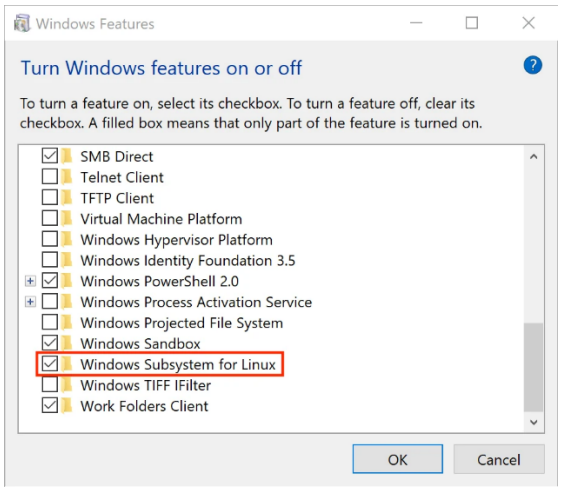
Setting up Sensor Tag (CC2650) for Windows

1. Setting up your environment

To enable WSL (Windows Subsystem for Linux), search and open "Turn Windows features on or off" in the start menu. Find "Windows Subsystem for Linux" in the list, enable it, and click “OK”.



Install WSL Ubuntu. You can find it in the Microsoft Store. Click “get” to install.

Alternatively, you can also use Windows PowerShell as an administrator and run the following command to install WSL Ubuntu.

```
wsl --install Ubuntu
```

Once the installation of windows subsystem for linux (WSL) is complete, you can launch it and the Bash Ubuntu shellwill appear. You will be prompted to create a username and password. It's important to note that the C drive will be displayed as /mnt/c and the D drive will be displayed as /mnt/d.

Please install Contiki-ng OS in your chosen directory:

```
git clone https://github.com/contiki-ng/contiki-ng.git --recursive
```

Next, we will install the ARM GCC compiler:

```
sudo apt-get install gcc-arm-none-eabi
```

2. Setting up UniFlash

You can find it here https://www.ti.com/tool/download/UNIFLASH

Under Downloads option, choose UniFlash Windows Installer and proceed with installation steps

3. Running your program

3.1 Compiling

Launch the WSL window. In the bash terminal, navigate to the folder /contiki-ng/examples/hello-world/. Enter the following command to compile the hello world program.

```
make TARGET=cc26x0-cc13x0 BOARD=sensortag/cc2650 hello-world
```

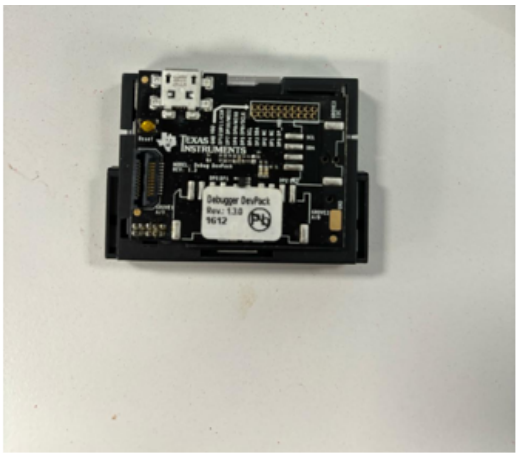
Then enter the following command,

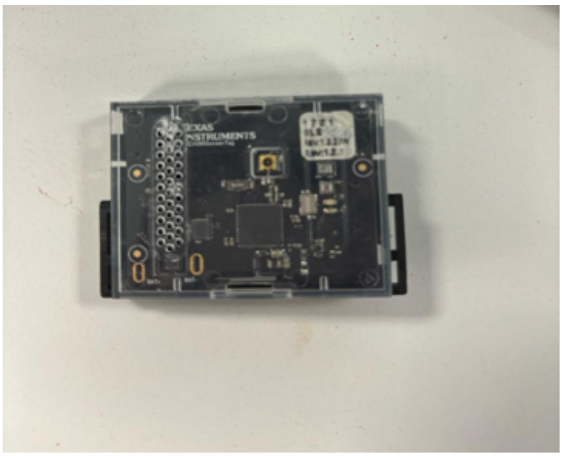
```
explorer.exe .
```

You will find the location of the compiled binary file ***hello-world.cc26x0-cc13x0.*** Copy this compiled file to an easily accessible location in your system.

3.2 Flashing the SensorTag with the binary file

Make sure that your Debugger is fixed to the sensor tag as shown below:

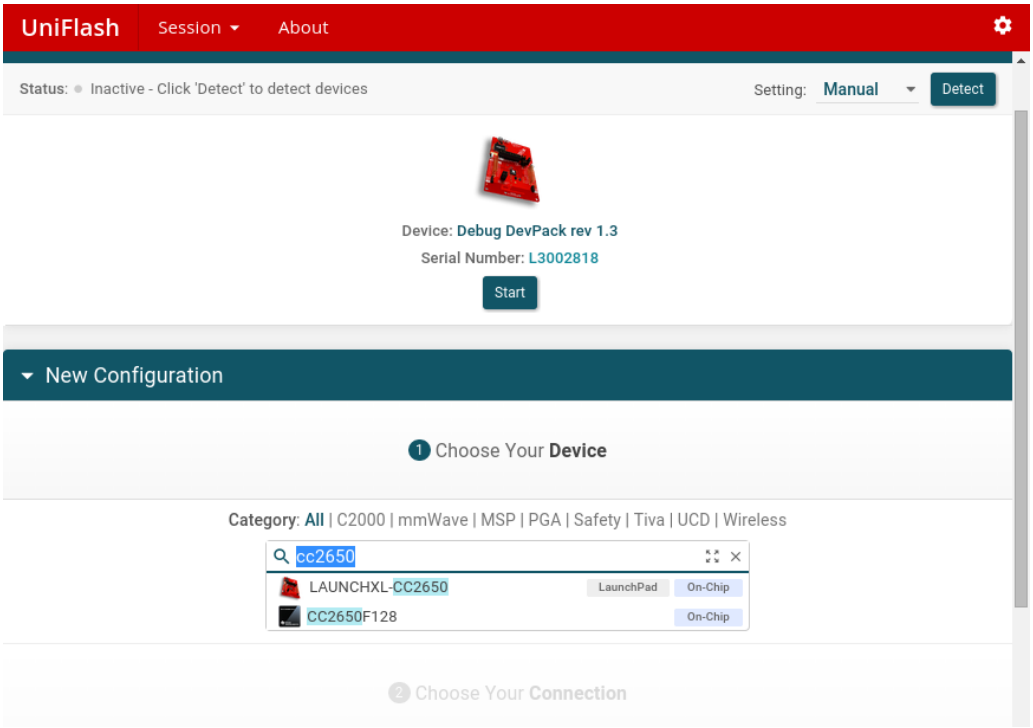




Connect your sensor tag to your computer. Change the baud rate of your sensor tag to 115200 using device manager.

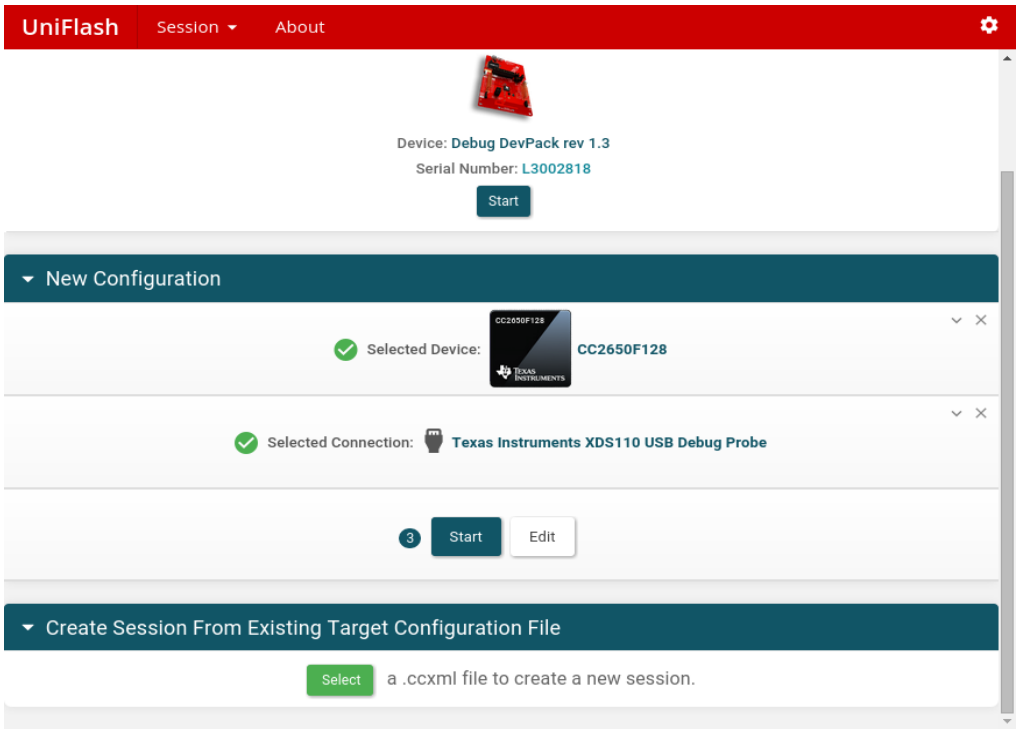
Session configuration

Open UniFlash. Choose your device by typing **CC2650** in the search field and select **CC2650F128**.

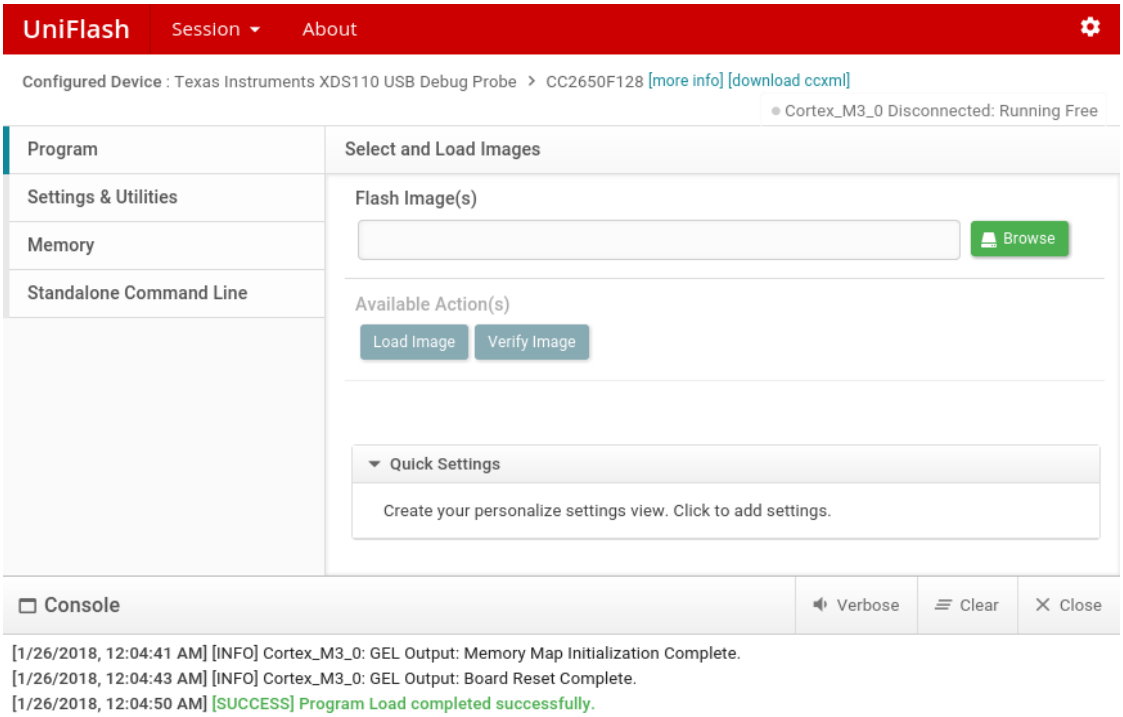


Important Note: Even if your device is not detected, you can directly proceed with choose your device step.

Next, choose the **Texas Instruments XDS110 USB Debug Probe** and hit the **Start** button as shown below.

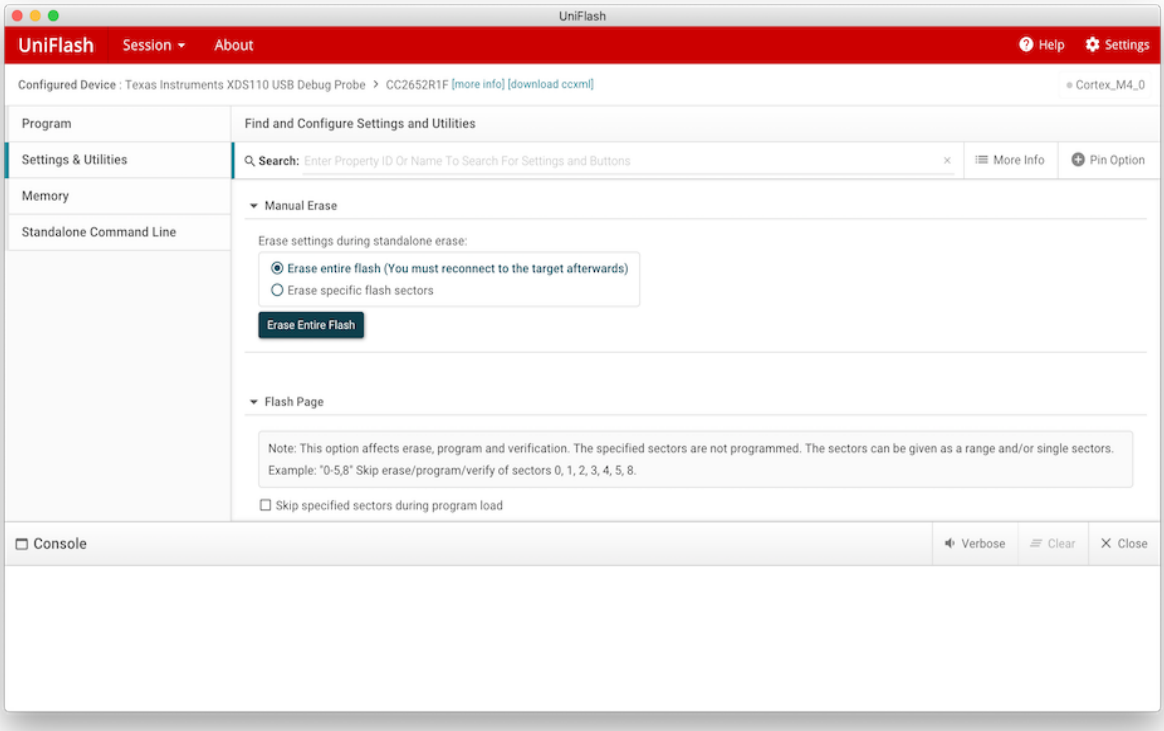


Finally, browse for the compiled file ***hello-world.cc26x0-cc13x0*** and click on load image.



While choosing the flash image, click the “Show Options” button and enable “All Files” type to be able to select the compiled file

If you encounter an **error** while flashing, go to Settings & Utilities and then perform ***Erase entire flash***. After this, go back to program tab and load your image.

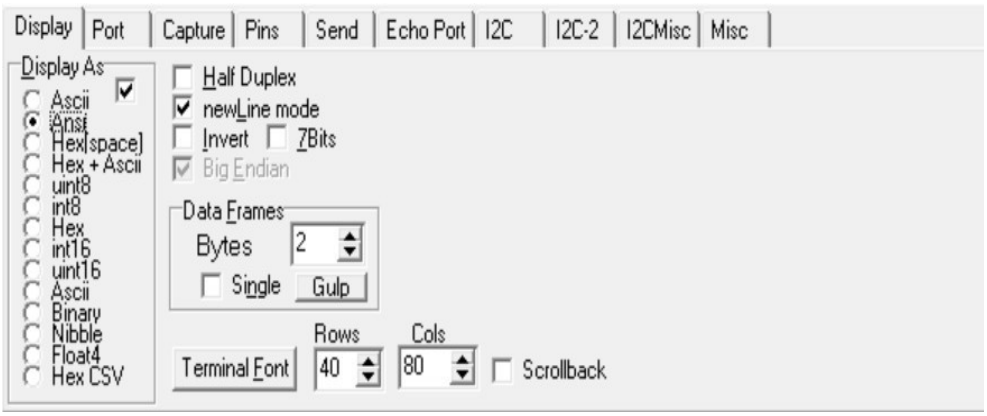


3.3 Showing the result

Unfortunately, Bash Ubuntu cannot detect your USB sensortag. To handle that, you can install realterm https://sourceforge.net/projects/realterm/

Once finished, open realterm and change the following settings.

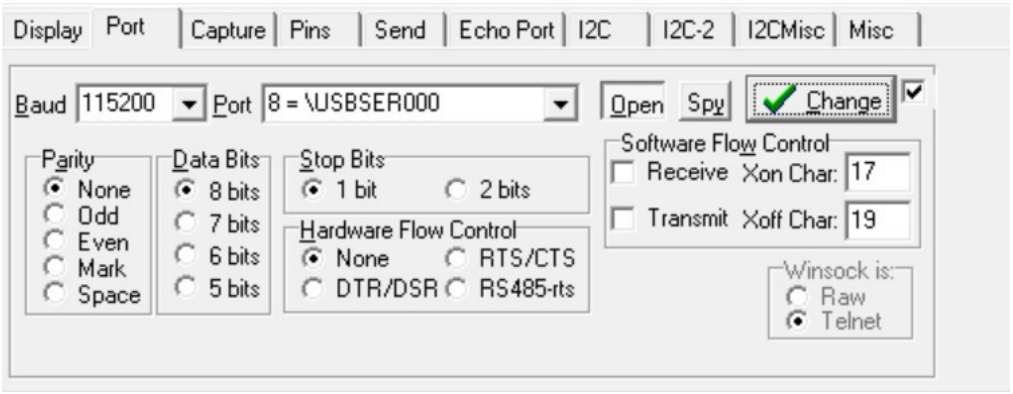
- In “Display” tab, choose Ansi instead of Ascii, check newLine mode, and change Rows to 40



- In “Port” tab, set baudrate to 115200 and Port number same as the one shown in your Device Manager. Choose the UART one, as two ports will be dispalyed. In my example, below it is COM8

  Ports (COM & LPT)
  - XDS110 Class Application/User UART (COM8)
  - XDS110 Class Auxiliary Data Port (COM7)

  Then press “Change” button. Lastly, press “Open”



Press the reset button (highlighted using red circle) on your sensor tag to read the following output: