A bit-error-rate (BER) tester is a basic tool for digital-communications measurements. Although many commercial BER testers are available, you can easily design and build an inexpensive version. The scheme in Figure 1 has performance similar to that of a commercial tester but requires you to perform a manual calculation based on displayed data. The tester displays received bits and received erroneous bits, and you must calculate the BER data using a handheld calculator, for example.

You can build the tester in Figure 1 from a piece of programmable logic, such as an FPGA or a CPLD, and two counter modules. You can buy the counter modules in kit form or in built form from numerous suppliers. The counters are available in LCD or LED-display formats with four or more digits. The counter modules must have overflow indicators, and they must allow pulse widths that are as narrow as half the data-clock period.

Figure 2 shows the core of the error detector. This detector uses the same pseudorandom-bit-sequence (PRBS) generator as the transmitter does but adds a trick. When the demodulator under test is not in lock, the shift register loads the receive data, and no error count takes place. In every demodulator, except for special burst-mode units, the BER decreases to some nominal rate before you declare the system "locked." Thus, the shift register is self-synchronized to the incoming sequence with great probability. When the demodulator is in lock, the Lock signal switches the multiplexer output so that locally generated data, which should be the same data as the transmitted data, shifts into the register while the measurement is running. Any divergence between the received data and the locally generated data constitutes a bit error. As long as the counter counts pulses, the error signal must combine with the clock in an RZ format so that the tester does not count two consecutive errors as only one.

An error that occurs just before the demodulator is in lock causes incorrect initialization of the shift register, and the local and received sequences is highly uncorrelated. Thus, the BER in this case is close to 0.5. You can easily detect this erroneous condition and restart the measurement.

The architecture in Figure 1 allows you to divide the number of errors and bits in the error-divider and bit-divider blocks. You can divide errors by 1, 10, 100 and 1000 and divide bits by 10⁴, 10⁵, 10⁶, and 10⁷. This division feature allows the tester to measure of a range of BERs from poor ones for which the tester must divide the error count to a low value to situations that require bit division or that entail long measurement periods. Two switches for each block can control the division rate in a simple way. To avoid mistakes, division control should also control a decimal point in the display, and an indication label under the displays should show the multiplication factor for the actual configuration.
Another feature is related to the overflow. When either counter unit overflows, the scheme in Figure 1 immediately stops the error count using the BIT_OFL and ERR_OFL flags so that the tester does not display erroneous data when taking unattended measurements or when taking measurements for long periods. When active, the BIT_OFL and ERR_OFL flags turn off the COUNT_ENABLE signal.

The Start, Stop, and Reset keys control the unit. They drive a finite state machine, which produces the variables C_ENABLE and C_RESET. The first variable controls the bit and error count, and the second controls the counter reset. (DI #2488)