TI 设计：TIDEP-0092
使用 AWR1642 的短距离雷达参考设计

说明

TIDEP-0092 为使用 AWR1642 评估模块 (EVM) 开发短距离雷达 (SRR) 应用奠定了基础。该设计可帮助用户在其高达 80m 的视野内估算和跟踪物体的位置（在方位平面中）和速度。

资源

TIDEP-0092	设计文件夹
AWR1642	产品文件夹
TCAN1042	产品文件夹
TMP112	产品文件夹
LP87524	工具文件夹

TI E2E™ Community

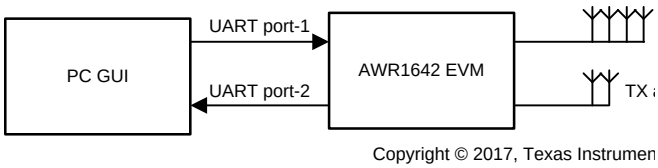
咨询我们的 E2E 专家

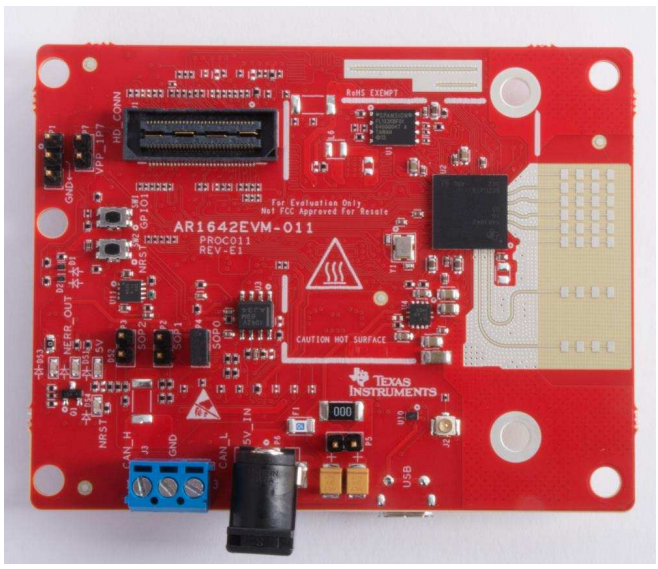
特性

- 适用于 SRR 的单芯片 FMCW 无线电探测和测距 (RADAR) 应用
- 检测
  80m 范围内的物体（例如汽车和卡车），距离分辨率为 35cm；以 4.3cm 的分辨率检测 20m 范围内的物体
- 对检测到的输出进行群集和跟踪处理
- 天线视野为 ±60º，且角度分辨率约为 15º
- 通过毫米波软件开发套件 (SDK) 提供了快速傅里叶变换 (FFT) 处理和检测源代码
- AWR1642 演示了设计
- 对雷达前端和检测配置进行了全面说明

应用

- 车道变换辅助系统 (LCA)
- 自主泊车
- 侧向来车警示系统 (CTA)
- 盲点监测系统 (BSD)





1 System Description

Autonomous control of a vehicle provides quality-of-life and safety benefits in addition to making the relatively mundane act of driving safer and less difficult. The quality-of-life features include the ability of a vehicle to park itself or to determine whether a lane change is possible and provide features like automatic cruise control—where a vehicle maintains a constant distance with respect to the car ahead of it, essentially tracking the velocity of the car in front of it. Autonomous breaking and collision avoidance are safety features that prevent accidents caused by driver inattention. These features work by observing the area in front of a car and alerting the autonomous driving subsystems if obstacles are observed that are likely to hit the car. Implementing these technologies require a variety of sensors to detect obstacles in the environment and track their velocities and positions over time.

1.1 Why Radar?

Frequency-modulated continuous-wave (FMCW) radars allow the accurate measurement of distances and relative velocities of obstacles and other vehicles; therefore, radars are useful for autonomous vehicular applications (such as parking assist and lane change assist) and car safety applications (autonomous breaking and collision avoidance). An important advantage of radars over camera and light-detection-and-ranging (LIDAR)-based systems is that radars are relatively immune to environmental conditions (such as the effects of rain, dust, and smoke). Because FMCW radars transmit a specific signal (called a chirp) and process the reflections, they can work in complete darkness and also bright daylight (radars are not affected by glare). When compared with ultrasound, radars typically have a much longer range and much faster time of transit for their signals.

1.2 TI SRR Design

The TIDEP-0092 is an introductory application that is configured for short range applications (that is to detect as many as 200 objects up to a distance of 80 m (260 feet) and track as many as 24 of them travelling as fast as 90 kph, which is approximately 55 mph). In the short range application, the AWR1642 sensor is configured as a multi-mode radar, which means that it can simultaneously track objects at 80 m while generating a rich point cloud of objects at 20 m, so that approaching vehicles and closer small objects can be detected at the same time. This reference design can be used as a starting point to design a standalone sensor for a variety of SRR automotive applications. A range of more than 80 m can be achieved with the design of an antenna with higher gain than the one included in the AWR1642.

1.3 Key System Specifications

This reference design has two sets of specifications because the radar is used as a multi-mode radar. The first specification is for the short range radar (SRR), which has a range of 80 m. The second specification is for the ultra-short-range radar (USRR), which has an effective range of only 20 m.

表 **1. Key System Specifications**

PARAMETER	SPECIFICATIONS	DETAILS
Maximum range	80 m (SRR), 2 0m (USRR)	This represents the maximum distance that the radar can detect an object representing an RCS of approximately 10 m^2.
Range resolution	36.6 cm (SRR), 4.3 cm (USRR)	Range resolution is the ability of a radar system to distinguish between two or more targets on the same bearing but at different ranges.
Maximum velocity	90 kph (SRR), 36 kph (USRR)	This is the native maximum velocity obtained using a two-dimensional FFT on the frame data. This specification will be improved over time by showing how higher-level algorithms can extend the maximum measurable velocity beyond this limit.

表 **1. Key System Specifications (continued)**

PARAMETER	SPECIFICATIONS	DETAILS
Velocity resolution	0.52 m/s (SRR), 0.32 m/s (USRR)	This parameter represents the capability of the radar sensor to distinguish between two or more objects at the same range but moving with different velocities.

2 System Overview

2.1 Block Diagram

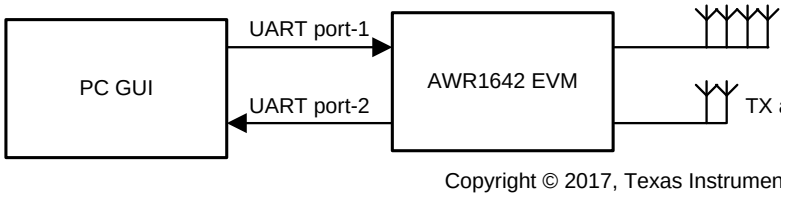


图 **1. SRR System Block Diagram**

2.2 Highlighted Products

2.2.1 AWR1642 Single-Chip Radar Solution

The AWR1642 is an integrated single-chip, frequency modulated continuous wave (FMCW) sensor capable of operation in the 76 to 81 GHz frequency band. The device is built with TI's low-power, 45-nm RFCMOS processor and enables unprecedented levels of analog and digital integration in an extremely small form factor. The device has four receivers and two transmitters with a closed-loop phase-locked loop (PLL) for precise and linear chirp synthesis. The sensor includes a built-in radio processor (BIST) for RF calibration and safety monitoring. Based on complex baseband architecture, the sensor device supports an IF bandwidth of 5 MHz with reconfigurable output sampling rates. The presence of ARM® Cortex® R4F and Texas Instruments C674x Digital Signal Processor (DSP) (fixed and floating point) along with 1.5MB of on-chip RAM enables high-level algorithm development.

2.2.2 AWR1642 Features

The AWR1642 has the following features:

1. AWR1642 radar device
2. Power management circuit to provide all the required supply rails from a single 5-V input
3. Two onboard TX antennas and four RX antennas
4. Onboard XDS110 that provides a JTAG interface, UART1 for loading the radar configuration on the AWR1642 device, and UART2 to send the object data back to the PC

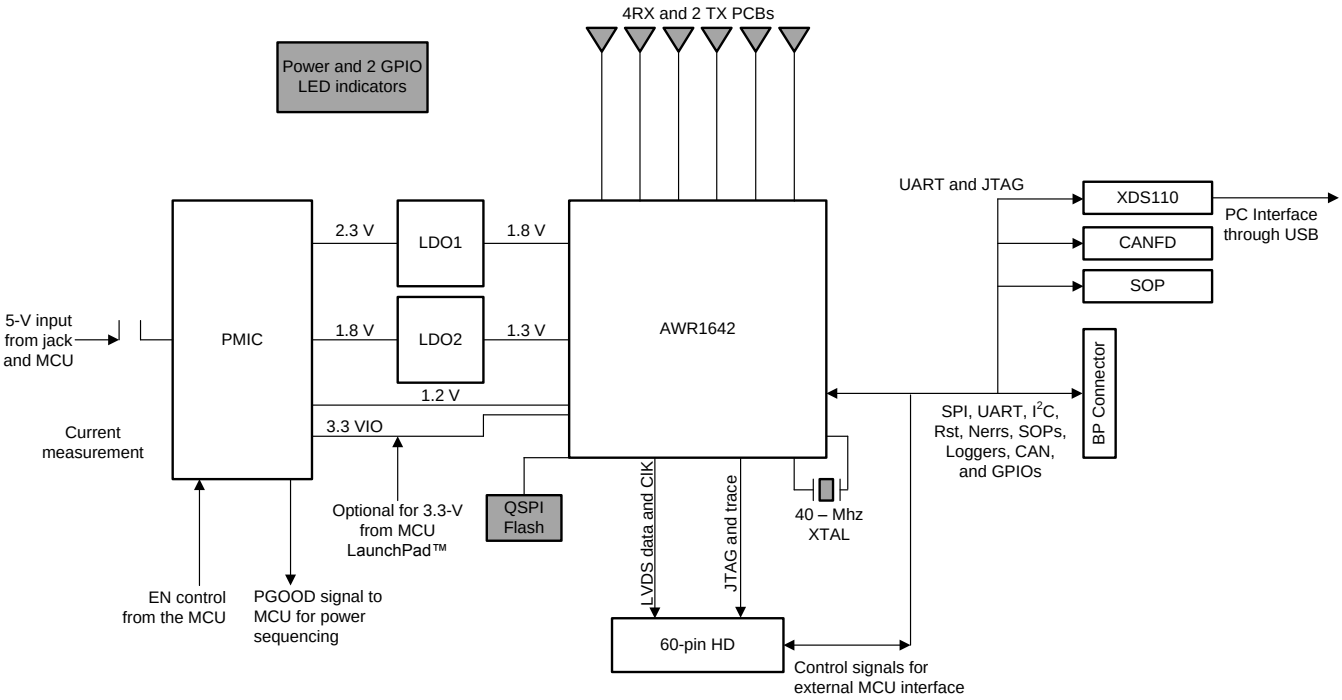


图 2. AWR1642 Block Diagram

For more details on the hardware, see *AWR1642 Evaluation Module (AWR1642BOOST) Single-Chip mmWave Sensing Solution*. The schematics and design database can be found in the following documents: *AWR1642 Evaluation Board Design Database* and *AWR1642BOOST Schematic, Assembly, and BOM*.

2.3 System Design Theory—Chirp Configuration

2.3.1 Antenna Configuration

The TIDEP-0092 uses four receivers and the two transmitters in two different chirp configurations. The first configuration (SRR) uses a simple non-multiple-input and multiple-output (MIMO) configuration with only TX1 transmitting. The second configuration (USRR) uses a time division multiplexed MIMO configuration (that is, alternate chirps in a frame transmit on TX1 and TX2 respectively.). The MIMO configuration synthesizes an array of eight virtual RX antennas, as shown in 图 3. This technique improves the angle resolution by a factor of two (compared to a single TX configuration).

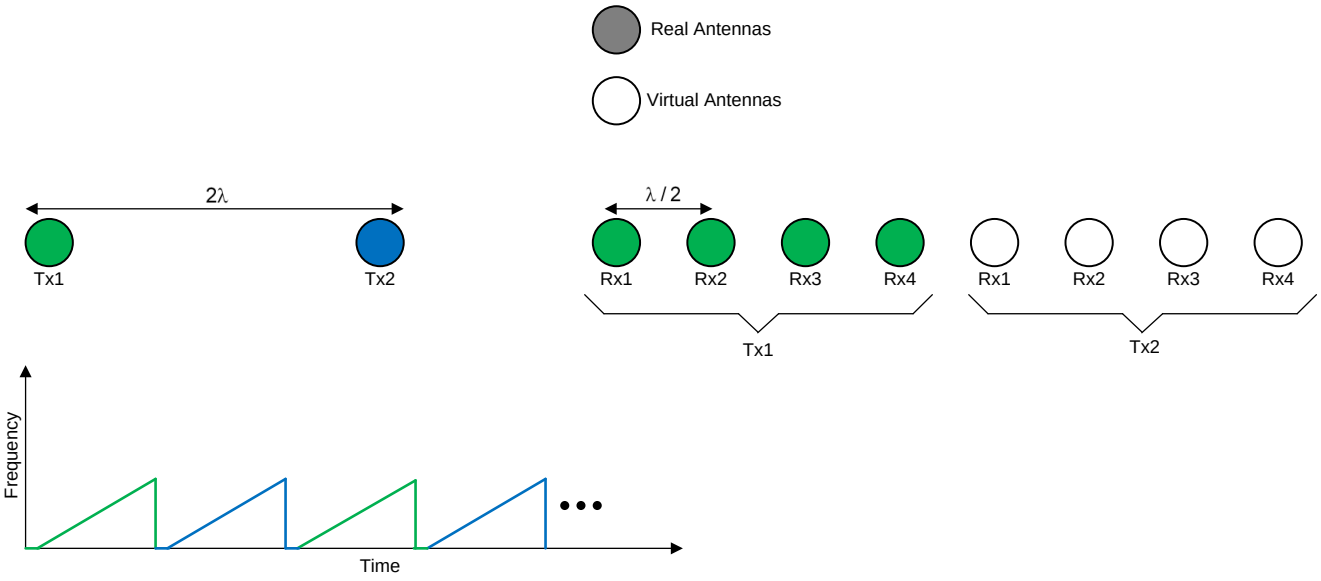


图 3. MIMO Antenna Configuration

2.3.2 Chirp Configuration and System Performance

To achieve the specific SRR use case with a visibility range of approximately 80 m and memory availability of AWR1642, the chirp configuration in 表 2 is used.

表 **2. Chirp Configuration**

PARAMETER	SPECIFICATIONS
Idle time (µs)	3 (SRR), 7 (USRR)
ADC start time (µs)	3 (SRR), 5 (USRR)
Ramp end time (µs)	56 (SRR), 87.3 (USRR)
Number of ADC samples	256 (SRR), 512 (USRR)
Frequency slope (MHz/µs)	8 (SRR), 42 (USRR)
ADC sampling frequency (ksps)	5000 (SRR), 6250 (USRR)
MIMO (1→yes)	0 (SRR), 1 (USRR)
Number of chirps per profile	128 (SRR), 64 (USRR)
Effective chirp time (usec)	51 (SRR), 82 (USRR)
Bandwidth (MHz)	409 (SRR), 3456 (USRR)
Frame length (ms)	7.3 (SRR), 6.03 (USRR)
Memory requirements (KB)	512 (SRR and USRR)

The 表 2 configuration is selected to achieve the system performance shown in 表 3. The primary goal was to achieve a maximum distance of about 80 m. Note that the product of the frequency slope and the maximum distance is limited by the available IF bandwidth (6.25 MHz for the AWR1642). Thus, a maximum distance of 85 m locks down the frequency slope of the chirp to about 8 MHz/µs. See *Programming Chirp Parameters in TI Radar Devices* for more details. The choice of the chirp periodicity is a trade-off between range resolution and maximum velocity. This design uses a range-resolution of about 0.3 m, which leaves a native maximum velocity of about 55 kph (1) . For details on the connection between the system performance and the chirp parameters, see *Programming Chirp Parameters in TI Radar Devices*. Through high-level algorithms, the maximum unambiguous velocity that can be detected is 90 kph.

A larger maximum distance translates to a lower range resolution (due to limitations on both the L3 memory and the IF bandwidth). A useful technique to work around this trade-off is to have multiple configurations with each tailored for a specific viewing range. For example, it is typical to have the SRR radar alternate between two modes: a low resolution mode targeting a larger maximum distance (such as 85 m with 0.3-m resolution) and a high resolution mode targeting a shorter distance (such as 20 m with a 4-cm resolution). This multi-mode capability is implemented in the current SRR design.

(1) Though not implemented in the current SRR design, note that there are several approaches that can improve the maximum detectable velocity several multiples beyond this native maximum.

表 **3. System Performance Parameters**

PARAMETER	SPECIFICATIONS
Range resolution (m)	0.36 (SRR), 0.043 (USRR)
Maximum distance (m)	80 (SRR), 20 (USRR)
Native maximum velocity (kph)	90 (SRR), 36 (USRR)

2.3.3 Configuration Profile

To meet the requirements for both USRR and SRR use cases, this reference design makes use of the 'advanced frame config' application programming interface (API) of the AWR1642 device. This API allows the construction of a frame consisting of multiple subframes, with each subframe being tuned to a particular application. Such a design is referred to as multi-mode radar. Each of these subframes tune to one application. In the case of the SRR configuration, use two subframes. One subframe is dedicated to the USRR context and the other to the SRR context.

The frame configuration utilizes the 'advanced frame config' API to generate two separate subframes: the SRR subframe and the USRR subframe (with the subframes being named after their principle design requirement). 图 4 shows the frame configuration.

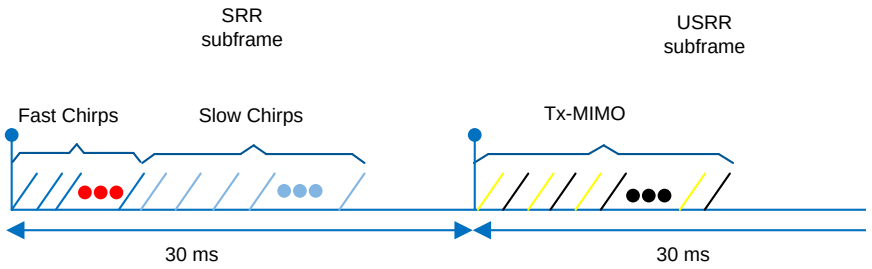


图 4. Frame Configuration

注: The design uses a fixed profile and changing the parameters requires the user to update the code. The main configuration header is */common/srr_config_consts.h*. Find the SRR configuration in */common/srr_config_chirp_design_SRR80.h* and the USRR configuration is available in */common/ srr_config_chirp_design_USRR20.h*. The design (and intent) of the two subframes are provided as follows.

- *SRR subframe*

  This subframe consists of two kinds of chirps (fast chirps and slow chirps). Both fast and slow chirps have the same slope; however, the slow chirp has a slightly higher 'chirp repeat periodicity' than the fast chirp. As a result, the slow chirps (when processed after 2D fast-Fourier transform (FFT)) have a lower maximum unambiguous velocity as compared to the fast chirps. Note that the fast and slow chirps do not alternate; instead, the fast chirp is repeated a certain number of times, followed by the slow chirp, which is again repeated an equal number of times.

  The purpose of this chirp design is to use the two separate estimations of target velocity from the 'fast chirp' and the 'slow chirp' along with the 'Chinese remainder theorem' to generate a consistent velocity estimate with a much higher max-velocity limit.

- *USRR subframe*

  This subframe consists of two alternating chirps. Each chirp utilizes one of the two Txs available on the AWR1642 device. Combined processing of this subframe allows the generation of a virtual Rx array of eight Rx antennas, which consequently has better angular resolution (approximately 14.3°) than the 4 Rx antenna array.

2.3.4 Data Path

The block diagram in 图 5 shows the processing data part to the SRR application.

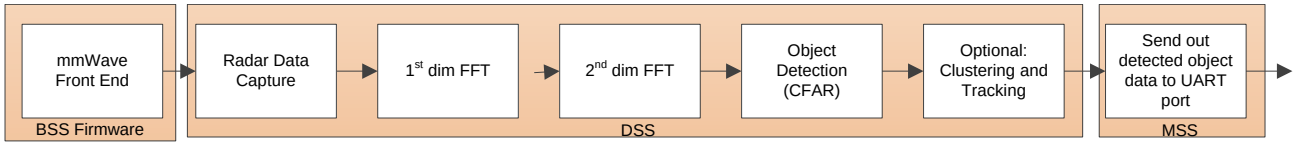


图 5. SRR Data Path or Processing Chain

2.3.5 Chirp Timing

图 6 shows the timing of the chirps and subsequent processing in the system.

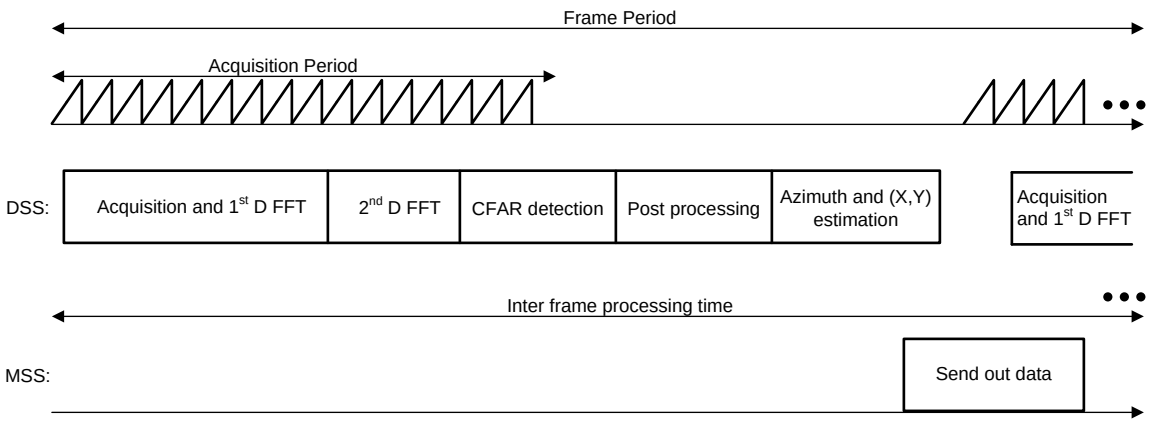


图 6. Top Level Data Path Timing

图 6 shows the data path processing, which is described as follows.

- The RF front end is configured by the BIST subsystem (BSS). The raw data obtained from the various front end channels is taken by the C67x DSP subsystem (DSS) for processing.
- Processing during the chirps as seen in 图 6 consists of:
  - 1D (range) FFT processing performed by the C674x that takes input from multiple receive antennae from the ADC buffer for every chirp (corresponding to the chirping pattern on the transmit antennae)
  - Transferring transposed output into the L3 RAM by enhanced direct memory access (eDMA)
- Processing during the idle or cool down period of the RF circuitry following the chirps until the next chirping period, shown as *Inter frame processing time* in 图 6. This processing consists of:
  - 2D (velocity) FFT processing performed by C674x that takes input from 1D output in L3 RAM and performs FFT to give a (range, velocity) matrix in the L3 RAM. The processing also includes the CFAR detection in Doppler direction. CFAR detection in range direction uses the mmWave library.
  - Peak grouping (for both Doppler and range) for the SRR subframe and Doppler for the USRR subframe
  - Direction of arrival (azimuth) estimation to map the X-Y location of object
  - Additional pruning based on the SNR and the 2D-FFT magnitude of the object to avoid ground clutter
  - Clustering of detected objects using the dBScan algorithm for both SRR and USRR
  - Tracking of clusters using an extended Kalman filter for the SRR

For more details on the application flow and processing, see *mmWave SDK User's Guide*.

Clustering

Clustering is performed using the dBscan algorithm [9] which is applied on the detected objects of both the SRR and the USRR subframes. The output of the clustering algorithm is the mean location of a cluster and its dimensions (assuming the cluster is a rectangle).

For the USRR subframe, the clustering output is sent as is to the graphical user interface (GUI). USRR clusters allow the grouping of dense point clouds to rectangles. In cross-traffic scenarios, these clusters can be used to identify vehicles crossing the field of vision (FoV) of the radar.

For the SRR algorithm, the clustering output is used as the basis for the input to the tracking algorithm. The strongest object in the cluster is provided as the representative object to the tracking algorithm. The intention here is essentially reduce the number of objects provided to the tracking algorithm and introduce hysteresis, so that the trackers only track strong reflectors, and do not switch between adjacent reflectors.

Tracking

The tracker is a fairly standard Extended Kalman Filter (EKF with four states [x, y, v_x, v_v] and three inputs [r, v, sin(θ)] or range, relative velocity, and sin of the azimuth). Compute the associated variances for the inputs using the associated signal-to-noise ratio (SNR) for each input. Use the Cramer-Rao lower bound (CRLB) formula for frequency variance to convert the SNR to a variance. The variance is lower-bounded by the resolution of the various inputs.

While the tracker is quite standard for an EKF, there are two functions in the tracker that can be modified based on the requirements. The first is the ‘track initialization function’ (initNewTracker), where the initial parameters of the track are populated. In the SRR design, the assumption is that the velocity of the object is only along the longitudinal axis. In other words, a vehicle with a relative velocity v is assumed to be travelling towards the radar with $v_x = 0$, and $v_v = v$, which works well in long-range highway traffic, but is less effective in cross-traffic situations.

The second function is the ‘data association function’ (isTargetWithinDataAssociationThresh), which associates new measurements with existing tracks. Association essentially requires assumptions on the movement of objects in the scene, including maximum velocities, directions of motion, accelerations when close to the radar, and so forth.

eDMA configuration

Large-scale data movement between memories is accomplished in both the out-of-box (OOB) demo of the mmWave SDK and the SRR using the eDMA. Using the eDMA is more efficient than using the processor to move data because, while the data movement is being completed, the DSP can continue to process data. The major data transfers necessary in the OOB demo include:

- Movement of ADC data from the ADC buffer to the DSP L2/L1 memory
- Movement of 1D-FFT data from the DSP L1/L2 memory to the L3 memory to form the ‘radarcube matrix’
- Fetching the 1D-FFT data from L3 to do 2D FFT
- Movement of the ‘sumAbs’ array from the L2/L1 to L3 to form the detection matrix
- Movement of slices of the detection matrix in L3 to L2/L1 to do detection
- Movement of 1D-FFT data from L3 to do angle estimation (after 2D-DFT)

Most of the eDMAs work on a ping-pong buffer, which means that as the ping buffer is being filled, the pong buffer can be used by the DSP for processing. The eDMA configuration in the SRR demo is very similar to the eDMA configuration on the OOB demo. The differences relate to the processing of two different subframes, and the processing of the SRR subframe.

In order to process the two different subframes, the SRR demo simply doubles the number of number of assigned eDMAs so that each subframe has its own subset of eDMAs to perform the necessary transfers. This method will not scale as the number of subframes increase, and it is recommended that if more than 3 subframes are to be programmed, then one should reprogram the eDMAs after a frame is processed (as is done in the OOB demo).

To process the 'max velocity enhancement' subframe, the eDMAs listed as 2, 3, and 6 must be changed, so that they transfer twice as many chirps.

Memory allocation

The AWR1642 (PG 2.0) has the following memories on the DSP.

1. L3 RAM of 768kB (PG 1.0 has 640 kB)
2. L2 RAM of 256kB
3. L1 RAM of 32kB each

The R4F has 512kB of code and data RAM. In the SRR design, the R4F is used only for configuration and for the universal asynchronous receiver/transmitter (UART) or loop-voltage differential-signaling (LVDS) communication. The RF4 memory consumption and allocation are not of any concern to the design. The DSP, therefore, is the main consumer of memory, for which the allocations are discussed below.

Of the 32KB for the available L1P RAM and L1D RAM, half (16kB) of the L1P RAM and half (16kB) of the L1D RAM are reserved for code and data storage. The remaining are used as cache. The code stored in L1P is typically algorithms like the FFTs or the CFAR, or some of the kernels of more complex algorithms like clustering and tracking. Storing these in L1 allows faster execution of these kernels as well as saving some space (in the case of this design, approximately 16kB) in the remaining memories. The L1D is used as fast RAM for certain commonly-used buffers.

L2 RAM is used for code-storage as well as scratch buffers and state information for the different algorithms, as well as the configuration information of the SRR. The scratch buffer is 49KB, of which about 20kB is free. Additionally, the L2 has approximately 70kB free. Therefore, In total. there is approximately 90kB available in L2 for the code and data.

L3 RAM is used for storing the 'Radar cube' as well as the 'Detection Matrix'. Some single-use initialization code is also stored in L3. When the code is used, L3 is cleared. The radar cube consumes 512kB and the detection matrix takes 32kB. The free space in L3 is approximately 224kB.

Processing radar signals require a large number of scratch buffers for each step each of the processing stages whether that be 1D-FFT, 2D-FFT, 3D-FFT, detection, angle estimation, or so forth. Efficiently using the 1.0MB available is important and is aided by the fact that the memory assigned to a scratch buffer used in a previous stage can be re-used in the current stage. In other words, memory locations can be overlaid for efficient memory utilization.

There are two subframes per frame, and both subframes are processed separately and in sequence; therefore, nearly every scratch buffer memory location can be overlaid between the two. The creation of the scratch buffer pointers for the two subframes is done in MmwDemo_dataPathConfigBuffers.

mmWave configuration: minimal mode with isolation

In contrast to the OOB demo, where the mmWave module runs in both DSS and MSS and maintains synchronization between them, in the SRR design, mmWave is configured to run in 'minimal mode with Isolation' on the master subsystem (MSS) only. The DSS does not include the mmWave or the mmWavelink modules.

注: The mmWave API is a set of high-level APIs used to program the radar and the mmWavelink is a set of lower-level APIs that are used to program the radar. For further detail, see *AWR1642 mmWave sensor: 76–81-GHz radar-on-chip for short-range radar applications*.

What this configuration means is that the MSS and the DSS exist as separate processors communicating only through explicit calls by their applications. Information of the fixed-chirp configuration used by the SRR is stored in *srr\common\srr_config_consts.h* and is used by both the MSS (to program the radar) and by the DSS (to process the radar signals). The DSS is then only capable of processing the radar signals (but unable to control the radar). The MSS is solely responsible for configuring the chirp and starting the sensor.

This divison of labor between the two processor necessitates that the chirp configuration must be known in advance and cannot be changed on the fly. However, this restriction is not absolute and it is possible to use the mailbox to convey messages from the MSS to the DSS for minor variations of the chirp configuration, which will be demonstrated in a forthcoming update to the reference design.

Maximum velocity disambiguation in SRR subframe

The SRR subframe achieves a maximum unambiguous velocity of 90 kph by using signal processing techniques that help disambiguate velocity. This method works by using two different estimates of velocity from the two kinds of chirps (‘fast chirps’ and ‘slow chirps’) transmitted in the SRR subframe. If the two velocity estimates do not agree, then velocity disambiguation is necessary. To disambiguate, it is necessary to rationalize the two velocity measurements and find out the disambiguation factor, k. If the native maximum unambiguous velocity of the 'fast chirp' is v_f , and that of the 'slow chirp' is v_s, then after the disambiguation process, the disambiguated velocity would be $2kv_f + v$, where v is the native estimated velocity from the ‘fast chirps’.

The disambiguation process works by using the 'fast chirp' velocity to compute different disambiguated velocity hypotheses. This computation works by taking the 'fast chirp' velocity and adding $2kv_f$, where $k \in \{-1,0,1\}$ (an unwrapping process on the velocity estimate). These hypotheses are then converted to indices of the 'slow chirp' by finding the equivalent estimated velocities in the 'slow chirp' configuration (essentially, undoing the unwrapping using v_s as the maximum unambiguous velocity).

If the index corresponding to one of the hypotheses has significant energy, then that hypothesis is considered to be valid. Disambiguation of up to 3x of the naive max-velocity is possible with this method; however, testing has only been done up to 90 kph.

3 Hardware, Software, Testing Requirements, and Test Results

3.1 Required Hardware and Software

The AWR1642 BoosterPack™ from Texas Instruments is an easy-to-use evaluation board for the AWR1642 mmWave sensing devices.

The short-range radar application runs on the AWR1642 EVM and connects to a visualization tool running on a PC connected to the EVM over USB.

For details regarding usage of this board, see *AWR1642 Evaluation Module (AWR1642BOOST) Single-Chip mmWave Sensing Solution*.

For details regarding the visualization tool, see *mmWave SDK User's Guide*.

3.1.1 Hardware

The AWR1642 core design includes:

- AWR1642 device: A single-chip, 77-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)
- The EVM also hosts a device to assist with onboard emulation and UART emulation over a USB link with the PC

3.1.2 Software and GUI

The associated software is hosted as the *mmWave Demo* in mmWave SDK distribution.

The GUI for the SRR TI design is located in the *\ti\mmwave_sdk_01_01_00_02\packages\ti\demo\xwr16xx\gui* path and is written in the Matlab programming language. As such, the GUI requires a specific Matlab runtime engine (v8.5.1 32-bit) to properly install. The executable provided only works in a Windows operating system.

The first tab allows the user to configure the demo (see 图 7). In the first tab (UART port options), UART ports can be configured (based on the device manager settings). If the radar is already running, there is no need to load the configuration, so the ‘reload configuration’ is not necessary when restarting the GUI; otherwise, set it to start the radar.

The second tab configures the ranges on the GUI. Note that it only configures the GUI and not the radar.

The final tab has a list of record and replay options that allow the user to record and then replay UART recordings.

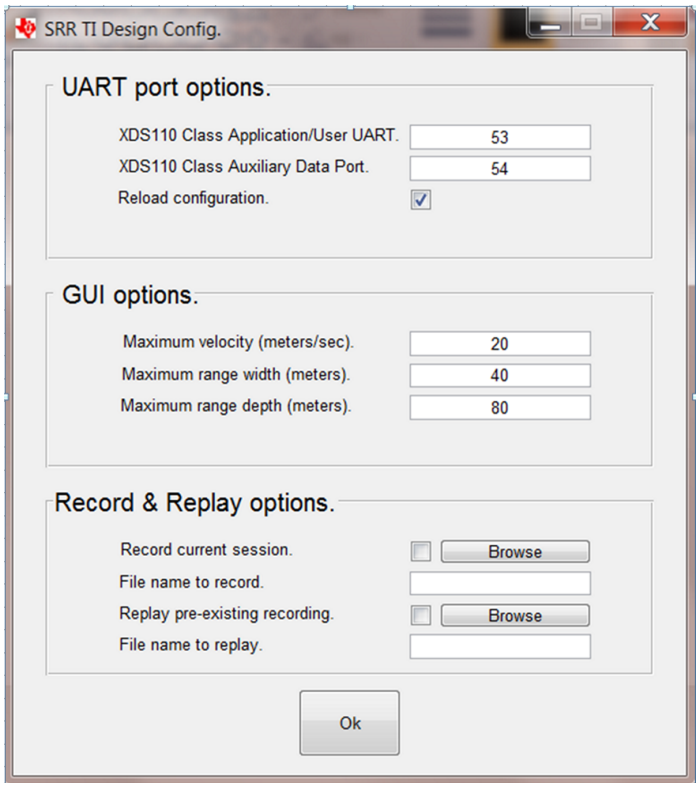


图 **7. GUI Configuration**

The default GUI starts as soon as the user presses the *OK* button. See 图 8 for a screenshot of the GUI with the different components labelled.

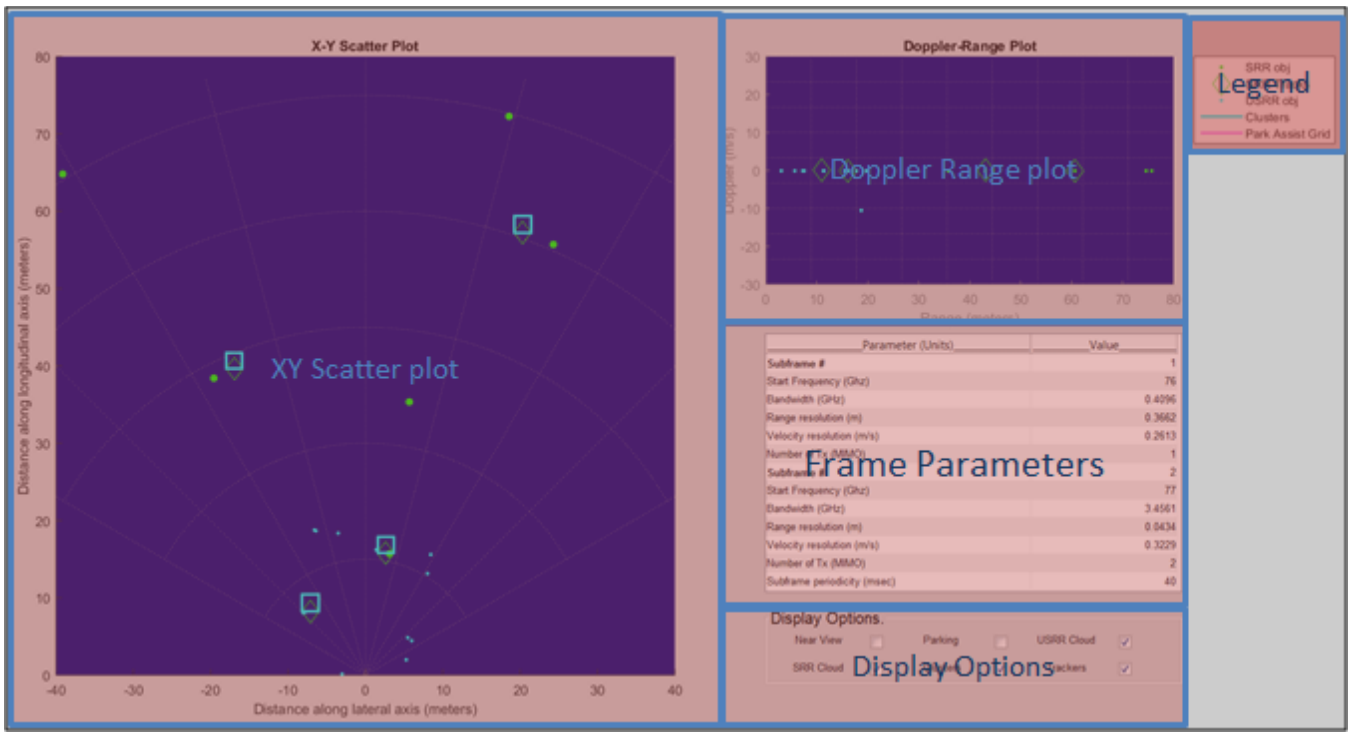


图 **8. GUI**

The Matlab GUI consists of five components.

- X-Y scatter plot – Displays the positions of the point clouds, the tracks, and the cluster.
- Doppler range plot – Displays the Doppler-range coordinates of the point cloud and the clusters.
- Frame parameters – Displays the (fixed) parameters of the SRR reference design.
- Legend – Description of the different kinds of points being displayed on the screen.
- Display options – The features of the SRR reference design can make the display crowded and sometimes difficult to understand. As such, different types of the point cloud can be enabled and disabled at the user's discretion during demonstration. For example, during long stretches of open highway, the user may only need to see the trackers. In heavy traffic, the USRR cloud makes more sense. In cases where there is cross traffic, the cluster output generates a better display of the traffic crossing over. In general, TI recommends to leave on the trackers and the parking grid.

3.2 Testing and Results

3.2.1 Test Setup

表 4 summarizes the time complexity of key building blocks in the processing chain (running on a C674x DSP hosted in the AWR1642).

表 **4. Software Algorithm Processing Characteristics**

ALGORITHM	CYCLES	TIMING (AT 600 MHz)	SOURCE AND FUNCTION NAME
128-point FFT (16 bit)	516	0.86 µs	DSPLIB (DSP_fft16x16)
256-point FFT (16 bit)	932	1.55 µs	DSPLIB (DSP_fft16x16)
128-point FFT (32 bit)	956	1.59 µs	DSPLIB (DSP_fft32x32)
Windowing (16 bit)	0.595N + 70	0.37 µs (for N=256)	mmwavelib (mmwavelib_windowing16x16)
Windowing (32 bit)	N + 67	0.32 µs (for N=128)	mmwavelib (mmwavelib_windowing16x32)
Log2abs (16 bit)	1.8N + 75	0.89 µs (for N=256)	mmwavelib (mmwavelib_log2Abs16)
Log2abs (32 bit)	3.5N + 68	0.86 µs (for N=128)	mmwavelib (mmwavelib_log2Abs32)
CFAR-CA detection	3N + 161	0.91 µs	mmwavelib (mmwavelib_cfarCadB)
Max of a vector of length 256	70	0.12 µs	DSPLIB (DSP_maxval)
Sum of complex vector of length 256(16 bit I,Q)	169	0.28 µs	—
Multiply two complex vectors of length 256 (16 bit)	265	0.44 µs	—

This system was used in field tests and a few observations are shown in 节 3.2.2, where a small car is continuously visible up to 80 m of distance and a motorcycle is detected up to 50-m away.

3.2.2 Test Results

The following results were obtained by performing field tests on the SRR system where a single small vehicle and motorcycle were driven away from the system while the results were being logged.

The second test involved driving on a highway and observing the max-velocity improvement. Natively, the SRR chirp supports a maximum velocity of 55 kph; however, in the test below, observe that it is able to track cars traveling as fast as 100 kph.

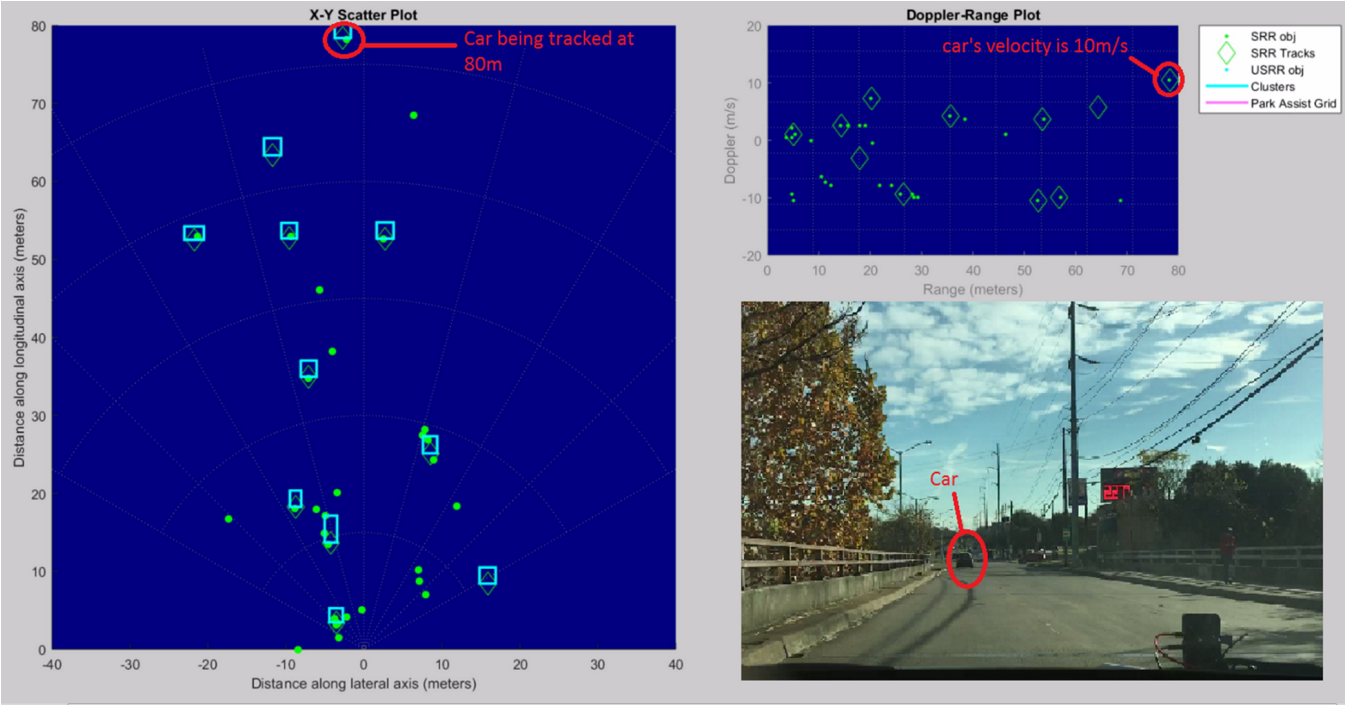


图 9. Tracking a Car at 80 m on Highway

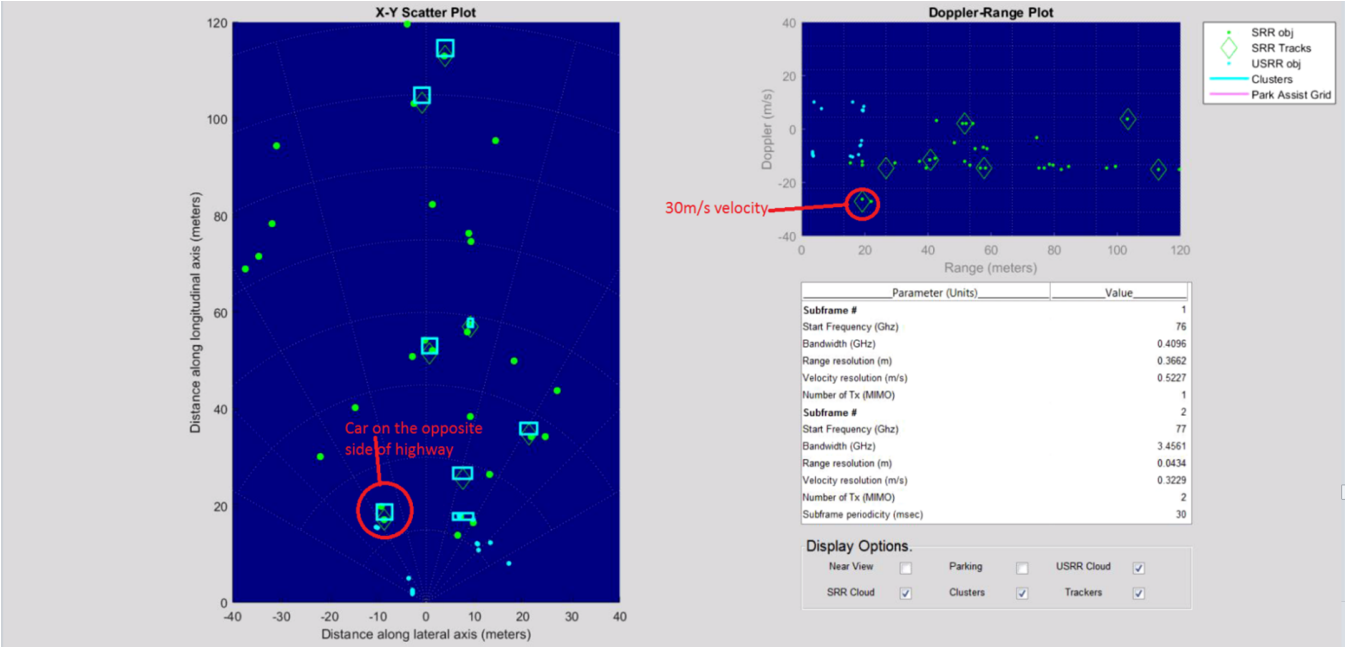


图 10. Max Velocity Test (Tracking a Vehicle Moving at 100 kph)

4 Software Files

To download the software files, see the design files at TIDEP-0092.

5 Related Documentation

1. Texas Instruments, *AWR1642 Evaluation Module (AWR1642BOOST) Single-Chip mmWave Sensing Solution*
2. Texas Instruments, *Programming Chirp Parameters in TI Radar Devices*
3. Texas Instruments, *AWR1642 Single-Chip 77- and 79-GHz FMCW Radar Sensor*
4. Texas Instruments, *AR14xx/16xx Technical Reference Manual*
5. Texas Instruments, *AWR1642 Evaluation Board Design Database*
6. Texas Instruments, *AWR1642BOOST Schematic, Assembly, and BOM*
7. Texas Instruments, *mmWave SDK User's Guide*
8. Texas Instruments, *AWR1642 mmWave sensor: 76–81-GHz radar-on-chip for short-range radar applications*

5.1 商标

6 About the Author

ANIL MANI is a system engineer at Texas Instruments where he is responsible for designing algorithms for the radar processing chain. Anil has been with TI since 2008 and has been involved in the design of various products related to wireless communication.

修订版本 B 历史记录

注：之前版本的页码可能与当前版本有所不同。

Changes from A Revision (May 2017) to B Revision **Page**

- 从 特性 下的列表项中删除了“摩托车”；添加了“卡车”。 1
- 在 特性 中增加内容：“以 4.3cm 的分辨率检测 20m 范围内的物体”。 1
- 已添加 向 特性 添加了列表项：“对检测到的输出进行群集和跟踪处理” 1
- 已添加 more specific details and values regarding the TI SRR design in 1.2 节 2
- Updated values and added specifications for both SRR and USRR in *Key System Specifications* 2
- Updated values and added specifications for both SRR and USRR in 表 2 7
- 已更改 stated primary goal of a maximum distance from 85 m to 80 m 7
- 已更改 stated available IF bandwidth from 5 MHz to 6.25 MHz for the AWR1642 7
- 已更改 the stated native maximum velocity of 30 mph to 55 kph 7
- 已更改 the stated maximum unambiguous velocity for detection from 90 mph to 90 kph 7
- Updated to specify that multi-mode capability has (since release) been implemented in the SRR design 7
- Updated values and added specifications for both SRR and USRR in 表 3 7
- Updated all content in 节 2.3.3; added 图 4 8
- 已添加 details to 'Peak grouping' list item: (for both Doppler and range) for the SRR subframe and the Doppler for the USRR subframe 10
- 已添加 content for "Clustering" 10
- 已添加 content to "Tracking" 11
- 已添加 content for "eDMA configuration" 11
- 已添加 content for "Memory allocation" 12
- 已添加 content for "mmWave configuration: minimal mode with isolation" 12
- 已添加 content for "Maximum velocity disambiguation in SRR subframe" 14
- Updated 节 3.1.2 title to *Software and GUI* and added content addressing the GUI 15
- 已添加 图 7 16
- 已添加 图 8 16
- Updated 图 9 18
- Updated 图 10 18

修订版本 A 历史记录

Changes from Original (April 2017) to A Revision	Page
• 已更改 5 节	19

有关 TI 设计信息和资源的重要通知

德州仪器 (TI) 公司提供的技术、应用或其他设计建议、服务或信息，包括但不限于与评估模块有关的参考设计和材料（总称“TI 资源”），旨在帮助设计人员开发整合了 TI 产品的 应用； 如果您（个人，或如果是代表贵公司，则为贵公司）以任何方式下载、访问或使用了任何特定的 TI 资源，即表示贵方同意仅为该等目标，按照本通知的条款进行使用。

TI 所提供的 TI 资源，并未扩大或以其他方式修改 TI 对 TI 产品的公开适用的质保及质保免责声明；也未导致 TI 承担任何额外的义务或责任。TI 有权对其 TI 资源进行纠正、增强、改进和其他修改。

您理解并同意，在设计应用时应自行实施独立的分析、评价和 判断， 且应全权负责并确保 应用的安全性， 以及您的 应用 （包括应用中使用的所有 TI 产品）） 应符合所有适用的法律法规及其他相关要求。你就您的 应用声明，您具备制订和实施下列保障措施所需的一切必要专业知识，能够 (1) 预见故障的危险后果，(2) 监视故障及其后果，以及 (3) 降低可能导致危险的故障几率并采取适当措施。您同意，在使用或分发包含 TI 产品的任何 应用前， 您将彻底测试该等 应用 和该等应用所用 TI 产品的 功能而设计。除特定 TI 资源的公开文档中明确列出的测试外，TI 未进行任何其他测试。

您只有在为开发包含该等 TI 资源所列 TI 产品的 应用时， 才被授权使用、复制和修改任何相关单项 TI 资源。但并未依据禁止反言原则或其他法理授予您任何TI知识产权的任何其他明示或默示的许可，也未授予您 TI 或第三方的任何技术或知识产权的许可，该等产权包括但不限于任何专利权、版权、屏蔽作品权或与使用TI产品或服务的任何整合、机器制作、流程相关的其他知识产权。涉及或参考了第三方产品或服务的信息不构成使用此类产品或服务的许可或与其相关的保证或认可。使用 TI 资源可能需要您向第三方获得对该等第三方专利或其他知识产权的许可。

TI 资源系“按原样”提供。TI 兹免除对 TI 资源及其使用作出所有其他明确或默认的保证或陈述，包括但不限于对准确性或完整性、产权保证、无屡发故障保证，以及适销性、适合特定用途和不侵犯任何第三方知识产权的任何默认保证。

TI 不负责任何申索，包括但不限于因组合产品所致或与之有关的申索，也不为您辩护或赔偿，即使该等产品组合已列于 TI 资源或其他地方。对因 TI 资源或其使用引起或与之有关的任何实际的、直接的、特殊的、附带的、间接的、惩罚性的、偶发的、从属或惩戒性损害赔偿，不管 TI 是否获悉可能会产生上述损害赔偿，TI 概不负责。

您同意向 TI 及其代表全额赔偿因您不遵守本通知条款和条件而引起的任何损害、费用、损失和/或责任。

本通知适用于 TI 资源。另有其他条款适用于某些类型的材料、TI 产品和服务的使用和采购。这些条款包括但不限于适用于 TI 的半导体产品 (http://www.ti.com/sc/docs/stdterms.htm)、评估模块和样品 (http://www.ti.com/sc/docs/sampterms.htm) 的标准条款。

邮寄地址：上海市浦东新区世纪大道 1568 号中建大厦 32 楼，邮政编码：200122