PLAZA, ELMO L.
BSCPE 3A

VHDL CODE FOR JK FLIP FLOP WITH ASYNCHRONOUS RESET: (Master Slave JK Flip-Flop)

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

entity JK_Flipflop is
   Port (
      j     : in  STD_LOGIC;
      k     : in  STD_LOGIC;
      clk   : in  STD_LOGIC;
      reset  : in  STD_LOGIC;   -- Active-low reset
      Q     : out STD_LOGIC
   );
end JK_Flipflop;

architecture Behavioral of JK_Flipflop is
   signal div   : unsigned(22 downto 0) := (others => '0');
   signal clkd  : STD_LOGIC := '0';
   signal q_int : STD_LOGIC := '0';
begin

   -- Clock divider: divide clk down to a slower clkd
   process(clk)
   begin
      if rising_edge(clk) then
         div <= div + 1;
      end if;
   end process;

   clkd <= div(22); -- Slow clock

   -- JK Flip-Flop with active-low reset and slower clock
   process(clkd, reset)
   begin
      if reset = '0' then      -- Active-low reset
         q_int <= '0';
      elsif rising_edge(clkd) then
         if (j = '0' and k = '0') then
            -- No change
            null;
         elsif (j = '0' and k = '1') then
            q_int <= '0';    -- Reset
         elsif (j = '1' and k = '0') then
            q_int <= '1';    -- Set
```

```
        elsif (j = '1' and k = '1') then
            q_int <= not q_int;  -- Toggle
        end if;
    end if;
end process;

Q <= q_int;

end Behavioral;
```