

COMSOL Multiphysics: MEMS gyroscope design focus on split vs etching

Giacomo Langfelder and Luca Pileri

06/11/2024

In this lab, you will learn how to analyze the resonance frequency of a capacitive MEMS gyroscope with COMSOL Multiphysics; you will learn how to set-up a parametric simulation, to evaluate over-/under-etch effects; you will learn how to re-design the gyroscope, to fulfill the resonance frequency variations requirements.

Start by downloading the following file: [E11 yaw gyroscope start.mph](#)

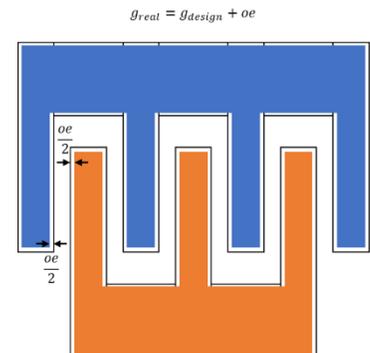
You can click on the file name above or find it on the course website (scroll down to “Comsol files”).

Introduction

A colleague of yours has just been fired after poor results obtained while designing a capacitive MEMS gyroscope. Your boss is promising you a gratification if you personally carry on the redesign of the gyroscope. You do not want to disappoint her, as you need money for upcoming Christmas presents.

The specifications to fulfill by design are reported in the table below.

Name	Value	Description
area	$(400 \times 800) \mu\text{m}^2$	Maximum area constraint
f_d	20 kHz	Drive resonance frequency
f_{d_ph}	14.14 kHz	In-phase drive resonance frequency
Δf	1.1 kHz	Drive-sense mismatch
w_{\min}	$1 \mu\text{m}$	Minimum spring width
OE	$\pm 0.05 \mu\text{m}$	Maximum over-/under-etch (double-sided)



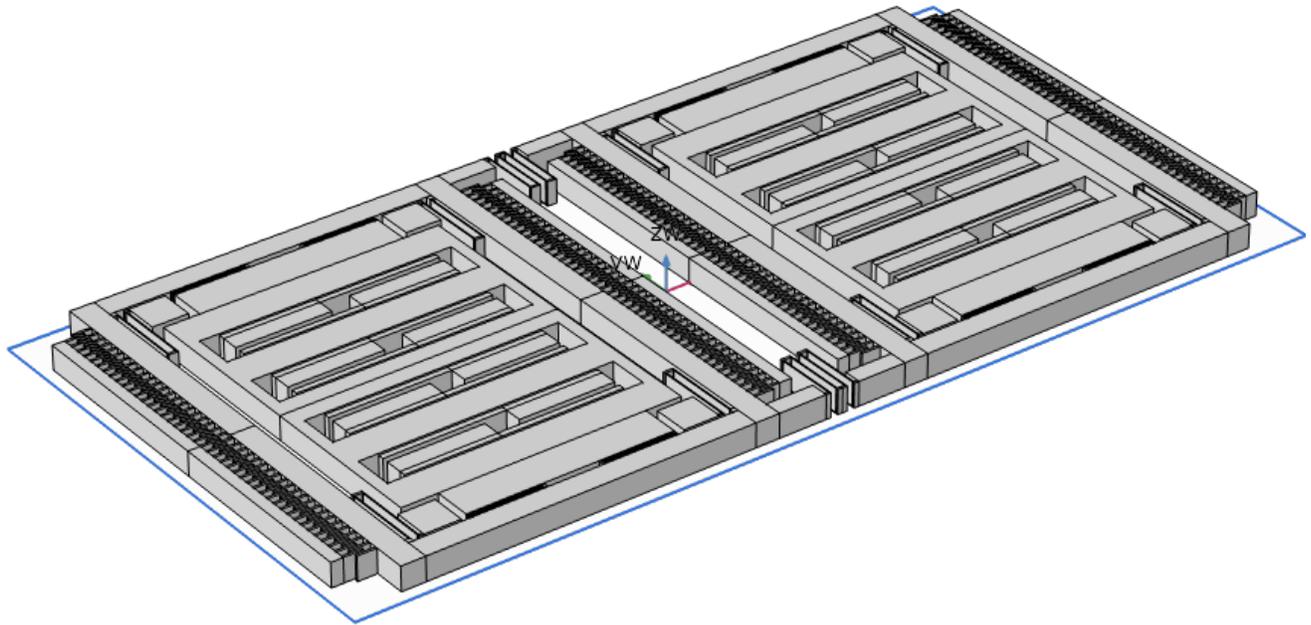
The request on the maximum fluctuation of the resonance frequency of the drive axis with respect to possible process fluctuations (the over/under etch) is $\pm 1 \text{ kHz}$. We want to analyze whether your former colleague matched or not this specification and, in case, solve this issue.

1. Structure

Open the downloaded file [E11_yaw_gyroscope_start.mph](#)

As you now know, capacitive MEMS gyroscopes are pretty complex structures (multiple frames, springs, anchor points, stators, different resonant modes, ...). The proposed geometry, reported in the figure below, emulates a real capacitive gyroscope for consumer applications.

You can observe how many parameters are necessary to programmatically define all the different instances. At least from this standpoint, your former colleague did a very good job!



- v
globe Global Definitions
 - P_i General Parameter
 - P_i Comb finger
 - P_i Tuning fork
 - P_i Drive mass
 - P_i Drive spring
 - P_i Drive sense spring
 - P_i Sense mass
 - P_i Parallel plate

Label: Tuning fork

Parameters

Name	Expression	Value	Description
lf	50.2e-6 [m]	5.02E-5 m	length fork
wf	1.07e-6 [m] + oe	1.07E-6 m	width fork
nff	2	2	number of the fold fork

Label: Drive spring

Parameters

Name	Expression	Value	Description
wd	1.07e-6 [m] + oe	1.07E-6 m	width drive spring
ld	63.2e-6 [m]	6.32E-5 m	length drive spring
nfd	2	2	# fold drive spring

Label: Drive sense spring

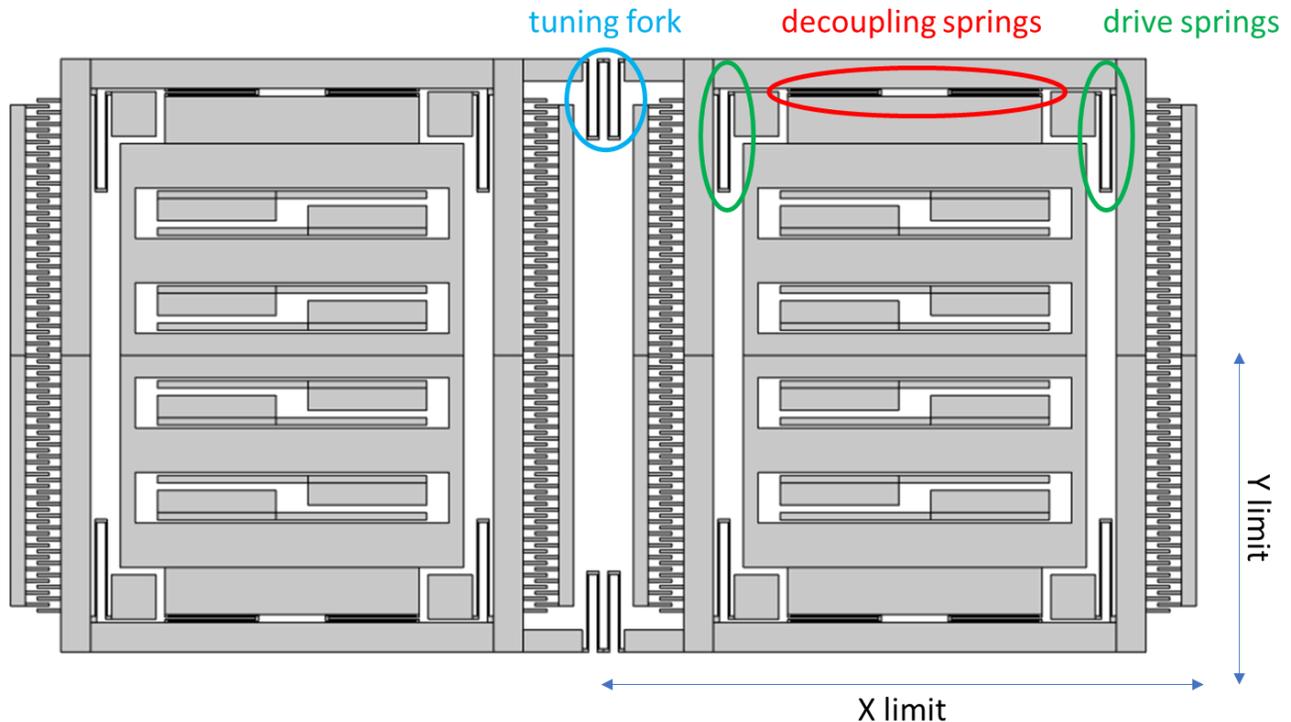
Parameters

Name	Expression	Value	Description
wdc	1.11e-6 [m] + oe	1.11E-6 m	width drive-sense spring
ldc	58.84e-6 [m]	5.884E-5 m	length drive-sense spring
nfdc	2	2	# fold drive-sense spring

For the sake of simplicity, our aim is to satisfy the requirements while changing only the characteristic dimensions of the springs: i.e. the length, the width and the number of folds. The corresponding parameters are highlighted in the figure above, in orange for the tuning fork, in green for the drive springs, and in purple for the drive/sense decoupling ones.

Their nominal values are summarized also in the table below:

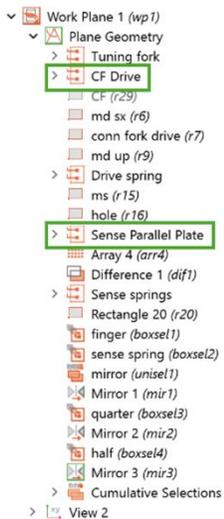
Parameter	Expression
lf	50.2e-6 [m]
wf	1.07e-6 [m]
nff	2
ld	63.2e-6 [m]
wd	1.07e-6 [m]
nfd	2
ldc	58.84e-6 [m]
wdc	1.11e-6 [m]
nfdc	2



As you can note from the top view of the gyroscope above, the chosen geometry consists of a dual-frame decoupled design: drive and Coriolis frames are different and the Coriolis frame coincides in this case with the sense frame (no double decoupling, see Lecture n. 12).

As we learned in previous COMSOL labs, we know that the software calculates the solution in a finite number of points defined by the Mesh: the higher the number of points, the more accurate the solution, but computational time slows down and RAM requirements are higher (for standard simulations, at least 32 GB of RAM are recommended).

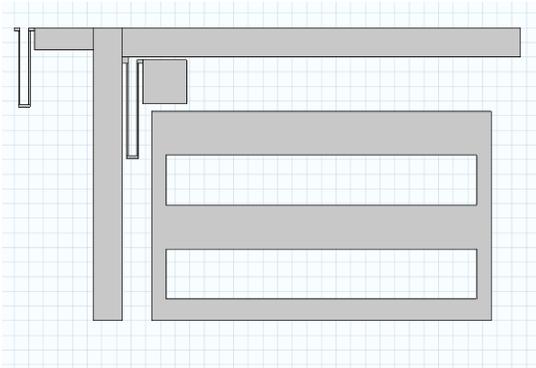
To reduce computational time, we can disable some “unnecessary” parts (e.g. parallel plates, comb finger stators, ..., i.e. those parts that are not involved in motion and thus in the resonance frequency definition) and simplify some structural parts (e.g., all comb finger arrays can be condensed as a rectangle with the same area).



To **disable** an instance, you can either right click on the instance and choose *Disable*, or left click on the instance and press F3.

Expand *Geometry1* tree → expand *Work Plane 1* tree → expand *Plane Geometry* → (hold CTRL for a multiple selection) select all the instances shown in the figure here on the left → press F3 (or right click and disable).

Press F8 to rebuild all the structure.



As you can note, we removed comb fingers and parallel plates.

While removing the combs, we also removed their mass, that indeed contributes to the mass of the structure. To compensate for this loss, we can add a rectangle with an area (i.e. mass) corresponding to all the combs, and place it roughly in the same design region.

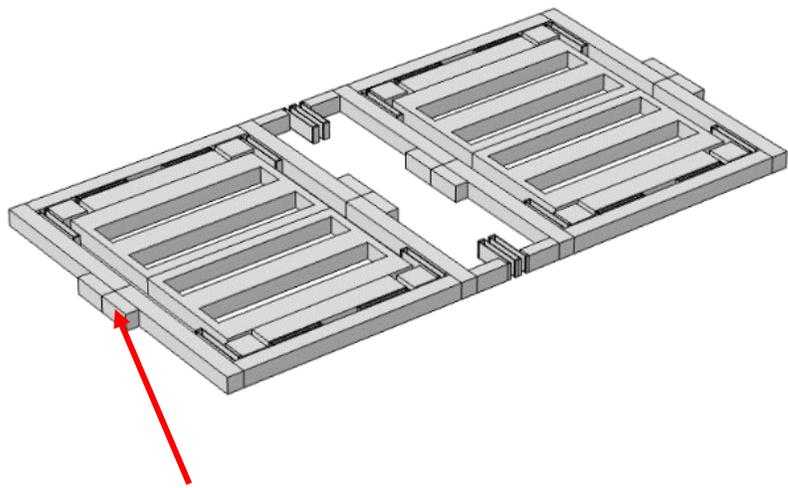
The rectangle is already defined; we should simply enable it. Click on CF → press F4.

We can check whether the new simplified geometry is correct.

Click on *Geometry 1* → press F8.

The geometry is now correct.

Additional note: when you have a geometry which is already designed, as in this case, you can check the design up to a given point by selecting the corresponding design row, and then clicking on “build selected”. At this point, in the geometry you will just see what was designed up to that row.



Volume (and mass) equivalent to the removed combs

2. Materials and Boundary Settings

From the point of view of materials and boundaries as well, the design is ready: we just check what your former colleague defined in his design.

Open *Material* tree → only one material is visible, if you click on it, the corresponding shapes are colored in blue: the whole structure is (correctly) made of polysilicon (Poly-Si).

Open *Solid Mechanics* tree → only *Fixed Constraint* has been added with respect to the default boundary conditions → click on it to check that only the anchorage to the substrate are fixed: 8 “cubes” will be highlighted at the 8 corners of the two frames.

3. Mesh

Today we learn a different way to create a mesh, often useful when computational complexity gets increased.

Open *Mesh* tree → click on *Free triangular*. Note that this type of mesh, which acts on a 2D surface, is associated to “*Box 1*”. Click F7 or click “build selected” to create this 2D mesh.

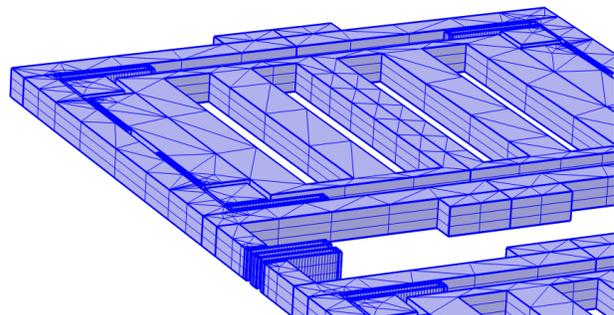
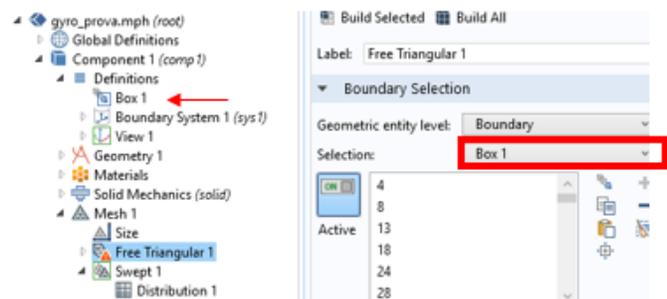
We note that only the top 2D side of the gyroscope is meshed, which indeed corresponds to *Box 1*.

Your colleague defined *Box 1* in *Definitions*: the difference between this *Box 1* and a *Box selection* (e.g., the one used for selecting instances in *Plane Geometry*) is the possibility to recall it also for *Materials*, *Solid Mechanics* and *Mesh*.

The *free triangular* command **meshes only the surface**, but obviously **we need to mesh the entire 3D volume**: the *Swept 1* command fulfills this issue. Sweeping a mesh means extruding it all along a distance, forming therefore **prisms instead of tetrahedrons** (which we used in the past class on accelerometers).

In practice, this command translates the mesh through the volume from one plane side towards the opposite faces; in particular, the command *Distribution* is used to define the number of sub-planes inside the volume. As you can easily understand, this mesh works only for structures which are evenly grown along one dimension, which is exactly the case of MEMS designs based on planar processes, for which the height is fixed and even along the whole device.

Note also that this is an *isotropic* mesh, and thus works poorly when you need to accurately evaluate motion along all the space dimensions. In that case, a tetrahedral mesh is preferred. For yaw-axis gyroscopes, however, where motion (of interest) is mostly in-plane, this solution works fine as well.



4. Study

The simulation study is set to eigenfrequency, as we already saw for accelerometers. In order to analyze the modes, we have to let the solver start the solution.

Click on *Study* → press F8 (or right-click on *Study*, and the on *Compute*). Once the simulation is finished, to show the different resonant modes, click on *Eigenfrequency* in the *Setting* window, and scroll through the desired eigen-frequency values. You will recognize the in-phase drive mode, the **anti-phase drive mode** (the desired one), the in-phase sense mode, the **anti-phase sense mode** (the desired one) and other spurious modes.

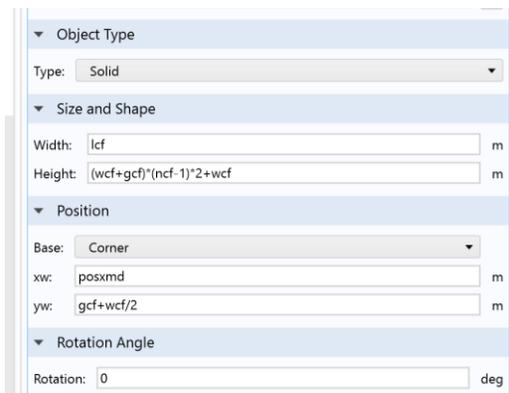
The fundamental resonant modes are oscillating near the correct value, and the mismatch is thus close to the desired split (the two in anti-phase):

$$f_d \approx 19.87 \text{ kHz}$$

$$f_s \approx 21.03 \text{ kHz}$$

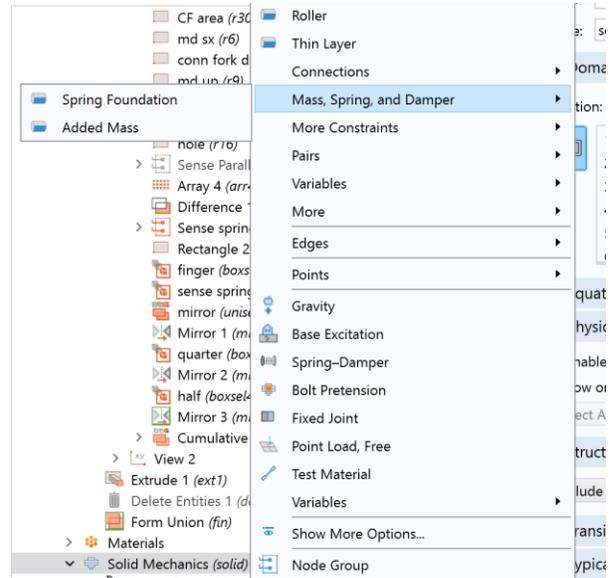
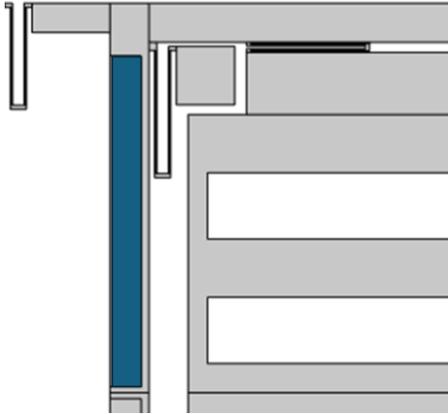
$$\Delta f_{ds} \approx 1.16 \text{ kHz}$$

Remember that COMSOL solved the study with a number of degrees of freedom lower than 600'000. To ensure good matching between simulation and reality the needed number of degrees of freedom is higher, at least 3'000'000. For the sake of saving time, we limited our analysis to a relatively small number of finite elements.

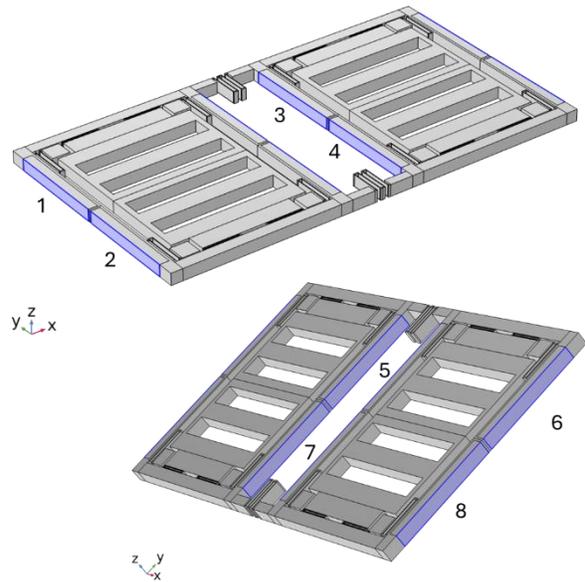


There is a better way to simulate the comb finger without designing them in the geometry. The actual role of the drive CF on system resonance frequencies is an addition in the overall mass on the drive frame. COMSOL can simulate it applying a new boundary on the mass frame.

In the geometry section, modify the CF as sketched on the left. An apparent side face appears on the overall structure in correspondences of the combs. Then, in the solid mechanics section, add the "Added mass" constraint shown in the figure below.



In the selection panel, select the 8 area where should be added the combs weight. In the end, change the mass type in “Total mass” and write the following formula:



If you recompute the modes, you see that practically nothing is changed, a part of the spurious mode at 60 kHz. However, this method requires less mesh points to be simulated and the weight is distributed in a more realistic way.

Then... if the gyroscope was properly designed, why was your colleague fired?

It is easy to obtain a good MEMS design assuming an “ideal” process, without spreads and without taking into account the statistic fluctuations of the fabrication process... It is more challenging to design a device which is robust also to process fluctuations.

Let us analyze these fluctuations in the next section.

5. Over-/Under-etch effects: parametric analyses

We can easily evaluate the effects of over-/under-etch with a parametric sweep simulation: note that, if you go back to the parameters definition, the width of all springs was defined with a nominal value “plus the overetch *oe*”. This *oe* parameter was initially set to 0 (note: a positive overetch value indicates actually here an under-etch, as the spring size increases).

ooooFO...	0		
lf	50.2e-6 [m]	oe	0e-6 [m]
wf	1.07e-6 [m] + oe		
nff	2		
hconnf	2*wf		
lconnf	2*TR/nff+2*wf		
ooooDR...	0		
hmd	ylimit		
wmdsx	20e-6 [m]		
posxmd	nff*(lconnf-wf)+TR+was+2*ld		
lmdup	xlimit-2*(posxmd+wmdsx)+nf		
area_d	(2*nc*wf*ldf)+(2*ylimit*wmd		
mass_d	rho*thickness*area_d		
ooooDR...	0		
wd	1.07e-6 [m] + oe		
ld	63.2e-6 [m]		
nfd	2		
lconnnd	2*TR/nfd+2*wd		
hconnnd	2*wd		
ooooDR...	0		
posxdc	posxms+lanc		
ldc	58.84e-6 [m]		
wdc	1.11e-6 [m] + oe		

To start the parametric analysis as a function of the etching value, right click on *Study* → *Parameter Sweep*.

Click on the PLUS button (*Add*), to add the sweeping variable. Scroll down into the menu, and then select *oe*.

Parametric Sweep

Compute Update Solution

Label: Parametric Sweep

Study Settings

Sweep type: Specified combinations

Parameter name	Parameter value list	Parameter unit
oe		

Output While Solving

Plot Plot group: Mode Shape (solid)

Probes: All

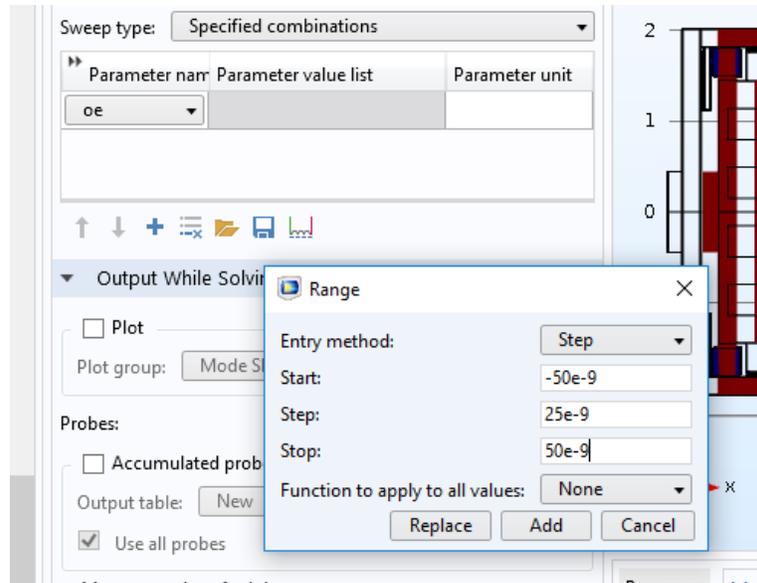
Accumulated probe table Output table: New

Use all probes

Memory settings for jobs

Keep solutions in memory: All

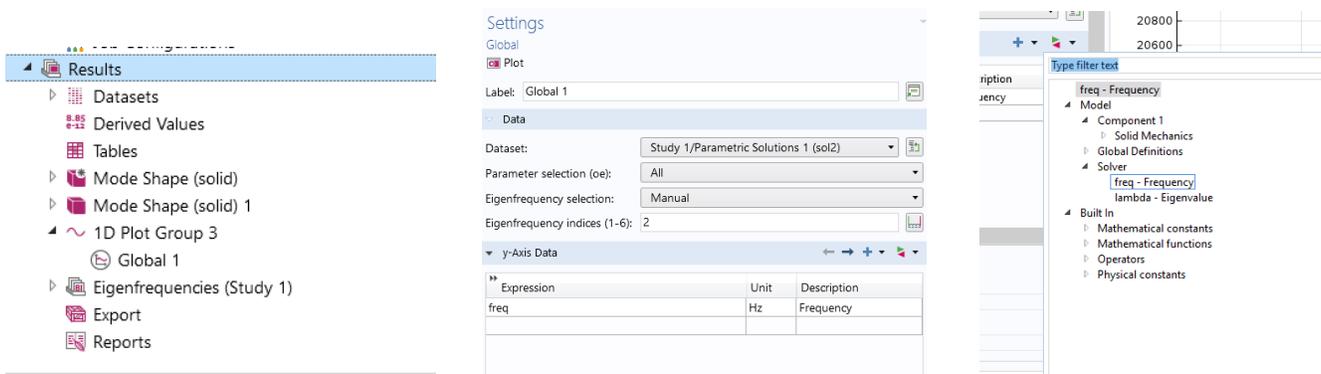
We now need to select the sweeping range: click on the *Range icon*, on the right side of *Save icon*. Enter the range, e.g. from $-50e-9$, to $50e-9$ (from -50 nm, to 50 nm), with a $25e-9$ step, i.e. with 5 simulation steps → click on *Replace* (note: you can try more simulation steps, e.g. by putting 12.5 nm step, but this will double the overall simulation time, which is already 5 min 7 sec on my PC).



Right click on *Study 1* → *Compute*, and wait for simulation results. It may take some time, as the solver is now computing the eigenfrequency solution for 5 consecutive times. You may ask questions during this time slot...

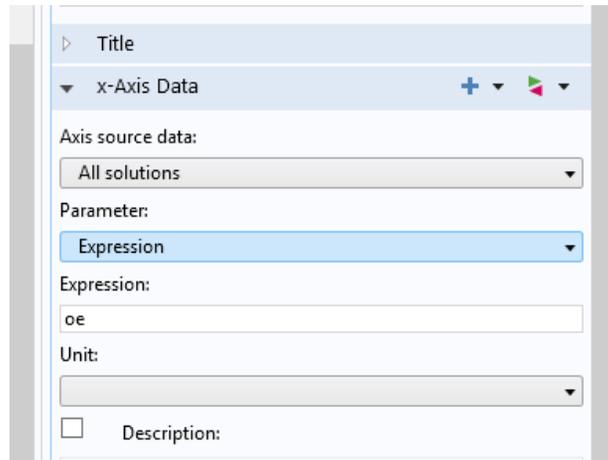
At the end, we are interested in plotting the results, i.e. the resonance frequency as a function of the over-etch. To do so, start by right clicking on *Results* → *1D Plot Group* (indeed, we want to plot a 1D curve of the frequency vs the overetch).

Now right-click on *1D Plot Group*, and then click *Global*: this means that you want to plot a Global result, i.e. a result which is not a function of the specific finite element that you select (e.g., the displacement is not a Global parameter, while the resonance frequency is a Global parameter).

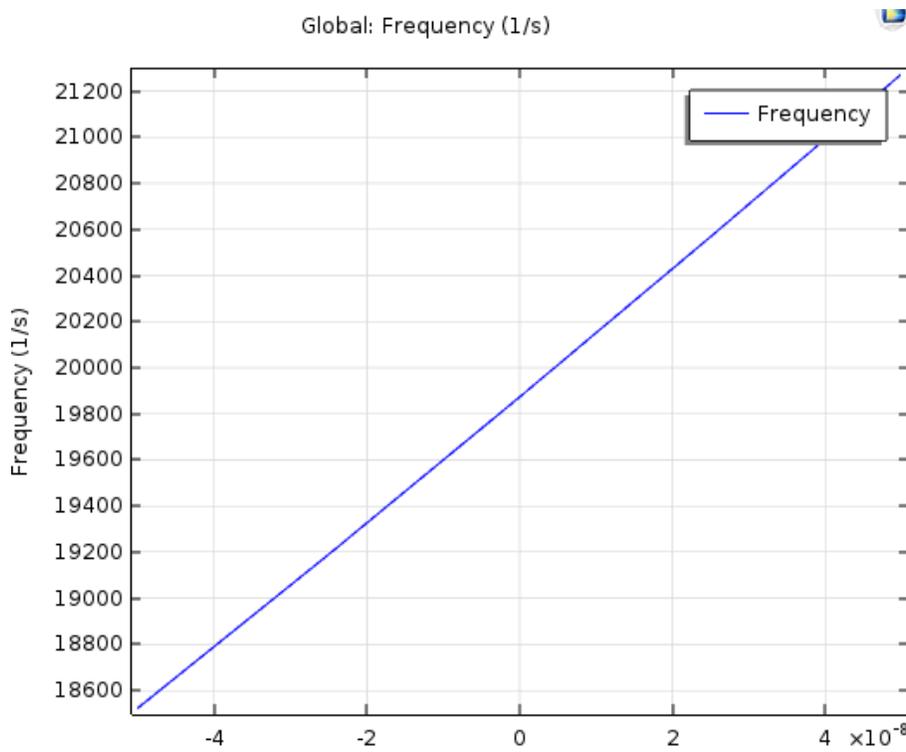


Next, in the *Settings* window of the *Global* tab, we need to specify (i) for which values of the *oe* we want to plot our results (select *All*), (ii) which mode we want to plot (start by selecting *Manual* → *2*, which corresponds to the anti-phase drive frequency), and (iii) which quantity we want to plot (select *Solver* → *freq* from the red/green double arrow on the *y-axis* data tab).

Next, we have to define what to use as the x-axis parameter on our graph. Obviously, we want to use the overetch value. Scroll down to x-Axis data. In Axis source data, select *All solutions*. In Parameter, select *Expression*, in Expression, enter *oe*.



Then, click on Plot, or press F8 to get your results plotted as below.

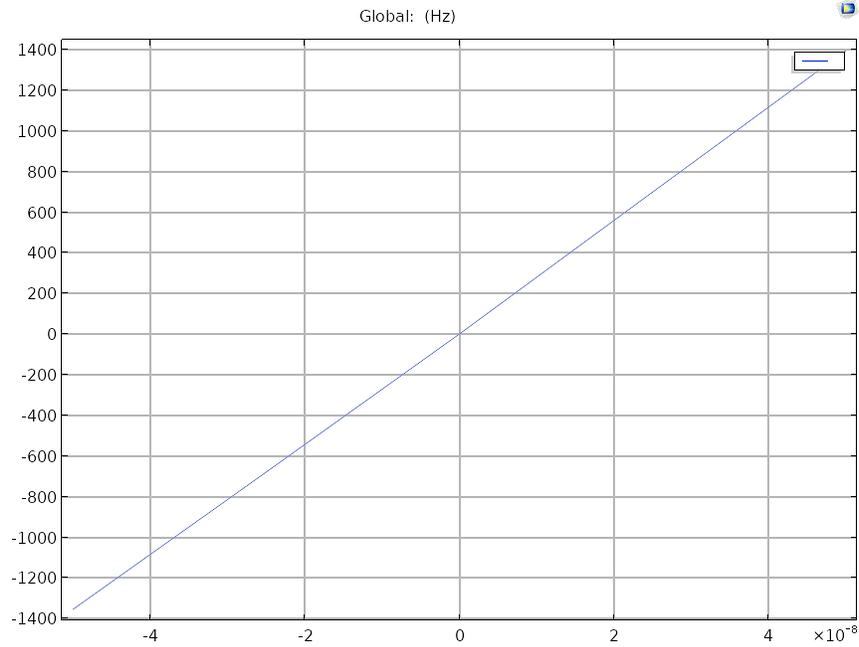


We note that the resonance frequency of the drive mode changes by about 1800 Hz from roughly 18.5 kHz to 21.3 kHz as a function of the process spreads. This is not in line with the specification requirements.

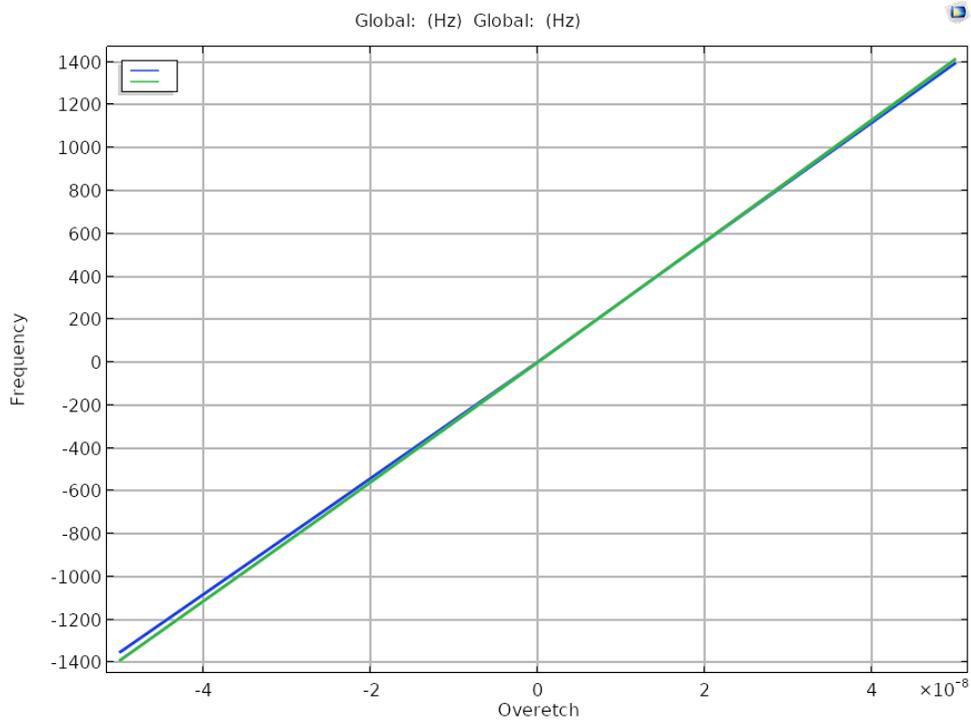
For the sake of completeness, you can then add the sense-mode resonance frequency and infer how the mismatch value changes with the overetch.

For this purpose, we first remove the value at zero-overetch from the graph, in order to have it plotted as a “difference from the nominal value”. We thus just add a “-19870” to the expression of the quantity to plot. The result is shown below.

Expression:
freq-19870



Instead of repeating every step a second time, you can just right click on *Global* and click on *Duplicate*. Then, in the *Data* tab of the new *Global* sub-tab you select 4 as the *Eigenfrequency index*. In the *y-axis* window, write now *freq-21030* (indeed, the value of the sense mode frequency was 21.03 kHz without overetch). You can then click on *Plot* to get the result below, which, showing that the two frequencies drift almost identically, indicate that the mode split remains rather robust against process fluctuations, with respect to the native split of 1.1 kHz.



6. Do it yourself...: robust re-design

The following lines help you in redesigning the gyroscope in such a way to have it more robust to process spreads. First of all, save your design with another file name, e.g. *E12_gyro_2024_my_redesign.mph*.

1) Resonance frequencies

$$f_{d,ip} = \frac{1}{2\pi} \sqrt{\frac{k_{d,ip}}{m_d + m_s}} = 14.4 \text{ kHz}$$

$$f_d = \frac{1}{2\pi} \sqrt{\frac{k_{d,ip} + k_{tf}}{m_d + m_s}} = 20 \text{ kHz}$$

$$f_s = \frac{1}{2\pi} \sqrt{\frac{k_s}{m_s}} = 21 \text{ kHz}$$

2) How can we design the springs? Over-/Under-etch affects the resonant frequency:

$$\Delta w = OE$$

$$\frac{\Delta k}{\Delta w} = 3 \frac{k}{w} \rightarrow \Delta k = 3 \frac{OE}{w} k$$

$$f = \frac{1}{2\pi} \sqrt{\frac{k}{m}} \rightarrow \frac{\Delta f}{\Delta k} = \frac{1}{2} \frac{f}{k} \rightarrow \Delta f = \frac{1}{2} \frac{\Delta k}{k} f = \frac{3}{2} \frac{OE}{w} f$$

$$w = \frac{f}{\Delta f} \frac{3}{2} OE$$

Given a certain double-sided over-etch, OE , given a certain nominal resonance frequency, f , given the desired maximum frequency variation due to over-etch, Δf , the last equation returns the desired spring width:

$$w_f = w_d = \frac{f_d}{\Delta f} \frac{3}{2} OE$$

Given the desired width, as the number of springs is fixed by the geometry, we can then act on the number of folds and on the length (pay attention to the maximum values) in order to guarantee the correct elastic stiffness.

The request for the frequency stability is formulated just for the drive resonance frequency, but we know that sensitivity depends on the mismatch, thus we have to extend it also to sense springs.

$$w_s = \frac{f_s}{\Delta f} \frac{3}{2} OE$$

In *Parameters*, you can define the resonance frequencies, the required frequency stability and the over-etch:

$$\mathbf{fdip} = 14.14e3 \text{ [Hz]}$$

$$\mathbf{fd} = 20e3 \text{ [Hz]}$$

$$\mathbf{fs} = 21e3 \text{ [Hz]}$$

$$\mathbf{dfmax} = 1e3 \text{ [Hz]}$$

$$\mathbf{oe} = 50e-9 \text{ [m]} \text{ (note that this parameter is already present in the Parameters Definition, just change the value)}$$

$$\mathbf{E} = 160e9 \text{ [Pa]}$$

Given the resonance frequencies, given the mass, you can calculate the required elastic stiffness of each spring:

$$\mathbf{kdp} = (\text{mass}_d + \text{mass}_s) * (2 * \pi * \mathbf{fdip})^2$$

$$\mathbf{kdf} = (\text{mass}_d + \text{mass}_s) * (2 * \pi * \mathbf{fd})^2 - \mathbf{kdp}$$

$$\mathbf{kdf} = (\text{mass}_s) * (2 * \pi * \mathbf{fs})^2$$

From specs, using previous equations, you can calculate the required spring widths:

$$\mathbf{wf1} = \mathbf{fd} / \mathbf{dfmax} * 3/2 * \mathbf{oe}$$

$$\mathbf{wd1} = \mathbf{fd} / \mathbf{dfmax} * 3/2 * \mathbf{oe}$$

$$\mathbf{wdf1} = \mathbf{fs} / \mathbf{dfmax} * 3/2 * \mathbf{oe}$$

Independently from the length of each beam and from the number of folds, this is the required width. Please, substitute the obtained values in **wf**, **wd**, and **wdf**. E.g., if **wf1** turns out to be 2 um, look for **wf** and write $2e-6 \text{ [m]} + \mathbf{oe}$.

Given geometry constraints, you can infer the maximum spring (beam) lengths:

$$\mathbf{lfmax} = \mathbf{ylimit} - 2 * \mathbf{hconnf} - \mathbf{gap} / 2$$

$$\mathbf{ldmax} = \mathbf{hmd} - \mathbf{wmdsx} - 3 * \mathbf{hconnnd} - \mathbf{gap} / 2$$

$$\mathbf{ldcmax} = \mathbf{lms} / 2 - \mathbf{lanc} - 2 * \mathbf{hconndc} - \mathbf{gap} / 2$$

Assuming maximum beam length, you can find the required number of folds

$$\mathbf{nff1} = \text{ceil}(\mathbf{E}^2 / \mathbf{kdf} * \text{thickness} * (\mathbf{wf1} / \mathbf{lfmax})^3)$$

$$\mathbf{nfd1} = \text{ceil}(\mathbf{E}^4 / \mathbf{kdp} * \text{thickness} * (\mathbf{wd1} / \mathbf{ldmax})^3)$$

$$\mathbf{nfdc1} = \text{ceil}(\mathbf{E}^4 / \mathbf{kdf} * \text{thickness} * (\mathbf{wdf1} / \mathbf{ldcmax})^3)$$

N.B.: With the given geometry, *the number of folds of all springs should be an **even number***.

Hence, given **nff1**, **nfd1**, **nfdc2**, change nff, **nfd**, **nfdc**, *accordingly*.

Once you have chosen the number of folds, you can infer the required length for each fold, to obtain the desired spring stiffness:

$$lf1 = wf1 * (E / (ktf * 2 / nff * thickness))^{1/3}$$

$$ld1 = wd1 * (E / (kdip * 4 / nfd * thickness))^{1/3}$$

$$ldc1 = wdc1 * (E / (ks * 4 / nfdc * thickness))^{1/3}$$

Now, you can compare these three lengths with the previously calculated maximum ones.

If all the lengths are lower than the maximum ones, you can substitute obtained values in the design ones. E.g., if **lf1** turns out to be 50 μm , you can change lf to $50e-6$ [m].

If one length is higher than the maximum one, you must change the number of folds, and re-calculate the corresponding length.

Once the values of **wf**, **wd**, **wdc**, **lf**, **ld**, **ldc** are *correctly changed*, you can solve your re-design (Study \rightarrow F8), and check whether the ± 1 kHz resonance frequency variation spec is satisfied (just re-plot the 1D Plot Group).

7. How to generate and save Movies

You may want to save the movies that you generate in Comsol Multiphysics. Below is a quick guide on how to do that. The solution should have been already performed at this point.

Click on *Mode Shape (Solid)*, in *3D Plot Group Setting* choose the *Eigenfrequency* (e.g., 20704 Hz).

In *Results*, right click on *Export*, choose *Animation*.

In *Animation Setting* \rightarrow Choose the *Filename* \rightarrow type "10" in *Frames per second* \rightarrow Change *Sequence type* in *Dynamic data extension* \rightarrow type "25" in *Number of frames* \rightarrow click on *Export*.

