ProASIC^{PLUS} Flash FPGAs I/O and LVPECL Input Pads

Introduction

Actel's ProASIC^{PLUS} family combines the advantages of ASICs with the benefits of reprogrammability and is live at power up through nonvolatile Flash technology. This enables engineers to create high-density systems using existing ASIC or FPGA design flows and tools. To meet the demands of complex and high-performance systems, ProASIC^{PLUS} I/O cells provide flexible I/O features and dedicated LVPECL input pads to enhance and simplify system-level designs.

This application note provides guidance in using ProASIC^{PLUS} flexible I/O features and techniques in interfacing with LVPECL (low voltage positive emitter coupled logic) input pads to achieve better overall system performance.

I/O Standards and Features

Depending on the value of the I/O supply voltage (V_{DDP}), each I/O cell can be configured to comply with different I/O standards. In addition, different I/O features such as low power mode, internal pull-up resistors, Schmitt trigger for inputs, and selectable drive strength and slew rate, are supported to simplify system-level and board-level design. Table 1 shows that when V_{DDP} = 3.3V, each I/O cell can be configured individually to comply with 3.3V LVTTL/LVCMOS/PCI and 2.5V I/O standards. Additionally, when V_{DDP} = 2.5V, it is in compliance with the 2.5V I/O standard as specified by JESD8-5. Threshold values for each I/O standard are listed in Table 2 on page 2.

For detailed electrical specifications and test conditions, please refer to the *ProASIC^{PLUS} Flash Family FPGAs* datasheet and EIA JEDEC website at www.jedec.org.

The flexibility of ProASIC^{PLUS} I/Os gives users a multitude of possible ways to drive the internal resources of the devices. For example, regular I/O macros can be used to drive global clock resources, global input pins can be used to drive local routing resources, and LVPECL inputs can also be used to drive local routing resources. All of these possible combinations are summarized in Table 3 on page 2. For detailed information about driving the device PLL clock conditioning circuit, please refer to Actel's *Using ProASIC^{PLUS} Clock Conditioning Circuits* application note.

Table 1 • *Supported I/O Features*

Features	V_{DDP} = 2.5V				V_{DDP} = 3.3V			
	IB[1]	OB[1]	IOB[1]	GL[1]	IB[1]	OB[1]	IOB[1]	GL[1]
2.5V[2]	✔	✔	✔	✔	✔	✔	✔	✔
3.3V PCI					✔	✔	✔	✔
3.3 V LVTTL/LVCMOS					✔	✔	✔	✔
Internal Pull-up Resistor	✔		✔	✔	✔		✔	✔
Selectable Slew Rate Control		✔	✔			✔	✔	
Selectable Driving Strength		✔	✔			✔	✔	
Schmitt Trigger Input[3]	✔				✔			
2.5V Input Tolerance	✔				✔			
3.3V Input Tolerance					✔			

Notes:

1. *IB – Input Buffer; OB – Output Buffer; IOB – Bidirectional Buffer; GL – Global / Global Multiplexed Input Buffer*
2. *When V_{DDP}=3.3V, 2.5V is generated by internal voltage-shifters.*
3. *The Schmitt trigger input has a typical hysteresis of about ±0.3V.*
   Maximum Rise/Fall Time on Inputs in Schmitt Mode: t_R/t_F=100 ns; Non-Schmitt Mode: t_R/t_F=10 ns

Table 2 • I/O Standards Specification

V_{DDP}	Supported Standard	V_{IH}		V_{IL}		V_{OH}	V_{OL}
		Min.	Max.	Min.	Max.	Min.	Max.
3.3V	LVTTL JESD 8-A	2.0	V_{DDP}+ 0.3	–0.3	0.8	2.4	0.4
	LVCMOS JESD 8-A	2.0	V_{DDP}+ 0.3	–0.3	0.8	V_{DDP}-0.2	0.2
	PCI Rev. 2.2	0.5 V_{DDP}	V_{DDP}+ 0.5	–0.5	0.3V_{DDP}	0.9V_{DDP}	0.1V_{DDP}
3.3V/2.5V	2.5V JESD 8-5	1.7	V_{DDP}+0.3	–0.3	0.7	2.0	0.4

Table 3 • Summary of Input Connectivity Options

Input Type[1]	Internal Resource[2]	Macro[3]
I/O	Routing	IB(x)/IOB(x)
I/O	Global	IB(x)/IOB(x) + GLINT
GL	Routing	IB(x)/IOB(x)
GL	Global	GL(x)/GLIB(x)/GLMIB(x)
GLMX	Routing	IB(x)/IOB(x)
GLMX	Global	GLMX
GLMX	PLL External Feedback	IB(x)/IOB(x)
PECL	Routing	GLPEMIB
PECL	Global	GLPE

Notes:

1. *Refer to the "Pin Description" section of the* ProASIC^PLUS Flash Family FPGAs *datasheet for input descriptions.*
2. *Defines whether the input will drive normal routing resources, a global network, or an external feedback to a PLL.*
3. *Indicates the type of macro to establish this configuration.*

Implementing I/O Standards

ProASIC^PLUS I/O cells can be configured as input, output, bidirectional or global input buffers. The supported I/O standards and features can be defined by the designer through the use of specific I/O macros. A complete list of macros and their descriptions can be found in Actel's *ProASIC^PLUS Macro Library Guide*.

In a schematic design flow, individual I/O buffers must be instantiated in the design. In the HDL-schematic design flow or pure HDL design flow, synthesis tools can automatically add the I/O buffers to the design. For example, by default Synplify inserts IB33 for input ports, OB33PH for output ports, and IOB33PH for bidirectional ports. Users can also instantiate other I/O macros in Verilog or VHDL code by using the following examples.

Using I/O Buffers in Verilog

```
module testand1 (AA, BB, QQ);
      input AA, BB;
      output QQ;
      wire aa_s, bb_s, qq_s;
IB25 u1 ( .PAD(AA), .Y(aa_s));
IB25 u2 ( .PAD(BB), .Y(bb_s));
assign qq_s = aa_s & bb_s;
OB25LL u3 ( .A(qq_s), .PAD(QQ));
endmodule
```

Using I/O Buffers in VHDL

```
library IEEE;
library APA;
use IEEE.std_logic_1164.all;
entity test is
port ( AA, BB : in std_logic;
      QQ : out std_logic);
end test;
architecture arch of test is
signal aa_s,bb_s,qq_s : std_logic;
component IB25
      port (Y :outstd_ulogic;
      PAD : in std_ulogic);
end component;
component OB25LL
      port ( PAD : out std_ulogic;
      A : in std_ulogic);
```

```
end component;
begin
      u1 : IB25
      port map ( PAD => AA, Y => aa_s);
      u2 : IB25
      port map ( PAD => BB, Y => bb_s);
      qq_s <= (aa_s and bb_s);
      u3 :OB25LL
      port map (A =>qq_s, PAD => QQ);
end arch;
```

LVPECL Input Pads

As today's demand for high-speed data transmission grows, differential I/O standards are used for high-performance, good noise immunity, and low power data transfer. LVPECL offers the characteristics of tight timing accuracy, well-balanced differential signals, low-level signals with low generated noise and power consumption that remains nearly constant with data rates. The LVPECL differential I/O standard requires data that is transferred using two signal wires. If noise is coupled onto the signal wires as a common mode, it will be rejected by the receiver. The receiver only responds to a differential voltage with input specifications defined in Table 4 between the two signal lines.

In ProASIC^PLUS there are two LVPECL differential input pairs to accommodate high-speed clock and data inputs. These are dedicated high-speed differential inputs that require no pull-ups. Table 4 shows the LVPECL input specification.

Using LVPECL Macros

In order to employ ProASIC^PLUS LVPECL input pads, users are required to instantiate either GLPEMIB or GLPE macros in their design (Figure 1). The GLPE and GLPEMIB macros read the difference between PPECL (I/P) and NPECL analog signals and return logic '1' if it is above a threshold. In the GLPE macro, the LVPECL input cannot be used to drive regular logic, but for the GLPMIB macro, the LVPECL input can be used to drive both regular logic and a global clock network at the same time by setting EN = '0.'

Table 4 • LVPECL Input Specification

Symbol	Parameter	Minimum	Maximum	Units
V_{IH}	Input High Voltage	1.49	2.72	V
V_{IL}	Input Low Voltage	0.86	2.125	V
V_{ID}	Differential Input Voltage	0.3	V_{DD}	V

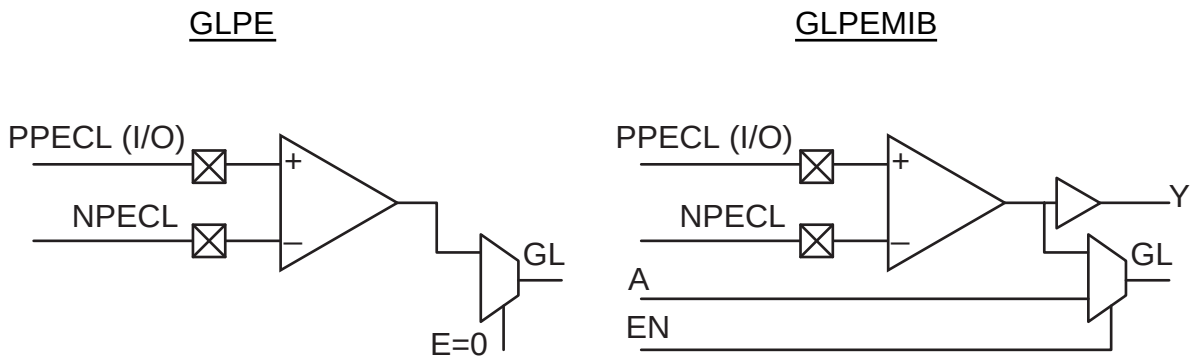

Figure 1 • GLPEMIB and GLPE Macros

GLPE Macro Instantiation VHDL

```
component GLPE
port     (GL : out std_logic;
       PECLIN : in std_logic;
       PECLREF : in std_logic);
end component;
component GLPEMIB
port     (GL : out std_logic;
       Y : out std_logic;
       A : in std_logic;
       EN : in std_logic;
       PECLIN : in std_logic;
       PECLREF : in std_logic);
end component;
```

GLPE Macro Instantiation Verilog

```
module GLPE(GL, PECLIN, PECLREF);
  output GL;
  input  PECLIN, PECLREF;
endmodule
```

GLPEMIB Macro Instantiation VHDL

```
component GLPEMIB
port     (GL : out std_logic;
       Y : out std_logic;
       A : in std_logic;
       EN : in std_logic;
       PECLIN : in std_logic;
       PECLREF : in std_logic);
end component;
```

GLPEMIB Macro Instantiation Verilog

```
module GLPEMIB(GL, Y, A, EN, PECLIN, PECLREF);
  output GL, Y;
  input  PECLIN, PECLREF, A, EN;
endmodule
```

Board-Level Considerations

In order to achieve high-speed data transmission, a proper termination technique is required when interfacing with LVPECL input pairs to avoid reflection and to reduce electromagnetic emission.

Figure 2 shows the recommended circuitry to interface LVDS with ProASIC^PLUS LVPECL input pairs. The resistor network shifts the LVDS output from 1.2V to around 1.7V LVPECL input for optimal performance. Figure 2 lists the recommended configurations, but users should test this on their board.

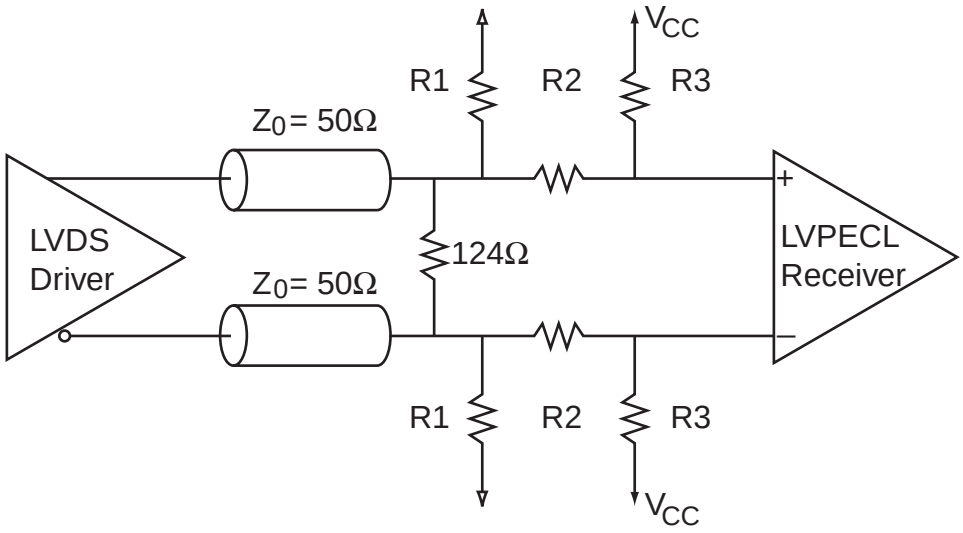

Figure 2 • LVDS to LVPECL Interface

When interfacing the LVPECL transmitter that meets the JEDEC standard with ProASIC^PLUS LVPECL receiver, only a 100Ω resistor is required for termination (Figure 3).

For regular user I/Os, if a signal is propagated at a very high speed through a long trace on the PCB without proper termination, a reflection would disturb the integrity of the signal. This may cause the signal level to transition into an unexpected state. In general, series and parallel termination are two basic ways to terminate the transmission line. Serial termination (Figure 4) consists of a resistor on the driver side that is equal to the impedance of the wire. Serial termination does not consume power but is only practical for a single point-to-point interconnection between the driver and the receiver.

In parallel termination, a resistor matching the line impedance is placed on the receiver side. Energy from the reflected wave is absorbed and steady-state level is established. Figure 5 shows different parallel termination configuration.

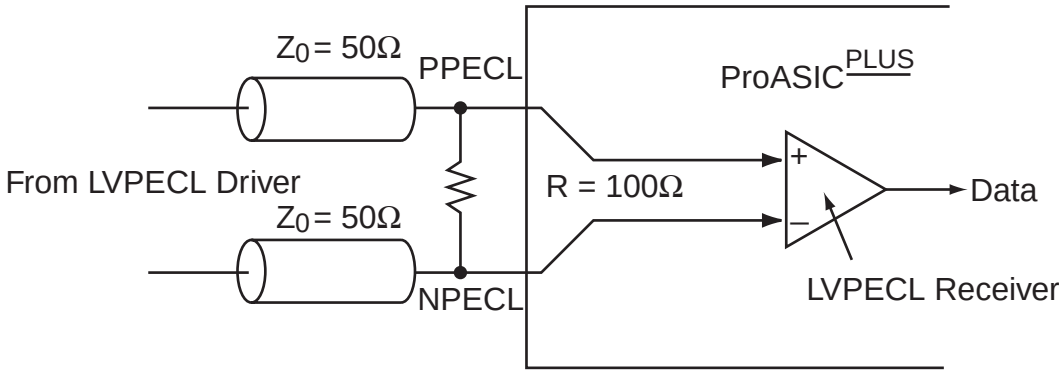

Figure 3 • LVPECL to LVPECL Interface

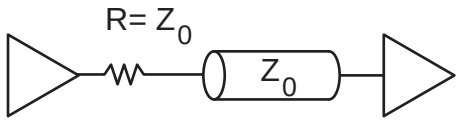

Figure 4 • Serial Termination of Transmission Line

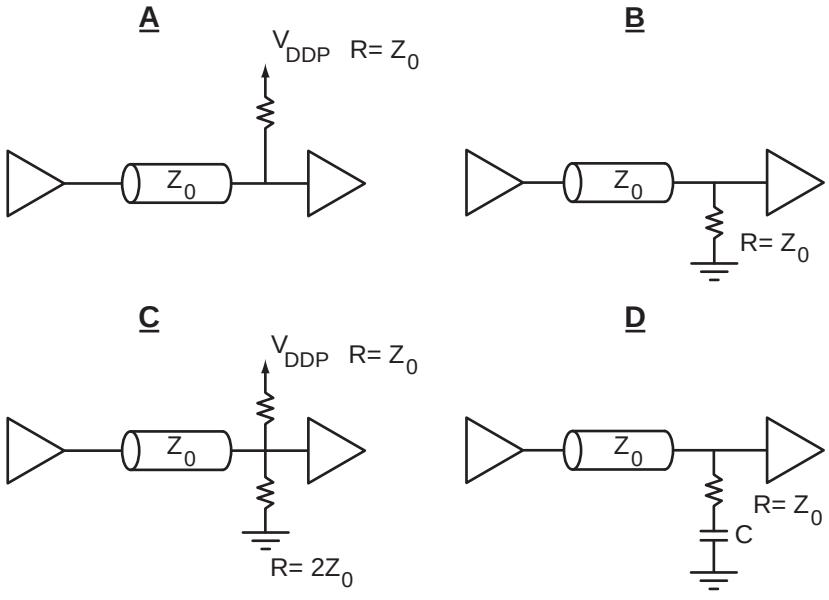

Figure 5 • Parallel Termination of Transmission Line

I/O Placement

Actel's Designer software's layout algorithm is optimized to place the I/Os for maximum routability and performance. Actel recommends that you let Designer software automatically assign I/O locations during layout. If you must manually assign the I/O locations, you can assign I/O locations in PinEdit (from Actel's Designer software) or in a GCF file (ProASIC^PLUS constraint file) that you import into Designer software as described in the following sections.

Using PinEdit

The PinEdit tool provides a graphical interface that allows users to set I/O placements (Figure 6). Prior to layout, you can use PinEdit to place and fix I/Os onto particular pins and Designer software will maintain those fixed pins during layout. Designer software will automatically assign pin locations for all unplaced I/Os for optimal performance. After the layout, automatically assigned I/Os can still be changed using PinEdit. Please refer to the *PinEdit* user's guide for more information on using PinEdit.

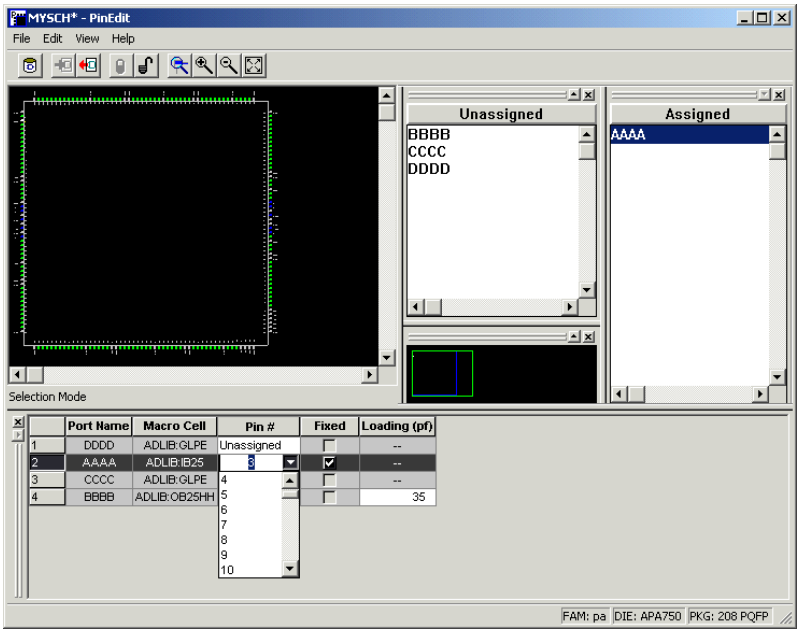

Figure 6 • *Pin Edit in GUI Designer Software*

Using GCF Constraint Files

After a netlist is imported into Designer software, import a GCF file (a constraint file for ProASIC^PLUS) that specifies the pin assignments. In the GCF file, "set_initial_io" and "set_io" are the options that the users can select for I/O placement. Both "set_io" and "set_initial_io" statements are used to assign package pins to I/O ports. However, "set_io" is a hard constraint and cannot be overruled by Designer. This may impact the timing result on the design as the routability may be restricted. If this is not suitable, use "set_initial_io" so the Designer can reassign or relocate the cells during layout. For more information about using different constraints in a GCF file, please refer to the *Designer* user's guide.

Syntax:

```
set_io  package_pin  io_port_name;
set_initial_io  package_pin  io_port_name;
```

For example:

```
set_io A11 in2;
set_initial_io A12 in3;
```

Unused I/Os and Special Pins

In ProASIC^PLUS, unused I/Os are automatically configured by Designer software as inputs with a pull-up resistor (IB33U or IB25LPU). Designers can leave the unused I/Os floating or terminate them to V_{DDP} or GND.

If LVPECL input pads are not used, PPECL (I/P) and NPECL pins can be left floating. Also, if PLL core is not employed, users can either leave both AVDD and AGND pins floating or connect AVDD to 2.5V and AGND to 0V.

I/O Reliability

All ProASIC^PLUS I/O cells include ESD (2000V per MIL-STD-883, Method 3015) and latch-up protection circuitry. Please visit Actel's website frequently for updates on detailed reliability information.

Multiple outputs can be tied together to increase the drive strength. However, it is important to balance the delay path for multiple outputs that are tied together to minimize current and prevent contention of the outputs during switching.

Source and Sink Currents

Maximum drive strength and complete I-V curves can be derived from Actel's IBIS models. These models, combined with instructions for calculating max source and sink current, can be obtained from Actel's website at:
http://www.actel.com/custsup/models/ibis.html

Conclusion

Actel's ProASIC^PLUS was designed to meet various design needs. Flexible I/O features, combined with high-speed LVPECL inputs, simplify board-level design and enhance overall performance. Its reprogrammability and live at power-up capabilities make ProASIC^PLUS an excellent choice for challenging designs.

http://www.actel.com

Actel Europe Ltd.
Maxfli Court, Riverside Way
Camberley, Surrey GU15 3YL
United Kingdom
Tel: +44 (0)1276 401450
Fax: +44 (0)1276 401490

Actel Corporation
955 East Arques Avenue
Sunnyvale, California 94086
USA
Tel: (408) 739-1010
Fax: (408) 739-1540

Actel Asia-Pacific
EXOS Ebisu Bldg. 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan
Tel: +81-(0)3-3445-7671
Fax: +81-(0)3-3445-7668

5192720-0/11.02