



# User Manual

AUTOMOTIVE version



# Contents

- 3 Chapter 1: About uRAD**
  - 3 Additional Information
  
- 4 Chapter 2: uRAD Basics**
  - 4 Hardware Description
  - 8 Firmware Description
  - 9 Main Features
  
- 11 Chapter 3: Uploading Firmware**
  - 11 Using the UniFlash Tool
  
- 14 Chapter 4: Demo Visualizer**
  - 14 Using the mmWave Demo Visualizer App
  
- 17 Chapter 5: Programming**
  - 17 sdk\_demo\_USB.py
  - 18 sdk\_demo\_USB\_GUI.py
  - 18 sdk\_demo\_USB\_temperature.py
  - 19 sdk\_demo\_single\_UART.py
  - 20 Configuring uRAD
  - 21 Receiving Data
  - 23 Other Laboratories
  
- 24 Chapter 6: Safety & Handling**
  - 24 Important Safety Information
  - 25 Important Handling Information
  
- 26 Chapter 7: Product Warranty**
  - 26 Manufacturing and Testing
  - 26 Limited Warranty Statement

# About uRAD

# 1

## Congratulations on purchasing uRAD



**WARNING:** To avoid any injury or damage, read all operating instructions in this guide and, specially, the safety and warranty information in “Chapter 6: Safety and Handling” and “Chapter 7: Product Warranty”, before using uRAD.

Through this manual you will learn how to use uRAD for an unlimited number of applications related with measuring distance, velocity and angle of any element of your surrounding world. uRAD is conceived as an evaluation platform which brings to you radar detecting technology with a simple but high-performance device.

## Additional Information

Hardware released:	URADAWR1843AOP20	15/07/2022
Purchase:	<a href="http://www.uRAD.es/en/">www.uRAD.es/en/</a>	
Technical specifications:	<a href="http://www.uRAD.es/en">www.uRAD.es/en</a>	
Software download:	<a href="http://www.uRAD.es/en/mi-cuenta/downloads">www.uRAD.es/en/mi-cuenta/downloads</a> (only with purchase)	
Contact:	<a href="mailto:contact@uRAD.es">contact@uRAD.es</a>	

Last version: 16/01/2023

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

uRAD AUTOMOTIVE is a tiny device which includes outstanding radar technology. However, its operation is simple to understand and easy to control.

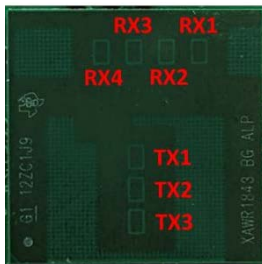
## Hardware Description

uRAD is a multilayer printed circuit board composed by many elements:

### 1. Radar chipset

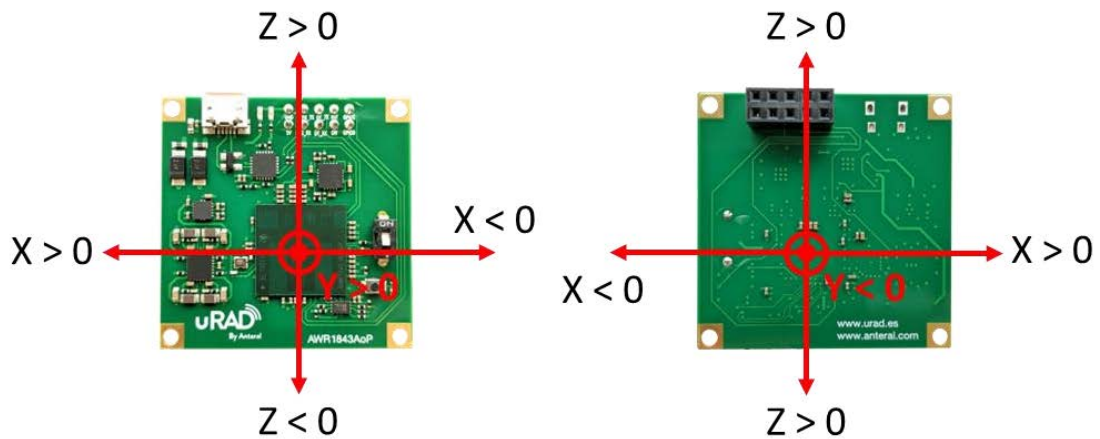
uRAD core is the integrated radar chipset **AWR1843AoP** by **Texas Instruments** (<https://www.ti.com/product/AWR1843AOP>). This integrated circuit is one of the most advanced radar chipsets in the market because of its high accuracy, small size and sensing simplified.

The AWR1843AoP chip has the antennas integrated on it (Antennas on Package). It has 3 transmitter and 4 receiver antennas. The emitted signal is perpendicular to this chip. Therefore, you should point the top face to the direction of interest. Be careful because it does not emit backwards.



**WARNING:** Do NOT cover the antennas with any metallic or electronic element, nor electromagnetic absorber. Most thin plastics are practically invisible to emission.

Default firmware consider the orientation of the chip as the following image. With this orientation, XYZ coordinates of the detected point cloud are as indicated in the image. The coordinate (0,0,0) is the center of the chip. For an observer placed behind the radar, X positive is at the right of the radar, Z positive is above the radar and Y positive is at the front of the radar.



## 2. 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. It corresponds with the internal UART channels of the AWR1843AoP chip: MSS\_UARTA\_TX (ball number U16) and MSS\_UARTA\_RX (ball number V16).
- **Standard COM** is the port for getting the data from the radar. It corresponds with the internal UART channels of the AWR1843AoP chip: MSS\_UARTB\_TX (ball number E2) and MSS\_UARTB\_RX (ball number U12).



*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 340 mA.

### 3. 10-pin connector

The 10-pin female header connector has two UART channels, one on/off, 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. It corresponds with the internal UART channel of the AWR1843AoP chip MSS\_UARTA\_TX (ball number U16).
- 4 **UART RX config**: RX line of the UART port used for configuration. It corresponds with the internal UART channel of the AWR1843AoP chip MSS\_UARTA\_RX (ball number V16).
- 5 **UART TX data**: TX line of the UART port used for data output. It corresponds with the internal UART channel of the AWR1843AoP chip MSS\_UARTB\_TX (ball number E2).
- 6 **UART RX data**: RX line of the UART port used for data output. It corresponds with the internal UART channel of the AWR1843AoP chip MSS\_UARTA\_RX (ball number U12).
- 7 **RESET**: with a 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**: this pin is internally connected to ball number L3 of the AWR1843AoP chip. Therefore, it can be programmed with a custom firmware as a General Purpose Input/Output pin. Default firmware does not use this pin.
- 10 **GPIO0**: this pin is internally connected to ball number M2 of the AWR1843AoP chip. Therefore, it can be programmed with a custom firmware as a General Purpose Input/Output pin. Default firmware does not use this pin.



**WARNING:** Pins 3 and 5 are TX lines from the chip side. Therefore, they are the RX lines from a computer (or any controller device) perspective. Vice-versa with the RX lines of pins 4 and 6.

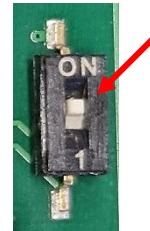
## 4. DIP switch

uRAD has a dip switch for establishing the radar mode when the device is powered on (Sense On Power). It connects with the SOP2 line of the chip. There are two modes that can be set, functional and flashing. See how to set the switch for the different modes:

Functional	Flashing
OFF (low)	ON (high)



Functional



Flashing

Flashing is for upload new firmware and functional is for automatically running the firmware.

## 5. Reset button

The board has a physical reset button to perform a reset of the device. A reset can be useful, for example, after uploading the firmware and changing the DIP switch to functional mode, to start working with the device without power off.

## 6. LED

The board has two LED:

- Red LED: it is connected to the 5V line. It indicates the radar is being powered
- Green LED: it is connected to UART TX data line. It indicates the radar is ready for transmitting data.

In Flashing mode, only red LED must be ON. In Function mode, both LED must be ON.

## 7. Four holes in the corners

The four holes in the corners are connected to ground. Therefore, they are very useful for heat dissipation. However, take care of this aspect if you use it for clamping purpose.

## Firmware Description

As you already know, uRAD is based on the AWR1843AoP chip from Texas Instruments. Therefore, any firmware that it is available for this chip can be easily uploaded by the micro-USB connector.

By default, your uRAD has the **Out of Box Demo** firmware of the **Software Development Kit 3.6 (MMWAVE-SDK 3.6)** of Texas Instruments already uploaded. This firmware is very useful because you can:

- Upload the configuration to uRAD and start receiving the point cloud of the detected targets very easily.
- Use the **mmWave Demo Visualizer** tool to visualize in real time the detected objects and other information in real-time. In **Chapter 4** you will learn how to use it.

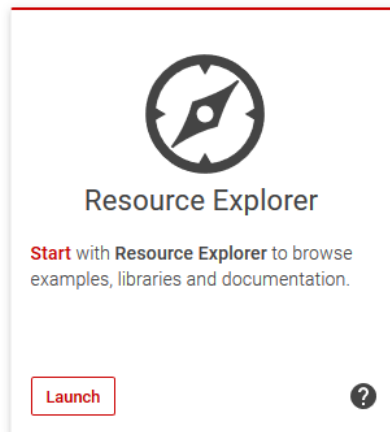


**ADVICE:** download all software and documentation of the **MMWAVE-SDK** in this link: <https://www.ti.com/tool/MMWAVE-SDK>.

The binary of this firmware *out\_of\_box\_1843\_aop.bin* is also included with the purchase.

Moreover, you can upload any firmware of the multiple laboratories that are already programmed for several specific applications (the **Out of Box Demo** is also there):

1. Go to **Texas Instruments Device Tools** website (<https://dev.ti.com/>)
2. Register as user and launch the **Resource Explorer**.



3. In the folder tree of the left part, go to *mmWave radar sensors > Embedded Software > Radar Toolbox > Examples* and you will find all information and software.

Finally, you can also create your own firmware or modify an existing one with the **Code Composer Studio**, an integrated development environment (IDE) that supports Texas Instrument's microcontroller and embedded processors portfolio. (<https://www.ti.com/tool/CCSTUDIO>).



## Main Features

Basically, a radar is a device that detects elements which are in its action radius. It works emitting an electromagnetic wave to the air. This wave is reflected by the target and comes back to the radar. The shape of the going and coming wave as well as the radar architecture determines the type of the radar.

uRAD detects the 3D spatial position of the target and the relative radial velocity to it. Moreover, the amount of reflected power and the noise level is also given, that serves as an estimation of the object size or the detection level. Whether not enough power is reflected, the element is not detected. Therefore, very small or very far targets will not be seen.

The main features of uRAD are:

<i>Parameter</i>	<i>Min.</i>	<i>Typ.</i>	<i>Max.</i>	<i>Unit</i>
<b>Operating conditions</b>				
Supply voltage	4.5	5	5.5	V
Supply current		340		mA
Digital signals		3.3		V
Operating temperature	-20		+55	°C
<b>RF Parameters</b>				
Frequency bandwidth	77		81	GHz
Output power		16		dBm
Number of transmitter antennas		3		
Number of receiver antennas		4		
Azimuthal field of view		160		deg
Elevation field of view		160		deg

<i>Parameter</i>	<i>Value</i>
<b>Performance</b>	
Mode of operation	Frequency Modulated Continuous Wave
Output data	Distance, velocity, angle and SNR
Distance range	70 m (car), 40 m (people)
Distance accuracy	±1 mm
Distance resolution	5 cm
Velocity range	45 m/s
Velocity accuracy	± 0.15 m/s
Velocity resolution	0.06 m/s
Angle accuracy	1 deg
Azimuthal angle resolution	30 deg
Elevation angle resolution	38 deg

Distance and velocity range, accuracy and resolution values depend on the configuration parameters of the transmitted waveform.

Resolution indicates the minimum distance, velocity or angle that two targets with similar reflectivity must be separated to be discerned as a single target each one.



**WARNING:** Detected velocity is only radial velocity: the component of the target's velocity that points in the direction of the line connecting the object and the radar.

# Uploading Firmware

# 3

In this chapter, you will learn how to upload new firmware to uRAD.

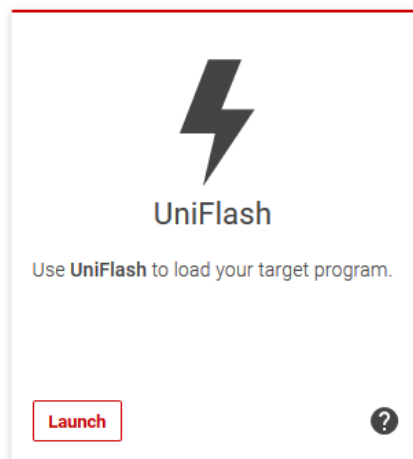
Because of uRAD is based on the AWR1843AoP chip, the best way of uploading new firmware is to use the **UniFlash** tool of Texas Instruments.

## Using the UniFlash Tool

**UniFlash** is a standalone tool used to program on-chip flash memory on TI MCUs and on-board flash memory for Sitara processors. **UniFlash** has a GUI, command line, and scripting interface. It is available free of charge.

**UniFlash** has a desktop version and an online version.

- Desktop version is available at <https://www.ti.com/tool/download/UNIFLASH>
- Online version is available going to **Texas Instruments Device Tools** website (<https://dev.ti.com/>), register as user and launch the **UniFlash** tool.

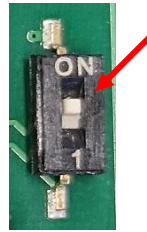


*ADVICE:* find the **UniFlash** Quick Start Guide at [https://downloads.ti.com/ccs/esd/uniflash/docs/latest\\_qsguide.html](https://downloads.ti.com/ccs/esd/uniflash/docs/latest_qsguide.html).

Follow these steps to upload firmware to uRAD:

1. Set the DIP switch in Flashing mode:

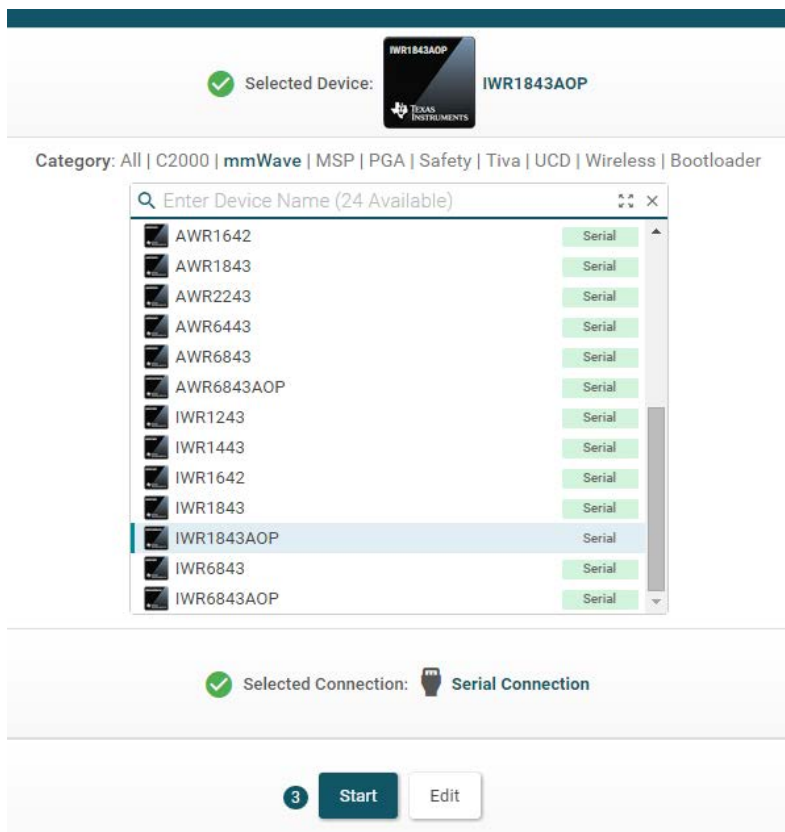
**Flashing**  
ON (high)



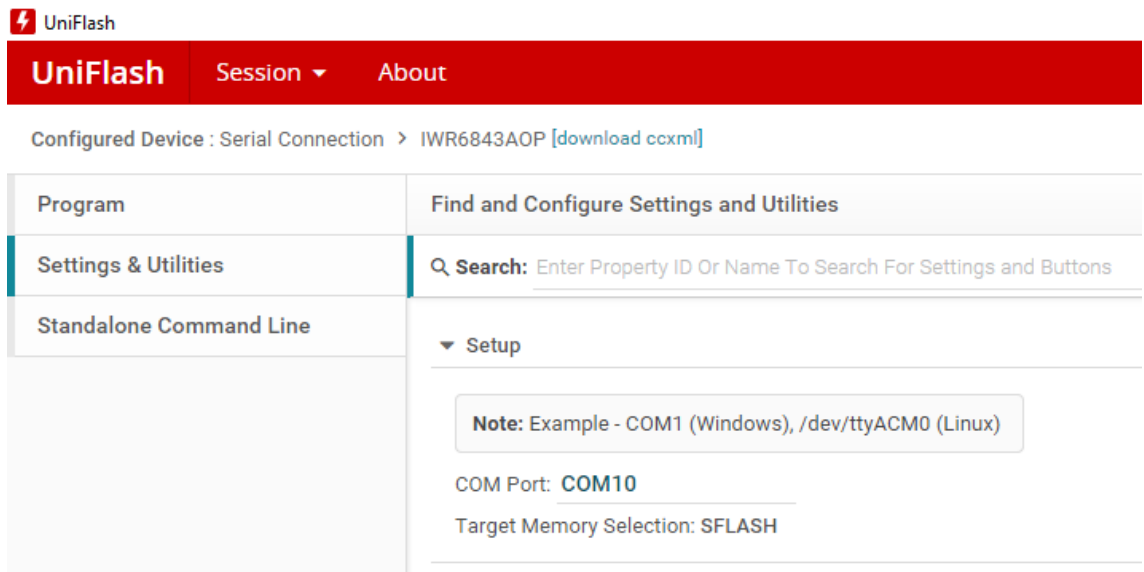
2. Connect uRAD with the USB connector and identify the COM ports with the Device Manager.

- ▼ Puertos (COM y LPT)
  - Silicon Labs Dual CP2105 USB to UART Bridge: Enhanced COM Port (COM10)
  - Silicon Labs Dual CP2105 USB to UART Bridge: Standard COM Port (COM9)
- Enhanced Port is the configuration port
- Standard Port is the data port

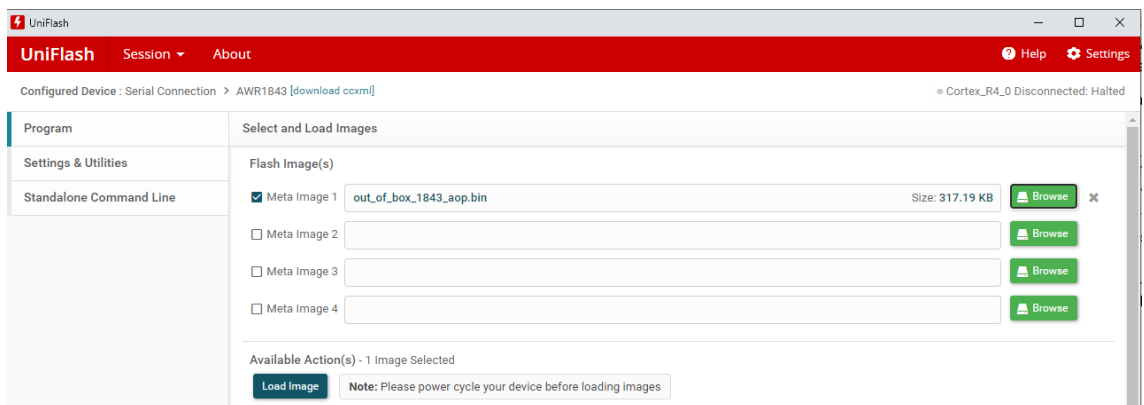
3. Launch **UniFlash**. Select the IWR1843AOP (AWR1843AOP is not available yet but it is equivalent) and click *Start*



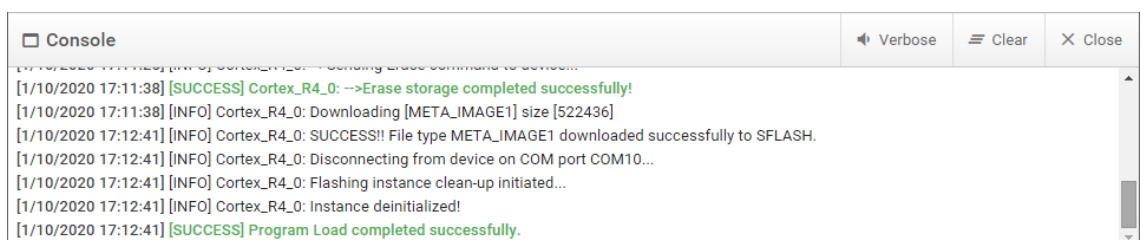
- Go to *Settings & Utilities* and write the number of your configuration port (Enhanced Port) in the COM Port box.



- Go to *Program*, browse your *.bin* file and click on *Load Image*.



- If everything goes well, you will see in the *Console* that the firmware has been successfully uploaded.



- Change the DIP switch to functional mode, press the reset button and start working.

# Demo Visualizer

# 4

## Read how to configure uRAD and visualize point cloud

The easiest and fastest way of testing uRAD is using the **Demo Visualizer**.

### Using the mmWave Demo Visualizer App

The **mmWave Demo Visualizer** is an app developed by Texas Instruments to be used in conjunction with the **mmWave SDK demo** firmware running on uRAD. The app is browser-based and can be run on any operating system. Chrome browser is recommended for the best plotting performance. Go to this link for launching the app:

[https://dev.ti.com/gallery/view/mmwave/mmWave\\_Demo\\_Visualizer/ver/3.6.0/](https://dev.ti.com/gallery/view/mmwave/mmWave_Demo_Visualizer/ver/3.6.0/)

**New version 4.2 is not compatible. Use version 3.6.** A desktop version is also available at: <https://dev.ti.com/gallery/>

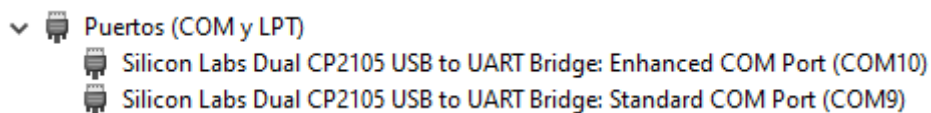


**ADVICE:** the *mmWave Demo Visualizer User's Guide* can be found at <https://www.ti.com/tool/MMWAVE-STUDIO>.

First of all, be sure that:

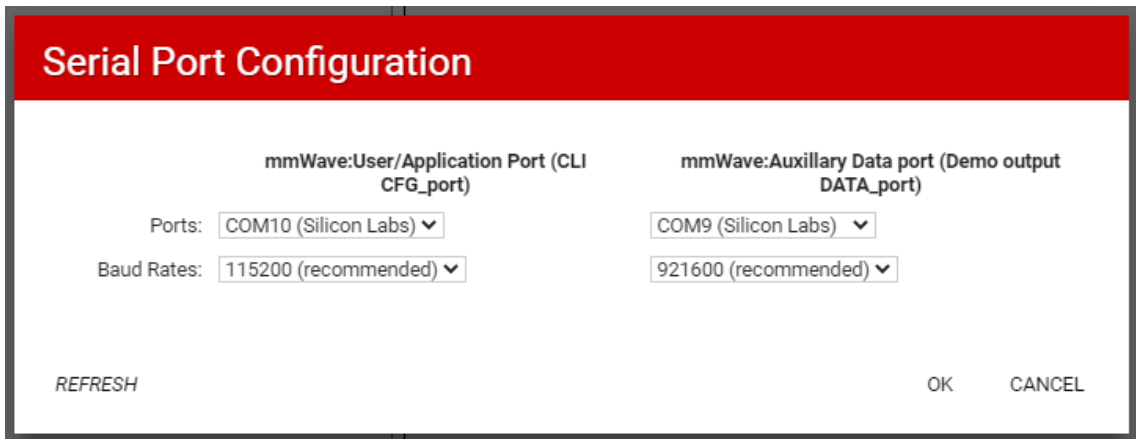
- **Out of Box Demo** firmware is uploaded in uRAD.
- DIP-switch is in functional mode.

1. Connect uRAD by the USB and identify with the *Device Manager* the COM ports:



- Enhanced port is the configuration port (CFG\_port)
- Standard port is the output data port (DATA\_port)

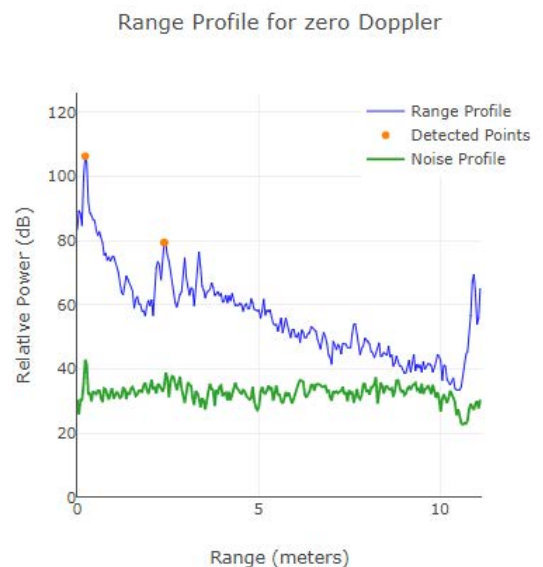
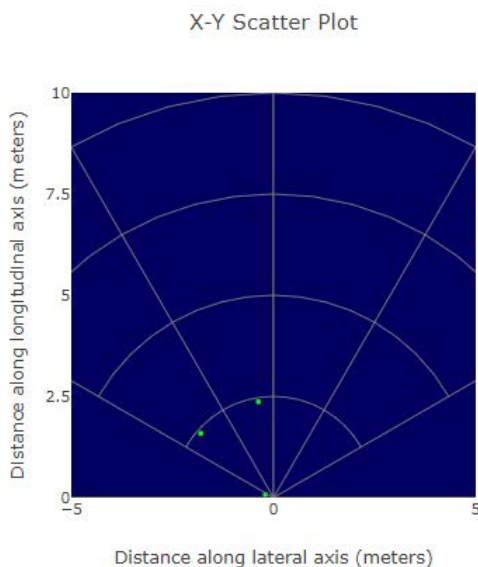
2. In the browser Demo Visualizer app, go to **Menu > Options > Serial Port** and set up the COM ports as the following image. You can increase the baud rate of the data port by selecting “*custom*” in the drop-down menu and setting the number manually. Maximum value is 3125000. We do not recommend to change the baud rate of the configuration port.



3. In the Configure tab select:
  - Platform: xWR18xx\_AOP
  - SDK version: 3.6
  - Other configuration parameters according your desirable configuration.
4. Click on SEND CONFIG TO MMWAVE DEVICE button to send the configuration and start measuring.  
 You can also save in a .cfg file the configuration with the SAVE CONFIG TO PC button. This is very useful to see what the configuration parameters are and how are the commands that you need to send to uRAD to configure it.

**ADVICE:** download the *mmwave\_sdk\_user\_guide* at [https://software-dl.ti.com/ra-processors/esd/MMWAVE-SDK/latest/index\\_FDS.html](https://software-dl.ti.com/ra-processors/esd/MMWAVE-SDK/latest/index_FDS.html). Go to *Section 3.4* to learn about the configuration parameters and file format.

5. Visualize in the Plots tab the 2D or 3D point cloud and range profile for targets with zero velocity and other plots or windows with configuration options.



Bottom right window allows you to change some configuration parameters in real time without stopping the sensor.

The screenshot shows the 'Real-Time Tuning' interface with the following settings:

- Group Peaks from Same Object:**  Range Direction,  Doppler Direction
- Additional Algorithm Processing:**  Remove Static Clutter
- CFAR Range Threshold (0-100dB):** Slider set to 0
- Doppler Range Threshold (0-100dB):** Slider set to 0
- Field of View:**
  - Angle of arrival (degrees):** Azimuth (Min: -90, Max: 90), Elevation (Min: -90, Max: 90). Buttons: RESET, SEND
  - Range (m):** (Min: 0, Max: 8.92). Buttons: RESET, SEND

You can also create or modify a .cfg file and load it to uRAD with the LOAD CONFIG FROM PC AND SEND button as well as record 10 seconds of the output data in a .dat file.

The screenshot shows the configuration interface with the following settings and buttons:

- Range Depth:** 10
- Range Width:** 5
- Range Profile Y max:** 2e6
- Range Profile Log Scale:**
- File Size Max (MB):** 1
- Record time max (s):** 10
- Buttons:** SENSOR START, RECORD START, LOAD CONFIG FROM PC AND SEND, EXPORT TUNED PROFILE



# Programming

# 5

## In this chapter, basic programming of uRAD is explained

We have created for you some Python demo programs to directly work with uRAD without using the **Demo Visualizer** app. The programs, together with some documents and configuration files are in the folder *uRAD\_out\_of\_box\_demo* delivered with the purchase.

### out\_of\_box\_demo\_USB.py

The *out\_of\_box\_demo\_USB.py* example code includes the basis to work with uRAD:

- Send the configuration by the configuration COM port.
- Start receiving the output data by the data COM port.
- Save output data in a .txt file.
- Print in the Python console the results.

Therefore, this simple code is very useful for developing purpose because it may be the starting point for any application.

At the beginning of the program, you must select whether you can save in the .txt file the results and/or printing them in the console.

```
savePointCloud = True
printPointCloud = True
```

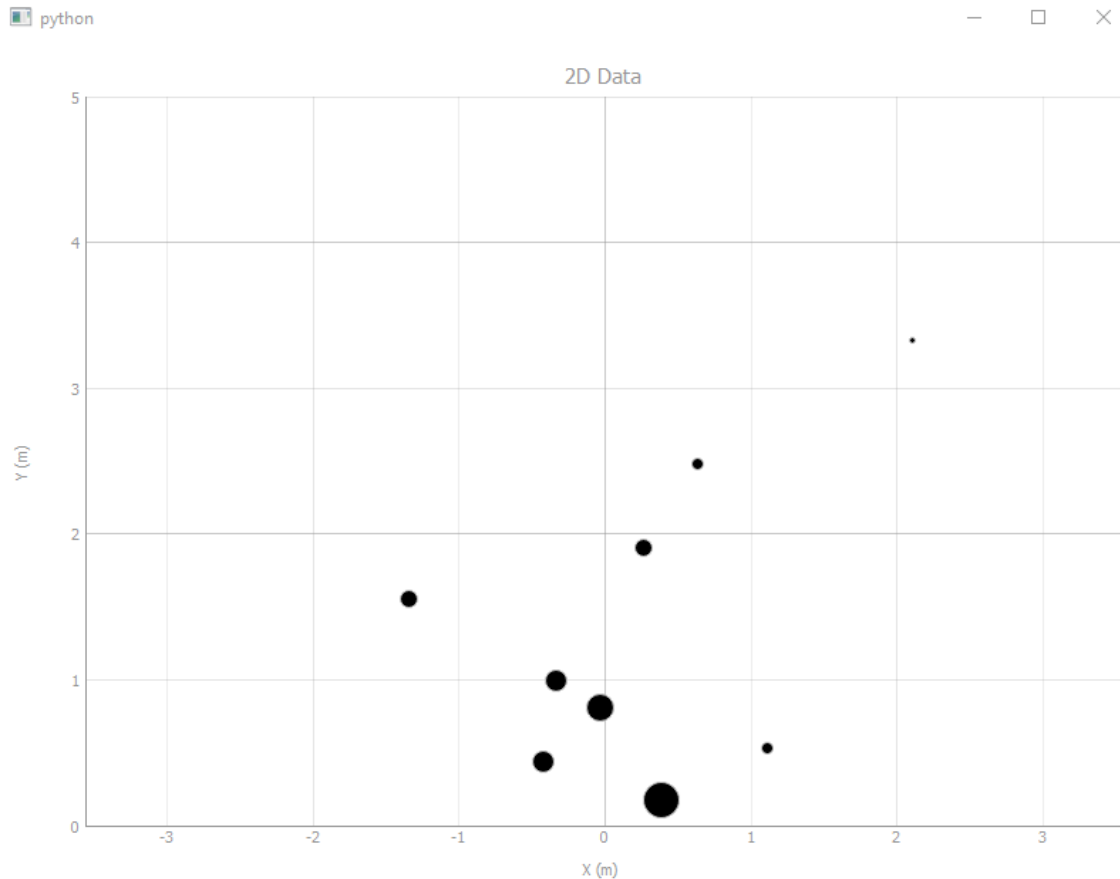
And the name of the COM ports as well as the name of the configuration and results files.

```
configPort_name = 'COM1'
dataPort_name = 'COM2'
configFile_name = './chirp_config/chirp_config.cfg'
pointCloud_fileName = './output_files/PointCloud.txt'
```

This program can be used with uRAD connected by the USB or with the 14-pin header connector. Just take into account to write the right name in the configuration and data port.

## out\_of\_box\_demo\_USB\_GUI.py

The *out\_of\_box\_demo\_USB\_GUI.py* is similar to the previous one but the point cloud is plotted in a 2D graph showing the X and Y components in real time. Therefore, you have to take into account the orientation of the PCB to correctly identify X and Y axes.



## out\_of\_box\_USB\_temperature.py

The *out\_of\_box\_demo\_USB\_temperature.py* is similar to *out\_of\_box\_demo\_USB.py* but with the particularity that the temperature of the different parts of the radar **AWR1843AoP** radar chipset is sent. The *guiMonitor* line of the *chirp\_config.cfg* file must be modified with a 1 at the end to ask the radar to send the temperature.

```
guiMonitor -1 1 0 0 0 0 1
```

Temperature values can be print on screen and also saved in a .txt with the corresponding variables.

```
printTemperature = True
saveTemperature = True
```

## out\_of\_box\_demo\_single\_UART.py

The *out\_of\_box\_demo\_single\_UART.py* example code is very similar to the USB one but with two modifications:

- It uses the RESET pin whether it is selected with *reset = True*. This simple piece of code makes one down and one high digital pulse in the RESET pin. Each time a rising edge is sent to the RESET pin, uRAD does a reset (this pin has a pull-up resistor that makes it being always high unless a low signal is applied). It has been programmed according to Python running in a Raspberry Pi where the uRAD RESET pin is connect to Raspberry Pi pin number 18.

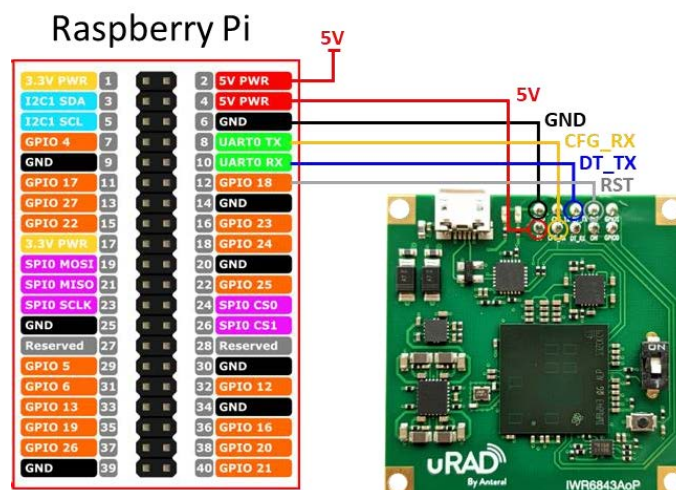
```
if (reset):
    from gpiozero import OutputDevice
    PinReset = OutputDevice(18)
    PinReset.off()
    sleep(10e-3)
    PinReset.on()
    sleep(1)
```

- It is intended for being used with the 10-pin header when only **one** UART port is available in the master device. In this case, sending the configuration and receiving the data are done with a single UART port. It is important to note that sending configuration has a different baudrate that receiving data. In this case, Raspberry Pi only has one UART port which is defined with */dev/serial0*.

```
reset = True
savePointCloud = True
printPointCloud = False

Port_name = '/dev/serial0'
configFile_name = './chirp_config/chirp_config.cfg'
pointCloud_fileName = './output_files/PointCloud.txt'
```

This sketch depicts how to connect uRAD with the Raspberry Pi using a single UART.



## Configuring uRAD

Together with the Python demo program, a configuration file named ***chirp\_config.cfg*** is attached. This file contains a command list that is necessary to send for a correct configuration.



**ADVICE:** examples of configuration files are available in the Texas Instruments SDK. You can also create your own with the **Demo Visualizer** app.

Take into account these considerations:

- Each line describes a command with parameters.
- Lines starting by % are not valid. They are for information purposes.
- Send each command line as a string and do not forget to send `\n` after each line.
- After sending each line, you will receive an ACK for confirming proper command.
- Always start with a *sensorStop* and finish with a *sensorStart* command.
- After sending each line, you will receive and ACK for confirming proper configuration.



**WARNING:** the command list depends on the firmware. The ***out\_of\_box\_demo\_XXX.py*** and ***chirp\_config.cfg*** are only valid for the **mmWave SDK Demo** firmware of the *Software Development Kit*.



**ADVICE:** find the **mmwave\_sdk\_user\_guide** inside the doc folder. Go to *Section 3.4* to learn about each configuration parameters and file format. This particular information has been extracted in a separate pdf named **standard\_mmwave\_sdk\_commands**.

## Receiving data

Once uRAD is configured, it starts sending the detected information through the data COM port. The packet structure is the following:

**Header (40 bytes) + TLVs + Padding Bytes (for 32-byte align)**

### 1. Frame Header (40 bytes)

A frame header is sent at the start of each packet. Use the Magic Word to find the start of each packet.

Value	Type	Bytes	Details
Magic Word	uint16_t	8	Output buffer magic word (sync word). It is initialized to {0x0102, 0x0304, 0x0506, 0x0708}
Version	uint32_t	4	SDK Version represented as (MajorNum x 2 <sup>24</sup> + MinorNum x 2 <sup>16</sup> + BugfixNum x 2 <sup>8</sup> + BuildNum)
Total Packet Length	uint32_t	4	Total packet length including frame header length in Bytes
Platform	uint32_t	4	Device Type
Frame Number	uint32_t	4	Frame number (resets to 0 when device is power cycled or rest. Not when sensor stop/start is issued)
Time	uint32_t	4	Time in CPU cycles when the message was created
Num Detected Obj	uint32_t	4	Number of detected objects (points) for the frame
Num TLVs	uint32_t	4	Number of TLV items for the frame
Subframe Number	uint32_t	4	0 if advanced subframe mode not enabled, otherwise the sub-frame number in the range 0 to (number of subframes – 1)

### 2. TLV Header (8 bytes)

The number of TLVs in the frame packet is extracted from the Frame Header. For each TLV in the packet, there is a TLV Header containing Type and Length information.

- The Type identifier indicates what kind of information is contained in the payload.
- The Length value gives the length of the payload.

Value	Type	Bytes	Details
Type	uint32_t	4	Indicates types of messages contained in payload
Length	uint32_t	4	Length of the payload in Bytes (does not include length of the TLV header)

### 3. TLV Type identifier

With the configuration file, you enable or disable whether the TLV type is included in the output frame packet. There are nine types of TLV. Here we describe Type 1 and 7 which are the most interesting related with the point cloud of the detected objects.

Type Identifier	Value Type	Conditions for output
1	Detected Points	<detected objects> is set to 1 or 2 AND there are detected objects for the frame, else this type is not sent for that frame
7	Detected Points	<detected objects> is set to 2 AND there are detected objects for the frame, else this type is not sent for that frame



**ADVICE:** the information of the remaining TLVs is shown in the *understanding\_oob\_uart\_data.pdf* document in the *doc* folder.

### 4. TLV Payload

- Type 1: detected points
  - Length: 16 Bytes x Num Detected Obj
  - Value: array of detected points. Each point is represented by 16 bytes giving position and radial Doppler velocity

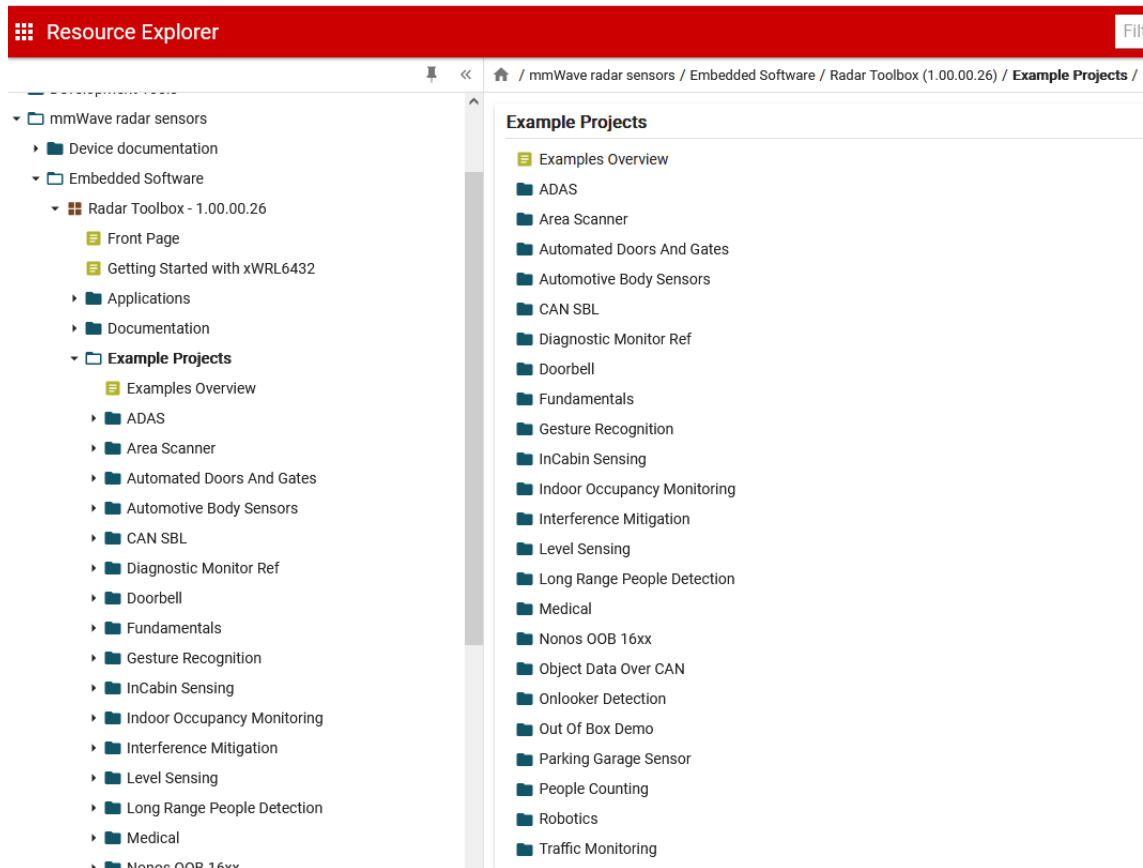
Value	Type	Bytes
X [m]	float	4
Y [m]	float	4
Z [m]	float	4
Doppler [m/s]	float	4

- Type 7: detected points
  - Length: 4 Bytes x Num Detected Obj
  - Value: payload consists of 4 bytes for each point in the point cloud.

Value	Type	Bytes
SNR [dB]	uint16_t	2
Noise [dB]	uint16_t	2

## Other Laboratories

As we already saw in [Chapter 2](#), Texas Instruments offers different laboratories with specific firmware and software that can be found in the **Texas Instruments Device Tools** website (<https://dev.ti.com/>) in the **Resource Explorer** section.



These laboratories exploit the most of the radar chips for each of those applications. Check which laboratories can be uploaded to the AWR1843AoP or IWR1843AoP and test them.

Take into account that for each particular firmware, the received information as well as the configuration files are different. But all information can be found in the **Resource Explorer**.



**ADVICE:** contact us at [contact@urad.es](mailto:contact@urad.es) for helping with any laboratory or application.

# Safety & Handling

# 6

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](mailto: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 -40°C to 85°C but we recommend operates uRAD in the range from -20°C to 55°C.

**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

# 7

## 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 not 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, notwithstanding 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.