cadence™

Allegro® Design Entry HDL EDIF 300 User Guide

Product Version 23.1
September 2023

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Contents

Preface

About This Manual

This user guide is based on the assumption that you are familiar with the development and design of schematics and with EDIF 300 semantics. This manual contains information about the following translators:

- Allegro Design Entry HDL EDIF 300 Schematic Reader that lets you read in schematic data, including graphical and logical connectivity information.
- Allegro Design Entry HDL EDIF 300 Schematic Writer that lets you write out schematic data in the EDIF 300 format.
- Allegro Design Entry HDL EDIF 300 Netlist Reader that lets you read in only connectivity data. Connectivity data is a nongraphical representation of the logical connectivity information of a design.
- Allegro Design Entry HDL EDIF 300 Netlist Writer that lets you write out connectivity data in the EDIF 300 format.

Finding Information in This Manual

The following table provides an overview of the topics described in this manual.

For Information About . . .	Read . . .
The EDIF 300 format and the EDIF 300 translators	Chapter 1, “Understanding EDIF”
Schematic Reader	Chapter 2, “Schematic Reader – Importing EDIF 300 Schematic Data”
Schematic Writer	Chapter 3, “Schematic Writer – Converting Schematic to EDIF 300”
Netlist Reader	Chapter 4, “Netlist Reader – Importing EDIF 300 Connectivity Data”

For Information About . . .	Read . . .
Netlist Writer	Chapter 5, “Netlist Writer – Creating EDIF 300 Connectivity Data”
Allegro Design Entry HDL EDIF 300 Dialog Box help	Appendix A, “Dialog Box Reference,”
Limitations of Allegro Design Entry HDL EDIF 300 Translators	Appendix B, “Error Messages and Limitations,”

1

Understanding EDIF

Topics covered in this chapter are:

Overview of EDIF

Electronic Design Interchange Format (EDIF) is a format used to exchange design data between different CAD systems. This nonproprietary, standard interchange format uses text to describe electronic design data. EDIF 300 uses a LISP-like syntax to express constructs that represent electronic design.

Allegro Design Entry HDL EDIF Reader (EDIF Reader) translates a given EDIF description to a Design Entry HDL view. Allegro Design Entry HDL EDIF Writer (EDIF Writer) writes out the EDIF description for a given Design Entry HDL design.

For more information about the EDIF 300 syntax and the functional and semantic descriptions of each EDIF 300 construct, see the *Electronic Design Interchange Format Version 300, Level 0, Reference Manual,* EIA-618, December 1993, Volume 2 of 4.

EDIF Libraries

In EDIF 300, a library can be considered as a group of cells and templates that share the same technology information.

EDIF 300 supports two types of libraries. These are external libraries and local or design libraries. External libraries are the reference libraries that are assumed to be present in the Cadence database. EDIF 300 design libraries are created in the specified run directory and contain details of cells and templates used in the current EDIF 300 file.

EDIF Cells and Clusters

A component in design architecture is represented as a cell in EDIF 300. An EDIF 300 cell is a design object that can have a variety of views associated with it. Each view describes its cells from a different perspective. The design process applied to a cell is dictated by the associated views such as, SchematicView, ConnectivityStructure, or SchematicSymbol.

Similarly, in EDIF 300, a cluster defines a design interface. All cells, such as views and symbols, that share a common interface are grouped under a cluster. An EDIF 300 cell can have multiple clusters. You can instantiate a cluster. When a cluster is instantiated, the instance is created from the cluster instead of from the individual views.

Overview of EDIF 300 Translators

Allegro Design Entry HDL EDIF 300 translators supports EDIF 300 Level 0 keywordLevel 0. Allegro Design Entry HDL EDIF 300 translators give you several ways to translate design data

into and out of the Allegro Design Entry HDL architecture. With the EDIF 300 Translators, you can do the following:

- Read in schematic data, which includes graphical and logical connectivity information, using the *Schematic Reader* `(e2csch)` option of Allegro Design Entry HDL EDIF 300.
- Read in connectivity data (only) using the *Netlist Reader* option of EDIF 300 (`e2conn`).

  Connectivity data is a nongraphical representation of the logical connectivity information of a design.
- Write out schematic data using the *Schematic Writer* option of EDIF 300 (`c2esch`).
- Write out connectivity data (only) using the *Netlist Writer* option of Allegro Design Entry HDL EDIF 300 (`c2econn`).

Do not use Allegro Design Entry HDL EDIF 300 for a schematic round-trip. It is not supported. For more information, see Problems after a Schematic Round-Trip***.***

EDIF 300 translators are commonly used for transferring schematic data from one vendor to those of another. You use EDIF 300 writer to convert a schematic design into an EDIF 300 file. An EDIF 300 reader then reads this EDIF 300 file to generate the schematic.

The EDIF 300 interface cannot be used to migrate an Allegro Design Entry HDL schematic to OrCAD Capture or the other way around. This is because Capture supports EDIF 200, and EDIF 200 files cannot be translated using EDIF 300 translators.

2

Schematic Reader – Importing EDIF 300 Schematic Data

This chapter discusses the following:

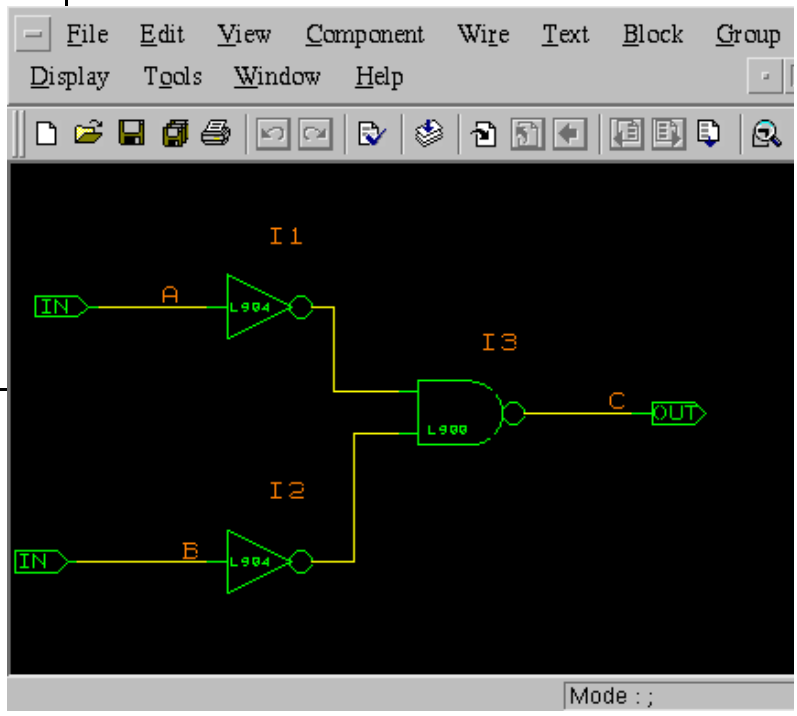
Overview

Allegro Design Entry HDL EDIF 300 Schematic Reader translates schematic data in an EDIF 300 file into the Cadence database. When you translate EDIF 300 schematic data with EDIF 300 Schematic Reader, you preserve both the graphical representation of the schematic and the connectivity information.

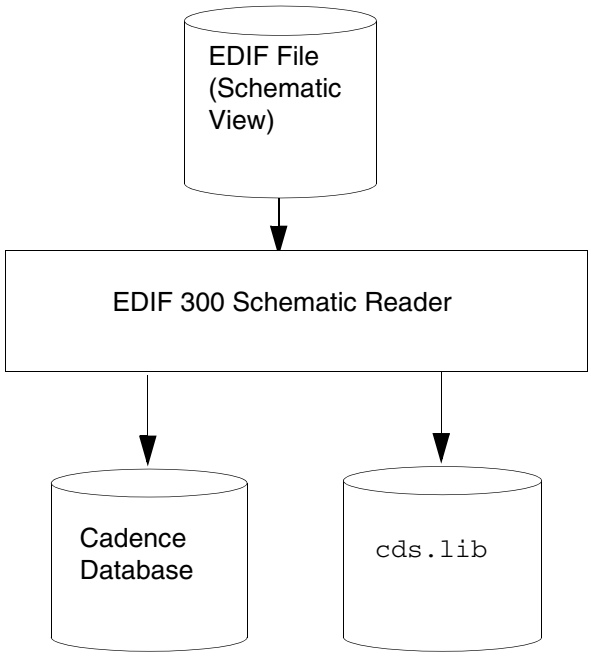
EDIF 300 Data File

```
(edif CADENCE_EDIF
  (edifVersion 300)
  (edifHeader
    (ediflevel 0)
      (keyWordMap (k0KeywordLevel))
      (unitDefns      (unit distance 1 (e
254001)))
      (fontDefinitions
      (fonts (setDistance (unitRef
distance))
            (font stick
          (typeface "stick")
          (fontProportions
            (fontHeight 10)
            (fontDescent 2)
            (fontCapitalHeight 8)
            (fontWidth 5)))))
```

Allegro Design Entry HDL
Schematic Data



The following figure shows all the information needed to translate an EDIF 300 file into the Cadence database.



Using Allegro Design Entry HDL EDIF 300 Schematic Reader

Allegro Design Entry HDL EDIF 300 provides two types of EDIF 300 readers, Schematic Reader and Netlist Reader. This chapter tells you how to use the Schematic Reader of EDIF 300. For information about using the *Netlist Reader*, see "Netlist Reader – Importing EDIF 300 Connectivity Data" on page 39.

Using Schematic Reader from Project Manager

To open Allegro Design Entry HDL EDIF 300 Schematic Reader from Project Manager:

1. Choose *Tools > EDIF 300*.

   The *EDIF 300* dialog box appears.

2. Select the *Schematic Reader* tab.

3. In the *Schematic Reader* tabbed page, specify the EDIF source file and the name of the output directory. To know more about the options in the Schematic Reader tabbed page, see Schematic Reader.
4. Click *Run* to start the translation.

Using Schematic Reader from the Command Line

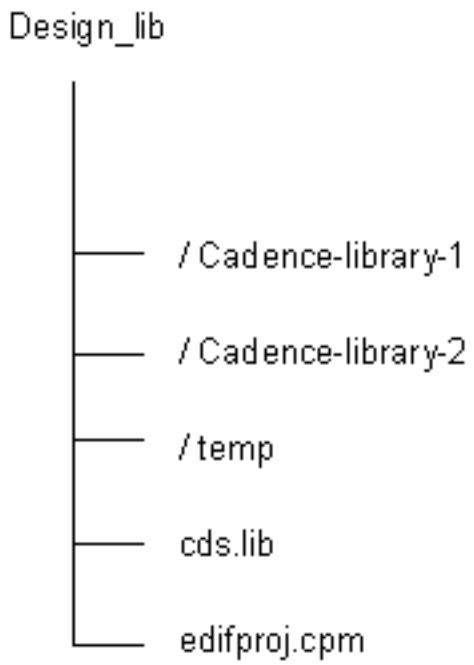
Allegro Design Entry HDL EDIF 300 Schematic Reader can also be used in the batch mode from the command line. The syntax for invoking the Schematic Reader from the command line is shown below:

```
e2csch -proj <edif_file> -out_file <location_of_the_output_directory>
```

EDIF files that are provided as an input to Allegro Design Entry HDL EDIF 300 Reader need to be syntactically and semantically correct.

Schematic Reader Output Files

After a successful translation, the following directory structure is created:



In addition to a translated Cadence Database library, EDIF 300 Schematic Reader produces log messages. The messages are displayed on the screen as well as written to a file, depending on how EDIF 300 Reader was invoked.

- When you use the EDIF 300 Reader to run a translation, the software writes the log messages to the `e2csch.log` file. This file is located in the `temp` folder of the design directory.
- When you start EDIF 300 Reader by typing `e2csch` at an operating-system prompt, the software displays the log messages on your screen.
- EDIF 300 Reader also creates the `cds.lib` file that contains the paths to the required Cadence libraries.

Functional Description

This section describes the mapping from EDIF 300 design objects to Allegro Design Entry HDL database. The mapping between EDIF 300 design objects and Allegro Design Entry HDL objects is described in Table 2-1 on page 16. Table 2-2 on page 18 covers the mapping between EDIF 300 connectivity objects and Allegro Design Entry HDL objects.

Table 2-1 Map Table for EDIF 300 Schematic Interfaces

EDIF 300 Objects	Allegro Design Entry HDL Objects
Units, UnitRef, setDistance	No Mapping
Schematic Unit	
Schematic Metric	hotspotGrid
External Library	Mapped to reference library
EDIF Design library	Library
Edif Property	Allegro Design Entry HDL user property
font Defs	No Mapping, use default mapping
technology	No mapping
nameInformation	Allegro Design Entry HDL object name
figureGroup	No Mapping
Cell (with Cluster)	Allegro Design Entry HDL Cell
cluster	See the cell mapping
interface (of a cluster)	No mapping
schematicView	pageX.csb file
schematicSymbol	symbol

Table 2-1 Map Table for EDIF 300 Schematic Interfaces

EDIF 300 Objects	Allegro Design Entry HDL Objects
logical connectivity	No mapping
schematicImplementation	
instance	
port	
portBundle	
signal	
signalGroup	
global ports	
Master Port Template	No Mapping
Global Port Template	
Off/On Page Connector Template	
Page Title Block Template	
Page Border Template	
Symbol Port Template	Not mapped in Allegro Design Entry HDL.
Schematic Ripper Template	
Schematic Junction Template	
Interconnect terminator template	
pages	pagex.csb
schematic Master Port Implementation	Allegro Design Entry HDL Port
schematic Symbol Port Implementation	Allegro Design Entry HDL Symbol
schematic Global Port Implementation	Allegro Design Entry HDLL Global Port
schematic Page Connector Implementation	No Mapping
schematic Instance Implementation	Allegro Design Entry HDL schematic instance
schematic Ripper Implementation	No Mapping required

Table 2-1 Map Table for EDIF 300 Schematic Interfaces

EDIF 300 Objects	Allegro Design Entry HDL Objects
schematic Junction Implementation	No Mapping
schematicNet	Allegro Design Entry HDL net in schematic
schematicBus	Allegro Design Entry HDL net in schematic
schematic NetJoin/BusJoin	No mapping, because schematic extractor builds the connectivity
figures in schematic symbol	Allegro Design Entry HDL body figure for drawing purpose
figures in schematicNet/schematicBus	Allegro Design Entry HDL 2 point paths in wire layer, drawing purpose
figures for schematic master port, global port and page connector template	Allegro Design Entry HDL body figures, pin purpose
schematic Figure Macro	Instance of figure macro cell
schematicInterconnectNameDisplay	Allegro Design Entry HDL label attached to a net displaying net name.(Signame)
annotate	Allegro Design Entry HDL normal label (Text)
commentGraphics->annotate	Allegro Design Entry HDL user data (entered Text)
Bounding Box	rectangular border for pageBorder, and corresponding templates

Table 2-2 Map Table for EDIF 300 Connectivity Objects into Allegro Design Entry HDL

EDIF 300 Object	Allegro Design Entry HDL Object
External Library	Mapped to reference library
EDIF Design library	Allegro Design Entry HDL library
cell (with Cluster)	cell

Table 2-2 Map Table for EDIF 300 Connectivity Objects into Allegro Design Entry HDL

EDIF 300 Object	Allegro Design Entry HDL Object
cluster	See the last mapping
logical connectivity	used to generate Allegro Design Entry HDL netlist when logicalConnectivity is under connectivityView and no connectivity structure is given.
connectivity Structure	Allegro Design Entry HDL netlist connectivity
instance	Allegro Design Entry HDL instance
port	Allegro Design Entry HDL terminal
portBundle	Allegro Design Entry HDL terminal
signal	used to create a net in Allegro Design Entry HDL netlist view when only logical connectivity is there.
signalGroup	used to create Allegro Design Entry HDL net in netlist view when only logical connectivity is there.
global ports	global terminals in Allegro Design Entry HDL
connectivityNet	Allegro Design Entry HDL net in connectivity view
connectivityBus	Allegro Design Entry HDL net in connectivity view
connectivityNetJoin	Using the endpoints in netJoin, explicit connection is established.
connectivityRipper	Different Allegro Design Entry HDL object will be created in place of ripper. For example, nets will be merged/ synonymed.

3

Schematic Writer – Converting Schematic to EDIF 300

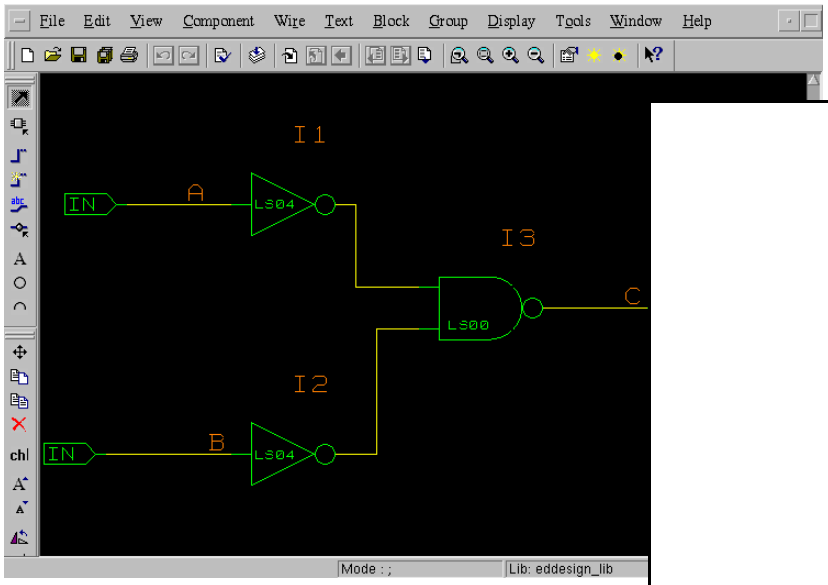
This chapter discusses the following:

Overview

The EDIF 300 Writer is used to translate schematic design data files from the Cadence database into the EDIF 300 format. When you translate schematic data with EDIF 300 Writer,

you can preserve both the graphical representation and the connectivity information of the schematic.

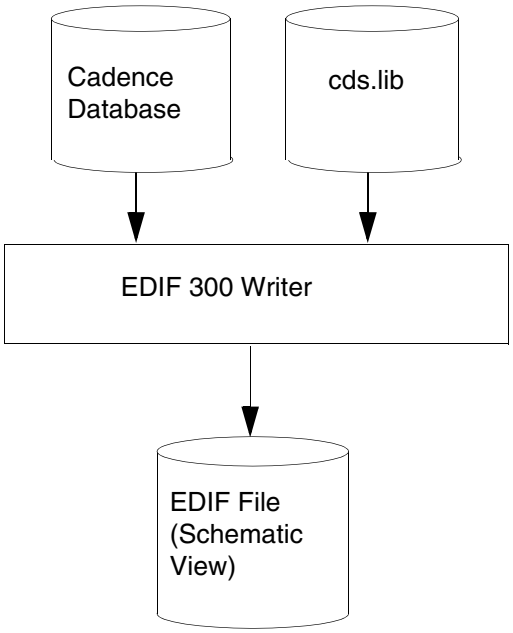
Allegro Design Entry HDL Schematic Data



EDIF 300 Data File

```
(edif CADENCE_EDIF
(edifVersion 300)
(edifHeader
  (ediflevel 0)
  (keyWordMap (k0KeywordLevel))
  (unitDefns    (unit distance 1 (e
254001)))
  (fontDefinitions
  (fonts (setDistance (unitRef
distance))
    (font stick
      (typeface "stick")
    (fontProportions
      (fontHeight 10)
      (fontDescent 2)
(fontCapitalHeight 8)
(fontWidth 5)))))
.
    .
    .
```

The following figure shows all the information needed to translate a Cadence database (Allegro Design Entry HDL Design) file into an EDIF 300 file.



Using EDIF 300 Schematic Writer

You can use EDIF 300 Schematic Writer either from Project Manager or in the batch mode.

Note: Before translation, you must check and save the schematic to ensure that there are no design errors. For more information on the Allegro Design Entry HDL schematic editor, see *Allegro Design Entry HDL User Guide*.

Using Schematic Writer from Project Manager

To invoke Schematic Writer from the Project Manager,

1. Choose *Tools > EDIF 300*.

   The *EDIF 300* dialog box appears.

2. Select the *Schematic Writer* tab.

3. In the *Project File* text box, specify the location of the `.cpm` file.

4. In the *Output File* text box, specify the name and the location of the file that will contain the schematic information in the EDIF 300 format.

   To know more about the options in the *Schematic Writer* tabbed page, see Appendix A, "Schematic Writer."

Using Schematic Writer from the Command Line

The syntax for invoking the schematic writer from the command line is:

```
c2esch -proj <name of the cpm file> -out_file <name_of_the_output_file>
```

Schematic Writer Output Files

In addition to a translated design data file, with input objects mapped to output objects, EDIF 300 Writer produces log messages. These messages are written to a log file and also displayed on the screen.

- When you use *Schematic Writer* to run a translation, the software writes the log messages to the `edbconfig.log` file in `<your_working_directory>/temp`.
- The translation file is created at the specified location. By default, the output file name is `c2esch.edif` and it is created in your current working directory.
- Along with the translation file, another file, `<output_filename>.lst`, is created in the same location as the translation file.

EDIF 300 Writer does not produce any output file if errors occur during translation. However, it does generate an output file if recoverable problems occur during translation. If warning messages appear in the log file, check the output file before you use it.

Functional Description

This section lists the mapping of Allegro Design Entry HDL objects to EDIF 300.

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Property	Property (property <propName> (string/integer ..) (nameInformation (primaryName ...))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Library	Library (library <libname> (libraryHeader (edifLevel 0) (nameCaseSensitivity...) (technology ...)) (physicalScaling (schematicUnits (setAngle (unitRef DEGS))) (figureGroup .. (nameInformation (primaryName ..)) (pathWidth ..) (displayAttributes (textHeight ..) (color ..) (fontRef .. .. ..))))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Cell	Cell (cell <cellName> (cellHeader (nameInformation... (property ... ... (cluster <clusterName> (interface (interfaceUnits) (designator "") (port <portName> (<portDirection>) ...

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Instance	Instance (instance <instance name> (clusterRef <cluster name> (cellRef <cell name> (libraryRef <library name> (instanceWidth <iteration>) (nameInformation (primaryName ..)) (designator ..) {(property ..)| (string ..) (nameInformation (primaryName))))} (instancePortAttributes .. (designator ..) (property .. (string ..)))) ....

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Note	Note (label) (annotate <string/name> (display (figureGroupOverride <layerName> (displayAttributes (textHeight ..) ...
Signal	Signal (signal <sigName> (signalJoined (portInstanceRef ..)| ... (globalPortRef ..)| (nameInformation (primaryName ...)))
Signal_Interface	(signal <sigName> (signalJoined (portInstanceRef ..) .. (portRef ..)) (nameInformation (primaryName ..))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
FLAG	schematicOffPageConnectorImplementation (schematicOffPageConnectorImplementation <name> (schematicOffPageConnectorTemplateRef ... (libraryRef <library> (transform ...))) (nameInformation (primaryName ..)) (property .. (string ..) (primaryName ..))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Synonym	Synonym (schematicRipperTemplate <name> (schematicTemplateHeader (schematicUnits (schematicMetric (setDistance (unitRef ..)) (property <propName> ... (nameInformation (primaryName ..)) (figure .. ...

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Alias (signal a is aliased to signal b)	(signal a (signalJoined (portInstanceRef .. (instanceRef .. (nameInformation (primaryName "a" (signalGroup b (signalList (signalRef a)) (nameInformation (primaryName "b" (nameStructure (complexName "b")))))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Drawing/Define	Drawing/Define (schematicFigureMacro <name> (schematicTemplateHeader (schematicUnits (schematicMetric ... (property ... ... (nameInformation primaryName ...)) (figure .. .. (annotate "DRAWING" (display (figureGroupOverride NOTE_LAYER (displayAttributes ... ... (It comes as (pageCommentGraphics (schematicComplexFigure (schematicFigureMacroRef ... (libraryRef STANDARD)) on instantiation)

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Page	Page (schematicImplementation (page <pageCount> (pageHeader (nameInformation (primaryName ..)) (property ORIGUNITS (string ..) (owner ..) (property ORIGHSGRID (integer ..) (owner ..) (originalBoundingBox (rectangle (pt ..) (pt ..))) (pageBorder (pageBorderTemplateRef <type_page> (libraryRef STANDARD))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
Page Border	Page Border (pageBorderTemplate <type_page> (schematicTemplateHeader (schematicUnits (schematicMetric (setDistance (unitRef ..)) (noHotspotGrid) .. (property ORIGUNITS (string ..)) (property .. .. (nameInformation (primaryName <page_type>)) (originalBoundingBox (rectangle (pt ..)(pt ..)) (property COMMENT_BODY(string ..)(nameInformation (primaryName "COMMENT_BODY"))) (usableArea (rectangle(pt ..)(pt ..)) (figure (figureGroupOverride BODY_LAYER (pathWidth ..)) (path (pointList(pt ..(pt ..))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
SLASH	SLASH (schematicFigureMacro SLASH_1 (schematicTemplateHeader (schematicUnits (schematicMetric (setDistance (unitRef UNIT0)) .. .. (property .. .. (nameInformation (primaryName "SLASH_1")) (originalBoundingBox (rectangle pointlist .. .. (It is intantiated as (pageCommentGraphics (schematicComplexFigure (schematicFigureMacroRef SLASH_1 (libraryRef STANDARD)) ...)

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
BITTAP	BIT TAP (schematicRipperTemplate <BIT TAP_2> (schematicTemplateHeader (schematicUnits (schematicMetric (setDistance .. (property .. (nameInformation (primaryName "BIT TAP_2") (originalBoundingBox (rectangle pointlist (property BIT (string ..) (nameInformation (primaryName "BIT"))) (property BODY_TYPE (string "PLUMBING") (nameInformation (primaryName "BODY_TYPE"))))

Table 3-1 Map Table for Allegro Design Entry HDL Objects into EDIF 300

Allegro Design Entry HDL Object	EDIF 300
	(ripperHotspot A_(hotspot (pt .. ..)) ((ripperHotspot B_(hotspot (pt .. ..)) (figure .. (path (pointlist ... ((propertyDisplay BIT (display (figureGroupOverride PROPERTY_LAYER (displayAttribute ...

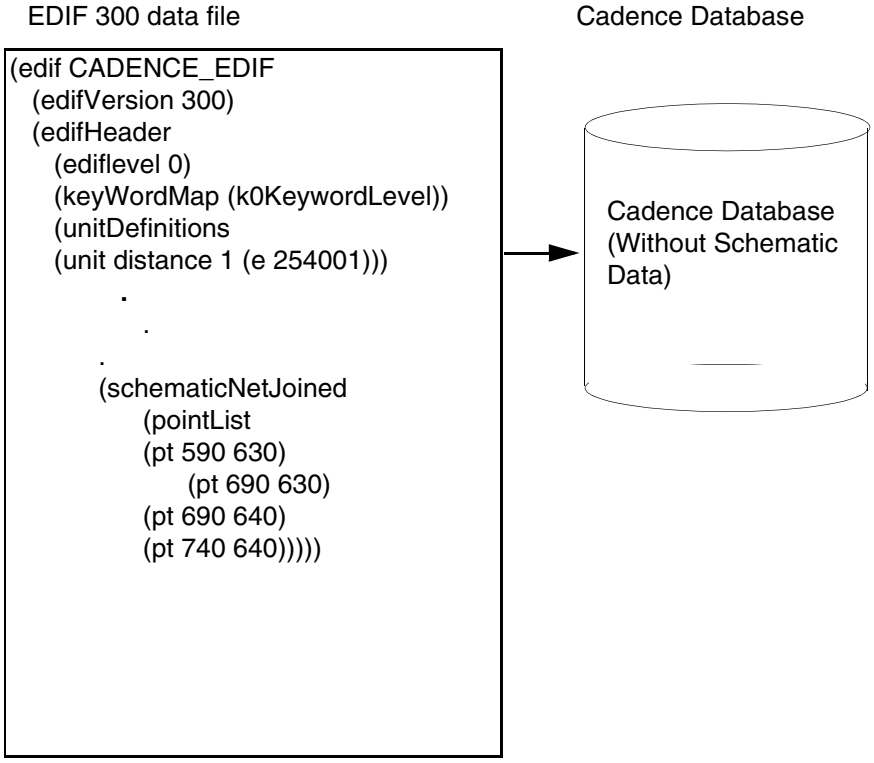
4

Netlist Reader – Importing EDIF 300 Connectivity Data

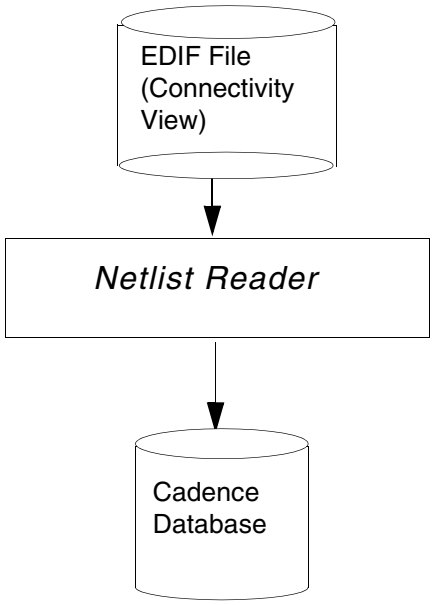
This chapter discusses the following:

Overview

The EDIF 300 *Netlist Reader* lets you translate a netlist containing information about the format view of design data files from the EDIF 300 format into the Cadence database. When you use *Netlist Reader*, the graphical representation of the schematic is lost and only structure and connectivity information is preserved. Therefore, *Netlist Reader* should be used when you have EDIF 300 files that contain only logical connectivity or connectivity structure information.



The following figure shows all the information needed to translate an EDIF 300 file into a Cadence database file.



Using Netlist Reader

You can invoke EDIF 300 *Netlist Reader* either from Project Manager or from an operating-system prompt.

Using Netlist Reader from Project Manager

To invoke *Netlist Reader* of EDIF 300 from Project Manager,

1. Choose *Tools > EDIF 300*.

   The *EDIF 300* dialog box appears.

2. Select the *Netlist Reader* tab.

3. In the *Netlist Reader* tabbed page, specify the EDIF source file and the name of the output directory. To know more about the options in the *Netlist Reader* tabbed page, see Appendix A, “Netlist Reader.”

4. Click *Run* to start the translation.

Using Netlist Reader from the Command Line

Netlist Reader of EDIF 300 can also be invoked in the batch mode from the command line. The syntax for invoking the Netlist Reader from the command line is given below:

```
e2conn -proj <project_file> [-out_file <out_file>]
```

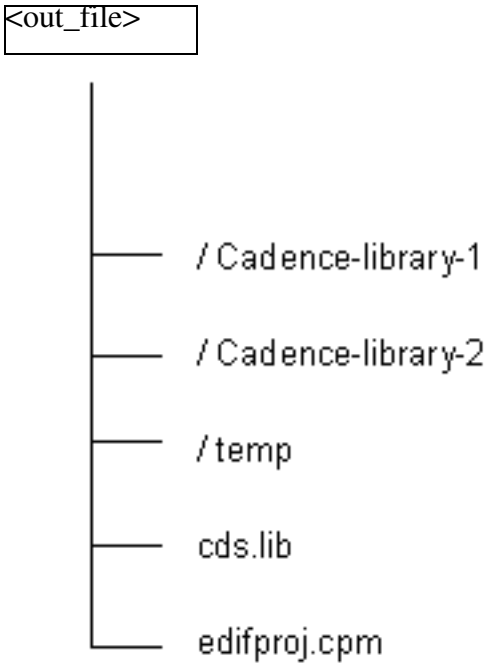
Where

project_file	Specifies the location of the `edif` file
`<out_file>`	Specifies the location and the name of the directory in which you want to save the Allegro Design Entry HDL design.

Netlist Reader Output Files

EDIF 300 Netlist Reader creates the following files:

- Design directory in line with the Allegro Design Entry HDL architecture (`lib:cell:view`).



  Under the design library, you have a cell with the design name that was specified in the source file. This cell has multiple views, such as `sym_1`, `sch_1`, and `cfg_package`.

- Log file, `e2conn.log`, which is placed in the `temp` folder of the design directory. This file contains the warnings and messages that were generated during translation.

Note: If you open the `.cpm` file generated using Netlist Reader, in Allegro Design Entry HDL, you will not be able to see any schematic. This is because EDIF 300 Netlist Reader reads only the connectivity information.

5

Netlist Writer – Creating EDIF 300 Connectivity Data

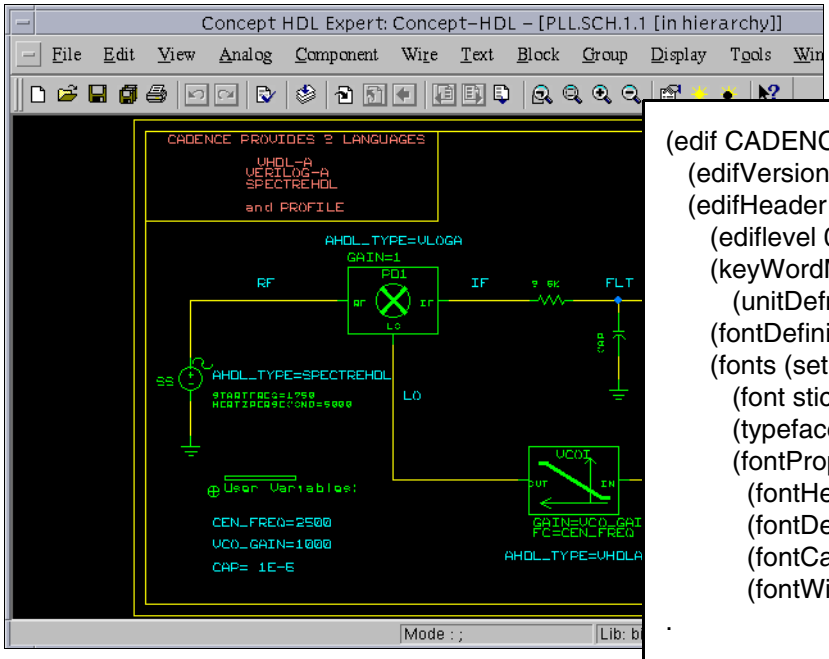
This chapter discusses the following:

Overview

Allegro Design Entry HDL EDIF 300 Netlist Writer lets you translate data from the Cadence database into the EDIF 300 format. When you use Netlist Writer to translate a cellview, the resulting EDIF 300 file contains only the connectivity information for that cellview, but not graphical information about the schematic.

The EDIF 300 file generated using the Netlist Writer can be used as an input to third-party simulators for simulating the design.

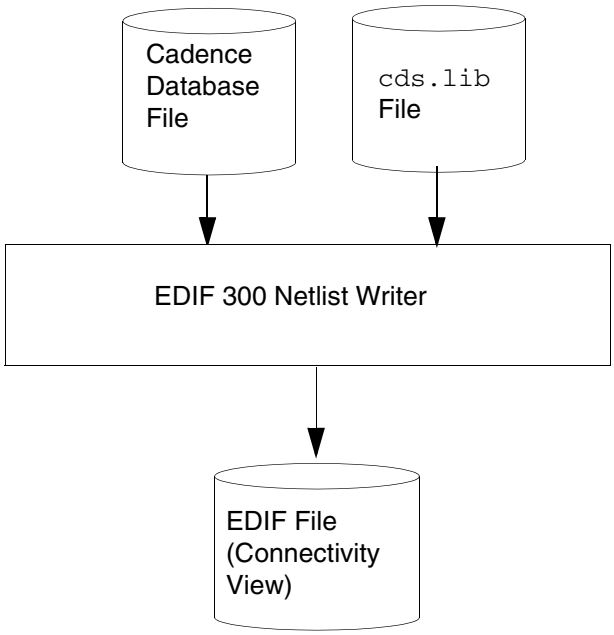
Allegro Design Entry HDL Schematic Data (Contains Graphical and Connectivity Information)



EDIF 300 Data File (Contains Only Connectivity Information)

```
(edif CADENCE_EDIF
  (edifVersion 300)
  (edifHeader
    (ediflevel 0)
    (keyWordMap (k0KeywordLevel))
      (unitDefns    (unit distance 1 (e 254001)))
    (fontDefinitions
    (fonts (setDistance (unitRef distance))
      (font stick
      (typeface "stick")
      (fontProportions
        (fontHeight 10)
        (fontDescent 2)
        (fontCapitalHeight 8)
        (fontWidth 5)))))
.
        .
          .
        (schematicNetJoined
         (pointList
```

The following figure shows all the information needed to translate a Cadence database file into an EDIF 300 file.



Using EDIF 300 Netlist Writer

You can use Netlist Writer of EDIF 300 Writer either from Project Manager or from the command line.

Note: Before translation, you must check and save the schematic in Allegro Design Entry HDL, to generate connectivity information. For more information, see *Allegro Design Entry HDL User Guide*.

Using Netlist Writer from the Command Line

The syntax for running Netlist Writer in the batch mode is:

```
c2econn -proj <project_file> [-out_file <out_file>] [-oversights] [-warnings]
[-single_node_nets] [-hier | -flat]
```

Where

project_file	Specifies the location of the `cpm` file

`<out_file>`	Specifies the name and location of the output EDIF 300 file.
`-oversights`	This option is not functional. This is an optional argument. Use it only if you want the oversights to be displayed.
`-warnings`	This option is not functional. This is an optional argument. Use it only if you want the warnings to be displayed.
`-single_node_nets`	This option is not functional. This is an optional argument. Use it only if you want a net with only one node to be left as a single-node net. By default, all single-node nets are removed.
`-hier | -flat`	Specify the expansion style. You can use either `-hier` to generate the EDIF 300 file in the hierarchical mode, or `-flat` to generate the EDIF 300 file in the flat mode.

Using Netlist Writer from Project Manager

To use Netlist Writer of Allegro Design Entry HDL EDIF 300 from Project Manager:

1. Choose *Tools > EDIF 300*.

   The *EDIF 300* dialog box appears.

2. Select the *Netlist Writer* tab.

3. In the *Netlist Writer* tabbed page, specify the location of the EDIF source file and the name of the output directory. To know more about the options in the *Netlist Writer* tabbed page, see Appendix A, “Netlist Writer.”

4. Click *Run* to start the translation.

   Any error or warning messages are printed to the screen, and the EDIF 300 file is created in the run directory.

The warning and error messages are also logged in `c2econn.log` in your current working directory.

Netlist Writer Output Files

In addition to a translated design data file representing connectivity information, Allegro Design Entry HDL EDIF 300 Netlist Writer produces log messages. The messages are written either to a file or to the screen, depending on how you started the Netlist Writer.

When you use the *Netlist Writer* to run a translation, log messages are written to the `c2econn.log` file in your current working directory.

When the *Netlist Writer* is used from the command line, log messages are displayed on your screen.

The Netlist Writer reports two types of information in the message file:

- An error message indicates that EDIF 300 Writer had problems processing an EDIF 300 construct.
- A warning message indicates that EDIF 300 Writer found a recoverable problem.

The Netlist Writer does not produce any output file if errors occur during translation. However, it does generate an output file if a warning is generated during translation. If warning messages appear in the `eo3conn.log` file, check the output file before you use it.

A

Dialog Box Reference

This section provides detailed explanations for each option in different tabbed pages of the *EDIF 3 0 0* dialog box.

Schematic Reader

This tab is used to create a design directory from an EDIF 300 file. The design directory is created according to the Allegro Design Entry HDL (`lib:cell:view`) architecture.

EDIF Source File	Specify the location of the EDIF 300 file that is to be imported to Allegro Design Entry HDL.
Output Directory	Specify the location where the Allegro Design Entry HDL design is to be created.

Overwrite Library	Select this option if you want that all the existing libraries with the same name should be overwritten when Allegro Design Entry HDL design is created. When selected, this option overwrites only the cells within a library and not the entire library. For example, consider that the *Output Directory* is specified as `/home/test` and the `test` folder has the `lsttl` library with two cells: `ls08` and `ls04`. While generating schematic from an EDIF 300 file if the *Overwrite Library* option is not selected, the `lsttl` library will not be generated. But, if the option is selected, cells other than `ls08` and `ls04` that are used in the schematic will also get added to the `lsttl` library.
Overwrite Cells	Select this option when you want the cells in an existing library to be overwritten while importing schematic data from an EDIF 300 file. By default, this option is selected.
Generate Flags	Select this option to generate Allegro Design Entry HDL-styled flag bodies for each of the interface signals. By default, this option is not selected. When selected, flags will be generated on separate pages of the design.

Properties to Ignore	Specify the properties that should be ignored while importing a design into Allegro Design Entry HDL. By default, all properties are preserved.
Instance Name	Select the way in which you want to put the path property in the Allegro Design Entry HDL design. The setting you choose determines the translation of instance names in the EDIF file.
Off	When you specify OFF, instance names are not translated.
On	When you specify ON, the Allegro Design Entry HDL property PATH is attached to each part and each part's EDIF instance name becomes the value for its PATH property.
Default	When you specify DEFAULT, the PATH property of each part is set by default as in Allegro Design Entry HDL.
Run	Click to start the process of importing schematic data from the EDIF 300 file to Allegro Design Entry HDL.

Schematic Writer

Schematic Writer is used to convert a Allegro Design Entry HDL design into the EDIF 300 format.

Project File	Specify the location of the `.cpm` file of the schematic design to be translated into the EDIF 300 format.
Browse...	Use this button to navigate to the `.cpm` file.
Output File	Specify the name and location of the EDIF 300 file to be generated after the Allegro Design Entry HDL design is exported to the EDIF 300 format.
Browse...	Use this button to navigate to the `.cpm` file.
Run	Click to start the process of importing schematic data from the EDIF 300 file to Allegro Design Entry HDL.

Netlist Reader

Netlist Reader is used to create a design directory from an EDIF 300 file. The design directory is created according to the Allegro Design Entry HDL (`lib:cell:view`) architecture. The Allegro Design Entry HDL design created using a Netlist Reader provides only the connectivity information. For example, information, such as libraries used in the schematic design and how components are connected, will be available, but data related to the

schematic is not available. Therefore, after the process of importing EDIF 300 data to Allegro Design Entry HDL is completed, you will not get a schematic.

EDIF Source File	Specify the location of the EDIF 300 file that is to be imported to Allegro Design Entry HDL.
Output Directory	Specify the location where the Allegro Design Entry HDL design is to be created.
Overwrite Library	Select this option if you want that all the existing libraries with the same name should be overwritten when the Allegro Design Entry HDL design is created. When selected, this option overwrites only the cells within a library and not the entire library. For example, consider that *Output Directory* is specified as `/home/test` and the `test` folder has `lsttl` library with two cells: `ls08` and `ls04`. While generating schematic from an EDIF 300 file, if the *Overwrite Library* option is not selected, the `lsttl` library will not be generated. But, if the option is selected, cells other than `ls08` and `ls04` that are used in the schematic will also get added to the `lsttl` library.
Overwrite Cells	Select this option when you want the cells in an existing library to be overwritten while importing schematic data from an EDIF 300 file. By default, this option is selected.

Generate Flags	Select this option to generate Allegro Design Entry HDL-styled flag bodies for each of the interface signals. By default, this option is not selected. When selected, flags will be generated on separate pages of the design.
Properties to Ignore	Specify the properties that should be ignored while importing a design into Allegro Design Entry HDL. By default, all properties are preserved.
Run	Click to start the process of importing schematic data from the EDIF 300 file to Allegro Design Entry HDL.

Netlist Writer

Netlist Writer is used to translate schematic data from Allegro Design Entry HDL into the EDIF 300 format. When you use Netlist Writer, the resulting EDIF 300 file contains only connectivity or netlisting information for the schematic design. The graphical information is not included in the EDIF 300 file.

Project File	Specify the location of the `.cpm` file of the schematic design to be translated into the EDIF 300 format.
Browse...	Use this button to navigate to the `.cpm` file.

Output File	Specify the name and location of the EDIF 300 file to be generated after the Allegro Design Entry HDL design is translated into the EDIF 300 format.
Browse...	Use this button to navigate to the `.cpm` file.
Oversights	This option is not functional and will be phased out in the coming releases. The Allegro Design Entry HDL EDIF 300 Netlist Writer generates an EDIF 300 netlist using default values. Select this to display oversights. The OVERSIGHTS directive sets the value of the global variable `KV_oversight_output` to TRUE or FALSE. Oversight message are displayed only when `KV_oversight_output` is set to TRUE. A design may work even without oversights messages being fixed.
Warnings	This option is nonfunctional will be phased out in the coming releases. Select this to display warning messages on the screen and in the log file.

Single Node Net	This check box does not work. It will be phased out in the coming releases. Irrespective of whether or not the check box is selected, the output EDIF 300 file does not change. Select this when you want a net with only one node to be left as a single-node net.
Expansion Style	Specifies the expansion style for the EDIF 300 file.
Hierarchical	Select this to generate an EDIF 300 file in the hierarchical mode.
Flat	Select this to generate an EDIF 300 file in the nonhierarchical or flat mode.
Run	Click to start the process of importing schematic data from the EDIF 300 file to Allegro Design Entry HDL.

B

Error Messages and Limitations

Allegro Design Entry HDL EDIF 300 errors can be classified as the following types:

- EDIF Reader Errors
- EDIF Writer Errors

EDIF Reader Errors

If the input EDIF file contains any syntactic or semantic errors, the system will display one of the following error messages:

- No EDIF source file specified. Exiting.
- Empty file given as input! Exiting!!
- A `cds.lib` file exists in the output directory. It will be overwritten.
- Library *<lib_name>* is defined as external file in your edif source. You will need to manually edit the `cds.lib` file created to include its definition.

EDIF Writer Errors

If the problems with the design cause the Allegro Design Entry HDL EDIF 300 Writer to stop processing, the system will display one of the following error messages:

- *<output directory>* already has a `cds.lib`. This will be overwritten. Do you want to continue?
- *<output directory>* failed to create!

Allegro Design Entry HDL EDIF 300 Translator Limitations

This sections lists various known limitations and problems for Allegro Design Entry HDL EDIF 300 translators.

Problems after Translation

- When you write a Allegro Design Entry HDL design into EDIF using `c2esch` and read it using `e2cSch`, the visibility of attributes is lost.

  You can change the visibility manually. To do this, use the *Attributes* form in Allegro Design Entry HDL.

- When you use Allegro Design Entry HDL EDIF 300 translators, some of the properties are lost.

  - The Allegro Design Entry HDL command `zoom fit` does not work on a design after a round-trip.
  - The size value of `slash` (plumbing body) is always reset to `1` in a schematic round-trip.
  - When a Allegro Design Entry HDL design is written using `c2eSch` and read using `e2cSch`, the size of the font changes. This is only a display problem and no connectivity is lost.
  - When a Allegro Design Entry HDL design is written using `c2eSch` and read using `e2cSch`, the line style changes. Again, this is only a display problem and connectivity information is not lost.
  - Another known display problem is that when a Allegro Design Entry HDL design is written using `c2eSch` and is read using `e2cSch`, the justification of text changes.

- During the translation from Allegro Design Entry HDL to EDIF 300 files, BN properties attached to the tap bodies are lost.

  Tap bodies, such as `CTAP,` have a property `BN` attached to their pins. The `BN` property is used to specify the bit tapped when `ctap` is instantiated in a design. In EDIF 300, all these tap bodies get translated to ripper templates. Ripper templates cannot have properties attached to their hotspots. Therefore, the generated EDIF 300 file does not have this information about the tapped bit number. Without this `BN` property, HDL-Direct is not able to correctly netlist the design. Allegro Design Entry HDL EDIF 300 Reader also fails to process this information.

- On Windows NT, Allegro Design Entry HDL EDIF 300 Schematic Reader is unable to create a schematic from the EDIF 300 file that has semantic problems.

- Schematic Writer cannot compile if the design name is in upper case. If a design is created with Project Manager with both design and project name in upper case, the Schematic Writer generates `ERROR (50)`, and stops processing the design.

  To overcome this error, change the design name to lower case in the `.cpm` file.

- In the EDIF 300 UI for Schematic Writer, the default path for the project file (`.cpm`) changes when you change the path for the output file.

  To avoid this problem, use the *Browse* tab to change the path for the project file.

- If an EDIF file generated using the `-flat` option of Allegro Design Entry HDL EDIF 300 Netlist Writer is given as an input to the Netlist Reader, it does not create a proper `verilog.v` file.

- After completing a Allegro Design Entry HDL EDIF 300 Reader or Writer run, and while viewing the log file, the EDIF UI may hang. This occurs only on Solaris. You will have to kill all related processes and re-start Allegro Design Entry HDL EDIF 300 translators.

- Allegro Design Entry HDL EDIF 300 UI may crash on Windows 2000 immediately after the EDIF 300 netlist has been generated, though the tool works properly in the batch mode.

Problems after a Schematic Round-Trip

- After a Schematic Writer and Schematic Reader round-trip, pin names for symbols seem to be disassociated. They are not displayed close to their associated pins on the symbol.

- A 90-degree rotated text string in Allegro Design Entry HDL loses the rotation information after a round-trip and appears as unrotated text. This is a limitation of Allegro Design Entry HDL EDIF 300 Reader.

- The Standard library symbol `DRAWING` is missing in the round-trip solution. This is a limitation of Schematic Writer.

- Symbols for components with vectored pins is not proper. If the symbol for a component has vector pins, such as `D<0>`, `D<1>`, `D<2>`, and `D<3>`, after a round-trip, the EDIF schematic shows each of the input pins as `D<3..0>` and the output pins as `Q<3..0>`.

Tool Limitations

- Allegro Design Entry HDL EDIF 300 translators do not support split parts. Therefore, if your design has two parts in a split group, in the EDIF 300 netlist generated using Allegro Design Entry HDL EDIF 300, they will appear as two parts and not one.

- EDIF 300 Reader cannot handle case-sensitive cell names.This problem occurs because Allegro Design Entry HDL does not support such case-sensitivity with respect to cell names.
- The EDIF 300 interface cannot be used to migrate a Allegro Design Entry HDL schematic to OrCAD Capture.
- Netlist Reader does not read flag or plumbing properties. This is because these are not present in logical connectivity.
- Symbols with the property `COMMENT_BODY=TRUE` is treated as a page border. This creates problems because EDIF allows only one page border per page.
- An EDIF 300 netlist generated after translating a multipage Allegro Design Entry HDL design is not correct. For a multipage design that contains an asymmetrical part, the Netlist Writer generates the EDIF 300 netlist only for page `1`. When the netlist reader is executed on this EDIF 300 netlist, it only generates the `page1.csv` file. Also, the Netlist Writer gives a message regarding the non availability of page `2` and onwards nets.
- Allegro Design Entry HDL EDIF 300 Netlist Writer fails when run from the UI with options set. The Netlist Writer fails when it is run from Project Manager. The following message is displayed:

  ```
  ERROR(210) Library/Cell path not found
  Path to Library project_lib not found
  ```

  You can run the EDIF300 netlist writer from the command line with the complete options set.
- The Netlist Writer does not support different versions of symbols. It uses only the first version of a symbol. A problem may arise if you have multiple versions of symbols in your libraries.
- The EDIF300 interface does not support vectored nets having negative indexes.
- If you have vectored buses in your design, EDIF 300 will generate incorrect syntax and range field. For example, if you have a bus `A<5..0>` in your design, the valid bits are `A<5>`, `A<4>`, `A<3>`, `A<2>`, `A<1>`, and `A<0>`. But, it may so happen that the generated EDIF file may have bit names that lie outside this range `5..0`. For example, you may have a bit named `A<8>`.
- There are multiple representations of vector pins in the EDIF 300 netlist that is generated by Allegro Design Entry HDL EDIF 300 Writer (Schematic and Netlist). For a vector pin in the symbol, the pin designator comes only for a single pin. For example in IC `LS191`, the vector pin `D<3..0>` represents four pins, `D<3>`, `D<2>`, `D<1>`, and `D<0>`. But, the netlist shows the pin designator for only one pin D<3>.

  The multiple representations for a vector signal, `A<3..0>`, are shown below:

First representation:

```
(signalGroup A_603_46_460_62
(signalList
(signalRef A_603_62)
(signalRef A_602_62)
(signalRef A_601_62)
(signalRef A_600_62))
```

Second representation:

```
(nameInformation
(primaryName "A<3..0>"
(nameStructure
(complexName
"A"
(nameDimension
(nameDimensionStructure
(sequence 3 0 (step 1)))
```

Third representation:

```
(signal A_603_62
(signalJoined
(portInstanceRef D_603_62
(instanceRef p1_I1)))
(nameInformation
(primaryName "A<3>"
(nameStructure
(complexName
"A"
(nameDimension
(nameDimensionStructure
3)))))))
(signal A_602_62
(signalJoined
(portInstanceRef D_602_62
(instanceRef p1_I1)))
(nameInformation
(primaryName "A<2>"
(nameStructure
(complexName
"A"
(nameDimension
```

```
(nameDimensionStructure
2)))))))
(signal A_601_62
(signalJoined
(portInstanceRef D_601_62
(instanceRef p1_I1)))
(nameInformation
(primaryName "A<1>"
(nameStructure
(complexName
"A"
(nameDimension
(nameDimensionStructure
1)))))))
(signal A_600_62
(signalJoined
(portInstanceRef D_600_62
(instanceRef p1_I1)))
(nameInformation
(primaryName "A<0>"
(nameStructure
(complexName
"A"
(nameDimension
(nameDimensionStructure
0)))))))
```

- Allegro Design Entry HDL EDIF 300 does not provide support for translating physical information, such as pin numbers.
- The EDIF 300 netlist generated by Allegro Design Entry HDL EDIF 300 writer does not represent implicit pins such as POWER or GND pins.

Index

E

F

N

O

S