Hideo Okawara's
Mixed Signal Lecture Series

DSP-Based Testing – Fundamentals 39
F-matrix Cable Simulation

Verigy Japan
August 2011

Preface to the Series

ADC and DAC are the most typical mixed signal devices. In mixed signal testing, analog stimulus signal is generated by an arbitrary waveform generator (AWG) which employs a D/A converter inside, and analog signal is measured by a digitizer or a sampler which employs an A/D converter inside. The stimulus signal is created with mathematical method, and the measured signal is processed with mathematical method, extracting various parameters. It is based on digital signal processing (DSP) so that our test methodologies are often called DSP-based testing.

Test/application engineers in the mixed signal field should have thorough knowledge about DSP-based testing. FFT (Fast Fourier Transform) is the most powerful tool here. This corner will deliver a series of fundamental knowledge of DSP-based testing, especially FFT and its related topics. It will help test/application engineers comprehend what the DSP-based testing is and assorted techniques.

Editor's Note

For other articles in this series, please visit the Verigy web site at
www.verigy.com/go/gosemi.

Preface

The wired LAN 100BASE-T system recommends using specified cables such as the category 5 twisted pair cables. DUT boards for RF or high-speed device testing often employ 50Ω micro-strip lines for routing signals. The signals traveling on these media are more or less deteriorated by the media characteristics. Therefore communication devices are often tested tolerance to the signal degradation such as distortion and jitter. In order to emulate such a mission mode environment, the simplest and the most straightforward method is to employ the real cables, boards and circuits used in the real environment. However, this approach has issues in practicability, reliability, maintainability and so on. The point is to generate a mission mode distorted signals. So it is acceptable to generate such signals by simulation. This month article discusses a procedure to synthesize a distorted waveform by combining the conventional F-matrix of transmission line model and FFT&IFFT method.

Introduction

A LAN transceiver is the front-end devices in local area network communication systems. This device is located between the computer I/O port inside the equipment and the transmission line outside. The 100BASE-T LAN transceiver is still used in many office and home environment. 100BASE-T LAN transceivers transmit and receive a specific three-level signal that is called MLT-3. The signal travels through maximum 100m twisted pair cable. Therefore the transceivers must be tested with using the signals containing specific distortion caused by such environment.
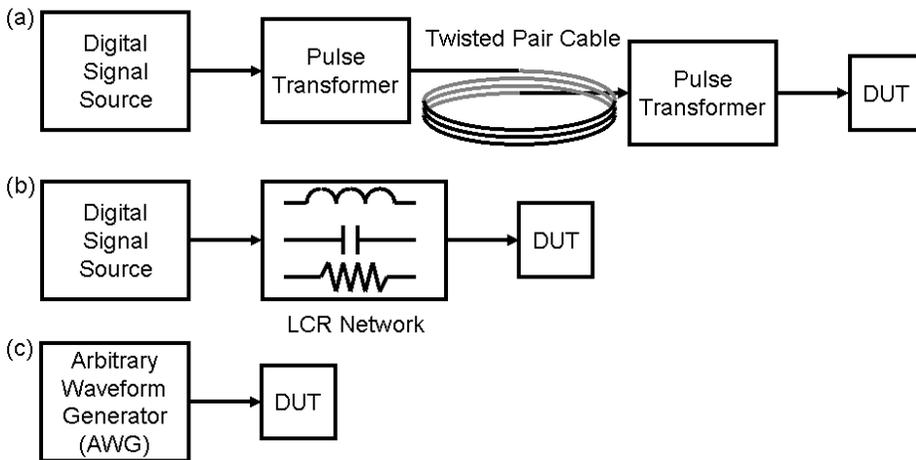
Figure 1:   Emulation of LAN Cable System

Figure 1 illustrates how specific distorted waveforms can be asserted to the DUT receiver. The configuration of (a) is a very straightforward method using real cable and components. The signal source generates MLT-3 square waveforms. There is no trick in this approach. The output signal is truly distorted and deteriorated by real components. It could be acceptable for the characterization on the bench with manually connecting, disconnecting and changing the length of cables or types of components. However, the 100m cable assembly is really a big reel so that you cannot keep it on your DUT board. You cannot provide different lengths of cables on the DUT board. Many cable manufacturers make the category 5 cables. The specification of the cable is not so precise or tight so that each cable has different performance. Consequently it is not easy to realize the same or similar test condition on multiple DUT boards in various test sites. Therefore this approach has big issues of poor flexibility, maintainability and repeatability. In (b) the LCR lump components emulates the cable system in (a). Probably the size of the circuits would be very much smaller than the real cable assembly. It could easily be implemented on the DUT board with reasonable size and space so this is more feasible approach. However, if you want to test several cable lengths, it would

consume space with switching mechanism. Consequently the second approach has still issues in flexibility and maintainability. In (c) there are no cumbersome components, nothing integrated on the board. All you have to do is just creating various waveforms you need for different cable lengths or various components, and the AWG emulates the distorted waveforms. This is definitely flexible. You can always stimulate exactly the same waveform at any test site so that this method is excellent in repeatability and maintainability. How you can simulate the waveforms is the theme of this article.

MLT-3

As seen in Figure 1(a), transformers are implemented in the 100BASE-T system so that DC cannot be transmitted. So the signal used in the 100BASE-T is encoded to an AC signal called MLT-3 (Multi Level Transmission 3-Level), which is a line code constructed with three levels of -1, 0 and +1. The embedded data is basically 0 and 1. When "0" comes, the current voltage level is kept unchanged. When "1" comes, if the current level is +1 or -1, the next level becomes 0; if the current level is 0, the next level becomes +1 or -1 depending on the former level. If the former level was -1 and the current level is 0, then the next level becomes +1. If the former level was +1 and the current level is 0, then the next level becomes -1. So the waveform becomes as Figure 2 depicts.

Data
1 1 0 1 0 0 1 1 0 1 0 1 1 1 0

MLT-3

1

0

-1
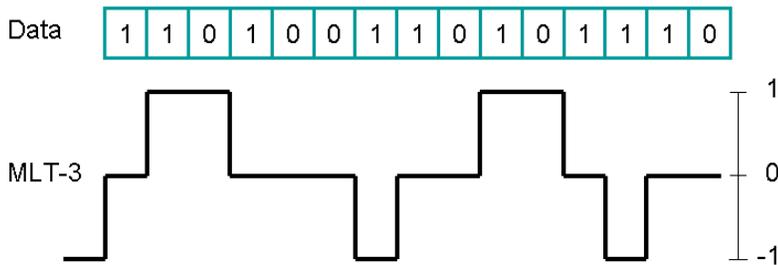
Figure 2:   Example of MLT-3

In the MLT-3 coding, "1" makes level changes but "0" does not. Continuous 1's become AC signal, but continuous 0's become DC of -1, 0 or +1. Therefore continuous 0's must be avoided in the data stream.

4B5B

The 4B5B conversion is provided to suppress long, continuous 0's in the data stream. Serial data stream is divided into 4-bit chunks, and a 4-bit data is converted into a specific 5-bit data as Table 1 illustrates. Some more 5B codes are actually assigned for specific signaling. At least one out of 5 bits is "1" in 5B code patterns.

4B	HEX	5B	4B	HEX	5B
0000	0	11110	1000	8	10010
0001	1	01001	1001	9	10011
0010	2	10100	1010	A	10110
0011	3	10101	1011	B	10111
0100	4	01010	1100	C	11010
0101	5	01011	1101	D	11011
0110	6	01110	1110	E	11100
0111	7	01111	1111	F	11101

Table 1:   4B5B Conversion Table

Consequently the original data in the 100BASE-T system is encoded according to the 4B5B rule in advance, and then converted into MLT-3 line code.

LAN Cable System Model

The configuration of Figure 1 (a) is actually a simplified model of the LAN cable system, which can be described as the circuit diagram in Figure 3. The components used here is a twisted pair cable and pulse transformers. The typical specifications of the components are listed in Figure 4.
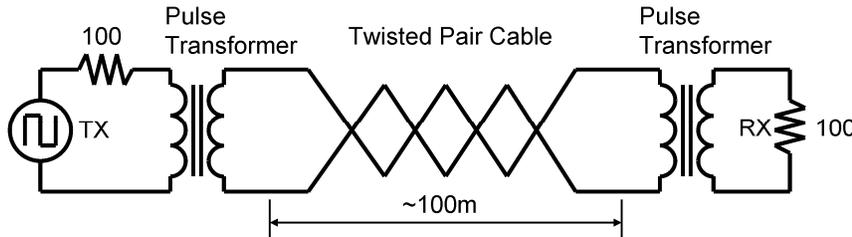
Figure 3:   LAN Cable System Model
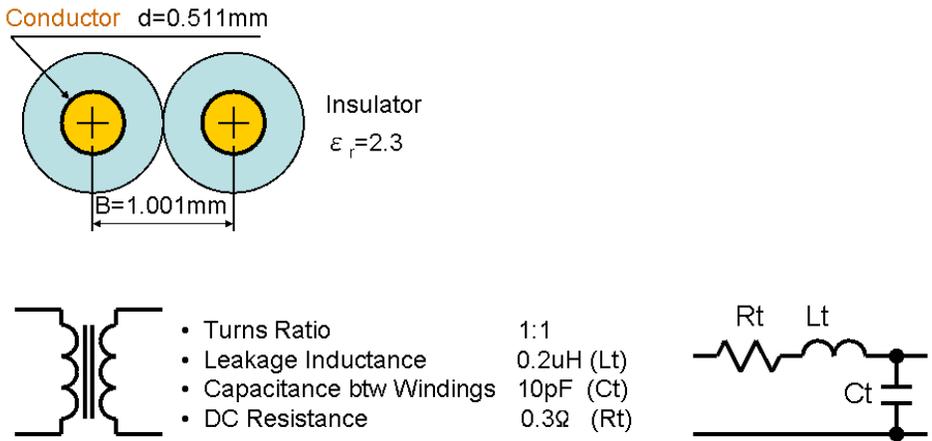
Figure 4:   Typical Cable Pair and Pulse Transformer Specifications

The skin effect is well known in very high frequency applications. The skin depth δ can be defined as Equation (1).

$$\delta = \sqrt{\frac{\rho}{\pi \cdot f \cdot \mu_o \mu_r}} \quad [m] \tag{1}$$

The conductor of the cable is made of copper. The resistivity of copper $\rho=1.72 \times 10^{-8}$ [Ω•m], f [Hz] denotes the frequency, the permeability of vacuum $\mu_o = 4\pi/10^7$ [H/m] and the relative permeability of copper $\mu_r=1.0$.

The resistance of the parallel pair cable per unit length that is 1[m] is denoted as Ro. Equation (2) defines Ro with using the skin depth.

$$R_o = \frac{2\rho}{\pi d \cdot \delta} \quad [\Omega/m] \tag{2}$$

Here is the absolute permittivity $\varepsilon_o=8.854 \times 10^{-12}$ [F/m] in the vacuum condition. The typical insulator of the cable is made of FEP (Fluorinated Ethylene Propylene) and its relative permittivity is $\varepsilon_r=2.3$. Then Equation (3) defines the unit capacitance of the parallel pair cable Co.

$$C_o = \frac{\pi \varepsilon_o \varepsilon_r}{\ln \dfrac{B + \sqrt{B^2 - d^2}}{d}} \quad [\text{F/m}] \tag{3}$$

The relative permeability of the insulator is $\mu_e = 1.0$ and the relative permeability of the copper is $\mu_r = 1.0$ so that Equation (4) defines the unit inductance of the parallel pair cable Lo.

$$L_o = \frac{\mu_o \mu_e}{\pi} \ln \frac{B + \sqrt{B^2 - d^2}}{d} + \frac{R_O}{\omega} \quad [\text{H/m}] \tag{4}$$

The first term is the external inductance between the conductors and the second term is the approximated internal inductance of the conductors influenced by the skin effect. The parameter ω is the angular frequency or $2\pi f$.

The FEP is extremely excellent insulator up to the frequency range of GHz so that the unit conductance Go is good enough to be approximated as zero.

F-matrix

The model of the system in Figure 3 can be described as the circuit diagram in Figure 5. The transformer is very simplified as the DC resistance, the leakage inductance and the capacitance between the coils. The transmission line is described as the distributed constant circuit with the four unit parameters Ro, Go, Lo and Co which are already discussed in the previous section. The block diagram includes five sections of F-matrix as $F1$, $F2$, $F3$, $F4$ and $F5$.
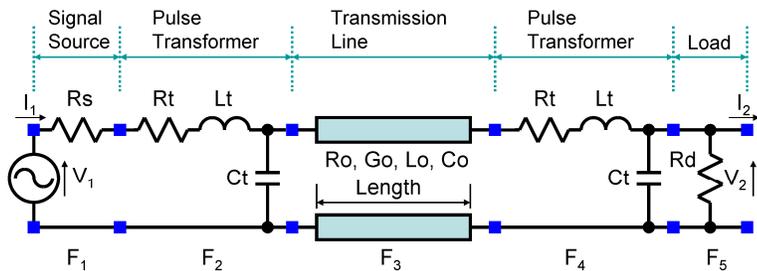
Figure 5:　　　**Equivalent Circuit and F-matrices**

The five F-matrices are combined together into a single F-matrix F_0 as Figure 6 shows.

$$F_0 = F_1 \cdot F_2 \cdot F_3 \cdot F_4 \cdot F_5$$
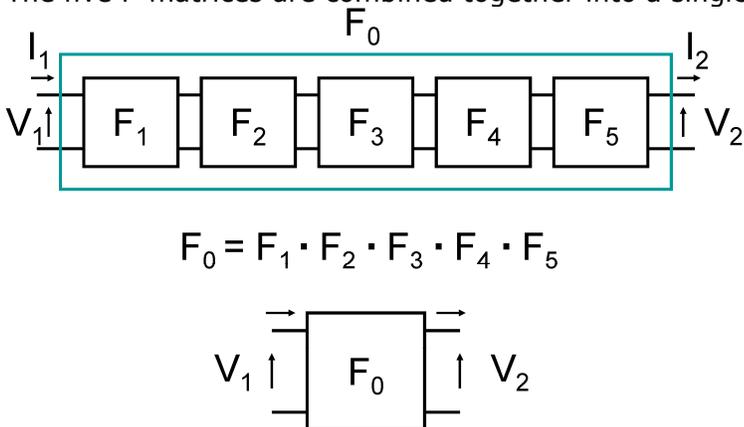
Figure 6:　　　**F-matrices**

Waveform Simulation

The waveform can be simulated according to the diagram in Figure 6. The original input signal which is actually square waveform of MLT-3 is provided at first. The N-point input signal waveform

from x_0 to x_{N-1} is FFT-converted to the frequency spectrum components from G_0 to $G_{N/2-1}$. The F-matrix F_0 is calculated from the DC to the frequency bin $N/2-1$, and multiplied to the G_k component ($k=0,1,2,…,N/2-1$) one by one, going to from H_0 to $H_{N/2-1}$. Each one of the processed spectrum component H_k derives the complex conjugate spectrum H_{N-k}, making up the whole spectrum set from H_0 through H_{N-1} for preparing to the inverse FFT (IFFT). (See appendix for the program.)
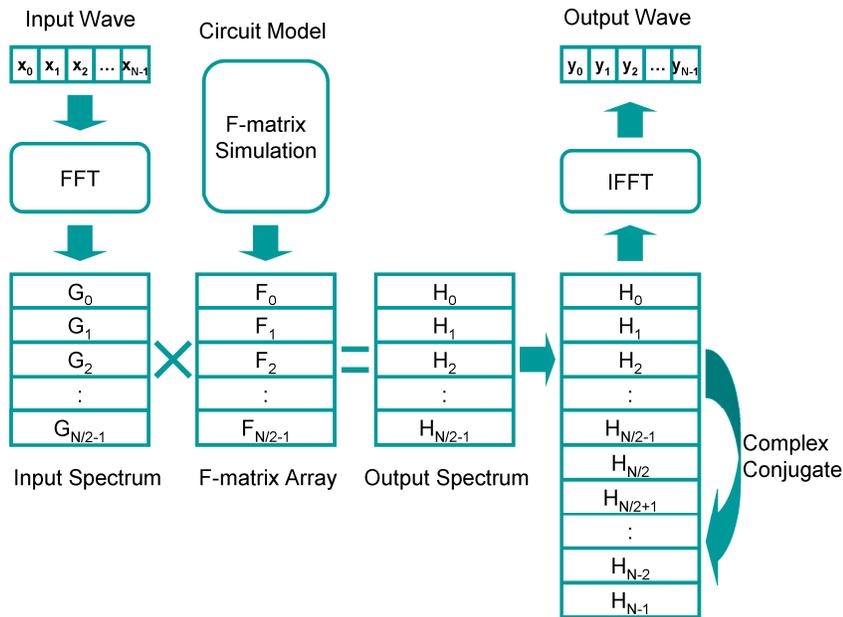
Figure 7:   Simulation Flow

The original data is 512 bits binary of PRBS bit stream at 100Mbps, which is encoded with the 4B5B rule so that the data length is modified to 640 bits at 125Mbps and then it is encoded with the MLT-3 signal. The sampling rate is set as 4Gsps so that the waveform size grows 20480 (=640x32) for the AWG.
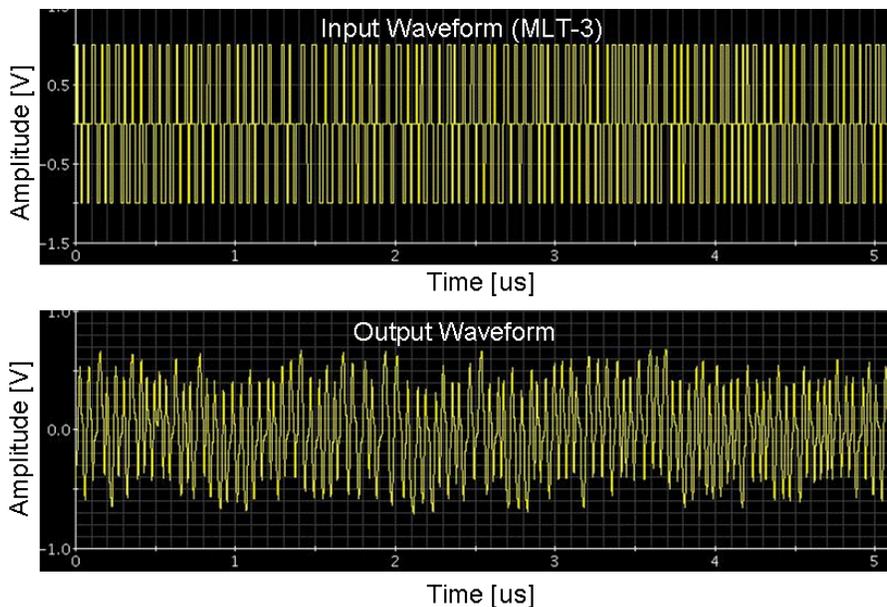
Figure 8:   Input/Output Waveforms

Figure 8 illustrates the input MLT-3 signal at V_1 in Figure 6 or $\{x_k\}$ in Figure 7 and the simulated cable output signal at V_2 in Figure 6 or $\{y_k\}$ in Figure 7. You may notice that even if the MLT-3 is DC balanced signal, the actual DC offset would drift at V_2 because of the frequency characteristics of the cable plus transformers system. This may cause bit errors when it is received. In the actual test specification "killer pattern" is provided, which generates extreme DC drift.

Figure 9 shows the close-up of 0 to 1us of Figure 8. You may notice the simulation result appears approximately 510nsec delayed. This is caused by the cable length of 100m, and the F-matrix simulation can show this effect. By the way, there is a line code violation in the MLT-3 signal, which is intentionally buried for ease of finding the location.
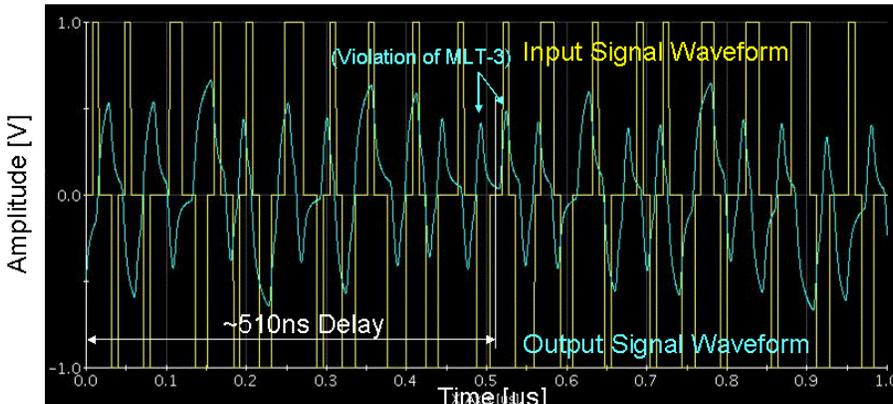
Figure 9:   Close-up Input/Output Waveforms

For clearer waveform comparison, the starting 500ns of the input and output waveforms are aligned and separately illustrated in Figure 10.
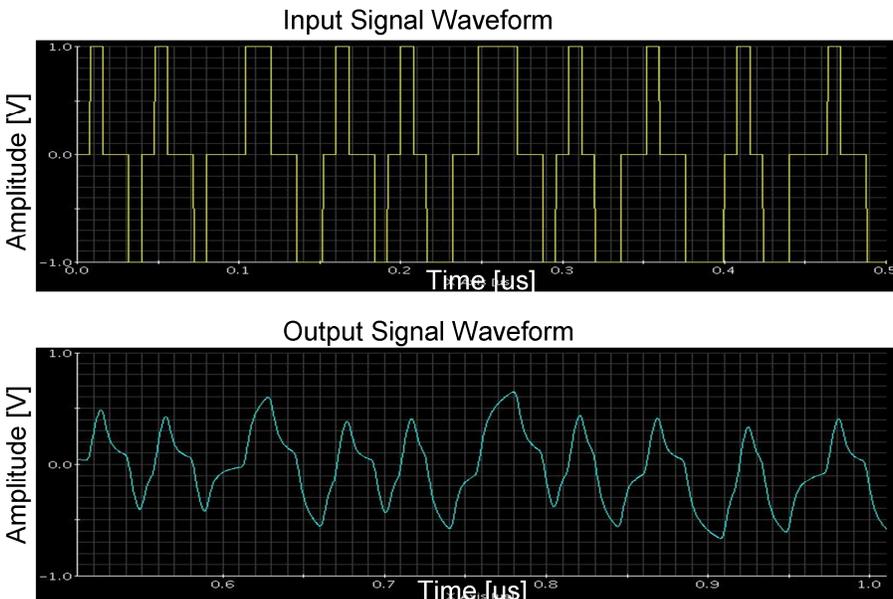
Figure 10:   Input/Output Waveforms

By performing the FFT to the waveforms, the frequency spectrum of the waveform in Figure 8 appears in Figure 11. The yellow shows the input spectrum and the blue shows the output spectrum. It represents the frequency characteristics of the transformers plus 100m cable system. This seems to be about 150MHz to 200MHz.
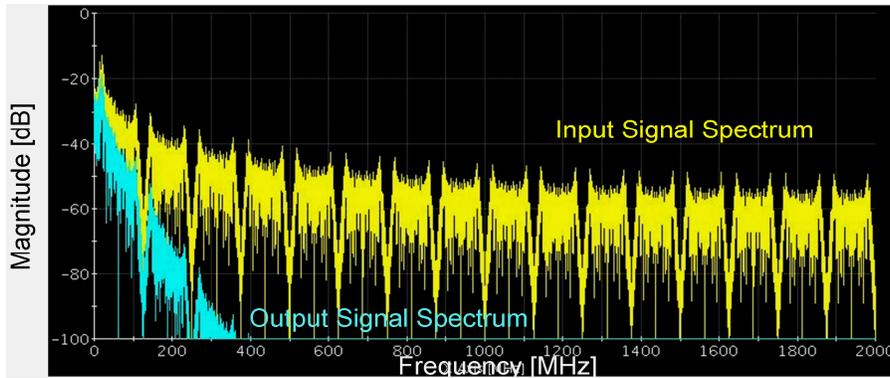
Figure 11:   Input/Output Spectra

RC Network Simulation

According to Figure 1 (b), a passive network is constructed as Figure 12, emulating the cable network in Figure 5. The differential configuration (a) is simplified to the single-ended (b). The simulation results are shown in Figure 13. The waveform and the spectrum look similar to the cable system.
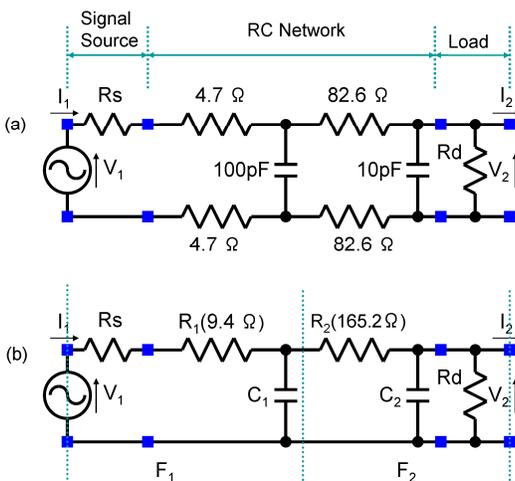
Figure 12:   RC Network
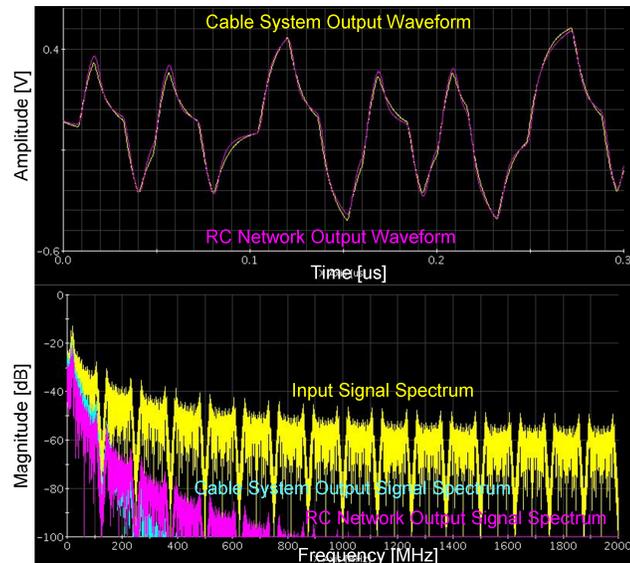
Figure 13:   Simulation Result (Red)

Appendix – Source Code Example

The following document should be helpful for more information about the code.
"DSP-Based Testing – Fundamentals 35   F-matrix Simulation"

```
01:  INT              i;
02:  DOUBLE           w,dFreq;
03:  DOUBLE           dFmlt3=125 MHz;// MLT-3 Rate 125Mbps
04:  INT              Nbits=512;      // 4B size=512 bits
05:  INT              N5b=Nbits/4*5; // 5B size
06:  INT              Nx=32;          // 32 times oversampling
07:  INT              N=Nx*N5b;       // Total Wave Size (for AWG)
08:  DOUBLE           dFs=dFmlt3*Nx; // AWG Sampling Rate
09:  INT              Nsp=N/2;        // Spectrum Size (Half)
10:  DOUBLE           dChresln=dFs/N; // Spectrum Resolution
11:  ARRAY_D          dWave,dWave1;   // Input/Output Waveform
12:  ARRAY_COMPLEX CSp,CSp1;          // Input/Output Spectrum
13:  ARRAY_COMPLEX CWave1;            // Output Waveform
14:  COMPLEX          One=COMPLEX(1.0,0.0);
15:  COMPLEX          Zero=COMPLEX(0.0,0.0);
21:  DOUBLE Rs=100.0;       // [ohms]
22:  DOUBLE Rd=100.0;       // [ohms]
23:  DOUBLE Rt=0.3;         // Pulse Transformer
24:  DOUBLE Ct=10.0 pF;     //
25:  DOUBLE Lt=200.0 nH;    // #define nH *1.0e-9
26:  DOUBLE Length=100.0;   // [m]
27:  DOUBLE B=1.001 mm;     // Distance #define mm *1.0e-3
28:  DOUBLE d=0.511 mm;     // diameter
29:  DOUBLE mur=1.0;        // Relative Permeability
30:  DOUBLE mu0=4.0*M_PI*1.0e-7; // [H/m] Absolute Permeability
31:  DOUBLE mu=mur*mu0;     // Permeability
32:  DOUBLE Lx=(mu/M_PI)*log((B+sqrt(B*B-d*d))/d); // [H/m]
33:  DOUBLE Lo;             // [H/m]
34:  DOUBLE er=2.3;
35:  DOUBLE e0=8.854e-12; // [F/m]
36:  DOUBLE epsilon=er*e0;
37:  DOUBLE Co=M_PI*epsilon/(log((B+sqrt(B*B-d*d))/d)); // [F/m]
38:  DOUBLE Go=1.0e-12;     // [S/m]
39:  DOUBLE rho=1.72e-8;   // [ohms/m] for Copper
40:  DOUBLE Rdc=(rho/(M_PI*(d/2.0)*(d/2.0)))*2.0; // [ohms/m]
41:  DOUBLE Ro;             // [ohms/m]
42:  ARRAY_COMPLEX CGain;
43:  ARRAY_D dGain,dPhase;
44:  Fmatrix F0,F1,F2,F3,F4,F5;
```

```
51:    CGain.resize(Nsp);
52:    for (i=0;i<Nsp;i++) {
53:        dFreq=i*dFresln;
54:        w=2.0*M_PI*dFreq; // w = Omega
55:        if (w==0.0) {
56:            Ro=Rdc;
57:            Lo=Lx;
58:        } else {
59:            depth=sqrt(rho/(M_PI*dFreq*mu)); // Skin Depth
60:            Ro=2.0*rho/(M_PI*d*depth); // Skin Effect 2.0:parallel
61:            Lo=Lx+Ro/w;
62:        }
63:
64:        F1=Fz(COMPLEX(Rs,0.0));
65:        F2=Fzy(COMPLEX(Rt,w*Lt),COMPLEX(0.0,w*Ct));
66:        F3=Fxline(Ro,Go,Lo,Co,Length,dFreq);
67:        F4=Fzy(COMPLEX(Rt,w*Lt),COMPLEX(0.0,w*Ct));
68:        F5=Fy(COMPLEX((1.0/Rd),0.0));
69:
70:        F0=F1*F2*F3*F4*F5;
71:        CGain[i]=One/F0.a;
72:    }
81:    // dWave[] contains the MLT-3 square waveform
82:    DSP_FFT(dWave,CSp,RECT);       // Half plane spectrum
83:    DSP_MUL_VEC(CSp,CGain,CSp1); // 1/A(F0)
84:    CSp1.resize(N);                // Full Plane Extention
85:    for (i=1;i<Nsp;i++) CSp1[N-i]=Conjugate(CSp1[i]);// for IFFT
86:    CSp1[0].real()=2.0*CSp1[0].real(); // DC Compensation
87:    CSp1[Nsp]=Zero;
88:
89:    IFFT(CSp1,CWave1);             // IFFT for N=non 2^n
90:    dWave1=CWave1.getReal();       // Output Waveform
91:
```