Proposal for a Common Methodology to Compute Error Vector Magnitude (EVM)

Greg D. Le Cheminant, Bernd Nebendahl, Joerg Pfeifle

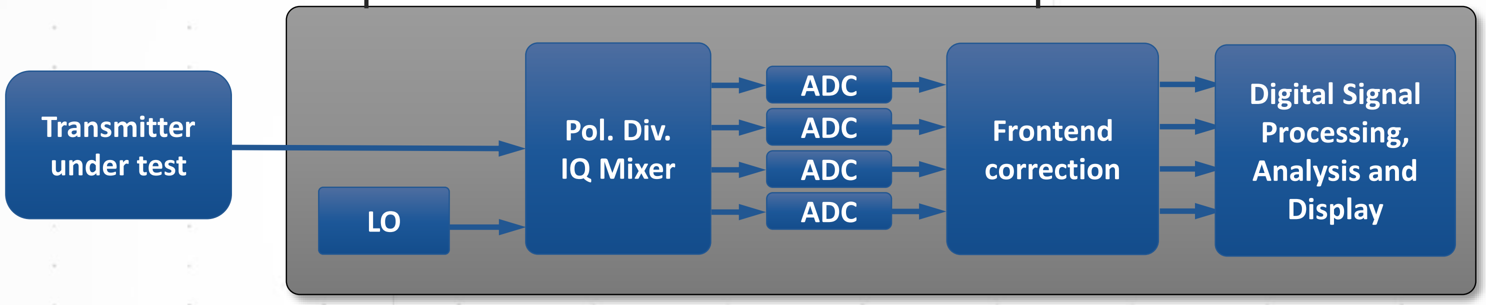
MAY X 2019

KEYSIGHT TECHNOLOGIES

EVM is extracted from an IQ detector and ADC's forming a reference receiver

COMPLEX HARDWARE AND SOFTWARE WORKING TOGETHER

Proposed reference receiver concept



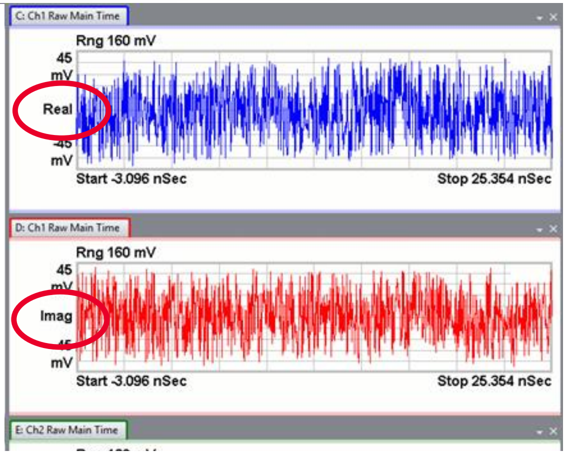
- The polarization diverse receiver detects optical magnitude and phase as electrical I and Q vectors from both polarization states (4 outputs total)
- The ADC function is commonly implemented using a real-time oscilloscope
- The ADC outputs require significant processing to yield a meaningful EVM
- If the processing steps are not well defined, EVM results can be inconsistent

Digital Signal Processing of Reference Receiver

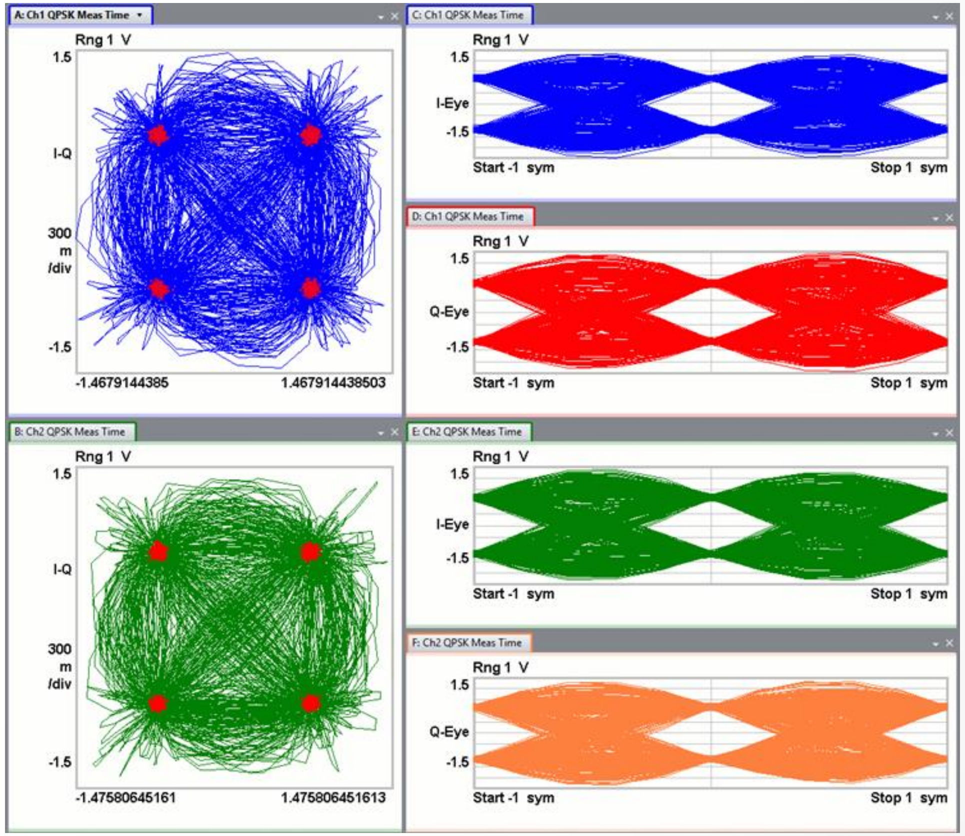
FROM DIGITIZED SIGNAL TO EVM

What the Digitizer / Oscilloscope receives:

Waveforms for both I and Q



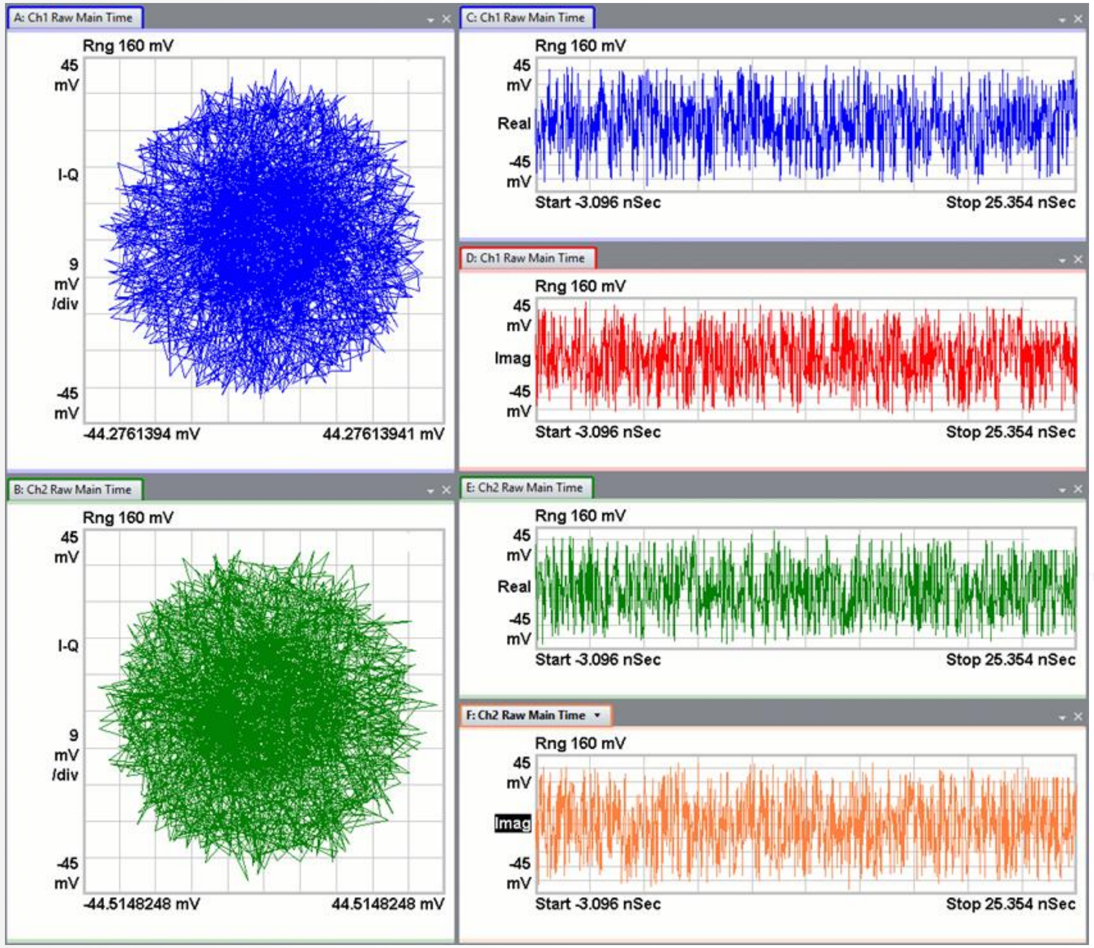
What we expect to see
and what is needed to evaluate EVM



Digital Signal Processing of Reference Receiver

STARTING POINT

What the Digitizer / Oscilloscope receives



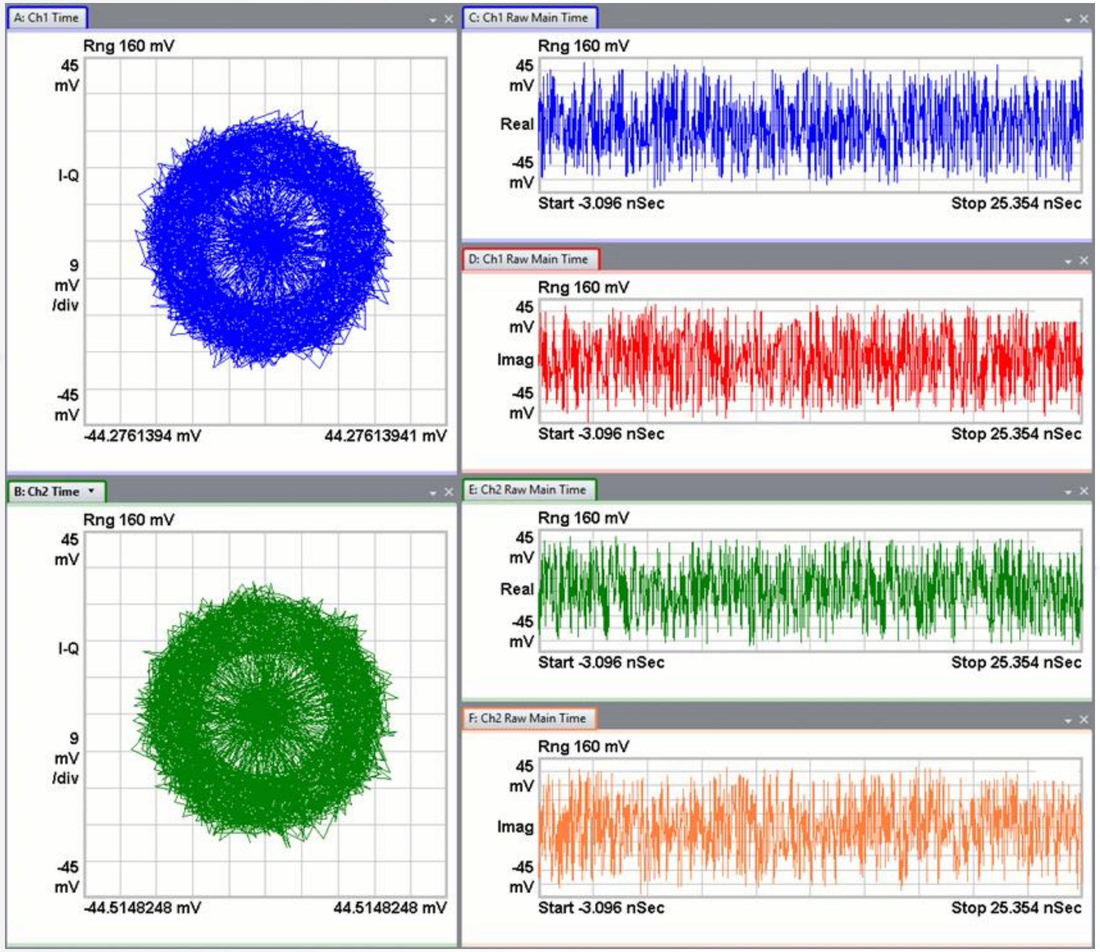
Including front-end corrections of channel imbalances, IQ phase angle errors, timing skew and differential imbalance

- Optical front-end
- Digitizer / Oscilloscope
- Polarization de-multiplexing
- IF offset estimation
- IF phase estimation
- Clock frequency and phase recovery
- EQ (trained with digitally noise loaded samples)
- EVM_{RMS} evaluation

Digital Signal Processing of Reference Receiver

STEP 1: POLARIZATION DE-MULTIPLEXING

After polarization de-multiplexing



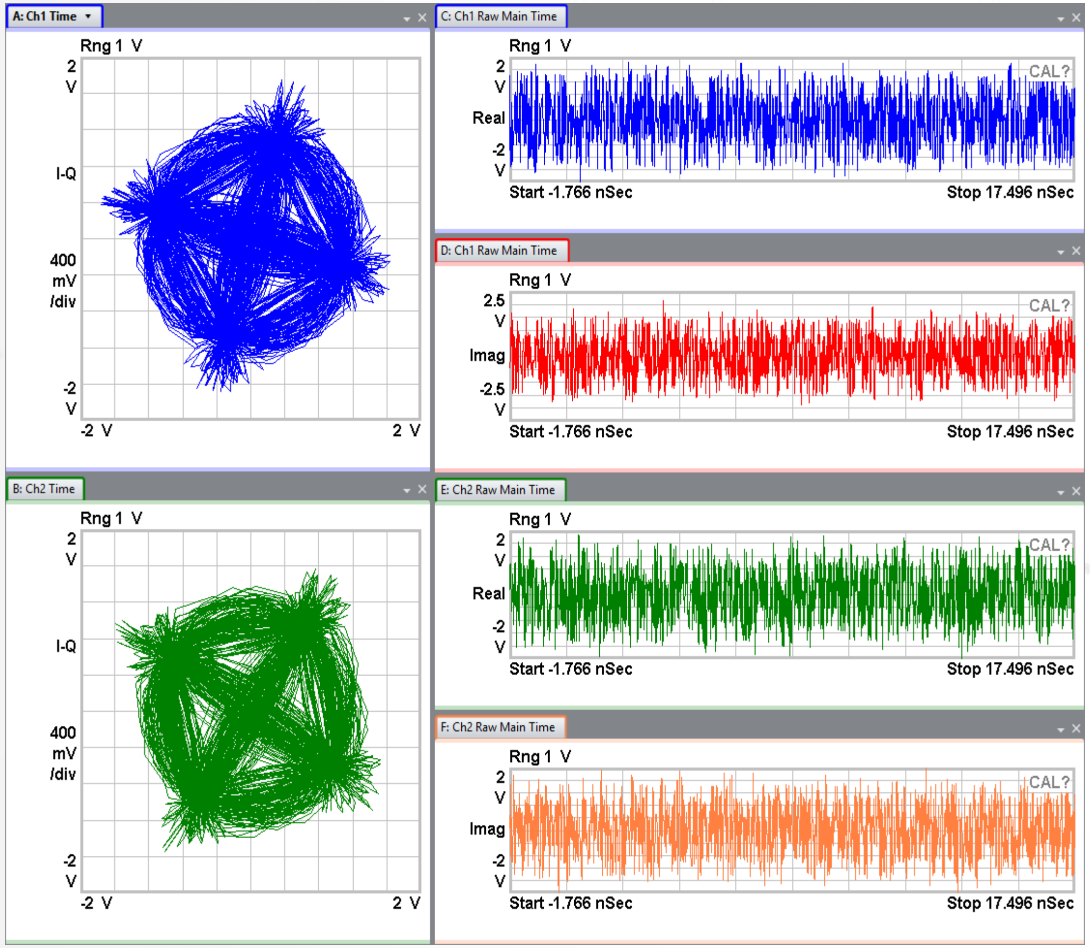
This step should neither improve nor impair the signal quality.

- Optical front-end
- Digitizer / Oscilloscope
- Polarization de-multiplexing
- IF offset estimation
- IF phase estimation
- Clock frequency and phase recovery
- EQ (trained with digitally noise loaded samples)
- EVM_{RMS} evaluation

Digital Signal Processing of Reference Receiver

STEP 2: FREQUENCY OFFSET ESTIMATION

After carrier frequency offset estimation



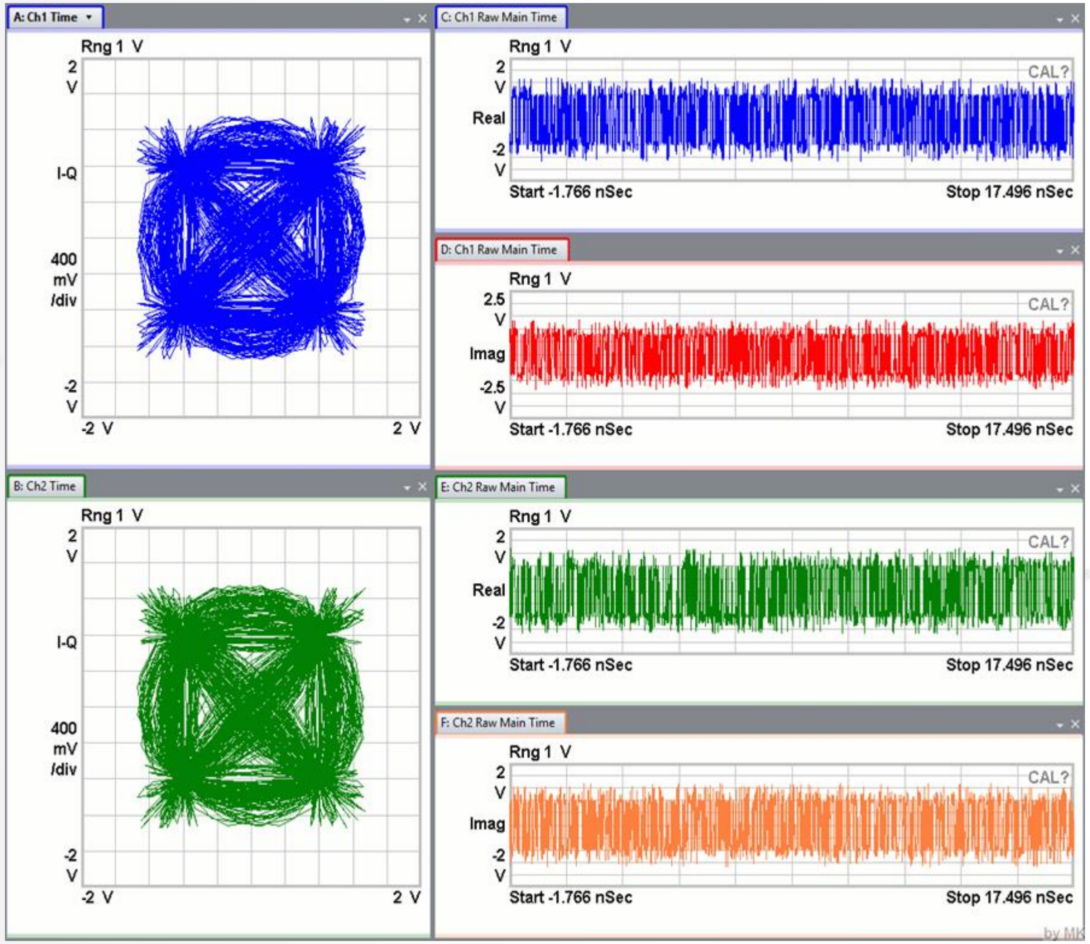
Assumes constant frequency offset (linear phase over time) for given block length.

- Optical front-end
- Digitizer / Oscilloscope
- Polarization de-multiplexing
- IF offset estimation
- IF phase estimation
- Clock frequency and phase recovery
- EQ (trained with digitally noise loaded samples)
- EVM_{RMS} evaluation

Digital Signal Processing of Reference Receiver

STEP 3: CARRIER PHASE ESTIMATION

After carrier phase estimation



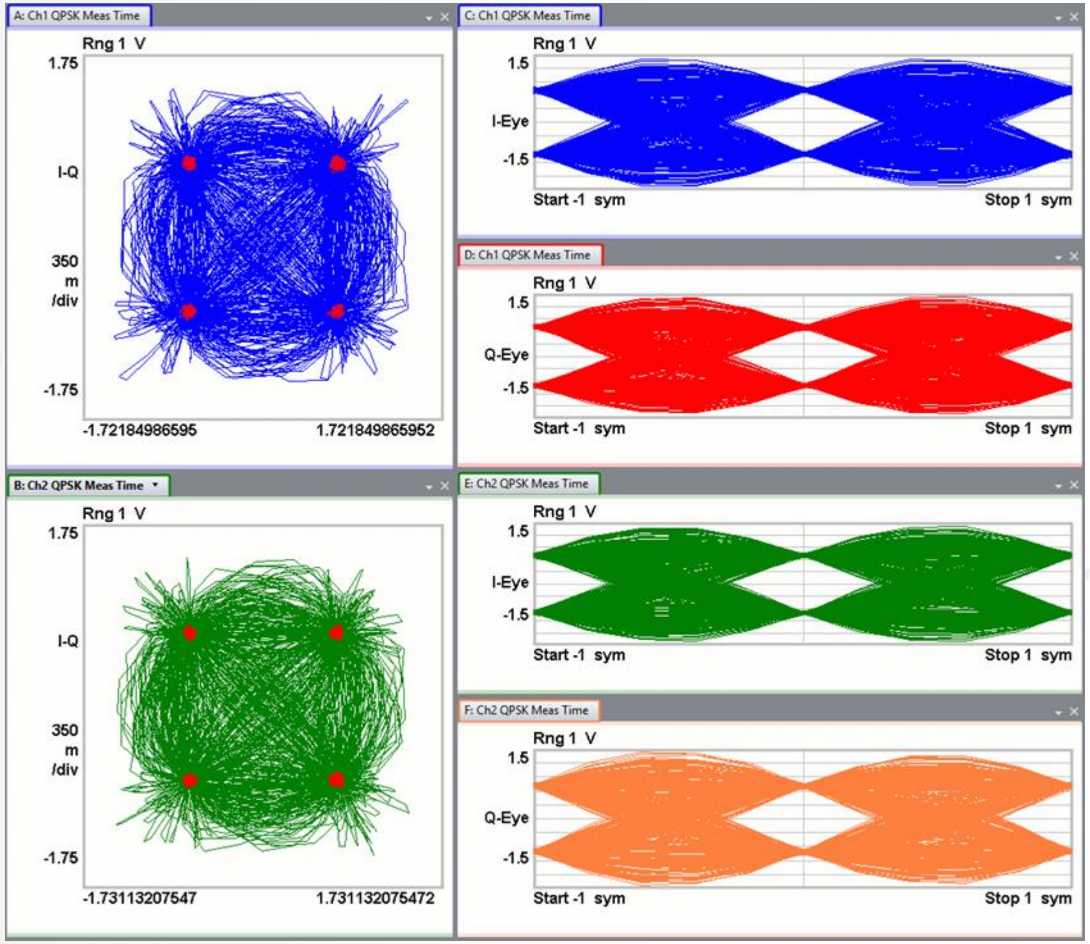
- Optical front-end
- Digitizer / Oscilloscope
- Polarization de-multiplexing
- IF offset estimation
- IF phase estimation
- Clock frequency and phase recovery
- EQ (trained with digitally noise loaded samples)
- EVM_{RMS} evaluation

KEYSIGHT TECHNOLOGIES

Digital Signal Processing of Reference Receiver

STEP 4: CLOCK FREQUENCY AND PHASE RECOVERY

After resampling and re-timing



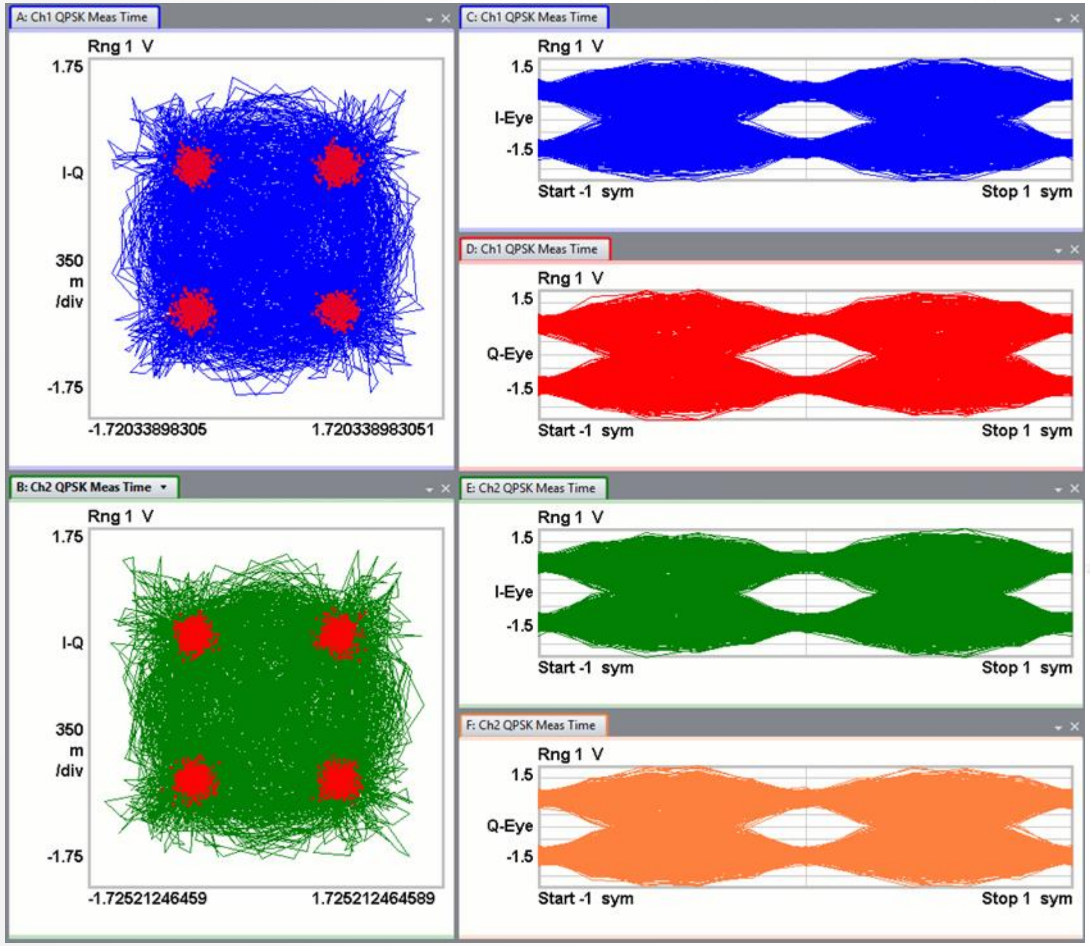
Includes normalization of the measured signal to the reference constellation as well as estimation and removal of IQ offset

- Optical front-end
- Digitizer / Oscilloscope
- Polarization de-multiplexing
- IF offset estimation
- IF phase estimation
- **Clock frequency and phase recovery**
- EQ (trained with digitally noise loaded samples)
- EVM_{RMS} evaluation

Digital Signal Processing of Reference Receiver

STEP 5: NOISE LOADING AND EQ TRAINING

After noise loading and equalization



Numerical result for EVM_{RMS} is reported

- Optical front-end
- Digitizer / Oscilloscope
- Polarization de-multiplexing
- IF offset estimation
- IF phase estimation
- Clock frequency and phase recovery
- EQ (trained with digitally noise loaded samples)
- EVM_{RMS} evaluation

Use well-defined signal processing methods for consistent computation of EVM

- A common method has been used in both ITU and OIF to achieve the previously described processes for computing EVM from waveforms generated from the ADC blocks
  - Used in ITU-T G.698.2 (Q6/SG15)
  - Used in OIF 400ZR
- Propose that the reference receiver concept and signal processing method be used in 802.3 ct and adapted as needed

A common script used to compute EVM from the ADC outputs has been used in both ITU and OIF and is available

Editor - C:\Users\bnebenda\Documents\Standard\ITU\ITU-T SG15\2013-2016\my stuff\script\QPSK_16QAM_20190425\Receiver.m

Receiver.m

```matlab
demuxPolarization = 1;   % do you want MIMO processing done?
                         % get back to Keysight to obtain the required
                         % Matlab files to do so!
demuxBlockCount = 10;    % number of retiming blocks for demultiplexing

retimeBlockCount = 10;   % number of EVM blocks used for retiming

blockSize = 1e5;         % blocksize for impairment and EVM measurement and removal --> 1000

numTaps = 7;             % number of equalizer taps (must be an odd number) --> 7

OSNR = 200;               % OSNR(193.6) at reference point R_s (table in Clause 8)
                         % used to calculate signal-to-noise-ratio used to calculate the amount of
                         % additional white gaussian noise added to
                         % the signal prior to finding the equalizer taps
                         % the EVM is then calculated with the original
                         % signal after noise adding

removeIQoffset = 1;      % optional IQ offset removal
unloadedEVM = 0;         % train equalizer with noise but calculate EVM without noise loading

createPlots = 1;
% output = 1;
savePlots = 1;
output = fopen('\\tcbbnas1\EUC_Gshare\BP-COE-public\R&D\Photonics\OMA\GMAX-OMA\GMAX-OMA R&D\GMAX SystemVue Modell\ITU\QAM16_OSNR200dB_none.txt', 'w');  % output to file

% enforce an odd number of taps
numTaps = floor(numTaps / 2) * 2 + 1;

% clear averages for polarization x
N_avg_x = 0;
P_sig_avg_x = 0;
P_xs_avg_x = 0;
f_offs_avg_x = 0;
EVMrms_avg_x = 0;
FIRtaps_avg_x = zeros(1, numTaps);
SNR_meas_avg_x = 0;

% clear averages for polarization y
N_avg_y = 0;
P_sig_avg_y = 0;
P_xs_avg_y = 0;
EVMrms_avg_y = 0;
f_offs_avg_y = 0;
FIRtaps_avg_y = zeros(1, numTaps);
SNR_meas_avg_y = 0;

delayXY_avg = 0;

% loop through all files
for j=size(filenames):-1:1

    % determine the file format from the extension
    fullname = strcat(filenames(j).folder, '\', filenames(j).name);
    splitname = strsplit(filenames(j).name, '.');

```

EVM code

- Consider the development of TDECQ for PAM4: Steady evolution and improvement was achieved through frequent use of beta TDECQ and results shared from many contributors.
  - Unlike TDECQ we have a big head start. The code has already been used and improved in ITU and OIF, so we are working with algorithms that are already working well for those standards. (Operates on both QPSK and 16QAM formats)
  - Important to relate EVM to system level performance (OSNR) and set EVM specifications. EVM specs cannot be set without a clear definition of the measurement process
- The code does not include polarization demultiplexing. Not complicated and easy to ‘plug’ into the current code
  - Available from participating T&M vendors or develop your own

EVM mathematics

AVAILABLE BUT NOT PART OF THIS PRESENTATION

Maximum error vector magnitude

The Error vector magnitude is measured using a reference receiver as defined in Section 14. EVM_{rms} uses the peak ref. vector for normalization.

Maximum I-Q offset

The I-Q offset of a modulated signal relates to the average signal amplitudes in the I- and Q-phases of that signal. The relative excess (unmodulated) power, P_{excess}, is a measure of this impairment and is obtained from the parameters I_{mean} and Q_{mean} and P_{Signal}, which are intermediate results during the evaluation of the Error Vector Magnitude:

$$P_{excess} = \frac{I_{mean}^2 + Q_{mean}^2}{P_{Signal}}$$

$$IQ_{offset} = 10\log_{10}(P_{excess})$$

Reference receiver for EVM and I-Q offset

The reference receiver includes the following hardware characteristics and processing steps:

- Hardware characteristics:
  - Dual-polarization coherent receiver. Ideally, the receiver should be calibrated over wavelength for:
    - Frequency response
    - Channel imbalances
    - IQ phase angle error
    - Timing skew
  - Real-time Nyquist sampler with sampling rate equal to or larger than the 400ZR symbol rate
- Processing steps[1]:
  - Polarization demultiplex.
  - Retime and resample to one sample per symbol using a Gaussian-shaped low pass filter anti-aliasing filter with a 3-dB bandwidth of 0.5 times the symbol rate.
  - Clock phase recovery
  - Frequency offset estimation and removal assuming a constant frequency offset over the given block size N
  - Carrier phase recovery
  - IQ-offset evaluation and compensation
  - Noise loading for EQ training and EVM evaluation
    - The amplitude A_{RMS} of the noise for each quadrature is calculated from the following equation:

$$A_{RMS} = \sqrt{\frac{0.814 \cdot R_{symbol}}{10^{\frac{OSNR}{10}} \cdot \Delta f_{ref}}}$$

where OSNR is 23 dB and

$$\Delta f_{ref} = \frac{c}{\lambda^2} \cdot RB$$

where c is the velocity of light in vacuum, λ is the optical wavelength and RB is the resolution bandwidth that is 0.1 nm.

  - Apply a 7-tap T-spaced FIR filter with the tap coefficients optimized for BER
    - The sum of all filter tap coefficients is equal to one, and the largest coefficient can be for any of the 7 taps. The individual filter taps are found by minimizing the EVM_{RMS} value
  - EVM evaluation
    - Find the peak vector normalization scaling factor[2]:

$$\alpha = \sqrt{\frac{\max\limits_{0\le k<K}(I_{ref}(k)^2 + Q_{ref}(k)^2)^{\,2}}{\frac{1}{K}\sum_{k=0}^{K-1}(I_{ref}(k)^2 + Q_{ref}(k)^2)}}$$

    - Normalize the sample pairs I_δ and Q_δ in each of the polarizations using the average power multiplied by the peak vector constellation scaling factor[3]:

$$\alpha_{peak} = \alpha\sqrt{\frac{1}{N}\sum_{n=0}^{N-1}(I_\delta(n)^2 + Q_\delta(n)^2)}$$

    - Find the nearest constellation pair $I_{ref}(n)$ and $Q_{ref}(n)$ for each normalized sample pair I_δ and Q_δ in each of the polarizations.
    - Calculate the error vector magnitude for each normalized sample pair I_δ and Q_δ in each of the polarizations:

$$EVM(n) = \sqrt{(I_\delta(n) - I_{ref}(n))^2 + (Q_\delta(n) - Q_{ref}(n))^2}$$

where n is the symbol number within the block starting at 0

Using all the N samples from the x-polarization calculate $EVM_{RMS,x}$:

$$EVM_{RMS,x} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1}EVM(n)^2}$$

Using all the N samples from the y-polarization and calculate $EVM_{RMS,y}$:

$$EVM_{RMS,y} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1}EVM(n)^2}$$

Then calculate EVM_{RMS} in percent from:

$$EVM_{RMS} = \sqrt{\frac{(EVM_{RMS,x}^2 + EVM_{RMS,y}^2)}{2}} \times 100\%$$

[1] The processing is done block wise with block size $N = 1000$. It is possible to group multiple blocks for some of the processing steps. The processing steps should perform only the tasks mentioned in the description. Processing steps can be consolidated and changed in order but not perform any additional signal processing with the purpose of compensating for signal distortions resulting for example from CD, PMD, skews, crosstalk, etc.

[2] k runs over all points in the constellation

[3] This assumes that all constellation points have equal probability in the sample pairs

Thank you!