Daniel DiMaggio

Dr Kwok
EE 172
12/19/12

Canonical Dual Mode Waveguide Cavity Filter

Table of Contents

Abstract

An electronic filter is a circuit that is designed to only allow specific frequencies to be useable, while blocking unwanted frequencies. Commonly, passive elements (resistors, inductors, and capacitors) are used to make filters. However, these types of filters are not as useful when dealing with higher frequencies. An elliptical dual mode waveguide can act as a passband filter for higher frequencies.

There are two types of elliptical filters, canonical and longitudinal. The main difference between the two is the the orientation of the elements. A longitudinal waveguide has the elements move linearly through the cylinder whereas a canonical waveguide goes through the center iris of the cylinder and then comes back. The figures below show the two types of filters.

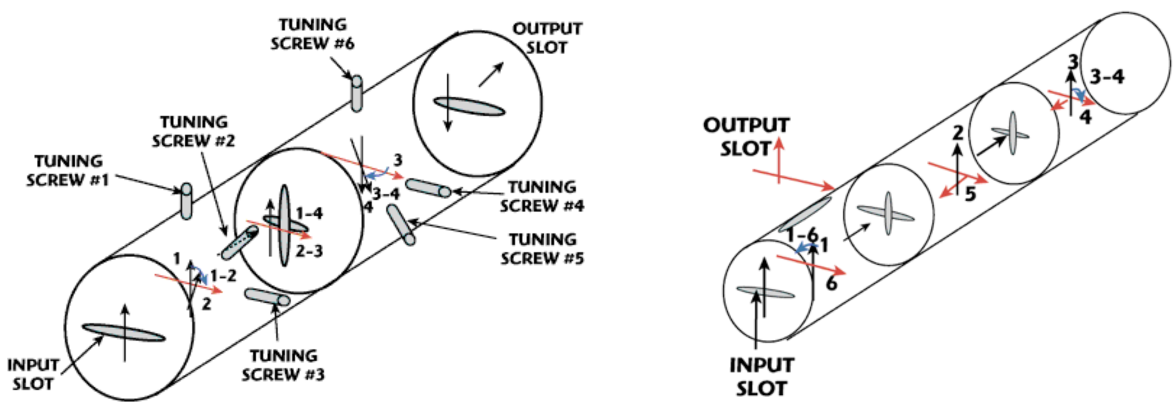


Figure 1 A four element longitudinal filter (left) and a six element canonical filter (right) [1]

Introduction

An elliptical filter is a filter that is most useful due to its steep transition between the passband and stopband. Other filter types such as a Butterworth or Chebyshev filter will require a high order filter to achieve a steepness similar to an elliptical or Cauer Filter. However, the steepness comes at a cost of non flatness in the stopband regions. Examples of filters of different types are shown in figure 2.

There are many advantages to using a cylindrical elliptical filter at higher frequencies including lower power loss, reduced size, and higher quality factor in comparison to rectangular resonators. The basis for the design comes from a patent filed on October 10, 1972. A design included in the patent is seen in figure 3 [2].

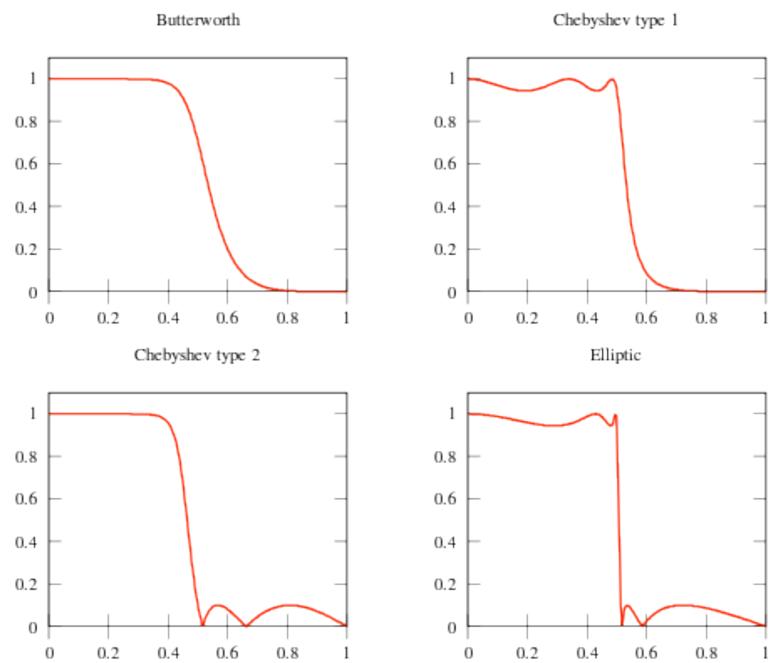


Figure 2 Fifth order low pass filters

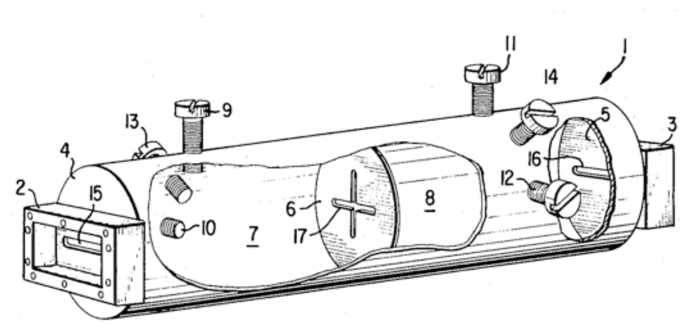


Figure 3 Plural cavity bandpass waveguide filter patented in 1972 by Bruno Blachier and Andre Champeau

Design Process

To build an elliptical filter you must start by using a cylindrical object that has a wall dividing two equal halves. Because of its cost, two soda cans were recommended for the project. The diameter of the cylinder determines the frequency of the passband. A soda can has a diameter of 2.5 inches which allows a frequency of approximately 2.9 GHz to pass.

For a canonical waveguide, the center wall must have a plus sign cut out of it. This will allow the wave to pass through only if it matches the slit (side ways or up and down). Slits sideways and up and down are needed because when element one passes to the other side its orientation is opposite of the orientation of element three passing back to the first half. In a longitudinal waveguide, only one slit is needed because only element two passes to the other half.

On the first can, two screws and two co-axial cable connectors are needed. The placement of these is shown in figure 4. It is recommended that two nuts for each screw be used to hold the screws in place while tuning. The first nut should be attached to the outside of the can over the hole and the second screwed in to keep the screws in place when tuning is complete.

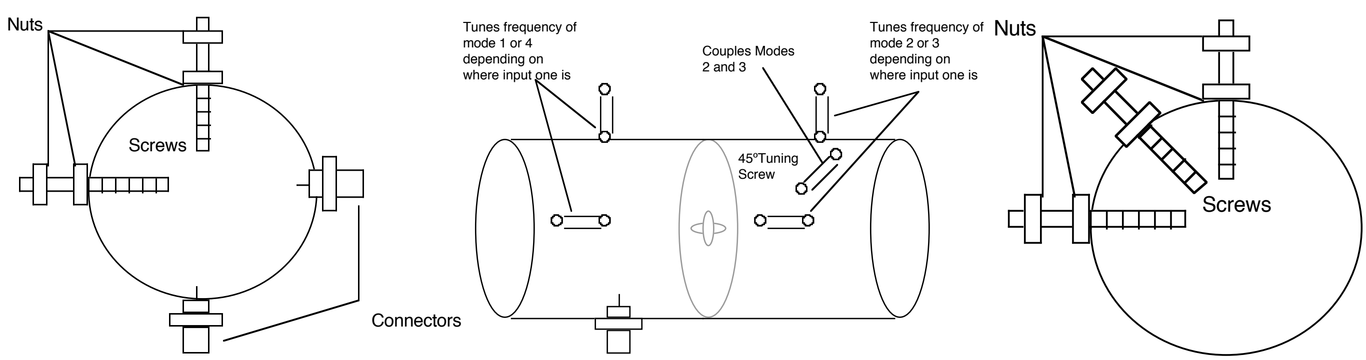


Figure 4 Design of the canonical filter, front view of first can (left) side view (center) and front view of second can (right)

The second can needs three screws for the canonical design. The screws placed 90 degrees apart tune the frequency of modes two and three, while the screw placed between the other two screws couples those two modes. When tuning, this screw will combine the peaks caused by these two modes.

If using a soda can, the connectors will likely prove to be the most difficult part. They will likely dent the can slightly which will ruin the cylinder and the filter will not work properly. This is why a second filter was attempted (discussed later) that would be more stable.

Figure 5 Soda can canonical waveguide filter 2.5" diameter

Figure 5 above shows the final design using two soda cans. Typically waveguides are made from metal that have low bulk resistivity [3]. It was recommended that brass screws be used for this design. Metal tape was used to attach the connectors and nuts to the can over each hole. The design worked as it was supposed to and did yield a passband at the frequency expected.

A second waveguide was attempted using coffee cans. It was designed exactly the same way as the soda cans but did not yield expected results. The likely problems was the size of the screws, the size of the iris, or the lack of a cylindrical shape.

Figure 6 Coffee can waveguide filter 4" in diameter

Results

The soda can filter worked as expected. Figure 7 on the right shows the four peaks corresponding to the four modes. On both sides of the peaks two “zeros” also called reject notches can be seen. This is what makes the filter so useful. Through the use of control theory, these zeros are placed and allow the peaks to occur much more sharp which results in a higher quality factor. The quality factor goes up with the increasing number of elements and also makes the passband more narrow.

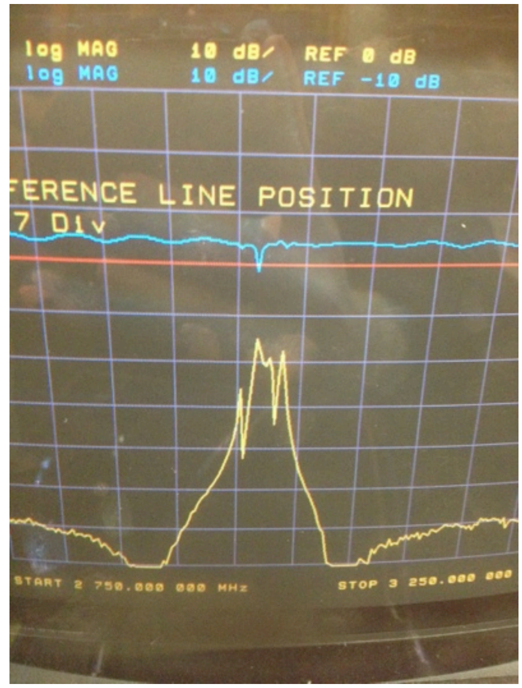


Figure 7 Measurements from the Network Analyzer showing S11 and S21 that shows the four peaks from the four elements

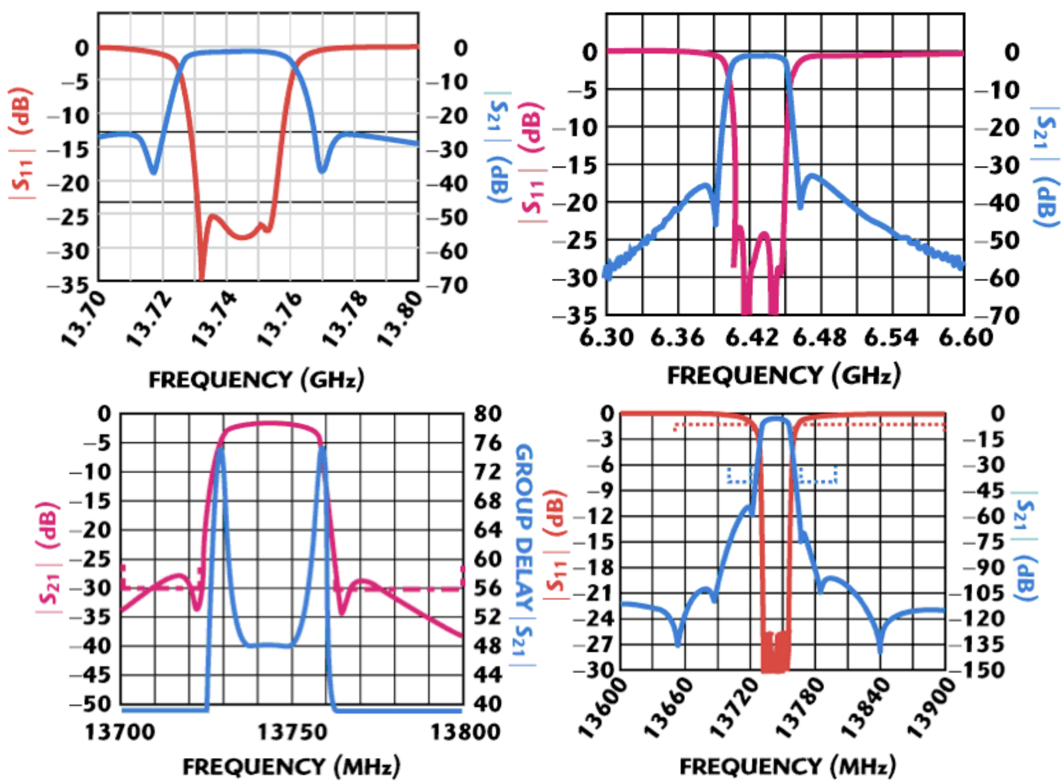


Figure 8 Simulations of multiple element dual mode responses (clockwise from top left, 4 elements, 5 elements, 6 elements, and 8 elements

Figures 9 and 10 show the results of the filter after attempting to tune the screws and calibrate the machine. All four peaks were not successfully combined into one sharp peak but the value of 2.93 GHz did have an insertion loss of only -.5749 dB. Figure 8 shows what happens when the you attempt to combine the left most peak with the larger peak. The zeros become much more defined, but the sharpness of the peaks decrease.

Figure 9 After tuning and calibration of the network analyzer, three peaks have been combined

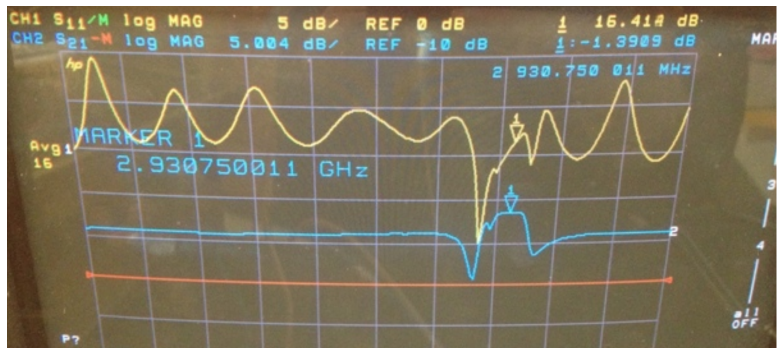


Figure 10 Shows how if left most peak is brought to the right, the one large peak moves to the right

Conclusion

While the soda can filter was build properly and worked as expected, tuning of the can was an even greater challenge than expected. Tuning requires a substantial amount of patience and skill that can only be gained through practice and training. Adjustments to the angle of the screws may be necessary to achieve proper tuning.

The design for an elliptical filter has been around for more than forty years and is still useful today. The different variety of filters available are useful for different purposes. A steeper transition period between the stopband and passband can be incredibly useful but comes at a cost. This project showed a cheap and effective way to achieve an elliptic bandpass filter for high frequencies.

References

[1] Lapidus, Alex (2008). Circuit Simulation of Dual Mode Waveguide Cavity Filters. Accessed Nov. 14, 2012 from Microwavejournal.com.

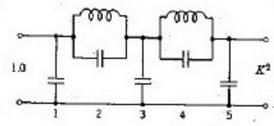
[2] Blachier, Bruno and Andre Champeau (1972). Plural Cavity Bandpass Waveguide Filter. United States Patent Office.

[3] P-N Designs, Inc. (2012). Waveguide Construction. Accessed December 1, 2012 from Microwaves101.com.

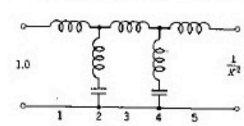
Phyo, Wai. EE 172 Project: 5th Order Elliptic Band-Pass Filter.

Simulation

I have come across an old paper (today) by Wai Phyo that showed a simulation done with lumped elements. After going through my old EE 185 notes I believe this can be a Cauer I filter that has one pole and two zeros. I believe this means the equation should be: z(s) = (s+a)(s +b) / (s+c). I remember that the realization for a Cauer I involved finding out the value of z(s) when s=0 and s=infinity. In this case, s=0 yields z=a constant and s=infinity yields infinity. So the first element should be an inductor if this is an LC circuit. So it should be a series inductor, a shunt capacitor, and finally a series inductor. I know how to calculate the values using the matrix method but I can't unless I have an equation for z(s) or y(s). But I was under the impression this only worked for low frequencies so I don't think this would be the proper way to simulate. My best guess would be that 4 peaks means 4th order filter which means the numerator should be s^4.



$K^2=1.0$							$K^2=\infty$						
C_1	C_2	L_2	C_3	C_4	L_4	C_5	C_1	C_2	L_2	C_3	C_4	L_4	C_5
0.9732	0.0000	1.372	1.802	0.0000	1.372	0.9732	0.4866	0.0000	1.1388	1.4496	0.0000	1.5271	1.3108
0.97315	0.00006	1.37219	1.80298	0.00020	1.37200	0.97302	0.4865	0.0001	1.1387	1.4494	0.0002	1.5264	1.3107
0.97296	0.00031	1.37192	1.80240	0.00080	1.37117	0.97245	0.4863	0.0004	1.1383	1.4488	0.0007	1.5260	1.3104
0.97265	0.00069	1.37148	1.80144	0.00181	1.36978	0.97153	0.4858	0.0008	1.1377	1.4479	0.0016	1.5245	1.3097
0.97221	0.00123	1.37086	1.80009	0.00322	1.36784	0.97022	0.4852	0.0015	1.1368	1.4466	0.0029	1.5225	1.3088
0.97164	0.00192	1.37007	1.79835	0.00503	1.36535	0.96854	4.4845	0.0023	1.1356	1.4449	0.0045	1.5198	1.3076
0.97095	0.00277	1.36909	1.79624	0.00726	1.36230	0.96649	0.4835	0.0033	1.1343	1.4428	0.0065	1.5166	1.3062
0.97014	0.00377	1.36794	1.79374	0.00989	1.35870	0.96406	0.4824	0.0046	1.1326	1.4404	0.0089	1.5128	1.3045
0.96920	0.00493	1.36662	1.79086	0.01295	1.35455	0.96126	0.4811	0.0060	1.1307	1.4376	0.0116	1.5085	1.3025
0.96813	0.00624	1.36511	1.78759	0.01642	1.34985	0.95808	0.4797	0.0076	1.1286	1.4344	0.0147	1.5035	1.3003
0.96693	0.00772	1.36342	1.78395	0.02031	1.34460	0.95451	0.4780	0.0093	1.1262	1.4309	0.0182	1.4980	1.2978
0.96561	0.00935	1.36156	1.77993	0.02464	1.33880	0.95061	0.4762	0.0113	1.1235	1.4269	0.0221	1.4919	1.2951
0.96416	0.01114	1.35952	1.77553	0.02941	1.33245	0.94630	0.4742	0.0135	1.1206	1.4226	0.0264	1.4852	1.2921
0.96258	0.01310	1.35729	1.77076	0.03461	1.32555	0.94163	0.4721	0.0159	1.1175	1.4180	0.0310	1.4780	1.2889
0.96087	0.01522	1.35489	1.76561	0.04028	1.31811	0.93657	0.4697	0.0185	1.1140	1.4130	0.0361	1.4701	1.2854
0.95903	0.01750	1.35230	1.76009	0.04640	1.31012	0.93115	0.4672	0.0213	1.1104	1.4076	0.0416	1.4617	1.2816
0.95706	0.01996	1.34953	1.75421	0.05299	1.30160	0.92534	0.4645	0.0234	1.1064	1.4018	0.0475	1.4528	1.2776
0.95496	0.02258	1.34658	1.74795	0.06007	1.29253	0.91915	0.4616	0.0276	1.1022	1.3957	0.0538	1.4432	1.2733
0.95272	0.02537	1.34344	1.74134	0.06764	1.28292	0.91259	0.4585	0.0310	1.0977	1.3892	0.0606	1.4331	1.2688
0.95035	0.02833	1.34012	1.73436	0.07572	1.27277	0.90565	0.4552	0.0347	1.0930	1.3824	0.0678	1.4225	1.2640
0.94785	0.03148	1.33661	1.72703	0.08432	1.26209	0.89833	0.4517	0.0387	1.0880	1.3752	0.0754	1.4113	1.2590
0.94521	0.03480	1.33292	1.71933	0.09346	1.25088	0.89062	0.4481	0.0428	1.0828	1.3677	0.0835	1.3995	1.2537
0.94244	0.03830	1.32903	1.71139	0.10316	1.23913	0.88253	0.4442	0.0473	1.0772	1.3598	0.0921	1.3872	1.2482
0.93953	0.04198	1.32495	1.70290	0.11343	1.22685	0.87406	0.4402	0.0519	1.0714	1.3516	0.1013	1.3743	1.2424
0.93647	0.04585	1.32068	1.69416	0.12429	1.21406	0.86521	0.4359	0.0569	1.0653	1.3430	0.1109	1.3609	1.2364
0.93328	0.04993	1.31622	1.68509	0.13577	1.20074	0.85596	0.4314	0.0621	1.0590	1.3341	0.1210	1.3469	1.2301
0.92994	0.05420	1.31156	1.67562	0.14789	1.18689	0.84633	0.4266	0.0675	1.0524	1.3248	0.1317	1.3324	1.2236
0.92646	0.05866	1.30670	1.66594	0.16068	1.17253	0.83631	0.4219	0.0733	1.0455	1.3152	0.1430	1.3173	1.2169
0.92284	0.06333	1.30164	1.65587	0.17418	1.15766	0.82590	0.4168	0.0794	1.0383	1.3053	0.1549	1.3017	1.2099
0.91906	0.06821	1.29637	1.64548	0.18840	1.14228	0.81509	0.4114	0.0858	1.0308	1.2950	0.1674	1.2856	1.2027
0.91514	0.07330	1.29090	1.63478	0.20339	1.12630	0.80388	0.4059	0.0925	1.0231	1.2845	0.1805	1.2690	1.1952
0.91106	0.07862	1.28522	1.62377	0.21919	1.11000	0.79227	0.4003	0.0995	1.0150	1.2736	0.1944	1.2518	1.1875
0.90683	0.08416	1.27933	1.61246	0.23584	1.09311	0.78026	0.3941	0.1070	1.0067	1.2624	0.2089	1.2341	1.1796
0.90244	0.08993	1.27322	1.60084	0.25339	1.07573	0.76784	0.3878	0.1147	0.9980	1.2508	0.2242	1.2159	1.1714
0.89789	0.09594	1.26686	1.58895	0.27188	1.05785	0.75502	0.3815	0.1229	0.9891	1.2390	0.2402	1.1972	1.1630
120mHz 0.89318	0.10220	1.26033	1.57677	0.29139	1.03850	0.74177	0.3745	0.1315	0.9798	1.2269	0.2571	1.1780	1.1544
0.88830	0.10871	1.25354	1.56431	0.31197	1.02066	0.73811	0.3674	0.1404	0.9703	1.2145	0.2749	1.1583	1.1456
0.88325	0.11549	1.24661	1.55159	0.33370	1.00135	0.71409	0.3601	0.1499	0.9604	1.2018	0.2935	1.1381	1.1365
0.87803	0.12254	1.23929	1.53862	0.35665	0.98158	0.69851	0.3525	0.1598	0.9502	1.1888	0.3133	1.1175	1.1272
0.87263	0.12987	1.23173	1.52540	0.38092	0.96134	0.68456	0.3447	0.1702	0.9397	1.1755	0.3340	1.0963	1.1177
0.86706	0.13749	1.22397	1.51195	0.40650	0.94065	0.66917	0.3365	0.1812	0.9288	1.1619	0.3559	1.0747	1.1080
0.86130	0.14542	1.21594	1.49827	0.43360	0.91952	0.65332	0.3280	0.1927	0.9176	1.1481	0.3790	1.0526	1.0980
0.85535	0.15367	1.20763	1.48438	0.46265	0.89794	0.63702	0.3192	0.2046	0.9061	1.1341	0.4033	1.0300	1.0879
0.84920	0.16225	1.19905	1.47030	0.49328	0.87594	0.62025	0.3101	0.2176	0.8942	1.1198	0.4291	1.0070	1.0776
0.84286	0.17119	1.19018	1.45603	0.52586	0.85352	0.60301	0.3006	0.2310	0.8820	1.1052	0.4564	0.9835	1.0670
0.83631	0.18048	1.18101	1.44159	0.56057	0.83069	0.58528	0.2907	0.2452	0.8694	1.0904	0.4852	0.9596	1.0563
0.82956	0.19016	1.17152	1.42699	0.59759	0.80746	0.56704	0.2805	0.2601	0.8564	1.0754	0.5159	0.9353	1.0453
0.82258	0.20025	1.16170	1.41227	0.63717	0.78385	0.54830	0.2699	0.2759	0.8430	1.0602	0.5485	0.9106	1.0342
0.81538	0.21076	1.15155	1.39742	0.67957	0.75986	0.52902	0.2589	0.2927	0.8293	1.0448	0.5833	0.8854	1.0228
0.80795	0.22173	1.14103	1.38249	0.72506	0.73551	0.50920	0.2474	0.3104	0.8151	1.0293	0.6202	0.8599	1.0114
0.80029	0.23317	1.13013	1.36748	0.77404	0.71083	0.48882	0.2355	0.3292	0.8006	1.0135	0.6598	0.8339	0.9998
0.79237	0.24510	1.11884	1.35242	0.82687	0.68582	0.46786	0.2231	0.3492	0.7856	0.9978	0.7022	0.8076	0.9880
0.78420	0.25765	1.10713	1.33735	0.88402	0.66050	0.44629	0.2102	0.3704	0.7700	0.9815	0.7477	0.7809	0.9760
0.77576	0.27075	1.09497	1.32228	0.94602	0.63490	0.42410	0.1966	0.3931	0.7541	0.9653	0.7967	0.7539	0.9639
0.76705	0.28446	1.08235	1.30726	1.01351	0.60905	0.40124	0.1828	0.4174	0.7377	0.9490	0.8498	0.7266	0.9516
0.75805	0.29891	1.06922	1.29232	1.08722	0.58296	0.37770	0.1682	0.4434	0.7208	0.9326	0.9068	0.6989	0.9393
0.74875	0.31408	1.05556	1.27750	1.16804	0.55666	0.35343	0.1530	0.4713	0.7034	0.9162	0.9690	0.6710	0.9268
0.73914	0.33006	1.04132	1.26284	1.25698	0.53019	0.32840	0.1371	0.5014	0.6855	0.8997	1.0368	0.6428	0.9142
0.72920	0.34694	1.02646	1.24841	1.35531	0.50358	0.30256	0.1204	0.5338	0.6671	0.8831	1.1111	0.6143	0.9015
0.71892	0.36479	1.01097	1.23424	1.46451	0.47688	0.27587	0.1030	0.5690	0.6481	0.8666	1.1926	0.5856	0.8888
0.70829	0.38374	0.99471	1.22041	1.58638	0.45013	0.24827	0.0847	0.6073	0.6285	0.8502	1.2826	0.5567	0.8761
L_1	L_2	C_2	L_3	L_4	C_4	L_5	L_1	L_2	C_2	L_3	L_4	C_4	L_5



The way suggested to simulate this most often was with Genesis. A code I found was

```
f=FREQ/1000
l1= 14.2
l2=17.2
F0=13.7425
Fcoin=7.87
Fcoout=7.87
fco1=9.261
fco2=10.943
Zw1=377/SQR(1-(fco1/f)^2)
Zw2=377/SQR(1-(fco2/f)^2)
Zin = 377/SQR(1-(fcoin/f)^2)
Zout=377/SQR(1-(fcoout/f)^2)
WGWL1=300/SQR(f^2-fco1^2)
Tet1=0.5*l1*360/WGWL1
WGWL2=300/SQR(f^2-fco2^2)
Tet2=0.5*l2*360/WGWL2
```

At the very least I can tell that F0 should be changed to 2.9.