cadence™

Allegro® Platform
Properties Reference

Series XL and GXL

Product Version 16.2
November 2008

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

Contents

Allegro Platform Properties

Overview

Properties serve important and varied functions. A property is a name or value pair, attached to certain objects in a design, that conveys information about the design and controls analysis processes.

A number of predefined properties are used by tools in the PCB design flow to record information needed by the Timing Verifier, the Simulator, and the Packager. You can define other properties to convey information to design programs, or pass through to other systems, such as simulators, physical design systems, and so on.

Properties also provide a mechanism for adding physical information to schematics (which represent only a logical design), that can be passed on to the Packager and other physical design systems.

A property has a name and an associated value. You can attach properties to certain objects on any schematic in Allegro Design Entry HDL or Allegro System Architect GXL, to symbols, signals, and pins, and to an entire schematic by attaching them to a DEFINE body or a page border.

Certain properties are used to override a constraint value set on a *Net*, *Net Class, Net Class-Class, Region, Region-Class,* or *Region Class-Class*. The constraint value can be directly-set on a container object or inherited from a referenced Constraint Set.

See

- *General Index of Constraints,* in the Allegro Platform Constraints Reference, for a listing of these constraint overrides.
- Data Sheet descriptions, in the Allegro Platform Constraints Reference, of each constraint override to learn about objects to which these constraint overrides apply.

Definition of a Property

For Allegro PCB Editor, the property name is an identifier—that is, a string of not more than 32 characters that includes letters, digits, and underscores (_) and starts with a letter. Some examples of property names are:

```
SIZE
ROUTE_PRIORITY
MY_PROP_NAME
THE_40TH_NAME
SATURDAY1027
COST_OF_PART
PIN_NUMBER
PART_NAME
```

An underscore is used instead of a space within the property name. Spaces are not allowed in property names because a space delimits a property name from a property value.

A property value is associated with each property name. The property value is a string of up to 255 printable characters for Allegro Design Entry HDL or Allegro System Architect GXL, and 1023 printable characters for the Allegro PCB Editor. Allegro PCB Editor allows all printable characters except the single quote (') and the exclamation point (!). Property values can be empty in Allegro Design Entry HDL and Allegro PCB Editor. However, in Allegro Design Editor GXL, you must not use empty property values. For information on the characters allowed in property names and property values in Allegro Design Entry HDL, Allegro Design Editor GXL, or Part Developer, see the user documentation for the respective products.

Here are some representative property values:

```
1
25oct82 10:31:46.03
(SIZE + 4) / 5 + 35 MOD A
This is a long property value
Value with chars @#$%*()~}{[]><
```

A property always consists of the property name and its associated value. You can attach a property to a component, symbol, pin, or a net. The term component refers to the logical characteristics of a library part. In the earlier releases of Allegro Design Entry HDL, it was called body.

A symbol is the symbolic representation of a library component that you add to your design. This drawing defines the shape, pins, and general properties of the library component.

If a property exists in both Allegro Design Entry HDL or Allegro Design Editor GXL and Allegro PCB Editor and you delete the property in Allegro Design Entry HDL or Allegro Design Editor GXL, the editor deletes the property during logic transfer only if you have not modified the property in the Allegro PCB Editor. As in previous releases, if you add or modify a property in Allegro Design Entry HDL or Allegro Design Editor GXL, the Allegro PCB Editor add or modify the property during logic transfer even if you have modified the property in the Allegro PCB Editor.

Note: Migrating designs that were created before Release 15.0 requires that you run the `netrev` command or import logic once before you can delete properties on subsequent `netrev` operations.

Use the *Edit – Property* menu command (`property edit`) to attach the properties in the Allegro PCB Editor.

See Storing Web Links as the Value of a Property in the *Allegro PCB Editor User Guide: Creating Design Rules* for information on setting a web link for the value of a user-defined property.

Properties and Use

Table 1-1 lists the Allegro platform properties and where they are used. You can add the properties listed under Logic Simulation in Allegro Design Entry HDL or Allegro Design Editor GXL. The following tools use the Logic Simulation properties:

- Allegro AMS Simulator 210
- AWB
- ATDM
- NC-Verilog
- NC-VHDL
- Verilog XL
- Leapfrog
- Verilog
- VHDL

To view a listing of properties by tool or area, refer to these sections in Appendix A:

- Allegro Design Entry HDL Properties
- System Connectivity Manager Properties
- Logic Simulation Properties
- Allegro PCB Editor Properties
- Constraint Manager Properties
- Allegro Package Designer/SiP Digital Architect/SiP Layout Properties

Note: Table 1-1 shows only the relevant properties for Allegro Design Entry HDL and Allegro Design Editor GXL. You can assign all the other properties of the Allegro PCB Editor in Allegro Design Entry HDL. The Constraint Manager properties can also be assigned in Allegro Design Entry HDL.

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
ALIGNED			X			X
ALLOW_CONNECT	X	X				
ALT_SYMBOLS	X	X	X			X
ALT_SYMBOLS_HARD	X	X	X			X
ARTWORK_PREFIX	X	X	X	X	X	X
ARTWORK_SUFFIX	X	X	X	X	X	X
ASSIGN_ROUTE_LAYER			X			X
ASSIGN_TOPOLOGY			X		X	X
AUTO_GENERATED_TERM	For internal use only					
AUTO_RENAME			X			X
BACKDRILL_EXCLUDE		X				
BACKDRILL_MAX_PTH_STUB		X				
BACKDRILL_MIN_PIN_PTH		X				
BACKDRILL_OVERRIDE		X				
BACKDRILL_PRESSFIT_CONNECTOR		X				
BBVIA_SEPARATION			X			
BIDIRECTIONAL	X					
BLOCK=TRUE	X					
BN	X					
BOARD_THICKNESS			X			X
BODY_NAME	X					
BODY_TYPE	X					
BOM_IGNORE	X	X	X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
BOND_PAD			X			X
BONDPAD_TO_BBV_SPACING						X
BONDPAD_TO_BONDPAD_DIFFP_SPC			X			v
BONDPAD_TO_BONDPAD_SPACING						X
BONDPAD_TO_MVIA_SPACING						X
BONDPAD_TO_SHAPE_SPACING						X
BONDPAD_TO_TESTPIN_SPACING						X
BONDPAD_TO_THRUVIA_SPACING						X
BONDPAD_COMP_EDGE	For internal use only					
BOND_WIRE			X			X
BONDWIRE_BONDPAD_SPC	For internal use only					
BONDWIRE_BONDWIRE_CONNECT_SPC	For internal use only					
BONDWIRE_BONDWIRE_SPC	For internal use only					
BONDWIRE_DIAMETER	For internal use only					
BONDWIRE_DIFF_PROFILE_SPC	For internal use only					
BONDWIRE_PIN_SPC	For internal use only					
BUBBLED	X					
BUBBLE_GROUP	X					
BUS_NAME			X			X
CDS_NOT_ON_SYM	X					

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
CDS_PARENT_FUNC_IDENT						X
CDS_PARENT_PIN_SPACING						X
CDS_XNET_NAME			X			X
CLASS	X					
CLIP_DRAW			X			X
CLIP_DRAWING			X			X
CLK_2OUT_MAX			X		X	X
CLK_2OUT_MIN			X		X	X
CLK_SKEW_MAX			X		X	X
CLK_SKEW_MIN			X		X	X
CLOCK_NET			X		X	X
COMMENT	X	X	X		X	X
COMMENT_BODY	X					
COMMENTS			X			X
COMP_NAME	X	X				
COMP_NAME_SUFFIX	X	X				
COMPONENT_WEIGHT			X			X
CONDUCTOR_MATERIAL						X
CONDUCTOR_THICKNESS						X
CPW_DISABLED			X			X
DEGAS_NO_VOID						X
DENSE_COMPONENT			X			X
DEVICE_LABEL	Currently not used					

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
DFA_DEV_CLASS						X
DIELECTRIC_MATERIAL						X
DIELECTRIC_THICKNESS						X
DIFFERENTIAL_PAIR	X		Obsolete for these tools.			
DIFFP_USES_PROPERTIES			X		X	X
DIFFP_COUPLED_MINUS			X		X	X
DIFFP_COUPLED_PLUS			X		X	X
DIFFP_GATHER_CONTROL			X		X	X
DIFFP_MIN_SPACE			X		X	X
DIFFP_NECK_GAP			X		X	X
DIFFP_PHASE_CONTROL	Currently not used					
DIFFP_PHASE_TOL (formerly DIFFP_LENGTH_TOL)			X		X	X
DIFF_PAIR_PINS_NEG						X
DIFF_PAIR_PINS_POS						X
DIFFP_PRIMARY_GAP			X		X	X
DIFFP_UNCOUPLED_LENGTH (formerly DIFFP_2ND_LENGTH)			X		X	X
DIFF_PAIR_PINS_NEG	X					
DIFF_PAIR_PINS_POS	X					
DRC_UNROUTED_MINPROP			X			X
DRC_UNROUTED_RELPROP			X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
DRIVER_TERM_VAL			X			X
DYN_CLEARANCE_OVERSIZE			X			X
DYN_CLEARANCE_TYPE			X			X
DYN_DELETED_ISLAND			X			X
DYN_DO_NOT_VOID			X			X
DYN_FIXED_THERM_WIDTH			X			X
DYN_MAX_THERMAL_CONNS			X			X
DYN_MIN_THERMAL_CONNS			X			X
DYN_OVERSIZE_THERM_WIDTH			X			X
DYN_THERMAL_BEST_FIT			X			X
DYN_THERMAL_CON_TYPE			X			X
ECL		X	X			X
ECL_TEMP			X			X
EDGE_SENS			X			X
ELECTRICAL_CONSTRAINT_SET			X			X
EMC_COMP_TYPE		X	X			X
EMC_CRITICAL_IC		X	X			X
EMC_CRITICAL_NET		X	X			X
EMC_CRITICAL_REGION			X			X
EMC_RUN_DIR			X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
ETCH_TURN_UNDER_PAD			X			
FAMILY	X					
FILLET			X			X
FIRST_INCIDENT			X			X
FIX_ALL			X			X
FIXED		X	X			X
FIXED_T_TOLERANCE			X			X
FP_BOARD_CLEARANCE	For internal use only					
FP_NOTES_NO_EDIT			X			
FP_NOTES_TEXT_BLOCK	For internal use only					
FP_REFDES_TEXT_BLOCK	For internal use only					
FP_ROOM_NAME_TEXT_BLOCK	For internal use only					
GROUP	X	X	X			X
HARD_LOCATION			X			X
HAS_FIXED_SIZE	X					
HDL_CONCAT	X					
HDL_LSBTAP	X					
HDL_MSBTAP	X					
HDL_NOT	X					
HDL_PORT	X					
HDL_POWER	X					
HDL_REPLICATE	X					
HDL_SLASH	X					
HDL_SYNONYM	X					

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
HDL_TAP	X					
HEIGHT	X	X				X
IC_DESIGN_CELL_INSTANCE_NAME		X				X
IC_DESIGN_CELL_MASTER_NAME		X				X
IC_DESIGN_CELL_PIN_NAME		X				X
IC_DESIGN_NET_NAME		X				X
IDF_OTHER_OUTLINE			X			X
IDF_OWNER			X			X
IMPEDANCE_RULE			X			X
INCLUDE_IN_RF_TOPOLOGY	X		X			
INLINE_PIN_VOIDS			X			X
INPUT_LOAD	X					
INSERTION_CODE	Currently not used					
ISRFELEMENT			X			
JEDEC_TYPE	X	X				
J_TEMPERATURE			X			X
LAST_MODIFIED	X					
LAST_PIN_SWAP		X				
LAYERSET_GROUP			X			X
LEAD_DIAMETER			X			X
LEFDEF_SPECIAL_NET			X			X
LIBRARYn				X		

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
LINE_OVERSIZE			X			X
LOAD_TERM_VAL			X			X
LOCATION	X					
LOCKED			X			X
LOGICAL_PATH			X			X
MAKE_BASE	X					
MATERIAL	Obsolete property					
MAX_BOND_LENGTH			X			X
MAX_BVIA_STAGGER			X			X
MAX_LINE_EXIT_ANGLE						X
MAX_POWER_DISSIPATION	x	x	x	x	x	x
MAX_EXPOSED_LENGTH			X			X
MAX_FINAL_SETTLE			X			X
MAX_LINE_EXIT_ANGLE	For internal use only					
MAX_LINE_WIDTH			X			X
MAX_OVERSHOOT			X			X
MAX_PARALLEL (formerly PARALLELISM)			X			X
MAX_PEAK_XTALK (formerly MAX_PEAK_CROSSTALK)			X			X
MAX_POWER_DISSIPATION						X
MAX_SSN			X		X	X
MAX_UNDERSHOOT					X	X*

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
MAX_VIA_COUNT			X		X	X
MAX_XTALK (formerly MAX_CROSSTALK)			X		X	X
MERGE_NC_PINS	X					
MERGE_POWER_PINS	X					
MIN_BOND_LENGTH			X			X
MIN_BVIA_GAP			X			X
MIN_BVIA_STAGGER			X			X
MIN_FIRST_SWITCH			X		X	X
MIN_HOLD			X		X	X
MIN_LINE_WIDTH			X			X
MIN_NECK_WIDTH			X			X
MIN_NOISE_MARGIN			X		X	X
MIN_SETUP			X		X	X
MIN_SHAPE_SIZE			X			X
MODEL_DIR				X		
MODEL_FILE				X		
NC_PINS	X					
NEEDS_NO_SIZE	X					
NET_SCHEDULE			X		X	X
NET_SHORT			X			X
NO_BACKANNOTATE		X	X			X
NO_BACKANNOTATE	X					
NO_DRC		X	X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
NODRC_COMPONENT_BOARD_OVERLAP			X			X
NODRC_ETCH_OUTSIDE_KEEPIN			X			X
NODRC_SYM_SAME_PIN			X			X
NODRC_VIAS_OUTSIDE_KEEPIN			X			X
NO_FILLET		X	X			X
NO_FILLET		X	X			X
NO_IO_CHECK	X					
NO_LOAD_CHECK	X					
NO_LIN2SHAPE_FAT			X			X
NO_PIN_ESCAPE		X	X			X
NO_RAT		X	X			X
NO_REP_PRIM				X		
NO_RIPUP		X	X			X
NO_ROUTE		X	X			X
NO_SHAPE_CONNECT			X			X
NO_SM_COVERAGE_CHECK	Currently not used					
NO_SWAP_COMP			X			X
NO_SWAP_GATE			X			X
NO_SWAP_GATE_EXT			X			X
NO_SWAP_PIN			X			X
NO_TEST		X	X			X
NO_VIA_CONNECT			X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
NO_WIREBOND						X
OK_DANGLE			X			
OK_UNASSIGNED_SHAPE			X			X
OUTPUT_LOAD	X					
OUTPUT_TYPE	X					
PACKAGE_HEIGHT_MAX and PACKAGE_HEIGHT_MIN			X			X
PACK_IGNORE	X	X				
PACK_SHORT	X	X				
PACK_TYPE	X					
PART_NAME	X					
PART_NUMBER	X	X				X
PATH	X	X				
PHYS_DES_PREFIX	X					
PIN_DELAY	X		X		X	X
PIN_DELAY_ENABLED	X		X		X	X
PIN_ESCAPE			X			X
PIN_GROUP	X					
PIN_NAME	X					
PIN_NUMBER	X					
PIN_SIGNAL_MODEL	For internal use only					
PINUSE	X	X	X			X
PLACE_TAG			X			X
PLATING			X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
PN	X					
PNN	X					
PORT_ORDER				X		
POWER_GROUP	X					
POWER_MAX			X			
POWER_OPR			X			
POWER_PINS	X					
PROBE_NUMBER		X	X			X
PROPAGATION_DELAY			X		X	X
PROPAGATION_DELAY_ACTUAL					X	
PULSE_PARAM			X		X	X
RATED_POWER	X	X	X	X	X	X
RATS_FACTOR	Currently not used					
RATSNEST_ SCHEDULE			X		X	X
REF_DES_FOR_ASSIGN			X			X
REF_DES_PATTERN	X					
REGION_NAME			X			X
RELATIVE_PROPAGATION_DELAY			X		X	X
REMOVE		X		X		X*
REUSE_ALT_MODULE	X		X			X
REUSE_ID	For internal use only					
REUSE_INSTANCE	X		X			X
REUSE_MODULE	X		X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
REUSE_NAME	X		X			X
REUSE_PID	For internal use only					
RF_NET	For internal use only					
RF_PIN_MAP	For internal use only					
RF_TLINE			X		X	X
RFELEMENTTYPE			X			
RFPCB_OBJECT			X			
ROOM	X	X	X			X
ROOM_TYPE			X			X
ROTATE	X					
ROUTE_PRIORITY		X	X			X
ROUTE_TO_SHAPE		X				X
SAME_NET			X			X
SAME_NET_XTALK_ENABLED			X			X
SAME_NET_XTALK_ENABLED			X			X
SCHEMATIC_NAME			X			X
SDFDELAYTYPE				X		
SDFFILE				X		
SDFSCALEFACTOR				X		
SDFSCALETYPE				X		
SEC	X					
SEC_TYPE	X					
SHAPE_OVERSIZE			X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
SHIELD_NET		X	X			X
SHIELD_TYPE			X			X
SHORTING_SCHEME			X			X
SIGNAL_MODEL			X			X
SIG_NAME	X					
SIM_BIND_VIEW				X		
SIM_MAP_VIEW		X		X		
SIZE	X	X		X		
SLOTNAME			X			X
SMD_BEST_FIT			X			X
SMD_CLEAR_TYPE			X			X
SMD_MAX_THERMS			X			X
SMD_MIN_THERMS			X			X
SMD_OVERSIZE			X			X
SMD_THERM_CONN			X			X
SMOOTH_MIN_GAP			X			X
SMOOTH_TRIM_CONTROL			X			X
SNAP_VOID_XHATCH			X			X
SOLDER_BALL_HEIGHT			X			X
SOV_CHECK						X
SPIF_CONSTANTS	For internal use only					
SPIF_TURRET	For internal use only					
SPLIT_INST				X		
SPLIT_INST_NAME				X		
STUB_LENGTH			X		X	X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
SUBDESIGN_MASTER	X					
SUBDESIGN_SUFFIX	X					
SUBNET_NAME			X			X
SWAP_GROUP			X			X
SWAP_INFO	X					
SYS_CONFIG_NAME			X			X
TECH	X					
TEMPORARY_PACKAGE_SYMBOL	For internal use only					
TERMINATOR_PACK			X			X
TESTER_GUARDBAND			X			X
TESTPOINT_ALLOW_UNDER			X			X
TESTPOINT_MAX_DENSITY			X			X
TESTPOINT_QUANTITY		X	X			X
TEXT_OVERSIZE			X			X
THERMAL_RELIEF			X			X
THETA_JB	x	x	x	x	x	x
THETA_JC	x	x	x	x	x	x
THICKNESS	Obsolete property					
THRU_BEST_FIT			X			X
THRU_CLEAR_TYPE			X			X
THRU_MAX_THERMS			X			X
THRU_MIN_THERMS			X			X
THRU_OVERSIZE			X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
THRU_THERM_CONN			X			X
TIMING_DELAY_OVERRIDE			X		X	X
TOL		X	X			X
TOPOLOGY_TEMPLATE	Obsolete property					
TOPOLOGY_TEMPLATE_REVISION			X			X
TOTAL_ETCH_LENGTH			X		X	X
TS_ALLOWED			X			X
UNFIXED_PINS	X		X			X
UNKNOWN_LOADING	X					
UNUSED_PADS_IGNORE	X	X	X	X	X	X
USEn				X		
VALUE	X	X	X			X
VER	X					
VERILOG_LIB				X		
VERILOG_MODEL		X		X		
VERILOG_NAME				X		
VERILOG_PORT_NAME	X					X
VERSION_ID	For internal use only					
VHDL_CONCAT	X					
VHDL_INIT				X		
VHDL_MODE	X					
VHDL_MODEL		X		X		
VHDL_NAME				X		
VHDL_SCALAR_TYPE				X		

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
VHDL_SLICE	X					
VHDL_VECTOR_TYPE				X		
VIA_AT_SMD_FIT			X			X
VIA_AT_SMD_THRU			X			X
VIA_CLEAR_TYPE			X			X
VIA_LIST		X	X			X
VIA_MAX_THERMS			X			X
VIA_MIN_THERMS			X			X
VIA_OVERSIZE			X			X
VIA_THERM_CONN			X			X
VIAS_ALLOWED			X			X
VIA_Z_ENABLED			X			X
VLOG_MODE	X					
VLOG_NET_TYPE				X		
VOID_SAME_NET			X			
VOLTAGE		X	X			X
VOLTAGE_SOURCE_PIN			X			X
VOLT_TEMP_MODEL		X				X
WB_LOOP_HEIGHT_GROUP			X			X
WEIGHT		X	X			X
WIREBOND_FINGER_SHAPE						X
WIREBOND_PROFILE_NAME						X
WIRE_LENGTH			X			X

Table 1-1 Properties and Use

Property	Allegro Design Entry HDL	System Connectivity Manager	Allegro PCB Editor	Logic Simulation	Constraint Manager	APD/ SiP Tools
WIREBOND_FINGER_SHAPE						X
WIREBOND_MATERIAL						X
WIREBOND_PROFILE_NAME						X
XHATCH_BORDER_WIDTH			X			X
XR	X					
XTALK_ACTIVE_TIME	X		X		X	X
XTALK_IGNORE_ NETS			X		X	X
XTALK_SENSITIVE_TIME	X		X		X	X
XY	X					

* means APD only, not the SiP tools.

Property Descriptions

ALIGNED

The ALIGNED property, attached to bondpads in APD, prohibits the attached bondpad from going out of alignment when you execute the `move` or `spin` commands. APD sets this property, based on the settings in the *Options* tab of the Control Panel.

ALLOW_CONN_SWAP

The ALLOW_CONN_SWAP property is attached to a component and governs how swaps are handled in the back-to-front flow. You can attach this property using the Property window, or add it in the chips.prt file.

- Not attached: Pin swaps are handled as net swaps for component where the Pin name = Pin number. On all co-design components, always pin swaps are performed.

- TRUE: Enables net swaps on components with this property.
- FALSE: Disables net swaps on components with this property. All swaps are handled as pin swaps.

Pin Swap and Net Swap Matrix

Net swaps have been introduced for BGA type components—where the Pin name is the same as Pin number—to retain the pin name and number consistency.

	Co-Design Components	Components where Pin Name = Pin Number	Components where Pin Name != Pin Number
ALLOW_CONN_SWAP is not present	Pin Swap	Net swap	Pin Swap
ALLOW_CONN_SWAP=TRUE	Net Swap	Net swap	Net swap
ALLOW_CONN_SWAP=FALSE	Pin Swap	Pin swap	Pin swap

Exceptions for Net Swaps

Net swaps are not performed in the following cases:

- For parts that belong to read-only, concept, or reuse blocks.
- If parent differential pin of the swapped have terminations attached.

ALLOW_CONNECT

The ALLOW_CONNECT property, attached to a component, symbol, net, or pin, allows different types of outputs to be connected without producing errors when OUTPUT_TYPE properties are checked. You can use the ALLOW_CONNECT property on a library component or in a logical design. Be sure that the value is set to TRUE.

If you attach the ALLOW_CONNECT property to a net, it applies to all output pins on the net. If you attach the property to a symbol, it applies to all output pins on the symbol. When attached to a pin, the property applies only to the pin to which you attached the property.

ALT_SYMBOLS

The ALT_SYMBOLS property, attached to an assigned component, comes in through the `pstchip.dat` file for Allegro Design Entry HDL or a device file for third-party tools. Alternate symbols cannot be used for an unassigned component. The property is not directly editable in the design editors.

This property lets you specify a list of alternate package symbol names that you can use to substitute the primary package symbol during interactive placement. During component placement, move, or mirror operations, selecting the ALT_SYMBOLS pop-up options allows you to switch to a different symbol (one symbol at a time) through the entire list.

By default, the symbol defined in the component's JEDEC_TYPE can always be placed on either the top or the bottom. The presence of the boolean ALT_SYMBOLS_HARD property on the same component overrides this behavior.

This is the syntax of the PACKAGEPROP record when you use ALT_SYMBOLS:

```
PACKAGEPROP ALT_SYMBOLS '(Subclass:Symbol,...;Subclass:Symbol,...)'
```

Subclass	Either TOP (or T) for top layer, or BOTTOM (or B) for bottom layer. If no subclass is specified, the alternate is legal for both sides.
Symbol	Standard Allegro PCB Editor package symbol name. Separate each symbol with a comma.

Note: You can enter more than one symbol statement by separating each symbol statement with a ; (semicolon). This syntax can also be used in a physical parts table. The entire statement, including the parentheses, must be enclosed in single quotations to be read correctly by the *netin param* program.

The following example shows a device file containing a PACKAGEPROP record for alternate symbols. In this sample file, five alternate symbols are defined for a resistor; the first two are for either TOP or BOTTOM, and the remaining three are for the BOTTOM:

```
CLASS DISCRETE
PACKAGE RES400
PACKAGEPROP   ALT_SYMBOLS   '(RES500,RES800;B:RES400B,RES500B,RES800B)'
PINCOUNT 2
...
END
```

Design Editor Schematic Notes

- The ALT_SYMBOLS property has the following attributes defined in the default `cdsprop.paf` file:

  `inherit(cell)`, `permit(cell)`, `uppercase value`

- This property is now inherited by all components of a hierarchical block of which they form a part.
- The ALT_SYMBOLS property and its values always appear in uppercase in the `pstxprt.dat` and `pstchip.dat` files, irrespective of the case you specify on the schematic instance, part table file (`.ptf`), or `chips.prt` file.

Syntax

- Property is attached to component definitions. Property type is String with the following syntax. Third party device file:

  ```
  PACKAGEPROP ALT_SYMBOLS '(T:DIP20_3;B:SOIC20)'
  PACKAGEPROP ALT_SYMBOLS_HARD
  ```

- Property is not editable in the System Connectivity Manager. Normal method of entering the property into a design is either by the `pstchip.dat` (Cadence Schematic) or the device files (3rd party).

Notes:

- The open and closed parentheses () are mandatory. If you do not add these parentheses, the following error is generated:

  `Error in ALT_SYMBOLS property for device <`*`device_name`*`>:'Encountered an error while parsing alternate symbol.'`

- You may use shorthand notation; T for “TOP”, B for “BOTTOM”.
- The keywords may appear in any order.
- If symbols are not qualified by TOP or BOTTOM keywords, then they can be placed on either the top or the bottom.
- If the ALT_SYMBOLS_HARD property is present, then the symbol defined by the JEDEC_TYPE is restricted to its definition layer in the ALT_SYMBOLS property. If the JEDEC_TYPE symbol does not appear in the ALT_SYMBOLS set, then the default JEDEC_TYPE symbol can be placed on either the top or bottom of the design.

Examples (ALT_SYMBOLS and ALT_SYMBOLS_HARD)

These examples use a JEDEC_TYPE `DIP20` to illustrate the use of ALT_SYMBOLS_HARD. The ALT_SYMBOLS_HARD property has no effect on examples one, two, and three.

1. The ALT_SYMBOLS property does not include the JEDEC_TYPE property and alternate symbols are not side-specific.

   `(SOIC20, DIP20_3)`

   TOP allows: `SOIC20, DIP20_3` and `Jedec_type DIP20`

   BOTTOM allows: `SOIC20, DIP20_3` and `Jedec_type DIP20`

2. The ALT_SYMBOLS property does not include the JEDEC_TYPE property, and alternate symbols are side-specific.

   `(TOP: SOIC20, DIP20_3; BOTTOM: DIP20_3)`

   TOP allows: `DIP20 SOIC20` and `Jedec_type DIP20`

   BOTTOM allows: `DIP20_3` and `Jedec_type DIP20`

3. The ALT_SYMBOLS property includes the JEDEC_TYPE property, and alternate symbols are not side-specific.

   `(DIP20, SOIC20)`

   TOP allows: `DIP20 SOIC20` and `Jedec_type DIP20`

   BOTTOM allows: `DIP20 SOIC20` and `Jedec_type DIP20`

4. The ALT_SYMBOLS property includes the JEDEC_TYPE property, and alternate symbols are side-specific.

ALT_SYMBOLS Property	Side	Placement without ALT_SYMBOLS_HARD	Placement with ALT_SYMBOLS_HARD
`( T:DIP20,DIP20_3,SOIC20 )`	TOP	`DIP20,DIP20_3,SOIC20`	`DIP20,DIP20_3,SOIC20`
	BOTTOM	`JEDEC_TYPE DIP20`	No permissible symbols
`( T:DIP20 )`	TOP	`DIP20`	`DIP20`
	BOTTOM	`JEDEC_TYPE DIP20`	No permissible symbols
`( B:DIP20,DIP20_3,SOIC20 )`	TOP	`JEDEC_TYPE DIP20`	No permissible symbols
	BOTTOM	`DIP20,DIP20_3,SOIC20`	`DIP20,DIP20_3,SOIC20`

(B DIP20)	TOP	JEDEC_TYPE DIP20	No permissible symbols
	BOTTOM	DIP20	DIP20
(T:DIP20,DIP20_3,SOIC20)	TOP	DIP20,DIP20_3	DIP20,DIP20_3
	BOTTOM	DIP20,SOIC20	SOIC20 (no JEDEC_TYPE)
(T:SOIC20;B:DIP20,DIP20_3)	TOP	DIP20,SOIC20	SOIC20 (no JEDEC_TYPE)
	BOTTOM	DIP20,DIP20_3	DIP20,DIP20_3

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

ALT_SYMBOLS_HARD

The ALT_SYMBOLS_HARD property, attached to a component, comes in through a device file for third-party tools or the `pstchip.dat` file for Allegro Design Entry HDL.The property type is boolean. It is used in conjunction with the ALT_SYMBOLS property to restrict the symbol defined by the JEDEC_TYPE to its definition layer specified in the ALT_SYMBOLS property. If the JEDEC_TYPE does not appear as one of the symbols specified in ALT_SYMBOLS, then it is permissible for either side of the design.

Notes:

- This property is not editable in System Connectivity Manager. The normal method of entering the property into a design is either via the `pstchip.dat` (Cadence Schematic) or the device files (3rd party).
- This property is ignored in older versions (pre-15.5.1) of Allegro PCB Editor.

Examples

See Examples (ALT_SYMBOLS and ALT_SYMBOLS_HARD) on page 42.

ARTWORK_PREFIX

The ARTWORK_PREFIX property, attached to a board (design), adds a user-defined, case-sensitive string before generated film filenames, allowing a maximum 250-character filename, such as a part or revision number, which may be useful for larger boards with many layers and numerous artwork films as a result. For example, if a board has a project number of CDS1234, adding a prefix of CDS1234_ creates artwork in the following format:

```
CDS1234_TOP.art

CDS1234_BOTTOM.art

CDS1234_SMASKT.art

CDS1234_SMASKB.art
```

Names must be legal filenames and cannot contain directory names. The value of this property is a string. Although filename affixes of 250 characters are permitted, many operating systems limit filenames to 256 characters (including extensions). Consequently, Cadence recommends film filenames (affixes plus filename plus extensions) be less than 256 characters.

ARTWORK_SUFFIX

The ARTWORK_SUFFIX property, attached to a board (design), appends a user-defined, case-sensitive string after generated film filenames, allowing a maximum 250-character filename, such as a part or revision number, which may be useful for larger boards with many layers and numerous artwork films as a result. For example, if a board has a revision number of Rev-3, adding a suffix of _Rev-3 creates artwork in the following format:

```
TOP_Rev-3.art

BOTTOM_Rev-3.art

SMASKT_Rev-3.art

SMASKB_Rev-3.art.
```

Names must be legal filenames and cannot contain directory names. The value of this property is a string. Although filename affixes of 250 characters are permitted, many operating systems limit filenames to 256 characters (including extensions). Consequently, Cadence recommends film filenames (affixes plus filename plus extensions) be less than 256 characters.

ASSIGN_ROUTE_LAYER

The ASSIGN_ROUTE_LAYER hierarchical property, attached to a pin or net, allows you to guide the `spider router` command when it is performing package routing (a property attached to a pin overrides any value specified for its owning net). A *fixed* property indicates that Spider Route must route this pin out on the specified layer. If there is no property attached or the property value is *free*, the router can use whichever layer is best for routing this connection.

ASSIGN_TOPOLOGY

The ASSIGN_TOPOLOGY property, attached to a net in a third-party schematic package, names the ECset assigned to the net. Once you import the netlist into a drawing, you can open the Constraint Manager and run the *Audit – Topology* command. This uses the ASSIGN_TOPOLOGY property to determine what ECset is to be assigned to each net. This command also has option for allowing you to specify where to find the data for each ECset, that is, from a topology file or from an ECset that is already loaded into the drawing.

AUTO_GENERATED_TERM

The AUTO_GENERATED_TERM is an internal Allegro SI property and is not accessible to the user.

AUTO_RENAME

The AUTO_RENAME property, attached to a reference designator (component), indicates that the Allegro PCB Editor should include the components in the next automatic renaming process. Be sure that you set the value to `TRUE`.

BACKDRILL_EXCLUDE

The BACKDRILL_EXCLUDE property, attached to vias, pins, and symbol instances, prevents backdrilling of the affected pins or vias. When attached to a symbol instance, it applies to all pins of the symbol. Subsequently assigning the via or pin to a net with the BACKDRILL_MAX_PTH_STUB property excludes the via or pin.

BACKDRILL_MAX_PTH_STUB

The BACKDRILL_ MAX_PTH_STUB property, attached to nets, specifies the maximum permitted length of the unused stub portion of the plated thru hole.

To completely remove a stub, specify zero to backdrill to the layer adjacent to the first layer on which the via or pin connects. A maximum stub length error may result, however, if the backdrill passes that contain the layer to which to backdrill is not a permitted layer. Even with a non-0 maximum length, maximum stub length errors may occur.

BACKDRILL_MIN_PIN_PTH

The BACKDRILL_MIN_PIN_PTH property, attached to a symbol or to individual pins, defines the minimum plated hole length that must remain after backdrilling. You may assign it so its value defaults to all symbol pins, or you may apply it to individual pins, when a majority of symbol pins require one length, but exception pins require a different length. If the remaining plated hole length is less than the value specified by this property, no backdrilling occurs for the pin hole.

BACKDRILL_OVERRIDE

The BACKDRILL_OVERRIDE property, attached to a pin or via, mandates backdrilling always occurs from the specified side to the specified layer, even if the result is drilling through a connection layer, or violating the maximum stub length by drilling insufficiently. The property may also force backdrilling of a pin or via normally excluded from backdrilling as a result of being a testpoint on that side.

Syntax

The <*side*> is either TOP or BOTTOM, and <*layer name*> is that of an etch layer and not a layer number.

```
BACKDRILL_OVERRIDE = <side>:<layer name>
```

Example

```
BACKDRILL_OVERRIDE = TOP:LAY4
```

To define overrides for backdrilling from both sides:

```
<side1>:<layer name1>:<side2>:<layer name>
```

```
BACKDRILL_OVERRIDE = TOP:LAY4:BOTTOM:LAY8
```

BACKDRILL_PRESSFIT_CONNECTOR

The BACKDRILL_PRESSFIT_CONNECTOR property, attached to symbols that are pressfit connector components, defines the range between which backdrilling is not allowed from either side of the board when any of its pins are backdrilled. Specified in current user database units, <*depth1*> and <*depth2*> represent where the pressfit pin contacts the board. Therefore plating should never be backdrilled within this range. Individual pressfit connector pins can be overridden with the BACKDRILL_OVERRIDE property to backdrill into the contact range.

Syntax

```
BACKDRILL_PRESSFIT_CONNECTOR = <depth1>:<depth2>
```

Example

```
BACKDRILL_PRESSFIT_CONNECTOR = <80>:<100>
```

BBVIA_SEPARATION

The BBVIA_SEPARATION property, attached to a board, allows setting a value less than the spacing value between two same net blind and buried vias.

BIDIRECTIONAL

The BIDIRECTIONAL property, attached to a pin, indicates that it is both an input and an output pin. A pin is *not* considered bidirectional unless you attach this property. This property is stored in *Pin Section* of the device file (third-party) or the `.pstchip.dat` file (Cadence format).

BLOCK=TRUE

The BLOCK=TRUE property, attached to a symbol, distinguishes a symbol as being a hierarchical block, and not just a part symbol.

BN

The BN (bit number) property, attached to the pin of a tap symbol, specifies the bit on a bus that you want to tap. The tap symbol is attached between the bus and the net to be tapped off the bus. The value is `INTEGER` or a range, for example, <1..0>.

BOARD_THICKNESS

The BOARD_THICKNESS property, generated automatically by Allegro PCB Editor and attached to a board (design), specifies the board (or layout) thickness. Allegro PCB Editor compute the thickness by adding all layer thicknesses in the cross-section. Cadence recommends that you do not assign the BOARD_THICKNESS property.

BODY_NAME

The BODY_NAME property, attached to a symbol (specified in the `chips.prt` file), defines the logical cell name associated with the defined physical part.

BODY_TYPE

The BODY_TYPE property, attached to a symbol, specifies the symbol type. It can have the following values:

`COMMENT`	The symbol is a comment and should be ignored. This property replaces the previous COMMENT_BODY property.
`FLAG_BODY`	The symbol indicates an I/O signal. It is used by Allegro Design Entry HDL to process module interface signals.
`PLUMBING`	The symbol is a plumbing symbol, used in the schematic to show connectivity between different schematic pages. It is ignored by Packager-XL and is not written to the `.pstchip` file. Examples of the plumbing symbol are `MERGER`, and `SYNONYM`. These symbols have the BODY_TYPE property set as PLUMBING.

BOM_IGNORE

The BOM_IGNORE, a reserved property, is attached to component instances. When you set this boolean property to TRUE, the component instance does not appear in the Bill of Materials report.

BOND_PAD

The BOND_PAD property, generated automatically by APD, is attached to bond pads created during the automatic or interactive wire bond process.

BONDPAD_TO_BBV_SPACING

The BONDPAD_TO_BBV_SPACING property, when attached to a Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region overrides the constraint spacing specified between a bond finger and a blind or buried via.

BONDPAD_TO_BONDPAD_DIFFP_SPC

The BONDPAD_TO_BONDPAD_DIFFP_SPC property holds the value for the design-level bond finger-to-bond finger differential pair constraint.

BONDPAD_TO_BONDPAD_SPACING

The BONDPAD_TO_BONDPAD_SPACING property, when attached to a Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region overrides the constraint spacing specified between a bond finger and another bond finger.

BONDPAD_TO_MVIA_SPACING

The BONDPAD_TO_MVIA_SPACING property, when attached to a Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region overrides the constraint spacing specified between a bond finger and a microvia.

BONDPAD_TO_SHAPE_SPACING

The BONDPAD_TO_SHAPE_SPACING property, when attached to a Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region overrides the constraint spacing specified between a bond finger and a shape.

BONDPAD_TO_TESTPIN_SPACING

The BONDPAD_TO_TESTPIN_SPACING property, when attached to a Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region overrides the constraint spacing specified between a bond finger and a test pin.

BONDPAD _TO_THRUVIA_SPACING

The BONDPAD_TO_THRUVIA_SPACING property, when attached to a Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region overrides the constraint spacing specified between a bond finger and a thru via.

BONDPAD_COMP_EDGE

Internal use only.

BOND_WIRE

The BOND_WIRE property, attached to clines, indicates that the cline is a wirebond (3D gold wire connecting a die pad to the bondpad on the package substrate) instead of a standard cline used for 2D routing on a given substrate layer.

BONDWIRE_BONDPAD_SPC

Internal use only.

BONDWIRE_BONDWIRE_CONNECT_SPC

Internal use only.

BONDWIRE_BONDWIRE_SPC

Internal use only.

BONDWIRE_DIAMETER

Internal use only.

BONDWIRE_DIFF_PROFILE_SPC

Internal use only.

BONDWIRE_PIN_SPC

Internal use only.

BUBBLED

The BUBBLED property, automatically attached to bubbled pins, indicates that only low-asserted signals can connect to them.

Note: Do not assign this property; it is automatically generated by the tool.

BUBBLE_GROUP

The BUBBLE_GROUP property, attached to the origin of a symbol, is used on the symbol drawing to indicate which pins must bubble simultaneously due to their logical association with each other. Each BUBBLE_GROUP property defines one bubble group.

BUS_NAME

The BUS_NAME property, attached to a net, indicates that during interactive and automatic routing, the tool should treat the specified net as a bus. Allegro Design Entry HDL automatically adds this property for signals identified as part of a bus. You can also attach this property interactively. The value is a string.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

CDS_NOT_ON_SYM

The CDS_NOT_ON_SYM property is assigned by Allegro Design Entry HDL in the `viewprps.prp` file, to those pins that belong to the split parts that are not initiated on the schematic, and therefore seem to be missing pins. When Packager-XL reads this property, it does not flag errors for missing pins.

For example, a part has 40 pins. It is split into four symbols with ten pins on each symbol. When one of these split symbols is instantiated on the schematic, then Allegro Design Entry HDL assigns the CDS_NOT_ON_SYM property for the remaining 30 pins, which are available on other symbols of the split part in the `viewprps.prp` file. Packager-XL matches the pin entries in the `chips.prt` file to the pin entries in the entity for the part. In this example, Packager-XL finds the CDS_NOT_ON_SYM property on pins that are present in the `chips.prt` file, but not on the symbol, and does not flag any error for missing pins.

CDS_XNET_NAME

The CDS_XNET_NAME property, attached to a net, specifies the net that becomes the name of the Xnet.

CLASS

The CLASS property, attached to a component, defines a logical grouping for the component. This property is stored in body section of the device file (third-party) or the `.pstchip.dat` file (Cadence format).

CLIP_DRAW

The CLIP_DRAW property, generated automatically by Allegro PCB Editor and attached to components, symbols, or the design, uses the format CLIP_n to indicate the total number of times, plus one, that the paste operation was used in the layout or symbol drawing. `CLIP_DRAW` always stores one more than the current number of operations.

You can change the `CLIP_`*n* value to another number by editing the property interactively using the *Edit – Property* menu item.

CLIP_DRAWING

The CLIP_DRAWING property, generated automatically by Allegro PCB Editor and attached to connect lines, devices, pins, filled rectangles, lines, rectangles, shapes, symbols, vias, and voids, uses the format `CLIP_`*n*, to indicate the number of times that an element in a clipboard file was pasted into a layout or symbol drawing.

CLK_2OUT_MAX

The CLK_2OUT_MAX property, attached to a net or pin, specifies the maximum delay from the active clock range at a latch to the output change. The value is `INTEGER`.

CLK_2OUT_MIN

The CLK_2OUT_MIN property, attached to a net or pin, specifies the minimum delay from the active clock range at a latch to the output change. The value is `INTEGER`.

CLK_SKEW_MAX

The CLK_SKEW_MAX property, attached to a data net or a pin in the data net, is used by the *Timing Setup/Hold* tab of the Constraint Manager. It defines the maximum skew in the clock signal between the launching and latching components in nanoseconds.

CLK_SKEW_MIN

The CLK_SKEW_MIN property, attached to either the data net or a pin in the data net, is used by the *Timing Setup/Hold* tab of the Constraint Manager. This property defines the minimum skew in the clock signal between the launching and latching components in nanoseconds.

CLOCK_NET

The CLOCK_NET property, attached to a net, is related to the Allegro PCB SI timing spreadsheet and the *File – Import – Timing* command. The *Import Timing* command adds this property to a net to store the name of the net that is used to clock the net's data. The value is `STRING`.

For example, if net `DATA1` is clocked by net `CLOCK1`, then `DATA1` has the CLOCK_NET property attached with the a value of `CLOCK1`.

COMMENT

The COMMENT property enables you to add a user comment to any design object. It was implemented primarily to support of the Waived DRC feature so that a comment can be added to a waived design rule check.

COMMENT_BODY

The COMMENT_BODY property, attached to components that have no electrical meaning, lets Allegro Design Entry HDL ignores the components and omit them from the netlist.

COMMENTS

The COMMENTS property, attached to the board root allows you to place text comments.

COMP_NAME

The COMP_NAME property is attached to a schematic instance that has component definition properties. The value of the COMP_NAME is used as the name for the alternate physical part.

If you do not specify the COMP_NAME property, the values of the component definition properties are appended to the physical part name. This is the default naming mechanism.

Syntax

```
COMP_NAME = <value>
```

Example

```
COMP_NAME = RES100
```

COMP_NAME_SUFFIX

The COMP_NAME_SUFFIX property is attached to a schematic instance that has component definition properties. The value of COMP_NAME_SUFFIX is added to the end of the physical part name.

If you do not specify the COMP_NAME_SUFFIX property, the values of the component definition properties are appended to the physical part name. This is the default naming mechanism.

For example, if the part name is `74LS00` and you add a COMP_NAME_SUFFIX of "`VERSION 2`", the new name is "`74LS00-VERSION 2`."

Syntax

```
COMP_NAME_SUFFIX = <value>
```

Example

```
COMP_NAME_SUFFIX = 100
```

If you assign this property to a schematic instance of the physical part type `74LS00`, this schematic instance is named as `74LS00-100` during packaging.

COMPONENT_WEIGHT

The COMPONENT_WEIGHT property, attached to a reference designator (component), is used by automatic placement to determine the relative importance of components. The value is an integer from `0` to `100`. All components have a default weight of `50`.

If you assign a heavy weight (greater than 50) to a component, automatic placement attempts to place its heavily connected components nearby. This weighting is useful when placing components tied to a connector.

You may want to initially run automatic placement with a weight greater than 50 on the connector, using the whole design. If you add this property to more than one component on a net, the effective weight is multiplied on the net.

The following table shows how various weights affect a component:

Weight	Effect on a Component
0	No consideration is given to placing the component close to other components on its net.
Less than 50	Placing this component close to interconnected components is less important than for components with no weight property attached.
50	This is the default value attached to all components. A value of 50 is the same as having no property attached.
Greater than 50	The component is given preference for placement near interconnected components over components with no COMPONENT_WEIGHT property attached.

CONDUCTOR_MATERIAL

The CONDUCTOR_MATERIAL property, attached to a symbol used as an interposer, represents the material of the conductive layer. This material is one listed in the material file, `mcm_mat.data`.

CONDUCTOR_THICKNESS

The CONDUCTOR_THICKNESS property, attached to a symbol used as an interposer, is a number (db units) depicting the thickness of the conductive layer.

CONNECTOR_SIGNAL_MODEL

The CONNECTOR_SIGNAL_MODEL property attaches a Connector model to a RefDes (component) which connects to another design (such as a male/female connector, PCI slot, etc.). The value is a string that matches the name of the Connector model assigned to the component.

CPW_DISABLED

The CPW_DISABLED property disables individual nets from being extracted as coplaner waveguides (CPWs) during Ems2d model generation.

DEGAS_NO_VOID

The DEGAS_NO_VOID property, attached to conducting elements on layers adjacent to degassed shapes, indicates that APD should not create degassing openings on the shape if the opening overlaps or comes too close to an element with this property attached. This property is available in APD and Constraint Manager.

DENSE_COMPONENT

The DENSE COMPONENT property, attached to a reference designator, indicates that the component is heavily connected to other components. During scheduling, the pins are put on these components at the end of their net schedule. Be sure that you set the value to TRUE.

DEVICE_LABEL

DEVICE_LABEL is used by the GenRad interface. Allowable values are the GenRad Circuit Description Language (CDL) symbol abbreviations: NC, RV, QN, QP, BOARD, IC*nn*, ICA*nn*, DIC*nn*, UK*nn*, GD or the device name for ICs. (*nn* equals the number of pins on the device.)

GenRad is not supported in versions 14.0 and higher.

DFA_DEV_CLASS

The DFA_DEV_CLASS property, attached to a symbol, classifies components according to real-time Design for Assembly (DFA), package-to-package spacing values defined in the DFA Constraints Dialog spreadsheet, available by choosing *Setup – DFA Constraint Spreadsheet* (`dfa spreadsheet` command).

You use the DFA Classification Editor dialog box to add or remove symbol definitions from user-defined classes. (The DFA Classification Editor is available by clicking *Show symbol classifications...* on the DFA Constraints Dialog spreadsheet.) Allegro PCB Editor treats these classes as components comprised of symbols to which the DFA spacing values defined for the class default.

For example, fifty versions of an 0805 package symbol may exist, all complying to the same DFA rule set. A single class line entry in the spreadsheet assumes the rules for each instance of the 0805 class of package symbols.

Clicking *Apply* on the DFA Classification Editor dialog box assigns the DFA_DEV_CLASS property to the symbol definitions in the classes you specified.

For more information on meeting DFA requirements, refer to the *Completing the Design* user guide in your documentation set.

DIELECTRIC_MATERIAL

The DIELECTRIC_MATERIAL property, attached to a symbol used as an interposer, represents the dielectric material (substrate). This material is one listed in the material file, `mcm_mat.data`.

DIELECTRIC_THICKNESS

The DIELECTRIC_THICKNESS property, attached to a symbol used as an interposer, is an number depicting the thickness of the substrate.

DIFFERENTIAL_PAIR

The DIFFERENTIAL_PAIR property, attached to a net, names the differential pair. To create a differential pair, assign two nets with a DIFFERENTIAL_PAIR property that have the same name.

The DIFFERENTIAL_PAIR property became obsolete in Allegro PCB Editor as of Release 15. It has been superseded by the DIFF_PAIR object in the Allegro database. The SPB Schematic tools still support the DIFFERENTIAL_PAIR property but when their netlists are loaded into an Allegro database, the property members are mapped to the new DIFF_PAIR database object.

DIFFP_USES_PROPERTIES

The design-level DIFFP_USES_PROPERTIES property preserves the constraint resolution (precedence) of differential pair overrides. When enabled, differential pair overrides have a higher precedence than constraint regions. This property is automatically applied during uprev if any differential pair has any one of the following properties attached:

- DIFFP_PRIMARY_GAP
- DIFFP_NECK_GAP
- MIN_LINE_WIDTH
- MIN_NECK_WIDTH.

You can manually select or deselect this option:

- In Constraint Manager, on the *Electrical Options* tab of the *Constraints Modes* and *Options* dialog box (choose *Analysis–Modes*).
- In a physical editor, (choose *Edit–Property*)

Note: You may not get the desired result if this option is enabled and differential pair Line and Gap constraints are applied by constraint region. The purpose of this property is to preserve the DRC status of an upreved design. Enabling this option is not recommend for new designs. You should review your constraints and eliminate the need for this option.

DIFFP_COUPLED_MINUS

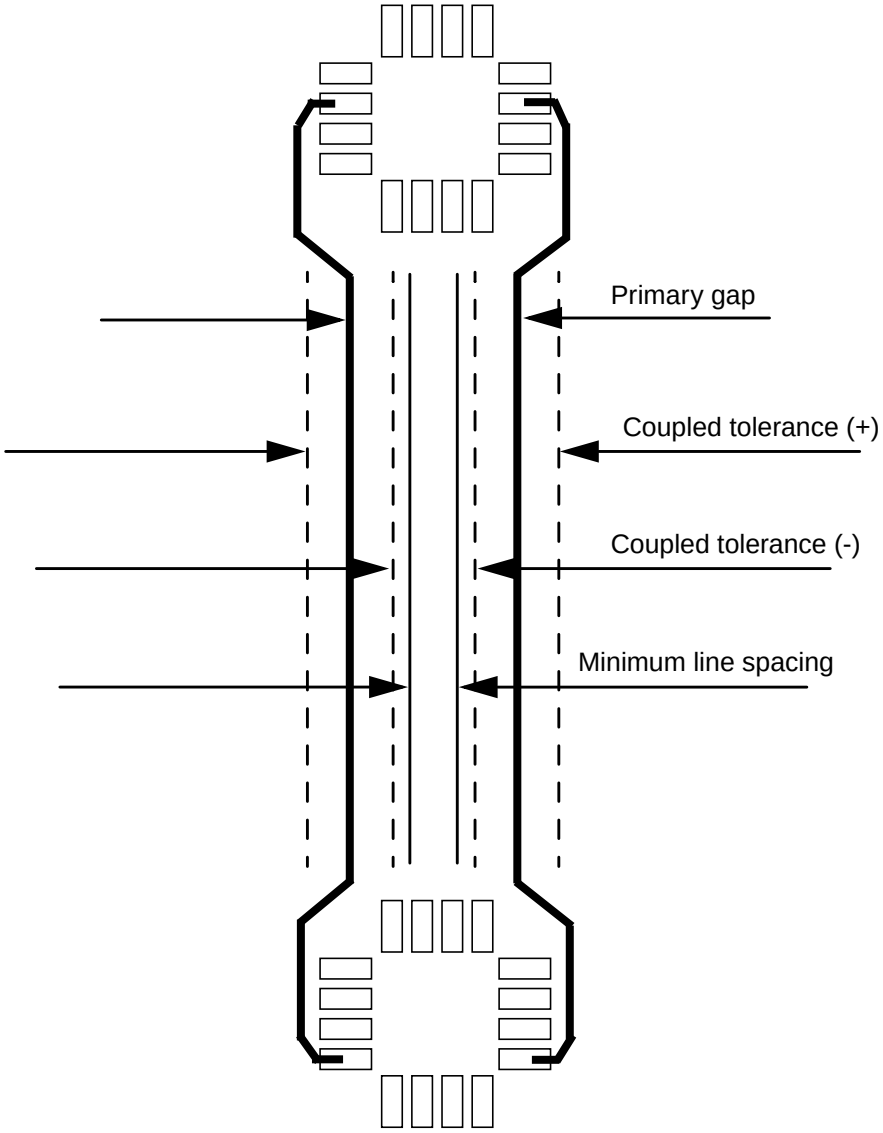
The DIFFP_COUPLED_MINUS property, attached to a net, overrides the *Coupled Tolerance (-)* electrical constraint for differential pairs. Together with the DIFFP_COUPLED_PLUS property, these two properties define two bands around the primary gap in which the lines of a pair can legally go beyond or closer than the value of the DIFFP_PRIMARY_GAP property.

When the lines of etch are within these bands, they are coupled. When they move from being uncoupled to coupled, this is a coupling event. When the lines move from being coupled to uncoupled, it is an uncoupling event. These events are used by phase tolerance (DIFFP_PHASE_TOL) and uncoupling (DIFFP_UNCOUPLING_LENGTH) checks.

The DIFFP_COUPLED_MINUS value should allow etch to remain above the DIFFP_MIN_SPACE value.

In Figure 1-1, the dotted lines represent the coupling bands.

Figure 1-1 Coupling Bands for Differential Pairs



A related property, DIFFP_UNCOUPLED_LENGTH, controls how much either line can uncouple and still be acceptable. For additional information about the uncoupling check, see DIFFP_UNCOUPLED_LENGTH.

By default, the *Coupled Tolerance (-)* electrical constraint is empty. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

Edits to the property at the net level bubble up to the differential pair group and the Xnet level. The value of this property is in design units. It is used by automatic routing and DRC checking.

DIFFP_COUPLED_PLUS

The DIFFP_COUPLED_PLUS property, attached to a net, overrides the *Coupled Tolerance (+)* electrical constraint for differential pairs. Together with the DIFFP_COUPLED_MINUS property, these two properties define two bands around the primary gap in which the lines of a pair can legally go beyond or closer than the value of the DIFFP_PRIMARY_GAP property. When the lines of etch are within these bands, they are coupled. See Figure 1-1.

A related property, DIFFP_UNCOUPLED_LENGTH, controls how much either line can uncouple and still be acceptable.

By default, the *Coupled Tolerance (+)* electrical constraint is empty. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

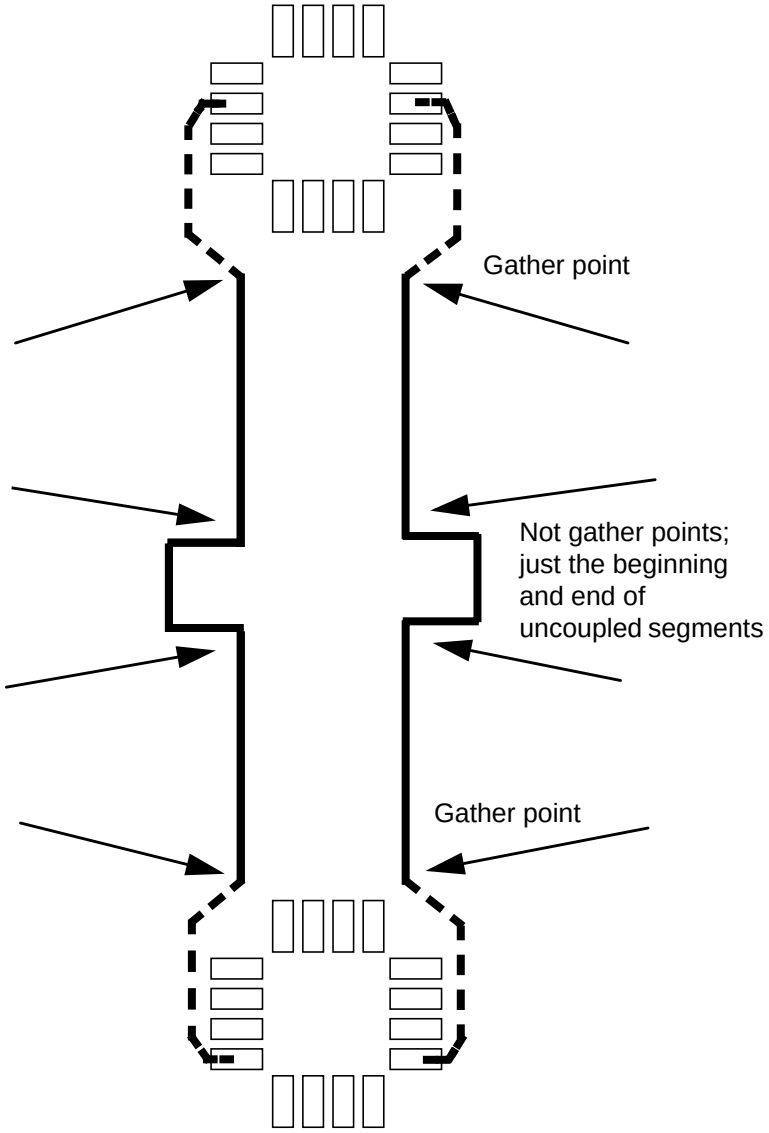
Edits to the property at the net level bubble up to the differential pair group and the Xnet level. The value of this property is in design units. It is used by automatic routing and DRC checking.

DIFFP_GATHER_CONTROL

The DIFP_GATHER_CONTROL property, attached to a net, overrides the *Gather control* electrical constraint for differential pairs. Set this property to *Include* or *Ignore* to indicate whether the DIFFP_UNCOUPLED_LENGTH DRC check includes or excludes the line segments that diverge as the pair of nets go from driver pins and to receiver pins.

The points at which the line segments converge after leaving driver pins or diverge as they move toward receiver pins are gather points. Any other divergences are simply uncoupled segments. Figure 1-2 shows the gather points for a differential pair. The dotted lines in the illustration mark the line segments that are included or ignored by this property.

Figure 1-2 Gather Points for Differential Pairs



By default, the *Gather control* constraint is empty. However, if an uncoupling check is performed and you have not set the *Gather control* constraint, it defaults to *Include*, meaning that the entire length of each line segment is checked. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

Edits to the property at the net level bubble up to the differential pair group and the Xnet level. This property is used by automatic routing and DRC checking.

DIFFP_MIN_SPACE

The DIFFP_MIN_SPACE property, attached to a net, overrides the *Minimum line spacing* electrical constraint for differential pairs. It defines a line-to-line manufacturing constraint that should always be maintained between the lines in the pair. The value of this property is in design units. Follow this guideline in defining this property and others:

DIFFP_PRIMARY_GAP – DIFFP_COUPLED_MINUS ≥ DIFFP_MIN_SPACE

See Figure 1-1.

If you do not enter a value for this property or the *Minimum line spacing* electrical constraint, the Allegro PCB Editor use the *Line to Line* rule from a spacing constraint set assigned to the nets in the differential pair.

By default, the *Minimum line spacing* electrical constraint is empty. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

Edits to the property at the net level bubble up to the differential pair group and the Xnet level. This property is used by automatic routing and DRC checking.

DIFFP_NECK_GAP

The DIFFP_NECK_GAP property, attached to a net, overrides the *Neck Gap* electrical constraint for differential pairs. It defines the edge-to-edge spacing between a pair as it goes through tight areas full of component pins and vias. The value of this property is in design units.

This property overrides any value in the DIFFP_PRIMARY_GAP property when the differential pair's spacing collapses to or below the value of the *Min neck width* rule in a physical constraint set assigned to the nets in the pair.

Be sure that the neck gap does not go below any DIFFP_MIN_SPACE value that you set.

Note: You do not need to define a neck gap if you set the DIFFP_COUPLED_MINUS property with a value that accounts for the needed neck gap.

By default, the *Neck Gap* electrical constraint is empty. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

Edits to the property at the net level bubble up to the differential pair group and to the Xnet level. This property is used by automatic routing and DRC checking.

DIFFP_PHASE_CONTROL

Currently, this property is not supported.

DIFFP_PHASE_TOL (formerly DIFFP_LENGTH_TOL)

The DIFFP_PHASE_TOL property, attached to a net, replaces the DIFFP_LENGTH_TOL property. Use this property to override the *Phase tolerance* electrical constraint for differential pairs. It defines the allowable difference between the length of the lines of etch in a differential pair. The value is a string—indicating either length or delay. (The previous value of a percent delay is no longer supported.) The units you enter define the type of value it is. If you do not enter any units, the tool assumes design units for a length measurement.

When you check phase tolerance dynamically, you must also set the DIFFP_COUPLED_PLUS or the DIFFP_COUPLED_MINUS properties, or both. Each time the lines become coupled (a coupling event), the Allegro PCB Editor measure both line lengths back to the driver pins and makes sure that the segment's length falls within the phase tolerance restriction. See Figure 1-1 for more information about coupling.

When you check phase tolerance statically, the total length of each line must meet the phase tolerance restriction.

By default, the *Phase tolerance* electrical constraint is empty. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

Edits to the property at the net level bubble up to the differential pair group and the Xnet level. This property is used by automatic routing and DRC checking.

DIFFP_PRIMARY_GAP

The DIFFP_PRIMARY_GAP property, attached to a net, overrides the *Primary Gap* electrical constraint for differential pairs. It indicates the ideal edge-to-edge spacing between the pair that should be maintained for the entire length of the pair. The value of this property is in design units.

Values you set for the DIFFP_NECK_GAP property override this property in areas that need smaller gaps to get through dense components.

By default, the *Primary Gap* electrical constraint is empty. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

Edits to the property at the net level bubble up to the differential pair group and the Xnet level. This property is used by automatic routing and DRC checking.

DIFFP_UNCOUPLED_LENGTH (formerly DIFFP_2ND_LENGTH)

The DIFFP_UNCOUPLED_LENGTH property, attached to a net, replaces the DIFFP_2ND_LENGTH property. Use this property to override the *Max uncoupled length* electrical constraint for differential pairs. This property sets the cumulative maximum distance that segments of one net in a differential pair can run inside or outside the coupling band. The value of this property is in design units. If you do not specify a value, Allegro PCB Editor does not perform a coupling DRC check.

The pair is coupled when the lines are neither closer nor farther than two bands around the primary gap, defined by the DIFFP_COUPLED_PLUS and DIFFP_COUPLED_MINUS values. See Figure 1-1.

Each time a line becomes uncoupled (an uncoupling event), the uncoupled length is captured by Allegro PCB Editor in this check. If you set the DIFFP_GATHER_CONTROL property to *Include* and the lines are uncoupled right from the driver pin, the beginning of each line is the first uncoupling event.

If you set the DIFFP_GATHER_CONTROL property to *Ignore*, the line segments that diverge as the pair of nets go from driver pins to the first coupling event and diverge again as they move from the last portion of coupling to receiver pins, are not included in the uncoupled length. In Figure 1-2, the dotted lines are not included in the uncoupled length check.

Notes:

- If a differential pair has multiple driver and receiver pins (PINUSE is set to OUT [driver], IN [receiver], or BI [bidirectional]), each driver and receiver combination for each line is treated as an independent path for checking uncoupling. Consequently, a check is performed for each portion of etch between a driver and receiver and for the entire line.
- The Allegro PCB Editor measures only nets and Xnets. It does not perform uncoupling checks on system extended nets (SXnets).

By default, the *Max uncoupled length* electrical constraint is empty. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI.

Edits to the property at the net level bubble up to the differential pair group and the Xnet level. This property is used by automatic routing and DRC checking.

DIFF_PAIR_PINS_NEG

The `DIFF_PAIR_PINS_NEG` property is associated with a pin at the package level to identify the negative pin of a differential pair. To create a differential pair, assign the `DIFF_PAIR_PINS_POS`

property to one pin and the DIFF_PAIR_PINS_NEG property to another with the differential pair name as the value for each property.

The DIFF_PAIR_PINS_NEG property is stored in chips.prt in the following syntax:

```
DIFF_PAIR_PINS_NEG='diff_pair_name';
```

Note: Two pins with the same *diff_pair_name* value constitute a differential pair.

DIFF_PAIR_PINS_POS

The DIFF_PAIR_PINS_POS property is associated with a pin at the package level to identify the positive pin of a differential pair. To create a differential pair, assign the DIFF_PAIR_PINS_POS property to one pin and the DIFF_PAIR_PINS_NEG property to another with the differential pair name as the value for each property.

The DIFF_PAIR_PINS_POS property is stored in chips.prt in the following syntax:

```
DIFF_PAIR_PINS_POS='diff_pair_name';
```

Note: Two pins with the same *diff_pair_name* value constitute a differential pair.

DRC_UNROUTED_MINPROP

The DRC_UNROUTED_MINPROP boolean property, attached to a design, includes the manhattan distance of unrouted connections when computing propagation delay. Make sure that the value is set to TRUE.

DRC_UNROUTED_RELPROP

The DRC_UNROUTED_RELPROP boolean property, attached to a design, includes the manhattan distance of unrouted connections when computing relative propagation delay (RPD).

DRIVER_TERM_VAL

The DRIVER_TERM_VAL property, attached to a net, is the value of a terminator component that is added to the driver end of the net. This property is used by the automatic terminator assignment program (*Logic – Terminator Assignment*).

DYN_CLEARANCE_OVERSIZE

The DYN_CLEARANCE_OVERSIZE property, attached to shapes, filled rectangles (frectangles), pins, vias, or clines, overrides the value in the *Oversize values* field in the *Clearances* tab in the Shapes Instance Parameters dialog box and increases the clearance around the specified element. The value is a positive integer. Negative integers are not supported.

For example, if the value in the *Oversize values* field is `6`, and the value of the DYN_CLEARANCE_OVERSIZE property is `4`, the total clearance is `10`.

DYN_CLEARANCE_TYPE

The DYN_CLEARANCE_TYPE property, attached to a pin or via, overrides the clearance type in the *Clearances* tab in the Shapes Instance Parameters dialog box.

A value of *Thermal/Anti* makes a void the size of thermal relief or antipad (defined in the padstack of a pin or a via). If this void is smaller than the void that would have been generated by the DRC clearance, the void is enlarged, but a DRC can still result. A DRC occurs in situations where the difference between the void that would be generated by DRC clearance is much larger than the void generated by the thermal relief/antipad. A value of *DRC* uses the DRC distance as the clearance around the pad. Values are `DRC_VALUE, ANTI_THERMAL,` and `NO_VOID`.

DYN_DELETED_ISLAND

The DYN_DELETED_ISLAND property, attached to voids generated through the `island_delete` command lets you later delete the island voids as a group, but retains user-added voids in dynamic shapes.

DYN_DO_NOT_VOID

The DYN_DO_NOT_VOID property, attached to shapes, frectangles, lines, or clines, prevents voiding.

DYN_FIXED_THERM_WIDTH

The DYN_FIXED_THERM_WIDTH property, attached to pins or vias, allows you to set a fixed value for the width of thermal lines regardless of the DRC rule or setting for thermal width

found in the shape parameters. This property also suppresses Min Line Width DRC errors for thermal clines in dynamic shapes.

DYN_MIN_THERMAL_CONNS

The DYN_MIN_THERMAL_CONNS property, attached to pins or vias, overrides the value in the *Minimum connects* field of the *Thermal Relief Connects tab* in the Shapes Instance Parameters dialog box.

DYN_MAX_THERMAL_CONNS

The DYN_MAX_THERMAL_CONNS property, attached to pins or vias, overrides the value in the *Maximum connects* field of the *Thermal Relief Connects tab* in the Shapes Instance Parameters dialog box.

DYN_OVERSIZE_THERM_WIDTH

The DYN_OVERSIZE_THERM_WIDTH property, attached to dynamic shapes, pins, or vias, specifies the width of the connect lines added as thermal relief. The width of the reliefs should be less than or equal to the width of the hatch line to which they connect.

DYN_THERMAL_BEST_FIT

The DYN_THERMAL_BEST_FIT property, attached to pins or vias, overrides the thermal connect style if the chosen style does not provide sufficient thermal connects within the specified minimum and maximum number of thermals in 15-degree increments. When you attach this property, it disables the *thermal connect style* field. Be sure that you set the value to TRUE.

DYN_THERMAL_CON_TYPE

The DYN_THERMAL_CON_TYPE property, attached to pins or vias, specifies how pins and vias with the same net name as the shape should be connected to the shape.

- *Orthogonal*: Connects straight up-down or left-right. The pin connects directly to the void outline or hatch lines.
- *Diagonal*: Connects diagonally upper left to lower right and lower left to upper right.

- *8-Way*: Connects lines from the thermal relief to the pin/via both diagonally and orthogonally.
- *Full Contact*: Creates no voids. For solid shapes, the shape completely fills around the pin. For crosshatched shapes, the hatch lines provide the connections or Allegro PCB Editor adds short connect lines.

ECL

The ECL property, attached to a net, identifies it a high-speed net. The value is either `TRUE` or `FALSE`. This property is used by automatic routing and ratsnest scheduling. If the ECL property is attached to a net, the editor assumes a stub length of zero and a ratsnest schedule of `SOURCE_LOAD_DAISY_CHAIN`.

You can add this property to an Allegro Design Entry HDL schematic.

You can override this property by assigning the STUB_LENGTH property or the RATSNEST_SCHEDULE property.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

ECL_TEMP

The ECL_TEMP property is attached to nets that are processed by the terminator assignment program in incremental mode (*Logic Terminator Assignment*).

EDGE_SENS

The EDGE_SENSE property, attached to a net, Xnet, or ECset, defines whether a receiver pin is sensitive to non-monotonicity in the waveform. The value of this constraint shows which edges of the waveform are sensitive, that is, rising edge only, falling edge only, both edges, or neither edge.

ELECTRICAL_CONSTRAINT_SET

The ELECTRICAL_CONSTRAINT_SET property, attached to a net, is the name of the electrical constraint set that you are applying to the net. Electrical override properties are supported on the NETCLASS object.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

EMC_COMP_TYPE

The EMC_COMP_TYPE property, attached to a component or device, is a string that specifies a variable identifying the component type. This property is used by EMControl.

EMC_CRITICAL_IC

The EMC_CRITICAL_IC property, attached to a component or device, is a string that identifies the class of a critical IC. This property is used by EMControl.

EMC_CRITICAL_NET

The EMC_CRITICAL_NET property, attached to a net, is a string that identifies the class of a critical net. This property is used by EMControl.

EMC_CRITICAL_REGION

The EMC_CRITICAL_REGION property, attached to critical-level components, nets, and regions (from Class 1 to Class 5), allows EMC to perform rule checking. A region is a rectangle, parallel with X or Y, on the BOARD GEOMETRY/TOP_ROOM or BOARD GEOMETRY/BOTTOM_ROOM layer.

EMC_RUN_DIR

The EMC_RUN_DIR property is automatically attached to a board when you save changes to the EMC Initialization dialog box. The EMC run directory name is stored in this property and is used by EMControl.

ETCH_TURN_UNDER_PAD

The ETCH_TURN_UNDER_PAD property is attached to components/component pins, or symbols/symbol pins. Property values are on | off. When on, it specifies that a wire can route and tune under a pad. When off, a wire that turns under a pad causes a violation. In cases where the property is attached to a symbol, the property is extended to all instances of the symbol.

EXTERNAL_DRC_VALUE

The EXTERNAL_DRC_VALUE property, attached to external DRC markers at the SKILL level, lets you assign a value to an external DRC.

FAMILY

The FAMILY property, attached to a component, specifies the logic family for the component.

Example

```
Family='74LSTTL';
```

FILLET

The FILLET property, attached to connect lines, indicates the connect lines that are for filleting around a pad or via. This property prevents DRC from identifying dangling connect lines.This property is assigned by the pad fillet glossing function. You can also add this property. Be sure that you set the value to `TRUE`.

FIRST_INCIDENT

The FIRST_INCIDENT constraint property, attached to a net, Xnet, or ECset, defines whether a signal is required to switch on the first incident wave. The value of this constraint shows which edges of the waveform must switch on their first incident wave. The values are *Rising* edge only, *Falling* edge only, *Both* edges or *Neither* edge.

FIX_ALL

The FIX_ALL property, attached to a reference designator (component), indicates that you cannot swap this component, its functions, or its pins.

FIXED

The FIXED property, attached to components, symbols, nets, pins, vias, clines, lines, filled rectangles (frectangles), rectangles, shapes, and groups, indicates that the editor prevents the following:

- Movement or deletion of the object

- The PCB Router from ripping up connections in the net
- Glossing on the net

Be sure that you set the value to TRUE.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the net properties command and the *Allegro Constraint Manager User Guide* for additional information.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

FIXED_T_TOLERANCE

The FIXED_T_TOLERANCE property, attached to a Tpoint, specifies a radius around a Tpoint that the router can route to the Tpoint. You must attach the FIXED_T_TOLERANCE property to all T points on all nets of an Xnet.

FP_BOARD_CLEARANCE

The FP_BOARD_CLEARANCE property is an Allegro PCB SI internal property and is not accessible to the user.

FP_NOTES_NO_EDIT

The FP_NOTEST_NO_EDIT property, added to a note after it has been added to the drawing, specifies that you cannot edit the note until the property has been removed. You can add notes only in Allegro PCB SI using the *Setup – Notes* command.

FP_NOTES_TEXT_BLOCK

The FP_NOTES_TEXT_BLOCK property is an Allegro PCB SI internal property and is not accessible to the user.

FP_REFDES_TEXT_BLOCK

The FP_REFDES_TEXT_BLOCK property is an Allegro PCB SI internal property and is not accessible to the user.

FP_ROOM_NAME_TEXT_BLOCK

The FP_ROOM_NAME_TEXT_BLOCK property is an Allegro PCB SI internal property and is not accessible to the user.

GROUP

The GROUP property, attached to a section on a component, identifies schematic instances that will be packaged together. This property lets you control packaging assignments without having to keep track of specific reference designators and sections.

See the *Placing the Elements* user guide in your documentation set.

HARD_LOCATION

The HARD_LOCATION property, attached to a reference designator (component) or a function designator (gate), prevents the reference designator of a component from being automatically or interactively renamed. Be sure that you set the value to TRUE.

HAS_FIXED_SIZE

The HAS_FIXED_SIZE property, attached to the nonvectored version of a component when one version of the component can be sized, passes size information to the simulation primitives to define the final size of the component. The HAS_FIXED_SIZE property is required; do not change it.

The HAS_FIXED_SIZE property identifies those symbols that have a fixed size. These symbols should not be given a SIZE property. The HAS_FIXED_SIZE property has the following functions:

- Specifies that the symbol it attaches to has a fixed known size (specified in the property value)
- Causes an error message if a SIZE property is found

Many of the symbols in the standard libraries have this property attached to them. It is used for versions of physical parts that display all sections. The vectored version of the symbol typically represents a one-bit section of the part. The second symbol version represents all sections of the part.

For example, if the part has four sections, then the second symbol version is given a HAS_FIXED_SIZE=4 property to specify that it represents four bits. This is important

because the models (for the Timing Verifier and Logic Simulator) are modeled as one-bit sections with the SIZE property specifying the actual number of bits for each instance. The HAS_FIXED_SIZE property causes a default 8 property to be attached to support the models. The presence of a user-assigned SIZE property on these symbols is always an error since none of the pins of the symbol or the definition of the symbol use the SIZE property.

You can attach the HAS_FIXED_SIZE property as a default symbol property (attached to the ORIGIN) or you can attach it to the symbol when used in a schematic.

HDL_CONCAT

The HDL_CONCAT property, attached to a symbol, specifies to the netlister that the component should be treated as a concatenation or merge symbol. These symbols allow you to merge smaller-width signals into a bussed signal or take a bussed signal and break it into smaller-width signals.

Add the HDL_CONCAT property to the symbol for a component that you want to use for:

- Merging a number of signals, ports, or signal aliases into a group and then routing the group to a port or instance with a single wire.
- Splitting a vectored signal or port into a number of signals of smaller width.

Ensure that the name of the output pin of the symbol is the highest alphanumeric value of all the pins on the symbol. For example, if the input pins are named AA, DD, and FF, the output pin cannot be named BB.

The pattern of input bits maps to the output bits. In these examples of input bits, A:C:D:B, C maps to the second bit of the output, and in the pattern of A:B:C:D, C maps to the third bit of the output.

See the section “Merge Symbols” in the chapter “Creating a Schematic” in the *Allegro Design Entry HDL User Guide*.

HDL_LSBTAP

The HDL_LSBTAP property, attached to a symbol, specifies to the netlister that the component should be treated as a tap symbol, which breaks out the lower eight bits of the bussed signal.

HDL_MSBTAP

The HDL_MSBTAP property, attached to a symbol, specifies to the netlister that the component should be treated as a tap symbol, which breaks out the upper eight bits of the bussed signal.

HDL_NOT

The HDL_NOT property, attached to a symbol, converts the symbol to a `NOT` symbol.

HDL_PORT

The HDL_PORT property, attached to a symbol, specifies to the netlister that the component should be treated as a port symbol. This indicates that the signal is an interface to the upper level of hierarchy.

HDL_POWER

The HDL_POWER property, attached to a symbol, specifies that it is a power or ground symbol. Allegro Design Entry HDL reads this property to identify power and ground symbols on the schematic.

If an unnamed signal is connected to an instance that has the HDL_POWER property, Allegro Design Entry HDL treats the signal as a global signal and assigns it the same name as the value of the HDL_POWER property. For example, if an unnamed signal is connected to an instance that has the `HDL_POWER=GND` property, Allegro Design Entry HDL assigns the `GND\G` name to the signal. To assign some other name to the signal, you must name the signal.

Design Entry HDL also assigns the `SIG_NAME=<`*`HDL_POWER property value`*`>\G` property on the pin to which the unnamed signal is connected. From Concept HDL 14.2, you cannot modify the `SIG_NAME=<`*`HDL_POWER property value`*`>\G` property automatically assigned on pins.

HDL_REPLICATE

The HDL_REPLICATE property, attached only to an instance that has two pins, specifies that the instance replicates signals. The input pin name must be `INPUT` and the output pin name must be `OUTPUT<SIZE-1..0>`.

You must also attach the SIZE property to the instance on which you have added the HDL_REPLICATE property. The SIZE property on the instance determines the number of times that the signal connected to the pin INPUT has to be replicated.

The instance, which has the HDL_REPLICATE property, works the same as the REPLICATE symbol in the `standard` library.

HDL_SLASH

The HDL_SLASH property, attached to a symbol, converts the symbol to a SLASH symbol.

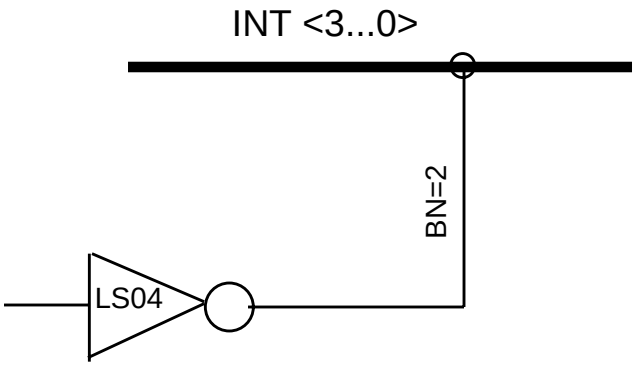
HDL_SYNONYM

The HDL_SYNONYM property, attached to a a two-pin device, specifies another name for a signal. The instance that has the HDL_SYNONYM property works the same as the SYNONYM symbol in the `standard` library.

HDL_TAP

The HDL_TAP property, attached to a symbol, specifies to the netlister that the component should be treated as a tap symbol. This means that it breaks out a set of bits indicated by the value of the BN property.

If a tapped signal is unnamed, Allegro Design Entry HDL assigns it the same base name as the name of the signal that is being tapped, along with the bits specified as the value of the BN property on the tap symbol.
1



In the example above, Allegro Design Entry HDL automatically assigns the name `INT<2>` to the tapped signal, where `INT` is the base name of the signal that is being tapped and 2 is the

1.

value of the BN property on the tap symbol. You move the mouse pointer over the signal to view the signal name. To assign some other name to the signal, you must name the signal.

Allegro Design Entry HDL also automatically assigns the SIG_NAME=*base name of signal being tapped*<*value of BN property on tap symbol*> property on the output pin of the tap symbol. In the above example, Allegro Design Entry HDL assigns the SIG_NAME=INT<2> property to the output pin of the tap symbol. From Concept HDL 14.2, you cannot modify the SIG_NAME=*base name of signal being tapped*<*value of BN property on tap symbol*> property automatically assigned on the output pin of tap symbols.

HEIGHT

The HEIGHT property, attached to component definitions in a schematic system, specifies the component height. If you specify the unit type in the property value, Allegro PCB Editor converts the value to database units prior to exporting. It is used by the Design Rule Checking (DRC) program to determine if there is a height spacing violation among components that are mounted under each other.

IC_DESIGN_CELL_INSTANCE_NAME

The IC_DESIGN_CELL_INSTANCE_NAME property, attached to symbol pins, provides the cell instance name of the I/O driver cell for this pin. This property lets you manipulate objects in SiP Layout based on the connected IC cell instance name.

IC_DESIGN_CELL_MASTER_NAME

The IC_DESIGN_CELL_MASTER_NAME property, attached to symbol pins, provides the master cell name of the I/O driver cell for this pin. This property lets you manipulate objects in SiP Layout based on the connected IC cell master.

IC_DESIGN_CELL_PIN_NAME

The IC_DESIGN_CELL_PIN_NAME property, attached to symbol pins, provides the name of the pin on the I/O driver cell for this die pin. This property lets you manipulate objects in SiP Layout based on the connected IC cell master pin.

IC_DESIGN_NET_NAME

The IC_DESIGN_NET_NAME property, attached to symbol pins, provides the name of the IC net connected to that pin in IO Planner. This property lets you manipulate objects in SiP Layout based on the connected IC net name.

IDF_OTHER_OUTLINE

The IDF_OTHER_OUTLINE property, attached to shapes, rectangles and filled rectangles, insures that translation occurs for these elements when exporting design drawing data into an Intermediate Data Format (IDF) file. IDF includes them in the pre-defined .OTHER_OUTLINE section. The syntax in the IDF file is: `class/subclass:n` where n exists only as an unique identifier. You must set the value to `TRUE`.

IDF_OWNER

The IDF_OWNER property, attached to entities, specifies that the entity is owned by an electrical tool, such as the Allegro PCB Editor or a mechanical tool, or by neither. You can edit entities to change the ownership. Permissible values are: ECAD, MCAD, and UNOWNED. For information on ownership rules, see the *Transferring Logic Design Data* user guide in your documentation set.

Note: If IDF_OWNER is set to `MCAD` when imported into the Allegro PCB Editor, the FIXED property is also set to `TRUE`.

IMPEDANCE_RULE

The IMPEDANCE_RULE property, attached to a net, specifies an impedance restriction between any two pins on a net or between any pin and Tpoint connection on a net. This property is used by DRC checking and routing.

If a connection has an impedance constraint specified by both an IMPEDANCE_RULE property and an electrical constraint set, the IMPEDANCE_RULE value is used. Use Constraint Manager for pin-pairs.

The format of the IMPEDANCE_RULE property in the $A_PROPERTIES section of a netlist is

`IMPEDANCE_RULE{`*`ALL`*`:`*`ALL`*`:`*`impedance`*`:`*`tolerance`*`:}... ; `*`netname`*

`ALL:ALL`	Specifies the checking of impedance of every connect line (cline) segment in the net.

impedance	Required. Specifies the impedance constraint value in ohms expressed as a decimal number.
tolerance	Specifies the tolerance for an impedance value specified either as a value in ohms or a percentage. You must follow a percentage with a % (percent sign). If you do not specify a value, 2% is assumed.
netname	Specifies the name of the net to which the impedance applies.

INCLUDE_IN_RF_TOPOLOGY

The INCLUDE_IN_RF_TOPOLOGY property is used in the RF PCB Import flow to determine whether a standard component connected through a non-DC net to a component in an RF topology is to be considered part of that RF topology. A standard component is any component other than a discrete component or an RF component.

To include a standard component in an RF topology, add the INCLUDE_IN_RF_TOPOLOGY property to the standard component in Design Entry HDL or PCB Editor with a value of 1. The property can flow in both directions in the board design flow.

INLINE_PIN_VOIDS

The INLINE_PIN_VOIDS property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the distance between each pad when a void is generated around an entire group of pads, mainly DIP patterns. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Void Controls* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

INPUT_LOAD

The INPUT_LOAD property, specified in the chips.prt file and attached to a pin, indicates the load a pin presents when it is used as an input, or when not driving the signal. An input pin should always have the INPUT_LOAD property. An output pin should carry an INPUT_LOAD property whenever that pin can also place an input load on the signal. For example, a tri-state or an open collector output also presents a load when not driving the signal. This load must be considered when calculating the load of the entire net.

INSERTION_CODE

INSERTION_CODE is used by the CIMLink interface and the SDRC interface to indicate a specific Universal component insertion machine. Allowable values are VCD, RADIAL, DIP, MULTIMOD, SIP, ZIP, AXIAL, and SMD.

CIMLink is not supported in versions 14.0 and higher.

This property is not currently used.

ISRFELEMENT

The ISRFELMENT property, defined in Allegro PCB Design XL and attached to a component, identifies that it is an RF component. Its value is an integer ranging from 0 to 1. A 0 means that it is a discrete RF component; 1 indicates that it is a normal RF component.

JEDEC_TYPE

The JEDEC_TYPE property, attached to a component, specifies the footprint to be used in the Allegro PCB Editor design for the component in the logical netlist. This property is typically placed in the body section of the `chips` file. Its value can be overridden by an entry in a physical part table. This is typically the case when describing components, such as resistors and capacitors, that have a large number of physical parts all sharing the same logical part.

J_TEMPERATURE

The J_TEMPERATURE property, attached to a reference designator (component), defines the junction temperature for the component in degrees centigrade.

LAST_MODIFIED

The LAST_MODIFIED property, attached to the `DRAWING` symbol signifies the last modification date and time when the drawing was saved to the hard disk.

LAST_PIN_SWAP

All pin swaps will be tracked with the property LAST_PIN_SWAP. The property will attach to the pin and the value will be the name of the pin with which it was last swapped. Updating

the Allegro design from the Front End or third party netin will cause all pin swap properties to be deleted.

LAYERSET_GROUP

The LAYERSET_GROUP property, attached to a net, Xnet, differential pair, or bus defines wiring rules to control impedance, shielding, or return path requirements. If you attach this property at the net level, it overrides an ECset. Edits to the property at the net level propagate to the Xnet level.

LEAD_DIAMETER

The LEAD_DIAMETER property, attached to a board or symbol, specifies the lead diameter of a pin's lead. This is used in calculating the span value used in lead span audit of DFA.

LEFDEF_SPECIAL_NET

The LEFDEF_SPECIAL_NET property, attached to nets read from a DEF file, is used to determine whether a net is a special net. The property attached to special nets has a value of TRUE; the property attached to regular nets has a value of FALSE. Nets already present in a design, but not read from a DEF file, do not have this property attached.

You can toggle the values to determine whether or not a net is special. DEF OUT PARTIALLY depends on this property to determine whether or not a net is special. The other factors that DEF OUT uses to determine a special net (irrespective whether the LEFDEF_SPECIAL_NET value is attached or set to TRUE) are:

- Net names VSS and VDD are always special nets.
- The attachment of the ATR_VOLTAGE property to a net also makes it as special net.

LIBRARYn

The LIBRARYn property, attached to a symbol or in a map file, provides a library name to be used in a VHDL library clause in the entity and architecture text generated for the schematic. Use the following syntax:

```
LIBRARYn = library_name
```

where n is a unique number and `library_name` is the name of the library

Note: When you use this property on a VHDL_DECS symbol, it does not override the value of the library clauses specified for the VHDL netlisting options in the *Output* tab of the Design Entry Options dialog box. Instead, the libraries you specify on the VHDL_DECS symbol are appended to the list of libraries you specified for the VHDL netlisting options in the *Output* tab of the Design Entry Options dialog box.

LINE_OVERSIZE

The LINE_OVERSIZE property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the oversize value for how far away the copper should be kept for lines and clines. This value is added to the DRC clearance for clines and lines being voided. By default, there is no oversizing of voids. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Clearances* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

LOAD_TERM_VAL

The LOAD_TERM_VAL property, attached to a net, specifies the value of a terminator component to be added to the load end of the net. This property is used by the terminator assignment program.

LOCATION

The LOCATION property, attached to a component, lets you control packaging by assigning a particular physical part to a logical component on a design. The value is any alphanumeric string. Use the *Text – Property* menu command to attach LOCATION properties to components on a drawing.

When you attach the LOCATION property and use the menu command *Component – Section* to make physical part assignments, Packager-XL and the physical design system do not override the assignments. If you include a LOCATION property after packaging a design, you must repackage the design for Packager-XL to use the specified information.

LOCKED

The LOCKED property, attached to symbols and modules, prevents modification of individual symbol or module parts (such as pins, padstacks, etc.) when placed; however, you can delete, move, or rotate the symbol or module as a single object. You can add the property to a symbol instance in a design, or use the Symbol Editor. You may edit text on placed symbols or modules to which the LOCKED property is assigned.

During design partitioning, a master designer attaches the LOCKED property to symbols and modules to prevent partition designers from making modifications because the LOCKED property cannot be removed from objects in exported partitions. For more information on design partitioning, see the *Placing the Elements* user guide in your documentation set.

LOGICAL_PATH

The LOGICAL_PATH property, automatically generated and attached to a function designator (gate) by the `netlist import` process (`netrev`), is used to backannotate Allegro Design Entry HDL schematics. You cannot change this property interactively. The LOGICAL_PATH property is supported on NETCLASS objects.

MAKE_BASE

When two signals are aliased or synonymed, Allegro Design Entry HDL chooses one of the signal names as the base signal. The name of the base signal becomes the name of corresponding physical net in the Allegro PCB Editor. You may want the name of a particular aliased or synonymed signal to pass to the Allegro PCB Editor as the physical net name.

Attach the `MAKE_BASE=TRUE` property to a signal to declare the name of the signal as the base signal name for all its aliases or synonyms. You can also add the suffix `\BASE` to a signal name to declare it as the base signal name.

MATERIAL

This property is obsolete and will be removed in a future release.

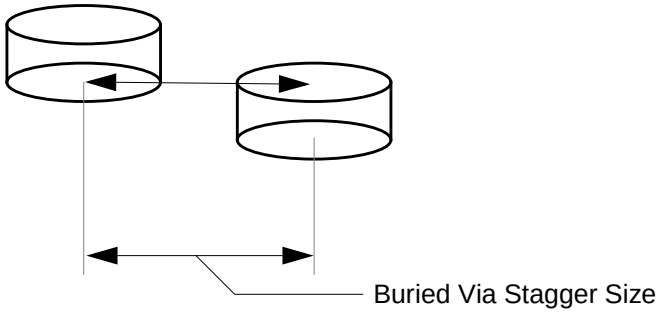
MAX_BOND_LENGTH

The MAX_BOND_LENGTH property, attached to a net or connect line, specifies the maximum length of bonding wire for a net. A bonding wire is any connect line on an *Etch*

subclass of type bonding wire. The MAX_BOND_LENGTH property is only supported at board level.

MAX_BVIA_STAGGER

The MAX_BVIA_STAGGER property, attached to a net, specifies the maximum center-to-center distance between the connect point of one pin or blind/buried via (the pin or via's x, y location) and the connect point of the other, where the two pins or vias are on the same net and have a single connect line joining them. The MAX_BVIA_STAGGER property has the same membership objects as the Physical and Spacing properties.



MAX_EXPOSED_LENGTH

The MAX_EXPOSED_LENGTH property, attached to a net or an ECset, is a length check for the total length of etch allowed on the outer etch/substrate subclasses that exist between the plane layers and air.

Any and all external etch lengths are added and compared to the value in constraints. (External means any layer that does not have a plane between it and air.)

```
TOP     <-------- external
INNER1  <-------- external
VCC     Plane
INNER2
INNER3
GND     Plane
INNER4 <--------- external
BOTTOM <--------- external
```

MAX_FINAL_SETTLE

The MAX_FINAL_SETTLE property, attached to a net, overrides the *Maximum Crosstalk Final Settle* electrical constraint. It defines the maximum final settle delay for the driver/receiver pin-pairs in the Xnet of which this net is part. A maximum final settle delay for a rising

(falling) edge is the time from when the driver starts switching to when the receiver passes up (down) through the Vil (Vih) switching threshold for the final time.

The format of the MAX_FINAL_SETTLE property in the $A_PR0PERTIES section of the netlist is:

```
MAX_FINAL_SETTLE [[pin1]:[pin2]:delay:]... ; netname
```

The legal values for *pin1* and *pin2* are:

`pin1:pin2`	Applies the given delay constraint to the specified pin-pair where each pin is defined as `refdes.pin#`. The pin-pair must form a driver/receiver pair in the Xnet of which this net is a part.
`AD:AR`	Applies the given delay constraint to each driver/receiver pin-pair combination in the Xnet of which this net is a part.
`delay`	Applies the maximum switch delay constraint to the specified driver/receiver pin-pairs. This is a decimal number defining delay in nanoseconds. You must type in ns, for example, AD:AR:3ns:3ns.
`netname`	Specifies the name of a net that is part of the Xnet to which the MAX_FINAL_SETTLE property is applied.

MAX_LINE_EXIT_ANGLE

For internal use only.

MAX_LINE_WIDTH

The MAX_LINE_WIDTH property, attached to a net, sets the maximum width of a cline. The value of this property is in design units. DRC uses the MAX_LINE_WIDTH property.

Use this property to override the *Max line width* rule in a physical constraint set. Allegro PCB Editor sets the precedence for the MAX_LINE_WIDTH property as follows:

If You Set...	Allegro PCB Editor Uses...

The MAX_LINE_WIDTH property to a net. and The *Max line width* rule in a physical constraint set.	The value of the MAX_LINE_WIDTH property.
The *Max line width* rule in a physical constraint set.	The value of the physical constraint, *Max line width*.

MAX_OVERSHOOT

The MAX_OVERSHOOT property, attached to a net, overrides the *Maximum Overshoot* electrical constraint. This is the maximum voltage overshoot tolerated by the net. The default value is `600` mV.

MAX_PARALLEL (formerly PARALLELISM)

The MAX_PARALLEL property, attached to a net or connect line (cline), overrides the *Max parallel* electrical constraint. The value is a character string of up to four different lengths and distances. Separate each length-distance value by a colon (`:`). Separate each length-distance pair by a semicolon (`;`). For example:

```
<len1>:<sep1>;<len2>:<sep2>;<len3>:<sep3>;<len4>:<sep4>
```

When determining whether segments of a line violate these constraints, the Allegro PCB Editor first look at the distances (`sep`*n*) of the segments from transmitter nets to segments of the receiver net, and categorize them according to the distances you defined.

For example, you define distances of `25` mils (`sep1`), `50` mils (`sep2`), `75` mils (`sep3`), and `100` mils (`sep4`). Each receiver segment's distance from transmitter segments is put into one of these distance "buckets." Segments that are between 1–25 mils from another net are put in the 25-mil "bucket." Segments that are between 26-50 mils from a net are put in the 50 mils "bucket," and so on.

Once all the line segments have been categorized, the Allegro PCB Editor total the line lengths for each distance bucket and determines if the length (`lenn`) you defined has been violated.

For example, the length-distance pairs you define could be `100` mil:`25` mil; `200` mil:`50` mil;`300` mil:`75` mil;`400` mil:`100` mil. In the 25-mil distance bucket, for instance, the total length

of those segments must be 100 mils or less. If they are not, a DRC error occurs. The DRC reports only the first violation it finds.

The MAX_PARALLEL property is also used with z-axis auditing of parallelism. (See `parallel` in the *Allegro PCB and Package Physical Layout Command Reference* for details on the `parallel` batch command.) Due to the manner in which z-axis auditing of parallelism is performed, the results (DRC detection) may not appear consistent. For example, DRC typically makes same-layer checks only; however, segments on adjacent layers that are exactly parallel in the z-axis produce DRC errors unless they are separated by a shield layer.

Before conducting a z-axis audit, consider the following conditions:

- Are only layers that are directly adjacent to each other being audited?

  This condition is defined as two etch/conductor layers separated by a single dielectric layer.

- What is the thickness of the dielectric layer?

- Does a shield layer separate the conductive layers being audited?

Your determination of these conditions influence the outcome of the z-axis audit. This becomes clear when you consider how a parallelism gap for two segments that are not on the same layer is computed:

```
gap = sqrt (xygap*xygap + zgap*zgap)
```

where `xygap` is the gap in the *x* or *y* dimension and `zgap` is the separation between layers

The following examples illustrate different scenarios:

Example 1

Two clines are positioned in exactly the same xy axis on adjacent layers TOP and INT1 (therefore, the parallelism gap is the zgap value). The nets to which the clines are attached are assigned to the same ECset that has a parallelism rule of 1000-mil length at a gap of 30 mils. In this example, a DRC is generated between the two nets because the gap between the clines is the thickness of the dielectric (30 mils) between layers TOP and INT1. If you change the parallelism constraint to a 29-mil gap, the DRC for the zgap value is eliminated.

Example 2

The cline on layer INT1 in the first example is moved to INT2, and the parallelism constraint is set to a gap of 55 mils. The zgap value between the two nets is now:

TOP - INT1 dielectric thickness	=	30 mils
INT1 thickness	=	1.2 mils
INT1 - INT2 dielectric thickness	=	23.0 mils
		_____
Total thickness	=	54.2 mils

The difference between the thickness and the gap constraint (0.8 mils) results in a DRC. If you change the parallelism constraint to a 54-mil gap, the DRC for the zgap value is eliminated.

Example 3

The cline on layer INT2 in the second example is moved to BOTTOM/BASE, and the parallelism constraint is set to a gap of 200 mils. This increases the zgap between the nets to 79.6 mils (the sum of the thickness of the dielectric and conductive layers between TOP/SURFACE and BOTTOM/BASE). Despite the disparity between the actual thickness and the gap constraint (123.4 mils), no DRC is generated because a shield layer separates the top and bottom layers; therefore parallelism between these layers is not considered.

MAX_PEAK_XTALK (formerly MAX_PEAK_CROSSTALK)

The MAX_PEAK_XTALK property, attached to a net, defines the high- and low-state maximum peak crosstalk constraints. The value is a string. The format of the string is:

<*high_value*>:<*low_value*>

See also MAX_XTALK (formerly MAX_CROSSTALK).

MAX_POWER_DISSIPATION

This property, attached to component definitions and instances, is used by a third-party thermal analysis tool. The value of this property is a string and specifies maximum power rating. If a user property already exists using this name, it becomes a system property.

MAX_SSN

The MAX_SSN property, attached to a net, overrides the *Maximum Simultaneous Switching Noise* electrical constraint. It is the maximum noise allowed on a net due to simultaneous switching.

MAX_UNDERSHOOT

The MAX_UNDERSHOOT property, attached to a net or cline, specifies the maximum undershoot tolerated by this net. Undershoot is a voltage swing back into the mid-range after the nominal steady state high or low level has been crossed. The default value is `250` mV.

MAX_VIA_COUNT

The MAX_VIA_COUNT property, attached to a net, specifies the maximum via count for a net.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

MAX_XTALK (formerly MAX_CROSSTALK)

The MAX_XTALK property, attached to a net or cline, specifies the maximum crosstalk summed across all neighbor nets tolerated by this net. Crosstalk is the transmission of a signal from one etch/conductor trace to another by electromagnetic field effects. On a printed circuit board/multi-chip module, parallel etch/conductor can exhibit significant crosstalk.

Backward crosstalk is when an active line that causes noise in a parallel trace propagates in the opposite direction as the active trace signal.

The value is a string. The format of the string is:

`<high_value>:<low_value>`

See also MAX_PEAK_XTALK (formerly MAX_PEAK_CROSSTALK).

MERGE_NC_PINS

The MERGE_NC_PINS property, attached to a component or symbol, is used when you have the NC_PINS property specified in the `chips.prt` or `ptf` file and you need to add a few

more pins (as specified by the NC_PIN property) in a Physical Part Table (PPT), in the `ptf` file, or in the schematic instance.

Note: You can define only one MERGE_NC_PINS or MERGE_POWER_PINS property at any level: schematic instance, Physical Part Table (`ptf`) file, or `chips.prt` files.

Syntax

`MERGE_NC_PINS = ( pin#,pin#,pin#...)`

Examples

Given the following property assignments:

...in the schematic:	...in the `ptf` file:	...in the `chips.prt`:
MERGE_NC_PINS='(1F, 1G)'	MERGE_NC_PINS='(1D,1E)'	NC_PINS='(1A,1B,1C)'

The primitive for the schematic instance in the `pstchip.dat` file has NC_PINS = '(1A,1B,1C,1D,1E,1F,1G)'.

The MERGE_NC_PINS property on the PPT is applied to the NC_PINS property on the `chips.prt` creating an intermediate NC_PINS= '(1A,1B,1C,1D,1E)'.

The MERGE_NC_PINS on the schematic instance is applied to the intermediate NC_PINS from the previous step creating NC_PINS = '(1A,1B,1C,1D,1E,1F,1G)'.

The MERGE_NC_PINS property has the following attributes in the default `cdsprop.paf` file:

`inherit(cell), permit(cell), uppercasevalue`

This property is now inherited by all components of the hierarchical block of which they form a part.

The MERGE_NC_PINS property and its values always appear in uppercase letters in the `pstxprt.dat` and `pstchip.dat` files, irrespective of the case you specify on the schematic instance, `ptf` file, or `chips.prt` files.

MERGE_POWER_PINS

You may need to add a few pins to an existing POWER_PINS or NC_PINS property. Adding the POWER_PINS or NC_PINS property at a higher level re-specifies the entire property.

Currently, there is no way to add more pins to an existing NC_PINS property. Instead two new properties, MERGE_POWER_PINS and MERGE_NC_PINS, provide this capability.

```
MERGE_NC_PINS='(1F, 1G)' MERGE_NC_PINS='(1D,1E)' NC_PINS='(1A,1B,1C)'
```

Use the MERGE_POWER_PINS property to add more power pins rather than override the existing POWER_PINS definition as would be the case if you used the POWER_PINS property. Refer to POWER_PINS on page 118 for more information.

Syntax

```
MERGE_POWER_PINS = (<supply>:pin# [,pin#...];[<supply>:pin#[,pin#...])
```

Example

Given the following assignments:

...in the `ptf` file:	...in the `chips.prt`:
MERGE_POWER_PINS=(VCC:15);	POWER_PINS=(VCC:14; GND:7)

The MERGE_POWER_PINS property has the following attributes defined in the default `cdsprop.paf` file:

```
inherit(cell), permit(cell), uppercasevalue
```

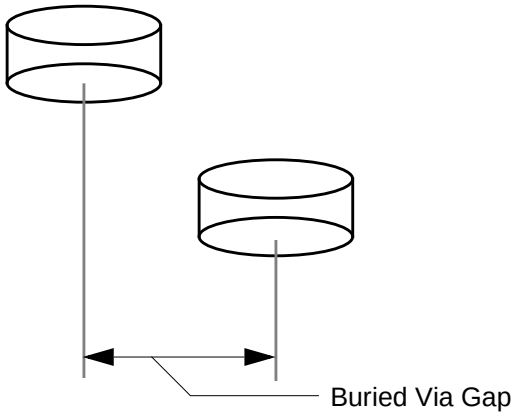
This property is now inherited by all components of a hierarchical block of which they form a part.

The MERGE_POWER_PINS property and its values always appear in uppercase letters in the `pstxprt.dat` and `pstchip.dat` files, irrespective of the case you specify on the schematic instance, `ptf` file, or `chips_prt` files.

MIN_BVIA_GAP

The MIN_BVIA_GAP property, attached to a net, specifies the minimum center-to-center spacing between the connect points of two buried vias that do not share a common layer. The

MIN_BVIA_GAP property has the same membership objects as the Physical and Spacing properties.

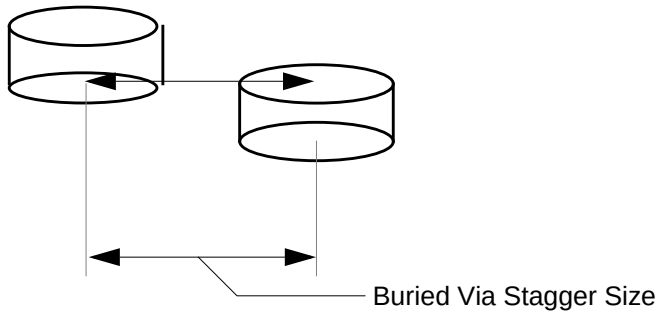


MIN_BOND_LENGTH

The MIN_BOND_LENGTH property, attached to a net or connect line, specifies the maximum length of bonding wire for a net. A bonding wire is any connect line on an *Etch* subclass of type bonding wire. The MIN_BOND_LENGTH property is only supported at board level.

MIN_BVIA_STAGGER

The MIN_BVIA_STAGGER property, attached to a net, specifies the minimum center-to-center distance between the connect point of one pin or via (the x,y location of the pin or via) and the connect point of the other, where the two pins or vias are on the same net and have a single connect line joining them. The MIN_BVIA_STAGGER property has the same membership objects as the Physical and Spacing properties.



MIN_FIRST_SWITCH

The MIN_FIRST_SWITCH, property, attached to a net, overrides the *Minimum Crosstalk First Switch* electrical constraint. It defines the minimum first switch delay for driver/receiver pin-pairs in the Xnet of which this net is a part. A minimum first switch delay for a rising (falling) edge is the time from when the driver starts switching to when the receiver first passes the Vil (Vih) switching threshold.

The format of the MIN_FIRST_SWITCH property in the $A_PROPERTIES section of a netlist is

```
MIN_FIRST_SWITCH [[pin1]:[pin2]:delay:]... ; netname
```

The legal values for *pin1* and *pin2* are:

`pin1:pin2`	Applies the given delay constraint to the specified pin-pair where each pin is defined as refdes.pin#. The pin-pair must form a driver/receiver pair in the Xnet of which this net is a part.
`AD:AR`	Applies the given delay constraint to each driver/receiver pin-pair combination in the Xnet of which this net is apart.
`delay`	Specifies the minimum first switch delay constraint that is applied to the specified driver/receiver pin-pairs. This is a decimal number defining delay in nanoseconds. You must type in ns, for example, AD:AR:3ns:3ns.
`netname`	Specifies the name of a net that is part of the Xnet to which the MIN_FIRST_SWITCH property is applied.

MIN_HOLD

The MIN_HOLD property, applied to a net or pin, defines the minimum hold time of a data signal relative to a clock signal. The value of the property is in nanoseconds and can be attached to either the data net or a pin of the data net. This property is used by the *Timing Setup/Hold* tab of the Constraint Manager.

MIN_LINE_WIDTH

The MIN_LINE_WIDTH property, attached to a net, sets the minimum width of a net or line. Use this property to override the *Min line width* rule in a physical constraint set. This property also overrides the *Line Width* electrical constraint for a differential pair, if set. The value of this property is in design units.

The MIN_LINE_WIDTH property is used by automatic routing and DRC.

For differential pairs, minimum line width is determined as follows:

If You Set....	Allegro PCB Editor Uses...
All of the following: ■ The MIN_LINE_WIDTH property on the differential pair ■ The *Line Width* constraint in an electrical constraint set on the differential pair ■ *The Min line width* rule in a physical constraint on either of the nets in the pair	The value of MIN_LINE_WIDTH property.
Both of the following: ■ The *Line Width* constraint in an electrical constraint set on the differential pair ■ *The Min line width* rule in a physical constraint on either of the nets in the pair	The value of the electrical constraint, *Line Width*.
■ *The Min line width* rule in a physical constraint on either of the nets in the pair	The value of the physical constraint, *Min line width.*

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

MIN_NECK_WIDTH

The MIN_NECK_WIDTH property, attached to a net, sets the minimum neck width of a net or line. Use this property to override the *Min neck width* rule in a physical constraint set. This property also overrides the *Neck Width* electrical constraint for a differential pair, if it is set. The value of this property is in design units. The MIN_NECK_WIDTH property has the same membership objects as the Physical and Spacing properties.

The MIN_NECK_WIDTH property is used by automatic routing and DRC checking.

For differential pairs, minimum neck width is determined as follows:

If You Set...	The Allegro PCB Editor Uses....
All of the following: ■ The MIN_NECK_WIDTH property on the differential pair ■ The *Neck Width* constraint in an electrical constraint set on the differential pair ■ *The Min neck width* rule in a physical constraint on either of the nets in the pair	The value of MIN_NECK_WIDTH property.
Both of the following: ■ The *Neck Width* constraint in an electrical constraint set on the differential pair ■ *The Min neck width* rule in a physical constraint on either of the nets in the pair	The value of the electrical constraint, *Neck Width*.
■ *The Min neck width* rule in a physical constraint on either of the nets in the pair	The value of the physical constraint, *Min neck width.*

MIN_NOISE_MARGIN

The MIN_NOISE_MARGIN property, attached to a net, overrides the *Minimum Noise Margin* electrical constraint. This is the minimum noise margin tolerated by this net. The default value is 0 mV.

MIN_SETUP

The MIN_SETUP property, attached to a net or pin, defines the minimum setup time of a data signal relative to a clock signal. The value of this property is in nanoseconds. The MIN_SETUP property is used by the *Timing Setup/Hold* tab of the Constraint Manager.

MIN_SHAPE_SIZE

The MIN_SHAPE_SIZE property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies in user units that any shape areas smaller than this property value are suppressed during automatic dynamic voiding. This may cause a shape to split into multiple shapes. This value is squared when calculating the minimum area. For example, if you enter 2 in the Shape Instance Parameters dialog box, the shape size is 4.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Void Controls* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

MODEL_DIR

The MODEL_DIR property, attached to a component or symbol, binds different instances of a component to different model libraries. This property specifies the directory that contains the model to be used for binding the component. The syntax used is:

```
MODEL_DIR = <path_to_the_model_library>
```

While specifying the location of the model library, ensure that only a forward slash (/) is used as a separator. Within the directory specified, you must have a file <*model_name*>.v that contains the model definition.

For example, consider that LS00 is instantiated in a design and the model used for simulating the LS00 component is ls00. If you add the MODEL_DIR property set to models_ver for instance-specific binding, then the models_ver directory must have a file ls00.v. Otherwise, the Allegro Design Entry HDL netlister generates a warning and ignores the MODEL_DIR property.

The values assigned to the MODEL_DIR property are case-sensitive.

Before using the MODEL_DIR property to specify the model library, ensure that the component binding is correct. If you do not want to use the default configuration, use the SIM_BIND_VIEW or SIM_MAP_VIEW property to specify the correct view for binding the components.

A part of the netlist that is generated after the MODEL_DIR property is attached to LS00 is shown below. The simulation model is ls00, and the value assigned to the MODEL_DIR property is wrapper_ver. The additions in the netlist because of the MODEL_DIR property are in bold typeface.

```
‘uselib dir = wrappers_ver libext=.v
  ls00 page1_i2  (.a(b),
        .b(unnamed_1_ls00_i2_b),
        .\y* (unnamed_1_ls00_i2_y));
  defparam page1_i2.size = 1;
‘uselib
```

Notice that when the MODEL_DIR property is attached to a component, a ‘uselib statement is added in the simulation netlist. The ‘uselib compiler directive specifies the location where Verilog-XL should look for the definitions of modules and user-defined primitives (UDPs) used in the schematic design. The model file extension should be .v. This is indicated by libext=.v in the ‘uselib statement.

MODEL_FILE

The MODEL_FILE property, attached to a component or symbol, is similar to the MODEL_DIR property, except that the MODEL_FILE property specifies the location of a verilog file containing the model definition for simulating the component. To specify the MODEL_FILE property on an instance, add the following:

```
MODEL_FILE = <path>/verilog.v
```

where *path* is the location of the verilog.v file containing the model definition. When specifying the path, be sure that you use the forward slash(/). Notice that the file specified by the MODEL_FILE property is always a verilog file.

Before using the MODEL_FILE property to specify the model library, ensure that the component binding is correct. If you do not want to use the default configuration, use the SIM_BIND_VIEW or the SIM_MAP_VIEW properties to specify the correct view for binding the components.

A part of the netlist that is generated after the MODEL_FILE property is attached to LS00 is shown below. The additions in the netlist because of the MODEL_FILE property are in bold typeface.

```
// begin instances
```

```
‘uselib file = ./models_ver/ls00/vlog_behavioral/verilog.v
  ls00 page1_i1  (._1A(/* unconnected */),
        ._1B(/* unconnected */),
        ._1Y(/* unconnected */),
        ._2A(/* unconnected */),
        ._2B(/* unconnected */),
        ._2Y(/* unconnected */),
        ._3A(/* unconnected */),
        ._3B(/* unconnected */),
        ._3Y(/* unconnected */),
        ._4A(a),
        ._4B(sel),
        ._4Y(unnamed_1_ls00_i1_y));
‘uselib
```

Note: You can specify either the MODEL_FILE property or the MODEL_DIR property on an instance. If both properties are specified on a single instance, the MODEL_DIR property is ignored.

The MODEL_DIR and MODEL_FILE properties, which are used for providing instance-specific model binding, are understood only by the Verilog-XL simulator. For instance-specific binding using NC Verilog simulator, use the VERILOG_LIB property.

NC_PINS

The NC_PINS property, attached to a component or symbol, specifies the assignment of pins which are present on a physical package but do not have any logical connections. These pins do not play any role in the logical simulation, but need to be taken into consideration during board layout.

The NC_PINS property has the following syntax:

NC_PINS=’(*pin list*,*pin list*, ...)’;

where *pin list* can be a single pin or a range of pins or any combination of the two. The range description can include alphanumeric pin designators and can be in increasing or decreasing order.

NEEDS_NO_SIZE

The NEEDS_NO_SIZE property, attached to a component, identifies those components that do not need the SIZE property. It is an error to attach a SIZE property to these components. The NEEDS_NO_SIZE property has the following functions:

- Specifies that the component to which it is attached does not need a SIZE property
- Generates an error message if a SIZE property is found

Many of the symbols in the standard libraries have this property attached to them. Most notable are the `NOT` and `MERGE` symbols. These symbols automatically conform to the widths of the signals to which they are attached. The presence of the SIZE property on these symbols is always an error since none of the pins of the symbol or the definition of the symbol uses the SIZE property.

You can attach the NEEDS_NO_SIZE property as a default symbol property, or attach it to the symbol when used in a schematic.

NET_SCHEDULE

The NET_SCHEDULE property, attached to a net or ECset, works with RATSNEST_SCHEDULE property and reflects if the DRC system ensures that the schedule is met when the net is routed. Values are `Verify` or `Do Not Verify`. For more Information, see RATSNEST_ SCHEDULE.

NET_SHORT

The NET_SHORT property, attached to a pin, via, or static shape, lets you connect two nets, such as GND and AGND, to a common pin or vias without a DRC error.

This property allows contact between two planes that have different net names and prevents a flagged or reported DRC error.

A DRC is reported if any connect lines other than the net of the actual pin, touch the origin point of the actual pin. The actual pin (or via) does not report a DRC for any logic objects that touch it.

Attempting to attach the property to a dynamic shape causes the following error to be output to the command line:

```
E- Invalid db element type
```

The syntax of the NET_SHORT property is:

```
<net 1>:<net 2>:….
```

For example:

```
NET_SHORT = GND1 : GND2 : GND3
```

NO_BACKANNOTATE

The NO_BACKANNOTATE property, attached to a component and specified in the schematic, prevents the update of properties in the schematic. For example, to disable the backannotation of all properties on an instance, specify the following property in the schematic:

```
NO_BACKANNOTATE = ALL
```

Using the NO_BACKANNOTATE Property

Use this property to prevent the backannotation of the master view of a subdesign. To preserve the hard SEC and hard PN properties in instances of a reused block, place the NO_BACKANNOTATE property on those instances in the schematic.

To disable backannotation on an instance:

➤ Add the NO_BACKANNOTATE property to the instance. This disables backannotation on the instance and each of its pins.

The NO_BACKANNOTATE property preserves the property value on the instance and discards the backannotation value.

Syntax

```
NO_BACKANNOTATE = property [property property ...]
```

Example

To prevent the reassignment of LOCATION for a particular instance, specify the NO_BACKANNOTATE property as NO_BACKANNOTATE=LOCATION.

To specify multiple properties:

➤ List each property separated by spaces:

```
NO_BACKANNOTATE=COMPONENT_WEIGHT LOCATION SEC
```

Note: You can add the NO_BACKANNOTATE property to a block. The NO_BACKANNOTATE property, added to a block, applies to its child blocks too.

If you add the NO_BACKANNOTATE = ALL property to a block and then package the design, Packager-XL displays an informational message stating that the NO_BACKANNOTATE = ALL property is set for the block and Packager-XL does not backannotate it. This message is recorded in the `pxl.log` file.

If you add the NO_BACKANNOTATE = ALL property to a read-only block and save it in Allegro Design Entry HDL, it generates an error stating that write permissions do not exist for the specified page. Allegro Design Entry HDL skips changes to such blocks.

NO_DRC

The NO_DRC property, attached to pins or vias, disables DRC checking against the pin or via.

NO_FILLET

The NO_FILLET property, attached to a net, pin, or via, prevents the creation or regeneration of fillets on these elements, even if the *Dynamic Fillets* option is enabled on the Pad and T Connection Fillet dialog box (available by choosing *Route – Gloss – Parameters*).

NO_GLOSS

The NO_GLOSS property, attached to a net, indicates that when you run the *Route – Gloss* menu command (`gloss` command), the Allegro PCB Editor do not gloss the specified net. Be sure that you set the value to `TRUE`.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net properties` command and the *Allegro Constraint Manager User Guide* for additional information.

NO_IO_CHECK

The NO_IO_CHECK property, attached to a library component or a pin, component, or net in a design, suppresses input and output checks on a particular component, pin, or net and takes these values:

`LOW`	Suppresses the “0 state” I/O check.
`HIGH`	Suppresses the “1 state” I/O check.
`BOTH` or `TRUE`	Suppresses both the “0 state” and the “1 state” I/O check.

Attaching the property to a net applies the property to all pins on the net. When used as a component property, it applies to all pins on the component. As a pin property, it applies only to that pin.

NO_LOAD_CHECK

The NO_LOAD_CHECK property, attached to a component or pin, suppresses device-loading calculations on a pin-by-pin basis and takes these values:

LOW	Suppresses the “0 state” I/O check.
HIGH	Suppresses the “1 state” I/O check.
BOTH or TRUE	Suppresses both the “0 state” and the “1 state” I/O check.

You can attach the NO_LOAD_CHECK property to library components or in your drawings. When used as a component property, it applies to all pins of the component. As a pin property, it applies only to the specified pin.

NO_LIN2SHAPE_FAT

The NO_LIN2SHAPE_FAT property, attached to a connect line, excludes the line from being fattened by the `fatten` command. Be sure that you set the value to TRUE.

NO_PIN_ESCAPE

The NO_PIN_ESCAPE property, attached to a reference designator, net, or pin, indicates that Allegro PCB Router should not place pin escapes on this component, net, or pin during routing. Be sure that you set the value to TRUE.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net_properties` command and the *Allegro Constraint Manager User Guide* for additional information.

NO_RAT

The NO_RAT property, attached to a net, indicates that the specified net does not display ratsnests for unconnected pins. Be sure that you set the value to TRUE.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net properties` command and the *Allegro Constraint Manager User Guide* for additional information.

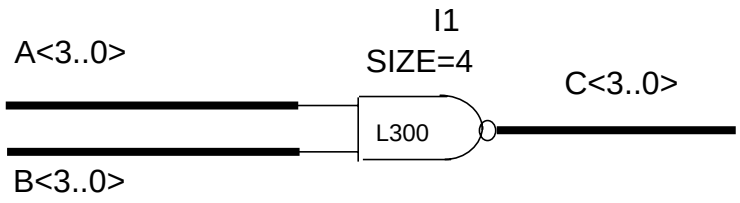
NO_REP_PRIM

The NO_REP_PRIM=TRUE property is attached to an instance:

- When you do not want the instance to be replicated in the netlist.
- When you do not want to pass the SIZE property with the correct value to the behavioral model.

Note: When there is no property NO_REP_PRIM=TRUE on an instance, the Design Entry HDL netlister automatically replicates the instance in the netlist with the actual value of the SIZE property specified on the instance.

In the figure below, an instance of component `ls00` has the property `SIZE=4`.



The instance is replicated four times in the netlist as below:

```
SN74LS00 page1_i1__1  (._1A(/* unconnected */),
      ._1B(/* unconnected */),
      ._1Y(/* unconnected */),
      ._2A(/* unconnected */),
      ._2B(/* unconnected */),
      ._2Y(/* unconnected */),
      ._3A(/* unconnected */),
      ._3B(/* unconnected */),
      ._3Y(/* unconnected */),
      ._4A(a[0]),
      ._4B(b[0]),
      ._4Y(c[0]));
```

```
SN74LS00 page1_i1__2  (._1A(/* unconnected */),
    ._1B(/* unconnected */),
    ._1Y(/* unconnected */),
    ._2A(/* unconnected */),
    ._2B(/* unconnected */),
    ._2Y(/* unconnected */),
    ._3A(/* unconnected */),
    ._3B(/* unconnected */),
    ._3Y(/* unconnected */),
    ._4A(a[1]),
    ._4B(b[1]),
    ._4Y(c[1]));

SN74LS00 page1_i1__3  (._1A(/* unconnected */),
    ._1B(/* unconnected */),
    ._1Y(/* unconnected */),
    ._2A(/* unconnected */),
    ._2B(/* unconnected */),
    ._2Y(/* unconnected */),
    ._3A(/* unconnected */),
    ._3B(/* unconnected */),
    ._3Y(/* unconnected */),
    ._4A(a[2]),
    ._4B(b[2]),
    ._4Y(c[2]));

SN74LS00 page1_i1__4  (._1A(/* unconnected */),
    ._1B(/* unconnected */),
    ._1Y(/* unconnected */),
    ._2A(/* unconnected */),
    ._2B(/* unconnected */),
    ._2Y(/* unconnected */),
    ._3A(/* unconnected */),
    ._3B(/* unconnected */),
    ._3Y(/* unconnected */),
    ._4A(a[3]),
    ._4B(b[3]),
    ._4Y(c[3]));
```

If you add the NO_REP_PRIM=TRUE property to the instance, the instance is not replicated in the netlist (see netlist sample below).

```
SN74LS00 page1_i1  (._1A(a[0]),
      ._1B(b[0]),
      ._1Y(c[0]),
      ._2A(a[1]),
      ._2B(b[1]),
      ._2Y(c[1]),
      ._3A(a[2]),
      ._3B(b[2]),
      ._3Y(c[2]),
      ._4A(a[3]),
      ._4B(b[3]),
      ._4Y(c[3]));
```

Be sure that you set the value to TRUE.

NO_RIPUP

The NO_RIPUP property, attached to a net, indicates that Allegro PCB Router should not remove the etch from the specified net. However, if you add any connections to a net after you assign this property, the router removes the etch from these new connections. Be sure that you set the value to TRUE.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the net_properties command and the *Allegro Constraint Manager User Guide* for additional information.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

NO_ROUTE

The NO_ROUTE property, attached to a net or reference designator (component), indicates that Allegro PCB Router should not route any missing connections on the specified net. Be sure that you set the value to TRUE.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the net_properties command and the *Allegro Constraint Manager User Guide* for additional information.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

NO_SHAPE_CONNECT

The NO_SHAPE_CONNECT property, attached to a pin or via, operates as follows:

- When you attach this property to a pin, the Allegro PCB Editor do not create a connection between the pin (that passes through a shape with the same net) and a shape.
- When you attach this property to a via, the Allegro PCB Editor do not create a connection between the via (that passes through a shape that is on the same net) and a shape.

Be sure that you set the value to TRUE.

NO_SM_COVERAGE_CHECK

This property is scheduled for implementation in a future release.

NO_SWAP_COMP

The NO_SWAP_COMP boolean property, attached to component instances, indicates that the `swap components` command does not allow the symbol associated with this component to be swapped. Be sure that you set the value to TRUE.

NO_SWAP_GATE

The NO_SWAP_GATE property, attached to a reference designator or function designator (gate), specifies the functions within the component that cannot be swapped. The function stays fixed in its current slot in the component. Be sure that you set the value to TRUE.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

NO_SWAP_GATE_EXT

The NO_SWAP_GATE_EXT property, attached to a function designator, specifies that the function cannot be swapped with one from another component. However, the function can be swapped among slots within its current component. Be sure that you set the value to TRUE.

NO_SWAP_PIN

The NO_SWAP_PIN property, attached to a reference designator, function designator (gate), or pin, specifies that the pins on this component or function cannot be swapped, either interactively or automatically. Be sure that you set the value to TRUE.

NO_TEST

The NO_TEST property, attached to a net, indicates that the Allegro PCB Editor cannot add test points during test point generation. Be sure that you set the value to TRUE.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net properties` command and the *Allegro Constraint Manager User Guide* for additional information.

Note: The NO_TEST property is listed as *Prohibit* in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager

NO_VIA_CONNECT

The NO_VIA_CONNECT property, attached to a pin or via, specifies that padstacks are not connected even when they overlap.

NO_WIREBOND

The NO_WIREBOND property, attached to nets or pins, causes the layout tool to automatically exclude those items from wire bond add operations.

NO_XNET_CONNECTION

To eliminate the creation of Xnets from the nets connected to pins defined by a model containing a PinConnections section, you must attach the NO_XNET_CONNECTION to the component you are attaching the ESpice model to.

NODRC_COMPONENT_BOARD_OVERLAP

The NODRC_COMPONENT_BOARD_OVERLAP property, attached to a shape (package boundary), suppresses DRC errors when components overlap the package keepin. When you add the NODRC_COMPONENT_BOARD_OVERLAP property to the place boundary of

a package or mechanical symbol at the end of the design cycle (recommended), the editor removes the Package-to-Place Keepin Spacing DRC marker.

The NODRC_COMPONENT_BOARD_OVERLAP property is not supported at the schematic level.

NODRC_ETCH_OUTSIDE_KEEPIN

The NODRC_ETCH_OUTSIDE_KEEPIN property, attached to an etch item, for example, a shape, rectangle, line, or cline, suppresses DRC errors when the etch element is outside or crosses the route keepin. When you add the NODRC_ETCH_OUTSIDE_KEEPIN property to an etch element at the end of the design cycle (recommended), Allegro PCB Editor removes the existing <*etch element*> to Route Keepin DRC marker.

The NODRC_ETCH_OUTSIDE_KEEPIN property is not supported at the schematic level.

NODRC_SYM_SAME_PIN

The NODRC_SYM_SAME_PIN property, attached to a board, symbol instance, or symbol definition, disables pin-to-pin conductive layer checking between pins of the same symbol. Pin-to-pin checking always occurs in a symbol editor (.dra). Pin-to-pin spacing checks between different symbols remain unaffected.

You can:

- Build this property into symbols at the package level by placing the property on the symbol drawing object. It is assumed that the symbol is built correctly in the library. No pin-to-pin spacing checks occur when all symbols of that type are placed on the board.
- Place this property at the design level (.brd or .mcm) on symbol instances by choosing *Symtype* in the *Find by Name* box in the Find Filter. No pin-to-pin spacing checks occur on these symbols.
- Place the property on symbol definitions. No pin-to-pin spacing checks occur on these symbols.
- Place the property at the root design level (.brd or .mcm) on the drawing object by choosing *Drawing* in the *Find by Name* box in the Find Filter. All symbols on the board have pin-to-pin spacing checks disabled.

If a symbol's pin is modified on the board, for example, you modify a padstack by instance, then the Allegro PCB Editor ignores this property and generates a normal DRC.

If you add the NODRC_SYM_SAME_PIN property to the symbol once it is placed in the board, running the *Update – Symbols* menu command (`refresh_symbol`) removes the property.

When you add the NODRC_SYM_SAME_PIN property at the end of the design cycle (recommended), Allegro PCB Editor removes the existing same component Pin-to-Pin DRC markers.

The NODRC_SYM_SAME_PIN property is not supported at the schematic level.

Note: Exploding pins cancel the property.

NODRC_VIAS_OUTSIDE_KEEPIN

The NODRC_VIAS_OUTSIDE_KEEPIN property, attached to a via, suppresses DRC errors when vias are outside a route keepin. When you add the NODRC_VIAS_OUTSIDE_KEEPIN property to a via at the end of the design cycle (recommended), Allegro PCB Editor removes the Via-to-Route Keepin Spacing DRC marker.

The NODRC_VIAS_OUTSIDE_KEEPIN property is not supported at the schematic level.

OK_DANGLE

The OK_DANGLE boolean property suppresses dangling clines or vias to suppress reporting them in the dangling line/via report. This property can be added to vias and clines.

OK_UNASSIGNED_SHAPE

The OK_UNASSIGNED_SHAPE property suppresses shapes without nets when you run the Shape Without Nets Report. This property can be added to static and dynamic shapes that you do not plan on assigning to a net.

OUTPUT_LOAD

The OUTPUT_LOAD property, attached to a pin, specifies the load presented by a pin when the pin is an output pin. If a pin does not have the OUTPUT_LOAD property attached, it is assumed to be an input pin. When an output pin is bi-directional or tri-state, you must attach both the INPUT_LOAD and OUTPUT_LOAD properties.

OUTPUT_TYPE

The OUTPUT_TYPE property, attached to the output pins of open-collector, open-emitter, and tri-state devices, serves the following purposes:

- Allows you to tie two or more output pins together
- Specifies the type of output so that only compatible outputs are connected together (an OC output cannot be tied to a TS output)
- The second value of the property specifies the logic function that is created when you tie two or more output pins together

The OUTPUT_TYPE property provides information that Packager-XL needs.

Each output pin that can be connected to other output pins must have an OUTPUT_TYPE property. The property value specifies the pin type and the logic function created by tying the outputs together. The syntax of OUTPUT_TYPE property is:

```
OUTPUT_TYPE='(output type,logic function)';
```

The *output type* variable can be `open collector`, `open emitter`, or `tri-state` (TS). The *logic function* variable can be AND, OR, or `tri-state` (TS). Make sure that there is no space after the comma in the property value.

You can combine the output type and logic functions as follows:

`OUTPUT_TYPE='(OC,AND)'`	Open Collector, AND logic function
`OUTPUT_TYPE=(OE,OR)`	Open Emitter, OR logic function
`OUTPUT_TYPE= (TS,TS)`	Tri-state, tri-state logic function

Tri-state pins need both INPUT_LOAD and OUTPUT_LOAD properties. When a pin is in tri-state mode, the tri-state loading is specified as the INPUT_LOAD.

PACKAGE_HEIGHT_MAX and PACKAGE_HEIGHT_MIN

The PACKAGE_HEIGHT_MAX and the PACKAGE_HEIGHT_MIN properties are attached to placebound rectangles or shapes at the symbol definition or instance level or to component definitions.

The component definition property HEIGHT in Allegro Design Entry HDL maps to PACKAGE_HEIGHT_MAX on the component definition when the logic data is imported into the physical editor.

If your design model controls height restrictions from the physical side, manage the height properties from the physical symbol library. If your design processes require front-end managed heights, then the component-definition method should be used. Designs may utilize a mixed model, but symbol-based heights take precendence over component heights. The component model does not support multiple heights.

PACK_IGNORE

The PACK_IGNORE property, attached to a component, identifies the parts that are ignored when packaging the design. You can specify the PACK_IGNORE property only in the schematic.

The PACK_IGNORE property excludes the schematic instances from being packaged as follows:

- If the schematic instance is a primitive, it is ignored and therefore is not packaged. The netlist output from Packager-XL does not reference this part. Nets connected to this schematic instance are left unconnected.
- If the instance is a hierarchical block, then the underlying hierarchy is removed.

Use of the PACK_IGNORE property may result in unconnected nets.

Syntax

```
PACK_IGNORE = <value>
```

Example

```
PACK_IGNORE = TRUE;
```

Note: Packager-XL does not look at the value of the PACK_IGNORE property. If the property is present, the instance is ignored.

PACK_SHORT

The PACK_SHORT property, attached to a pin, specifies the shorting of pins of a part. You can specify the PACK_SHORT property in the schematic.

Multiple groups of pins, each group having two or more pins, can be shorted. You must use logical schematic pin names with this property.

You should use the PACK_SHORT property with the PACK_IGNORE property to specify shorting for nets attached to the instance.

Syntax

PACK_SHORT=(<*group1*>)(<*group2*>)[<*group3*>]

where <*group*> indicates (*logicPin1*, *logicPin2* ... [*logicPinN*])

Example

Consider the assignment, PACK_SHORT = (A1, B1, Y1) (A2, B2). It shorts together:

- The nets attached to logic pins A1, B1, and Y1.
- The nets attached to pins A2 and B2.

PACK_TYPE

The PACK_TYPE property, attached to a component, specifies the type of package used for a part. Packager-XL uses the PACK_TYPE property to determine which physical part entry in the chips.prt file to use when selecting physical parts.

PART_NAME

The PART_NAME property, attached to a component, specifies the name of a primitive component. When Packager-XL chooses a primitive component, it first looks at the component name. Normally, the component name is the name of the primitive component. Sometimes, you may want the primitive component name to be different from the logical component name.

For example, the LSTTL library components are called LS00, LS01, LS02, and so on. However, the parts are known in the Compiler expansion file as 74LS00, 74LS01, and so on, since these are more explicit names. The '74' that is left off the logical component name makes the name easier to type. Giving the LS00 the PART_NAME 74LS74 is incorrect, as these two components serve different purposes.The PART_NAME properties are found in the body section of the chips file for the component.

PART_NUMBER

The PART_NUMBER property, attached to a component in a physical part table (PPT) file, is an injected property used by the CIMLink interface for Universal Test equipment, as well as to update the USER PART NUMBER label on the mechanical symbol. An injected property appears on part rows in the PPT. It further defines the unique characteristics of each unique line definition of a part.

You may change, add, or delete the PART_NUMBER property or other injected properties in the part table without affecting the schematic because when the schematic is packaged, the part table definition of the injected property takes precedence over the instance definition.

CIMLink is not supported in versions 14.0 and higher.

PATH

The PATH property, automatically attached to each component in a drawing as it is created, uniquely identifies it. The values are *I1, I2, I3*. These are unique across a page of a design.

If you modify the value of the PATH property on a component, do not use the back slash (\) in the value of the PATH property.

PHYS_DES_PREFIX

The PHYS_DES_PREFIX property, attached to a component, specifies the reference designator prefix for the default Packager-XL naming pattern.

You can specify the PHYS_DES_PREFIX property in the `chips` file, the `ptf` file, or the schematic.

Using PHYS_DES_PREFIX

The PHYS_DES_PREFIX property identifies classes of parts such as resistors (R), capacitors (C), and inductors (L).

By default, Packager-XL searches for the PHYS_DES_PREFIX property when assigning a reference designator value. If it does not find the property, it assumes the value of the PHYS_DES_PREFIX property as ‘U’.

You can use the REF_DES_PATTERN directive or property to modify the default naming scheme.

Syntax

```
PHYS_DES_PREFIX = <value>
```

Example

```
PHYS_DES_PREFIX = R
```

PHYSICAL_PATH

Reserved for future support of physical path names for place by schematic page number.

PIN_DELAY

You can include pin delay in DRC calculations for DIFFERENTIAL PAIR PHASE TOLERANCE, PROPAGATION DELAY, and RELATIVE PROPAGATION DELAY by assigning the PIN_ DELAY property to component instance or definition pins.

The PIN_ DELAY property specifies the time delay or length from the internal package connection to the pin's mounting layer. Use the PIN_DELAY property to factor inter-package delay into timing requirements.

Enter the value for the PIN_ DELAY property in length or time, which is multiplied by the *Pin Delay Propagation Velocity Factor,* a constant used to convert between time and length units.

To factor pin delay into these DRC calculations, you enable the *Pin Delay Include in All Propagation Delays and in DiffPair Phase Checks* field in the *Options tab* of the Electrical Constraints Dialog Box, accessible by running the `cns electrical` command in the Allegro PCB Editor, described in the *Allegro PCB and Package Physical Layout Command Reference.*

You can use the `pin delay out` command to create a file containing pin delay values as defined by this property from an external source, and then use the `pin delay in` command to import the values in this file and assign them to component instance pins in another design.

For more information on pin delay, see the *Creating Design Rules* user guide in your documentation set.

PIN_DELAY_ENABLED

The PIN_DELAY_ENABLED boolean property, attached to a design, enables pin delay accounting for differential pairs, relative propagation delay, and delay DRC checks. Make sure that the value is set to TRUE.

PIN_ESCAPE

The PIN_ESCAPE property, attached to a reference designator or a pin, specifies that Allegro PCB Editor places a pin escape on every pin of this component when the automatic router runs. Be sure that you set this value to TRUE.

PIN_GROUP

The PIN_GROUP property, specified in the `chips` file, identifies swappable pins on a part.

Using the PIN_GROUP Property

Pins in a group are swappable if they are logically equivalent and belong to the same section. If you swap two nets between two pins within a swappable group (two or more logical pins with the same PIN_GROUP value), you do not alter the logical function of the circuit.

Syntax

`PIN_GROUP=<value>`

Example

```
PIN_GROUP = A
```

A common example of pin swapping occurs for the inputs of a NAND gate. The two input pins are physically equivalent in terms of logical function, loading, and propagation delay from input to output. If you swap the nets to the input pins, the logic of the circuit does not change.

PIN_NAME

The PIN_NAME property, attached to a symbol, defines the pin's name. The value is `STRING`.

PIN_NUMBER

The PIN_NUMBER property, attached to a pin, provides the logical-to-physical mapping for the pin. Each pin of every library component (and all package outlines of a component) must have a PIN_NUMBER property. Packager-XL obtains the following information from the PIN_NUMBER property:

- Physical pin number of the pin
- Number of sections of the component in a package
- Pin numbers for each section

There must be one pin number for each pin in each section. For example, if a package has four sections (as in `LS00`), there must be four pin numbers associated for each pin. The packager prints an error message when a pin is found without a PIN_NUMBER property or without the correct number of pins per section.

PIN_SIGNAL_MODEL

The PIN_SIGNAL_MODEL property, attached to a pin, defines a pin-level override for a buffer model used in signal integrity simulations. This property is used internally by the software for implementing programmable buffers. Cadence recommends that you do not use this property to specify your own pin-level overrides for a buffer model.

PINUSE

The PINUSE property, attached to a pin, indicates its pin type. This value overrides any pinuse value specified in the device file. Pinuse code values are: `IN`, `OUT`, `BI`, `TRI`, `OCA`, `OCI`, `POWER`, `GROUND`, `NC`, and `UNSPEC`.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

PLACE_TAG

The PLACE_TAG property, attached to a reference designator (component), specifies that Allegro PCB Editor places the component during the next automatic placement session. Be sure that you set the value to `TRUE`. Make sure that you remove the PLACE_TAG property before placing another group of components.

PLATING

The PLATING property, attached to a shape, specifies that Allegro PCB Editor attach the shape to a plating bar.

PN

The PN property, attached to a pin, assigns a physical pin number (specified in the `chips.prt` file) to a logical pin in the schematic. This property is used to perform manual pin swaps on physical parts. Use the *Component – Swap Pins* menu command in Allegro Design Entry HDL to assign this property.

PNN

The PNN property is automatically generated by Packager-XL for those logical nets that have their physical nets names different from their logical nets. PNNs are generated for plotting purposes. You can annotate the PNN property to display the physical net name on the schematic canvas.

PORT_ORDER

The PORT_ORDER property, specified in the Verilog map file, is used if you check the *Position Mapping* box in the *Netlist* tab of the Allegro Design Entry HDL Digital Simulation Interface. The result is that the Allegro Design Entry HDL pins are mapped to model ports by position in the netlist.

Syntax

`PORT_ORDER = (`*`port1, port2 ...`*`);`

Use this property to specify the port ordering for a module described in an external library.

- If the port is a vector, a full description for the port (such as `A[3:0]`) is required. The vector information is useful for Allegro Design Entry HDL to reconstruct the bus connected to the port.
- If the PORT_ORDER property is not specified for a part in an external library, Allegro Design Entry HDL uses the PIN_NUMBER information for making the connection by position. The PIN_NUMBER information is read from the `chips.prt` file.

- If you check the *Position Mapping* box in the *Netlist* tab of the Allegro Design Entry HDL Digital Simulation Interface and you have not specified a PORT_ORDER or a PIN_NUMBER property, Allegro Design Entry HDL displays an error message.

Note: To obtain the port order information:

1. Read the `verilog.v` file to obtain the port names.
2. If there is no `verilog.v` file, use the Verilog names in the `verilog.map` file.
3. If there is no `verilog.v` file and there is a PORT_ORDER property in the `verilog.map` file, use the PORT_ORDER property.
4. If there is no `verilog.v` file and no PORT_ORDER property in the `verilog.map` file, use the order from the `chips.prt` file.

POWER_GROUP

The POWER_GROUP property, attached to a pin, modifies the POWER_PINS property at any given level. Multiple occurrences of the POWER_GROUP property are allowed in the `chips.prt` file and the schematic instance. Only a single occurrence is allowed in the PPT. At each level of physical part processing (`chips.prt`, `ptf`, and schematic instance), the POWER_PINS property is derived and the POWER_GROUP property is applied to the resulting pins specified by the POWER_PINS property.

You cannot use the POWER_GROUP property in the Occurrence Edit mode.

POWER_MAX

The POWER_MAX property, attached to a component is an optional Intermediate Data Format record that specifies maximum power rating in milliwatts.

POWER_OPR

The POWER_OPR property, attached to a component, is an optional Intermediate Data Format record that specifies operating power rating in milliwatts.

POWER_PINS

The POWER_PIN property, attached to a pin, specifies power and ground pins that exist on the physical part but are not shown on the schematic symbol. Typically, the POWER_PINS

property is specified in the `chips` file, but you can also specify it in the `ptf` file or in the schematic.

PROBE_NUMBER

The PROBE_NUMBER property, attached to a net, is an arbitrary character string that can represent the testpoint reference designator. If you checked the *Text: Display* field in the Testprep Parameters dialog box, this string is placed on the manufacturing probe layer when a probe is created. The PROBE_NUMBER property overrides the *Text Display* options: *net-Alphabetic, net-Numeric*, and *stringNumeric* only for the nets that have the property attached. If multiple test points are created for a net, the string becomes a prefix for the additional testpoints on the net. For example, if you specify `XYZ` for PROBE_NUMBER on netname clock1, the testpoints for the nets are designated `XYZ`, `XYZ-1`, `XYZ-2`, and so on.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net_properties` command and the *Allegro Constraint Manager User Guide* for additional information.

PROPAGATION_DELAY

The PROPAGATION_DELAY property, attached to a net or ECset, defines the minimum and maximum propagation delay constraint between any pair of pins or Rat Ts in a net or an extended net (Xnet). The value is a string and the format is:

`<pin_pairs>:<min>:<max>[<pin_pairs>:<min>:<max>]...`

where each `<pinpairs>:<min>:<max>` defines a minimum and maximum propagation delay constraint between one or more pin or Rat T pairs in an extended net. The format for `<pin_pairs>` is:

`<pin_pair>`	Defines a generic pin or Rat T pair
`AD:AR`	Specifies every driver and receiver pin-pair in the extended net
`D:R`	Specifies the longest and shortest driver and receiver pair in the extended net
`L:S`	Specifies the longest and shortest pin-pairs in the extended net (previously defined as "`::`")

The `<min>` and `<max>` fields define the minimum and maximum propagation constraints applied to the selected pin-pairs—except when you specify the D:R and L:S pin-pair formats. When the latter is the case, the `<min>` value is applied only to the shortest pin-pair and the

<*max*> value is applied only to the longest pin-pair. Both the <*min*> and <*max*> values are formatted as <*value_with_units*> where the legal value types are:

PROP_DELAY	Specifies that the constraint is a delay value
PERCENTAGE	Specifies that the constraint is a percentage of the manhattan distance between the pins
DB_DIMENSION	Specifies that the value is a length measurement

If the <*min*> and <*max*> values have no units, it is assumed that the value is a *DB_DIMENSION* and the units of the drawing are design units. Either the <*min*> or <*max*> field can be omitted as long as the colon separators are specified.

Sample Syntax for Adding Percentage To Propagation Delay

This example defines minimum and maximum propagation delay for a user-defined pin-pair: U18 pin D3 to U21 pin Y26. The minimum and maximum values are both set to 75%.

```
U18.D3:U21.Y26:75 %:75 %
```

An example of the longest and shortest pin-pairs in the extended net using percentage is:

```
L:S:80%:80%
```

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

PROPAGATION_DELAY_ACTUAL

The PROPAGATION_DELAY_ACTUAL property, attached to a net or ECset, returns the calculated propagation delay.

PULSE_PARAM

The PULSE_PARAM property, attached to a net, Xnet, bus, or differential pair, is used by the *Net – Timing – Setup/Hold* tab of the Constraint Manager spreadsheet to store the description of a clock pulse. The value of this property is a string of the following format:

<*frequency*>:<*duty_cycle*>:<*jitters*>:<*cycle_to_measure*>

If no frequency units are specified, then hertz (Hz) is assumed. If no jitter units are specified, then picoseconds (ps) are assumed.

RATED_POWER

The RATED_POWER property, attached to component definitions, specifies nominal power dissipation. The value of this property is a string.

RATS_FACTOR

This property is scheduled for implementation in a future release.

RATSNEST_ SCHEDULE

The RATSNEST_SCHEDULE property, attached to a net or ECset, specifies the type of ratsnest calculation that Constraint Manager performs on this net.The possible values are described below:

- POWER_AND_GROUND indicates that the net does not display a ratsnest. Unconnected power or ground pins in the net will be displayed by a boxed X figure rather than a rat to the unconnected pins. Displays can also appear at the end of dangling connect lines off each unconnected power/ground pin if ratsnest points are set to the closest endpoint.

  Note: Unlike other ratsnest schedules, you cannot set POWER_AND_GROUND in an electrical constraint set (ECset).

- MIN_TREE indicates that the net rat should be displayed with the minimum spanning tree algorithm. This can form Ts at pins.

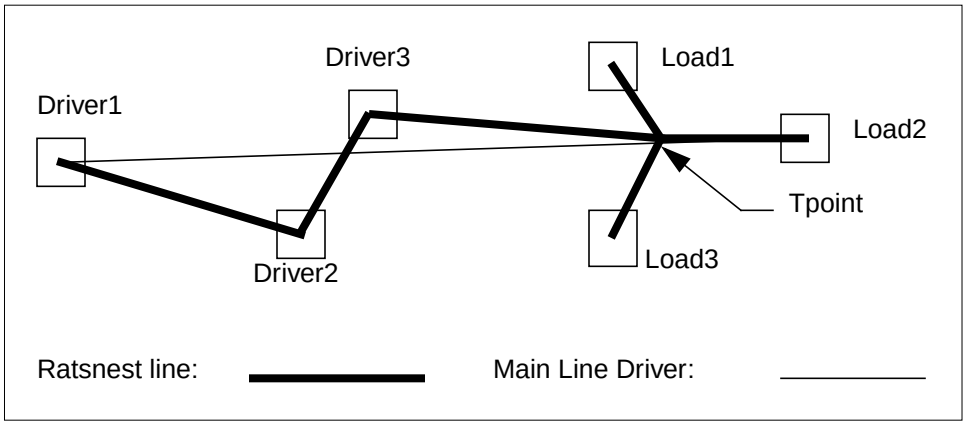
- MIN_DAISY_CHAIN indicates that a minimum length daisy-chain schedule is formed.

- SOURCE_LOAD_DAISY_CHAIN indicates that a source-to-load ECL daisy-chain schedule is used.

- FAR_END_CLUSTER automatically places a single Tpoint in a schedule at a calculated location.

  The Tpoint is automatically placed when the last component on the net is placed. No ratsnest is visible until then. All pins with a PINUSE = INPUT are scheduled from the Tpoint. If there are multiple drivers present, they are classified as main line and other drivers. The main line driver is the longest driver/receiver pair on the net. All other drivers are daisy- chained together. If there is a terminator pin on the net, it is connected to the Tpoint.

  If the Tpoint is not referenced in a PROPAGATION_DELAY value, it is placed along the main line approximately 90% from the main line driver to the furthest receiver pin.



If there is a terminator pin on the net, but the net has no driver pin present, the net is terminated. In this case, the terminator pin is scheduled first, as the main line driver.

- STAR specifies a ratsnest similar to FAR_END_CLUSTER without the Tpoint added.

  All the driver pins are daisy-chained together and all the receiver pins are connected to one end of the daisy chain.

The following is the designation of pins with the pinuse for the STAR and FAR_END_CLUSTER algorithms:

- Power, ground, loadout, tri, OCA, OCL are all treated as DRIVERs.
- Loadin pins are always LOADs.
- Pins with BI are always DRIVERs, unless there are no LOADs and no unspecified pins.
- Unspecified pins are always LOADs, unless there are no DRIVERs and no BIs.
- If there are no DRIVERs and no LOADs, all pins are daisy-chained together.

Be aware of the following special conditions related to the RATSNEST_SCHEDULE property:

- If there is no RATSNEST_SCHEDULE property specified for a net, then a MIN_TREE schedule is assumed.
- If you explicitly select a MIN_TREE schedule or default to it, and you set the TS_ALLOWED constraint to TS_NOT_ALLOWED, then a MIN_DAISY_CHAIN schedule is generated instead.
- If you explicitly select a MIN_TREE schedule or default to it, and the net contains an ECL property, then the SOURCE_LOAD_DAISY_CHAIN schedule is generated instead.

- The NO_RAT net property still operates normally and independently of the RATSNEST_SCHEDULE constraint. If you specify a NO_RAT property, then Allegro PCB Router interconnects using a `MIN_TREE` algorithm.
- If you defined the net's schedule, then Allegro PCB Editor ignores the value of the RATSNEST_SCHEDULE constraint.
- If you select `POWER_AND_GROUND`, you cannot add a BUS name to the net.

  Note: `POWER_AND_GROUND` scheduling is not supported in pre-14.2 versions. You must delete this schedule value before opening your design database in earlier versions.

REF_DES_FOR_ASSIGN

The REF_DES_FOR_ASSIGN property is automatically attached to a function (gate) by Allegro PCB Editor when loading a preassigned netlist using the `netin` command. It is used by Gate Assignment as the first choice when choosing a component for assigning the function. It is reserved even after gate assignment or swapping or both, and can then be compared to the reference designator of the component that contains the function.

REF_DES_PATTERN

The REF_DES_PATTERN property, attached to a component, lets you specify a pattern to use when assigning a reference designator to a particular part type or instance. You can specify the REF_DES_PATTERN property in the `chips` file, the `ptf` file, and the schematic.

REGION_NAME

The REGION_NAME property, attached to a rectangle or shape on the REGION_CLASS layer, lets you specify a name to be assigned to a constraint region.

RELATIVE_PROPAGATION_DELAY

Important

Cadence recommends that you use Constraint Manager to apply this property. See Working with Objects in the *Allegro Constraint Manager User Guide* for more information.

The RELATIVE_PROPAGATION_DELAY property replaces the MATCHED_DELAY property in version 14.0. It is an electrical constraint attached to pin-pairs on a net. It specifies a group

of pin-pairs that are required to have interconnect propagation delays matching a specified delta (offset) and tolerance. A RELATIVE_PROPAGATION_DELAY group has one or more reference pin-pairs against which all other pin-pairs in the group are compared.

The format of a relative propagation delay as a property is:

```
<gp>:<scope>:<p1>:<p2>:<delta>:<tol>[:<gp>:<scope>:<<p1>:<p2>:<delta>:<tol>]...
```

where:

gp	Defines the name of the matched group
scope	Defines whether the constraint is unique to a net or extended net, global to the entire design, or a bus. Legal values are: L for local to the net/extended net. G for global to the design. This value has meaning only for constraint values that come from a constraint set. When "flattened" to a specific net or extended net, the group name is modified to make it unique for the net if the scope value is L. If the value is G, the group name is assigned as is. If the scope value is omitted, L is assumed. B for a bus. You can only specify this value in an electrical constraint set. For more information about the B scope, see Match Groups in the Constraint Manager User Guide.
<p1>/<p2>	Defines the two pins and/or Rat Ts in the pin-pair. These can be specific pins AD:AR, D:R, or L:S

The pin1 and pin2 syntax supports the following variations for defining pin-pairs:

pin1:pin2	Adds the specified pin-pair to the matched group. Specify the reference designator and pin number for both pins of the pin-pair. Either pin's reference designator and pin number can reference a T point, which is specified as T.*<number>*. Both pins or T points must be a part of the net to which the property is attached.
D:R	Adds the longest driver/receiver pin-pair combination in the net to the matched group. If no driver/receiver pin-pairs are found, treat the property value as if the group ::: tolerance format was specified.

AD:AR	Adds every driver/receiver pin-pair combination in the net to the matched group. If no driver/receiver pin-pairs are found, treat the property as if the group ::: tolerance format was specified.
L:S	Adds the longest pin-pair in the net to the matched group. All terminator pins in the net are ignored in the search for the longest pin-pair.
delta	Specifies the difference or offset between each pin-pair member and the target pin-pair. If the delta is unspecified, all members need to match. If the delta is other than null (including zero), the group is a *relative* match group.
tol	The skew allowed when matching members in the group to each other or to the delta value. Tolerance is the only required attribute. If Tolerance is specified (and *Delta* and *Target* are not specified), the group is a *relative* match group.

One pin-pair is always selected as the target pin-pair, following these rules:

- If only one pin-pair has no delta value, it is selected.
- If all of the pin-pairs have delta values, the pin-pair with the smallest delta is selected.
- If there is more than one pin-pair with the smallest delta value, the one with the longest pin-pair length is selected.
- If there is more than one pin-pair without a delta value, the one with the longest pin-pair length is selected.

The CHECK_MIN_DELAYS environment variable associated with the obsolete MAX_DELAY constraint does not apply to relative delays. Instead, you can use the environment variable CHECK_UNROUTED_RELATIVE_PROP_DELAYS when you want the DRC checker to check unrouted pin-pairs based on the manhattan distance of their ratsnest connections.

Following is an example that illustrates the use of the RELATIVE_PROPAGATION_DELAY property:

```
M1:G:L:S:250mil:10mil
M1:G:L:S::
```

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

REMOVE

This body property is used either on bodies in the library or on instances of bodies in the schematic design.

Syntax

```
REMOVE LINK | EXCLUDE | AUTO | FALSE
```

The syntax is explained below:

LINK

The REMOVE=LINK property removes the component and connects the pins of the component. For example, use REMOVE=LINK on resistors.

If you have attached the property REMOVE=LINK on the resistor, the resistor is replaced with a wire of same connectivity in the simulation netlist. In the above example, signal ABC is alias to signal RESET.

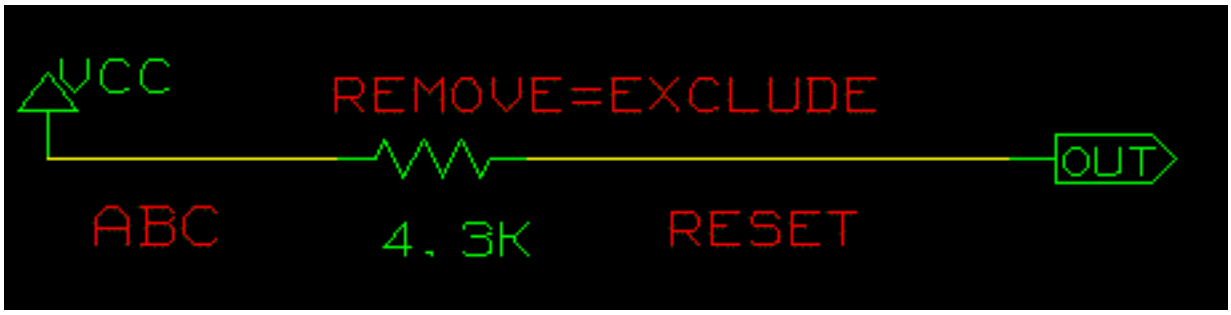
Note: The resistor will be visible in the packaging netlist.

The simulation netlist generated by Design Entry HDL is given below:

```
module remove (reset );
  output  reset;
  // global signal glbl.vcc;
  wire  abc;
    alias_bit alias_inst1 (abc, glbl.vcc);
    alias_bit alias_inst2 (abc, reset);
endmodule
```

EXCLUDE

The REMOVE=EXCLUDE property removes the component and all nodes on the component. For example, use REMOVE=EXCLUDE on capacitors and non-series terminating resistors.



Once you have attached the property REMOVE=EXCLUDE on the resistor, the resistor is replaced by an open in the simulation netlist.

Note: The resistor is visible in the packaging netlist.

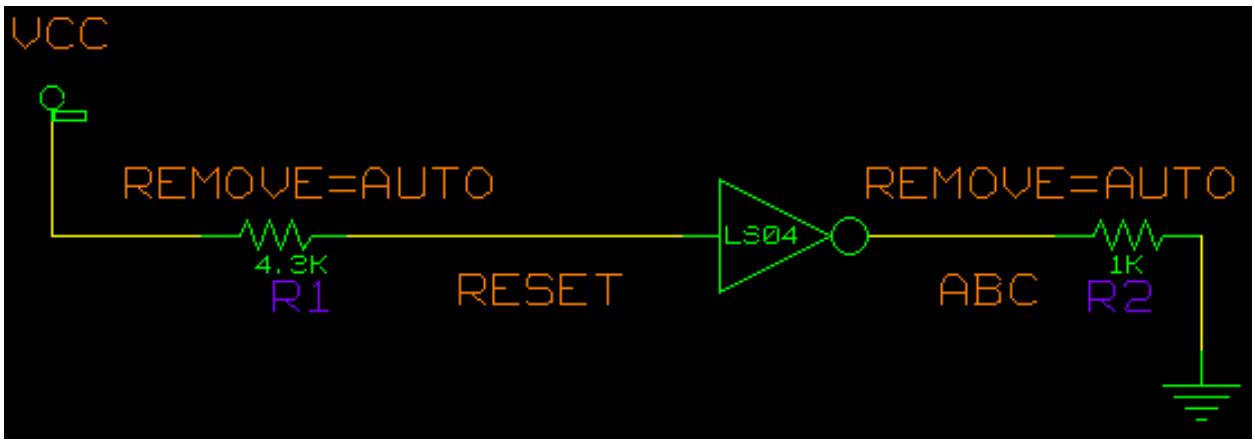
The netlist generated by Design Entry HDL is given below:

```
module remove (reset );
  output  reset;
  // global signal glbl.vcc;
  wire  abc;
  alias_bit alias_inst1 (abc, glbl.vcc);
endmodule
```

AUTO

The REMOVE=AUTO property removes the component and connects the pins of the component if the component is connected between two internal nodes. If one pin of the component is connected to a power supply, for example VCC, Design Entry HDL replaces the component with a pull-up in the simulation netlist. If one pin of the component is connected

to a ground, for example GND, Design Entry HDL replaces the component with a pull-down in the Verilog netlist. For example, use REMOVE=AUTO on resistors.



In this example, the property REMOVE=AUTO is attached to resistors R1 and R2. Resistor R1 is connected to the power symbol VCC, resulting in a pull-up in the netlist. Resistor R2 is connected to the ground symbol GND, resulting in a pull-down in the netlist.

The netlist generated by Design Entry HDL is shown below:

```
pulldown (abc);
pullup (reset);
```

To use the REMOVE=AUTO property on a component

1. Specify the name of the ground symbol (for example, `GND` and `GNDD`) in the *Supply 0* field in the *Verilog Netlist* dialog box.

   To access the *Verilog Netlist* dialog box, do the following:

   a. In Design Entry HDL, choose *Tools – Options*.

      The *Design Entry Options* dialog box appears.

   b. Select the *Output* tab.

      Ensure that the *Create Netlist* check box is selected.

   c. Click the *Options* button against the *Verilog* check box.

      The *Verilog Netlist* dialog box appears.

2. Specify the name of the power symbol (for example, `VCC` and `VDC`) in the *Supply 1* field in the *Verilog Netlist* dialog box.

3. Attach the REMOVE=AUTO property to the component.
4. Connect a wire to this component.
5. Add a power symbol, such as VCC or GND, that is added to the SUPPLY1 or SUPPLY0 list, to one of the terminals of the component. This results in a pull-up or pull-down net.

Note: If you use the REMOVE=AUTO property on a component that has only one pin in each section and

- If one of the terminals is connected to a power symbol that is declared as SUPPLY1 or SUPPLY0, then Design Entry HDL changes it to a pull-up or pull-down.
- If a power symbol is not declared as SUPPLY1 or SUPPLY0, Design Entry HDL treats the REMOVE=AUTO property as the REMOVE=LINK property.

FALSE

If the REMOVE property is defined on the symbol of a component, the property will be available on every instance of the component you place in your design. If you do not want to use the REMOVE property on a specific instance of the component in your design, specify the REMOVE=FALSE property on that instance. Design Entry HDL will ignore the REMOVE property on an instance if the value of the property is FALSE.

REUSE_ALT_MODULE

The REUSE_ALT_MODULE property, attached to a component, uses multiple physical modules for the same logical module. For example, assume that you have a logical module named `base_level` and you have three physical modules `base_level1`, `base_level2`, and `base_level3` corresponding to the logical module. By selecting `REUSE_ALT_MODULE = base_level3`, you can put the `base_level3` module on the board.

Note: The REUSE_ALT_MODULE property is not supported by the Allegro PCB Editor.

REUSE_ID

The REUSE_ID property, automatically attached to a component, allows for the correct assignment of logic within a module instance. The REUSE_ID property is a number that uniquely identifies each symbol and component within a module. It is used to reconnect reference designators when a module is used in a design, thus allowing for the correct assignment of logic within a module instance. The property is stored within the module file and the Allegro PCB Editor design file that represents the module information, and is backannotated to the schematic that represents the module's logic.

Note: Because this property is for Cadence internal use only, you must not edit it.

REUSE_INSTANCE

The REUSE_INSTANCE property, assigned on the reuse block when instantiating it in Allegro Design Entry HDL, uniquely identifies the instance of the reuse block. Allegro PCB Editor uses the REUSE_INSTANCE property to differentiate among multiple instances of a reuse module. Unlike other schematic properties, the REUSE_INSTANCE property, defined on the topmost block, wins in the case of nested blocks.

REUSE_MODULE

The REUSE_MODULE property, attached to a component, represents the name of the MDD file that the Allegro PCB Editor loads when it finds this property on the components in a reuse module.

REUSE_NAME

The REUSE_NAME property, attached to a component, represents a module definition name and is stored within the module file and the design in which the module resides. It identifies which definition should be used for each module instance. This property is backannotated to the schematic when logic is exported to the front-end schematic capture tool. The value is a string.

REUSE_PID

This property is for Cadence internal use only.

RFELEMENTTYPE

The RFELEMENTTYPE property, defined in Allegro PCB Design XL and attached to a component, identifies the types of different RF components. This property is an integer ranging from 0 to 299. Each integer in the range identifies a particular RF component type based on its physical or electrical characteristics.

RFPCB_OBJECT

The RFPCB_OBJECT is a system property that you attach to shapes and lines (not clines) in Allegro PCB Design XL, to identify an RFPCB object as one on which you can use the RFPCB Flexible Shape Editor for RFPCB processing. Its value is a BOOLEAN TRUE.

RF_TLINE

The RF_TLINE property, attached to tline components that are transferred as Pcells from a front end schematic, identifies the corresponding tline component cell name in the schematic.

ROOM

The ROOM property, attached to a reference designator (component) or function designator, indicates that the function is to be assigned to a component containing other functions in the same room. The value is a string.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

ROOM_TYPE

The ROOM_TYPE property is attached to a room boundary or to the root of the design. The values that can be assigned to this property are:

HARD:	Allows components belonging to this room to be placed entirely within its room boundary. DRC errors occur when you place a component outside this room. Any components that are not members of this room, yet are placed entirely within the room boundary, cause DRC errors.
SOFT:	Generates no DRC errors for any components placed in this room. When present on the design root, then rooms without the ROOM_TYPE property inherit the root value.
INCLUSIVE:	Results in behavior similar to that produced by the *HARD* value, but DRC errors occur when components belonging to this room straddle the room boundary.
HARD_STRADDLE:	Results in behavior similar to that produced by the *HARD* value, but allows components belonging to this room to straddle the

room boundary without generating DRC errors. DRC errors occur when non-member components are placed completely inside the room boundary.

INCLUSIVE_STRADDLE: Results in behavior similar to that produced by the *HARD* value, but allows all components to be placed entirely in the room or to straddle the boundary without generating DRC errors. DRC errors occur when non-member components belonging to this room are placed entirely outside the room.

ROTATE

The ROTATE property, attached to a component, indicates the rotation of an instance of the component. You can assign this property by choosing the *Edit – Rotate* menu command in Allegro Design Entry HDL.

ROUTE_PRIORITY

The ROUTE_PRIORITY property, attached to a net, assigns a routing priority. Allegro PCB Router routes the net according to the priority order. Nets with the lowest number have the highest priority. To route certain nets first, tag those critical nets with a ROUTE_PRIORITY property value of 1.

Allegro PCB Router considers the highest number to have the highest priority, with the Allegro PCB Router translator adjusting accordingly.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net properties` command and the *Allegro Constraint Manager User Guide* for additional information.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

ROUTE_TO_SHAPE

This property is obsolete.

ROUTES_ALLOWED

The ROUTES_ALLOWED boolean property, attached to a shape or rectangle of the Route Keepout class, permits clines to route through the keepout area.

When you choose *Setup – Areas – Shape Keepout* (keepout shape command), the ROUTES_ALLOWED, as well as the VIAS_ALLOWED properties are added to the newly created shape. Adding the VIAS_ALLOWED and ROUTES_ALLOWED properties to a route keepout creates a shape keepout, meaning vias are allowed to drill through the keepout, and routing is permitted through it.

SAME_NET

The SAME_NET property, attached to a net, specifies whether Allegro PCB Editor checks the elements on the same net for spacing violations. This property overrides the default setting of O*ff* in the *Same Net DRC* field in the Default Values dialog box for the specified net only. Be sure that you set the value to TRUE. The SAME_NET property has the same membership objects as the Physical and Spacing properties.

SAME_NET_XTALK_ENABLED

The SAME_NET_XTALK_ENABLED property, attached to a net, specifies that crosstalk checks occur between a net and itself. Cadence recommends that you use the Electrical Constraints dialog box to set-up crosstalk checks.

SCHEMATIC_NAME

The SCHEMATIC_NAME property, attached automatically by Allegro PCB Editor to a board (design), represents the name of the schematic from which the layout logic (netlist) was derived. Cadence recommends that you do not use this property.

SDFDELAYTYPE

The SDFDELAYTYPE property, attached to a symbol, defines the scale type. The values that can be assigned to this property are MINIMUM, TYPICAL, and MAXIMUM. The default value assigned to the property is TYPICAL.

SDFFILE

The SDFFILE property, specified on the instances of a component in the schematic, is the name of the SDF file. For example,`./data/design.sdf`.

When you add this property to a component, the netlister inserts the *$sdf_annotate* directive in the netlist. After specifying the SDFFILE property, if you do not specify other special properties such as SDFDELAYTYPE, SDFSCALEFACTOR, and SDFSCALETYPE, the default values of these properties are added in the netlist.

SDFSCALEFACTOR

The SDFSCALEFACTOR property, attached to a symbol, defines the scale factor. The scale factor can be `<real|integer>:<real|integer>:<real|integer>`.

The default value is `1:1:1`.

SDFSCALETYPE

The SDFSCALETYPE property, attached to a symbol, defines the scale type. The possible scale types are `FROM_MINIMUM, FROM_TYPICAL, FROM_MAXIMUM,` and `FROM_MTM.`

The default value of the SDFSCALETYPE property is `FROM_MTM`.

SEC

The SEC property, attached to a component, assigns the logical component to a particular section within a physical part. Use the *Component – Section* menu command in Allegro Design Entry HDL to assign this property.

SEC_TYPE

The SEC_TYPE property, attached to a component, identifies the package type in the `chips.prt` file used to get pin number assignments when sectioning a part. Use the *Component – Section* menu command in Allegro Design Entry HDL to assign this property.

SHAPE_OVERSIZE

The SHAPE_OVERSIZE property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies how far away the copper should be kept for shapes and rectangles. This value is added to the DRC clearance for clines and lines being voided. By default, there is no oversizing of voids. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Clearances* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SHIELD_NET

The SHIELD _NET property, attached to a net, allows you to shield a net. It causes a “shield” statement to be added to the “net” statement. The value is the name of the net to use for shielding the specified net, for example GND shields DATA1 net. This property allows you to control Allegro PCB Router.

Allegro PCB Router allows for three different types of shielding `parallel` (on the same layer as the routed net), `tandem` (above and below the net on adjacent layers) and `coax` (both parallel and tandem shielding together). The default is `parallel`.

If you attach the SHIELD_NET property to a net, you must also attach the SHIELD_TYPE property. If you use parallel shielding, the default, Allegro PCB Router handles the addition of the SHIELD_TYPE property.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net_properties` command and the *Allegro Constraint Manager User Guide* for additional information.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

SHIELD_TYPE

The SHIELD_TYPE property, added to a net, indicates the type of shielding. The value of this property is used in a “type” statement within the “shield” statement. The value should be one of Allegro PCB Router keywords `parallel`, `tandem`, or `coax`. For example in the `.dsn` file:

```
(net sig1 ..... (shield on (type parallel) (use_net GND)))
```

where GND was from the SHIELD_NET property and parallel was from the SHIELD_TYPE property. This allows you to control Allegro PCB Router.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the net_properties command and the *Allegro Constraint Manager User Guide* for additional information.

SHORTING_SCHEME

The SHORTING_SCHEME property is automatically attached by APD or Allegro PCB Editor to selected pins or vias in the nets or subnets that connect to power or ground planes when you use the
Route – Define Short menu command.

SIGNAL_MODEL

The SIGNAL_MODEL property, attached to a reference designator (component), is the name of the Packaged Device Model, which defines the electrical, I/O models, and package parasitics used by SigNoise to characterize devices for simulations. For this property to be used in Signal Integrity analysis, the string value entered in Allegro PCB Editor must match the exact name of a model in a library, and it is case-sensitive. If not, the user can create a dummy model for simulation. By assigning the signal model with *Analyze – SI/EMI Sim – Model,* you can choose a model using the library browser.

SIG_NAME

The SIG_NAME property is attached to a signal. Allegro Design Entry HDL interprets all signals with the same name as being the same signal and uses this property to connect signals across multipage drawings.

SIG_NAME values must be legal HDL signal names. These name restrictions apply:

- Names must start with a letter.
- Names must use only letters, numbers, and underscores. You cannot include spaces or other characters in names.
- Names cannot be VHDL and Verilog keywords.
- Allegro Design Entry HDL is not case sensitive; Allegro Design Entry HDL treats two names that differ only in uppercase or lowercase letters as the same name.

For more information on Allegro Design Entry HDL naming conventions, see the *Allegro Design Entry HDL Reference Guide* (<*your_instal_dir*>/doc/conref).

SIM_BIND_VIEW

The SIM_BIND_VIEW property, attached to a symbol or component, specifies the binding for the Verilog models that appear in the mapped netlist. If you use wrappers for simulation, specify a wrapper view name as the value for this property. If you use map files for simulation, use the property value as a binding for the mapped Verilog model.

Example

If you use a `lsoo` part in the schematic and specify:

```
SIM_BIND_VIEW=vlog_model
```

the `lsoo` part is bound to the `vlog_model` view of `lsoo`. However, if you specify:

```
VERILOG_MODEL=TTLOO
```

and

```
SIM_BIND_VIEW=vlog_model
```

the `lsoo` part is bound to the `vlog_model` view of `TTL00`.

If you use parts having shared pins (physically different pins with the same pin name on different symbols), you must specify the wrapper's view name also as the property value for the SIM_BIND_VIEW property. Allegro Design Entry HDL finds the location of the wrapper from this view.

You must place the SIM_BIND_VIEW property on any one of the split parts if there are common or shared pins across different split parts. If Allegro Design Entry HDL finds the same property SIM_BIND_VIEW or the same parameter on more than one instance of the same SPLIT_INST_NAME group, it takes the property from the instance having lower path values and ignores the property from the other instances in the same split instance group. For more information, see SPLIT_INST_NAME.

SIM_MAP_VIEW

The SIM_MAP_VIEW instance property, attached to a symbol or component, overrides the default map file viewlist specified in the simulation configuration. For example, if the default map file viewlist for the `cfg_verilog` simulation configuration is `swift_map`, you can specify `SIM_MAP_VIEW=vlog_map` to override the map view binding.

SIZE

The SIZE property, attached to a component, specifies the width of pin names and signal names, and defines size expansion. Using the SIZE property can greatly minimize the number of parts and interconnections.

The SIZE property, when used within the hierarchy, generates specific versions of a module. Most of the primitives in Design Entry HDL use the SIZE property to specify the number of bits of the primitive. The value of the SIZE property can be fixed or variable (SIZE=SIZE). In this case, the size of the primitive is taken from the SIZE property attached to the component being modeled.

For simple primitives, the SIZE property is interpreted as multiple instances of the primitive. The bits of a bus connected to a sized pin of a primitive are split between the replicated instances of the primitive. For example, if bus A<*1..0*> is connected to a sized pin of a BUF primitive with SIZE=2, the bit A<0> is connected to the first instance and the bit A<1> is connected to the second instance. Simple SIZED primitives are translated by Allegro Design Entry HDL as multiple instances of the primitive.

For more complex primitives, the SIZE property cannot be directly interpreted as a replication of the primitive. In an example of an ADDER primitive with SIZE greater than two, the CarryOut of one stage needs to be connected to the CarryIn of the next stage. To accurately model the SIZE property on this type of primitive, the primitive must be described as a parameterized behavioral Verilog model.

- When the property NO_REP_PRIM is found in the library description for a primitive, Design Entry HDL does not replicate this primitive or pass the SIZE parameter with the correct value to the behavioral model.
- When the property NO_REP_PRIM=TRUE is not found in the library description for a primitive, Design Entry HDL assumes it to be a simple gate and automatically replicates this gate with the actual value of the SIZE property.

SLOTNAME

The SLOTNAME property is attached automatically to a function by Allegro PCB Editor when loading a preassigned netlist with the `netin` command. During gate assignment, this slot name is the first choice when choosing the slot in a component. This slot name is reserved even after gate assignment or swapping, or both, and can then be compared to the slot that contains the function.

SMD_BEST_FIT

The SMD_BEST_FIT property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, overrides the thermal connect style for the smd pin if the chosen style does not provide sufficient thermal connects within the specified minimum and maximum number of thermals in 15-degree increments. The property values are TRUE *(Use best contact*) and FALSE (*Do not use best contact)*. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects Tab* in the Shape Instance Parameters Dialog Box in the*Allegro PCB and Package Physical Layout Command Reference.*

SMD_CLEAR_TYPE

The SMD_CLEAR_TYPE property, attached to dynamic shapes, specifies where Allegro PCB Editor gets the clearance value for how far away the copper should be kept from the thru pins: `Thermal/anti, DRC,` or `No Void`.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Clearances Tab* in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SMD_MAX_THERMS

The SMD_MAX_THERMS property, attached to dynamic shapes, specifies how many connect lines are created for thermal relief for smd pins. Values are `1-8.` Up to four are allowed on orthogonal and diagonal clines. Up to eight are allowed on the 8-way option.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SMD_MIN_THERMS

The SMD_MIN_THERMS property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the minimum number of connections for smd pins. Values are 1-8. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SMD_OVERSIZE

The SMD_OVERSIZE property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the oversize value for how far away the copper should be kept from the smd pins. This value is added to the clearance value of *DRC* or *Thermal/Anti*. By default, the oversize value is 0.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

SMD_THERM_CONN

The SMD_THERM_CONN property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies how smd pins with the same net name as the shape should be connected to the shape: o*rthogonal, diagnonal, 8-way, full contact,* and *none*.This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SMOOTH_MIN_GAP

The SMOOTH_MIN_GAP property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the minimum gap width used for the corner

radius (round) or length (chamfered). For vector artwork, this doubles as the minimum aperture for artwork fill.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Void Controls* Tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SMOOTH_TRIM_CONTROL

The SMOOTH_TRIM_CONTROL property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies a solid outline corner style (round or chamfered) for solid shapes for raster artwork formats. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Void Controls* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SNAP_VOID_XHATCH

The SNAP_VOID_XHATCH property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, is for crosshatched shapes only. It specifies how the clearance areas for voids are created and snaps the created voids to the hatch grid. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Void Controls* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

SOLDER_BALL_HEIGHT

The SOLDER_BALL_HEIGHT property, attached to a symbol, specifies the height of the solder ball of a package. The value is in the current design unit. This property is required for the PNC interface.

SOV_CHECK

The SOV_CHECK property, attached to one or more nets, means that the tool checks only those nets when you run the `highlight sov` command. Normally when you run the command, the tool checks all nets except those with the VOLTAGE property attached. If you assign a net with both the SOV_CHECK and VOLTAGE properties, the VOLTAGE property overrides and the tool does not check the net.

SPIF_CONSTANTS

The SPIF_CONSTANTS property is for Cadence internal use only. The value is taken as a colon-(`:`) delimited list. Each value is placed in the `.dsn` file as a “constant” statement. It is not used in Allegro PCB Router, but is used in the Mentor-to-Allegro SI interface.

SPIF_TURRET

The SPIF_TURRET property is for Cadence internal use only. The value is added in a “turret” statement within the “path” statement. It is not used in Allegro PCB Router, but is used in the Mentor-to-Allegro SI interface.

SPLIT_INST

The `SPLIT_INST= TRUE` property is attached to the symbol of all split parts of a device by the librarian developing the split parts. If this property is not attached to the symbols, you cannot attach the property to the split parts you use in your schematic.

In the simulation netlist, all parts with the `SPLIT_INST =TRUE` property are merged into a single instance. In cases where the schematic has multiple instances of a split part, then along with the `SPLIT_INST=TRUE` property, you should also have the `$LOCATION` property specified on all split parts. A group of split parts of the same device is known as split instance group. Parts having same value for the `$LOCATION` property form one split instance group.

For example, if there is a large pin count device, ASYM_PART, which is split into four parts, all the four split parts must have the SPLIT_INST property set to `TRUE`, attached to them. Now

if a schematic has only one instance of all four ASYM_PART split parts, the Allegro Design Entry HDL netlister merges all devices into a single instance in the simulation netlist. The instance name in the generated netlist is ASYM_PART_SPLIT_1. But in case there are multiple instances of a ASYM_PART split part, the Allegro Design Entry HDL netlist generates a warning and uses internal logic to group the parts into different split inst groups.

To remove this warning, attach the $LOCATION property with the same value on all split parts that should form a single split inst group. When you save a schematic in Allegro Design Entry HDL, all split parts with the same value for the $LOCATION property are merged into a single instance in the netlist.

SPLIT_INST_NAME

Attach the $SPLIT_INST_NAME property on all split parts that have to be merged into a single instance in the simulation netlist. By default, the value assigned to the $SPLIT_INST_NAME property is ?. If the default value is used, in the simulation netlist all split parts are merged into a single instance named <*device_name*>_split_1.

For example, consider a device, ASYM_PART, which is split into four parts. All four parts have the $SPLIT_INST_NAME property set to ?. Allegro Design Entry HDL netlister reads the property and merges the four split parts to generate one instance of the part with the instance name as asym_part_split_1.

If you need multiple instances of a split part, specify the property $SPLIT_INST_NAME with different values for each instance. The property value becomes the instance name in the simulation netlist and therefore, must be the same for all split parts in a split inst group.

For example,

- Attach the SPLIT_INST_NAME =INST_FIRST property to all the body drawings of the split part for the first instance.
- Attach the SPLIT_INST_NAME= INST_SECOND property for the second instance.

The output in the netlist is:

```
SPLIT_PART INST_FIRST(.....);
SPLIT_PART INST_SECOND(.....);
```

Do not use the instance name in this format: i<integer>.
This can cause a naming conflict because Allegro Design Entry HDL generates instance names such as I1P, I2P, and so on.

STUB_LENGTH

The STUB_LENGTH property, attached to ECL nets, overrides the *Maximum Stub Length* electrical constraint, which is the maximum length allowed for a stub in database units. The STUB_LENGTH property overrides the ECL property, which indicates that the net does not allow stubs. A `STUB_LENGTH=0` property means that there are no stubs.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in the your documentation set.

SUBDESIGN_MASTER

The SUBDESIGN_MASTER property, attached to a symbol or component in the schematic, identifies which instance in a top-level design is by the `GEN_SUBDESIGN` directive to generate subdesign state files. If the top-level design includes multiple instances of a subdesign, you may specify a particular instance, whose packaging is most optimal, as the source for the subdesign state file.

To specify a particular instance with optimal packaging, attach the SUBDESIGN_MASTER property to a particular instance of the subdesign. If you do not specify this property, the first instance of the subdesign to be parsed by the Compiler is used as the default subdesign master.

Note: Packager-XL does not look at the value of the SUBDESIGN_MASTER property. If the property is present, the instance is used to generate the subdesign state files.

Syntax

```
SUBDESIGN_MASTER = <value>
```

Example

```
SUBDESIGN_MASTER = TRUE
```

SUBDESIGN_SUFFIX

The SUBDESIGN_SUFFIX property, attached to a symbol in the schematic, specifies the suffix to be added to all reference designators in a subdesign module. To create unique reference designators within instances of subdesigns, append a suffix to the reference designator found in the subdesign master.

For example, a reference designator of U1 in a subdesign master could become U1_1 for one occurrence of that subdesign. The SUBDESIGN_SUFFIX property lets you control the assignment of suffixes to subdesign instances and facilitates the association of packages in the board layout with the schematic design.

Note: Each SUBDESIGN_SUFFIX property must be unique across all subdesign modules. This means that you cannot have SUBDESIGN_SUFFIX = 1 on more than one subdesign even when the subdesigns themselves are different. This minimizes the possibility of reference designator collisions. If there are duplicates, a warning is generated and one of the duplicates is ignored.

Syntax

SUBDESIGN_SUFFIX = <*string*>

Example

To assign a suffix 3 to instance 1P of subdesign SUB1, add the following property to 1P on the schematic:

SUBDESIGN_SUFFIX 3

This causes all reference designators in instance 1P of subdesign SUB1 to have a suffix 3 attached to them.

SUBNET_NAME

The SUBNET_NAME property, attached to one or more pins on the same net, specifies a subnet and is used by Allegro PCB Editor. You identify a subnet with a subnet name, either in the netlist or interactively using the *Edit – Properties* menu command.

For example, if pins U4.1, U10.6 and U17.20 have the SUBNET_NAME value called ABC, then those pins form the subnet ABC.

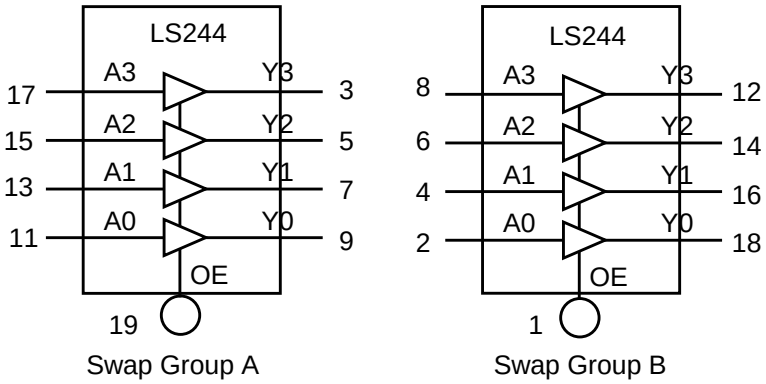
SWAP_GROUP

The SWAP_GROUP property, attached to a function designator, controls swapping for functions generated by Allegro Design Entry HDL bodies that have the HAS_FIXED_SIZE property. The HAS_FIXED_SIZE property tells the Allegro Design Entry HDL, Compiler, and Packager applications that a single schematic symbol (body) has more than one function. The value for SWAP_GROUP is a string.

For example, if you create a schematic using the bodies in Figure 1-3, any of the four gates can be swapped in either Swap Group A or Swap Group B, but swapping is not allowed between the two swap groups.

The property HAS_FIXED_SIZE = 4B means there are four functions represented by one body. The fact that multiple functions are represented by a single body needs to be passed to Allegro PCB Editor to control the interactive and automatic swapping algorithms.

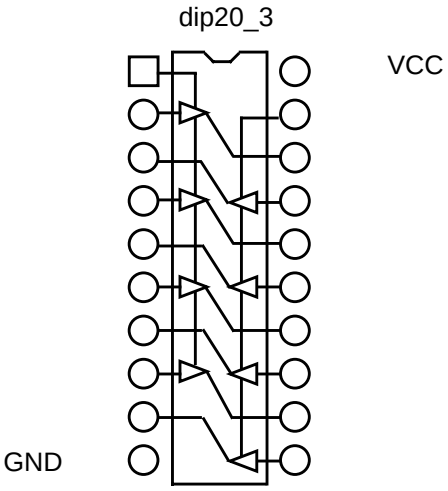
Figure 1-3 Example of Schematic Bodies



During fet2a processing, any schematic body with the HAS_FIXED_SIZE = n property is automatically assigned a SWAP_GROUP = *x* property in Allegro PCB Editor, where *x* is the logical path name to the schematic symbol in Allegro Design Entry HDL. Functions within the same swap group can be swapped with one another, but not outside the swap group.

In Allegro PCB Editor, this example translates to four functions within the SWAP_GROUP property, although this represents one-half of a LS244 Allegro PCB Editor body (see example below). Swapping is allowed among the four functions in Swap Group A above or between the functions in Swap Group B but no swaps are allowed between Swap Groups A and B.

This component contains Swap Group A and Swap Group B and is represented as follows:



SWAP_INFO

The SWAP_INFO property, in the `chips.prt` file, allows the swapping of pins across split symbols in the Allegro PCB Editor. A common example of pin swapping occurs for devices that have a large pin count. In the example given below, the device symbol is split across five smaller physical sections in Allegro Design Entry HDL. These symbols are then grouped into two logical sections in the `chips.prt` file using the SWAP_INFO property.

Example

```
SWAP_INFO=’(S1+S2), (S3+S4+S5)';
```

The SWAP_INFO property allows the swapping of pins with the same PIN_GROUP property across split symbols that belong to the same logical section.

For example, pins with the same PIN_GROUP property in S1 and S2 can be swapped with each other. Similarly, pins with the same PIN_GROUP property in S3, S4, S5 can be swapped with each other, where S1, S2, S3, S4, and S5 represent physical sections in Allegro Design Entry HDL.

SYS_CONFIG_NAME

The SYS_CONFIG_NAME property, attached to a board, specifies the system configuration name.

TECH

The TECH property, attached to a component, selects the correct interface element for the mixed signal (Digital and Analog circuits) simulation. Some examples are:

```
TECH= ’74LS’;
TECH= ’74HC’;
TECH= ’100K’;
```

TEMPORARY_PACKAGE_SYMBOL

The TEMPORARY_PACKAGE_SYMBOL property is an Allegro PCB SI internally generated property and is not user-accessible.

TERMINATOR_PACK

The TERMINATOR_PACK property, attached to a device (through a device file), indicates that the device contains terminator resistors. Be sure that you set the value to TRUE. This property is used by the terminator assignment program in Allegro PCB Editor to match the correct terminator with the appropriate ECL net. The value is either ON or OFF. No additional value is needed. The PACKAGEPROP definition indicates to the ECL scheduler that the device is a terminator.

The syntax of the TERMINATOR_PACK property in the device file is:

```
PACKAGEPROP TERMINATOR_PACK
```

TESTER_GUARDBAND

The TESTER_GUARDBAND property, attached to a net or a pin of a net where the net is the data net of a timing check, is used by the *Timing Setup/Hold* tab of the Constraint Manager. It can be used to define a fudge factor for the setup and hold calculation. The value is in nanoseconds.

TESTPOINT_QUANTITY

The TESTPOINT_QUANTITY property, attached to a net, specifies the number of testpoints that Automatic Testprep attempts to achieve on the net. If you generate testpoints automatically using the *Manufacture – Testprep – Automatic* menu command, Allegro PCB Editor does not exceed the number specified.

When you choose the *Manufacture – Testprep – Properties* menu command to edit the environment for Testprep, rather than adding the property with the *Edit – Properties* menu command, adding the TESTPOINT_QUANTITY property has a default value of 1.

If you generate test points using the *Manufacture – Testprep – Manual* menu command, the editor ignores this property and allows you to add as many test points as want.

Examine the Testprep report to see the specified and actual number of testpoints for the net. A net without the TESTPOINT_QUANTITY property appears with the TESTPOINT_QUANTITY property field blank.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the net_properties command and the *Allegro Constraint Manager User Guide* for additional information.

TESTPOINT_ALLOW_UNDER

The TESTPOINT_ALLOW_UNDER property, attached to a symbol, allows testpoints underneath a component instance of a symbol and overrides the *Allow Under Component* field on the Testprep Parameters dialog box, if it is enabled. When you attach this property to a symbol, only that instance of the symbol is affected. Testpoints are allowed directly on pin pads, when enabled, on the same side of the pad from which the component pin would be inserted. Normally, this is not allowed as the component pin interferes with the testpoint probe. Refreshing a symbol maintains any setting of this property on a symbol instance.

TESTPOINT_MAX_DENSITY

The TESTPOINT_MAX_DENSITY property, attached to symbols, verifies the maximum testpoint allocation beneath component instances of symbols but on the opposite side to that on which the components/symbols are placed. For example, if a 2000-pin BGA occurs on layer TOP, Allegro PCB Editor checks for a maximum testpoint allocation on layer BOTTOM, but within that component's place-bound region.

This property works in conjunction with the *Component Area Check* on the Testprep Density Check dialog box, available by choosing *Manufacture – Testprep – Density Check* (testprep density command). For example, if a 2000-pin BGA occurs on layer TOP, Allegro PCB Editor checks for a maximum testpoint allocation on layer BOTTOM, but within that component's place-bound region, or optionally, ASSEMBLY region, depending on the *Component Representation* setting on the General Parameters tab of the Testprep Parameters dialog box, available by choosing *Manufacture – Testprep – Automatic* (testprep automatic command). ASSEMBLY data is used if it is a SHAPE or RECTANGLE entity, or a single multi-segment LINE entity that forms a closed shape. ASSEMBLY data resembling a rectangle, but actually comprised of four different LINE

entities, is not used. Arcs are recognized in a SHAPE or LINE entity.

To use the *Component Area Check,* you must attach the TESTPOINT_MAX_DENSITY property to the symbols requiring it, where the associated value specifies the maximum number of testpoints allowed under the symbols.

The TESTPOINT_MAX_DENSITY property on a symbol has no impact on testpoints created when you choose *Manufacture – Testprep – Manual* (`testprep manual` command) or *Manufacture – Testprep – Automatic,* or if you move or delete vias that happen to be testpoints. Only running a *Component Area Check* flags violations. You must manually change the number of testpoints to meet the specified maximum.

For additional testprep information, see the *Preparing Manufacturing Data* user guide in your documentation set.

TEXT_OVERSIZE

The TEXT_OVERSIZE property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies how far away the copper should be kept for text. This value is added to the DRC clearance for clines and lines being voided. By default, there is no oversizing of voids. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Clearances* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

THERMAL_RELIEF

The THERMAL_RELIEF property is automatically attached to a thermal connect line by autovoid.

THETA_JB

This property, attached to component definitions and instances, is used by a third-party thermal analysis tool. The value of this property is a string and specifies junction-to-board thermal resistance. If a user property already exists using this name, it becomes a system property.

THETA_JC

This property, attached to component definitions and instances, is used by a third-party thermal analysis tool. The value of this property is a string and specifies junction-to-board thermal resistance. If a user property already exists using this name, it becomes a system property.

THICKNESS

This property is obsolete and will be removed in a future release.

THRU_BEST_FIT

The THRU_BEST_FIT property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, overrides the thermal connect style for the thru pin if the chosen style does not provide sufficient thermal connects within the specified minimum and maximum number of thermals in 15-degree increments. The property values are TRUE (*Use best contact*) and FALSE (*Do not use best contact*). This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

THRU_CLEAR_TYPE

The THRU_CLEAR_TYPE property, attached to dynamic shapes, specifies where Allegro PCB Editor gets the clearance value for how far away the copper should be kept from the thru pins. The values are: *Thermal/anti, DRC*, or *No Void*.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Clearances* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

THRU_MAX_THERMS

The THRU_MAX_THERM property, attached to dynamic shapes, specifies how many connect lines are created for thermal relief for thru pins. Values are 1-8. Up to four are allowed on orthogonal and diagonal clines. Up to eight are allowed on the 8-way option.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects Tab* in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

THRU_MIN_THERMS

The THRU_MIN_THERM property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the minimum number of connections for thru pins. Values are 1-8. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

THRU_OVERSIZE

The THRU_OVERSIZE property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the oversize value for how far away the copper should be kept from the thru pins. This value is added to the clearance value of *DRC* or *Thermal/Anti*. By default, the oversize value is 0.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

THRU_THERM_CONN

The THRU_THERM_CONN property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies how thru pins with the same net name as

the shape should be connected to the shape. The values are *orthogonal, diagnonal, 8-way, full contact*, and *none*. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

TIMING_DELAY_OVERRIDE

The TIMING_DELAY_OVERRIDE property, attached to a pin or net, overrides any *First Switch* or *Final Settle* delays that are computed for a net. This value is used by the Constraint Manager when checking setup and hold violations for a net.

TOL

TOL is used by the GenRad interface to prepare the `.ckt` file, as well as the report programs. The value is the percent of tolerance. It is also used to update the TOL label on the mechanical symbol. Used for discretes only.

GenRad is not supported in versions 14.0 and higher.

The syntax of the TOL property in the device file is:

```
PACKAGEPROP VALUE <tolerance in percent>
```

TOPOLOGY_TEMPLATE

The TOPOLOGY_TEMPLATE property, attached to a net, was used in Version 13.6 to record the name of the topology template that was assigned to the net. In Version 14.0, this information is recorded as part of an ECset but the property remains for compatibility with 13.6 drawings. This value can be upreved to 14.0 ECset assignments using the *Audit – Topology Templates* menu command in the Constraint Manager.

TOPOLOGY_TEMPLATE_REVISION

Beginning in Version 14.0, this property is no longer used. A topology template is imported by the Constraint Manager to create an ECset and then the ECset is assigned to nets or Xnet. The revision of the topology template is now stored as part of the ECset and is therefore no longer stored as a property on the net.

TOTAL_ETCH_LENGTH

The TOTAL_ETCH_LENGTH property, attached to an Xnet, net, bus or differential pair, overrides the constraint by the same name. The value of this property is a string with a format of `<min>:<max>`, where both `<min>` and `<max>` are etch length values with optional units. If you do not specify any units, then the units of the drawing are assumed. Either value is optional. If only a maximum value is specified, the leading colon is required. If only a minimum value is defined, the trailing colon is optional.

By default, this constraint is empty, that is, it contains neither a minimum nor a maximum value. The constraint is visible and active in Allegro PCB Design XL, Allegro PCB Performance L, APD, and Allegro PCB SI. Edits to the property at the net level bubble up to the Xnet level.

TS_ALLOWED

The TS_ALLOWED property, attached to a net, specifies whether you can make T connections, and if so, the location of the T connections. The values are:

- PINS_ONLY — Ts are only allowed at a pin. Note that whether Ts are legal at a pin is controlled by this net's schedule.

  Allegro PCB Router treats this property the same as PINS_VIAS_ONLY.
- PINS_VIAS_ONLY — Ts can only be created at a pin or via.
- ANYWHERE — Ts can be formed at a pin, via, or on a connect line.
- NOT_ALLOWED — Ts are not legal.

This property overrides the constraint rule for all layers. When the property is on a connect line, it overrides the constraint value for only that connect line on that subclass. The TS_ALLOWED property has the same membership objects as the Physical and Spacing properties.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

UNFIXED_PINS

The UNFIXED_PINS property, attached to a board file, symbol drawing, or symbol, allows movement of symbol pins during `move` or `spin` commands. Removal of the property does not undo instance edits.

UNKNOWN_LOADING

The UNKNOWN_LOADING property, attached to library components or in your drawings, turns off load checking When used as a component property, it applies to all pins of the component. As a pin property, it applies to the entire net to which the pin is attached. Attaching the NO_LOAD_CHECK property to a pin with the UNKNOWN_LOADING property, suppresses load checking only for that pin.

UNUSED_PADS_IGNORE

The UNUSED_PADS_IGNORE property, attached to symbols, nets, pins, or vias, flags these elements as exceptions during the suppression of unused pads, and as a result, retains them as follows:

- Net: All pads on all pins and vias on the specified net.
- Pin: All pads on the specific pin.
- Via: All pads on the specific via.
- Symbol: All pads of all pins and vias on the specific symbol.

USEn

The USEn property, attached to a symbol, provides names to be used in VHDL use clauses in the entity and architecture generated for the schematic. Use the following syntax:

```
USEn = library_name
```

where n is a unique number and `library_name` is the name of the library.

Note: When you use this property on a VHDL_DECS symbol, it does not override the value of the use clauses specified for the VHDL netlisting options in the *Output* tab of the Design Entry Options dialog box. Instead, the values you specify on the VHDL_DECS symbol are appended to the list of use clauses you specified for the VHDL netlisting options in the *Output* tab of the Design Entry Options dialog box.

VALUE

VALUE is used by the GenRad interface for preparing the `.ckt` file and reports. The value is the device value. It is also used to update the VALUE label on the mechanical symbol. Used for discretes only.

GenRad is not supported in versions 14.0 and higher.

The syntax of the VALUE property in the device file is:

```
PACKAGEPROP VALUE <device value>
```

where <*device value*> is the value of the discrete part, for example, resistance.

VER

The VER property, attached to a component, defines the version of the instance of a component used in a schematic. You can replace a schematic component with its next version by using the *Component – Version* menu command. Two versions of a component may differ graphically but functionally, they are the equivalent representation of the component.

VERILOG_LIB

The VERILOG_LIB property, attached on the instance of a part, supports instance-specific binding for a library in the NC Verilog simulation flow. For example, if `ls00` is instantiated on a drawing, you have to add the following for instance-specific library binding support:

```
VERILOG_LIB=lsttl_models
```

where VERILOG_LIB is the property name and `lsttl_models` is the associated Verilog logical library name.

The property is translated to the following compiler directive in the simulation netlist:

```
`uselib lib=lwb_des_ex
`uselib
```

NC Verilog parses the netlist file (`verilog.v`), and wherever it finds such an attribute, it searches the logical library to find the model for the `ls00` part.

VERILOG_MODEL

The VERILOG_MODEL property, placed in the library description for a primitive, in a physical part table, or directly on an instance, changes the name of the Verilog module generated or selects a specific MODEL entry in a Verilog map file.

Syntax

```
VERILOG_MODEL=name
```

Example

```
VERILOG_MODEL=SN74LS00
```

By default, the name of the Verilog module generated is the same name as the original DRAWING name unless the part has a PART_NAME property defined. Use the VERILOG_MODEL property to specify the exact name of the Verilog module. A VERILOG_MODEL property placed on an instance overrides the same property placed in the library.

The VERILOG_MODEL property value is case-sensitive. If you do not adhere to the proper case while naming models, warnings may appear during netlist generation. For example, for this property assignment to an instance in a schematic - VERILOG_MODEL=vmodel, the corresponding verilog.map file must have a model defined as: MODEL 'vmodel'. An entry, such as MODEL 'VMODEL', does not work and warnings appear during the generation of the simulation netlist.

VERILOG_NAME

The VERILOG_NAME property, placed in the Verilog map file, specifies the actual name of the Verilog model. Internally, all names for modules are in lowercase characters. If the name of the module you are using is either in uppercase characters or contains a mixture of lowercase and uppercase characters, use the VERILOG_NAME property to specify how the module name appears in the netlist.

Syntax

```
VERILOG_NAME='name'
```

Example

```
VERILOG_NAME='TTL00'
```

VERILOG_PORT_NAME

The VERILOG_PORT_NAME property, attached to a component definition pin of a co-design die, contains the name of the port connected to the IC design net on this pin. This property is passed among IO Planner (IOP), SiP Layout, and System Connectivity Manager (SCM) to identify the logical name of pins. It also identifies pins for cross-probing between SiP Layout and IOP.

VERSION_ID

For internal use only.

VHDL_CONCAT

The VHDL_CONCAT property is attached to the symbol for a component that you want to use for:

- Merging a number of signals, ports, or signal aliases into a group and then routing the group to a port or instance with a single wire.
- Splitting a vectored signal or port into a number of signals of smaller width.

Ensure that the name of the output pin of the symbol is the highest alphanumeric value of all the pins on the symbol. For example, if the input pins are named AA, DD, and FF, you cannot name the output pin BB.

VHDL_INIT

The VHDL_INIT property, attached to a signal or the pins of a symbol, lets you assign the initial value of the signal for VHDL. The power signal is assigned that value in the VHDL netlist, generated by Allegro Design Entry HDL.

VHDL_MODE

The VHDL_MODE property, attached to interface signals or to pins on the symbol, lets you specify the direction of ports for the generated VHDL description.

Note: If a port has several pins, you need to attach the property on only one of the pins.

Add property...	To...
VHDL_MODE=IN	Declare the direction of the port as input.
VHDL_MODE=OUT	Declare the direction of the port as output.
VHDL_MODE=INOUT	Declare the direction of the port as inout.
VHDL_MODE=BUFFER	Declare the port as a buffer port.
VHDL_MODE=LINKAGE	Declare the port as a linkage port.

VHDL_MODEL

The VHDL_MODEL property, placed in the library description for a primitive (`chips.prt` file), in a physical part table, or directly on an instance, changes the name of the VHDL module generated or to selects a specific MODEL entry in a VHDL map file.

Syntax

```
VHDL_MODEL=name
```

Example

```
VHDL_MODEL=SN74LS241
```

By default, the name of the VHDL module generated is the same name as the original DRAWING name unless the part has a PART_NAME property defined. Use the VHDL_MODEL property to specify the exact name of the VHDL module. A VHDL_MODEL property placed on an instance overrides the same property placed in the library.

VHDL_NAME

The VHDL_NAME property, placed in the Verilog map file, specifies the actual name of the VHDL model. Internally, all names for modules are in lowercase characters. If the name of the module you are using is either in uppercase characters or contains a mixture of lowercase and uppercase characters, use the VHDL_NAME property to specify how the module name will appear in the netlist.

Syntax

```
VHDL_NAME='name'
```

Example

```
VHDL_NAME='SN74LS153'
```

VHDL_SCALAR_TYPE

The VHDL_SCALAR_TYPE property, attached to a signal, pin, or symbol, sets the default VHDL logic type for all scalar ports and signals in your drawing. You can place a VHDL_DECS symbol on your drawing and attach the VHDL_SCALAR_TYPE property to it to specify the default VHDL logic type for all scalar ports and signals in your drawing. If you do not use this

property on the VHDL_DECS symbol, all scalar ports and signals in your design are the STD_LOGIC type.

You can change the VHDL logic type for a specific scalar port by attaching the VHDL_SCALAR_TYPE property to a pin of the port. You can change the VHDL logic type for a specific scalar signal by attaching the property to the signal.

The legal values are STD_LOGIC, BIT, and all other legal VHDL scalar types.

VHDL_SLICE

The VHDL property is attached to a symbol. It specifies to the netlister that the component should be treated as a tap symbol that breaks out a set of bits indicated by the value of the BN property on the pin on the component.

VHDL_VECTOR_TYPE

The VHDL_VECTOR_TYPE property, attached to a symbol, pin, or signal, sets the default VHDL logic type for all vectored ports and signals in your drawing. You can place a VHDL_DECS symbol on your drawing and attach the VHDL_VECTOR_TYPE property to it to specify the default VHDL logic type for all vectored ports and signals in your drawing. If you do not use this property on the VHDL_DECS symbol, all vectored ports and signals in your design are the STD_LOGIC_VECTOR type.

You can change the VHDL logic type for a specific vectored port by attaching the VHDL_VECTOR_TYPE property to a pin of the port. You can change the VHDL logic type for a specific vectored signal by attaching the property to the signal.

The legal values are STD_LOGIC_VECTOR, BIT_VECTOR, and all other legal VHDL vector types.

VIA_AT_SMD_FIT

The VIA_AT_SMD_FIT property is attached to components/component pins, or symbols/ symbol pins. Property values are on or off. When on, it specifies that a via is allowed inside an SMD pin, but only if it is completely covered by the pin. When off, just the center of the via needs to be inside the pin. In cases where the property is attached to a symbol, the property is extended to all instances of the symbol.

VIA_AT_SMD_THRU

The VIA_AT_SMD_THRU property is attached to components/component pins, or symbols/ symbol pins. Property values are on or off. When on, it specifies that a through hole via, blind via, or microvia is allowed inside an SMD pin. When off, only partial vias are allowed. In cases where the property is attached to a symbol, the property is extended to all instances of the symbol.

VIA_BEST_FIT

The VIA_BEST_FIT property is attached to dynamic shapes when you complete the Shape Instance Parameters dialog box. It overrides the thermal connect style for the via if the chosen style does not provide sufficient thermal connects within the specified minimum and maximum number of thermals in 15-degree increments. The property values are TRUE (*Use best contact*) and FALSE (*Do not use best contact*). This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

VIA_CLEAR_TYPE

The VIA_CLEAR_TYPE property, attached to dynamic shapes, specifies where the Allegro PCB Editor gets the clearance value for how far away the copper should be kept from the thru pins: *Thermal/anti, DRC,* or *No Void*.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Clearances Tab* in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

VIA_LIST

The VIA_LIST property, attached to a net, is a list of the via names (wildcards allowed) that can be used for connections in this net. The VIA_LIST property has the same membership objects as the Physical and Spacing properties.

For information on this property mapping to Allegro PCB Router, see the *Routing the Design* user guide in your documentation set.

VIA_MAX_THERMS

The VIA_MAX_THERMS property, attached to dynamic shapes, specifies how many connect lines are created for thermal relief for vias. Values are 1-8. Up to four are allowed on orthogonal and diagonal clines. Up to eight are allowed on the 8-way option.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

VIA_MIN_THERMS

The VIA_MIN_THERMS, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the minimum number of connections for vias. Values are 1-8. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

VIA_OVERSIZE

The VIA_OVERSIZE property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies the oversize value for how far away the copper should be kept from the vias. This value is added to the clearance value of *DRC* or *Thermal/Anti.* By default, the oversize value is 0.

This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

VIA_THERM_CONN

The VIA_THERM_CONN property, attached to dynamic shapes when you complete the Shape Instance Parameters dialog box, specifies how vias with the same net name as the shape should be connected to the shape: *orthogonal, diagnonal, 8-way, full contact,* and *none*. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Thermal Relief Connects* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

VIA_Z_ENABLED

The VIA_Z_ENABLED boolean property, attached to a design, enables via z accounting for differential pairs, relative propagation delay, and delay DRC checks. Make sure that the value is set to TRUE.

VIAS_ALLOWED

The VIAS_ALLOWED property, attached to a route keepout, permits vias within the keepout.

VLOG_MODE

The VLOG_MODE property, attached to a pin or net, allows you to specify the direction of ports for the generated Verilog description. To specify the direction of ports, add the VLOG_MODE property to interface signals or pins on the symbol.

Note: If a port has several pins, attach the property to only one of the pins.

Add property...	To...
VLOG_MODE=INPUT	Declare the direction of the port as input.
VLOG_MODE=OUT	Declare the direction of the port as output.
VLOG_MODE=INOUT	Declare the direction of the port as inout.

Add property...	To...
VLOG_MODE=BUFFER	Declare the port as a buffer port.
VLOG_MODE=LINKAGE	Declare the port as a linkage port.

For more information on declaring port modes, see “Declaring Port Modes” in the *Working with Libraries and Components* chapter of the *Allegro Design Entry HDL User Guide*.

VLOG_NET_TYPE

The VLOG_NET_TYPE property, attached to a symbol, pin, or signal, sets the default Verilog logic type for all ports and signals in your drawing. You can place a VERILOG_DECS symbol on your drawing and attach the VLOG_NET_TYPE property to it to specify the default Verilog logic type for all ports and signals in your drawing. If you do not use this property on the VERILOG_DECS symbol, all the ports and signals in your design are the `WIRE` type.

You can change the Verilog logic type for a specific port by attaching the VLOG_NET_TYPE property to a pin of the port. You can change the Verilog logic type for a specific signal by attaching the property to the signal.

The legal values are `WIRE, WAND`, `WOR`, and all other legal Verilog types.

VOID_SAME_NET

The VOID_SAME_NET property, attached to a cline, allows you to selectively void a cline that is on the same net and layer as a shape to permit routing of return paths. Be sure that you set the value to TRUE.

VOLTAGE

The VOLTAGE property, attached to a net to indicate that it is a DC net, lets you indicate the voltage value, for example, a value of `2.2 V.` By default, nets with the VOLTAGE property display ratsnest connections using the POWER_AND_GROUND ratsnest display, that is, the boxed X feature. You can change this display by defining another type of ratsnesting for the net (adding RATSNEST_SCHEDULE property to the net).

You can define the VOLTAGE property on the POWER or GND symbols in the library or on the design instance. This will be propagated into the flow.

You should apply the VOLTAGE property if you are performing high-speed analysis on a net. If you do not mark a net as a VOLTAGE net, then SigNoise simulates it as a normal signal net. This can cause the creation of an Xnet, containing a large numbers of nets, since a DC net can typically tie together many different pull-up and pull-down resistors.

The `signal model` command (*Analyze– SI/EMI Sim– Model*) displays a warning if it finds nets that appear to be DC nets but do not have a VOLTAGE property.

You may experience significant performance improvements when you add the VOLTAGE property to nets with large pin counts.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the `net_properties` command and the *Allegro Constraint Manager User Guide* for additional information.

Note: Release 15.2 and higher no longer supports CHASSIS as a valid value for this property.

VOLTAGE_SOURCE_PIN

The VOLTAGE_SOURCE_PIN property, attached to a pin, identifies the voltage source when there are no independent voltage planes. This property is used by EMControl.

VOLT_TEMP_MODEL

The VOLT_TEMP_MODEL property, attached to a pin, indicates the voltage temperature model for the pin. This property is used by DF/Viable, a Cadence tool that performs reliability analysis. The values that are allowed depend on how other properties are used.

WB_LOOP_HEIGHT_GROUP

The WB_LOOP_HEIGHT_GROUP, automatically created for bondwires when they are added using APD's wirebond commands (*Route – Wirebond*), lets you associate this bondwire with a loop height profile. Thus, all bondwires with the same integer value for this property are associated with the same 3D arcing profile. This lets you determine how far from the edge of the die a cross between two wires of the same group can be before they are too close (hit each other) in 3D space. The range of values (`0-32767`) is the maximum number of profiles allowed.

WEIGHT

The WEIGHT property, attached to a pin, assigns a level of importance to a net using integer values of 0 to 100. A high weight value indicates that Allegro PCB Editor should shorten the net as much as possible.

Both automatic placement and automatic gate swapping use the WEIGHT property. Automatic placement uses the WEIGHT property to keep the components close together on the net. The following table shows the effect on component placement of various weights added to a net.

Weight	Effect on a Component
0	No consideration is given to placing the component close to other components on this net.
Less than 50	Placing components close to interconnected components is less important than for nets without a WEIGHT property attached.
50	The default value assigned to all nets. A value of 50 is the same as assigning no property.
Greater than 50	Components on this net are given preference for placement near interconnected components over nets with no WEIGHT property attached. Cadence recommends that you do not attach a single high value (90, for example) to every net on the design. If you do, automatic placement cannot determine which nets are critical. You could, for example, attach values of 89, 90, and 91, respectively, to three nets to show that all three are critical, and that the third is the most critical.

This net property and other general net properties appear in the *All* worksheet in the *Nets: General Properties* workbook of the Constraint Manager. See the net_properties command and the *Allegro Constraint Manager User Guide* for additional information.

WIREBOND_FINGER_SHAPE

The WIREBOND_FINGER_SHAPE property, used only by the tool, is automatically placed on shapes created by the *Wire Bond – Select – Fingers – Merge Fingers* operation. It indicates to the system that these are merged fingers and not a power/ground ring shape.

WIREBOND_PROFILE_NAME

The WIREBOND_PROFILE_NAME, attached to bond wires (clines), is a string that gives the name of the 3D profile for that wire. Replaced the WB_LOOP_HEIGHT_group property.

WIRE_LENGTH

The WIRE_LENGTH property, attached to a net, specifies the length of a routed connection for calculating a transmission line delay on a *BONDING_WIRE* layer in APD. This property is used by SigNoise.

WIREBOND_FINGER_SHAPE

Available in a future release.

WIREBOND_MATERIAL

Specifies a property that is automatically created and maintained whenever you use the Wire Profile Editor to change the MATERIAL name (default setting is GOLD) for a bond wire profile. This property is referenced by the wirebond commands and the analysis tools. All the wires associated with the specified profile are updated to match the new setting. You can view the new setting in the WIREBOND_MATERIAL field in the Show Element dialog box.

XHATCH_BORDER_WIDTH

The XHATCH_BORDER_WIDTH property, attached to dynamic crosshatch shapes, specifies the width of the shape boundary when you select *Xhatch* for fill style. The border width is not less than the crosshatch line width. This property appears in the Show Element dialog box only if you disable *Etch* and enable *Boundary* in the *Stackup Group* in the Color and Visibility dialog box. However, it is not available for editing in the Edit Property dialog box.

For additional information, see the description of the *Fill* tab in the Shape Instance Parameters Dialog Box in the *Allegro PCB and Package Physical Layout Command Reference.*

XTALK_ACTIVE_TIME

The XTALK_ACTIVE_TIME property, attached to a net, indicates the time when this net is active (changing voltage). This property is used by SigNoise to determine when this net has an effect on other nets. It is also used by constraints for crosstalk analysis.

XTALK_IGNORE_ NETS

The XTALK_IGNORE_NETS, attached to a net, overrides the *Crosstalk Ignore Nets* electrical constraint. It is a list of names (wildcards allowed) that SigNoise should ignore when doing crosstalk analysis of the net with this property. The ignore names can mean one of the following:

- A net name
- The name value of an ELECTRICAL_CONSTRAINT_SET property

  SigNoise ignores a neighbor net as a source of crosstalk if the primary (victim) net has an `IGNORE = NEIGHBOR NET PROPERTY` on it. This property is also used by constraints.

XTALK_SENSITIVE_TIME

The XTALK_SENSITIVE_TIME property, attached to a net, overrides the *Crosstalk Sensitive Time* electrical constraint. It is a list of times when this net is noise-sensitive to voltage changes of another net. The property is used by constraints and SigNoise to determine when this net is affected by other nets.

XR

The XR property, assigned by CRefer on nets, cross-references nets within a flat or hierarchical design.

For example, if a net occurs on two pages, page 1 and page 2 of a block, and also down the hierarchy from page 1, then CRefer assigns XR properties on page 1 for the hierarchy below it and one XR property for page 2. You can read these properties to cross-reference nets up or down a hierarchical design, or within a flat design.

Note: XR is a read-only property.

XY

The XY property, attached to a component, defines the x,y Allegro Design Entry HDL coordinates. The X and Y values range between `-16000` and `16000`, and indicate where the particular instance is placed in the schematic.

Appendix A: Property List by Product

Allegro Design Entry HDL Properties

Table A-1 shows the Allegro Design Entry HDL properties.

Table A-1 Allegro Design Entry HDL Properties

Property	Attach to	Tools that use this property
ALLOW_CONNECT	Components, symbols, nets, pins	ERC-DX, CheckPlus
ALT_SYMBOLS	Components	Packager XL, CheckPlus, Allegro PCB Editor
ALT_SYMBOLS_HARD	Components	Packager XL, CheckPlus, Allegro PCB Editor
BIDIRECTIONAL	Pins	ERC-DX, CheckPlus
BLOCK=TRUE	Symbols	Allegro Design Entry HDL
BN	Pins	Allegro Design Entry HDL
BODY_NAME	Components	Allegro Design Entry HDL, Packager XL
BODY_TYPE	Symbols	Allegro Design Entry HDL
BOM_IGNORE	Components	Allegro Design Entry HDL
BUBBLED	Components	Allegro Design Entry HDL
BUBBLE_GROUP	Components	Allegro Design Entry HDL
CDS_NOT_ON_SYM	In the `viewprps.prp` file for split parts	Allegro Design Entry HDL, Packager XL
CLASS	Components	Allegro Design Entry HDL, Packager XL, Allegro PCB Editor

Table A-1 Allegro Design Entry HDL Properties

Property	Attach to	Tools that use this property
COMMENT	Waived DRC or any other design object	Allegro Design Entry HDL, Allegro System Architect , Allegro PCB Editor, Constraint Manager, APD
COMMENT_BODY	Components	Allegro Design Entry HDL
COMP_NAME	Components	Packager XL
COMP_NAME_SUFFIX	Components	Packager XL
DIFFERENTIAL_PAIR	Nets	Allegro Design Entry HDL
DIFF_PAIR_PINS_NEG	Nets	Allegro Design Entry HDL
DIFF_PAIR_PINS_POS	Nets	Allegro Design Entry HDL
FAMILY	Components	Packager XL
GROUP	Section, components	Packager XL, Allegro PCB Editor
HAS_FIXED_SIZE	Components	Allegro Design Entry HDL
HDL_CONCAT	Symbols	Allegro Design Entry HDL
HDL_LSBTAP	Symbols	Allegro Design Entry HDL
HDL_MSBTAP	Symbols	Allegro Design Entry HDL
HDL_NOT	Symbols	Allegro Design Entry HDL
HDL_PORT	Symbols	Allegro Design Entry HDL
HDL_POWER	Symbols	Allegro Design Entry HDL
HDL_REPLICATE	Symbols	Allegro Design Entry HDL
HDL_SLASH	Symbols	Allegro Design Entry HDL
HDL_SYNONYM	Symbols	Allegro Design Entry HDL
HDL_TAP	Symbols	Allegro Design Entry HDL
HEIGHT	Components	Allegro Design Entry HDL
INCLUDE_IN_RF_TOPOLOGY	Component	Allegro Design Entry HDL, Allegro PCB Editor
INPUT_LOAD	Physical pins	ERC-DX, CheckPlus

Table A-1 Allegro Design Entry HDL Properties

Property	Attach to	Tools that use this property
JEDEC_TYPE	Components	Packager XL, CheckPlus, Allegro PCB Editor
LAST_MODIFIED	Drawing symbol	Allegro Design Entry HDL
LAYERSET_GROUP	Nets, buses, differential pairs	Allegro Design Entry HDL, Allegro PCB Editor, Constraint Manager, APD
LOCATION	Components	Allegro Design Entry HDL, Packager XL, Allegro PCB Editor, Variant Editor
MAKE_BASE	Nets	Allegro Design Entry HDL, Packager XL
MERGE_NC_PINS	Components, symbols	Packager XL
MERGE_POWER_PINS	Components, symbols	Packager XL
NC_PINS	Components, symbols	Packager XL
NEEDS_NO_SIZE	Components	ERC-DX, CheckPlus
NO_BACKANNOTATE	Components	Packager XL
NO_IO_CHECK	Components, pins	ERC-DX, CheckPlus
NO_LOAD_CHECK	Components, pins	ERC-DX, CheckPlus
OUTPUT_LOAD	Pins	ERC-DX, CheckPlus
OUTPUT_TYPE	Pins	Packager XL, ERC-DX
PACK_IGNORE	Components	Packager XL
PACK_SHORT	Pins	Packager XL
PACK_TYPE	Components	Allegro Design Entry HDL, Packager XL
PART_NAME	Components	Allegro Design Entry HDL, Packager XL
PART_NUMBER	Components	Allegro Design Entry HDL
PATH	Components	Allegro Design Entry HDL
PHYS_DES_PREFIX	Components	Packager XL

Table A-1 Allegro Design Entry HDL Properties

Property	Attach to	Tools that use this property
PIN_DELAY	Components, pins	Allegro Design Entry HDL
PIN_DELAY_ENABLED	Design	Allegro Design Entry HDL
PIN_GROUP	In `chips.prt` file	Packager XL
PIN_NAME	Symbols	Allegro Design Entry HDL
PIN_NUMBER	Pins	Packager XL
PINUSE	Pins	Allegro Design Entry HDL, Part developer
PN	Pins	Allegro Design Entry HDL
PNN	Nets	Packager XL
POWER_GROUP	Pins	Packager XL
POWER_PINS	Pins	Allegro Design Entry HDL, Packager XL
REF_DES_PATTERN	Components	Allegro Design Entry HDL
REUSE_ALT_MODULE	Components	yes
REUSE_ID	Components	Packager XL, Allegro PCB Editor
REUSE_INSTANCE	Components, symbols	Allegro Design Entry HDL
REUSE_INSTANCE	Components	Packager XL, Allegro PCB Editor
REUSE_NAME	Components	Packager XL, Allegro PCB Editor
REUSE_PID	Components	Packager XL
ROOM	Components	Packager-XL
ROTATE	Components	Packager XL, Allegro PCB Editor
SEC	Components	Packager XL
SEC_TYPE	Components	Allegro Design Entry HDL, Packager XL
SIG_NAME	Nets	Allegro Design Entry HDL

Table A-1 Allegro Design Entry HDL Properties

Property	Attach to	Tools that use this property
SIZE	Components	Allegro Design Entry HDL
SUBDESIGN_MASTER	Components, symbols	Packager XL
SUBDESIGN_SUFFIX	Symbols	Packager XL
SWAP_INFO	In the chips.prt file	Packager XL
TECH	Components	CheckPlus
UNFIXED_PINS	Components, pins	ERC-DX, CheckPlus
UNKNOWN_LOADING	Components, pins	ERC-DX, CheckPlus
VALUE	Components	Packager XL, Allegro PCB Editor
VER	Components	Packager XL
VHDL_CONCAT	Symbols	Allegro Design Entry HDL
VHDL_MODE	Pins, nets	Allegro Design Entry HDL
VHDL_SLICE	Symbols	Allegro Design Entry HDL
VLOG_MODE	Pins, nets	Allegro Design Entry HDL
XR	Nets	CRefer
XY	Components	Allegro Design Entry HDL

System Connectivity Manager Properties

Table A-2 shows the System Connectivity Manager properties.

Table A-2 System Connectivity Manager Properties

Property	Attach to
ALLOW_CONN_SWAP	Components
ALLOW_CONNECT	Components, symbols, nets, pins
ALT_SYMBOLS	Components
ALT_SYMBOLS_HARD	Components
BOM_IGNORE	Components

Table A-2 System Connectivity Manager Properties

Property	Attach to
COMMENT	Any design object
COMP_NAME	Components
COMP_NAME_SUFFIX	Components
ECL	Nets
EMC_COMP_TYPE	Components
EMC_CRITICAL_IC	Components
EMC_CRITICAL_NET	Nets
FIXED	Nets
GROUP	Components
HEIGHT	Components
IC_DESIGN_CELL_INSTANCE_NAME	Symbol pins
IC_DESIGN_CELL_MASTER_NAME	Symbol pins
IC_DESIGN_CELL_PIN_NAME	Symbol pins
IC_DESIGN_NET_NAME	Symbol pins
JEDEC_TYPE	Components
NO_BACKANNOTATE	Nets
NO_DRC	Pins
NO_FILLET	Nets
NO_PIN_ESCAPE	Nets
NO_RAT	Nets
NO_RIPUP	Nets
NO_ROUTE	Nets

Table A-2 System Connectivity Manager Properties

Property	Attach to
NO_TEST	Nets
PACK_IGNORE	Components
PACK_SHORT	Components
PART_NUMBER	Components
PATH	Components
PINUSE	Pins
PROBE_NUMBER	Nets
REMOVE	Components
ROOM	Components
ROUTE_PRIORITY	Nets
ROUTE_TO_SHAPE	Nets
SIM_MAP_VIEW	Components
SIZE	Components
TESTPOINT_QUANTITY	Nets
TOL	Components
VALUE	Components
VERILOG_MODEL	Components
VERILOG_PORT_NAME	Component definition pin
VHDL_MODEL	Components
VIA_LIST	Nets
VOLTAGE	Components, nets
VOLT_TEMP_MODEL	Components
WEIGHT	Nets

Logic Simulation Properties

Table A-3 shows the properties that are supported for simulation. These properties can be added in Allegro Design Entry HDL and are used by these tools:

- Allegro AMS Simulator 210
- AWB
- ATDM
- NC-Verilog
- NC-VHDL
- Verilog XL
- Leapfrog
- Verilog
- VHDL

Table A-3 Simulation Properties

Property	Attached to	Tools that use this property
COMMENT	Any design object	Simulation Netlister, Allegro Design Entry HDL
LIBRARYn	Symbol In map file	Simulation Netlister
MODEL_DIR	Components, symbols	Simulation Netlister
MODEL_FILE	Components, symbols	Simulation Netlister
NO_REP_PRIM	Components, symbols	Simulation Netlister
PORT_ORDER	In map file, Components	Simulation Netlister, Allegro Design Entry HDL
REMOVE	Components, symbols	Simulation Netlister
SDFDELAYTYPE	Symbols	Simulation Netlister
SDFFILE	Symbols	Simulation Netlister
SDFSCALEFACTOR	Symbols	Simulation Netlister
SDFSCALETYPE	Symbols	Simulation Netlister

Table A-3 Simulation Properties

Property	Attached to	Tools that use this property
SIM_BIND_VIEW	Components, symbols	Simulation Netlister
SIM_MAP_VIEW	Components, symbols	Simulation Netlister
SIZE	Components, symbols	Simulation Netlister, Allegro Design Entry HDL
SPLIT_INST	Components, symbols	Simulation Netlister
SPLIT_INST_NAME	Components, symbols	Simulation Netlister
USEn	Symbols	Simulation Netlister
VERILOG_LIB	Components, symbols	Simulation Netlister
VERILOG_MODEL	Components, symbols In Verilog map file	Simulation Netlister
VERILOG_NAME	In Verilog map file	Simulation Netlister
VHDL_INIT	Symbols, signal	Simulation Netlister
VHDL_MODEL	Components, symbols, In VHDL map file	Simulation Netlister
VHDL_NAME	In Verilog map file	Simulation Netlister
VHDL_SCALAR_TYPE	Symbols, pins, signal	Simulation Netlister
VHDL_VECTOR_TYPE	Symbols, pins, signal	Simulation Netlister
VLOG_NET_TYPE	Symbols, pins, signal	Simulation Netlister

Allegro PCB Editor Properties

Table A-4 shows the properties that are used in Allegro PCB Editor.

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
ALIGNED	Vias	no

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
ALT_SYMBOLS	Device	yes
ALT_SYMBOLS_HARD	Device	yes
ASSIGN_ROUTE_LAYER	Pins, nets	no
ASSIGN_TOPOLOGY	Nets	yes
AUTO_GENERATED_TERM	Components	no
AUTO_RENAME	Refdes	no
BOARD_THICKNESS	Design	no
BOM_IGNORE	Components	yes
BOND_PAD	Vias	no
BOND_WIRE	Clines	no
BUS_NAME	Nets	yes
CDS_XNET_NAME	Nets	yes
CLIP_DRAW	Design (board), symbols	no
CLIP_DRAWING	Clines, device, pins, filled rectangles, lines, rectangles, shapes, symbols, vias, voids	no
CLK_2OUT_MAX	Nets, pins	no
CLK_2OUT_MIN	Nets, pins	no
CLK_SKEW_MAX	Nets, pins	no
CLK_SKEW_MIN	Nets, pins	no
CLOCK_NET	Nets	yes
COMMENT	Waived DRC or any other design object	yes
COMPONENT_WEIGHT	Reference designators	yes
DENSE_COMPONENT	Reference designators	yes
DFA_DEV_CLASS	Board, symbols	no

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
DIFFP_COUPLED_MINUS	Nets	yes
DIFFP_COUPLED_PLUS	Nets	yes
DIFFP_GATHER_CONTROL	Nets	yes
DIFFP_MIN_SPACE	Nets	yes
DIFFP_NECK_GAP	Nets	yes
DIFFP_PHASE_TOL (formerly DIFFP_LENGTH_TOL)	Nets	yes
DIFFP_PRIMARY_GAP	Nets	yes
DIFFP_UNCOUPLED_LENGTH (formerly DIFFP_2ND_LENGTH)	Nets	yes
DRC_UNROUTED_MINPROP	Design	no
DRC_UNROUTED_RELPROP	Design	no
DRIVER_TERM_VAL	Nets	no
DYN_CLEARANCE_OVERSIZE	Shapes, frectangle, pins, vias, clines, lines	no
DYN_CLEARANCE_TYPE	Pins, vias	no
DYN_DELETED_ISLAND	Shapes	no
DYN_DO_NOT_VOID	Shapes, frectangles, lines, clines	no
DYN_FIXED_THERM_WIDTH	Pins, vias	no
DYN_MAX_THERMAL_CONNS	Pins, vias	no
DYN_MIN_THERMAL_CONNS	Pins, vias	no
DYN_OVERSIZE_THERM_WIDTH	Pins, vias, dynamic shapes	no
DYN_THERMAL_BEST_FIT	Pins, vias	no
DYN_THERMAL_CON_TYPE	Pins, vias	no
ECL	Nets	yes

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
ECL_TEMP	Nets	yes
EDGE_SENS	Nets (XNet in Constraint Manager)	no
ELECTRICAL_CONSTRAINT_SET	Nets	yes
EMC_COMP_TYPE	Components, device	yes
EMC_CRITICAL_IC	Components, device	yes
EMC_CRITICAL_NET	Nets	yes
EMC_CRITICAL_REGION	Shapes	yes
EMC_RUN_DIR	Board	no
ETCH_TURN_UNDER_PAD	Components, Component pins, Symbols, Symbol pins	no
EXTERNAL_DRC_VALUE	Board	no
FILLET	Clines	no
FIRST_INCIDENT	Nets	no
FIX_ALL	Reference designators	yes
FIXED	Reference designators, symbols, clines, filled rectangles, lines, nets, pins, rectangles, shapes, vias	yes
FIXED_T_TOLERANCE	TPoints	no
FP_BOARD_CLEARANCE	Not accessible to user	no
FP_NOTES_NO_EDIT	Shapes	no
FP_NOTES_TEXT_BLOCK	Not accessible to user	no
FP_REFDES_TEXT_BLOCK	Not accessible to user	no

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
FP_ROOM_NAME_TEXT_BLOCK	Not accessible to user	no
GROUP	Functions	yes
HARD_LOCATION	Reference designators, function designators, Components	yes, but not seen in schematic as LOCATION
IDF_OTHER_OUTLINE	Shapes, rectangles, filled rectangles	yes
IDF_OWNER	All objects except components	no
IMPEDANCE_RULE	Nets	no
INCLUDE_IN_RF_TOPOLOGY	Component	yes
INLINE_PIN_VOIDS	Dynamic shapes	no
ISRFELEMENT	Components	no
J_TEMPERATURE	Reference designators	yes
LAYERSET_GROUP	Nets, buses, differential pairs	yes
LAST_PIN_SWAP	Pins	no
LEAD_DIAMETER	Board, pins	no
LEFDEF_SPECIAL_NET	Nets	no
LINE_OVERSIZE	Dynamic shapes	no
LOAD_TERM_VAL	Nets	no
LOGICAL_PATH	Function designators (Component)	yes, but assigned by PXL, not user-defined
MATERIAL	Obsolete property	
MAX_BOND_LENGTH		no
MAX_BVIA_STAGGER	Nets	no
MAX_LINE_EXIT_ANGLE	Symbols	

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
MAX_EXPOSED_LENGTH	Nets	yes
MAX_FINAL_SETTLE	Nets	yes
MAX_LINE_WIDTH	Nets, XNets, buses, differential pairs	yes
MAX_OVERSHOOT	Nets	yes
MAX_PARALLEL (formerly PARALLELISM)	Nets, clines	no
MAX_PEAK_XTALK (formerly MAX_PEAK_CROSSTALK)	Nets	no
MAX_POWER_DISSIPATION	Obsolete property	
MAX_SSN	Nets	no
MAX_UNDERSHOOT	Clines	yes
MAX_VIA_COUNT	Nets	yes
MAX_XTALK (formerly MAX_CROSSTALK)	Nets, clines	no
MIN_BVIA_GAP	Nets	no
MIN_BOND_LENGTH	Nets, clines	no
MIN_BVIA_STAGGER	Nets	no
MIN_FIRST_SWITCH	Nets	no
MIN_HOLD	Nets, pins	yes
MIN_LINE_WIDTH	Nets, clines	yes
MIN_NECK_WIDTH	Nets, clines	yes
MIN_NOISE_MARGIN	Net	yes
MIN_SETUP	Nets, pins	yes
MIN_SHAPE_SIZE	Dynamic shapes	no
NET_SCHEDULE	Nets	no

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
NET_SHORT	Pins, vias	yes
NO_BACKANNOTATE	Nets, shapes	yes
NO_DRC	Pins, vias	no
NODRC_COMPONENT_BOARD_OVERLAP	A shape (package boundary)	no
NODRC_ETCH_OUTSIDE_KEEPIN	An etch item, for example, shapes, rectangles, lines, or clines	no
NODRC_SYM_SAME_PIN	Design, symbol, symbol definition	no
NODRC_VIAS_OUTSIDE_KEEPIN	Vias	no
NO_FILLET	Nets, pins, or vias	yes
NO_LIN2SHAPE_FAT	Clines	no
NO_PIN_ESCAPE	Reference designators, nets, pins	yes
NO_RAT	Nets	yes
NO_RIPUP	Nets	yes
NO_ROUTE	Reference designators, nets	yes
NO_SHAPE_CONNECT	Pins, vias	yes
NO_SWAP_COMP	Components	no
NO_SWAP_GATE	Reference designators, function designators, components	yes, but assigned by PXL. See PXL documentation.
NO_SWAP_GATE_EXT	Function designators, components	yes, but assigned by PXL. See PXL documentation.

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
NO_SWAP_PIN	Reference designators, function designator, pins, components	yes, but assigned by PXL. See PXL documentation.
NO_TEST	Nets	yes
NO_VIA_CONNECT	Pins, vias	no
PACKAGE_HEIGHT_MAX and PACKAGE_HEIGHT_MIN	Rectangles, shapes	no
PIN_DELAY	Pins	yes
PIN_DELAY_ENABLED	Pins	
PIN_ESCAPE	Reference designators, pins, components	yes
PIN_SIGNAL_MODEL	Pins	no
PINUSE	Pins	yes, but assigned by PXL. See PXL documentation.
PLACE_TAG	Reference designators	no
PLATING	Shapes	no
PROBE_NUMBER	Nets	yes
PROPAGATION_DELAY	Nets	no
PULSE_PARAM	Nets, bus, differential pairs	no
RATSNEST_ SCHEDULE	Nets	no
REF_DES_FOR_ASSIGN	Functions	no
RELATIVE_PROPAGATION_DELAY	Nets	no
REUSE_ALT_MODULE	Not accessible to user	
REUSE_ID	Components, symbols	
REUSE_INSTANCE	Components, symbols, nets	yes

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
REUSE_MODULE	Components, symbols	yes
REUSE_NAME	Components, symbols	yes
REUSE_PID	For internal use only	
RFELEMENTTYPE	Components	
RFPCB_OBJECT	Shapes and lines	
RF_TLINE	Components, tlines	yes
ROOM	Reference Designators, function designators, components, shapes	yes
ROOM_TYPE	Shapes	no
ROUTE_PRIORITY	Nets	yes
ROUTES_ALLOWED	Shapes, rectangles	no
SAME_NET	Nets	no
SAME_NET_XTALK_ENABLED	Nets	no
SCHEMATIC_NAME	Board	yes, but passed by PXL, not user-assigned
SHAPE_OVERSIZE	Dynamic shapes	no
SHIELD_NET	Nets	yes
SHIELD_TYPE	Nets	yes
SHORTING_SCHEME	Pins, vias, clines, shapes, nets	no
SIGNAL_MODEL	Reference designators, clines	yes
SLOTNAME	Functions	no
SMD_BEST_FIT	Dynamic shapes	no

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
SMD_CLEAR_TYPE	Dynamic shapes	no
SMD_MAX_THERMS	Dynamic shapes	no
SMD_MIN_THERMS	Dynamic shapes	no
SMD_OVERSIZE	Dynamic shapes	no
SMD_THERM_CONN	Dynamic shapes	no
SMOOTH_MIN_GAP	Dynamic shapes	no
SMOOTH_TRIM_CONTROL	Dynamic shapes	no
SNAP_VOID_XHATCH	Dynamic shapes	no
SOLDER_BALL_HEIGHT	Symbols	no
SPIF_CONSTANTS	Board	no
SPIF_TURRET	Clines, frect, shapes	no
STUB_LENGTH	Nets, clines	yes
SUBNET_NAME	Pins, nets, vias, clines, shapes	yes
SWAP_GROUP	Function designators	yes, but assigned by PXL, not user-assigned
SYS_CONFIG_NAME	Board	no
TEMPORARY_PACKAGE_SYMBOL	Reference designators	yes
TERMINATOR_PACK	Device	no
TESTER_GUARDBAND	Nets, pins	no
TESTPOINT_QUANTITY	Nets	yes
TESTPOINT_ALLOW_UNDER	Symbols	yes
TEXT_OVERSIZE	Dynamic shapes	no
THERMAL_RELIEF	Thermal clines	no
THICKNESS	Obsolete property	

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
THRU_BEST_FIT	Dynamic shapes	no
THRU_CLEAR_TYPE	Dynamic shapes	no
THRU_MAX_THERMS	Dynamic shapes	no
THRU_MIN_THERMS	Dynamic shapes	no
THRU_OVERSIZE	Dynamic shapes	no
THRU_THERM_CONN	Dynamic shapes	no
TIMING_DELAY_OVERRIDE	Nets, pins (seen in Constraint Manager)	no
TOL	Device	yes
TOPOLOGY_TEMPLATE	Nets	yes
TOPOLOGY_TEMPLATE_REVISION	Nets	yes
TOTAL_ETCH_LENGTH	Nets, bus, differential pairs	no
TS_ALLOWED	Nets	yes
UNFIXED_PINS	Board, symbols	no
VALUE	Discrete device	yes
UNUSED_PADS_IGNORE	Symbols, nets, pins, or vias	no
VIA_AT_SMD_FIT	Components, Component pins, Symbols, Symbol pins, Dynamic shapes	no
VIA_AT_SMD_THRU	Components, Component pins, Symbols, Symbol pins, Dynamic shapes	no
VIA_CLEAR_TYPE	Dynamic shapes	no
VIA_LIST	Nets	no
VIA_MAX_THERMS	Dynamic shapes	no
VIA_MIN_THERMS	Dynamic shapes	no

Table A-4 Allegro PCB Editor Properties

Property	Attach to	Property can be assigned in the schematic?
VIA_OVERSIZE	Dynamic shapes	no
VIA_THERM_CONN	Dynamic shapes	no
VIA_Z_ENABLED	Design	no
VIAS_ALLOWED	Route keepout, shapes	no
VOLTAGE	Reference designators, nets	yes
VOLTAGE_SOURCE_PIN	Pins	yes
WB_LOOP_HEIGHT_GROUP	Clines	no
WEIGHT	Nets	no
WIREBOND_FINGER_SHAPE	Shapes	no
WIREBOND_PROFILE_NAME	Bond wires (clines)	no
WIRE_LENGTH	Clines	yes
XHATCH_BORDER_WIDTH	Dynamic shapes	no
XTALK_ACTIVE_TIME	Nets	yes
XTALK_IGNORE_ NETS	Nets	no
XTALK_SENSITIVE_TIME	Nets, pins	yes

Constraint Manager Properties

Table A-5 shows the Constraint Manager properties.

Table A-5 Constraint Manager Properties

Property	Attach to	Property can be assigned in the schematic?
ASSIGN_TOPOLOGY	Nets	yes
CLK_2OUT_MAX	Nets, pins	no
CLK_2OUT_MIN	Nets, pins	no
CLK_SKEW_MAX	Nets, pins	no
CLK_SKEW_MIN	Nets, pins	no
CLOCK_NET	Nets	yes
COMMENT	Waived DRC or any other design object	yes
DEGAS_NO_VOID	Nets	no
DIFFP_COUPLED_MINUS	Nets	yes
DIFFP_COUPLED_PLUS	Nets	yes
DIFFP_GATHER_CONTROL	Nets	yes
DIFFP_MIN_SPACE	Nets	yes
DIFFP_NECK_GAP	Nets	yes
DIFFP_PHASE_TOL (formerly DIFFP_LENGTH_TOL)	Nets	yes
DIFFP_PRIMARY_GAP	Nets	yes
DIFFP_UNCOUPLED_LENGTH (formerly DIFFP_2ND_LENGTH)	Nets	yes
EDGE_SENS	Xnet	no
FIRST_INCIDENT	Xnet	no
LAYERSET_GROUP	Nets, buses, differential pairs	yes
MAX_OVERSHOOT	Nets	yes

Table A-5 Constraint Manager Properties

Property	Attach to	Property can be assigned in the schematic?
MAX_SSN	Nets	no
MAX_VIA_COUNT	Nets	yes
MAX_XTALK (formerly MAX_CROSSTALK)	Nets, clines	no
MIN_FIRST_SWITCH	Nets	no
MIN_HOLD	Nets	yes
MIN_NOISE_MARGIN	Nets	yes
MIN_SETUP	Nets, pins	yes
NET_SCHEDULE	Nets	no
PIN_DELAY	Pins	Allegro Design Entry HDL
PIN_DELAY_ENABLED	Design	
PROPAGATION_DELAY	Nets	no
PROPAGATION_DELAY_ACTUAL	Nets	no
PULSE_PARAM	Nets, Xnets, buses, differential pairs	no
RATSNEST_ SCHEDULE	Nets	no
RELATIVE_PROPAGATION_DELAY	Nets	no
STUB_LENGTH	Nets	yes
TIMING_DELAY_OVERRIDE	Nets, pins	yes
TOTAL_ETCH_LENGTH	Nets	no
XTALK_ACTIVE_TIME	Xnet, nets, buses, differential pairs	yes
XTALK_IGNORE_ NETS	Nets	no
XTALK_SENSITIVE_TIME	Nets	yes

Allegro Package Designer/SiP Digital Architect/SiP Layout Properties

Table A-6 shows the properties found in APD and SiP Digital Architect/SiP Layout.

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
ALIGNED	Bond pads (vias)	no
ALT_SYMBOLS	Device	yes
ALT_SYMBOLS_HARD	Device	yes
ASSIGN_ROUTE_LAYER	Pins, nets	no
AUTO_RENAME	Reference designators	no
BOND_PAD	Bond pads (vias)	no
BONDPAD_TO_BBV_SPACING	Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region	no
BONDPAD_TO_BONDPAD_DIFFP_SPC	board	no
BONDPAD_TO_BONDPAD_SPACING	Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region	no
BONDPAD_TO_MVIA_SPACING	Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region	no
BONDPAD_TO_SHAPE_SPACING	Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region	no
BONDPAD_TO_TESTPIN_SPACING	Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region	no
BONDPAD_TO_THRUVIA_SPACING	Net, Class_Class, XNet, Pin_Pair, Diff_Pair, Bus, Net_Class, or Region	no

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
BUS_NAME	Nets	yes
CDS_XNET_NAME	Nets	no
CLIP_DRAW	Design (board), symbols, components	no
CLIP_DRAWING	Clines, device, pins, filled rectangle, lines, rectangle, shapes, symbols, vias, voids	no
CLK_2OUT_MAX	Nets, pins	no
CLK_2OUT_MIN	Nets, pins	no
CLK_SKEW_MAX	Nets, pins	no
CLK_SKEW_MIN	Nets, pins	no
CLOCK_NET	Nets	yes
COMMENT	Waived DRC or any other design object	yes
COMPONENT_WEIGHT	Reference designators	yes
CONDUCTOR_MATERIAL	Symbols used as interposers	no
CONDUCTOR_THICKNESS	Symbols used as interposers	no
DEGAS_NO_VOID	Nets, pins, vias, shapes, and clines in APD Nets in Constraint Manager	no
DENSE_COMPONENT	Reference designators	yes
DFA_DEV_CLASS	Board, symbols	no
DIELECTRIC_MATERIAL	Symbols used as interposers	no

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
DIELECTRIC_THICKNESS	Symbols used as interposers	no
DIFFP_COUPLED_MINUS	Nets	yes
DIFFP_COUPLED_PLUS	Nets	yes
DIFFP_GATHER_CONTROL	Nets	yes
DIFFP_MIN_SPACE	Nets	yes
DIFFP_NECK_GAP	Nets	yes
DIFFP_PHASE_TOL (formerly DIFFP_LENGTH_TOL)	Nets	yes
DIFFP_PRIMARY_GAP	Nets	yes
DIFFP_UNCOUPLED_LENGTH (formerly DIFFP_2ND_LENGTH)	Nets	yes
DRC_UNROUTED_MINPROP	Design	
DRC_UNROUTED_RELPROP	Design	
DRIVER_TERM_VAL	Nets	no
DYN_CLEARANCE_OVERSIZE	Shapes, frectangles, pins, vias, clines	no
DYN_CLEARANCE_TYPE	Pins, vias	no
DYN_DELETED_ISLAND	Shapes	no
DYN_DO_NOT_VOID	Shapes, frectangles, lines, clines	no
DYN_FIXED_THERM_WIDTH	Pins, vias	no
DYN_MAX_THERMAL_CONNS	Pins, vias	no
DYN_MIN_THERMAL_CONNS	Pins, vias	no
DYN_OVERSIZE_THERM_WIDTH	Pins, vias, dynamic shapes	no
DYN_THERMAL_BEST_FIT	Pins, vias	no

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
DYN_THERMAL_CON_TYPE	Pins, vias	no
ECL	Nets	yes
ECL_TEMP	Nets	yes
EDGE_SENS	Nets	no
ELECTRICAL_CONSTRAINT_SET	Nets	yes
EMC_COMP_TYPE	Components, device	yes
EMC_CRITICAL_IC	Components, device	yes
EMC_CRITICAL_NET	Nets	yes
EMC_CRITICAL_REGION	Shapes	yes
EMC_RUN_DIR	Board	no
EXTERNAL_DRC_VALUE	Board	no
FILLET	Clines	no
FIRST_INCIDENT	Nets	no
FIX_ALL	Reference designators	yes
FIXED	Reference designators, symbols, clines, filled rectangles, lines, nets, pins, rectangles, shapes, vias	yes
FIXED_T_TOLERANCE	TPoints	no
FP_BOARD_CLEARANCE	Board	no
FP_NOTES_TEXT_BLOCK	Board	no
FP_REFDES_TEXT_BLOCK	Board	no
FP_ROOM_NAME_TEXT_BLOCK	Board	no
HARD_LOCATION	Reference designators, function designators	yes, but not seen in schematic as LOCATION

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
IDF_OTHER_OUTLINE	Shapes, rectangles, filled rectangles	yes
IDF_OWNER	All objects	no
IMPEDANCE_RULE	Nets	no
INLINE_PIN_VOIDS	Dynamic shape	
J_TEMPERATURE	Reference designators	yes
LAYERSET_GROUP	Nets, buses, differential pairs	yes
LEAD_DIAMETER	Board, symbols	no
LEFDEF_SPECIAL_NET	Nets	
LINE_OVERSIZE	Dynamic shapes	
LOAD_TERM_VAL	Nets	no
LOGICAL_PATH	Function designators	yes, but assigned by PXL, not user-defined
MAX_BOND_LENGTH	Nets	no
MAX_BVIA_STAGGER	Nets	yes
MAX_EXPOSED_LENGTH	Nets	yes
MAX_FINAL_SETTLE	Nets	yes
MAX_LINE_EXIT_ANGLE	Symbols	no
MAX_LINE_WIDTH	Nets, XNets, buses, differential pairs	yes
MAX_OVERSHOOT	Nets	yes
MAX_PARALLEL (formerly PARALLELISM)	Nets, clines	no
MAX_PEAK_XTALK (formerly MAX_PEAK_CROSSTALK)	Nets	no
MAX_SSN	Nets	no

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
MAX_UNDERSHOOT	Nets, clines	yes
MAX_VIA_COUNT	Nets	yes
MAX_XTALK (formerly MAX_CROSSTALK)	Nets, clines	no
MIN_BVIA_GAP	Nets	no
MIN_BOND_LENGTH		
MIN_BVIA_STAGGER	Nets	no
MIN_FIRST_SWITCH	Nets	no
MIN_HOLD	Nets, pins	yes
MIN_LINE_WIDTH	Nets, clines	yes
MIN_NECK_WIDTH	Nets, clines	yes
MIN_NOISE_MARGIN	Nets	yes
MIN_SETUP	Nets, pins	yes
MIN_SHAPE_SIZE	Dynamic shapes	
NET_SCHEDULE	Nets	no
NET_SHORT	Pins, vias	yes
NO_BACKANNOTATE	Nets, constraint areas (shapes, rectangle)	yes
NO_DRC	Pins, vias	no
NODRC_COMPONENT_BOARD_OVERLAP	Shapes (package boundary)	no
NODRC_ETCH_OUTSIDE_KEEPIN	An etch item, for example, shapes, rectangles, lines, or clines	no
NODRC_SYM_SAME_PIN	Board, symbols, symbol definitions	no

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
NODRC_VIAS_OUTSIDE_KEEPIN	Vias	no
NO_FILLET	Nets, pins, or vias	yes
NO_LIN2SHAPE_FAT	Clines	no
NO_PIN_ESCAPE	Reference designators, nets, pins	yes
NO_RAT	Nets	yes
NO_RIPUP	Nets	yes
NO_ROUTE	Reference designators, nets	yes
NO_SHAPE_CONNECT	Pins, vias	yes
NO_SWAP_COMP	Components	
NO_SWAP_GATE	Reference designators, function designators	yes, but assigned by PXL. See PXL documentation.
NO_SWAP_GATE_EXT	Function designators	yes, but assigned by PXL. See PXL documentation.
NO_SWAP_PIN	Reference designators, function designators, pins	yes, but assigned by PXL. See PXL documentation.
NO_TEST	Nets	yes
NO_VIA_CONNECT	Pins, vias	no
NO_WIREBOND	Pins, nets	yes
PACKAGE_HEIGHT_MAX and PACKAGE_HEIGHT_MIN	Rectangles, shapes	no
PIN_DELAY	Components, pins	Allegro Design Entry HDL
PIN_DELAY_ENABLED		

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
PIN_ESCAPE	Reference designators, pins	yes
PIN_SIGNAL_MODEL	Pins	no
PINUSE	Pins	yes, but assigned by PXL. See PXL documentation.
PLACE_TAG	Reference designators	no
PLATING	Shapes	no
PROBE_NUMBER	Nets	yes
PROPAGATION_DELAY	Nets	no
PULSE_PARAM	Nets, buses, differential pairs	no
RATSNEST_ SCHEDULE	Nets	no
REF_DES_FOR_ASSIGN	Functions	no
RELATIVE_PROPAGATION_DELAY	Nets	no
REUSE_ID	Components	yes
REUSE_INSTANCE	Components, symbols	
REUSE_MODULE	Components	yes
REUSE_NAME	Components	yes
REUSE_PID	Components, symbols	
ROOM	Reference designators function designators	yes
RF_TLINE	Components, tlines	yes
ROOM_TYPE	Room boundary	no
ROUTE_PRIORITY	Nets	yes
ROUTE_TO_SHAPE	Nets	no
SAME_NET	Nets	no

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
SAME_NET_XTALK_ENABLED	Nets	no
SCHEMATIC_NAME	Board	yes, but passed by PXL, not user-assigned
SHAPE_OVERSIZE	Dynamic shapes	no
SHIELD_NET	Nets	yes
SHIELD_TYPE	Nets	yes
SHORTING_SCHEME	Pins, vias, clines, shapes, nets	no
SIGNAL_MODEL	Reference designators	yes
SLOTNAME	Function	no
SMD_BEST_FIT	Dynamic shapes	no
SMD_CLEAR_TYPE	Dynamic shapes	no
SMD_MAX_THERMS	Dynamic shapes	no
SMD_MIN_THERMS	Dynamic shapes	no
SMD_OVERSIZE	Dynamic shapes	no
SMD_THERM_CONN	Dynamic shapes	no
SMOOTH_MIN_GAP	Dynamic shapes	no
SMOOTH_TRIM_CONTROL	Dynamic shapes	no
SNAP_VOID_XHATCH	Dynamic shapes	no
SOLDER_BALL_HEIGHT	Symbols	no
SOV_CHECK	Nets	no
SPIF_CONSTANTS	Board	no
SPIF_TURRET	Clines, frects, shapes	no
STUB_LENGTH	Nets	yes
SUBNET_NAME	Pins, nets, vias, clines, shapes	yes

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
SWAP_GROUP	Function designators	yes, but assigned by PXL, not user-assigned
SWAP_INFO		
SYS_CONFIG_NAME	Board	no
TEMPORARY_PACKAGE_SYMBOL	Reference designators	yes
TERMINATOR_PACK	Device	no
TESTER_GUARDBAND	Nets, pins	no
TESTPOINT_QUANTITY	Nets	yes
TESTPOINT_ALLOW_UNDER	Symbols	yes
TEXT_OVERSIZE		no
THERMAL_RELIEF	Thermal clines	no
THICKNESS	Obsolete property	
THRU_BEST_FIT	Dynamic shapes	no
THRU_CLEAR_TYPE	Dynamic shapes	no
THRU_MAX_THERMS	Dynamic shapes	no
THRU_MIN_THERMS	Dynamic shapes	no
THRU_OVERSIZE	Dynamic shapes	no
THRU_THERM_CONN	Dynamic shapes	no
TIMING_DELAY_OVERRIDE	Nets, pins (seen in Constraint Manager)	no
TOL	Device	yes
TOPOLOGY_TEMPLATE	Nets	yes
TOPOLOGY_TEMPLATE_REVISION	Nets	yes
TOTAL_ETCH_LENGTH	Nets, buses, differential pairs	yes
TS_ALLOWED	Nets	yes

Table A-6 APD/SiP Digital Architect/SiP Layout Properties

Property	Attach to	Property can be assigned in the schematic?
UNFIXED_PINS	Board, symbols	no
VALUE	Discrete device	yes
VIA_AT_SMD_FIT	Dynamic shapes	no
VIA_CLEAR_TYPE	Dynamic shapes	no
VIA_LIST	Nets	no
VIA_MAX_THERMS	Dynamic shapes	no
VIA_MIN_THERMS	Dynamic shapes	no
VIA_OVERSIZE	Dynamic shapes	no
VIA_THERM_CONN	Dynamic shapes	no
VOLTAGE	Reference designators	yes
VOLTAGE_SOURCE_PIN	Nets	yes
VOLT_TEMP_MODEL	Pins	yes
WB_LOOP_HEIGHT_GROUP	Bondwires (clines)	no
WEIGHT	Pins	no
WIREBOND_FINGER_SHAPE	Shapes	no
WIREBOND_PROFILE_NAME	Bond wires (clines)	no
WIRE_LENGTH	Nets	yes
XHATCH_BORDER_WIDTH	Dynamic shapes	no
XTALK_ACTIVE_TIME	Nets	yes
XTALK_IGNORE_ NETS	Nets	no
XTALK_SENSITIVE_TIME	Nets	yes