**WIC2003: Tutorial 02**

**NOR FAKHIRA BT YUSOF**

**U2000780**

1. List the declaration and body unit design unit.

**- Entities**

**- Architecture**

**- Packages**

**- Package bodies**

**- Configurations**

1. Why is a design entity separated into an entity declaration and an architecture body?

**The entity defines the interface, the architecture defines the function. The entity**

**declaration names the entity and defines the interface to its environment. The**

**architecture body describes the operation of the component. The entity\_name in**

**the architecture statement must be the same as the entity declaration that**

**describes the interface to the outside world.**

1. What information does an entity declaration provide about a design entity?

**The architecture body provides an "internal" view; it describes the behavior or the structure of the component.**

1. Which of the following can be used as a valid identifier in a VHDL program and which cannot? For those that cannot, explain why.
   * 1. Adder

**valid**

* + 1. result\_

**valid**

* + 1. units

**not valid because it is reserved word**

* + 1. 4\_to\_1\_mux

**valid**

* + 1. top\_level

**valid**

* + 1. port

**not valid because it is reserved word**

1. For each port mode, list the mode and the possible direction(s) of information transfer. Specify these directions as in, out, or in and out. Which port mode is not acceptable for synthesis?

**- In: Data comes into this port and can only be read within the entity. It can**

**appear only on the right side of a signal or variable assignment.**

**- Out: The value of an output port can only be updated within the entity. It**

**cannot be read. It can only appear on the left side of a signal assignment.**

**- Inout: The value of a bi-directional port can be read and updated within the**

**entity model. It can appear on both sides of a signal assignment.**

**- Buffer: Used for a signal that is an output from an entity. The value of the**

**signal can be used inside the entity, which means that in an assignment**

**statement the signal can appear on the left and right sides of the <=**

**operator. Not recommended to be used in the synthesizable code.**

**Buffer is not acceptable for synthesis**

1. In each of the following concurrent signal assignment statements, indicate which signals are written, which are read, and to which signals each statement is sensitive.
   * 1. f <= a **and** b **and** c;

**Statement (a) is sensitive to signals a, b and c, which are the signals that are read. Signal f is written.**

* + 1. g <= **not** (w **and** x **and** y);

**Statement (b) is sensitive to signals w, x and y, which are the signals that are read. Signal f is written.**

* + 1. eq <=(**not** x **and not** y) or (x **and** y);

**Statement (c) is sensitive to signals x and y, which are the signals that are read. Signal eq is written.**

* + 1. eq <= (x\_bar **and** y\_bar) **or** (x **and** y);

**Statement (d) is sensitive to signals x\_bar, y\_bar, x and y, which are the signals that are read. Signal eq is written.**

* + 1. alarm <= armed **and** (d1 **or** d2 **or** d3) **or** panic;

**Statement (d) is sensitive to signals armed, d1,d2,d3 and panic, which are the signals that are read. Signal alarm is written.**

1. Given the following architecture body, write the associated entity declaration:

architecture csa of and\_or is signal s1, s2: std\_logic ;

begin

s1 <= a and b; s2 <= a and c; f<= s1 or s2;

end csa;

**entity and\_or is**

**port(a,b: in std\_logic; f: out std\_logic);**

**end and\_or;**

1. What is the difference in operation of a port of mode inout and one of mode buffer?

|  |  |
| --- | --- |
| **Inout** | **Buffer** |
| **The value of a bi-directional port can be read and updated within the entity model. It can appear on both sides of a signal assignment.** | **Used for a signal that is an output from an entity. The value of the signal can be used inside the entity, which means that in an assignment statement the signal can appear on the left and right sides of the <= operator. Not recommended to be used in the synthesizable code.** |

1. What information about a design entity does an architecture body provide?

**The architecture body represents the internal description of the design entity its behaviour, its structure, or a mixture of both.**

1. Into what two parts is an architecture body divided? What is the purpose of each part?

**- a declarative part**

**- a body of statements**

**A name declared in a declarative part is visible from the end of the declaration**

**itself down to the end of the corresponding statement part. Within this area we**

**can refer to the declared name. Before the declaration, within it and beyond the**

**end of the statement part, we cannot refer to the name because it is not visible.**

1. What are the three different pure coding styles for an architecture body and how is each distinguished?

**- A dataflow architecture uses only concurrent signal assignment statements.**

**- A behavioral architecture uses only process statements.**

**- A structural architecture uses only component instantiation statements.**