Semester project in ANEMS laboratory

FEM simulation:
Thermal properties of thin film quartz and silicon resonators

---

Professor: Guillermo Villanueva
Supervisor: Marco Liffredo
Author: Eliott Sizey

2021 - Spring semester

Abstract

This semester project aimed at simulating two approaches for low temperature dependence of resonance frequency of resonators. The first one consists in using AT-cut quartz as the resonator's material. The AT-cut quartz wafers are made using a 35° angle with respect to the z-axis and they show very low resonance frequency shift in the temperature range [-40, 100]°C for thickness shear modes. The second approach is based on adding a Lithium Niobate thin film on top of a silicon resonator to compensate for the resonance frequency shift (due to temperature) of the later. Initially, the change of material properties due to temperature were investigated through literature review and simulations on Matlab. Once those properties were defined, more detailed simulations were carried out using COMSOL Multiphysics. The minimum resonance frequency shift achieved for 0.35.82 (ZXZ Euler angles) rotated quartz was 0.36ppm/°C. A frequency shift of 2.7ppm/°C was obtained for a $2\mu m$ thick silicon resonator coated with $0.5\mu m$ thick, 0.38.90 rotated (ZXZ Euler angles) Lithium Niobate.

Acknowledgement

The realisation of this project could not have been possible without the help and guidance of professor Guillermo Villanueva. I am grateful for the help provided by Marco Liffredo along the project. I thank Silvan Stettler for providing the matrix rotation MATLAB code and for his advice.

Contents

List of Figures

1 Literature research

The main goal of the literature researches was to find the temperature dependent stiffness matrices of the AT cut quartz and silicon such that, for each term of the matrix :

$$C_{ij} = C_{0ij} \cdot (1 + Tc_{ij}^{(1)}(T - T_0) + Tc_{ij}^{(2)}(T - T_0)^2 + Tc_{ij}^{(3)}(T - T_0)^3) \quad (1)$$

where $Tc_{ij}^{(1)}$, $Tc_{ij}^{(2)}$, $Tc_{ij}^{(3)}$ are the first, second and third order temperature coefficients of the stiffness matrix of the material, C_{0ij} are the stiffness coefficients at reference temperature T_0 (generally 25°C) and T is the temperature of the material. The matrix is 6x6 but is symmetric so there only are 21 different coefficients.

In the case of AT-cut quartz, we also needed the relative permittivity and the piezoelectric matrices as well as their temperature dependency.

The values retrieved from the literature and used in this project are given in A.1.

1.1 AT-cut Quartz

According to the researches conducted, there doesn't exist any scientific paper that clearly reviews the matrices and coefficients of the AT cut quartz (35° rotation from the z-axis). However, there are many sources for the coefficients of non rotated quartz that all report similar results ([War], [R D89]). In the paper that served as a source for this semester project [Bec], the authors measured the first, second and third order temperature coefficients of the stiffness matrix of quartz and compared their results to the existing literature. See A.1 for the coefficients. Once those values were retrieved, they needed to be converted to AT cut quartz by rotating the different matrices. This was done using the tensor transformation method of B.A.Auld [B A73] in which we consider the x-convention of the Euler angles [Wei]: there are three rotation angles, ϕ about the initial z axis, θ about the new x'-axis and ψ about the new z'-axis as shown in figure 1.

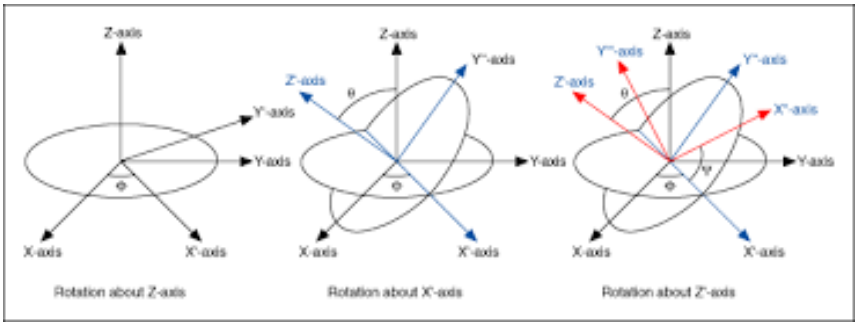

Figure 1: ZXZ Euler angles convention

Using this convention, the angles corresponding to the AT-cut quartz would be $\phi = 0°$, $\theta = 35°$ and $\psi = 0°$.

1.2 Silicon and Lithium Niobate

In the case of the Silicon, mainly one source was used to determine the material properties [Jaa]. In this paper, they investigate the effect of doping on the stiffness coefficients of the silicon for different resonating modes. Given the symmetry of silicon, only three different coefficients are necessary to determine the whole stiffness matrix (A.1): c_{11}, c_{12} and c_{44}. Those coefficients and their 1st and 2nd order temperature dependency were retrieved from this paper for different doping levels (A.1).

For Lithium Niobate, all of the material properties were directly given to me by Prof. Guillermo Villanueva with 0.0.0 angles using the ZXZ Euler angles convention. Moreover, only the linear dependency with temperature was considered such that equation 1 becomes: $C_{ij} = C_{0ij} \cdot (1 + Tc_{ij}^{(1)}(T - T_0))$.

2 Matrix rotation and optimal properties

2.1 AT-cut Quartz

2.1.1 Principle and algorithm

Once the values for the 0.0.0 quartz were determined, the angles of the ZXZ Euler angles convention for which we obtain AT-cut quartz like behaviour (2) needed to be found. In the case of the quartz, three resonating modes were investigated: f1 mode, f4 mode and f6 mode. The first one corresponds to a longitudinal mode in which we observe an horizontal expansion along the x-axis. The f4 and f6 correspond to face-shear and thickness shear modes respectively. For the later modes, the displacement should be mainly along the third axis (z) and the second axis (y) respectively. The matter of displacement modes is described in more details in the Comsol simulation part (3) of this report.

The resonance frequency of a resonator with given dimensions can be approximated as $f_{res} = \frac{1}{2W} \cdot \sqrt{\frac{E}{\rho}}$ were W is either the thickness or the pitch of the resonator (depending on the resonating mode), E is the Young's modulus of the material and ρ the density. This expression is derived from the speed v of the acoustical wave going through a resonator $v = \lambda f = \sqrt{\frac{E}{\rho}}$ where $\lambda = 2W$ is the wavelength of the wave.
For the mathematical simulations carried out on MATLAB, the resonance frequencies of the three different modes were approximated by applying to each particular case the aforementioned approximation such that:

$$f_1 = \frac{1}{\sqrt{\rho} \cdot 2W} \cdot \sqrt{c_{11} - \frac{c_{13}^2}{c_{33}}} \tag{2}$$

$$f_4 = \frac{1}{2W} \cdot \sqrt{\frac{c_{44}}{\rho}} \tag{3}$$

$$f_6 = \frac{1}{2W} \cdot \sqrt{\frac{c_{66}}{\rho}} \tag{4}$$

where the coefficients c_{ij} are the values of the stiffness matrix corresponding to the indexes. Those coefficients vary depending on the temperature and on how the matrix is rotated.
The change in resonance frequency in ppm at a temperature T with respect to a reference temperature T_0 can then be computed as :

$$\Delta f_i = \frac{f_i(T) - f_i(T_0)}{f_i(T_0)} \cdot 10^6, \qquad i = 1, 4, 6. \tag{5}$$

where T_0 is equal to 25°C for all of the simulations.

The principle of the first iteration of the MATLAB algorithm is the following:

- Input the 0.0.0 stiffness matrix and temperature dependence matrices of 1st, 2nd and 3rd order.
- Sweep the angle θ and the in-plane angle (ψ) of the ZXZ Euler angles convention (1) between -90 and 90 degrees.
- For each angle, the matrices are rotated **separately** using the algorithm provided by Silvan Stettler and then summed up for each temperature of the temperature range [-40, 100]°C.
- The parameter we look to minimise is the norm of the Δf_i previously defined over the temperature range.
- Once the θ angle that minimises the norm is found, the norm is plotted against the in-plane angle to determine manually which one is the best.

This first iteration wasn't successful since even the smallest variation found was of the order of 2850ppm over a temperature range of 140°C. (4).

The mistake made in that previous algorithm was that the stiffness matrix and its temperature variations coefficients matrices were rotated separately and then summed up for each temperature. In the second iteration (3), the matrices are added for each temperature before applying the different rotations. The rest of the algorithm is similar to the first iteration.

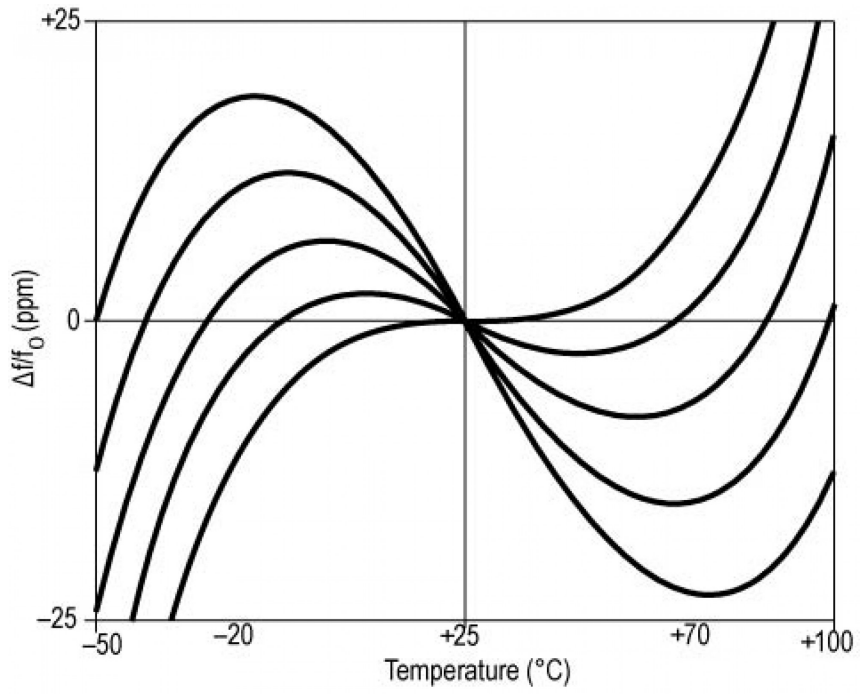

Figure 2: Typical frequency shift with temperature of AT-cut quartz

```
for i = 1:141
Temp=i-41;

new_C(:,:,i) = C0*10^-9.*(ones(6,6) + dC0_T*10^-6*(Temp-T0) + ...
    dC0_T2*10^-9*(Temp-T0)^2 + dC0_T3*10^-12*(Temp-T0)^3);

end

a_inplane = linspace(0, 180, 181);
phi_cut = 0; theta_cut = 35; psi_cut = 0;
in_plane_angle =0;

for u = 1:181
    theta_cut = u-91;
    R_cut = rotation_matrix_zxz(phi_cut, theta_cut, psi_cut);

    for i_a = 1:length(a_inplane)
        a_inpl = a_inplane(i_a);
        R_inplane = rotation_matrix_zxz(a_inpl, 0, 0);
        R = R_inplane*R_cut;
        M = bond_matrix(R);

        c_rot = zeros(6,6,141);
        c_t0 = M * C0 * M' * 10^-9;

        for j = 1:141
            c_rot(:,:,j) = M * new_C(:,:,j) * M';
            delta_f4(j) = (sqrt(c_rot(4,4,j))-sqrt(c_t0(4,4)))/sqrt(c_t0(4,4));
            delta_f6(j) = (sqrt(c_rot(6,6,j))-sqrt(c_t0(6,6)))/sqrt(c_t0(6,6));
            f11 = sqrt(c_rot(1,1,j) - (c_rot(1,3,j)^2)/c_rot(3,3,j));
            f01 = sqrt(c_t0(1,1) - (c_t0(1,3)^2)/c_t0(3,3));
            delta_f1(j) = (f11-f01)/f01;
        end

        f_norm(i_a) = norm(delta_f6);
        if(i_a == 9)

            C_keep = c_rot;
            C0_keep = c_t0;


            hold off
            plot(T, delta_f6*10^6);
            ylabel("(f-f0)/f0 [ppm]")
            xlabel("Temperature [degrees]")
            title('\Deltaf_6/f_{06} for 0.35.82 rotation')
        end
    end
    f_min(u) = min(f_norm);
end
[minim , index] = min(f_min)
plot(a_inplane, f_norm)
```

Figure 3: Extract of the MATLAB code for the 2nd iteration of the algorithm (the full MATLAB code can be found in A.2)

2.1.2 Results

Here is a plot of the smallest Δf_i vs Temperature for the first iteration of the algorithm:

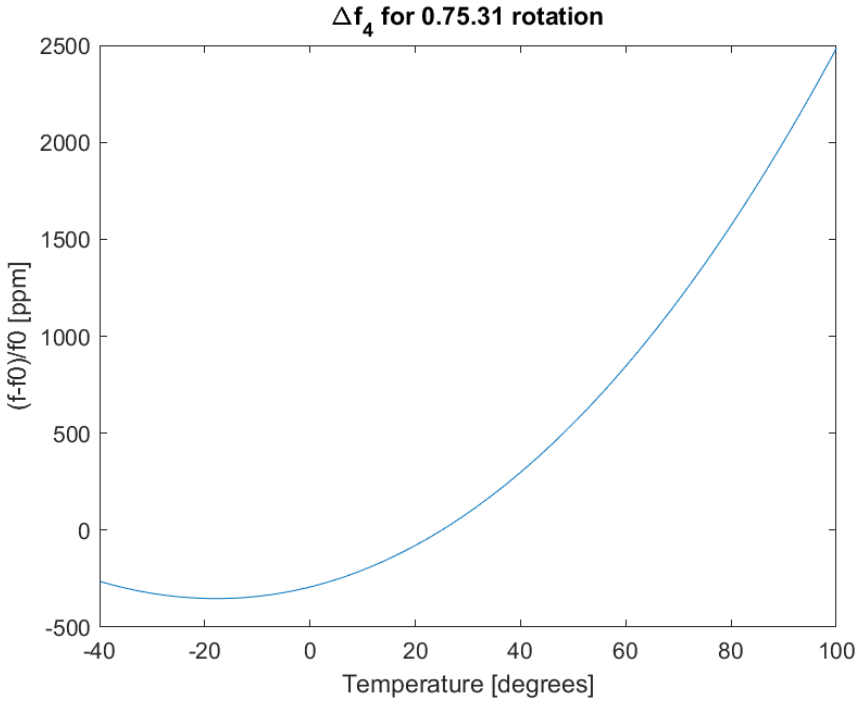

Figure 4: Plot of Δf_4 against the temperature for a rotation of 0.75.31. (1)

As we can see, this correspond to a variation of roughly 20.36 ppm/°C which is much more than the results found in literature for AT cut Quartz. Moreover, the angles 0.75.31 are far from any AT-cut defined in literature.

In the case of the second iteration of the algorithm, three theta angles (one for each mode) minimising the norm were obtained : 35° for f6 mode, -56° for f4 mode and 39° for f1 mode. The in-plane angles (ψ) for which the norm is minimal were measured from figure 5 for each case. The three optimised sets of angles are thus : 0.35.82, 0.-56.85 and 0.39.3. The corresponding Δf vs T plots were then compared to the expected AT-cut quartz behaviour (2). As we can see in figure 6, the f1 mode is the less interesting one as it has no inflexion point (which is expected for AT-cut) and varies more than the other ones. The f4 and f6 modes both look correct but the f4 increases significantly around 100°C and doesn't correspond to the expected $\theta = 35°$ angle of the AT-cut Quartz. Thus, the mode that seemed the more promising for further simulations in Comsol is the f6 mode with Euler angles 0.35.82.

Once the set of angles was chosen, the piezoelectric and relative permittivity tensors as well as their first temperature coefficients were rotated similarly.

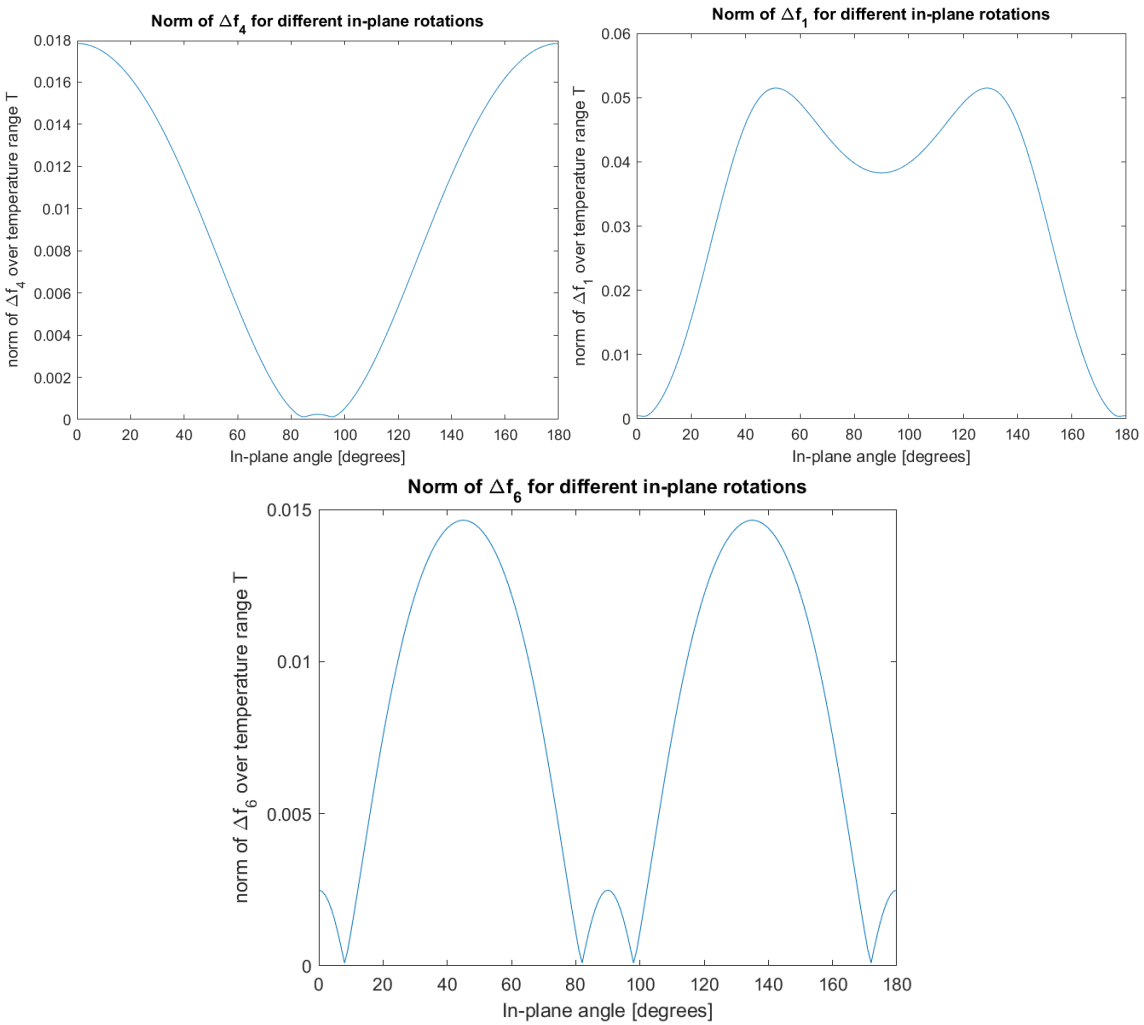

Figure 5: The three plots represent the variation of the norm of Δf_i over a temperature range [-40,100]°C with respect to the in-plane rotation angle. On the upper left: f4 mode with θ = -56°. On the upper right: f1 mode with θ = 39°. At the bottom: f6 mode with θ = 35°.

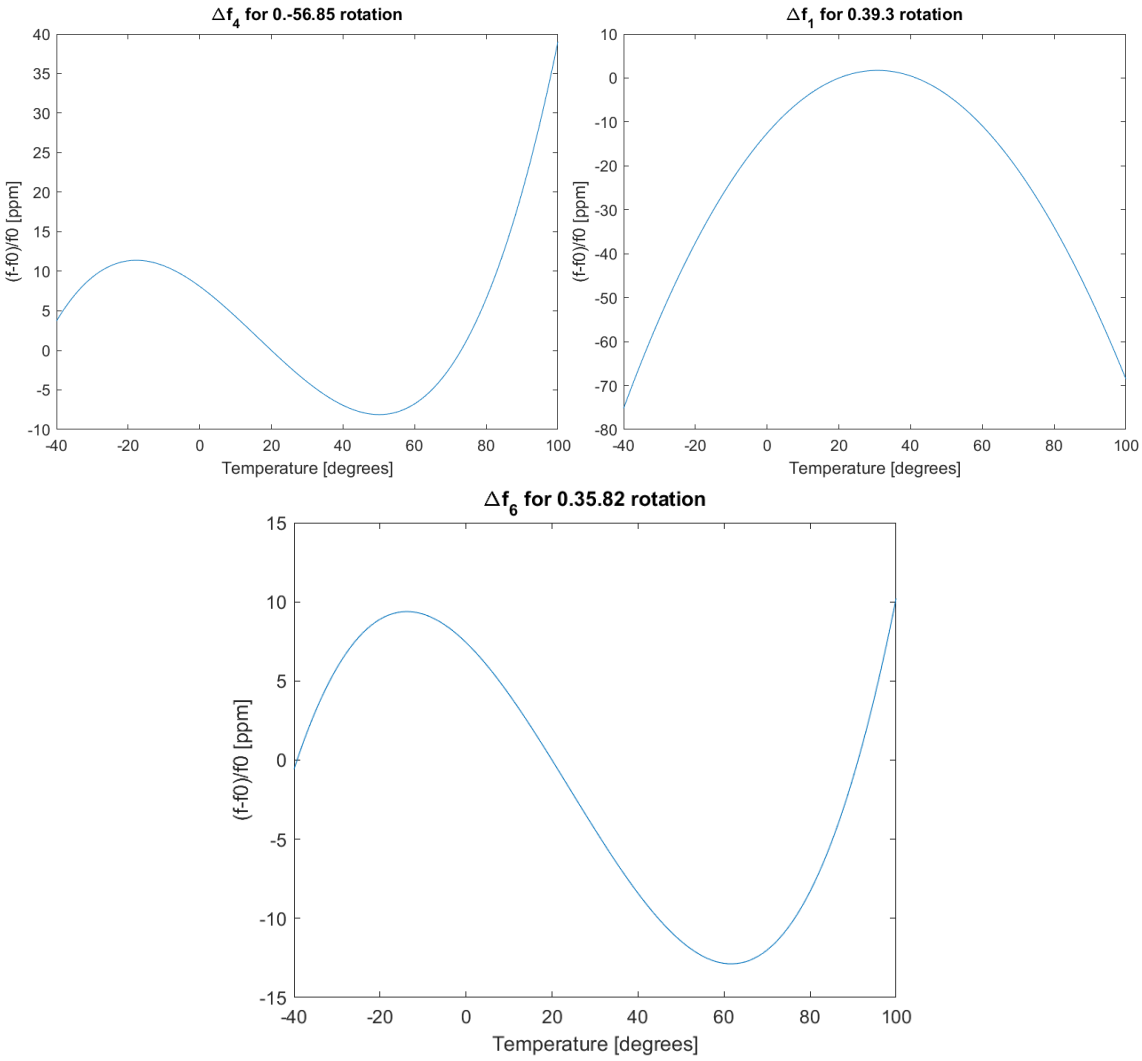

Figure 6: Δf_i over a temperature range [-40,100]°C for optimised angles. On the upper left: Δf_4 with 0.-56.85. On the upper right: Δf_1 with 0.39.3. At the bottom: Δf_6 with 0.35.82.

2.2 Silicon

The silicon optimisation was quite straightforward: the goal was to compare the variations in resonance frequency for the different doping levels (which correspond to different stiffness coefficients and temperature dependence of those coefficients) used in the paper "Determination of Doping and Temperature-Dependent Elastic Constants of Degenerately Doped Silicon From MEMS Resonators" (1.2) and determine which one is the most appropriate one knowing that it will be compensated by Lithium Niobate (see 2.3). The mode we consider is the f_1 mode as defined in 2 except that, for silicon, $c_{33} = c_{11}$.
The Δf_1 curves for each doping level were plotted on Matlab over the temperature range [-40,100]°C. We want the resonance frequency to increase when the temperature increases because the Lithium Niobate shows opposite behaviour (this way the two material variations will compensate). As can be seen on figure 7, the most appropriate doping level for our application is the phosphorus doping of carrier concentration $7.5 \cdot 10^{19} cm^{-3}$. It is interesting to note that the carrier concentration investigated are only the ones from the paper. However, we can clearly see that, for As and P dopants, the more we increase the carrier concentration, the more the shape of the curve tends to have a positive slope. Thus, we can infer that using even higher carrier concentration would be an even better solution as the curve would probably be increasing over the whole temperature range instead of diminishing back around 80°C.

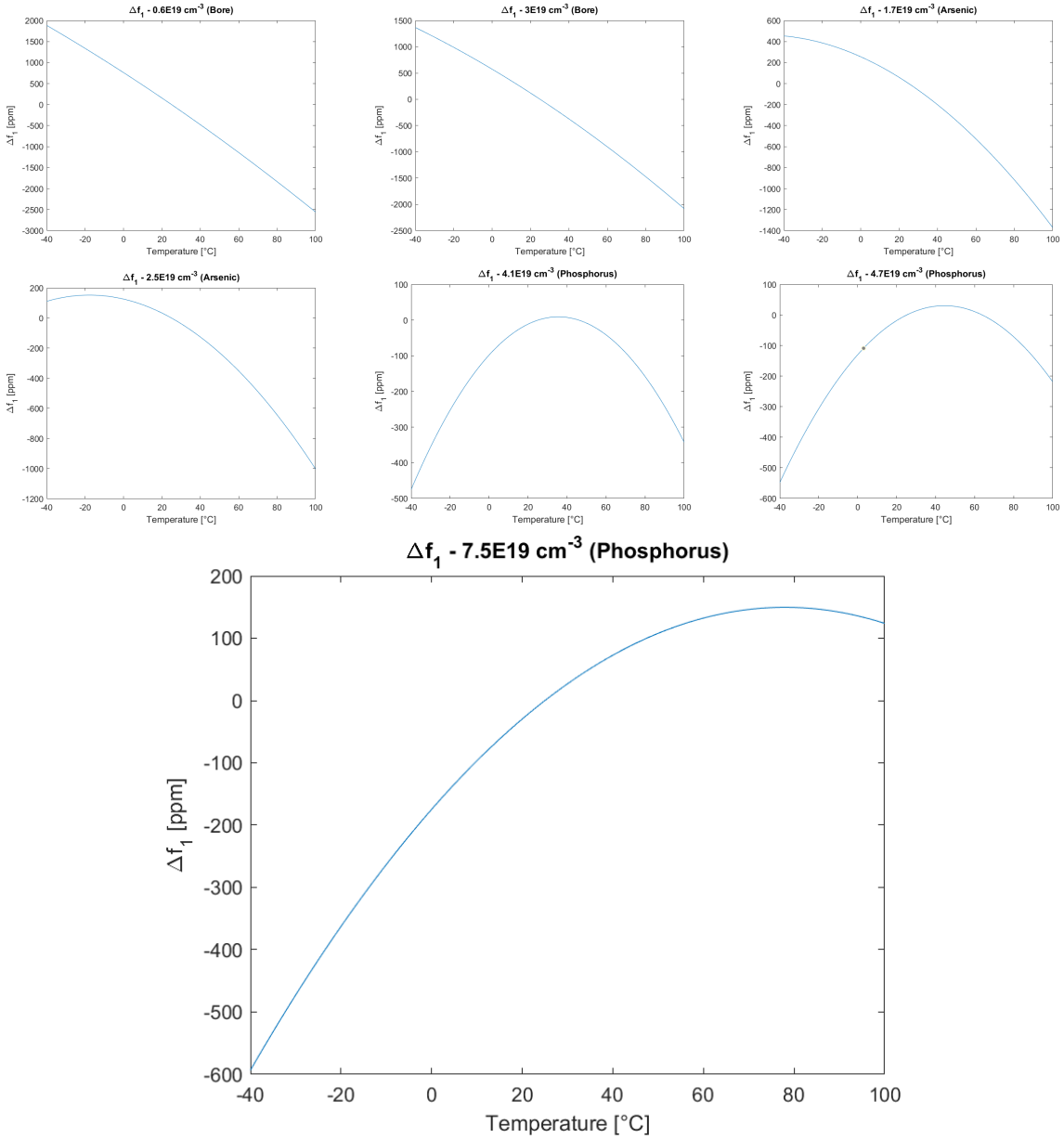

Figure 7: Δf_i over a temperature range [-40,100]°C for doped silicon with different carrier concentrations.

2.3 Lithium Niobate

For the rotation angles of the Lithium Niobate, two simple sets proposed by Prof. Guillermo Villanueva were explored: 0.38.90 and 0.-90.0. The algorithm used to make those rotations is the same as in 2.1.1 although there is no need for sweeping this time as we already know the Euler angles to use. As the Silicon and the Lithium Niobate will form one beam, we consider the same mode as for silicon: f1. The resulting Δf_1 obtained for the two sets of angles are shown in figure 8. As the variation for the angles 0.-90.0 seems orders of magnitude too small (roughly 10^{-2} ppm over the whole temperature range) to compensate for the variations of the silicon layer, the logical choice is to select the other set of angles (0.38.90) for which the variation in resonance frequency is of the same order of magnitude as for Si. The computed material properties matrices are available in A.3.

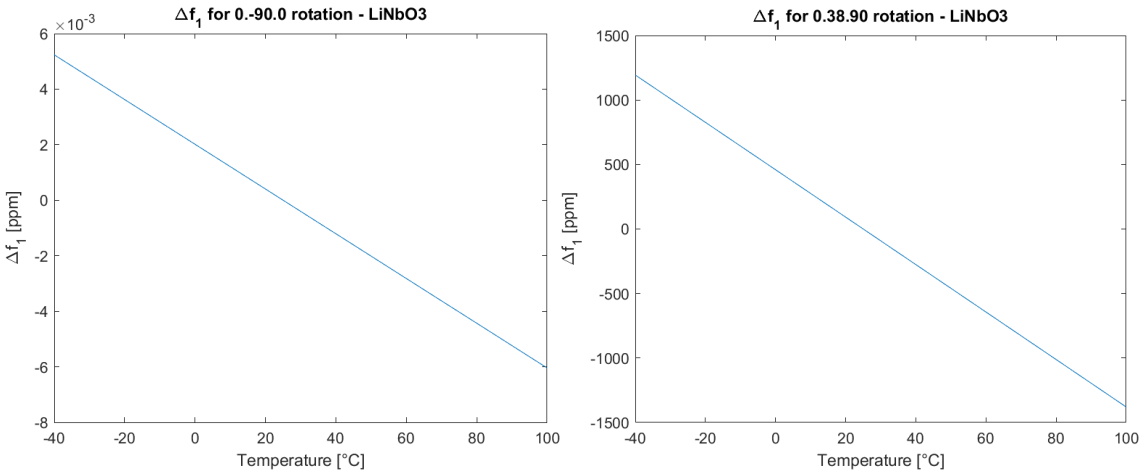

Figure 8: Δf_i of Lithium Niobate over the temperature range [-40,100]°C for 2 sets of angles. On the left: Δf_1 with 0.-90.0. On the right: Δf_1 with 0.38.90

2.4 Retrieving the material properties

In the 2nd iteration of the algorithm, we obtained one 6 by 6 matrix (for the stiffness) for each temperature T (141 different temperatures in this case). For each matrix, the coefficients are determined according to the equation 1. The matrices are then all rotated and we thus don't have separated matrices for the stiffness coefficients and for the temperature dependence coefficients. This means that the first, second and third order temperature dependence coefficients must be estimated from the coefficients of the 141 rotated matrices. Practically, for each term C_{ij} of the 6 by 6 matrix, we need to approximate the coefficients p_1, p_2, p_3 and p_4 such that :

$$C_{ij} = p_1 \cdot T^3 + p_2 \cdot T^2 + p_3 \cdot T + p_4 \tag{6}$$

This is done using the MATLAB function fit() as can be seen in figure 9. An example of the fitted curve obtained through this method is given in A.3 for the C_{66} coefficient of the 0.35.82 rotated Quartz.

```
for i = 1:6
    for j = 1:6
        c = reshape(C_keep(i,j,:), [141,1]);
        i,j
        curvefit = fit(T', c, 'poly3')
    end
end
```

Figure 9: Code used to approximate the stiffness matrix terms

This method was used to retrieve the stiffness, piezoelectric and permittivity matrices and their temperature dependence coefficients after rotation for Quartz and Lithium Niobate as can be seen in the code in A.2.

3 Comsol simulations

Only 2D Comsol simulations were carried out during this semester project for which the axis were defined similarly to the ZXZ Euler angles convention described previously: the x-axis towards the right, the z-axis up (perpendicular to the plane of the resonator) and the y-axis as the out-of-plane axis (coming out of the screen).

3.1 Quartz

3.1.1 Geometry

Once the material properties were computed, they could then be inserted into Comsol as a new material for further simulations. The geometry of the set-up for the simulation was determined by looking at the coupling (piezoelectric) matrix at reference temperature (25°C). Since we are looking to excite the thickness-shear mode f6, we choose the larger coefficient of the 6^{th} column of the 3x6 matrix which is e_{16} according to figure 10.

-0.0465	0.0490	-0.0025	-0.1042	0.0240	0.0849
0.1415	-0.1594	0.0180	-0.0240	-0.0631	0.0570
0.0185	-0.0185	0	0.0746	-0.0105	-0.0644

Figure 10: Values of the coupling matrix of 0.35.82 rotated quartz at reference temperature of 25°C

Considering this e_{16} coefficient, a longitudinal voltage along the x-axis of the chosen coordinate system need to be applied to have the largest thickness-shear frequency response for this quartz resonator. This electric potential is applied using the geometry of figure 11 in which the two sides electrodes and the middle one are at a 1V potential and the two other electrodes are connected to the ground. Of course, in Comsol we could simply apply a potential on the right side of the resonator and put the left side to the ground but this isn't a realistic approach as at least one of this side would be clamped in reality. Moreover, in practice 1-port configurations are more sensitive to parasitic noise.
The dimensions chosen for the Quartz simulation are the following: a length of 60um, a thickness of 0.5um, a width (out-of-plane) of 150um and a pitch of 15um between the electrodes.

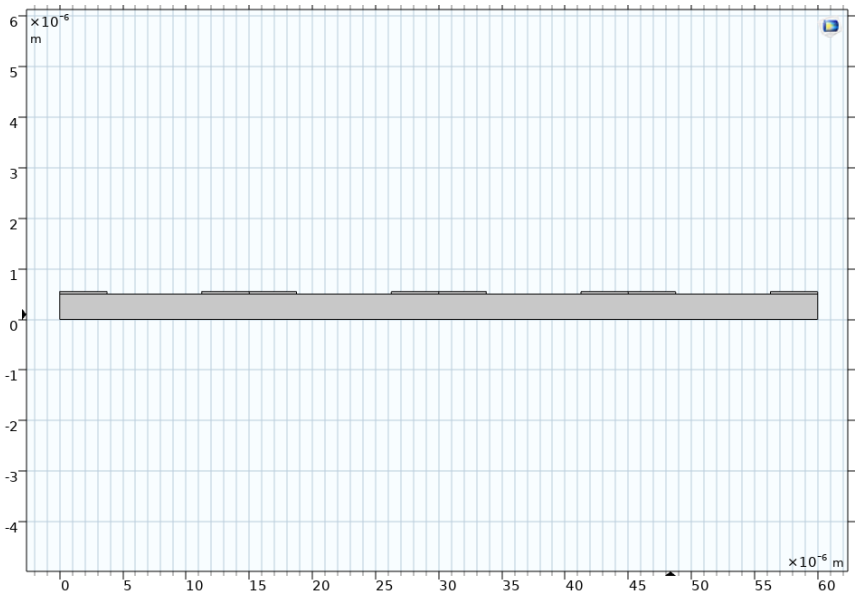

Figure 11: Geometry of the quartz resonator for the Comsol simulation

3.1.2 Simulation and results

Once the geometry of the simulation had been defined, the mechanical and electrical boundary conditions were chosen. A condition that is worth mentioning is the fact that all boundaries are

defined as mechanically free in order to simplify the simulations.
In order to determine around which frequency the simulation should be computed, a resonance frequency estimation needs to be done by hand using equation 4 with the following parameters: $c_{66}(T_0) = 31.2116$ GPa, W = 15um and $\rho = 2648\frac{kg}{m^3}$. The estimated resonance frequency obtained is $f_6 = 114,44MHz$. A large frequency domain study was thus done in the range [100MHz, 500MHz] with a step size of 0.05MHz yielding the absolute impedance graph depicted in figure 12.

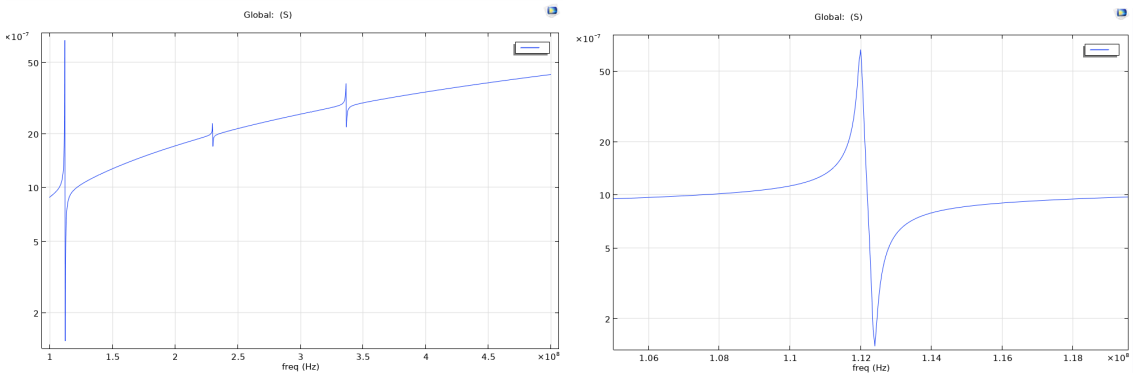

Figure 12: Absolute admittance of Quartz from Comsol simulation. On the left: 100MHz to 500MHz. On the right: zoom onto the 112MHz resonance frequency

A clear frequency peak can be observed at 112MHz which corresponds closely to the 114.44MHz estimation. This let us suppose that this indeed corresponds to the f6 mode. When looking at the displacement field in each direction (figure 13), a clear displacement can be observed along the x-axis and the z-axis for which the amplitude of the displacement is approximately one order of magnitude larger than along x. Indeed, the maximum displacements along x and z are $4.12 \cdot 10^{-11}$m and $3.88 \cdot 10^{-10}$m respectively. A maximum displacement of $1.34 \cdot 10^{-12}$m can also be observed along the y-axis but it is negligible.
The axis along which the maximum displacement was expected is the y-axis (out-of-plane axis as defined at the beginning of this section) as it should correspond to the thickness shear f6 mode. A possible explanation for the fact that Comsol shows a maximum displacement along the so-called "Z component" instead of "Y component" is that it actually refers to the default coordinate system of Comsol for which the y and x axes are inverted compared to the coordinate system defined previously. This statement was verified by using the same geometry but changing the coupling matrix so that only the e_{16} coefficient is different than 0. This change allows to make sure that only the f6 mode is excited. The displacement fields obtained from this verifying simulation are as expected: similar to the previous simulation with complete coupling matrix (results are shown in A). Thus, the resonance frequency of 112MHz does correspond to the f6 mode.

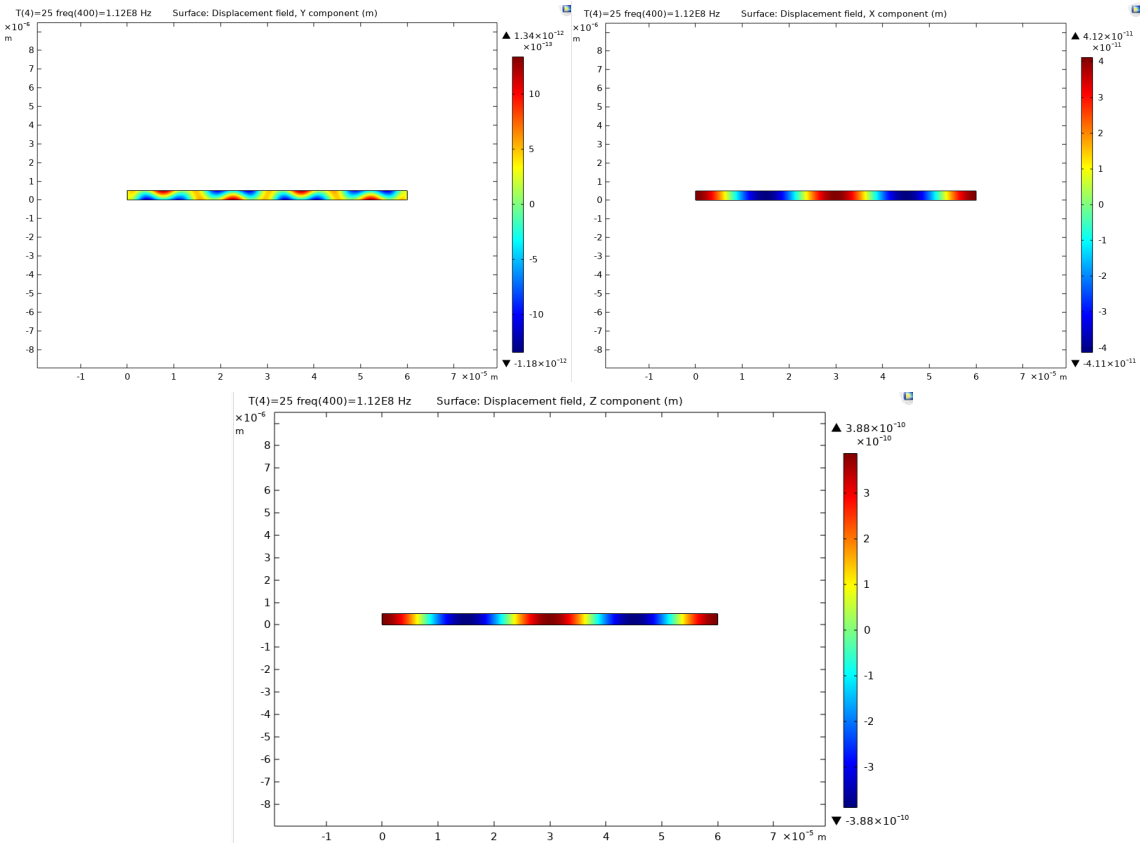

Figure 13: Displacement field in the x (top right), y (top left) and z (bottom) directions - Quartz Comsol simulation

A second frequency domain study was carried out over the temperature range [-50, 100]°C to determine the resonance frequency shift with temperature. Ideally, the frequency shift should be similar to the approximated one shown in figure 6. The resulting frequency shift over the temperature range is represented in figure 14. It is clear that it doesn't correspond to the expected behaviour as the resonance frequency is linearly increasing with a shift of 11.3ppm/°C.
This is probably due to the fact that, when simulating with Matlab, the f6 mode is approximated as depending only on the c_{66} coefficient of the stiffness matrix of quartz but in Comsol, all coefficients are taken into account and thus we don't have a pure f6 mode. Other simulations where thus conducted to determine which other mode is interfering. All the temperature coefficients (1st, 2nd and 3rd degree) of one stiffness coefficient of the Quartz was set to 0 for each simulation (except c_{66}) in order to find out which were influencing the displacement at 112MHz. It was found that the coefficients responsible for the interference are: $c11$, $c14$, $c15$, $c16$, $c36$ and $c46$. If the temperature coefficients of those indexes are set to 0, the results obtained are much closer to the Matlab approximation as can be seen in figure 15. The maximum frequency shift from the resonance frequency at reference temperature (25°C) is then $\frac{|f(25C)-f(75C)}{f(25C)} \cdot 10^6 = \frac{|f(25C)-f(-25C)}{f(25C)} \cdot 10^6 =$ 17.86ppm which corresponds to 0.36ppm/°C.

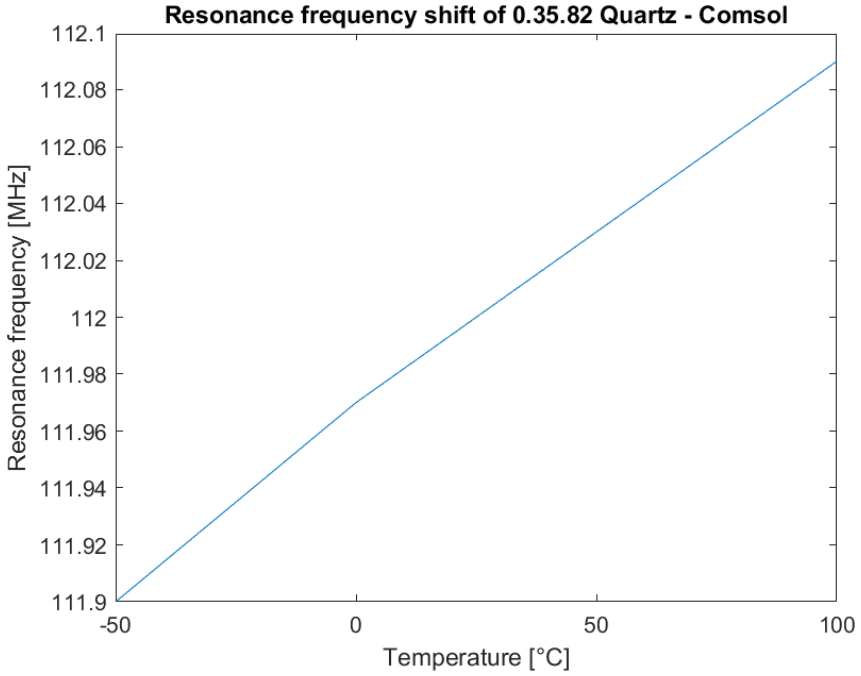

Figure 14: Resonance frequency shift 0.35.82 rotated Quartz - first simulation

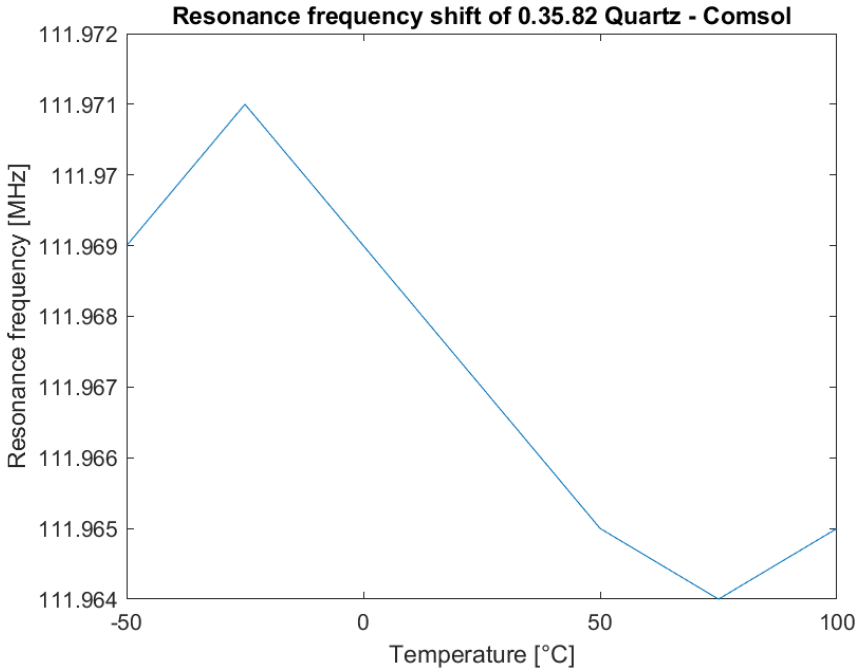

Figure 15: Resonance frequency shift 0.35.82 rotated Quartz without interfering modes

3.2 Silicon

3.2.1 Geometry

The geometry of the silicon simulation was defined similarly as for the quartz. According to the coupling matrix of the Lithium Niobate at reference temperature (figure 16), the larger coefficient that can be used to excite the f1 mode is $e_{11} = 4.5333\ C/m^2$. Consequently, the placement of the electrodes on the Lithium Niobate layer is the same as for Quartz (see figure 17).
The dimensions chosen for the Quartz simulation are the following: a length of 60um, a thickness of 0.5um for the Lithium Niobate layer, a width (out-of-plane) of 150um and a pitch of 15um between the electrodes. For the thickness of the Silicon layer, different values were tried in order to determine which one gave the best frequency shift compensation: 1um, 2um and 5um.

4.5333	-1.7223	-1.3519	0	0.2174	0
0	0	0	4.3977	0	0.3648
-2.4540	1.7263	2.5953	0	0.7352	0

Figure 16: Values of the coupling matrix of 0.38.90 rotated LiNb at reference temperature of 25°C

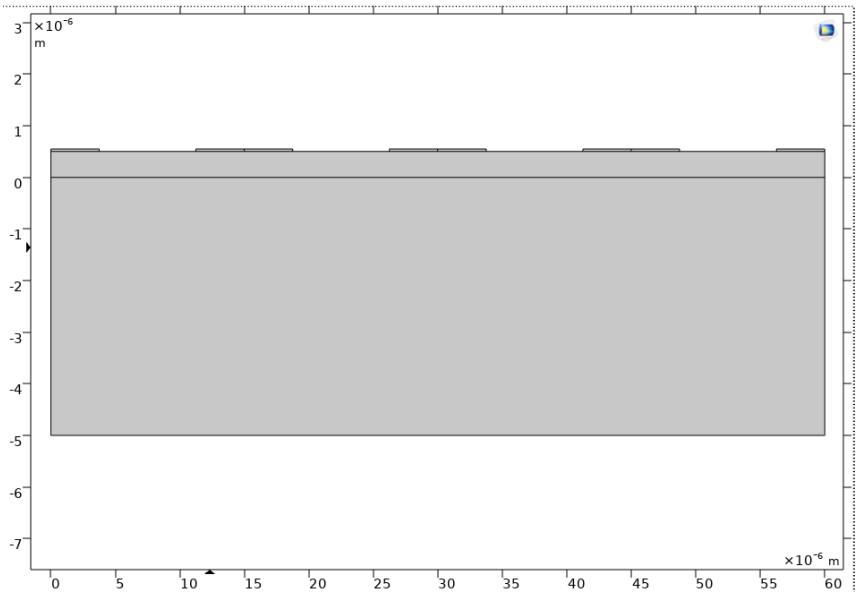

Figure 17: Geometry of the silicon resonator for a silicon thickness of 5um

3.2.2 Simulation and results

The goals of the simulations for each silicon thickness were to:

- Determine the resonance frequency at which we can observe the f1 mode.
- Compute the coupling coefficient at this resonance frequency.
- Sweep over the temperature range [-50, 100]°C to observe the frequency shift with respect to temperature.

As the geometry is a bit more complex since we have 2 layers of different materials, it was not possible to calculate by hand an estimation of the resonance frequency using the equations of section 2.1.1 as for Quartz. The mode shape of several resonance frequencies were thus investigated to determine which one corresponded the best to the f1 mode.
The coupling coefficient k_T^2 can be calculated using the equation:

$$k_T^2 = \frac{\pi^2}{4} \cdot \frac{|f_{res} - f_{anti-res}|}{f_{res}} \tag{7}$$

where f_{res} and $f_{anti-res}$ are the resonance frequency and anti-resonance frequency respectively and the result is a percentage of electrical energy converted into mechanical energy.

The impedance graph representing the f1 mode resonance frequency as well as the corresponding displacement field in x for each silicon thickness are shown in figure 18. In each case, the mode shape (displacement field in x, y and z) of several resonance frequencies were studied before finding the appropriate resonance frequency: a dominating lateral displacement along the x axis with minimum displacement in the two other directions. Practically, for the frequencies depicted in figure 18, the maximum displacement along the x axis is 10 times larger than along the y axis and there is no displacement along the z axis.

The frequency domain study were done over the temperature range [-50, 100]°C with a step of 25°C meaning that the simulation was run for 7 different temperatures. The frequency shift with respect to temperature for each of the three thicknesses are represented in 19. Using the values from this graph as well as the following equation : $\Delta f = \frac{f_{max} - f_{min}}{f_{max})} \cdot 10^6$ which is similar to equation 5, the

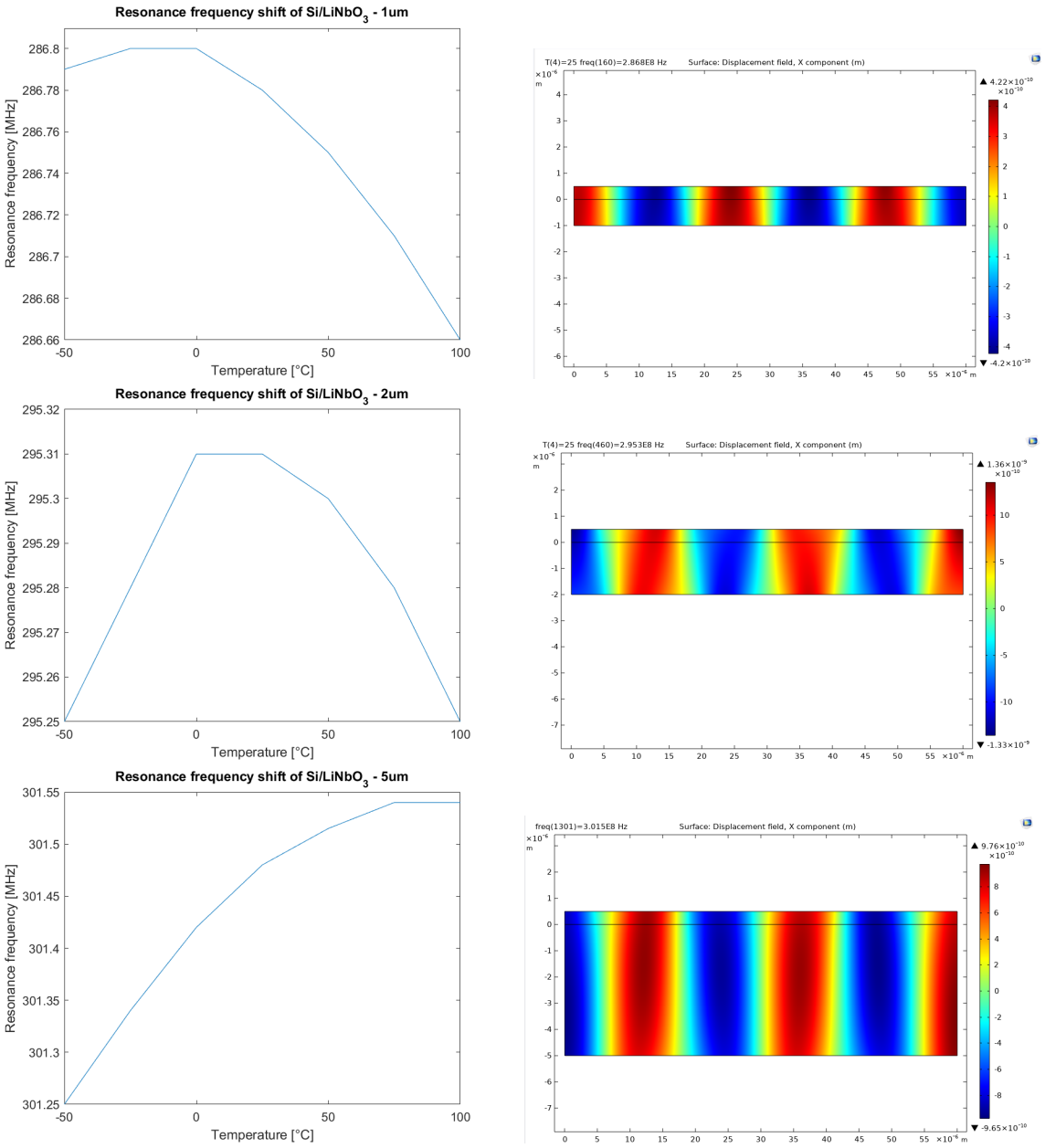

Figure 18: Resonance frequency of the f1 mode with corresponding displacement field in x for each silicon layer thickness.

maximum change in resonance frequency in ppm over the temperature range can be calculated as well as the corresponding change of resonance frequency per temperature unit.
For each Si thickness, the resonance frequency at which we observe the same f1 mode, the coupling coefficient, the maximum change in resonance frequency and the change of resonance frequency per temperature unit are summed up in table 1.

Thickness of Si layer	1um	2um	5um
Resonance frequency at T_0 [MHz]	286.8	295.3	301.5
Anti-resonance frequency [MHz]	291.95	299.25	303.17
Coupling coefficient (in %)	44.3	33	13.7
Maximum change in resonance frequency [ppm]	488	203	961.7
Change of resonance frequency per °C [ppm/°C]	4.88	2.7	6.4

Table 1: Results of the Comsol simulations of Si/LiNbO_3

As can be observed from the results, the thickness for which the change of resonance frequency (295.3 MHZ at reference temperature) is minimal is 2um: only -2.7ppm/°C over the temperature

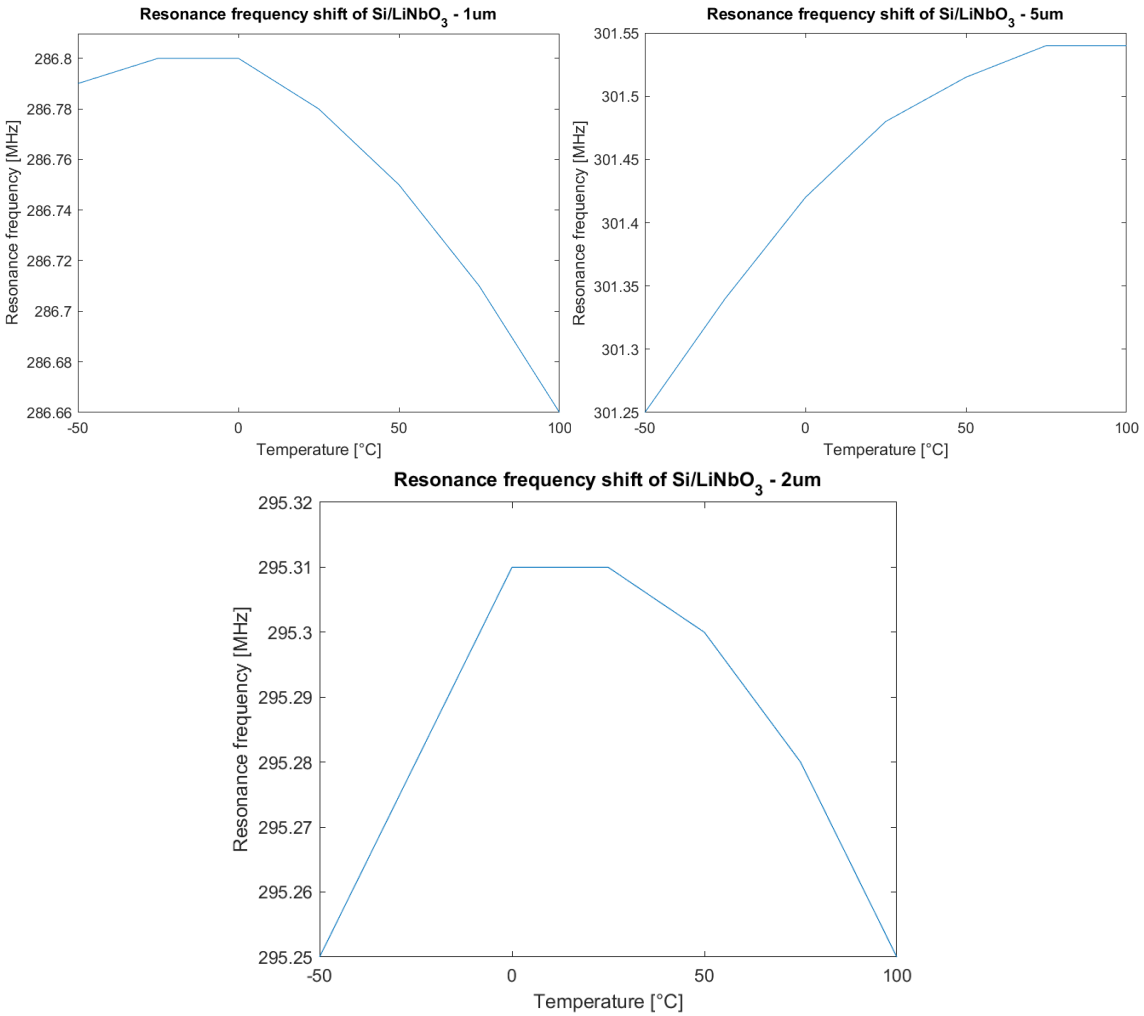

Figure 19: Geometry of the quartz resonator for the Comsol simulation

range [25, 100]°C and +2.7ppm/°C over the temperature range [-50, 25]°C. Moreover, the coupling coefficient of 33% ensures a good energy conversion with limited losses.

4 Conclusion

In the case of the Quartz, it was found in this semester project that the rotation angles that minimise the resonance frequency shift due to temperature change are: $\phi = 0°$, $\theta = 35°$ and $\psi = 82°$ following the x-convention of the Euler angles. Using the properties of this rotated quartz for the Comsol simulation, a minimum frequency shift of 0.36ppm/°C over the temperature range [-50, 100]°C was found for an f6 actuation mode (thickness shear at 112MHz - reference temperature) without considering the stiffness matrix coefficients responsible for interfering modes. However, it will be necessary to find a solution to get rid of those interfering modes if this piezoelectric actuator is to be fabricated in reality. For the hybrid Silicon/Lithium Niobate resonator, a resonance frequency shift of less than 2.7ppm/°C was obtained for a 2um thick Silicon layer actuated via a 0.5um thick piezoelectric Lithium Niobate layer with rotation 0.38.90 (Euler angles). This result corresponds to an f1 actuation mode (longitudinal) of resonance frequency of 295.3 MHz at 25°C. The resonance frequency shift obtained for silicon is correct but it could be even lower if the optimal silicon thickness was used instead of picking the best thickness between 3 possibilities (1um, 2um and 5um). Moreover, taking into account the 2nd order temperature coefficients of the stiffness matrix of Lithium Niobate (instead of just the linear coefficients) could result in a much better shift compensation between the two materials.

References

[B A73] B. A. Auld. Acoustic Fields and Waves in Solids. 1973.

[Bec] Bechmann, R. and Ballato, A. D. and Lukaszek, T. J. "Higher-Order Temperature Coefficients of the Elastic Stiffnesses and Compliances of Alpha-Quartz". In: Proceedings of the IRE 50 8 (). DOI: https://doi.org/10.1109/JRPROC.1962.288222.

[Jaa] Jaakkola Antti, Mika Prunnila, Tuomas Pensala, James Dekker, Panu Pekko. "Determination of Doping and Temperature-Dependent Elastic Constants of Degenerately Doped Silicon from MEMS Resonators". In: IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Contr 61 (). DOI: https://doi.org/10.1109/TUFFC.2014.3007.

[R D89] R. D. Mindlin and D. Gazis. Strong resonances of rectangular AT-cut quartz plates. 1989.

[War] Ward, R.W. "The Constants of Alpha Quartz". In: 38th Annual Symposium on Frequency Control (). DOI: 10.1109/FREQ.1984.200732.

[Wei] Eric W Weisstein. Euler angles. URL: https://mathworld.wolfram.com/EulerAngles.html. From MathWorld.

A Appendix

A.1 Material properties

```
C0 = 6×6
10^11 ×
    0.8674    0.0699    0.1191   -0.1791         0         0
    0.0699    0.8674    0.1191    0.1791         0         0
    0.1191    0.1191    1.0720         0         0         0
   -0.1791    0.1791         0    0.5794         0         0
         0         0         0         0    0.5794   -0.1791
         0         0         0         0   -0.1791    0.3988
```

Figure 20: Quartz stiffness matrix 0.0.0 (units: Pa)

TABLE 4
ELASTIC STIFFNESSES $c_{\lambda\mu}$ IN 10^9 Nm^{-2} FOR ALPHA-QUARTZ AT 20°C

$\lambda\mu$	STIFFNESSES OF QUARTZ Bechmann (9)		STIFFNESSES OF QUARTZ Mindlin and Gazis (15)
	$c_{\lambda\mu}^{E\sigma}$	$c_{\lambda\mu}^{D\sigma}$	$c_{\lambda\mu}$
11	86.74	87.49	86.75
33	107.2	107.2	107.2
12	6.99	6.23	5.95
13	11.91	11.91	11.91
44	57.94	57.98	57.8
66	39.88	40.63	40.4
14	−17.91	−18.09	−17.8

TABLE 7
VALUES FOR THE FIRST-ORDER TEMPERATURE COEFFICIENTS OF THE STIFFNESSES FOR ALPHA-QUARTZ IN 10^{-6}/°C

	BECHMANN [5] 1934	MASON [19] 1940	ATANASOFF [20] AND HART 1941	MASON [22] 1951	GPO [23] RESEARCH REPORT No. 13524 1951	KOGA [24] et al. 1958	NEW VALUES 1961
		T_o = 20°	T_o = 35°	T_o = 50°	T_o = 20°	T_o = 20°	T_o = 25°
$Tc_{11}^{(1)}$	−48	−54	−49.7	−53.5	——	−44.3	−48.5
$Tc_{33}^{(1)}$	−208	−251	−213	−165	——	−188	−160
$Tc_{12}^{(1)}$	−2115	−2350	−3000	−3030	——	−2930	−3000
$Tc_{13}^{(1)}$	−530	−687	−580	−510	——	−492	−550
$Tc_{44}^{(1)}$	−151	−160	−169	−171	−169	−172	−177
$Tc_{66}^{(1)}$	144	161	170.1	168	168	180	178
$Tc_{14}^{(1)}$	82	96	107	90	94	98	101

VALUES FOR THE SECOND-ORDER TEMPERATURE COEFFICIENTS OF THE STIFFNESSES FOR ALPHA-QUARTZ IN 10^{-9}/(°C)^2

	MASON [22] 1951	GPO [23] RESEARCH REPORT No. 13524 1951	KOGA, et al. [24] 1958	NEW VALUES 1961
	T_o = 50° C	T_o = 20° C	T_o = 20° C	T_o = 25° C
$Tc_{11}^{(2)}$	−75	——	−407	−107
$Tc_{33}^{(2)}$	−187	——	−1412	−275
$Tc_{12}^{(2)}$	−1500	——	−7245	−3050
$Tc_{13}^{(2)}$	−2000	——	−596	−1150
$Tc_{44}^{(2)}$	−212	−233	−225	−216
$Tc_{66}^{(2)}$	−5	193	201	118
$Tc_{14}^{(2)}$	−270	−82	−13	−48

VALUES FOR THE THIRD-ORDER TEMPERATURE COEFFICIENTS OF THE STIFFNESSES FOR ALPHA-QUARTZ IN 10^{-12}/(°C)^3

	MASON [22] 1951	KOGA et al [24] 1958	NEW VALUES 1961
	T_o = 50° C	T_o = 20° C	T_o = 25° C
$Tc_{11}^{(3)}$	−15	−371	−70
$Tc_{33}^{(3)}$	−410	−243	−250
$Tc_{12}^{(3)}$	1910	4195	−1260
$Tc_{13}^{(3)}$	600	−5559	−750
$Tc_{44}^{(3)}$	−65	−190	−216
$Tc_{66}^{(3)}$	−167	−777	21
$Tc_{14}^{(3)}$	−630	−625	−590

Figure 21: Stiffness coefficients of non-rotated quartz and its temperature coefficients retrieved from [Bec]

```
D0 = [3.92 0 0; 0 3.92 0; 0 0 4.1] * 8.854e-12;
dD_T = [
    28.0   0.00    0.00
    0.00   28.0    0.00
    0.00   0.00    39.0
    ];
```

Figure 22: Quartz permittivity matrix and its first order temperature coefficients

```
D0 = [3.92 0 0; 0 3.92 0; 0 0 4.1] * 8.854e-12;
dD_T = [
    28.0    0.00    0.00
    0.00    28.0    0.00
    0.00    0.00    39.0
    ];
```

Figure 23: Quartz piezoelectric matrix and its first order temperature coefficients

$$\begin{bmatrix} c_{11} & c_{12} & c_{12} & & & & \\ c_{12} & c_{11} & c_{12} & & & & \\ c_{12} & c_{12} & c_{11} & & & & \\ & & & c_{44} & & & \\ & & & & c_{44} & & \\ & & & & & c_{44} & \end{bmatrix}$$

Figure 24: Structure of Silicon's stiffness matrix

TABLE III. VALUES OF TEMPERATURE COEFFICIENTS OF THE ELASTIC PARAMETERS c_{ij}

Dopant	n ($\times 10^{19}$ cm^{-3})	c_{11}^0 (GPa)	a_{11} (ppm/K)	b_{11} (ppb/K^2)	c_{12}^0 (GPa)	a_{12} (ppm/K)	b_{12} (ppb/K^2)	c_{44}^0 (GPa)	a_{44} (ppm/K)	b_{44} (ppb/K^2)	$c_{11}^0 - c_{12}^0$ (GPa)	a_{11-12} (ppm/K)	b_{11-12} (ppb/K^2)
B	3.0	164.2	−62.5	−73	63.5	−102.9	−11	78.5	−19.3	−136	100.7	−37.1	−112
B	0.6	165.5	−69.9	−59	63.7	−90.6	−22	79.6	−51.7	−81	101.8	−56.9	−83
As	1.7	164.0	−48.5	−111	64.3	−114.7	25	79.5	−63.7	−58	99.7	−5.8	−198
As	2.5	163.8	−44.2	−111	64.9	−124.6	34	79.4	−65.1	−55	98.9	8.5	−206
P	4.1	163.0	−34.5	−115	65.4	−133.7	22	79.2	−67.8	−51	97.6	31.9	−206
P	4.7	162.5	−32.5	−110	65.7	−131.8	18	79.1	−68.7	−43	96.8	34.9	−196
P	7.5	161.4	−30.7	−78	66.1	−134.9	−12	78.5	−71.9	−31	95.3	41.6	−124
		Δc_{11}^0	Δa_{11}	Δb_{11}	Δc_{12}^0	Δa_{12}	Δb_{12}	Δc_{44}^0	Δa_{44}	Δb_{44}	$\Delta(c_{11}^0 - c_{12}^0)$ (GPa)	Δa_{11-12} (ppm/K)	Δb_{11-12} (ppb/K^2)
Total uncertainty (rms sum)		±0.3	±0.7	±12	±0.1	±1.5	±33	±0.2	±0.4	±10	±0.2	±0.3	±7
E_1: frequency error		±0.3	±0.39	±9.4	±0.1	±1.20	±28.5	±0.2	±0.16	±3.6	±0.2	±0.16	±3.5
E_2: temperature error			±0.26	±6.6		±0.64	±17.1		±0.34	±9.1		±0.20	±5.6
E_3: linearization point error			±0.54	±0.8		±0.69	±1.1		±0.01	±0.0		±0.02	±0.0
E_4: thermal expansion error			±0.09			±0.09			±0.09			±0.09	
E_5: thickness variation error			±0.02	±0.0		±0.04	±0.0		±0.00	±0.0		±0.00	±0.0

Corresponding data points are plotted in Fig. 8. Confidence intervals Δc_{ij}^0, Δa_{ij}, and Δb_{ij} are based on the error analysis of Section V-D.

Figure 25: Silicon's stiffness coefficients (c_{11}, c_{12} and c_{44}) and their 1st and 2nd order temperature coefficients for different doping levels (extracted from [Jaa])

A.2 MATLAB code

Quarz Material Properties sweep

```
% Material properties as matrices
%---------------------------------------------------------------
% Stiffness tensor
%---------------------------------------------------------------

% Trig. 32 -Auld
% THIS TIME USED THE COEFFICIENT FROM THE MILITARY PAPER
C0 = 1e11 * [
    0.8674    0.0699  0.1191   -0.1791   0.0000    0.0000
    0.0699    0.8674  0.1191    0.1791   0.0000    0.0000
    0.1191    0.1191  1.0720    0.0000   0.0000    0.0000
   -0.1791    0.1791  0.0000    0.5794   0.0000    0.0000
    0.0000    0.0000  0.0000    0.0000   0.5794   -0.1791
    0.0000    0.0000  0.0000    0.0000  -0.1791    0.3988
    ]

dC0_T = [
   -48.50   -3000.0  -550.00  +101.00     0.00     0.00
  -3000.0   -48.50  -550.00  -101.00     0.00     0.00
  -550.00  -550.00  -160.00     0.00     0.00     0.00
  +101.00  -101.00     0.00  -177.00     0.00     0.00
     0.00     0.00     0.00     0.00  -177.00  +101.00
     0.00     0.00     0.00     0.00  +101.00  +178.00
    ]

dC0_T2 = [
    -107.00   -3050.00   -1150.00   -48.00     0.00      0.00
    -3050.00   -107.00   -1150.00   48.00     0.00      0.00
   -1150.00  -1150.00    -275.00     0.00     0.00      0.00
     -48.00    48.00       0.00  -216.00     0.00      0.00
       0.00      0.00       0.00     0.00  -216.00    -48.00
       0.00      0.00       0.00     0.00   -48.00    +118.00
       ];

dC0_T3 = [
    -70.00   -1260.00   -750.00   -590.00      0.00      0.00
   -1260.00    -70.00   -750.00   590.00      0.00      0.00
   -750.00   -750.00    -250.00      0.00      0.00      0.00
   -590.00   590.00      0.00   -216.00      0.00      0.00
      0.00      0.00      0.00      0.00   -166.00   -590.00
      0.00      0.00      0.00      0.00   -590.00    21.00
      ];


%---------------------------------------------------------------
% Piezoelectric tensor
%---------------------------------------------------------------

E0 = [
    0.171   -0.171  0.000   -0.0406  0.000    0.000
    0.000    0.000  0.000    0.000   0.0406  -0.171
    0.000    0.000  0.000    0.000   0.000    0.000
```

1

Figure 26: MATLAB code p1

```
    ];

dE_T = [
    -16.0   -16.0   0.00   -144.0    0.00    0.00
   0.00    0.00   0.00     0.00  -144.0   -16.0
   0.00    0.00   0.00     0.00    0.00    0.00
    ];

%--------------------------------------------------------------
% Permittivity tensor
%--------------------------------------------------------------


D0 = [3.92 0 0; 0 3.92 0; 0 0 4.1] * 8.854e-12;
dD_T = [
    28.0   0.00    0.00
    0.00   28.0    0.00
    0.00   0.00    39.0
    ];

%--------------------------------------------------------------
% CTEs
%--------------------------------------------------------------

alfa_T1 = [
    13.71    0.00    0.00
     0.00   13.71    0.00
     0.00    0.00    7.47
     ];

alfa_T2 = [
     6.5   0.0    0.0
    0.0   6.5    0.0
    0.0   0.0    2.9
     ];

alfa_T3 = [
            -1.9   0.0    0.0
            0.0  -1.9    0.0
            0.0   0.0   -1.5
];
```

```
T = -40:1:100;
T0 = 20;
new_C = zeros(6,6,141);
C_keep = zeros(6,6,141);
C0_keep = zeros(6,6);
```

```
%-------------Piezoelectric--------------%
new_e = zeros(3,6,141);
```

2

Figure 27: MATLAB code p2

```
e_rot = zeros(3,6,141);

phi_cut = 0; theta_cut = 35; psi_cut = 82;
R = rotation_matrix_zxz(phi_cut, theta_cut, psi_cut);
M = bond_matrix(R);

for i = 1:141
    Temp = i-41;
    new_e(:,:,i) = E0.*(ones(3,6) + dE_T*10^-6*(Temp - T0));
    e_rot(:,:,i) = R * new_e(:,:,i) * M';
end

for i = 1:3
    for j = 1:6
        e = reshape(e_rot(i,j,:), [141,1]);
        i,j
        curvefit = fit(T', e, 'poly1')
    end
end



%-------------Permittivity--------------%
new_k = zeros(3,3,141);
k_rot = zeros(3,3,141);

phi_cut = 0; theta_cut = 35; psi_cut = 8;
R = rotation_matrix_zxz(phi_cut, theta_cut, psi_cut);
M = bond_matrix(R);

for i = 1:141
    Temp = i-41;
    new_k(:,:,i) = (D0/(8.854*10^-12)).*(ones(3,3) + dD_T*10^-6*(Temp - T0));
    k_rot(:,:,i) = R * new_k(:,:,i) * R';
end

for i = 1:3
    for j = 1:3
        k = reshape(k_rot(i,j,:), [141,1]);
        i,j
        curvefit = fit(T', k, 'poly1')
    end
end
```

```
%-------------STIFFNESS--------------%
for i = 1:141
Temp=i-41;

new_C(:,:,i) = C0*10^-9.*(ones(6,6) + dC0_T*10^-6*(Temp-T0) + dC0_T2*10^-9*(Temp-T0)^2 + dC0_T3

end
```

3

Figure 28: MATLAB code p3

```
a_inplane = linspace(0, 180, 181);
phi_cut = 0; theta_cut = 35; psi_cut = 0;
in_plane_angle =0;

%for u = 1:181
%    theta_cut = u-91;
    R_cut = rotation_matrix_zxz(phi_cut, theta_cut, psi_cut);

    for i_a = 1:length(a_inplane)
        a_inpl = a_inplane(i_a);
        R_inplane = rotation_matrix_zxz(a_inpl, 0, 0);
        R = R_inplane*R_cut;
        M = bond_matrix(R);

        c_rot = zeros(6,6,141);
        c_t0 = M * C0 * M' * 10^-9;

        for j = 1:141
            c_rot(:,:,j) = M * new_C(:,:,j) * M';
            delta_f4(j) = (sqrt(c_rot(4,4,j))-sqrt(c_t0(4,4)))/sqrt(c_t0(4,4));
            delta_f6(j) = (sqrt(c_rot(6,6,j))-sqrt(c_t0(6,6)))/sqrt(c_t0(6,6));
            f11 = sqrt(c_rot(1,1,j) - (c_rot(1,3,j)^2)/c_rot(3,3,j));
            f01 = sqrt(c_t0(1,1) - (c_t0(1,3)^2)/c_t0(3,3));
            delta_f1(j) = (f11-f01)/f01;
        end

         f_norm(i_a) = norm(delta_f6);
         if(i_a == 83)

             C_keep = c_rot;
             C0_keep = c_t0;


             hold off
             plot(T, delta_f6*10^6);
             ylabel("(f-f0)/f0 [ppm]")
             xlabel("Temperature [degrees]")
             title('\Deltaf_1 for 0.39.3 rotation')
         end
    end
%      f_min(u) = min(f_norm);
%end
%[minim , index] = min(f_min)
plot(a_inplane, f_norm)
ylabel("norm of \Deltaf_4 over temperature range T")
xlabel("In-plane angle [degrees]")
title('Norm of \Deltaf_4 for different in-plane rotations')
```

Single transformation:

```
for i = 1:6
```

4

Figure 29: MATLAB code p4

```
    for j = 1:6
        c = reshape(C_keep(i,j,:), [141,1]);
        i,j
        curvefit = fit(T', c, 'poly3')
    end
end


plot(curvefit, T', c')
```

```
% Angles that specify the cut in degrees
phi_cut = 0; theta_cut = 35; psi_cut = 82;
% Angle to specify the in-plane orientation in degrees
in_plane_angle = 0;
% Define rotation matrices to transform to cut and for in-plane orientation
R_cut = rotation_matrix_zxz(phi_cut, theta_cut, psi_cut);
R_inplane = rotation_matrix_zxz(in_plane_angle, 0, 0);

% Overall rotation
R = R_inplane*R_cut;
M = bond_matrix(R);

% Transformation laws
c_t = M * C0 * M' * 10^-9
dC_T_35 = M * dC0_T * M'
dC_T2_35 = M * dC0_T2 * M'
dC_T3_35 = M * dC0_T3 * M'
e_t = R * E0 * M'
k_t = R * D0 * R' % NEED TO DIVIDE BY 8.854*10^-12
dk_t = R * dD_T * R'
```

Sweep in-plane angle:

```
% ---------------------------------- WRONG WAY OF DOING IT-----------------------------
a_inplane = linspace(0, 180, 181);
% Angles that specify the cut in degrees
phi_cut = 0; theta_cut = 35.15; psi_cut = 0;

f_min = zeros(1,181);
 for i = 1:181
     theta_cut = i-91;
    % Rotation matrix for cut
    R_cut = rotation_matrix_zxz(phi_cut, theta_cut, psi_cut);

    c = C0*10^-9; % Scaling to convert to [GPa]

    T0 = 25;
    T = -40:1:100;

    c11_transformed = zeros(size(a_inplane));
    Tc11_1 = zeros(size(a_inplane));
    Tc11_2 = zeros(size(a_inplane));
```


Figure 30: MATLAB code p5

```
    Tc11_3 = zeros(size(a_inplane));

    c13_transformed = zeros(size(a_inplane));
    Tc13_1 = zeros(size(a_inplane));
    Tc13_2 = zeros(size(a_inplane));
    Tc13_3 = zeros(size(a_inplane));

    c33_transformed = zeros(size(a_inplane));
    Tc33_1 = zeros(size(a_inplane));
    Tc33_2 = zeros(size(a_inplane));
    Tc33_3 = zeros(size(a_inplane));

    c44_transformed = zeros(size(a_inplane));
    Tc44_1 = zeros(size(a_inplane));
    Tc44_2 = zeros(size(a_inplane));
    Tc44_3 = zeros(size(a_inplane));

    c66_transformed = zeros(size(a_inplane));
    Tc66_1 = zeros(size(a_inplane));
    Tc66_2 = zeros(size(a_inplane));
    Tc66_3 = zeros(size(a_inplane));

    f1_norm = zeros(size(a_inplane));
    f4_norm = zeros(size(a_inplane));
    f6_norm = zeros(size(a_inplane));

    for i_a = 1:length(a_inplane)
        a_inpl = a_inplane(i_a);
        R_inplane = rotation_matrix_zxz(a_inpl, 0, 0);
        R = R_inplane*R_cut;
        M = bond_matrix(R);
        c_t0 = M * c * M';
        dc_T = M * dC0_T * M';
        dc_T2 = M * dC0_T2 * M';
        dc_T3 = M * dC0_T3 * M';

        c11_transformed(i_a) = c_t0(1,1);
        Tc11_1(i_a) = dc_T(1,1);
        Tc11_2(i_a) = dc_T2(1,1);
        Tc11_3(i_a) = dc_T3(1,1);

        c13_transformed(i_a) = c_t0(1,3);
        Tc13_1(i_a) = dc_T(1,3);
        Tc13_2(i_a) = dc_T2(1,3);
        Tc13_3(i_a) = dc_T3(1,3);

        c33_transformed(i_a) = c_t0(3,3);
        Tc33_1(i_a) = dc_T(3,3);
        Tc33_2(i_a) = dc_T2(3,3);
        Tc33_3(i_a) = dc_T3(3,3);

        c44_transformed(i_a) = c_t0(4,4);
        Tc44_1(i_a) = dc_T(4,4);
        Tc44_2(i_a) = dc_T2(4,4);
```


Figure 31: MATLAB code p6

```
        Tc44_3(i_a) = dc_T3(4,4);

        c66_transformed(i_a) = c_t0(6,6);
        Tc66_1(i_a) = dc_T(6,6);
        Tc66_2(i_a) = dc_T2(6,6);
        Tc66_3(i_a) = dc_T3(6,6);

        c11 = c11_transformed(i_a)*(1+ Tc11_1(i_a)*10^-6*(T-T0) + Tc11_2(i_a)*10^-9*(T-T0).^2 +
        c13 = c13_transformed(i_a)*(1+ Tc13_1(i_a)*10^-6*(T-T0) + Tc13_2(i_a)*10^-9*(T-T0).^2 +
        c33 = c33_transformed(i_a)*(1+ Tc33_1(i_a)*10^-6*(T-T0) + Tc33_2(i_a)*10^-9*(T-T0).^2 +
        c44 = c44_transformed(i_a)*(1+ Tc44_1(i_a)*10^-6*(T-T0) + Tc44_2(i_a)*10^-9*(T-T0).^2 +
        c66 = c66_transformed(i_a)*(1+ Tc66_1(i_a)*10^-6*(T-T0) + Tc66_2(i_a)*10^-9*(T-T0).^2 +

        f1 = sqrt(c11 - (c13.^2)./c33);
        f0 = sqrt(c11_transformed(i_a) - (c13_transformed(i_a))^2/c33_transformed(i_a));
        f = (f1-f0)/f0;
        f1_norm(i_a) = norm(f);

        f0_4 = sqrt(c44_transformed(i_a));
        f4 = ((sqrt(c44) -f0_4)/f0_4);
        f4_norm(i_a) = norm(f4);

        f0_6 = sqrt(c66_transformed(i_a));
        f6 = ((sqrt(c66) -f0_6)/f0_6);
        f6_norm(i_a) = norm(f6);

        if(i_a == 91)
            hold off
            plot(T, f4*10^6)
            ylabel("(f-f0)/f0 [ppm]")
           xlabel("Temperature [degrees]")
            title('\Deltaf_4/f_{04} for 0.-27.51 rotation')
        end


    end
    f_min = min([min(f1_norm), min(f4_norm), min(f6_norm)]);
end
[minim , index] = min(f_min)
figure
hold on
plot(a_inplane, f1_norm./140, 'DisplayName', 'norm \Deltaf1')
plot(a_inplane, f4_norm./140, 'DisplayName', 'norm \Deltaf4')
plot(a_inplane, f6_norm./140, 'DisplayName', 'norm \Deltaf6')
xlabel('In-plane angle [deg]')
ylabel('norm of \Deltaf over temperature range T')
title('norm of \Deltaf  0.75.inplanrotation')
legend;
```

7

Figure 32: MATLAB code p7

```
function m = bond_matrix(r)
% Returns the 6 x 6 Bond matrix corresponding to an arbitrary transformation
% specified by the 3 x 3 matrix r
% Reference for Bond matrix: B. A. Auld, Acoustic Fields and Waves in
% Solids, pp. 55-86

% First 3 rows of the Bond matrix
l1 = [r(1,1)^2 r(1,2)^2 r(1,3)^2 2*r(1,2)*r(1,3) 2*r(1,3)*r(1,1) 2*r(1,1)*r(1,2);
      r(2,1)^2 r(2,2)^2 r(2,3)^2 2*r(2,2)*r(2,3) 2*r(2,3)*r(2,1) 2*r(2,1)*r(2,2);
      r(3,1)^2 r(3,2)^2 r(3,3)^2 2*r(3,2)*r(3,3) 2*r(3,3)*r(3,1) 2*r(3,1)*r(3,2)];

% Rows 3 to 6, first 3 columns
l12 = [r(2,1)*r(3,1) r(2,2)*r(3,2) r(2,3)*r(3,3);
       r(3,1)*r(1,1) r(3,2)*r(1,2) r(3,3)*r(1,3);
       r(1,1)*r(2,1) r(1,2)*r(2,2) r(1,3)*r(2,3)];
% Rows 3 to 6, last 3 columns
l22 = [r(2,2)*r(3,3) + r(3,2)*r(2,3) r(2,3)*r(3,1) + r(3,3)*r(2,1) r(2,1)*r(3,2) + r(3,1)*r(2,2
       r(3,2)*r(1,3) + r(1,2)*r(3,3) r(3,3)*r(1,1) + r(1,3)*r(3,1) r(3,1)*r(1,2) + r(1,1)*r(3,2
       r(1,2)*r(2,3) + r(2,2)*r(1,3) r(1,3)*r(2,1) + r(2,3)*r(1,1) r(1,1)*r(2,2) + r(2,1)*r(1,2

m = [l1;
     l12 l22];
end

function n = bond_strain_matrix(r)
% Returns the 6 x 6 Bond strain matrix corresponding to an arbitrary transformation
% specified by the 3 x 3 matrix r
% Reference for Bond matrix: B. A. Auld, Acoustic Fields and Waves in
% Solids, pp. 55-86
    M = bond_matrix(r);
    m11 = M(1:3, 1:3);
    m22 = M(4:6, 4:6);
    m12 = M(1:3, 4:6);
    m21 = M(4:6, 1:3);

    n = [m11 m21/2;
         2*m12 m22];
end

function r = rotation_matrix_zxz(phi, theta, psi)
% Return the 3 x 3 rotation matrix for a rotation around the phi (z-axis),
% theta (x'-axis) and psi (z''-axis)  Euler angles in degrees.
% z-x'-z'' (x-convention, see https://mathworld.wolfram.com/EulerAngles.html)

a11 = cosd(psi)*cosd(phi) - cosd(theta)*sind(phi)*sind(psi);
a21 = -sind(psi)*cosd(phi) - cosd(theta)*sind(phi)*cosd(psi);
a31 = sind(theta)*sind(phi);
a12 = cosd(psi)*sind(phi) + cosd(theta)*cosd(phi)*sind(psi);
a22 = -sind(psi)*sind(phi) + cosd(theta)*cosd(phi)*cosd(psi);
a32 = -sind(theta)*cosd(phi);
a13 = sind(theta)*sind(psi);
a23 = cosd(psi)*sind(theta);
```

8

Figure 33: MATLAB code p8

```
a33 = cosd(theta);

r = [a11 a12 a13;
     a21 a22 a23;
     a31 a32 a33];

end

function r = rotation_matrix_zyx(phi, theta, psi)
% Return the 3 x 3 rotation matrix for a rotation around the phi (z-axis),
% theta (y'-axis) and psi (x''-axis) Euler angles in degrees.
% z-y'-x'' ((pitch, roll, yaw) convention, see https://mathworld.wolfram.com/EulerAngles.html)

a11 = cosd(theta)*cosd(phi);
a12 = cosd(theta)*sind(phi);
a13 = -sind(theta);
a21 = sind(psi)*sind(theta)*cosd(phi) - cosd(psi)*sind(phi);
a22 = sind(psi)*sind(theta)*sind(phi) + cosd(psi)*cosd(phi);
a23 = cosd(theta)*sind(psi);
a31 = cosd(psi)*sind(theta)*cosd(phi) + sind(psi)*sind(phi);
a32 = cosd(psi)*sind(theta)*sind(phi) - sind(psi)*cosd(phi);
a33 = cosd(theta)*cosd(psi);

r = [a11 a12 a13;
     a21 a22 a23;
     a31 a32 a33];
end
```

9

Figure 34: MATLAB code p9

A.3 MATLAB results

(186.8-0.008431*T)[GPa]	(66.31-1.412*(10^-8)*T)[GPa]	(80.63-0.005147*T)[GPa]	0	(6.149+0.006587*T)[GPa]	0
(66.31-1.412*(10^-8)*T)[GPa]	(198.4-3.452*(10^-8)*T)[GPa]	(53.54-1.002*(10^-8)*T)[GPa]	0	(6.957+1.258*(10^-9)*T)[GPa]	0
(80.63-0.005147*T)[GPa]	(53.54-1.002*(10^-8)*T)[GPa]	(209.2-0.008431*T)[GPa]	0	(5.827+0.01079*T)[GPa]	0
0	0	0	(56.62-9.814*(10^-9)*T)[GPa]	0	(-4.005-1.328*(10^-9)*T)[GPa]
(6.149+0.006587*T)[GPa]	(6.957+1.258*(10^-9)*T)[GPa]	(5.827+0.01079*T)[GPa]	0	(74.81+0.008431*T)[GPa]	0
0	0	0	(-4.005-1.328*(10^-9)*T)[GPa]	0	(74.86-1.263*(10^-8)*T)[GPa]

Figure 35: Stiffness matrix (with 1st order temperature coefficients) of the 0.38.90 rotated Lithium Niobate

4.5333[C/m^2]	-1.7223[C/m^2]	-1.3519[C/m^2]	0[C/m^2]	0.2174[C/m^2]	0[C/m^2]
0[C/m^2]	0[C/m^2]	0[C/m^2]	4.3977[C/m^2]	0[C/m^2]	0.3648[C/m^2]
-2.4540[C/m^2]	1.7263[C/m^2]	2.5953[C/m^2]	0[C/m^2]	0.7352[C/m^2]	0[C/m^2]

Figure 36: Piezoelectric matrix of the 0.38.90 rotated Lithium Niobate

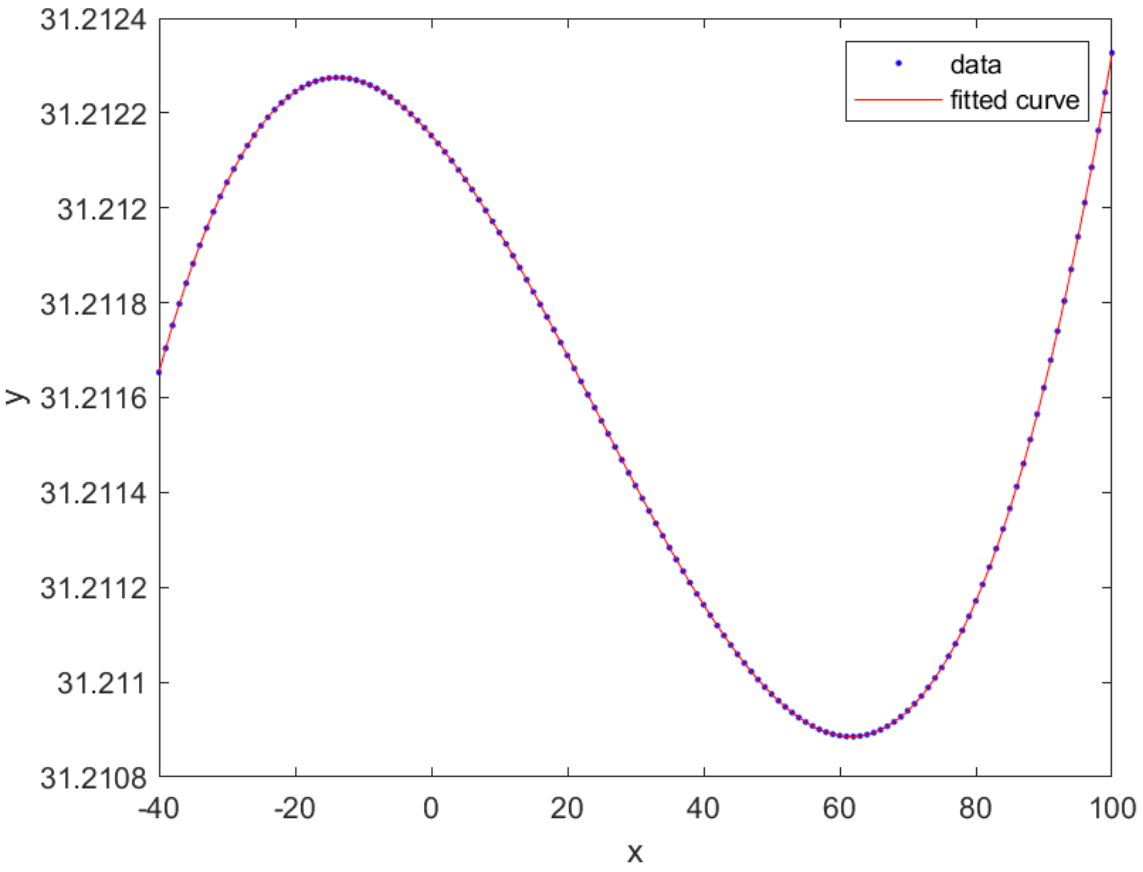

Figure 37: Fitted curve obtained using MATLAB function fit() for the c_{66} coefficient of the 0.35.82 rotated quartz

A.4 COMSOL results

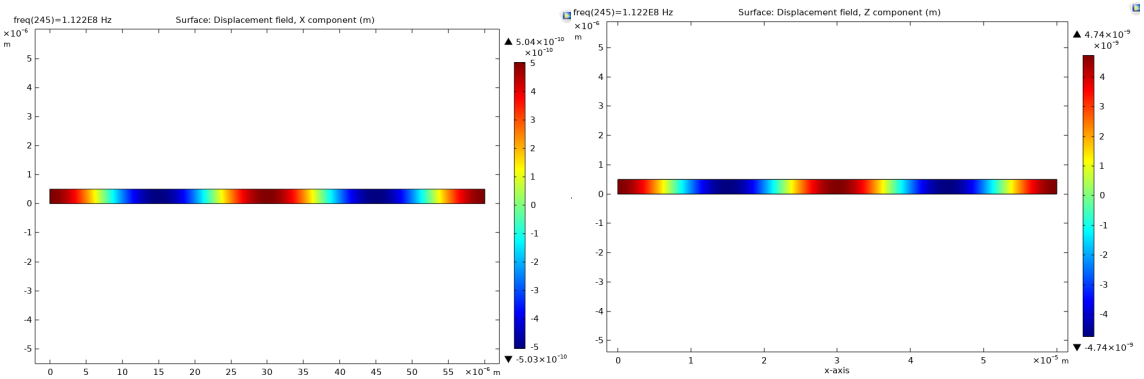

Figure 38: Displacement fields in x (left) and z (right) directions for the verifying simulation of axis orientation in Comsol

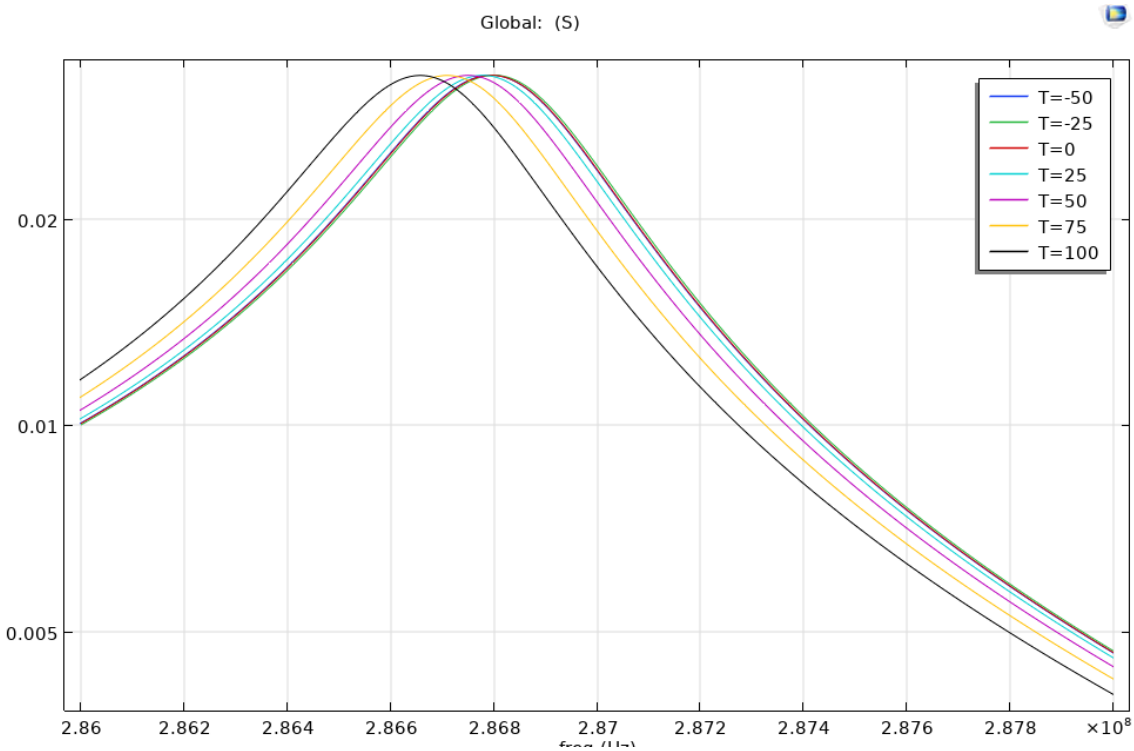

Figure 39: Impedance of the Silicon/Lithium Niobate frequency domain study for 7 different temperatures - 1um thick Si

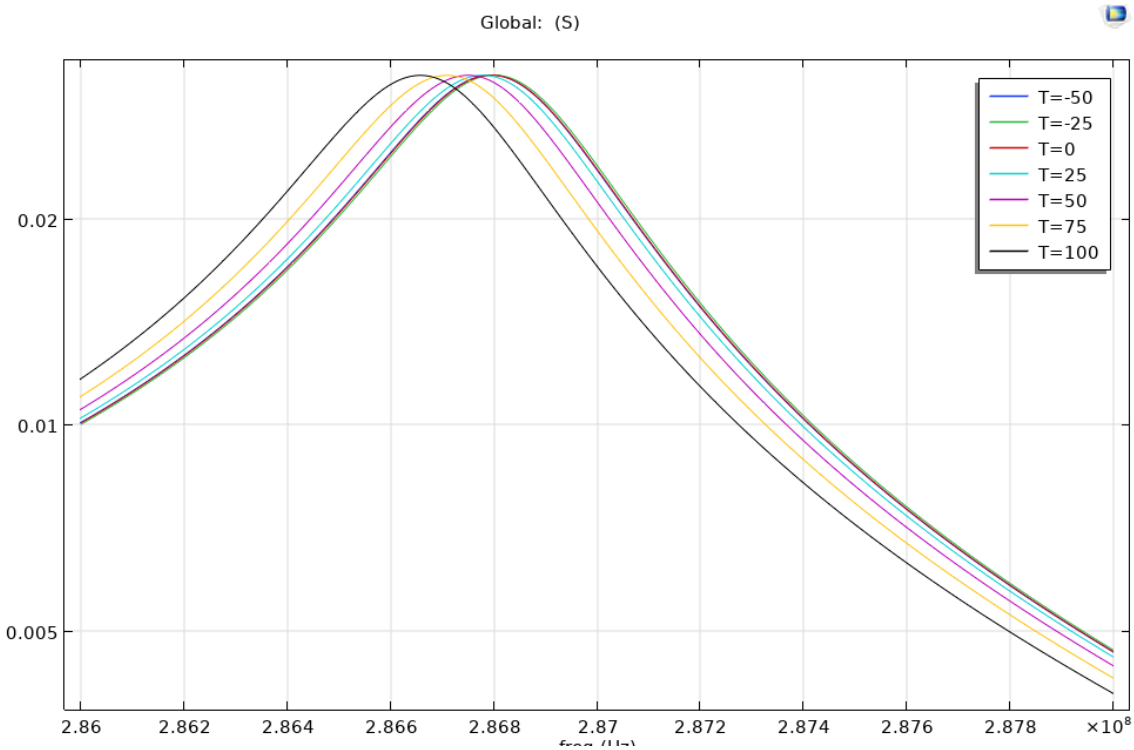

Figure 40: Impedance of the Silicon/Lithium Niobate frequency domain study for 7 different temperatures - 2um thick Si

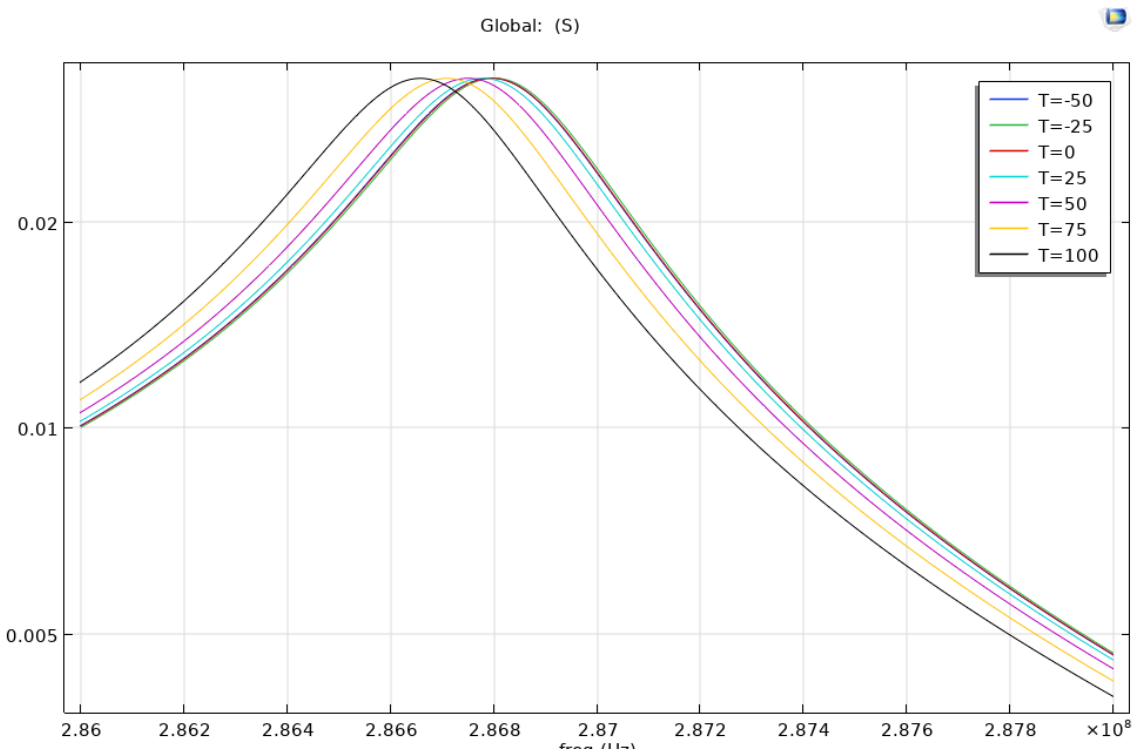

Figure 41: Impedance of the Silicon/Lithium Niobate frequency domain study for 7 different temperatures - 5um thick Si