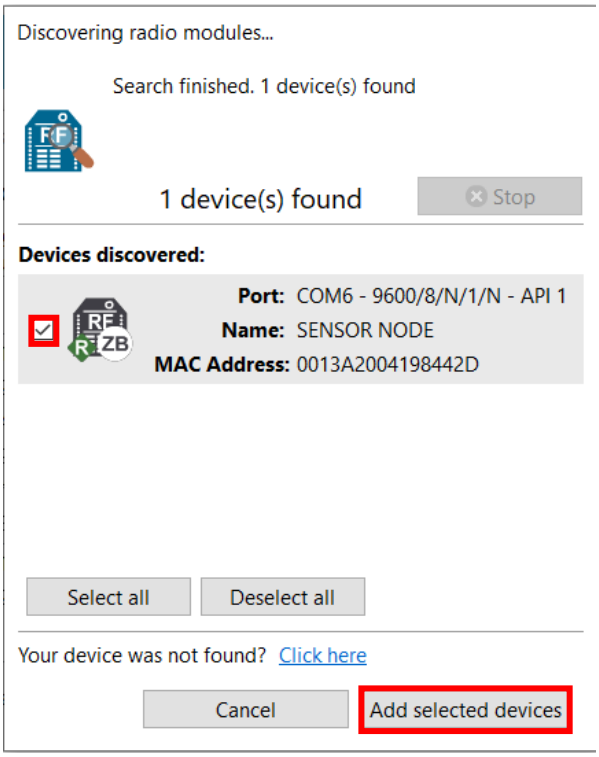
5. Select the **COM Port** which you identified in Device Manager and click **Next**.



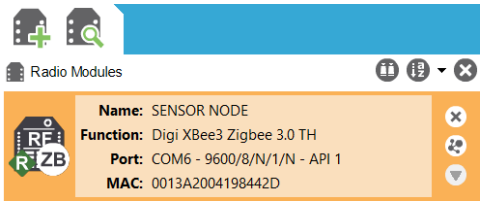
6. Configure the serial port parameters as shown in the image below and click **Finish**.



7. Select the discovered XBee3 and click **Add Selected Devices**.

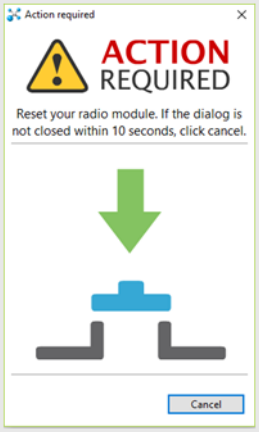


At this point, the XCTU control panel will show the XBee3.



NOTE

When you are shown this prompt (below), press the **RESET** button located near the XBee3 to proceed. This is because the XBee3 module might still be sleeping.

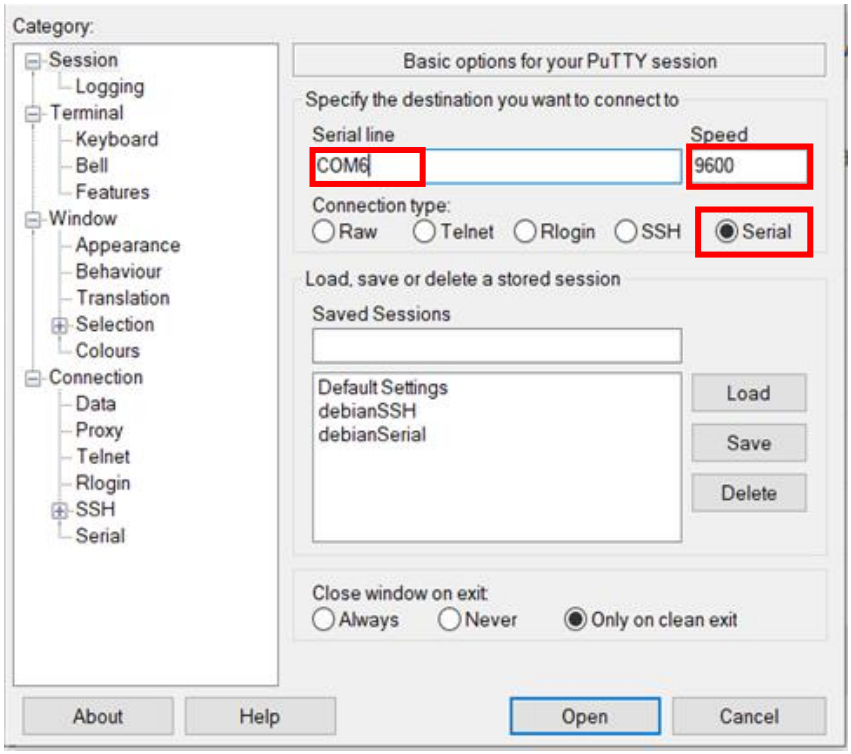


8. Click the XBee3 device on the XCTU control panel to change the module settings.
9. Click the **Default** button to reset the firmware settings back to their default values. Then, click the **Write** button to perform the write operation and click **Read** to confirm the reset value of XBee3 module.

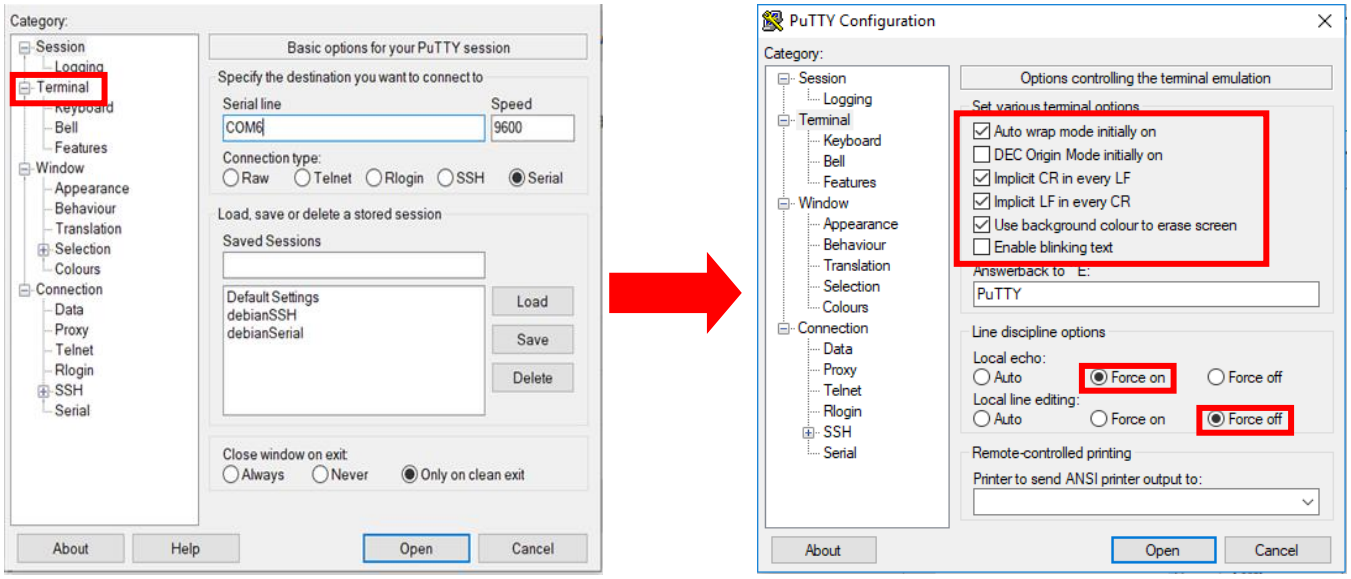
10. Change the **IO Sampling Rate (IR)** to **2000 ms** and **Supply Voltage High Threshold (V+)** to **5 V**. Then, click **Write** to save the setting. Use the Parameter search box to set these parameters.

Parameter	Value	Description
IR	7D0	Set the IO sampling rate to 2000 ms to enable periodic sampling.
V+	5	If the supply voltage measurement equals or drops below this threshold, the supply voltage will be appended to an IO sample transmission.

11. Close XCTU after you have configured the XBee3 XB1 or XB2 Transceiver.
12. Launch PuTTY and use the following configurations to connect to the XBee3 XB1 or XB2 Transceiver.



13. Click **Terminal** and configure the terminal option as shown below.



14. Click **Open** to connect to the XBee3 XB1 or XB2 Transceiver and log in to the PuTTY terminal.
15. Enter **+++** in the PuTTY terminal and wait for **OK** to appear. You will need to wait from two to 6.4 seconds (64x0.1s = 6.4s is the default command mode timeout).

> **NOTE**
>
> When you did not receive any response, verify your connection configuration before trying to reconnect and retry the command again.

16. Enter **AT** and press **Enter** on your keyboard. When you see the **OK** response, this means that the XBee3 module is now in AT command mode.

```
+++OK
AT
OK
```

Next, you will configure one of the digital IO lines from the XBee3 into an ADC input (only DIO1 to DIO3 can be configured for analog inputs). You will use DIO3 to acquire the ANALOG TEMP SENSOR output voltage values.

17. First, you will need to check the current DIO3 configuration. In the PuTTY terminal, enter **+++** and wait for the **OK** response.
18. Enter **ATD3** and press **Enter** on your keyboard. It should return a number to indicate its current configuration as shown in the example below.

```
+++OK
ATD3
0
```

19. Enter **+++** and wait for the **OK** response.

20. Enter **ATD3 2** and press **Enter** on your keyboard. You will receive an **OK** response upon successful configuration.
21. Enter **+++** and wait for the **OK** response.
22. Enter **ATD3** and press **Enter** on your keyboard to verify the current configuration. The correct response is as shown in the example below.

```
+++OK
ATD3 2
OK
+++OK
ATD3
2
```

NOTE

The following are the six configuration options for each digital IO line:

- 0: Disables I/O on that pin
- 1: Built-in function, if available on pin
- 2: Analog input, only pins D0 to D3
- 3: Digital input
- 4: Digital output, LOW (0 volts)
- 5: Digital output, HIGH (3.3 volts)

For more information, you may refer to **Enter Command mode** and/or **Send AT commands** in the XBee3 Zigbee 3.0 Module User Guide.

Enter Command mode

To get a device to switch into this mode, you must issue the following sequence: **GT** + **CC**(**+++**) + **GT** and observe guard times before and after the command character.

The default sequence to transition to Command mode is:

- No characters sent for one second [**GT** (Guard Times) parameter = 0x3E8].
- Three plus characters (**+++**) input within one second [**CC** (Command Sequence Character) parameter = 0x2B.].
- No characters sent for one second [**GT** (Guard Times) parameter = 0x3E8].

When you send the command mode sequence, the device sends **OK** out the UART pin. The device may delay sending the **OK** if it has not transmitted all of the serial data it received.

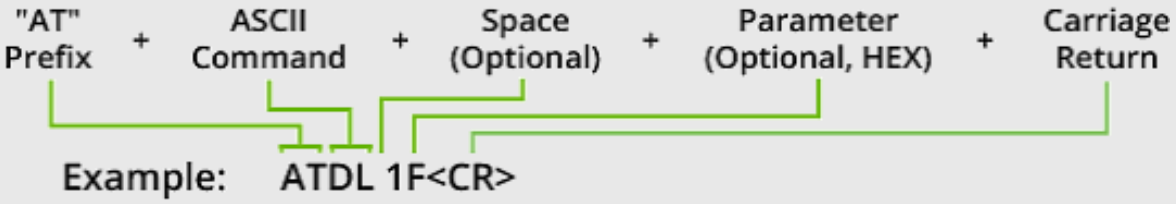
When the device is in Command mode, it starts the Command mode timer (**CT** command) and can receive AT commands on the UART port.

You can customize the command character, the guard times and the timeout in the device's configuration settings. For more information, see CC (Command Character), CT (Command Mode Timeout) and GT (Guard Times).

Send AT commands

Once the device enters Command mode, use the syntax in the following figure to send AT commands. Every AT command starts with the letters **AT** (which stands for "attention"), followed by two characters that indicate which command is being issued and several optional configuration values.

To read a parameter value stored in the device's register, omit the parameter field.



The preceding example changes the device's destination address (Low) to 0x1F.

To store the new value to non-volatile (long term) memory, send the **WR** (Write) command. This allows parameter values that you modify to persist in the device's registry after a reset. Otherwise, the device will restore parameters to the previous values after a reset.

Errors in Command Line

Errors in the command line often caused by an unintended backspace or space entered when you configure the XBee3.

Enter the +++OK command again before running the <AT command> and press Enter.

```
+++OK
ATD3
ERROR
+++OK
ATD3
0
```

Use of the XB Transceiver on the U3810

You may also try AT commands with the XB Transceiver on the U3810 by using a PuTTY terminal with a Micro USB Serial (COM) connection at J103. When this USB is connected it will automatically configure XB serial communication through USB.

Verifying the XB with AT Commands using the BeagleBone CPU and a minicom terminal

You may also try AT commands from the BeagleBone CPU using a terminal program such as **minicom** which can be installed on and run inside the BeagleBone:

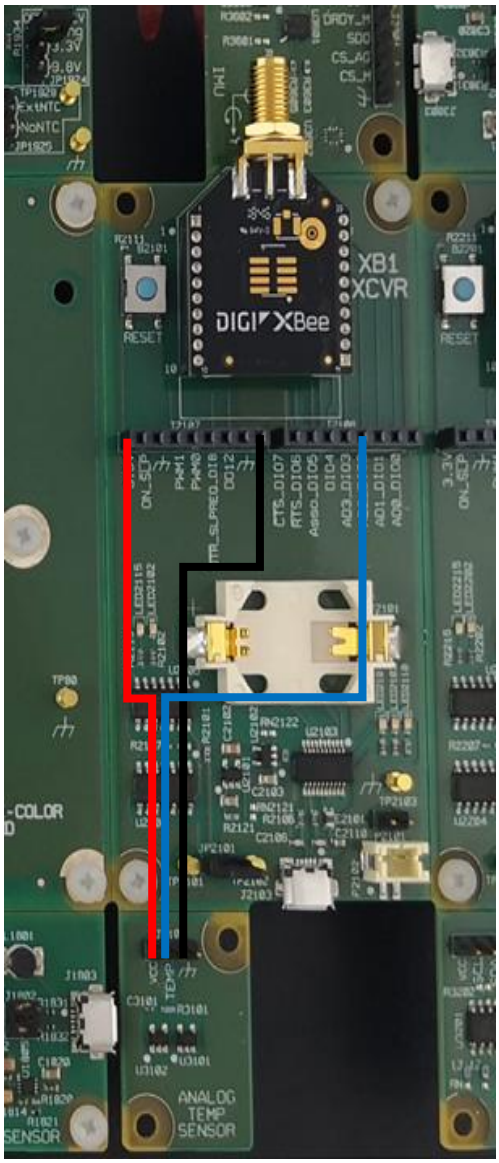
```
sudo apt install minicom
```

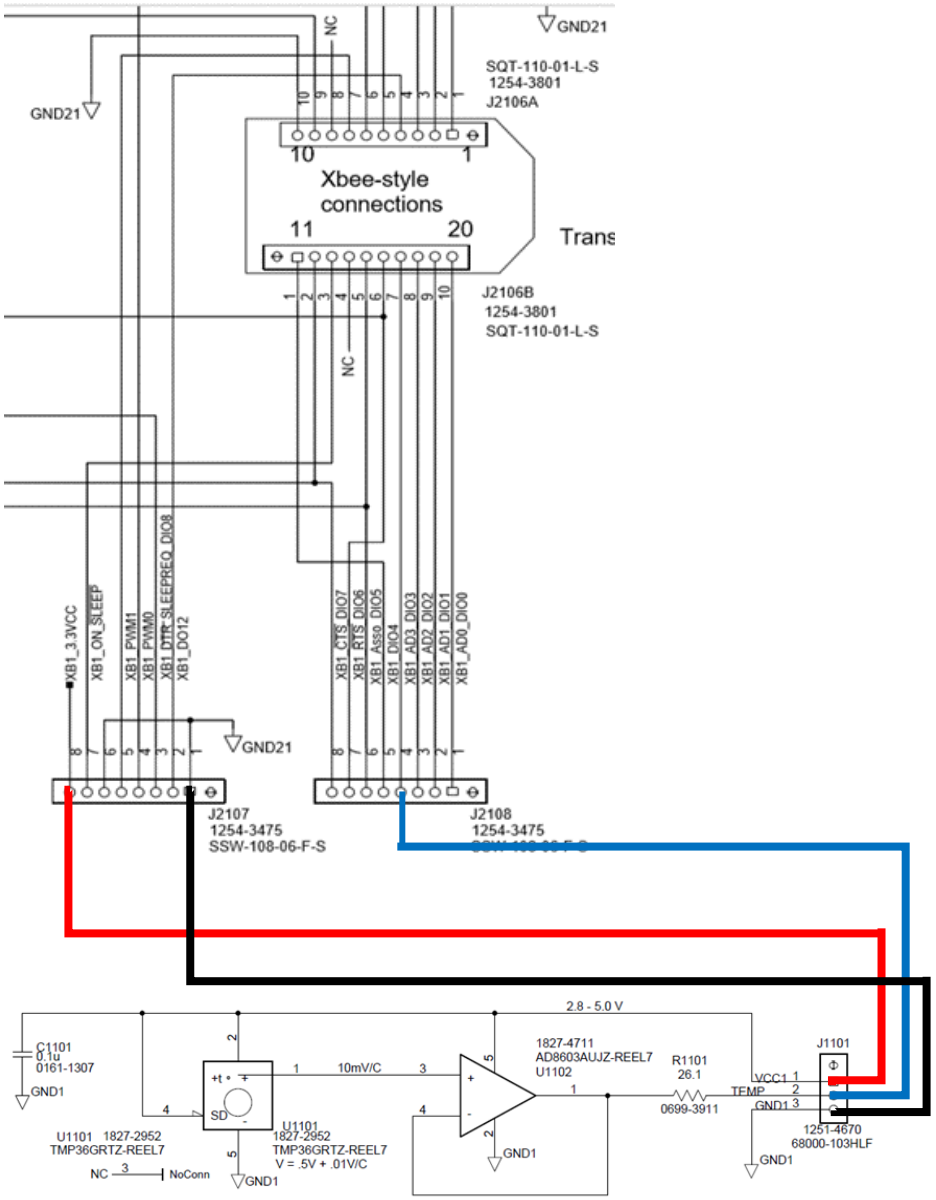
```
minicom -D /dev/ttyS1 -b 9600
```

Task 1a – Analog Temperature Sensor to XBee3 Jumper Wire Setup

1. Connect the TMP36 to the XBee3 (V+ to 3.3V, GND to GND, and IN to DIO3) using jumper wires.

From	To
ANALOG TEMP SENSOR J3101 (pin VCC)	XB1 XVCR J2107 (pin 3.3V)
ANALOG TEMP SENSOR J3101 (pin TEMP)	XB1 XVCR J2108 (pin AD3_DIO3)
ANALOG TEMP SENSOR J3101 (pin GND)	XB1 XVCR J2107 (pin GND)





The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

2. Force a Sample of the I/O which includes the output voltage level, enter **+++** and **ATIS** followed by **<Enter>**.

NOTE

The temperature sensor output (V_{OUT}) is an analog value. For the XBee3 to understand the output, this analog value needs to be represented as a digital value. The XBee3's 10-bit analog-to-digital converter (ADC) is used to do this conversion. In this task, channel 3 (DIO3) as the input pin to read the analog signal.

For the TMP36 convert the digital value into temperature (in degrees Celsius) using this formula:

$$\text{Temperature} = \left(\frac{\text{ADC Value}}{1024} \times 1250\ \text{mV} - 500\text{mV}\right) \div 10\ \text{mV/C}$$

For the LM35 (not provided with U3810A) convert the digital value into temperature (in degrees Celsius) using this formula:

$$\text{Temperature} = \left(\frac{\text{ADC Value}}{1024} \times 1250\ \text{mV}\right) \div 10\ \text{mV/C}$$

The ADC value is the digital value obtained from the ADC. For a 10-bit ADC, the maximum value is 1023 while the minimum is 0. Multiply the ratio by 1250 mV because the ADC's input range is from 0 V to 1.25 V. However, 1 °C change corresponds to 10 mV, so divide the entire equation by 10 mV to obtain the actual temperature.

3. Convert the last input, the voltage level, into degrees Celsius. An example is as shown below.

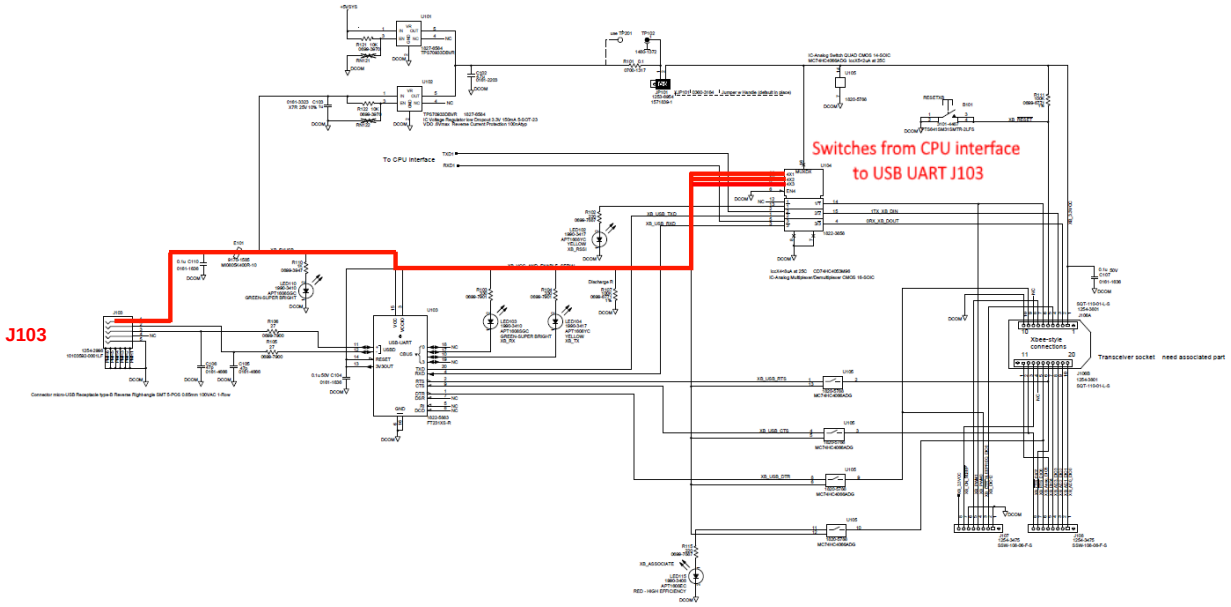
```
+++OK
ATIS
01
2000
08
2000
0265
```

Returned ADC Value (Hex)

Returned ADC value (Hex)	
Returned ADC value (Decimal)	
TMP36 output voltage (mV)	
Temperature (°C)	

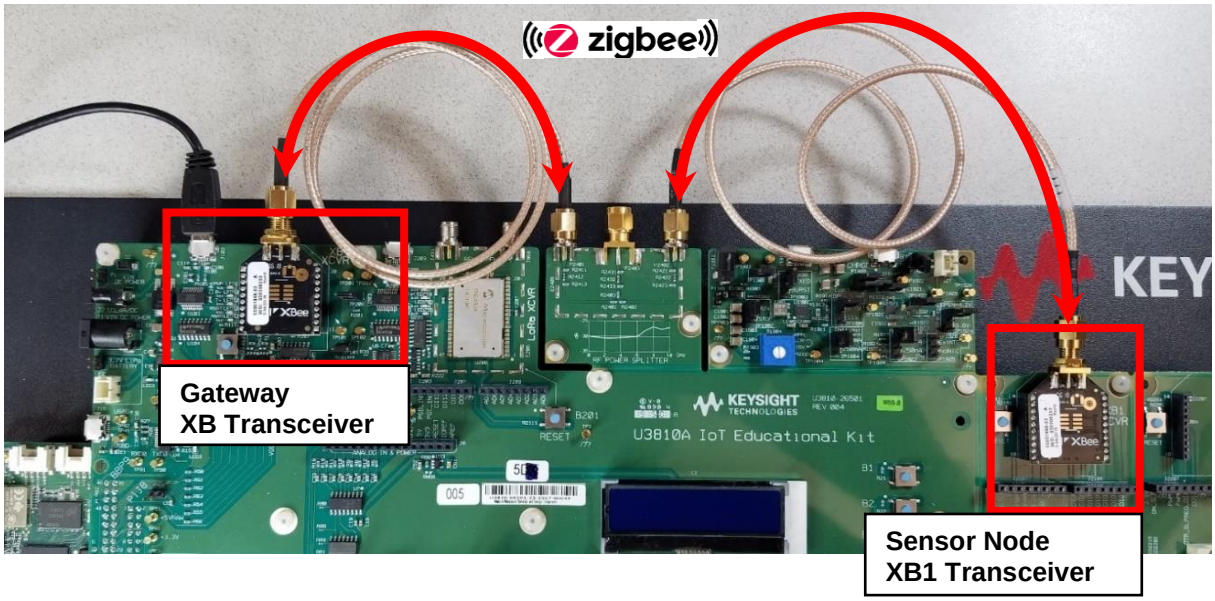
Task 2 – Remote Sensing and Actuation with Zigbee

On the Keysight U3810A, the XBee3 communications are designed to switch between the USB serial port and the CPU interface via a multiplexer/demultiplexer. Note that when the XBee3's USB connection is made, power from the connection automatically reconfigures the XBee3 to change its API Command Interface from the BeagleBone CPU's UART1 to the USB Serial (U103 FT231X).



The XBee3 provides wired (or wireless) communication between the sensor node and gateway.

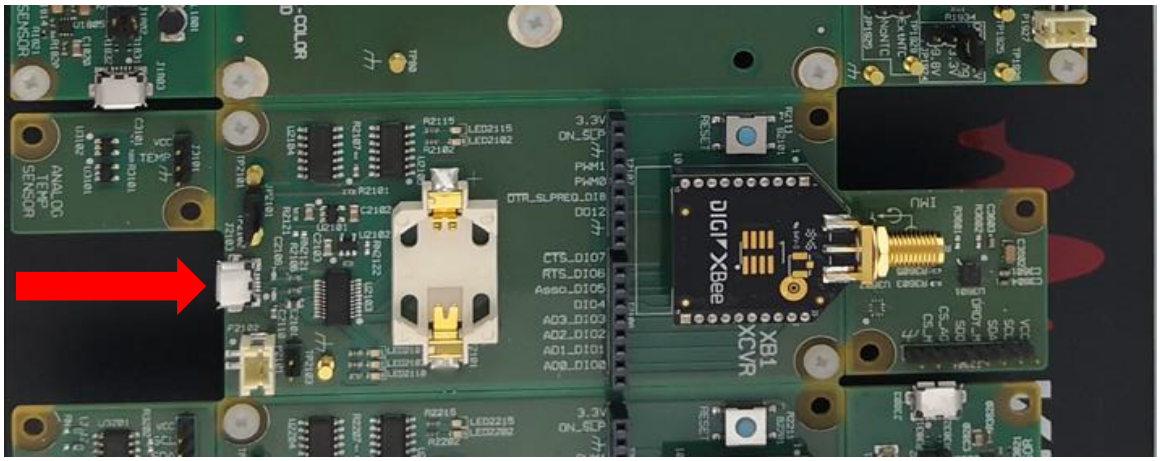
In this task, the U3810A's XB Transceiver circuit will act as a gateway. The second XBee3 (on-board XB1 or XB2 Transceiver board) will act as the sensor node. The sensor node will digitize the output of the analog temperature sensor and report the measurement to the gateway at fixed intervals.



The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors, and components.

Task 2a – Configure the Sensor Node as a Zigbee Router

1. Use a micro-USB cable to connect the XB1 (or XB2) accessory to your PC.

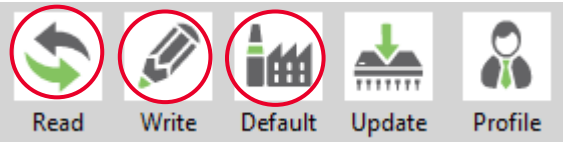


The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

NOTE

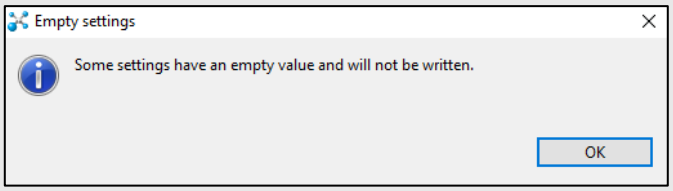
Ensure that jumper JP2101 (or JP2201) is installed with black jumper.

2. Use XCTU to discover the device and select the device to display its parameters on XCTU. You can refer to Task 1 – Interfacing the Analog Temperature Sensor to the XBee3 for more details.
3. Click the **Default** button and answer **Yes** to reset all the firmware settings to their default values (Refer to Task 1 if the device has not been added to XCTU). Click the **Write** button to perform the write operation, followed by the **Read** button to perform the read operation.



NOTE

Due to the empty **Zigbee Stack Profile (ZS)** by default, you will need to click **OK** when you see the prompt below. The **ZS** setting will remain as its last configuration setting.



4. Configure the XBee3 XB1 or XB2X Transceiver according to the settings shown below. You may use the **Parameter** search field at the top right to locate each setting.

Parameter	Value	Description
CE	Join Network [0]	Sets the device as router or end device.
ID	2020	Defines the network that a radio will attach to. This must be the same for all radios in your network.
ZS	1	Set or read the Zigbee stack profile value. This must be the same on all devices that will join the same network.
JV	Enabled [1]	Verifies if a coordinator exists on the same channel to join the network or to leave if it cannot be found.
NI	SENSOR NODE	Defines the node identifier, a human-friendly name for the module. The default NI value is a blank space. Make sure the configuration is blank before you change the value.

5. Set the **I/O Settings** and **I/O Sampling** section according to the table below to allow the device to read the voltage on the analog temperature sensor repeatedly.

Parameter	Value	Description
D0	Disable [0]	Disable I/O DIO0/AD0/Commissioning Button Configuration from receiving input.
D1	Disable [0]	Disable I/O DIO1/AD1 Configuration from receiving input.
D2	Disable [0]	Disable I/O DIO2/AD2 Configuration from receiving input.
D3	ADC [2]	Enable I/O DIO3/AD3 Configuration to receive analog DC input
D4	Disable [0]	Disable I/O DIO4 Configuration from receiving input.
D5	Disable [0]	Disable I/O DIO5/Association LED Configuration from receiving input.
D8	Disable [0]	Disable I/O DIO8/DTR/Sleep_Rq Configuration from receiving input.
D9	Disable [0]	Disable I/O pin 7 from receiving input.
P0	Disable [0]	Disable I/O pin 0 from receiving input.
PR	7FFF	Set/read bitfield to configure internal pullup resistors status for I/O lines. 1=internal pullup enabled, 0=no internal pullup.
IR	7D0	Set the IO sampling rate to 2000 ms to enable periodic sampling.
V+	5	When the supply voltage measurement is equal or below this threshold, the supply voltage will be appended to an IO sample transmission.

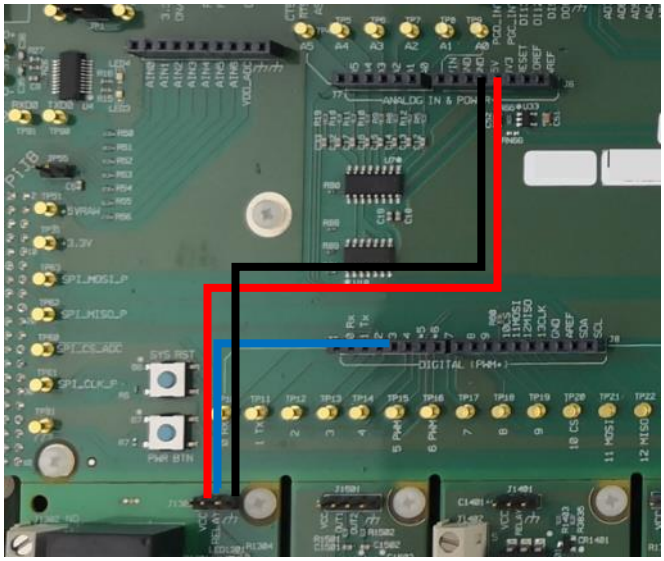
6. Click **Write** button to apply these settings into the XBee3. Then, click **Read** to read back the settings.
7. Based on the connection in Analog Temperature Sensor to Zigbee Jumper Wire Setup, use jumper wires to connect the analog temperature sensor to the **XBee3 XB1** or **XB2 Transceiver**.

   The XBee3 XB1 or XB2 Transceiver (Sensor Node) will periodically sample temperature and transmit it to the XBee3 XB Transceiver (Gateway) at fixed intervals (2000 ms as configured).

Task 2b – Configure the Gateway as Zigbee Coordinator

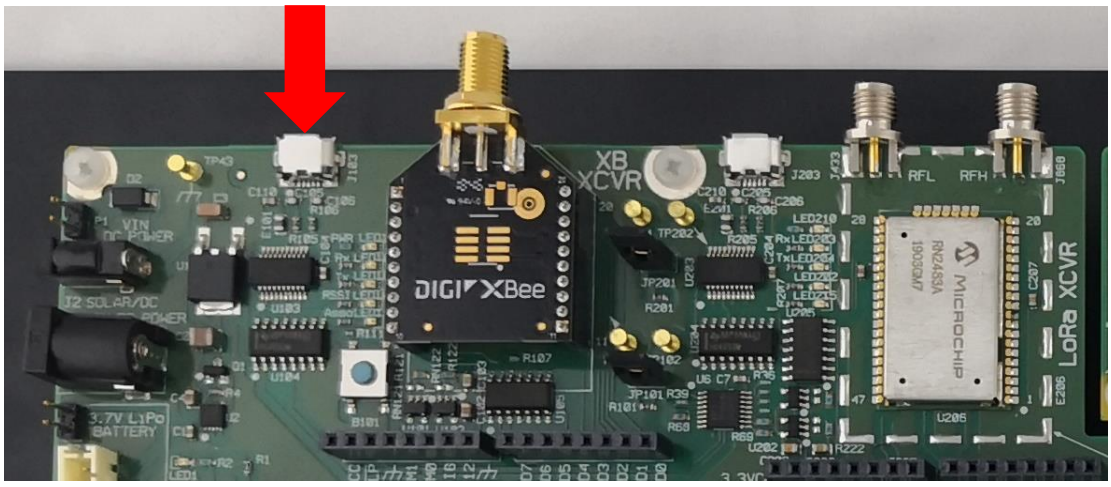
1. Make the following connections between the relay actuator and Keysight U3810A using jumper wires.

From	To
RELAY ACTUATOR J1301 (pin VCC)	J6 (pin 5V)
RELAY ACTUATOR J1301 (pin RELAY)	J9 (pin 2)
RELAY ACTUATOR J1301 (pin GND)	J6 (pin GND)



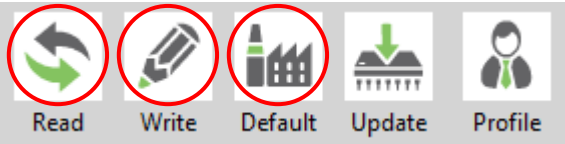
The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

2. Connect J103 on the XBee3 XB Transceiver to your PC via USB to micro-USB cable.



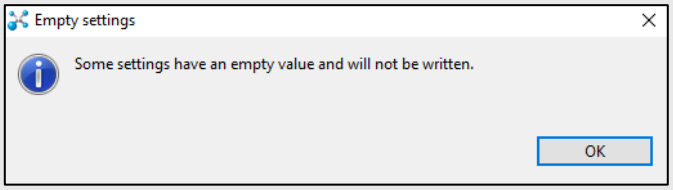
3. Discover the device using XCTU and select the device to display its parameters on XCTU. You can refer to Task 1 – Interfacing the Analog Temperature Sensor to the XBee3 for more details.

4. Click the **Default** button and answer **Yes** to reset all the firmware settings to their default values. Click the **Write** button to perform the write operation, followed by the **Read** button to perform the read operation. Do this step for both the gateway XB and the sensor XB1 or XB2.



NOTE

Due to the empty **Zigbee Stack Profile (ZS)** by default, you will need to click **OK** when you see the prompt below. The **ZS** setting will remain as its last configuration setting.



5. Configure the **XBee3 XB Transceiver (Gateway)** on the U3810A according to the settings shown below.

Parameter	Value	Description
CE	Form Network [1]	Sets the device as coordinator.
ID	2020	Defines the network that a radio will attach to. This must be the same for all radios in your network.
ZS	1	Set or read the Zigbee stack profile value. This must be the same on all devices that will join the same network.
JV	Disabled [0]	Verifies if a coordinator exists on the same channel to join the network or to leave if it cannot be found.
NI	GATEWAY	Defines the node identifier, a human-friendly name for the module. The default NI value is a blank space. Make sure to delete the space when you change the value

NOTE

API mode in detail

API mode provides a structured interface where data is communicated through the serial interface in organized packets and in a determined order. This enables you to establish complex communication between devices without having to define your own protocol.

By default, Zigbee devices are configured to work in transparent mode: all data received through the serial input is queued up for radio transmission and data received wirelessly is sent to the serial output exactly as it is received, with no additional information.

Due to this behavior, devices working in Transparent mode have some limitations:

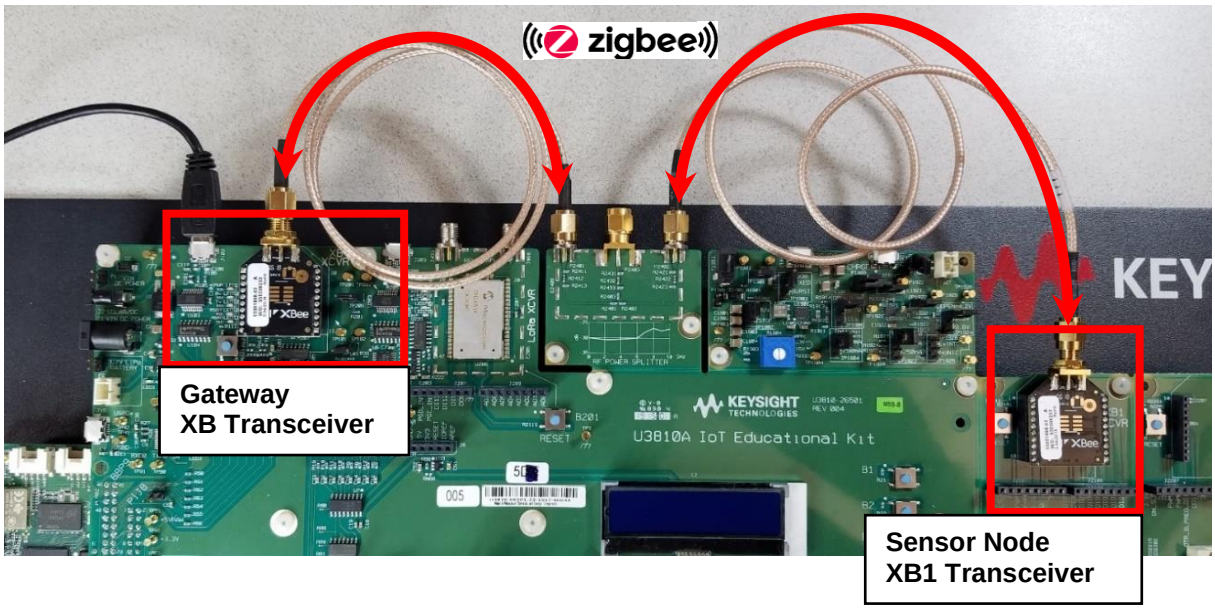
- first transition the device into Command mode.
- If a device needs to transmit messages to different devices, you must update its configuration to establish a new destination. The device must enter Command mode to set up the destination.
- A device operating in Transparent mode cannot identify the source of a wireless message it receives. If it needs to distinguish between data coming from different devices, the sending devices must include extra information known by all the devices so it can be extracted later. To do this, you must define a robust protocol that includes all the information you think you need in your transmissions.

To minimize the limitations of the transparent mode, devices provide an alternative mode called Application Programming Interface (API). API mode provides a structured interface where data is communicated through the serial interface in organized packets and in a determined order. This enables you to establish complex communication between modules without having to define your own protocol.

6. Click the **Write** button to apply the settings to the XBee3 on the U3810A. Then click **Read** to read back the configurations.
7. Check the Association Indication (AI) parameter for both Zigbee modules (Gateway and Sensor Node). It should now show 0 to indicate that it is associated with a network. You may have to wait and re-read the configuration to see the AI change to 0.

Task 2c – Testing the Connection between Sensor Node and Gateway

1. Use the RF cables to connect the Sensor Node to the Gateway. At this point, ensure that you have completed Task 2a - Configure the Sensor Node as Zigbee Router and Task 2b - Configure the Gateway as Zigbee Coordinator.

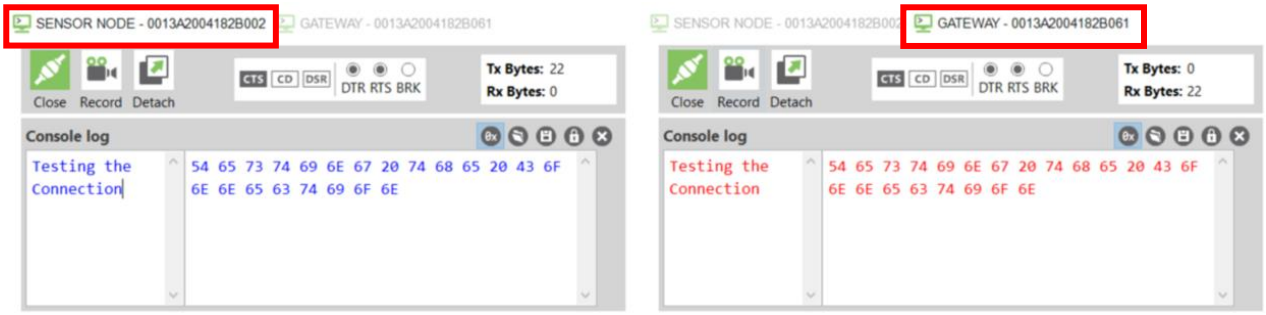


The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

2. On the XCTU, click the second icon (top right) to switch to **Consoles working mode**  .

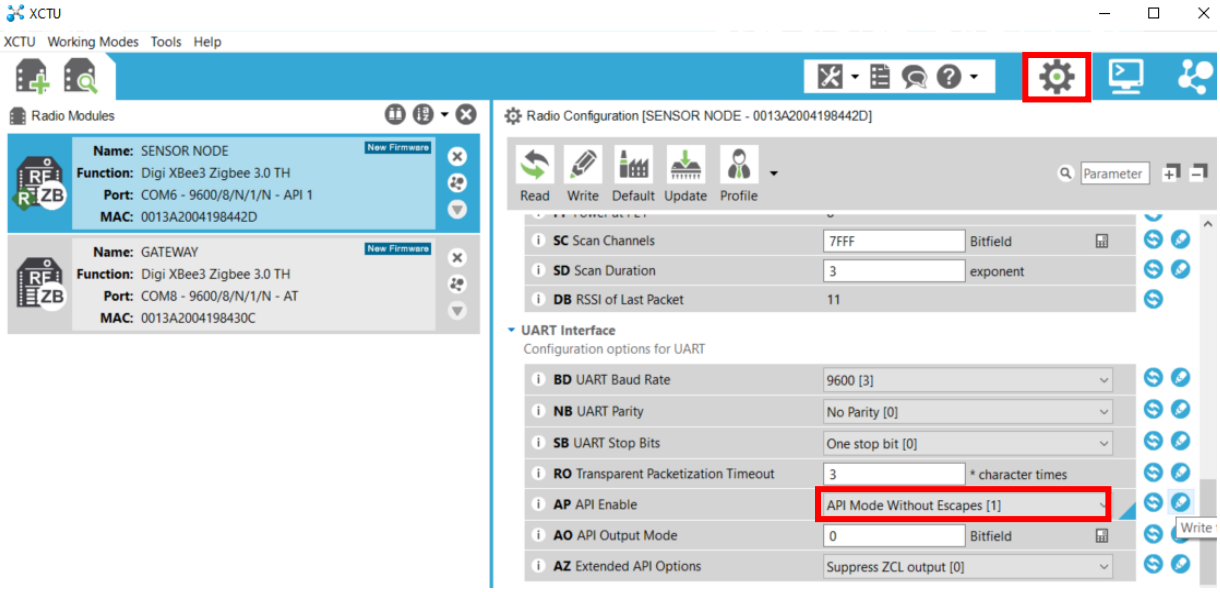
3. Click the **Open** icon for both **Sensor Node and Gateway** as shown below.

4. On the Sensor Node tab, type anything into the **Console log** and observe the text being sent to the **Console log** in the Gateway tab. Here is an example below.



Task 2d – Enable Temperature Data to Transmit from SENSOR NODE to Gateway

1. On the XCTU, return to the Configuration Mode. Configure the **Gateway and Sensor Node's AP API Enable** parameter to **API enabled without Escape [1]**. Click **Write** to write the setting into both devices. Click **Read** to read the settings and ensure that they are correct.

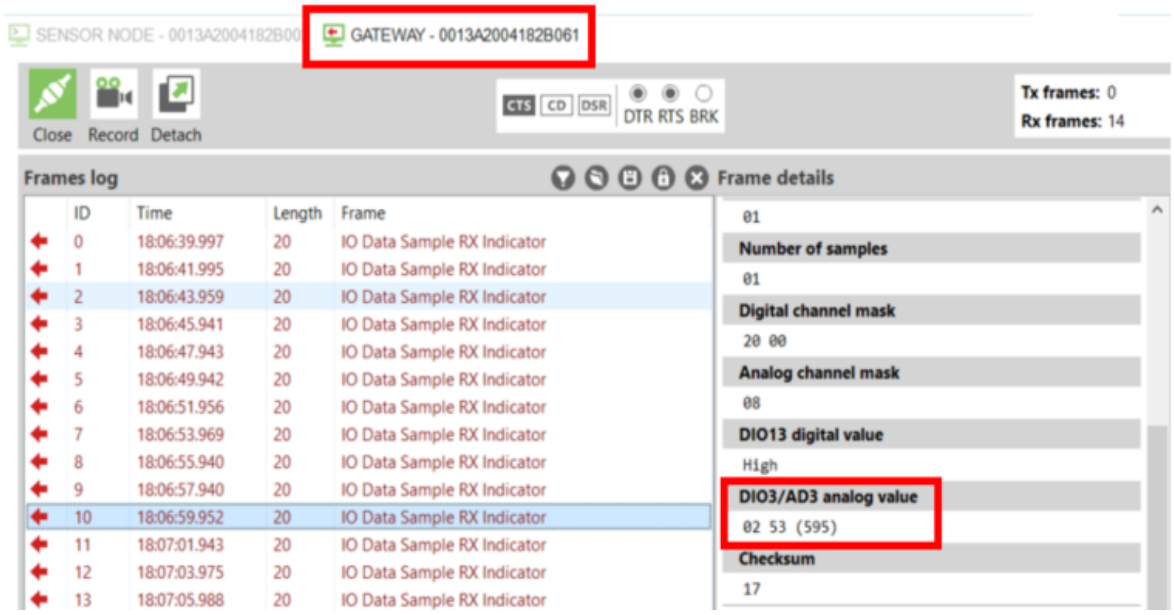


NOTE

The API mode provides a structured interface where data is communicated through the serial interface in organized packets and in a determined order. This enables you to establish complex communication between devices without having to define your own protocol.

2. Switch to **Consoles working mode** ( ) and click the **Open** ( ) icon for both **Sensor Node** and **Gateway.**

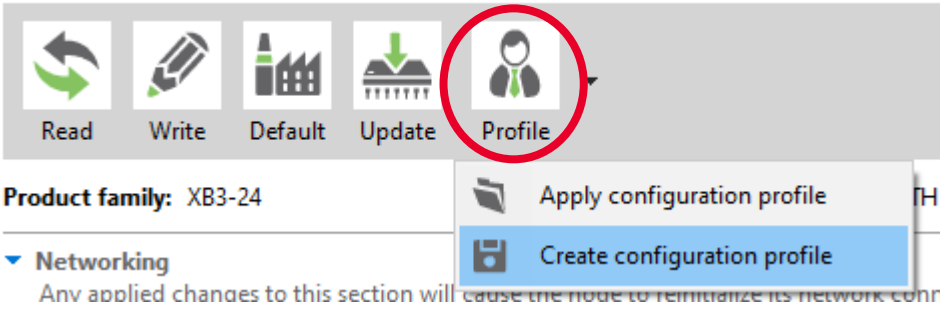
3. Observe the Frames log in the **Gateway** tab for data transmitted from the **Sensor Node**. Click one of the frames and observe the **DIO3/AD3 analog value** in **Frame details**, as shown below.



4. Using the formulas in Task 2, convert the temperature data can be converted and fill in your answers in the table below.

Returned ADC value (Hex)	
Returned ADC value (Decimal)	
Temperature (°C)	

5. Click **Profile icon > Save configuration profile** to save the XBee3 device configurations for both Sensor and Gateway. You will be using this in subsequent labs.



Zigbee Device NI profile	Save the configuration file as
SENSOR	M1_L4_T2_SENSOR
GATEWAY	M1_L4_T2_GATEWAY

NOTE

If you have not already saved these XBee3 configurations as profiles, you may save time and assure accurate configurations by copying and extracting the files from the lab code directory in the BeagleBone to your computer. You may then use the **M1_L4_T2_GATEWAY.xpro** and **M1_L4_T2_SENSOR.xpro** profiles for this task.

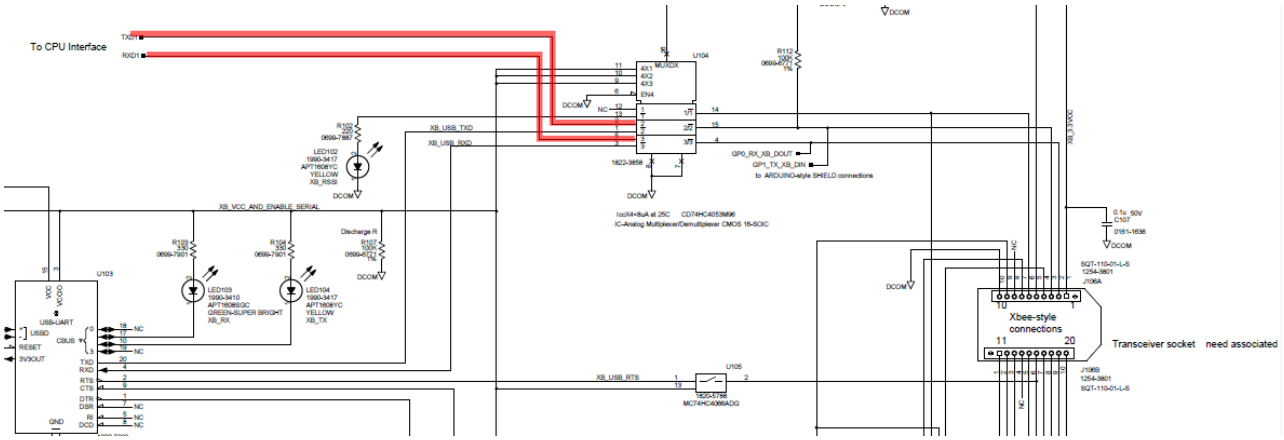
Go to **Profile** > **Create configuration profile** and select the profile. For the gateway (U3810A XB), you will need to change PAN ID (ID) to match the OP sensor node (XB1 or XB2) and save the gateway profile once you have customized to your network.

The **xpro** file saved by XCTU is a zip format file. You can use zip software to unzip the file and examine the contents. Do not use XCTU versions older than 6.3.10 as configuration files for those versions were saved in XML format, which is incompatible with later releases.

Read the XCTU change log (**XCTU > Help > Change Log**) for more details.

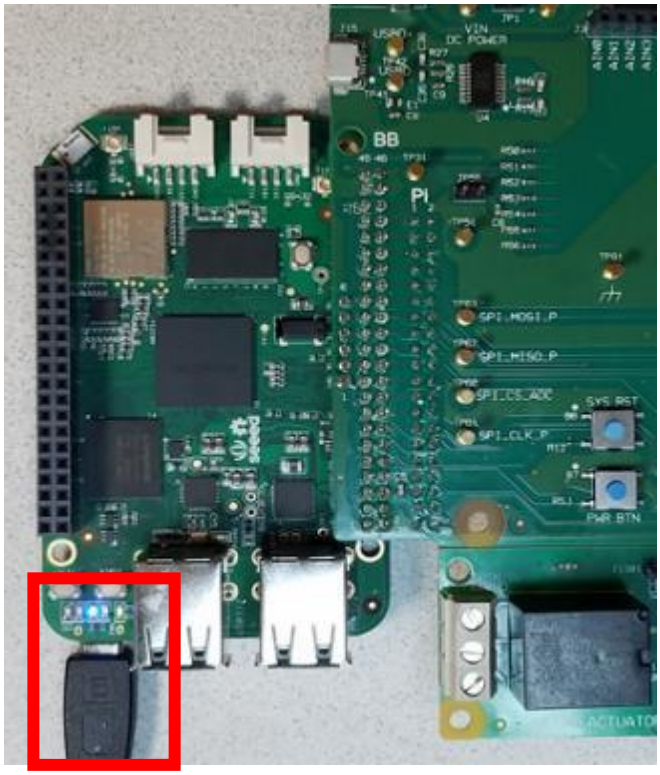
Task 2e – Enable Relay Actuator to Respond to Temperature Changes

1. Disconnect the micro-USB cable from J103. This configures the XBee3 XB Transceiver to change its API Command Interface from the USB Serial (U103 FT231X) to the BeagleBone CPU's UART1.



2. Before you continue on this lab session, ensure that the jumper wire setup on analog temperature sensor and relay actuator are still connected on U3810A. Otherwise, refer to Analog Temperature Sensor to Zigbee Jumper Wire Setup and Task 2b - Configure the Gateway as Zigbee Coordinator for instructions on how to set it up.

3. Use a micro-USB cable to connect the BeagleBone CPU to your PC. This is to power up the U3810A.



The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

4. Use PuTTY to log in into the BeagleBone module with the following details.

Host Name	192.168.7.2
Port	22
Connection type	SSH
Username	debian
Password	temppwd

> **NOTE**
>
> This connection from PuTTY to The BeagleBone is not using serial connection. It uses the USB RNDIS device network which is a virtual Ethernet network so that you can use TCP connection to BeagleBone. This method is used since USB2 and its UART will be required for communication to the XBee3 module in this lab. It is also possible to use an IDE and this is covered in Appendix A – Cloud 9 IDE Usage.

5. At the beginning of this lab, you will need to reset any modifications on the lab codes from the previous lab session (if any). Run the following commands to reset the lab codes.

   ```
   cd ~
   sh LabCodeReset.sh
   ```

   Select **Y** when prompted to erase all existing files in the LabCode folder.

6. Run the following commands to go to M1-L4 LabCode directory

   ```
   cd /home/debian/LabCode/M1-L4
   ```

7. Use the following commands to compile and run it.

   ```
   ./temp_relay
   ```

> **NOTE**
>
> If the code file does not exist, enter the following to recompile the code:
>
> `gcc M1_L4_T2_ZigBee_Temp_Relay.c -lmraa -o temp_relay`

The program will read the temperature values received by the XB1 (or XB2) module and display it on the LCD. Next to the temperature value is the temperature threshold value where the actuator will turn on if the temperature is over the threshold.

Conditions	Results
When the value reported by the sensor (28.2) is less than value of threshold (29) 	Relay Off
When the value reported by the sensor (29.1) is more than value of threshold (29) 	Relay On

NOTE

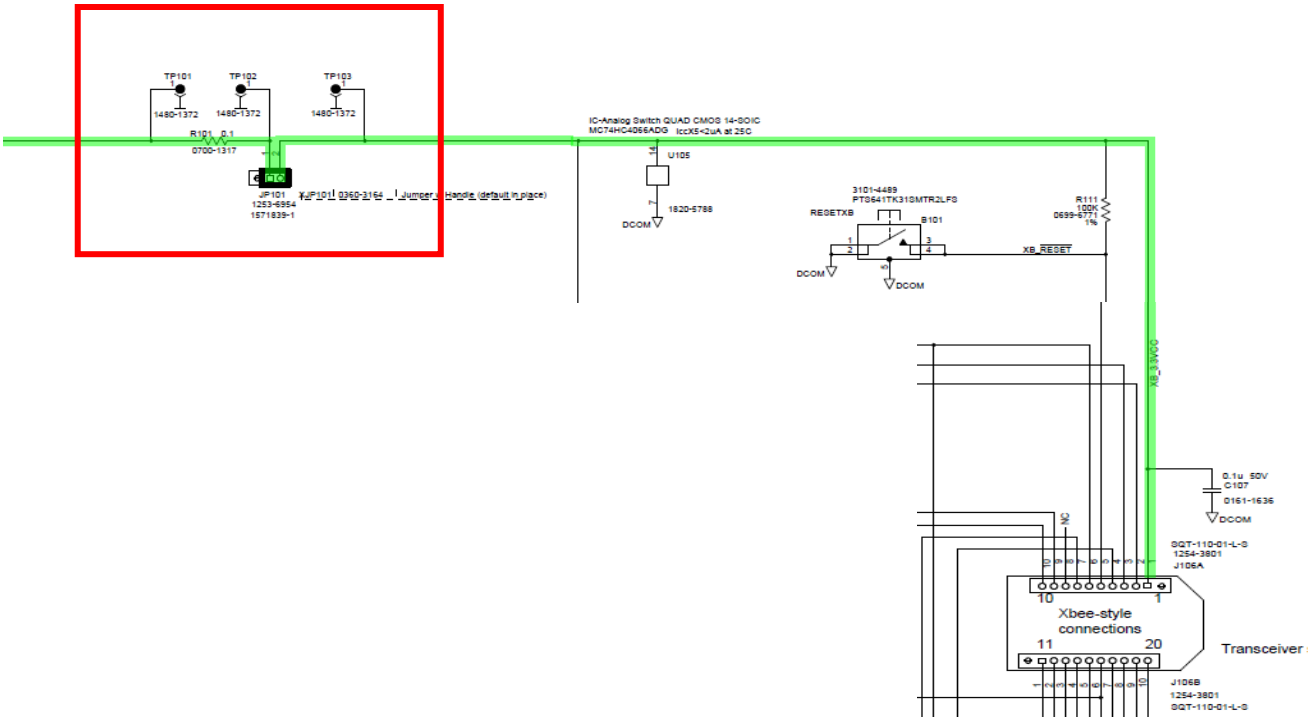
When the relay actuator is not switching, press **Ctrl + C** to stop running the code and run the code again. The ADC may sometimes need a retrigger to start working.

Exercise

a. Write a simple code to switch the relay actuator when the temperature is 26 °C.
b. Write a simple code to turn the relay actuator to NC when it is above the temperature threshold. Hint: Reverse the switching.
c. Write a simple code to set up an alert on the LCD when the temperature goes above 30 °C. You may use a hair dryer to simulate this event.

Task 3 – Measuring XBee3 Current Consumption

On the Keysight U3810A, current consumption is designed to be measured via the 0.1 Ω resistors. By measuring the voltage across the resistor and utilizing Ohms Law, you can analyze the current going through the resistor to XBee3 XB Transceiver.



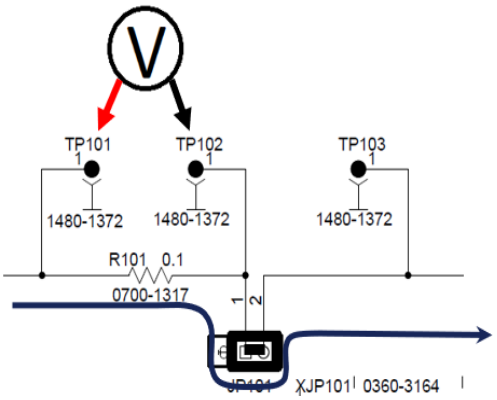
Using Ohm's Law: $\text{Current, I} = \frac{\text{Voltage,V}}{\text{Resistance,R}}$

WARNING

Do not connect voltages greater than 3.6 V to VCC or GPIO pins on the XB, XB1, XB2, LoRa or Lora1 Transceiver as this is the absolute maximum of their operating range and higher voltage may damage the device. The 3.6 Vmax pins include TPnn01, TPnn02, TPnn03, Jnn07 and Jnn08, where nn = 1, 2, 21, 22 and 23. Potential over-voltage sources include the VIN pin on the Arduino Shield and DC Power connectors, and +5VRAW and +5VSYS on interface connectors such as J10, JP55, and TP51.

Task 3a – Set Up Digital Multimeter for Current Measurement

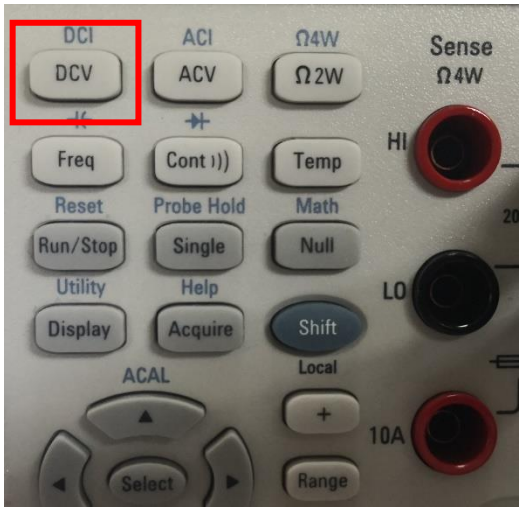
1. Set up the digital multimeter for DC current measurement. Connect the test leads to test pin TP101 and TP102 to the Keysight 34465A 6½ Digit Multimeter's Input V HI and Input V LO respectively.



The dark curved lines indicate the path of the current flow in the measurement circuit.

The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

2. Press the **DCV** button to measure DC voltage.



3. Press **Range > 100mV**.
4. Press **[Display] > Display > Trend Chart**.
5. Press **Vertical Scale > Vertical > Manual**.
6. Set **Low** to 0 mV.
7. Set **High** to 2 mV.
8. Press **Done**.
9. Press **[Acquire] > Acquire > Data Log**.
10. Set **Sample Interval** to 0.5 seconds.
11. Set **Duration** to 30 seconds.

Next, you will proceed to set up the XBee3 for periodic transmission.

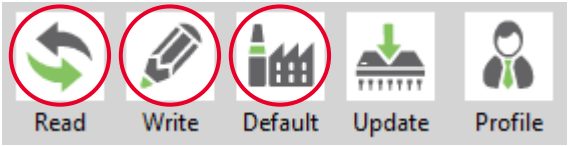
Task 3b – Set Up XBee3 for Periodic Transmission Mode

1. Connect the XBee3 XB Transceiver J103 to your computer via a micro-USB cable.

The diagram above might appear dark in print outs. Refer to Appendix B – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

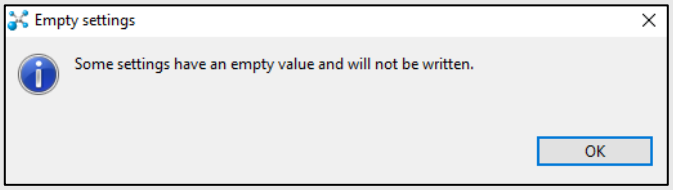
2. Use XCTU software to discover the XBee3 device and select the device to display its parameters on XCTU. You can refer to Task 1 – Interfacing the Analog Temperature Sensor to the XBee3 for more details.

3. Click the **Default** button and answer **Yes** to reset all the firmware settings to their default values. Click the **Write** button to perform the write operation, followed by the **Read** button to perform the read operation. Do this step for both the gateway XB and the sensor XB1 or XB2.



NOTE

Due to the empty **Zigbee Stack Profile (ZS)** by default, you will need to click **OK** when you see the prompt below. The **ZS** setting will remain as its last configuration setting.



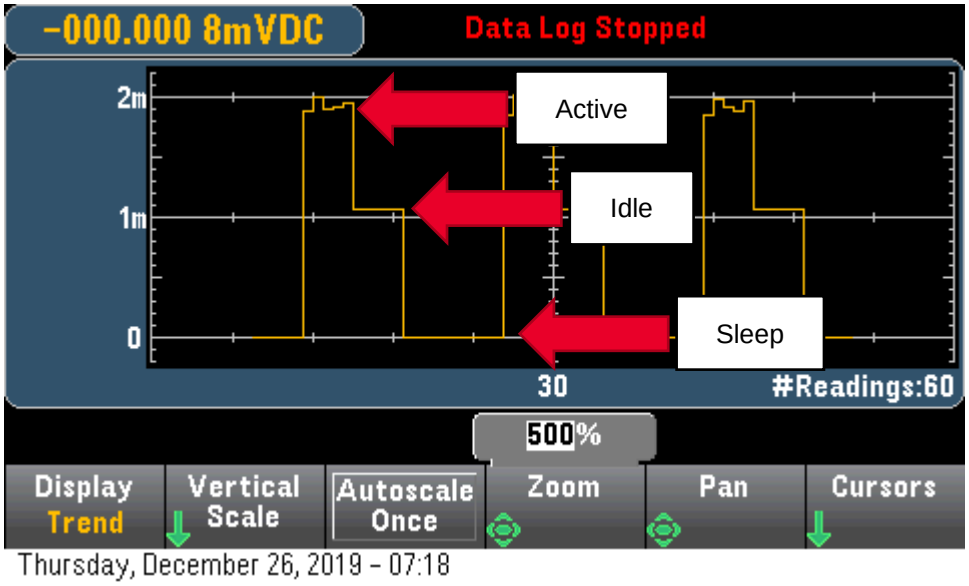
4. Configure the **Zigbee XB Transceiver (Gateway)** module on the U3810A according to the settings shown below:

Parameter	Value	Description
CE	Join Network [0]	Sets the device as a router or end device.
ZS	1	Set or read the Zigbee stack profile value. This must be the same on all devices that will join the same network.
JV	Disabled [0]	Verifies if a coordinator exists on the same channel to join the network or to leave if it cannot be found.
NI	DEVICE	Defines the node identifier, a human-friendly name for the module. The default NI value is a blank space. Make sure to delete the space when you change the value.
SM	Cyclic Sleep [4]	Sets the sleep mode of the Zigbee to Cyclic Sleep mode. This mode triggers transmission, followed by idle mode, and sleep mode periodically.
SP	1F4	Sets the Zigbee to wake sleep for 5000 ms after every transmission before waking up to transmit again.
ST	1388	Sets the Zigbee to sleep only after 5000 ms after every transmission. In other words, setting Zigbee to idle mode.
IR	7D0	Set the IO sampling rate to 2000 ms to enable periodic sampling.

5. Click the **Write** button to apply the settings to the XB on the U3810A. Click the **Read** button to read back the configurations.
6. Press **[Run/Stop]** on the digital multimeter to start the DC voltage measurement across the 0.1 Ω resistor. The digital multimeter will record the measurement for 30 seconds before it stops.

   Observe and record the DC voltage reading on the digital multimeter. Then, use Ohm's law to calculate the current going through the 0.1 Ω resistor. This is the DC current consumption for the XBee3 during Zigbee periodic transmission.

7. Set the **Zoom** to 500%. You would expect the measurement result as shown below.



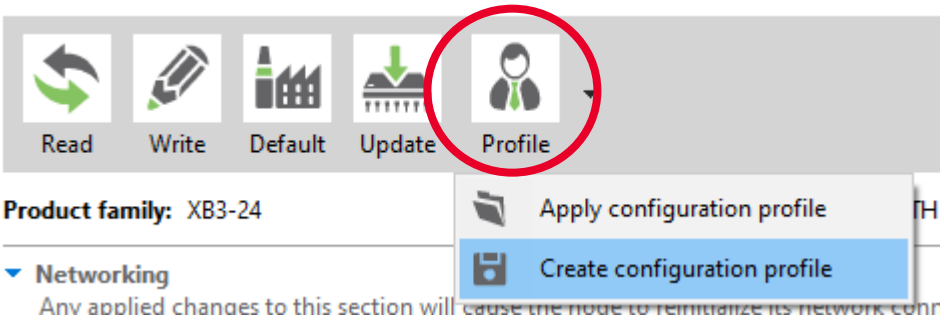
8. Use the **Cursors** feature in the digital multimeter and measure the DC voltage during active transmission, idle, and sleep states of the XBee3. Complete and calculate the current consumption in the table below.

DC voltage measurement during active transmission (mV)	
DC voltage measurement during idle state (mV)	
DC voltage measurement during sleep state (mV)	
DC current consumption during active transmission (mA)	
DC current consumption during idle state (mA)	
DC current consumption during sleep state (mA)	

9. Compare the data collected during sleep, idle and active states and calculate the XBee3 delta current.

XBee3 Delta Current between active and idle states (mA)	
XBee3 Delta Current between active and sleep states	
XBee3 Delta Current between sleep and idle states (mA)	

10. Click the **Profile icon > Create configuration profile** to save both XB device configuration (DEVICE).



Zigbee Device NI profile	Save the configuration file as
DEVICE	M1_L4_T3_DEVICE

NOTE

If you have not already saved these XBee3 configurations as profiles, you may save time and assure accurate configurations by copying and extracting the files from the lab code directory in the BeagleBone to your computer. You may then use the **M2_L4_T3_DEVICE.xpro** profiles for this task.

Go to **Profile** > **Create configuration profile** and select the profile.

The **xpro** file saved by XCTU is a zip format file. You can use zip software to unzip the file and examine the contents. Do not use XCTU versions older than 6.3.10 as configuration files for those versions were saved in XML format, which is incompatible with later releases.

Read the XCTU change log (**XCTU > Help > Change Log**) for more details.

Exercise

Compare the current consumption measurements during the different states (active, idle, and sleep states). Discuss your observations. Suggest how to improve the power consumption of IoT devices.

Task 4 – Resetting the Lab Code in BeagleBone

You have reached the end of this lab and will need to reset the lab codes for future lab sessions. This is to remove any modifications made in the lab. Run the following commands to reset the lab code.

```
cd ~
sh LabCodeReset.sh
```

NOTE

Resetting the lab code will erase any changes made in the LabCode directory. If there are any files that you want to keep, copy them out of the LabCode directory tree first.

References

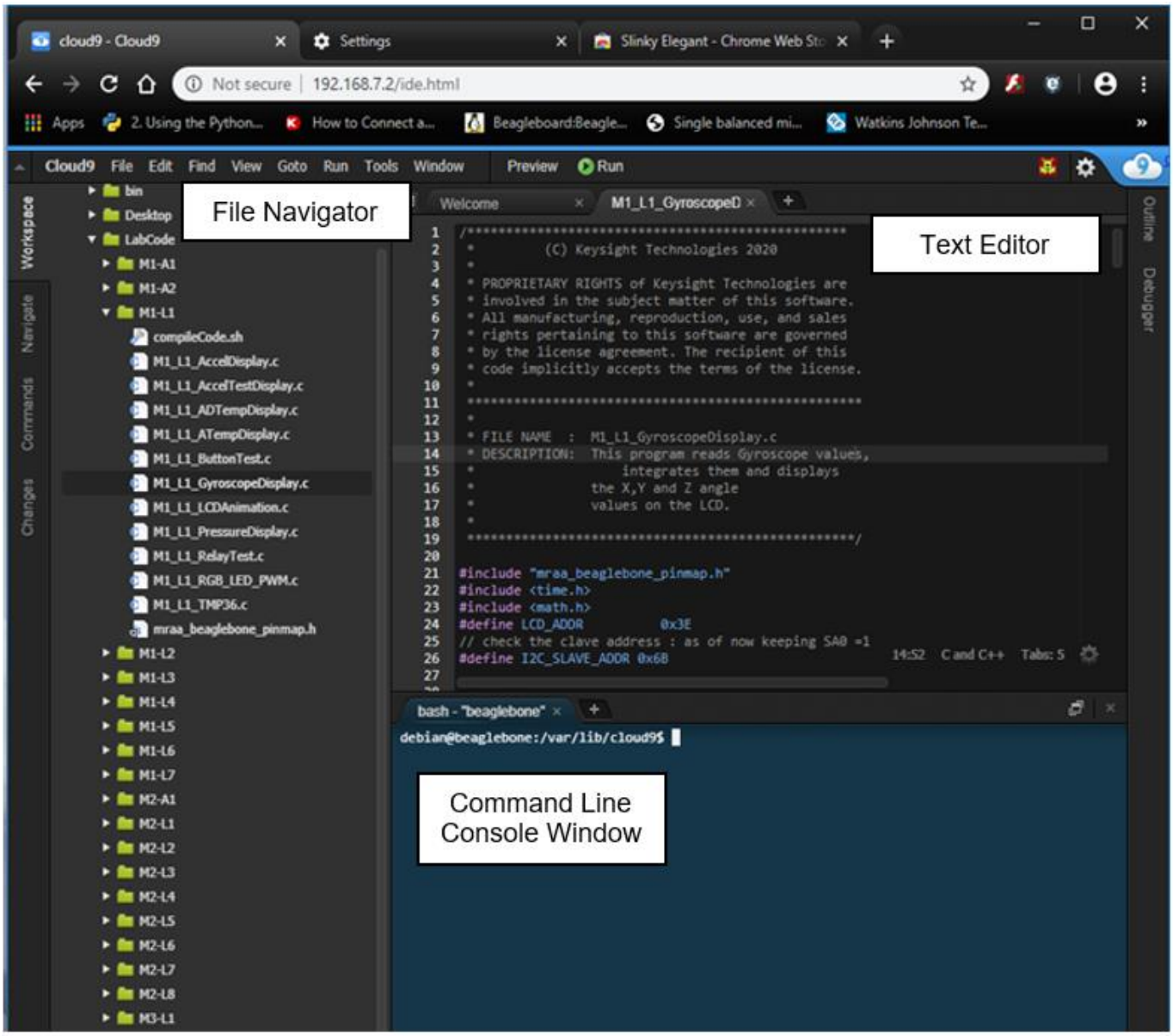
[1] LM75: Maxim Digital Temperature Sensor with Thermal Watchdog with Two-Wire Interface

[2] MPL3115A2: I^2C Precision Pressure Sensor with Altimetry

[3] MCP3204/3208: 2.7 V 4-Channel/8-Channel 12-Bit A/D Converters with SPI Serial Interface

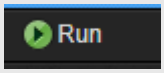
Appendix A – Cloud 9 IDE Usage

Over the RNDIS connection, the Cloud9 IDE (Integrated Development Environment) can be seen by opening a web browser to http://192.168.7.2. The default page or the last saved state for the IDE should come up. The page has three major sections which are the file navigator, text editor, and the console window.

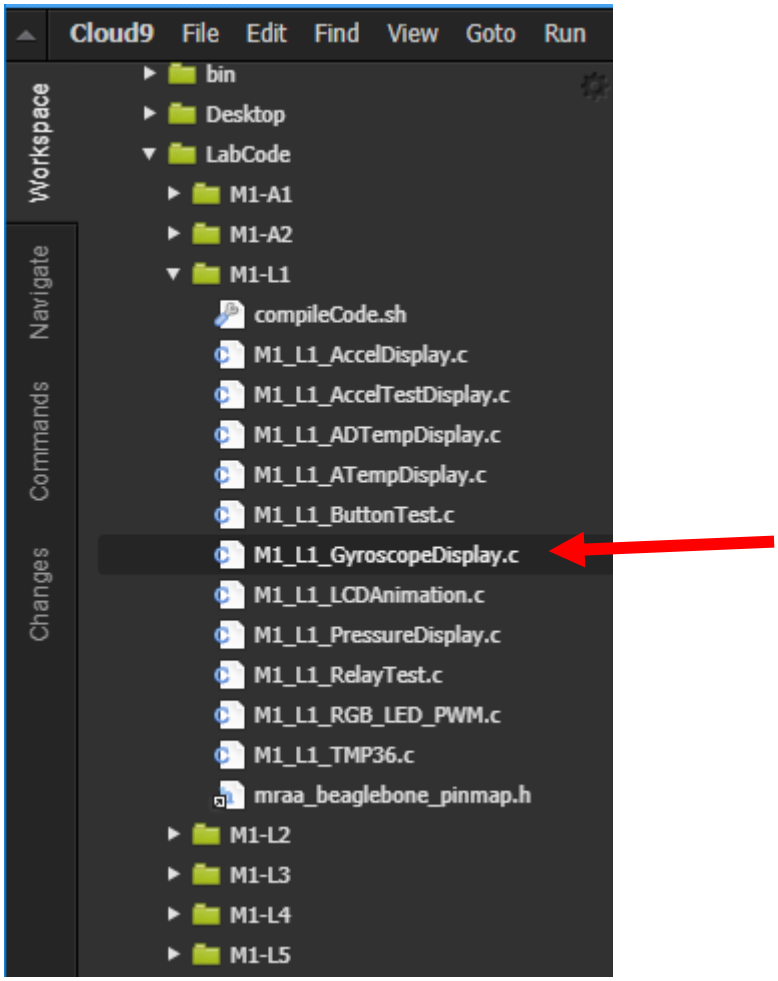


NOTE

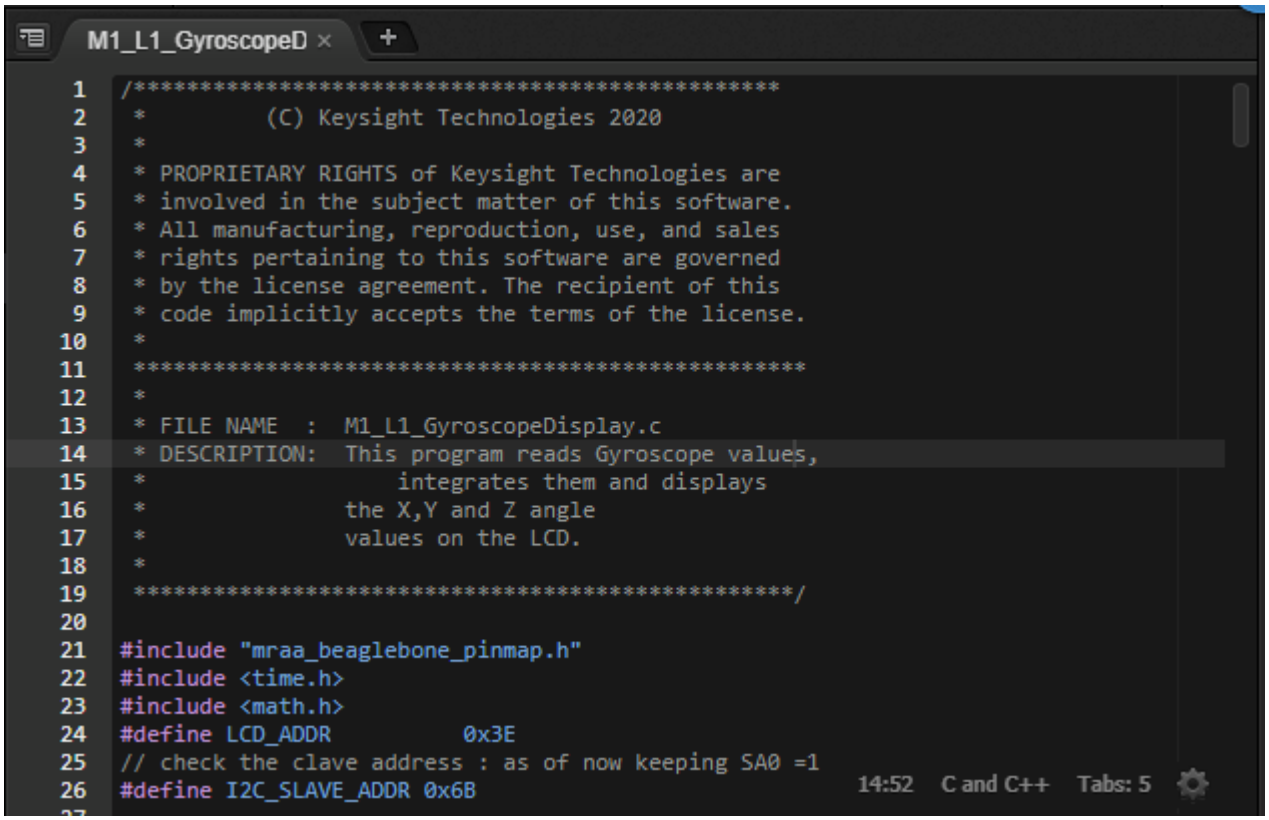
At the present time only .js and .php files run using the debugger mode.



1. Open a file in the editor and double-click the **M1_L1_GyroscopeDisplay.c** file in the File Navigator.



The Editor window should now show the file below. This is a very intuitive text editor that uses the conventional **Ctrl + C** to copy and **Ctrl + V** to paste. Once the file has been modified, go to the console window to compile it.

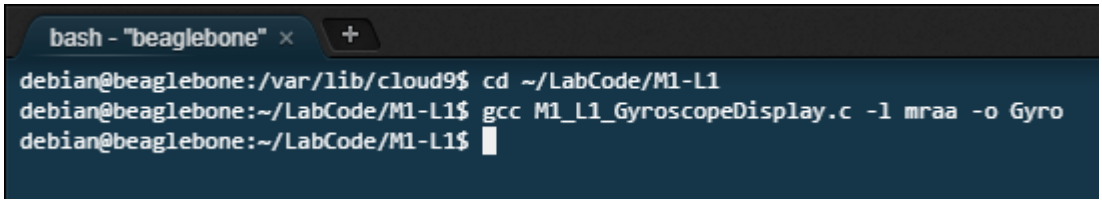


2. Run the following command in the console window to change to the directory that you are working in.

   `cd ~/LabCode/M1-L1`

3. Run the following command in the console window to compile the C code.

   `gcc M1_L1_GyroscopeDisplay.c -l mraa -o Gyro`

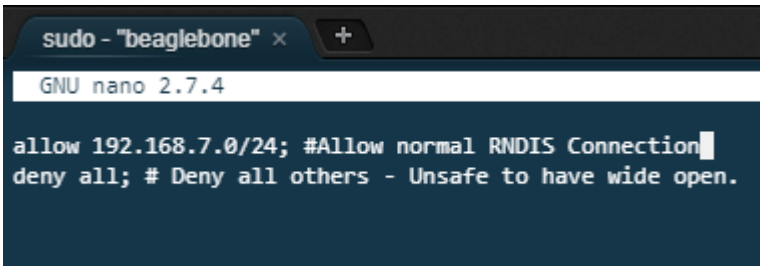


4. Enter **`./Gyro`** to run the code.

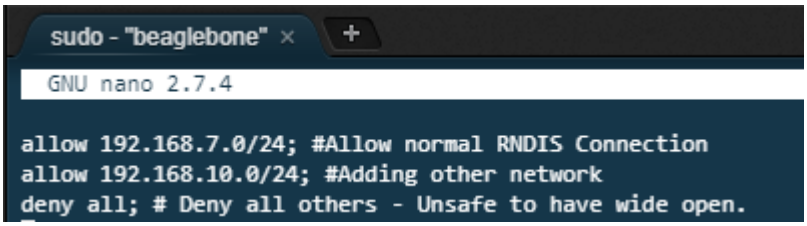
The Cloud9 IDE is secured so that it can only be accessed via the RNDIS port on 192.168.7.2.

5. In order to enable other network access, it can be opened using the **`nginx server.blacklist.`** This file is located at **/etc/nginx/server.blacklist**.
    a. To add other networks, add the networks to the **allow** section. This file will need to be edited with elevated permissions.

    `sudo nano /etc/nginx/server.blacklist`



    Once the file has been edited write it out and exit the editor.



    b. To allow the new network access, the nginx service will need to be restarted. To do this, run the command **`sudo service nginx restart`**. Access from other web browsers on the specified network can be made as long as the BeagleBone is connected to that network. Web browsers form different network locations will all see the same Cloud9 IDE. That is the information entered and display is the same. This works well for collaboration on problems. An instructor can open a browser window on a student IDE and help debug the problem.

WARNING It is strongly discouraged to enable all networks access to the Could9 IDE. It bypasses the login credentials.

Appendix B – Keysight U3810A Technical Documents

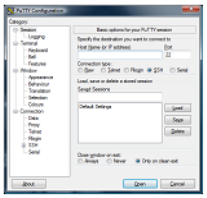
Board Component Locator and Schematic diagrams are available in the BeagleBone in this folder **…/LabCode**.

I. Board Component Locator diagram - /home/debian/LabCode/asm_U3810-66501.pdf
II. Schematic diagram - /home/debian/LabCode/U3810-66501.pdf

Transfer these to your computer using WinSCP for your use.

Appendix C – Establish Serial Communications between BeagleBone and PC

1. If not already done so, download and install PuTTY from http://www.putty.org/. Choose the 32-bit or 64-bit, whichever is compatible with your operating system.

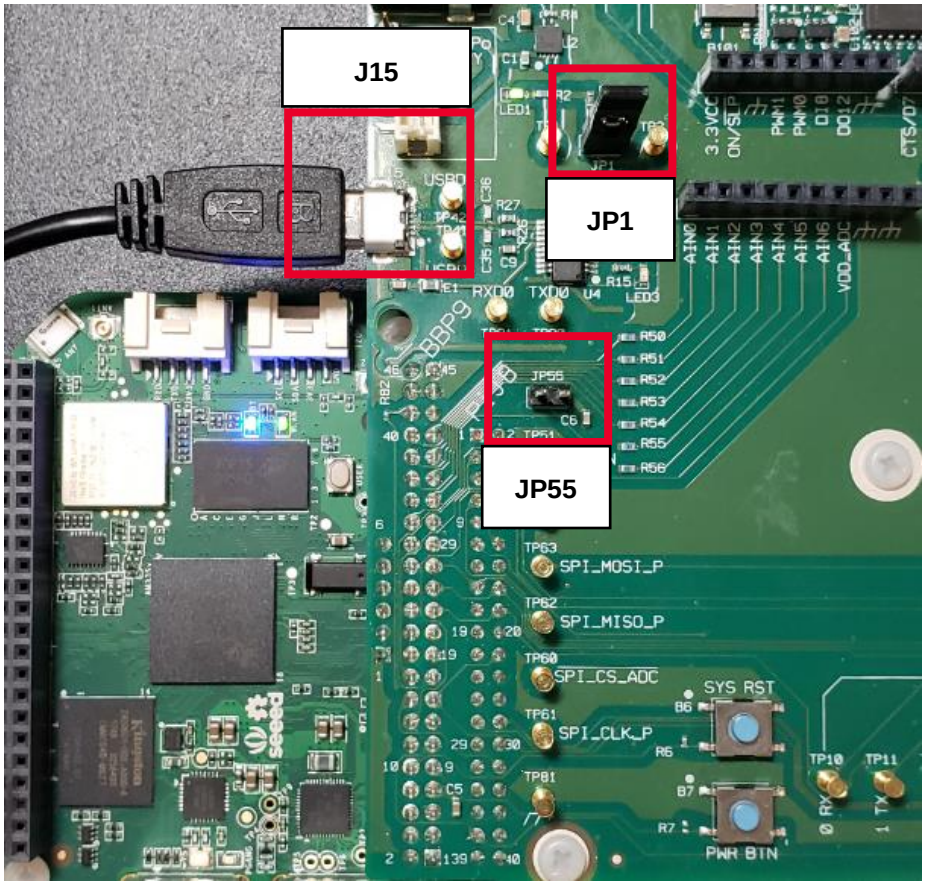


Download PuTTY

PuTTY is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform. PuTTY is open source software that is available with source code and is developed and supported by a group of volunteers.

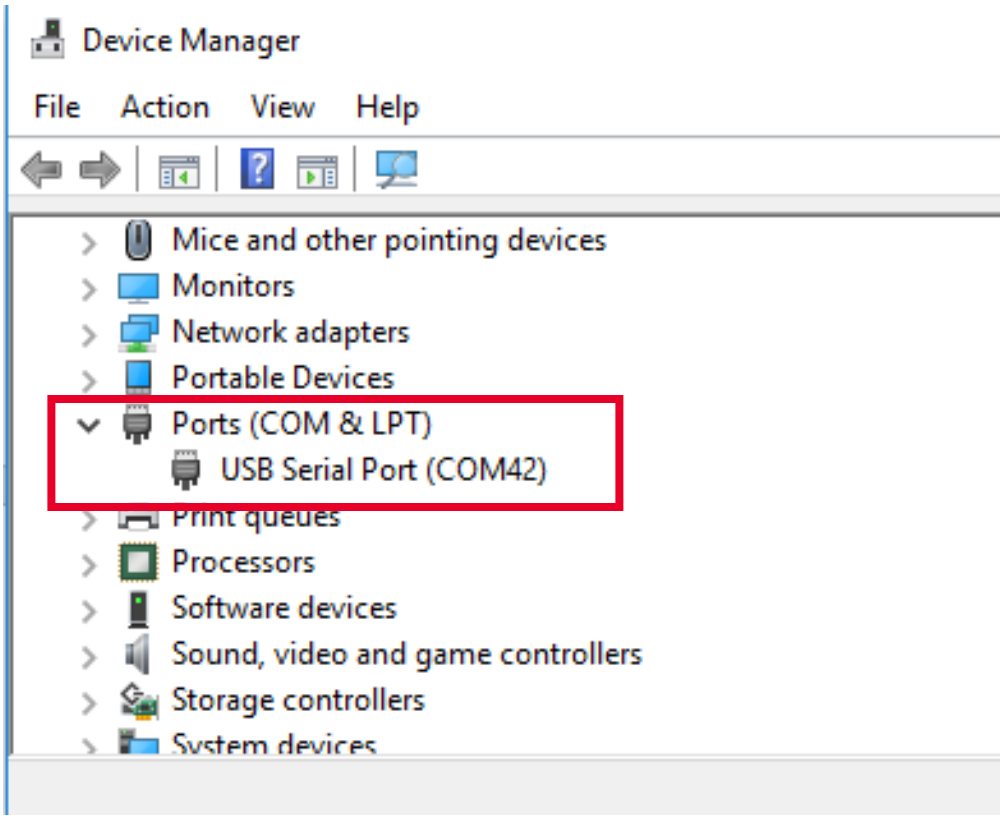
You can download PuTTY here.

2. Only connect a USB cable from your computer to J15 of the U3810A. The JP1 should be in place while the JP55 should not be placed.

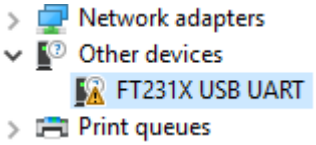


The diagram above might appear dark in print outs. Refer to Appendix E – Keysight U3810A Technical Documents for the searchable PDF to help you locate the locations of the jumpers, connectors and components.

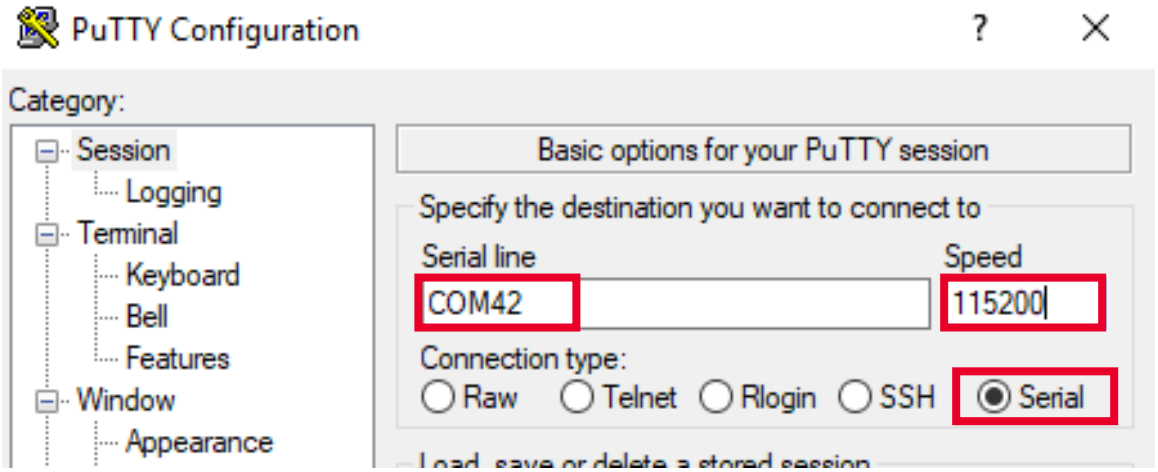
3. Press the Windows key and type Device Manager or type the Run and type **devmgmt.msc** to open the Device Manager. Find the COM port that connects to U3810A. Note down the port number. You will need this to configure the serial communication using PuTTY.



The warning below indicates that the driver was not automatically installed by Windows.



4. Run your PuTTY software and connect to the COM port that you have identified earlier using Serial connection at 115200 Baud-rate.

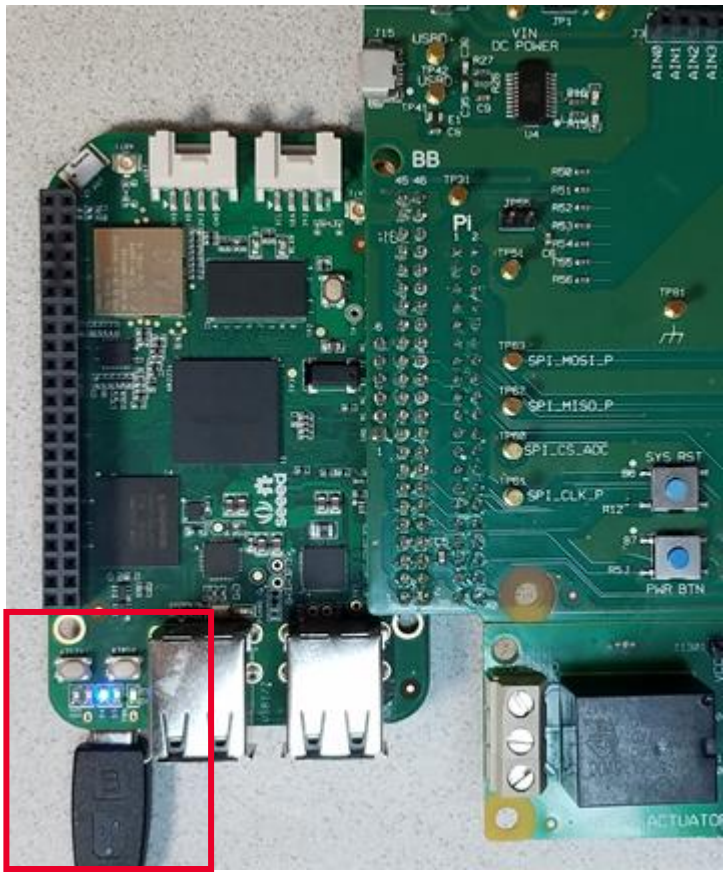


Appendix D - Establish Secure Shell (SSH) Communication between BeagleBone and PC

In this exercise, you will connect the PC (Host) to Beagle Bone via a USB cable and establish a RNDIS connection. RNDIS is the Remote Network Driver Interface Specification. It defines internet connection via USB and this connection provides a virtual network to the Beagle Bone that supports various network protocols including SSH and HTTP. Once the connection is established, a PuTTY terminal using SSH can be used. The local documentation on the webpage can also be explored. The RNDIS Network IP address of the BeagleBone will be **192.168.7.2** while your PC will be at **192.168.7.1.**

WARNING When JP1 is in place, do not connect a USB cable to both the BeagleBone and J15 at the same time or anomalous behavior may result.

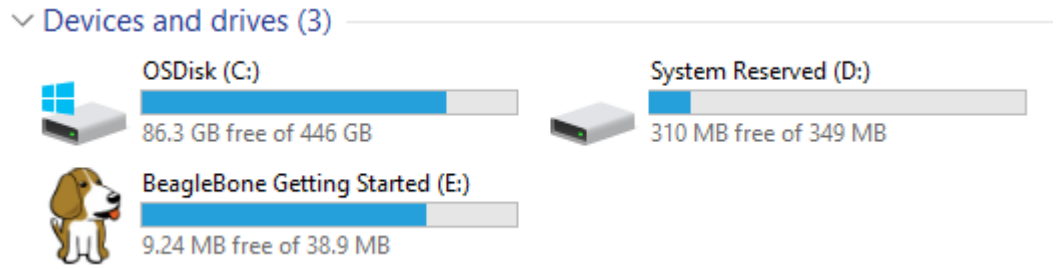
1. Remove the USB cable from J15 and connect it to the BeagleBone CPU USB port to your PC. This will also power up the U3810A. It may take up to one minute to complete the boot process.



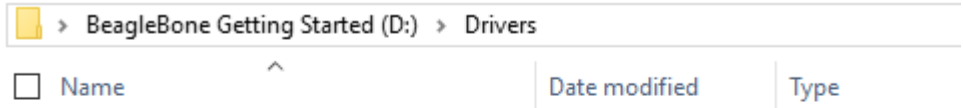
Once the boot process is complete, you should see the BeagleBone Getting Started window or drive listed in the directory shown in the next image.

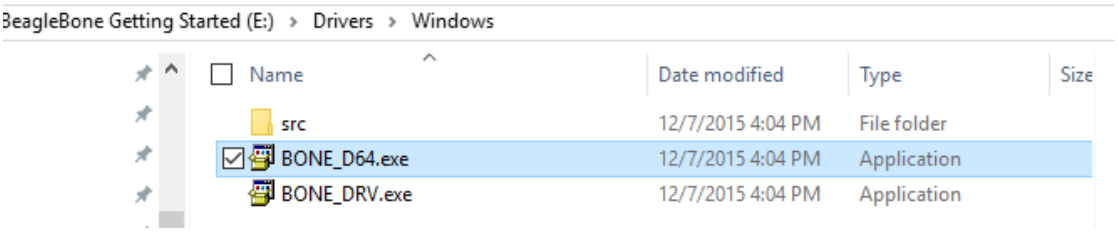
Install RNDIS drivers

2. If the drivers have not already been installed, open the **BeagleBone Getting Started** drive using File Explorer. Select the driver for your OS from the Drivers folder and install.

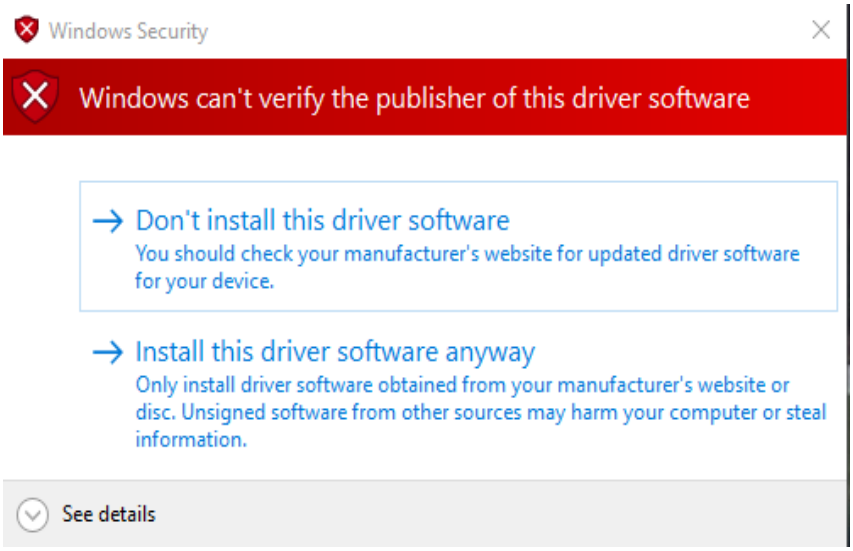


3. Select the driver for your OS from the Drivers folder and install the BONE_D64.exe file.

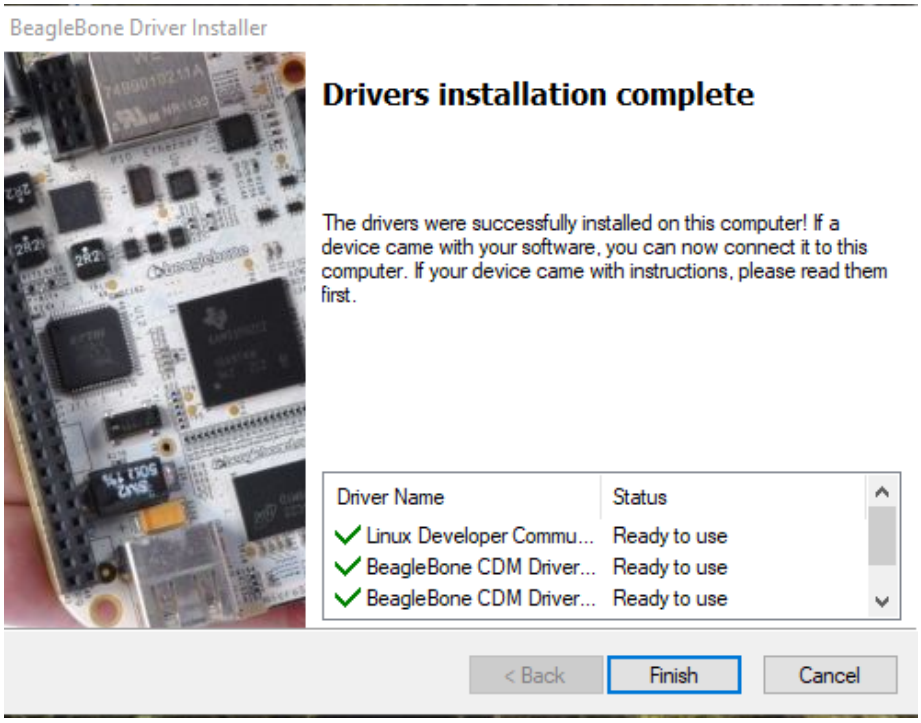




4. During the installation, Windows 10 users may see this message. Select the **Install this driver software anyway**.



Successful installation will show the message below.

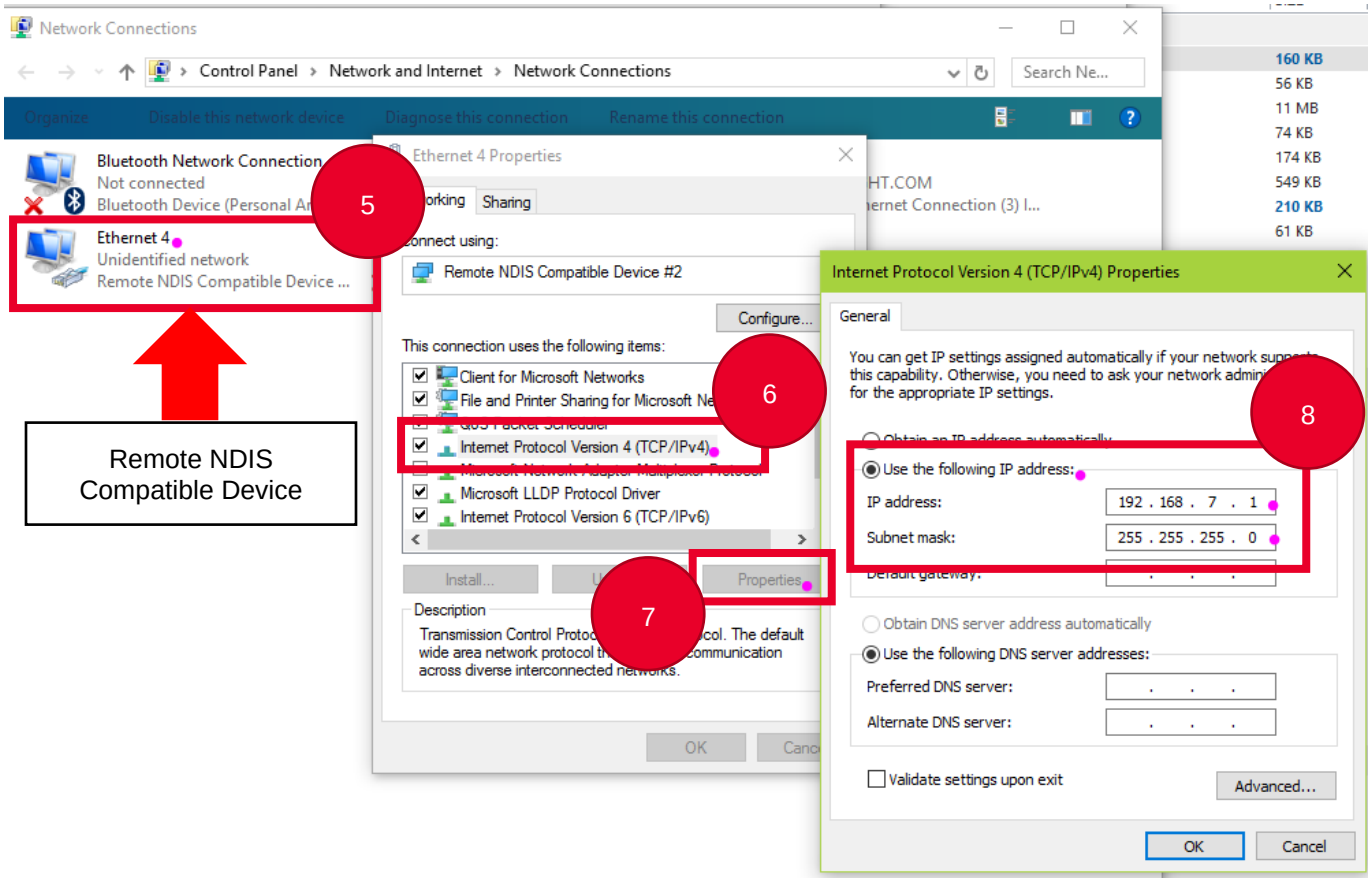


Configure RNDIS adapter

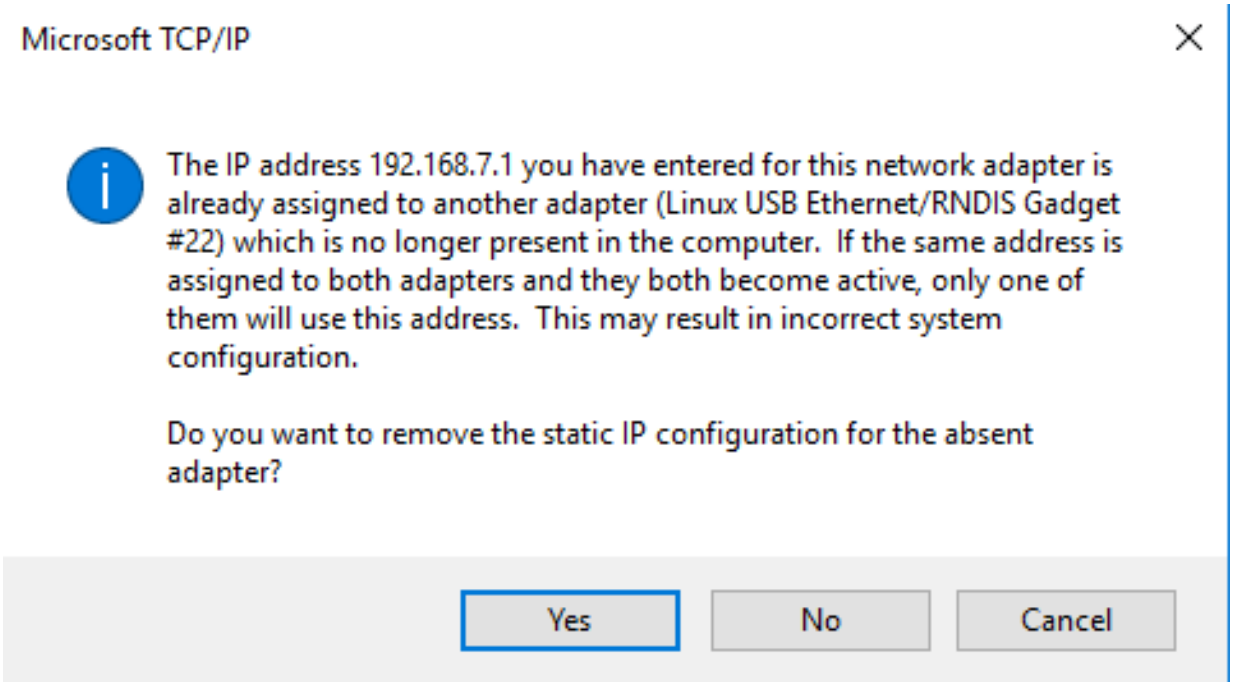
Your PC will need to be on the same subnet using the RNDIS connection. This does not have DHCP so your PC address needs to be set to **192.168.7.1**.

> **NOTE**
>
> You may need to run this step each time a different BeagleBone is connected to the PC.

5. Go to **Network Settings**, right-click Remote NDIS Compatible device, and select Properties.
6. Click **<your Remote NDIS Adapter>** and click **Internet Protocol Version 4 (TCP/IPv4)**.
7. Click **Properties**.
8. Set up the general settings for the IP address as shown below.

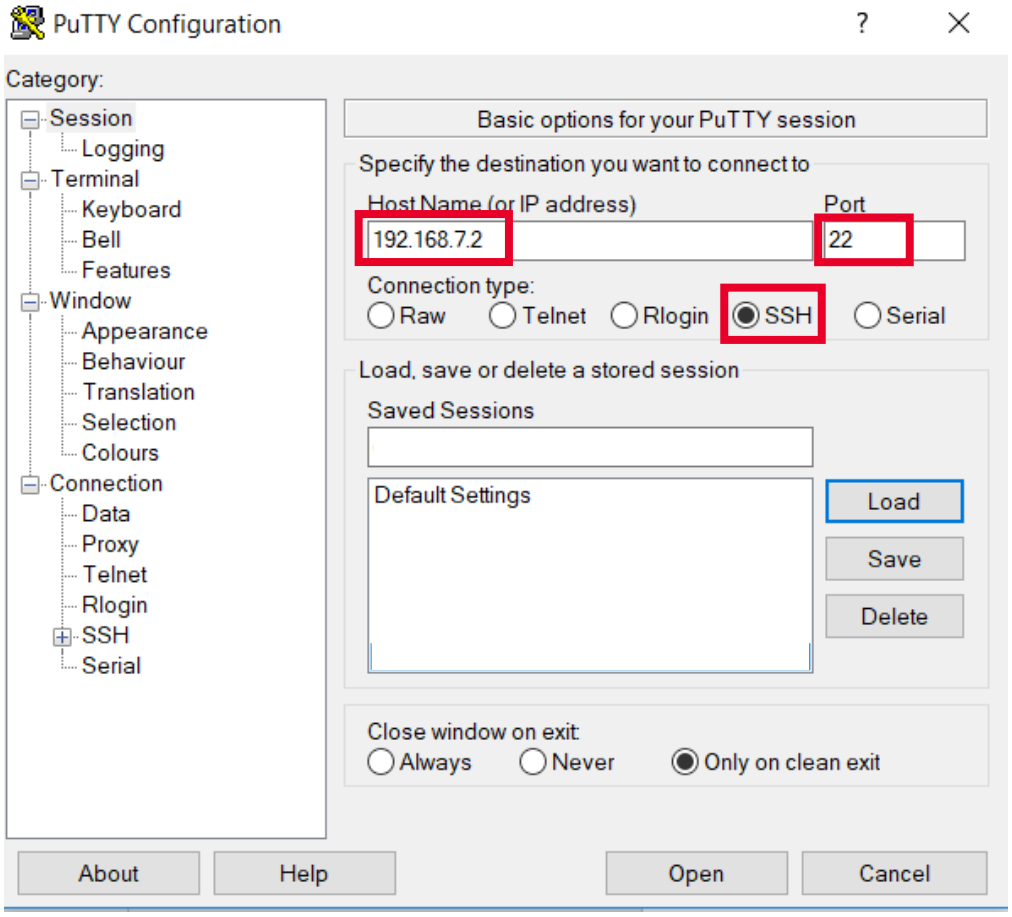


You may receive the following message if you have previously connected a BeagleBone or other device on this address. Click **Yes** when the other device is not in use and click **No** when both devices are not present.

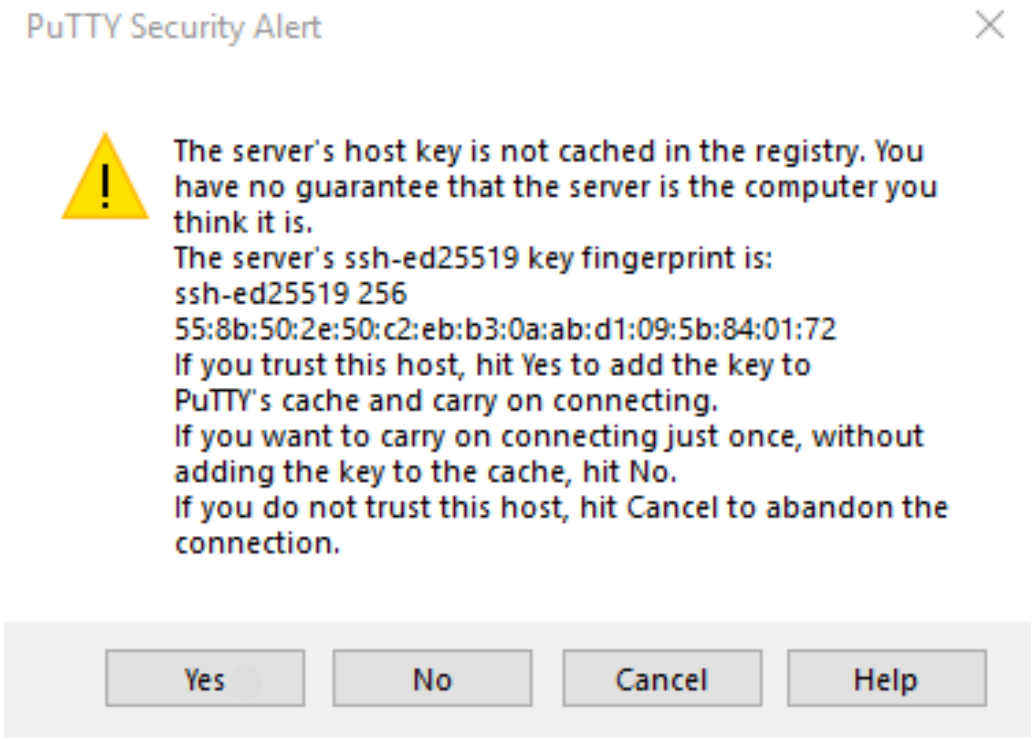


Set Up SSH connection

9. Once the ping command comes back with a reply and a response time, double-click the PuTTY.exe to launch the PuTTY terminal program.
10. A PuTTY Configuration window will pop up to determine the connection type. Enter **192.168.7.2** as the IP address, select **SSH** as the Connection type and Port **22**.



11. You will receive the following message if this is the first time your PC is connecting to the BeagleBone. Click **Yes**.



12. Click **Open** to open the terminal window. Press **Enter** on your keyboard to check and verify connectivity. Otherwise, refer to **Getting Started Guide** to upgrade the firmware.

Default username : debian
Default password : temppwd

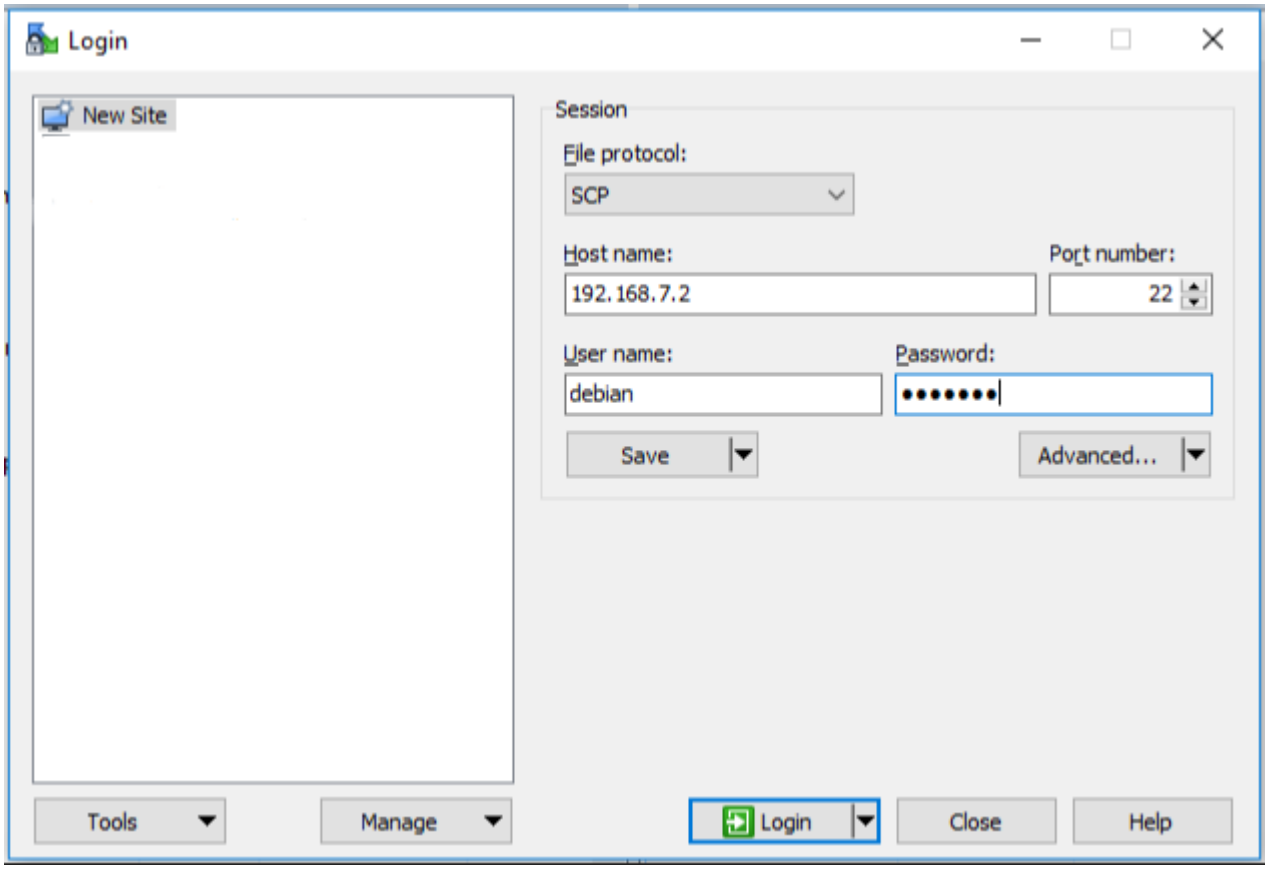
Appendix E – Copy and Edit Files with WinSCP

NOTE

After power or reset, the boot process may take some time to complete before the 192.168.7.2 port becomes active.

Set Up WinSCP

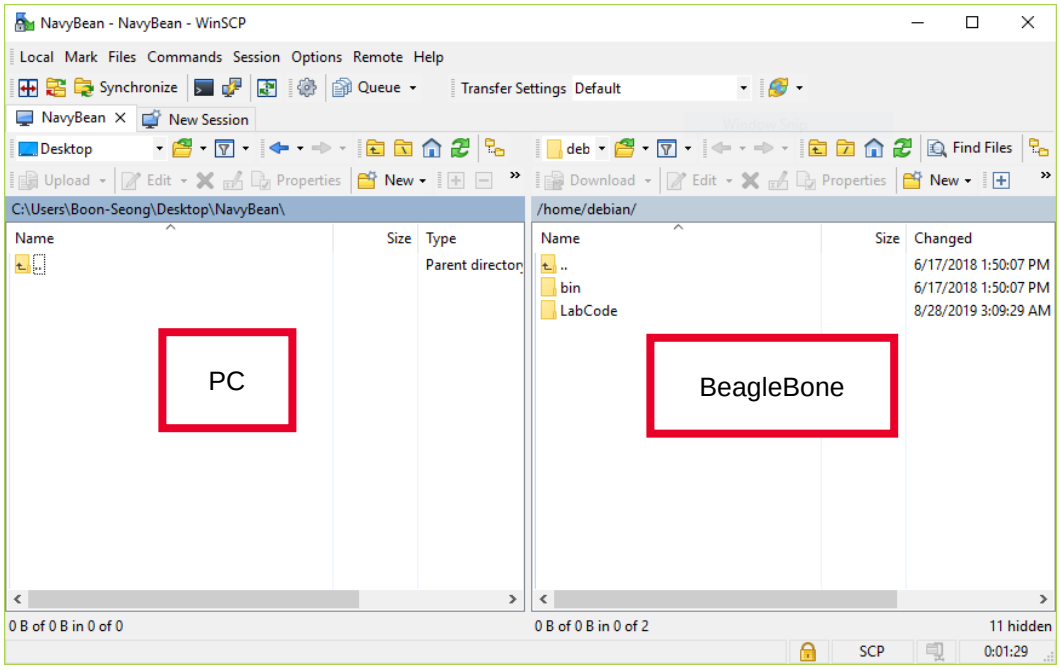
1. For Windows users, download and install a copy of WinSCP from https://winscp.net/eng/download.php. You should see a WinSCP icon on your desktop.
2. Launch WinSCP and click **New Site**. Select **SCP** protocol, enter **192.168.7.2** for Host Name, Port Number **22**, **debian** as username, and **temppwd** as the password.



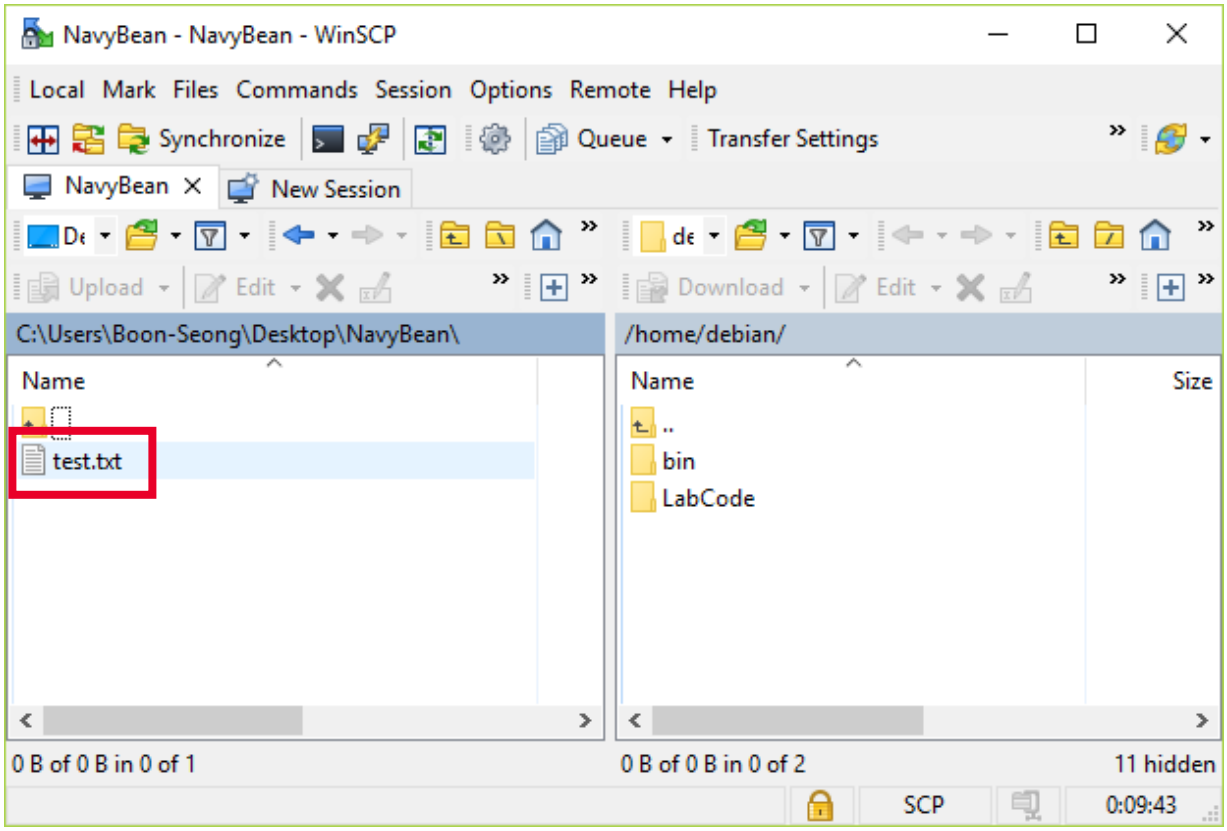
3. First time users who are connecting WinSCP to the BeagleBone, select **Yes** when prompted with a message about connecting to an unknown server.

Copy Files with WinSCP

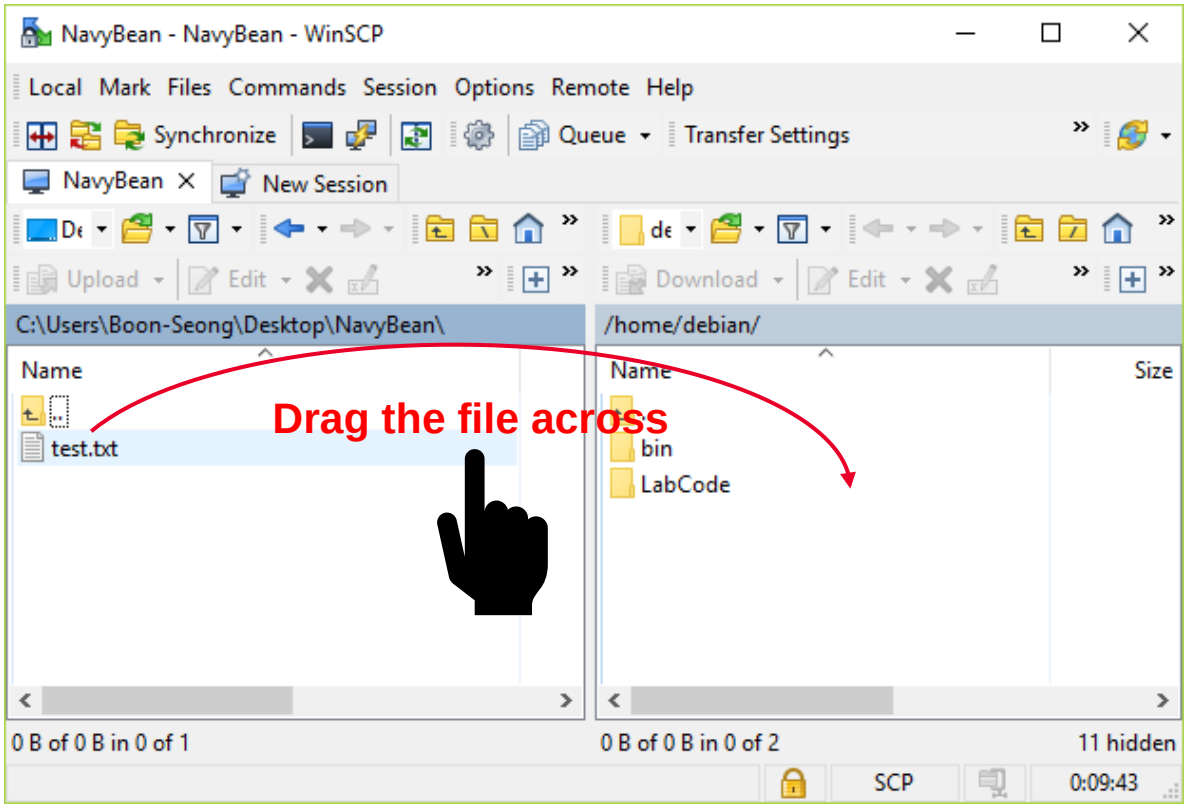
You should see the GUI below where you can drag files across, to transfer it from the PC to the BeagleBone and vice-versa.



4. Create a “test.txt” text file in your desktop.



5. Drag the “test.txt” file across to copy it over to the BeagleBone using WinSCP.

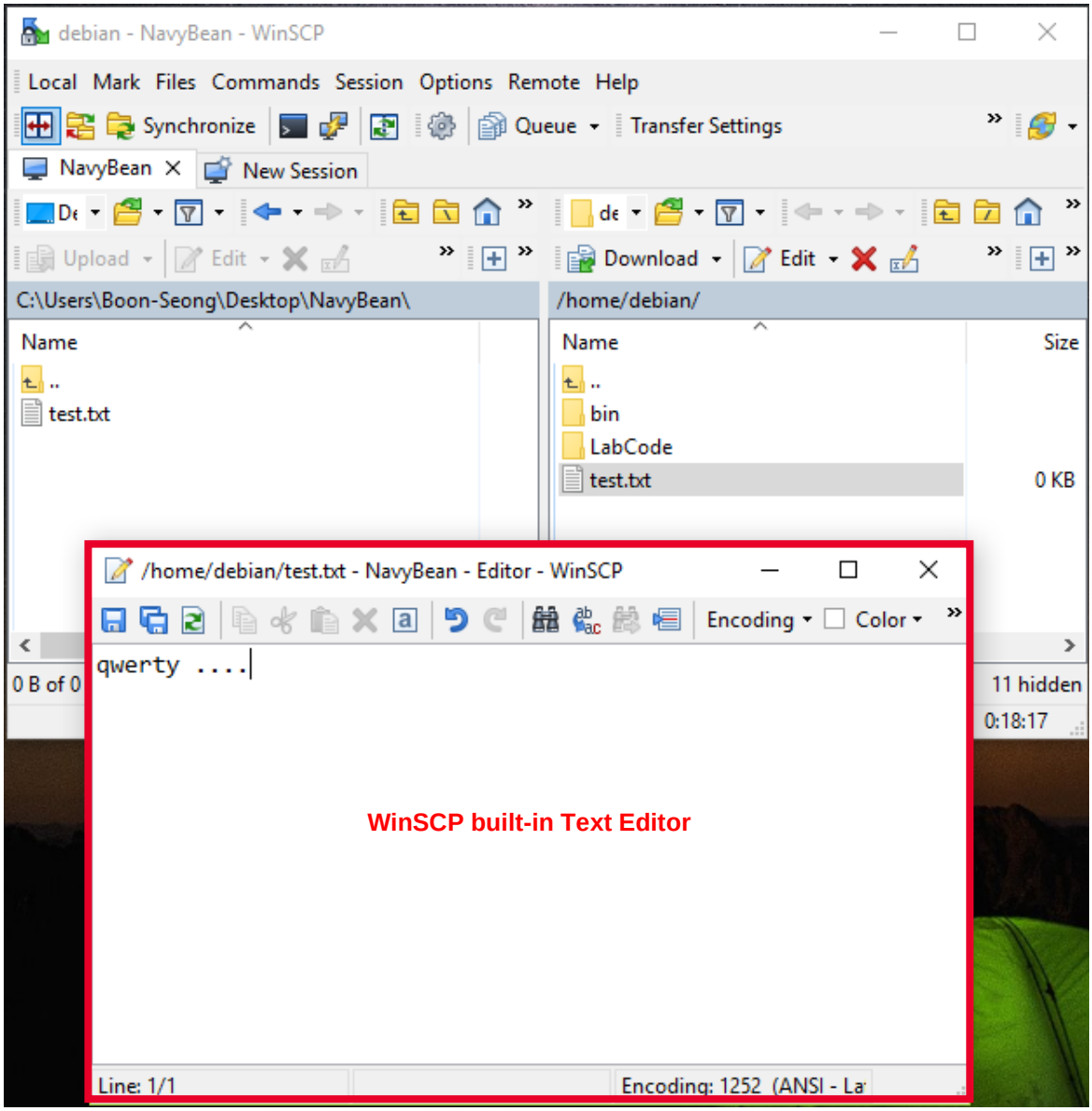


NOTE

For Linux based systems, copy the file using **scp M1-L1.zip debian@192.168.7.2** command.

Edit Files with WinSCP

6. With the file in BeagleBone, right-click the “test.txt” file and click **Edit**. It should prompt a built-in text editor where you will use it to edit shell scripts with a GUI text editor from PC.



7. Save your changes and close the text editor.

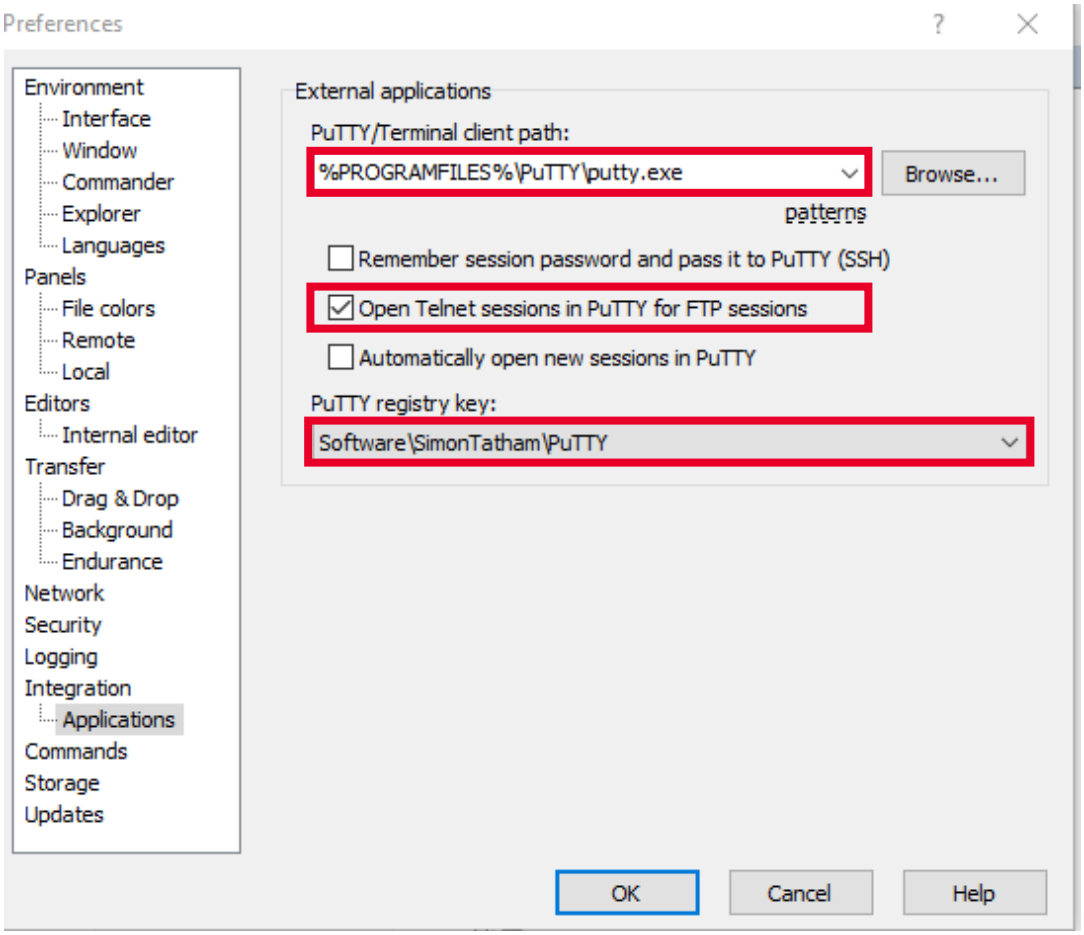
NOTE

It is recommended to save your changes frequently as you edit the file, to minimize the risk of losing your changes when there are disconnection between your PC and BeagleBone.

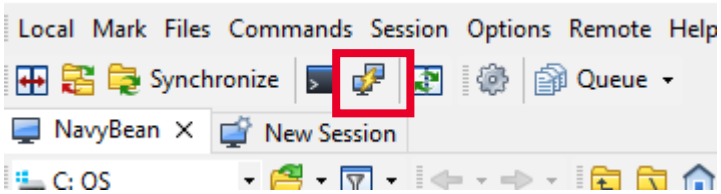
Start PuTTY SSH connection from WinSCP

With the PuTTY software installed in your Windows PC, you can integrate PuTTY to WinSCP where you can easily start an SSH connection with BeagleBone without having to set up the connection properties in PuTTY.

8. In your WinSCP window, go to **Options > Preferences**.
9. Go to **Integration > Applications**.
10. In the window below, verify the following settings. Click **OK** to apply.



11. Click the button below and PuTTY will automatically log in to BeagleBone with the credentials used in WinSCP.



This information is subject to change without notice.

© Keysight Technologies 2020
Edition 1, June 2020

Published in Malaysia

www.keysight.com