MPA1000 Product Description

Motorola Programmable Array (MPA) products are a high density, high performance, low cost, solution for your reconfigurable logic needs. When used with our automatic high performance design tools, MPA delivers custom logic solutions in minutes rather than weeks. And the low cost keeps those solutions competitive throughout the product lifecycle.

The MPA architecture has solved the historical problems associated with fine grain devices without sacrificing re–programmability, reliability, or cost. MPA1000 devices are reprogrammable SRAM based products manufactured on a standard 0.5µ Leff CMOS process with logic capacities from 3,500 to more than 22,000 equivalent FPGA gates. MPA Logic resources hold a single gate or storage element providing a highly efficient, adaptable, design implementation medium. Gate level logic resources, abundant hierarchical interconnection resources and automatic, timing driven, tools work together to quickly provide design implementations that meet timing constraints without sacrificing device utilization.

Staying focused on end product design rather than implementation tools or device architecture gets the design done faster and, unlike other programmable solutions, without programmable logic device specificity to impede future design migration efforts. The combination of automatic tools and gate level architecture is ideal for traditional schematic driven or high level language based design methodologies. In fact, logic synthesis tools were originally designed for and produce the most efficient results when targeting gate level devices.

High MPA1000 register count and controlled clock skew is ideal for designs employing pipelining techniques such as communications. The unique set of MPA1000 I/O programming options make these devices suitable for industrial and computer Interfacing circuits.

Features

- Multiple I/O from 80–200 I/O Pins
- Programmable 3V/5V I/O at Any Site
- Multiple Packaging Options
- Fine Grain Structure Is Optimized for Logic Synthesis
- Programmable Output Drive, 6/12mA @ 5.0V
- High Register Count, with 560–2,900 Flip–Flops
- IEEE 1149.1 JTAG Boundary Scan
- Eight Low–Skew (<1ns) Clocks

Table 2–1. MPA1000 Family Members

FPGA Gates	Part No.	Logic Cells	Internal Flip–Flops	I/O Cell Flip–Flops	Signal I/O Pads Max.	Packages	Availability
3500	MPA1016FN MPA1016DD	1600	400	160	80	84–Pin PLCC 128–Pin PQFP	April 1996 April 1996
8000	MPA1036FN MPA1036DD MPA1036DH MPA1036HI	3600	900	240	120	84–Pin PLCC 128–Pin PQFP 160–Pin PQFP 181–Pin PGA	NOW April 1996 NOW NOW
14200	MPA1064DH MPA1064DK MPA1064KE	6400	1600	320	160	160–Pin PQFP 208–Pin PQFP 224–Pin PGA	April 1996 2Q96 1Q96
22000	MPA1100DK MPA1100HV	10000	2500	400	200	208–Pin PQFP 299–Pin PGA	3Q96 3Q96

MPA17000 Serial EPROM Family

Capacity	MPA1000 Companion	Part No.	Packages	Availability
64K	MPA1016	MPA1765P MPA1765D MPA1765FN	8–Pin DIP 8–Pin SOIC 20–Pin PLCC	NOW
128K	MPA1036	MPA17128P MPA17128D MPA17128FN	8–Pin DIP 8–Pin SOIC 20–Pin PLCC	NOW
256K	MPA1064	MPA17256FN	20–Pin PLCC	1996

MPA1000 Capacity

Programmable logic gate capacity is difficult to ascertain because it is design and design tool dependent. Programmable logic capacities can only be meaningfully compared using identical designs and automatic tools. Figure 2–1 shows that under these circumstances, the MPA1036 contains from 2.1 to 1.3 Xc3190 devices.

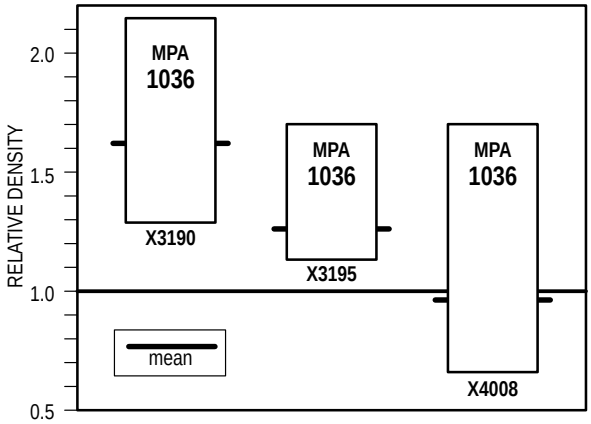


Figure 2–1. Equivalent Gate Capacity

Table 2–1 on page 2–7 shows the members of the MPA1000 family and lists the I/O, logic cell, flip flop and gate capacities for each device. To facilitate Customer device selection, Motorola rates MPA device capacity in FPGA equivalent gates. The equivalent gate counts shown were derived using identical designs and a push button implementation methodology. While this method is useful in a comparative sense, actual device capacity remains a design dependent quantity. Designs with high register gate or XOR gate to total gate ratio will pack more efficiently than the averages shown in Figure 2–1.

MPA1000 Performance

Device performance is more design and design tool dependent than device capacity. Table 2–2 shows selected cell performance figures for a typical MPA1000 device. Calculating MPA1000 DFF toggle rate from this information yields an unrealistically high expectation for device performance. Some manufacturers publish specifications for small functional blocks like counters. While more useful than toggle rates, they are based on ideal placement and routing conditions seldom achievable without manual intervention. Industry benchmarks are useful for relative comparisons of benchmark design performance, but benchmark designs don't end up in products. In addition, the design methodology used requires, manual, architecture dependent, design optimization and expert level architectural and design tool expertise. Using this design methodology for real designs means a costly learning curve, severe technology migration limitations and many hours of extra design effort for each end product. If the incentive to use a programmable solution is time to market and product flexibility, this is not the ideal approach. A push button, gate level, approach increases design flexibility and improves time to market. The MPA1000 and MPA design system have been engineered to deliver a high performance gate level solution. Gate level design is widely understood, technology independent and synthesis friendly. A library of common MSI functions with optimized gate level representations are provided to reduce design implementation time.

Table 2–2. Selected MPA1000 Performance Figures

	Typical
MEDIUM BUS DELAY	1.2ns
DFF CLK TO Q	1.2ns
DFF SETUP TIME	1.5ns
TYPICAL DFF TOGGLE RATE	256MHz

(25° C, V_{CC} = 5V)

If identical designs and timing constraints are used with automatic, timing driven, design tools, a more appropriate performance comparison can be made. Figure 2–2 compares the MPA1036 vs. the XC4008 for 7 designs. The typical MPA1036 device is 48% faster than the XC4008–6 and 28% faster than the XC4008–4 for 7 identical, complex, chip level designs. In real design situations, gate level flexibility and hierarchical routing coupled with sophisticated, timing driven, design tools results in significant performance gains and reduced time to market.

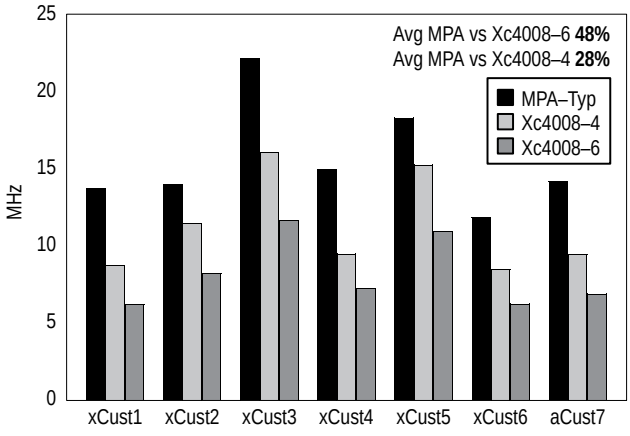


Figure 2–2. MPA1036 versus XC4008 – 7 Push Button Designs

If step and repeat style designs typical of industry benchmarks are used (Figure 2–3), MPA retains it's performance edge. While the performance gap shrinks by about 10%, absolute design performance increases dramatically compared to those shown in Figure 2–2. As critical path depth decreases, design performance increases as expected. In general these benchmarks tend to have narrowly distributed performance constraints and shallow path depths atypical of many real design implementations. In either case using benchmark information to estimate product performance for arbitrary designs is unlikely to yield reliable results. This information is intended to illustrate the range of performance enhancement possible when MPA is selected.

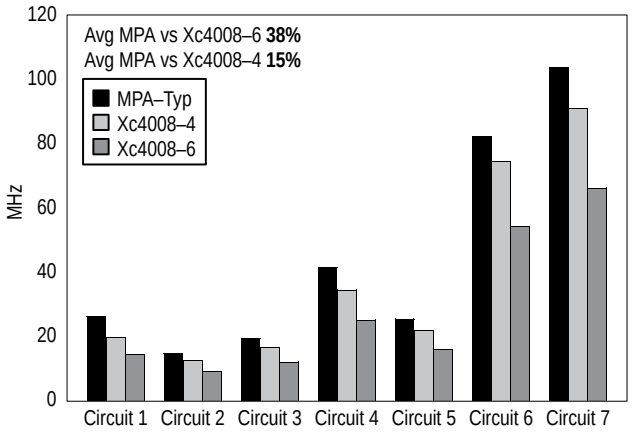


Figure 2–3. MPA1036 versus XC4008 – 7 Push Button, Step & Repeat Designs

2

MPA1000 Architectural Overview

MPA1000 Architecture

MPA1000 is a high density, high performance, low cost device family which maximizes application flexibility and minimizes time to market by delivering a gate level reprogrammable logic solution. Combined with automatic high performance design tools, the MPA1000 family is ideally suited to logic synthesis or gate level (gate array like) design methods.

Logic resources in the MPA1000 are fine grained – each logic cell holds a single gate or a storage element. This provides a highly efficient, adaptable, design implementation medium. Gate level logic resources, abundant hierarchical interconnection resources and automatic, timing driven, tools work together to quickly provide design implementations that meet timing constraints without sacrificing device utilization.

The MPA1000 architecture has solved the historical problems associated with fine grain architectures without sacrificing re–programmability, reliability, or cost. Previous reprogrammable fine grain architectures utilized routing architectures substantially similar to that of coarse grained products. Other fine grained architectures resorted to antifuse programming elements to address performance issues, increasing cost, while reducing reliability and abandoning reconfigurability. MPA utilizes a new routing structure which takes advantage of fine logic block granularity to achieve superior design performance.

MPA1000 devices are manufactured using a standard submicron CMOS process. SRAM cells comprise device configuration memory. MPA1000 devices can be quickly and infinitely reprogrammed.

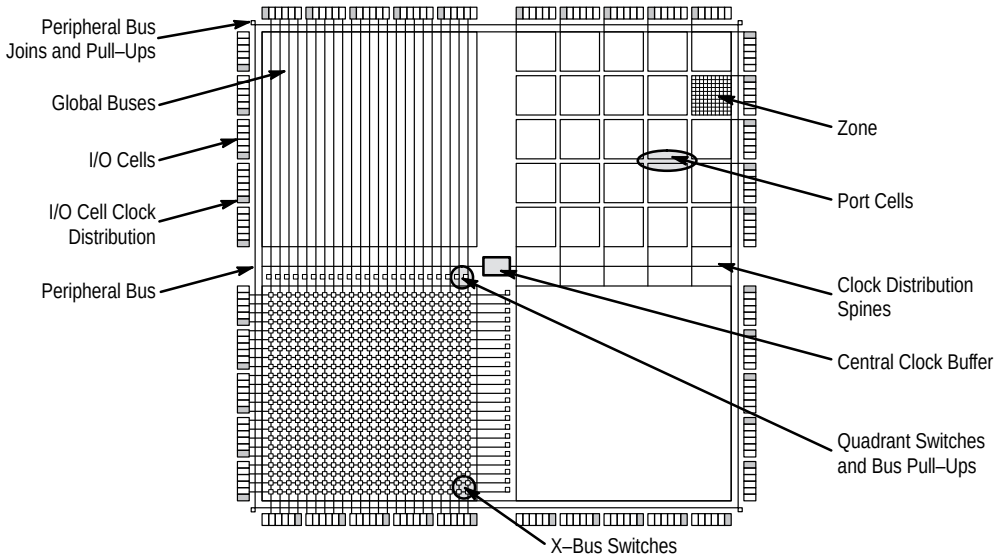


Figure 2–4. MPA Architectural Overview

Partitioned Resources

Each device is a multilevel partitioned array of cells. At the highest level of hierarchy each device is partitioned into 4 equal sized sections called quadrants. I/O cells surround the quadrants. Each quadrant is further subdivided into zones. A zone consists of a 10x10 array of core cells, 20 port cells and a clock distribution cell (Figure 2–6). Zone core cells are organized into 2x2 groups called tiles. The number of zones per quadrant defines a particular device as shown in Figure 2–5. Partitioning the device in this manner minimizes bus loading and provides an opportunity to segment device level placement and routing. This speeds design implementation time, especially if multiple processors are used. Figure 2–4 is a synopsis of the overall MPA structure.

2

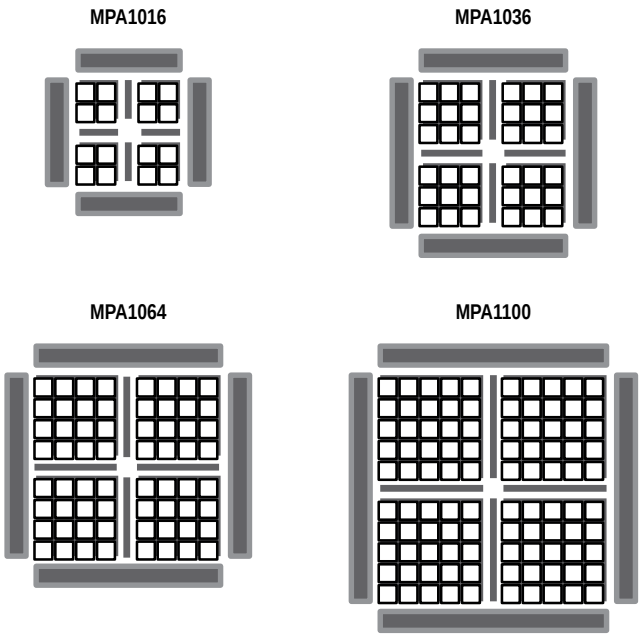


Figure 2–5. The MPA 1016, MPA1036, MPA1064 and MPA1100

Hierarchical functionality complements the robust routing resource to deliver extremely efficient design realizations. While the look up table approach of non–gate level devices can provide any function of its inputs, this flexibility is costly when simple functions are required. In contrast the simplicity, small size, and hierarchical organization of the MPA1000 delivers a more silicon efficient implementation. Logic blocks of arbitrary size and aspect ratio are automatically constructed, optimized and interconnected based on design constraints and gate level design representations. This capability complements logic synthesis technology and maximizes design migration potential. As FPGA device capacity increases, design diversity will also increase. The malleable granularity and adjustable routing resource of the MPA can accommodate this diversity with consistent silicon efficiency and performance.

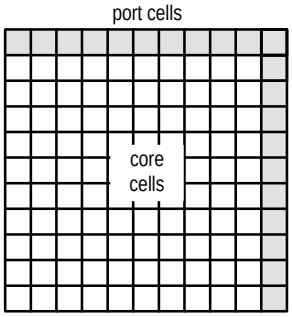


Figure 2–6. Zone Structure

Core Cells

Each core cell has 2 inputs, each input is configured to receive signals from 1 of 7 potential sources (Figure 2–7). 5 sources are from local interconnect and 2 are from zone level interconnect. Each cell output connects to 8 other cells via local interconnect and is configured to connect to up to 4 medium buses. Cells are sometimes used to provide additional routing resource. The ability to use a core cell as a routing resource or as logic provides a programmable means of adjusting routing resource to fit design specific requirements.

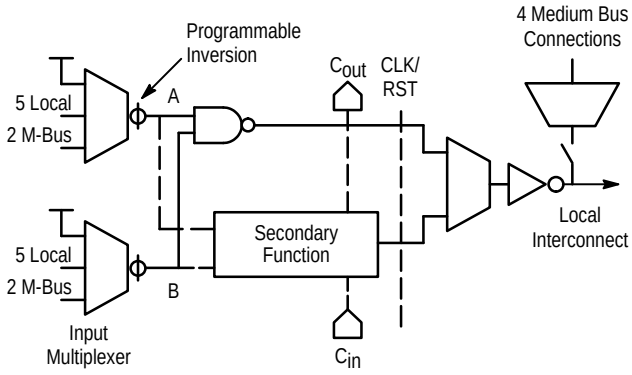


Figure 2–7. Core Cell Structure

Each cell has three states; repowering buffer, primary function, and secondary function. In addition, all cell inputs have programmable input inversion. MPA1000 core cells are organized in 2x2 groups called tiles. Within a tile, each of the 4 cells has a different secondary function (Figure 2–8). The core cell primary function is a 2 input NAND. Secondary functions include; XOR, register, and wired OR. The register element are configured as a DFF or latch with clock enable and set or reset. There is also special circuitry which used to group the blocks together to form compact multi–cell functions like; 1 bit full adder, 2 input multiplexer, JKFF. A special, 1ns skew, network is provided to drive register clock and reset/set pins. High performance, gate level, cells necessitate controlled clock skew to avoid negative setup time situations. The MPA cell states were chosen based on a careful analysis of macrocell utilization statistics from a large number of ASIC designs implemented in Motorola's H4C array.

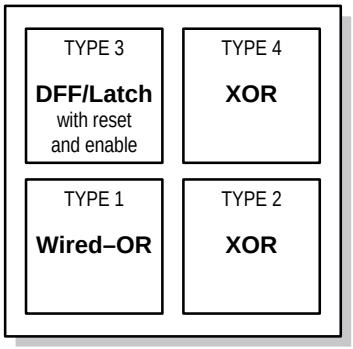


Figure 2–8. Core Cell Secondary Function

I/O cells

I/O cells are located at the device periphery surrounding the quadrants (Figure 2–4 on page 2–9). Besides direct input and output, each I/O cell is configured to be; input, output, bidirectional, registered input, registered output, registered I/O. The two registers can be independently configured as a latch or D–type flip flop. Input register setup time is adjustable to compensate for clock network input delay. Input buffer threshold adjustment provides either TTL or CMOS levels. Output buffer drive capability is programmable to 6ma or 12ma. And each output can be independently programmed to either 3V or 5V levels with slew rate control. True 3V outputs are useful for system level designs containing any 3V devices which are tolerant to 5V inputs. The output buffer can be configured as an open drain to facilitate system level wired OR applications. Figure 2–9 sums up I/O cell structure. Dedicated, fully IEEE 1149.1 compliant boundary scan is also provided.

Hierarchical Routing Resources

The MPA interconnection structure is partitioned into 3 levels; Global, Zonal (or medium), and Local. Local interconnection is used to connect a core cell to 8 of it's perpendicular neighbors (Figure 2–10). Zonal interconnect consists of the medium buses and connects groups of cells within a zone (Figure 2–11). Global interconnect includes global buses, x buses and interquadrant switches (Figure 2–4 on page 2–9). Global buses provide quadrant and chip level inter–zone and zone to I/O cell interconnections. Special interconnection resources are also present and consist of clock distribution, wired OR and peripheral bus. Routing specialization provides an opportunity for level specific performance optimization. Specialization also diminishes the amount of interconnection options required at each core cell, reducing cell size and boosting silicon efficiency.

Local Interconnect

Local interconnect provides the fastest path between 8 neighboring core cells. Local interconnect is continuous across the device and is not effected by zonal boundaries. Local interconnection favors frequently used connections, the cell to the immediate left and immediate right of the driving cell have 2 connections. Local connections are used for high performance intrazone connections and are also used to cross zone boundaries when necessary.

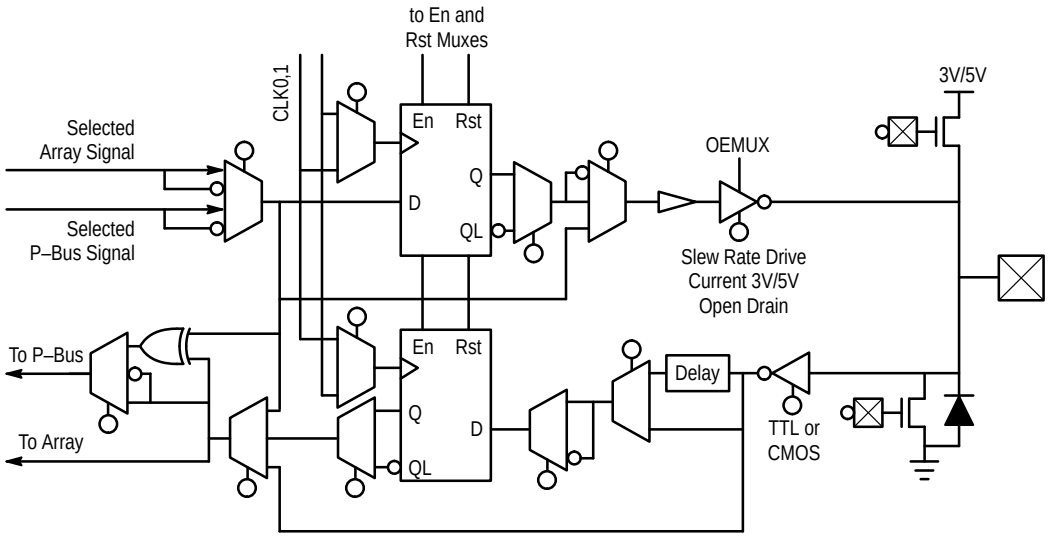


Figure 2–9. Input/Output Cell Structure

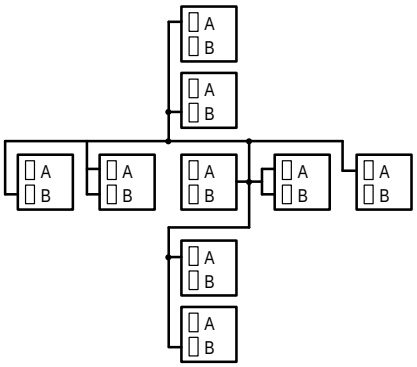


Figure 2–10. Local Interconnect

2

Medium Interconnect

Medium interconnect spans a single zone and provides intrazone connections beyond the span of local interconnect or for connection of zone cells to global signals through the port cells. There are 4 horizontal and 4 vertical medium buses per core cell. Medium bus connectivity to core cells is sparse to minimize loading and limit core cell input multiplexer size. This connectivity is arranged so that a tile can be fully connected to the 16 medium buses which cross it.

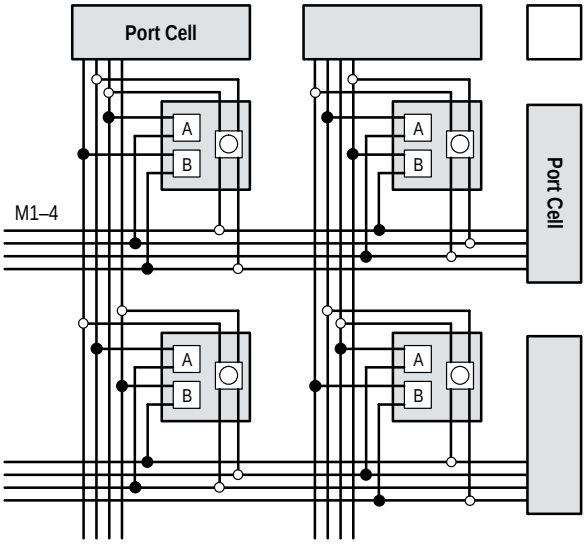


Figure 2–11. Medium Interconnect

Port Cells

At zone edges, port cells provide a bridge between global resources and zonal resources. Port cells transport signals into and out of a zone and are the only interface between zonal and global resources (Figure 2–12). All 4 medium buses, 4 global buses and the x bus in a given row or column connect to the port cell.

Port cells also provide connections to 4 of the 8 low skew clock distribution lines which span the device. Port cells also provide global to x bus access and serve as a pathway for zonal wired OR buses to connect to global busses.

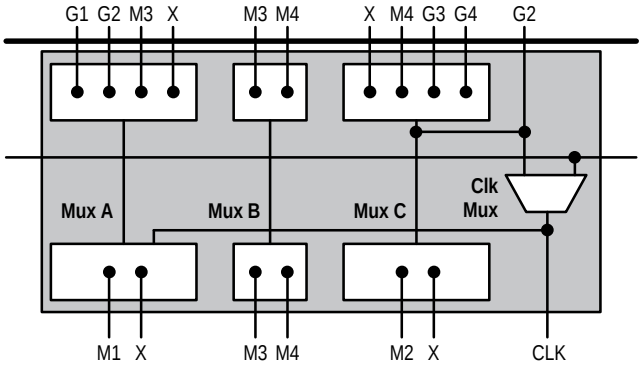


Figure 2–12. Vertical Port Cells

Global Interconnect

Global interconnect consists of global buses, x buses, interquadrant switches. There are 4 horizontal and 4 vertical global buses passing over each core cell. All Global buses only connect to the port cells, I/O cells and interquadrant switches. Global buses span a quadrant and are used to interconnect the zones within the quadrant together. Between quadrants, interquadrant switches connect two global buses together to form a device level connection.

Each core cell contains a x–bus switch (Figure 2–13) which is independent of cell logic or interconnect functions. A single vertical and a single horizontal x bus passes over each core cell and connects to this switch. Each x bus connects to all the core cells is a single zone column or row and terminates at the port cells on opposite edges of the zone. Each x bus has 10 connections inside the zone and 2 port cell connections. Port cell connections are used to make x to global, x to x and medium to x connections. Medium to x connections are used to hop over a single zone X buses are used to facilitate 90° global bus turns and provide a means for global bus fanout.

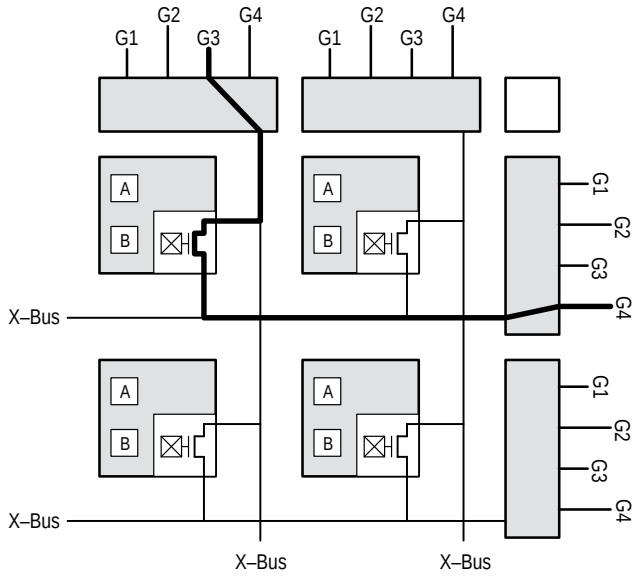


Figure 2–13. Global Bus Turn Using the X–Bus

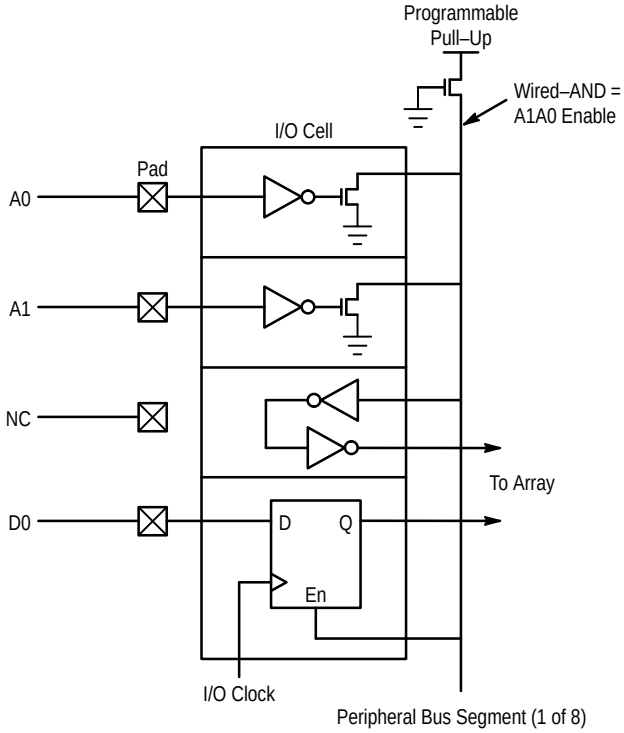


Figure 2–14. Using the Peripheral Bus for Address Decoding

2

I/O Cell Connections and Peripheral Bus

I/O cells are a pathway between array and bonding pads. Global buses, x buses and adjacent zone medium buses can be connected to I/O cells at quadrant edges. Each I/O cells is directly connected to the adjacent bonding pad.

A specialized bus, called the peripheral bus, resides in the I/O cell – quadrant interface (Figure 2–4 on page 2–9). The peripheral bus comprises 8 lines which are interrupted at device corners by a peripheral bus switch similar to the interquadrant switch. This switch joins peripheral bus segments to create connections spanning more than a single device edge. Peripheral buses carry I/O control signals common to two or more I/O cells such as a latch enable or tristate control signal. The I/O cells can also drive these buses with an open drain device. When combined with programmable pullups located in the corners of the device, the peripheral bus can be used to form wide gates for address decoding (Figure 2–14).

Wired OR Nets

Wired OR nets are constructed using type 1 core cells. When the type 1 secondary function is enabled, the NAND drives an open drain device directly connected to a special bus shared by all the type 1 cells in the same zone row. This bus, the zone wired OR bus, terminates in the port cell and has a single, dedicated, pullup. When this bus is used, the port cell wired OR to global bus connection and the global bus pullup located near the interquadrant switch are enabled. These resources are used to map 3–state buses onto the MPA1000 device.

Clock Distribution

Clock distribution is implemented through a dedicated, low skew, network consisting of; 8 dedicated clock input lines connected to 2 I/O cells on each device edge, a central clock buffer, a distribution comb structure, zone corner clock selection cells and the zone port cells along the top of each zone. The zone corner cell selects 2 of the 8 lines for zone clocks and 2 of the 8 lines for zone reset (Figure 2–15). Zone registers are connected to these clock and reset signals through the top row of port cells. The comb extends into the I/O cells via a similar clock selection cell attached to each group of 5 I/O cells. This group is called an I/O zone. All 8 clock lines can be driven from the I/O bonding pad or the array. The distribution network is balanced and has a skew of < 1ns between any two register clock inputs.

Connections in the port cells allow the clock network to drive zone logic in addition to the register clock and reset pins. Unused clock lines can be used for efficient distribution of any high fanout signal. If there are more clocks in the design than clock resources, the MPA design system automatically constructs a comb from global buses to generate a secondary clock network with a skew of < 3ns. Secondary clock construction is facilitated by a port cell connection which provides non–clock network access to zone register clock and reset pins.

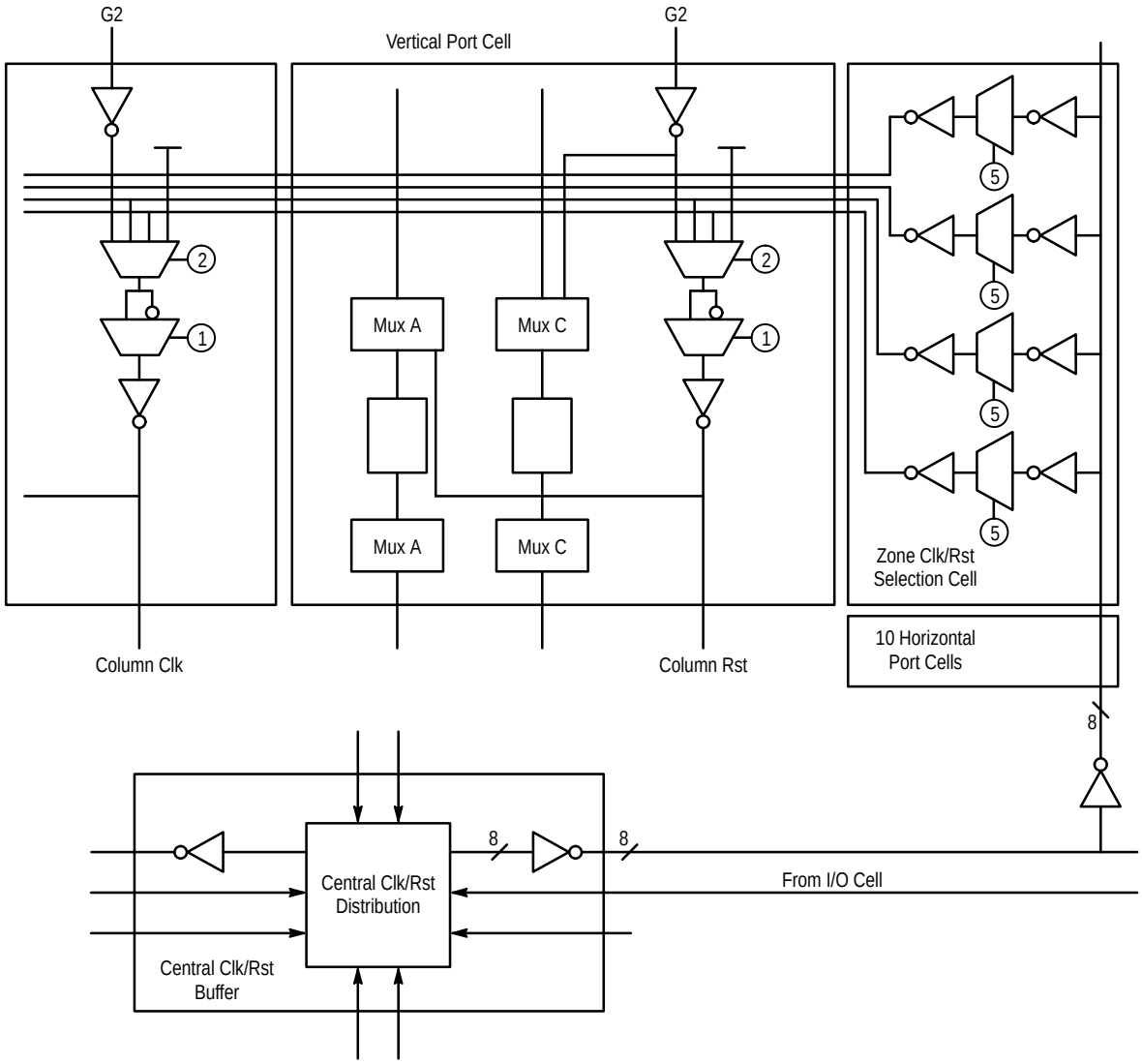


Figure 2–15. Clock Distribution Network Connectivity

MPA1000 Device Configuration

Configuration Overview

MPA1000 devices have an SRAM configuration memory. Configuration memory contents completely define MPA device function. The MPA1000 design system generates configurations from completed layouts. On chip control logic loads configurations in one of two modes automatically on power up or under external control. MPA1000 devices have a very rapid configuration load cycle, infinite reload and are in system reconfigurable. The configuration modes are; Boot From ROM (BFR) and microprocessor peripheral or MICRO Mode. In either mode, multiple devices can be daisy chained to form a large programmable subsystem.

In all BFR modes, the MPA device controls configuration and loads from either a byte wide or serial memory. In BFR mode 1 (Figure 2–17), the device generates 18 bits of address and reads 8 bits of configuration data. MPA devices generate 18 bits of address or 262K bytes (e.g., 17 MPA1036 devices). If a larger address range is required, BFR mode 3 (Figure 2–21 on page 2–20) can be used. In BFR mode 3 an external address generator is used to extend the address space. BFR mode 2 is a special case of BFR 3. In this case the address generator is resident in a serial PROM and data is presented to the device 1 bit at a time. The MPA design system download POD and the MPA17000 serial EPROMs are used with this mode.

In MICRO Mode, the MPA1000 device becomes an 8 bit peripheral slave device. A microcontroller or microprocessor controls the configuration process. MICRO Mode provides more control over configuration and user mode device behavior than other modes. For example MICRO Mode can be used to randomly access and load individual rows of device configuration memory.

Configuration information generated by the MPA Design System includes Error Check Bytes (ECBs). ECBs are used to detect configuration data corruption while configurations are loaded into the device. The configuration process halts and error status is indicated if an ECB mismatch is detected anytime during the configuration process. ECB checks insure the integrity of configuration data and protect MPA devices from damage.

Depending on the selected configuration mode, some user I/O pins become unavailable for post configuration use. These pins are listed as “dedicated” in Table 2–5 on page 2–24 and Table 2–4 on page 2–17. The system level interface for MPA configuration is shown in Figure 2–16. Note that the meaning of the F[4:0] pins is mode specific, refer to Table 2–5 and Table 2–4 for detailed signal descriptions.

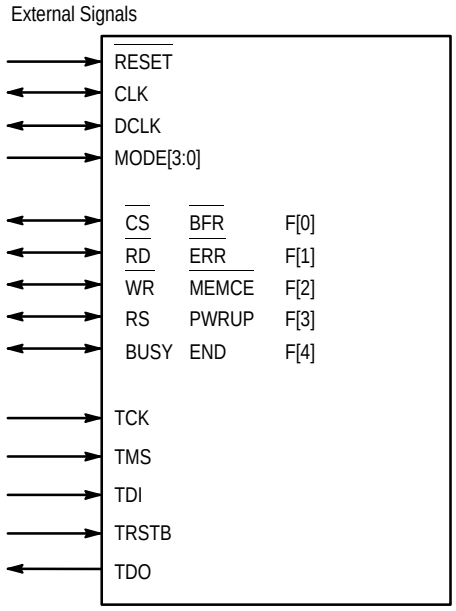


Figure 2–16. Configuration Interface Signals

Table 2–3. MODE[3:0] Pin Programming

Mode Bits		
[1]	[0]	Description
0	0	**MICRO Mode** — Micro–processor/controller interface circuitry with parallel (byte wide) data.
0	1	**BFR Mode (1)** — Boot From ROM, byte wide data. MPA generates ROM addresses.
1	0	**BFR Mode (2)** — Boot From ROM, serial data. (Low pin count serial EPROM generates own addresses.)
1	1	**BFR Mode (3)** — Boot From ROM, byte wide data. MPA does not generate ROM addresses.
Mode Bits		
[3]	[2]	Description
X	1	Use external clock for configuration.
1	X	Enable JTAG circuitry and pins.

Configuration Clock

The MPA1000 device has an internal oscillator. The internal configuration clock is derived from the oscillator and is presented at the CLK pin when MODE[2] is low. When MODE[2] is high, the internal clock is disconnected from the oscillator and an external clock must be presented on the CLK pin to configure the device.

Bootstrap Voltage

Signal pathways in the MPA 1000 device are controlled with n–channel transistors. The gates of these transistors are connected to individual SRAM configuration memory cells. To pass a rail to rail signal through these transistors during user operation, the gate voltage must be elevated above V_{DD} to compensate for transistor threshold and body effect voltage drops. MPA1000 devices contain a charge pump to generate this elevated voltage, called the bootstrap voltage. The charge pump is connected to the supply line of each SRAM cell and is driven by the internal oscillator.

Since configuration memory is generally not dynamically changing during user operation, the charge pump must only supply small leakage current losses and is not designed to supply sufficient current for SRAM read/write operations. During configuration, the charge pump (bootstrap) is internally disabled by shunting the SRAM supply to V_{DD} through a large p–channel device. In order for the charge pump to operate properly, the internal oscillator as well as the bootstrap circuitry must be enabled. In MICRO Mode, the processor has control over these functions. In BFR modes, the on chip configuration controller insures proper sequencing of these controls.

The MPA1000 device is only guaranteed to function properly with bootstrap enabled. The internal oscillator must be running and bootstrap should be activated 100µs before user inputs or outputs are enabled. If dynamic configuration modification is desired, the bootstrap voltage can be supplied externally on the V_{PP} pin and MICRO Mode can be used to disable bootstrap and on board oscillator. The bootstrap voltage should be V_{DD} + 1.5V. At no time should V_{PP} exceed 6.5V.

JTAG

The MPA1000 device contains dedicated JTAG IEEE 1149.1 boundary scan circuitry. JTAG can be used on unconfigured or fully configured devices. JTAG is enabled any time MODE[3] is raised. When MODE[3] is high, 5 user I/O pins become JTAG controls and user mode operation of those pins is interrupted. When a valid JTAG command sequence is issued, the JTAG TAP controller can take control of all device pad circuits regardless of device configuration state. Board level testing can be accomplished without first configuring the device. Since the TAP controller can take control of all device pins, care must be used to prevent the TAP controller from interfering with device user mode or configuration operation.

Boot From ROM (BFR) Modes

In BFR modes, the MPA device controls device configuration and assumes a memory–processor interface to the configuration store. The MPA device either asserts addresses directly (internal address generation) or issues address reset and increment pulses (external address generation). Data is read either serially or 8 bits at a time. Table 2–4 describes BFR interface signal operation. ADD[17:0] are only used in BFR mode 1. DATA[7:1] are not used in BFR mode 2 (serial data).

A BFR load sequence is initiated by a falling edge of $\overline{\text{RESET}}$, device power up or a rising edge of RESET. MEMCE falls to indicate the start of a configuration load sequence. On subsequent alternate rising edges of CLK, the data bus value is latched. The configuration process terminates when a complete configuration is successfully loaded and END is asserted or when a configuration error is detected and ERR is asserted. After END is asserted, the device will begin user mode operation 2 clocks after PWRUP is asserted or 2 clocks after END if PWRUP was already high. All configuration timing is synchronous with the internal or externally supplied configuration clock. Figure 2–19 describes BFR sequence timing details.

All BFR sequences begin with an internal device reset sequence where the entire configuration memory is reset. The duration of this sequence depends on the size of the MPA device being configured. A falling MEMCE edge indicates configuration commencement and data loads begin after 2 subsequent configuration clocks. The first positive edge of DCLK signals the external address generator to increment the byte or bit address. Prior to MEMCE assertion, DCLK is tristated.

The duration of the configuration process is also dependent on device size. Configuration duration can be estimated for BFR 1,3 by dividing the total number of configuration bytes by 1/2 the configuration clock frequency. For example, the MPA1036 device has 139 rows of 105 bytes including the ECB or 14,595 bytes. If the configuration clock is 2MHz, configuration will take approximately 15ms. If BFR 2 is used, the configuration process will take approximately 8 times longer. See “Device Configuration Memory Organization” on page 2–31 for device specific configuration memory sizes.

Table 2–4. BFR Mode Configuration Control Pins

Pin Name	BFR	I/O		Description
MODE[3:0]	MODE[3:0]	I	Dedicated*	**Configuration mode**
$\overline{\text{RESET}}$	$\overline{\text{RESET}}$	I	Dedicated	**Configuration reset** — Clear configuration memory. Configure when released.
CLK	CLK	I/O	Dedicated	**Configuration clock** — If MODE[2] is low, the internal configuration clock is presented. If MODE[2] is high, an external clock must be supplied.
F0	$\overline{\text{BFR}}$	I	Dedicated	**BFR initiate** — A falling edge starts a reset and configure sequence.
F1	$\overline{\text{ERR}}$	O	Dedicated	**Error** — Configuration checksum (ECB) or incorrect device ID error. Open drain output.
F2	$\overline{\text{MEMCE}}$	O	Dedicated	**Memory Enable** — Active low during configuration sequence.
F3	PWRUP	I	Dedicated	**Power up** — After configuration complete; enable bootstrap, enable user inputs, enable user outputs.
F4	END	O	Dedicated	**Configuration completed** — Asserted when a configuration has been successfully loaded into the device.
DCLK	DCLK	I/O	Dedicated	**Data clock** — Each pulse indicates current data bus value has been latched and data address should increment.
DATA[7:0]	DATA[7:0]	I	User/Data	**Data port**
ADD[17:0]	ADD[17:0]	O	User/Address	**Address output** — If internal address generation is selected. (BFR Mode 1)
JTAG[4:0] =		I/O	User/JTAG	**JTAG pins** — Active when MODE[3] is asserted.

* Dedicated — Pins used for configuration. Not available for user I/O.

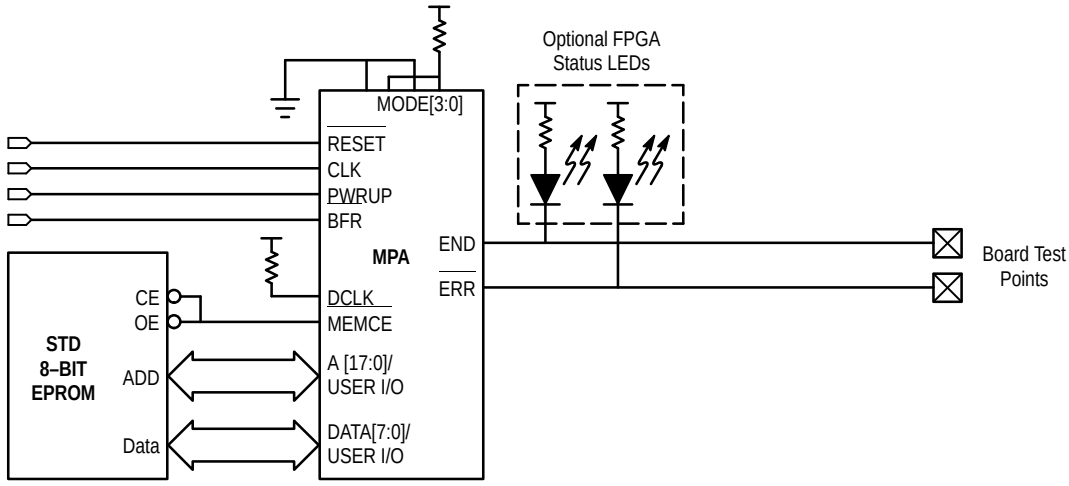


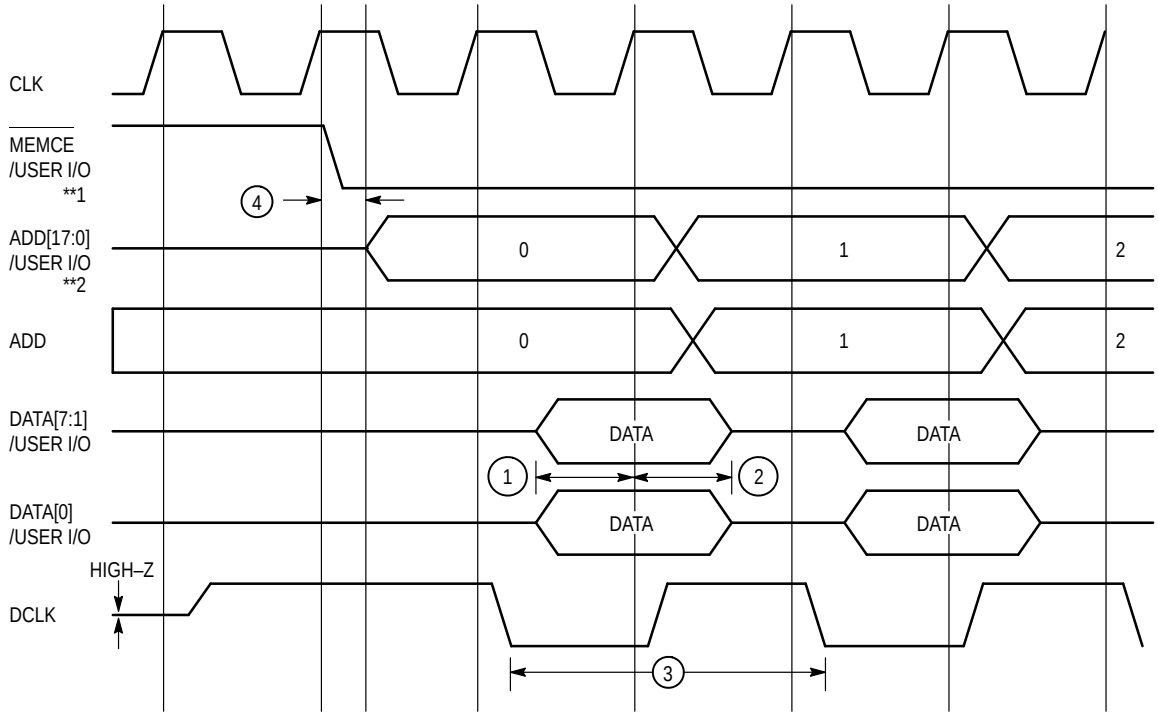
Figure 2–17. BFR Mode 1: 8–Bit Data, Internal Address, External Clock

BFR Mode 1 Operation: 8 bit data, Internal Address Generation

In BFR 1, MPA configuration logic asserts an 18 bit address and reads data 8 bits at a time as shown in Figure 2–17 on page 2–17. A paging scheme could also be used where additional upper address bits were provided by an external page register. Multiple configurations could be accessed by writing the page register, asserting BFR, and self loading the referenced configuration.

ADD[17:0] are tristated during device reset, asserted during configuration and released for user mode operation. DCLK is tristated until 1 clock prior to MEMCE assertion. The first address is asserted coincident with the falling edge of MEMCE and the data bus is latched 2 configuration clocks later. The internal address counter is incremented on each positive DCLK edge (Figure 2–18). This process proceeds until an entire row of configuration data is loaded into the internal row data register and the ECB is verified. ADD[17:0] (current address) and DCLK (=1) hold while the internal write cycle takes place. Start Access (SA) marks the beginning of the write cycle and End Access (EA) marks write completion (Figure 2–22). After the write completes, the address presentation and data latching process resumes. When the entire device configuration is loaded, END is asserted and 1

clock later user inputs are enabled, MEMCE is deasserted and DCLK tristated. One additional clock and user outputs are enabled and user mode operation commences. If the written ECB does not match the internally calculated value, ERR is asserted 2 clocks after the ECB is written. Once ERR is asserted, the configuration process halts and cannot be restarted until a new configuration process is initiated using BFR, RESET or a power down. When END is asserted, DCLK becomes an input and the internal address counter remains active until PWRUP is asserted. Figure 2–25 shows how this can be used in a multiple device subsystem. Because DCLK becomes an input, it must be tied high with a weak pullup when used in a single device configuration (Figure 2–17) to prevent a floating input condition.



**1 – Internally generated Address

**2 – Externally generated Address

Figure 2–18. BFR Data Access Detail

Number	Characteristic	Min	Max	Unit	Notes
1	Data Setup to CLK	20		ns	
2	Data Hold after CLK	0		ns	
3	DCLK Period (When Active)	2	2	CLK	
4	CLK to Address Valid (Internal Generator)	15		ns	

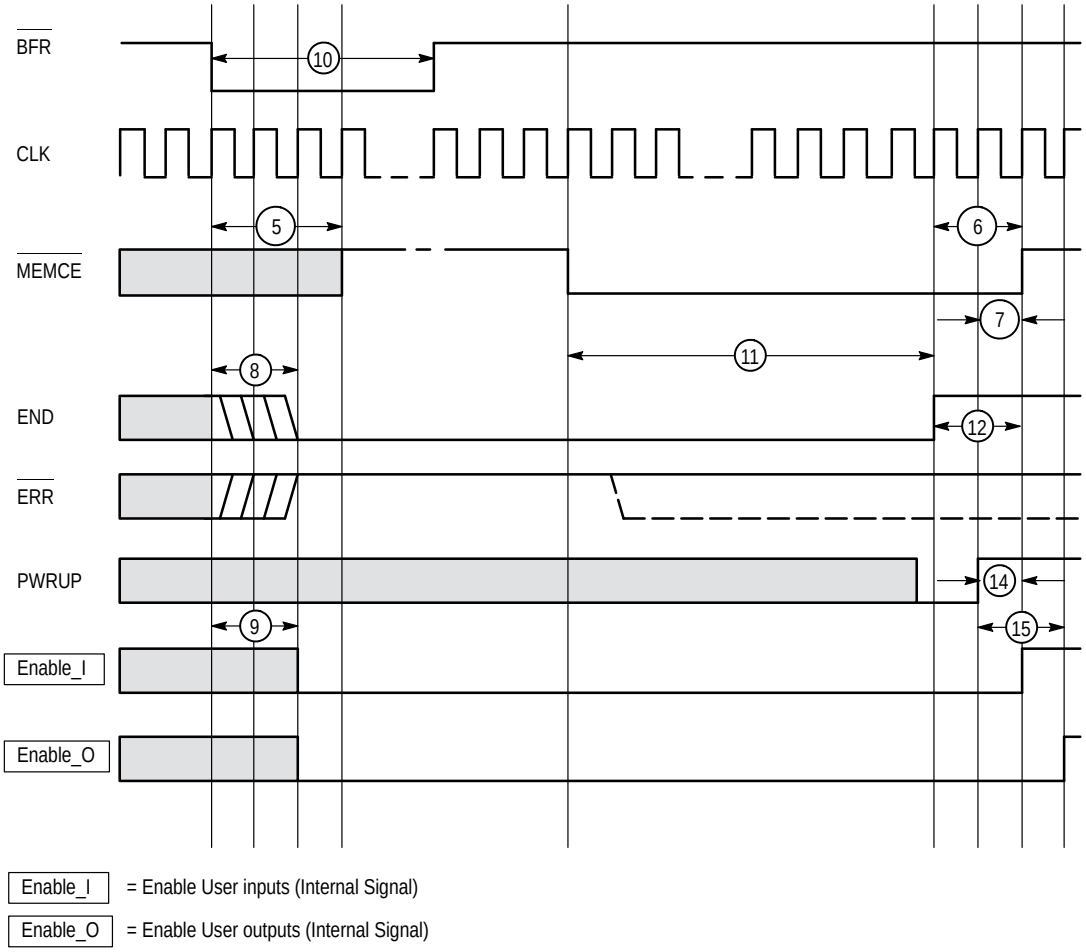


Figure 2–19. BFR Sequence

Number	Characteristic	Min	Max	Unit	Notes
5	BFR Low to MEMCE High		3	CLK	
6	END High to MEMCE High	1		CLK	
7	PWRUP to MEMCE High	1		CLK	
8	BFR Low to END Low	0	3	CLK	
9	BFR Low to Internal Disable	0	3	CLK	
10	BFR Pulse Width	50		ns	
11	Configuration Sequence Duration				Configuration sequence dependent on device size
12	END to Enable User Inputs	1		CLK	If PWRUP asserted
13	END to Enable User Outputs	2		CLK	If PWRUP asserted
14	PWRUP to Enable User Inputs	1		CLK	
15	PWRUP to Enable User Outputs	2		CLK	

BFR Mode 2 Operation: 1 bit (serial) data, External Address Generation

BFR mode 2 is used for connecting MPA devices to a serial configuration memory. The MPA device provides an address increment signal (DCLK) rather than an internally generated address as in BFR mode 1. Low pin count serial memories, like the MPA17128, contain address generation logic which responds to a single increment signal. Addressing is sequential starting at zero. Multiple MPA17128 devices can be daisy chained if a larger memory is required (See MPA17128 data sheet on page 2–52). Serial memories are programmed (written) in the opposite bit order from the way they are read. The MPA Design System configuration generation program will generate a correctly formatted PROM programming file by reflecting each configuration byte prior to writing the file.

2

MEMCE is high until configuration commences. MEMCE is connected to the RST/OE pin of the MPA17128 holding its internal address counter at 0 and its outputs tristated. The falling edge of MEMCE enables the memory data pin and 2 clocks later a data bit is latched into the MPA1000 device. The first rising edge of DCLK signals the memory to index its address register and present the next locations data bit. Each time 8 bits are accumulated by the MPA1000, they are written to the internal row data register. As in BFR 1, this process proceeds until a complete row is loaded and the ECB is verified. DCLK holds while the row data register is written to the current configuration memory row. After the write completes, additional bits are loaded until the next row boundary is reached. Configuration completion and error indications are identical to BFR 1.

BFR Mode 3 Operation: 8 bit data, External Address Generation

BFR mode 3 is identical to BFR mode 2 except that 8 bits of data are loaded rather than one. An external address generator is used and responds to the MPA address increment signal (DCLK). BFR mode 3 is useful because BFR 1 requires 18 user I/O signals (ADD[17:0]) during configuration. While these are subsequently released, it does impose restrictions on surrounding circuitry complicating overall system design. Secondly in applications requiring rapid configuration of a large number of MPA devices or many alternate configurations, the MPA 18 bit address space may not be large enough and an external counter (address generator) would required anyway.

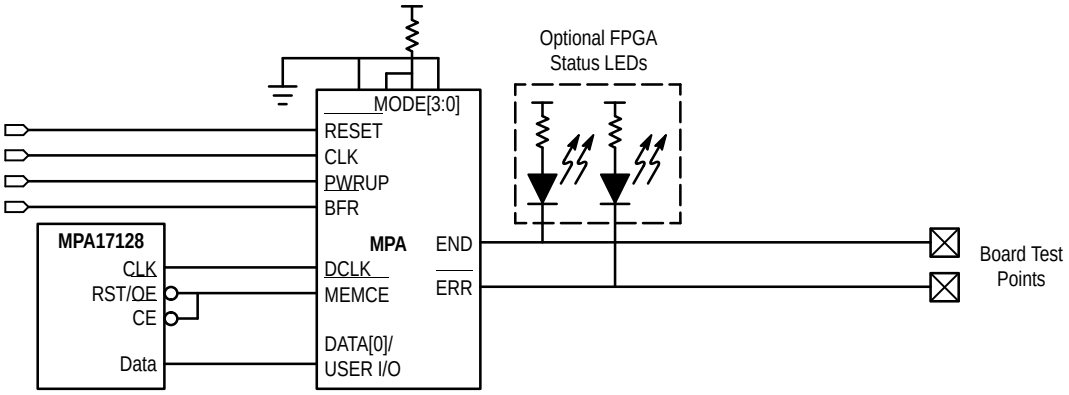


Figure 2–20. BFR Mode 2: 1–Bit (Serial) Data, External Address, External Clock

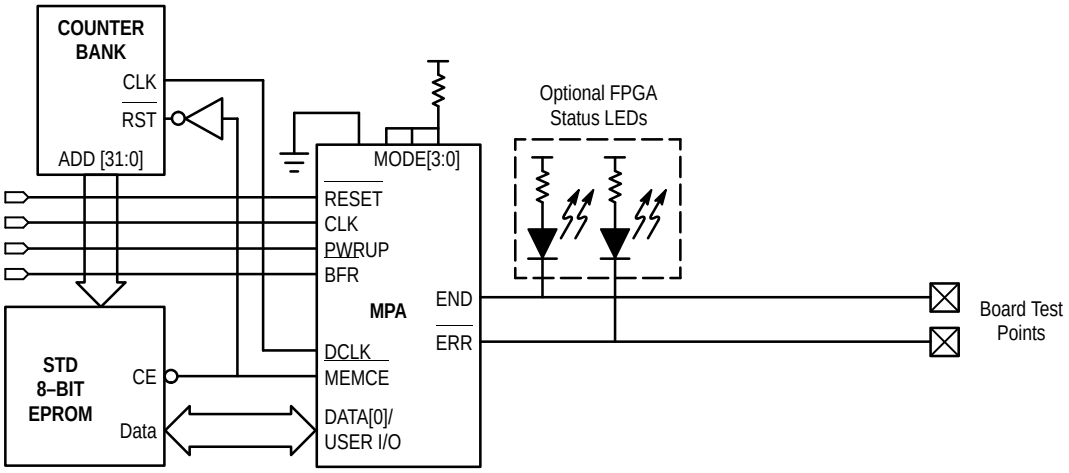


Figure 2–21. BFR Mode 3: 8–Bit Data, External Address, External Clock

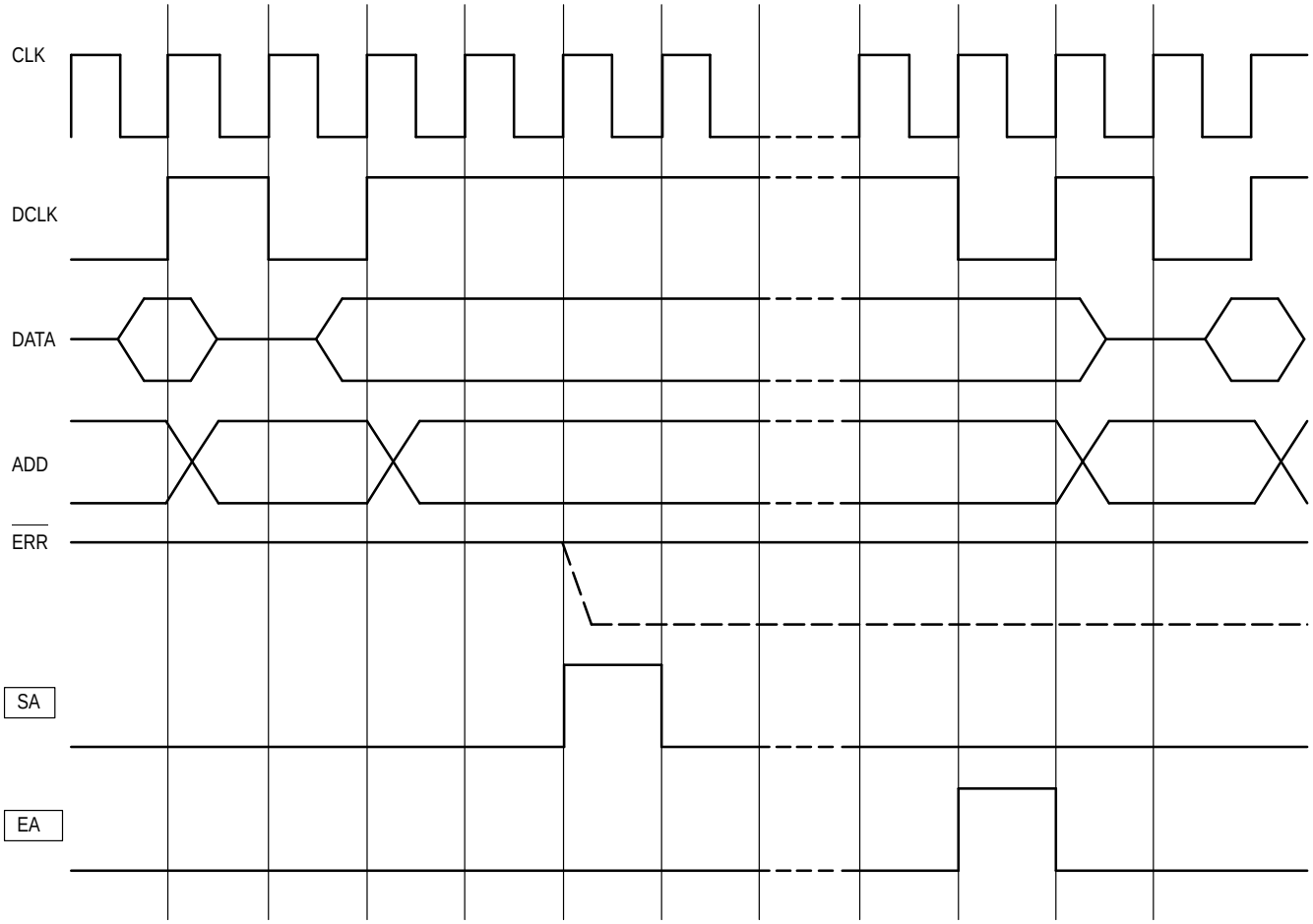


Figure 2–22. End of Row Behavior

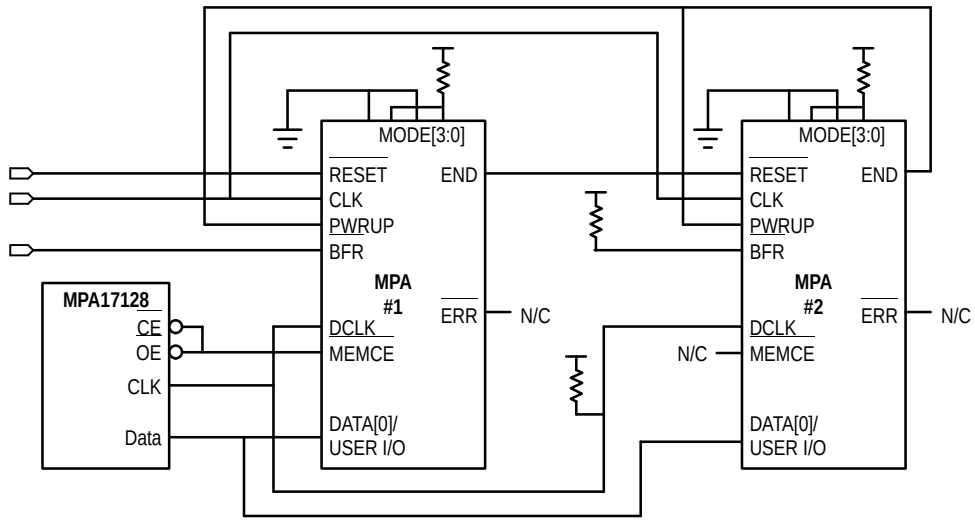


Figure 2–23. Multiple Device Subsystem: BFR2; Serial Data, External Address, External Clock

BFR Multiple Device Subsystems

If multiple devices are used together in BFR mode, the first device loads first and the END signal on each device is connected to the RESET pin of the next device. As an upstream device completes configuration, a configuration sequence is initiated on the next device. This daisy chain extends to the last device. This devices END is connected to the PWRUP pins of all subsystem MPA devices. All devices enter user mode when the last device successfully configures.

Care must be taken to insure proper operation. $\overline{\text{BFR}}$ on all but the first device must be tied high and the subsystems composite DCLK line must be pulled up to eliminate spurious clock signals as one device tristates DCLK and the next device asserts it. Figure 2–24 illustrates the control signal hand off.

When constructing a subsystem in which the first device asserts the 18 bit address (BFR mode 1), this device provides address generation for all devices in the subsystem. The DCLK pin of the first device becomes an input when it successfully configures and its internal counter remains active. Positive edges applied to this pin will increment the first devices internal address counter and present the resulting address on the first devices 18 bit address bus. Subsequent devices in this subsystem should use BFR mode 3 (external address, 8 bit data).

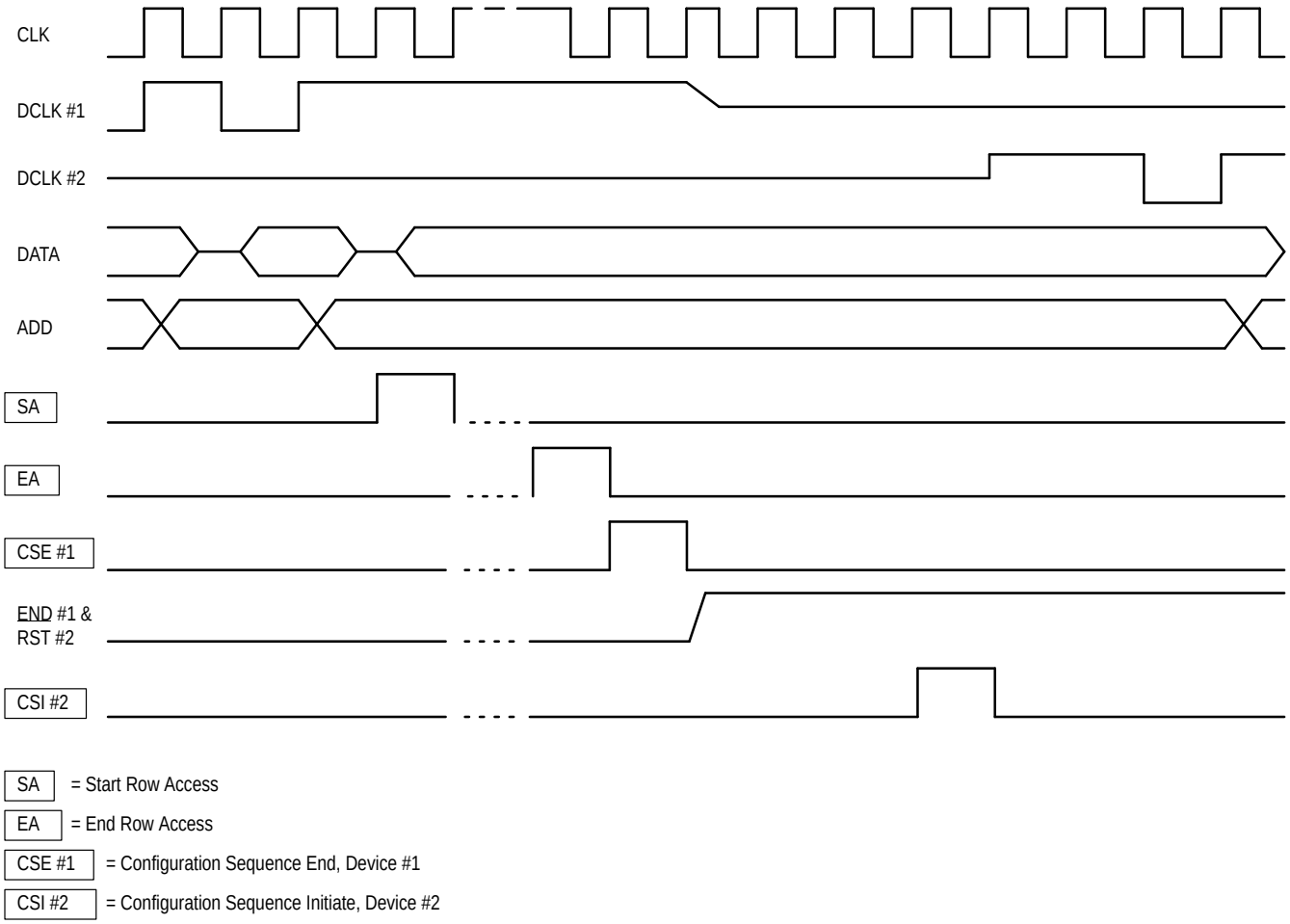


Figure 2–24. BFR Mode Daisy Chain Timing

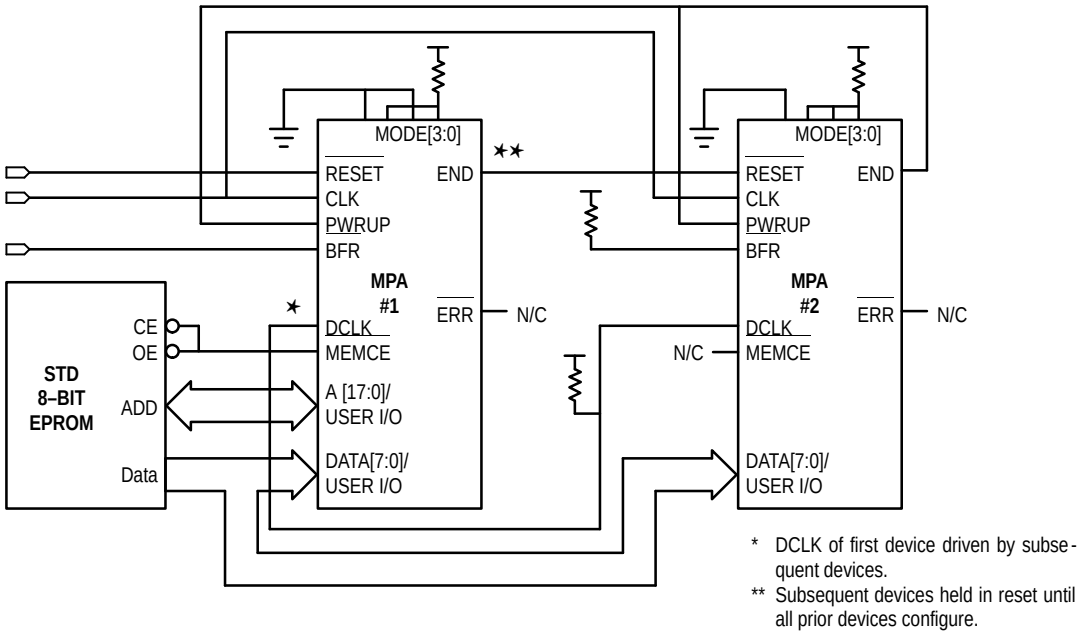


Figure 2–25. Multiple Device Subsystem: BFR1 and BFR3; 8 Bit Data, Internal Address, External Clock

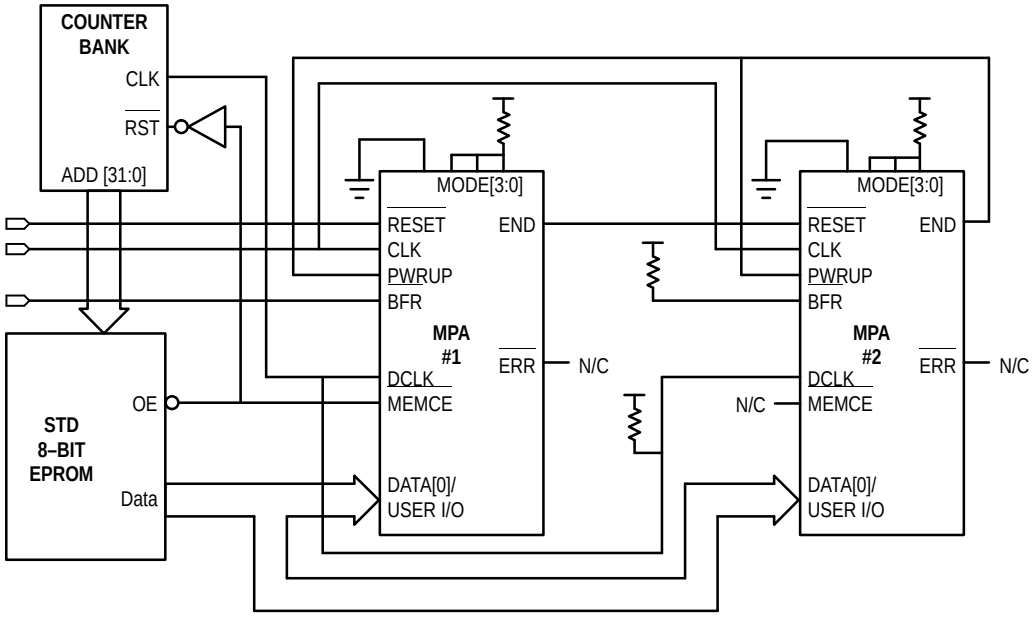


Figure 2–26. Multiple Device Subsystem: BFR3, 2; 8 Bit Data, External Address, External Clock

MICRO Mode

In MICRO Mode the MPA device behaves as an asynchronous microprocessor peripheral. Table 2–5 on page 2–24 details MICRO Mode configuration pin function. A chip select (/CS) is derived from the processor address and enables a single MPA device. In a multiple device subsystem, a chip select for each MPA device is required. When a device is selected, the data bus is used to write commands, read status, write configuration data and read configuration data. There are two device configuration registers, the function register (RS=0) and the data/status register (RS=1). Configuration commands are written to the function register. Subsequent behavior is specific to the command issued and is documented in Table 2–6. The data register is either used to read device status, read device configuration data or write device configuration data. RS is normally connected to the least significant address line to map the function register to address A and the data/status register to address A+1.

Configuration data format information can be found in the "Device Configuration Data Format" section on page 2–31. Configuration data is generated by the MPA design system configuration generator after a layout is complete.

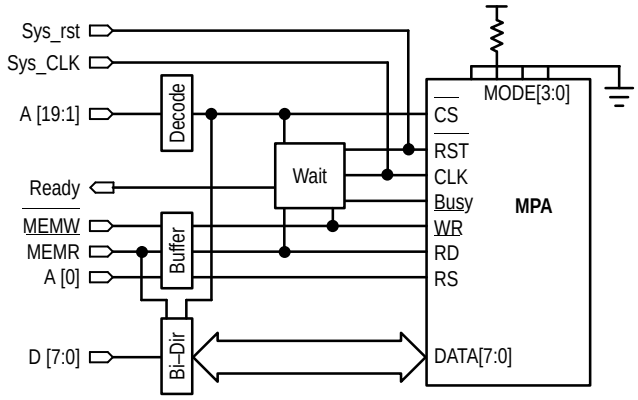


Figure 2–27. MICRO Mode: Single Device With External Clock and Wait State Insertion

Table 2–5. MICRO Mode Configuration Control Pins

Pin Name	Micro	I/O		Description
MODE[3:0]	MODE[3:0]	I	Dedicated*	**Mode Pins**
$\overline{\text{RESET}}$	$\overline{\text{RESET}}$	I	Dedicated	**General configuration reset**
CLK	CLK	I/O	Dedicated	**Clock for configuration circuitry** — If external clock is selected, pin is an input. If not selected internal configuration clock is used and output through this pin.
F0	$\overline{\text{CS}}$	I	Dedicated	**Chip select** for device in MICRO Mode.
F1	$\overline{\text{RD}}$	I	Dedicated	**Micro read signal**
F2	$\overline{\text{WR}}$	I	Dedicated	**Micro write signal**
F3	RS	I	Dedicated	**Register select** — Two register locations are active: Function Register (RS = 0) and Data (RS = 1).
F4	Busy	O	Dedicated	**Busy signal** — Active high when device is not ready to accept data, i.e. while device is resetting data in array or a data register to array transfer is taking place.
DATA[7:0]	DATA[7:0]	I/O	Dedicated	**Micro data port** — for configuration logic.
JTAG [4:0]	J [4:0]	I/O	User/JTAG	**JTAG pins** — JTAG or User I/O is selected by MODE[3].

* Dedicated — Pins used for configuration. Not available for user I/O.

Table 2–6. MICRO Mode Function Register

DATA [7:4]	DATA [3:0]	Function
XXXX	0000	**Normal operation** — No function performed.
XXXX	0001	**Reset Device** — Entire device configuration is reset. BUSY is asserted until reset completes.
XXXX	0010	**Load Configuration** — After writing this command, an entire normal format device configuration is presented to the data register in 8 bit segments starting with the configuration header block. At any time during the loading process, a read to the data register will return status register contents. As complete rows including ECB are loaded, BUSY is temporarily asserted while row data is internally transferred from the internal data register to the currently addressed memory row. Once this write operation is complete, BUSY is deasserted and additional data can be written. Each time BUSY is deasserted, the status register should be checked for incorrect ID or row configuration data error(s). Once an error is detected, NO further write accesses to the data register will be accepted until the device is reset or another load configuration command is issued.
XXXX	0011	**Reset Row** — Indicates that the next data written to the data register will be a device row address. After the address is written, the contents of that configuration memory row are reset. BUSY is asserted after the address is written and deasserted when the operation is complete.
XXXX	0100	**Load Row** — The next data written to the data register consists of a row address followed by configuration data for that row including the terminating ECB. After the ECB is written, BUSY will be asserted during internal write and deasserted when the write completes. Reading the data register returns status register contents. The status register should be checked for row configuration data error(s). Once an error has been detected, NO further write accesses to the data register will be accepted until the device is reset or a load configuration command is issued.
XXXX	0101	**Read Row** — The next data written to the data register will be interpreted as a row address. After the row address is written, BUSY is asserted while row data is read into the internal data register. BUSY is deasserted when the transfer is completed. Subsequent successive reads from the data register will return row configuration data. No ECB is returned. The row data read back is in the same order as it is written, rightmost byte first.
XXXX	0110	**Read Device ID** — 4 subsequent reads from the data register return device ID. The most significant ID byte is read first. Refer to “configuration data format” for individual device ID values.
—	0111	**Bits [3:0]** — Reserved pattern.
—	1XXX	**Bits [3:0]** — Reserved pattern.
1XXX	XXXX	**Bootstrap Enabled** — Should be enabled after configuration is completed and disabled during configuration.
X1XX	XXXX	**Internal Oscillator Disabled** — Normally always enabled. Can be disabled if external clock and V_{PP} are supplied. If internal configuration clock is used, oscillator cannot be disabled.
XX1X	XXXX	**User Inputs Enabled** — Normally user inputs are enabled after a configuration is successfully loaded into the device.
XXX1	XXXX	**User Outputs Enabled** — Normally user outputs are enabled one or more clocks after user inputs are enabled to insure valid input values have propagated into the device.

Table 2–7. MICRO Mode Status Register

Bit Position								Function
[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	
0	0	0	0	0	X	X	1	Incorrect Device ID.
0	0	0	0	0	X	1	X	Row configuration data error. ECB mismatch.
0	0	0	0	0	1	X	X	Busy signal asserted. Allows software handshaking if hardware wait states are not to used.

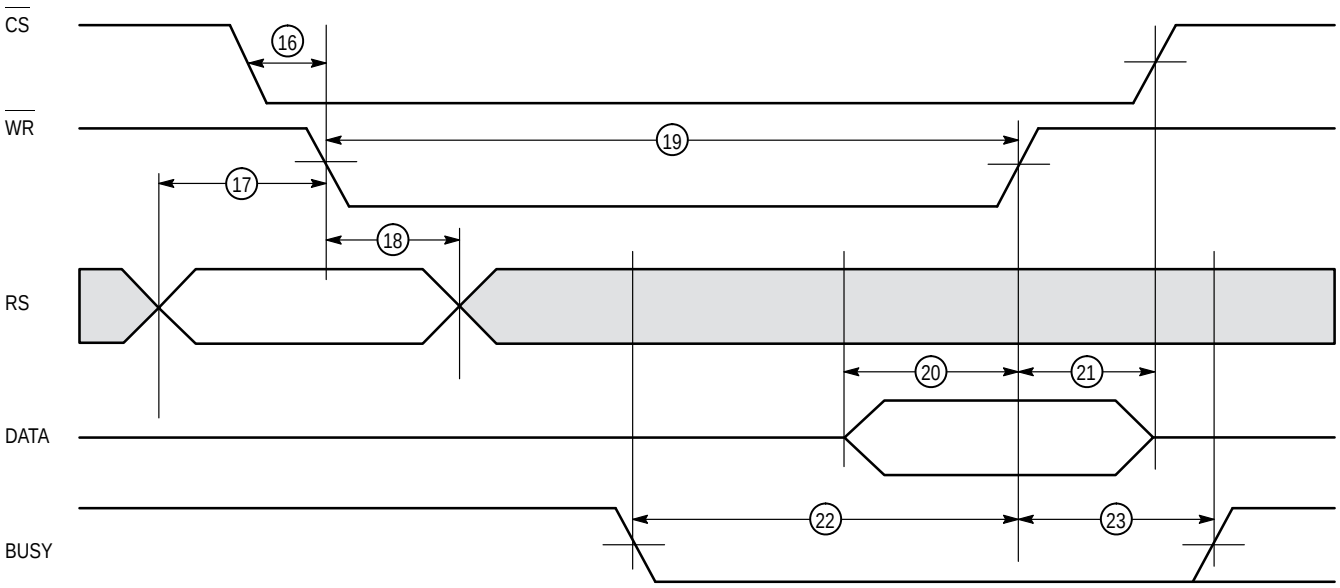


Figure 2–28. MICRO Mode External Timings (Write Cycle)

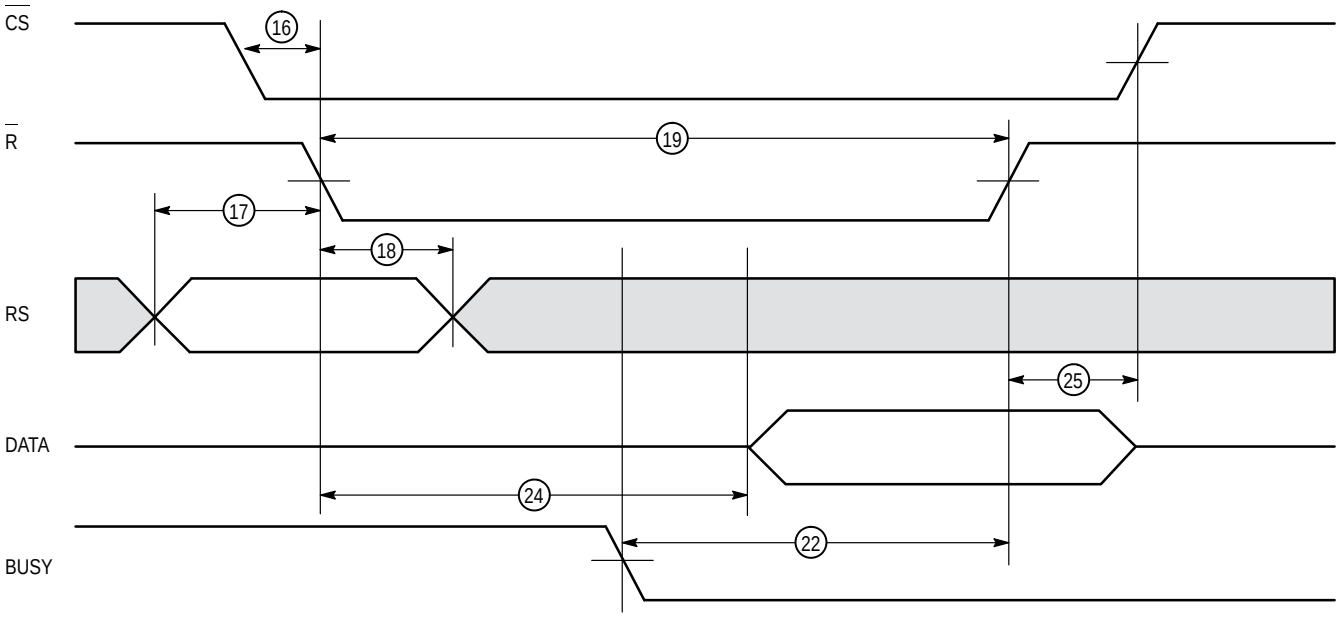


Figure 2–29. MICRO Mode External Timings (Read Cycle)

Number	Characteristic	Min	Max	Unit	Notes
16	$\overline{CS}$ Setup before Read/Write	10		ns	
17	RS Setup before Read/Write	10		ns	
18	RS Hold after Read/Write Falling Edge	10		ns	
19	Read/Write Pulse Width	50		ns	
20	Data Setup to End of Write	20		ns	
21	Data Hold after Write	10		ns	
22	Busy Inactive before End of Read/Write	50		ns	
23	Busy Active after Write	0	20	ns	
24	Data Access Time	20	40	ns	
25	Data Hold Time after Read	0	10	ns	

MICRO Mode Maximum Data Transfer Rate

The maximum MICRO Mode data transfer rate is governed by the R/W timing described in Figure 2–28 and Figure 2–29. The processor must only write data when BUSY is inactive. BUSY is only asserted when data cannot be accepted at the maximum rate. The specific behavior of BUSY for each MICRO Mode function is described in Table 2–6. When the device is powered up, an internal reset sequence is initiated and BUSY is asserted (see “Behavior During Power–On–Reset”). BUSY will be deasserted when the internal reset sequence completes. The processor can monitor BUSY directly or the status register can be read.

If processor R/W cycles are faster than the timing shown, external circuitry must be used to insert wait states. Figure 2–27 and Figure 2–34 show an application circuit consisting of one or more MPA devices and an optional wait state insertion block used to lengthen R/W timing based on CS, MEMW, MEMR, BUSY, and RESET using an externally provided clock.

Multiple Devices in MICRO Mode

If multiple devices are used in MICRO Mode, external logic is required to individually address each MPA device using CS (chip select) signals. After configuration, the processor must write bootstrap enable, enable inputs and enable outputs commands to each device. A subsystem BUSY signal can be derived by OR–ing the BUSY signals from each individual device. Refer to Figure 2–34.

Internal Clock Specification

The internal ring oscillator is a clock source with possible frequencies ranging from 10MHz to 40Mhz. This variation is expected and does not present a problem for proper charge pump or configuration operation. The internal configuration clock is derived by dividing the oscillator frequency by 8. The internal configuration clock can be used for user mode operation and is presented on the CLK pin when MODE[2] is low.

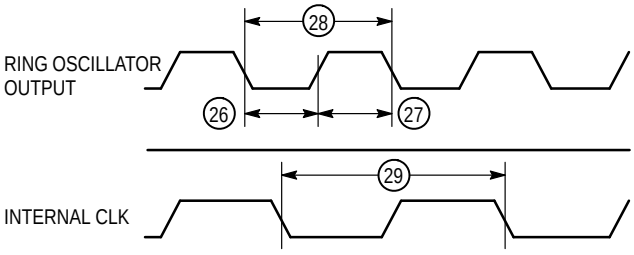


Figure 2–30. Internal Oscillator and Clock Specification

Num	Characteristic	Min	Typ	Max	Unit
26	Ring Oscillator Low	10	25	50	ns
27	Ring Oscillator High	10	25	50	ns
28	Ring Oscillator Period	25	20	100	ns
29	Internal Clock Period	200	400	800	ns

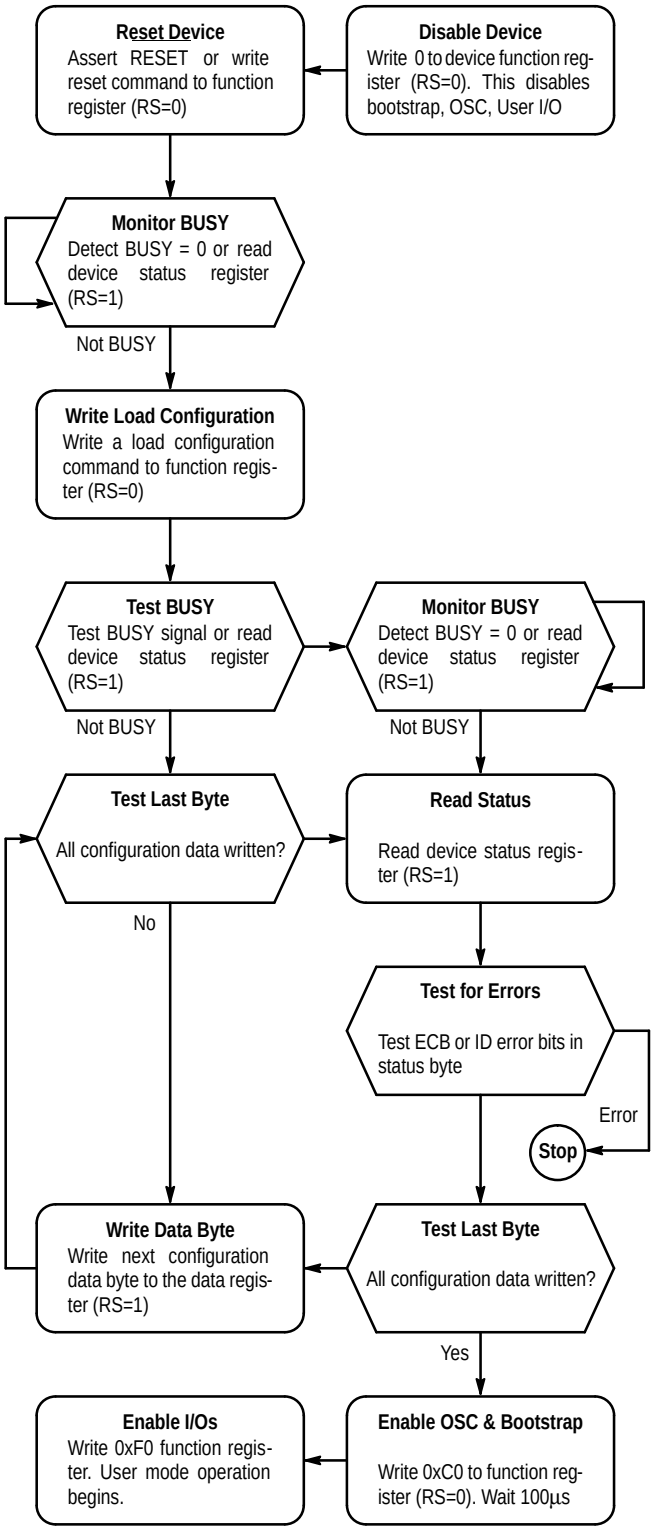


Figure 2–31. MICRO Mode Configuration Load Sequence Example

External Clock Specification

To improve configuration performance an external clock can be connected to the CLK pin when MODE[2] is asserted. The specifications for this clock are given in figure 1–26.

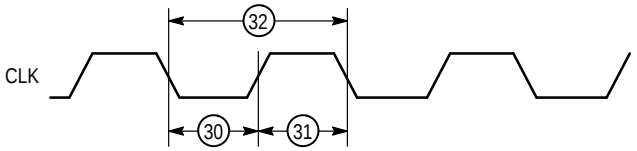


Figure 2–32. External Clock Specification

2

Num	Characteristic	Min	Max	Unit	Notes
30	External Clock Low	10		ns	
31	External Clock High	10		ns	
32	External Clock Period	25		ns	

Power On Reset Operation

The MPA1000 devices contain circuitry to insure reliable self configuration when power is applied to the device. An counter clocked by the internal configuration clock and triggered by an analog power on reset circuit delays configuration until the power supply has been given sufficient settling time (Figure 2–35).

The analog power on reset circuit provides a reliable signal (APOR) to indicate that V_{DD} is sufficient to reliably operate device logic. While APOR is low a 17 bit counter is held reset. When APOR is asserted, the counter is enabled and POR occurs when the most significant counter bit reaches 1. Between APOR assertion and POR, the configuration circuitry is continuously resetting configuration memory row by row. When POR is asserted, a final internal reset sequence is performed (Figure 2–35). If an external clock is selected by asserting MODE[2], this selection will not take effect until 4 internal configuration clocks after POR occurs.

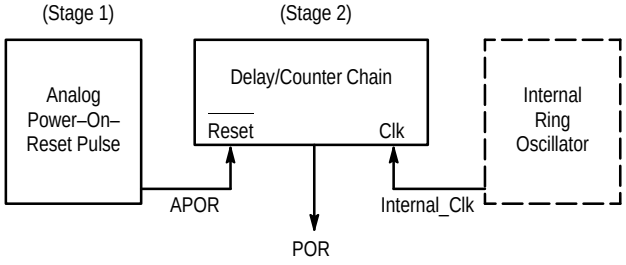


Figure 2–33. 2–Stage Power–On–Reset

External Reset

A reset sequence is initiated by a falling edge of RESET. Once a reset sequence is initiated it cannot be terminated by a subsequent rising edge of RESET.

If RESET is low when the internal reset sequence completes, configuration will not commence until RESET is deasserted (Figure 2–36). This feature can be used to hold off configuration until other external events occur. This feature is used in conjunction with the multiple device daisy chain. Figure 2–37 shows RESET effect on other configuration mode, internal reset and internal configuration signals.

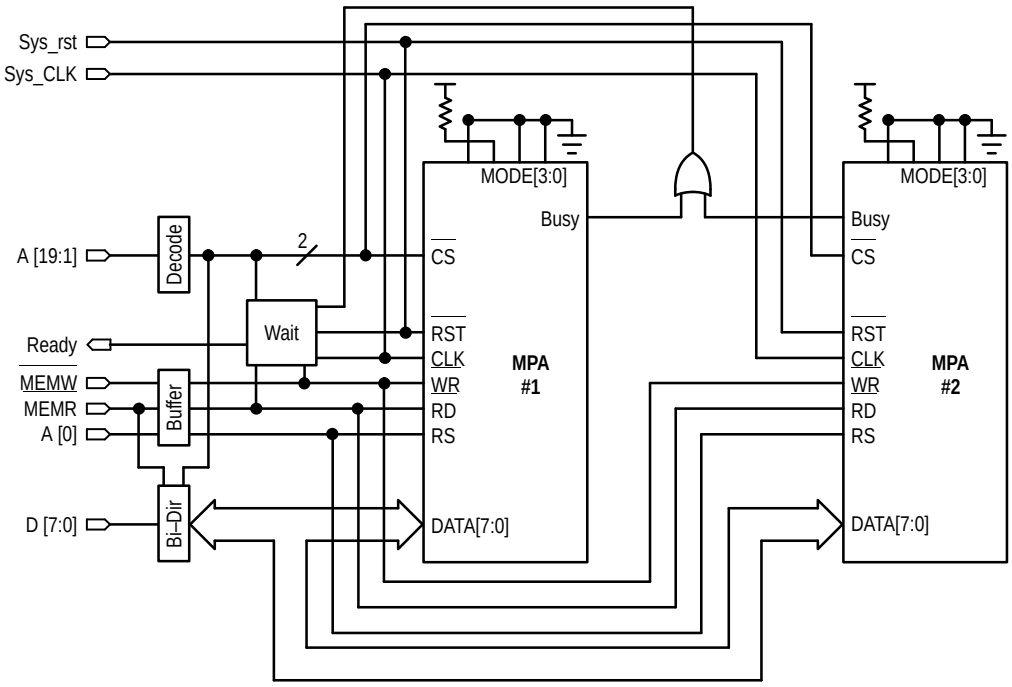


Figure 2–34. MICRO Mode: Multiple Devices With External Clock and Wait State Insertion

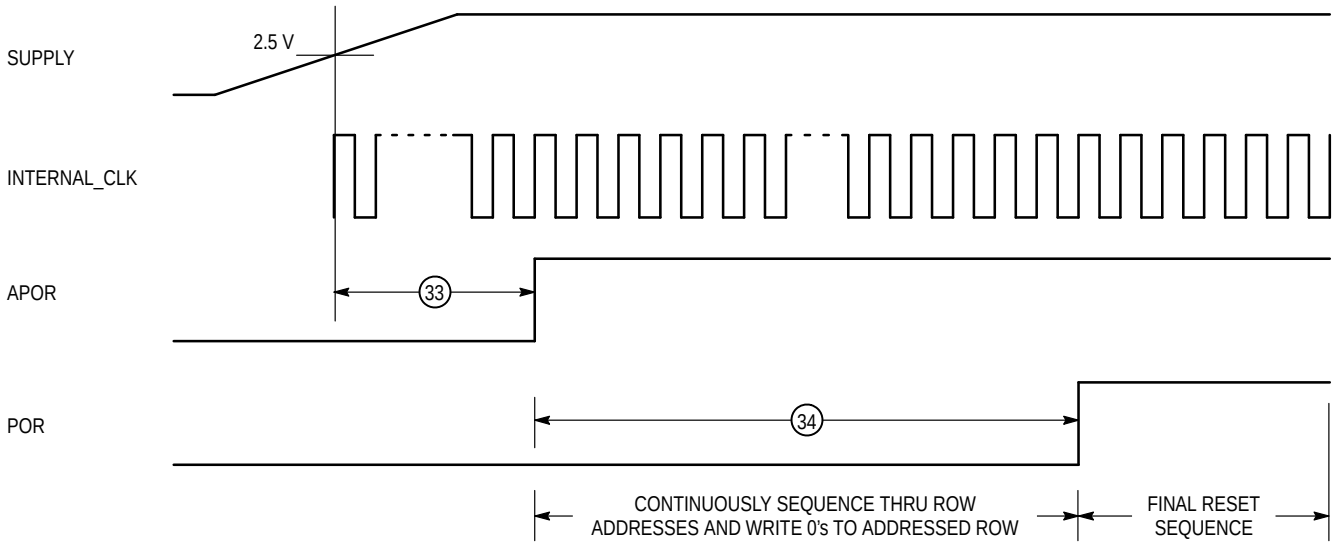


Figure 2–35. Power–On–Reset Circuitry Timing

Number	Characteristic	Min	Max	Unit	Notes
33	APOR	10	1000	µs	
34	POR (Active)	13.2	52.5	ms	

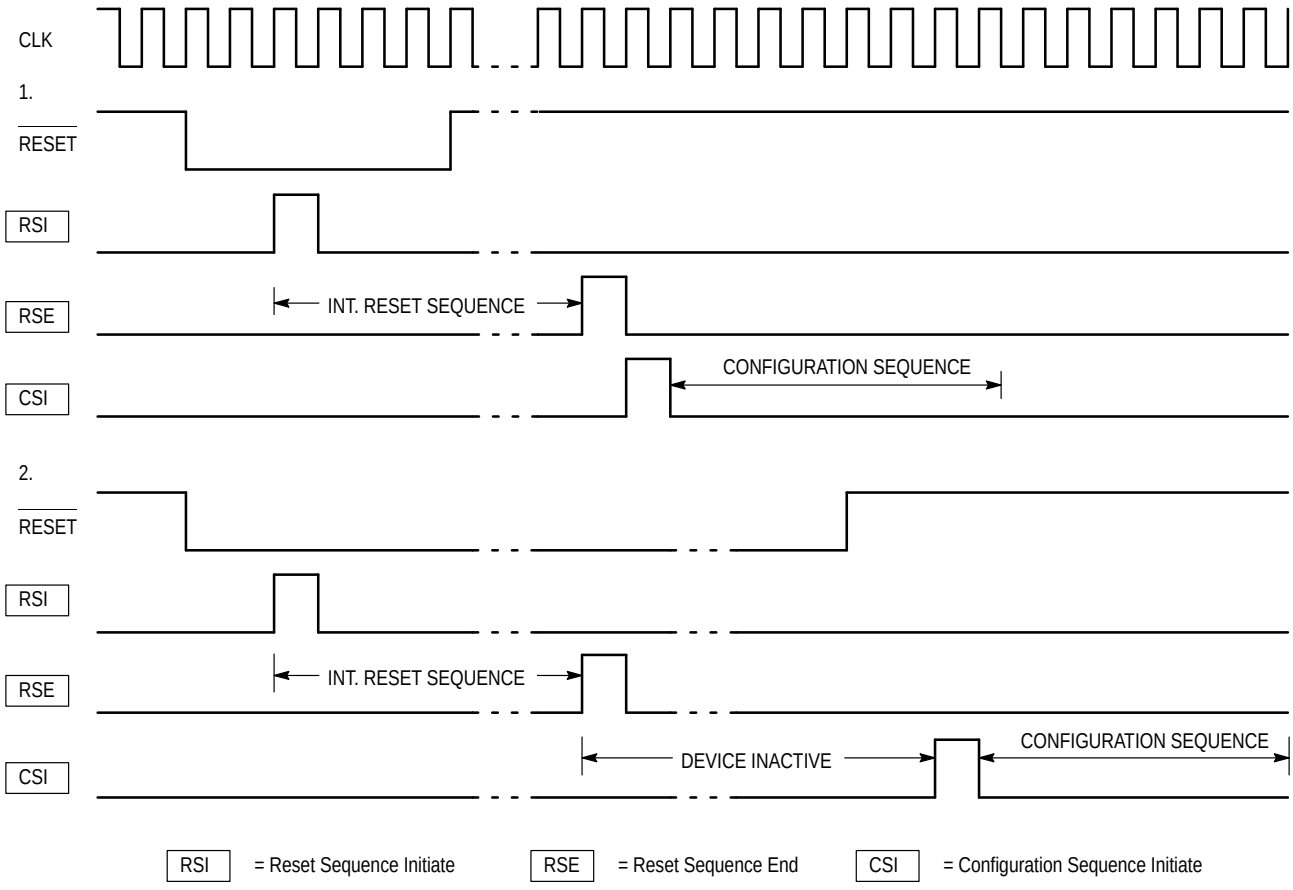
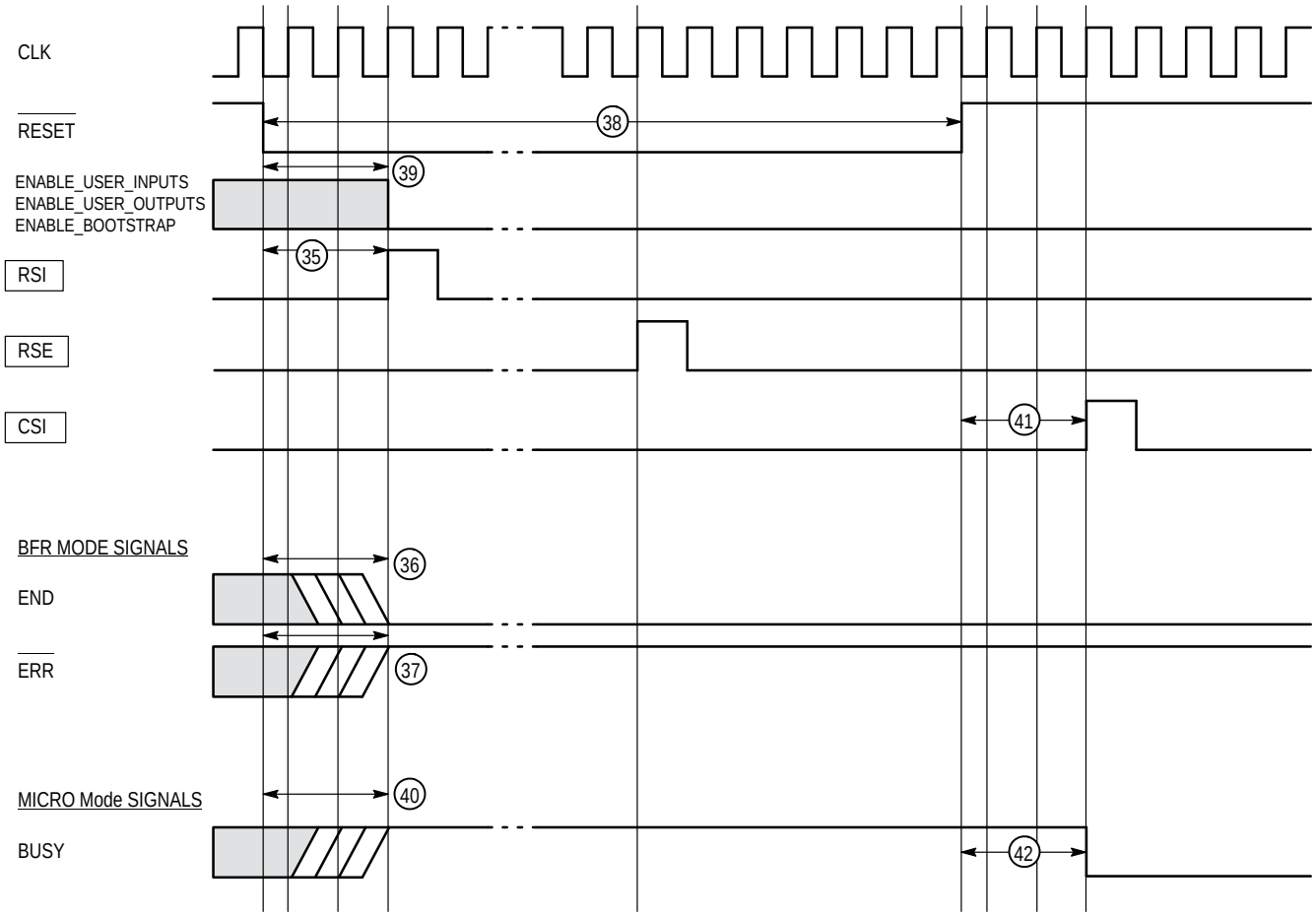


Figure 2–36. External Reset Behavior



RSI = Reset Sequence Initiate

RSE = Reset Sequence End

CSI = Configuration Sequence Initiate

Figure 2–37. External Reset Timing

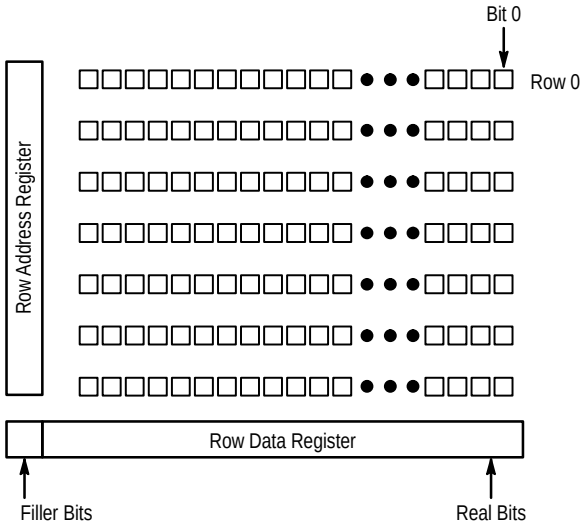
Number	Characteristic	Min	Max	Unit	Notes
35	RESET Low to Reset Sequence	2	3	CLK	
36	RESET Low to END Low	0	3	CLK	
37	RESET Low to ERR High	0	3	CLK	
38	RESET Pulse Width	50		ns	
39	RESET Low to Internal Disable	0	3	CLK	
40	RESET Low to Busy Active	0	3	CLK	
41	RESET High to CSI Pulse	2		CLK	RESET Released After RSE
42	RESET High to Busy Inactive	2		CLK	RESET Released After RSE

In MICRO Mode, the busy signal remains high while the reset signal is asserted and until the internal reset sequence is completed.

MPA1000 Configuration Data Format

Device Configuration Memory Organization

The MPA1000 devices are programmed by loading configuration data into on chip configuration memory constructed of SRAM cells. This memory is organized differently from standard memory products. The configuration SRAM is distributed throughout the MPA device. Data is read and written to the device 1 row at a time via the internal row data register (RDR). Individual rows are addressed via the row address register (RAR). Each device has a different size RDR and RAR (Figure 2–38). The configuration logic is responsible for the control of these resources.



Device	# Rows	# Real Bits	# Filler	# ECB	Total Row Bits
1016	95	562	6	8	576
1036	139	828	4	8	840
1064	183	1090	6	8	1104
1100	227	1352	0	8	1360

Figure 2–38. Device Memory Organization

Configuration logic writes data to the leftmost (most significant) RDR byte and reads from the rightmost (least significant) RDR byte. Each of these transfers occurs in 8 bit increments. When serial data is presented, the bits are accumulated into a byte before RDR transfer. Each configuration logic RDR write operation first shifts the RDR eight 8 bits and transfers the new byte into the leftmost RDR byte position. Configuration read operations transfer the rightmost RDR byte to the configuration logic and then shifts RDR contents right 8 bits.

The RAR enables a single configuration memory row. MPA configuration logic writes a row addresses into the RAR. Subsequent read or write operations are performed between the RDR and the RAR selected row in parallel.

Filler bits are used to round the RDR up to the nearest byte boundary. The ECB is not part of the RDR. During configuration a single row data vector is written to the RDR and an ECB is calculated from the data written. The calculated value is compared to the ECB contained in the data vector. If a mismatch is detected, ERR is asserted and the configuration process terminates. The ECB mechanism prevents data write disturbances from causing unpredictable device function.

Device Configuration Data Formats

When whole configurations are loaded into a device, the first 40 bits contain a 32 bit device ID followed by an 8 bit data type field. The device ID is the same as the JTAG device ID described in “JTAG Boundary Scan”. If an incorrect ID is presented, ERR is asserted and configuration stops. Device ID comparison prevents incompatible configurations from causing unpredictable device behavior. The data type field identifies subsequent data format. Recognized data types are shown in Table 2–8.

Table 2–8.

[7:3]	[2]	[1]	[0]	Data Type
00000	X	X	0	Sequential data (Normal data)
00000	X	X	1	Test data – Multiple row access
00000	X	0	X	Unencrypted data
	X	1	X	Encrypted data – Not supported on first product. **Reserved** for future implementations
	0	X	X	Uncompressed data
	1	X	X	Compressed data – Not supported on first product. **Reserved** for future implementations

Header Block

Device ID [3]
Device ID [2]
Device ID [1]
Device ID [0]
Data Type

Two data formats are supported; Normal data and test data. Normal data is generated by the MPA Design System and is the only data type users are expected to use. Test data is a special format developed to aid device testing where many very regular configuration patterns must be rapidly loaded during production test. Test mode data only results in a memory savings when many rows of configuration memory contain identical information. Since this is unlikely for real designs, test mode data offers little or no benefit for reducing user configuration memory storage requirements. If configuration data compression is desired a simple Run Length Encoding (RLE) approach can be used to exploit the fact that most configuration memory bits will be 0.

Normal data consists of a series of configuration memory row images including filler bits and ECB. Each device has a different number of bytes per row and a different number of rows. A generalized normal data representation is shown in Figure 2–39. Bytes are presented to the device from left to right and from top–most row (row 0) to bottom–most row. The ECB is calculated by summing the row data byte by byte, complementing the carry and using this as the carry into the next addition.

Data 0 (*Row 0*)	Data 1 (*Row 0*)	~	~	~	ECB 0
Data 0 (*Row 1*)	Data 1 (*Row 1*)	~	~	~	ECB 1
		”	”	”	
		”	”	”	
Data 0 (*Row x*)	Data 1 (*Row x*)	~	~	~	ECB x
Data 0 (*Row y*)	Data 1 (*Row y*)	~	~	~	ECB y

Figure 2–39. Configuration Data Block (Normal Data)

Test data format is similar to normal data except that a row count and address list follows the ECB. The RDR is loaded and the ECB calculated normally. Each address is written to the RAR, a write cycle initiated to transfer the RDR to the addressed configuration memory row, the expected address count is decremented and the next address is loaded until the expected address count reaches zero. The next byte is assumed to be the first byte of a new row data vector. Configuration ends when a row address of 255 is presented. Figure 2–40 shows the generalized test data format.

Data 0	Data 1	~	ECB M	No. Rows	Row A
					Row B
					Row C
					Row D
			”		
Data 0	Data 1	~	ECB N	No. Rows	Row E
					Row F
					Row G
					Row 255

(Row 255 = Configuration Terminating Byte)

Figure 2–40. Test Data Configuration

MPA1000 JTAG Boundary Scan

JTAG Boundary Scan Functions

JTAG is a standardized boundary scan methodology used for board level testing to detect faults in package and board connections, as well as internal circuitry. The MPA1000 JTAG boundary scan cell is designed to meet the IEEE std. 1149.1 for testability test of an integrated circuit.

IEEE 1149.1 Architecture

Figure 2–41 shows the general diagram of the IEEE 1149.1 MPA1000 JTAG system. Its design is compatible to Motorola H4C and H4C+ family of arrays. The MPA1000 JTAG design is hard wired.

TAP and I/O Periphery Signals

The TAP (Test Access Port) consists of five externally accessible signals which are used to control and observe boundary scan data. These five pins; TCK, TMS, TDI, TRSTB, and TDO are multiplexed with normal signal pins. After JTAG testing, these pins can be programmed as normal I/O pins when MODE[3] is deasserted. The test clock pin, TCK, is used to synchronize all JTAG functions. The TCK, TMS and TRSTB control the TAP controller. TDI is the test data input pin and TDO is the test data output pin.

JTAG Control and Test Register

The **TAP Controller** is a synchronous, 16–state machine, which selects the mode of operation for the test circuitry. An example of the operation of the TAP controller is shown in Figure 2–42 where the TAP controller is sequenced through most of its test states.

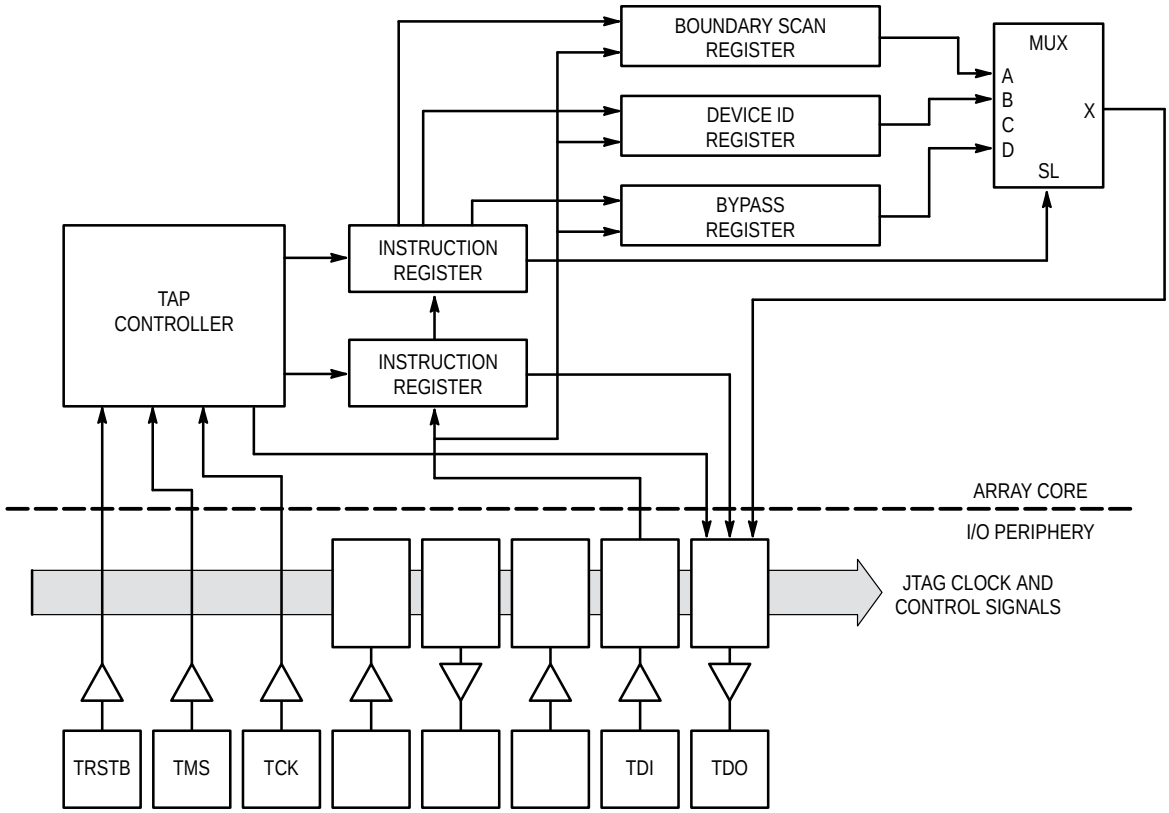
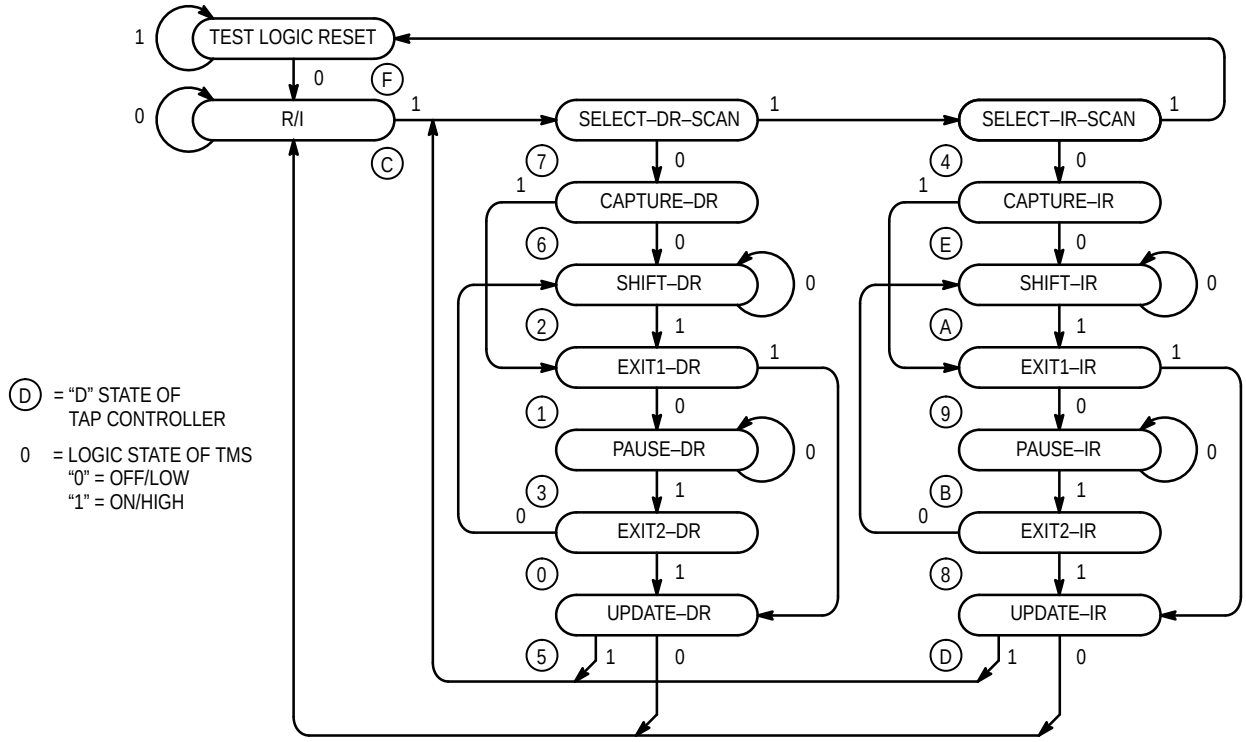


Figure 2–41. JTAG System



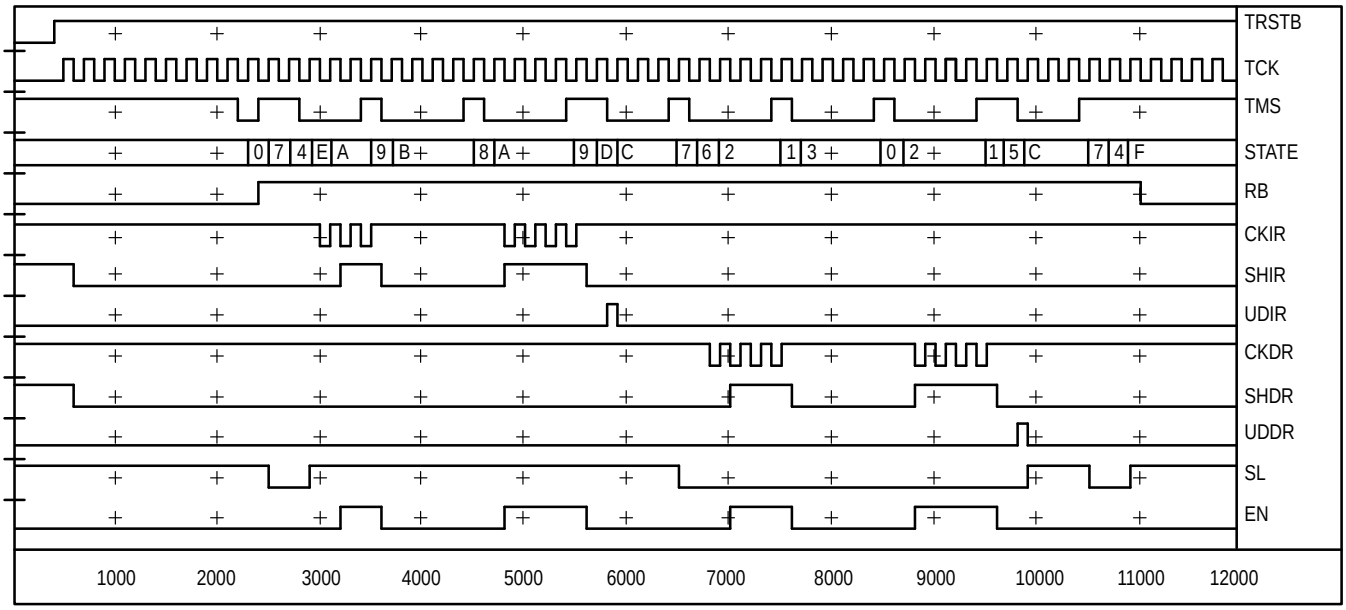


Figure 2–42. TAP Controller and Test Cycle

The **Instruction Register** is a 3–bit shift register, which permits an instruction to be shifted into the design to select the test to be performed. The **Instruction Decode** translates the instruction into separate control signals. Table 2-1 shows the basic public instructions supported by Motorola's FPGA:

Table 2-1. Basic Public Instruction

I_2	I_1	I_0	Public Instruction	Register Selected
0	0	0	EXTEST	Boundary Scan Cell
0	0	1	INTEST	Boundary Scan Cell
0	1	0	SAMPLE	Boundary Scan Cell
1	0	0	IDCODE	Device Register
1	1	1	BYPASS	Bypass Register

- **EXTEST** (external test) is the boundary scan test that checks board interconnections between integrated circuits(I.C.s).
- **INTEST** (internal test) checks the logic internal to I.C.s.
- **SAMPLE** test samples data at the I/O pins of an I.C. during normal operating mode.
- **IDCODE** instruction outputs the identification code of the I.C.
- **BYPASS** instruction redirects the test data from TDI directly to TDO, effectively removing the I.C. from the boundary scan chain.

The **Bypass Register** is a single–bit shift register used to provide a shortest path between TDI and TDO.

The **Device Identification Register** is a 32–bit register which holds a manufacturer's identity code, part number and version code. The bit assignment for the ID code is given in Table 2-2.

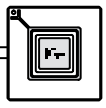
Table 2-2. Device Register ID Codes

Bit Number	Code Use
0–11	Motorola Identification
12–21	Array Identification
22–27	Programmable Logic Products Identification
28–31	Version Number

For example, for MPA1036 & MPA1064, the ID codes are listed as follows:

Array	ID code
MPA1016	0001 001110 0100001110 000000011101
MPA1036	0001 001110 0100011110 000000011101
MPA1064	0001 001110 0100110100 000000011101
MPA1100	0001 001110 0101000000 000000011101

The **Boundary Scan Register** is the chain of JTAG boundary scan cells that are linked together to form a shift register around the periphery of the array. The test data enters the boundary scan register through the TDI pin, the rising edge of CKDR when SHDR is asserted, then is shifted around the array through each I/O cell in a counter clockwise direction, and finally exits through the TDO pin. Since each I/O pin is designed as a bidirectional pin, a 2–bit shift register resides in each I/O cell, one for monitor either the input or output, and the other to monitor the enable pin of the 3–state output buffer. For every two clock cycles, the data shifts from one I/O site to the other. The boundary scan cell resides in every I/O site with the exception of TDI, TCK, TMS, TRSTB and TDO pins.



MPA1000 Pin Definitions

Table 2–9. MPA1000 Package Pinout Compatibility

Device	FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP	DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP
MPA1016	•	•		
MPA1036	•	•	•	
MPA1064			•	•
MPA1100				•

Table 2–10. Pin Definitions

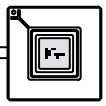
Pin	Definition
5V Int Vdd	Internal array power (V_{DD})
5V Ext Vdd	Pad driver power for I/Os programmed to 5V
3V Ext Vdd	Pad driver power for I/Os programmed to 3V. If no I/Os are programmed to 3V, tie to 5V Ext Vdd. If 3V I/Os are used, connect is a 3.0V or 3.3V supply. These pins must be ≤ 5V Ext Vdd.
Ext Vss	Pad driver V_{SS}
Int Vss	Internal array V_{SS}
I/O	User I/O
I/O Clk	User I/O with optional clock input

MPA1000 Pin Assignments

Pinouts for MPA1016

Pad	Pad Type	Pin Location	
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP
1	5V Int Vdd		1
2	Ext Vss		
3	3V Ext Vdd	12	2
4	5V Ext Vdd		
5	Ext Vss		3
6	I/O (A16)	13	4
7	I/O (A15)	14	5
8	I/O (A14)	15	6
9	I/O (A13)	16	7
10	I/O (A12)	17	8
11	5V Ext Vdd		
12	I/O (A11)	18	9
13	I/O		10
14	I/O (A10)	19	11
15	I/O		12
16	I/O Clk	20	13
17	Int Vss	21	14
18	I/O Clk	22	15
19	I/O (A9)	23	16
20	I/O (A8)	24	17
21	I/O		18
22	I/O (A7)	25	19
23	Ext Vss	26	20
24	I/O		21
25	I/O (A6)	27	22
26	I/O		23
27	I/O (A5)	28	24
28	I/O		25
29	F[4]	29	26
30	Int Vss		27
31	F[3]	30	28
32	Ext Vss		
33	F[2]	31	29
34	5V Ext Vdd		30
35	F[0]	32	31
36	3V Ext Vdd		32

Pad	Pad Type	Pin Location	
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP
37	Ext Vss		
38	Ext Vss		33
39	5V Ext Vdd		
40	RESET	33	34
41	3V Ext Vdd		35
42	F[1]	34	36
43	I/O (A4)	35	37
44	I/O (A3)	36	38
45	I/O (A2)	37	39
46	I/O (A1)	38	40
47	I/O (A0)	39	41
48	5V Ext Vdd	40	42
49	I/O		43
50	I/O		44
51	I/O (D7)	41	45
52	I/O		46
53	I/O Clk	42	47
54	5V Int Vdd	43	48
55	I/O Clk	44	49
56	I/O		50
57	I/O		51
58	I/O (D6)	45	52
59	I/O		53
60	Ext Vss	46	54
61	I/O (D5)	47	55
62	I/O (D4)	48	56
63	I/O (D3)	49	57
64	I/O (D2)	50	58
65	I/O (D1)	51	59
66	MODE[0]	52	60
67	Ext Vss		61
68	MODE[1]	53	62
69	3V Ext Vdd		
70	5V Ext Vdd		63
71	Probe Pad	NOT BONDED	
72	Ext Vss		66



Pinouts for MPA1016 (continued)

Pad	Pad Type	Pin Location	
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP
73	5V Ext Vdd		
74	MODE[2]	54	67
75	5V Int Vdd		
76	MODE[3]	55	68
77	Vpp		
78	Clk	56	69
79	3V Ext Vdd	57	70
80	Ext Vss		71
81	I/O (DCLK)	58	72
82	I/O		73
83	I/O (D0)	59	74
84	I/O		75
85	I/O (TDO)	60	76
86	Ext Vss		77
87	I/O (TDI)	61	78
88	I/O (TMS)	62	79
89	I/O		80
90	I/O (TRSTB)	63	81
91	I/O Clk	64	82
92	Int Vss	65	83
93	I/O Clk	66	84
94	I/O		85
95	I/O	67	86
96	I/O		87
97	I/O	68	88
98	Ext Vss		89
99	I/O (TCK)	69	90
100	I/O	70	91
101	I/O	71	92
102	I/O	72	93
103	I/O	73	94
104	5V Ext Vdd		
105	3V Ext Vdd		
106	Ext Vss	74	95
107	5V Ext Vdd		
108	5V Int Vdd		96
109	Int Vss		97
110	Ext Vss		

Pad	Pad Type	Pin Location	
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP
111	5V Ext Vdd	75	98
112	3V Ext Vdd		99
113	Ext Vss		
114	I/O	76	100
115	I/O	77	101
116	I/O	78	102
117	I/O	79	103
118	I/O	80	104
119	Ext Vss		105
120	I/O	81	106
121	I/O	82	107
122	I/O	83	108
123	I/O		109
124	I/O Clk	84	110
125	5V Int Vdd	1	111
126	I/O Clk	2	112
127	I/O		113
128	I/O	3	114
129	I/O	4	115
130	I/O	5	116
131	5V Ext Vdd		117
132	I/O	6	118
133	I/O	7	119
134	I/O	8	120
135	I/O	9	121
136	I/O (A17)	10	122
137	Ext Vss		123
138	5V Ext Vdd		124
139	Ext Vss	11	125
140	3V Ext Vdd		126
141	Int Vss		127
	NC		64,65,128

Pinouts for MPA1036

Pad	Pad Type	Pin Location			
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP	DH Suffix 160–Pin QFP	HI Suffix 181–Pin PGA
1	5V Int Vdd		1		A2
2	Ext Vss				VSSE
3	3V Ext Vdd	12	2	40	A1
4	5V Ext Vdd				Vdd
5	Ext Vss		3		VSSE
6	I/O (A16)	13	4	39	B2
7	I/O			38	C2
8	I/O (A15)	14	5	37	D4
9	I/O			36	B1
10	I/O (A14)	15	6	35	C3
11	5V Ext Vdd				VDD
12	I/O (A13)	16	7	34	D3
13	I/O			33	C1
14	I/O (A12)	17	8	32	D2
15	I/O			31	D1
16	I/O (A11)	18	9	30	E3
17	Ext Vss			29	VSSE
18	I/O		10	28	F3
19	I/O (A10)	19	11	27	E1
20	I/O			26	E2
21	I/O		12	25	F1
22	I/O Clk	20	13	24	G3
23	5V Int Vdd			23	G1
24	Int Vss	21	14	22	VSSI
25	I/O Clk	22	15	21	G2
26	I/O			20	F2
27	I/O			19	H1
28	I/O (A9)	23	16	18	H3
29	I/O			17	H2
30	5V Ext Vdd			16	VDD
31	I/O (A8)	24	17	15	J1
32	I/O		18	14	J2
33	I/O (A7)	25	19	13	K1
34	I/O			12	K2
35	I/O			11	L1
36	Ext Vss	26	20	10	VSSE
37	I/O		21	9	M1
38	I/O (A6)	27	22	8	L2

Pad	Pad Type	Pin Location			
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP	DH Suffix 160–Pin QFP	HI Suffix 181–Pin PGA
39	I/O		23	7	N1
40	I/O (A5)	28	24	6	J3
41	I/O		25	5	P1
42	F[4]	29	26	4	K3
43	Int Vss		27		VSSI
44	F[3]	30	28	3	M2
45	Ext Vss				VSSE
46	F[2]	31	29	2	L3
47	5V Ext Vdd		30		VDD
48	F[0]	32	31	1	M3
49	3V Ext Vdd		32		P2
50	Ext Vss				VSSE
51	Ext Vss				VSSE
52	Ext Vss		33		VSSE
53	5V Ext Vdd				VDD
54	$\overline{\text{RESET}}$	33	34	160	R1
55	3V Ext Vdd		35		N2
56	F[1]	34	36	159	R2
57	I/O (A4)	35	37	158	N3
58	I/O			157	R3
59	I/O (A3)	36	38	156	N4
60	I/O			155	R4
61	I/O (A2)	37	39	154	P3
62	Ext Vss			153	VSSE
63	I/O			152	N5
64	I/O (A1)	38	40	151	R5
65	I/O			150	P4
66	I/O (A0)	39	41	149	R6
67	I/O			148	N6
68	5V Ext Vdd	40	42	147	VDD
69	I/O		43	146	P5
70	I/O		44	145	R7
71	I/O (D7)	41	45	144	N7
72	I/O		46	143	R8
73	I/O Clk	42	47	142	N8
74	Int Vss			141	VSSI
75	5V Int Vdd	43	48	140	P6
76	I/O Clk	44	49	139	P8

Pinouts for MPA1036 (continued)

Pad	Pad Type	Pin Location			
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP	DH Suffix 160–Pin QFP	HI Suffix 181–Pin PGA
77	I/O		50	138	P7
78	I/O		51	137	R9
79	I/O (D6)	45	52	136	P9
80	I/O		53	135	R10
81	Ext Vss	46	54	134	VSSE
82	I/O (D5)	47	55	133	R11
83	I/O			132	N9
84	I/O (D4)	48	56	131	R12
85	I/O			130	P10
86	I/O (D3)	49	57	129	P11
87	5V Ext Vdd				VDD
88	I/O			128	R13
89	I/O (D2)	50	58	127	N10
90	I/O			126	R14
91	I/O (D1)	51	59	125	N11
92	I/O			124	P13
93	MODE[0]	52	60	123	P12
94	Ext Vss		61		VSSE
95	MODE[1]	53	62	122	N12
96	3V Ext Vdd			121	P14
97	5V Ext Vdd		63		VDD
98	Probe Pad	NOT BONDED			
99	Ext Vss				VSSE
100	Ext Vss		66		VSSE
101	5V Ext Vdd				VDD
102	MODE[2]	54	67	120	M12
103	5V Int Vdd				R15
104	MODE[3]	55	68	119	N13
105	Vpp				P15
106	Clk	56	69	118	L13
107	3V Ext Vdd	57	70	117	N15
108	Ext Vss		71		VSSE
109	I/O (Dclk)	58	72	116	L14
110	I/O		73	115	M13
111	I/O (D0)	59	74	114	M15
112	I/O		75	113	N14
113	I/O (Tdo)	60	76	112	K14
114	Ext Vss		77	111	VSSE

Pad	Pad Type	Pin Location			
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP	DH Suffix 160–Pin QFP	HI Suffix 181–Pin PGA
115	I/O (Tdi)	61	78	110	L15
116	I/O			109	K13
117	I/O			108	K15
118	I/O (Tms)	62	79	107	M14
119	I/O			106	J15
120	5V Ext Vdd			105	VDD
121	I/O		80	104	H14
122	I/O (Trstb)	63	81	103	J13
123	I/O			102	H15
124	I/O			101	J14
125	I/O Clk	64	82	100	G14
126	Int Vss	65	83	99	VSSI
127	5V Int Vdd			98	G15
128	I/O Clk	66	84	97	H13
129	I/O		85	96	F15
130	I/O	67	86	95	G13
131	I/O		87	94	E15
132	I/O	68	88	93	F14
133	Ext Vss		89	92	VSSE
134	I/O (Tck)	69	90	91	F13
135	I/O			90	D15
136	I/O	70		89	E14
137	I/O		91	88	C15
138	I/O	71		87	E13
139	5V Ext Vdd				VDD
140	I/O		92	86	D13
141	I/O	72		85	D14
142	I/O		93	84	C13
143	I/O			83	B15
144	I/O	73	94	82	D12
145	Ext Vss				VSSE
146	3V Ext Vdd				C12
147	Ext Vss	74	95	81	VSSE
148	5V Ext Vdd				VDD
149	5V Int Vdd		96		C14
150	Int Vss		97		VSSI
151	Ext Vss				VSSE
152	5V Ext Vdd	75	98	80	VDD

Pinouts for MPA1036 (continued)

Pad	Pad Type	Pin Location			
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP	DH Suffix 160–Pin QFP	HI Suffix 181–Pin PGA
153	3V Ext Vdd		99		A15
154	Ext Vss			79	VSSE
155	I/O	76	100	78	B14
156	I/O			77	C11
157	I/O	77	101	76	B13
158	I/O			75	B12
159	I/O	78	102	74	A14
160	5V Ext Vdd			73	VDD
161	I/O	79	103	72	A13
162	I/O			71	C10
163	I/O	80		70	A12
164	I/O		104	69	B11
165	I/O	81		68	A11
166	Ext Vss		105	67	VSSE
167	I/O	82	106	66	A10
168	I/O		107	65	B10
169	I/O	83	108	64	A9
170	I/O		109	63	C9
171	I/O Clk	84	110	62	B8
172	5V Int Vdd	1	111	61	B9
173	Int Vss			60	VSSI
174	I/O Clk	2	112	59	C8
175	I/O		113	58	A8
176	I/O	3	114	57	B7
177	I/O		115	56	A7
178	I/O	4	116	55	C7
179	5V Ext Vdd		117		VDD
180	I/O	5		54	B6
181	I/O		118	53	A6
182	I/O	6		52	C6

Pad	Pad Type	Pin Location			
		FN Suffix 84–Pin PLCC	DD Suffix 128–Pin QFP	DH Suffix 160–Pin QFP	HI Suffix 181–Pin PGA
183	I/O			51	A5
184	I/O	7	119	50	B5
185	Ext Vss			49	VSSE
186	I/O	8	120	48	C5
187	I/O			47	A4
188	I/O	9	121	46	B4
189	I/O			45	A3
190	I/O (17)	10	122	44	C4
191	Ext Vss		123	43	VSSE
192	5V Ext Vdd		124	42	VDD
193	Ext Vss	11	125		VSSE
194	3V Ext Vdd		126	41	B3
195	Int Vss		127		VSSI
	NC		64,65, 128		E5

181PGA NOTES:
VSSE: G12, E12, K12, D10, M10, G4, E4, K4, D6, M6
VSSI: E8, L8, H11, M11, H5, D5
VDD: D8, M8, H12, F12, J12, L12, D9, M9, D11, H4, F4, M4, J4, L4, D7, M7, M5

Pinouts for MPA1064

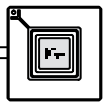
Pad	Pad Type	Pin Location		
		DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP	KE Suffix 224–Pin PGA
1	5V Int Vdd			VDD
2	Ext Vss			VSSE
3	3V Ext Vdd	40		E4
4	5V Ext Vdd			VDD
5	Ext Vss			VSSE
6	I/O (A16)	39	1	C4
7	I/O		2	B2
8	I/O		3	D4
9	I/O	38	4	C2
10	I/O		5	C3
11	Ext Vss		6	VSSE
12	I/O (A15)	37	7	D3
13	I/O		8	B1
14	I/O	36	9	D2
15	I/O		10	C1
16	I/O (A14)	35	11	G4
17	5V Ext Vdd		12	VDD
18	I/O (A13)	34	13	E3
19	I/O	33	14	D1
20	I/O (A12)	32	15	E2
21	I/O	31	16	E1
22	I/O (A11)	30	17	F3
23	Ext Vss	29	18	VSSE
24	I/O	28	19	G3
25	I/O (A10)	27	20	F1
26	I/O	26	21	G2
27	I/O	25	22	G1
28	I/O Clk	24	23	J4
29	5V Int Vdd	23	24	VDD
30	Int Vss	22	25	VSSI
31	I/O Clk	21	26	H1
32	I/O		27	H3
33	I/O	20	28	J2
34	I/O		29	H2
35	I/O	19	30	K1
36	Ext Vss		31	VSSE
37	I/O		32	L1
38	I/O (A9)	18	33	J3

Pad	Pad Type	Pin Location		
		DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP	KE Suffix 224–Pin PGA
39	I/O		34	L2
40	I/O	17	35	K3
41	I/O		36	M1
42	5V Ext Vdd	16	37	VDD
43	I/O (A8)	15	38	N1
44	I/O	14	39	K2
45	I/O (A7)	13	40	P1
46	I/O	12	41	L3
47	I/O	11	42	N2
48	Ext Vss	10	43	VSSE
49	I/O	9	44	R1
50	I/O (A6)	8	45	M3
51	I/O	7	46	T1
52	I/O (A5)	6	47	L4
53	I/O	5	48	R2
54	F[4]	4	49	N3
55	Int Vss			VSSI
56	F[3]	3	50	P2
57	Ext Vss			VSSE
58	F[2]	2	51	P3
59	5V Ext Vdd			VDD
60	F[0]	1	52	P4
61	3V Ext Vdd			N4
62	Ext Vss			VSSE
63	Ext Vss		53	VSSE
64	5V Ext Vdd			VDD
65	$\overline{\text{RESET}}$	160	54	R3
66	3V Ext Vdd		55	P5
67	F[1]	159	56	T2
68	I/O (A4)	158	57	R4
69	I/O		58	T3
70	I/O	157	59	P6
71	I/O		60	U2
72	I/O (A3)	156	61	T4
73	Ext Vss			VSSE
74	I/O		62	R5
75	I/O	155	63	U3
76	I/O		64	T5

Pinouts for MPA1064 (continued)

Pad	Pad Type	Pin Location		
		DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP	KE Suffix 224–Pin PGA
77	I/O (A2)	154	65	U4
78	I/O		66	P7
79	Ext Vss	153	67	VSSE
80	I/O	152	68	R6
81	I/O (A1)	151	69	U5
82	I/O	150	70	R7
83	I/O (A0)	149	71	U6
84	I/O	148	72	P8
85	5V Ext Vdd	147	73	VDD
86	I/O	146	74	T7
87	I/O	145	75	U7
88	I/O (D7)	144	76	R8
89	I/O	143	77	U8
90	I/O Clk	142	78	T8
91	Int Vss	141	79	VSSI
92	5V Int Vdd	140	80	VDD
93	I/O Clk	139	81	T9
94	I/O	138	82	R9
95	I/O	137	83	U10
96	I/O (D6)	136	84	R10
97	I/O	135	85	T10
98	Ext Vss	134	86	VSSE
99	I/O (D5)	133	87	U11
100	I/O	132	88	P10
101	I/O (D4)	131	89	T11
102	I/O	130	90	R11
103	I/O (D3)	129	91	U12
104	5V Ext Vdd			VDD
105	I/O	128	92	U13
106	I/O		93	P11
107	I/O (D2)	127	94	U14
108	I/O		95	R12
109	I/O	126	96	T13
110	Ext Vss		97	VSSE
111	I/O		98	U15
112	I/O (D1)	125	99	R13
113	I/O		100	U16
114	I/O	124	101	T14

Pad	Pad Type	Pin Location		
		DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP	KE Suffix 224–Pin PGA
115	I/O		102	T15
116	MODE[0]	123	103	R14
117	Ext Vss			VSSE
118	MODE[1]	122	104	R15
119	3V Ext Vdd	121		P12
120	5V Ext Vdd			VDD
121	Probe Pad	NOT BONDED		
122	Ext Vss			VSSE
123	5V Ext Vdd		105	VDD
124	MODE[2]	120	106	T16
125	5V Int Vdd			P13
126	MODE[3]	119	107	T17
127	Vpp			P14
128	Clk	118	108	P16
129	3V Ext Vdd	117	109	N14
130	Ext Vss		110	VSSE
131	I/O (DCLK)	116	111	R16
132	I/O		112	R17
133	I/O	115	113	L14
134	I/O		114	N16
135	I/O (D0)	114	115	P15
136	Ext Vss			VSSE
137	I/O	113	116	N15
138	I/O		117	P17
139	I/O		118	M15
140	I/O		119	N17
141	I/O (TDO)	112	120	L15
142	Ext Vss	111	121	VSSE
143	I/O (TDI)	110	122	L16
144	I/O	109	123	M17
145	I/O	108	124	K15
146	I/O (TMS)	107	125	L17
147	I/O	106	126	K16
148	5V Ext Vdd	105	127	VDD
149	I/O	104	128	J15
150	I/O (TRSTB)	103	129	K17
151	I/O	102	130	J14
152	I/O	101	131	J16



Pinouts for MPA1064 (continued)

Pad	Pad Type	Pin Location		
		DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP	KE Suffix 224–Pin PGA
153	I/O Clk	100	132	H16
154	Int Vss	99	133	VSSI
155	5V Int Vdd	98	134	VDD
156	I/O Clk	97	135	H17
157	I/O	96	136	H15
158	I/O	95	137	G17
159	I/O	94	138	G16
160	I/O	93	139	F17
161	Ext Vss	92	140	VSSE
162	I/O		141	E17
163	I/O		142	G15
164	I/O (TCK)	91	143	D17
165	I/O	90	144	F15
166	I/O	89	145	C17
167	5V Ext Vdd		146	VDD
168	I/O	88	147	E16
169	I/O		148	G14
170	I/O	87	149	D16
171	I/O	86	150	E15
172	I/O	85	151	B17
173	Ext Vss			VSSE
174	I/O	84	152	C16
175	I/O		153	D15
176	I/O	83	154	B16
177	I/O		155	D14
178	I/O	82	156	C15
179	Ext Vss			VSSE
180	3V Ext Vdd			E14
181	Ext Vss	81		VSSE
182	5V Ext Vdd			VDD
183	5V Int Vdd			VDD
184	Int Vss			VSSI
185	Ext Vss		157	VSSE
186	5V Ext Vdd	80		VDD
187	3V Ext Vdd		158	D13
188	Ext Vss	79		VSSE
189	I/O		159	C14
190	I/O	78	160	B15

Pad	Pad Type	Pin Location		
		DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP	KE Suffix 224–Pin PGA
191	I/O		161	D12
192	I/O	77	162	A16
193	I/O	76	163	C13
194	Ext Vss		164	VSSE
195	I/O		165	B13
196	I/O	75	166	B14
197	I/O		167	D11
198	I/O	74	168	A15
199	I/O		169	C12
200	5V Ext Vdd	73	170	VDD
201	I/O	72	171	C11
202	I/O	71	172	A14
203	I/O	70	173	B11
204	I/O	69	174	A13
205	I/O	68	175	C10
206	Ext Vss	67	176	VSSE
207	I/O	66	177	B10
208	I/O	65	178	A12
209	I/O	64	179	C9
210	I/O	63	180	A11
211	I/O Clk	62	181	D9
212	5V Int Vdd	61	182	VDD
213	Int Vss	60	183	VSSI
214	I/O Clk	59	184	A10
215	I/O	58	185	B9
216	I/O	57	186	A8
217	I/O	56	187	B8
218	I/O	55	188	A7
219	5V Ext Vdd		189	VDD
220	I/O	54	190	B7
221	I/O	53	191	C8
222	I/O	52	192	A6
223	I/O	51	193	C7
224	I/O	50	194	A5
225	Ext Vss	49	195	VSSE
226	I/O		196	A4
227	I/O	48	197	D7
228	I/O		198	B5

Pinouts for MPA1064 (continued)

Pad	Pad Type	Pin Location		
		DH Suffix 160–Pin QFP	DK Suffix 208–Pin QFP	KE Suffix 224–Pin PGA
229	I/O	47	199	C6
230	I/O	46	200	A3
231	Ext Vss			VSSE
232	I/O		201	B4
233	I/O	45	202	D6
234	I/O		203	A2
235	I/O (A17)	44	204	C5
236	I/O		205	B3
237	Ext Vss	43		VSSE
238	5V Ext Vdd	42		VDD
239	Ext Vss		206	VSSE
240	3V Ext Vdd	41	207	D5
241	Int Vss		208	VSSI

NOTES:
VSSI: E8, E10, H5, J13, K5, N9
VSSE: A1, A9, A17, E7, E9, E11, F4, F14, H13, J1, J5, J17, K13, M4, M14, N7, N11, P9, U1, U9, U17
VDDI: D8, D10, K4, K14, N8
VDDE: B6, B12, F2, F16, H4, H14, M2, M16, N10, T6, T12

Pinouts for MPA1100

Pad	Pad Type	Pin Location	
		DK Suffix 208–Pin QFP	HV Suffix 299–Pin PGA
1	5V Int Vdd		VDDI
2	Ext Vss		VSSE
3	3V Ext Vdd		F5
4	5V Ext Vdd		VDDE
5	Ext Vss		VSSE
6	I/O		C2
7	I/O		D3
8	I/O (A16)	1	B1
9	I/O		E3
10	I/O	2	C1
11	Ext Vss		VSSE
12	I/O	3	D1
13	I/O		F3
14	I/O	4	E2
15	I/O		G4
16	I/O	5	E1
17	Ext Vss	6	VSSE
18	I/O (A15)	7	F2
19	I/O	8	H4
20	I/O	9	F1
21	I/O	10	H3
22	I/O (A14)	11	G2
23	5V Ext Vdd	12	VDDE
24	I/O (A13)	13	G1
25	I/O	14	J4
26	I/O (A12)	15	H2
27	I/O	16	J3
28	I/O (A11)	17	H1
29	Ext Vss	18	VSSE
30	I/O	19	J1
31	I/O (A10)	20	K4
32	I/O	21	K2
33	I/O	22	K3
34	I/O Clk	23	K1
35	5V Int Vdd	24	VDDI
36	Int Vss	25	VSSI
37	I/O Clk	26	L3
38	I/O	27	L1

Pad	Pad Type	Pin Location	
		DK Suffix 208–Pin QFP	HV Suffix 299–Pin PGA
39	I/O	28	L4
40	I/O	29	L2
41	I/O	30	M2
42	Ext Vss	31	VSSE
43	I/O	32	M3
44	I/O (A9)	33	M1
45	I/O	34	M4
46	I/O	35	N1
47	I/O	36	N2
48	5V Ext Vdd	37	VDDE
49	I/O (A8)	38	N3
50	I/O	39	P1
51	I/O (A7)	40	N4
52	I/O	41	P2
53	I/O	42	P3
54	Ext Vss	43	VSSE
55	I/O	44	P4
56	I/O		R1
57	I/O (A6)	45	R3
58	I/O		R2
59	I/O	46	R4
60	Ext Vss		VSSE
61	I/O		T3
62	I/O (A5)	47	T1
63	I/O		T4
64	I/O	48	T2
65	I/O		U3
66	F[4]	49	U1
67	Int Vss		VSSI
68	F[3]	50	V1
69	Ext Vss		VSSE
70	F[2]	51	W1
71	5V Ext Vdd		VDDE
72	F[0]	52	V2
73	3V Ext Vdd		R5
74	Ext Vss		VSSE
75	Ext Vss	53	VSSE
76	5V Ext Vdd		VDDE

Pinouts for MPA1100 (continued)

Pad	Pad Type	Pin Location	
		DK Suffix 208–Pin QFP	HV Suffix 299–Pin PGA
77	RESET	54	V3
78	3V Ext Vdd	55	T6
79	F[1]	56	U5
80	I/O		W2
81	I/O		V5
82	I/O (A4)	57	Y2
83	I/O		V6
84	I/O	58	Y3
85	Ext Vss		VSSE
86	I/O	59	Y4
87	I/O	60	U7
88	I/O		W5
89	I/O (A3)	61	V7
90	I/O		Y5
91	Ext Vss		VSSE
92	I/O	62	Y6
93	I/O	63	U8
94	I/O	64	W7
95	I/O (A2)	65	V8
96	I/O	66	Y7
97	Ext Vss	67	VSSE
98	I/O	68	W8
99	I/O (A1)	69	U9
100	I/O	70	Y8
101	I/O (A0)	71	V9
102	I/O	72	W9
103	5V Ext Vdd	73	VDDE
104	I/O	74	Y9
105	I/O	75	U10
106	I/O (D7)	76	W10
107	I/O	77	V10
108	I/O Clk	78	Y10
109	Int Vss	79	VSSI
110	5V Int Vdd	80	VDDI
111	I/O Clk	81	V11
112	I/O	82	Y11
113	I/O	83	U11
114	I/O (D6)	84	W11

Pad	Pad Type	Pin Location	
		DK Suffix 208–Pin QFP	HV Suffix 299–Pin PGA
115	I/O	85	W12
116	Ext Vss	86	VSSE
117	I/O (D5)	87	V12
118	I/O	88	Y12
119	I/O (D4)	89	U12
120	I/O	90	Y13
121	I/O (D3)	91	W13
122	5V Ext Vdd		VDDE
123	I/O	92	V13
124	I/O	93	Y14
125	I/O (D2)	94	U13
126	I/O	95	W14
127	I/O	96	V14
128	Ext Vss	97	VSSE
129	I/O	98	U14
130	I/O		Y15
131	I/O (D1)	99	V15
132	I/O		Y16
133	I/O	100	U15
134	Ext Vss		VSSE
135	I/O		V16
136	I/O	101	Y17
137	I/O		V17
138	I/O	102	Y18
139	I/O		V18
140	MODE[0]	103	Y19
141	Ext Vss		VSSE
142	MODE[1]	104	W19
143	3V Ext Vdd		T16
144	5V Ext Vdd		VDDE
145	Probe Pad	NOT BONDED	
146	Ext Vss		VSSE
147	5V Ext Vdd	105	VDDE
148	MODE[2]	106	V19
149	5V Int Vdd		U17
150	MODE[3]	107	W20
151	Vpp		U18
152	Clk	108	V20

Pinouts for MPA1100 (continued)

Pad	Pad Type	Pin Location	
		DK Suffix 208–Pin QFP	HV Suffix 299–Pin PGA
153	3V Ext Vdd	109	R16
154	Ext Vss	110	VSSE
155	I/O		T17
156	I/O		U20
157	I/O (DCLK)	111	T18
158	I/O		T19
159	I/O	112	R17
160	Ext Vss		VSSE
161	I/O	113	R18
162	I/O	114	T20
163	I/O		R19
164	I/O (D0)	115	R20
165	I/O		P17
166	Ext Vss		VSSE
167	I/O	116	P18
168	I/O	117	P19
169	I/O	118	N17
170	I/O	119	P20
171	I/O (TDO)	120	N18
172	Ext Vss	121	VSSE
173	I/O (TDI)	122	N19
174	I/O	123	N20
175	I/O	124	M17
176	I/O (TMS)	125	M20
177	I/O	126	M18
178	5V Ext Vdd	127	VDDE
179	I/O	128	M19
180	I/O (TRSTB)	129	L19
181	I/O	130	L17
182	I/O	131	L20
183	I/O Clk	132	L18
184	Int Vss	133	VSSI
185	5V Int Vdd	134	VDDI
186	I/O Clk	135	K20
187	I/O	136	K18
188	I/O	137	K19
189	I/O	138	K17
190	I/O	139	J20

Pad	Pad Type	Pin Location	
		DK Suffix 208–Pin QFP	HV Suffix 299–Pin PGA
191	Ext Vss	140	VSSE
192	I/O	141	H20
193	I/O	142	J18
194	I/O (TCK)	143	H19
195	I/O	144	J17
196	I/O	145	G20
197	5V Ext Vdd	146	VDDE
198	I/O	147	G19
199	I/O	148	H18
200	I/O	149	F20
201	I/O	150	H17
202	I/O	151	F19
203	Ext Vss		VSSE
204	I/O	152	E20
205	I/O		G17
206	I/O	153	E19
207	I/O		F18
208	I/O	154	D20
209	Ext Vss		VSSE
210	I/O	155	C20
211	I/O		E18
212	I/O		B20
213	I/O	156	D18
214	I/O		C19
215	Ext Vss		VSSE
216	3V Ext Vdd		F16
217	Ext Vss		VSSE
218	5V Ext Vdd		VDDE
219	5V Int Vdd		VDDI
220	Int Vss		VSSI
221	Ext Vss	157	VSSE
222	5V Ext Vdd		VDDE
223	3V Ext Vdd	158	E15
224	Ext Vss		VSSE
225	I/O		C18
226	I/O	159	B19
227	I/O		C17
228	I/O	160	A19

Pinouts for MPA1100 (continued)

Pad	Pad Type	Pin Location	
		DK Suffix 208–Pin QFP	HV Suffix 299–Pin PGA
229	I/O		C16
230	Ext Vss		VSSE
231	I/O	161	D15
232	I/O		A18
233	I/O	162	C15
234	I/O		A17
235	I/O	163	D14
236	Ext Vss	164	VSSE
237	I/O	165	C14
238	I/O	166	A16
239	I/O	167	D13
240	I/O	168	A15
241	I/O	169	C13
242	5V Ext Vdd	170	VDDE
243	I/O	171	B13
244	I/O	172	A14
245	I/O	173	D12
246	I/O	174	A13
247	I/O	175	C12
248	Ext Vss	176	VSSE
249	I/O	177	B12
250	I/O	178	A12
251	I/O	179	D11
252	I/O	180	A11
253	I/O Clk	181	C11
254	5V Int Vdd	182	VDDI
255	Int Vss	183	VSSI
256	I/O Clk	184	A10
257	I/O	185	C10
258	I/O	186	B10
259	I/O	187	D10
260	I/O	188	A9
261	5V Ext Vdd	189	VDDE
262	I/O	190	B9
263	I/O	191	C9
264	I/O	192	A8
265	I/O	193	D9
266	I/O	194	B8
267	Ext Vss	195	VSSE
268	I/O	196	A7
269	I/O	197	C8
270	I/O	198	B7
271	I/O	199	D8
272	I/O	200	A6
273	5V Ext Vdd		VDDE
274	I/O	201	A5
275	I/O		C7
276	I/O	202	B5
277	I/O		C6
278	I/O	203	A4
279	Ext Vss		VSSE
280	I/O		A3
281	I/O (A17)	204	C5
282	I/O		A2
283	I/O		C3
284	I/O	205	B2
285	Ext Vss		VSSE
286	5V Ext Vdd		VDDE
287	Ext Vss	206	VSSE
288	3V Ext Vdd	207	E6
289	Int Vss	208	VSSI

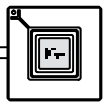
NOTES:

VSSE: A1, A20, B3, B6, B14, B16, B18, C4, D2, D5, D7, D16, D19, E4, E8, E13, E17, G3, G18, H5, H16, J2, J19, N5, N16, T5, T8, T13, U2, U6, U16, U19, V4, W3, W16, W18, Y1, Y20

VSSI: E10, E12, J16, K5, L16, M5, T10, T12

VDDE: B4, B11, B15, B17, D4, D6, D17, E5, E7, E14, E16, F4, F17, G5, G16, P5, P16, T7, T14, U4, W4, W6, W15, W17

VDDI: E9, E11, J5, K16, L5, M16, T9, T11



MPA1000 Electrical Specifications

Absolute Maximum Ratings*

Symbol	Parameter	Min	Max	Unit
V_{dd}, V_{ddo}	DC Supply Voltage	–0.5	6.5	V
V_{out}	DC Output Voltage	–0.5	V_{DD} + 0.5	V
V_{in}	DC Input Voltage	–0.5	V_{DD} + 0.5	V
I	DC Current Drain per Pin, Any Single Input or Output		50	mA
T_A	Commercial Operating Temperature Range (In Free Air)	0	70	°C
T_{stg}	Storage Temperature Range	–65	150	°C

* Maximum Ratings are those values beyond which damage to the device may occur. Functional operation should be restricted to the Recommended Operating Conditions. This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

Recommended Operating Conditions

Symbol	Parameter		Min	Max	Unit
V_{dd}	DC Supply Voltage (Note 1)		4.75	5.25	V
V_{ddo}	Output Supply Voltage	5V Output Supply (5V ext) 3V Output Supply, No I/Os Programmed to 3V 3V Output Supply, 1 or more I/Os Programmed to 3V	4.75 4.75 3.0	5.25 5.25 3.6	V
V_{IH}	High Level Input Voltage	TTL CMOS	2.0 70	V_{dd} 100	V %V_{ddo}
V_{IL}	Low Level Input Voltage	TTL CMOS	0 0	0.8 20	V %V_{ddo}

1. Unused inputs must always be tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{dd})

DC Electrical Characteristics

Symbol	Parameter		Min	Max	Unit
V_{OH}	High Level Output Voltage	I_{OH} = 6 or 12mA; V_{ddo} = 4.75V (V_{ddo}min 5V) I_{OH} = 4 or 8mA; V_{ddo} = 3.0V (V_{ddo}min 3V)	2.4	V_{ddo}	V
V_{OL}	Low Level Output Voltage	I_{OL} = 6 or 12mA; V_{ddo} = Max		0.4	V
I_{IL}	Leakage Current		–10	10	µA
I_{PU}	Pad Pull–Up (When Selected) at V_{in} = 0V		20	200	µA
I_{PD}	Pad Pull–Down (When Selected) at V_{in} = V_{ddo}		40	110	µA
C_{IN}	Input Capacitance (Sample Tested)	PGA Packages Plastic Packages		15 10	pF

MPA1000 Cell AC Characteristics

Symbol	Parameter	Min	Typ[1]	Max	Unit
T_{cn}	Core Cell Delay–AND		0.7		ns
T_{cx}	Core Cell Delay–XOR		1.3		ns
T_{dsu}	DFF Data Setup		1.5		ns
T_{dho}	DFF Data Hold		0		ns
T_{cq}	Clock to Q Delay		0.6		ns
T_{ip}	Pad Input Delay		2.0		ns
T_{op}	Pad Output Delay[2]		7.0		ns

1. Typical process, V_{dd} = Min, T = 25°C.
2. 5V, Low Slew Rate, Low Drive, 22pF Load

MPA1000 Interconnect Characteristics

Symbol	Parameter	Min	Typ[1]	Max	Unit
T_{li}	Local Interconnect (1 Active Load)		0.4		ns
T_{mb}	Medium Bus (Same Zone, 1 Active Load)		0.8		ns
T_{mbt}	Medium Bus Turn (Same Zone, 1 Active Load)		0.7		ns
T_{g}	Global Bus (Same Quadrant, 1 Active Load)		1.6		ns
T_{gg}	Global Bus (Adjacent Quadrants, 1 Active Load)		2.6		ns
T_{xt}	X Bus Turn (Fanout of 1)		1.6		ns
T_{pb}	Peripheral Bus (Adjacent I/O Cells)		2.5		ns
T_{iwo}	Internal Wired–OR (Across Device, 4 Active Pull–Ups)		6.0		

1. 5V, Low Slew Rate, Low Drive, 22pF Load. All delays are incremental. For example, cell–to–cell using M bus = 0.7 + 0.8 = 1.5ns.

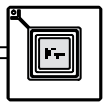
2

MPA1000 Primary Clock Characteristics

Symbol	Parameter	Min	Typ[1]	Max	Unit
T_{ckin}	Primary Clock Pad to Register Delay		5.6		ns
T_{cks}	Primary Clock Skew		1.0		ns
T_{cwh}	Clock High Time		2.0		ns
T_{cwl}	Clock Low Time		2.0		ns

MPA1000 Primary Clock Characteristics

Symbol	Parameter	Min	Typ[1]	Max	Unit
F_{cjtag}	Shift Clock Frequency			16	MHz

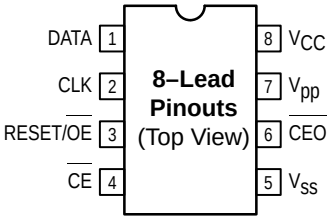


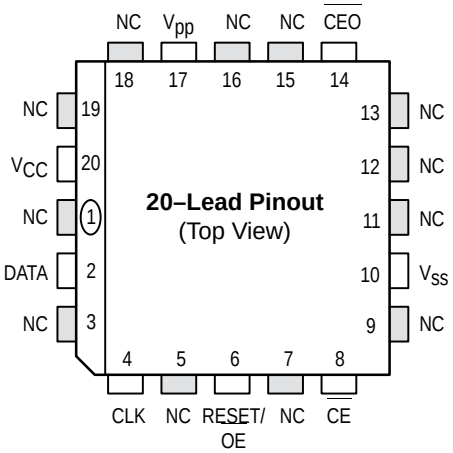
MPA17000 Serial EPROMs

The MPA17128 and MPA1765 are serial OTP EPROMs. They provide a compact, low pin count, non–volatile configuration store for the MPA1000 devices.

MPA17000 devices can be cascaded for increased memory capacity when needed. They are available in the standard 8–pin plastic DIP (P suffix), 8–pin SOIC (D suffix) and 20–pin PLCC (FN suffix) packages.

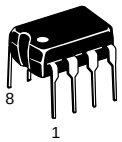
- Configuration EPROM for MPA1000 Devices
- Voltage Range — 4.5 to 6.0V
- Maximum Read Current of 10mA
- Standby Current of 10μA, Typical
- Industry Standard Synchronous Serial Interface
- Full Static Operation
- 10MHz Maximum Clock Rate at 5.0V
- Programmable Polarity on Hardware Reset
- Programs With Industry Standard Programmers
- Electrostatic Discharge Protection > 2000 Volts
- 8–Pin PDIP and SOIC; 20–Pin PLCC Packages
- Commercial (0 to +70°C) and Industrial (–40 to +85°C)





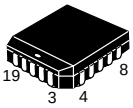
MPA17128
MPA1765

128K, 64K SERIAL EPROM



P SUFFIX
PLASTIC PACKAGE
CASE 626–05

D SUFFIX
PLASTIC SOIC PACKAGE
CASE 751–05



FN SUFFIX
PLCC PACKAGE
CASE 775–02

PIN NAMES

Pins	Function
DATA	Data I/O
CLK	Clock
RESET/OE	Reset Input and Output Enable
CE	Chip Enable Input
VSS	Ground
CEO	Chip Enable Output
VPP	Programming Voltage Supply
VCC	+4.5 to 6.0V Power Supply
NC	Not Connected

MAXIMUM RATINGS*

Parameter	Value	Unit
V_{CC} and Input Voltages W.R.T. V_{SS}	–6.0 to V_{DD} + 0.6	V
V_{PP} Voltage W.R.T. V_{SS} During Programming	–0.6 to +14.0	V
Output Voltage W.R.T. V_{SS}	–0.6 to V_{CC} + 0.6	V
Storage Temperature Range	–65 to +150	°C
Ambient Temperature With Power Applied	–65 to +125	°C
Soldering Temperature of Leads (10 Seconds)	+300	°C
ESD Protection on All Leads	≥2	kV

NOTE: Maximum Ratings are those values beyond which damage to the device may occur. Functional operation should be restricted to the Recommended Operating Conditions.

2

DC CHARACTERISTICS (V_{CC} = 4.5 to 6.0V; Commercial (C) T_A = 0 to +70°C; Industrial (I) T_A = –40 to +85°C)

Symbol	Characteristic		Min	Max	Unit	Condition
V_{IH}	Input Voltage High	DATA, CE, CEO, Reset	2.0	V_{CC}	V	
V_{IL}	Input Voltage Low	DATA, CE, CEO, Reset	–0.3	0.8	V	
V_{OH}	Output Voltage High	DATA, CE, CEO, Reset	3.86 2.40		V	I_{OH} = –4mA; V_{CC} ≥ 4.5V
V_{OL}	Output Voltage Low	DATA, CE, CEO, Reset		0.32	V	I_{OL} = 4.0mA
I_{LI}	Input Leakage Current		–10	10	µA	V_{IN} = 0.1V to V_{CC}
I_{LO}	Output Leakage Current		–10	10	µA	V_{OUT} = 0.1V to V_{CC}
C_{INT}	Internal Capacitance	(All Inputs/Outputs)		10	pF	V_{CC} = 5.0V (Note 1); T_A = 25°C; f_{clk} = 1MHz
I_{CC} Read	Operating Current			10	mA	V_{CC} = 6.0V; CLK = 10MHz
I_{CCS}	Standby Current			500	µA	V_{CC} = 6.0V

1. This parameter is initially characterized and not 100% tested.

Applications Information

DATA

Three–state DATA output for reading and function as the input during programming.

CLOCK

Clock input. Used to increment the internal address and bit counters for reading and programing.

RESET/$\overline{OE}$

Reset and Output Enable input. A Low level both the $\overline{CE}$ and RESET/OE inputs enables the data output driver. A High level on RESET/OE resets both the address and bit counters. In the MPA17128, the logic polarity of this input is programmable as either RESET/OE or OE/RESET. This document describes the pin as RESET/OE although the opposite polarity is also possible, this option is defined and set at device program time.

$\overline{CE}$

Chip Enable input. Used for device selection. A Low level on both CE and OE enables the data output driver. A High level on CE disables both the address and bit counters and forces the device into a low power mode.

$\overline{CEO}$

Chip Enable Out output. This signal is asserted Low on the clock cycle following the last bit read from the memory. It will stay Low as long as CE and OE are both Low. It will then follow CE until OE goes High. Thereafter CEO will stay High until the entire PROM is read again. This pin also used to sense the status of RESET polarity when program mode is entered.

V_{PP}

Programming Voltage Supply. Used to enter programming mode (+10V) and to program the memory (+13V) Must be connected directly to V_{CC} for normal Read operation. No overshoot above +15.5V permitted.

USING THE MPA17000 WITH MPA1000 DEVICES

Connections between the MPA devices and the Serial EPROMs are:

- The DATA output of the MPA17000 drives D0 (data in).
- The CLK input of the MPA17000 is driven by the data clock DCLK output.
- MPA17000s can be cascaded using the $\overline{CEO}$ output to drive the CE input of the next MPA17000.
- For normal Read operations V_{PP} must be connected to V_{CC}. Do not leave V_{PP} open.

The connections between an MPA device and an MPA17000 device are shown in Figure 2–43. The MPA D[0] line is connected to the MPA17000 CLK. At power–up or upon reconfiguration, the MEMCE signal goes Low, enabling the MPA17000 DATA output. During the configuration process, D[0] reads data from the MPA17000 on every rising DCLK edge. The MEMCE signal goes High at the end of configuration and resets the internal address counters of the MPA17000.

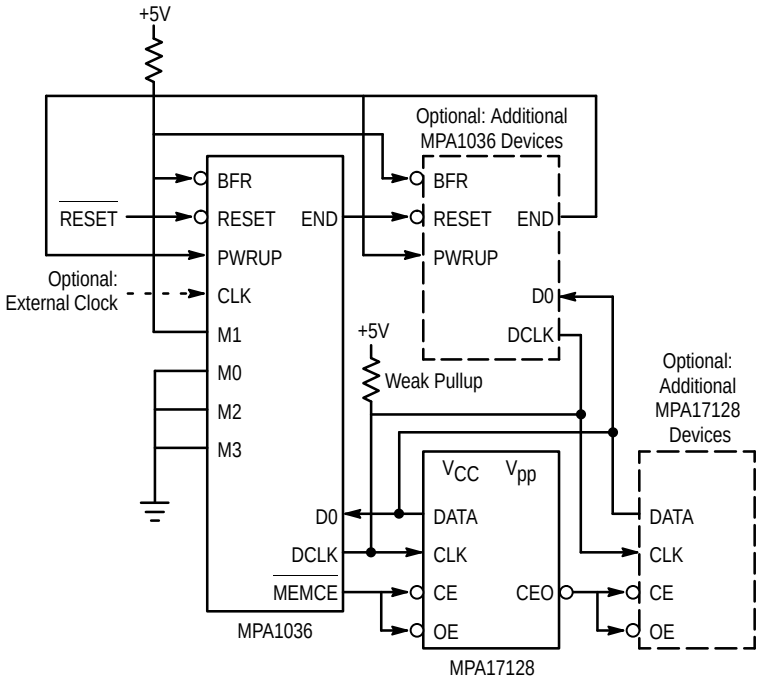


Figure 2–43. MPA1036 Configuration Using MPA17128 Serial EPROM

Cascading Serial Configuration PROMs

Cascading MPA17000s provide additional memory for multiple MPA1000s or for MPA1000s requiring larger configuration memories.

When the last bit from the first MPA17000 is read, the next clock signal to the MPA17000 asserts its CEO output Low and disables its DATA line. The second MPA17000 recognizes the Low level on its CE input and enables its DATA output. (See Figure 2–43).

Additional logic may be required if cascaded memories are so large that the rippled chip enable is not fast enough to activate successive MPA17000s.

STANDBY MODE

The MPA17128 enters a low power standby mode whenever CE is High. In standby mode, the MPA17000 consumes less than 500µA of current. The output will remain in a high impedance state regardless of the state of the OE input.

PROGRAMMING MODE

Programming mode is entered by holding V_{PP} High for at least two clock edges and is exited by removing power from the device or by a Low on both CE and OE. Figure 2–46 through Figure 2–51 shows the programming algorithm.

MPA17128 RESET POLARITY

The MPA17128 lets the user choose the reset polarity as either RESET/OE or OE/RESET. Any third–party commercial programmer should prompt the user for the desired reset polarity.

The programming of the overflow word should be handled transparently by the PROM programmer; it is mentioned here as supplemental information only.

The polarity is programmed into the first overflow word location, max address+1. 00000000 in these locations makes the reset active Low, FFFFFFFF in these locations makes the reset active High. The default condition is RESET active High.

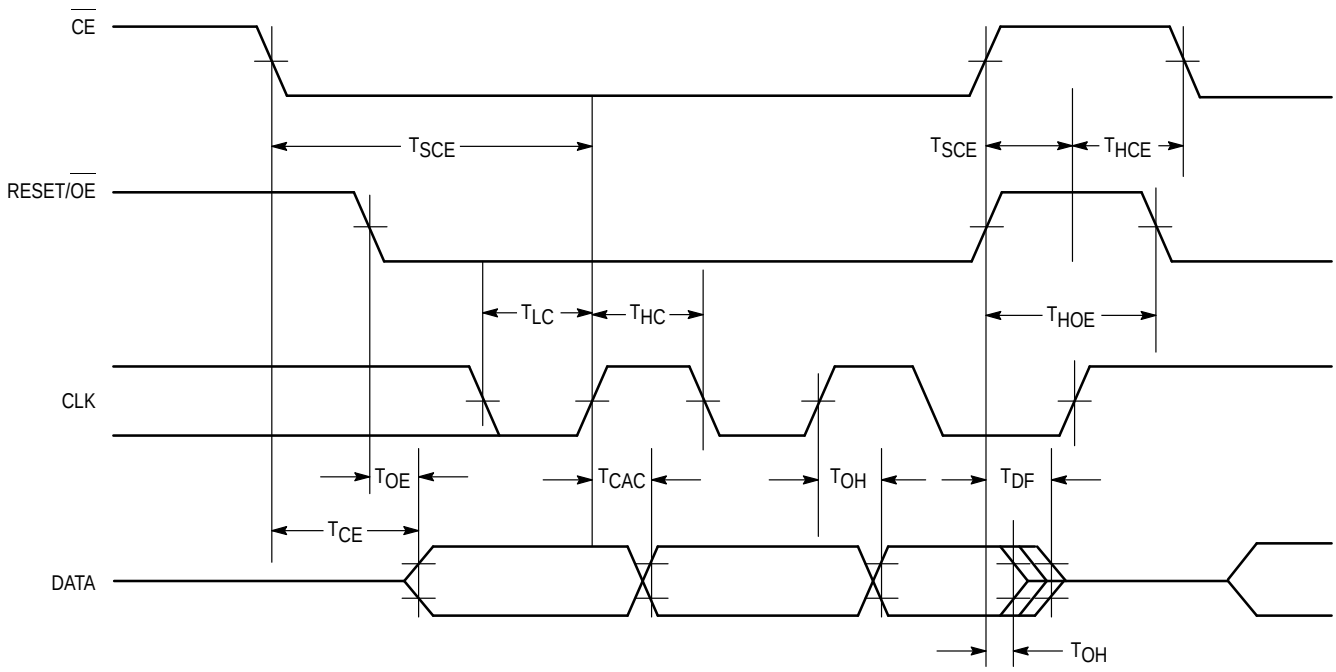


Figure 2–44. AC Characteristics Over Operating Conditions

AC OPERATING CONDITIONS

Symbol	Parameter	Limit 4.5V ≤ V_{CC} ≤ 6.0V		Unit	Condition
		Min	Max		
T_{OE}	$\overline{OE}$ to Data Delay		45	ns	
T_{CE}	$\overline{CE}$ to Data Delay		50	ns	
T_{CAC}	CLK to Data Delay		60	ns	
T_{OH}	Data Hold From $\overline{OE}$, $\overline{CE}$ or CLK	0		ns	
T_{DF}	$\overline{OE}$ or $\overline{CE}$ to Data Float Delay		50	ns	Note 1
T_{LC}	CLK Low Time	25		ns	Note 2
T_{HC}	CLK High Time	25		ns	Note 2
T_{SCE}	$\overline{CE}$ Setup Time to CLK (To Guarantee Proper Counting)	25		ns	
T_{HCE}	$\overline{CE}$ Hold Time to CLK (To Guarantee Proper Counting)	0		ns	Note 2
T_{HOE}	$\overline{OE}$ High Time (Guarantees Counters are Reset)	20		ns	Note 2
CLK_{max}	Clock Frequency		10	MHz	

1. Float delays are measured with minimum tester AC load and maximum DC load.
2. Guarantee by design, not tested.

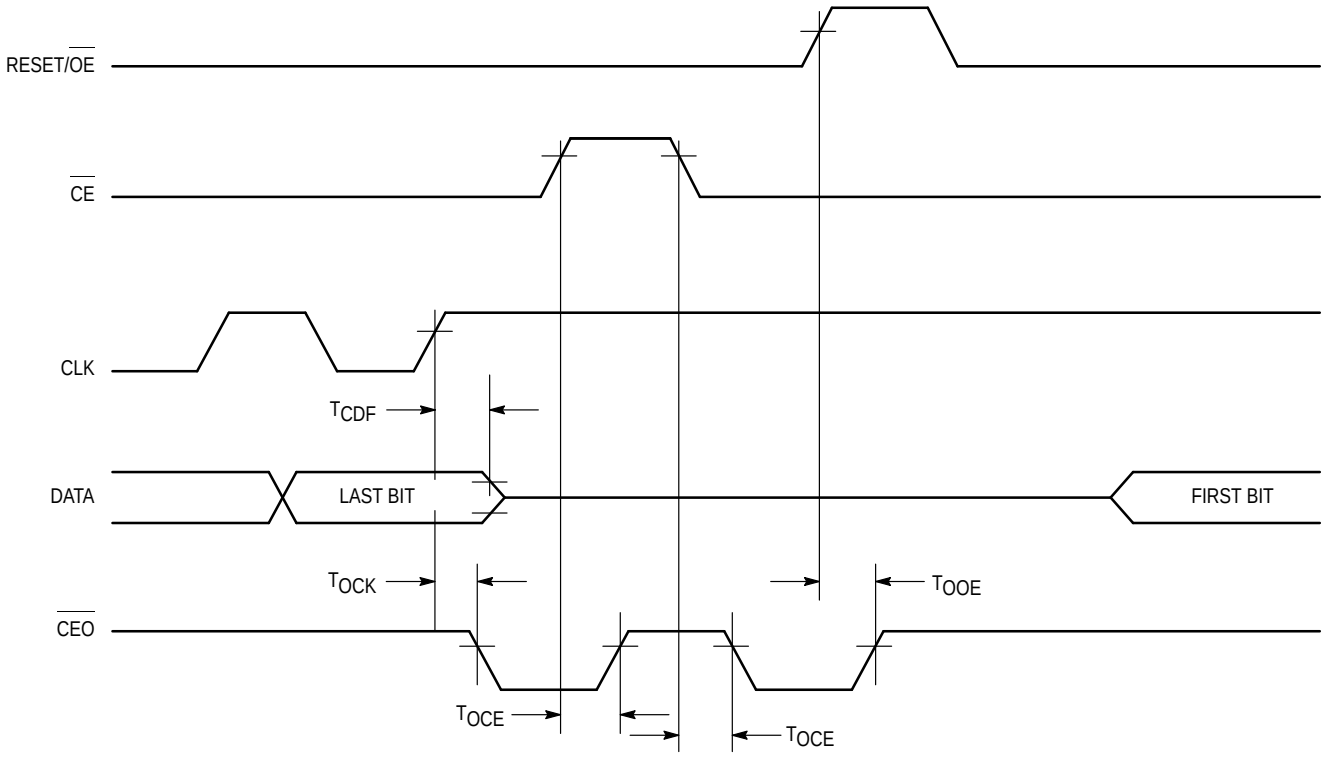


Figure 2–45.

Symbol	Parameter	Limit 4.5V ≤ V_{CC} ≤ 6.0V		Unit	Condition
		Min	Max		
T_{CDF}	CLK to Data Float Delay		50	ns	
T_{OCK}	CLK to $\overline{CEO}$ Delay		40	ns	
T_{OCE}	$\overline{CE}$ to $\overline{CEO}$ Delay		40	ns	
T_{OOE}	RESET/$\overline{OE}$ to $\overline{CEO}$ Delay		40	ns	

PIN ASSIGNMENTS IN THE PROGRAMMING MODE

Pin Name	DIP	PLCC	I/O	Function
DATA	1	2	I/O	The rising edge of the clock shifts a data word in or out of the PROM one bit at a time.
CLK	2	4	I	Clock input. Used to increment the internal address/word counter for reading and programming operation.
RESET/$\overline{OE}$	3	6	I	The rising edge of CLK shifts a data word into the PROM when $\overline{CE}$ and $\overline{OE}$ are High; it shifts a data word out of the PROM when $\overline{CE}$ is Low and $\overline{OE}$ is High. The address/word counter is incremented on the rising edge of CLK while $\overline{CE}$ is held High and $\overline{OE}$ is held Low. Note: Any modified polarity of the RESET/$\overline{OE}$ pin is ignored in the programming mode.
$\overline{CE}$	4	8	I	The rising edge of CLK shifts a data word into the PROM when $\overline{CE}$ and $\overline{OE}$ are High; it shifts a data word out of the PROM when $\overline{CE}$ is Low and $\overline{OE}$ is High. The address/word counter is incremented on the rising edge of CLK while $\overline{CE}$ is held High and $\overline{OE}$ is held Low.
GND	5	10	—	Ground pin.
$\overline{CEO}$	6	14	O	The polarity of the RESET/$\overline{OE}$ pin can be read by sensing the $\overline{CEO}$ pin. Note: The polarity of the RESET/$\overline{OE}$ pin is ignored while in the programming mode. In final verification, this pin must be monitored to go Low one clock cycle after the last data bit has been read.
V_{PP}	7	17	—	Programming Voltage Supply. Programming mode is entered by holding $\overline{CE}$ and $\overline{OE}$ High and V_{PP} at V_{PP1} for two rising clock edges and then lowering V_{PP} to V_{PP2} for one more rising clock edge. A word is programmed by strobing the device with V_{PP} for the duration TPGM V_{PP} must be tied to V_{CC} for normal operation.
V_{CC}	8	20	—	+5 V power supply input.

DC PROGRAMMING SPECIFICATIONS

Symbol	Parameter	Limit		Unit	Condition
		Min	Max		
V_{CCP}	Supply Voltage During Programming	5.0	6.0	V	
V_{IL}	Input Voltage Low	0	0.5	V	
V_{IH}	Input Voltage High	2.4	V_{CC}	V	
V_{OL}	Output Voltage Low		0.4	V	
V_{OH}	Output Voltage High	3.7		V	
V_{PP1}	Programming Voltage	12.5	13.5	V	Note 1
V_{PP2}	Programming Mode Access Voltage	V_{CCP}	V_{CCP} + 1	V	
I_{PPP}	Supply Current in Programming Mode		100	mA	
I_L	Input or Output Leakage Current	–10	10	µA	
V_{CCL}	First Pass Supply Voltage Low for Final Verification	2.8	3.0	V	
V_{CCH}	Second Pass Supply Voltage High for Final Verification	6.0	6.2	V	

1. No overshoot is permitted on this signal. V_{PP} must not be allowed to exceed V_{PP1} max.

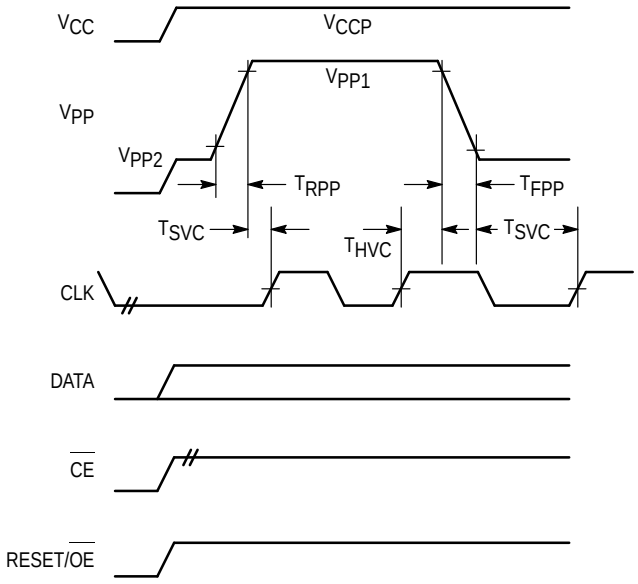


Figure 2–46. Enter Programming Mode

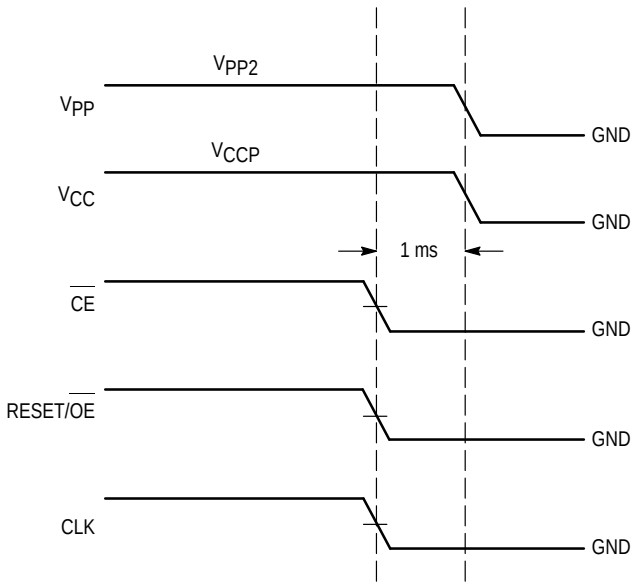


Figure 2–47. Exit Programming Mode

2

AC PROGRAMMING SPECIFICATIONS

Symbol	Parameter	Limit		Unit	Condition
		Min	Max		
T_{RPP}	Rise Time of V_{PP} (10 to 90%)	50		ns	
T_{FPP}	Fall Time of V_{PP} (90 to 10%)	50		ns	
T_{PGM}	V_{PP} Programming Pulse Width	0.95	1.05	ms	
T_{SVC}	V_{PP} Setup to CLK for Entering Programming Mode	100		ns	
T_{HVC}	V_{PP} Hold from CLK for Entering Programming Mode	300		ns	
T_{SDP}	Data Setup to CLK for Programming	50		ns	
T_{HDP}	Data Hold from CLK for Programming	0		ns	
T_{SCC}	$\overline{CE}$ Setup to CLK for Programming/Verifying	100		ns	Note 1
T_{HCC}	$\overline{CE}$ Hold from CLK for Programming/Verifying	200		ns	
T_{SCV}	$\overline{CE}$ Setup to V_{PP} for Programming	100		ns	
T_{HCV}	$\overline{CE}$ Hold from V_{PP} for Programming	50		ns	
T_{SIC}	$\overline{OE}$ Setup to CLK for Incrementing Address Counter	100		ns	
T_{HIC}	$\overline{OE}$ Hold from CLK for Incrementing Address Counter	0		ns	
T_{CAC}	CLK to Data Valid		400	ns	
T_{OH}	Data Hold from CLK	0		ns	
T_{CE}	$\overline{CE}$ Low to Data Valid		250	ns	

1. While in programming mode, CE should only be changed while CLK is High and has been High for 200ns.

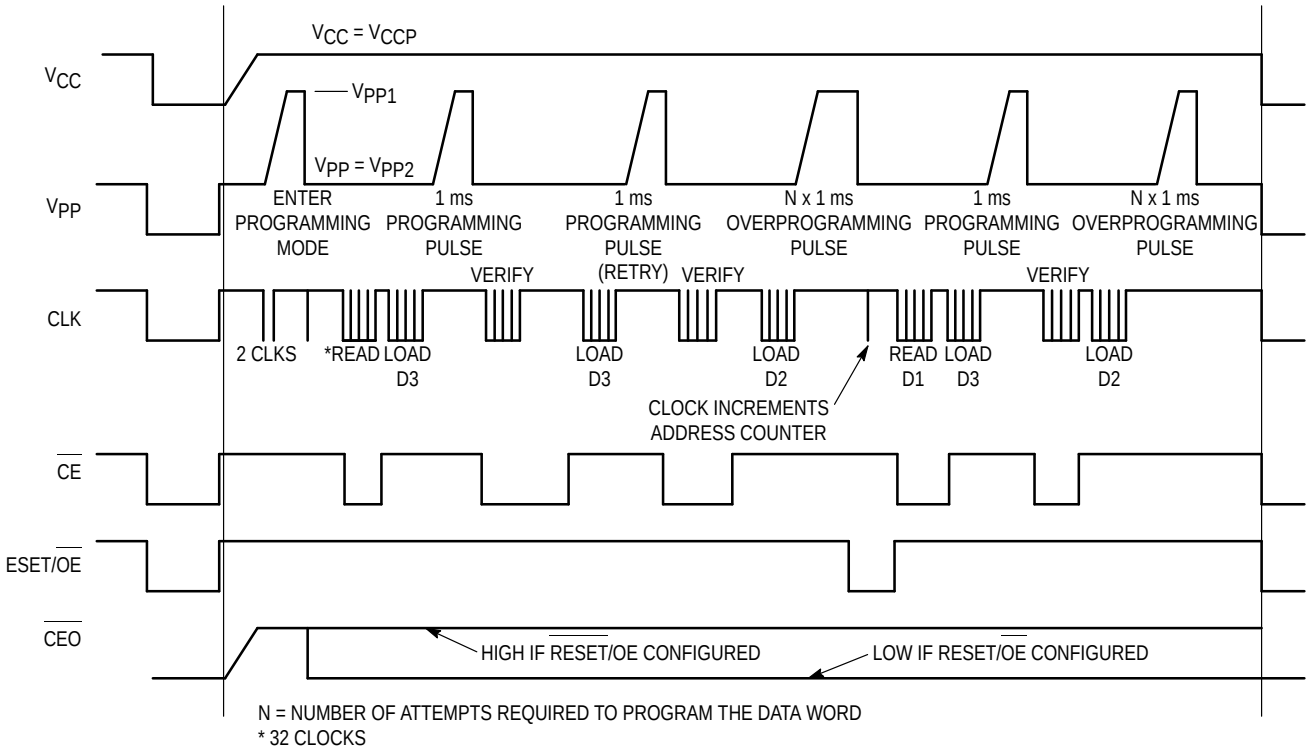


Figure 2–48. Programming Cycle Overview

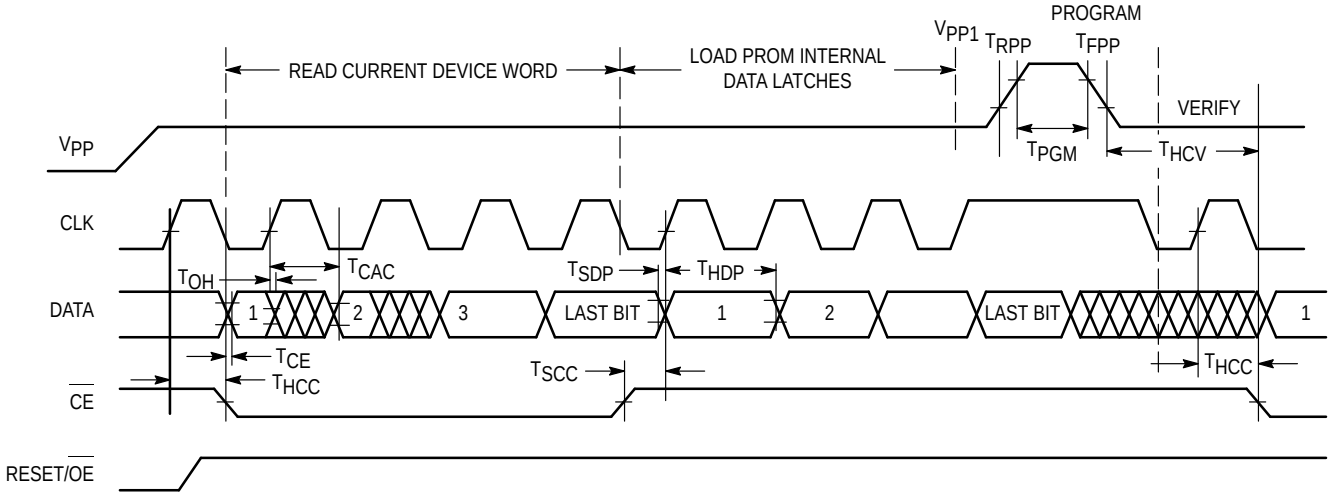


Figure 2–49. Details of Read/Program/Verify Cycle

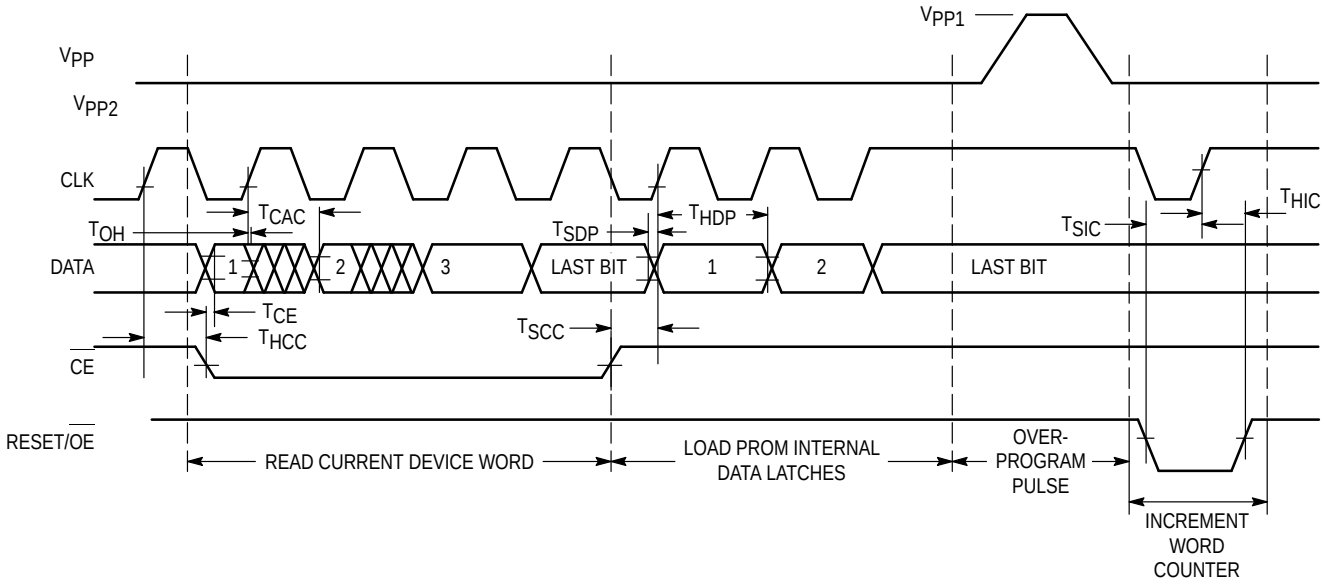


Figure 2–50. Overprogramming Detail

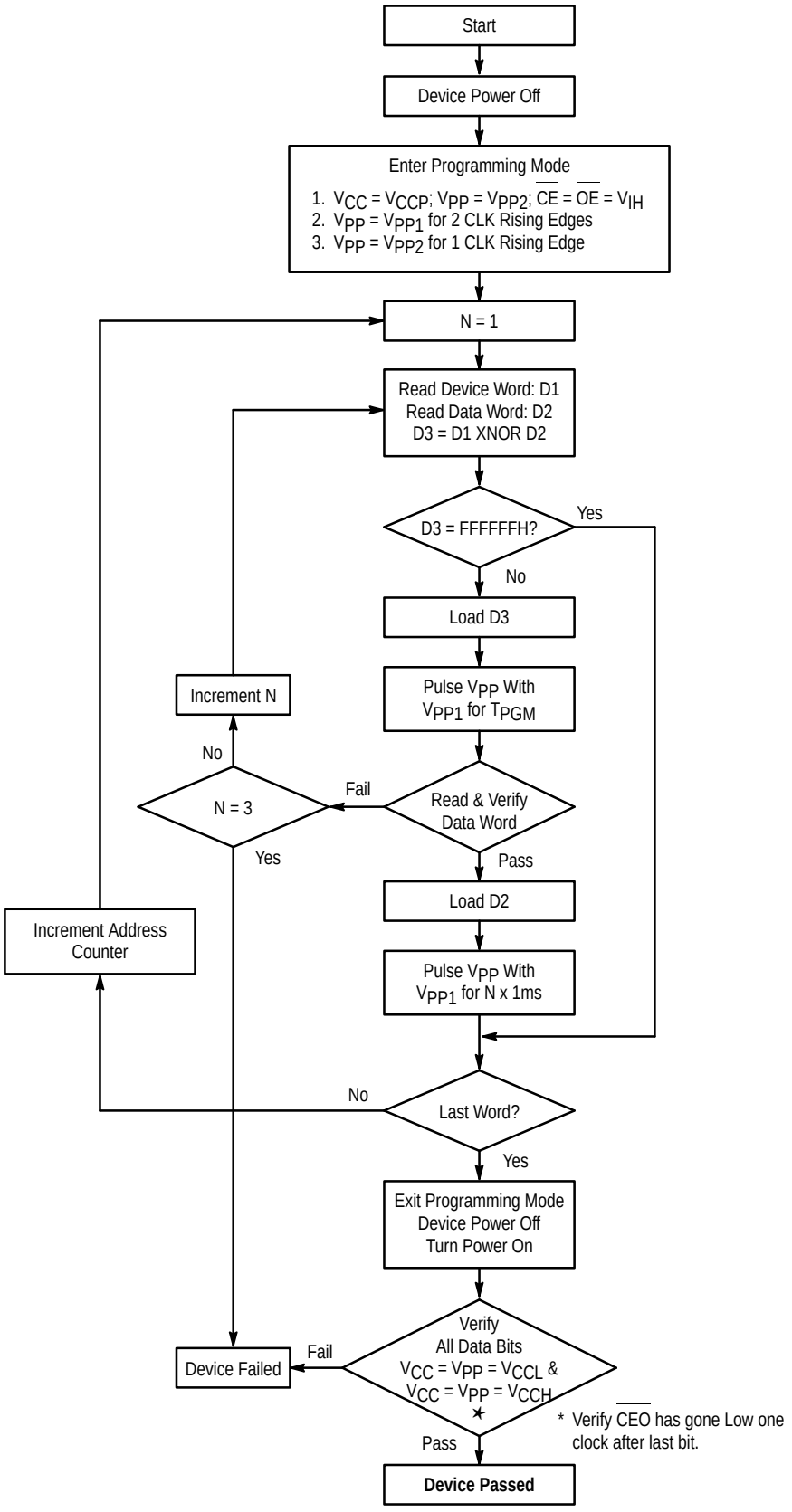


Figure 2–51. MPA17128 Programming Spec

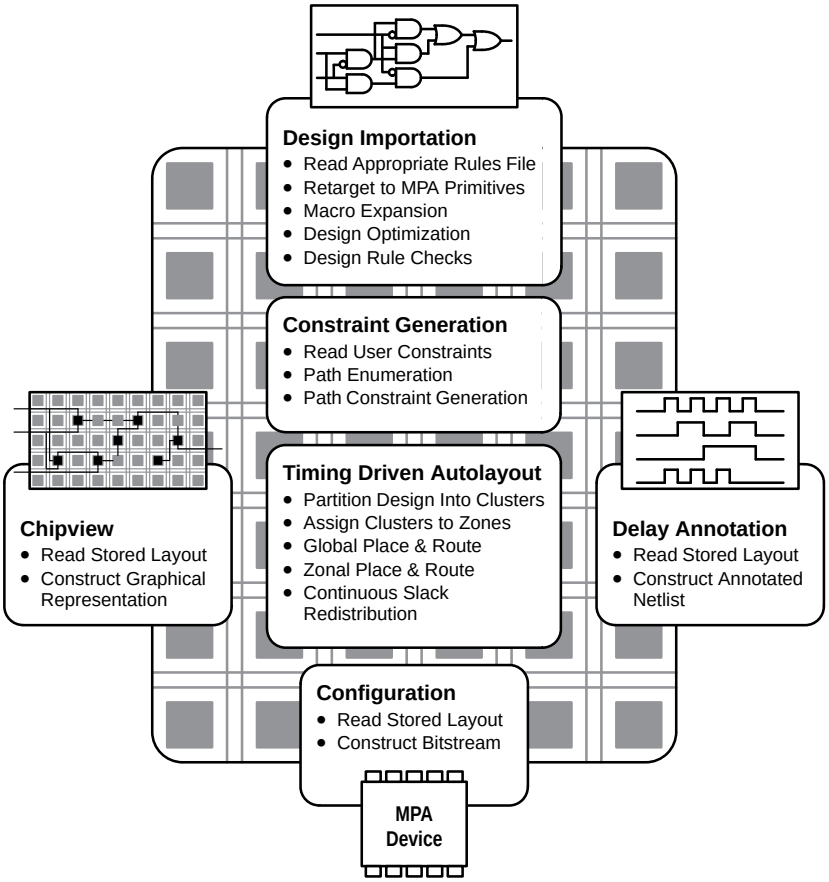
MPA1000 Design System Product Description

Overview

The Motorola Programmable Array (MPA) design system is a bridge between a design capture environment and Motorola field programmable arrays. The MPA design system automatically transforms designs into device configurations which, when loaded into an MPA device, realize a design**.** A design is automatically analyzed, optimized, transformed into MPA cells, partitioned, placed and routed based on timing constraints for every path in the design. MPA design tools understand and optimally utilize the MPA device architecture; this eliminates the need to learn a new set of rules and makes these tools ideally suited for use with logic synthesis. Full incremental design support reduces design implementation time and powerful library retargeting capabilities allow you to reuse designs which may have been implemented on less capable devices. The MPA design system operates on existing hardware platforms and supports design capture and simulation tools from more than 10 vendors. All these features plus on–line, hypermedia, help make the MPA design system a powerful yet extremely easy to use design implementation engine.

Features

- Push Button Implementation
- Optimal Use of MPA Device Resources
- Optimal Results with Gate Level Design Input
- Library of Common MSI Functions
- Design Flow Manager
- Design Retargeter
- Timing Driven with Integrated Static Timing Analysis
- Layout Delay extraction for post layout simulation
- Layout viewer
- Incremental design support
- On–line, hypermedia, documentation
- Supports all popular design capture and simulation tools
- Lowest cost FPGA development systems.
- Instant access; Downloading via the internet (WWW, ftp).



Push Button Design Implementation

The MPA design system minimizes training investment and automatically generates design implementations which meet timing constraints.

The gate level logic and abundant hierarchical routing resources of the MPA device present a rich implementation media for design implementation. MPA design tools understand and optimally utilize the MPA device resources so there are no elaborate rules to learn or design modifications required to begin design capture. Staying focused on end product design rather than implementation tools or device architecture gets the design done faster and, unlike other programmable solutions, without programmable logic device specificity to impede future design migration efforts. The combination of automatic tools and gate level architecture is ideal for traditional schematic driven or high level language based design capture methods. In fact, logic synthesis tools were originally designed for and produce the most efficient results for targeting gate level devices.

A design is analyzed, optimized, transformed into MPA cells, partitioned, placed and routed based on timing constraints for all paths in the design – automatically. A netlist from one of the popular design capture systems or an existing XNF or LPM netlist is imported into the MPA design system. The logic is mapped to a series of MPA cells and the entire resulting netlist is optimized and checked. Based on a simple clock specification, the MPA design system generates timing constraints for all paths in the design. During automatic partitioning, placement and routing path slack time is constantly redistributed insuring only the resources required to meet timing requirements are consumed. Because MPA tools implement the design according to constraints, tool induced design iterations are virtually eliminated. Completed layouts can be transformed into device configurations, as well as annotated simulation netlists. A layout browser is also available.

The MPA design system also includes complete on–line, hypermedia, help covers the device, the design system and the integration kits. Integration kits for Viewlogic, Exemplar, VHDL (1076), Verilog (OVI) and OrCAD are included (contact your vendor for additional kits).All these features add up to a powerful yet extremely easy to use design implementation engine for the MPA product family.

Design Importation

Designs can be captured using schematics, a high level language, or a combination of these entry methods using commercially available design capture and logic synthesis software and the appropriate interface kit. Alternatively, existing designs can be retargeted from other programmable logic devices to the MPA device using commercial logic synthesis tools or the powerful retargetting capabilities provided with MPA design system.

Design importation begins with a netlist and an optional clock specification file. The clock specification file provides a mechanism for the user or design capture tools to document system level timing requirements. In addition, a rich set of attributes can be attached to specific components or nets within the design to specify timing and design pinout constraints.

A retargetting rules file is read and the input netlist is transformed into a series of MPA cells and associated interconnections. Rules files provide a mechanism to perform attribute mapping, cell mapping and macro expansion. By creating custom rule files, the user can extend the importation process from arbitrary sources. The MPA design system comes with rules for it's native library/EDIF. The resulting netlist is optimized to clip unused logic and remove redundant logic. For example: each MPA cell has programmable input inversion capability. All Inverters or non–inverting buffers can be removed from the netlist and replaced with signal sense information attached to each input.

A series of design rule checks are performed to insure design integrity before the layout process begins.

Constraint Generation

Timing constraints, the optimized MPA netlist and static timing analysis is used to generate path slack constraints for all paths in the design. Each unique signal pathway between a register output and a register input throughout the design are enumerated. The total logic and estimated or real wire delays along the path are summed. The time between the active upstream register clock edge and the next active downstream clock edge minus the downstream register setup time is subtracted from the total path delay. This difference is called path slack. If any path in the design has a negative slack value, the implementation will not function at the required clock rate(s).

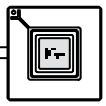
Path constraints are utilized throughout the layout process to insure that a design implementation which meets timing constraints is automatically generated. If no clock or timing specifications are provided, the MPA design system uses the fastest possible clock based on very small net delay estimates to generate the path constraints. This usually results in the best possible implementation, but may take longer than the time required to generate a satisfactory rather than best possible result.

Contrast this to other programmable logic design tools which only provide manual net constraint annotation or net criticality assignment. In these cases significant effort is necessary to generate constraints and many costly iterations are required to tune these constraints for a given design. If any changes are made to the design, another costly round of iterations is required.

Autolayout

The autolayout process makes use of the hierarchical organization of the MPA device to minimize run time and deliver implementations that meet timing requirements. Designs which have diverse timing requirements are ideally implemented because path slack estimates are refined throughout the autolayout process insuring only the resources required to meet timing requirements are consumed.

The process begins by flattening the design and partitioning it into small component groups of approximately



the same size called clusters. A cluster boundary delay estimation is applied to pull the most tightly constrained paths into a minimum number of clusters. The clusters are then assigned to zones talking into account zonal boundary delay cost and relative zone placement delay costs. Other costs like total number of port connections per zone and are also considered. As assignment proceeds, cluster and zone boundary delay costs are added to each path and slack is recomputed.

Next global placement and routing is done. Global routes begin and end on either I/O cells or port cells. Intrazone placement and routing is deferred to a later phase. During global routing all the port cell and I/O cell locations are fixed and the connections between them established. High fanout nets are constructed in a highly regular manner to insure efficient resource utilization. As in partitioning, slack estimates are refined throughout global routing.

Finally the intrazonal placement and routing is done. Cells assigned to a particular zone are placed and routed to other zone cells or zone port cells. Port cells and core cells are constructed to allow port swapping. Core cells can be routed through if necessary. Allowing core cells to act as routing cells allows dynamic adjustment of routing resources within the zone. Dynamic resource adjustment is a powerful design specific adaptation mechanism.

This process produces a layout from which device configurations, delay back annotations, and chipviews can be generated.

Incremental Design Support

When specification changes necessitate design iterations, simply push the button again. Constraints are automatically recalculated and autolayout only reworks those portions of the design which have changed. Full incremental design support means simple design changes to facilitate design verification can be made quickly and easily.

Delay Back Annotation

Designs can be verified through numerous methods. One particularly useful method is the annotation of device and implementation specific delays back into the original simulation environment to improve system or device level simulation accuracy. A MPA device layout can be transformed into an appropriately formatted delay annotation file or annotated netlist quickly and easily. The annotated delay information represents the worst case delays for a given device speed grade.

Chipview

While the MPA design system provides a rich set of reports describing the implementation of a design, a graphical view of the implementation can be indispensable for reviewing overall layout quality. Chipview provides a graphical view of a completed layout. Chipview can be useful during initial design iterations to visually verify I/O pin placements before commencing PCB layout, for example.

Configuration

A layout can be transformed into a device configuration which, when loaded into the appropriate MPA device, produces a physical design realization. Many formatting options are available. The MPA download pod can be used to emulate a serial PROM. Using the pod, device configuration files can be downloaded to a device directly from the PC or workstation development environment.

Integration Kits

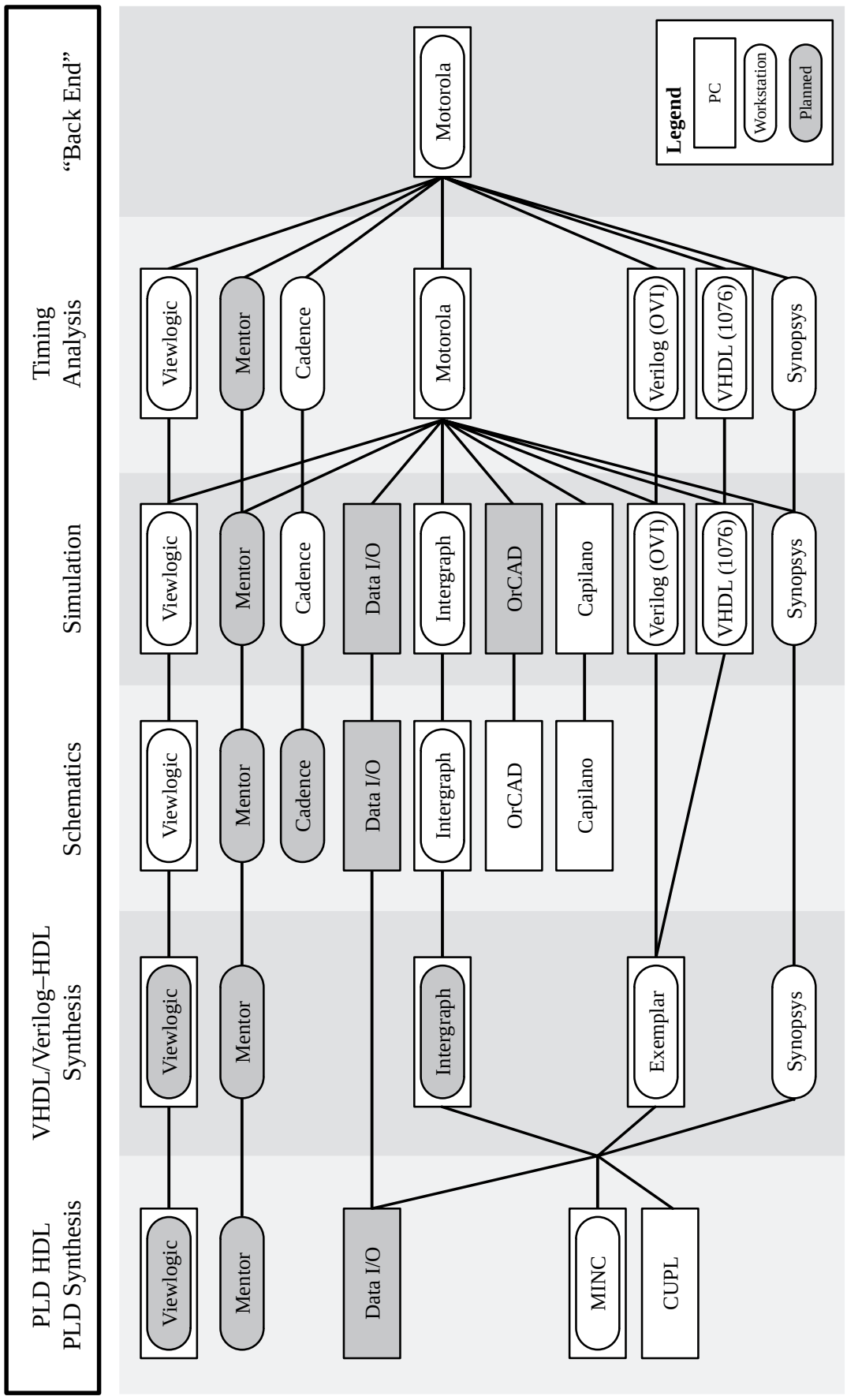
The MPA design system can be used with a large number of commercial electronic design automation software. Figure X–X shows the currently supported vendors and tools. For each supported vendor, an integration kit is provided which facilitates MPA design within that vendors' environment. Many of these kits are available from Motorola and included at no charge on the MPA design system CD–ROM. Other kits can be acquired directly from the vendor. Refer to the MPA Design System Product List for more information.

Low Cost, Easy Access

MPA Design systems are easy to use, competitively priced and widely available. Copies of MPA design system software supporting up to 8000 gates can be downloaded from the World Wide Web (WWW) @ http://Design–NET.com/fpga. Complete kits including download pod, evaluation board, MPA device, CD–ROM and documentation can be ordered from your local authorized Motorola distributor or Motorola sales representative (see appendix Z).

Fast, Efficient Design Implentation With Minimal Investment.

That's MPA!


SOFTWARE FLOWS – WORKSTATION and PC
PLD HDL
PLD Synthesis
VHDL/Verilog–HDL
Synthesis
Schematics
Simulation
Timing
Analysis
"Back End"
Viewlogic
Mentor
Cadence
Data I/O
Intergraph
OrCAD
Capilano
Verilog (OVI)
VHDL (1076)
Synopsys
Exemplar
MINC
CUPL
Motorola
Legend
PC
Workstation
Planned

Design System Product List

MPA Design Kits and Options

Part Number	Description
MPA1E/P MPA1E/W	Entry Level PC with 6 Months Maintenance Entry Level Workstation with 6 Months Maintenance
MPA1S/P MPA1S/W	Standard Level PC with 6 Months Maintenance Standard Level Workstation with 6 Months Maintenance
MPA1M12/P MPA1M12/W	12 Months Maintenance PC 12 Months Maintenance Workstation
MPA1CD/P MPA1CD/W	MPA Design System CDROM PC MPA Design System CDROM Workstation (Requires License)
MPA1/POD	Configuration Download POD
MPA1/BRD	Evaluation Board with MPA Device

Schematic Capture and Simulation

Part Number	Description
MPA1/SCH/P MPA1/SCH/W MPA1/SCH/WF	Schematic Capture PC Schematic Capture Workstation, Node Locked Schematic Capture Workstation, Floating
MPA1/SSM/P MPA1/SSM/W MPA1/SSM/WF	Schematic Capture and 20K Simulation PC Schematic Capture and 20K Simulation Workstation, Node Locked Schematic Capture and 20K Simulation Workstation, Floating
MPA1/SSU/P MPA1/SSU/W MPA1/SSU/WF	Schematic and Simulation UPGRADE** PC Schematic and Simulation UPGRADE** Workstation Node Locked Schematic and Simulation UPGRADE** Workstation Floating
MPA1M12/SCH/P MPA1M12/SCH/W MPA1M12/SSM/P MPA1M12/SSM/W	Schematic Maintenance, 12 Months, PC Schematic Maintenance, 12 Months, Workstation Schematic & Simulation Maintenance, 12 Months, PC Schematic & Simulation Maintenance, 12 Months, Workstation

** Upgrades existing vendor locked Viewlogic for MPA support.

MPA Design Kit Description

- MPA Design System Software on CDROM
  - Design Import and Retargeting
  - Timing Driven Placement and Routing
  - Layout Viewer
  - Layout Delay Extraction (Annotation)
  - Incremental Design
  - On–Line MPA Device and Design Kit Help
- MPA Device Support
  - Entry Level: MPA1016, MPA1036
  - Standard Level: All MPA1000 Devices
- Evaluation Board with MPA Device (MPA1/BRD)
- Download POD (MPA1/POD)
- 6 Months Maintenance
- All Integration Kits*

*The MPA Design System CDROM contains integration kits for Viewlogic, Exemplar, Synopsys, VHDL (1076), Verilog (OVI), and OrCAD. For other integration kits, contact your EDA vendor.

MPA Design System Maintenance

- Support Line Access 1–800–521–6274
- Upgrades

MPA Design System Download POD

- RS232 Connection to Host Computer
- Emulates Serial PROM
- Loads MPA Device via Host Computer

MPA Design System Evaluation Board

- MPA Device
- Simple PCB Facillitating MPA Evaluation

Platform Requirements

- PC Platform – 33MHz 486, 16Mb RAM, 32Mb Swap, 40MB Free Disk Space, Serial Port, Windows 3.1 or Later, Windows/NT
- Sun Platform Requirements: Sun SPARC Compatible, 32Mb RAM, 40Mb Swap, 60Mb Free Disk Space, SunOS 4.1.3, Solaris 2.3, Windows Manager: OSF/MOTIF 1.2 X11r5