Lesson 6

Literals and Operators

Mohd Saufy Rohmad
Technical Trainer and Consultant

Literals

- You probably already know different numbering systems like binary, hexadecimal, octal etc.
- In this section, we will discuss how to express them in VHDL.
- It is possible to declare binary or hexadecimal numbers in VHDL in the following way,
- Binary: B"11000100"; Hexadecimal: X"FF";
- Or in the following way,
- Binary: 2#11000100#;
- Hexadecimal: 16#FF#;
- For readability its possible to use _ in betweeen. It will have no effect, for example,
- Binary: B"1100_0100";

Literals

- You can also declare integer or real data types, like,
- constant FREEZE : integer := 32; constant TEMP : real := 32.0;
- However, the real and integers are tricky to use as you dont know how they will be translated into a hardware.
- It is very important to know about different number systems and fixed point data types. Please go through the lectures provided in the link,
- https://sites.google.com/site/shahriarshahabuddin/teaching#TOC-Lecture:
- The lecture 2 about fixed point implementation will give you an idea about different data formats and how they can be applied for digital designs.

Operator

- Operators are means for constructing expressions, a list of basic operators are given here:

Operators	Function
**	exponentiation
abs	absolute value
not	complement
*	multiplication
/	division
mod	modulo
rem	remainder
+	unary plus
-	unary minus
+	addition
-	subtraction
&	concatenation
sll	shift left logical
srl	shift right logical
sla	shift left arithmetic

sra	shift right arithmetic
rol	rotate left
ror	rotate right
=	test for equality, result is boolean
/=	test for inequality, result is boolean
<	test for less than, result is boolean
<=	test for less than or equal, result is boolean
>	test for greater than, result is boolean
>=	test for greater than or equal, result is boolean
and	logical and
or	logical or
nand	logical complement of and
nor	logical complement of or
xor	logical exclusive or
xnor	logical complement of exclusive or

Thank You