STLIB 4.4 COMPONENTS LIBRARY

USER'S MANUAL

STlib User Manual

1 Introduction

This library along with its related files are managed by the Design Kit environment. The aim of *STlib* library is to provide the user with a front end technology independent library (no layout provided) suited for simulation purposes and which completes the Cadence *analogLib* library.

This library should be used along with the *devices_symbols* library provided for a particular technology, by the Design Kit environment as some of the components of *STlib* are mapped on to the *devices_symbols* i.e. these components have their schematic representation built up of the components of the *devices_symbols*.

Please note, that a number of *devices_symbols* libraries are present catering for different technologies but the user will be able to access only one *devices_symbols* suited for the type of technology (s)he desires. Since the *STlib* library is technology independent and parameterization is used for those components which have to be customised for the technology chosen by the user, only one "STlib" library is used for the different *devices_symbols* libraries mentioned above.

The *STlib* library is automatically installed in the users library search path when initializing a design kit.

Many files are used by the *STlib* for its operation but it must be noted that the management of these files is transparent to the user as it is done by the Design Kit environment. The only file which may need to be modified by the user is the ".simrc" file present in the users home directory. This file **should be modified only if the user wants to obtain a CDL netlist at the gate level** (refer to chapter 4 for full details).

A complete organization of all the categories and examples of the cells within follows.

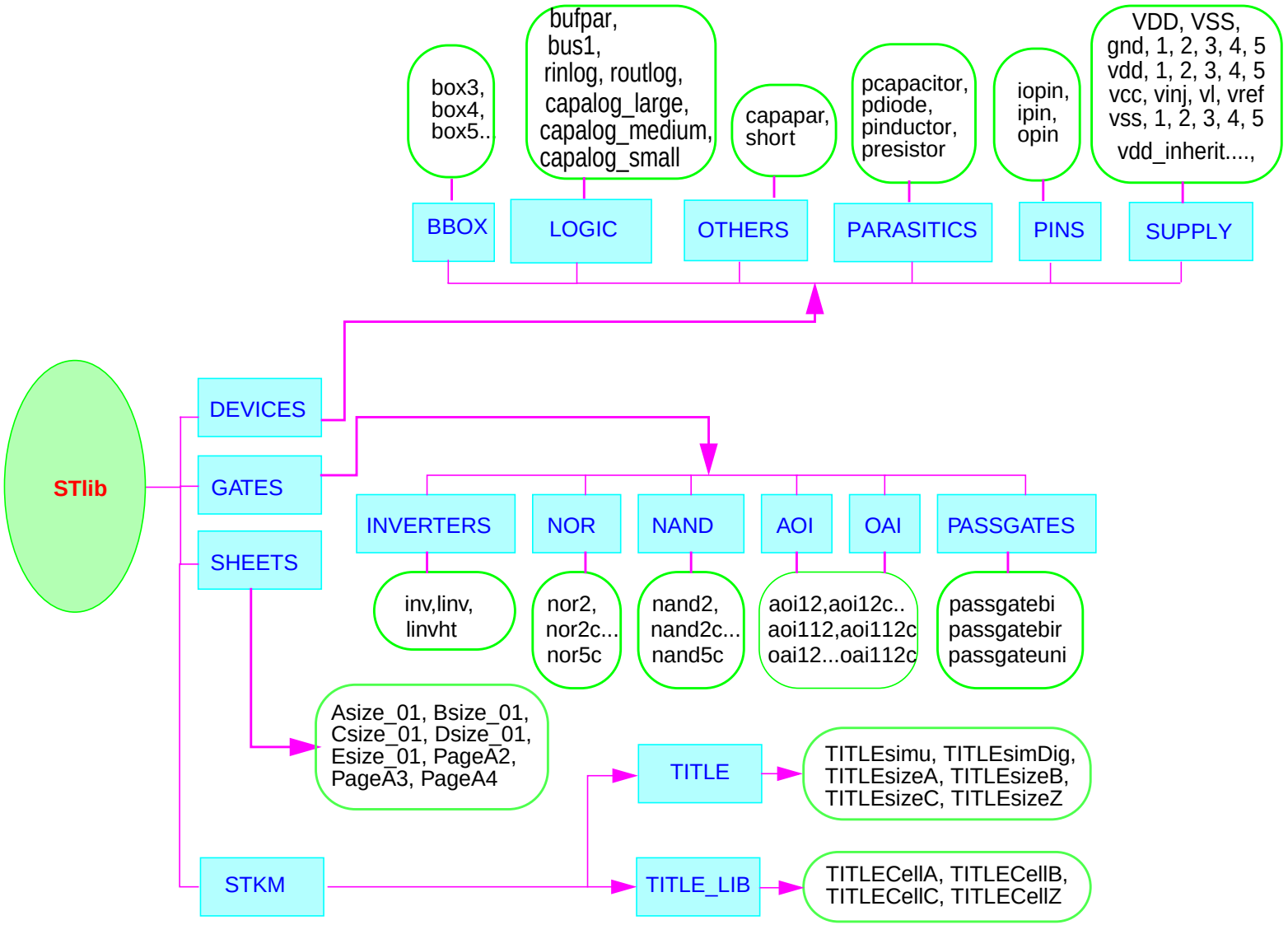
1.1 New features and remarks

This version of STlib supports **OpenAccess** 2.0 Opus database. The product contains two libraries: one for CDBA in "STlib" directory and one for OA in "STlib_oa" directory. The file "DK_STlib.cdslib" defines the STlib library path depending on the environment variable "opusDbtype" which is set-up in "uniopus" product.

It must be used at least with Opus **5.1.41_usr1**.

Please refer to release notes for more information about STlib content updates.

ORGANIZATION OF *STlib*



2 STlib DEVICES Category list

The current version of the library contains the following blocks sorted by categories. All of these are described in the following pages.

The PINS Sub-Category is only present for techno down to 90n. For other technologies, pins from basic are used.

2.1 DEVICES Category

- ❏ BBOX Sub-Category
  - box3
  - box4
  - box5
  - box6
  - box7
  - box8
- ❏ LOGIC Sub-Category
  - bufpar
  - bus1
  - capalog_large
  - capalog_medium
  - capalog_small
  - rinlog
  - routlog
- ❏ OTHERS Sub-Category
  - short
  - capapar
  - capalog
  - noConn
- ❏ PARASITICS Sub-Category
  - pcapacitor
  - pdiode
  - pinductor
  - presistor
- ❏ PINS Sub-Category **(only present for techno down to 90nm)**
  - ipin

- opin
- iopin

❏ SUPPLY Sub-Category

- InfoSupply
- VDD
- VSS
- gnd
- gnd1
- gnd2
- gnd3
- gnd4
- gnd5
- vcc
- vdd
- vdd1
- vdd2
- vdd3
- vdd4
- vdd5
- vdiggnd
- vinj
- vl
- vref
- vss
- vss1
- vss2
- vss3
- vss4
- vss5
- vsubs
- VMINUS_inherit
- VPLUS_inherit
- gnd1_inherit
- gnd_inherit
- gnds_inherit
- gndsub_inherit
- gndi_inherit

- gndis_inherit
- gndo_inherit
- gndos_inherit
- vcc_inherit
- vdd1_inherit
- vdd5_inherit
- vdd_inherit
- vdde1v2_inherit
- vdde1v5_inherit
- vdde1v8_inherit
- vdde2v5_inherit
- vdde3v3_inherit
- vdds_inherit
- vddi_inherit
- vddis_inherit
- vddo_inherit
- vddos_inherit
- vddse1v2_inherit
- vddse1v5_inherit
- vddse1v8_inherit
- vddse2v5_inherit
- vddse3v3_inherit
- vddsub_inherit
- vss_inherit

❏ GLOBAL Sub-Category

- nlpglobals

❏ VERSION Sub-Category

- InfoRelease

3 STlib GATES Category list

The current version of the library contains the following blocks sorted by categories. All of these are described in the following pages.

3.1 GATES Category

- ❏ AOI Sub-Category
  - aoi12
  - aoi12c
  - aoi22
  - aoi22c
  - aoi112
  - aoi112c
- ❏ OAI Sub-Category
  - oai12
  - oai12c
  - oai22
  - oai22c
  - oai112
  - oai112c
- ❏ INVERTERS Sub-Category
  - inv
  - linv
  - linvht
- ❏ NAND Sub-Category
  - nand2
  - nand2c
  - nand3
  - nand3c
  - nand4
  - nand4c
  - nand5
  - nand5c
- ❏ NOR Sub-Category
  - nor2
  - nor2c
  - nor3

- nor3c
- nor4
- nor4c
- nor5
- nor5c

❏ PASSGATES Sub-Category

- passgatebi
- passgatebir
- passgateuni

4 STlib TITLES Category list

The current version of the library contains the following blocks sorted by categories. All of these are described in the following pages.

4.1 SHEETS Category

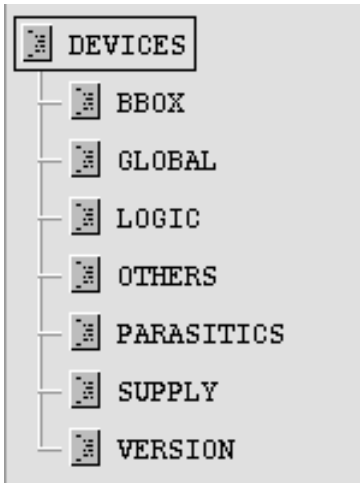
- Asize_01
- Bsize_01
- Csize_01
- Dsize_01
- Esize_01
- PageA2
- PageA3
- PageA4

4.2 STKM Category

- ❏ TITLE Sub-Category
  - TITLEsimu
  - TITLEsimDig
  - TITLEsizeA
  - TITLEsizeB
  - TITLEsizeC
  - TITLEsizeZ
- ❏ TITLE_LIB Sub-Category
  - TITLECellA
  - TITLECellB
  - TITLECellC
  - TITLECellZ

5 DEVICES Category Components

The DEVICES category contains primary devices and is divided in further sub-categories as shown below:



The **BBOX** category contains 3 devices that aim to support the black box parasitic extraction and post-layout simulation.

The **GLOBAL** category contains the *nlpglobals* global cell which is read by netlister when netlisting begins in order to initialize output formatter depending on selected simulator.

The **LOGIC sub-category** contains *capalog* cells which are used by verilog netlister. The *buspar* and *bus1* components are also used by Verilog netlister and simulation. The *rinlog* and *routlog* components are used to prevent CADENCE warnings on floating inputs or outputs.

The **OTHERS sub-category** contains *capapar* kept for compatibility with EDGE, *capalog* cell is a trireg which is also kept for compatibility and which can be used only with ADVanceMS, *noConn* cell which is used to prevent CADENCE warnings on floating inputs or outputs and *short* cell which is used as short-circuit in the LVS netlist.

The **PARASITICS sub-category** contains parasitic devices used:

- by designers to place them in their schematic in order to simulate at the schematic level the parasitic effects.

- by diva drcExtract rules to take into account parasitic devices during layout extraction.

It may be possible that such devices have to be customized for a specific process; in this case the diva tool will automatically search for them in the *devices_symbols* library of the corresponding design kit.

The **PINS sub-category** includes the input, output and the input-output pins and are accessed automatically by using the Create -> Pin command.

This sub-category is present for techno up to 90 nm. For the other techno, pins are taken from basic library.

The **SUPPLY sub-category** contains different supply symbols useful in some various application:

- different supplies for CMOS and BIPOLAR part in case of a bicmos design
- different power supplies for the core and the IO's of a circuit.

The following pages are describing all the cells included in the DEVICES category of the *STlib* library. For each cell the following information is given:

- Description of the Cell.
- Exact Cell name in the library.
- Position of the cell in the library.
- Table of properties prompted on Placement.

The following method was used for detailing the property values:

① There are some cases where the default value of a parameter cannot be easily defined (default value of a resistor, default value of area of a diode, etc.). For such properties the default value has been set to "REQUIRED"; in this case the user **MUST** enter an actual value replacing the "REQUIRED" otherwise the "REQUIRED" will appear in the netlist and the simulator will fail.

② The "storeDefault" column of each table details the management of the corresponding property at the placement time:

- if set to "yes", the property will be attached to the placed instance whatever its value

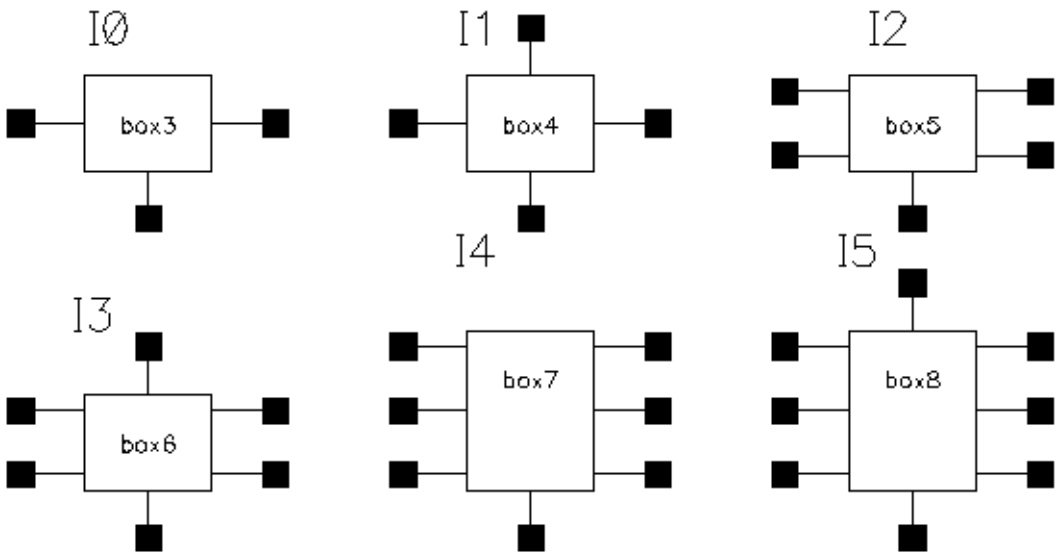
- if set to “no”, the property will be attached to the placed instance only if the user enters a value different from the default value.

5.1 BBOX Sub-category

Description: Black Box device for black box extraction methodology

Cell Names: box3, box4, box5, box6, box7, box8

Category: DEVICES / BBOX



This sub-category contains 6 devices: "box3", "box4", "box5", "box6", "box7" and "box8". The aim of these devices is to support black-box extraction for some cells of the layout for which designers wish to use their own models: for example "box3" is dedicated to 3 pin devices such as micro-strip, "box4" is dedicated to 4 pin devices such as T-strip and "box5" is dedicated to 5 pin devices such as cross-strip, the last port being always used for substrate connection. Refer to design-kit documentation to know if black-box extraction methodology is supported by your design-kit.

The views present for these 6 devices are:

DEVICE	VIEWS
box3, box4 box5, box6 box7, box8	ADVance_MSS, ads, eldo, eldoD, hspiceS, schematic, spectre, symbol, verilogams

The *schematic* view is used by the CDL netlister which has been customized by a netlisting function contained in STlib ".simrc" file The *verilogams* view is used by the AMS-Designer simulator and consists in a Verilog-AMS wrapper of Spectre user model.

Properties prompted on placement

Properties	storeDefault	Default Value
Model Name (macro)	no	""

In order to include user model libraries in the final netlist, refer to "ArtistKit User's Manual" or your simulator "Artist integration" documentation or Cadence "AMS-Designer" documentation.

5.2 LOGIC Sub-category

The views present for the following cells are:

DEVICE	VIEWS
capalog_large, capalog_small, capalog_medium	functional, auLvs, lvs, eldo, spectre..., symbol.
bufpar, rinlog, routlog	verilog, cdl, lvs, eldo, spectre..., symbol
bus1	gate_sch, gate_cdl, verilog, symbol

The *capalog_large*, *capalog_medium*, *capalog_small* components are trireg capacitors used for Verilog netlister and simulation. These components have been added in the *STlib* library in order to replace the previous (and unique) *capalog*. In addition, you have the ability to parameterize the "rise", "fall" and "delay" times. There is one component per value of "charge strength" parameter i.e "large", "small" and "medium".

The *bufpar* and *bus1* components are also used by Verilog netlister and simulation. Note that *bus1* is also netlisted by analog and CDL netlisters. *bufpar* is not netlisted by analog simulators (nlAction=ignore).

The *rinlog* and *routlog* components are never netlisted but are used to prevent CADENCE warnings on floating inputs or outputs when performing a "Check and Save" on a design.

5.2.1 Trireg capacitor with "large" charge strength

Description: Trireg charge storage capacitor used for Verilog netlisting & simulation.

Cell Name: capalog_large

Category: DEVICES / LOGIC

Properties prompted on placement

Properties	storeDefault	Default Value
Charge Strength	yes	large
Rise Time	yes	0
Fall Time	yes	0
Decay Time	yes	30

5.2.2 Trireg capacitor with "medium" charge strength

Description: Trireg charge storage capacitor used for Verilog netlisting & simulation.

Cell Name: capalog_medium

Category: DEVICES / LOGIC

Properties prompted on placement

Properties	storeDefault	Default Value
Charge Strength	yes	medium
Rise Time	yes	0
Fall Time	yes	0
Decay Time	yes	30

5.2.3 Trireg capacitor with "small" charge strength

Description: Trireg charge storage capacitor used for Verilog netlisting & simulation.

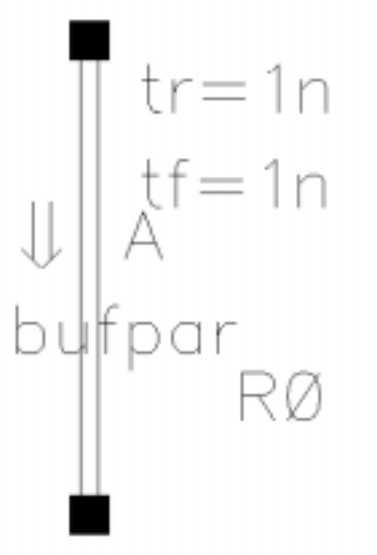
Cell Name: capalog_small

Category: DEVICES / LOGIC

Properties prompted on placement

Properties	storeDefault	Default Value
Charge Strength	yes	small
Rise Time	yes	0
Fall Time	yes	0
Decay Time	yes	30

5.2.4 Bufpar



Description: Log gate used for Verilog netlisting & simulation.

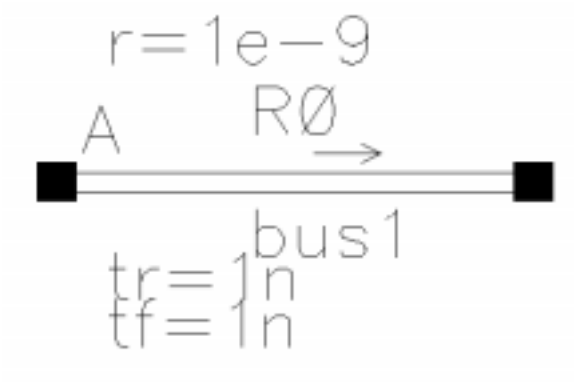
Cell Name: bufpar

Category: DEVICES / LOGIC

Properties prompted on placement

Properties	storeDefault	Default Value
Low_Strength	yes	strong0
High_Strength	yes	strong1
r	yes	1e-9
tr	yes	1n
tf	yes	1n

5.2.5 Bus1



Description: Log gate used for Verilog netlisting & simulation.

Cell Name: bus1

Category: DEVICES / LOGIC

Properties prompted on placement

Properties	storeDefault	Default Value
Low_Strength	yes	strong0
High_Strength	yes	strong1
r	yes	1e-9
tr	yes	1n
tf	yes	1n

5.2.6 Rinlog

Description: Not netlisted

Cell Name: rinlog

Category: DEVICES / LOGIC

There are no properties.

5.2.7 Routlog

Description: Not netlisted

Cell Name: routlog

Category: DEVICES / LOGIC

There are no properties.

5.3 OTHERS Sub-category

The views present for the following devices are:

DEVICE	VIEWS
capapar	auLvs, lvs, functional, eldo, spectre, spectreS, ivpcell, symbol, pathmill, powrmill, timemill
capalog	symbol, vhdlams
noconn	symbol
short	auLvs, lvs, auCdl, cdl, eldo, spectre, spectreS, ivpcell, symbol

The *lvs* and *auLvs* views present for the *capapar* cell have the *nlAction* property set to “ignore” which means that this device will not appear in the *LVS* netlist.

When extracting a layout with parasitics, the *ivpcell* view of *capapar* is placed in the extracted view as an instance.

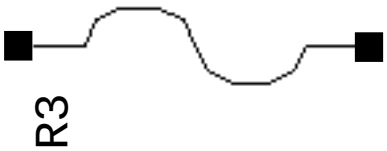
The *capapar* cell is kept in the *STlib* library for compatibility with EDGE. For your new design, it is mandatory to use the components from LOGIC and PARASITICS sub-categories.

The *capapar* cell is kept in the STlib library for compatibility and can be used only with ADVanceMS simulator. Components from LOGIC category should be used.

The *noConn* cell is just a copy of the *noConn* cell of Cadence "basic" library.

The *short* cell can be used to link two components in order to obtain in the *LVS* (or *CDL*) netlist two different terminals names for the same connection.

5.3.1 Short resistor



Description: Element used as short-circuit between two components in order to have two different terminals names in the LVS netlist.

Cell Name: short

Category: DEVICES / OTHERS

Properties prompted on placement

Properties	storeDefault	Default Value
r	yes	1
Device temperature (Eldo)	no	nil
Suppress noise (Eldo)	no	nil
AC resistance (Eldo)	no	nil

For "**ADVance_MSS**" simulator, the short is netlisted as an ideal voltage source. When the simulator "ADVance_MSS" is selected, another parameter named "DC voltage (ADMS)" appears. Its value is by default equal to 0.

5.3.2 Parasitic capacitor

Description: Parasitic capacitor, used for Analog & Digital simulations and also for the extraction of the layout

Cell Name: capapar

Category: DEVICES / OTHERS

Properties prompted on placement

Properties	storeDefault	Default Value
c	yes	REQUIRED
IC	no	nil

5.4 PARASITICS Sub-category

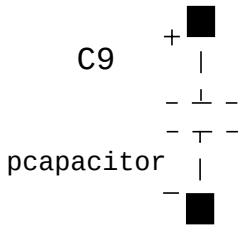
The views present for the following devices are:

DEVICE	VIEWS
pcapacitor, pinductor	cdsSpice, auLvs, lvs, functional, hspice, eldo, spectre, spectreS, spice, ivpcell, symbol
presistor, pdiode	cdsSpice, auLvs, lvs, functional, eldo, spectre, spectreS, ivpcell, symbol

The *lvs* and *auLvs* views present for the above mentioned cells have the *nlAction* property set to “ignore” which means that the device will not appear in the *LVS* netlist.

When extracting a layout with parasitics, the *ivpcell* view of these components is placed in the extracted view as an instance.

5.4.1 Parasitic capacitor



Description: Parasitic capacitor, used for Analog & Digital simulations and also for the extraction of the layout.

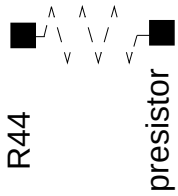
Cell Name: pcapacitor

Category: DEVICES / PARASITICS

Properties prompted on placement

Properties	storeDefault	Default Value
c	yes	REQUIRED
ic	no	nil

5.4.2 Parasitic resistor



Description: Parasitic resistor, used for Analog & Digital simulations and also for the extraction of the layout.

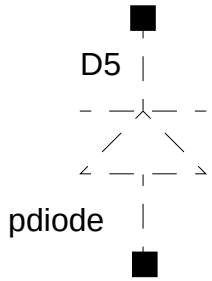
Cell Name: presistor

Category: DEVICES / PARASITICS

Properties prompted on placement

Properties	storeDefault	Default Value
r	yes	REQUIRED
Device temperature (Eldo)	no	nil
Suppress noise (Eldo)	no	nil
AC resistance (Eldo)	no	nil

5.4.3 Parasitic diode



Description: Parasitic diode, used for Analog & Digital simulations and also for the extraction of the layout.

Cell Name: pdiode

Category: DEVICES / PARASITICS

Properties prompted on placement

Properties	storeDefault	Default Value
area	yes	1
peri	no	0u
ox	no	nil
off	no	no
ModelName	yes	IDEAL_DIODE
Device temperature (Eldo)	no	nil
Suppress noise (Eldo)	no	nil

5.4.4 Parasitic inductor

Description: Parasitic inductor, used for Analog & Digital simulations and also for the extraction of the layout.

Cell Name: pinductor

Category: DEVICES / PARASITICS

Properties prompted on placement

Properties	storeDefault	Default Value
l	yes	REQUIRED
IC	no	0

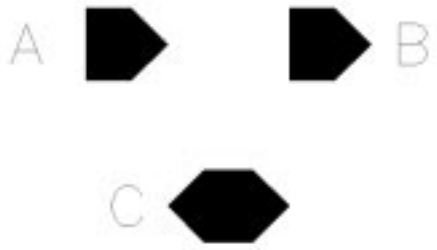
5.5 PINS Sub-category (only for techno down to 90nm)

The views present for the following pins are:

PINS	VIEWS
iopin	symbol symboll symbollOff, symbolr symbolOff
ipin	symbol, symbolrOff
opin	symbol, symbollOff

These pins are automatically placed in the schematic on using the standard Opus command: Create -> Pin.

5.5.1 Pins



Description: Input, Output & inputOutput pins. These are automatically taken from *STlib* and not from *basic* (in the design kit environment)

Cell Name: ipin, opin, iopin

Category: DEVICES / PINS

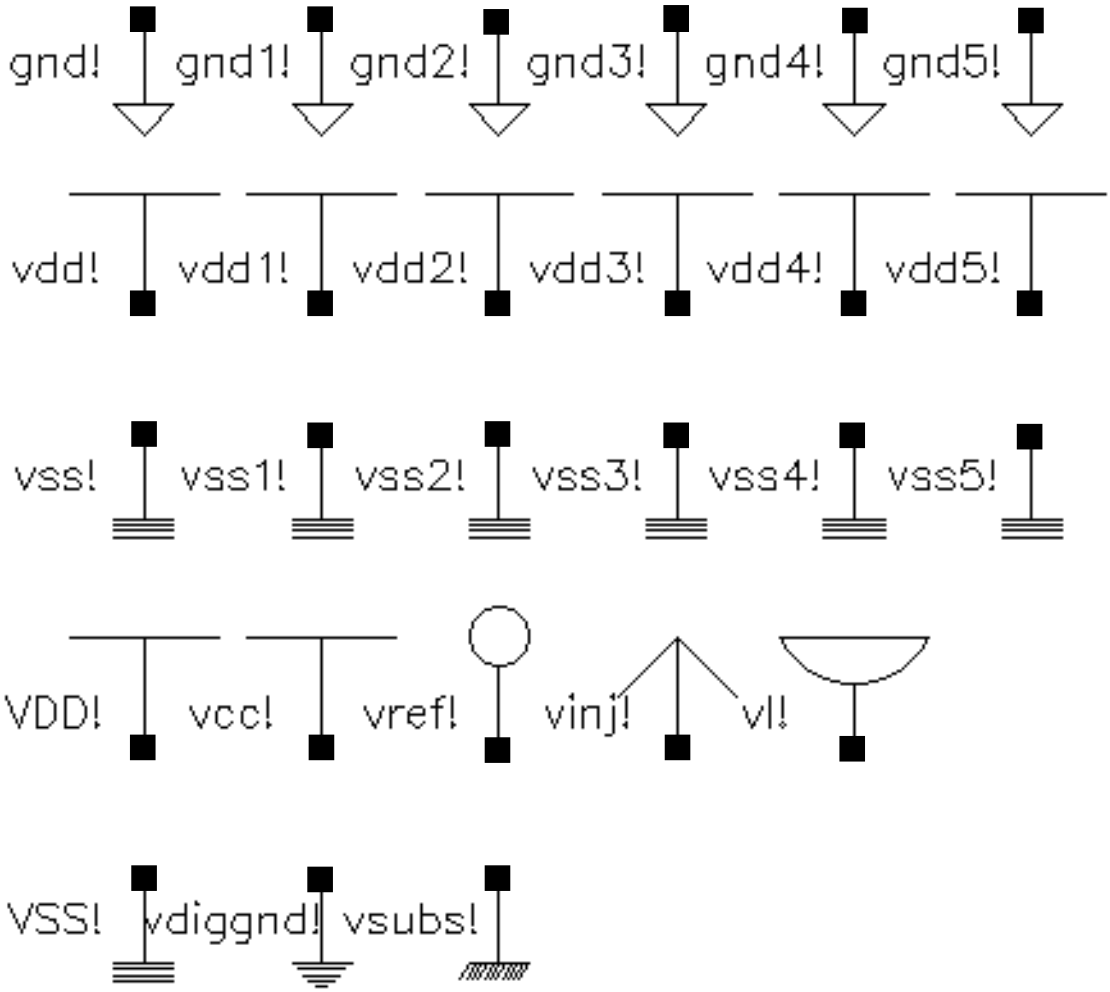
For 90nm, 65nm and lower technologies, the pins of Cadence "basic" library must be used.

5.6 SUPPLY Sub-category

The views present for the following supplies are:

SUPPLIES	VIEWS
All of them	schematic, symbol

5.6.1 Supplies



Description: Different supplies exist, to be used in the schematic.

Cell Names: VDD, VSS, gnd, gnd1, gnd2, gnd3, gnd4, gnd5, vcc, vdd, vdd1, vdd2, vdd3, vdd4, vdd5, vdiggnd, vinj, vl, vref, vss, vss1, vss2, vss3, vss4, vss5, vsubs

VMINUS_inherit
VPLUS_inherit

gnd1_inherit
gnd_inherit
gnds_inherit
gndsub_inherit
gndi_inherit
gndis_inherit
gndo_inherit
gndos_inherit

vcc_inherit
vdd1_inherit
vdd_inherit
vdds_inherit
vddsub_inherit
vss_inherit
vddi_inherit
vddis_inherit
vddo_inherit
vddos_inherit

Category: DEVICES / SUPPLY

No properties are prompted on placement of these symbols.

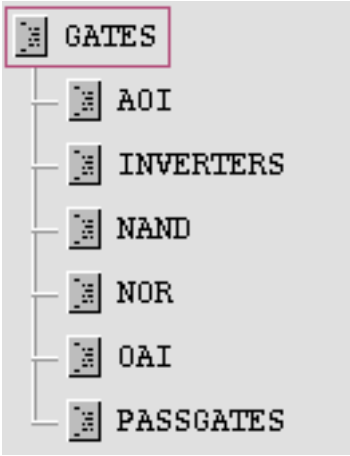
For **multi-power supply management**, inherited power and ground supply devices have been added: *gnd_inherit*, *vss_inherit*, *vdd_inherit*, *vcc_inherit*, *gnds_inherit*, *vdds_inherit*, *vdd1_inherit*... These devices have been copied from CADENCE "basic" library. They are useful to define easily inherited power supplies without setting manually *net expressions* on wires. Refer for example to Opus 4.4.3.100 documentation "Composer: Design Entry User Guide, Connectivity and Naming Conventions, Inherited Connections".

For **multi-power supply management**, it is forbidden to use the other power and ground supply devices. Because of, in the cdl netlist, global pins gnd and

vdd appear even if their values are specified by netsets.

6 GATES Category Components

The GATES category contains logical gates. It contains further sub-categories as shown below:



All gates are parametrized with process-dependent sizes. For example. if the user is working in the HCMOS7 technology then his/her environment is customized by the Design Kit environment so that only the HCMOS7 default sizes are used. Thus the user is able to obtain a part of the *STlib* library customized for HCMOS7 technology.

❍ **CDL Netlisting with auCdl (Artist) netlister**

In the following, the actions to be performed to obtain CDL netlists, from a schematic including gates (from *STlib*), either at gate or at transistor level are described.

➥To obtain the *CDL* netlist at the gate level:

By default the netlist obtained will be at the transistor level. **To obtain the cdl netlist at the gate level, two additional lines detailing the cdlSimViewList and cdlSimStopList as:**

cdlSimViewList = ‘(“auCdl” “auGate.cdl” “auGate_sch” “schematic” “symbol”)
cdlSimStopList = ‘(“auCdl” “auGate_cdl”)

need to be added by the user in the .simrc file present in the home directory. The above lines should be added <u>after</u> the following line in the

.simrc file.

```
UkerSimrc = t
(loadi (strcat (getShellEnvVar "UNICAD_KERNEL_ROOT") "/skill/
Uker.ile"))
UkerSimrc = nil
```

➥To obtain the *CDL* netlist at the transistor level:

By default, on running the netlister in background or foreground, the cdl netlist will be obtained at the transistor level and the user needs not modify anything in the .simrc file

If the user has already customized it to obtain netlists at gate level as described in the previous section, then of course the two lines specifying *cdlSimViewList* and *cdlSimStopList* have to be commented to recover the “default” behavior.

❍ **Verilog-XL Netlisting**

The following table illustrates, which cells have verilog view and which have functional view.

CELLS	VIEWS
AOI Category	functional
OAI Category	functional
NAND Category	verilog
NOR Category	verilog
inv, passgateuni	verilog
linv, linvht, passgatebi, passgatebir	No, verilog or functional views
capapar, rpolar	functional

➥Verilog View

The gates containing the verilog view as detailed above are mapped on the verilog primitives. The properties defined in the verilog views are for

instance dedicated to nand gates:

hnlVerilogFormatInst = hnlVerilogPrintLogGate(“nand”)
High_Strength = strong1
Low_Strength = strong0
tr = 1n
tf = 1n

The last four properties are defined in all the verilog views. During the placement of these devices the user is prompted for the values of these properties. These properties will be attached to the instance only if their value is different from the default, otherwise the verilog netlister will use the default values specified in the verilog view itself.

➥Functional View

This functional view is used when no verilog primitive exists for the cell. The file which was used by “Verilog In” to create the functional views is given below for reference only:

```
//////////////////////////////////////////////////
///// aoi, oai series //////////////////////
//////////////////////////////////////////////////
// “aoi112” description

module aoi112 (Y,A,B,C,D);
output Y;
input A,B,C,D;

 and M1(I,D,C);
 nor (strong0,strong1) #(1,1) M2(Y,I,B,A);

endmodule
//////////////////////////////////////////////////
// “aoi112c” description

module aoi112c (Y,A,B,C,D);
output Y;
input A,B,C,D;

 and M1(I,A,B);
 nor (strong0,strong1) #(1,1) M2(Y,I,C,D);
```

```
endmodule
////////////////////////////////////////////////////
// “aoi12” description

module aoi12 (Y,A,B,C);
output Y;
input A,B,C;
  and M1(I,C,B);
 nor (strong0,strong1) #(1,1) M2(Y,I,A);

endmodule
////////////////////////////////////////////////////
// “aoi12c” description

module aoi12c (Y,A,B,C);
output Y;
input A,B,C;

 and M1(I,A,B);
 nor (strong0,strong1) #(1,1) M2(Y,I,C);

endmodule
////////////////////////////////////////////////////
// “aoi22” description

module aoi22 (Y,A,B,C,D);
output Y;
input A,B,C,D;

 and M1(K,C,D);
 and M2(I,A,B);
 nor (strong0,strong1) #(1,1) M3(Y,I,K);

endmodule
////////////////////////////////////////////////////
// “aoi22c” description

module aoi22c (Y,A,B,C,D);
output Y;
input A,B,C,D;
```

```
 and M1(K,C,D);
 and M2(I,A,B);
 nor (strong0,strong1) #(1,1) M3(Y,I,K);
 endmodule

////////////////////////////////////////////////////
// “oai112” description

module oai112 (Y,A,B,C,D);
output Y;
input A,B,C,D;

 or M1(I,D,C);
 nand (strong0,strong1) #(1,1) M2(Y,I,B,A);

endmodule
////////////////////////////////////////////////////
// “oai112c” description

module oai112c (Y,A,B,C,D);
output Y;
input A,B,C,D;

 or M1(I,A,B);
 nand (strong0,strong1) #(1,1) M2(Y,I,C,D);

endmodule
////////////////////////////////////////////////////
// “oai12” description

module oai12 (Y,A,B,C);
output Y;
input A,B,C;

 or M1(I,C,B);
 nand (strong0,strong1) #(1,1) M2(Y,I,A);

endmodule
////////////////////////////////////////////////////
// “oai12c” description
```

```
 module oai12c (Y,A,B,C);
output Y;
input A,B,C;

 or M1(I,A,B);
 nand (strong0,strong1) #(1,1) M2(Y,I,C);

endmodule
////////////////////////////////////////////////////
// “oai22” description

module oai22 (Y,A,B,C,D);
output Y;
input A,B,C,D;

 or M1(K,C,D);
 or M2(I,A,B);
 nand (strong0,strong1) #(1,1) M3(Y,I,K);

endmodule
////////////////////////////////////////////////////
// “oai22c” description

module oai22c (Y,A,B,C,D);
output Y;
input A,B,C,D;

 or M1(K,C,D);
 or M2(I,A,B);
 nand (strong0,strong1) #(1,1) M3(Y,I,K);

endmodule
////////////////////////////////////////////////////
```

Those components which do not have either the verilog or the functional view have the *auGate_sch* view and the netlister descends their hierarchy to the transistor level i.e. at *nmos* and *pmos* level and looks at their verilog views.

❍ **Support of Cadence AMS Designer simulator**

Cadence provides a new mixed-signal simulator named **AMS Designer** which is based on "spectre" for the analog part and "ncsim" for the digital one. However, "verilog" views can not be netlisted by the new "amsdirect" netlister but only "functional" views. This limitation is documented in "Cadence AMS Environment Known Problems and Solutions" document and referenced as PCR 301515: hnlVerilogPrintLogGate issue. Therefore, functional" views have been added for the following cells:

- bufpar
- bus1
- inv
- nand2
- nand2c
- nand3
- nand3c
- nand4
- nand4c
- nand5
- nand5c
- nor2
- nor2c
- nor3
- nor3c
- nor4
- nor4c
- nor5
- nor5c
- passgateuni: note that the functional description does not take into account the "str" property which was used for Verilog-XL netlister. When str equals "R", Verilog-XL netlists the passgate as "rcmos" instead of "cmos". The functional description instances the "cmos" Verilog primitive.

❍ **Support of Mentor Graphics ADVanceMS simulator**

In order to use functional Verilog code, "**adms**" views must be selected. They have been created by compiling the Verilog code of the functional views and by importing Verilog module definition into Opus. Switch from "auGate_sch" view to "adms" view either by using the Hierarchy Editor or by changing the order of the views in the "Switch View List" of "ADVance_MSS" (or "ADVance_MS") simulator in the "Setup -> Environment" form.

The Verilog files have been compiled using "<STlib>/adms/compileAdmsSTlib" C-shell script with one specific ADVanceMS version and one platform only corresponding to the development and test environment. The ADVanceMS compiled library is stored in <STlib>/adms/$admsver/<platform>/STlib" directory. In order to use that compiled library, enter the following *Unix* command in the directory where Cadence Opus Design Framework is launched:

```
vamap STlib $DK_STlibROOT/adms/$admsver/Linux/STlib.
```

where Linux is the name of the operating-system. In order to run another version of ADVanceMS on another platform, STlib library must be re-compiled using the Csh script "<STlib>/adms/compileAdmsSTlib" in the following way:

```
compileAdmsSTlib <STlib-Root-Directory>
<Compilation-Directory>
```

A default ADVanceMS converter file is provided in "adms" directory: $DK_STlibROOT/adms/converters.conv file. This file gives a list of ELDO converters that will be used as interface elements on the "mixed-signal" nets. The content of this file is:

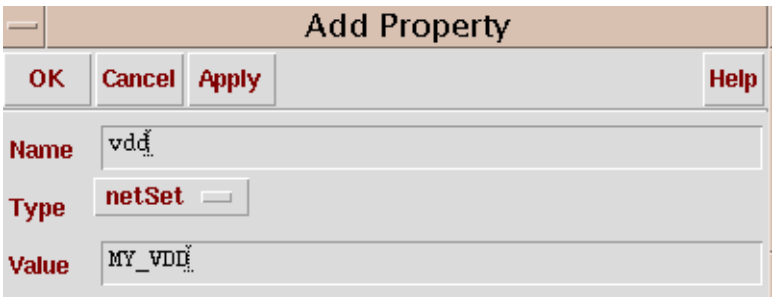
```
GROUP_NAME=default
CONVLIST
A2D MODE=bit VTH=0.5
D2A MODE=bit VHI=1.0 VLO=0.0
A2D MODE=std_logic STR=STRONG VTH1=0.4 VTH2=0.6
D2A MODE=std_logic VHI=1.0 VLO=0.0 TRISE=1P TFALL=1P
ENDCONV
```

Of course, values will have to be changed depending on the power supplies. This file is provided as an example.

By default, the ADVanceMS SKILL variable named "**geaAdmsConverterFiles**" is set to this file.

❍ **Inherited Connections of auGate_sch views**

auGate_sch views are transistor level schematic views. Positive and negative power supplies are defined by inherited connections method using two properties: "vdd" for positive power supply and "gnd" for negative one. By default, "vdd" is corresponding to net "vdd!" and "gnd" to net "gnd!". In order to define power suppplies instance per instance, add "netSet" properties on gate instances ("vdd" and/or "gnd") and define the new nets:



The following pages describe all the cells included in the GATES category of the *STlib* library. For each cell the following information is given:

- Description of the Cell.
- Exact Cell name in the library.
- Position of the cell in the library.
- Properties prompted on Placement.

The "storeDefault" column of each table details the management of the corresponding property at the placement time:

- if set to "yes", the property will be attached to the placed instance whatever its value

- if set to "no", the property will be attached to the placed instance only if the user enters a value different from the default value.

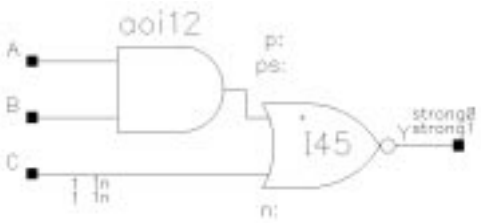
All the cells gets the "**mfactor**" factor as a property which allows to define several gates in parallel. The "**mfactor**" factor is inherited by all mosfet transistors in the gate schematic description (*auGate_sch* view).

6.1 AOI Sub-category

The views present for the following gates in this category are:

GATES	VIEWS
aoi12, aoi12c, aoi22, aoi22c, aoi112, aoi112c	auGate_cd, functional, auGate_sch, symbol

6.1.1 aoi12 gate



Description: CMOS static and/nor gate used for Analog & Digital simulations. All n transistors have the same sizing. All p transistors have the same sizing.

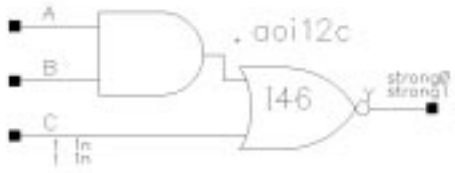
Cell Name: aoi12

Category: GATES / AOI

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wps	yes	Parameterized
lps	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.1.2 aoi12 complex gate



Description: Complex CMOS static and/nor gate used for Analog & Digital simulations. Each n transistor can be sized. Each p transistor can be sized.

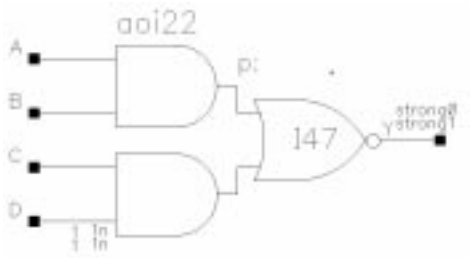
Cell Name: aoi12c

Category: GATES / AOI

Properties prompted on placement

Properties	storeDefault	Default Value
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.1.3 aoi22 gate



Description: CMOS static and/nor gate used for Analog & Digital simulations. All n transistors have the same sizing. All p transistors have the same sizing.

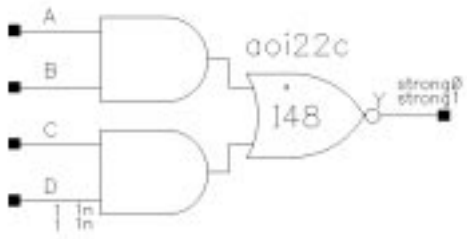
Cell Name: aoi22

Category: GATES / AOI

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.1.4 aoi22 complex gate



Description: Complex CMOS static and/nor gate used for Analog & Digital simulations. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: aoi22c

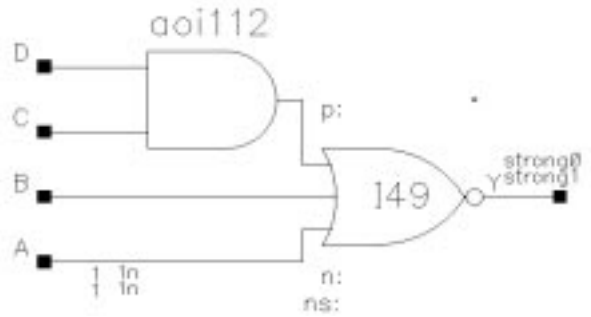
Category: GATES / AOI

Properties prompted on placement

Properties	storeDefault	Default Value
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
lpD	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized
wnD	yes	Parameterized

Properties	storeDefault	Default Value
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.1.5 aoi112 gate



Description: CMOS static and/nor gate used for Analog & Digital simulations. All n transistors have the same sizing. All p transistors have the same sizing.

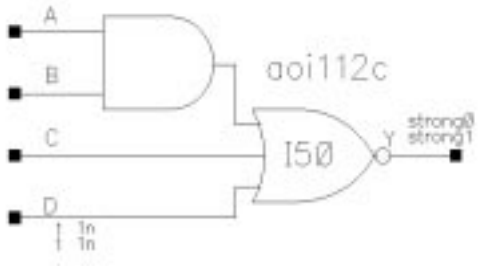
Cell Name: aoi112

Category: GATES / AOI

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wns	yes	Parameterized
lns	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.1.6 aoi112 complex gate



Description: Complex CMOS static and/nor gate used for Analog & Digital simulations. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: aoi112c

Category: GATES / AOI

Properties prompted on placement

Properties	storeDefault	Default Value
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
lpD	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized

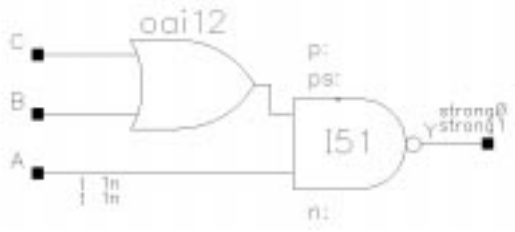
Properties	storeDefault	Default Value
wnD	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.2 OAI Sub-category

The views present for the following gates in this category are:

GATES	VIEWS
oai12, oai12c, oai22, oai22c, oai112, oai112c	auGate_cdl, functional, auGate_sch, symbol

6.2.1 oai12 gate



Description: CMOS static and/nor gate used for Analog & Digital simulations. All n transistors have the same sizing. All p transistors have the same sizing.

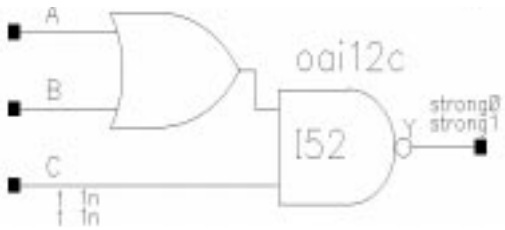
Cell Name: oai12

Category: GATES / OAI

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wps	yes	Parameterized
lps	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.2.2 oai12 complex gate



Description: Complex CMOS static and/nor gate used for Analog & Digital simulations. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: oai12c

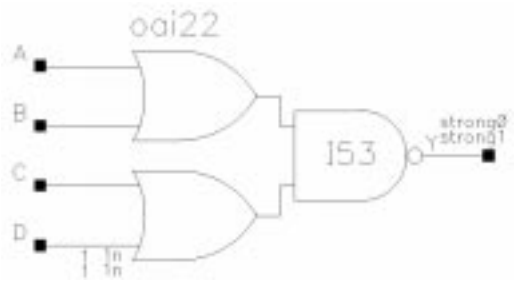
Category: GATES / OAI

Properties prompted on placement

Properties	storeDefault	Default Value
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized

Properties	storeDefault	Default Value
mfactor	yes	1

6.2.3 oai22 gate



Description: CMOS static and/nor gate used for Analog & Digital simulations. All n transistors have the same sizing. All p transistors have the same sizing.

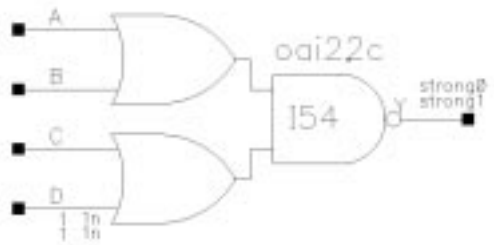
Cell Name: oai22

Category: GATES / OAI

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.2.4 oai22 complex gate



Description: Complex CMOS static and/nor gate used for Analog & Digital simulations. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: oai22c

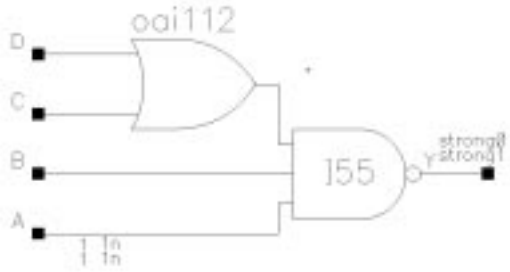
Category: GATES / OAI

Properties prompted on placement

Properties	storeDefault	Default Value
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
lpD	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized
wnD	yes	Parameterized

Properties	storeDefault	Default Value
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.2.5 oai112 gate



Description: CMOS static and/nor gate used for Analog & Digital simulations. All n transistors have the same sizing. All p transistors have the same sizing.

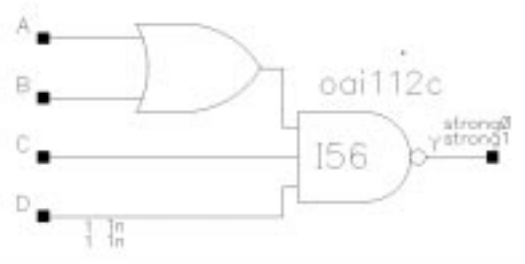
Cell Name: oai112

Category: GATES / OAI

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.2.6 oai112 complex gate



Description: Complex CMOS static and/nor gate used for Analog & Digital simulations. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: oai112c

Category: GATES / OAI

Properties prompted on placement

Properties	storeDefault	Default Value
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
lpD	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized
wnD	yes	Parameterized

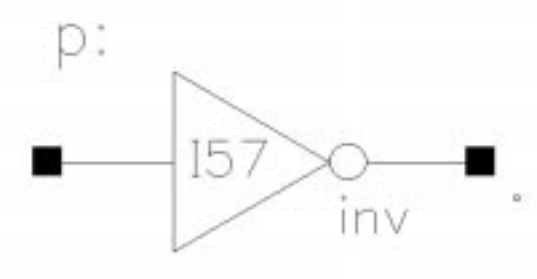
Properties	storeDefault	Default Value
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
tr	no	Parameterized
tf	no	Parameterized
mfactor	yes	1

6.3 INVERTERS Sub-category

The views present for the following gates in this category are:

GATES	VIEWS
inv	auGate_cdl, verilog, auGate_sch,,symbol
linv, linvht	auGate_cdll, auGate_sch, symbol

6.3.1 Inverter



Description: CMOS static Inverter, used for Analog & Digital simulations. The macro model of this component is also present.

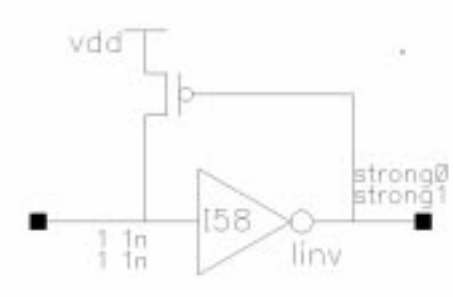
Cell Name: inv

Category: GATES / INVERTERS

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.3.2 Latched inverter



Description: CMOS static "latched" inverter, used for Analog & Digital simulations.

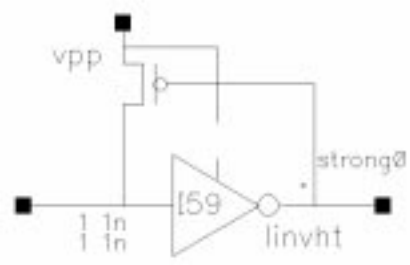
Cell Name: linv

Category: GATES / INVERTERS

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
wpr	yes	Parameterized
lpr	yes	Parameterized
mfactor	yes	1

6.3.3 Latched inverter with VPP supply



Description: CMOS static "latched" inverter gate with VPP supply, used for Analog & Digital simulations.

Cell Name: linvht

Category: GATES / INVERTERS

Properties prompted on placement

Properties	storeDefault	Default Value
Voltage for logic 1	no	Parameterized
Voltage for logic 0	no	0 V
Threshold input voltage	no	Parameterized
Threshold input voltage (out fall)	no	2.5 V
Threshold input voltage (out rise)	no	2.5 V
Transit time	no	Parameterized
Time to reach VTLO (output falling)	no	1n S
Time to reach VTLI (output rising)	no	1n S
Input capacitance	no	Parameterized
tr	no	Parameterized
tf	no	Parameterized

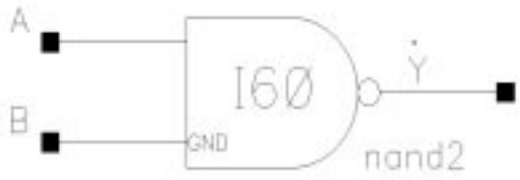
Properties	storeDefault	Default Value
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
wpr	yes	Parameterized
lpr	yes	Parameterized
mfactor	yes	1

6.4 NAND Sub-category

The views present for the following gates in this category are:

GATES	VIEWS
nand2, nand3	auGate_cdl, verilog, auGate_sch,symbol
nand2c, nand3c, nand4, nand4c, nand5, nand5c	auGate_cdl, verilog, auGate_sch, symbol

6.4.1 2 input nand gate



Description: 2 input CMOS static nand gate. The macro model of this component is also present. All n transistors have the same sizing. All p transistors have the same sizing.

Cell Name: nand2

Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.4.2 2 input complex nand gate

Description: 2 input complex CMOS static nand gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nand2c

Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
mfactor	yes	1

6.4.3 3 input nand gate

Description: 3 input CMOS static nand gate. The macro model of this component is also present. All n transistors have the same sizing. All p transistors have the same sizing.

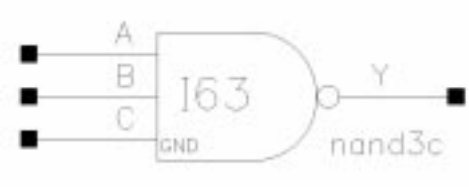
Cell Name: nand3

Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.4.4 3 input complex nand gate



Description: 3 input complex CMOS static nand gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nand3c

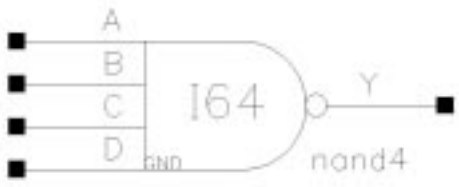
Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized

Properties	storeDefault	Default Value
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
mfactor	yes	1

6.4.5 4 input nand gate



Description: 4 input CMOS static nand gate. All n transistors have the same sizing. All p transistors have the same sizing.

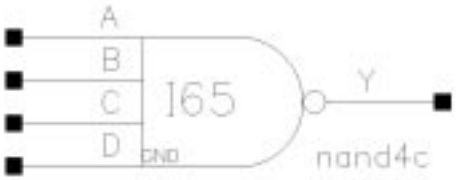
Cell Name: nand4

Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.4.6 4 input complex nand gate



Description: 4 input complex CMOS static nand gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nand4c

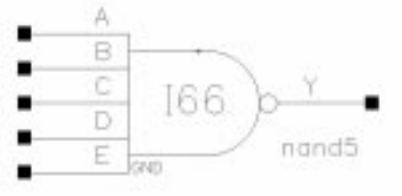
Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
lpD	yes	Parameterized
wnA	yes	Parameterized

Properties	storeDefault	Default Value
wnB	yes	Parameterized
wnC	yes	Parameterized
wnD	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
mfactor	yes	1

6.4.7 5 input nand gate



Description: 5 input CMOS static nand gate. All n transistors have the same sizing. All p transistors have the same sizing.

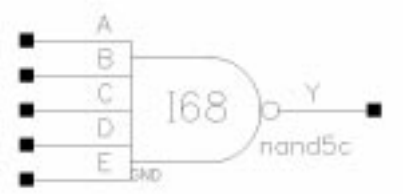
Cell Name: nand5

Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.4.8 5 input complex nand gate



Description: 5 input complex CMOS static nand gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nand5c

Category: GATES / NAND

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
wpE	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
lpD	yes	Parameterized

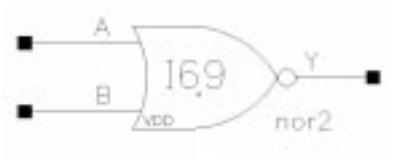
Properties	storeDefault	Default Value
lpE	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized
wnD	yes	Parameterized
wnE	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
lnE	yes	Parameterized
mfactor	yes	1

6.5 NOR Sub-category

The views present for the following gates in this category are:

GATES	VIEWS
nor2, nor3	auGate_cdl,verilog, auGate_sch, symbol
nor2c, nor3c, nor4, nor4c, nor5, nor5c	auGate_cdl, verilog, auGate_sch, symbol

6.5.1 2 input nor gate



Description: 2 input CMOS static nor gate. The macro model of this component is also present. All n transistors have the same sizing. All p transistors have the same sizing.

Cell Name: nor2

Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.5.2 2 input complex nor gate

Description: 2 input complex CMOS static nor gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nor2c

Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
mfactor	yes	1

6.5.3 3 input nor gate

Description: 3 input CMOS static nor gate. The macro model of this component is also present. All n transistors have the same sizing. All p transistors have the same sizing.

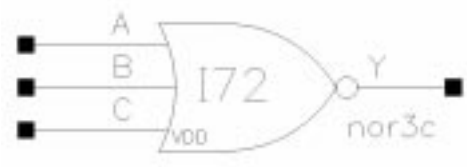
Cell Name: nor3

Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.5.4 3 input complex nor gate



Description: 3 input complex CMOS static nor gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nor3c

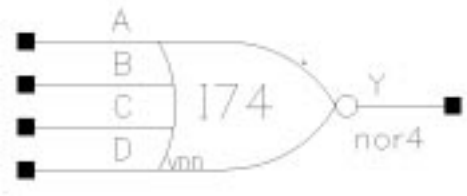
Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized

Properties	storeDefault	Default Value
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
mfactor	yes	1

6.5.5 4 input nor gate



Description: 4 input CMOS static nor gate. All n transistors have the same sizing. All p transistors have the same sizing.

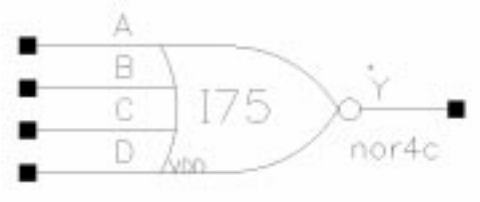
Cell Name: nor4

Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.5.6 4 input complex nor gate



Description: 4 input complex CMOS static nor gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nor4c

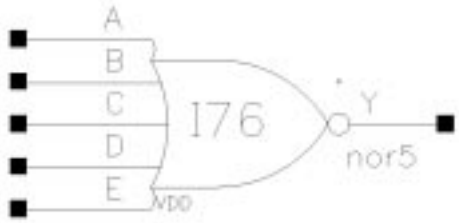
Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized
lpD	yes	Parameterized
wnA	yes	Parameterized

Properties	storeDefault	Default Value
wnB	yes	Parameterized
wnC	yes	Parameterized
wnD	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
mfactor	yes	1

6.5.7 5 input nor gate



Description: 5 input CMOS static nor gate. All n transistors have the same sizing. All p transistors have the same sizing.

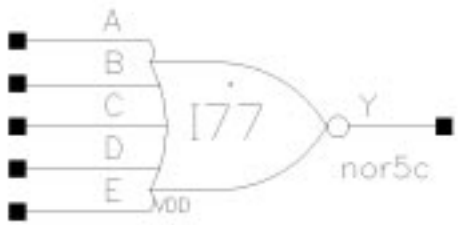
Cell Name: nor5

Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.5.8 5 input complex nor gate



Description: 5 input complex CMOS static nor gate. Each n transistor can be sized. Each p transistor can be sized.

Cell Name: nor5c

Category: GATES / NOR

Properties prompted on placement

Properties	storeDefault	Default Value
tr	no	Parameterized
tf	no	Parameterized
Low_Strength	no	strong0
High_Strength	no	strong1
wpA	yes	Parameterized
wpB	yes	Parameterized
wpC	yes	Parameterized
wpD	yes	Parameterized
wpE	yes	Parameterized
lpA	yes	Parameterized
lpB	yes	Parameterized
lpC	yes	Parameterized

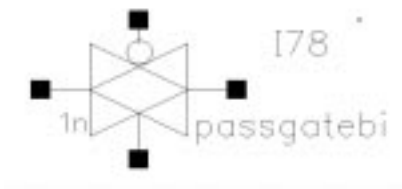
Properties	storeDefault	Default Value
lpD	yes	Parameterized
lpE	yes	Parameterized
wnA	yes	Parameterized
wnB	yes	Parameterized
wnC	yes	Parameterized
wnD	yes	Parameterized
wnE	yes	Parameterized
lnA	yes	Parameterized
lnB	yes	Parameterized
lnC	yes	Parameterized
lnD	yes	Parameterized
lnE	yes	Parameterized
mfactor	yes	1

6.6 PASSGATES Sub-category

The views present for the following gates in this category are:

GATES	VIEWS
passgateuni	auGate_cdl, auGate_sch, verilog, symbol
passgatebi, passgatebir	auGate_cdl, auGate.sch, symbol

6.6.1 Bi-directional passgate



Description: Bi-directional transfer gate.

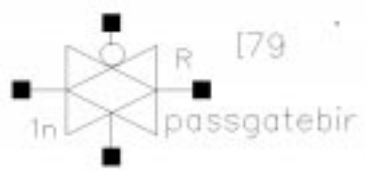
Cell Name: passgatebi

Category: GATES / PASSGATES

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.6.2 Bi-directional resistive passgate



Description: Bi-directional resistive transfer gate.

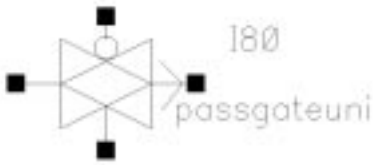
Cell Name: passgatebir

Category: GATES / PASSGATES

Properties prompted on placement

Properties	storeDefault	Default Value
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

6.6.3 Uni-directional passgate



Description: Uni-directional transfer gate.

Cell Name: passgateuni

Category: GATES / PASSGATES

Properties prompted on placement

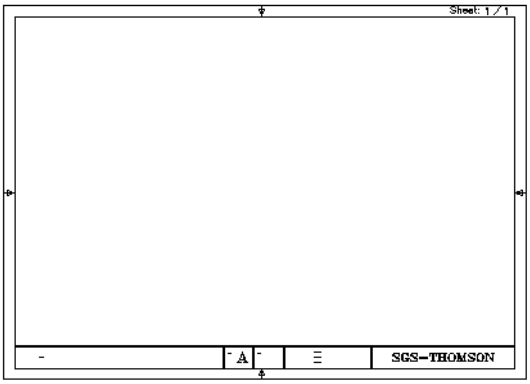
Properties	storeDefault	Default Value
str	no	nil
tr	no	Parameterized
tf	no	Parameterized
wp	yes	Parameterized
lp	yes	Parameterized
wn	yes	Parameterized
ln	yes	Parameterized
mfactor	yes	1

7 SHEETS and STKM Categories Components

The following pages describe all the cells included in the SHEETS and STKM categories of the *STlib* library. For each cell the following information is given:

- Description of the Cell.
- Exact Cell name in the library.
- Position of the cell in the library.
- Properties prompted on Placement.

7.1 Sheets



Description: Eight different sizes of sheets are present in the *STlib* and they are customized for the company needs. An example of the looks is given above.

Cell Names: `Asize_01, Bsize_01, Csize_01, Dsize_01, Esize_01, PageA2, PageA3, PageA4`

Category: SHEETS

Properties prompted on placement

Properties	storeDefault	Default Value
title	no	nil
author	no	nil
rev	no	nil
date	no	nil
sheetNumber	no	1
maxSheets	no	1
confidentiality	no	yes

Cell Names: `PageA2, PageA3, PageA4`

Category: SHEETS

Properties prompted on placement

Properties	storeDefault	Default Value
Cell Number in Hierarchy	yes	1
Function (1st line)	yes	nil
Function (2nd line)	yes	nil
Author	yes	nil
Revision number	yes	1
Date is automatically set by	yes	nil
Sheet number	yes	1
Max sheets number	yes	1
Project Name	yes	nil
Format	yes	A2
confidentiality	no	nil

7.2 STKM Category Components

The components contained in this category are sheet titles that must be used only when the ADS design environment of ANACA has been loaded otherwise some properties will not be set.

Refer to the ADS documentation from ANACA for more information on these components.