uRAD

User Manual

Level Sensing

Contents

About uRAD Level Sensing 1

Congratulations on purchasing uRAD Level Sensing

> ⚠ *WARNING:* To avoid any injury or damage, read all operating instructions in this guide and, specially, the safety and warranty information in "Chapter 4: Safety and Handling" and "Chapter 5: Product Warranty", before using uRAD.

Through this manual you will learn how to use uRAD Level Sensing solutions for applications related with measuring frontal distance with high accuracy and ease of operation.

Additional Information

Hardware released:	URADLSIWR20	1/11/2022
	URADLSAWR20	1/11/2022
Purchase:	www.uRAD.es/en/	
Technical specifications:	www.uRAD.es/en	
Software download:	www.uRAD.es/en/mi-cuenta/downloads (only with purchase)	
Contact:	contact@uRAD.es	

Last version: 1/11/2022

uRAD Level Sensing Basics

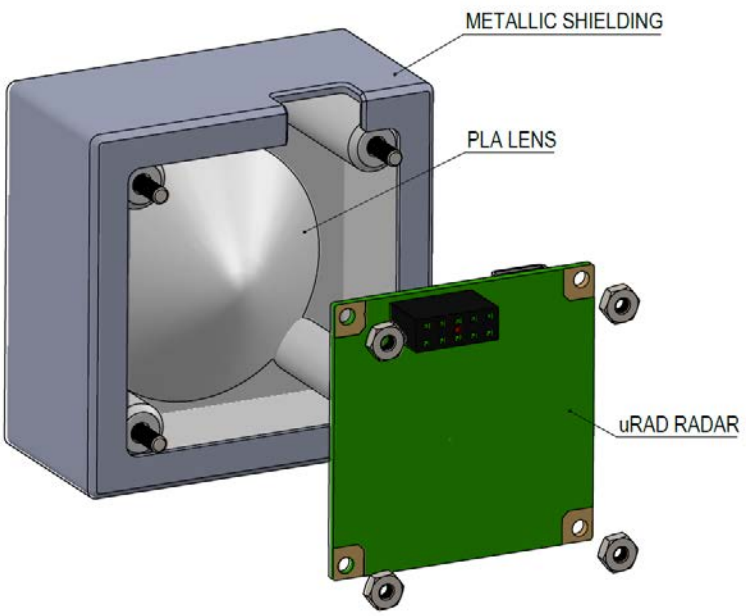
2

Read this chapter to learn about the features of uRAD Level Sensing, how to use it, and more.

uRAD Level Sensing solution is a millimeter wave radar sensor specifically designed to measure frontal distance with outstanding accuracy.

Hardware Description

uRAD Level Sensing is comprised of two main parts, a radar board and a plastic case with integrated lens.



1. Radar board

uRAD Level Sensing core is one of the standard radars of uRAD. Two options are available:

- uRAD Industrial (IWR): working at 60-64 GHz frequency band and based on the radar chipset IWR6843AoP by Texas Instruments.
- uRAD Automotive (AWR): working at 77-81 GHz frequency band and based on the radar chips AWR1843AoP by Texas instruments.

These integrated circuits are one of the most advanced radar chipsets in the market because of its high accuracy, small size and sensing simplified. Frequency selection should depend on specific application and regulation of each region.

At https://urad.es/en/descargas/ datasheet and user manual of uRAD Industrial and Automotive can be downloaded for widening the information of these standard products.

2. Case

The housing is a plastic case printed in PLA that contains an integrated lens in order to focus the beam and narrowing the field of view. The lens reduces the field of view from 160 degrees to 6 degrees. In this way, all the emitted power is focused in the frontal direction.

Moreover, the case is surrounded by aluminum metallic tape to avoid any undesired radiation at wide angles.

The radar board and the housing are attached with 4 screws and nuts of metric M2.5 mm that you can also use to attach the sensor to your particular holder.

> *WARNING:* The device emits through the front side. Do NOT cover it with any metallic or electronic element, nor electromagnetic absorber.

3. USB connector

The micro-USB connector is used for flashing firmware, configuring the radar and getting the results.

The USB is connected with the radar by two different UART channels. uRAD uses the following chip for using the two UART ports with one USB: **Silicon Labs Dual CP2105 USB to UART bridge**. When you connect the radar to a computer, two new USB devices should be detected in the Device Manager:

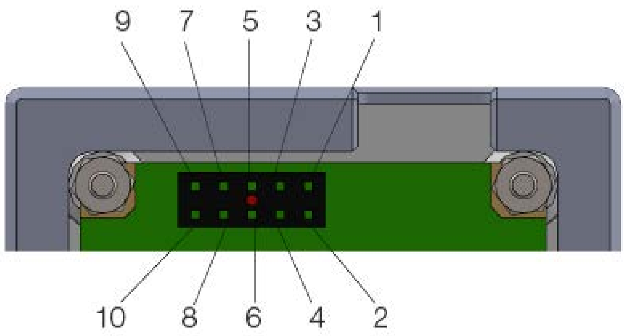
- **Enhanced COM** is the port for sending the configuration from the computer to the radar.
- **Standard COM** is the port for getting the data from the radar.

> *ADVICE:* Install the Virtual COM Port (VCP) drivers of the CP2105 USB to UART bridge from the Silicon Labs website in the case your computer does not recognize the USB port (https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers).

The radar is powered directly by the micro-USB with 5V. The power consumption depends on the operating mode but it is around 140 mA.

4. 10-pin connector

The 10-pin female header connector has two UART channels, one on/off pin, one reset pin and two GPIOs. Spacing between the pins is 2.54 mm. The pins distribution is as follows:



1 **GND**: ground plane of the PCB.
2 **5V**: power the device with this pin. Voltage range is from +4.5 V to +5.5V.
3 **UART TX config**: TX line of the UART port used for configuration.
4 **UART RX config**: RX line of the UART port used for configuration.
5 **UART TX data**: TX line of the UART port used for data output.
6 **UART RX data**: RX line of the UART port used for data output.
7 **RESET**: with the rising edge of a digital pulse in this pin, the radar performs a reset. This pin has a pullup resistor.
8 **ON/OFF**: a HIGH/LOW digital level in this pin, TURN ON/OFF the radar. In OFF state, the radar consumption is reduced to the minimum (< 1mA). This pin has a pullup resistor.
9 **GPIO1**: not used by default. Useful for firmware customization.
10 **GPIO0**: not used by default. Useful for firmware customization.

Firmware Description

As you already know, there are two version of uRAD Level Sensing and, therefore, the firmware for each model is different. Both firmware, delivered with the purchase, can be found in the *prebuilt_binary* folder.

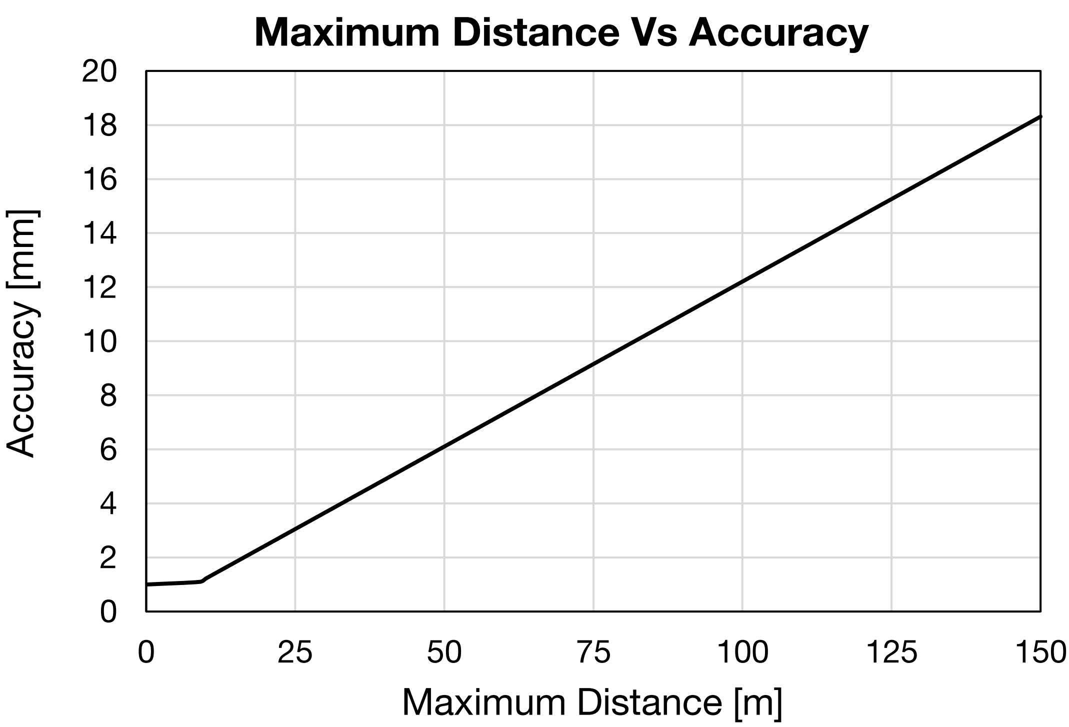
- uRAD LS IWR: firmware file is ***uRAD_LevelSensing_IWR6843AoP.bin***
- uRAD LS AWR: firmware file is ***uRAD_LevelSensing_AWR1843AoP.bin***

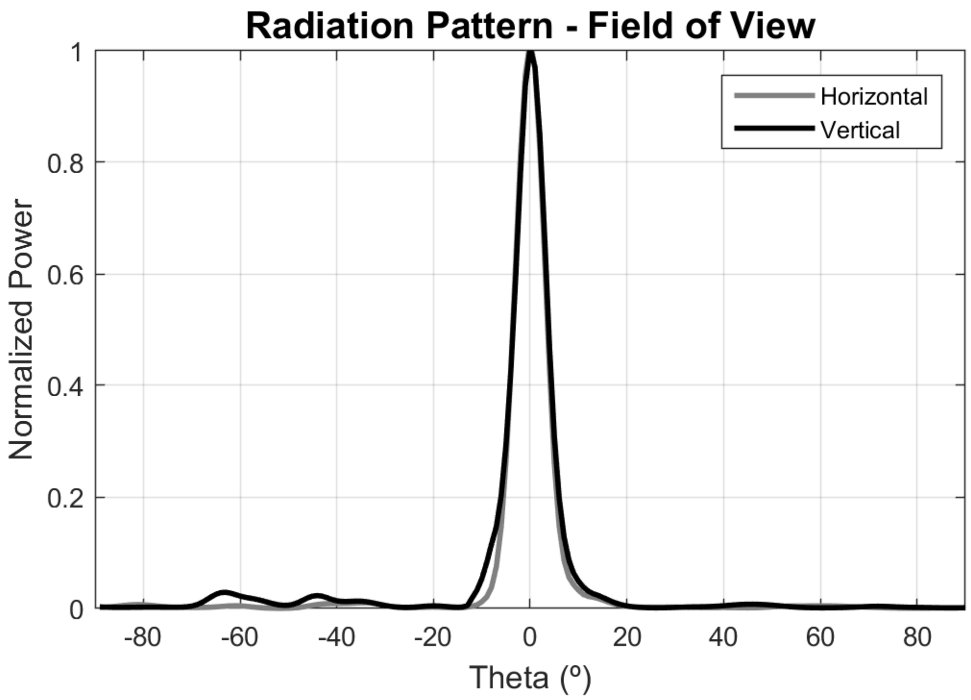
> *ADVICE:* your radar board is already flashed with its corresponding firmware. Take a look at chapter 3 of the general user manual to learn how to upload new firmware to your board.

Technical Features

The main technical features of uRAD Level Sensing are:

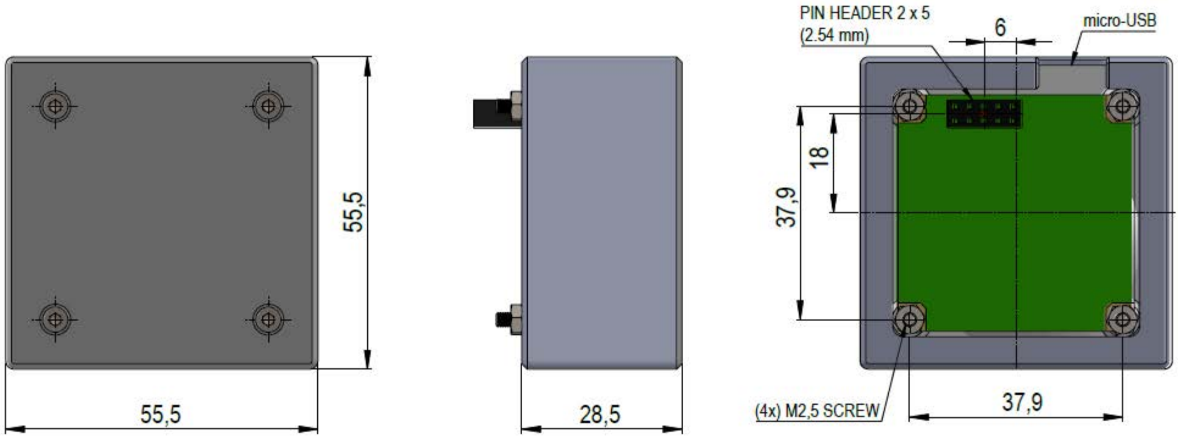
Parameter	*Typical value*
Operating conditions	
Supply voltage	4.5 – 5.5 V
Supply current	140 mA
Digital signals	3.3 V
Operating temperature	-20 to +85ºC
Performance	
Frequency bandwidth	
• IWR model	60 – 64 GHz
• AWR model	77 – 81 GHz
Range	0 to 150 m
Accuracy	1 to 18 mm
Field of view	6 x 6 deg





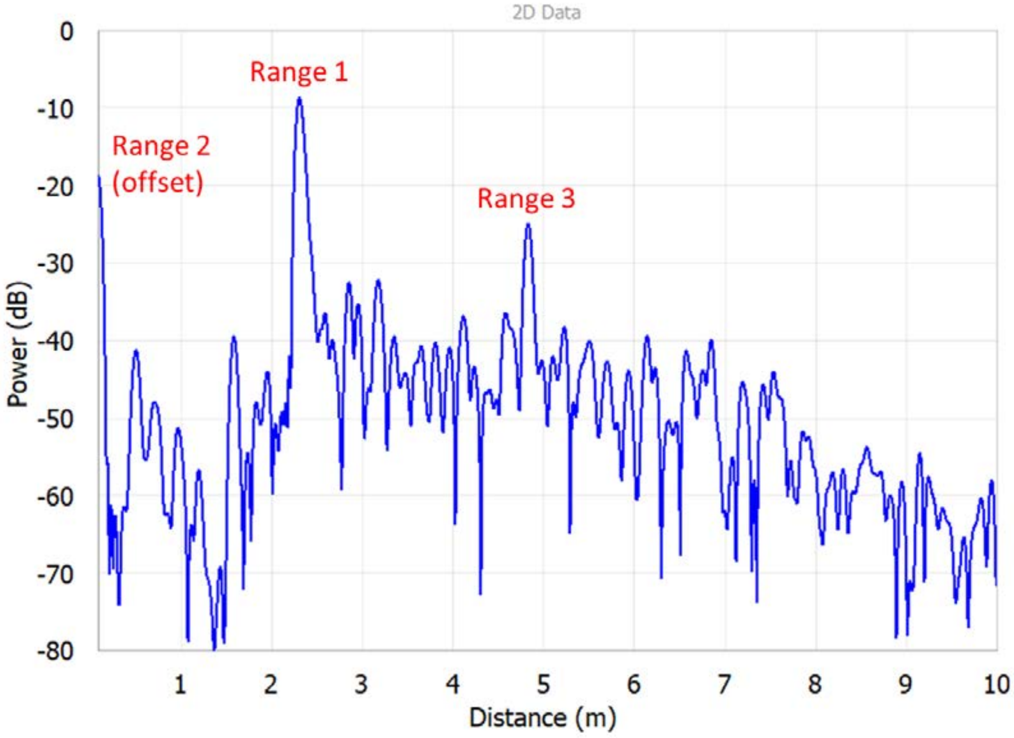
Mechanical Features

Parameter	*Value*
Dimensions	55.5 x 55.5 x 28.5 mm
Weight	41.4 g



Theoretical Aspects

The radar calculates the distances to the targets using the Fast Fourier transform (FFT) of the received signal. By looking at the spectrum of the FFT, you can identify higher level peaks, such as reflections from objects or surfaces. The level sensing software allows you to visualize this spectrum and returns the range of the three peaks with the highest level. Here is an example:



In the example, the radar is fixed on a table aiming at the ceiling that is at a distance of 2.5 meters. Three peaks are identified in the graph.

- The most significant one, range 1, at 2.5 meters and that has the greatest amplitude is the ceiling, which provides the greatest reflection.
- There is a second peak, range 2, practically at 0, which does not correspond to any real value. At a very short distance, generally less than 10 cm, there is always a peak that is not real, called offset, which is inherent in radar technology and must be ruled out.
- A third peak, range 3, at 5 meters, corresponds to the double reflection produced by the table and the ceiling, which the radar is also capable of detecting.

The sensing software returns the distance of the three peaks with the greatest amplitude, but **the configuration allows you to select the range of distances to search for the three peaks.** Therefore, it is possible to easily eliminate the offset from the results or discard all those peaks outside the range of interest.

Software

3

In this chapter, basic programming of uRAD Level Sensing is explained

Along with the hardware and firmware, various Python and C++ scripts are also provided to control the sensor, either by its USB connector or by the pin connector via UART.

WARNING: Do NOT forget to install or update all the necessary Python libraries listed at the beginning of the scripts.

General Functioning

The sensor working is quite simple. The master device, that controls uRAD, sends the configuration commands to the radar by USB or UART. Once uRAD received the commands, starts sending the measurements values at the configured sampling rate.

Most of the commands are set by default. Only a few have to be configured. In the examples, we have done this process transparent for the user. Therefore, it has to be configured:

```
#### CONFIGURATION PARAMETERS ####
model = 'IWR'            # 'IWR' or 'AWR'
maximum_distance = 9   # maximum distance to measure. From 9 to 150 m
range_min = 0            # lower limit of the range of interest
range_max = 10           # upper limit of the range of interest
sampling_rate = 2        # min = 2, max = 20 samples per second
offset = 0               # apply an offset to all range measured values
```

- **model:** 'IWR' if your radar model is uRAD Industrial and works at 60 GHz, or 'AWR' if it is uRAD Automotive working at 77 GHz.
- **maximum_distance**: radar can be configured to measure a maximum distance from 9 meters up to 150 meters. **It is better to choose the lowest possible value to have the best resolution.**
- **range_min** and **range_max**: distance range of interest. Limit the results to this range of interest.
- **sampling_rate**: measurements per second the radar sends to the master device. From 2 up to 20 samples per second.
- **offset**: add an offset (in meters) to the measurement values.

level_sensing_UART.py

This script is useful when sensor is controlled by the pin connector using a single UART channel. Only the distance results are sent (**FFT plot and amplitude values are not received**).

Some extra configuration parameters have to be set:

```
#### OUTPUT RESULTS ####
saveResults = True            # Save results in .txt file
printResults = True           # Print results in the console
numberOfPeaksToSave = 3       # Number of peaks to save (max = 3)
fileSizeMinutes = 2           # Minutes to split the .txt file

#### UART PORT ####
port_name = '/dev/serial0'

#### RESET ####
reset_pin_number = 6    # GPIO pin number of the Master Device
reset = True
```

- **saveResults**: results are saved inside *output_files* folder in ***YYYY_MM_DD_HH_mm_results.txt*** file. Each results file has a header name with date and time.
- **printResults**: print results on the Python console.
- **numberOfPeaksToSave**: choose the number of peaks (most significant distance values) to save. Maximum 3.
- **fileSizeMinutes**: split the results file by minutes, to limit the size of each file.

Also, the UART port name and whether you want to use a reset pin (and its pin number) has to be configured.

level_sensing_USB.py

Similar to the previous one but controlled by USB. It does not include the reset pin. **Therefore, each time you want to run again the code, you have to manually reset the radar with the physical reset button.**

```
#### OUTPUT RESULTS ####
saveResults = True            # Save results in .txt file
printResults = True           # Print results in the console
numberOfPeaksToSave = 3       # Number of peaks to save (max = 3)
fileSizeMinutes = 2           # Minutes to split the .txt file

#### USB PORT ####
configPort_name = 'COM1'
dataPort_name = 'COM1'
```

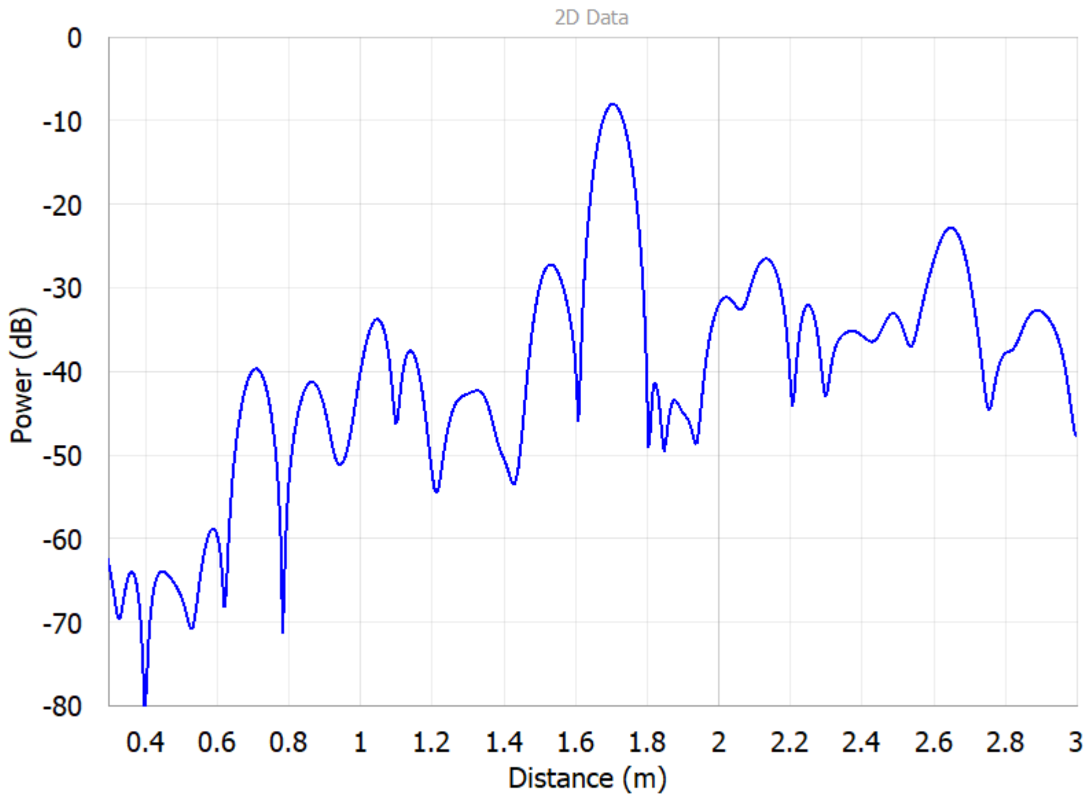
level_sensing_UART_GUI.py

The most complete one because allows you to obtain **distance results and also amplitude of the peaks**. The configuration commands include to send the IQ values for the FFT plot. Therefore, the peaks plot (FFT) can be visualized if selected. This is especially useful for calibration.

```
#### OUTPUT RESULTS ####
calculateAmplitude = True   # Calculate amplitude of the peaks besides range
saveResults = True          # Save results in .txt file
printResults = True         # Print results in the console
plotSpectrum = True         # Show Amplitude Vs Distance plot
numberOfPeaksToSave = 3     # Number of peaks to save (max = 3)
fileSizeMinutes = 2         # Minutes to split the .txt file

#### UART PORT ####
port_name = '/dev/serial0'

#### RESET ####
reset_pin_number = 6    # GPIO pin number of the Master Device
reset = True
```



level_sensing_USB_GUI.py

Similar to the previous one, but via USB. **Remember that every time you want to launch the code again, you must manually reset the radar with the physical reset button**.

```
#### OUTPUT RESULTS ####
calculateAmplitude = True   # Calculate amplitude of the peaks besides range
saveResults = True          # Save results in .txt file
printResults = True         # Print results in the console
plotSpectrum = True         # Show Amplitude Vs Distance plot
numberOfPeaksToSave = 3     # Number of peaks to save (max = 3)
fileSizeMinutes = 2         # Minutes to split the .txt file

#### USB PORT ####
configPort_name = 'COM2'
dataPort_name = 'COM1'
```

level_sensing_function.py

This example is a script that shows how to create a simple function to take a single measurement (averaged over the number of samples you want) each time the function is called. Therefore, it allows the user to take only one measurement at the desired time.

level_sensing_UART.cpp

This C++ script is equivalent to the Python script ***level_sensing_UART.py***. The difference in this script is that, instead of defining the configuration parameters, these have to be directly introduced (the equivalent number) in the configuration commands.

The configuration commands, in the code, are these lines:

```
"flushCfg\n", \
"dfeDataOutputMode 1\n", \
"channelCfg 1 1 0\n", \
"adcCfg 2 1\n", \
"adcbufCfg 0 1 1 1\n", \
"profileCfg 0 60 7 7 114.4 0 0 31.23 1 512 5000 0 0 48\n", \
"chirpCfg 0 0 0 0 0 0 0 1\n", \
"frameCfg 0 0 10 0 500 1 0\n", \
"lowPower 0 0\n", \
"guiMonitor 1 0 0 0 0 1\n", \
"RangeLimitCfg 2 1 0.1 9.0\n", \
"sensorStart\n"
```

The 5 relevant numbers to set are highlighted in yellow that corresponds to:

- start frequency according to ***model***, 60 for IWR or 77 for AWR.
- chirp slope according to ***maximum_distance***, from 1.87 (150 meters) to 31.23 (9 meters). Both values are related with the formula:

$$chirp\_slope\ [MHz/\mu s] = \frac{281.055429375}{maximum\_distance\ [m]}$$

- time between measurements according to ***sampling_rate***, from 50 (20 samples/s) to 500 (2 samples/s). Both values are related with the formula:

$$measurement\_periodicity\ [ms] = \frac{1000}{sampling\_rate}$$

- *range_min* and *range_max*.

Moreover, a similar code called ***uRAD_LevelSensing.ino*** is also provided for Arduino platform.

Example with Real Measurements

Next, a real case is presented.

The setup is as follows. The radar is placed at a distance of 1.7 meters from a metal sheet. Halfway through, a crystal will be placed. Three situations are going to be measured: without glass, with the glass perpendicular to the radar and with the glass tilted.

The radar is connected to a computer. It is powered directly by the USB where it is also configured and received the data.

The script used is ***level_sensing_USB_GUI.py*** since we want to show the peak graph in addition to saving the results.

The configuration is:

```
#### CONFIGURATION PARAMETERS ####
model = 'IWR'            # 'IWR' or 'AWR'
maximum_distance = 9   # maximum distance to measure. From to 150
meters
range_min = 0.3          # lower limit for the range of interest
range_max = 3            # upper limit for the range of interest
sampling_rate = 2        # min = 2, max = 20 samples per second
offset = 0               # apply an offset to all range measured values
```

We connect the radar and see the COM ports that have been assigned. We configure the rest of the script parameters as follows:

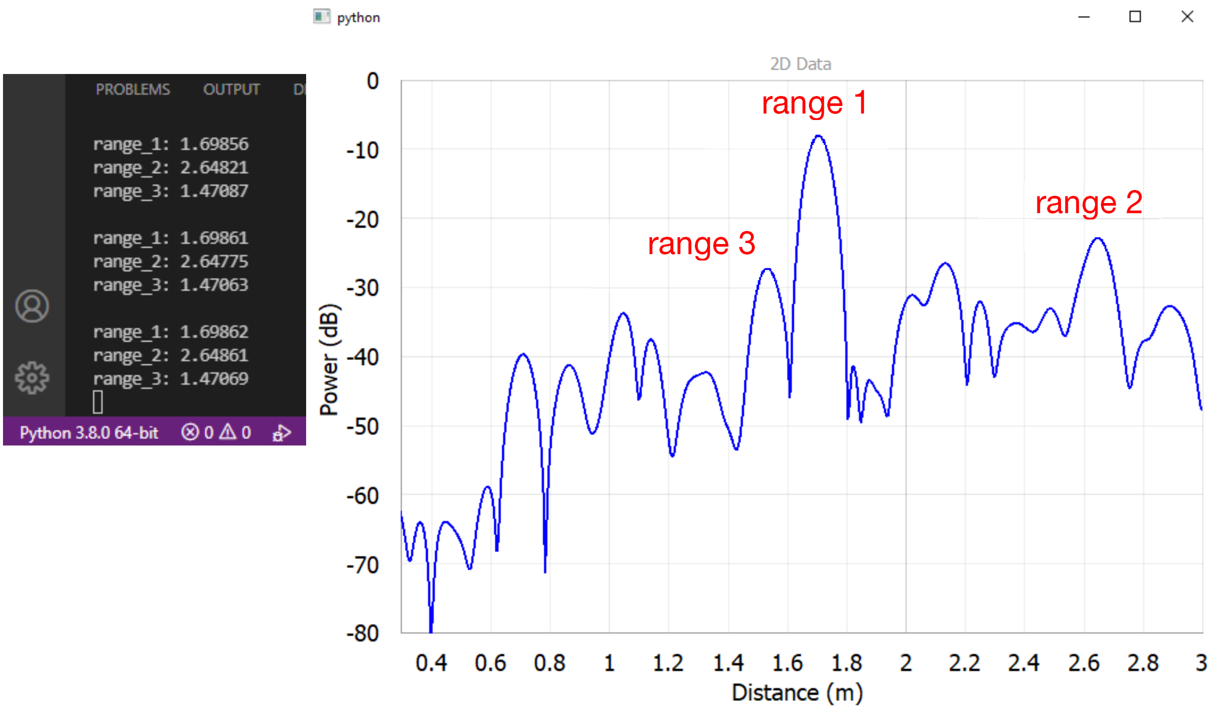
```
#### OUTPUT RESULTS ####
calculateAmplitude = False   # Calculate amplitude of the peaks besides
saveResults = True          # Save results in .txt file
printResults = True         # Print results in the console
plotSpectrum = True         # Show Amplitude Vs Distance plot
```

```
numberOfPeaksToSave = 3      # Number of peaks to save (max = 3)
fileSizeMinutes = 2          # Minutes to split the .txt file

#### USB PORT ####
configPort_name = 'COM1'
dataPort_name = 'COM1'
```

- **Without glass**

After running the script, we obtain the following measured results:

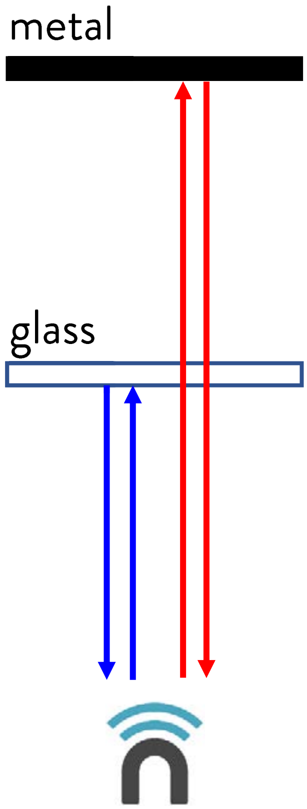


Looking at the results both in the Python console and in the graph, we see the three peaks that are identified. The results are ordered from highest to lowest amplitude of the peak.

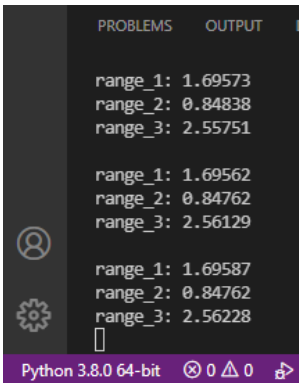
At the distance of 1.69856 meters, the metal has been measured with an amplitude well above the other two peaks, which do not actually correspond to any surface of interest.

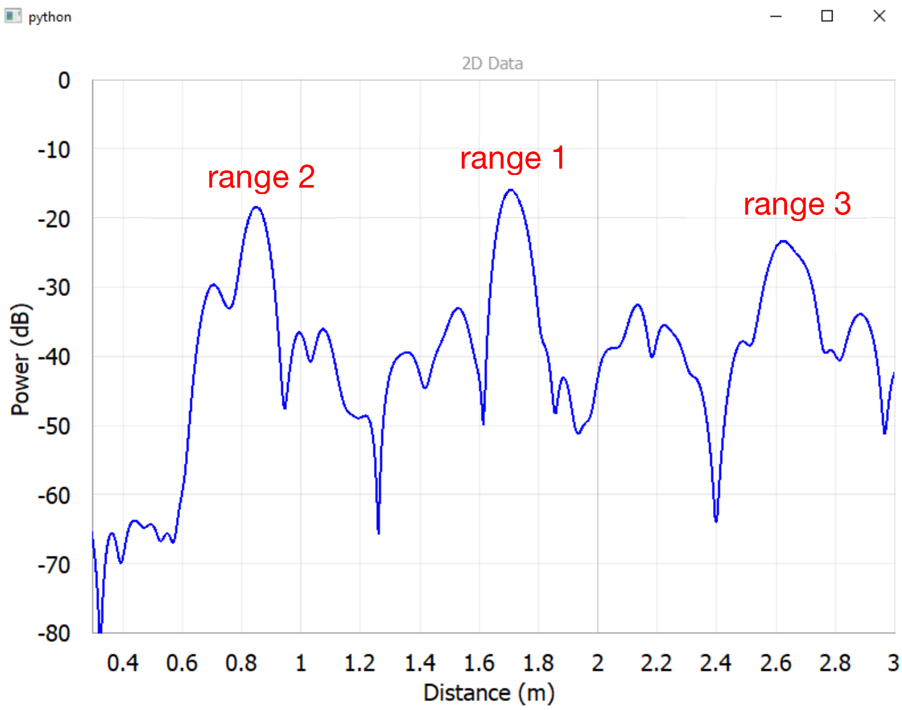
The rest of the peaks of the spectrum, including range 2 and 3, are produced by the FFT processing itself and by rebounding from the environment.

- **Perpendicularly placed glass**



The obtained results are:



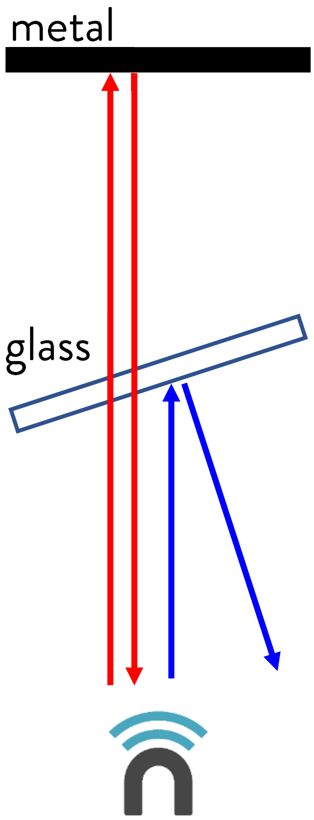


In this case, how the glass appears at a distance of 0.84838 meters and the metal continues to appear at a distance of 1.69573 meters is observed. In this case, the amplitude level of both peaks is very similar, so the value of range 1 could be jumping between both values or even attribute the peak of greater amplitude to the glass. In addition, the peak corresponding to the metal has decreased in amplitude as a consequence of the attenuation of the wave through the glass.

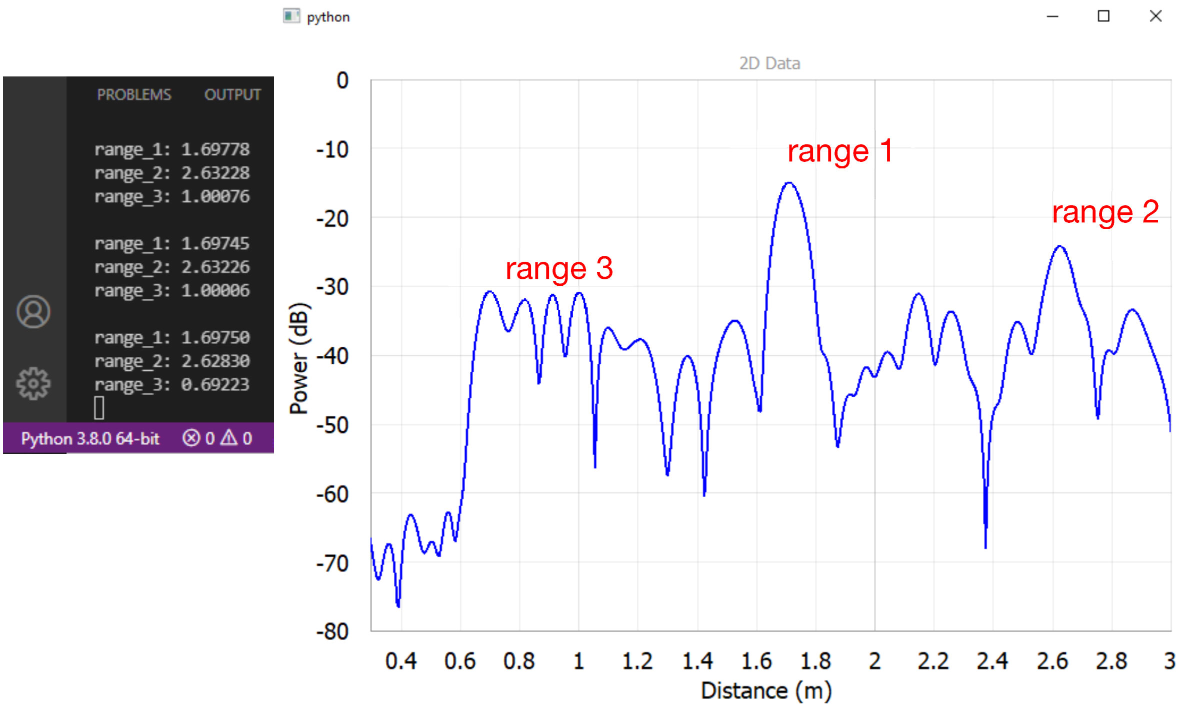
- **Tilted placed glass**

In this case, we tilt the glass to observe its peak disappears. With a tilt of 20-30 degrees, it is enough for the peak to disappear.

This can be very useful, for example, in real situations where it is required to isolate the radar with a cover or where it is unavoidable to pass through various materials. In that case, a tilt on that surface will mitigate the effect of the reflection.



By tilting the glass, most of the reflected power does not come back to the radar, as shown in the image above.



How the peak of the glass has been completely mitigated is observed. In this new situation, range 1 is unequivocally identified with the metal and its amplitude being much higher than the rest of the peaks.

The text file generated with the results, is saved in the *output files* folder. The format is as follows:

```
1.69778 2.63228 1.00076 1622787713.261
1.69745 2.63226 1.00006 1622787713.762
1.69750 2.62830 0.69223 1622787714.261
1.69761 2.63123 0.69210 1622787714.762
1.69776 2.63211 1.00037 1622787715.260
```

The values of each measure are saved in each row, one frame per row, ordering by columns:

```
range1    range2    range3    time_stamp
```

Safety & Handling

4

This chapter includes important safety and handling information for uRAD.

Read all safety and handling information below as well as the operating instructions before using uRAD in order to avoid any injury or damage.

Keep this user guide on hand for future reference.

Important Safety Information

WARNING: Failure to follow this safety instructions could result in fire, electric shock, or other injury or damage.

Proper handling uRAD contains sensitive electronic components. Do not drop, disassemble, crush, bend, deform, puncture, shred, microwave, incinerate, paint, or insert foreign objects into uRAD.

Water and wet locations Do not expose uRAD to water or rain, or handled near washbasins or other wet locations without a proper case. Take care not to spill any food or liquid on uRAD. In case uRAD gets wet, allow it to dry thoroughly before turning it on again. Do not attempt to dry uRAD with an external heat source, such as a microwave oven or hair dryer.

uRAD repairs Never attempt to repair or modify uRAD by yourself. Disassembling may cause damage that is not covered under the warranty. If uRAD is damaged, malfunctions, or comes in contact with liquid, contact us at contact@urad.es.

Radio frequency interference Observe signs and notices that prohibit or restrict the use of radio frequency devices. Emissions from uRAD can negatively affect the operation of other radio frequency equipment operating in the same frequency band. Turn off uRAD when use is prohibited, such as traveling in aircraft, or when asked to do so by authorities.

Important Handling Information

WARNING: Failure to follow this handling instructions could result in damage to uRAD or other property.

Carrying uRAD contains sensitive electronic components. Do not bend, drop or crush it.

Cleaning To clean use a soft lint-free tip and isopropyl alcohol. Dust can be removed with compressed air of low power.

Plugging Never force the connector or apply excessive pressure because this may cause damage that is not covered under the warranty. Check for obstructions.

Operating Temperature Keeping uRAD within acceptable temperatures. uRAD components operate from -40ºC to 85ºC but we recommend operates uRAD in the range from -20ºC to 55ºC.

Disposal and Recycling Information Your uRAD must be disposed of properly according to local laws and regulations. Because this product contains electric components, the product must be disposed of separately from household waste. Contact your local authorities to learn about recycling options.

Product Warranty 5

Manufacturing

All components and solder alloys used in this product comply with the RoHS Directive. The RoHS Directive prevents all new electrical and electronic equipment placed on the market in the European Economic Area from containing more than agreed levels of lead, cadmium, mercury, hexavalent chromium, poly-brominated biphenyls (PBB) and poly-brominated diphenyl ethers (PBDE).

Testing

Each uRAD device is subject to strict tests to make sure they are not faulty:

- First, it is thoroughly tested for short circuits and open connections.
- Second, it is powered to check there are no over-range voltage.
- Then, the microcontroller is programmed and debugged.
- Finally, the board is plugged in a computer and several test programs are run to check its overall functionality.

Limited Warranty Statement

IMPORTANT: BY USING uRAD PRODUCTS YOU ARE AGREEING TO BE BOUNDED BY THE TERMS OF THIS LIMITED WARRANTY STATEMENT. DO NOT USE YOUR PRODUCTS UNTIL YOU HAVE READ THE TERMS OF THE WARRANTY. IF YOU DO NOT AGREE TO THE TERMS OF WARRANTY, DO NOT USE THE PRODUCTS AND RETURN THEM. THIS LIMITED WARRANTY IS THE END-USER'S SOLE AND EXCLUSIVE REMEDY AGAINST uRAD, WHERE PERMITTED BY LAW.

1. Warranties

1.1 uRAD warrants that its products will conform the specifications detailed in the corresponding datasheet. Warranty lasts for 1 year from the date of sale if the device is bought outside the EU and last for 2 years if bought in the EU. uRAD shall not be liable for any defects that are caused by neglect, misuse or mistreatment, including any products that have been altered or modified by any way by the Customer.

1.2 If any uRAD product fails to conform to the warranty set forth above, uRAD's sole liability shall be to replace or repair such products. uRAD's liability shall be limited to products that are determined by uRAD not to conform to such warranty. If uRAD elects to replace or repair such products, uRAD shall be given a reasonable time to provide replacements. Replaced or repaired products shall be warranted for a new full warranty period.

1.3 The Customer agrees no to use uRAD products for any applications or in any components used in life support devices or to operate nuclear facilities or for use in other mission-critical applications or components where human life or property may be at stake. The Customer acknowledges and agrees that any such use is solely at the Customer's risk, and that the Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.4 uRAD may provide technical, applications or design advice. The Customer acknowledges and agrees that providing these services shall not expand or otherwise alter uRAD's warranties, as set forth above, and that no additional obligations or liabilities shall arise from uRAD providing such services.

1.5 uRAD disclaims all other warranties, expressed or implied, regarding products, including, but not limited to, any implied warranties of merchantability or fitness for a particular purpose.

1.6 The Customer acknowledges and agrees that the Customer is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning the products and any use of uRAD products in the Customer's applications, not with- standing any applications-related information or support that may be provided by uRAD.

1.7 In no event shall uRAD be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether uRAD has been advised of the possibility of such damages. This section will survive the termination of the warranty period.