```
--------------------------------------------------------------------------------------------------------------------------
--
-- CORSAIR TECHNOLOGY CONFIDENTIAL
--
--------------------------------------------------------------------------------------------------------------------------
-- (C) Copyright 2016-17 Corsair Technology Limited
-- All Rights Reserved
--
-- NOTICE: All information contained herein is, and remains the property of Corsair Technology Limited and its
-- suppliers, if any.  The intellectual and technical concepts contained herein are proprietary to Corsair Technology
-- Limited and its suppliers and may be covered by U.S. and Foreign Patents, patents in process, and are protected
-- by trade secret or copyright law. Dissemination of this information or reproduction of this material is strictly
-- forbidden unless prior written permission is obtained from Corsair Technology Limited.
--
-- File Name  : xxtea_encrypt_tb.vhd
--
-- Author     : E Lister
--
-- Version    : V00.01
--
--------------------------------------------------------------------------------------------------------------------------
--
-- Purpose :
--
--------------------------------------------------------------------------------------------------------------------------
-- (C) Copyright 2016-17 Corsair Technology Limited. All rights reserved
--------------------------------------------------------------------------------------------------------------------------
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

--------------------------------------------------------------------------------------------------------------------------
entity xxtea_encrypt_tb is
end entity xxtea_encrypt_tb ;

--------------------------------------------------------------------------------------------------------------------------
architecture TB_ARCHITECTURE of xxtea_encrypt_tb is

component xxtea_encrypt
```

```
  port
  (
    ------------------------------------------------------------------------------------------------------------------
    -- Inputs
    ------------------------------------------------------------------------------------------------------------------
    clk                         : in  std_logic ;
    en_encrypt                  : in  std_logic ;
    encryption_key              : in  unsigned(127 downto  0) ;
    counter                     : in  unsigned( 63 downto  0) ;
    ------------------------------------------------------------------------------------------------------------------
    -- Outputs
    ------------------------------------------------------------------------------------------------------------------
    cypher                      : out unsigned( 63 downto  0)
  ) ;
end component;

------------------------------------------------------------------------------------------------------------------
-- CONSTANTS
------------------------------------------------------------------------------------------------------------------
constant CLK_PERIOD             : time := 100 ns ;
------------------------------------------------------------------------------------------------------------------
-- SIGNALS
------------------------------------------------------------------------------------------------------------------
signal clk                      : std_logic                 := '0' ;
signal en_encrypt               : std_logic                 := '0' ;
signal encryption_key           : unsigned(127 downto  0) := (others => '0') ;
signal counter                  : unsigned( 63 downto  0) := (others => '0') ;
signal cypher                   : unsigned( 63 downto  0) := (others => '0') ;



------------------------------------------------------------------------------------------------------------------
begin

------------------------------------------------------------------------------------------------------------------
-- Create the system clock running at 10MHz.
------------------------------------------------------------------------------------------------------------------
system_clk : process is
begin
  clk                           <= '0' ;
```

```
  wait for CLK_PERIOD/2 ;
  clk                          <= '1' ;
  wait for CLK_PERIOD/2 ;
end process system_clk ;

------------------------------------------------------------------------------------------------------------------------
-- Create the enable signal after a delay of 5 system clock cycles
------------------------------------------------------------------------------------------------------------------------
enabler : process is
begin
  en_encrypt                   <= '0' ;
  wait for CLK_PERIOD*5 ;
  en_encrypt                   <= '1' ;
  wait for 30 us ;
  en_encrypt                   <= '0' ;
  wait for CLK_PERIOD*5 ;
  en_encrypt                   <= '1' ;
  wait for 30 us ;
  en_encrypt                   <= '0' ;
  wait for CLK_PERIOD*5 ;
  en_encrypt                   <= '1' ;
  wait for 30 us ;
  en_encrypt                   <= '0' ;
  wait for CLK_PERIOD*5 ;
  en_encrypt                   <= '1' ;
  wait for 30 us ;
  en_encrypt                   <= '0' ;
  wait ;
end process enabler ;

------------------------------------------------------------------------------------------------------------------------
--  Create a sequence of test vector pairs
------------------------------------------------------------------------------------------------------------------------
test_vector : process is
begin
  encryption_key               <= x"00000000000000000000000000000000" ;
  counter                      <= x"0000000000000000" ;
  wait for 30 us ;
  encryption_key               <= x"0102040810204080fffefcf8f0e0c080" ;
```

```
  counter                     <= x"0000000000000000" ;
  wait for 30 us ;
  encryption_key              <= x"9e3779b99b9773e9b979379e6b695156" ;
  counter                     <= x"ffffffffffffffff" ;
  wait for 30 us ;
  encryption_key              <= x"0102040810204080fffefcf8f0e0c080" ;
  counter                     <= x"fffefcf8f0e0c080" ;
  wait ;
end process test_vector ;

-------------------------------------------------------------------------------------------------------------------------
UUT : xxtea_encrypt
-------------------------------------------------------------------------------------------------------------------------
port map
  (
    clk                       => clk ,
    en_encrypt                => en_encrypt ,
    encryption_key            => encryption_key ,
    counter                   => counter ,
    cypher                    => cypher
  ) ;

end architecture TB_ARCHITECTURE;

-------------------------------------------------------------------------------------------------------------------------
-- Configure the testbench instances
-------------------------------------------------------------------------------------------------------------------------
configuration TESTBENCH_FOR_xxtea_encrypt of xxtea_encrypt_tb is
  for TB_ARCHITECTURE
    for UUT : xxtea_encrypt
    use entity work.xxtea_encrypt(rtl) ;
  end for ;
  end for ;
end configuration TESTBENCH_FOR_xxtea_encrypt ;

-------------------------------------------------------------------------------------------------------------------------
-- (C) Copyright 2016-17 Corsair Technology Limited. All rights reserved
-------------------------------------------------------------------------------------------------------------------------
```