ProASIC®3/E I/O Usage Guide

Introduction

Users designing I/O solutions are faced with a number of implementation decisions and configuration choices that can directly impact the efficiency and effectiveness of their final design. The ProASIC3 and ProASIC3E (ProASIC3/E) devices feature a flexible I/O structure, supporting a wide variety of voltages and I/O standards that enable users to meet the growing challenges of their many diverse applications. The Actel Libero® Integrated Design Environment (IDE) provides an easy way to implement I/O that will result in robust I/O design. This application note describes all available features of ProASIC3/E I/Os as well as how to take full advantage of these with the help of Actel Libero IDE.

Advanced I/O Structure

ProASIC3/E I/Os are divided into multiple technology banks. The number of banks is device-dependent. The ProASIC3E family has eight banks (two per side), and the ProASIC3 family has two to four banks. Each bank has its own V_{CCI} power supply pin. Multiple I/O standards can co-exist within a single I/O bank.

In ProASIC3E devices, each I/O bank is subdivided into V_{REF} minibanks. These are used by voltage-referenced I/Os. V_{REF} minibanks contain 8 to 18 I/Os. All I/Os in a given minibank share a common V_{REF} line (only one V_{REF} pin is needed per V_{REF} minibank). Therefore, if an I/O in a V_{REF} minibank is configured as a V_{REF} pin, the remaining I/Os in that minibank will be able to use the voltage assigned to that pin. If the location of the V_{REF} pin is selected manually in the software, the user must satisfy V_{REF} rules (refer to the "Appendix" on page 28 for V_{REF} rules). If the user does not pick the V_{REF} pin manually, the software automatically assigns the V_{REF} pin.

Figure 1 is a snapshot of a section of the I/O ring, showing the basic elements of an I/O tile, as viewed from the Designer place-and-route tool's MultiView Navigator (MVN).

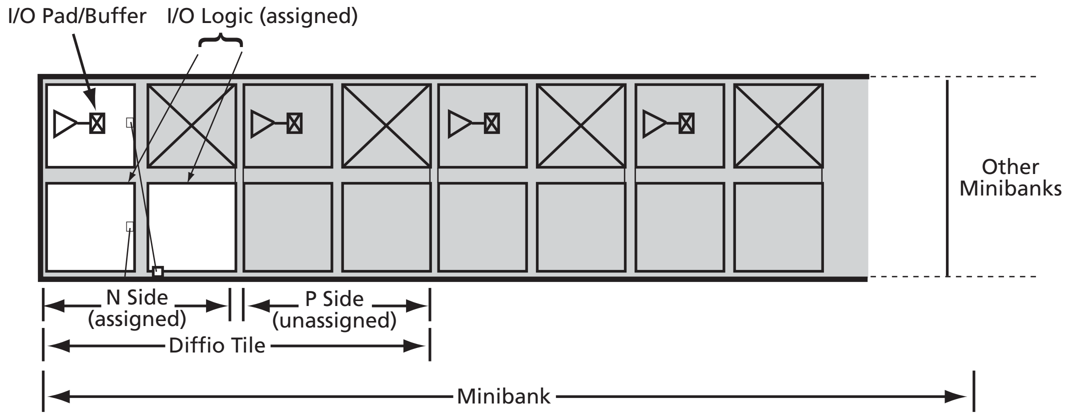


Figure 1 • **Snapshot of an I/O Tile**

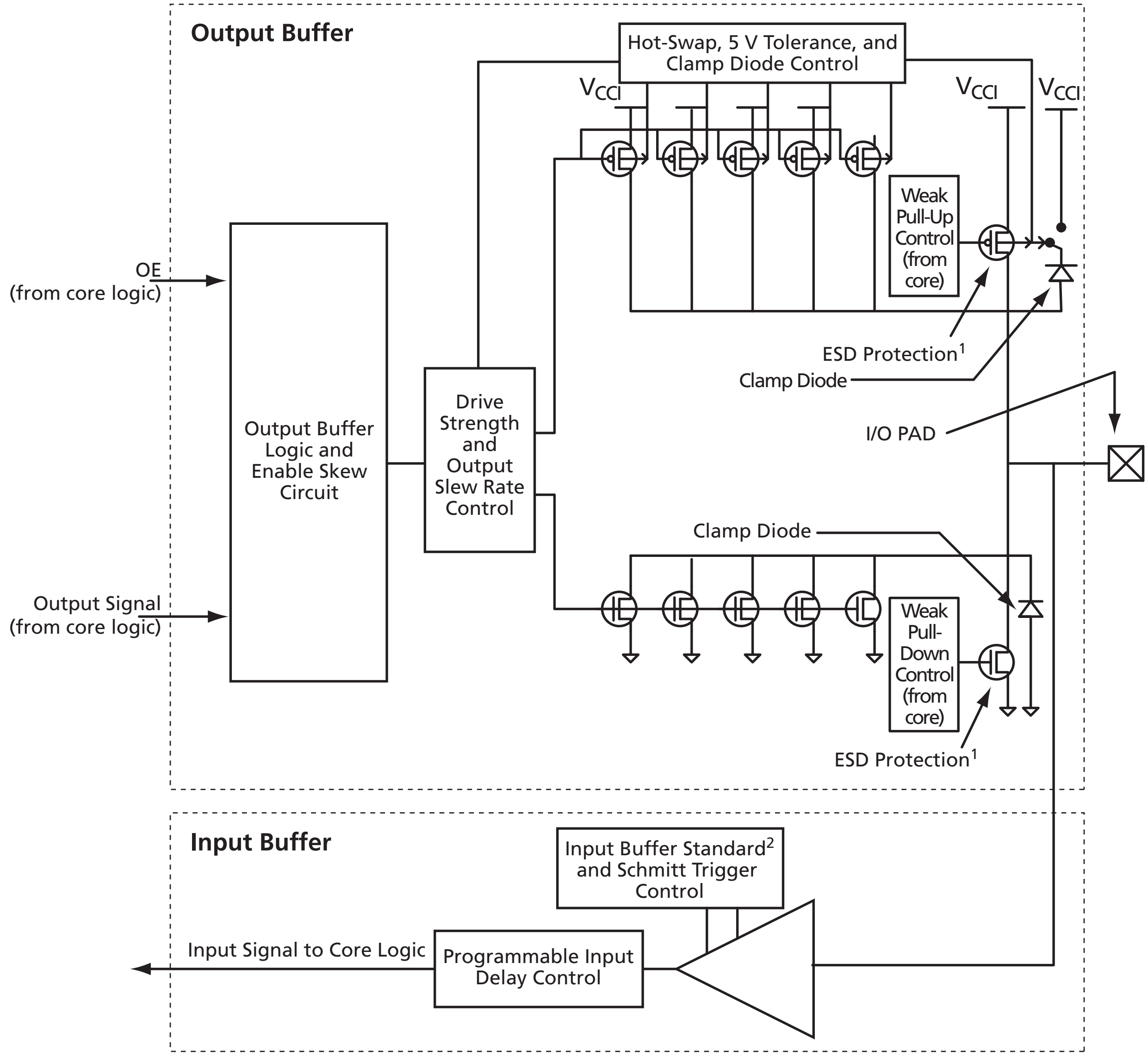
ProASIC3/E I/Os are implemented using two tile types: I/O and differential I/O (diffio).

The diffio tile is built up using two I/O tiles, which form an I/O pair (P side and N side). These I/O pairs are used according to differential I/O standards. Both the P and N sides of the diffio tile include an I/O buffer and two I/O logic blocks (auxiliary and main logic).

Every minibank (ProASIC3E only) is built up from multiple diffio tiles. The number of the minibank depends on the different ProASIC3E dies. Refer to the "Pro I/Os" section of the *ProASIC3E datasheet* for an illustration of the minibank structure.

Figure 2 shows a simplified diagram of the I/O buffer circuitry. The Output Enable signal (OE) enables the output buffer to pass the signal from the core logic to the pin. The output buffer contains electrostatic discharge (ESD) protection circuitry, an n-channel transistor that shunts all ESD surges (up to the limit of the device ESD specification) to GND. This transistor also serves as an output pull-down resistor.

Each output buffer also contains programmable slew rate, drive strength, programmable power-up state (pull-up/down resistor), hot-swap, 5 V tolerance, and clamp diode control circuitry. Multiple Flash switches (not shown in Figure 2) are programmed by user selections in the software to activate different I/O features.


Output Buffer
Hot-Swap, 5 V Tolerance, and Clamp Diode Control
VCCI
VCCI
VCCI
Weak Pull-Up Control (from core)
OE (from core logic)
Output Buffer Logic and Enable Skew Circuit
Drive Strength and Output Slew Rate Control
ESD Protection1
Clamp Diode
I/O PAD
Clamp Diode
Output Signal (from core logic)
Weak Pull-Down Control (from core)
ESD Protection1
Input Buffer
Input Buffer Standard2 and Schmitt Trigger Control
Input Signal to Core Logic
Programmable Input Delay Control

Notes:

1. *All NMOS transistors connected to the I/O pad serve as ESD protection.*
2. *See Table 1 on page 4 for available I/O standards.*

Figure 2 • **Simplified I/O Buffer Circuitry**

The I/O logic block contains several input, output, and enable registers. These registers can be combined with I/Os for better performance. Consult the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for a simplified representation of the I/O logic block. Refer to the "I/O Register Combining" section on page 8 for detailed architecture rules for this implementation.

Advanced I/O Usage

ProASIC3/E I/Os provide more design flexibility, allowing the user to control specific features by enabling certain I/O standards. Some features are selectable only for certain I/O standards, whereas others are available for all I/O standards. For example, slew control is not supported by differential I/O standards. Conversely, I/O register combining is supported by all I/O standards (see the "I/O Feature Description" section on page 7 for details). The ProASIC3/E datasheets provide the complete list of this I/O standard vs. I/O features matrix.

Figure 3 shows how to take advantage of the flexibility of the I/O assignments in the design. A detailed description is provided in later sections.

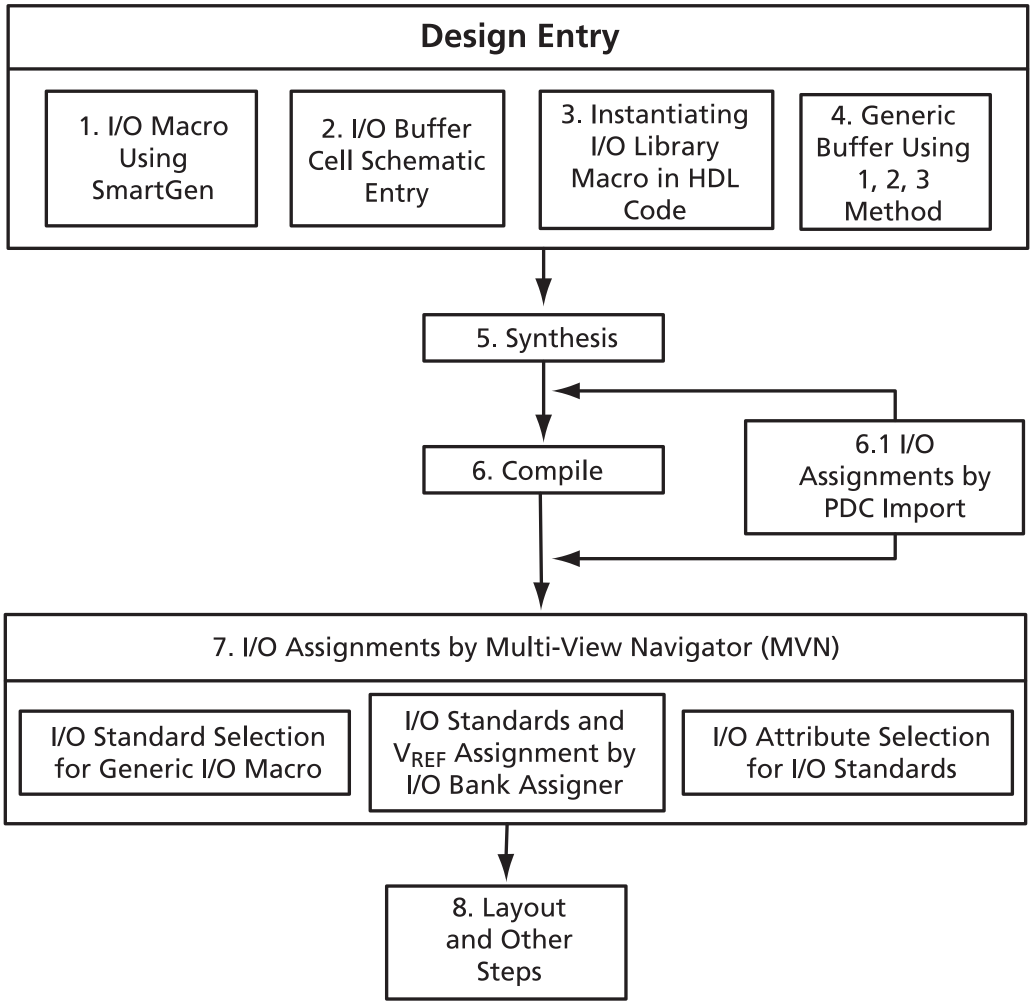

Design Entry
1. I/O Macro Using SmartGen
2. I/O Buffer Cell Schematic Entry
3. Instantiating I/O Library Macro in HDL Code
4. Generic Buffer Using 1, 2, 3 Method
5. Synthesis
6. Compile
6.1 I/O Assignments by PDC Import
7. I/O Assignments by Multi-View Navigator (MVN)
I/O Standard Selection for Generic I/O Macro
I/O Standards and V_{REF} Assignment by I/O Bank Assigner
I/O Attribute Selection for I/O Standards
8. Layout and Other Steps

Figure 3 • **User I/O Assignment Flow Chart**

I/O Standards

ProASIC3/E devices support up to 19 different I/O standards in three basic groups. Refer to the "Appendix" on page 28 for a detailed description of these I/O standards.

Table 1 shows the I/O standards supported within each of these basic groups.

Table 1 • **I/O Standard Matrix**

		I/O Standards		
		Single-Ended	**Differential**	**Voltage-Referenced**
Family	**Device**	**LVTTL, LVCMOS 3.3 V/2.5 V/1.8 V/1.5 V, 3.3 V PCI/3.3 V PCI-X, LVCMOS 2.5 V/5.0 V**	**LVDS, BLVDS, M-LVDS, LVPECL**	**GTL+ 2.5 V/3.3 V GTL 2.5 V/3.3 V HSTL Class 1 and 2 SSTL2 Class 1 and 2 SSTL3 Class 1 and 2**
ProASIC3	A3P030	Yes*	No	No
	A3P060	Yes	No	No
	A3P125	Yes	No	No
	A3P250	Yes	Yes	No
	A3P400	Yes	Yes	No
	A3P600	Yes	Yes	No
	A3P1000	Yes	Yes	No
ProASIC3E	A3PE600	Yes	Yes	Yes
	A3PE1500	Yes	Yes	Yes
	A3PE3000	Yes	Yes	Yes

Note: **A3P030 does not support 3.3 V PCI/PCI-X*

Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for V_{CCI}, V_{REF}, and board termination voltage requirements for the standards listed in Table 1.

I/O Banks and Standard Compatibility

ProASIC3/E I/Os are partitioned into multiple I/O voltage banks. There are eight I/O banks (two per side) in all ProASIC3E devices. For ProASIC3, the number of banks is device dependent; there are four I/O banks in all devices from A3P250 to A3P1000. The A3P030, A3P060, and A3P125 devices have two I/O banks.

Each I/O voltage bank has a separate ground and power plane for input and output circuits (VMV/GNDQ for input buffers and V_{CCI}/GND for output buffers). This isolation is necessary to minimize simultaneous switching noise from the input and output (SSI and SSO). The switching noise (ground bounce and power bounce) is generated by the output buffers and transferred into input buffer circuits, and vice versa.

Since voltage bounce originates on the package inductance, the VMV and V_{CCI} supplies have separate package pin assignments. For the same reason, GND and GNDQ also have separate pin assignments.

The VMV and V_{CCI} pins must be shorted to each other on the board. Also, the GND and GNDQ pins must be shorted to each other on the board. This will prevent unwanted current draw from the power supply.

ProASIC3E I/O banks are partitioned into multiple minibanks. A minibank is the scope of a V_{REF} pin. Any user I/O in a V_{REF} minibank can be configured as a V_{REF} pin. Only one V_{REF} is needed to control the entire minibank. The location and scope of the V_{REF} minibanks can be determined by the I/O name. For details, see the "User I/O Naming Convention" section of the *ProASIC3E datasheet*.

Because of these dedicated supplies, only I/Os with compatible standards can be assigned to the same I/O voltage bank. Two I/O standards are compatible under the following conditions:

- Their V_{CCI} and VMV values are identical
- If both of the standards need a V_{REF}, their V_{REF} values are identical (ProASIC3E only)

Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for the legal I/O usage matrix that shows compatible I/O standards for various values of V_{CCI} and V_{REF}.

BLVDS/M-LVDS

Bus LVDS (BLVDS) refers to bus interface circuits based on LVDS technology. Multipoint LVDS (M-LVDS) specifications extend the LVDS standard to high-performance multipoint bus applications. Multidrop and multipoint bus configurations may contain any combination of drivers, receivers, and transceivers. Actel LVDS drivers provide the higher drive current required by BLVDS and M-LVDS to accommodate the loading. The driver requires series terminations for better signal quality and to control voltage swing. Termination is also required at both ends of the bus, since the driver can be located anywhere on the bus. These configurations can be implemented using TRIBUF_LVDS and BIBUF_LVDS macros along with appropriate terminations. Multipoint designs using Actel LVDS macros can achieve up to 200 MHz with a maximum of 20 loads. A sample application is given in Figure 4. The input and output buffer delays are available in the LVDS sections in the *ProASIC3 datasheet* and the *ProASIC3E datasheet*.

Example: For a bus consisting of 20 equidistant loads, the terminations given in EQ 1 provide the required differential voltage, in worst case industrial operating conditions, at the farthest receiver:

$$R_S = 60\ \Omega,\ R_T = 70\ \Omega, \text{ given } Z_O = 50\ \Omega\ (2") \text{ and a } Z_{stub} = 50\ \Omega\ (\sim 1.5").$$

EQ 1

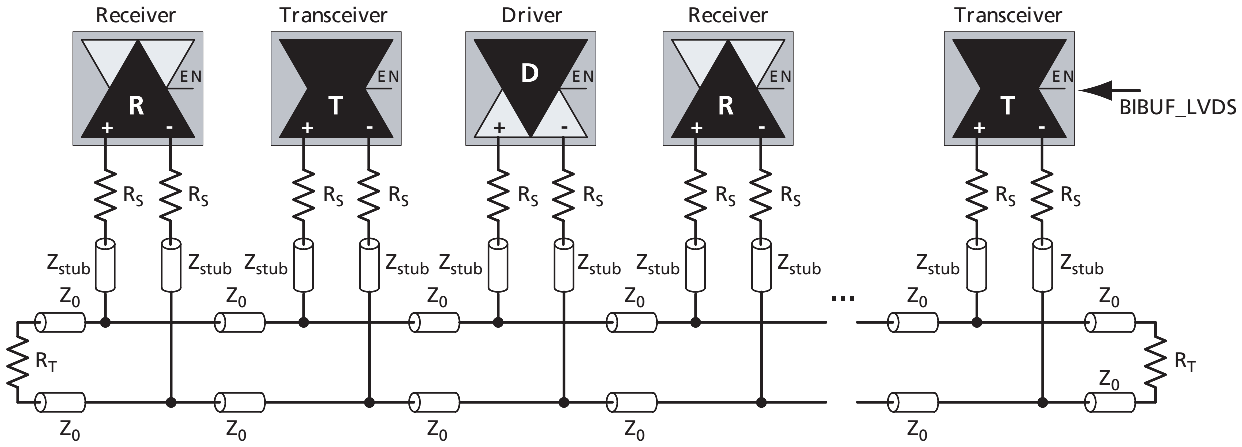


Figure 4 • **A BLVDS/M-LVDS Multipoint Application Using LVDS I/O Buffers**

I/O Features

ProASIC3/E devices offer many flexible I/O features to support a wide variety of board designs. Some of the features are programmable, with a range for selection. Table 2 lists programmable I/O features and their ranges.

Table 2 • **Programmable I/O Features (user control via I/O Attribute Editor)**

Feature	Description	Range
Slew Control	Output slew rate	HIGH, LOW
Output Drive (mA)	Output drive strength	2, 4, 6, 8, 12, 16, 24
Skew Control	Output tristate enable Delay option	ON, OFF
Resistor Pull	Resistor pull circuit	Up, Down, None
Input Delay	Input delay	OFF, 0–7
Schmitt Trigger	Schmitt trigger for input only	ON, OFF

Note: *Limitations of these features with respect to different ProASIC3/E devices are discussed in later sections.*

Users may modify these programmable I/O attributes using the I/O Attribute Editor. Modifying an I/O attribute may result in a change of state in Designer. Table 3 details which steps have to be re-run as a function of modified I/O attribute.

Table 3 • **Designer State (resulting from I/O attribute modification)**

	Designer States				
I/O Attribute	Compile	Layout	Fuse	Timing	Power
Slew Control	No	No	Yes	Yes	Yes
Output Drive (mA)	No	No	Yes	Yes	Yes
Skew Control	No	No	Yes	Yes	Yes
Resistor Pull	No	No	Yes	Yes	Yes
Input Delay	No	No	Yes	Yes	Yes
Schmitt Trigger	No	No	Yes	Yes	Yes
OUT_LOAD	No	No	No	Yes	Yes
COMBINE_REGISTER	Yes	Yes	N/A	N/A	N/A

Note: *No = Remains the same, Yes = Re-run the step, N/A = Not applicable*

The I/O feature support slightly differs between the ProASIC3 and ProASIC3E families. Table 4 lists the I/O features supported by ProASIC3/E devices. Not all features are supported by all I/O standards. Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for the I/O Attributes vs. I/O Standard Applications table.

Table 4 • **I/O Features Support by Devices**

		I/O Features								
Family	**Device**	**Output Drive**	**Slew**	**Resistor Pull**	**Schmitt Trigger**	**Input Delay**	**Skew**	**Output Load**	**Use I/O Register**	**Hot-Swappable**
ProASIC3	A3P030	Yes	Yes	Yes	No	No	Yes	Yes	Yes	Yes
	A3P060	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
	A3P125	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
	A3P250	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
	A3P400	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
	A4P600	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
	A3P1000	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
ProASIC3E	A3PE600	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	A3PE1500	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	A3PE3000	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

I/O Feature Description

ProASIC3/E devices support multiple I/O features that make the board design easier. For example, an I/O feature like Schmitt Trigger in the ProASIC3E input buffer saves the board space that would be used by an external Schmitt trigger for a slow or noisy input signal. These features are also programmable for each I/O, which in turn gives flexibility in interfacing with other components. The following is a detailed description of all available features in ProASIC3/E devices.

Output Slew Rate Control

The slew rate is the amount of time an input signal takes to get from logic low to logic high or vice versa. It is commonly defined as the propagation delay between 10% and 90% of the signal's voltage swing.

Slew rate control is available for the output buffers of ProASIC3/E devices. The output buffer has a programmable slew rate for both high-to-low and low-to-high transitions. Slew rate control is available for LVTTL, LVCMOS, and PCI-X I/O standards. The other I/O standards have a preset slew value.

The slew rate can be implemented by using the Physical Design Constraints (PDC) command (see Table 6 on page 16), setting "High" or "Low" in the I/O Attribute Editor in Designer, or instantiating a special I/O macro. The default slew rate value is "High."

Schmitt Trigger

A Schmitt trigger is a buffer used to convert a slow or noisy input signal into a clean one before passing it to the FPGA. Using Schmitt trigger buffers guarantees a fast, noise-free input signal to the FPGA.

ProASIC3E devices have Schmitt triggers built into their I/O circuitry. The Schmitt trigger is available for the LVTTL, LVCMOS, and 3.3 V PCI I/O standards.

This feature can be implemented by using the PDC command (see Table 6 on page 16) or by selecting a check box in the I/O Attribute Editor in Designer. The check box is cleared by default.

Skew

ProASIC3/E devices have a configurable skew block in the output buffer circuitry that can be enabled to delay output buffer assertion without affecting de-assertion time. Since this skew block is only available for the OE signal, the feature can be used in tristate and bidirectional buffers. A typical 1.2 ns delay is added to the OE signal in order to prevent potential bus contention. Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for detailed timing diagrams and descriptions.

The Skew feature is available for all I/O standards.

This feature can be implemented by using the PDC command (see Table 6 on page 16) or by selecting a check box in the I/O Attribute Editor in Designer. The check box is cleared by default.

Output Drive

The output buffers of ProASIC3/E devices can provide multiple drive strengths to meet signal integrity requirements. The LVTTL and LVCMOS standards have selectable drive strengths. Other standards have a preset value.

Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for the selectable drive strengths for the LVTTL and LVCMOS standards.

Hot-Swap

Hot-swap devices prevent failures by managing the inrush current during live insertion and removal of line cards. Hot-swap, hot plug, and hot dock are terms used interchangeably to refer to hot insertion and removal.

A pull-up clamp diode must not be present in the I/O circuitry if the hot-swap feature is used. The 3.3 V PCI standard requires a pull-up clamp diode on the I/O, so it cannot be selected if hot-swap capability is required. The A3P030 device does not support 3.3 V PCI, so it is the only device in the ProASIC3/E families that supports the hot-swap feature. All devices in the ProASIC3E family are hot-swappable. All standards except LVCMOS 2.5/5.0 V and 3.3 V PCI/PCI-X support the hot-swap feature.

The hot-swap feature appears as a read-only check box in the I/O Attribute Editor that shows whether an I/O is hot-swappable or not.

Refer to the *Power-Up/Down Behavior of ProASIC3/E Devices* application note for details on hot-swapping.

Cold Sparing

"Cold sparing" refers to the ability of a device to leave system data undisturbed when the system is powered up, while the component itself is powered down, or when power supplies are floating.

Cold sparing is supported on ProASIC3E devices only when the user provides resistors from each power supply to ground. The resistor value is calculated based on the decoupling capacitance on a given power supply. The RC constant should be greater than 3 µs.

To remove resistor current during operation, it is suggested that the resistor be disconnected (e.g., with an NMOS switch) from the power supply after the supply has reached its final value. Refer to the *Power-Up/Down Behavior of ProASIC3/E Devices* application note for details on cold sparing.

I/O Register Combining

Every ProASIC3/E I/O has several embedded registers in the I/O tile that are close to the I/O pads. Rather than using the internal register from the core, the user has the option of using these registers for faster clock-to-out timing. When combining these registers at the I/O buffer, some architectural rules must be met. Provided these rules are met, the user can enable register combining globally during Compile (as shown in Figure 11 on page 17).

This feature is supported by all I/O standards.

Rules for Registered I/O Function:

1. The fanout between an I/O pin (D, Y, or E) and a register must be equal to one for combining to be considered on that pin.
2. All registers (Input, Output, and Output Enable) connected to an I/O must share the same clear or preset function:
   - If one of the registers has a CLR pin, all the other registers that are candidates for combining in the I/O must have a CLR pin.
   - If one of the registers has a PRE pin, all the other registers that are candidates for combining in the I/O must have a PRE pin.
   - If one of the registers has neither a CLR nor a PRE pin, all the other registers that are candidates for combining must have neither a CLR nor a PRE pin.
   - If the clear or preset pins are present, then they must have the same polarity.
   - If the clear or preset pins are present, they must be driven by the same signal (net).
3. Registers connected to an I/O on the Output and Output Enable pins must have the same clock and enable function:
   - Both the Output and Output Enable registers must have an E pin (clock enable), or none at all.
   - If the E pins are present, then they must have the same polarity. The CLK pins must also have the same polarity.

In some cases, the user may want registers to be combined with the input of a bibuf while maintaining the output as-is. This can be achieved by using the PDC command, as follows:

```
set_io <signal name> -REGISTER yes ------register will combine

set_preserve <signal name> ----register will not combine
```

5 V Input and Output Tolerance

ProASIC3/E devices are both 5 V input and 5 V output tolerant if certain I/O standards are selected. Table 5 shows the I/O standards that support 5 V input tolerance. Only 3.3 V LVTTL/LVCMOS standards support 5 V output tolerance. Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for detailed description and configuration information.

This feature is not shown in the I/O Attribute Editor.

Table 5 • **5 V Input Tolerance**

	5 V Input Tolerance		
I/O Standard	**A3P030**	**Other ProASIC3**	**ProASIC3E**
3.3 V LVTTL/LVCMOS	Yes[1]	Yes[1]	Yes[1]
3.3 V PCI/PCI-X	N/A	Yes[1]	Yes[1]
LVCMOS 2.5 V	Yes[1]	N/A	No
LVCMOS 2.5 V / 5.0 V	N/A	Yes[2]	Yes[2]
LVCMOS 1.8 V	No	No	No
LVCMOS 1.5 V	No	No	No
Voltage-Referenced	N/A	N/A	No
Differential	N/A	No	No

Notes:

1. Can be implemented with an external IDT bus switch or resistor divider, or with Zener diode with resistor

2. Can be implemented with an external resistor divider and an internal clamp diode

Implementing ProASIC3/E I/Os in Actel Software

Actel Libero IDE is integrated with design entry tools such as the SmartGen macro builder, the ViewDraw schematic entry tool, and an HDL editor. It is also integrated with the Synthesis and Designer tools. In this section, all necessary steps to implement the ProSIC3/E I/Os will be discussed.

Design Entry

There are three ways to implement ProASIC3/E I/Os in a design:

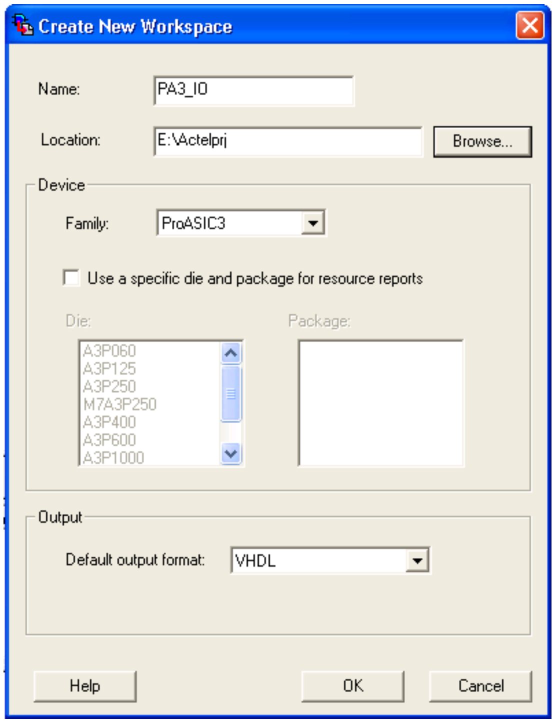
1. Use the SmartGen macro builder to configure I/Os by generating specific I/O library macros and then instantiating them in top-level code.
2. Use an I/O buffer cell in a schematic design.
3. Manually instantiate specific I/O macros in the top-level code.

If technology-specific macros, such as INBUF_LVCMOS33 and OUTBUT_PCI, are used in the HDL code or schematic, the user will not be able to change the I/O standard later on in Designer. If generic I/O macros are used, such as INBUF, OUTBUF, TRIBUFF, CLKBUF, and BIBUF, the user can change the I/O standard using the Designer I/O Attribute Editor tool.

Using SmartGen for I/O Configuration

The SmartGen tool in Designer provides a GUI-based method of configuring the I/O attributes. The user can select certain I/O attributes while configuring the I/O macro in SmartGen. The steps to configure an I/O macro with specific I/O attributes are as follows:

1. Invoke SmartGen by selecting **Start** > **Designer v7.0** (or **Actel Libero IDE v7.0**) > **SmartGen**.
2. From the **File** menu, select **New Workspace**.
3. The **Create New Workspace** window will appear (Figure 5). Type a name and select a location, family, and netlist format. Click **OK**.



Create New Workspace

Name: PA3_IO

Location: E:\Actelprj

Device

Family: ProASIC3

□ Use a specific die and package for resource reports

Die: A3P060, A3P125, A3P250, M7A3P250, A3P400, A3P600, A3P1000

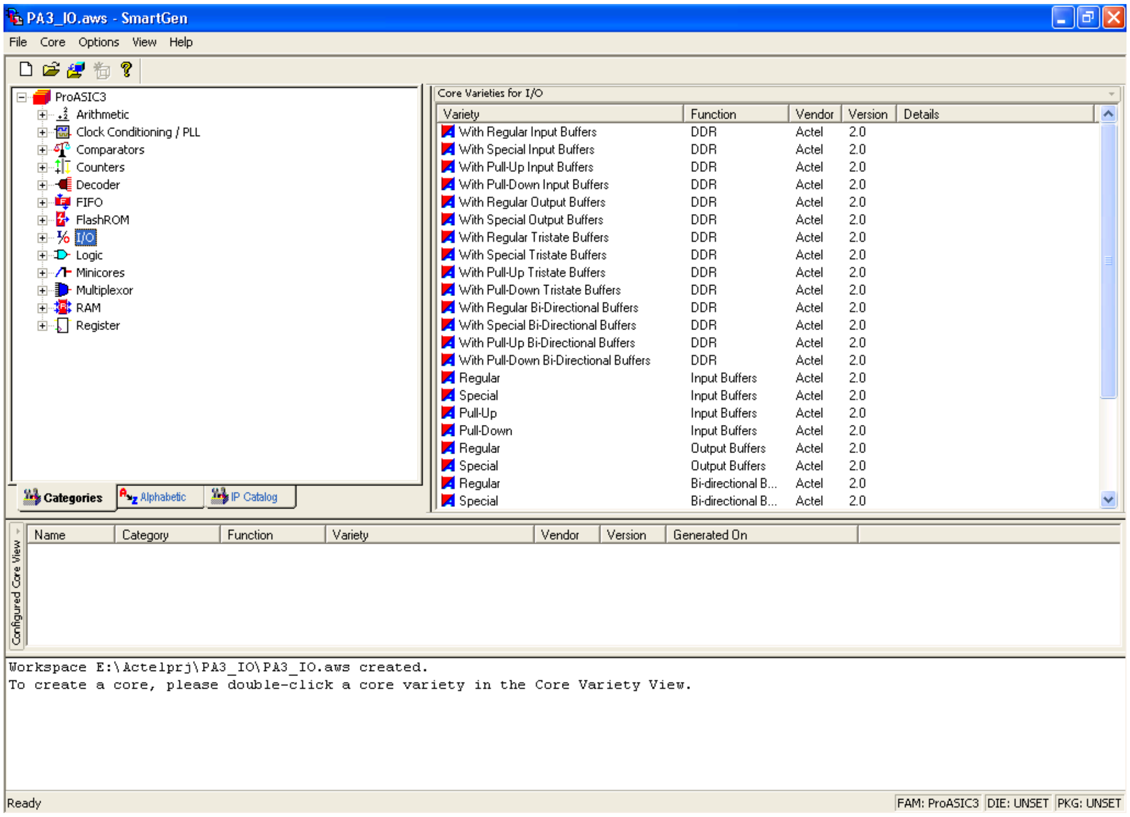
Package:

Output

Default output format: VHDL

Figure 5 • **Create New Workspace**

4. Click the **I/O** icon under the **Categories** tab. All configurable I/O macros will appear in the Core Varieties for I/O box (Figure 6).
5. Double-click any of the varieties. The I/O configuration window opens (Figure 7).



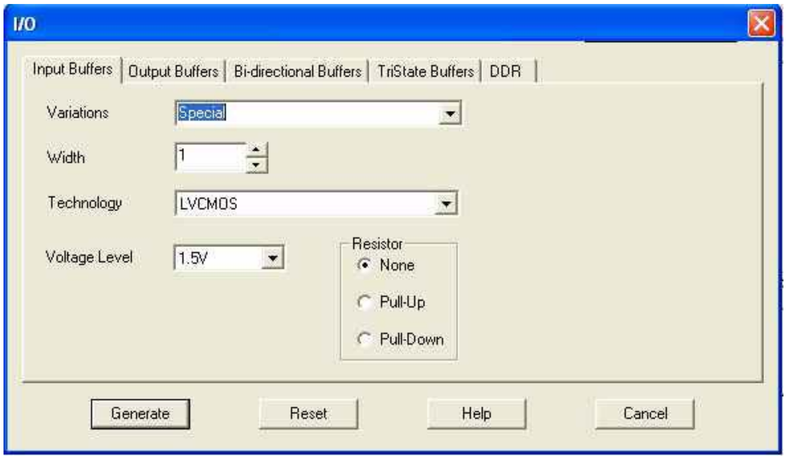
Core Varieties for I/O

Variety	Function	Vendor	Version	Details
With Regular Input Buffers	DDR	Actel	2.0	
With Special Input Buffers	DDR	Actel	2.0	
With Pull-Up Input Buffers	DDR	Actel	2.0	
With Pull-Down Input Buffers	DDR	Actel	2.0	
With Regular Output Buffers	DDR	Actel	2.0	
With Special Output Buffers	DDR	Actel	2.0	
With Regular Tristate Buffers	DDR	Actel	2.0	
With Special Tristate Buffers	DDR	Actel	2.0	
With Pull-Up Tristate Buffers	DDR	Actel	2.0	
With Pull-Down Tristate Buffers	DDR	Actel	2.0	
With Regular Bi-Directional Buffers	DDR	Actel	2.0	
With Special Bi-Directional Buffers	DDR	Actel	2.0	
With Pull-Up Bi-Directional Buffers	DDR	Actel	2.0	
With Pull-Down Bi-Directional Buffers	DDR	Actel	2.0	
Regular	Input Buffers	Actel	2.0	
Special	Input Buffers	Actel	2.0	
Pull-Up	Input Buffers	Actel	2.0	
Pull-Down	Input Buffers	Actel	2.0	
Regular	Output Buffers	Actel	2.0	
Special	Output Buffers	Actel	2.0	
Regular	Bi-directional B...	Actel	2.0	
Special	Bi-directional B...	Actel	2.0	

Name	Category	Function	Variety	Vendor	Version	Generated On

```
Workspace E:\Actelprj\PA3_IO\PA3_IO.aws created.
To create a core, please double-click a core variety in the Core Variety View.
```

Figure 6 • **Selecting an I/O Macro**



Variations: Special

Width: 1

Technology: LVCMOS

Voltage Level: 1.5V

Resistor
● None
○ Pull-Up
○ Pull-Down

Figure 7 • **I/O Configuration Window**

As seen in Figure 7, there are five tabs to configure the I/O macro: Input Buffers, Output Buffers, Bidirectional Buffers, Tristate Buffers, and DDR.

Input Buffers

There are two variations: Regular and Special.

If the **Regular** variation is selected, only the Width (1 to 128) needs to be entered. The default value for Width is 1.

The **Special** variation has Width, Technology, Voltage Level, and Resistor Pull-Up/Down options (see Figure 7 on page 11). All the I/O standards and supply voltages (V_{CCI}) supported for the device family are available for selection.

Output Buffers

There are two variations: Regular and Special.

If the **Regular** variation is selected, then only the Width (1 to 128) needs to be entered. The default value for Width is 1.

The **Special** variation has Width, Technology, Output Drive, and Slew Rate options.

Bidirectional Buffers

There are two variations: Regular and Special.

The **Regular** variation has Enable Polarity (Active High, Active Low) in addition to the Width option.

The **Special** variation has Width, Technology, Output Drive, Slew Rate, and Resistor Pull-Up/Down options.

Tristate Buffers

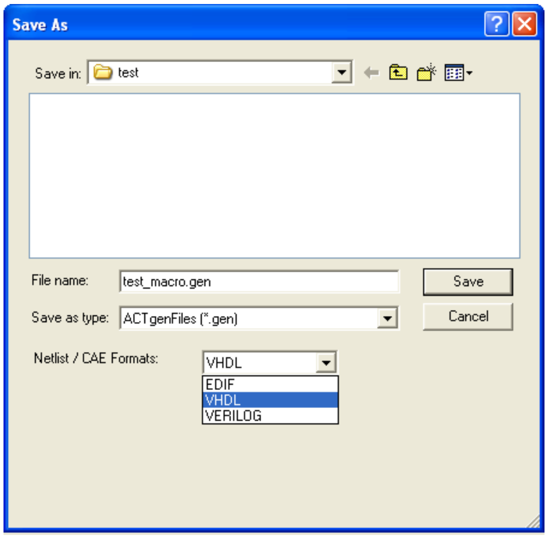
Same as Bidirectional Buffers.

DDR

There are eight variations: DDR with Regular Input Buffers, Special Input Buffers, Regular Output Buffers, Special Output Buffers, Regular Tristate Buffers, Special Tristate Buffers, Regular Bidirectional Buffers, and Special Bidirectional Buffers.

These variations resemble the options of the previous I/O macro. For example, the Special Input Buffers variation has Width, Technology, Voltage Level, and Resistor Pull-Up/Down options.

6. Once the desired configuration is selected, click the **Generate** button. The **Save As** window opens (Figure 8).
7. Enter a name for the macro and select the **Netlist** format (VHDL or VERILOG). Click **Save**. A netlist will be created and saved in the selected folder (Figure 9 on page 13).



Save in: test

File name: test_macro.gen

Save as type: ACTgenFiles (*.gen)

Netlist / CAE Formats: VHDL

Figure 8 • **Saving the I/O Macro Netlist**

8. Instantiate the I/O macro in the top-level code and include the SmartGen-created netlist file during Synthesis.

   The user must instantiate the DDR_REG or DDR_OUT macro in the design. Use SmartGen to generate both these macros and then instantiate them in your top level. To combine the DDR macros with the I/O, the following rules must be met:

   Rules for the DDR I/O Function

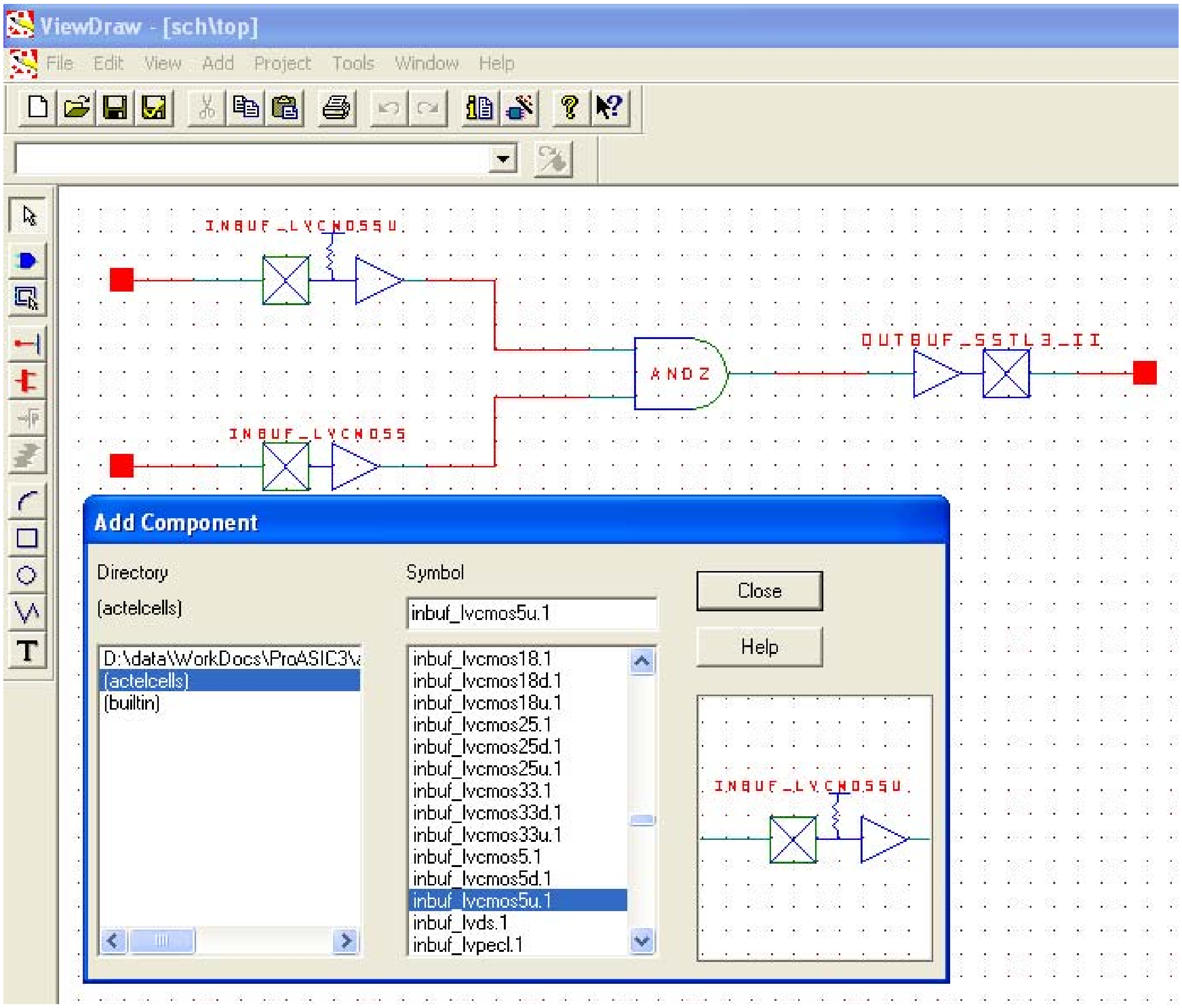
   – The fanout between an I/O pin (D or Y) and a DDR (DDR_REG or DDR_OUT) macro must be equal to one for the combining to happen on that pin.

   – If a DDR_REG macro and a DDR_OUT macro are combined on the same bidirectional I/O, they must share the same clear signal.

   – Registers will not be combined in an I/O in the presence of DDR combining on the same I/O.

Using the I/O Buffer Schematic Cell

Libero IDE has includes the ViewDraw schematic entry tool. Using ViewDraw, the user can insert any supported I/O buffer cell in the top-level schematic. Figure 9 shows a top level schematic with different I/O buffer cells. When synthesized, the netlist will contain the same I/O macro.



Directory (actelcells)

Symbol inbuf_lvcmos5u.1

Figure 9 • **I/O Buffer Schematic Cell Usage**

Instantiating in HDL code

All the supported I/O macros can be instantiated in the top-level HDL code (refer to the *Fusion and ProASIC3/E Macro Library Guide* for a detailed list of all I/O macros). The following is an example:

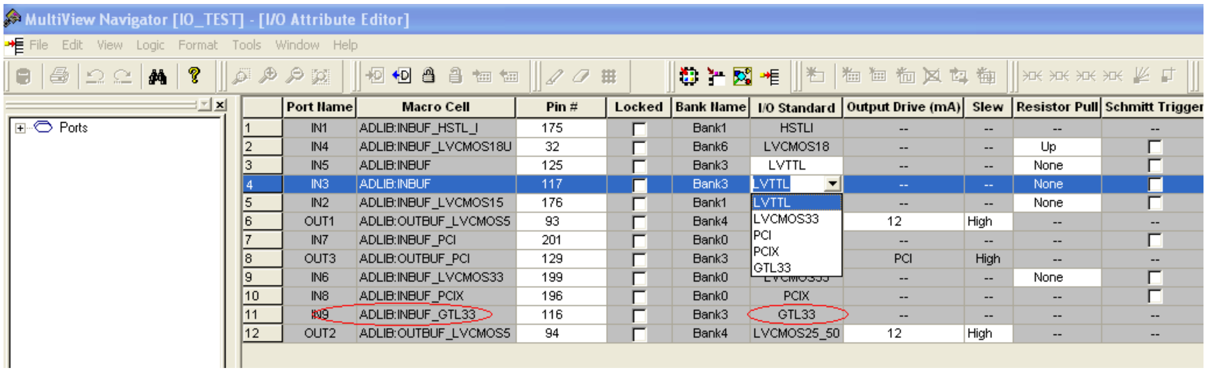
```
library ieee;
use ieee.std_logic_1164.all;
library proasic3e;

entity TOP is
    port(IN2, IN1 : in std_logic; OUT1 : out std_logic);
end TOP;

architecture DEF_ARCH of TOP is

  component INBUF_LVCMOS5U
    port(PAD : in std_logic := 'U'; Y : out std_logic);
  end component;

  component INBUF_LVCMOS5
    port(PAD : in std_logic := 'U'; Y : out std_logic);
  end component;

  component OUTBUF_SSTL3_II
    port(D : in std_logic := 'U'; PAD : out std_logic);
  end component;

  Other component …..

    signal x, y, z…….other signals : std_logic;

begin

  I1 : INBUF_LVCMOS5U
    port map(PAD => IN1, Y =>x);
  I2 : INBUF_LVCMOS5
    port map(PAD => IN2, Y => y);
  I3 : OUTBUF_SSTL3_II
    port map(D => z, PAD => OUT1);

  other port mapping…

end DEF_ARCH;
```

Synthesizing the Design

Libero IDE integrates with the Synplify® synthesis tool. Other synthesis tools can also be used with Libero IDE. Refer to the *Actel Libero IDE User's Guide* or Libero IDE online help for details on how to set up the Libero IDE tool profile with synthesis tools from other vendors.

During synthesis, the following rules apply:

- Generic macros:
  - Users can instantiate generic INBUF, OUTBUF, TRIBUF and BIBUF macros.
  - Synthesis will automatically infer generic I/O macros.
  - The default I/O technology for these macros is LVTTL.
  - Users will need to use the I/O Attribute Editor in Designer to change the default I/O standard if needed (see Figure 10).
- Technology-specific I/O macros:
  - Technology-specific I/O macros, such as INBUF_LVCMO25 and OUTBUF_GTL25, can be instantiated in the design. Synthesis will infer these I/O macros in the netlist.
  - The I/O standard of technology-specific I/O macros cannot be changed in the I/O Attribute Editor (see Figure 10).
  - The user MUST instantiate differential I/O macros (LVDS/LVPECL) in the design. This is the only way to use these standards in the design.
  - To implement the DDR I/O function, the user must instantiate a DDR_REG or DDR_OUT macro. This is the only way to use a DDR macro in the design.



MultiView Navigator [IO_TEST] - [I/O Attribute Editor]

	Port Name	Macro Cell	Pin #	Locked	Bank Name	I/O Standard	Output Drive (mA)	Slew	Resistor Pull	Schmitt Trigger
1	IN1	ADLIB:INBUF_HSTL_I	175	□	Bank1	HSTLI	--	--	--	--
2	IN4	ADLIB:INBUF_LVCMOS18U	32	□	Bank6	LVCMOS18	--	--	Up	□
3	IN5	ADLIB:INBUF	125	□	Bank3	LVTTL	--	--	None	□
4	IN3	ADLIB:INBUF	117	□	Bank3	LVTTL	--	--	None	□
5	IN2	ADLIB:INBUF_LVCMOS15	176	□	Bank1		--	--	None	□
6	OUT1	ADLIB:OUTBUF_LVCMOS5	93	□	Bank4		12	High	--	--
7	IN7	ADLIB:INBUF_PCI	201	□	Bank0		--	--	--	□
8	OUT3	ADLIB:OUTBUF_PCI	129	□	Bank3		PCI	High	--	--
9	IN6	ADLIB:INBUF_LVCMOS33	199	□	Bank0	LVCMOS33	--	--	None	□
10	IN8	ADLIB:INBUF_PCIX	196	□	Bank0	PCIX	--	--	--	□
11	IN9	ADLIB:INBUF_GTL33	116	□	Bank3	GTL33	--	--	--	--
12	OUT2	ADLIB:OUTBUF_LVCMOS5	94	□	Bank4	LVCMOS25_50	12	High	--	--

Figure 10 • **Assigning a Different I/O Standard to the Generic I/O Macro**

Performing Place-and-Route on the Design

The netlist created by the synthesis tool should now be imported into Designer and compiled. During Compile, the user may specify the I/O placement and attributes by importing the PDC file. The user can also specify the I/O placement and attributes using ChipPlanner and the I/O Attribute Editor, under MVN.

Defining I/O Assignments in the PDC

A PDC file is a Tcl script file specifying physical constraints. This file can be imported to and exported from Designer.

Table 6 shows I/O assignment constraints supported in the PDC file.

Table 6 • **PDC I/O Constraints**

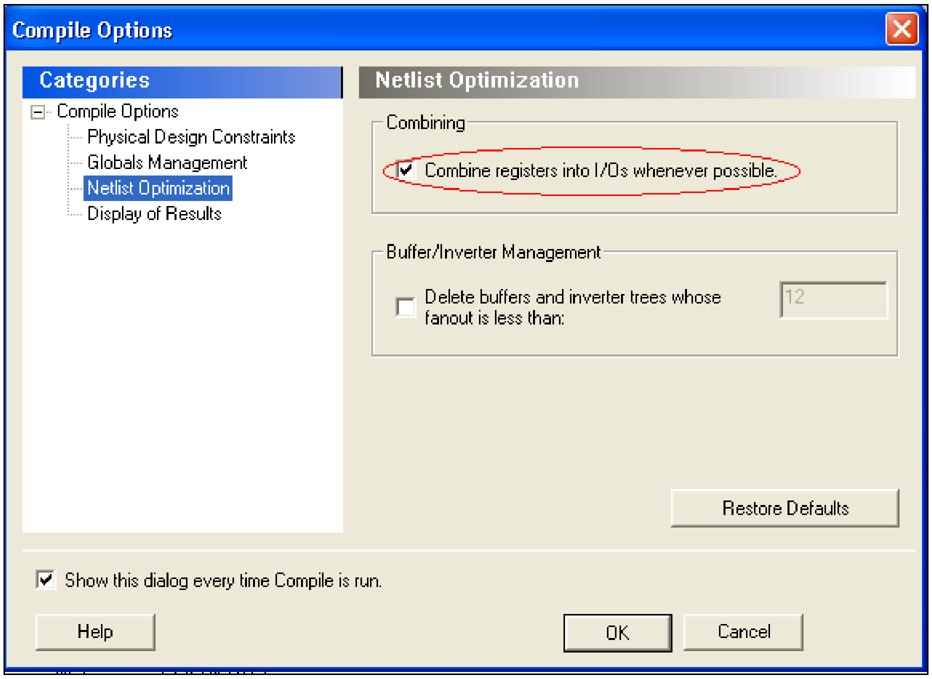
Command	**Action**	**Example**	**Comment**
I/O Banks Setting Constraints			
set_iobank	Sets the I/O supply voltage, V_{CCI}, and the input reference voltage, V_{REF}, (ProASIC3E only) for the specified I/O bank	set_iobank bankname [-vcci vcci_voltage] [-vref vref_voltage] set_iobank Bank7 -vcci 1.50 -vref 0.75	Must use in case of mixed I/O voltage (V_{CCI}) design
set_vref	Assigns a V_{REF} pin to a bank. The ProASIC3E family supports V_{REF} pins, but the ProASIC3 family does not.	set_vref -bank [bankname] [pinnum] set_vref -bank Bank0 685 704 723 742 761	Must use if voltage-referenced I/Os are used
set_vref_defaults	Sets the default V_{REF} pins for the specified bank. This command is ignored if the bank does not need a V_{REF} pin.	set_vref_defaults bankname set_vref_defaults bank2	
I/O Attribute Constraint			
set_io	Sets the attributes of an I/O	set_io portname [-pinname value] [-fixed value] [-iostd value] [-out_drive value] [-slew value] [-res_pull value] [-schmitt_trigger value] [-in_delay value] [-skew value] [-out_load value] [-register value] set_io IN2 -pinname 28 -fixed yes -iostd LVCMOS15 -out_drive 12 -slew high -RES_PULL None -SCHMITT_TRIGGER Off -IN_DELAY Off –skew off -REGISTER No	Arguments schmitt_trigger and in_delay are not available for ProASIC3. If the I/O macro is generic (e.g., INBUF) or technology-specific (INBUF_LVCMOS25), then all I/O attributes can be assigned using this constraint. If netlist has an I/O macro that specifies one of its attributes, then that attribute cannot be changed using this constraint, though other attributes can be changed. Example: OUTBUF_S_24 (low slew, output drive 24 mA) Slew and output drive cannot be changed.
I/O Region Placement Constraints			
define_region	Defines either a rectangular region or a rectilinear region	define_region -name [region_name] -type [region_type] x1 y1 x2 y2 define_region -name test -type inclusive 0 15 2 29	If any number of I/Os must be assigned to a particular I/O region, then such a region can be created with this constraint.
assign_region	Assigns a set of macros to a specified region	assign_region [region name] [macro_name...] assign_region test U12	This constraint assigns I/O macros to the I/O regions. When assigning an I/O macro, PDC naming conventions must be followed if the macro name contains special characters; e.g., if the macro name is \\$1I19\\, then the correct use of escape characters is \\\\\$1I19\\\\.

Note: *Refer to the Actel Libero IDE User's Guide for Software v7.0 for detailed rules on PDC naming and syntax conventions.*

Compiling the Design

During Compile, a PDC I/O constraint file can be imported along with the netlist file. If only the netlist file is compiled, then certain I/O assignments need to be completed before proceeding to Layout. All constraints that can be entered in PDC can also be entered using ChipPlanner, I/O Attribute Editor, and PinEditor.

There are certain rules that must be followed in implementing I/O register combining and the I/O DDR macro (refer to the "I/O Register Combining" section on page 8 and the "DDR" section on page 12 for details). Provided these rules are met, the user can enable or disable I/O register combining by using the PDC command (set_io portname –register yes|no) in the I/O Attribute Editor or selecting a check box in the Compile Options dialog box (see Figure 11). The Compile Options dialog box appears when the design is compiled for the first time. It can also be accessed by selecting **Options** > **Compile** during successive runs. I/O register combining is off by default. The PDC command overrides the setting in the Compile Options dialog box.



Compile Options

Netlist Optimization

Combining

☑ Combine registers into I/Os whenever possible.

Buffer/Inverter Management

☐ Delete buffers and inverter trees whose fanout is less than: 12

☑ Show this dialog every time Compile is run.

Figure 11 • **Setting Register Combining During Compile**

Understanding the Compile Report

The I/O bank report is generated during Compile and displayed in the log window. This report lists the I/O assignments necessary before Layout can proceed.

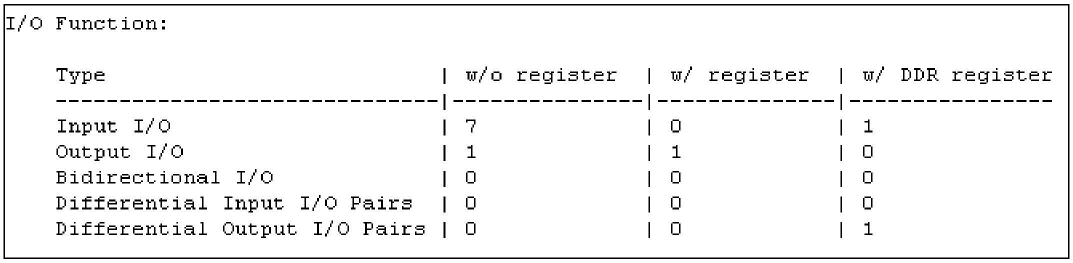
When Designer v6.3 is started, the I/O Bank Assigner tool is run automatically if the Layout command is executed. The I/O Bank Assigner takes care of the necessary I/O assignments. However, these assignments can also be made manually with MVN or by importing the PDC file. Refer to the "Assigning Technologies and V_{REF} to I/O Banks" section on page 20 for further description.

The I/O bank report can also be extracted from Designer by selecting **Tools** > **Report** and setting the Report Type to **IOBank**.

This report has the following tables: I/O Function, I/O Technology, I/O Bank Resource Usage, and I/O Voltage Usage. This report is useful if the user wants to do I/O assignments manually.

I/O Function

Figure 12 shows an example of the I/O Function table included in the I/O bank report:



```
I/O Function:

    Type                          | w/o register   | w/ register   | w/ DDR register
    ------------------------------|----------------|---------------|----------------
    Input I/O                     | 7              | 0             | 1
    Output I/O                    | 1              | 1             | 0
    Bidirectional I/O             | 0              | 0             | 0
    Differential Input I/O Pairs  | 0              | 0             | 0
    Differential Output I/O Pairs | 0              | 0             | 1
```

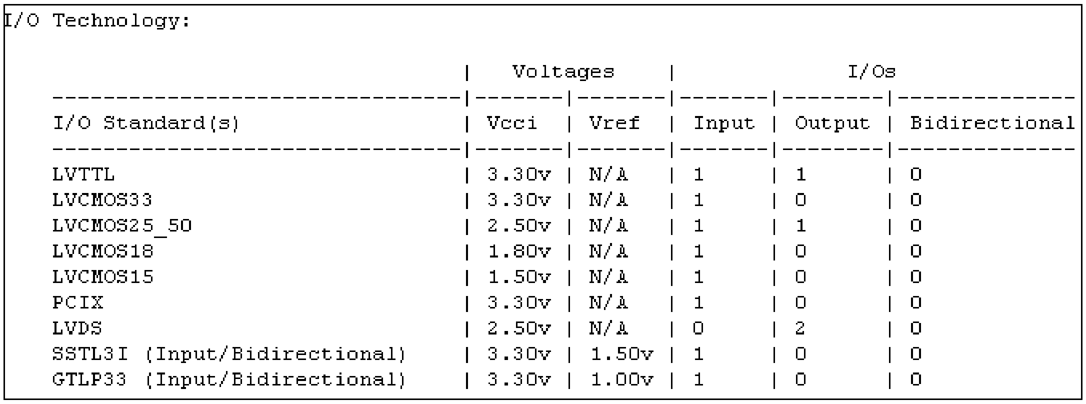
Figure 12 • **I/O Function Table**

This table lists the number of input I/Os, output I/Os, bidirectional I/Os, and differential input and output I/O pairs that use I/O and DDR registers.

Certain rules must be met to implement registered and DDR I/O functions (refer to the "I/O Register Combining" section on page 8 and the "DDR" section on page 12).

I/O Technology

The I/O Technology table (shown in Figure 13) gives the values of V_{CCI} and V_{REF} (reference voltage) for all the I/O standards used in the design. The user should assign these voltages appropriately.

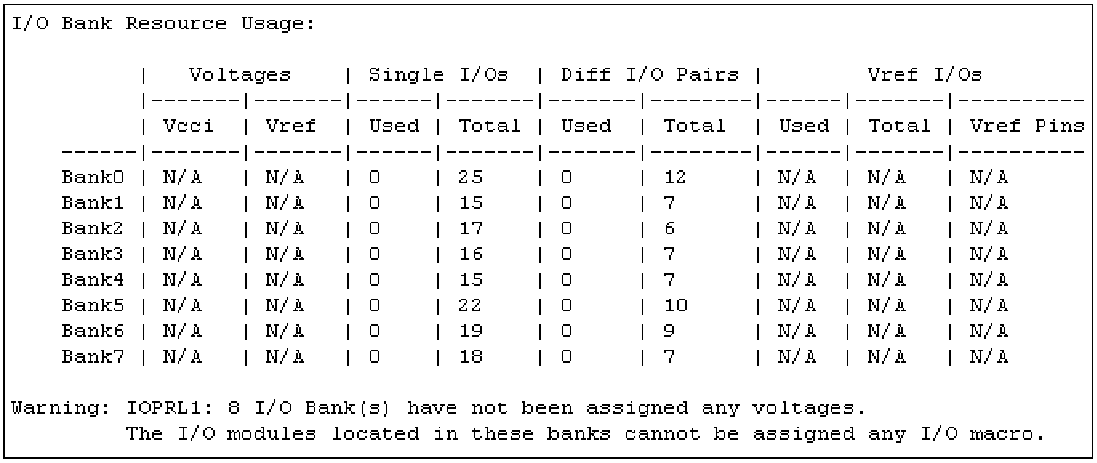


```
I/O Technology:

                                    |   Voltages    |             I/Os
    --------------------------------|-------|-------|-------|--------|--------------
    I/O Standard(s)                 | Vcci  | Vref  | Input | Output | Bidirectional
    --------------------------------|-------|-------|-------|--------|--------------
    LVTTL                           | 3.30v | N/A   | 1     | 1      | 0
    LVCMOS33                        | 3.30v | N/A   | 1     | 0      | 0
    LVCMOS25_50                     | 2.50v | N/A   | 1     | 1      | 0
    LVCMOS18                        | 1.80v | N/A   | 1     | 0      | 0
    LVCMOS15                        | 1.50v | N/A   | 1     | 0      | 0
    PCIX                            | 3.30v | N/A   | 1     | 0      | 0
    LVDS                            | 2.50v | N/A   | 0     | 2      | 0
    SSTL3I (Input/Bidirectional)    | 3.30v | 1.50v | 1     | 0      | 0
    GTLP33 (Input/Bidirectional)    | 3.30v | 1.00v | 1     | 0      | 0
```

Figure 13 • **I/O Technology Table**

I/O Bank Resource Usage

This is an important portion of the report. The user must meet the requirements stated in this table. Figure 14 shows the I/O Bank Resource Usage table included in the I/O bank report:



```
I/O Bank Resource Usage:

          |   Voltages    | Single I/Os  | Diff I/O Pairs |          Vref I/Os
          |-------|-------|------|-------|-------|--------|------|-------|----------
          | Vcci  | Vref  | Used | Total | Used  | Total  | Used | Total | Vref Pins
    ------|-------|-------|------|-------|-------|--------|------|-------|----------
    Bank0 | N/A   | N/A   | 0    | 25    | 0     | 12     | N/A  | N/A   | N/A
    Bank1 | N/A   | N/A   | 0    | 15    | 0     | 7      | N/A  | N/A   | N/A
    Bank2 | N/A   | N/A   | 0    | 17    | 0     | 6      | N/A  | N/A   | N/A
    Bank3 | N/A   | N/A   | 0    | 16    | 0     | 7      | N/A  | N/A   | N/A
    Bank4 | N/A   | N/A   | 0    | 15    | 0     | 7      | N/A  | N/A   | N/A
    Bank5 | N/A   | N/A   | 0    | 22    | 0     | 10     | N/A  | N/A   | N/A
    Bank6 | N/A   | N/A   | 0    | 19    | 0     | 9      | N/A  | N/A   | N/A
    Bank7 | N/A   | N/A   | 0    | 18    | 0     | 7      | N/A  | N/A   | N/A

Warning: IOPRL1: 8 I/O Bank(s) have not been assigned any voltages.
         The I/O modules located in these banks cannot be assigned any I/O macro.
```

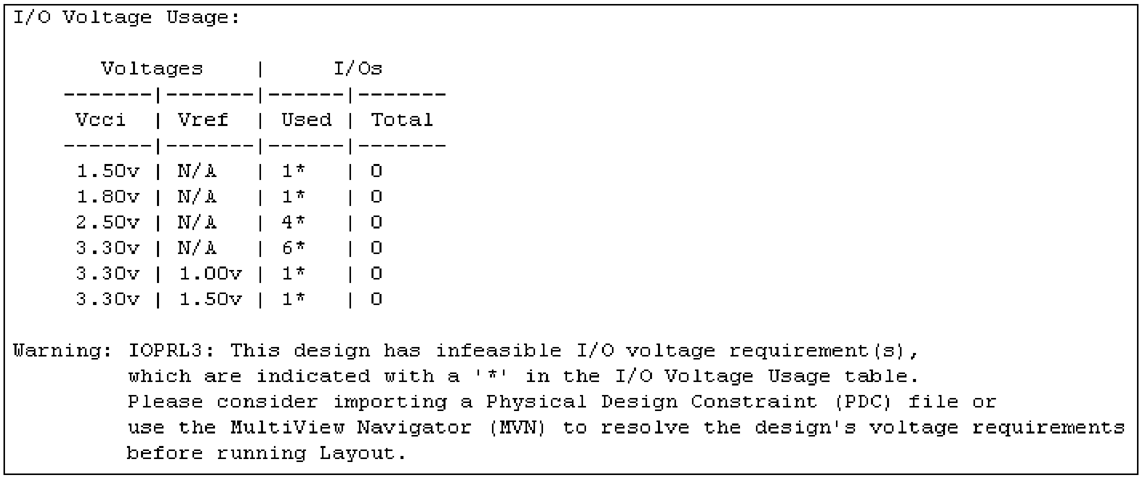
Figure 14 • **I/O Bank Resource Usage Table**

The example in Figure 14 shows that none of the I/O macros are assigned to the bank because more than one V_{CCI} is detected.

I/O Voltage Usage

The I/O Voltage Usage table provides the number of V_{REF} (ProASIC3E only) and V_{CCI} assignments required in the design. If the user decides to make I/O assignments manually (PDC or MVN), the issues listed in this table must be resolved before proceeding to Layout. As stated earlier, V_{REF} assignments must be made if there are any voltage-referenced I/Os.

Figure 15 shows the I/O Voltage Usage table included in the I/O bank report.



```
I/O Voltage Usage:

       Voltages     |      I/Os
    -------|-------|------|-------
     Vcci  | Vref  | Used | Total
    -------|-------|------|-------
     1.50v | N/A   | 1*   | 0
     1.80v | N/A   | 1*   | 0
     2.50v | N/A   | 4*   | 0
     3.30v | N/A   | 6*   | 0
     3.30v | 1.00v | 1*   | 0
     3.30v | 1.50v | 1*   | 0

Warning: IOPRL3: This design has infeasible I/O voltage requirement(s),
         which are indicated with a '*' in the I/O Voltage Usage table.
         Please consider importing a Physical Design Constraint (PDC) file or
         use the MultiView Navigator (MVN) to resolve the design's voltage requirements
         before running Layout.
```

Figure 15 • **I/O Voltage Usage Table**

The table in Figure 15 on page 19 indicates that there are two voltage-referenced I/Os used in the design. Even though both of the voltage-referenced I/O technologies have the same V_{CCI} voltage, their V_{REF} voltages are different. As a result, two I/O banks are needed to assign the V_{CCI} and V_{REF} voltages.

In addition, there are six single-ended I/Os used that have the same V_{CCI} voltage. Since two banks are already assigned with the same V_{CCI} voltage and there are enough unused bonded I/Os in those banks, the user does not need to assign the same V_{CCI} voltage to another bank. The user needs to assign the other three V_{CCI} voltages to three more banks.

Assigning Technologies and V_{REF} to I/O Banks

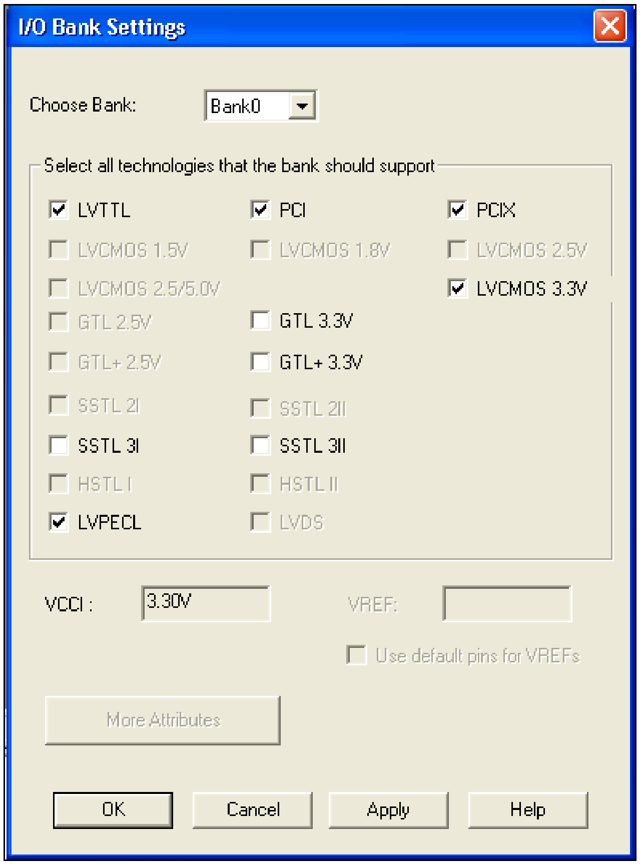
The ProASIC3/E families offer a wide variety of I/O standards, including voltage-referenced standards. Before proceeding to Layout, each bank must have the required V_{CCI} voltage assigned for the corresponding I/O technologies used for that bank. The voltage-referenced standards require the use of a reference voltage (V_{REF}). This assignment can be done manually or automatically. The following sections describe this in detail.

Manually Assigning Technologies to I/O Banks

The user can import the PDC at this point and resolve this requirement. The PDC command is

```
set_iobank [bank name] -vcci [vcci value]
```

Another method is to use the I/O Bank Settings dialog box (**MVN** > **Edit** > **I/O Bank Settings**) to set up the V_{CCI} voltage for the bank (Figure 16).



I/O Bank Settings

Choose Bank: Bank0

Select all technologies that the bank should support

☑ LVTTL ☑ PCI ☑ PCIX

□ LVCMOS 1.5V □ LVCMOS 1.8V □ LVCMOS 2.5V

□ LVCMOS 2.5/5.0V ☑ LVCMOS 3.3V

□ GTL 2.5V □ GTL 3.3V

□ GTL+ 2.5V □ GTL+ 3.3V

□ SSTL 2I □ SSTL 2II

□ SSTL 3I □ SSTL 3II

□ HSTL I □ HSTL II

☑ LVPECL □ LVDS

VCCI : 3.30V

VREF:

□ Use default pins for VREFs

Figure 16 • **Setting V_{CCI} for a Bank**

The procedure is as follows:

1. Select the bank to which you want V_{CCI} to be assigned from the **Choose Bank** list.
2. Select the I/O standards for that bank. If you select any standard, the tool will automatically show all compatible standards that have a common V_{CCI} voltage requirement.
3. Click **Apply**.
4. Repeat steps 1–3 to assign V_{CCI} voltages to other banks. Refer to Figure 15 on page 19 to find out how many I/O banks are needed for V_{CCI} bank assignment.

Manually Assigning V_{REF} Pins

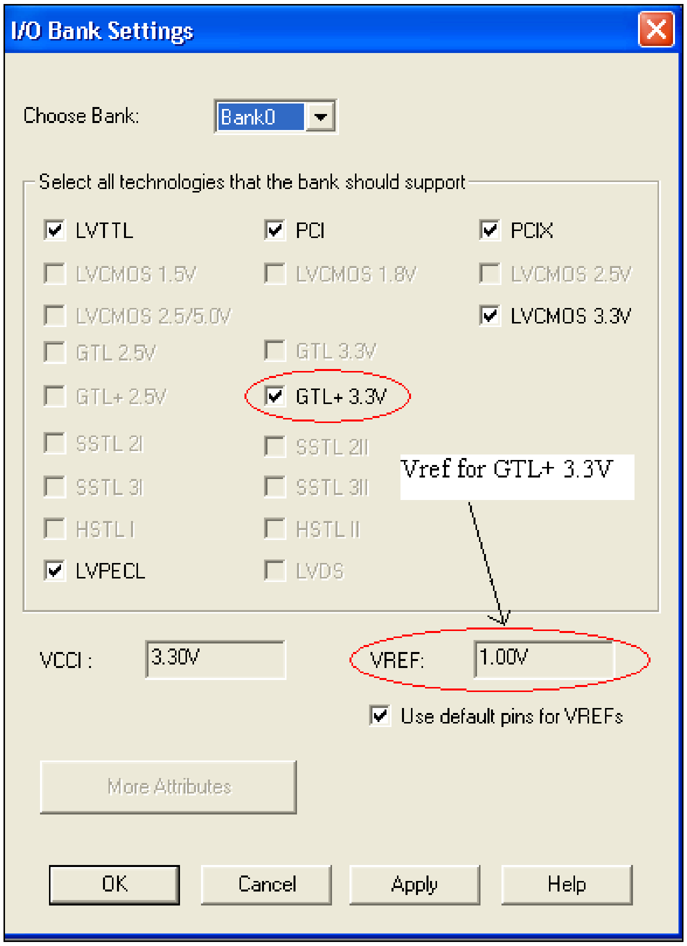
Voltage-referenced inputs require an input reference voltage (V_{REF}). The user must assign V_{REF} pins before running Layout. Before assigning a V_{REF} pin, the user must set a V_{REF} technology for the bank to which the pin belongs.

Assigning the V_{REF} Voltage to a Bank

When importing the PDC file, the V_{REF} voltage can be assigned to the I/O bank. The PDC command is as follows:

```
set_iobank -vref [value]
```

Another method for assigning V_{REF} is by using **MVN** > **Edit** > **I/O Bank Settings** (Figure 17).



I/O Bank Settings

Choose Bank: Bank0

Select all technologies that the bank should support

☑ LVTTL ☑ PCI ☑ PCIX
☐ LVCMOS 1.5V ☐ LVCMOS 1.8V ☐ LVCMOS 2.5V
☐ LVCMOS 2.5/5.0V ☑ LVCMOS 3.3V
☐ GTL 2.5V ☐ GTL 3.3V
☐ GTL+ 2.5V ☑ GTL+ 3.3V
☐ SSTL 2I ☐ SSTL 2II
☐ SSTL 3I ☐ SSTL 3II
☐ HSTL I ☐ HSTL II
☑ LVPECL ☐ LVDS

VCCI : 3.30V

VREF: 1.00V

☑ Use default pins for VREFs

Vref for GTL+ 3.3V

Figure 17 • **Selecting V_{REF} Voltage for the I/O Bank**

Assigning V_{REF} Pins for a Bank

The user can use default pins for V_{REF}. In this case, select the **Use default pins for V_{REF}s** check box (Figure 17 on page 21). This option guarantees full V_{REF} coverage of the bank. The equivalent PDC command is as follows:

```
set_vref_default [bank name]
```

To be able to choose V_{REF} pins, adequate V_{REF} pins must be created to allow legal placement of the compatible voltage-referenced I/Os.

To assign V_{REF} pins manually, the PDC command is as follows:

```
set_vref -bank [bank name] [package pin numbers]
```

For ChipPlanner/PinEditor to show the range of a V_{REF} pin, perform the following steps:

1. Assign V_{CCI} to a bank using **MVN** > **Edit** > **I/O Bank Settings**.
2. Open **ChipPlanner**. Zoom in on an I/O package pin in that bank.
3. Highlight the pin and then right-click. Choose **Use Pin for V_{REF}**.
4. Right-click and then choose **Show V_{REF} range**. All the pins covered by that V_{REF} pin will be highlighted (Figure 18).

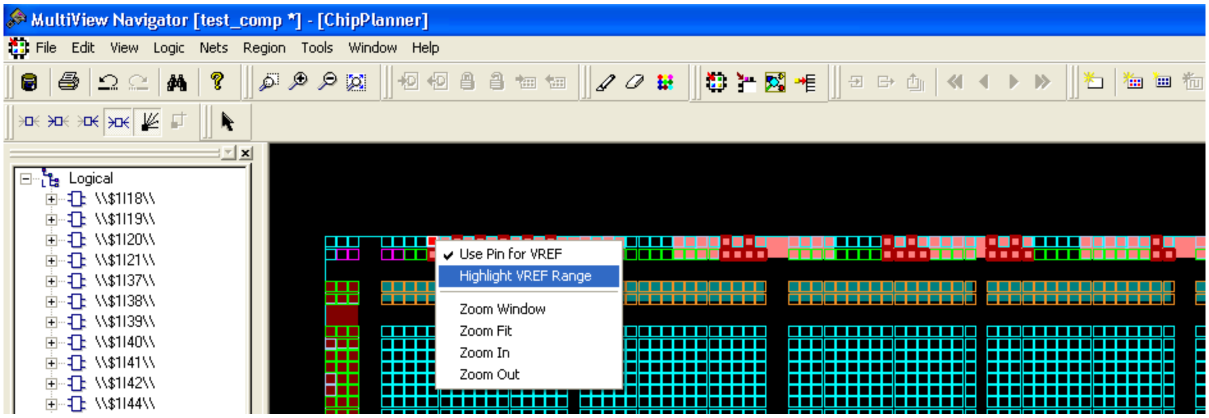


Figure 18 • **V_{REF} Range**

Using PinEditor or ChipPlanner, V_{REF} pins can also be assigned (Figure 19).

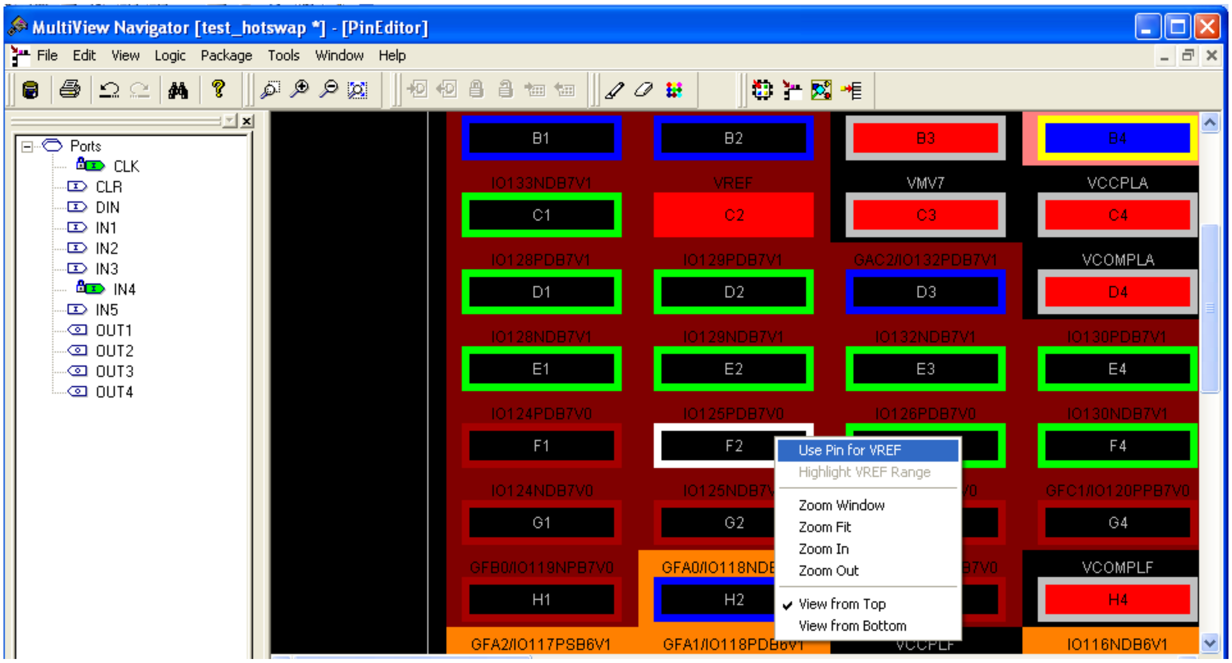


Figure 19 • **Assigning V_{REF} from PinEditor**

To unassign a V_{REF} pin:

1. Select the pin to unassign.
2. Right-click and choose **Use Pin for V_{REF}.** The check mark next to the command disappears. The V_{REF} pin is now a regular pin.

Resetting the pin may result in unassigning I/O cores, even if they are locked. In this case, a warning message appears so you can cancel the operation.

After you assign the V_{REF} pins, right-click a V_{REF} pin and choose **Highlight VREF Range** to see how many I/Os are covered by this pin. To unhighlight the range, choose **Unhighlight All** from the **Edit** menu.

Automatically Assigning Technologies to I/O Banks

The I/O Bank Assigner (IOBA) tool runs automatically when you run Layout. You can also use this tool from within the MultiView Navigator (Figure 21 on page 24). The IOBA tool automatically assigns technologies and V_{REF} pins (if required) to every I/O bank that does not currently have any technologies assigned to it. This tool is available when at least one I/O bank is unassigned.

To automatically assign technologies to I/O banks, choose **I/O Bank Assigner** from the **Tools** menu (or click the I/O Bank Assigner's toolbar button, shown in Figure 20).

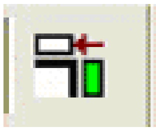
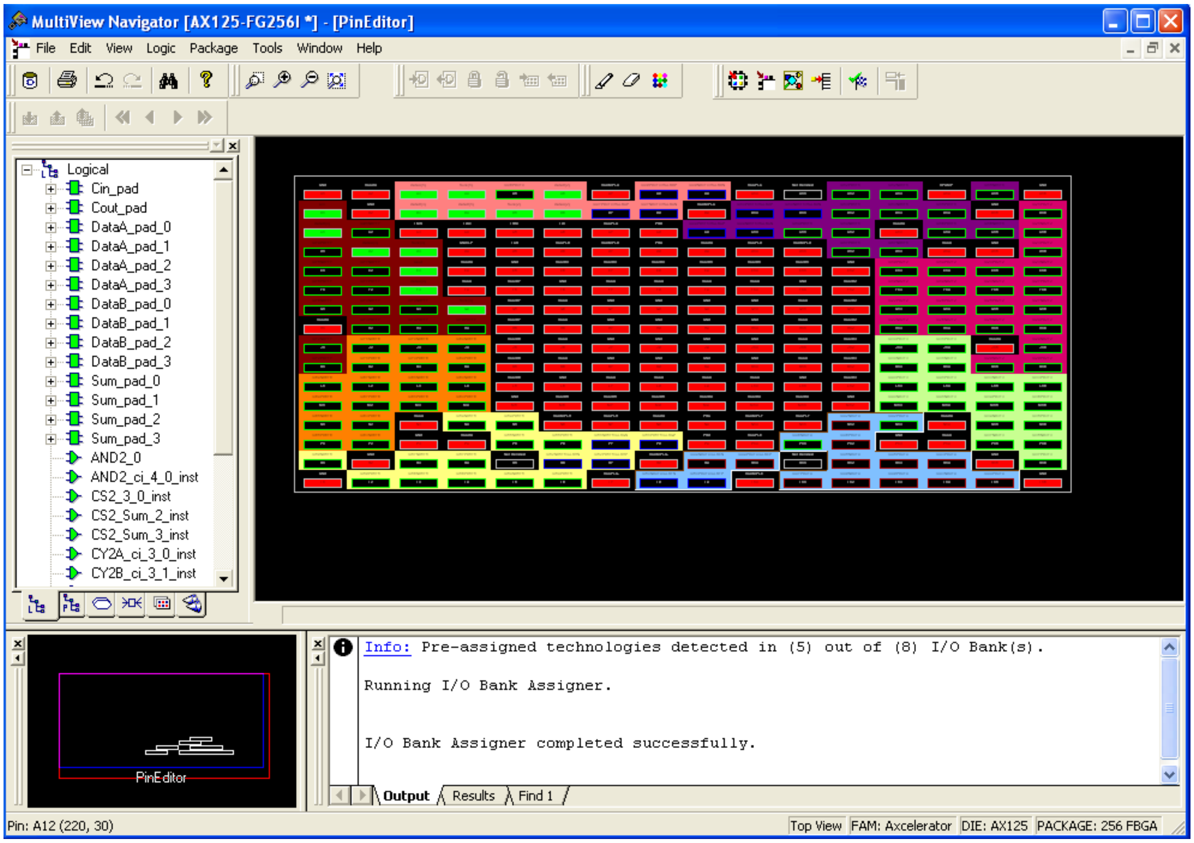


Figure 20 • **I/O Bank Assigner's Toolbar Button**

Messages will appear in the Output window informing you when the automatic I/O bank assignment begins and ends. If the assignment is successful, the message, "I/O Bank Assigner completed successfully," appears in the Output window, as shown in Figure 21.



```
Info: Pre-assigned technologies detected in (5) out of (8) I/O Bank(s).

Running I/O Bank Assigner.


I/O Bank Assigner completed successfully.
```

Figure 21 • **I/O Bank Assigner Displays Messages in Output Window**

If the assignment is not successful, an error message appears in the Output window.

To undo the I/O bank assignments, choose **Undo** from the **Edit** menu. Undo removes the I/O technologies assigned by the IOBA. It does not remove the I/O technologies previously assigned.

To redo the changes undone by the Undo command, choose **Redo** from the **Edit** menu.

To clear I/O bank assignments made before using the Undo command, manually unassign or reassign I/O technologies to banks. To do so, choose **I/O Bank Settings** from the **Edit** menu to display the I/O Bank Settings dialog box.

Board-Level Considerations

ProASIC3/E devices have robust I/O features that can help in reducing board-level components. The ProASIC3/E families offer single-chip solutions, which makes the board layout simpler and more immune to signal integrity issues. Although, in many cases, ProASCI3/E devices resolve many board-level issues, special attention should be given to overall signal integrity issues. This section covers important board-level considerations to facilitate optimum device performance.

Termination

Proper termination of all signals is essential for good signal quality. Nonterminated signals, especially clock signals, can cause malfunctioning of the device.

For general termination guidelines, refer to the *Board-Level Considerations* application note for Actel FPGAs. Also refer to the "Pin Descriptions" sections in of the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for termination requirements for specific pins.

ProASIC3/E I/Os are equipped with on-chip pull-up/down resistors. The user can enable these resistors either by instantiating them in the top level of the design (refer to the *ProASIC3/E Macro Library Guide* for the available I/O macros with pull-up/down) or in the I/O Attribute Editor in Designer if generic input or output buffers are instantiated in the top level. Unused I/O pins are configured as inputs with pull-up resistors.

As mentioned earlier, ProASIC3/E devices have multiple programmable drive strengths, and the user can eliminate unwanted overshoot and undershoot by adjusting the drive strengths.

Power-Up Behavior

ProASIC3/E devices are power-up/down friendly; i.e., no particular sequencing is required for power-up and power-down. This eliminates extra board components for power-up sequencing, such as a power-up sequencer.

During power-up, all I/Os are tristated, irrespective of I/O macro type (input buffers, output buffers, I/O buffer with weak pull-ups, weak pull-downs, etc.). Once I/Os become activated, they are set to the user-selected I/O macros. Refer to the *Power-Up/Down Behavior of ProASIC3/E Devices* application note for details.

Drive Strength

ProASIC3/E devices have up to seven programmable output drive strengths. The user can select the drive strength of a particular output in the I/O Attribute Editor or can instantiate a specialized I/O macro, such as OUTBUF_S_24 (slew = low, out_drive = 24 mA).

The maximum available drive strength is 24 mA per I/O. Though no I/O should be forced to source or sink more than 24 mA indefinitely, I/Os may handle a higher amount of current (refer to the device IBIS model for maximum source/sink current) during signal transition (AC current). Every device package has its own power dissipation limit, hence power calculation must be performed accurately to determine how much current can be tolerated per I/O within that limit.

I/O Interfacing

ProASIC3/E devices are 5 V input and output tolerant without adding any extra circuitry. Along with other low-voltage I/O macros, this 5 V tolerance makes these devices suitable for many types of board component interfacing.

Table 7 shows some high-level interfacing examples using ProASIC3/E devices.

Table 7 • **ProASIC3/E High-Level Interface**

	Clock		I/O			
Interface	**Type**	**Frequency**	**Type**	**Signals In**	**Signals Out**	**Data I/O**
GM	Src Sync	125 MHz	LVTTL	8	8	125 Mbps
TBI	Src Sync	125 MHz	LVTTL	10	10	125 Mbps
XSBI	Src Sync	644 MHz	LVDS	16	16	644 Mbps
XGMI	Src Sync DDR	156 MHz	HSTL1	32	32	312 Mbps
FlexBus 3	Sys Sync	104 MHz	LVTTL	≤32	≤32	≤104
Pos-PHY3/SPI-3	Sys Sync	104	LVTTL	8,16,32	8,16,32	≤104 Mbps
FlexBus 4/SPI-4.1	Src Sync	200 MHz	HSTL1	16,64	16,64	200 Mbps
Pos-PHY4/SPI-4.2	Src Sync DDR	≥311 MHz	LVDS	16	16	≥622 Mbps
SFI-4.1	Src Sync	622 MHz	LVDS	16	16	622 Mbps
CSIX L1	Sys Sync	≤250 MHz	HSTL1	32,64,96,128	32,64,96,128	≤250 Mbps
Hyper Transport	Sys Sync DDR	≤800 MHz	LVDS	2,4,8,16	2,4,8,16	≤1.6 Gbps
Rapid I/O Parallel	Sys Sync DDR	250 MHz – 1 GHz	LVDS	8,16	8,16	≤2 Gbps
Star Fabric	CDR		LVDS	4	4	622 Mbps

Note: *Sys Sync = System Synchronous Clocking, Src Sync = Source Synchronous Clocking, and CDR = Clock and Data Recovery.*

Conclusion

ProASIC3/E support for multiple I/O standards minimizes board-level components and makes possible a wide variety of applications. The Actel Designer software, integrated with Actel Libero IDE, presents a clear visual display of I/O assignments, allowing users to verify I/O and board-level design requirements before programming the device. The ProASIC3/E device I/O features and functionalities ensure board designers can produce low cost and low power FPGA applications fulfilling the complexities of contemporary design needs.

Related Documents

Datasheets

ProASIC3 Flash Family FPGAs

http://www.actel.com/documents/PA3_DS.pdf

ProASIC3E Flash Family FPGAs

http://www.actel.com/documents/PA3E_DS.pdf

Application Notes

Power-Up/Down Behavior of ProASIC3/E Devices

http://www.actel.com/documents/ProASIC3_E_PowerUp_AN.pdf

Board-Level Considerations

http://www.actel.com/documents/BoardLevelCons_AN.pdf

User's Guides

Actel Libero IDE User's Guide for Software v7.0

http://www.actel.com/documents/libero_ug.pdf

Fusion and ProASIC3/E Macro Library Guide

http://www.actel.com/documents/pa3_libguide_ug.pdf

Appendix

Single-Ended Standards

These I/O standards use a push-pull CMOS output stage with a voltage referenced to system ground to designate logical states. The input buffer configuration, output drive, and I/O supply voltage (V_{CCI}) vary among the I/O standards (Figure 22).

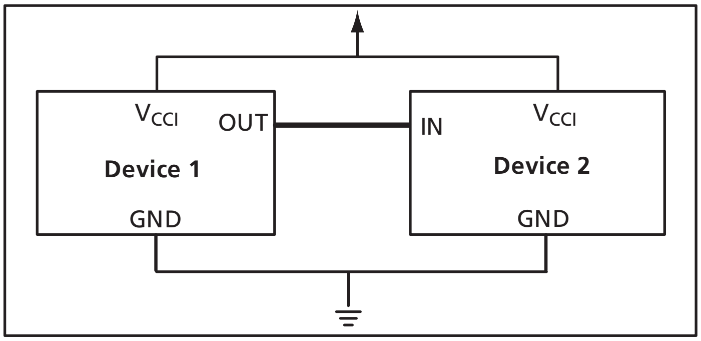


Figure 22 • **Single-Ended I/O Standard Topology**

The advantage of these standards is that a common ground can be used for multiple I/Os. This simplifies board layout and reduces system cost. Their low-edge-rate (*dv/dt*) data transmission causes less electromagnetic interference (EMI) on the board. However, they are not suitable for high frequency (>200 MHz) switching due to noise impact and higher power consumption.

LVTTL (Low-Voltage TTL)

This is a general purpose standard (EIA/JESD8-B) for 3.3 V applications. It uses an LVTTL input buffer and a push-pull output buffer. The LVTTL output buffer can have up to seven different programmable drive strengths; ProASIC3 has six drive strengths. The default drive strength is 12 mA. V_{CCI} is 3.3 V. Refer to "I/O Features" on page 6 for details.

LVCMOS (Low-Voltage CMOS)

The ProASIC3/E family provides four different kinds of LVCMOS: LVCMOS 3.3 V, LVCMOS 2.5 V, LVCMOS 1.8 V, and LVCMOS 1.5 V. LVCMOS 3.3 V is an extension of the LVCMOS standard (JESD8-B–compliant) used for general purpose 3.3 V applications. LVCMOS 2.5 V is an extension of the LVCMOS standard (JESD8-5–compliant) used for general purpose 2.5 V applications. LVCMOS 2.5 V for the A3P030 device has a clamp diode to V_{CCI}, but for all other ProASIC3/E devices there is no clamp diode.

There is yet another standard supported by ProASIC3/E devices (except A3P030): LVCMOS 2.5/5.0 V. This standard is similar to LVCMOS 2.5 V, with the exception that it can support up to 3.3 V on the input side (2.5 V output drive). LVCMOS 1.8 V is an extension of the LVCMOS standard (JESD8-7–compliant) used for general purpose 1.8 V applications. LVCMOS 1.5 V is an extension of the LVCMOS standard (JESD8-11–compliant) used for general purpose 1.5 V applications. The V_{CCI} values for these standards are 3.3 V, 2.5 V, 1.8 V, and 1.5 V, respectively. All these versions use a 3.3 V–tolerant CMOS input buffer and a push-pull output buffer. Like LVTTL, the output buffer has up to seven different programmable drive strengths (2, 4, 6, 8, 12, 16, and 24 mA). Refer to "I/O Features" on page 6 for details.

3.3 V PCI (Peripheral Component Interface)

This standard specifies support for both 33 MHz and 66 MHz PCI bus applications. It uses an LVTTL input buffer and a push-pull output buffer. With the aid of an external resistor, this I/O standard can be 5 V–compliant for ProASIC3/E devices. It does not have programmable drive strength.

3.3 V PCI-X (Peripheral Component Interface Extended)

An enhanced version of the PCI specification, 3.3 V PCI-X can support higher average bandwidths; it increases the speed that data can move within a computer from 66 MHz to 133 MHz. It is backward compatible, which means that devices can operate at conventional PCI frequencies (33 MHz and 66 MHz). PCI-X is more fault tolerant than PCI. It also does not have programmable drive strength.

Voltage-Referenced Standards

I/Os using these standards are referenced to an external reference voltage (V_{REF}) and are supported on ProASIC3E devices only.

HSTL Class I and II (High-Speed Transceiver Logic)

These are general purpose, high-speed 1.5 V bus standards (EIA/JESD 8-6) for signaling between integrated circuits. The signaling range is 0 V to 1.5 V, and signals can be either single-ended or differential. HSTL requires a differential amplifier input buffer and a push-pull output buffer. The reference voltage (V_{REF}) is 0.75 V. These standards are used in the memory bus interface with data switching capability of up to 400 MHz. The other advantages of these standards are low power and fewer EMI concerns.

HSTL has four classes, of which ProASIC3/E devices support Class I and II. These classes are defined by standard EIA/JESD 8-6 from the Electronic Industries Alliance (EIA):

- Class I – Unterminated or symmetrically parallel terminated
- Class II – Series terminated
- Class III – Asymmetrically parallel terminated
- Class IV – Asymmetrically double-parallel terminated

SSTL2 Class I and II (Stub Series Terminated Logic 2.5 V)

These are general purpose 2.5 V memory bus standards (JESD 8-9) for driving transmission lines, designed specifically for driving the DDR SDRAM modules used in computer memory. SSTL2 requires a differential amplifier input buffer and a push-pull output buffer. The reference voltage (V_{REF}) is 1.25 V.

SSTL3 Class I and II (Stub Series Terminated Logic 3.3 V)

These are general purpose 3.3 V memory bus standards (JESD 8-8) for driving transmission lines. SSTL3 requires a differential amplifier input buffer and a push-pull output buffer. The reference voltage (V_{REF}) is 1.5 V.

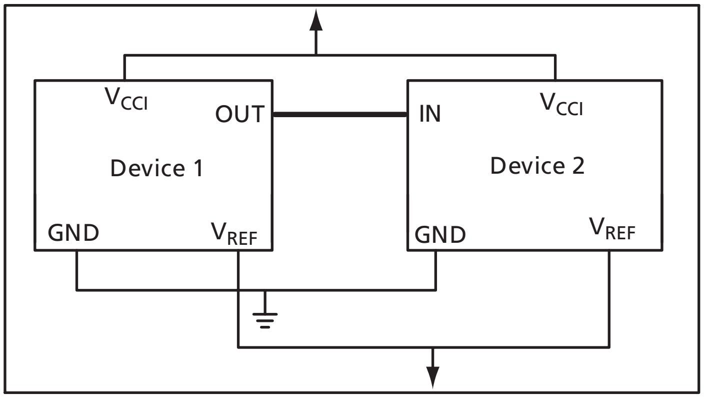


Figure 23 • **SSTL and HSTL Topology**

GTL 2.5 V (Gunning Transceiver Logic 2.5 V)

This is a low power standard (JESD 8.3) for electrical signals used in CMOS circuits that allows for low electromagnetic interference at high transfer speeds. It has a voltage swing between 0.4 V and 1.2 V and typically operates at speeds of between 20 and 40 MHz. V_{CCI} must be connected to 2.5 V. The reference voltage (V_{REF}) is 0.8 V.

GTL 3.3 V (Gunning Transceiver Logic 3.3 V)

This is the same as GTL 2.5 V above, except V_{CCI} must be connected to 3.3 V.

GTL+ (Gunning Transceiver Logic Plus)

This is an enhanced version of GTL that has defined slew rates and higher voltage levels. It requires a differential amplifier input buffer and an open-drain output buffer. Even though the output is open-drain, V_{CCI} must be connected to either 2.5 V or 3.3 V. The reference voltage (V_{REF}) is 1 V.

Differential Standards

These standards require two I/Os per signal (called a "signal pair"). Logic values are determined by the potential difference between the lines, not with respect to ground. This is why differential drivers and receivers have much better noise immunity than single-ended standards. The differential interface standards offer higher performance and lower power consumption than their single-ended counterparts. Two I/O pins are used for each data-transfer channel. Both differential standards require resistor termination.

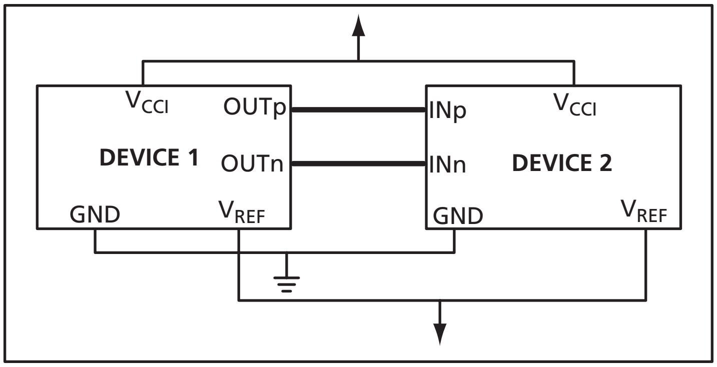


Figure 24 • **Differential Topology**

LVPECL (Low-Voltage Positive Emitter Coupled Logic)

LVPECL requires that one data bit be carried through two signal lines; therefore, two pins are needed per input or output. It also requires external resistor termination. The voltage swing between the two signal lines is approximately 850 mV. When the power supply is +3.3 V, it is commonly referred to as Low-Voltage PECL (LVPECL). Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for the full implementation of the LVPECL transmitter and receiver.

LVDS (Low-Voltage Differential Signal)

LVDS is a moderate-speed differential signaling system, in which the transmitter generates two different voltages that are compared at the receiver. LVDS uses a differential driver connected to a terminated receiver through a constant impedance transmission line. It requires that one data bit be carried through two signal lines; therefore, the user will need two pins per input or output. It also requires external resistor termination. The voltage swing between the two signal lines is approximately 350 mV. V_{CCI} is 2.5 V. ProASIC3/E devices contain dedicated circuitry supporting a high-speed LVDS standard that has its own user specification. Refer to the *ProASIC3 datasheet* and the *ProASIC3E datasheet* for the full implementation of the LVDS transmitter and receiver.

V_{REF} Rules for the Implementation of Voltage-Referenced I/O Standards

The V_{REF} rules are as follows:

1. Any I/O (except JTAG I/Os) can be used as a V_{REF} pin.
2. One V_{REF} pin can support up to 15 I/Os. It is recommended, but not required, that eight of them be on one side and seven on the other side (in other words, all 15 can still be on one side of V_{REF}).
3. SSTL3 (I) and (II): Up to 40 I/Os per north or south bank in any position
4. LVPECL / GTL+ 3.3V / GTL 3.3V: Up to 48 I/Os per north or south bank in any position
5. SSTL2 (I) and (II) / GTL+ 2.5V / GTL 2.5V: Up to 72 I/Os per north or south bank in any position.
6. V_{REF} minibanks partition rule: Each I/O bank is physically partitioned into V_{REF} minibanks. The V_{REF} pins within a V_{REF} minibank are interconnected internally, and consequently, only one V_{REF} voltage can be used within each V_{REF} minibank. If a bank does not require a V_{REF} signal, the V_{REF} pins of that bank are available as user I/Os.

   The first V_{REF} minibank includes all I/Os starting from one end of the bank to the first power triple and eight more I/Os after the power triple. Therefore, the first V_{REF} minibank may contain (0 + 8), (2 + 8), (4 + 8), (6 + 8), or (8 + 8) I/Os.

   The second V_{REF} minibank is adjacent to the first V_{REF} minibank and contains eight I/Os, a power triple, and eight more I/Os after the triple. An analogous rule applies to all other V_{REF} minibanks but the last.

   The last V_{REF} minibank is adjacent to the previous one but contains eight I/Os, a power triple, and all I/Os left at the end of the bank. This bank may also contain (8 + 0), (8 + 2), (8 + 4), (8 + 6), or (8 + 8) available I/Os.

Example:

4 I/Os → Triple → 8 I/Os, 8 I/Os → Triple → 8 I/Os, 8 I/Os → Triple → 2 I/Os

i.e., minibank A = (4 + 8) I/Os, minibank B = (8 + 8) I/Os, minibank C = (8 + 2) I/Os

www.actel.com

Actel Corporation

2061 Stierlin Court
Mountain View, CA
94043-4655 USA

Phone 650.318.4200
Fax 650.318.4600

Actel Europe Ltd.

Dunlop House, Riverside Way
Camberley, Surrey GU15 3YL
United Kingdom

Phone +44 (0) 1276 401 450
Fax +44 (0) 1276 401 490

Actel Japan
www.jp.actel.com

EXOS Ebisu Bldg. 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan

Phone +81.03.3445.7671
Fax +81.03.3445.7668

Actel Hong Kong
www.actel.com.cn

Suite 2114, Two Pacific Place
88 Queensway, Admiralty
Hong Kong

Phone +852 2185 6460
Fax +852 2185 6488