

Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Microelectronics



Master thesis:

CW Radar Sensor for Patient's Biological
Functions Monitoring

Jakub Richter

Communications, Multimedia and Electronics

July 2017

Thesis advisor

doc. Ing. Přemysl Hudec, CSc.

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Richter** Jméno: **Jakub** Osobní číslo: **399637**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávací katedra/ústav: **Katedra mikroelektroniky**
Studijní program: **Komunikace, multimédia a elektronika**
Studijní obor: **Elektronika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

CW radarový senzor pro sledování biologických funkcí pacientů

Název diplomové práce anglicky:

CW Radar Sensor for Patients' Biological Functions Monitoring

Pokyny pro vypracování:

1. Prostudujte konstrukci a vlastnosti CW radarových senzorů.
2. Na frekvenci 11 GHz navrhnete a realizujete senzor schopný na vzdálenost cca 1,5 m sledovat biologické funkce pacienta ležícího na lůžku.
3. Sestavte programové vybavení schopné analyzovat snímané signály.
4. Zaměřte se zejména na následující funkce:
 - Sledování dechu;
 - Sledování pohybů těla;
 - Indikace krizových stavů.
5. Výsledné parametry ověřte měřením a praktickými testy.
6. Zhodnoťte dosažené výsledky.

Seznam doporučené literatury:

- [1] Droitcour et al.: Range Correlation and I/Q Performance Benefits in Single-Chip Silicon Doppler Radars for Noncontact Cardiopulmonary Monitoring, IEEE MTT Vol.52 No.3 2004, str. 838 - 848.
[2] Chang K.: RF and Microwave Wireless Systems, John Wiley 2000.

Jméno a pracoviště vedoucí(ho) diplomové práce:

doc. Ing. Přemysl Hudec CSc., katedra elektromagnetického pole FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **16.02.2016**

Termín odevzdání diplomové práce: **31.07.2017**

Platnost zadání diplomové práce: **08.01.2018**

Podpis vedoucí(ho) práce

Podpis vedoucí(ho) ústavu/katedry

Podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Thesis instructions

1. Study construction and properties of CW radar sensors
2. Design and implement sensor working in 11 *GHz* capable to monitor biological functions of patients laying on the bed.
3. Design software capable to analyze measured signals.
4. Focus mainly in following functions:
 - Monitoring of respiration
 - Monitoring of body movements
 - Indication of critical conditions
5. Proof results with measurements and practical tests
6. Discuss obtained results

Declaration

I declare, that I have completed this master thesis on my own with help of my thesis advisors and consultants. In this work I used only literature which stands in the list of references and knowledge I have gained during my study.

I declare that I have no objections against borrowing or publicizing this work or its part with the permission of the Department of Microelectronics of CTU.

In Prague, July 2017

.....

signature

Acknowledgment

I would like to sincerely thank to both of my thesis advisors. Thank to prof. Chao-Hsiung Tseng from National Taiwan University of Science and Technology (NTUST), who led my thesis in this school, gave me many suggestions and advises for solutions. Thank to prof. Přemysl Hudec from Czech Technical University (CTU), who led my thesis in CTU and gave me many advises.

Thank to researcher and postgradual student Ing. Viktor Adler from CTU, for rich advices about Matlab programming and giving me principal solutions.

Abstrakt

Cílem této práce je navrhnout radarový senzor s kontinuální vlnou pro sledování biologických funkcí pacientů, jako je dýchání a bití srdce, do vzdálenosti 1.5 m a zpracováním signálu vypočítat naměřené výsledky sledování některých stavů pacientů.

Jádrem našeho radarového systému je radarový čip BGT24MTR11, který obsahuje veškeré nezbytné vysokofrekvenční komponenty. Proto musíme pouze navrhnout anténu, obvod pro zpracování výstupní soufázové i kvadrurní složky smíšeného signálu, aby byly uloženy do počítače pro další zpracování těchto signálů, a napájecí obvody a ochranu celého obvodu. Vysokofrekvenční signál je naladěn na $f_r \approx 24.2$ GHz.

Radarový čip BGT24MTR11 musí být nastaven příkazem skrz sběrnici SPI. Pro tento účel a pro získávání dat do počítače používáme desku Arduino UNO. Arduino UNO obsahuje piny sběrnice SPI a analogové vstupní piny s rozlišením 10-bitů.

Nakonec grafické uživatelské rozhraní v program Matlab je použito pro svoji schopnost a eleganci ve výpočtech zpracování signálu. A tedy zobrazení výstupní soufázové i kvadrurní složky smíšeného signálu, výpočet rychlé Fourierovy transformace a zobrazení spektra, aby byly vidět frekvence bití srdce a dýchání pacienta pro hodnocení výsledků.

Klíčová slova

Dopplerův radar, dýchání, bití srdce, patch anténa, Arduino, Matlab, spektrum

Abstract

The goal of this work is to design continuous-wave radar sensor for patient's biological functions, such as respiration and heartbeat, to be monitored, up to 1.5 m distance and use signal processing to compute measured results of some patient's condition observation.

The kernel of our radar system is radar chip BGT24MTR11 which include all necessary high frequency components. Therefore we only need to design the antenna, circuit to process in-phase and quadrature signals to be acquired by computer for signal processing, and supply voltage and protection of whole circuit. High frequency signal is tuned to $f_r \approx 24.2$ GHz.

The radar chip BGT24MTR11 has to be set up by command through SPI bus. For this purpose and for acquisition data to computer, board Arduino UNO is used. Arduino UNO includes SPI pins and analog input pins with 10-bits resolution.

Finally, graphical user interface in Matlab was used for its suitability and capability on signal processing, plotting in-phase and quadrature signals, computation for fast Fourier transform to get spectrum and plot the said spectrum to see frequencies of heartbeat and respiration to discuss results.

Key words

Doppler radar, respiration, heartbeat, patch antenna, Arduino, Matlab, spectrum

Contents

Thesis instructions.....	iii
Declaration	iv
Abstrakt.....	vi
Klíčová slova.....	vi
Abstract	vii
Key words	vii
Contents	viii
List of figures	xii
List of tables.....	xiv
Shortcuts.....	xv
I. Introduction.....	1
II. Basic of continuous wave (CW) radar	3
II.I. Doppler effect	3
II.II. Continuous wave (CW) radar	4
II.III. Radar equation	7
III. Antenna	8
III.I. Antenna basics	8
III.I.I. Near field and far field.....	8
III.I.II. Directivity and Gain of antenna	9
III.I.III. Half power beamwidth (HPBW)	10

III.I.IV.	Free space path loss (FSPL).....	10
III.II.	Transmission line	11
III.II.I.	Characteristic impedance	11
III.II.II.	Reflection coefficient.....	12
III.II.III.	Impedance matching.....	12
III.II.IV.	Wavelength in transmission line	13
III.II.V.	Microstrip line.....	13
III.III.	Patch antenna	14
III.III.I.	Computations for designing of single patch antenna	14
III.III.II.	Designing single patch antenna by software	17
III.III.III.	Designing 2x2 patch antenna array by software	21
III.III.IV.	Designing 4x4 patch antenna array by software.....	25
III.III.V.	Implementation into board.....	28
III.IV.	Antenna measurements	30
IV.	Construction of circuit	33
IV.I.	Basic arrangement.....	33
IV.I.I.	Low-noise amplifier (LNA).....	33
IV.I.II.	In-phase/Quadrature mixer.....	34
IV.I.III.	RF chip BGT24MTR11	34
IV.I.IV.	Block diagram of main part of circuit.....	36
IV.I.V.	Arduino UNO.....	37
IV.I.VI.	Differential amplifier	37

IV.I.VII.	Supply voltage	38
IV.I.VIII.	SPI high level consideration.....	39
IV.II.	Final configuration.....	39
V.	Software development.....	41
VI.	Main concept.....	41
V.II.	Commands for Arduino UNO	41
V.II.I.	Serial Pheripheral Interface bus (SPI).....	41
V.III.	Sending data to PC via port COMX.....	43
V.IV.	Signal processing in Matlab.....	44
V.IV.I.	Spectrum	44
V.IV.II.	Windowing.....	48
V.IV.III.	Matlab GUI	49
VI.	Experimental measurements	52
VI.I.	Calm breathing.....	52
VI.I.	Fast breathing.....	54
VI.II.	Deep breathing.....	55
VI.III.	Holding a deep breath	56
VI.IV.	Speech	57
VI.V.	Moving with body.....	58
VII.	Conclusion	60
References	62
Appendix	63

A.	Schematic of whole circuit.....	63
B.	Top layer layout	64
C.	Middle layer layout.....	65
D.	Bottom layer layout.....	66
E.	Radar photo front side.....	67
F.	Radar photo back side	68
G.	Photo of radar connected to Arduino UNO and PC	69
H.	Arduino code – SPI command	70
I.	Arduino code – collect data	71
J.	Photo from measurement	72
K.	Matlab openCSV function code.....	73
L.	Matlab plotSpec function code.....	74
M.	Matlab HammingWin function code.....	75

List of figures

Figure 1 System block diagram of the developed 24 GHz Doppler radar module [12].....	2
Figure 2 Approaching motorcycle – higher tone.....	3
Figure 3 Receding motorcycle – lower tone.....	4
Figure 4 Principle of radar used for measuring respiration and heart beating [2].....	5
Figure 5 Imagination of near field and far field [Wikipedia]	8
Figure 6 Schematic representation of elementary components of transmission line [1].....	11
Figure 7 Reflection coefficient in circuit [Wikipedia]	12
Figure 8 Impedance matching in circuit.....	13
Figure 9 Microstrip line [1].....	14
Figure 10 Patch antenna with resonator configuration of feeding line	15
Figure 11 Patch antenna with inner feeding line configuration.....	17
Figure 12 Single patch with 100 Ω inner feeding line configuration design.....	18
Figure 13 S11 parameter of designed single patch antenna in rectangular plot	19
Figure 14 Radiation pattern of designed single patch antenna in E-plane	20
Figure 15 Radiation pattern of designed single patch antenna in H-plane.....	21
Figure 16 Dividing power of electromagnetic wave to four paths.....	21
Figure 17 2x2 patch antenna array with 100 Ω inner feeding line configuration design	23
Figure 18 S11 parameter of designed 2x2 patch antenna array in rectangular plot	23
Figure 19 Radiation pattern of designed 2x2 patch antenna array in E-plane.....	24
Figure 20 Radiation pattern of designed 2x2 patch antenna array in H-plane	25
Figure 21 4x4 patch antenna array with 100 Ω inner feeding line configuration design	26
Figure 22 S11 parameter of designed 4x4 patch antenna array in rectangular plot	26
Figure 23 Radiation pattern of designed 4x4 patch antenna array in E-plane.....	27
Figure 24 Radiation pattern of designed 4x4 patch antenna array in H-plane	28

Figure 25 Dimensions and distances in mm of vias for connector [5].....	29
Figure 26 Designed 4x4 patch antenna array layout for measurement	29
Figure 27 Photo of designed 4x4 antenna array with connector	30
Figure 28 Measured vs. simulated S11 parameter of designed 4x4 patch antenna array in rectangular plot.....	31
Figure 29 Near-field measurement configuration	31
Figure 30 Near-field radiation pattern measurement principle.	32
Figure 31 Measured radiation pattern for designed 4x4 patch antenna array in H-plane	33
Figure 32 IQ-mixer principle [6].....	34
Figure 33 BGT24MTR11 Block Diagram	36
Figure 34 Block diagram of main part of whole circuit	37
Figure 35 Differential amplifier with shifted reference voltage.....	38
Figure 36 Structure of layers in the board [3]	40
Figure 37 Compensation structure [3].....	40
Figure 38 Basic concept of software for radar	41
Figure 39 SPI command time sequence to set up RF chip.....	43
Figure 40 Explanation of “twiddle factor” W_N^{nk} in DFT.....	45
Figure 41 Example of DFT for two sinusoidal.....	45
Figure 42 Even and Odd samples.....	47
Figure 43 Signal flow diagram for implementation FFT from DFT $N=4$	48
Figure 44 Hamming window.....	49
Figure 45 Principal diagram of Matlab code to show data and spectrum	50
Figure 46 Empty GUI of human biological functions monitor.	51
Figure 47 Measurement of calm breathing from distance 1.2 m, basic view.....	53
Figure 48 Measurement of calm breathing from distance 1.2 m, zoomed view	53
Figure 49 Measurement of calm breathing from distance 1.2 m, zoomed view, 2x Hamming	

window	54
Figure 50 Measurement during fast breathing from distance 1.5 m, basic view without Hamming window	55
Figure 51 Measurement during deep breathing from distance 1.5 m, zoomed view	56
Figure 52 Measurement during holding a deep breath from distance 1.5 m, zoomed view.....	56
Figure 53 Measurement during speech from distance 1.5 m with zoomed view and 2x Hamming window	57
Figure 54 Measurement during part off time moving with 1 m away body	59
Figure 55 Measurement during part off time moving with 1 m away body, with Hamming window.....	59

List of tables

Table 1 List of substrate RO4350B properties provided by vendor [4]	15
Table 2 Final dimensions of designed single patch antenna with 100Ω	17
Table 3 Final dimensions of designed 2x2 patch antenna array with 100Ω	22
Table 4 Final dimensions of designed 4x4 patch antenna array with 100Ω	25
Table 5 Values of bits in SPI command used to set up RF chip	42

Shortcuts

Z	characteristic impedance (complex) (Ω)
R	resistance (real part of Z) (Ω)
X	reactance (imaginary part of Z) (Ω)
Ω	Ohm, unit of impedance
i, j	imaginary unit of complex number
AC	alternating current
DC	direct current
c	velocity of wave, if light $c = 299792458$ m/s
m	meter, unit of length
s	second, unit of time
λ	wavelength (m)
V	Voltge (V - Volt)
I	Current (A)
A	Amper, unit of current
f	frequency (Hz)
Hz	Hertz, unit of frequency
GHz	Gigahertz ($10^9 Hz$)
ω	phase frequency $\omega = 2 * \pi * f$ (s^{-1})
μ_0	permeability of vacuum $\mu_0 = 4 * \pi * 10^{-7} \cong 12.566 * 10^{-7}$
ε_0	permittivity of vacuum $\varepsilon_0 = \mu_0^{-1} * c^{-2} \cong 8.854 * 10^{-12}$
E	intensity of electric field (V/m)
H	intensity of magnetic field (A/m)
I/Q	in-phase, quadrature
EKG	Electrocardiogram

I. Introduction

One of basic procedures done in hospitals for people needed to be treated is to get respective data depending on the condition the patient is under. Such data are heartbeat, respiration and many others. In doing so, this requires an equipment or tool to determine a specific data. For example, getting heartbeat, an electrocardiogram is commonly used. However, after prolonged time of usage, it can be uncomfortable for the patient. So scientists in the 1970s tried to find solution, with low power consumption, to observe both respiration and heartbeat frequency without physical contact with patient. This solution is called Doppler radar sensor.

Radar (RADio Detection And Ranging) is used in many applications. The most used ones are in military and or in airport for monitoring flying objects. Radar uses Doppler effect, as described in the next chapter, with high frequency electromagnetic waves to detect moving objects.

The system block diagram of the developed 24 GHz Doppler radar module is shown in Figure 1. We are using 24 GHz because I am developing this radar in university NTUST and NTUST is focused in other frequencies than 11 GHz.

Kernel of whole circuit is radar chip which includes all necessary radar components described below. The high frequency signal is generated by local oscillator and splitted to two paths. Signal in one path is transmitted by antenna, reflected by 1.5 m away patient's chest and received by receiving antenna. This received signal is amplified by Low-noise amplifier and mixed with signal of local oscillator from second path, which is splitted to in-phase and $\pi/2$ delayed "quadrature" signals. The output are two signals which carry information about differences in frequency. These signals has to be amplified to be recognized by analog input of Arduino UNO. Signals are sent to computer for signal processing from Arduino UNO as digital values. Arduino UNO also control the gain of the power amplifier in radar transmission section by using SPI bus. Signal processing in computer shows spectrum where user can find the peaks of spectra.

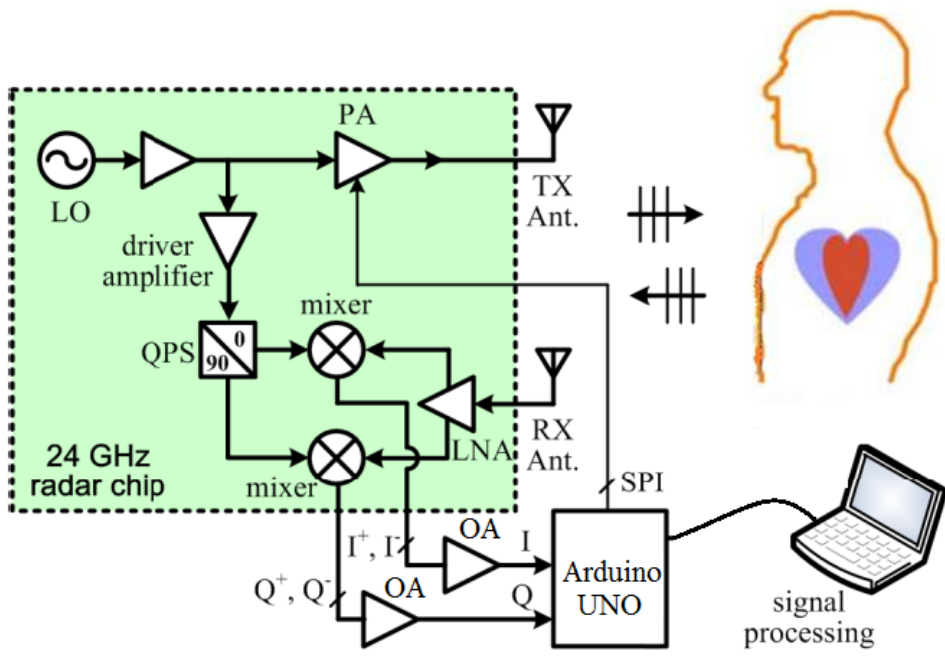


Figure 1 System block diagram of the developed 24 GHz Doppler radar module [12]

II. Basic of continuous wave (CW) radar

II.I. Doppler effect

Doppler effect means that frequency or wavelength of any wave is different than frequency of source of radiation, if source or receiver or both are moving to each other. This effect found Austrian physicist Christian Doppler and proposed it in 1842 in Prague. The **Doppler effect** is defined as

$$f = \left(\frac{c + v_r}{c - v_s} \right) * f_0 \quad (1)$$

where	f	received frequency
	f_0	transmitted frequency
	c	velocity of wave
	v_s	velocity of source of wave
	v_r	velocity of receiver

To imagine how these equation works I draw example from common life to next figures to understand. In Figure 2 we can see, that source of acoustic wave (motorcycle) is moving with velocity v_s toward listener. Velocity of listener is $v_r = 0 \text{ m/s}$. The velocity of acoustic wave in air is $c \approx 340 \text{ m/s}$. According the equation (1), the listener hear higher tone of acoustic wave than the source produce.



Figure 2 Approaching motorcycle – higher tone

But in case of Figure 3 if source of acoustic wave (motorcycle) is receding from listener with velocity $-v_s$ and listener is not moving $v_r = 0 \text{ m/s}$, it mean that according the equation (1) listener hear lower tone of acoustic wave than the source produce.



Figure 3 Receding motorcycle – lower tone

In our case it is electromagnetic wave with velocity of light is $c \approx 3 * 10^8$ m/s.

II.II. Continuous wave (CW) radar

In naturally speaking radar is equipment which can detect movement of conductive materials in direction of radiation of antenna toward or from antenna. Continuous electromagnetic wave radiated from antenna is in some part of power reflected from conductive object and received back to antenna. Another part of power is dissipated in object and transformed into heat. If the object is moving in direction of radiation antenna with finite velocity, the **Doppler effect** occurs and electromagnetic wave is reflected with different frequency and received by antenna. This received signal is combined in mixer with the same signal as in transmitter and output of this process is low frequency signal, which carry differences in frequency, defined by time-varying displacement $x(t)$ of subject under test (SUT) in time. If $x(t)$ is too and value of output reach maximum, it reverse its value. For example direction of $x(t)$ is toward the antenna, then output signal grows. If output signal reach maximum, the value of output start to descend.

Some radars use one antenna as transceiver and receiver for one direction, some radars use two antennas, one as transceiver and one as receiver but they always have to be just beside each other with considering of isolation and look at the same direction.

CW radars are used mostly for measurement of speed, in military or in airports for observing flying objects like planes or missiles, but also in Healthcare to observe movements of patients which is our case.

The principle how to use radar in Healthcare is seen in Figure 4. The goal is to obtain information about respiration and heart beating. Antenna has to be in front of lower part of chest of lying or sitting patient. Lower part of chest because there supposed to be heart and also some patients move only with stomach during respiration so the stomach also should be in part of radiation pattern of antenna. When patient inhale, his chest or stomach will grow in size and when patient exhale, his chest or stomach will dwindle. So the direction of radiation of antenna should be perpendicular to patient's chest to get the best results. The principle of heart beating is the same.

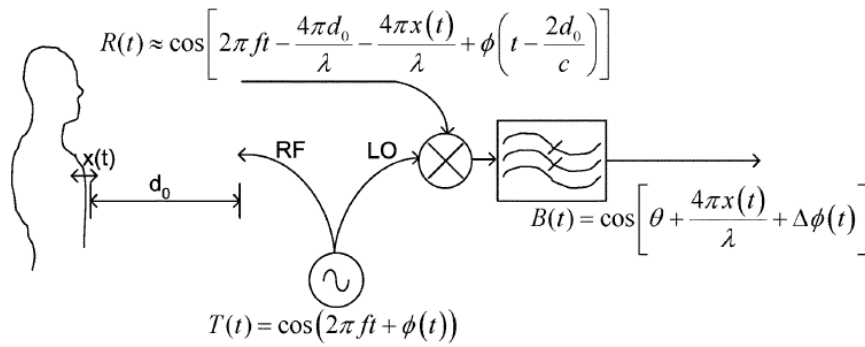


Figure 4 Principle of radar used for measuring respiration and heart beating [2]

The basic component is oscillator which send signal to antenna and the same signal to mixer. Received signal is combined in mixer with original signal and output of mixer is low frequency signal with higher frequency noise which has to be filtered.

The next paragraph is cited from [2].

If we don't consider amplitude, the transmitted electromagnetic wave can be defined as

$$T(t) = \cos(2 * \pi * f * t + \phi(t)) \quad (2)$$

where f is oscillation frequency, t is elapsed time and $\phi(t)$ is phase noise of oscillator. This electromagnetic wave is reflected from SUT which is in distance d_0 from antenna of radar. Suppose that SUT is moving with time-varying displacement $x(t)$, so the total distance between transmitter and receiver is $2 * d(t) = 2 * d_0 + 2 * x(t)$. Then we can find received signal as

$$R(t) = \cos\left[2 * \pi * f * \left(t - \frac{2 * d * \left(t - \frac{d(t)}{c}\right)}{c}\right) + \phi\left(t - \frac{2 * d * \left(t - \frac{d(t)}{c}\right)}{c}\right)\right] \quad (3)$$

where c is propagation's signal velocity. By substituting $d(t)$ and neglecting term $d(t)/c$ and term $x * (t - d(t)/c)/c$ in phase-noise term, since $x(t) \ll d_0$, we can get equation

$$R(t) \approx \cos \left[2 * \pi * f * t - \frac{4 * \pi * d_0}{\lambda} - \frac{4 * \pi * x(t)}{\lambda} + \phi \left(t - \frac{2 * d_0}{c} \right) \right] \quad (4)$$

“The received signal is similar to the transmitted signal with a time delay determined by the nominal distance of the target and with its phase modulated by the periodic motion of the target. The information about the periodic target motion can be demodulated if this signal is multiplied by LO signal that is derived from the same source as the transmitted signal.” [2]

When the received and LO signals are mixed and the output is low-pass filtered, the resulting baseband signal is

$$B(t) = \cos \left[\theta + \frac{4 * \pi * x(t)}{\lambda} + \Delta\phi(t) \right] \quad (5)$$

where

$$\Delta\phi(t) = \phi(t) - \phi \left(t - \frac{2 * d_0}{c} \right) \quad (6)$$

is the residual phase noise and

$$\theta = \frac{4 * \pi * d_0}{\lambda} + \theta_0 \quad (7)$$

is the constant phase shift dependent on the nominal distance to the SUT d_0 . Several factors affect the value of θ_0 such as the phase shift at the reflection surface (near 180°) and any distance between the mixer and antenna. [2]

II.III. Radar equation

Radar equation is represented by relationship between transmitted power from radar and received power to radar. There are many aspects which affect received power like power density decreasing by spherical wave propagation $1/(4 * \pi * d_0^2)$, Gain of transmitting G_t and receiving G_r antenna, radar cross-section RCS and radiation pattern factor of transmitting $f_t(\theta, \phi)$ and receiving $f_r(\theta, \phi)$ antenna where (θ, ϕ) are angles in E and H plane. The distance between antennas and monitored object is d_0 .

For power density of transmitted wave S_1 impacting to monitored object we can say that

$$S_1 = P_t * \frac{G_t * f_t^2(\theta, \phi)}{4 * \pi * d_0^2} \quad (8)$$

The wave is reflected from SUT also as spherical wave. So the power density of reflected wave S_2 is defined as

$$S_2 = \frac{S_1 * RCS}{4 * \pi * d_0^2} \quad (9)$$

Radar cross-section RCS is effective area of object from which the wave was reflected. The power or received wave is defined as

$$P_r = A_{r_{eff}} * f_r^2(\theta, \phi) * S_2 = \frac{G_r * \lambda^2}{4 * \pi} * f_r^2(\theta, \phi) * S_2 \quad (10)$$

So if we insert equations (8) and (9) to equation (10) we will get radar equation

$$P_r = P_t \frac{f_t^2(\theta, \phi) * G_t * G_r * \lambda^2 * f_r^2(\theta, \phi) * RCS}{(4 * \pi)^3 * d_0^4} \quad (11)$$

III. Antenna

III.I. Antenna basics

Antenna is electronic system which employs electromagnetic waves up to frequencies of gigahertz. Antenna is essential part of a radio system. It is device which can radiate and receive electromagnetic energy in an efficient and desired manner. Antennas are mostly made from metal but some other materials can be used. [1]

There are many kinds of antenna for many purposes. Basic types of antenna are dipole, monopole, Yagi-Uda, patch and a lot of other.... In this work we are using patch antenna.

People are using antennas daily with their cellphones, televisions, radio, radars, etc.

III.I.I. Near field and far field

Near field and far field are regions of electromagnetic field around the object like antenna where electromagnetic waves are propagated in different way. To imagine near field and far field I found the best picture in Wikipedia

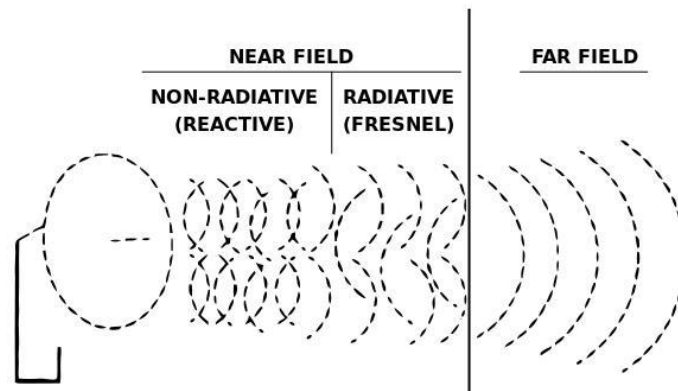


Figure 5 Imagination of near field and far field [Wikipedia]

For easy comprehension let's start explanation about far field. In far field the electromagnetic waves propagate as the real radio waves in direction from transmitting antenna. The intensity of electric and magnetic field are perpendicular to each other so they are predictable. The boundary between near field and far field is called the Fraunhofer distance, which is defined as

$$d_F = \frac{2 * D^2}{\lambda} \quad (12)$$

where d_F is Fraunhofer distance in meters, D is the longest diameter of antenna and λ is wavelength.

Near field is splitted into two regions. Non-radiative and radiative region called Fresnel. The boundary between these two regions is about $\lambda/(2 * \pi)$ from transmitting antenna. The non-radiative region called also reactive is the nearest region to antenna. It is called reactive because in this region the electromagnetic field can be absorbed by another component. Intensity of electric and magnetic fields are too complex to be predicted. The Fresnel region is radiative and electromagnetic wave inside the Fresnel region cannot be coupled. But intensity of electric and magnetic field are still complex so they still cannot be predicted.

Most of applications with antennas, included our case, are used in far field region.

III.I.II. Directivity and Gain of antenna

Directivity is concentration of radiated power from in a particular direction. It is defined as the ratio of the radiation intensity in a given direction from the antenna to the radiation intensity averaged over all directions. The average radiation intensity is equal to the total radiated power divided by $4 * \pi$. If the direction is not specified, the direction of maximum radiation is implied. [1] In mathematic formula, **directivity** is described as

$$D = \frac{U(\theta, \phi)}{U(\theta, \phi)_{av}} = \frac{4 * \pi * U(\theta, \phi)}{P_t} \quad [1] \quad (13)$$

where P_t is total radiated power in Watts, $U(\theta, \phi)$ is radiation intensity in ($W/unit$) and (θ, ϕ) are angles in E and H plane.

In case of directional antenna the directivity is closely linked to the half-power beamwidth in the E and H-planes. If beamwidth is known, we can say that directivity is

$$D = \frac{4 * \pi}{\theta_{HP} * \phi_{HP}} \quad (14)$$

where θ_{HP}, ϕ_{HP} are half-power beamwidths of two principal orthogonal planes. [1]

Gain is defined similar to directivity but in this case we are not talking about intensity in angle but in one direction and it is related to total input power accepted by antenna. So the gain is

$$G = \frac{4 * \pi * U}{P_{in}} \quad (15)$$

where P_{in} is total input power accepted by antenna from circuit in Watts and U is radiations intensity in ($W/unit$). As you can see, the difference between gain and directivity is in power because in real life transmitted power from antenna P_t is less than total input power accepted by antenna from circuit P_{in} because of impedance matching between antenna and input transmission line or connector. [1] The impedance matching will be described in chapter afterwards.

Following equation show the difference described above.

$$G = \frac{P_t}{P_{in}} * D = \eta_e * D \quad (16)$$

where η_e is called the radiation efficiency factor of the antenna. [1]

III.I.III. Half power beamwidth (HPBW)

Half power beamwidth is defined as angle of radiation pattern in H or E plane of antenna, where the power is higher than half of maximum power. In logarithmic scale it is higher than $\log 0.5 = 3 \text{ dB}$ and less than maximum of power. For example if $G_{max} = 10 \text{ dB}$, the half power beamwidth is as big as $G_{angle} > 7 \text{ dB}$.

III.I.IV. Free space path loss (FSPL)

Power loss of radiated electromagnetic wave is proportional to the square of distance between transceiver and receiver and its frequency. If electromagnetic wave is spherical, we can say that

$$\text{FSPL} = \left(\frac{4 * \pi * d}{\lambda} \right)^2 \quad (17)$$

where d is distance between transceiver and receiver and λ is wavelength.

III.II. Transmission line

Before the electromagnetic wave is radiated by antenna, it has to be transferred from local oscillator (LO) to antenna. It can be transferred through waveguide, coaxial cable or transmission line. Transmission line is the best solution in consideration of space, which is used in our case and contain copper line, copper ground plane and dielectric substrate between.

Transmission line has its own impedance, which is composed of serial resistance and inductive and parallel capacitive and conductance characters as you can see in Figure 6. The resistance R represents conductive loss of transmission line over length of line, conductance L represent self-inductance of transmission line over length of line, the capacitance C between two conductors is represented by shunt capacitor and conductance G represents dielectric between two conductors. [1]

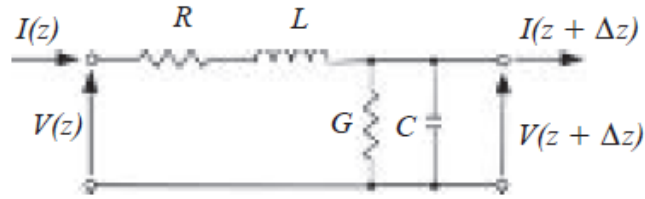


Figure 6 Schematic representation of elementary components of transmission line [1]

Where $I(z + \Delta z)$ is current in distance $(z + \Delta z)$ and $U(z + \Delta z)$ is voltage in distance $(z + \Delta z)$. The Δz is length of line. [1]

III.II.I. Characteristic impedance

Because it is not main focus of this thesis, I will skip many derivations which can be found in many microwave text books. I am familiar with textbook [1].

After long deriving we can get important parameter of transmission line which is called characteristic impedance and it is defined as

$$Z_0 = \frac{V_+(z)}{I_+(z)} = \sqrt{\frac{R + j\omega * L}{G + j\omega * C}} \quad (18)$$

for lossy transmission lines and by neglecting R and G it is

$$Z_0 = \sqrt{\frac{L}{C}} \quad (19)$$

for lossless transmission lines. [1]

III.II.II. Reflection coefficient

Characteristic impedance is main impedance of high-frequency circuit. It is mostly 50 Ω , 75 Ω or 100 Ω but it can be any value which designer choose. If there is different impedance on the way, some power of electromagnetic wave can be reflected back. For this the term **reflection coefficient** has been founded as

$$\Gamma_0 = \frac{Z_L - Z_0}{Z_L + Z_0} \quad (20)$$

and shown in Figure 7.

The equation (20) says that part of incident electromagnetic wave coming from transmission line with characteristic impedance Z_0 can be reflected on the boundary of for example load with impedance Z_L , if $Z_L \neq Z_0$ as much as value of the reflection coefficient is further from zero. The wave is reflected with opposite phase.

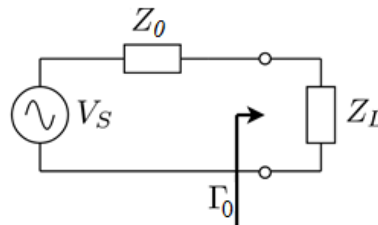


Figure 7 Reflection coefficient in circuit [Wikipedia]

III.II.III. Impedance matching

Some components in circuit has different impedance so we have to compute impedance matching. If components are not matched, some power can be reflected as described in previous chapter.

For example, if we want to split power to two symmetrical ways, the electromagnetic wave see in view of impedance this two ways as parallel combination. So according the formula for parallel circuit $Z_0 = Z_L * Z_L / Z_L + Z_L$, the Z_L should be $Z_L = 2 * Z_0$.

But if there is some transmission line or component with different characteristic impedance in

serial path of electromagnetic wave for example connector with impedance Z_L , we should match impedance by **resonator** with impedance Z_r between connector and transmission line with impedance Z_{in} . This is shown in Figure 8.

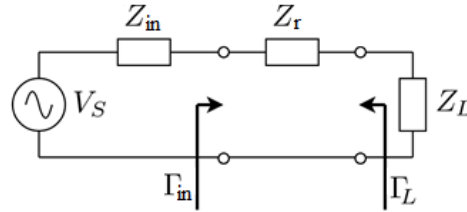


Figure 8 Impedance matching in circuit

After long deriving we can get simple equation to compute impedance of this resonator

$$Z_r = \sqrt{Z_{in} * Z_L} \quad (21)$$

Transmission line impedance resonators are usually $\lambda/4$ long because if some small part of wave is reflected from the end of resonator, it meets incoming wave in opposite phase and it's deleted by incoming wave and no standing wave occur.

III.II.IV. Wavelength in transmission line

We can also need to consider that wavelength depends not only on frequency and speed of light. Every transmission line has some substrate with **relative permittivity** ϵ_r between line and ground plane and electromagnetic wave in high frequency propagate also in the closest region around line. And the ϵ_r has influence to velocity of propagated electromagnetic wave in term of

$$v_f = \frac{c}{\sqrt{\epsilon_r}} \quad (22)$$

So in definition of wavelength and considering of equation (22) we can say that

$$\lambda = \frac{v_f}{f} = \frac{c}{f * \sqrt{\epsilon_r}} \quad (23)$$

III.II.V. Microstrip line

There are many kinds of transmission line like microstrip line, stripline, complanar waveguide and many others. The most used is microstrip line which is also used in this work and it's shown in Figure 9, where black area is copper and grey area is substrate with dielectric constant ϵ_r , t is

thickness of microstrip, w is width of microstrip and h is height of substrate. These parameters are provided by vendor. Figure 9 also shows the fields distributions.

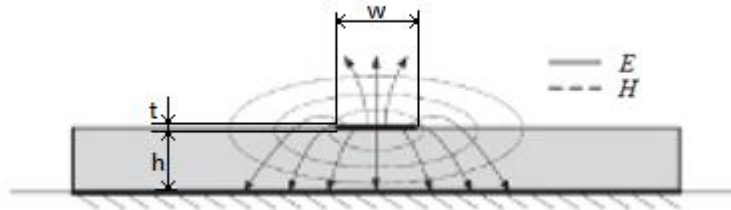


Figure 9 Microstrip line [1]

All these parameters have influence to impedance of microstrip line but the most is in width w which is indirect to impedance of line. Because the formulas to compute this impedance are quite difficult, we are using simple calculators like TXLine provided by National Instruments or LineCalc provided by Advanced design system.

III.III. Patch antenna

Patch antenna is also called microstrip antenna for reason. It is nothing else then wider microstrip resonator. This antenna is very popular because it costs nothing if designer want to have antenna as part of microwave circuit on the board. The shape of patch antenna is rectangle if we neglect thickness of copper. There are many consideration about feeding line. Mostly the feeding line is connected to antenna by quarter-wave transformer configuration as seen in Figure 10 or directly by inner configuration as seen in Figure 11.

To get better gain and more narrow beamwidth we can use solution of array of patches. For good symmetry are very popular 2x2 or 4x4 patch antennas.

III.III.I. Computations for designing of single patch antenna

First is good to make list of provided values and start with the most important parameter, which is frequency. The chip radar BGT24MTR11, which is heart of high-frequency part of my device, can receive frequencies 24 ÷ 26 GHz according the datasheet [3]. As there is chamber in the school,

which can measure in range 23 ÷ 25 GHz, it would be good to choose frequency just a little bit higher than 24 GHz. So for designing we can choose frequency $f = 24.25$ GHz.

Because in our laboratory we use substrate RO4350B, vendor Rogers Corporation provide following parameters according the datasheet [4]. The Table 1 shows properties of used substrate.

Table 1 List of substrate RO4350B properties provided by vendor [4]

Dielectric constant	ϵ_r	3.66
Loss tangent	$\text{tg}\delta$	0.0037
Thickness of substrate	h	0.254 mm
Thickness of copper	t	35 μm

Because we will need to design 2x2 or 4x4 patch antenna to get better gain, we will design antenna matched to 100 Ω transmission line for easier implementation which will be seen later.

First is good to start all computations for antenna with resonator feeding line as seen in Figure 10.

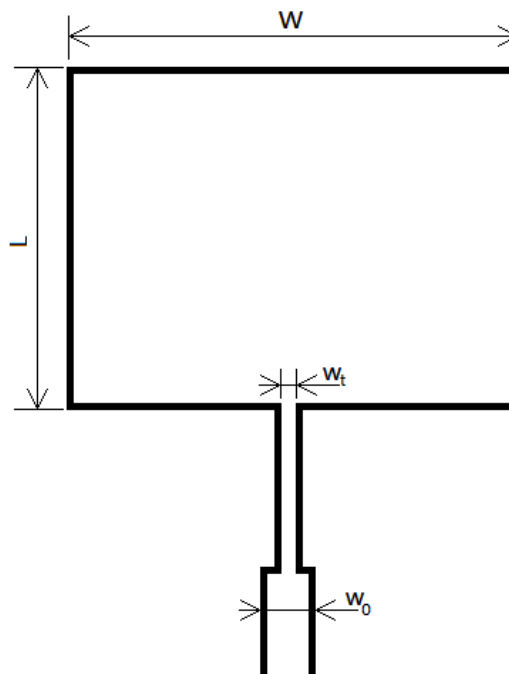


Figure 10 Patch antenna with resonator configuration of feeding line

The next designing process is cited from [1].

Width of patch is defined as

$$W = \frac{\sqrt{2/(\epsilon_r + 1)}}{2 * f * \sqrt{\mu_0 * \epsilon_0}} \quad (24)$$

so if we set up values we get $W \cong 4.05 \text{ mm}$. Now because the electric field is affected not just by dielectric but also by air around microstrip, we have to compute effective dielectric constant

$$\epsilon_{reff} = \frac{\epsilon_r + 1}{2} + \frac{\epsilon_r - 1}{2 * \sqrt{1 + 12 * h/W}} \quad (25)$$

In our case the $\epsilon_{reff} \cong 3.335$. Now we want to get length of antenna L . But according of direction of propagation of electromagnetic wave, the fringing effect occur. That mean that true effective length fulfilling resonant condition of patch antenna is longer than length of microstrip.

$$L_{eff} = L + 2 * \Delta L \quad (26)$$

Where effective length L_{eff} and additive length ΔL are known by equations

$$L_{eff} = \frac{1}{2 * f * \sqrt{\epsilon_{reff}} * \sqrt{\epsilon_0 * \mu_0}} \quad (27)$$

$$\Delta L = (\epsilon_{reff} + 0.3) * (W/h + 0.264) / [(\epsilon_{reff} - 0.258) * (W/h + 0.8)] \quad (28)$$

So if we set up all values to equations (27) and (28) we will get $L_{eff} \cong 3.385 \text{ mm}$ and $\Delta L \cong 0.12 \text{ mm}$ and set up these results to equation (26), we will get $L \cong 3.15 \text{ mm}$.

So the dimensions of antenna are set. Now it's time to match the quarter-wavelength transformer. The typical value of impedance at the edge of rectangular patch is within 100Ω and 400Ω and it can be approximated as

$$Z_a \approx 90 \frac{\epsilon_r^2}{\epsilon_{r-1}} \left(\frac{L}{W}\right)^2 (\Omega) \quad (29)$$

In our design the impedance at the edge of rectangular patch is $Z_a \cong 273.5 \Omega$. So if we want to match 100Ω transmission line to the edge of rectangular patch, we have to use equation (21) to compute quarter-wavelength transformer impedance. So $Z_T = \sqrt{100 * 273.5} \cong 165.375 \Omega$.

Now we just need to compute width of transformer. For this purpose I used tool of Advanced Design system called LineCalc and I got $W_T = 0.01 \text{ mm}$. Since fabrication limitation of microstrip's width is 0.1 mm this solution is useless.

So let's design solution of inset feeding line into antenna which is seen in Figure 11, where g is gap.

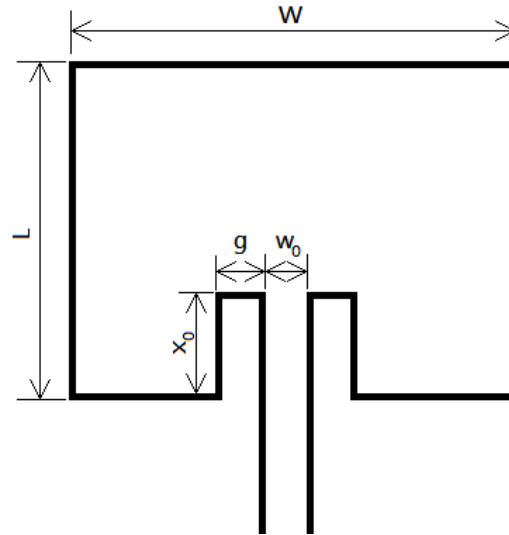


Figure 11 Patch antenna with inner feeding line configuration

The first dimension to compute is inset length x_0 . The equation for inset length is defined as

$$x_0 = \frac{L}{\pi} * \cos^{-1} \sqrt{\frac{Z_0}{Z_T}} \quad (30)$$

where $Z_0 = 100 \Omega$. In our solution, the inset length is $x_0 \cong 0.68 \text{ mm}$. About length of gap it is more difficult because nobody want to publish information how to design proper length of gap so gap will be designed by trying values in designing software.

Also transmission line need to be set. For 100Ω transmission line it is $w_0 = w_{100} = 0.12 \text{ mm}$ which is on the edge of fabrication limit. This value I computed with LineCalc.

III.III.II. Designing single patch antenna by software

Our laboratory use very popular software Ansoft HFSS for designing of antenna. Version in my computer is 13.0 which is quite enough to design patch antenna.

After simulations I needed to change some values to get better results. The Table 2 shows final values for designing of single patch. The final configuration is shown in Figure 12.

Table 2 Final dimensions of designed single patch antenna with 100Ω

W (mm)	L (mm)	x_0 (mm)	g (mm)	w_{100} (mm)
4.05	3.15	0.688	0.36	0.122

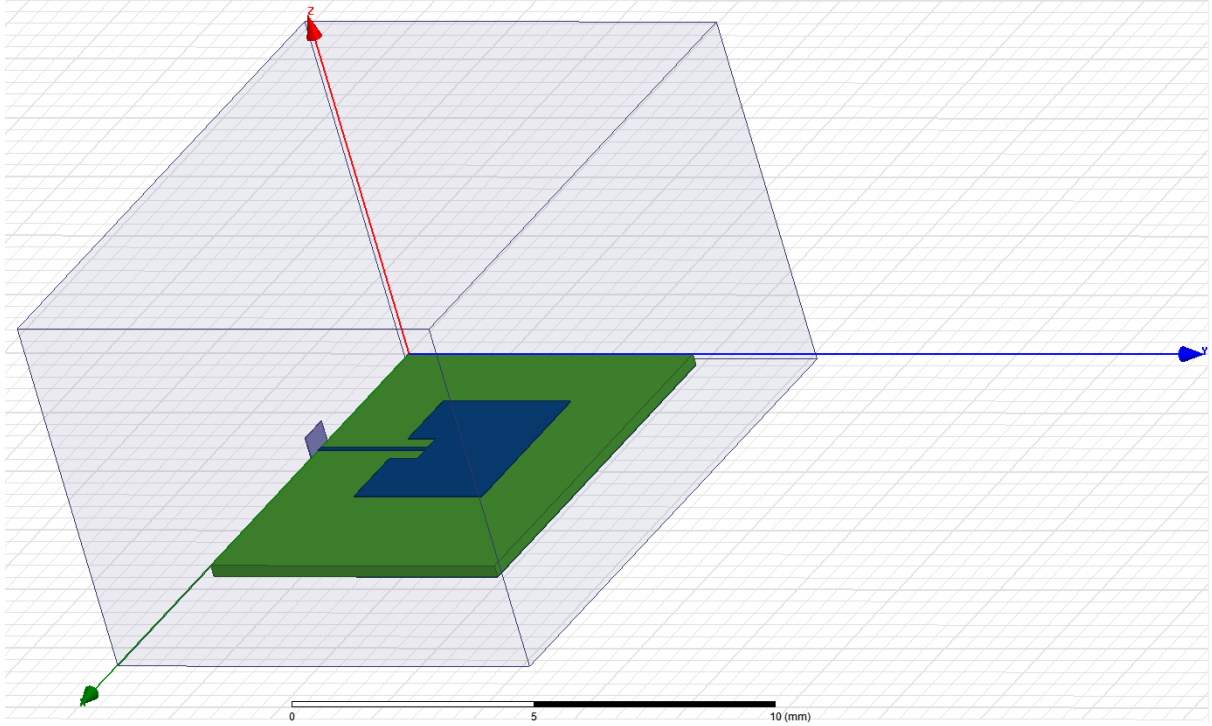


Figure 12 Single patch with 100 Ω inner feeding line configuration design

The length of substrate, I've choosed, is 7 mm and width is 9 mm which is enough to get good simulation. Port for measuring S_{11} parameter and for making excitation of electromagnetic wave is $6 * w_{100}$ length in X and Z axis, and t length in Y axis. Port is defined as perfect electrical conductor PEC and touches ground plane by edge. The transparent block is boundary of radiation in air and it has dimensions as in X axis is about W longer than substrate, in Y axis is L longer than substrate and in Z axis is $2 * W$ tall. For X and Y axis it is centered and for Z axis the lower part is $5 * h$.

The important parameter to be measured is parameter S_{11} . S parameters are defined as

$$S_{ab} = 20 * \log \frac{V_b^-}{V_a^+} \text{ (dB)} \quad (31)$$

where V_a^+ is voltage of incident wave in port a and V_b^- is voltage of output wave in port b. In our case we have only one port so it is

$$S_{11} = 20 * \log \frac{V_1^-}{V_1^+} \text{ (dB)} \quad (32)$$

which says how high voltage compared to input voltage has been reflected, in decibels.

Frequency characteristic of S11 parameter of our single patch antenna is shown in Figure 13.

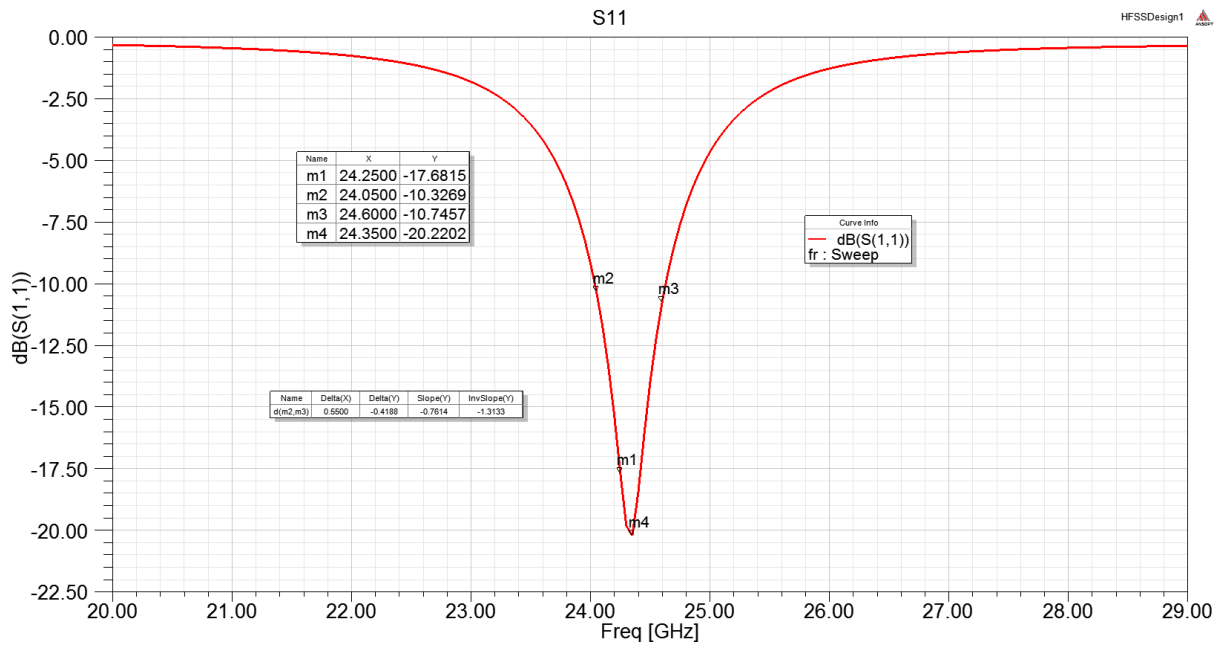


Figure 13 S11 parameter of designed single patch antenna in rectangular plot

The result seems very good. The bandwidth of $S_{11} < 10$ dB is from 24.05 GHz to 24.6 GHz, so this solution seems very useful for our purpose. But we also need to check gain and beamwidth.

To see total gain in decibels and beamwidth we should plot gain in radiation pattern in H-plane and E-plane. E-plane contain electric field vector. It means that direction of propagation of electromagnetic wave in antenna define E-plane. According Figure 12 it is Y-Z plane. H-plane contain magnetic field vector and magnetic field vector is perpendicular to electric field vector. So according Figure 12 it is X-Z plane.

Radiation pattern of designed single antenna in E-plane is shown in Figure 14. In this plane we can see that antenna has gain about $G_{max} \cong 6.94$ dBi and $HPBW \cong 90^\circ$.

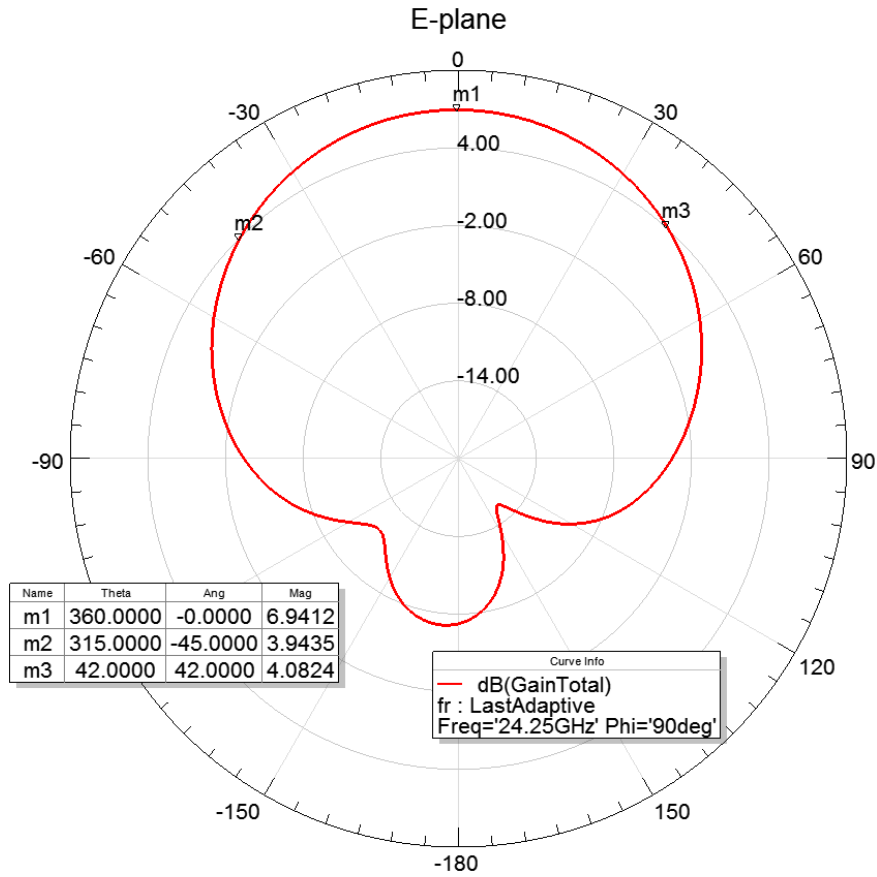


Figure 14 Radiation pattern of designed single patch antenna in E-plane

Radiation pattern of designed single antenna in H-plane is shown in Figure 15. In this plane we can see that antenna has gain about $G_{max} \cong 6.94$ dBi and HPBW $\cong 72^\circ$. For 1.5 m we need gain more than 10 dBi and half-power beamwidth much more narrow so the so we should design at least 2x2 antenna.

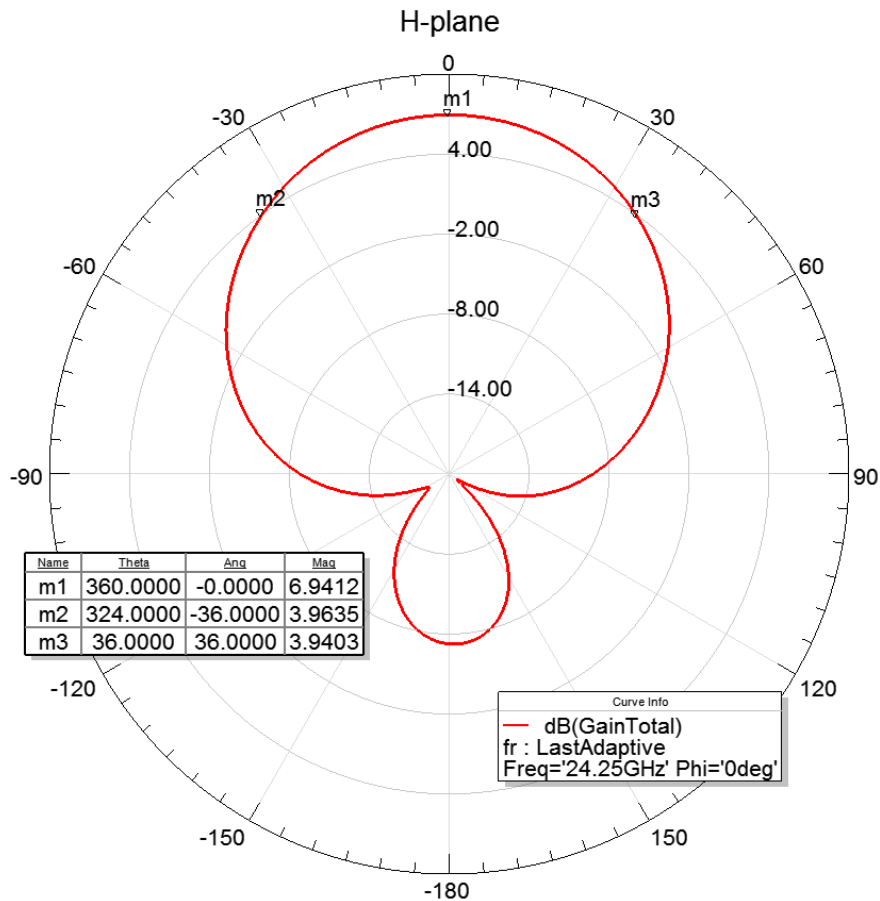


Figure 15 Radiation pattern of designed single patch antenna in H-plane

III.III.III. Designing 2x2 patch antenna array by software

If we want symmetric construction we need to divide power to four the same paths. The construction of this dividing is shown in Figure 16.

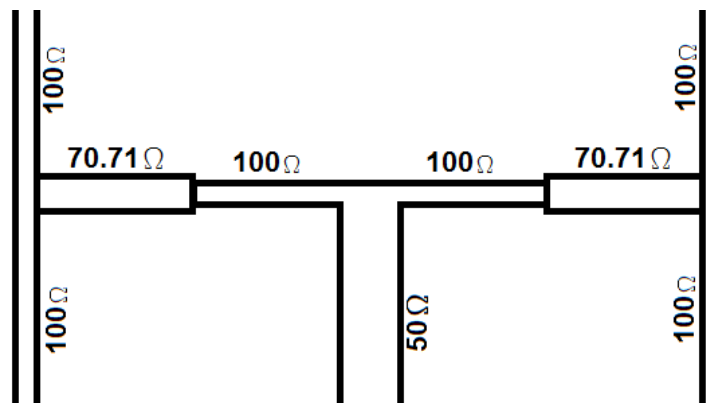


Figure 16 Dividing power of electromagnetic wave to four paths

The characteristic impedance of circuit is $Z_0 = 50 \Omega$ and it is also input transmission line. Power in this path is divided by two with two parallel 100Ω quarter-wavelength transformers. Since

$100 * 100 / (100 + 100) = 50 \Omega$ the condition of impedance matching is satisfied. Now we need each of these two path divide by two again. As we know from previous chapter, the 100Ω transmission line is on the edge of fabrication limitation so we cannot use two 200Ω transmission lines because it would be even narrower and not possible to fabricate. But we can use two parallel 100Ω transmission lines again, which together is 50Ω and use quarter-wavelength impedance transformer between. So by using equation (21) we can compute impedance of transformer $Z_r = \sqrt{100 * 50} = \sqrt{2} * 50 \cong 70.71 \Omega$.

Now it is seen why patch antenna with 100Ω feeding line is used in this solution. If we choose 50Ω feeding line, we need more impedance transformers and loose more energy on discontinuities and also whole structure would be much more complicated.

Now we need values of width and length of these transmission lines. All were computed by LineCalc. For transmission line with characteristic impedance 50Ω we got width $w_{50} = 0.54 \text{ mm}$. For the quarter-wavelength resonators with impedance 70.71Ω we got width $w_{70} = 0.285 \text{ mm}$ and length $l_{70} = 1.88 \text{ mm}$. For 100Ω quarter wavelength transformer we got length $l_{100} = 1.95 \text{ mm}$ and width $w_{100} = 0.12 \text{ mm}$ is still the same.

The final design if 2×2 patch antenna array is shown in Figure 17. Some values needed to be changed to get better results. The list of values is in Table 3.

Table 3 Final dimensions of designed 2×2 patch antenna array with 100Ω

W (mm)	L (mm)	x_0 (mm)	g (mm)	w_{100} (mm)	l_{100} (mm)	w_{70} (mm)	l_{70} (mm)	w_{50} (mm)
4.07	3.17	0.688	0.36	0.122	1.98	0.278	1.92	0.56

As it is seen, the W and L a little bit changed. And also dimensions of transmission lines. Also I need to note that length of 100Ω transmission lines between power divider and knee beside antenna is the same as width of antennas W, and length of 100Ω feeding line is $l_{100}/2 + x_0$. The knees are quarters of rings with the same width as transmission lines so the impedance of knees is the same.

Port dimensions are $4 * w_{50}$ in Y dimension, $3 * w_{50}$ in Z dimension and t in X dimension. Configuration is the same as in previous chapter. Center of antenna array is in 8.07 mm in

X dimension from Y axis and 5.21 mm in Y dimension from X axis. The dimensions radiation boundary of air are $4 * W + \text{width of substrate}$ for X axis, $4 * L + \text{length of substrate}$ for Y axis and $7 * W$ for Z axis. The boundary is centered in X-Y plane to antenna array and the lowest boundary is $10 * h$ below the X-Y plane in Z dimension.

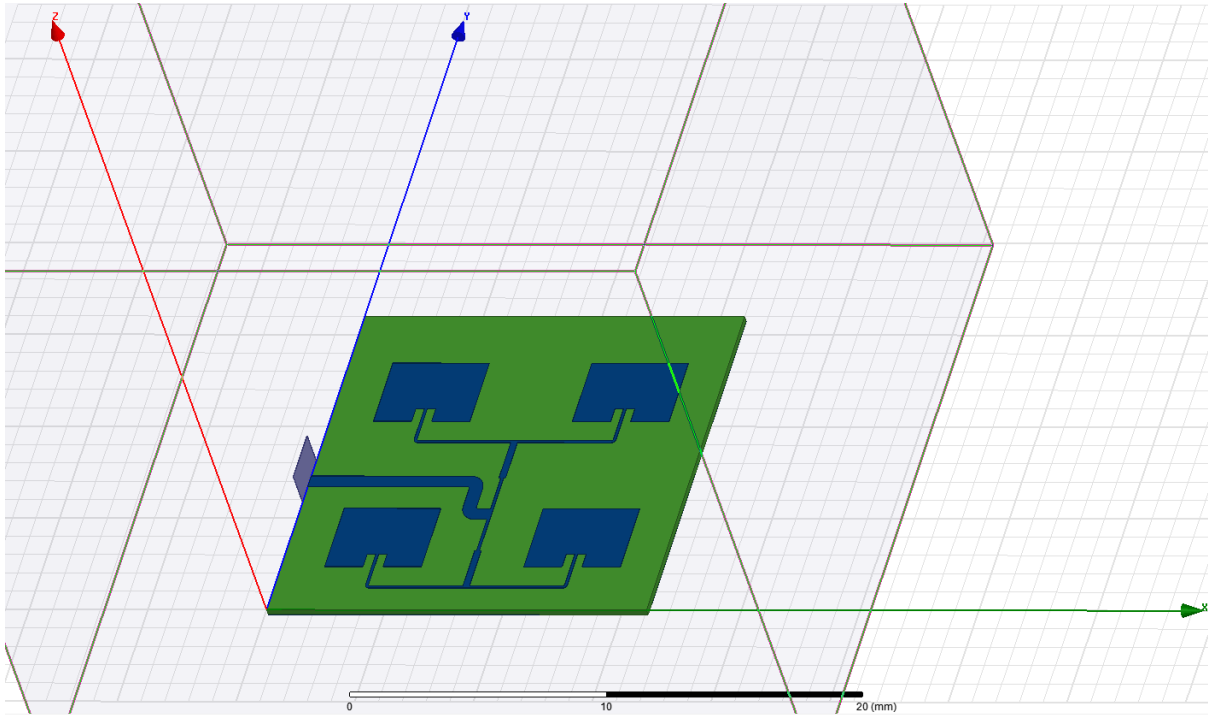


Figure 17 2x2 patch antenna array with 100Ω inner feeding line configuration design

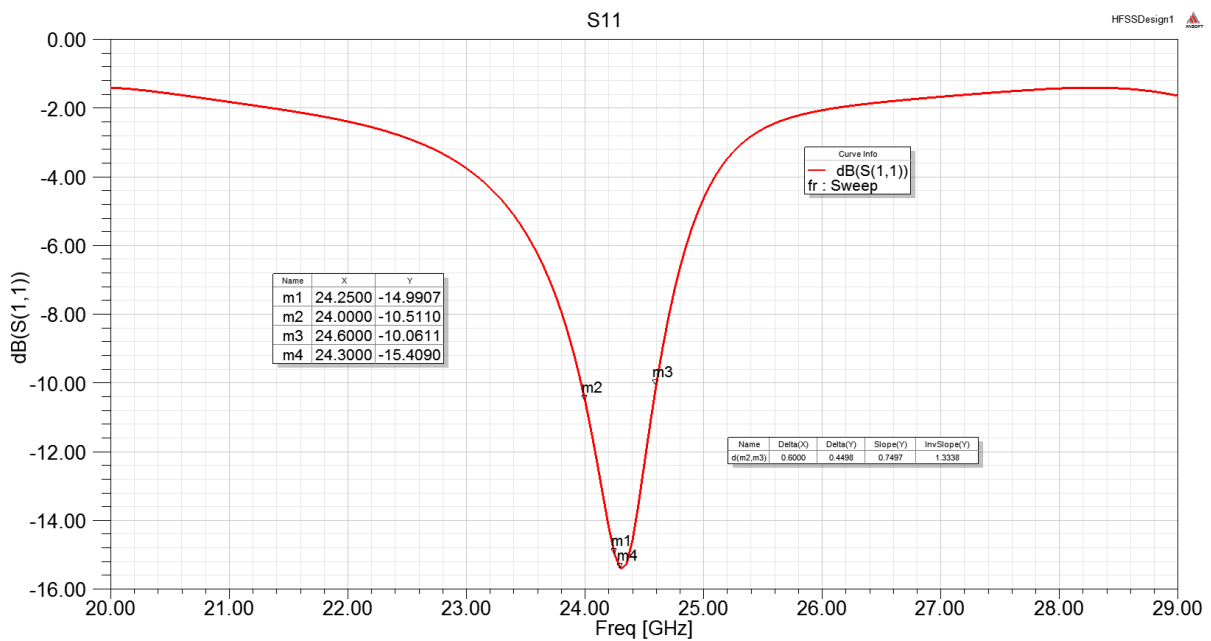


Figure 18 S11 parameter of designed 2x2 patch antenna array in rectangular plot

If we plot S11 parameter to Figure 18 and compare with S11 parameter of single patch antenna in Figure 13, we can see some differences. The bandwidth of $S_{11} < -10$ dB is within 24 GHz and 24.6 GHz, which is wider than for single patch antenna case. The center frequency is 24.3 GHz and value of S11 in center frequency is -15.4 dB which is a little bit worse than single antenna. But we suppose much better gain and half-power beamwidth.

The radiation pattern of designed 2x2 patch antenna array is shown in Figure 19. The achieved maximum gain is 12.4 dB and $HPBW \cong 39^\circ$ which is much better than for single antenna as shown in Figure 14.

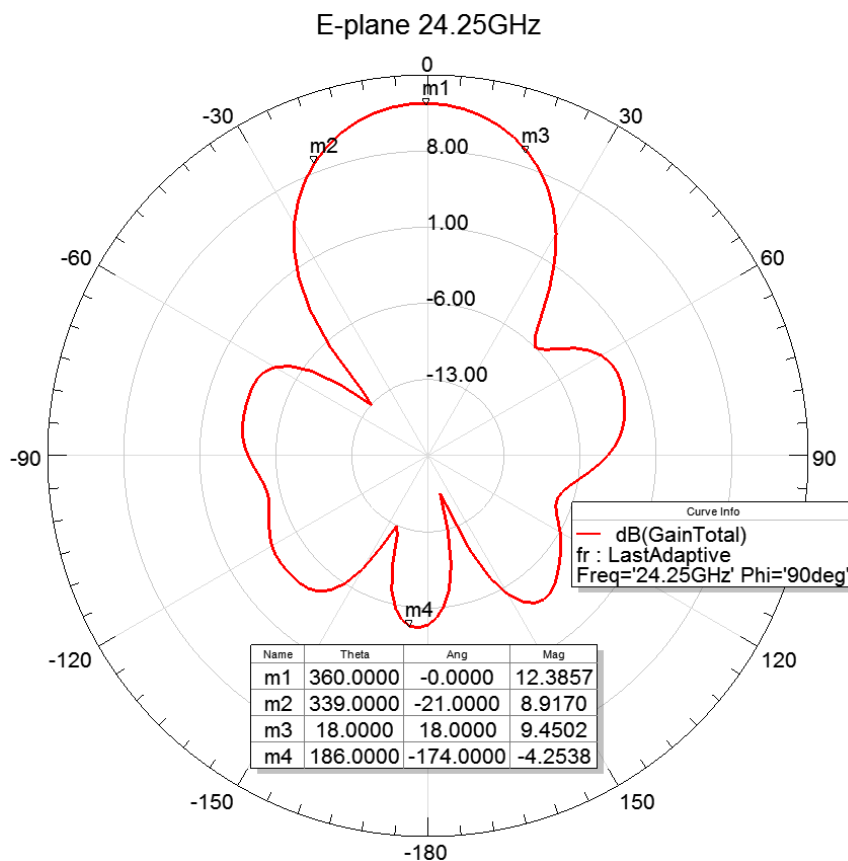


Figure 19 Radiation pattern of designed 2x2 patch antenna array in E-plane

Let's look on the radiation pattern in H-plane shown in the Figure 20. Maximum gain is 12.4 dB and $HPBW \cong 39^\circ$.

These results are quite good but we need to keep in mind that HFSS suppose very high efficiency. So maybe real properties of this antenna could be worse. To make sure, that gain is high enough I decided to design 4x4 patch antenna array

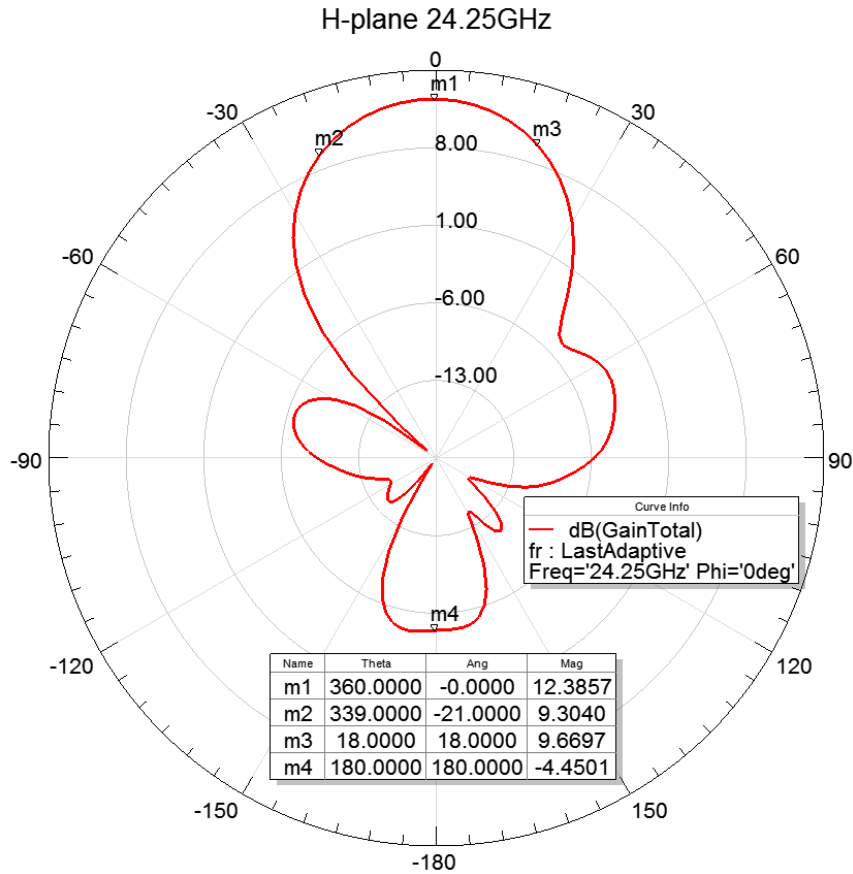


Figure 20 Radiation pattern of designed 2x2 patch antenna array in H-plane

III.III.IV. Designing 4x4 patch antenna array by software

To make this configuration I just copy and flip 2x2 antenna array and inserted more power dividers and transmission lines. The concept and principle is the same so I don't need to explain it.

The final design of 4x4 patch antenna array is shown in Figure 21. Also here some values has been changed.

Table 4 Final dimensions of designed 4x4 patch antenna array with 100 Ω

W (mm)	L (mm)	x_0 (mm)	g (mm)	w_{100} (mm)	l_{100} (mm)	w_{70} (mm)	l_{70} (mm)	w_{50} (mm)
4	3.1	0.68	0.36	0.12	1.9	0.28	1.8	0.54

Width and length of substrate are both $W_{sub} = L_{sub} = 32 \text{ mm}$. The input 50Ω transmission line is $W_{sub}/2 - w_{100}/2$ long. This power is divided to two paths where 50Ω transmission line is 4.24 mm long. The rest of designed dimension are the same as in previous chapter. But radiation boundary of air is $4 * W + W_{sub}$ long in X direction, $4 * L + L_{sub}$ long in Y direction and $10 * W$

tall in Z direction and the lowest boundary is $10 * h$ below the X-Y plane in Z dimension.

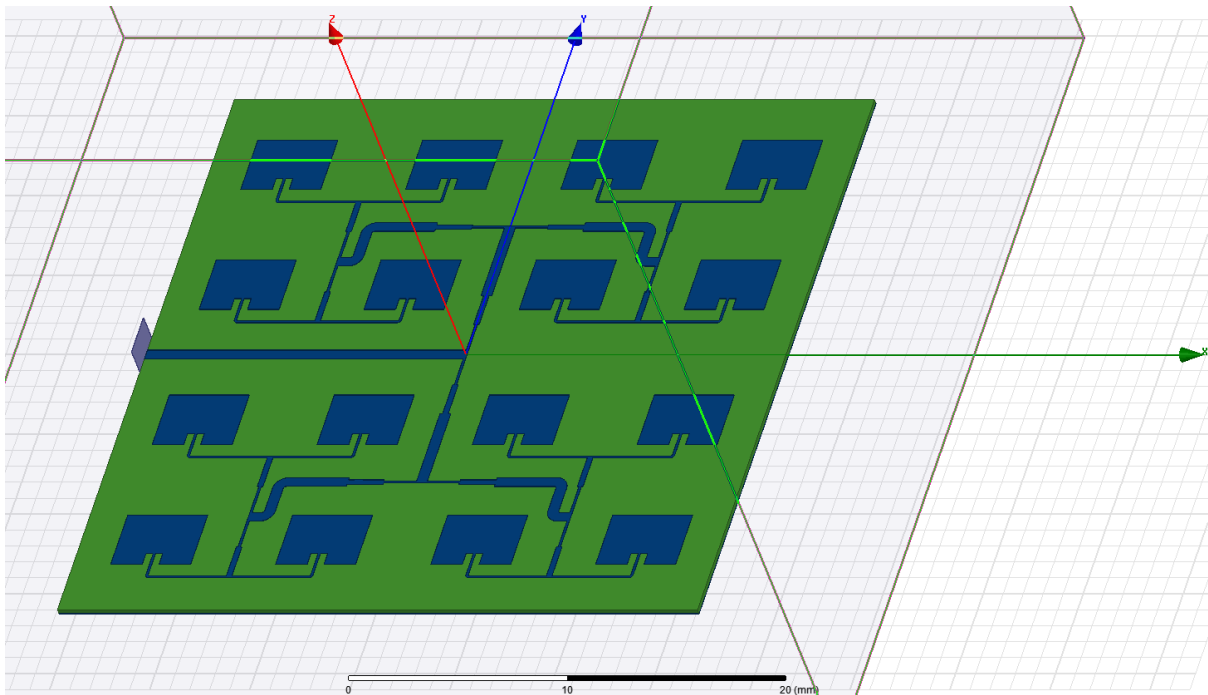


Figure 21 4x4 patch antenna array with 100Ω inner feeding line configuration design

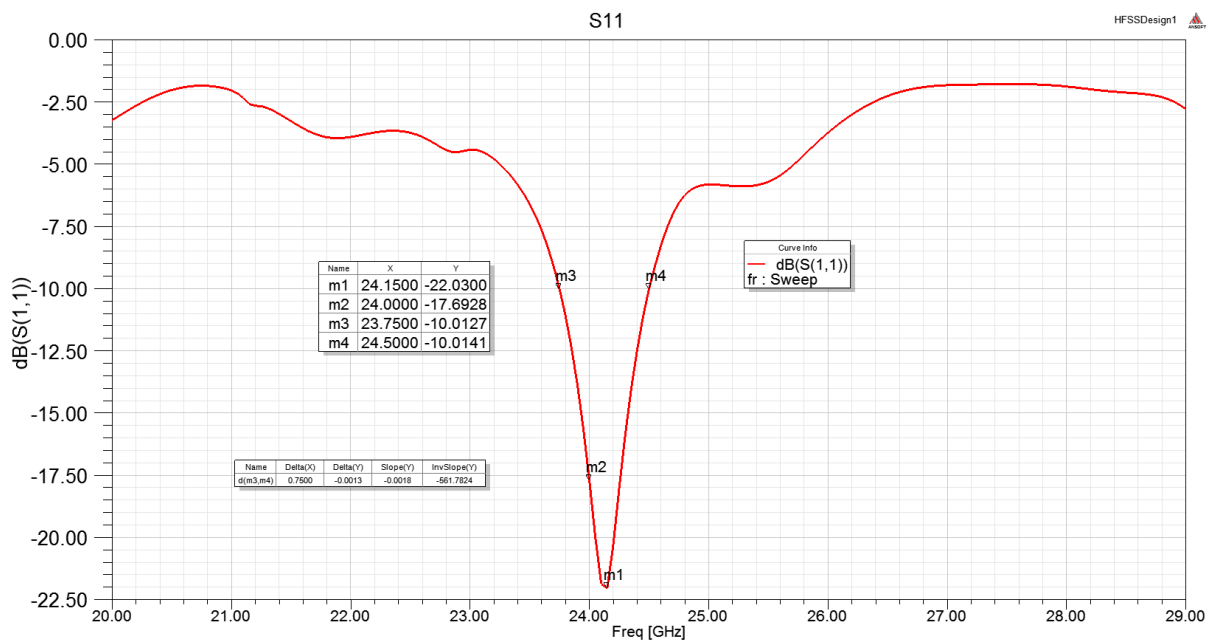


Figure 22 S11 parameter of designed 4x4 patch antenna array in rectangular plot

The Figure 22 shows the S11 parameter of designed 4x4 patch antenna array. If we compare it with S11 parameters of 2x2 patch antenna array shown in Figure 18, we can see that the resonant frequency also a little bit shifted to $f_r = 24.15 \text{ GHz}$ and S11 parameter on resonant frequency is

-22 dB which is much better than previous solution. The bandwidth of $S_{11} < 10$ dB is within 23.75 GHz and 24.5 GHz, which is still very good for our purpose.

The radiation pattern of designed 4x4 antenna array in E-plane shown in Figure 23 says, that maximum achieved gain is $G_{max} = 16.66$ dBi and $HPBW \cong 20^\circ$. We can also see the side lobe which has maximum gain $G_{SL} = 6.61$ dBi. To compare with G_{max} it is neglectable but we should take care that nothing in this angle from antenna cannot be close to antenna because we could get wrong signal.

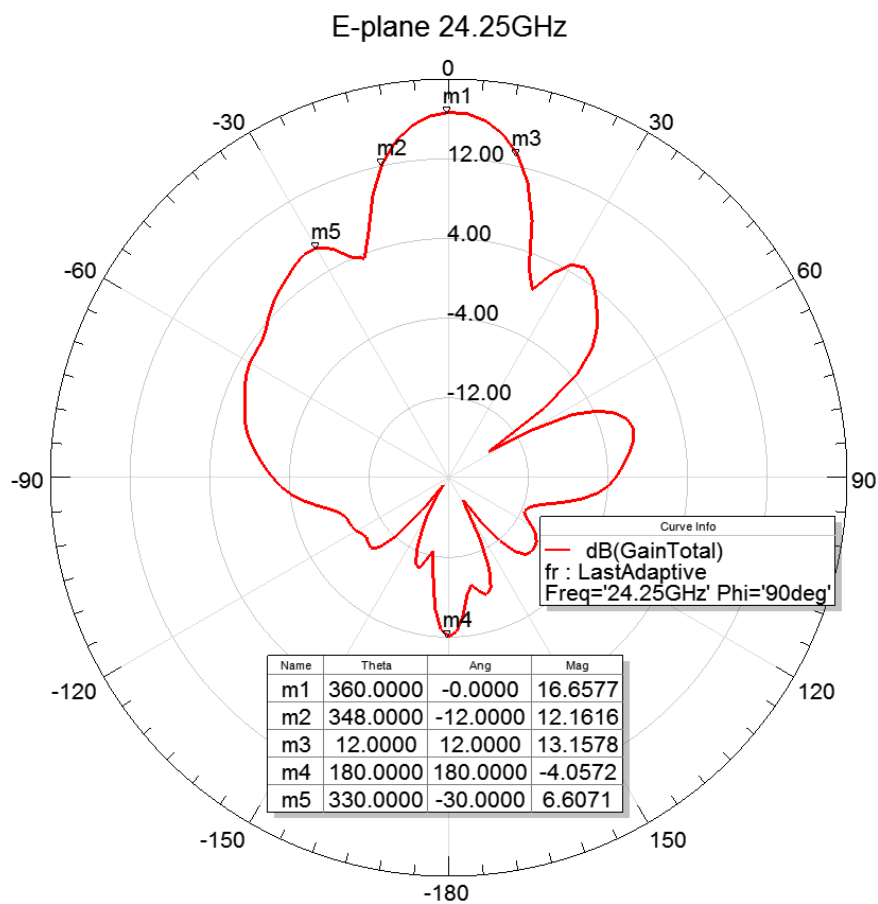


Figure 23 Radiation pattern of designed 4x4 patch antenna array in E-plane

The radiation pattern of designed 4x4 antenna array in H-plane shown in Figure 24 says, that maximum achieved gain is $G_{max} = 16.66$ dBi and $HPBW \cong 20^\circ$. These results are quite perfect for our purpose. Half-power beamwidth for 20° is very good. It means that even for distance 1.5 m the antenna can be focused to chest and only some small area around.

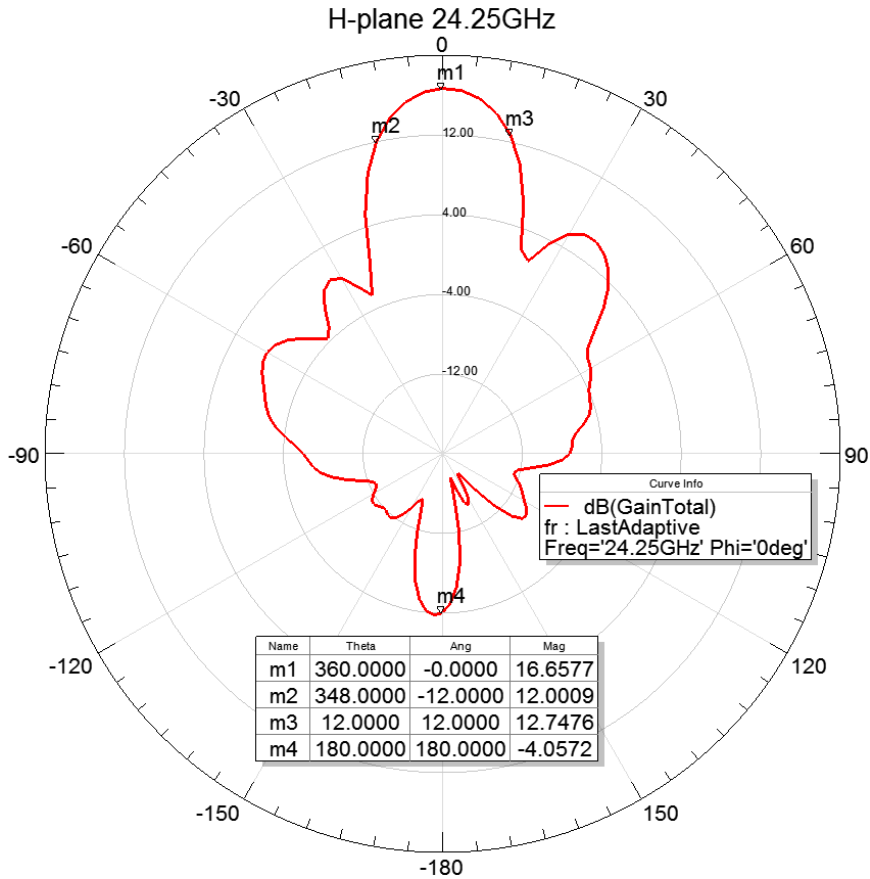


Figure 24 Radiation pattern of designed 4x4 patch antenna array in H-plane

III.III.V. Implementation into board

The design in HFSS (only antenna without substrate and ground plane) has been saved as *.gds file, opened in Advanced Design System and saved again, because it converts format to proper way. Then we opened that file in Altium Designer to implement antenna design to real board. Next configuration is for antenna measurement. Because of calibrations finished in laboratory long time ago I added 20 mm transmission line and put two vias for connector.

Connector for measurement is designed by brand called Southwest Microwave and it is 2.4mm female and connector has been launched with number 1492-02A6. The dimensions needed by this connector are provided in technical documentation [5] and are shown in Figure 25. The centers of via holes has to be 9.53 mm from each other and 2.79 mm from the edge of the board. The diameter of via holes should be 1.98 mm.

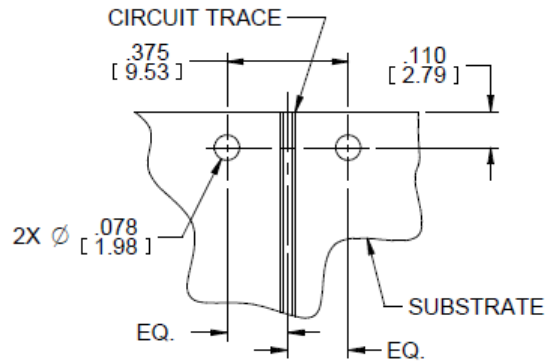


Figure 25 Dimensions and distances in mm of vias for connector [5]

The final layout of antenna is shown in Figure 26 where red and grey regions mean copper, brown regions mean hole and the rest of colors mean dielectric. The ground plane is covered by dielectric parts. The photo of manufactured antenna with connector is shown in Figure 27.

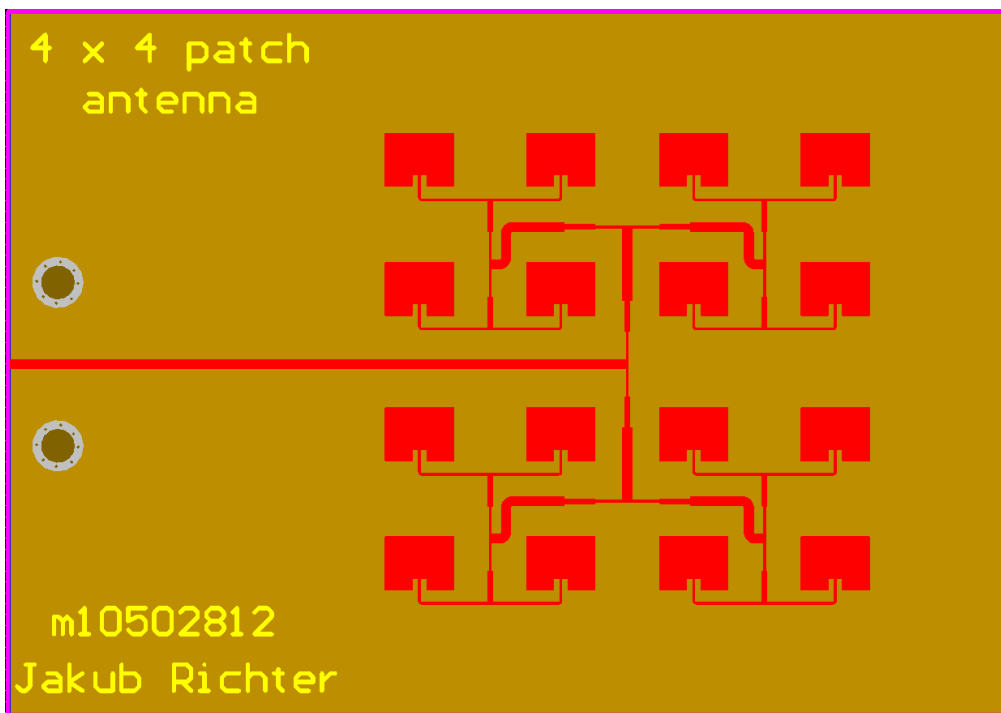


Figure 26 Designed 4x4 patch antenna array layout for measurement

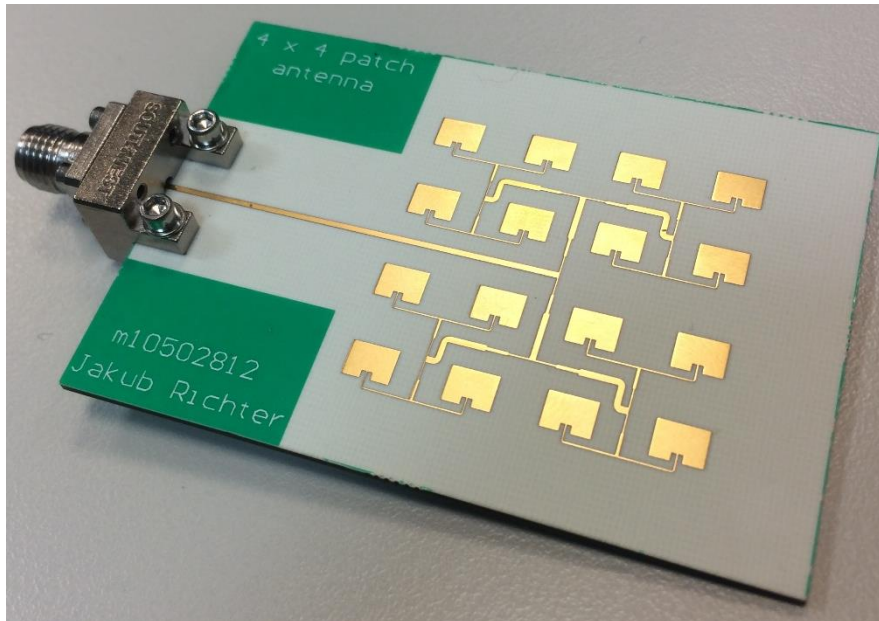


Figure 27 Photo of designed 4x4 antenna array with connector

III.IV. Antenna measurements

There are some chambers in the school but only one set up for measurement of 24 GHz antennas. This chamber is made for measuring near-field but computer which is collecting measurement contain transformation computations to far field. This thesis is not focused only to designing and measurement of antenna so I will skip theory about near-field to far-field transformation. But whole system of chamber was designed by brand NSI-MI.

First I would like to show measured S11 parameter and compare it with simulated values. The Figure 28 shows this comparison. Because all equipment in chamber is made for measuring very narrow bandwidth, we cannot see wide frequency characteristics. The results are quite different than we expected. Blue line shows simulated values and beige line shows measured values. There are two local minimums instead of one. In 24 GHz the S11 is high but still useful.

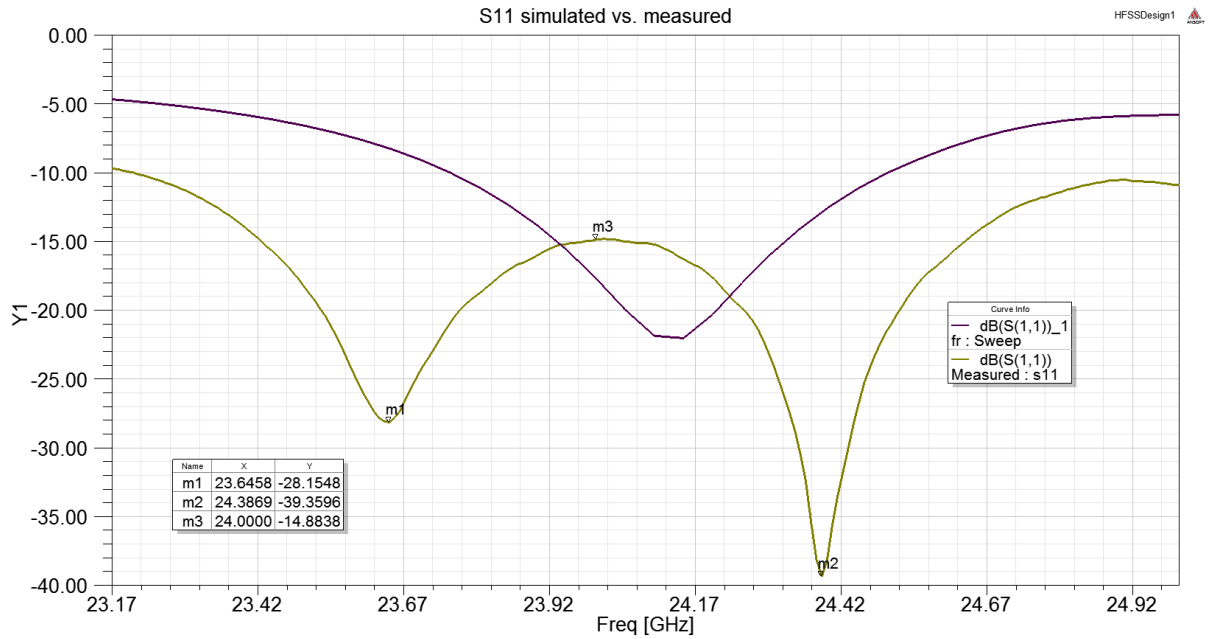


Figure 28 Measured vs. simulated S11 parameter of designed 4x4 patch antenna array in rectangular plot

For measurement of near-field we should know, where is **boundary of near-field**. According equation (12) we can compute, that if wavelength with frequency 24 GHz is $\lambda = c/f = 12.5 \text{ mm}$ and diameter of aperture is $D \cong 4 \text{ cm}$, the Fraunhofer distance is about $d_F = \frac{2 \cdot D^2}{\lambda} \cong 25.6 \text{ cm}$.

The configuration of measured antenna and reference antenna is shown in Figure 29, where green board is measured antenna and grey stick is reference antenna. According Fraunhofer distance we can say that we will measure near-field radiation pattern.



Figure 29 Near-field measurement configuration

Measurement of near-field principle is described in Figure 30. The grey rectangular represents reference antenna. The reference antenna is following arrow in its movement and measure received power.

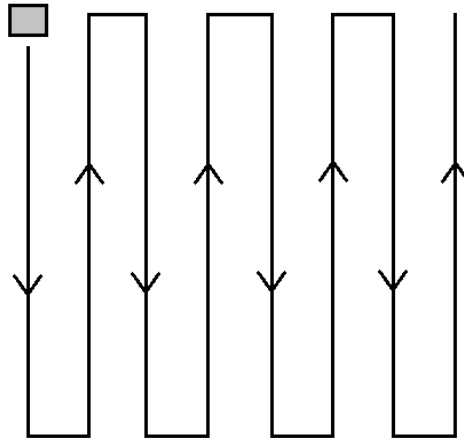


Figure 30 Near-field radiation pattern measurement principle.

Near-field radiation pattern has been measured and recomputed to far-field radiation pattern with software provided by NSI-MI. I plotted data for H-plane radiation pattern of 24.2 GHz measurement in Microsoft Excel because I am using only R2015b version of Matlab and **polaraxes** function is introduced from version R2016a. The results are shown in Figure 31.

The gain is much lower than simulation. It reaches value $G_{max} = 11.5$ dBi. It means that efficiency is very low. But half-power beamwidth seems still good. The reason of so low gain can be caused by many aspects. First aspect is that 100Ω transmission line is very thin and fabrication cannot be as accurate enough. So the impedance of transmission lines can be much different than 100Ω . Another aspect are possible errors in measurement. But anyway, this solution works and our time is limited to make new one.

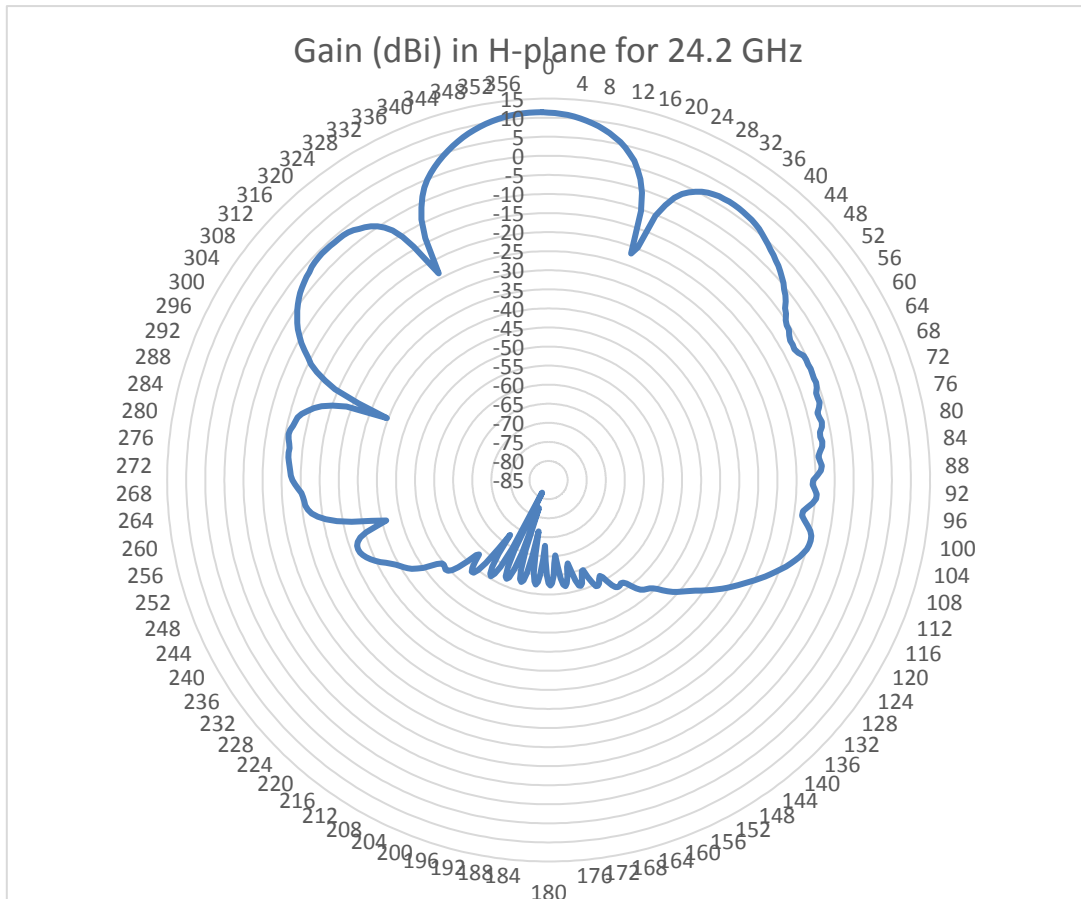


Figure 31 Measured radiation pattern for designed 4x4 patch antenna array in H-plane

IV. Construction of circuit

IV.I. Basic arrangement

Now when we have designed antenna we have to design whole circuit to control transmitting and receiving of electromagnetic wave and prepare result for signal processing. As we know the core of our radar is chip BGT24MTR11. So first I will explain some parts of RF chip and then I will explain block diagram of whole circuit.

IV.I.I. Low-noise amplifier (LNA)

Low-noise amplifier is high frequency amplifier which is mainly focused to amplify signal with as less noise as possible. If designer want to get good results, the solution is usually compromise between the lowest noise and highest amplification of input signal. The basic component of LNA is transistor, which is generally high-electron-mobility transistor (HEMT).

IV.I.II. In-phase/Quadrature mixer

IQ-mixer is necessary component for radar which show the differences between frequency of received electromagnetic wave and frequency of local oscillator. The basic principle of IQ-mixer is shown in Figure 32 IQ-mixer principle [6].

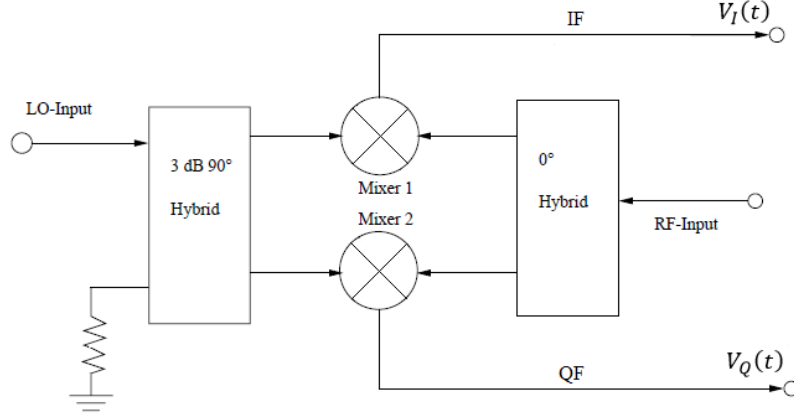


Figure 32 IQ-mixer principle [6]

The signal from local oscillator is divided to two path where one path carry 90° delayed signal. This signal is mixed with received signal and output is low-frequency signal. In-phase and quadrature signals are represented as complex number, where in-phase signal is real part and quadrature signal is imaginary part. These two signals are represented by following equations obtained from [6]

$$\text{Mixer 1} \quad V_I(t) = a_L(t) * a_R(t) * \cos(\omega_L * t) * \cos(\omega_R * t + \alpha) \quad (33)$$

$$\text{Mixer 2} \quad V_Q(t) = a_L(t) * a_R(t) * \sin(\omega_L * t) * \cos(\omega_R * t + \alpha) \quad (34)$$

where $V_I(t)$ is output voltage of in-phase signal, $V_Q(t)$ is output voltage of quadrature signal, $a_L(t)$ is amplitude of oscillator, $a_R(t)$ is amplitude of received signal, ω_L is angular frequency of local oscillator, ω_R is angular frequency of receiver and α is delay between local oscillator and received signal.

IV.I.III. RF chip BGT24MTR11

The principle block diagram of BGT24MTR11 chip is shown in Figure 33. According the datasheet [3], the chip has its own oscillator controlled by voltage in pin 4 called FINE and transmission and receiving pins. The chip also include low-noise power amplifier and I/Q-mixer. So

the output of BGT24MTR11 chip is in-phase and quadrature low frequency signal. As we can see in Figure 33, the I/Q-mixer mix two more signals. These signals are the same signals but with opposite phase. So for example output in-phase signal is two signals. In-phase signal and signal with opposite phase than in-phase signal. This is way how to get higher amplification in the next process. PA means Power Amplifier and LO is Local Oscillator. BGT24MTR11 start working after it receive proper command by Serial Pheripheral Interface (SPI) bus which will be described later.

Many pins and signals are not used in our solution like temperature sensor pins, COARSE, TXOFF, ANA. Q2, Q1 and Q1N are frequency prescalers. Q1 and Q1N are prescaler outputs for 1.5 GHz so we will keep this output on by 50Ω load with capacitor. Q2 is prescaler output for 23 kHz so we don't need to load it. Output LO is also loaded to 50Ω .

RF output is divided to two outputs. TX is in-phase and TXX is in opposite phase. To get higher power we will combine these two outputs at the same phase.

As I described above we need to tune frequency by voltage in the pin FINE. As it is shown in Appendix I used voltage divider by resistors $R18 = 10 k\Omega$ and $R19 = 910 \Omega$. We want to tune frequency to $f_r \cong 24.2 GHz$. According measurements which made member of our laboratory in the past $V_{FINE} \cong 0.4 V$ would tune RF chip to this frequency. If $V_{CC5} = 5 V$, we can say that $V_{FINE} = V_{CC5} * R19/R18 * R19 \cong 0.41 V$.

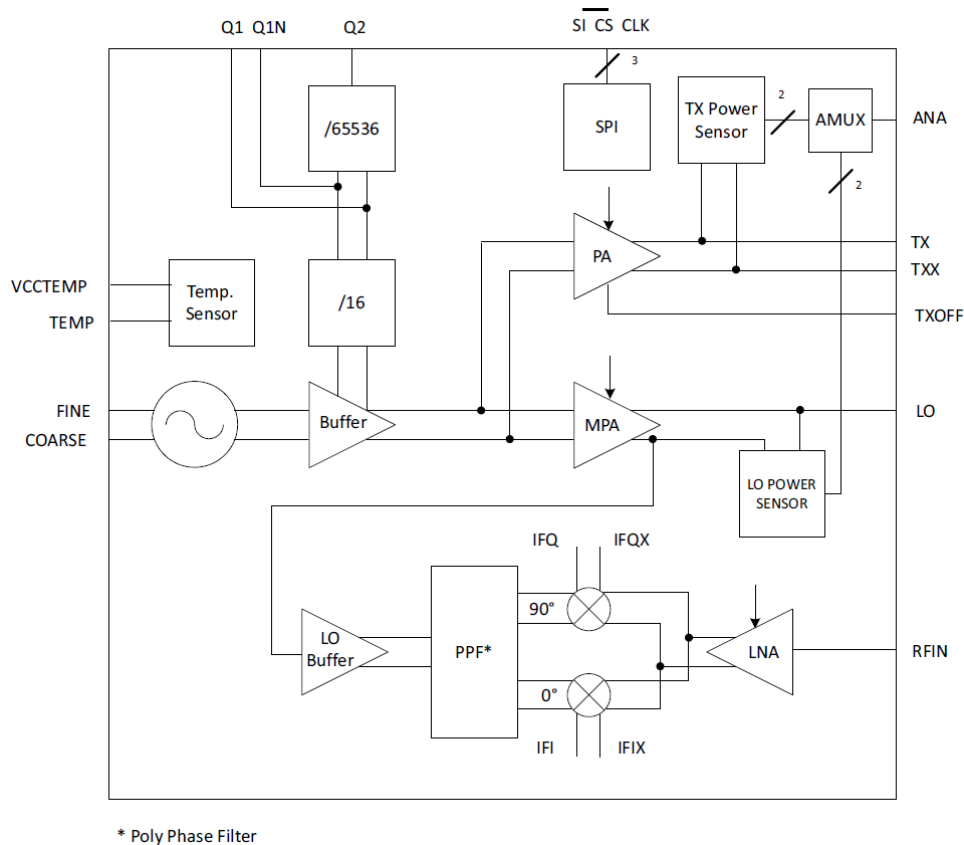


Figure 33 BGT24MTR11 Block Diagram

IV.I.IV. Block diagram of main part of circuit

The block diagram of main part of whole circuit is shown in Figure 34. PC set up SPI command for Arduino to send the command to RF chip. RF chip start transmitting and receiving data and I/Q-signals are sent to operation amplifier because I/Q-signals are too weak to be recognized by analog-digital converter in Arduino UNO. Because the I/Q-signals contain also signals with opposite phase so the operation amplifier get four signals as inputs. The output of operation amplifier are only two signals which carry information about in-phase and quadrature signals. These two signals are sent to analog input of Arduino. By sending command from PC to collect data from analog inputs of Arduino, the Arduino start collecting data and send them through port COMX to PC. These data are stored for later signal processing.

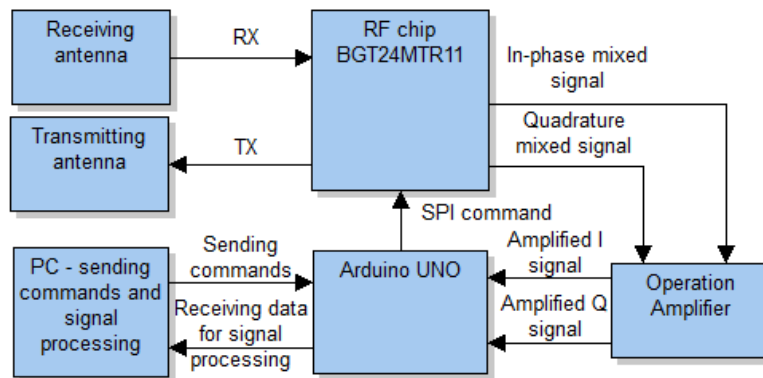


Figure 34 Block diagram of main part of whole circuit

IV.I.V. Arduino UNO

Arduino is Italian company which is developing kits prepared to connect to periphery like sensors, other chips, displays or many components and also provide software with simplified programming. Arduino UNO is very popular and simple board which has two Atmega chips, its own oscillator and communication between PC and board is with USB but it can also be powered by adapter. Chip Atmega16U2 is made for Universal Synchronous/Asynchronous Receiver and Transmitter (UART) communication with PC and chip Atmega328p. The chip Atmega328p is made for communication with periphery and obtaining data. Arduino UNO can run SPI communication and contain 6 analog input. That's what we need.

IV.I.VI. Differential amplifier

Because the I/Q-signals are quite weak and they are two pairs and we need to amplify them, the operational amplifier (OA) is the best option.

Operational amplifier is component composed of transistors which has two differential inputs and one output. The amplification can be very high but output voltage cannot exceed supply voltage. Otherwise the output signal is in saturation and information is not clear. In our solution we use LM358 from company STMicroelectronics founded in [8] which contain two operational amplifier as we need for in-phase and quadrature signals.

There are many types of applications with operational amplifier. Because each both in-phase and quadrature components are consists of two signals (one signal is in opposite phase) the differential

amplifier is suitable option and final configuration is shown in Figure 35.

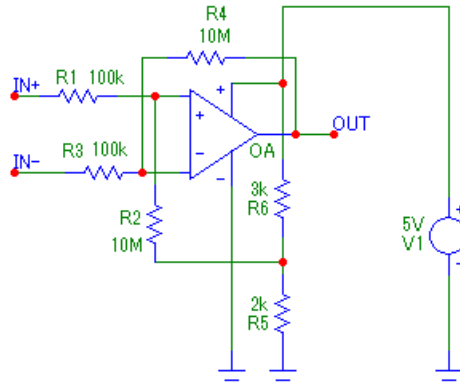


Figure 35 Differential amplifier with shifted reference voltage

The main signal comes on wire IN+ and signal with opposite phase comes on wire IN-. The resistor R4 is connected between invert input of OA and output of OA. The directions of currents in resistors R3 and R4 are both toward invert input of OA as one current. This current appear in non-invert input and continues to resistor R2 with current from resistor R1 and thus toward reference voltage in R5.

Thus if reference voltage in R5 is $V_5 = 0V$, $R_1 = R_3$ and $R_2 = R_4$, we can say that

$$V_{out} = -\frac{R_2}{R_1} * (V_{IN-} - V_{IN+}) \quad (35)$$

In our case, amplification given by resistors is $-R_2/R_1 = -100$. Because in-phase and quadrature signal contain negative values and analog input of Arduino UNO can accept only positive values, we just use voltage reference to shift virtual ground. In our case it is $R_6/R_5 * 5V = 2V$. So the output voltage will be also lifted up by 2V.

IV.I.VII. Supply voltage

Whole circuit should be supplied by its own voltage source (not from Arduino), because if we use voltage from Arduino board, there will be less power left. So I decided to use microUSB connector so the whole circuit can be supplied by PC or by power bank.

To make sure that if voltage provided by power bank will drop and we want to keep our voltage as $V_{CC5} = 5V$, we use Voltage charge pump in regulator SC632 provided from brand SEMTECH. According datasheet [7] the input voltage can drop to $V_{CC} = 2.9V$ and output voltage will be stip

kept for $V_{CC5} = 5\text{ V}$ and it has also short circuit, over-voltage, and over-temperature protection. The circuit arrangement is fixed and provided by datasheet [7]. It is also shown in whole schematic in Appendix.

According datasheet [3], the RF chip BGT24MTR11 can be supplied by $V_{CC3} = 3.3\text{ V}$. For this purpose LM6206N3 voltage regulator was used. We can suppose 250 mA output current as described in datasheet [8].

The ground of DC part and ground of high frequency part of circuit are connected by coil with $L = 100\text{ nH}$. It is because we need to avoid some ripple caused by high frequency part. The coil itself has some DC resistance which is $\cong 4\ \Omega$ so we should suppose that V_{CC5} is a little bit less than 5 V from view of high frequency part of circuit.

IV.I.VIII. SPI high level consideration

According datasheet [3] the SPI high voltage level is recognizable in range of $0.2\text{ V} \div V_{CC3} = 3.3\text{ V}$. The SPI high voltage level is 5 V . To consider that we can use another board with high level voltage $3 - 3.3\text{ V}$ I did not set voltage divider to the circuit of radar but as separated board. The basic concept is very easy and so for CLK, SS and MOSI I used voltage divider with two $2\text{ k}\Omega$ resistors. So the output voltage of high level of SPI suppose to be $V_{SPIhigh} = 2.5\text{ V}$. For current regulation I placed one extra $2\text{ k}\Omega$ resistor serial behind the divider.

IV.II. Final configuration

The whole circuit schematic is too big to be in text so it is possible to find it in in Appendix A. All schematic components has been described above.

Because we have high frequency part of circuit and low frequency part of circuit, we use board with four metal layers and two substrate layers as shown in Figure 36. The top metal layer of in Figure 36 is top layer of our layout shown in Appendix B represented by red color and it is reserved for high frequency part of circuit. RO4350 is substrate described above, represented by beige color in Appendix B, and FR4 is flame resistant material. The 2nd top metal layer is ground plane shown in

Appendix C, represented by turquoise color, and the bottom in Figure 36 layer is bottom layer of our layout shown in Appendix D, represented by blue color, which is reserved for DC part of circuit.

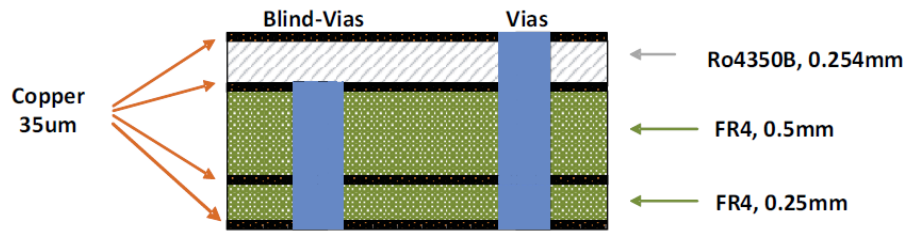


Figure 36 Structure of layers in the board [3]

In layout we just added two antenna arrays with transmission lines, pins to connect our board to arduino and microUSB connector. As it shown in Appendix B, the TX and TXX output are combined together in phase by power combiner. The pins TX, TXX, RFIN and LO has different impedance than characteristic impedance of circuit. Datasheet of RF chip BGT24MTR11 provide dimensions of impedance transformers in [3] chapter 3.4. Dimmension are also shown in Figure 37 Compensation structure [3]

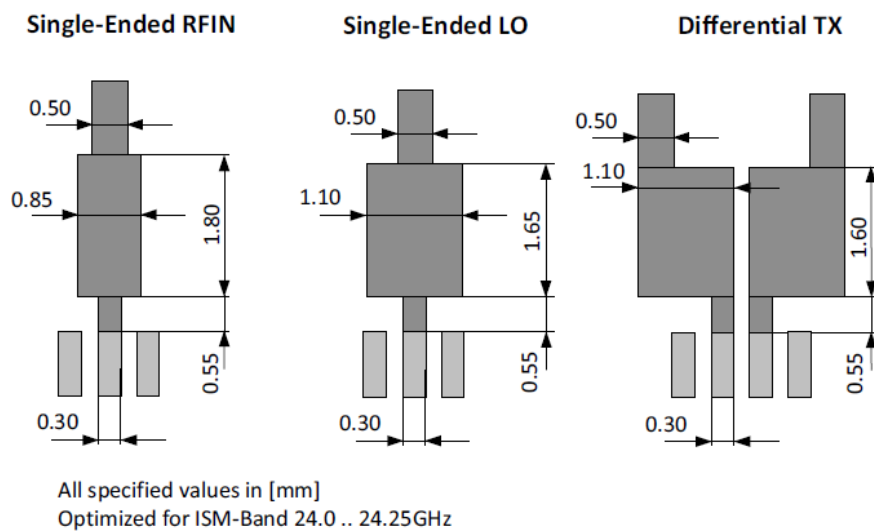


Figure 37 Compensation structure [3]

The photo of front side of radar is shown in Appendix E and photo of back side of radar is shown in Appendix F. The photo of radar connected to Arduino UNO and PC and prepared for measurement is shown in Appendix G. The GND, I-signal and Q signal are connected to Arduino UNO directly. SPI wires are connected to Aruino UNO through voltage divider. Arduino UNO and radar are powered separately by USB cables.

V. Software development

V.I. Main concept

The main concept of software to collect data for signal processing sounds very simple and it is shown in Figure 38 Basic concept of software for radar. First we need to set up radar chip and start its work. Then we need to send command to Arduino to start collecting data from analog inputs and send them to PC in proper format. Then we need to read data, plot them and compute and plot spectrum.

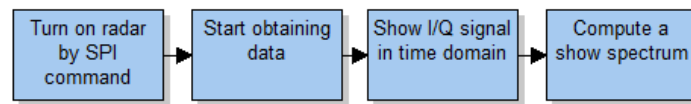


Figure 38 Basic concept of software for radar

Originally I wanted to create graphical user interface (GUI) in Matlab which will control everything and plot data in real time because Matlab can communicate with Arduino by using hardware support package for Arduino. I spent long time for that task I gave to myself and finally I realized, that it is almost impossible to obtain and plot live data from Arduino by Matlab because Matlab functions are too slow to get live data directly. This will be explained deeply in later chapters. So finally GUI in Matlab was used only for signal post processing.

V.II. Commands for Arduino UNO

V.II.I. Serial Peripheral Interface bus (SPI)

This bus is made for serial communication between head device of circuit called Master and other periphery called Slaves. The basic concept is that Master send commands to Slave and Slave execute command and send results to Master if required. This bus consists four wires. Slave Select (SS or CS) is defaultly in high level of voltage and by low voltage level value communication starts with defined Slave device. Serial Clock (CLK) ensure the synchronous communication. Master Out Slave In (MOSI) is data wire in direction from Master to Slave and Master In Slave Out (MISO) is

data wire in direction from Slave to Master. Data are synchronized with CLK. In our case MISO is not used.

According datasheet [3] page 17 the data are valid with falling edge of CLK and 15th bit is Most Significant Bit (MSB). In other words 15th bit goes first. The command to say to RF chip to start working is shown in Table 5, where H means high and L means low voltage logic value:

Table 5 Values of bits in SPI command used to set up RF chip

bit	value	description
15	H	because we don't want to reduce Gain of LNA
14,13	L	not used
12	L	enable TX power
11	L	we don't use AMUX2
10	L	test bit – has to be low, otherwise malfunction
9	L	test bit – has to be low, otherwise malfunction
8	H	we don't use AMUX1
7	H	we don't use AMUX0
6	L	disable 64k divider because we don't use 23 kHz prescaler
5	H	enable 16 divider because we use 1.5 GHz prescaler
4	L	LO is connected directly to load so we can keep 4 dB reduced LO-output power
3	L	high TX buffer output power enabled
2 ÷ 0	variable	HHH means TX power is reduced as much as possible. With LLL TX power reach maxima

This sequence of bits was set up to board Arduino UNO with programming software Arduino. And the code is shown in Appendix H. The SS pin is pin D10, MOSI is pin 11 and CLK is pin 13. SPI configuration has been set as MODE1 because this mode define that data are valid with falling edge of clock signal. Before and after clock signal 0.1 ms is reserved for be sure of value of SS.

The time sequence of this command is shown in Figure 39. The value of SS already dropped down. Bits 2 ÷ 0 are set as HHH so in this case I used measurement for the lowest TX power. The period of clock signal is $T_{CLK} \cong 7.8 \mu s$ and high value of voltage is 2.5 ÷ 2.6 V.

When command is sent once, the radar follows this command as long as we send another command. Doesn't matter how many times we power off the RF chip.

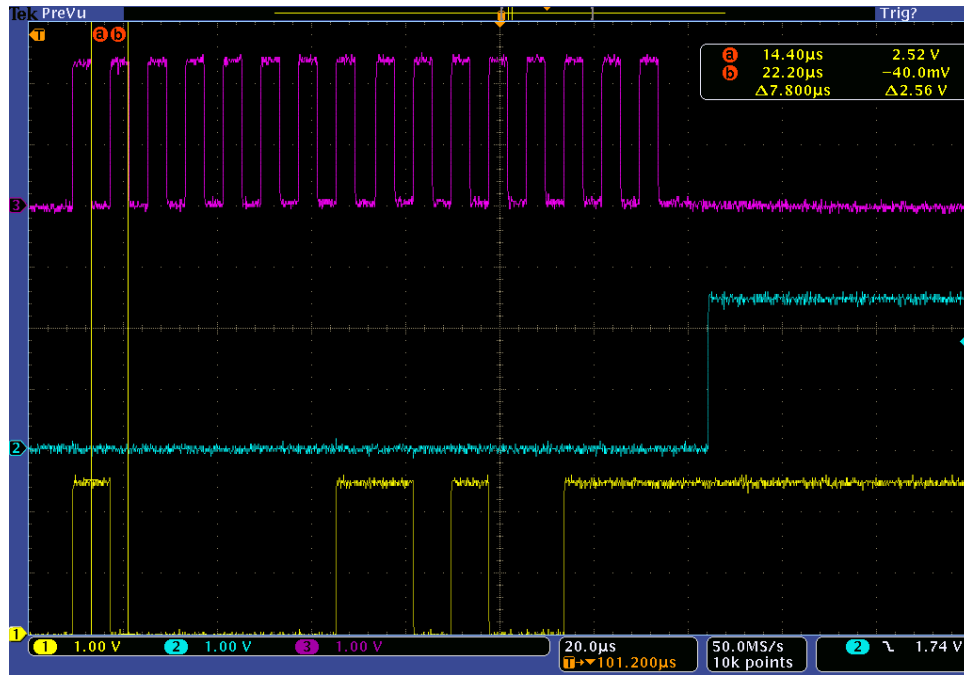


Figure 39 SPI command time sequence to set up RF chip

V.III. Sending data to PC via port COMX

The code for sending data to PC is written separately and it is shown in Appendix I. The analog input A0 has been set for in-phase signal from radar and analog input A1 has been set for quadrature signal from radar. Next step is to start serial communication through Universal Asynchronous Receiver and Transmitter (UART) to COMX, where X means number. The speed of communication is set as typical value 9600 baud (bits/s). In loop section we read voltage values on the pins and save to variables, save the time to variable, then we transfer values to volts because Arduino UNO reads values in 10-bits resolution so as numbers between 0 and 1023, and finally send values through UART to port in format "time,I-Voltage,Q-Voltage" because we will save these data later as CSV file. The end of iteration of loop is paused by 5 ms and then continue for another iteration.

Comma-separated values (CSV) is text format of data where colloms are set by commas. This file type can be easily opened by Matlab with embeded function called **cvsread**. But these data has to be saved. **CoolTerm** is suitable terminal to see live data received into the port and also to save them to text file.

V.IV. Signal processing in Matlab

Matlab is programming language of 4th generation and its main purpose is to compute difficult mathematical computations usually focused in physics and engineering. Matlab can create Graphical User Interface (GUI) and plot data which is what we need. In this work I am using version R2015b.

Before I start to explain the code I would like to introduce some basics about signal processing.

V.IV.I. Spectrum

Spectrum is representation of signal energy in frequency domain. There are many methods how to get spectrum. We are using the most popular method the **Fast Fourier Transform** (FFT).

To describe FFT I need to explain **Discrete Fourier Transform** (DFT), because FFT is based on DFT. DFT is defined as

$$S(n) = \Delta t * \sum_{k=0}^{N-1} x(t_k) * e^{(-j2*\pi*k*n/N)} \quad n = 0,1, \dots, N - 1 \quad (36)$$

where $S(n)$ is the n^{th} coefficient of the DFT and $x(t_k)$ denotes the k^{th} of the time series which consist of N samples and $j = \sqrt{-1}$. $x(t_k)$ can be complex numbers and $S(n)$ can be also complex numbers. The Δt is time between two samples. To simplify equation (36) it is often expressed as

$$S(n) = \Delta t * \sum_{k=0}^{N-1} x(t_k) * W_N^{nk} \quad n = 0,1, \dots, N - 1 \quad (37)$$

where W_N^{nk} is twiddle factor

$$W_N^{nk} = e^{(-j2*\pi*n*k/N)} \quad (38)$$

The paragraph above has been cited from [10]. Just for imagination equation (38) I plotted “twiddle factor” W_N^{nk} for $n = 0,1,2,3$ in Matlab. This is shown in Figure 40 and it say that for every $x(t_k)$ in equation (38) we combine W_N^{nk} with $x(t_k)$ and by summing them we get energy on coefficient $S(n)$ which is related to frequency f_n .

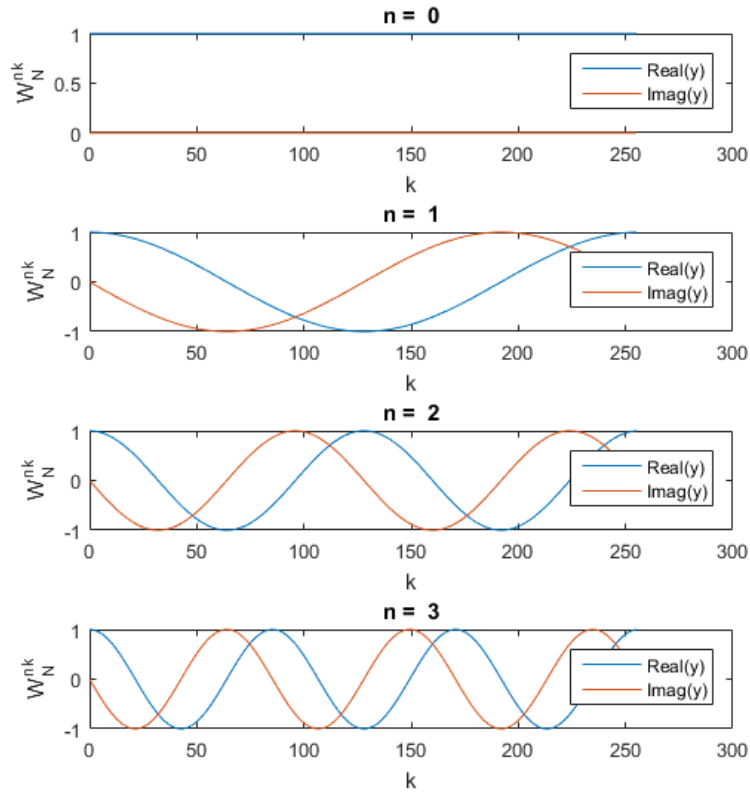


Figure 40 Explanation of “twiddle factor” W_N^{nk} in DFT

For example I have created two sinusoidal. One carry frequency 50 Hz and amplitude 10 and second carry frequency 120 Hz with amplitude 6. This two sinusoidal I combined together, plot and computed DFT. Because spectrum is symmetric, I cut the mirror part. The result is shown in Figure 41

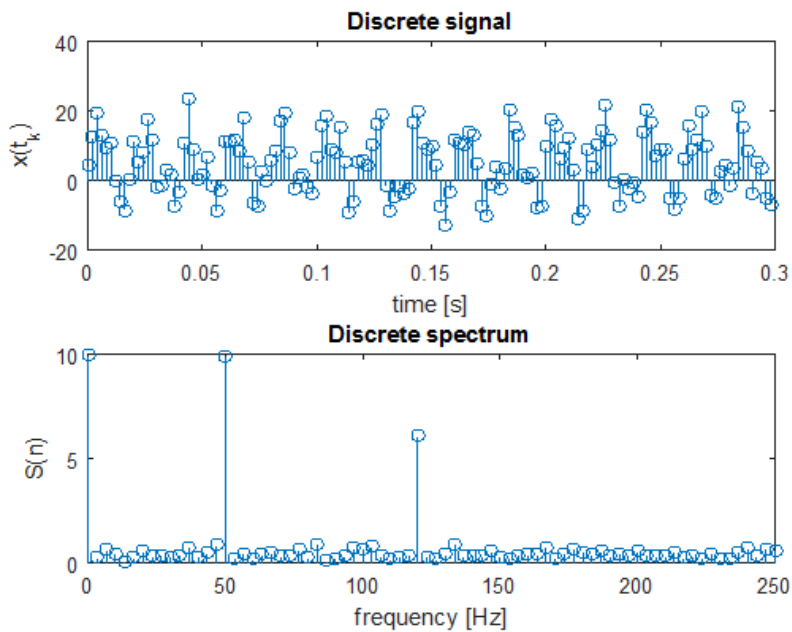


Figure 41 Example of DFT for two sinusoidal

But if we have high number of samples the computation time grows with N^2 . This is not very efficient. However Cooley and Tukey founded solution in 1965. It is known as Fast Fourier transform.

“**Fast Fourier Transform** (FFT) is a method for efficiently computing the DFT. The efficiency of this method is such that solutions to many problems can now be obtained substantially more economically than by using DFT.” [10]

“FFT can reduce not only computation time but also round-off errors. Both are reduced by factor of $(\log_2 N)/N$ with compare with DFT. For example if we have $1024 = 2^{10}$ samples, then $N * \log_2 N = 10240$ computing operations are needed for FFT. In compare DFT requires $N^2 = 1048576$ computing operations.” [10]

“Now let define term **decimation in time**. The DFT and its inverse are of the same form so that procedure, machine, or sub-routine capable of computing one can be used for computing the other by simply exchanging the roles of $x(t_k)$ and $S(n)$, and make appropriate scale-factor and sign changes. Reversing roles $S(n)$ and $x(t_k)$ gives the form called decimation in frequency.” [10]

Suppose a time of N samples. Let's divide it to two functions. Function $y(t_k)$ is composed from even-numbered points ($x(0), x(2), x(4), \dots$) and function $z(t_k)$ is composed from ($x(1), x(3), x(5), \dots$). These functions are shown in Figure 42 and could be written formally as

$$\begin{aligned} x_e(t_k) &= x(t_{2k}) \\ x_o(t_k) &= x(t_{2k+1}) \quad k = 0, 1, 2, \dots, N/2 - 1 \end{aligned} \quad (39)$$

and these functions have DFT defined by

$$\begin{aligned} S_e(n) &= \frac{2}{N} \sum_{k=0}^{\frac{N}{2}-1} x_e(t_k) * e^{(-j4*\pi*n*k/N)} \\ S_o(n) &= \frac{2}{N} \sum_{k=0}^{\frac{N}{2}-1} x_o(t_k) * e^{(-j4*\pi*n*k/N)} \quad n = 0, 1, 2, \dots, \frac{N}{2} - 1 \end{aligned} \quad (40)$$

According equations above, we could say that

$$S(n) = S_e(n) + W^n S_o(n) \quad 0 \leq n < N/2 \quad (41)$$

In equation (41) we write that $W^n = W_N^{nk}$ to simplify expression for use afterward.

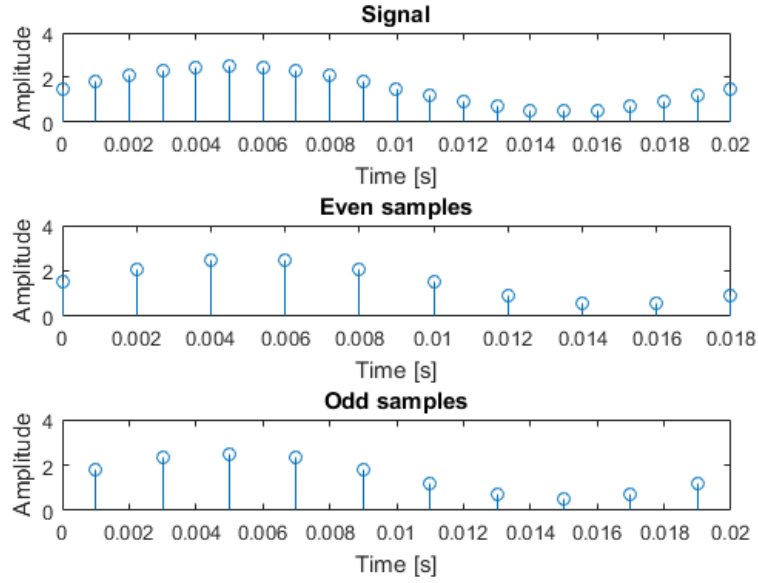


Figure 42 Even and Odd samples

For values greater than $N/2$, the discrete Fourier transforms $S_e(n)$ and $S_o(n)$ repeat periodicity of values taken on when $n < N/2$. Therefore, substituting $n + N/2$ for n in (41), we obtain

$$\begin{aligned}
 S\left(n + \frac{N}{2}\right) &= S_e(n) + \left(-j2 * \pi * \left[n + \frac{N}{2}\right]/N\right) * S_o(n) \\
 S\left(n + \frac{N}{2}\right) &= S_e(n) - (-j2 * \pi * n/N) * S_o(n) \\
 S\left(n + \frac{N}{2}\right) &= S_e(n) - W^n * S_o(n) \quad 0 \leq n < N/2
 \end{aligned} \tag{42}$$

The signal flow chart will look like Figure 43 Signal flow diagram for implementation FFT from DFT $N=4$ for the case of DFT with $N = 4$. The inputs left are points of $x_0(t_k)$ and outputs on the right side are points of transform $S(n)$. The united arrows represent adding and W^x are multiplying coefficients. Thus with this process we obtain $N * \log_2 N$ computations which contain only addition and multiplication.

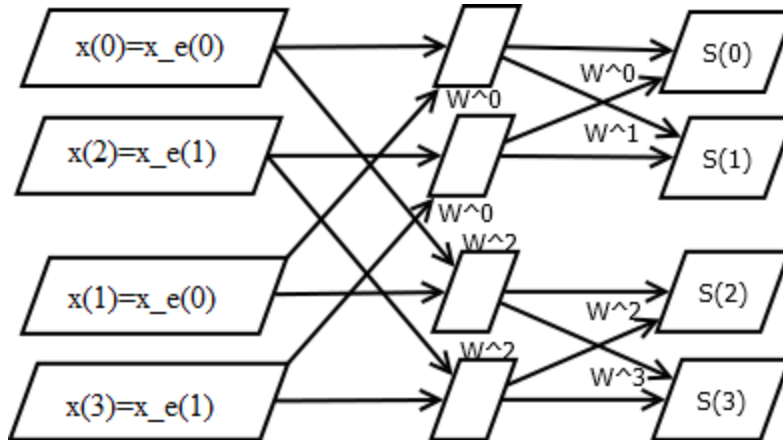


Figure 43 Signal flow diagram for implementation FFT from DFT N=4

V.IV.II. Windowing

Because signal we get is contain many other noisy components in time and the spectrum could not be clear, for this purpose windowing was made. In simple words, we modify signal in time domain like that for spectrum more important parts will be clear. [11]

The window looks like the amplitude in the middle is the highest and in the beginning or in the end it is lowest. There are many types of windows. Hamming window, Blackman window, Hann window and more.

In our case the Hamming window is used. Example of Hamming window with $N = 50$ samples is shown in Figure 44.

Hamming function is defined as

$$w(n) = \alpha - \beta * \cos\left(\frac{2 * \pi * n}{N - 1}\right) \quad (43)$$

where $\alpha = 0.54$ and $\beta = 1 - \alpha$ are constants, N is number of samples and n is actual sample. To compare with many other windows, this window don't fall values to zero on the edges. Application is easy. It is just multiplying signal in time domain by Hamming window with the same number of samples as signal has, see Matlab code in Appendix M.

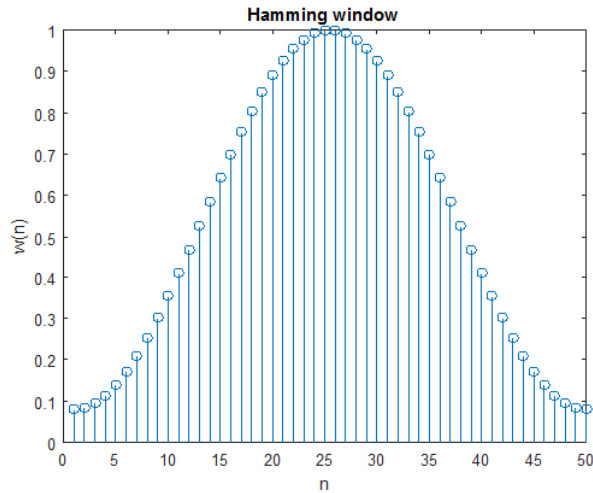


Figure 44 Hamming window

V.IV.III. Matlab GUI

The principal diagram of our code designed by Matlab is shown in Figure 45 and parts of code are shown in Appendices K, L and M. After opening **CSV** file, see Matlab code in Appendix K, we need to save data to variables, because we need to modify this values for next use. First, we should delete the first and the last row, because **CoolTerm** save data in real time so the first and the last row can be uncompleted. Then shift beginning of time domain to 0 s. The I and Q data also have to be shifted to zero, because we shifted them in differential amplifier to be read by Arduino UNO. After modifying data, we will save data to **line** objects which means that they will be plotted.

When data in time domain are plotted, we call **plotSpec** function, see Matlab code in Appendix L. Function **plotSpec** compute absolute values of FFT, cut negative frequencies and change frequency domain to units min^{-1} to see respiration and heartbeat clearly. Because we suppose frequencies $0 \div 150 min^{-1}$ we can also cut some part of spectrum which is not interesting for us.

Because we want to have amplitude of spectrum normalized, the highest peak has value 1 and rest of spectrum is normalized to this value. Finally we can save variables to **line** objects which means that we will plot spectrum in current Figure.

To see better the frequency of main peak we call function **max** to find position of maximum in spectrum and set marker there by saving frequency and magnitude to **line** object of marker. Marker itself is invisible in the beginning but values are shown in text control.

In the end, we set boundaries of plotted spectrum. Later we can change boundaries and cut spectrum to see results better.

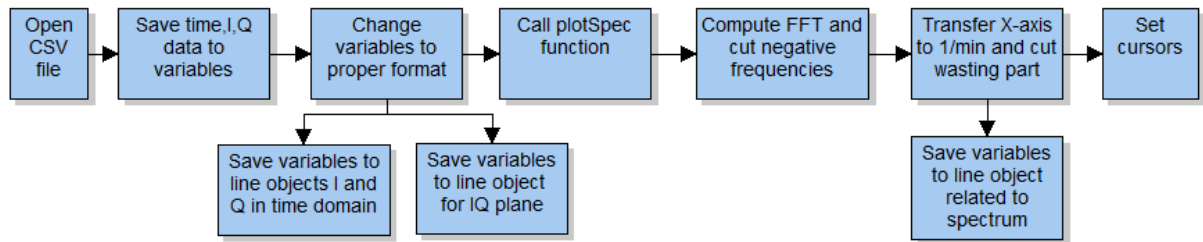


Figure 45 Principal diagram of Matlab code to show data and spectrum

The final GUI is shown in Figure 46. Pushbutton **Open CSV file** open browser to find and open CSV text file.

Time cursors section shows boundaries of signal in time domain. This boundary can be set by writing value and to text-control and clicking somewhere else in current Figure. The code find the closest value of sample and set boundary there. The result is to plot black vertical line which define boundary (time cursor). By clicking pushbutton **Cut** the signals in time domain are zoomed in and spectrum is recomputed.

Frequency cursors section works exactly the same as **Time cursors** section. Only exception is that by clicking pushbutton **Cut** no values are changed. Only spectrum is zoomed in.

Show Marker section show the position of marker in frequency domain. If user check the check-button **Show Marker**, the marker with triangle shape appear in spectrum in position defined in text controllers. The position of marker can be changed by clicking to spectrum line or by changing value in **Freq [1/min]** text controller.

All text-controls are defined by following description. If text-control is empty, nothing can be written by user. If typed text is not number, the previous value is kept. If typed value is out of range, the closest value is set. Typed comma “,” is automatically changed to dot “.”.

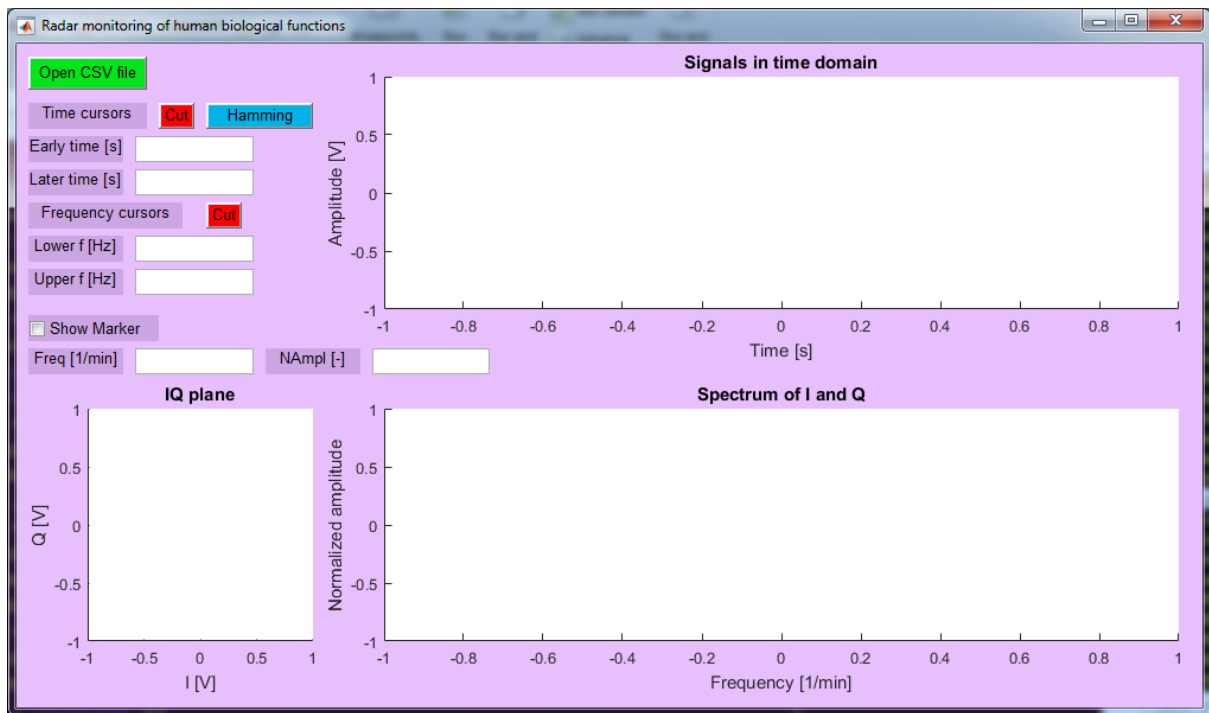


Figure 46 Empty GUI of human biological functions monitor.

If user type for example to **Early time [s]** text-control in **Time cursors** section value, which is higher than value in **Later time [s]** text-control, the value is automatically changed to the same value as in **Later time [s]**. Opposite system is the same. This also works in **Frequency cursors** section.

If user type anything to **NAmp [-]** text-control, it is automatically changed to previous value because any value of Normalized amplitude can be the same at more frequencies.

The pushbutton **Hamming** multiply signals in time domain by Hamming window and call **plotSpec** function again, see Matlab code in Appendix M.

VI. Experimental measurements

The arrangement of measurement is shown in Appendix J. The subject under test (me) was sitting in distance $1 \div 1.5 \text{ m}$ from radar depends on type of measurement. Radar is in the middle of that photo and it is fixed to aim to my chest. PC is behind the radar. PC is connected to Arduino UNO and Arduino UNO is connected to radar with all radar input and output pins. I held wireless mouse to control measurement on PC. To be sure that we are getting proper results, the oscilloscope show voltage of in-phase and quadrature signal. Measurement has been executed at the weekend in morning time, so I was alone in the laboratory to avoid disturbing elements.

I simulated many options like calm breathing, deep breathing, fast breathing and holding a deep breath. The results are following.

VI.I. Calm breathing

Next measurement has been taken from distance $\sim 1.2 \text{ m}$ with reducing of TX by SPI command as bit 2 is "H", bit 1 is "H" and bit 0 is "L". This is measurement of calm breathing and it is shown in Figure 47. As we can see, even this measurement was too long, the respiration is recognized as about 12.5 min^{-1} . But heartbeat is hidden between lobes of disturbing elements and also if we have too long measurement, the respiration and heartbeat can change. Thus as you can see cursors in Figure 47 I cut the signals to short part from 70 s to 100 s to reduce influence of disturbing elements to spectrum.

The cut signals are shown in Figure 48. The heart beat is still a little bit hidden so we used Hamming window two times and result is shown in Figure 49. Here the heartbeat is definitely recognizable. I placed marker on heartbeat lobe and it says that observed heart beat in frequency $\sim 74.7 \text{ beats/min}$.

For considering of space next measurements will be shown as final post processing. In-phase signal in time domain is always blue. Quadrature signal in time domain is always red.

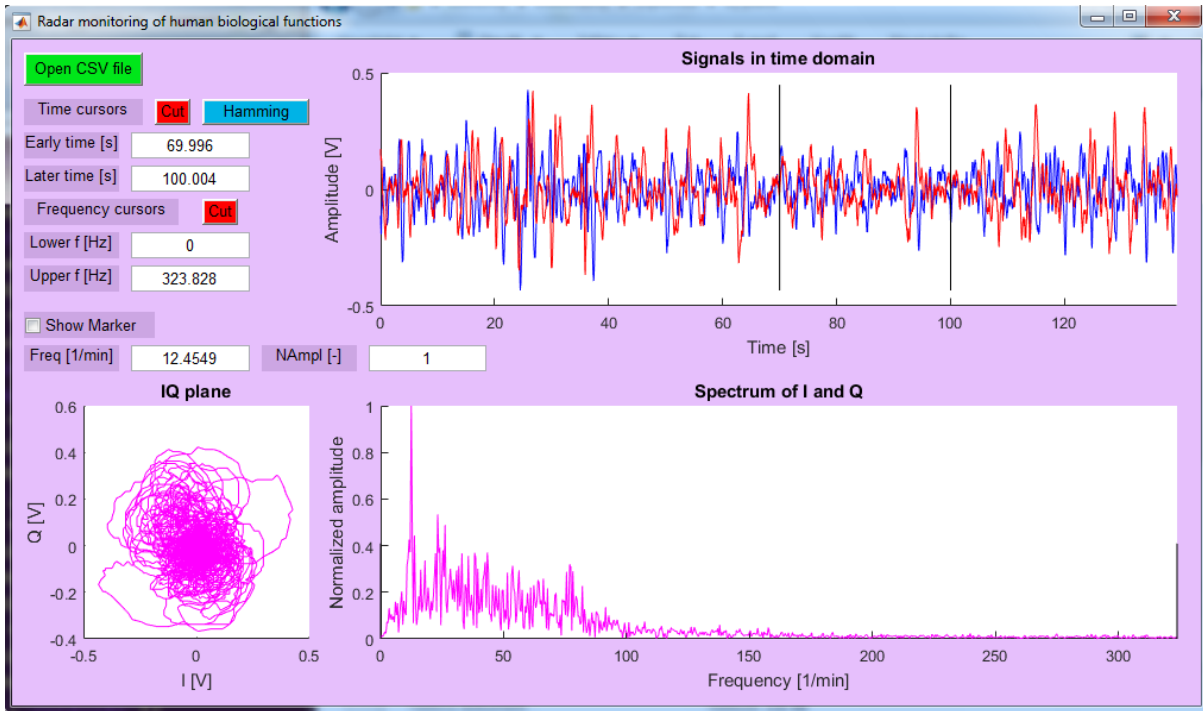


Figure 47 Measurement of calm breathing from distance 1.2 m, basic view

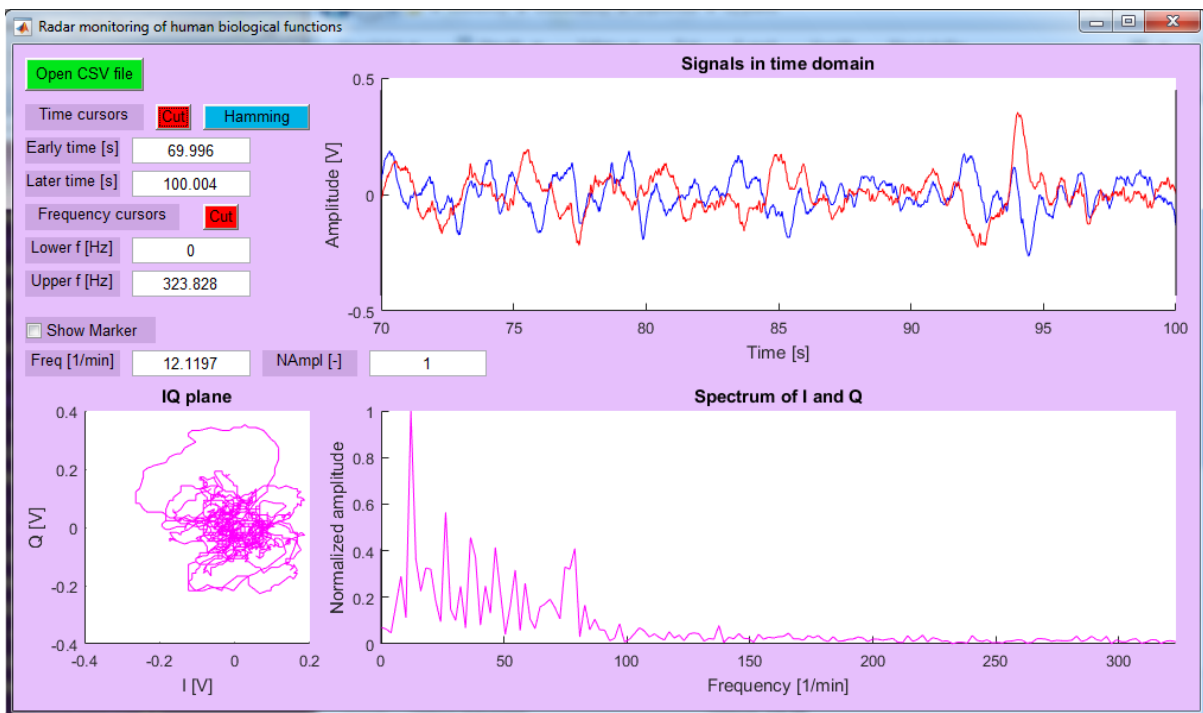


Figure 48 Measurement of calm breathing from distance 1.2 m, zoomed view

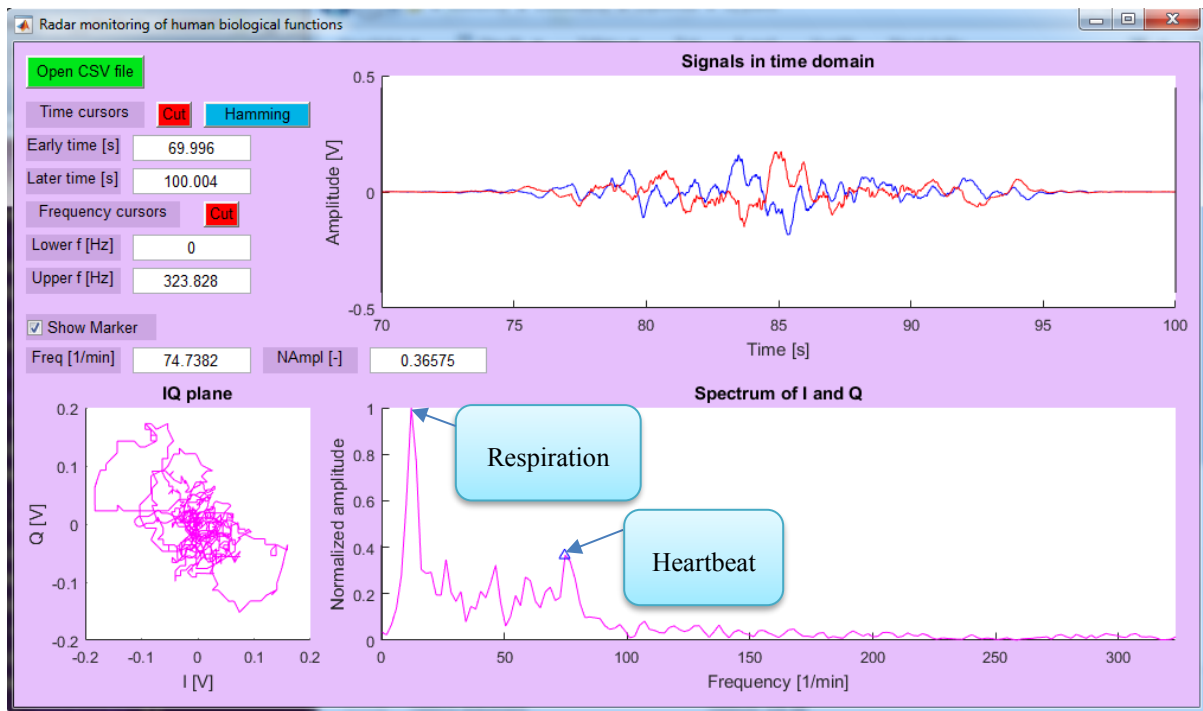


Figure 49 Measurement of calm breathing from distance 1.2 m, zoomed view, 2x Hamming window

VI.I. Fast breathing

To simulate some critical condition of patient I measured very fast respiration. The result of measurement is shown in Figure 50. This measurement was made from distance ~ 1.5 m with no reduction of TX power. We can see, that I/Q-signal voltage reached saturation of differential amplifier for many times but not so rapidly.

Because this measurement was very short, the frequency of my heartbeat did not grow. The highest peak of spectrum show frequency of heartbeat. But our marker is placed on the peak of respiration. It was very frequent respiration which was faster than two times per second. So the case of fast respiration of patient can also be observed. But I need to note that in this case Hamming window was not used.

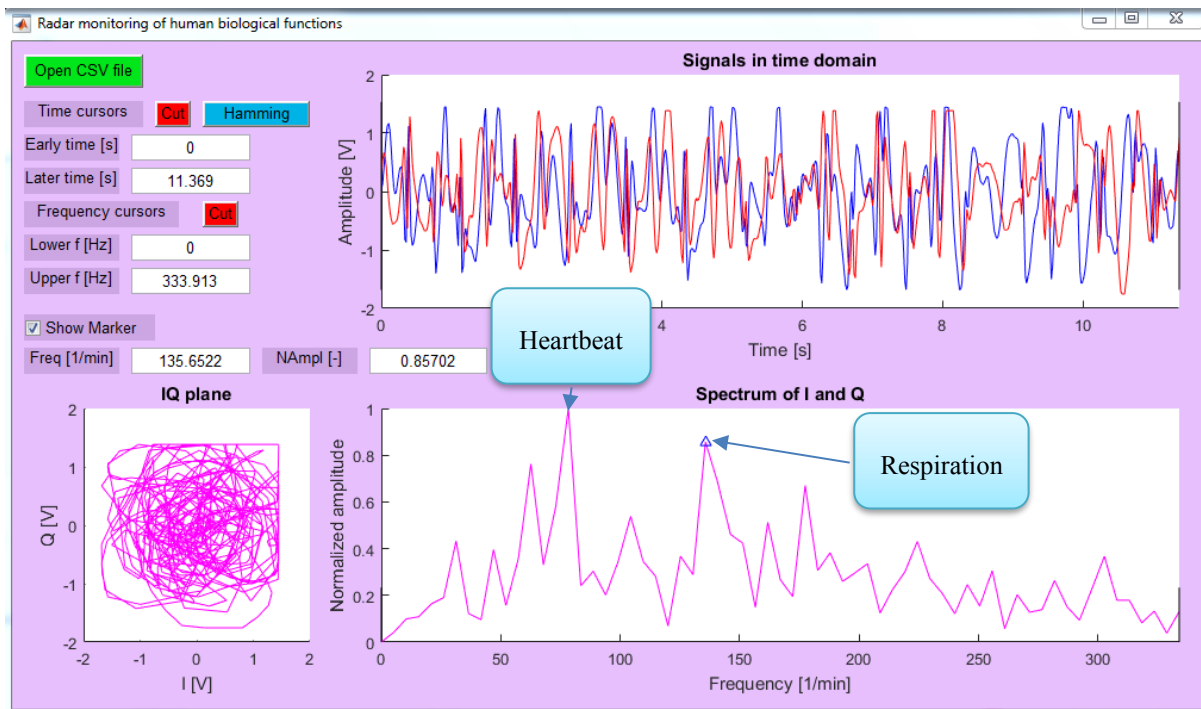


Figure 50 Measurement during fast breathing from distance 1.5 m, basic view without Hamming window

VI.II. Deep breathing

So now what about case if patient breathe with normal frequency but very deeply. This has been measured from distance ~ 1.5 m and because of high amplitude of I/Q-signal, I reduced TX power to minimum by SPI command “HHH” for bits $2 \div 0$ to avoid saturation differential amplifier.

The measurement is shown in Figure 51. During dep breathing my heartbeat frequency is about ~ 94.6 beats/min but respiration has not been observed. The signal has been multiplied once by Hamming window. This spectrum suppose respiration about ~ 35 min⁻¹ which means that time between two inhales supposed to be about ~ 1.7 s which is much faster than I did. The reason of getting this result is because length of movement of the chest, during deep breathing, is too long so the radar itself reverse value of I/Q-signal. In other words, the one inhale will look like many waves instead of one increasing wave which is exactly what we get.

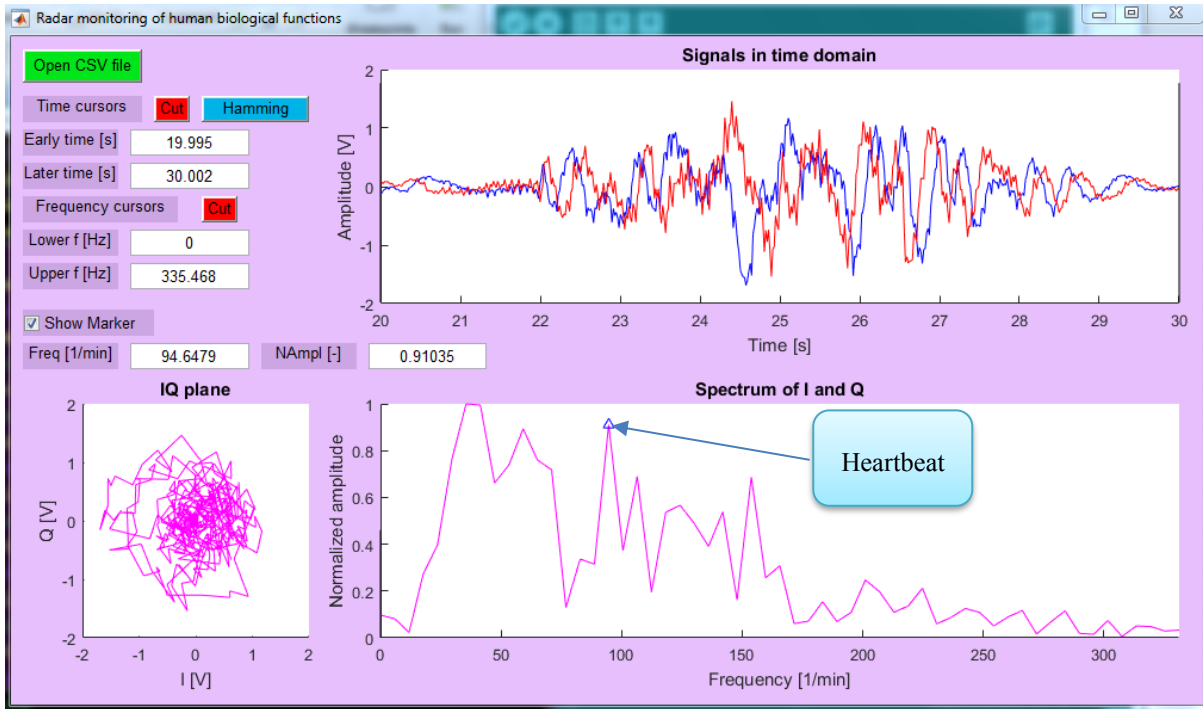


Figure 51 Measurement during deep breathing from distance 1.5 m, zoomed view

VI.III. Holding a deep breath

When I inhaled as much oxygen as possible and hold the breath, the magnitude and frequency of my heartbeat increased rapidly. This is the best time to observe heartbeat.

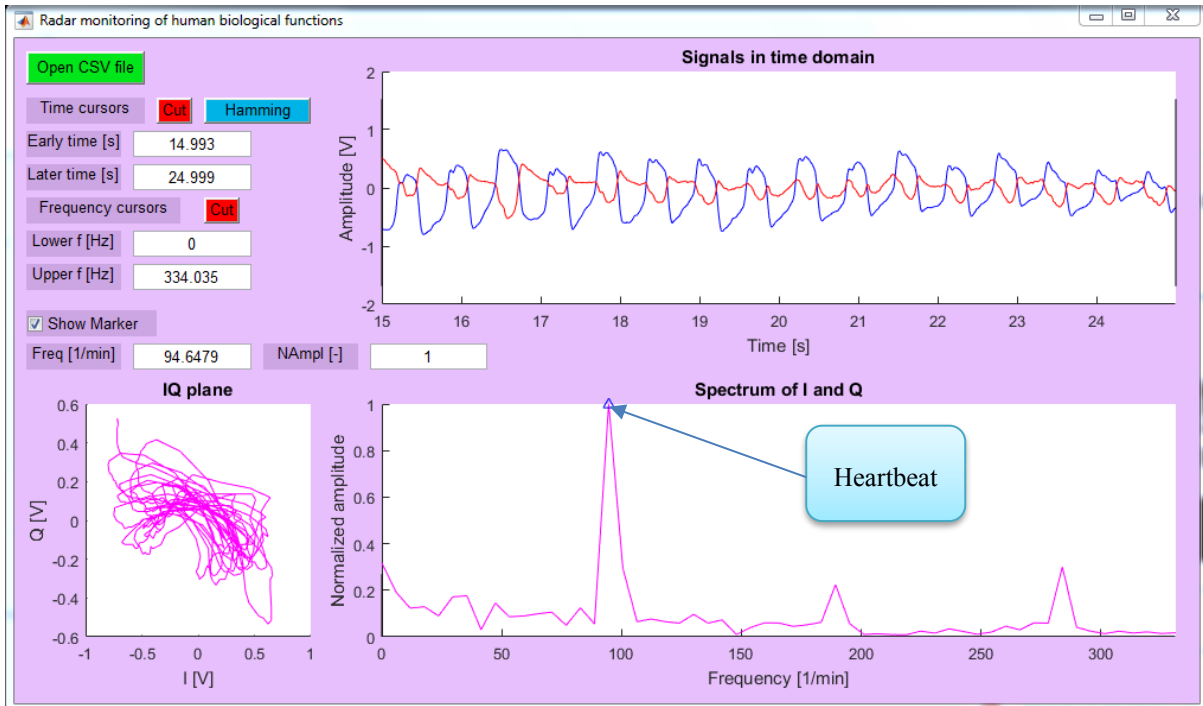


Figure 52 Measurement during holding a deep breath from distance 1.5 m, zoomed view

The measurement in Figure 52 has been taken from distance $\sim 1.2\text{ m}$ with maximum TX power setting (The bits 2 ÷ 0 of SPI command are all “L”). The respiration is of course not measured but heartbeat is $\sim 94.6\text{ beats/min}$.

VI.IV. Speech

Last measurement is made during speech. Because people a little bit exhale with every syllable, the spectrum can look like in Figure 53. The SUT was 1.5 m from radar and radar has been set to maximum TX power. I/Q-signals has been multiplied by Hamming window twice to see better results of spectrum.

Respiration here is hidden to more components. One is frequency, when patient inhale, another ones are frequencies of slabs he/she say. So in this case, respiration and heartbeat cannot be measured very well because respiration is spreaded to more frequencies and these frequencies can cover heartbeat frequency.

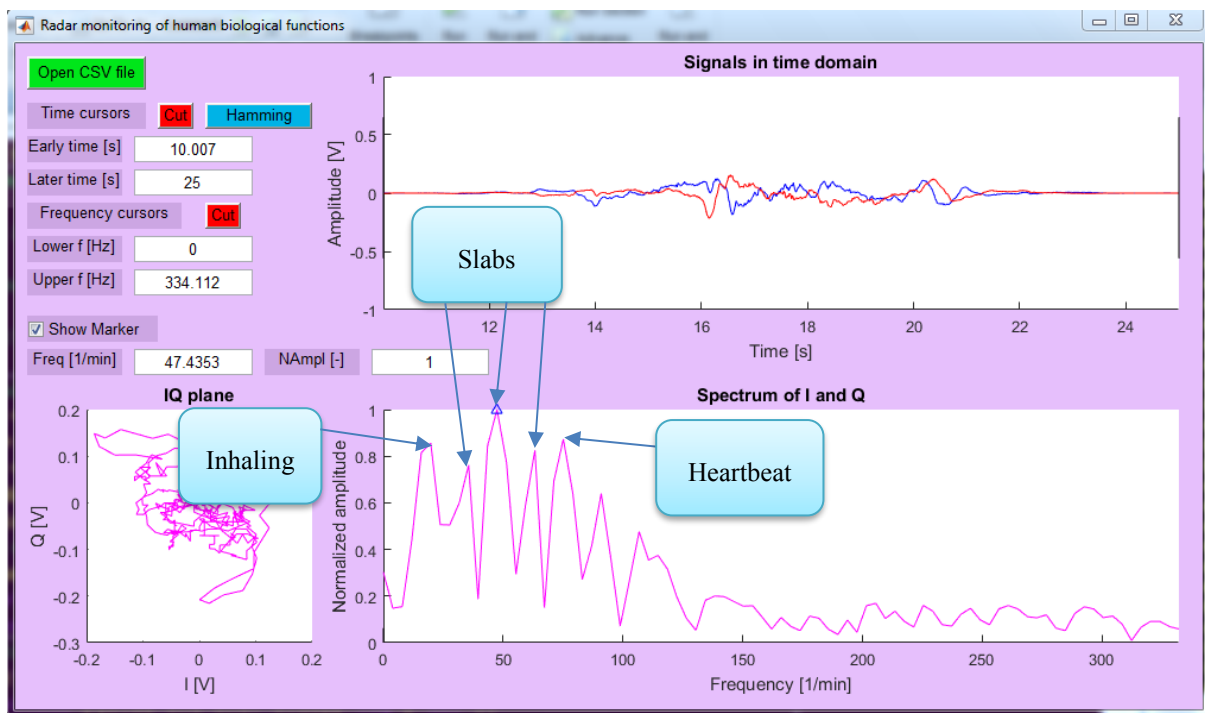


Figure 53 Measurement during speech from distance 1.5 m with zoomed view and 2x Hamming window

VI.V. Moving with body

Next measurement shows signals and spectrum, when patient can for example turn his body on the bed. I measured myself from distance 1 m and I turned my body for four times and in the end I moved to stop of recording data. I used gain reduction for transmitting power in order bit $2 \div 0$ is “HHL”.

The measurement is shown in Figure 54, where you can see the signal was in saturation for many times. The peak of spectrum was founded as $\sim 57.2 \text{ min}^{-1}$, which in that moment was not heartbeat or neither respiration frequency. So we got some unrecognizable information. But if we look at the Figure 55, the Hamming window was used and we can find heartbeat as $\sim 77.3 \text{ min}^{-1}$. But it is only because there are some parts of time without moving body and thanks for Hamming window these moving parts are almost abolished. Respiration is not measured, because during moving the respiration is not regular.

So when patient is moving or shaking, we cannot measure any parameter so we can say that if patient is moving for very long time, the spectrum is changing quickly and it usually doesn't show any useful information.

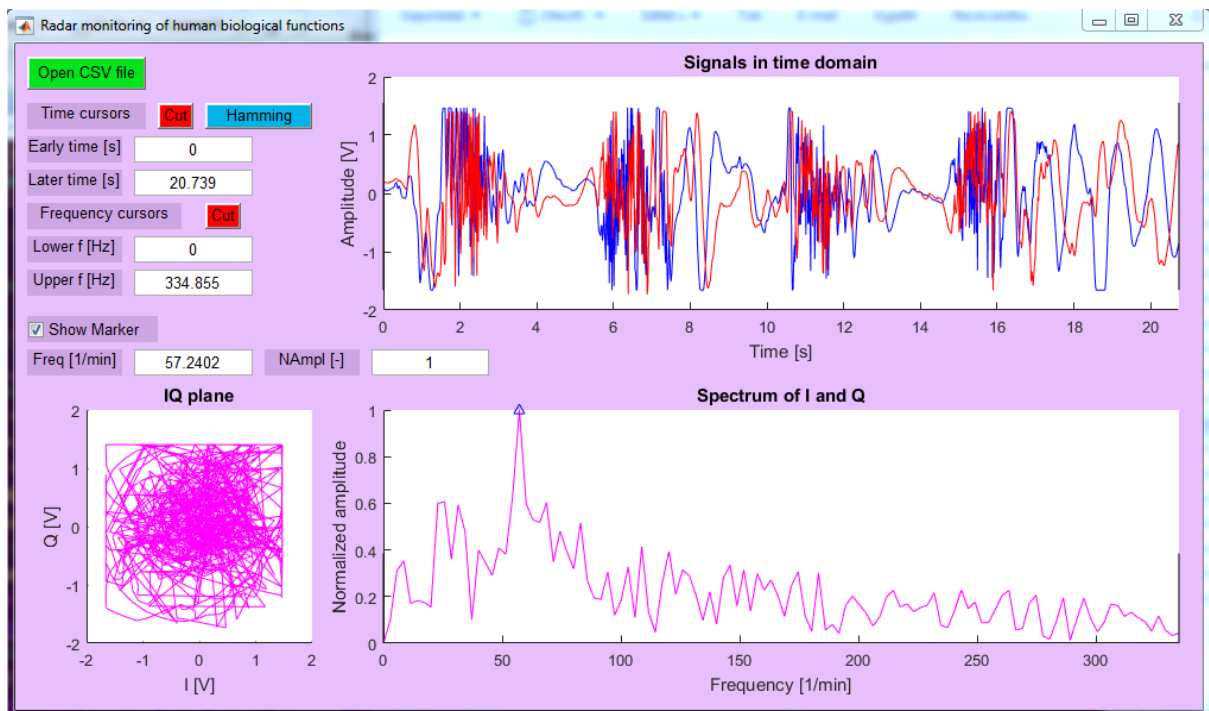


Figure 54 Measurement during part off time moving with 1 m away body

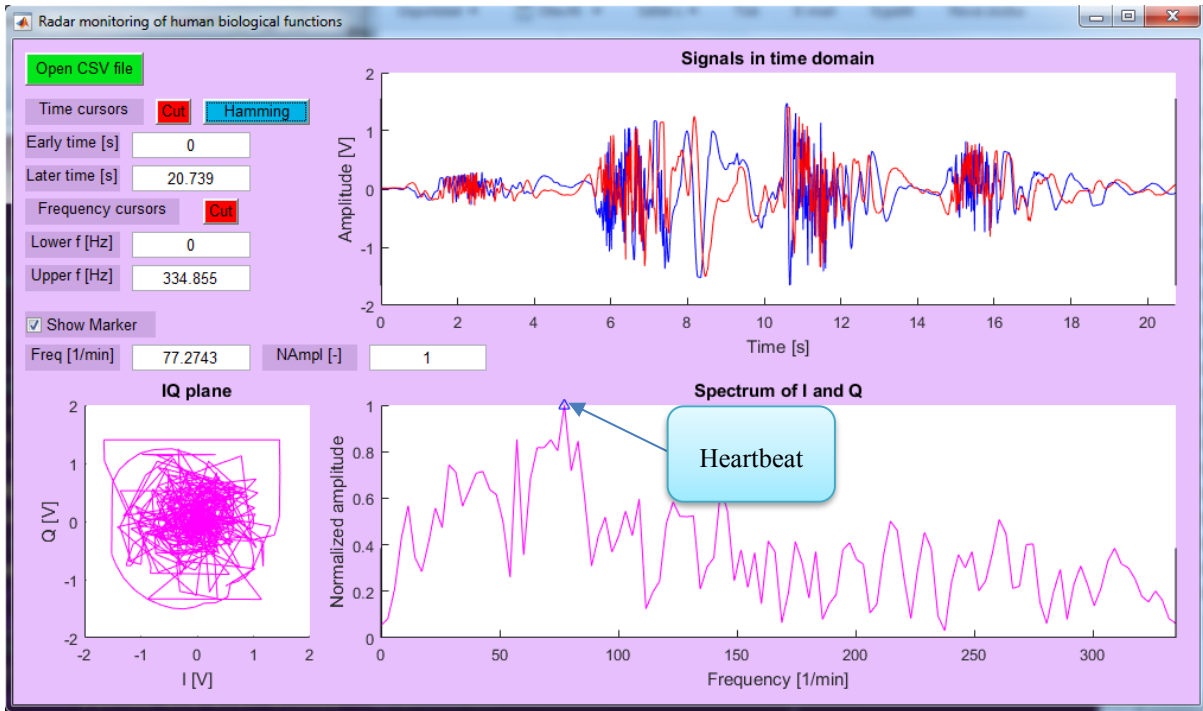


Figure 55 Measurement during part off time moving with 1 m away body, with Hamming window

VII. Conclusion

Successfully designed and established a radar for patient's biological functions monitoring which executed measurements of some patient's conditions and computed signal processing to show results.

For transmitting and receiving continuous wave, 4x4 patch antenna array has been designed with simulated center frequency $f_{rs} \cong 24.15$ GHz but measured center frequency is $f_{rm} \cong 24.39$ GHz. The bandwidth of 4x4 patch antenna array for $S_{11} < -10$ dB is within $23.75 \div 24.5$ GHz in simulation in HFSS and $23.2 \div 24.9$ GHz in measurement. The maximum gain of 4x4 patch antenna is $G_{max} = 16.66$ dBi and $HPBW \cong 20^\circ$ in simulation and $G_{max} = 11.5$ dBi and $HPBW \cong 20^\circ$ in measurement. The reason of this difference can be justified by sing very narrow transmission lines which cause lower accuracy in fabrication and impedance these transmission lines can be different.

The core of circuit is RF chip BGT24MTR11 which works in range $24 \div 26$ GHz include all necessary high frequency components like transmitter, receiver, low-noise amplifier, power amplifier, local oscillator tuned by DC voltage and I/Q-mixer with four outputs which are 100x amplified with differential amplifier. The center frequency is tuned to $f_r \approx 24.2$ GHz.

Arduino UNO send command through SPI to set up RF chip and receive I/Q-signals by analog inputs and these values are sent through USART to PC port COMX and saved by terminal CoolTerm as CSV text file.

The CSV text file is opened by Matlab GUI, signals are plotted and spectrum is computed by using FFT. The highest peak in spectrum is founded automatically with marker but position of marker can be changed to see another peaks. Hamming window can be used manually to see better results.

Measurement showed results for case of normal condition of patient, deep breathing, fast breathing and holding a deep breath. It is always fine to cut signal for $15 \div 30$ s long time, because the respiration and heartbeat frequency can change. In case of deep breathing, respiration is difficult to recognize, because of long distance movement and in case of speech, nothing can be clearly measured because people exhale every syllable so the respiration frequencies can cover heartbeat.

For normal condition it is suitable to multiply signal by Hamming window about $1x \div 3x$ to get good results and the respiration can be recognized as about $10 \div 15 \text{ min}^{-1}$ and heartbeat frequency can be recognized as about $65 \div 80$ beats/min. In case of fast breathing, the heartbeat frequency grows and respiration and heartbeat frequency can be recognized without windowing I/Q-signals. In deep breathing case it is more complicated to identify respiration frequency because the movement of chest is too long. But heartbeat is still measurable. In case of moving with body, no parameter can be recognized.

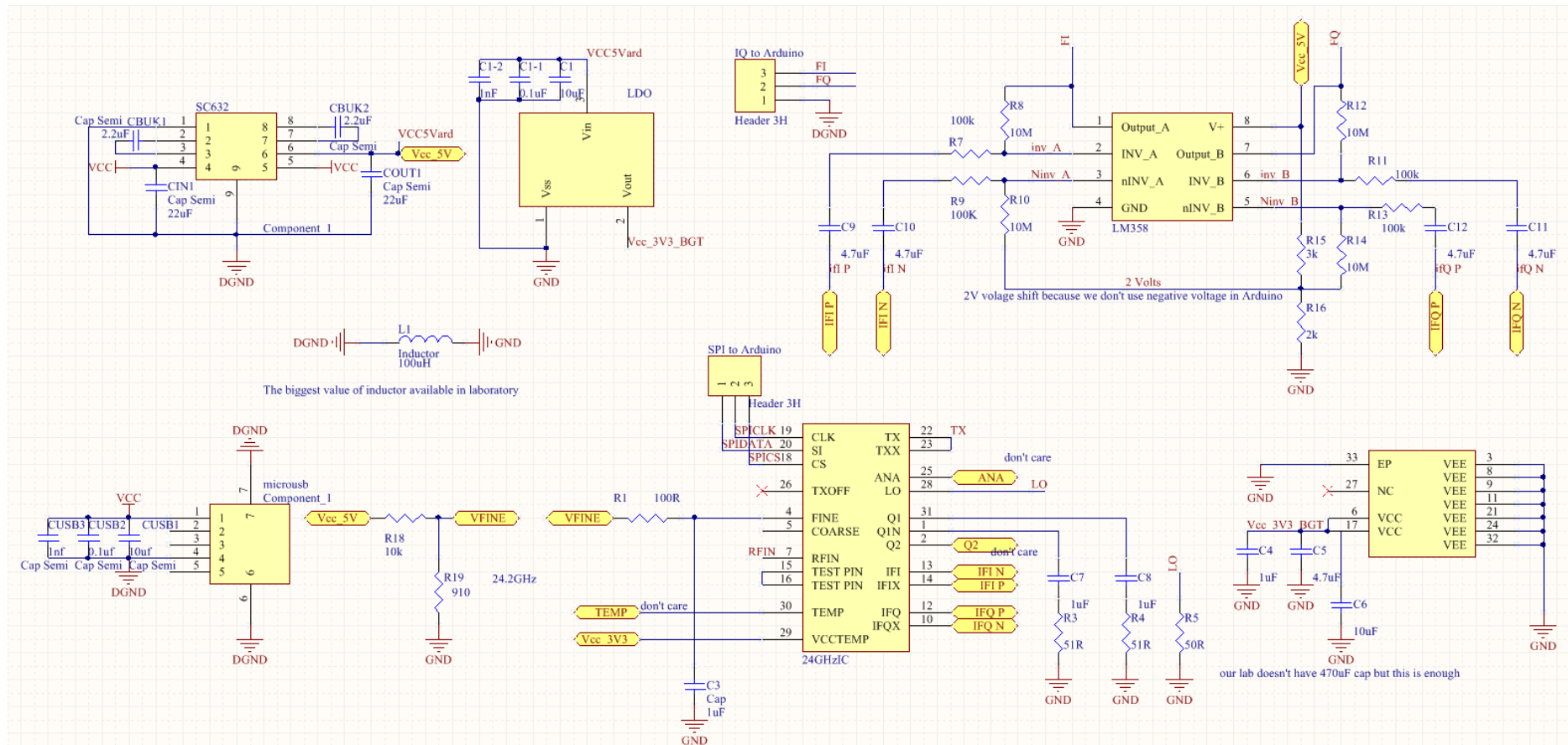
In the future view I would like to put CPU chip directly on the board and use Bluetooth or Wi-Fi to send signals to PC where signals and spectrum would be observed in real time.

References

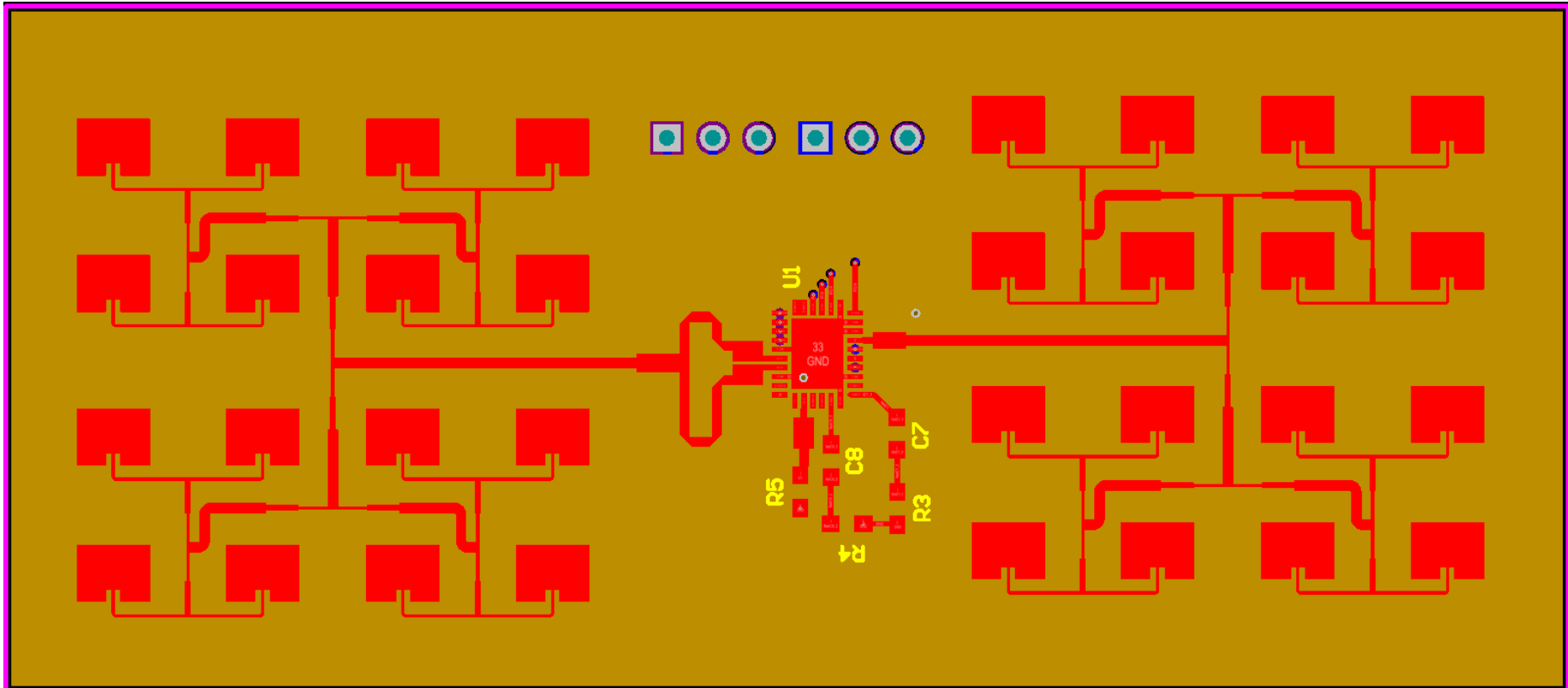
- [1] Yi HUANG and Kevin BOYLE, "Antennas: from theory to practice". Chichester, UK: John Wiley & Sons, 2008.
- [2] A. D. Droitcour, O. Boric-Lubecke, V. M. Lubecke, J. Lin and G. T. A. Kovacs, "Range correlation and I/Q performance benefits in single-chip silicon Doppler radars for noncontact cardiopulmonary monitoring", *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 3, pp. 838-848, March 2004.
- [3] Infineon Technologies AG, "Silicon Germanium 24 GHz Transceiver MMIC", BGT24MTR11 datasheet Revision 3.1, 2014-03-25
- [4] Rogers Corporation, "RO4000® Series", datasheet, Revised 2017
- [5] Southwest Microwave, "2.40mm JACK (FEMALE)END LAUNCH CONNECTORLOW PROFILE", 1492-06A-6 datasheet, 2012-09-26
- [6] S. Sabah and R. Lorenz, "Design and calibration of IQ-mixers "
- [7] SEMTECH, "Fixed 5.0V Output Charge Pump Regulator", SC632 datasheet, 2011
- [8] CYStech Electronics Corp., "Low Dropout CMOS Voltage Regulators", LM6206N3 datasheet
- [9] STMicroelectronics, "Low-power dual operational amplifiers", LM158, LM258, LM358 datasheet
- [10] W. T. Cochran *et al.*, "What is the fast Fourier transform?," in *Proceedings of the IEEE*, vol. 55, no. 10, pp. 1664-1674, Oct. 1967.
- [11] WELCH, Thad B., Cameron H. G. WRIGHT a Michael G. MORROW. Real-time digital signal processing from MATLAB to C with the TMS320C6x DSPs. 2nd ed. Boca Raton: CRC Press/Taylor & Francis Group
- [12] T. W. Hsu and C. H. Tseng, "Compact 24-GHz Doppler radar module for non-contact human vital-sign detection," *2016 International Symposium on Antennas and Propagation (ISAP)*, Okinawa, 2016, pp. 994-995.

Appendix

A. Schematic of whole circuit



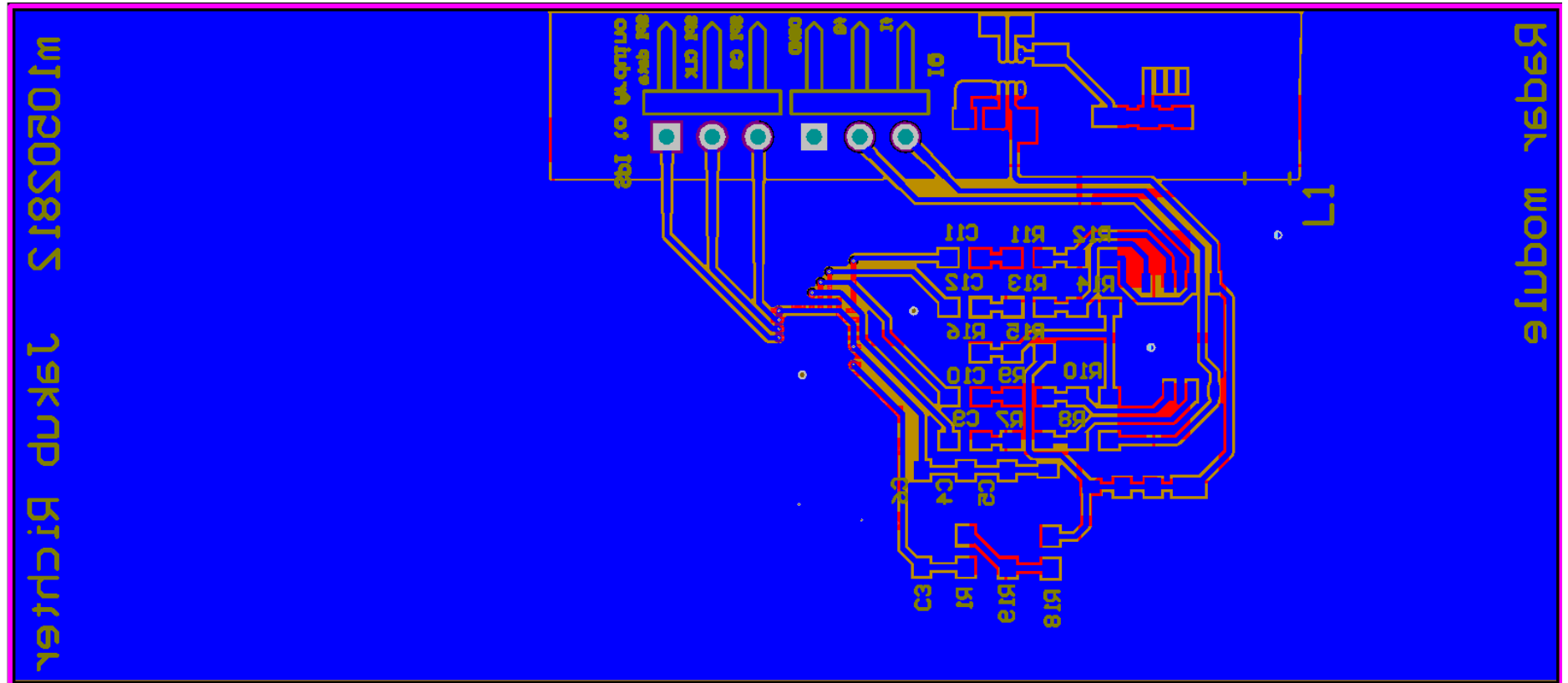
B. Top layer layout



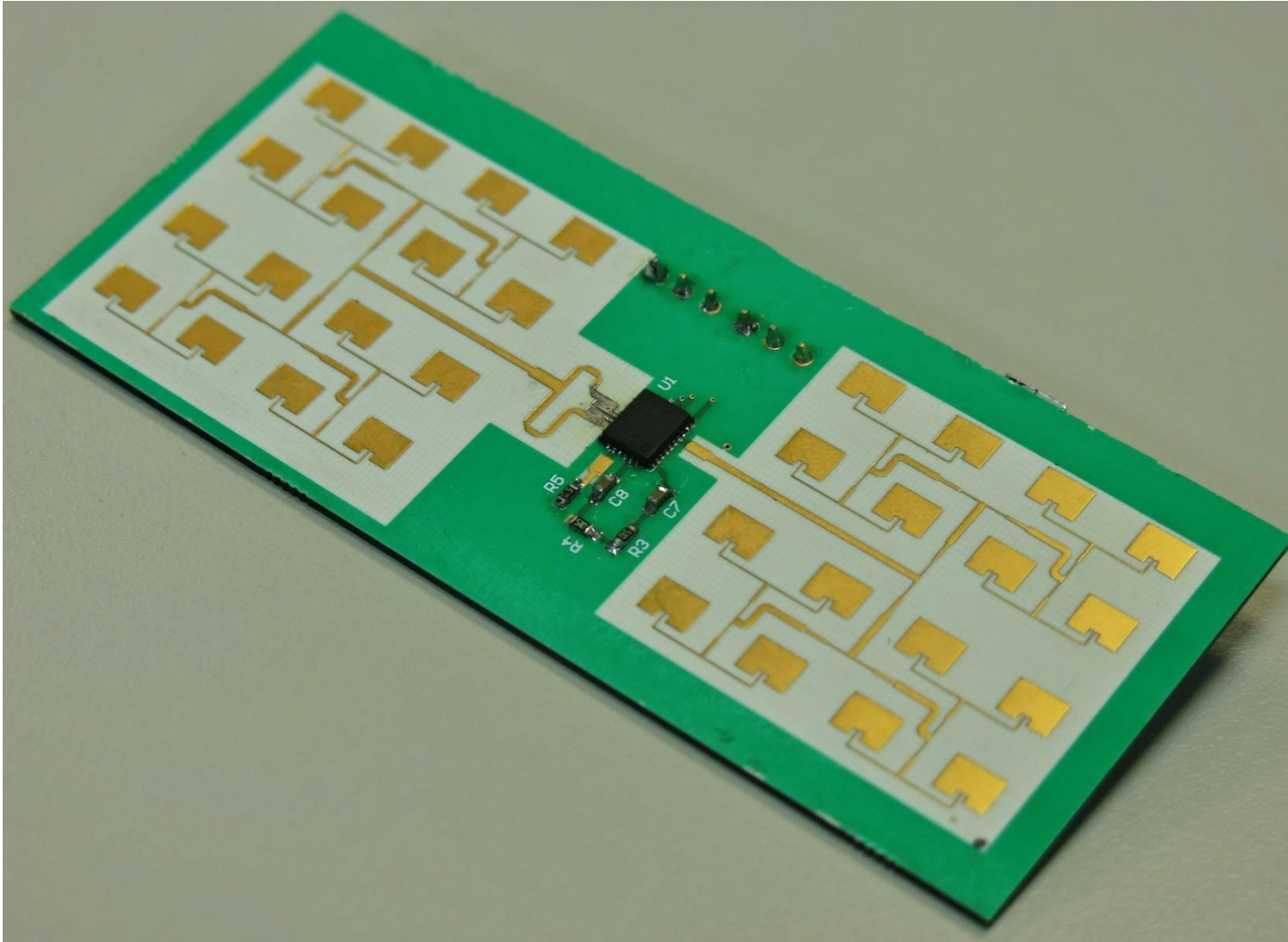
C. Middle layer layout



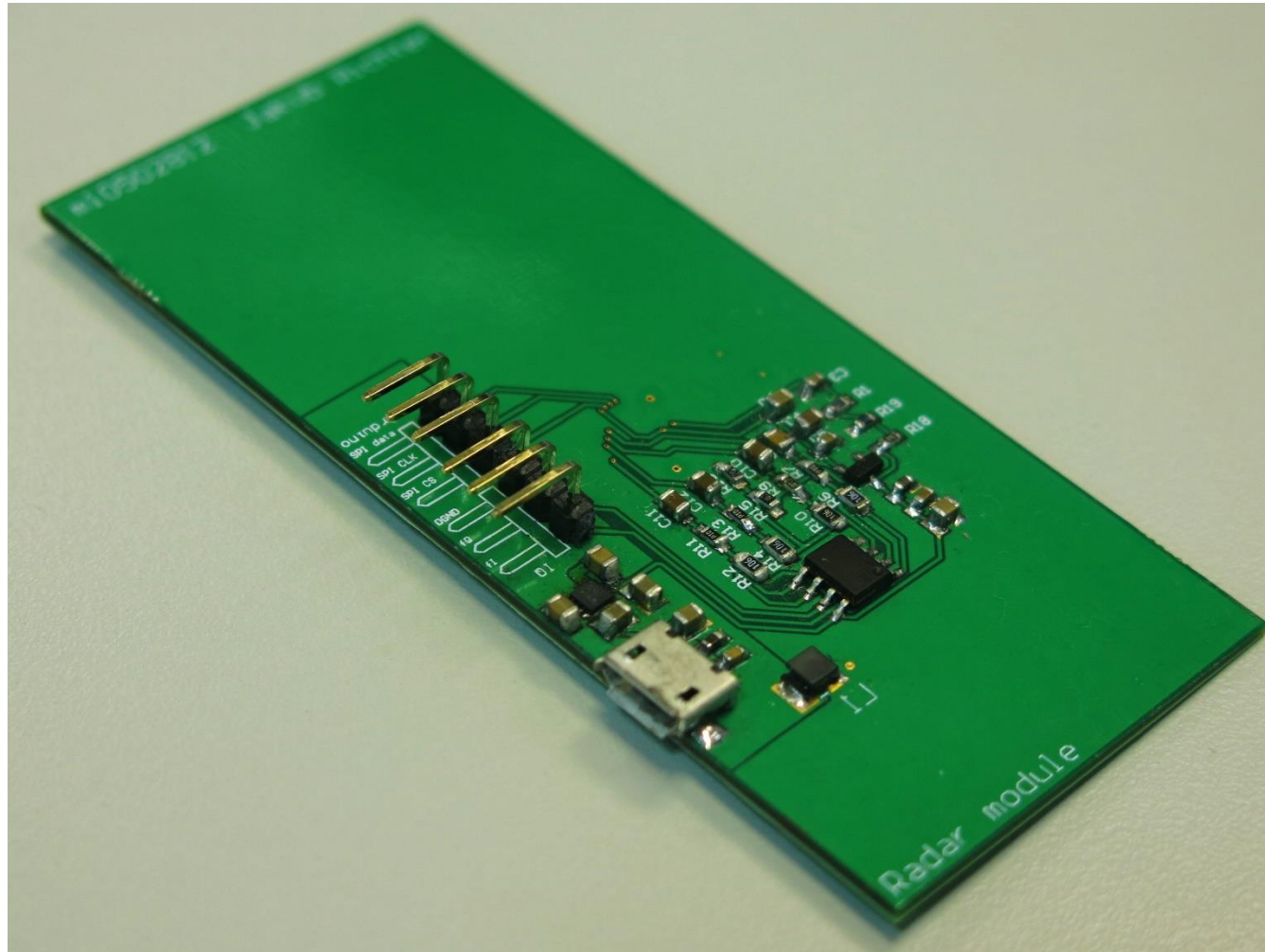
D. Bottom layer layout



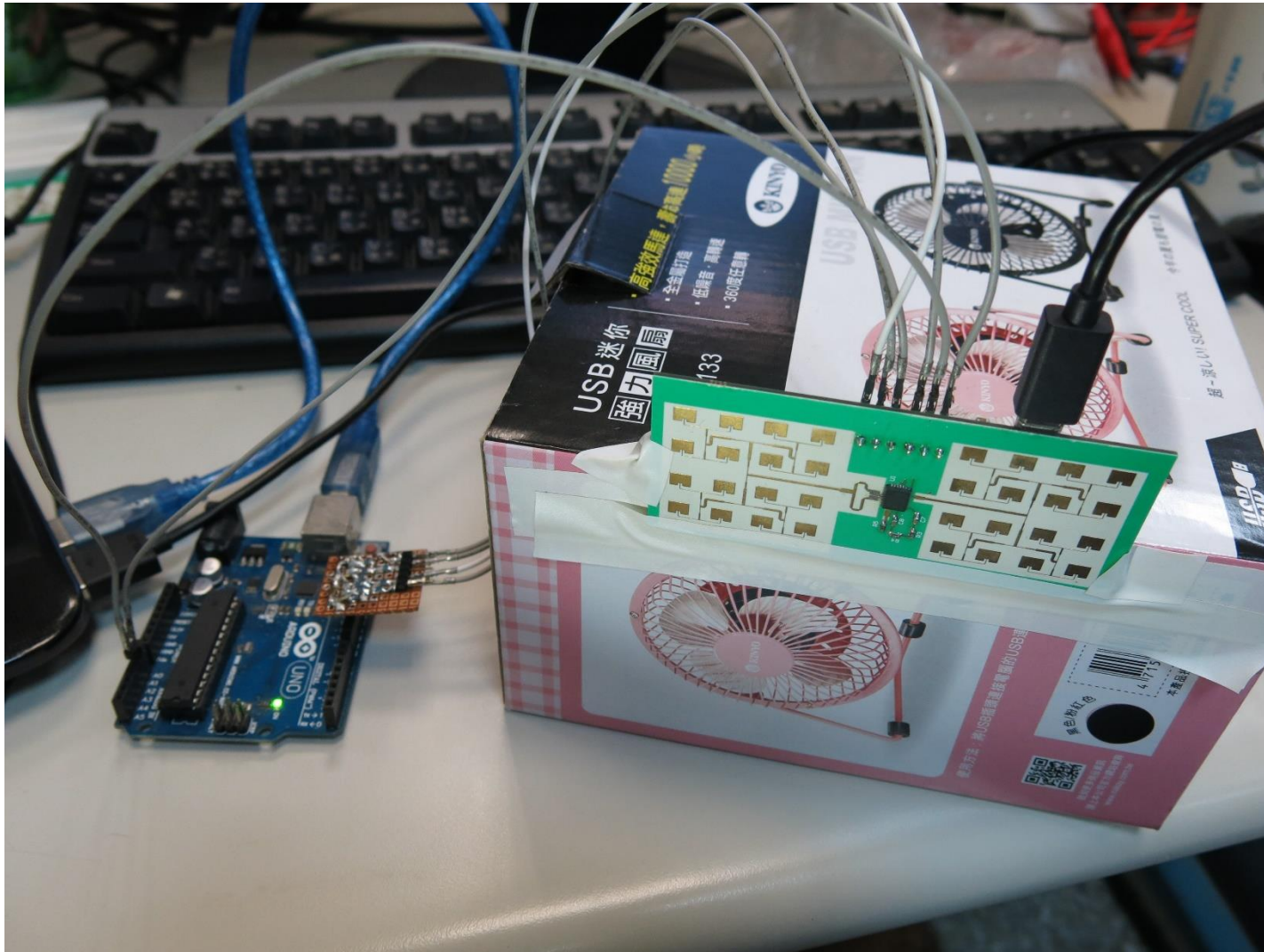
E. Radar photo front side



F. Radar photo back side



G. Photo of radar connected to Arduino UNO and PC



H. Arduino code – SPI command

```
//Include the SPI library
#include <SPI.h>

//Set Slave Select pin, other SPI pins are set automatically
const int SISe = 10;
const int hod = 13;
const int slaveData = 11;

// Set command variable. We have 16bit command so separate it to two 8-bit
const byte commandMSB = B10000001;
const byte commandLSB = B10100111;
//bit 0-2 are for reduction of IQ output -> 111 means that output power is reduced to minimum

void setup() {
  // Set SS pin pirection, others are handled automatically
  pinMode(SISe,OUTPUT);
  pinMode(hod,OUTPUT);
  pinMode(slaveData,OUTPUT);

  //Initialize SPI
  SPI.beginTransaction(SPISettings(160000, MSBFIRST, SPI_MODE1));
  //MODE1 because radar module read when falling edge occure

  //Write command
  digitalWrite(SISe,HIGH);
  delay(100);
  digitalWrite(SISe,LOW);
  delay(0.1);
  SPI.transfer(commandMSB);
  SPI.transfer(commandLSB);
  delay(0.1);
  digitalWrite(SISe,HIGH);

  //End SPI communication
  SPI.end();
}
```

I. Arduino code – collect data

```
// constants
const int A0Pin = A0;
const int A1Pin = A1;

//variables
int A0Value = 0;
int A1Value = 0;
float A0Volt = 0;
float A1Volt = 0;
unsigned long t = 0;
void setup() {
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  A0Value = analogRead(A0Pin);
  A1Value = analogRead(A1Pin);
  t = millis();
  A0Volt = float(A0Value)*5/1023;
  A1Volt = float(A1Value)*5/1023;

  Serial.print(t);
  Serial.print(",");
  Serial.print(A0Volt);
  Serial.print(",");
  Serial.print(A1Volt);
  Serial.println();
  delay(5);
}
```

J. Photo from measurement



K. Matlab openCSV function code

```
function OpenCSV
FileName = uigetfile('*.*txt','Select the CSV File');
if FileName ~= 0
    % data format is time(ms),A0,A1
    IQData = csvread(FileName,1,0);
    % start reading from 2nd row because 1st row is usually not filled
    % data format is Collom1: time(ms) Colom2: A0 Collom3: A1
    IQData = IQData(1:end-1,:);
    % delete last row because it is usually not filled and transpose
    % data format is Row1:time(ms) Row2: A0 Row3: A1
    IQData(1,:)=IQData(1,+)/1000;
    %% normalize time
    StartTime = IQData(1,1);
    IQData(1,:)= IQData(1,)-StartTime;    % start time from zero
    %% normalize to middle
    avI=mean(IQData(2,:));
    avQ=mean(IQData(3,:));
    IQData(2,:)=IQData(2,)-avI;
    IQData(3,:)=IQData(3,)-avQ;
    %% plot data
    plotSignals(IQData(1,:),IQData(2,:),IQData(3,:));    % time,I,Q
    plotIQ(IQData(2,:),IQData(3,:));                    % I,Q
    %% plot spectrum
    [perSh,fAx] = plotSpec(IQData(1,:),IQData(2,:),IQData(3,:));    % time,I,Q
    % points to show
    setCursorsT(IQData(1,:),1,length(IQData(1,)),IQData(2,:));
    setCursorsF(fAx,1,length(fAx),perSh);
end
end
```


L. Matlab plotSpec function code

```
function [perSh,fAx] = plotSpec(timeR,I,Q)
%% Just compute and plot positive part of fft
%
% inputs
% timeR - time
% I      - I signal
% Q      - Q signal
%
% outputs
% perSh - data of y axis of spectrum (energy)
% fAx   - data of x axis of spectrum (frequency)
%
%%
compIQ=complex(I,Q);
fs=1/mean(timeR(2:end)-timeR(1:end-1));           %frequency of samples
N = round(fs*256/20);                             %number of samples
per=1/N*abs(fft(compIQ));                          % use in case of -f 0 f spectrum
PosHalf = round(length(per)/2)+1;                 % position of middle
perSh(:) = per(1:PosHalf); % we want just positive frequencies
xlimit = round(length(perSh)*0.2);
perShSm(:)=perSh(1:xlimit);
[M,I] =max(perShSm);                               %find maximum
perShSm(:)=perShSm(:)/M;                          % normalize amplitude that maximum is 1
fAx = linspace(0,round(0.5*fs),length(perSh));
fAx = fAx(1:xlimit)*60;
% use for 0 f spectrum, set limitation of size and convert to 1/min
spec=findall(gcf,'Tag','spec');
set(spec,'XData',fAx,'YData',perShSm);
FAxes=findall(gcf,'Tag','axF');
set(FAxes,'XLim',[fAx(1) fAx(end)]);
%% set cursor to highest peak point ans show values
cur = findobj('Tag','FreqCursor');
curA = findobj('Tag','cursorAV');
curF = findobj('Tag','cursorFV');
set(cur,'XData',fAx(I),'YData',perShSm(I));
set(curA,'String',num2str(perShSm(I)));
```

```
set(curF,'String',num2str(fAx(I)));  
end
```

M. Matlab HammingWin function code

```
function hammWin(dataI,dataQ)  
%% Compute Hamming window  
%  
%inputs  
% dataI - line object with information about in-phase data in time domain  
% dataQ - line object with information about quadrature data in time domain  
%  
%%  
I=dataI.YData(:);  
Q=dataQ.YData(:);  
hI=I(:).*hamming(length(I));  
hQ=Q(:).*hamming(length(Q));  
set(dataI,'YData',hI);  
set(dataQ,'YData',hQ);  
plotSpec(dataI.XData(:),hI,hQ);  
plotIQ(hI,hQ);  
end
```