# FLASH370™ CPLD Family

## UltraLogic™ High-Density Flash CPLDs

### Features

- **Flash erasable CMOS CPLDs**
- **High density**
  - **32–128 macrocells**
  - **32–128 I/O pins**
  - **Multiple clock pins**
- **High speed**
  - **$t_{PD}$ = 8.5 – 12 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**
  - **Prevents stealing of neighboring product terms**
- **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, and TQFP packages**
- ***Warp2™/Warp2+™***
  - **Low-cost, text-based design tool, PLD compiler**
  - **IEEE 1164-compliant VHDL**
  - **Available on PC and Sun platforms**
- ***Warp3™* CAE development system**
  - **VHDL input**
  - **ViewLogic graphical user interface**
  - **Schematic capture (ViewDraw™)**
  - **VHDL simulation (ViewSim™)**
  - **Available on PC, HP, and Sun platforms**

### General Description

The FLASH370™ 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 devices are electrically erasable and reprogrammable, simplifying product inventory and reducing costs.

The FLASH370 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 one logic block to another as well as all inputs from pins.

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 CY7C374 and CY7C375 both feature 128 macrocells. On the CY7C374 half of the macrocells are buried and the device is available in 84-pin packages. On the CY7C375 all of the macrocells are fed to I/O pins and the device is available in 160-pin packages. *Figure 1* shows a block diagram of the CY7C374/5.

### 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 pincount 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 FLASH370 family.

An important feature of the PIM involves 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 FLASH370 devices. The worst-case PIM delays are incorporated in all appropriate FLASH370 specifications.

### FLASH370 Selection Guide

| Device | Pins | Macrocells | Dedicated Inputs | I/O Pins | Flip-Flops | Speed ($t_{PD}$) | Speed ($f_{MAX}$) |
|---|---|---|---|---|---|---|---|
| 371 | 44 | 32 | 6 | 32 | 44 | 8.5 | 143 |
| 372 | 44 | 64 | 6 | 32 | 76 | 10 | 125 |
| 373 | 84 | 64 | 6 | 64 | 76 | 10 | 125 |
| 374 | 84 | 128 | 6 | 64 | 140 | 12 | 100 |
| 375 | 160 | 128 | 6 | 128 | 140 | 12 | 100 |

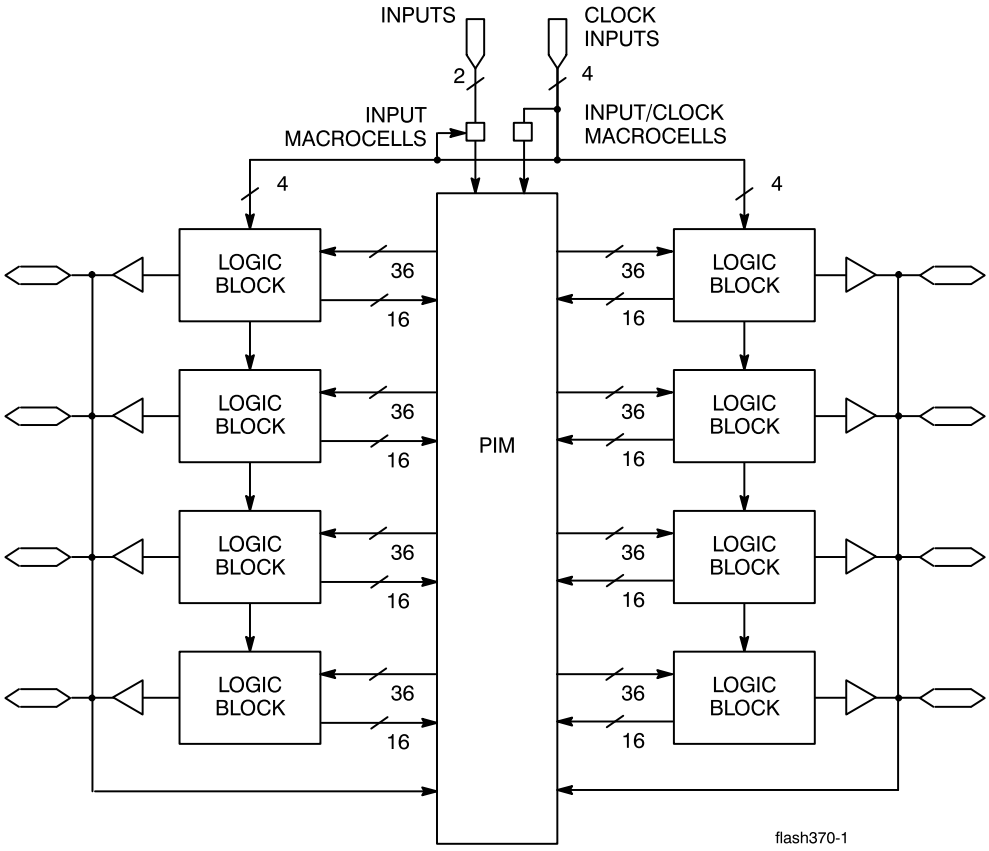


**Figure 1. CY7C374/5 Block Diagram**

## Functional Description (continued)

Routing signals through the PIM is completely invisible to the user. All routing is accomplished 100% by software—no hand routing is necessary. *Warp* and third-party development packages automatically route designs for the FLASH370 family in a matter of minutes. Finally, the rich routing resources of the FLASH370 family accommodate last minute logic changes while maintaining fixed pin assignments.

### Logic Block

The logic block is the basic building block of the FLASH370 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 FLASH370 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 internally fed back to the PIM only. This organization is designed for register-intensive applications and is displayed in *Figure 3*. Note that at each FLASH370 density (except the smallest), 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 as needed. 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.

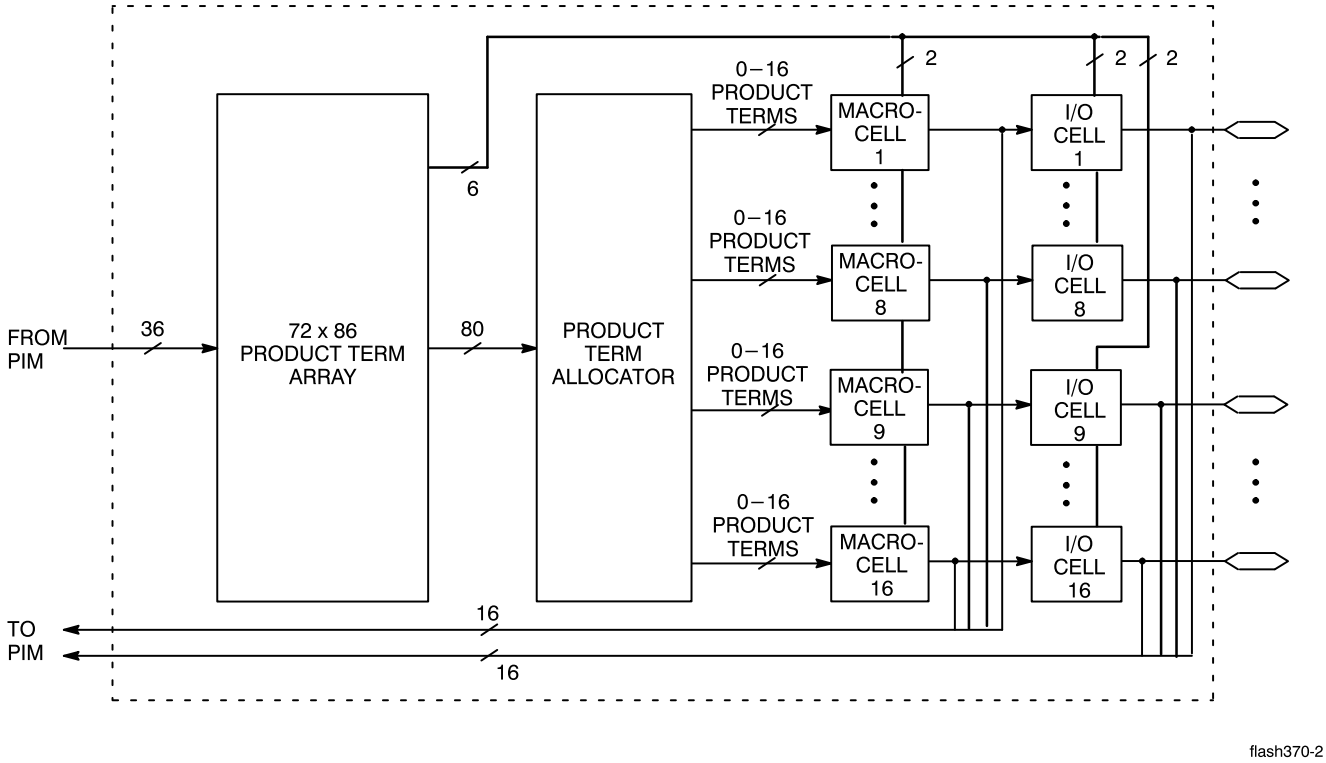


**Figure 2. Logic Block for CY7C371, CY7C373, and CY7C375 (I/O Intensive)**

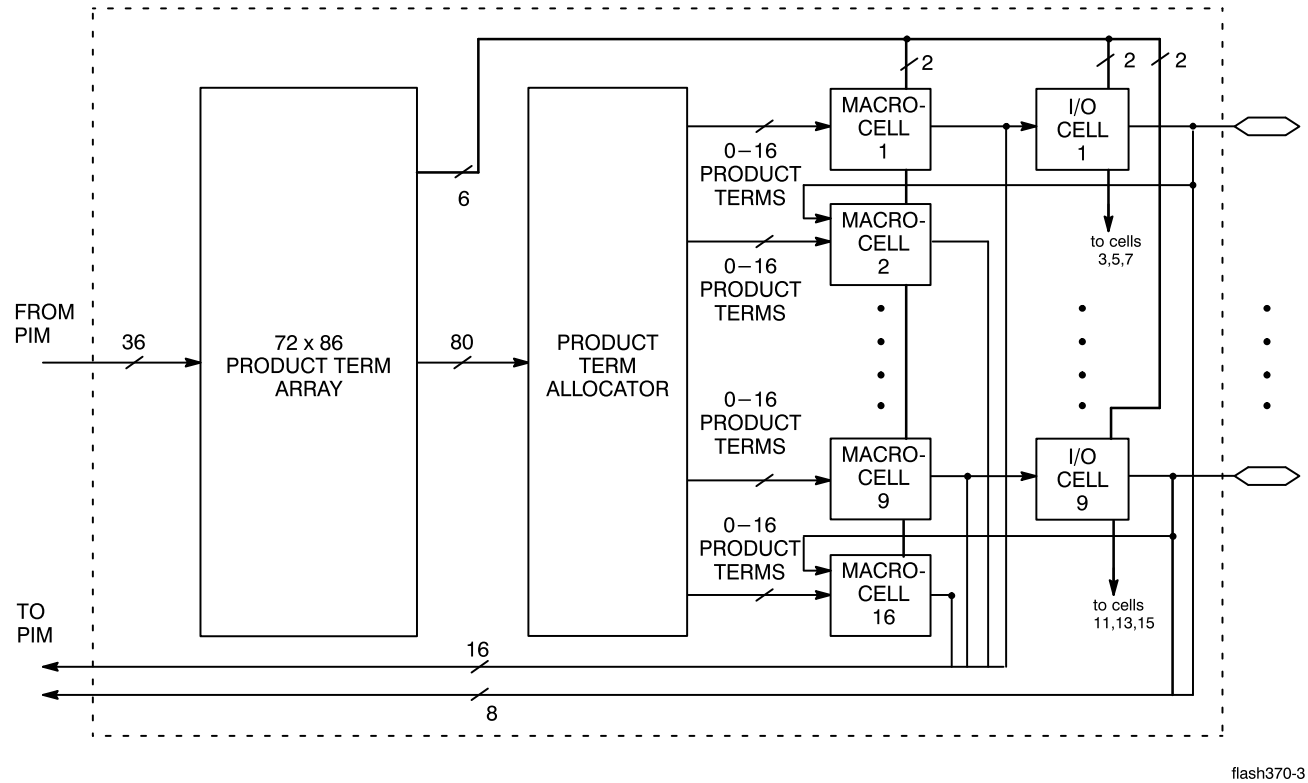


**Figure 3. Logic Block for CY7C372 and CY7C374 (Register Intensive)**

*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 FLASH370 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.

*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 FLASH370 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 FLASH370 devices.

**FLASH370 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 programmable 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 FLASH370 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 FLASH370 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 one on 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.

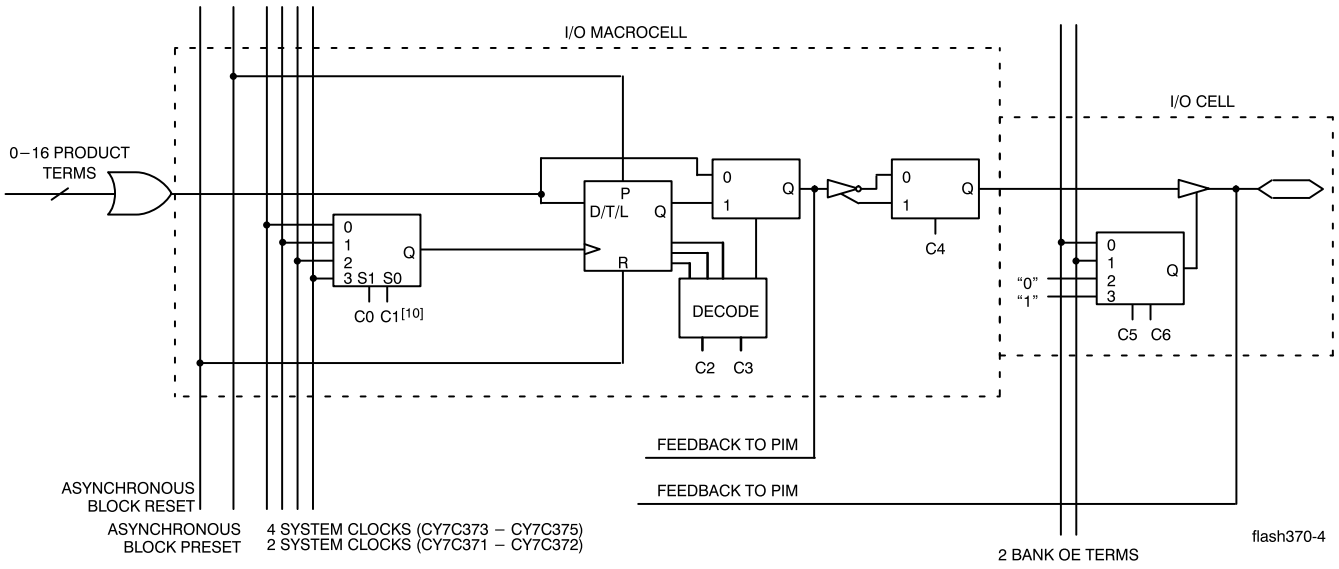


**Figure 4. I/O Macrocell**

**Note:**
10. C1 is not used on the CY7C371 and CY7C372 since the mux size is 2:1

### FLASH370 I/O Cell

The I/O cell on the FLASH370 devices is illustrated along with the I/O macrocell in *Figures 4* and *5*. 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 FLASH370 family are designated as input-only. There are two types of dedicated inputs on FLASH370 devices: input pins and input/clock pins. *Figure 6* illustrates the ar-

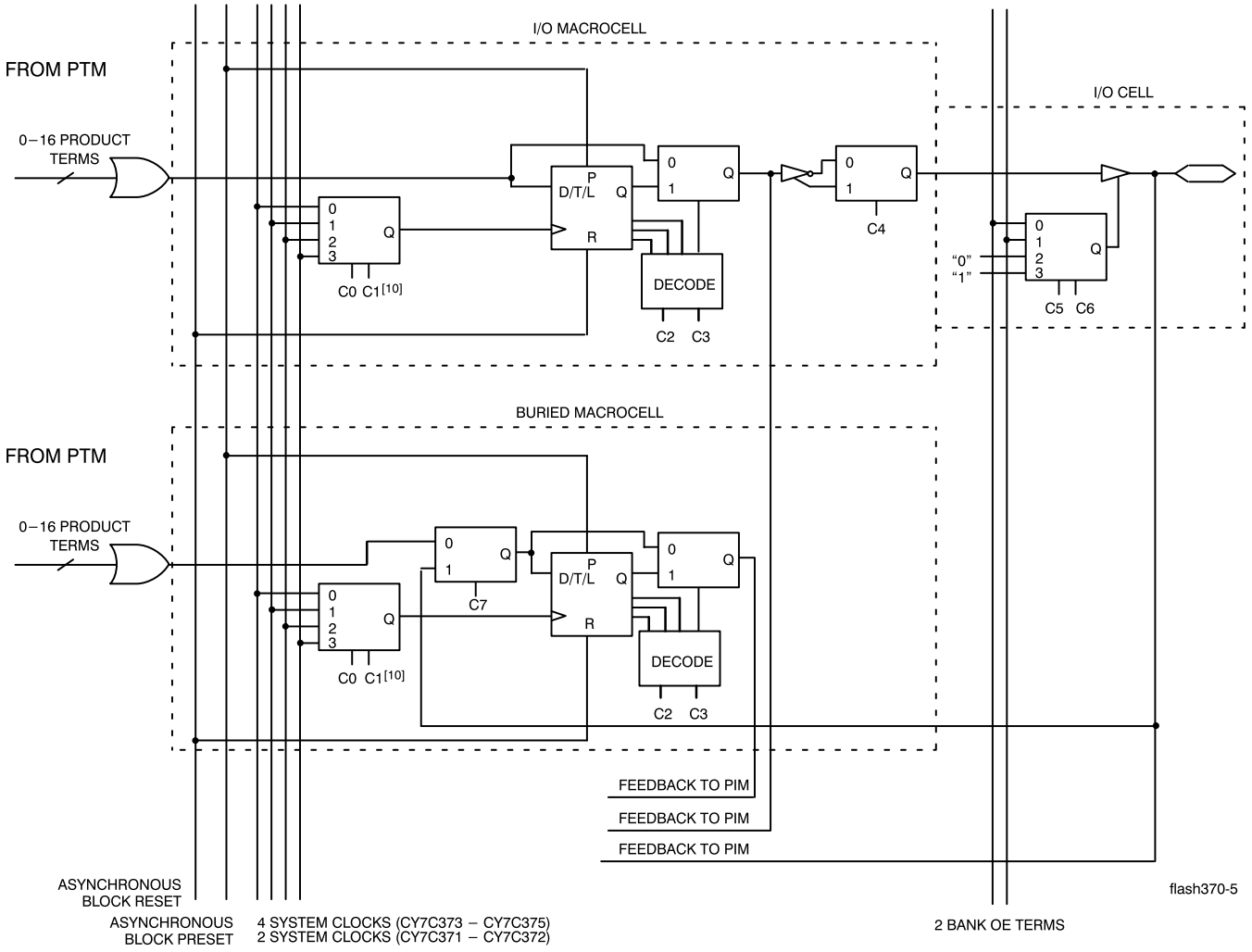


**Figure 5. I/O and Buried Macrocells**

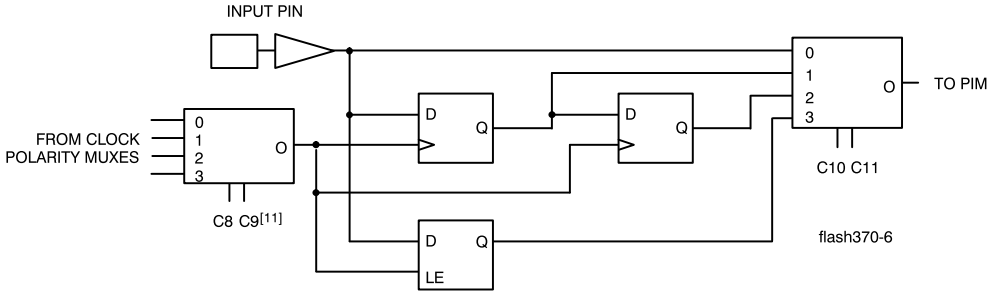


**Figure 6. Input Pins**

**Note:**
11. C9 is not used on the CY7C371 and CY7C372 since the mux size is 2:1

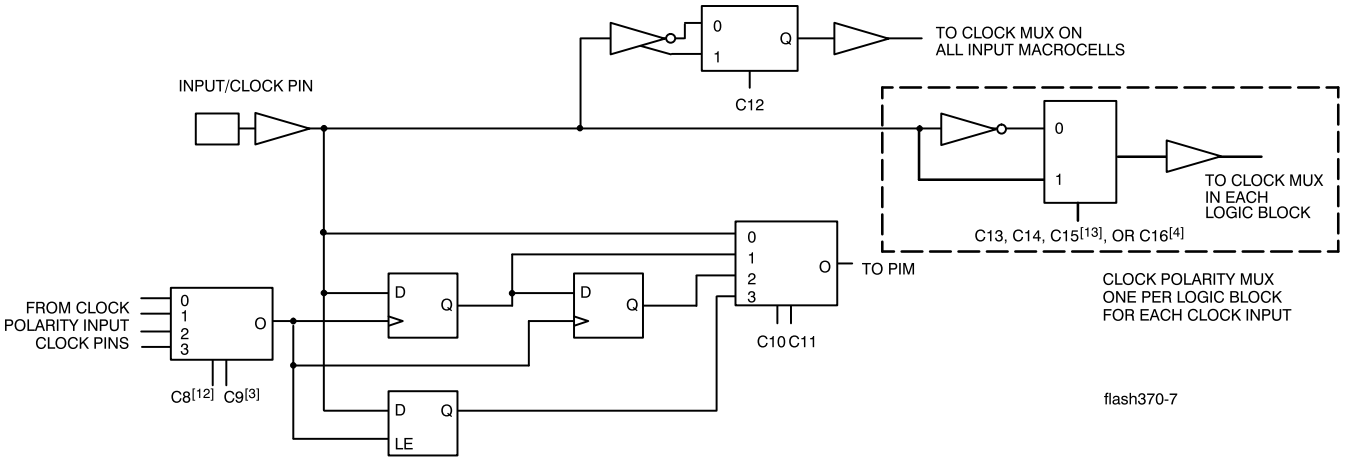


**Figure 7. Input/Clock Pins**

**Notes:**
3. C8 and C9 are not included on the CY7C371 and CY7C372 since each input/clock pin has the other input/clock pin as its clock.
4. C15 and C16 are not used on the CY7C371 and CY7C372 since there are two clocks.

chitecture 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* illustrates the architecture of input/clock pins. There are either two or four input/clock pins available, depending on the device selected. (The CY7C371 and CY7C372 have two input/clock pins while the other devices 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 FLASH370 family is the simplicity of its timing. All delays are worst case and system performance is unaffected by the features used or 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.

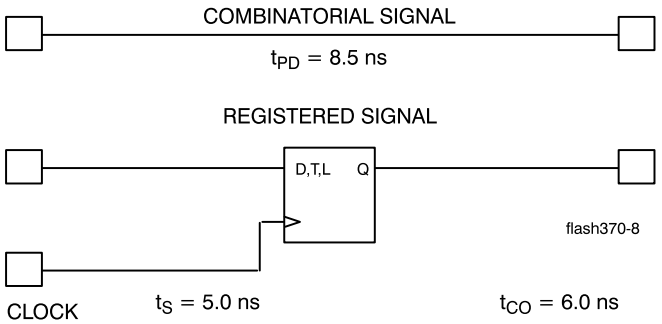


**Figure 8. Timing Model for CY7C371**

Again, these measurements are for any output and clock, regardless of the logic used.

Stated another way, the FLASH370 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 FLASH370 family eliminates unexpected performance penalties.

## Development Software Support

***Warp2/Warp2+***

*Warp2/Warp2+* are state-of-the-art VHDL compilers for designing with Cypress PLDs and PROMs. *Warp2/Warp2+* utilize a proper subset of IEEE 1164 VHDL as the Hardware Description Language (HDL) for design entry. VHDL provides a number of significant benefits for the design entry. VHDL provides a number of significant benefits for the design engineer. *Warp2/Warp2+* accepts VHDL input, synthesizes and optimizes the entered design, and outputs a JEDEC map for the desired device. For simulation, *Warp2/Warp2+* provides the graphical waveform simulator called Nova.

VHDL (VHSIC Hardware Description Language) is an open, powerful, non-proprietary language that is a standard for behavioral design entry and simulation. It is already mandated for use by the Department of Defense and supported by every major vendor of CAE tools. VHDL allows designers to learn a single language that is useful for all facets of the design process. See separate data sheet for further information.

***Warp3***

*Warp3* is a sophisticated design tool that is based on the latest version of ViewLogic's CAE design environment. *Warp3* features

schematic capture (ViewDraw™), VHDL waveform simulation (ViewSim™), a VHDL debugger, and VHDL synthesis, all integrated in a graphical design environment. *Warp3* is available on PCs using Windows 3.1 or subsequent versions, and on HP and Sun workstations. See separate data sheet for further information.

**Third-Party Software**

Cypress maintains a very strong commitment to third-party design software vendors. All major third-party software vendors (including ABEL™, LOG/iC™, CUPL™, and Minc) will provide support for the FLASH370 family of devices. To expedite this support, Cypress supplies vendors with all pertinent architectural information as well as design fitters for our products.

Document #: 38−00215−C

## Programming

The *Impulse3*™ device programmer from Cypress will program all Cypress PLDs, CPLDs, and PROMs. This unit is a programmer that connects to any IBM-compatible PC via the printer port.

**Third-Party Programmers**

As with development software, Cypress strongly supports third-party programmers. All major third-party programmers (including Data I/O, Logical Devices, Minato, SMS, and Stag) will support the FLASH370 family.