

SNR MDNEXT commit request 4p8p_8

Richard Mellitz, Samtec

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

Commit request: New computation, SNR_MDNEXT

- ❑ Purpose: Evaluate proposal for SNR_MDNEXT
- ❑ Reference: mellitz_3dj_03_2503
 - Fork Namespace: Richard Mellitz / com_code
 - Branch: SNR_MDFEXT
- ❑ Computed only for s-parameters (s4p) passed into COM code between frequencies f_1 and f_2 .
 - Like MDFEXT_ICN
- ❑ Does not require A_v or A_{ft}
- ❑ Does require
 - f_t , f_{ft} (assumed to be the same)
 - f_1 and f_2

FD_Processing.m

Changes

- ❑ SNR_{MDFEXT} = *normalized signal dB power – normalized MDFEXT dB power*

- ❑ $SNR_{MDFEXT} = 10 \log_{10} \left(\frac{\hat{P}_{signal}}{\hat{\sigma}_{mdfext}^2 \sigma_x^2} \right)$

- $\hat{P}_{signal} = 2\Delta f \sum w(f) 10^{-IL(f)/10}$
- $\hat{\sigma}_{mdfext}^2 = 2f \sum w(f) 10^{-MDFEXT_{loss}(f)/10}$

$$w(f) = \frac{1}{f_b} \text{sinc} \left(\frac{f}{f_b} \right)^2 \left[\frac{1}{1 + \left(\frac{f}{f_{ft}} \right)^4} \right] \left[\frac{1}{1 + \left(\frac{f}{f_r} \right)^8} \right]$$

$$IL(f) = -20 \log_{10}(|sdd21(f)|)$$

$$\sigma_x^2 = (L^2 - 1) / 3(L - 1)^2$$

- ❑ Code highlights

- P_{signal} is $\hat{P}_{signal}$
- sqr_{mdfext} is $\hat{\sigma}_{mdfext}^2$
- $SNR_MDFEXT = 10 * \log_{10}(P_signal / sqr_{mdfext})$

```

src/FD_Processing.m
View file @ 211caee9

...   @@ -151,6 +152,19 @@ for i=1:param.number_of_s4p_files
151   % from ba spec, this is basically ICN
152   faxis_6Hz = chdata(i).faxis/1e9;
153   if isequal(chdata(i).type, 'THRU')
154
155   +   %% IL aignal power RIM 04/14/2025
156   +   W = @(f,ft,fr,fb) 1/fb* sinc(f/fb).^2.* (1./(1+(f/ft).^4)) .* (1./(1+(f/fr).^8)); % eq 93A-57
157   +   W_v=@(f,dt,fr,Av,fb) (Av^2/fb) *W(f); % modifuied for AV_v
158   +   % compute IL powerl
159   +   IL_dB=-20*log10(squeeze(abs(chdata(i).sdd21f)));
160   +   P_signal=2*chdata(i).delta_f*...
161   +       sum(W(chdata(i).faxis(index_f1:index_f2),chdata(i).ftr,fr,param.fb).*10.^(-IL_dB(index_f1:index_f2)/10) );
162   +   P_signal_sigma=sqrt(P_signal);
163   +
164   +
165   +
166   +   %
167   +   %%
168
169   [ILD_magft chdata(i).fit_orig] = get_ILN(chdata(i).sdd21f(index_f1:index_f2), chdata(i).faxis(index_f1:index_f2));
170   % find fitted loss values by interpolation using full data, no indexing - Adeo 2022-08-28
171   [~, chdata(i).fit_orig] = get_ILN(chdata(i).sdd21f, chdata(i).faxis);
172
173   @@ -294,8 +308,16 @@ for i=1:param.number_of_s4p_files
294   else % NEXT or FEXT
295       if isequal(chdata(i).type, 'FEXT')
296           MDFEXT=sqrt(abs(chdata(i).sdd21f).^2+MDFEXT.^2); % power sum xtk
297           +   MDFEXT_dBloss=-20*log10(MDFEXT);
298           MDFEXT_ICN=sqrt(2*chdata(i).delta_f/param.fb*sum( chdata(i).Aicn^2*PWF(index_f1:index_f2).*abs(MDFEXT(index_f1:index_f2)).^2)); %eq
299           46 corrected for fb
300           output_args.MDFEXT_ICN_92_47_mV=MDFEXT_ICN*1000;
301           +   % rim 04/14/2025
302           PS_MDFEXT= (W(chdata(i).faxis,chdata(i).ftr,fr,param.fb) .*10.^(-MDFEXT_dBloss/10));
303           pst_mdfext=PS_MDFEXT(index_f1:index_f2);
304           sqr_{mdfext}=(2*chdata(i).delta_f*sum(pst_mdfext)*(param.sigma_X)^2);
305           sigma_mdfext=sqrt(sqr_{mdfext});
306           SNR_MDFEXT=10*log10(P_signal/sqr_{mdfext});
307           output_args.SNR_MDFEXT=SNR_MDFEXT;
308
309       elseif isequal(chdata(i).type, 'NEXT')
310           MDNEXT=sqrt(abs(chdata(i).sdd21f).^2+MDNEXT.^2); % power sum xtk
311           MDNEXT_ICN=sqrt(2*chdata(i).delta_f/param.fb*sum( chdata(i).Aicn^2*PWF(index_f1:index_f2).*abs(MDNEXT(index_f1:index_f2)).^2)); %eq
312           47 corrected for fb

```

I/O Changes

☐ Changes to config

- None

☐ Changes to output

- `output_args.SNR_MDFEXT`

☐ Download beta test code

- https://opensource.ieee.org/richard.mellitz/com_code/-/raw/SNR_MDFEXT/release/com_ieee8023_4p9p0_beta_SNR_MDFEXT_01.m?ref_type=heads&inline=false

Thank You!