



Welcome to [E-XFL.COM](https://www.e-xfl.com)

### Understanding [Embedded - CPLDs \(Complex Programmable Logic Devices\)](#)

Embedded - CPLDs, or Complex Programmable Logic Devices, are highly versatile digital logic devices used in electronic systems. These programmable components are designed to perform complex logical operations and can be customized for specific applications. Unlike fixed-function ICs, CPLDs offer the flexibility to reprogram their configuration, making them an ideal choice for various embedded systems. They consist of a set of logic gates and programmable interconnects, allowing designers to implement complex logic circuits without needing custom hardware.

### Applications of Embedded - CPLDs

#### Details

Product Status	Obsolete
Programmable Type	In System Programmable
Delay Time tpd(1) Max	12 ns
Voltage Supply - Internal	3V ~ 3.6V
Number of Logic Elements/Blocks	32
Number of Macrocells	512
Number of Gates	10000
Number of I/O	120
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LQFP
Supplier Device Package	144-TQFP (20x20)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/intel/epm7512aetc144-12n">https://www.e-xfl.com/product-detail/intel/epm7512aetc144-12n</a>

- Software design support and automatic place-and-route provided by Altera's development systems for Windows-based PCs and Sun SPARCstation, and HP 9000 Series 700/800 workstations
- Additional design entry and simulation support provided by EDIF 2 0 0 and 3 0 0 netlist files, library of parameterized modules (LPM), Verilog HDL, VHDL, and other interfaces to popular EDA tools from manufacturers such as Cadence, Exemplar Logic, Mentor Graphics, OrCAD, Synopsys, Synplicity, and VeriBest
- Programming support with Altera's Master Programming Unit (MPU), MasterBlaster™ serial/universal serial bus (USB) communications cable, ByteBlasterMV™ parallel port download cable, and BitBlaster™ serial download cable, as well as programming hardware from third-party manufacturers and any Jam™ STAPL File (.jam), Jam Byte-Code File (.jbc), or Serial Vector Format File- (.svf) capable in-circuit tester

## General Description

MAX 7000A (including MAX 7000AE) devices are high-density, high-performance devices based on Altera's second-generation MAX architecture. Fabricated with advanced CMOS technology, the EEPROM-based MAX 7000A devices operate with a 3.3-V supply voltage and provide 600 to 10,000 usable gates, ISP, pin-to-pin delays as fast as 4.5 ns, and counter speeds of up to 227.3 MHz. MAX 7000A devices in the -4, -5, -6, -7, and some -10 speed grades are compatible with the timing requirements for 33 MHz operation of the PCI Special Interest Group (PCI SIG) *PCI Local Bus Specification, Revision 2.2*. See [Table 2](#).

**Table 2. MAX 7000A Speed Grades**

Device	Speed Grade					
	-4	-5	-6	-7	-10	-12
EPM7032AE	✓			✓	✓	
EPM7064AE	✓			✓	✓	
EPM7128A			✓	✓	✓	✓
EPM7128AE		✓		✓	✓	
EPM7256A			✓	✓	✓	✓
EPM7256AE		✓		✓	✓	
EPM7512AE				✓	✓	✓

The MAX 7000A architecture supports 100% transistor-to-transistor logic (TTL) emulation and high-density integration of SSI, MSI, and LSI logic functions. It easily integrates multiple devices including PALs, GALs, and 22V10s devices. MAX 7000A devices are available in a wide range of packages, including PLCC, BGA, FineLine BGA, Ultra FineLine BGA, PQFP, and TQFP packages. See [Table 3](#) and [Table 4](#).

**Table 3. MAX 7000A Maximum User I/O Pins** *Note (1)*

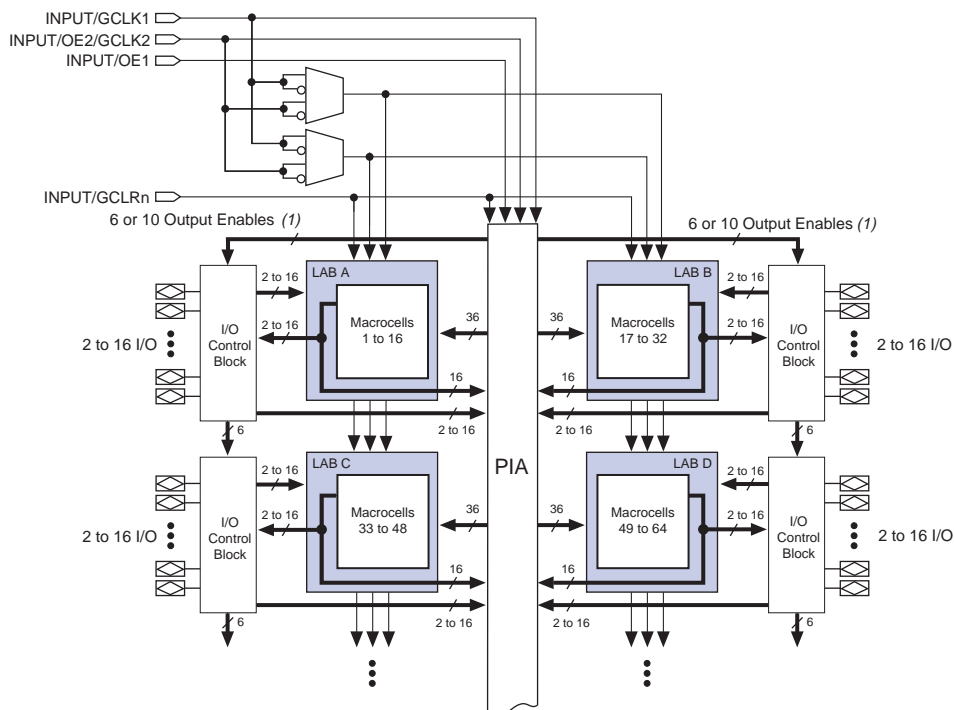
Device	44-Pin PLCC	44-Pin TQFP	49-Pin Ultra FineLine BGA (2)	84-Pin PLCC	100-Pin TQFP	100-Pin FineLine BGA (3)
EPM7032AE	36	36				
EPM7064AE	36	36	41		68	68
EPM7128A				68	84	84
EPM7128AE				68	84	84
EPM7256A					84	
EPM7256AE					84	84
EPM7512AE						

**Table 4. MAX 7000A Maximum User I/O Pins** *Note (1)*

Device	144-Pin TQFP	169-Pin Ultra FineLine BGA (2)	208-Pin PQFP	256-Pin BGA	256-Pin FineLine BGA (3)
EPM7032AE					
EPM7064AE					
EPM7128A	100				100
EPM7128AE	100	100			100
EPM7256A	120		164		164
EPM7256AE	120		164		164
EPM7512AE	120		176	212	212

**Notes to tables:**

- (1) When the IEEE Std. 1149.1 (JTAG) interface is used for in-system programming or boundary-scan testing, four I/O pins become JTAG pins.
- (2) All Ultra FineLine BGA packages are footprint-compatible via the SameFrame™ feature. Therefore, designers can design a board to support a variety of devices, providing a flexible migration path across densities and pin counts. Device migration is fully supported by Altera development tools. See [“SameFrame Pin-Outs” on page 15](#) for more details.
- (3) All FineLine BGA packages are footprint-compatible via the SameFrame feature. Therefore, designers can design a board to support a variety of devices, providing a flexible migration path across densities and pin counts. Device migration is fully supported by Altera development tools. See [“SameFrame Pin-Outs” on page 15](#) for more details.

**Figure 1. MAX 7000A Device Block Diagram****Note:**

- (1) EPM7032AE, EPM7064AE, EPM7128A, EPM7128AE, EPM7256A, and EPM7256AE devices have six output enables. EPM7512AE devices have 10 output enables.

## Logic Array Blocks

The MAX 7000A device architecture is based on the linking of high-performance LABs. LABs consist of 16-macrocell arrays, as shown in [Figure 1](#). Multiple LABs are linked together via the PIA, a global bus that is fed by all dedicated input pins, I/O pins, and macrocells.

Each LAB is fed by the following signals:

- 36 signals from the PIA that are used for general logic inputs
- Global controls that are used for secondary register functions
- Direct input paths from I/O pins to the registers that are used for fast setup times

For registered functions, each macrocell flipflop can be individually programmed to implement D, T, JK, or SR operation with programmable clock control. The flipflop can be bypassed for combinatorial operation. During design entry, the designer specifies the desired flipflop type; the Altera software then selects the most efficient flipflop operation for each registered function to optimize resource utilization.

Each programmable register can be clocked in three different modes:

- Global clock signal. This mode achieves the fastest clock-to-output performance.
- Global clock signal enabled by an active-high clock enable. A clock enable is generated by a product term. This mode provides an enable on each flipflop while still achieving the fast clock-to-output performance of the global clock.
- Array clock implemented with a product term. In this mode, the flipflop can be clocked by signals from buried macrocells or I/O pins.

Two global clock signals are available in MAX 7000A devices. As shown in [Figure 1](#), these global clock signals can be the true or the complement of either of the global clock pins, GCLK1 or GCLK2.

Each register also supports asynchronous preset and clear functions. As shown in [Figure 2](#), the product-term select matrix allocates product terms to control these operations. Although the product-term-driven preset and clear from the register are active high, active-low control can be obtained by inverting the signal within the logic array. In addition, each register clear function can be individually driven by the active-low dedicated global clear pin (GCLRn). Upon power-up, each register in a MAX 7000AE device may be set to either a high or low state. This power-up state is specified at design entry. Upon power-up, each register in EPM7128A and EPM7256A devices are set to a low state.

All MAX 7000A I/O pins have a fast input path to a macrocell register. This dedicated path allows a signal to bypass the PIA and combinatorial logic and be clocked to an input D flipflop with an extremely fast (as low as 2.5 ns) input setup time.

### *Parallel Expanders*

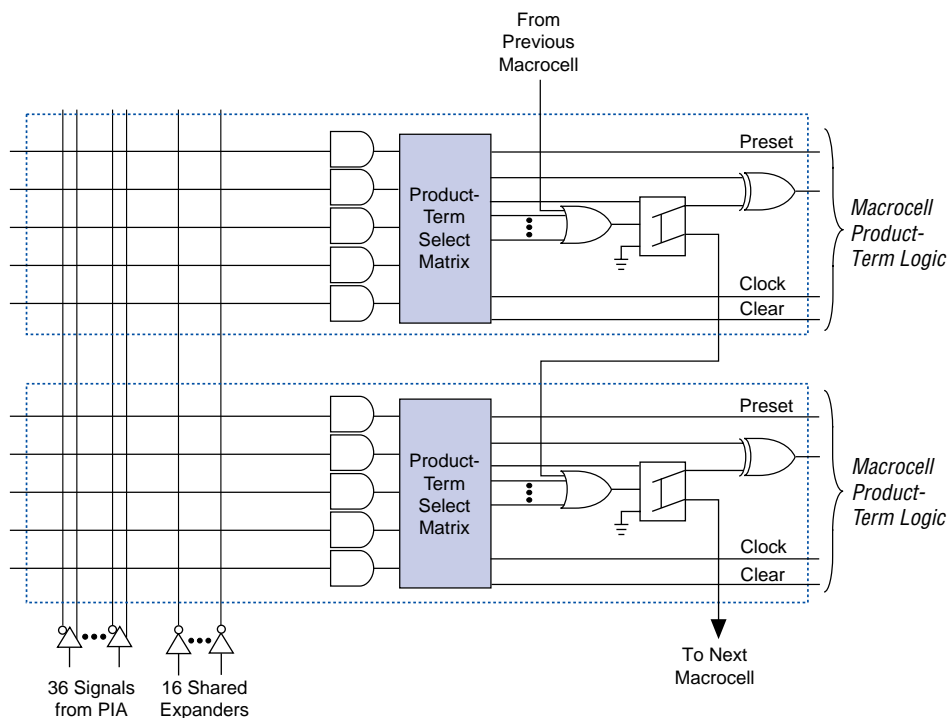
Parallel expanders are unused product terms that can be allocated to a neighboring macrocell to implement fast, complex logic functions. Parallel expanders allow up to 20 product terms to directly feed the macrocell OR logic, with five product terms provided by the macrocell and 15 parallel expanders provided by neighboring macrocells in the LAB.

The compiler can allocate up to three sets of up to five parallel expanders to the macrocells that require additional product terms. Each set of five parallel expanders incurs a small, incremental timing delay ( $t_{PEXP}$ ). For example, if a macrocell requires 14 product terms, the compiler uses the five dedicated product terms within the macrocell and allocates two sets of parallel expanders; the first set includes five product terms, and the second set includes four product terms, increasing the total delay by  $2 \times t_{PEXP}$ .

Two groups of eight macrocells within each LAB (e.g., macrocells 1 through 8 and 9 through 16) form two chains to lend or borrow parallel expanders. A macrocell borrows parallel expanders from lower-numbered macrocells. For example, macrocell 8 can borrow parallel expanders from macrocell 7, from macrocells 7 and 6, or from macrocells 7, 6, and 5. Within each group of eight, the lowest-numbered macrocell can only lend parallel expanders, and the highest-numbered macrocell can only borrow them. [Figure 4](#) shows how parallel expanders can be borrowed from a neighboring macrocell.

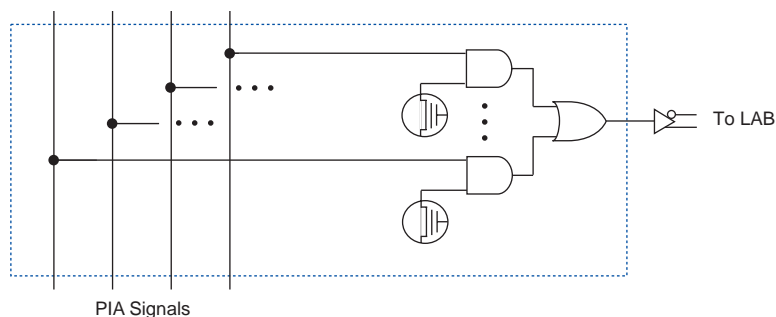
**Figure 4. MAX 7000A Parallel Expanders**

*Unused product terms in a macrocell can be allocated to a neighboring macrocell.*



## Programmable Interconnect Array

Logic is routed between LABs on the PIA. This global bus is a programmable path that connects any signal source to any destination on the device. All MAX 7000A dedicated inputs, I/O pins, and macrocell outputs feed the PIA, which makes the signals available throughout the entire device. Only the signals required by each LAB are actually routed from the PIA into the LAB. Figure 5 shows how the PIA signals are routed into the LAB. An EEPROM cell controls one input to a 2-input AND gate, which selects a PIA signal to drive into the LAB.

**Figure 5. MAX 7000A PIA Routing**

While the routing delays of channel-based routing schemes in masked or FPGAs are cumulative, variable, and path-dependent, the MAX 7000A PIA has a predictable delay. The PIA makes a design's timing performance easy to predict.

## I/O Control Blocks

The I/O control block allows each I/O pin to be individually configured for input, output, or bidirectional operation. All I/O pins have a tri-state buffer that is individually controlled by one of the global output enable signals or directly connected to ground or  $V_{CC}$ . Figure 6 shows the I/O control block for MAX 7000A devices. The I/O control block has 6 or 10 global output enable signals that are driven by the true or complement of two output enable signals, a subset of the I/O pins, or a subset of the I/O macrocells.



## In-System Programmability

MAX 7000A devices can be programmed in-system via an industry-standard 4-pin IEEE Std. 1149.1 (JTAG) interface. ISP offers quick, efficient iterations during design development and debugging cycles. The MAX 7000A architecture internally generates the high programming voltages required to program EEPROM cells, allowing in-system programming with only a single 3.3-V power supply. During in-system programming, the I/O pins are tri-stated and weakly pulled-up to eliminate board conflicts. The pull-up value is nominally 50 k $\Omega$ .

MAX 7000AE devices have an enhanced ISP algorithm for faster programming. These devices also offer an ISP\_Done bit that provides safe operation when in-system programming is interrupted. This ISP\_Done bit, which is the last bit programmed, prevents all I/O pins from driving until the bit is programmed. This feature is only available in EPM7032AE, EPM7064AE, EPM7128AE, EPM7256AE, and EPM7512AE devices.

ISP simplifies the manufacturing flow by allowing devices to be mounted on a PCB with standard pick-and-place equipment before they are programmed. MAX 7000A devices can be programmed by downloading the information via in-circuit testers, embedded processors, the Altera MasterBlaster serial/USB communications cable, ByteBlasterMV parallel port download cable, and BitBlaster serial download cable. Programming the devices after they are placed on the board eliminates lead damage on high-pin-count packages (e.g., QFP packages) due to device handling. MAX 7000A devices can be reprogrammed after a system has already shipped to the field. For example, product upgrades can be performed in the field via software or modem.

In-system programming can be accomplished with either an adaptive or constant algorithm. An adaptive algorithm reads information from the unit and adapts subsequent programming steps to achieve the fastest possible programming time for that unit. A constant algorithm uses a pre-defined (non-adaptive) programming sequence that does not take advantage of adaptive algorithm programming time improvements. Some in-circuit testers cannot program using an adaptive algorithm. Therefore, a constant algorithm must be used. MAX 7000AE devices can be programmed with either an adaptive or constant (non-adaptive) algorithm. EPM7128A and EPM7256A device can only be programmed with an adaptive algorithm; users programming these two devices on platforms that cannot use an adaptive algorithm should use EPM7128AE and EPM7256AE devices.

The Jam Standard Test and Programming Language (STAPL), JEDEC standard JESD 71, can be used to program MAX 7000A devices with in-circuit testers, PCs, or embedded processors.

## Programming Times

The time required to implement each of the six programming stages can be broken into the following two elements:

- A pulse time to erase, program, or read the EEPROM cells.
- A shifting time based on the test clock (TCK) frequency and the number of TCK cycles to shift instructions, address, and data into the device.

By combining the pulse and shift times for each of the programming stages, the program or verify time can be derived as a function of the TCK frequency, the number of devices, and specific target device(s). Because different ISP-capable devices have a different number of EEPROM cells, both the total fixed and total variable times are unique for a single device.

### *Programming a Single MAX 7000A Device*

The time required to program a single MAX 7000A device in-system can be calculated from the following formula:

$$t_{PROG} = t_{PPULSE} + \frac{Cycle_{PTCK}}{f_{TCK}}$$

where:  $t_{PROG}$  = Programming time  
 $t_{PPULSE}$  = Sum of the fixed times to erase, program, and verify the EEPROM cells  
 $Cycle_{PTCK}$  = Number of TCK cycles to program a device  
 $f_{TCK}$  = TCK frequency

The ISP times for a stand-alone verification of a single MAX 7000A device can be calculated from the following formula:

$$t_{VER} = t_{VPULSE} + \frac{Cycle_{VTCK}}{f_{TCK}}$$

where:  $t_{VER}$  = Verify time  
 $t_{VPULSE}$  = Sum of the fixed times to verify the EEPROM cells  
 $Cycle_{VTCK}$  = Number of TCK cycles to verify a device

Figure 8 shows timing information for the JTAG signals.

**Figure 8. MAX 7000A JTAG Waveforms**

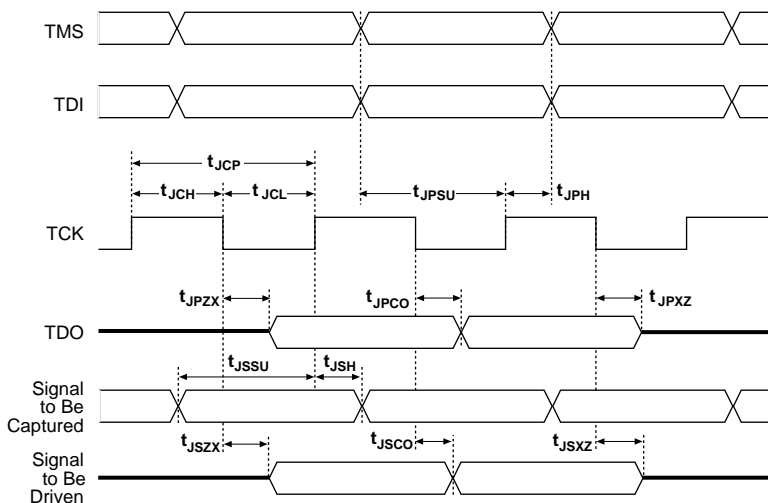


Table 11 shows the JTAG timing parameters and values for MAX 7000A devices.

**Table 11. JTAG Timing Parameters & Values for MAX 7000A Devices** *Note (1)*

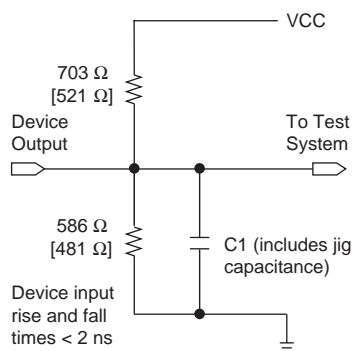
Symbol	Parameter	Min	Max	Unit
$t_{JCP}$	TCK clock period	100		ns
$t_{JCH}$	TCK clock high time	50		ns
$t_{JCL}$	TCK clock low time	50		ns
$t_{JPSU}$	JTAG port setup time	20		ns
$t_{JPH}$	JTAG port hold time	45		ns
$t_{JPCO}$	JTAG port clock to output		25	ns
$t_{JPZX}$	JTAG port high impedance to valid output		25	ns
$t_{JPXZ}$	JTAG port valid output to high impedance		25	ns
$t_{JSSU}$	Capture register setup time	20		ns
$t_{JSH}$	Capture register hold time	45		ns
$t_{JSCO}$	Update register clock to output		25	ns
$t_{JSZX}$	Update register high impedance to valid output		25	ns
$t_{JSXZ}$	Update register valid output to high impedance		25	ns

**Note:**

(1) Timing parameters shown in this table apply for all specified VCCIO levels.

**Figure 9. MAX 7000A AC Test Conditions**

Power supply transients can affect AC measurements. Simultaneous transitions of multiple outputs should be avoided for accurate measurement. Threshold tests must not be performed under AC conditions. Large-amplitude, fast-ground-current transients normally occur as the device outputs discharge the load capacitances. When these transients flow through the parasitic inductance between the device ground pin and the test system ground, significant reductions in observable noise immunity can result. Numbers in brackets are for 2.5-V outputs. Numbers without brackets are for 3.3-V outputs.



## Operating Conditions

Tables 13 through 16 provide information on absolute maximum ratings, recommended operating conditions, operating conditions, and capacitance for MAX 7000A devices.

**Table 13. MAX 7000A Device Absolute Maximum Ratings** *Note (1)*

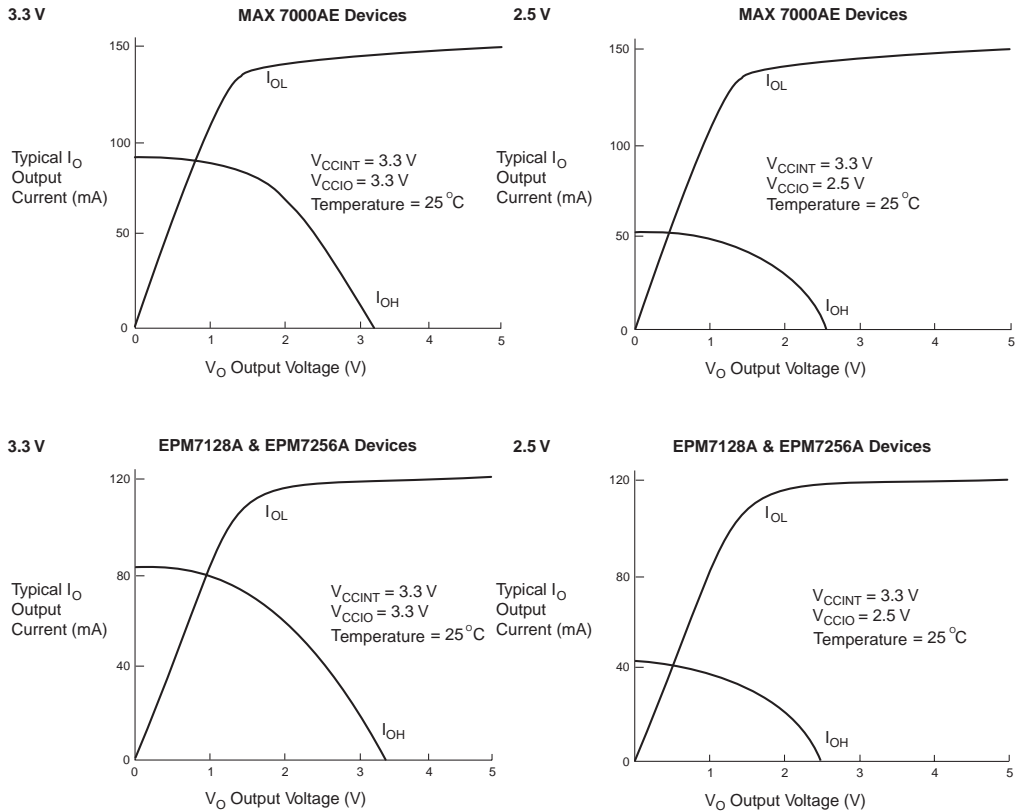
Symbol	Parameter	Conditions	Min	Max	Unit
$V_{CC}$	Supply voltage	With respect to ground (2)	-0.5	4.6	V
$V_I$	DC input voltage		-2.0	5.75	V
$I_{OUT}$	DC output current, per pin		-25	25	mA
$T_{STG}$	Storage temperature	No bias	-65	150	°C
$T_A$	Ambient temperature	Under bias	-65	135	°C
$T_J$	Junction temperature	BGA, FineLine BGA, PQFP, and TQFP packages, under bias		135	°C

**Table 14. MAX 7000A Device Recommended Operating Conditions**

Symbol	Parameter	Conditions	Min	Max	Unit
$V_{CCINT}$	Supply voltage for internal logic and input buffers	(3), (13)	3.0	3.6	V
$V_{CCIO}$	Supply voltage for output drivers, 3.3-V operation	(3)	3.0	3.6	V
	Supply voltage for output drivers, 2.5-V operation	(3)	2.3	2.7	V
$V_{CCISP}$	Supply voltage during in-system programming		3.0	3.6	V
$V_I$	Input voltage	(4)	−0.5	5.75	V
$V_O$	Output voltage		0	$V_{CCIO}$	V
$T_A$	Ambient temperature	Commercial range	0	70	°C
		Industrial range (5)	−40	85	°C
$T_J$	Junction temperature	Commercial range	0	90	°C
		Industrial range (5)	−40	105	°C
		Extended range (5)	−40	130	°C
$t_R$	Input rise time			40	ns
$t_F$	Input fall time			40	ns

Figure 10 shows the typical output drive characteristics of MAX 7000A devices.

**Figure 10. Output Drive Characteristics of MAX 7000A Devices**



## Timing Model

MAX 7000A device timing can be analyzed with the Altera software, a variety of popular industry-standard EDA simulators and timing analyzers, or with the timing model shown in Figure 11. MAX 7000A devices have predictable internal delays that enable the designer to determine the worst-case timing of any design. The software provides timing simulation, point-to-point delay prediction, and detailed timing analysis for device-wide performance evaluation.

**Table 20. EPM7064AE Internal Timing Parameters (Part 2 of 2)** *Note (1)*

Symbol	Parameter	Conditions	Speed Grade						Unit
			-4		-7		-10		
			Min	Max	Min	Max	Min	Max	
$t_{EN}$	Register enable time			0.6		1.0		1.2	ns
$t_{GLOB}$	Global control delay			1.0		1.5		2.2	ns
$t_{PRE}$	Register preset time			1.3		2.1		2.9	ns
$t_{CLR}$	Register clear time			1.3		2.1		2.9	ns
$t_{PIA}$	PIA delay	(2)		1.0		1.7		2.3	ns
$t_{LPA}$	Low-power adder	(6)		3.5		4.0		5.0	ns

**Table 21. EPM7128AE External Timing Parameters** *Note (1)*

Symbol	Parameter	Conditions	Speed Grade						Unit
			-5		-7		-10		
			Min	Max	Min	Max	Min	Max	
t <sub>PD1</sub>	Input to non-registered output	C1 = 35 pF (2)		5.0		7.5		10	ns
t <sub>PD2</sub>	I/O input to non-registered output	C1 = 35 pF (2)		5.0		7.5		10	ns
t <sub>SU</sub>	Global clock setup time	(2)	3.3		4.9		6.6		ns
t <sub>H</sub>	Global clock hold time	(2)	0.0		0.0		0.0		ns
t <sub>FSU</sub>	Global clock setup time of fast input		2.5		3.0		3.0		ns
t <sub>FH</sub>	Global clock hold time of fast input		0.0		0.0		0.0		ns
t <sub>CO1</sub>	Global clock to output delay	C1 = 35 pF	1.0	3.4	1.0	5.0	1.0	6.6	ns
t <sub>CH</sub>	Global clock high time		2.0		3.0		4.0		ns
t <sub>CL</sub>	Global clock low time		2.0		3.0		4.0		ns
t <sub>ASU</sub>	Array clock setup time	(2)	1.8		2.8		3.8		ns
t <sub>AH</sub>	Array clock hold time	(2)	0.2		0.3		0.4		ns
t <sub>ACO1</sub>	Array clock to output delay	C1 = 35 pF (2)	1.0	4.9	1.0	7.1	1.0	9.4	ns
t <sub>ACH</sub>	Array clock high time		2.0		3.0		4.0		ns
t <sub>ACL</sub>	Array clock low time		2.0		3.0		4.0		ns
t <sub>CPPW</sub>	Minimum pulse width for clear and preset	(3)	2.0		3.0		4.0		ns
t <sub>CNT</sub>	Minimum global clock period	(2)		5.2		7.7		10.2	ns
f <sub>CNT</sub>	Maximum internal global clock frequency	(2), (4)	192.3		129.9		98.0		MHz
t <sub>ACNT</sub>	Minimum array clock period	(2)		5.2		7.7		10.2	ns
f <sub>ACNT</sub>	Maximum internal array clock frequency	(2), (4)	192.3		129.9		98.0		MHz



**Table 22. EPM7128AE Internal Timing Parameters (Part 1 of 2)** *Note (1)*

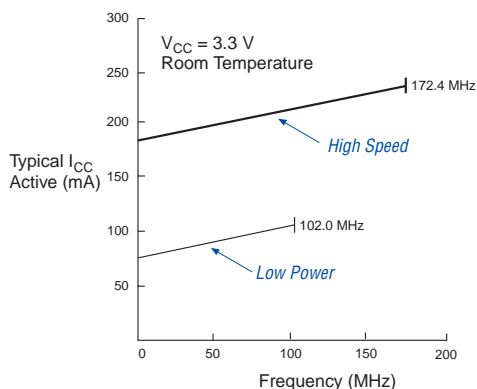
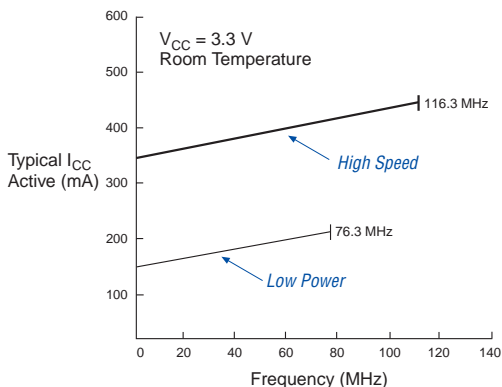
Symbol	Parameter	Conditions	Speed Grade						Unit
			-5		-7		-10		
			Min	Max	Min	Max	Min	Max	
$t_{IN}$	Input pad and buffer delay			0.7		1.0		1.4	ns
$t_{IO}$	I/O input pad and buffer delay			0.7		1.0		1.4	ns
$t_{FIN}$	Fast input delay			2.5		3.0		3.4	ns
$t_{SEXP}$	Shared expander delay			2.0		2.9		3.8	ns
$t_{PEXP}$	Parallel expander delay			0.4		0.7		0.9	ns
$t_{LAD}$	Logic array delay			1.6		2.4		3.1	ns
$t_{LAC}$	Logic control array delay			0.7		1.0		1.3	ns
$t_{IOE}$	Internal output enable delay			0.0		0.0		0.0	ns
$t_{OD1}$	Output buffer and pad delay, slow slew rate = off $V_{CCIO} = 3.3\text{ V}$	$C1 = 35\text{ pF}$		0.8		1.2		1.6	ns
$t_{OD2}$	Output buffer and pad delay, slow slew rate = off $V_{CCIO} = 2.5\text{ V}$	$C1 = 35\text{ pF}$ (5)		1.3		1.7		2.1	ns
$t_{OD3}$	Output buffer and pad delay, slow slew rate = on $V_{CCIO} = 2.5\text{ V}$ or $3.3\text{ V}$	$C1 = 35\text{ pF}$		5.8		6.2		6.6	ns
$t_{ZX1}$	Output buffer enable delay, slow slew rate = off $V_{CCIO} = 3.3\text{ V}$	$C1 = 35\text{ pF}$		4.0		4.0		5.0	ns
$t_{ZX2}$	Output buffer enable delay, slow slew rate = off $V_{CCIO} = 2.5\text{ V}$	$C1 = 35\text{ pF}$ (5)		4.5		4.5		5.5	ns
$t_{ZX3}$	Output buffer enable delay, slow slew rate = on $V_{CCIO} = 3.3\text{ V}$	$C1 = 35\text{ pF}$		9.0		9.0		10.0	ns
$t_{XZ}$	Output buffer disable delay	$C1 = 5\text{ pF}$		4.0		4.0		5.0	ns
$t_{SU}$	Register setup time		1.4		2.1		2.9		ns
$t_H$	Register hold time		0.6		1.0		1.3		ns
$t_{FSU}$	Register setup time of fast input		1.1		1.6		1.6		ns
$t_{FH}$	Register hold time of fast input		1.4		1.4		1.4		ns
$t_{RD}$	Register delay			0.8		1.2		1.6	ns
$t_{COMB}$	Combinatorial delay			0.5		0.9		1.3	ns
$t_{IC}$	Array clock delay			1.2		1.7		2.2	ns

Table 29. EPM7256A External Timing Parameters *Note (1)*

Symbol	Parameter	Conditions	Speed Grade								Unit
			-6		-7		-10		-12		
			Min	Max	Min	Max	Min	Max	Min	Max	
t <sub>PD1</sub>	Input to non-registered output	C1 = 35 pF (2)		6.0		7.5		10.0		12.0	ns
t <sub>PD2</sub>	I/O input to non-registered output	C1 = 35 pF (2)		6.0		7.5		10.0		12.0	ns
t <sub>SU</sub>	Global clock setup time	(2)	3.7		4.6		6.2		7.4		ns
t <sub>H</sub>	Global clock hold time	(2)	0.0		0.0		0.0		0.0		ns
t <sub>FSU</sub>	Global clock setup time of fast input		2.5		3.0		3.0		3.0		ns
t <sub>FH</sub>	Global clock hold time of fast input		0.0		0.0		0.0		0.0		ns
t <sub>CO1</sub>	Global clock to output delay	C1 = 35 pF	1.0	3.3	1.0	4.2	1.0	5.5	1.0	6.6	ns
t <sub>CH</sub>	Global clock high time		3.0		3.0		4.0		4.0		ns
t <sub>CL</sub>	Global clock low time		3.0		3.0		4.0		4.0		ns
t <sub>ASU</sub>	Array clock setup time	(2)	0.8		1.0		1.4		1.6		ns
t <sub>AH</sub>	Array clock hold time	(2)	1.9		2.7		4.0		5.1		ns
t <sub>ACO1</sub>	Array clock to output delay	C1 = 35 pF (2)	1.0	6.2	1.0	7.8	1.0	10.3	1.0	12.4	ns
t <sub>ACH</sub>	Array clock high time		3.0		3.0		4.0		4.0		ns
t <sub>ACL</sub>	Array clock low time		3.0		3.0		4.0		4.0		ns
t <sub>CPPW</sub>	Minimum pulse width for clear and preset	(3)	3.0		3.0		4.0		4.0		ns
t <sub>CNT</sub>	Minimum global clock period	(2)		6.4		8.0		10.7		12.8	ns
f <sub>CNT</sub>	Maximum internal global clock frequency	(2), (4)	156.3		125.0		93.5		78.1		MHz
t <sub>ACNT</sub>	Minimum array clock period	(2)		6.4		8.0		10.7		12.8	ns
f <sub>ACNT</sub>	Maximum internal array clock frequency	(2), (4)	156.3		125.0		93.5		78.1		MHz

**Table 30. EPM7256A Internal Timing Parameters (Part 1 of 2)** *Note (1)*

Symbol	Parameter	Conditions	Speed Grade								Unit
			-6		-7		-10		-12		
			Min	Max	Min	Max	Min	Max	Min	Max	
$t_{IN}$	Input pad and buffer delay			0.3		0.4		0.5		0.6	ns
$t_{IO}$	I/O input pad and buffer delay			0.3		0.4		0.5		0.6	ns
$t_{FIN}$	Fast input delay			2.4		3.0		3.4		3.8	ns
$t_{SEXP}$	Shared expander delay			2.8		3.5		4.7		5.6	ns
$t_{PEXP}$	Parallel expander delay			0.5		0.6		0.8		1.0	ns
$t_{LAD}$	Logic array delay			2.5		3.1		4.2		5.0	ns
$t_{LAC}$	Logic control array delay			2.5		3.1		4.2		5.0	ns
$t_{IOE}$	Internal output enable delay			0.2		0.3		0.4		0.5	ns
$t_{OD1}$	Output buffer and pad delay, slow slew rate = off $V_{CCIO} = 3.3\text{ V}$	$C1 = 35\text{ pF}$		0.3		0.4		0.5		0.6	ns
$t_{OD2}$	Output buffer and pad delay, slow slew rate = off $V_{CCIO} = 2.5\text{ V}$	$C1 = 35\text{ pF}$ (5)		0.8		0.9		1.0		1.1	ns
$t_{OD3}$	Output buffer and pad delay slow slew rate = on $V_{CCIO} = 2.5\text{ V}$ or $3.3\text{ V}$	$C1 = 35\text{ pF}$		5.3		5.4		5.5		5.6	ns
$t_{ZX1}$	Output buffer enable delay slow slew rate = off $V_{CCIO} = 3.3\text{ V}$	$C1 = 35\text{ pF}$		4.0		4.0		5.0		5.0	ns
$t_{ZX2}$	Output buffer enable delay slow slew rate = off $V_{CCIO} = 2.5\text{ V}$	$C1 = 35\text{ pF}$ (5)		4.5		4.5		5.5		5.5	ns
$t_{ZX3}$	Output buffer enable delay slow slew rate = on $V_{CCIO} = 2.5\text{ V}$ or $3.3\text{ V}$	$C1 = 35\text{ pF}$		9.0		9.0		10.0		10.0	ns
$t_{XZ}$	Output buffer disable delay	$C1 = 5\text{ pF}$		4.0		4.0		5.0		5.0	ns
$t_{SU}$	Register setup time		1.0		1.3		1.7		2.0		ns
$t_H$	Register hold time		1.7		2.4		3.7		4.7		ns
$t_{FSU}$	Register setup time of fast input		1.2		1.4		1.4		1.4		ns
$t_{FH}$	Register hold time of fast input		1.3		1.6		1.6		1.6		ns
$t_{RD}$	Register delay			1.6		2.0		2.7		3.2	ns

**Figure 13.  $I_{CC}$  vs. Frequency for MAX 7000A Devices (Part 2 of 2)****EPM7256A & EPM7256AE****EPM7512AE**

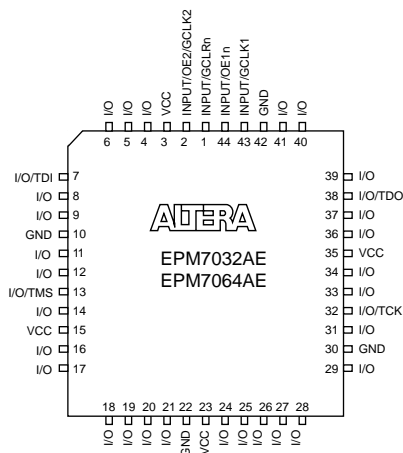
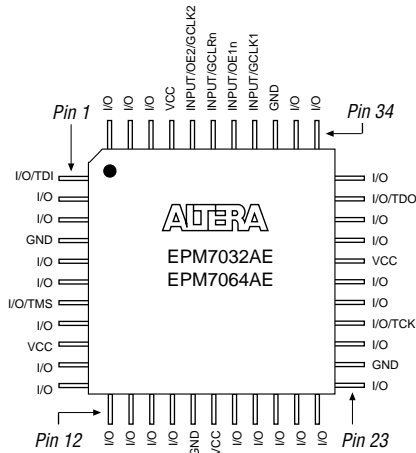
## Device Pin-Outs

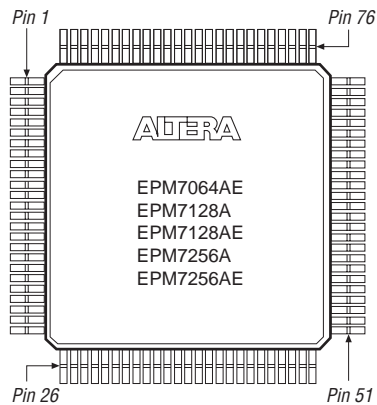
See the Altera web site (<http://www.altera.com>) or the *Altera Digital Library* for pin-out information.

Figures 14 through 23 show the package pin-out diagrams for MAX 7000A devices.

**Figure 14. 44-Pin PLCC/TQFP Package Pin-Out Diagram**

Package outlines not drawn to scale.

**44-Pin PLCC****44-Pin TQFP**

**Figure 17. 100-Pin TQFP Package Pin-Out Diagram***Package outline not drawn to scale.***Figure 18. 100-Pin FineLine BGA Package Pin-Out Diagram***Package outline not drawn to scale.*