Supplement to

Logic and Computer Design Fundamentals
4th Edition[1]

VLSI Programmable Logic Devices: Altera Flex 10K Families

This reading supplement covers specific families of VLSI programmable logic devices from Altera® Corporation. The device families chosen are those most likely to be used in beginning logic laboratories that use PLDs. This supplement is referenced at the end of Chapter 6 of the text.

Coverage of a VLSI PLD family is strongly recommended if the course has an associated laboratory component using the family. Coverage of one or both of the VLSI PLD families is recommended as a basic introduction to VLSI PLDs.

The advantage of using a PLD in the design of digital systems is that it can be programmed to incorporate complex logic functions within a single IC. But for larger or more complex functions, VLSI technology is appropriate. VLSI (Very Large Scale Integrated) refers to digital systems that contain thousands to millions of gates within a single IC chip.

In the last two decades, VLSI approaches have been developed for PLDs to handle designs that in the past were implemented by many small chips or with gate arrays having from 1,000 to millions of gates. The new approaches yield high-capacity programmable logic devices typically sharing the following properties:

1. substantial amounts of uncommitted combinational logic;
2. pre-implemented flip-flops and/or latches;
3. programmable interconnections between the combinational logic, flip-flops, and the chip input/outputs;
4. memories for storing information; and
5. a volatile or non-volatile programming technology.

.

ALTERA® FLEX 10K® EMBEDDED PROGRAMMABLE LOGIC DEVICES[2]

The Altera Flex 10K Embedded Programmable Logic Devices (EPLDs) includes two major families: 1) The basic 10K family and the enhanced 10KE family. There are many important differences in these families, including electrical characteristics and propagation delays. However, these parts with the same basic part numbers (except for the E) have similar architectures and logical structures. As a consequence, the discussion given here, while based on the basic 10K family, applies as well to most of the features of the enhanced 10KE family as well. Significant differences will be pointed out in the discussion.

Altera Flex 10K EPLDs use SRAM technology to store the programming information. After power is applied to the circuit, the program data defining the logic configuration must be loaded into the EPLD SRAM. There are a number of different ways of loading the information, a process referred to as *configuration*. Once the programming information is loaded, the EPLD switches from the programming mode to the operational mode in which the logic is available for use. The logic remains until either the EPLD is reprogrammed or the power is turned off. The ability to reprogram the EPLD allows different logic to be implemented in a system by the same EPLD at different times, leading to the concept of *reconfigurable systems.*

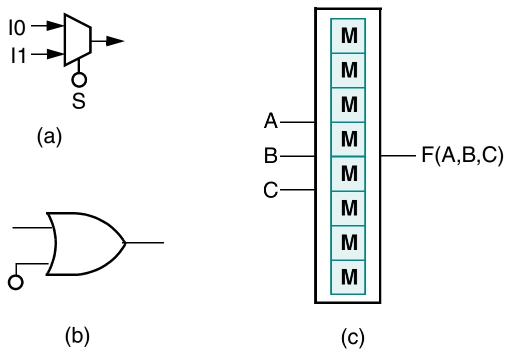


FIGURE 1
SRAM Bit Use in Altera® EPLDs

---

[2] Altera and Flex 10K are registered trademarks of Altera Corporation. FastTrack is a trademark of Altera Corporation. This material in this section is intended for educational use only. For all other uses, consult the documentation available from the Altera Corporation website listed in the References.

Values loaded into SRAM bits during configuration control the logic implemented in an Altera EPLD. Three techniques, illustrated in Figure 1 (multiplexer control, gate control, and lookup table implementation) are used to convert the stored 0's and 1's into logic. In addition, a portion of the SRAM bits reside in the embedded array blocks (EABs) and can be used as stored information that is fixed or dynamic during normal EPLD operation. An embedded array block with fixed contents is used as a ROM, which can implement complex combinational logic, and with variable contents is used as a RAM.

In Figure 1(a), an SRAM cell represented by a small circle is attached to the select input *S* of a 2–to–1 multiplexer. If the SRAM cell contains a 0, then the value on the I0 input of the multiplexer is passed to the multiplexer output. If the SRAM cell contains a 1, then the value on the I1 input is passed to the multiplexer output. The structure is used to make selections between two signals. Sometimes there are two SRAM cells driving a 4–to–1 multiplexer.

In Figure 1(b), an OR gate with one input attached to an SRAM cell is shown. If the SRAM cell contains a 0, then the signal on the other input to the OR gate is pass through to the gate output. If the SRAM cell contains a 1, then the output to the OR gate is a fixed 1. This is one form of enabling circuit, as introduced in Chapter 3. An AND gate with one input attached to an SRAM cell is also used.

The final specialized use of SRAM cells is to build a lookup table, as in Figure 1(c). In the figure, a lookup table for a three-variable function *F(A, B, C)* is illustrated (The actual lookup tables in a Flex 10K EPLD implement four-variable functions). The SRAM cells in the table store the actual truth table of the function, so each cell contains the value of function *F* for the corresponding minterm. The lookup table is functionally equivalent to a multiplexer with the SRAM bits applied to the data inputs and the input variables *A*, *B*, and C on the selection inputs. For example, if (*A*, *B*, *C*) = 0 1 0, the value in SRAM cell 2 (binary 010) appears on the output of the circuit. So the lookup table is actually a multiplexer implementation of combinational logic, as discussed in Chapter 3, with the SRAM cells providing the data inputs.

Architecture

The Altera® Flex 10K™ EPLD structure is shown in Figure 2. The logic within the EPLD is implemented in an array of Logic Array Blocks (LABs) and SRAMs are implemented within Embedded Array Blocks (EABs). An EAB also contains logic that permits it to be treated as a ROM, a memory composed of latches, or a memory composed of flip-flops. Inputs to and outputs from the EPLD are handled by Input/Output Elements (IOEs) along the edges of the EPLD. The LABs and IOEs are interconnected using horizontal rows of fixed wires and vertical columns of fixed wires. A set of wires is referred to as a *channel*. There are programmable connections between the channels and the LABs, the channels and EABs, and the channels and the IOEs. In addition, there are programmable connections between the row interconnect channels and the column interconnect channels.

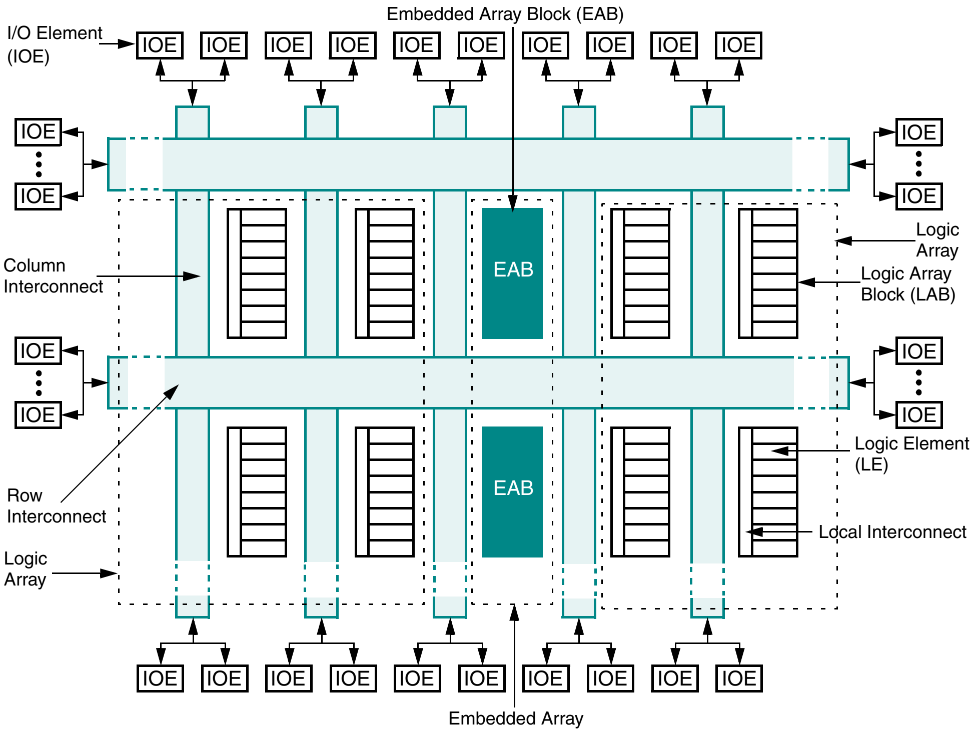
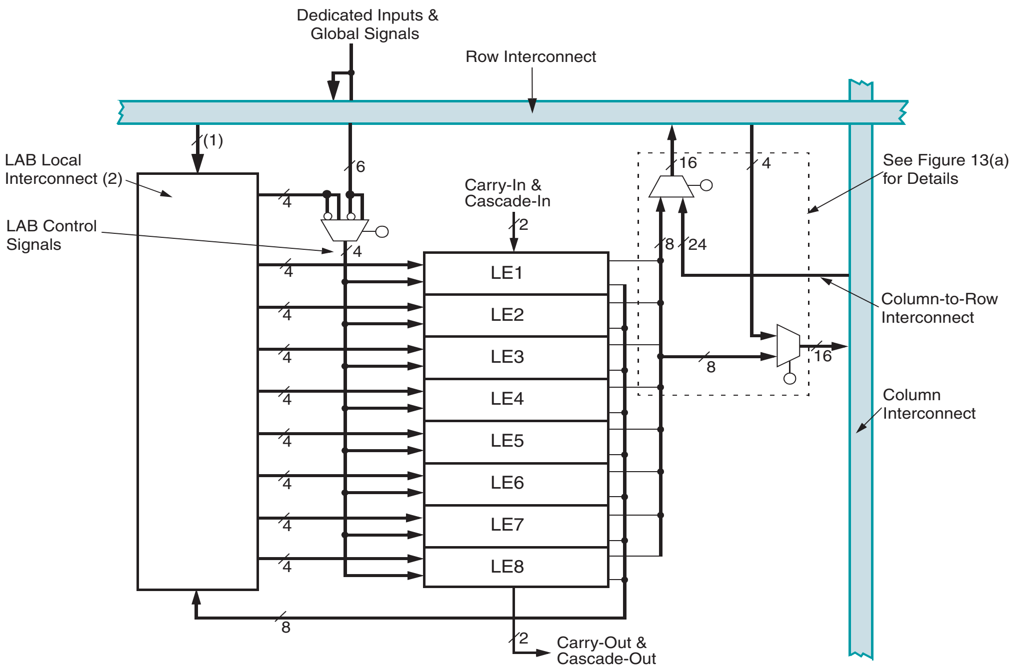


FIGURE 2
Altera Flex 10K EPLD Structure (Reprinted with permission of Altera Corporation. © 2004 Altera Corporation)

Logic

The logic circuits in the Altera Flex 10K EPLD lie within the LABs and the IOEs. Both of these structures are internally programmable and fairly complex. We will look in detail at the LAB and then sketch the main features of the IOE.

Logic Array Block (LAB) A diagram of a LAB appears in Figure 3. The core of the LAB consists of eight logic elements (LEs), each of which contains an identical block of implementation logic. In order to interconnect the LEs to each other, LAB local interconnect is provided. In addition, there is a set of four signals that enter all of the LEs which are used for controlling the LE storage element. These four signals can be driven externally or from the LAB local interconnect. The outputs of the LEs can be attached to row interconnect above and the column interconnect to the right of LAB. Connections can also be made within the LAB from column interconnect to row interconnect and from row interconnect to column interconnect. Finally, there are a Carry-in and a Carry-out for implementing arithmetic functions and a Cascade-in and Cascade-out for implementing a limited class of functions with large numbers of inputs.



Notes:
(1) EPF10K10, EPF10K10A, EPF10K20, EPF10K30, EPF10K30A, EPF10K40, EPF10K50, and EPF10K50V devices have 22 inputs to the LAB local interconnect channel from the row; EPF10K70, EPF10K100, EPF10K100A, EPF10K130V, and EPF10K250A devices have 26.
(2) EPF10K10, EPF10K10A, EPF10K20, EPF10K30, EPF10K30A, EPF10K40, EPF10K50, and EPF10K50V devices have 30 LAB local interconnect channels; EPF10K70, EPF10K100, EPF10K100A, EPF10K130V, and EPF10K250A devices have 34.

FIGURE 3
Diagram of an Altera Flex 10K Logic Array Block (LAB) (Reprinted with permission of Altera Corporation. © 2004 Altera Corporation)

Logic Element (LE) A diagram of the Flex 10K LE appears in Figure 4. A lookup table (LUT) is provided for implementing a 4-input, 1-output combinational function. The lookup table is attached to Carry Chain block which also has a Carry-in from the LE above it and a Carry-out to the LE below it. The combination of the LUT and the Carry Chain block provides two 3-input functions that implement the sum and carry functions of a full-adder. In the Cascade Chain block, the LUT output is ANDed with Cascade-in.

Each LE contains a storage element that can be configured to be either an edge-triggered D flip-flop or a level-sensitive latch. The D input is driven by a multiplexer that can be programmed to select between the output of the Cascade Chain block and the LE input data, permitting the storage element to be used with or independently of the LE combinational logic. The clock for the storage element can be selected by a programmed multiplexer from two control inputs. The storage

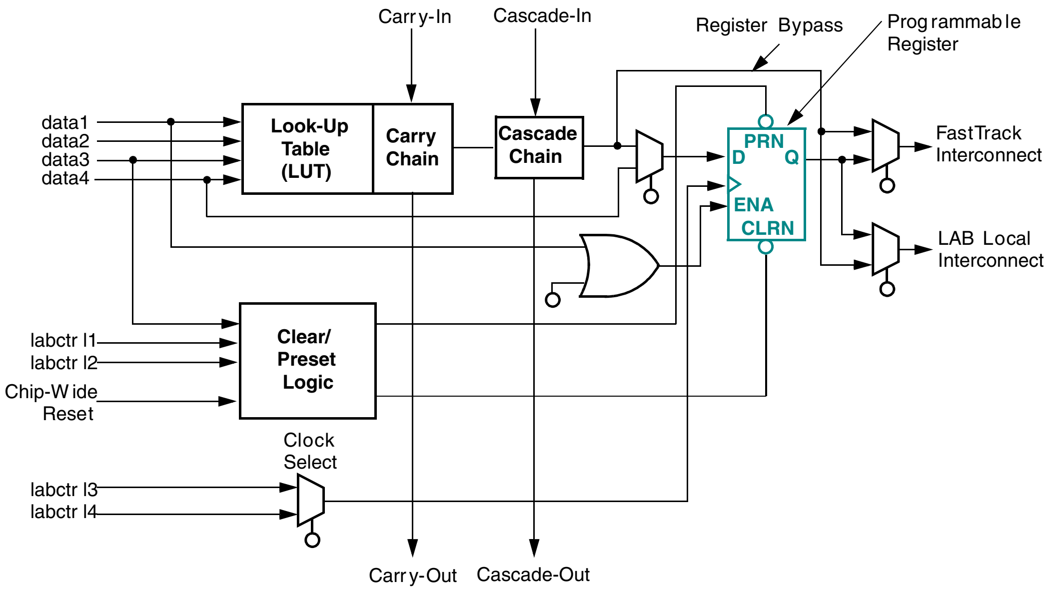


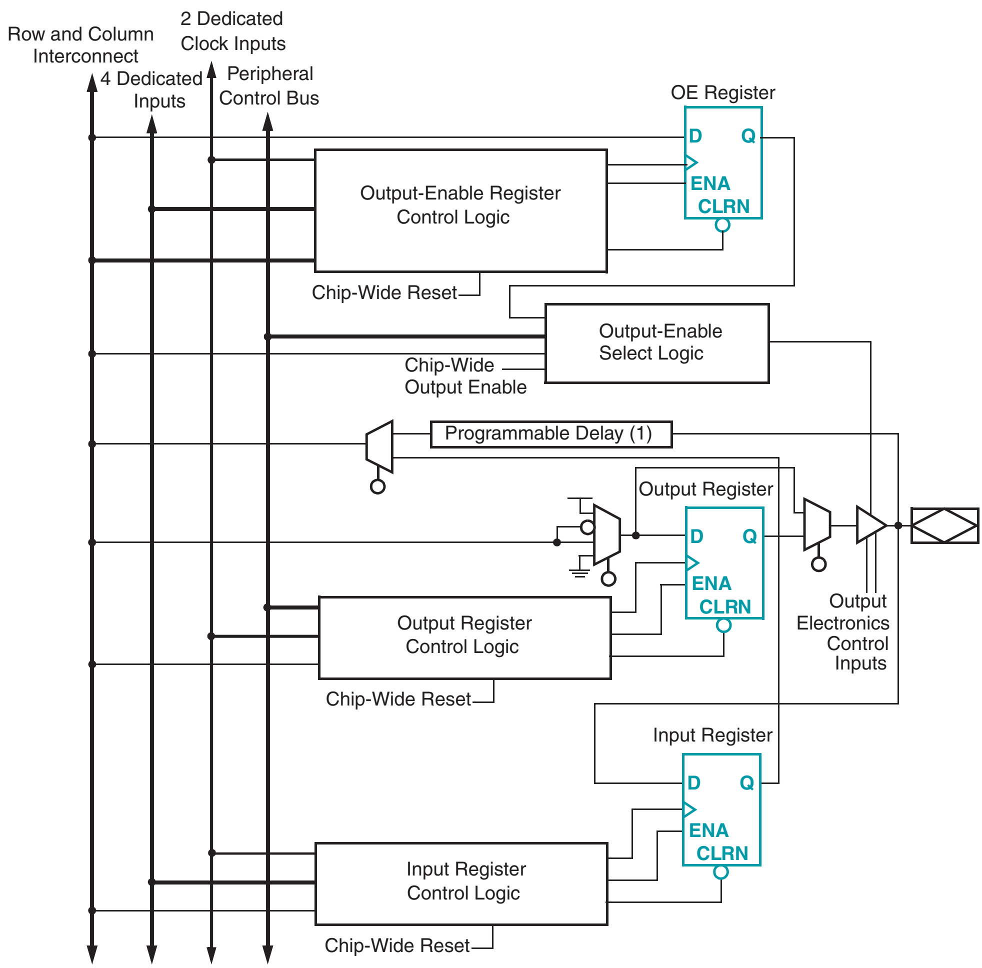
FIGURE 4
Diagram of an Altera Flex 10K Logic Element (LE) (Reprinted with permission of Altera Corporation. © 2004 Altera Corporation)

element also has Clear and Preset inputs that are control by Clear/Preset logic having a mixture of data and control inputs.

Two of the outputs from the LE attach to the row and column interconnect channels (called FastTrack™ interconnect) and to the local interconnect respectively. Two outputs are used to allow independent selection using programmed multiplexers from two sources. One source, an output of the Cascade Chain, is combinational and the other, the output of the storage element, is sequential. The other two outputs of the LE are Carry-out and Cascade-out.

INPUT/OUTPUT ELEMENT (IOE) The Flex 10K Input/Output Element (IOE) shown in Figure 5 has its I/O pin driven by a three-state output buffer and permits the signal on the pin to be used as an input. This permits the pin to be configured as an output, an input, or a bidirectional connection. There are three primary internal signals associated with I/Os: 1) the input signal from the I/O pin, 2) the output signal that goes to the 3-state buffer, and 3) the output enable signal for the 3-state buffer. Each of these signals may be optionally driven by a local storage register or may be driven by signals from elsewhere within the EPLD. The storage registers each have a clock, a load enable input, and a direct clear input.

The combinational logic within an IOE selects 1) whether or not each of these three internal signals is driven by a local storage register or by a signal from elsewhere and 2) selects the sources for the signals that control the local storage registers. A simplified version of this logic appears in Figure 5. The inputs to this logic includes data signals, control signals, and SRAM-configuration bits. Begin-



Note:
(1) All FLEX 10KE devices (except the EPF10K50E and EPF10K200E devices) have a programmable input delay buffer on the input path.

FIGURE 5
Simplified Diagram of an Altera Flex 10K Input/Output Element (IOE) (Adapted with permission of Altera Corporation. © 2004 Altera Corporation)

ning at the input signal from the I/O pin, there are two paths to the row or column interconnect selected by a multiplexer controlled by an SRAM bit. One of the multiplexer inputs is the input signal itself. For some parts in the 10K family and most of the parts within the 10KE family, there is a programmable delay available in this path that is used to delay the input so that the hold time with respect to the clock is zero. The other multiplexer input is the output of the Input register. In addition, there is control logic for selection of the input register clock, enable, and clear signals. The inputs to this logic come from row and column interconnect, four dedicated inputs, two dedicated clock lines, and a constant 1. The selection process is controlled by SRAM bits. In addition, the result of the selection for the clear is ANDed with the Chip-wide Reset to produce the clear output signal from this logic.

Beginning at the input to the 3-state buffer, two multiplexers controlled by SRAM bits are used to select whether or not the output signal is stored in the output register and to select one of four signals to act as the output and the input to the output register. There four signals are a constant 1, a constant 0, and true and inverted versions of a signal from the row and column interconnect. In addition, there is control logic for selection of the output register clock, enable, and clear signals. The inputs to this logic come from the row or column interconnect, the peripheral control bus, two dedicated clock lines, and a constant 1. The selection process is controlled by SRAM bits. In addition, the result of the selection for the clear is ANDed with the Chip-wide Reset to produce the clear output signal from this logic. Finally, for the output control enable input to the 3-state buffer, there is selection logic controlled by SRAM bits for selecting from the row or column interconnect, the peripheral control bus, or the output of the OE register. The input of the OE register can come only from the row or column interconnect. Also, the Chip-wide Reset is ANDed with the result of the selection for the clear to produce the clear output signal from this logic. In addition, there is control logic for selection of the output register clock, enable, and clear signals. The inputs to this logic come from the row or column interconnect, 4-dedicated inputs, two dedicated clock lines, and a constant 1. The selection process is controlled by SRAM bits.

Embedded Array Block

The Flex 10K parts contain several Embedded Array Blocks (EABs). Each EAB in the 10K parts contains $2^{11} = 2{,}048$ bits of storage and in the 10KE parts contains $2^{12} = 4{,}096$ bits of storage. The 10K EABs can be configured to have 2^m words of 2^n bits with $m + n = 11$ and $0 \le n \le 3$ and the 10KE parts can be configured to have 2^m words of 2^n bits with $m + n = 12$ and $1 \le n \le 4$. The EABs can be configured as a number of useful functional blocks for system design. The simplest is as a ROM, effectively using the EAB as a large LUT. A second configuration is as a RAM which has a single port consisting of control, address, and data inputs and one data output. A dual-port RAM configuration, which has two sets of control, address, data inputs and data outputs for reading and writing data, is available in the 10KE family. Data inputs and data outputs for a port can be merged into a single bidirectional data input/output if desired. The clock signals, read and write enable signals, and use of registers on inputs and outputs is flexible and programmable.

FastTrack™ Interconnections

The row interconnection channels are divided into two groups, full-length channels and half-length channels. A full length channel can be connected to all LABs in a row. A half-length channel can be connected to all LABs in the left half or right

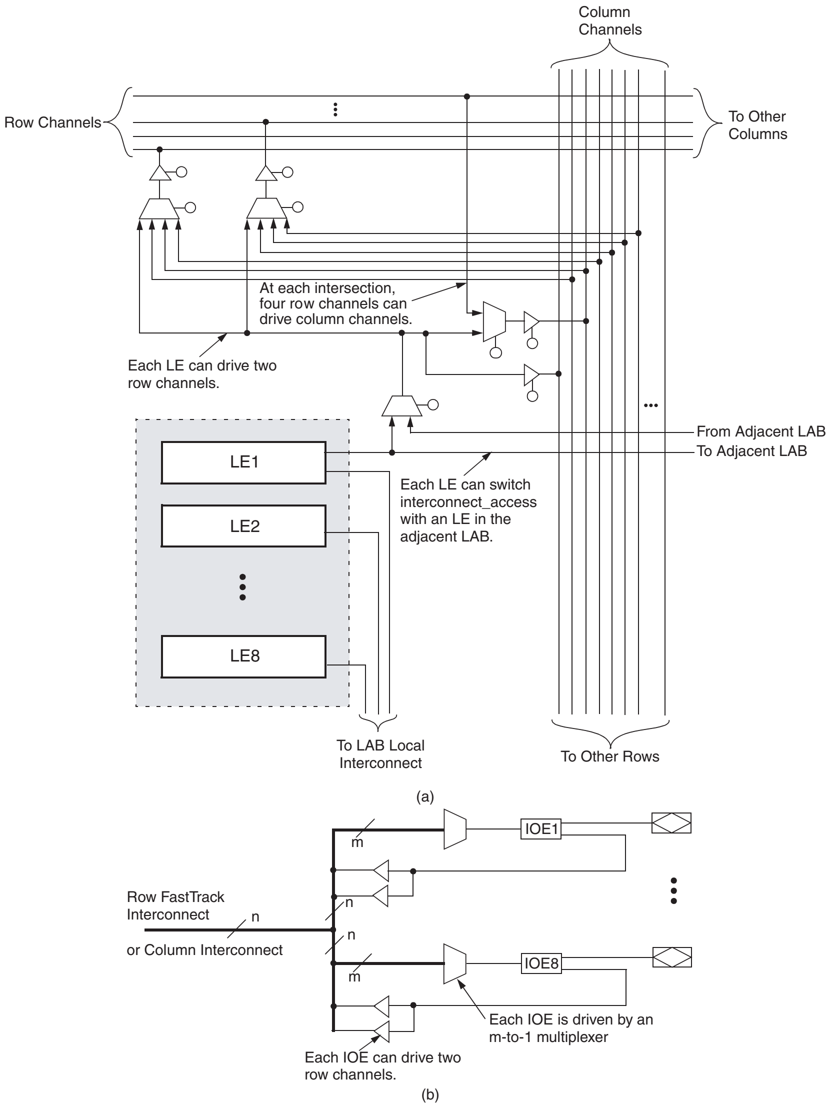


FIGURE 6

Diagrams of Altera Flex 10K Interconnect (Reprinted with permission of Altera Corporation. © 2004 Altera Corporation)

half of the array. Two diagrams showing the connections associated with the row and column interconnection structure are given in Figure 6.

The programmable connections between an LE, an interconnect row and an interconnect column all lie within a LAB are shown in Figure 6(a). The LE output can be programmed to from zero to two row channels and to from zero to two column channels. In addition, for flexibility in routing signals on the channels, the same connections for an LE can be made from the adjacent LAB. If the connections are to be made for the corresponding LE to channels, then due to the limits of the structure shown, the connection access can only be switched and not mixed between the two LABs. The connections to the row channels for the LE are shared with connections from column channels to row channels as shown. Likewise, one of the connections from the LE to the column channels is shared with a connection from a row channel. In each LAB, this overall structure appears eight times, once for each of the eight LEs. The inputs to the LAB from the row channels are shown in Figure 6. There are from 22 to 26 such signals entering the LAB Local interconnect depending on the size of the particular FLEX 10K part.

The programmable connections between an interconnect row and IOEs are shown in Figure 6(b). An IOE can drive two separate row channels as shown. Each IOE is driven by a multiplexer that selects from a subset of the row channels. The structure for connections between an interconnect column and IOEs is similar except that there are only two IOEs at each end of a column.

Design Methodology

The overall structure of the interconnections, LABs, and LEs, IOEs, and EAB is complicated. A designer having to deal with 10's of LABs, hundreds of IOEs, and thousands of interconnection points in an EPLD has a very difficult job. As a consequence, CAD tools are provided that take a design in the form of a schematic or HDL description, automatically partition the design into pieces that fit within the LEs of a LAB, place the pieces into specific LEs and LABs and route the connections between the LABs. The end result of this process is thousands of bits of programming information that can be loaded into the EPLD to implement the desired logic.

References

1. Altera Corporation, FLEX 10K Embedded Programmable Logic Device Family Data Sheet, January 2003, version 4.2 (http://www.altera.com/literature/ds/dsf10k.pdf).
2. Altera Corporation, FLEX 10KE Embedded Programmable Logic Device Data Sheet, January 2003, version 2.5 (http://www.altera.com/literature/ds/dsf10ke.pdf).