

mmWave Radar Visualizer (mmWave Low Power Devices)

mmWave Radar Visualizer User Guide



Product Version: 5.5.4.1

Contents

mmWave Radar Visualizer User Guide	1
Contents	2
List of Figures	3
1. Getting Started	4
1.1. Introduction	4
1.2. Quick Start Guide	5
2. Quick Evaluation	6
3. Configuration Dashboard	8
3.1. Device Connection	8
3.2. Data Logging	9
3.3. Configuration Selection	9
3.4. Generate Configuration File	10
3.5. Configuration Info	11
3.6. CLI Command Window	12
3.6.1. CLI command and response	12
3.6.2. Rx channel gain and phase measurement	12
4. Presence Motion and Tracking	13
4.1. Plots	13
4.2. Info Section	15
4.3. Resource Section	15
5. Plots	16
6. Power Estimation	17
7. Flash	18
8. Error Handling	22
8.1. ATE Calibration Unavailable	22
8.2. Failed to Connect to COM port	22
8.3. Error in Device Reset	23
8.4. Error in Configuring the Device	23
8.5. Error in Flashing	24
8.6. Version Mismatch Error	24
8.7. No Plots displayed / No Visualizer updates / Visualizer Freeze	24
9. Limitations	25

List of Figures

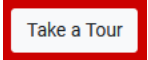
Figure 1: Home Page	4
Figure 2: Quick Evaluation	6
Figure 3: Manual Reset of device	7
Figure 4: Overview of Configuration Dashboard	8
Figure 5: (a) Device Connection (b) Configuration Selection	9
Figure 6: Data Logging	9
Figure 7: Uploading Configuration to device	10
Figure 8: Device Configuration Input	12
Figure 9: Presence Motion and Tracking Overview	13
Figure 10: Presence Plot	14
Figure 11: Tracker Plot	14
Figure 12: Presence Motion & Tracking Info section	15
Figure 13: Miscellaneous Plots	16
Figure 14: Overview of Power Estimator	17
Figure 15: Overview of Flash Tab	18
Figure 16: Note on Flashing	18
Figure 17: Flashing Process	20
Figure 18: Flashing Success message	20
Figure 19: ATE Calibration Error	22
Figure 20: COM Port Failed Error	22
Figure 21: Error in Halting Device	23
Figure 22: Device Configuration Error	23
Figure 23: Flashing Error	24
Figure 24: Version Mismatch Error	24




1. Getting Started

1.1. Introduction

The Visualizer executable (exe) file is located at <SDK_INSTALL_ROOT>/tools/visualizer/ folder with the name visualizer.exe

When the above executable is run, the Visualizer application opens up in the system default browser with the Welcome Page (Home Page) as shown below. This page provides an overview of the Visualizer and links to various documentation that can be referred to while working with the mmWave device and the Visualizer, along with short descriptions of the different application tabs.

Clicking  the button at the top right corner of the Home Page gives a quick tour on how to navigate through various sections of the Visualizer.

Clicking on the  button in the title bar will redirect to this page. The two buttons on the top right corner of the title bar,  or  provides hints to zoom the screen in or out respectively. This is useful when working with screens of different resolutions. Please note however that this shortcut may not work on all browsers, which is when the user can do a manual zoom in / zoom out from the browser settings (alternatively through appropriate keyboard shortcuts – commonly Ctrl + or Ctrl -).

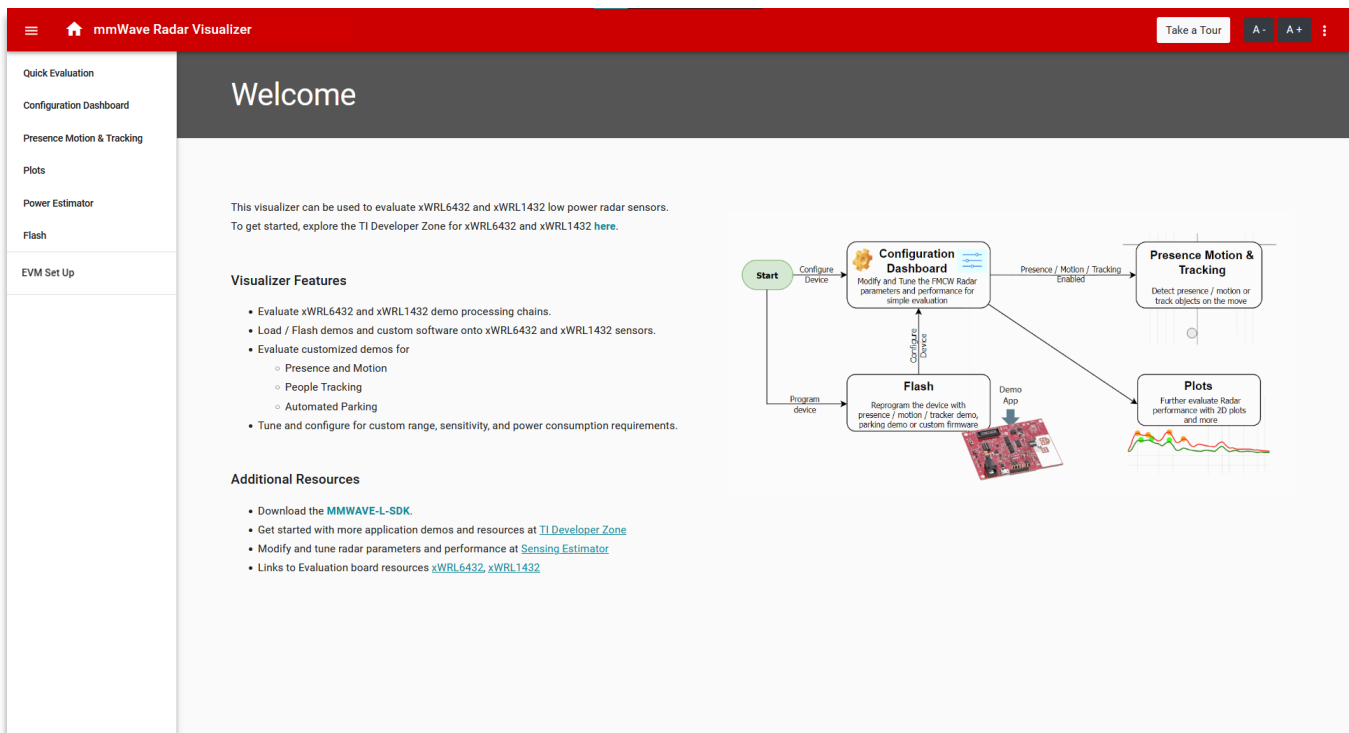






Figure 1: Home Page

The current version of the Visualizer has 6 functional tabs enunciated below

- **Quick Evaluation** – This tab enables to quickly evaluate the default demo running on the device with a single click, which would then display a presence detection plot.
- **Configuration Dashboard** – This tab provides the functionality to modify and tune the FMCW radar parameters and performance for simple evaluation.
- **Presence Motion & Tracking** – This tab displays the plot for presence / motion detection or object tracking and related information.
- **Plots** – This tab displays various plots to evaluate the radar's performance.
- **Power Estimator** – This tab helps estimates the average power consumption for a given configuration (chirp parameters along with other inputs) and compares it against the average power reported by the device.
- **Flash** – This tab can be used to reprogram the device with presence / motion / tracker demo, parking demo or any non-default firmware / custom appimage.

1.2. Quick Start Guide

1. Connect the EVM to the PC via USB.
2. Optional Step: Evaluate the pre-built demo.
 1. Navigate to the **Quick Evaluation** tab.
 2. Click on  to start evaluation of the pre-built demo. This should show a presence detect plot.
3. Optional Step: Program a non-default firmware / app-image
 1. Navigate to the **Flash** tab.
 2. Wait for AUTO detection of COM ports (else press ). Alternatively, manually select device COM port (if not already selected).
 3. Follow the subsequent steps shown in the Visualizer.
4. Configure Device
 1. Navigate to the **Configuration Dashboard** tab.
 2. Wait for AUTO detection of COM ports (else press ). Alternatively, manually select device COM port (if not already selected).
 3. Select preset configuration under **Configuration Selection** drop-down list.
 4. Click on .
5. **Plots** tab should display point cloud information.

2. Quick Evaluation

This tab enables to quickly evaluate the default demo running on the device with a single click of a button, which would then display a presence detection plot as shown in the figure below. This tab houses two buttons **(Re)Start Eval** and **Stop Eval** (disabled by default) aside displaying the current frame and the presence detection plot. After connecting the device, navigate to the Quick Evaluation tab, click **(Re)Start Eval** button and wait for the device data to be visualized in the application. The then enabled **Stop Eval** button is used to halt the ongoing evaluation configuration.



Figure 2: Quick Evaluation

In case of a custom device, the Visualizer will prompt with the following popup, for ensuring that the device is reset manually.

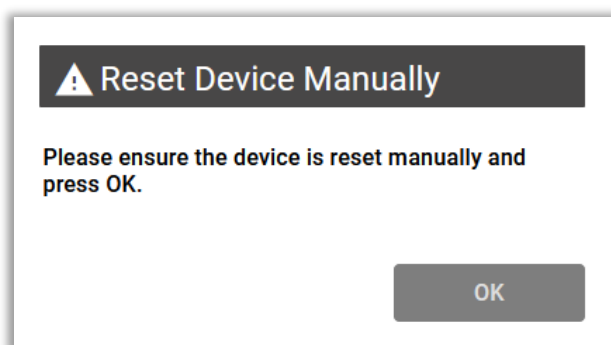


Figure 3: Manual Reset of device

3. Configuration Dashboard

The Configuration Dashboard helps configure the device and also displays some basic configuration information. Before configuring the device, please make sure the EVM is in `FUNCTIONAL MODE` (shown as a popup first time when the Visualizer opens this tab).

Clicking the [Take a Tour](#) button at the top right corner of the Page gives a quick tour on how to navigate through various sections of the Configuration Dashboard.

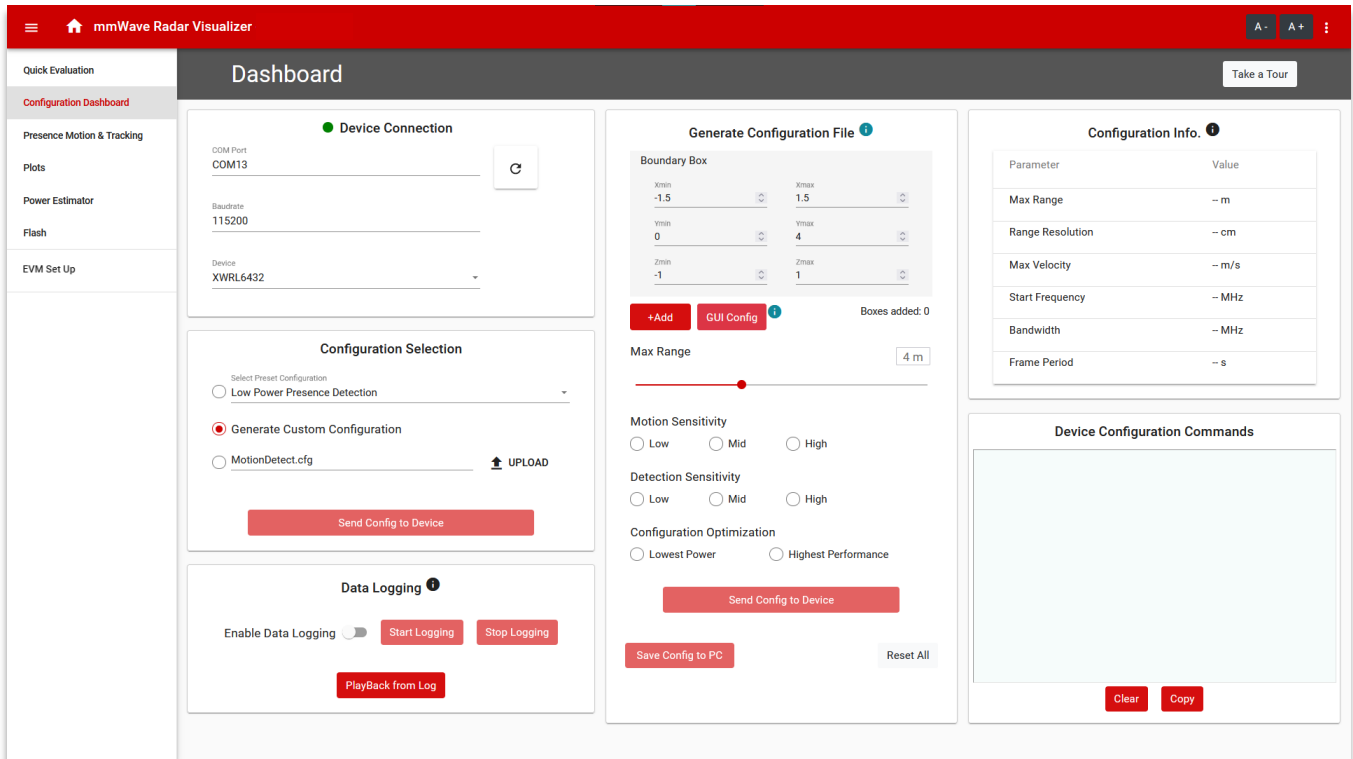


Figure 4: Overview of Configuration Dashboard

The following sections elucidate various elements of the **Configuration Dashboard** tab.

3.1. Device Connection

This section enables selection of the Device, COM Port it is connected to and its Baudrate. Please wait for a few seconds after connecting the device and opening this tab, and the device COM Port should be automatically detected and populated. In this case, the circular symbol turns green (see [Figure 5 \(a\)](#)). If the Visualizer couldn't recognize the device COM Port the circular symbol remains red. Clicking the refresh icon fetches all available XDS110 User UART COM Ports which can then be selected from the dropdown list. If the Visualizer fails to auto-detect COM Port, please check if the device connected is recognized by the system under *Device Manager*, or enter it manually (may be required especially in case of a custom EVM).

Note: To know the COM port of the device, look for the XDS110 Class Application/User UART (COMx) or any other COM Port specific to the connected EVM (x in COMx can vary from PC to PC) inside the Ports (COM & LPT) of Device manager.

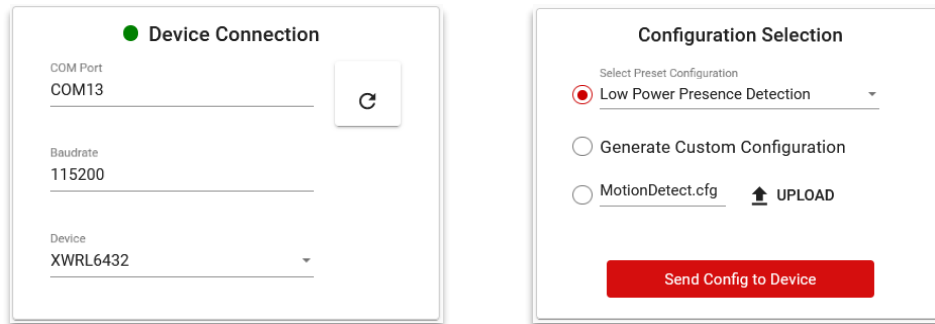


Figure 5: (a) Device Connection

(b) Configuration Selection

3.2. Data Logging

This section enables to log data output from the device. Toggling will capture the logs as the device starts running any configuration (starting few frames may get skipped). The **Start Logging** and **Stop Logging** buttons can be used to dynamically start and stop capture of data output from the device respectively. In either case, a file with a name `frameData*.log` gets created under a temp folder in the same directory from where the Visualizer executable is running.

The **PlayBack from Log** button, helps select a previously generated log file (generated by the Visualizer) to replay data offline (w/o the device connected).

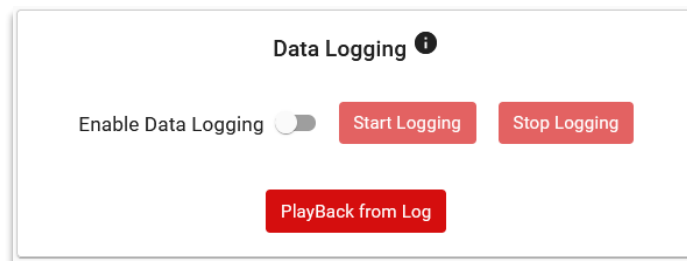


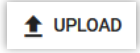
Figure 6: Data Logging

3.3. Configuration Selection

This section enables configuration of the device with configuration files (*.cfg file) or any custom configuration generated with the Visualizer. A configuration contains a bunch of CLI commands used to configure the device.

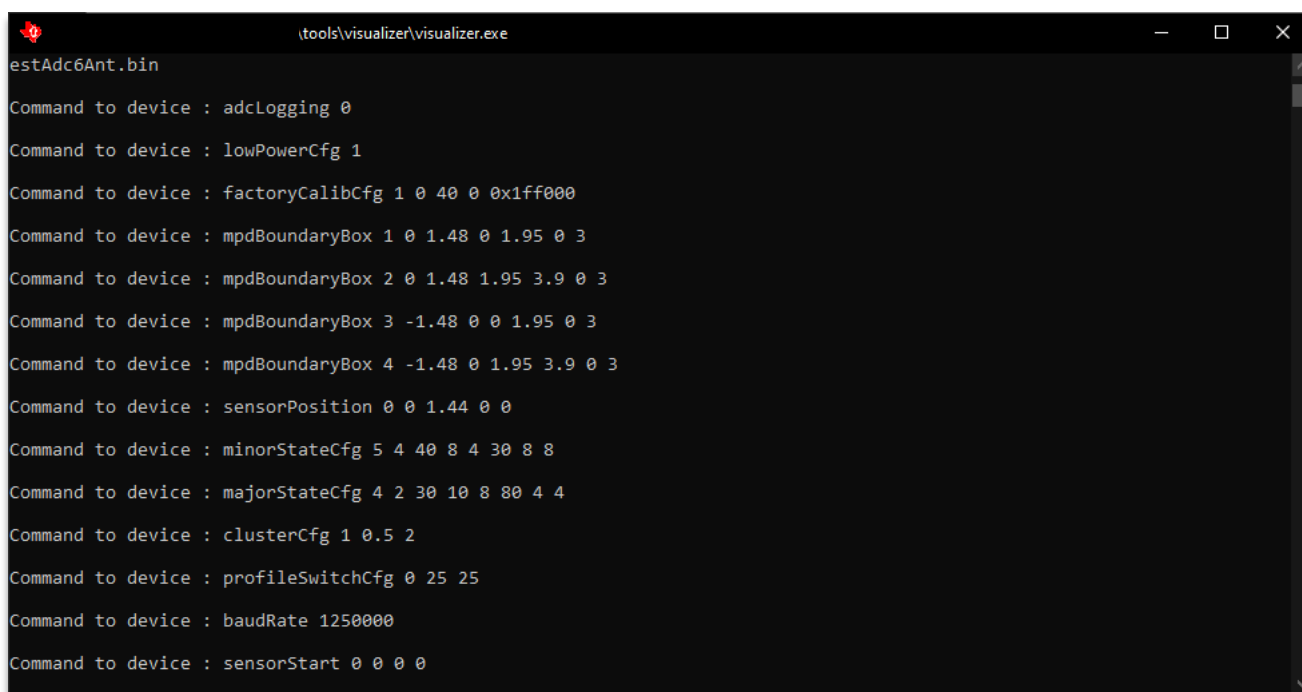
The Visualizer provides three options to configure the device (see [Figure 5 \(b\)](#)). The user may select any of the following three options and then click on

Send Config to Device

1. **Select Preset Configuration** – Any preset configuration (either Presence/ Motion Detection/ Parking or Tracker) can be selected with this dropdown list.
2. **Generate Custom Configuration** – Selecting this option enables the **Generate Configuration File** section which can be used to generate and send custom configuration to the device.
3. **Upload** – Any custom configuration can be uploaded using this option. Click to upload  any configuration file (* .cfg file) for sending to the device.

In case of a custom device, the Visualizer will prompt with a popup for ensuring that the device is reset manually (see [Figure 3](#)).

The CLI commands to configure the device will be output on the console window as shown below as well as the **Configuration Dashboard** (see [Figure 8](#)).



```


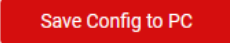
estAdc6Ant.bin
Command to device : adcLogging 0
Command to device : lowPowerCfg 1
Command to device : factoryCalibCfg 1 0 40 0 0x1ff000
Command to device : mpdBoundaryBox 1 0 1.48 0 1.95 0 3
Command to device : mpdBoundaryBox 2 0 1.48 1.95 3.9 0 3
Command to device : mpdBoundaryBox 3 -1.48 0 0 1.95 0 3
Command to device : mpdBoundaryBox 4 -1.48 0 1.95 3.9 0 3
Command to device : sensorPosition 0 0 1.44 0 0
Command to device : minorStateCfg 5 4 40 8 4 30 8 8
Command to device : majorStateCfg 4 2 30 10 8 80 4 4
Command to device : clusterCfg 1 0.5 2
Command to device : profileSwitchCfg 0 25 25
Command to device : baudRate 1250000
Command to device : sensorStart 0 0 0 0

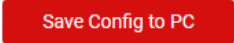

```

Figure 7: Uploading Configuration to device

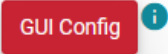

When all the configuration commands are sent successfully, the application starts displaying the frame number (in the **Presence Motion & Tracking** Tab as well as in the **Plots** Tab).

3.4. Generate Configuration File

This section helps generate a custom configuration with the help of the Visualizer. The  and  buttons are enabled when **Max Range** is configured or at least one boundary Box is added along with the other options, and stays disabled otherwise. A config file with name `generatedConfig*.cfg`

gets created upon clicking the . Clicking  sends the configuration directly to the device without having to save it first.


The various parameters that can be used for generating a custom configuration file are described below.

1. **Boundary Box** – Used to configure boundary boxes for motion and presence demo. The boundary boxes can also be configured through GUI by clicking on . Click on  next to this button to view help on the GUI config tool usage.
2. **Max Range** – Set the maximum range up to which the radar should detect objects. This parameter will be used to determine the bandwidth of the chirp, using the maximum bandwidth for the range selection. The text box on the right displays the range currently set.
3. **Motion Sensitivity** – Sensitivity to motion in the scene. Higher sensitivity will detect slower moving objects. Lower sensitivity will disregard slower moving objects.
4. **Detection Sensitivity** – Sensitivity for object detection threshold. Increasing sensitivity will detect more points, but may increase noise. Decreasing sensitivity will detect fewer points and may improve resistance to noise.
5. **Configuration Optimization** – Optimizing for lower power will decrease the number of chirps in a frame. This will consume lower power but may provide fewer detected points at maximum range. Optimizing for higher performance will increase the number of chirps in a frame. This will increase power consumption but also will improve detection at maximum range but consume higher power.

3.5. Configuration Info

This section displays config information based on the configuration selected in the **Configuration Selection** pane. The list of info. is described below.

1. **Max Range** –Maximum distance the radar sensor can detect an object.
2. **Range Resolution** – Smallest distance between two objects that allows them to be detected as separate objects.
3. **Max Velocity** – Maximum relative velocity of an object that can be detected by the radar sensor.
4. **Start Frequency** – Start frequency of the selected chirp configuration.
5. **Bandwidth** – Frequency bandwidth output by the device based on the chirp configuration.
6. **Frame Period** – Time in between the start of two frames.

Clicking on  populates this section as per the current configuration of the device.

3.6. CLI Command Window

3.6.1. CLI command and response

The CLI Command Windows shows the configuration commands being sent to the device and the response received from the device.

3.6.2. Rx channel gain and phase measurement

Because of imperfections in antenna layouts on the board there is a need to calibrate the sensor to compensate for bias in the range estimation and the receive channel gain and phase imperfections. The motion detection demo provides the ability to do the measurement and compensation. The measurement procedure is configured using a CLI command `measureRangeBiasAndRxChanPhase`.

The coefficients are transmitted to the host within the TLV packet each frame. These coefficients are displayed on the CLI Command Window, when used with the right demo and appropriate `cfg` file. More details on this feature can be found in the “Motion and Presence Detection OOB Demo” section of the SDK User Guide ([README_FIRST_xWRL6432.html](#)).

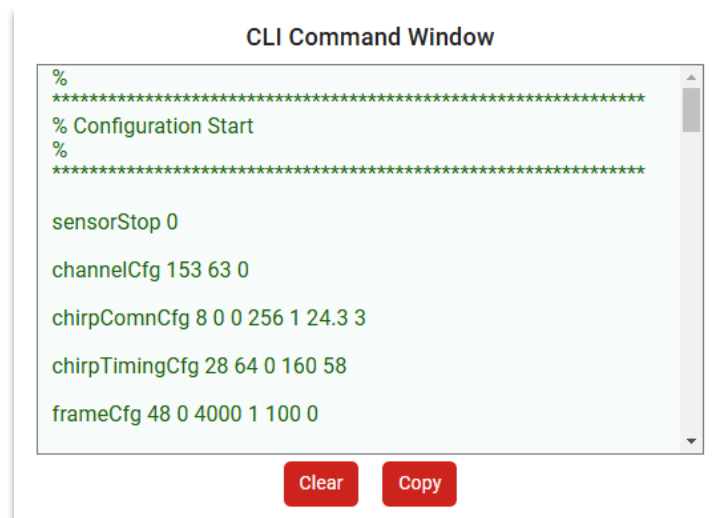


Figure 8: Device Configuration Input

4. Presence Motion and Tracking

This tab shows the presence / motion / object tracking information when the device is configured with any config that has presence, motion or tracking enabled (to know how to enable presence, motion or tracking please refer to `*_TuningGuide.pdf` document under `<SDK_INSTALL_ROOT>/docs/` folder).

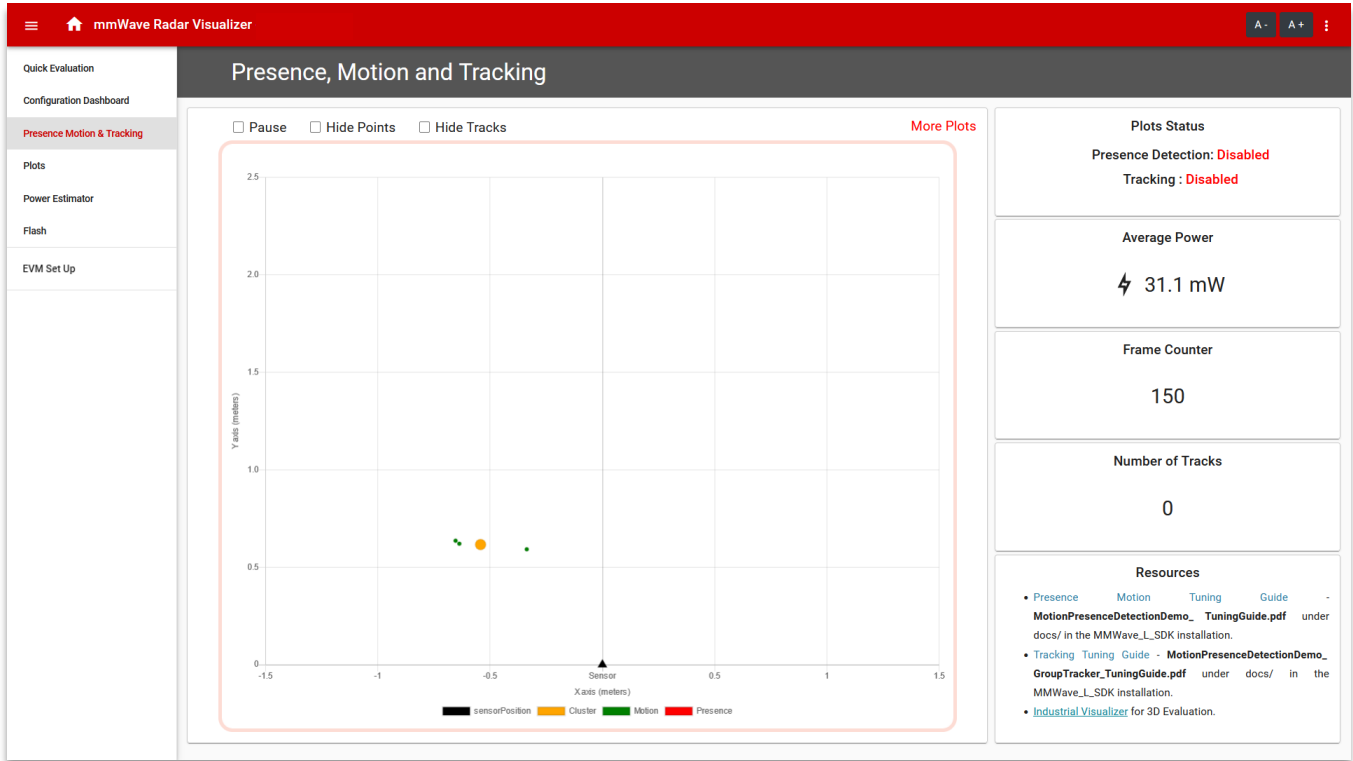


Figure 9: Presence Motion and Tracking Overview

The following sections elucidate various elements of the **Presence Motion and Tracking** tab.

4.1. Plots

This is the most important section, which displays different plots based on the current device configuration. The different kinds of plots displayed are detailed below:

1. **Motion** – If the device is configured for motion detection, it displays the point cloud information along with clusters centroids of the same. The points detected are shown in red dots. The larger orange dots represent centroids of different clusters of points.
2. **Presence** – If the device is configured for presence detection, it displays the point cloud information along with the highlighting the zones in which presence is detected. The zones in which presence (motion) is detected is highlighted in green and the points detected each of these zones are shown in green dots. The larger orange dots represent centroids of different clusters of points.

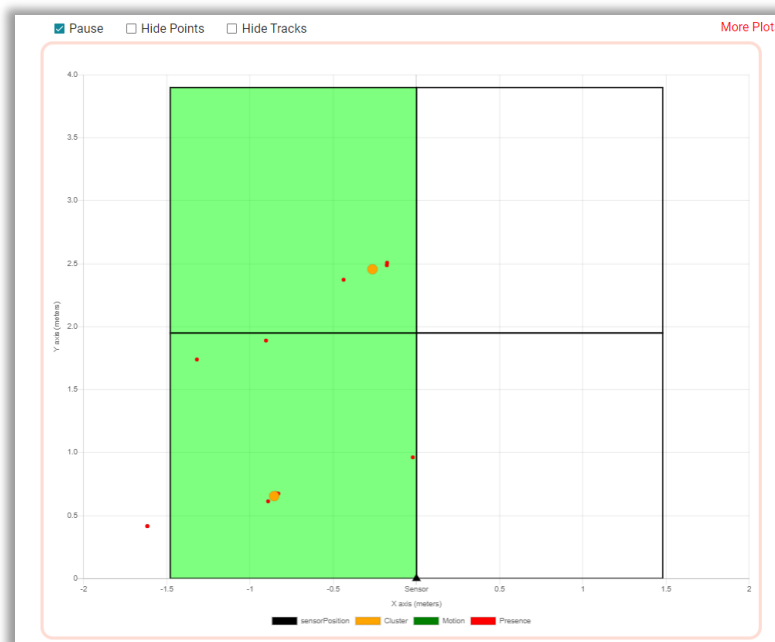


Figure 10: Presence Plot

3. **Tracking** – If the device is configured for object tracking, it displays the point cloud information along with the object tracking information. The tracked objects appear as bubbles. On hover these bubbles display the associated ID.

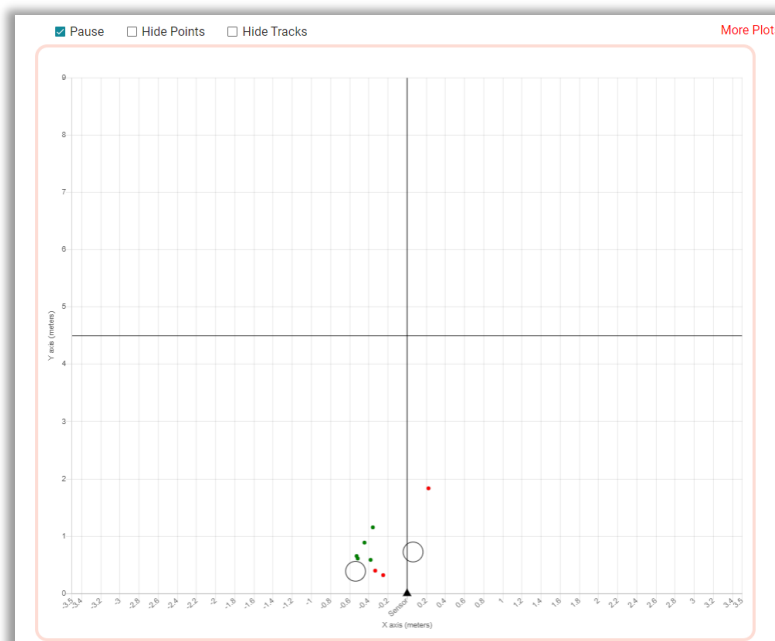


Figure 11: Tracker Plot

There are three checkboxes in the plots ☐ **Pause** ☐ **Hide Points** ☐ **Hide Tracks**, which when enabled freezes the plot in its current state, hides the point cloud information and hides the tracked objects respectively.

There is also a hyperlink on the top right corner of this section [More Plots](#). This links to the **Plots** tab.

4.2. Info Section

This section displays information relevant to the application that is running on the device and based on its configuration viz., if presence or tracking is enabled in the current configuration, a running count of the number of frames received, the number of object tracks detected (if tracking is enabled) and the average power consumption reported.

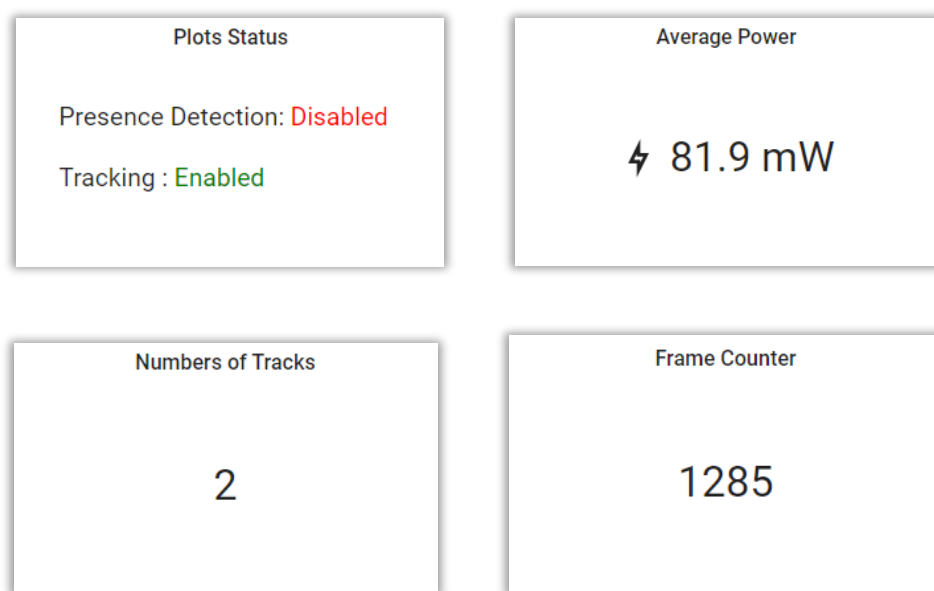


Figure 12: Presence Motion & Tracking Info section

4.3. Resource Section

This section links to additional resources including the tuning guides for various configurations.

5. Plots

This tab shows four important plots that can be used for further analysis viz., Range Profile, X-Y plot, Y-Z plot and Power graph. The number of Frames received and number of Points Detected also shown.



Figure 13: Miscellaneous Plots

The points in the plot will be either Red or Green in color based on the configuration (Presence - Red, Motion - Green).

The Pause checkbox when selected halts any further updates to the visualized plots.

6. Power Estimation

This tab estimates the average power consumption (**Estimated Average Power**) for a given configuration (chirp parameters along with other inputs shown below) and displays it against the average power reported by the device (**Reported Average Power**).

Before switching to this tab, please ensure that the device is running a demo and is configured via the Visualizer (which is when you would see the Power Calculator Inputs pre-populated).

The number fields on the left are input fields and those on the right are output fields. The **mmWave device** should be automatically picked up by the power estimator based on the device configuration. In case it isn't, the user can select the **mmWave Device** from the drop-down list, and the **EVM Board** would then be picked up automatically.

While the Power Calculator Inputs are auto populated by the Visualizer (based on the device configuration), the user can also change the parameters to estimate the power based on custom inputs. This can be done by toggling the **Edit** switch at the top right of **Power Calculator Inputs** section. Any subsequent changes in the input fields should automatically update the output fields. The user can click Sync with Device to reset the Power Calculator inputs to match the current device configuration.

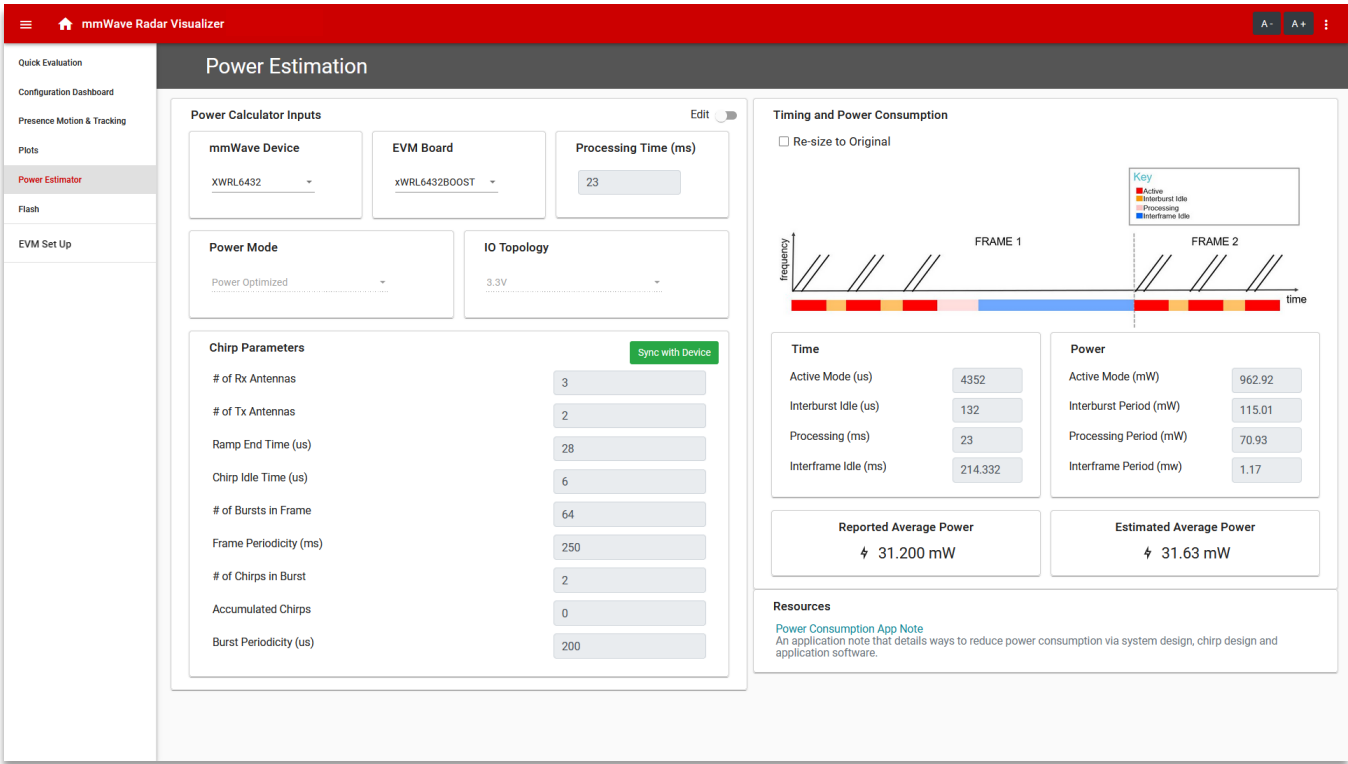


Figure 14: Overview of Power Estimator

7. Flash

This tab enables flashing an `appimage` generated by the SDK. The device runs a default built-in `appimage`. This step is optional and allows flashing any non-default `appimage` onto the device.

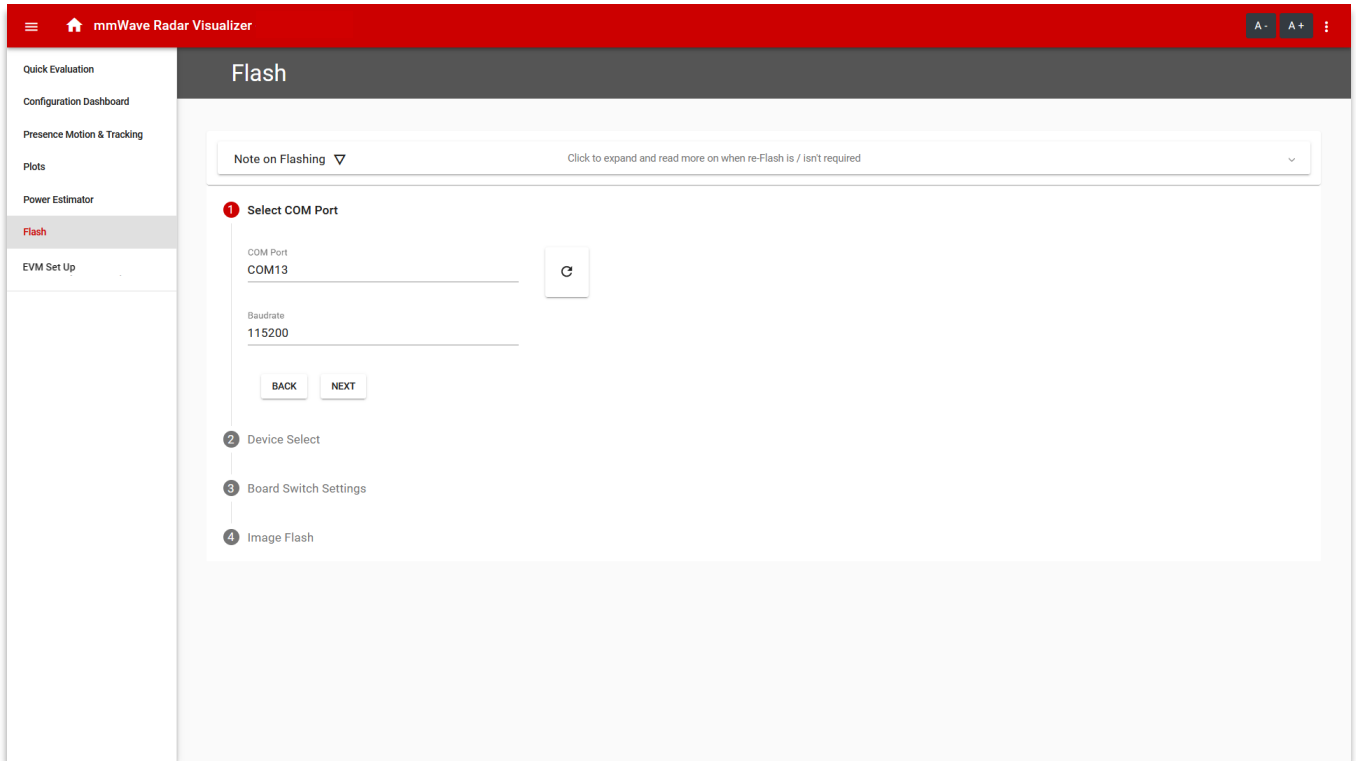


Figure 15: Overview of Flash Tab

There is a blurb at the top of Flash Tab which suggests when a reflash is/is not required.

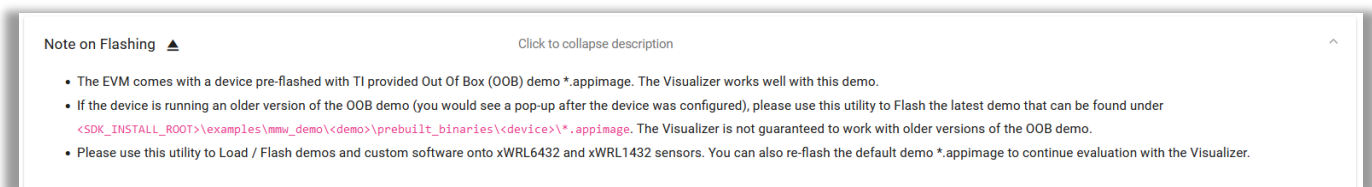




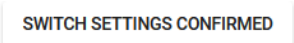
Figure 16: Note on Flashing



Click open the expansion panel to see it as shown in the above figure. Click it again to collapse it.

Flashing the device using the Visualizer involves the following steps

1. **Select COM Port** – Please wait for a few seconds after connecting the device and opening this tab, and the device COM Port should be automatically detected and populated. Otherwise, you can click on refresh icon to detect the COM Port. To know the COM Port of the device, look for XDS110 Class Application/User UART (COMx) or any other, COM Port specific to the connected EVM under the Ports (COM & LPT) of system Device manager. The COM Port can also be entered manually.

During device configuration, the Baudrate is set to 115200 by default. Select from options available in the drop-down list or enter a custom value into the Baudrate field. After the COM Port is selected, and Baudrate is set, click .

2. **Device Select** – Select the device to be flashed with the appimage. After selecting the device click .
3. **Board Switch Settings** – The application then displays information on configuring the device. This step is mandatory, it changes the state of the device from FUNCTIONAL MODE to FLASHING MODE. After doing the necessary switch click .
4. **Image Flash** – The last step is to upload the appimage. There are three options to select the appimage to flash the device with. The first two radio buttons are enabled only when the Visualizer is run from the standard location (within the installed package) and these can be used to flash the pre-built appimage

that comes with the SDK. To upload a custom appimage, click on , select the appimage and click **OK**. Click on  to flash the appimage on to the device.

Below are snapshots of the application window and the console after clicking FLASH button.

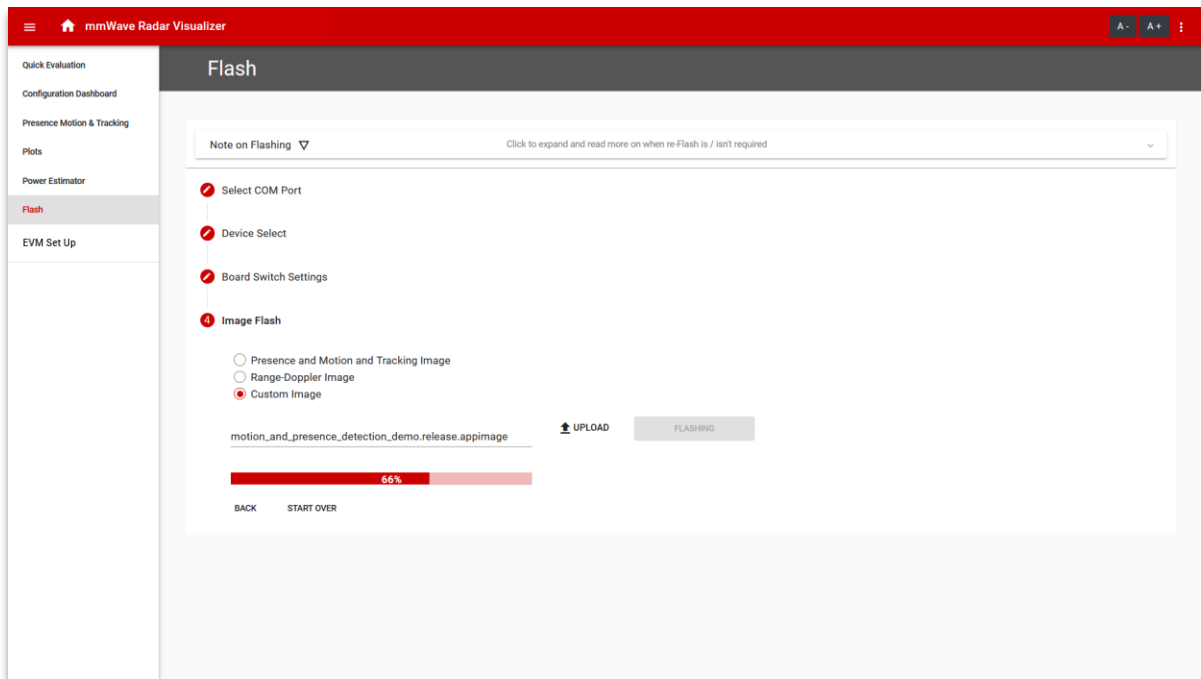


Figure 17: Flashing Process

If the Flashing was successful, a success message is shown as in the figure below.

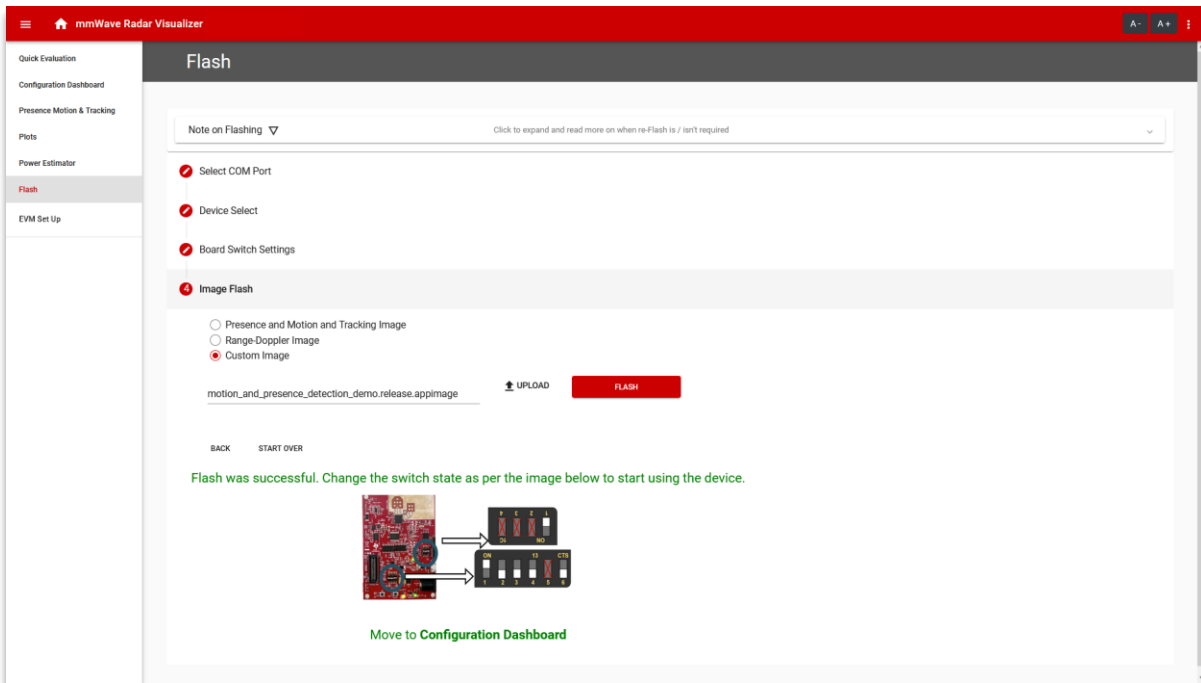


Figure 18: Flashing Success message

If the Flashing was unsuccessful, an error message will be shown in place of the Success message.

Upon reaching this step where you got the flash success message, change the switch settings of the device as per the image displayed on the application image. The device is now ready to be used in the FUNCTIONAL MODE.

Note: In case of any error shown in the console or as a popup, refresh the application window, re-connect the device, check the switch settings and repeat the above steps again.

8. Error Handling

This section details the various errors reported by the Visualizer.

8.1. ATE Calibration Unavailable

If the latest ATE calibration appimage is not flashed on to the device, this pop-up is displayed.

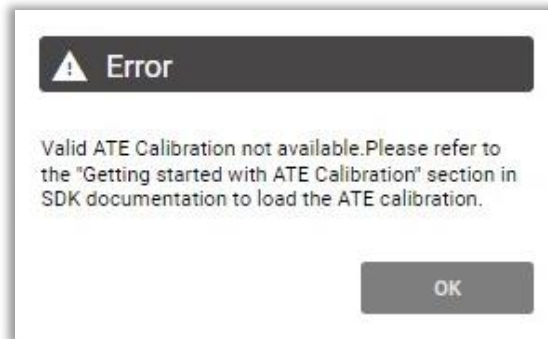


Figure 19: ATE Calibration Error

To resolve the above error, please flash the device with the latest ATE calibration appimage.

8.2. Failed to Connect to COM port

This error will mostly popup when the Visualizer is unable to connect to the device COM port. Please check if the entered COM port is correct and ensure that no other application (serial terminal) is connected to the device COM port.

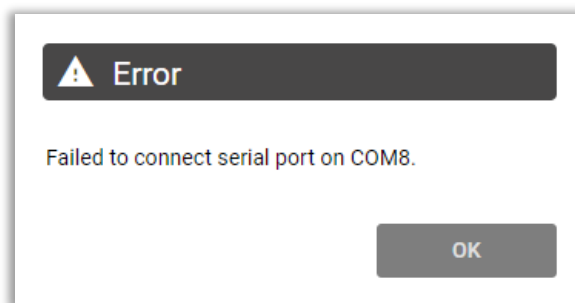


Figure 20: COM Port Failed Error

8.3. Error in Device Reset

The following error is seen when the Visualizer is unable to halt a running device. Ensure that the device is indeed connected, and if the error persists power cycle the EVM (and re-connect) and restart the Visualizer.

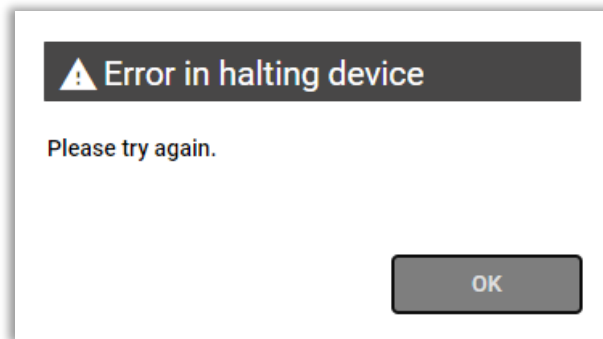


Figure 21: Error in Halting Device

8.4. Error in Configuring the Device

If the device is in `FLASHING MODE` and you try to send configuration to the device, either of the following two errors might be reported.

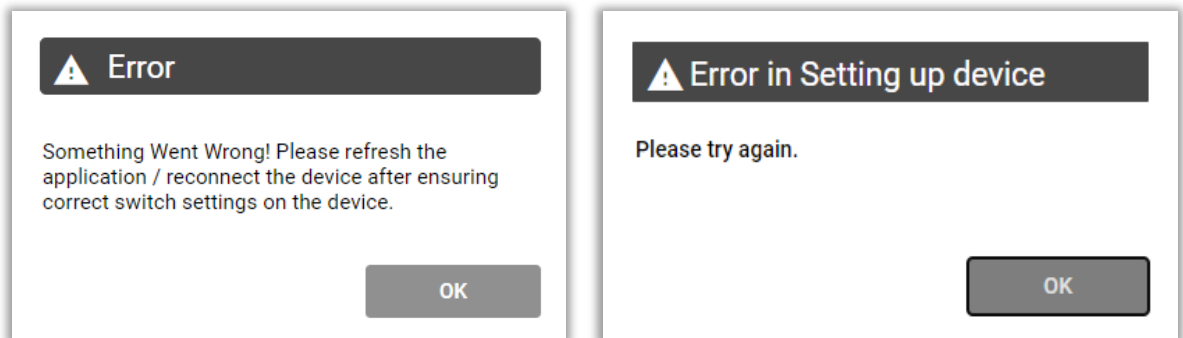


Figure 22: Device Configuration Error

8.5. Error in Flashing

If flashing the device takes too long and you see the following warning, please check if the device is switched to FLASHING MODE.

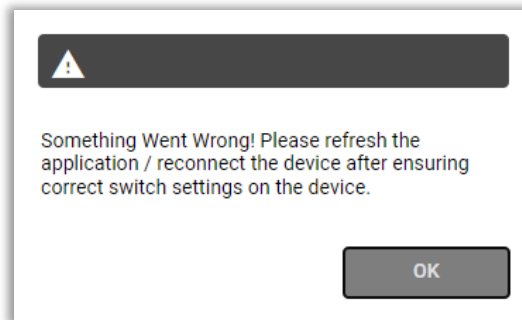


Figure 23: Flashing Error

The previous error mostly occurs when you try to flash the device that is in FUNCTIONAL MODE. It is also recommended to reconnect the USB and restart the visualizer in the event of such an error.

8.6. Version Mismatch Error

The current version of the Visualizer is supported on the latest version of the SDK it is coupled with. If the device runs an older version of the mmWave SDK demo, then the Visualizer reports an error similar as shown in the following figure.

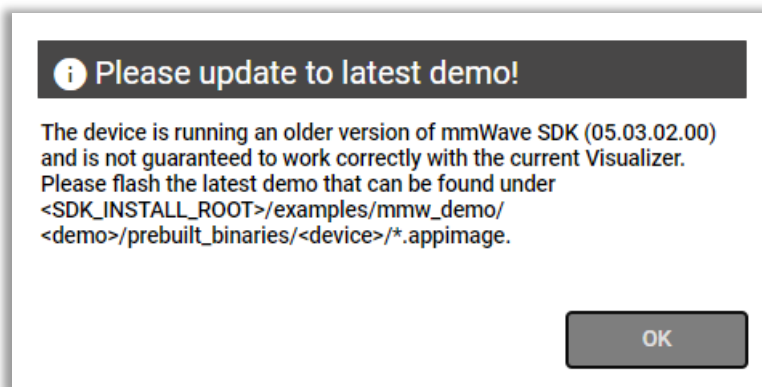


Figure 24: Version Mismatch Error

8.7. No Plots displayed / No Visualizer updates / Visualizer Freeze

If the Visualizer appears to be frozen i.e., there are no updates seen, try switching to a different tab and back to see if the updates resume. If after configuring the device from the Configuration Dashboard, no plot is displayed, then try re-starting the Visualizer application and re-configure the device. Some of these issues are also related to

the way different browsers handle tabs, when they halt and optimize tabs running without user interaction for a long time.

9. Limitations

If the device is connected after starting the Visualizer application, the `COM Port` may have to be selected / refreshed manually.

Quick Evaluation currently works with `Baudrate` set to 115200.

Generate Configuration File supports presence / motion detection and not object tracking.