

Compute Modulation Error Ratio (MER) for QAM

This post defines the Modulation Error Ratio (MER) for QAM signals, and shows how to compute it. As we'll see, in the absence of impairments other than noise, the MER tracks the signal's Carrier-to-Noise Ratio (over a limited range). A Matlab script at the end of the article computes MER for a simplified QAM-64 system.

Figure 1 is a simplified block diagram of a QAM system. The transmitter includes a source of QAM symbols, a root-Nyquist pulse-shaping filter and a Quadrature modulator. The receiver has a Quadrature downconverter, root-Nyquist filter, and decision block. The wide lines represent complex (I/Q) signals, and the narrow lines represent the real modulated signal. The matched root-Nyquist filters perform the dual tasks of limiting the bandwidth of the transmitted signal while minimizing the interference from other symbols at the sampling instant [1,2]. For our discussion of MER, I have left out the usual key receiver systems of AGC, carrier recovery, clock recovery, and adaptive equalization.

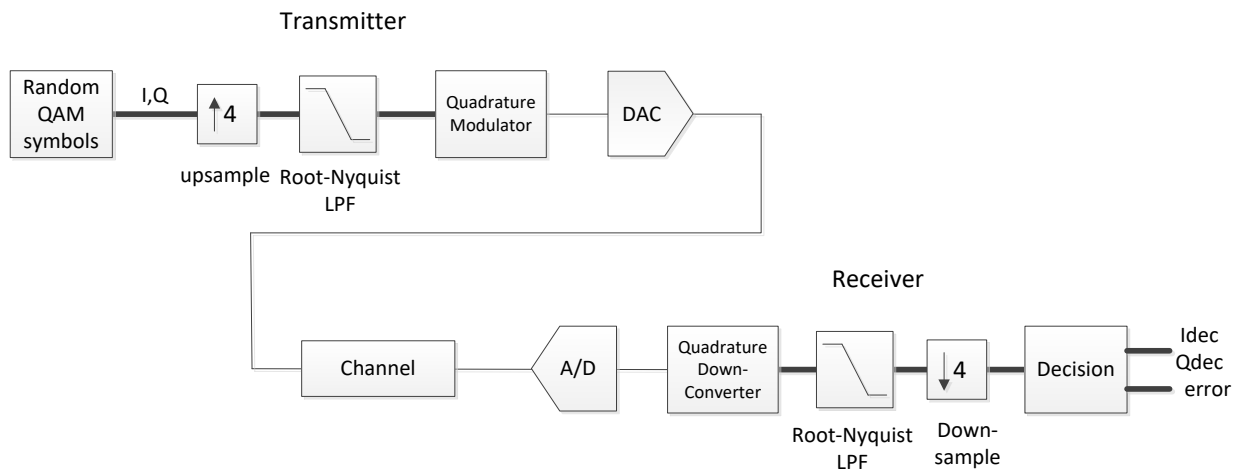


Figure 1. Simplified QAM system block diagram.

For our purposes, we can model the channel at complex baseband [3], so we can eliminate the Quadrature up and down conversions. Also, we will model the entire system as discrete-time, so the D/A and A/D converters can be left out. We then have the block diagram of Figure 2a, where the channel includes Gaussian noise (AWGN) added to I and Q, plus a block labeled "other impairments". These could include linear impairments such as multipath, phase noise, and interference. In this article, our focus is on noise, so we'll leave out the other impairments.

Figure 3 plots some of the labeled signals from Figure 2 for a QAM-16 system with noise. Figure 3a shows the random QAM symbols as a Q_{map} vs I_{map} constellation plot. Figure 3b is an eyeplot of the output of the receiver's root-Nyquist filter, I_{rx} . At this point, the signal has 4 samples per symbol, so the

valid sampling instants are spaced 4 samples apart. For our simplified model, the eye opening aligns exactly with every fourth sample (A real-world receiver must compute the correct sample value using a variable interpolator). The eyes are partially closed due to the noise added to the signal. Downsampling I_{rx} and Q_{rx} by 4 at the correct sample alignment gives I_{down} and Q_{down} , whose constellation plot is shown in Figure 3c. The noise is visible as “fuzz” around the nominal constellation points. The final outputs I_{dec} and Q_{dec} are computed in the decision block by rounding (slicing) I_{down} and Q_{down} to the nearest allowed symbol value, reproducing the original transmitter constellation (Figure 3d).

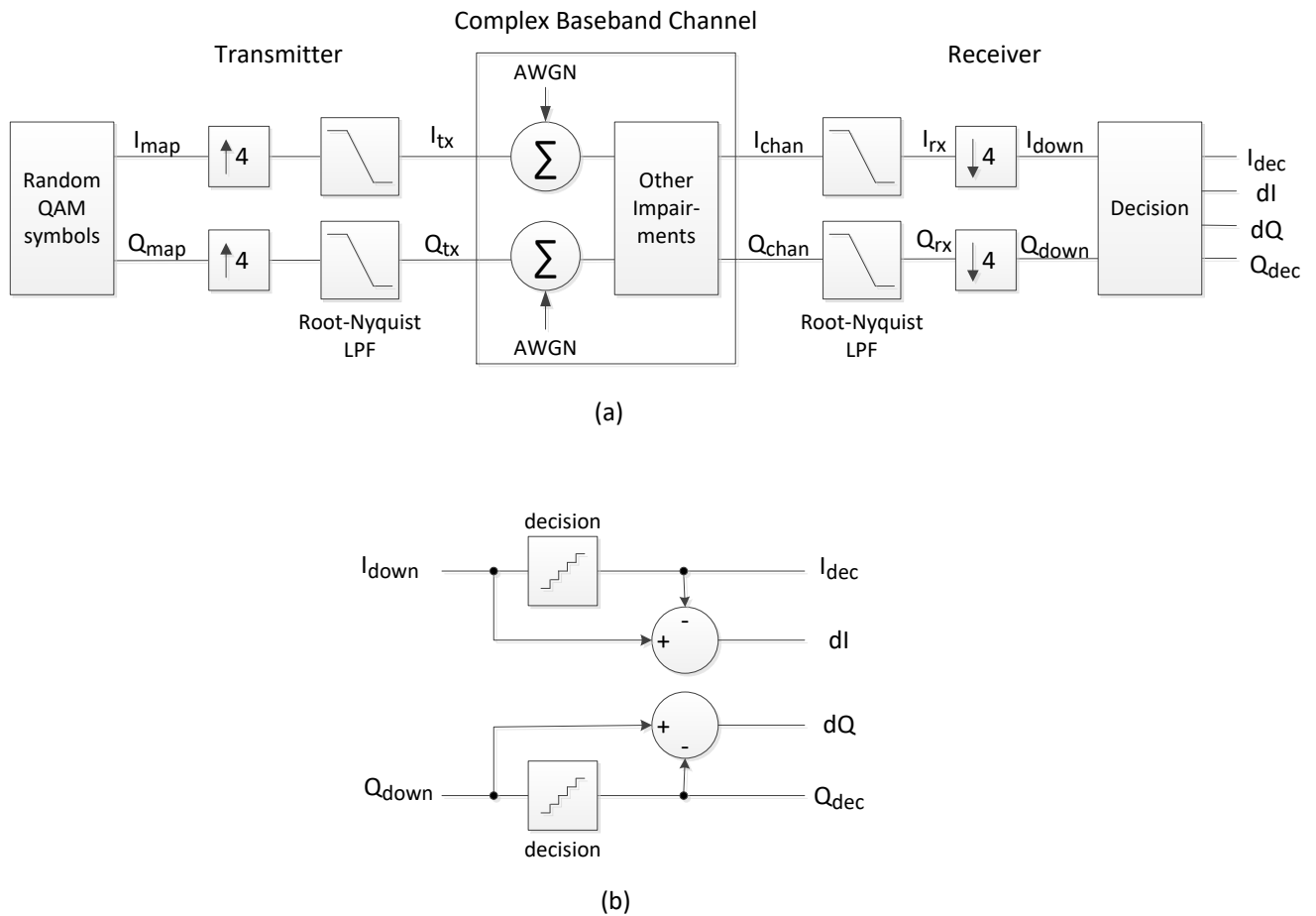


Figure 2. Simplified QAM system at complex baseband.

a. Overall block diagram. AWGN: additive white Gaussian noise.

b. Decision block. $dI = I_{down} - I_{dec}$; $dQ = Q_{down} - Q_{dec}$

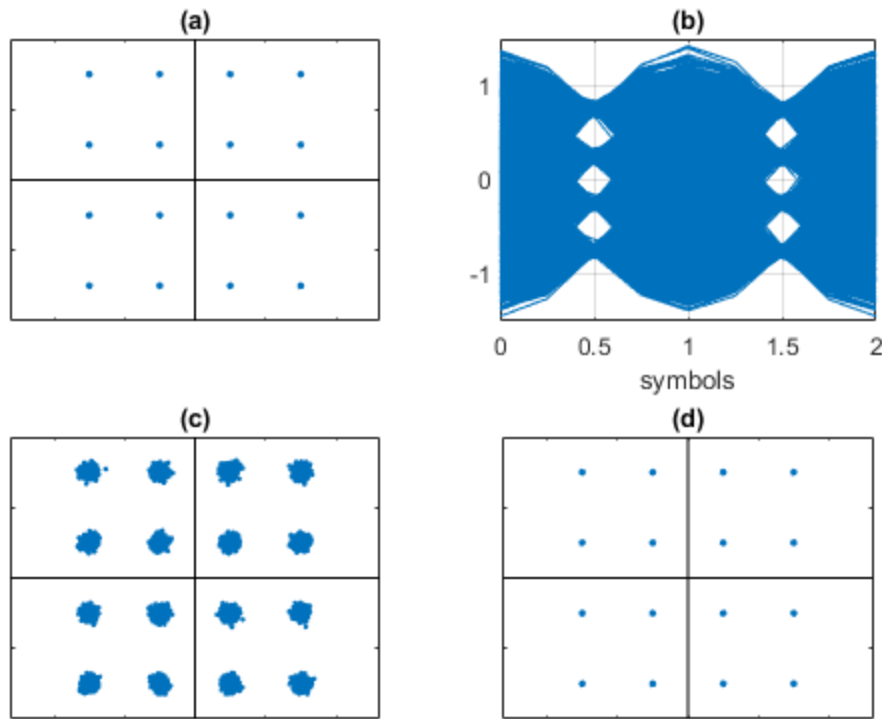


Figure 3. Signals from Figure 2 for QAM-16 modulation with added noise.

a. I_{map}/Q_{map} constellation b. I_{rx} eyeplot c. I_{down}/Q_{down} constellation d. I_{dec}/Q_{dec} constellation

MER Formula

The Modulation Error Ratio is a way to quantify the constellation noise seen in Figure 3c. Figure 4 shows the QAM-16 symbols Q_{dec} vs I_{dec} at the output of the decision block. Also shown is a single received symbol of I_{down}/Q_{down} at the input of the decision block. The modulation error ratio of this single symbol is defined as:

$$MER (one\ symbol) = \frac{I_{dec}^2 + Q_{dec}^2}{dI^2 + dQ^2}$$

Where dI and dQ are the errors with respect to the target symbol I_{dec}/Q_{dec} . In words, MER is the ratio of the target symbol power to the error power. dI and dQ are computed as shown in the block diagram of the decision block (Figure 2b): $dI = I_{down} - I_{dec}$; $dQ = Q_{down} - Q_{dec}$.

If N symbols are received, the MER is defined as the average:

$$MER = \frac{1}{N} \sum_{i=1}^N \frac{I_{dec}(i)^2 + Q_{dec}(i)^2}{dI(i)^2 + dQ(i)^2}$$

If N is large and all symbols are equally likely, we can calculate MER as the ratio of average target symbol power to average error power [4]:

$$MER = \frac{\text{avg. target symbol power}}{\text{avg. error power}} = \frac{\frac{1}{N} \sum_{i=1}^N I_{dec}(i)^2 + Q_{dec}(i)^2}{\frac{1}{N} \sum_{i=1}^N dI(i)^2 + dQ(i)^2} \quad (1)$$

For a given constellation order, the numerator is a constant – call it P_{av} :

$$MER = \frac{P_{av}}{\frac{1}{N} \sum_{i=1}^N dI(i)^2 + dQ(i)^2} \quad (2)$$

Since P_{av} is a constant, we don't have to evaluate it over all of the received symbols. Instead, we can find P_{av} by evaluating the numerator of Equation 1 over N = 4 constellation points in the first quadrant of QAM-16. For a square constellation, we'll use I and Q from the set $\pm k/M^{1/2}$, where k is odd and M is constellation order. For M= 16, the set is $\pm \{1/4, 3/4\}$, and the constellation points in the first quadrant are $(\frac{1}{4}, \frac{1}{4})$; $(\frac{1}{4}, \frac{3}{4})$; $(\frac{3}{4}, \frac{1}{4})$; $(\frac{3}{4}, \frac{3}{4})$. The average constellation power of QAM-16 is thus:

$$P_{av} (QAM16) = \frac{1}{4} * \left(\left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{3}{4}\right)^2 + \left(\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{3}{4}\right)^2 + \left(\frac{3}{4}\right)^2 \right) = \frac{5}{8}$$

We can write the MER in dB as:

$$MER (dB) = 10 \log_{10} P_{av} - 10 \log_{10} \left(\frac{1}{N} \sum_{i=1}^N dI(i)^2 + dQ(i)^2 \right) \quad (3)$$

If noise is the only signal impairment, and if matched (root-Nyquist) filters are used in transmitter and receiver, the MER can equal the signal's carrier-to-noise ratio (C/N). In reality, accurate estimation of C/N is only possible over a limited range. The maximum MER is limited by impairments of the transmitter and receiver themselves. The low end is also limited: when C/N is low enough to cause symbol errors, the calculated MER loses accuracy. This occurs because each decision error causes the

error power calculation for that symbol to be too low. When there are many errors, the calculated MER will be too high.

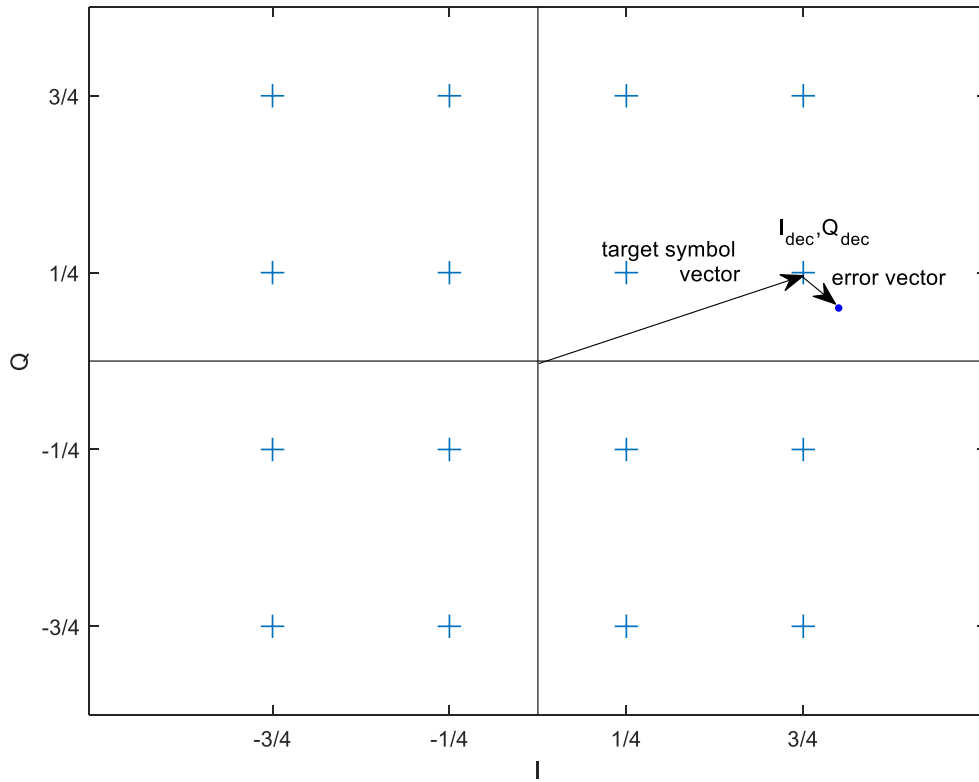


Figure 4. QAM-16 constellation points (symbols), with target signal vector and error vector.

Example

This example computes MER of a QAM-64 signal with added noise. Matlab code that implements the simplified QAM system is listed below. As shown in Figure 2a, Gaussian noise is added to the I and Q channels. Figure 5a shows the spectrum of $I_{\text{chan}} + jQ_{\text{chan}}$, which has a C/N of about 30 dB. An eyeplot of the receiver signal I_{rx} is shown in Figure 5b. The receiver constellation is shown in Figure 5c, along with the value of MER calculated by Equation 3. As expected, the calculated MER matches the C/N.

In calculating MER for QAM-64, we used $P_{\text{av}} = 21/32 = -1.83$ dB. The I and Q values of QAM-64 come from the set $\pm \{1/8, 3/8, 5/8, 7/8\}$. The root-Nyquist filters used have excess bandwidth $\alpha = 0.25$. This gives an occupied signal bandwidth of $1.25 * f_{\text{symbol}} = 1.25 * f_s / 4 = .3125 f_s$ (Figure 5a).

A convenient way to calculate the average error power is to filter $dI^2 + dQ^2$ using a one-pole IIR filter. The output of the filter can be used to compute a running estimate of MER, as shown in Figure 5d.

Finally, Figure 6 shows the receive constellation with no noise added to the transmitted signal. The MER of 41 dB is a measure of the Inter-symbol interference (ISI) [1,2] of the cascaded transmit and receive root-Nyquist filters in our simplified model.

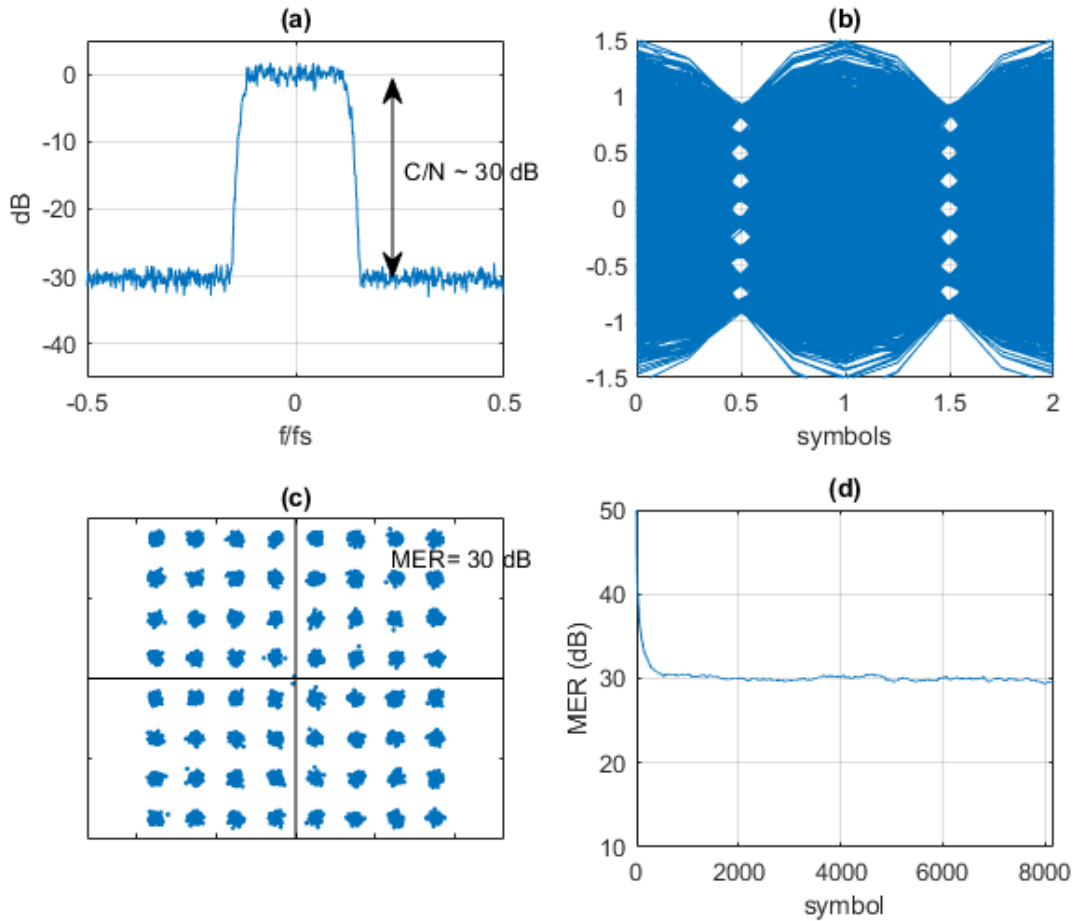


Figure 5. QAM-64 Model Outputs

- a. Spectrum at RX input, showing added noise.
- b. Eyeplot of I_{rx}
- c. Receiver constellation plot.
- d. MER calculated by filtering $dI^2 + dQ^2$ using a one-pole IIR filter.

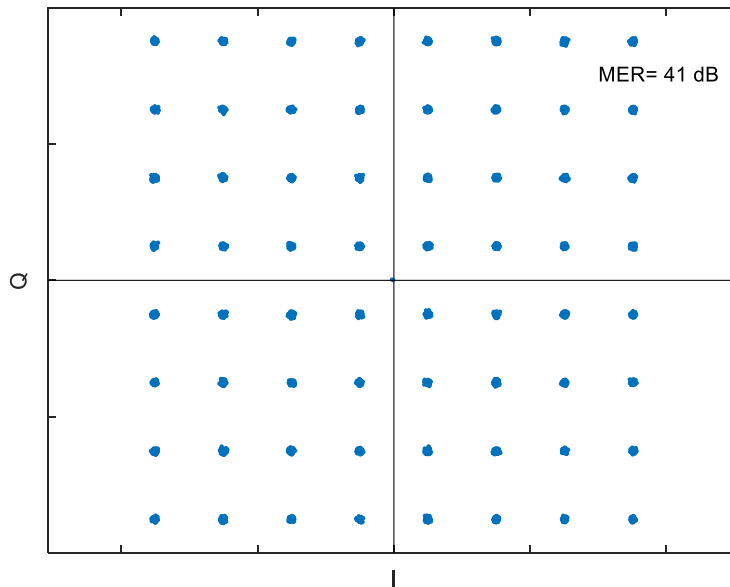


Figure 6. Receive Constellation with no noise added to the transmitted signal.

Matlab script to compute MER of QAM-64 and function eyeplot.m for plotting eye diagram

This script uses two different random number generators:

`rand(1,N)` creates N uniformly-distributed random numbers over [0 1].

`randn(1,N)` creates N normally-distributed (Gaussian) random numbers with mean of zero and variance of one.

```
% mer_qam64.m 10/29/19 rev 4/17/20 Neil Robertson
% Demonstrate MER calculation for QAM-64

% root-nyquist FIR with alpha = .25 and fs= 4*fsymbol from root_nyq3.m
b1= [12 -1 -10 -18 -17 -1 20 34 25 -7 -46 -62 -36 27 92 108 46 -77 -192 ...
    -206 -53 260 643 958];
b_nyq= [b1 1079 flip1r(b1)]/4096;

% COMPLEX BASEBAND TRANSMITTER (simplified)
% generate random equally likely QAM-64 symbols from the set
% +/-{1/8 3/8 5/8 7/8}

N= 8192; % number of symbols
u= round(8*rand(1,N) -.5); % equally likely integers over [0 7]
v= round(8*rand(1,N) -.5);
```

```

Imap= (2*u- 7)/8;           % QAM-64 symbols
Qmap= (2*v- 7)/8;

% upsample by 4 to 4 samples/symbol
Iup= zeros(1,4*N);
Qup= zeros(1,4*N);

Iup(1:4:4*N)= Imap;
Qup(1:4:4*N)= Qmap;

% Filter Iup/Qup using root-Nyquist filter
Itx= 4*filter(b_nyq,1,Iup);
Qtx= 4*filter(b_nyq,1,Qup);

% COMPLEX BASEBAND CHANNEL
% Impairment = Gaussian noise
sigma= .0495;           % noise standard deviation

Ichan= Itx + sigma/sqrt(2)*randn(1,4*N);
Qchan= Qtx + sigma/sqrt(2)*randn(1,4*N);

% COMPLEX BASEBAND RECEIVER (simplified)
% input sample rate = 4 samples/symbol

% root-Nyquist receive filter
Irx= filter(b_nyq,1,Ichan);
Qrx= filter(b_nyq,1,Qchan);

% downsample filter output by 4, choosing correct timing
Idown= Irx(41:4:end);
Qdown= Qrx(41:4:end);

% Decision block ( slicer ) error
levels= [-7 -5 -3 -1 1 3 5 7]/8;           % target levels
L= length(Idown);
for i= 1:L
    dI(i)= min(abs(Idown(i) - levels));    % error
    dQ(i)= min(abs(Qdown(i) - levels));
end

% MER calculation
Pavg = -1.83;           %dB Pav of QAM-64 = 10*log10(21/32)
e_sq= (dI.^2 + dQ.^2); % error squared
e_sq_avg= sum(e_sq)/L;

mer_dB= Pavg - 10*log10(e_sq_avg)

% Filter e_sq using first-order IIR
% IIR coeffs
b= 1/256;
a= [1 b-1];

e_sq_filt= filter(b,a,e_sq); % filter e_sq
mer_filt_dB= Pavg -10*log10(e_sq_filt);

```



```

% PLOTTING
% Constellation at transmitter mapper output
%plot([-2 2],[0 0],'k',[0 0],[-2 2],'k',Imap,Qmap, '.', 'markersize',8)
%axis([-1 1 -1 1]),axis('equal'),title('(a)')

% Spectrum of signal + noise at receiver input
y= Ichan(1:16*1024) + j*Qchan(1:16*1024);

[P,f]= pwelch(y*.62,hanning(1024),512,1024,1);
P= fftshift(P);           % center spectrum at 0 Hz
f= f- .5;
PdB= 10*log10(P);
subplot(221),plot(f,PdB),grid
axis([-0.5 0.5 -45 5]),text(0.26, -14, 'C/N ~ 30 dB')
xlabel('f/fs'),ylabel('dB'),title('(a)')

% eyeplot of Irx at receiver root Nyquist filter output (4 samples/symbol)
subplot(222),eyeplot(Irx(51:end),4)
xlabel('symbols'),title('(b)')

% receiver constellation plot before decision block
mer_dB= round(10*mer_dB)/10;           % round to tenths place
subplot(223),plot(Idown,Qdown, '.', [-2 2],[0 0],'k',[0 0],[-2 2],'k')
axis([-1 1 -1 1]),axis('equal'),title('(c)')
xticks([-1 -0.5 0 0.5 1]),yticks([-1 -0.5 0 0.5 1])
xticklabels({}),yticklabels({})
text(.6,.76,['MER= ',num2str(mer_dB),' dB'])

% output of MER filter
subplot(224),plot(mer_filt_dB),grid
axis([0 L 10 50])
xlabel('symbol'),ylabel('MER (dB)'),title('(d)'),figure

%eyeplot.m  10/29/19 Neil Robertson
% Plot eye diagram of vector x
% fs = samples per symbol (integer)
% A = matrix with each column containing 2*fs + 1 samples.
%     Each column is a trace in the plot.
%
function eyeplot(x,fs)
L= 2*fs;
cols= fix(length(x)/L -1);
x= x(1:L*(cols+1)); % truncate x to an integer multiple of L

for j= 1:cols % column index
    k= j-1;
    A(:,j)= x(k*L+1:(k+1)*L+1); % column = one trace of length 2*fs +1
end

t= 0:1/fs:2;
plot(t,A, 'color',[0 .447 .741]),grid
axis([0 2 -1.5 1.5])

```

References

1. Rice, Michael, *Digital Communications*, Pearson, 2009, Section A.2.
2. Course Notes EE4061, "Intersymbol Interference/Nyquist Pulse Shaping", Georgia Institute of Technology, 2011. <http://wireless-systems.ece.gatech.edu/4601/lectures-2011/week10.pdf>
3. William H. Tranter, et. al., *Communication Systems Simulation*, Prentice Hall, 2004, section 4.1.1.
4. Cisco Systems, "Digital Transmission: Carrier-to-Noise Ratio, Signal-to-Noise Ratio, and Modulation Error Ratio", 2006.
http://www.mdmit.pl/apps/mdmit/download/Mibs%20for%20cable%20modems/Hi_res_CNR-SNR_WP_1115b.pdf

Neil Robertson November, 2019
 revised 12/7/19
 revised mer_qam64.m 4/17/20
 revised 12/5/20