Drexel
UNIVERSITY
Electrical and Computer Engineering

Counting Circuit Using Arithmetical Packages

By Prawat Nagvajara

Synopsis

This note covers a synchronous circuit for accumulating the number ones appeared at the input after a reset.

Code

The code uses a function “+” from the std_logic_arith package.

```
--------------------------------------
-- Company: Drexel ECE
-- Engineer: Pravat
-- Description: A synchronouse circuit
-- counting the number of ones appeared
-- at the input after a reset
--------------------------------------
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity count_ones is
generic (n : natural := 4);
port (x, ck, reset: in std_logic;
    z: out std_logic_vector(n-1 downto 0));
end count_ones;

architecture beh of count_ones is
signal count : std_logic_vector(n-1 downto 0);
begin
z <= count;
process(ck)
begin
if ck='1' and ck'event then
 if reset = '1' then
    count <= (others => '0');
 else
 if x = '1' then count <= count + 1; end if;
 end if;
end if;
end process;
end beh;
```

Verification

After a reset the circuit accumulates the number of ones appeared at the input (Fig. 1). The addition of n-bit vector and unsigned integer is modulo 2^n. The number of bits n is default to 4. The addition modulo 2^n manifests when the signal count in Fig. 2 transitions from “1111” to “0000” (the last two cycles in Fig. 2).

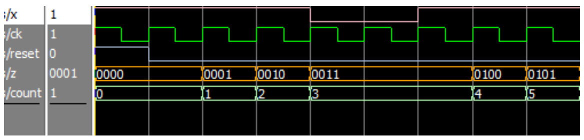

Fig.1 Simulation Wave from Reset

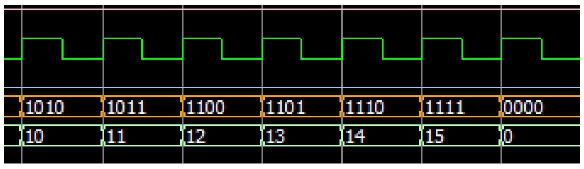

Fig.2 Simulation Wave Continued