Kris M. Vallozo
BSCpE- 3A

STEPPER MOTOR INTERFACE

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity STEPPER_MOTOR_INTERFACE is
Port (
   clk      : in  STD_LOGIC;
   reset_n  : in  STD_LOGIC;
   enable_n : in  STD_LOGIC;
   dir_n    : in  STD_LOGIC;

   coil_n   : out STD_LOGIC_VECTOR(3 downto 0)
);
end STEPPER_MOTOR_INTERFACE;

architecture Behavioral of STEPPER_MOTOR_INTERFACE is
   signal step_counter : unsigned(1 downto 0) := "00";
   signal clk_div      : unsigned(20 downto 0) := (others => '0');
   signal step_clk     : STD_LOGIC := '0';
   signal enabled      : STD_LOGIC := '0';
   signal direction    : STD_LOGIC := '0';

   type step_sequence is array (0 to 3) of std_logic_vector(3 downto 0);
   constant full_step : step_sequence := (
      "1100",
      "0110",
      "0011",
      "1001"
   );
begin

   process(clk)
   begin
      if rising_edge(clk) then
         clk_div <= clk_div + 1;
         step_clk <= clk_div(20);
      end if;
   end process;

  process(step_clk, reset_n)
   begin
      if reset_n = '0' then
         step_counter <= "00";
```

```
            coil_n <= "1111";
        elsif rising_edge(step_clk) then
            if enabled = '1' then
                if direction = '1' then
                    step_counter <= step_counter + 1;
                else
                    step_counter <= step_counter - 1;
                end if;

                coil_n <= full_step(to_integer(step_counter));
            else
                coil_n <= "1111";
            end if;
        end if;
    end process;

    enabled <= not enable_n;
    direction <= not dir_n;

end Behavioral;
```