NX-CADAT
User's Manual

August 1986

p-cad®
PERSONAL CAD SYSTEMS INC.

000-0124-00

COPYRIGHT

Copyright (c) 1986 by Personal CAD Systems, Inc. (P-CAD).

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Personal CAD Systems, Inc.

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.

TRADEMARKS

NX-CADAT, PC-CAPS, PC-LINK, and PC-NODES are trademarks of Personal CAD Systems, Inc. (P-CAD).

CADAT and Personal CADAT are trademarks of HHB Systems, Inc.

ABOUT THIS MANUAL

This manual describes NX-CADAT, a program that converts P-CAD schematic database files to ASCII files compatible with the CADAT simulator system.

Chapter 1, INTRODUCTION, provides an overview of NX-CADAT and installation instructions.

Chapter 2, PREPARING THE SCHEMATIC DATABASE, gives instructions for using PC-CAPS to create a schematic database to be used with NX-CADAT.

Chapter 3, PREPARING THE INPUT FILES, gives instructions for creating the files to be input into NX-CADAT.

Chapter 4, USING NX-CADAT, gives instructions for configuring and running NX-CADAT.

Chapter 5, VIEWING THE OUTPUT FILES, tells how to view, print, and interpret the NX-CADAT output files.

Appendix A, ERROR MESSAGES, explains error messages that may be produced during NX-CADAT program operation.

Appendix B, NX-CADAT SCHEMATIC SYMBOLS, shows the library of primitive symbols provided with NX-CADAT.

Appendix C, ATTRIBUTE SUMMARY, summarizes attributes assigned using PC-CAPS and used with NX-CADAT.

Appendix D, SAMPLE FILES, explains the sample designs supplied with NX-CADAT and shows several sample files.

NOTATION

This manual gives step-by-step procedures and examples. To make it easy for you to follow these procedures, we use the following notation.

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

Type PCPLOTS and press [Return].

lower Lowercase letters in a syntax definition of a command, attribute, or other keyboard entry indicate a variable name for which you should enter the appropriate information. For example:

NXCADAT infile

[ ] Square brackets indicate an optional element in a command, attribute, or other keyboard entry. For example:

NX-CADAT infile [outfile]

Square brackets also indicate the name of a key. For example:

[Return]

[Return] [Return] indicates the key that is used to execute a command or accept an option. This key may be labeled differently depending on your system. For example:

[RETURN], [↵], [Enter], [Enter↵], [ENTER]

[ ]-[ ] Square brackets connected with a hyphen indicate keys that must 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, the DOS command

DIR *.SYM

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

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

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

CONTENTS

CHAPTER 1. INTRODUCTION. 1-1
Overview. 1-1
System Requirements. 1-5
Installation 1-5

CHAPTER 2. PREPARING THE SCHEMATIC DATABASE. 2-1
Schematic Database Structure. 2-1
Hierarchical Structure. 2-2
Multisheet Structure 2-2
Creating CADAT-Compatible Symbols 2-3
Component Type 2-4
Pin Order and Pin Names. 2-5
Unpackaged and Packaged Designs 2-5
Packaging Preassignment 2-6
Using Unpackaged and Packaged Components 2-7
Creating CADAT-Compatible Schematics 2-8
Assigning Net Names. 2-9
Entering Terminal Pins. 2-9
Using a Symbol. 2-9
Using Pin Components 2-10
CDPIN Attribute. 2-11
Creating CADAT-Compatible Models. 2-13
Subcircuit Models. 2-13
Behavioral Models 2-14
ROM Models. 2-15
Using Aliases With Models. 2-15
Assigning Attributes. 2-17
Attributes Assigned to a Schematic. 2-19
Attributes Assigned to Components. 2-24

CONTENTS (Continued)

CHAPTER 3. PREPARING THE INPUT FILES 3-1
Extracting and Expanding the Schematic Netlist 3-1
Creating the Batch Control File 3-2

CHAPTER 4. USING NX-CADAT. 4-1
Interactive Mode. 4-2
Configuring NX-CADAT 4-3
Running NX-CADAT 4-6
Command Line Mode 4-10
Batch Mode 4-12
Updating a Schematic 4-13

CHAPTER 5. VIEWING THE OUTPUT FILES . . 5-1
Viewing and Printing Files. 5-1
Network Description File 5-2
HEADER Section. 5-2
MODEL Section. 5-3
PARTS Section. 5-4
VECTORS Section. 5-5
CONNECTIONS Section. 5-5
EXTERNALS Section 5-6
SPECIAL INFO Section. 5-7
Model Description File. 5-7
PINS Subsection. 5-8
DEFAULTS Subsection. 5-9
TYPES Subsection 5-10
Digital Stimulus File 5-11

APPENDIX A. ERROR MESSAGES A-1
File Access and System Errors. A-1
Command Line/Batch Control File Errors. A-4
Netlist Processing Errors A-6

APPENDIX B. NX-CADAT SCHEMATIC SYMBOLS B-1

CONTENTS (Continued)

APPENDIX C. ATTRIBUTE SUMMARY..... C-1

APPENDIX D. SAMPLE FILES D-1
DEMO Sample..... D-1
Schematic Database: DEMOI.SCH..... D-3
Network Description File: DEMOI.CKT.... D-4
DSL File: NXCADAT.DSL..... D-5
Back-Annotation Command File:
DEMOI.CMD D-6
Updated Schematic Database:
DEMOF.SCH D-7
DF115 Sample..... D-8
Schematic Symbol: DF115.SYM..... D-9
Model Description File: DF115.MDL D-10

FIGURES

1-1. NX-CADAT Input and Output 1-4
3-1. Sample Batch Control File..... 3-4
4-1. NX-CADAT Opening Menu..... 4-2
4-2. NX-CADAT Configuration Screen 4-4
4-3. NX-CADAT Program Screen..... 4-7
4-4. Sample Program Screen..... 4-9
4-5. Command Line Mode Screen 4-12

TABLES

C-1. Attribute Summary C-2

CHAPTER 1. INTRODUCTION

The NX-CADAT interface program translates P-CAD circuit schematics to an ASCII format that is compatible both with the CADAT simulation system, which runs on several computer systems and workstations, and with Personal CADAT, which runs on the IBM PC/AT with MS-DOS version 3.1 or higher.

NX-CADAT generates files in CADAT's Network Description Language (NDL) and Model Description Language (MDL). It does not generate files in CADAT's Behavioral Modeling Language (BMD) or ROM Description Language (RDL).

NX-CADAT can package a design into parts corresponding to those in CADAT's standard simulation libraries. When this capability is used, the design is called a packaged design.

NX-CADAT allows behavioral models and fault simulation. If you use the Personal CADAT system, be aware that Personal CADAT does not support these features.

This chapter provides an overview of NX-CADAT and installation instructions.

OVERVIEW

NX-CADAT, together with PC-NODES and, if needed, PC-LINK, enables you to produce a CADAT-compatible file from a schematic created with PC-CAPS. NX-CADAT also allows you, as an option, to package

the schematic components into CADAT library parts and to back-annotate the schematic with the packaging information.

There are five stages in this process:

1. Using PC-CAPS, you prepare the schematic design. This design consists of one or more schematic files (<filename>.SCH), which are created by interconnecting symbols (<filename>.SYM and <filename>.SCH). 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 netlist file (<filename>.NLT) from each schematic file. The netlist file contains component and interconnection information for the schematic. For any design that consists of only one schematic file, you can input the netlist file directly to NX-CADAT. In this case, you do not need to perform step 3 or 4.

3. If your design is packaged and contains more than one schematic file, after you extract the netlist for each separate sheet, you must use PC-LINK to link all the netlists into a single expanded netlist file (<filename>.XNL).

4. If your design is unpackaged and 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 (<filename>.BCN). This file contains the names of all the netlist files to be processed. NX-CADAT will repeatedly read the batch control file and process the next netlist file until all the files have been translated.

5. You then input the netlist file, expanded netlist file, or batch control file to NX-CADAT and select the language for the output file, either Network Description Language or Model Description Language.

   NX-CADAT generates the following files:

   - The CADAT file describes the circuit in the language you selected. It is named <filename>.CKT for a network description file or <filename>.MDL for a model description file. If you selected the network description file, the digital stimulus language file (NXCADAT.DSL) is also created.

   - The log file (<filename>.LOG) contains all messages displayed during the execution of NX-CADAT. This file is useful for examining any error messages that are reported by NX-CADAT.

   - The back-annotation command file (<filename>.CMD) contains the packaging information that was assigned by NX-CADAT. This file is produced only for packaged designs. You can input this file to PC-CAPS to update the original schematic.

Figure 1-1 illustrates the process described above.

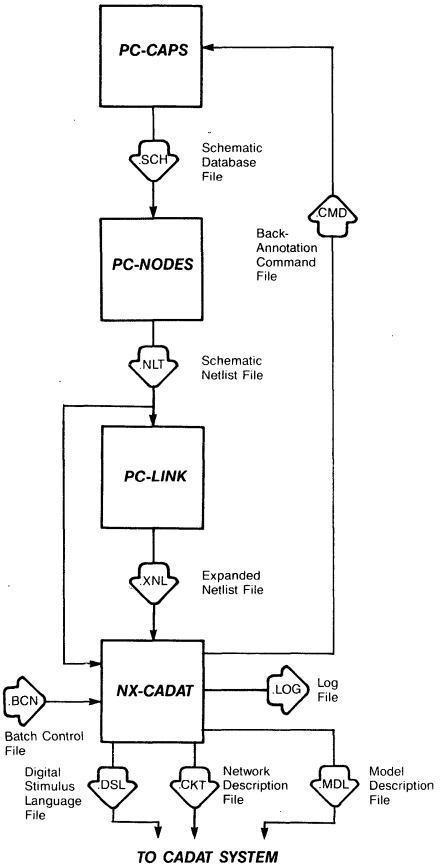


Figure 1-1. NX-CADAT Input and Output

SYSTEM REQUIREMENTS

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

- IBM PC/XT, PC/AT, TI Professional, Tandy TRS 2000, or equivalent
- 640K of RAM
- P-CAD supported graphics board and monitor
- DOS 2.0 or higher operating system
- The CONFIG.SYS file in the root directory, containing a BUFFERS value of at least 12 and a FILES value of at least 15
- The PCADDRV.SYS file (created automatically when you use the P-CAD INSTALL program) in the root directory and the appropriate loadable device driver files in the appropriate directory

INSTALLATION

Your NX-CADAT package contains a program diskette and three diskettes that contain a symbol library of CADAT primitives.

The program diskette contains the following files:

Program file:

NXCADAT.EXE

Sample data files:

DEMO Sample:	DF115 Sample:
DEMOI.SCH	DF115.SYM
NXCADAT.CFG	DF115.NLT
DEMOI.NLT	DF115.MDL
DEMOI.CKT	
NXCADAT.DSL	
DEMOI.CMD	
DEMOF.SCH	

You can use the sample data to see how the program works. Appendix D, "Sample Files," provides more information about the sample data.

Appendix B, "NX-CADAT Schematic Symbols," lists and explains the primitives on the library diskette.

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

Install the program file 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:*.EXE [Return]

Next, copy 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:DEMO*.* [Return]
COPY A:DF115.* [Return]
COPY A:*.CFG [Return]
```

Remove the program diskette.

Create a new directory for the CADAT symbol library by typing:

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

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

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

Then copy the files by inserting each diskette in turn and typing, for each diskette:

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

CHAPTER 2. PREPARING THE SCHEMATIC DATABASE

Before using NX-CADAT, you must use PC-CAPS to create a schematic database. This chapter describes PC-CAPS schematic structure and gives guidelines for entering the special information required to construct a CADAT-compatible schematic. To ensure that your database is compatible with CADAT, you should read this chapter before creating your schematic.

SCHEMATIC DATABASE STRUCTURE

A PC-CAPS schematic consists of an interconnected group of predefined symbols. Each symbol is a logical and pictorial representation of a schematic component. For example, a 7400 symbol is a single 2-input NAND gate.

With NX-CADAT, P-CAD supplies a library of symbols that correspond to the primitives in CADAT's library. You can use these symbols as they are, modify them, use symbols from P-CAD's TTL schematic symbol library, or create your own symbols. Before modifying or creating symbols, read the CADAT requirements in the following sections and Appendix B, "NX-CADAT Schematic Symbols."

Symbols are created in PC-CAPS SYMB mode as files with the .SYM filename extension. You assemble symbols into a schematic in DETL mode, producing a schematic database file with a .SCH filename extension.

NX-CADAT supports hierarchical and multisheet schematic structures. These structures are described below.

Hierarchical Structure

Hierarchical structure is the usage of symbols to represent subcircuits of a schematic. With PC-CAPS, you can create a symbol to represent a subcircuit, then use the symbol to represent the subcircuit in a higher level schematic. PC-CAPS allows up to 15 levels of hierarchy. To use hierarchical structure with NX-CADAT, for each hierarchical element, you must create a CADAT subcircuit model and enter it in the CADAT library. For information about creating CADAT models, refer to "Creating CADAT-Compatible Models."

NX-CADAT does not support hierarchical structure for packaged designs.

Multisheet Structure

A multisheet schematic is created by producing a number of single-sheet schematics. Connections between sheets are indicated by giving connected wires the same net name on each sheet. The P-CAD system allows up to 100 sheets in a multisheet schematic.

NX-CADAT supports multisheet schematics at all levels of the design hierarchy.

To indicate that a given schematic is a single sheet of a multisheet design, assign the SHEET attribute to the sheet. In PC-CAPS SYMB mode, use the ATTR/ACOM

command and enter, anywhere in the schematic, the following attribute:

SHEET=sheet_id

where:

sheet_id is one to three characters and is typically a number, with the first sheet being sheet 1 and so on. Sheet IDs in a multisheet schematic must be unique.

This attribute is used by PC-NODES to assign names to unnamed components and nets. It is not used by NX-CADAT.

For an unpackaged design, you must use NX-CADAT batch mode to process a multisheet schematic.

For a packaged design, you must use PC-LINK to link together the netlists from all the sheets of the design before inputting the design to NX-CADAT. PC-LINK merges together wires that have the same name and are on different sheets.

Unpackaged and packaged designs are further described below in "Unpackaged and Packaged Designs."

CREATING CADAT-COMPATIBLE SYMBOLS

The primitive symbols provided with NX-CADAT are described in Appendix B, "NX-CADAT Schematic Symbols." You can create more basic schematic symbols as you need them, provided that a corresponding CADAT primitive exists and that you assign the appropriate component type, pin order, and pin names

as described in this section. By P-CAD convention, each symbol file has the .SYM filename extension.

For a packaged design, you can use the symbols in P-CAD's schematic symbol libraries, such as the TTL library, or you can create your own symbols. (Before using a P-CAD TTL symbol, be sure that a corresponding CADAT TTL simulation model exists.) You also must assign packaging information to each symbol as explained in "Packaging Preassignment" below.

Component Type

The component type is the name of the CADAT primitive that corresponds to the symbol. The component type is used in several places in the CADAT-compatible files output by NX-CADAT.

To assign the component type, use the ATTR/ACOM command in PC-CAPS SYMB mode to assign the CDTYPE attribute to the symbol. Assign the attribute in the format:

CDTYPE="type"

For example:

CDTYPE="3SINV-H/3"

This example shows the component type for the 3-bit wide tri-state driver with inverted data.

If the CDTYPE attribute is not assigned, NX-CADAT uses the filename of the symbol (without the extension) as the component type. CADAT component types might be incompatible with DOS filename conventions, so P-CAD recommends assigning the CDTYPE attribute to all the primitive symbols you create. All the symbols in

the library supplied with NX-CADAT have this attribute assigned. P-CAD's TTL library symbols have the same filenames as the corresponding CADAT symbols, so you can use the TTL symbols without assigning the CDTYPE attribute.

Pin Order and Pin Names

Use the ENTR/PIN command in SYMB mode to enter each pin for a symbol. NX-CADAT assigns pin order in the order you enter the pins, so enter the pins in the same sequence and with the same names as the corresponding CADAT primitive.

UNPACKAGED AND PACKAGED DESIGNS

The CADAT system can simulate designs composed of either schematic symbols or packaged parts. The input to NX-CADAT is always a schematic design. When you run NX-CADAT, you can choose whether to process the design as an unpackaged design or as a packaged design.

An unpackaged design is one for which you want to simulate the design in terms of CADAT primitives and subcircuit models; for example, a semi-custom IC design. For an unpackaged design, NX-CADAT simply translates the P-CAD netlist information into the CADAT format.

A packaged design is one for which you plan to simulate the components as packaged parts; for example, a PCB design. Using logic symbols to design a packaged circuit allows greater ease of design and enables you to use CADAT's simulation libraries of packaged parts.

For a packaged design, NX-CADAT assigns each schematic component to a specific section of a specific physical part in a CADAT library. The output CADAT-format file contains information about physical parts rather than schematic components. Each part in the design is identified with a reference designator.

All symbols used in a packaged design must have packaging information assigned as described in "Packaging Preassignment."

A design must be either entirely unpackaged or entirely packaged. However, you can use a custom circuit in a packaged design or a CADAT TTL model in an unpackaged design by following the instructions in "Using Unpackaged and Packaged Components" below.

Packaging Preassignment

For a packaged design, all symbols used in the schematic must have packaging information preassigned using the PC-CAPS SCMD/PNLC command in SYMB mode. This command allows you to set the number of gates that are to be assigned to each physical package and the pin numbers that correspond to each gate location. This command also allows you to specify locations for the reference designator and pin numbers. All the symbols in P-CAD's schematic component libraries have been preassigned this information.

When you place symbols on a schematic, you might want to preassign several logic symbols to the same physical package or to preassign a symbol to a specific gate in a package. You can use the SCMD/PNUM command in PC-CAPS DETL mode to preassign a reference designator and a section or package pin numbers to a

schematic component. (A package pin number refers to the location of the pin in the physical part rather than in the schematic symbol.)

If you do not use the SCMD/PNUM command for a component, when you run NX-CADAT, it assigns the component a reference designator and package pin numbers. You can instruct NX-CADAT either to use the preassignments you make or to assign new reference designators and sections to all components (see "Running NX-CADAT" in Chapter 4).

For more information on the SCMD/PNLC and SCMD/PNUM commands, refer to the *PC-CAPS User's Manual.*

Using Unpackaged and Packaged Components

A packaged design must contain packaging information for all schematic components. You cannot directly use a primitive in the same design as a TTL component. However, you can use a custom circuit in a packaged design in either of two ways:

- Create a model for the circuit. (See "Creating CADAT-Compatible Models" for information on models.) When creating the symbol for the model in PC-CAPS SYMB mode, use the SCMD/PNLC command to specify one gate per package and to enter the pins in the appropriate order. When placing the symbol for the model on the schematic in PC-CAPS DETL mode, you can, if you want, preassign packaging information to the model. To preassign the model, use the SCMD/PNUM command to enter package pin numbers that are the same as the component pin numbers.

- Create a model description file for the circuit using a text editor and add the model to the CADAT library. If you use this method, use PC-CAPS to create only a symbol for the model, using the CDTYPE attribute and the SCMD/PNLC command as described above.

You can use a packaged model (such as a TTL model) as a component in an unpackaged design as long as you follow the following procedures:

- If a symbol does not already exist for the component (for example, a P-CAD TTL library symbol), create a symbol for the entire package. The symbol must include all the pins in the package, including nonlogic pins such as power and ground. You must enter the pins in order by package pin number.

- When you enter the symbol in the design in PC-CAPS DETL mode, do not make any connections to the nonlogic pins.

- Before you use NX-CADAT to translate the design, be sure the Part Pin Representation configuration option is set to "part-name.pin-number".

When you use a packaged model in an unpackaged design NX-CADAT produces a warning message and processes the component as if it were unpackaged, ignoring the packaging information.

CREATING CADAT-COMPATIBLE SCHEMATICS

When you create your schematic to input to NX-CADAT, you must name the nets (signals) on the schematic and you must enter terminal pins for the

schematic. This section explains usage of net names and describes how to enter terminal pins.

Assigning Net Names

Use the PC-CAPS NAME/NET command in DETL mode to name each net whose name you want to appear on the output reports. If you do not name a net, when you extract the netlist from the schematic, PC-NODES will assign the name in the format UNsssnnn, where **sss** is the sheet ID of the schematic and **nnn** is an integer assigned sequentially beginning with 0. However, NX-CADAT will not include net names in this format in the output reports. (The output report will show the connection between the part pins but will not show the signal name.)

Entering Terminal Pins

NX-CADAT requires you to identify the terminal pins and their order in all schematics. You can specify terminals in either of two ways, which are described in the following two sections. NX-CADAT also recognizes an attribute, described in "CDPIN Attribute," that specifies additional information about terminal pins.

Information about terminal pins is used in the EXTERNALS section of the network description file.

Using a Symbol

You can specify terminal pins by creating a symbol for the entire schematic in PC-CAPS SYMB mode. Use the ENTR/PIN command to enter pin information.

NX-CADAT will assign pin order in the order you enter the pins. Each pin name must be the same as the name of the net to which it is connected.

> NOTE: PWR, VCC, GND, NC, and other names are CADAT reserved words. You cannot use these words as pin names in a schematic to be translated into a Network Description file. For a complete list of CADAT reserved words, refer to your CADAT manuals.

For a multisheet schematic, create the symbol on the first page only.

Using Pin Components

The library supplied with NX-CADAT includes four special pin component symbols. You can use the ENTR/COMP command in DETL mode to attach these pin components to the terminal wires of the circuit. The filenames and pin types of these pin components are as follows:

PININ.SYM input terminals
PINOUT.SYM output terminals
PINBUS.SYM bus terminal
PINSIM.SYM simulation terminal

Do not use these symbols on a schematic that you intend to use as a CADAT model; NX-CADAT will not generate a model description file from a netlist that includes pin components.

You must use the NAME/NET command in PC-CAPS DETL mode to name each net connected to a pin component.

To assign pin order to pin components, you can use the CDPIN attribute, which is described in the next section. If you do not use this attribute, NX-CADAT assigns pin order and generates a warning message.

The pin components supplied with NX-CADAT have the component type ID of -1 (assigned using the SCMD/SCAT command). NX-CADAT does not generate a part for a schematic component with this component type ID. (Also, if you plan to use PC-PACK to package the schematic into a PC-CARDS PCB database, PC-PACK does not generate a part for a component with this component type ID.)

CDPIN Attribute

The CDPIN attribute allows you to specify driver/sensor numbers for both types of terminal pins, pin order for pin components, and pin type for symbol pins.

Enter the CDPIN attribute using the ATTR/ACOM command. This attribute is entered differently for each type of terminal pins.

If pins were entered in PC-CAPS SYMB mode as pins on the symbol of the circuit, assign the CDPIN attribute in SYMB mode in the format:

CDPIN="pinname = [pintype] [driver/sensor]"

where:

pinname is the name (assigned using the ENTR/PIN command) of the pin.

pintype is optional and is IN, OUT, BUS, or SIM. If pin type is not specified here, NX-CADAT uses the pin type specified using the ENTR/PIN command (if I/O was specified, BUS is used).

driver/sensor is the driver/sensor number and is optional.

Example:

```
CDPIN1="MATCH = SIM 5"
CDPIN3="OUT = 10"
```

If the pin components supplied with NX-CADAT were used, assign the CDPIN attribute to the pin component in DETL mode in the format:

```
CDPIN="[pinnumber] [DS = driver/sensor]"
```

where:

pinnumber is used to specify the order in which NX-CADAT lists the pins in the output file.

DS is a keyword that must be used when assigning a driver/sensor number.

driver/sensor is the driver/sensor number and is optional.

Example:

```
CDPIN="6 DS = 33"
```

CREATING CADAT-COMPATIBLE MODELS

NX-CADAT allows you to create CADAT-compatible models and use them in PC-CAPS schematics in the same way you use primitive symbols. CADAT recognizes three types of models: interconnect (subcircuit) models, behavioral models, and ROM models. This section describes these types of models and how to use them.

Subcircuit Models

A subcircuit model describes a schematic that can be used as a subcircuit in a schematic design or in another subcircuit model.

Before using a subcircuit model in a PC-CAPS schematic, you must create the subcircuit and its symbol and then create a model description file, as described below.

To create a subcircuit model and its symbol, enter the subcircuit in PC-CAPS DETL mode. Then change to SYMB mode and enter the symbol. Each net of the subcircuit that connects to a pin of the symbol must have the same name as the pin. If you plan to use the model in a packaged design, use the SCMD/PNLC command to assign packaging information to the symbol.

To create a model description file, run the schematic file through PC-NODES to produce a schematic netlist. Then, input the netlist file into NX-CADAT and select the Model Description Language file output option.

After you create the model description file, use the ADDMODEL command in the CADAT system's library mode to add the model to a CADAT model library.

Behavioral Models

A behavioral model describes the input/output behavior of a device rather than its internal elements and their interconnections.

> NOTE: Personal CADAT does not support the use of behavioral models.

NX-CADAT does not generate the Behavioral Model Description Language file. To use a behavioral model, you must use a text editing program to create this file and use the ADDBHMOD command of the CADAT system's library mode to add the file to the CADAT library.

To create a PC-CAPS schematic file to represent a behavioral model, create a symbol representation for the model and enter the following information.

- Use the CDTYPE attribute to enter the component type in the same way as for a primitive. The component type must be the same name specified in the TYPES subsection of the MODEL INFO section of the Behavioral Model Description Language file.

- Use the ENTR/PIN command to enter pins. You must enter the pins in the same sequence and with the same names as they have in the PINS subsection of the MODEL INFO section of the Behavioral Model Description Language file.

- If you plan to use the model in a packaged design, use the SCMD/PNLC command to assign packaging information.

ROM Models

A ROM model defines a ROM primitive element with a specified content.

NX-CADAT does not generate the ROM Model Description Language file. To use a ROM model, you must use a text editing program to create this file and use the ADDROM command of the CADAT system's library mode to add the file to the CADAT library.

To create a PC-CAPS schematic file to represent a ROM model, create a symbol representation for the model and enter the following information.

- Use the CDTYPE attribute to enter the component type in the same way as for a primitive. The component type must be the same name specified in the ROM NAME section of the ROM Description Language file.

- Use the ENTR/PIN command to enter pins. You must enter the pins in the same sequence and with the same names required by the CADAT system for the corresponding ROM primitive. (Refer to the CADAT manuals for more information).

- If you plan to use the model in a packaged design, use the SCMD/PNLC command to assign packaging information.

USING ALIASES WITH MODELS

An alias is a version of a model that is slightly different from the original model. CADAT allows you to associate up to 19 aliases with a subcircuit model. (Aliases are not used with ROM and behavioral models.)

All the differing information for all aliases is entered as attributes in the PC-CAPS circuit for the original model. You can then enter the symbol for the model in a schematic and specify whether it represents the original model or an alias of the model.

To indicate that a circuit has an alias and to assign a name to that alias, use the ATTR/ACOM command in PC-CAPS SYMB mode to assign the CDALIAa attribute to the circuit in the format:

CDALIAa="aliasname"

where:

a is a letter that specifies the particular alias.

aliasname is the name you assign to the alias.

Example:

CDALIAA="MODELA"
CDALIAM="MOSMODEL"

The differences between aliases are expressed as different values for three attributes: CDTIMa, CDFMLYa, and CDTHLa, which are described in "Assigning Attributes."

After you create a model with information for aliases, use NX-CADAT to create a model description file, and then use the ADDMODEL command in CADAT's library mode to add the model to a CADAT library.

To use an alias of a model in a schematic, place the symbol for the model in the schematic. Then, in PC-CAPS DETL mode, use the ATTR/ACOM or ATTR/SCHG command to enter or change the CDTYPE

attribute, giving it the same value as the CDALIAa attribute value of the appropriate alias.

ASSIGNING ATTRIBUTES

NX-CADAT uses several attributes as information in the CADAT-compatible output files. This section explains these attributes, except for the CDPIN, CDTYPE, and CDALIAa attributes, which are explained above. This section also gives the function of each attribute in the NX-CADAT output files.

All attributes used by NX-CADAT are summarized in Appendix C, "Attribute Summary."

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

To assign attributes, select the PC-CAPS ATTR/ACOM command in either SYMB or DETL mode as specified below. Type the attribute in the format:

```
key[a][n]="value"
```

where:

key is the attribute keyword, which names the attribute.

a is used only if the attribute is to be used for an alias of a model and is the letter that represents the alias. Only certain attributes can be assigned to aliases, as noted in the attributes explanations.

n is an optional sequence number. Sequence numbers are described below.

value is the contents of the attribute and must be enclosed in quotation marks. Square brackets ([ ]) are used to indicate that a part of the value is optional.

You can assign an attribute more than once to a component by appending a sequence number (n) from 0 to 99 to the keyword.

If the sequence numbers are consecutive (for example, the first n is 0, the next n is 1, and the next is 2), NX-CADAT appends each value to the previous one, to give one value for the keyword. This convention is useful when an attribute value is longer than the PC-CAPS limit (39 characters for both keyword and value).

When the sequence numbers are not consecutive (for example, the first n is 0 and the next n is 2), NX-CADAT creates multiple attribute values. NX-CADAT allows this usage only for certain attributes as noted in the explanations of the attributes.

You can assign an attribute several times to a component using both consecutive and nonconsecutive sequence numbers. For an example, see the explanation of the CDTIM attribute in "Attributes Assigned to Components."

NOTE: NX-CADAT recognizes a sequence number with or without a leading zero. For example, it considers 1 and 01 to be the same number, and it considers 01 and 2 to be consecutive numbers.

Attributes Assigned to a Schematic

The following attributes are assigned to the entire schematic in SYMB mode.

CDNAME - Specifies the circuit name, or the name of the schematic. The attribute value is used in the CIRCUIT NAME subsection of the HEADER section of either type of CADAT-compatible output file. If this attribute is not assigned, the filename of the schematic is used.

Example:

```
CDNAME="NX-CADAT Test Circuit"
```

CDTECH - Specifies the semiconductor technology used for the circuit. The attribute value is used in the TECHNOLOGY subsection of the HEADER section of either type of CADAT-compatible output file. This information is optional; if the technology is not specified, CADAT assumes it is TTL.

Example:

```
CDTECH="CMOS"
```

CDLU - Specifies the date and time of the last update to the circuit. The attribute value is used in the LAST UPDATE subsection of the HEADER section of either type of CADAT-compatible output file. If this attribute is not assigned, the date and time of the NX-CADAT program run are used.

Example:

```
CDLU="Aug 12 1986 11:57 AM"
```

CDDD - Specifies the default delay. This information is optional; if no delay is specified, CADAT uses 1 ns for the default delay. The attribute value is used in the DEFAULT DELAY subsection of the HEADER section of either type of CADAT-compatible output file. This attribute is entered in the format:

CDDD="<rise time>,<fall time>"

The rise and fall times must be entered according to CADAT conventions.

Example:

CDDD1="<20US,25US,30US>,<25US,"
CDDD2="30US,35US>"

CDDT - Specifies the default tolerances for minimum and maximum timing delays from a typical delay value in percentages. This information is optional; if no tolerance is specified, CADAT uses a tolerance of 0. The attribute value is used in the DEFAULT TOLERANCE subsection of the HEADER section of either type of CADAT-compatible output file. This attribute is entered in the format:

CDDT="<a,b>,<c,d>"

where:

a and **b** are the minus and plus tolerances for the rise time.

c and **d** are the minus and plus tolerances for the fall time.

Example:

```
CDDT1="<-5%,+10%>,"
CDDT2="<-20.51%,+15.97%>"
```

CDREV - Specifies revision or version of the circuit. The attribute value is used in the REVISION subsection of the HEADER section of either type of CADAT-compatible output file. This information is optional.

Example:

```
CDREV="2.2"
```

CDCOM - Specifies comments. The attribute value is used in the COMMENTS subsection of the HEADER section of either type of CADAT-compatible output file. This information is optional.

Example:

```
CDCOM="SAMPLE"
```

CDVECT - Specifies a signal vector, which is several signals under a common name. All vectors are shown in the VECTORS section of either type of CADAT-compatible output file. This attribute is entered in the format:

```
CDVECT="vector"
```

where:

vector is a CADAT signal vector.

PC-CAPS does not support vectors, so you must enter vector interconnections by entering the connections for each individual net within the vector and naming the

nets according to CADAT conventions for vector components.

You can enter several separate vectors by using nonconsecutive sequence numbers with this attribute.

Example:

```
CDVECT1="BUSA[1-3]"
CDVECT3="BUSB[4-7]"
```

CDLDD - Specifies load-dependent delay for a pin of a model. This information is used in the PINS subsection of the MODEL INFO section of the model description file. This attribute is entered in the format:

```
CDLDD="pinname load-dependent-delay"
```

where:

pinname is the name of the pin.

load-dependent-delay contains the LOAD identifier (for loading capacitance) and the DRIVE identifier (for drive capability) and the value.

You can specify more than one delay value by using nonconsecutive sequence numbers with this attribute.

Example:

```
CDLDD0="CLK DRIVE = 2.5, 3.0 ns/pF"
CDLDD2="CLK LOAD = 10 pF"
```

In this example, two values are specified for the pin named IN.

CDFMLY[a] - Specifies a family type to be associated with a model or alias of a model. The attribute value is used in the FAMILY subsection of the DEFAULTS subsection of the MODEL INFO section of the model description file. This information is optional.

Example:

```
CDFMLY="LSTTL"
CDFMLYB="LSTTL"
```

CDTHL[a] - Specifies threshold information for output pins of an IC subcircuit model or for an alias of such a model. The attribute value is used in the THRESHOLD subsection of the DEFAULTS subsection of the MODEL INFO section of the model description file. This information is optional.

You can specify more than one value for this attribute using nonconsecutive sequence numbers.

Example:

```
CDTHL1="1, 2, 3 = LSTTL"
CDTHL3="6 = .2, 1.8"
```

CDINFOF - Specifies the filename of a file containing text to replace the attributes defined for the SPECIAL INFO section of the network description file or the DEFAULTS and TYPES subsections of the MODEL INFO section of the model description file. This attribute is used when there is a large amount of text to include for these sections. The filename can include a directory path if the file is not in the directory from which NX-CADAT is run. Note that this information will override any other attributes that have been assigned for these sections.

Example:

CDINFOF="\PCAD\PROJ0\TEXT\COUNT.INF"

CDNOFT - Specifies terminal pins to be included in the NOFAULTS list. This attribute is assigned in SYMB mode to the schematic only when symbol pins are used. For more information, see the explanation of CDNOFT in "Attributes Assigned to Components."

Attributes Assigned to Components

The attributes described in this section are assigned to individual components. Normally you assign these attributes in DETL mode to an instance of a component on a schematic. You can also assign these attributes in SYMB mode to a symbol definition so that the attribute will apply to all instances of the symbol whenever it is used in a schematic.

Several of these attributes require you to use a dollar sign ($) to signify the instance name. NX-CADAT replaces the $ with the instance name in the output file. The instance name is either the name assigned with the PC-CAPS NAME/COMP command or (if NAME/COMP was not used for the component) the name assigned by PC-NODES.

CDLOCAL - Specifies a local definition of an INTERCONNECT (subcircuit), ROM, or BEHAVIORAL model to be used in the circuit. It is used only when you want to include a model that exists only for the circuit and is not in the CADAT model library. When this attribute is used, NX-CADAT inserts the entire file describing the model in the MODEL section of either type of CADAT-compatible output file, in the appropriate subsection (ROM MODEL, INTERCONNECT MODEL, or BEHAVIORAL MODEL).

This attribute is assigned in the format:

CDLOCAL="mtype filename"

where:

mtype specifies the type of model. The value must be RDL (for ROM), MDL (for INTERCONNECT), or BMD (for BEHAVIORAL).

filename names the file containing the description of the local model to be included. If necessary, it can include the path to the directory where the file is located.

Example:

CDLOCAL="RDL TESTFILE.ROM"

CDPART - Specifies the part function and/or location, in the format:

CDPART="[function] [location]"

where:

function is IC, INTERFACE, or SIMULATOR. The CADAT default function is IC.

location is used for IC and INTERFACE parts only and shows x y coordinates in the format:

LOC = (x, y)

Specifying a function other than IC causes NX-CADAT to include the component in the corresponding subsection (INTERFACE PARTS or SIMULATOR PARTS) of the PARTS section of either type of

CADAT-compatible output file. (These subsections are not created if no components have the function.)

Specifying a location causes NX-CADAT to include the location with the component in the PARTS section.

For a packaged design, the CDPART attribute must apply to an entire package. Before using CDPART, you must preassign the same reference designator and appropriate sections or package pin numbers to all the components of the package. Then you must assign the same value for the CDPART attribute to each component.

Example:

CDPART="INTERFACE LOC = (5, 10)"

CDNOFT - Specifies that the component, model, node attached to a pin of the component, or connection to or state of an internal device of the model is not to be considered for fault analysis. NX-CADAT includes this information in the NOFAULTS subsection of the SPECIAL INFO section of the network description file or the NOFAULTS subsection of the DEFAULTS subsection of the MODEL INFO section of the model description file.

NOTE: Personal CADAT does not support fault simulation.

The attribute value is the $ to signify the component instance name and, if applicable, the pin or internal device to be included in the nofault list. This value must follow CADAT conventions.

You can assign more than one attribute value to a component using nonconsecutive sequence numbers.

Example:

```
CDNOFT="ALL $"
```

CDTIM[a] - Specifies MOS decay timing for the component or alias or specifies delay overrides for one of the component's pins. NX-CADAT includes this information in the TIMING subsection of the SPECIAL INFO section of the network description file or the TIMING subsection of the DEFAULTS subsection of the MODEL INFO section of the model description file. This attribute is entered in the format:

```
CDTIM="$[.pin] = timing"
```

where:

$ signifies the instance name of the component.

pin is the number or name of the pin to which the information applies. It is present for delay information but omitted for MOS decay information.

timing is the rise and fall times to be used. It must follow CADAT conventions.

You can specify more than one attribute value using nonconsecutive sequence numbers.

Example:

```
CDTIMB0="$.Z4 = <27NS,28NS,30NS>,"
CDTIMB1="<10NS,13NS,17NS>"
CDTIMB3="$.Y = <20NS,22NS,27NS>,"
CDTIMB4="<10NS,18NS,99NS>"
```

This example produces one entry for each of two pins of alias B of the component to which the attribute is assigned:

```
compname.Z4 = <27NS,28NS,30NS>,<10NS,13NS,17NS>
compname.Y = <20NS,22NS,27NS>,<10NS,18NS,99NS>
```

CDLOAD - Specifies a modification in load characteristics (capacitance) associated with a node of a circuit. This attribute can be assigned to any component that has a pin associated with the node. NX-CADAT includes this information in the LOADING subsection of the SPECIAL INFO section of the network description file or the LOADING subsection of the DEFAULTS subsection of the MODEL INFO section of the model description file. This attribute is entered in the format:

```
CDLOAD="[NODE] $.pin = load"
```

where:

NODE is an optional keyword.

$ signifies the instance name of the component.

pin is the name or number of the pin attached to the node.

load is the load characteristic to be used. It must follow CADAT conventions.

You can specify more than one attribute value using nonconsecutive sequence numbers.

Example:

```
CDLOAD00="NODE $.3 = 0.05 pF"
CDLOAD02="$.OUT = 0.2 pF"
```

CDTECH - Specifies the technology for a primitive component or for an output pin or a bus pin of a component. The technology specified in this attribute will override the one specified in the TECHNOLOGY section of the HEADER section of the network description file. NX-CADAT includes this information in the TECHNOLOGY subsection of the SPECIAL INFO section of the network description file or the TECHNOLOGY subsection of the DEFAULTS subsection of the MODEL INFO section of the model description file. This attribute is entered in the format:

CDTECH="$[.pin] = technology"

where:

$ signifies the instance name of the component.

pin is the pin name or number. It is used only if the attribute applies to a pin rather than an entire primitive. The pin must be an output or bus pin.

technology is the technology to be used. It must follow CADAT conventions.

You can specify more than one attribute value using nonconsecutive sequence numbers.

Example:

CDTECH="$ = CMOS"

CHAPTER 3. PREPARING THE INPUT FILES

This chapter describes how to prepare the files that are to be input to NX-CADAT.

After your schematic is complete, you must use PC-NODES to extract the schematic netlist(s). If your schematic consists of just one schematic file, you can input the netlist directly into NX-CADAT.

If your design is a packaged design and has multiple sheets, you must use PC-LINK to combine the schematic netlists from all the sheets into one expanded netlist file.

If your design is unpackaged and has multiple sheets, you must create a batch control file to input into NX-CADAT. If you want to process several designs consecutively, you can also create a batch control file.

The following sections explain how to create and expand the binary netlist and how to create the batch control file.

EXTRACTING AND EXPANDING THE SCHEMATIC NETLIST

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

If your packaged design has multiple sheets, run PC-LINK to link together the netlists. Use the instructions in the *PC-LINK User's Manual.*

CREATING THE BATCH CONTROL FILE

The batch control file consists of one or more commands; each command tells NX-CADAT to process a design that has one or more sheets. Use a text editor to create the batch control file.

A batch control file can contain as many commands as necessary. The format of each batch control file command is shown below.

```
[ PACKAGED [REASSIGN] [PREFIX <prefix>] ]
<filetype> [<outfile>]
<netlists>
```

where:

PACKAGED specifies that the design is a packaged design. If this word is not present, NX-CADAT treats the design as an unpackaged design.

REASSIGN specifies that NX-CADAT is to assign packaging information to all components including those that were preassigned. REASSIGN is used only for packaged designs. If this word is not present, NX-CADAT uses the preassigned information and assigns new information only to components that were not preassigned.

PREFIX specifies that NX-CADAT is to use the following characters as a prefix when assigning reference designators. PREFIX is used only for packaged designs. If no prefix is specified, NX-CADAT uses the default prefix, UP.

prefix is the reference designator prefix. It can be one to three alphanumeric characters.

filetype is the type of output file to be created; it must be either NETWORK or MODEL.

outfile is the name of the output file. If no name is specified, NX-CADAT will assign the name of the input netlist file with the .CKT or .MDL extension. (For a multisheet design, the name of the first netlist file is used.)

netlists is a list of all the netlists in the design. For a packaged design, only one filename can be used. The list must begin on a separate line after the output file information. Filenames are separated by a blank space. A plus sign (+) after a blank space at the end of a line indicates that the list continues on the next line. If you do not enter the filename extension, NX-CADAT assumes it is .NLT for an unpackaged design or .XNL for a packaged design.

All the information before the netlist names must be on a single line.

The percent sign (%) can be used in the batch control file to indicate a comment. NX-CADAT ignores all text from the % to the end of the line.

You must use a .BCN filename extension when naming the batch control file.

Figure 3-1 shows an example of a batch control file.

```
% This example will create four separate CADAT files
%
MODEL TEST1.OUT    % single-sheet model with the
TEST 1             % output file named TEST1.OUT
%
NETWORK            % five-sheet design with the default
SH1 SH2 SH3 +      % output filename SH1.CKT
SH4 SH5
%
PACKAGED REASSIGN PREFIX tty NETWORK
TEST3              % packaged design with packaging
                   % reassigned, the prefix TTY, and the
                   % default output filename TEST3.CKT
%
PACKAGED NETWORK   % packaged design with the default
TEST4.NLT          % output filename TEST4.CKT
```

Figure 3-1. Sample Batch Control File

"Batch Mode" in Chapter 4 describes how to input the batch control file into NX-CADAT.

CHAPTER 4. USING NX-CADAT

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

Before running NX-CADAT, be sure that:

- Your system is correctly configured.
- You have installed the NX-CADAT program file (NXCADAT.EXE).
- You have assembled the schematic circuit.
- You have extracted and, if necessary, linked the input netlist files.

NX-CADAT has three operating modes: interactive, command line, and batch. In interactive mode, NX-CADAT displays a series of screens and you select options and specify filenames. In command line mode, you specify filenames and options on the command input line when you start the program and NX-CADAT processes the input file and produces the desired output file automatically. In batch mode, when you start the program you specify the name of a batch file that instructs NX-CADAT to process several files automatically.

The following sections describe how to use interactive mode to start, configure, and run NX-CADAT and how to use command line and batch modes.

INTERACTIVE MODE

In interactive mode, you must first start NX-CADAT to display the Opening Menu.

To start NX-CADAT, be sure you are in the appropriate project directory, then type:

NXCADAT [Return]

When the NX-CADAT Title Screen appears, press any key to continue. The system displays the Opening Menu as shown in Figure 4-1.

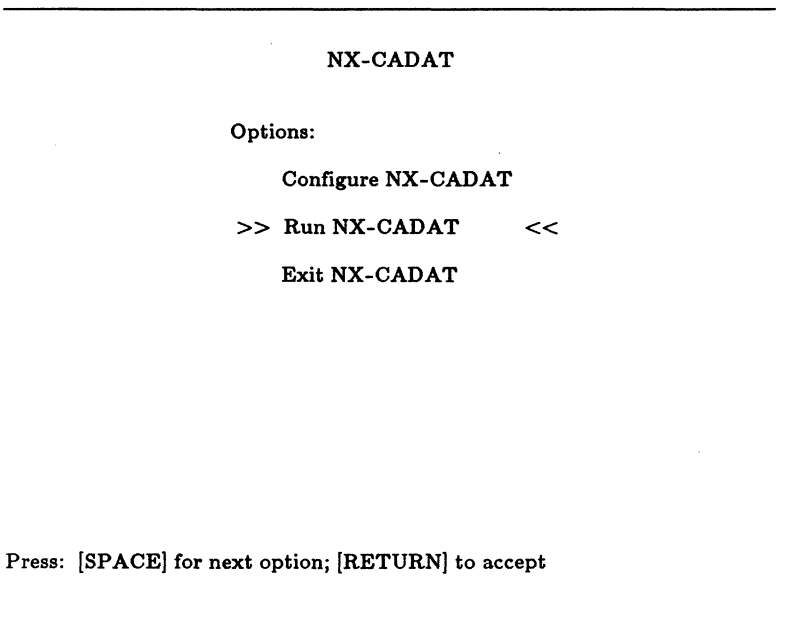


Figure 4-1. NX-CADAT Opening Menu

This menu provides the following options.

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

Run NX-CADAT - Allows you to translate the P-CAD netlist to a CADAT-format file.

Exit NX-CADAT - Returns you to DOS.

If you want to set or change the configuration, use the procedure in "Configuring NX-CADAT." If you do not want to set or change the configuration, continue to "Running NX-CADAT."

Configuring NX-CADAT

NX-CADAT allows you to set several options. These settings are stored in the configuration file, NXCADAT.CFG, which is used each time you run the program.

If you do not create the configuration file before the first time you run NX-CADAT, the system uses the default settings.

You can use **Configure NX-CADAT** to create the configuration file before the first time you run NX-CADAT, to change the settings in the configuration file, or to change the settings for just one operating session.

To configure NX-CADAT, follow the steps below.

1. At the NX-CADAT Opening Menu, press the space bar or use the cursor keys to select **Configure NX-CADAT**, then press [Return]. The system

displays the Configuration Screen with the default settings as shown in Figure 4-2.

NX-CADAT

Configuration

Bar Character: N

Part Pin Representation : part-name.pin-number

Type of Design : unpackaged

Default Prefix : UP

Enter the character; Press [Return] to accept; [Esc] to quit.

Figure 4-2. NX-CADAT Configuration Screen

2. The first configuration option allows you to select a character to replace the apostrophe to represent a bar. In PC-CAPS, any name terminated with an apostrophe is displayed with a bar over the entire name. The apostrophe is not a valid CADAT character; CADAT requires letters and digits only.

   NX-CADAT supports bar characters for signal names, component names, and pin names only.

The default character is "N." Press [Return] to accept the default or type a different character and press [Return].

3. The next option allows you to select the notation for part pins in the NX-CADAT output files. The settings are "part-name.pin-number" and "part-name.pin-name." The default is "part-name.pin-number."

   NOTE: This option applies only to unpackaged designs. For packaged designs, the part pin is always represented as "reference-designator.package-pin-number."

   Press [Return] to accept the default, or press the space bar to display the other setting and press [Return].

4. The next option is the type of design. The settings are "unpackaged" and "packaged." The default is "unpackaged."

   Press [Return] to configure NX-CADAT to process unpackaged designs. To configure NX-CADAT to process packaged designs, press the space bar to display the "packaged" option and press [Return].

5. The next option is the default prefix to be used when NX-CADAT creates reference designators in packaged designs. This prefix can be one to three letters and/or digits. The default is UP.

   Press [Return] to accept the default prefix or type another prefix and press [Return].

After you set the last option, the system displays the following prompt.

Save This Configuration? No

6. To save the configuration for this session and future sessions, type Y for Yes and press [Return]. The system saves the modified configuration to the NXCADAT.CFG file.

   To use the configuration for this session only, press [Return] to accept "No."

   When configuration is complete, the system returns you to the Opening Menu.

Running NX-CADAT

Run NX-CADAT allows you to generate CADAT files from your P-CAD netlist files. To run NX-CADAT, follow the steps below.

1. At the Opening Menu, use the space bar or the cursor keys to select **Run NX-CADAT** and press [Return].

   The system displays the NX-CADAT Program Screen and prompts for the input netlist filename as shown in Figure 4-3. This screen also shows the type of design selected in configuration.

NX-CADAT

[UNPACKAGED DESIGN]
Net List Filename : <Filename>.NLT

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

Figure 4-3. NX-CADAT Program Screen

2. Type the name of the netlist to be read by NX-CADAT and press [Return]. If you do not enter the filename extension, NX-CADAT assumes it is .NLT for an unpackaged design or .XNL for a packaged design.

   If your configuration is set for unpackaged designs, the next prompt does not appear and you can continue to step 4.

   If your configuration is set for packaged designs, the system prompts for packaging assignment. The default is "INCREMENTAL," which instructs NX-CADAT to use the preassignments made in

PC-CAPS and to assign packaging information only to the components that are not preassigned. The alternate setting is "REASSIGN ALL," which instructs NX-CADAT to assign new reference designators and package pin numbers to all components including those that were preassigned.

3. To use incremental assignment, press [Return]. To reassign packaging information to all components, press the space bar to display the "REASSIGN ALL" option and press [Return].

   The system prompts for the CADAT description language for the output file. The choices are "NETWORK" and "MODEL." The default is "NETWORK."

4. Press [Return] to accept "NETWORK" or press the space bar to display the "MODEL" choice and press [Return].

   The system prompts for the filename of the CADAT-compatible output file. The default is the input netlist filename with the .CKT or .MDL extension, depending on the CADAT description language you chose.

5. Press [Return] to accept the default or type another filename and press [Return].

   If your configuration is set for unpackaged designs, the next prompt does not appear.

   If your configuration is set for packaged designs, the system prompts for the name of the output back-annotation command file.

Figure 4-4 shows a sample screen using default values for a packaged design; for an unpackaged design, the screen is slightly different.

```
                              NX-CADAT

[PACKAGED DESIGN]
Net List Filename                 : TESTFILE.XNL
Packaging Assignment              : INCREMENTAL
CADAT Description Language        : NETWORK
CADAT Description Filename        : TESTFILE.CKT
Back Annotation Filename          : TESTFILE.CMD




Enter the filename; Press [Return] to accept.
```

Figure 4-4. Sample Program Screen

After you enter the last filename, NX-CADAT 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.

COMMAND LINE MODE

Command line mode allows you to specify input and output filenames and format when you start NX-CADAT.

To use command line mode, at the DOS prompt, type the command line and press [Return].

The format of the command line is shown below. Square brackets ([ ]) indicate optional information.

NXCADAT infile [outfile] [-M] [-P [-R] [prefix]]

where:

NXCADAT initiates the program.

infile is the name of the input netlist file. If you do not include the filename extension, NX-CADAT uses the .NLT extension for an unpackaged design or the .XNL extension for a packaged design.

outfile is the name of the output file to be created. If you specify a name with no extension, NX-CADAT uses the .MDL or .CKT extension, for the model or network description files respectively. If you do not specify a name, NX-CADAT uses the input netlist filename with the appropriate extension.

-M specifies that the output file is to be in the Model Description Language. If you do not use the -M, NX-CADAT will create the file in the Network Description Language.

-P specifies that the design is a packaged design. If you do not use the -P, NX-CADAT will check the

configuration file to determine whether the design is packaged or unpackaged.

-R specifies that NX-CADAT is to reassign packaging information to all components. The -R can only be used when the -P is used. If you do not use the -R for a packaged design, components that are preassigned will retain their preassignments.

prefix is the reference designator prefix NX-CADAT is to assign to all components that were not preassigned reference designators. You can only specify a prefix when the -P is used. If you do not specify a prefix for a packaged design, NX-CADAT uses the default prefix, UP.

Three examples of command lines are shown below.

```
NXCADAT TESTFILE
NXCADAT TESTFILE TEST.NEW -M
NXCADAT TESTFILE -P -R XY
```

The first example causes NX-CADAT to create a network description file from the TESTFILE.NLT file. The output file will have the default filename TESTFILE.CKT.

The second example causes NX-CADAT to create a model description file from the TESTFILE.NLT file and give the output file the filename TEST.NEW.

The third example causes NX-CADAT to create a network description file from the TESTFILE.XNL file, which is a packaged design; to reassign packaging information to all components; and to use XY as the reference designator prefix. The output file will have the default filename TESTFILE.CKT.

After you enter the command line, the program runs automatically with no further input from you. First, the NX-CADAT Title Screen appears, followed by the Command Line Mode Screen. Figure 4-5 shows a sample Command Line Mode Screen for an unpackaged design.

NX-CADAT

Type of Design	:	UNPACKAGED
Net List Filename	:	TESTFILE.NLT
CADAT Description Language	:	NETWORK
CADAT Description Filename	:	TESTFILE.CKT

Figure 4-5. Command Line Mode Screen

NX-CADAT displays progress reports and error messages, if any, as it processes the file. When processing is complete, the system returns you to DOS.

BATCH MODE

Batch mode is a variation of command line mode where the input file is a batch control file (<filename>.BCN) instructing NX-CADAT to process one or more designs, each of which can have one or more sheets.
The batch control file is an ASCII file that you create using a text editing program. Refer to Chapter 3, "Preparing the Input Files," for more information about the batch control file.

To input the batch control file, at the DOS prompt, type:

NXCADAT @<filename> [Return]

You do not need to enter the extension; NX-CADAT assumes it is .BCN.

NX-CADAT displays the Title Screen, then the Batch Mode Screen, which shows the same information as the Command Line Mode Screen. NX-CADAT processes the files sequentially, displaying each netlist filename in turn, and displays progress reports and error messages, if any. When processing is complete, the system returns you to DOS.

Batch mode produces all the output files specified in the batch control file.

UPDATING A SCHEMATIC

After you process a packaged design, you can input the back-annotation command file to PC-CAPS to update the original schematic with the packaging information added by NX-CADAT.

To update a schematic, follow the steps below.

1. Start PC-CAPS and select **Edit database** at the Opening Menu.

   The PC-CAPS Command Menu appears.

2. Type:

   /

   The system prompts:

   Menu command:

3. Type:

   EXE [Return]

   The system prompts:

   Exec filename:

4. Type:

   <filename>.CMD [Return]

   The system loads the appropriate schematic and updates each symbol on the schematic to its new physical configuration. If the schematic is a multisheet schematic, each sheet is automatically updated and saved in turn.

CHAPTER 5. VIEWING THE OUTPUT FILES

NX-CADAT has five types of ASCII output files, the CADAT-compatible network description file (<filename>.CKT), the CADAT-compatible model description file (<filename>.MDL), the CADAT-compatible digital stimulus language file (NXCADAT.DSL), the log file (<filename>.LOG), and the back-annotation command file (<filename>.CMD).

This chapter describes how to view and print the output files and how to interpret the CADAT-format files.

The back-annotation command file is in standard P-CAD executable format, which is explained in the *PC-CAPS User's Manual.*

Appendix C, Attribute Summary, shows sample circuit description, model description, and back-annotation command files.

The log file contains error and information messages displayed during NX-CADAT operation. Appendix A explains the error messages.

VIEWING AND PRINTING FILES

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

TYPE TESTFILE.LOG [Return]

or, to print the 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.

NETWORK DESCRIPTION FILE

The network description file (<filename>.CKT) can contain up to seven sections.

This section describes the information in each section of the network description file and tells where the information originates in the PC-CAPS schematic. For a more detailed description, refer to the appropriate CADAT manual.

Square brackets ([ ]) around a subsection name indicate that the subsection is optional and is not produced if NX-CADAT does not find the appropriate information.

> NOTE: By CADAT convention, each section or subsection name and each line of data ends with a dollar sign ($). The dollar signs are not shown here.

HEADER Section

The HEADER section is the first section of the file. It contains information about the circuit as a whole. It consists of several subsections, which are described below. The file includes each subsection name on a line before the accompanying information.

CIRCUIT NAME - The name of the circuit, specified using the CDNAME attribute. If the attribute is not assigned, NX-CADAT uses the filename of the schematic.

[TECHNOLOGY] - The semiconductor technology used for the circuit, specified using the CDTECH attribute.

LAST UPDATE - The most recent circuit revision, specified using the CDLU attribute. If the attribute is not assigned, NX-CADAT uses the date and time of the current NX-CADAT program run.

[DEFAULT DELAY] - The delay, overriding the CADAT default delay of 1 ns, specified using the CDDD attribute.

[DEFAULT TOLERANCE] - Tolerances, in percentages, for minimum and maximum timing delays from a typical delay value, specified using the CDDT attribute.

[REVISIONS] - Circuit revision version, specified using the CDREV attribute.

[COMMENTS] - General comments, specified using the CDCOM attribute.

MODEL Section

The MODEL section shows local definitions of ROM, interconnect (subcircuit), and behavioral models. Each symbol that represents a local model must be assigned the CDLOCAL attribute in PC-CAPS DETL mode, with a value that is the filename of the model description file. The specified model description file is included in the appropriate subsection of the MODEL section.

These subsections, ROM MODEL, INTERCONNECT MODEL and BEHAVIORAL MODEL, are explained below.

[ROM MODEL] - The ROM description file named using the CDLOCAL attribute.

[INTERCONNECT MODEL] - The model description file named using the CDLOCAL attribute.

[BEHAVIORAL MODEL] - The behavioral model description file named using the CDLOCAL attribute.

PARTS Section

The PARTS section lists all the components in the circuit and their types. It may contain up to three subsections, IC PARTS, INTERFACE PARTS, and SIMULATOR PARTS, which are described below. If only IC parts are used, none of the subsection headings are used.

IC PARTS - All components that are assigned the CDPART attribute with a value of IC. Each component is defined in the format:

compname type [LOC = (x, y)]

where:

compname is the name of the component, assigned using the NAME/COMP command in DETL mode or assigned by PC-NODES in the format UCsssnnn.

type is the component type, assigned using the CDTYPE attribute. If the attribute is not assigned, the name of the component symbol file is used.

x, y are the x and y coordinates of the component's location, specified in the CDPART attribute.

[INTERFACE PARTS] - All components that are assigned the CDPART attribute with a value of INTERFACE. Each component is defined in the same format used in the IC PARTS section.

[SIMULATOR PARTS] - All components that are assigned the CDPART attribute with a value of SIMULATOR. Each component is defined in the same format used in the IC PARTS section except that the location option is not used.

VECTORS Section

The VECTORS section lists any vectors specified using the CDVECT attribute. For each vector, the value entered for the attribute is given. If the attribute is not used in the circuit, this section is not generated.

CONNECTIONS Section

The CONNECTIONS section lists all the circuit interconnections. Each node is defined in the format:

compname.pin compname.pin ... [signal]

where:

compname is the name of the component. For an unpackaged design, this name is assigned using the NAME/COMP command in DETL mode or assigned by PC-NODES in the format UCsssnnn. For a packaged design, this name is the reference

designator, either preassigned in PC-CAPS (using the SCMD/PNUM command) or assigned by NX-CADAT.

pin is the pin name or pin number. For an unpackaged design, it can be either the name or number (both assigned using the ENTR/PIN command in SYMB mode), depending on the part pin representation option chosen during NX-CADAT configuration. For a packaged design, it is the package pin number, either preassigned in PC-CAPS (using the SCMD/PNUM command) or assigned by NX-CADAT.

signal is the name of the signal, assigned using the NAME/NET command in DETL mode. (Signal names assigned by PC-NODES to unnamed nets do not appear here; the NAME/NET command must be used for the signal name to be included.)

EXTERNALS Section

The EXTERNALS section identifies the primary inputs and outputs to the circuit. Each pin is listed in the format:

pinname [type] [driver/sensor] signal

where:

pinname is the name of the pin

type is the pin type, assigned using the ENTR/PIN command in PC-CAPS SYMB mode or assigned using the CDPIN attribute.

driver/sensor is the driver/sensor number assigned to the pin using the CDPIN attribute.

signal is the name of the signal or part pin to which the pin is connected, assigned using the NAME/NET command in PC-CAPS DETL mode or expressed in the format part.pin.

SPECIAL INFO Section

The SPECIAL INFO section contains special information about the circuit in up to four subsections, NOFAULTS, TIMING, LOADING, and TECHNOLOGY, which are described below. If the circuit does not contain information for any of these subsections, this section is not generated.

[NOFAULTS] - A list of the nodes, model connections, models, and states internal to devices that are not to be considered for fault analysis. The list includes each item identified with the CDNOFT attribute, assigned to the appropriate component in DETL mode or assigned to the external name in SYMB mode.

[TIMING] - MOS decay timing and delay overrides for components or component pins, assigned to the component using the CDTIM attribute.

[LOADING] - Modifications to load characteristics (capacitance) for nodes, assigned to a component connected to the node using the CDLOAD attribute.

[TECHNOLOGY] - The technology for components or component pins (overriding the technology specified in the HEADER section), assigned to the component using the CDTECH attribute.

MODEL DESCRIPTION FILE

The model description file (<filename>.MDL) can contain up to six sections.

The first five sections, HEADER, MODEL, PARTS, VECTORS, and CONNECTIONS, are identical to the sections with the same name in the network description file. Refer to the explanation of the section in "Network Description File" for further information.

The last section, MODEL INFO, contains three main subsections. The information in these subsections and its origin in the PC-CAPS schematic are described below.

The notation used for the network description file is also used for the model description file.

PINS Subsection

This subsection lists the pins on the model and associates each pin with the interconnections in the circuit. Each pin is listed in the following format.

pinnumber:pinname connect [delay]

where:

pinnumber shows the sequence in which the pin was entered in PC-CAPS with the ENTR/PIN command. The first pin entered is pin 1; it is listed first, and the rest of the pins are numbered and listed sequentially.

pinname is the name assigned to the pin using the ENTR/PIN command.

connect is the name of the node to which the pin is connected, assigned using the NAME/NET command.

delay is the load-dependent delay or drive capability specified for the pin using the CDLDD attribute. If the attribute is not assigned, no load-dependent delay is displayed.

DEFAULTS Subsection

This subsection lists any special information for the model. It is very similar to the SPECIAL INFO section of the network description file. It consists of six subsections, which are described below. If the circuit does not contain information for any of the six subsections, the entire subsection is not generated.

[NOFAULTS] - This subsection is the same as the NOFAULTS subsection of the SPECIAL INFO section of the network description file.

[TIMING] - This subsection is the same as the TIMING subsection of the SPECIAL INFO section of the network description file.

[LOADING] - This subsection is the same as the LOADING subsection of the SPECIAL INFO section of the network description file.

[FAMILY] - Family type of the model, assigned to the model using the CDFMLYa attribute.

[THRESHOLD] - Threshold information for output pins of an IC subcircuit model, assigned to the model using the CDTHLa attribute.

[TECHNOLOGY] - This subsection is the same as the TECHNOLOGY subsection of the SPECIAL INFO section of the network description file.

TYPES Subsection

This subsection gives the name that identifies the model in the PARTS section of a network description file or of another model description file. It also gives alias names for the model and special information for the aliases.

The names are listed in the following format.

NAME = name[, alias, alias, ...]

where:

name is the component type specified using the CDTYPE attribute.

alias is an alias name specified using the CDALIAa attribute. A model can be assigned up to 19 aliases.

For each alias, a special subsection with up to three subsections may exist with information specific to that alias.

Each alias subsection begins with the name of the alias in the format:

NAME = alias

This line is followed by at least one of the following subsections: TIMING, FAMILY, or THRESHOLD, described below. (If none of this information is assigned, the alias subsection is not generated.)

[TIMING] - MOS decay timing and delay overrides for this alias of the model, specified using the CDTIMa attribute.

[FAMILY] - The family type for this alias of the model, specified using the CDFMLYa attribute.

[THRESHOLD] - Threshold information for this alias of the model, specified using the CDTHLa attribute.

DIGITAL STIMULUS FILE

Whenever NX-CADAT creates a circuit description file, it also creates the framework for the CADAT digital stimulus language (DSL) file for the circuit. This file contains the CIRCUIT, TIMEDEF, OUTPUT, INPUT, BUS (if applicable), and END statements. You can add to this file as required for your CADAT simulation purposes.

The DSL file created by NX-CADAT is always named NXCADAT.DSL. You should rename this file before running NX-CADAT again to produce a circuit description file; otherwise the DSL file will be overwritten.

The sample DSL file included with NX-CADAT is shown in Appendix D, "Sample Files."

APPENDIX A. ERROR MESSAGE

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

This appendix lists and explains NX-CADAT error messages. Errors are grouped into three types: file access and system errors, command line/batch control file errors, and netlist processing errors. Within each group, error messages are listed in alphabetical order.

FILE ACCESS AND SYSTEM ERRORS

Errors in this group are due to system problems. You cannot run the program until the error is corrected.

Message: Cannot create DSL file

Cause: Your system might have insufficient memory or an inadequate CONFIG.SYS file.

Action: Be sure your system has 640K of memory available. Make sure the value of the files parameter in the CONFIG.SYS file is 15 or greater.

Message: Database error (<filename>)

Cause: The netlist named is unusable.

Action: Rerun PC-NODES to generate a new netlist file. If the error persists, check the schematic database file and correct it if necessary.

Message: Disk is full

Cause: Your disk does not have enough space to store the output file.

Action: Delete files as necessary to make space.

Message: Error opening back-annotation file

Error opening log file <filename>

Error opening work file

Cause: These errors can be caused by insufficient memory or by an inadequate CONFIG.SYS file.

Action: Be sure your system has 640K of memory available. Make sure the value of the FILES parameter in the CONFIG.SYS file is 15 or greater.

Message: File not found : <filename>

Cause: NX-CADAT cannot locate the named file.

Action: Make sure that you enter the correct filename and use a drive designator or directory path if necessary.

Message: Filenames must be different

Cause: The output filename you specified is the same as the input filename.

Action: Use a different extension for the output filename.

Message: Memory allocation error

Cause: The system might have insufficient memory. NX-CADAT requires 640K.

Action: Be sure you have 640K. If so, reboot your system. If the error recurs, call P-CAD.

Message: Reserved filename <filename>

Cause: NX-CADAT reserves certain special filenames for its own use.

Action: Use a different filename for the file.

Message: System Error
Free() Error

Cause: The system might have insufficient memory. NX-CADAT requires 640K.

Action: Be sure you have 640K. If so, reboot your system. If the error recurs, call P-CAD.

Message: Unable to process <filename>

Cause: This message accompanies several other error messages.

Action: Correct the error specified in the accompanying error message.

COMMAND LINE/BATCH CONTROL FILE ERRORS

The following errors occur during batch processing.

Message: Batch control file <filename> not found!

Cause: NX-CADAT cannot find the named file.

Action: Be sure to enter the correct filename, including the directory path or drive designator if necessary.

Message: Default prefix <prefix> too long!

Cause: The reference designator prefix specified in the command line or in the batch control file is longer than three characters. NX-CADAT cannot process the design.

Action: Correct the command line or the batch control file, and rerun NX-CADAT.

Message: Error accessing <filename> net-list file

Cause: The named netlist file, which is listed in the batch control file, is unusable.

Action: Be sure the batch control file specifies the correct file. If necessary, rerun PC-NODES to generate a new netlist file from the schematic. If the error persists, check the schematic database file and correct it if necessary.

Message: Incompatible database format

Cause: The input file is damaged.

Action: Be sure the filename is correct. If necessary, rerun PC-NODES to produce a new netlist, then rerun NX-CADAT.

Message: Invalid character in prefix

Cause: The reference designator prefix specified in the command line or in the batch control file contains a character that is not a letter or digit. NX-CADAT cannot process the design.

Action: Correct the command line or the batch control file, and rerun NX-CADAT.

Message: Missing NETWORK or MODEL keyword

Cause: The batch control file does not contain either of the required keywords.

Action: Be sure you are using the correct batch control file. Edit the batch control file if necessary.

NETLIST PROCESSING ERRORS

The following messages describe errors in the construction of the input database.

Message: Duplicate reference designator

Cause: The same reference designator was assigned either to more than one type of component or to more components than allowed in the part.

Action: NX-CADAT reassigns the component(s) in error. Check the CADAT-compatible output file to see if the reassignment is acceptable.

Message: Element name same as circuit name

Cause: The component type of a component is the same as the component type of the schematic.

Action: Use the PC-CAPS ATTR/SCHG command to change the CDTYPE attribute for the component or the schematic, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: External name <name> connected to PWR, VCC or GND

Cause: The named external in the database is connected to power or ground. CADAT does not allow this type of connection.

Action: Use PC-CAPS to correct the schematic, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Error in assignment

Cause: Either a component is preassigned to a nonexistent section of a package or two components are assigned to the same section of a package.

Action: NX-CADAT reassigns the component in error. Check the CADAT-compatible output file to see if the reassignment is acceptable.

Message: INFO file <filename> not found

Cause: NX-CADAT cannot locate the file specified using the CDINFOF attribute.

Action: Be sure the correct filename is specified in the CDINFOF attribute, including a directory path or drive designator if necessary. If necessary, use the PC-CAPS ATTR/SCHG command to correct the attribute, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Invalid syntax in CDPIN

Cause: The syntax used in the value of the CDPIN attribute does not follow NX-CADAT requirements.

Action: Use the PC-CAPS ATTR/SCHG command to correct the attribute value, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Local model file <filename> not found

Cause: NX-CADAT cannot locate the local model file specified using the CDLOCAL attribute.

Action: Be sure the correct filename is specified in the CDLOCAL attribute, including a directory path or drive designator if necessary. If necessary, use the PC-CAPS ATTR/SCHG command to correct the attribute, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: No symbol pins defined in the model

Cause: The database does not have terminal pins entered as required for producing a model description file. (NX-CADAT does not allow use of pin components for a model description file.)

Action: Use the PC-CAPS ENTR/PIN command in SYMB mode to enter terminal pins, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Unconnected pin

Cause: The database has an unconnected external. CADAT does not allow an unconnected external in its network description file.

Action: Use PC-CAPS to correct the schematic, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Unpackaged component found

Cause: NX-CADAT found a component without packaging information in a packaged design. NX-CADAT cannot continue processing.

Action: Use PC-CAPS to add packaging information to the component. Then use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Unrecognized local model type <type>

Cause: The local model type specified using the CDLOCAL attribute is not MDL, BMD, or RDL.

Action: Use PC-CAPS to correct the attribute in the schematic, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Warning -- Circuit symbol is defined

Cause: The schematic contains both pin components and symbol pins. NX-CADAT treated the pin components as regular components.

Action: Check the output report, noting the connectivity between symbol pins and the circuit. If necessary, use PC-CAPS to correct the schematic, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Warning -- Missing '$' attribute <key> in component <compname>

Cause: The value of the specified attribute for the named component does not contain the $ character to signify the component instance name. The attribute value is not included in the CADAT-compatible output file.

Action: Use PC-CAPS to correct the attribute value, then rerun PC-NODES and NX-CADAT.

Message: Warning -- No external pins defined

Cause: No terminal pins are defined in the database.

Action: If necessary, use PC-CAPS to add pins to the schematic, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Warning -- Nonconsecutive pin numbers

Cause: The pin numbers assigned to pin components using the CDPIN attribute are not consecutive.

Action: If the pins are numbered as desired, no action is necessary. If there is an error in pin numbering, use the PC-CAPS ATTR/SCHG command to change the pin number specified in the CDPIN attribute for each affected pin, then use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Warning -- Number of aliases exceeds limit

Cause: More than nineteen aliases are specified for the component. NX-CADAT ignores all aliases after the nineteenth.

Action: If necessary, use PC-CAPS to correct the schematic, use PC-NODES to extract a corrected netlist, and rerun NX-CADAT.

Message: Warning - Packaged component found

Cause: NX-CADAT found a component with packaging information in an unpackaged design. The packaging information is ignored.

Action: Check your design to be sure it is correct. If so, no action is necessary.

Message: Warning -- Unused attribute <key> in component <compname>

Cause: The specified attribute, which begins with CD but is not a valid NX-CADAT attribute keyword, was assigned to the named component.

Action: Be sure the attribute keyword is the one you wanted to enter. If necessary, use PC-CAPS to delete the attribute and enter a corrected attribute, then rerun PC-NODES and NX-CADAT.

APPENDIX B. NX-CADAT SCHEMATIC SYMBOLS

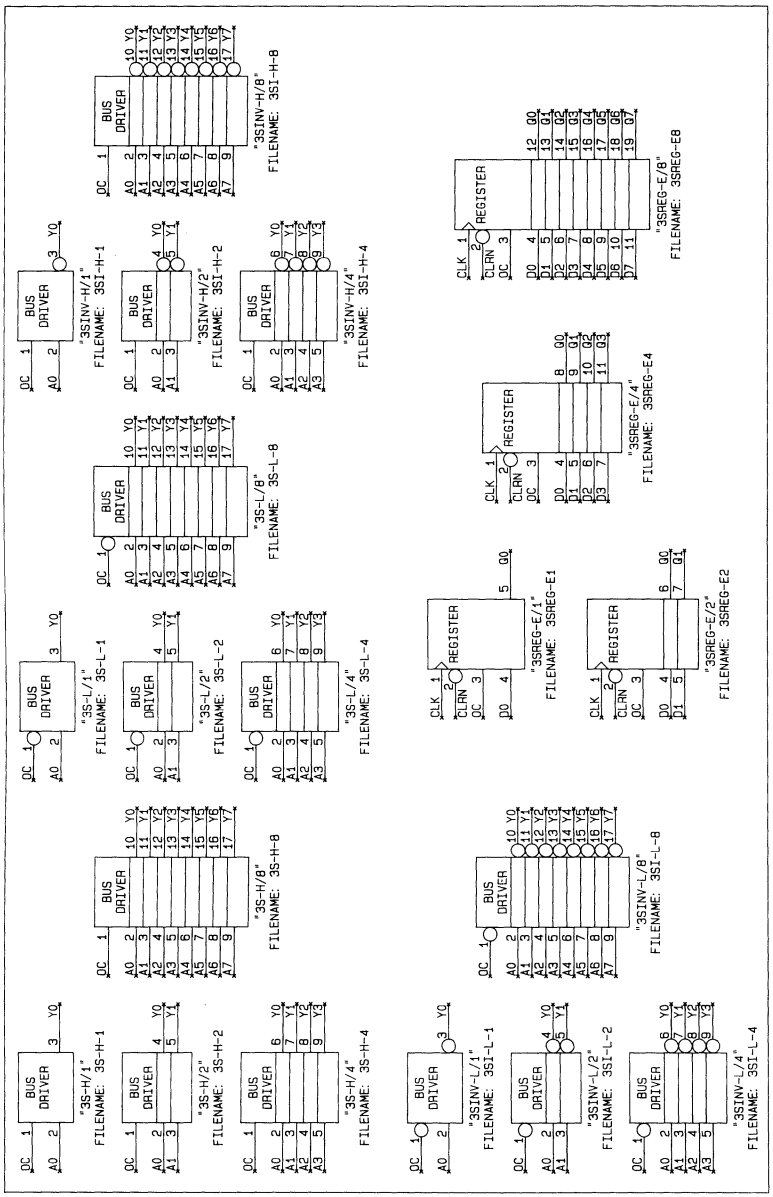
This appendix lists and shows the PC-CAPS schematic symbols in the library supplied with NX-CADAT. The library includes symbols for over 200 CADAT primitives and for the pin component symbols described in Chapter 2, "Preparing the Schematic Database." The library also includes the LAYS.SCH file, which is the layer structure file used to create the symbols.

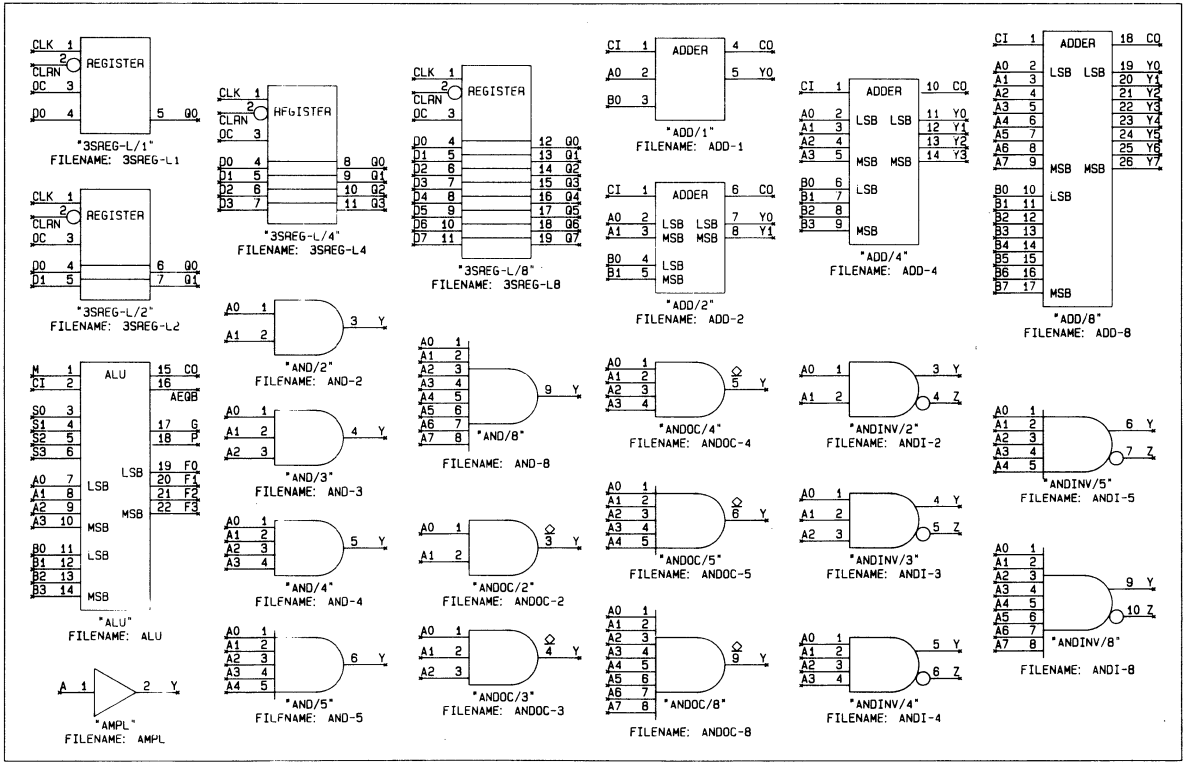
The P-CAD filenames (without the .SYM extension) of all the symbols are shown in the following list. This list is followed by a series of plots that include all the symbols. The plot for each symbol shows the name of the corresponding CADAT primitive (as assigned using the CDTYPE attribute) directly below the symbol. The pin name is shown next to each pin, and the pin order is shown by numbers next to the pin names.

3S-H-1	ANDI-5	CNTDAA	NANDOC-5
3S-H-2	ANDI-8	CNTDAS	NANDOC-8
3S-H-4	ANDIOC-2	CNTDSS	NFET
3S-H-8	ANDIOC-3	COMPAR-4	NFETD
3S-L-1	ANDIOC-4	COMPAR-8	NLOAD
3S-L-2	ANDIOC-5	DECBCD	NMOS
3S-L-4	ANDIOC-8	DECBIN-1	NMOSD
3S-L-8	ANDN-1	DECBIN-2	NOR-2
3SI-H-1	ANDN-2	DECBIN-3	NOR-3
3SI-H-2	ANDN-3	DECBIN-4	NOR-4
3SI-H-4	ANDN-4	DELAY	NOR-5
3SI-H-8	ANDNG-1	DFF-E	NOR-8
3SI-L-1	ANDNG-2	DFF-L	NORLCH
3SI-L-2	ANDNGO-1	ENCBCD	NOROC-2
3SI-L-4	ANDNGO-2	ENCBIN-4	NOROC-3
3SI-L-8	ANDNOC-1	ENCBIN-8	NOROC-4
3SREG-E1	ANDO-1	INV	NOROC-5
3SREG-E2	ANDOC-2	INVOC	NOROC-8
3SREG-E4	ANDOC-3	JKFF-E	NXOR
3SREG-E8	ANDOC-4	JKFF-L	NXOROC
3SREG-L1	ANDOC-5	JKFF-S	ONESHOT
3SREG-L2	ANDOC-8	MUX1-2-4	OR-2
3SREG-L4	ANDOOC-1	MUX1-2-8	OR-3
3SREG-L8	BUF	MUX2-4-2	OR-4
ADD-1	BUFOC	N1FET	OR-5
ADD-2	C1MOS	N1FETD	OR-8
ADD-4	C1MOSD	N1LOAD	ORA-1
ADD-8	CMOS	N1MOS	ORAOC-1
ALU	CMOSD	N1MOSD	ORI-2
AMPL	CNTBAA-4	NAND-2	ORI-3
AND-2	CNTBAA-6	NAND-3	ORI-4
AND-3	CNTBAA-8	NAND-4	ORI-5
AND-4	CNTBAS-4	NAND-5	ORI-8
AND-5	CNTBAS-6	NAND-8	ORIOC-2
AND-8	CNTBAS-8	NANDLC	ORIOC-3
ANDI-2	CNTBSS-4	NANDOC-2	ORIOC-4
ANDI-3	CNTBSS-6	NANDOC-3	ORIOC-5
ANDI-4	CNTBSS-8	NANDOC-4	ORIOC-8

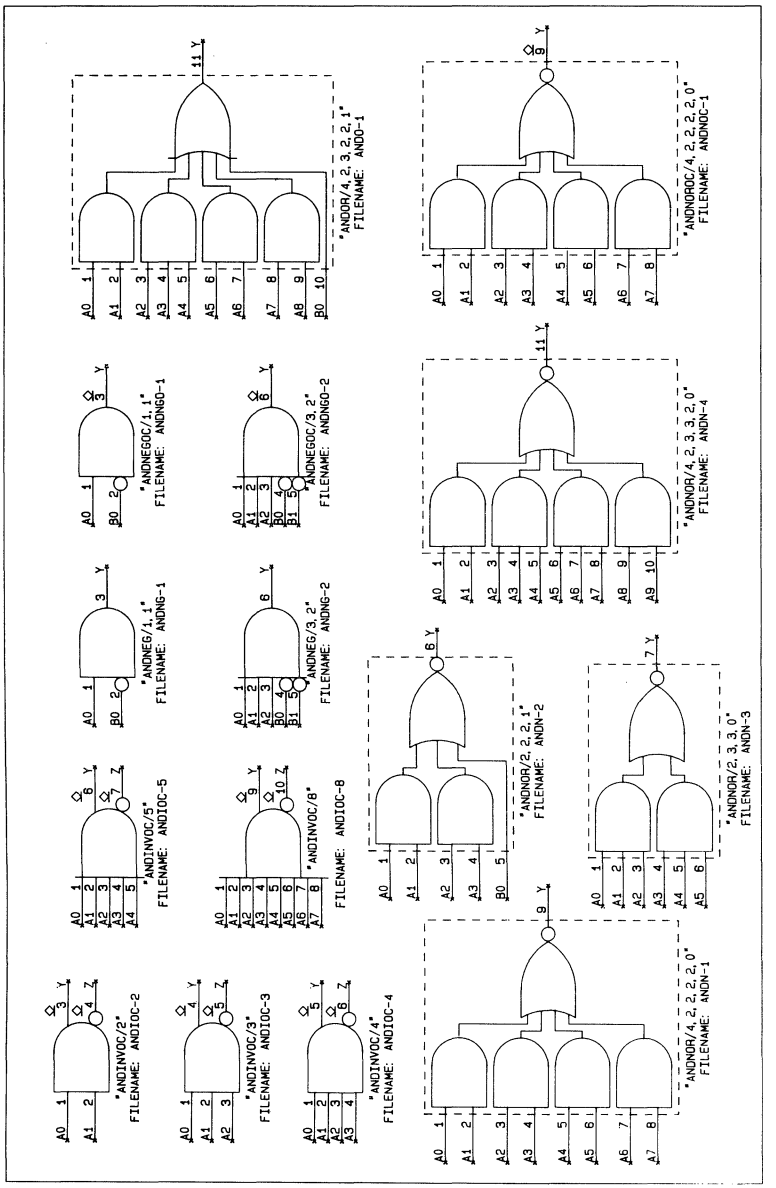
ORNA-1
ORNAOC-1
ORNG-1
ORNGO-1
OROC-2
OROC-3
OROC-4
OROC-5
OROC-8
P1FET
P1FETD
P1LOAD
P1MOS
P1MOSD
PARIT
PFET
PFETD
PINBUS
PININ
PINOUT
PINSIM
PLOAD
PMOS
PMOSD
PULLDOWN
PULLR
PULLUP
RAM-11-8
REG-E-1
REG-E-2
REG-E-4
REG-E-8
REG-L-1
REG-L-2
REG-L-4
REG-L-8
RESIST
ROM-13-8
SEL-1-2
SEL-2-4
SEL-3-8
SEL-4-16
SEL-5-32
SR-P-4
SR-P-5
SR-P-8
SR-S-4
SR-S-5
SR-S-8
UNIDELAY
UNIRES
XCVR-2
XCVR-4
XCVR-8
XCVR-I-2
XCVR-I-4
XCVR-I-8
XOR

Plot 1

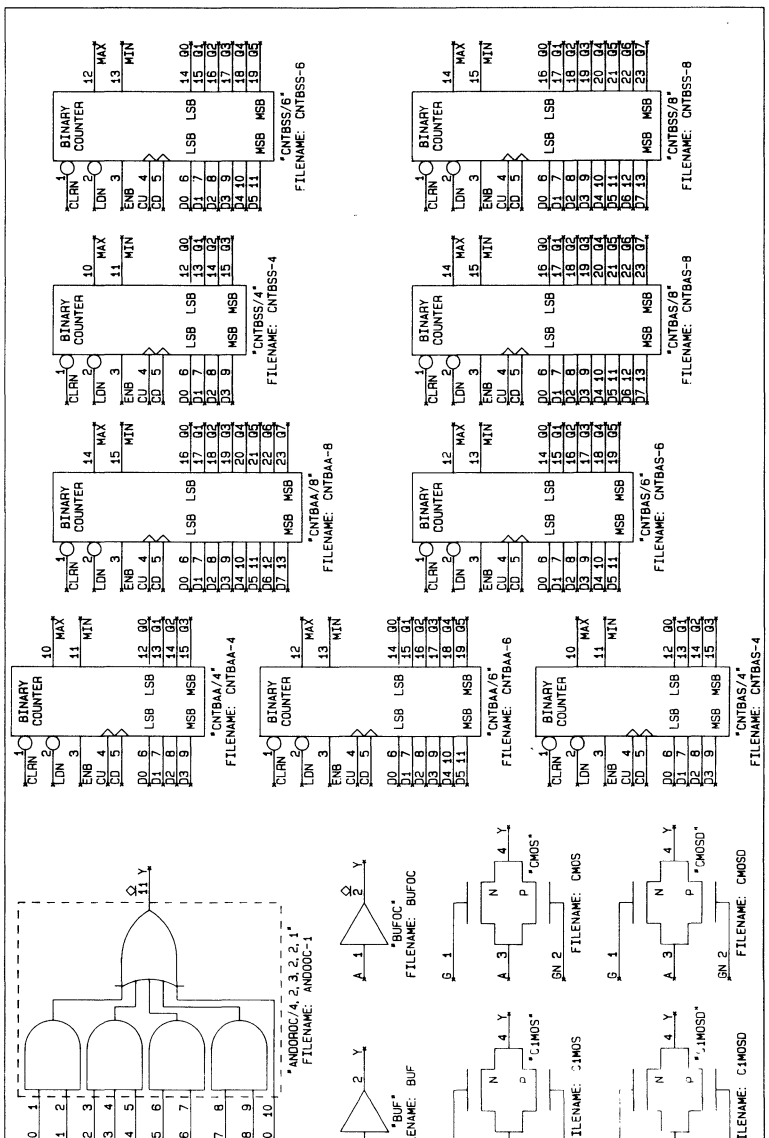

BUS DRIVER
REGISTER
"3S-H/1" FILENAME: 3S-H-1
"3S-H/2" FILENAME: 3S-H-2
"3S-H/4" FILENAME: 3S-H-4
"3S-H/8" FILENAME: 3S-H-8
"3S-L/1" FILENAME: 3S-L-1
"3S-L/2" FILENAME: 3S-L-2
"3S-L/4" FILENAME: 3S-L-4
"3S-L/8" FILENAME: 3S-L-8
"3SINV-H/1" FILENAME: 3SI-H-1
"3SINV-H/2" FILENAME: 3SI-H-2
"3SINV-H/4" FILENAME: 3SI-H-4
"3SINV-H/8" FILENAME: 3SI-H-8
"3SINV-L/1" FILENAME: 3SI-L-1
"3SINV-L/2" FILENAME: 3SI-L-2
"3SINV-L/4" FILENAME: 3SI-L-4
"3SINV-L/8" FILENAME: 3SI-L-8
"3SREG-E/1" FILENAME: 3SREG-E1
"3SREG-E/2" FILENAME: 3SREG-E2
"3SREG-E/4" FILENAME: 3SREG-E4
"3SREG-E/8" FILENAME: 3SREG-E8


"3SREG-L/1"
FILENAME: 3SREG-L1
"3SREG-L/2"
FILENAME: 3SREG-L2
"3SREG-L/4"
FILENAME: 3SREG-L4
"3SREG-L/8"
FILENAME: 3SREG-L8
"ALU"
FILENAME: ALU
"AMPL"
FILENAME: AMPL
"AND/2"
FILENAME: AND-2
"AND/3"
FILENAME: AND-3
"AND/4"
FILENAME: AND-4
"AND/5"
FILENAME: AND-5
"AND/8"
FILENAME: AND-8
"ANDOC/2"
FILENAME: ANDOC-2
"ANDOC/3"
FILENAME: ANDOC-3
"ADD/1"
FILENAME: ADD-1
"ADD/2"
FILENAME: ADD-2
"ADD/4"
FILENAME: ADD-4
"ADD/8"
FILENAME: ADD-8
"ANDOC/4"
FILENAME: ANDOC-4
"ANDOC/5"
FILENAME: ANDOC-5
"ANDOC/8"
FILENAME: ANDOC-8
"ANDINV/2"
FILENAME: ANDI-2
"ANDINV/3"
FILENAME: ANDI-3
"ANDINV/4"
FILENAME: ANDI-4
"ANDINV/5"
FILENAME: ANDI-5
"ANDINV/8"
FILENAME: ANDI-8

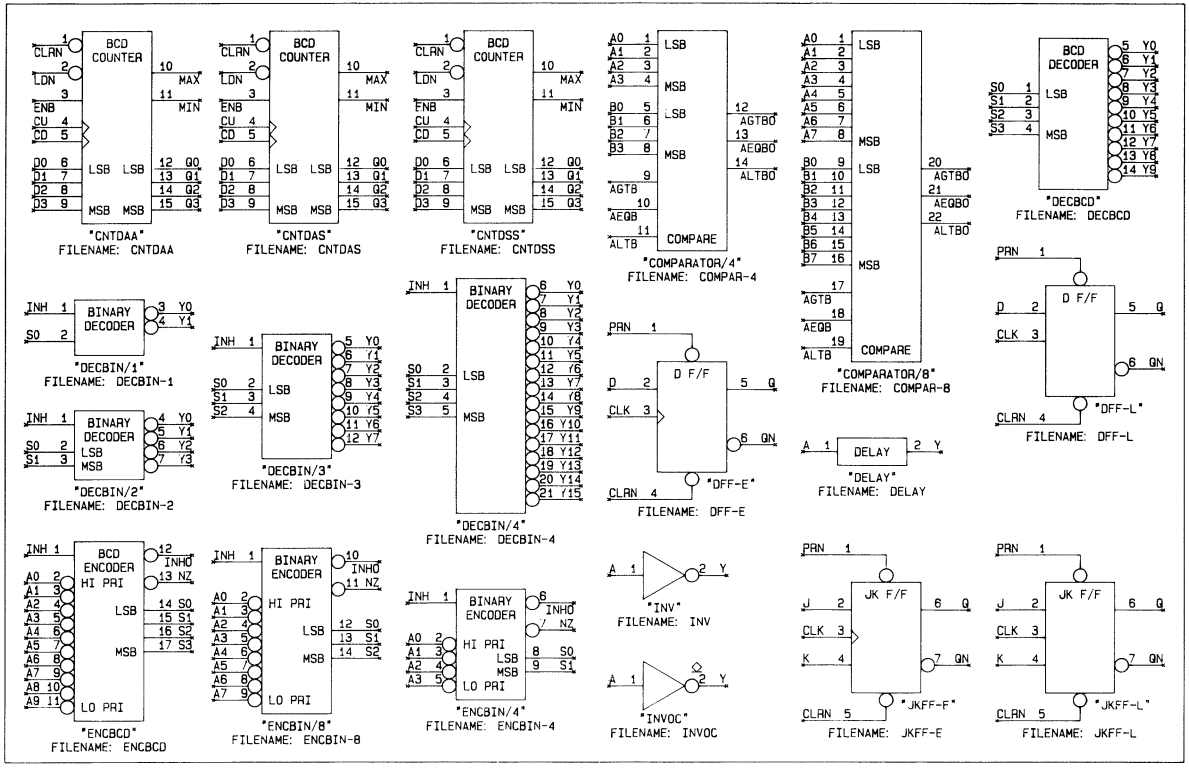
Plot 3


"ANDINVOC/2"
FILENAME: ANDIOC-2
"ANDINVOC/3"
FILENAME: ANDIOC-3
"ANDINVOC/4"
FILENAME: ANDIOC-4
"ANDINVOC/5"
FILENAME: ANDIOC-5
"ANDINVOC/8"
FILENAME: ANDIOC-8
"ANDNEG6/1,1"
FILENAME: ANDNG-1
"ANDNEG6/3,2"
FILENAME: ANDNG-2
"ANDNEGOC/1,1"
FILENAME: ANDNGO-1
"ANDNEGOC/3,2"
FILENAME: ANDNGO-2
"ANDNOR/4,2,3,2,2,1"
FILENAME: ANDO-1
"ANDNOR/4,2,2,2,2,0"
FILENAME: ANDN-1
"ANDNOR/2,2,2,1"
FILENAME: ANDN-2
"ANDNOR/2,3,3,0"
FILENAME: ANDN-3
"ANDNOR/4,2,3,3,2,0"
FILENAME: ANDN-4
"ANDNOROC/4,2,2,2,2,0"
FILENAME: ANDNOC-1

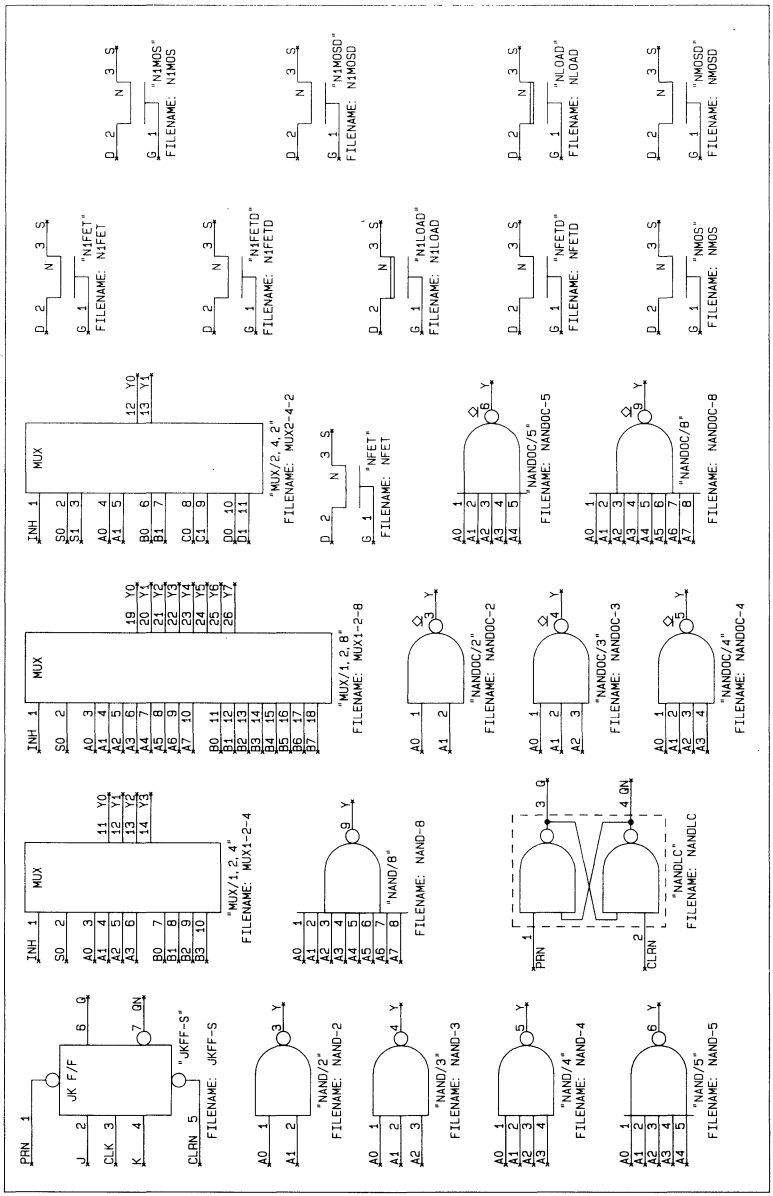
Plot 4


BINARY COUNTER
"CNTBAA/4"
FILENAME: CNTBAA-4
"CNTBAA/8"
FILENAME: CNTBAA-8
"CNTBAS/6"
FILENAME: CNTBAS-6
"CNTBAS/8"
FILENAME: CNTBAS-8
"CNTBSS/4"
FILENAME: CNTBSS-4
"CNTBSS/6"
FILENAME: CNTBSS-6
"CNTBSS/8"
FILENAME: CNTBSS-8
"ANDOOC-1"
FILENAME: ANDOOC
"BUF"
FILENAME: BUF
"BUFOC"
FILENAME: BUFOC
"CMOS"
FILENAME: CMOS
"CMOSD"
FILENAME: CMOSD

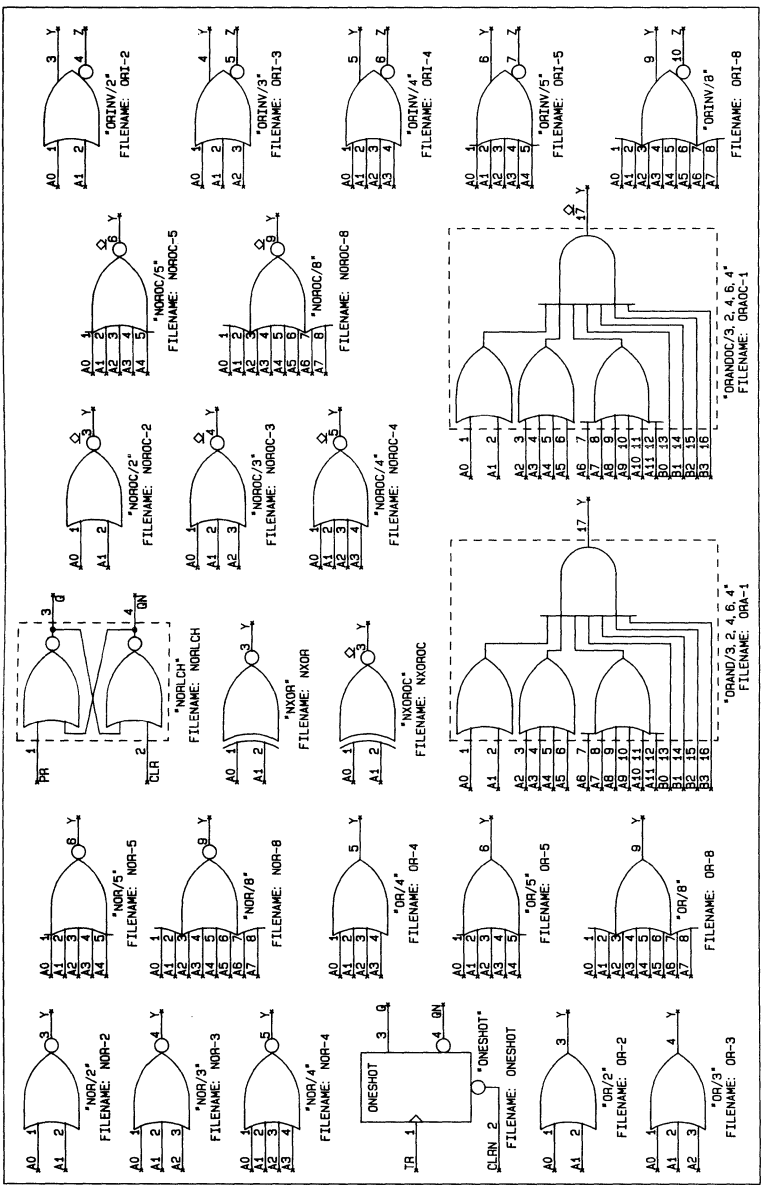
Plot 5


BCD COUNTER
CLRN
LDN
ENB
CU
CD
D0 D1 D2 D3
LSB LSB
MSB MSB
MAX
MIN
Q0 Q1 Q2 Q3
"CNTDAA"
FILENAME: CNTDAA
"CNTDAS"
FILENAME: CNTDAS
"CNTDSS"
FILENAME: CNTDSS
A0 A1 A2 A3
B0 B1 B2 B3
LSB
MSB
AGTB
AEQB
ALTB
COMPARE
"COMPARATOR/4"
FILENAME: COMPAR-4
"COMPARATOR/8"
FILENAME: COMPAR-8
BCD DECODER
S0 S1 S2 S3
Y0 Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8 Y9
"DECBCD"
FILENAME: DECBCD
BINARY DECODER
INH
"DECBIN/1"
FILENAME: DECBIN-1
"DECBIN/2"
FILENAME: DECBIN-2
"DECBIN/3"
FILENAME: DECBIN-3
"DECBIN/4"
FILENAME: DECBIN-4
D F/F
PRN
D
CLK
Q QN
CLRN
"DFF-E"
FILENAME: DFF-E
"DFF-L"
FILENAME: DFF-L
DELAY
A Y
"DELAY"
FILENAME: DELAY
BCD ENCODER
HI PRI
LO PRI
INHO NZ
S0 S1 S2 S3
"ENCBCD"
FILENAME: ENCBCD
BINARY ENCODER
"ENCBIN/8"
FILENAME: ENCBIN-8
"ENCBIN/4"
FILENAME: ENCBIN-4
"INV"
FILENAME: INV
"INVOC"
FILENAME: INVOC
JK F/F
J K
"JKFF-E"
"JKFF-L"
FILENAME: JKFF-E
FILENAME: JKFF-L

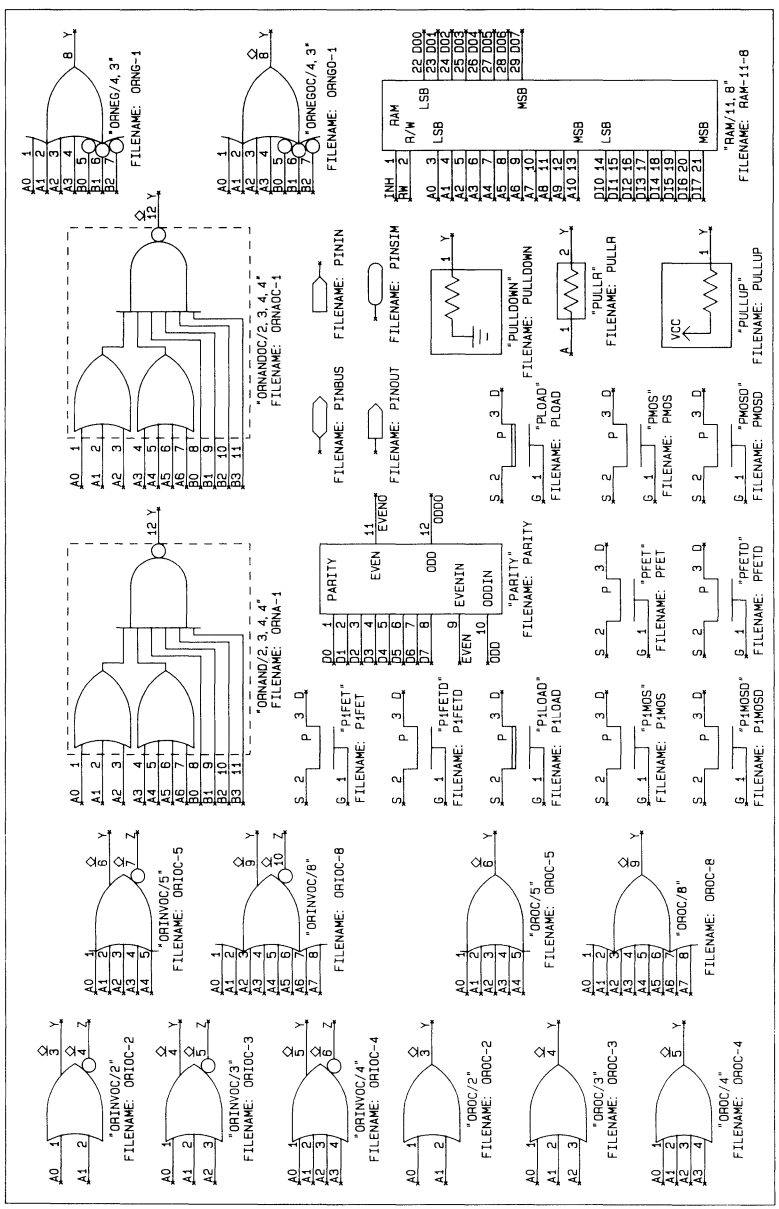
Plot 6


PRN 1
J 2
CLK 3
K 4
CLRN 5
JK F/F
6 Q
7 QN
"JKFF-S"
FILENAME: JKFF-S
INH 1
S0 2
A0 3
A1 4
A2 5
A3 6
B0 7
B1 8
B2 9
B3 10
MUX
11 Y0
12 Y1
13 Y2
14 Y3
"MUX/1, 2, 4"
FILENAME: MUX1-2-4
A0 1
A1 2
A2 3
A3 4
A4 5
A5 6
A6 7
A7 8
B0 11
B1 12
B2 13
B3 14
B4 15
B5 16
B6 17
B7 18
19 Y0
20 Y1
21 Y2
22 Y3
23 Y4
24 Y5
25 Y6
26 Y7
"MUX/1, 2, 8"
FILENAME: MUX1-2-8
A0 1
A1 2
A2 3
A3 4
A4 5
A5 6
A6 7
A7 8
9 Y
"NAND/8"
FILENAME: NAND-8
A0 1
A1 2
3 Y
"NAND/2"
FILENAME: NAND-2
A0 1
A1 2
A2 3
4 Y
"NAND/3"
FILENAME: NAND-3
A0 1
A1 2
A2 3
A3 4
5 Y
"NAND/4"
FILENAME: NAND-4
A0 1
A1 2
A2 3
A3 4
A4 5
6 Y
"NAND/5"
FILENAME: NAND-5
PRN 1
3 Q
4 QN
CLRN 2
"NANDLC"
FILENAME: NANDLC
A0 1
A1 2
A2 3
A3 4
5 Y
"NANDOC/4"
FILENAME: NANDOC-4
A0 1
A1 2
A2 3
A3 4
6 Y
"NANDOC/5"
FILENAME: NANDOC-5
A0 1
A1 2
A2 3
A3 4
A4 5
A5 6
A6 7
A7 8
9 Y
"NANDOC/8"
FILENAME: NANDOC-8
INH 1
S0 2
S1 3
A0 4
A1 5
B0 6
B1 7
C0 8
C1 9
D0 10
D1 11
MUX
12 Y0
13 Y1
"MUX/2, 4, 2"
FILENAME: MUX2-4-2
D 2
N
3 S
G 1
"NFET"
FILENAME: NFET
D 2
N
3 S
G 1
"N1FET"
FILENAME: N1FET
D 2
N
3 S
G 1
"N1MOS"
FILENAME: N1MOS
D 2
N
3 S
G 1
"N1FETD"
FILENAME: N1FETD
D 2
N
3 S
G 1
"N1MOSD"
FILENAME: N1MOSD
D 2
N
3 S
G 1
"N1LOAD"
FILENAME: N1LOAD
D 2
N
3 S
G 1
"NFETD"
FILENAME: NFETD
D 2
N
3 S
G 1
"NLOAD"
FILENAME: NLOAD
D 2
N
3 S
G 1
"NMOS"
FILENAME: NMOS
D 2
N
3 S
G 1
"NMOSD"
FILENAME: NMOSD

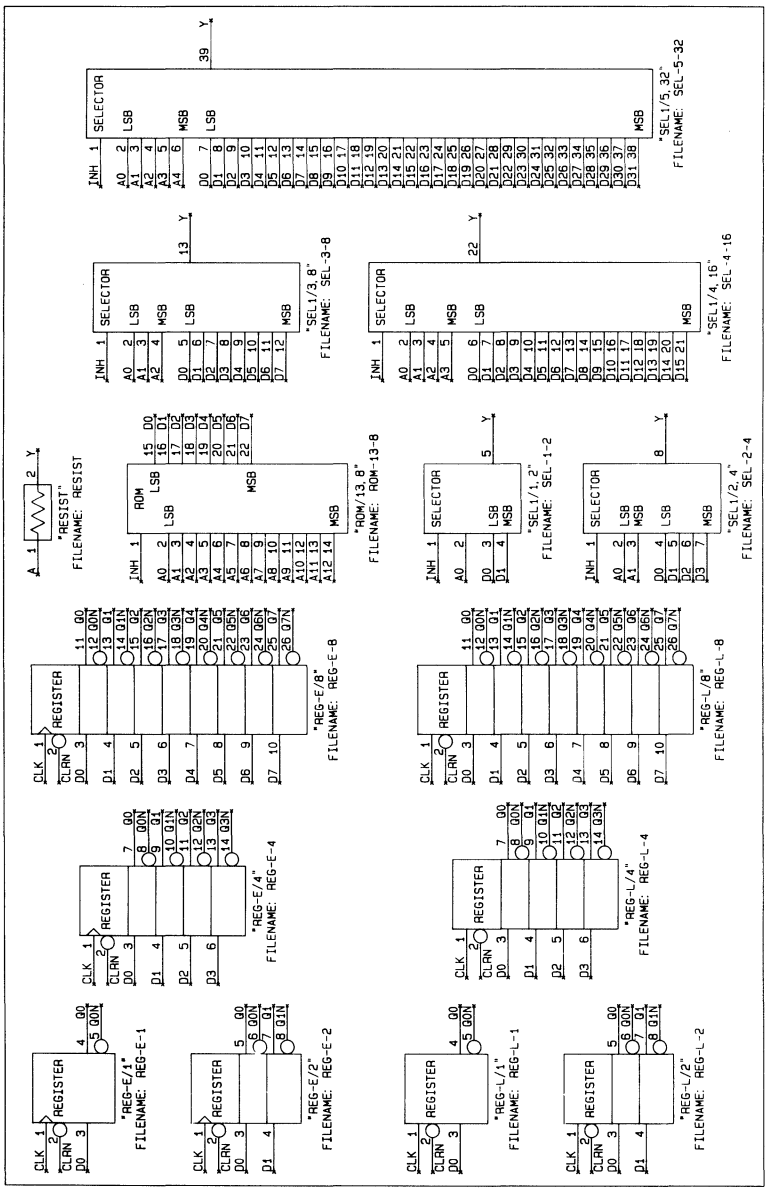
Plot 7

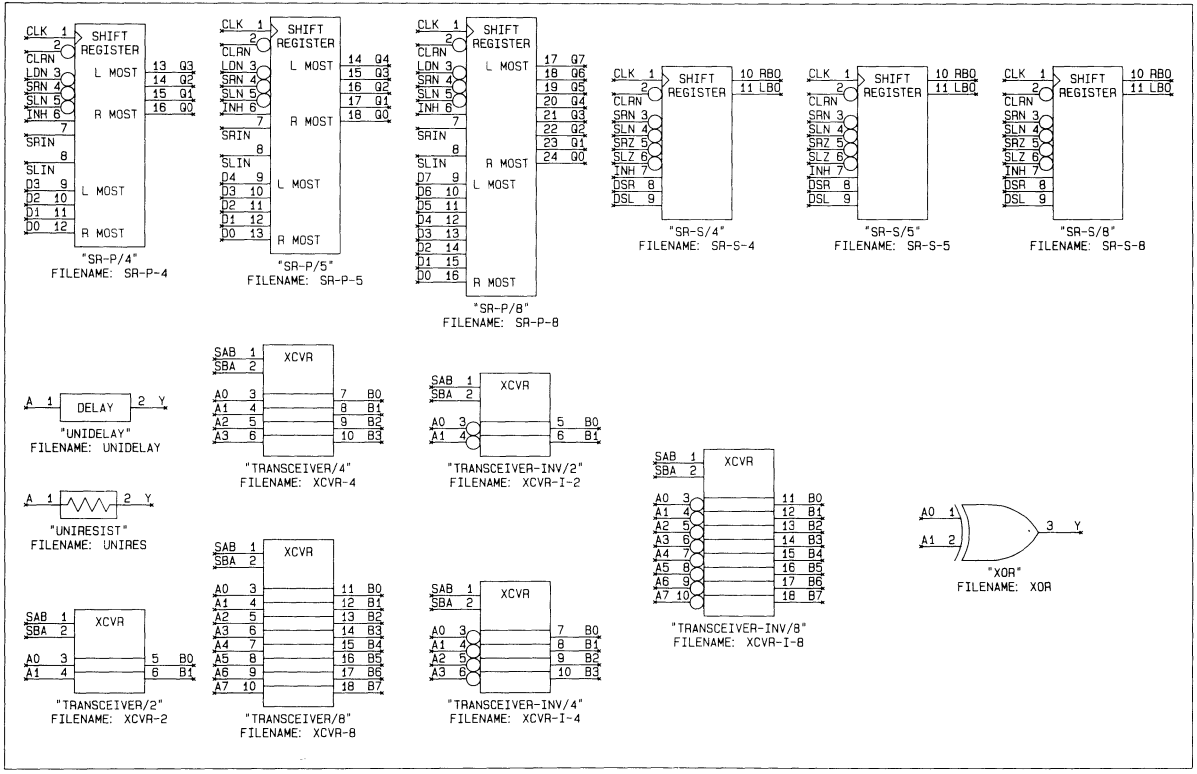

"NOR/2"
FILENAME: NOR-2
"NOR/3"
FILENAME: NOR-3
"NOR/4"
FILENAME: NOR-4
ONESHOT
"ONESHOT"
FILENAME: ONESHOT
"OR/2"
FILENAME: OR-2
"OR/3"
FILENAME: OR-3
"NOR/5"
FILENAME: NOR-5
"NOR/8"
FILENAME: NOR-8
"OR/4"
FILENAME: OR-4
"OR/5"
FILENAME: OR-5
"OR/8"
FILENAME: OR-8
PR
CLR
"NORLCH"
FILENAME: NORLCH
"NXOR"
FILENAME: NXOR
"NXOROC"
FILENAME: NXOROC
"NOROC/2"
FILENAME: NOROC-2
"NOROC/3"
FILENAME: NOROC-3
"NOROC/4"
FILENAME: NOROC-4
"NOROC/5"
FILENAME: NOROC-5
"NOROC/8"
FILENAME: NOROC-8
"DRAND/3, 2, 4, 6, 4"
FILENAME: DRA-1
"DRANDOC/3, 2, 4, 6, 4"
FILENAME: DRAOC-1
"DRINV/2"
FILENAME: DRI-2
"DRINV/3"
FILENAME: DRI-3
"DRINV/4"
FILENAME: DRI-4
"DRINV/5"
FILENAME: DRI-5
"DRINV/8"
FILENAME: DRI-8

Plot 8


"ORINVOC/2"
FILENAME: ORIOC-2
"ORINVOC/5"
FILENAME: ORIOC-5
"ORINVOC/3"
FILENAME: ORIOC-3
"ORINVOC/8"
FILENAME: ORIOC-8
"ORINVOC/4"
FILENAME: ORIOC-4
"ORNAND/2, 3, 4, 4"
FILENAME: ORNA-1
"ORNANDOC/2, 3, 4, 4"
FILENAME: ORNAOC-1
"ORNEG/4, 3"
FILENAME: ORNG-1
"ORNEGOC/4, 3"
FILENAME: ORNGO-1
"OROC/2"
FILENAME: OROC-2
"OROC/5"
FILENAME: OROC-5
"OROC/3"
FILENAME: OROC-3
"OROC/8"
FILENAME: OROC-8
"OROC/4"
FILENAME: OROC-4
"P1FET"
FILENAME: P1FET
"P1FETD"
FILENAME: P1FETD
"P1LOAD"
FILENAME: P1LOAD
"P1MOS"
FILENAME: P1MOS
"P1MOSD"
FILENAME: P1MOSD
"PARITY"
FILENAME: PARITY
"PFET"
FILENAME: PFET
"PFETD"
FILENAME: PFETD
"PLOAD"
FILENAME: PLOAD
"PMOS"
FILENAME: PMOS
"PMOSD"
FILENAME: PMOSD
FILENAME: PINBUS
FILENAME: PINOUT
FILENAME: PININ
FILENAME: PINSIM
"PULLDOWN"
FILENAME: PULLDOWN
"PULLR"
FILENAME: PULLR
"PULLUP"
FILENAME: PULLUP
"RAM/11,8"
FILENAME: RAM-11-8

Plot 9


"REG-E/1"
FILENAME: REG-E-1
"REG-E/2"
FILENAME: REG-E-2
"REG-E/4"
FILENAME: REG-E-4
"REG-E/8"
FILENAME: REG-E-8
"REG-L/1"
FILENAME: REG-L-1
"REG-L/2"
FILENAME: REG-L-2
"REG-L/4"
FILENAME: REG-L-4
"REG-L/8"
FILENAME: REG-L-8
"RESIST"
FILENAME: RESIST
"ROM/13,8"
FILENAME: ROM-13-8
"SEL1/1,2"
FILENAME: SEL-1-2
"SEL1/2,4"
FILENAME: SEL-2-4
"SEL1/3,8"
FILENAME: SEL-3-8
"SEL1/4,16"
FILENAME: SEL-4-16
"SEL1/5,32"
FILENAME: SEL-5-32


SHIFT REGISTER
L MOST
R MOST
"SR-P/4"
FILENAME: SR-P-4
"SR-P/5"
FILENAME: SR-P-5
"SR-P/8"
FILENAME: SR-P-8
"SR-S/4"
FILENAME: SR-S-4
"SR-S/5"
FILENAME: SR-S-5
"SR-S/8"
FILENAME: SR-S-8
DELAY
"UNIDELAY"
FILENAME: UNIDELAY
"UNIRESIST"
FILENAME: UNIRES
XCVR
"TRANSCEIVER/2"
FILENAME: XCVR-2
"TRANSCEIVER/4"
FILENAME: XCVR-4
"TRANSCEIVER/8"
FILENAME: XCVR-8
"TRANSCEIVER-INV/2"
FILENAME: XCVR-I-2
"TRANSCEIVER-INV/4"
FILENAME: XCVR-I-4
"TRANSCEIVER-INV/8"
FILENAME: XCVR-I-8
"XOR"
FILENAME: XOR

APPENDIX C. ATTRIBUTE SUMMARY

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

keyword[a][n]="value"

where:

keyword is the name of the attribute.

a is the name of the component alias to which the keyword is assigned.

n is the sequence number.

value is the value you specify for it. For NX-CADAT, you must always enclose the value in quotation marks.

An example of an attribute is shown below.

CDTECH="CMOS"

Table C-1 lists, in alphabetical order, all the attributes used with NX-CADAT, the PC-CAPS mode used to assign them, the types of description language files in which they are used (indicated with the asterisk (*) in the column headed NDL for Network Description Language and the column headed MDL for Model Description Language), and brief descriptions.

Table C-1. Attribute Summary

Attribute	PC-CAPS	NDL	MDL	Description
CDALIAa	SYMBOL		*	Alias name
CDCOM	SYMBOL	*	*	General comments
CDDD	SYMBOL	*	*	Default delay for circuit
CDDT	SYMBOL	*	*	Default tolerance for circuit
CDFMLY	SYMBOL		*	Family type of model
CDFMLYa	SYMBOL		*	Family type of alias
CDINFOF	SYMBOL	*	*	Text file to replace other attributes
CDLDD	SYMBOL		*	Load-dependent delay (for symbol pins of model)
CDLOAD	DETAIL	*	*	Loading information
CDLOCAL	DETAIL	*	*	Local model filename
CDLU	SYMBOL	*	*	Date and time of last update
CDNAME	SYMBOL	*	*	Circuit name
CDNOFT	DETAIL	*	*	Nofault list
CDNOFT	SYMBOL	*		Nofault list (for external name only)
CDPART	DETAIL	*	*	Part function and/or location
CDPIN	DETAIL	*		Pin number and/or driver/sensor
CDPIN	SYMBOL	*		Pin type and/or driver/sensor number (for symbol pins only)
CDREV	SYMBOL	*	*	Revision (comment)
CDTECH	SYMBOL	*	*	Technology of circuit
CDTECH	DETAIL	*	*	Technology of component
CDTHL	SYMBOL		*	Threshold of model
CDTHLa	SYMBOL		*	Threshold of alias
CDTIM	DETAIL	*	*	Timing information for a component or model
CDTIMa	DETAIL	*	*	Timing information for an alias
CDTYPE	SYMBOL	*	*	Part type
CDVECT	SYMBOL	*	*	Vector

APPENDIX D. SAMPLE FILES

This appendix explains the sample databases supplied with NX-CADAT and shows several of the sample files.

You can use the sample files in several ways to become familiar with NX-CADAT.

- You can look at the schematics shown in this appendix and use PC-CAPS to examine the schematic database files to see how specific aspects of schematic preparation affect the NX-CADAT output.
- You can use the sample files as test input to experiment with running NX-CADAT.
- You can examine the CADAT format output files and the back-annotation command files as an aid to understanding the structure of these files.

The sample files include two designs. The DEMO sample is a packaged design, and the DF115 is an unpackaged design to be used as a model.

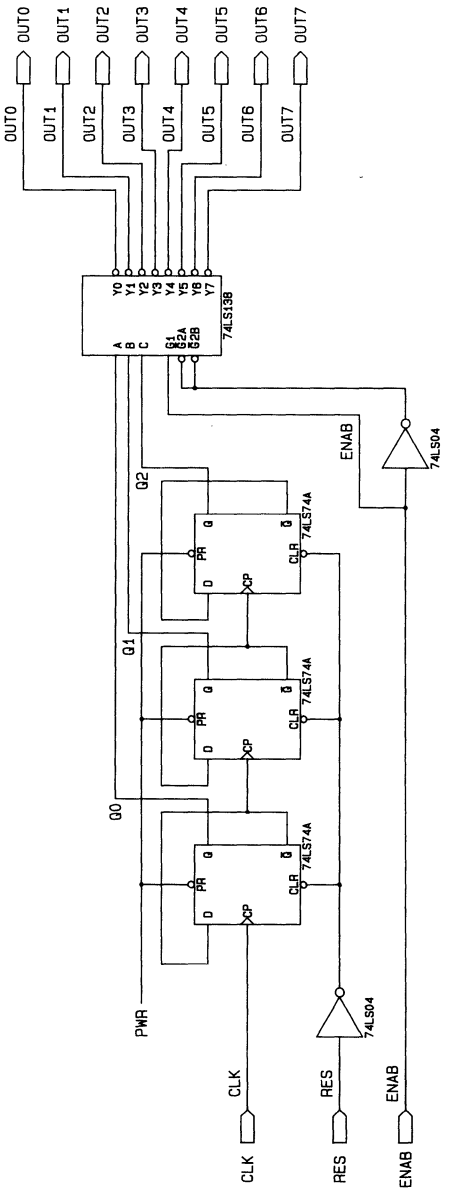
DEMO SAMPLE

The DEMO sample is a 3-bit ripple counter feeding a 3-to-8 decoder. It includes the following sample files.

DEMOI.SCH	Initial schematic database file
DEMOI.NLT	Netlist file
NXCADAT.CFG	Configuration file
DEMOI.CKT	Network description file
NXCADAT.DSL	Design stimulus language file
DEMOI.CMD	Back-annotation command file
DEMOF.SCH	Updated schematic database file

The initial schematic is shown below, followed by the two NX-CADAT output files, followed by the final schematic updated with packaging information.

Schematic Database: DEMOI.SCH



000-0124-00

Network Description File: DEMOI.CKT

```
HEADER $
     CIRCUIT NAME $
          DEMOI.SCH $
     LAST UPDATE $
          JUN 27 1986 10:18 AM $
PARTS $
     UP1 74LS74A $
     UP2 74LS74A $
     UP3 74LS138 $
     UP4 74LS04 $
CONNECTIONS $
     UP1.6 UP1.2 UP1.11 $
     UP1.3 CLK $
     UP1.5 UP3.1 Q0 $
     UP1.8 UP1.12 UP2.3 $
     UP1.9 UP3.2 Q1 $
     UP2.6 UP2.2 $
     UP1.1 UP1.13 UP2.1 UP4.2 $
     UP1.4 UP1.10 UP2.4 PWR $
     UP2.5 UP3.3 Q2 $
     UP3.4 UP3.5 UP4.4 $
     UP3.6 UP4.3 ENAB $
     UP4.1 RES $
     UP3.12 OUT3 $
     UP3.11 OUT4 $
     UP3.13 OUT2 $
     UP3.14 OUT1 $
     UP3.15 OUT0 $
     UP3.10 OUT5 $
     UP3.9 OUT6 $
     UP3.7 OUT7 $
EXTERNALS $
     CLK (IN) CLK $
     RES (IN) RES $
     ENAB (IN) ENAB $
     OUT0 (OUT) OUT0 $
     OUT1 (OUT) OUT1 $
     OUT2 (OUT) OUT2 $
     OUT3 (OUT) OUT3 $
     OUT4 (OUT) OUT4 $
     OUT5 (OUT) OUT5 $
     OUT6 (OUT) OUT6 $
     OUT7 (OUT) OUT7 $
END $

          NETWORK DESCRIPTION FILE:  DEMOI.CKT
```

DSL File: NXCADAT.DSL

```
CIRCUIT DEMOI;

TIMEDEF PERIOD = ? SENSE DELAY = ? ;
OUTPUT  OUTO OUT1 OUT2 OUT3 OUT4 OUT5
OUT6 OUT7;
INPUT   CLK RES ENAB;

END.
```

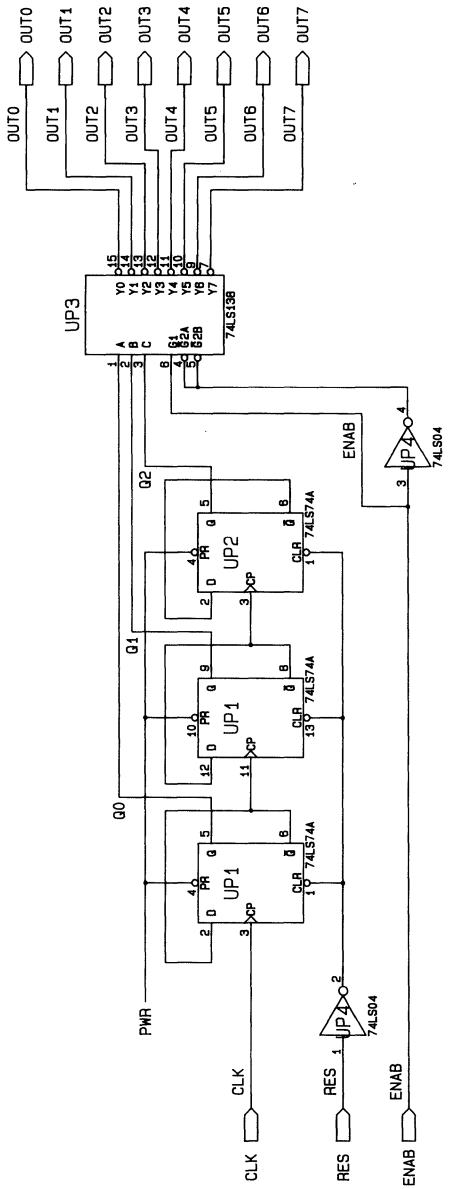
DSL FILE: NXCADAT.DSL

Back-Annotation Command File: DEMOI.CMD

```
LOGFILE
Command  FILE
Command  LOAD
Text     DEMOI.SCH
Command  DETL
Command  SCMD
Command  PNUM
B1       630 820
Text     UP1( 5 6 3 2 4 1 )
B1       880 820
Text     UP1( 9 8 11 12 10 13 )
B1       1130 820
Text     UP2( 5 6 3 2 4 1 )
B1       1490 970
Text     UP3( 15 14 13 12 11 10 9 7 1 2 3 6 >
Text          4 5 )
B1       450 755
Text     UP4( 2 1 )
B1       1310 655
Text     UP4( 4 3 )
Command  FILE
Command  SAVE
Text     DEMOI.SCH
```

BACK-ANNOTATION COMMAND FILE: DEMOI.CMD

Updated Schematic Database: DEMOF.SCH



000-0124-00

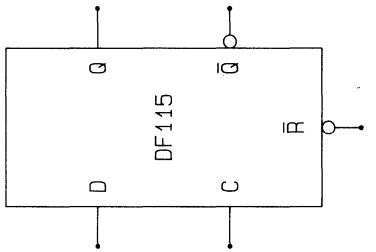
DF115 SAMPLE

The DF115 sample is a symbol built for a D flip-flop. It contains a number of attributes recognized by NX-CADAT. This sample includes the following files.

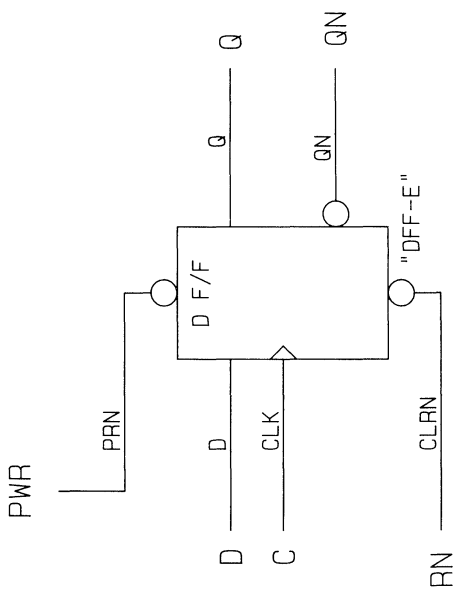
DF115.SYM	Schematic symbol file
DF115.NLT	Netlist file
DF115.MDL	Model description file

The schematic symbol is shown below, followed by the output model description file.

Schematic Symbol: DF115.SYM




```
"3.6ns, 3.6ns"
"Q DRIVE=1.5,  1.5 ns/pF"
"QN DRIVE=1.5,  1.5 ns/pF"
"D LOAD=0.06 pf"
"C LOAD=0.08 pf"
"RN LOAD=0.17 pf"
```




```
"$.QN = 2.7ns, 2.7ns"
```

000-0124-00

Model Description File: DF115.MDL

```
HEADER $
     CIRCUIT NAME $
          DF115.SYM $
     LAST UPDATE $
          JUN 27 1986 01:55 PM $
     DEFAULT DELAY $
          3.6ns,3.6ns $
PARTS $
     UC000000 DFF-E $
CONNECTIONS $
     UC000000.5 Q $
     UC000000.6 QN $
     UC000000.4 RN $
     UC000000.3 C $
     UC000000.2 D $
     UC000000.1 PWR $
MODEL INFO $
     PINS $
          1:Q Q  DRIVE=1.5, 1.5 ns/pF $
          2:QN QN  DRIVE=1.5, 1.5 ns/pF $
          3:D D  LOAD=0.06 pf $
          4:C C  LOAD=0.08 pf $
          5:RN RN  LOAD=0.17 pf $
     DEFAULTS $
          TIMING $
               UC000000.QN = 2.7ns,2.7ns $
     TYPES $
          NAME = DF115 $
END $
```

MODEL DESCRIPTION FILE: DF115.MDL