**Welcome to 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 | 10 ns |
| Voltage Supply - Internal | 4.75V ~ 5.25V |
| Number of Logic Elements/Blocks | 20 |
| Number of Macrocells | 320 |
| Number of Gates | 6000 |
| Number of I/O | 132 |
| Operating Temperature | 0°C ~ 70°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 208-BFQFP Exposed Pad |
| Supplier Device Package | 208-RQFP (28x28) |
| Purchase URL | https://www.e-xfl.com/product-detail/intel/epm9320arc208-10n |

## General Description

The MAX 9000 family of in-system-programmable, high-density, high-performance EPLDs is based on Altera's third-generation MAX architecture. Fabricated on an advanced CMOS technology, the EEPROM-based MAX 9000 family provides 6,000 to 12,000 usable gates, pin-to-pin delays as fast as 10 ns, and counter speeds of up to 144 MHz. The -10 speed grade of the MAX 9000 family is compliant with the ***PCI Local Bus Specification, Revision 2.2***. Table 3 shows the speed grades available for MAX 9000 devices.

*Table 3. MAX 9000 Speed Grade Availability*

| Device | Speed Grade | | |
|---|---|---|---|
| | -10 | -15 | -20 |
| EPM9320 | | ✓ | ✓ |
| EPM9320A | ✓ | | |
| EPM9400 | | ✓ | ✓ |
| EPM9480 | | ✓ | ✓ |
| EPM9560 | | ✓ | ✓ |
| EPM9560A | ✓ | | |

Table 4 shows the performance of MAX 9000 devices for typical functions.

*Table 4. MAX 9000 Performance* *Note (1)*

| Application | Macrocells Used | Speed Grade | | | Units |
|---|---|---|---|---|---|
| | | -10 | -15 | -20 | |
| 16-bit loadable counter | 16 | 144 | 118 | 100 | MHz |
| 16-bit up/down counter | 16 | 144 | 118 | 100 | MHz |
| 16-bit prescaled counter | 16 | 144 | 118 | 100 | MHz |
| 16-bit address decode | 1 | 5.6 (10) | 7.9 (15) | 10 (20) | ns |
| 16-to-1 multiplexer | 1 | 7.7 (12.1) | 10.9 (18) | 16 (26) | ns |

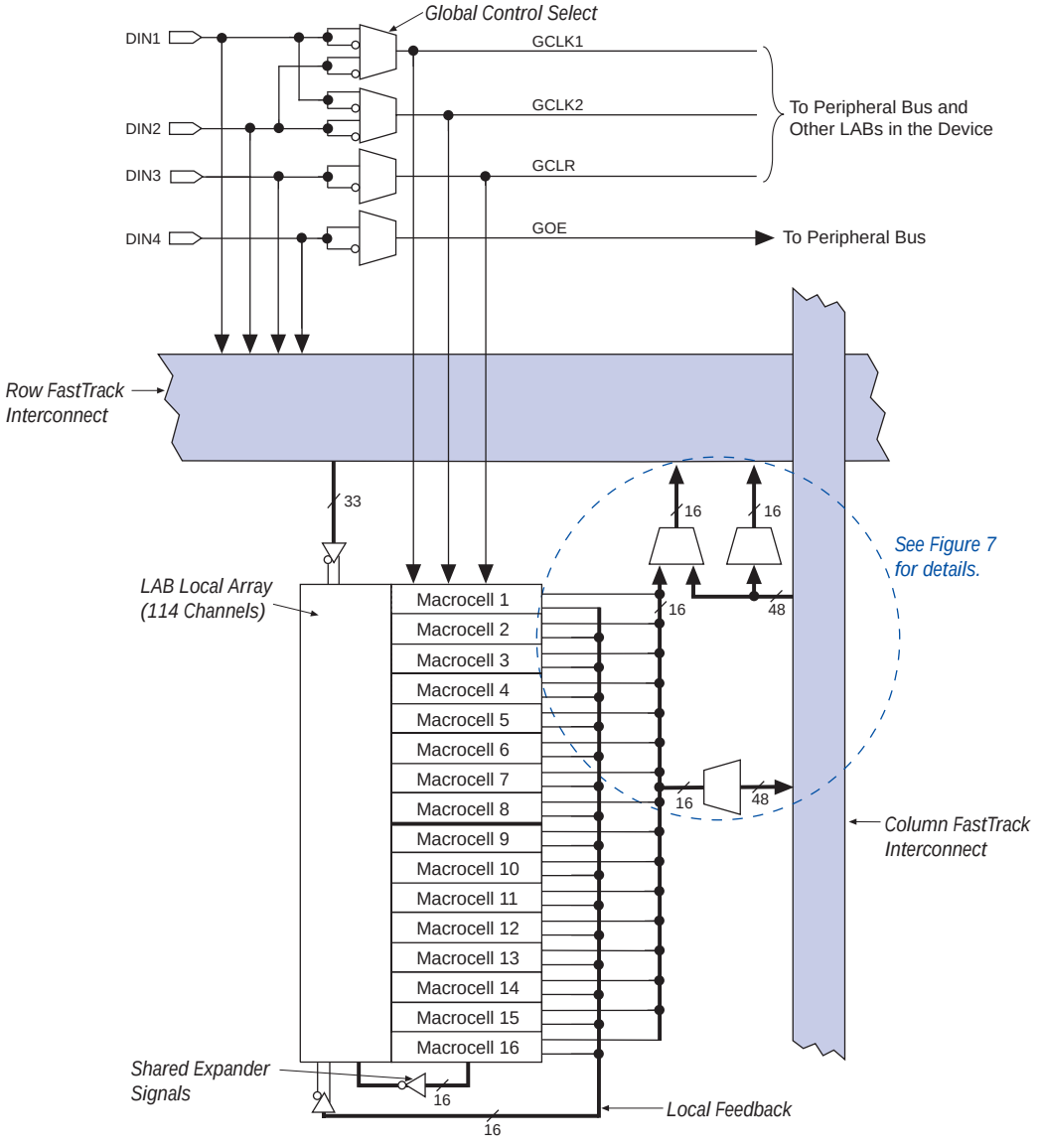
***Note:***

(1) Internal logic array block (LAB) performance is shown. Numbers in parentheses show external delays from row input pin to row I/O pin.

The MAX 9000 architecture supports high-density integration of system-level logic functions. It easily integrates multiple programmable logic devices ranging from PALs, GALs, and 22V10s to field-programmable gate array (FPGA) devices and EPLDs.

LABs drive the row and column interconnect directly. Each macrocell can drive out of the LAB onto one or both routing resources. Once on the row or column interconnect, signals can traverse to other LABs or to the IOCs.

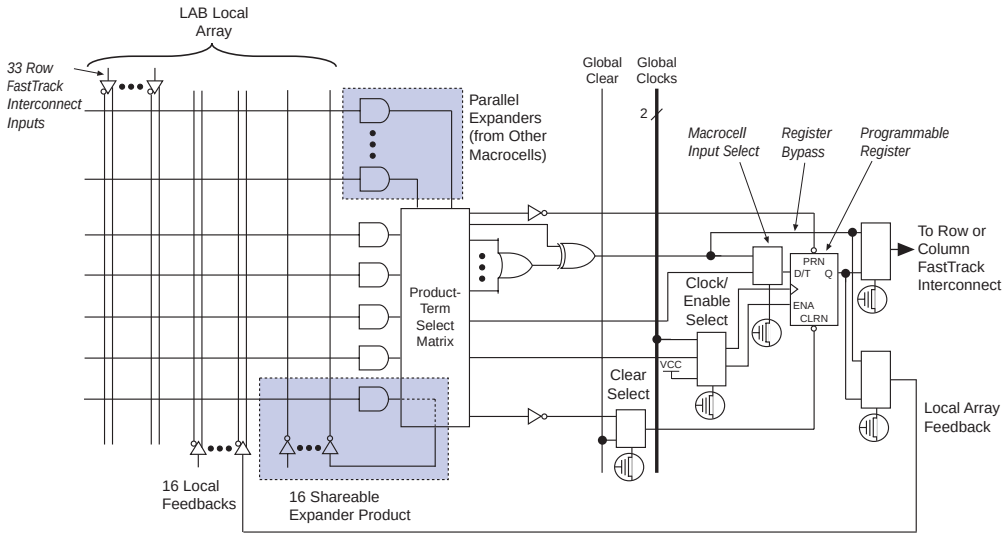
***Figure 2. MAX 9000 Logic Array Block***



## Macrocells

The MAX 9000 macrocell consists of three functional blocks: the product terms, the product-term select matrix, and the programmable register. The macrocell can be individually configured for both sequential and combinatorial logic operation. See Figure 3.

*Figure 3. MAX 9000 Macrocell & Local Array*




Combinatorial logic is implemented in the local array, which provides five product terms per macrocell. The product-term select matrix allocates these product terms for use as either primary logic inputs (to the OR and XOR gates) to implement combinatorial functions, or as secondary inputs to the macrocell's register clear, preset, clock, and clock enable control functions. Two kinds of expander product terms ("expanders") are available to supplement macrocell logic resources:

- Shareable expanders, which are inverted product terms that are fed back into the logic array
- Parallel expanders, which are product terms borrowed from adjacent macrocells

The MAX+PLUS II software automatically optimizes product-term allocation according to the logic requirements of the design.

## Expander Product Terms

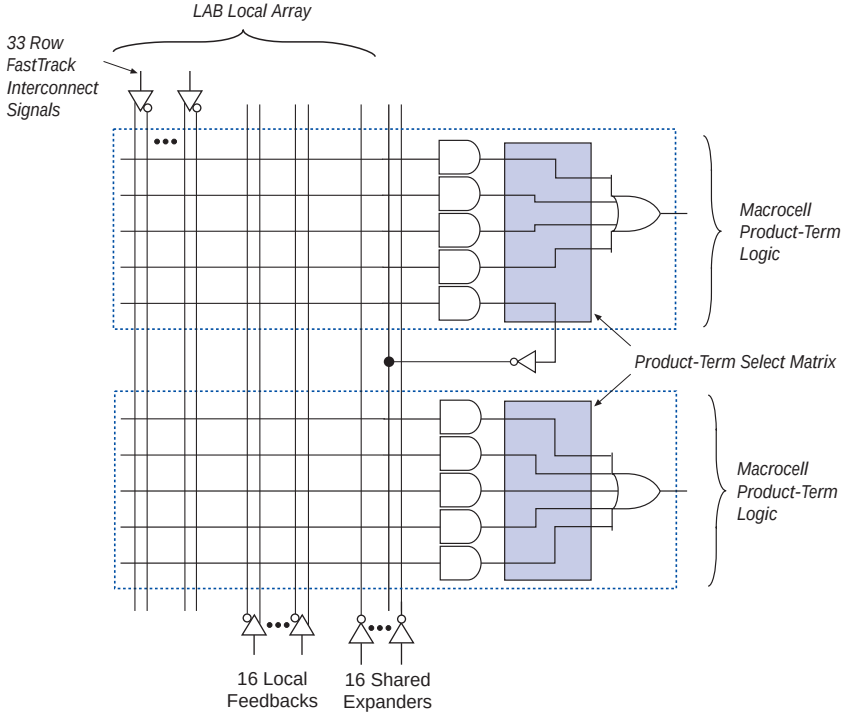
Although most logic functions can be implemented with the five product terms available in each macrocell, some logic functions are more complex and require additional product terms. Although another macrocell can supply the required logic resources, the MAX 9000 architecture also offers both shareable and parallel expander product terms that provide additional product terms directly to any macrocell in the same LAB. These expanders help ensure that logic is synthesized with the fewest possible logic resources to obtain the fastest possible speed.

### *Shareable Expanders*

Each LAB has 16 shareable expanders that can be viewed as a pool of uncommitted single product terms (one from each macrocell) with inverted outputs that feed back into the LAB local array. Each shareable expander can be used and shared by any or all macrocells in the LAB to build complex logic functions. A small delay ($t_{LOCAL}$ + $t_{SEXP}$) is incurred when shareable expanders are used. Figure 4 shows how shareable expanders can feed multiple macrocells.

***Figure 4. MAX 9000 Shareable Expanders***

*Shareable expanders can be shared by any or all macrocells in the LAB.*



The MAX+PLUS II Compiler automatically allocates as many as three sets of up to five parallel expanders to macrocells that require additional product terms. Each set of 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 $2 \times t_{PEXP}$.

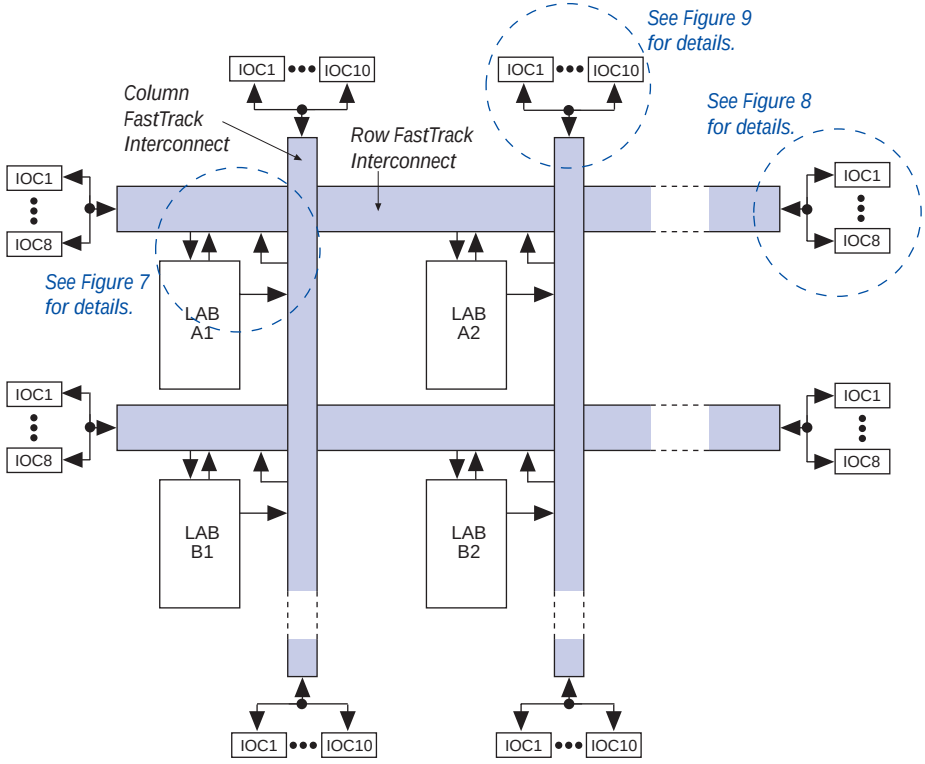
Two groups of eight macrocells within each LAB (e.g., macrocells 1 through 8 and 9 through 16) form two chains to lend or borrow parallel expanders. A macrocell borrows parallel expanders from lower-numbered macrocells. For example, macrocell 8 can borrow parallel expanders from macrocell 7, from macrocells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

## FastTrack Interconnect

In the MAX 9000 architecture, connections between macrocells and device I/O pins are provided by the FastTrack Interconnect, a series of continuous horizontal and vertical routing channels that traverse the entire device. This device-wide routing structure provides predictable performance even in complex designs. In contrast, the segmented routing in FPGAs requires switch matrices to connect a variable number of routing paths, increasing the delays between logic resources and reducing performance. Figure 6 shows the interconnection of four adjacent LABs with row and column interconnects.

***Figure 6. MAX 9000 Device Interconnect Resources***

*Each LAB is named on the basis of its physical row (A, B, C, etc.) and column (1, 2, 3, etc.) position within the device.*




The LABs within MAX 9000 devices are arranged into a matrix of columns and rows. Table 5 shows the number of columns and rows in each MAX 9000 device.

***Table 5. MAX 9000 Rows & Columns***

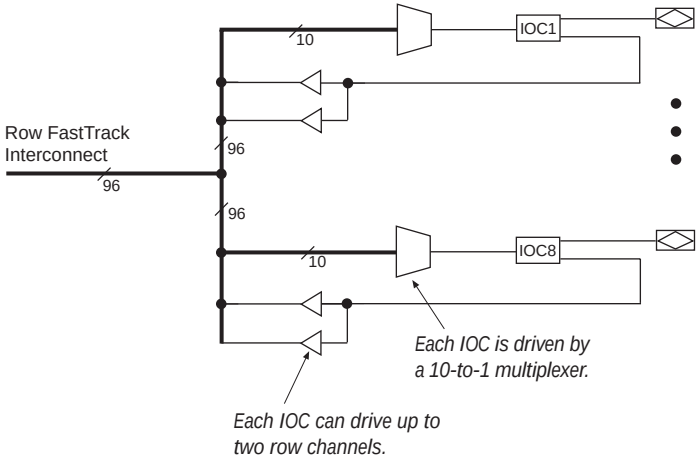
| Devices | Rows | Columns |
|---|---|---|
| EPM9320, EPM9320A | 4 | 5 |
| EPM9400 | 5 | 5 |
| EPM9480 | 6 | 5 |
| EPM9560, EPM9560A | 7 | 5 |

A row interconnect channel can be fed by the output of the macrocell through a 4-to-1 multiplexer that the macrocell shares with three column channels. If the multiplexer is used for a macrocell-to-row connection, the three column signals can access another row channel via an additional 3-to-1 multiplexer. Within any LAB, the multiplexers provide all 48 column channels with access to 32 row channels.

### *Row-to-I/O Cell Connections*

Figure 8 illustrates the connections between row interconnect channels and IOCs. An input signal from an IOC can drive two separate row channels. When an IOC is used as an output, the signal is driven by a 10-to-1 multiplexer that selects the row channels. Each end of the row channel feeds up to eight IOCs on the periphery of the device.

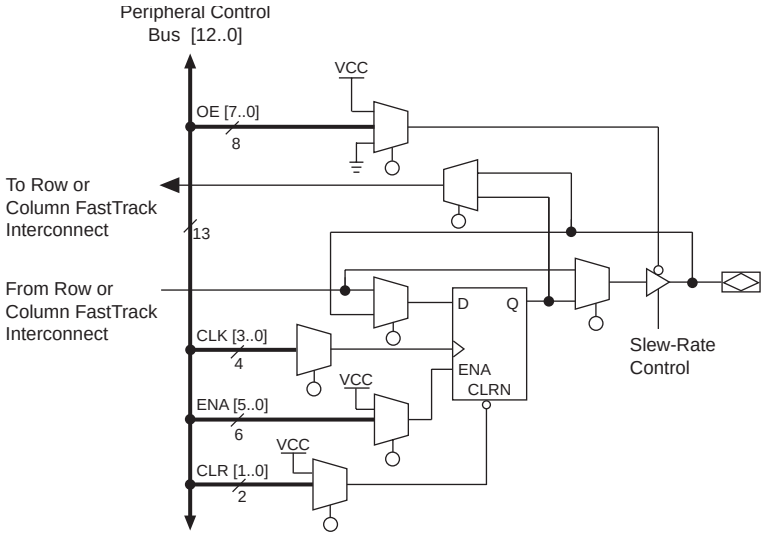
***Figure 8. MAX 9000 Row-to-IOC Connections***




### *Column-to-I/O Cell Connections*

Each end of a column channel has up to 10 IOCs (see Figure 9). An input signal from an IOC can drive two separate column channels. When an IOC is used as an output, the signal is driven by a 17-to-1 multiplexer that selects the column channels.

*Figure 10. MAX 9000 IOC*




I/O pins can be used as input, output, or bidirectional pins. Each IOC has an IOC register with a clock enable input. This register can be used either as an input register for external data that requires fast setup times, or as an output register for data that requires fast clock-to-output performance. The IOC register clock enable allows the global clock to be used for fast clock-to-output performance, while maintaining the flexibility required for selective clocking.

The clock, clock enable, clear, and output enable controls for the IOCs are provided by a network of I/O control signals. These signals can be supplied by either the dedicated input pins or internal logic. The IOC control-signal paths are designed to minimize the skew across the device. All control-signal sources are buffered onto high-speed drivers that drive the signals around the periphery of the device. This "peripheral bus" can be configured to provide up to eight output enable signals, up to four clock signals, up to six clock enable signals, and up to two clear signals. Table 6 on page 18 shows the sources that drive the peripheral bus and how the IOC control signals share the peripheral bus.

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 9000 Device*

The time required to program a single MAX 9000 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 9000 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

## Programming with External Hardware

MAX 9000 devices can be programmed on Windows-based PCs with an Altera Logic Programmer card, the Master Programming Unit (MPU), and the appropriate device adapter. The MPU performs continuity checking to ensure adequate electrical contact between the adapter and the device.

For more information, see the *Altera Programming Hardware Data Sheet*.

The MAX+PLUS II software can use text- or waveform-format test vectors created with the MAX+PLUS II Text Editor or Waveform Editor to test a programmed device. For added design verification, designers can perform functional testing to compare the functional behavior of a MAX 9000 device with the results of simulation.

Data I/O, BP Microsystems, and other programming hardware manufacturers also provide programming support for Altera devices.

For more information, see *Programming Hardware Manufacturers.*

## IEEE Std. 1149.1 (JTAG) Boundary-Scan Support

MAX 9000 devices support JTAG BST circuitry as specified by IEEE Std. 1149.1-1990. Table 10 describes the JTAG instructions supported by the MAX 9000 family. The pin-out tables starting on page 38 show the location of the JTAG control pins for each device. If the JTAG interface is not required, the JTAG pins are available as user I/O pins.

*Table 10. MAX 9000 JTAG Instructions*

| JTAG Instruction | Description |
|---|---|
| SAMPLE/PRELOAD | Allows a snapshot of signals at the device pins to be captured and examined during normal device operation, and permits an initial data pattern output at the device pins. |
| EXTEST | Allows the external circuitry and board-level interconnections to be tested by forcing a test pattern at the output pins and capturing test results at the input pins. |
| BYPASS | Places the 1-bit bypass register between the `TDI` and `TDO` pins, which allows the BST data to pass synchronously through a selected device to adjacent devices during normal device operation. |
| IDCODE | Selects the IDCODE register and places it between `TDI` and `TDO`, allowing the IDCODE to be shifted out of `TDO`. Supported by the EPM9320A, EPM9400, EPM9480, and EPM9560A devices only. |
| UESCODE | Selects the user electronic signature (UESCODE) register and allows the UESCODE to be shifted out of `TDO` serially. This instruction is supported by MAX 9000A devices only. |
| ISP Instructions | These instructions are used when programming MAX 9000 devices via the JTAG ports with the BitBlaster or ByteBlasterMV download cable, or using a Jam File (**.jam**), Jam Byte-Code File (**.jbc**), or Serial Vector Format (**.svf**) File via an embedded processor or test equipment. |

## Operating Conditions

Tables 14 through 20 provide information on absolute maximum ratings, recommended operating conditions, operating conditions, and capacitance for MAX 9000 devices.

*Table 14. MAX 9000 Device Absolute Maximum Ratings* *Note (1)*

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $V_{CC}$ | Supply voltage | With respect to ground *(2)* | –2.0 | 7.0 | V |
| $V_I$ | DC input voltage | | –2.0 | 7.0 | V |
| $V_{CCISP}$ | Supply voltage during in-system programming | | –2.0 | 7.0 | V |
| $I_{OUT}$ | DC output current, per pin | | –25 | 25 | mA |
| $T_{STG}$ | Storage temperature | No bias | –65 | 150 | ° C |
| $T_{AMB}$ | Ambient temperature | Under bias | –65 | 135 | ° C |
| $T_J$ | Junction temperature | Ceramic packages, under bias | | 150 | ° C |
| | | PQFP and RQFP packages, under bias | | 135 | ° C |

*Table 15. MAX 9000 Device Recommended Operating Conditions*

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $V_{CCINT}$ | Supply voltage for internal logic and input buffers | *(3)*, *(4)* | 4.75 (4.50) | 5.25 (5.50) | V |
| $V_{CCIO}$ | Supply voltage for output drivers, 5.0-V operation | *(3)*, *(4)* | 4.75 (4.50) | 5.25 (5.50) | V |
| | Supply voltage for output drivers, 3.3-V operation | *(3)*, *(4)* | 3.00 (3.00) | 3.60 (3.60) | V |
| $V_{CCISP}$ | Supply voltage during in-system programming | | 4.75 | 5.25 | V |
| $V_I$ | Input voltage | | –0.5 | $V_{CCINT}$ + 0.5 | V |
| $V_O$ | Output voltage | | 0 | $V_{CCIO}$ | V |
| $T_A$ | Ambient temperature | For commercial use | 0 | 70 | ° C |
| | | For industrial use | –40 | 85 | ° C |
| $T_J$ | Junction temperature | For commercial use | 0 | 90 | ° C |
| | | For industrial use | –40 | 105 | ° C |
| $t_R$ | Input rise time | | | 40 | ns |
| $t_F$ | Input fall time | | | 40 | ns |

**Table 16. MAX 9000 Device DC Operating Conditions** *Notes (5), (6)*

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $V_{IH}$ | High-level input voltage | *(7)* | 2.0 | $V_{CCINT}$ + 0.5 | V |
| $V_{IL}$ | Low-level input voltage | | –0.5 | 0.8 | V |
| $V_{OH}$ | 5.0-V high-level TTL output voltage | $I_{OH}$ = –4 mA DC, $V_{CCIO}$ = 4.75 V *(8)* | 2.4 | | V |
| | 3.3-V high-level TTL output voltage | $I_{OH}$ = –4 mA DC, $V_{CCIO}$ = 3.00 V *(8)* | 2.4 | | V |
| | 3.3-V high-level CMOS output voltage | $I_{OH}$ = –0.1 mA DC, $V_{CCIO}$ = 3.00 V *(8)* | $V_{CCIO}$ – 0.2 | | V |
| $V_{OL}$ | 5.0-V low level TTL output voltage | $I_{OL}$ = 12 mA DC, $V_{CCIO}$ = 4.75 V *(8)* | | 0.45 | V |
| | 3.3-V low-level TTL output voltage | $I_{OL}$ = 12 mA DC, $V_{CCIO}$ = 3.00 V *(8)* | | 0.45 | V |
| | 3.3-V low-level CMOS output voltage | $I_{OL}$ = 0.1 mA DC, $V_{CCIO}$ = 3.00 V *(8)* | | 0.2 | V |
| $I_I$ | I/O pin leakage current of dedicated input pins | $V_I$ = –0.5 to 5.5 V *(9)* | –10 | 10 | µA |
| $I_{OZ}$ | Tri-state output off-state current | $V_I$ = –0.5 to 5.5 V | –40 | 40 | µA |

**Table 17. MAX 9000 Device Capacitance: EPM9320, EPM9400, EPM9480 & EPM9560 Devices** *Note (10)*

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $C_{DIN1}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 18 | pF |
| $C_{DIN2}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 18 | pF |
| $C_{DIN3}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 17 | pF |
| $C_{DIN4}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 20 | pF |
| $C_{I/O}$ | I/O pin capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 12 | pF |

**Table 18. MAX 9000A Device Capacitance: EPM9320A & EPM9560A Devices** *Note (10)*

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $C_{DIN1}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 16 | pF |
| $C_{DIN2}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 10 | pF |
| $C_{DIN3}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 10 | pF |
| $C_{DIN4}$ | Dedicated input capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 12 | pF |
| $C_{I/O}$ | I/O pin capacitance | $V_{IN}$ = 0 V, f = 1.0 MHz | | 8 | pF |

**Table 19. MAX 9000 Device Typical $I_{CC}$ Supply Current Values**

| Symbol | Parameter | Conditions | EPM9320 | EPM9400 | EPM9480 | EPM9560 | Unit |
|---|---|---|---|---|---|---|---|
| $I_{CC1}$ | $I_{CC}$ supply current (low-power mode, standby, typical) | $V_I$ = ground, no load *(11)* | 106 | 132 | 140 | 146 | mA |

Tables 21 through 24 show timing for MAX 9000 devices.

**Table 21. MAX 9000 External Timing Characteristics** *Note (1)*

| Symbol | Parameter | Conditions | | Speed Grade | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | -10 | | -15 | | -20 | | |
| | | | | Min | Max | Min | Max | Min | Max | |
| $t_{PD1}$ | Row I/O pin input to row I/O pin output | C1 = 35 pF *(2)* | | | 10.0 | | 15.0 | | 20.0 | ns |
| $t_{PD2}$ | Column I/O pin input to column I/O pin output | C1 = 35 pF *(2)* | EPM9320A | | 10.8 | | | | | ns |
| | | | EPM9320 | | | | 16.0 | | 23.0 | ns |
| | | | EPM9400 | | | | 16.2 | | 23.2 | ns |
| | | | EPM9480 | | | | 16.4 | | 23.4 | ns |
| | | | EPM9560A | | 11.4 | | | | | ns |
| | | | EPM9560 | | | | 16.6 | | 23.6 | ns |
| $t_{FSU}$ | Global clock setup time for I/O cell | | | 3.0 | | 5.0 | | 6.0 | | ns |
| $t_{FH}$ | Global clock hold time for I/O cell | | | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{FCO}$ | Global clock to I/O cell output delay | C1 = 35 pF | | 1.0 *(3)* | 4.8 | 1.0 *(3)* | 7.0 | 1.0 *(3)* | 8.5 | ns |
| $t_{CNT}$ | Minimum internal global clock period | *(4)* | | | 6.9 | | 8.5 | | 10.0 | ns |
| $f_{CNT}$ | Maximum internal global clock frequency | *(4)* | | 144.9 | | 117.6 | | 100.0 | | MHz |

*Table 24. Interconnect Delays*

| Symbol | Parameter | Conditions | Speed Grade | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | -10 | | -15 | | -20 | | |
| | | | Min | Max | Min | Max | Min | Max | |
| $t_{LOCAL}$ | LAB local array delay | | | 0.5 | | 0.5 | | 0.5 | ns |
| $t_{ROW}$ | FastTrack row delay | *(6)* | | 0.9 | | 1.4 | | 2.0 | ns |
| $t_{COL}$ | FastTrack column delay | *(6)* | | 0.9 | | 1.7 | | 3.0 | ns |
| $t_{DIN\_D}$ | Dedicated input data delay | | | 4.0 | | 4.5 | | 5.0 | ns |
| $t_{DIN\_CLK}$ | Dedicated input clock delay | | | 2.7 | | 3.5 | | 4.0 | ns |
| $t_{DIN\_CLR}$ | Dedicated input clear delay | | | 4.5 | | 5.0 | | 5.5 | ns |
| $t_{DIN\_IOC}$ | Dedicated input I/O register clock delay | | | 2.5 | | 3.5 | | 4.5 | ns |
| $t_{DIN\_IO}$ | Dedicated input I/O register control delay | | | 5.5 | | 6.0 | | 6.5 | ns |

***Notes to tables:***

(1) These values are specified under the MAX 9000 device recommended operating conditions, shown in Table 15 on page 27.

(2) See *Application Note 77 (Understanding MAX 9000 Timing)* for more information on test conditions for $\mathbf{t_{PD1}}$ and $\mathbf{t_{PD2}}$ delays.

(3) This parameter is a guideline that is sample-tested only. It is based on extensive device characterization. This parameter applies for both global and array clocking as well as both macrocell and I/O cell registers.

(4) Measured with a 16-bit loadable, enabled, up/down counter programmed in each LAB.

(5) The $t_{LPA}$ parameter must be added to the $t_{LOCAL}$ parameter for macrocells running in low-power mode.

(6) The $t_{ROW}$, $t_{COL}$, and $t_{IOC}$ delays are worst-case values for typical applications. Post-compilation timing simulation or timing analysis is required to determine actual worst-case performance.

## Power Consumption

The supply power (P) versus frequency ($\mathbf{f_{MAX}}$) for MAX 9000 devices can be calculated with the following equation:

$$P = P_{INT} + P_{IO} = I_{CCINT} \times V_{CC} + P_{IO}$$

The $P_{IO}$ value, which depends on the device output load characteristics and switching frequency, can be calculated using the guidelines given in *Application Note 74 (Evaluating Power for Altera Devices)*. The $I_{CCINT}$ value depends on the switching frequency and the application logic.

The $I_{CCINT}$ value is calculated with the following equation:

$$I_{CCINT} = (A \times MC_{TON}) + [B \times (MC_{DEV} - MC_{TON})] + (C \times MC_{USED} \times \mathbf{f_{MAX}} \times \mathbf{tog_{LC}})$$

The parameters in this equation are shown below:

$MC_{TON}$ = Number of macrocells with the Turbo Bit option turned on, as reported in the MAX+PLUS II Report File (**.rpt**)
$MC_{DEV}$ = Number of macrocells in the device
$MC_{USED}$ = Number of macrocells used in the design, as reported in the MAX+PLUS II Report File
$\mathbf{f_{MAX}}$ = Highest clock frequency to the device
$\mathbf{tog_{LC}}$ = Average percentage of logic cells toggling at each clock (typically 12.5%)
A, B, C = Constants, shown in Table 25

*Table 25. MAX 9000 $I_{CC}$ Equation Constants*

| Device | Constant A | Constant B | Constant C |
|---|---|---|---|
| EPM9320 | 0.81 | 0.33 | 0.056 |
| EPM9320A | 0.56 | 0.31 | 0.024 |
| EPM9400 | 0.60 | 0.33 | 0.053 |
| EPM9480 | 0.68 | 0.29 | 0.064 |
| EPM9560 | 0.68 | 0.26 | 0.052 |
| EPM9560A | 0.56 | 0.31 | 0.024 |

This calculation provides an $I_{CC}$ estimate based on typical conditions with no output load, using a typical pattern of a 16-bit, loadable, enabled up / down counter in each LAB. Actual $I_{CC}$ values should be verified during operation, because the measurement is sensitive to the actual pattern in the device and the environmental operating conditions. Figure 15 shows typical supply current versus frequency for MAX 9000 devices.

*Figure 15. $I_{CC}$ vs. Frequency for MAX 9000 Devices (Part 1 of 2)*

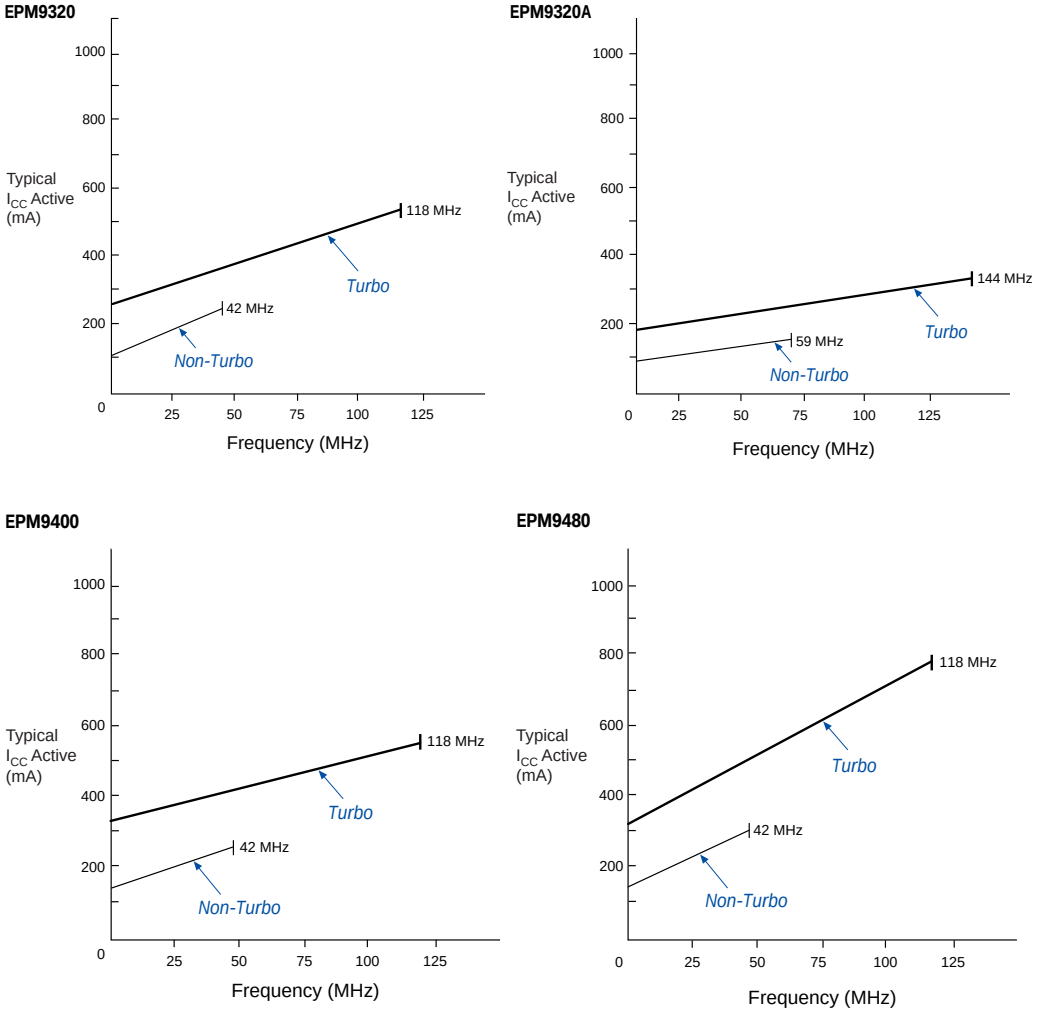
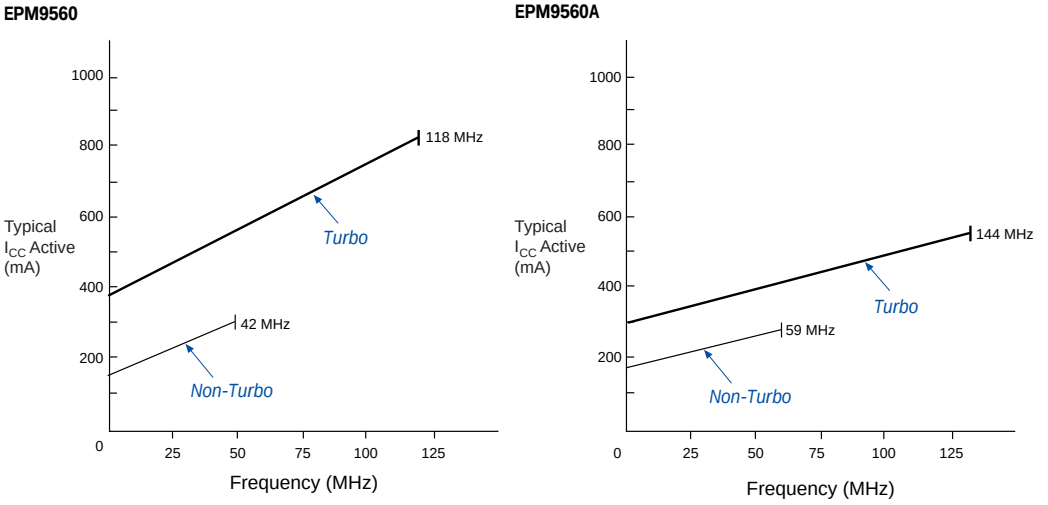


*Figure 15. $I_{CC}$ vs. Frequency for MAX 9000 Devices (Part 2 of 2)*




## Device Pin-Outs

Tables 26 through 29 show the dedicated pin names and numbers for each EPM9320, EPM9320A, EPM9400, EPM9480, EPM9560, and EPM9560A device package.

*Table 26. EPM9320 & EPM9320A Dedicated Pin-Outs (Part 1 of 2)* *Note (1)*

| Pin Name | 84-Pin PLCC *(2)* | 208-Pin RQFP | 280-Pin PGA *(3)* | 356-Pin BGA |
|---|---|---|---|---|
| DIN1 (GCLK1) | 1 | 182 | V10 | AD13 |
| DIN2 (GCLK2) | 84 | 183 | U10 | AF14 |
| DIN3 (GCLR) | 13 | 153 | V17 | AD1 |
| DIN4 (GOE) | 72 | 4 | W2 | AC24 |
| TCK | 43 | 78 | A9 | A18 |
| TMS | 55 | 49 | D6 | E23 |
| TDI | 42 | 79 | C11 | A13 |
| TDO | 30 | 108 | A18 | D3 |

*Table 29. EPM9560 & EPM9560A Dedicated Pin-Outs (Part 1 of 2)* *Note (1)*

| Pin Name | 208-Pin RQFP | 240-Pin RQFP | 280-Pin PGA *(2)* | 304-Pin RQFP *(2)* | 356-Pin BGA |
|---|---|---|---|---|---|
| DIN1 (GCLK1) | 182 | 210 | V10 | 266 | AD13 |
| DIN2 (GCLK2) | 183 | 211 | U10 | 267 | AF14 |
| DIN3 (GCLR) | 153 | 187 | V17 | 237 | AD1 |
| DIN4 (GOE) | 4 | 234 | W2 | 296 | AC24 |
| TCK | 78 | 91 | A9 | 114 | A18 |
| TMS | 49 | 68 | D6 | 85 | E23 |
| TDI | 79 | 92 | C11 | 115 | A13 |
| TDO | 108 | 114 | A18 | 144 | D3 |
| GND | 14, 20, 24, 31, 35, 41, 42, 43, 44, 46, 47, 66, 85, 102, 110, 113, 114, 115, 116, 118, 121, 122, 132, 133, 143, 152, 170, 189, 206 | 5, 14, 25, 34, 45, 54, 65, 66, 81, 96, 110, 115, 126, 127, 146, 147, 166, 167, 186, 200, 216, 229 | D4, D5, D16, E4, E5, E6, E15, E16, F5, F15, G5, G15, H5, H15, J5, J15, K5, K15, L5, L15, M5, M15, N5, N15, P4, P5, P15, P16, R4, R5, R15, R16, T4, T5, T16 | 13, 22, 33, 42, 53, 62, 73, 74, 102, 121, 138, 155, 166, 167, 186, 187, 206, 207, 226, 254, 273, 290 | A9, A22, A25, A26, B25, B26, D2, E1, E26, F2, G1, G25, G26, H2, J1, J25, J26, K2, L26, M26, N1, N25, P26, R2, T1, U2, U26, V1, V25, W25, Y26, AA2, AB1, AB26, AC26, AE1, AF1, AF2, AF4, AF7, AF20 |
| VCCINT (5.0 V only) | 10, 19, 30, 45, 112, 128, 139, 148 | 4, 24, 44, 64, 117, 137, 157, 177 | D15, E8, E10, E12, E14, R7, R9, R11, R13, R14, T14 | 12, 32, 52, 72, 157, 177, 197, 217 | D26, F1, H1, K26, N26, P1, U1, W26, AE26, AF25, AF26 |
| VCCIO (3.3 or 5.0 V) | 5, 25, 36, 55, 72, 91, 111, 127, 138, 159, 176, 195 | 15, 35, 55, 73, 86, 101, 116, 136, 156, 176, 192, 205, 220, 235 | D14, E7, E9, E11, E13, R6, R8, R10, R12, T13, T15 | 3, 23, 43, 63, 91, 108, 127, 156, 176, 196, 216, 243, 260, 279 | A1, A2, A21, B1, B10, B24, D1, H26, K1, M25, R1, V26, AA1, AC25, AF5, AF8, AF19 |

# Revision History

Information contained in the *MAX 9000 Programmable Logic Device Family Data Sheet* version 6.5 supersedes information published in previous versions.

## Version 6.5

Version 6.6 of the *MAX 9000 Programmable Logic Device Family Data Sheet* contains the following change:

- Added Tables 7 through 9.
- Added "Programming Sequence" on page 20 and "Programming Times" on page 20

## Version 6.4

Version 6.4 of the *MAX 9000 Programmable Logic Device Family Data Sheet* contains the following change: Updated text on page 23.

## Version 6.3

Version 6.3 of the *MAX 9000 Programmable Logic Device Family Data Sheet* contains the following change: added Note (7) to Table 16.

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com
Applications Hotline:
(800) 800-EPLD
Customer Marketing:
(408) 544-7104
Literature Services:
lit_req@altera.com