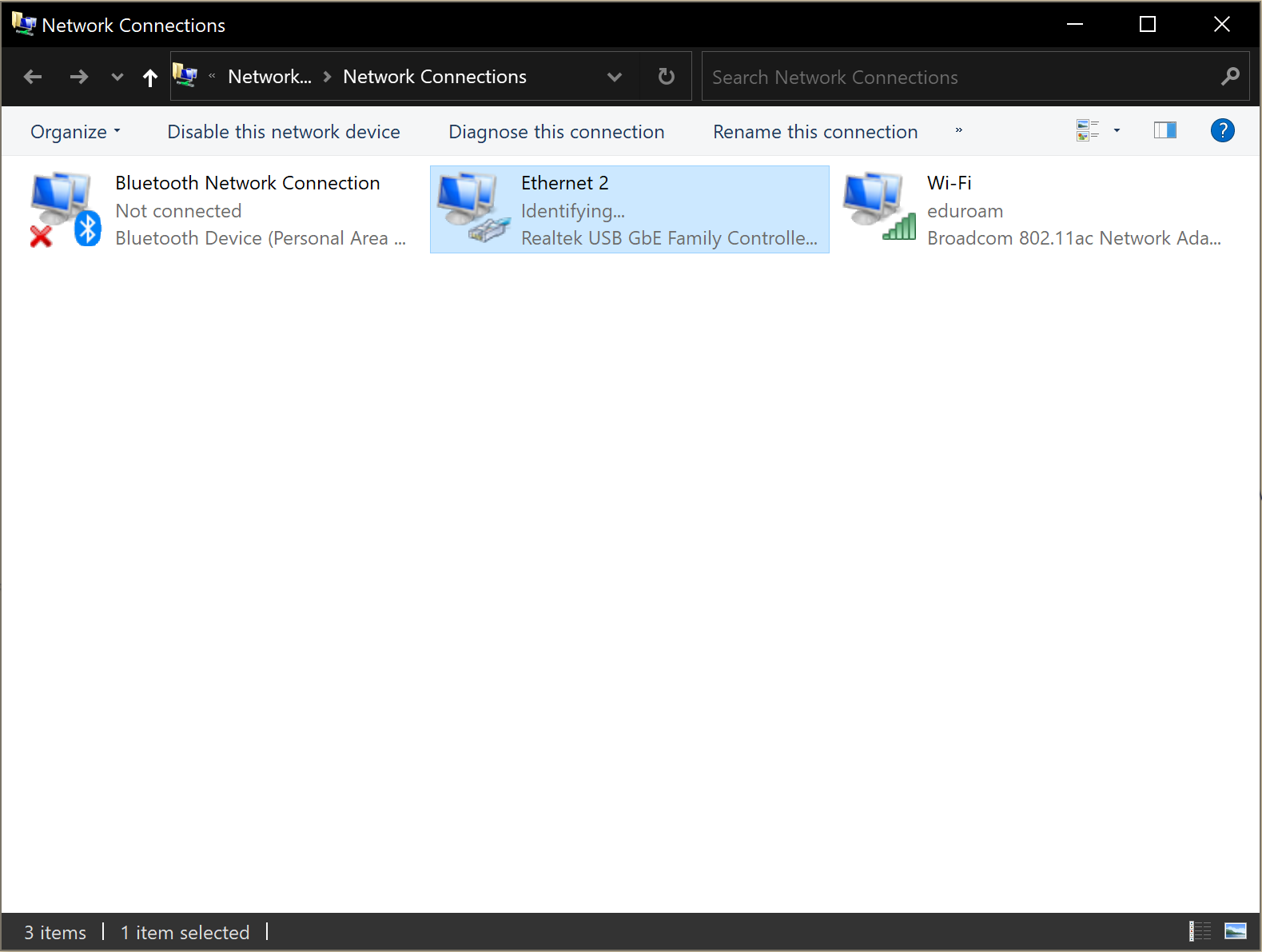
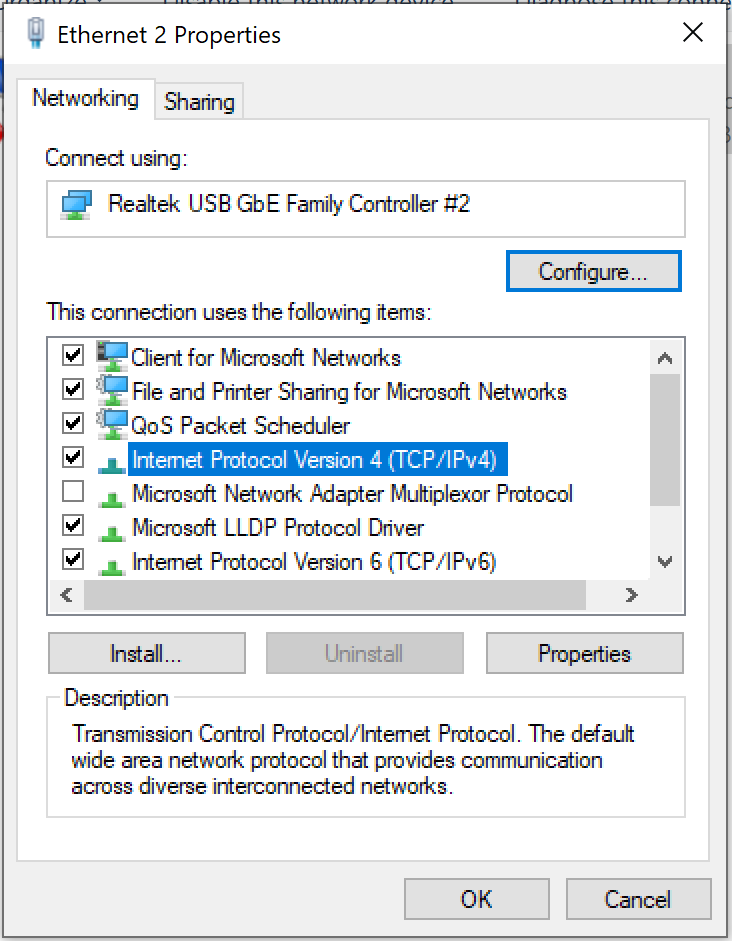
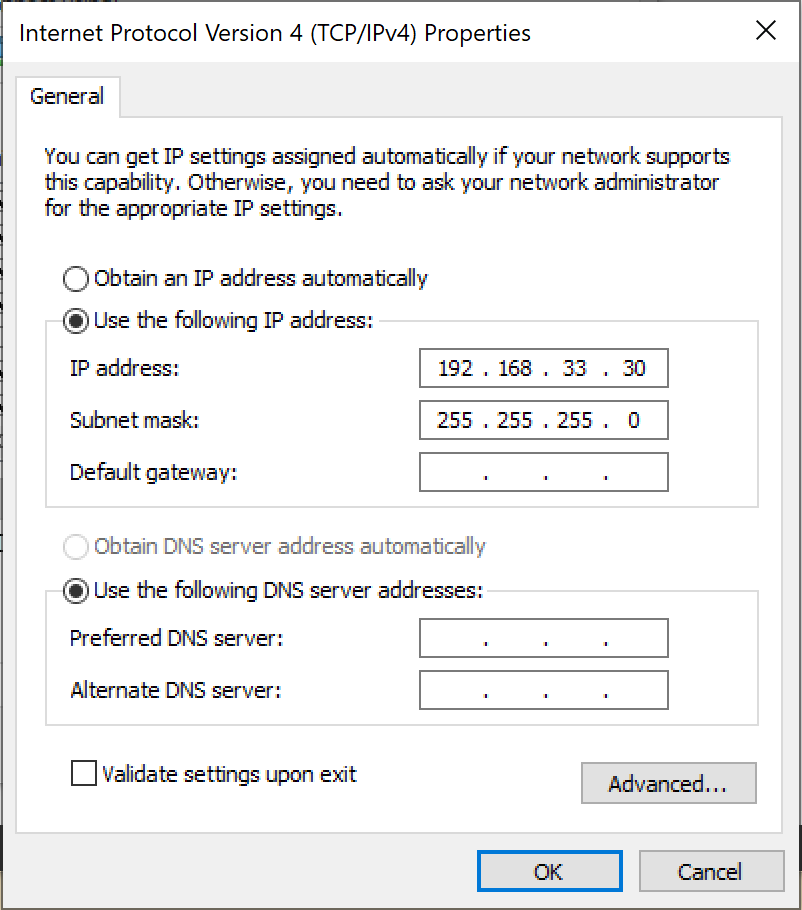
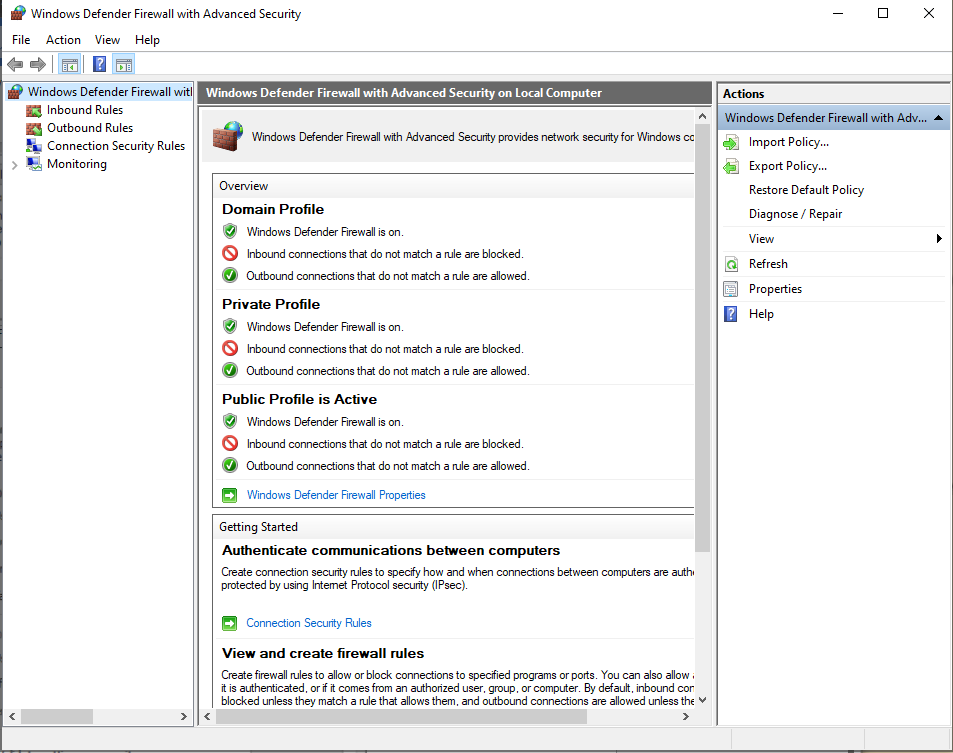
Installing and using mmWave with a TI AWR1642Boost radar sensor and a DCA1000EVM data capture board.

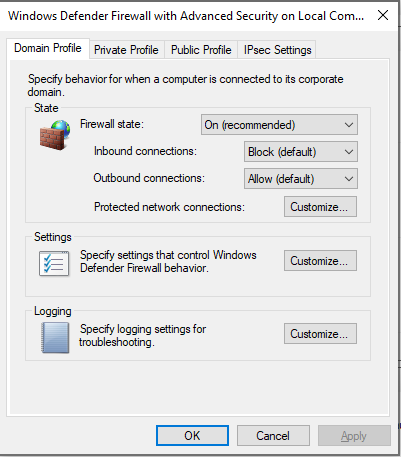
# Firewall Configuration

1. Connect the radar sensor and the data capture board to the designated computer (has to be a Windows computer, cannot be a Mac or Linux unless it is running Bootcamp or a Windows VM) using the three micro USB connectors and the Ethernet port.
2. Power the board up.
3. In the windows search menu, search for Network Connections.
4. Find the Ethernet adapter that you’re using and select Properties.
5. Make sure that the protocol version is selected as IPv4, select properties.
6. Use the following IP address: 192.168.33.30. The mask is set automatically.

Close this window.

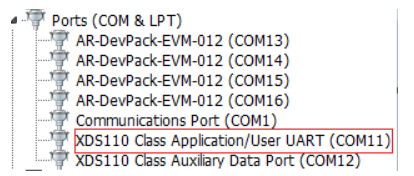
1. In the windows search menu, search for Windows Defender Firewall.
2. On the left side of the panel, go to Advanced Settings and Windows Defender Properties.



1. For Domain Profile, Private Profile and Public Profile, go to State:Customize and uncheck the Ethernet port displayed there. 
2. This is a crucial and necessary step for the Data Capture board to be able to communicate with mmWave Studio.
3. Unplug the power from the radar device.

# Installing mmWave Studio

1. Make sure the latest version of Matlab is installed on the computer.
2. Install mmWave Studio from the installer package. **DO NOT OPEN MMWAVE STUDIO AFTER IT FINISHES INSTALLING.**
3. Install Matlab Runtime Engine that is included in the folder. For reference, it **needs** to be version R2015aSP1, or 8.5.1. **No other version will work.**
4. Power the device and verify in Device Manager, under Ports and COM that there are 4 USB serial port objects present as well as XDS110 Class Application/User UART and XDS110 Class Auxiliary Data Port present. It can happen that the XDS110 objects do not load properly so you must install them externally, Use the FTDI driver package included in the folder (*ti\_emupack\_setup\_9.2.0.00002\_win32*) and follow the instructions. When that completes, verify that these are present now. At the end it should look something like the image below.



Note that the COM numbers will differ between systems, but do note the COM port of the red-boxed object. You will need that later.

1. Unplug power from the device.
2. Install the Microsoft Virtual C++ 2013 Redistributable if your computer is running Windows 10.
3. Now you can power the device again, making sure that all USBs and the Ethernet cable are connected to the device and the computer. **Start the latest version of Matlab.**
4. Using the mmWave desktop shortcut is highly unreliable, so I recommend searching for mmWave in the Windows search bar and then select Run as administrator. For me this is the most reliable way to start up this software.
5. It takes a little while to start up. If it shows Not Responding, just be patient. It’s just the Matlab runtime engine starting up while being dramatic.
6. After it loads, I recommend portioning a part of the screen for the output log. Go to View -> Output, which will open in a new tab. Just grab it and put it wherever visible, which will give you better feedback for whatever you will be doing.
7. The first page that will open up otherwise is the Connection page. In Board Control -> RS232 Operations, select the COM port that you noted in step 4. It should be available if everything has been installed correctly.
8. At this point you may run the data capture demo that comes with the software to see if it works, but it will load configuration files you will have to replace if you want to use your own profiles.

# Customizing profiles for mmWave Studio.

MmWave by it’s nature is highly unpredictable and will crash when you least expect it. The method I’ve used to load and customize chirp profiles for the system are as follows:

1. Start Matlab.
2. Start mmWave Studio in administrator mode.

Note: If the application freezes with Not Responding, it means its working. Just wait a little bit.

### 1 - Connect

Board Control:

* Reset Control : Set (1)

RS232 Operations : Connect (2)

* Make sure the COM port is directed to the XDS110 Class Application/User UART port.

Files:

* BSS FW : Redirect to C:\ti\mmwave\_studio\_02\_01\_01\_00\rf\_eval\_firmware\radarss\xwr16xx\_radarss.bin

Press Load (3)

* MSS FW : Redirect to C:\ti\mmwave\_studio\_02\_01\_01\_00\rf\_eval\_firmware\masterss\xwr16xx\_masterss.binPress Load (4)

Note: These paths will be saved for the next time the application is launched.

SPI Operations:

* Press SPI Connect (5)
* Press RF Power-up (6)

### 2 - Static Configuration

Basic Configuration:

* Channel Config:
* Uncheck Tx channel Tx1
* Uncheck Rx channels Rx1, Rx2, Rx3

ADC Config

* Set Format to Complex2x
* Press Set

Advanced Configuration:

* LP ADC Mode
* Press Set

Frequency Limits Configuration:

* Press Set

Press RF Init

### 3 - Data Configuration

Data Path Configuration:

* Press Set

Clock Configuration

* Press Set

LVDS Lane Configuration

* Press Set

### 4 - Sensor Configuration:

Profile:

* Input profile of choice.
* Press Set

Chirp:

* Press Set

Frame:

* Input profile of choice.
* Press Set

### Capture and Post Processing.

On the far left of the application screen, select Set Up DCA1000 to open up a pop-up window.

* Press Connect, Reset and Configure.
* Do not close the pop-up, just minimize.

Press DCA1000 ARM

* Wait at least 3 seconds.
* Press Trigger Frame

Once the output displays Record is done sucessfully, press **PostProc**.