Electrical and Computer Engineering

Finding Maximum Number Using Array Structure

By Prawat Nagvajara

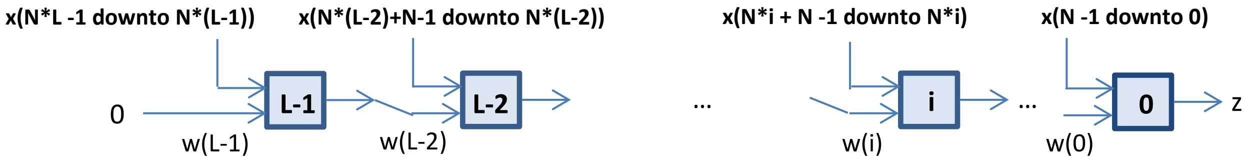


Fig. 1 Maximum Number Finder Array (max_array) Comprises Processing Elements

Synopsis

An implementation on FPGA - an array structure that outputs the maximum N-bit unsigned number out of L input n-bit unsigned numbers.

Figure 1 shows a schematic where L numbers are the N-bit chunks of the inputs x : in std_logic_vector(N*L – 1 downto 0); and the out z: out std_logic_vector(N-1 downto 0) is the maximum.

The Processing Element – PE each compares the two inputs x and y, and multiplexes out the greater, that is, if x > y then z <= x; else z <= y; end if; (Fig. 2). The number zero, 0 (an input to the leftmost PE in Fig. 1), is a vector of all ‘0’. The VHDL syntax is (others => ‘0’). The array structure propagates the maximum out to the output z.

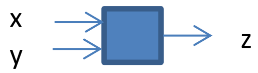


Fig. 2 Processing Element

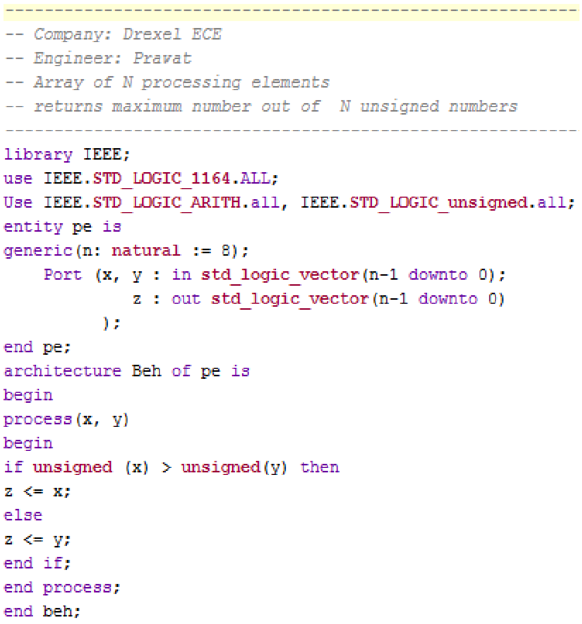
Algorithm

The array implements the following algorithm. The max_unsigned function takes as the inputs N*L –bit vector where x(N*i +N-1 downto n*i), 0≤ i ≤ L-1 is the ith N-bit unsigned, and outputs the maximum.

```
------------------------------------
-- max_usigned function
-- Input x: L N-bit unsigned
-- Initialize temp to 0, iteratively
-- compare each number to temp and
-- update temp as a current maximum
------------------------------------
function max_unsigned
(x:std_logic_vector; L, N: natural)
return std_logic_vector is
variable temp:
  std_logic_vector(N-1 downto 0)
  := (others =>'0');
begin
  for i in L-1 downto 0 loop
    if unsigned (x(N*i +N-1 downto N*i))
    > unsigned (temp) then
       temp :=  x(N*i +N-1 downto N*i);
    end if;
  end loop;
return temp;
end function max_unsigned;
```

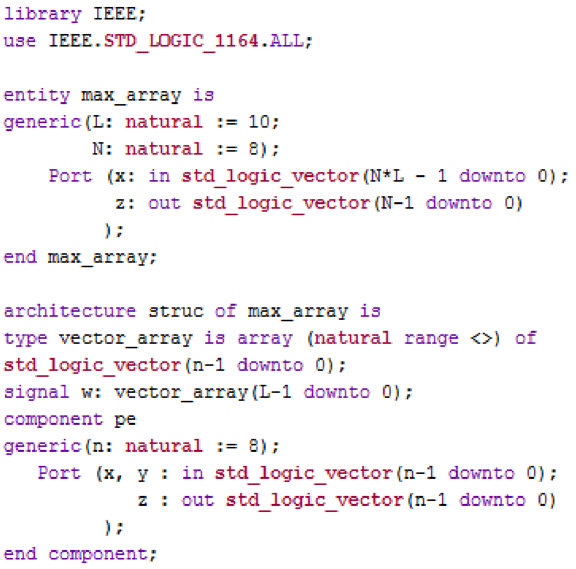
Code

VHDL code of the PE (below) uses the IEEE std_logic_arith and the std_logic_unsigned packages to output the maximum number. The inputs x and y are cast to unsigned. If the problem were signed numbers cast x and y to signed .



```
------------------------------------------------------------
-- Company: Drexel ECE
-- Engineer: Pravat
-- Array of N processing elements
-- returns maximum number out of  N unsigned numbers
------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
Use IEEE.STD_LOGIC_ARITH.all, IEEE.STD_LOGIC_unsigned.all;
entity pe is
generic(n: natural := 8);
    Port (x, y : in std_logic_vector(n-1 downto 0);
             z : out std_logic_vector(n-1 downto 0)
          );
end pe;
architecture Beh of pe is
begin
process(x, y)
begin
if unsigned (x) > unsigned(y) then
z <= x;
else
z <= y;
end if;
end process;
end beh;
```

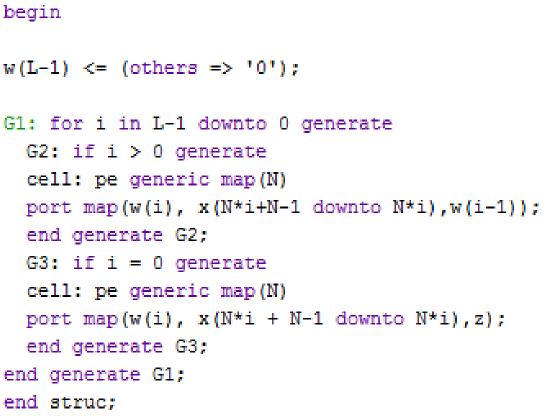
The max_array entity below consists of the processing elements connected by the signal w(i), i = L-1, …, 0. (Fig. 1) and. w(L-1) is assigned to 0, i.e., (others => ‘0’).



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

entity max_array is
generic(L: natural := 10;
        N: natural := 8);
    Port (x: in std_logic_vector(N*L - 1 downto 0);
           z: out std_logic_vector(N-1 downto 0)
          );
end max_array;

architecture struc of max_array is
type vector_array is array (natural range <>) of
std_logic_vector(n-1 downto 0);
signal w: vector_array(L-1 downto 0);
component pe
generic(n: natural := 8);
   Port (x, y : in std_logic_vector(n-1 downto 0);
            z : out std_logic_vector(n-1 downto 0)
          );
end component;
```

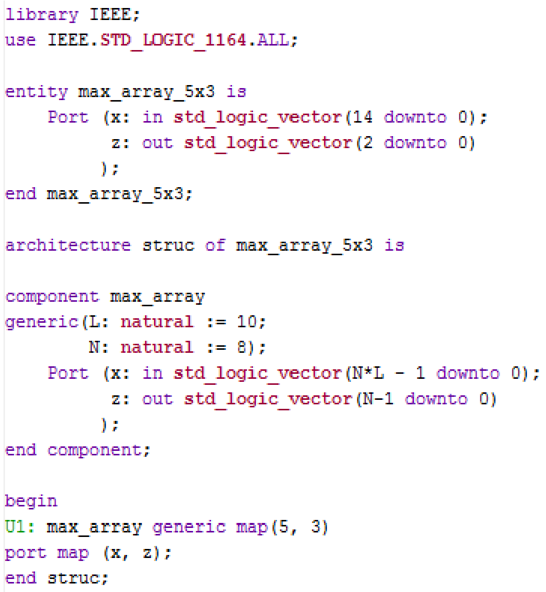


```
begin

w(L-1) <= (others => '0');

G1: for i in L-1 downto 0 generate
  G2: if i > 0 generate
  cell: pe generic map(N)
  port map(w(i), x(N*i+N-1 downto N*i),w(i-1));
  end generate G2;
  G3: if i = 0 generate
  cell: pe generic map(N)
  port map(w(i), x(N*i + N-1 downto N*i),z);
  end generate G3;
end generate G1;
end struc;
```

An instance of max_array with L = 5 numbers and N = 3, 3-bit number ranges from 0 to 7. The input vector x is L*N = 15 bits makes up the five 3-bits chunks.



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

entity max_array_5x3 is
    Port (x: in std_logic_vector(14 downto 0);
           z: out std_logic_vector(2 downto 0)
          );
end max_array_5x3;

architecture struc of max_array_5x3 is

component max_array
generic(L: natural := 10;
        N: natural := 8);
    Port (x: in std_logic_vector(N*L - 1 downto 0);
           z: out std_logic_vector(N-1 downto 0)
          );
end component;

begin
U1: max_array generic map(5, 3)
port map (x, z);
end struc;
```

Exercises

Reader can experiment with a signed maximum finder, where the smallest N-bit number is -2N-1, i.e., (2**(N-1) => ‘1’, others => ‘0’) in VHDL syntax.

Reader can design an array sorter using L rows of a modified max_array whose processing elements sort the maximum and minimum of the two inputs. The number l inputs of the ith row reduces, l = L-i, i=0, …, L-1, as each row finds the maximum. The L outputs are the sorted list of the L numbers.