

Child-Presence and Occupant-Detection Reference Design Using 60-GHz Antenna-on-Package mmWave Sensor



Description

This reference design demonstrates the use of the AWR6843AOP, a 60GHz single chip mmWave sensor with integrated antenna. The sensor also has integrated DSP, MCU and a hardware accelerator for FFT processing. The 60GHz TI AWR6843 AOP sensor can help developers to address a wide range of incabin sensing applications, including uses cases like detection of child across 2 rows of a car, seat belt warning, and intruder detection. The small form factor enables placement of sensor in locations like overhead console around the rear-view mirror, in the headliner, in the B/C pillars and the seats. It thus provides flexibility to Tier1's and car makers in planning the end use cases.

Resources

TIDEP-01023	Design Folder
TIDEP-01001	Design Folder
AWR6843AOPEVM	Tool Folder
AWR6843AOP	Product Folder

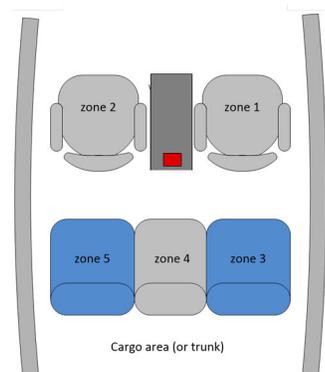
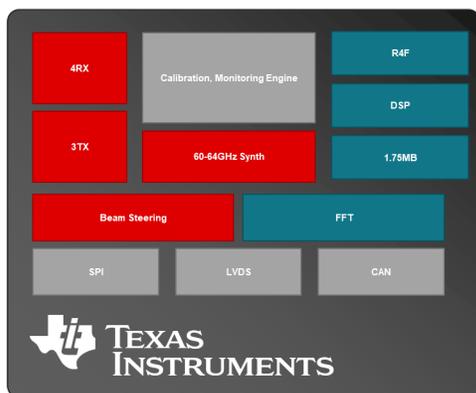


Features

- Demonstration hardware and software using AWR6843AOP for multi-functional sensing: child presence detection, seat occupancy detection, and intruder detection
- Overhead roof mounted capable of sensing across two rows of a vehicle with approximate azimuth field of view of 120 degrees and elevation field of view of 120 degrees (Refer to [TIDEP-01001](#) for Side Mount Sensing alternative)
- Signal processing for point cloud detection handled onboard AWR6843 using the integrated HWA, DSP, and R4F

Applications

- [Vehicle Occupant Detection Sensor](#)
- [Driver Vital Sign Monitoring](#)
- [Gesturing](#)



1 System Description

Today's vehicles require robust and reliable information about the in-cabin occupancy. Smart airbag deployment systems, air condition controls, detecting children and disabled people left behind in vehicles relies upon this information. The TIDEP-01023 provides a reference for creating a vehicle occupant detection application, using TI's AWR6843AOP sensor based on 60-GHz mmWave radio-frequency complementary metal-oxide semiconductor (RF-CMOS) technology. TI's mmWave sensing devices integrate a 60-GHz to 64-GHz mmWave radar front end with ARM® microcontroller (MCU) and TI DSP core for single-chip systems.

This reference design demonstrates the suitability of the AWR6843AOP for overhead mount vehicle occupant detection applications. The design targets the implementation of a wide azimuth and elevation field of view ($\pm 60^\circ$) and close range (3 m) sensor configuration, which can detect life forms across multiple regions (zones) of interest, and can localize occupant signature within a region. This reference design implements algorithms for generating an azimuth-range and elevation-range heat maps, detection, and decision using an AWR6843AOP device on a TI EVM module. The design provides a list of required hardware, schematics, and foundational software to quickly begin vehicle occupancy detection product development. It describes the example usage case as well as the design principle, implementation details, and engineering tradeoffs made in the development of this application. High-level instructions for replicating the design are provided.

Table 1-1. Key Performance Specifications

PERFORMANCE PARAMETER	CONFIGURATION
Device	AWR6843AOP
Field of view	120° horizontal, 120° vertical
Maximum range	2.7 m
Range resolution	5.3 cm
Maximum velocity	1.7 m/s
Velocity resolution	1.5 cm/s

2 System Overview

2.1 Block Diagram

2.1.1 Hardware Block Diagram

The TIDEP-01000 is implemented on the IWR6843 ISK EVM see [Figure 2-1](#). The EVM is connected to a host PC through a universal asynchronous receiver-transmitter (UART) for visualization.

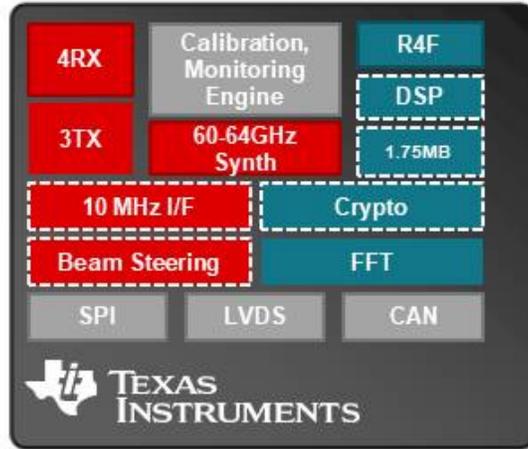


Figure 2-1. Hardware Block Diagram

2.1.2 Software Block Diagram

2.1.2.1 mmWave SDK Software Block Diagram

The mmWave software development kit (SDK) enables the development of mmWave sensor applications using the AWR6843AOP SOC and EVM. The SDK provides foundational components that allow end users to focus on their applications. In addition, the SDK provides several demonstration applications, which serve as a guide for integrating the SDK into end-user mmWave applications. This reference design is a separate package installed in addition to the SDK package, and is the *application* portions of the block diagram.

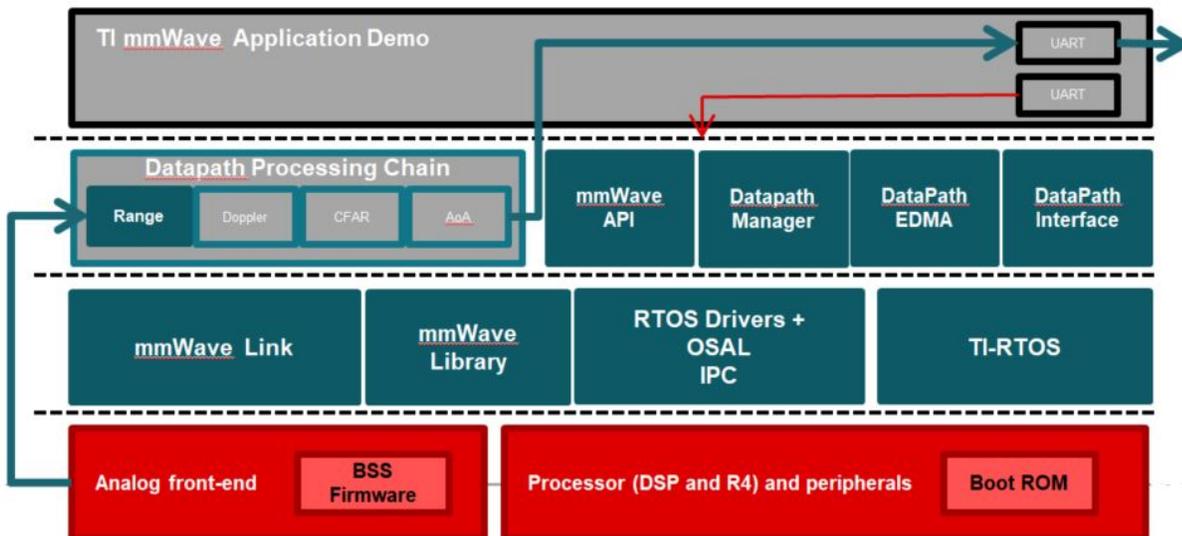


Figure 2-2. mmWave SDK Software Block Diagram

2.1.2.2 Software Block Diagram of Application

As shown in [Figure 2-3](#), the implementation of the application demo on the AWR6843AOP consists of a signal chain running on the C674x DSP, and the tracking module running on the ARM® Cortex®-R4F processor.

At its core, the demo does two things:

1. Use the radar data to produce a point cloud with each point containing a range, azimuth angle, elevation angle, and signal to noise ratio (SNR)
2. Finds and determines seat occupancy status from the point cloud based on defined zone locations.

Processing Overview:

- Range processing:
 - For each antenna, 1D windowing, and 1D fast Fourier transform (FFT)
 - Range processing is interleaved with the active chirp time of the frame
 - Implemented on HWA and Cortex R4F
- Capon Beamforming (BF):
 - Static clutter removal
 - Covariance matrix generation, angle spectrum generation, and integration is performed
 - Outputs range-azimuth heat map
 - Implemented on c674 DSP
- CFAR detection algorithm:
 - Two-pass, constant false-alarm rate
 - First pass cell averaging smallest of CFAR-CASO in the range domain, confirmed by second pass cell averaging smallest of CFAR-CASO in the angle domain, to find detection points.
 - Implemented on c674 DSP
- Elevation Beamforming
 - Capon BF algorithm is applied again for each point detected in Range-Azimuth heatmap
 - 1-D Elevation Spectrum is generated and strongest signal is taken as the detected angle
 - Implemented on c674 DSP
- Zone Occupancy Decision:
 - Operates on point cloud
 - Maps the point cloud to zone definition in a car
 - For each zone, based on the number of detected points and the quality of these detected point in a zone, and the previous occupancy state, a decision of occupancy state will be updated.
 - Implemented on PC as part of the visualizer

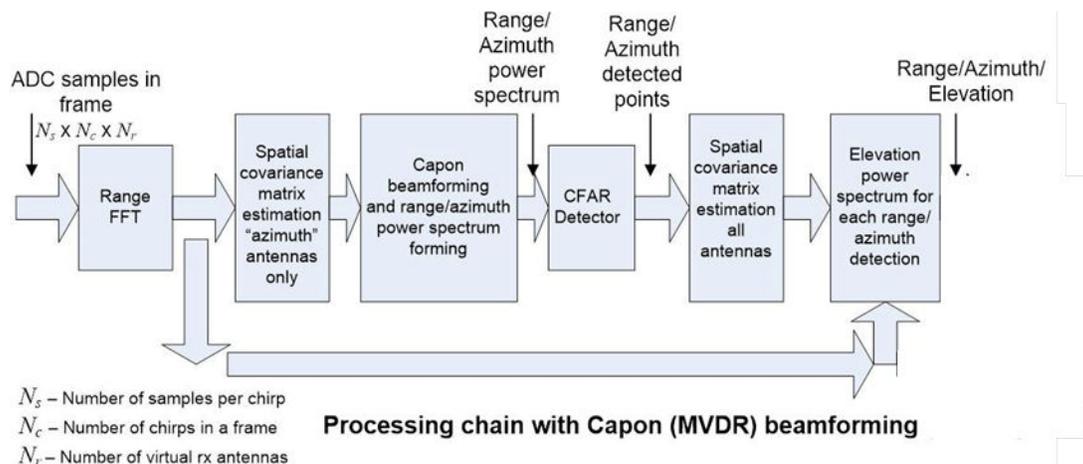


Figure 2-3. Application Block Diagram

2.2 Highlighted Products

2.2.1 AWR6843AOP

The AWR6843AOP is an integrated, single-chip, frequency modulated continuous wave (FMCW) sensor capable of operation in the 60 to 64 GHz band. The sensor is built with the low-power, 45nm, RFCMOS process from TI and enables unprecedented levels of integration in an extremely small form factor. The AWR6843AOP is an ideal solution for low-power, self-monitored, ultra-accurate radar systems in the in-cabin sensing space.

There are several benefits of an antenna on the package technology:

- Small form factor: Antenna on the PCB takes up almost ~30% of the board space. With the antenna now integrated on the package, the size of the sensor reduces by ~30% compared to conventional sensors. This helps in easy vehicle integration.
- Faster design cycle: Developer need not spend time on simulation and characterization of the antenna parameters.
- Lower PCB cost: Low cost FR4 based PCB can be chosen for design instead of expensive roger's material-based PCB. This saves cost.

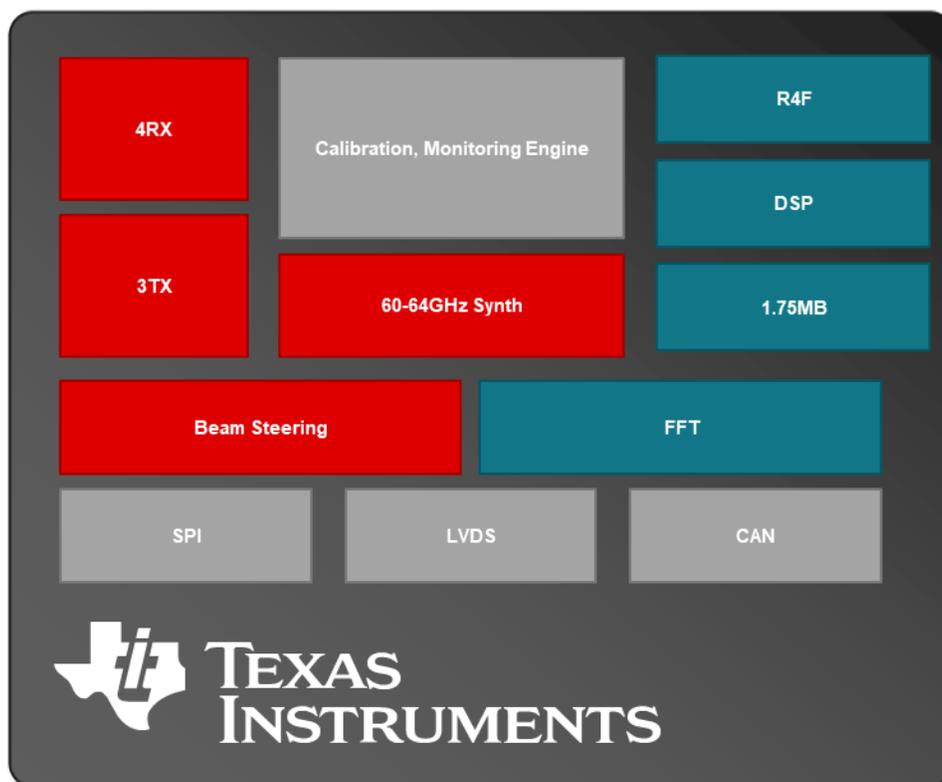


Figure 2-4. IWR6843 Block Diagram

The AWR6843AOP has the following features:

- FMCW transceiver
 - Integrated PLL, transmitter, receiver, baseband, and A2D
 - 60 to 64GHz coverage, with 4GHz available bandwidth
 - Four receive channels
 - Three transmit channels
 - Ultra-accurate chirp (timing) engine based on fractional-N PLL
 - TX power
 - 12 dBm
 - RX noise figure

- 12 dB (60 to 64 GHz)
- Phase noise at 1 MHz
 - –92 dBc/Hz (60 to 64 GHz)
- Built-in calibration and self-test (monitoring)
 - ARM Cortex-R4F-based radio control system
 - Built-in firmware (ROM)
 - Self-calibrating system across frequency and temperature
- C674x DSP for FMCW-signal processing
 - On-chip memory: 1.75MB
- Cortex-R4F MCU for object detection, and interface control
 - Supports autonomous mode (loading the user application from QSPI flash memory)
- Integrated peripherals
 - Internal memories with ECC
 - Up to six ADC Channels
 - Up to two SPI Channels
 - Up to two UARTs
 - CAN interface
 - I²C
 - GPIOs
 - Two-lane LVDS interface for raw ADC data and debug instrumentation

2.2.2 mmWave SDK

The mmWave SDK is the foundational software component of the mmWave projects and includes the following smaller components:

- Drivers
- OSAL
- mmWaveLink
- mmWaveLib
- mmWave API
- Data Path Manager
- BSS firmware
- Board setup and flash utilities

For more information, see the mmWave SDK user's guide.

2.3 System Design Theory

2.3.1 Use-Case Geometry and Sensor Considerations

The AWR6843AOP is a radar-based sensor that integrate a fast FMCW radar front end with both an integrated ARM R4F MCU and the TI C674x DSP for advanced signal processing. The key performance parameters of the system are influenced by the configuration of the transmit signal and the RF transceiver, the design of the antenna array, and the available memory and processing power.

The key performance parameters at issue follow with brief descriptions:

- Maximum Range - maximum range at which a human can be detected
- Range Resolution - minimum distance require between two objects for the radar to detect them as separate objects
- Maximum Velocity - the maximum unambiguous velocity that can be detected. Objects moving faster than this may have incorrect velocity measurements
- Velocity resolution - minimum velocity difference between two objects for the radar to detect them as separate objects

When designing the frame and chirp configuration for a vehicle occupancy detection use case, start by considering increasing range resolution and velocity resolution over maximum range and velocity; because objects are within short range, defined zones will be relatively stationary. [Table 2-1](#) lists example chirp configurations with good range and Doppler resolution and memory for a incabin sensing application.

Table 2-1. Performance Parameters of Example Chirp Design

KEY INPUT PARAMETERS	
Starting frequency (GHz)	60
Maximum range, R_{max} (m)	2.72
Range resolution (cm)	5.3
Maximum velocity (m/s)	1.7
Velocity resolution (m/s)	0.0154
Idle time (μ s)	205
ADC valid start time (μ s)	11
Periodicity (ms)	200
Valid sweep bandwidth (MHz)	2822
Ramp slope (MHz/ μ s)	97
Sampling frequency (Msps)	2.2
Number of samples per chirp	64
Ramp end time (μ s)	41
Number of chirp loops	220
Active frame time (ms)	162

2.3.2 Low-Level Processing

An example of a processing chain for overhead in-cabin sensing is implemented on the AWR6843AOP EVM.

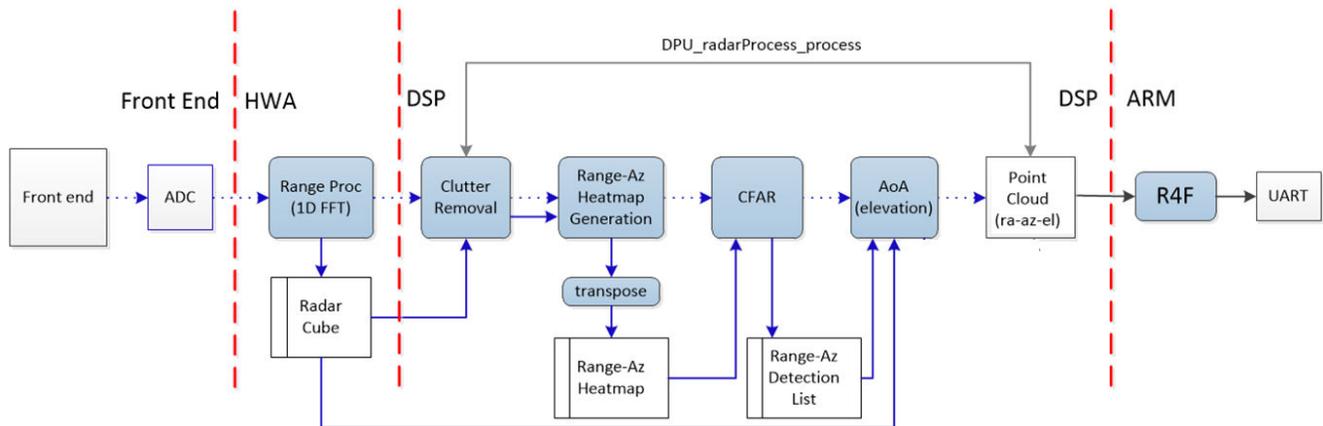
The processing chain is implemented on the DSP and Cortex-R4F together. [Table 2-2](#) lists the several physical memory resources used in the processing chain.

Table 2-2. Memory Configuration

SECTION NAME	SIZE (KB) AS CONFIGURED	MEMORY USED (KB)	DESCRIPTION
L1D SRAM	16	16	Layer one data static RAM is the fastest data access for DSP and is used for most time-critical DSP processing data that can fit in this section.
L1D cache	16	16	Layer one data cache caches data accesses to any other section configured as cacheable. LL2, L3, and HSRAM are configured as cache-able.

Table 2-2. Memory Configuration (continued)

SECTION NAME	SIZE (KB) AS CONFIGURED	MEMORY USED (KB)	DESCRIPTION
L1P SRAM	28	28	Layer one program static RAM is the fastest program access RAM for DSP and is used for most time-critical DSP program that can fit in this section.
L1P cache	4	4	Layer one cache caches program accesses to any other section configured as cacheable. LL2, L3, and HSRAM are configured as cache-able.
L2	256	2328	Local layer two memory is lower latency than layer three for accessing and is visible only from the DSP. This memory is used for most of the program and data for the signal processing chain.
L3	768	744	Higher latency memory for DSP accesses primarily stores the radar cube and the range-azimuth heat map. It also stored system code not required to be executed at high speed.
HSRAM	32	20	Shared memory buffer used to store slow, non-runtime code.

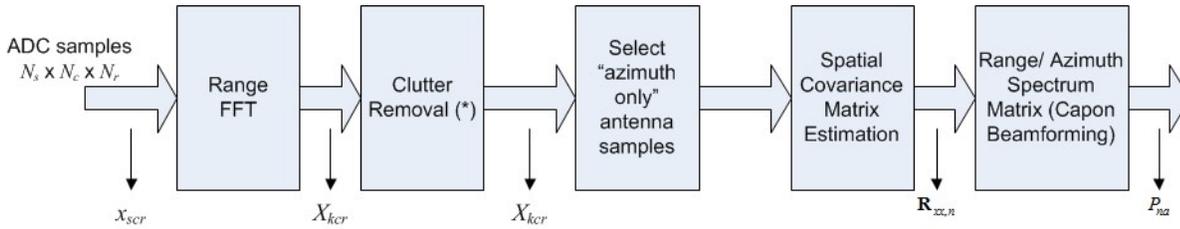

Figure 2-5. Processing Chain Flow: Detection Tracking Visualization

As shown in [Figure 2-5](#), the implementation of the over in-cabin sensing example in the signal-processing chain consists of the following blocks implemented on both the DSP and Cortex R4F. In the following section we break this process into the following smaller blocks:

1. Range FFT through Range Azimuth Heatmap with Capon BF
2. Object Detection with CFAR and Elevation Estimation

Range FFT through Range Azimuth Heatmap with Capon BF

- As shown in the block diagram, Raw Data is processed with a 1-D FFT (Range Processing) and Static Clutter Removal is applied to the result. Then Capon Beamforming is used to generate a range-azimuth heatmap. These are explained in depth below.



N_s - Number of samples per chirp
 N_c - Number of chirps
 N_r - Number of virtual receive antennas
 $N_{r,a}$ - Number of virtual azimuth antennas

Figure 2-6. Range FFT through Range-Azimuth Heatmap

Range processing

- For each antenna, EDMA is used to move samples from the ADC output buffer to the FFT Hardware Accelerator (HWA), controlled by the CortexR4F. A 16-bit, fixed-point 1D windowing and 16-bit, fixed-point, 1D FFT are performed. EDMA is used to move output from the HWA local memory to the radar cube storage in layer three (L3) memory. Range processing is interleaved with active chirp time of the frame. All other processing occurs each frame, except where noted, during the idle time between the active chirp time and the end of the frame.

Static Clutter Removal

- Once the active chirp time of the frame is complete, the inter-frame processing can begin, starting with static clutter removal. 1D FFT data is averaged across all chirps for a single Virtual Rx antenna. This average is then subtracted from each chirp from the Virtual Rx antenna. This cleanly removes the static information from the signal, leaving only the signals returned from moving objects. The formula is

$$X_{nr} = \frac{1}{N_c} \sum_{c=1}^{N_c} X_{ncr} \quad c = 1 \dots N_c \quad X_{ncr} = X_{ncr} - X_{nr} \tag{1}$$

- With N_c = Number of chirps; N_r = Number of receive antennas; X_{nr} = Average samples for a single receive antenna across all chirps; X_{ncr} = Samples of a Single Chirp from a receive antenna

Capon Beamforming

- The Capon BF algorithm is split into two components: 1) the Spatial Covariance Matrix computation and 2) Range-Azimuth Heatmap Generation. The final output is the Range-Azimuth heatmap with beamweights. This is passed to the CFAR algorithm.
- Spatial Covariance Matrix is calculated as the following:
 - First, the spacial covariance matrix is estimated as an average over the chirps in the frame as $R_{xx,n}$ which is 8x8 for ISK and 4x4 for ODS:

$$R_{xx,n} = \frac{1}{N_c} \sum_{c=1}^{N_c} X_{nc} X_{nc}^H \quad X_{nc} = [X_{nc1}, \dots, X_{ncNr}]^T \tag{2}$$

- Second, diagonal loading is applied to the R matrix to ensure stability

$$R_{xx,n} = R_{xx,n} + \alpha \frac{\text{tr}(R_{xx,n})}{N_r} I_{N_r} \tag{3}$$

Range-Azimuth Heatmap Generation

- First, the Range-Azimuth Heatmap P_{na} is calculated using the following equations
- Subscript a indicates values across azimuth bins

$$P_{na} = \frac{1}{a_a H R x x, n-1 a_a} a_a = [1, e^{j\mu_a}, \dots, e^{j(N_r-1)\mu_a}]^T \mu_a = 2\pi \frac{d}{\lambda} \sin(\theta_a) \quad (4)$$

- In AOP antenna pattern, there are two sets of antennas that can be used for range-azimuth heatmap generation. In this application, the two sets are combined as defined below to achieve the final range-azimuth heatmap. Where P_{na1} represent the heatmap generated from the first set of azimuth antenna array using equation and P_{na2} represent the heatmap generated from the second set of azimuth antenna array.

$$P_{na} = P_{na1}^2 + P_{na2}^2 \quad (5)$$

Object Detection with CFAR and Elevation Estimation

- Using the heatmap generated in the above steps, 2 Pass CFAR is used to generate detected points in the Range-Azimuth spectrum. For each detected point, Capon is applied to generate a 1D elevation angular spectrum, which is used to determine the elevation angle of the point

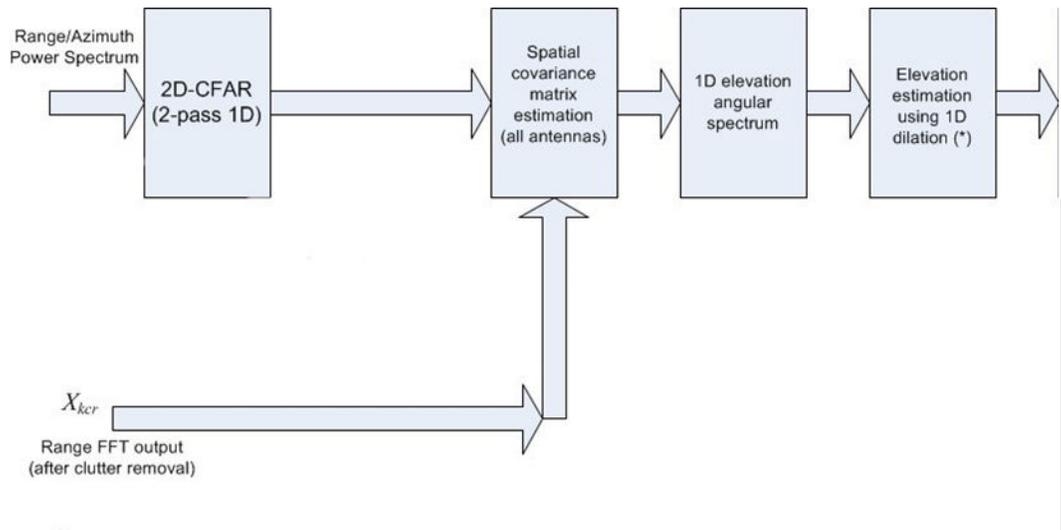


Figure 2-7. CFAR and Elevation Estimation

Object detection

- Two pass CFAR algorithms is used on the range azimuth heat map to perform the object detection using the CFAR *smallest of* method. First pass is done per angle bin along the range domain. Second pass in the angle domain is used confirm the detection from the first pass. The output detected point list is stored in L2 memory.

Elevation Estimation with Capon BF

- Full 2D 12 antenna Capon Beamforming is performed at the azimuth of each detected point. This is done following the same steps used to generate the range-azimuth heatmap:
 1. Generate spacial covariance matrix
 2. Generate 1D elevation angle spectrum (similar to the heatmap)
- Then a single peak search is performed to find the elevation angle of each point. This step does not generate new detection points.
- Spatial Covariance matrix is similar to before, with input based on detections

$$R_{x x, m} = \frac{1}{N_c} \sum c = 1 N_c X_{k c} X_{k c} H_{k c} = r_{d e t, m} \quad (6)$$

- With diagonal loading matrix

$$R_{x x, m} = R_{x x, m} + \alpha_2 \frac{\text{tr}(R_{x x, m})}{N_r} I_{N_r} \quad (7)$$

- 1D Elevation Spectrum is as follows

$$P_m = \frac{1}{a_m H R \times x, m - 1 a_m} a_m = a(\mu_m, u_m) = a(\mu_m) \otimes a(u_m) a(\mu_m) = [1, e^{j\mu_m}, \dots, \quad (8)$$

$$e^{j(N_r - 1)\mu_m}]^T a(u_m) = [1, e^{ju_m}, \dots, e^{j(N_r - 1)u_m}]^T$$

All the above processing except the range processing happens during inter-frame time. After DSP finishes frame processing, the results are written in shared memory (L3/HSRAM) for Cortex-R4F to input for the group tracker.

Zone Occupancy Detection

- This algorithm implements the localization processing and works on the point cloud data from DSP. Using range, azimuth, elevation, and SNR for each point in the point cloud, it outputs occupancy decision for each defined zone.

[Table 2-3](#) lists the results of benchmark data measuring the overall MIPS consumption of the signal processing chain running on the DSP. Time remaining assumes a 50 ms total frame time.

Table 2-3. MIPS Use Summary

PARAMETER	TIME USED (ms)	LOADING (ASSUMING 200 ms FRAME TIME)
Active Frame time	158.76	79.4%
Remaining Inter-Frame Time	41.24	20.6%
Range-Azimuth Heatmap Generation	2.18	1.09%
2 Pass CFAR	0.30	0.15%
Elevation Estimation	2.76	1.38%
Zone Assignment (not currently running on DSP)	2.0 (estimate)	1%
Occupancy State Machine (not currently running on DSP)	1.0 (estimate)	0.5%
Total Active Inter-Frame Time	7.24	3.62%
Total Time	36.235	18.11%

2.3.3 High-Level Processing Details

2.3.3.1 DPM Model

Processing on the device is handled by the Data Path Manager (DPM). The DPM coordinates processing on the R4F and DSP. The group tracker (gtrack) is run on Cortex R4F, in a separate task. Gtrack is encapsulated in a DPU, so it can be initialized and configured by the DPM. During runtime, gtrack runs in a separate task. This task pends on a semaphore, which is released when the point cloud is received and stored from the DSP. This allows the Cortex R4F to transmit data on UART or process ADC data through the HWA while the tracker is running on the point cloud from the previous frame. The main task in mss is referred to as a Mailbox Task below.

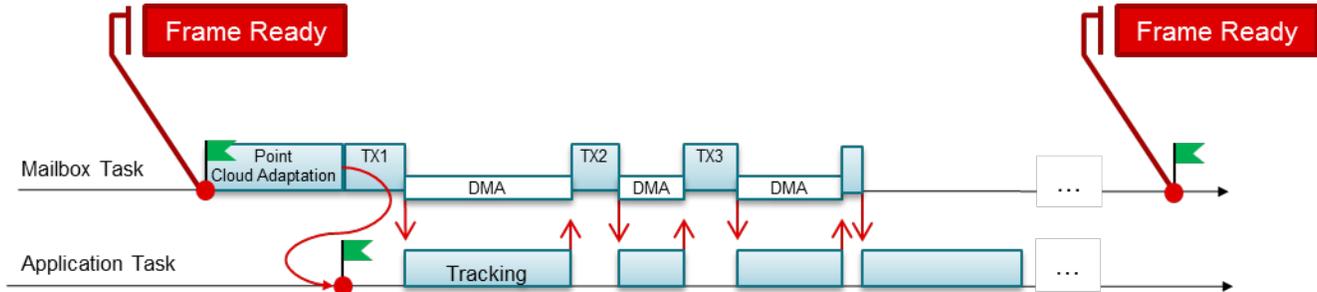


Figure 2-8. High-Level Processing Task Model

2.3.3.2 Occupancy Detection State Machine

The Occupancy Detection State Machine is the processing that examines the detections within each zone and makes yes or no occupancy decisions each frame. [Figure 2-9](#) shows the steps algorithm goes during each frame call.

- Entry conditions
 - Small number of detections with a high average SNR
 - – or –
 - Larger number of detections with smaller average SNR
- Stay condition
 - num detections passing SNR threshold requirement
- Forget condition
 - exceeds number of frames failing the Stay condition
- Overload condition
 - High energy level (vehicle entry, exit, or someone changing seats). This causes all zone states to be frozen until the overload subsides.

All condition parameters and thresholds are configurable via CLI interface. The values used in the application demo and the state machine are not definitive. Continued improvements and customization of the state machine is recommended to fine-tune the processing to fit application needs.

Zone State Machine

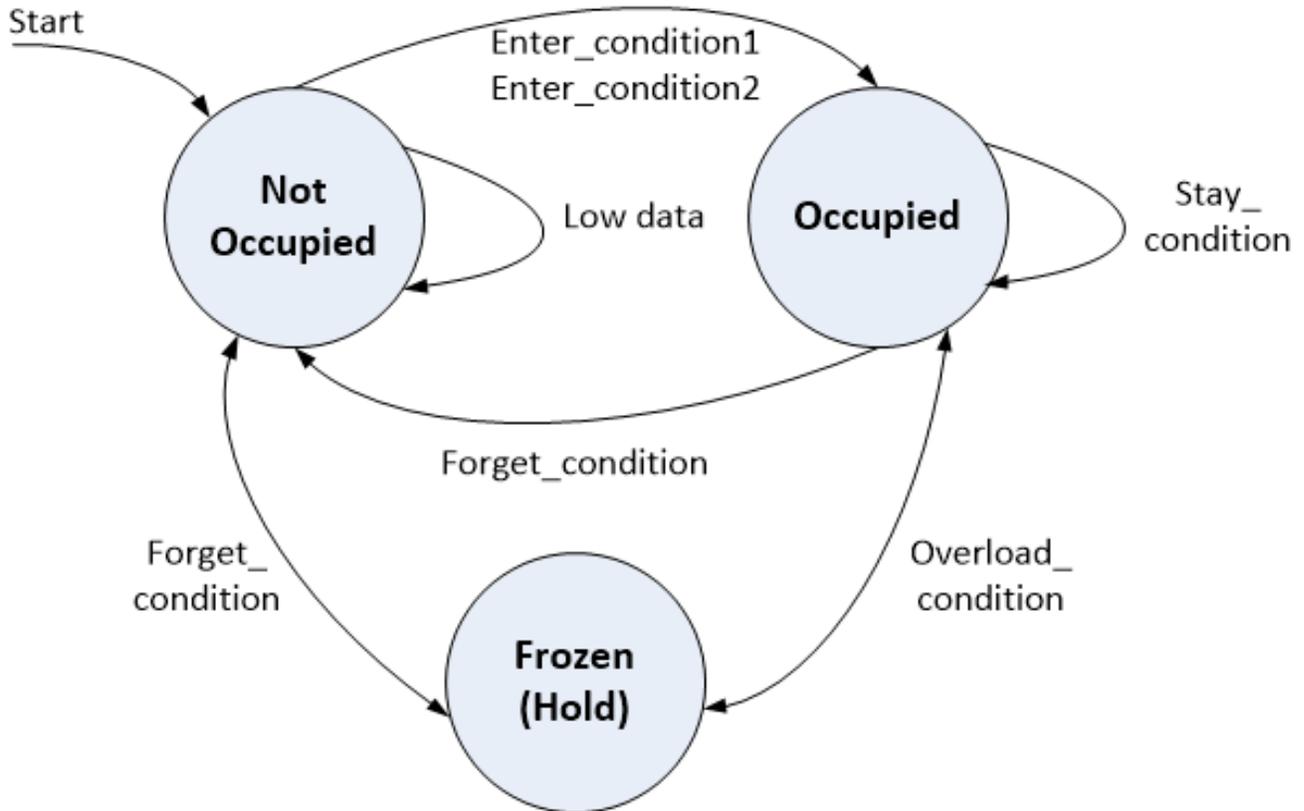


Figure 2-9. Zone State Machine

2.3.4 Output Through UART

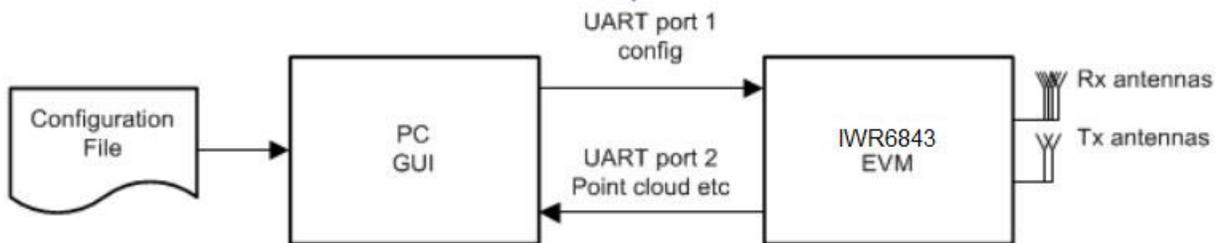


Figure 2-10. UART Communication

As shown in [Figure 2-10](#), the example processing chain uses one UART port to receive input configuration from the front end and signal-processing chain, and uses the second UART port to send out processing results for display. See the information included in the software package for details on the format of the input configuration and output results.

3 Hardware, Software, Testing Requirements, and Test Results

3.1 Required Hardware and Software

3.1.1 Hardware

The in-cabin sensing application runs on the AWR6843AOP EVM and connects to a visualization tool running on a PC via USB. For details regarding usage of this board, see the [AWR6843 Evaluation Module \(AWR6843ISKM\) Single-Chip mmWave Sensing Solution](#).

The AWR6843 core design includes:

- AWR6843 device: A single-chip, 60-GHz radar device with an integrated DSP
- Power management network using a low-dropout linear regulator (LDO) and power management integrated circuit (PMIC) DC/DC supply (TPS7A88, TPS7A8101-Q1, and LP87524B-Q1)

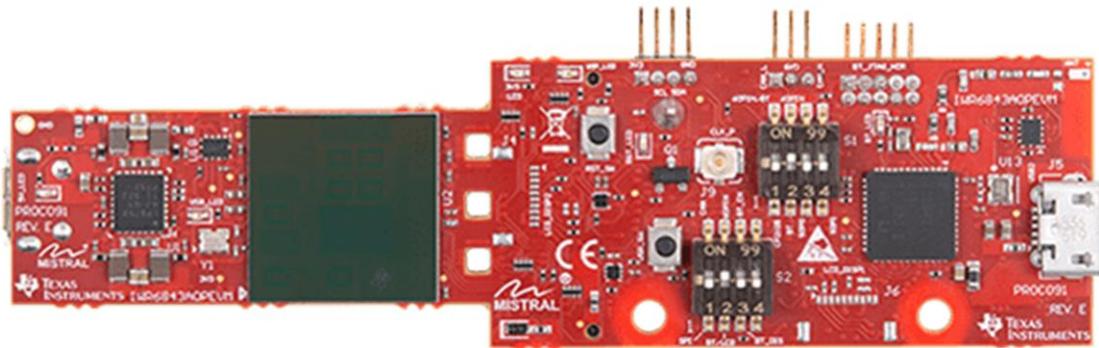


Figure 3-1. AWR6843 Intelligent mmWave Sensor Antenna-On-Package (AoP) Evaluation Module Board Image

3.1.2 Software

- The mmWave SDK can be downloaded from the [mmWave software development kit \(SDK\)](#). The installation program will also install all required tool components.
- Download the [overhead incabin occupancy detection application](#) software from the Automotive Toolbox on TI Resource Explorer (TI-Rex).
- Details on how to run the pre-built binaries and how to rebuild the demonstration application are provided in the user guide in TI-Rex.

3.2 Testing and Results

This section discusses the use case testing for child presence detection, occupancy detection, and intruder detection use cases.

3.2.1 Test Setup

Sensor Mounting Position: The sensor was mounted on the ceiling of the vehicle centered over the console separating the driver and passenger seat.

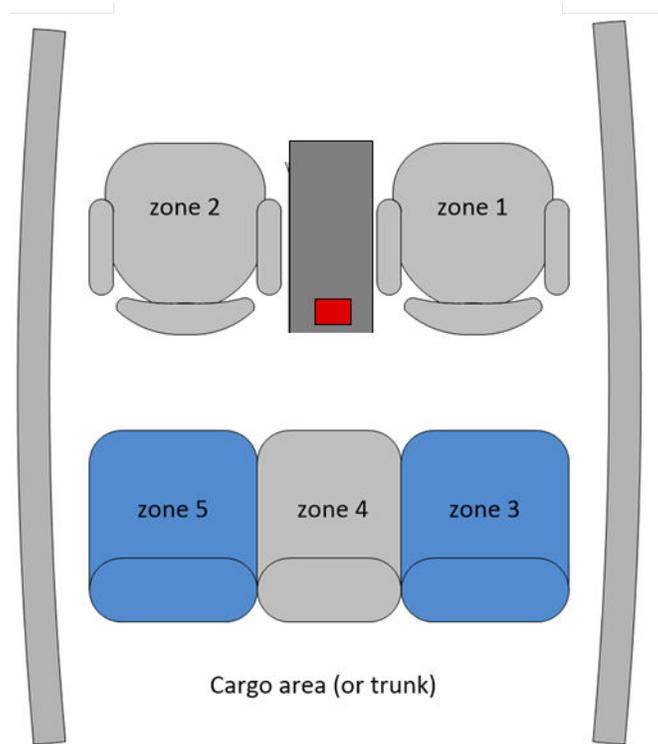


Figure 3-2. Sensor Position

Zone Definitions: Based on the mounting position of the sensor. The XYZ definitions of each zone (seat) position in the car was measured and updated in the chirp configuration file. The demo project provides example chirp configurations for each use case and supported device. These files are located in the **/chirp_configs** folder of the demo. **NOTE:** Due to variations in mounting position and vehicle geometries, users should measure and update zone definitions for their test setup in the configuration file. Details are provided in the demo user guide with the downloaded software.

3.2.2 Test Results

Child Presence Detection

To simulate a child left behind, a baby doll capable of simulated breathing was positioned in various locations in the vehicle. The results of the various positions tested are as follows:

ZONE POSITION	CAR SEAT ORIENTATION	DETECTED?
2	Rear-facing	Yes
3	Front-facing	Yes
4	Rear-facing	Yes
5	Front-facing	Yes
Footwell of 3	n/a	Yes
Footwell of 4	n/a	Yes
Footwell of 5	n/a </tr	

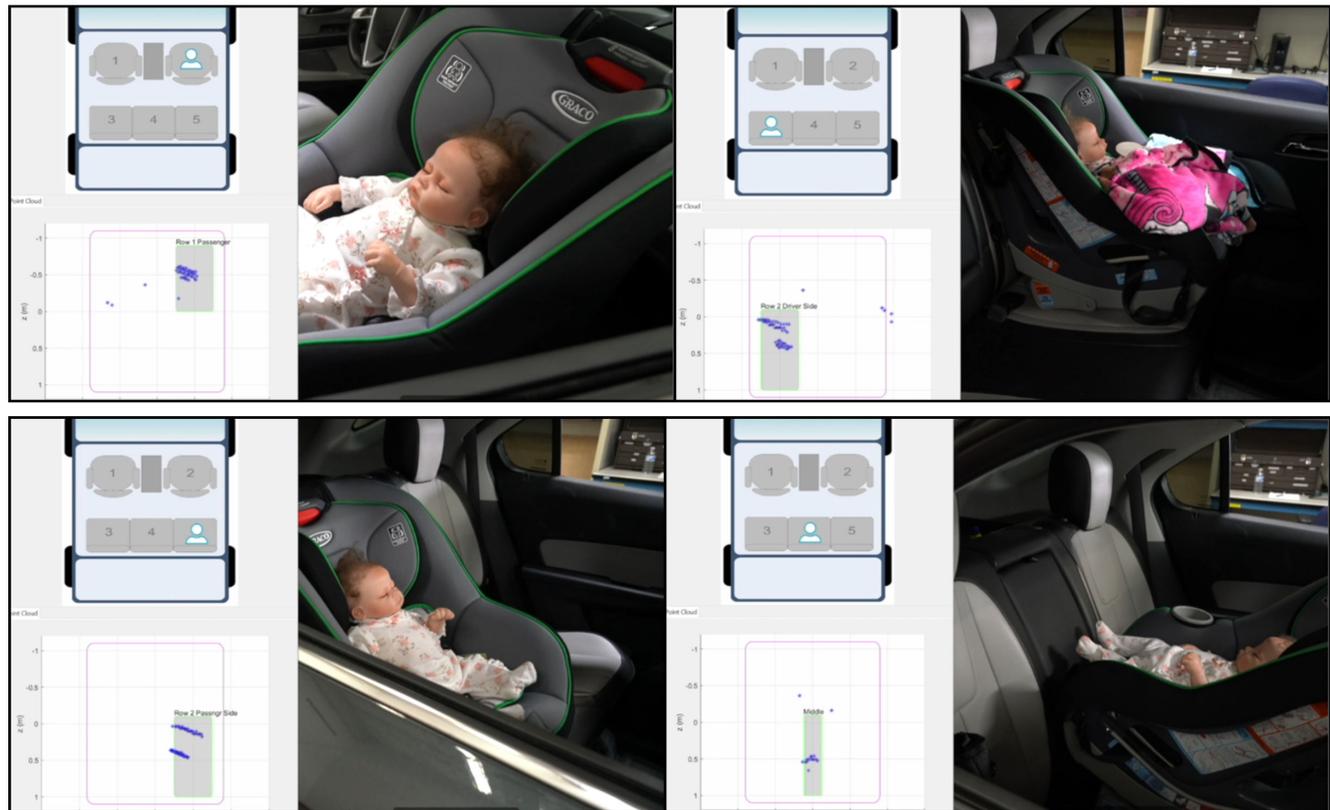


Figure 3-3. Child Presence Detection

Seat Occupancy Detection

Testing was performed for 2 person and 3 person with baby-doll use cases.

2-Person Occupancy was tested for various pairs of locations in the vehicle. The results of the various positions tested are as follows:

POSITION PAIRS	DETECTED?
1 and 2	Yes
3 and 4	Yes
1 and 4	Yes
2 and 3	Yes

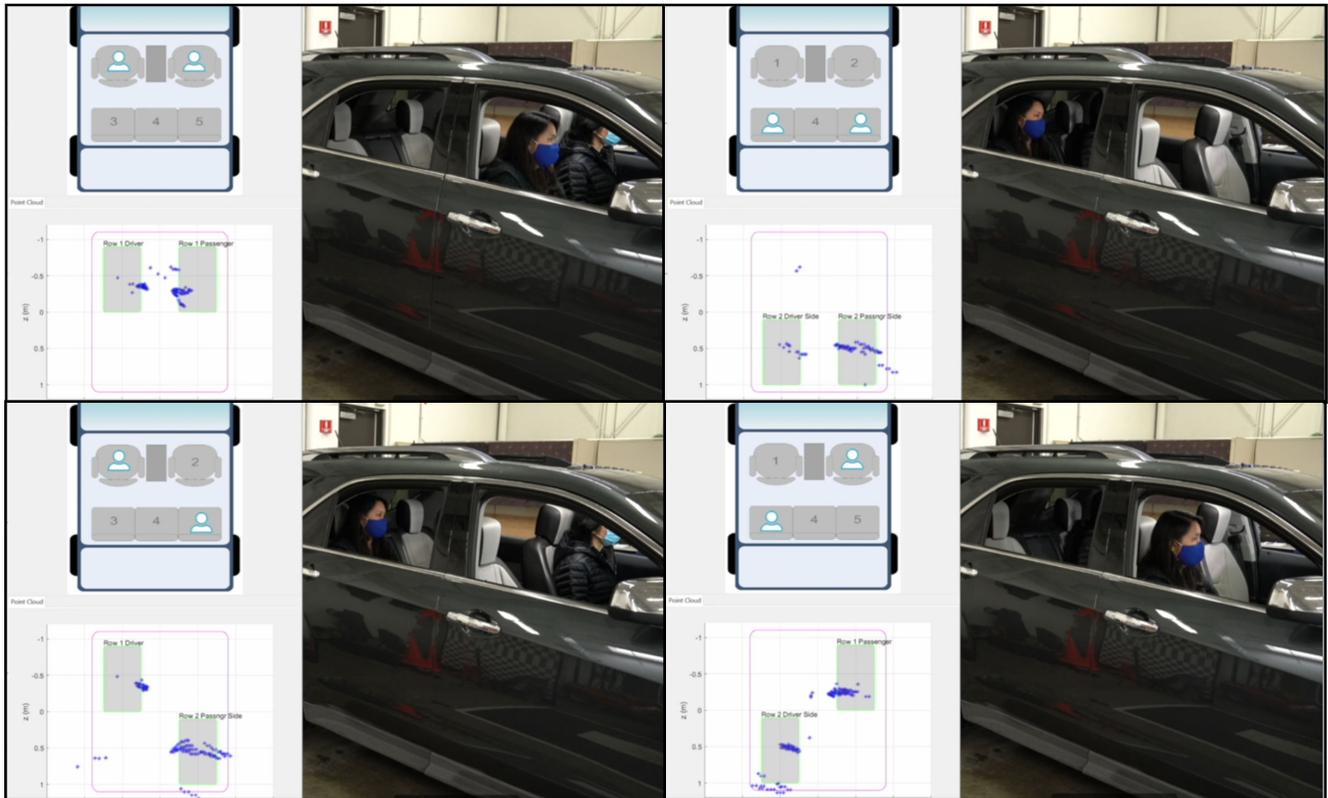


Figure 3-4. Two Person Occupancy

3-Person Occupancy with Baby Doll was tested with each adult person entering the car separately. The results test are as follows:

ACTION SEQUENCE	ZONE(S) DETECTED
A) Adult person in Zone 3 and Baby doll in Zone 5	Zones 3 and 5
B) Additional person sits in Zone 1	Zones 1, 3, and 5
C) Additional person sits in Zone 2	Zones 1, 2, 3, and 5
D) Zone 2 person exits	Zones 1, 3, and 5
E) Zone 1 person exits	Zone 3 and 5
F) Zone 3 person exits	Zone 5

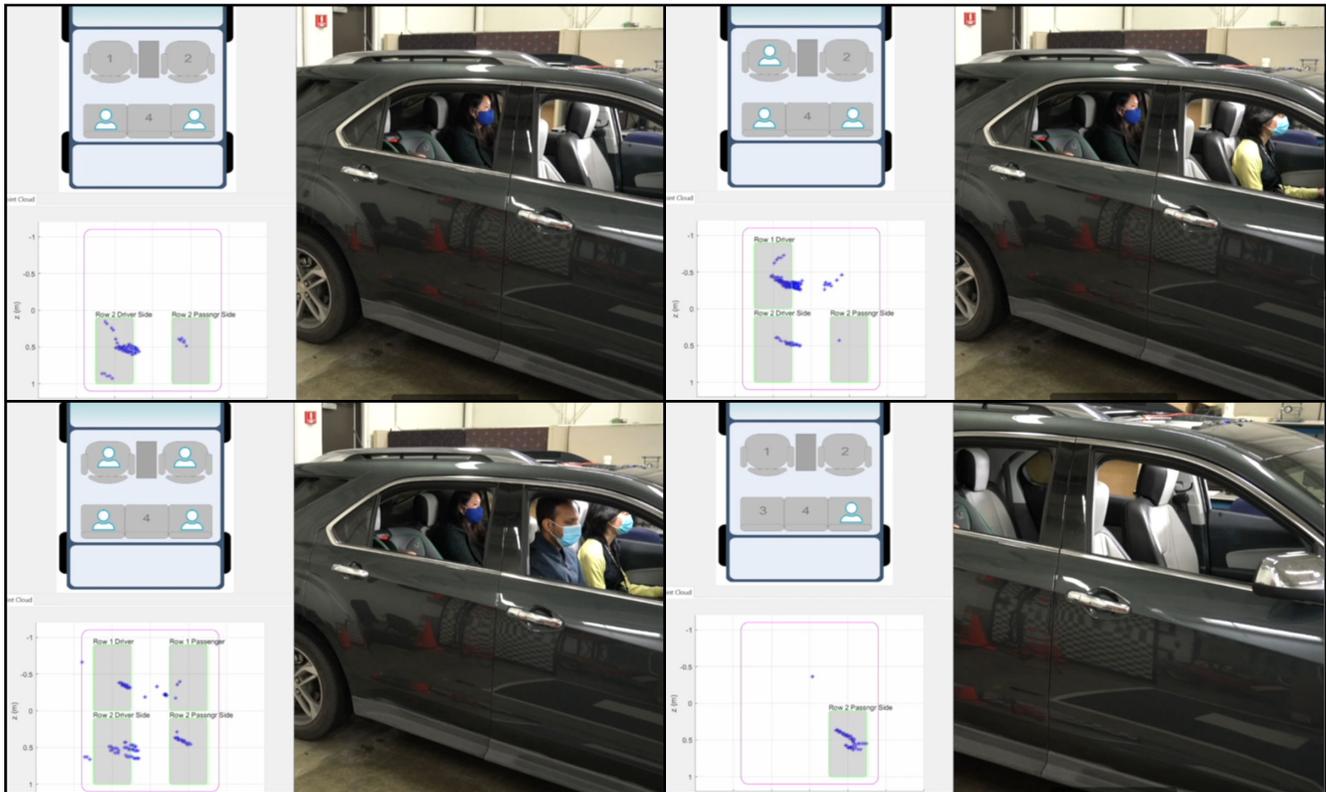


Figure 3-5. 3-Person Occupancy with Baby Doll

Intruder Detection was tested with no people/child in the car and only various inanimate objects including a book bag. Intruder zones were defined on the sides of the vehicle along the doors. The example image below shows the intruder detection functionality:

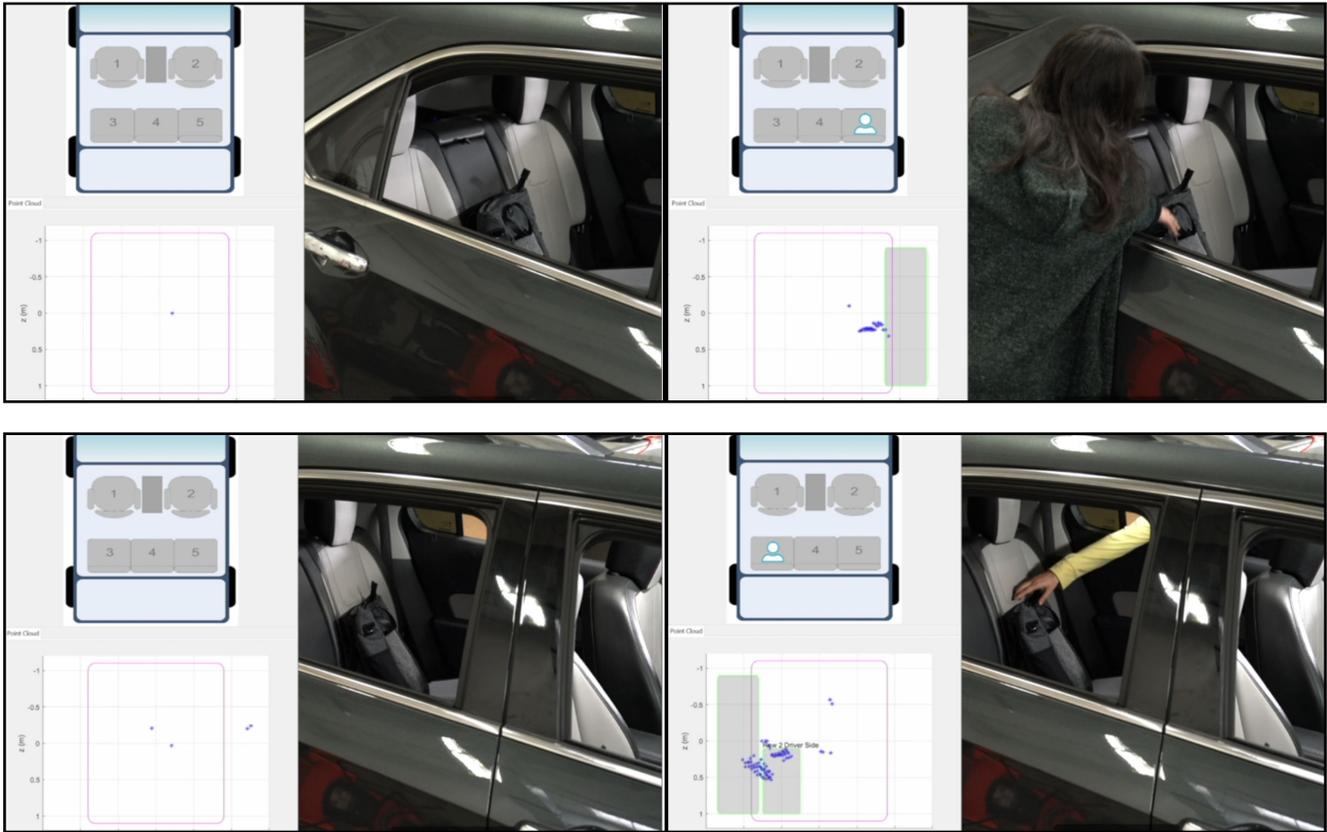


Figure 3-6. Intruder Detection Example

4 Design Files

4.1 Schematics

To download the schematics, see the design files at [IWR6843](#).

4.2 Bill of Materials

To download the bill of materials (BOM), see the design files at [IWR6843](#).

4.3 Altium Project

To download the Altium Designer® project files, see the design files at [IWR6843](#).

5 Software Files

To download the software files, see the design files at [IWR6843](#).

6 Related Documentation

1. Texas Instruments, [AWR6843AOP Data Sheet](#)
2. Texas Instruments, [AWR18xx/16xx/14xx /68xx Technical Reference Manual](#)
3. Texas Instruments, [mmWave SDK](#), tools folder
4. Texas Instruments, [Overhead Mount Vehicle Occupancy Detection](#)

6.1 Trademarks

TI E2E™ is a trademark of Texas Instruments.

ARM® and Cortex® are registered trademarks of Arm Limited.

Altium Designer® is a registered trademark of Altium LLC.

All trademarks are the property of their respective owners.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2022, Texas Instruments Incorporated