CYPRESS

UltraLogic™ High-Density Flash CPLDs

Features

- **Flash In-System Reprogrammable (ISR™) CMOS CPLDs**
  - **Combines on board reprogramming with pinout flexibility and a simple timing model**
  - **Design changes don't cause pinout or timing changes**
  - **JTAG interface**
- **High density**
  - **32–128 macrocells**
  - **32–128 I/O pins**
  - **Multiple clock pins**
- **Fully PCI compliant**
- **Bus Hold capabilities on all I/Os and dedicated inputs**
- **3.3-V or 5.0-V I/O operation on all devices**
- **High speed**
  - t_{PD} = 8.5–10 ns
  - t_S = 5–7 ns
  - t_{CO} = 6–7 ns
- **Fast Programmable Interconnect Matrix (PIM)**
  - **Uniform predictable delay, independent of routing**
- **Intelligent product term allocator**
  - **0–16 product terms to any macrocell**
  - **Provides product term steering on an individual basis**
  - **Provides product term sharing among local macrocells**
- **Simple timing model**
  - **No fanout delays**
  - **No expander delays**
  - **No dedicated vs. I/O pin delays**
  - **No additional delay through PIM**
  - **No penalty for using full 16 product terms**
  - **No delay for steering or sharing product terms**
- **Flexible clocking**
  - **2–4 clock pins per device**
  - **Clock polarity control**
- **Security bit and user ID supported**
- **Packages**
  - **44–160 pins**
  - **PLCC, CLCC, PGA, CQFP, and TQFP packages**

General Description

The FLASH370i™ family of CMOS CPLDs provides a range of high-density programmable logic solutions with unparalleled performance. Each member of the family is designed with Cypress's state-of-the-art 0.65-micron Flash technology.

All of the UltraLogic FLASH370i devices are electrically erasable and In-System Reprogrammable (ISR), which simplifies both design and manufacturing flows, thereby reducing costs. Because of the superior routability of the FLASH370i devices, ISR allows users to change existing logic designs without changing pinout assignments or timing. The Cypress ISR function is implemented through a JTAG serial interface. Data is shifted in and out through the SDI and SDO pins, respectively. The ISR interface is enabled from the programming voltage pin (ISR_{EN}). The entire family is fully compliant with the PCI Local Bus specification, meeting all the electrical and timing requirements. Also, the entire family features bus-hold capabilities on all I/Os and dedicated inputs. Additionally, the entire family is security bit and user ID supported (when the security bit is programmed, all locations cannot be verified).

FLASH370i Selection Guide

Device	Pins	Macrocells	Dedicated Inputs	I/O Pins	Flip-Flops	Speed (t_{PD})	Speed (f_{MAX})
371i	44	32	5	32	44	8.5	143
372i	44	64	5	32	76	10	125
373i	84/100	64	5	64	76	10	125
374i	84/100	128	5	64	140	10	125
375i	160	128	5	128	140	10	125

General Description (continued)

The FLASH370i family is designed to bring the flexibility, ease of use and performance of the 22V10 to high-density CPLDs. The architecture is based on a number of logic blocks that are connected by a Programmable Interconnect Matrix (PIM). Each logic block features its own product term array, product term allocator array, and 16 macrocells. The PIM distributes signals from the logic block outputs and all input pins to the logic block inputs.

The family features a wide variety of densities and pin counts to choose from. At each density there are two packaging options to choose from—one that is I/O intensive and another that is register intensive. For example, the CY7C374i and CY7C375i both feature 128 macrocells. On the CY7C374i, available in 84-pin packages, half of the macrocells are buried. On the CY7C375i, available in 160-pin packages, all of the macrocells are fed to I/O pins. *Figure 1* shows a block diagram of the CY7C374i/5i.

Functional Description

Programmable Interconnect Matrix

The Programmable Interconnect Matrix (PIM) consists of a completely global routing matrix for signals from I/O pins and feedbacks from the logic blocks. The PIM is an extremely robust interconnect that avoids fitting and density limitations. Routing is automatically accomplished by software and the propagation delay through the PIM is transparent to the user. Signals from any pin or any logic block can be routed to any or all logic blocks.

The inputs to the PIM consist of all I/O and dedicated input pins and all macrocell feedbacks from within the logic blocks. The number of PIM inputs increases with pin count and the number of logic blocks. The outputs from the PIM are signals routed to the appropriate logic block(s). Each logic block receives 36 inputs from the PIM and their complements, allowing for 32-bit operations to be implemented in a single pass through the device. The wide number of inputs to the logic block also improves the routing capacity of the FLASH370i family.

An important feature of the PIM is simple timing. The propagation delay through the PIM is accounted for in the timing specifications for each device. There is no additional delay for traveling through the PIM. In fact, all inputs travel through the PIM. Likewise, there are no route-dependent timing parameters on the FLASH370i devices. The worst-case PIM delays are incorporated in all appropriate FLASH370i specifications.

Routing signals through the PIM is completely invisible to the user. All routing is accomplished by software—no hand routing is necessary. *Warp™* and third-party development packages automatically route designs for the FLASH370i family in a matter of minutes. Finally, the rich routing resources of the FLASH370i family accommodate last minute logic changes while maintaining fixed pin assignments.
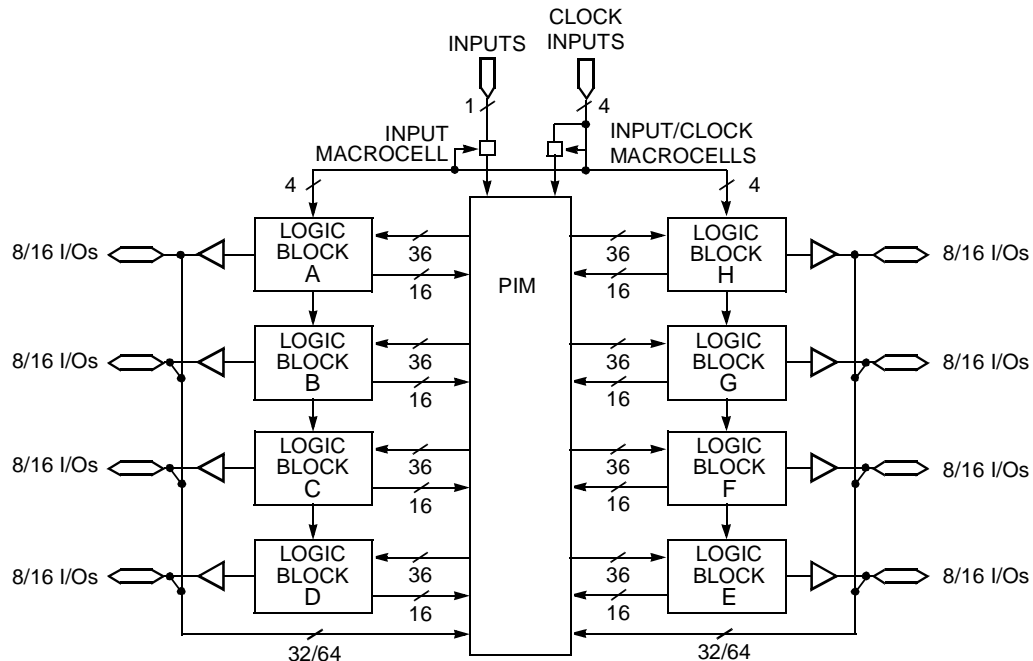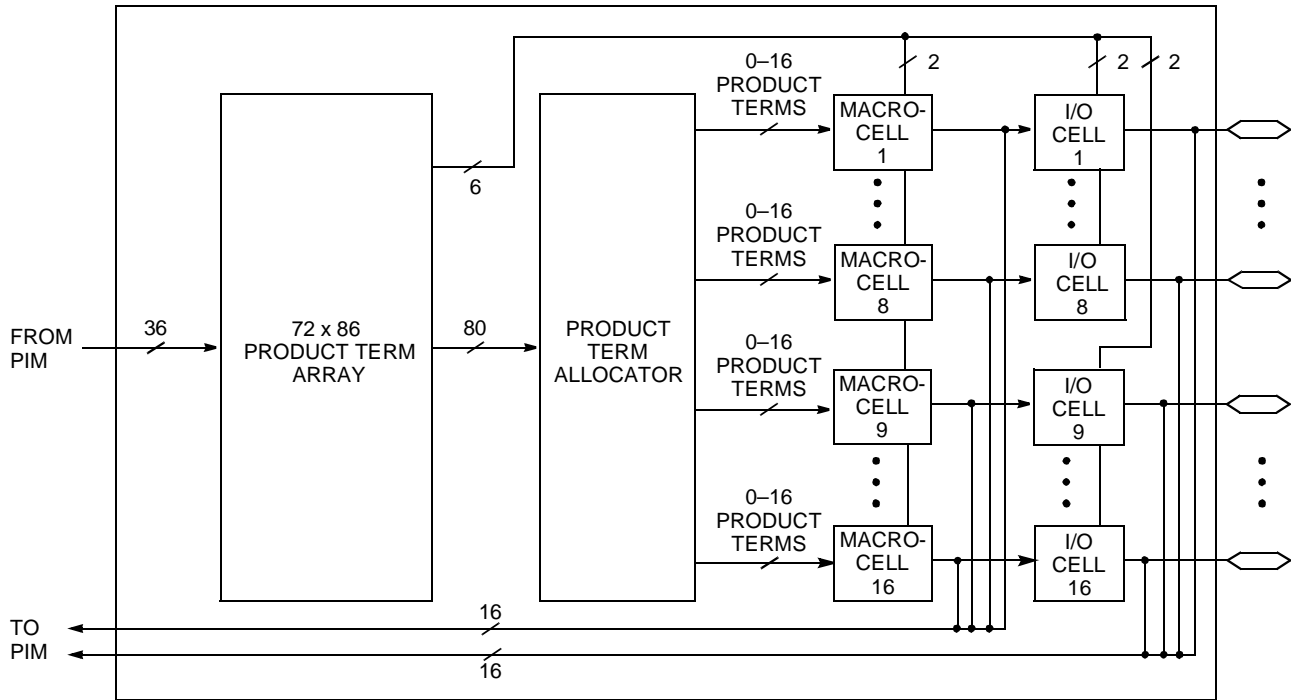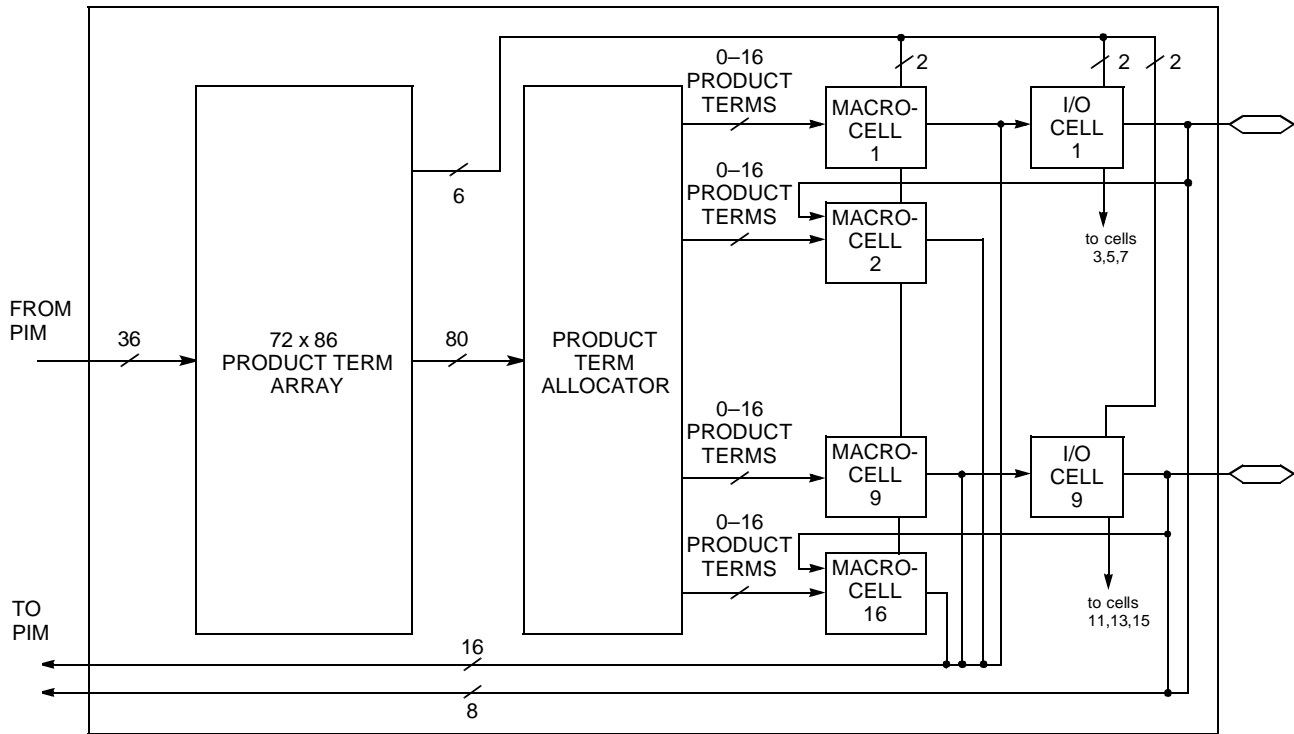
Logic Block Diagram

Figure 1. CY7C374i/5i Block Diagram

2

flash370i–2

Figure 2. Logic Block for CY7C371i, CY7C373i, and CY7C375i (I/O Intensive)

3

flash370i–3

Figure 3. Logic Block for CY7C372i and CY7C374i (Register Intensive)

Logic Block

The logic block is the basic building block of the FLASH370i architecture. It consists of a product term array, an intelligent product-term allocator, 16 macrocells, and a number of I/O cells. The number of I/O cells varies depending on the device used.

There are two types of logic blocks in the FLASH370i family. The first type features an equal number (16) of I/O cells and macrocells and is shown in *Figure 2*. This architecture is best for I/O-intensive applications. The second type of logic block features a buried macrocell along with each I/O macrocell. In other words, in each logic block, there are eight macrocells that are connected to I/O cells and eight macrocells that are only internally fed back to the PIM. This organization is designed for register-intensive applications and is displayed in *Figure 3*. Note that at each FLASH370i density (except the 32-macrocell CY7C371i), an I/O intensive and a register-intensive device is available.

Product Term Array

Each logic block features a 72 x 86 programmable product term array. This array is fed with 36 inputs from the PIM, which originate from macrocell feedbacks and device pins. Active LOW and active HIGH versions of each of these inputs are generated to create the full 72-input field. The 86 product terms in the array can be created from any of the 72 inputs.

Of the 86 product terms, 80 are for general-purpose use for the 16 macrocells in the logic block. Four of the remaining six product terms in the logic block are output enable (OE) product

terms. Each of the OE product terms controls up to eight of the 16 macrocells and is selectable on an individual macrocell basis. In other words, each I/O cell can select between one of two OE product terms to control the output buffer. The first two of these four OE product terms are available to the upper half of the I/O macrocells in a logic block. The other two OE product terms are available to the lower half of the I/O macrocells in a logic block. The final two product terms in each logic block are dedicated asynchronous set and asynchronous reset product terms.

Product Term Allocator

Through the product term allocator, software automatically distributes product terms among the 16 macrocells in the logic block. A total of 80 product terms are available from the local product term array. The product term allocator provides two important capabilities without affecting performance: product term steering and product term sharing.

Product Term Steering

Product term steering is the process of assigning product terms to macrocells as needed. For example, if one macrocell requires ten product terms while another needs just three, the product term allocator will "steer" ten product terms to one macrocell and three to the other. On FLASH370i devices, product terms are steered on an individual basis. Any number between 0 and 16 product terms can be steered to any macrocell. Note that 0 product terms is useful in cases where a particular macrocell is unused or used as an input register.

4

Product Term Sharing

Product term sharing is the process of using the same product term among multiple macrocells. For example, if more than one output has one or more product terms in its equation that are common to other outputs, those product terms are only programmed once. The FLASH370i product term allocator allows sharing across groups of four output macrocells in a variable fashion. The software automatically takes advantage of this capability—the user does not have to intervene. Note that greater usable density can often be achieved if the user "floats" the pin assignment. This allows the compiler to group macrocells that have common product terms adjacently.

Note that neither product term sharing nor product term steering have any effect on the speed of the product. All worst-case steering and sharing configurations have been incorporated in the timing specifications for the FLASH370i devices.

FLASH370i Macrocell

I/O Macrocell

Within each logic block there are 8 or 16 I/O macrocells depending on the device used. *Figure 4* illustrates the architecture of the I/O macrocell. The macrocell features a register that can be configured as combinatorial, a D flip-flop, a T flip-flop, or a level-triggered latch.

The register can be asynchronously set or asynchronously reset at the logic block level with the separate set and reset product terms. Each of these product terms features programmable polarity. This allows the registers to be set or reset based on an AND expression or an OR expression.

Clocking of the register is very flexible. Depending on the device, either two or four global synchronous clocks are available to clock the register. Furthermore, each clock features program-

mable polarity so that registers can be triggered on falling as well as rising edges (see the Dedicated/Clock Inputs section). Clock polarity is chosen at the logic block level.

At the output of the macrocell, a polarity control mux is available to select active LOW or active HIGH signals. This has the added advantage of allowing significant logic reduction to occur in many applications.

The FLASH370i macrocell features a feedback path to the PIM separate from the I/O pin input path. This means that if the macrocell is buried (fed back internally only), the associated I/O pin can still be used as an input.

Buried Macrocell

Some of the devices in the FLASH370i family feature additional macrocells that do not feed individual I/O pins *Figure 5*. displays the architecture of the I/O and buried macrocells for these devices. The I/O macrocell is identical to the I/O macrocell for devices without buried macrocells.

The buried macrocell is very similar to the I/O macrocell. Again, it includes a register that can be configured as combinatorial, a D flip-flop, a T flip-flop, or a latch. The clock for this register has the same options as described for the I/O macrocell. The primary difference between the I/O macrocell and the buried macrocell is that the buried macrocell does not have the ability to output data directly to an I/O pin.

One additional difference on the buried macrocell is the addition of input register capability. The buried macrocell can be configured to act as an input register (D-type or latch) whose input comes from the I/O pin associated with the neighboring macrocell. The output of all buried macrocells is sent directly to the PIM regardless of its configuration.
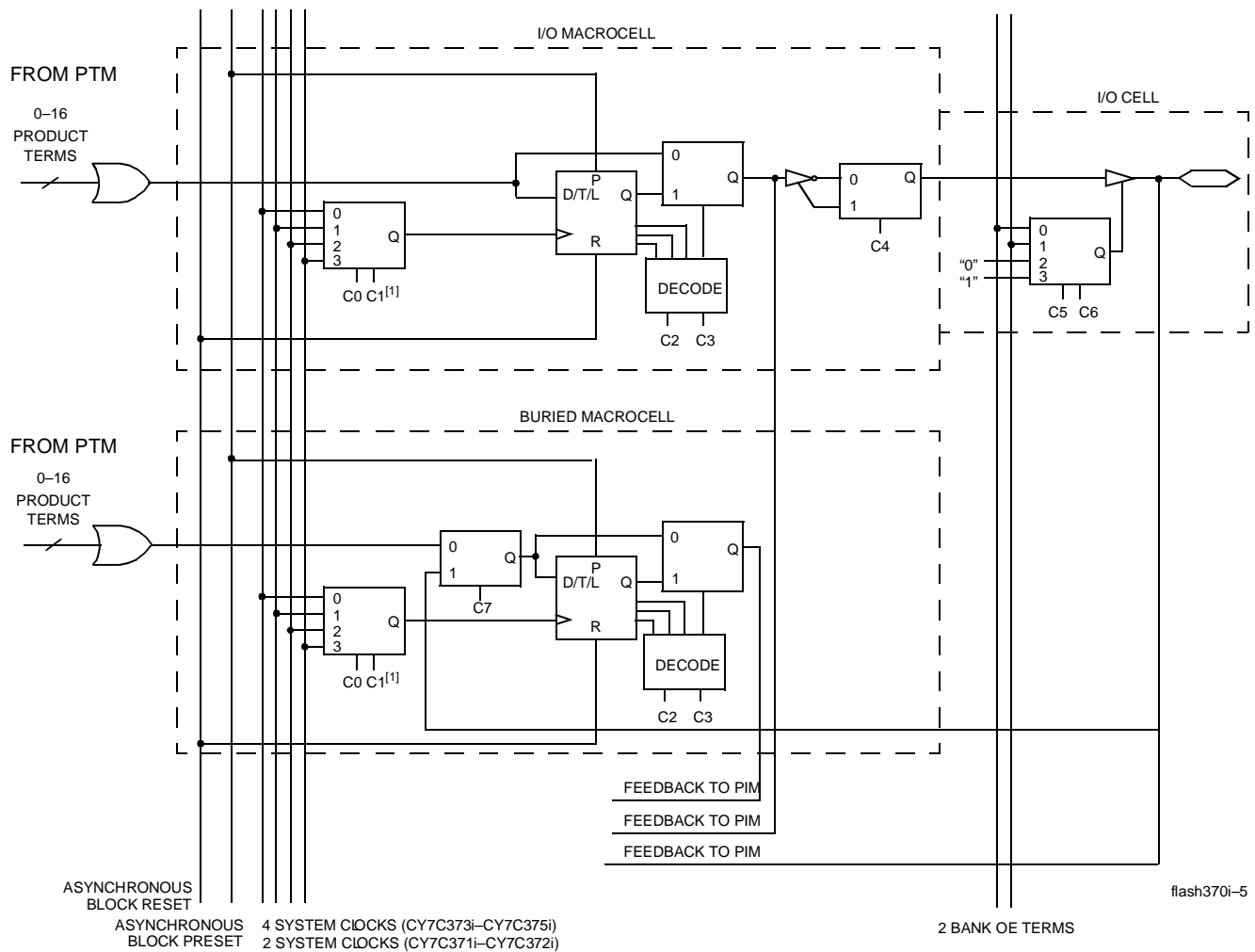
Figure 4. I/O and Buried Macrocells
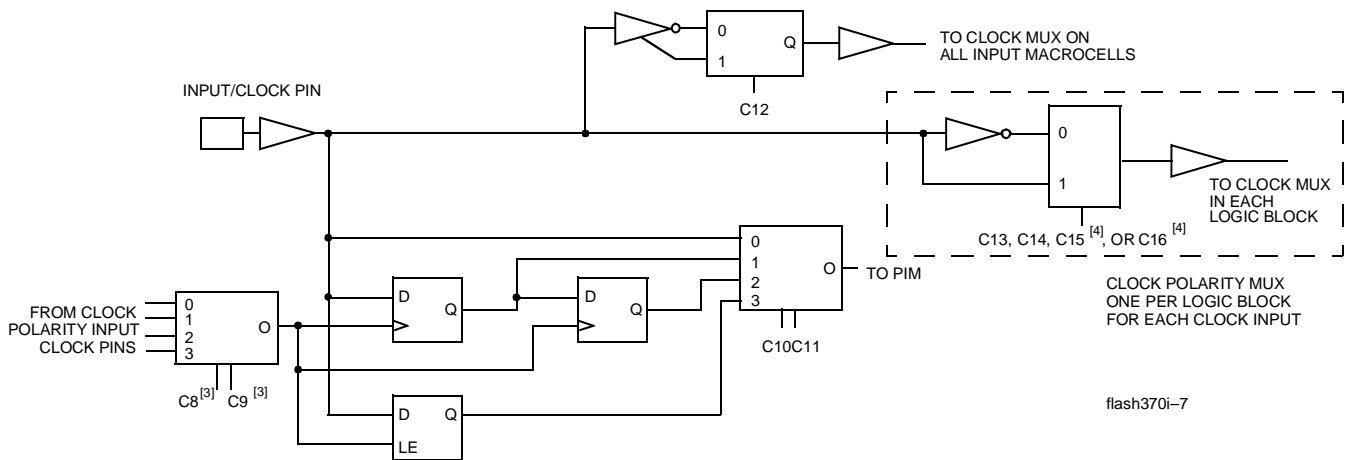
Figure 5. Input Pins

Notes:
1. C1 is not used on the CY7C371i and CY7C372i since the mux size is 2:1.
2. C9 is not used on the CY7C371i and CY7C372i since the mux size is 2:1.
3. C8 and C9 are not included on the CY7C371i and CY7C372i since each input/clock pin has the other input/clock pin on its clock.
4. C15 and C16 are not used on the CY7C371i and CY7C372i since there are two clocks.
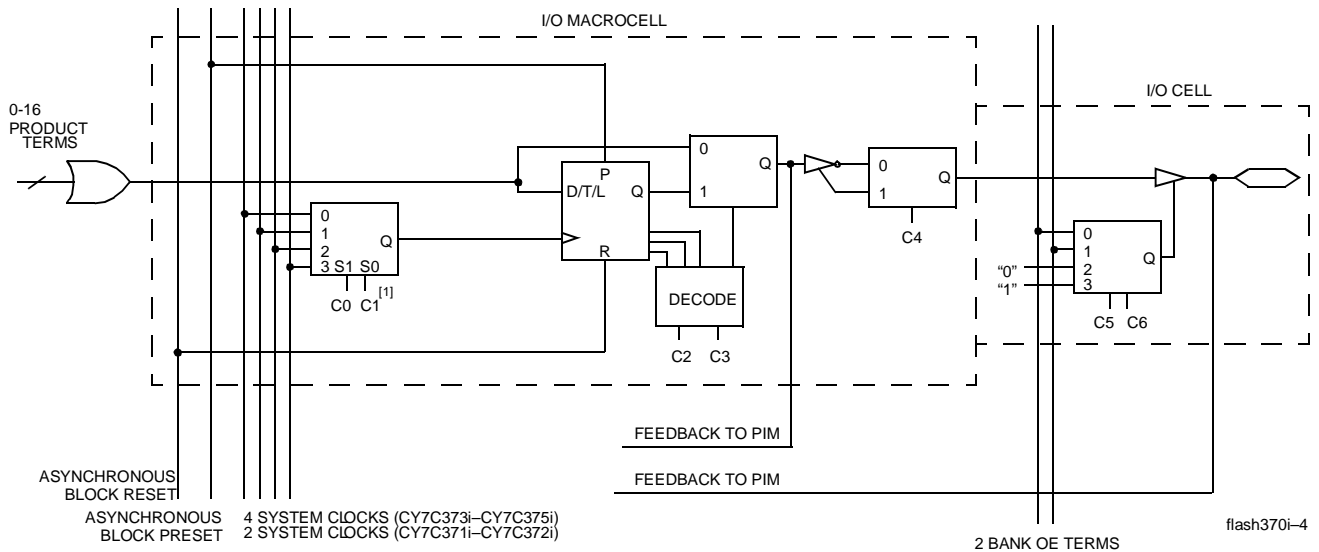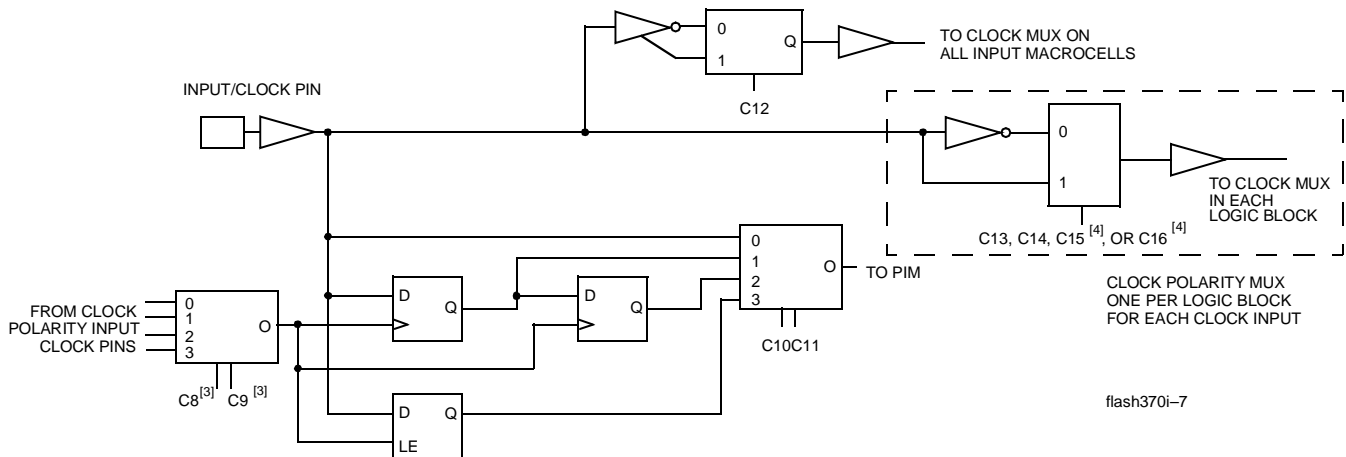
6

Figure 6. I/O Macrocell

Figure 7. Input/Clock Pins

FLASH370i I/O Cell

The I/O cell on the FLASH370i devices is illustrated along with the I/O macrocell in *Figures 6* and *4*. The user can program the I/O cell to change the way the three-state output buffer is enabled and/or disabled. Each output can be set permanently on (output only), permanently off (input only), or dynamically controlled by one of two OE product terms.

Dedicated/Clock Inputs

Six pins on each member of the FLASH370i family are designated as input-only. There are two types of dedicated inputs on FLASH370i devices: input pins and input/clock pins. *Figure 5* illustrates the architecture for input pins. Four input options are available for the user: combinatorial, registered, double-registered, or latched. If a registered or latched option is

selected, any one of the input clocks can be selected for control. *Figure 7* shows the input/clock macrocell. The CY7C371i and CY7C372i have two input/clock pins while the other devices in the family have four input/clock pins. Like the input pins, input/clock pins can be combinatorial, registered, double registered, or latched. In addition, these pins feed the clocking structures throughout the device. The clock path at the input is user-configurable in polarity. The polarity of the clock signal can also be controlled by the user. Note that this polarity is separately controlled for input registers and output registers.

Timing Model

One of the most important features of the FLASH370i family is the simplicity of its timing. All delays are worst case and system performance is unaffected by the features used or
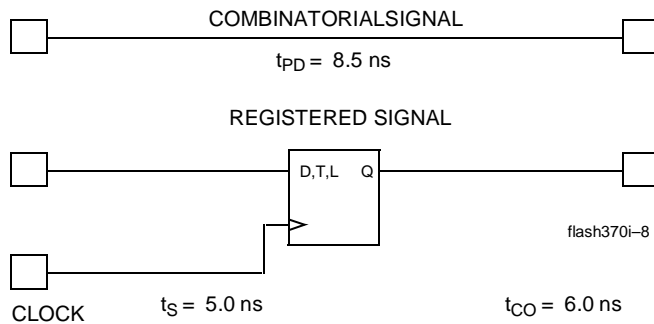
7

$$t_{PD} = 8.5 \text{ ns}$$

REGISTERED SIGNAL

$$t_S = 5.0 \text{ ns} \qquad t_{CO} = 6.0 \text{ ns}$$

Figure 8. Timing Model for CY7C371i

not used on the parts. *Figure 8* illustrates the true timing model for the 8.5-ns devices. For combinatorial paths, any input to any output incurs an 8.5-ns worst-case delay regardless of the amount of logic used. For synchronous systems, the input set-up time to the output macrocells for any input is 5.0 ns and the clock to output time is also 6.0 ns. These measurements are for any output and clock, regardless of the logic used.

Stated another way, the FLASH370i features:

- no fanout delays
- no expander delays
- no dedicated vs. I/O pin delays
- no additional delay through PIM
- no penalty for using 0–16 product terms
- no added delay for steering product terms
- no added delay for sharing product terms
- no routing delays
- no output bypass delays

The simple timing model of the FLASH370i family eliminates unexpected performance penalties.

PCI Compliance

The FLASH370i family of CMOS CPLDs are fully compliant with the PCI Local Bus Specification published by the PCI Special Interest Group. The simple and predictable timing model of FLASH370i ensures compliance with the PCI AC specifications independent of the design. On the other hand, in CPLD and FPGA architectures without simple and predictable timing, PCI compliance is dependent upon routing and product term distribution.

3.3V or 5.0V I/O operation

The FLASH370i family can be configured to operate in both 3.3V and 5.0V systems. All devices have two sets of V_{CC} pins: one set, V_{CCINT}, for internal operation and input buffers, and another set, V_{CCIO}, for I/O output drivers. V_{CCINT} pins must always be connected to a 5.0V power supply. However, the V_{CCIO} pins may be connected to either a 3.3V or 5.0V power supply, depending on the output requirements. When V_{CCIO} pins are connected to a 5.0V source, the I/O voltage levels are compatible with 5.0V systems. When V_{CCIO} pins are connected to a 3.3V source, the input voltage levels are compatible with both 5.0V and 3.3V systems, while the output voltage lev-

els are compatible with 3.3V systems. There will be an additional timing delay on all output buffers when operating in 3.3V I/O mode. The added flexibility of 3.3V I/O capability is available in commercial and industrial temperature ranges.

Bus-Hold Capabilities on all I/Os and Dedicated Inputs

In addition to ISR capability, a new feature called bus-hold has been added to all FLASH370i I/Os and dedicated input pins. Bus-hold, which is an improved version of the popular internal pull-up resistor, is a weak latch connected to the pin that does not degrade the device's performance. As a latch, bus-hold recalls the last state of a pin when it is three-stated, thus reducing system noise in bus-interface applications. Bus-hold additionally allows unused device pins to remain unconnected on the board, which is particularly useful during prototyping as designers can route new signals to the device without cutting trace connections to V_{CC} or G_{ND}.

Development Software Support

Warp™

Warp is a state-of-the-art compiler and complete CPLD design tool. For design entry, *Warp* provides an IEEE-STD-1076/1164 VHDL text editor, an IEEE-STD-1364 Verilog text editor, and a graphical finite state machine editor. It provides optimized synthesis and fitting by replacing basic circuits with ones pre-optimized for the target device, by implementing logic in unused memory and by perfect communication between fitting and synthesis. *Warp* provides other tools such as graphical timing simulation and analysis.

Warp Professional™

Warp Professional contains several additional features. It provides an extra method of design entry with its graphical block diagram editor. It allows up to 5 ms timing simulation instead of only 2 ms. It allows comparison of waveforms before and after design changes.

Warp Enterprise™

Warp Enterprise provides even more features. It provides unlimited timing simulation and source-level behavioral simulation as well as a debugger. It has the ability to generate graphical HDL blocks from HDL text. It can even generate testbenches.

Warp is available for PC and UNIX platforms. Some features are not available in the UNIX version. For further information see the *Warp* for PC, *Warp* for UNIX, *Warp* Professional and *Warp* Enterprise data sheets.

Third-Party Software

Although *Warp* is a complete CPLD development tool on its own, it interfaces with nearly every third party EDA tool. All major third-party software vendors provide support for the FLASH370i family of devices. To expedite this support, Cypress supplies vendors with all pertinent architectural information as well as design fitters for our products.

Programming

There are four programming options available for FLASH370i devices. The first is to use a PC with the FLASH370i ISR programming cable and software. With this method, the ISR pins of the FLASH370i devices are routed to a connector at the edge

8

of the printed circuit board. The ISR programming cable is then connected between the parallel port of the PC and this connector. A simple configuration file instructs the ISR software of the programming operations to be performed on each of the FLASH370i devices in the system. The ISR software then automatically completes all of the necessary data manipulations required to accomplish the programming, reading, verifying, and other ISR functions. For more information on Cypress ISR Interface, see the ISR Programming Kit.

The second method for programming FLASH370i devices is on an ATE (automatic test equipment). If this method is selected, the FLASH370i ISR software generates programming test vectors for the target tester.

The third programming option for FLASH370i devices is to utilize the embedded controller or processor that already exists in the system. The FLASH370i ISR software assists in this method by converting the device JEDEC maps into the ISR serial stream that contains the ISR instruction information and the addresses and data of locations to be programmed. The embedded controller then simply directs this ISR stream to the chain of FLASH370i devices to complete the desired reconfiguring or diagnostic operations.

The fourth method for programming FLASH370i devices is to use the same parallel programmer that is currently being used to program FLASH370 devices. Since the programming algorithms are the same, normal third party programming support for the FLASH370i family is readily available.

For all pinout, electrical, and timing requirements, refer to device data sheets. For ISR cable and software specifications, refer to ISR Kit data sheets. For detailed description of ISR capabilities, refer to the Cypress application note, "An Introduction to In System Reprogramming with FLASH370i."

Third-Party Programmers

As with development software, Cypress supports third-party programmers. All major third-party programmers (including Data I/O, Logical Devices, Minato, SMS, and Stag) will support the FLASH370i family.

Document #: 38-00493-D

ISR, UltraLogic, FLASH, *Warp*, *Warp* Professional, and *Warp* Enterprise are trademarks of Cypress Semiconductor Corporation. ViewSim and ViewDraw are trademarks of ViewLogic.