Hideo Okawara's
Mixed Signal Lecture Series

DSP-Based Testing – Fundamentals 43
Jitter Estimation from Phase Noise Data

Verigy Japan
January 2012

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

There is a popular signal source analyzer such as Agilent's E5052B providing phase noise data for various clock sources. From a high-speed digital applications' point of view, jitter value is easier to understand the integrity of signal than phase noise data. Let's exercise how to estimate a jitter from the phase noise data.

Jitter from Phase Noise Representation

Phase noise data is often listed as a table of dBc/Hz magnitude vs. frequency offset from the test signal frequency. Table 1 is an example of phase noise data.

Offset	10Hz	100Hz	1kHz	10kHz	100kHz
Phase Noise	-100dBc/Hz	-130dBc/Hz	-140dBc/Hz	-145dBc/Hz	-145dBc/Hz

Table 1 Typical Phase Noise Data

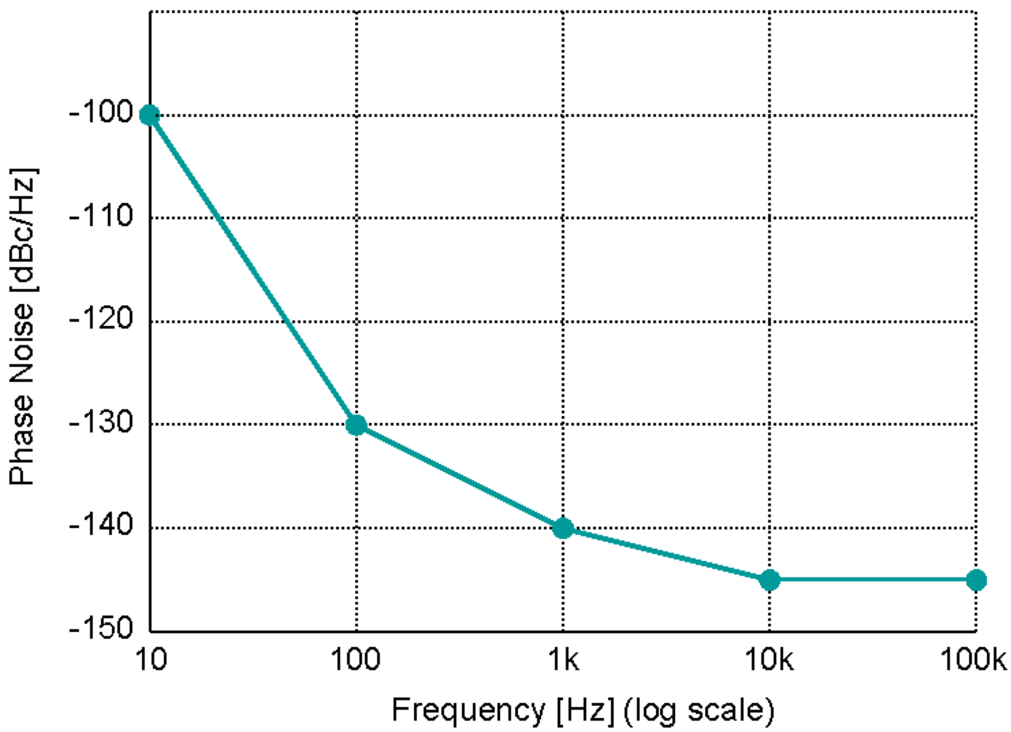

Figure 1 Graphical Representation of Typical Phase Noise Data in Table 1

Figure 1 illustrates the content of Table 1 as a graphical image. Phase noise to jitter conversion is to integrate the phase noise power with a step of 1 Hz bandwidth. The horizontal axis is usually displayed with logarithmic scale, and the phase noise value is often specified by the unit of [dBc/Hz] so that you have to be careful to calculate the phase noise power with the 1 Hz bandwidth.
In order to estimate the phase noise magnitude at every Hz between indicated frequencies, a linear approximation is good enough. See Figure 2. The X-axis value of a frequency F is represented as X=log(F). When a phase noise level is indicated as Y [dBc/Hz], the power of the phase noise P can be figured out as $P=10^{Y/10}$ based on the equation of Y=10log(P).

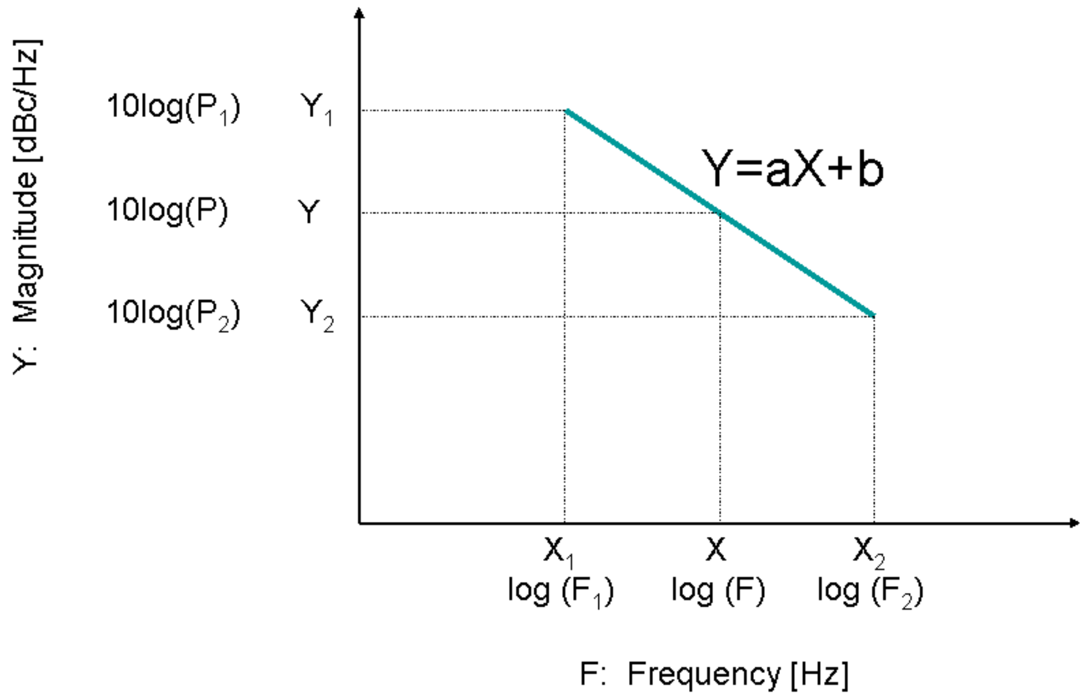

Figure 2 **Linear Approximation**

The linear approximation of a phase noise level Y at a frequency X between F_1 and F_2 can be represented as follows;

$$Y = a \cdot X + b \qquad (1)$$

Two frequencies of F_1 and F_2 are given, they are converted to $X_1=\log(F_1)$ and $X_2=\log(F_2)$ at first. Then the slope *a* and the offset *b* can be calculated as follows;

$$Y_1 = a \cdot X_1 + b \qquad (2)$$
$$Y_2 = a \cdot X_2 + b \qquad (3)$$
$$Y_1 - Y_2 = a \cdot (X_1 - X_2) \qquad (4)$$
$$a = \frac{Y_1 - Y_2}{X_1 - X_2} \qquad (5)$$
$$b = Y_1 - a \cdot X_1 \qquad (6)$$

So the phase noise [dBc/Hz] at a frequency F [Hz] between F_1 and F_2 is calculated as $Y=a\bullet\log(F)+b$. Then the power of the 1 Hz bandwidth is calculated as $P=10^{Y/10}$.
Figure 3 is an example phase noise data measure with a pure clock source N2397A by using Agilent E5052B. The signal frequency is 122.88 MHz so that the phase noise is integrated from 10 Hz to 245.76 MHz. List 1 shows the whole calculation processing. At Line 45, firstly the total power is doubled because the phase noise exists in the high and the low sidebands of the signal tone. Secondly the power is square rooted for the rms value and finally it is divided by the angular frequency of the test signal.[1] The integral is performed by the step of 1 Hz so that it may take several tens of seconds to complete based on the computer speed. There may be cleverer method to do this job. However, the calculation time is not a matter for this calculation. The calculation

[1] DSP-Based Testing – Fundamentals 42 Jitter Calculation by Spectrum

result shows 0.423 [ps.rms] of jitter, which is really excellent performance. But this is an example test result and it does not always guarantee this performance to N2397A.

```
INT     i,j,N;
DOUBLE  Frequency,PhaseNoise,P;
DOUBLE  X1,Y1,X2,Y2,X,Y,a,b,Jitter,Jps;
ARRAY_D F,dBcHz;
Frequency=122.88 MHz;
F.resize(8);         dBcHz.resize(8);
F[0]=10.0 Hz;        dBcHz[0]= -100.1;      // [dBc/Hz]
F[1]=100.0 Hz;       dBcHz[1]= -124.5;      // [dBc/Hz]
F[2]=1.0 kHz;        dBcHz[2]= -142.1;      // [dBc/Hz]
F[3]=10.0 kHz;       dBcHz[3]= -152.4;      // [dBc/Hz]
F[4]=100.0 kHz;      dBcHz[4]= -156.1;      // [dBc/Hz]
F[5]=1.0 MHz;        dBcHz[5]= -156.7;      // [dBc/Hz]
F[6]=10.0 MHz;       dBcHz[6]= -156.7;      // [dBc/Hz]
F[7]=2.0*Frequency; dBcHz[7]=dBcHz[6];      // [dBc/Hz] 245.76 MHz
PhaseNoise=0.0;
for (i=0;i<7;i++) {                         // 0,1,2,..,6
   X1=log10(F[i]);    Y1=Y[i];              // 0..6
   X2=log10(F[i+1]);  Y2=Y[i+1];            // 1..7
   a=(Y1-Y2)/(X1-X2);
   b=Y1-a*X1;
   N=(INT)(F[i+1]-F[i]);
   for (j=0;j<N;j++) {
      X=log10(F[i]+(DOUBLE)j);                      // log [Hz]
      Y=a*X+b;                                      // [dBc/Hz]
      P=pow(10.0,Y/10.0);
      PhaseNoise=PhaseNoise+P;
   }
}
Jitter=sqrt(2.0*PhaseNoise)/(2.0*M_PI*Frequency); // [s.rms]
Jps=Jitter/(1.0 ps);                              // [ps.rms]
```

List 1: Example Program Code to Convert Phase Noise to Jitter

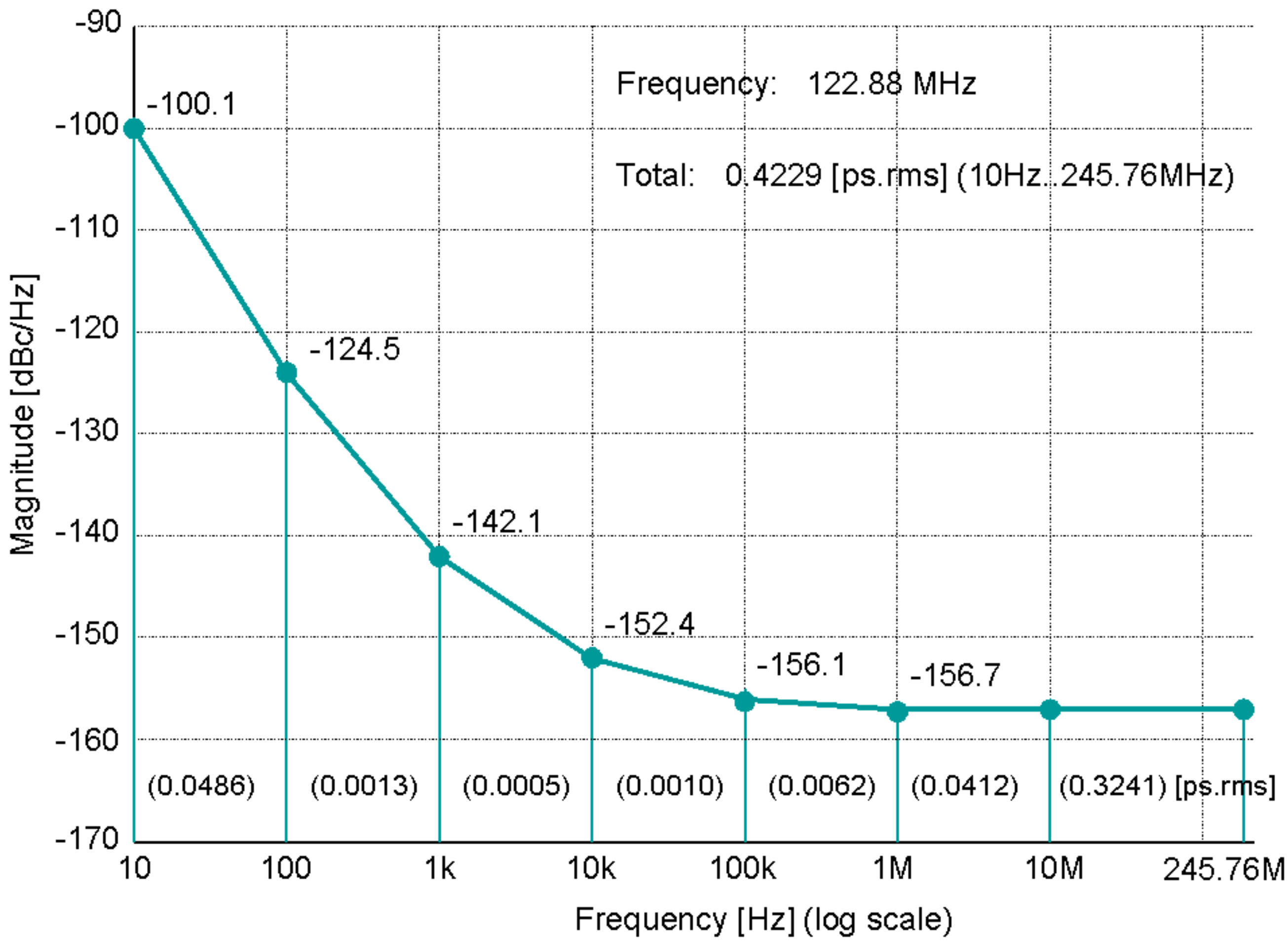

Figure 3 **Measurement Result Example of Pure Clock Source N2397A**