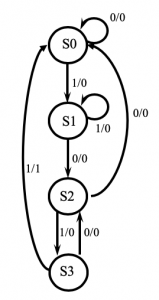
**EXPERIMENT – 9**

**AIM:** WRITE VHDL CODE FOR DETECTING NON-OVERLAPPING 1101 SEQUENCE IN

MEALY MACHINE TYPE AND SIMULATE IT USING MODELSIM.

**THEORY**: A **Mealy machine** is a finite-state machine whose output values are determined both by its current state and the current inputs. This is in contrast to a Moore machine, whose (Moore) output values are determined solely by its current state. A Mealy machine is a deterministic finite-state transducer: for each state and input, at most one transition is possible.



**VHDL CODE:**

LIBRARY ieee;

USE ieee.std\_logic\_1164.all;

USE ieee.std\_logic\_arith.all;

ENTITY SEQ\_DET\_NO1011 IS

port(CLOCK ,seq : in std\_logic; det\_vld :out std\_logic := '0');

END ENTITY SEQ\_DET\_NO1011;

--

ARCHITECTURE SEQ\_DET\_NO1011\_BEHAV OF SEQ\_DET\_NO1011 IS

type state\_type is(A,B,C,D); --Defines the type for states in the state machine

signal state:state\_type:= A;--Declare the signal with the type corresponding state type

BEGIN

process (CLOCK)

begin

if(rising\_edge(CLOCK)) then --calculates the next statebased on current state and input bit.

case state is

when A=> --when the current state is A.

det\_vld<= '0';

if( seq= '0')then

state<= A;

else

state<= B;

end if;

when B=>--when the current state is B.

if( seq= '0')then

state<= C;

else

state<= B;

end if;

when C=>--when the current state is C.

if( seq= '0')then

state<= A;

else

state<= D;

end if;

when D=>--when the current state is D.

if( seq= '0')then

state<= C;

else

state<= A;

det\_vld<= '1'; --Output is asserted when thepattern "1011" is found in the sequence.

end if;

when others=> NULL;

end case;

end if;

end process;

END ARCHITECTURE SEQ\_DET\_NO1011\_BEHAV;

**RESULT:**

