DATA SHEET

Xilinx has acquired the entire Philips CoolRunner Low Power CPLD Product Family. For more technical or sales information, please see: www.xilinx.com

XCR3320

320 macrocell SRAM CPLD

Preliminary specification

1998 Jul 22

IC27 Data Handbook

Philips Semiconductors

PHILIPS

320 macrocell SRAM CPLD

XCR3320

Xilinx has acquired the entire Philips CoolRunner Low Power CPLD Product Family. For more technical or sales information, please see: www.xilinx.com

FEATURES

- 320 macrocell SRAM based CPLD
- Multiple power-up configuration modes
  - Master serial
  - Slave serial
  - Master parallel-up
  - Master parallel-down
  - Slave parallel
  - Synchronous peripheral
  - Other modes available, contact Philips at 1–888–CoolPLD
- Configuration times of under 1.0 seconds
- IEEE 1149.1 compliant JTAG testing capability
  - 5 pin JTAG interface
  - IEEE 1149.1 TAP controller
- 3.3 volt device
- 5 V tolerant I/O
- Innovative XPLA2 Architecture combines extreme flexibility and high speeds
- 8 synchronous clock networks with programmable polarity at every macrocell
- Up to 32 asynchronous clocks support complex clocking needs
- Innovative XOR structure at every macrocell provides excellent logic reduction capability
- Logic expandable to 36 product terms on a single macrocell
- PCI compliant (except for clamp diode to V_{CC} rail due to 5 V tolerance)
- Advanced 0.35µ SRAM process
- Design entry and verification using industry standard and Philips CAE tools
- Innovative Control Term structure provides either sum terms of product terms in each logic block for:
  - 3-State buffer control
  - Asynchronous macrocell register reset/preset
- Global 3-State pin facilitates ‘bed of nails’ testing without sacrificing logic resources
- Programmable slew rate control
- Small form factor packages with high I/O counts
- Available in commercial and industrial temperature ranges

Table 1. PZ3320C/PZ3320N Features

	PZ3320C/PZ3320N
Usable gates	10,000
Maximum inputs	192
Maximum I/Os	192
Number of macrocells	320
Propagation delay (ns)	7.5
Packages	160 pin LQFP 256 pin PBGA

DESCRIPTION

The PZ3320 device is a member of the CoolRunner™ family of high-density SRAM-based CPLDs (Complex Programmable Logic Device) from Philips Semiconductors. This device combines high speed and deterministic pin-to-pin timing with high density. The PZ3320 uses the patented Fast Zero Power (FZP) design technique that combines high speed and low power for the first time ever in a CPLD. FZP allows the PZ3320 to have true pin-to-pin timing delays of 7.5ns, and standby currents of 100 microamps without the need for ‘turbo bits’ or other power down schemes. By replacing conventional sense amplifier methods for implementing product terms (a technique that has been used since the bipolar era) with a cascaded chain of pure CMOS gates, both standby and dynamic power are dramatically reduced when compared to other CPLDs. The FZP design technique is also what allows Philips to offer a true CPLD architecture in a high density device.

The Philips PZ3320C/PZ3320N devices use the new patent-pending XPLA2™ (eXtended Programmable Logic Array) architecture. This architecture combines the best features of both PAL- and PLA-type logic structures to deliver high speed and flexible logic allocation that results in superior ability to make design changes with fixed pinouts. The XPLA2™ architecture is constructed from 80 macrocell Fast Modules that are connected together by an interconnect array. Within each Fast Module are four Logic Blocks of 20 macrocells each. Each Logic Block contains a PAL structure with four dedicated product terms for each macrocell. In addition, each Logic Block has 32 additional product terms in a PLA structure that can be shared through a fully programmable OR array to any of the 20 macrocells. This combination efficiently allocates logic throughout the Logic Block, which increases device density and allows for design changes without re-defining the pinout or changing the system timing. The PZ3320 offers pin-to-pin propagation delays of 7.5ns through the PAL array of a Fast Module; and if the PLA array is used, an additional 1.5ns is added to the delay, no matter how many PLA product terms are used. If the interconnect array between Fast Modules is used, there is a second fixed addition to the propagation delay of 4.0ns. This means that the worst case pin-to-pin propagation delay within a fast module is 7.5 + 1.5 = 9.0 ns, and the delay from any pin to any other pin across the entire chip is 7.5 + 4.0 = 11.5ns if only the PAL array is used, and 7.5 + 1.5 + 4.0 = 13.0ns if the PLA array is used. This deterministic timing allows you to establish system timing before the logic design is even started.

Each macrocell also has a two input XOR gate with the dedicated PAL product terms on one input and the PLA product terms on the other input. This patent-pending Versatile XOR structure allows for very efficient logic optimization compared to competing XOR structures that have only one product term as the second input to the XOR gate. The Versatile XOR allows an 8 bit XOR function to be implemented in only 20 product terms, compared to 65 product terms for the traditional XOR approach.

The PZ3320 is SRAM-based, which means that it is configured at power up by one of many different methods. The device may be reconfigured any number of times. See the configuration section of this data sheet for more information. The device supports the full JTAG specification (IEEE 1149.1) through an industry standard JTAG interface.

Software support for the PZ3320 is through industry standard CAE tools (Cadence, Mentor, Synopsys, Synario, Viewlogic, MINC, Exemplar Logic, and Orcad) as well as Philips' own XPLA Designer. Entry methods include both text (ABEL, PHDL, VHDL, Verilog) and/or schematic. Design verification uses industry standard simulators for functional and timing simulation, and development tools are supported on personal computer, SPARC, and HP Workstation platforms. Device fitting uses either MINC or Philips Semiconductors developed tools.

ORDERING INFORMATION

ORDER CODE	PACKAGE, PROPAGATION DELAY	DESCRIPTION	DRAWING NUMBER
PZ3320C7xx	160-pin LQFP, 7.5 ns t_{PD}	Commercial temp. range, 3.3 volt power supply ±10%	
PZ3320C7yy	256-pin PBGA, 7.5 ns t_{PD}	Commercial temp. range, 3.3 volt power supply ±10%	
PZ3320N8xx	160-pin LQFP, 7.5 ns t_{PD}	Industrial temp. range, 3.3 volt power supply ±10%	
PZ3320N8yy	256-pin PBGA, 7.5 ns t_{PD}	Industrial temp. range, 3.3 volt power supply ±10%	

XPLA2 ARCHITECTURE

Figure 1 shows a high level block diagram of the PZ3320 implementing the XPLA2 architecture. The XPLA2 architecture is a multi-level, modular hierarchy that consists of Fast Modules interconnected by a Global Zero Power Interconnect Array (GZIA). The GZIA is a virtual crosspoint switch that connects the Fast Modules together. Each Fast Module accepts 64 bits from the GZIA and outputs 64 bits to the GZIA. Each Fast Module is essentially an 80 macrocell CPLD with four logic blocks of 20 macrocells each inside. There are eight dedicated, low-skew, global clocks for the device; and each Fast Module has access to any two of these clocks (there are additional asynchronous clocks available in the Fast Modules, see Figure 3). There are also Global 3-state (gts) and Global Reset (rstn) pins that are common to all Fast Modules. When gts is pulled high, all output buffers in the device will be disabled, causing all I/O pins to be tri-stated. When rstn is pulled low, all flip-flops of the device will be reset.

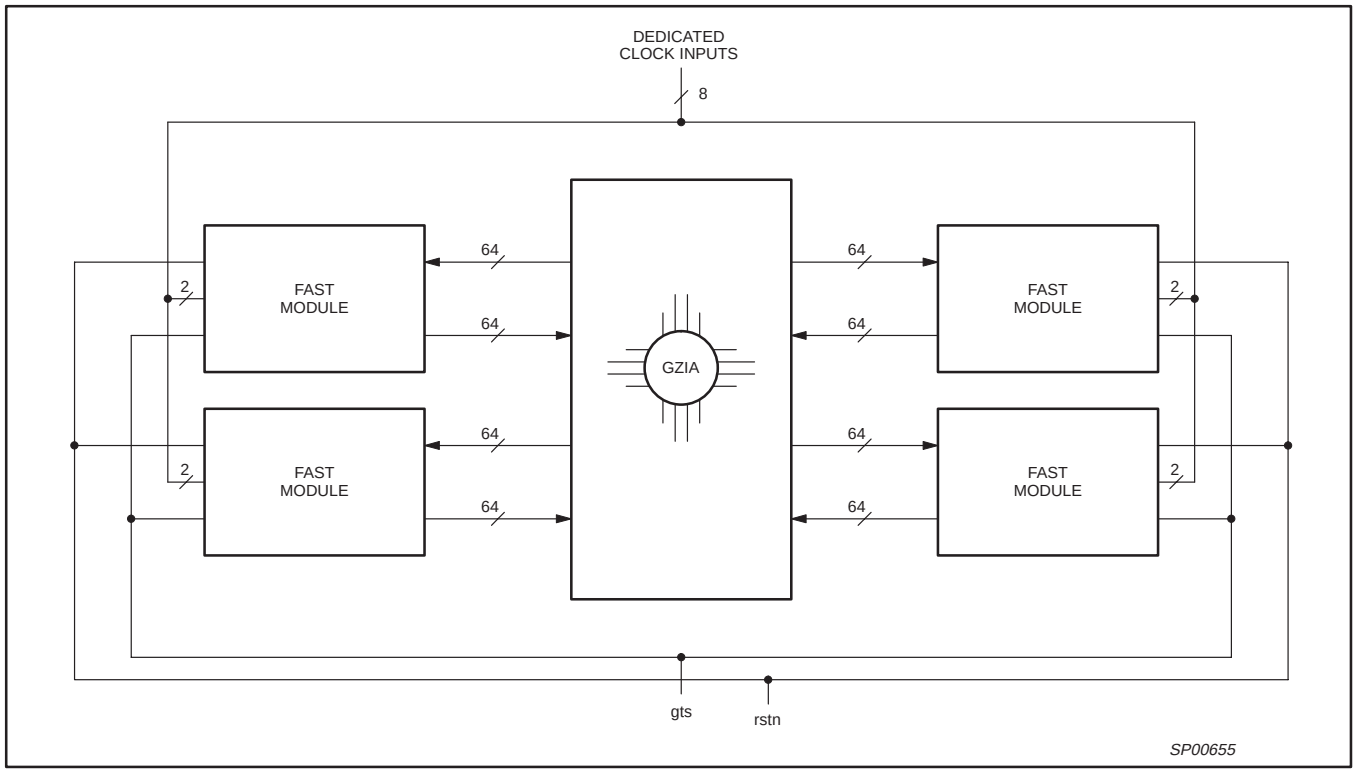


Figure 1. Philips XPLA2 CPLD Architecture

XPLA2 Fast Module

Each Fast Module consists of four Logic Blocks of 20 macrocells each. Depending on the package, either 8 or 12 of the 20 macrocells in each Logic Block are connected to I/O pins, and the remaining macrocells are used as buried nodes. These four Logic Blocks are connected together by the Local Zero Power Interconnect Array (LZIA). The LZIA is a virtual crosspoint switch that connects the Logic Blocks to each other and to the GZIA. The feedback from all 80 macrocells, input from the I/O pins, and the 64 bit input bus from the GZIA are input into the LZIA. The LZIA outputs 36 signals into each Logic Block and 64 signals into the GZIA.

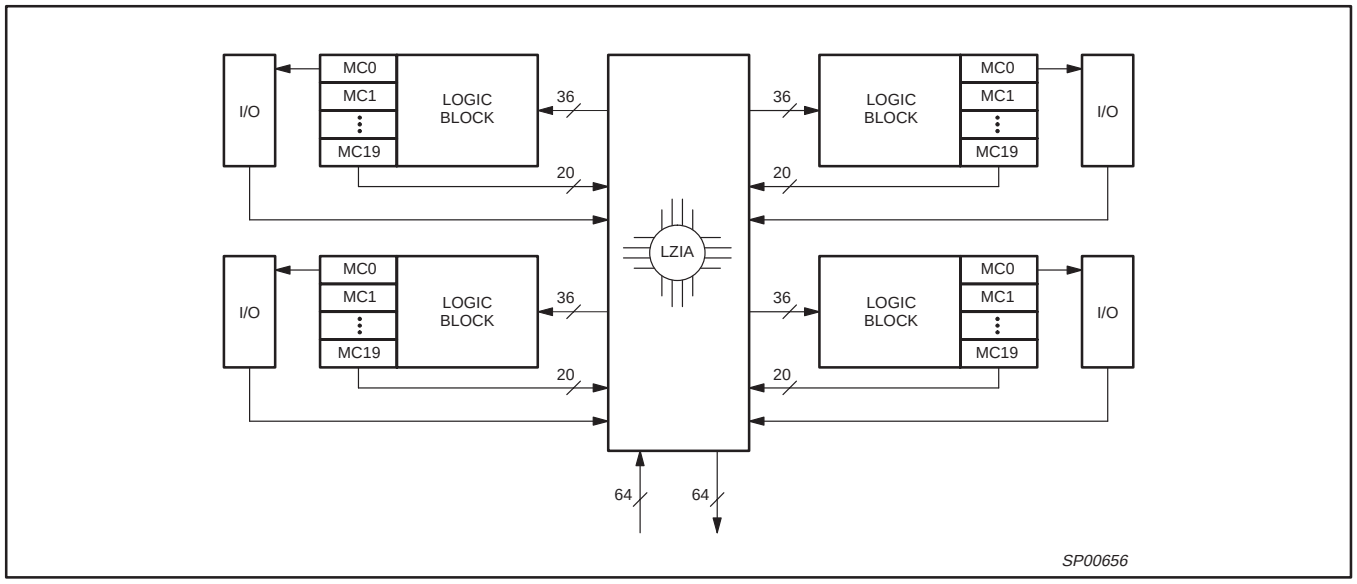


Figure 2. Philips XPLA2 Fast Module

XPLA2 Logic Block Architecture

Figure 3 illustrates the XPLA2 Logic Block architecture. Each Logic Block contains 8 control terms, a PAL array, a PLA array, and 20 macrocells. The 36 inputs from the LZIA are available to all control terms and to each product term in both the PAL and the PLA array. The 8 control terms can individually be configured as either SUM or PRODUCT terms, and are used to control the asynchronous preset and reset functions of the macrocell registers, the output enables of the 20 macrocells, and for asynchronous clocking. The PAL array consists of a programmable AND array with a fixed OR array, while the PLA array consists of a programmable AND array with a programmable OR array.

Each macrocell has 4 dedicated product terms from the PAL array. When additional logic is required, each macrocell takes the extra product terms from the PLA array. The PLA array consists of 32 extra product terms that are shared between the 20 macrocells of the Logic Block. The PAL product terms can be connected to the PLA product terms through either an OR gate or an XOR gate. One input to the XOR gate can be connected to all the PLA terms, which provides for extremely efficient logic synthesis. An eight bit XOR function can be implemented in only 20 product terms. Each macrocell can use the output from the OR gate or the XOR gate in either normal or inverted state.

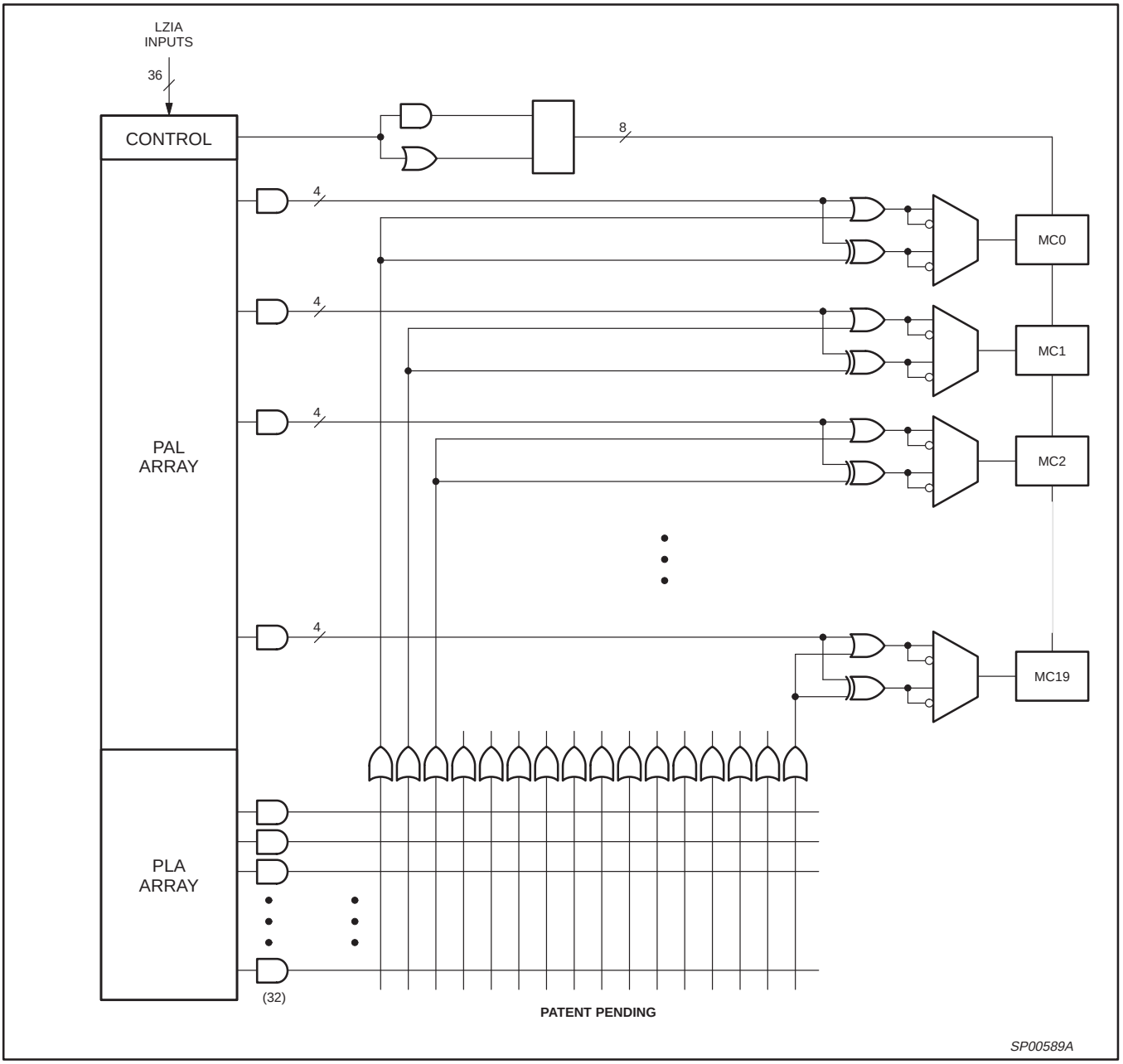


Figure 3. Philips XPLA2 Logic Block Architecture

XPLA2 Macrocell Architecture

Figure 4 shows the XPLA2 macrocell architecture used in the PZ3320. The macrocell can be configured as either a D- or T-type flip-flop or a combinatorial logic function. A D-type flip-flop is generally more useful for implementing state machines and data buffering while a T-type flip-flop is generally more useful in implementing counters. Each of these flip-flops can be clocked from any one of four sources. Two of the clock sources (CLK0 and CLK1) are from the eight dedicated, low-skew, global clock networks designed to preserve the integrity of the clock signal by reducing skew between rising and falling edges. These clocks are designated as a "synchronous" clocks and must be driven by an external source. Both CLK0 and CLK1 can clock the macrocell flip-flops on either the rising edge or the falling edge of the clock signal. The other clock sources are designated as "asynchronous" and are connected to two of the eight control terms (CT6 and CT7) provided in each logic block. These clocks can be individually configured as any PRODUCT term or SUM term equation created from the 36 signals available inside the logic block. Thus, in each Logic Block, there are up to four possible clocks; and in each Fast Module, there are up to 10 possible clocks. Throughout the entire device, there are up to 40 possible clocks—eight from the dedicated, low-skew, global clocks, and two for each of the 16 logic blocks.

The remaining six control terms of each logic block (CT0–CT5) are used to control the asynchronous preset/reset of the flip-flops and the enable/disable of the output buffers in each macrocell. Control terms CT0 and CT1 are used to control the asynchronous preset/reset of the macrocell's flip-flop. Note that the power-on reset leaves all macrocells in the "zero" state when power is properly applied, and that the preset/reset feature for each macrocell can also be disabled. Each macrocell can choose between an asynchronous reset or an asynchronous preset function, but both cannot be simultaneously used on the same register. The global rstn function can always be used, regardless of whether or not asynchronous reset or preset control terms are enabled. Control terms CT2, CT3, CT4 and CT5 are used to enable or disable the macrocell's output buffer. Having four dedicated output enable control terms ensures that the CoolRunner™ devices are PCI compliant. The output buffers can also be always enabled or always disabled. All CoolRunner™ devices also provide a Global 3-State (gts) pin, which, when pulled high, will 3-State all the outputs of the device. This pin is provided to support "In-Circuit Testing" or "Bed-of-Nails" testing used during manufacturing.

For the macrocells in the Logic Block that are associated with I/O pins, there are two feedback paths to the LZIA: one from the macrocell, and one from the I/O pin. The LZIA feedback path before the output buffer is the macrocell feedback path, while the LZIA feedback path after the output buffer is the I/O pin feedback path. When these macrocells are used as outputs, the output buffer is enabled, and either feedback path can be used to feedback the logic implemented in the macrocell. When the I/O pins are used as inputs, the output buffer of these macrocells will be 3-Stated and the input signal will be fed into the LZIA via the I/O feedback path. In this case the logic functions implemented in the buried macrocell can be fed back into the LZIA via the macrocell feedback path. For macrocells that are not associated with I/O pins, there is one feedback path to the LZIA. Logic functions implemented in these buried macrocells are fed back into the LZIA via this path. All unused inputs and I/O pins should be properly terminated. Please refer to the section on terminations.

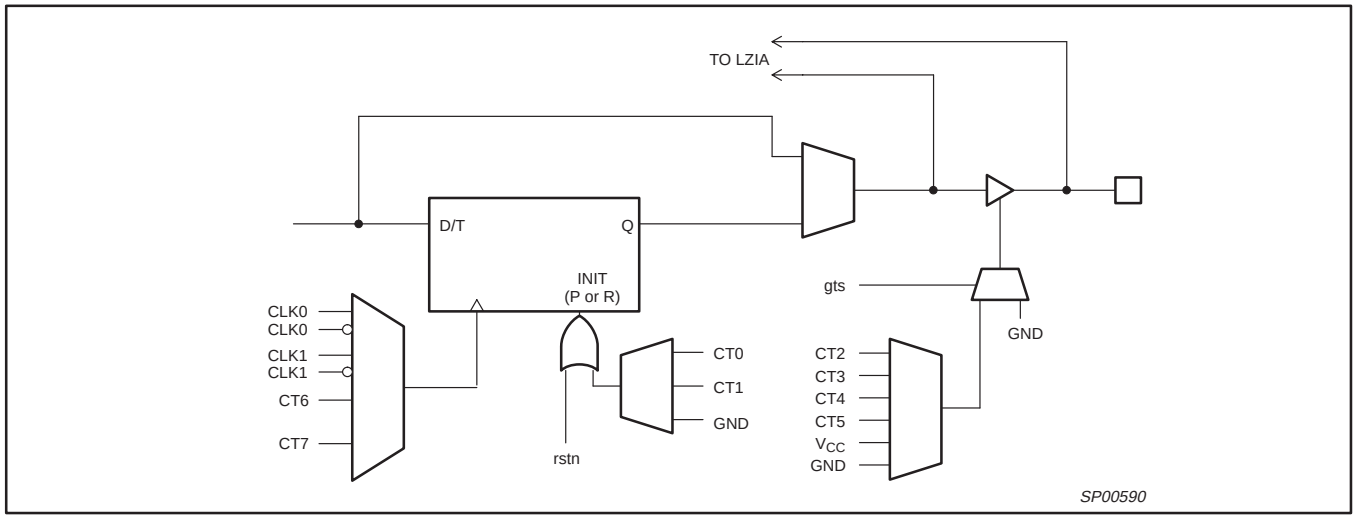


Figure 4. PZ3320 Macrocell Architecture

Simple Timing Model

Figure 5 shows the PZ3320 timing model. The PZ3320 timing model is very simple compared to the models of competing architectures. There are three main timing parameters: the pin-to-pin delay for combinatorial logic functions (t_{PD}), the input pin to register set up time (t_{SU}), and the register clock to valid output time (t_{CO}). As the model shows, timing is only dependent on whether or not you use the PLA array, and whether or not the logic function is created within a single Fast Module or uses the GZIA. The timing starts with a set time for t_{PD} and t_{SU} through the PAL array in a Fast Module, and there are fixed delays added for use of the PLA array or the GZIA. The t_{CO} timing specification never changes. For example, a combinatorial logic function of four or fewer product terms constructed from inputs within the same logic block would have a t_{PD} delay of 7.5ns. If the logic function were more than four product terms wide, the delay would be t_{PD} plus the fixed PLA delay, or 7.5 + 1.5 = 9.0ns. A function that used the PAL array and inputs from a different Fast Module would have a propagation delay of t_{PD} plus the fixed GZIA delay, or 7.5 + 4.0 = 11.5ns.

This simple timing model allows designers to determine whether or not the device will meet system timing specifications up front. In competing devices, the user is unable to determine if the design will meet system timing requirements until after the design has been fit into the device. This is because the timing models of competing architectures are very complex and include such things as timing dependencies on the number of parallel expanders borrowed, the fan-out of a signal, the varying number of X and Y routing channels used, etc. The simplicity of the PZ3320 timing model gives you pin-to-pin delay information before the design is set. Further, the timing in the PZ3320 device will not vary with place and route iterations caused by design changes. This allows the PZ3320 device to meet your timing requirements even when you make changes to the design.

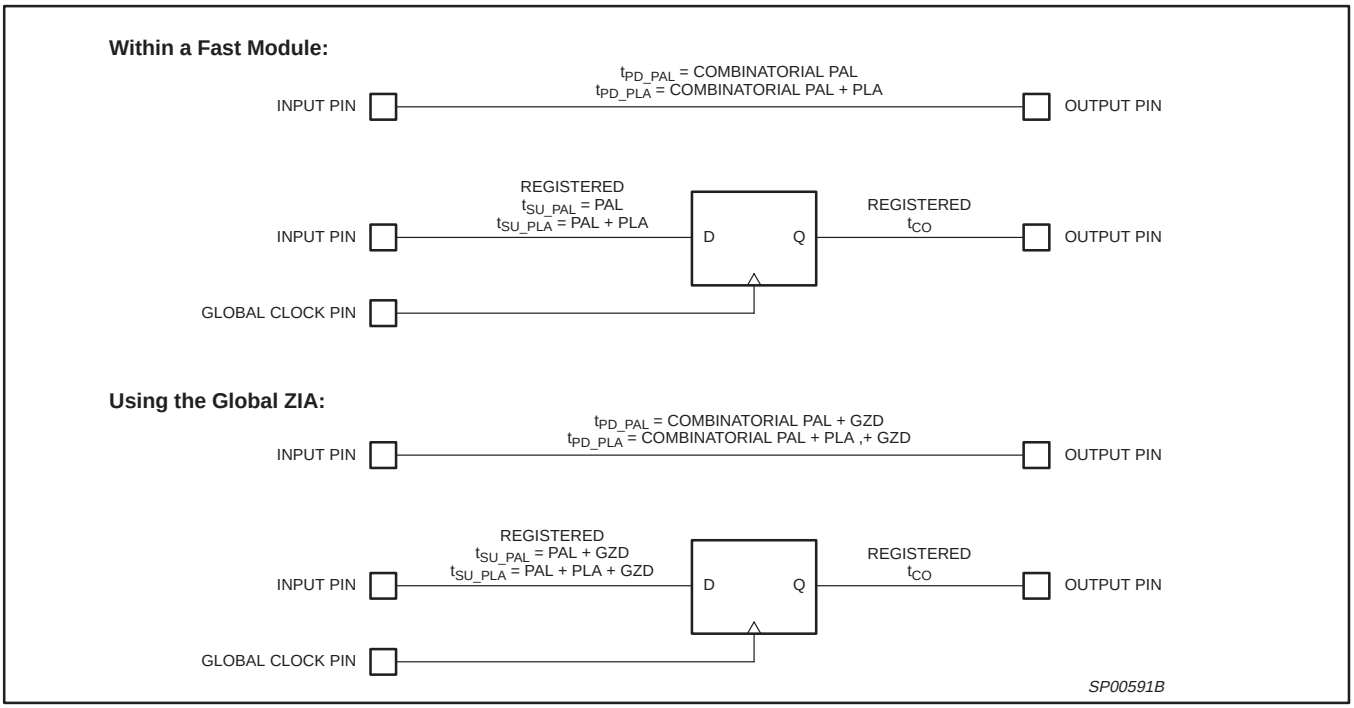


Figure 5. PZ3320 Timing Model

TotalCMOS™ Design Technique for Fast Zero Power

Philips is the first to offer a TotalCMOS™ CPLD, both in process technology and design technique. Philips employs a cascade of CMOS gates to implement its product terms instead of the traditional sense amp approach. This CMOS gate implementation allows Philips to offer CPLDs which are both high performance and low power, breaking the paradigm that to have low power, you must have low performance. This also makes it possible to manufacture high density CPLDs like the PZ3320 that consume a fraction of the power of competing devices. Refer to Figure 6 and Table 2 showing the I_{DD} vs. Frequency of the PZ3320 TotalCMOS™ CPLD (data estimated with 20 16-bit counters @ 3.3V, 25°C).

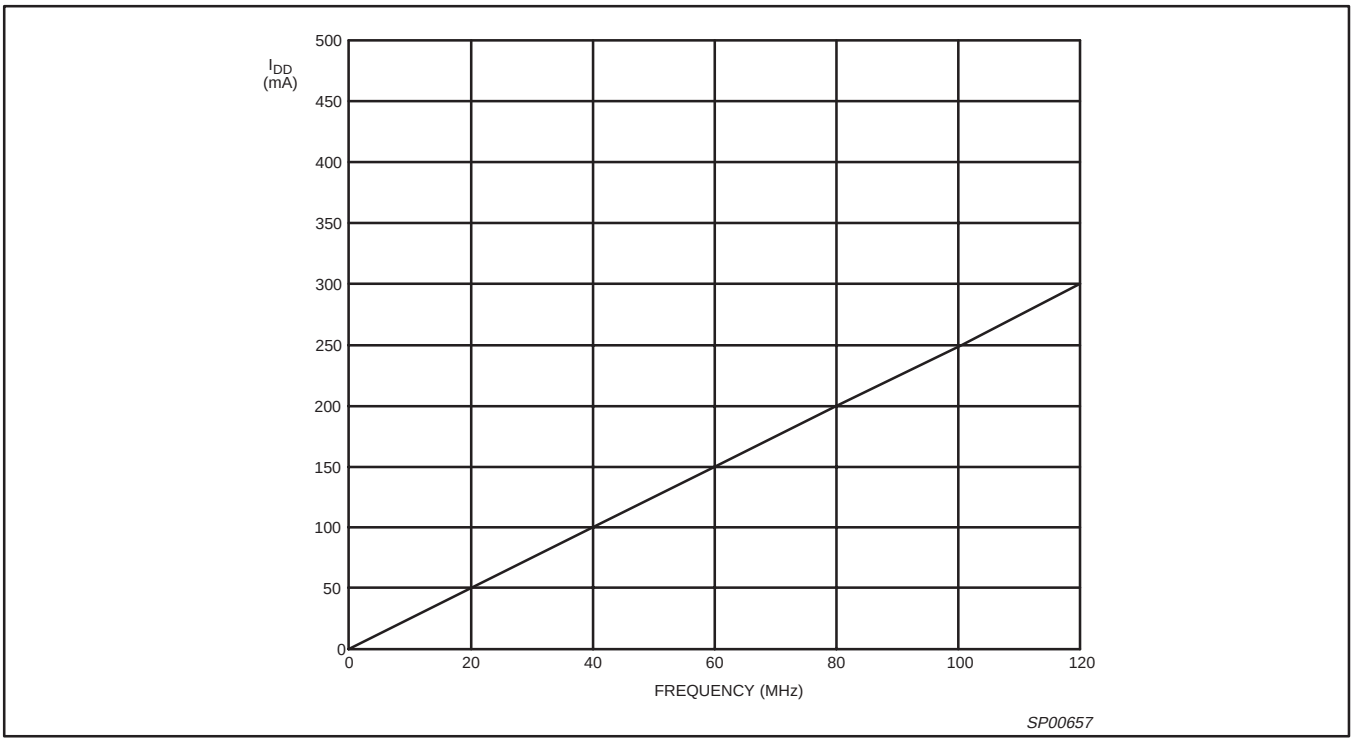


Figure 6. I_{DD} vs. Frequency @ V_{DD} = 3.3V, 25°C

Table 2. I_{DD} vs. Frequency

V_{DD} = 3.3V

FREQUENCY (MHz)	0	1	20	40	60	80	100	120
Typical I_{DD} (mA)	0.1	4.1	50	100	150	200	250	300

Terminations

The CoolRunner™ PZ3320C/PZ3320N CPLDs are TotalCMOS™ devices. As with other CMOS devices, it is important to consider how to properly terminate unused inputs and I/O pins when fabricating a PC board. Allowing unused inputs and I/O pins to float can cause the voltage to be in the linear region of the CMOS input structures, which can increase the power consumption of the device. It can also cause the voltage on a configuration pin to float to an unwanted voltage level, interrupting device operation.

The PZ3320C/PZ3320N CPLDs have programmable on-chip pull-down resistors on each I/O pin. These pull-downs are automatically activated by the fitter software for all unused I/O pins. Note that an I/O macrocell used as buried logic that does not have the I/O pin used for input is considered to be unused, and the pull-down resistors will be turned on. We recommend that any unused I/O pins on the PZ3320C/PZ3320N device be left unconnected.

There are no on-chip pull-down structures associated with dedicated pins used for device configuration or special device functions like global reset and global 3-state. Philips recommends that these pins be terminated consistent with the description given in Table 9. Philips recommends the use of weak pull-up and pull-down resistors for terminating these pins. These pins can be directly connected to V_{CC} or GND, but using the external pull-up resistors maintains maximum design flexibility.

When using the JTAG Boundary Scan functions, it is recommended that 10k pull-up resistors be used on the tdi, tdo, tck, and trstn pins. The tdo signal pin can be left floating unless it is connected to the tdi of another device. Letting these signals float can cause the voltage on tms to come close to ground, which could cause the device to enter JTAG/ISP mode at unspecified times.

CONFIGURATION INTRODUCTION

The Philips CoolRunner™ series are available in technologies which use non-volatile (EEPROM-based) and volatile (SRAM based) configuration memory. The functionality of the XPLA2 family of the CoolRunner™ series is defined by on-chip SRAM. The devices are configured in a manner similar to that of most FPGAs. This section describes the configuration of the PZ3320, and applies to all similarly configured devices to be produced by Philips.

Either the Philips or Minc fitter, XPLA Designer and PL-Designer, respectively, is used to generate a JEDEC file. The JEDEC file contains the configuration data, which is loaded into the PZ3320 configuration memory to control the PZ3320 functionality. This is done at power-up and/or with configure command. This section provides some of the trade-offs in selecting a configuration mode, and provides debug hints for configuration problems.

There are several different methods of configuring the PZ3320. The mode used is selected using the mode select pins. There are three basic configuration methods: master, slave, and peripheral. The configuration data can be transmitted to the PZ3320 serially or in parallel bytes. As a master, the PZ3320 generates the clock and control signals to strobe configuration data into the PZ3320. As a slave device, a clock is generated externally, and provided into the PZ3320's cclk pin. In the peripheral mode, the PZ3320 interfaces as a microprocessor peripheral. Table 3 lists the configuration modes.

Design Flow Overview

Figure 7 is a diagram of the steps used in configuring the PZ3320. The development system is used to generate configuration data in the JEDEC file. Using the <design>.jed file, there are two general methods of configuring the PZ3320. The utility **download** can load the configuration data from a PC or workstation hard disk into the PZ3320. This is one of the methods used on the PZ3320 evaluation board. Alternately, the PZ3320 can be loaded from non-volatile ICs such as serial or parallel EEPROMs.

Table 3. Configuration Modes

M2	M1	M0	cclk	CONFIGURATION MODE	DATA FORMAT
			Output	Master serial	Serial
			Input	Slave parallel	Parallel
			Reserved		
			Input	Synchronous peripheral	Parallel
			Output	Master parallel – up	Parallel
			Reserved		
			Output	Master parallel – down	Parallel
			Input	Slave serial	Serial

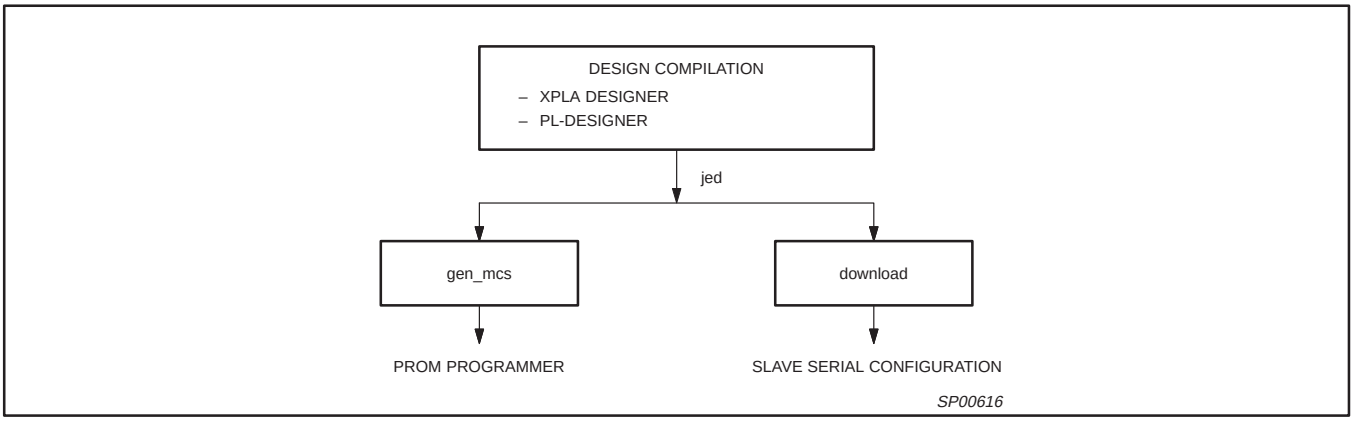


Figure 7. Design flow

PZ3320 STATES OF OPERATION

Prior to becoming operational, the PZ3320 goes through a sequence of states, including initialization, configuration, and start-up. This section discusses these three states. In the master configuration modes, the PZ3320 is the source of configuration clock (cclk). In this mode, the Initialization state is extended to ensure that, in daisy-chain operation, all daisy-chained slave devices are ready.

When configuration is initiated, a counter in the PZ3320 is set to 0 and begins to count configuration clock cycles applied to the PZ3320. As each configuration data frame is supplied to the PZ3320, it is internally assembled into data words. Each data word is loaded into the internal configuration memory. The configuration loading process is complete when the internal length count equals the loaded length count in the length count field, and the required end of configuration frame is written.

All configuration I/Os used as inputs operate with TTL-level input thresholds during configuration. All I/Os that are not used during the configuration process are 3-Stated with internal pull-downs. During configuration, registers are reset. The combinatorial logic begins to function as the PZ3320 is configured. Figure 8 shows the flow between the initialization, configuration, and start-up states. Figure 9 gives the general timing information for configuring the device.

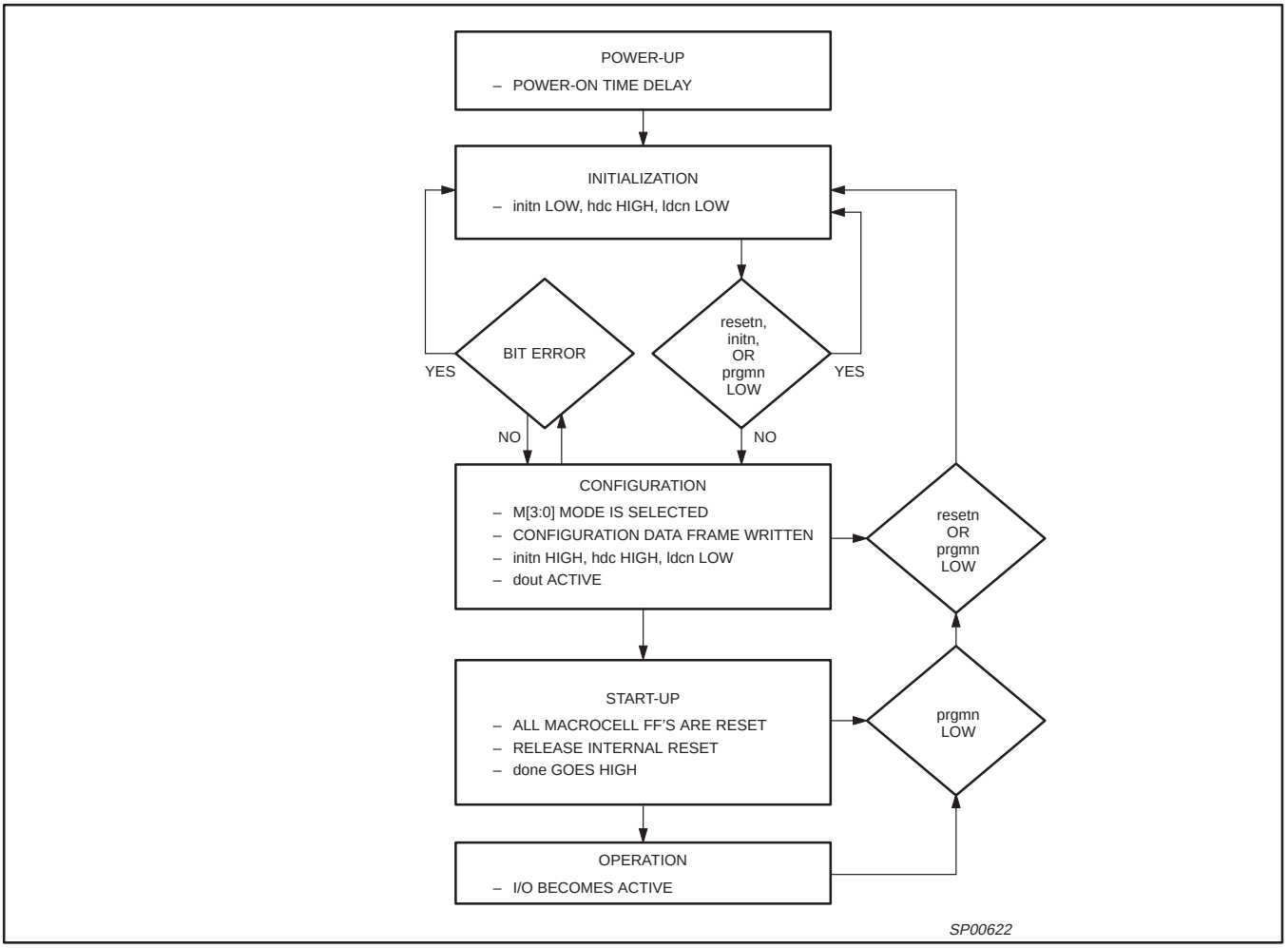


Figure 8. Flow chart of initialization, configuration, and operating states

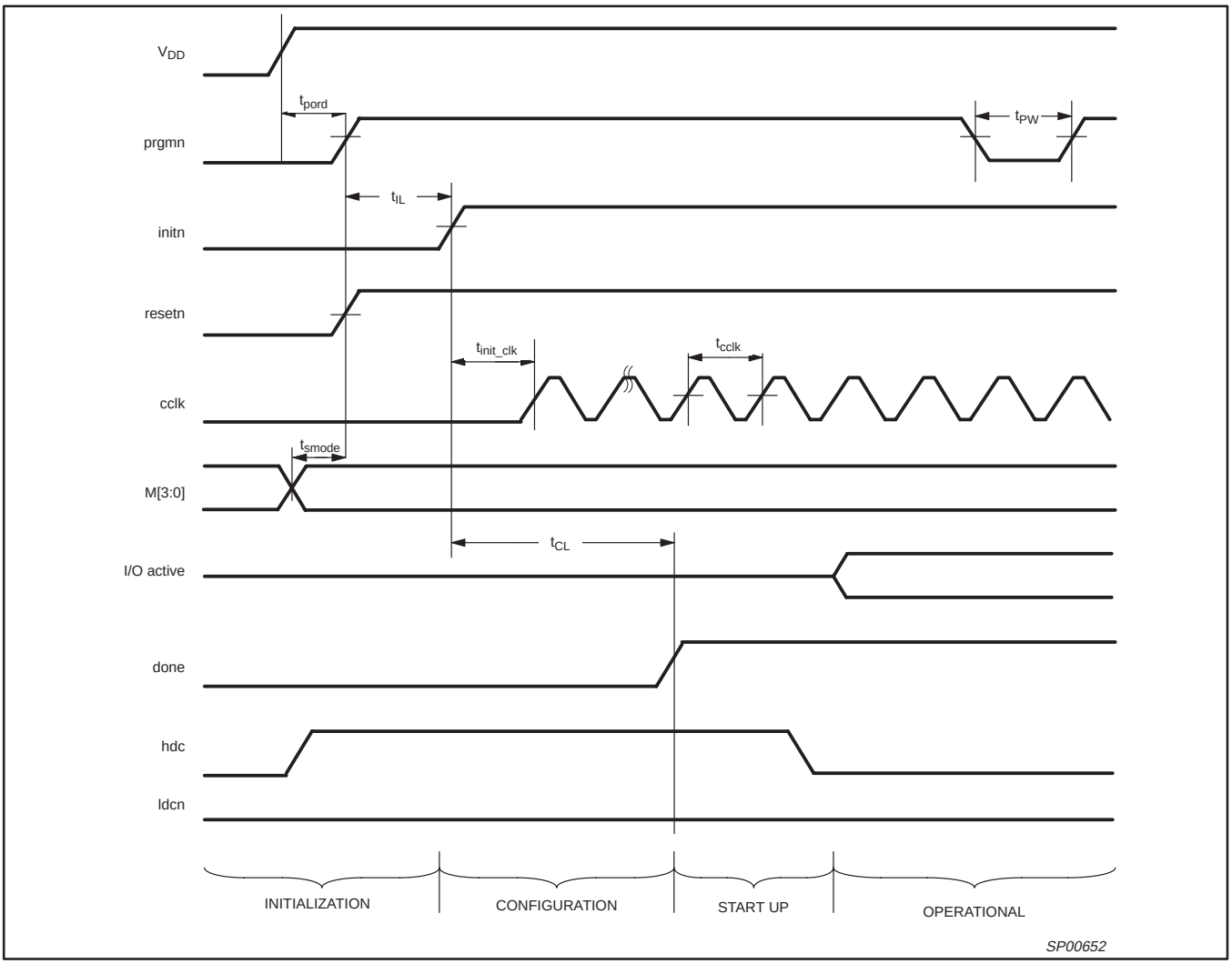


Figure 9. General configuration mode timing diagram

Initialization

Upon power-up, the device goes through an initialization process. First, an internal power-on-reset circuit is triggered when power is applied. When V_{DD} reaches the voltage at which portions of the PZ3320 begin to operate (1.5V), the configuration pins are set to be inputs or outputs based on the configuration mode, as determined by the mode select inputs M[2:0]. A time-out delay is initiated when V_{DD} reaches between 1.0V and 2.0V to allow the power supply voltage to stabilize. The initn and done outputs are low. At power-up, if the power supply does not rise from 1.0V to V_{DD} in less than 25ms, the user should delay configuration by inputting a low into prgmn, initn, or resetn until V_{DD} is greater than the recommended minimum operating voltage (2.75V for commercial devices).

When initialization is complete, the active-low initialization signal initn is released and must be pulled high by an external resistor. To synchronize the configuration of multiple PZ3320s, one or more initn pins should be wire-ANDed. If initn is held low by one or more PZ3320s or an external device (the PZ3320 remains in the initialization state), initn can be used to signal that the PZ3320s are not yet initialized. After initn goes high for two internal clock cycles, the mode select lines are sampled and the PZ3320 enters the configuration state.

The High During Configuration (hdc), Low During Configuration (ldcn), and done signals are active outputs in the PZ3320's initialization and configuration states. hdc, ldcn, and done can be used to provide control of external logic signals such as reset, bus enable, or EEPROM enable during configuration. For master parallel configuration modes, these signals provide EEPROM enable control and allow the data pins to be shared with user logic signals.

If configuration has begun, an assertion of resetn or prgmn initiates an abort, returning the PZ3320 to the initialization state. The resetn and/or prgmn pins must be pulled back high before the PZ3320 will enter the configuration state. During the start-up and operating states, only the assertion of prgmn causes a re-configuration.

During initialization and configuration, all I/O's are 3-stated and the internal weak pull-downs are active. See the section on terminations for more information.

Start-up

After configuration, the PZ3320 enters the start-up phase. This phase is the transition between the configuration and operational states. This transition occurs within three cclk cycles of the done pin going high (it is acceptable to have additional cclk cycles beyond the three required). The system design task in the start-up phase is to ensure that multi-function pins (see pin function on page 34) transition from configuration signals to user definable I/Os without inadvertently activating devices in the system or causing bus contention. The done signal goes high at the beginning of the start up phase, which allows configuration sources to be disconnected so that there is no bus contention when the I/Os become active. In addition to controlling the PZ3320 during start-up, additional start-up techniques to avoid contention include using isolation devices between the PZ3320 and other circuits in the system, re-assigning I/O locations, and keeping I/Os 3-stated until contentions are resolved. For example, Figure 10 shows how to use the global tri-state (gts) signal to avoid signal contention when the mode select pins (M3...M0) are used as I/O after configuration is finished. Holding gts high until after the mode pins are disconnected from the driving source allows pins M3 through M0 to transition from configuration pins to user definable I/O without signal contention. In this case, the I/O become active a t_{gtsr} delay after the gts pin is pulled low.

The flip-flops are reset one cycle after done goes high so that operation begins in a known state. The done outputs from multiple PZ3320s can be wire ANDed and used as an active-high ready signal, to disable PROMs with active-low enable(s), or to reset to other parts of the system (see Figure 28).

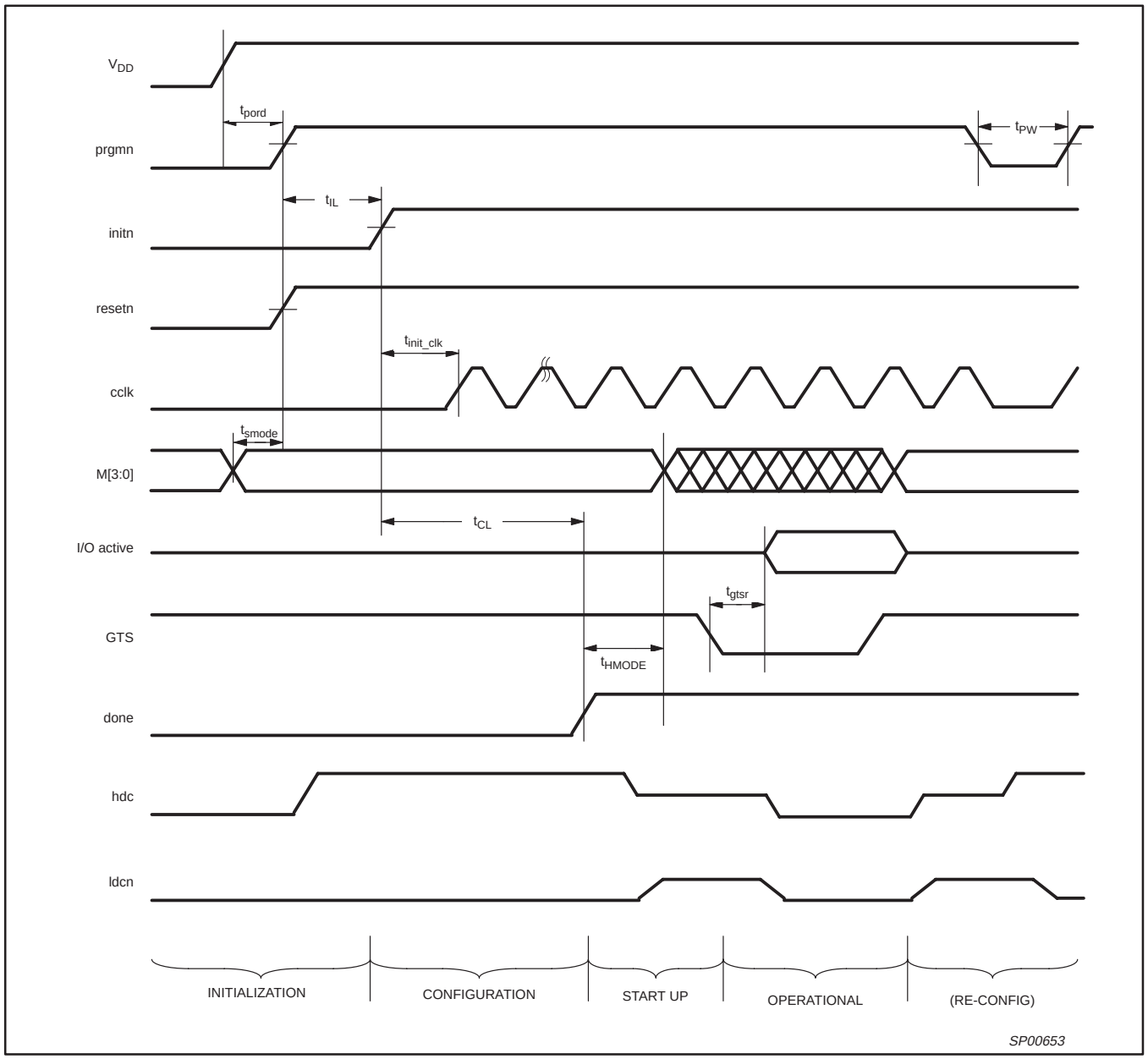


Figure 10. Using gts signal with power up to avoid signal contention with mode select pins

CONFIGURATION DATA FORMAT OVERVIEW

The PZ3320 functionality is determined by the state of internal configuration RAM. This section discusses the configuration data format, and the function of each field in configuration data packets.

Configuration Data Packets

Configuration of the PZ3320 is done using configuration packets. The configuration packet is shown in Figure 11. The data packet consists of a header and a data frame. There are five type of data frames. The header is shifted into the device first, followed by one data frame. Configuration of a single PZ3320 requires 1010 data packets, one for each address. All preceding data must contain only 1s. Once a device is configured, it re-transmits data of any polarity. Before and during configuration, all data re-transmitted out the daisy-chain port (dout) are 1s.

The ordering of the data packets may be random, but they cannot be mixed with other devices' data packets. Alignment bits are not required between data packets. If used, alignment bits must be included in the length count, and they must be at least 2 bits long.

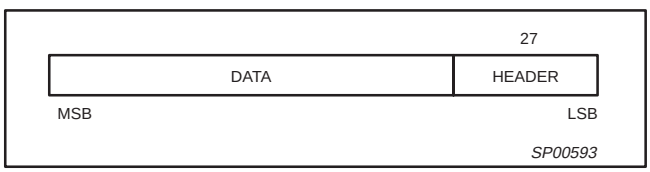


Figure 11. Data Packet

Table 4. Configuration Frame Size

DEVICE	PZ3320
Number of frames	
Data bits/standard frame	
Data bits/compressed frame	
Data bits/user_code frame	
Data bits/isc_code frame	
Data bits/security frame	
Maximum configuration data— # bits/frame × # frames	

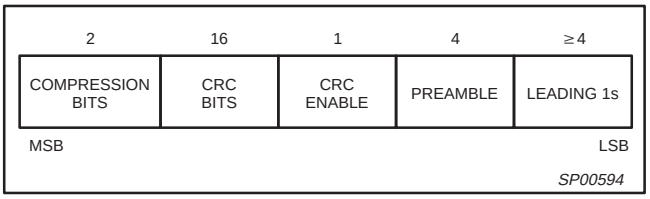


Figure 12. 27-bit Header

The header is fixed and consists of five fields:
– Leading 1s,
– Preamble,
– CRC Enable,
– CRC Bits,
– Compression Bits.

The leading 1s enter the device first. The following is a description of each field in the header.

Leading 1s:
This is a four or greater bit field consisting of 1s.

Preamble:
This is a four bit field which indicates the start of a frame when the least significant bit of the preamble is a 0.
There are two valid preambles:
0010 – indicates that the data packet configures the device receiving the 0010 preamble)
0100 – indicates end of configuration of the device receiving the 0100 preamble
All other values of the preamble field force configuration of the entire system to restart.

The segments CRC Enable, CRC Bits, and Compression Bits are valid only if the Preamble field is 0010.

Cyclic Redundancy Check (CRC) Enable:
In this single bit field, a 0 disables CRC checking of the data stream. If the CRC is disabled the 16 bit CRC field must be the default described below. A 1 enables CRC error checking of the data stream.

CRC Error Checking:
The CRC field is a 16 bit field. The default value is 1010_1010_1010_1010. The calculated value is from data, address, stop bit, and first alignment bit (starting with crc_reg[15:0] = [0]). Using verilog operators, the crc is calculated as:

```
crc_reg[14:2] <= cr_reg[14:2] << 1;
cr_reg[2] <= cr_reg[15]^din^cr_reg[1];
cr_reg[1] <= cr_reg[0];
cr_reg[0] < cr_reg[15]^din;
cr_reg[15] <= cr_reg[15]^din^cr_reg[14];
```

If a CRC error is detected, configuration is halted and must be restarted.

Compression Bits:
This 2-bit field defines the use of compression of the data packets.
00 – Standard mode:
The data packet contains both address and data
01 – Reset mode:
The data packet contains only the address field.
This pattern causes the configuration register to be reset.
10 – Hold mode:
The data packet contains only the address field.
This pattern causes the configuration register to hold its value.
11 – Set mode:
The data packet contains only the address field.
This pattern causes the configuration register to be set.

Data Frames

The five types of data frames are standard, compressed, user_code, isc_code, and security. All fields must be completely filled, with 1s used to fill unused bits. The security frame must be the last frame sent to a device. The definition of each frame is described below:

Standard frame

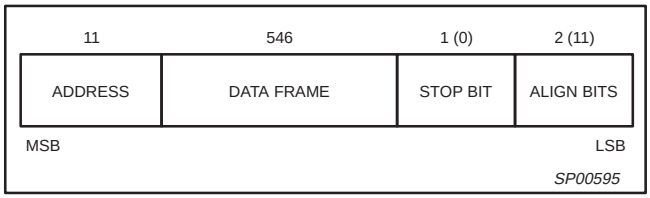


Figure 13. Standard Frame

Address:
This is an 11 bit filed for providing 1011 (1008 SRAM plus 3 user) addresses.

Data:
546 bit field.

Stop bit:
This is a one bit field which must be 0.

Align bit:
This is a two bit field which must be 11.

Compressed frame

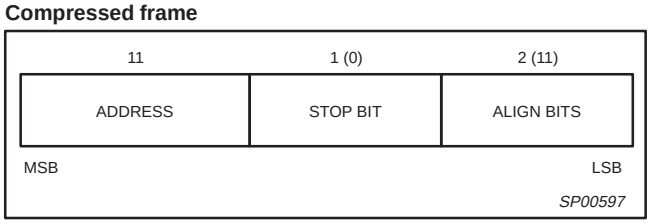


Figure 14. Compressed Frame

The compressed frame contains no data.

User code frame
The user code is located at address 1008D.

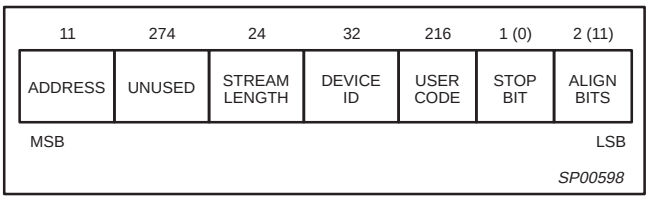


Figure 15. User code Frame

Stream length:
This is a 24 bit field containing the length of the data stream transmitted to configure all of the devices in the daisy chain. This field is only used by a PZ3320 if it is in the master mode.

Device ID:
This is a 32-bit field containing PZ3320 device ID:
492 SBGA: 0000_001_001_101000_1_000_00000010101_1

User code:
This is a 216 bit field reserved for user information.

ISC code frame
The isc_code address is 1009.

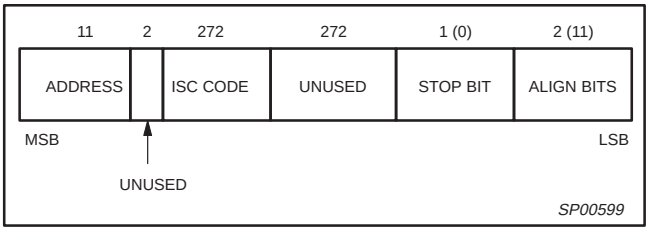


Figure 16. ISC Frame

The ISC frame allows the user to write an ISC code to the device.

Security frame

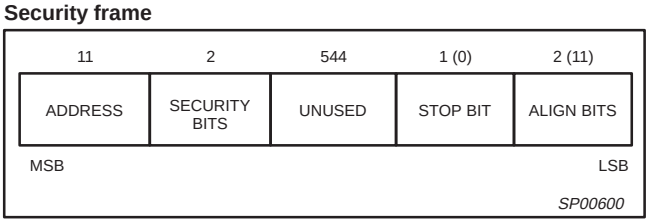


Figure 17. Security Frame

Security bits:
This is a two bit field specifying the level of security.
00 – Unlimited readback allowed.
01 – Readback operation allowed once.
10 – Readback operation allowed once.
11 – Readback operation is disabled.

Re-configuration

To reconfigure the PZ3320 when the device is operating in the system, a low pulse is input into prgmn. The configuration data in the PZ3320 is cleared, and the I/Os not used for configuration are 3-Stated. The PZ3320 then samples the mode select inputs and begins re-configuration. When configuration is compete, done is released, allowing it to be pulled high.

Bit Stream Error Checking

There are three different types of bit stream error checking in the PZ3320:
– ID frame,
– Frame alignment, and
– CRC checking.

An optional ID data frame can be sent to a specified address in the PZ3320. This ID Frame contains a unique code which is compared with the value in the PZ3320 ID register. Any differences are flagged as an ID error.

CRC checking is done on each frame if enabled by setting the CRCen bit in the header. If there is an error, a CRC error is flagged. When an error occurs, the PZ3320 is forced into the initialization state, forcing initn low. The PZ3320 remains in this state until either the resetn or prgmn pins is asserted.

PZ3320 CONFIGURATION MODES

The method for configuring the PZ3320 is selected by the M0, M1, and M2 inputs. The M3 input is used to select the frequency of the internal oscillator, which is the source for cclk in master configuration modes. The nominal frequencies of the internal oscillator are 1.25MHz and 10MHz. The 1.25MHz frequency is selected when the M3 input is unconnected or driven to a high state.

Master Serial Mode

In the master serial mode, the PZ3320 loads the configuration data from an external serial ROM. The configuration data is either loaded automatically at start-up or on a command to reconfigure. Serial EEPROMs from Altera, Atmel, Lucent, Microchip, and Xilinx can be used to configure the PZ3320 in the master serial mode. This provides a simple four-pin interface in an eight-pin package. Serial EEPROMs are available in 32K, 64K, 128K, 256K, and 1M bit densities.

Configuration in the master serial mode can be done at power-up and/or upon a configure command. The system or the PZ3320 must activate the serial EEPROM's $\overline{\text{RESET}}$/OE and $\overline{\text{CE}}$ inputs. At power-up, the PZ3320 and serial EEPROM each contain internal power-on reset circuitry which allows the PZ3320 to be configured without the system providing an external signal. The power-on reset circuitry causes the serial EEPROMs' internal address pointer to be reset. After power-up, the PZ3320 automatically enters its initialization phase.

The serial EEPROM/PZ3320 interface used depends on such factors as the availability of a system reset pulse, availability of an intelligent host to generate a configure command, whether a single serial EEPROM is used or multiple serial ROMs are cascaded, whether the serial EEPROM contains a single or multiple configuration programs, etc.

Data is read into the PZ3320 sequentially from the serial ROM. The DATA output from the serial EEPROM is connected directly into the din input of the PZ3320. The cclk output from the PZ3320 is connected to the CLOCK input of the serial EEPROM. During the configuration process, cclk clocks one data bit into the PZ3320 on each rising edge.

Since the data and clock are direct connects, the PZ3320/serial EEPROM interface task is to use the system or PZ3320 to enable the $\overline{\text{RESET}}$/OE and $\overline{\text{CE}}$ of the serial EEPROM(s). There are several methods for enabling the serial ROM's $\overline{\text{RESET}}$/OE and $\overline{\text{CE}}$ inputs. The serial EEPROM's $\overline{\text{RESET}}$/OE is programmable to function with

resetn active-high and $\overline{OE}$ active-low, or resetn active-low and OE active-high.

In Figure 18, three serial EEPROMs are cascaded to configure a PZ3320. The host generates a 500ns low pulse into the PZ3320's prgmn input and into the serial EEPROMs' $\overline{RESET}$/OE input, which has been programmed to function with resetn active-low and OE active-high. The PZ3320 done is routed to the $\overline{CE}$ pin. The low on done enables the serial EEPROMs. At the completion of configuration, the high on the PZ3320's done disables the EEPROM(s).

When configuration data requirements exceed the capacity of a single serial EEPROM, multiple serial EEPROMs can be cascaded to support the configuration of a single (or multiple) PZ3320(s). After the last bit from the first serial ROM is read, the serial ROM outputs $\overline{CEO}$ low and 3-States the DATA output. The next serial ROM recognizes the low on $\overline{CE}$ input and outputs configuration data on the DATA output. After configuration is complete, the PZ3320's done output into $\overline{CE}$ disables the serial EEPROMs.

In applications in which a serial EEPROM stores multiple configuration programs, the subsequent configuration program(s) are stored in EEPROM locations that follow the last address for the previous configuration program. The user must ensure that a high output on the PZ3320 done signal does not reset the serial EEPROM address pointer, causing the first configuration to be reloaded.

Contention on the PZ3320's din pin must be avoided. During configuration, din receives configuration data. After configuration, it is a user I/O at start-up. An alternative is to use ldcn to drive the serial EEPROMs' $\overline{CE}$ pin.

Master Parallel Mode

The master parallel configuration mode is generally used to interface to industry-standard byte-wide memory such as 256K and larger EEPROMs. Figure 20 provides the interface for master parallel mode. The PZ3320 outputs a 20-bit address on A[19:0] to memory and reads one byte of configuration data on the rising edge of rclk. The parallel bytes are internally serialized starting with the least significant bit, D0. There are two parallel master modes: master up, and master down. In master up, the starting memory address is 00000 Hex and the PZ3320 increments the address for each byte loaded. In master down, the starting memory address is FFFFFH and the PZ3320 decrements the address.

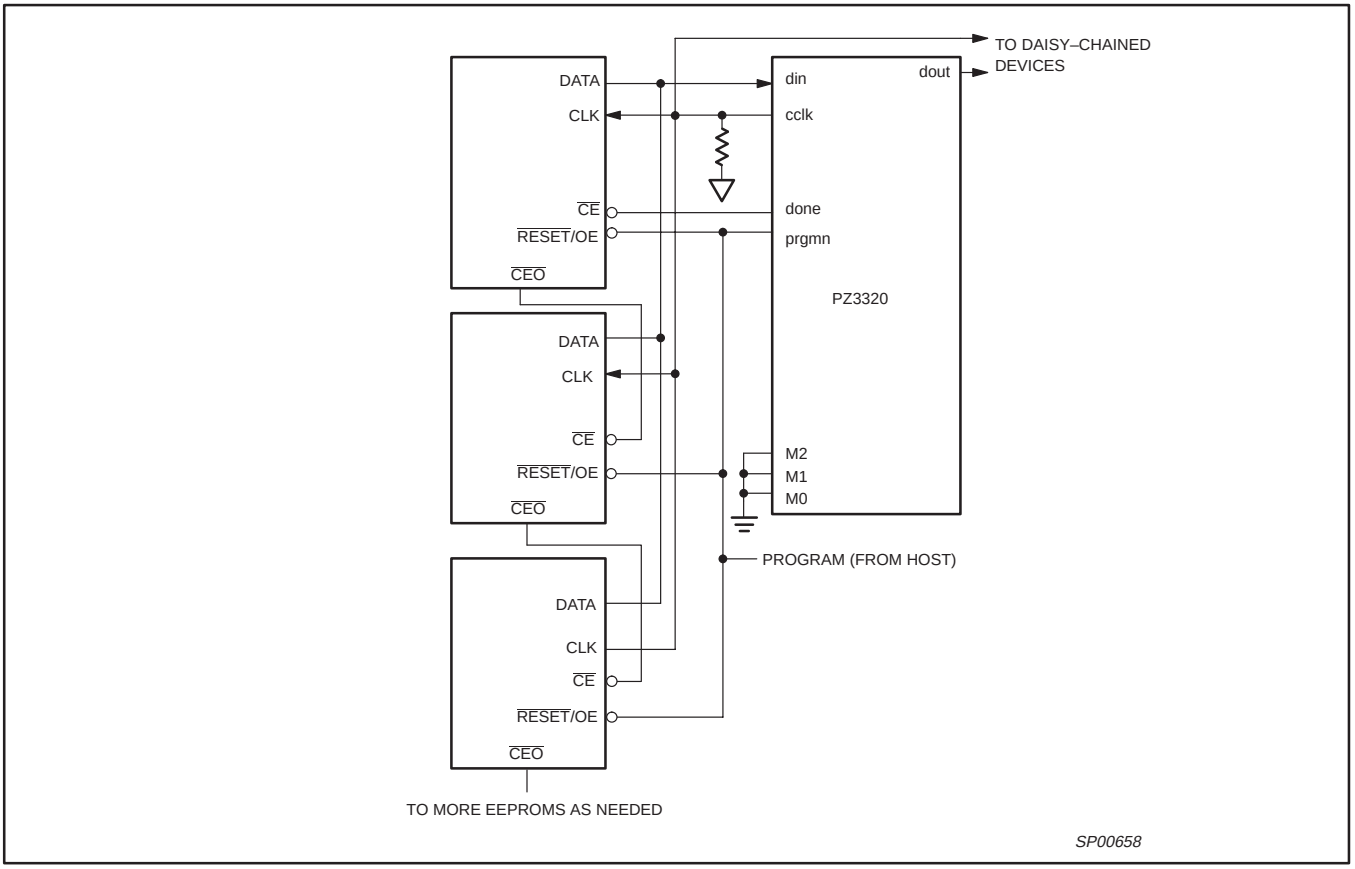


Figure 18. Master Serial Configuration

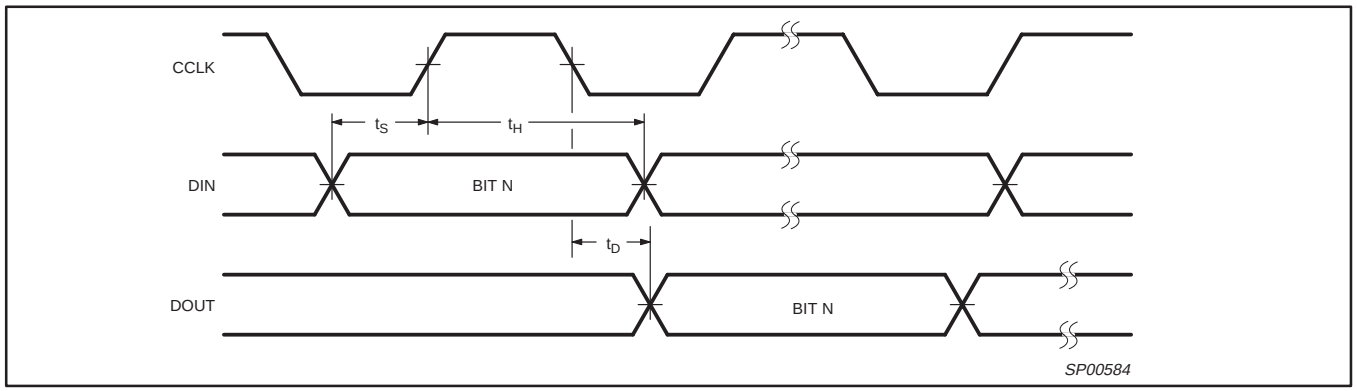


Figure 19. Master Serial Configuration Mode Timing Diagram

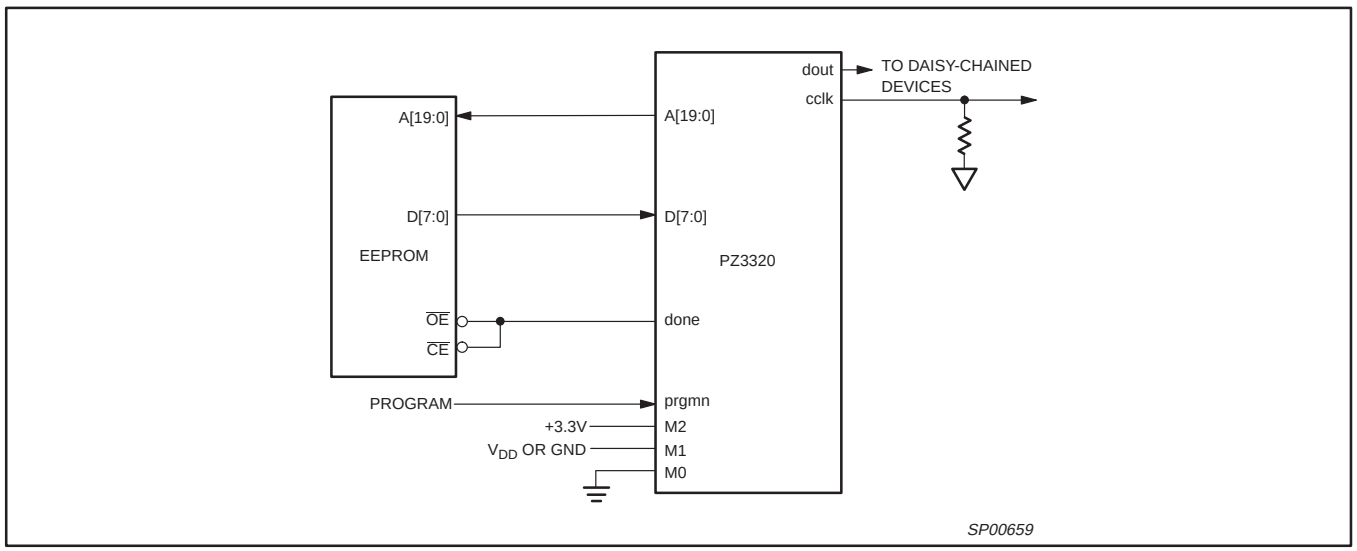


Figure 20. Master Parallel Configuration

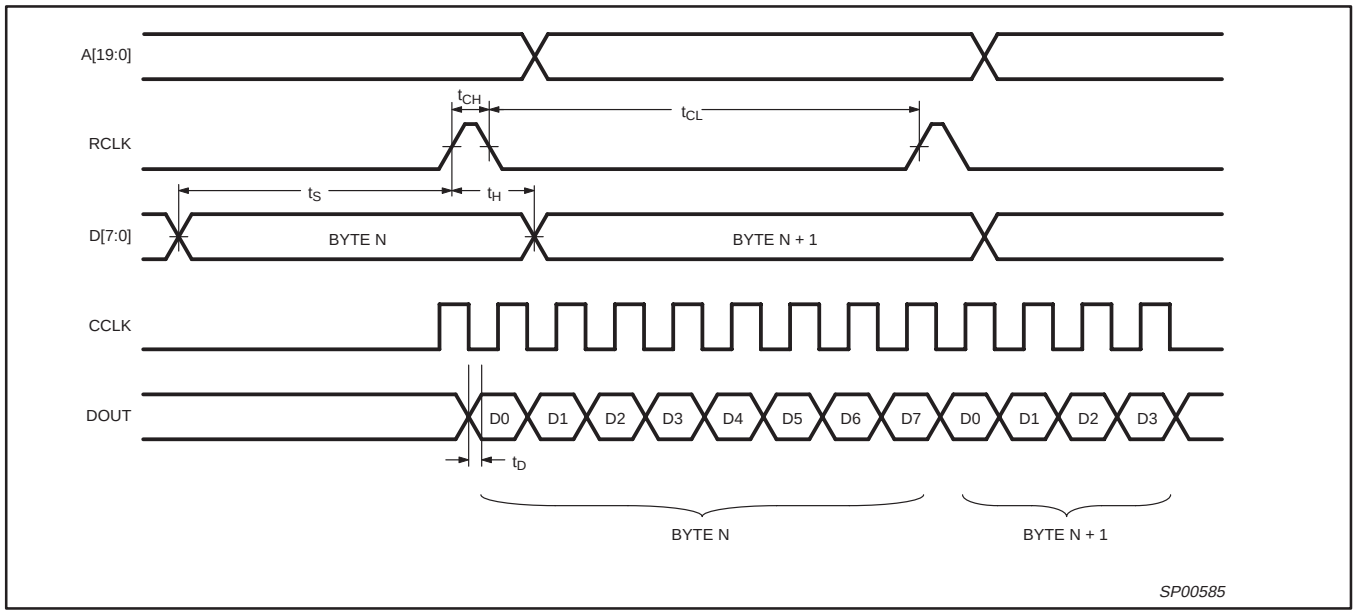


Figure 21. Master Parallel Configuration Mode Timing Diagram

Synchronous Peripheral Mode

In the synchronous peripheral mode, byte-wide data is input into D[7:0] on the rising edge of the cclk input. The first data byte is clocked in on the second cclk after initn goes high. Subsequent data bytes are clocked in on every eighth rising edge of cclk. The rdy_busyn signal is an output which acts as an acknowledge. rdy_busyn goes high one cclk after a byte of data is clocked in on D[7:0] and returns low one cclk cycle later. The process repeats until all of the data is loaded into the PZ3320. The serial data begins shifting out on dout 0.5 cycles after the parallel data was loaded. It requires additional cclks after the last byte is loaded to complete the shifting. Figure 22 shows the interface for synchronous peripheral mode.

As with master modes, the peripheral modes can be used as the lead PZ3320 for daisy-chained PZ3320s.

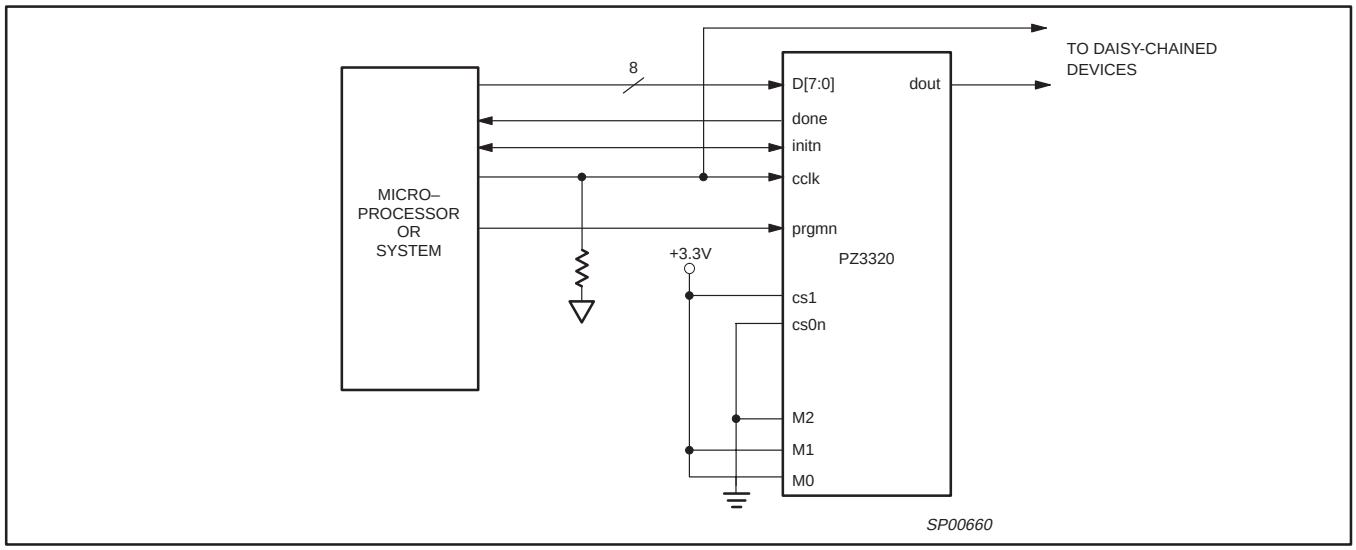


Figure 22. Synchronous Peripheral Configuration

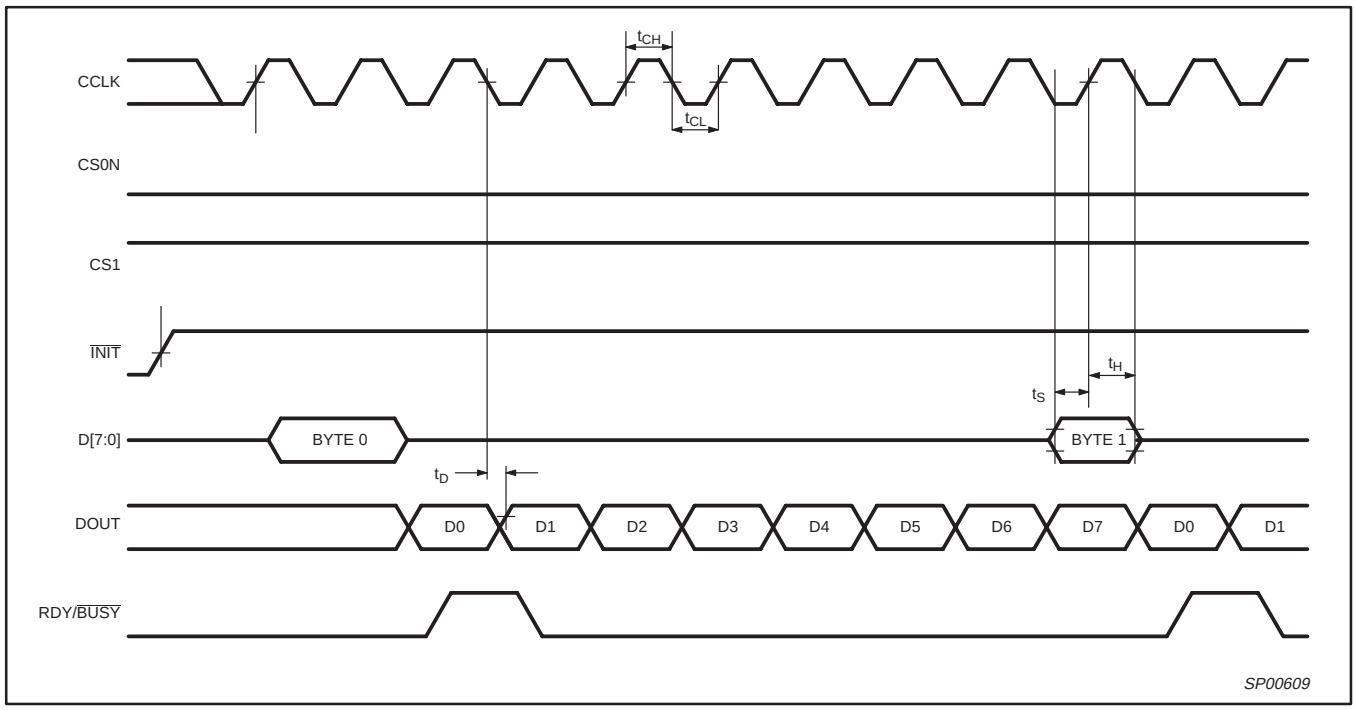


Figure 23. Synchronous Peripheral Configuration Mode Timing Diagram

Slave Serial Mode

The slave serial mode is primarily used when multiple PZ3320s are configured in a daisy-chain. The serial slave serial mode is also used on the PZ3320 evaluation board, which interfaces to the download cable. A device in the slave serial mode can be used as the lead device in a daisy-chain. Figure 24 shows the interface for the slave serial configuration mode.

The configuration data is provided into the PZ3320's din input synchronous with the configuration clock cclk input. After the PZ3320 has loaded its configuration data, it re-transmits incoming configuration data on dout. When used in daisy-chained operation, cclk is routed into all slave serial mode devices in parallel.

Multiple slave PZ3320s can be loaded with identical configurations simultaneously. This is done by loading the configuration data into the din inputs in parallel.

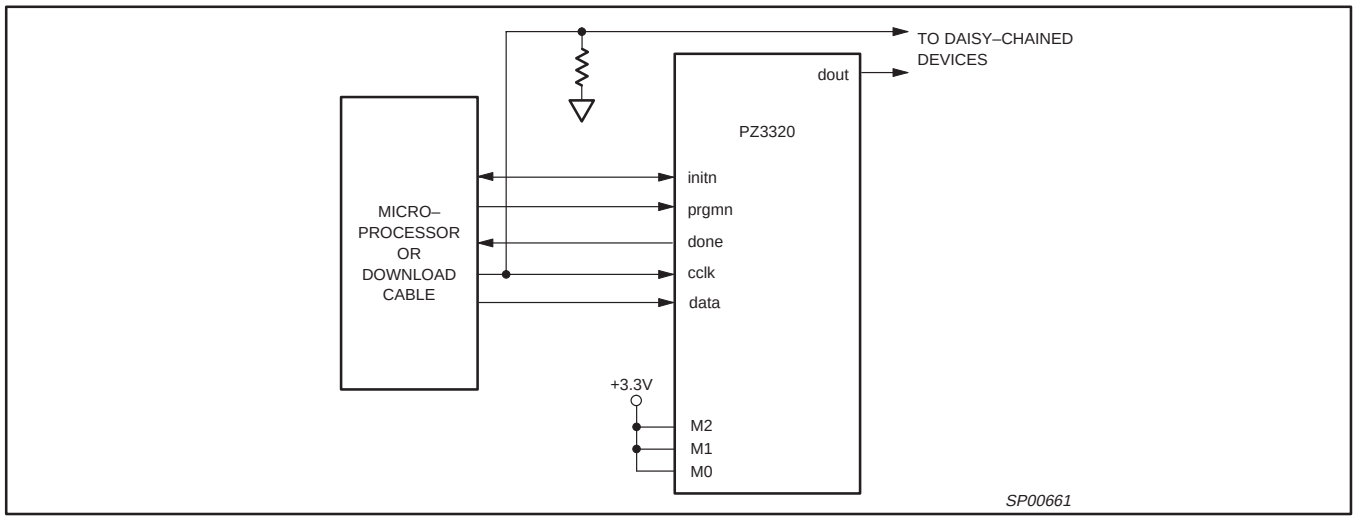


Figure 24. Slave Serial Configuration Schematic

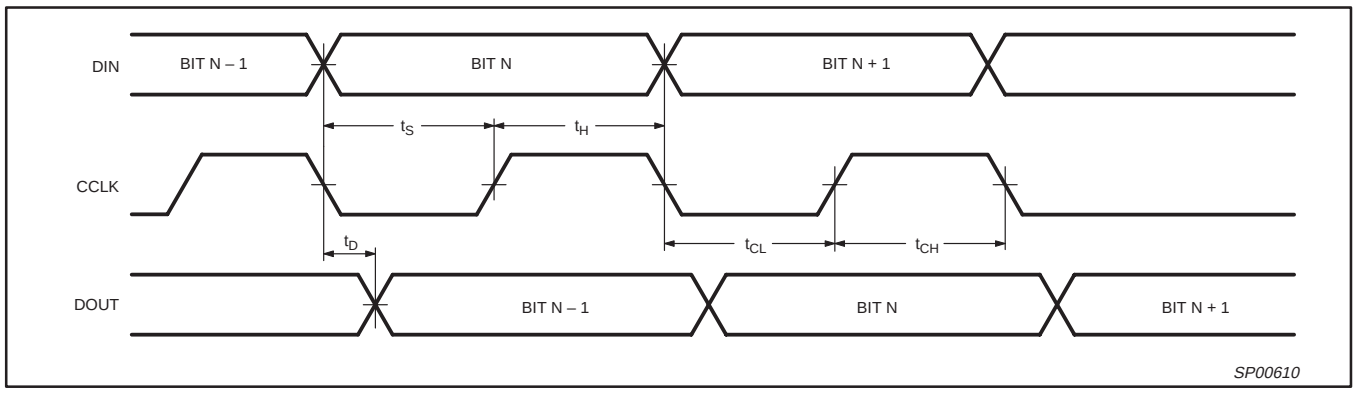


Figure 25. Slave Serial Configuration Mode Timing Diagram

Slave Parallel Mode

The slave parallel mode is essentially the same as the synchronous peripheral mode, except that cs1 and cs0n do not need to be driven, and there is no rdy_bsyn output. As in the synchronous peripheral mode, byte-wide data is input into D[7:0] on the rising edge of the cclk input. The first data byte is clocked in on the second cclk after initn goes high. Subsequent data bytes are clocks in on every eighth rising edge of cclk. The process repeats until all of the data is loaded into the PZ3320. The serial data begins shifting out on dout 0.5 cycles after the parallel data was loaded. It requires additional cclks after the last byte is loaded to complete the shifting. Figure 26 shows the interface for slave parallel mode.

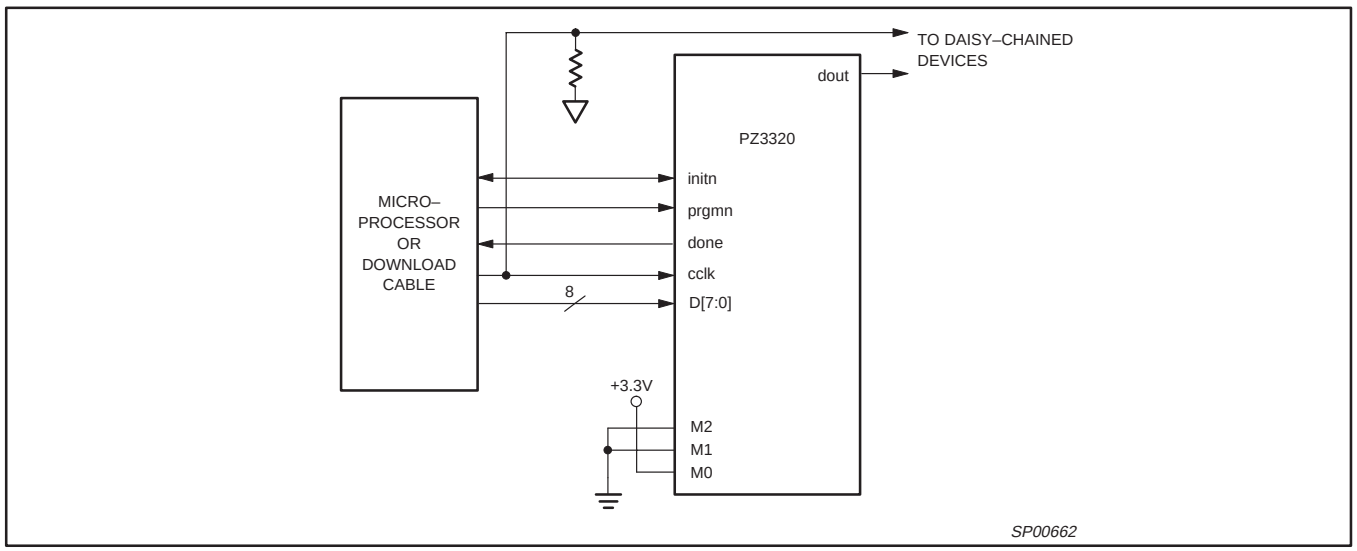


Figure 26. Slave Parallel Configuration Schematic

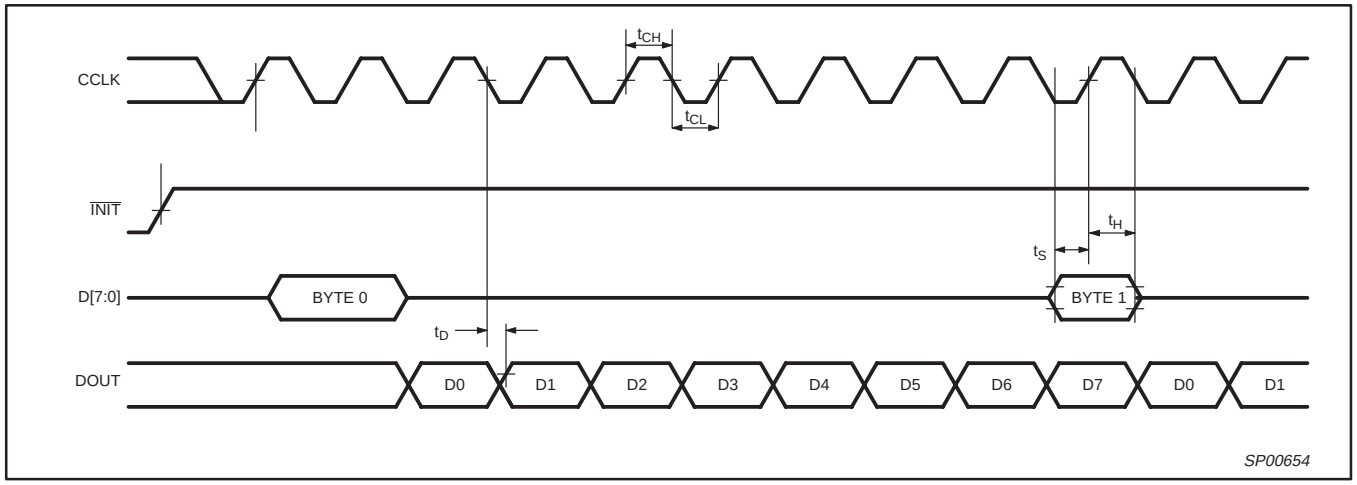


Figure 27. Slave Parallel Configuration Mode Timing Diagram

DAISY CHAIN OPERATION

Multiple PZ3320s can be configured by using a daisy-chain of PZ3320s. Daisy-chaining uses a lead PZ3320 and one or more PZ3320s configured in slave serial mode. The lead PZ3320 can be configured in any mode, but master parallel is typically used. Figure 28 shows the connections for loading multiple PZ3320s in a daisy-chain configuration.

Daisy-chained PZ3320s are connected in series. An upstream PZ3320 which has received the preamble outputs a high on dout until it has received the appropriate number of data frames. This ensures that downstream PZ3320s do not receive frame start bits. After loading and re-transmitting the preamble to a daisy-chain of slave devices, the lead device loads its configuration data frames. The loading of configuration data continues after the lead device has received its configuration data if the lead device's internal frame bit counter has not reached the length count. When the configuration RAM is full and the number of bits received is less than the length count field, the PZ3320 shifts data out on dout.

The generation of cclk for the daisy-chained devices which are in slave serial mode differs depending on the configuration mode of the lead device. A master parallel mode device uses its internal timing generator to produce an internal cclk at eight times its memory address rate (rclk). If the lead device is configured in either synchronous peripheral, slave serial mode, or slave parallel mode, cclk is routed to the lead device and to all of the daisy-chained devices. The configuration data is read into din of slave devices on the positive edge of cclk, and shifted out dout on the negative edge of cclk.

The development software can create a composite configuration file for configuring daisy-chained PZ3320s. The configuration data consists of multiple concatenated data packets. As seen in Figure 28, the initn pins for all of the PZ3320s are connected together. This is required to guarantee that power-up and initialization function correctly. In general, the done pins for all of the PZ3320s are also connected together as shown to guarantee that all of the PZ3320s enter the start-up state simultaneously. This may not be required, depending upon the start-up sequence desired.

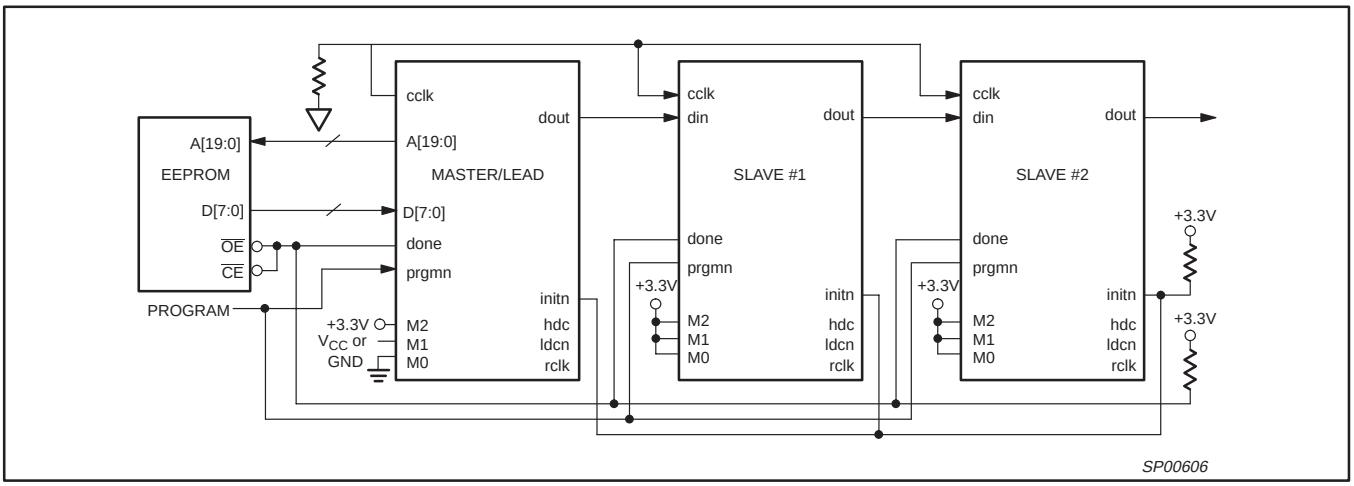


Figure 28. Daisy-chain Schematic

JTAG Testing Capability

JTAG is the commonly-used acronym for the Boundary Scan Test (BST) feature defined for integrated circuits by IEEE Standard 1149.1. This standard defines input/output pins, logic control functions, and commands which facilitate both board and device level testing without the use of specialized test equipment. BST provides the ability to test the external connections of a device, test the internal logic of the device, and capture data from the device during normal operation. BST provides a number of benefits in each of the following areas:

- Testability
  - Allows testing of an unlimited number of interconnects on the printed circuit board
  - Testability is designed in at the component level
  - Enables desired signal levels to be set at specific pins (Preload)
  - Data from pin or core logic signals can be examined during normal operation
- Reliability
  - Eliminates physical contacts common to existing test fixtures (e.g., “bed-of-nails”)
  - Degradation of test equipment is no longer a concern
  - Facilitates the handling of smaller, surface-mount components
  - Allows for testing when components exist on both sides of the printed circuit board
- Cost
  - Reduces/eliminates the need for expensive test equipment
  - Reduces test preparation time
  - Reduces spare board inventories

The Philips PZ3320’s JTAG interface includes a TAP Port and a TAP Controller, both of which are defined by the IEEE 1149.1 JTAG Specification. As implemented in the Philips PZ3320, the TAP Port includes five pins (refer to Table 5) described in the JTAG specification: tck, tms, tdi, tdo, and trstn. These pins should be connected to an external pull-up resistor to keep the JTAG signals from floating when they are not being used.

Table 6 defines the dedicated pins used by the mandatory JTAG signals for the PZ3320.

The JTAG specifications define two sets of commands to support boundary-scan testing: high-level commands and low-level commands. High-level commands are executed via board test software on an a user test station such as automated test equipment, a PC, or an engineering workstation (EWS). Each high-level command comprises a sequence of low level commands. These low-level commands are executed within the component under test, and therefore must be implemented as part of the TAP Controller design. The set of low-level boundary-scan commands implemented in the PZ3320 is defined in Table 7. By supporting this set of low-level commands, the PZ3320 allows execution of all high-level boundary-scan commands.

Table 5. JTAG Pin Description

PIN	NAME	DESCRIPTION
tck	Test Clock Output	Clock pin to shift the serial data and instructions in and out of the tdi and tdo pins, respectively. tck is also used to clock the TAP Controller state machine.
tms	Test Mode Select	Serial input pin selects the JTAG instruction mode. tms should be driven high during user mode operation.
tdi	Test Data Input	Serial input pin for instructions and test data. Data is shifted in on the rising edge of tck.
tdo	Test Data Output	Serial output pin for instructions and test data. Data is shifted out on the falling edge of tck. The signal is tri-stated if data is not being shifted out of the device.
trstn	Test Reset	Forces TAP controller to test logic reset state. This signal is active low.

Table 6. PZ3320 JTAG Pinout by Package Type

DEVICE	(PIN NUMBER / MACROCELL #)				
	tck	tms	tdi	tdo	trstn

Table 7. PZ3320 Low-Level JTAG Boundary-Scan Commands

INSTRUCTION (Instruction Code) *Register Used*	**DESCRIPTION**
SAMPLE/PRELOAD (00010) *Boundary-Scan Register*	The mandatory SAMPLE/PRELOAD instruction allows a snapshot of the normal operation of the component to be taken and examined. It also allows data values to be loaded onto the latched parallel outputs of the Boundary-Scan Shift-Register prior to selection of the other boundary-scan test instructions.
EXTEST (00000) *Boundary-Scan Register*	The mandatory EXTEST instruction allows testing of off-chip circuitry and board level interconnections. Data would typically be loaded onto the latched parallel outputs of Boundary-Scan Shift-Register using the SAMPLE/PRELOAD instruction prior to selection of the EXTEST instruction.
BYPASS (11111) *Bypass Register*	Places the 1 bit bypass register between the tdi and tdo pins, which allows the BST data to pass synchronously through the selected device to adjacent devices during normal device operation. The BYPASS instruction can be entered by holding tdi at a constant high value and completing an Instruction-Scan cycle.
IDCODE (00001) *Boundary-Scan Register*	Selects the IDCODE register and places it between tdi and tdo, allowing the IDCODE to be serially shifted out of tdo. The IDCODE instruction permits blind interrogation of the components assembled onto a printed circuit board. Thus, in circumstances where the component population may vary, it is possible to determine what components exist in a product.
HIGHZ (00101) *Bypass Register*	The HIGHZ instruction places the component in a state in which all of its system logic outputs are placed in an inactive drive state (e.g., high impedance). In this state, an in-circuit test system may drive signals onto the connections normally driven by a component output without incurring the risk of damage to the component. The HIGHZ instruction also forces the Bypass Register between tdi and tdo.

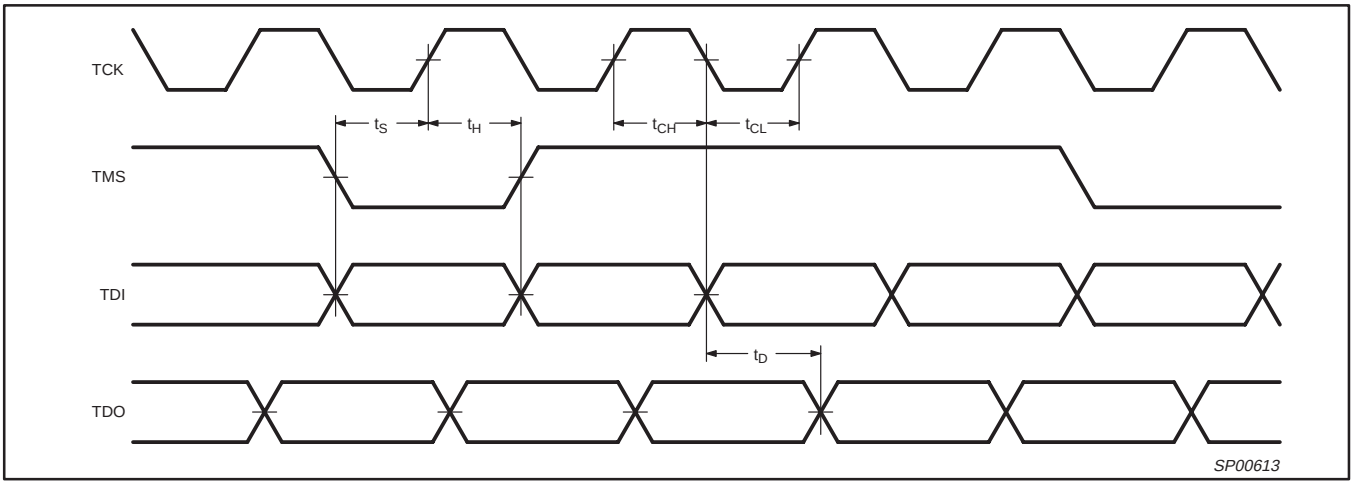


Figure 29. Boundary Scan Timing Diagram

Table 8. Boundary scan timing characteristics

SYMBOL	PARAMETER	MIN	MAX	UNIT
t_S	tdi/tms to tck setup time	20	–	ns
t_H	tdi/tms from tck hold time	0	–	ns
t_{CH}	tck high time	50	–	ns
t_{CL}	tck low time	50	–	ns
f_{TCK}	tck frequency	–	10	MHz
t_D	tck to tdo delay	–	20	ns

ABSOLUTE MAXIMUM RATINGS[1]

SYMBOL	PARAMETER	MIN	MAX	UNIT
V_{DD}	Supply voltage	–0.5	4.6	V
V_{IN}	Input voltage	–1.2		V
V_{OUT}	Output voltage	–0.5	V_{DD}+0.5	V
I_{IN}	Input current	–30	30	mA
T_J	Junction temperature range	–40	150	°C
T_{STG}	Storage temperature range	–65	150	°C

NOTE:
1. Stresses above these listed may cause malfunction or permanent damage to the device. This is a stress rating only. Functional operation at these or any other condition above those indicated in the operational and programming specification is not implied.

OPERATING RANGE

PRODUCT GRADE	TEMPERATURE	VOLTAGE
Commercial	0 to 70°C	3.3 ±10% V
Industrial	–40 to 85°C	3.3 ±10% V

DC ELECTRICAL CHARACTERISTICS FOR COMMERCIAL GRADE DEVICES

Commercial temperature range: V_{DD} = 3.0V to 3.6V; 0°C < T_{amb} < 70°C

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
V_{IH}	Input high voltage	V_{DD} = 3.6V	2.0	V_{DD}+0.3	V
V_{IL}	Input low voltage	V_{DD} = 3.0V	–0.3	0.8	V
V_{OH}	Output high voltage	V_{DD} = 3.0V; I_{OH} = –8mA	2.4	–	V
V_{OL}	Output low voltage	V_{DD} = 3.0V; I_{OH} = 8mA	–	0.4	V
I_I	Input leakage current	V_{DD} = 3.6V; 0 < V_{IN} < V_{DD}	–10	10	µA
I_{DDSB}	Standby current	T_{amb} = 25°C; no output loads, inputs at V_{DD} or V_{SS}.	–	100	µA
C_{IN}	Input capacitance	T_{amb} = 25°C; V_{DD} = 3.3V; f = 1MHz	–	10	pF
C_{IO}	I/O capacitance	T_{amb} = 25°C; V_{DD} = 3.3V; f = 1MHz	–	10	pF
C_{CLK}	Clock pin capacitance	T_{amb} = 25°C; V_{DD} = 3.3V; f = 1MHz	–	12	pF
R_{DONE}	done pull-up resistor	V_{DD} = 3.0 V; V_{IN} = 0 V	10	30	kΩ
R_{PD}	Unused I/O pull-down resistor	V_{DD} = 3.6V; V_{IN} = V_{DD}	100	400	kΩ

AC ELECTRICAL CHARACTERISTICS FOR COMMERCIAL GRADE DEVICES

Commercial temperature range: V_{DD} = 3.0V to 3.6V; 0°C < T_{amb} < 70°C

SYMBOL	PARAMETER	MIN	MAX	UNIT
Timing requirements				
t_{CL}	Clock LOW time	2.5		ns
t_{CH}	Clock HIGH time	2.5		ns
t_{SU_PAL}	PAL setup time (Global clock)	4.0		ns
t_{SU_PLA}	PLA setup time (Global clock)	5.5		ns
t_{SU_XOR}	XOR setup time (Global clock)	6.5		ns
t_H	Hold time (Global clock)		0	ns
Output characteristics				
t_{PD_PAL}	Input to output delay through PAL		7.5	ns
t_{PD_PLA}	Input to output delay through PLA		9.0	ns
t_{PD_XOR}	Input to output delay through XOR		10.0	ns
t_{PDF_PAL}	Input (or feedback node) to internal feedback node delay time through PAL		4.0	ns
t_{PDF_PLA}	Input (or feedback node) to internal feedback node delay time through PLA		5.5	ns
t_{PDF_XOR}	Input (or feedback node) to internal feedback node delay time through XOR		6.5	ns
t_{CF}	Global clock to feedback delay		2.5	ns
t_{CO}	Global clock to out delay		6.0	ns
t_{CS}	Clock skew (variance for switching outputs with common global clock)		1.0	ns
f_{MAX1}	Maximum flip-flop toggle rate $\left(\frac{1}{t_{CL} + t_{CH}}\right)$	200		MHz
f_{MAX2}	Maximum internal frequency $\left(\frac{1}{t_{SU\_PAL} + t_{CF}}\right)$	154		MHz
f_{MAX3}	Maximum external frequency $\left(\frac{1}{t_{SU\_PAL} + t_{CO}}\right)$	100		MHz
t_{BUFF}	Output buffer delay (fast)		3.5	ns
t_{SSR}	Slow slew rate incremental delay		8.0	ns
t_{EA}	Output enable delay		8.0	ns
t_{ER}	Output disable delay[1]		8.0	ns
t_{GTSH}	Global 3-State enable		40.0	ns
t_{GTSR}	Global 3-State disable		40.0	ns
t_{RR}	Input to register reset		10.5	ns
t_{RP}	Input to register preset		10.5	ns
t_{GRR}	Global reset to register reset		40	ns
t_{GZIA}	Global ZIA delay		4.0	ns

NOTE:

1. Output C_L = 5.0pF.

DC ELECTRICAL CHARACTERISTICS FOR INDUSTRIAL GRADE DEVICES

Industrial temperature range: V_{DD} = 3.0V to 3.6V; –40°C < T_{amb} < 85°C

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
V_{IH}	Input high voltage	V_{DD} = 3.6V	2.0	V_{DD}+0.3	V
V_{IL}	Input low voltage	V_{DD} = 3.0V	–0.3	0.8	V
V_{OH}	Output high voltage	V_{DD} = 3.0V; I_{OH} = –8mA	2.4	–	V
V_{OL}	Output low voltage	V_{DD} = 3.0V; I_{OH} = 8mA	–	0.4	V
I_I	Input leakage current	V_{DD} = 3.6V; 0 < V_{IN} < V_{DD}	–10	10	µA
I_{DDSB}	Standby current	T_{amb} = 25°C; no output loads, inputs at V_{DD} or V_{SS}.	–	100	µA
C_{IN}	Input capacitance	T_{amb} = 25°C; V_{DD} = 3.3V; f = 1MHz	–	10	pF
C_{IO}	I/O capacitance	T_{amb} = 25°C; V_{DD} = 3.3V; f = 1MHz	–	10	pF
C_{CLK}	Clock pin capacitance	T_{amb} = 25°C; V_{DD} = 3.3V; f = 1MHz	–	12	pF
R_{DONE}	done pull-up resistor	V_{DD} = 3.0 V; V_{IN} = 0 V	10	30	kΩ
R_{PD}	Unused I/O pull-down resistor	V_{DD} = 3.6V; V_{IN} = V_{DD}	100	400	kΩ

AC ELECTRICAL CHARACTERISTICS FOR INDUSTRIAL GRADE DEVICES

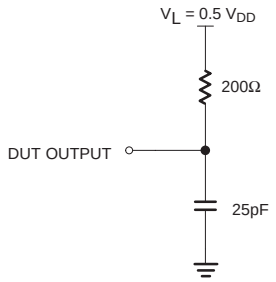
Industrial temperature range: V_{DD} = 3.0V to 3.6V; –40°C < T_{amb} < 85°C

SYMBOL	PARAMETER	MIN	MAX	UNIT
Timing requirements				
t_{CL}	Clock LOW time	2.5		ns
t_{CH}	Clock HIGH time	2.5		ns
t_{SU_PAL}	PAL setup time (Global clock)	4.5		ns
t_{SU_PLA}	PLA setup time (Global clock)	6.0		ns
t_{SU_XOR}	XOR setup time (Global clock)	7.0		ns
t_H	Hold time (Global clock)		0	ns
Output characteristics				
t_{PD_PAL}	Input to output delay through PAL		8.0	ns
t_{PD_PLA}	Input to output delay through PLA		9.5	ns
t_{PD_XOR}	Input to output delay through XOR		10.5	ns
t_{PDF_PAL}	Input (or feedback node) to internal feedback node delay time through PAL		4.0	ns
t_{PDF_PLA}	Input (or feedback node) to internal feedback node delay time through PLA		5.5	ns
t_{PDF_XOR}	Input (or feedback node) to internal feedback node delay time through XOR		6.5	ns
t_{CF}	Global clock to feedback delay		2.5	ns
t_{CO}	Global clock to out delay		6.5	ns
t_{CS}	Clock skew (variance for switching outputs with common global clock)		1.0	ns
f_{MAX1}	Maximum flip-flop toggle rate $\left(\frac{1}{t_{CL} + t_{CH}}\right)$	200		MHz
f_{MAX2}	Maximum internal frequency $\left(\frac{1}{t_{SU\_PAL} + t_{CF}}\right)$	143		MHz
f_{MAX3}	Maximum external frequency $\left(\frac{1}{t_{SU\_PAL} + t_{CO}}\right)$	91		MHz
t_{BUFF}	Output buffer delay (fast)		4.0	ns
t_{SSR}	Slow slew rate incremental delay		8.0	ns
t_{EA}	Output enable delay		8.5	ns
t_{ER}	Output disable delay[1]		8.5	ns
t_{GTSH}	Global 3-State enable		40.0	ns
t_{GTSR}	Global 3-State disable		40.0	ns
t_{RR}	Input to register reset		11.0	ns
t_{RP}	Input to register preset		11.0	ns
t_{GRR}	Global reset to register reset		40	ns
t_{GZIA}	Global ZIA delay		4.5	ns

NOTE:

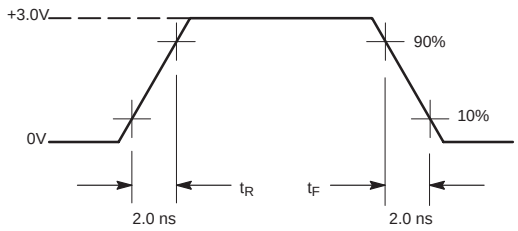
1. Output C_L = 5.0pF.

THEVENIN EQUIVALENT



SP00629

VOLTAGE WAVEFORM



SP00630

MEASUREMENTS:
All circuit delays are measured at the +1.5V level of inputs and outputs, unless otherwise specified.

Input Pulses

Table 9. Pin Description

SYMBOL	TYPE	DESCRIPTION
V_{DD}	–	Positive power supply.
GND	–	Ground supply.
resetn	I	During configuration, resetn forces the start of initialization (see Figure 8). After configuration, resetn is a direct input which can be used to asynchronously reset all the flip-flops. If the global reset is not being used, this pin should be pulled high.
cclk	I/O	In the master modes, cclk is an output which strobes configuration data in. In the slave or synchronous peripheral mode, cclk is an input synchronous with the data on din or D[7:0]. After configuration, this pin should be pulled low.
done	I/O	done is a bi-directional signal with a weak pull-up resistor attached. As an output, done pulling high indicates configuration is complete. As an input, a low level on done will delay device initialization and the enabling of user I/O. If only one device is used, this pin can be left floating. If multiple devices are daisy chained, an external pull-up should be used (see Figure 28).
pgmn	I	pgmn is an active-low input that forces the restart of configuration and initialization (see Figure 8) and resets the boundary-scan circuitry. After configuration, the pin should be pulled high.
rdy_busyn	O	During configuration in peripheral mode, rdy_busyn indicates another byte can be written to the PZ3320. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
rclk	O	During the master parallel configuration mode, rclk is an output signal to an external memory. rclk is not normally used. After configuration, this pin is a user-programmable I/O pin, and no external termination is required. See the section on terminations for more information.
din	I	During slave serial or master serial configuration modes, din accepts serial configuration data synchronous with cclk. During parallel configuration modes, din is the D[0] input. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
M2 M0 M1	I	M2/M1/M0 are used to select the configuration mode as defined in Table 3. After configuration, the pins are user-programmable I/O, and no external termination is required. See the section on terminations for more information.
M3	I	M3 is used to select the frequency of the internal oscillator during configuration. When M3 is low, the oscillator is nominally 10MHz. When M3 is high, the oscillator is nominally 1.25MHz. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
tdi tdo tck tms trstn	I O I I I	Test Data In, Test Data Out, Test Clock, Test Mode Select, Test Reset are dedicated pins for boundary-scan through the JTAG port. If JTAG is not being used, tdi, tck, tms, and trstn should be terminated with a weak pull-up resistor. tdo can be left unterminated. See section on terminations for more information.
hdc	O	High During Configuration (hdc) is output high when the PZ3320 is in the configuration state. hdc is used as a control output indicating that configuration is in progress. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
ldcn	O	Low During Configuration (ldcn) is output low when the PZ3320 is in the configuration state. ldcn is used as a control output indicating that configuration is in progress. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
initn	I/O	initn is an active-low bi-directional pin that holds the PZ3320 in a wait state before the start of configuration. During configuration, an internal pull-up is enabled. If only one device is used, this pin can be left floating. If multiple devices are daisy chained, an external pull-up should be used (see Figure 28). After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
gts	I	Global 3-State is an active-high dedicated input used to 3-state the I/Os. If this feature is not used, the pin should be pulled low.
cs0n cs1	I	cs0n/cs1 are used in the peripheral configuration mode. The PZ3320 is selected when cs0n is low and cs1 is high. After configuration, these pins are user-programmable I/O, and no external termination is required. See the section on terminations for more information.
A[19:0]	O	In the master parallel configuration mode, A[19:0] address the configuration EEPROM. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
D[7:0]	I	During master parallel, peripheral, and slave parallel configuration modes, D[7:0] receive configuration data. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.
dout	O	During configuration, dout is the serial data out that is used to drive the din of daisy-chained slave devices. Data on dout changes on the falling edge of cclk. After configuration, the pin is a user-programmable I/O, and no external termination is required. See the section on terminations for more information.

Data sheet status

Data sheet status	Product status	Definition [1]
Objective specification	Development	This data sheet contains the design target or goal specifications for product development. Specification may change in any manner without notice.
Preliminary specification	Qualification	This data sheet contains preliminary data, and supplementary data will be published at a later date. Philips Semiconductors reserves the right to make chages at any time without notice in order to improve design and supply the best possible product.
Product specification	Production	This data sheet contains final specifications. Philips Semiconductors reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.

[1] Please consult the most recently issued datasheet before initiating or completing a design.

Definitions

Short-form specification — The data in a short-form specification is extracted from a full data sheet with the same type number and title. For detailed information see the relevant data sheet or data handbook.

Limiting values definition — Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.

Application information — Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Disclaimers

Life support — These products are not designed for use in life support appliances, devices or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

Right to make changes — Philips Semiconductors reserves the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

Philips Semiconductors
811 East Arques Avenue
P.O. Box 3409
Sunnyvale, California 94088–3409
Telephone 800-234-7381

print code Date of release: 07-98

Document order number: 9397 750 04178