Table of Contents


```
clear
```

First steps

- define spec values and calculated/rediresd param-values in this file "parallelFiler.m
- Run the file.
- This script will calculate/define the design spec for the desired filter.
- Then will call "nfilter3.m" to calculate the structural params of the device using the stirght forward theory.

```
lam1 =1530*(10^(-9)); % Wavelength for the first channel of each path,
 (design spec)
nt=16; %total number of channels (design spec)
n1=8; % numebers of channles in on path (design param)
np=nt/n1; % number of pathes
dlamCh=2*(10^(-9)); %(design spec)
dlamTot=(nt-1)*dlamCh;
lam(1,1)=lam1;
nch =0;
for i=1:1:np
    for j=1:1:n1
        nch=nch+1;
        lam(i,j)=lam(1,1)+(nch-1)*dlamCh;
    end
end
lam(1,8);
fsr=lam(2,1)-lam(1,1);
m=lam(1,1)/fsr;
m=vpa(int64(m))-2; % design param, number 2 is just choesn intuitively
 and trial and error.
neff = 1.6532; %(design spec/param)
BW = 25*(10^(-10)); %(design spec)
fsr=lam(1,1)/m;

% creating data output structur3 with time stamp
dateStamp = datestr(now,'ddmmmyy_HHMM');
```

```
designFilesDir='designSpecs';
mkdir(designFilesDir);
designFile=strcat('./',designFilesDir,'/
designSpecPara_',dateStamp,'.txt');
outID=fopen(designFile,'a+');

% Calling nchfilter3.m for each path to calculate design parameters
 for
% each path.
```

```
Warning: Directory already exists.
```

nfilter3.m. Uses Straight forward theory and design specs

params

1. neff: material/structure dependent.
2. m: wavelenght number, m=Lam/FSR.

Spec

1. n: Number of channels.
2. lam11: Wavelength window = [lam11, lam11+FSR1]
3. BW; This will set the minimun cross-couplingvalue, which is related to the gap between the waveguide and the ring.
4. The limits on losses will put limits on the ring sizes. I haven't considred this effect here for now.
5. FSR: is defined based on wavelength diff between channles and the fist wavelength. (FSR > n*lam11, where n is the number of channels in one path.)

Design constraints:

1. Avoid resonant spliting; which gives us the maximun value for cross-coupling coeff.
2. Lower crosstalk which calls for high-Q filter, i.e. higher ordef filters.

```
Note:
Spec is to have the 8 channles in the first channel's first FSR interval
Meaning the channels window is [lam1 lam1+fsr1]
n=8 for an 8-ch filter
We chose to have equally spced channeles with wavelength difference as dla
```

Function's Inputs

1. designFile: which is the full filename (including the path) of output file. This output file contained required netlist params, design spec and some output info about the channel wavelength, FSRs, FWHMs.

2. neff: Index of reflection which is choosen.
3. lam11: Wavelength for the first channel (comes from desin spec)
4. dlam1: Space between channels (comes from desin spec). We have chosen to work on equally spaced channels. It is simple to accomodate not-equally-spaced channels if needed.
5. m: Wavelength number, It is chosen based on wavelength window size. (The calculation is done in parallerFilter.m.
6. n: Number of channels which is given as design spec.

Function's Outputs:

1. Ring sizes,
2. cross-coupling coef.
3. Will also print the calculated, channel wavelenght, FSRs and ... at location "./designSpecs/designSpecPara_<TIME STAMP>.txt"

```
for i=1:1:np
    fprintf(outID,'\n\n************     PATH %d     ************\n
\n',i);
    nchfilter3(designFile,neff,lam(i,1),dlamCh,eval(m),n1, BW)
end
```

Next steps

- Next is updating the netlist files with the calculated design params at *./designSpecs/designSpecPara<TIME STAMP>.txt* .
- Then run the netlist files.
- Then use the ploting scripts
- * chParallerChecks.m which will call PlotAllCrossTalkImvCh8.m

Published with MATLAB® R2015b