



*EDIF200 Schematic Converter Module Writer*

# Contents

Chapter 1	EDIF Writer	1-1
1.1	Basic Operations of the Data ConverterMenus	1-2
1.1.1	Starting Data Converter	1-2
1.1.2	Exiting the Tool	1-4
1.1.3	Selecting the Tool	1-4
1.1.4	Changing the Startup Parameters	1-4
1.1.5	File Opener	1-5
1.1.6	Executing the Tool	1-6
1.1.7	Confirming Error and Warning Messages	1-6
1.2	Setting the Parameters	1-7
1.3	Property conversion file	1-9
1.4	Setting file by Vendor	1-10
1.4.1	Convert to DCS	1-10
1.4.2	Conversion from VIEW	1-10
1.4.3	Conversion from MENTOR	1-10
1.5	Restrictions	1-11
Chapter 2	Launch the Shell (Batch)	2-1
2.1	Format of Startup Options	2-2
2.2	The launch options used for the EDIF 200 Converter Module Writer	2-3
Appendix A	Schematic View Output Specifications	A-1
A.1	EDIF200 Overview	A-2
A.1.1	Overview of the EDIF Schematic View File	A-2
A.1.2	Special Properties	A-3
A.1.3	Hierarchical Structure	A-3

A.2	EDIF Schematic View Syntax	A-4
A.2.1	edif	A-4
A.2.2	Status	A-5
A.2.3	Libraries	A-5
A.2.4	Scale	A-6
A.2.5	Property Group	A-6
A.3	Description of Symbol	A-8
A.3.1	Definition of External Pins	A-9
A.3.2	Symbol Sheet Size	A-10
A.3.3	Symbol Property Viewer	A-10
A.3.4	Symbol Pins	A-11
A.3.5	Symbol Figures	A-12
A.3.6	Symbol Comment text	A-13
A.3.7	Pin Arrays	A-13
A.3.8	Displaying Properties	A-14
A.4	Connector Symbols	A-15
A.4.1	Ripper Symbol	A-16
A.5	Schematic Sheet	A-17
A.5.1	Schematic Sheet Size	A-17
A.5.2	Instance Symbol	A-18
A.5.3	Instance Symbol Pin	A-19
A.5.4	Instance Symbol Reference	A-19
A.5.5	Instance Symbol Property	A-19
A.5.6	Instance Symbol Pin Number	A-20
A.5.7	Instance Symbol Pin Property	A-20
A.6	Connectors	A-21
A.6.1	Hierarchy Connector	A-22
A.6.2	Sheet Connector	A-22
A.6.3	Power Supply and Ground	A-23
A.7	Nets	A-24
A.7.1	Bus	A-25
A.7.2	Figures on the Schematic Sheet	A-26
A.8	DCS Mode Specific Specification	A-28
A.8.1	Handling Connector Symbols	A-28
A.8.2	Reference	A-29

---

## Chapter 1 EDIF Writer

---

EDIF200 Converter Module Writer is a batch program.

There are two ways to launch this module: use the Data Converter or execute from the Shell .

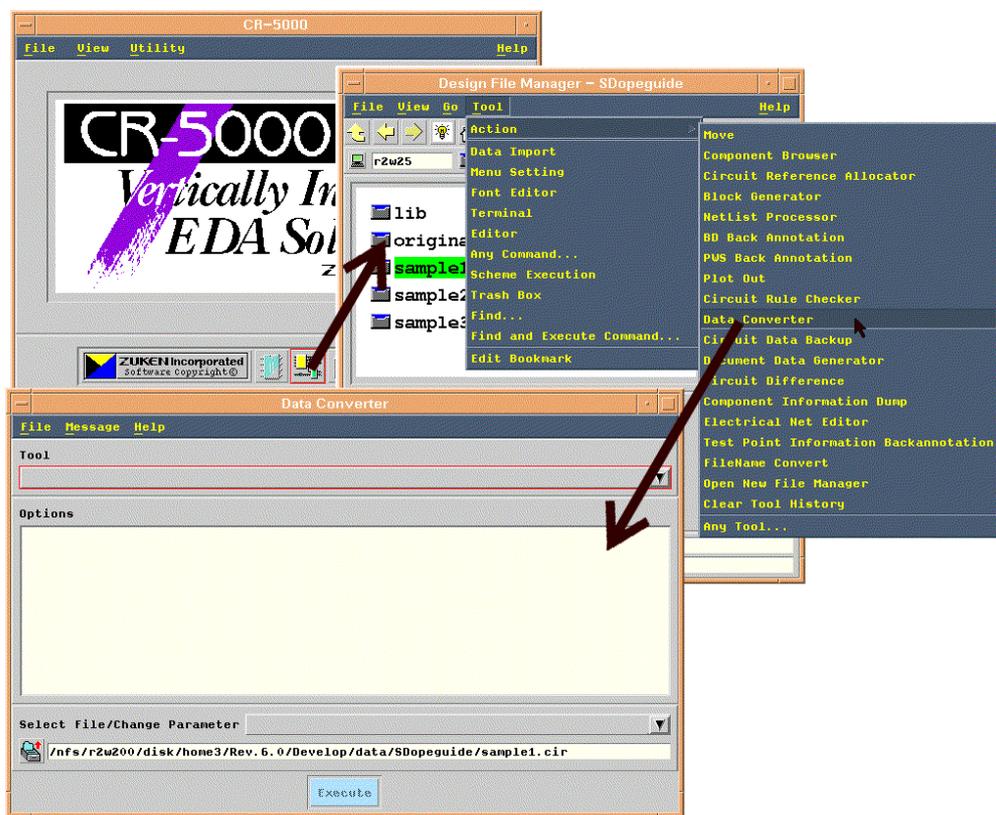
This manual explains the basic operations of the Data Converter and how to set various parameters.

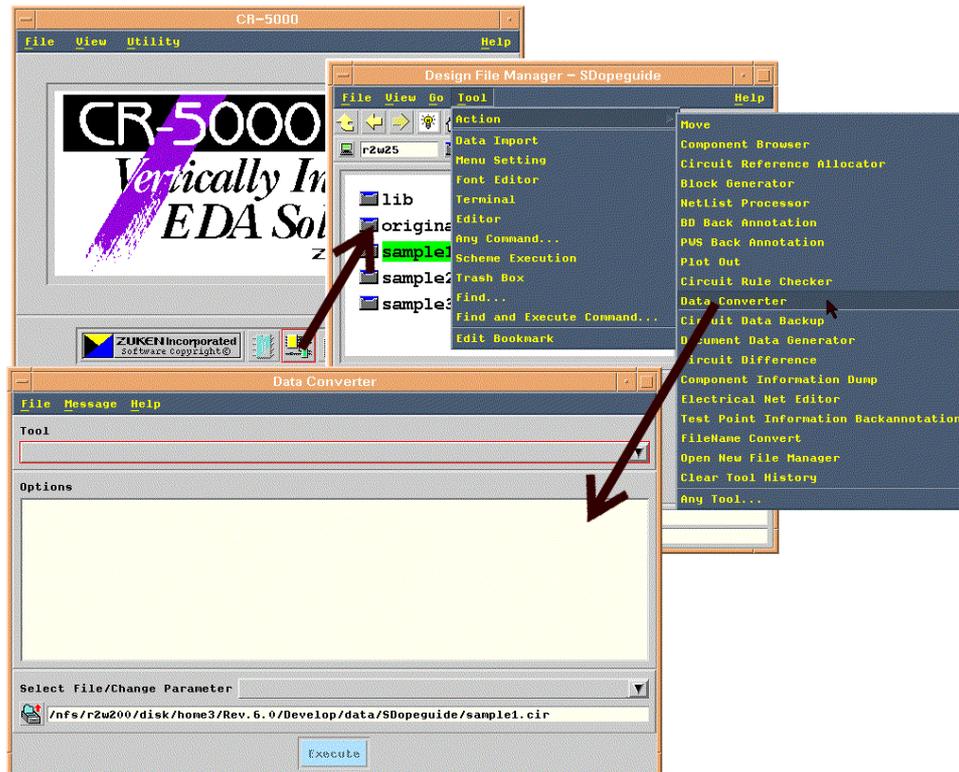
# 1.1 Basic Operations of the Data ConverterMenus

## 1.1.1 Starting Data Converter

Invoke the Translator menu from the cr5000 root menu.(UNIX)

```
% cr5000 [Return]
```





Start the CR-5000 root menu.

Click the [SWS Conversion] button on the Translator menu to start Data Converter.

You can start Data Converter from the System Designer editor utility or from File Manager.

Launch the file manager in the System Designer, and select [SWS Schematic sheet] or [Common Device Database File].

In the default installation environment, you can start Data Converter from FileManager for the following file types:

- Circuit directory (.cir)
- Parameters for components library for schematic design (.prf)
- SWS circuit block /SWS schematic sheet (.001Å`.256)
- EDIF 200 Schematic view (.eds)
- Part property master file (.pcp)

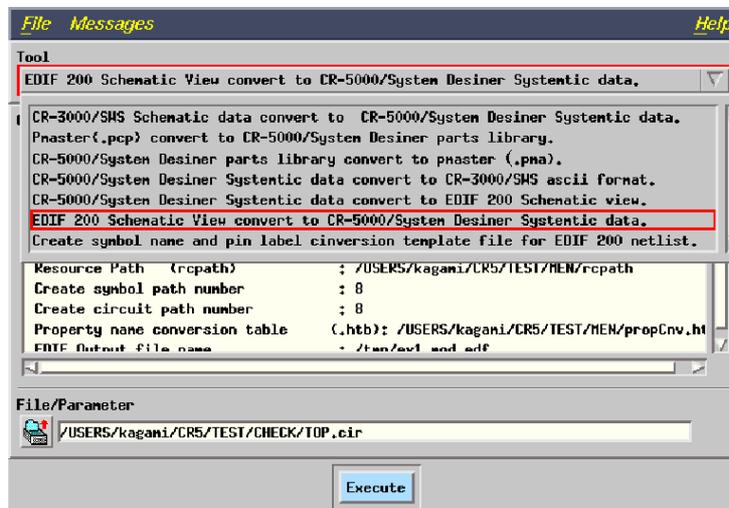
### 1.1.2 Exiting the Tool

Select "File..." from the Data Converter menu to bring up a Pull-down menu. Select "Close..." from the Pull-down menu to exit the tool.

When you try to end the tool or select [Save] from the Pull-down menu, a confirmation dialog box appears asking you whether to save the startup parameters shown in the List Box. Click OK to save the settings in \$HOME/CR5000/ds/battool.ini.

### 1.1.3 Selecting the Tool

Select the desired tool from the Option list for tool selection.



When you select the tool, the startup parameters are loaded and a list of startup options appear in the List Box.

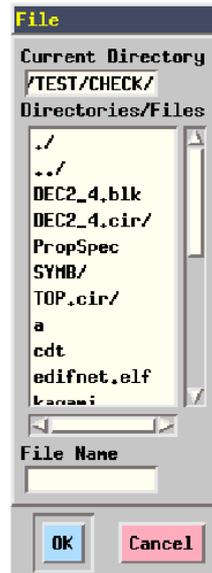
The startup parameters are read from \$HOME/CR5000/ds/battool.ini. If this file is not found, the parameters are read from \$ZDSROOT/info/battool.rsc.

### 1.1.4 Changing the Startup Parameters

Click a parameter you want to change from the List Box to copy it to the "Select File" field. After changing it, press the Return key to write it back to the List Box.

## 1.1.5 File Opener

### Directory Name



By selecting the directory name field on the upper part of the File Opener , you can enter a directory name from the keyboard.

Select the check mark to the right of the field or press the Return key to display the names of files and directories.

Selecting the "x" mark restores the previous directory name.

You can also select a directory name using the list that is explained below:

### List

The opener list is the area located at the center of File Opener showing file and directory names.

Single-click a file name and click the OK button or press the Return key to select a file. You can also select a file by double-clicking the file name. The file name appears in the file name field on the lower part of the File Opener .

Double-clicking a directory name lists the names of files and directories contained in it.

### Buttons

Clicking the OK button inputs the file name.

Clicking the "Cancel" button erases the File Opener and returns you to the previous screen.

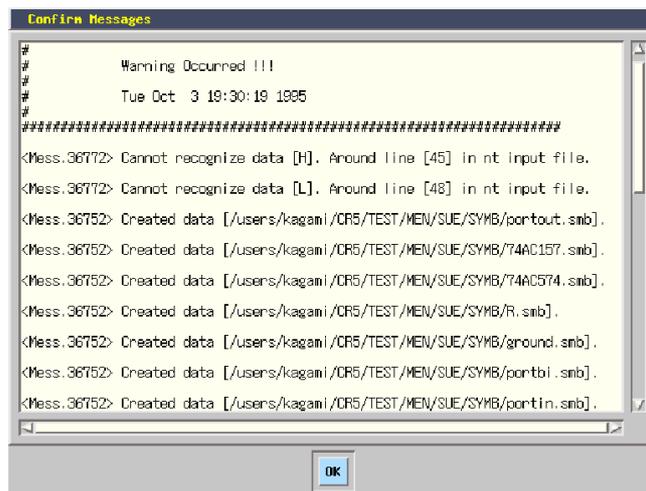
### 1.1.6 Executing the Tool

After setting all the parameters, click the "Execute" button to start the tool.

A guide message appears when starting or exiting the tool. Check the message and then click the "Ok" button.

### 1.1.7 Confirming Error and Warning Messages

Click "Error" or "Warning" on the Pull-down menu . If the program has created an error or warning file ("program name + err" or "program name+ wrn"), the contents are displayed in the message confirmation dialog box. An example of the dialog is shown below.



- Note)  
If an error or warning file is specified in the "Select File" field, that file appears in the message confirmation dialog box.

## 1.2 Setting the Parameters

This section explains the parameters required for starting EDIF Writer .

### vendor definition file

This file is an information file for converting data so that it can be read by each vendor's Edif Reader.

With the installation settings, data is output in actual dimensions (except for DCS ).

```
#####
#
#   if DS_UNIT = 0 THEN auto scale
#   DS_UNIT = 1000000 and EDIF_UNIT = 0.79375
#   4 mm --> 1/8inch(3.175mm)
#
config {
    (PORT_NAME   T)
    (PORT_RECT   dot)
    (FLOAT_CONV  cut)
    (SCHEMA_FIG  yes)
    (DS_UNIT     0)    <-- Scale control parameter
    (EDIF_UNIT   0.00254) <--Scale control parameter
    .....
}
```

If DS\_UNIT = 0, the scale of the EDIF file is output in actual dimensions. This is the same as the following.

```
(DS_UNIT   4)
(EDIF_UNIT 0.00254)
```

If a value other than 0 is specified in DS\_UNIT, the value is described in the scale of the EDIF file.

Suppose you enter as follows:

```
(DS_UNIT   4)
(EDIF_UNIT 0.00254)
```

Then in the scale of the EDIF file, the value will be described as follows:

```
(technology
  (numberDefinition
    (scale 4
      (e 254 -5)
      (unit DISTANCE)))
```

This indicates that the value entered in the EDIF file is "4 = 254e-5".  
To adjust the output scale, change the following parameters.

### **Property conversion file**

The Property conversion file is used to convert System Designer property names to other CAD property names. If the Property conversion file is not

specified, System Designer property names are converted to EDIF 200 Schematic view property names without changing the names.

For the format of the Property conversion file , refer to the description of the "1.3 Property conversion file" .

Only the "properties" table is used by the EDIF 200 Schematic View reader for CR-5000/System Designer . The "fileChartr" table is not referenced.

### **Output file**

Be sure to specify the Output file.

### **Target**

Specify the circuit directory.

## 1.3 Property conversion file

The file name of the Property conversion file for EDIF 200 Schematic view must be specified with "up to eight characters + .htb".

(1) "properties"

This is used to output the System Designer property names using different ones.

One line consists of three fields, corresponding to one property name.

- Field 1

This is "the property name to be output to the EDIF file".

- Field 2

This is "the property name of the System Designer".

Please describe the property name which has been registered in the property definition file (\$ZDSROOT/etc/jpn/PropSpec).

- Field 3

This field contains the viewer number of System Designerdata. This is for the EDIF Reader in Rev. 4.0 and before, not for the EDIF Writer..

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

The property names not described in this table will be output to the EDIF file, with the names described in the file.

(2) "ripperMode"

"ripperMode" specifies whether ripper is output to the EDIF file.

"ripperMode" is set to "yes" by default.

Keyword	Contents
yes	Outputs EDIF file "ripper"
no	Does not output EDIF file "ripper".

## 1.4 Setting file by Vendor

### 1.4.1 Convert to DCS

The following files are provided in order to output the EDIF 200 Schematic View for DCS.

Type	File name
Property conversion file <sup>a</sup>	\$ZDSROOT/info/propCvtD.htb
vendor definition file	\$ZDSROOT/info/formCnvD.evf

a.User defined file

### 1.4.2 Conversion from VIEW

The following files are provided to convert the data of EDIF 200 Schematic view for VIEW.

Type	File name
Property conversion file <sup>a</sup>	\$ZDSROOT/info/propCvtV.htb
vendor definition file	\$ZDSROOT/info/formCnvV.evf

a.User defined file

### 1.4.3 Conversion from MENTOR

The following files are provided to convert MENTOR EDIF 200 Schematic view .

Type	File name
Property conversion file <sup>a</sup>	\$ZDSROOT/info/propCvtM.htb
vendor definition file	\$ZDSROOT/info/formCnvM.evf

a.User defined file

## 1.5 Restrictions

- Reference  
In the DCS mode, the reference of an instance symbol is not output.
- Interface  
The interface at the topmost hierarchy outputs the data for the hierarchy connector in the schematic sheet at the topmost hierarchy.
- Line style  
Line style is not output.
- Color  
Color is not output. The default value defined in the vendor format file is used.
- Automatic generation of hierarchy connector  
In System Designer, hierarchies are connected using block pin IDs and signal names. In EDIF, however, hierarchies are connected using block pin names and corresponding pins of schematic sheets. Therefore if the hierarchy connector is not entered correctly, hierarchies are disconnected. In this case, read the data using the vendor's reader and edit it with an editor.
- Hatching  
Starting from the Shell

---

## **Chapter 2 Launch the Shell (Batch)**

---

The EDIF 200 Converter Module Writer can be launched from the shell using a batch script. This chapter explains how to launch programs from the shell.

## 2.1 Format of Startup Options

Options are passed to each program in the unified format shown below.

Program name [optional parameters] target data name
---

The following option parameters are provided:

Option name and argument	Meaning
-reference file name	Specifies reference file name
-l Library file name	Specifies reference file name
-p Parameter file name	Specifies parameter file
-x Extension name	Specifies output file identifier
-o Output file path name	Specifies output file path name
-m Mode name	Specifies processing mode
-p: Parameter name, value	Specifies parameters
-V	Outputs version
-e Error file path name	Specifies error message output file
-w Log/warning file path name	Specifies log/warning output file

\* Note:

- You must use the absolute path name to specify each file or directory.
- You must use a slash (/) as a delimiter for the path name.
- If you use a back slash (\), some operations will not be completed properly.

## 2.2 The launch options used for the EDIF 200 Converter Module Writer

(1) Program name

HP-UX : \$ZDSROOT/bin/HP64/zds2edif.sh

Solaris : \$ZDSROOT/bin/SOLARIS/zds2edif.sh

Windows : \$ZDSROOT/bin/Win32/zds2edif.exe

(2) Startup options

Option name	argument	Meaning
-r	File path (.evf)	vendor definition file <sup>a</sup>
-r	File path (.htb)	Property conversion file
-o	1 or greater nu-meric value	Output file <sup>a</sup>
-V	None	Outputs program version

a. Required

(3) Target

- Specify the circuit directory of System Designer
- If you specify a symbol file as a target, only one symbol will be generated in the EDIF file.
- If you specify a regular directory as a target, the entire symbols in that directory are, as a library, generated in the EDIF file.

---

# **Appendix A Schematic View Output Specifications**

---

The format of EDIF 200 Schematic View output varies depending on the vendor format file.

The following sections deal with the most common format generated by this program.

## A.1 EDIF200 Overview

This chapter outlines the EDIF200 syntax this program supports.

### A.1.1 Overview of the EDIF Schematic View File

This section outlines the EDIF file organization.

In the examples below, <xxxx> represents the above data and "xxxx" character data. Descriptions follow a colon (:).

```
(edif
  (library
    (technology ...) : Defines a group such as
                      character properties and colors.
    (cell          : Start of symbol and circuit description
      (view
        (interface : Start of connection information
          (port ...) : Definition of external pin
            (symbol ...) : Description of symbol figure
          )
        (contents
          (offPageConnector ...) : Definition of sheet connector
          (page                : Description of schematic sheet
            (portImplementation ...) : Description of connector
            (instance          : Description of instance symbol
              (portInstance ...) : Description of instance symbol pin
            )
          (net ...)           : Description of net
        )
      )
    )
  )
)
```

## A.1.2 Special Properties

In the System Designer database, net names and pin numbers are attached to objects as properties. The edif writer outputs a property by converting it to a specified property name according to the property conversion table. The following properties, however, are not output as properties but output according to the following specifications.

	Property	Output specification
1	Reference of symbol	(designator "reference")
2	Symbol pin name ("pinLabel")	(port <pin name>)
3	Symbol pin number	(designator "pin number")
4	Reference of instance symbol	(designator "reference")
5	Pin number of instance symbol	(designator "pin number")
6	Net name ("netLabel")	(net "net name")

## A.1.3 Hierarchical Structure

The following figure illustrates the hierarchical structure of System Designer.

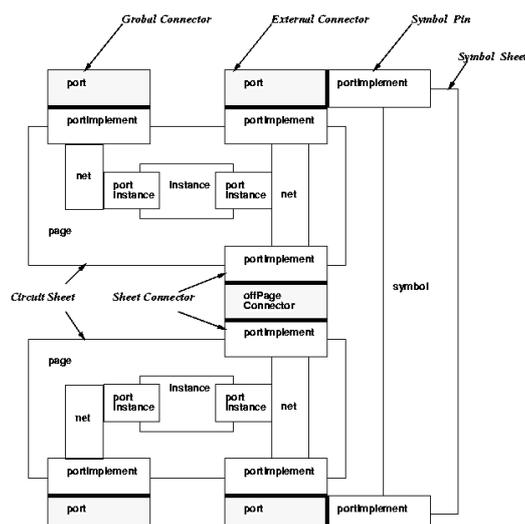


Figure A.1 Hierarchical Structure

## A.2 EDIF Schematic View Syntax

The following sections describe the output format of the System Designer EDIF output program.

Although the format slightly differs among vendors, a format not described here does not necessarily cause an error because the EDIF input program is able to interpret non-standard formats to a certain extent.

### A.2.1 edif

edif: This keyword indicates the start of an edif file.

```
(edif <schematic name> : <schematic name>
                        describes
                        : the circuit directory name (except for .cir)
                        : of the topmost hierarchy.
)
```

Define the following items in this keyword.

- (edifVersion)  
Represents the edif version. Fixed at (edifVersion 2 0 0)
- (edifLevel)  
Fixed at (edifLevel 0).
- (keywordMap)  
Fixed at (keywordMap (keywordLevel 0)).
- (status)
- (library)

## A.2.2 Status

Describes information such as the date of file creation.

```
(status
  (written
    (timeStamp 1995 3 17 23 7 37) : Year, month, day, hour, minute,
                                and second of creation
    (author "Zuken Incorporated") : Contents of format file
    (program "Zuken Schematic edif writer" : Contents of format file
      (version "1.201") : Version number of program
    )
  )
)
```

## A.2.3 Libraries

The keyword (library) indicates the start of the circuit library. More than one library can be defined in an (edif).

```
(library <library name> : Specify in <library name>
                        : the circuit directory name (except for .cir)
                        : of the topmost hierarchy.
)
```

Define the following items in this keyword.

- (edifLevel)  
Fixed at (edifLevel 0).
- (technology)  
Defines scale information (scale) and various default properties (figureGroup).
- (cell)  
Symbols and circuit information.

## A.2.4 Scale

Defines the unit of circuit information for each library. Because floating point is not supported in edif, a scale value is expressed with a combination of three integers as shown below:

```
(scale 1          : The unit of "1" defined in the file
  (e 1 -6)       : is "1" multiplied by "10 to the minus 6th power"
  (unit DISTANCE) : meters, used only for distance.
)
```

No other description is provided for data of System Designer.

Circuit systems using inch units such as DCS may have the following description.

- (Scale 1 (E 254 -8) (Unit DISTANCE))  
1 unit = 2.54 micrometers

## A.2.5 Property Group

This group defines various default properties.

The program outputs only the following three properties for each group. The default values output here are defined in the vendor format file.

- color
- textHeight
- visible
- pathWidth
- borderWidth

```
(figureGroup <property group name> : Format file
  (color <R> <G> <B> )           : Format file
  (textHeight <text height>)     : Format file
  (visible (true)                : "true" is always output.
)
```

The following 12 property groups are available. They are fixed by the system.

	Property group name	Contents
1	SYMBOL_FGP	Used for figures in a symbol.
2	NET_FGP	For single-bit signal lines
3	BUS_FGP	For bus lines
4	TERM_FGP	For pins
5	TERMN_FGP	For pin names
6	PINNUM_FGP	For pin numbers
7	PROPERTY_FGP	For various properties
8	TEXT_FGP	For text
9	ARC_FGP	For arc
10	CIRC_FGP	For circle
11	LINE_FGP	For line
12	FRAM_FGP	For frame

## A.3 Description of Symbol

Unit used for representing a symbol or circuit.

```
(cell <cell name>          : <cell name>
                           is a symbol name (except for .smb) or a circuit
                           directory name (except for .sht).
                           For circuit blocks, symbol names are used.
(cellType <cell type>)    : Format file
(view <view name>)        :
(viewType SCHEMATIC)     : SCHEMATIC => circuit
                           : NETLIST => edif-net

(interface                 : Start of interface

(descriptor "reference")  : "reference" <= Property value of "reference"
(property <property name> : <property name> <= Property name of symbol
                           property
(string "property value") : "property value" <= Property value of symbol
                           property
)
(port <pin name>          : <pin name> <= Property value of "part Name" of
                           symbol pin
(descriptor "pin number") : "pin number" <= Property value of "pinNumber"
(property <property name> : <property name> <= Property name of symbol
                           pin property
(string "property value") : "property value" <= Property value of symbol pin
                           property
)
)
(symbol                   : Start of symbol figure information
.....
)
)
)
)
```

### A.3.1 Definition of External Pins

- Bit pins

```
(interface
  (port <pin name>) : <pin name> <= Property value of
                    "partName" of symbol pin
)
```

- Multiple pins

```
(interface
  (port
    (array <pin name> <bit count>) :<pin name> = Property value of
                                     "partName" of symbol pin
    <bit count>= 2 or greater integer
  )
)
```

### A.3.2 Symbol Sheet Size

```
(symbol
  (pageSize          : Drawing size
    (rectangle       : Coordinate specification of drawing size
      (pt <X coordinate> <Y coordinate>) : Lower left coordinate of
                                          drawing size
      (pt <X coordinate> <Y coordinate>) : Upper right coordinate of
                                          drawing size
    )
  )
)
```

### A.3.3 Symbol Property Viewer

- Symbol name

```
(keywordDisplay CELL
  (display PROPERTY_FGP
    (origin
      (pt <X coordinate> <Y coordinate>) : Coordinates of
                                          symbol name for
                                          display
    )
  )
)
```

- Reference

```
(keywordDisplay DESIGNATOR
  (display PROPERTY_FGP
    (origin
      (pt <X coordinate> <Y coordinate>) : Coordinates of
                                          reference designator
                                          for display
    )
  )
)
```

- Property

```

(propertyDisplay <property name>
  (display PROPERTY_FGP
    (origin
      (pt <X coordinate> <Y coordinate>): Coordinates of
                                     symbol property for
                                     display
    )
  )
)

```

### A.3.4 Symbol Pins

Define the viewer for pins and pin properties as follows:

```

(portImplementation <pin name> : Specified
                                     with <pin name>
                                     <= (port <pin name>)
(keywordDisplay DESIGNATOR : Displaying pin number
  (display PROPERTY_FGP
    (origin
      (pt <X coordinate> <Y coordinate>) : Coordinates of pin
                                     number for display
    )
  )
)
(propertyDisplay <property name> : Displaying symbol pin property
  (display PROPERTY_FGP
    (origin
      (pt <X coordinate> <Y coordinate>) : Coordinates of
                                     symbol pin property
                                     for display
    )
  )
)
(connectLocation : Pin coordinates
  (figure <property group name> : <property group name>
    <= TERM@
  )
)
(dot
  (pt <X coordinate> <Y coordinate>) : Pin coordinates
)
)
)
)
)
)

```

Since pin shapes in System Designer are points, they are expressed in dots as shown above.

The DESIGNATOR attached to a symbol is the reference and the DESIGNATOR for a pin is the pin number.

### A.3.5 Symbol Figures

The following explains the description of figures (graphic elements) that can be defined in a symbol.

- Polyline

```
(figure LINE_FGP
  (path          : Symbol shape (line)
  (pointList     : Start of coordinates list
    (pt <X coordinate> <Y coordinate>) : Line construct point
    .....
  )))
```

- Circle

```
(figure CIRC_FGP
  (circle        : Symbol shape (circle)
  (pt <X coordinate> <Y coordinate>) : Point 1 of circle diameter
  (pt <X coordinate> <Y coordinate>) : Point 2 of circle diameter
  ))
```

- Arc

```
(figure ARC_FGP
  (openShape     : Open shape
  (curve         : 3-point specification
  (arc           : Symbol shape (arc)
    (pt <X coordinate> <Y coordinate>) : Start point of arc
    (pt <X coordinate> <Y coordinate>) : Middle point of arc
    (pt <X coordinate> <Y coordinate>) : End point of arc
  ))))
```

- Polygon

```
(figure LINE_FGP
  (polygon          : Symbol shape (polygon)
  (pointList       : Start of coordinates list
    (pt <X coordinate> <Y coordinate>) : Polygon construct point
    .....
  )))
```

### A.3.6 Symbol Comment text

Comment text for a symbol is expressed as follows:

```
(annotate
  (stringDisplay <string> : Comment
  (display TEXT_FGP
  (origin
    (pt <X coordinate> <Y coordinate>) : Coordinates of comment
                                     for display
  )
  )
)
```

### A.3.7 Pin Arrays

Multiple pins are expressed as follows:

```
(port
  (array <pin name> <bit count>) : <pin name> <= Property value of
                                   "partName" of symbol pin
  )                               <bit count> <= 2 or greater integer
```

### A.3.8 Displaying Properties

The keyword (display) is used for displaying various properties.

```
(display <property group name> : Default
                                property group name
(origin
 (pt <X coordinate> <Y coordinate>) : Coordinates for display
)
(justify <corner point> : UPPERLEFT, UPPERCENTER, UPPERRIGHT
                        : CENTERLEFT,CENTERCENTER,CENTERRIGHT
                        : LOWERLEFT, CENTERCENTER,LOWERRIGHT
(orientation<rotation> : R0, R90, R180, R270(without mirroring characters)
)
)
```

A corner point is a corner used as the origin of character data such as properties.

The angle of rotation is output in units of 90 degrees (edif specification).

Therefore property angles are rounded to a multiple of 90 degrees.

To change the part of the default properties defined at the beginning of (figureGroupe), use the keyword (figureGroupOverride).

To change how a property is displayed, define as follows:

```
(display
 (figureGroupOverride <property group name> : Default property group
                                                name
 (textHeight <height> ) : <height> <= text height
 (visible (false)) : Property is made invisible.
)
(origin (pt <X coordinate> <Y coordinate>))
)
```

# A.4 Connector Symbols

In the System Designer database, connectors exist as symbols that are categorized by "com-ponent kind". In System Designer edif, they are expressed as pins on the schematic sheet and the shapes are expressed using symbols (schematic sheet) as instances.

Symbols for connector shapes are not interfaced. Shapes are output so that the pins are generated at the symbol origin.

```
(cell <cell name>          : Default property
                                group name
  (view <view name>
    (viewType SCHEMATIC)
    (interface          : Not interfaced
    (contents
      (pageSize ... )    : Sheet size

      (figure LINE_FGP   : Line "" Same as for other figures(See Symbol
                                Figures)
    (path
      (pointList
        (pt <X coordinate> <Y coordinate>)
          .....
      )))
    (commentGraphics
      (property <property name> : Properties attached to connectors are
        (string "property value") expressed in comment graphics.
      ))
    )
  )
)
```

This format applies to symbols that have the following component kinds. Other symbols are generated as normal symbols.

- Sheet connector
- Hierarchy connector
- Power supply
- Ground

## A.4.1 Ripper Symbol

In the System Designer database, connection between a signal line and a bus line or between bus lines is expressed with the "hook shape" property that belongs to the signal line (bus line). In edif, connections are made via the "ripper" symbol. As this symbol is not provided in the System Designer database, it is automatically generated as follows:

```
(cell
  (rename DiagSplitter "DiagSplitter"):cell name
                                     (Specified in the format file)
  (cellType Ripper)
  (view Graphic_Schematic_1
    (viewType Schematic)(interface
      (port
        (array
          (rename RightSide "RightSide") 2048)) : Multiple pins (Pin name
                                                    and bit count are specified
                                                    in the format file)
        (port
          (array
            (rename LeftSide "LeftSide") 2048))
        (joined
          (portRef RightSide) : Connect RightSide to LeftSide)
          (portRef LeftSide)
        (symbol
          (portImplementation RightSide
            (connectLocation
              (figure TERM_FGP
                (dot
                  (pt 0 2000)))))) : Pin coordinates
          (portImplementation LeftSide
            (connectLocation
              (figure TERM_FGP
                (dot
                  (pt -2000 0)))))) : Pin coordinates
          (figure SYMBOL_FGP : Shape
            (path
              (pointList
                (pt 0 2000)
                (pt -2000 0)
                (pt -2000 500)
                (pt -1500 500)))))))))
```

## A.5 Schematic Sheet

A circuit block has a block symbol and an equivalent schematic sheet.

A block symbol is described in the same way as the symbol file mentioned earlier.

```
(cell <cell name>
  (view <view name>
    (viewType SCHEMATIC)

    (interface          : Interface
      (port ...)       : Definition of external interface

      (symbol ...)     : Block symbol shape
                        : (Only when an equivalent symbol exists)
    )
  )
  (contents             : Start of sheet
    (offPageConnector <sheet connector name>)
    (page <page number> : <page number> <= Sheet number
      (portImplementation ... ) : Sheet connector, hierarchy connector, power
                                supply, and ground
      (instance ... )         : Description of instance symbol
      (net ... )              : Description of net
      (commentGraphics ... ) : Description of figures and comments on
                                sheet.
    )
  )
)
)
```

### A.5.1 Schematic Sheet Size

The same as the description of the symbol drawing size.

```
(pageSize          : Drawing size
  (rectangle       : Coordinate specification of drawing size
    (pt <X coordinate> <Y coordinate>) : Lower left coordinate of
                                        drawin size
    (pt <X coordinate> <Y coordinate>) : Upper right coordinate of
                                        drawing size
  )
)
```

## A.5.2 Instance Symbol

The instance symbol is a symbol entered in the schematic sheet.

```
(instance <instance name>      : <instance name> <= Symbol ID
  (viewRef <view name>        : Specify the view to reference
    (cellRef <cell name>      : Specify the symbol to reference
      (libraryRef <library name> ) : Specify the library to reference
        ) : Omit this if the library for reference
      )
    )
  )
  )

(transform
  (scaleX <numerator> <denominator>) : Scale of instance symbol
                                       Omit if scale is 1.0
  (scaleY <numerator> <denominator>) : Output using X = Y
  (orientation <rotation>)           : <rotation> <= R0,R90,R180,R270,
                                       NX,MYR90,MX,MXR90 (origin
  (pt <X coordinate> <Y coordinate>) : Coordinates of instance symbol
  )
)

(designator <reference> ) : Reference
(property <reference> )   : Property of instance symbol

(portInstance ... )      : Instance symbol pin
)
```

All instance names are output using (rename). Instance names and rename names are generated as follows:

- Instance name = "cmp" + component ID number
- rename name = "I\_" + sheet number + "\_" + component ID

### A.5.3 Instance Symbol Pin

```
(portInstance <pin name> : <pin name> <= Equal to symbol pin name
  (designator "pin number")
  (property <property name>
    (string "property value")
  )
)
```

### A.5.4 Instance Symbol Reference

Define the reference of the instance symbol as follows:

- When the property viewer is not provided for the "reference" property

```
(designator "reference")
```

- When the property viewer is provided for the "reference" property

```
(designator
  (stringDisplay "reference"
    (display
      .....
    )
  )
)
```

### A.5.5 Instance Symbol Property

Define the instance symbol property as follows:

- When the property viewer is not provided

```
(property <property name>
  (string "property value")
)
```

- When the property viewer is provided

```
(property <property name>
  (string
    (stringDisplay "property value"
      (display
        .....
      )
    )
  )
)
```

## A.5.6 Instance Symbol Pin Number

Define the pin number of an instance symbol as follows:

- When the property viewer is not provided for the "pinNumber" property

```
(designator "pin number" )
```

- When the property viewer is provided for the "pinNumber" property

```
(designator
  (stringDisplay "pin number"
    (display
      .....
    )
  )
)
```

## A.5.7 Instance Symbol Pin Property

Define the instance symbol pin property in the same way as for the instance symbol property.

## A.6 Connectors

Define connectors as pins on the schematic sheet.

```
(contents
  (page &1 "1"

  (portImplementation <pin name>
    (connectLocation
      (figure TERM_FGP
        (dot
          (pt <X coordinate> <Y coordinate>) : Coordinates of pin on
                                          schematic sheet
        ))
      ))
    (instance <instance name>
      (viewRef <view name>
        (cellRef <cell name>
          )
      (transform
        (orientation <rotation>)
        (origin
          (pt <X coordinate> <Y coordinate>) : Coordinates of connector
                                          symbol on schematic
                                          sheet Normally they are
                                          the same as the
                                          coordinates of the pin on
                                          the schem atic sheet.
        ))
      ))
    ))
  ))
))
```

### A.6.1 Hierarchy Connector

The hierarchy connector is a pin on the schematic sheet created with the pin name defined in (interface).

```
(cell ..
  (view ..
    (interface
      (port <pin name> ) <-----+
      (symbol ...)          |
    )                        | The pin names are the same
    (contents                |
      (page &1 "1"          |
        (portImplementation <pin name><----+
          .....
        ))))
```

### A.6.2 Sheet Connector

The sheet connector is a pin on the schematic sheet created with the pin name defined as follows.

```
(cell ..
  (view ..
    (interface ...)
    (contents
      (offPageConnector           : Definition of sheet connector
        (rename opc_XXX "connector name" ) : opc_XXX Rename connector
                                           name
      )
      (page &1 "1"                : Sheet1
        (portImplementation opc_XXX   : Sheet connector
          .....
        )
      )
      (page &2 "2"                : Sheet2
        (portImplementation  opc_XXX   : Sheet connector
          .....
        )
      )
    )
  )
)
```

### **A.6.3 Power Supply and Ground**

Define the instance symbol with the power supply/ground property as an undefined pin on the schematic sheet.

## A.7 Nets

Nets are generated using the following expression:

```
(net
  (rename <RENAME> "net name") : Property value of "netLabel"
  (joined : Start of connection information

    (portRef <instance pin name> : Name of instance symbol pin to connect
      (instanceRef <instance name>): Name of instance symbol to connect
    )

    (portRef
      (member <instance pin name> <bit number>) : Bit number of multiple
                                                pin
      (instanceRef <instance name>): Name of instance symbol to connect
    )
  )
  (figure NET_FGP : Net shape
    (path
      (pointList
        (pt <X coordinate> <Y coordinate>)
        .....
      )))
  )))
```

Use the following expression to display a net name.

```
(net
  (rename <RENAME>
    (stringDisplay "net name"
      (display ...))
  )))
```

Use the following expression to display more than one net name.

```
(net
  (rename <RENAME>
    (stringDisplay "net name"
      (display ...)
      (display ...)
      .....
    )))
```

## A.7.1 Bus

A bus line is expressed using an array of net names. To connect bus lines and signal lines, use the ripper (splitter).

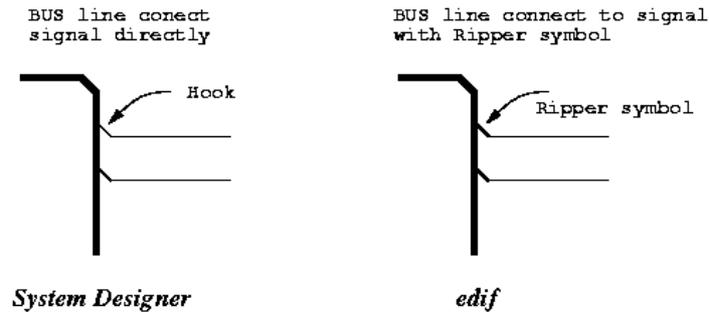


Figure A.2 Bus connection

A bus line is expressed with an array of net names.

```
(net
  (array <bus name> <bit count>)
  (joined
    (portRef <pin name>)      : Connect to connector

    (portList
      (portRef                : (portRef) of bit count is described in
      (portList)
        (member <pin name> <bit number>): Bit number pin is connected to
                                          signal with the same bit number
                                          on bus

        (instanceRef <instance name>) : Instance symbol of ripper
      )
      .....
      )
      .....
      (Figure ...)
    )
  )
)
```

Use the following expression to display a bus name.

```
(net
  (array
    (rename <RENAME>
      (stringDisplay "bus name"
        (display ...))
    )
  )
  <bit count>
)
```

Use the following expression to display more than one bus name.

```
(net
  (array
    (rename <RENAME>
      (stringDisplay "bus name"
        (display ...)
        (display ...)
        .....
      )
    )
  )
  <bit count>
)
```

For bus names, parentheses and range characters are converted according to the vendor format file.

## A.7.2 Figures on the Schematic Sheet

Use the following expressions to output figures and text on a schematic sheet and the property of the schematic sheet.

- Polyline

```
(commentGraphics
  (figure LINE_FGP
    (path ...) : Symbol shape (line)
  ))
```

- Circle

```
(commentGraphics
  (figure CIRC_FGP
    (circle ...)      : Symbol shape (circle)
  ))
```

- Arc

```
(commentGraphics
  (figure ARC_FGP
    (openShape        : Open shape
      (curve          : 3-point specification
        (arc ...)
      )))
```

- Polygon

```
(commentGraphics
  (figure LINE_FGP
    (polygon ...)    : Symbol shape (polygon)
  ))
```

- Text

```
(commentGraphics
  (annotate
    (stringDisplay <String> : Comment
      (display TEXT_FGP
        (origin
          (pt <X coordinate> <Y coordinate>) : Coordinates of
            comment for display
        )))
```

- Property

```
(commentGraphics
  (property
    (rename <RENAME> "property name")
    (stringDisplay <property value>
      (display ...)
    )))
```

# A.8 DCS Mode Specific Specification

## A.8.1 Handling Connector Symbols

In the DCS mode, connector symbols (component type: hierarchy connector, sheet connector, power supply, and ground) are not output as symbols. Connectors are expressed as pins on the schematic sheet and their shapes are directly expressed in the coordinate system on the schematic sheet.

- General Output

```

(cell <connector symbol name>          <-----+
  (view <view name>                    <-----+ |
    (interface)                        | |
    (contents                            | |
      (commentGraphics : Shape-only symbol (schematic sheet) | |
        ....                             | |
        )))                               | |
    ....                                 | |
  (cell <schematic>                     | |
    (interface)                         | |
    (contents                             | |
      (page &1                            | |
        (instance ...)                   | |
        (portImplementation <pin name>   | |
          (connectLocation ...)          | |
          (instance <instance name>     | |
            (viewRef <view name>         -----+ |
              (cellRef <connector symbol name> ) -----+
            ))))
  ))))

```

- DCS mode

```

(cell <schematic>
  (interface)
  (contents
    (page &1
      (instance ...)
      (portImplementation TT_12
        (property portShape <---- Property for recognizing
          connector shape

          (string "14")
          (property portOrientation <---- Property for recognizing
            connector orientation
            (string "0") <---- 0,1,2,3
            (connectLocation ...)
            (figure LINE_FGP
              (path
                (pointList
                  (pt 44000 56000)
                  (pt 41500 56000)
                  (pt 39000 57000)
                ))
                _ Output shapes using
                _ coordinate system on
                _ schematic sheet
              ))
            ))
          ))
        ))
      (figure LINE_FGP
        (path
          (pointList
            (pt 39000 55000)
            (pt 41500 56000)
          ))
        ))
      ))
    ))
  ))

```

## A.8.2 Reference

In the DCS mode, reference designators are not output. Use the DCS utility to assign components.