

Synthesis of Ladder (IEF) filters

GVR Trade SA
(Written by **V. Plessky**, 1998 - 2014)

1. Introduction

This package of 35 MATLAB scripts and functions is designated for **AUTOMATIC SYNTHESIS OF LADDER (IEF) SAW FILTERS**.

The filters are constructed as a ladder of Impedance elements (SAW resonators). In current version of the software the ladder scheme is limited one of about 10 variants:

- ❖ **2Π** -sections , with 2 series connected resonators $R_1 \equiv R_s$, and 4 parallel resonators ($R_p \equiv R_2$)
- ❖ **N-2Π-N**, as above, but 2 Notch resonators are added in series with input and output (see **Fig.1 below**). Notch resonator has its own period and other parameters and is used to create additional notch in the passband.
- ❖ **Rs-2Π-Rs** as above, but notch resonator is the same as series resonator. This variant is used if very steep right hand side skirt is demanded by Spec.
- ❖ Etc.

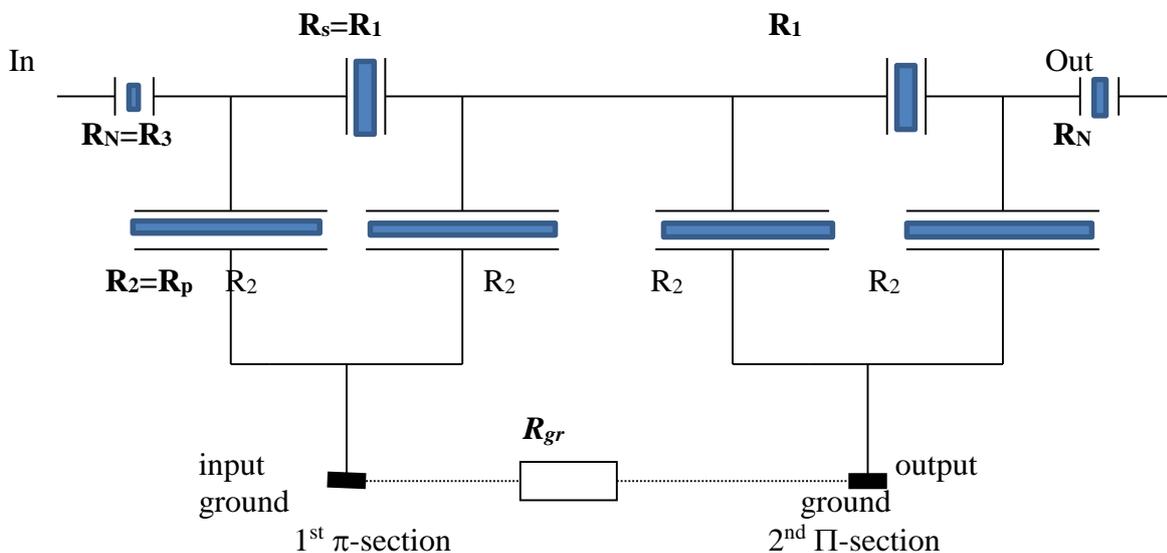


Fig.1 Electric scheme of the ladder filter

The input ground (first 2 parallel elements R_2) and the output ground (second 2 R_2) are separated in packaged device and the ground pins are connected through PCB, with small resistance R_{gr} . When probed on the wafer, there is a ground connection (ruined at dicing) which can be modelled by R_{gr} with higher value.

The Impedance element, or resonators, represent a long transducer limited on both ends by reflectors. The pitches of transducer p_t , reflector p_g , its aperture W , and the gap S between the

transducer and reflector (equal on both sides from the IDT) are considered as design parameters. The aperture W is currently the same for all elements (a factor can be introduced to change aperture of a particular resonator).

The synthesis (optimization) procedure is highly dependent on person doing it – it depends on your intuitive choice of substrate material, electrodes shape and duty factor, initial guess, goal function and its weights, constrains in FUN.m, (con.m in later versions) etc. – finally, on your experience. Local minimums also do exist, when reasonably good variant seems to be stable and final. Try again with strongly different initial data and check to what result it converges. Do not put too many restrictions at the beginning – to see what the best solution you can find without restrictions (such as limited chip size, for example). The final solution with restrictions will hardly be better.

The program fails in some cases with the diagnostics "Divide by zero". The reason is that the optimization routine can make different trial steps, even physically impossible, such as resistivity $R < 0$, or negative number of electrodes, etc., and in some of this cases the divergence occurs.

To avoid it, re-start the running with slightly different initial values.

2. General description of all scripts and functions

*possible changes in the scripts are written in rectangular areas,
as well as called subroutines*

The below description follows more or less the logic and sequence of calculations.

(Attention: Later versions may have modifications not described here)

2.1. **DODESIGN.m** is the main program. If you type DODESIGN in command MATLAB window followed by "Enter" key it starts the synthesis of the filter. In this script you should check:

2.1.1. **Path** to the library (if any used), %nt it , if you have no Lib.
2.1.2. **"CINEMA"** parameter- frequency of change of the pictures in figures
2.1.3. **Synch=0,1,2** – main optimization parameter
*dodesign.m calls: **FREQ.M**, **IN_DAT.M**, **SPECNUM.M**, **CONSTR('FUN.M')**- obsolete ,
changed later to **fmincon.m** and **PLSP.M***

2.2. If **DODESIGN.M** has been run and stopped, you can continue running from the same point using **CONT_RUN.M** instead of DODESIGN. If you need to change parameters of 2.1 do it always in DODESIGN routine.

*cont_run.m calls **DODESIGN.M** and puts **FLAG=1***

2.3. **CONT_RUN** reads current values of optimization variables from **CURR.DAT** file. CURR.dat file normally is NOT used for manual change-reading-writing of parameters. But it is possible.

2.4. Before starting DODESIGN.m program you should change the data in a few scripts: **FREQ.M** to specify the frequency range (not necessarily exactly as in Spec.) and the number of points

2.5. **IN_DAT.M** introduces the initial data for geometry of resonators, COM-parameters of substrate material, parasitics of package. Directly in **IN_DAT.M** you can change and/or check:

2.5.1. "Spectitl" – the header in pictures

- 2.5.2. Choose COM-parameter data file
 - 2.5.3. Choose package data
 - 2.5.4. Rin/Rout – environment
 - 2.5.5. Initial value of matching inductances L_{in-out} (if any)
 - 2.5.6. To (des)activate fitting of the matching inductances: **$KK_L = (0)I$** .
- In_dat.m calls GEOSYNCH.M, COMPAR=, Pack5x5.M*

- 2.6. Use script GEOSYNCH.M to introduce geometry of resonators
- 2.7. Use library files COM_XXX.M to introduce the COM-parameters for different materials/structures/frequencies etc.
- 2.8. Use **packSxS.m** file to introduce parasitics of packages (SxS = (e.g.) 3x3 –size).
- 2.9. **SPECNUM.M** script reads the SPEC.DAT file, describing the specification. Spec.dat file can be saved in the library with different, more descriptive, name.

specnum.m calls SPEC.DAT

- 2.10. **SPEC.DAT** file has a rigid structure: 22 rows and 3 columns : frequency and internal/external lines of the Spec. tafaret. See it and exactly follow the example. The spec structure: 4_leftside_suppressions, gap, passband, gap, 4_rightside_suppressions is quite general and applicable for most of the specifications. If you have less different suppression levels, subdivide one of them artificially into 2 or more. If you have more suppression levels, try to simplify spec, excluding frequency intervals, which are most probable satisfied automatically.

- 2.11. PLSP.M file plots the spec. onto figures. Here you can

- 2.11.1. change the frequency "**Shift**" on plots due to temperature variations,
- 2.11.2. Put Max=max(Loss) if suppression levels are relative, or MAX=0, if they are absolute

- 2.12. **FUN.M** forms the goal function: a sum of F1-deviation from spec, F2- deviation from needed SWR, F3- ripples in passband, and F4- min frequency gaps to critical points. NOTE: critical points are considered to be passband corners and the first close-in suppressions. In this script:

- 2.12.1. Change the goal value of SWR
 - 2.12.2. Introduce estimated frequency range for transition of the performance curve between critical levels of the Spec., for example from close in suppression level to passband level.
 - 2.12.3. Weights for different components F1-F4.
 - 2.12.4. Restrictions for variables (at the end of the script)
- Fun.m calls PARVAR.M, S_PARAM.M, FSPEC.M*

- 2.13. PARVAR.M script prepares parameters for use in FUN. and other routines. Dependence on formal X(i) optimization parameter is introduced here.

- 2.13.1. Know-how coefficient 1.02 in C3 can be layout dependent

- 2.14. S_PARAM.M is used to call scripts calculating P-matrixes of IDT and reflectors, adds the gaps, calls the RES.M program to calculate admittance of resonators

S_PARAM.M calls P_IDT.M, P_REF.M, RES.M, CURVE.M

- 2.15. P_IDT.M and P_REF.M are library functions used to calculate P-matrixes for transducers and reflectors.
- 2.16. RES.M is a function used to calculate the admittance of the one port resonator, when P-matrixes for the IDT and reflectors are already known.
- 2.17. CURVE.M finally calculates the performance curve (in dB), max. VSWR and ripple in the passband. Plots and "cinema" are done here. Current values of optimized

variables are saved here in CURR.DAT file, used by CONT_RUN.M routine to continue running of program after being stopped.

Calls Lad_PAR.M, PLSP.M and SMITH.M, creates CURR.DAT

2.18. Lad_Par.m script calculates impedances of resonators, adding leaky wave term *Gen*, and calls MATRIX##.M routine, wherein the Y-parameters of the selected Ladder network are found. Then adds some parasitics and calculates Y and S parameters of whole filter

Calls GEN.M, MATRIX??.M

2.19. GEN.M is a function describing bulk wave generation contribution to the admittance

2.20. MATRIX??.M exists in a few versions for different ladder schemes. It is basically a matrix of Kirchhoff's equations for currents in the ladder network. Calculates Y-parameters of the network, using solve6x6.m script.

Calls solve6x6.m

2.21. SOLVE6x6.M is a routine to solve 6 linear equations.

2.22. FSPEC.M forms the goal function, which includes a) term proportional to the deviations of the performance curve from the Specified levels b,c) SWR term and ripple, both found in CURVE.M, d) frequency gaps to most critical points of the Spec, using CR_POINT.M

Calls CR_POINT.M

2.23. CR_POINT.M finds the distance to most critical point of the Sec. 4 points are considered: passband corners and first suppression corners.

2.24. LIMX.M function is introduced to limit variations of some parameters and to avoid "divide by zero" message – not always works because used not for all parameters.

3. How to begin

3.1. Put all files into the same work directory. Keep one copy of software as archive in another "archive" directory — never change files of this copy.

3.2. Some universal files, such as *solve6x6.m*, *P_IDT.M*, *P_REF.M*, etc. you can put into your private Matlab library, but then you should include the Lib. into the Path (see DODESIGN.M)

3.3. Change Matlab directory to your work directory, by typing in command window: *cd path\to\your\work\dir*

3.4. Follow instructions of the next paragraph. If the issue is not clear look at the description above. Look also directly into the MATLAB scripts – they are commented in detail. The MATLAB scripts are just normal text files and can be edited by any editor, for example "NOTEPAD".

3.5. Type DODESIGN + "ENTER" in command MATLAB window to start program – in this case as INITIAL data the numbers from IN-DAT.M files will be used, or CONT_RUN + "ENTER" to continue running, if program has been run, but was stopped – in this case as the initial data the ones stored in CURR.DAT file are used

3.6. In case of more serious problem do not hesitate to contact the author,

Dr. Victor P. Plessky:

GVR Trade SA
Switzerland

Email victor.plassky@gvrtrade.com
Or victor.plessky@gmail.com

4. What to do with a new SPEC:

4.1. *Basic changes – absolutely a MUST*

- ◆ Change **Spec.dat** file following exactly the provided sample. It must have 22 lines (frequency points) and 3 columns: frequency, internal spec's limits, external gabarit)
 - The structure is rigid: 4 suppression levels (2 frequency points for each level: beginning and end), gap, passband, gap, 4 suppression levels
 - If in the real *Spec* there is, for example, less than 4 different suppression levels on the left side, subdivide the interval artificially, so that you have 2 or more frequency ranges with the same level of suppression
 - Note that in pairs of frequency points (frequency interval) the suppression levels are equal.
- ◆ Change frequency range in **FREQ.M** file
- ◆ Change (or delete, using % percent sign) the **PATH** to library files (if you have such)
- ◆ *(Already at this stage you can try to run the program, typing "dodesign"+ "Enter". Normally the program will show some previous variant of some filter, determined by **GEOSYNCH.M** file –geometry, and **COMpar** – choice of material.)*
- ◆ Give some reasonable initial geometrical values to the resonator's parameters in **GEOSYNCH.M** script. Convenient (easy) way to do it is just to scale parameters (mainly pitches) of a similar filter (*on the same substrate material!*) with other center frequency, if such an example is available. If not, take for series resonators $pt10= V0/(2Fc)*(1-\kappa/2\pi)$ and for parallel resonators $pt20=V0/(2Fc)$; $V0$ - is SAW velocity, Fc –center frequency, κ - reflectivity of fingers (per pair). Change manually roughly the number of fingers in parallel transducer to get reasonable suppressions.
- ◆ Change (select) **COMpar** file in **IN_DAT.M** file, if needed
- ◆ Check that the package parasitics file (e. g. **Pack5x5.m**) corresponds to a package type you are going to use.
- ◆ Observe physical units: we use MHz, μm , pF, nH, Ohms
- ◆ *Try to run the **DODESIGN**. It should start to work reasonably. Try to keep initial value of **FUNCTION** (displayed in **COMMAND MATLAB** window) as low, as possible, selecting proper initial geometry. Ideally, it must be close to **UNIT**. **The MATLAB optimization routine will probably not work if the initial guess is too bad , for example, goal function is 10^4 or higher.***
- ◆ You can stop program (by Ctrl+C) and than continue to run from the same point (that is the same values of independent optimized variables) , using **CONT_RUN** command. Periodically use **CLEAR ALL** command (especially after changing *Synch* parameter)– it doesn't erase current values saved in **CURR.dat** file.

4.2 Very desirable changes

- ◆ Select (re-write) the **Synch** parameter in DODESIGN script : **Synch=0**. That choice will limit the geometry of resonators to the "synchronous" type, that is only pitches, and numbers of fingers will be changed (and aperture W). No gaps change, no separate change in reflectors periods.
- ◆ Put $S10=S20=S30 =0.5$ (in geoSynch.m) , put **pg0=pt0**, the same pitch in gratings, as in IDTs, for all resonators.
- ◆ Select type of Ladder network in LAD_PAR.M, choosing matrix6, or mat2pi, etc.
- ◆ Run the synthesis with limited number of variables, with synchronous resonators as building blocks. You can always stop program, make whatever changes, e.g. put Synch=1, which will introduce the gap's variation, and continue to run it using CONT_RUN command.
- ◆ You can introduce gradually the change of gaps, Synch=1, and the independent change of pitches in reflectors (synch=2) if needed to improve filter performance.

4.3 Cosmetic changes

- ◆ You can change "cinema" parameter in DODESIGN routine to make changes of figures faster. Normally they are changed after every 100 internal optimization steps.
- ◆ Change SPECTITL character variable to get correct titles on top of figures.
- ◆ Change Shift parameter in PLSP.M, to indicate larger or narrower tolerances in SPEC. Current value corresponds only to the TEMPERATURE shifts, not including other possible components (aging, for example)
- ◆ Put $Max=max(Loss)$ in PLSP.M if your spec is formulated in relative suppressions. Current option $Max=0$ corresponds to absolute levels of suppressions.

4.4 Master level of design

- ◆ The freedom you have: choice of material, Al thickness and metallization ratio, choice of ladder network, weights in goal function.

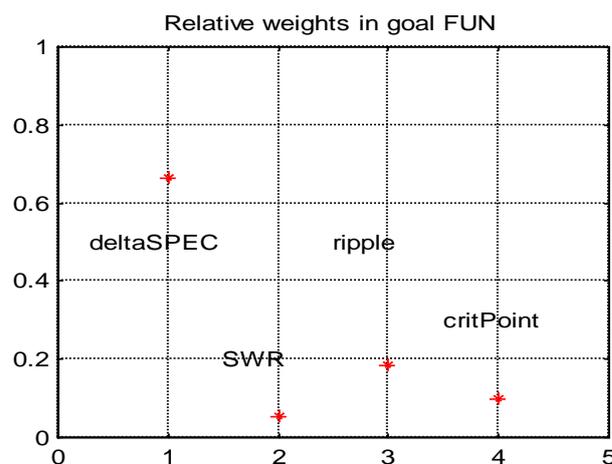


Fig.2

- ◆ Fig. 2 (here and on display) shows *relative weights* of different components of goal function: deviation from the spec., VSWR, passband ripple, and min. distance to critical corners of the spec. Changing weights of components in FUN.M file you can set you

preferences: very flat passband, or low SWR, or.... Note that the change of the number of frequency points N_{point} in `freq.m` changes the relative weights, because only the first term ("deviation from spec") is proportional to the number of points. Keep all components nonzero.

- ◆ Although it depends on you aim, try to keep the weights in Fig.2 at reasonable levels. If, for example, the VSWR term dominates, you will hardly design a good filter in spec. Basically, the first term must always be available at level 30% to 50% or even higher.
- ◆ Last term, min gaps, mathematically is different from 3 previous ones. All 3 first terms tend to zero for most optimal case, remaining positive. It is NOT so for the 4th term, which is simply minimized, but NOT NECESSARILLY TENDS TO ZERO. Because of that this term (and even all function F, can, in principle, become negative. To keep min. of this term close to Zero, select carefully **TRANS** parameter in FUN.M
- ◆ The suppression levels are transferred to FUN.M directly from SPEC.DAT file, with 3 dB increase. You CAN change this additional suppression in FSPEC.M script and in this way you can introduce arbitrary suppressions in all 8 intervals. In principle, you can move here the boundaries of the intervals. Currently the passband is artificially increased at 2dfT – twice temperature shifts, while the most close suppression points are moved toward the passband at dfT each.
- ◆ The number of electrodes in IDT and reflectors is supposed to be integer. If in optimization process non integer numbers are used, one should imagine that the boundary or reference planes remain at the center of electrodes, always separated by integer number of pitches. Although it is not very important for synchronous resonators, it is better to keep the number of fingers EVEN in the IDTs. (The reason- in formulas the IDT are supposed to be symmetric)
- ◆ Remember that the program DOESN'T change substrate material and electrode's geometry itself. The success of design and the stability of parameters to technological changes may depend on COMpar file choice .
- ◆ The program can design filters with different input and output impedances. You can change reference impedances in **In_dat.m** file.
- ◆ Parallel matching inductance (equal at both in/out puts) can be introduced and optimized. See end of **In_dat** script. To activate optimization of the matching inductance put **KK_L=1**.
- ◆ Note that the CURR.dat file is written every time when "cinema" pictures are changed. It means that the file corresponds to 100N internal search step of optimization routine, not necessarily, coinciding with the optimized step really done by the program and manifested in COMMAND window. It means that the difference between the goal function **stop** value you see in the command window and the **restart** fun. value is possible. The difference is not predictable and can be rather big , especially if the run number of iterations is not too big.
- ◆ Analysis of sensitivity to technology variations currently can be done only manually: change COM parameters and re-start program by "DODESIGN". Use "HOLD on" command to keep old curve in the figure.

5. A list of modifications of version 1.0

5.1. Version 1.1:

5.1.1.

To be added

6. Possible improvements, partly included but not described in this document

- 6.1. Automatic analysis of sensitivity to technological variations
- 6.2. Increased library of "substrate material" and "package parasitics" files
- 6.3. Hic_Up resonators as a building blocks
- 6.4. Additional attenuation and conductivity due to bulk wave scattering / generation will be described by separate routine, more exactly modelled. Dependence on aperture, number of fingers etc. will be introduced.



15 May. 14, Gorgier, Switzerland