# Radar System Laboratory

## A.A. 2022/2023

*Master Degree in Telecommunication Engineering*
*University of Pisa*

francesco.mancuso@phd.unipi.it

# Activities Overview

## FMCW Radar Lab Measurements

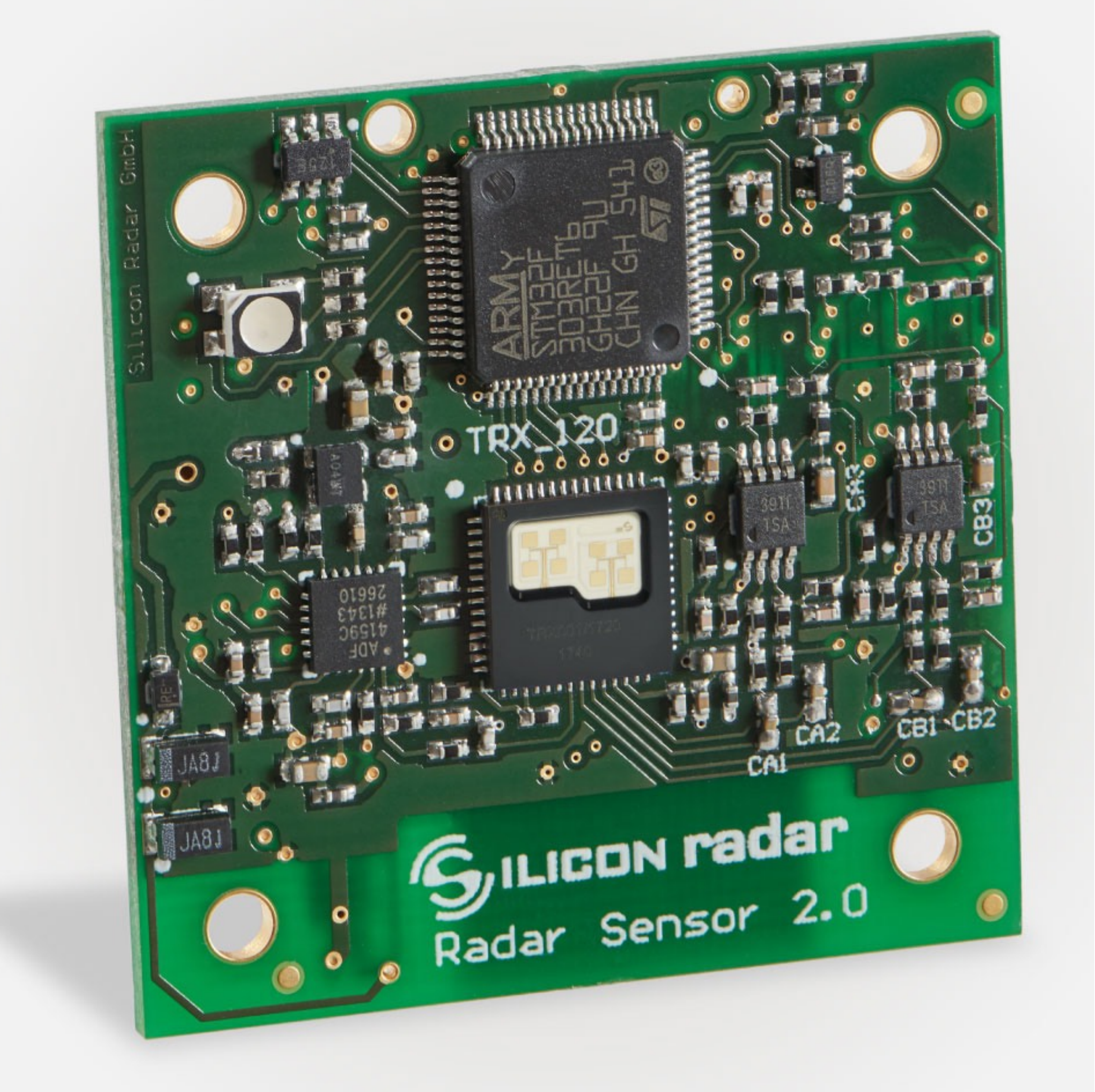
- EVALKIT SiRad Simple®
- Board configuration
- MATLAB acquisition
- A little bit of processing

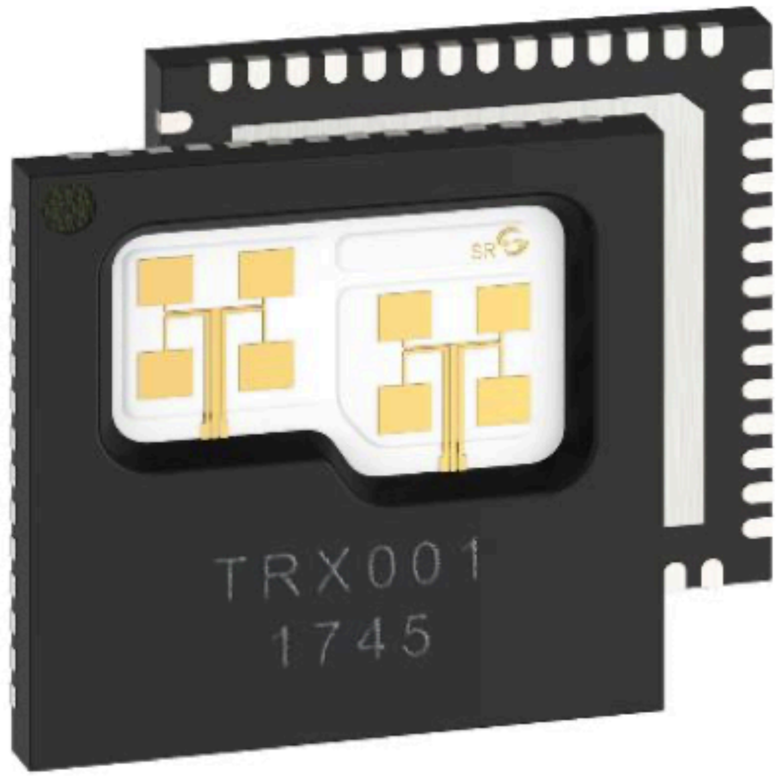
# EVALKIT SiRad Simple®

## 120GHz radar evaluation kit

…recently deprecated 😬

# TRX_120_001 (1)

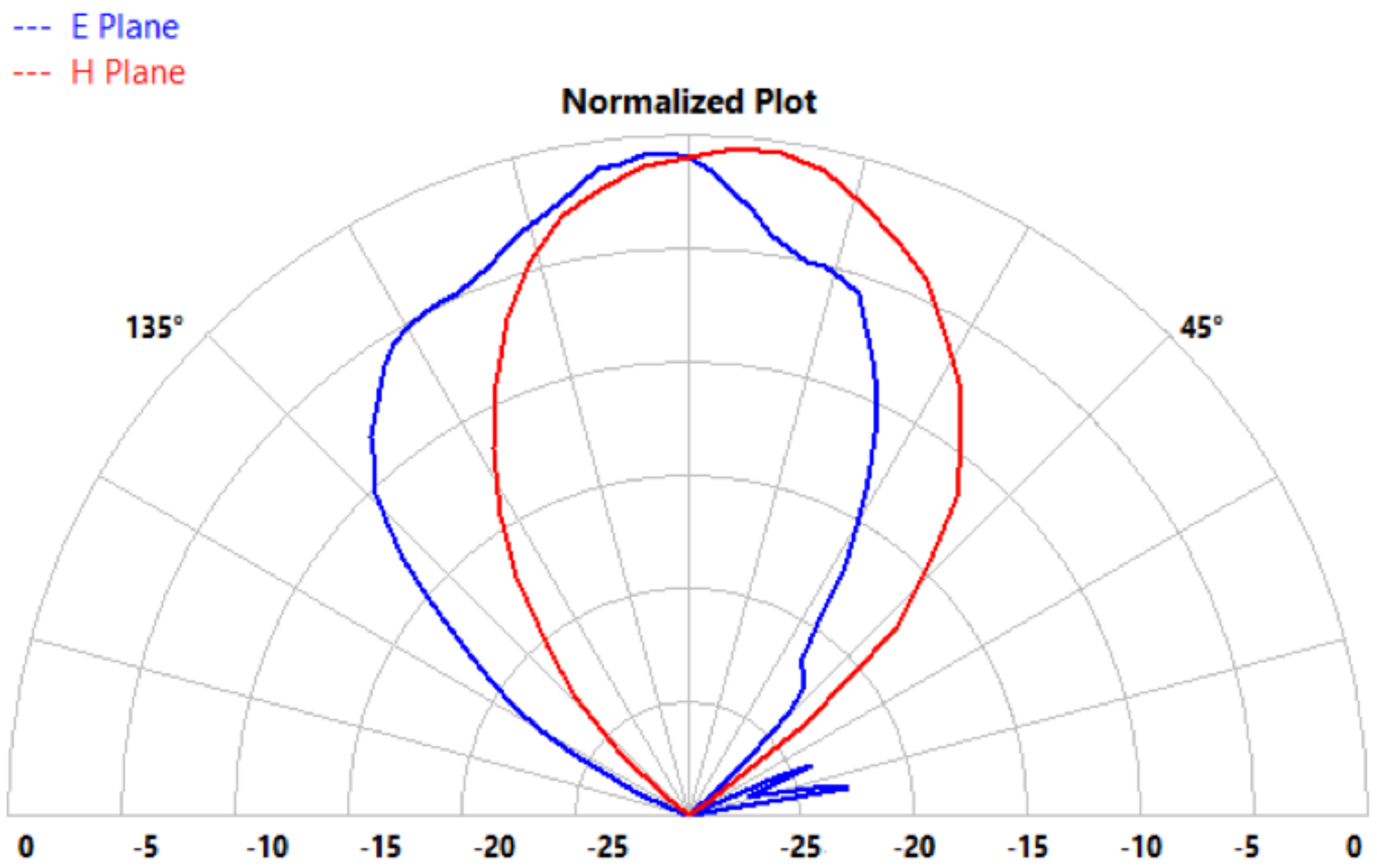


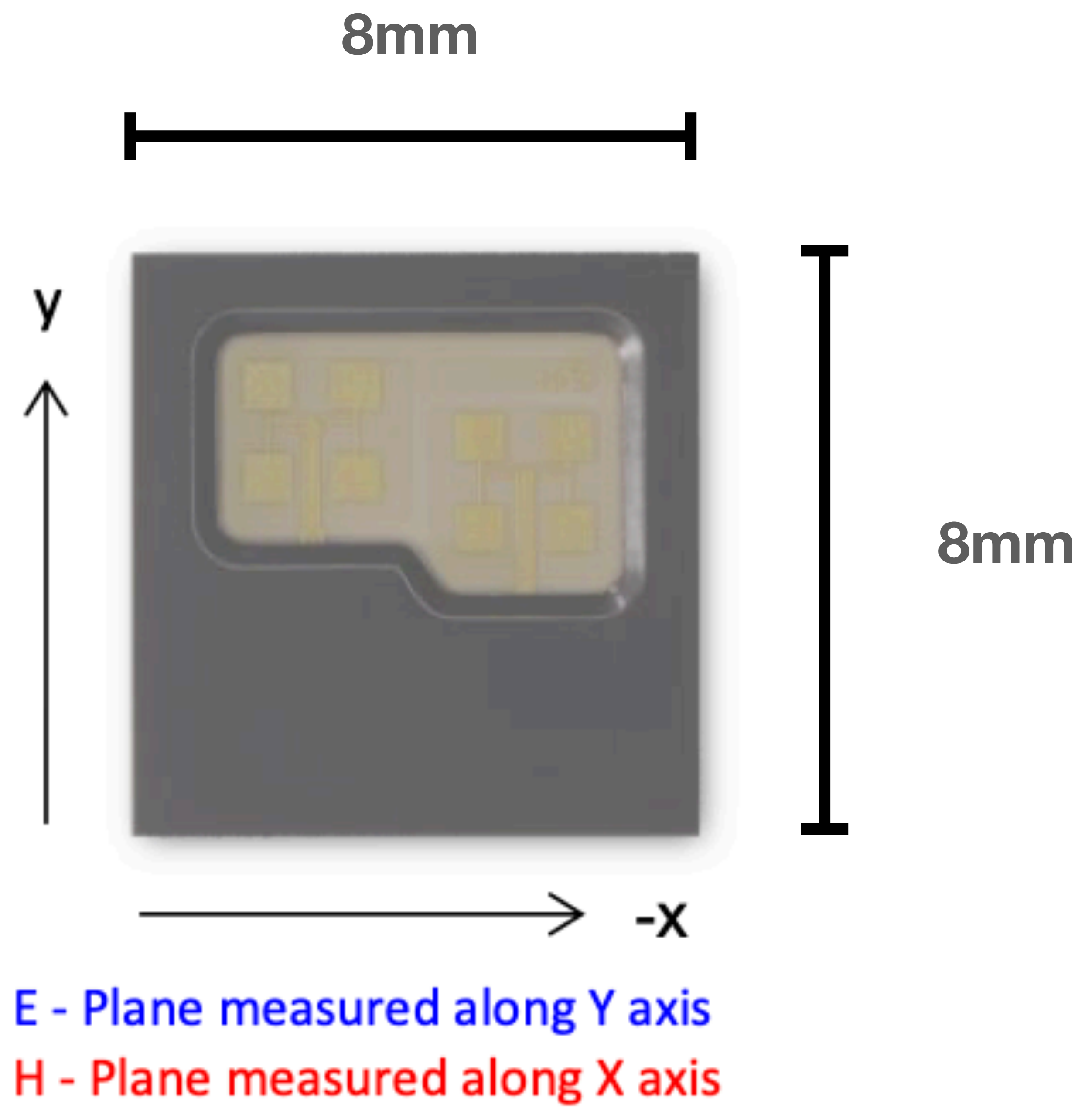




- Radar front end (RFE) with antennas in package for 122-GHz ISM band
- Wide bandwidth of up to 6 GHz
- Single supply voltage of 3.3 V

https://siliconradar.com/datasheets/Datasheet_TRX_120_001_V1.4.pdf

# TRX_120_001 (2)



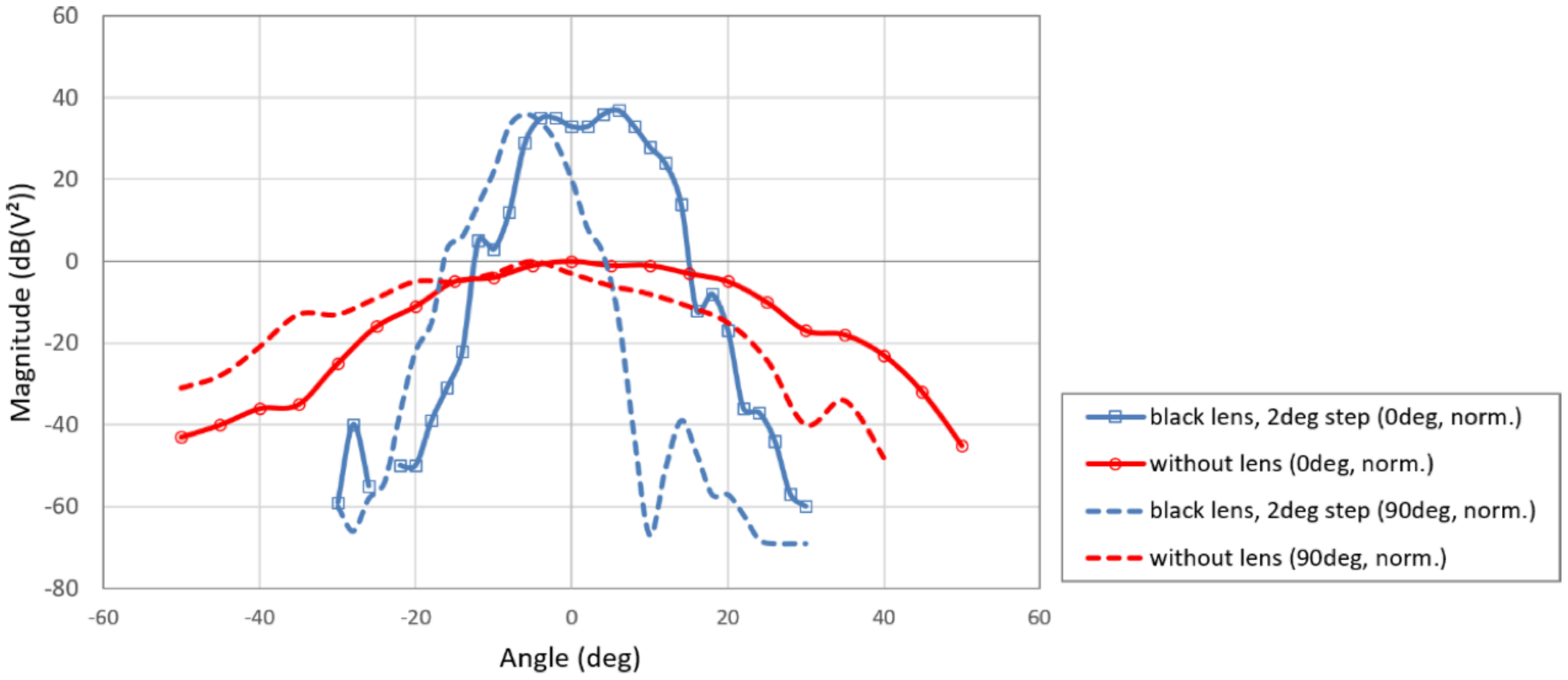



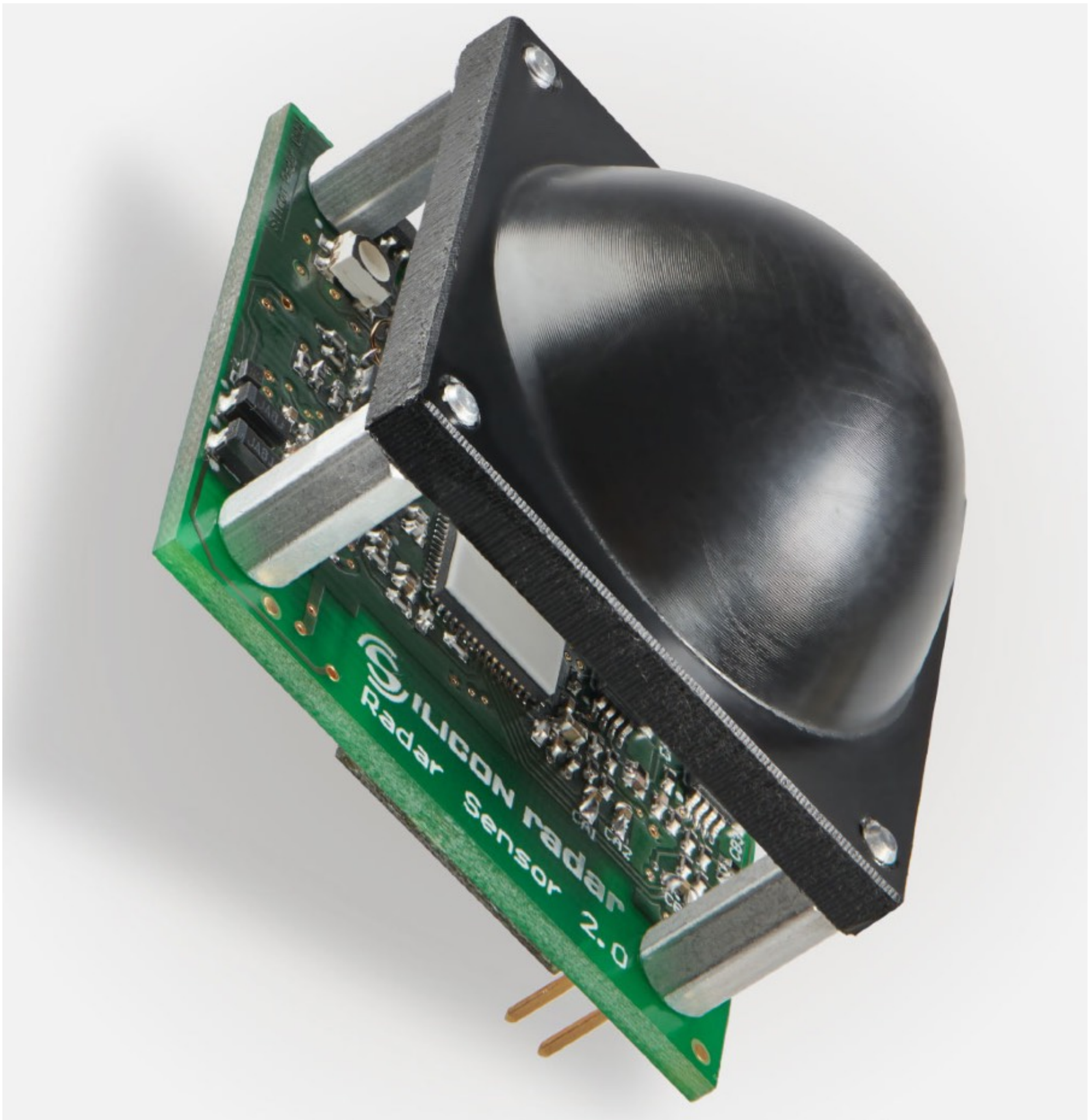

**Combined Radiation Pattern Measurements of TX and RX Patch Antennas**

https://siliconradar.com/datasheets/Datasheet_TRX_120_001_V1.4.pdf

# Collimator lens

- Optional accessory for evaluation kit ***SiRad Simple®***, it focuses the beam of the radar front end



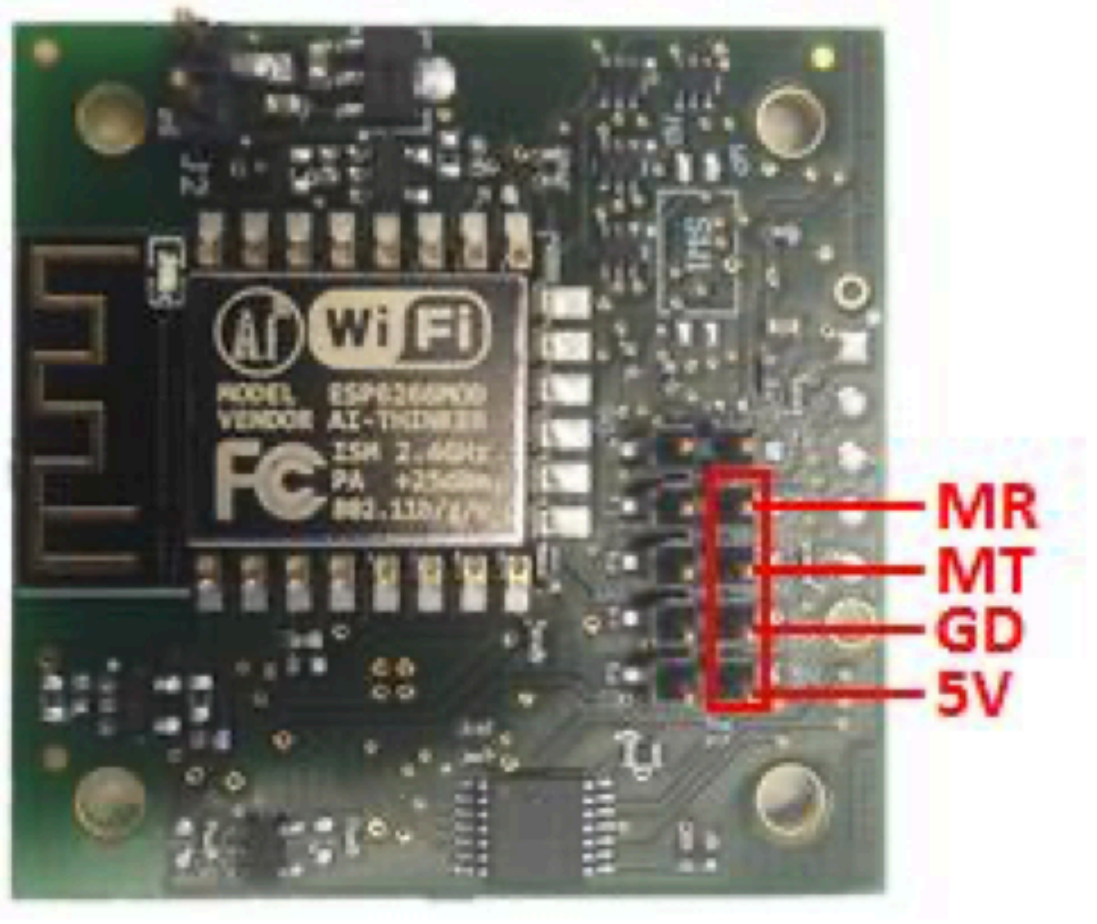

**±4° @ -6dB**

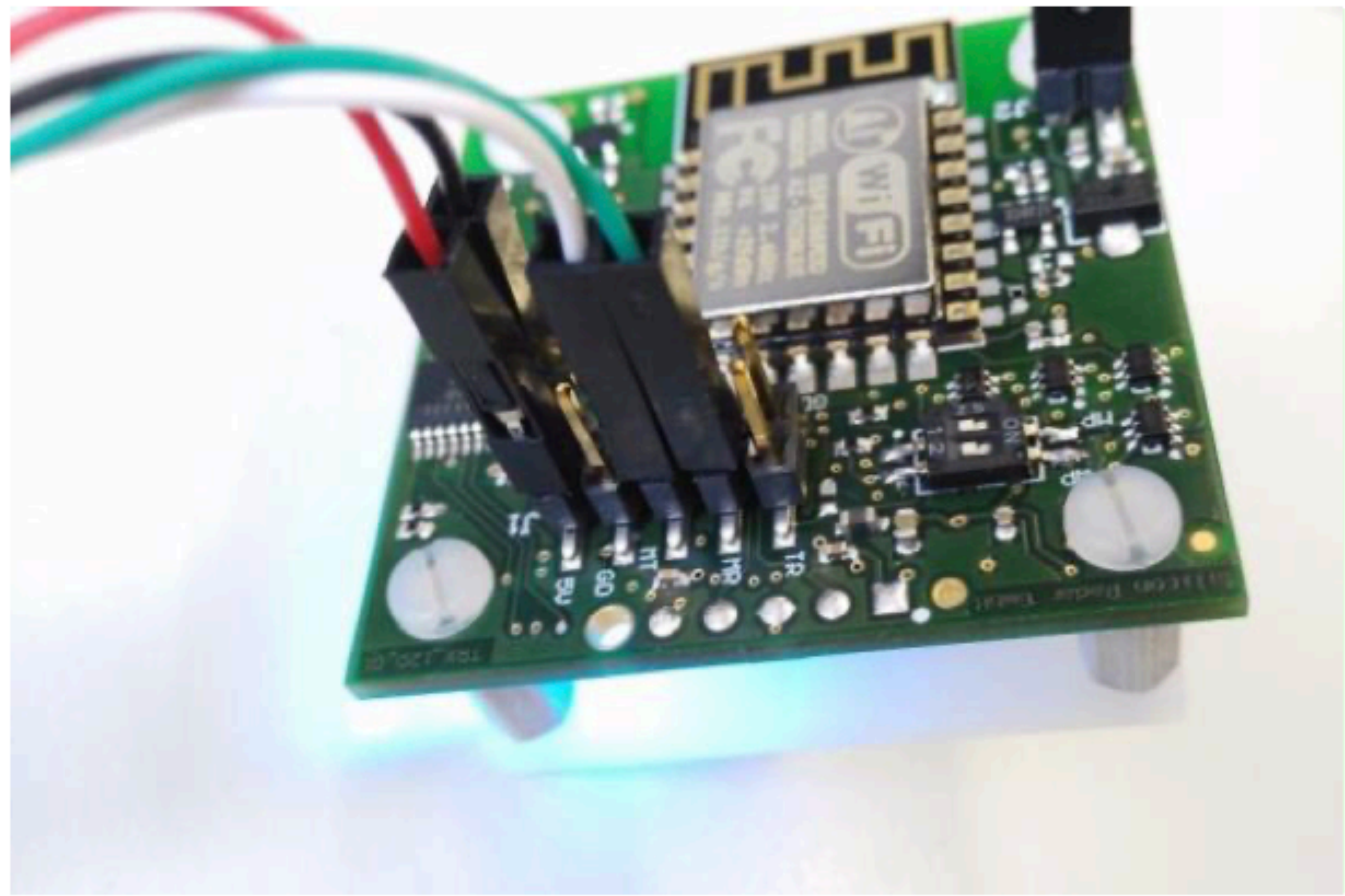
https://siliconradar.com/datasheets/Datasheet_Collimator_Lens_V1.1.pdf

# Board configuration

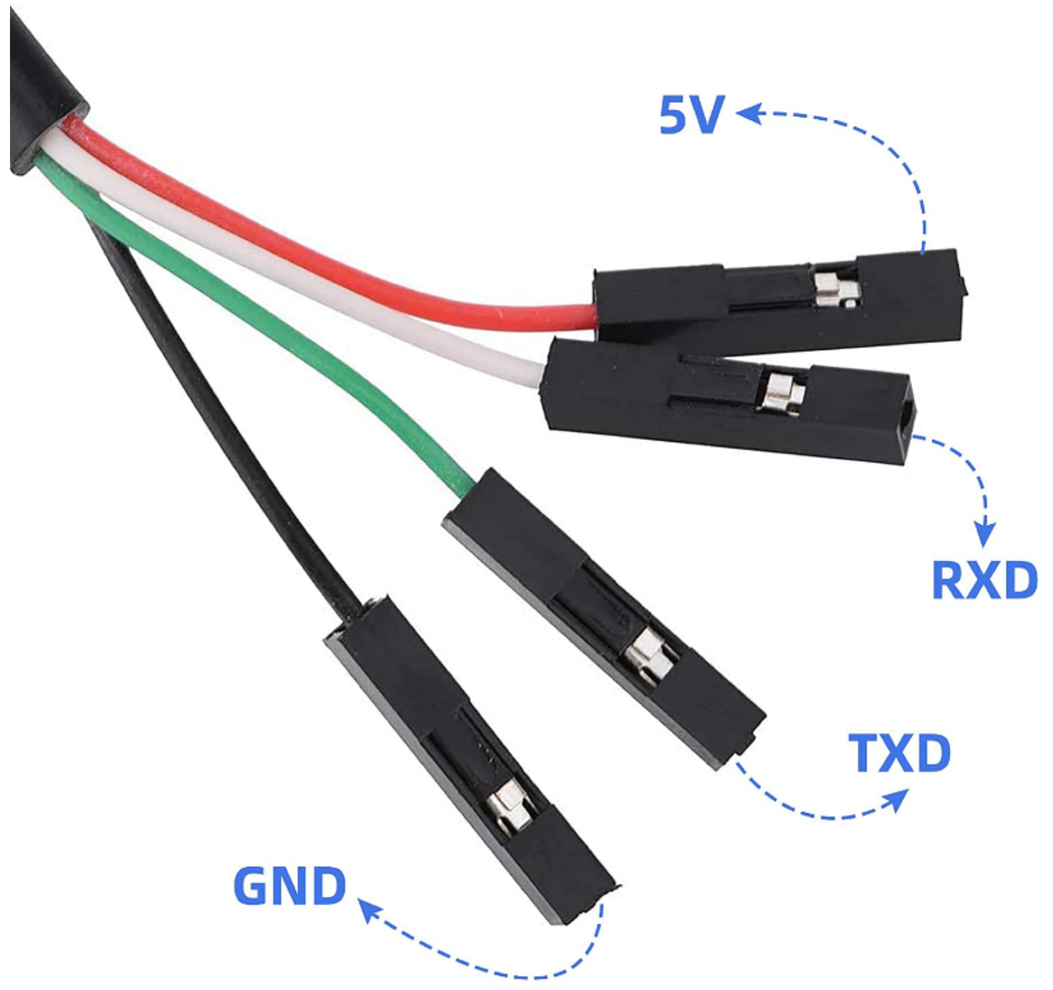
## USB-to-UART Bridge

# Serial/USB data connection



| FTDI cable | Board |
|---|---|
| 5V → | 5V |
| GND → | GD |
| TXD → | MR |
| RXD → | MT |




Make sure that both DIP switches are in their OFF positions and the power jumper (J2) for the WiFi module is open (switched off).

# System configuration

| Bit | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SYS_CONFIG | SelfTrigDelay | | | res | LOG | FMT | LED | | reserved | | | | Protocol | | | AGC | Gain | | SER2 | SER1 | ERR | ST | TL | C | R | P | CPL | RAW | res | res | SLF | PRE |

| SelfTrigDelay | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 ms |
| 0 | 0 | 1 | 2 ms |
| 0 | 1 | 0 | 4 ms |
| 0 | 1 | 1 | 8 ms |
| 1 | 0 | 0 | 16 ms |
| 1 | 0 | 1 | 32 ms |
| 1 | 1 | 0 | 64 ms |
| 1 | 1 | 1 | 128 ms |

| LED | | |
|---|---|---|
| 0 | 0 | off |
| 0 | 1 | 1st target |
| 1 | 0 | reserved |
| 1 | 1 | reserved |

| Gain | | |
|---|---|---|
| 0 | 0 | 8 dB |
| 0 | 1 | 21 dB |
| 1 | 0 | 43 dB |
| 1 | 1 | 56 dB |

| SER2 | |
|---|---|
| 0 | output on SER2 off |
| 1 | output on SER2 on |

| SER1 | |
|---|---|
| 0 | output on SER1 off |
| 1 | output on SER1 on |

| FMT | |
|---|---|
| 0 | TL mm distance |
| 1 | TL cm distance |

| LOG | |
|---|---|
| 0 | log MAG |
| 1 | linear MAG |

| data frames | | off | on |
|---|---|---|---|
| RAW | raw ADC | 0 | 1 |
| CPL | cmplx FFT | 0 | 1 |
| P | phase | 0 | 1 |
| R | magnitude | 0 | 1 |
| C | CFAR | 0 | 1 |
| TL | target list | 0 | 1 |
| ST | status | 0 | 1 |
| ERR | error | 0 | 1 |

| Protocol | | | |
|---|---|---|---|
| 0 | 0 | 0 | WebGUI |
| 0 | 0 | 1 | TSV output |
| 0 | 1 | 0 | BIN output |
| x | x | x | reserved |

| SLF | |
|---|---|
| 0 | external trig mode |
| 1 | standard mode |

| PRE | |
|---|---|
| 0 | standard mode |
| 1 | use pre-trigger |

| AGC | |
|---|---|
| 0 | auto gain control off |
| 1 | auto gain control on |

*Reserved bit will have 0 as default value

000 X 1 0 00 XXXX 001 0 00 0 1 00000001 X X 1 0

!S 0000 1000 0000 0010 0001 0000 0001 0010 \r\n → '!S08021012\r\n'

https://siliconradar.com/datasheets/Protocol_Description_Easy_Simple_V2.4.pdf

# PLL configuration

| | Bit | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PLL_CONFIG | | reserved | | | | | | | | | | | | | | | | Bandwidth [MHz] (16 Bits) | | | | | | | | | | | | | | | |

| Bandwidth [MHz] (16 Bits) | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -2 | MHz |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | -4 | MHz |
| ... | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -65536 | MHz |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | MHz |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 2 | MHz |
| ... | | | | | | | | | | | | | | | | ... | |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | +65534 | MHz |

| Format Field | Field Size | Description |
|---|---|---|
| Bandwidth | 16 bits | Negative values result in a falling ramp slope, positive values in a rising saw tooth shape; representation is in two's complement |

*Reserved bit will have 0 as default value

**XXXXXXXXXXXXXXXX 0000000111110100**

**BW = 1 GHz**

**!P 0000 0000 0000 0000 0000 0001 1111 0100 \r\n** ⟶ **'!P000001F4\r\n'**

https://siliconradar.com/datasheets/Protocol_Description_Easy_Simple_V2.4.pdf

# Baseband configuration

| Bit | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BB_CONFIG | WIN | FIR | DC | CFAR | | CFAR Threshold [dB] | | | | CFAR Size | | | | CFAR Grd | | Average n | | FFT Size | | | Down sample | | | # Ramps | | | # Samples | | | ADC ClkDiv | | |

| WIN | |
|---|---|
| 0 | windowing off |
| 1 | windowing on |

| FIR | |
|---|---|
| 0 | FIR filter off |
| 1 | FIR filter on |

| DC | |
|---|---|
| 0 | DC cancellation off |
| 1 | DC cancellation on |

| FFT Size | | | |
|---|---|---|---|
| 0 | 0 | 0 | 32 |
| 0 | 0 | 1 | 64 |
| 0 | 1 | 0 | 128 |
| 0 | 1 | 1 | 256 |
| 1 | 0 | 0 | 512 |
| 1 | 0 | 1 | 1024 |
| 1 | 1 | 0 | 2048 |
| 1 | 1 | 1 | reserved |

| Down sample | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 2 |
| 0 | 1 | 1 | 4 |
| 1 | 0 | 0 | 8 |
| 1 | 0 | 1 | 16 |
| 1 | 1 | 0 | 32 |
| 1 | 1 | 1 | 64 |

| # Ramps | | | |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 2 |
| 0 | 1 | 0 | 4 |
| 0 | 1 | 1 | 8 |
| 1 | 0 | 0 | 16 |
| 1 | 0 | 1 | 32 |
| 1 | 1 | 0 | 64 |
| 1 | 1 | 1 | 128 |

| # Samples | | | |
|---|---|---|---|
| 0 | 0 | 0 | 32 |
| 0 | 0 | 1 | 64 |
| 0 | 1 | 0 | 128 |
| 0 | 1 | 1 | 256 |
| 1 | 0 | 0 | 512 |
| 1 | 0 | 1 | 1024 |
| 1 | 1 | 0 | 2048 |
| 1 | 1 | 1 | reserved |

| ADC ClkDiv | | | MS/s |
|---|---|---|---|
| 0 | 0 | 0 | 2,571 |
| 0 | 0 | 1 | 2,400 |
| 0 | 1 | 0 | 2,118 |
| 0 | 1 | 1 | 1,800 |
| 1 | 0 | 0 | 1,125 |
| 1 | 0 | 1 | 0,487 |
| 1 | 1 | 0 | 0,186 |
| 1 | 1 | 1 | 0,059 |

| CFAR | | |
|---|---|---|
| 0 | 0 | CA-CFAR |
| 0 | 1 | CFAR_GO |
| 1 | 0 | CFAR_SO |
| 1 | 1 | reserved |

| CFAR Threshold [dB] | | | | dB |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 2 |
| | | | | ... |
| 1 | 1 | 1 | 1 | 30 |

| CFAR Size | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| ... | | | | |
| 1 | 1 | 1 | 1 | 15 |

| CFAR Grd | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 2 |
| 1 | 1 | 3 |

| Average n | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 2 |
| 1 | 1 | 3 |

**0 0 1 00 0000 0000 00 00 000 000 000 000 000**

**!B 0010 0000 0000 0000 0000 0000 0000 0000 \r\n** → **'!B20000000\r\n'**

https://siliconradar.com/datasheets/Protocol_Description_Easy_Simple_V2.4.pdf

# MATLAB acquisition

```
s = serialport("COM3", 1000000);
```

# Serial port configuration

```
sPort = serialportlist;
baudRate = 1e6;

serialPort = sPort(2);
fprintf("Connecting to %s …\n", serialPort)
com_port = serialport(serialPort, baudRate);

set(com_port, "DataBits", 8);
set(com_port, "Parity", 'none');
set(com_port, "StopBits", 1);
set(com_port, "Timeout", 1);
configureTerminator(com_port,'CR/LF');
```

`serialportlist()` lists all the serial devices

On my PC, it was the second element; this must be adapted

Configuration of the serial port

**Note for Linux users:** I had some problems with Ubuntu, the OS wanted all the permissions to the serial ports. Type in Terminal: `sudo chmod 666 /dev/tty*`

https://www.mathworks.com/help/matlab/ref/serialport.html

# Data acquisition

## External trigger

```
while true
    writeline(com_port, "!N");
    buf = '';
    while isempty(buf) || ~startsWith(buf,'!M')
        buf = readline(com_port);
    end

    frameM = split(buf);
    frameM = frameM(4:end);

    dataM = str2double(frameM);
    dataI = dataM(1:2:end-1);
    dataQ = dataM(2:2:end);

    complexData = (dataI + 1i*dataQ).';

    /--- GENERIC PROCESSING ---/
end
```

Trigger acquisition

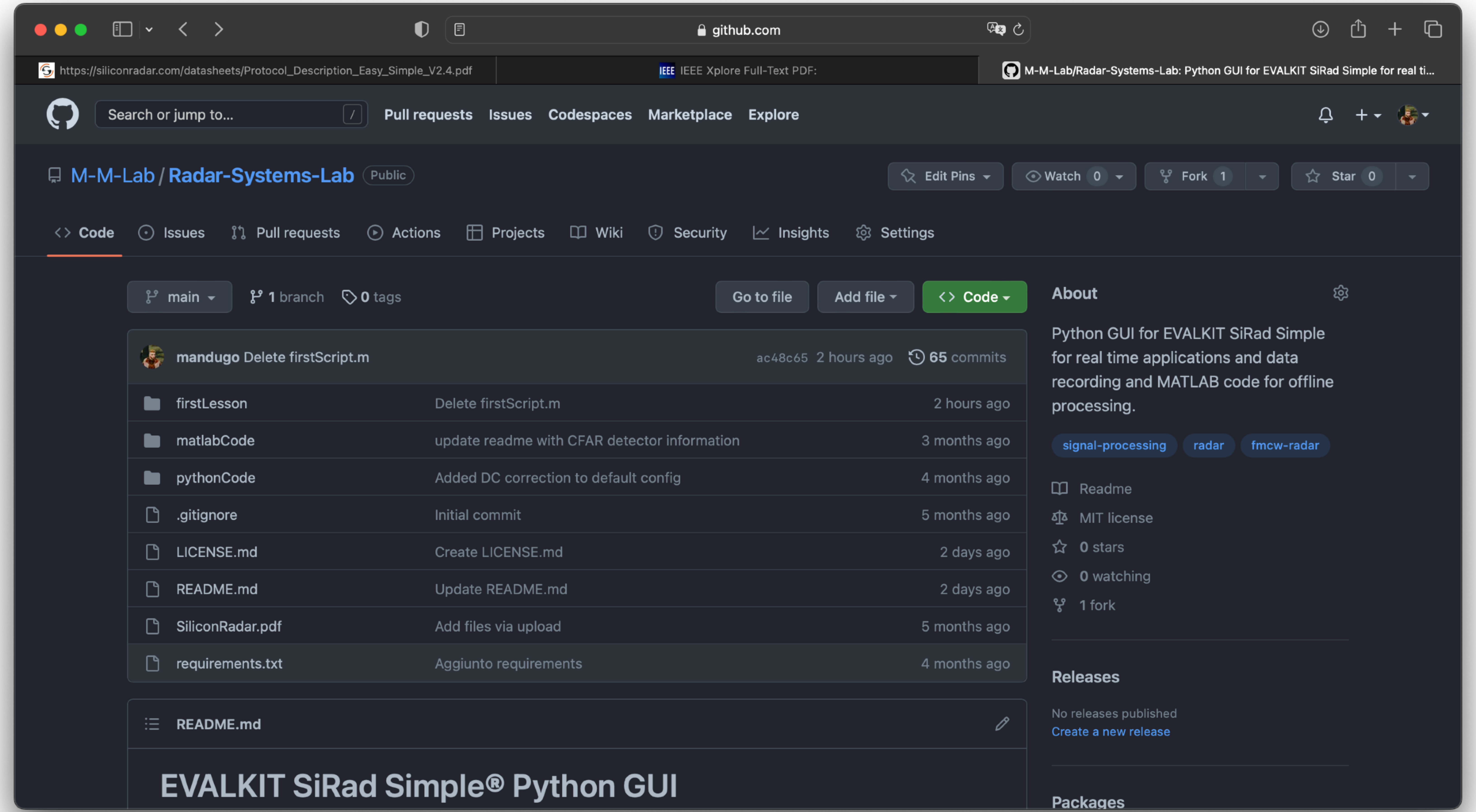
Since it may happen that the received buffer is empty, or it does not start with the right control character (i.e., error frame, information frame), the `readline()` is repeated until the data is acquired properly

The I samples are on odd indices, while the Q samples are on even indices

# A little bit of processing

## Range profiles evaluation

# GitHub repo Radar-Systems-Lab



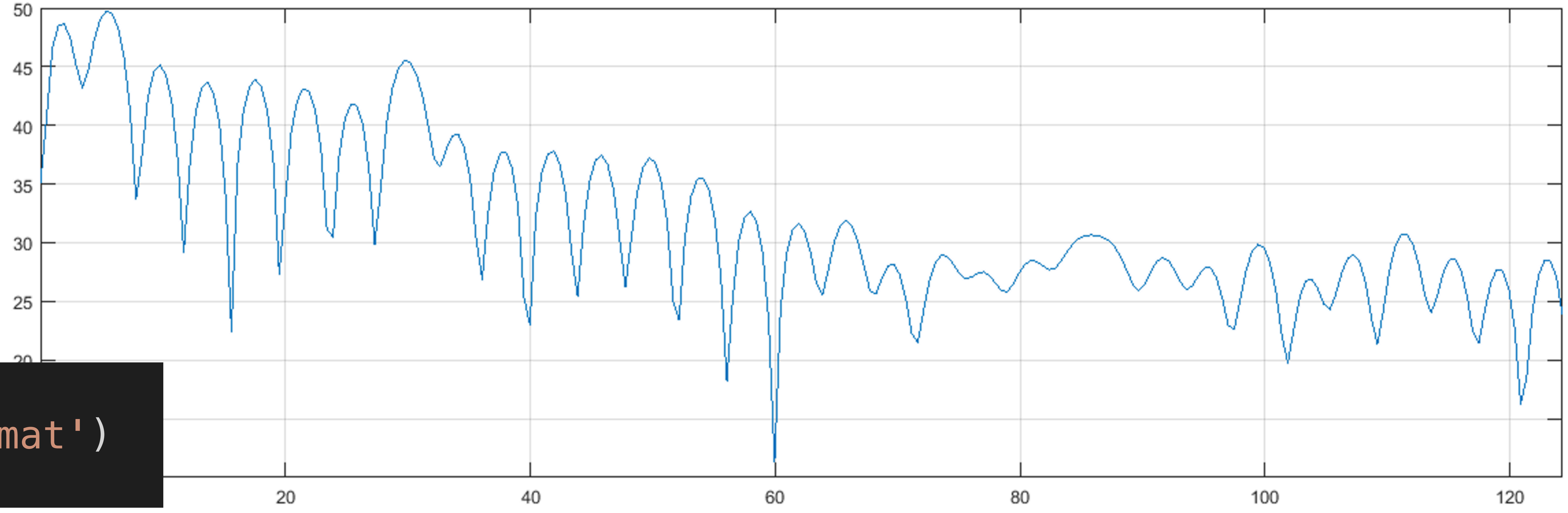

https://github.com/M-M-Lab/Radar-Systems-Lab

# Radar-Sistems-Lab/firstLesson

- First example: range profile evaluation
- Second example: live range profile plot
- Third example: basic signal processing
- Fourth example: CFAR detection
- Fifth example: range profiles waterfall

# .../firstLesson/firstExample.mlx

```
if length(complexData) == nSamples
    rangeProfile = fft(complexData,nfft);
    curveToPlot = abs(rangeProfile(1:end/2));
    [m,i] = max(curveToPlot);
    set(h(1),'YData',20*log10(curveToPlot));
    set(h(2),'XData',rangeVec(i),'YData',mag2db(m),'Marker','o','Color','r');
    title('Range Profile',strcat("maximum at ",num2str(round(rangeVec(i),2))," cm"))
end
```




```
load('target30.mat')
```

# .../firstLesson/secondExample.mlx

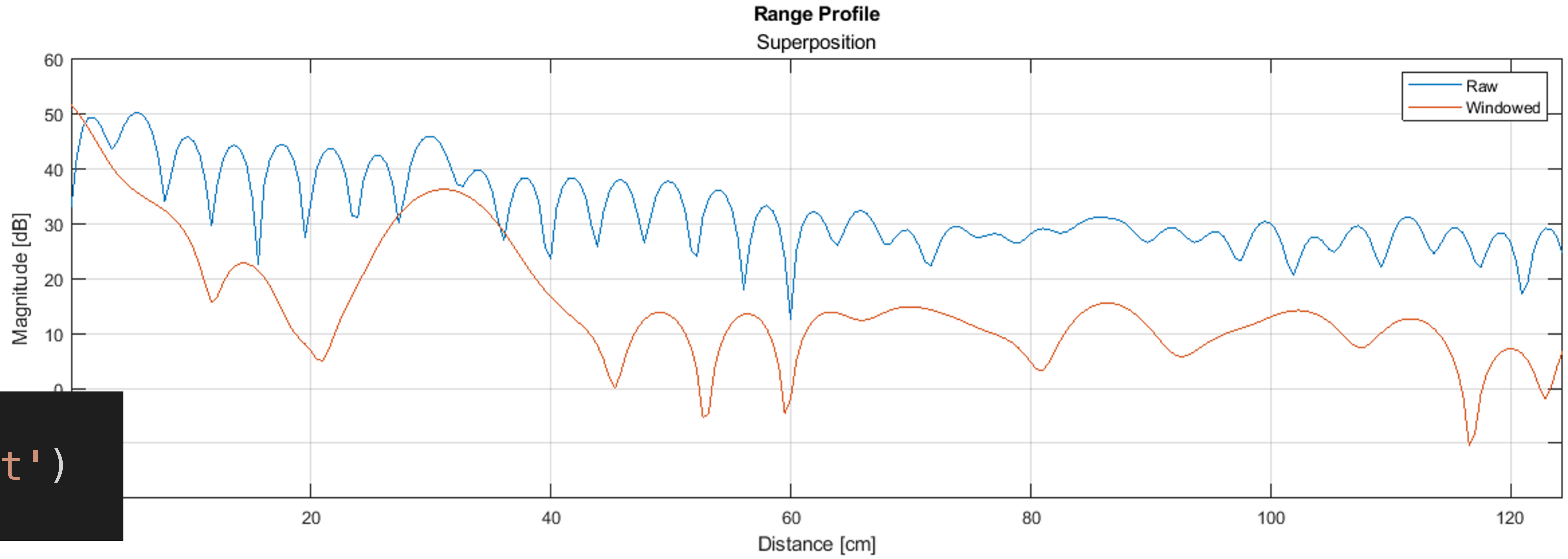
```
if length(complexData) == nSamples
    rangeProfile = fft(complexData,nfft);
    curveToPlot = abs(rangeProfile(1:end/2));
    [m,i] = max(curveToPlot);
    set(h(1),'YData',20*log10(curveToPlot));
    set(h(2),'XData',rangeVec(i),'YData',mag2db(m),'Marker','o','Color','r');
    title('Range Profile',strcat("maximum at ",num2str(round(rangeVec(i),2))," cm"))
end
```

**Same processing as firstExample.mlx, but live: the range profile is updated for every received frame and the board uses its internal trigger.**

# .../firstLesson/thirdExample.mlx

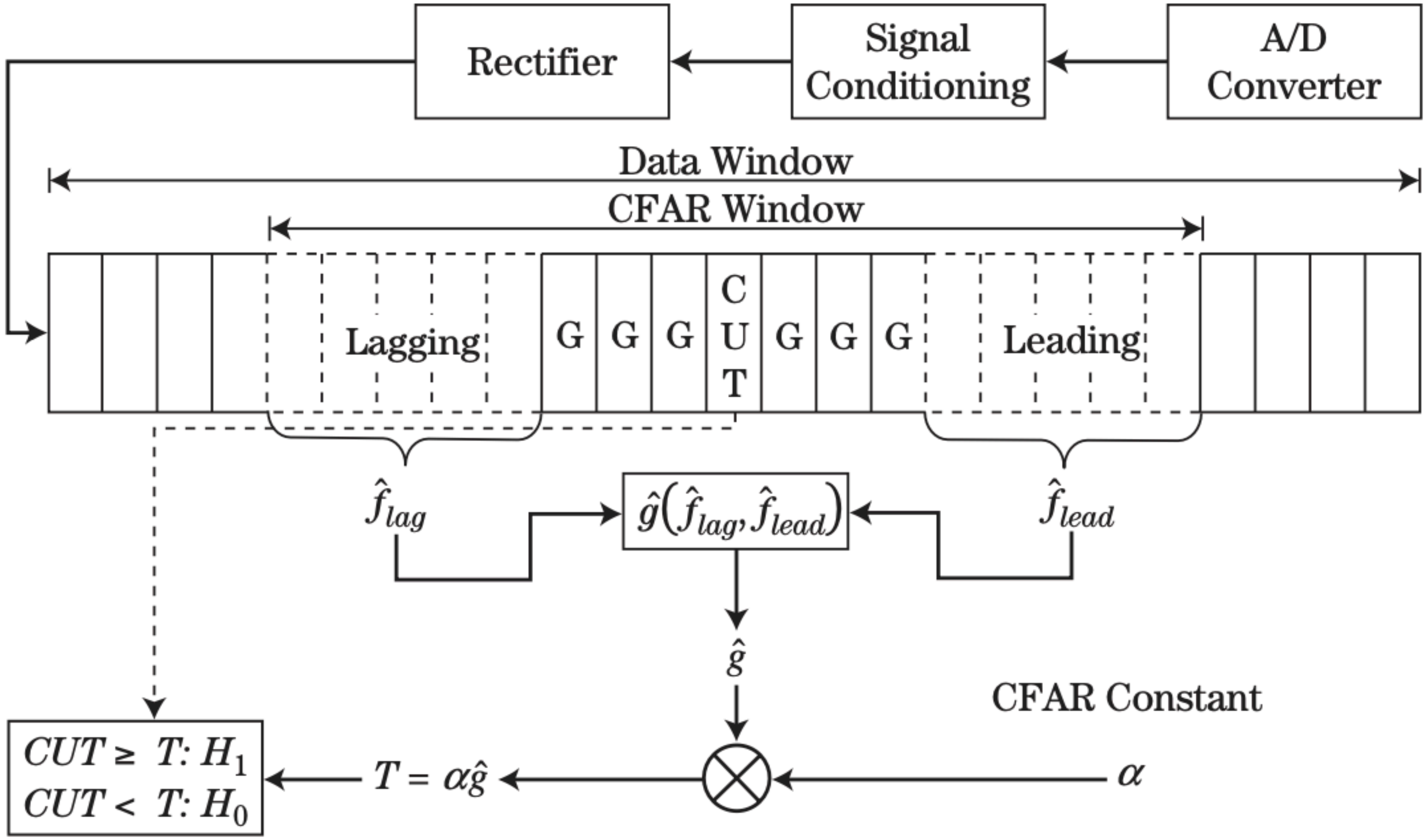
```
windowHann = hann(nSamples).';
rangeProfileWHann = fft(complexData.*windowHann,nfft);
curveToPlotWHann = abs(rangeProfileWHann(1:end/2));

figure
plot(rangeVec,20*log10(curveToPlot))
hold on
plot(rangeVec,20*log10(curveToPlotWHann))
```




```
load('target30.mat')
```

# .../firstLesson/fourthExample.mlx



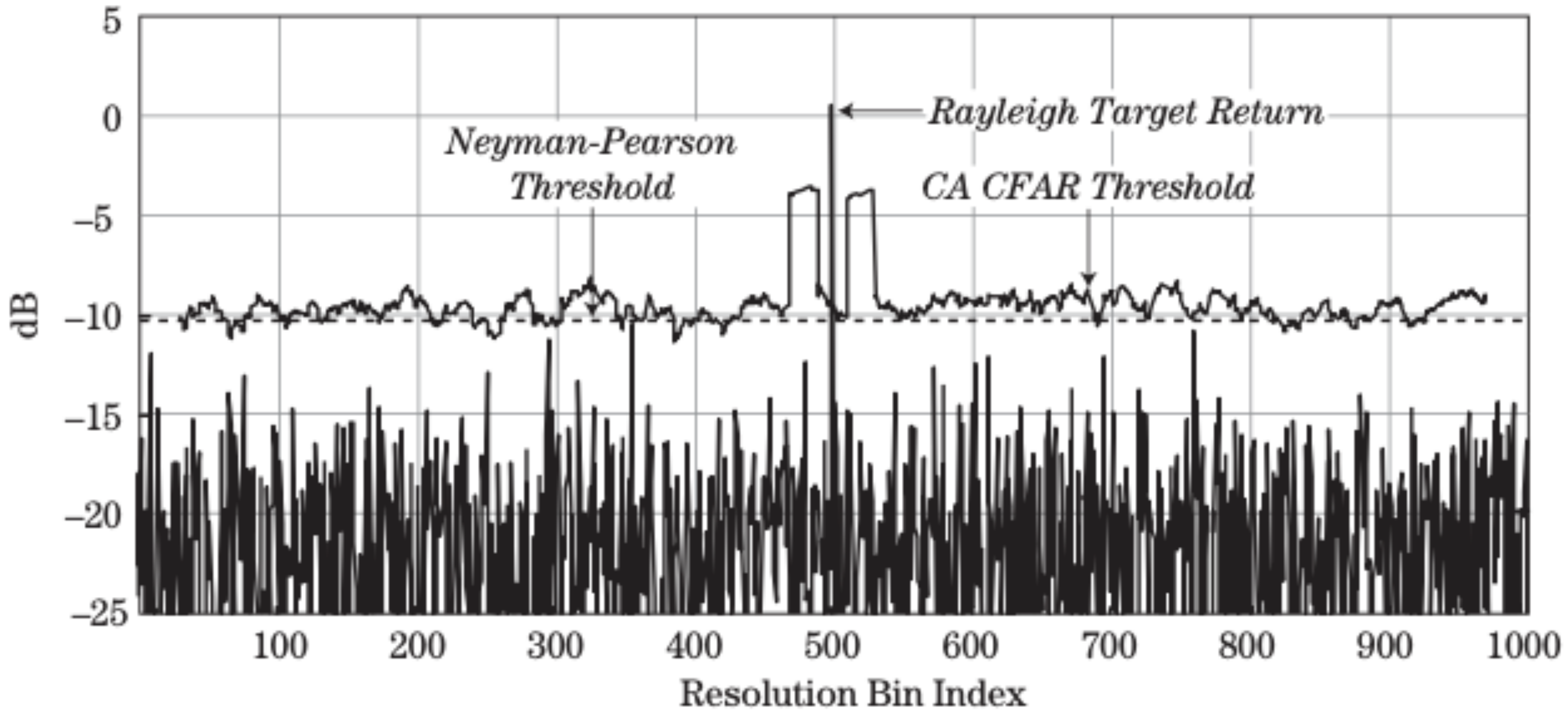

CFAR detectors estimate statistics of the interference from radar measurements and adjust the detector threshold to maintain a constant false alarm rate.

$$T = \alpha(P_{FA}) \cdot \hat{g}$$

Richards, M.A. and Scheer, J.A. and Scheer, J. and Holm, W.A. "Principles of Modern Radar: Basic Principles, Volume 1". Institution of Engineering and Technology (2010), ISBN 9781891121524

# …/firstLesson/fourthExample.mlx

## Cell Averaging (CA) CFAR




$$T = \alpha_{CA}(P_{FA}) \cdot \hat{g_{CA}}(z)$$

$$\alpha_{CA}(P_{FA}) = N\left[P_{FA}^{-1/N} - 1\right]$$

$$\hat{g_{CA}}(z) = \frac{1}{N}\sum_{n=1}^{N} z_n$$

**CA-CFAR's performance may degrade significantly in presence of interfering targets and clutter boundaries.**


Richards, M.A. and Scheer, J.A. and Scheer, J. and Holm, W.A. "Principles of Modern Radar: Basic Principles, Volume 1". Institution of Engineering and Technology (2010), ISBN 9781891121524

# …/firstLesson/fourthExample.mlx

## Greatest-Of (GO) CFAR

$$T = \alpha_{GO}(P_{FA}) \cdot \hat{g_{GO}}(z)$$

**GO-CFAR reduces clutter edge false alarms.**

$$\hat{g_{GO}}(z) = \max\left(\sum_{n=1}^{N/2} z_n, \sum_{n=N/2+1}^{N} z_n\right)$$

$$P_{FA-GO} = 2\left(1+\alpha_{GO}\right)^{-N/2} - 2\sum_{k=0}^{N/2-1}\binom{N/2+k-1}{k}\left(2+\alpha_{GO}\right)^{-(N/2+k)}$$

Richards, M.A. and Scheer, J.A. and Scheer, J. and Holm, W.A. "Principles of Modern Radar: Basic Principles, Volume 1". Institution of Engineering and Technology (2010), ISBN 9781891121524

# .../firstLesson/fourthExample.mlx

## Smallest-Of (GO) CFAR

$$T = \alpha_{SO}(P_{FA}) \cdot \hat{g_{SO}}(z)$$

**SO-CFAR addresses mutual target masking.**

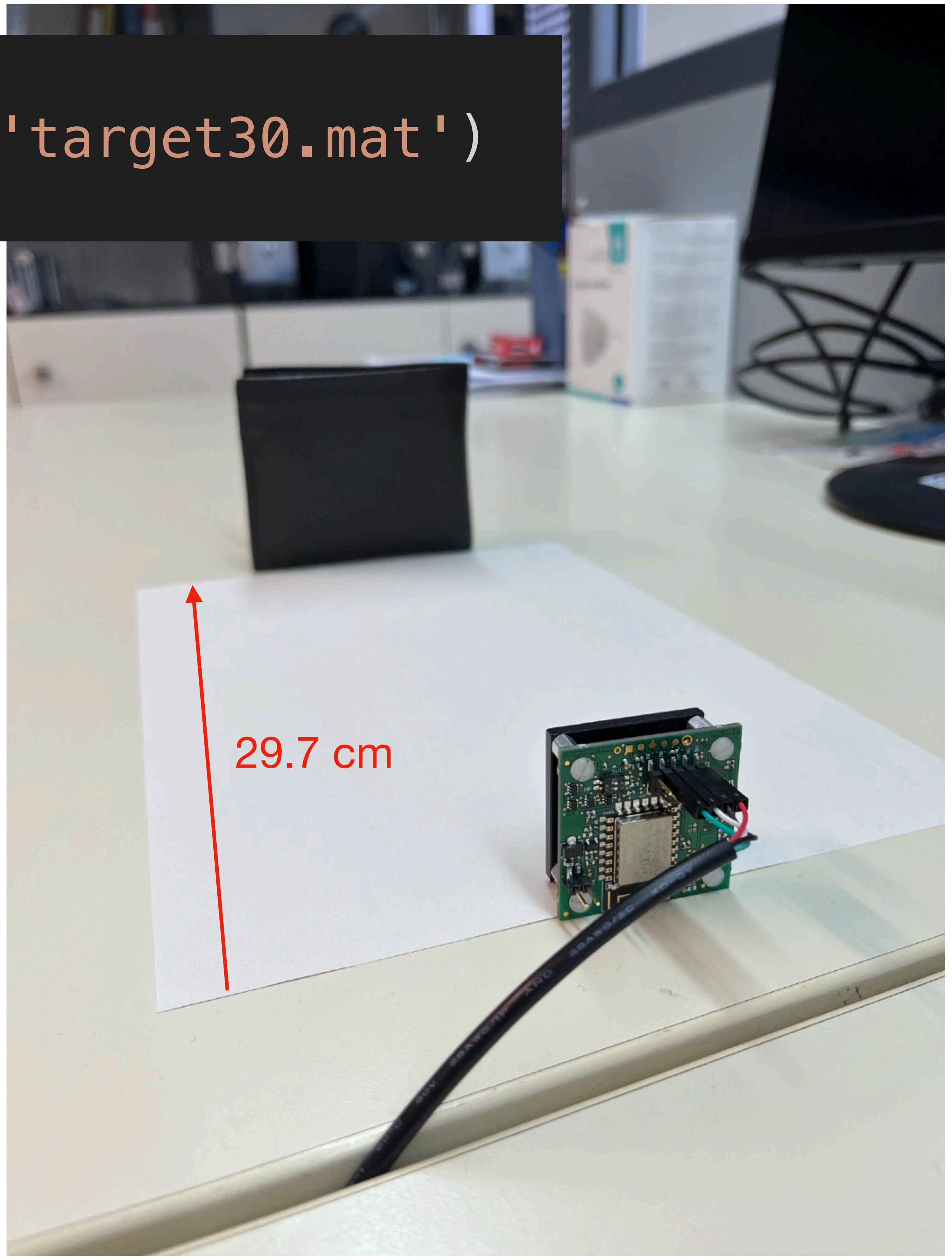
$$\hat{g_{SO}}(z) = \min\left(\sum_{n=1}^{N/2} z_n, \sum_{n=N/2+1}^{N} z_n\right)$$

$$P_{FA-SO} = 2\left(2+\alpha_{SO}\right)^{-N/2} \sum_{k=0}^{N/2-1} \binom{N/2+k-1}{k} \left(2+\alpha_{SO}\right)^{-k}$$

Richards, M.A. and Scheer, J.A. and Scheer, J. and Holm, W.A. "Principles of Modern Radar: Basic Principles, Volume 1". Institution of Engineering and Technology (2010), ISBN 9781891121524

# …/firstLesson/fourthExample.mlx

```
load('target30.mat')
```




```
load(‘target60.mat')
```