# University Program Design Laboratory Package

August 1997, ver. 1 — User Guide

## Introduction

The University Program Design Laboratory Package was designed to meet the needs of universities teaching digital logic design with state-of-the-art development tools and programmable logic devices (PLDs). The package provides all of the necessary tools for creating and implementing digital logic designs, including the following features:

- MAX+PLUS® II version 7.21 Student Edition development software
- UP 1 Education Board
  - EPM7128S device in an 84-pin plastic J-lead chip carrier (PLCC) package
  - EPF10K20 device in a 240-pin power quad flat pack (RQFP) package
- ByteBlaster™ Parallel Port Download Cable

### MAX+PLUS II Version 7.21 Student Edition Software

The MAX+PLUS II version 7.21 Student Edition software contains many of the features available in the commercial version of MAX+PLUS II, including a completely integrated design flow and an intuitive graphical user interface. This software supports schematic capture and text-based hardware description language design entry, including the Altera® Hardware Description Language (AHDL™) and VHDL. It also provides design programming, compilation, and verification support for the EPM7128S and EPF10K20 devices.

To maximize learning, the MAX+PLUS II software includes complete and instantly accessible on-line help. The student version of the MAX+PLUS II software can be freely distributed to students for installation on their personal computers.

1 For information on how to install the MAX+PLUS II version 7.21 Student Edition software on your computer, see “Software Installation” on page 17.



A-UG-UP1-01

### UP 1 Education Board

The UP 1 Education Board is a stand-alone experiment board based on two of Altera's leading device families: MAX® 7000 and FLEX® 10K. Its simple design, when used with the MAX+PLUS II software, provides a superior platform for learning digital logic design using high-level development tools and PLDs.

The UP 1 Education Board was designed to meet the needs of the educator and the design laboratory environment. The UP 1 Education Board supports both product-term based and look-up table (LUT)-based architectures and includes two PLDs. The EPM7128S device can be programmed in-system with the ByteBlaster download cable. The EPF10K20 device can be configured in-system with either the ByteBlaster download cable or an EPC1 Configuration EPROM (not included).

#### *EPM7128S Device*

The EPM7128S device, a mid-density member of the high-density, high-performance MAX 7000S family, is based on EEPROM elements. The EPM7128S device comes in a socket-mounted 84-pin PLCC package and has 128 macrocells. Each macrocell has a programmable-`AND`/fixed-`OR` array as well as a configurable register with independently programmable clock, clock enable, clear, and preset functions. With a capacity of 2,500 gates and a simple architecture, the EPM7128S device is ideal for introductory designs as well as larger combinatorial and sequential logic functions.

For more information on MAX 7000 devices, go to the *MAX 7000 Programmable Logic Device Family Data Sheet*.

#### *EPF10K20 Device*

The EPF10K20 device, a member of Altera's high-density FLEX 10K family, is based on reconfigurable SRAM elements. The EPF10K20 device comes in a 240-pin RQFP package and has 1,152 logic elements (LEs) and 6 embedded array blocks (EABs). Each LE consists of a 4-input look-up table (LUT), a programmable flipflop, and dedicated signal paths for carry and cascade functions. Each EAB provides 2,048 bits of memory, which can be used to create RAM, ROM, or first-in first-out (FIFO) functions. The EABs can also be used to implement logic functions, such as multipliers, microcontrollers, state machines, and digital signal processing (DSP) functions. With a typical gate count of 20,000 gates, the EPF10K20 device is ideal for advanced designs, including computer architecture, communications, and DSP applications.

For more information on FLEX 10K devices, go to the *FLEX 10K Embedded Programmable Logic Family Data Sheet*.



University Program Design Laboratory Package User Guide

### ByteBlaster Parallel Port Download Cable

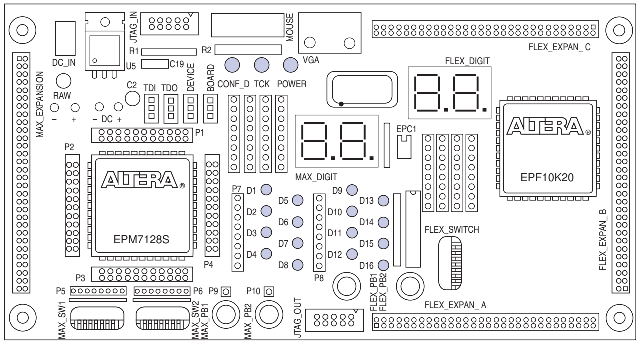
Designs can be easily and quickly downloaded into the UP 1 Education Board using the ByteBlaster download cable, which is a hardware interface to a standard parallel port. This cable channels programming or configuration data between the MAX+PLUS II software and the UP 1 Education Board. Because design changes are downloaded directly to the devices on the board, prototyping is easy and multiple design iterations can be accomplished in quick succession.

For more information on the ByteBlaster download cable, go to the *ByteBlaster Parallel Port Download Cable Data Sheet*.

## UP 1 Education Board Description

The UP 1 Education Board contains the elements described in this section. Figure 1 shows a block diagram of the UP 1 Education Board.

**Figure 1. UP 1 Education Board Block Diagram**



## DC_IN & RAW Power Input

The DC_IN power input accepts a 2.5 mm × 5.55 mm female connector. The acceptable DC input is 7 to 12 V DC at a minimum of 250 mA. The RAW power input consists of two holes for connecting an unregulated power source. The hole marked with a plus sign (+) is the positive input; the hole marked with a minus sign (–) is board common.

## On-Board Voltage Regulator

The on-board voltage regulator, an LM340T, regulates the DC positive input at 5 V. The DC input consist of two holes for connecting a 5-V DC regulated power source. The hole marked with a plus sign (+) is the positive input; the hole marked with a minus sign (–) is board common. A green light-emitting diode (LED) labeled POWER is illuminated when current is flowing from the 5-V DC regulated power source.

## Oscillator

The UP 1 Education Board contains a 25.175-MHz crystal oscillator. The output of the oscillator drives the global clock input on the EPM7128S device (pin 83) and the global clock input on the EPF10K20 device (pin 91).

## JTAG_IN Header

The 10-pin female plug on the ByteBlaster download cable connects with the JTAG_IN 10-pin male header on the UP 1 Education Board. The UP 1 Education Board provides power and ground to the ByteBlaster download cable. Data is shifted into the devices via the `TDI` pin and shifted out of the devices via the `TDO` pin. Table 1 identifies the JTAG_IN pin names when the ByteBlaster is operating in JTAG mode.





*Table 1. JTAG_IN 10-Pin Header Pin-Outs*

| Pin | JTAG Signal |
|---|---|
| 1 | `TCK` |
| 2 | `GND` |
| 3 | `TDO` |
| 4 | `VCC` |
| 5 | `TMS` |
| 6 | No Connect |
| 7 | No Connect |
| 8 | No Connect |
| 9 | `TDI` |
| 10 | `GND` |

## Jumpers

The UP 1 Education Board contains four three-pin jumpers (TDI, TDO, DEVICE, and BOARD) that set the JTAG configuration. You can set the JTAG chain for a variety of configurations (i.e., to program only the EPM7128S device, to configure only the EPF10K20 device, to configure/program both devices, or to connect multiple UP 1 Education Boards together). Figure 1 shows the positions of the three connectors (C1, C2, and C3) on each of the four jumpers.

*Figure 2. Position of C1, C2 & C3 Connectors*

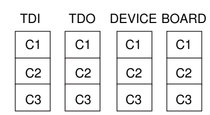



Table 2 defines the settings for each configuration.

| *Table 2. JTAG Jumper Settings* | | | | |
|---|---|---|---|---|
| **Desired Action** | **TDI** | **TDO** | **DEVICE** | **BOARD** |
| Program EPM7128S device only | C1 & C2 | C1 & C2 | C1 & C2 | C1 & C2 |
| Configure EPF10K20 device only | C2 & C3 | C2 & C3 | C1 & C2 | C1 & C2 |
| Program/configure both devices, *Note (1)* | C2 & C3 | C1 & C2 | C2 & C3 | C1 & C2 |
| Connect multiple boards together, *Note (2)* | C2 & C3 | OPEN | C2 & C3 | C2 & C3 |

*Notes:*
(1) The first device in the JTAG chain is the EPF10K20, and the second device is the EPM7128S.
(2) The first device in the JTAG chain is the EPF10K20, and the second device is the EPM7128S. The last board in the chain must be set for a single board configuration (i.e., for programming only the EPM7128S device, configuring only the EPF10K20 device, or configuring/programming both devices). The last board cannot be set for connecting multiple boards together.

During configuration, the green CONF_D LED will turn off and the green TCK LED will modulate to indicate that data is transferring. After the device has successfully configured, the CONF_D LED will illuminate.

1 For information on how to program or configure EPM7128S and EPF10K20 devices, see "Programming or Configuring Devices" on page 18.

## EPM7128S Device

The UP 1 Education Board provides the following resources for the EPM7128S device.

- Socket-mounted 84-pin PLCC package
- Signal pins that are accessible via female headers
- JTAG chain connection for the ByteBlaster
- 2 momentary push-button switches
- 2 octal dipswitches
- 16 LEDs
- Dual-digit 7-segment display
- On-board oscillator (25.175 MHz)
- Expansion port with 42 I/O pins and the dedicated global `CLR`, `OE1`, and `OE2/GCLK2` pins



University Program Design Laboratory Package User Guide

The pins from the EPM7128S device are not pre-assigned to switches and LEDs on the board, but are instead connected to female headers. With direct access to the pins, students can concentrate on design fundamentals and learn about the programmability of I/O pins and PLDs. After successfully compiling and verifying a design with MAX+PLUS II, students can easily connect the assigned I/O pins to the switches and LEDs using common hook-up wire. Students can then download their design into the device and compare their design's simulation to the actual hardware implementation.

## EPM7128S Prototyping Headers

The EPM7128S prototyping headers are female headers that surround the device and provide access to the device's signal pins. The 21 pins on each side of the 84-pin PLCC package connect to one of the 22-pin, dual-row 0.1-inch female headers. The pin numbers for the EPM7128S device are printed on the UP 1 Education Board; an X indicates an unassigned pin. Table 3 lists the pin numbers for the four female headers: P1, P2, P3, and P4. The power, ground, and JTAG signal pins are not accessible through these female headers.

| *Table 3. Pin Numbers for Each Prototyping Header* *Note (1)* | | | | | | | |
|---|---|---|---|---|---|---|---|
| **P1** | | **P2** | | **P3** | | **P4** | |
| **Outside** | **Inside** | **Outside** | **Inside** | **Outside** | **Inside** | **Outside** | **Inside** |
| 75 | 76 | 12 | 13 | 33 | 34 | 54 | 55 |
| 77 | 78 | 14 | 15 | 35 | 36 | 56 | 57 |
| 79 | 80 | 16 | 17 | 37 | 38 | 58 | 59 |
| 81 | 82 | 18 | 19 | 39 | 40 | 60 | 61 |
| 83 | 84 | 20 | 21 | 41 | 42 | 62 | 63 |
| 1 | 2 | 22 | 23 | 43 | 44 | 64 | 65 |
| 3 | 4 | 24 | 25 | 45 | 46 | 66 | 67 |
| 5 | 6 | 26 | 27 | 47 | 48 | 68 | 69 |
| 7 | 8 | 28 | 29 | 49 | 50 | 70 | 71 |
| 9 | 10 | 30 | 31 | 51 | 52 | 72 | 73 |
| 11 | X | 32 | X | 53 | X | 74 | X |

*Note:*
(1) Inside refers to the row of female headers closest to the device; outside refers to the row of female headers furthest from the device.

## MAX_PB1 & MAX_PB2 Push-Buttons

MAX_PB1 and MAX_PB2 are two push-buttons that provide active-low signals and are pulled-up through 10-KΩ resistors. Connections to these signals are easily made by inserting one end of the hook-up wire into the push-button female header. The other end of the hook-up wire should be inserted into the appropriate female header assigned to the I/O pin of the EPM7128S device.

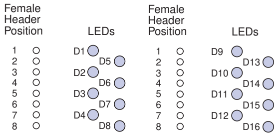
## MAX_SW1 & MAX_SW2 Switches

MAX_SW1 and MAX_SW2 each contain eight switches that provide logic-level signals. These switches are pulled-up through 10-KΩ resistors. Connections to these signals are easily made by inserting one end of the hook-up wire into the female header aligned with the appropriate switch. The other end of the hook-up wire should be inserted into the appropriate female header assigned to the I/O pin of the EPM7128S device. The switch output is set to logic 1 when the switch is open and set to logic 0 when the switch is closed.

## D1 through D16 LEDs

The UP 1 Education Board contains 16 LEDs that are pulled-up with a 330-Ω resistor. An LED is illuminated when a logic 0 is applied to the female header associated with the LED. LEDs D1 through D8 are connected in the same sequence to the female headers (i.e., D1 is connected to position 1, and D2 is connected to position 2). LEDs D9 through D16 are connected in the same sequence to the female headers (i.e., D9 is connected to position 1, and D10 is connected to position 2). See Figure 3.

*Figure 3. LED Positions*








## MAX_DIGIT Display

MAX_DIGIT is a dual-digit seven-segment display connected directly to the EPM7128S device. Each LED segment of the display can be illuminated by driving the connected EPM7128S device I/O pin with a logic 0. Figure 4 shows the name of each segment.

*Figure 4. Display Segment Name*

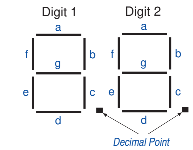



Table 4 lists the pin assignments for each segment.

*Table 4. MAX_DIGIT Segment I/O Connections*

| Display Segment | Pin for Digit 1 | Pin for Digit 2 |
|---|---|---|
| a | 58 | 69 |
| b | 60 | 70 |
| c | 61 | 73 |
| d | 63 | 74 |
| e | 64 | 76 |
| f | 65 | 75 |
| g | 67 | 77 |
| Decimal point | 68 | 79 |

## MAX_EXPANSION

MAX_EXPANSION is a dual row of 0.1-inch spaced holes for accessing signal I/O pins and global signals on the EPM7128S device, power, and ground. Figure 5 shows the numbering convention for the holes.

*Figure 5. MAX_EXPANSION Numbering Convention*

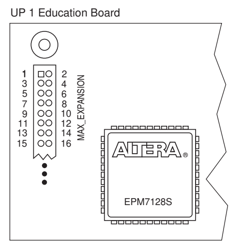



Table 5 lists the signal names and the EPM7128S device pins connected to each hole.

*Table 5. MAX_EXPANSION Signal Names & Device Connections (Part 1 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 1 | RAW | 2 | GND |
| 3 | VCC | 4 | GND |
| 5 | VCC | 6 | GND |
| 7 | No Connect | 8 | No Connect |
| 9 | No Connect | 10 | No Connect |
| 11 | No Connect | 12 | GCLRn/1 |
| 13 | OE1/84 | 14 | OE2/GCLK2/2 |
| 15 | 4 | 16 | 5 |
| 17 | 6 | 18 | 8 |
| 19 | 9 | 20 | 10 |
| 21 | 11 | 22 | 12 |
| 23 | 15 | 24 | 16 |
| 25 | 17 | 26 | 18 |
| 27 | 20 | 28 | 21 |
| 29 | 22 | 30 | 24 |
| 31 | 25 | 32 | 27 |
| 33 | 28 | 34 | 29 |
| 35 | 30 | 36 | 31 |
| 37 | 33 | 38 | 34 |





*Table 5. MAX_EXPANSION Signal Names & Device Connections (Part 2 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 39 | 35 | 40 | 36 |
| 41 | 37 | 42 | 39 |
| 43 | 40 | 44 | 41 |
| 45 | 44 | 46 | 45 |
| 47 | 46 | 48 | 48 |
| 49 | 49 | 50 | 50 |
| 51 | 51 | 52 | 52 |
| 53 | 54 | 54 | 55 |
| 55 | 56 | 56 | 57 |
| 57 | VCC | 58 | GND |
| 59 | VCC | 60 | GND |

## EPF10K20 Device

The UP 1 Education Board provides the following resources for the EPF10K20 device. The pins from the EPF10K20 device are pre-assigned to switches and LEDs on the board.

- JTAG chain connection for the ByteBlaster
- Socket for an EPC1 Configuration EPROM
- 2 momentary push-button switches
- 1 octal dipswitch
- Dual-digit 7-segment display
- On-board oscillator (25.175 MHz)
- VGA port
- Mouse port
- 3 expansion ports, each with 42 I/O pins and 7 global pins

## FLEX_PB1 & FLEX_PB2 Push-Buttons

FLEX_PB1 and FLEX_PB2 are two push-buttons that provide active-low signals to two general-purpose I/O pins on the EPF10K20 device. FLEX_PB1 is connected to pin 28, and FLEX_PB2 is connected to pin 29. Each push-button is pulled-up through a 10-KΩ resistor.

## FLEX_SW1 Switches

FLEX_SW1 contains eight switches that provide logic-level signals to eight general-purpose I/O pins on the EPF10K20 device. An input pin is set to logic 1 when the switch is open and set to logic 0 when the switch is closed. Table 6 lists the pin assignment for each switch.

*Table 6. FLEX_SW1 Pin Assignments*

| Switch | EPF10K20 Pin |
|---|---|
| FLEX_SWITCH-1 | 41 |
| FLEX_SWITCH-2 | 40 |
| FLEX_SWITCH-3 | 39 |
| FLEX_SWITCH-4 | 38 |
| FLEX_SWITCH-5 | 36 |
| FLEX_SWITCH-6 | 35 |
| FLEX_SWITCH-7 | 34 |
| FLEX_SWITCH-8 | 33 |

## FLEX_DIGIT Display

FLEX_DIGIT is a dual-digit seven-segment display connected directly to the EPF10K20 device. Each LED segment on the display can be illuminated by driving the connected EPF10K20 device I/O pin with a logic 0. See Figure 4 on page 9 for the name of each segment. Table 7 lists the pin assignment for each segment.

*Table 7. FLEX_DIGIT Segment I/O Connections*

| Display Segment | Pin for Digit 1 | Pin for Digit 2 |
|---|---|---|
| a | 6 | 17 |
| b | 7 | 18 |
| c | 8 | 19 |
| d | 9 | 20 |
| e | 11 | 21 |
| f | 12 | 23 |
| g | 13 | 24 |
| Decimal point | 14 | 25 |

## VGA Interface

The VGA interface allows the EPF10K20 device to control an external video monitor. This interface is composed of a simple diode-resistor network and a 15-pin D-sub connector (labeled VGA), where the monitor can plug into the UP 1 Education Board. The diode-resistor network and D-sub connector are designed to generate voltages that conform to the VGA standard.





Information about the color of the screen, and the row and column indexing of the screen, are sent from the EPF10K20 device to the monitor via five signals. Three VGA signals are red, green, and blue, while the other two signals are horizontal and vertical synchronization. Manipulating these signals allows images to be written to the monitor's screen.

1 See "VGA Driver Operation" on page 25 for details on how the VGA interface operates.

Table 8 lists the D-sub connector and the EPF10K20 device connections.

*Table 8. D-Sub Connections*

| Signal | D-Sub Connector Pin | EPF10K20 Pin |
|---|---|---|
| `RED` | 1 | 236 |
| `GREEN` | 2 | 237 |
| `BLUE` | 3 | 238 |
| `GND` | 6, 7, 8, 10, 11 | – |
| `HORIZ_SYNC` | 13 | 240 |
| `VERT_SYNC` | 14 | 239 |
| No Connect | 4, 5, 9, 15 | – |

## MOUSE Connector

The MOUSE interface, which consists of a 6-pin mini-DIN connector, allows the EPF10K20 device to receive data from a PS/2 mouse or a PS/2 keyboard. The UP 1 Education Board provides power and ground to the attached mouse or keyboard. The EPF10K20 device outputs the `DATA_CLOCK` signal to the mouse and inputs the data signal from the mouse. Table 9 lists the signal names and the mini-DIN and EPF10K20 pin connections.

1 See "MOUSE Interface Operation" on page 27 for details on how the MOUSE interface operates.

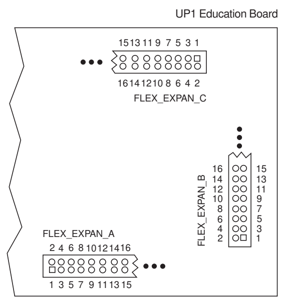
*Table 9. MOUSE Connections*

| Mouse Signal | Mini-DIN Pin | EPF10K20 Pin |
|---|---|---|
| `MOUSE_CLK` | 1 | 30 |
| `MOUSE_DATA` | 3 | 31 |
| `VCC` | 5 | – |
| `GND` | 2 | – |

### FLEX_EXPAN_A, FLEX_EXPAN_B & FLEX_EXPAN_C

FLEX_EXPAN_A, FLEX_EXPAN_B, and FLEX_EXPAN_C are dual rows of 0.1-inch spaced holes for accessing signal I/O pins and global signals on the EPF10K20 device, power, and ground. Figure 6 shows the numbering convention for these holes.

*Figure 6. FLEX_EXPAN_A, FLEX_EXPAN_B & FLEX_EXPAN_C Numbering Convention*




Tables 10, 11, and 12 list the signal name and the EPF10K20 device pin connected to each hole.

*Table 10. FLEX_EXPAN_A Signal Names & Device Connections (Part 1 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 1 | RAW | 2 | GND |
| 3 | VCC | 4 | GND |
| 5 | VCC | 6 | GND |
| 7 | No Connect | 8 | DI1/99 |
| 9 | DI2/92 | 10 | DI3/210 |
| 11 | DI4/212 | 12 | DEV_CLR/209 |
| 13 | DEV_OE/213 | 14 | DEV_CLK2/211 |
| 15 | 45 | 16 | 46 |
| 17 | 48 | 18 | 49 |
| 19 | 50 | 20 | 51 |
| 21 | 53 | 22 | 54 |





*Table 10. FLEX_EXPAN_A Signal Names & Device Connections (Part 2 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 23 | 55 | 24 | 56 |
| 25 | 61 | 26 | 62 |
| 27 | 63 | 28 | 64 |
| 29 | 65 | 30 | 66 |
| 31 | 67 | 32 | 68 |
| 33 | 70 | 34 | 71 |
| 35 | 72 | 36 | 73 |
| 37 | 74 | 38 | 75 |
| 39 | 76 | 40 | 78 |
| 41 | 79 | 42 | 80 |
| 43 | 81 | 44 | 82 |
| 45 | 83 | 46 | 84 |
| 47 | 86 | 48 | 87 |
| 49 | 88 | 50 | 94 |
| 51 | 95 | 52 | 97 |
| 53 | 98 | 54 | 99 |
| 55 | 100 | 56 | 101 |
| 57 | VCC | 58 | GND |
| 59 | VCC | 60 | GND |

*Table 11. FLEX_EXPAN_B Signal Names & Device Connections (Part 1 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 1 | RAW | 2 | GND |
| 3 | VCC | 4 | GND |
| 5 | VCC | 6 | GND |
| 7 | No Connect | 8 | DI1/99 |
| 9 | DI2/92 | 10 | DI3/210 |
| 11 | DI4/212 | 12 | DEV_CLR/209 |
| 13 | DEV_OE/213 | 14 | DEV_CLK2/211 |
| 15 | 109 | 16 | 110 |
| 17 | 111 | 18 | 113 |
| 19 | 114 | 20 | 115 |
| 21 | 116 | 22 | 117 |
| 23 | 118 | 24 | 119 |
| 25 | 120 | 26 | 126 |

*Table 11. FLEX_EXPAN_B Signal Names & Device Connections (Part 2 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 27 | 127 | 28 | 128 |
| 29 | 129 | 30 | 131 |
| 31 | 132 | 32 | 133 |
| 33 | 134 | 34 | 136 |
| 35 | 137 | 36 | 138 |
| 37 | 139 | 38 | 141 |
| 39 | 142 | 40 | 143 |
| 41 | 144 | 42 | 146 |
| 43 | 147 | 44 | 148 |
| 45 | 149 | 46 | 151 |
| 47 | 152 | 48 | 153 |
| 49 | 154 | 50 | 156 |
| 51 | 157 | 52 | 158 |
| 53 | 159 | 54 | 161 |
| 55 | 162 | 56 | 163 |
| 57 | VCC | 58 | GND |
| 59 | VCC | 60 | GND |

*Table 12. FLEX_EXPAN_C Signal Names & Device Connections (Part 1 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 1 | RAW | 2 | GND |
| 3 | VCC | 4 | GND |
| 5 | VCC | 6 | GND |
| 7 | No Connect | 8 | DI1/99 |
| 9 | DI2/92 | 10 | DI3/210 |
| 11 | DI4/212 | 12 | DEV_CLR/209 |
| 13 | DEV_OE/213 | 14 | DEV_CLK2/211 |
| 15 | 175 | 16 | 181 |
| 17 | 182 | 18 | 183 |
| 19 | 184 | 20 | 185 |
| 21 | 186 | 22 | 187 |
| 23 | 188 | 24 | 190 |
| 25 | 191 | 26 | 192 |
| 27 | 193 | 28 | 194 |
| 29 | 195 | 30 | 196 |



University Program Design Laboratory Package User Guide

*Table 12. FLEX_EXPAN_C Signal Names & Device Connections (Part 2 of 2)*

| Hole Number | Signal/Pin | Hole Number | Signal/Pin |
|---|---|---|---|
| 31 | 198 | 32 | 198 |
| 33 | 200 | 34 | 201 |
| 35 | 202 | 36 | 203 |
| 37 | 204 | 38 | 206 |
| 39 | 207 | 40 | 208 |
| 41 | 214 | 42 | 215 |
| 43 | 217 | 44 | 218 |
| 45 | 219 | 46 | 220 |
| 47 | 221 | 48 | 222 |
| 49 | 223 | 50 | 225 |
| 51 | 226 | 52 | 227 |
| 53 | 228 | 54 | 229 |
| 55 | 230 | 56 | 231 |
| 57 | VCC | 58 | GND |
| 59 | VCC | 60 | GND |

## Software Installation

This section describes how to install the MAX+PLUS II version 7.21 Student Edition software for the following operating systems:

- Windows 3.11 and Windows NT 3.51
- Windows 95 and Windows NT 4.0

After installation, students can register to obtain an authorization code via the Altera world-wide web site at the following URL: **http://www.altera.com/maxplus2-student**.

f For complete installation instructions, refer to the **read.me** file on the ***MAX+PLUS II 7.21 Student Edition CD-ROM*** or go to the ***MAX+PLUS II Getting Started*** manual.

### Windows 3.1 & Windows NT 3.51

Follow the steps shown below to install the MAX+PLUS II version 7.21 Student Edition software on your PC.

1. Insert the ***MAX+PLUS II 7.21 Student Edition CD-ROM*** into your CD-ROM drive.

2. In the Windows Program Manager, choose **Run** (File menu).

3. Type: **<***CD-ROM drive***>**`:\pc\maxplus2\install` and choose **OK**. You are guided through the installation procedure.

### Windows 95 & Windows NT 4.0

Follow the steps shown below to install the MAX+PLUS II version 7.21 Student Edition software on your PC.

1. Insert the ***MAX+PLUS II 7.21 Student Edition CD-ROM*** into your CD-ROM drive.

2. Choose **Run** (Start menu).

3. Type: **<***CD-ROM drive***>**`:\pc\maxplus2\install` and choose **OK**. You are guided through the installation procedure.

## Programming or Configuring Devices

Programming or configuring the devices on the UP 1 Education Board requires setting the on-board jumpers and the JTAG programming options in MAX+PLUS II, and connecting the ByteBlaster download cable to the PC's parallel port and to the JTAG_IN connector on the UP 1 Education Board. This section describes how to set these options to perform the following actions:

- Program only the EPM7128S device
- Configure only the EPF10K20 device
- Configure/program both devices
- Connect multiple UP 1 Education Boards together in a chain

### EPM7128S Programming

This section describes the procedures for programming only the EPM7128S device, (i.e., how to set the on-board jumpers, connect the ByteBlaster download cable, and set options in the MAX+PLUS II software).

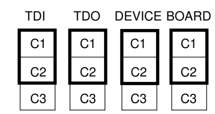




#### *Setting the On-Board Jumpers for EPM7128S Programming*

To program only the EPM7128S device in a JTAG chain, set the jumpers TDI, TDO, DEVICE, and BOARD as shown in Figure 7.

***Figure 7. Jumper Settings for Programming Only the EPM7128S Device***




#### *Connecting the ByteBlaster Download Cable for EPM7128S Programming*

Attach the ByteBlaster directly to the PC's parallel port and to the JTAG_IN connector on the UP 1 Education Board. For more information on setting up the ByteBlaster, go to the *ByteBlaster Parallel Port Download Cable Data Sheet*.

#### *Setting the JTAG Options in MAX+PLUS II for EPM7128S Programming*

The following steps describe how to use the MAX+PLUS II software to program the EPM7128S device in a JTAG chain. For more information on how to use MAX+PLUS II, see MAX+PLUS II Help.

1. To program more than one EPM7128S device, turn on the *Multi-Device JTAG Chain* command (JTAG menu) in the MAX+PLUS II Programmer.

2. Choose **Multi-Device JTAG Chain Setup** (JTAG menu).

3. In the **Multi-Device JTAG Chain Setup** dialog box, select *EPM7128S* in the *Device Name* drop-down list box.

4. Type the name of the programming file for the EPM7128S device in the *Programming File Name* box. The **Select Programming File** button can also be used to browse your computer's directory structure to locate the appropriate programming file.

5. Choose **Add** to add the device and associated programming file to the *Device Names & Programming File Names* box. The number to the left of the device name shows the order of the device in the JTAG chain. The device's associated programming file is displayed on the same line as the device name. If no programming file is associated with a device, "<none>" is displayed next to the device name.

6. Choose **Detect JTAG Chain Info** to have the ByteBlaster check the device count, JTAG ID code, and total instruction length of the JTAG chain. A message just above the **Detect JTAG Chain Info** button reports the information detected by the ByteBlaster. You must manually verify that this message matches the information in the *Device Names & Programming File Names* box.

7. To save the current settings to a JTAG Chain File (**.jcf**) for future use, choose **Save JCF**. In the **Save JCF** dialog box, type the name of the file in the *File Name* box and then select the desired directory in the *Directories* box. Choose **OK**.

8. Choose **OK** to save your changes.

9. In the MAX+PLUS II Programmer, choose **Program**.

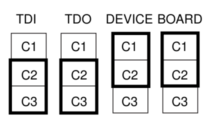
## EPF10K20 Configuration

This section describes the procedures for configuring only the EPF10K20 device, (i.e., how to set the on-board jumpers, connect the ByteBlaster download cable, and set options in the MAX+PLUS II software).

### *Setting the On-Board Jumpers for EPF10K20 Configuration*

To configure only the EPF10K20 device in a JTAG chain, set the jumpers TDI, TDO, DEVICE, and BOARD as shown in Figure 8.

***Figure 8. Jumper Settings for Configuring Only the EPF10K20 Device***






University Program Design Laboratory Package User Guide

### *Connecting the ByteBlaster Download Cable for EPF10K20 Configuration*

Attach the ByteBlaster directly to the PC's parallel port and to the JTAG_IN connector on the UP 1 Education Board. For more information on setting up the ByteBlaster, go to the *ByteBlaster Parallel Port Download Cable Data Sheet*.

### *Setting the JTAG Options in MAX+PLUS II for EPF10K20 Configuration*

The following steps describe how to use MAX+PLUS II to configure the EPF10K20 device in a JTAG chain. For more information on how to configure a device, see MAX+PLUS II Help.

1. To configure more than one EPF10K20 device, turn on the *Multi-Device JTAG Chain* command (JTAG menu) in the MAX+PLUS II Programmer.

2. Choose **Multi-Device JTAG Chain Setup** (JTAG menu).

3. In the **Multi-Device JTAG Chain Setup** dialog box, select *EPF10K20* in the *Device Name* drop-down list box.

4. Type the name of the programming file for the EPF10K20 device in the *Programming File Name* box. The **Select Programming File** button can also be used to browse your computer's directory structure to locate the appropriate programming file.

5. Choose **Add** to add the device and associated programming file to the *Device Names & Programming File Names* box. The number to the left of the device name shows the order of the device in the JTAG chain. The device's associated programming file is displayed on the same line as the device name. If no programming file is associated with a device, "<none>" is displayed next to the device name.

6. Choose **Detect JTAG Chain Info** to have the ByteBlaster check the device count, JTAG ID code, and total instruction length of the JTAG chain. A message just above the **Detect JTAG Chain Info** button reports the information detected by the ByteBlaster. You must manually verify that this message matches the information in the *Device Names & Programming File Names* box.

7. To save the current settings to a JCF for future use, choose **Save JCF**. In the **Save JCF** dialog box, type the name of the file in the *File Name* box and then select the desired directory in the *Directories* box. Choose **OK**.

8. Choose **OK** to save your changes.

9. In the MAX+PLUS II Programmer, choose **Configure**.

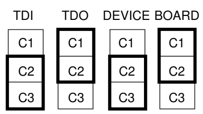
### Configure/Program Both Devices

This section describes the procedures for configuring/programming both the EPF10K20 and EPM7128S devices in a JTAG chain, (i.e., how to set the on-board jumpers, connect the ByteBlaster download cable, and set options in the MAX+PLUS II software).

#### *Setting the On-Board Jumpers for Configuring/Programming Both Devices*

To configure/program both the EPF10K20 and EPM7128S devices in a multi-device JTAG chain, set the jumpers TDI, TDO, DEVICE, and BOARD as shown in Figure 9.

***Figure 9. Jumper Settings for Configuring/Programming Both Devices***




#### *Connecting the ByteBlaster Download Cable for Configuring/Programming Both Devices*

Attach the ByteBlaster directly to the PC's parallel port and to the JTAG_IN connector on the UP 1 Education Board. For more information on setting up the ByteBlaster, go to the *ByteBlaster Parallel Port Download Cable Data Sheet*.

#### *Setting the JTAG Options in MAX+PLUS II for Configuring/Programming Both Devices*

The following steps describe how to use MAX+PLUS II to configure/program both devices in a multi-device JTAG chain. For more information on how to program or configure a device, see MAX+PLUS II Help.





1. Turn on the *Multi-Device JTAG Chain* command (JTAG menu).

2. Choose **Multi-Device JTAG Chain Setup** (JTAG menu).

3. In the **Multi-Device JTAG Chain Setup** dialog box, select the first target device name in the *Device Name* drop-down list box.

4. In the *Programming File Name* box, type the name of the programming file for the device listed in the *Device Name* box. The **Select Programming File** button can also be used to browse your computer's directory structure to locate the appropriate programming file.

5. Choose **Add** to add the device and associated programming file to the *Device Names & Programming File Name* box. The number to the left of the device name shows the device's order in the JTAG chain. The device's associated programming file is displayed on the same line as the device name. If no programming file is associated with a device, "<none>" is displayed next to the device name.

6. Repeat steps 3 through 5 to add information for each device in the JTAG chain.

7. Choose **Detect JTAG Chain Info** to have the ByteBlaster check the device count, JTAG ID code, and total instruction length of the multi-device JTAG chain. A message just above the **Detect JTAG Chain Info** button reports the information detected by the ByteBlaster. You must manually verify that this message matches the information in the *Device Names & Programming File Names* box.

8. To save the current settings to a JCF for future use, choose **Save JCF**. In the **Save JCF** dialog box, type the name of the file in the *File Name* box and then select the desired directory in the *Directories* box. Choose **OK**.

9. Choose **OK** to save the changes.

10. In the MAX+PLUS II Programmer, choose **Configure** to configure all the EPF10K20 devices in the JTAG chain. Then, choose **Program** to program all the EPM7128S devices in the JTAG chain.

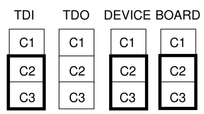
### Connect Multiple UP 1 Education Boards Together in a Chain

This section describes the procedures for connecting multiple UP 1 Education Boards together, (i.e., how to set the on-board jumpers, connect the ByteBlaster download cable, and set options in the MAX+PLUS II software).

### *Setting the On-Board Jumpers for Connecting Multiple UP 1 Education Boards Together*

To configure/program EPM7128S and EPF10K20 devices on multiple UP 1 Education Boards connected in a multi-device JTAG chain, set the jumpers TDI, TDO, DEVICE, and BOARD for all boards except the last board in the chain as shown in Figure 10.

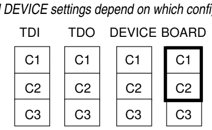
***Figure 10. Jumper Settings for All Boards Except the Last Board in the Chain***




The last UP 1 Education Board in the chain can configure/program one or both devices. However, the BOARD jumper must be set as shown in Figure 11.

***Figure 11. Jumper Settings for the Last Board in the Chain***

*The TDI, TDO, and DEVICE settings depend on which configuration you use.*




### *Connecting the ByteBlaster Download Cable for Connecting Multiple UP 1 Education Boards Together*

Attach the ByteBlaster directly to your PC's parallel port and to the JTAG_IN connector on the UP 1 Education Board. For more information on setting up the ByteBlaster, go to the *ByteBlaster Parallel Port Download Cable Data Sheet*.

### *Setting the JTAG Options in MAX+PLUS II for Connecting Multiple UP 1 Education Boards Together*

For information on how to set the JTAG Options in MAX+PLUS II, see "Setting the JTAG Options in MAX+PLUS II for Configuring/Programming Both Devices" on page 22.

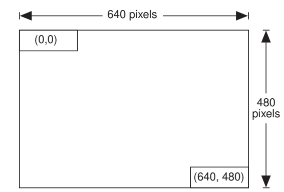




## VGA Driver Operation

A standard VGA monitor consists of a grid of pixels that can be divided into rows and columns. A VGA monitor typically contains 480 rows, with 640 pixels per row, as shown in Figure 12. Each pixel can display various colors, depending on the state of the red, green, and blue signals.

***Figure 12. VGA Monitor***




Each VGA monitor has an internal clock that determines when each pixel is updated. This clock operates at the VGA-specified frequency of 25.175 MHz. The monitor refreshes the screen in a prescribed manner that is partially controlled by the horizontal and vertical synchronization signals. The monitor starts each refresh cycle by updating the pixel in the top left-hand corner of the screen, which can be treated as the origin of an X–Y plane (see Figure 12). After the first pixel is refreshed, the monitor refreshes the remaining pixels in the row. When the monitor receives a pulse on the horizontal synchronization, it refreshes the next row of pixels. This process is repeated until the monitor reaches the bottom of the screen. When the monitor reaches the bottom of the screen, the vertical synchronization pulses, causing the monitor to begin refreshing pixels at the top of the screen (i.e., at [0,0]).

### VGA Timing

For the VGA monitor to work properly, it must receive data at specific times with specific pulses. Horizontal and vertical synchronization pulses must occur at specified times to synchronize the monitor while it is receiving color data. Figures 13 and 14 show the timing waveforms for the color information with respect to the horizontal and vertical synchronization signals.

*Figure 13. Horizontal Refresh Cycle*

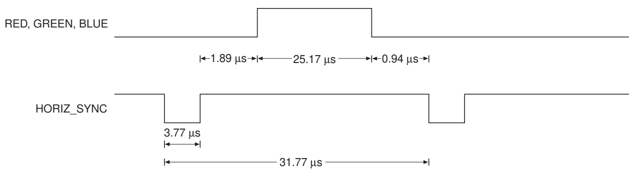
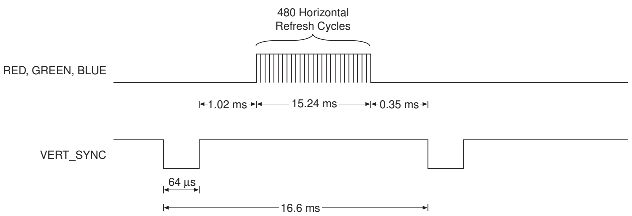



*Figure 14. Vertical Refresh Cycle*




The frequency of operation and the number of pixels that the monitor must update determines the time required to update each pixel, and the time required to update the whole screen. The following equations roughly calculate the time required for the monitor to perform all of its functions.





$$
\begin{aligned}
T_{pixel} &= 1/f_{CLK} \\
&= 40\text{ ns}
\end{aligned}
$$

$$
\begin{aligned}
T_{ROW} &= A \\
&= (T_{pixel} \times 640\text{ pixels})/(\text{row} + \text{guard bands}) \\
&= 25\ \text{s} + B + C + E \\
&= 31.77\ \text{s}
\end{aligned}
$$

$$
\begin{aligned}
T_{screen} &= (T_{ROW} \times 480\text{ rows}) + \text{guard bands} \\
&= 15.5\text{ ms} + P + Q + S \\
&= 16.6\text{ ms}
\end{aligned}
$$

$$
\begin{aligned}
f_{RR} &= 1/T_{ROW} \\
&= 31.5\text{ KHz}
\end{aligned}
$$

$$
\begin{aligned}
f_{SR} &= 1/T_{screen} \\
&= 60\text{ Hz}
\end{aligned}
$$

Where:
$T_{pixel}$ = Time required to update a pixel
$f_{CLK}$ = 25.175 MHz
$T_{ROW}$ = Time required to update one row
$T_{screen}$ = Time required to update the screen
$f_{RR}$ = Row refresh frequency
$f_{SR}$ = Screen refresh frequency

The monitor writes to the screen by sending red, green, blue, horizontal sync, and vertical synchronization signals when the screen is at the expected location. Once the timing of the horizontal and vertical synchronization signals is accurate, the monitor only needs to keep track of the current location, so it can send the correct color data to the pixel.

## MOUSE Interface Operation

You can connect a mouse to the UP 1 Education Board via the 6-pin mini-DIN connector. The data is sent using a synchronous serial protocol, and the transmission is controlled by the `CLK` and `DATA` signals. During non-transmission, `CLK` is at logic 1 and `DATA` can be either logic 0 or logic 1.

Each transmission contains one start bit, eight data bits, odd parity, and one stop bit. Data transmission starts from the least significant bit (LSB), i.e., the sequence of transmission is start bit, `DATA0` through `DATA7`, parity, stop bit. Start bits are logic 0, and stop bits are logic 1. Each clock period is 30 to 50  sec; the data transition to the falling edge of the clock is 5 to 25  sec. Table13 shows the data packet format.

*Table 13. Data Packet Format* *Note (1)*

| Packet Number | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| 1 | YV | XV | YS | XS | 1 | 0 | R | L |
| 2 | X7 | X6 | X5 | X4 | X3 | X2 | X1 | X0 |
| 3 | Y7 | Y6 | Y5 | Y4 | Y3 | Y2 | Y1 | Y0 |

*Note:*
(1) where: L = Left button state (1 = left mouse button is pressed down)
R = Right button state (1 = right mouse button is pressed down)
X0 – X7 = Movement in X direction
Y0 – Y7 = Movement in Y direction
XS, YS = Movement data sign (1 = negative)
XV, YV = Movement data overflow (1 = overflow has occurred)

The mouse operates on a Cartesian coordinate system (i.e., moving to the right is positive, moving to the left is negative, moving up is positive, and moving down is negative). The magnitude of the movement is a function of the mouse's rate of movement. The faster the mouse moves, the greater the magnitude.

ALTERA®

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com
University Program:
university@altera.com
Literature Services:
(888) 3-ALTERA
lit_req@altera.com