NX-HILO™
User's Manual

October 1985

p-cad™
PERSONAL CAD SYSTEMS INC.

000-0076-01

Personal CAD Systems, Inc. provides this manual "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. P-CAD may make improvements and/or changes in the product(s) and/or the program(s) described in this manual at any time and without notice.

Although P-CAD has gone to great effort to verify the integrity of the information herein, this publication could contain technical inaccuracies or typographical errors. Changes are periodically made to the information herein. These changes will be incorporated in new editions of this publication.

ABOUT THIS MANUAL

This manual introduces you to NX-HILO, a netlist translation program that converts P-CAD netlist database files to files in the format of the GenRad HILO logic simulator.

Chapter 1, Introduction, provides an overview of NX-HILO and instructions for program installation.

Chapter 2, Preparing the Schematic Database, gives instructions for using PC-CAPS to create the schematic database that is compatible with NX-HILO.

Chapter 3, Preparing the Input Files, gives instructions for creating the files needed for NX-HILO.

Chapter 4, Using NX-HILO, gives instructions for configuring and running NX-HILO.

Chapter 5, Viewing the Output Files, gives instructions for viewing, printing, and interpreting the NX-HILO output files.

The appendixes show symbols supplied with NX-HILO, a summary of PC-CAPS attributes used with NX-HILO, error messages, and output files for sample circuits.

NOTATION

The operating instructions in this manual use the following notation.

<xxxxx> Angle brackets around lowercase letters indicate variable names that may be entered by the system or by you.

UPPER Uppercase letters indicate a command name or an element that must be typed as shown.

[ ] Square brackets set off the names of keys. For example:

[Return]

[ ] - [ ] Square brackets connected with a hyphen set off names of keys to be pressed simultaneously. For example:

Press [Ctrl] - [Alt] - [Del]

/ A forward slash separates main menu and submenu command combinations. For example:

DRAW/ARC

*   An asterisk in a filename or in a filename extension indicates that any character(s) can occupy that position and all the remaining positions in the filename or extension. For example:

```
DIR *.SYM
```

displays a list of all the filenames with the extension .SYM in the current directory.

(.NNN)   Parentheses around a period and up to three uppercase letters indicate a preferred or default filename extension. For example:

The layer structure file (.SCH) is in the SYM directory.

TESTFILE   TESTFILE is a sample filename, which you must replace with the filename you intend to use. For example:

```
Database Filename   : TESTFILE.SCH
Netlist Filename    : TESTFILE.NLT
```

TESTFILE is used here as a sample input to show where you should enter your filename.

CONTENTS

FIGURES

TABLES

CHAPTER 1 - INTRODUCTION

NX-HILO converts a binary netlist produced by PC-NODES from a PC-CAPS schematic database into a format compatible with GenRad's HILO logic simulation program. NX-HILO is compatible with both HILO-2 and HILO-3, and fully supports multi-sheet and hierarchical designs.

OVERVIEW

NX-HILO, together with PC-NODES, enables you to extract information from a PC-CAPS created schematic database and produce a HILO-compatible file.

There are five stages in this process:

1. Using PC-CAPS, you prepare the schematic design, which consists of one or more schematic files (.SCH). Each schematic file is created by interconnecting symbols (.SYM and .SCH files). The types of symbols you can use in your schematic are described in Chapter 2, "Preparing the Schematic Database."

2. Using PC-NODES, you extract a binary netlist file (.NLT) from each schematic file in your design. Netlist files contain non-graphical component and interconnection information.

3. If your design contains more than one schematic file and you want to translate all the files at one time, you next use a text editor to create a batch control file (.BCF). This file contains the names of all netlist files to be processed. NX-HILO will repeatedly read the batch control file and process the next netlist file until all the files have been translated.

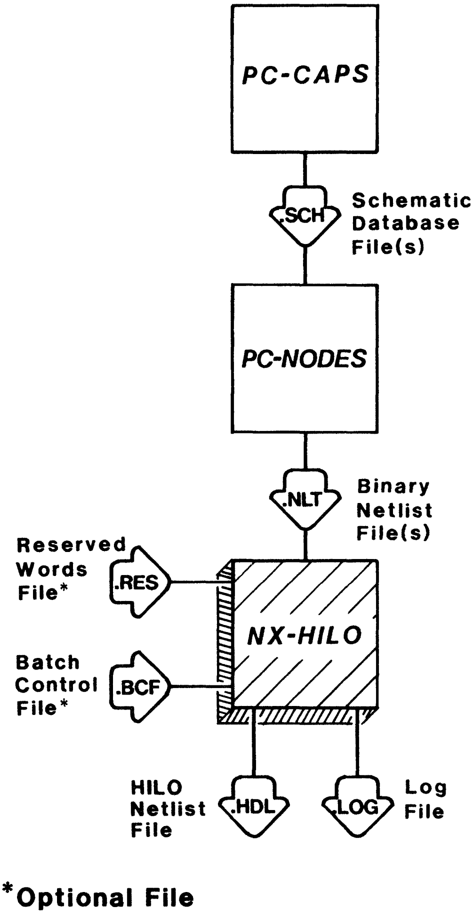
4. If you want to add or delete reserved words, which NX-HILO will check for in your input file(s), you edit the reserved words file.

5. You then input the netlist file or the batch file into NX-HILO. NX-HILO uses the netlist(s) to produce two files:

   o **The HILO Netlist File (.HDL)** is a HILO-format netlist file compatible with HILO.

   o **The NX-HILO Log File (.LOG)** contains all messages displayed during the execution of NX-HILO. The log file has the same filename as the HILO output file but has the extension .LOG. This file is useful for examining any errors that are reported by NX-HILO.

Figure 1-1 shows the process of creating a HILO net list from PC-CAPS created schematics.



85023

Figure 1-1. NX-HILO System Overview

SYSTEM REQUIREMENTS

Before you install NX-HILO, your computer system must have the following minimum configuration:

- IBM PC, XT, AT, TI PC, Tandy TRS 2000, or equivalent
- 640k of RAM
- Color or monochrome monitor
- DOS 2.0 or higher operating system
- CONFIG.SYS in the root directory, containing the following text:

```
BUFFERS=12
FILES=15
```

INSTALLATION

Your NX-HILO diskette contains the following files:

Program file:

NXHILO.EXE

Reserved Words file:

NXHILO.RES

Symbol files:

NOT.SYM
NOT2.SYM
BUF.SYM
BUF2.SYM
NAND2.SYM
NAND3.SYM
NAND4.SYM
NAND5.SYM
NAND6.SYM
NAND7.SYM
NAND8.SYM
NOR2.SYM
NOR3.SYM
NOR4.SYM
NOR5.SYM
NOR6.SYM
NOR7.SYM
NOR8.SYM
AND2.SYM
AND3.SYM
AND4.SYM
AND5.SYM
AND6.SYM
AND7.SYM
AND8.SYM
OR2.SYM
OR3.SYM
OR4.SYM
OR5.SYM
OR6.SYM
OR7.SYM
OR8.SYM
NOTIF0.SYM
NOTIF1.SYM
BUFIF0.SYM
BUFIF1.SYM
MOVEIF0.SYM
MOVEIF1.SYM
TRANIF0.SYM
TRANIF1.SYM
CAP.SYM
CLOCK0.SYM
CLOCK1.SYM
BALR.SYM
PIN$IN.SYM
PIN$OUT.SYM
PIN$IO.SYM

Sample data files:

MOSFF.SCH
HALFADD.FUN
ADDER.SCH
4ADDER.SCH
BCDADDER.SCH
BCDADD2.SCH
BCDADDER.BCF

To install NX-HILO on your hard disk, insert the diskette in Drive A and use the following procedures. These procedures assume that you are using the P-CAD recommended directory structure.

Install NXHILO.EXE in your \PCAD\EXE directory. To do this, first change to the \PCAD\EXE directory by typing:

```
CD \PCAD\EXE  [Return]
```

Copy the file by typing:

```
COPY A:NXHILO.EXE  [Return]
```

Next, create a new directory for the HILO symbol library by typing:

```
MD \PCAD\SYM\HILO  [Return]
```

Install the symbol library in the new directory. First, change directories by typing:

```
CD \PCAD\SYM\HILO  [Return]
```

Then copy the files by typing:

```
COPY A:*.SYM  [Return]
```

Copy the .RES file and the sample data files to your working project directory. For example, if you are using the PROJ0 directory, change to that directory by typing:

```
CD \PCAD\PROJ0  [Return]
```

Then copy the files by typing:

```
COPY A:NXHILO.RES  [Return]

COPY A:*.SCH  [Return]

COPY A:HALFADD.FUN  HALFADD.SYM [Return]

COPY A:BCDADDER.BCF  [Return]
```

CHAPTER 2 - PREPARING THE SCHEMATIC DATABASE

Before running NX-HILO, you create a schematic database using PC-CAPS. This section describes PC-CAPS schematic structure and the features you must use to construct a HILO-compatible schematic. To ensure that your database is compatible with HILO, read this section before constructing your schematic.

SCHEMATIC DATABASE STRUCTURE

The PC-CAPS schematic consists of an interconnected group of pre-defined symbols. Each symbol is a logical and pictorial representation of a component.

With NX-HILO, P-CAD supplies a library of symbols that are compatible with HILO. You can use these symbols as they are, modify them, or create your own symbols. Before modifying or creating symbols, read the HILO requirements in the following sections and Appendix B, "Symbol Summary."

Hierarchical Structure

Hierarchical structure is the usage of symbols to represent subcircuits of a schematic. With PC-CAPS, you can design a symbol to identify a circuit, then use the symbol to represent the circuit in another schematic. PC-CAPS allows up to 15 levels of hierarchy. NX-HILO fully supports hierarchical structure.

Multi-Sheet Structure

A multi-sheet schematic is created by producing a number of single-sheet schematics. You indicate connections between sheets by giving connected wires the same net name on each sheet. NX-HILO allows up to 100 sheets in a multi-sheet schematic.

NX-HILO supports multi-sheet schematics at all levels of the design hierarchy. You must use NX-HILO batch mode to process a multi-sheet schematic.

SPECIAL PC-CAPS INPUT FOR NX-HILO

As you assemble your schematic database, you must input certain information for NX-HILO. This information and the PC-CAPS commands used to enter it are described in the following sections.

Creating HILO-Compatible Symbols

Appendix A, "NX-HILO Schematic Symbols," describes the symbols provided with NX-HILO. You can create more symbols as you need them, providing you assign the appropriate Component Type ID (summarized in Appendix B).

After you create a symbol, use the SCMD/SCAT command in PC-CAPS SYMB mode to assign the symbol's Component Type ID. Use the ENTR/PIN command in SYMB mode to enter each pin for a symbol. NX-HILO assigns pin order in the order you enter the pins.

You can create three types of symbols for use with NX-HILO:

- Symbols for HILO primitives. When creating one of these symbols, use one of the symbols provided with NX-HILO as a guide for graphics and pin order. By P-CAD convention, each symbol file has a .SYM filename extension.

- Symbols for HILO functional models. Create these symbols in the same way as primitive symbols, using the .SYM filename extension. You must give the symbol the same name as the corresponding HILO model, either as the filename or by assigning the NAME attribute.

- Symbols for hierarchical subcircuits. Create subcircuit symbols in PC-CAPS SYMB mode. Assign a Component Type ID of 255. This type of file has an .SCH filename extension.

Entering Terminal Pins

For all schematics, you must identify the terminal pins and their order for NX-HILO. You can specify terminals in either of two ways:

- Create a symbol for the entire schematic in PC-CAPS SYMB mode in the same way you create a subcircuit symbol. Use the ENTR/PIN command to enter pin information. NX-HILO will assign pin order in the order you enter the pins. The default Component Type ID for a schematic with a symbol is 255, so you do not need to assign a Component Type ID.

  For a multi-sheet schematic, create the symbol on the first page only, as described in the next section.

- Use the special pin component symbols supplied with NX-HILO and described in Appendix A. Use the ENTR/COMP command in DETL mode to attach these pin symbols to the terminal wires. These symbols indicate pin type (input, output, i/o).

  When you use these pin symbols, NX-HILO uses the ORD attribute to determine pin order. Each pin symbol has a preset ORD value of 0. Terminal pins are ordered in ascending sequence beginning with 1. Use the ATTR/SCHG command in DETL mode to change the ORD attribute for each pin on the schematic. The ORD value must be a number. For example, you would give the first pin symbol an ORD value of 1 by entering:

  ```
  ORD=1
  ```

  If you do not change the ORD value from 0 or if you assign an invalid ORD value (i.e., non-numeric, duplicate, or less than 1), NX-HILO will produce an error message and will arbitrarily assign pin order.

Handling Multi-Sheet Schematics

NX-HILO requires the information described below for multi-sheet schematics.

Sheet Identification

To indicate that a schematic is a single sheet of a multi-sheet design, assign the SHEET attribute to the sheet. In SYMB mode, use the ATTR/ACOM command and enter, anywhere on the sheet, the following attribute:

```
SHEET=<sheet id>
```

Each sheet must have a unique <sheet id> identifier . NX-HILO requires the identifier to be alphanumeric and to be no more than three characters long. For example, you could assign the following attributes to three sheets that constitute one design:

```
SHEET=01
SHEET=02
SHEET=03
```

NOTE: Use the same number of characters in the identifiers for all sheets, using leading zeroes when necessary; the system cannot tell the difference between 1, 10, and 100; 2, 20, and 200; etc.

Hierarchically Embedded Sheets

If a multi-sheet schematic is embedded within a hierarchical design, you must represent it with a symbol. To do this, use one sheet as the primary sheet and include in it the symbol for the entire multi-sheet module. When placing the symbol on the schematic, use the filename of the primary sheet.

> NOTE: The primary sheet must be first in the list of sheet names included in the batch control file that you submit to NX-HILO.

Top Level Sheets

A multi-sheet module at the top of the design hierarchy does not need a symbol. However, you still need to specify terminal pins, as explained above. If you create a symbol for the schematic and enter pins using the ENTR/PIN command, you must create a symbol for the entire multi-sheet module, as you do for hierarchically embedded sheets. If you use the HILO pin symbols, you do not need a symbol for the module and you can locate the pin symbols on any sheet or combination of sheets.

Assigning NX-HILO Attributes

NX-HILO uses several attributes as information in the .HDL output file. These attributes are explained below and summarized in Appendix C, "Attribute Summary." The list below does not include the SHEET or ORD attributes, which are explained in the previous sections.

None of the attributes listed is required. If you do not assign the attribute, the corresponding field in the .HDL file will be empty, unless otherwise noted below.

To assign attributes, use the PC-CAPS ATTR/ACOM command in either SYMB or DETL mode as specified below.

Attributes of a Schematic

You can use the following attributes for any circuit or subcircuit. Assign these attributes to the schematic in SYMB mode. If the circuit is multi-sheet, assign these attributes to the primary sheet only.

Each of these attributes corresponds to a field of the circuit header section of the .HDL output file.

NAME - Can be used to specify the <circuit name> field of the circuit header. If it is not present, NX-HILO uses the filename of the circuit (without the extension). When the symbol is an element in a higher-level circuit, this attribute also specifies the <element definition> field of the element declaration.

This attribute is useful if the desired name does not conform to DOS filename conventions.

Example: NAME=VERYLONGNAME

The NAME attribute can also be assigned to a primitive or functional model symbol and is also used in the element declaration section of the .HDL file.

TYPE - Specifies the <type> field of the circuit header. It must be IC, PCB, or CCT. If you do not enter the attribute, the type will be CCT. NX-HILO does not check the value you enter, so you must be sure it is valid.

Example: TYPE=IC

TECH - (Used by HILO-3 only.) Specifies the <technology> (default strength) field of the circuit header. The HILO values are TTL, TTLOC, ECL, MOS, PMOS, NMOS, or CMOS. NX-HILO does not check the value you enter, so you must be sure it is valid.

Example: TECH=ECL

DLY - Specifies the <delay scale> field of the circuit header. The HILO values are ns 1, ns 10, ns 100, ps 1, ps 10, and ps 100. NX-HILO does not check the value you enter, so you must be sure it is valid. Note that because of the space within the value, you must enclose it in quotation marks.

Example: DLY="ns 10"

PARM1 - Specifies the <timing> values (delay parameters) of the circuit header for a subcircuit. You must separate values with a comma. If you use any spaces in the attribute, you must enclose the values in quotation marks.

Example: PARM1=a,b,c

PARMn - Continues the delay parameters from PARM1 if necessary. The PC-CAPS ATTR/ACOM command allows 39 characters, including the attribute keyword and the equal sign. If you must specify more parameters than fit in the PARM1 attribute, use PARM2 to continue the parameters, then PARM3, and so on, up to a maximum n value of 99. If the parameters are continued in the next PARMn attribute, NX-HILO inserts a comma.

Example: PARM1="a, b, c, d, e"
PARM2="f, g"

yields (a,b,c,d,e,f,g)

The numbers used with PARM must be sequential. NX-HILO looks for the next number in sequence; if the numbers are not sequential, some values will be ignored.

Example: PARM1=a,b,c
PARM2=d,e
PARM4=f,g

yields (a,b,c,d,e)

Attributes of Schematic Elements

The following attributes supply information about a component or hierarchical symbol in a circuit. Assign these attributes to the component in DETL mode of the schematic.

Each of these attributes corresponds to a field in the element declaration section of the .HDL output file.

PARM1...PARMn - These attributes specify the <timing> parameters in the element declaration section. Assign these attributes as described in the previous section, but use actual delay values instead of symbolic names. These values will be used for this occurrence of the symbol.

Example: PARM1="1, 2"
PARM2=3
PARM3=4

yields (1,2,3,4)

You can also use PARM1 to assign a capacitance value to CAPACITOR primitives.

HI and LO - (Used by HILO-3 only.) These attributes specify <strength> values in the element declaration section. HI specifies high logic level and LO specifies low logic level.

Example: HI=STRONG
LO=WEAK

Wire Declaration Attributes

The following attributes supply information about wires in a circuit.

Because PC-CAPS does not allow you to assign an attribute to a wire, you must assign the attributes to any component connected to the wire. Be careful to declare each wire only once, for example, at one component only.

Assign these attributes to schematic components in PC-CAPS DETL mode.

Each of these attributes corresponds to a field in the wire declaration section of the .HDL output file.

W$<pinname> - Specifies the HILO wire type of the wire. A wire will be declared in the .HDL file only if this attribute is assigned. <pinname> is the name of the component pin to which the wire is attached.

If the <pinname> is omitted, NX-HILO assumes the first output pin in the component symbol's pin order. P-CAD recommends omitting the pin name only if the symbol has only one output pin.

D$<pinname> - Specifies the HILO delay parameters. Parentheses around the list of parameters are not required; NX-HILO adds the parentheses in the .HDL file.

<pinname> is the name of the component pin to which the wire is attached. It is treated the same way as the W$ attribute.

Example of wire declaration attributes:

> Component U1 has input pins named A, B, and C and a single output pin Y. The input pins are connected to wires W1, W2, and W3, respectively, and the output pin is attached to W4.
>
> The following attributes are assigned to U1 using the ATTR/ACOM command in PC-CAPS DETL mode:

```
    W$A=TRI          D$A=2,4
    W$C=UNID         D$=5,7
    W$=UNID
```

When NX-HILO is run, the following wire declarations are produced:

```
TRI (2,4) W1
UNID W3
UNID (5,7) W4
```

The attributes above produced no declaration for wire W2; however, wire W2 could be declared if the attribute(s) were assigned to another component attached to it.

CHAPTER 3 - PREPARING THE INPUT FILES

After your schematic is completed, you must use PC-NODES to extract binary netlists for each sheet and for each hierarchical element.

If your schematic consists of just one schematic file, you can input the netlist directly into NX-HILO, or you can use a batch control file to process it with other files.

If the schematic is hierarchical and you want to submit the entire circuit, or if the schematic has multiple sheets, you must create a batch control file to input all the netlists to NX-HILO.

NX-HILO can check your input file(s) for HILO reserved words, which HILO will not accept. The words NX-HILO checks for are listed in the reserved words file, which you can edit to suit your requirements.

The following sections contain instructions for creating the binary netlist and the batch control file and for editing the reserved words file.

EXTRACTING THE BINARY NETLISTS

Run PC-NODES to extract binary netlists from each sheet or hierarchical element of your schematic. Use the instructions in the *PC-NODES User's Manual.*

Do not use PC-LINK to link the netlists; NX-HILO must process the PC-NODES netlist for each sheet or subcircuit.

CREATING THE BATCH CONTROL FILE

The batch control file lists all the files to be processed during an NX-HILO batch mode session. Use a text editor such as the DOS EDLIN program to create the file. You can give the file any name you choose, but the extension must be .BCF.

List the names of the netlists in the order you want them processed. If all the netlists are to go into one output file, you must use HILO conventions for the proper order of the modules.

You do not need to include the .NLT extension; NX-HILO assumes it as the default.

For multi-sheet designs, list all the filenames on the same line, with one space between filenames. If the names do not all fit on the same line, enter a plus (+) at the end of the line to be continued. The maximum line length is 80 characters.

NX-HILO considers all text to the right of and including a percent symbol (%) to be a comment.

Sample .BCF File:

```
% NX-HILO batch file example
%
MODULE1                    % filenames of single-
MODULE2                    % sheet modules
MODULE3 3SH2 3SH3          % three-sheet module
MODULE4 4SH2 4SH3 +
 4SH4                      % four-sheet module
MODULE5                    % single-sheet module
```

EDITING THE RESERVED WORDS FILE

HILO does not allow the use of certain reserved words. The NX-HILO reserved words file ensures that your .HDL output file does not include any of these words.

When you run NX-HILO, it looks for the NXHILO.RES file in the current directory. If the file is present, NX-HILO checks each word in the file against your input netlist and produces an error message if a reserved word is used as the name of a component, a component type, or a wire.

If the reserved words file is not present in the current directory, NX-HILO does not check for reserved words.

The reserved words file supplied with NX-HILO, NXHILO.RES, includes the standard set of HILO reserved words. You can add and delete words from this file using a text editor.

If you have specified additional HILO reserved words, you may want to add them to the reserved words file. The more words you have in this file, the longer program operation will take. If you avoid the use of reserved words when designing your circuits, you may want to shorten or delete this file.

The reserved words file does not require any special format; every word in the file is considered a reserved word, and comments are not supported. For your convenience, P-CAD recommends listing one word per line, as in the following example:

```
AND
ELSE
X
Z
0
1
```

CHAPTER 4 - USING NX-HILO

This chapter describes the required conditions and procedures for running NX-HILO.

Before you run NX-HILO, be sure that:

- The system is correctly configured and NX-HILO is installed.

- You have assembled the schematic circuit.

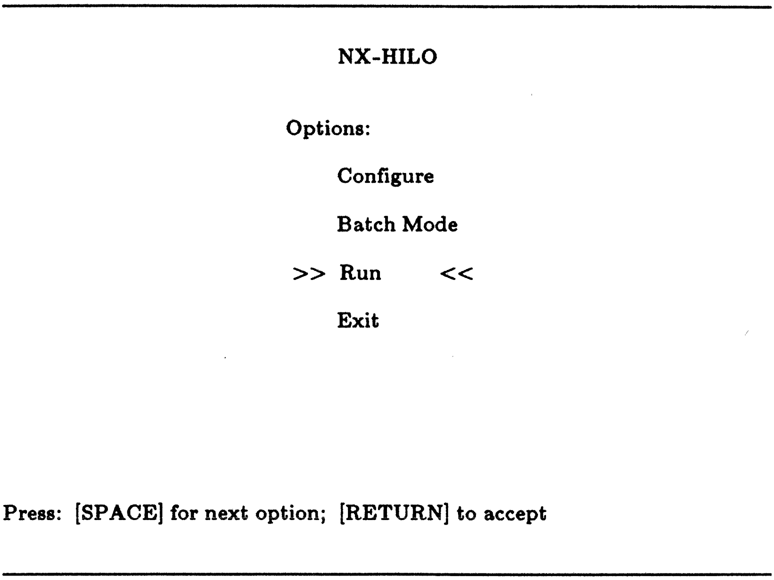
- You have extracted the netlist files and the netlists are present in the current directory.

- You have created a batch control file, if necessary, and it is present in the current directory.

- If you are using the reserved words file, it is present in the current directory.

To start NX-HILO, use the following steps:

1. From the appropriate project directory, type NXHILO and press [Return]. The NX-HILO title screen is displayed. Press any key to continue. The system displays the opening menu, shown in Figure 4-1.



Press: [SPACE] for next option; [RETURN] to accept

Figure 4-1. NX-HILO Opening Menu

This screen provides four options.

Configure - Allows you to set or change NX-HILO configuration options.

Batch Mode - Allows you to specify a batch control file, then processes all the files in the batch.

Run - Lets you begin file processing when you want to process only one file.

Exit - Returns you to DOS.

The default option, "Run", is highlighted. You can use the space bar to move to the next option or use the cursor keys to move up or down between options.

2. If you want to set or change the configuration, use the next procedure. If you do not want to set or change the configuration, continue to "Running Batch Mode" or "Running NX-HILO."

CONFIGURING NX-HILO

"Configure" allows you to set certain standard operating parameters. You can use a configuration for just one operating session or you can save it to the configuration file, to which NX-HILO refers for the default parameters.

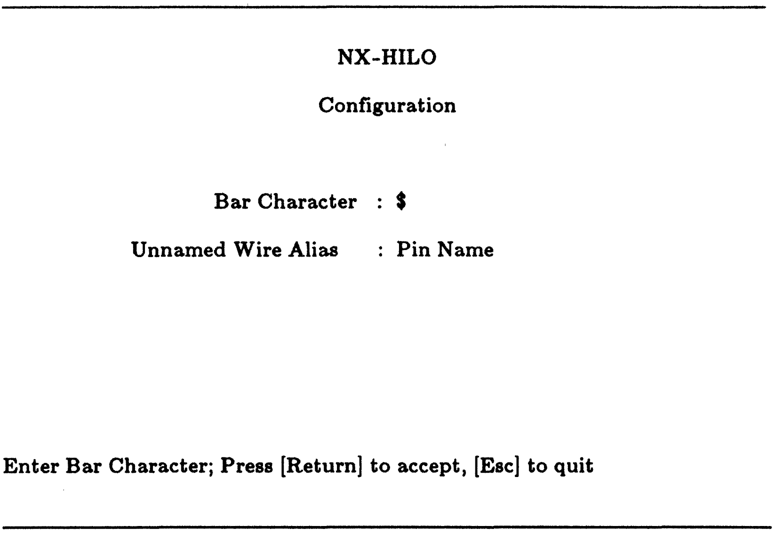
Select "Configure" at the opening menu to specify a new configuration for a program session or to change the defaults in the configuration file.

To configure NX-HILO, use the following steps.

1. At the NX-HILO opening menu, press the space bar or use the cursor key to select "Configure", then press [Return].

   The system displays the configuration screen with the current values, as shown in Figure 4-2.

   You can now accept or change the configuration values.



```
                         NX-HILO

                      Configuration


             Bar Character  : $

     Unnamed Wire Alias     : Pin Name



Enter Bar Character; Press [Return] to accept, [Esc] to quit
```

Figure 4-2. NX-HILO Configuration Screen

2. The first prompt asks for the Bar Character. In PC-CAPS, any name terminated with an apostrophe (') is displayed with a bar over the entire name. NX-HILO supports bar characters for wire names only.

   The apostrophe is not a valid HILO character. Valid characters are letters, numbers, and the dollar sign ($).

   Enter the character that you want to substitute for the apostrophe. You can press [Return] to accept the $ default.

> NOTE: Early versions of HILO-2 do not recognize the $ character, so if you want to use the $, make sure that your version accepts it.

3. The next prompt asks for the Unnamed Wire Alias.

   To assign names to unnamed nodes in a circuit, PC-NODES uses the format UNsssnnn, where sss is the sheet identification string, and nnn is a sequence of digits starting at 000.

   NX-HILO can use the PC-NODES format or the following convention: the name of the component that is the source for the unnamed wire, followed by a dollar sign ($), then either the name or the number of the component pin that sources the wire.

   > NOTE: If you are using an early version of HILO-2 that does not recognize $ as a valid character, you can not use an alias; you must use the PC-NODES convention.

   This configuration option allows the user to choose between the pin name format, the pin number format, and the PC-NODES convention. The default is "Pin Name." The alternate choices are "Pin Number" and "No Alias." You can press the space bar to cycle through the three options. When the option you want is displayed, press [Return] to select it.

   > NOTE: Any unnamed wire that is not connected to an output pin cannot be given an alias.

4. When you have set both parameters, the system displays the update prompt at the bottom of the screen.

Configuration File : Update

Press: [SPACE] for the next option; [RETURN] to accept

Press [Return] to accept the default of "Update", or press the space bar to display the "No Update" option, then press [Return] to accept it.

The "Update" option saves the configuration to the configuration file. The "No Update" option causes the onscreen configuration to be used for the current session only.

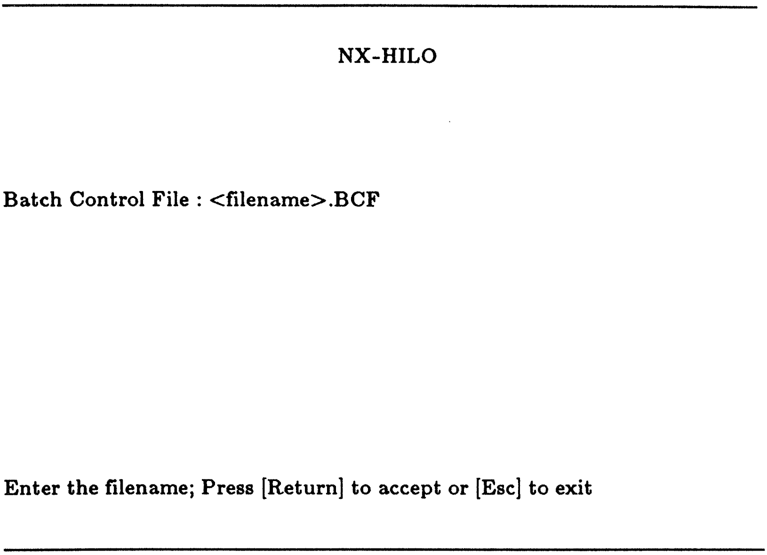
NOTE: Remember that the configuration file must be in the current directory.

When configuration is complete, the program returns to the opening menu.

RUNNING BATCH MODE

You must use batch mode for processing multi-sheet designs. Batch mode also allows you to make multiple runs of NX-HILO automatically.

1. Select "Batch Mode" at the opening menu. The system prompts you for the name of the batch control file, as shown in Figure 4-3.



NX-HILO

Batch Control File : <filename>.BCF

Enter the filename; Press [Return] to accept or [Esc] to exit

Figure 4-3. NX-HILO Batch Mode Screen

2. Enter the name of the batch control file. You do not need to enter the extension, which must be .BCF.

3. The next prompt asks:

   One Output File?

   If you answer yes, NX-HILO will put the translated data from all the input files into one output file. If you answer no, NX-HILO will put the translated data from each input file into a separate output file (except with a multi-sheet schematic, which will produce a single output file).

   The default is "No." To display the "Yes" option, press the space bar. When the option you want is displayed, press [Return] to accept it.

4. If you selected "Yes" in step 3, another prompt appears:

   Output File Name :

   The default output filename is the name of the batch control file with a .HDL extension. You can press [Return] to accept it or type the name you want and press [Return].

   NOTE: If you answered no to the "One Output File" prompt, NX-HILO takes the names of the output files from the names of the .NLT input files, substituting the .HDL extension. For multi-sheet modules, NX-HILO will use the name of the first sheet.

   When you have answered all the prompts, NX-HILO begins processing the first .NLT file. After it processes each file, it reads the batch control file to find the name of the next file to be processed.

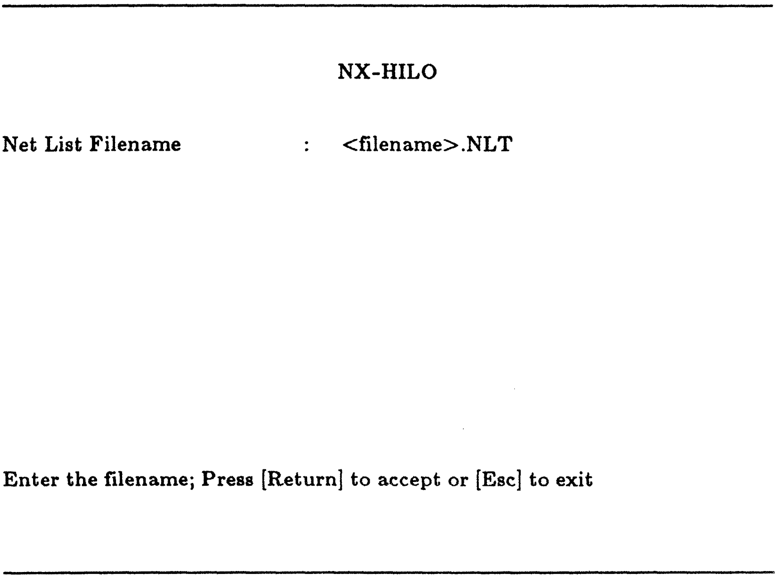
   If you answered no to the "One Output File" prompt, the system displays the output .LOG filename and, as it creates them, the .HDL filenames.

   When NX-HILO completes processing, it returns to the opening menu. Select "Exit" and press [Return] to exit to DOS, or select one of the other options to process another file.

RUNNING NX-HILO

Select "Run" to process one single-sheet netlist.

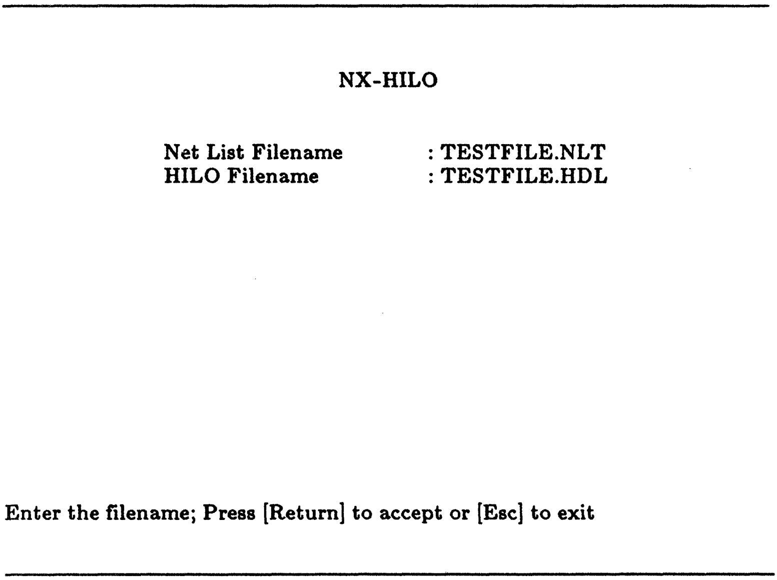
1. Select "Run" at the opening menu. The system prompts you for the name of the netlist file, as shown in Figure 4-4.



```
                         NX-HILO

Net List Filename          :   <filename>.NLT





Enter the filename; Press [Return] to accept or [Esc] to exit
```

Figure 4-4. NX-HILO Program Screen

2. Enter the name of the P-CAD netlist file to be read by NX-HILO and press [Return]. You do not need to enter the .NLT extension; the system automatically inserts it (see Figure 4-5).



```
                              NX-HILO

             Net List Filename        : TESTFILE.NLT
             HILO Filename            : TESTFILE.HDL








Enter the filename; Press [Return] to accept or [Esc] to exit
```

Figure 4-5. Sample Program Screen

3. The system prompts for the name of the HILO-compatible output file to be created. The default is the netlist filename with a .HDL extension. Press [Return] to accept the default, or enter another name and press [Return].

NOTE: At either of these two fields, if you decide not to proceed with the program, you can press [Esc] to quit and return to the opening menu.

When you enter the output filename, NX-HILO begins file processing. It displays progress reports and error messages, if any, at the bottom of the screen.

When processing is complete, the system displays the opening menu. You can select "Exit" to exit to DOS or one of the other options to process another file.

CHAPTER 5 - VIEWING THE OUTPUT FILES

NX-HILO produces two output files, the HILO-compatible netlist file (.HDL) and the log file (.LOG).

This section describes how to view and print the output files and how to interpret the .HDL file. Error messages that might be in the .LOG file are listed and explained in Appendix D, "Error Messages."

VIEWING AND PRINTING FILES

To display or print an output file, you can use the DOS TYPE and [Ctrl]-[P] commands. For example, to display a file, type:

```
TYPE TESTFILE.LOG  [Return]
```

Or, to print a file, type:

```
TYPE TESTFILE.LOG [Ctrl]-[P]  [Return]
```

When the file is printed, press [Ctrl]-[P] again to turn off the print function.

You can also use the DOS PRINT command to print a file.

THE .HDL FILE

The .HDL netlist is made up of three sections, the circuit header, element declaration, and wire declaration. They are preceded by the P-CAD header, If you ran batch mode and specified a single output file, the file contains all three sections of the first file, followed by all three sections of the second file, and so on.

This section describes the information in each part of the .HDL file and tells where the information originates in the PC-CAPS schematic. Appendix E, "Sample Circuit: MOSFF," and Appendix F, "Sample Circuit: BCDADDER," contain sample .HDL files for two designs.

Circuit Header

The circuit header is the first section of each HILO circuit description block. It consists of one record that identifies the circuit. Each field of the record is described below.

<type> - Specified with the TYPE attribute in SYMB mode. If no type is specified, the default is CCT.

<technology> - (Used by HILO-3 only.) Specified with the TECH attribute in SYMB mode. If no technology is specified, this field is left empty and HILO uses TTL as the default value.

<delay scale> - Specified with the DLY attribute in SYMB mode. If no delay scale is specified, this field is left empty and HILO uses ns 1 as the default value.

<parameters> - Specified with the PARM1...PARMn parameters in SYMB mode. No parameters are listed if the attributes are unassigned.

<circuit name> - The name of the file in which the schematic for the circuit is found. For multi-sheet schematics, the name of the first sheet listed in the batch control file. If the NAME attribute is assigned in SYMB mode, it is used instead.

<wire list> - The names (assigned using the NAME/NET command) of wires attached to the symbol for the schematic, if any, or from wires attached to HILO pin components. The order of wires is the same as the order in which pins are assigned to the symbol, or the order specified by the ORD attribute of the pin components. Unnamed wires are assigned names in the format <componentname>$<pinname> or <componentname>$<pinnumber> or in the PC-NODES format, UNsssnnn, depending on the configuration option chosen for "Unnamed Wire Alias." <componentname> is the name of the component that is the source for the wire and <pinname/number> is the pin that sources the wire.

> NOTE: Any unnamed wire that is not connected to an output pin cannot be given an alias, so its name appears in the PC-NODES format.

Element Declaration

The element declaration section is the next part of the HILO netlist. It contains one record for each component in the circuit. Each record contains several fields, which are described below.

<element definition> - For HILO primitives, this is determined by the Component Type ID number assigned to the primitive symbol. See Appendix A, "NX-HILO Schematic Symbols." A component with a Component Type ID number below zero or above 254 is non-primitive. Its element definition name is taken either from the name of the file where the component symbol is defined, or from the value of the NAME attribute if it is assigned to the symbol.

<delays> - Specified with the PARM1...PARMn attributes in DETL mode. No parameters are listed if the attributes are unassigned.

<strength> - (Used by HILO-3 only.) Specified by assigning the HI and LO attributes to a component in DETL mode of PC-CAPS. If the attributes are not assigned, the field is left blank.

<instance name> - Assigned using NAME/COMP in DETL mode. If a name is not assigned, PC-NODES assigns a name in the format UCsssnnn, where sss is the sheet identifier, and nnn is a sequence of digits starting with 000.

<wire list> - The names (assigned using the NAME/NET command) of wires attached to the component in the schematic. The order of wires is the same as the order in which pins are assigned to the symbol. Unnamed wires are assigned names in the format <componentname>$<pinname> or <componentname>$<pinnumber> or in PC-NODES format, depending on the configuration option chosen for "Unnamed Wire Alias." <componentname> is the name of the component which is the source for the wire, and <pinname/number> is the pin which sources the wire.

Wire Declaration

The wire declaration section is the final part of the .HDL file.

A wire is listed in this section only if the W$ attribute identifying the wire has been assigned to a schematic component. Each record contains several fields, which are described below.

<wire type> - Specified by assigning the W$ attribute to a component attached to the wire.

<wire delay> - Specified by assigning the D$ attribute to a component attached to the wire. If the attribute is not assigned, the field is left blank.

<wire name> - Specified by using the NAME/NET command in PC-CAPS DETL mode or assigned by PC-NODES in the format UNsssnnn or assigned by NX-HILO in the format selected at the "Unnamed Wire Alias" configuration option.

APPENDIX A. NX-HILO SCHEMATIC SYMBOLS

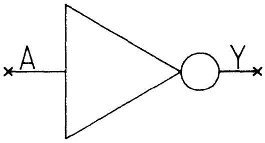
This section describes the symbols that are supplied with NX-HILO. For each symbol, this list shows graphics, the name of the symbol file, pin names, pin order, and pin type.

HILO PRIMITIVE SYMBOLS

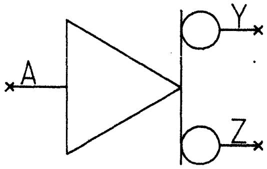
The following primitive symbols are supplied with NX-HILO. You can create more symbols as necessary, using the provided symbols as models.

NOT Symbols (TYPE ID = 1)

1-Output NOT
FILENAME: NOT.SYM
PINS:
Output: Y
Input: A

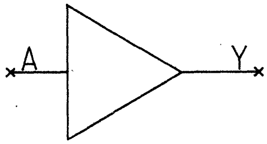


2-Output NOT
FILENAME: NOT2.SYM
PINS:
Output: Y,Z
Input: A

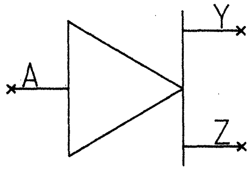


BUF Symbols (TYPE ID = 2)

1-Output BUF
FILENAME: BUF.SYM
PINS:
Output: Y
Input: A

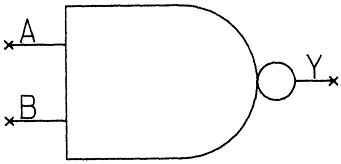


2-Output BUF
FILENAME: BUF2.SYM
PINS:
Output: Y,Z
Input: A

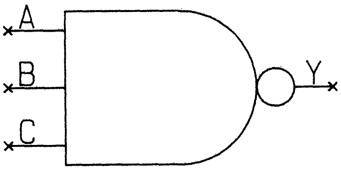


NAND Symbols (TYPE ID = 4)

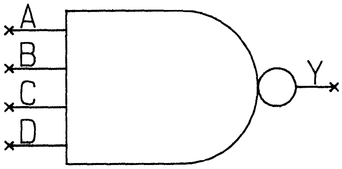
2-Input NAND
FILENAME: NAND2.SYM
PINS:
Output: Y
Input: A,B



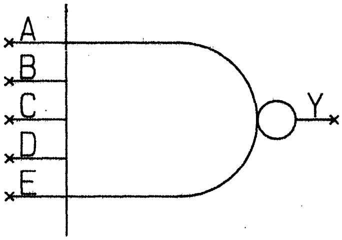
3-Input NAND
FILENAME: NAND3.SYM
PINS:
Output: Y
Input: A,B,C



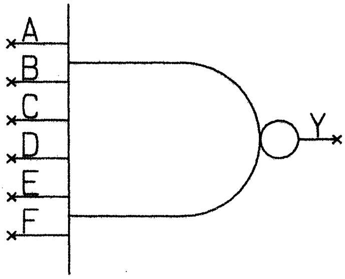
4-Input NAND
FILENAME: NAND4.SYM
PINS:
Output: Y
Input: A,B,C,D



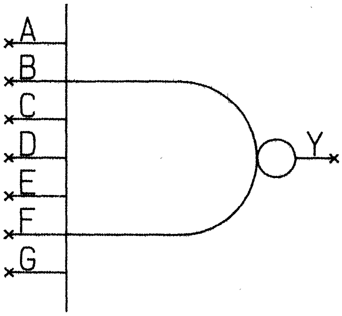
5-Input NAND
FILENAME: NAND5.SYM
PINS:
Output: Y
Input: A,B,C,D,E



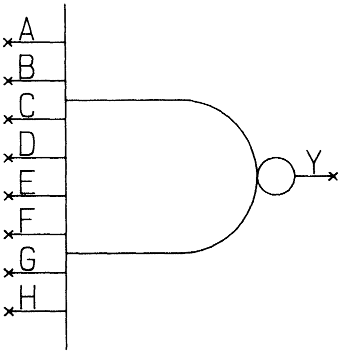
6-Input NAND
FILENAME: NAND6.SYM
PINS:
Output: Y
Input: A,B,C,D,E,F



7-Input NAND
FILENAME: NAND7.SYM
PINS:
Output: Y
Input: A,B,C,D,E,F,G

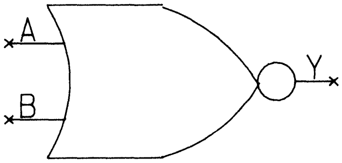


8-Input NAND
FILENAME: NAND8.SYM
PINS:
Output: Y
Input: A,B,C,D,
E,F,G,H

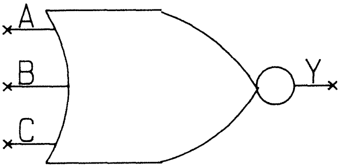


NOR Symbols (TYPE ID = 5)

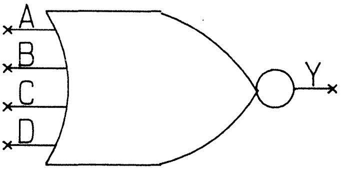
2-Input NOR
FILENAME: NOR2.SYM
PINS:
Output: Y
Input: A,B



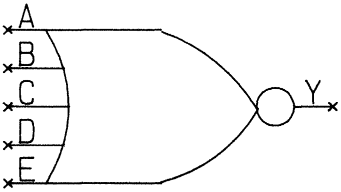
3-Input NOR
FILENAME: NOR3.SYM
PINS:
Output: Y
Input: A,B,C



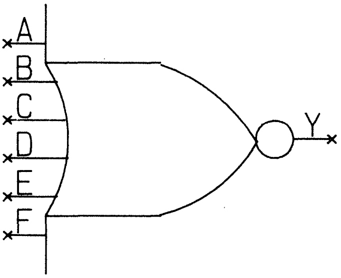
4-Input NOR
FILENAME: NOR4.SYM
PINS:
Output: Y
Input: A,B,C,D



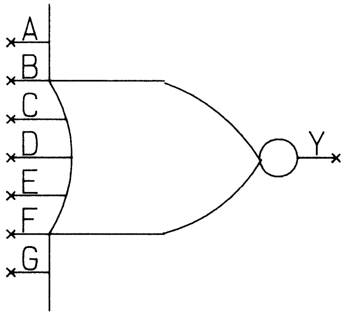
5-Input NOR
FILENAME: NOR5.SYM
PINS:
Output: Y
Input: A,B,C,D,E



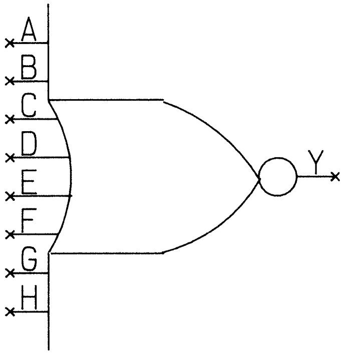
6-Input NOR
FILENAME: NOR6.SYM
PINS:
Output: Y
Input: A,B,C,D,E,F



7-Input NOR
FILENAME: NOR7.SYM
PINS:
Output: Y
Input: A,B,C,D,
E,F,G

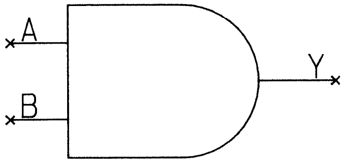


8-Input NOR
FILENAME: NOR8.SYM
PINS:
Output: Y
Input: A,B,C,D,
E,F,G,H

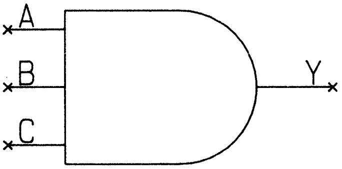


AND Symbols (TYPE ID = 6)

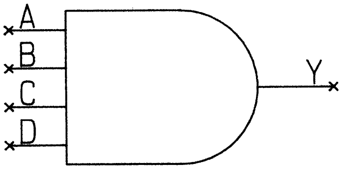
2-Input AND
FILENAME: AND2.SYM
PINS:
Output: Y
Input: A,B



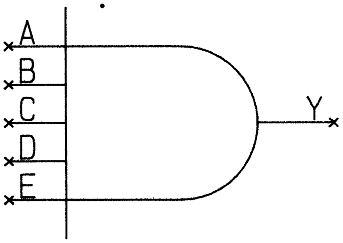
3-Input AND
FILENAME: AND3.SYM
PINS:
Output: Y
Input: A,B,C



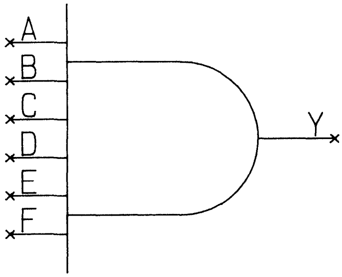
4-Input AND
FILENAME: AND4.SYM
PINS:
Output: Y
Input: A,B,C,D



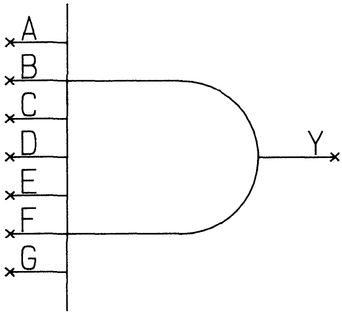
5-Input AND
FILENAME: AND5.SYM
PINS:
Output: Y
Input: A,B,C,D,E



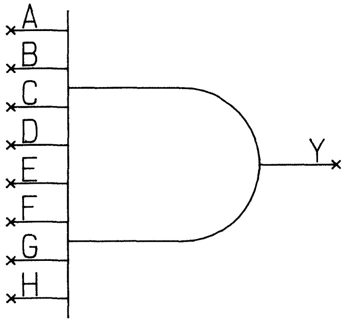
6-Input AND
FILENAME: AND6.SYM
PINS:
Output: Y
Input: A,B,C,
D,E,F



7-Input AND
FILENAME: AND7.SYM
PINS:
Output: Y
Input: A,B,C,D,
E,F,G

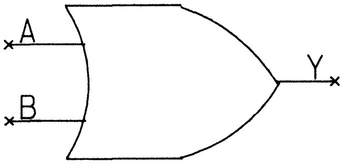


8-Input AND
FILENAME: AND8.SYM
PINS:
Output: Y
Input: A,B,C,D,
E,F,G,H

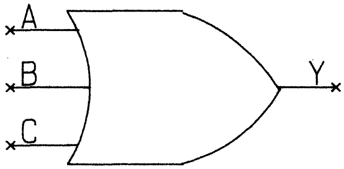


OR Symbols (TYPE ID = 7)

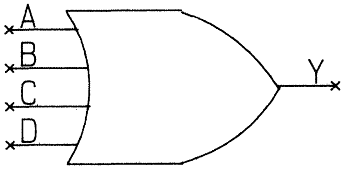
2-Input OR
FILENAME: OR2.SYM
PINS:
Output: Y
Input: A,B



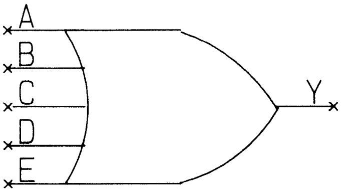
3-Input OR
FILENAME: OR3.SYM
PINS:
Output: Y
Input: A,B,C



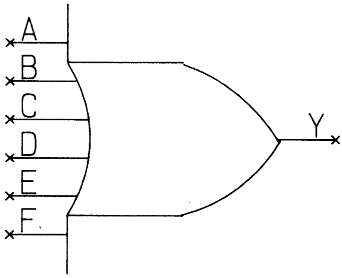
4-Input OR
FILENAME: OR4.SYM
PINS:
Output: Y
Input: A,B,C,D



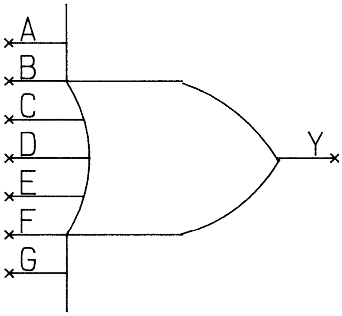
5-Input OR
FILENAME: OR5.SYM
PINS:
Output: Y
Input: A,B,C,D,E



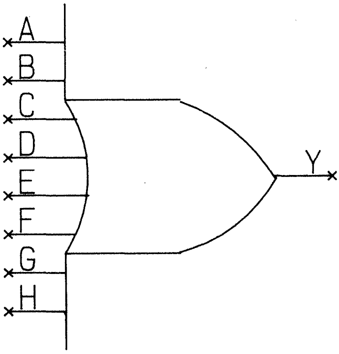
6-Input OR
FILENAME: OR6.SYM
PINS:
Output: Y
Input: A,B,C,
D,E,F



7-Input OR
FILENAME: OR7.SYM
PINS:
Output: Y
Input: A,B,C,D,
E,F,G

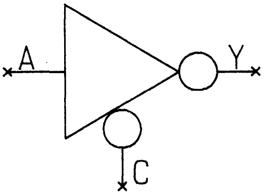


8-Input OR
FILENAME: OR8.SYM
PINS:
Output: Y
Input: A,B,C,D,
E,F,G,H



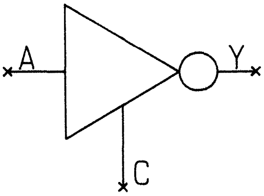
NOTIF0 Symbol (TYPE ID = 8)

NOTIF0
FILENAME: NOTIF0.SYM
PINS:
Output: Y
Input: A,C



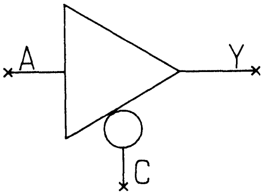
NOTIF1 Symbol (TYPE ID = 9)

NOTIF1
FILENAME: NOTIF1.SYM
PINS:
Output: Y
Input: A,C



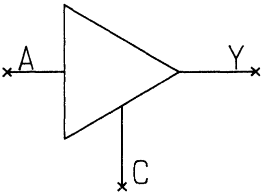
BUFIF0 Symbol (TYPE ID = 10)

BUFIF0
FILENAME: BUFIF0.SYM
PINS:
  Output: Y
  Input: A,C



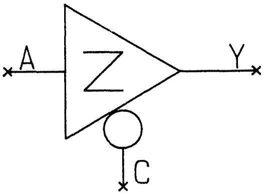
BUFIF1 Symbol (TYPE ID = 11)

BUFIF1
FILENAME: BUFIF1.SYM
PINS:
  Output: Y
  Input: D,C



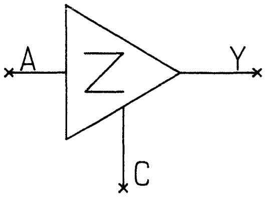
MOVEIF0 Symbol (TYPE ID = 12)

MOVEIF0
FILENAME: MOVEIF0.SYM
PINS:
  Output: Y
  Input: A,C



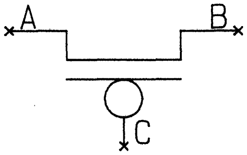
MOVEIF1 Symbol (TYPE ID = 13)

MOVEIF1
FILENAME: MOVEIF1.SYM
PINS:
- Output: Y
- Input: A,C



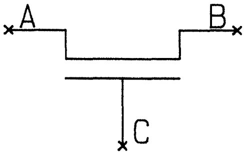
TRANIF0 Symbol (TYPE ID = 14)

TRANIF0
FILENAME: TRANIF0.SYM
PINS:
- IO: A,B
- Input: C



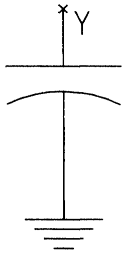
TRANIF1 Symbol (TYPE ID = 15)

TRANIF1
FILENAME: TRANIF1.SYM
PINS:
- IO: A,B
- Input: C



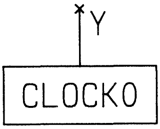
Capacitor Symbol (TYPE ID = 16)

CAPACITOR
FILENAME: CAP.SYM
PINS:
Output: Y



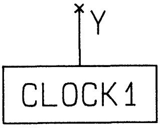
CLOCK0 Symbol (TYPE ID = 17)

CLOCK0
FILENAME: CLOCK0.SYM
PINS:
Output: Y



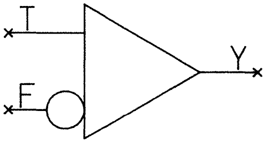
CLOCK1 Symbol (TYPE ID = 18)

CLOCK1
FILENAME: CLOCK1.SYM
PINS:
Output: Y



BALR Symbol (TYPE ID = 19)

BALR
FILENAME: BALR.SYM
PINS:
 Output: Y
 Input: T,F

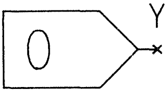


NX-HILO PIN SYMBOLS

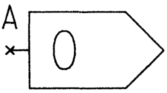
You can enter the following symbols in the top-level schematic to specify terminal pins. If you want to specify pin order, use the ATTR/SCHG command when you enter each pin to change the ORD attribute to the desired number (as described in "Preparing the Schematic Database").

Pin Symbols (TYPE ID = 255)

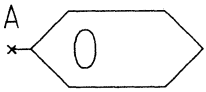
INPUT PIN
FILENAME: PIN$IN.SYM
PIN:
 Output: Y



OUTPUT PIN
FILENAME: PIN$OUT.SYM
PIN:
 Input: A



I/O PIN
FILENAME: PIN$IO.SYM
PIN:
IO: A



APPENDIX B. SYMBOL SUMMARY

Table B-1 summarizes all symbols recognized by NX-HILO. Any symbol you create must be one of these types of components and must be assigned the appropriate Component Type ID.

Table B-1. Symbol Summary

TYPE ID	SYMBOL TYPE
1	NOT
2	BUF
3	(not used)
4	NAND
5	NOR
6	AND
7	OR
8	NOTIF0
9	NOTIF1
10	BUFIF0
11	BUFIF1
12	MOVEIF0
13	MOVEIF1
14	TRANIF0
15	TRANIF1
16	CAPACITOR
17	CLOCK0
18	CLOCK1
19	BALR
255	PIN$IN
255	PIN$OUT
255	PIN$IO
255	<functional model>
<0 or >254	<subcircuit>

APPENDIX C. ATTRIBUTE SUMMARY

This appendix lists the PC-CAPS attributes that are used with NX-HILO. Assign all attributes by using the ATTR/ACOM command and entering the attribute in the format:

```
<keyword>=<value>
```

where <keyword> is the name of the attribute and <value> is the value you specify for it (for example, SHEET=01). If the value contains embedded blanks, you must enclose it in quotation marks (for example, DLY="ns 10").

Table C-1 lists, for each attribute, its name, a brief description, its use in NX-HILO, where in the PC-CAPS design you use it, and the PC-CAPS mode in which you assign it.

Table C-1. Attribute Summary

		NX-HILO	PC-CAPS	
Attribute	Description	Result	Assign to:	Mode
SHEET	sheet identifier	identifies multiple sheets	sheet of schematic	SYMB
ORD[1]	pin order	determines pin order	HILO terminal pins	DETL
NAME	renames component (replaces filename)	circuit[2] or element[3] name	primitive, subcircuit, or main schematic	SYMB
TYPE	circuit type	circuit type[2]	schematic	SYMB
TECH	default strength	technology[2]	schematic	SYMB
DLY	delay scale	delay scale[2]	schematic	SYMB
PARM1 PARMn	delay parameters (symbolic names)	timing[2]	schematic	SYMB
	delay parameters (actual values)	timing[3]	component	DETL
HI LO	logic level	strength[3]	component	DETL
W$	wire type	wire type[4]	component connected to wire	DETL
D$	delay parameters	wire delay[4]	component connected to wire	DETL

1. The ORD attribute is preassigned as 0 to each HILO pin symbol. When you place a pin symbol on a schematic, you use the ATTR/SCHG command to enter the appropriate pin order.

2. Field of the circuit header record in the .HDL file.

3. Field of an element declaration record in the .HDL file.

4. Field of a wire declaration record in the .HDL file.

APPENDIX D. ERROR MESSAGES

NX-HILO displays error messages on the screen and stores them in the .LOG log file, which you can view or print.

This appendix lists and explains NX-HILO error messages.

SHEET IDENTIFIER ERRORS

Message: ****** Sheet Identifier Error**
No Sheet Name

Cause: The batch control file references a netlist as one sheet of a multi-sheet schematic, but the SHEET attribute is not assigned to the schematic.

Action: Assign a sheet ID to the schematic or correct the batch control file.

Message: ****** Sheet Identifier Error**
Sheet Name Too Long : <sheetname>

Cause: The sheet identifier name is longer than three characters.

Action: Change the sheet identifier name.

PIN SYMBOL ERRORS

Message: ****** Invalid Pin Symbol <element name> Circuit Symbol is defined**

Cause: A schematic for which a symbol has been defined also has NX-HILO pin symbols.

Action: Remove either the SYMB view or the pin symbol from the schematic.

Message: ****** Unordered Terminal Wires**

Cause: NX-HILO cannot determine wire order in the main circuit because of an error in assigning the ORD attribute. Possible errors are: an ORD value that is 0 or non-numeric, more than one pin with the same ORD value, no ORD attribute.

Action: If you require the terminal wires to have a specific order, check and correct the ORD attribute values for the pin symbols; otherwise, no action is necessary.

DATABASE AND SYSTEM ERRORS

Message: ****** Database Error (<identifier>)**

Cause: The netlist named by <identifier> is incomplete or contaminated.

Action: Rerun PC-NODES to generate a new .NLT file. If the error persists, check the original schematic.

Message: **** **Database Error <filename>**
Database File Not Found

Cause: The named file is not present in the current directory.

Action: Make sure the file to process is in the current directory and that you enter the correct filename.

Message: **** **Memory Allocation Error**

**** **System Error**
Free() Error

**** **Not Enough Memory**

Cause: System error.

Action: Be sure you have 640K available. If so, call P-CAD customer support.

Message: **** **Error Opening <file type> File**

Cause: NX-HILO cannot open the log file or a work file.

Action: Try again. If the error persists, call P-CAD customer support.

Message: **Incompatible Database**

Cause: The netlist was extracted using the wrong version of PC-NODES.

Action: Use PC-NODES 1.24 or higher.

OTHER ERRORS

Message: ****** Invalid SCAT # (<scatvalue>)**
Component/Module : <compname>

Cause: The specified component has a type ID (set with SCMD/SCAT) of 1 through 254 that does not correspond to a HILO primitive.

Action: Check the symbol definition of the component and assign a correct value.

Message: ****** Invalid Name Detected <name>**

Cause: A name assigned in PC-CAPS does not conform to HILO conventions.

Action: Correct the name in the schematic.

Message: ****** File Not Found : <filename>**

Cause: NX-HILO cannot find the named file.

Action: Make sure that the file is in the current directory and that you enter the correct filename.

Message: ****** File extension must be .BCF**

Cause: The file you specified as the batch control file does not have a .BCF extension.

Action: Rename the file using the .BCF extension.

Message: **** Invalid Character Entered

Cause: You entered an invalid value for the Bar Character option during configuration.

Action: Enter either a letter, number, or $.

Message: **** Reserved Word Used

Cause: A wire, component, or circuit name is the same as a name listed in the reserved words file.

Action: Rename the wire, component, or circuit.

Message: **** Element Has Same Name As Circuit Name

Cause: The circuit being processed has the same name as one of the components used within the circuit.

Action: Rename the circuit file or change its NAME attribute value.

APPENDIX E. SAMPLE .HDL FILE: MOSFF

This appendix shows the HILO netlist for a subcircuit. This example illustrates the use of wire declaration attributes.

The MOSFF schematic is contained in the sample data files supplied with NX-HILO.

```
** ****************************************************************************
** *                                                                          *
** *                               HILO NETLIST                               *
** *                                                                          *
** *  NX-HILO Version 1.24                                                    *
** *  Copyright (C) 1985 - Personal CAD Systems, Inc.                         *
** *                                                                          *
** *  Date     : SEP 09 1985                                                  *
** *  Time     : 04:51 PM                                                     *
** *  File In  : MOSFF.NLT                                                    *
** *  File Out : MOSFF.HDL                                                    *
** *  Log File : MOSFF.LOG                                                    *
** *                                                                          *
** ****************************************************************************


IC MOS NS 100 ( WD1,WD2,WD3 )
MOSFF ( Q, D, C, S )

NOR
  U2 ( U2$1, S, U1$2 );
NOT
  U4 ( U4$1, U2$1 );
NOR
  U7 ( U7$1, S, U6$2 );
NOT
  U5 ( C$, C );
NOT
  U9 ( Q, U7$1 );
TRANIF1
  U1 ( D, U1$2, C$ );
TRANIF1
  U3 ( U1$2, U4$1, C );
TRANIF1
  U6 ( U4$1, U6$2, C );
TRANIF1
  U8 ( U6$2, Q, C$ );

UNID U2$1 ;
INPUT S ;
UNID (WD2,WD3) U7$1 ;
INPUT C ;
UNID C$ ;
INPUT D ;
TRI U1$2 ;
TRI U4$1 ;
TRI U6$2 ;
TRI (WD1,WD3) Q ;
```

APPENDIX F. SAMPLE OUTPUT FILES: BCDADDER

This appendix shows the log file and the HILO netlists for a sample circuit. The BCDADDER circuit is hierarchical and its top-level circuit has two sheets. It was processed in batch mode.

The schematic and batch control files used in the BCDADDER example are contained in the sample data files supplied with NX-HILO.

```
*******************************************************************************
*                                                                             *
*                                NX-HILO LOGFILE                              *
*                                                                             *
*  NX-HILO Version 1.24                                                       *
*  Copyright (C) 1985 - Personal CAD Systems, Inc.                            *
*                                                                             *
*  Date     : SEP 10 1985                                                     *
*  Time     : 12:01 PM                                                        *
*  File In  : BCDADDER.BCF                                                    *
*  Log File : BCDADDER.LOG                                                    *
*                                                                             *
*******************************************************************************

Multi-sheet Module : BCDADDER.NLT


Loading BCDADDER.NLT Database . . .
Processing BCDADDER.NLT . . .


Loading BCDADD2.NLT Database . . .
Processing BCDADD2.NLT . . .
End Multi-sheet Module


Loading 4ADDER.NLT Database . . .
Processing 4ADDER.NLT . . .


Loading ADDER.NLT Database . . .
Processing ADDER.NLT . . .

Execution completed
```

```
** ******************************************************************************
** *                                                                            *
** *                               HILO NETLIST                                 *
** *                                                                            *
** *  NX-HILO Version 1.24                                                      *
** *  Copyright (C) 1985 - Personal CAD Systems, Inc.                           *
** *                                                                            *
** *  Date     : SEP 10 1985                                                    *
** *  Time     : 12:01 PM                                                       *
** *  File In  : BCDADDER.NLT                                                   *
** *  File Out : BCDADDER.HDL                                                   *
** *  Log File : BCDADDER.LOG                                                   *
** *                                                                            *
** ******************************************************************************

IC PMOS PS 100
BCDADDER (CARRY, SUM3, SUM2, SUM1, SUM0, Y3, Y2, Y1, Y0, X3, X2, X1, X0)

** Sheet = 001

OR
  UC001001 ( UC001001$Y, S1, S2 );
AND
  UC001003 ( UC001003$Y, S3, UC001001$Y );
4ADDER
  UC001010 ( UC001010$COUT, S3, S2, S1, SUM0, Y3, Y2, Y1, Y0, X3, X2, X1, X0 );
OR
  UC001011 ( CARRY, UC001003$Y, UC001010$COUT );

** Sheet = 002

HALFADD ( 1,7,9 )
  UC002001 ( SUM3, , UC002003$COUT, S3 );
HALFADD ( 1,3,7 )
  UC002002 ( SUM1, UC002002$CO, S1, CARRY );
FULLADDER ( 4,8,9 )
  UC002003 ( SUM2, UC002003$COUT, UC002002$CO, S2, CARRY );
```

```
** ****************************************************************************
** *                                                                          *
** *                                HILO NETLIST                              *
** *                                                                          *
** *  NX-HILO Version 1.24                                                    *
** *  Copyright (C) 1985 - Personal CAD Systems, Inc.                         *
** *                                                                          *
** *  Date     : SEP 10 1985                                                  *
** *  Time     : 12:01 PM                                                     *
** *  File In  : 4ADDER.NLT                                                   *
** *  File Out : 4ADDER.HDL                                                   *
** *  Log File : BCDADDER.LOG                                                 *
** *                                                                          *
** ****************************************************************************

IC PMOS PS 100
4ADDER ( COUT, S3, S2, S1, S0, Y3, Y2, Y1, Y0, X3, X2, X1, X0 )

FULLADDER ( 4,8,9 )
  UC000000 ( S3, COUT, UC000001$COUT, X3, Y3 );
FULLADDER ( 4,8,9 )
  UC000001 ( S2, UC000001$COUT, UC000002$COUT, X2, Y2 );
FULLADDER ( 4,8,9 )
  UC000002 ( S1, UC000002$COUT, UC000003$COUT, X1, Y1 );
FULLADDER ( 4,8,9 )
  UC000003 ( S0, UC000003$COUT, NC, X0, Y0 );

INPUT0 NC ;
```

```
** *****************************************************************************
** *                                                                           *
** *                               HILO NETLIST                                *
** *                                                                           *
** *  NX-HILO Version 1.24                                                     *
** *  Copyright (C) 1985 - Personal CAD Systems, Inc.                          *
** *                                                                           *
** *  Date     : SEP 10 1985                                                   *
** *  Time     : 12:01 PM                                                      *
** *  File In  : ADDER.NLT                                                     *
** *  File Out : ADDER.HDL                                                     *
** *  Log File : BCDADDER.LOG                                                  *
** *                                                                           *
** *****************************************************************************


IC PMOS PS 100 ( D1,D2,D3 )
FULLADDER ( SUM, COUT, CIN, A, B )

HALFADD ( D1,D2,D3 )
  UC000000 ( SUM, UC000000$CO, CIN, UC000001$SUM );
HALFADD ( D1,D2,D3 )
  UC000001 ( UC000001$SUM, UC000001$CO, A, B );
OR ( HI = WEAK, LO = STRONG )( 1:3:7,4:8:10 )
  UC000002 ( COUT, UC000000$CO, UC000001$CO );
```