The first script file below is used to test the Bert transmitter to receiver connection using a loopback cable. It will work with either the TTL or ECL Bert. You should have a 40Mhz TTL xtal in the transmitter socket. If you don't have a 40Mhz TTL xtal available, you can use other speeds or declare the clock source as PLL instead of XTAL. You must change the clock parameter in the script file to reflect the new clock speed.

For the TTL bert transmitter / receiver:

1. The transmitter's J3 "clock select" jumper must be set for XTAL if using an xtal or to PLL if not using an xtal.
2. The receiver's clock select jumpers must be set for "rising edge" and "registered mode". See figure 15 on page 8 of the Xbert manual.

For the ECL Bert transmitter / receiver:

1. The Bert receiver clock inversion jumpers (J4, J3) must be set to "inverted" (jumpers are perpendicular to the bottom of the Bert receiver).

logfile loopback.log *// Open Log file*
xrmask 3 *// Enable both transmitter and receiver*
chip xbert.xcf rbert.xcf pbert.xcf *// Load new FPGA files*
clock 40 xtal dav_hold async *// Set clock parameters*
burst 3 *// Max number of errors before bert reset*
new_flags *// Clear existing flag definitions*
flag C TRUN *// Define RUN flag*
flag 8 /TREADY *// Transmitter ready*
flag 0 TRESET *// Transmitter reset*
flag 1C RRUN *// Receiver RUN*
flag 18 /RREADY *// Receiver ready*
flag 10 RRESET *// Receiver Reset*
flag 16 /NULL
new_handshake *// Define a new handshake sequence*
handshake RRESET TRESET =
handshake = RREADY TREADY
mask ffffffff *// All 32 bits of data are tested*
davbit 0 *// DAV bit is at location 0 of the data*
offset 0 *// No offset*
pattern =seq *// Set the data pattern to sequential*
load *// load the pattern*
run *// Start the test*