

TI mmWave Labs

Vital Signs Measurement

Contents

- Overview
- Requirements
- Software setup
 - Pre-requisites
 - Downloading the Lab Project
 - Building the project
- Hardware setup
 - Preparing the EVM
 - Connecting the EVM
- Running the Lab
 - Loading Program
 - Running the GUI

Lab Overview

- This lab exercise demonstrates the ability of TI IWR 14xx mmWave sensor to measure chest displacements due to breathing and heart beat
- Typical vital signs parameters for adults

Vital Signs	Amplitude	Frequency
Breathing Rate (Adults)	1- 12 mm	0.1 – 0.5 Hz
Heart Rate (Adults)	0.1 – 0.5 mm	0.8 – 2 Hz

- To measure these small scale vibrations/displacements, we measure the change in phase of the FMCW signal with time at the target range bin

$$\Delta\phi_b = \frac{4\pi}{\lambda} \Delta R$$

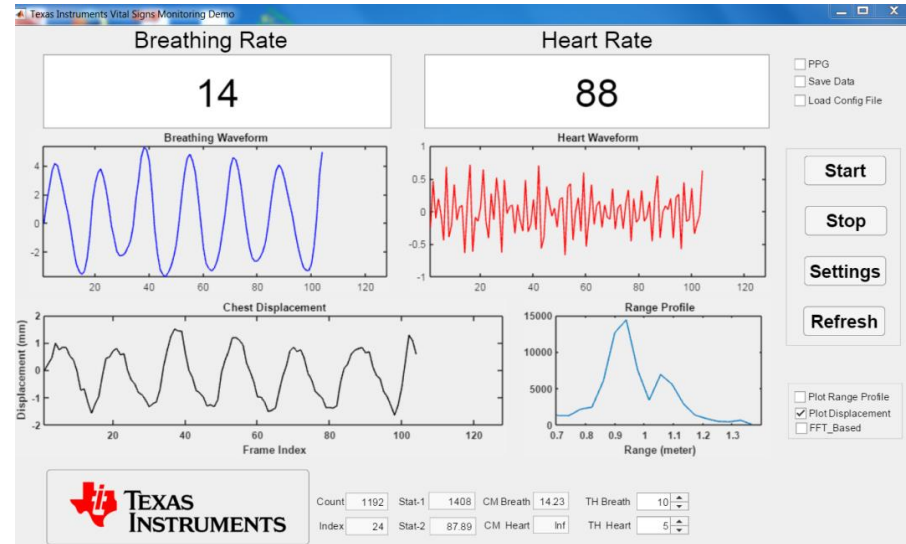
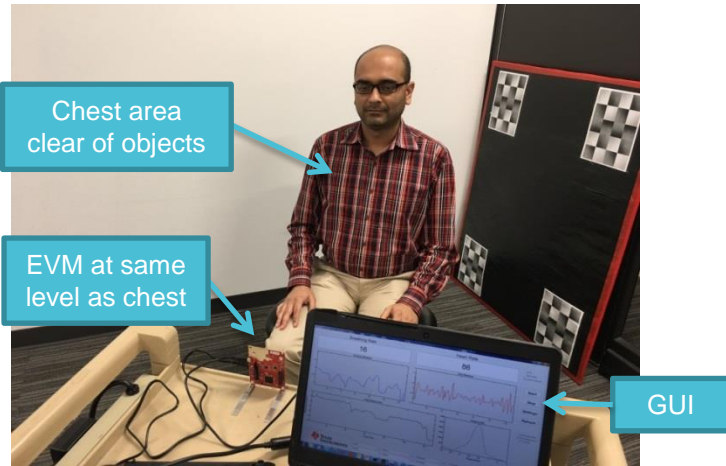
- $\Delta\phi_b$ corresponds to the change in phase when the target moves a distance ΔR

- Note that a smaller wavelength λ will give better displacement sensitivity

- Code Composer Studio (CCS) project along with source code is provided for this lab

Lab Overview

- TI mmWave sensor measures the chest displacement when it is pointed towards the chest of a person sitting in front of the sensor
- The onboard programmable processing cores on the TI IWR 14xx mmWave sensor are used to filter out the breathing and heart beat pattern from chest displacements and estimate the Breathing and Heart-rate
- GUI displays the chest displacements, filtered waveforms and the estimated Breathing and Heart-rate



Note: Although not an absolute requirement, a lens in front of the sensor will improve the performance

1. Requirements

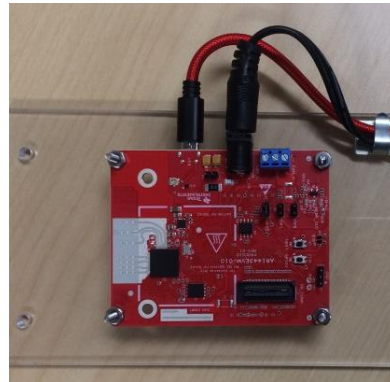
- Software

- **Pre-requisites**

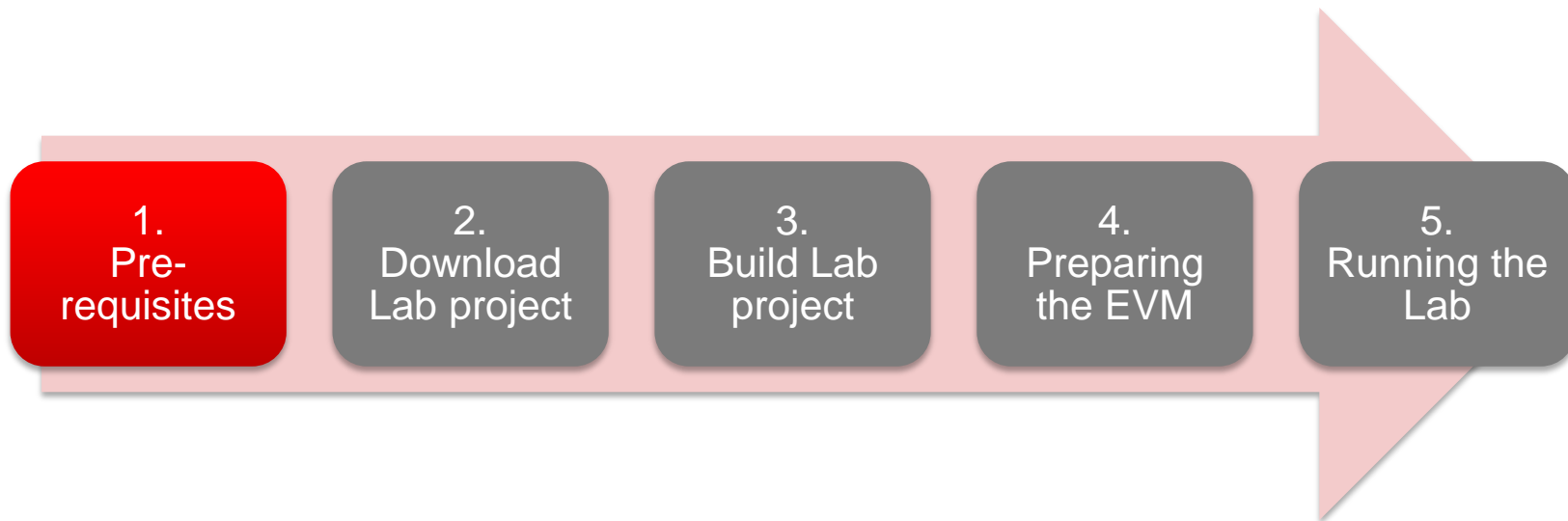
- [Latest TI mmWave SDK](#) and all related dependencies installed as mentioned in the mmWave SDK release notes.
 - Vital Signs Lab CCS Project
 - Download from [TI Resource Explorer](#)
 - UniFlash
 - For flashing firmware images onto
 - Download from [TI.com/tool/uniflash](https://www.ti.com/tool/uniflash)
 - XDS110 Drivers
 - For EVM XDS device support
 - Included with CCS Installation, or standalone through [TI XDS Emulation Software](#)
 - MATLAB runtime R2016b (9.1)
 - For running the Vital Signs Lab GUI
 - Download from [MATLAB website](#)

- Hardware

- AWR14xx/IWR14xx EVM
 - Micro USB cable (included in the EVM package)
 - 5V/5A Power Supply
 - [Purchase from Digikey](#)
 - A lens/concentrator to direct the radar waves towards the chest



Steps



1. Pre-requisites

1. Install Pre-requisites

2

3

4

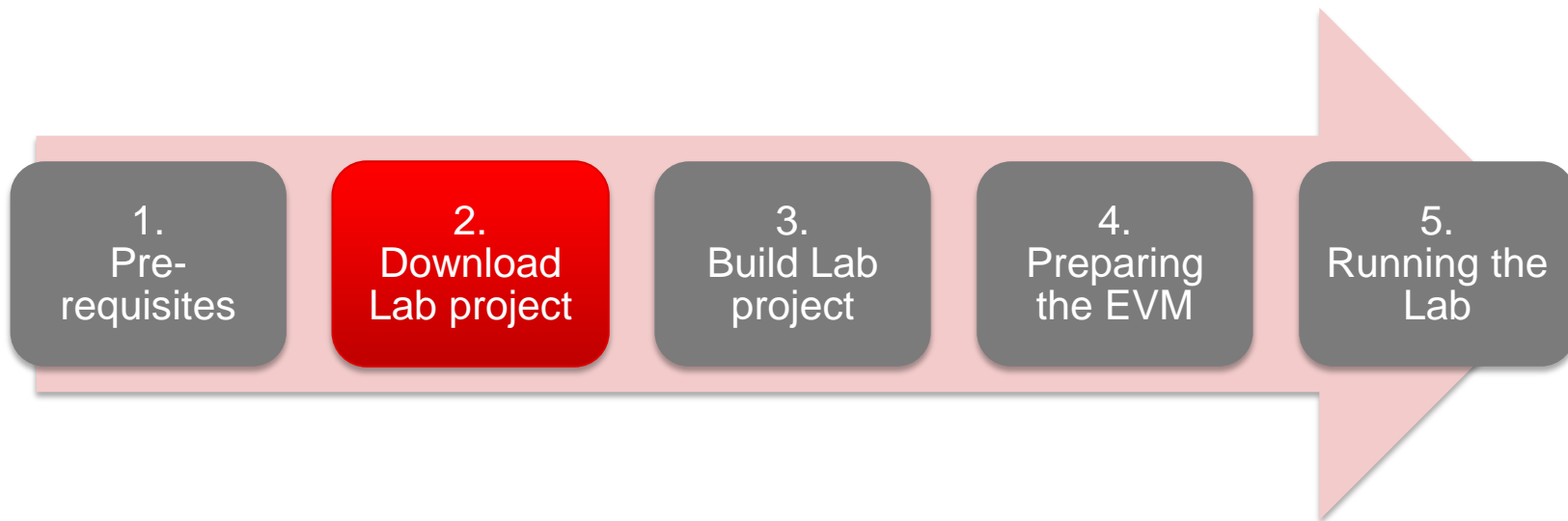
5

- It is assumed that you have the latest TI mmWave SDK and all the related tools dependencies installed as mentioned in the mmWave SDK release notes.
 - The tools include Code Composer Studio, TI SYS/BIOS, XDC tools, TI ARM compiler, and Perl among other dependencies mentioned in the mmWave SDK release notes.
 - The mmWave SDK release notes include the links for downloading the required versions of the above tools.
 - Note: For this video, we used mmWave SDK 1.0.0.5 and followed the corresponding release notes. Please follow the release notes for the latest SDK version. **The snapshot shown here corresponds to mmWave SDK 1.0.0.5**
- Please follow the mmWave SDK release notes to install the latest mmWave SDK and all the required tools if you don't have these setup already.
- If you have already installed the mmWave SDK and all the required tools, you can move on to the next step i.e. downloading the lab on to your machine.

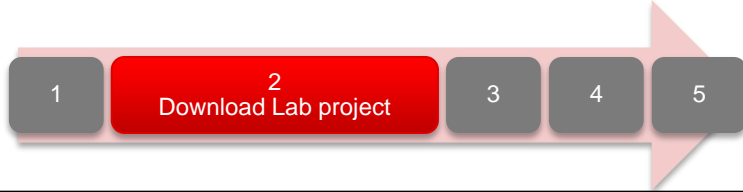
Tool	Version	Download link
CCS	7.1 or later	download link Please note that CCS v7.1 or later is mandatory. CCSv6.x cannot be used
TI SYS/BIOS	6.50.1.12 or later	download link
TI ARM compiler	16.9.1.LTS or later	Included in CCS v7.1
TI CGT compiler	8.1.3 or later	Included in CCS v7.1
XDC	3.50.00.10 or later	download link Use the installer that has no mention of JRE which by default has JRE (this installer will be bigger than the one without JRE). Also it is recommended that this be installed separately even if CCS has this version installed since the CCS version is without JRE.
C64x+ DSPLIB	3.4.0.0	Please download the installer for C64x+. download link
C674x MATHLIB (little-endian, elf/coff format)	3.1.2.1	Please download the installer for C674x. download link
Perl	5.20.2 or later	download link
CRC.pm	0.21	download link [This is an add-on to the Perl installation. This file needs to be copied to <PERL installation folder>\perl\lib\Digest folder.
Mono JIT compiler	3.2.8	Only for Linux builds
mmwave device support packages	1.5.3 or later	Upgrade to the latest using CCS update process
TI Emulators package	6.0.0576.0 or later	Upgrade to the latest using CCS update process

7

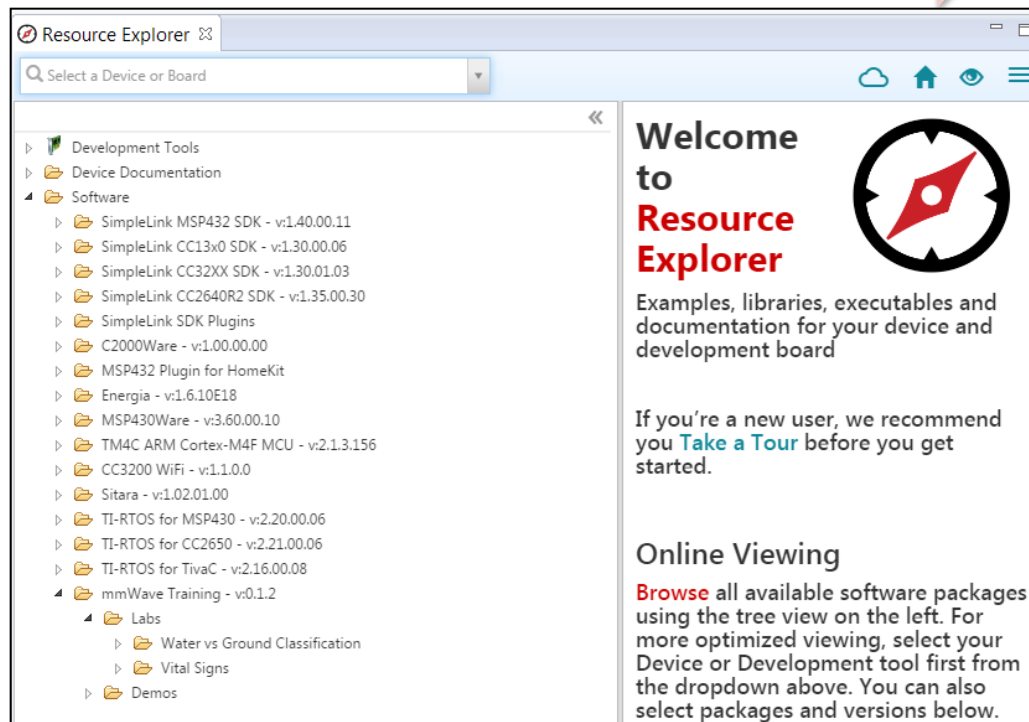
Steps



2. Download the Lab project




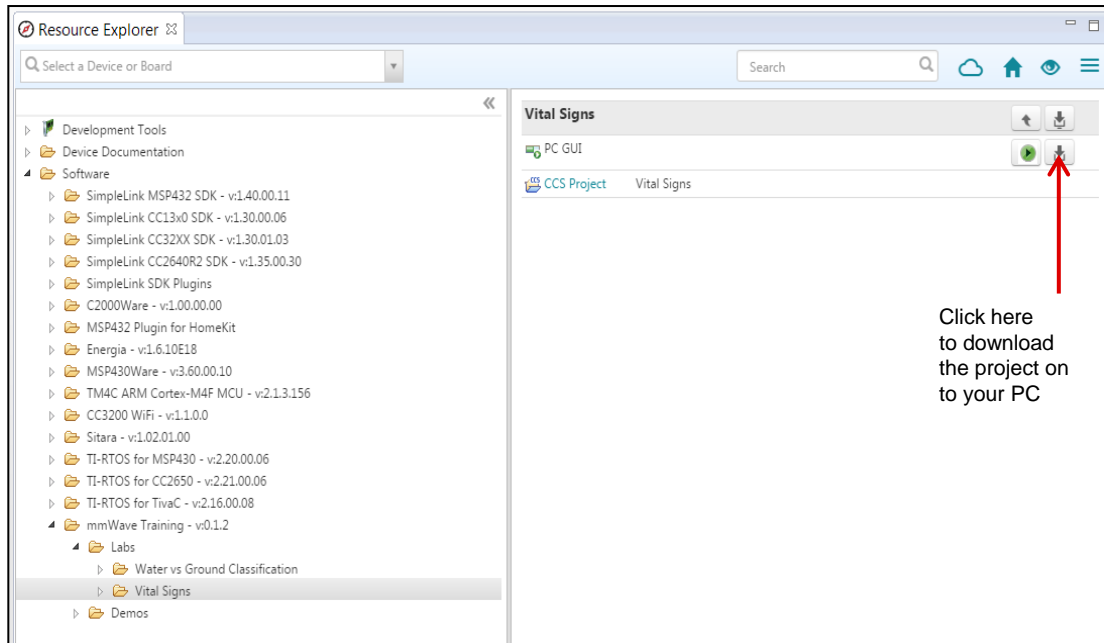
- The mmWave Lab projects are available under **mmWave Training** in CCS Resource Explorer.
- To download the Vital Signs Lab, start CCS v7.1 (or later) and select **View ► Resource Explorer** to open the Resource Explorer.
- In the Resource Explorer Window, select **Software ► mmWave Training ► Labs**.



2. Download - continued




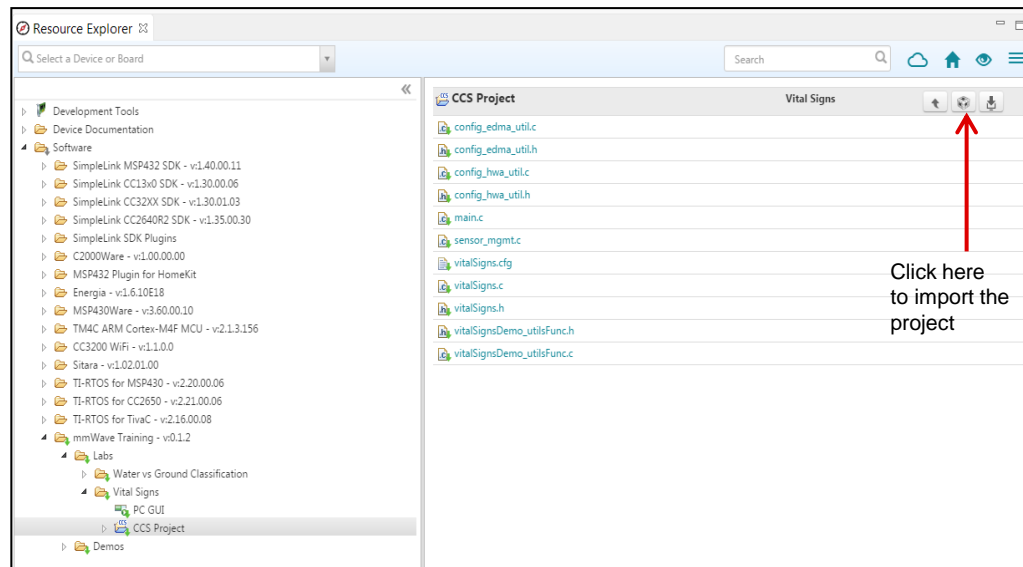
- Select the Vital Signs Lab in the left view.
- The right view shows the contents of the Lab which contains the CCS Project and the PC GUI.
- Click on the **Download and Install** button  in the top right corner as shown.
- Select the **Make Available Offline** option from the drop down to start downloading the Lab.



2. Download - continued



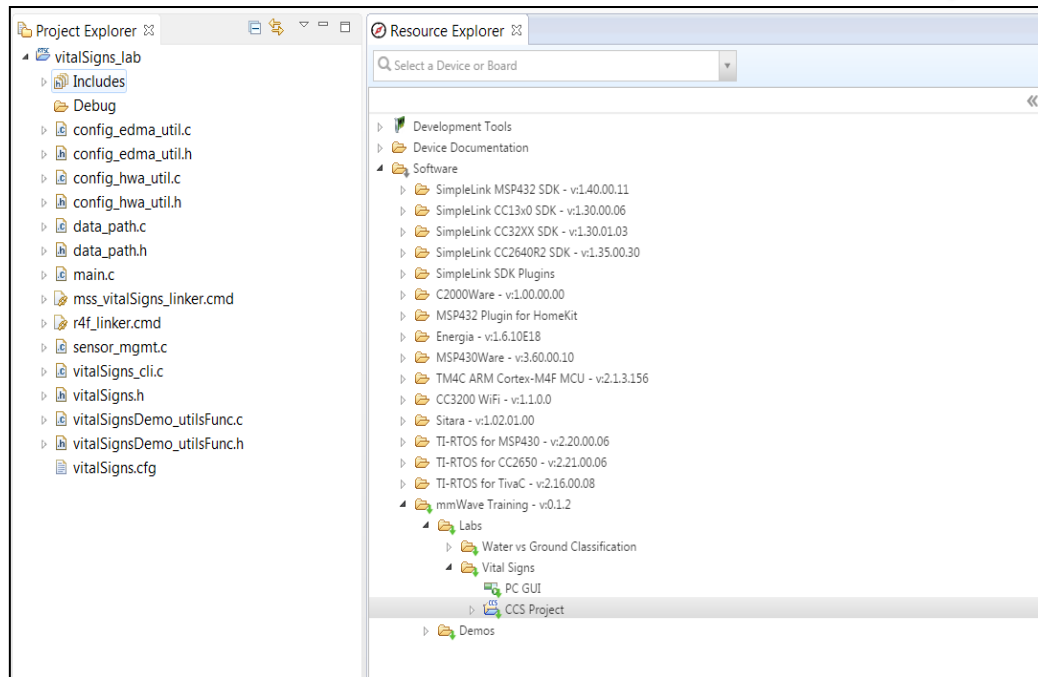
- The project will be downloaded in C:\ti\mmwave_training
- Click on the **Import to IDE**  button which should be visible in the right side view after a successful download.
- This copies the project in the user's workspace and imports it into the CCS project explorer.
 - It is important to note that the copy created in the workspace is the one that gets imported in CCS. The original project downloaded in mmwave_training is not touched.



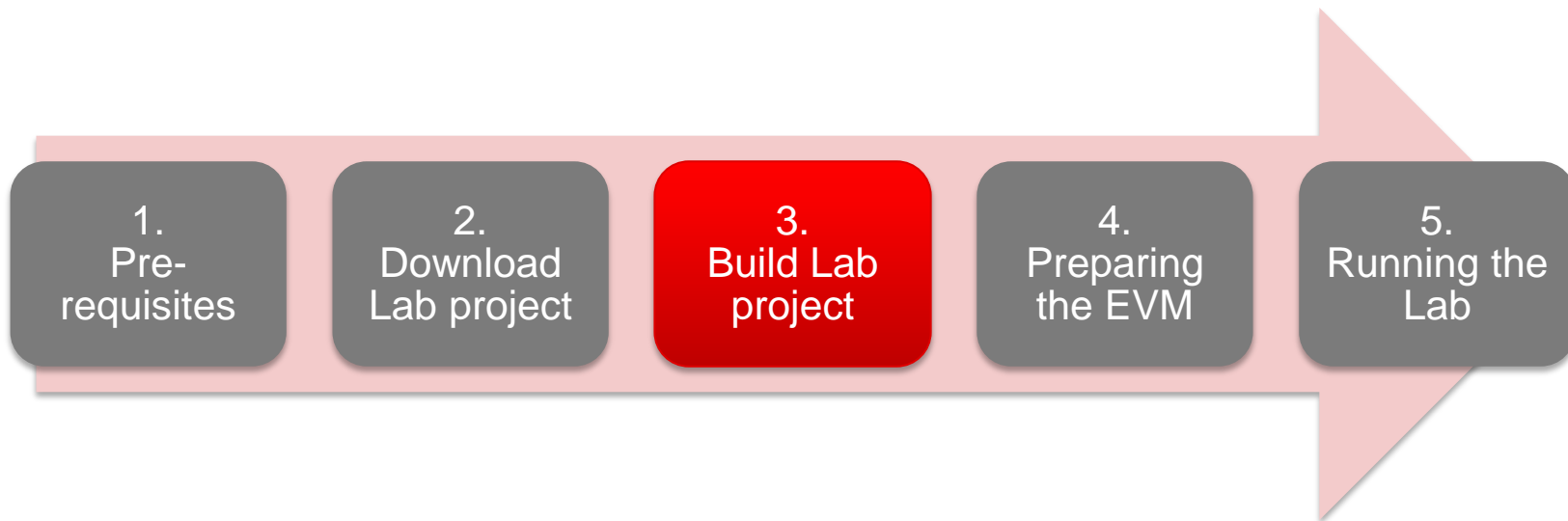
2. Download - continued



- After successfully completing the **Import to IDE** operation, the project should be visible in CCS Project Explorer as shown here.
- At this point, we have successfully downloaded the Vital Signs Lab and imported it in CCS.
- We are ready to move on to the next step i.e. Building the project.



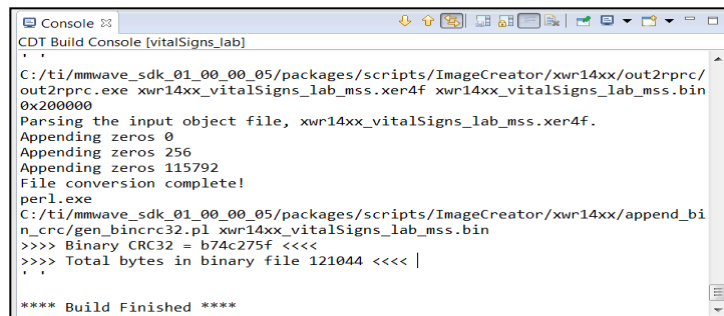
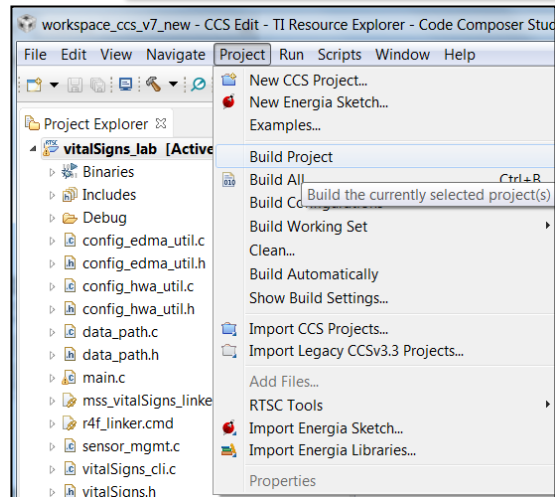
Steps



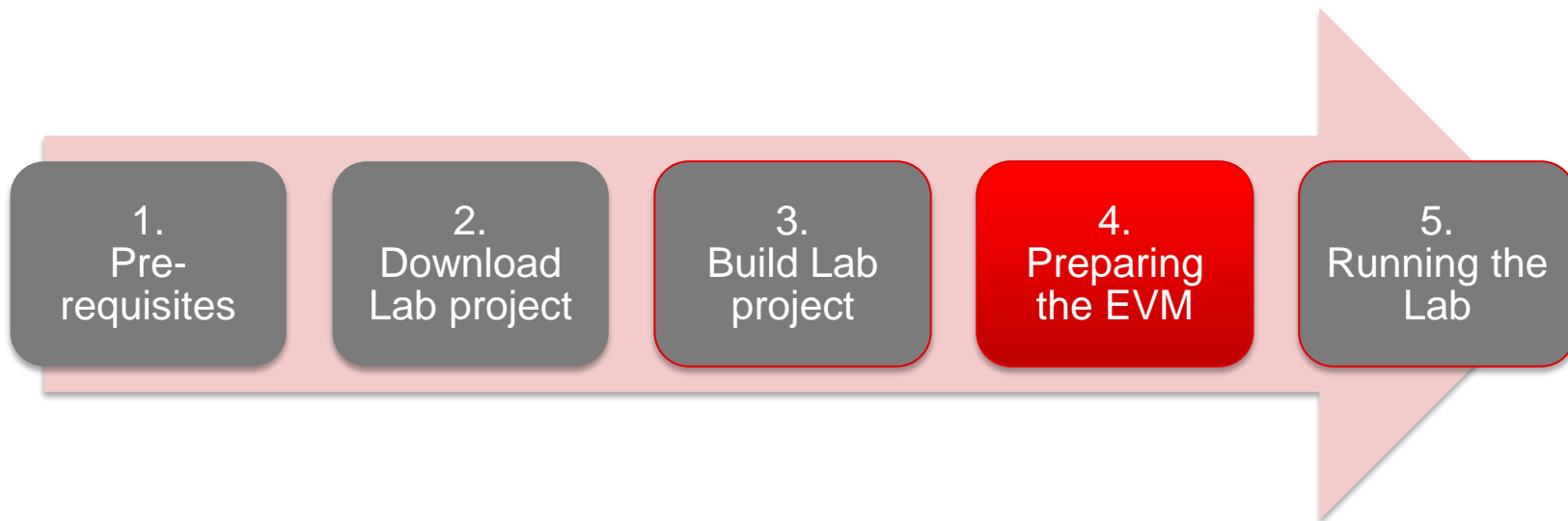
3. Build the Lab



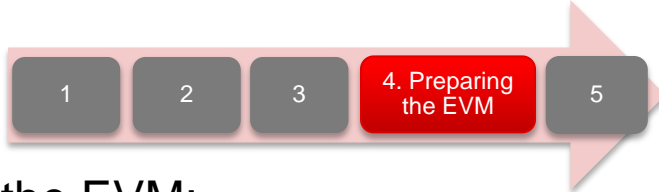
- With the vitalSigns_lab project selected in Project Explorer, open the **Project** menu and select **Build Project**.
- On successful completion of the build, you should see the output in CCS console as shown here and the following two files should be produced in the project debug directory
 - xwr14xx_vitalSigns_lab_mss.xer4f
 - xwr14xx_vitalSigns_lab_mss.bin
- If the build fails with errors, please ensure that all the pre-requisites are installed as mentioned in the mmWave SDK release notes.



Steps



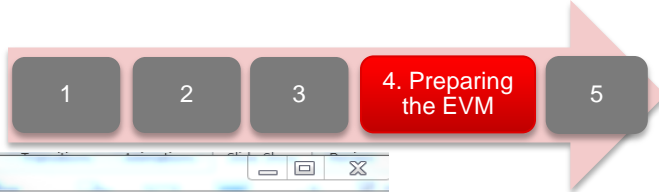
4.1 Preparing the EVM



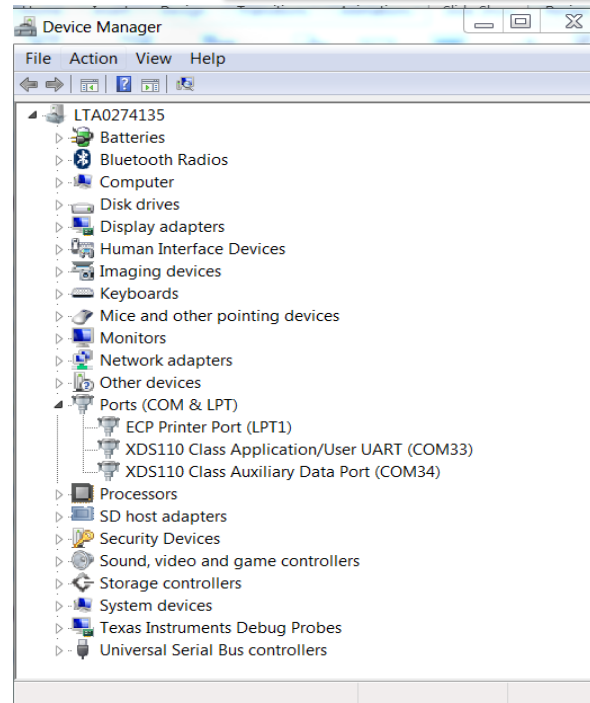
- There are two ways to execute the compiled code on the EVM:
 - Deployment mode: Flashing the binary (.bin image) on to the EVM serial flash
 - In this mode, the EVM boots autonomously from flash and starts running the bin image.
 - Debug mode: Downloading and running the executable (.xer4f image) from CCS.
 - You will need to flash a small CCS debug firmware on the EVM (one time) to allow connecting with CCS. This debug firmware image is provided with the mmWave SDK.
 - As a recap, the build process in Step 3 produces both the .bin and .xer4f images.

- This presentation explains the second method i.e. Debug mode (CCS).
 - To prepare the EVM for debug mode, we start with flashing the CCS debug firmware image.
 - Please note that the same flashing process can be used to flash the Lab binary to run it in deployment mode.

4.2 Connecting to the EVM



- Power on the EVM using a 5V/5A power supply.
- Connect the EVM to your PC and check the COM ports in Windows Device Manager
- The EVM exports two virtual COM ports as shown below:
 - XDS110 Class Application/User UART (COM_{UART}):
 - Used for passing configuration data and firmware to the EVM
 - XDS110 Class Auxiliary Data Port (COM_{AUX})
 - Used to send processed radar data output
- Note the COM_{UART} and COM_{AUX} port numbers, as they will be used later for flashing and running the Lab.



COM_{UART} : COM33 COM_{AUX} : COM34

- The actual port numbers on your machine may be different

4.3 Flashing CCS debug firmware

1

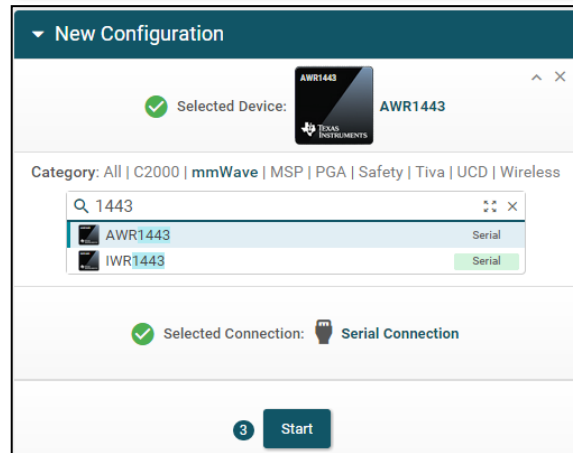
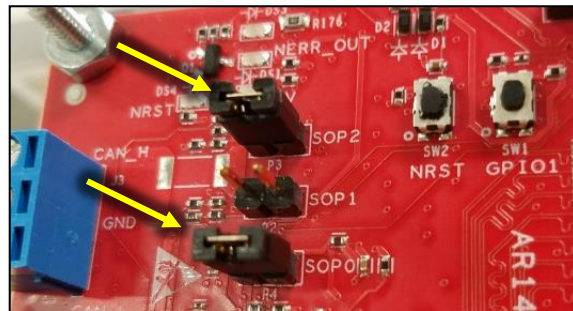
2

3

4. Preparing the EVM

5

1. Put the EVM in flashing mode by connecting jumpers on SOP0 and SOP2 as shown in the image.
2. Open the **UniFlash** tool
3. In the **New Configuration** section, locate and select the appropriate device (AWR1443 or IWR1443)
4. Click **Start** to proceed



4.3 Flashing CCS debug firmware



1. In the **Program** tab, browse and locate the Radar SS and MSS images shown below:

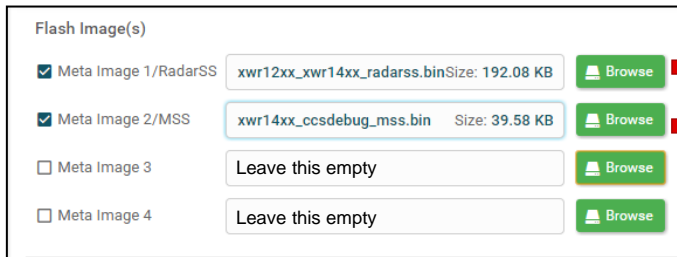


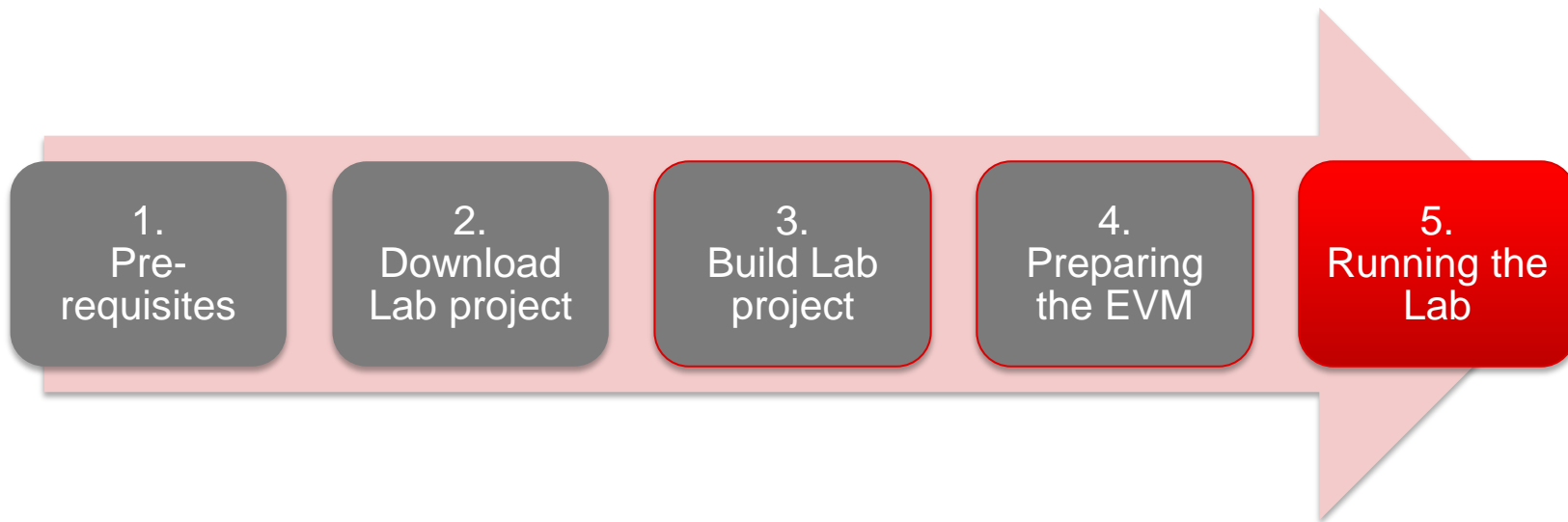
Image	Location
Meta Image 1/RadarSS	C:\ti\mmwave_sdk_<ver>\firmware\bss\xwr12xx_xwr14xx_radarss.bin
Meta Image 2/MSS	C:\ti\mmwave_sdk_<ver>\packages\ti\utils\ccsdebug\xwr14xx_ccsdebug_mss.bin

2. In the **Settings & Utilities** tab, fill the **COM Port** text box with the Application/User UART COM port number (**COM_{UART}**) noted earlier



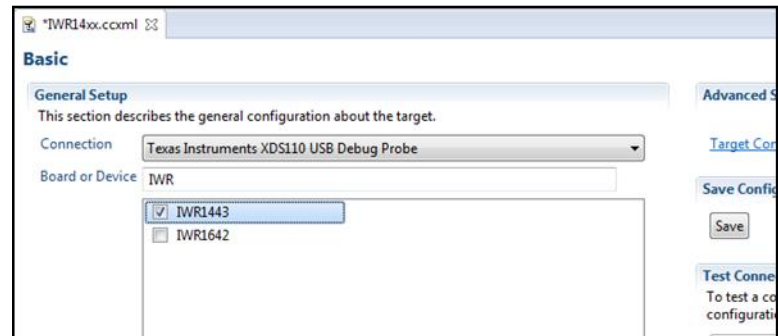
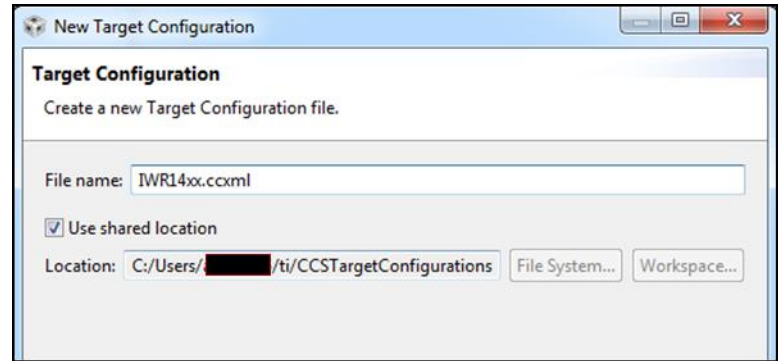
3. Return to the **Program** tab, power cycle the device and click on **Load Images**
4. When the flash procedure completes, UniFlash's console should indicate: [SUCCESS] Program Load completed successfully
5. Power off the board and remove the jumper from only header **SOP2** (this puts the board back in functional mode)

Steps




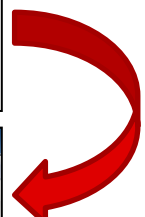
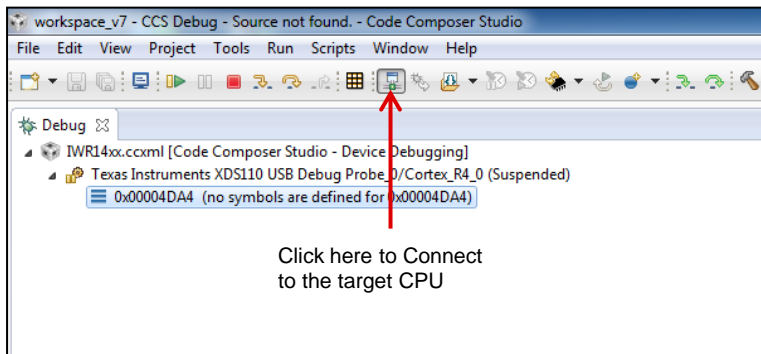
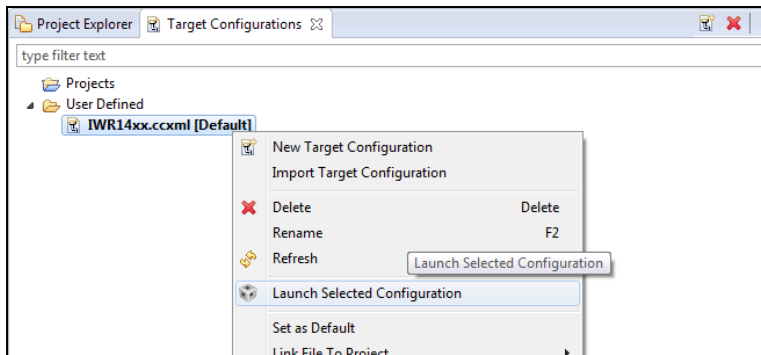
5.1 Connecting EVM to CCS

- It is assumed that you were able to download and build the Lab in CCS (completed steps 1, 2 and 3)
- To connect the Radar EVM to CCS, we need to create a target configuration
 - Go to File ► New ► New Target Configuration File
 - Name the target configuration accordingly and check the “Use shared location” checkbox. Press Finish
 - In the configuration editor window:
 - Select “Texas Instruments XDS110 USB Debug Probe” for **Connection**
 - Type **IWR** in the **Board or Device** text box and select **IWR1443** device.
 - Press the **Save** button to save the target configuration.
 - You can press the **Test Connection** button to check the connection with the board.



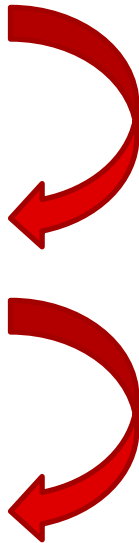
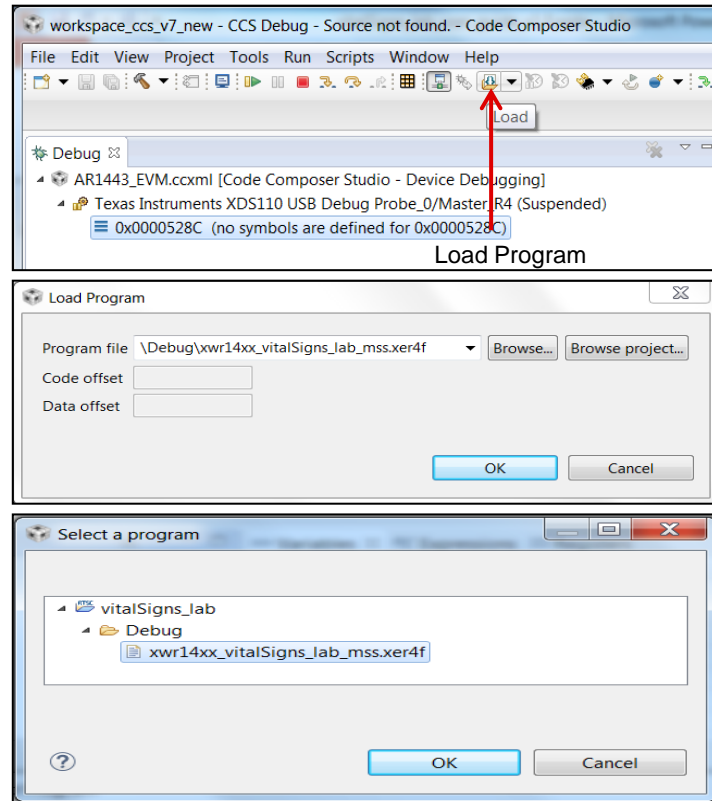
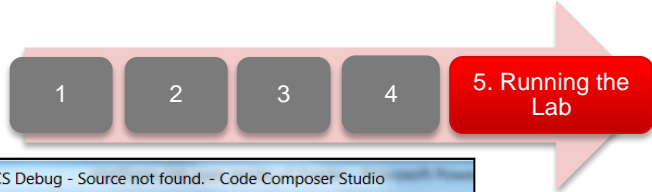
5.1 Connecting - continued

- Go to **View ► Target Configurations** to open the target configuration window.
- You should see your target configuration under **User Defined** configurations.
- Right click on the target configuration and select **Launch Selected Configuration**.
- This will launch the target configuration in the debug window.
- Select the Texas Instruments XDS110 USB Debug probe and press the **Connect Target** button 

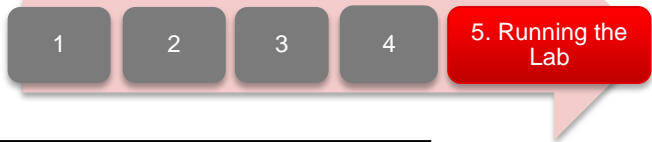



5.2 Loading the binary

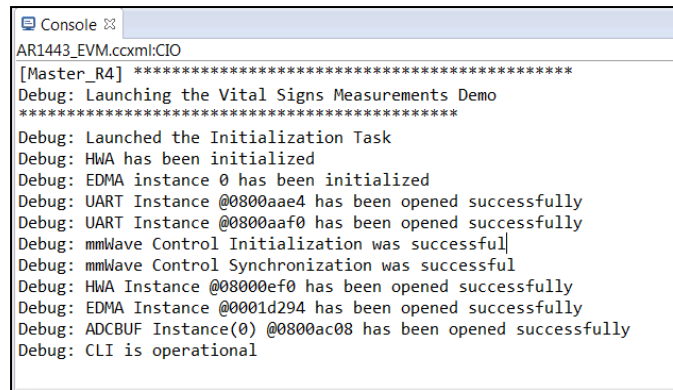
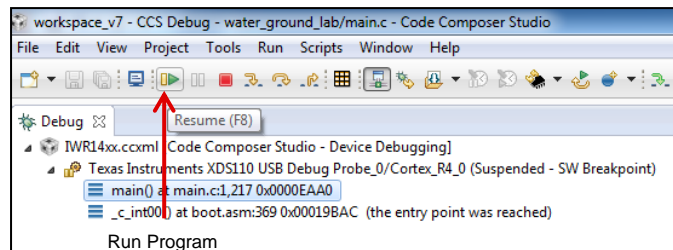
- With the target connected, click on the **Load** button in the toolbar.
- In the **Load Program** dialog, press the **Browse Project** button .
- Select the lab executable (.xer4f) as shown and press OK.
- Press OK again in the **Load Program** dialog.



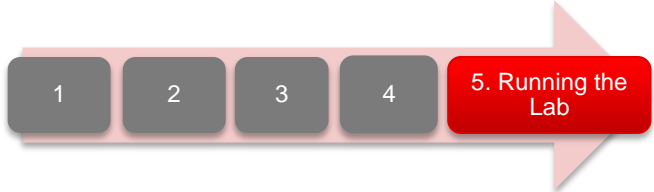
5.3 Running the binary



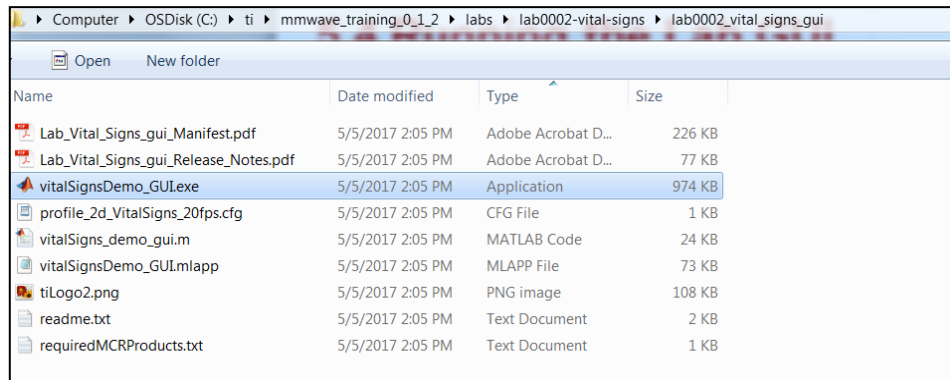
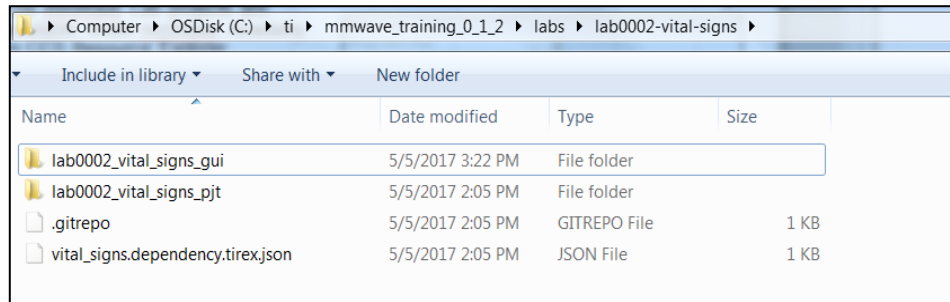
- With the executable loaded, press the Run/Resume button 
- The program should start executing and generate console output as shown.
- If everything goes fine, you should see the “CLI is operational” message which indicates that the program is ready and waiting for the sensor configuration.
- The sensor configuration is sent using the Lab GUI which is based on Matlab.
 - Note: Please ensure that MATLAB runtime R2016b (9.1) is installed as mentioned in the pre-requisites section.



5.4 Running the Lab GUI



- Navigate to the Lab GUI directory as shown
 - The GUI gets downloaded in the Lab directory under mmwave_training.
- Execute the Lab GUI executable (vitalSignsDemo_GUI.exe).



5.4 Running GUI - continued

1

2

3

4

5. Running the Lab

- Press the **Settings** button.
- Update the User UART and Data port numbers with the COM_{UART} and COM_{AUX} ports noted earlier.
- The **Configuration File name** text box should automatically show the name of the profile provided in the GUI directory.
- Press **Start** to load the configuration and start the visualization.
 - At this point, you should see the sensor configuration being sent in the CLI command line window opened by the GUI.

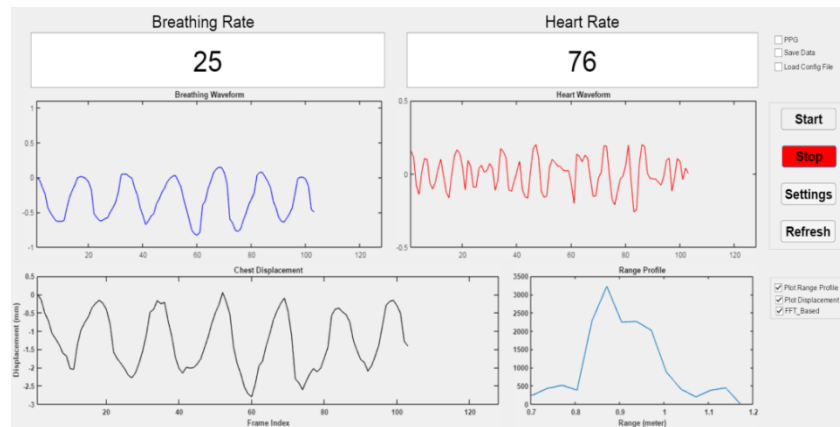


5.4 Running GUI - continued

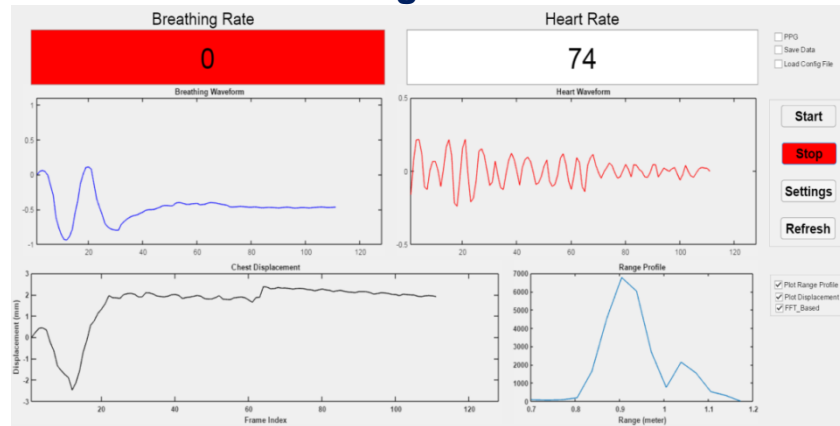
- Have the subject sit comfortably on the chair. As sub-mm chest displacements are being measured the subject is required to be **very still** for accurate measurements
- Make sure that the “**Plot Range Profile**” check box is checked and a peak corresponding to the subject can be seen. Once a peak is seen, press the **Refresh** button.
- Now, chest displacements due to breathing should be clearly visible in the **Breathing Waveform** plot
- Once a few chest displacements have been seen, ask the subject to hold their breath. The breathing-rate should go to zero and turn red, the breathing waveform plot should be more or less constant and the heart rate waveforms should still be **visible**. If the breathing rate does not go to zero OR the heart rate waveform is not visible then either the subject is not properly aligned with the radar or there is interference coming from other moving objects within the Radar field-of-view.
- Wait 20-30 seconds so that enough data frames are received for an accurate estimate of the vital signs

(Note: If the GUI appears to be moving very slowly, disable the Range Profile plot. Once the “Refresh” button is pressed, this plot is no longer needed and can slow down the GUI.)

Normal Breathing



Holding Breath



Learn more about TI mmWave Sensors

- Learn more about xWR1x devices, please visit the product pages
 - IWR1443: <http://www.ti.com/product/IWR1443>
 - IWR1642: <http://www.ti.com/product/IWR1642>
 - AWR1443: <http://www.ti.com/product/AWR1443>
 - AWR1642: <http://www.ti.com/product/AWR1642>
- Get started evaluating the platform with xWR1x EVMs, purchase EVM at
 - IWR1443 EVM: <http://www.ti.com/tool/IWR1443BOOST>
 - IWR1642 EVM: <http://www.ti.com/tool/IWR1642BOOST>
 - AWR1443 EVM: <http://www.ti.com/tool/AWR1443BOOST>
 - AWR1642 EVM: <http://www.ti.com/tool/AWR1642BOOST>
- Download mmWave SDK @ <http://www.ti.com/tool/MMWAVE-SDK>
- Ask question on TI's E2E forum @ <http://e2e.ti.com>



© Copyright 2017 Texas Instruments Incorporated. All rights reserved.

This material is provided strictly “as-is,” for informational purposes only, and without any warranty.
Use of this material is subject to TI’s **Terms of Use**, viewable at TI.com