Configuration Declaration

- It is used to select one of the possibly many architecture bodies.
- Used to bind components.
- Used to represent structure in that architecture body.

Example of Configuration Declaration

- Consider the following configuration declaration for the HALF_ADDER entity.

```
library CMOS_LIB, MY_LIB;
configuration HA_BINDING of
   HALF_ADDER is
for HA-STRUCTURE
   for X1:XOR2
```

Example of Configuration Declaration cont..

```
  use entity CMOS_LIB.XOR_GATE(DATAFLOW);
end for;
for A1:AND2
  use configuration MY_LIB.AND_CONFIG;
  end for;
end for;
end HA_BINDING;
```

Package Declaration

- A package declaration is used to store a set of common declarations like components, types, procedures, and functions.

- These declarations can then be imported into other design units using a use clause.

Example of Package Declaration

```
package EXAMPLE_PACK is
type SUMMER is (MAY, JUN, JUL, AUG,
   SEP);

component D_FLIP_FLOP
   port (D, CK: in BIT; Q, QBAR: out BIT);
end component;

constant PIN2PIN_DELAY: TIME := 125 ns;

function INT2BIT_VEC (INT_VALUE: INTEGER)
   return BIT_VECTOR;
end EXAMPLE_PACK;
```

Package Declaration cont..

- Assume that this package has been compiled into a design library called DESIGN_LIB. Consider the following clauses associated with an entity declaration.

library DESIGN_LIB;

use DESIGN_LIB.EXAMPLE_PACK.all;

entity RX is . . .

Package Declaration cont..

- It is also possible to selectively import declarations from a package declaration into other design units. For example:

```
library DESIGN_LIB;
use DES[GN_LIB.EXAMPLE_PACK.D_FLIP_FLOP;
use DESIGN_LIB.EXAMPLE_PACK.PIN2PIN_DELAY;
  architecture RX_STRUCTURE of RX is . . .
```

Package Body

- A package body is primarily used to store the definitions of functions and procedures that were declared in the corresponding package declaration.
- Used to do constant declarations for any deferred constants that appear in the package declaration.

Package Body Cont..

- A package body is always associated with a package declaration.
- A package declaration can have at most one package body associated with it.
- Contrast this with an architecture body and an entity declaration where multiple architecture bodies may be associated with a single entity declaration

Example of Package Body

- Here is the package body for the package EXAMPLE_PACK declared.

```
package body EXAMPLE_PACK is
function INT2BIT_VEC (INT_VALUE: INTEGER)
   return BIT_VECTOR is
begin
   --Behavior of function described here.
end INT2BIT_VEC;
end EXAMPLE_PACK;
```

Model Analysis

- Once an entity is described in VHDL, it can be validated using an analyzer and a simulator that are part of a VHDL system.
- The analyzer takes a file that contains one or more design units and compiles them into an intermediate form.
- During compilation, the analyzer validates the syntax and performs static semantic checks

Model Analysis cont..

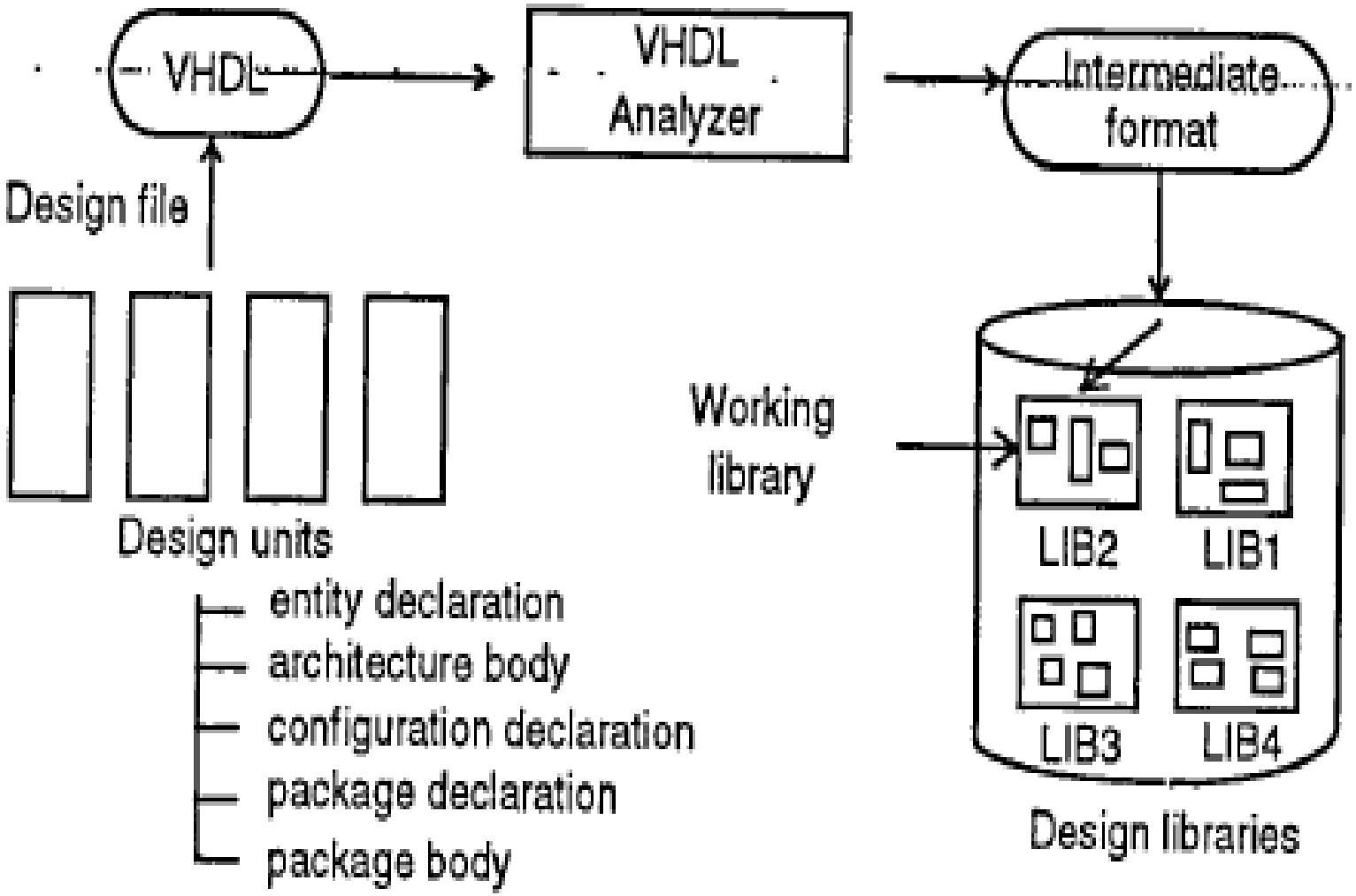

Figure 2.8 The compilation process.

Model Analysis cont..

- The generated intermediate form is stored in a specific design library, that has been designated as the working library.
- A design library is a location in the host environment (the environment that supports the VHDL system) where compiled descriptions are stored.

Simulation

- Once the model description is successfully compiled into one or more design libraries, the next step in the validation process is simulation.

- A simulation can be performed on either one of the following:
- an entity declaration and an architecture body pair,
- **a configuration.**

Simulation cont..

Actual simulation consists two major steps:

1. *Elaboration phase*
2. *Initialization phase*

Simulation cont..

1. *Elaboration phase: In this phase, the hierarchy of the entity is expanded and linked, components are bound to entities in a library .Also storage is allocated for all data objects (signals, variables, and constants) declared in the design units. Initial values are also assigned to these objects.*

Simulation cont..

2. *Initialization phase: The effective values for all explicitly declared signals are computed, are assigned values, processes are executed once until they suspend, and simulation time is reset to 0 ns.*