TIDEP-0091 – Level Sensing Reference Design

Getting Started Guide – Software & Hardware

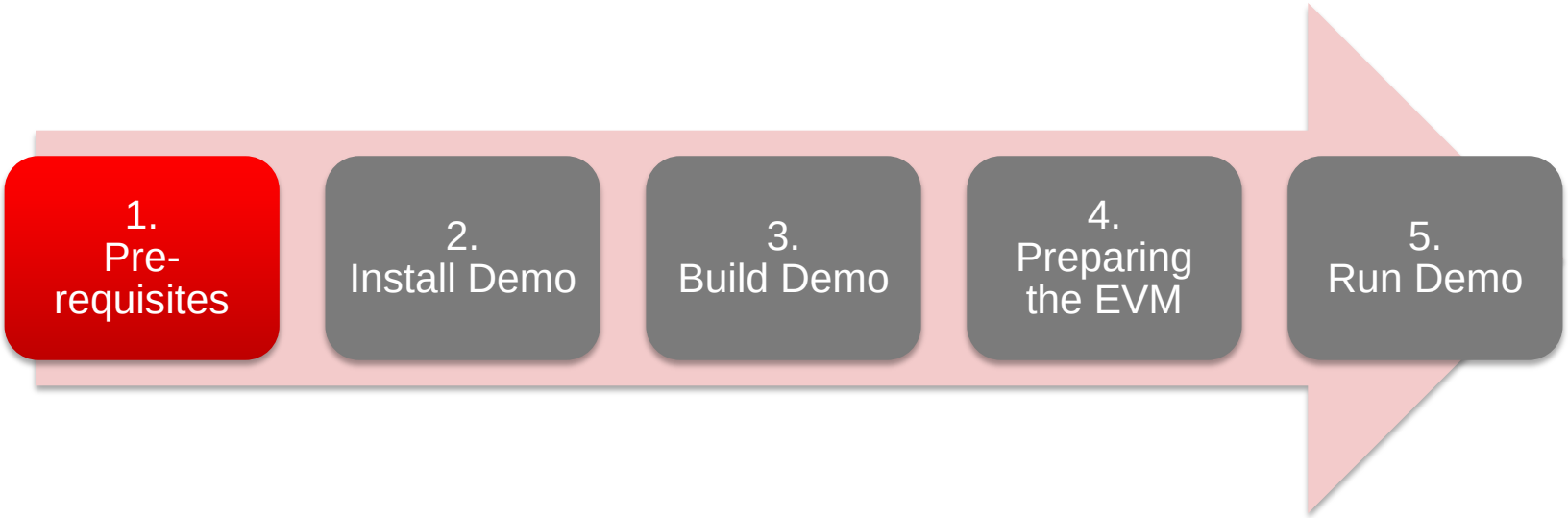
Software Requirements

- Software
  - **Pre-requisites**
    - mmWave-SDK2.1 and all related dependencies installed as mentioned in the mmWave SDK release notes, and installed by the SDK installer.
  - mmWave Level Sensing Demo Application
    - Download from the TI Design Page
  - UniFlash
    - For flashing firmware images onto
    - Download from TI.com/tool/uniflash
  - CCS 8.1 or newer, with the following components installed:
    - mmWave Sensors
    - MSP430 ultra-low power MCUs
    - TI XDS110 Drivers
    - Important Notes:
      - The CCS projects will not install without these components!
      - To add components, re-run the CCS installer and make sure all the components are checked. If they are already checked, you do not need to continue with the install update.
  - MSP432 SimpleLink 1.40

Hardware Requirements

- Hardware
  - IWR1443 EVM
    - IWR1443 EVM
  - MSP432P401R EVM
    - MSP432P401R EVM
  - Micro USB cable (included in the EVM packages)
  - 5V/5A Power Supply
    - Purchase from Digikey
- Hardware Modifications
  - LP87524J-Q1 PMIC
    - LP87524J-Q1
  - Three Ferrite Beads of .18 uH, 120 oh at 100MHz
    - Recommend Murata BLM18KG121TH1
  - Two-pin Header
  - Thin wire

Steps

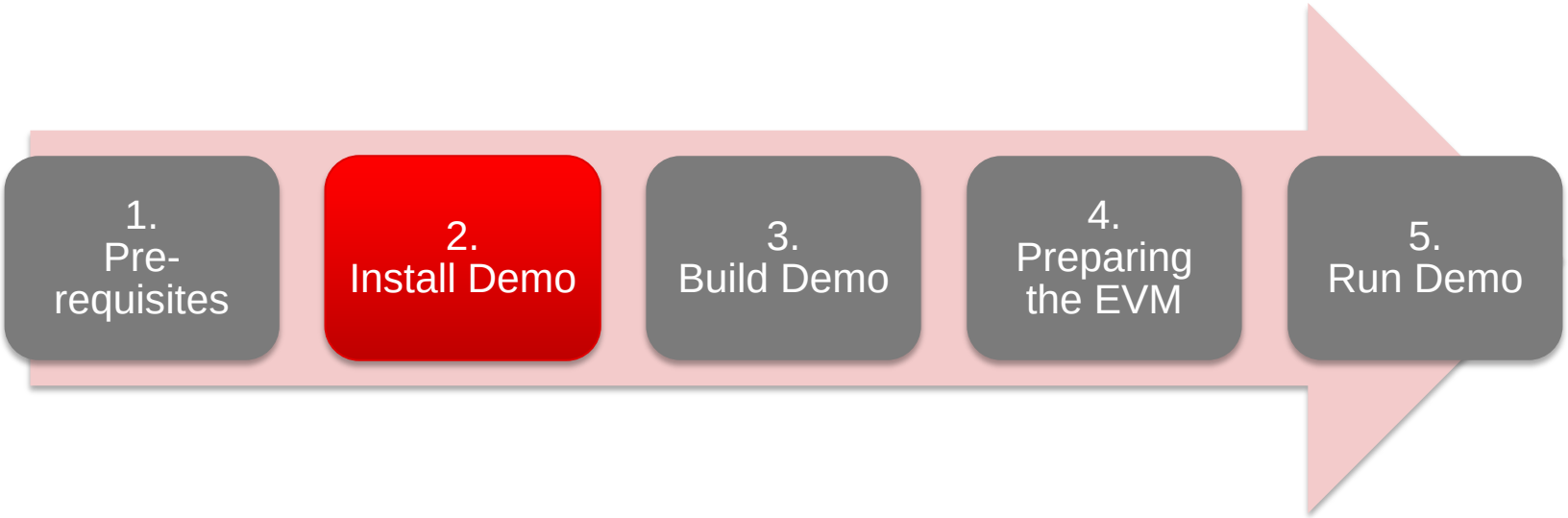

1.
Pre-
requisites
2.
Install Demo
3.
Build Demo
4.
Preparing
the EVM
5.
Run Demo

1. Pre-requisites

- It is assumed that you have the mmWave SDK 2.1 and all the related tools installed as mentioned in the mmWave SDK release notes.
  - The mmWave SDK installer now includes the required TI components and installs them automatically.
- For the MSP432, you need to install MSP432 SimpleLink 1.40 SDK (Please note that newer versions are not compatible with this demo). It is recommended to install all TI components in the default C:\ti folder.
- 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 demo onto your machine.

Tool	Version	Download Link
mmWave SDK	2.1.0.4	download link
CCS	8.1.0	download link
TI SYS/BIOS	6.53.02.00	Included in mmwave sdk installer
TI ARM Compiler	16.9.6.LTS	Included in mmwave sdk installer
TI CGT Compiler	8.1.3	Included in mmwave sdk installer
XDC	3.50.04.43	Included in mmwave sdk installer
C64x+ DSPLIB	3.4.0.0	Included in mmwave sdk installer
C674x DSPLIB	3.4.0.0	Included in mmwave sdk installer
C674x MATHLIB (little-endian, elf/coff format)	3.1.2.1	Included in mmwave sdk installer
mmwave device support packages	1.5.9 or later	Upgrade to the latest using CCS update process (see SDK user guide for more details)
TI Emulators package	7.0.188.0 or later	Upgrade to the latest using CCS update process (see SDK user guide for more details)
Uniflash	latest	Uniflash tool is used for flashing xWR1xxx devices Cloud version (Recommended): https://dev.ti.com/uniflash Offline version: http://www.ti.com/tool/uniflash
mmWave Demo Visualizer	latest	TI Gallery APP for configuring mmWave sensors and visualizing the point cloud objects generated by the mmWave SDK demo https://dev.ti.com/mmWaveDemoVisualizer

Steps


1.
Pre-
requisites
2.
Install Demo
3.
Build Demo
4.
Preparing
the EVM
5.
Run Demo

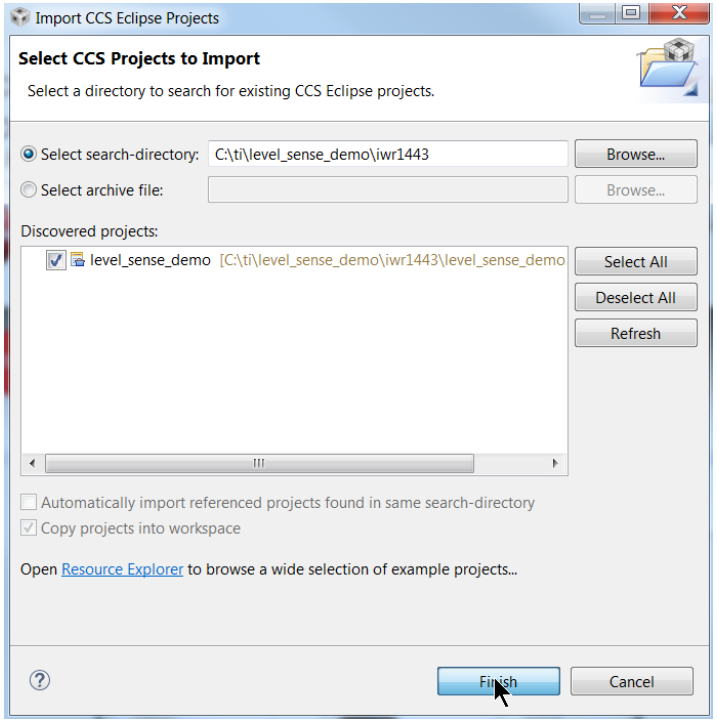
2. Install Demo

- Download the mmWave Level Sensing demo software package (zip file) to a convenient working space on your PC.
  - Download locations:
    - Level Sensing TI Design
    - Level Sensing Software Package

- Unzipping the software package yields three folders:
  - /docs: The location of this Getting Started Guide, Release Notes, and other docs.
  - /iwr1443: The folder containing source code for the IWR1443 level sensing executable and CCS project.
  - /msp432: The folder containing source code for the MSP432 level sensing executable and CCS project.

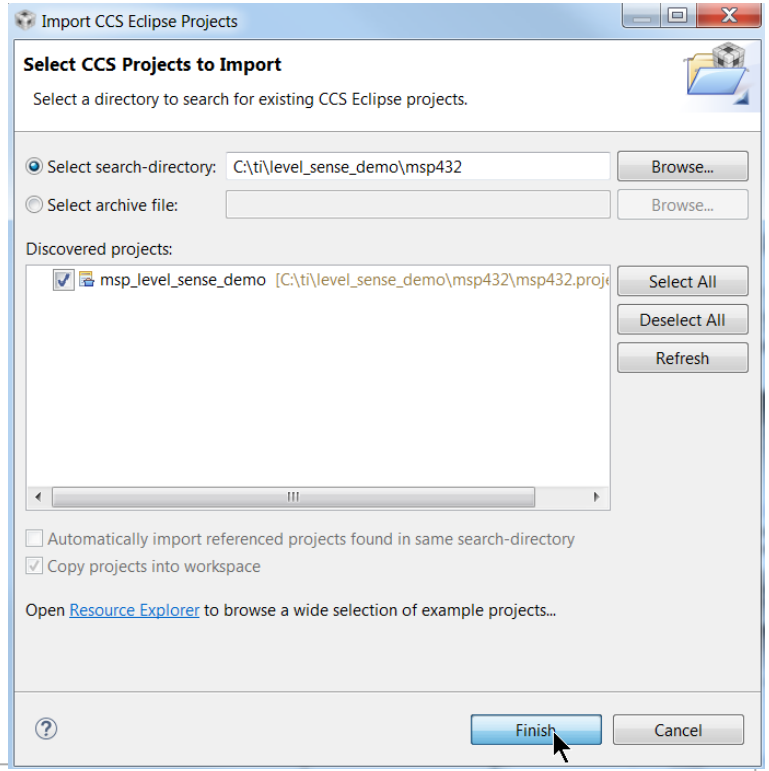
2. Install Demo - continued

- Create the IWR1443 CCS Project
  - Start CCS 8.1 (or newer) and click on Project -> Import CCS Project.
  - Navigate the search-directory Browser to <install path>\level_sense_demo\iwr1443
  - The window should look like the image to the right.
  - Click Finish.
  - Note: By default, CCS will copy the source code into a new folder in your CCS Workspace. Editing source in your install folder will not get compiled.

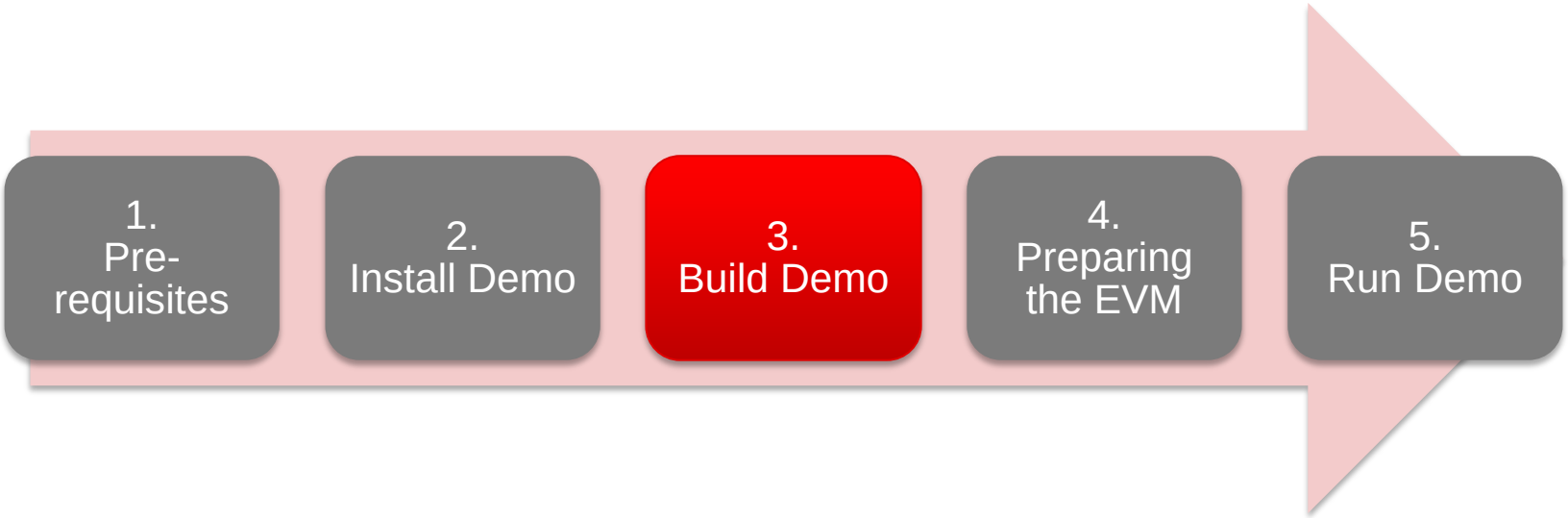


2. Install Demo - continued

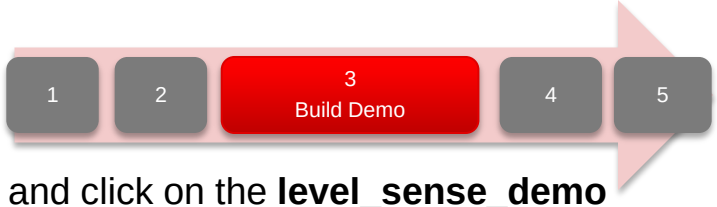
- Create the MSP432 CCS Project
  - Make sure MSP432 SimpleLink 1.40 is installed first.
  - If updating from an older version of Simplelink, please delete any existing msp_level_sense_demo and tirtos_builds_MSP432P401R_debug_ccs projects (deleting from disk – save your code updates!) and re-import the lsdemo_msp432_m4f CCS project.
  - Start CCS 8.1 (or newer) and click on Project -> Import CCS Project.
  - Navigate the search-directory Browser to <install path>\level_sense_demo\msp432
  - The window should look like the image to the right.
  - Click Finish.
  - Note: By default, CCS will copy the source code into a new folder in your CCS Workspace. Editing source in your install folder will not get compiled.



Steps


1.
Pre-
requisites
2.
Install Demo
3.
Build Demo
4.
Preparing
the EVM
5.
Run Demo

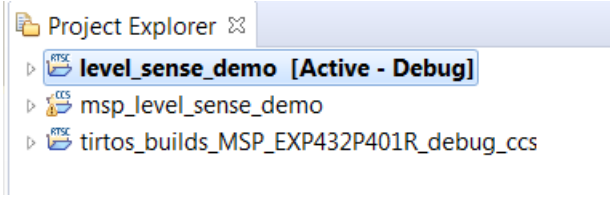
3. Build Demo



1. Building the Level Sense Demo executables is performed in CCS.

2. Note: The software package includes pre-built executables if you would like to run the demo without having to build. They are located here:
   - <install path>\level_sense_demo\iwr1443\Debug
   - <install path>\level_sense_demo\msp432\Debug
   - .bin files are for flashing, .xer4f and .out are for CCS debug.

3. Start CCS, and click on the **level_sense_demo** project as shown here:

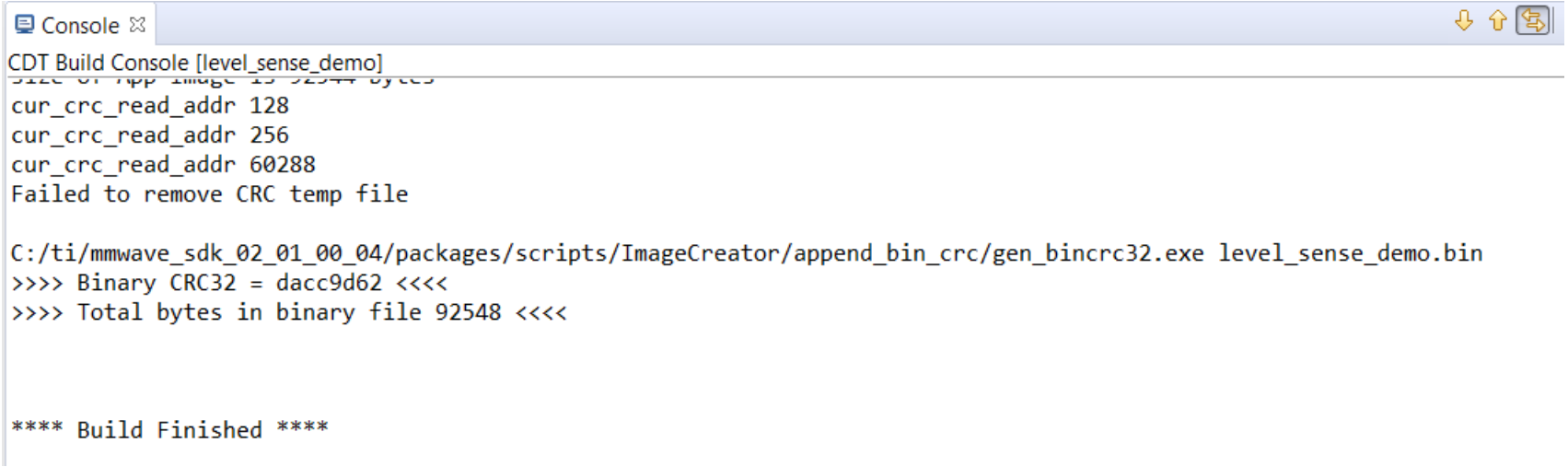


4. Click Project -> Build Project.

Note: The order of project building is not important. You must have both executables before you can flash and run the demo.

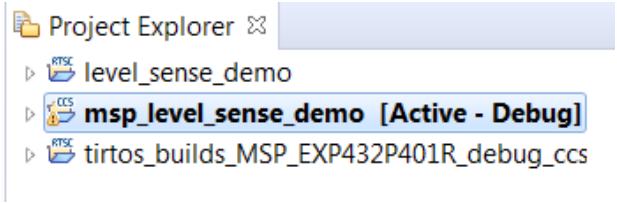
3. Build Demo - continued

6. Check the CCS Console for the results of the IWR1443 build:
7. The executables will be located here: <ccs workspace>\level_sense_demo\Debug\level_sense_demo.bin and .xer4f.



3. Build Demo - continued

7. In the CCS Project Explorer, click on the MSP432 project as shown to the right --->
8. Click Project -> Build Project.
9. Check the Console window for the build results (see below).
10. The executable will be located here: <ccs workspace>\msp_level_sense_demo\Debug\msp_level_sense_demo.out.

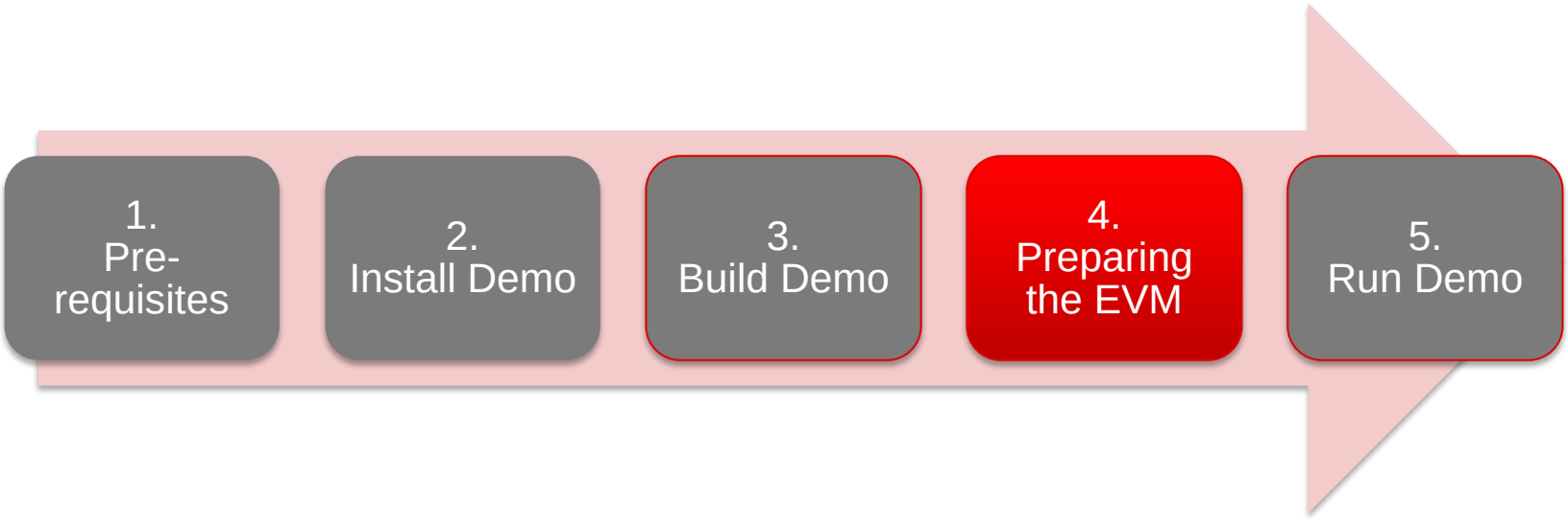


Console

CDT Build Console [msp_level_sense_demo]

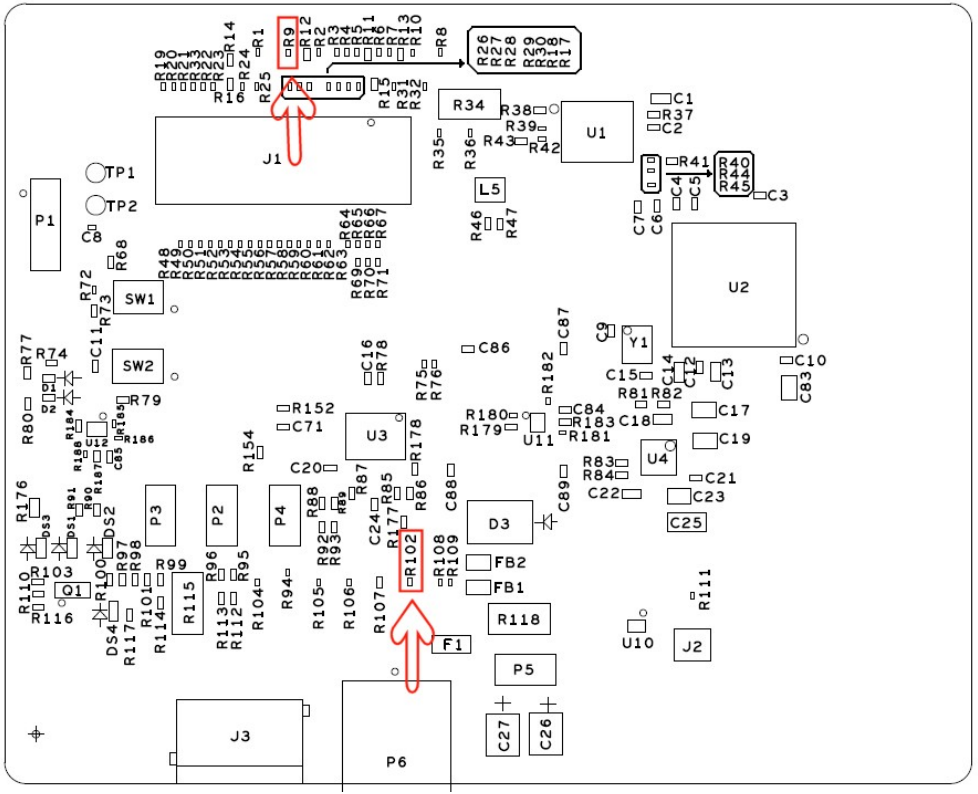
```
"./source/MSP_EXP432P401R.obj" "./source/msp_gpio.obj" "./source/msp_spi.obj" "./source/msp_timer.obj" "./source/msp_uart.obj"
"../MSP_EXP432P401R_TIRTOS.cmd" -l"C:/Users/A0227075/workspace_v8_LevelS359/tirtos_builds_MSP_EXP432P401R_debug_ccs/Debug/configPkg/linker.cmd"
-l"C:/ti/simplelink_msp432_sdk_1_40_01_00/source/ti/display/lib/display.aem4f"
-l"C:/ti/simplelink_msp432_sdk_1_40_01_00/source/ti/drivers/lib/drivers_msp432p4xx.aem4f"
-l"C:/ti/simplelink_msp432_sdk_1_40_01_00/kernel/tirtos/packages/ti/dpl/lib/dpl_msp432p4xx.aem4f"
-l"C:/ti/simplelink_msp432_sdk_1_40_01_00/source/third_party/fatfs/lib/fatfs.aem4f"
-l"C:/ti/simplelink_msp432_sdk_1_40_01_00/source/ti/devices/msp432p4xx/driverlib/ccs/msp432p4xx_driverlib.lib" -llibc.a
<Linking>
Finished building target: "msp_level_sense_demo.out"


**** Build Finished ****
```

Steps


1.
Pre-
requisites
2.
Install Demo
3.
Build Demo
4.
Preparing
the EVM
5.
Run Demo

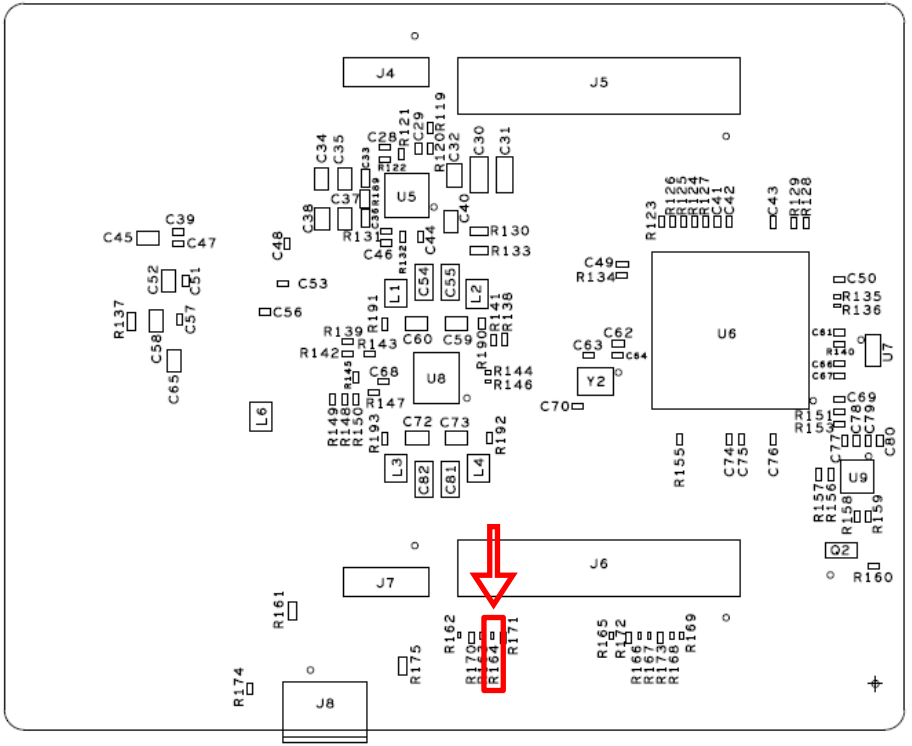
4.1 Preparing the EVM -- Mandatory

- Before we flash the boards, there are hardware modifications to be made. The first set of instructions are **Mandatory**. The second set are highly **Recommended** to achieve lowest possible power but are not required for functionality.
- First, zero ohm resistor R102 must be added on the front side of the IWR1443 EVM as shown. This is for PMIC_EN line.
- Second, zero ohm resistor R9 must be added as shown. This is for GPIO_1.



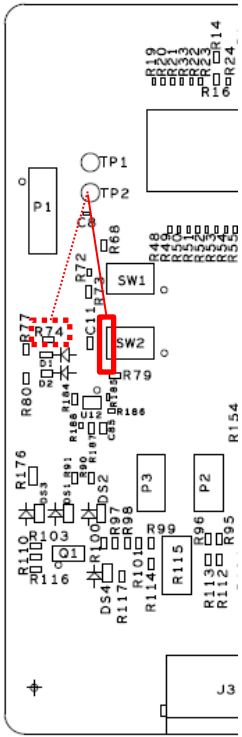
4.1 Preparing the EVM -- Mandatory

- Third, zero ohm resistor R164 must be added on the back side as shown. This is for GPIO_0.

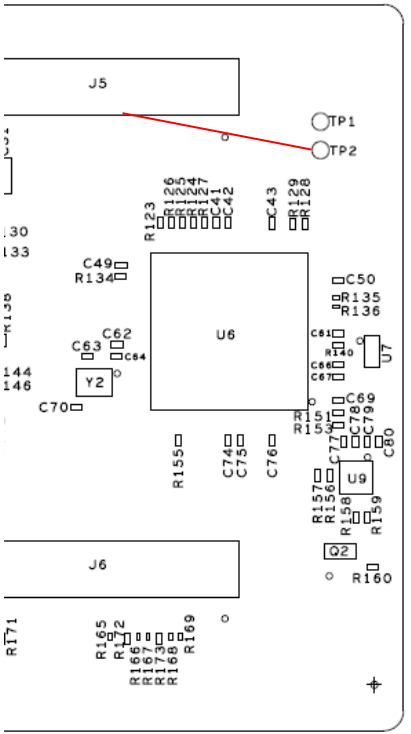


4.1 Preparing the EVM

- Next a wire must be added from either R74 or one of the bottom nodes of SW2 to the 7th slot of the J5 connector (J5-7) on the back side. (Pictures on next slide)
  - This wire should pass through hole TP2

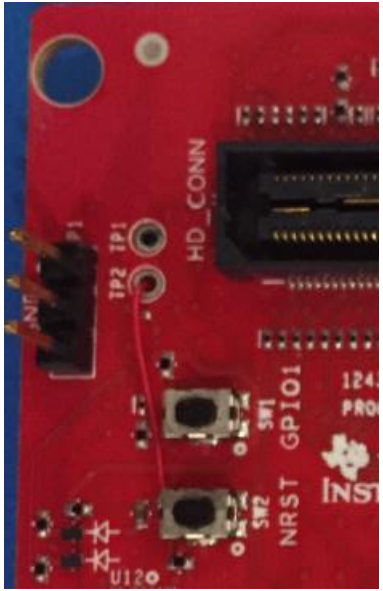


Front

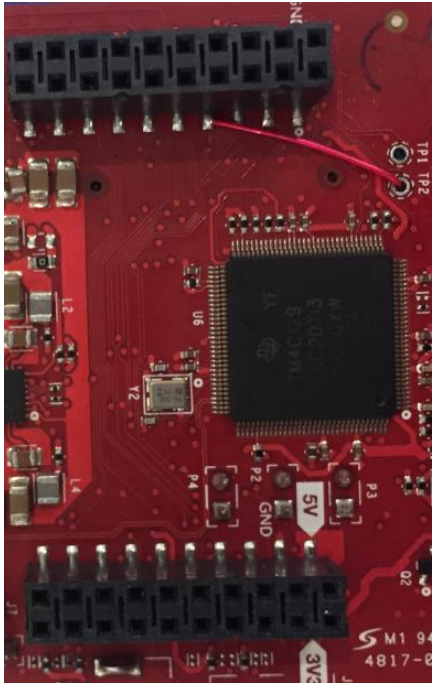


Back

4.1 Preparing the EVM -- Mandatory



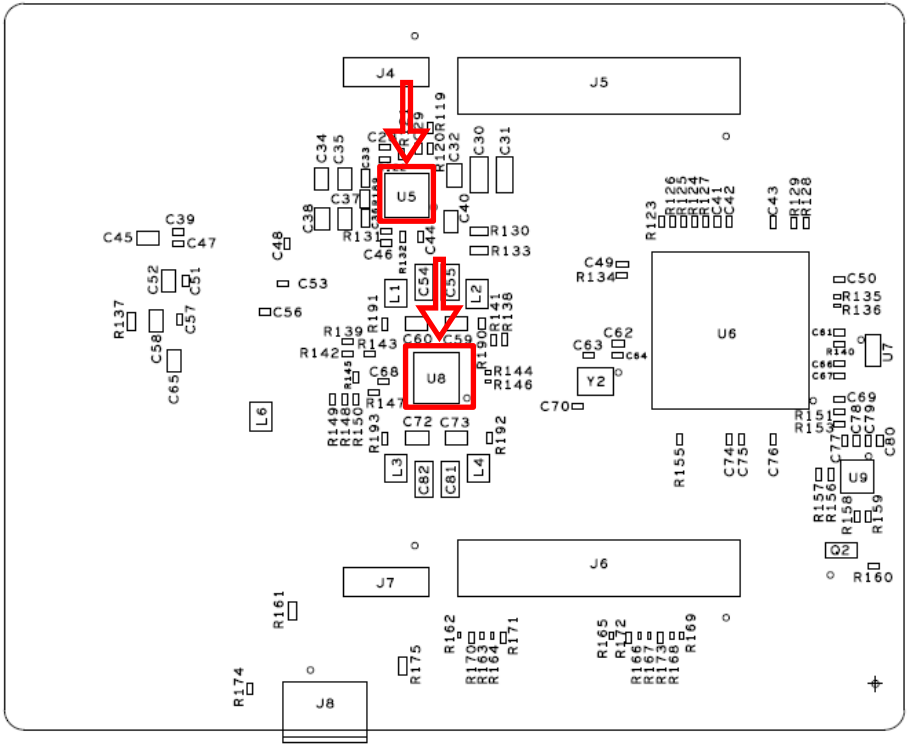
Front



Back

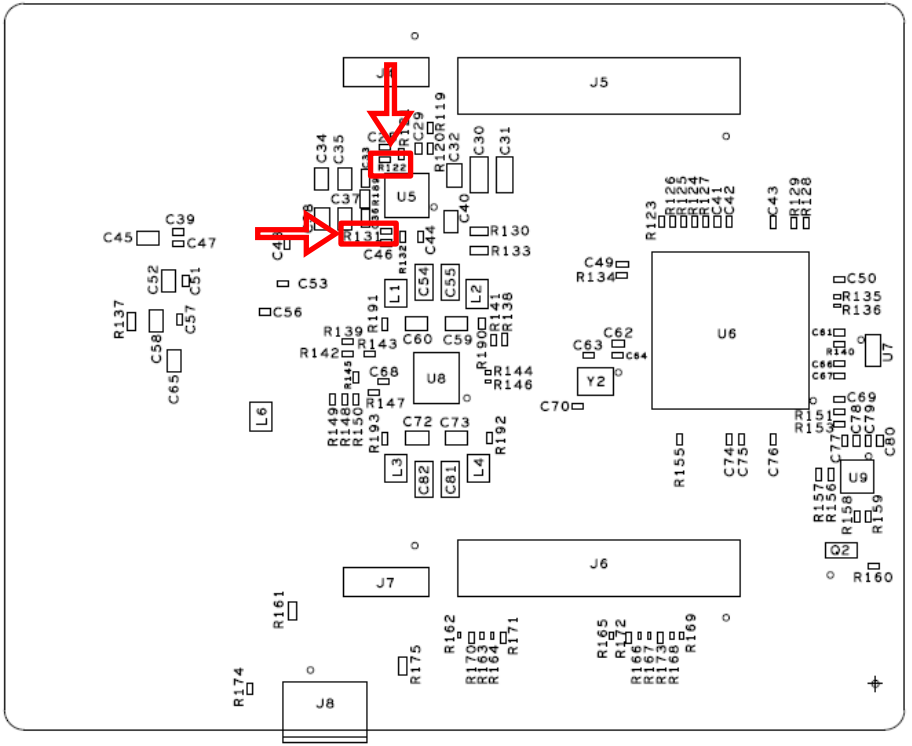
4.1 Preparing the EVM

- The following changes are not required for functionality but highly recommended for low-power.
- Replace U8 with LP87524J on back side of board. This is to implement a higher efficiency PMIC.
- Remove U5 from the board. This is a 1.8v ->1.3v LDO.



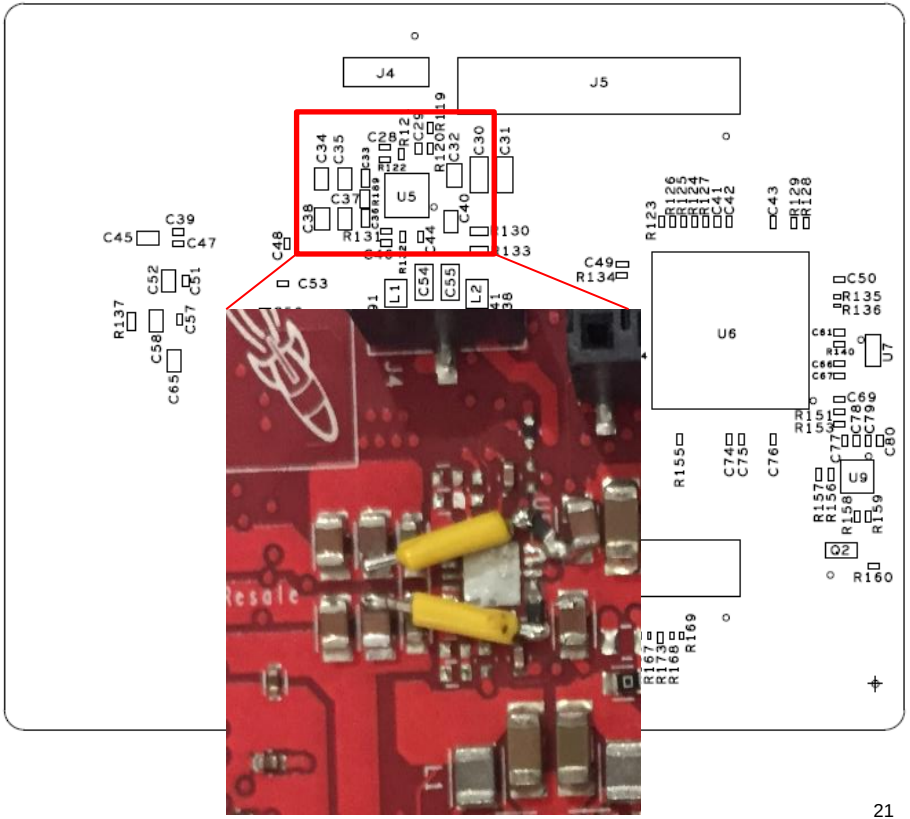
4.1 Preparing the EVM

- Remove resistor R122
- Remove resistor R131



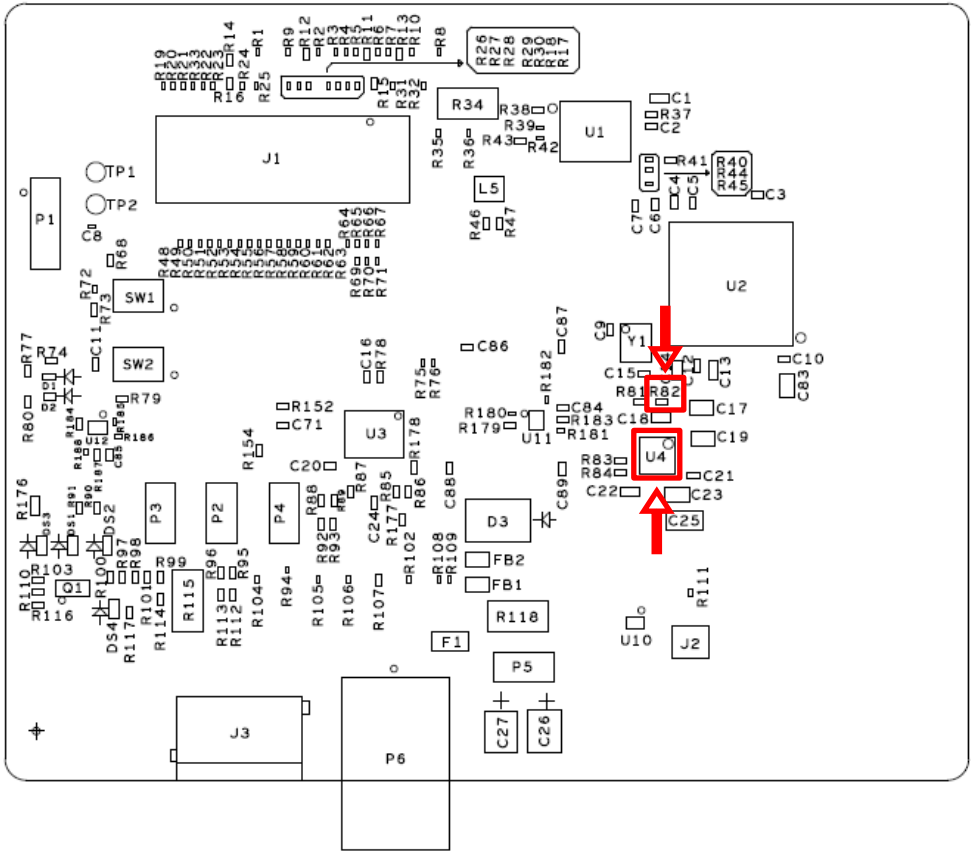
4.1 Preparing the EVM

- Install new inductor (ferrite bead) across U5 Node 14,15 to U5 node 1, 2.
- Install new inductor across U5 node 11,12 to U5 node 4,5.
- See picture for clarification.



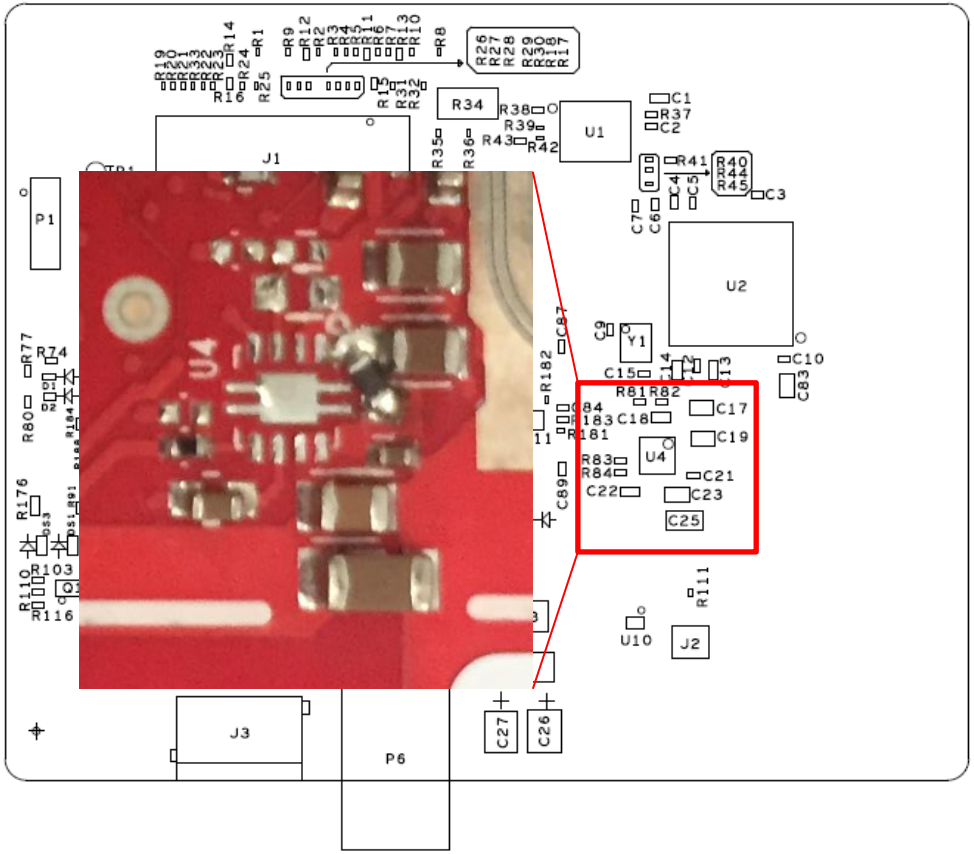
4.1 Preparing the EVM

- Remove U4, a 2.3v -> 1.8v LDO from front of EVM.
- Remove sense resistor R82.



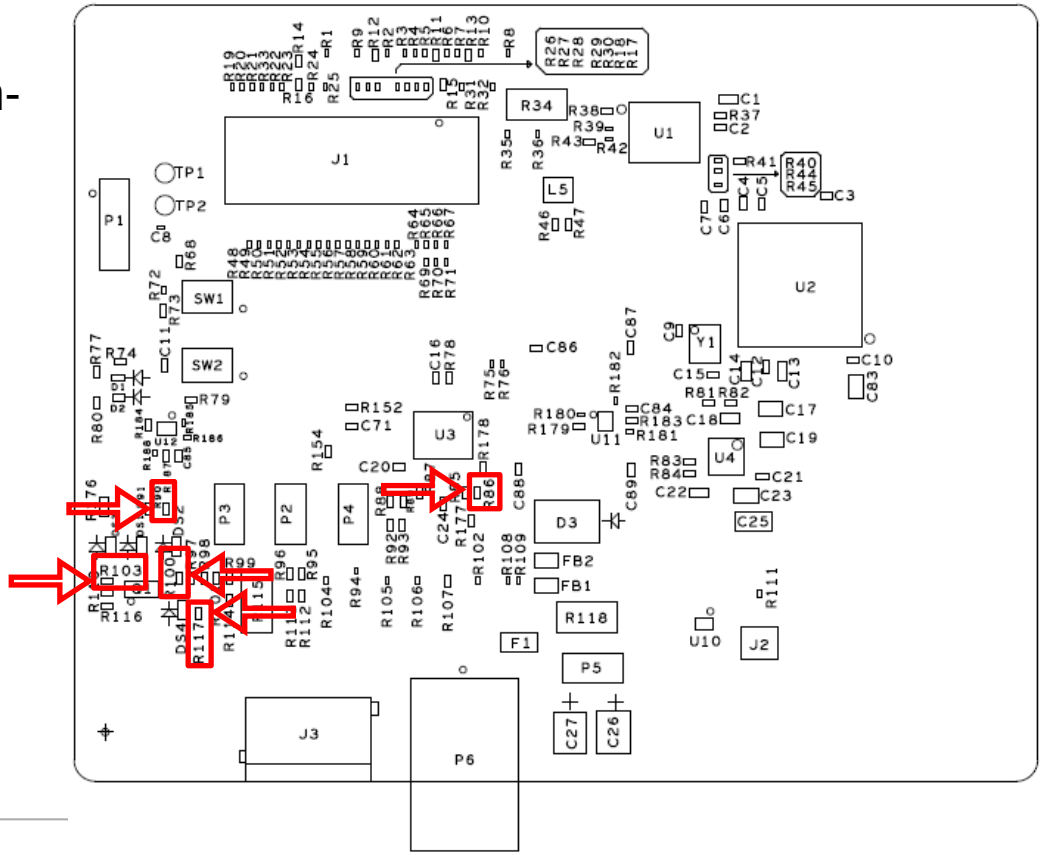
4.1 Preparing the EVM

- Add inductor (ferrite bead) across U4 node 7,8 to U4 node 1,2.
- See picture for clarification



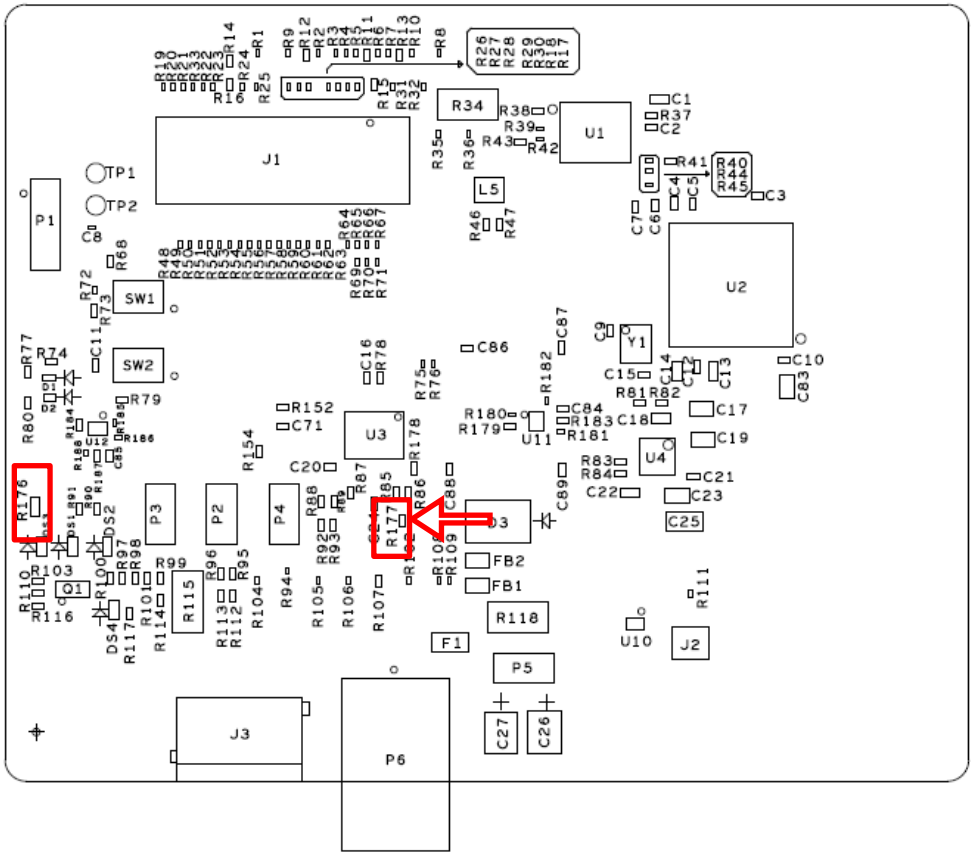
4.1 Preparing the EVM

- Remove resistors R90, R100, R103, R117. This removes power to the on-board LEDs.
- Remove resistor R86.



4.1 Preparing the EVM

- Install a 10K resistor at R177
- Remove R176 and glue/install a two-pin jumper as shown. This is to remove/apply power to the XDS110.
  - Note: This connection is fragile, strong glue or epoxy is recommended.

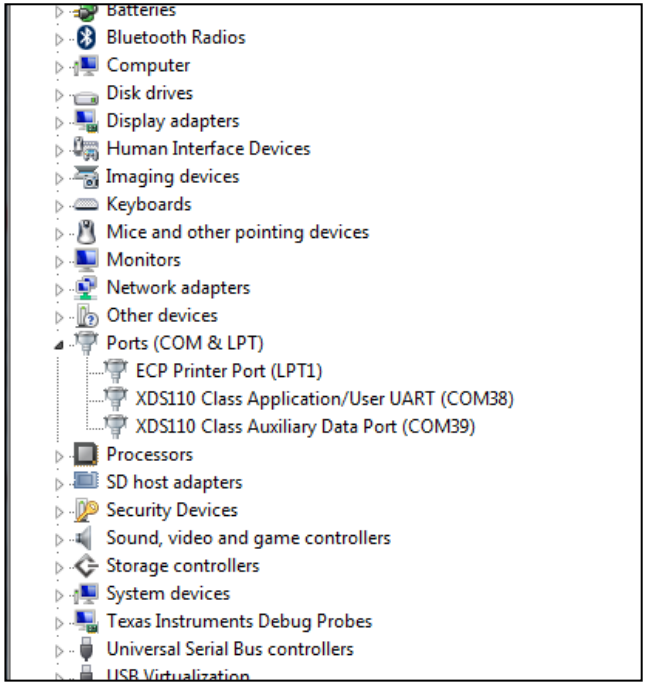


4.1 Preparing the EVM

- There are two ways to run the Level Sensing Demo:
  - Deployment mode: Requires flashing the level_sense_demo.bin to the IWR1443 EVM and msp_level_sense_demo.out to the MSP432 EVM. The demo is then run after attaching the two boards together (described in the next section). This is the mode that should be run when taking measurements for power consumption.
  - Debug mode: Requires flashing CCS debug firmware to the IWR1443 EVM, then loading level_sense_demo.xer4f via the CCS Debugger. This debug firmware image is provided with the mmWave SDK. The demo is then run without the MSP432 attached to the IWR1443.
  - Please note that the IWR1443 flashing process is the same for both modes, just using different files.

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 Application.

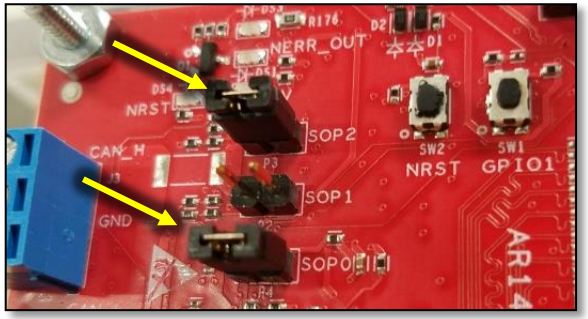


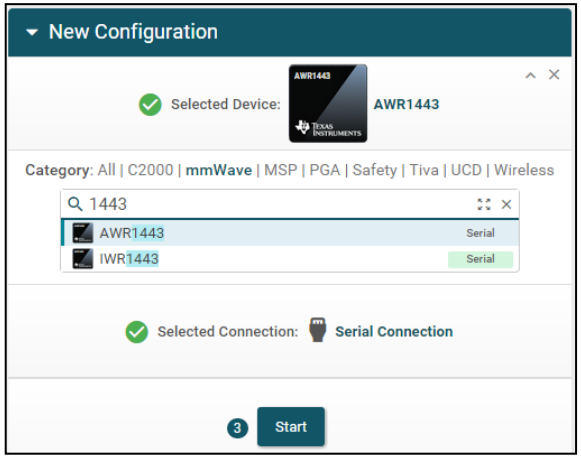
COM_{UART}: COM38 **COM_{AUX}**: COM39

- The actual port numbers on your machine may be different

4.3 Flashing CCS debug firmware

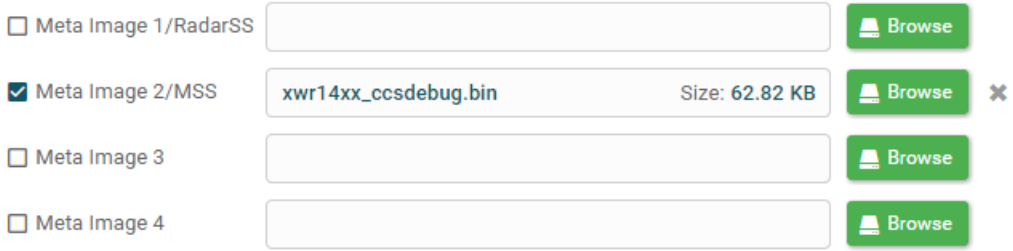
1. Put the EVM in flashing mode by connecting jumpers on SOP0 and SOP2 as shown in the image.
2. Open the **UniFlash** tool
3. In the **New Configuration** section, locate and select the IWR1443 device.
4. Click **Start** to proceed



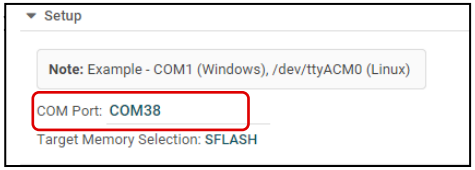


4.3 Flashing CCS debug firmware

1. In the **Program** tab, browse and locate the **xwr14xx_ccsdebug.bin** image shown below:



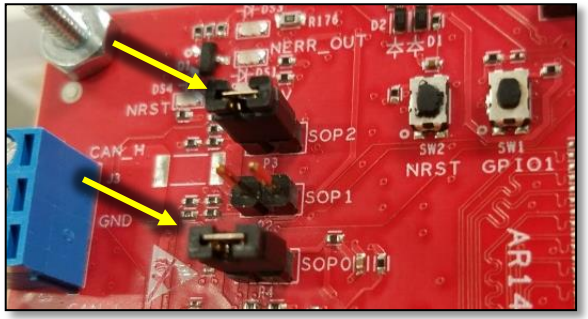
2. In the **Settings & Utilities** tab, fill the **COM Port** text box with the Application/User UART COM port number (**COM_{UART}**) noted earlier

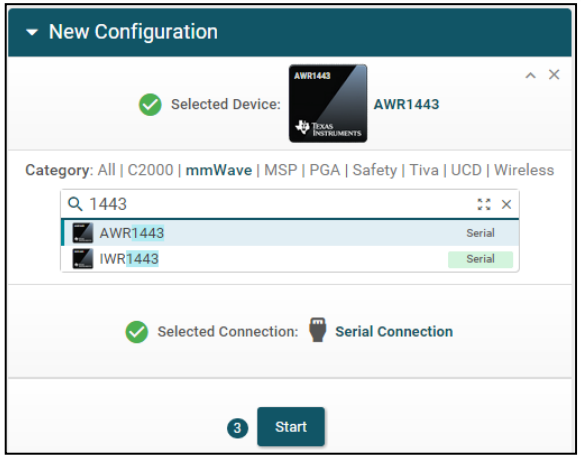


3. Return to the **Program** tab, power cycle the device and click on **Load Image**
4. When the flash procedure completes, UniFlash's console should indicate: [SUCCESS] Program Load completed successfully
5. Power off the board and remove the jumper from only header **SOP2** (this puts the board back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

4.4 Flashing IWR1443 Demo

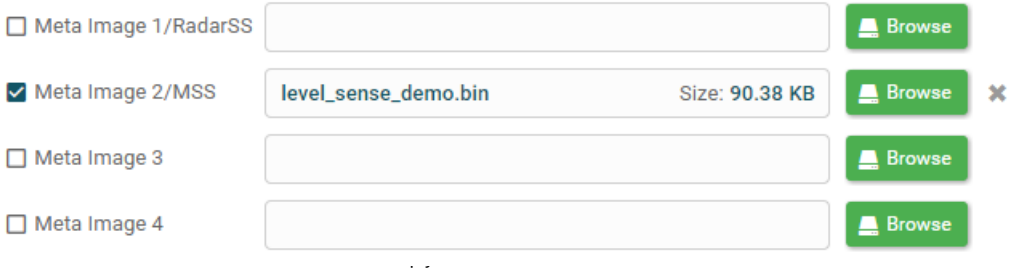
1. Put the EVM in flashing mode by connecting jumpers on SOP0 and SOP2 as shown in the image.
2. Open the **UniFlash** tool
3. In the **New Configuration** section, locate and select the IWR1443 device.
4. Click **Start** to proceed



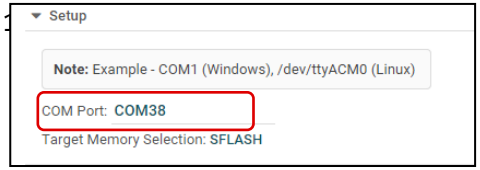


4.4 Flashing IWR1443 Demo

1. In the **Program** tab, browse and locate the **level_sense_demo.bin** image shown below:



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

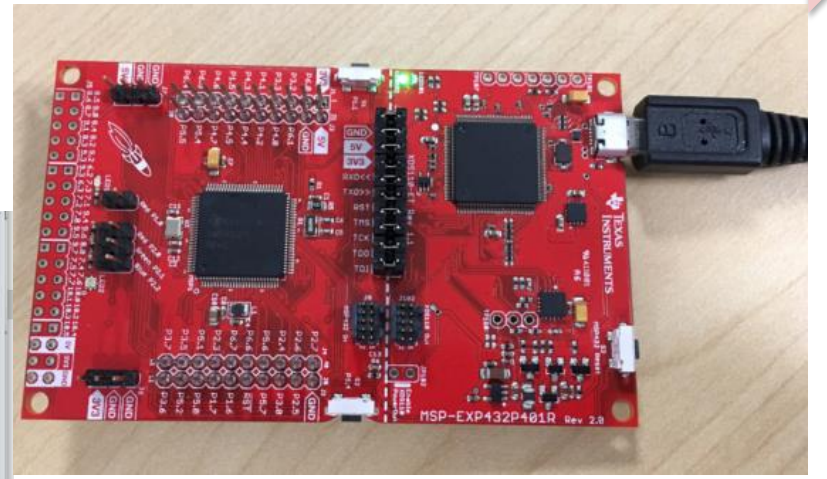


3. Return to the **Program** tab, power cycle the device and click on **Load Image**
4. When the flash procedure completes, UniFlash's console should indicate: [SUCCESS] Program Load completed successfully

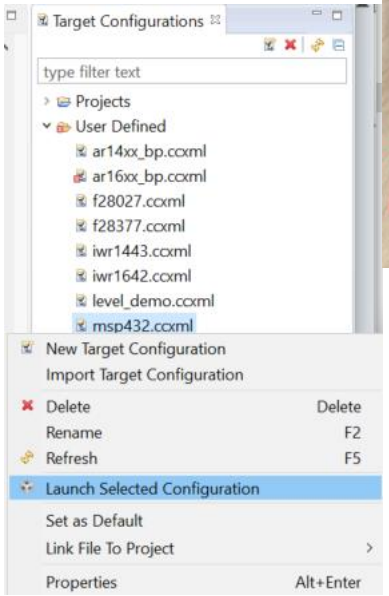
8. Power off the board and remove the jumper from only header **SOP2** (this puts the board back in functional mode)

4.5 Flashing the MSP432

1. Flashing the MSP432 image is performed using CCS.
2. First, attach a USB cable to the MSP432 and PC as shown to the right →

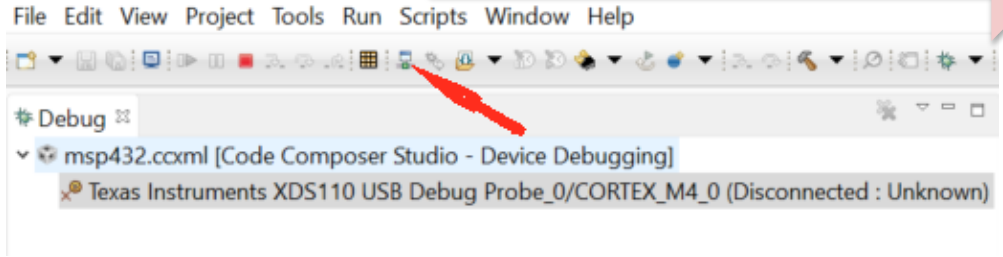


3. In the CCS Target Configuration window, right click on msp432.ccxml and select “Launch Selected Configuration”. (msp432.ccxml is provided in the software package zip if you don’t have it).
4. This will start the CCS Debugger.



4.5 Flashing the MSP432

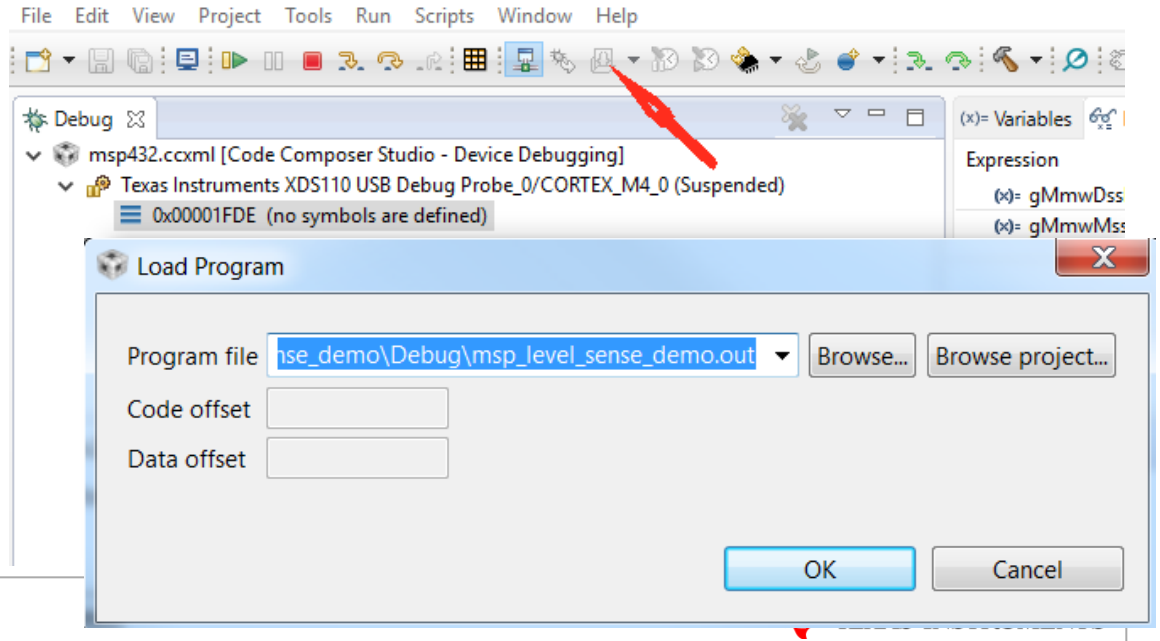
1. In the CCS Debugger, click on the “Connect” icon →



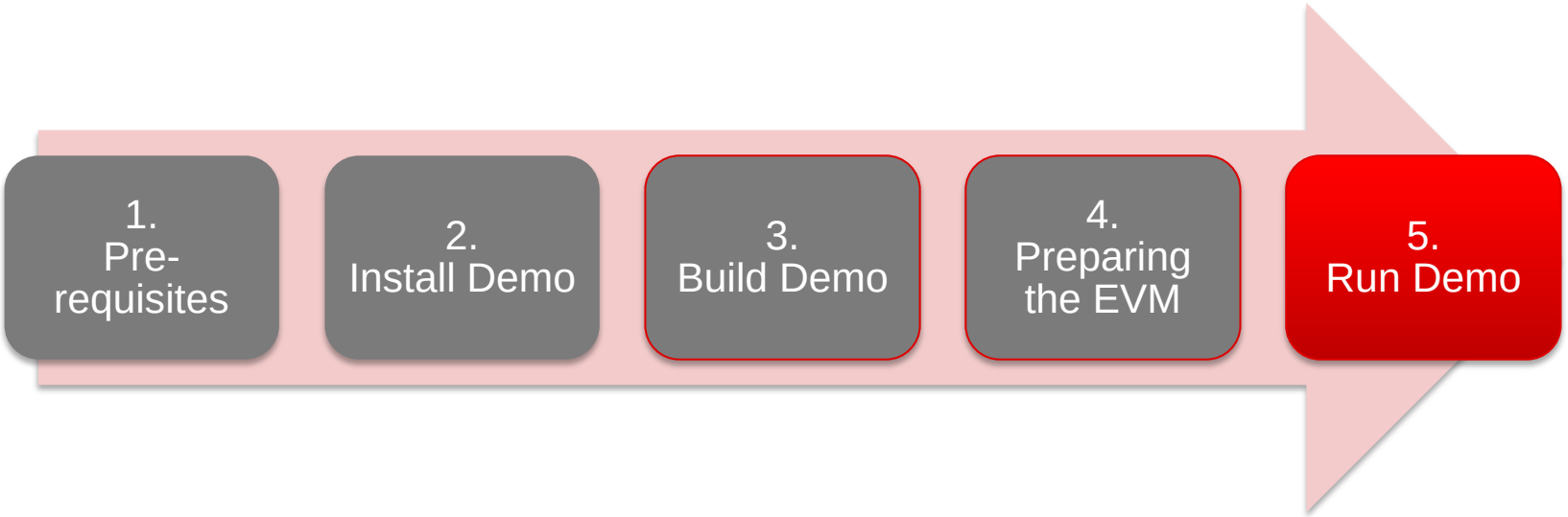
2. Next, click the “Load Target” icon, then browse to msp_level_sense_demo.out →

3. That’s it! Loading the program via the CCS Debugger also writes it to flash, so it will run automatically the next time the USB cable is attached.

4. Exit the CCS Debugger.

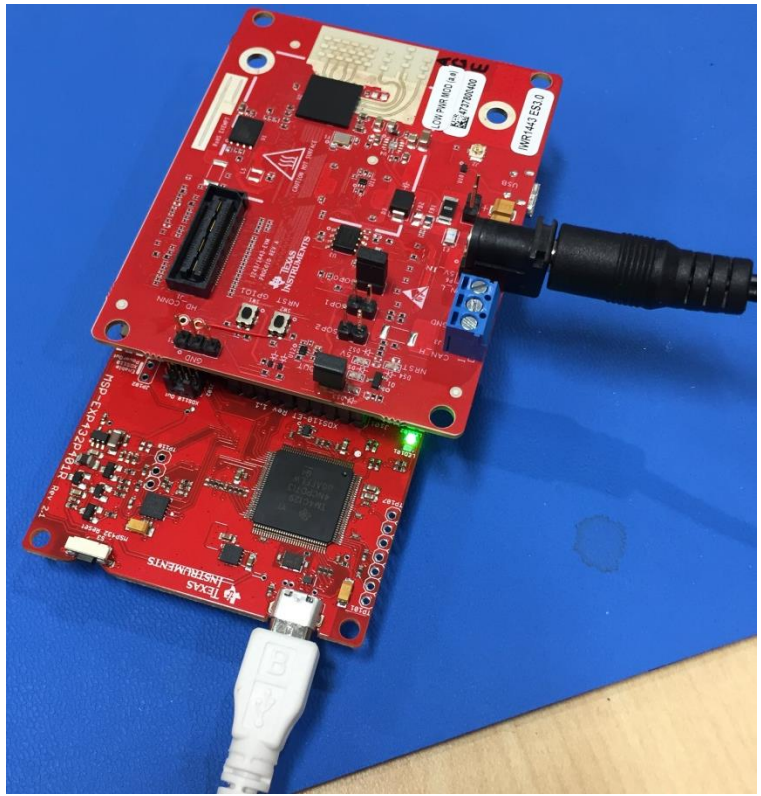


Steps


1.
Pre-
requisites
2.
Install Demo
3.
Build Demo
4.
Preparing
the EVM
5.
Run Demo

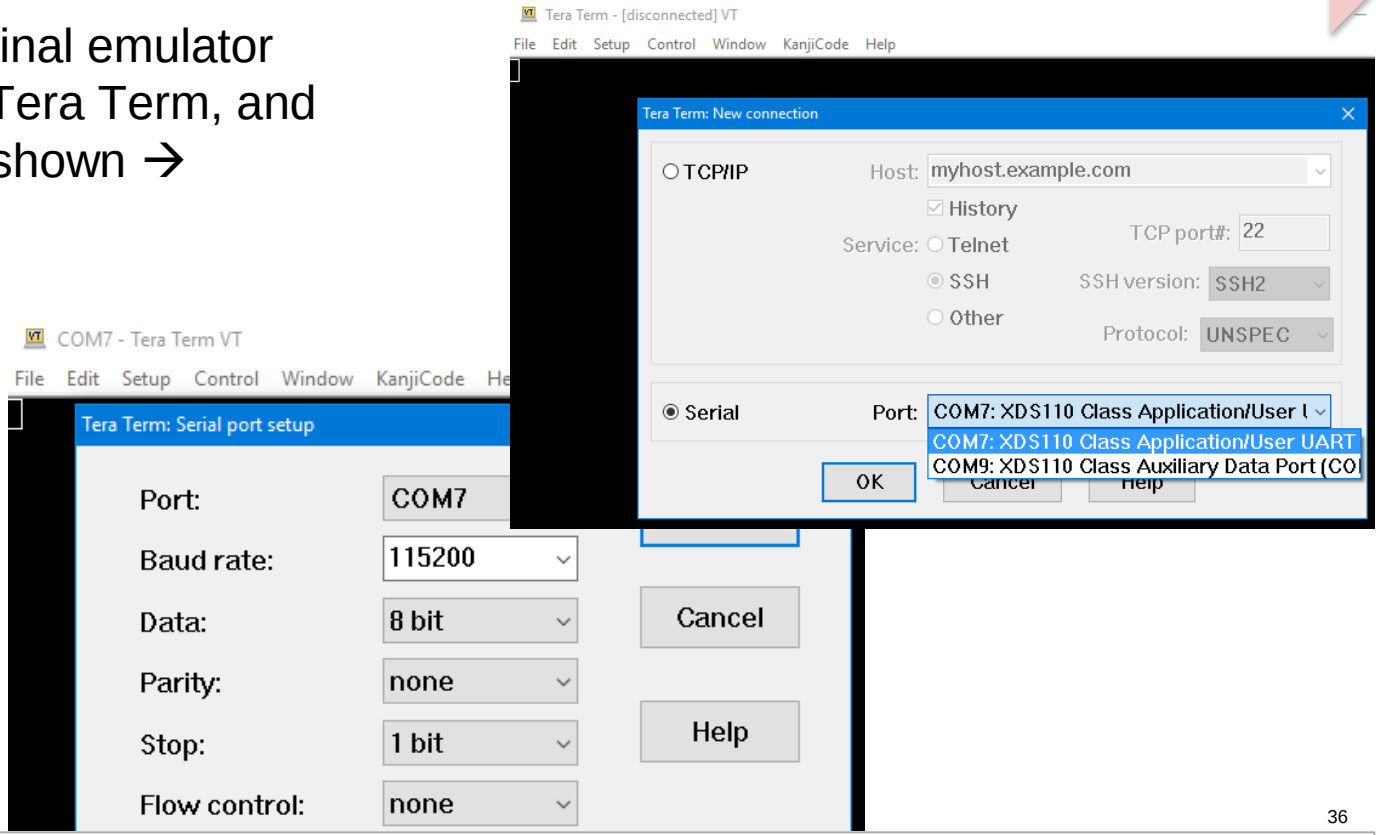
5.1 Run Demo with MSP432

- Once the IWR1443 and MSP432 EVMs are flashed with demo executables (discussed in the previous section), you are ready to run.
- First, attach the IWR1443 to the MSP432 as shown here. Carefully align the 20-pin and 3-pin headers and press straight down with even pressure.
- Next, connect the USB cable to the MSP432, and finally the 5V power cable to the IWR1443.
- The result should look like this →



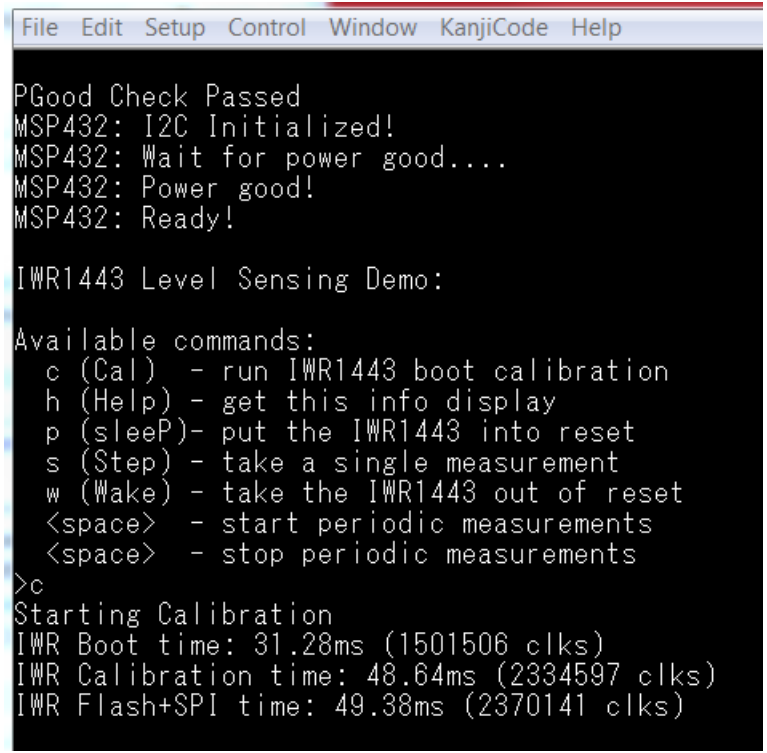
5.1 Run Demo with MSP432

- Next, start a Terminal emulator program such as Tera Term, and click on Serial as shown →
- Select the “Class Application/User” port.
- Make sure the serial port settings are the same as these →



5.1 Run Demo with MSP432

- Press the reset button on the MSP432 and the demo's menu will appear.
  - The MSP performs an I2C operation to set the PMIC-J to PFM mode for higher efficiency
- Commands are activated with single key presses.



5.1 Run Demo with MSP432

- The first operation must be a calibration with the ‘c’ key.
  - This operation must only be performed the first time software is loaded on to the device.

```
Available commands:
  c (Cal)  - run IWR1443 boot calibration
  h (Help) - get this info display
  p (sleeP)- put the IWR1443 into reset
  s (Step) - take a single measurement
  w (Wake) - take the IWR1443 out of reset
  <space>  - start periodic measurements
  <space>  - stop periodic measurements
>c
Starting Calibration
IWR Boot time: 31.28ms (1501506 clks)
IWR Calibration time: 48.64ms (2334597 clks)
IWR Flash+SPI time: 49.38ms (2370141 clks)

Calibration Successful
>
```

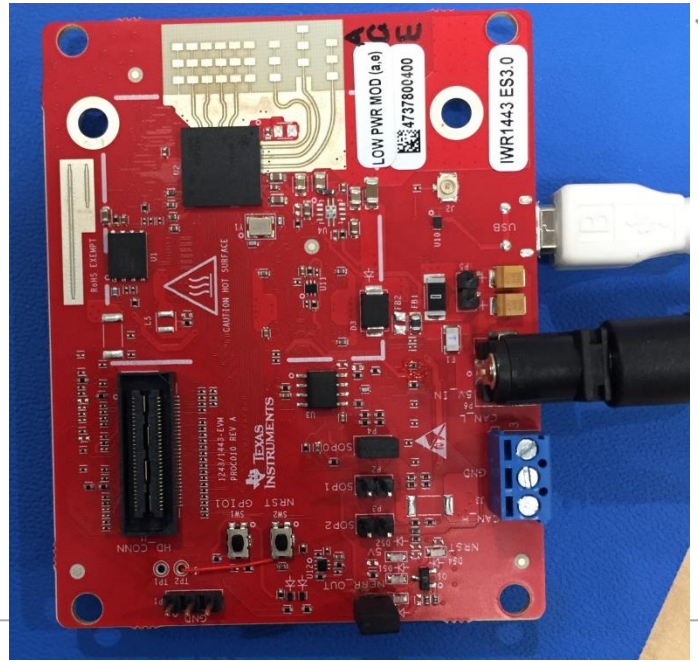
5.1 Run Demo with MSP432

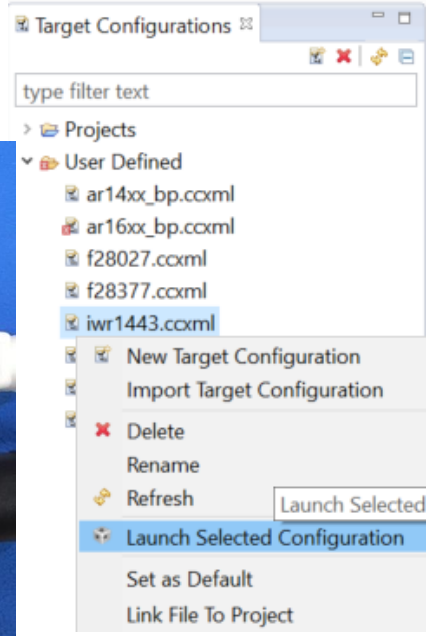
- After this one can press the ‘s’ key to perform a measurement.

```
s
Start one-shot measurement
IWR Boot time: 32.06ms (1539031 clks)
IWR Config time: 11.09ms (532395 clks)
IWR Chirp time: 2.39ms (114499 clks)
IWR Post time: 0.95ms (45577 clks)

IWR1443 Data:
Dist  1.353159(m), power 467063469
Dist  1.390747(m), power 418227422
Dist  1.315572(m), power 119002159
>
```

5.2 Debug the demo in CCS

- To load the IWR1443 into the CCS Debugger, you must first flash the IWR1443 with the “CCS Debug” image as described in section 4.3.
- You should see your iwr1443 target configuration under **User Defined** configurations →

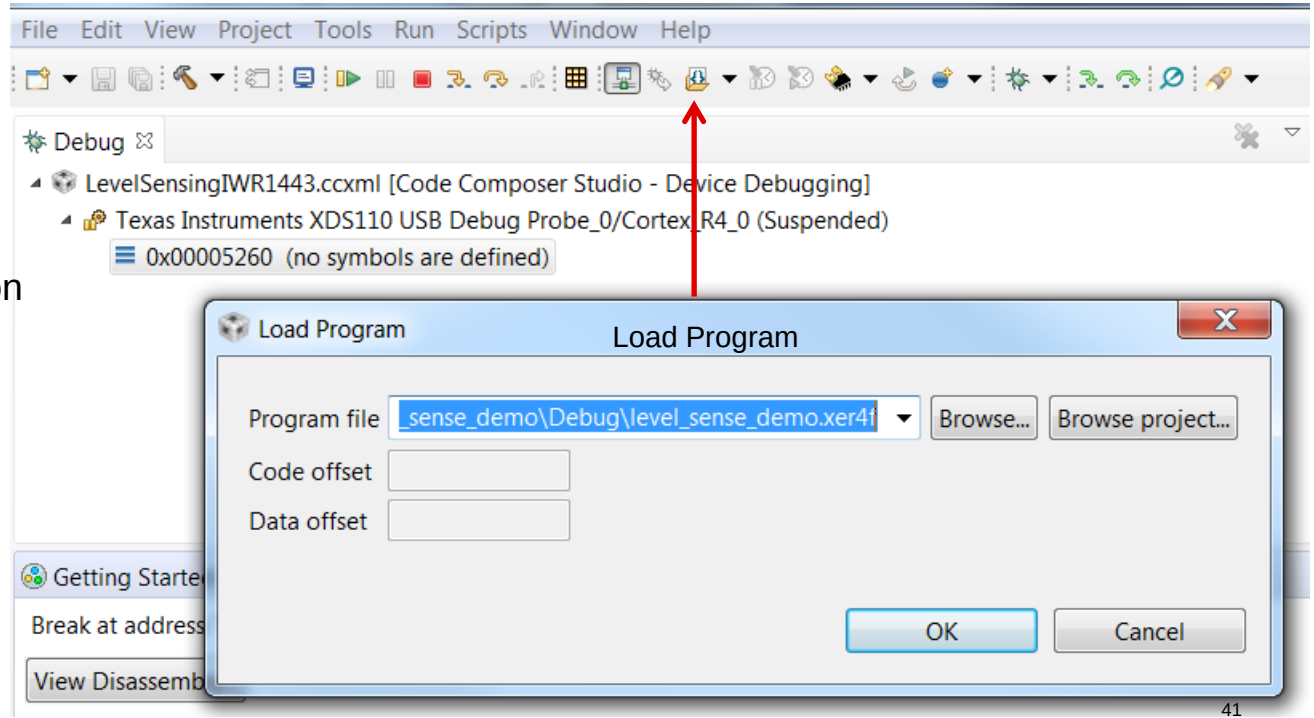




- Detach the IWR1443 from the MSP432 gently by pulling and rocking it slightly in the same direction as the 20-pin headers.
- Attach the USB and power cables as shown.
- Make sure the SOP2 jump is not in place.
- Right click on the target configuration and select **Launch Selected Configuration**.

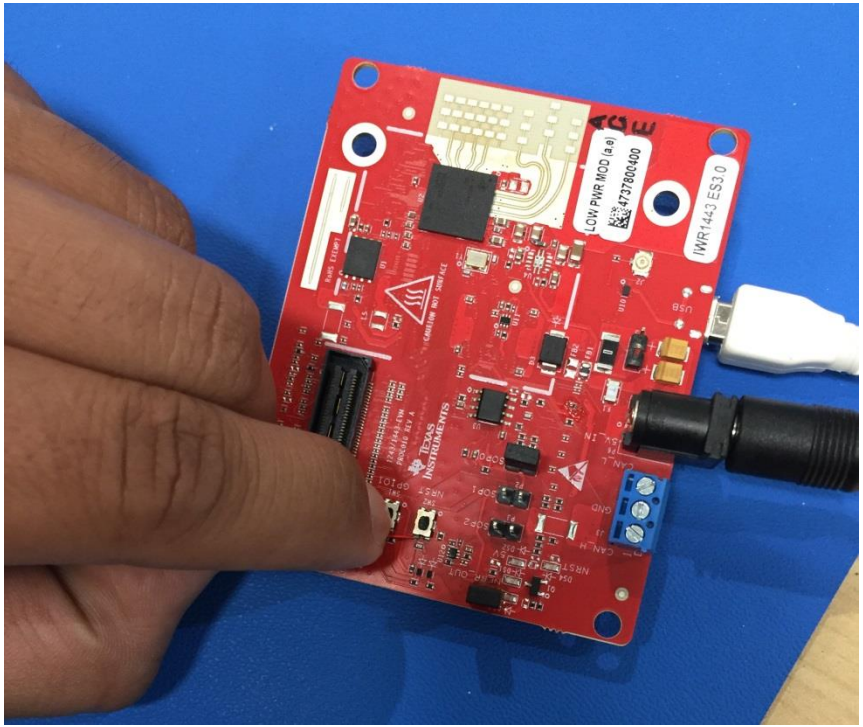
5.2 Debug the demo in CCS

- With the target connected, click on the **Load** button in the toolbar.
- In the **Load Program** dialog, browse to the level_sense_demo.xer4f file and click OK.
- Before proceeding a calibration is required.



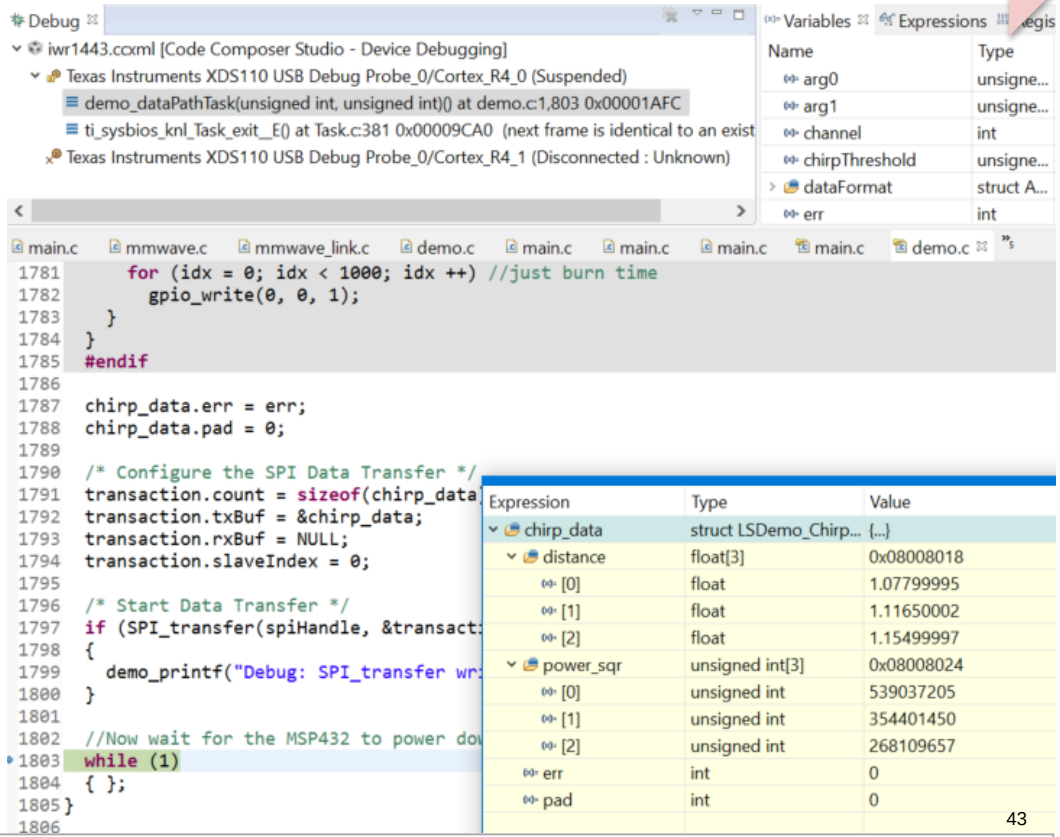
5.2 Debug the demo in CCS - Calibration Step

- We'll need to perform a manual calibration before taking measurements.
- Hold down the GPIO1 button on the EVM and then press the Run button in CCS. 
- Press the Suspend Button. 
- Press the CPU Reset Button. 
- Then press the Run button again to perform a measurement. 



5.3 Debug the demo in CCS

- If you let the demo run to the end, it will stop in a while (1) loop. This is because it is waiting for the MSP432 to power it down.
- You can then hover the mouse over “chirp_data” in the code and it will open a new window showing the structure values.
- chirp_data is the structure that is passed to the MSP432 via the SPI interface.



Additional Resources

- TIDEP-0091 – Level Sensing TI Design:
  - http://www.ti.com/tool/TIDEP-0091
- Learn more about xWR1443 devices, please visit the product pages
  - IWR1443: http://www.ti.com/product/IWR1443
  - AWR1443: http://www.ti.com/product/AWR1443
- Get started evaluating the platform with xWR1443 EVMs, purchase EVM at
  - IWR1443 EVM: http://www.ti.com/tool/IWR1443BOOST
  - AWR1443 EVM: http://www.ti.com/tool/AWR1443BOOST
- Download mmWave SDK @ http://www.ti.com/tool/MMWAVE-SDK
- Ask question on TI's E2E forum mmWave Sensors forum @ https://e2e.ti.com/support/sensor/mmwave_sensors/