



Getting Started Guide

mmWave Diagnostic & Monitoring TI Design

This demo is a reference application to showcase the usage of device's inbuilt Diagnostic & Monitoring feature using mmWave SafeTI Diagnostic Library (SDL) and RadarSS respectively. This demo assists in the development of software applications involving functional safety. Here is the list of features this demo provides

- Secondary Bootloader (SBL) runs few of destructive diagnostic tests.
- Main application implements remaining diagnostic test and device monitor features.
- TI-RTOS based MSS implementation of diagnostic tests.
- Non-OS based DSS implementation of diagnostic tests.
- All the test status are reported over UART.
- MSS runs periodic diagnostic tests.

Requirements

1. Hardware and Software Requirements

Item	Details
Device	AWR1843BOOST , AWR6843ISK , IWR6843ISK .
Computer	PC with Windows 7 or 10. If a laptop is used, please use the 'High Performance' power plan in Windows.
Micro USB Cable	Micro USB cable for mmwave sensor.
Power Supply	5V, >3.0A with 2.1-mm barrel jack (center positive).

Software

Tool	Version	Required For	Download Link
TI mmWave SDL	1.0.0.x	Diagnostic library for different device variants.	Link For access to SDL please contact your local TI sales representative to get a Safety NDA signed with TI.
TI mmWave SDK	3.5.0.x	Compilation of device application for different device variants.	TI mmWave SDK 3.5.0.x and all the related tools are required to be installed as specified in the mmWave SDK release notes
Uniflash	Latest	Quickstart Firmware	Download offline tool or use cloud version

Quickstart

The quickstart guide will cover setting up the EVM, flashing firmware, and running the demo.

1. Setup the EVM for Flashing Mode

- For AWR1843: follow the instructions for [Hardware Setup of xWRXXXBOOST for Flashing Mode](#)
- For xWR6843ISK in Standalone/Modular Mode: Follow the instructions for [Hardware Setup of xWR6843ISK/ODS for Flashing Mode](#)

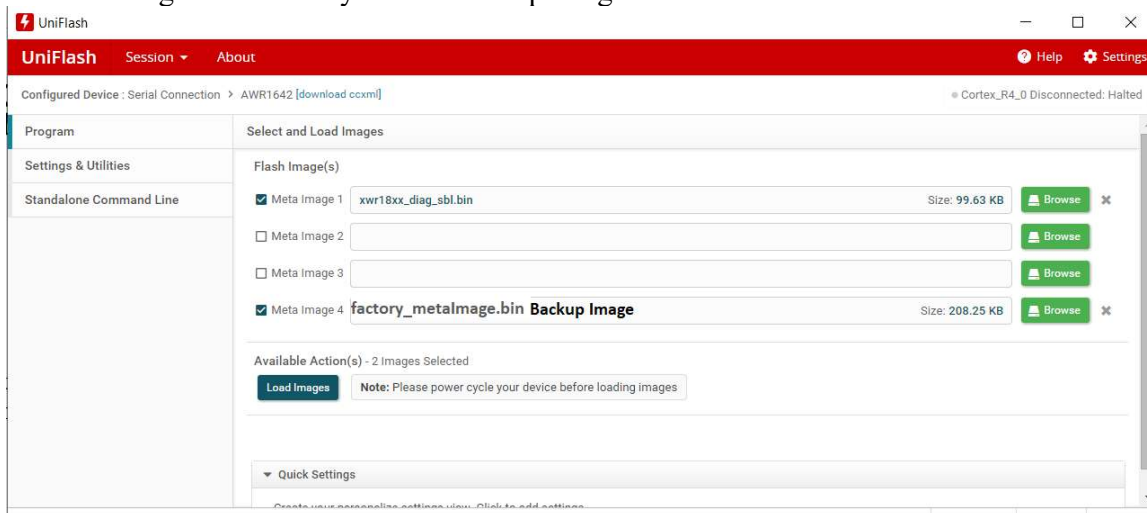
2. Flash the EVM using Uniflash

This reference TI design contain SBL as well as Main application, user needs to flash both type of binary files.

Flash the SBL binary (<device>_diag_sbl*.bin) associated with the EVM platform listed below using UniFlash. Follow the instructions for [using UniFlash](#)

Platform	BIN Name	Location
18xx	xwr18xx_diag_sbl.bin	diagnostic_monitor_ref\ti\demo\diag_mon_ref\prebuilt_binaries
	xwr18xx_diag_mon_demo.bin	diagnostic_monitor_ref\ti\utils\diag_sbl\prebuilt_binaries
68xx	xwr68xx_sbl.bin	diagnostic_monitor_ref\ti\demo\diag_mon_ref\prebuilt_binaries
	xwr68xx_diag_mon_demo.bin	diagnostic_monitor_ref\ti\utils\diag_sbl\prebuilt_binaries

- Load the SBL image and a factory default backup image as shown below



- Find the COM port which shows as “XDS110 Class Application/User UART” under device manager. That COM no. needs to be set in ‘Setting & Utilities’ option of UniFlash.

Note: Same xwr68xx_diag_sbl.bin & xwr68xx_diag_mon.bin binary files for both of IWR6843 and AWR6843 devices.

3. Setup the EVM for Functional Mode

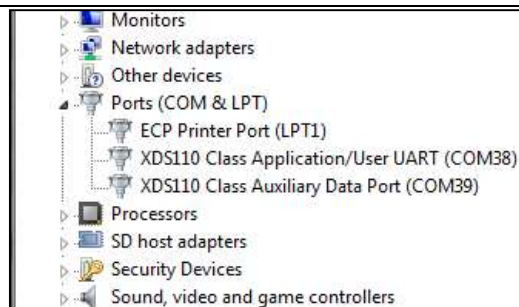
- For xWR6843ISK in Standalone/Modular Mode: Follow the instructions for [Hardware Setup of IWR6843ISK/ODS for Functional Mode](#)
- For xWR1843: Follow the instructions for [Hardware Setup of IWRXXXXBOOST for Functional Mode](#)
- For MMWAVEICBOOST + Antenna Module setup: Follow the instructions for [Hardware Setup of MMWAVEICBOOST + Antenna Module for Functional Mode](#)

At this point, the EVM should be powered, connected to the PC, flashed with the demo, and put in functional mode. The hardware setup is now complete.

4. Execute the application

By this time SBL metaImage is already flashed to the EVM. Follow the below steps to load the main image.

- User COM port number which is enumerated as 'XDS110 Class Application/User UART' in the device Manager.



1. Connect to the COM port found in previous step (using any serial terminal application like TeraTerm, Hyperterminal etc.) and configure the port (@baudrate 115200).
2. On bootup, SBL performs PBIST and STC diagnostic test on MSS and DSS core, then proceed for main application load. It will print the following message prompting the user to press CR or SPACE to stop the autoboot process and load a new application meta image over UART.

```
COMS - Tera Term VT
File Edit Setup Control Window Help
Debug: Testing MSS PBIST
Diag R4F PBIST (MSS PBIST ROM) Passed
Diag R4F PBIST (MSS STC ROM) Passed
Diag R4F PBIST (SU BUFFER) Passed
Diag R4F PBIST (DCAM Memory SRAM) Passed
Diag R4F PBIST (DCAM Memory FRAM) Passed
Diag R4F PBIST (DMA RAM) Passed
Diag R4F PBIST (MISPIA RAM) Passed
Diag R4F PBIST (MISPIB RAM) Passed
Diag R4F PBIST (DIME RAM) Passed
Diag R4F PBIST (Secure RAM) Passed
Diag R4F PBIST (TraceBuffer RAM) Passed
Diag R4F PBIST (Mailbox) Passed
Diag R4F PBIST (L3 Bank 0&1) Passed
Diag R4F PBIST (L3 Bank 2&3) Passed
Diag R4F PBIST (L3 Bank 4) Passed
Diag R4F PBIST (L3 Bank 5) Passed
Diag R4F PBIST (L3 Bank 6) Passed
Diag R4F PBIST (L3 Bank 7) Passed
Diag R4F PBIST (DSS ADCBUF and CQ) Passed
Diag R4F PBIST (DSS FFT) Passed
Diag R4F PBIST (DSS IPCC) Passed
Diag R4F PBIST (DSS Data T2FP RAM & HSRAM) Passed
Diag R4F PBIST (DSS CBUFF FIFO ECC) Passed
Diag R4F PBIST (DSS CBUFF FIFO) Passed
Diag R4F PBIST (DSS L2) Passed
Diag R4F PBIST (DSS PROGFIL) Passed
Diag R4F PBIST (DSS BFM) Passed
Diag R4F PBIST (DSS GEN PBIST ROM) Passed
Diag R4F PBIST (DSS STC ROM) Passed
Diag R4F PBIST (Fault injection) Passed
Diag R4F PBIST (All test) Passed
Debug: Testing MSS STC...
Diag R4F STC (Test type: Normal) Passed

=====
Debug: Secondary Bootloader Application Start
Press CR key or Space key to stop auto boot and Update Meta Image...
Press 'a' to skip this counter and load Main App...
Loading existing Meta Image from Flash in '90' 87 88 87 86 85 84 83 82 81
```

3. Press space key to stop the autoboot process. You will see the following prints. The flash at SBL_METAIMAGE_OFFSET address will be erased.

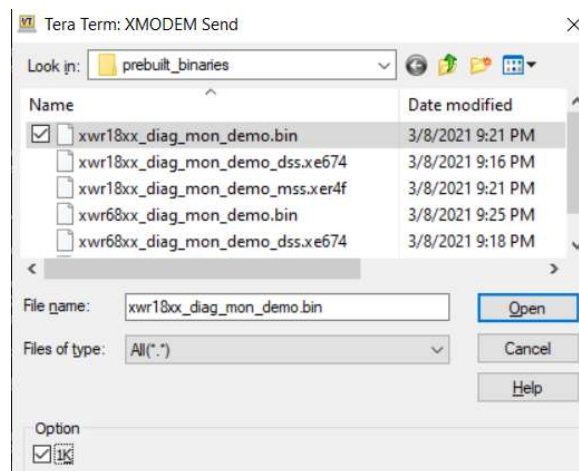
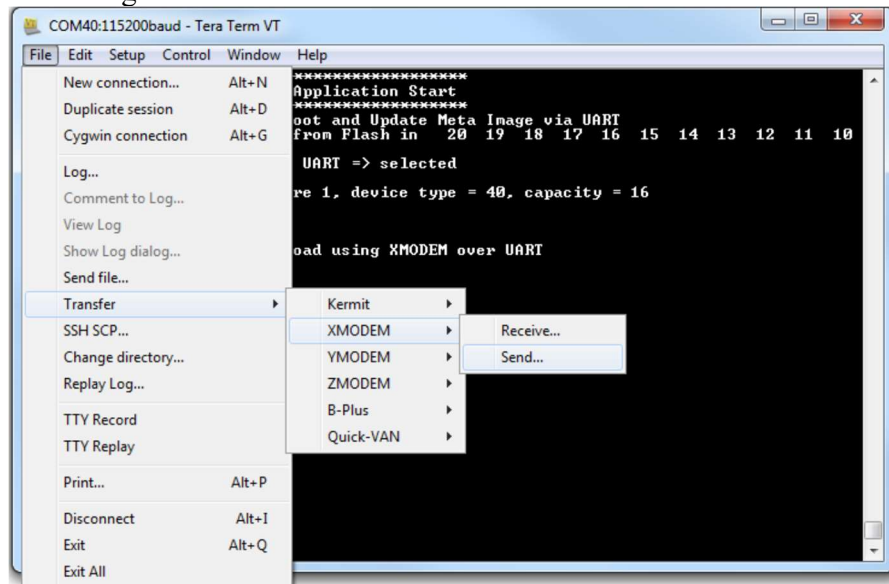
```

*****
Debug: Secondary Bootloader Application Start
*****
Press CR key or Space key to stop auto boot and Update Meta Image...
Press 'a' to skip this counter and load Main App...
Loading existing Meta Image from Flash in  90 89 88 87
Debug: Update Meta Image selected

qspiClk 198000000Debug: Device info: Manufacturer: 1, Device type = 40, Capacity =
16
Debug: Loading application metaImage from Flash address: c0040000
Debug: Erasing SFlash...
Debug: Flash Erase complete
Debug: Start the image download using XMODEM over UART
CCCCC

```

4. Start the file download using XMODEM via the Teraterm



5. Once the image is downloaded, SBL verifies the image and loads from SFLASH to RAM.

Note: If the application meta image boot fails for any reason, the backup image is loaded from SFLASH to RAM. If both the images fail to load, SBL resets the board and goes back to original step.

```

COM5 - Tera Term VT
File Edit Setup Control Window Help
Debug: Start the image download using XMODEM over UART
CCDDebug: Total data written = 0x34180
Debug: Parsing completed
*****
mmWave Safety Diagnostic Library Demo
*****
[SUCCESS] Diag VIM ECC 1-bit Error
[SUCCESS] Diag VIM ECC 2-bit Error
[SUCCESS] Diag Mailbox ECC 1-bit Error [MSS to BSS]
[SUCCESS] Diag Mailbox ECC 2-bit Error [MSS to BSS]
[SUCCESS] Diag Mailbox ECC 1-bit Error [BSS to MSS]
[SUCCESS] Diag Mailbox ECC 2-bit Error [BSS to MSS]
[SUCCESS] Diag Mailbox ECC 1-bit Error [DSS to MSS]
[SUCCESS] Diag Mailbox ECC 2-bit Error [DSS to MSS]
[SUCCESS] Diag Mailbox ECC 1-bit Error [MSS to DSS]
[SUCCESS] Diag Mailbox ECC 2-bit Error [MSS to DSS]
[SUCCESS] Diag Mailbox ECC 1-bit Error [BSS to DSS]
[SUCCESS] Diag Mailbox ECC 2-bit Error [BSS to DSS]
[SUCCESS] Diag Mailbox ECC 1-bit Error [o=o=o=o=o=o=oa$]
[SUCCESS] Diag Mailbox ECC 2-bit Error [o=o=o=o=o=o=oa$]
[SUCCESS] Diag TCMA 1 Bit ECC
[SUCCESS] Diag TCMA Parity
[SUCCESS] Diag TCMB0 Parity
[SUCCESS] Diag TCMB1 Parity
[SUCCESS] Diag DMA-0 MPU
[SUCCESS] Diag DMA-1 MPU
[SUCCESS] Diag DMA-0 Parity
[SUCCESS] Diag DMA-1 Parity
[SUCCESS] MIBSPI-0 ECC Diagnostic TX RAM, Buffer Index[10]
[SUCCESS] MIBSPI-0 ECC 1-Bit Diagnostic RX RAM, Buffer Index[10]
[SUCCESS] MIBSPI-0 ECC Diagnostic TX RAM, Buffer Index[10]
[SUCCESS] MIBSPI-0 ECC 2-Bit Diagnostic RX RAM, Buffer Index[10]
[SUCCESS] MIBSPI-1 ECC Diagnostic TX RAM, Buffer Index[10]
[SUCCESS] MIBSPI-1 ECC 1-Bit Diagnostic RX RAM, Buffer Index[10]
[SUCCESS] MIBSPI-1 ECC Diagnostic TX RAM, Buffer Index[10]
[SUCCESS] MIBSPI-1 ECC 2-Bit Diagnostic RX RAM, Buffer Index[10]
[SUCCESS] Diag Watchdog Test
[SUCCESS] Diag R4F CCM-A Test
[SUCCESS] Diag R4F CCM-A Test
[SUCCESS] Diag R4F CCM-B Test
[SUCCESS] Diag R4F CCM-B Test
[SUCCESS] Diag R4F CCM-A Negative Test
[SUCCESS] Diag R4F CCM-A Negative Test
[SUCCESS] Diag R4F CCM-B Negative Test
[SUCCESS] Diag R4F CCM-B Negative Test
[SUCCESS] Diag ESM Static Configuration
[SUCCESS] Diag VIM StaticCfg Configuration
[SUCCESS] Diag DMA StaticCfg Configuration
[SUCCESS] Diag DMA StaticCfg Configuration
[SUCCESS] Diag MibSPI-0 StaticCfg Configuration
[SUCCESS] Diag MibSPI-1 StaticCfg Configuration
[SUCCESS] Diag DCAN StaticCfg Configuration
[SUCCESS] Diag MCAN StaticCfg Configuration
[SUCCESS] Diag RCM StaticCfg Configuration
[SUCCESS] Diag R4F StaticCfg Configuration
[SUCCESS] Diag RTI Static Config
[SUCCESS] Diag DMA Static Configuration Verify MPU
[SUCCESS] Diag DMA Static Configuration Verify MPU
[SUCCESS] Diag ESM StaticCfg Verify Violation LTC Preload

```

- If main application is flashed already then user can hit 'a' to skip the autoboot counter and SBL will load that from sFlash and it will be executed.
- Main application will run the miscellaneous Diagnostic test (MSS/DSS) and enable the monitoring feature of the device. On the serial terminal screen, you would see status of those tests.

```

COM5 - Tera Term VT
File Edit Setup Control Window Help
[SUCCESS] Diag MCAN Static Configuration Verify Config
[SUCCESS] Diag RCM StaticCfg Verify Violation Mailbox ECC
[SUCCESS] Diag R4F StaticCfg Verify Violation TCMB1 ext error status
*** DSS Diagnostic Status: ***
[SUCCESS] DSS EDMA IPCC0 PARITY Diagnostic.
[SUCCESS] DSS EDMA IPCC1 PARITY Diagnostic.
[SUCCESS] DSS LIP PARITY Diagnostic.
[SUCCESS] DSS L2P UMAP0 PARITY Diagnostic.
[SUCCESS] DSS L2P UMAP1 PARITY Diagnostic.
[SUCCESS] DSS L2 ECC 1Bit Error Diagnostic.
[SUCCESS] DSS L2 ECC 2Bit Error Diagnostic.
[SUCCESS] DSS L3 ECC 1Bit Error Diagnostic.
[SUCCESS] DSS L3 ECC 2Bit Error Diagnostic.
[SUCCESS] DSS TXFR RAM ECC 1Bit Error Diagnostic.
[SUCCESS] DSS TXFR RAM ECC 2Bit Error Diagnostic.
[SUCCESS] DSS HSRAM ECC 1Bit Error Diagnostic.
[SUCCESS] DSS HSRAM ECC 2Bit Error Diagnostic.
[SUCCESS] DSS HWA ECC 1Bit Error Diagnostic.
[SUCCESS] DSS HWA ECC 2Bit Error Diagnostic.
[SUCCESS] DSS HWA Lockstep Error Diagnostic.
*** Frame Triggered... ***
*** PERIODIC Static Diagnostic Test ***
[SUCCESS] Diag ESM Static Configuration
[SUCCESS] Diag VIM StaticCfg Configuration
[SUCCESS] Diag DMA StaticCfg Configuration
[SUCCESS] Diag RCM StaticCfg Configuration
[SUCCESS] Diag R4F StaticCfg Configuration
[SUCCESS] Diag RTI Static Config
*** PERIODIC Static Diagnostic Test ***

```

- After MSS and DSS boot time Diagnostic Injection test, MSS configures device for misc monitor features and print its failure state if any.
- At every FTTI interval it checks for any failure monitoring report, executes set of static diagnostic test and prints the status over UART.

Appendix

- Refer Release Notes for limitation and known issues.
- Refer Developer's Guide for in depth detail of this application.