LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;

ENTITY speed_control IS
PORT (clock_100kHz : IN STD_LOGIC;
speed : IN STD_LOGIC_VECTOR ( 2 DOWNTO 0 );
speed_pwm : OUT STD_LOGIC);
END speed_control;

ARCHITECTURE servo_controller OF speed_control IS
SIGNAL count_motor: STD_LOGIC_VECTOR ( 10 DOWNTO 0 );
BEGIN
PROCESS
BEGIN
-- Count motor is a 20ms timer => [ 0 , 2000 ] based on a true 100kHz clock --
WAIT UNTIL clock_100kHz'EVENT AND clock_100kHz = '1';
IF count_motor /= 2000 THEN
  count_motor <= count_motor + 1;
ELSE
  count_motor <= "00000000000";
END IF;
-- Speed Setting #1 of 8 => BRAKE or OFF = 000 --
IF count_motor >= 1700 AND count_motor < 1838 THEN
  IF speed = "000" THEN
    speed_pwm <= '0';
  ELSE
    speed_pwm <= '1';
  END IF;
-- At this point the pulse is already 1 ms long --
-- Decide how long to allow pulse to remain high. [ 1ms , 2ms ] --
-- Speed Setting #2 of 8 => Quarter Reverse (CW) --
ELSIF count_motor >= 1838 AND count_motor < 1844 THEN
  CASE speed IS
    WHEN "010" =>
      speed_pwm <= '0';
    WHEN "001" =>
      speed_pwm <= '1';
    WHEN "011" =>
      speed_pwm <= '1';
    WHEN "100" =>
      speed_pwm <= '1';
    WHEN "101" =>
      speed_pwm <= '1';
    WHEN "110" =>
      speed_pwm <= '1';
    WHEN "111" =>
      speed_pwm <= '1';
    WHEN OTHERS => NULL;
  END CASE;
ELSE
  speed_pwm <= '0';
END IF;
-- Speed Setting #3 of 8 => Eighth Reverse (CW) --
ELSIF count_motor >= 1844 AND count_motor < 1850 THEN
--
IF speed /="000" THEN
  CASE speed IS
    WHEN "010" =>
      speed_pwm <= '0';
    WHEN "001" =>
      speed_pwm <= '0';
    WHEN "011" =>
      speed_pwm <= '1';
    WHEN "100" =>
      speed_pwm <= '1';
    WHEN "101" =>
      speed_pwm <= '1';
    WHEN "110" =>
      speed_pwm <= '1';
    WHEN "111" =>
      speed_pwm <= '1';
    WHEN OTHERS => NULL;
  END CASE;
ELSE
  speed_pwm <= '0';
END IF;

-- Speed Setting #4 of 8 => Neutral (1.5 ms pulse) --
ELSIF count_motor >= 1850 AND count_motor < 1856 THEN
  IF speed /="000" THEN
    CASE speed IS
      WHEN "010" =>
        speed_pwm <= '0';
      WHEN "001" =>
        speed_pwm <= '0';
      WHEN "011" =>
        speed_pwm <= '0';
      WHEN "100" =>
        speed_pwm <= '1';
      WHEN "101" =>
        speed_pwm <= '1';
      WHEN "110" =>
        speed_pwm <= '1';
      WHEN "111" =>
        speed_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
  ELSE
    speed_pwm <= '0';
  END IF;

-- Speed Setting #5 of 8 => Eighth Forward (CCW) --
ELSIF count_motor >= 1856 AND count_motor < 1863 THEN
  IF speed /="000" THEN
    CASE speed IS
      WHEN "010" =>
        speed_pwm <= '0';
      WHEN "001" =>
        speed_pwm <= '0';
      WHEN "011" =>
        speed_pwm <= '0';
      WHEN "100" =>
        speed_pwm <= '0';
      WHEN "101" =>
        speed_pwm <= '1';
      WHEN "110" =>
        speed_pwm <= '1';
      WHEN "111" =>
        speed_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
  ELSE
    speed_pwm <= '0';
  END IF;

-- Speed Setting #5 of 8 => Eighth Forward (CCW) --
ELSIF count_motor >= 1856 AND count_motor < 1863 THEN
  IF speed /="000" THEN
    CASE speed IS
      WHEN "010" =>
        speed_pwm <= '0';
      WHEN "001" =>
        speed_pwm <= '0';
      WHEN "011" =>
        speed_pwm <= '0';
      WHEN "100" =>
        speed_pwm <= '0';
      WHEN "101" =>
        speed_pwm <= '1';
      WHEN "110" =>
        speed_pwm <= '1';
      WHEN "111" =>
        speed_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
  ELSE
    speed_pwm <= '0';
  END IF;

ELSE

END FILE;
speed_pwm <= '0';
END IF;

-- Speed Setting #6 of 8 => Quarter Forward (CCW) --
ELSIF count_motor >= 1863 AND count_motor < 1875 THEN
  IF speed /= "000" THEN
    CASE speed IS
      WHEN "010" =>
        speed_pwm <= '0';
      WHEN "001" =>
        speed_pwm <= '0';
      WHEN "011" =>
        speed_pwm <= '0';
      WHEN "100" =>
        speed_pwm <= '0';
      WHEN "101" =>
        speed_pwm <= '0';
      WHEN "110" =>
        speed_pwm <= '0';
      WHEN "111" =>
        speed_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
  ELSE
    speed_pwm <= '0';
  END IF;
END PROCESS;

-- All of the case statements with <=1 and reoccurring <=0 could be deleted.--
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;

ENTITY direction_control IS
  PORT (clock_100kHz : IN STD_LOGIC;
        position : IN STD_LOGIC_VECTOR ( 2 DOWNTO 0 );
        direction_pwm : OUT STD_LOGIC);
END direction_control;

ARCHITECTURE servo_controller OF direction_control IS
  SIGNAL count_motor: STD_LOGIC_VECTOR ( 10 DOWNTO 0 );
BEGIN
  PROCESS
    BEGIN
      -- Count_motor is a 20ms timer => [ 0 , 2000 ] --
      WAIT UNTIL clock_100kHz'EVENT AND clock_100kHz = '1';
      IF count_motor /= 2000 THEN
        count_motor <= count_motor + 1;
      ELSE
        count_motor <= "00000000000";
      END IF;

      -- Direction Setting #1 of 8 => SIGNAL OFF = 000 --
      IF count_motor >= 1700 AND count_motor < 1818 THEN
        IF position = "000" THEN
          direction_pwm <= '0';
        ELSE
          direction_pwm <= '1';
        END IF;
      END IF;

      -- At this point the pulse is already 1 ms long
      -- Decide how long to allow pulse to remain high. [ 1ms , 2ms ] --

      -- Direction Setting #2 of 8 => 45 degrees LEFT or CW from center (1830) --
      ELSIF count_motor >= 1818 AND count_motor < 1830 THEN
        CASE position IS
          WHEN "011" =>
            direction_pwm <= '0';
          WHEN "010" =>
            direction_pwm <= '1';
          WHEN "001" =>
            direction_pwm <= '1';
          WHEN "100" =>
            direction_pwm <= '1';
          WHEN "101" =>
            direction_pwm <= '1';
          WHEN "110" =>
            direction_pwm <= '1';
          WHEN "111" =>
            direction_pwm <= '1';
          WHEN OTHERS => NULL;
          END CASE;
        ELSE
          direction_pwm <= '0';
        END IF;

      -- Direction Setting #3 of 8 => 20 degrees LEFT or CW from center (1845) --
      ...
-- After trial and error the usable value was determined to be 1830 --
ELSIF count_motor >= 1830 AND count_motor < 1838 THEN
  IF position /="000" THEN
    CASE position IS
      WHEN "011" =>
        direction_pwm <= '0';
      WHEN "010" =>
        direction_pwm <= '0';
      WHEN "001" =>
        direction_pwm <= '1';
      WHEN "100" =>
        direction_pwm <= '1';
      WHEN "101" =>
        direction_pwm <= '1';
      WHEN "110" =>
        direction_pwm <= '1';
      WHEN "111" =>
        direction_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
  ELSE
    direction_pwm <= '0';
  END IF;
END IF;

-- Direction Setting #4 of 8 => 10 degrees LEFT or CW from center (1850) --
-- After trial and error the usable value was determined to be 1838 --
ELSIF count_motor >= 1838 AND count_motor < 1845 THEN
  IF position /="000" THEN
    CASE position IS
      WHEN "011" =>
        direction_pwm <= '0';
      WHEN "010" =>
        direction_pwm <= '0';
      WHEN "001" =>
        direction_pwm <= '0';
      WHEN "100" =>
        direction_pwm <= '1';
      WHEN "101" =>
        direction_pwm <= '1';
      WHEN "110" =>
        direction_pwm <= '1';
      WHEN "111" =>
        direction_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
  ELSE
    direction_pwm <= '0';
  END IF;
END IF;

-- Direction Setting #5 of 8 => 0 degrees or center (1856) --
-- After trial and error the usable value was determined to be 1845 --
ELSIF count_motor >= 1845 AND count_motor < 1862 THEN
  IF position /="000" THEN
    CASE position IS
      WHEN "011" =>
        direction_pwm <= '0';
      WHEN "010" =>
        direction_pwm <= '0';
      WHEN "001" =>
        direction_pwm <= '0';
      WHEN "100" =>
        direction_pwm <= '0';
      WHEN "101" =>
        direction_pwm <= '1';
      WHEN "110" =>
        direction_pwm <= '1';
      WHEN "111" =>
        direction_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
  ELSE
    direction_pwm <= '0';
  END IF;
END IF;
WHEN "111" =>
direction_pwm <= '1';
WHEN OTHERS => NULL;
END CASE;
ELSE
direction_pwm <= '0';
END IF;

-- Direction Setting #6 of 8 => 10 degrees RIGHT or CCW from center (1862) --
ELSIF count_motor >= 1862 AND count_motor < 1868 THEN
  IF position /="000" THEN
    CASE position IS
      WHEN "011" =>
        direction_pwm <= '0';
      WHEN "010" =>
        direction_pwm <= '0';
      WHEN "001" =>
        direction_pwm <= '0';
      WHEN "100" =>
        direction_pwm <= '0';
      WHEN "101" =>
        direction_pwm <= '0';
      WHEN "110" =>
        direction_pwm <= '0';
      WHEN "111" =>
        direction_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
    ELSE
      direction_pwm <= '0';
    END IF;
  ELSE
    direction_pwm <= '0';
  END IF;
-- Direction Setting #7 of 8 => 20 degrees CCW from center (1868) --
ELSIF count_motor >= 1868 AND count_motor < 1882 THEN
  IF position /="000" THEN
    CASE position IS
      WHEN "011" =>
        direction_pwm <= '0';
      WHEN "010" =>
        direction_pwm <= '0';
      WHEN "001" =>
        direction_pwm <= '0';
      WHEN "100" =>
        direction_pwm <= '0';
      WHEN "101" =>
        direction_pwm <= '0';
      WHEN "110" =>
        direction_pwm <= '0';
      WHEN "111" =>
        direction_pwm <= '1';
      WHEN OTHERS => NULL;
    END CASE;
    ELSE
      direction_pwm <= '0';
    END IF;
  ELSE
    direction_pwm <= '0';
  END IF;
-- Direction Setting #8 of 8 => 45 degrees CCW from center (1882) --
ELSE
  direction_pwm <= '0';
END IF;

END PROCESS;
END servo_controller ;

--All of the case statements with <=1 and reoccurring <=0 could be deleted.--