-- MAQUINA DE MOORE

--detecta 11

library ieee;

use ieee.std\_logic\_1164.all;

entity MOORE is

port(

clk : in std\_logic;

W : in std\_logic;

reset : in std\_logic;

Z : out std\_logic

);

end entity;

architecture rtl of MOORE is

-- Build an enumerated type for the Q machine

type state\_type is (A, B, C);

-- Register to hold the current Q

signal Q : state\_type;

begin

-- Logic to advance to the next Q

process (clk, reset)

begin

if reset = '0' then

Q <= A;

elsif (rising\_edge(clk)) then

case Q is

when A=>

if W = '1' then

Q <= B;

else

Q <= A;

end if;

when B=>

if W = '1' then

Q <= C;

else

Q <= A;

end if;

when C=>

if W = '1' then

Q <= C;

else

Q <= A;

end if;

end case;--------------------

end if;

end process;

-- Z depends solely on the current Q

process (Q)

begin

case Q is

when A =>

Z <= '0';

when B =>

Z <= '0';

when C =>

Z <= '1';

end case;

end process;

end rtl;