Transferring Projects to Another Computer

To insure successful project copies, **always** use ACTIVE-CADs built-in copy option. The copy option organizes all file extensions which belong with the project, thus eliminating the most remote possibility of human error.

To copy project to disk:

1. Open ACTIVE CADs **Project Manager**.

2. Select project to be copied by clicking on it in the **Projects** menu. Project name will highlight when selected.

3. Click on the **Copy** button. This will generate the **Copy Project** window.

4. The Copy Project window will show selected project and its directory path in the Source section. A suggested name appears in the destination section. Name to be assigned to copy may be changed to suit preference.

5. Click in the directory field of the destination section and enter destination path, **A:** would be a typical path for copying a project to drive A. This path may also be set by clicking on the Browse button, opening the Browse window, then selecting drive and path with clicks of the mouse button.

6. Click on box for ***Copy all files from project directory.***

7. After verifying all information to be correct, click on the OK button . Project will now be copied.

To copy project from disk:

1. Insert source disk.

2. Open Project Manger.

3. Double click on the **[..]** symbol at the top of the Directories menu. To load from drive A, double click on . This will display projects contained on drive A in the Projects menu.

4. Click on project to be copied, which will highlight when selected from the Projects menu.

5. Click on the copy button, invoking the Copy Project window.

6. Project selected on drive A now appears in the Source sections name field, with its path in the Directory field.

7. Sugested name apears in the Destination sections name feild, which may be modified. A typical path to enter in the directory field would be **C:\ACTIVE\PROJECTS.** Path may also be set using browse function as described above.

8. Click on box for ***Copy all project files.***

9. After verifying all information to be correct, click on OK button and copy from drive A to drive C will be effected.

APN: SIM003

USING MULTIPLE CLOCKS IN A DESIGN

Test vectors have a limited duration, as defined by their formula. Clocks on the other hand are repetitious signals that have no time limitation, except for the simulator limitations. Binary counter outputs may be used as clock signals. However, if more complex clocks are needed, define them under SUSIE-CAD/ACTIVE-CAD as per the following procedure:

Creating Clocks

1. Select the **Add Stimulators** option from within the **Stimulator** menu. .
2. When the keyboard display appears, select the **Formula** button at the bottom of the keyboard display.
3. In response, the **Select Clock** window appears:

- select the **Clocks** option from the **MODE** field.
- double-click on the selected clock ***C1-C4*** in the **Select Clock** field.
- enter the clock formula in the **Selected Clock FORMULA** field.
- click on the **Assign Formula** button.
- select another clock from the **Select Clock** field and repeat the above two steps.

Note that as a clock formula is entered, it is displayed in the **Defined Assignments** window. Finish assignment by clicking on the **Close** button.

Applying Clocks

To assign multiple clocks, follow this procedure:

1. Click over selected signal in the **Signal** field, highlighting it blue.
2. Select the **Add Stimulators** option from the **Stimulator** menu.
3. When the keyboard display appears, click on the desired clock button ***C1-C4*** (located to the right of the keyboard display), which turns black when defined. Buttons remain gray, having not had a formula assigned.

Note that the selected clock has been assigned to the previously selected signal line. The same clock signal may be assigned to any number of signals. Up to four different formula defined clocks may be assigned to the design. In addition, any number of binary counter output based clock signals (Bo-B15 and N0-N15) may be assigned.

Importing A Design Netlist for Virtual Hardware Control

Since it is much easier to work with the Virtual Hardware from the schematic editor than from the simulator screen, you need to invoke the schematic editor screen. Next, select **Import Netlist** from the **Options** menu.

The **Import Netlist** option has a **Schematic** check-off box. If you click on this box, the imported netlists will generate schematics which you can probe while the simulation is in progress. It is a nice feature, however, the size of the schematic is limited to 64 Kbytes under the present release. Future releases (June 1994) will allow you unlimited schematic sizes. To make sure that the netlist loads without any problems, do not load it as a schematic.

After the netlist is loaded successfully, Virtual Hardware Editor displays an outline of the design on the schematic screen. All design input signals will be located on the left-hand side of the block and the output signals will be on the right-hand side.

Select the VHE option by clicking on the **VHE** mode button, which is located right above the simulator button. A VHE menu appears and allows you to select the design pins as input or outputs, for direct interaction with the external hardware.

Use the *Reference Manual* to set up the Virtual Hardware operational environment.

APN-007

Renaming Components

If you want to use some SUSIE components under different names, you have three (3) choices:

- rename existing symbols under the Library Manager and use them under the new names (see Library Manager).
- load an existing symbol into symbol editor and perform all the required changes: change part name, pins, shape, etc. These symbols will coexist with the original ones.
- if the symbols have been used, select the CHANGES mode in schematic, click on the symbol and then on the ATTR(ibutes) button. When the editing window appears, enter the new part name. You need to change each part individually.

To modify any of the SUSIE-CAD (ALDEC supplied) symbols:

- using Library Manager copy the desired symbol from the SUSIE library into the Project library.
- select the symbol from the Project library and modify it under the Library Manager.

APN-10

LPT1 as a Virtual Hardware port

If you have a SUSIE-CAD or ACTIVE-CAD demo, you can directly control any external hardware from your schematic sheet. Also, the schematic designs can be directly subjected to external hardware signals.

To activate the parallel LPT1 port:

- plug into LPT1 the cable used for communication with the actual hardware
- ground pin 11 of the LPT1 port (BUSY signal) at the end of this cable
- Remove (disable) the printer under Windows:
  - select the Windows CONTROL PANEL icon
  - select the PRINTERS option within the CONTROL PANEL
  - select the printer to be removed and click on the REMOVE button

Now you can draw circuits and assign schematic input/output ports to the LPT1 and drive external hardware. To assign the schematic pins as LPT1 ports, click on the **VHE** button in the schematic editor and select test points for direct interaction with external hardware.

*** After each I/O pin assignment you must generate a **new netlist** for the simulator!

NOTE: *Only ports and signal lines with names can be assigned to the LPT1 port. Nameless pins and signal lines cannot be assigned to the LPT1 port.*

2,000 TEST VECTOR LIMITATION

SUSIE can display and save up to 1,000 design test points. Since these test points may produce or receive some long and complex waveforms, SUSIE allows up to 2,000 test vectors.

NOTE: Test vector is a list of logical states of all displayed signals at the given instance in time. These instances are seen by the simulator only if there is at least one signal transition. Typically, a test vector may have tens of signal transitions.

If you reach the 2,000 SUSIE-CAD test vector limitation, you can continue the simulation up to unlimited number of simulation cycles, provided that each 2,000 test vectors you do the following steps:

- save and/or print the current simulation results
- click on the **Waveform Delete** button to delete all waveforms
- reload the ASCII or binary test vector files(*)
- continue the simulation from where you finished before

(*) If you are using the SUSIE proprietary test vectors, such keyboard keys, binary counter outputs or formula test vectors, or mark with the **Cs** marker all test vectors to be retained from the original test vector file, then you do not need to reload any test vector files. They will be automatically available for the extended (over 2,000) simulations. In such a case, you only need to save the current test vector files, click on the **Waveform Delete** button and continue the simulation.

APN-014

Entering Any Tap Names

The bus tap names can be totally unrelated to the bus name. Desired tap names may be entered two ways:

- if the bus tap name already exists, read the ***Renaming Bus Taps*** section.
- if the bus tap exists with no name assigned, read the section ***Assigning Any Names To Bus Taps.***
- if no bus taps yet exists and special names are to be assigned, draw taps using one of the procedures described in **Automatic Bus Taps**. Name or rename taps as per the procedures described in ***Assigning Any Names To Bus Taps*** and ***Renaming Bus Taps***, respectively.

Renaming BUS Taps

To change an existing bus tap name, follow this procedure:

- select the **CHANGES** mode (Schematic Editor, fig. 2-1, button 12).
- click on the selected bus tap name; a red box will appear around the tap name.
- click on the **EDIT** button in the **CHANGES** toolbox.
- when the tap name editing window appears, enter the desired tap name and click the **OK** button to close this window. The new tap name instantly replaces the old tap name.

Assigning Any Names To Bus Taps

If a bus tap has no name, follow this procedure to assign a name:

- select the **CHANGES** mode (Schematic Editor, fig. 2-1, button **12**).
- click on the selected bus tap name, it will turn red.
- click on the **ATTR**(ibute) button in the **CHANGES** toolbox.
- when the **Net Attributes** window appears, click on the **Net Name** button which displays the **Net Name** window.
- enter the desired tap name into the **Net Name** window and click on the **OK** button. The selected tap name instantly appears at the selected spot on the tap.
- use the **MOVE** and **DRAG** options to change the tap name location.

Connecting Hidden Power Pins

Schematic symbols can have some hidden pins. These are usually the power pins. All such hidden power pins are displayed explicitly in Figure 2-22 when the netlist generation is in process. This allows you for example to connect the Vss power pins to the GND signal line or Vdd to the +12V signal line. To make such connections from Figure 2-22, follow this procedure:

- Select **Create Netlist** from the **Options** menu, the **Power Pins** window (Schematic Editor, fig. 2-22) will appear which lists all *hidden* pins.
- Click on the selected *hidden* pin, next click on the **Add** button, the selected pin will instantly move to the right-hand window.
- Click on the selected pin in right-hand window, pin will appear in the **Net Names** window.
- Override the pin name in the **Net Names** window with the desired signal name.
- Click on any pin name in the right-hand window, note that the signal name being entered into the **Net Names** window is automatically assigned to the previously selected pin.

For additional information see the *Hidden Pins In The Netlist* section.

APN-016

Recovering a corrupted project structure:

Projects must always be copied with the ACTIVE-CAD Copy option!
Always copy projects using the **Copy** option in the ACTIVE-CAD **Project Manager.** This requirement is imposed by the fact that all directory names and library paths must be moved together with the project. Otherwise, the project will be left without its data base and will be empty.

Project structure
To correct any problems, you need to understand the structure of project files:
1. The project name that you see in **Design Manager** (.pdf) contains the addresses of all projects (their directories), libraries and netlists. This file typically contains from 300 to 500 bytes. The .pdf files are continuously updated as you make new selections in **Project Manager**, e.g. when you select new resources, new libraries and devices, etc. The .pdf files are residing in directories:
C:\SUSIE\PROJECT (for SUSIE-CAD) or C:\ACTIVE\PROJECTS (for ACTIVE-CAD)

A typical .pdf file for a project (FIB) is listed below:

```
[project]
name=fib                                    ; project name
directory=C:\ACTIVE\PROJECTS\FIB            ; address of project directory
netlist=C:\ACTIVE\PROJECTS\FIB\XNFBA.alr    ; address of post-layout netlist
contents=fib                                  ;
date=06/20/94
time=19:33:23
lib_order=1000 71 53                        ; sequence in which the libraries are used
Type=XILINX                                 ; used by Design Manager to create the design flowchart
generics=MAX                                ; timing parameters (MIN, AVG., MAX) used for simulation

[libraries]
1000=                                       ; the user (project) library that contains his/her macros
71=c:\active\syslib\xbloxu                  ; system library used in the FIB project
53=c:\active\syslib\x4000u                          ; system library used in the FIB project

[xilinx]                                    ; description of the family of models (library)
family=2
xfam=XC4000A

[timings]                                           ; listing of test vector files (names) used in FIB
project
pre=
signals=
routed=

[netlist]                                   ; description of netlist parameters
name=C:\ACTIVE\PROJECTS\FIB\XNFBA.XNF
format=Xilinx
external=FALSE

 [device]                                   ; part number used for the design
part=4002APC84-5
```

xilinx=4002APC84-5

2. If you have a project FIB, then its directory will be:
C:\SUSIE\PROJECTS\FIB (for SUSIE-CAD) or
C:\ACTIVE\PROJECTS\FIB (for ACTIVE-CAD)

The FIB directory includes the following files:

fib.alb	; netlist generated from the current schematic
fib.prj	; list of all schematic sheets used in the project
fib1.sch	; schematic sheet FIB1 developed under current project
fib2.sch	
fib1.bsc	; backup of schematic sheets (this one is for fib1.sch)
fib.xnf	; output ASCII netlist file (Xilinx)
fib.alr	; backannotated (post-layout) netlist file

3. If the project is hierarchical, then its macros will be stored in the LIB subdirectory. If you have created special parts for your project, they will also be stored in the LIB subdirectory. The FIB project would have :
C:\ACTIVE\PROJECTS\FIB\LIB
The following 15 files reside within the LIB subdirectory:

fib.blk	fib.map
fib.dir	fib.mod
fib.fig	fib.net
fib.flg	fib.pin
fib.gnr	fib.sym
fib.hdr	fib.syn
fib.id	fib.vis
fib.ini	

These files are used by the Btrieve database management software to handle the projects. All of these files constitute one project library and they cannot be separated from each other.

If you use the **Copy** option from **Project Manager**, then the entire project structure is copied to the new directory. However, if you use the DOS or Windows copy operations, then only selected files will be copied and the project integrity will be destroyed.

If you have any corrupted project files, you must restore the above-described project structure. There are 3 basic errors that may occur if you copy the projects under DOS or Windows:

- missing .pdf file and **Project Manager** does not show (does not list) the project
- there are library number errors reported
- Library data errors are present

APN-17

What do you need to send to ALDEC when you find a problem?

1. Design problems before routing

The SUSIE-CAD and ACTIVE-CAD have the same file structures. Their designs reside in the following directories:

C:\SUSIE\PROJECTS\project_name\LIB ; for SUSIE-CAD
C:\ACTIVE\PROJECTS\project_name\LIB ; for ACTIVE-CAD

a. You need to send to ALDEC all items from the project_name directory, which includes:
- netlist
- schematics
- project contents (.prj file)

b. You need to send all 15 files created by SUSIE-CAD or ACTIVE-CAD in the LIB directory.

2. You have a design problem with the post-layout netlist

Since SUSIE-CAD and ACTIVE-CAD use system libraries only for post-layout, you need to send only the files from the project_name directory. The netlist file would be sufficient in most cases, however, sending other files helps ALDEC technical staff to faster trace the problems.

How to send the problem design file?

You can zip or pkzip all files and send to BBS: (805)498-4086
the BBS operates in MODE: 8 bits - 1 Stop - no parity; 1,200 bauds to 28,800 bauds

You can also E-MAIL your comments, questions and files to:
support@aldec.com

APN-18

Installing ACTIVE-CAD in place of SUSIE-CAD software

If you have been using SUSIE-XILINX or other products from the SUSIE-CAD family and want to install the advanced ACTIVE-CAD software, you have two choices:

- If you change the ACTIVE-CAD default directory from ACTIVE to SUSIE during the software installation, all the designs that you have developed under SUSIE-CAD will de directly available under the newly installed ACTIVE-CAD. No modifications are needed to use the old SUSIE-CAD products with the new ACTIVE-CAD.
- If you install the new ACTIVE-CAD in the default ACTIVE directory, then the old designs (projects) will not be displayed in the **Project Manager**. To access these projects, you need to copy them into the current project directory . To copy projects, use the **Copy** and **Browse** options in **Project Manager**.

If you are installing SUSIE-CAD in place of ACTIVE-CAD, you need to use a similar software installation procedure.

APN-19

Simulating FPGA Designs Created In Other EDA Tools

ACTIVE-CAD can simulate designs that have been developed in other schematic editors such as OrCAD, Viewlogic or Mentor. Follow this simple seven (7) step process for simulation of other editors:

1. Start *New* project in **Project Manager**; assign any name to it . Select the proper design environment from the listing (e.g. *Xilinx, Actel, Altera,* etc.).
2. Start ACTIVE-CAD (SUSIE-CAD) simulator from its icon; it is better to close the schematic editor if it has been opened.
3. Load netlist; use the **Load Netlist** option in the simulator **File** menu. Select the format of the loaded netlist (in the **From Format** field) and the name of the netlist (**Input Netlist** field).
4. From the **Signals** menu, select **Add Signals** to assign test points in the simulator window. This will select signals for viewing ; Additional signals can be added at any time.
5. Assign stimulators (B0-Bn, F0-Fn, keyboard keys, etc.) to the test points and test vectors;Test vector files may be addedat any time without destroying the existing stimulators on the screen.
6. Simulate the design; use Short Step, Long Step or long simulation (**Simulation Stop** in the **Options** menu); activate the **Stop** button for long simulations.
7. Save test vectors, which control and display the design behavior. Save as many test vectors as is necessary to have a complete documentation of the system behavior.

APN-20

Simulating PLD And CPLD Designs Created In Any EDA Tool

You can simulate in ACTIVE-CAD designs that has been developed in other schematic editors such as OrCAD, Viewlogic or Mentor. The simulation process is simple if you follow these seven (7) step process:

1. Start *New* project in **Project Manager**; you can assign to it any name. However, you should select the proper design environment from the listing (e.g. *Xilinx, Actel, Altera,* etc.)
2. Start ACTIVE-CAD (SUSIE-CAD) simulator from its icon; it is better to close the schematic editor if it has been opened
3. Load netlist; use the **Load Netlist** option in the simulator **File** menu. Select the format of the loaded netlist (in the **From Format** field) and the name of the netlist (**Input Netlist** field)
4. Select test points to the simulator window (use **Add Signals** in the **Signals** menu). Select signals for viewing and stimulation with signals; Additional signals can be added at any time.
5. Assign to the test points the stimulators (B0-Bn, F0-Fn, keyboard keys, etc.) and test vectors; You can add test vector files without destroying the existing stimulators on the screen.
6. Simulate the design; use Short Step, Long Step or long simulation (**Simulation Stop** in the **Options** menu); activate the **Stop** button for long simulations.
7. Save test vectors, which control and display the design behavior. Save as many test vectors as is necessary to have a complete documentation of the system behavior.

APN-21

Adding New Part Names *(Synonyms)*

The system libraries list parts by their most popular names. However, at times you may need to use a different name for the same part. To assign a new name for the same part, use the Synonyms options in the Library Manager.

We will demonstrate the use of the Synonyms option by adding 16L8 to the PAL16L8 family of parts.

1. Select **Library Manager** either directly from the icon, or from **Project Libraries**
2. Select by double-clicking on the library in which the reference part (PAL16L8) resides. In this case it is the PAL library.
3. When the window *LM - List objects in:PAL* appears, click on the *PAL16L8* item.
4. Select the **Synonyms** option from the **Object** menu.
5. When the new window *LM - List Object Synonyms* appears, click on the **Add** button.
6. Enter the synonym name (*16L8*) into the new window and click on the **Add** button in the same window.
7. The newly entered part (16L8) will appear in the Synonyms field of the *LM - List Object Synonyms* window.
8. Press the **Close** buttons and exit **Library Manager**.
9. Your simulator will now load any netlist which includes the 16L8 part.

If you want to add the 16L8 part to the symbols library and use it in the schematic editor, follow this procedure:

1. Place *PAL16L8* symbol on the schematic sheet.
2. Select the **Changes** mode and click on the *PAL16L8* device.
3. Select the **Symbol Editor** option from the **Options** menu.
4. In the **Symbol** field replace the old (*PAL16L8*) name with the new one (*16L8*).
5. Edit the pin assignment in the symbol (pin numbers); do not change pin names!
6. Change the library name to the desired one, e.g. select the current project name from the list of libraries.
7. Perform the *Save As* operation and save the new symbol as *16L8.*
8. Exit **Symbol Editor** via its **Symbol** (**File**) menu.
9. Go to the schematic editor and notice that the symbol *16L8* is now readily available.
10. If the new device has different timing, you cannot create a synonym but need to consult *APN-22 Creating New Timing Models From Functional Equivalents.*

If you draw the new symbol on a schematic sheet, its simulation model, created in the *Synonyms* section (above), will be fed into the simulator and listed in the simulator tables. Notice that the *16L8* symbol will be shown in white and will not simulate till you load it with a JEDEC file.

APN-22

Creating New Timing Models From Functional Equivalents

The system libraries have all of the most popular parts. However, at times you may need to use some older or unique parts which are not in the system library. You can create new models quickly if you can find their functional equivalents in the system library.

The 93S16 (4-bit binary counter) is used as an example to show the process of creating new parts. From the catalog we have found that this counter is identical in functional operation to the 74LS161 device.

1. Select **Library Manager** either directly from the icon, or from **Project Libraries**; Select by double-clicking on the library in which the reference part (74LS161) resides. In this case it is the TTL library.
2. When the window *LM - List objects in:TTL* appears, click on the *74LS161* item.
3. Select the **Copy** option from the **Object** menu
4. When the new window *LM - Copy Object* appears, select the destination library (e.g. *MYLIB*) from the **Destination Library** window and click on the **Copy** button. Remember that you can copy models only into your own libraries. You cannot copy any models into the system libraries because it would destroy their integrity.
5. Double-click on the destination (*MYLIB*) library. It lists the newly copied model in the *LM - Lists Objects In*: window.
6. Click on the copied part (74LS161). It turns blue.
7. Select the **Rename** option from the **Options** menu. Enter the new name (e.g. 93S16) into the **Enter The New Logic Name** field, and click on the **Rename** button. The new part name instantly replaces the old one.
7. Click on the new part (e.g. 93S16) listed in the **LM - List Objects In:...** window and select the **Generics** option from the **Options** menu.
8. A new **LM - Change Propagation Times** window appears. Click on the selected timing parameter in the **Propagation Times** window (its function is listed in the **Description** field).
9. The current MIN, MAX and AVG. timing parameter values are listed in the **Propagation Times Values** window. Edit these values to comply with the catalog data of the new part (e.g. 93S16). Click on the **Modify** button to store the newly entered data.
10. Edit all timing parameters listed in the **Propagation Times** window. Click on the **Change** button when all timing parameters have been edited.
11. If a schematic sheet has been opened, you need to select the **Update Libraries** option within the **File** menu. This operation will update the current library status. However, if you are starting a new project, it will automatically use a library (e.g. *MYLIB*) with updated timing parameters.

GENERAL NOTES:

To store all special parts in one user library, create one by selecting the **Create** option in the **Library** menu of the **Library Manager**. This is generally better than storing all special parts with their projects. If you need to reassign pin numbers in the new model, you can do it under the Symbol Editor (YOU CANNOT CHANGE PIN NAMES!).

APN-23

Rebuilding Lost Project Libraries

If due to improper operations your project (e.g. *FANCTRL*) lost all its libraries, you can restore them by applying this 4-step process:

1. Create a new project (e.g. *DECODER*) in **Project Manager**;

Create this project within the same design environment (Xilinx, Actel, Lattice, etc.) and attach the same libraries that the corrupted project (with the lost libraries) had.

2. Since you cannot do any operations on the current project, select a third project in the **Project Manager**.

3. Select the new project (*DECODER*) in **Project Manager** and click on the **Copy** button. When the **Copy Project** window appears, enter the corrupted project (*FANCTRL*) and its directory as the destination project. Click on the **OK** button.

4. ACTIVE-CAD will warn you that project (*FANCTRL*) already exists and will ask you if you want to overwrite the project. Respond with *YES.*

By performing these operations, you will restore the hidden library numbers that are used by the **Library Manager** to control the use of system and user libraries.

APN-24

How To Correct The BTRIEVE Errors

The ACTIVE-CAD and SUSIE-CAD software includes the BTRIEVE engine from Btrieve technologies. This database engine is used by many software packages, such as FAXWORKS and others. In such a case there may be another copy of the BTRIEVE program somewhere in your computer system.

Since ACTIVE-CAD and SUSIE-CAD (REV. 1.8.7 and higher) require BTRIEVE 6.15 or later, they will not work if you have had some older versions of these programs installed on your computer.

Also, if you are using some other programs that are based on older versions of BTRIEVE, ACTIVE-CAD and SUSIE-CAD will not start. You will typically get a message Error 29, Btrieve incorrectly initialized, or similar.

To resolve this problem, you need to copy five (5) files from the ACTIVE\EXE or SUSIE\EXE directory whose names start with WBT*.*, into WINDOWS directory. You should remove all other instances of the WBTRCALL. DLL file from the hard drive. They can typically be found in the directory WINDOWS\SYSTEM.

By updating your system to the latest BTRIEVE software drivers, you will be able to use ACTIVE-CAD and other software installed on your computer.

APN-25 ViewLogic Schematic Import Option for ACTIVE-CAD

This preliminary documentation applies to ACTIVE-CAD REV.1.9 ViewLogic Import option (SC_ALV.DLL ver. 1.7.3). The current version of the ViewLogic import software is semi-automatic and does not automatically search for ABEL files. FPGA libraries are automatically imported, schematics need only be selected for import to ACTIVE-CAD.

The fully automated ViewLogic import software will be released in June, 1995.

Introduction

ACTIVE Schematic imports designs created with any ViewLogic schematic capture program. This includes DOS, Windows and workstation versions of ViewDraw. To import a design, all original schematics and symbol libraries that were used to create the design will be needed.

ACTIVE-CAD can import ViewLogic schematics and libraries in their native format. After the conversion, the schematics can be directly edited and simulated in ACTIVE-CAD.

NOTE: No ViewLogic keylock or license is required to import ViewLogic designs. The imported designs are no longer subject to any design size limitations that may have existed in ViewLogic software products.

ViewLogic directory structure

To import ViewLogic designs, it is important to understand their directory structure.

Schematic files

ViewLogic schematics do not have file extensions that would distinguish them from other design files. Instead, all ViewLogic schematics and hierarchical macros are located in the **SCH** directory. For example, there can be three SCMAC.1 files in three different directories. The SCMAC.1 file located in the SCH directory is a schematic file, the SCMAC.1 file located in the SYM directory is a symbol file, and the SCMAC.1 file located in the WIR directory is a netlist.

Schematic files often have the same name but different extensions, e.g. SCMAC.1, SCMAC.2, SCMAC.3. These files represent different pages of the same schematic (multipage schematic).

Symbol files

The symbol files exist only in hierarchical designs. Flat designs do not have any symbol files. All symbols of the hierarchical ViewLogic macros are located in the **SYM** directory. The symbol files are automatically searched for by ACTIVE-CAD during the schematic import, so that the entire hierarchical design can be imported without any user intervention.

Library files

All system symbol libraries are located in the directory having the same name as the FPGA or CPLD library itself. For example, the XC5200 library is located in the XC5200 directory and consists of several files. ACTIVE-CAD uses only the SCH.LIB and SYM.LIB files to import the libraries and macros. It is recommended that these libraries be located on the same drive as the imported design files. This will allow ACTIVE-CAD to automatically search for all libraries with no user intervention.

Netlist files

Netlist files in the WIR directory are not used during the import because ACTIVE-CAD will regenerate its own netlist(s) from the imported schematics.

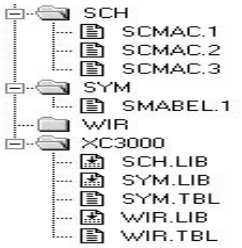
ABEL files

All ABEL files are located in the main project directory. For example, if the project SCMAC has three ABEL macros, all three ABEL files (macros) will reside within the SCMAC directory.

Example

This example demonstrates what files you will need to import a Xilinx 3000-based design:

SCMAC



In the above example ACTIVE-CAD will import the SCMAC.1, SCMAC.2 and SCMAC.3 schematic files from the **SCH** directory. If ACTIVE-CAD finds any schematic macros on these schematic sheets, it will automatically read them from the **SYM** directory. All system symbols for flip-flops, counters, etc. will be read from the SYM.LIB file located in the FPGA (XC3000) directory.

NOTE: Since the ViewLogic symbols for Xilinx Unified libraries, LATTICE and ACTEL are identical to the ACTIVE-CAD symbols, they do not have to be imported. ACTIVE-CAD will use its own system libraries for these FPGA families.

Setting up the ACTIVE-CAD Project

Before importing any ViewLogic files, a new ACTIVE-CAD project will need to be create d. In the **New Project** window, select the device type and FPGA family identical to that used by theViewLogic design being imported. For example, in order to import an XC3000-based ViewLogic design, select the *XILINX* type and *XC3000* family in the **New Project** window. Selecting the part number may be performed at a later time.

Selecting Libraries for ACTIVE-CAD project

When ACTIVE-CAD creates a new project with the FPGA family selected , all appropriate libraries are automatically selected. Library selection may be verified by clicking on the **Libraries** button in the **Project Manager** window, which displays the library listing for review.

NOTE: If for some reason, ViewLogic design import has been aborted, delete the project in the **Project Manager**, and then restart project before performing any new design import operations.

Importing ViewLogic Schematics

If the ViewLogic libraries are on the same drive as the schematic files, the import process may completed with no user intervention. However, if the libraries are on a different drive, you may need to browse drives and directories in order to manually set paths to ViewLogic libraries.

Importing schematic sheets

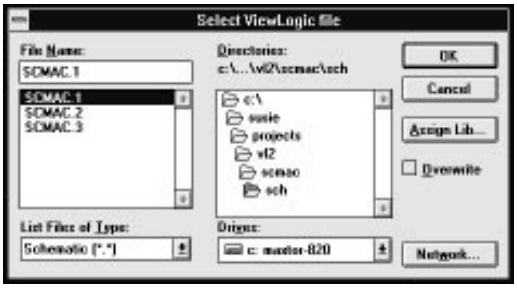
To import schematic sheets, select the **ViewLogic Schematic** option from the schematic editors' **File/Import** menu and apply one of the following procedures:

a. The project libraries are on the same drive as the project

1. When the **Assign ViewLogic Libraries** window appears, select the **Automatic Library Search** box (activates the automatic search (file load) mode), and click on the **OK** button. ACTIVE-CAD will automatically search for all required ViewLogic libraries and display them in the **Assign ViewLogic Libraries** window.
2. Deselecting the **Always Show Libraries** box will allow ACTIVE-CAD to skip the **Assign ViewLogic Libraries** window the next time the **ViewLogic Schematic** option is selected from the **Import** menu.
3. Click on the **OK** button. This displays the **Select ViewLogic File** window. Select the *Schematic (*.*)* option from the **List Types of Files** field.

5. Select all schematic files from the **File Name** field, e.g. by dragging the mouse cursor or by using the **Ctrl** key, and click on the **OK** button.

NOTE: The ViewLogic schematic files have numbers as file name extensions, e.g. SCMAC.1, SCMAC.2, etc. These extensions enumerate all schematic sheets in the project. If the file name extension is not a number, then the file is not a schematic.

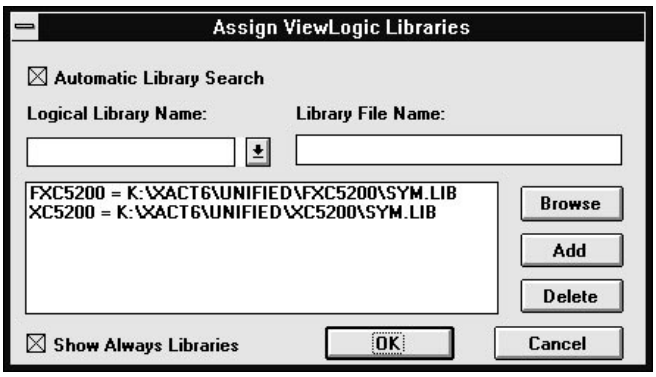


The selected ViewLogic schematic file will automatically be converted into an ACTIVE-CAD schematic with the same name as the original schematic and will be a part of the current ACTIVE-CAD project.

If your original schematics have a name with 8 characters, the imported schematics will have that last character replaced with the page number. For example, when importing SCXLXSCH.1, SCXLXSCH.2 and SCXLXSCH.3 schematics, the ACTIVE-CAD schematics will be called SCXLXSC1.SCH, SCXLXSC2.SCH and SCXLXSC3.SCH respectively.

b. The project libraries are on a different drive than the project

If the libraries are on a different drive than the project, ViewLogic libraries containing symbols and macros must be manually imported.



NOTE: The libraries listed in the above window include all ViewLogic libraries found in your system. When importing a particular design, only the libraries used in that design will be added to the list. This list of libraries will grow as more designs are imported using different libraries. This list of ViewLogic library locations is saved in the SUSIE.INI or ACTIVE.INI file and can be used by all ACTIVE-CAD projects.

Manual library selection

Click on the **Browse** button in the **Assign ViewLogic Libraries** window and locate the SYM.LIB file that belongs with the library to be used. Next, type in the **Logical Library Name** field the library name. If the library name is not known, use the following procedure:

1. Without selecting the library, proceed with importing schematics.
2. All libraries that could not be found, will be reported in the ACTIVE-CAD **Messages** window.
3. Next time the **Assign ViewLogic Libraries** window is opened, it will show a list of libraries found which have been assigned directories. This window will also list libraries that have not been found and thus have no associated directory.
4. For each of these libraries not found, specify the location (directory) of the SYM.LIB file in the **Logical Library Name** field and then click the **Add** button.
5. No schematics should be imported untill all the libraries have been assigned their directories.

Importing Hierarchical schematics

When importing a design that has hierarchical macros created by the user, only the top level schematicsneed to be imported. All the macros will be found automatically and the entire design will be converted in one step.

NOTE: If a hierarchical macro contains multiple pages of schematics, it cannot be automatically converted to an ACTIVE-CAD macro at this point. Either move all schematics into one page in ViewDraw and then import it again or contact ALDEC Technical Support for assistance. The ACTIVE-CAD REV. 1.9.2 will resolve this problem.

Importing a schematic with non-schematic (ABEL) macros

Some FPGA designs contain symbols that are represented by text macros, e.g. XABEL state machine, or VHDL text. These symbols will be imported as empty macros and will need to be assigned contents (netlist file) after converting the design into the ACTIVE-CAD schematic.

To better understand the ABEL file import, please review the ViewLogic file structure section. The *Example* in that section lists SMABEL symbol file. Since the SCH directory does not list any schematic with the same name, it is thus a macro described by an ABEL file rather than a schematic file. This file is located directly in the SCMAC project file.

There are two (2) methods of reading ABEL macros:

Importing an ABEL file after Macro symbol import

ACTIVE-CAD automatically imports all symbols when schematics are imported. The schematics are imported from the SCH directory and the symbols are imported from the SYM directory (see *Example* directory structure).

ACTIVE-CAD automatically creates a symbol each time it imports an ABEL file in the Design Manager. If a ViewLogic macro symbol is imported, then an ABEL file is imported, the ACTIVE-CAD generated symbol will override the ViewLogic symbol and any schematic that uses the ACTIVE-CAD macro symbol will not fit the ViewLogic schematic wiring.

Edit these mismatched wires by selecting the **Changes** mode in the ACTIVE-CAD schematic editor and dragging the wires to the proper pins. This is the preferred method of importing an ABEL file which may be subject to many changes.

Importing a Macro symbol after an ABEL file

To make certian that an ABEL file imported to ACTIVE-CAD did not destroy the ViewLogic symbol geometry and to allow the symbol and its wiring to match perfectly, restore the symbol to its original shape by importing it again from ViewLogic.

To import a macro symbol, follow these steps:

1. Select the **ViewLogic Symbols** option from the **File/Import** menu. In response, ACTIVE-CAD displays the **Select ViewLogic File** window. Note that the **List Files of Type** field shows *Symbol (*.*).*
2. Select the **Overwrite** box and the appropriate ABEL symbol (file) in the Directories. Click on the **OK** button to import the selected symbol.

This is the preferred method when the imported ABEL file will not be modified and imported again.

Correcting Conversion Errors

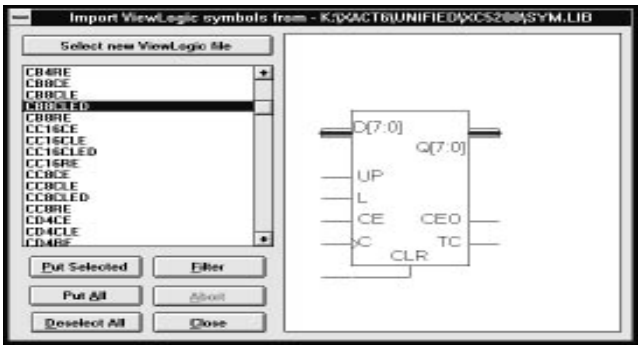
If your ACTIVE-CAD schematics do not look identical to the ViewDraw schematics, please report this problem to ALDEC Technical Support (805-499-6867). All errors can be corrected manually by editing the schematic in the **Changes** mode.

Importing Symbol Libraries

You do not need to read this section when you are importing ViewLogic FPGA designs because ACTIVE-CAD takes automatically takes care of symbol libraries. Only when you are importing system level designs that include TTL, ECL, CMOS, etc., you should read this section.

The **File/Import** menu has a **ViewLogic Symbols** import option. This option is very useful when moving an entire library of custom symbols into ACTIVE-CAD. However, when you are using the **Import Schematic** option in the **File/Import** menu, all symbols are automatically searched for and converted into the ACTIVE-CAD format, and this option does not have to be used.

To import a symbol library, select the **ViewLogic Symbols** option from the **File/Import** menu. Next, select in the **Select ViewLogic File** window the SYM.LIB file and press the **OK** button. This displays **Import ViewLogic Symbols From** window which lists all symbols in the selected library.

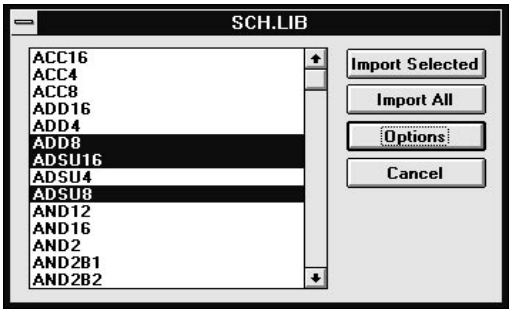


You can select the desired symbols by clicking on the symbol names, and then pressing the **Put Selected** button. You can also import all symbols by pressing the **Put All** button. Each time you select a symbol on the list, a preview of that symbol is shown in the right-hand field of the import symbols window.

Importing Schematic Libraries

Importing schematic libraries is needed only when you import a library which contains some schematics that you want to edit. Remember that the imported symbols have no models and you cannot use them for the simulation purpose, unless you write or assign existing models in ACTIVE-CAD.

To import a schematic library, proceed the same way as with importing a schematic, but in the **Select ViewLogic File** window, select *Schematic Library* (SCH.LIB) and then browse the directory to the location of the SCH.LIB file. After selecting the library file, a listing of all its macros will appear.



After you have selected the desired macros to be imported, press the **Selective Import** button. If you press the **Import All** button, all schematics in the library will be imported.

NOTE: All schematics imported from the schematic library will be automatically saved in the current project library as macros. They will have their ViewLogic symbol names preserved.

ViewLogic built-in libraries

When importing FPGA libraries, you will find that the ViewLogics BUILT-IN libraries are being used for some low level models. These libraries provide the simulation models for the ViewSim simulator and are not being used by ACTIVE-CAD, which has its own symbols and models. Thus, these BUILT-IN libraries do not need to be imported.

Preserving Schematic Properties

When importing a schematic, all symbols, nets and pin parameters are also imported. The only exceptions are the properties that start with the @ character. These properties are used for passing simulation parameters to ViewSim simulator and are not used by ACTIVE-CAD.

Error messages

The list of error messages reported during the conversion process will be released in the next version of this document. You should, however, follow these basic rules:

1. Ignore the Negative Coordinates messages because ACTIVE-CAD will draw the schematic correctly despite this message.
2. Ignore the Cannot find *xxx* symbol message; it means that the imported schematic does not have any symbol because it is a flat design.
3. Ignore the Cannot find *xyz* schematic because the symbol associated with this message is defined not by a schematic but in ABEL.

APN-26

How To Attach Corrected System Libraries

When you receive a new system library from ALDEC , such as ALTERA, IFX80 (INTEL), etc., you need to update your old system libraries. Using the **Windows File Manager**:

1. Remove (Delete) the old library that you will be replacing. That file resides in SYSLIB directory (e.g. C:\active\syslib).
2. Copy the new library into the SYSLIB directory

Select **Project Libraries** from the **File** menu and perform these additional steps:

1. Click on the **Attach** button in the **ACTIVE-CAD(***SUSIE-CAD***)-Libraries** window
2. When the **Attach Library** window appears, click on the drive on which the SYSLIB directory resides (e.g. ***C:***), and then click on the **Update** button. ACTIVE-CAD lists all system and user libraries in the **Libraries Found** window.
3. Click on the selected library in the **Libraries Found** window and then click on the **Attach** button. This instantly moves the selected library (e.g. IFX780) into the **ACTIVE-CAD(** *SUSIE-CAD***)-Libraries** window. Exit both attach windows by clicking on their **Cancel** buttons.
4. Click on the selected library (e.g. IFX780) in the **ACTIVE-CAD(***SUSIE-CAD***)-Libraries** window and then on the **Add** button. The selected library is instantly moved to the **Project Libraries** window, located on the right-hand side of the **ACTIVE-CAD(***SUSIE-CAD* **)-Libraries** window. This library is now available and you can start project operations, such as design entry and simulation.

APN-27

How To Connect XBLOX Bus Pins to Single Wire Pins

ACTIVE-CAD allows you to connect XBOLX bus pins to symbols with single-wire pins. A typical example of such connection is shown in Figure 1, where INPUTS bus output is connected directly to the single-wire input pin WR_EN. To make this connection work, follow these steps:

1. Draw INPUTS and SRAM symbols
2. Select the **Changes** mode in the Schematic Editor. This mode is used for redefining symbols, wires and pins.
3. Double-click on the INPUTS symbol to edit its pin parameters. In response, ACTIVE-CAD displays the **Edit Symbol** window, shown in Figure 2.
4. Click on the BOUNDS item in the **Parameters** field. It is instantly copied into the **Name** field.
5. Enter into the **Description** field the upper and lower bounds of the bus by typing in: *0:0*
6. Click on the **Done** button to complete the BOUNDS limit assignment.
7. Click on the ENCODING item in the **Parameters** field and enter into the **Description** field: *UBIN*. Click on the **Done** button to complete the entry. To display the selected parameters on the schematic sheet, click on the **Display All** button or double-click on the selected parameters. All parameters selected for schematic display have a double asterisk in front of the parameter name.
8. Click on the **Update** button to update the INPUTS schematic symbol. Click on **Close** to exit the setup window.
9. Select the **Wire** mode and click on the **Bus** button in the toolbox.
10. Draw a bus wire from the INPUTS output pin. Click the left mouse button to anchor the corners of the bus, if needed. Draw the bus across from the WR_EN pin of the SRAM module so that you can connect it with a bus tap, as shown in Figure 1.
11. Click the right mouse button to end the bus drawing process.
12. Click on the **Name** button in the toolbox. When the **Edit Bus** window appears, enter the destination pin name, e.g. *WR_EN*. Set both the upper and lower bus limits to *0*.
13. Click on the **Tap** button in the toolbox.
14. Click on the bus name, e.g. WR_EN, and then on the single-wire pin which has the same name. A tap is drawn between the pin and the bus.
15. Click on the **Query** button and then click on the bus. Note that the bus is connected to the INPUTS symbol pin.
16. Click on the wire tap from the WR_EN pin and note that it is connected through the bus wire as the WR_EN0 net.
17. You can generate an XNF netlist (**Options/Export** netlist option) to confirm the connectivity between the INPUTS symbol bus pin and the WR_EN single-wire pin of the SRAM symbol.

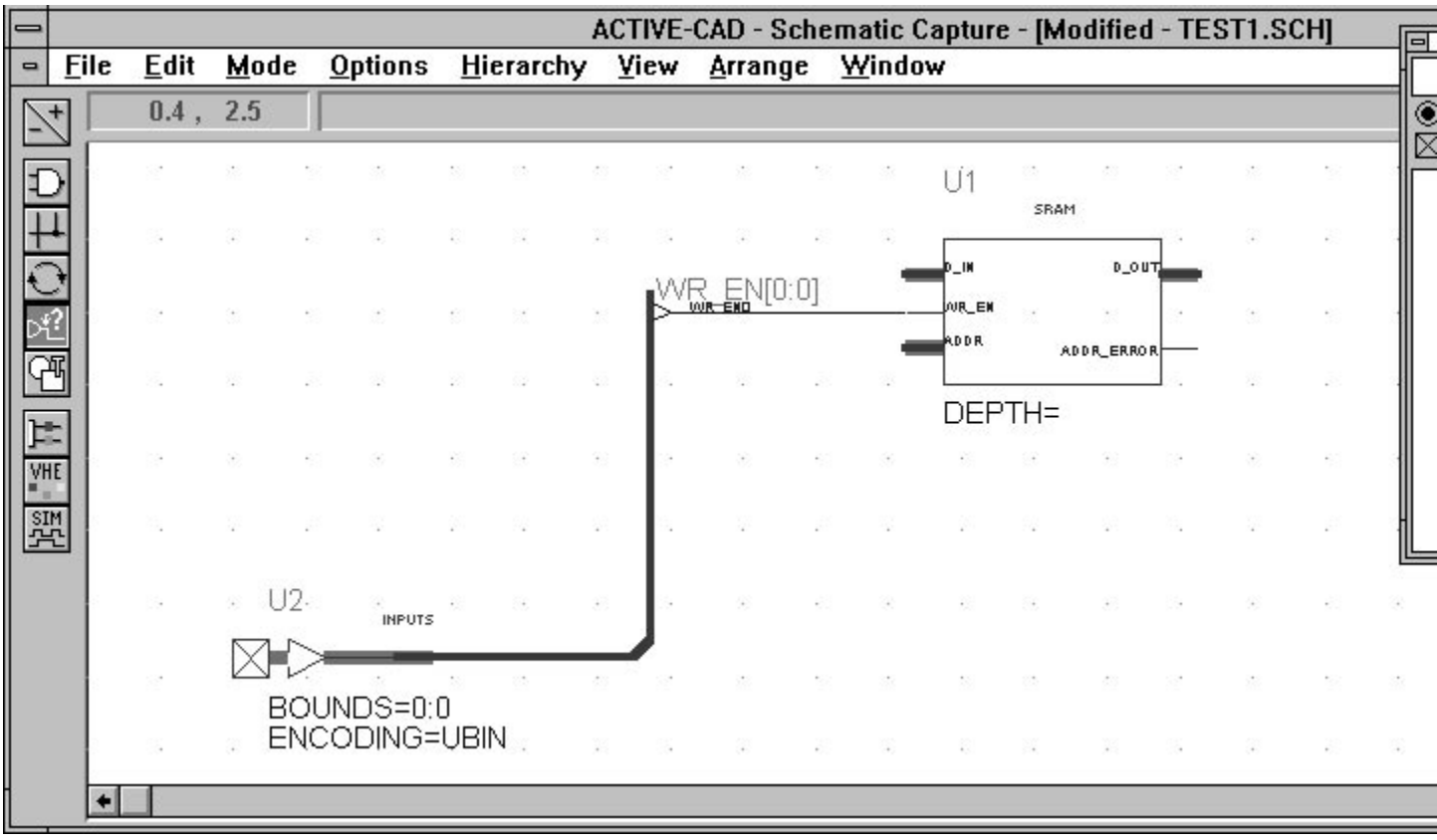


Figure 1. Connecting Bus Pins To Single-Wire Pins.

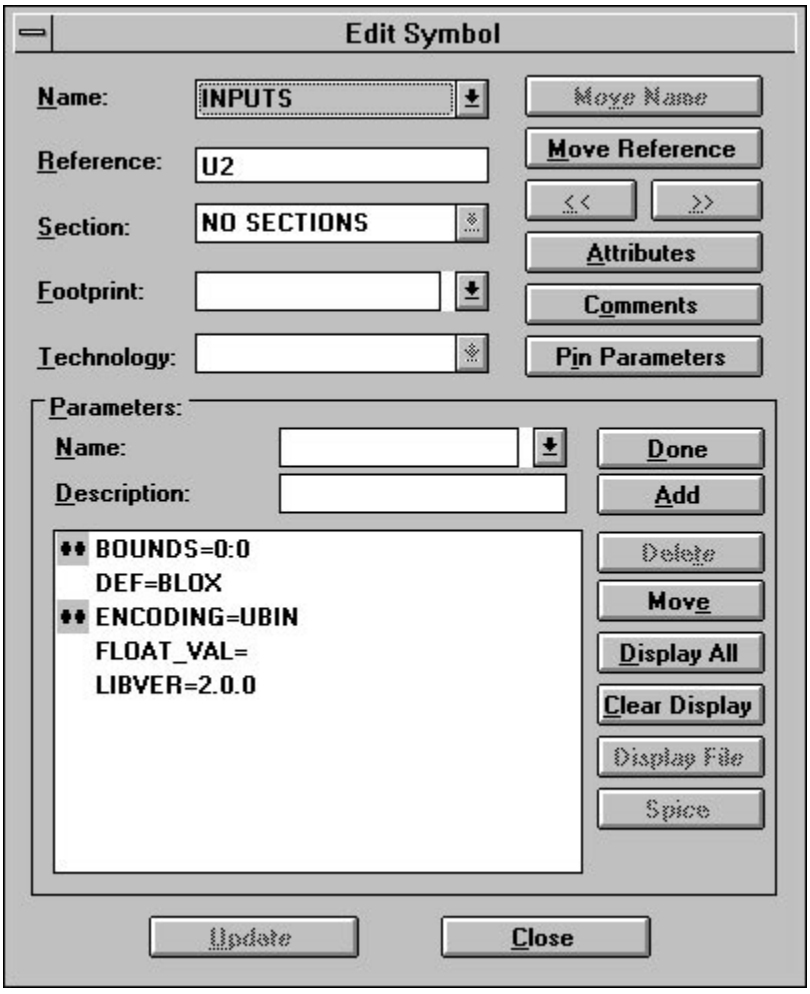


Figure 2. Setting Up The Pin Parameters.

APN-28

Installation Instructions

Software Installation

ACTIVE-XILINX has a *self-adapting* **Design Manager** which automatically looks for all your Xilinx-related design tools and puts them into one control flowchart. For the **Design Manager** to be able to find your Xilinx tools, they must be installed prior to installing the ACTIVE-XILINX software. Once you have verified that all these tools are present, you may install ACTIVE-XILINX:

1. Start Microsoft Windows (do not install ACTIVE-XILINX with other programs opened!).
2. Insert DISK 1 of ACTIVE- XILINX into your floppy drive.
3. Choose the **Run** command from the Windows Program Managers **File** menu.
4. When the **Run** windows appears, enter A:***install*** (for B drive, enter B:***install***).
5. Click on the **OK** button.
6. Follow the installation instructions. Remember that the installation program modifies your CONFIG.SYS file as follows: *FILES=80, BUFFERS=30.*
7. Your SYSTEM.INI is modified as follows: [*386Enhanced*], *TrapTimerPorts=FALSE.*
8. Upon completion of the installation, ACTIVE-XILINX produces a set of ACTIVE-CAD icons.
9. **Exit Windows!!!**
10. Start Windows again. Click on the selected ACTIVE-CAD icon.

For additional information, refer to the XILINX Applications manual, page 1-3.

Keylock installation

1. Turn off your computer
2. Insert the ACTIVE-CAD keylock into the LPT1 or LPT2 port (it can be inserted into Xilinx keylock placed on that port)
3. Turn the computer on.

Starting The ACTIVE-XILINX Software

To use the ACTIVE-XILINX software, you must start a new project. If you are not ready to start your own project, you can open one of the sample projects that come with the ACTIVE-XILINX package.

1. Double-click on the schematic or **Design Manager** icon.
2. ACTIVE-XILINX warns you to open a project and opens the Project Manager window so that you can choose one of the existing designs.
3. Double-click on a sample project in the **Project Manager** window (e.g. *FIB* or *DACDEMO*).
4. ACTIVE-XILINX opens the project internally; when you click on the **Schematic Editor** icon or **Edit Design** button in the **Design Manager**, a schematic screen with the selected schematic design will open.

You can open the simulator using similar design steps.

APN-30

Designing With XBLOX

Introduction

The XBLOXTM logic design and synthesis tool from XILINX provides a library of functional blocks that can be used to describe a system operation. Using these high-level functional blocks instead of gate primitives, greatly simplifies the design process because you can operate with the desired design functions rather than their physical implementations.

The functional blocks in the XBLOX library complement the macro and gate level cells that are provided with the Xilinx XC4000 FPGA development software. The XBLOX synthesis tool automatically expands the functional blocks to accommodate the desired bus width and I/O complexity. Furthermore, the XBLOX design tool has a unique method for global definition of bus structures and data formats that simplifies design entry and modification. The XBLOX design entry has the following benefits:

- Simplifies design entry by using the familiar MSI & LSI components.
- Simplifies design modification; single data path entry may redefine the entire design
- Easier design optimization because the system performance and operation criteria can be directly entered on the schematic.

The XBLOX modules implementation is optimized for the specific needs of each module. The structure of the data paths must be defined only once, for one block only, even though the data paths may propagate through many XBLOX logic modules. The bus width and data type automatically propagate through all levels of design hierarchy and resize all design blocks along the way. The bus size of an entire design can be modified by changing just a few fields on the schematic.

Design Flow

You can quickly learn the benefits of X-BLOX design technology by following the enclosed example which shows a Fibonacci sequence generator. A Fibonacci sequence is one in which every value is the sum of the last two values. The following is a Fibonacci sequence:
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

A Fibonacci sequence generator schematic is shown on the attached sheet. It contains an adder which feeds Registers A and B. The Register A delays the adder output by one clock cycle and the Register B delays it by two clock cycles. The adder adds both registers outputs and generates a new Fibonacci sum.

1. Start a new Project

To start an X-BLOX schematic design, select **Project Manager**, click on the **New** button and enter the new project name. For our sample design, enter *blox* and press the **OK** button. Following this, select the **Project Libraries** option and add the XILINX4 and XBLOX libraries to the new project. Next, enter the schematic shown on the attached sheet. Notice that connecting the wires to the XBLOX bus pins makes them look thicker. These wires are still treated as single wires because their (data) width and type have not yet been defined. The XBLOX wires color and thickness is defined by selecting the appropriate options within **Preferences**, being a part of the **Options** menu.

You can mix on the same XBLOX schematic the XBLOX symbols and XILINX4 macros and primitives.

If you start simulating a schematic that has any XBLOX symbols, they will be reported as having missing models. The XBLOX symbols can be simulated only after a synthesis is performed by the XBLOX software. The XILINX4 macros and primitives can be simulated immediately after the schematic entry, without any additional synthesis.

2. Define parameters

After you have entered the basic schematic, you must define the XBLOX symbol data types and specific functions performed by each block. To enter this data, select the **Changes** mode, double click on the selected symbol and when SUSIE displays the **Edit Symbol** window, enter the desired design data into the **Parameters** section. Each parameter has a **Name** and **Description** field. For example to assign an external pin, enter *LOC* into the **Name** field and *P51* into the **Description** field. Then click on the **Add** button so that the new parameter is added to the list. When you finish entering symbol parameters, click on the **Update** button. If you want the parameters to be displayed next to the symbol, click on the **Display All** button. For the sample *blox* design, enter the parameters from the table below.

Symbol	***Parameters***
U12 -OUTPUTS	ENCODING=UBIN BOUNDS=3:0 LOC[2]=P58 LOC[3]=P57
U9-FORCE	VALUE=0
U1-FORCE	VALUE=16#00#

It is a good practice to label all buses, nets, modules and other symbols in your design. The XBLOX software refers to these names in the synthesized design, and it is much easier to correlate your X-BLOX modules with the compiled design data.

3. Create an XNF netlist

After the schematic is finished, save it and generate an XNF netlist using the **Export Netlist** option in the schematic **Options** menu. In the **Export Netlist** window select: the *Xilinx* netlist format from the **To Format** menu, the *blox.alb* file from the **Input Netlist**, and check the **Flat Netlist** setting, then click on the **OK** button. In response, a window **Enter XILINX Part Name** appears. Enter, e.g. *4003PC84*, as the parts name. This tells the Xilinx software what Xilinx device should be used for the place and route of the *blox* design.

4. Run the XBLOX program

To run the XBLOX synthesis software on the above generated XNF netlist, enter the following command from the DOS prompt.:

```
XBLOX   BLOX
```

The XBLOX software processes the *blox* design netlist and sends to the screen messages describing the propagation of the Data Types, optimization and netlist generation. The XBLOX software will produce an expanded XNF netlist, called blox.xg. It will also produce a *.hm (Hard Macro) file for this design.

5. Place and Route the design

The design can now be completed by running the place and route (PPR) program:

```
PPR  BLOX.XG
```

PPR will read the XG file and produce a design routed for the selected XILINX device. The PPR program produces an *.LCA netlist that contains the LCA layout and programming information.

Next, you need to run the MAKEBITS program to calculate the layout timing delays:

```
MAKEBITS -w BLOX
```

This program updates the LCA file with the timing information.

6. Prepare the XNF file for simulation

To simulate the routed design, you need to generate an XNF file with the timing information. Enter under the XILINX software:

```
LCA2XNF BLOX
```

This program produces a new BLOX.XNF file that overwrites the existing BLOX.XNF file, used as an input for the device layout. If you want to keep the original (pre-layout) .XNF file for documentation purposes, you need to save it under a different name before you run this program.

7. Load the XNF netlist for simulation

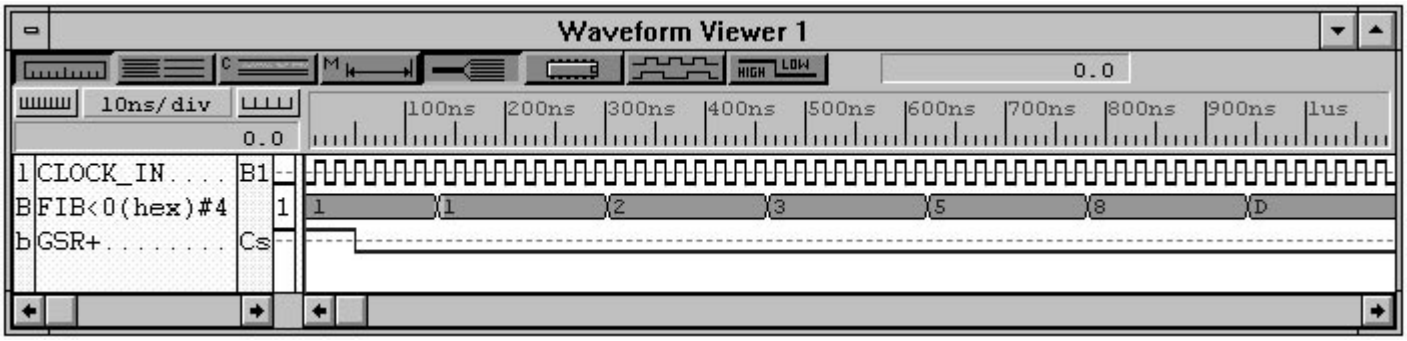
To simulate the synthesized *blox* design, you need to load into the simulator the external netlist that has been generated by the LCA2XNF program. You can still use the schematic to select the test probes, but any schematic changes that you may make will not be updated to the simulator tables and the simulator will continue to simulate the original (compiled) design.

To load the LCA2XNF program-generated netlist, select the **Load Netlis**t button from the **File** menu. Next, select the Xilinx input netlist format and locate the routed XNF on your hard disk. After you click on the **OK** button, the simulator will automatically load the selected routed XNF file for simulation.

8. Simulate the design

To simulate the X-BLOX design, you need to apply two test vectors:
- CLOCK-IN signal which clocks the registers
- GSR+ design reset signal, which should be set to High for 20ns and then left Low for the rest of the simulation. (design requires resetting of all flip-flops).

The sample test vectors shown in the attached sheet are provided in the BLOX1.TIM file as a part of the BLOX design. These test vectors can be loaded from the simulator **File** menu, using the **Load Test Vectors** option. The design outputs can be displayed as a bus (in the hex format) to clearly show the Fibonacci sequence of numbers.


Waveform Viewer 1
0.0
10ns/div
0.0
100ns
200ns
300ns
400ns
500ns
600ns
700ns
800ns
900ns
1us
1 CLOCK_IN.... B1
B FIB<0(hex)#4 1
b GSR+..... Cs
1
1
2
3
5
8
D

APN-31

How To Create Xilinx Timing Files For Simulation

If you do not set up the netlist generation command correctly (in the XACT software), then the newly generated file will have the same name as the pre-layout file and will override it. Because of that file override you may loose the previous simulation data associated with the pre-layout file.

To prevent overwriting of the pre-layout file with a post-layout one, you need to generate a file with a new file name extension. To do so, use the following procedure:

type in XACT software:
LCA2XNF - g netlist netlist.XNR

This generates a netlist with the *.XNR* extension and does not override the original pre-layout *.XNF* netlist file.

Next, type in XACT:
XNFBA netlist netlist.XNR - onetlist.BAX

The new .BAX netlist will be used by SUSIE-XILINX to simulate the timing design behavior.

Creating Hierarchical Macros From a Schematic

In order to condense schematic size, it is very often desirable to assign repetitive blocks of circuitry to *Hierarchy Symbols* referred to as *Macros*.

To create a hierarchy symbol from a schematic follow these steps:

1. Open a new project sheet and draw the schematic to be contained within the intended hierarchy symbol, taking care to end each wire with respective input, output, or bi-directional terminals from the wire menu (connector pins or net names will cause problems). Circuitry may also be copied from existing schematics using cut or copy and paste options from the edit menu.
2. From the **Hierarchy** menu, select ***Create Macro Symbol***. This selection will generate the ***Create Symbol*** window.
3. Within the ***Create Symbol*** window assign the symbols name. Add comments, such as circuit definition, ect.(optional). Symbol will have an H1, H2, ...Hxx reference designation unless otherwise specified in the reference section. Select contents to be assigned to symbol, in this case it will be a netlist. Select symbols shape, dual in line have inputs at left and outputs to the right, squares may be random. Confirm input and output or bi-directional signal assignments in the pins section. Power signals are automatically assigned to hidden pins within the netlist.
4. Click on OK, Active-CAD will now report that schematic has been saved as a *hierarchical symbol* and ask if symbol contents is to be edited.

The Macro will automatically be assigned to the project library. Using the Library Manager, this macro may be transferred to a user defined macro library.

APN-34

How To Change The Symbol Name

A symbols name is an integral part of the symbol and can be changed only by editing the symbol in the Symbol Editor. If editing is peformed on a part belonging to a system library, this part can only be saved in a user created library or in the project library. It cannot be saved back with the original part in the system library because it would destroy the integrity of the libraries supplied with the ACTIVE-CAD and SUSIE-CAD products. The system libraries are set to read only access.

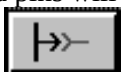
To change the symbol name follow this procedure:

1. Select the **CHANGES** mode in the schematic editor.
2. Click on the symbol to be edited, selected part will be outlined in red.
3. Select the **Symbol Editor** option from the **Options** menu, the symbol editor window will appear.
4. Activate the **EDIT** button in the symbol editor toolbox.
5. Select the symbol name to be changed by click of mouse button, (changes color when selected) and delete by pressing the **Del** key on the keyboard.
6. Select the **ABC** text button in the toolbox.
7. Click at the location within the symbol where the new symbol name should be placed; a new **Text Edit** window appears.
8. Select the **Settings** option to choose the desired font size.
9. Enter the name of the new symbol in the **Text** field and click on the **OK** button.
10. Since the new symbol cannot be saved in the system library, select the **Save As** option from the **File** menu. Thewill **Save As** window appear.
11. In the **Save As** window, enter:
    - the new symbol name in the **Symbol** field
    - the destination library name in the **Library** field. This can be the current project name or any other user defined library. This library can be directly selected from the **Libraries** field
    - click on the **OK** button to complete the Save As operation.
12. Swap the old symbol with the new one that has the corrected symbol name (see *APN-35 Swapping Schematic Parts Without Rewiring*).

APN-35

Swapping Schematic Part Symbols Without Rewiring

Replace one part symbol with another, maintaining the original schematic wiring by following this procedure:

1. Select the CHANGES mode.
2. Click on the part to be replaced. Part will be outlined in red when selected.
3. Click on the **Disconnect**  button. All disconnected wires will be marked by red crossed circles referred to as dummy terminals.
4. The disconnected part may now be **DELETE**d, **MOVE**d**,** or **DRAG**ged from the target area. Note that all wires remain at their current disconnected locations.
5. Select SYMBOL mode and place the preferred new part symbol between the dummy terminals, with as many of its pins as possible centered over disconnection points. However, any remaining non-centered pins will be of no major consequence.
6. Click on the  connect button. The red dummy terminals will disappear as the wires are directly connected to the pins of the new symbol. Any remaining disconnected wires may be dragged or MOVEd to the appropriate pin. Upon release the mouse button, the wires will connect directly to the selected pins

March 27, 1995

APN-36 How To Connect XBLOX Bus Pins to Single Wire Pins

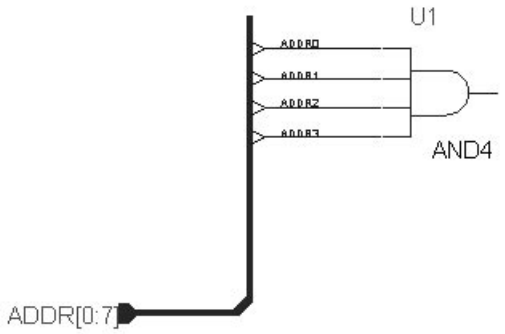
ACTIVE-CAD allows you to connect XBOLX bus pins to symbols with single-wire pins. A typical example of such connection is shown in Figure 1, where INPUTS bus output is connected directly to the single-wire input pin WR_EN. To make this connection work, follow these steps:

1. Draw INPUTS and SRAM symbols
2. Select the **Changes** mode in the Schematic Editor. This mode is used for redefining symbols, wires and pins.
3. Double-click on the INPUTS symbol to edit its pin parameters. In response, ACTIVE-CAD displays the **Edit Symbol** window, shown in Figure 2.
4. Click on the BOUNDS item in the **Parameters** field. It is instantly copied into the **Name** field.
5. Enter into the **Description** field the upper and lower bounds of the bus by typing in: *0:0*
6. Click on the **Done** button to complete the BOUNDS limit assignment.
7. Click on the ENCODING item in the **Parameters** field and enter into the **Description** field: *UBIN*. Click on the **Done** button to complete the entry. To display the selected parameters on the schematic sheet, click on the **Display All** button or double-click on the selected parameters. All parameters selected for schematic display have a double asterisk in front of the parameter name.
8. Click on the **Update** button to update the INPUTS schematic symbol. Click on **Close** to exit the setup window.
9. Select the **Wire** mode and click on the **Bus** button in the toolbox.
10. Draw a bus wire from the INPUTS output pin. Click the left mouse button to anchor the corners of the bus, if needed. Draw the bus across from the WR_EN pin of the SRAM module so that you can connect it with a bus tap, as shown in Figure 1.
11. Click the right mouse button to end the bus drawing process.
12. Click on the **Name** button in the toolbox. When the **Edit Bus** window appears, enter the destination pin name, e.g. *WR_EN*. Set both the upper and lower bus limits to *0*.
13. Click on the **Tap** button in the toolbox.
14. Click on the bus name, e.g. WR_EN, and then on the single-wire pin which has the same name. A tap is drawn between the pin and the bus.
15. Click on the **Query** button and then click on the bus. Note that the bus is connected to the INPUTS symbol pin.
16. Click on the wire tap from the WR_EN pin and note that it is connected through the bus wire as the WR_EN0 net.
17. You can generate an XNF netlist (**Options/Export** netlist option) to confirm the connectivity between the INPUTS symbol bus pin and the WR_EN single-wire pin of the SRAM symbol.

APN-37

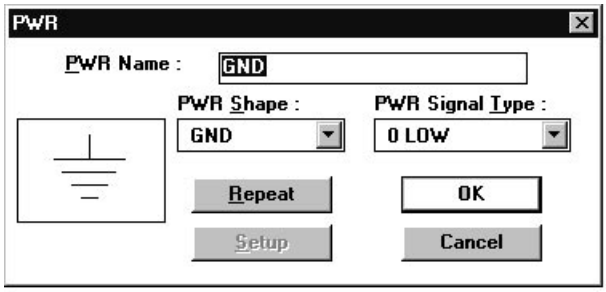
How To Connect Ground Taps To A BUS

To connect individual bus elements to ground, follow the procedure described below. The figure below shows an 8-bit bus with 4 wires connected to **U1** inputs. The remaining 4 bus wires will be connected to GROUND.

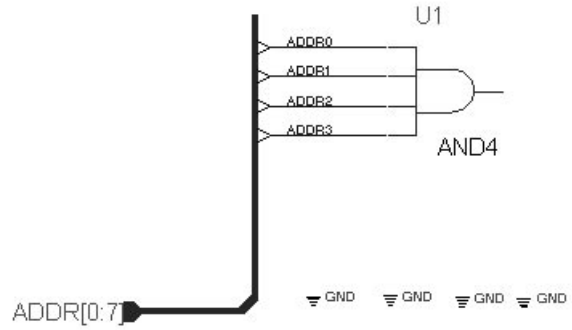


To add grounds to the *ADDR* bus:

- Select **WIRE** mode which invokes the **WIRE** toolbox.
- From the **WIRE** toolbox, select the **Wire** option and click on the **PWR** button.
- Click anywhere on the schematic, the **PWR** screen appears (see figure below), select the GND option from the **PWR Shape** field and **0 Low** option from the **PWR Signal Type** field.
- Since multiple GND signals will be connect to the bus, click on the **Repeat** button.

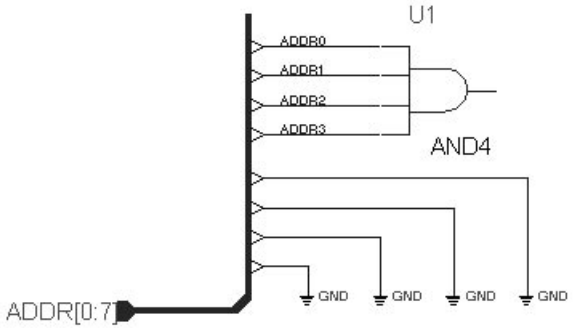


- Place four (4) GND symbols below the AND4 gate by clicking the mouse button, as shown below.



- Click on the **Wire** button in the **WIRE** toolbox and connect all the GND symbols to the bus, as show

below. Notice all bus connections to GND remain unnamed.

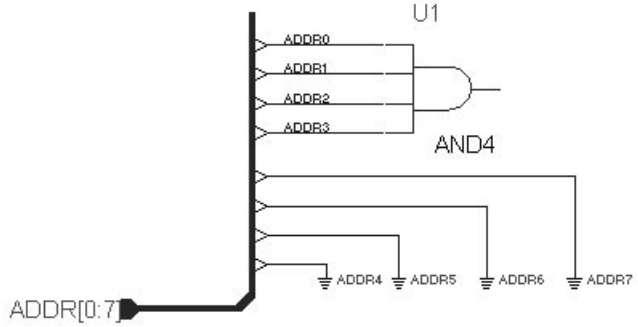


To add names to each bus tap connected to ground:

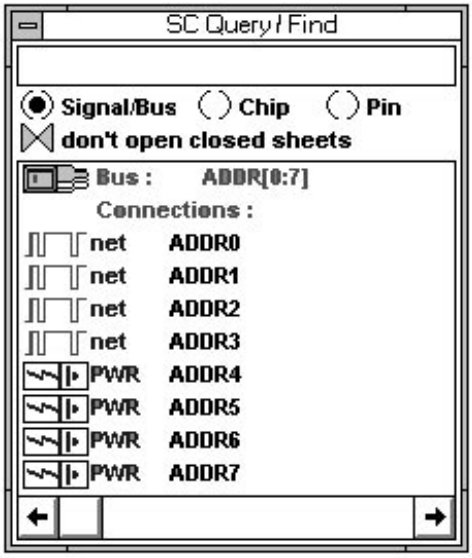
1. Click on the **Tap** button in the **WIRE** toolbox.
2. Click on the *ADDR* bus name. A blue square appears and follows each cursor movement. The top line of the ACTIVE-CAD screen reads: *Expand Bus tap: ADDR0.*

Click on the up-pointing keyboard cursor key to set this display to read ... *ADDR4*.

4. Click on the bottom-most bus tap connected to the GND symbol. Note that the associated GND name will instantly change to *ADDR4*.
5. Click on the remaining GND tap connections and notice that all GND markers will be changed to the consecutive wire tap names (*ADDR5*, *ADDR6*, and *ADDR7*), result should be screen shown below.



Activate on the **QUERY** button and then click on the *ADDR* bus. In response, ACTIVE-CAD displays all the connectivity of the *ADDR* bus as shown below.



Notice ADDR4-ADDR7 are now directly connected to GND.

Recovering library data errors:

If saving or opening of library components (e.g. Schematic macros) reports the *"LM: Error reading/writing to disk*" message, you need to verify the library integrity. If you have experienced some application errors during the library **Save** operation, some macros in the library may have inconsistent data structure. In such a case the library has to be recovered by moving all macros that have valid data to a new library and recreating the damaged macros.

LIBRARY TEST:
To verify library corruption, start the **Library Manager** and select the library being tested. Then select the **Optimize** option in the **Library** menu. If this operation fails with the message "*Error 54 - Contact Aldec*" then the library data is inconsistent and has to be repaired. Otherwise the library data is correct and the above procedure for fixing the library error can be applied.

LIBRARY REPAIR:
1. Using Windows **File Manager**, create a new sub-directory (e.g. *NEW*) in the PROJECT directory. Since you cannot edit the current (FIB) project that has library data errors, you must open another project in **Project Manager**.

2. Start the **Library Manager** and select **Create** from the **Library** menu. When asked for the library name, give it the same name as the library with the problem (FIB) and place it the *NEW* sub-directory created in Step #1.

3. Double click on the library being repaired (FIB) to get a listing of objects in the library. Select all the objects in the library (click on the first item and holding down **Shft** key click on the last item).

4. Without closing the list of objects, select the **Copy** option from the **Object** menu. Select (FIB) for the destination library,
(same name as the problem library).

5. If the operation fails with the Error 54, select the new library and double click on it to get the list of objects. The last object on that list was the last object successfully copied. The first object in the source library that follows it is damaged.

6. Select the first object after the damaged one and repeat the **Copy Objects** procedure.

7. If another error is reported, repeat the procedure for copying the library objects (Steps #5&6) and skip all the damaged objects.

8. After all valid macros have been copied to the new library, select both the new and the corrupted libraries. Then choose the **Detach** option in the **Library** menu.

9. Exit **Library Manager**.

10. Rename the LIB sub-directory to BAK and the NEW sub-directory to LIB. This way the new library will become the new project library.

11. Open the recovered project using the **Projects Manager**. This will automatically attach the new library.

12. The macros that were damaged should be recreated from their original source schematic file in the project directory (load schematic-edit-save as hierarchical macro).

Remember that the changes made to the hierarchical macros are saved in the library and not in the schematic file in the project directory. If you do frequent editing of the macros and want to save them as schematic files, use the **Save As** option in the schematic **File** menu. This will save the schematic as a top level sheet and will update the .SCH file in the project directory. Before you simulate or export the design you must remove the sheets saved this way from the project contents because they are duplicates of the macros and would be added to the netlist as library macros. To delete these extraneous schematics, open **Project Contents** in the **Hierarchy** menu and delete each schematic from the listing using the **Del** keyboard key.

Recovering library number errors:

If saving a library, macro or symbol or exporting a netlist causes *"LM: Library XXXX not found*" and LM: Error 3041 errors, it means that the internal library number is not consistent with the libraries index in the LIBDIR file. This can be caused by manually copying the library under DOS or Windows **File Manager**. The following procedure describes the steps to restore the correct library number:

1. Since you cannot operate on the current (open) project library, open any other project in order to modify the project that has a problem.

2. Start the **Library Manager** program:
- Select the library that has a problem and select the **Detach** option from the **Library** menu.
- Close the **Library Manager**.

3. Open the project that has a problem. The **Project Manager** automatically detects that the project library is not attached; It attaches it and updates all references to the new library number.

4. To restore the library number, each hierarchical macro (one that uses other macro symbols), must be saved again in the library. To save a macro, start the schematic editor and using the **Open** option from the **File** menu open or load the macro. Since the **Save** option is disabled till some changes are made to the macro, switch to the **Changes** mode and make any change in its schematic (e.g. move back-and-forth a net or symbol). The **Save** option in the **File** menu will be activated. Perform the **Save** operation. This will generate a netlist for the macro with the new project library number.

NOTE: All macros that use exclusively system libraries symbols do not have to be re-saved!!!

5. This concludes the recovery process of the internal library number.

Restoring the missing .pdf file

You have copied only the files within the project (e.g. from B:\ACTIVE\PROJECTS\FIB to C:ACTIVE\PROJECT\FIB). Because the FIB.pdf file is not copied by the DOS/Windows procedures, you will not see the FIB project name listed by **Project Manager**, despite that all its files reside within the new directory. To see the projects name, you must restore its .pdf file:

1. Select the Windows **File Manager**. Double-click on the copied project name (FIB in directory C:\ACTIVE\PROJECTS\FIB). A LIB sub-directory appears below the projects name (e.g. below FIB). Using the **Rename** option in the (**File Managers**) **File** menu, rename this LIB sub-directory to BAK. This protects your design files from being overwritten when you create a new project with the same name as the one copied under DOS/Windows.

2. Using the **New** option in **Project Manager**, create a new project (e.g. FIB) that has the same name as the project that has been copied under DOS/Windows. To simplify the process, you need to select the same design environment (e.g. Xilinx, Actel, etc.) as the copied project. The FIB project name now appears in the **Project Manager** listing.

3. Since the currently open project cannot be edited, you must open another project to modify (edit) the FIB project.

4. Go back to the Windows **File Manager** and delete the LIB directory created by **Project Manager** in step 2. Next, rename the BAK sub-directory to LIB. However, you cannot use this project yet because its address pointers are still incorrect (3803LM internal error, Libraries directory access error). We will use the ACTIVE-CAD internal operations to restore these pointers.

5. The project FIB still has wrong addresses to its libraries. To correct these addresses, detach the FIB libraries in **Project Libraries**:

- Select **Library Manager**
- Click on the FIB library in the libraries listing
- Select the **Detach** option in the **Library** menu of **Library Manager**

6. Open the recovered (e.g. FIB) project.

7. Verify the project libraries in the **Project Libraries** option. If any libraries are missing, you may add them to the project at this point.

8. Start the schematic editor.

9. If any missing symbols or errors were reported on the schematic, **DO NOT (!!!)** save the schematic (changes) but again verify the **Project Library** list (repeat Step #7).

10. Open the **Project Contents** window from the schematic editors **Hierarchy** menu.

11. You will see a list of schematic sheets with the old directory paths from which they were copied under DOS/Windows (e.g. . b:\ACTIVE\PROJECTS\FIB). Select the **Add Sheets to Project** option from the **Hierarchy** menu.

12. Using the **Browse** button add all the sheets from their new location

(e.g. C:\ACTIVE\PROJECTS\FIB), and close the **Add Sheets to Project** window.

13. Open the **Project Contents** window. You should now see both the old and new schematic sheets directories. Click on the wrong schematic sheets directory, e.g. b:\ACTIVE\PROJECTS\FIB and delete it.

You can also individually delete each schematic sheet directory by clicking on each line that has the wrong path and pressing the **Del** keyboard key. The message "Error reading file" should be displayed in the status line.

14. Close the **Project Contents** window. Select the **Update Hierarchy** option from the **Hierarchy** window.

15. Select the **Project Contents** option once more and verify that the project structure contains all desired sheets and macros.

17. This concludes the design recovery process. You can now create a netlist from the **Options** menu and activate a simulator or printed circuit layout software, etc.

BTRIEVE Initialization Problems:

After some debugging of customer problems with Btrieve initialization errors we have found that changing the BTI.INI file can help. The following change will allow the Btrieve library engine to run with the low amount of conventional DOS memory:

[Btrieve Client]
options=/L:20 /F:80 /H:200 /T:15 /M:512 /U:64

NOTE: previously there was "/U:200".

The BTI.INI file is located in the WINDOWS directory.

This change applies to all ACTIVE-CAD, SUSIE-CAD, and DEMO versions 1.87 and later.

The above change should be applied with all installations that report Btrieve errors upon starting the Project Manager or any other ACTIVE-CAD application.

APN-45 How To Use XABEL for XC7000 family

ACTIVE-CAD 1.9 improperly handles XABEL designs for the XC7000 family.
Users who report this error should obtain an updated XABEL interface for ALDEC BBS. The file name is ABEL7000.zip. The file contains updated XABEL interface and a sample design called TESTPLD with two ABEL macros.

(Corrections to the ALDECs XILINX Applications manual, Rev.1.9.1).

The changes start in chapter Using XABEL, after point 4. of the translation procedure:

5. The ABL2PLD program of the XACT software runs in a text window (7000 EPLD designs). If the translation was successful a symbol will be generated and saved in the project library. If the converter did not run successfully, an error report will be loaded into the Notepad.

6. The ABL2PLD generated PLD file is automatically converted to a custom PLD symbol which can be placed on the schematic. The new symbol has an attached "PLD=filename" attribute.

7. The new symbol, which has the same name as the Abel file, is displayed in the symbol list (**SC Symbols** window) and you can place it on the schematic.

8. The schematic symbol is created based on the *.XSF file generated by XABEL. This file lists input and output pins of your macro. The symbol generated automatically will have input pins on the left and output pins on the right.

9. To edit the symbol created automatically, you need to highlight it on the schematic and select the Symbol Editor option from the Schematic Options menu. After editing and saving the symbol, the changes will be updated on the schematic.

10. To update a macro created with XABEL, you need to select the XABEL button in the Flow manager and recompile the ABL file. **NOTE** that if the I/O pins have changed, the old symbol will be replaced with the newly generated one.

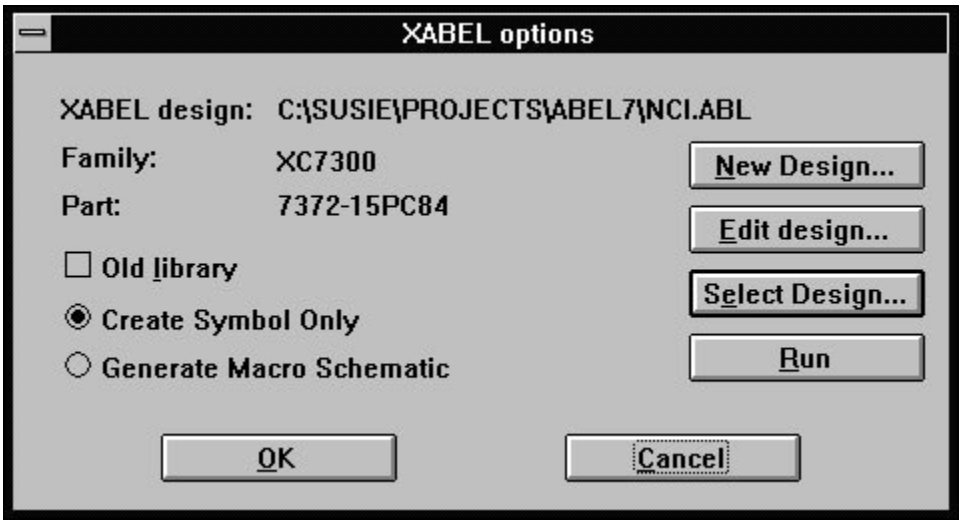
NOTES: The following restrictions apply to using XABEL using the XC7000 family:

- **Your ABEL filename and the MODULE section inside ABEL file must be the same.**

- **Pre-layout simulation from the schematic is not available. You need to rune Place and Route and then go to timing simulation to simulate the design with XABEL sections. Then you can select Timing or Functional simulation mode.**

XABEL Options:

The following window shows options for the XABEL program:



- **Old Library** setting will create netlist compatible with older versions of XACT Software.
- **Create Symbol Only** option will generate the symbol for the XABEL macro.
- **Generate Macro Schematic** will not work with XC7000 family, because ABL2PLD does not generate XNF netlist that can be used to create a macro schematic.
- **Run** will invoke the ABL2PLD program

APN-46 Using I/O terminals in FPGA designs

ACTIVE-CAD Schematic editor has a built in symbols called I/O terminals. These symbols are sometimes misused in FPGA designs and cause various problems with design compilation.

Top level schematics
The I/O terminals on the top level schematics can only be used to specify external device pins. Each I/O terminal is specified in the FPGA netlist exported for compilation as I/O PAD and will be assigned a device pin number by the FPGA place and route tools.

You cannot use I/O terminals as page to page connections. The regular net names on all schematics will be joined together in the design netlist, so you do not have to use any special symbol to indicate that the signals are connected across different schematic pages.

Hierarchical macros
The I/O terminals in hierarchical macros are used to specify external pin of that macro. This means that each I/O terminal will be converted to a symbol pin of that macro. To specify the external pin inside of the macro you cannot use I/O terminal symbol. Instead, you have to use the PAD symbol from the FPGA library. For example, in Xilinx designs, you can use symbols: IPAD, OPAD, IPAD8, IPAD16, etc.

Terminal direction
The direction of I/O terminal is crucial for the simulator. If you use the input terminal on the schematic macro, the simulator will only allow the signals to be passed into the macro. Any signal generated inside the macro will not be output outside that macro. Also when you use an output terminal, no signals from outside the macro will be propagated inside the macro. You can use bidirectional terminals to specify that the signal are propagated both inside and outside the macro.

Bus terminals
You can use the bus terminals on buses. The bus terminal serves the same purpose as the single I/O terminal. The Bus terminals can be used to specify external device pins on the top level in the design, but the pin numbers will not be backannotated (displayed) on the schematic. To specify pin numbers on the bus pin, you need to assign the net attributes to the individual nets, e.g. by using bus taps.

APN-47 Defining pin numbers for bus pins

Symbols with bus pins are typically used in FPGA designs to connect entire data or address bus into a macro without drawing individual signal lines. In FPGA designs there is no need to define pin numbers for the macros, because FPGA netlists use pin names and not pin numbers for hierarchical macro connections.

In board designs, however, you can also use bus pins in some symbols to minimize the number of individual pins. When creating symbols with the Symbol Editor, there is no option to assign specific numbers to the individual pins of a bus pin.

To specify the pin number of bus pins follow this procedure:

1. In the Symbol Editor specify one pin number for the bus pin e.g. 12. The symbol editor will increment that number for each single pin of the bus pin (12, 13, 14, 15, ...)
2. In the Library Manager, select the library (double-click) and the symbol (single click). Then without closing the library listing window select the Pin Numbering option from the Object Menu. This option allows to change the individual pin numbers for each single pin.

Future version of the Symbol Editor should also allow to edit individual pin numbers for bus pins.