Introduction to Actel FPGA Architecture

This overview of the different Actel device families covers the major architectural features in sufficient detail to ensure the reader is familiar enough with Actel devices to get the most out of the application notes in the rest of this section. Details on the functionality of each device are available in the individual device data sheets.

FPGA Architecture—User Requirements

Digital system design is becoming more difficult. Systems require ever-increasing complexity and performance, but time-to-market pressures continue to limit development times. System cost is also an important constraint, so a solution must meet stringent financial targets. These competing requirements demand a digital logic design solution optimized to meet all of the capacity, performance, cost, and time-to-market requirements. An optimized architecture is needed to balance all of these competing demands. The Actel architecture meets all of these requirements by providing the correct balance of capacity, performance, cost, and ease of use through its innovative combination of an optimized logic module, abundant interconnect resources, efficient silicon usage, and powerful software design tools.

Actel Device Architecture

The underlying architecture of an Actel FPGA is very similar to that of a conventional gate array. The core of the device consists of simple logic modules used to implement the required logic gates and storage elements. These logic modules are interconnected with an abundance of segmented routing tracks. Unlike gate arrays, the segment lengths are predefined and can be connected with low-impedance switching elements to create the precise routing length required of the interconnect signal. Surrounding the logic core is the interface to the I/O pads of the devices. This interface consists of I/O modules that translate and interconnect the logic signals from the core of the device to the FPGA output pads. A block diagram of a generic Actel FPGA is given in Figure 1.

The major elements of the Actel FPGA architecture are thus the I/O modules, interconnect resources, clocking resources, and logic modules. Each Actel FPGA family has a slightly different mix of these resources, optimized for different cost, performance, and density requirements. Table 1 shows the capabilities of each Actel FPGA family. Each capability is defined in the sections following Table 1.

Logic Module Descriptions

The optimal logic module should provide the user with the correct mix of performance, efficiency, and ease of design required to implement the application. If the logic module provides performance without efficiency, the cost and capacity requirements of the design might not be achieved. Similarly, if cost and capacity targets are achieved at the expense of performance and ease of use, the device may not be usable. The optimal logic module must balance these trade-offs carefully to ensure that the many conflicting goals of the designer are achievable.

Simple Logic Module

The first Actel logic module was the Simple Logic Module used in the ACT 1 family. It is shown in Figure 2. It is a multiplexer-based logic module. Logic functions are implemented by interconnecting signals from the routing tracks to the data inputs and select lines of the multiplexers. Inputs can also be tied to a logical 1 or 0 if required, since these signals are always available in the routing channel.

A surprising number of useful logic functions can be implemented with this module. Clearly, multiplexing is very efficient, but random logic and sequential logic functions are also efficient. These options provide the designer with an excellent mix of logic capabilities, required for applications demanding a variety of logic functions. Figure 3 shows an example logic function implemented with the Actel Simple Logic Module. Notice that latches can be implemented in a single logic module per bit and that registers require two logic modules per bit. The ACT 1 logic module is thus extremely flexible in covering a wide range of combinatorial and sequential logic mixes.

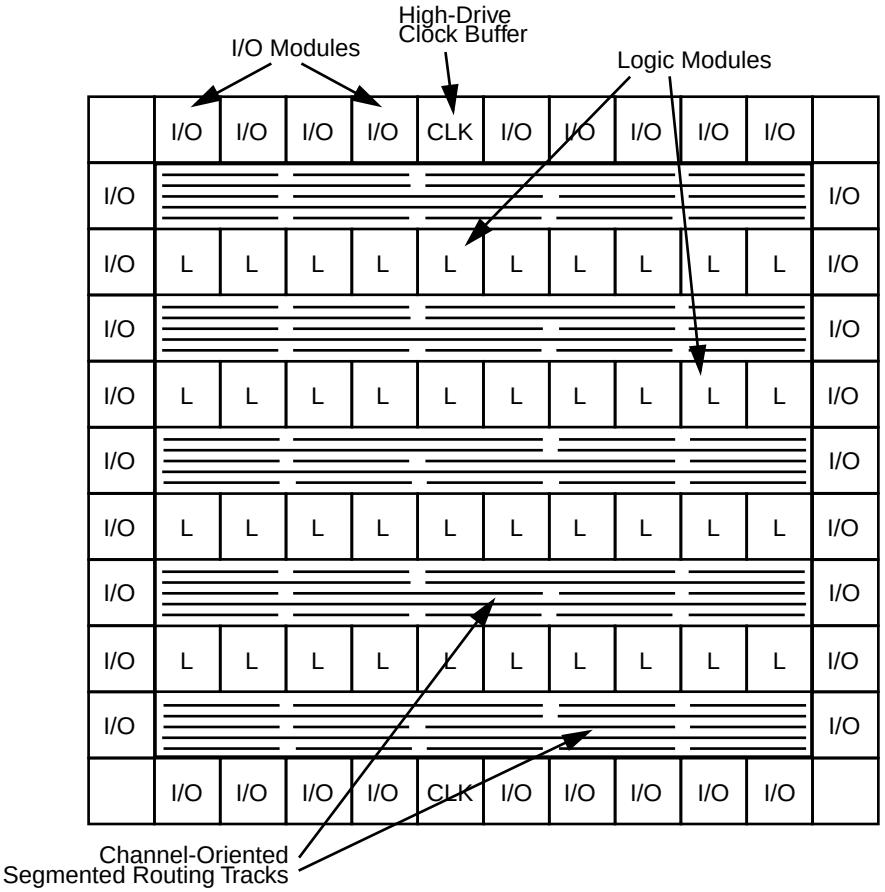

Figure 1 • *Basic Actel FPGA Architecture*

Table 1 • *Quick Summary of Key Architectural Features of Actel FPGA Families*

Capability	**ACT 1**	**ACT 2/1200XL**	**3200DX**	**ACT 3**
Core Module	Simple Logic Module	Combinatorial and Sequential Modules	Combinatorial and Sequential Modules	Combinatorial and Enhanced Sequential Modules
			Wide Decode Modules	
			Embedded Dual-Port SRAM	
Interconnect	Channeled	Channeled	Channeled	Channeled
Clocking Resources	Routed Clock (1)	Routed Clocks (2)	Routed Clocks (2)	Routed Clocks (2)
			Quad Clocks (4)	Dedicated Array Clock
				Dedicated I/O Clock
I/O Module	Simple I/O Module	Latched I/O Module	Latched I/O Module	Registered I/O Module

Combinatorial Logic Module

Some improvements were made to the Simple Logic Module when the second generation ACT 2 family was developed. The Simple Logic Module was replaced with two different logic modules, one for implementing combinatorial logic, (the Combinatorial Logic Module) and one for implementing storage elements (the Sequential Logic Module). The Combinatorial Logic Module, shown in the diagram in Figure 4, is similar to the Simple Logic Module, but an additional logic gate was placed on the first-level multiplexer. The added gate improves the implementation of some combinatorial functions. (Some five-input gates are now available.) Also, the first-level multiplexer lines in the Simple Logic Module were combined in the Combinatorial Logic Module. In the Simple Logic Module, the separate multiplexer select lines were used to implement latches and registers efficiently. This was not required in the Combinatorial Logic Module because of the addition the Sequential Logic Module. Figure 5 shows an example of a logic function implemented with the Combinatorial Logic Module.

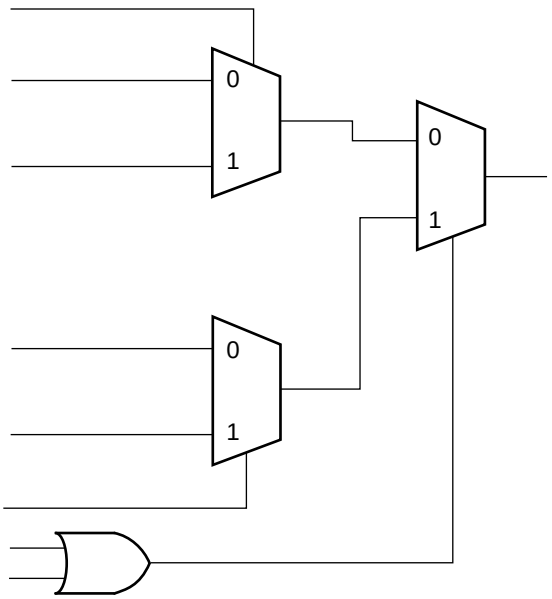

Figure 2 • *Simple Logic Module*

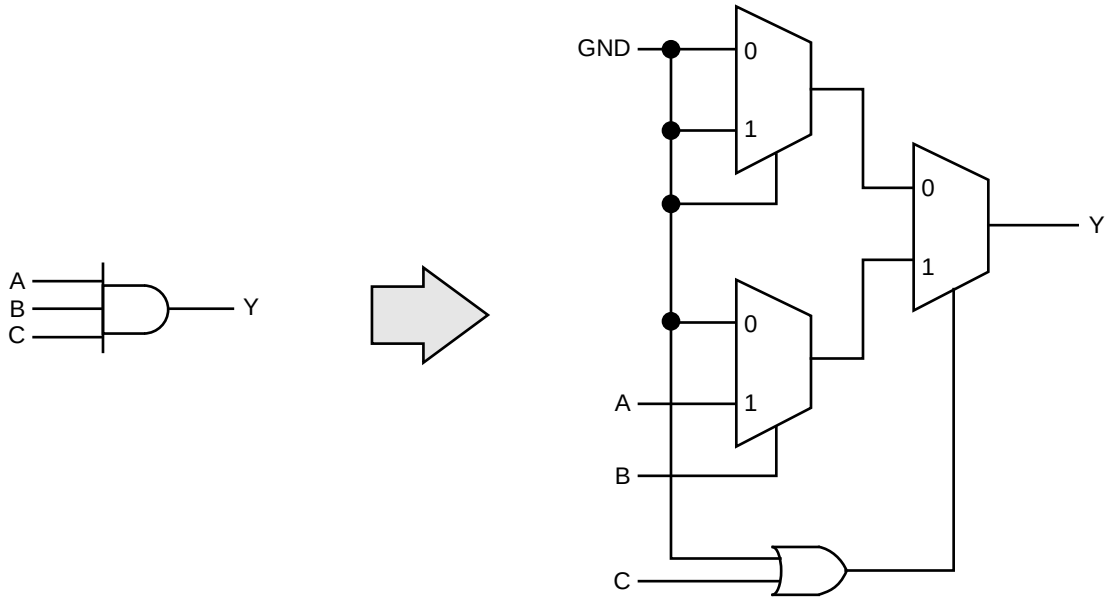

Figure 3 • *Example of a Logic Function Implemented with the ACT 1 Simple Logic Module*

Sequential Logic Module

The Sequential Logic Module, shown in the diagram in Figure 6, has a combinatorial logic front end with a dedicated storage element on the output of the logic module. The storage element can be either a register or a latch. (It can also be bypassed so the logic module can be used as a Combinatorial Logic Module.) The clock input can be selected to be either active high or active low. One of the logic gates is missing on the combinatorial logic section, making it slightly different from the Combinatorial Logic Module. The exclusion of this one logic gate allows the reset signal, which is shared with the combinatorial logic section, to be made available to the storage element without increasing the number of total module inputs required. If the storage element is bypassed, the reset signal is used to implement the required combinatorial module input. In the Integrator Series, sequential and combinatorial modules are interleaved, resulting in a 50-50 mix of logic modules. This

has been determined to be an optimal mix for a wide variety of designs and results in excellent utilization.

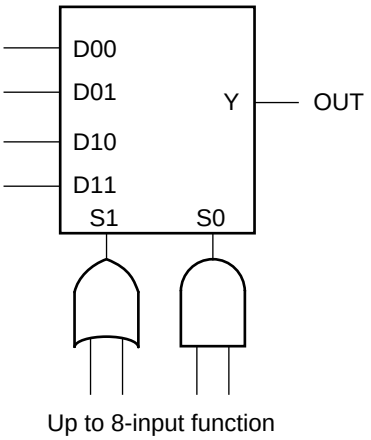

Figure 4 • *Combinatorial Logic Module*

Wide Decode Logic Module

Every member of the 3200DX family has a number of special logic modules optimized for implementing wide-input combinatorial logic functions directly driving device output pads. The Wide Decode Logic Module consists of a seven-input AND gate with selectable inversion on the output. The output of this module bypasses the normal routing network and connects directly to a particular output buffer. This feature minimizes the delay from the module output to the device pad and is ideal for implementing wide-decode functions typically implemented in small PLD devices. The Wide Decode Logic Module output is also available to the core logic modules, so it can be used in conjunction with other logic functions internal to the device. For more details on the Wide Decode Logic Module, see the "Integrator Series FPGAs" data sheet and the "3200DX Wide Decode Modules" application note in this Data Book.

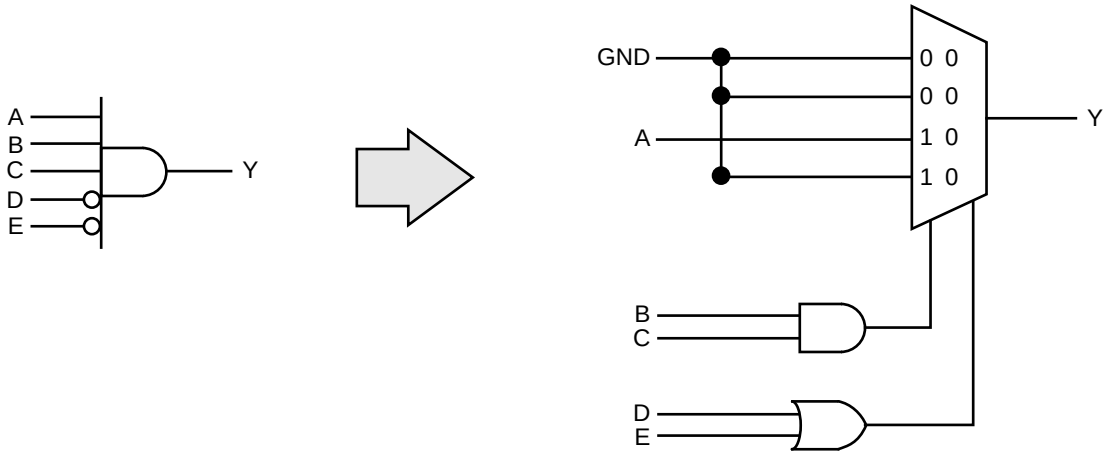

Figure 5 • *Example of a Logic Function Implemented with the Combinatorial Logic Module*

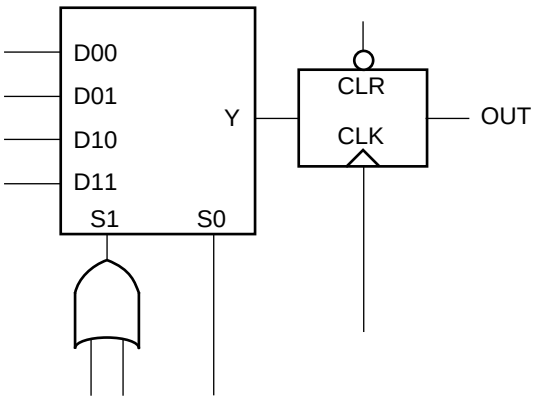

Figure 6 • *Sequential Logic Module*

Embedded Dual-Port SRAM

Some members of the 3200DX family include dedicated blocks of high-speed dual-port SRAM. The SRAM blocks are arranged in 256-bit blocks, which can be configured as 32 x 8 or 64 x 4. SRAM blocks can be cascaded to form wider or deeper memory blocks. The SRAM is dual ported, with separate read and write addresses, a separate data input port (for writes), and a separate data output port (for reads). Reads and writes are controlled by separate clocked read and write enables, easing timing requirements for using the SRAM. The dual-port structure is ideal for implementing FIFO, burst buffers, and internal register storage for status, control, or constant data. The 3200DX devices have from 8 SRAM blocks (on the A32100DX) to 16 SRAM blocks (on the A32400DX). For more detail on the structure and functionality of the dual-port SRAM blocks, see the

"Integrator Series FPGAs" data sheet and the "3200DX Dual-Port Random Access Memory" application note in this Data Book.

Enhanced Sequential Logic Module

The Enhanced Sequential Logic Module used in the ACT 3 family is a refinement of the Sequential Logic Module and is shown in the diagram in Figure 7. The reset input on the register in the sequential section is not shared with the combinatorial logic function, so the full combinatorial logic is available in the Combinatorial Logic Module to be used in front of the register. This makes all single module combinatorial logic functions usable in front of the storage element, simplifying design via schematics or synthesis inputs, and it can result in speed improvements for wide-input functions.

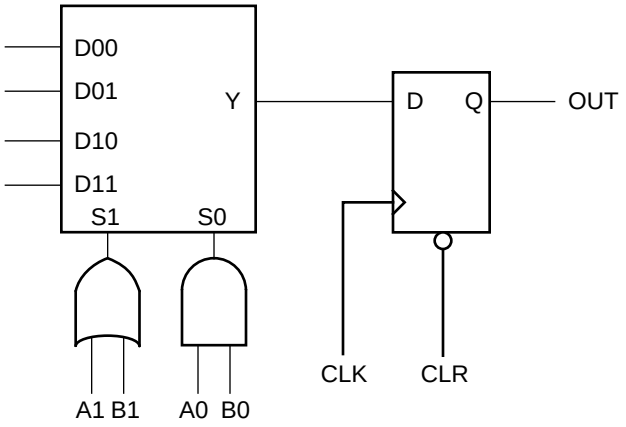

Figure 7 • *S-Module Diagram*

Channeled Interconnect

All Actel devices use a channeled interconnect architecture to make connections between internal logic modules and device I/O pins. This architecture is similar to that of a channeled gate array, in that horizontal tracks span the length of the array with a variety of predefined segmentation lengths. This makes a huge amount of routing resources available and ensures that signals usually have the length of available track that they require. In addition, tracks can be joined together to construct longer tracks, when required, by programming an interconnect fuse. Logic module outputs span four channels (two above and two below) and can be connected to any track. This means that most signals require only two fuses to connect any logic module output to any logic module input. There are enough routing resources available in Actel devices so that place and route is an automatic task. No hand routing is required. For more details on the interconnect resources available in Actel devices, refer to the device family data sheets.

Clocking Resources

Actel devices have a wide range of clocking flexibility. Every sequential element's clock input can be connected to regular interconnects within the channel, as well as to optimized clocking resources. Regular interconnects offer the most flexibility, allowing for thousands of potential separate clocks. Each Actel device also has dedicated clocking resources on-chip to improve clock performance and to simplify the design of sequential signals. Clocking resources can also be used, in most cases, as high-drive global signals like reset, output enable, or select signals. Each FPGA family is slightly different in the way it implements clocking functions. For more details on each type of clocking resource, refer to the associated device data sheets and application notes.

Routed Clocks

All Actel FPGA families have one or two special buffers that provide high-drive, low-skew signals and that can be used to drive any signal requiring these characteristics. These routed clocks are distributed to every routing channel and are available to every logic module. This allows a routed clock signal to be used by both Sequential and Combinatorial Logic Modules, offering maximum flexibility with slightly lower performance than dedicated clocks.

Dedicated Array Clock

The ACT 3 family has an additional clocking resource consisting of a high-speed dedicated clock buffer optimized for driving the sequential modules in the core array. This clock buffer can be driven from an external pin or from an internal signal. The dedicated array clock is optimized for driving sequential modules and can't drive storage elements built from combinatorial modules.

Dedicated I/O Clock

The ACT 3 family has another clocking resource consisting of a high-speed dedicated clock buffer optimized for driving the sequential modules in the I/O modules. The dedicated I/O clock is optimized for driving I/O modules and can't drive storage elements in the array. If all storage elements need to be driven from a common clock, the array clock and I/O clock can be connected together externally.

Quad Clocks

The 3200DX family has an additional clocking resource consisting of four special high-drive buffers called quadrant clocks. Each buffer provides a high-drive signal that spans about one-quarter of the device (a quadrant). These buffers can be used for fast local clocks (perhaps for prescaled shifters or counters), for wide-width mux selects, or for I/O enables. Note that since these are quadrant oriented, only a single quadrant clock can be used per quadrant. Quad clocks can be connected together internally to span up to one-half of

the device. Additionally, the quad clocks can be sourced from internal signals as well as external pins. Thus they can be used as internally driven high-fanout nets. Refer to the "3200DX Quadrant Clocks" application note in this Data Book.

I/O Module Descriptions

Each Actel FPGA family has a slightly different I/O module. The Simple I/O Module, found in the ACT 1 family, is optimized for low cost, and the Latched I/O Module, found in the Integrator Series, offers a balance of speed and cost (value). The Registered I/O Module in ACT 3 is optimized for high speed in synchronous applications. More details on each I/O module can be found in the associated device data sheets and application notes.

Simple I/O Module

The Simple I/O Module (Figure 8) used in the ACT 1 family was the first I/O module developed by Actel and is a simple I/O buffer with interconnections to the logic array. All input, output, and three-state control signals are available to the array. Outputs are TTL and CMOS compatible and sink or source 10 mA at TTL levels.

Latched I/O Module

The Latched I/O Module, shown in the diagram in Figure 9, is used in the Integrator Series and is slightly more complicated than the Simple I/O Module. The Latched I/O Module contains input and output latches that can be used as such or combined with internal latches to construct input or output registers. Outputs are TTL and CMOS compatible and sink or source 10 mA at TTL levels.

Registered I/O Module

The Registered I/O Module, used in the ACT 3 family devices, is optimized for speed and functionality in synchronous system designs. It contains complete registers at both the input and output paths, as shown in the diagram in Figure 10. Data can be stored in the output register (under control of the ODE, output data enable, signal), or it can bypass the register if the OTB control bit is tied low. Both the output register and the input register can be cleared or preset via the global IOPCL signal, and both are clocked via the IOCLK global signal. Notice that the output of the output register can be selected as an input to the array (on the Y signal). This allows state machines, for example, to be built right into the I/O module for fast clock-to-output requirements. Refer to the "Using ACT 3 Family I/O Macros" application note in this Data Book.

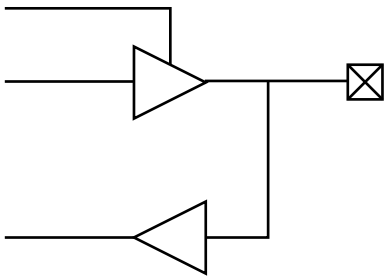

Figure 8 • *Simple I/O Module*

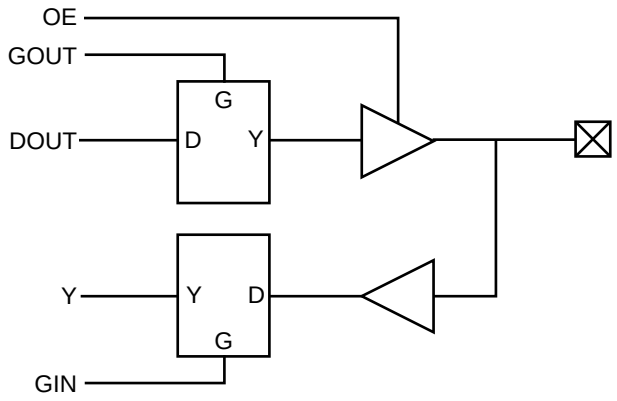

Figure 9 • *Latched I/O Module*

Summary

This introduction has given the reader a quick start to the architecture of Actel FPGA families. The reader can now continue exploring Actel devices and applications by turning to a variety of applications and data sheet resources.

- If you are interested in finding out more about a particular device, consult the appropriate data sheet.
- If you are interested in seeing the application of Actel devices to your particular design problem, or one similar to it, consult the table of contents of the applications sections of the Data Book.
- If you want to see estimates of performance and capacity for real-world functions implemented in Actel devices, turn to the applications sections to find information on estimating performance and capacity of Actel FPGAs.

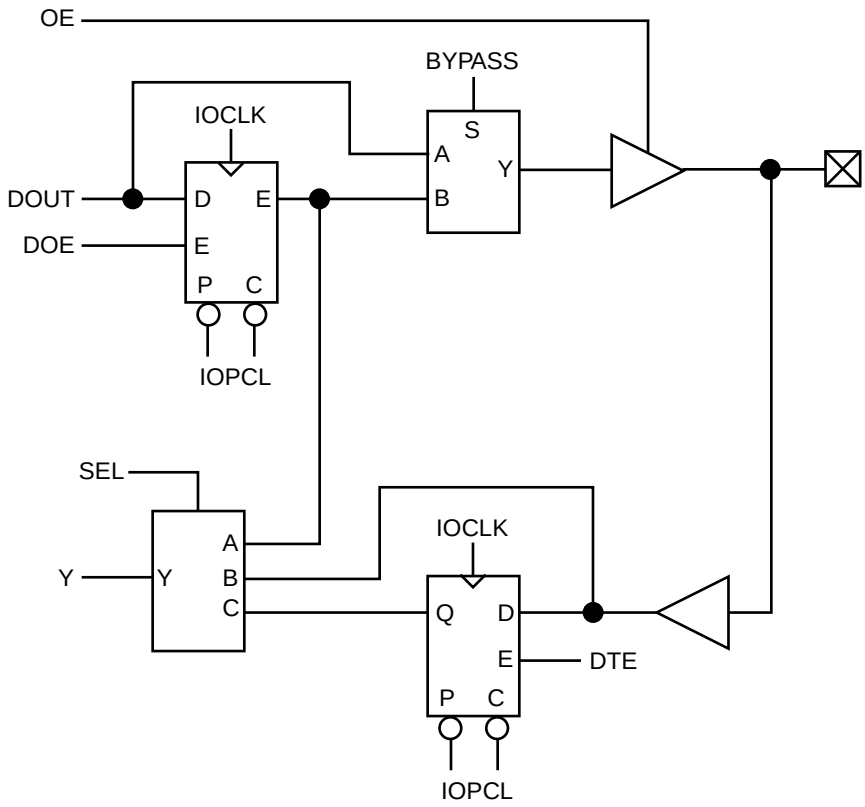

Figure 10 • *Registered I/O Module*