CNT_EPP_TB

```
library ieee;
use ieee.std_logic_1164.all;

entity cnt_epp_tb is
end entity cnt_epp_tb;

architecture cnt_epp of cnt_epp_tb is

  component cnt_epp is
    port (
      CLK         : in std_logic;
      RST         : in std_logic;
      ASTRB : in std_logic;
      DSTRB : in std_logic;
      DATA       : inout std_logic_vector(7 downto 0);
      PWRITE: in std_logic;
      PWAIT : out std_logic;
      DATO_RD : in std_logic_vector(7 downto 0);
      CE_RD : out std_logic;
      DIR          : out std_logic_vector (7 downto 0);
      DIR_VLD : out std_logic;
      DATO       : out std_logic_vector (7 downto 0);
      DATO_VLD : out std_logic);
  end component;

  component epp_device is
  port (
    DATA   : inout std_logic_vector(7 downto 0);
    PWRITE : out   std_logic;
    DSTRB  : out   std_logic;
    ASTRB  : out   std_logic;
    PWAIT  : in    std_logic);
  end component;

  signal CLK_i    : std_logic := '0';
  signal RST_i    : std_logic := '1';
  signal ASTRB_i  : std_logic := '1';
  signal DSTRB_i  : std_logic := '1';
  signal DATA_io  : std_logic_vector(7 downto 0);
  signal PWRITE_i : std_logic := '1';
  signal PWAIT_o  : std_logic;
  signal DATO_RD_i: std_logic_vector(7 downto 0) := (others => '0');
  signal CE_RD_o  : std_logic;
  signal DIR_o    : std_logic_vector(7 downto 0) := (others => '0');
  signal DIR_VLD_o: std_logic;
  signal DATO_o   : std_logic_vector (7 downto 0);
  signal DATO_VLD_o: std_logic;

begin  -- architecture cnt_epp

  DUT: entity work.cnt_epp
```

```
    port map (
      CLK      => CLK_i,
      RST      => RST_i,
      ASTRB    => ASTRB_i,
      DSTRB    => DSTRB_i,
      DATA     => DATA_io,
      PWRITE   => PWRITE_i,
      PWAIT    => PWAIT_o,
      DATO_RD  => DATO_RD_i,
      CE_RD    => CE_RD_o,
      DIR      => DIR_o,
      DIR_VLD  => DIR_VLD_o,
      DATO     => DATO_o,
      DATO_VLD => DATO_VLD_o);

   epp: entity work.epp_device
    port map (
                      DATA   => DATA_io,
                            PWRITE => PWRITE_i,
                            DSTRB  => DSTRB_i,
                            ASTRB  => ASTRB_i,
                            PWAIT  => PWAIT_o);

  CLK_i <= not CLK_i after 5 ns;
  RST_i <= '1', '0' after 25 ns;
  DATO_RD_i <= x"33" after 6250 ns, x"00" after 7250 ns;

end architecture cnt_epp;
```