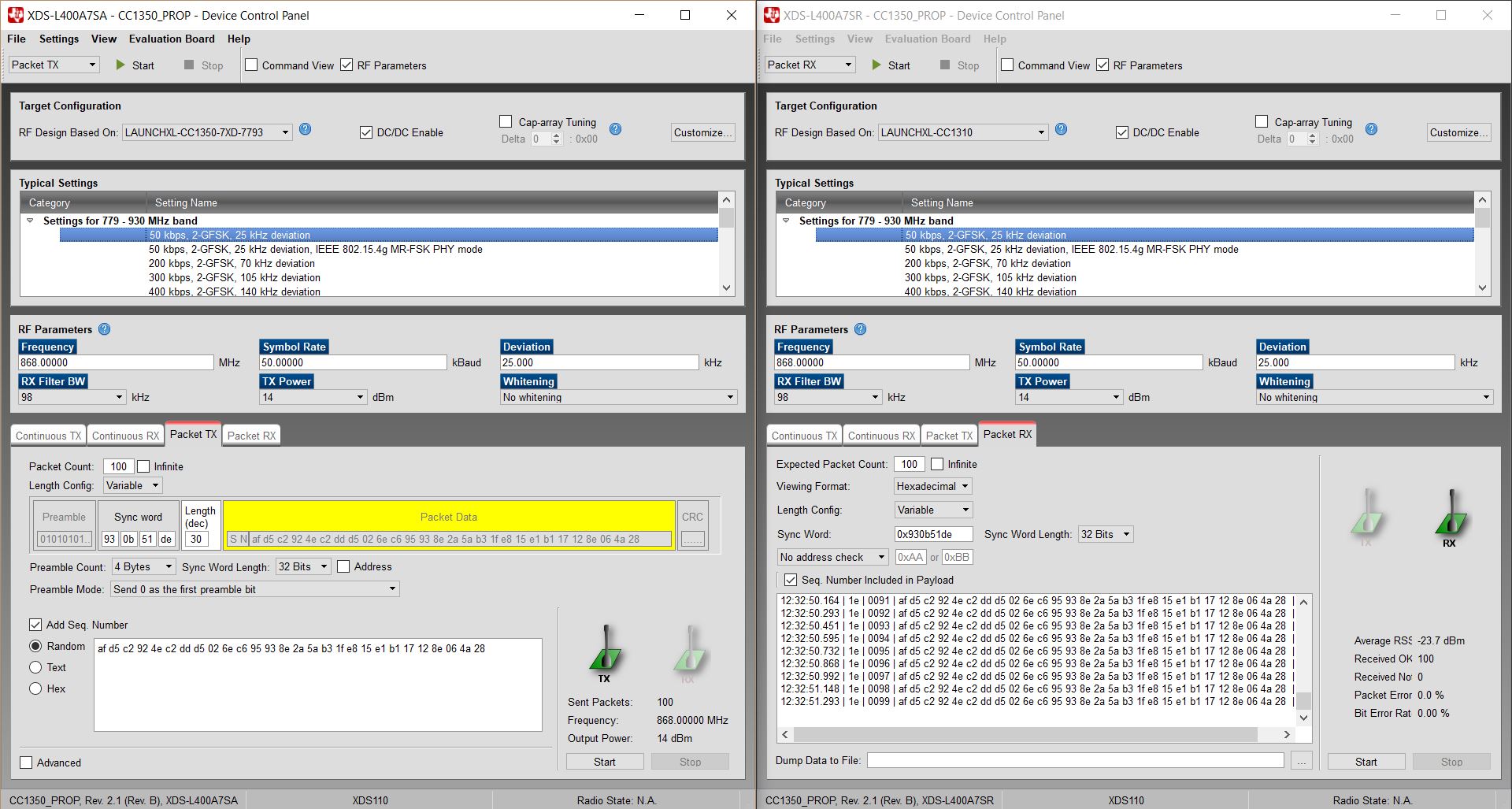
**Date Submitted: 11/19/18**

**Task 01: SmartRF Studio <-> SmartRF Studio**

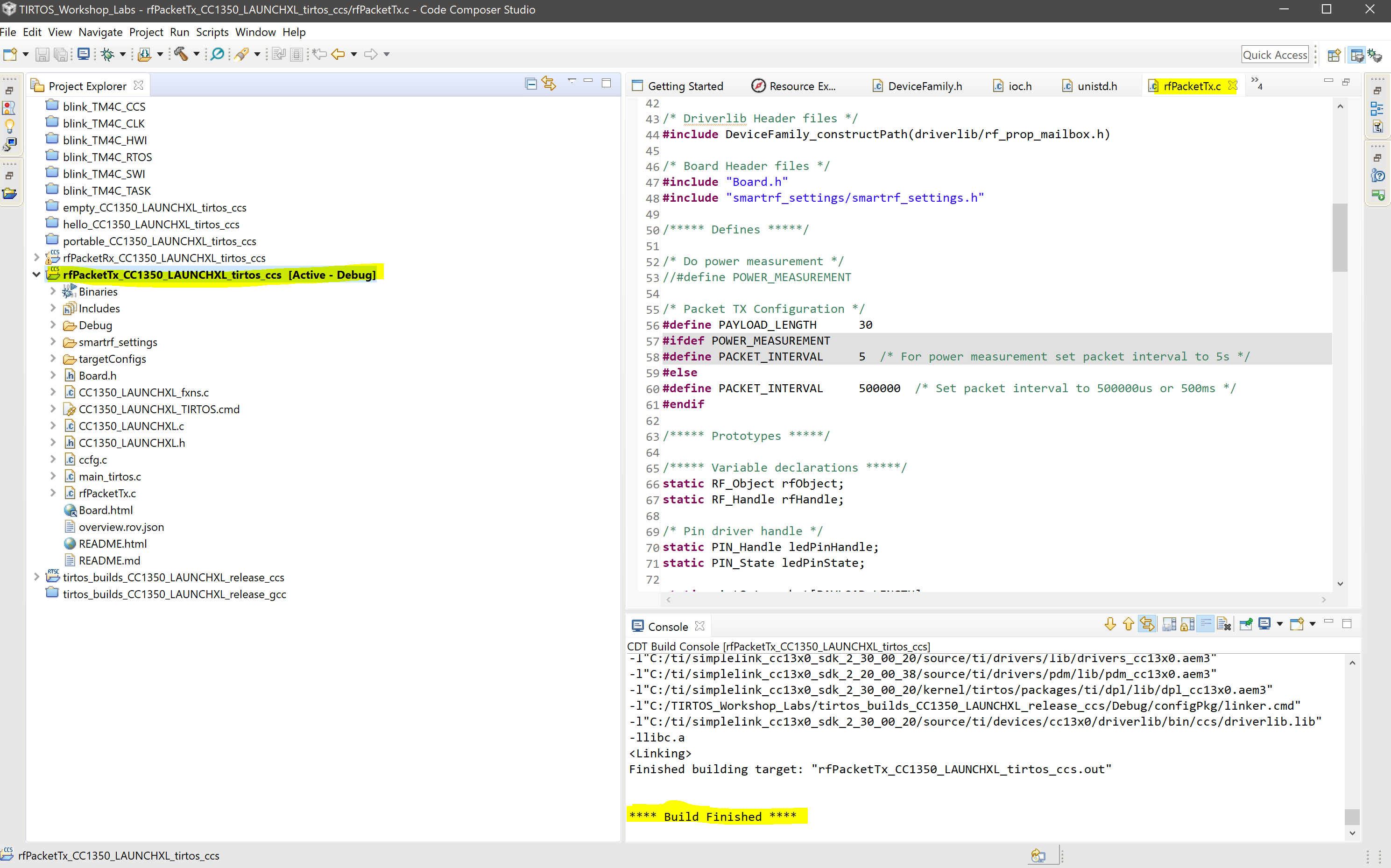
The purpose of Task 01 is to become familiar with the SmartRF Studio interface. First we would like to make sure our CC1350 Launchpads (RX & TX) connects appropriately with the SmartRF Studio (left). The next step is to configure RX and TX. On the right side of the photos, we can see that 100 packets have been sent through TX and have been received through RX.



**Task 02: Importing and running the rfPacketTx example**

Youtube Link: <https://www.youtube.com/watch?v=_S2o9WMwHss>

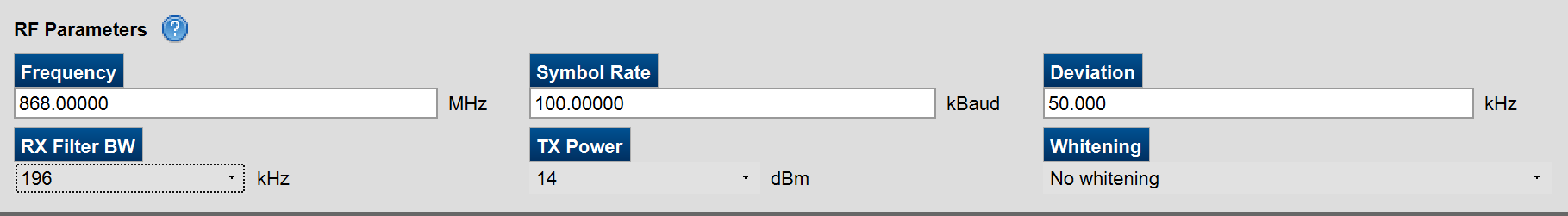
The purpose of Task 02 is to import the example given called rfPacketTx found in the Resource Explorer. This will provide a fundamental base for the rest of the Lab and to give a guide for programming a Launchpad to be a transmitter. Below is a screenshot of the rfPacketTx\_CC1350\_LAUNCHXL\_tirtos\_ccs project successfully completing a build. \*The video includes testing for rfPacketRx.

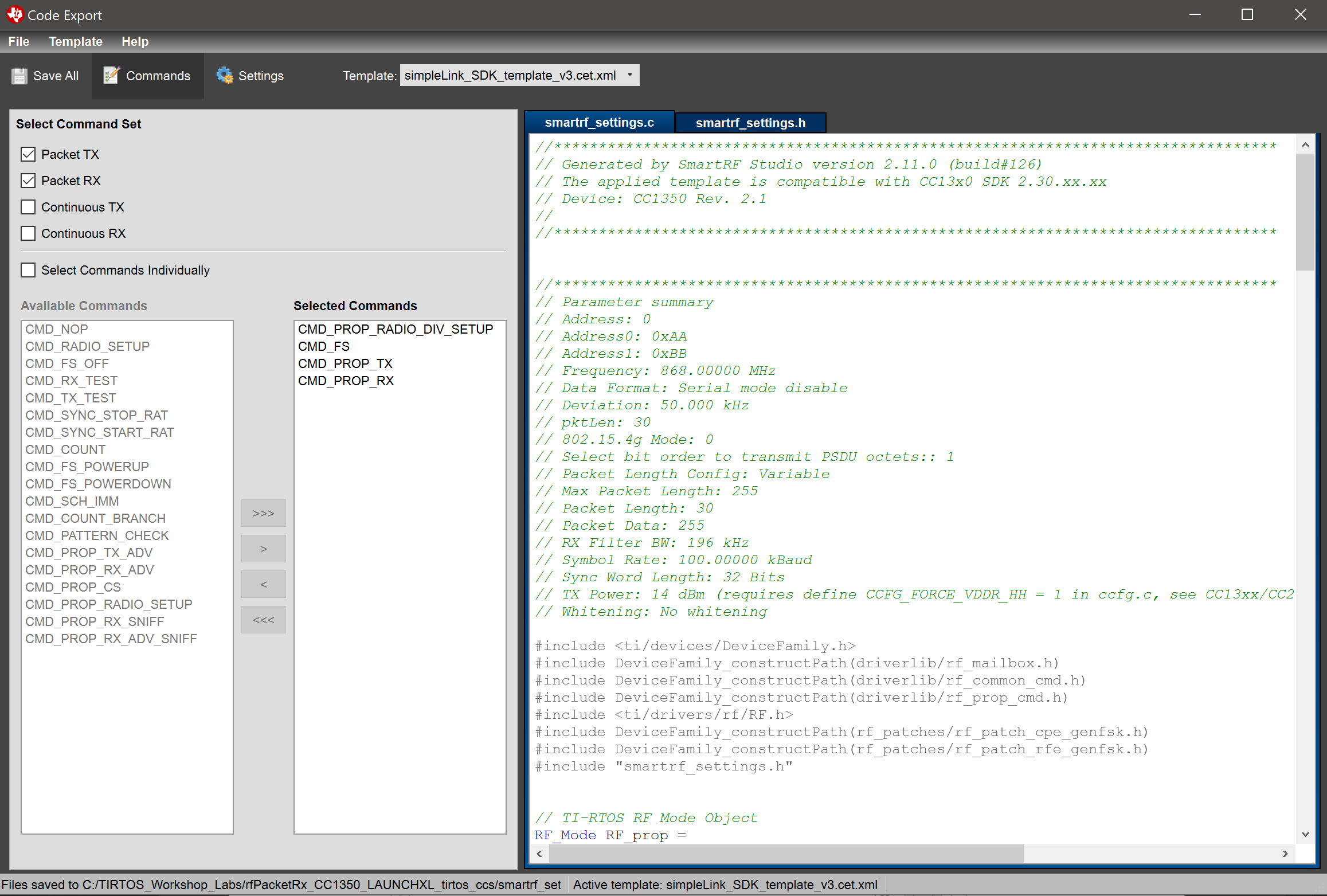


**Task 03: Exporting and using RF configuration**

Youtube Link: <https://www.youtube.com/watch?v=d-1cy_4OsRw>

This Task is to show how to export and configure the RF settings on the CC1350 Launchpad. This is performed using the SmartRF software provided by Texas Instruments. First, we had to set the RF Parameters, and then after we had to export the code. After exporting the code, we had to replace those files within our Code Composer Studio smartrf path within the specific projects. \*The video also includes the configuration set for RX.

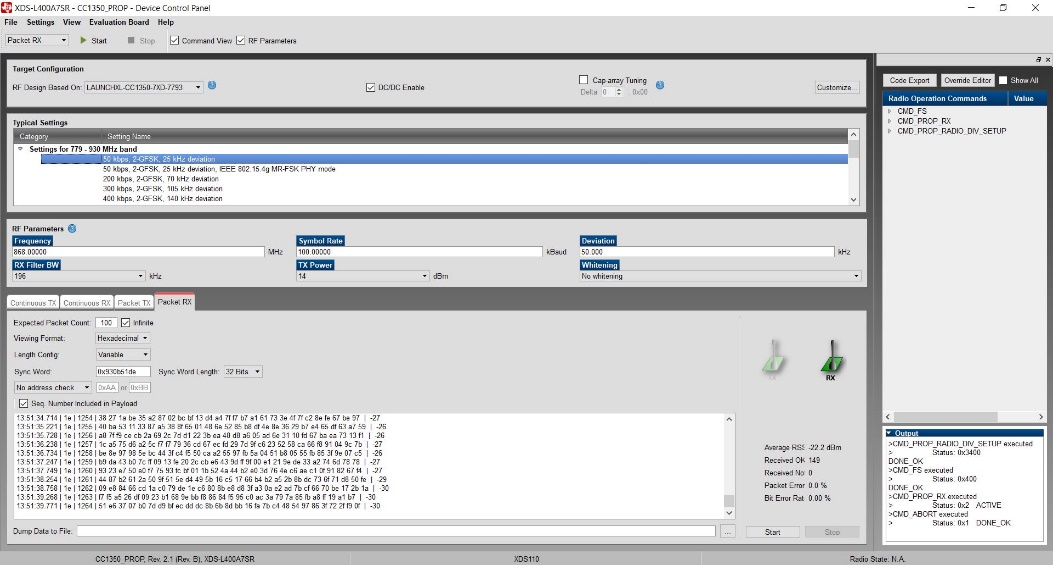




**Task 04: Firmware TX -> SmartRF Studio RX**

Youtube Link: <https://www.youtube.com/watch?v=A4CGBYKIjdk>

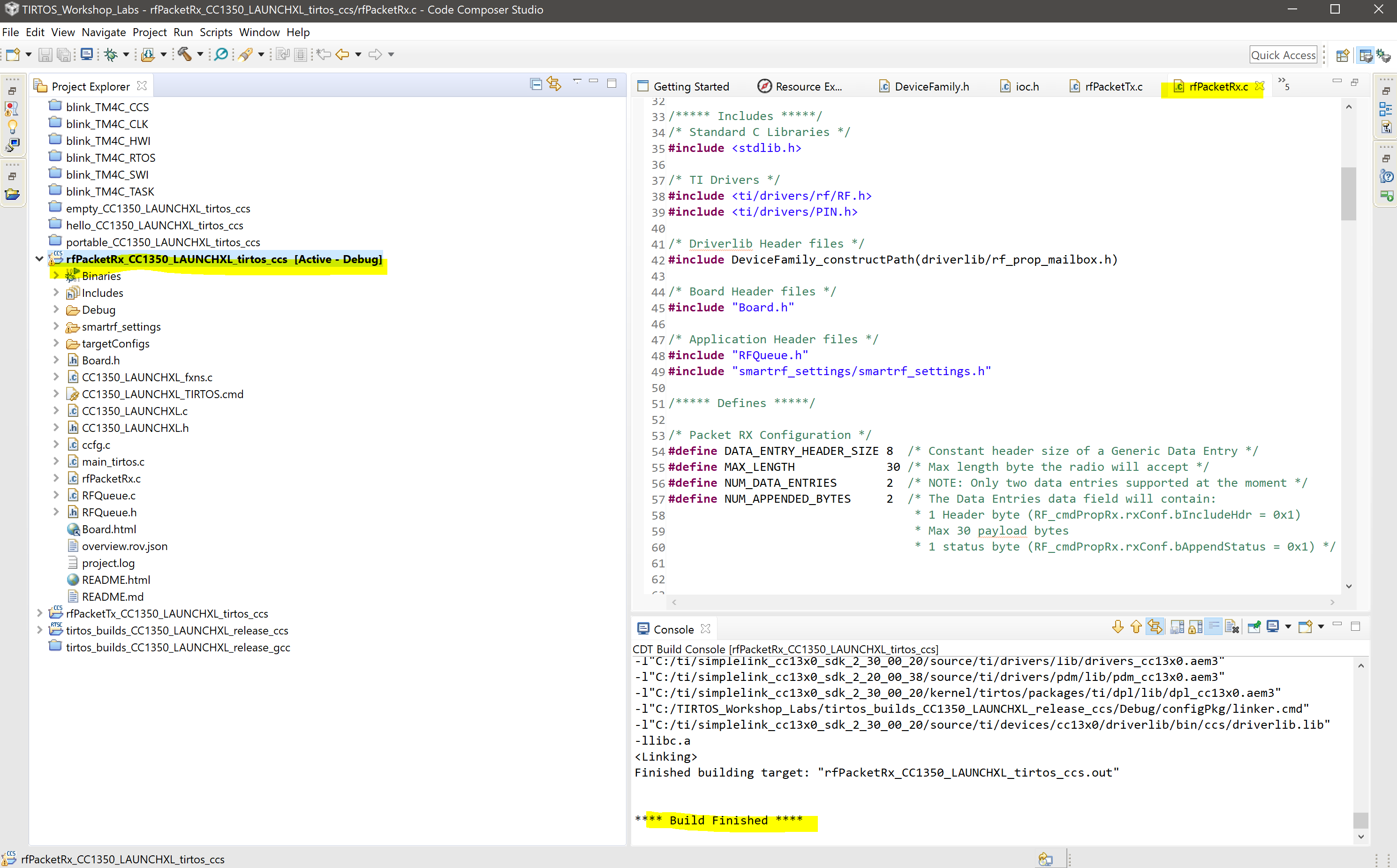
The purpose of this Task is to configure the RF settings on the RX Launchpad, similar to that of last task (Task 03). The other purpose is to receive data from the TX program in the Launchpad and to the RX as the smartRF studio.



**Task 05: Importing and Modifying rfPacketRx**

Youtube Link: <https://www.youtube.com/watch?v=_S2o9WMwHss>

The purpose of this task is to import the existing example from Resource Explorer or from the SDK directory called rfPacketRx. Here below we can see that in the Code Composer user interface the rfPacketRx build has finished successfully with no issues. \*The video shows both RX and TX running



**Task 06: Firmware TX -> Firmware RX**

Youtube Link: <https://www.youtube.com/watch?v=_S2o9WMwHss>

This task is to ensure that the firmware of TX and RX are working accordingly. In the video we can see the LEDs toggle interchangeably to ensure that information is being sent and received properly.

**Task 07: SmartRF Studio TX -> Firmware RX**

Youtube Link: <https://www.youtube.com/watch?v=GvhEH6fVuIk>

Lastly, this task requires to send information from the SmartRF Studio and to receive that information on the RX Launchpad.

