NRZ TEST-PATTERN PROPERTIES, SUCH AS DATA RATE AND PATTERN LENGTH, DETERMINE IMPORTANT TIME- AND FREQUENCY-DOMAIN CHARACTERISTICS.

Spectral content of NRZ test patterns

Many digital-communications systems use non-return-to-zero (NRZ) signaling, and system designers have created many NRZ test patterns to test and verify their products. These patterns usually either simulate actual data or stress certain aspects of the system. To understand the effects of the various test patterns on a system, it is important to understand the frequency characteristics of both the test pattern and the system under test.

NRZ TEST PATTERNS

NRZ waveforms are inherently time-domain signals. NRZ-signaling systems assign a unique time slot of duration $T_B$—the bit period—to each bit. The signal is either high, representing a one, or low, representing a zero, during the entire bit period.

Each bit in a random NRZ sequence has a 50% probability of being a one or a zero, regardless of the state of the preceding bits. Therefore, large sequences of CIDs (consecutive identical digits) are possible.

Designing high-speed systems that can work with random data can be difficult due to the low-frequency content from the long sequences of CIDs in the data signal.

Data encoding or scrambling converts random data into a more manageable form. Ethernet, Fibre Channel, and high-speed-video applications use 8b10b encoding, which replaces 8 bits of data with a 10-bit symbol. The extra bits balance the pattern, making the number of ones equal to the number of zeros for a given interval, and limit the maximum number of CIDs. The encoding algorithm also improves the BER (bit-error rate) by mapping each 8-bit word to specific symbols in the 10-bit signal space that the receiver can easily distinguish from other 10-bit symbols. Other methods, such as scrambling or 64b66b encoding, are common to SONET and SDH telecommunications systems. Scrambling and 64b66b encoding also work to balance the pattern and improve the BER to allow much larger runs of CIDs.

Diverse test patterns stress various aspects of system or component performance in applications. For example, a K28.5± pattern (11000001010011111010) helps assess the deterministic-jitter performance of systems that use 8b10b encoding. A PRBS (pseudorandom bit stream), on the other hand, is one of a family of general-purpose test patterns in encoded, random, and scrambled NRZ applications.

Common practice typically denotes pseudorandom patterns as $2^X-1$ PRBS. The power, X, indicates

From the NRZ pattern’s time-domain waveform (a), you can calculate its autocorrelation function (b) and, by application of the Fourier transform, the power spectrum (c).
the length of the shift register that creates the pattern. Each \(2^N - 1\) PRBS contains every possible combination of \(X\) number of bits, except one. Engineers testing Ethernet, Fibre Channel, or high-speed-video applications use a short sequence, such as the \(2^5 - 1\) PRBS (127 bits), because it provides a good approximation to an 8b10b-encoded NRZ data stream. A \(2^{23} - 1\) PRBS generates a pattern almost 8.4 Mbytes long and provides a good representation of scrambled or random NRZ data. Such long sequences provide the low-frequency content that SONET and SDH telecommunications-system tests require.

**Comparing the Power Spectrum**

Each NRZ test pattern has an associated PSD (power spectral density) that indicates the frequency distribution of the pattern’s signal power. The two primary methods of computing PSD are squaring the magnitude of the Fourier transform of the pattern or computing the Fourier transform of the autocorrelation function of the pattern ([Reference 1](#)). The first method is generally simpler for signals that you can mathematically write in a finite, closed form, as in \(s(t) = A \cos(\omega t)\). The second method computes the power spectrum for more complicated signals, such as long sequences of NRZ data or random bit streams. To apply these methods, review some of the basics of Fourier analysis ([Reference 2](#)):

- You can think of the delta function, \(\delta(t)\), as an infinitely narrow rectangular pulse with area \(A\). It has a nonzero value only when the argument of the function equals zero. By convention, a vertical arrow graphically represents the function.
- The comb function, \(A \sum_{n=-\infty}^{\infty} \delta(t - nT)\), consists of an infinite number of equal-area delta functions spaced at a uniform interval, \(T\).
- The Fourier transform of a comb function is also a comb function, in which the interval is \(n/T\) and the areas of the delta functions are \(A/T\).
- Convolution in the time domain, which \(\ast\) represents symbolically, is equivalent to multiplication in the frequency domain and vice versa.
- Convolution of a signal with a delta function produces a copy of the signal shifted to the location of the delta function.
- Multiplying a signal by a delta function—sampling—results in a delta function in which the magnitude of the signal at the delta function’s location modifies its area.

**Figure 1** shows an application of these rules, computing the PSD of an NRZ test pattern. A sequence of ones and zeros forms a pattern with a defined bit period, \(T_b\), and a total pattern length, \(L = nT_b\). Convolving the finite-length test pattern with a comb function that has a spacing interval equal to the pattern length results in an infinite repetition of the pattern ([Figure 1a](#)). Computing the autocorrelation functions for each component of the test pattern shows that the autocorrelation of the test pattern approximates a triangle ([Figure 1b](#)). The accuracy of this approximation improves as the length and randomness of the pattern increase. Finally, the Fourier transform of the autocorrelation functions reveals the power spectrum ([Figure 1c](#)).

The power spectrum resulting in this example shows an infinite sequence of discrete spectral lines—delta functions—scaled by a sinc\(\pi(f)\) envelope, where

\[
\sin \pi(f) \approx \frac{\Delta \sin(\pi f)}{(nf)}
\]

- The nulls in the sinc\(\pi(f)\) envelope occur at integer multiples of the data rate.
- Spectral lines are evenly spaced at an interval that is the inverse of the pattern length.
- The magnitude of the sinc\(\pi(f)\) envelope decreases as the data rate or pattern length increases. In the limit, as the pattern length approaches infinity, the spacing between the spectral lines becomes infinitesimal, and the spectrum shape approaches a continuous sinc\(\pi(f)\) function.

For example, you can calculate the spectral-line spacing, amplitude, and spectral nulls resulting from the 6-bit pattern that [Figure 1a](#) shows, assuming a transmission range of 1.25 Gbps ([Figure 2](#)). Note that the sinc\(\pi(f)\) envelope in [Figure 2](#) is an approximation of the 6-bit pattern’s PSD. The accuracy of this approximation improves as the pattern length or randomness increases.

**Spectral Measurements**

You can demonstrate these equations and principles on the bench using a high-speed pattern generator and a spectrum analyzer. A simple example is the spectrum resulting from transmitting a 4-bit pattern (1110) at 1.25 Gbps ([Figure 3](#)). The spectral nulls are at 1.25 GHz (1/T\(_b\)) and...
design feature NRZ spectra

2.5 GHz (2/Tb), and the line spacing is 312.5 GHz (1/L). The power-spectrum envelope follows sincπ(f).

The slight deviations in magnitude result from the short pattern.

Increasing the pattern length to 20 bits (K28.5± test pattern) and keeping the transmission rate at 1.25 Gbps result in spectral nulls in the same locations (1.25 and 2.5 GHz). The spectral-line spacing, however, reduces to 125 MHz due to the longer pattern length (Figure 4). The spectral-line envelope is also a more accurate representation of sincπ(f) than is the case with the 4-bit pattern.

The 20-bit K28.5± test pattern’s pattern spectral-line spacing measures 125 MHz when you transmit at 1.25 Gbps, which corresponds to a 10-bit test pattern. This discrepancy is due to the fact that the K28.5± pattern comprises a K28.5+ sequence (1100000101) and its inverse, the K28.5− sequence (0011111010). In the frequency domain, the K28.5− sequence contains the same spectral information as the K28.5+ sequence. The pattern, therefore, spectrally repeats every 10 bits resulting in the 125-MHz spacing.

The sincπ(f) envelope becomes more apparent as the pattern length increases further. A 2²−1 PRBS pattern (127 bits) at 2.5 Gbps illustrates this point (Figure 5). This longer pattern reduces the delta spacing to approximately 19.7 MHz. The spectral nulls appear at 2.5 and 5 GHz, corresponding to the higher data rate. Figure 6 illustrates the difference in the spectral-line magnitude and spacing for a 2²−1 PRBS pattern at 1.25 and 2.5 Gbps.

EXAMPLE APPLICATIONS

Knowledge of the power spectrum of NRZ test patterns can lead to significant improvements in digital-communications-system design. Three applications—receiver bandwidth, adaptive equalizers, and EMI—illustrate this concept.

Receiver bandwidth: The design process for a receiver inevitably includes questions about the necessary bandwidth. Too narrow a bandwidth attenuates the high-frequency components of the received signal, resulting in linear distortion. Too wide a bandwidth increases complexity and cost and admits excess noise, reducing SNR (Reference 3). Armed with knowledge of the spectral content of the signals that the receiver must process, you can make the bandwidth decision to admit the critical spectral components but no more.

Adaptive equalizer: Adaptive equalizers reverse linear distortion caused by nonideal transmission media. The MAX3800 adaptive cable equalizer, for example, reverses the distortion that skin-effect losses cause in copper cables at data rates as high as 3.2 Gbps (Reference 4). It accomplishes this task by comparing the power in the input signal at two discrete frequencies, 200 and 600 MHz. Based on the sincπ(T, f) envelope of the power spectrum with the first null at 3.2 GHz, the ratio of the power at these two frequencies should be:

\[
\frac{\sin c\pi f_1}{\sin c\pi f_2} = \frac{0.987}{0.890} = 1.11.
\]

When the equalizer measures a differ-
ent proportion, it changes its compensation to restore the correct ratio. This scenario works well for high data rates and long data patterns, but using your knowledge of NRZ-test-pattern spectral content, you can predict that some patterns may cause problems.

For example, if you reduce the data rate to 622 Mbps, the sinc$\pi^2(f)$ envelope with a first null at 622 MHz will result in a 200- to 600-MHz power-detector ratio of 0.703/0.00134=525 instead of the expected 1.11. As the equalizer tries to restore the expected power ratio, it may distort the output. Spectral lines for short patterns are spaced at larger intervals. In the case of, say, a 10-bit pattern at a data rate of 3.2 Gbps, 320 MHz separates the spectral lines, with the first few at 0, 320, and 640 MHz. For this type of pattern and data rate, there may be little or no power to detect at 200 or 600 MHz, which can cause signal distortion due to the equalizer's operating outside its specified range.

**EMI**: Altering the magnitude or frequencies of the signal-power spectrum can reduce or eliminate effects of EMI in a system. You can accomplish this task by changing the data rate or the pattern length.

As you increase the data rate, the nulls of the spectrum spread farther apart, attenuating the magnitude of each spectral line by pushing a fraction of the signal power to higher frequencies. Spreading the power over a larger frequency range leaves less at the frequencies of interest. One way to achieve this effect is by adding extra bits to the original data stream to effectively increase the data rate.

Pattern length also plays a role in EMI, because spectral-line magnitude and spacing vary as the pattern length changes. A longer pattern reduces the magnitude and spacing, and a shorter pattern increases them. To reduce EMI at a specific frequency, change the pattern length to shift the spectral line away from a sensitive frequency range, or use a longer pattern to reduce the magnitude of the EMI.

References


Author’s biographies

Justin Redd, PhD, serves as strategic-applications engineer at Maxim Integrated Products.

Craig Lyon, also at Maxim, is a customer-applications engineer.

Talk to us

Post comments via TalkBack at the online version of this article at www.edn.com.