Radar Signal Processing Mastery

Theory and Hands-On Applications with mmWave MIMO Radar Sensors

Date: 7-11 October 2024
Time: 9:00AM-11:00AM ET (New York Time)

Mohammad Alaee-Kerahroodi
Research scientist
SnT, University of Luxembourg
Email: Mohammad.alaee@uni.lu
Website: https://radarmimo.com/

Outline

Time: 9:00AM-11:00AM ET (New York Time)

Lecture	Duration	Date
Lecture 1: Radar Systems Fundamental	**2 Hours**	October 7th , 2024
Lecture 2: Advanced Radar Systems	**2 Hours**	October 8th , 2024
Lecture 3: Practical Radar Signal Processing - Motion Detection	**2 Hours**	October 9th , 2024
Lecture 4: Practical Radar Signal Processing - Breathing and Heart Rate Estimation	**2 Hours**	October 10th , 2024
Lecture 5: Practical Radar Signal Processing – Angle estimation with MIMO radar	**2 Hours**	October 11th , 2024

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Lecture 3

Motion Detection with Infineon BGT60LTR11AIP

Lecture 3: Motion Detection with Infineon BGT60LTR11AIP

What we learn in Lecture 3

- **Getting started with BGT60LTR11AIP**
- **Doppler and Micro-Doppler effects**
- **Motion detection principle**
- **Real-time data measurement**
- **Signal processing and motion detection**

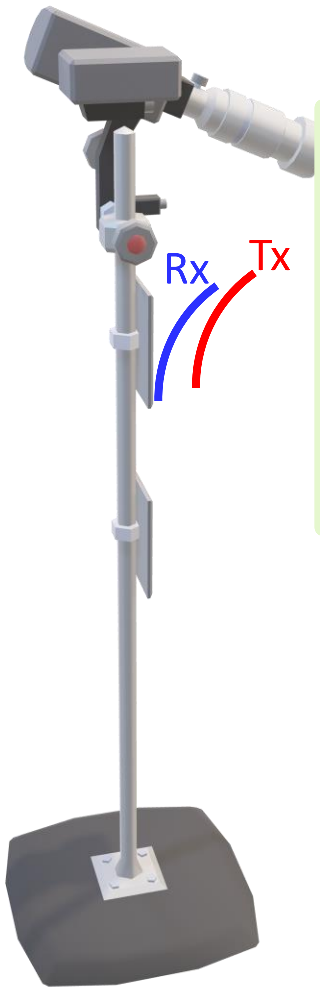
- ❑ Doppler and Micro-Doppler
- ❑ Radar Target Classification based on Micro-Doppler

Scan the QR code for access to the codes

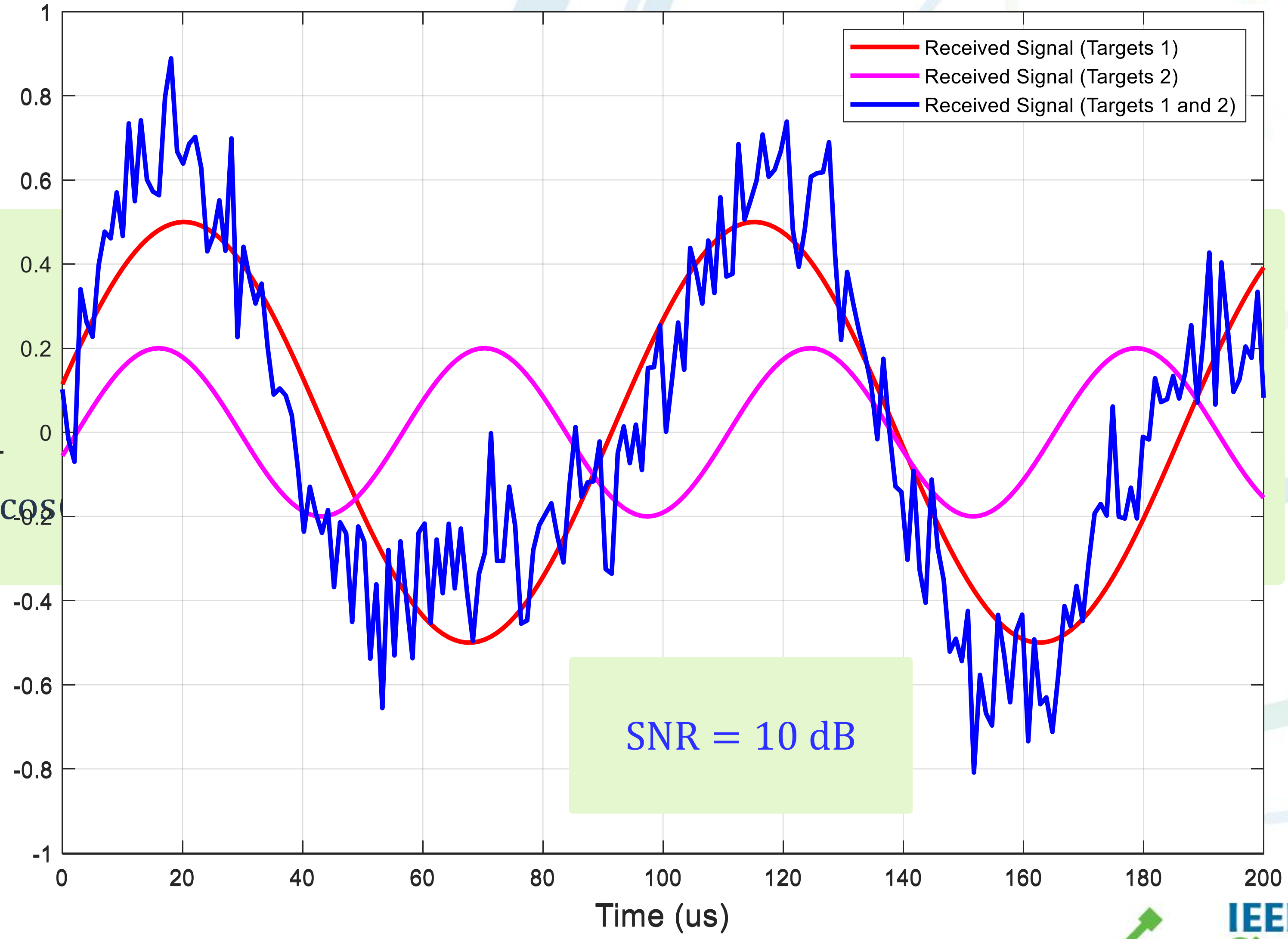
website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Recall from lecture 1 - CW Radar

Doppler Effect



$s_{rx}(t) = A_r\ \cos($



SNR = 10 dB

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Infineon DEMO BGT60LTR11AIP

Product	Description
BGT60LTR11AIP	XENSIV™ 60 GHz first completely autonomous radar sensor for motion sensing
BGT60LTR11SAIP	XENSIV™ 60 GHz first completely autonomous radar sensor for motion sensing
DEMO BGT60LTR11AIP	XENSIV™ BGT60LTR11AIP 60 GHz radar sensor pulsed Doppler demo board
SHIELD_AUTONOM_BGT60	Shield for autonomous operation of BGT60LTR11AIP; directly fits on Arduino MKR board
REF BGT60LTR11AIP M0	Reference design with Cortex®-M0 MCU for data processing
S2GO RADAR BGT60LTR11	Shield2Go version
BGT60TR13C	XENSIV™ 60 GHz radar sensor for advanced sensing
DEMO BGT60TR13C	XENSIV™ 60 GHz radar sensor demo board for advanced sensing
BGT60UTR11AIP	XENSIV™ highly integrated 60 GHz FMCW radar sensor
DEMO BGT60UTR11AIP	XENSIV™ BGT60UTR11AIP 60 GHz radar sensor FMCW demo boar

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Infineon DEMO BGT60LTR11AIP

https://www.infineon.com/cms/en/product/evaluation-boards/demo-bgt60ltr11aip/

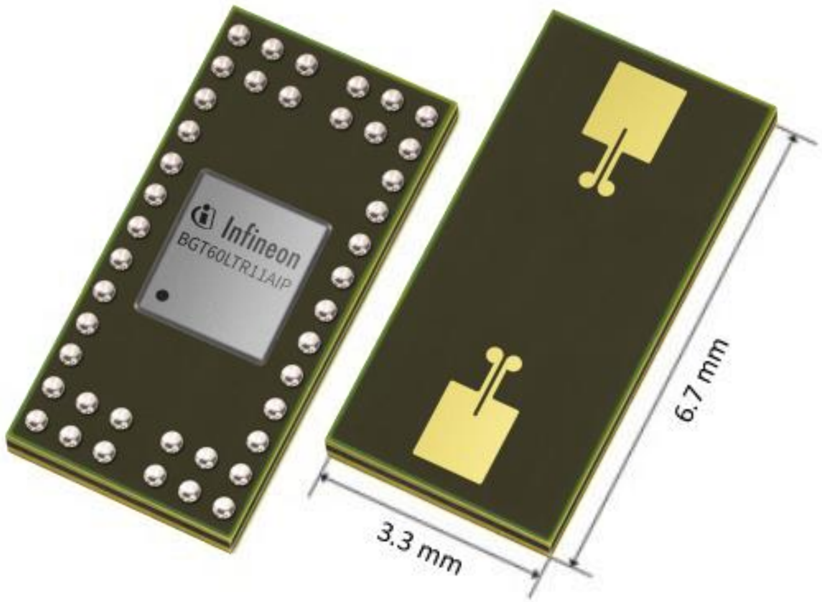
Parametrics	DEMO BGT60LTR11AIP
Antenna	Antennas in package
Board Type	Demo Board
Direction of Motion	Yes
Field of View	90
Frequency min max	61 GHz 61.5 GHz
Max Detection Range	11 m
Min Detection Range	0.6 m
Number of Rx Antennas	1
Number of Tx Antennas	1
Target Application	Automated door openers ; Contactless switches ; Displays such as TVs ; monitors ; laptops or tablets ; Lighting systems and lighting control (mainly indoor lighting) ; Multicopter and drones ; Smart Building ; Smart appliances ; Smart home security and alarm systems including IP cameras ; Smart Home devices ; Air conditioners

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

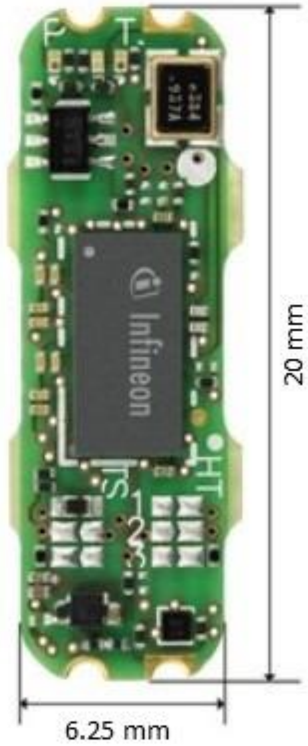
Infineon DEMO BGT60LTR11AIP

https://www.infineon.com/cms/en/product/evaluation-boards/demo-bgt60ltr11aip/

BGT60LTR11AIP MMIC



BGT60LTR11AIP Shield



DEMO BGT60LTR11AIP

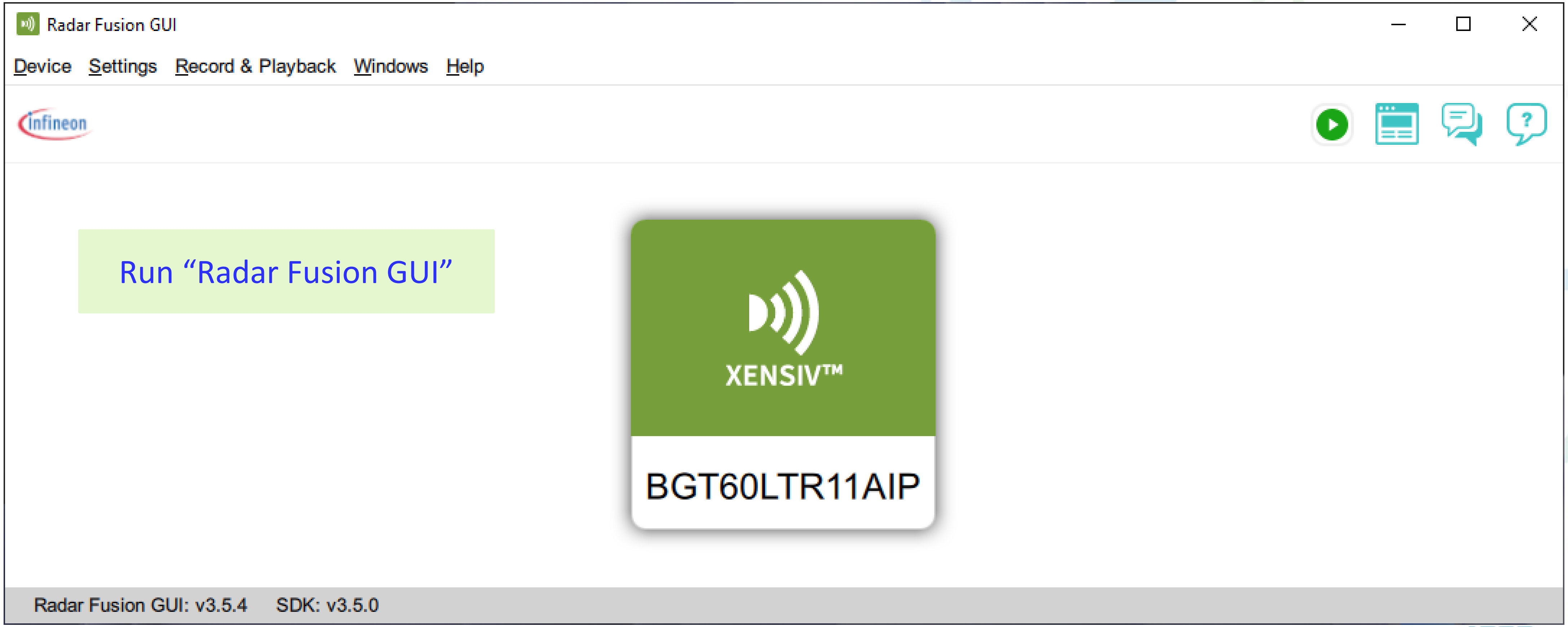
= Radar Baseboard MCU7
+ BGT60LTR11AIP Shield

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

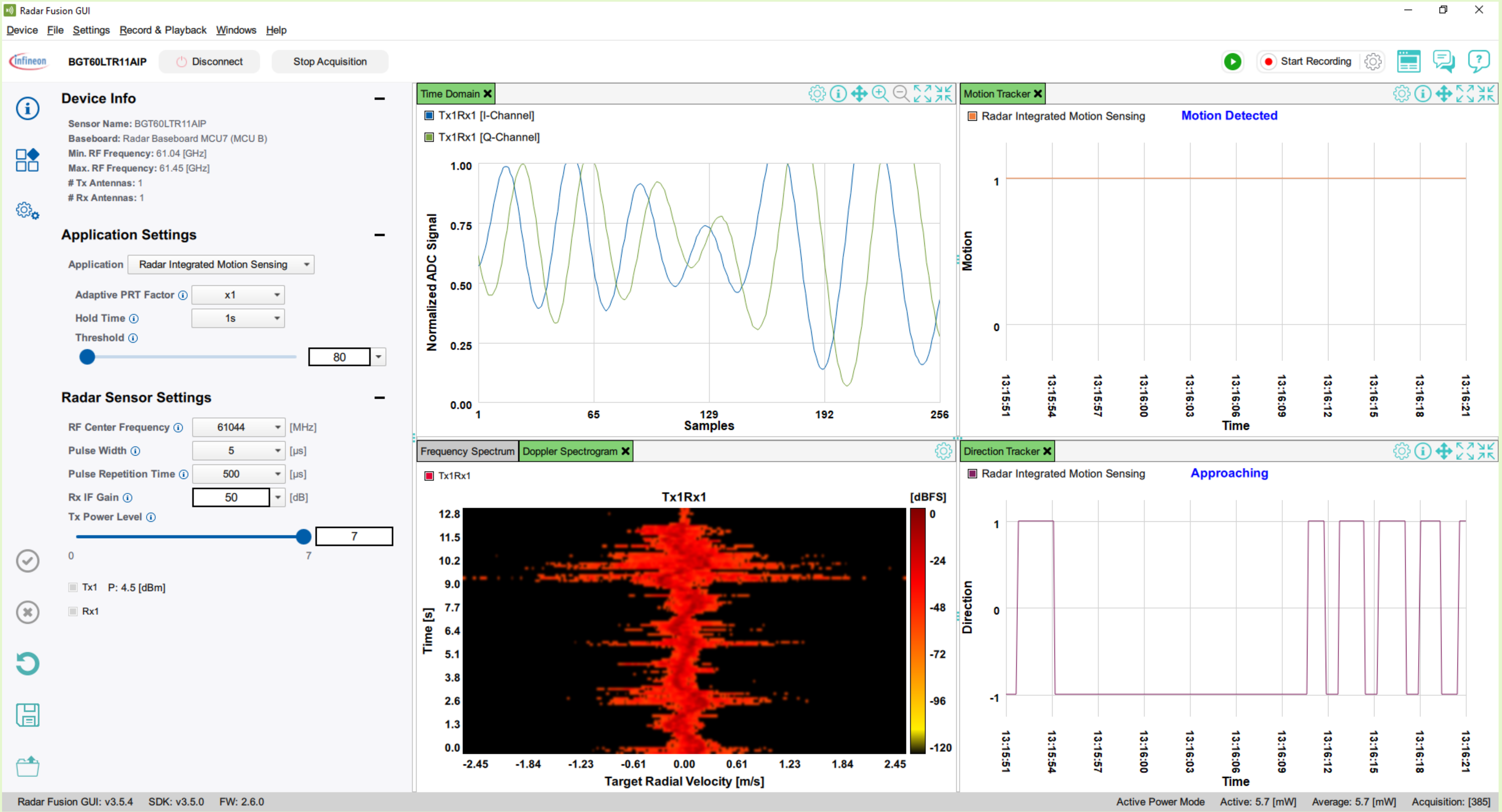
https://www.infineon.com/cms/en/product/evaluation-boards/demo-bgt60ltr11aip/#!designsupport

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board



website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board



website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

Run “Radar Development Kit”

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

Releases Quickstart Software Hardware Documentation

Software

Radar Software Development Kit (SDK)

Download

radar_sdk.zip

The Radar SDK is used to evaluate the following XENSIV™ Radar sensors.:

- FMCW Radars
  - BGT60 UTR11AIP
  - BGT60 TR13C
  - BGT60 ATR24C
- Doppler Radars
  - BGT60 LTR11AIP

Radar SDK allows users to configure and fetch raw data from above mentioned XENSIV™ radar sensors using C/C++, Python and Matlab programming languages. In addition, it contains sophisticated propriety algorithms to solve target detection and localization use-cases, and some useful tools to e.g. data recording and flash firmware etc.

Name
- 3rd_party
- build
- cmake
- CMakeFiles
- doc
- examples
- external
- libs
- python_wheels
- sdk
- tools
- CMakeCache
- CMakeLists
- VERSION

- ifxradarsdk-3.5.0+8c595dbb-py3-none-linux_armv7l.whl
- ifxradarsdk-3.5.0+8c595dbb-py3-none-linux_x86_64.whl
- ifxradarsdk-3.5.0+8c595dbb-py3-none-macosx_10_14_universal2.whl
- ifxradarsdk-3.5.0+8c595dbb-py3-none-win_amd64.whl

```
cd C:\path\to\your\whl\file

pip install ifxradarsdk-3.5.0+8c595dbb-py3-none-win_amd64.whl
```


website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

Name
- 3rd_party
- build
- cmake
- CMakeFiles
- doc
- examples
- external
- libs
- python_wheels
- sdk
- tools
- CMakeCache
- CMakeLists
- VERSION

Name
- c
- matlab
- py

C

Name
- BGT24ATR22
- BGT60ATR24C
- BGT60LTR11AIP
- BGT60TR13C
- BGT60UTR11AIP
- CMakeLists

Matlab

Name
- BGT60ATR24C
- BGT60TR13C
- BGT60UTR11AIP

Python

Name
- BGT60ATR24C
- BGT60TR13C
- BGT60UTR11AIP

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

```
from ifxradarsdk import get_version
from ifxradarsdk.ltr11 import DeviceLtr11
from ifxradarsdk.ltr11.types import Ltr11Config
```

$$f_c = 60 \text{ GHz}$$

$$\lambda = 0.005 \text{ m}$$

$$\text{assume } v_r = 1 \frac{\text{m}}{\text{s}}$$

$$f_d = \frac{2v_r}{\lambda} = \frac{2}{0.005} = 400 \text{ Hz}$$

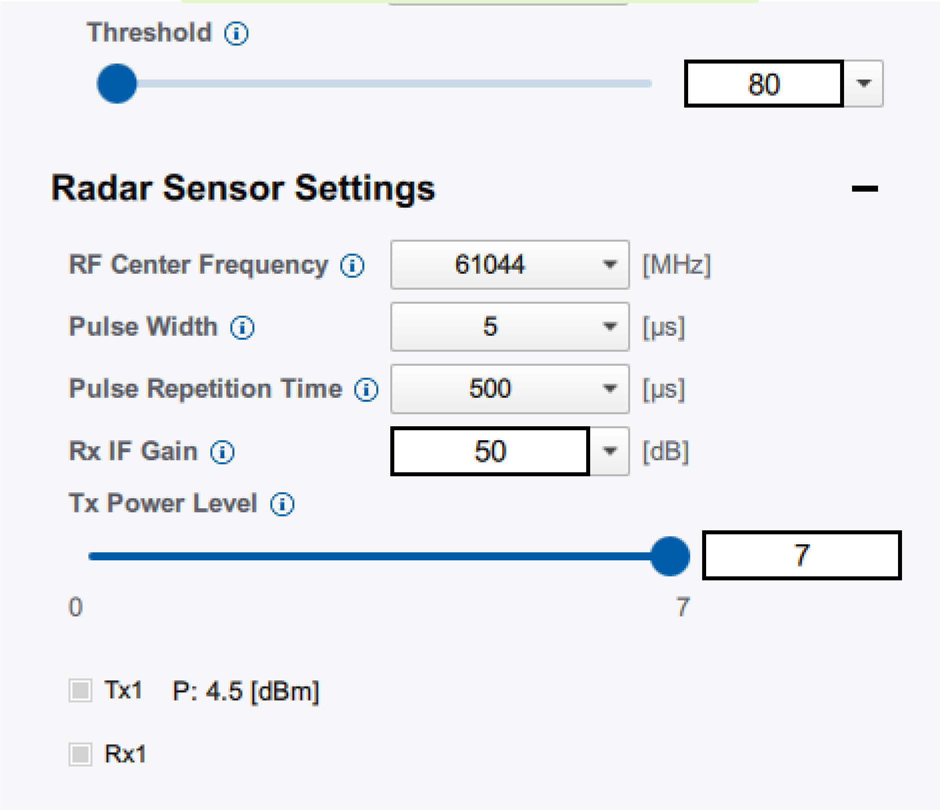
```
prt_index = 1  # 0 = 4000 Hz,  1 = 2000 Hz, 2 = 1000 Hz, 3 = 500 Hz
if prt_index == 0:
    sample_rate = 4000
elif prt_index == 1:
    sample_rate = 2000
elif prt_index == 2:
    sample_rate = 1000
else:
    sample_rate = 500
```

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

"Radar Fusion GUI"




```
config = Ltr11Config(
    aprt_factor=4,
    detector_threshold=80,
    disable_internal_detector=False,
    hold_time=8,
    mode=0,  # 0: continuous wave mode, -- 1: pulse mode
    num_of_samples=num_of_samples,
    prt=prt_index,
    pulse_width=3,
    rf_frequency_Hz=61044000000,
    rx_if_gain=8,
    tx_power_level=7,
)
```

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

Initialize the parameters

```
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

# Initialization
ENABLE_I_Q_PLOT = True
sample_time = 1/sample_rate
num_of_samples = 256
window_time = 1  # second
buffer_time = 5 * window_time  # second
figure_update_time = 25  # m second
num_rx_antennas = 1
raw_data_size = int(buffer_time * sample_rate)
IQ_xaxis = np.linspace(1, buffer_time, raw_data_size)
epsilon_value = 0.00000001
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

Initialize the data queue

```
# data queue
data_queue = queue.Queue()
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

def read_data(device):
    while True:
        frame_contents = device.get_next_frame()
        for frame in frame_contents:
            data_queue.put(frame)
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

```
class myProcessorClass:
   def process_data(self):
      global raw_data
      while True:
         time.sleep(1/sample_rate)
         if not data_queue.empty():
            frame = data_queue.get()
            if np.size(frame) == num_of_samples:
               raw_data = np.roll(raw_data, -num_of_samples)
               raw_data[-num_of_samples:] = frame
      #
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Define the process class

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

```
def generate_iq_plot():
    plot = pg.plot(title='Inphase and Quadrature')
    plot.showGrid(x=True, y=True)
    plot.setLabel('bottom', 'Time [s]')
    plot.setLabel('left', 'Amplitude')
    plot.addLegend()
    plots = [
        ('lightblue', 'Inphase [I]'),
        ('gold', 'Quadrature [Q]')
    ]
    plot_objects = [[] for _ in range(len(plots))]
    for j, (color, name) in enumerate(plots):
        line_style = {'color': color, 'style': [QtCore.Qt.SolidLine, QtCore.Qt.DashLine, QtCore.Qt.DotLine][0]}
        plot_obj = plot.plot(pen=line_style, name=f'{name}')
        plot_obj.setVisible(True)
        plot_objects[j].append(plot_obj)
    return plot, plot_objects
# Usage:
if ENABLE_I_Q_PLOT:
    iq_figure, I_Q_PLOT = generate_iq_plot()
    iq_figure.show()
```

Create plot

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Update plot by timer

```
def update_plots():
    if ENABLE_I_Q_PLOT:
        I_Q_PLOT[0][0].setData(IQ_xaxis, np.real(raw_data))
        I_Q_PLOT[1][0].setData(IQ_xaxis, np.imag(raw_data))
    # ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
timer = QTimer()
timer.timeout.connect(update_plots)
timer.start(figure_update_time)  # Update the plots based on figure_update_time
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Lect3_example1_BGT60LTR11AIP.py

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

Connect to sensor

```
if __name__ == "__main__":
    # ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    # connect to the device
    pp = pprint.PrettyPrinter()
    with DeviceLtr11() as device:
        print("Radar SDK Version: " + get_version())
        print("Sampling Frequency [Hz]: ", sample_rate)

        sampling_frequency = device.get_sampling_frequency(prt_index)
        config_defaults = device.get_config_defaults()

        config = Ltr11Config(
            aprt_factor=4,
            detector_threshold=80,
            disable_internal_detector=False,
            hold_time=8,
            mode=0,  # 0: continuous wave mode, -- 1: pulse mode
            num_of_samples=num_of_samples,
            prt=prt_index,
            pulse_width=3,
            rf_frequency_Hz=61044000000,
            rx_if_gain=8,
            tx_power_level=7,
        )
        device.set_config(config)
```

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board

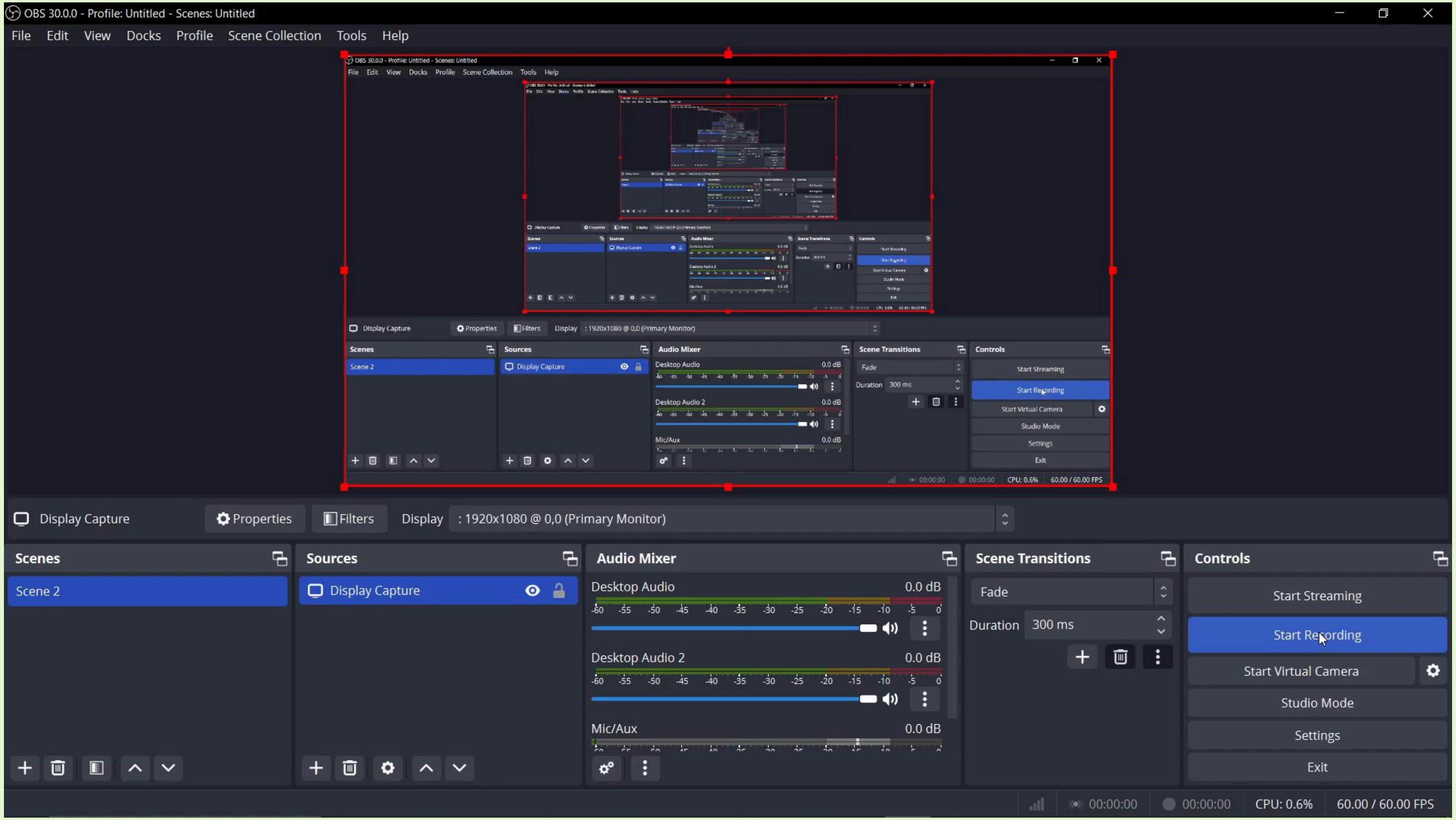
Start the data_thread and process_thread

```
# initialization
raw_data = np.zeros(raw_data_size, dtype=np.complex128)
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

# Threads for reading data and processing
data_thread = threading.Thread(target=read_data, args=(device,))
data_thread.start()

radar_processor = myProcessorClass()
process_thread = threading.Thread(target=radar_processor.process_data, args=())
process_thread.start()
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

sys.exit(app.exec_())
```

Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

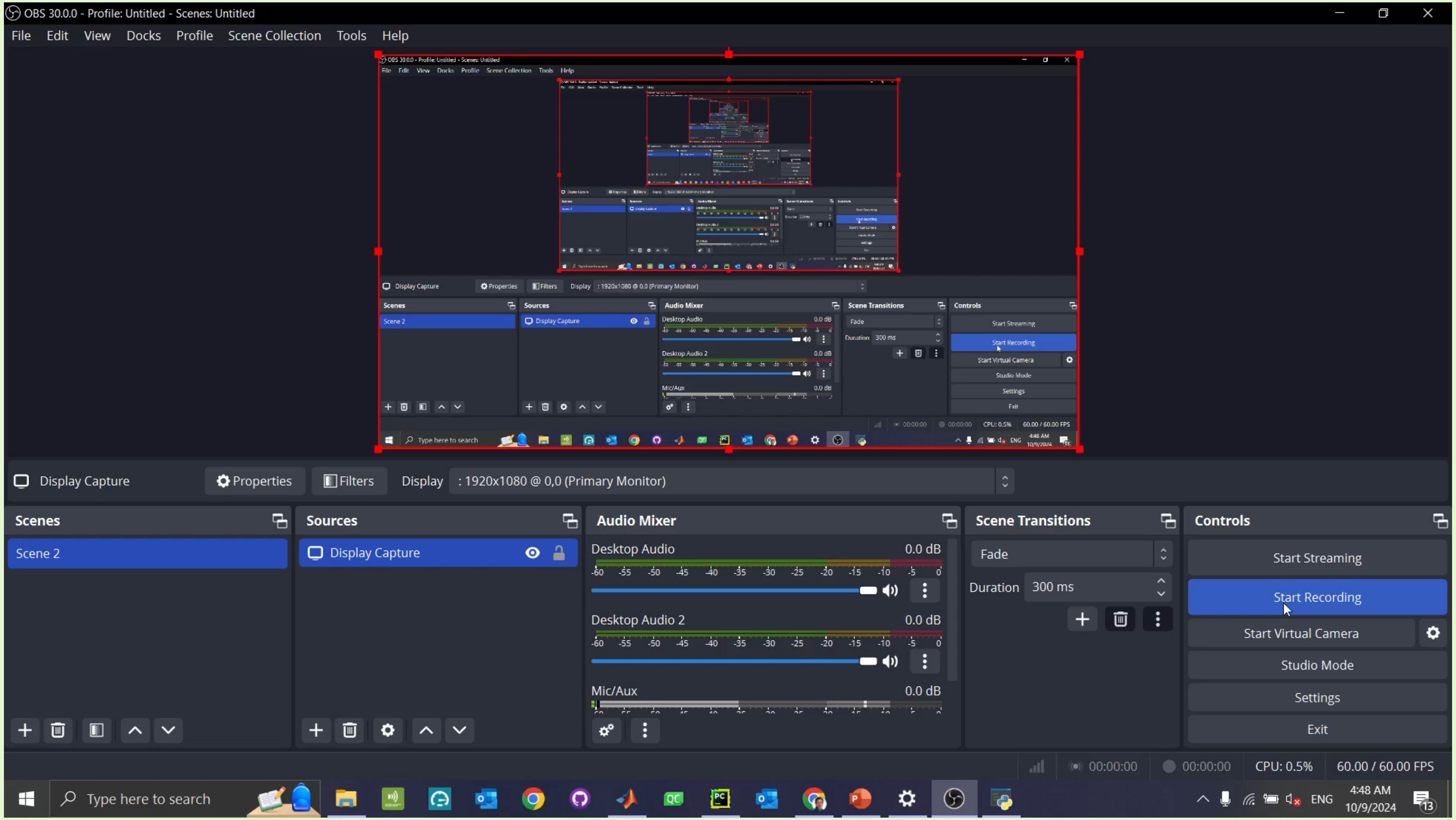
Getting started with the XENSIV™ 60GHz BGT60LTR11AIP radar sensor demo board



Lect3_example1_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Record Data Demo BGT60LTR11AIP



Lect3_example1_BGT60LTR11AIP.py

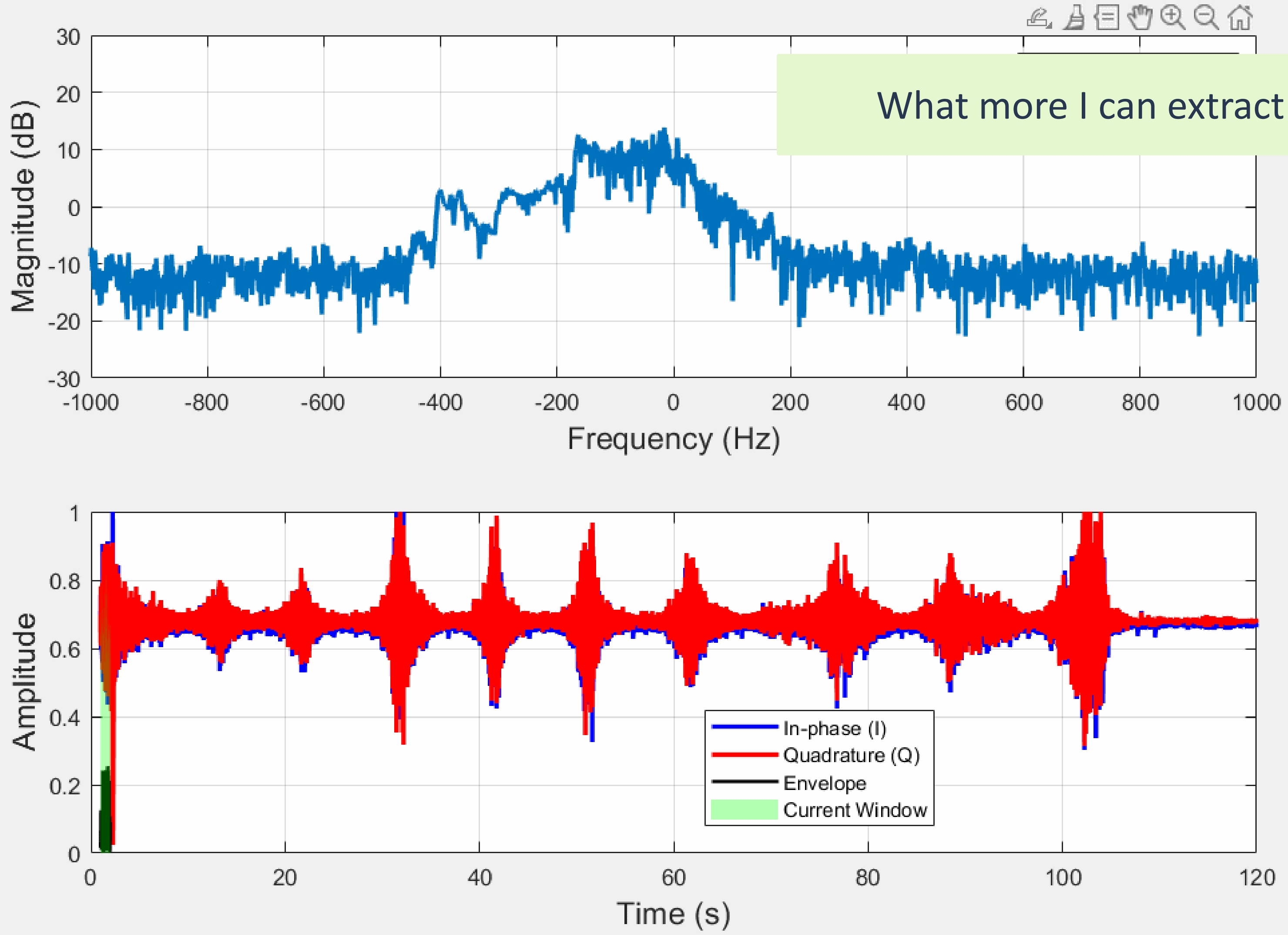
website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Recorded Data Demo BGT60LTR11AIP

Lect3_example2.m

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Recorded Data Demo BGT60LTR11AIP

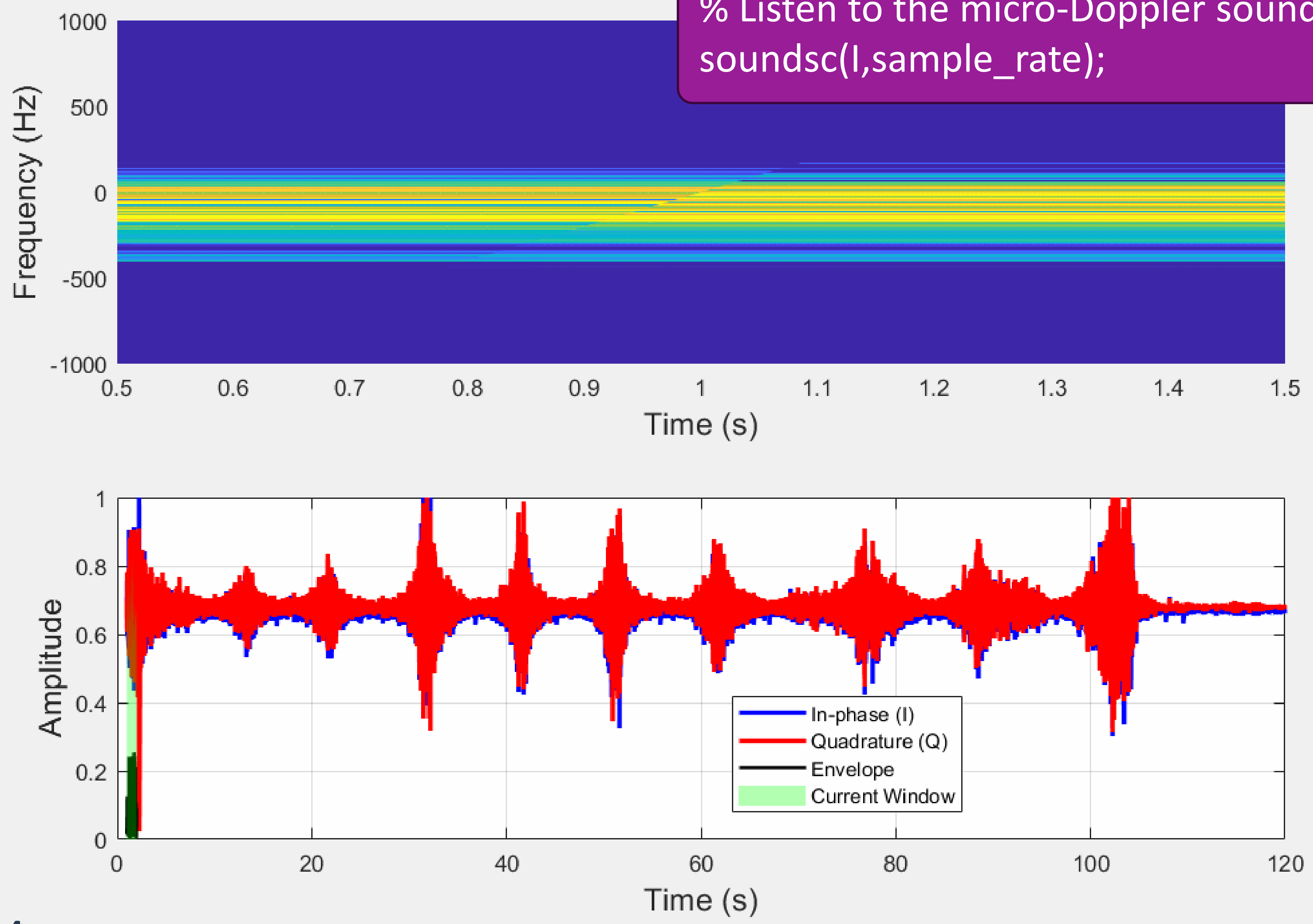


What more I can extract from this signal?

Lect3_example3.m

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Micro Doppler with BGT60LTR11AIP

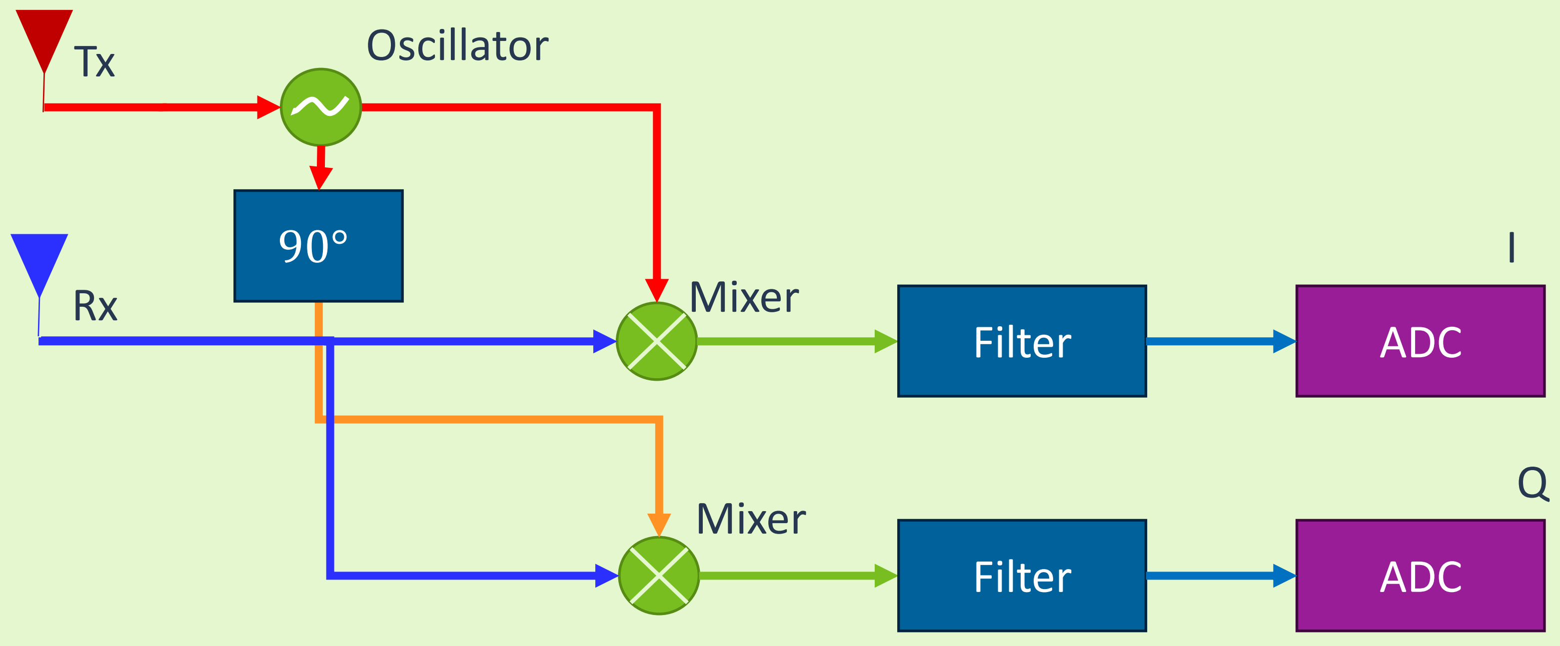



```
% Listen to the micro-Doppler sound
soundsc(I,sample_rate);
```

Lect3_example4.m

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

I and Q Modulator in CW Radar



$$\bar{s}_{rx}(t) = A_r\, e^{(j2\pi f_d t)}$$

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Micro-Doppler

$$\bar{s}_{rx}(t) = A_r \, \mathrm{e}^{(j2\pi f_d t)}$$

$$f_d = \frac{2v_r}{\lambda} = \frac{2}{\lambda}\left(v_{target} + \sum_{i=1}^{n} v_{component,i}\right)$$

v_{target} : Velocity of the main target

$v_{component,i}$: Velocity of the i-th moving component contributing to the micro-Doppler effect

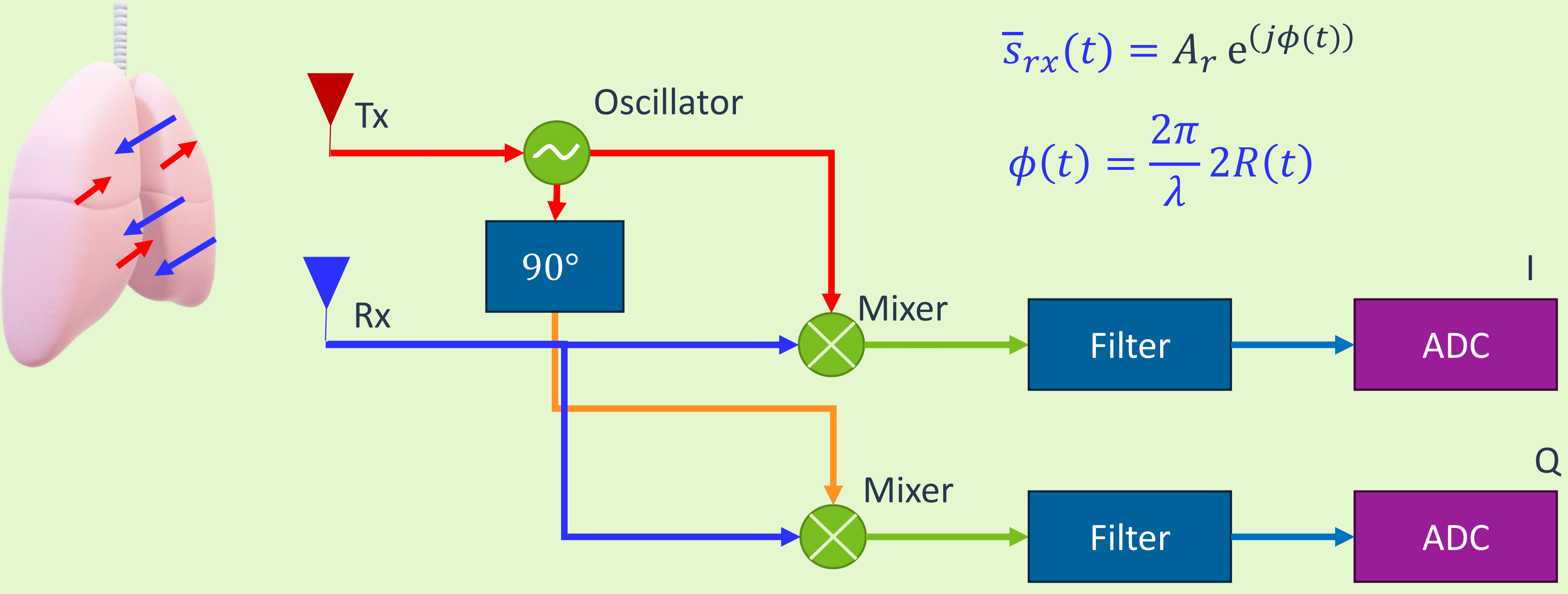
n : number of components contributing to the micro-Doppler signature

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Chest Movement Effect

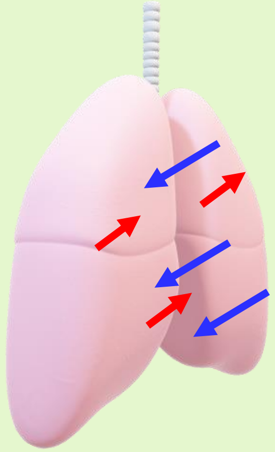
$$\bar{s}_{rx}(t) = A_r\, e^{(j\phi(t))}$$

$$\phi(t) = \frac{2\pi}{\lambda} 2R(t)$$



website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Chest Movement Effect



$$\phi(t) = \frac{2\pi}{\lambda} 2R(t) = \frac{4\pi}{\lambda}\left(R_0 + x(t)\right)$$

Distance to radar

Chest + heart + body movement

$$x(t) = x_h(t) + x_c(t) + x_b(t)$$

Heart

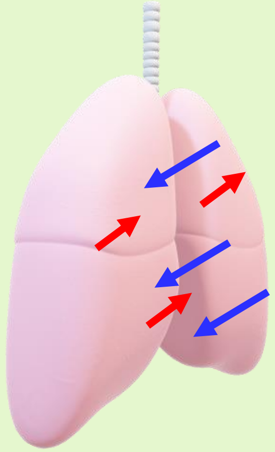
Chest

Body

	Displacement (mm)
Heartbeat	0.15 – 0.5
Chest	0.01 – 0.15

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Chest Movement Effect



$$\phi(t) = \frac{2\pi}{\lambda} 2R(t) = \frac{4\pi}{\lambda}\left(R_0 + x(t)\right)$$

Distance to radar

Chest + heart + body movement

$$x(t) = x_h(t) + x_c(t) + x_b(t)$$

Heart

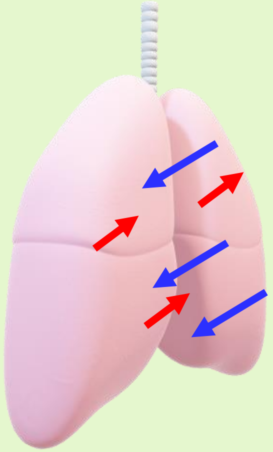
Chest

Body

	Displacement (mm)
Heartbeat	0.15 – 0.5
Chest	0.01 – 0.15

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Chest Movement Effect



$$\bar{s}_{rx}(t) = A_r\, \mathrm{e}^{(j\phi(t))} = I + jQ$$

$$\hat{\phi} = \arctan\frac{Q}{I}$$

$$f_c = 60 \text{ GHz}$$

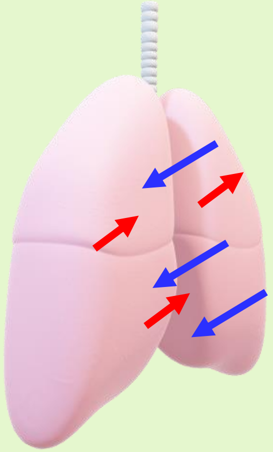
$$\lambda = 0.005 \text{ m}$$

$$\text{Displacement (mm)} = 0.5 \text{ mm} = 0.005 \text{ m}$$

$$\Delta\phi \geq \frac{4\pi}{0.005}\, 0.005 = 4\phi$$

Folding happens

Breathing Rate Estimation



I and Q signal

Phase angle estimation

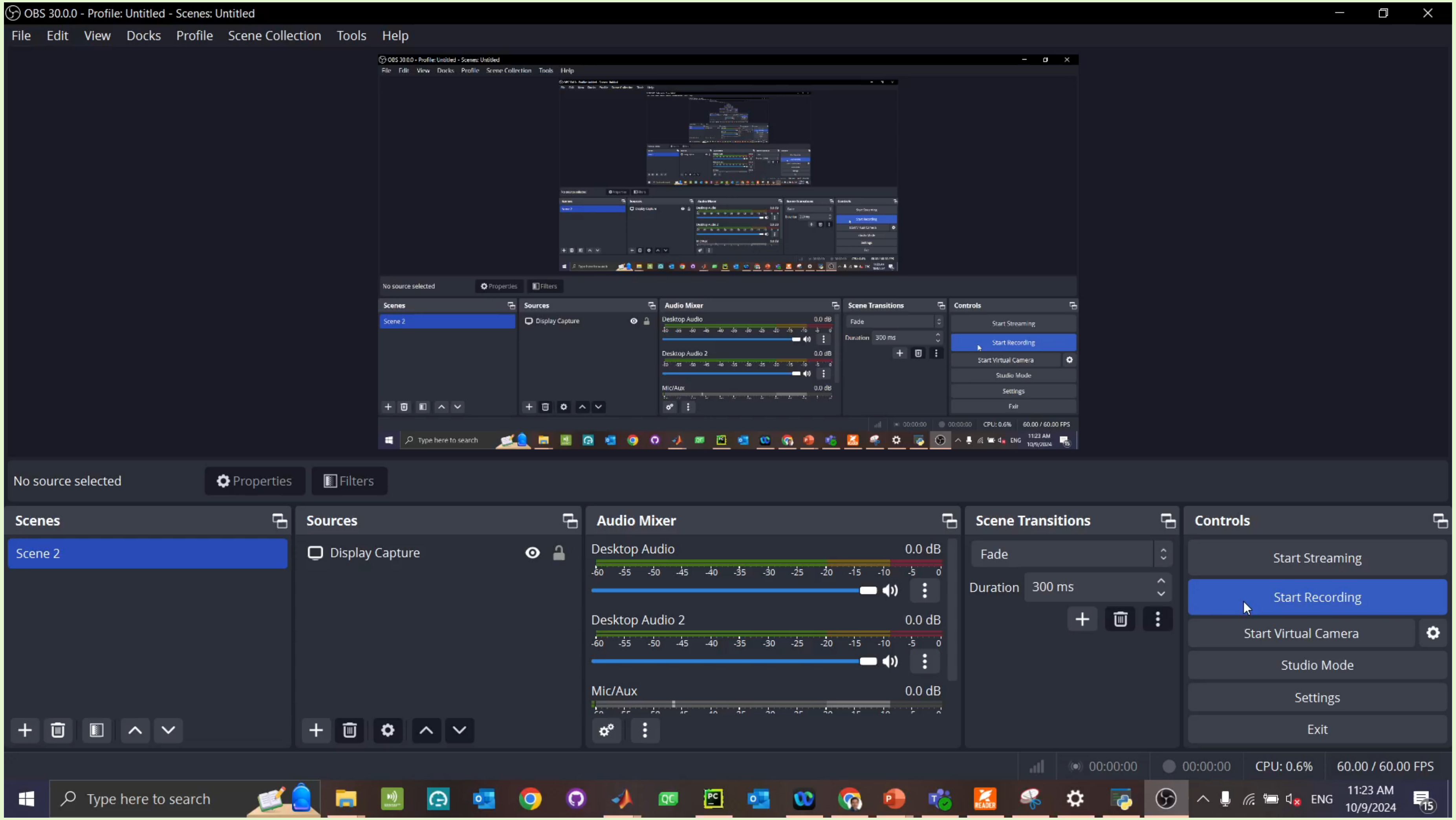
Phase unwrap

Appropriate Filtering

Breathing and Herat Rate Estimation (Spectrum Analysis)

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

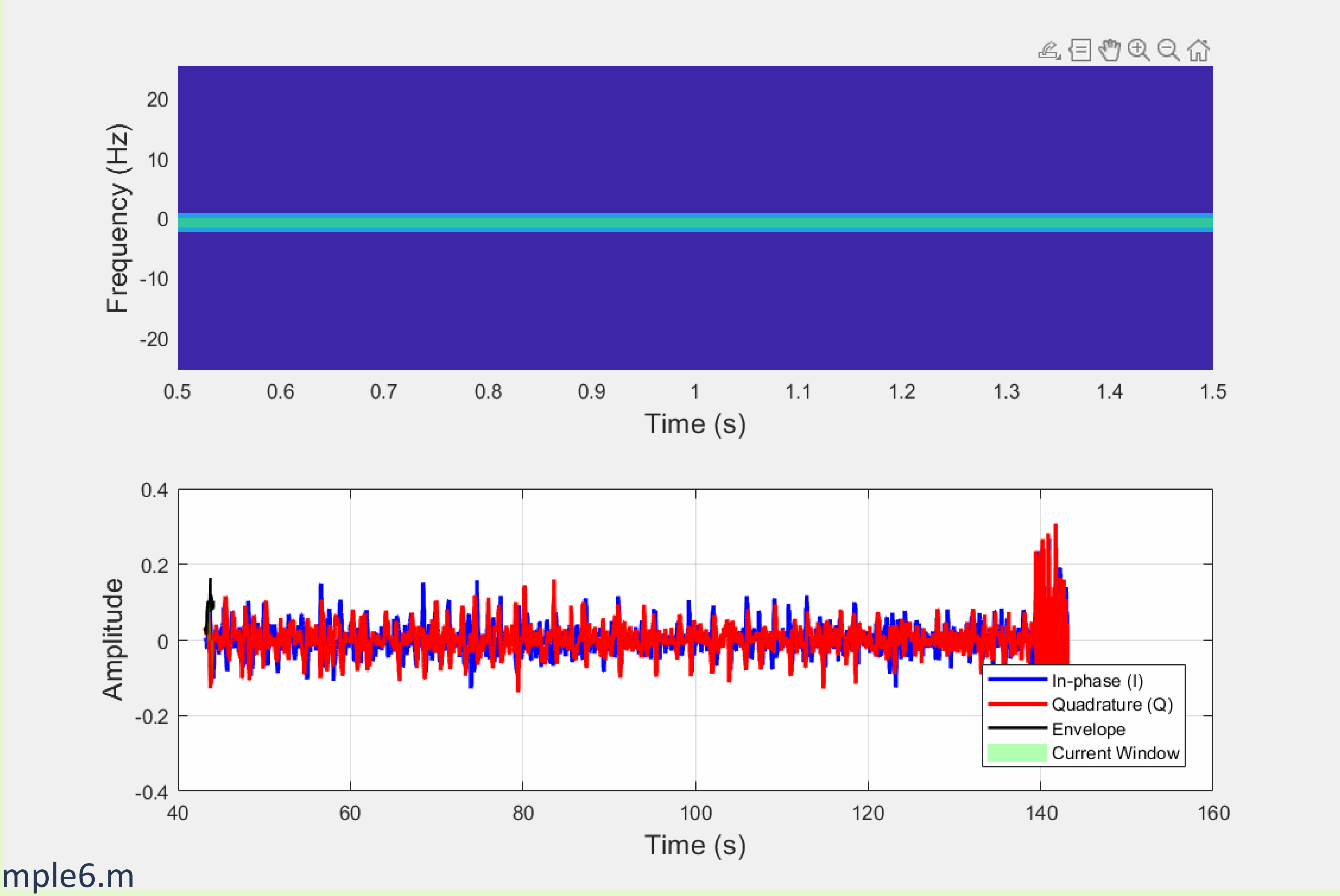
Breathing Rate Estimation



Lect3_example5_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

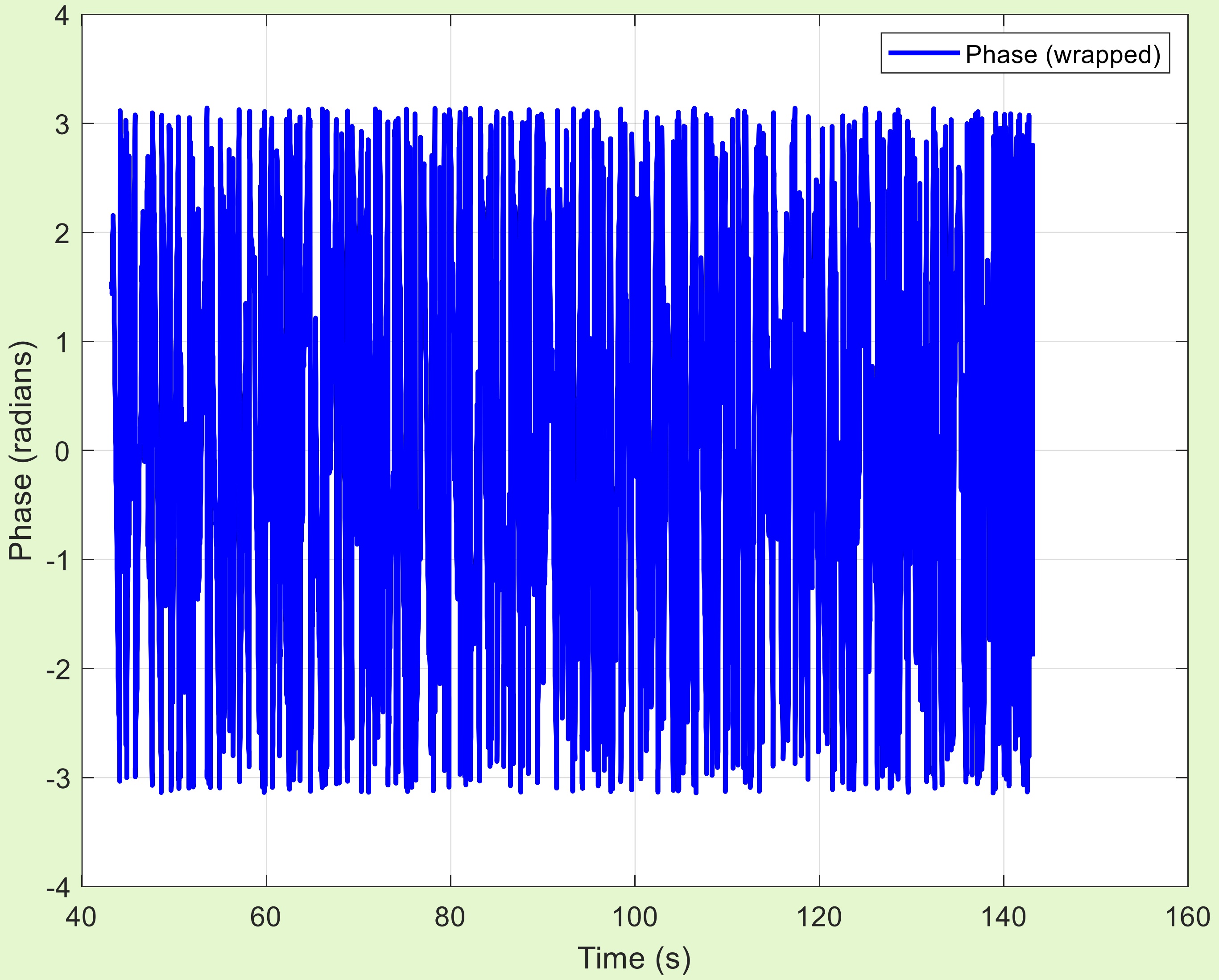
Breathing Rate Estimation



Lect3_example6.m

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Breathing Rate Estimation



Lect3_example7.m

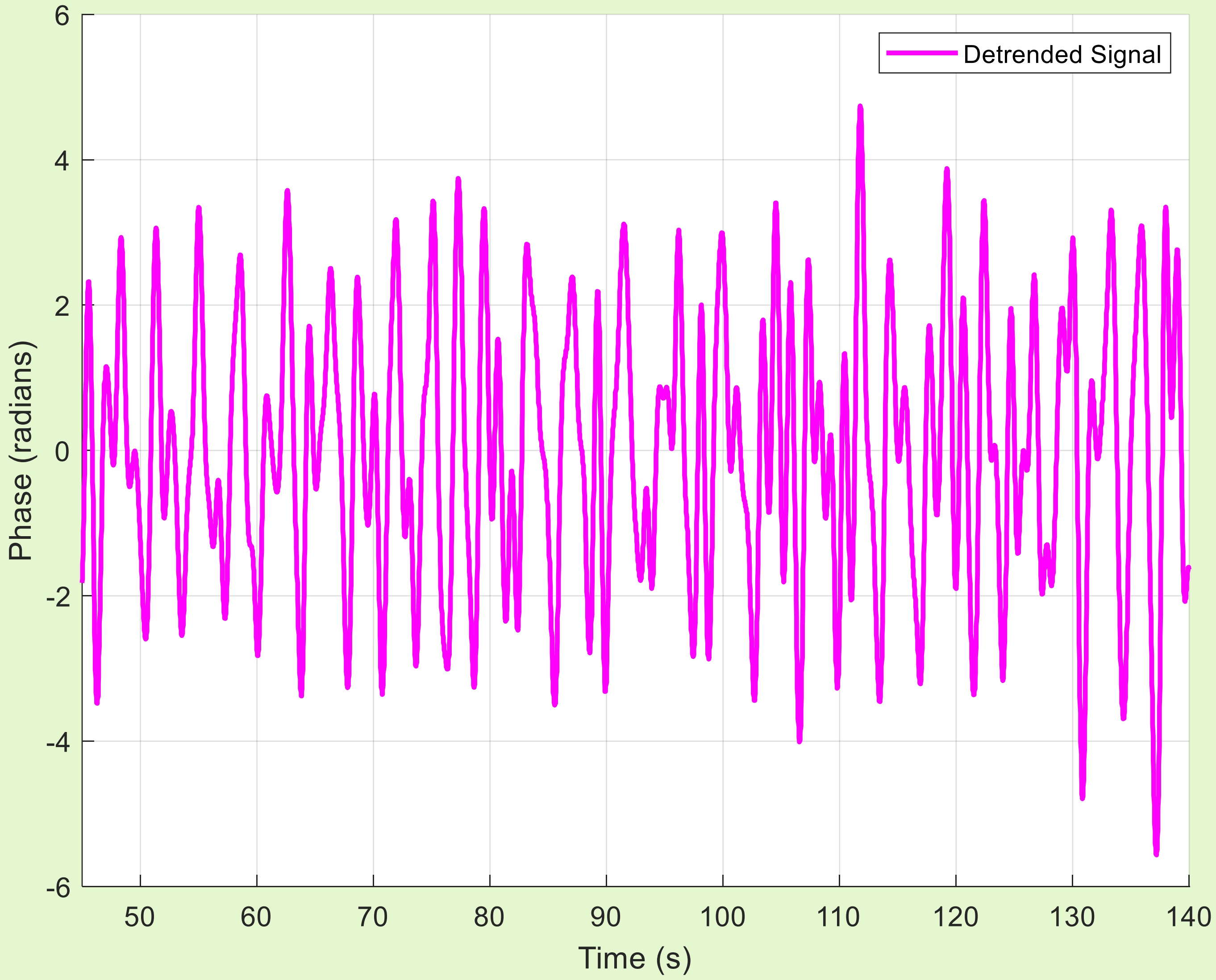
website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Breathing Rate Estimation

Lect3_example7.m

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

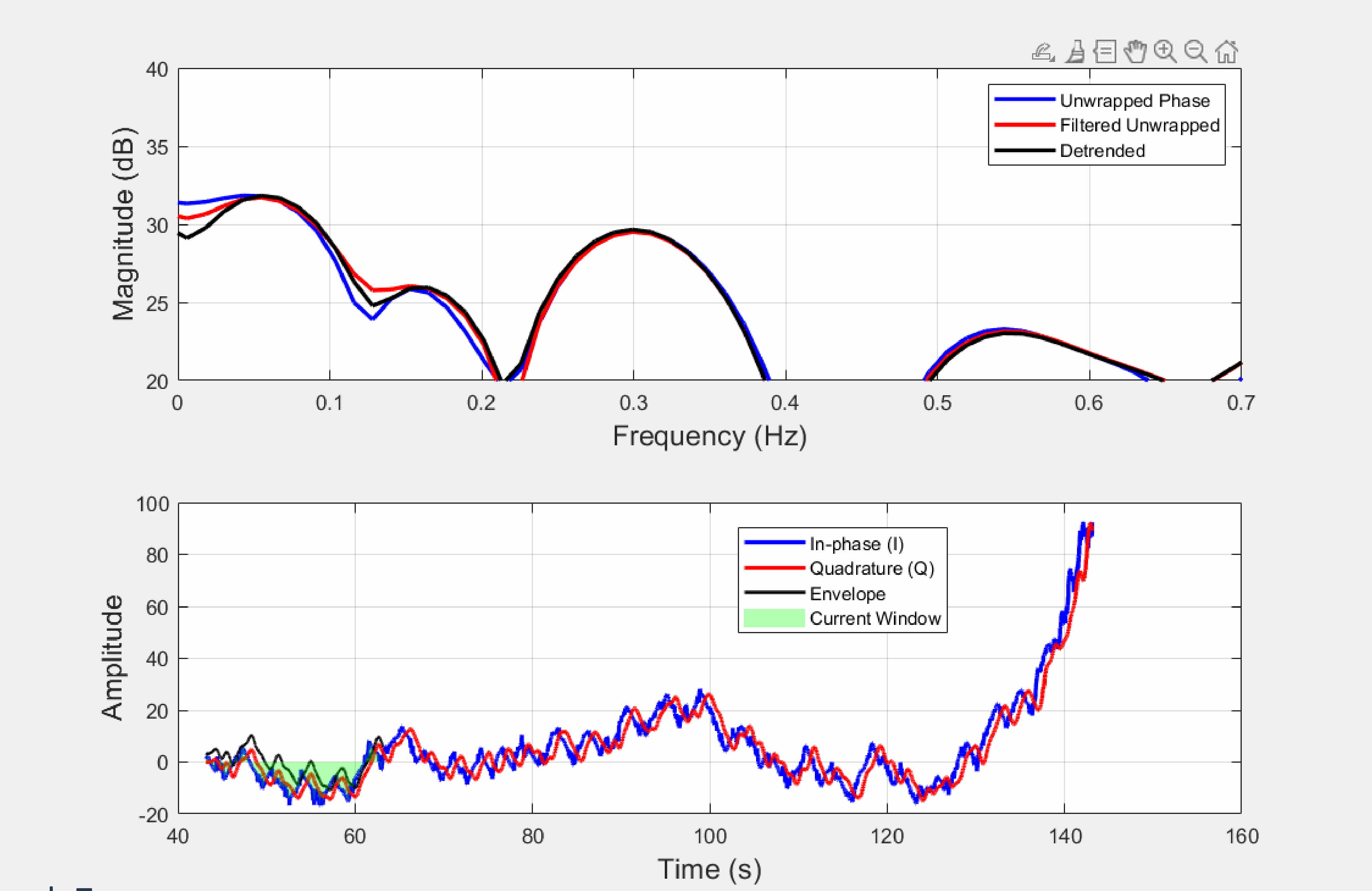
Breathing Rate Estimation



Lect3_example7.m

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

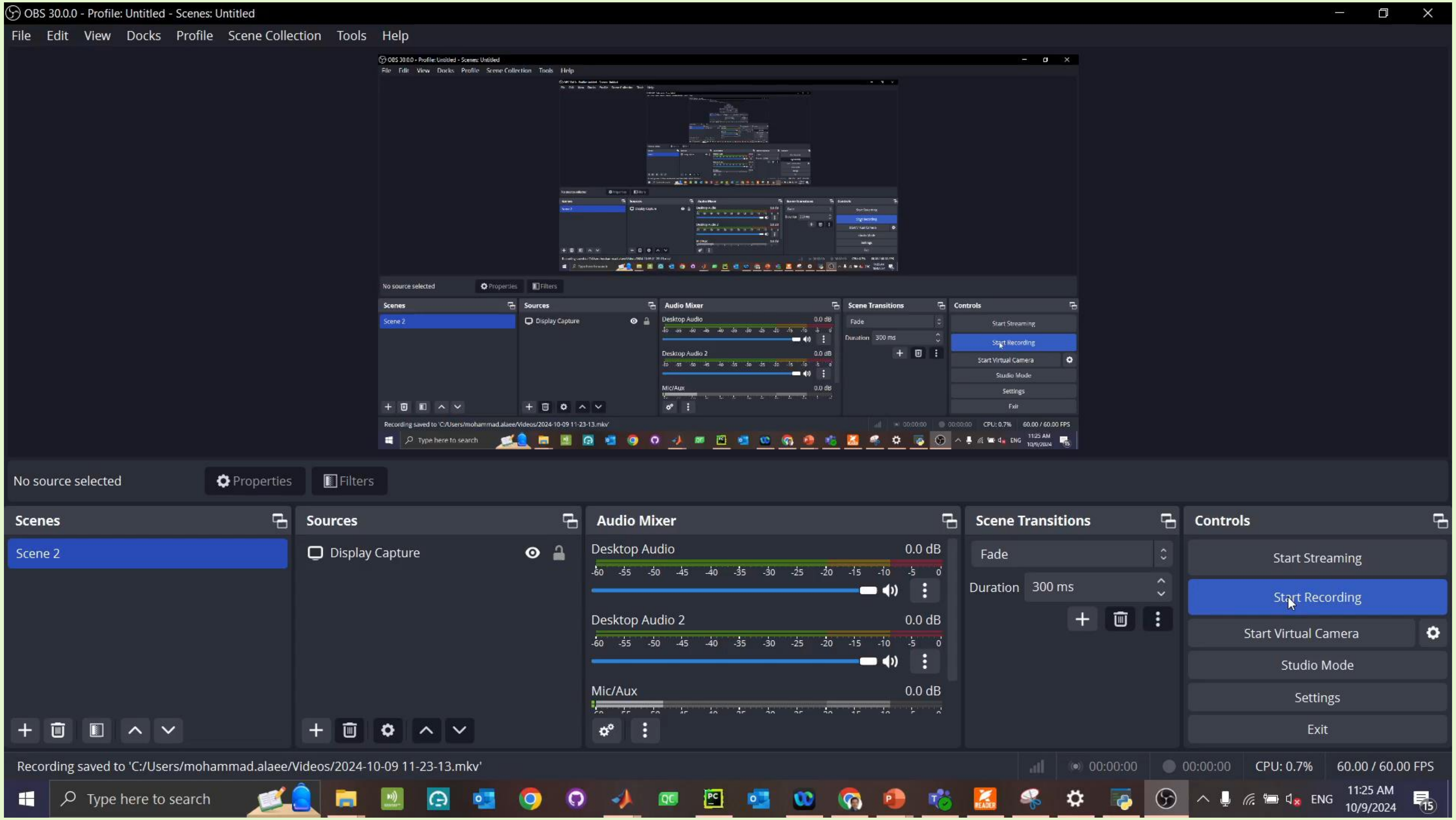
Breathing Rate Estimation



Lect3_example7.m

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

Breathing Rate Estimation



Lect3_example6_BGT60LTR11AIP.py

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu

What we learned from Lecture 3

- In Lecture 3 we used BGT60LTR11AIP to capture real data and processed it to extract Doppler, micro-Doppler and breathing pattern of human. Different signal processing techniques to this end has been applied in real-time operation.

Scan the QR code for access to the codes

Q & A

Using a FMCW radar, how can we better monitor vital signs of human?

website: https://radarmimo.com/ email: mohammad.alaee@uni.lu