A parallel input/output (PIO) module is a convenient memory-mapped interface between a Nios™ CPU and user-defined logic. Each PIO is generated by a MegaWizard® Plug-In in the Quartus™ II software and may interface up to 32 bits. A complete system may contain any number of PIO modules-limited only by the capacity of the target Altera® device.

This module has two distinct applications:

- Providing a PIO interface between the Nios processor and user-defined logic that also resides within the same device.
- Providing a PIO interface between the Nios processor and user peripheral logic that resides outside the device.

By default, the PIO block has only one internal address (Data-in/Data-out). The other two registers only can be accessed if the $PIO_TRISTATE_PINS or $PIO_INTERRUPT parameters are set to YES.
Figure 2. Nios Embedded Tri-State PIO

Figure 3. Nios Embedded Input-Only PIO

Figure 4. Nios Embedded Output-Only PIO
PIO Connections Within a Device

Every bit in the PIO block is associated with an input register and output register. Writing to a PIO output register sets a value on the PIO output and reading from a PIO input register reads the current value from the PIO inputs. The input and output connections are distinct.

By default, the PIO block has both input and output pins. The parameter $PIO_INPUT_PINS controls the input connections and the parameter $PIO_OUTPUT_PINS controls the output connections. These parameters accept true or false as values.

PIO Connections Outside a Device

The PIO block can have bidirectional pins instead of separate input and output pins, i.e., the outputs can only connect to external device pins. To use the PIO block with tri-state buffers in devices, the $PIO_TRISTATE_PINS parameter should be set to true at compile-time. In this case, the PIO register set also includes a data-direction register for software control of the output drivers.

If you set the $PIO_TRISTATE_PINS parameter to true, this module’s in_port and out_port connections will be replaced by a single n-bit connection named bidir_port. It is the designer’s responsibility to connect all the individual bits of bidir_port to an I/O port on the top-level design.

Variable Width

The PIO block width provides between 1 and 32 input pins and between 1 and 32 output pins. These I/O pins may also be configured to provide a bidirectional interface.
**Nios Embedded Processor Parallel I/O Module**

**Edge Capture**

The PIO edge capture detects a transition on an input to the PIO and sets a corresponding bit in the CPU-readable edge capture register. The edge capture parameter may be set to one of the following values:

- **Rising**—Detects the transition from logical 0 to logical 1.
- **Falling**—Detects the transition from logical 1 to logical 0.
- **Any**—Detects any logic level transition.
- **None**—Detect only the logic level.

None is the default value.

A write-operation to the edge-capture register clears all bits.

**Interrupt Control**

The PIO interrupt control feature may be enabled to create a CPU interrupt. By default, the PIO block does not generate an interrupt, and has no interrupt-control logic or registers. The PIO interrupt parameter can be set to one of the following values:

- **Level**—Generate an interrupt when a PIO input logic level is detected.
- **Edge**—Generate an interrupt when a PIO input logic level transition is detected.

The PIO block includes both an irq-pin to the CPU and an internal interrupt-masking register when the interrupt control feature is used. The edge feature is only available if the PIO edge capture parameter is enabled.

**PIO Software Routines**

If there is one or more PIO peripheral present in the Nios system, the PIO peripheral software routines are available in the Nios library (.lib folder in the custom software development kit).

For more information regarding software routine calls and custom software development kits, please refer to the *Nios Software Development Reference Manual*.

Copyright © 2001 Altera Corporation. All rights reserved.