AUTO-DRIVING / TECHNICAL REPORTS / SENSORS

Share

TR_Tin_UserGuideRadar_v1_20211024

T

Created by tin.giang.s2.tt

Oct 24, 2021 • 3 min read • 3 people viewed

mmWave Demo ... 9c).pdf
24 Oct 2021, 04:07 PM

MS_60GHz_AoP... 1.0.pdf
24 Oct 2021, 04:05 PM

TI Demonstration Code Texas Instruments offer several demonstration codes for various purposes. **The most basic demo is the Out-of-Box Demo** that has a **web-based GUI**. In the browser, it has an interface for configuration and shows live plots: a scatter plot of identified objects, a range plot with reflection power for different distances from the sensor, and a doppler-range plot for objects' velocity. All plots only show the data live and does not save any data for further processing. Another available demo is the Area Scanner for Zone Occupancy Detection [17]. This demo has a graphical MatLab interface that both communicates with the radar through UART and processes and plots the received data. The purpose of this demo is to detect people and objects as they enter and exits a zone, and therefore the chip does not send information about permanent still objects.

For this study, the data from the Out of Box Demo is relevant, but it needs to be captured. Therefore the binary file from the Out of Box Demo is used together with a modified MatLab script from the Area Scanner demo. The modifications are to match the output format, store the data, and plot the relevant data.

mmWave Demo Visualizer (ti.com)

1. USER GUIDE

- Download Document for mmWave Demo

Software Documentation - Mistral Solutions | Software Downloads

- Hardware, how to run Demo

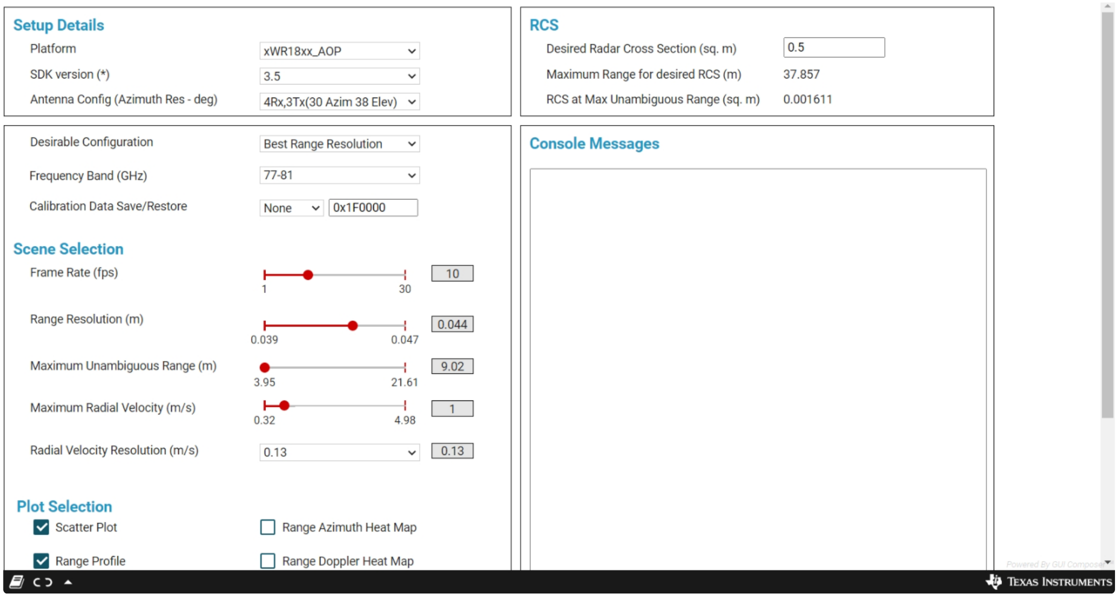
60GHz AoPCB mmWave Industrial Radar on Module User Guide (PDF)

2. CONFIGURE

- configuring mmWave sensors and visualizing point cloud objects generated by the mmWave SDK demo

mmWave Demo Visualizer (PDF)

- CONFIGURE Tab



Setup Details

Platform	xWR18xx_AOP
SDK version (*)	3.5
Antenna Config (Azimuth Res - deg)	4Rx,3Tx(30 Azim 38 Elev)
Desirable Configuration	Best Range Resolution
Frequency Band (GHz)	77-81
Calibration Data Save/Restore	None 0x1F0000

RCS

Desired Radar Cross Section (sq. m)	0.5
Maximum Range for desired RCS (m)	37.857
RCS at Max Unambiguous Range (sq. m)	0.001611

Console Messages

Scene Selection

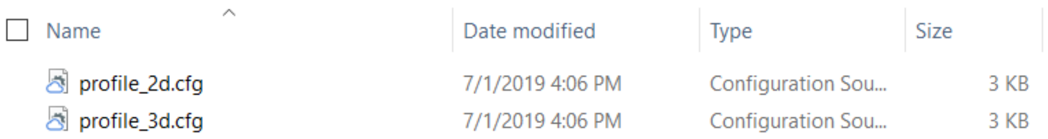
Frame Rate (fps)	1	30	10
Range Resolution (m)	0.039	0.047	0.044
Maximum Unambiguous Range (m)	3.95	21.61	9.02
Maximum Radial Velocity (m/s)	0.32	4.98	1
Radial Velocity Resolution (m/s)	0.13		0.13

Plot Selection

☑ Scatter Plot ☐ Range Azimuth Heat Map

☑ Range Profile ☐ Range Doppler Heat Map

- Config file demo



Name	Date modified	Type	Size
profile_2d.cfg	7/1/2019 4:06 PM	Configuration Sou...	3 KB
profile_3d.cfg	7/1/2019 4:06 PM	Configuration Sou...	3 KB

3. FLASH PROGRAMMING

- The SOP Switch to mode Flash Programming Mode

The SOP Switch (SW2) setting determines the mode of operation of AoPCB. The following table details its function.

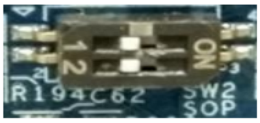
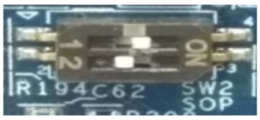
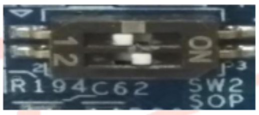
SOP Setting	Mode of Operation	Image
SW2-1 position → OFF SW2-2 position → OFF	Functional Mode	
SW2-1 position → ON SW2-2 position → OFF	Flash Programming Mode	
SW2-1 position → OFF SW2-2 position → ON	Development /Raw Data Capture mode	

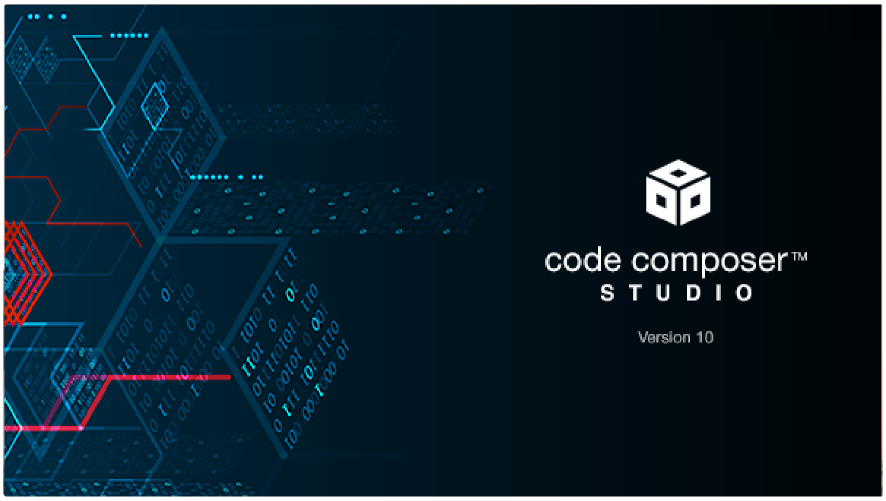
Table 1 : SOP Switch setting

- Uniflash cloud version UniFlash (ti.com)
- Uniflash offline version UNIFLASH Software programming tool | TI.com

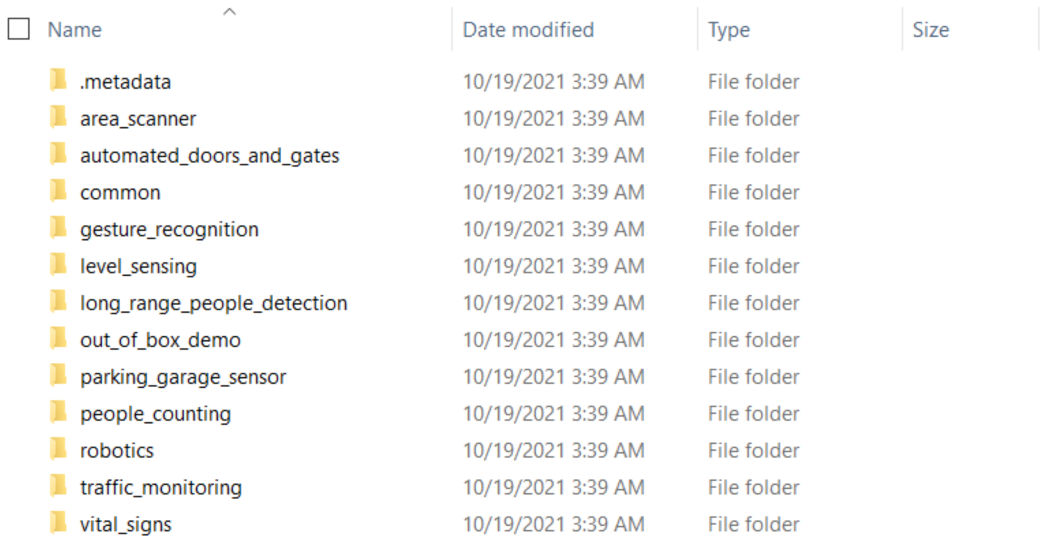
4. Developer's Guide

68xx AoP - mmWave SDK Demo (ti.com)

- Start CCS and setup workspace as desired



- Programing mmwave_industrial_toolbox_4_9_0

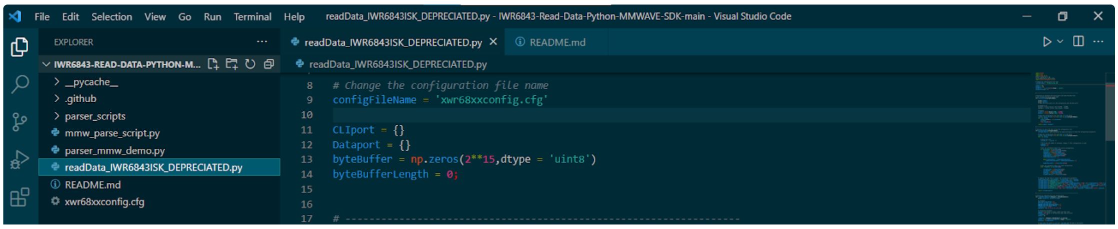


Name	Date modified	Type	Size
.metadata	10/19/2021 3:39 AM	File folder	
area_scanner	10/19/2021 3:39 AM	File folder	
automated_doors_and_gates	10/19/2021 3:39 AM	File folder	
common	10/19/2021 3:39 AM	File folder	
gesture_recognition	10/19/2021 3:39 AM	File folder	
level_sensing	10/19/2021 3:39 AM	File folder	
long_range_people_detection	10/19/2021 3:39 AM	File folder	
out_of_box_demo	10/19/2021 3:39 AM	File folder	
parking_garage_sensor	10/19/2021 3:39 AM	File folder	
people_counting	10/19/2021 3:39 AM	File folder	
robotics	10/19/2021 3:39 AM	File folder	
traffic_monitoring	10/19/2021 3:39 AM	File folder	
vital_signs	10/19/2021 3:39 AM	File folder	

5. IWR6843-Read-Data-Python-MMWAVE-SDK

kirkster96/IWR6843-Read-Data-Python-MMWAVE-SDK: Read IWR6843ISK sensor serial data using Python (github.com)

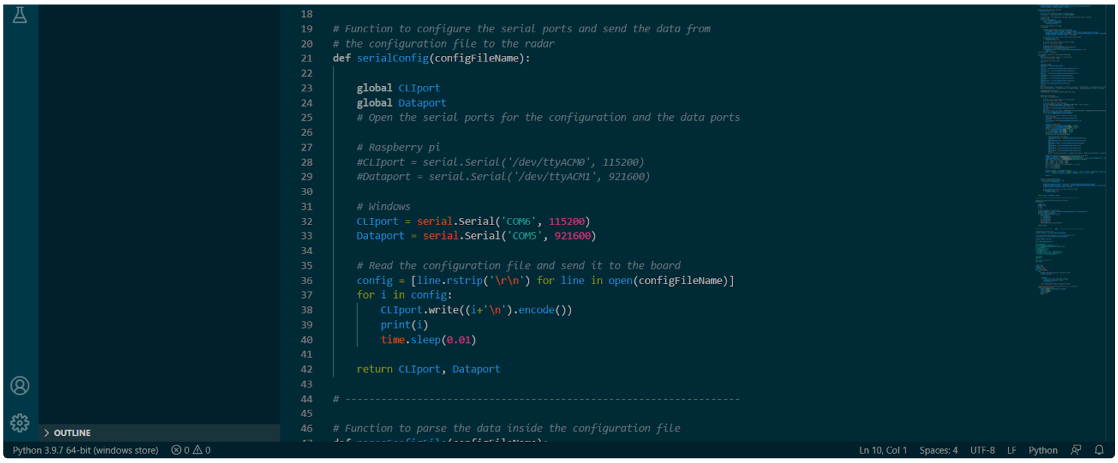
Read and plot IWR6843ISK sensor serial data using a Python program.


File
Edit
Selection
View
Go
Run
Terminal
Help
readData_IWR6843ISK_DEPRECIATED.py - IWR6843-Read-Data-Python-MMWAVE-SDK-main - Visual Studio Code
EXPLORER
IWR6843-READ-DATA-PYTHON-M...
pycache
.github
parser_scripts
mmw_parse_script.py
parser_mmw_demo.py
readData_IWR6843ISK_DEPRECIATED.py
README.md
xwr68xxconfig.cfg
readData_IWR6843ISK_DEPRECIATED.py
README.md
readData_IWR6843ISK_DEPRECIATED.py

```python
# Change the configuration file name
configFileName = 'xwr68xxconfig.cfg'

CLIport = {}
Dataport = {}
byteBuffer = np.zeros(2**15,dtype = 'uint8')
byteBufferLength = 0;


# ------------------------------------------------------------------
```



```python

# Function to configure the serial ports and send the data from
# the configuration file to the radar
def serialConfig(configFileName):

    global CLIport
    global Dataport
    # Open the serial ports for the configuration and the data ports

    # Raspberry pi
    #CLIport = serial.Serial('/dev/ttyACM0', 115200)
    #Dataport = serial.Serial('/dev/ttyACM1', 921600)

    # Windows
    CLIport = serial.Serial('COM6', 115200)
    Dataport = serial.Serial('COM5', 921600)

    # Read the configuration file and send it to the board
    config = [line.rstrip('\r\n') for line in open(configFileName)]
    for i in config:
        CLIport.write((i+'\n').encode())
        print(i)
        time.sleep(0.01)

    return CLIport, Dataport

# ----------------------------------------------------------------

# Function to parse the data inside the configuration file
```

- OUTPUT

When running the **Out of Box Demo** binary on the chip and after configuring it over UART, the point cloud data is sent frame by frame to the computer.

Each frame contains header information, a range and noise profile, and data about

each identified point. Each detected point in the frame has the information: x-value, y-value, z-value, doppler (velocity), SNR, and noise

Like Be the first to like this No labels