**Channel Notching in NetSim**

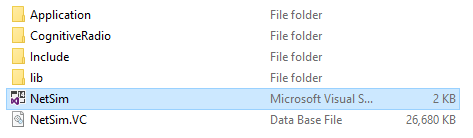
**Software Used:** NetSim Standard v10.1 (64-bit), Visual Studio 2015.

**Note:** Channel notching code will work only for a single Channel, single CR-CPE and for at-most one Incumbent.

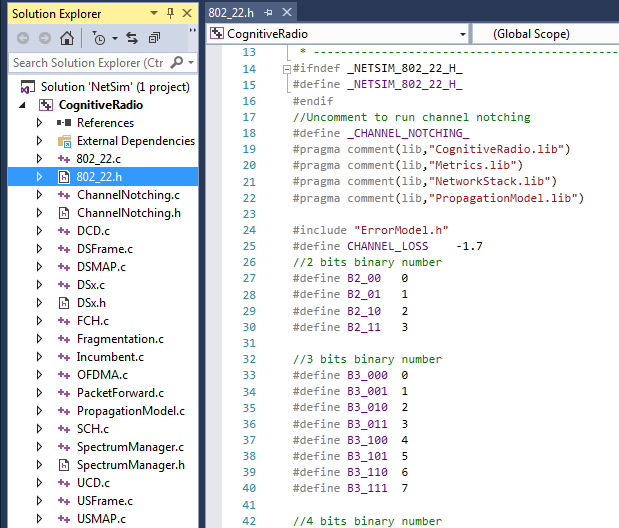
# Steps:

1. Open the **NetSim.sln** folder from Code folder that is present inside the extracted Channel Notching folder.

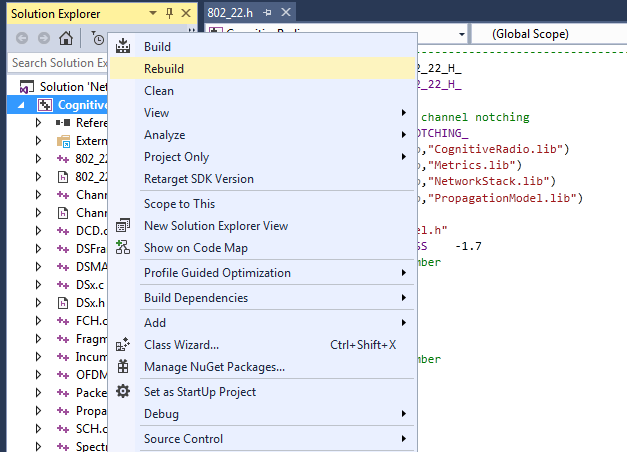




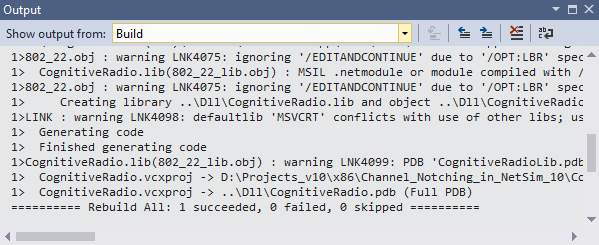
1. In the Solution Explorer, go to **Cognitive Radio 🡪 802\_22.h** and open it.
2. If you want to enable **Channel Notching**, uncomment (if commented)

#define \_CHANNEL\_NOTCHING\_

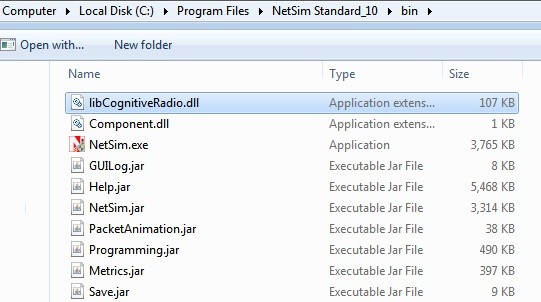
1. Right click on **Cognitive Radio p**roject🡪 **Rebuild**



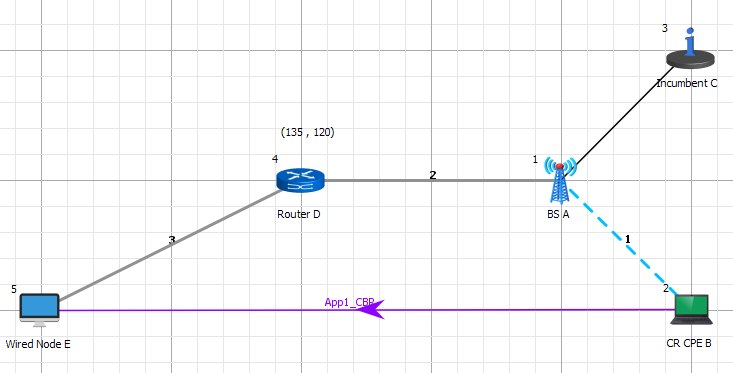
1. You should see a message in the **Output** window as shown in the following figure.



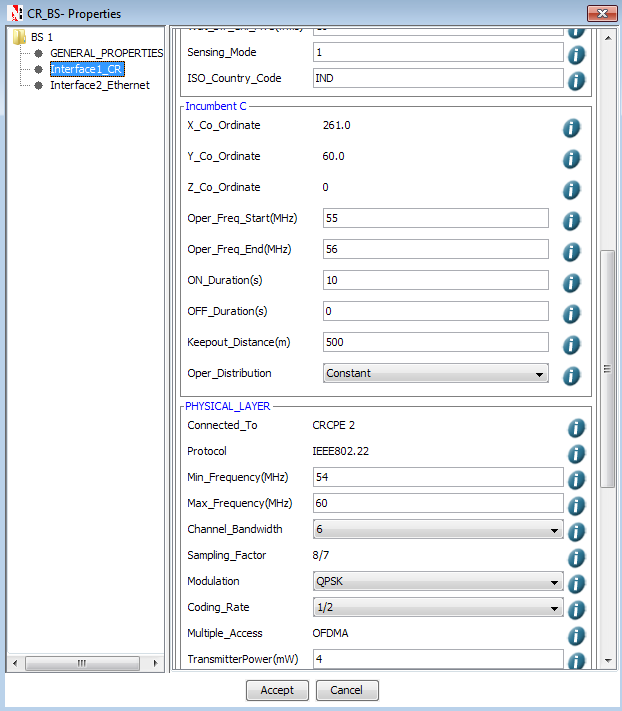
1. Inside the extracted **Channel Notching** folder, go to **DLL** folder and copy the file **libCognitiveRadio.dll**



1. Go to the installed directory where NetSim is installed and open **bin** folder.( It will be something like **“C:\Program Files\NetSim Standard\bin”)**
2. Rename **libCognitiveRadio.dll** file which is already present there to something like **libCognitiveRadio\_default.dll** for backup purpose. In case if something goes wrong, you can again rename this file to **libCognitiveRadio.dll**.
3. Paste the new **libCognitiveRadio.dll** file that you copied from the **DLL** folder.
4. Now executing Cognitive Radio in NetSim will including **channel notching.**
5. A sample **Configuration.xml** file is provided inside the zip folder.
6. Or create a scenario as followed



1. And set BS properties as



1. Open that scenario in NetSim and run it for both the cases: with channel notching and without it.
   * To run without Channel Notching- rename the present **libCognitiveRadio.dll** to something like **libCognitiveRadio\_notching.dll** and rename original dll to **libCognitiveRadio.dll (**earlier renamed to **libCognitiveRadio\_default.dll).**
2. The throughputs obtained will be **0.009694** and **0.000000** respectively.

# Explanation:

Channel Notching basically allows the CR-CPE to use the free subchannels which are not occupied by the incumbent, as compared to the standard case (without channel notching), where the entire channel is blocked even if some of the subchannels of that channel are being used by the Incumbent.

Hence enabling Channel Notching will give more throughput.