MATLAB Radar System API

24 GHz Distance2GoL

About this document

Scope and purpose

A guide for installing the MATLAB Interface for the 24 GHz Distance2GoL.

Intended audience

This document is for customers, wishing to get an overview about the 24 GHz Distance2GoL MATLAB Interface.

Table of Contents

About this document **1**
Table of Contents **1**
1 Introduction **2**
1.1 MATLAB Application Interface Overview 2
1.2 MATLAB API Guide and Raw Data Fetching 3
1.2.1 MATLAB Code to Extract Data from 24 GHz Distance2GoL SC-FMCW Board 3
1.2.2 Radar system API Object Description 4
Revision history **7**

1 Introduction

This section contains a small overview of the available code and a small guide on the MATLAB Interface used.

1.1 MATLAB Application Interface Overview

The MATLAB code and interface allows the user to demonstrate the capabilities of the 24 GHz Distance2GoL modules. All of the interfaces' classes and functions are contained in the RadarSystemImplementation folder.

The main class representing a hardware evaluation board as a radar system are contained in the folder:

- @RadarSystem

A modular radar system used for Distance2GoL got only two endpoints to use the radar functionality. The provided endpoints are contained in these folders:

- oEPRadarBaseBoard
- oEPRadarD2GL

Additional helper functions needed by the API are located in these files:

- clearSP.m
- findRSPort.m
- getAvailablePorts.m
- protocol_read_payload_float.m
- protocol_read_payload_int32.m
- protocol_read_payload_uint8.m
- protocol_read_payload_uint16.m
- protocol_read_payload_uint32.m
- protocol_read_payload_uint64.m
- protocol_write_payload_uint8.m
- protocol_write_payload_uint16.m
- protocol_write_payload_uint32.m
- protocol_write_payload_uint32.m
- resetRS.m

1.2 MATLAB API Guide and Raw Data Fetching

This section describes and enumerates the set of control parameters and methods that can be used for controlling the 24 GHz Distance2GoL SC-FMCW demo board using MATLAB. It involves a quick introduction for reviewing and setting up the radar properties and how to fetch raw data from the board in MATLAB.

Coding examples and demos can be found in the folder RadarSystemExamples\GettingStarted.

1.2.1 MATLAB Code to Extract Data from 24 GHz Distance2GoL SC-FMCW Board

To get out raw data form the evaluation board, the following steps are performed:

1. Create the radar system object.
2. Set the endpoint radar properties to customize the frequency chirps.
3. Fetch raw data.
4. Plot the data.

Code Listing 1 Matlab code example

```
% 1. Create radar system object
  szPort = findRSPort; % find the right COM Port
  oRS = RadarSystem(szPort); % creates the Radarsystem API object

% 2. Set endpoint properties
  % get the current parameters from the device
  oRS.oEPRadarD2GL.get_parameters;

  % then change the parameters at the TJPU endpoint:
  oRS.oEPRadarD2GL.parameters.frame_time_sec = 0.1500;
  oRS.oEPRadarD2GL.parameters.number_of_samples = 256;
  oRS.oEPRadarD2GL.parameters.number_of_chirps_per_frame = 10;
  oRS.oEPRadarD2GL.parameters.max_distance_m = 20;
  oRS.oEPRadarD2GL.apply_parameters;

% 3. Trigger radar chirp, get the raw data and plot it
  mxRawData = oRS.oEPRadarD2GL.get_raw_data;

% 4. plot data
plot([real(mxRawData.sample_data(:,1)), imag(mxRawData.sample_data(:,1))]);
```

1.2.2 Radar system API Object Description

The radar system API object (oRS) contains the endpoint objects oEPRadarBaseBoard and oEPRadarD2GL.

The object oEP is alternatingly used to access all the properties of the different endpoint objects.

a) oEPRadarBaseBoard
This endpoint is used to provide the common data from the the XMC4700 Radar Base board:

Public methods

Method name	Description
get_board_info	Reads the board info from the connected device
get_consumption_info	Reads the power and current consumption data from the device
get_consumption_info_definition	Reads the information what kind of consumption information is available on the device
reset_parameters	Resets to parameters to the factory settings

b) oEPRadarD2GL
This endpoint is used to set the parameters of the Distance2GoL SC-FMCW shield and has the following properties and methods:

Writeable properties

Property name	Description
apply_parameters	Applies the previously changed parameters to the board
get_frame_data	Reads the radar raw data from the device
get_raw_data	Alias to “get_frame_data”
get_parameter	Reads the parameter from the device
get_shield_info	Reads general information about the connected RF shield
get_result	Reads the calculation results of the Doppler algorithm running on the device
get_result_and_raw_data	Reads the raw data AND the results from the device in a combined call; use this method, if results and raw data is read at the same time.
get_parameters_def	Reads the limits for the parameters that can be applied.

Available parameters

Property name	Description	Valid values
max_distance_m	The maximum distance in m; acts as a filter for the algorithm, to provide distance information	[2 … 30]
min_distance_m	The minimum distance in m; acts as a filter for the algorithm, to provide distance information	[1 … 25]
frame_time_sec	The frame time in seconds	[0.02 … 0.200]
number_of_samples	The number of samples per chirp	[32, 64, 128, 256]
number_of_chirps_per_frame	The number of chirps per frame	[1 … 16]

Property name	Description	Valid values
pulse_repetition_time_sec	The pulse/chirp repetition time in seconds	[0.005 … 0.010]
target_activation_threshold	The threshold used to activate a target	[1 … 1000]
target_hold_threshold	The threshold used to hold a target	[1 … 1000]
baseband_gain_stage	Select of (0) high gain or (1) low gaindata output shall be used for signal processing and target tracking	[0, 1]
activate_cfar	A switch to use (0) default or (1) cfar-based target detection	[0, 1]

Parameter limit	Description
max_distance_m_lower_boundary	The lower boundary of the maximum distance in m
max_distance_m_upper_boundary	The upper boundara of the maximum distance in m
min_distance_m_lower_boundary	The lower boundary of the minimum distance in m
min_distance_m_upper_boundary	The upper boundary of the minimum distance in m
frame_time_sec_lower_boundary	The lower boundary of the frame time in sec
frame_time_sec_upper_boundary	The upper boundary of the frame time in sec
sample_list	The valid values for the number of samples per frame
number_of_chirps_per_frame_lower_boundary	The lower boundary of the number of chirps per frame
number_of_chirps_per_frame_upper_boundary	The upper boundary of the number of chirps per frame
target_activation_threshold_lower_boundary	The lower boundary of the target activation threshold
target_activation_threshold_upper_boundary	The upper boundary of the target activation threshold
target_hold_threshold_lower_boundary	The lower boundary of the target hold threshold
target_hold_threshold_upper_boundary	The upper boundary of the target hold threshold
pulse_repetition_time_sec_lower_boundary	The lower boundary of the pulse repetition time in sec
pulse_repetition_time_sec_upper_boundary	The upper boundary of the pulse repetition time in sec

Structure info of the **raw data** where received frame information data is stored:

Variable name	Description
frame_number	Index of the corresponding frame
num_chirps	Number of chirps in a frame
num_rx_antennas	Number of Rx antennas active
num_samples_per_chirp	Number of samples in a chirp
rx_mask	Active receiving antennas
data_format	Determines if the returned data contains only I or Q signal (‘Real Data’), I and Q signals in separate data blocks (‘Complex Data’) or I and Q signals in one interleaved data block (Complex Interleaved Data).
adc_resolution	ADC Resolution

Variable name	Description
interleaved_rx	Bool flag to indicate if the received data from different antennas stored in an interleaved pattern in the raw data stream.
sample_data	Contains the actual raw data from the radar sensor

Structure info of the **result data** where received frame information data is stored:

Variable name	Description
result_cnt	A count for the results that is incremented, everytime a result is delivered
frame_count	The frame number, could be used to sync raw data and results for analysis
range_m	The range in m
strength	The target signal strength (corresponding peak in target activation data if above threshold, else corresponding peak in target hold data)
velocity_mps	The velocity in m/s with sign
target_status	0: no target detected, 1: target is static, 2: transit state, 3: target is moving
direction_of_movement	0: no target detected, 1: target is static, 2: target is approaching, 3: target is departing
activation_data	The array of activation data
hold_data	The array of hold data

Revision history

Document version	Date of release	Description of changes
V1.0	2021-03-31	Initial version

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2021-03-31

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2021 Infineon Technologies AG. All Rights Reserved.

Do you have a question about this document?

Email: erratum@infineon.com

Document reference
MATLAB Radar System API

IMPORTANT NOTICE

The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.