Method for Restricting Micro-Reflections

Contribution to IEEE 802.3dg

Ragnar Jonsson
Marvell

November 2022

Introduction

Echo cancelation is a powerful method to increase the performance of a communication system, and echo-canceler based communication systems have better performance than systems using other duplexing methods

The echo canceler size will increase with increasing sampling rate and increasing cable length

To mitigate this, we suggest to define limits for micro-reflections

In this presentation we share some of the new ideas that were adopted for 802.3cy automotive 25Gbps Ethernet

Background

This presentation is largely based on earlier presentation for 802.3cy:

- jonsson_3cy_01a_10_14_20

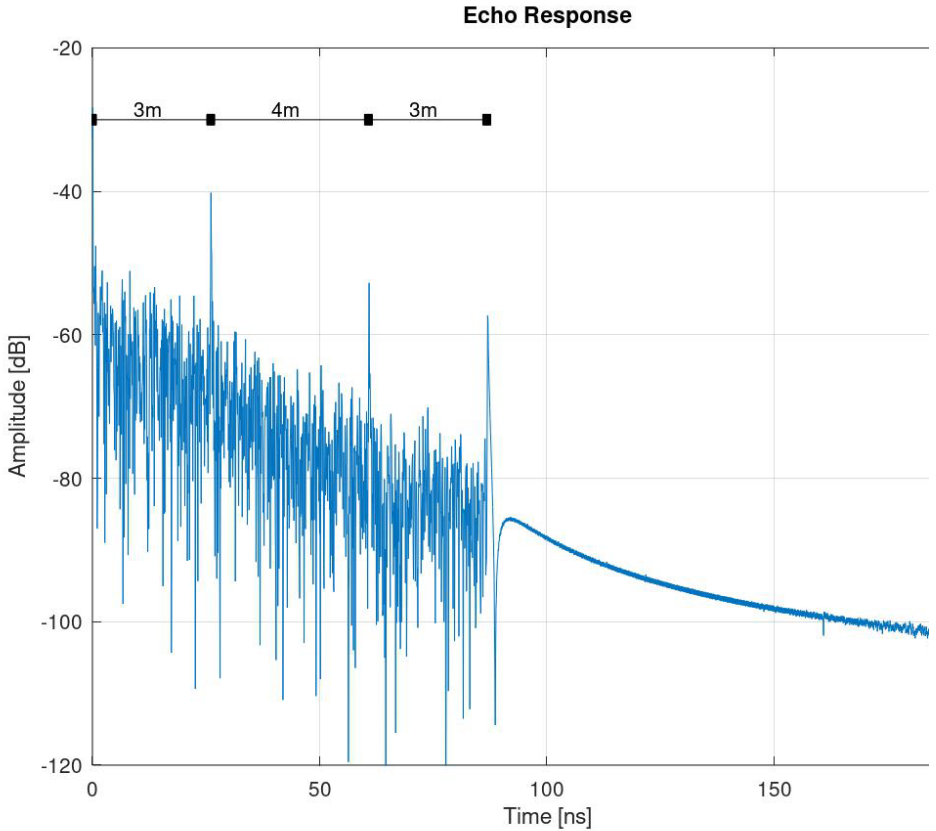
The micro-reflection limits were adopted for 802.3cy in clause 165.7.1.3.2, “Echo tail and residual echo metrics”

There were several presentations on limiting micro-reflections for 802.3cy, including:

- jonsson_3cy_01a_0720
- sedarat_3cy_01_0920
- jonsson_3cy_01a_10_14_20
- sedarat_3cy_02_10_14_20
- sedarat_3cy_02_1120
- jonsson_3cy_01_12_08_20
- jonsson_3cy_01_03_16_21
- jonsson_3cy_01_08_10_21
- jonsson_3cy_02_11_09_21
- jonsson_3cy_01_12_21_21

What are Micro-Reflections?

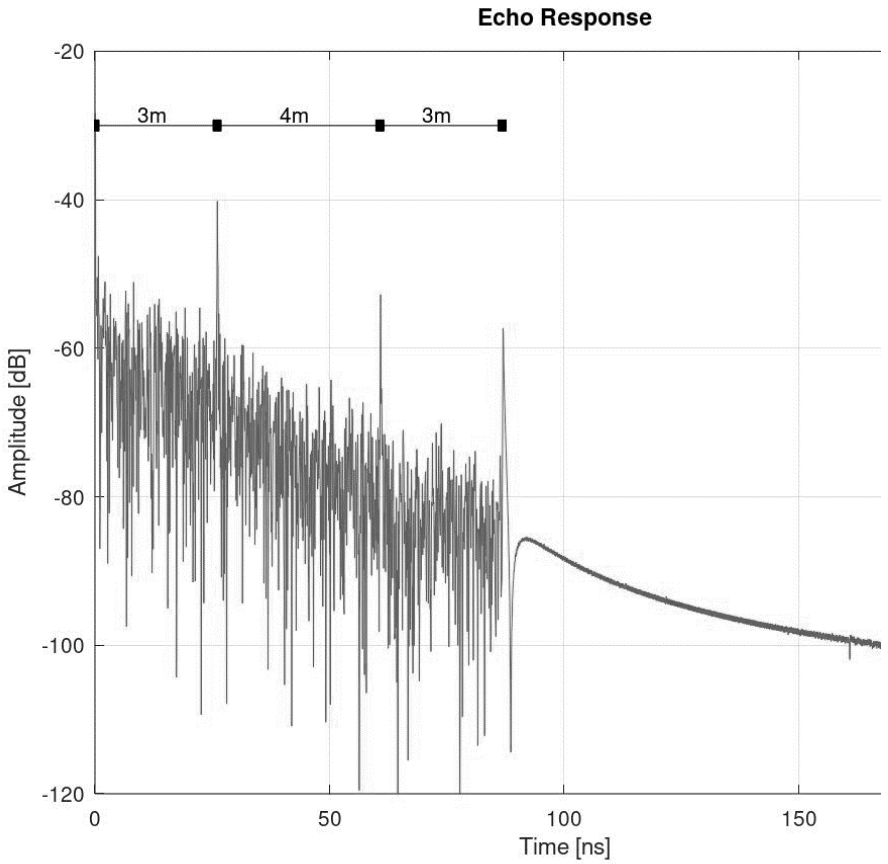
- The term micro-reflection can sometimes cause confusion
- In this presentation we use the term micro-reflections to indicate that we are interested in the time-domain structure of the channel reflections
- We will distinguish between larger reflection at connectors and the much smaller micro-reflections along the cable



Why Restrict Micro-Reflections?

Micro-Reflections can impact

- Achievable bit rate on the link
- Achievable reliability of the link
- Complexity of echo cancelers
- Complexity of equalization
- Complexity of ADC
- Complexity of Analog Front End



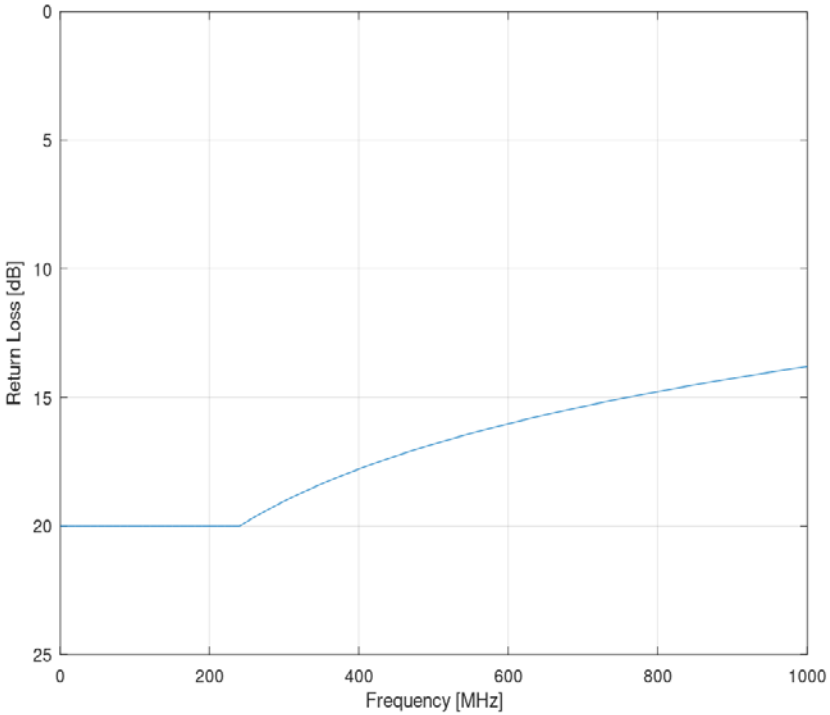
Traditional Limit-Line Cable Specifications

Traditionally echo limitation is defined as frequency mask that limits the return loss

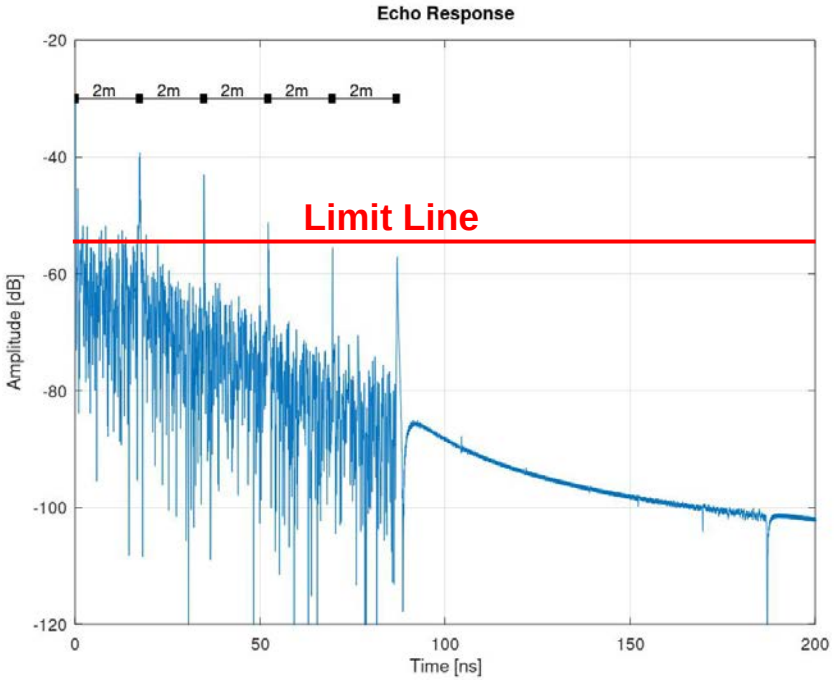
The drawback with this method is that it does not restrict the phase or time domain structure of the channel reflections

With higher data rates we need more optimized transceiver implementations

This requires constraints on the time domain structure of the channel reflections



Time-Domain Approach to Limit Echo

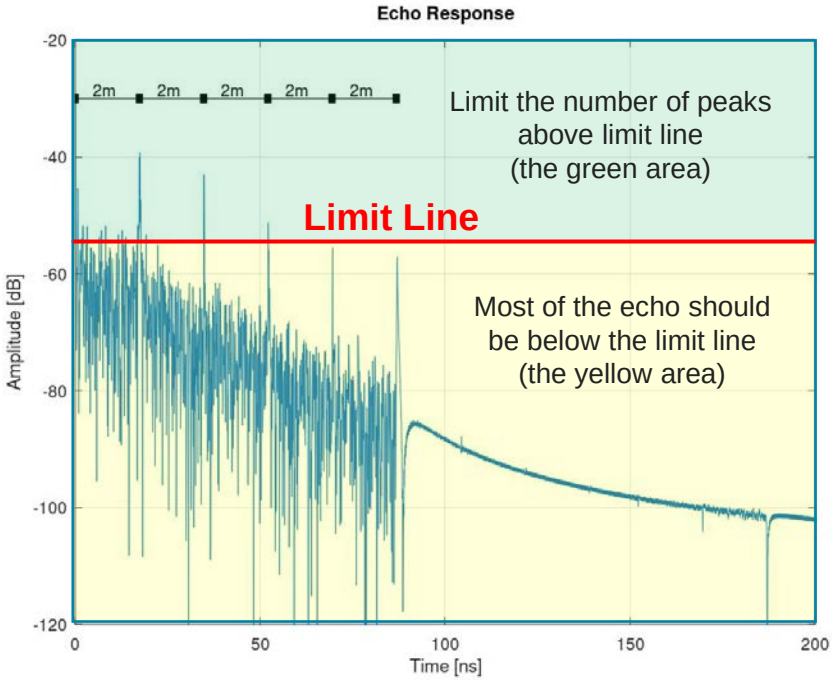


The time-domain structure of the echo signal is more important than the frequency domain properties

The PHY design can take advantage of the echo time-domain structure to implement more efficient echo cancelers

We suggest using a simple limit line that **most** of the echo must stay below

Time-Domain Approach to Limit Echo

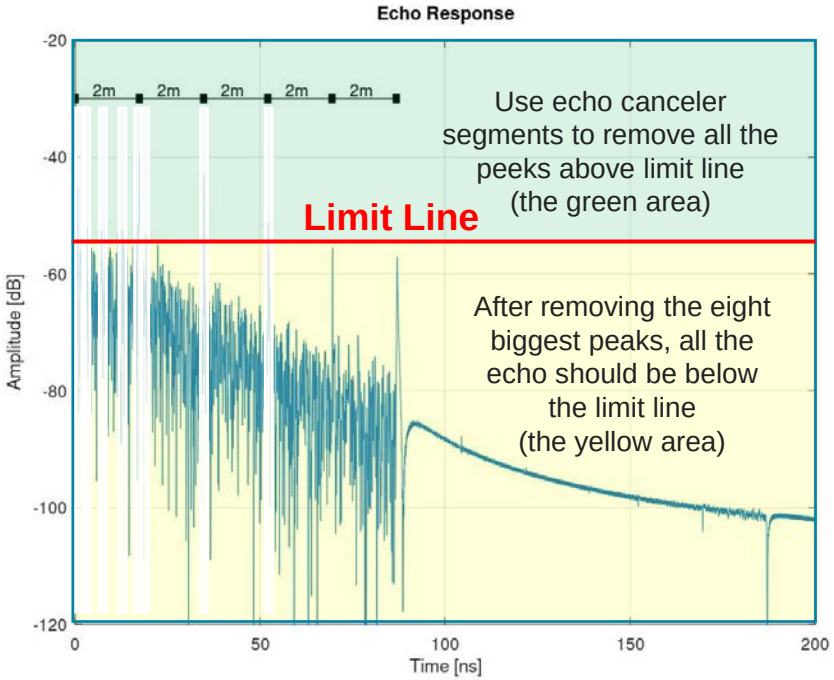


Most of the signal should be low the limit line

Some of the echo peaks can go above the limit line

The number of peeks going above the limit line is restricted both in number and duration

Time-Domain Approach to Limit Echo

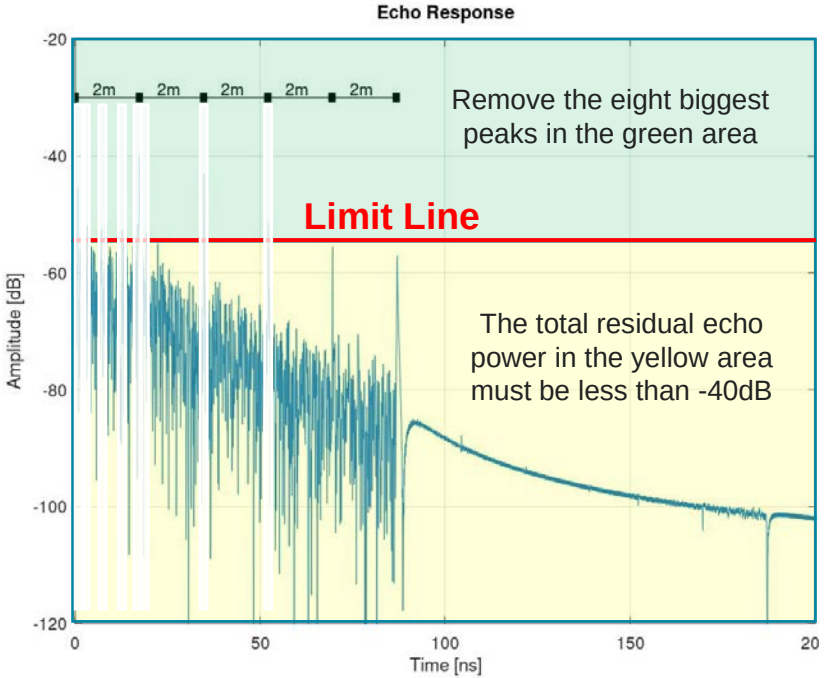


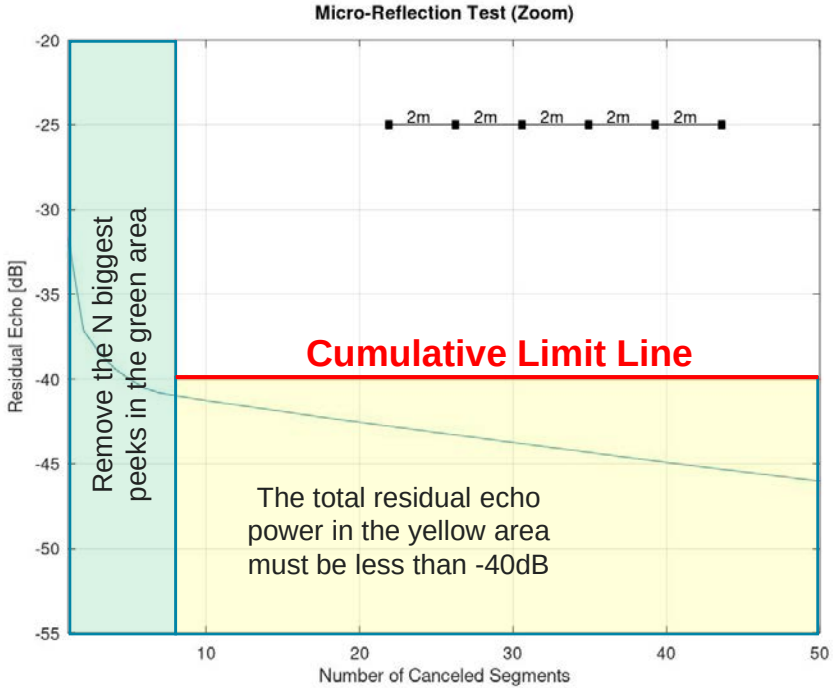
The PHY design can concentrate on canceling the peeks above the limit line

This can be done by deploying several movable echo canceler segments to the echo peaks

This leads to significant savings in the echo canceler implementation

Limit Echo Power After Removing Biggest Peaks

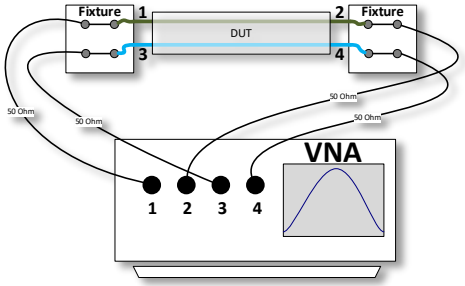




We can relax the requirements by using limit on the cumulative echo power after removing the biggest peaks.

Measuring Micro-Reflections

- One way to evaluate the micro-reflections is to use normal Vector Network Analyzer to measure S-parameters for the channel under test (DUT)
- The differential S11 and S22 parameters (magnitude and phase) can be converted to time-domain echo signals
- The time domain echo signals can be analyzed to evaluate micro reflection characteristics
- See code at: https://www.ieee802.org/3/cy/public/sep21/jonsson_3cy_01_09_28_21.m




```
function h = RJf2t(H,f,T,N)

%RJf2t - Impulse (time) response for a given frequency response.
%Usage:
%   h = RJf2t(H,f,T,N)
% where <H> is the frequency response given at frequencies <f>,
% <T> is the sampling interval, and <N> is the number of output
% samples (must be even).


%%% find size %%%
NN = prod(size(H));

%%% test arguments %%%
if( nargin < 2 )
  f = [0:NN-1]./(NN-1)*pi;
end;
if( nargin < 3 )
  T = 1;
end;
if( nargin < 4 )
  N = 256;
end;
N2 = ceil(N/2);

%%% find problem spots %%%
ix = find(H == H);
H = H(ix);
f = f(ix);

%%% re-shape arguments %%%
H = H(:);

%%% interpolate frequency response %%%
Hs1 = spline(f*T,H,[0:N2]/N2/2);
ang_N = angle(Hs1(N2+1));
x0 = ang_N/(pi);
Hs1 = Hs1.*exp(-j*2*pi*x0*[0:N2]/N2/2);
Hs = [real(Hs1(1)) Hs1(2:N2) real(Hs1(N2+1)) conj(Hs1(N2:-1:2))];

%%% find impulse response from IDFT %%%
h = real(ifft(Hs));

%%%%%%%%%%%%%%%%%%%% End of RJf2t.m %%%%%%%%%%%%%%%%%%%%
```

This is experimental code that is provided "as is".

Conclusion

Limits on micro-reflections were introduced in 802.3cy

We have provided high level summary of micro-reflection limits

We suggest that similar micro-reflection limits should be considered for 802.3dg

Essential technology, done right™