

## Multi-hop Communication with Xbee S2C

If the implementation of point to point communication is already done, it is time for the multi-hop communication.

For each group, we will be considering a very small multi-hop network of 5 node. Among these 5 nodes, one will act as the Coordinator node (central node), one as Router node (for relaying the data over long distances), and three IoT Edge nodes.

Nodes	Configuration
Coordinator	CE=1 SM=0
Router	CE=0 SM=0
End Node	CE=0 SM=1

### ***Coordinator:***

- You have already set up the coordinator node in the previous lab. **Change the PAN ID to a newer value (It's a must) and note down the PAN ID of your coordinator. Set the SC of the coordinator as 7FFF and Scan Duration -SD as 7FF.**
- **Put the Destination Address DH=0 and Destination Address DL =FFFF. This will set the coordinator in broadcast mode.**
- **Write the changed settings.**

### ***Router:***

- Now set up the **router node** as you set up the End node before. Router settings are CE→0 and SM →0, SC→7FFF, SD → 7FF. Keep the same PAN ID as your coordinator (New one).
- Give the parameter →'NI' a name like 'Router 1'.
- Write the changed settings.

### ***Coordinator - Router Communication:***

- Now, if you go to the XCTU terminal and close the serial communication. You should be able to make the router and the coordinator talk to each other over RF with keyboard inputs.

## ***Multi-hop-Setup:***

In multi hop network, End nodes usually send the data to the coordinator directly if it is within the range of the coordinator. And if it goes out of the range of the coordinator, then it looks for the 'router node' to relay the data from end node to the coordinator.

- But due to space constrain of the lab, we will be forcing the end node to relay all the data through router to send it to the coordinator.
- For this, go to the coordinator settings again and **set the Scan Duration, SD=0**. Write the settings. But keep the router as it is.
- After you set up the router and coordinator, plug them out. Now set up the End node as before with the PAN ID same as coordinator and router. If needed, follow the first ab document to set it up.

If you have come up to this, you have probably set up the simplest mesh network. Time to evaluate it.

- Connect coordinator, router and end node. Give it 20-30 seconds to setup, connect and synchronize. Now if you write anything on the XCTU terminal of end node it should be printed to the Coordinator terminal in red.
- Try this few times. If everything's okay, disconnect the router and connect it to a remote power source (power bank). You should be able to communicate as before.
- Disconnect the router from the power source. Try to communicate between end node and coordinator. But you should not be able to communicate at this point if you have done everything alright.
- It is because the data is routed through the router node to the coordinator. If the router node id not powered up, you should not be able to make any communication.

## ***Setting Up Arduino:***

Now again,

Write the hello world code in Arduino such that, it prints "Hello from Node 1" in every five seconds .

Connect the Arduino to the Xbee End node:

<b>Arduino</b>		<b>Xbee S2C</b>
----------------	--	-----------------

TX	➔	Din
RX	➔	Dout
5V	➔	5V_in
Gnd	➔	Gnd

- Provided the router, Arduino and coordinator are powered up, you should see “Hello from End Node 1” in every five seconds.
- Build another End Node in the same way as instructed. write the code to print “Hello From End Node 2” in every 5 seconds. It should be automatically connected to the network when you are done.
- You can power up your Arduino with the power bank. And also keep the router connected to the power bank.
- Keep the coordinator connected to your PC. In any serial monitor – XCTU terminal / Arduino serial monitor / Putty, you should see two printed strings in every five seconds ➔

“Hello from End Node 1”

“Hello from End Node 2”

Best of luck