CR-5000

DCS Translator

Contents

Chapter 1 OVERVIEW

This chapter provides an overview of data conversion from CR-3000/DCS to CR-5000/System Designer by DCS translation software and its program configuration.

1.1 OVERVIEW

This product is the software for converting the schematic data and part property data designed with CR-3000/DCS to the schematic data and the components database for schematic design (LCDB) for CR-5000/System Designer without the need for making complicated settings.

This product consists of two programs, DCS Translator and DCS Translator Sub-kit, described in the following two sections.

1.1.1 DCS Translator Sub-kit

DCS Translator Sub-kit is invoked from within DCS .

DCS Translator Sub-kit is the program that converts those schematics and symbol libraries designed with DCS to data that System Designer can capture. Figures information in the schematics and symbol libraries is converted to the transfer file dedicated to System Designer , called a SD transfer file . Part property data is converted directly to the components database for schematic design (LCDB).
The program also supports DeMorgan symbols and multi-component symbols.

This program can generate the SD transfer file and the components database for schematic design (LCDB) either both at the same time by a single operation in DCS or separately.

1.1.2 DCS Translator

DCS Translator is invoked from within System Designer.

DCS Translator is the program that converts the SD transfer file generated by DCS Translator Sub-kit to System Designer schematic data.

1.2 Purposes

This product is designed for purposes of:

(1) Generating a SD transfer file from schematic data designed with DCS .

(2) Converting DCS part property data to the components database for schematic design (LCDB) for System Designer .

(3) Checking DCS data to be converted to notify the user of restrictions before conversion.

(4) Generating resource files required for System Designer .

(5) Converting the SD transfer file generated from DCS schematic data to System Designer schematic data.

(6) And finally, enabling a smooth migration from the DCS design environment to System Designer .

The purposes (1)-(4) and (5) can be achieved by functions of DCS Translator Sub-kit and DCS Translator , respectively.

WARNING: This product does not support data conversion from System Designer to DCS .

1.3 Product Configuration

The configuration of this product is illustrated below.

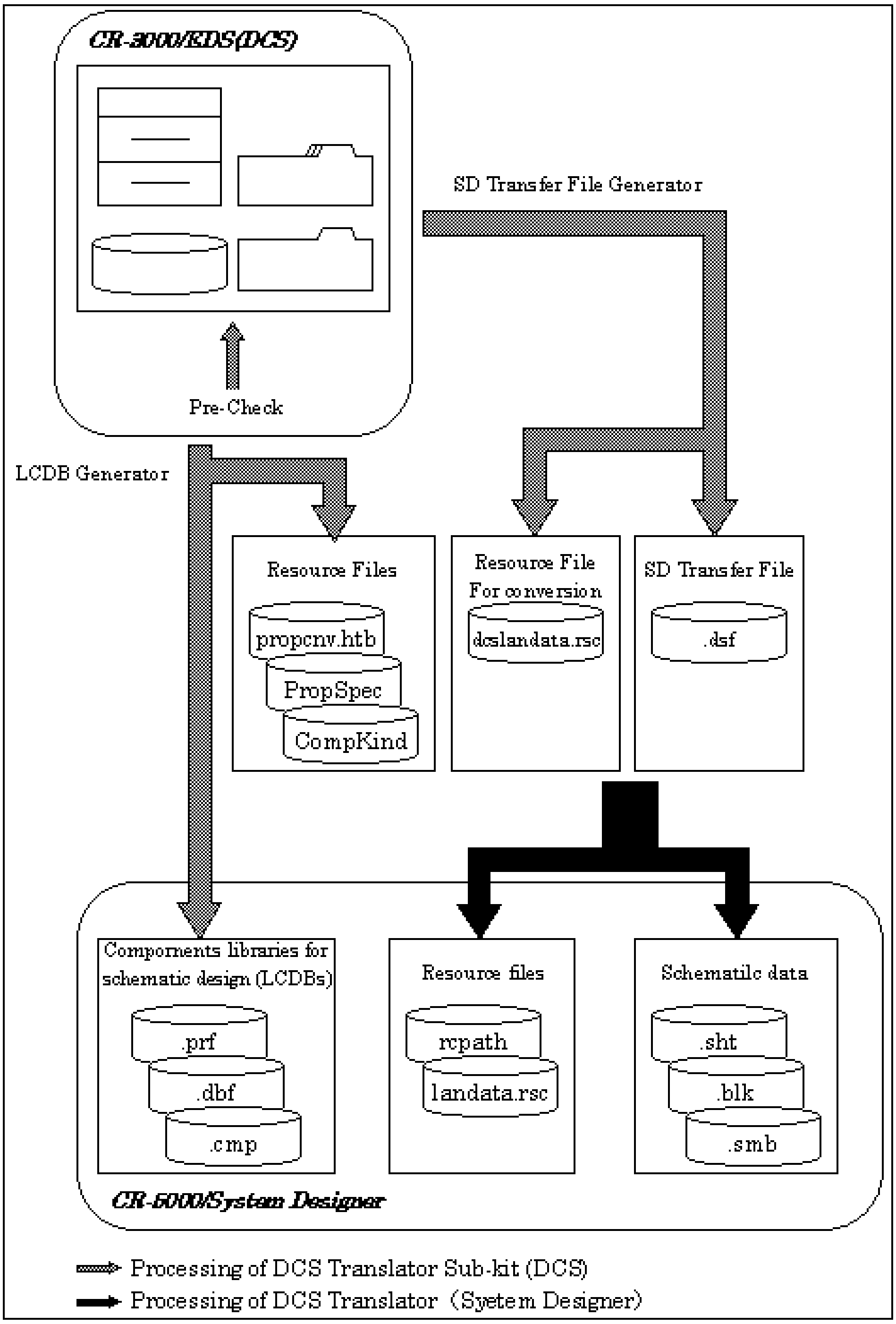


Figure 1.1 Product Configuration

1.4 Functional Configuration

DCS Translator Sub-kit consists of the following functions:

(1) Pre-Checker

(2) SD Transfer File Generator

(3) LCDB Generator

DCS Translator consists of the following function:

(1) DCS Translator program

1.4.1 Configuration of Pre-Checker

Pre-Checker checks DCS schematic data and part property data for their items to be restricted during conversion to data for System Designer.
Pre-Checker is configured as shown below.

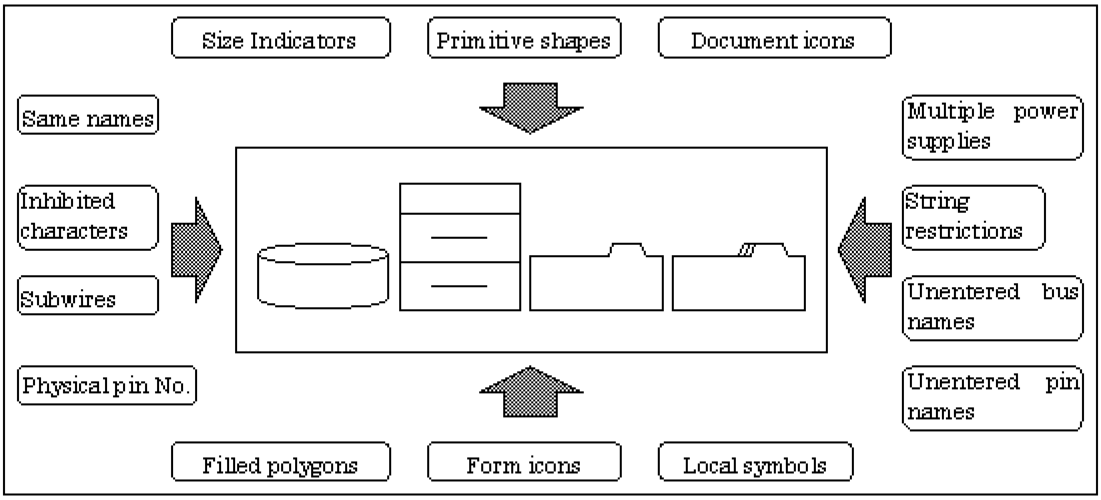


Figure 1.2 Configuration of Pre-Checker

1.4.2 Configuration of SD Transfer File Generator

SD Transfer File Generator generates a SD transfer file (.dsf) and resource file (dcslandata.rsc) from DCS schematic data and symbol library figures information.
The DCS schematic data can be used by System Designer by converting the SD transfer file in System Designer .

This function converts figures information and generates the following types of data:

- Symbol shapes, pin shapes (positive/negative logic symbols)
- Multi-component symbols
- Power box symbols based on implied pin information
- Symbols of forms inserted on the circuit page

SD Transfer File Generator is configured as shown below.

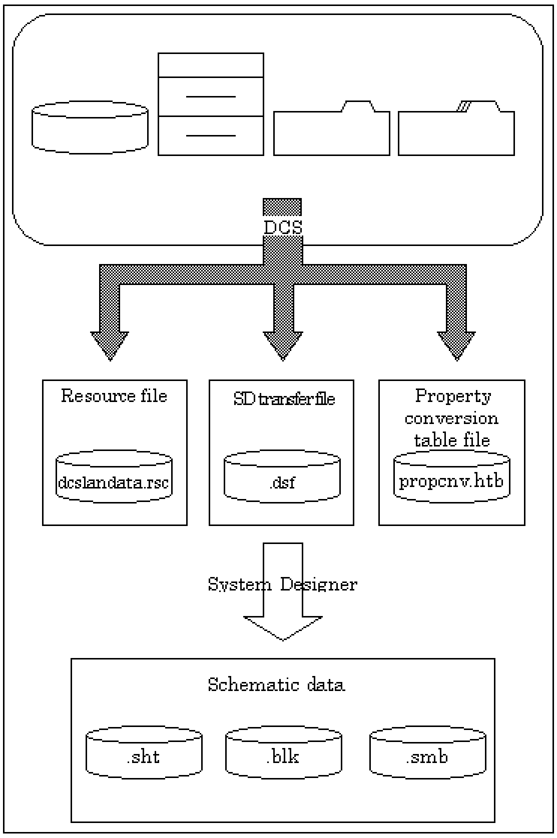


Figure 1.3 Configuration of SD Transfer File Generator

1.4.3 Configuration of LCDB Generator

LCDB Generator generates components database for schematic design (LCDBs) and resource files from DCS part property data .

This function generates the following files:

- Parameter file (.prf)
- Database file (.dbf)
- Component file (.cmp)
- Function type definition file (CompKind)
- Property conversion table file (propcnv.htb)
- Property specification file (PropSpec)

LCDB Generator is configured as shown below.

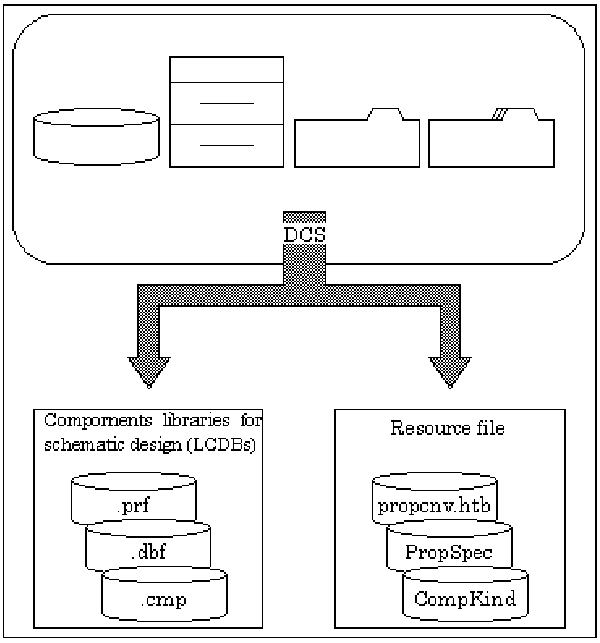


Figure 1.4 Configuration of LCDB Generator

1.4.4 Configuration of DCS Translator program

The DCS Translator program generates schematic data for System Designer from the SD transfer file generated by DCS Translator Sub-kit.
This function generates the following data and resource files:

- Schematic data files (.sht, .blk, .smb)
- Resource path files (rcpath)
- Data resource file (landata.rsc)

The DCS Translator program is configured as shown below.

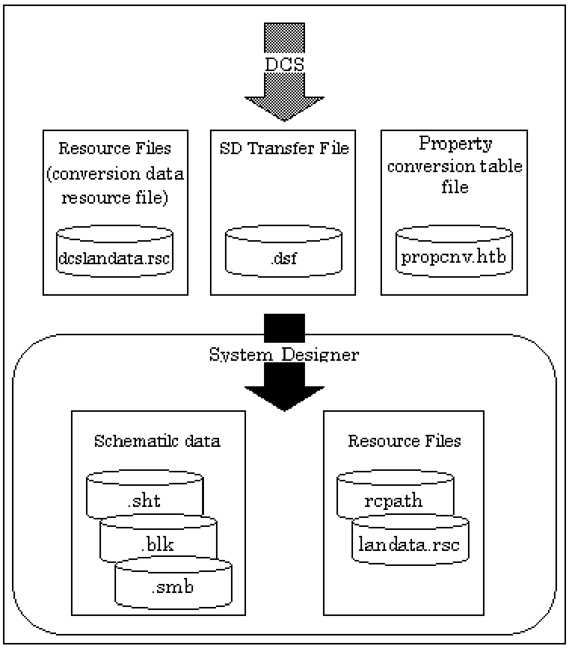


Figure 1.5 Configuration of DCS Translator program

Chapter 2 OPERATING PROCEDURES

This chapter explains how to use the programs included in DCS Translator Sub-kit and DCS Translator . Before starting operation for conversion, be sure to read Chapter 4, "DATA CONVERSION SPECIFICATIONS."

2.1 DCS Translator Sub-kit

DCS Translator Sub-kit provides the following programs for converting the schematic data and part property data designed with DCS :

(1) Pre-Checker

(2) SD Transfer File Generator

(3) LCDB Generator

(4) Data Converter

2.1.1 Before Starting Data Conversion

DCS and System Designer are systems which are different both in basic design and in restriction. In System Designer , therefore, some types of data may result in an error or be handled as invalid data. Such data must be corrected in DCS before conversion.

The Pre-Checker program is thus provided to check data before conversion for this purpose. Before converting DCS schematic data or part property data , be sure to run Pre-Checker .

If the Pre-Checker program has returned any error or warning , be sure to follow the instruction for the corresponding message in Chapter 6 "MESSAGES" to correct the relevant item of data before retrying conversion.

Note that, if you convert that SD transfer file in System Designer which has been generated from but not yet corrected in DCS , data conversion may be unsuccessful. If you do not perform correction in DCS, in addition, the resulting components database for schematic design (LCDBs) may not be used normally.

2.1.2 Pre-Checker

Pre-Checker is the program for checking DCS schematic data and part property data for their items to be restricted during conversion to data for System Designer.

Starting Pre-Checker

To execute the program, select the following menu command string on the screen,collector, or circuit page :

```
/PERFORM/INTERFACE/System Designer/DATA CONVERSION/Pre-Check
```

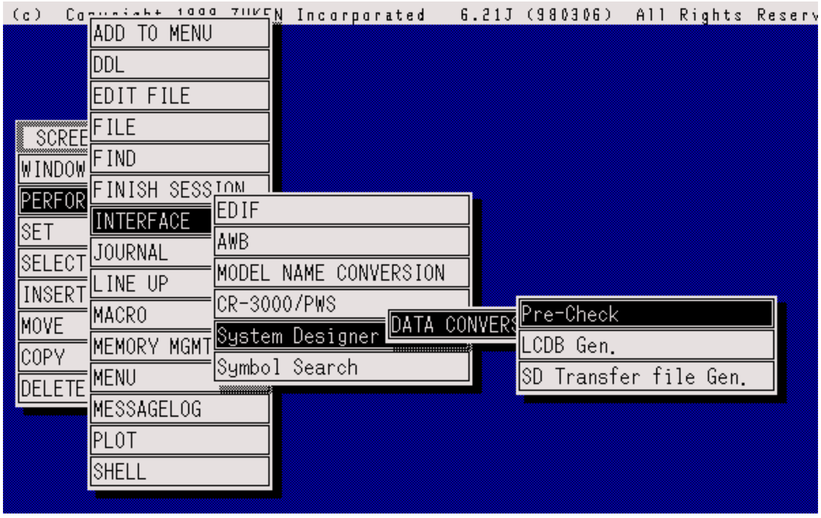
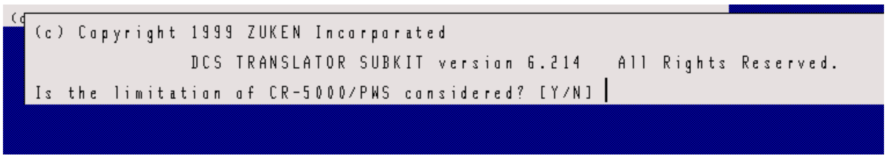


Figure 2.1 Menu Selection for Pre-Checker

Selecting the above command string causes the program to execute the precheck operation illustrated in Figure 1.2 "Configuration of Pre-Checker " in Section 1.4.1 "Configuration of Pre-Checker ." For details on the *contents of the precheck, see Section 4.1 "Pre-Checker."

(1) **Selecting a mode**

When using the CR-5000 PWS from System Designer, select if conversion should be executed with respect to the restrictions on CR-5000 PWS such as small letters and prohibited characters).



```
(c) Copyright 1999 ZUKEN Incorporated
                DCS TRANSLATOR SUBKIT version 6.214    All Rights Reserved.
Is the limitation of CR-5000/PWS considered? [Y/N] |
```

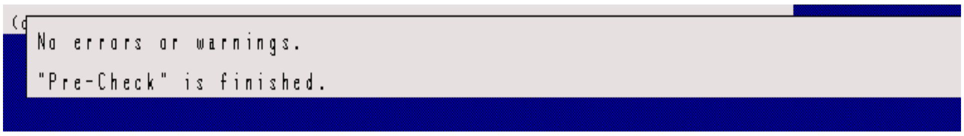
Figure 2.2 Selecting a mode

- If "Y" is selected, the restrictions on CR-5000 PWS will be taken into consideration during conversion.
  Items to be considered are Design Icon Name, Physical Reference, Pin Name, Pin Number (Alphabetic Pin Number), and Net Name.
  - Target texts are all converted to capital letters.
  - Texts are converted so that each text has 20 characters at most.
  - If any of the following characters
    " ! $ ’ ( ) * , ; < > \ ‘ { | } - # = ,is included in a physical reference, pin name, pin number (alphabetic pin number), or net name, the character will be converted.
- If "N" is selected, conversion will be executed in consideration of the restrictions on \lana.

Quitting Pre-Checker

The banner looks different depending on whether any error or warning has occurred when the program terminates.

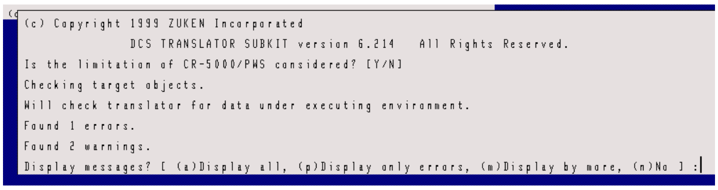
Normal termination : When no error or warning has been detected, the program terminates with the following message:



```
No errors or warnings.
"Pre-Check" is finished.
```

Figure 2.3 Message on Normal Termination

Termination with an error or warning : If an error or warning has been detected, the program terminates with a message(s) as shown below:
]



```
(c) Copyright 1999 ZUKEN Incorporated
              DCS TRANSLATOR SUBKIT version 6.214   All Rights Reserved.
Is the limitation of CR-5000/PWS considered? [Y/N]
Checking target objects.
Will check translator for data under executing environment.
Found 1 errors.
Found 2 warnings.
Display messages? [ (a)Display all, (p)Display only errors, (m)Display by more, (n)No ] :
```

Figure 2.4 Error /Warning Messages

Input character	Response
a	Displays the error and warning messages. (Default)
p	Displays only the error message.
m	Opens the shell window for confirming and editing messages.
n	Displays no message.

If any error or warning has been detected, check the contents of the files listed below and follow the instruction for the corresponding message in Chapter 6 "MESSAGES" to correct the relevant item of data.

Error , warning , and log messages are recorded in the following files:

Output file	File name
Error file	PreCheck.err
Warning file	PreCheck.wrn
Log file	PreCheck.log

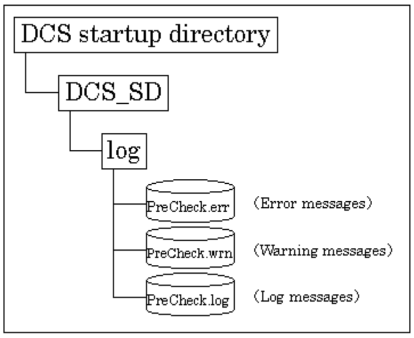


Figure 2.5 Directory Structure

2.1.3 SD Transfer File Generator

SD Transfer File Generator is the program providing the following three functions:

(1) Function of converting schematic data designed with DCS and symbol library figures information to a SD transfer file.

(2) Function of generating a conversion data resource file (dcslandata.rsc) for generating a System Designer data resource file (landata.rsc).

(3) Function of generating a System Designer resource files (propcnv.htb).

Starting SD Transfer File Generator

To execute the program, select the following menu command string on the screen,collector, or circuit page :

```
/PERFORM/INTERFACE/System Designer/DATA CONVERSION/SD Transfer File Gen.
```

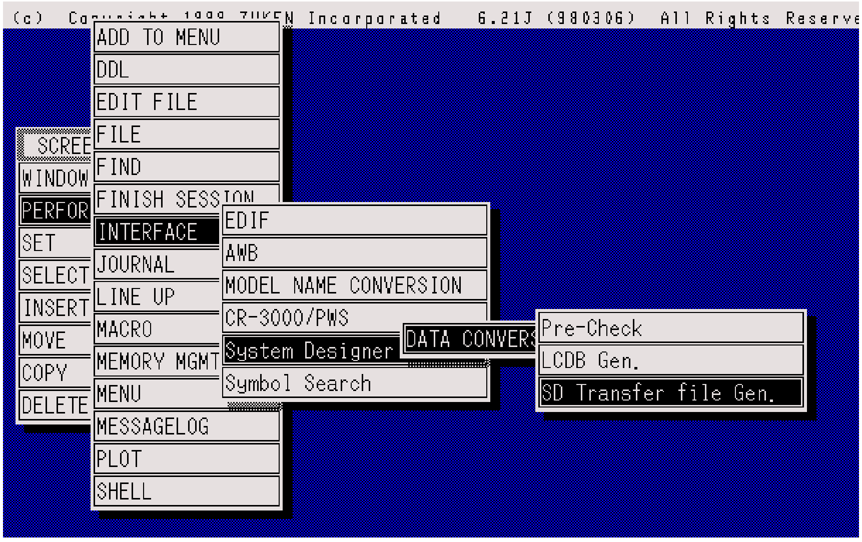
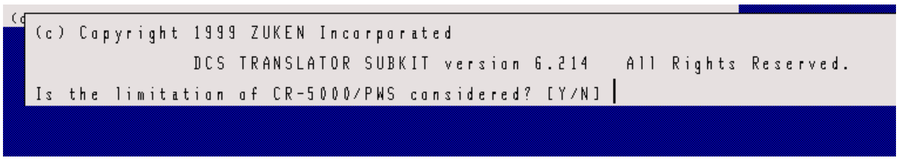


Figure 2.6 Menu Selection for SD Transfer File Generator

Selecting the above command string starts the program. Follow the procedure below according to the prompts and instructions on the banner.

(1) Selecting a mode

When using the CR-5000 PWS from System Designer , select if conversion should be executed with respect to the restrictions on CR-5000 PWS (such as small letters and prohibited characters).



```
(c) Copyright 1999 ZUKEN Incorporated
                DCS TRANSLATOR SUBKIT version 6.214   All Rights Reserved.
Is the limitation of CR-5000/PWS considered? [Y/N]
```

Figure 2.7 Selecting a mode

- If "Y" is selected, the restrictions on CR-5000 PWS will be taken into consideration during conversion.
  Items to be considered are Design Icon Name, Physical Reference, Pin Name, Pin Number (Alphabetic Pin Number), and Net Name.

  -Target texts are all converted to capital letters.

  -Texts are converted so that each text has 20 characters at most.

  -If any of the following characters
  " ! $ ' ( ) * , ; < > \ ` { | } - # = ,is included in a physical reference, pin name, pin number (alphabetic pin number), or net name, the character will be converted.

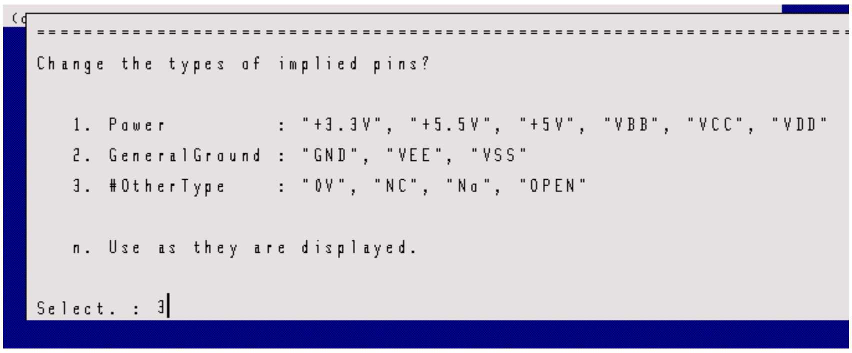
- If "N" is selected, conversion will be executed in consideration of the restrictions on System Designer.

(2) Setting implied pins

The implied pin setting menu appears with implied pins automatically classified into the power, ground, and other types of pins.

Implied Pins must be set correctly because implied pin type information is required for generating power box symbols (detailed in Section 4.2.3 "Power Box Symbols."

The implied pin setting menu is not displayed when the data to be conver does not contain any implied pin .



```
==================================================================
Change the types of implied pins?

   1. Power         : "+3.3V", "+5.5V", "+5V", "VBB", "VCC", "VDD"
   2. GeneralGround : "GND", "VEE", "VSS"
   3. #OtherType    : "0V", "NC", "No", "OPEN"

   n. Use as they are displayed.

Select. : 3
```

Figure 2.8 Implied Pin Settings

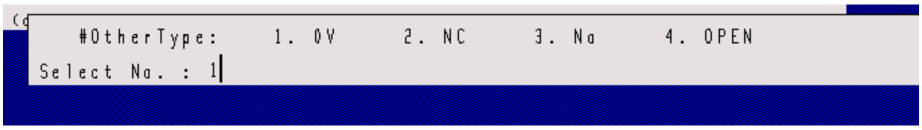
At the above prompt, check whether pins have been set correctly. To change a pin setting, enter the type number (1, 2, or 3) of that pin. If you make no change to the settings, enter "n".

Implied Pin types are classified by the rules listed below, depending the name of each pin.

Note, however, that these rules do not apply if the setting has been previously changed.

Type	Implied Pin name
Power supply (Power)	String containing VCC, VDD, Power, or a + String beginning with V String made up of numeric characters only
Ground (GeneralGround)	String containing GND,VEE,VSS,Ground, or a -
Others (#OtherType)	NC Other than the above

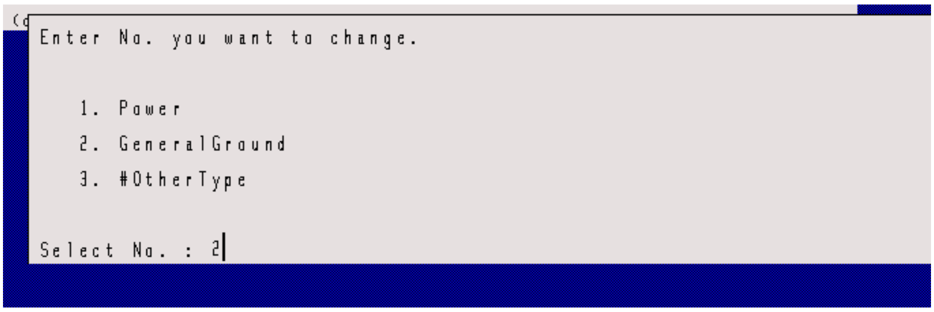
If you enter the type number of an implied pin whose setting you want to change, a menu appears as shown below:



```
   #OtherType:    1. 0V     2. NC     3. No     4. OPEN
Select No. : 1
```

Figure 2.9 Selecting an Implied Pin

Enter the number of the implied pin whose type you want to change. You can select multiple implied pins by entering their numbers separated by a comma (,).

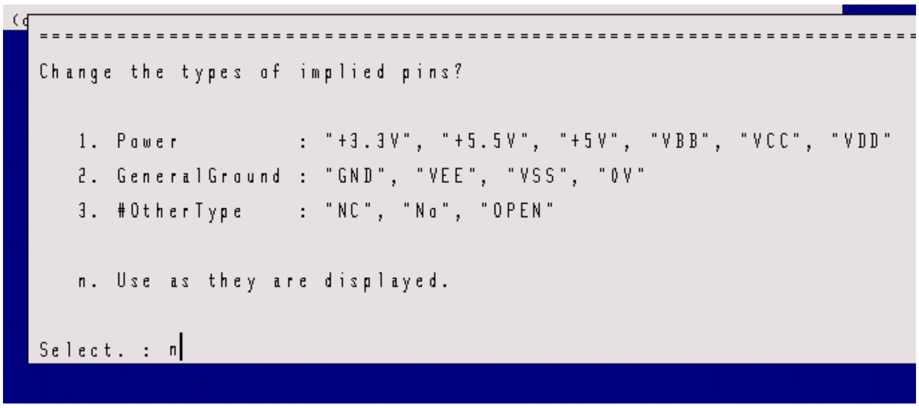


```
Enter No. you want to change.


   1. Power
   2. GeneralGround
   3. #OtherType

Select No. : 2
```

Figure 2.10 Selecting an Implied Pin Type

Enter the number of the new type of the implied pin(s) at the prompt. The banner will then return to the implied pin setting menu, listing the implied pins classified according to the new settings.

.

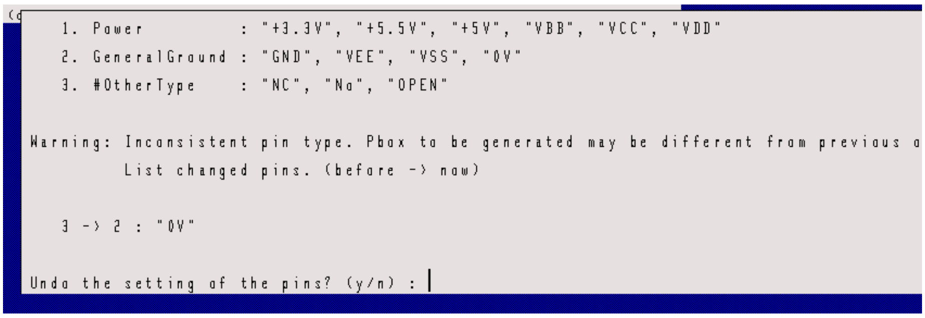


```
==================================================================
Change the types of implied pins?


   1. Power         : "+3.3V", "+5.5V", "+5V", "VBB", "VCC", "VDD"
   2. GeneralGround : "GND", "VEE", "VSS", "0V"
   3. #OtherType    : "NC", "No", "OPEN"

   n. Use as they are displayed.

Select. : n
```

Figure 2.11 Changed Implied Pin Settings

To change the setting for another pin, repeat the above procedure. To accept all of the current settings, enter "n".

If the settings accepted by entering "n" contains a pin’s type setting different from its previous implied pin type, the warning message to that effect appears as shown below:



```
   1. Power         : "+3.3V", "+5.5V", "+5V", "VBB", "VCC", "VDD"
   2. GeneralGround : "GND", "VEE", "VSS", "0V"
   3. #OtherType    : "NC", "No", "OPEN"

Warning: Inconsistent pin type. Pbox to be generated may be different from previous o
         List changed pins. (before -> now)

   3 -> 2 : "0V"

Undo the setting of the pins? (y/n) :
```

Figure 2.12 Warning Message after Setting Change

When there is no problem with the new type settings of the pin, enter "n" at the above prompt.

In this case, the identically named pins of the previously created power box and the new one are different in type.

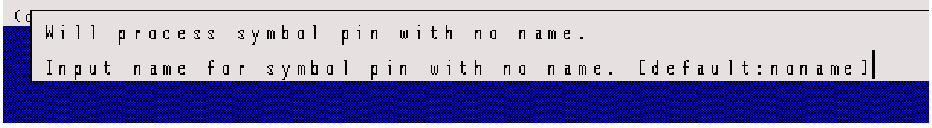
To cancel the new settings and revert to the previous type, enter "y" at the prompt.

Use the above procedure to change implied pin settings.

(3) Setting a name for an unnamed pin

For a symbol pin unnamed on the symbol page , set an arbitrary pin name to be given to that pin.

This setting is needed because System Designer requires pin labels (pin names) while DCS can handle pins without pin names.



```
Will process symbol pin with no name.
Input name for symbol pin with no name. [default:noname]
```

Figure 2.13 Setting a Name for Unnamed Pin

Enter an arbitrary pin name to be given to a symbol pin with no pin name.

Characters invalid to pin names are: ( ) | [ ] { } : , < > and space.

The character string entered last is used as the default pin name to be displayed at the next execution. To use the default name being displayed,simply press RETURN. The complete pin name to be assigned is <input string>_<@LogicalNumber>.

If you reflect LCDB information in the schematic as described in Section 2.2.3 when pin name entered here does not match the pin name entered in step (6) "Setting a name for an unnamed pin" in Section 2.1.4, the pin name in output data becomes that defined in the components database for schematic design (LCDB) .

(4) Generating a SD transfer file
When you have made the above settings, a SD transfer file is generated from DCS .

Quitting SD Transfer File Generator

The program terminates normally with the following files generated:

Output file	File name
SD transfer file	<top hierarchical level name>[1].dsf
Instance information file	<top hierarchical level name>.ref
Resource file	dcslandata.rsc
Property conversion tablep file	propcnv.htb

[1] The top hierarchical level name indicates the icon name of the execution environment from DCS. If you have invoked the program from the screen, for example, the top hierarchical level name is "screen". If you have invoked it from the design or collector icon, the name is "<design name> or "<collector name>", respectively.

[<top hierarchical level name>.dsf]
This file contains all symbol and circuit information used for the current design. Schematic data for System Designer is generated from this file.

[<top hierarchical level name>.ref]
This file contains instance information of a circuit. This file is necessary for System Designer to reflect instance information during conversion.

[dcslandata.rsc]
This file is the resource file required for generating the Data resource file (landata.rsc) for use during conversion of the SD transfer file in System Designer .

[propcnv.htb]
This file is the Property conversion table file defining the properties used in System Designer that are to be converted from those used in DCS . Only the

default setting can be specified. To add properties, execute "LCDB Generator " described in Section 2.1.4 or "Data Converter " in Section 2.1.5.

If any error or warning has been detected, check the contents of the files listed below and follow the instruction for the corresponding message in Chapter 6 "MESSAGES" to correct the relevant item of data before retrying execution of the program.

Error , warning , and log messages are recorded in the following files:

Output file	File name
Error file	SDFileGen.err
Warning file	SDFileGen.wrn
Log file	SDFileGen.log <top hierarchical level name>.log

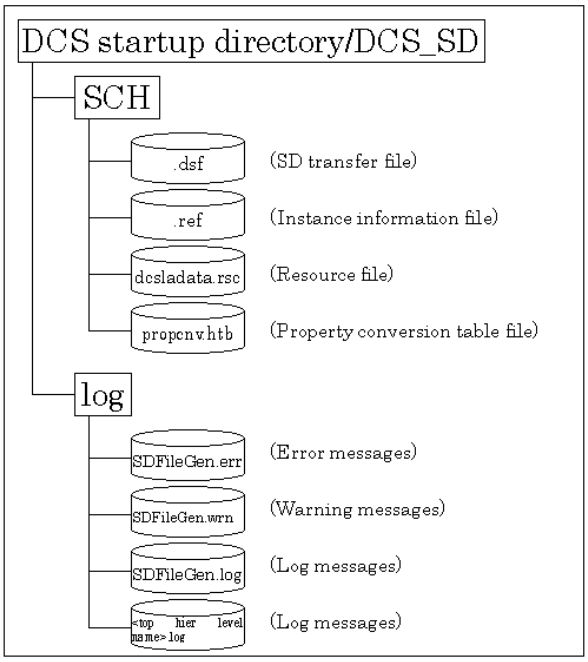


Figure 2.14 Directory Structure

2.1.4 LCDB Generator

LCDB Generator is the program providing the following two major functions:

(1) Function of converting part property data designed with DCS to the components database for schematic design (LCDB) for use in System Designer.

(2) Function of generating resource files (CompKind, propcnv.htb, and Prop Spec) for System Designer.

Starting LCDB Generator

To execute the program, select the following menu command string on the screen or collector page:

```
/PERFORM/INTERFACE/System Designer/DATA CONVERSION/LCDB Gen.
```

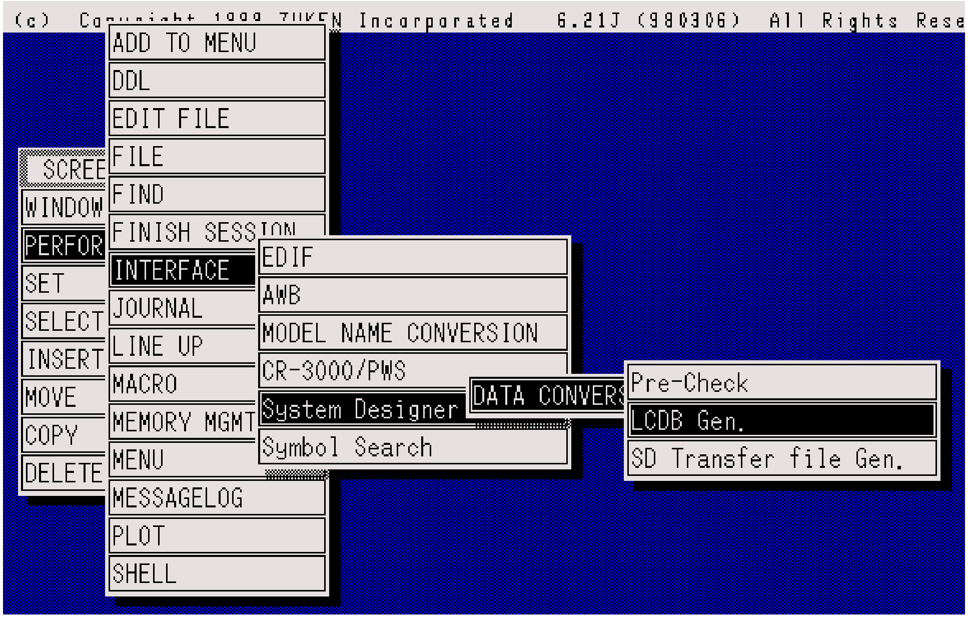


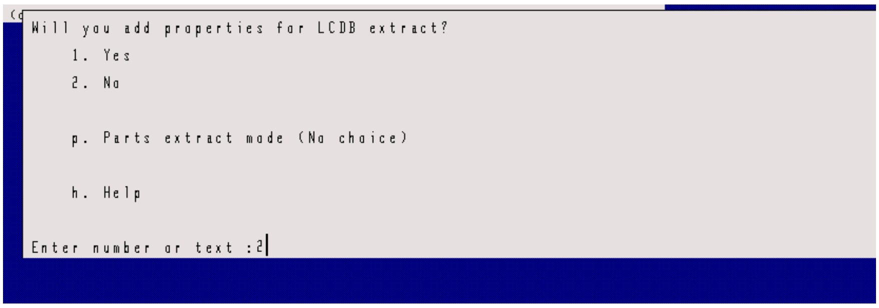
Figure 2.15 Menu Selection for LCDB Generator

Selecting the above command string starts the program. Follow the procedure below according to the prompts and instructions on the banner.

(1) Selecting a mode
Take this step in the same way as in (1) "Selecting a mode" in Section 2.1.3 "SD Transfer File Generator".

(2) Selecting whether to add properties

Select whether to add properties to the components database for schematic design (LCDB) .



```
Will you add properties for LCDB extract?
    1. Yes
    2. No

    p. Parts extract mode (No choice)

    h. Help

Enter number or text :2
```

Figure 2.16 Selecting Whether to Add Properties

Each item on the above menu is described below:

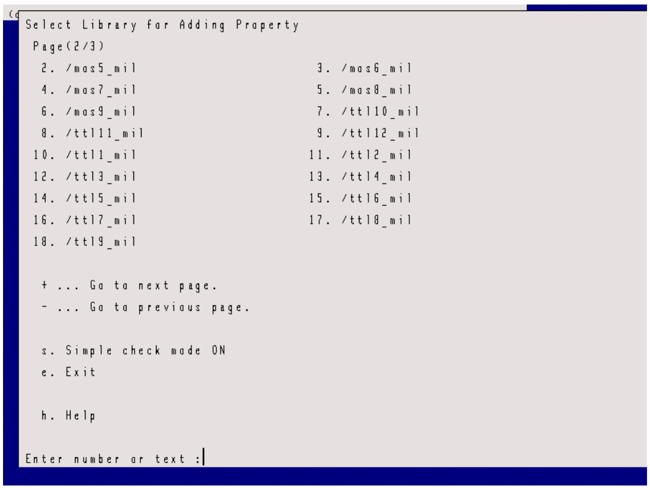
- Selecting "1" adds properties to the components database for schematic design (LCDB) . The banner changes to the menu shown in (3) "Selecting a library."
- Selecting "2" adds no property to the components database for schematic design (LCDB) . The banner changes to the menu shown in (6) "Setting implied pins ."
- Selecting "p" toggles Parts Extraction Mode between "(no choice)" and "(1 gate/multiple gate parts choice)", selecting the part to be extracted. In "Parts Extraction Mode (No choice)," gate parts and package parts are both registered in the components database for schematic design (LCDB).
  "Parts Extraction Mode (1 gate/multiple gate parts choice)," depending on the value "@SubUnit" of the physical page, gate parts or package parts will be registered in the components database for schematic design(LCDB) .

@SubUnits	Respinse
One	It registers as package parts
Two or more	It registers as gate parts

- Selecting "h" displays the help message.

(3) Selecting a library

In this step, select the library to which you want to add properties.



```
Select Library for Adding Property
 Page(2/3)
  2. /mos5_mil                       3. /mos6_mil
  4. /mos7_mil                       5. /mos8_mil
  6. /mos9_mil                       7. /ttl10_mil
  8. /ttl11_mil                      9. /ttl12_mil
 10. /ttl1_mil                      11. /ttl2_mil
 12. /ttl3_mil                      13. /ttl4_mil
 14. /ttl5_mil                      15. /ttl6_mil
 16. /ttl7_mil                      17. /ttl8_mil
 18. /ttl9_mil

  + ... Go to next page.
  - ... Go to previous page.

  s. Simple check mode ON
  e. Exit

  h. Help

Enter number or text :
```

Figure 2.17 Selecting a Library

Each item on the above menu is described below:

- Page(2/3)
  Indicates the current page number out of the total number of pages of the library list. The fraction is: Page(<current page number>/<total number of pages>).
- Selecting the number specifies a library which properties are to be added to. You can select multiple libraries by entering their numbers separated by a space. The banner changes to the menu shown in (4) "Selecting a page."
- Selecting "s" toggles Simple check mode between ON and OFF, selecting the range of search for the property to be added.
  When Simple check mode is OFF, all design icons are searched. Set the mode to OFF, for example, when the properties defined for design icons have not been unified in each library.
  When Simple check mode is ON, only one design icon in the selected library is searched, resulting in shorter wait time. Simple check mode defaults to ON.
- Selecting "e" generates components libraries for schematic design (LCDBs).

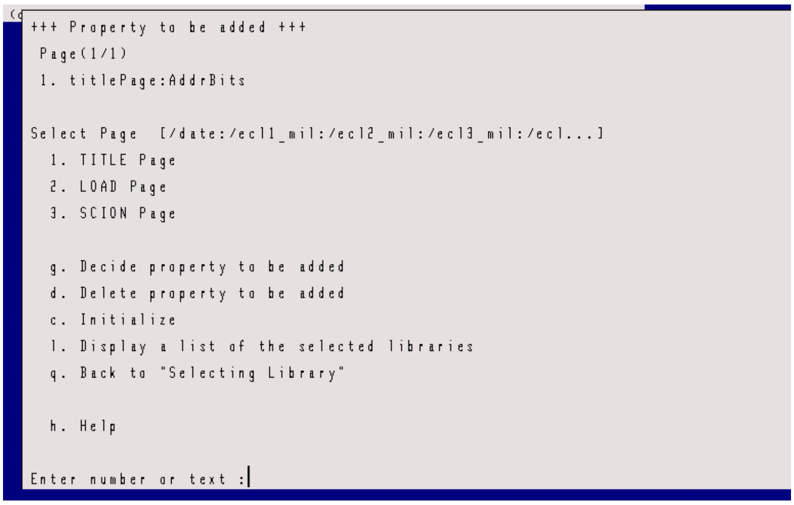
- Selecting "h" displays the help message.

Notes:

- The library you select here does not represent the one from which a components database for schematic design (LCDB) is to be generated; it is simply a library to which information is added. The components database for schematic design (LCDB) is generated from all libraries in the execution environment.
- If the desired page or property has not been displayed when you have set Simple check mode to ON, set the mode to OFF,then select the library again.

(4) Selecting a page

In this step, select the page which properties are to be extracted from.



```
+++ Property to be added +++
 Page(1/1)
 1. titlePage:AddrBits

Select Page  [/date:/ecl1_mil:/ecl2_mil:/ecl3_mil:/ecl...]
  1. TITLE Page
  2. LOAD Page
  3. SCION Page

  g. Decide property to be added
  d. Delete property to be added
  c. Initialize
  l. Display a list of the selected libraries
  q. Back to "Selecting Library"

  h. Help

Enter number or text :
```

Figure 2.18 Selecting a Page

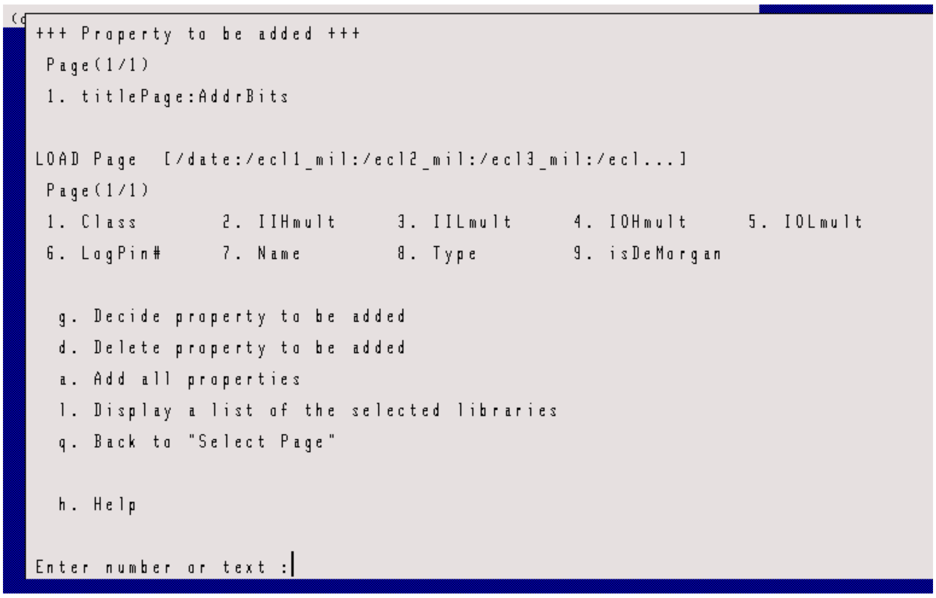
Each item on the above menu is described below:

- Selecting the number specifies a page which properties are to be extracted from. The banner changes to the menu shown in (5) "Selecting properties". The pages from which properties can be extracted are the TITLE, LOAD and SCION pages. Note, however, that nonexistent pages are preceded by "" in place of a number, indicating that they cannot be selected.
  When properties have already been added in (5) "Selecting properties," the property names are displayed under "+++ Property to be added +++" at the top of the menu.

- Selecting "g" accepts information on the properties displayed under "+++ Property to be added +++".
- Selecting "d" deletes the last property selected.
  There are three ways to specify the command as shown below.
  1. d : The additional property selected most recently will be deleted.
  2. d <No>: The additional properties selected as <No> will be deleted.
  3. d [No.1]-[No.2]:The selected sequential additional properties [No.1] - [No.2] will be deleted.
- Selecting "c" initializes the settings.
- Selecting "l" displays the library currently being selected.
- Selecting "q" returns you to the menu shown in (3) "Selecting a library."
- Selecting "h" displays the help message.

(5) Selecting properties

In this step, select properties to be added.



```
+++ Property to be added +++
 Page(1/1)
 1. titlePage:AddrBits

LOAD Page  [/date:/ecl1_mil:/ecl2_mil:/ecl3_mil:/ecl...]
 Page(1/1)
 1. Class        2. IIHmult      3. IILmult      4. IOHmult      5. IOLmult
 6. LogPin#      7. Name         8. Type         9. isDeMorgan

  g. Decide property to be added
  d. Delete property to be added
  a. Add all properties
  l. Display a list of the selected libraries
  q. Back to "Select Page"

  h. Help

Enter number or text :
```

Figure 2.19 Selecting Properties

- Selecting the number specifies a property to be added. You can select multiple properties by entering their numbers separated by a space. Note that a property which has already been selected is preceded by "_" in place of a number. The property name is displayed in the <page name>:<property name> format under "+++ Property to be added +++".
- If "a" is selected, all the properties on the subject pages can be selected.

- Selecting "g" accepts information on the properties displayed under "+++ Property to be added +++".
- Selecting "d" deletes the last property selected.
  There are three ways to specify the command as shown below.
  1. d :The additional property selected most recently will be deleted.
  2. d <No> :The additional properties selected as <NO>will be deleted.
  3. d [No.1]-[No.2]:The selected sequential additional properties [No.1] - [No.2] will be deleted.
- Selecting "l" displays the library currently being selected.
- Selecting "q" returns you to the menu shown in (4) "Selecting a page."
- Selecting "h" displays the help message.
- If you have selected the SCION page in (4) "Selecting a page," the following option is added:

```
      p. PCMAC OFF
```

Selecting "p" toggles the setting between ON and OFF to enable or disable the PWS_PCMAC_NO property in System Designer.
"PCMAC ON" enables the PWS_PCMAC_NO property.
"PCMAC OFF" disables the PWS_PCMAC_NO property.

NOTE: If you directly select a PWS_PCMAC_NO property, the value_following ":" is added as it is.

If the contents of the menu for "selecting a library, page, or property" cannot fit on one page of the menu, the following options are added to the menu:

```
      + ... Next page
      - ... Previous page
```

- Selecting "+" displays the next page of the menu.
  Selecting "-" displays the previous page of the menu.

(6) Setting implied pins
Take this step in the same way as in (2) "Setting implied pins " in Section 2.1.3 "SD Transfer File Generator ."

(7) Setting a name for an unnamed pin
Take this step in the same way as in (3) "Setting a name for an unnamed pin" in Section 2.1.3 "SD Transfer File Generator ."

Quitting LCDB Generator

The program terminates normally with the following files generated:

Output file	File name
Parameter file	<library name>.prf
Component file	<library name>.cmp
Database file	<library name>.dbf
Property conversion tablepfiler	propcnv.htb
Function type definition file	CompKind
Property specification file	PropSpec

[.prf] This file is the parameter file defining the properties to be set in the System Designer database file and component file.

[.cmp] This file is the component file defining pin information to be used for each component in System Designer .

[.dbf] This file is the database file defining the user defined properties to be used for each component in System Designer .

[propcnv.htb] This file is Property conversion table file defining the properties used in System Designer that are to be converted from those used in DCS.

[CompKind] This file is Function type definition file for System Designer to distinguish gate components and part components further by function. The header characters for reference names are also defined in this file.

[PropSpec] This file is Property specification file defining the properties to be assigned to individual objects by System Designer . Properties cannot be referenced or edited with Schematic Editor unless they have been defined in this file.

NOTE: <library name> indicates a library name consisting of up to 10 characters if your hardware platform is an HP-CISC machine. The excess characters are ignored.

If any error or warning has been detected, check the contents of the files listed below and follow the instruction for the corresponding message in Chapter 6 "MESSAGES" to correct the relevant item of data.

Error , warning , and log messages are recorded in the following files:

Output file	File name
Error file	LcdbGen.err
Warning file	LcdbGen.wrn
Log file	LcdbGen.log

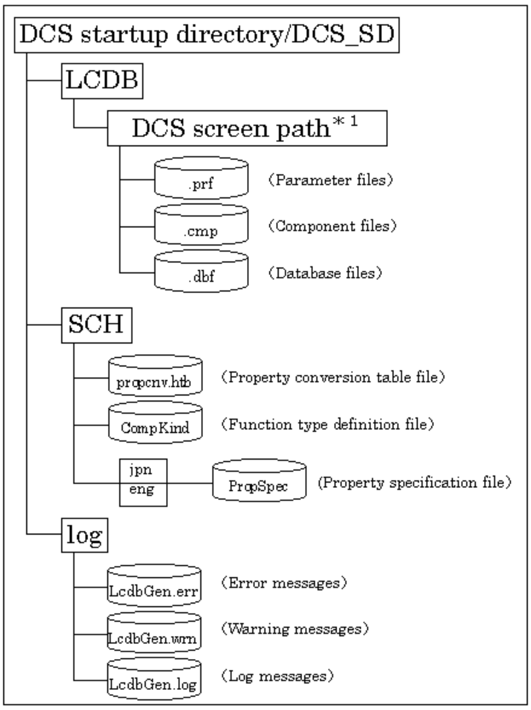


Figure 2.20 Directory Structure

*1 Although DCS screen path shows the path from the screen environment of DCS,Two kinds of specification can be performed by setup.
For details, see section 4.2.9 "Turning transfer data tree structure ON/OFF".

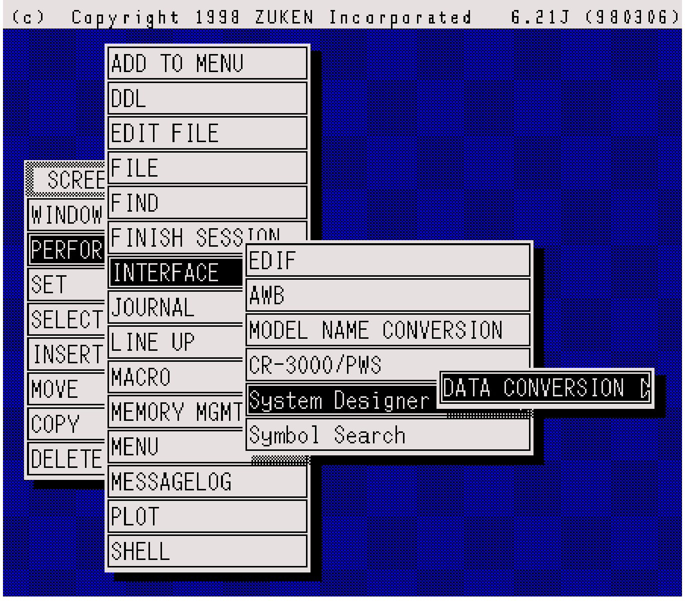
2.1.5 Data Converter

Data Converter is the program that executes a batch of functions provided by "Pre-Checker ", "SD Transfer File Generator ", and "LCDB Generator" describes in Sections 2.1.2, 2.1.3, and 2.1.4.

Starting Data Converter

To execute the program, select the following menu command string on the screen,collector, or circuit page :

/PERFORM/INTERFACE/System Designer/DATA CONVERSION



(c) Copyright 1998 ZUKEN Incorporated 6.21J (980306)

Figure 2.21 Menu Selection for Data Converter

Selecting the above command string starts the program. Follow the procedure below according to the prompts and instructions on the banner.

(1) Setting a mode
Take this step in the same way as in (1) "Setting a mode" in Section 2.1.3 "SD Transfer File Generator".

(2) Selecting whether to add properties
Take this step in the same way as in (2) "Selecting whether to add properties" in Section 2.1.4 "LCDB Generator".

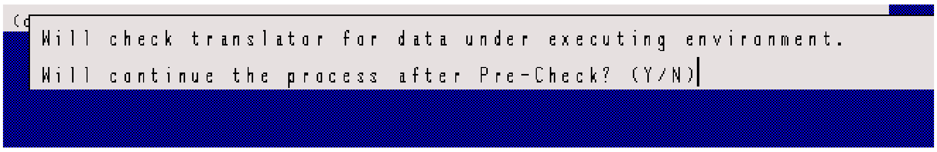
(3) Selecting a library
Take this step in the same way as in (3) "Selecting a library" in Section 2.1.4 "LCDB Generator".

(4) Selecting a page
Take this step in the same way as in (4) "Selecting a page" in Section 2.1.4 "LCDB Generator".

(5) Selecting properties
Take this step in the same way as in (5) "Selecting properties" in Section 2.1.4 "LCDB Generator".

(6) Setting implied pins
Take this step in the same way as in (2) "Setting implied pins" in Section 2.1.3 "SD Transfer File Generator".

(7) Selecting whether to continue processing after prechecking
Select whether to continue processing after Pre-Checker has issued a warning .



```
Will check translator for data under executing environment.
Will continue the process after Pre-Check? (Y/N)
```

Figure 2.22 Selecting Whether to Continue Processing after Pre-Check

Enter "Y" to continue processing even when Pre-Checker has issued a warning.
To quit processing after that, enter "N".
If an error is detected, the program terminates even though "Y" is entered at this prompt.

WARNING: If "Y" has been entered, data resulting in a warning may also results in an error or warning in System Designer . You should therefore convert the data in System Designer as long as the conversion develops no problem after confirming the message.

(8) Setting a name for an unnamed pin
Take this step in the same way as in (3) "Setting a name for an unnamed pin" in Section 2.1.3 "SD Transfer File Generator".

(9) Generating a SD transfer file
When you have made the above settings, a SD transfer file is generated from DCS.

Quitting Data Converter

The program terminates normally with the following files generated:

Output file	File name
SD transfer file	<top hierarchical level name>.dsf
Instance information file	<top hierarchical level name>.ref
Resource file	dcslandata.rsc
Parameter files	<library name>.prf
Component files	<library name>.cmp
Database files	<library name>.dbf
Property conversion table file	propcnv.htb
Function type definition file	CompKind
Property specification file	PropSpec

If any error or warning has been detected, check the contents of the files listed below and follow the instruction for the corresponding message in Chapter 6 "MESSAGES" to correct the relevant item of data.

Error , warning , and log messages are recorded in the following files:

Output file	File name
Error file	DataConv.err
Warning file	DataConv.wrn
Log file	DataConv.log

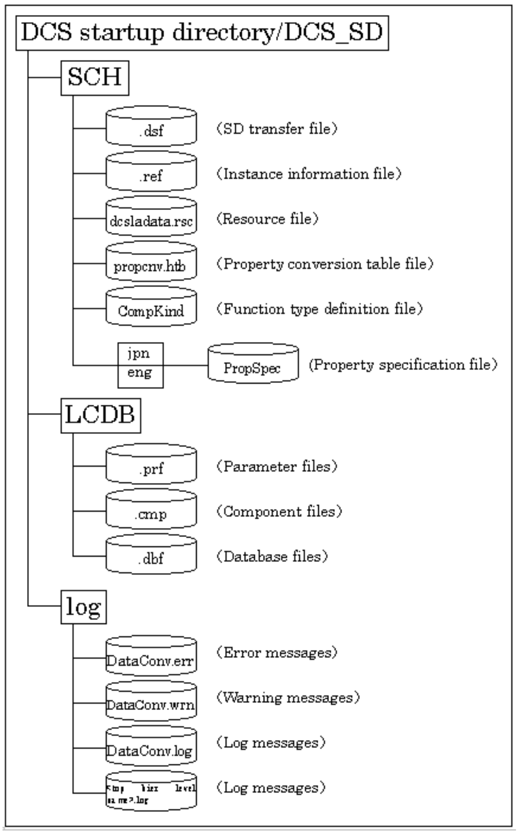


Figure 2.23 Directory Structure

2.2 DCS Translator

DCS Translator provides the following program for converting DCS schematic data to System Designer data:

- DCS Translator program

2.2.1 Before Starting Data Conversion

Before converting DCS schematic data and symbol library figures information to System Designer data, perform the following preparations:

(1) Adjusting the scale
DCS is a system based on the inch-measured coordinate system;
System Designer is a system based on the millimeter-measured coordinate system.
Therefore, converting DCS figures information to System Designer data using the original DCS scale outputs off-grid data. (If DCS data has been designed with a pin spacing of 1/8 inch, for example, converting the data using the same scale results in System Designer data with a pin spacing of 3.175 millimeters, which do not fit normal grid points (spaced in millimeters).)
You should therefore apply an arbitrary scale factor to DCS figures information when converting it to System Designer data so that the output data fit on the grid.

- Check the grid used for schematic data.
  Pins (signal lines) are designed usually to be placed at grid points. Assume that a 1/8-inch (3.175 millimeters) grid has been used.
- Determine the grid to be used for System Designer schematic data. Assuming that a minimum grid spacing has been set to 1 millimeter, obtain the closest integer multiple of the grid spacing. For the dimension close to the actual dimension of 1/8-inch (3.175 millimeters) used for the DCS libraries, you should select 4 or 2 millimeters. (Select the value considering the overall scale and the pitch used in symbol libraries provided by System Designer .)

The symbol libraries provided by System Designer conform to IEEE Std 91-1984, defining data dimensioned to be 1/4 of the standard. The symbol libraries provided by DCS are based on the dimensions in the same standard and MIL standards. Applying a scale factor of 1/4 to DCS data results in data closer in dimension to figures defined in the symbol libraries provided by System Designer.

(2) Customizing configuration files

Before converting System Designer schematic data, check each configuration file. The following configuration files are generated during conversion from DCS :

- Property conversion table file (propcnv.htb)
- Function type definition file (CompKind)
- Property specification file (PropSpec)

These three files are described below.

a. Property conversion table file (propcnv.htb)

Specify this file when you want to convert property names used in DCS to different property names in System Designer . The file is generated as the following file:

```
./DCS SD/SCH/propcnv.htb
```

The property names in the property conversion table file generated from DCS have been set to be identical between DCS and System Designer. To use a property name different from the one used in DCS , therefore, edit this file using an editor such as vi.

For details on editing the file, see Section 4.3.2 "Property Conversion Table File ."

b. Function type definition file (CompKind)

This file is generated as the following file during execution of the "DATA CONVERSION" or "LCDB Gen". command:

```
./DCS SD/SCH/CompKind
```

The type display characters in the function type definition file generated from DCS have been set to be the same names as

reference header characters. To assign a different name to a type display character, therefore, edit the parameter.
For details on editing this file, see Section 4.3.3 "Function Type Definition File".

c. Property specification file (PropSpec)
This file is generated as the following file during execution of the "DATA CONVERSION" or "LCDB Gen". command:

```
./DCS_SD/SCH/jpn/PropSpec
./DCS_SD/SCH/eng/PropSpec
```

The display character strings in the Property specification file generated from DCS have been set to be the same names as property names. To assign a different name to a display character string, therefore, edit the parameter.
For details on editing this file, see Section 4.3.4 "Property Specification File".

(3) File Transfer
If DCS and System Designer have been installed on different machines, transfer a series of files generated from DCS (directories and files in "<DCS startup directory>/DCS_SD" to the machine on which System Designer has been installed.

NOTE: The directory in ".../DCS_SD" contains the resource files required for conversion in System Designer. It is therefore prohibited to perform conversion after moving only the SD transfer file (.dsf) to a different directory.

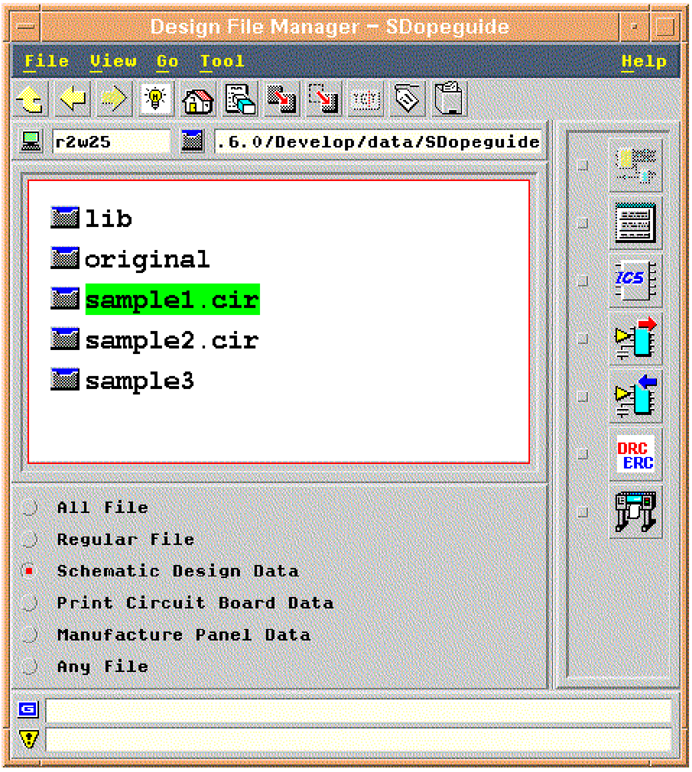
2.2.2 DCS Translator program

The DCS Translator program converts the SD transfer file generated from DCS to System Designer schematic data.

Starting the DCS Translator program

Use the following procedure to convert the SD transfer file to System Designer schematic data.

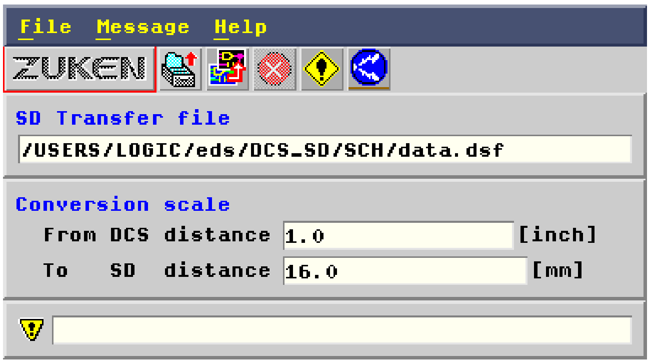
(1) Invoke File Manager for System Designer in a directory of <arbitrary directory (DCS startup directory)>/DCS SD/SCH.



○ All File
○ Regular File
● Schematic Design Data
○ Print Circuit Board Data
○ Manufacture Panel Data
○ Any File

Figure 2.24 File Manager

(2) Double-click on the SD transfer file to be converted or select it and click on DCS Translator on the action menu to invoke DCS Translator.



SD Transfer file
/USERS/LOGIC/eds/DCS_SD/SCH/data.dsf

Conversion scale
From DCS distance 1.0 [inch]
To SD distance 16.0 [mm]

Figure 2.25 Setting up DCS Translator

Each item on the above menu is described below:

- "Open" button
  Lets you select the SD transfer file to be converted.

- "Convert" button
  Executes DCS Translator.

- "Stop" button
  Stops processing of conversion.

- "Errors" button
  Displays error messages in the message confirmation dialog.

- "Warnings" button
  Displays warning messages in the message confirmation dialog.

- SD transfer file
  Displays the name of the SD transfer file (.dsf) to be converted.

- Conversion scale
  Specify the values (real numbers) for determining the scale to be used during conversion. The example in Figure 2.25 converts 1.0 inch in DCS to 16.0 millimeters in System Designer . The scale factor is therefore 16.0 millimeters/1.0 inch (25.4 millimeters) = 0.629921.
  To determine the scale factor, see (1) "Adjusting the scale" in Section 2.2.1.

(3) After selecting the SD transfer file to be converted, click on the "Convert" button to execute the program.

(4) If a components database for schematic design (LCDB) has been generated, the program to "Component Attribute reload from LCDB" is executed automatically to obtain additional information from the LCDB after conversion of all schematic data has been completed. (The program is not executed when only symbol libraries are converted.)

(5) If the program terminates with an error or warning , the message confirmation dialog displays the corresponding error or warning message.

NOTE: If the schematic and symbol libraries currently being converted have the samedirectory structure as previously converted data, the previous output is over written with the current output.

Quitting DCS Translator program

The program terminates normally with the following files generated, including symbol libraries, schematic sheets, and circuit blocks:

Output file	File name
Schematic directories	<schematic name>.cir
Schematic sheets	<001-999>.sht
Symbol libraries	<symbol name>.smb
Circuit blocks	<symbol name>.blk
Resource path files	rcpath
Data resource file	landata.rsc

Error and warning messages are recorded in the following files:

Output file	File name
Error file	ed2sd.err
Warning file	ed2sd.wrn

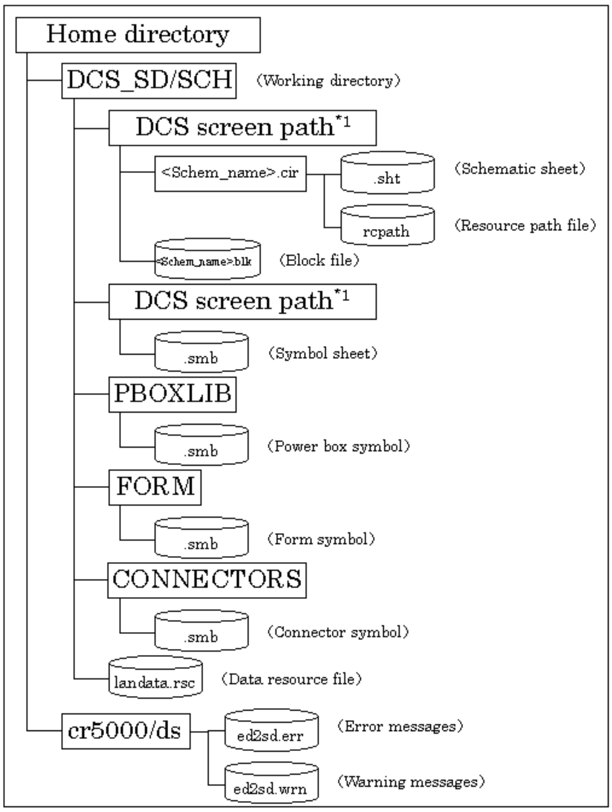


Figure 2.26 Directory Structure

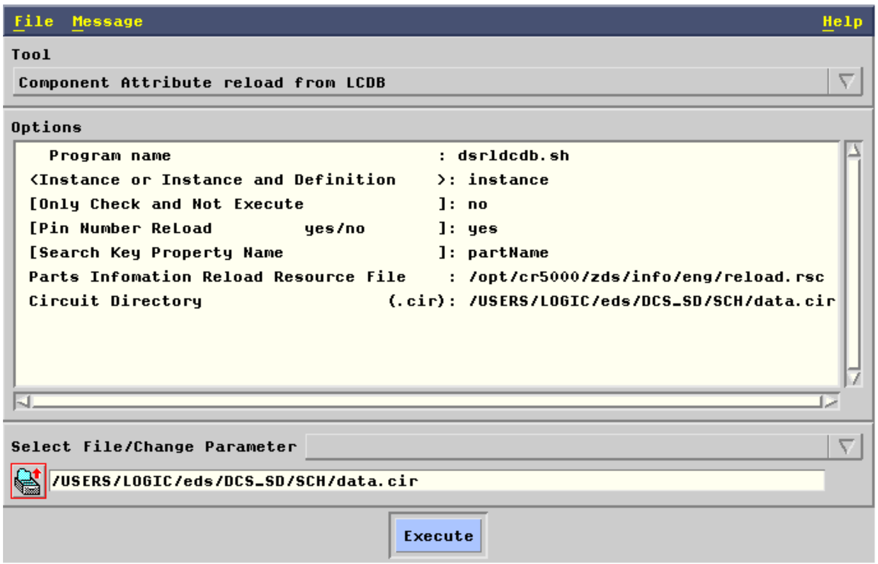
*1 Although DCS screen path shows the path from the screen environment of DCS,Two kinds of specification can be performed by setup.
For details, see section 4.2.9 "Turning transfer data tree structure ON/OFF".Reflecting LCDB Information in Schematic

2.2.3 Reflecting LCDB Infomation in Schematic

DCS Translator executes the LCDB reload program automatically after conversion of schematic data. If you modify the components database for schematic design (LCDB) after conversion of schematic data, reflect the updated information in the schematic data using the procedure described below.

To reflect the contents of the LCDB in the schematic, use the LCDB reload pro gram. The program is executed by selecting "Component Attribute reload from

LCDB" from "Data Converter".



Tool

Component Attribute reload from LCDB

Options

```
   Program name                               : dsrldcdb.sh
 <Instance or Instance and Definition        >: instance
 [Only Check and Not Execute                 ]: no
 [Pin Number ReLoad          yes/no          ]: yes
 [Search Key Property Name                   ]: partName
 Parts Infomation Reload Resource File        : /opt/cr5000/zds/info/eng/reload.rsc
 Circuit Directory                      (.cir): /USERS/LOGIC/eds/DCS_SD/SCH/data.cir
```

Select File/Change Parameter

/USERS/LOGIC/eds/DCS_SD/SCH/data.cir

Figure 2.27 Component Attribute reload from LCDB

Each item on the above menu is described below:

- Program name
  Fixed
- Instance or Instance & Definition
  To reflect only instance information, set this item to "instance". To reflect both instance and definition levels, set it to "all".
- Pin Number ReLoad
  Select whether to reflect pin numbers. Set this item to "yes" to reflect pin numbers or "no" not to reflect them.
- Search Key Property Name
  Specify the property to be used for searching the components database for schematic design (LCDB) . Normally, specify "partName", "partNumber", or "cdbName".
- Circuit Directory
  Specify the directory containing the schematic

For details on the LCDB reload program, see Section 4.15 "Component information library for circuit design Reload Program" in the "System Designer User’s Guide Vol. 1".

2.2.4 Power and Ground Nets Connected to Implied Pins

Although power and ground pins created as implied pins are converted to power box symbols during schematic conversion, they are not reflected in the schematic.
You need to add the power net or power box symbol using the System Designer editor.

You can use either of the following two methods to add a power supply.

(1) In case of a multiple-power circuit
Insert the power box in the schematic.

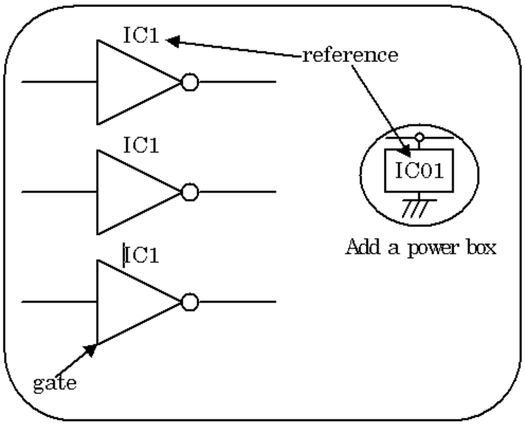


Figure 2.28 Adding a Power Box

(2) In case of a single-power circuit
If a components database for schematic design (LCDB) has been generated, set schematic sheet properties of "Default Power Net" and "Default Ground Net" to the "power net name" and "ground net name", respectively.
If the components database for schematic design (LCDB) is not generated, you need to insert the power box in the schematic.

Chapter 3　RESTRICTIONS

This chapter describes restrictions on the programs included in DCS Translator Sub-kit and DCS Translator.

(1) Updating parts
If you change figures information (such as a shape, pin information, or property) in a symbol library in DCS , the obsolete pin or property information is output to the SD transfer file unless "UPDATE PARTS" has been executed on the schematic using that symbol. An attempt to convert the SD transfer file in System Designer results in fail. When you have modified a symbol, be sure to execute "UPDATE PARTS" on the schematic using that symbol.

(2) "Leaf" property
If the "Leaf" property for a hierarchical symbol used in the schematic has been set to "Y" (defining the hierarchical symbol as a leaf symbol) in DCS, conversion with the reference name not displayed on the circuit page fails to convert the symbol as a leaf symbol. To convert the symbol as a leaf symbol, be sure to assign or update the reference name on the circuit page in advance.

(3) Form icons
Information on a form icon cannot be converted alone. To convert information on a form, insert the form on the circuit page.

(4) Document icons
Document icons cannot be converted.

(5) Filled Polygons
Filled areas of polygons with crossing lines may not be converted normally.

(6) Size indicators

Size indicators in DCS indicate bus width. In System Designer , however, they correspond to polygons and text without having such a function as in DCS .

(7) Subwires

Subwires in DCS are displayed as dotted signal lines. In System Designer, they appear as general signal lines which cannot be distinguished as subwires.

(8) Primitive Shapes

Each primitive shape in DCS is a single geometric shape. In System De signer , however, it consists of several polygons. Therefore, be careful when moving them.

(9) Text table No.

For the properties and text comments converted as schematic data by the DCS Translator program , the table numbers closest in character height are selected from the text table for data resources used during conversion.

When both single-byte and double-byte characters are used together, the double-byte character width is twice the single-byte character width.

Characters scaled in DCS are not converted into their correct size.

(10) Color information

For color information, the default values in the data resource file are used.

Color information used in DCS is not reflected.

(11) Line width

Line width table used in DCS (four types: fine to bold) corresponds to the line width table number (0-3) in System Designer .

(12) Line types

Line types used in DCS correspond to the solid line (solid), dashed line (dash), single-dot stich line (dash1), and two-dot chain line (dash2) in System Designer.

Line types in DCS correspond to those in System Designer as listed below:

DCS line types	System Designer line types
(1)SOLID/0.1	(0)Solid line
(2)SOLID/0.2	(0)Solid line
(3)SOLID/0.3	(0)Solid line
(4)SOLID/0.3	(0)Solid line
(5)DASH/0.1	(1)Dashed line
(6)CENTER	(2)Single-dot stich line
(7)DIMENSION	(0)Solid line
(8)HIDDEN	(1)Dashed line
(9)PHANTOM	(3)Two-dot chain line
(10)SECTION	(0)Solid line
(11)STITCH	(1)Dashed line
(12)CUTTING	(1)Dashed line
(13)VISIBLE	(0)Solid line

The parenthesized number for each line type indicates its system line type number.

(13) Negative logic representing characters

The DCS Translator program does not convert negative logic representing characters. For circuits with negative logic representations in signal names or terminal names, set the negative logic representing characters in the data resource file "landata.rsc" (as nlogicStartChar and nlogicEndChar).

(14) text consisting of several lines.

If the text consists of several lines, the lines will not be aligned when displayed in System Designer . In DCS, the lower left of the first line of a text will be the reference position (default) while in System Designer , the lower left position is set as the reference point (default) regardless of the number of lines. As a result, the texts will not be aligned.

(15) Overlap of properties (property name that contains "=")

If a property text on the symbol page in DCS contains "=", the text will be converted to two viewers under the same property name in System Designer .

(16) Disagreement of a pin and a connector

In the case of a hierarchical symbol, when the connector name on the symbol pin on a symbol page and a circuit page is inharmonious, the connector on a circuit page is deleted.

Chapter 4 DATA CONVERSION SPECIFICATIONS

This chapter describes the items to be checked by Pre-Checker, the method for converting each type of data, and the contents of configuration files.

4.1 Pre-Checker

This section describes the check items of Pre-Checker executed in DCS .

4.1.1 Same Names

Description:The program checks the data to be converted to see if there are the same names of files, libraries, and/or partition icons in the environment at the same hierarchical level. In DCS , you can place identically named icons in the environment at the same level as long as the icons are different in type.

When they are converted to System Designer data, however, the names are duplicated because icon names are converted to directory or file names.

Response:DCS stops processing in this case.

Be sure to change the relevant icon names to make all icon names unique.

4.1.2 Unentered Bus Names

Description:The program checks bus wires on the circuit page for unentered bus names.

The unique net names in the "%NETxxxxx" format assigned by DCS are recognized as single wires in System Designer , losing net consistency in System Designer .

Response:DCS assigns bus names automatically to such bus wires, with the exception of some bus wires to which bus names cannot be assigned. For such bus wires, be sure to enter a bus name on the circuit page.

4.1.3 Unentered Pin Names

Description:The program checks pins on the symbol page for unentered pin names .
While DCS can handle unnamed pins, System Designer requires pin labels(pin names).

Response:No problem arises during conversion.
DCS assigns pin names automatically to such pins . Pin names can be set arbitrarily in DCS . Even when no pin name is set for a pin, the default name is assigned.
When assigning a pin name, enter it for the corresponding pin on the symbol page .

4.1.4 Physical Pin Numbers

Description:The program checks for missing physical pin numbers on the physical page.
Physical pin numbers are required when a components database for schematic design (LCDB) is generated.

Response:No problem arises during conversion.
DCS assigns physical pin numbers automatically to such physical pins.
When assigning a physical pin number, set it on the physical page.

4.1.5 Form Icons

Description:The program checks for presence of form icons.
Form icon information cannot be converted independently.

Response:To convert information on a form, insert the form on the circuit page .
No problem arises during conversion, except that forms not inserted on the circuit page are not converted.

4.1.6 Document Icons

Description:The program checks for presence of document icons.
Document icon information cannot be converted.

Response :No problem arises during conversion, except that document icon information is not converted.

4.1.7 String Length

Description:The program checks the string length of file and directory names. The maximum string length of directory names is 14 characters on the HP-CISC machine; it is 31 characters on other types of machines. The maximum string length of file names (basenames) is 10 characters on the HP-CISC machine or 27 characters on another type of machine, allowing for the filename extension.

Response :If a character string exceeds the maximum length, DCS truncates the excess characters automatically.

4.1.8 Inhibited Characters

Description:The program checks file, library, partition, design icon, pin, property, and net names for use of any character inhibited in System Designer.
Inhibited characters are given below:

- Pin name: Space " \
- Net name: Space " \
- Property name: Space " \ # : ( ) { }
- Other names: Space ( ) { } [ ] < > % : @ ^ # ? ’ " * $ & | ; ‘ / ! ~

Response:DCS replaces inhibited characters automatically according to the Inhibited character conversion table file.
To replace an inhibited character with a desired character, change the default in the Inhibited character conversion table file. For

details, see Section 4.3.1 "Inhibited Character Conversion Table File."

4.1.9 Subwires

Description:The program checks for presence of subwires on the circuit page . Subwires are displayed as dotted lines in DCS . In System Designer , they are converted to wires to be displayed as solid lines.

Response:Although no problem arises during conversion, keep in mind that wires and subwires cannot be distinguished in System Designer.

4.1.10 Multiple Power Supplies

Description:The program checks for presence of those power supplies and grounds on the circuit page which are different from implied pin names.

DCS can add new properties for the power supply or ground to package instance in the schematic to change the supply voltage value. In System Designer , however, the property of the power supply or ground is handled as simple text without having the function as a power supply or ground in DCS terminology.

NOTE: This processing is not executed unless the symbol in theschematic exists in the screen environment.

Response:No problem arises during conversion.Power box symbol information is generated from DCS . In System Designer , insert the power box symbol in the schematic for multi-power design.

4.1.11 Size Indicators

Description:The program checks for presence of size indicators on the circuit page.

Size indicators indicate bus width in DCS . In System Designer , however, they correspond to simple polygons and text without having such a function as in DCS.

Response:No problem arises during conversion.

4.1.12 Primitive Shapes

Description:The program checks for presence of primitive shapes on the symbol and circuit pages.
Each primitive shape is a single geometric shape in DCS . In System Designer , however, it consists of several polygons (objects).

Response:No problem arises during conversion.

4.1.13 Filled Polygons

Description:The program checks for presence of filled polygons in the symbol pages,circuit pages, and forms on circuit pages .
Since the filling position of each polygon with crossing lines is determined depending on the number of intersections in DCS , the position may not be converted normally in System Designer .

Response:This item is a restriction.

4.1.14 Local Symbols

Description:The program checks for presence of local symbols on the circuit page .
Unsaved local symbols are not converted.

Response:Execute "SAVE SYM AS COMP" on the circuit page in DCS .

4.2 Data Conversion

This section describes the method for converting each type of data.

4.2.1 DeMorgan Symbols

DCS allows a single symbol to have positive-logic and negative-logic symbol shapes. In System Designer , however, the positive- and negative-logic symbols must be generated separately. From DCS , information on two of the positive- and negative-logic symbol shapes is output to the SD transfer file .

Assuming that "74L00" (NAND symbol) has been defined as a DeMorgan symbol, for example, both of the positive- and negative-logic symbols are generated in System Designer .
The DeMorgan symbol names in System Designer are assigned as shown below:

	Symbol name
Positive logic	Same name as in DCS
Negative logic	<positive-logic symbol name>_N

Figure 4.1 below gives an example of converting "74L00"as a DeMorgan symbol.
The symbol names in System Designer are "74L00" for the positive-logic symbol

and "74L00_N" for the negative-logic symbol.

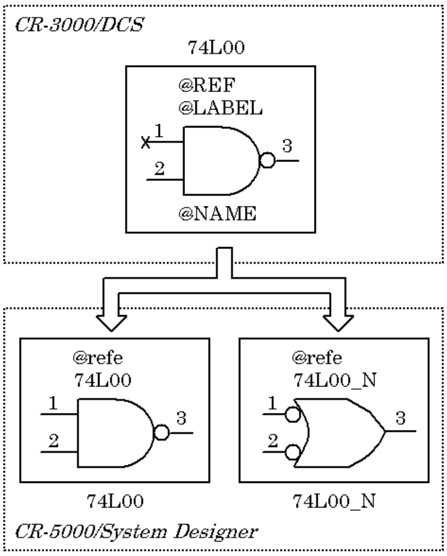


Figure 4.1 Example of Converting a DeMorgan Symbol

4.2.2 Multi-component Symbols

DCS allows multiple symbol shapes to be defined for a single symbol. In System Designer , however, only one symbol shape can be defined for each symbol. From DCS , information on one symbol in each symbol partition is out put to the SD transfer file .

Assuming that "OpAmp" (operational amplifier symbol) has been defined as a multi-component symbol in DCS , for example, the symbol is generated for each symbol partition in System Designer .

The multi-component symbol names in System Designer are assigned in the following format:

```
<DCS symbol name>_<symbol partition number>
```

Figure 4.2 below gives an example of converting "OpAmp" as a multi-component symbol.

The symbol names in System Designer are "OpAmp_1", "OpAmp_2", "OpAmp_3",and "OpAmp_4".

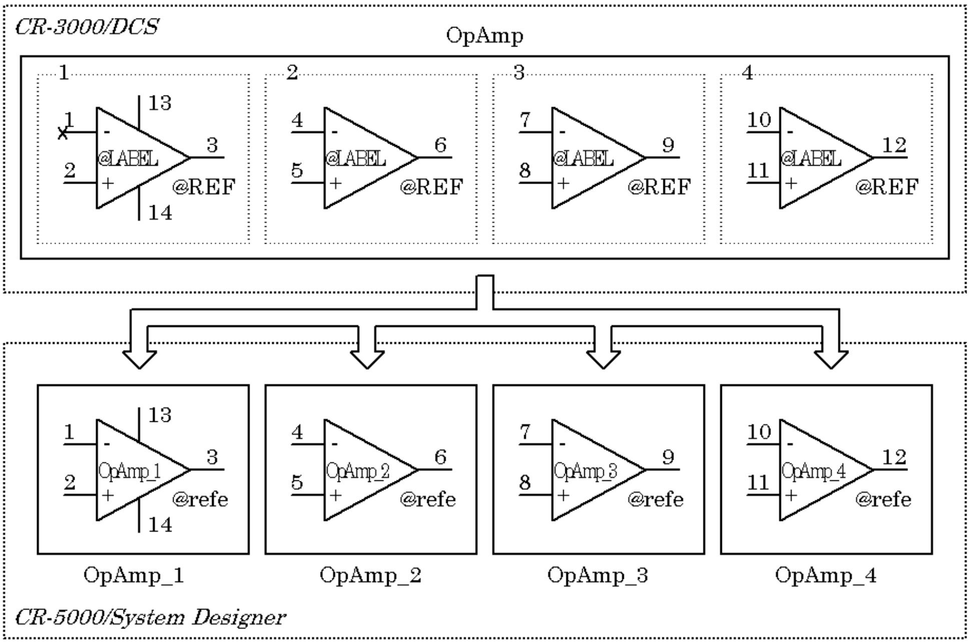


Figure 4.2 Example of Converting a Multi-component Symbol

4.2.3 Power Box Symbols

In DCS , you can define the power supply and ground as implied pins . In addition, you can insert a property viewer to each component with respect to the implied pins to set the power and ground values. (Multi-power design) In this case, power box symbol information is output from DCS to the SD transfer file based on the implied pin names.
Power box symbols in System Designer are generated in the following directory:

```
./DCS_SD/SCH/PBOXLIB
```

The power box symbol names in System Designer are assigned in the following format:

```
pbox<number of implied pins >
```

When power box symbols have the same number of implied pins with different names (in the same shape), the symbol names are assigned as follows:

```
pbox<number of implied pins >_<No.>
```

<No.> is incremented by 1, starting at 1.

Figure 4.3 below gives an example of converting "74L04" as a power box symbol.

The symbol name in System Designer is "pbox2" because it has two implied pins.

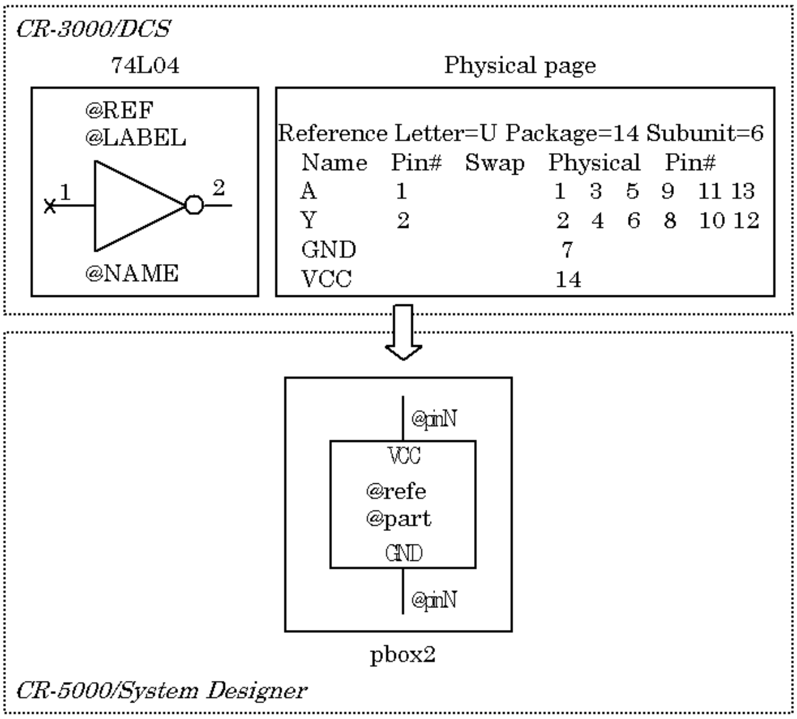


Figure 4.3 Example of Converting a Power Box Symbol

4.2.4 Form Symbols

Symbol information of forms is output from DCS to the SD transfer file only when the forms have been inserted on the circuit page .
Form symbols in System Designer are generated in the following directory:

```
./DCS_SD/SCH/FORM
```

The form symbol names in System Designer are assigned in the following format:

```
FORM<No.>
```

If the symbol name to be assigned to the symbol currently being generated has already been used for another symbol, the <No.> is incremented for the current symbol.

4.2.5 Connectors

Connectors in DCS are converted as symbols in System Designer .
Connector symbols in System Designer are generated in the following directory:

```
~/DCS_SD/SCH/CONNECTORS
```

The connector symbol names in System Designer are assigned as shown below:

```
cport0<@PinGraphics>
```

4.2.6 Unentered Bus Names

If no bus name has been entered for a bus line on the circuit, DCS assigns a unique net name in the "%NETxxxxx" format to the bus line. Since bus lines are recognized as general signal lines in System Designer , however, bus names are assigned automatically to unnamed bus lines and output to the SD transfer file .

DCS uses the following three methods to assign bus names.

(1) DCS assigns the bus name as a bundle of the net names of signal lines over the splitters connected to the bus line.

```
<net name>,<net name>,...<net name>
```

Figure 4.4 below gives an example of converting an unentered bus name.

In this example, the bus line has no bus name assigned. Wires with net names "Net_A" and "Net_C" have been connected to the splitters with position numbers 0 and 2, respectively. In addition, the wire with the net name "%NET00001" assigned by DCS has been connected to the splitter with position number 1. In this case, the bus name in System Designer is "Net_C,NET00001,Net_A" in the bundle format.

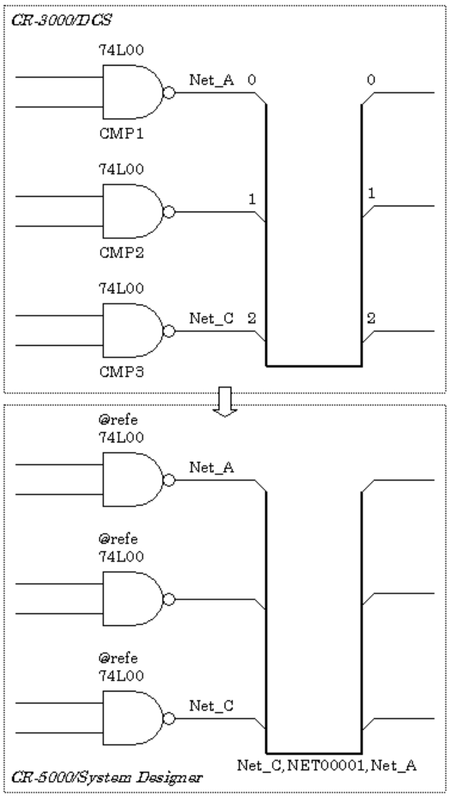


Figure 4.4 Example 1 of Converting an Unentered Bus Name

(2) If splitters have been connected "BY POSITION" to the bus line and no net name has been entered for the wires connected to the splitters, DCS assigns a range-specified bus name in the following format:

```
DcsSd<No.>[m:n]
```

<No.> represents the database identifier (@UID) unique to the bus line. Figure 4.5 below gives an example of converting an unentered bus name. In this example, no net name has been entered for any of the wires connected to the splitters with position numbers 0, 1, and 2. The bus name is "%NET00001" assigned by DCS . In this case, the bus name in System

Designer is "DcsSd1[2:0]" specifying the range of position numbers.

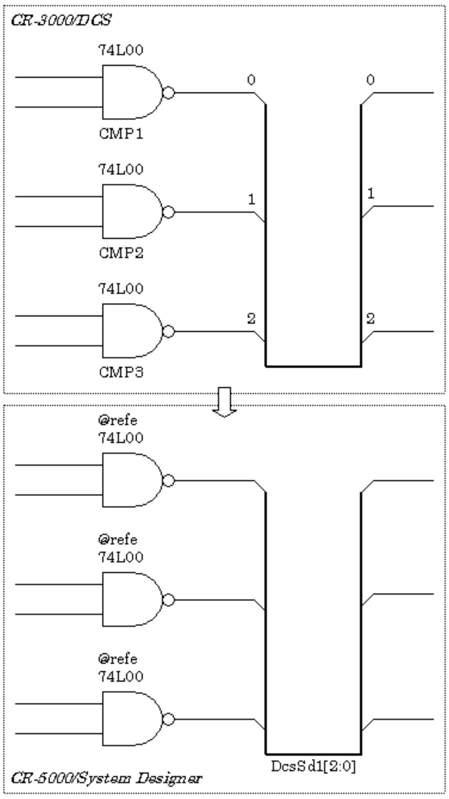


Figure 4.5 Example 2 of Converting an Unentered Bus Name

(3) If the bus line has been connected to the specified range of pins , DCS assigns the range-specified bus name in the following format:

```
DcsSd<No.>[m:n]
```

<No.> represents the database identifier (@UID) unique to the bus line. Figure 4.6 below gives an example of converting an unentered bus name.

In this example, no bus name has been assigned to the bus line and connected as "%NET00001" named by DCS to the specified range of pins . In this case, the bus name in System Designer is a range-specified bus name of DcsSd1[2:0]".

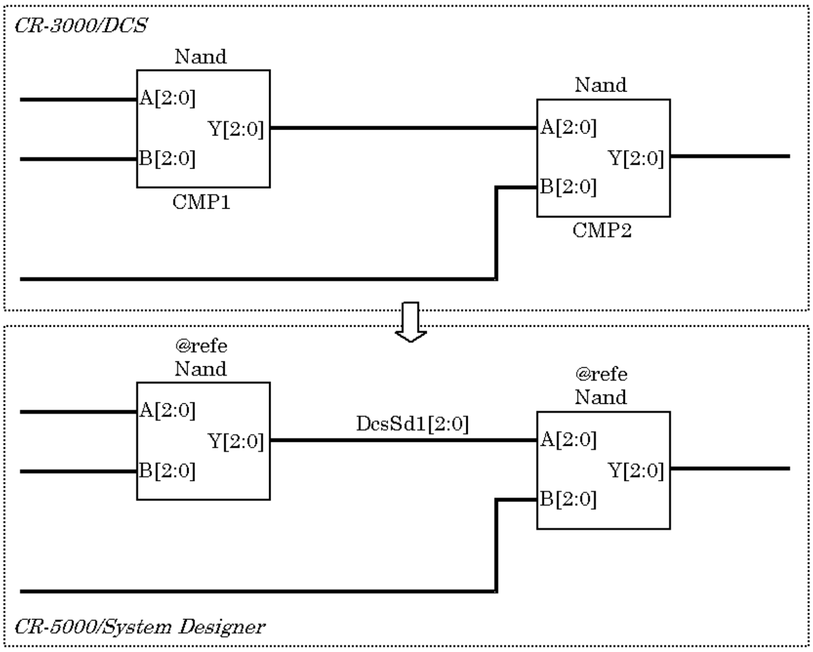


Figure 4.6 Example 3 of Converting an Unentered Bus Name

4.2.7 Unentered Pin Names

(1) How to Set up Turning Unentered Pin Labels

If no pin name has been entered for a pin on the symbol page , the pin name set in DCS is automatically assigned to that pin.

Pin names are assigned in the following format:

```
<input string>_<No.>
```

<input string> is the character string set in the step for "Setting a name for un-named pins" in Sections 2.1.3, 2.1.4, and 2.1.5. The default string is "noname".

<No.> is the logical pin number (@LogicalNumber) unique to the symbol pin .

The string set in the step for "Setting a name for unnamed pins" must be the same in all of the three sections above. If you set different names, the schematic and the components database for schematic design (LCDB) have the different pin names in System Designer . Note also that execution of "Reflecting LCDB Information in Schematic" in Section 2.2.3 causes the schematic to reflect the pin names in the components database for schematic design (LCDB) .

Figure 4.7 below gives an example of converting an unentered pin name .

If you specify the default pin name "noname" when "setting a name for unnamed pins" with a logical pin number of 2, the pin name in System Designer is "noname_2" as shown below.

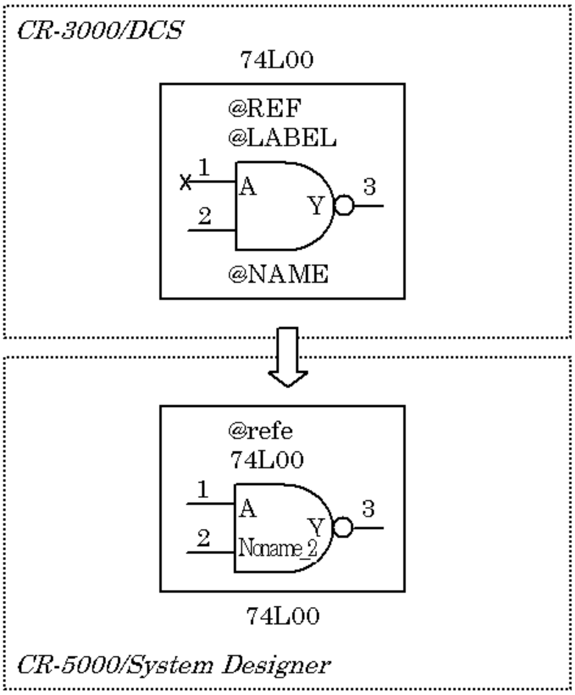


Figure 4.7 Example of Converting an Unentered Pin Name

(2) How to Set up Turning Unentered Pin Labels ON/OFF
When creating a SD transfer file, for unentered pin labels in.
DCS symbol page, it is possible to set up turning pin lables ON or OFF
In symbol sheets of System Designer.

This function can be realized by editing the following file using a text editor.

```
/usr/hp74200/lib/dcs2sd/etc/defName.txt
```

Format

One line consists of two fields.

To divide them, use "|" as a separator.

- First field

  Set up "PinDisplay".

- Second field

  Set up display status.

  The value is either "TRUE", "FALSE" or "NoChange".

Value	Contents
TRUE	Turns ON a pin label.
FALSE	Turns OFF a pin label.
NoChange	Keeps the display status in the symbol page. Label turned ON in the symbol page: ON Label turned OFF in the symbol page: OFF

e.g.) PinDisplay|FALSE (Turns an unentered pin label OFF.)

```
PIN   |   noname
```

-->

```
PIN          |   noname
PinDisplay   |   FALSE
```

Caution: Set the permission mode of this file to 666(rw-rw-rw-).

4.2.8 Bus Representation

Bus representation differs in DCS and System Designer . For this reason, descriptions that do not comply with range and bit specifications, and bundle bus representation, will be converted to bundle bus (decomposed) in System Designer to make it an valid description in System Designer. Representations subject to such conversion are as follows.

- A name that contains a character after "[]" is invalid. It will be converted to a bundle bus by converting "[ ]" to "(( ))".
- A range or bit specification that contains "( )" is invalid. It will be converted to a bundle bus.

- A range or bit specification that contains "{ }" is invalid. It will be converted to a bundle bus by deleting "{ }".
- A range or bit specification that contains "{{ }}" is invalid. It will be converted to a bundle bus by converting "{{ }}" to "{ }".

Examples of conversion are shown below.

DCS	System Designer
BUS[2:0]	BUS[2-0]
BUS[2:0]N	BUS((2))N,BUS((1))N,BUS((0))N
BUS(2:0)	BUS(2),BUS(1),BUS(0)
BUS{2:0}	BUS2,BUS1,BUS0
BUS{{2:0}}	BUS{2},BUS{1},BUS{0}

4.2.9 Turning Transfer Data tree structure ON/OFF

It can set up whether the directory composition of conversion data is constituted from a tree structure, or it constitutes from a flat.

This function can be realized by editing the following file using a text editor.

```
/usr/hp74200/lib/dcs2sd/etc/defName.txt
```

Format

One line consists of two fields. To divide them, use "|" as a separator.

- First field
  Set up "PATH"
- Second field
  Set up display status. The value is either "ON" or "OFF".

Value	Contents
No-setting line	Transfer Data is created as a tree structure
ON	Transfer Data is created as a tree structure
OFF	Transfer Data is NOT created as a tree structure

e.g.)PATH|OFF (Transfer Data is NOT created as a tree structure.)

```
PIN    |    noname
```

-->

```
PIN            |    noname
PATH           |    OFF
```

NOTE: Set the permission mode of this file to 666(rw-rw-rw-).

Example

For Example, in the composition of figure 4.8 DCS data, the partition icon "Sym1" and "Sym2" are in the file icon "LIB".

Following data is created when the symbols and the schematic sheets are stored in that partition icon.

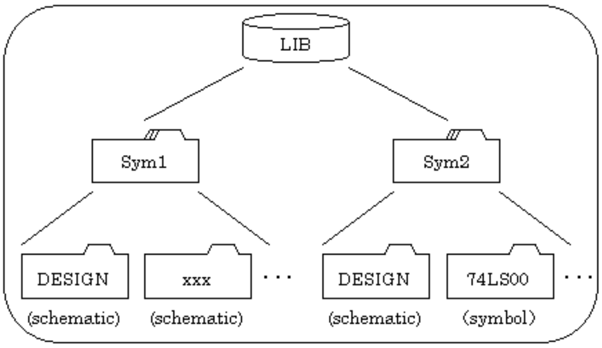


Figure 4.8 Configuration of DCS data

- PATH|OFF (It doesn’t create Transfer Data as a tree structure)

(1) SD Transfer File(It is data after conversion at System Designer)

The symbols and the schematic sheets in "Sym1" and "Sym2" are created collectively underneath "LIB" directory.

The following directory is created.

```
./LIB/74LS00.smb
./LIB/xxx.smb, xxx.cir, xxx.blk
./LIB/Sym1/DESIGN.smb, DESIGN.cir, DESIGN.blk
./LIB/Sym2/DESIGN.smb, DESIGN.cir, DESIGN.blk
```

Warning: When a duplicate design icon name is in DCS data to converted, a tree structure is created for each design as usual even if the setting "OFF". In this case, since "Sym1" and "Sym2" icon of same name "DESIGN" exist, as for these, data is created by the tree structure.

(2) LCDB

LCDB named "LIB", which has information of both "Sym1" and "Sym2", is created.

The following file is created.

```
./LIB.prf  (dbf , cmp)
```

- PATH|ON(Transfer Data is created as a tree structure)

(1) SD Transfer File(It is data after conversion at System Designer)

The directory "Sym1" and "Sym2" are created underneath the directory "LIB" and each directory has a symbol and a schematic sheet.

The following directory is created.

```
./LIB/Sym1/xxx.smb,  xxx.cir,  xxx.blk
./LIB/Sym1/DESIGN.smb,  DESIGN.cir,  DESIGN.blk
./LIB/Sym2/74LS00.smb
./LIB/Sym2/DESIGN.smb,  DESIGN.cir,  DESIGN.blk
```

(2) LCDB

LCDB files named "Sym1" and "Sym2" are created underneath the directory "LIB".

The following directory and file is created.

```
./LIB/Sym1.prf (dbf , cmp)
./LIB/Sym2.prf (dbf , cmp)
```

4.2.10 CR-5000/PWS restricted matters and the conversion in consideration of it

When operating CR-5000/PWS from System Designer, you can perform the conversion in consideration of PWS restricted (Pre-Check, LCDB Gen., SD Transfer File Gen.)

<DCS date to be checked>

Design icon name

Reference number

Pin name

Pin number(Alphabetic pin number)

Net name

<CR-5000/PWS restricted matters to be checked>

Alphabet small letter

Inhibited character(Space, ! " $ ’ ( ) , ; < > \ ‘ { | } #[1] *[1] /[1] =[1] ?[1]

length of a text string (Within 20 letters)

[1] is only for Alphabetic pin number

<Conversion rule for restricted matters>

Restricted matters	Support
Alphabet small letter	Converte into Alphabet capital letter
Inhibited character	Replace with optional characters[1]
When a text string has over 20 letter	Cut the characters that exceed 20 letters

[1] By setting it as Inhibited Character Conversion Table File (/usr/hp74200/lib/dcs2sd/etc/KeyList.txt), it is convertible for arbitrary characters. For details, see section 4.3.1 "Inhibited Character Conversion Table File".

4.3 Configuration Files

This section describes the contents of each configuration file.

4.3.1 Inhibited Character Conversion Table File

The Inhibited character conversion table file defines the characters inhibited in System Designer and the characters replacing them.

Inhibited characters are converted to their respective replacing characters according to the Inhibited character conversion table file. The following types of strings are subject to inhibited character conversion:

- File names (symbol, design names)
- Directory names (symbol, design path names)
- Pin names
- Property names
- Net names

In file and directory names, all inhibited characters defined in the Inhibited character conversion table are converted. In pin, property, and net names, only the characters listed below are converted.

	Inhibited characters
Pin name	Space " \
Property name	Space " \ # : ( ) { }
Net name	Space " \

The Inhibited character conversion table file is installed as the following file:

```
/usr/hp74200/lib/dcs2sd/etc/KeyList.txt
```

Format

The Inhibited character conversion table file is in the fixed format. Each line must consist of two fields containing a combination of an inhibited character and the replacing character(s), each enclosed in double quotes, separated by a tab.

- First field
  An inhibited character.
- Second field
  The character(s) to replace the inhibited character.

To change the default replacing character(s) to another character(s), use a text editor such as vi to edit only the second field. Do not edit the first field of any line.

```
Example:
###########################################################
##
##   Inhibited character conversion table file
##
###########################################################
              " "          "_32_"
              "!"          "_33_"
              """"         "_34_"
              "$"          "_36_"
              "&"          "_38_"
```

4.3.2 Property Conversion Table File

The Property conversion table file is generated with a file name of "propcnv.htb" by DCS Translator Sub-kit.

The Property conversion table file reflects the property names selected in (5) "Selecting properties" in Section 2.1.4. When the file already exists, property names are added to the file.

The Property conversion table file contains the "propertys" table.

Format

The "propertys" table is used for converting property names in the SD transfer file to the System Designer equivalents.

Each line corresponds to one property, consisting of three fields.

- First field
  Property name in the SD transfer file
- Second field
  Property name to be used in System Designer after conversion
  It is the property name registered in the Property specification file ($ZDSROOT/etc/jpn(eng)/PropSpec).
- Third field
  System Designer data viewer number. If the property has been display specified in the SD transfer file , the property viewer is generated based on the viewer number. If the number has already been used in the same object, a free number is selected.

The property which is selected during command execution on DCS is defined with the same name in the first and second fields.

To change the property name in System Designer , use a text editor such as vi to edit only the second field. Do not edit any other field.

Example:
```
    #-------------------------------------------------------------------
    #   property convert table sample file for CR-3000/DCS.
    #   Keyword     # Property Name     # pViewer ID
    #-------------------------------------------------------------------
    propertys {
      (DcsSdPartName partName       1)
      (Value                value           2)
      (Package            package         3)
}
```

Property names not defined in this table are converted to System Designer data as defined in DCS .

Note that property names unregistered in the Property specification file cannot be checked with the System Designer editor.

4.3.3 Function Type Definition File

The Function type definition file is generated by executing Data Converter or LCDB Generator in DCS .

When the Function type definition file already exists, property names are added to the file.

Format

The Function type definition file is in the fixed format. Each line corresponds to one function type, consisting of three fields.

- First field
  The type number of System Designer data. It is an integer greater than 100.
- Second field
  The type display string to be used in System Designer after conversion.
- Third field
  The DCS reference letter.

The Function type definition file cannot be localized. The system administrator must therefore reflect the file generated from DCS in the system master file by either copying or editing.

Compare the system master file with the "CompKind" file generated from DCS and, if only the added items are different, copy the generated file.

Otherwise, add the items different from the contents of the system master file to the system master file.

The system master file is installed as the following file:

```
$ZDSROOT/etc/CompKind
```

If the type number has already been used in the system master file, DCS selects a free number. To change the default type display character to another_character, use a text editor such as vi to edit only the second field.Do not edit any other field.

```
Example:
sNetNameHeader : SIGN   # netName Header
bNetNameHeader : B        # busName Header
#
# default reference header
#
dBlockRefheader    : X     # block
dSymbolRefheader : IC    # component
################################################################
# function table
################################################################
#
ComponentKind 3 {
#
#   System functions
#
#   Function No.    Display strings          reference header
    0                  "DEFAULT LOGIC"      IC
    1                  "JUMPER"                 J

    #############################################################
    # No.1 - No.100 System Functions; No.101 - No.256 User Functions
    #############################################################
    #
    #        Function No.    Display strings     reference header
    #  ex)       101            "User Parts"        U
                 101            "U"                 U
                 102            "C"                 C
}
```

If the corresponding reference letter has not been defined in the Function type definition file when this file is not reflected, the reference letter for type number "0" is set.

4.3.4 Property Specification File

The Property specification file is generated by executing Data Converter or LCDB Generator in DCS .

When the Property specification file already exists, property names are added to the file.

Format

The Property specification file is in the fixed format. Each line corresponds to one property, consisting of six fields.

- First field
  The property name to be used in System Designer after conversion.
- Second field
  The type, corresponding to "text" in DCS .
- Third field
  The display string to be used in System Designer after conversion.
- Fourth field
  The edit flag.
- Fifth field
  LCDB reference.
- Sixth field
  The viewer color number.

The Property specification file cannot be localized. The system administrator must therefore reflect the file generated from DCS in the system master file by either copying or editing.

Compare the system master file with the "PropSpec" file generated from DCS and, if only the added items are different, copy the generated file.

Otherwise, add the items different from the contents of the system master file to the system master file.

The system master file is installed as the following file:

```
    $ZDSROOT/etc/jpn/PropSpec
    $ZDSROOT/etc/eng/PropSpec
```

To change the default display string to another string, use a text editor such as vi to edit only the third field. Do not edit any other field.

```
Example:
###################################################################
#  System Property
###################################################################
$Sheet {
   ( author        text    "Author"         ON      -1)
                                  :
                                  :
}
$Component {
    ( partName      text    "Part Name"      ON      CIRC -1)
                                  :
                                  :
    ( Title            text    Title               ON      CIRC -1)
    ( Engineer       text    Engineer          ON      CIRC -1)
}
                                  :
                                  :
$cPin{
   ( pinLabel      text    "Pin Name"       ON      CIRC -1)
                                  :
                                  :
   ( isDeMorgan     text    isDeMorgan       ON      CIRC -1)
}
                                  :
                                  :
```

Note that property names unregistered in the Property specification file cannot be checked with the System Designer editor.

Chapter 5　OPERATION

This chapter provides examples of operation for converting DCS data to System Designer data.

5.1 Operation Example 1: Converting DCS Symbol Libraries

5.1.1 Process Flow

This example uses the following process flow to convert DCS symbol libraries.

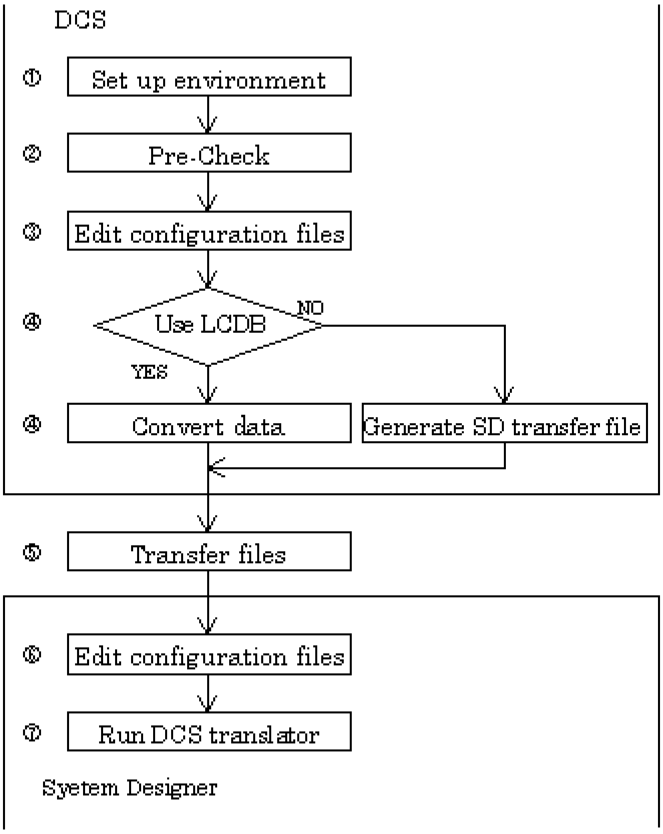


Figure 5.1 Process Flow for Converting Symbol Libraries

5.1.2 Procedure

(1) Setting up the environment
Mount the symbol libraires to be converted in the DCS screen environment.
Start operation for conversion from the screen or collector page.

(2) Pre-Checking
Execute Pre-Check according to Section 2.1.2 "Pre-Checker".

If any error or warning has been detected, check the content of the error or warning message file and follow the instruction for the corresponding message in Chapter 6 "MESSAGES" to correct the relevant item of data.

(3) Editing configuration files
If it is necessary to change an inhibited character conversion rule, edit the Inhibited character conversion table file. For details, see Section 4.3.1 "Inhibited Character Conversion Table File".

(4) Selecting whether to use the LCDB
Convert symbol libraries.
Select one of the following two methods of conversion, depending on whether the components database for schematic design (LCDB) is used in System Designer.

YES: Data Converter
For conversion using a components database for schematic design (LCDB) in System Designer , execute "DATA CONVERSION" according to Section 2.1.5 "Data Converter". The program generates the SD transfer file and the components database for schematic design (LCDB) from symbol library figures information and part property data .

NO : SD Transfer File Generator
For conversion without using the components database for schematic design (LCDB) in System Designer , execute "SD Transfer fileGen." according to Section 2.1.3 "SD Transfer File Generator".

The program generates the SD transfer file from symbol library figures information, without generating the components database for schematic design (LCDB) .

(5) Sending files
Send the files generated from DCS to the machine on which System Designer has been installed. (Sending files is not required in an environment with the disk shared.)
For details, see "File Transfer" in Section 2.2.1.

(6) Editing resource files
Set the resource files (propcnv.htb, CompKind, and PropSpec) for

System Designer if necessary. Note that you can perform neither editing nor checking in System Designer without these files set up correctly.
For details, see Section 4.3.2 "Property Conversion Table File" , Section 4.3.3 "Function Type Definition File", and Section 4.3.4 "Property Specification File".

(7) DCS Translator
Execute the DCS Translator program according to Section 2.2.2 "DCS Translator program".

5.2 Operation Example 2: Converting DCS Symbol Libraries and Schematic (while Generating LCDB)

The procedure for converting the DCS symbol libraries and schematic is the same as in operation example 1. Note, however, that since this example of operation also generates a components database for schematic design (LCDB) , the process flow is YES route of "Use LCDB?" which is described in Section 5.1.1 "Process Flow."

Pay attention also to the following point for conversion.

- If you perform conversion from the collector page in DCS, the components database for schematic design (LCDB) is generated for the symbol libraries existing under the execution environment. If no symbol used in the schematic exists in the collector page to be subject to conversion, the components database for schematic design (LCDB) is not generated for that symbol.

In that case, perform conversion in an environment containing the symbols used in the schematic.

5.3 Operation Example 3: Converting only a DCSSchematic and the Symbols Used in the Schematic (without Generating the LCDB)

5.3.1 Process Flow

This example uses the following process flow to convert a DCS schematic.

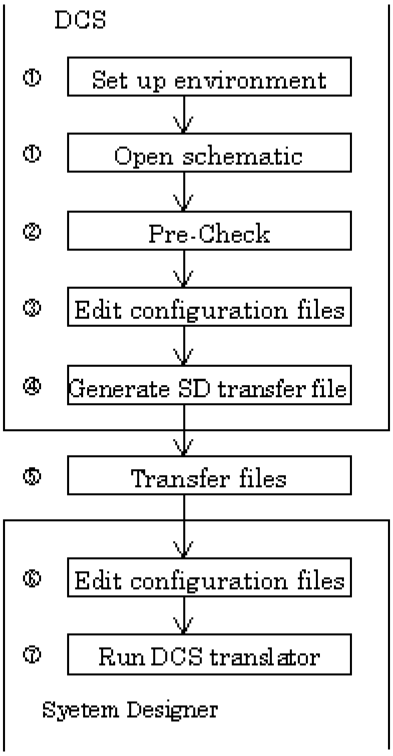


Figure 5.2 Process Flow for Converting a Schematic

5.3.2 Procedure

(1) Setting up the environment (opening the schematic)

Mount the schematic to be converted and the symbol libraries used for the schematic in the DCS screen environment. Start operation for conversion from the circuit page.

(2) Pre-Checking

Execute Pre-Check according to Section 2.1.2 "Pre-Checker".

If any error or warning has been detected, check the content of the error or warning message file and follow the instruction for the corresponding message in Chapter 6 "MESSAGES" to correct the relevant item of data.

(3) Editing configuration files

If it is necessary to change a inhibited character conversion rule, edit the Inhibited character conversion table file. For details, see Section 4.3.1 "Inhibited Character Conversion Table File".

(4) Generating a SD transfer file

Execute SD Transfer File Generator according to Section 2.1.3 "SD Transfer File Generator".

The program generates the SD transfer file from the schematic and symbol library figures information.

NOTE: For conversion using a components database for schematic design (LCDB) in System Designer , generate the LCDB in DCS in ad-vance. After schematic conversion in System Designer , the LCDB reload program is executed automatically. For details on the LCDB reload program, see Section 2.2.3 "Reflecting LCDB Information in Schematic."

(5) Sending files

Send the files generated from DCS to the machine on which System Designer has been installed. (Sending files is not required in an environment with the disk shared.)

For details, see "File Transfer" in Section 2.2.1.

(6) Editing resource files

Set the resource files (propcnv.htb, CompKind, and PropSpec) for System Designer . Note that you can perform neither editing nor checking in System Designer without these files set up correctly.

For details, see Section 4.3.2 "Property Conversion Table File", Section 4.3.3 "Function Type Definition File" , and Section 4.3.4 "Property

Specification File".
For conversion using that components library for schematic design (LCDB) in System Designer which is in a directory different from the directory containing the data to be converted, edit the data resource file "landata.rsc" to set "lcdbPathTable" to the path to the components database for schematic design (LCDB) in advance.

(7) Running DCS Translator
Execute the DCS Translator program according to Section 2.2.2 "DCS Translator program".

5.4 Restrictions on Operation

This product cannot be used for the following operations.

(1) Conversion of System Designer data to DCS data and execution of DDL and HILO in DCS

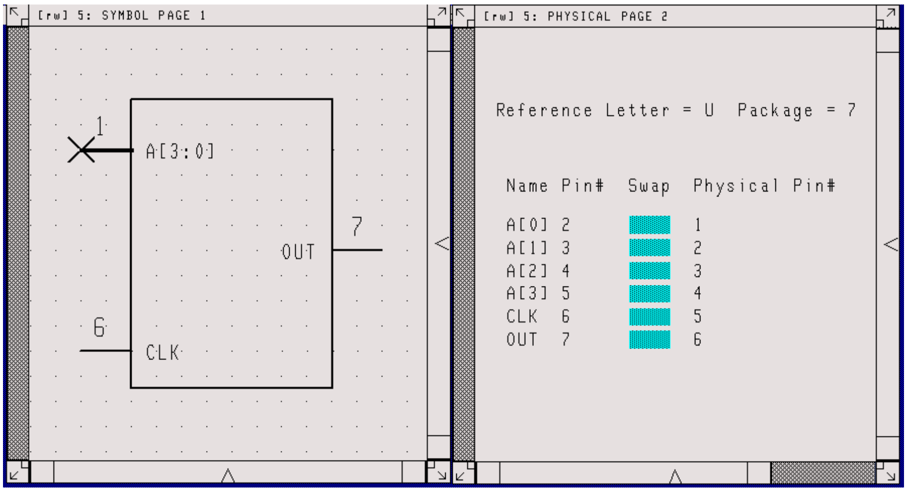
(2) Using System Designer Rev. 3.0 to convert the transfer file generated by DCS Translator Sub-kit in DCS Rev. 6.1 or earlier

(3) Updating the schematic data converted by DCS Translator in System Designer Rev. 2.x or earlier to reflect the components database for schematic design (LCDB) generated by this product. This attempt results in inconsistency of data because this product checks inhibited characters in symbol,pin, and property names.

(4) Updating the schematic data converted by this product to reflect the components database for schematic design (LCDB) generated in DCS Rev. 6.1 or earlier or in System Designer Rev. 2.x or earlier. This attempt also results in inconsistency of data because this product checks inhibited characters in symbol, pin, and property names.

(5) Restrictions on Using Symbols with Bus Pins and Operating LCDB First, examples demonstrating how to convert the shape and part data (LCDB) of a DCS symbol with a bus pin using DCS Translator Subkit will be shown.

Here are examples of a symbol page and physical page of DCS.



Reference Letter = U Package = 7

Name	Pin#	Swap	Physical Pin#
A[0]	2	▨	1
A[1]	3	▨	2
A[2]	4	▨	3
A[3]	5	▨	4
CLK	6	▨	5
OUT	7	▨	6

Figure 5.3 Symbol Page,Physical Page

The numbers on the pins in the symbol page are logical pin numbers.

In the physical page, the pin information is defined only for the implied pins A[3], A[2], A[1] and A[0], but not for the bus pin A[3:0].

The symbol sheet and the component file of SD created by converting the symbol above using DCS Translator are as follows:

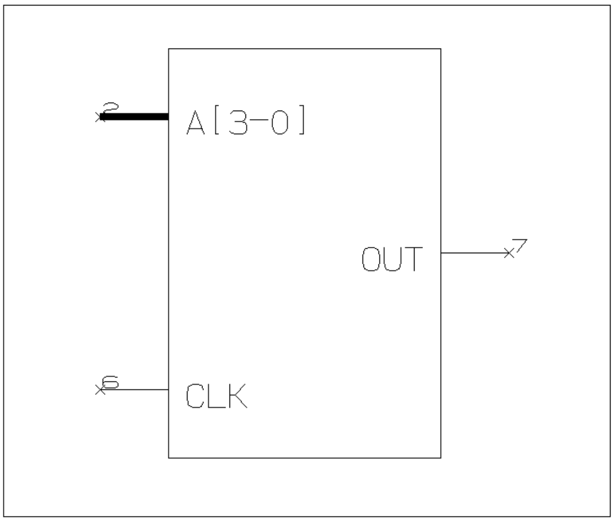
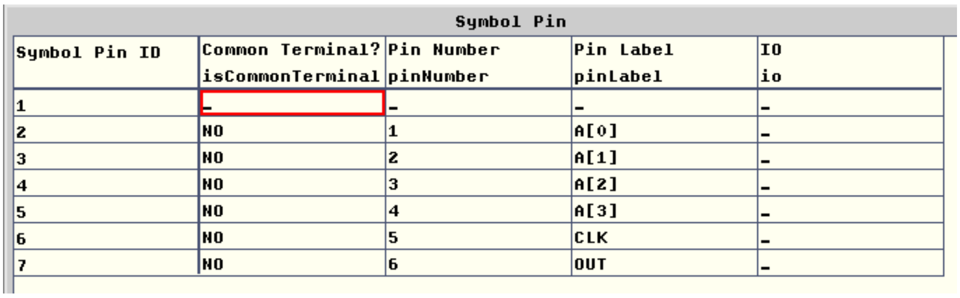


Figure 5.4 Symbol Sheet



Symbol Pin				
Symbol Pin ID	Common Terminal? isCommonTerminal	Pin Number pinNumber	Pin Label pinLabel	IO io
1	_	-	-	-
2	NO	1	A[0]	-
3	NO	2	A[1]	-
4	NO	3	A[2]	-
5	NO	4	A[3]	-
6	NO	5	CLK	-
7	NO	6	OUT	-

Figure 5.5 LCDB

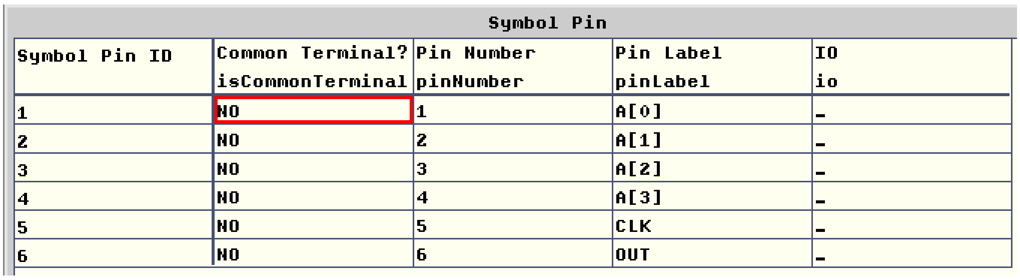
The logical pin number of the bus pin A[3:0] starts from 1 in DCS, however, in System Designer, the symbol pin ID starts from 2. This is because symbol pin IDs are based on logical pin numbers in a DCS physical page. (The numbers on the pins in the symbol sheet are the symbol pin IDs.)

The following section will explain the restrictions of bus pins based on the case above.

The Restrictions of Generating LCDB without DCS Translator Sub-kit
When generating LCDB using other than DCS Translator Sub-kit and converting schematics and symbol shapes using DCS Translator Sub-kit, becareful about the following.

If schematics and/or symbol shapes are converted by DCS Translator Sub-kit, the results will be like the one shown in Figure 5.4 To generate LCDB,set the pin numbers as shown in Figure 5.5: Component File.

Even though the logical pin number of the bus pin A[3:0] in DCS starts from 1, do not set the symbol pin IDs of the component file like an example shown in Figure 5.6. If you design a schematic using the symbol and reload this wrong LCDB, the pin numbers and labels will be converted incorrectly as shown in Figure 5.7.



Symbol Pin				
Symbol Pin ID	Common Terminal? isCommonTerminal	Pin Number pinNumber	Pin Label pinLabel	IO io
1	NO	1	A[0]	-
2	NO	2	A[1]	-
3	NO	3	A[2]	-
4	NO	4	A[3]	-
5	NO	5	CLK	-
6	NO	6	OUT	-

Figure 5.6 Incorrect Component File

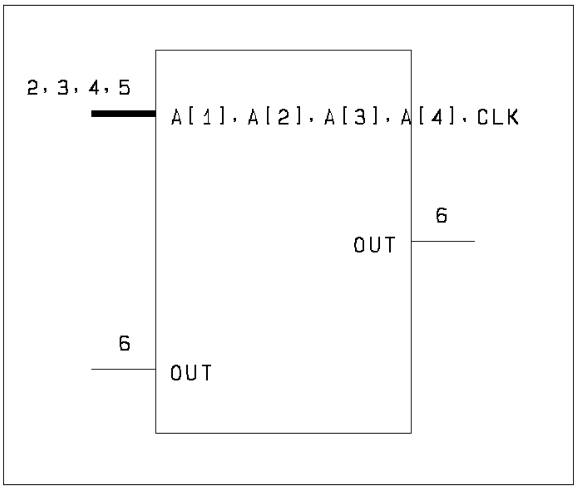
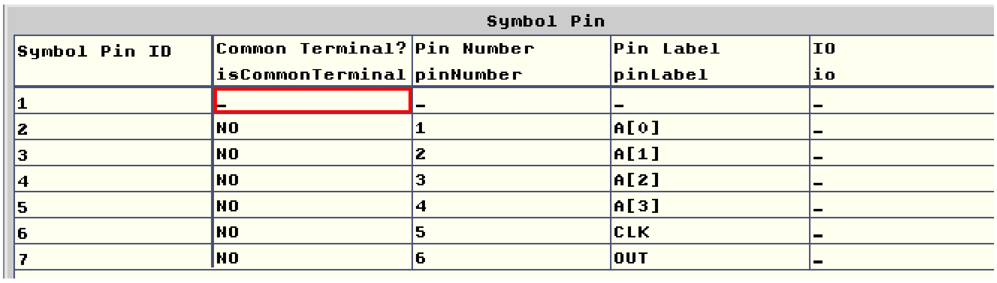


Figure 5.7 Incorrect Schematic Sheet after Reloaded

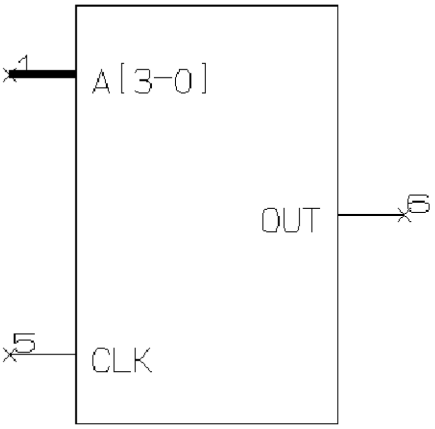
Restrictions of Generating LCDB by DCS Translator Sub-kit When generating LCDB using DCS Translator Sub-kit and creating symbol shapes using System Designer for the first time, be careful of the following.
If converting component properties using DCS Translator Sub-kit, the results will be as shown in the component file in Figure 5.5. When editing schematics and/ or symbol shapes using System Designer, be careful not to start the symbol pin IDs of bus pins from 1; See Figure 5.4: Symbol Sheet.

If starting symbol pin IDs from 1 as shown in Figure 5.9 and reloading LCDB of the component file in Figure 5.8, the pin numbers and labels are converted incorrectly as shown in Figure 5.10.



Symbol Pin				
Symbol Pin ID	Common Terminal? isCommonTerminal	Pin Number pinNumber	Pin Label pinLabel	IO io
1	-	-	-	-
2	NO	1	A[0]	-
3	NO	2	A[1]	-
4	NO	3	A[2]	-
5	NO	4	A[3]	-
6	NO	5	CLK	-
7	NO	6	OUT	-

Figure 5.8 Correct Component File



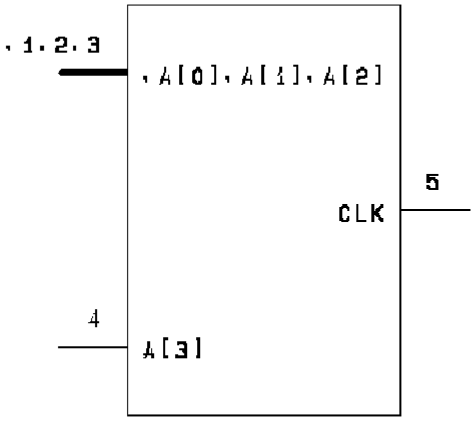


Figure 5.9 Incorrect Symbol SheetFigure

Figure 5.10 Incorrect Schematic Sheet after Reloading LCDB

Chapter 6 MESSAGES

This chapter describes the messages output by DCS Translator Sub-kit.

6.1 Messages

The product outputs error, warning , and log messages. This section lists the error and warning messages to be output during command execution. Note that the messages output by DCS itself are not included.

Each message begins with a mark and number followed by the text, classified as shown below.

Mark	Error number	Warning number	Module
C	1 - 99	100 - 2999	Common
P			Pre-Checker
M,B,D,N			SD Transfer File Generator
L,B,N			LCDB Generator
All above			Data Converter

Messages from each command are output to the following files:

Command	Error	Warning	Message
Data Converter	DataConv.err	DataConv.wrn	DataConv.log
Pre-Checker	PreCheck.err	PreCheck.wrn	PreCheck.log
SD Transfer File Generator	SDFilegen.err	SDFilegen.wrn	SDFilegen.log
LCDB Generator	LcdbGen.err	LcdbGen.wrn	LcdbGen.log

6.2 List of Error Messages

6.2.1 Number C Messages

Number	Message	Meaning/Corrective Action, Countermeasure
C1	No available license.	Security file is not normal.Check the permit code is set correctly, and contact Zuken Inc.
C2	Cannot alloate enough memory.	No enough memory. Quit dispensable process or increase swap area.If that is not enough, divide data and re-execute.
C3	Cannot make directory<XXX >.	Permission of directory under which designated directory is to be made is irrelevant. Change the permission.
C4	Cannot make file <XXX >.	Permission of directory under which file is to be made is irrelevant. Either change the permission or contact Zuken Inc.after checking phenomenon.
C5	Cannot open file <XXX >.	Either no file or permission is irrele-vant. Either make file or change file’s permission.
C6	Cannot delete file <XXX >.	Permission of directory which has the file is irrelevant. Change the permission.
C7	No permission to read status file.	No permission to read status file.Delete "./ .DcsSdLOCK" and re-execute.
C8	No permission to write status file.	No permission to write status file.Delete "./ .DcsSdLOCK" and re-execute.
C9	Cannot make temporary file <XXX >.	Permission of directory under which file is to be made is irrelevant. Change the permission.
C10	Cannot open temporary file <XXX >.	Either no file or permission is irrelevant. Either make file or change the permission.
C11	Cannot delete temporary file <XXX >.	Permission of directory which hasthe file is irrelevant. Change the permission.

Number	Message	Meaning/Corrective Action, Countermeasure
C12	Format mistakes in the Nth line in prohibited text conversion table file.	Format of prohibited text conversion table file is the following: [tab]"prohibited text"[tab]"conversiontext" Change designated line.
C13	The Nth line in prohibited text conversion table file has prohibited text ’XXX ’.	Prohibited text is set to conversion text setting in prohibited text conversion table file. Change designated line.
C14	Icon name <XXX > exists N.	More than one icon has the same name. Change the name not to duplicate.
C15	Symbol <XXX> on schematic doesn’t exist on screen environment. [YYY]	Insert symbol in on screen environment.
C16	Failed to convert file <XXX> code.	File text code.may be irrelevant. Contact Zuken Inc.
C17	Cannot make directory <XXX > because file(s) exist.	Cannot make directory because file with the same name exists. Move the file.
C18	No permission to read directory <XXX>.	No permission to read directory. Change the permission to read the directory.
C19	No permission to write directory <XXX >	No permission to write directory. Change the permission to write the directory.
C20	No permission to execute directory <XXX>	No permission to execute directory. Change the permission to execute the directory.
C21	Program <XXX > is terminated irregularly.	Check message file and program name that terminated irregularly, and contact Zuken Inc.

6.2.2 Number B Messages

Number	Message	Meaning/Corrective Action, Countermeasure
B20	Error in <XXX > format.	Error in Power/Ground setting file <XXX> format. This error won't happen unless modify file <XXX > by hand.
B21	Cannot output to file <XXX>.	Cannot output to Power/Ground setting file <XXX >. Check the file permission to write.

6.2.3 Number D Messages

Number	Message	Meaning/Corrective Action, Countermeasure
D20	Since cannot read resource file <XXX >, <YYY > is not made.	Cannot make <./DCS SD/SCH/ dcslandata.rsc> because cannot read resource file </usr/hp74200/lib/dcs2sd/etc/landata.txt>. This error won't happen if installation is normally done. Re-install the program.
D21	Error in resource file format.	Error in the resource file format.If $ZD-SROOT is properly set, check if format of $ZDSROOT/info/landata.rsc is correct. This error won't happen if $ZDSROOT is not set. Re-install the program.

6.2.4 Number L Messages

Number	Message	Meaning/Corrective Action, Countermeasure
L21	No permission to write power/ground setting file <XXX>.	Check if file has permission to write.
L22	No permission to read power/ ground setting file <XXX>.	Check if file has permission to read.
L25	No permission to write file <XXX >.	Check if file has permission to write.
L41	Symbol pin <XXX > exists outside symbol partition.(YYY)	Place ’symbol pin <XXX >of symbol set in multi-component’ inside symbol partition.
L80	Cannot recognize format of <XXX > correctly. No change will be made.	Check format of file <XXX>.

6.3 List of Warning Messages

6.3.1 Number C Messages

Number	Message	Meaning/Corrective Action, Countermeasure
C100	Added prohibited text "XXX" to prohibited text conversion table file because it is not defined.	No definition of prohibited text and conversion text for the prohibited text in prohibited text conversion table.This message won’t be issued unless deleting def inition line in prohibited text conversion table file by hand.
C101	Convert icon name <XXX >into <YYY>.	Will treat icon name <XXX >as <YYY>.
C103	Convert pin name <XXX>into <YYY>. [ZZZ]	Pin name has prohibited text. Will treat pin name <XXX > as <YYY >.
C104	Convert net name <XXX>into<YYY>. [ZZZ]	Net name has prohibited text. Will treat net name <XXX> as <YYY>.
C105	Convert property name <XXX> into <YYY >. [ZZZ]	Property name has prohibited text.Will treat property name <XXX> as <YYY>.
C106	Convert implied pin name <XXX> into <YYY>. [ZZZ]	Implied pin name has prohibited text.Will treat implied pin name <XXX >as <YYY >.
C107	Convert part name <XXX> into <YYY>. [ZZZ]	Parts name has prohibited text. Will treat parts name <XXX > as <YYY >.
C200	Pin <XXX> is an overlapping pin name.[ZZZ]	When using PWS, overlap of pin names is not allowed. Change the pin name so that there is not overlap.

Number	Message	Meaning/Corrective Action, Countermeasure
C201	A partName consists of more than 20 characters. The part Name<XXX> will be set to <YYY>.[ZZZ].	When using PWS, a partName should not consist of more than 20 characters. The partName is changed to consist of 20 characters.
C202	A pin name that consistsof only numbers is not allowed. Pin<XXX> will be set to <LPXXX>.[ZZZ]	When using PWS, a pin name that consists of only numbers is not allowed. The pin name will be changed to <LPXXX>.
C203	Pin<XXX> consists of more than 20 characters. Pin<XXX> will be set to <YYY>.[ZZZ].	When using PWS, a pin name should not consist of more than 20 characters. The pin name is changed to consist of 20 characters.
C204	Node<XXX> consists of more than 20 characters. Node<XXX> will be set to <YYY>.[ZZZ]	When using PWS, a node name shuld not consist of more than 20 characters. The node name is changed to consist of 20 characters.

6.3.2 Number B Messages

Number	Message	Meaning/Corrective Action, Countermeasure
B100	Cannot recognize type of implied pin <XXX>. #OtherType will be set.	Considers type of implied pin <XXX> #OtherType and continues process because cannot recognize the type. This cannot be happened. Re-execute the program.
B101	Implied pin <XXX> has text that cannot be used for power or ground. #OtherType will be set.	Implied pin <XXX> name includes text that does not allow you to set it as either Power or Ground, so that will treat it as #OtherType and continue process. To treat it as Power or Ground, change the implied pin name and re-execute.
B102	Power box to be generated may be inconsistent because implied pin type is different from it was. Lists the different implied pin names.(Before -> Now) XXX	Implied pin type is different from it was,so that generated LCDB and power box are different from they were.Will list the differences. Make change as little as possible. <Previous Type-> Present Type:"Pin Name", ... "Pin Name">
B103	Implied pin with physical pin number <XXX > is ignored because it has no name.<YYY>	Implied pin of parts <YYY > has no name but has physical pin number <XXX>. Will ignore the implied pin and continue process. This warning won’t be issued if deleting physical pin number or naming pin.

6.3.3 Number D Messages

Number	Message	Meaning/Corrective Action, Countermeasure
D100	Will use default because can not read resource file<XXX>.	Cannot read resource file <$ZDSROOT/info/landata.rsc>, so that will use /usr/hp74200/lib/dcs2sd/etc/landata.rsc and continue process. Check if $ZDSROOT is set correctly and resource file <$ZDSROOT/info/landata.rsc> has permission to read.

6.3.4 Number L Messages

Number	Message	Meaning/Corrective Action, Countermeasure
L600	Failed to initialize LCDB user definition file. (XXX)	Setting(s) of property(s) were not cleared.
L800	Cannot recognize type of pin.<XXX> Will set <YYY>,(ZZZ)	Check @Type of pin. Rules of symbol pin type are as follows: (1)INPUT:"I" (2)OUTPUT:"O","C","E" (3)VCC:Implied pin set to "Power" (4)GND:Implied pin set to "General Ground". (5)NC:Implied pin set to "NC" (6)BIDIRECT:Others
L810	Couldn’t set pin <XXX>name with reference to pack-age. (YYY)	Not reflected in pin name in parts of ".cmp despite 2 or more @Subunits or multi component symbol.
L850	Can display property value of <XXX> up to N	Property value(s) of page <XXX> can be displayed up to N .
L900	Cannot recognize power name. Judges pin name <VCC> power.	Specify "Power" for name of implied pin you desire to set to Power.

Number	Message	Meaning/Corrective Action, Countermeasure
L901	Cannot recognize ground name. Judges pin name <GND> ground.	Specify "GeneralGround" for name of im-plied pin you desire to set to ground.
L1001	No symbol pin. (XXX)	If it is an object of LCDB output, insert symbol pin. If not, delete symbol page.
L1100	Value of <XXX> of scion pages not set. (YYY)	Load page property <XXX >mult can not be reflected correctly.Set property <XXX>.
L1200	Pin <XXX>'s N th package's physical number is not set. Will set <YYY>. (ZZZ)	Pin <XXX>'s N th package physical number is not set. Will set <YYY> to it. <ZZZ>
L1201	Physical number of implied pin <XXX>is not set. Will set <YYY>. (ZZZ)	Set physical number of corresponding pin.
L1210	Cannot recognize swap text string of pin <XXX>. The swap text should be up to 3 letters. (YYY)	Re-set swap text within 3 letters.
L1500	The M th value <N> of property PWS_PCMAC_NO in-cludes text other than nu-meric figure. (XXX)	Specify scion page property PWS_PC_MAC_NO in numeric figures. (Separator is ":".)
L1501	The M th value <N > of prop-erty PWS_PCMAC_NO is out of range. The value range is from 1 to 12288. (XXX)	Specify scion page property PWS_PC_MAC_NO in numeric figures ranging from1 to 12288. (Separator is ":".)
L2010	There is setting(s) that Key-word is <DcsSdPartName>but Property_Name is not <partName>. (XXX)	Should set property name DcsSdPart-Name to partName.

Number	Message	Meaning/Corrective Action, Countermeasure
L2011	There is setting(s) that Prop-erty_Name is <partName> but Keyword is not <DcsSd-PartName>. (XXX)	Should set property name partName to DcsSdPartName.
L2021	No <DcsSdPartName> in Keyword. (XXX)	Property name, DcsSdPartName, is in-dispensable. Added automatically if not exist.
L2022	More than one <DcsSdPart-Name> in Keyword. (XXX)	Set just one DcsSdPartName.
L2030	Cannot set fileChar{ }(XXX)	fileChar{ } setting exists. Do not set fileChar { }.
L2300	Cannot set <XXX > to Comp-Kind. Can newly set up to 155	The number of @Reference Letter set to CompKind is beyond the limit.
L2330	FunctionNo <N> is out range. (XXX)	FofunctionNo of UserFunctions that is newly registered is over 256. Delete User Functions in existing CompKind or reduce FunctionNo of UserFunction.
L2340	The number of UserFunctions is already maximum. (XXX)	Displays existing CompKind in the follow-ing cases. (1)When the number of reference header text is over 256. (2)When FunctionNo is over 256. Change file XXX not to meet either of above-mentioned cases.
L2400	Couldn’t make <XXX>.	Couldn’t make file XXX because warning,etc. was issued while making it.
L2410	..No <XXX >. (YYY)	File YYY does not have XXX . Check format of file YYY .

6.3.5 Number M Messages

Number	Message	Meaning/Corrective Action, Countermeasure
M511	Relational icon <XXX > will adopt master icon <YYY> figure. (ZZZ)	Relational symbol <XXX> used in schematic uses master symbol <YYY>'s figure.

6.3.6 Number P Messages

Number	Message	Meaning/Corrective Action, Countermeasure
P100	Form icon exists. [XXX]	Converts form(s) only in circuit page.To convert form, insert form you desire to convert in circuit page.
P101	Document icon is not converted. [XXX]	Does not convert document icon.
P121	N painted polygon(s) exist.[XXX]	If polygons have crossing lines,the filling position of each polygon may not be converted normally in System Designer .
P124	Local symbol <XXX > exists.[YYY]	Does not convert local symbol. When converting local symbol as symbol, save the local symbol.
P125	Symbol <XXX> has pin(s) with no name.	Symbol <XXX> has some pin(s) with no name. Default name will be assigned if executing conversion without adding pin name.
P127	A pin name that consists of only numbers is not allowed.	When using PWS, a pin name that consists of only numbers is not allowed. Please change a pin name <XXX>.
P128	Pin<XXX> is an overlapping pin name. [ZZZ]	When using PWS, overlap of pin names is not allowed. Please change so that a pin name dose not overlap.

Number	Message	Meaning/Corrective Action, Countermeasure
P129	Pin<XXX> consists of more than 20 characters.	When using PWS, a pin name shuld not consist of more than 20 characters. Please change so that a pin name becomes 20 characters.
P130	Node<XXX> consists of more than 20 characters.	When using PWS, a node name shuld not consist of more than 20 characters. Please change so that a node name becomes 20 characters.
P150	No bus type net name [XXX].[YYY] Will consider it singlewire.	Bus type wire has no name. Cannot recognize bit width, so that after conversion, it'll be single wire. To convert as bus type wire, add net name.
P151	No bus type net name [XXX].	Bus type wire has no name. Although default name is assigned when converting, add net name if necessary.
P160	Alias <XXX > of power/ground is not set. [YYY]	No alias of Power/Ground on circuit page.If it is multiple Power circuit, add alias(es) of Power/Ground. If not, delete property.
P161	Power/Ground name <XXX> of a reference <RRR> are changed to <YYY>. [ZZZ]	Power/ground <XXX> on circuit page have changed to <YYY>. When inserting power box into converted schematic in System Designer , connect it with <YYY>.
P162	Cannot set implied pin <XXX> as power or ground. [YYY]	Implied pin <XXX > name in cludes text that does not allow you to set it as either Power or Ground. To treat it as Power or Ground, convert the implied pin name and re-execute.
P163	Pin(s) with no physical pin number exist(s). [XXX]	Symbol has pin(s) with no physical pin number(s). Translator automatically assigns physical pin numbers.
P164	No pin name for physical pin number <XXX >. [YYY]	No pin name for physical pin number. This implied pin will be ignored.

Number	Message	Meaning/Corrective Action, Countermeasure
P170	Title page property ’XXX’ is duplicate. [YYY] The secondone and after are ignored.	Title page property is duplicate.The second one and after are ignored.Change their duplicative property(s) in title page.
P171	Scion page property ’XXX’ is ignored. [YYY]	The property ’XXX ’ in title and scion page is duplicate. The property as one on scion page is ignored. Change the property name.

6.3.7 Number N Messages

Number	Message	Meaning/Corrective Action, Countermeasure
N101	Will consider bus type wire <XXX> single wire. [YYY]	Wire has no name. Cannot recognize bit width, so that will consider it single wire.
N102	Will change name of bus type wire <XXX> to <YYY>.[ZZZ]	Wire has no name. Will set default net name <YYY>.
N150	Data to be made may be in-consistent because pin name is different from it was. (Before -> Now) XXX -> YYY	Generated symbol, LCDB and circuit data are different from they were because default name setting of symbol pin is different from it was. Check the differences. Make change as little as possible.