

VHDL Synthesis

The IEEE Std. 1076.3 (summary)

Bert Molenkamp
Department of Electrical Engineering, Mathematics and Computer Science
University of Twente
PO Box 217
7500 AE Enschede
the Netherlands
email: e.molenkamp@utwente.nl



Interpretation of vector

Given the variables:

```
v1 : std_logic_vector(1 DOWNTO 0);  
v2 : std_logic_vector(2 DOWNTO 0);  
res : std_logic_vector(2 DOWNTO 0);
```

```
v1 := "11";  
v2 := "000";
```

```
res := v1 + v2;
```



IEEE "synthesis packages"

- An IEEE standard Std 1076.3 - 1997
- Two similar packages:
 - **numeric_bit:** vector is based on type bit
use ieee.numeric_bit.all;
 - **numeric_std:** vector is based on type std_(u)logic
use ieee.numeric_std.all;
- Two types are declared in these packages:
 - **signed:** twos complement representation
type SIGNED is array (NATURAL range <>) of STD_LOGIC;
 - **unsigned:** unsigned representation
type UNSIGNED is array (NATURAL range <>) of STD_LOGIC;

Use at most one of these packages in a design unit.



© 1996-2011 E. Molenkamp, University of Twente, the Netherlands
1076.3

3

The subprograms in numeric_std and numeric_bit

- Arithmetic
- Comparison
- Shift/rotate
- Resize
- Conversion
- Std_match



© 1996-2011 E. Molenkamp, University of Twente, the Netherlands
1076.3

4

Arithmetic

Type of operands	signed	unsigned	natural	integer
signed	Std. 1076.3	not supported	Std. 1076.3	Std. 1076.3
unsigned	not supported	Std. 1076.3	Std. 1076.3	not supported
natural	Std. 1076.3	Std. 1076.3	Std. 1076	Std. 1076
integer	Std. 1076.3	not supported	Std. 1076	Std. 1076

unary: abs, - (both only for type signed)

binary: +, -, *, /, rem, mod



Arithmetic /3

Range of the result of addition and subtraction

	R	signed	integer
L			
signed		max(L'length,R'length)-1 downto 0	L'length-1 downto 0
integer		R'length-1 downto 0	Std. 1076-1987/1993

Range of the result of multiplication

	R	signed	integer
L			
signed		L'length + R'length-1 downto 0	2×L'length-1 downto 0
integer		2×R'length-1 downto 0	Std. 1076-1987/1993



Arithmetic /6

Range of the result of division, mod, rem

	R	signed	integer
L			
signed		L'length -1 downto 0	L'length-1 downto 0
integer		R'length-1 downto 0	Std. 1076-1987/1993



functions *Shift / Rotate*

- ⇒ **shift_left (vec, pos);**
 - vec is shifted 'pos' to the left. Vacated positions filled with '0'.
 - pos should be ≥ 0 .
- ⇒ **shift_right (vec, pos);**
 - vec is shifted 'pos' to the right. Vacated positions filled with:
 - '0' for unsigned vec
 - vec(vec'left) for signed vec
 - pos should be ≥ 0 .
- ⇒ **rotate_left (vec, pos); rotate_right (vec, pos);**
 - rotate_left or rotate_right over 'pos' positions.
 - pos should be ≥ 0 .

Also overloaded the predefined functions sll, srl, rol, ror (std. 1076-1993). In these functions "pos" may also be negative.



Function *Resize*

⇒ **Resize (vec, length);**

- **result is a vector of the specified length.**
- **unsigned:**
if length > vec'length then left most bit positions filled with '0'.
otherwise left most bits are dropped.
- **signed:**
if length > vec'length then left most bit positions filled with vec(vec'left).
otherwise left most bits are dropped BUT the sign bit is retained.

sign	a	b	c	d	e	f	g
------	---	---	---	---	---	---	---

resize (vec, 5), with vec of type signed

sign	d	e	f	g
------	---	---	---	---



© 1996-2011 E. Molenkamp, University of Twente, the Netherlands
1076.3

9

Conversion functions

⇒ **to_integer(vec);**

- **returns the integer value of 'vec'. Two overloaded functions for vec of type signed and unsigned.**
- **if 'vec' contains other values then '0','1','L','H' then**
the result is 0, and
a warning is raised

⇒ **to_unsigned(value,length);**

- **returns an unsigned vector of the specified length.**
- **if value is out of range a warning raised.**

⇒ **to_signed(value,length);**

- **returns a signed vector of the specified length.**
- **if value is out of range a warning raised.**



© 1996-2011 E. Molenkamp, University of Twente, the Netherlands
1076.3

10

Omission in the package std_logic_1164

```
SIGNAL a : std_logic_vector(0 TO 2);  
..  
IF a = "1-1" THEN ..
```

equal?

```
SIGNAL a : std_logic_vector(0 TO 2);  
..  
IF a(0)='1' AND a(2)='1' THEN ..
```



Std_match

- ➔ Numeric_std contains a functions std_match, input of type:

- std_ulogic
- std_logic
- unsigned
- signed
- std_logic_vector
- std_ulogic_vector

```
SIGNAL a : std_logic_vector(0 TO 2);  
..  
IF std_match(a, "1-1") THEN ..
```

- ➔ Functions returns TRUE only if:
- both values are logical values ('0', '1', 'L', 'H') and the same.
 - one value is '0' and the other is 'L'
 - one value is '1' and the other is 'H'
 - at least one of the values is '-'
- ➔ If the two arguments are vectors:
- the operands should be of the same length, and
 - the function is applied on each element

