Zero Pad Commit Request 4p9_4

Richard Mellitz, Samtec

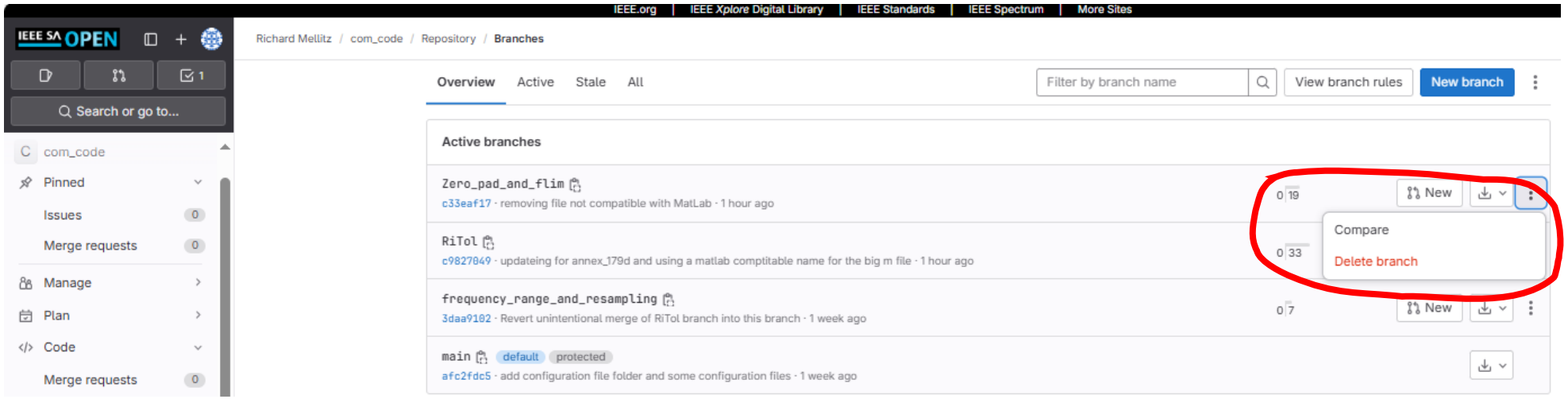
Adam Gregory, Samtec

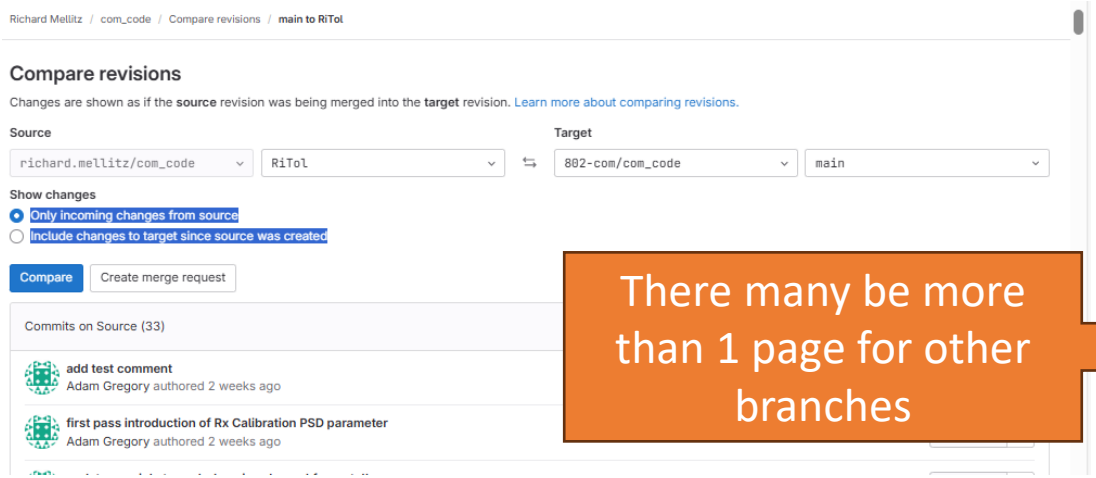
IEEE 802.3 Channel Operating Margin (COM) Open Source Project Ad Hoc,
New Orleans, LA May 2025 Interim

Purpose for Commit Request

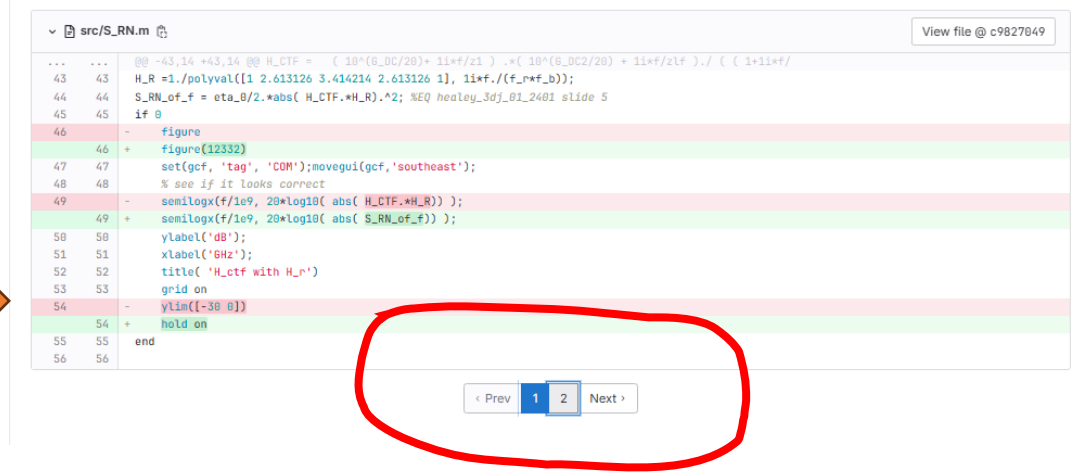
- ❑ Reduce delta COM and ERL indicated in mellitz_3dj_03_2505 using zero padding before performing the iFFT in the COM code
- ❑ Provide a mechanism to trim the frequency range of passed s-parameters
  - Useful for the debug of the zero-pad option
- ❑ Branch: Zero_pad_and_flim
- ❑ Branch test code (will disappear after merge)
  - https://opensource.ieee.org/richard.mellitz/com_code/-/raw/Zero_pad_and_flim/release/com_ieee8023_4p9p1_Zero_pad_and_flim.m?ref_type=heads&inline=false

Checking changes of a branch to the last release



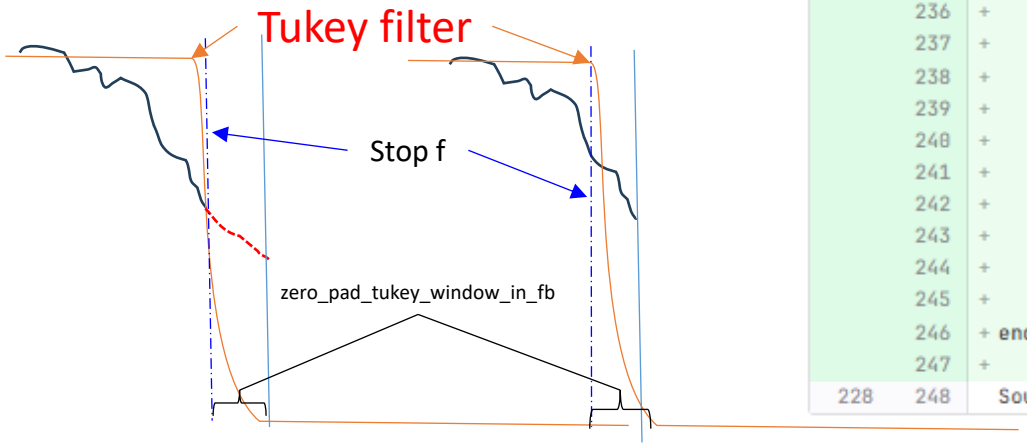


There many be more than 1 page for other branches



interp_Sparam

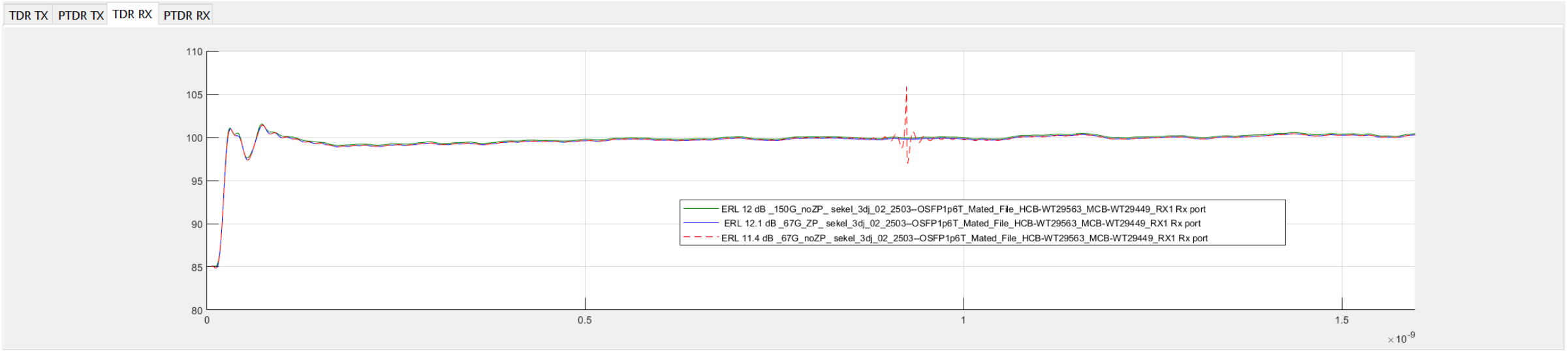
- Add param to call
- Zero padding : Add a near zero number (eps(0) to the IFFT required frequency extension beyond the data (Sin) provided.
- if keyword zero_pad_tukey_window_in_fb is not zero a Tukey filter is applied before the zero padding.
- zero_pad_tukey_window_in_fb can be positive or negative in terms and in terms of fb

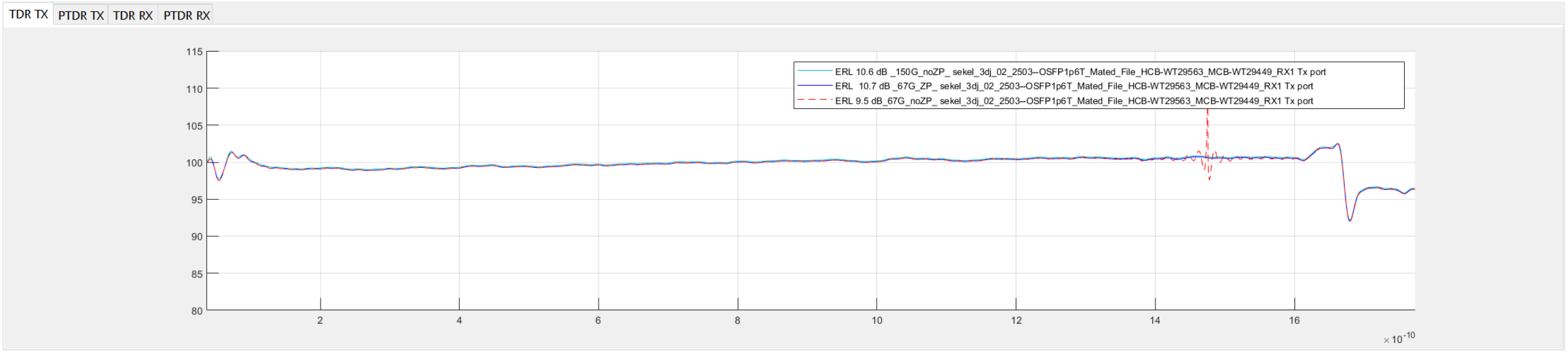


src/interp_Sparam.m — View file @ aa184b1e

```
-  1    function [Sout] = interp_Sparam(Sin,fin,fout, ...
-  2        opt_interp_Sparam_mag, opt_interp_Sparam_phase,OP)
+     1  function [Sout] = interp_Sparam(Sin,fin,fout, opt_interp_Sparam_mag, opt_interp_Sparam_phase,OP, param)
   3  2  %% License Notice
   4  3  %
   5  4  % Copyright 2025 802-COM Authors
 ...  ...  @@ -225,4 +224,25 @@ switch opt_interp_Sparam_phase
 225  224            'debug_interp_Sparam valid values are "old", "zero_DC", "interp_to_DC", "interp_and_shift_to_DC",
"trend_and_shift_to_DC", "interp_cubic_to_dc_linear_to_inf"');
 226  225  end
 227  226  H_i = H_mag_i.*exp(1j*H_ph_i);
+     227  +
+     228  + if OP.ZERO_PAD
+     229  +     % tukey_limit is the beginning or end of the tukey window (depending on sign of
param.zero_pad_tukey_window_in_fb)
+     230  +     tukey_limit = fin(end)+param.zero_pad_tukey_window_in_fb*param.fb;
+     231  +     % zp_freq is where zero padding starts
+     232  +     zp_freq = fin(end);
+     233  +     if param.zero_pad_tukey_window_in_fb == 0
+     234  +         H_tw = 1;
+     235  +     elseif param.zero_pad_tukey_window_in_fb > 0
+     236  +         % Apply Tukey from fin(end) to the tukey limit
+     237  +         H_tw = Tukey_Window(fout,[],  fin(end), tukey_limit   );
+     238  +         % only when tukey_limit is > fin(end) does zp_freq need to be modified
+     239  +         zp_freq = tukey_limit;
+     240  +     else
+     241  +         % Apply Tukey from the tukey limit to fin(end)
+     242  +         H_tw = Tukey_Window(fout,[],  tukey_limit  ,   fin(end) );
+     243  +     end
+     244  +     H_i=H_tw.*H_i;
+     245  +     H_i(fout > zp_freq)= eps(0);
+     246  + end
+     247  +
 228  248  Sout=H_i;
```

Zero padding reduces delta ERL to < 0.1 dB
for *sekel_3dj_02_2503--OSFP1p6T_Mated_File_HCB-WT29563_MCB-WT29449_RX1*





read_Nport_touchstone

- Add rangelimit function to where s-parameters are read in

src/read_p4_s4params.m — View file @ aa184b1e

```
...  ...  @@ -62,6 +62,8 @@ end
62   62
63   63   %AJG:  fast touchstone read for any number of ports
64   64   [sch,schFreqAxis]=read_Nport_touchstone(infile,ports);
     65 + [sch,schFreqAxis,data.limited, param ]=rangelimit(sch,schFreqAxis,param,OP);
     66 + data.flim=param.flim;
65   67   % matrix to introduce p or n skew on Tx or Rx RIM 12/29/2023
66   68   % Sigma's will be form exp(2i*pi*f*skew*1e-12). i.e. if skew = 0 sigma = 1
67   69   % need to swap sigma for 1 and 3 and 2 and 4 not RIM 12/29/2023
...  ...
```

rangelimit

- Trims the s parameters base on param.flim
- The keyword flim is specified in hertz in the spreadsheet

src/rangelimit.m 0 → 100644 — View file @ aa184b1e

```
+ function [schout,schFreqAxisout,limited,param]=rangelimit(sch,schFreqAxis,param,OP)
+ %% License Notice
+ %
+ % Copyright 2025 802-COM Authors
+ %
+ % Redistribution and use in source and binary forms, with or without
+ % modification, are permitted provided that the following conditions are
+ % met:
+ %
+ % - Redistributions of source code must retain the above copyright
+ %   notice, this list of conditions and the following disclaimer.
+ %
+ % - Redistributions in binary form must reproduce the above copyright
+ %   notice, this list of conditions and the following disclaimer in the
+ %   documentation and/or other materials provided with the distribution.
+ %
+ % - Neither the name of the copyright holder nor the names of its
+ %   contributors may be used to endorse or promote products derived from
+ %   this software without specific prior written permission.
+ %
+ % THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
+ % "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
+ % LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
+ % A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
+ % HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
+ % SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
+ % LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
+ % DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
+ % THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
+ % (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
+ % OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
+ %
+ % SPDX-License-Identifier: BSD-3-Clause
+ iend=find(schFreqAxis>=param.flim,1,'first');
+ if ~isempty(iend)
+     schFreqAxisout=schFreqAxis(1:iend);
+     schout = sch(1:iend,:,:);
+     limited=1;
+ else
+     schFreqAxisout=schFreqAxis;
+     schout=sch;
+     limited=0;
+     param.flim=schFreqAxis(end);
+ end
```

New Keywords

- Zero_pad default 0 (use what is in prior versions of CO)
- Flim defaults 1000e9
- zero_pad_tukey_window_in_fb defaults to 0
- See template configuration file https://opensource.ieee.org/richard.mellitz/com_code/-/tree/Zero_pad_and_flim/config_templates/testing_zero_pad?ref_type=heads
  - **config_06-04-2025_448G_BW_PAM_experiment_with_zero_pad.xlsx**
  - Values and settings in the spread sheet may not be align with a standard or ongoing work. Please check before using.

Operational			
ERL Pass threshold	7.3	dB	
COM Pass threshold	3	db	
DER_0	2.00E-05		
T_r	0.00235	ns	
FORCE_TR	1	logical	
PMD_type	C2C		
EW	0		
MLSD	1	logical	
ts_anchor	1		
sample_adjustment	[-12 0]		
num_ui_RXFF_noise	2048		
N_qb	6.5		
zero_pad	1		
flim	10.0E+12	Hz	
zero_pad_tukey_window_in_fb	0	fb	

read_ParamConfigFile

- Adding new keywords

```
...  ...  @@ -192,7 +192,7 @@ else
192  192      OP.RxFFE=false;
193  193  end
194  194  param.num_ui_RXFF_noise=xls_parameter(parameter, 'num_ui_RXFF_noise', true,2048); % Rx FFE precursor tapn limit
195       -
     195  + param.flim=xls_parameter(parameter, 'flim', true,1000e9); % limit sparameter freq range RIM 05/30/2025
196  196  param.g_DC_HP_values = xls_parameter(parameter, 'g_DC_HP', true,[]); % CTF AC-DC gain list (GDC2)
197  197  param.f_HP = 1e9*xls_parameter(parameter, 'f_HP_PZ', true, []); % CFT pole pole zero pair in GHz for low
     frequency CTF
198  198  param.f_HP_Z = 1e9*xls_parameter(parameter, 'f_HP_Z', true, []); % CFT zero fz1 is in GHz. Normally a list for
     120e. Not normally use elsewise
...  ...  @@ -400,6 +400,7 @@ param.LOCAL_SEARCH=xls_parameter(parameter,'Local Search',true,0); % Decreases C
400  400  % Operational control variables
401  401  %OP.include_pcb = xls_parameter(parameter, 'Include PCB (table 92-13)', false, 0);
402  402  param.Tukey_Window=xls_parameter(parameter,'Tukey_Window',true,0); % required for ERL. Set to 1. Default is 0.
     403  + param.zero_pad_tukey_window_in_fb=xls_parameter(parameter,'zero_pad_tukey_window_in_fb',true,0); % window
     relative between fstop and fb as a percentage of fb
403  404  param.Noise_Crest_Factor= xls_parameter(parameter, 'Noise_Crest_Factor', true, 0); % Normally not used. If set
     this is q factor used for quantized Gaussian PDFs
404  405  param.AC_CM_RMS = xls_parameter(parameter, 'AC_CM_RMS', true, 0); % AC_CM_RMS is the CM BBN AWGN RMS at COM
     source point. Default is 0. Adds common mode noise source to the COM signal path for the through channel
405  406  param.ACCM_MAX_Freq=xls_parameter(parameter, 'ACCM_MAX_Freq', true, param.fb); % F max for integrating ACCM
     voltage in Hz. Default is fb
...  ...  @@ -527,6 +528,7 @@ OP.LIMIT_JITTER_CONTRIB_TO_DFE_SPAN = xls_parameter(parameter, 'LIMIT_JITTER_CON
527  528  OP.impulse_response_truncation_threshold = xls_parameter(parameter, 'Impulse response truncation threshold',
     false, 1e-3); % zero padding threshold in fraction of IR peak for the impulse response. Effectively controls the
     length of time for the PR. Larger values decrease run time and accuracy. Default is 1e-3.
528  529  OP.interp_sparam_mag = xls_parameter(parameter, 'S-parameter magnitude extrapolation policy', false,
     'linear_trend_to_DC'); % magnitued extrapolation method
529  530  OP.interp_sparam_phase = xls_parameter(parameter, 'S-parameter phase extrapolation policy', false,
     'extrap_cubic_to_dc_linear_to_inf'); % phase extrapolation method
     531  + OP.ZERO_PAD= xls_parameter(parameter, 'ZERO_PAD', false, 0); % phase extrapolation method
530  532  OP.PMD_type= xls_parameter(parameter, 'PMD_type', false,'C2C'); %  Either C2C or C2M. C2M is for computing VEC
     and VEO
531  533  OP.PHY= xls_parameter(parameter, 'PHY', false, OP.PMD_type); % The keyword OP.PMD_type is now used
532  534  if strcmpi(OP.PHY,'C2M')
...  ...
```

src/read_p2_s2params.m View file @ aa184b1e

Outputs

- fstop_GHz – the stop frequency used in GHz
  - flim may exceed the frequency range
- Flim_GHz –the requested frequency stop in GHz
- P_signal_sigma – use to aid in the debugging specified in mellitz_COM_02b_2505 (detect a change in bandwidth efficieny for flim and zero_pad.

Pass through call addition of param in functions

Several functions were touched. See following slides

Pass through call addition of param in functions

src/COM_FD_to_TD.m — View file @ aa184b1e

```
@@ -63,13 +63,13 @@ for i=1:param.number_of_s4p_files
     [chdata(i).uneq_imp_response, ...
         chdata(i).t, ...
         chdata(i).causality_correction_dB, ...
-        chdata(i).truncation_dB] = s21_to_impulse_DC(chdata(i).sdd21 ,chdata(i).faxis, param.sample_dt, OP) ;
+        chdata(i).truncation_dB] = s21_to_impulse_DC(chdata(i).sdd21 ,chdata(i).faxis, param.sample_dt, OP,param) ;
     if ~OP.RX_CALIBRATION || i==1 % DC (common to differentail model is not good used for RX_Calibrataion channel
         chdata(i).uneq_imp_response=chdata(i).uneq_imp_response*chdata(i).A; % adjust IRx for amplitude
         [chdata(i).uneq_CD_imp_response, ...
             chdata(i).t_DC, ...
             chdata(i).causality_correction_DC_dB, ...
-            chdata(i).truncation__DC_dB] = s21_to_impulse_DC(chdata(i).sdc21 ,chdata(i).faxis, param.sample_dt, OP) ;
+            chdata(i).truncation__DC_dB] = s21_to_impulse_DC(chdata(i).sdc21 ,chdata(i).faxis, param.sample_dt, OP,param) ;
     end
     % adjust voltage derive here once it's decided what to use
     %------------------------------------------------------------
```

src/FD_Processing.m — View file @ aa184b1e

```
@@ -161,6 +161,7 @@ for i=1:param.number_of_s4p_files
         P_signal=2*chdata(i).delta_f*...
             sum(W(chdata(i).faxis(index_f1:index_f2),chdata(i).ftr,fr,param.fb).*10.^(-Il_dB(index_f1:index_f2)/10)  );
         P_signal_sigma=sqrt(P_signal);
+        output_args.P_signal_sigma=P_signal_sigma;
         %
         %%
         [ILD_magft chdata(i).fit_orig] = get_ILN(chdata(i).sdd21f(index_f1:index_f2), chdata(i).faxis(index_f1:index_f2));
```

src/RILN_TD.m — View file @ aa184b1e

```
@@ -53,12 +53,12 @@ if exist('OP','var')
     [RILN_TD_struct.REF.FIR, ...
         RILN_TD_struct.REF.t, ...
         RILN_TD_struct.REF.causality_correction_dB, ...
-        RILN_TD_struct.REF.truncation_dB] = s21_to_impulse_DC(sdd21.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP) ;
+        RILN_TD_struct.REF.truncation_dB] = s21_to_impulse_DC(sdd21.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP,param) ;
     RILN_TD_struct.REF.PR=filter(ones(1, param.samples_per_ui), 1, RILN_TD_struct.REF.FIR);
     [RILN_TD_struct.FIT.FIR, ...
         RILN_TD_struct.FIT.t, ...
         RILN_TD_struct.FIT.causality_correction_dB, ...
-        RILN_TD_struct.FIT.truncation_dB] = s21_to_impulse_DC(RIL.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP) ;
+        RILN_TD_struct.FIT.truncation_dB] = s21_to_impulse_DC(RIL.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP,param) ;
     RILN_TD_struct.FIT.PR=filter(ones(1, param.samples_per_ui), 1, RILN_TD_struct.FIT.FIR);
     ipeak=find(RILN_TD_struct.REF.PR==max(RILN_TD_struct.REF.PR),1,'first');
     NrangeUI=1000;
```

Pass through call addition of param in functions

src/read_ParamConfigFile.m — View file @ aa184b1e

```
false, 1e-3); % zero padding threshold in fraction of IR peak for the impulse response. Effectively controls
the length of time for the PR. Larger values decrease run time and accuracy. Default is 1e-3.
OP.interp_sparam_mag = xls_parameter(parameter, 'S-parameter magnitude extrapolation policy', false,
'linear_trend_to_DC'); % magnitued extrapolation method
OP.interp_sparam_phase = xls_parameter(parameter, 'S-parameter phase extrapolation policy', false,
'extrap_cubic_to_dc_linear_to_inf'); % phase extrapolation method
+ OP.ZERO_PAD= xls_parameter(parameter, 'ZERO_PAD', false, 0); % phase extrapolation method
OP.PMD_type= xls_parameter(parameter, 'PMD_type', false,'C2C'); %  Either C2C or C2M. C2M is for computing VEC
and VEO
OP.PHY= xls_parameter(parameter, 'PHY', false, OP.PMD_type); % The keyword OP.PMD_type is now used
if strcmpi(OP.PHY,'C2M')
...
```

src/read_p2_s2params.m — View file @ aa184b1e

```
- function [data, SDD, SDC] = read_p2_s2params(infile, plot_ini_s_params, plot_dif_s_params, ports,OP)
+ function [data, SDD, SDC] = read_p2_s2params(infile, plot_ini_s_params, plot_dif_s_params, ports,OP,param)
%% License Notice
%
% Copyright 2025 802-COM Authors
@@ -78,7 +78,8 @@ end

%AJG:  fast touchstone read for any number of ports
[sch,schFreqAxis]=read_Nport_touchstone(infile,ports);
-
+ [sch,schFreqAxis,data.limited, param ]=rangelimit(sch,schFreqAxis,param,OP);
+ data.flim=param.flim;


D=NaN(size(sch));
...
```

src/read_s4p_files.m — View file @ aa184b1e

```
@@ -49,7 +49,7 @@ for i=1:num_files
    if (~isfield(chdata(i), 'faxis')) || isempty(chdata(i).faxis)
        switch lower(chdata(i).ext)
            case '.s2p' % for differential return loss
-               [Sch,SDDch] = read_p2_s2params(chdata(i).filename,  0, 0, param.snpPortsOrder, OP);
+               [Sch,SDDch] = read_p2_s2params(chdata(i).filename,  0, 0, param.snpPortsOrder, OP, param);
                chdata(i).fmaxi = length(Sch.freq);
                chdata(i).faxis = Sch.freq;
                chdata(i).sdd11_raw = transpose(SDDch(1:chdata(i).fmaxi,1,1));
...
```

Pass through call addition of param in functions

src/get_RILN_cmp_td.m — View file @ aa184b1e

```
@@ -112,14 +112,14 @@ if exist('OP','var')
     [RILN_TD_struct.REF.FIR, ...
         RILN_TD_struct.REF.t, ...
         RILN_TD_struct.REF.causality_correction_dB, ...
-        RILN_TD_struct.REF.truncation_dB] = s21_to_impulse_DC(sdd21.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP) ;
+        RILN_TD_struct.REF.truncation_dB] = s21_to_impulse_DC(sdd21.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP,param) ;
     RILN_TD_struct.REF.PR=filter(ones(1, param.samples_per_ui), 1, RILN_TD_struct.REF.FIR);


     [RILN_TD_struct.FIT.FIR, ...
         RILN_TD_struct.FIT.t, ...
         RILN_TD_struct.FIT.causality_correction_dB, ...
-        RILN_TD_struct.FIT.truncation_dB] = s21_to_impulse_DC(FIT.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP) ;
+        RILN_TD_struct.FIT.truncation_dB] = s21_to_impulse_DC(FIT.*H_bw.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP,param) ;
     RILN_TD_struct.FIT.PR=filter(ones(1, param.samples_per_ui), 1, RILN_TD_struct.FIT.FIR);


@@ -131,7 +131,7 @@ if exist('OP','var')
     [RILN_TD_struct.RIL.FIR, ...
         RILN_TD_struct.RIL.t, ...
         RILN_TD_struct.RIL.causality_correction_dB, ...
-        RILN_TD_struct.RIL.truncation_dB] = s21_to_impulse_DC(RIL.*H_bw.*H_t.*H_tw ,RIL_f, param.sample_dt, OP) ;
+        RILN_TD_struct.RIL.truncation_dB] = s21_to_impulse_DC(RIL.*H_bw.*H_t.*H_tw ,RIL_f, param.sample_dt, OP,param) ;
     RILN_TD_struct.RIL.PR=filter(ones(1, param.samples_per_ui), 1, RILN_TD_struct.RIL.FIR);


@@ -154,7 +154,7 @@ if exist('OP','var')
     [RILN_TD_struct.REF_noise.FIR, ...
         RILN_TD_struct.REF_noise.t, ...
         RILN_TD_struct.REF_noise.causality_correction_dB, ...
-        RILN_TD_struct.REF_noise.truncation_dB] = s21_to_impulse_DC(port2_reflection_rereflection_noise.*H_bw.*H_t.*H_tw ,f_reflection_rereflection_noise, param.sample_dt, OP) ;
+        RILN_TD_struct.REF_noise.truncation_dB] = s21_to_impulse_DC(port2_reflection_rereflection_noise.*H_bw.*H_t.*H_tw ,f_reflection_rereflection_noise, param.sample_dt, OP,param) ;
     RILN_TD_struct.REF_noise.PR=filter(ones(1, param.samples_per_ui), 1, RILN_TD_struct.REF_noise.FIR);

     ipeak=find(RILN_TD_struct.REF.PR==max(RILN_TD_struct.REF.PR),1,'first');
```

src/get_TDR.m — View file @ aa184b1e

```
@@ -176,7 +176,7 @@ TDR_results.tx_filter=H_t;


 [IR, t, causality_correction_dB, truncation_dB] = ...
-    s21_to_impulse_DC(RLf,  sdd.Frequencies(:), param.sample_dt,OP);
+    s21_to_impulse_DC(RLf,  sdd.Frequencies(:), param.sample_dt,OP,param);


 %
```

Pass through call addition of param in functions

src/calculate_delay_CausalityEnforcement.m — View file @ aa184b1e

```
111 111 IL=interp_Sparam(ILin,freq_array,fout, ...
112     -     OP.interp_sparam_mag, OP.interp_sparam_phase,OP);
    112 +     OP.interp_sparam_mag, OP.interp_sparam_phase,OP,param);
113 113   IL_nan = find(isnan(IL));
114 114   for in=IL_nan
115 115       IL(in)=IL(in-1);
... ... @@ -122,7 +122,7 @@ clear IL IL_nan IL_symmetric
122 122
123 123   ILin=sdd21_causality_enforced;
124 124   IL=interp_Sparam(ILin,freq_array,fout, ...
125     -     OP.interp_sparam_mag, OP.interp_sparam_phase,OP);
    125 +     OP.interp_sparam_mag, OP.interp_sparam_phase,OP,param);
126 126   IL_nan = find(isnan(IL));
127 127   for in=IL_nan
128 128       IL(in)=IL(in-1);
```

src/com_ieee8023_.m — View file @ aa184b1e

```
... ... @@ -344,6 +344,8 @@ while (OP.RX_CALIBRATION==1 || DO_ONCE==true)
344 344           else
345 345               %fill in chada with s-parameters
346 346               [chdata, SDDch, SDDp2p ] = read_s4p_files(param, OP, chdata);
    347 +             output_args.fstop_GHz=min(param.flim,chdata(1).faxis(end))/1e9;
    348 +             output_args.flim_GHz=param.flim/1e9;
347 349               [chdata, param] = process_sxp(param, OP, chdata, SDDch);
348 350           end
349 351           if OP.BREAD_CRUMBS
```

src/get_ILN_cmp_td.m — View file @ aa184b1e

```
... ... @@ -79,13 +79,13 @@ if exist('OP','var')
79 79      [TD_ILN.REF.FIR, ...
80 80           TD_ILN.REF.t, ...
81 81           TD_ILN.REF.causality_correction_dB, ...
82     -         TD_ILN.REF.truncation_dB] = s21_to_impulse_DC(sdd21.*H_bt.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP) ;
   82 +         TD_ILN.REF.truncation_dB] = s21_to_impulse_DC(sdd21.*H_bt.*H_t.*H_tw ,faxis_f2, param.sample_dt,
   OP,param) ;
83 83       TD_ILN.REF.PR=filter(ones(1, param.samples_per_ui), 1, TD_ILN.REF.FIR);
84 84
85 85       [TD_ILN.FIT.FIR, ...
86 86           TD_ILN.FIT.t, ...
87 87           TD_ILN.FIT.causality_correction_dB, ...
88     -         TD_ILN.FIT.truncation_dB] = s21_to_impulse_DC(FIT.*H_bt.*H_t.*H_tw ,faxis_f2, param.sample_dt, OP) ;
   88 +         TD_ILN.FIT.truncation_dB] = s21_to_impulse_DC(FIT.*H_bt.*H_t.*H_tw ,faxis_f2, param.sample_dt,
   OP,param) ;
89 89       TD_ILN.FIT.PR=filter(ones(1, param.samples_per_ui), 1, TD_ILN.FIT.FIR);
90 90       ipeak=find(TD_ILN.REF.PR==max(TD_ILN.REF.PR),1,'first');
91 91       %     NrangeUI=1000;
```

src/get_RAW_FIR.m — View file @ aa184b1e

```
... ... @@ -35,6 +35,6 @@ H_r = 1./polyval([1 2.613126 3.414214 2.613126 1], 1i*f./(0.75*param.fb));
35 35   if ~iscolumn(H), H=H.';end
36 36   if ~iscolumn(H_r), H_r=H_r.';end
37 37   H=H(:).*H_r;
38     - [FIR, t, ~,~] = s21_to_impulse_DC(H ,f, param.sample_dt, OP) ;
   38 + [FIR, t, ~,~] = s21_to_impulse_DC(H ,f, param.sample_dt, OP,param) ;
39 39   % SBR=filter(ones(1, param.samples_per_ui), 1, FIR);
40 40
```

Thank You!