



## Course 4

# Library Management

## Table of Contents

Library Management in Batch mode .....	3
Creating a New Library .....	3
Listing Mapped Libraries.....	3
Listing Library Contents .....	4
Setting the Working Library .....	5
Deleting a Library Unit .....	5
Attaching a Library .....	5
Linking Libraries.....	5
Updating (Refreshing) Library Contents .....	6
Library Configuration File (library.cfg).....	6
Library Management in GUI mode .....	7
Creating and Attaching Libraries .....	8
Creating a New Library .....	8
Attaching an Existing Library .....	9
Deleting a Library Unit .....	9
Emptying a Library .....	9
Refreshing Libraries.....	9
Compact Library.....	9
Installing FPGA Vendor Libraries .....	10

## Library Management in Batch mode

For the complete reference on the command usage please refer **VSimSA Command Reference** section inside the help documentation located at **Help | Product Help**.

Creating a library is the first step before you can start compiling your source files.

### Creating a New Library

The **vlib** command creates a new library and modifies the library list that describes mapping between logical names and library index files (\*.lib). The command can also be used to compact a library.

The command given below creates library fmeter and adds library reference (mapping) to the local library.cfg file

```
$ vlib work
ALIB: Library 'fmeter' attached.
work = C:\My_Designs\Samples_91\Freq_meter\work\work.lib
```

The command given below creates library constpackage and adds library reference to the global library.cfg file. Such library will be 'visible' from any directory.

```
$ vlib -global constpackage
ALIB: Library 'constpackage' attached.
constpackage =
C:\My_Designs\Samples_91\Freq_meter\constpackage\constpackage.lib
```

### Listing Mapped Libraries

The **vlist** command lists all the libraries visible from the current directory. It first looks for the library.cfg file in the current directory and then in the vlib\ subdirectory of Active-HDL installation directory. If the \$VSIMSALIBRARYCFG environment variable is defined, Active-HDL looks for the library.cfg file in the directory pointed by \$VSIMSALIBRARYCFG before checking the vlib\ subdirectory

The command given below prints the structure of a library tree. Mappings imported from the global library.cfg files are indented.

```
$ vlist -structure freq_meter
freq_meter C:\My_Designs\Samples_91\Freq_meter\freq_meter.LIB
work C:\My_Designs\Samples_91\Freq_meter\work\work.lib
# test C:\My_Designs\test\test.LIB
-$VSIMSALIBRARYCFG
cyclonev C:\Aldec\Active-HDL 9.1\Vlib\cyclonev\cyclonev.LIB
    arriav_hssi C:\Aldec\Active-HDL 9.1\Vlib\arriav_hssi\arriav_hssi.LIB
    arriav C:\Aldec\Active-HDL 9.1\Vlib\arriav\arriav.LIB
    axcelerator C:\Aldec\Active-HDL 9.1\Vlib\axcelerator\axcelerator.LIB
    smartfusion C:\Aldec\Active-HDL 9.1\Vlib\smartfusion\smartfusion.LIB
    iglooplus C:\Aldec\Active-HDL 9.1\Vlib\iglooplus\iglooplus.LIB
    iglooe C:\Aldec\Active-HDL 9.1\Vlib\iglooe\iglooe.LIB
    igloo C:\Aldec\Active-HDL 9.1\Vlib\igloo\igloo.LIB
    proasic3l C:\Aldec\Active-HDL 9.1\Vlib\proasic3l\proasic3l.LIB
    proasic3e C:\Aldec\Active-HDL 9.1\Vlib\proasic3e\proasic3e.LIB
```

```
proasic3 C:\Aldec\Active-HDL 9.1\Vlib\proasic3\proasic3.LIB
```

## Listing Library Contents

To list library units contained within a library, use the **vdir** command. **vdir** can be used to print all library units and all library source files, print the source file of a selected unit, list package contents and list all architectures of an entity.

The below command displays all units from the **v\_fmter** library

```
$vdir -lib v_fmter
module: $root
module: BIN2BCD
module: CNT_10B
module: CONTROL
module: HEX2LED
module: freq_m
top module: freq_m_tb
top module: frequency_meter_tb
```

The below command displays all units from the library **fmter** along with the VHDL architectures in the parentheses, and also shows the correspondent source files:

```
$ vdir -lib freq_meter -source -witharch
C:\My_Designs\Samples_91\Freq_meter/src/and2.vhd
entity: and2 (and2)
C:\My_Designs\Samples_91\Freq_meter/src/cnt_4b.vhd
entity: CNT_4B (CNT_4B)
C:\My_Designs\Samples_91\Freq_meter/compile/CNT_BCD.vhd
entity: CNT_BCD (CNT_BCD)
entity: CNT_BCD (CNT_BCD2)
C:\My_Designs\Samples_91\Freq_meter/compile/CONTROL.vhd
entity: control (control_arch)
C:\My_Designs\Samples_91\Freq_meter/compile/freq_top.vhd
entity: freq_top (freq_top)
C:\My_Designs\Samples_91\Freq_meter/src/TestBench/freq_top_TB.vhd
entity: freq_top_tb (TB_ARCHITECTURE)
C:\My_Designs\Samples_91\Freq_meter/src/hex2led.vhd
entity: hex2led (hex2led)
C:\My_Designs\Samples_91\Freq_meter/src/TestBench/testbench.vhd
entity: TestBench (StimulusFromFile)
C:\My_Designs\Samples_91\Freq_meter/src/testbench_cnt_bcd_conf.vhd
configuration: testbench_cnt_bcd_conf
configuration: TESTBENCH_FOR_freq_top
C:\My_Designs\Samples_91\Freq_meter/src/TestBench/TestBenchPack.vhd
package: TestBenchPack
```

## Setting the Working Library

The working (default) library is the library into which the source files are compiled by default, unless the target library is specified in the command line.

You may set current working library in the Active-HDL tool shell (vsimsa or GUI) by setting the internal **worklib** variable:

```
$ set worklib <library_name>
where library_name is the logical name of the working library.
```

Note that you may only set the working library in the tool shell (vsimsa) or when working in the GUI framework. If you are running the standalone commands of Active-HDL you have to always use **-work** switch for the compilation commands and **-lib** switch for the vsim command.

## Deleting a Library Unit

You can delete a selective library unit by using the **vdel** command.

```
$ vdel -lib test vc1 structure
```

The above example removes the architecture structure of the entity **vc1** from the library **test**.

```
$ vdel -lib test vc2
```

Removes the entity **vc2** and all its architectures from the library **test**.

To remove all units from the library use **vdel -all** command:

```
$ vdel -lib bjack -all
```

## Attaching a Library

Logical names of the libraries are mapped in the library index file (**library.cfg**). To attach a library (i.e. map a logical name into a library index file), use the following command:

```
vmap <logical_name> <physical_name>
```

The following command maps the logical name **test** to the **C:\My\_Designs\test1\test1.lib** library.

```
$ vmap test C:\My_Designs\test1\test1.lib
```

To detach a mapped library, use the following command:

```
vmap -del <logical_name>
```

## Linking Libraries

Instead of individually mapping the libraries that are already mapped in some other **library.cfg** file (in a different directory), you may map the whole **library.cfg** using the command below:

```
vmap -link <directory_path>
```

The following command adds a link to all libraries mapped in the **library.cfg** file located in the work directory. All libraries mapped in the work directory are now accessible in the current directory as well.

```
$ vmap -link C:\My_Designs\Samples_91\Freq_meter\work\work.lib
```

To remove a link to the libraries mapped in the **library.cfg** file located elsewhere in the directory tree, use the following command:

```
vmap -unlink <directory_path>
```

The following command removes a link to libraries mapped in the **library.cfg** file located in the **bjack** directory. Libraries mapped in the **bjack** directory will no longer be accessible in the current directory.

```
$ vmap -unlink C:\My_Designs\Samples_91\Freq_meter\work\work.lib
```

## Updating (Refreshing) Library Contents

To update the contents of a library compiled with an older version of the VHDL compiler, use the following command:

```
vcom -refresh <library_name>
```

To refresh a library compiled with an older version of the Verilog compiler, use the following command:

```
vlog -refresh <library_name>
```

You need to refresh (update) the libraries compiled with the older version of Active-HDL if you want to use them in the current version. Updating the libraries does not require having the source files.

For the libraries containing both Verilog and VHDL units use both commands mentioned above sequentially.

## Library Configuration File (library.cfg)

A library.cfg file may include:

- Library mappings
- \$INCLUDE records that point to another library.cfg file

Library mappings follow the following syntax:

```
<logical_name> = <library_index_file>
```

The name of the library index file can include an absolute or a relative path. If the path name contains space characters, it should be enclosed in quotation marks.

```
The default library.cfg file contains the following mappings:std = ".\std\std.lib"
v1 = ".\v1\v1.lib"
ieee = ".\ieee\ieee.lib"
vital95 = ".\vital95\ieee.lib"
synopsys = ".\synopsys\synopsys.lib"
aldec = ".\aldec\aldec.lib"
vtl = ".\vtl\vtl.lib"
vtl_dbg = ".\vtl\vtl_dbg.lib"
assertions = ".\assertions\assertions.lib"
ieee_proposed = ".\ieee_proposed\ieee_proposed.lib"
```

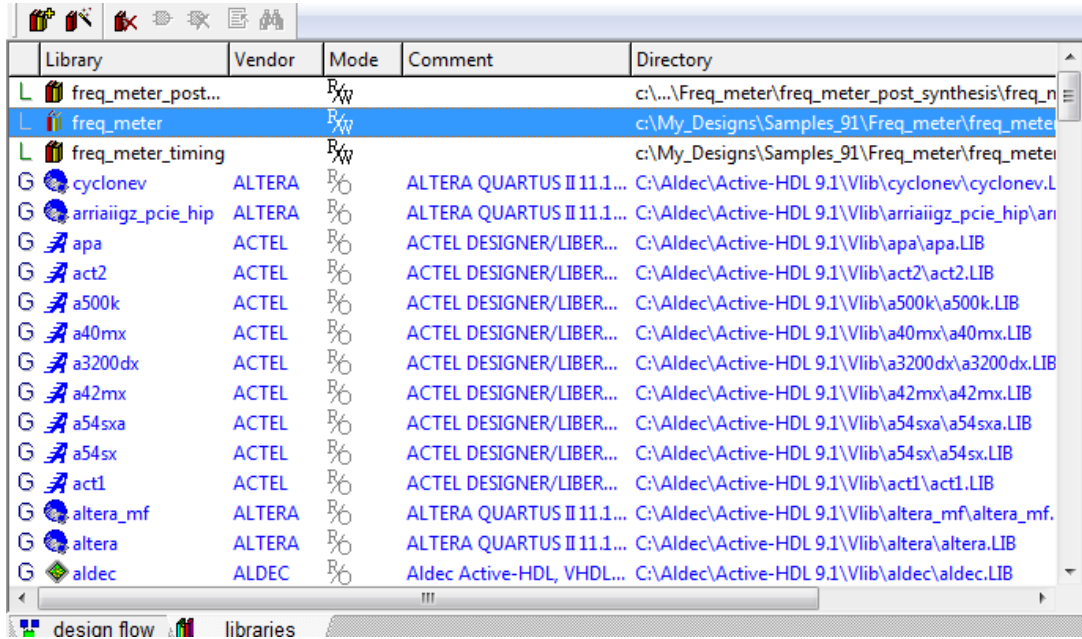
### \$INCLUDE Records

A library.cfg file may also contain \$INCLUDE records.


```
$INCLUDE = "C:\Aldec\Active-HDL 9.1\vlib\library.cfg unisim =
"C:/Aldec/Active-HDL 9.1/Vlib/unisim/unisim.LIB"
```

An \$INCLUDE record points to a directory containing another library.cfg file. Adding an \$INCLUDE record imports mappings from the library.cfg file in the specified directory. To add a link to another library.cfg file use vmap -link command (refer to vmap command syntax for more information).

## Library Management in GUI mode



**Figure 1 Library Manager**

In order to open the **Library Manager** window, choose **Library Manager** (Alt+F7) from the **View** menu or click the **View Library Manager** icon  on the toolbar.

The **Library Manager** window is shown in Figure 1.

You can perform following operations in library manger on libraries and their contents:

- Creating new libraries and setting up their working mode
- Attaching, detaching, and deleting libraries
- Editing logical names of libraries
- Adding libraries to the Symbols Toolbox
- Compacting the contents of libraries
- Viewing the contents of libraries
- Viewing library packages and source files of specific library units
- Deleting simulation data, specific library units, or the entire contents of libraries
- Finding design units in libraries
- Refreshing library contents
- Importing and exporting block diagram symbols

All of the above mentioned operations can be performed using right click context menu from library manger. Right click on any library and you will have context menu as shown below.

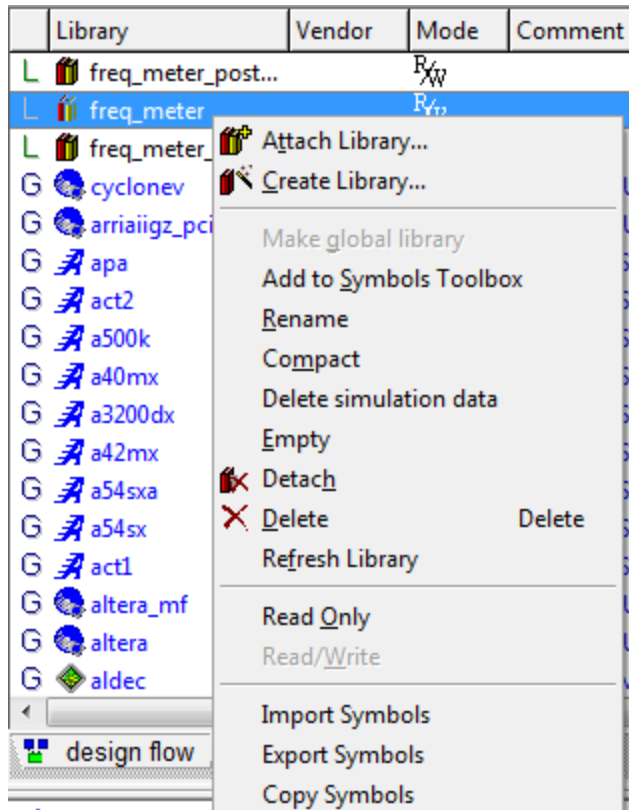


Figure 2 Right click context menu


## Creating and Attaching Libraries

The Library Manager allows you to create new libraries and attach existing libraries. When a new library is created, the Library Manager creates the library index file (.lib) and a mapping to this library in a library configuration file (library.cfg). See Library Mapping in help section of Active-HDL for more details.

In addition to creating new libraries, the Library Manager also allows you to attach libraries. You can only attach libraries that already exist. To attach a library means to create a mapping for it in a library.cfg file.

## Creating a New Library


To create a new library, do the following:

- Click the Create library button  on the toolbar
- Enter the logical name of the library. The logical name is used to reference the library in source code (e.g. in the VHDL library statement) and in macro commands.
- Select the location of the library index file and click NEXT. Note that the NEXT button is inactive until you enter a valid name for the library.

The Library Manager will create a new library and store a mapping to it in the library.cfg file located in the current directory. If there is no library.cfg file in the current directory, it will be created automatically.



## Attaching an Existing Library


Existing libraries can be attached to the library tree with the **Attach library**  button on the toolbar. You can attach both libraries that are not included in the library tree visible from the current directory and libraries that are already mapped. (Each physical library on disk can be mapped under many logical names in the library tree.)


To attach a library, do the following:

- Click on the Attach library button on the toolbar. The Attach Library dialog box will be displayed
- Point to the location where you have .lib extension library stored. Then select the library and click OK

The Library Manager will create a new library and store the mapping to it in the library.cfg file located in the current directory. If there is no library.cfg file in the current directory, it will be created automatically.

## Deleting a Library Unit

To delete a unit from the library, highlight that unit, right click and choose **Delete**  option from the context menu. The Library Manager will ask for confirmation before the selected unit is deleted.

You cannot remove the working library. But you are able to detach it using the **Detach**  option from the context menu. Detached library is not visible in the Library Manager, but you can see .lib files in the File Browser window.

**Note:** You cannot detach library if the selected library is working library.

You can delete attached library using the **Remove Library**  button from the context menu.

## Emptying a Library

To delete all units from a library, select that library in the Library Manager, right click and select **Empty** option from the context menu. The Library Manager will ask for confirmation before the selected unit is deleted.

## Refreshing Libraries

To refresh the contents of a library created with an earlier version of the compiler use **Refresh library** button from the context menu. You need to refresh (update) the libraries compiled with the older version of Active-HDL if you want to use them in the current version. Updating the libraries does not require having the source files.

## Compact Library

You can reduce the size of the library using the **Compact** option from the context menu. This has effect for libraries with design units which have been recompiled several times. Libraries without recompiled units are unaffected.

## Installing FPGA Vendor Libraries

You can install FPGA vendor libraries during installation of the Active-HDL tool. Installer of the tool allows you to install vendor libraries. These Vendor libraries are pre-compiled for respective Active-HDL version you are installing.

You can also download and install vendor libraries from Aldec customer support portal located at <http://www.aldec.com/support>