(12) UK Patent Application (19) GB (11) 2 351 824 (13) A

(43) Date of A Publication **10.01.2001**

(21) Application No **0016223.0**

(22) Date of Filing **30.06.2000**

(30) Priority Data
(31) **60142141** (32) **02.07.1999** (33) **US**

(51) INT CL^7
H03K 19/177

(52) UK CL (Edition S)
G4H HU H13D H14A

(56) Documents Cited
WO 98/10517 A1 **WO 97/30517 A1**

(58) Field of Search
UK CL (Edition R) **G4H HU**
INT CL^7 **H03K**

(71) Applicant(s)
Altera Corporation
(Incorporated in USA - Delaware)
101 Innovation Drive, San Jose, California 95134,
United States of America

(72) Inventor(s)
Tony Ngai
Sergey Shumarayev
Wei-Jen Huang
Rakesh H Patel
Tin Lai

(74) Agent and/or Address for Service
Haseltine Lake & Co
Imperial House, 15-19 Kingsway, LONDON,
WC2B 6UD, United Kingdom

(54) Abstract Title
Programmable logic with embedded memory blocks

(57) A high-performance programmable logic architecture has embedded memory arranged at peripheries or edges of the integrated circuit. This enhances the performance of the programmable logic integrated circuit by shortening the lengths of the programmable interconnect (748). In a specific embodiment, the memory blocks (703,706) are organized in rows along the top and bottom edges of the integrated circuit. The logic elements can be directly programmable routed and connected to driver blocks of the logic blocks in adjacent rows and columns. This permits fast interconnection of signals without using the global programmable interconnect resources. Using similar direct programmable interconnections the logic blocks can directly programmable connect to the memory blocks without using the global programmable interconnect resources. The present invention also provides technique of flexibly combining or stitching multiple memories together to form memories of a desired size.

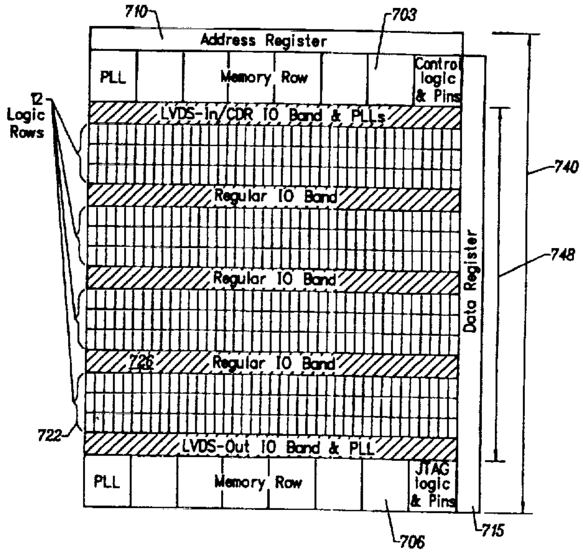


FIG. 7

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

GB 2 351 824 A

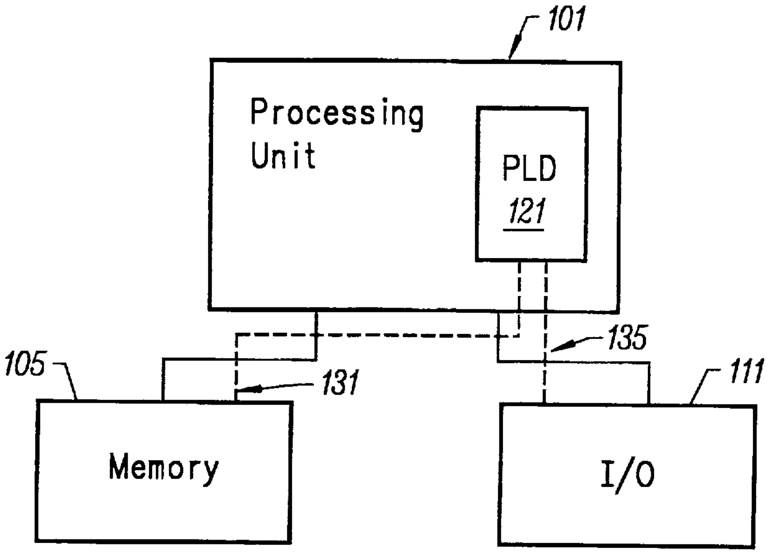

101
Processing Unit
PLD
121
105
131
135
111
Memory
I/O

FIG. 1

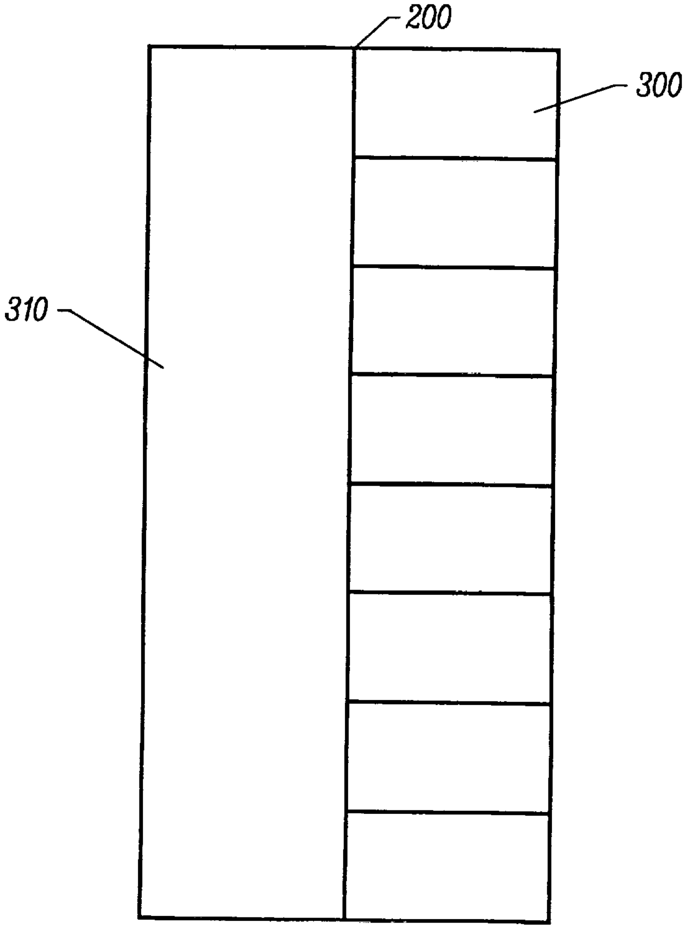

200
300
310

FIG. 3

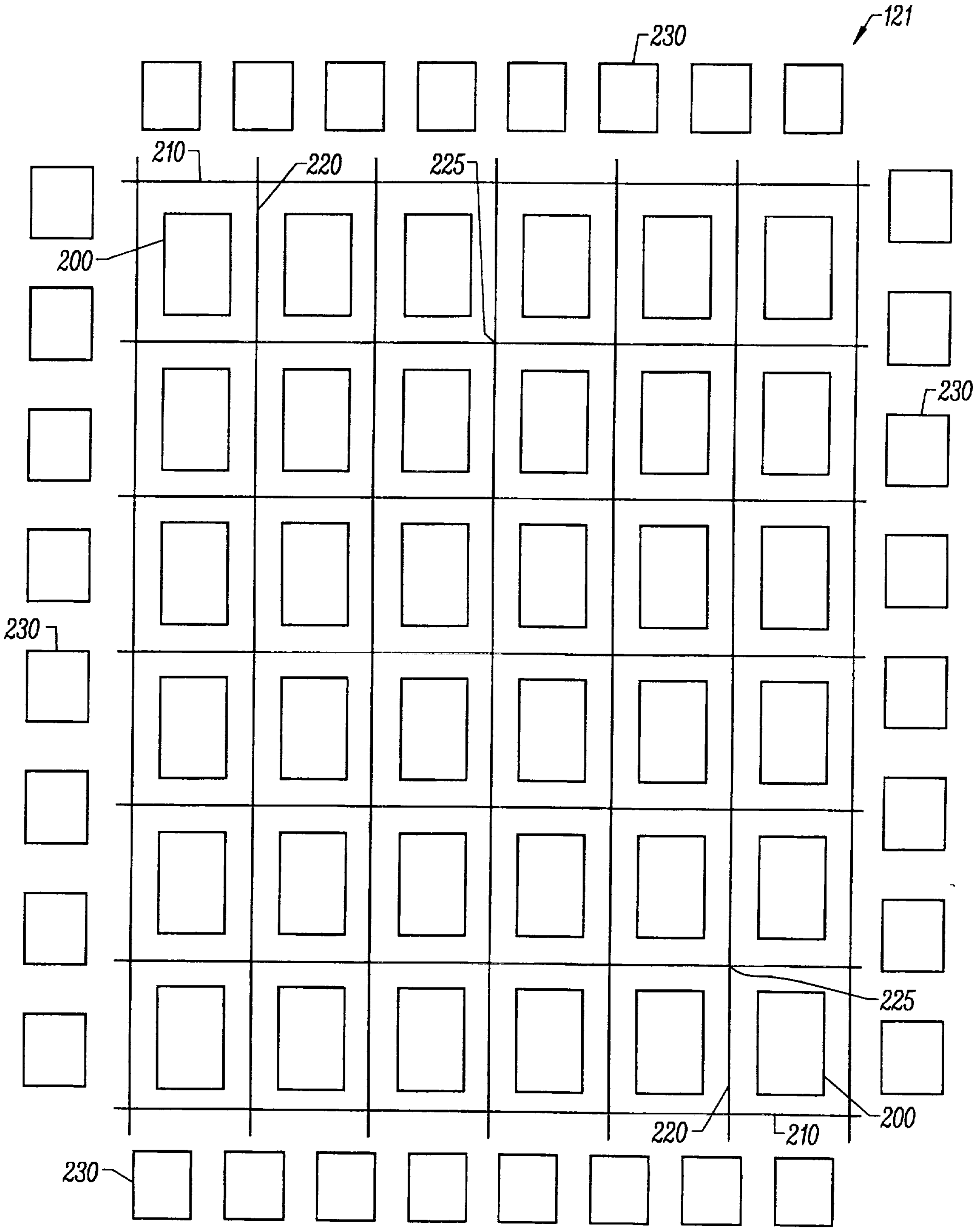


FIG. 2

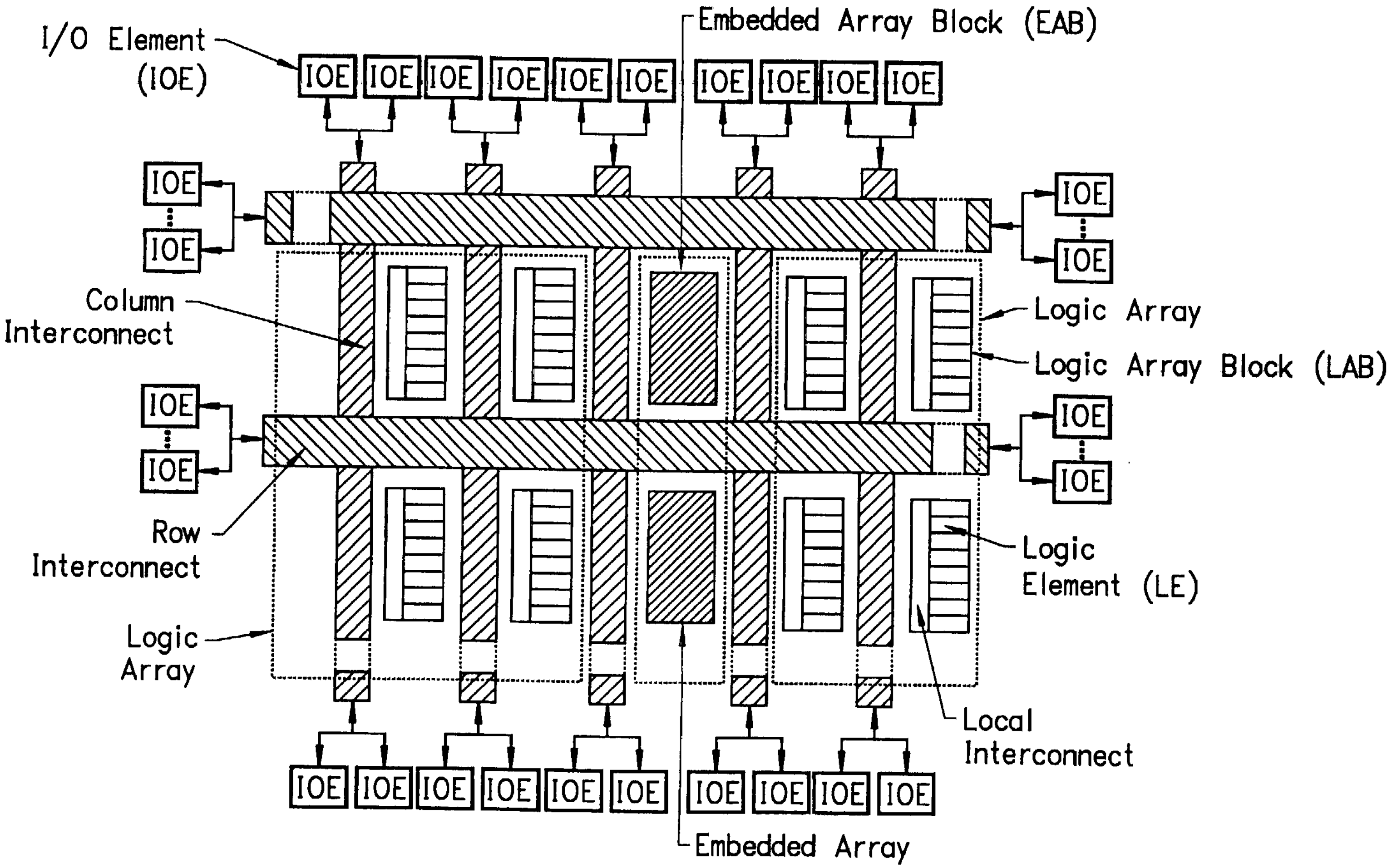

I/O Element (IOE)
IOE
Embedded Array Block (EAB)
Column Interconnect
Row Interconnect
Logic Array
Logic Array
Logic Array Block (LAB)
Logic Element (LE)
Local Interconnect
Embedded Array

FIG. 4

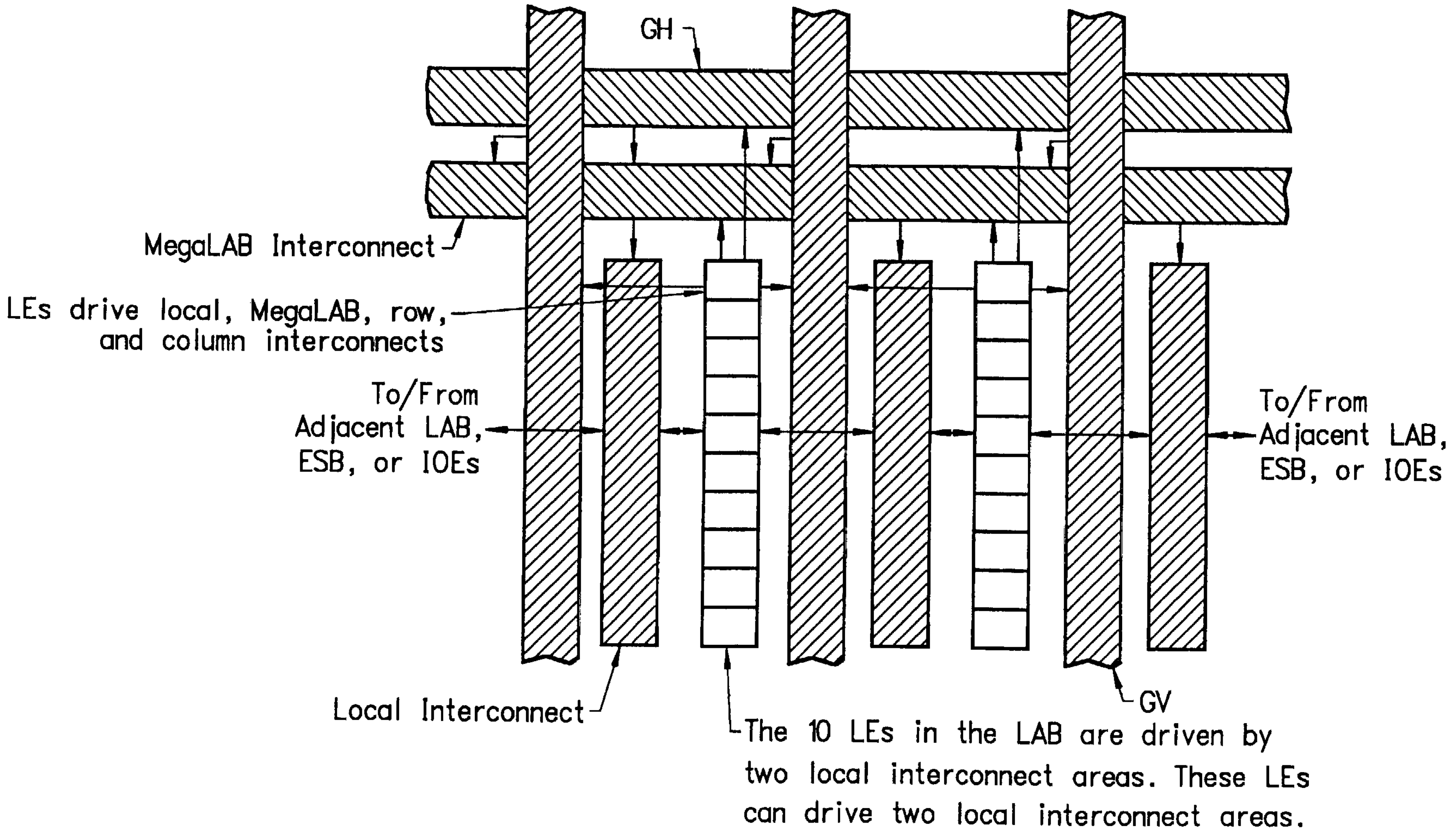

GH
MegaLAB Interconnect
LEs drive local, MegaLAB, row, and column interconnects
To/From Adjacent LAB, ESB, or IOEs
To/From Adjacent LAB, ESB, or IOEs
Local Interconnect
GV
The 10 LEs in the LAB are driven by two local interconnect areas. These LEs can drive two local interconnect areas.

FIG. 5

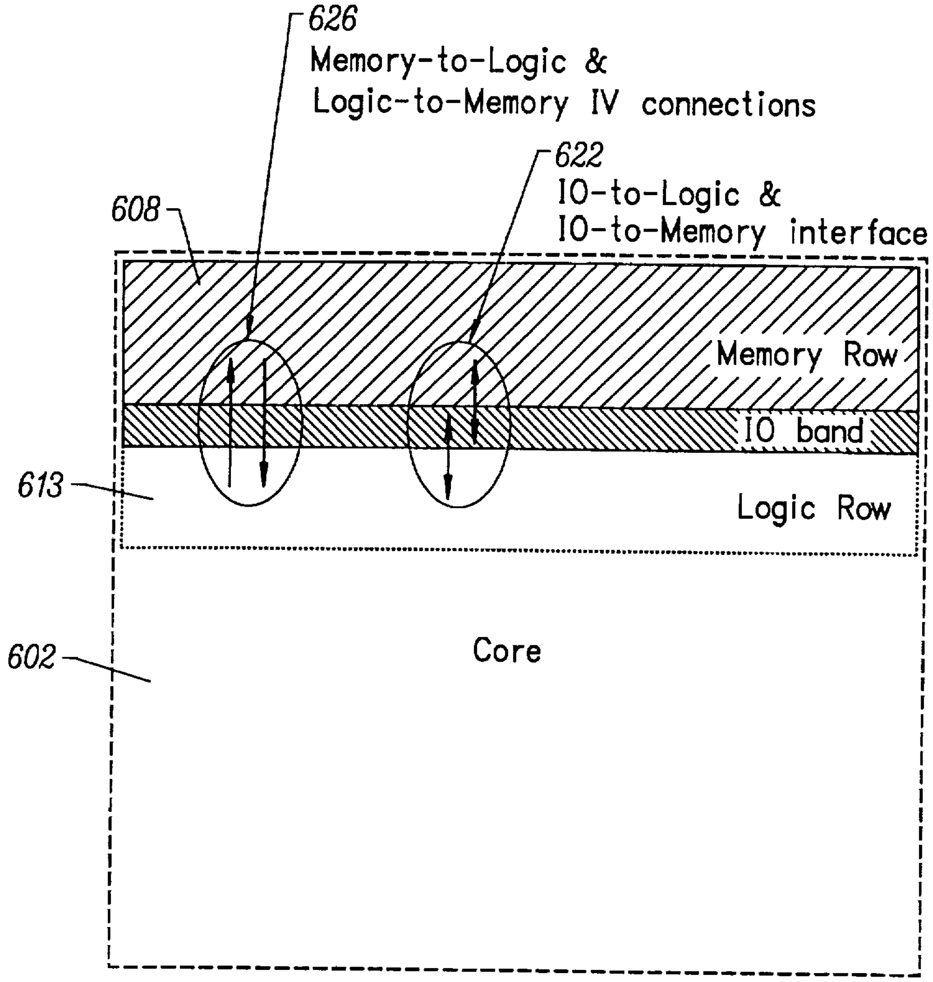

626
Memory-to-Logic &
Logic-to-Memory IV connections
622
IO-to-Logic &
IO-to-Memory interface
608
Memory Row
IO band
613
Logic Row
602
Core

FIG. 6

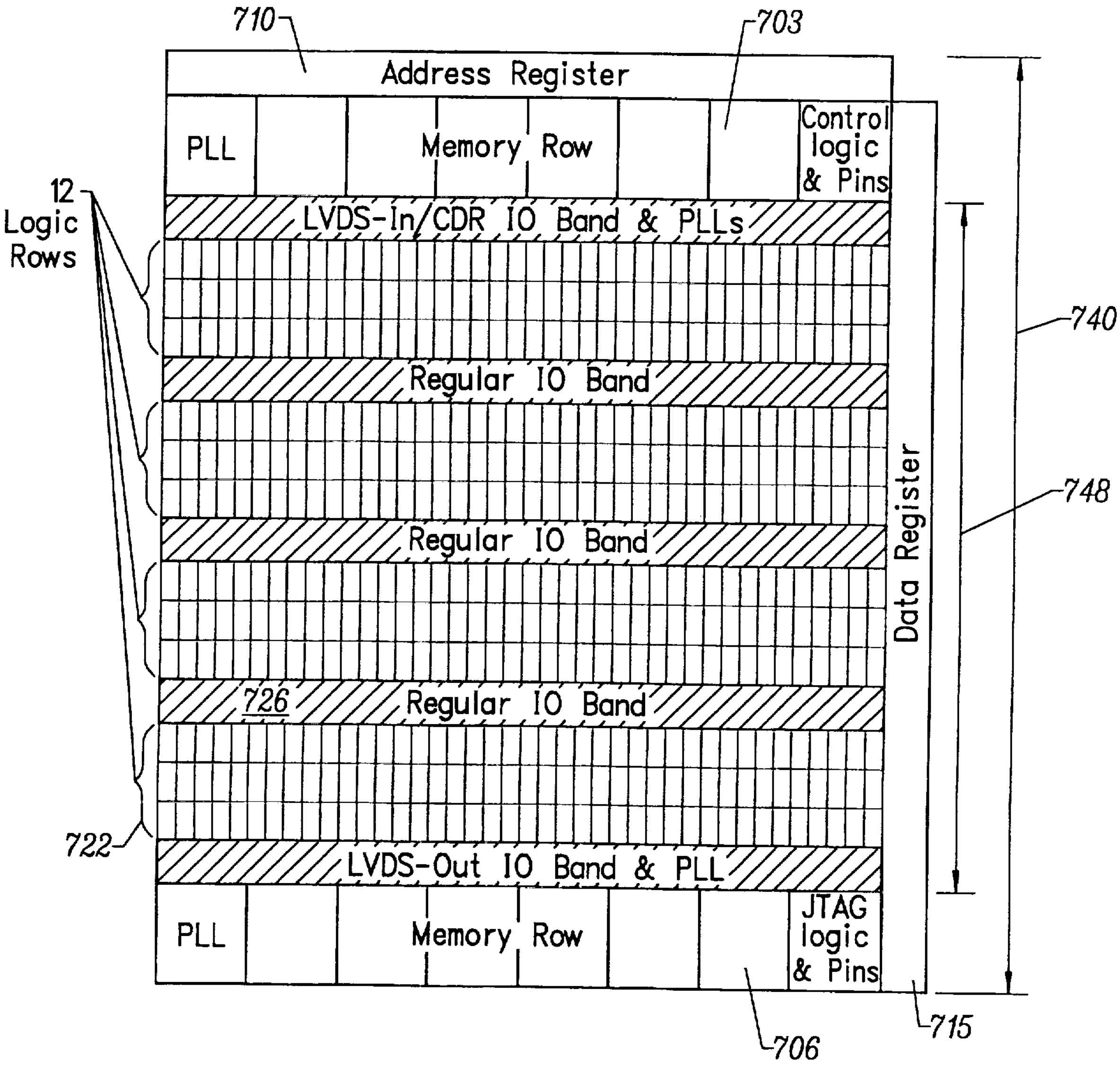
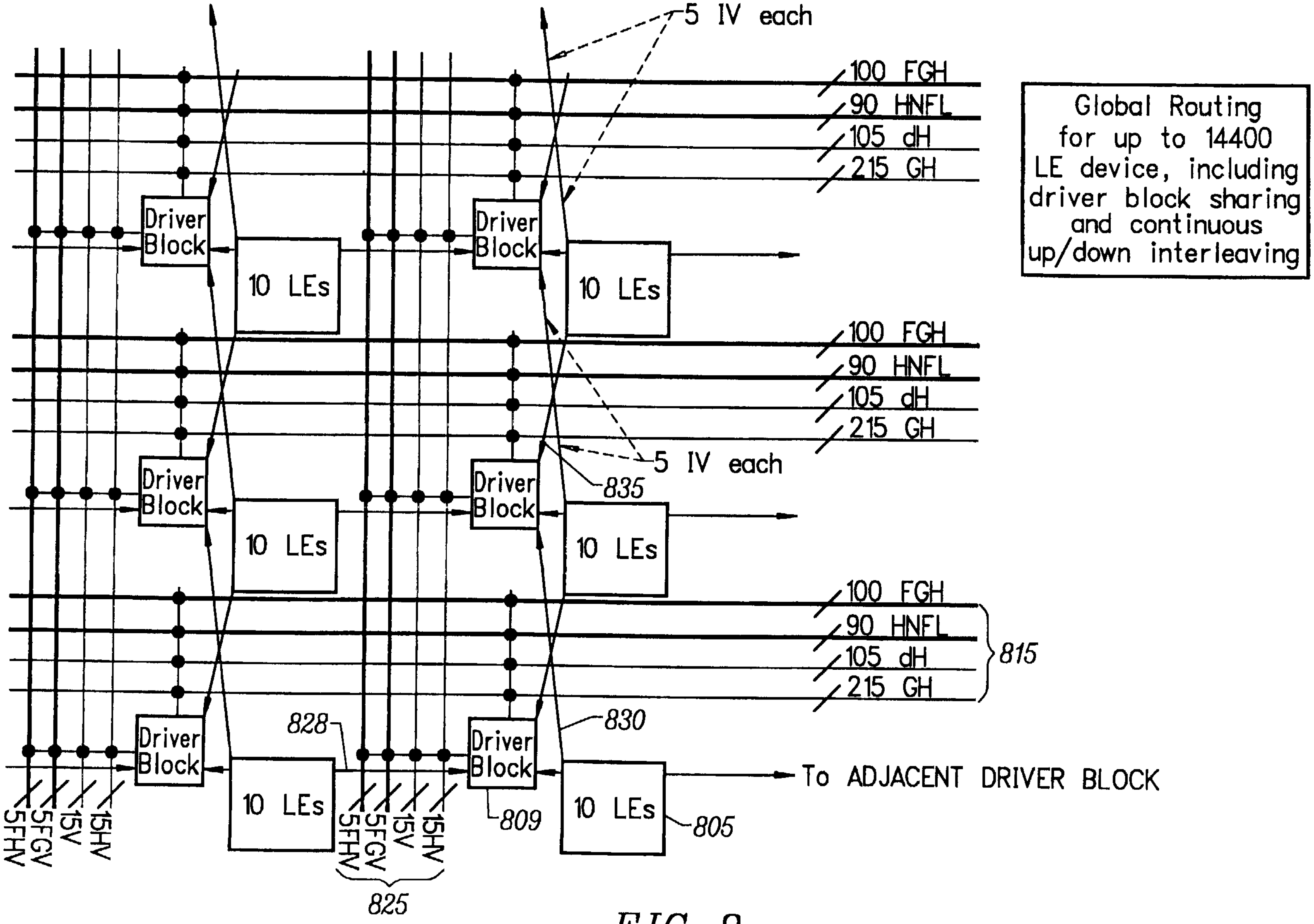

710
703
Address Register
PLL
Memory Row
Control logic & Pins
12 Logic Rows
LVDS-In/CDR IO Band & PLLs
Regular IO Band
Regular IO Band
726
Regular IO Band
740
748
Data Register
722
LVDS-Out IO Band & PLL
PLL
Memory Row
JTAG logic & Pins
706
715

FIG. 7



Global Routing for up to 14400 LE device, including driver block sharing and continuous up/down interleaving

FIG. 8

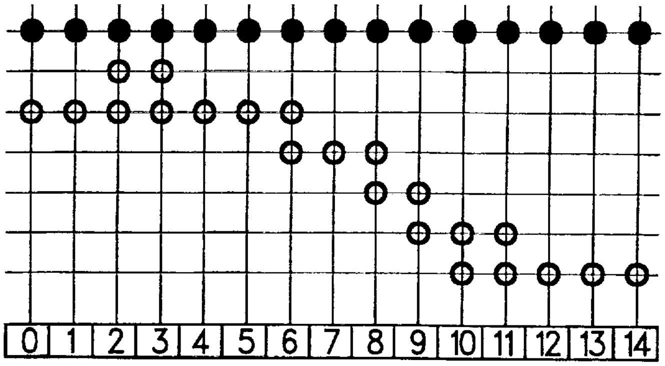


FIG. 9

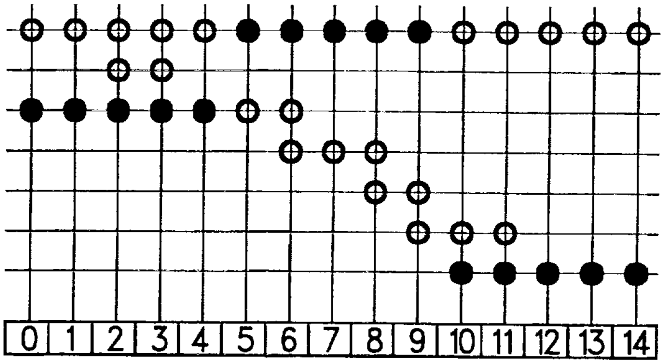


FIG. 10

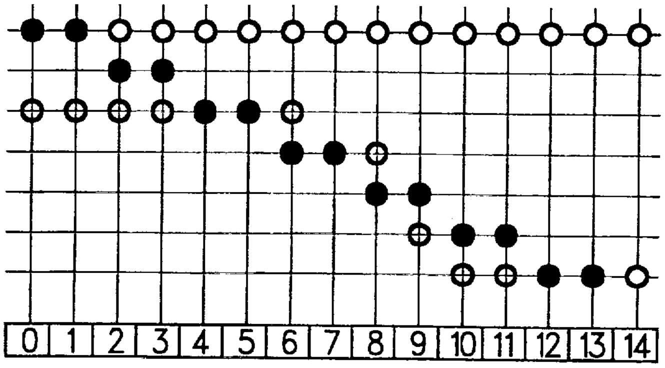


FIG. 11

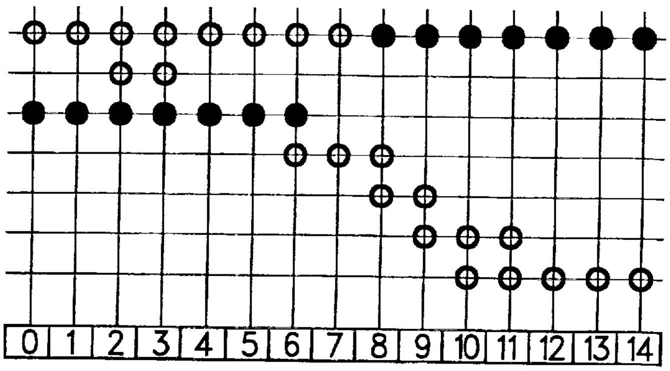


FIG. 12

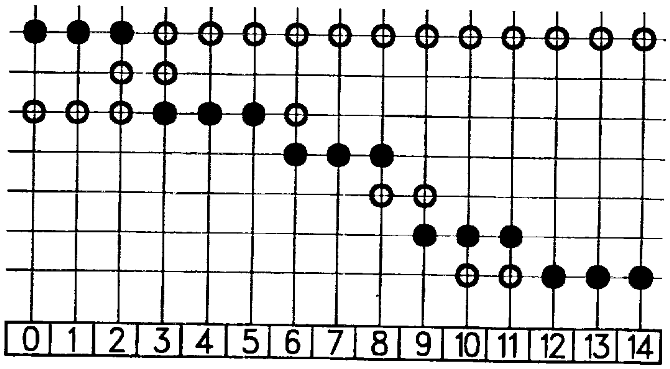


FIG. 13

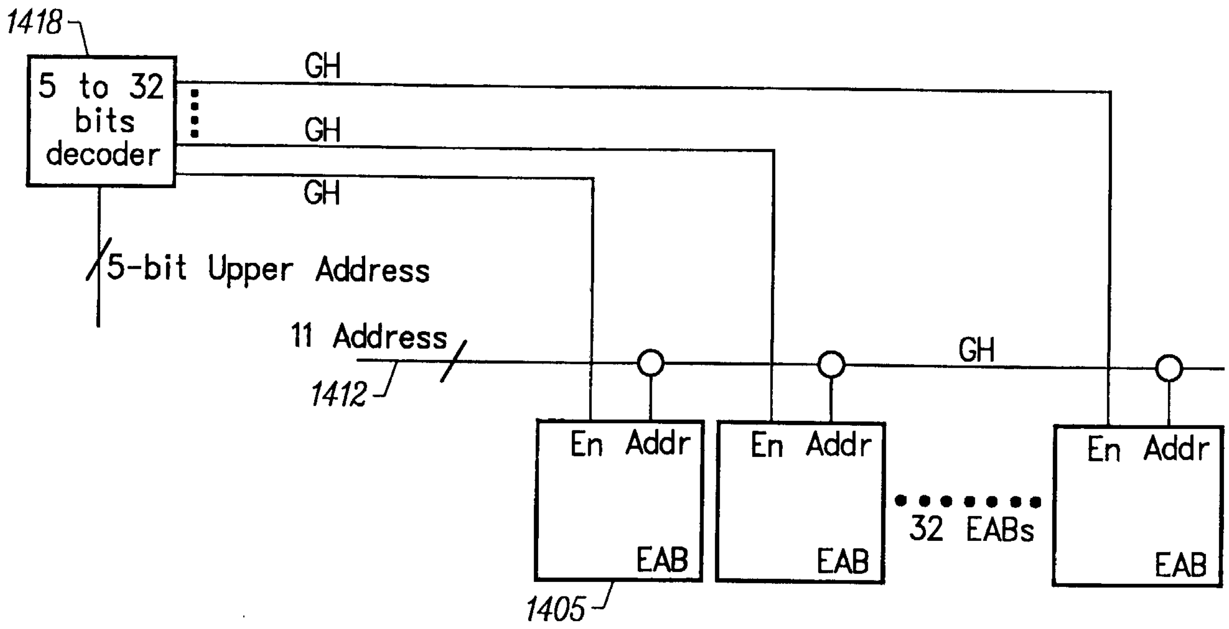

1418
5 to 32
bits
decoder
GH
GH
GH
5-bit Upper Address
11 Address
1412
GH
En Addr
EAB
En Addr
EAB
32 EABs
En Addr
EAB
1405

FIG. 14

2351824

EMBEDDED MEMORY BLOCKS FOR PROGRAMMABLE LOGIC

CROSS-REFERENCES TO RELATED APPLICATIONS

This application claims the benefit of U.S. provisional application 60/142,141, filed July 2, 1999, which is incorporated by reference along with all references cited in this application.

BACKGROUND OF THE INVENTION

The present invention relates to the field of integrated circuits and in particular, to architectures for memory within a programmable logic integrated circuit to improve performance.

Programmable logic integrated circuits such as PALs, PLDs, FPGAs, LCAs, and others are becoming more complex and continually evolving to provide more user-programmable features on a single integrated circuit. Modern programmable logic integrated circuits incorporate programmable logic including logic gates, products terms, or look-up tables. Programmable logic integrated circuits also included embedded user-programmable memory or RAM.

Despite the success of programmable logic, there is a continuing desire to provide greater functionality in a programmable logic integrated circuit, but at the same time, provide greater performance. There is a need to provide more higher performance user memories. Larger memory sizes or providing more memory blocks typically negatively impacts the performance of the integrated circuit because the integrated circuit is larger. There is programmable interconnect (e.g., GH and GV lines) will usually be longer and there will be greater capacitance. This reduces performance. It is desirable to have a high-speed interface and interconnections between the logic blocks and the memory blocks. Further, it is desirable to provide flexible stitching together or combining of the user memories in space-efficient and high performance manner.

Therefore, there is a need to provide high performance address decoding techniques and circuitry in order to improve the performance of the integrated circuit.

SUMMARY OF THE INVENTION

The present invention provides a high-performance programmable logic architecture with embedded memory. Memory blocks are arranged at the peripheries or edges of the integrated circuit so in order to shorten the lengths of the programmable interconnect. In a specific embodiment, the memory blocks are organized in memory block rows along the top and bottom edges of the integrated circuit. The memory blocks may also be organized in columns at the left and right edges for similar benefits. Each logic block is separated into a driver block and logic elements. Flip chip technology is used so IO bands for the integrated circuit are within the core of the integrated circuit; there are no IO bands between the memory blocks and the edges of the integrated circuit. The logic elements can be directly routed and connected to driver blocks and programmable routing resources of the logic block in adjacent rows and columns. This permits fast interconnection of signals without using the global programmable interconnect resources. Using similar direct programmable interconnections, the logic blocks can directly connect to the programmable routing resources of the memory blocks without using the global programmable interconnect resources. The present invention also provides technique of flexibly combining or stitching multiple memories together to form memories of a desired size.

In one embodiment, the invention is a programmable logic integrated circuit including a number of programmable interconnect lines arranged in rows and columns. There are a number of logic array blocks between rows and columns of the programmable interconnect lines, where the logic array blocks are arranged in rows and columns and programmably connected to the programmable interconnect lines, and the logic array blocks are programmably configurable to perform logical functions. A number of embedded array blocks are between rows and columns of the programmable interconnect lines, where the embedded array blocks are arranged in a row and each embedded array block includes a RAM block. The RAM block includes at least 2048 bits of memory. The logic array blocks include logic elements. Two embedded array blocks of a row can be programmably combined to provide a larger memory than available with a single embedded array block. A row of embedded array blocks comprises thirty-two embedded array blocks. A row of embedded array blocks is programmably configurable to be one 30Kx1 memory, two 14Kx1 memories, three 10Kx1 memories, five 6Kx1 memories, or seven 4Kx1 memories.

The programmable logic integrated circuit may further include a decoder connected to receive a first address and generating a plurality of output control lines, each connected to an enable input one of the embedded array blocks. The output control lines are programmable interconnect lines. Some of the programmable interconnect lines are coupled to provide a second address-to-address inputs of the embedded array blocks. The first address comprises five bits. The second address comprises eleven bits.

In another embodiment, the invention is a programmable logic integrated circuit including a top row of memory blocks formed along a top edge of the integrated circuit and a bottom row of memory blocks formed along a bottom edge of the integrated circuit. A number of logic blocks are arranged in rows and columns and are programmably configurable to implement logic functions. The logic blocks are formed between the top and bottom rows of memory blocks. There are IO bands between the rows of memory blocks. In a specific flip chip implementation, there are no IO bands between the memory blocks and the edge of the integrated circuit. A number of horizontal programmable interconnect lines are programmably connected to the logic blocks. A number of vertical programmable interconnect lines are programmably connected to the logic array blocks. In a specific implementation, the vertical programmable interconnect lines do not extend an entire length from the top to bottom rows of the memory block. In particular, the vertical programmable interconnect lines extend from an inside edge of the top row of memory blocks to an inside edge of the bottom row of memory blocks.

The top row of memory blocks comprises six blocks or more of memory which can be programmably connected together to form larger memory blocks. The logic blocks are grouped into sections of logic blocks and between each section is an IO band. An IO band includes IO pads and input and output circuitry. Each section has three rows of logic blocks and at least forty columns of logic blocks. The programmable logic integrated circuit further includes a phase locked loop circuit formed along with the top row of memory blocks.

A logic block includes a number of logic elements and a driver block, where the number of logic elements is directly programmably connected to a driver block or programmable routing resources of an adjacent logic array block in the same row, without using the horizontal or vertical programmable interconnect. A logic block includes a number of logic elements and a driver block, where the number of logic elements is directly programmably connected to a driver block in an adjacent logic array block in an above row, without using the horizontal or vertical programmable interconnect. A logic block includes a number of logic elements and a driver block, where the number of logic elements is directly programmably connected to a memory block and in particular, the programmable routing resources, in the top row, without using the horizontal or vertical programmable interconnect.

Other objects, features, and advantages of the present invention will become apparent upon consideration of the following detailed description and the accompanying drawings, in which like reference designations represent like features throughout the figures.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is diagram of a digital system with a programmable logic integrated circuit.

Figure 2 is a diagram showing an architecture of a programmable logic integrated circuit.

Figure 3 is a simplified block diagram of a logic array block (LAB).

Figure 4 shows an architecture of a programmable logic integrated circuit with embedded array blocks (EABs).

Figure 5 shows an architecture of a programmable logic integrated circuit with megaLABs.

Figure 6 shows a memory row and logic row of a programmable logic architecture having rows of memory blocks.

Figure 7 shows a overall diagram of a programmable logic integrated circuit with memory blocks arranged in rows and IO bands between the memory rows.

Figure 8 shows a more detailed diagram of the programmable interconnect routing and interleaving vertical lines of the programmable logic integrated circuit.

Figure 9 shows a partially populated stitching pattern to form one 30K memory.

Figure 10 shows a partially populated stitching pattern to form three 10K memories.

Figure 11 shows a partially populated stitching pattern to form seven 4K memories.

Figure 12 shows a partially populated stitching pattern to form two 15K memories.

Figure 13 shows a partially populated stitching pattern to form five 6K memories.

Figure 14 shows a decoder circuit scheme for forming deep memories from multiple memory blocks.

DETAILED DESCRIPTION

Figure 1 shows a block diagram of a digital system within which the present invention may be embodied. The system may be provided on a single board, on multiple boards, or even within multiple enclosures. Figure 1 illustrates a system 101 in which a programmable logic device 121 may be utilized. Programmable logic devices are sometimes referred to as a PALs, PLAs, FPLAs, PLDs, CPLDs, EPLDs, EEPLDs, LCAs, or FPGAs and are well-known integrated circuits that provide the advantages of fixed integrated circuits with the flexibility of custom integrated circuits. Such devices allow a user to electrically program standard, off-the-shelf logic elements to meet a user's specific needs. See, for example, U.S. patent number 4,617,479, incorporated by reference for all purposes. Programmable logic devices are currently represented by, for example, Altera's MAX®, FLEX®, and APEX™ series of PLDs. These are described in, for example, U.S. patent numbers 4,871,930, 5,241,224, 5,258,668, 5,260,610, 5,260,611, 5,436,575, and the *Altera Data Book* (1999), all incorporated by reference in their entirety for all purposes. Programmable logic integrated circuits and their operation are well known to those of skill in the art.

In the particular embodiment of Figure 1, a processing unit 101 is coupled to a memory 105 and an I/O 111 and incorporates a programmable logic device (PLD) 121. PLD 121 may be specially coupled to memory 105 through connection 131 and to I/O 111 through connection 135. The system may be a programmed digital computer system, digital signal processing system, specialized digital switching network, or other processing system. Moreover, such systems may be designed for a wide variety of applications such as, merely by way of example, telecommunications systems, automotive systems, control systems, consumer electronics, personal computers, and others.

Processing unit 101 may direct data to an appropriate system component for processing or storage, execute a program stored in memory 105 or input using I/O 111, or other similar function. Processing unit 101 may be a central processing unit (CPU), microprocessor, floating point coprocessor, graphics coprocessor, hardware controller, microcontroller, programmable logic device programmed for use as a controller, or other processing unit. Furthermore, in many embodiments, there is often no need for a CPU. For example, instead of a CPU, one or more PLDs 121 may control the logical operations of the system. In some embodiments, processing unit 101 may even be a computer system. Memory 105 may be a random access memory (RAM), read only memory (ROM), fixed or flexible disk media, PC Card flash disk memory, tape, or any other storage retrieval means, or any combination of these storage retrieval means. PLD 121 may serve many different purposes within the system in Figure 1. PLD 121 may be a logical building block of processing unit 101, supporting its internal and external operations. PLD 121 is programmed to implement the logical functions necessary to carry on its particular role in system operation.

Figure 2 is a simplified block diagram of an overall internal architecture and organization of PLD 121 of Figure 1. Many details of PLD architecture, organization, and circuit design are not necessary for an understanding of the present invention and such details are not shown in Figure 2.

Figure 2 shows a six-by-six two-dimensional array of thirty-six logic array blocks (LABs) 200. LAB 200 is a physically grouped set of logical resources that is configured or programmed to perform logical functions. The internal architecture of a LAB will be described in more detail below in connection with Figure 3. PLDs may contain any arbitrary number of LABs, more or less than shown in PLD 121 of Figure 2. Generally, in the future, as technology advances and improves, programmable logic devices with greater numbers of logic array blocks will undoubtedly be created. Furthermore, LABs 200 need not be organized in a square matrix or array; for example, the array may be organized in a five-by-seven or a twenty-by-seventy matrix of LABs.

LAB 200 has inputs and outputs (not shown) which may or may not be programmably connected to a global interconnect structure, comprising an array of global horizontal interconnects (GHs) 210 and global vertical interconnects (GVs) 220. Although shown as single lines in Figure 2, each GH 210 and GV 220 line may represent a plurality of signal conductors. The inputs and outputs of LAB 200 are programmably connectable to an adjacent GH 210 and an adjacent GV 220. Utilizing GH 210 and GV 220 interconnects, multiple LABs 200 may be connected and combined to implement larger, more complex logic functions than can be realized using a single LAB 200.

In one embodiment, GH 210 and GV 220 conductors may or may not be programmably connectable at intersections 225 of these conductors. Moreover, GH 210 and GV 220 conductors may make multiple connections to other GH 210 and GV 220 conductors. Various GH 210 and GV 220 conductors may be programmably connected together to create a signal path from a LAB 200 at one location on PLD 121 to another LAB 200 at another location on PLD 121. A signal may pass through a plurality of intersections 225. Furthermore, an output signal from one LAB 200 can be directed into the inputs of one or more LABs 200. Also, using the global interconnect, signals from a LAB 200 can be fed back into the same LAB 200. In specific embodiments of the present invention, only selected GH 210 conductors are programmably connectable to a selection of GV 220 conductors. Furthermore, in still further embodiments, GH 210 and GV 220 conductors may be specifically used for passing signal in a specific direction, such as input or output, but not both.

In other embodiments, the programmable logic integrated circuit may include special or segmented interconnect that is connected to a specific number of LABs and not necessarily an entire row or column of LABs. For example, the segmented interconnect may programmably connect two, three, four, five, or more LABs.

The PLD architecture in Figure 2 further shows at the peripheries of the chip, input-output drivers 230. Input-output drivers 230 are for interfacing the PLD to external, off-chip circuitry. Figure 2 shows thirty-two input-output drivers 230; however, a PLD may contain any number of input-output drivers, more or less than the number depicted. Each input-output driver 230 is configurable for use as an input driver, output driver, or bidirectional driver.

Figure 3 shows a simplified block diagram of LAB 200 of Figure 2. LAB 200 is comprised of a varying number of logic elements (LEs) 300, sometimes referred to as "logic cells," and a local (or internal) interconnect structure 310. LAB 200 has eight LEs 300, but LAB 200 may have any number of LEs, more or less than eight.

A general overview of LE 300 is presented here, sufficient to provide a basic understanding of the present invention. LE 300 is the smallest logical building block of a PLD. Signals external to the LAB, such as from GHs 210 and GVs 220, are programmably connected to LE 300 through local interconnect structure 310. In one embodiment, LE 300 of the present invention incorporates a function generator that is configurable to provide a logical function of a number of variables, such a four-variable Boolean operation. As well as combinatorial functions, LE 300 also provides support for sequential and registered functions using, for example, D flip-flops.

LE 300 provides combinatorial and registered outputs that are connectable to the GHs 210 and GVs 220, outside LAB 200. Furthermore, the outputs from LE 300 may be internally fed back into local interconnect structure 310; through local interconnect structure 310, an output from one LE 300 may be programmably connected to the inputs of other LEs 300, without using the global interconnect structure's GHs 210 and GVs 220. Local interconnect structure 310 allows short-distance interconnection of LEs, without utilizing the limited global resources, GHs 210 and GVs 220.

Figure 4 shows a PLD architecture similar to that in Figure 2. The architecture in Figure 4 further includes embedded array blocks (EABs). EABs contain user memory, a flexible block of RAM. More discussion of this architecture may be found in the *Altera Data Book* (1999) in the description of the FLEX 10K product family and also in U.S. patent number 5,550,782, which are incorporated by reference. In the FLEX 10K products, the memory blocks are arranged in columns, embedded within the array of LABs.

Figure 5 shows a further embodiment of a programmable logic integrated circuit architecture. Figure 5 only shows a portion of the architecture. The features shown in Figure 5 are repeated horizontally and vertically as needed to create a PLD of any desired size. In this architecture, a number of LABs are grouped together into a megaLAB. In a specific embodiment, a megaLAB has sixteen LABs, each of which has ten LEs. There can be any number of megaLABs per PLD. A megaLAB is programmably connected using a megaLAB interconnect. This megaLAB interconnect may be considered another interconnect level that is between the global interconnect and local interconnect levels. The megaLAB interconnect can be programmably connected to GVs, GHs, and the local interconnect of each LAB of the megaLAB. Compared to the architecture of Figure 2, this architecture has an additional level of interconnect, the megaLAB interconnect. Such an architecture is found in Altera's APEX™ family of products, which is described in detail in the *APEX 20K Programmably Logic Device Family Data Sheet* (August 1999), which is incorporated by reference. In a specific implementation, a megaLAB also includes an embedded system block (ESB) to implement a variety of memory functions such as CAM, RAM, dual-port RAM, ROM, and FIFO functions. ESBs provide even greater functionality than EABs.

Programmable logic integrated circuits are types of integrated circuits. Programmable logic integrated circuits are sometimes referred to as programmable logic devices (PLDs) or field programmable gate arrays (FPGAs). There is further discussion of programmable logic in the *Altera 1999 Device Data Book*, which is incorporated by reference.

One particular design for a programmable logic integrated circuit is Altera's FLEX® 10K product line, which is described in Altera's data book. This programmable logic architecture includes logic array blocks (LABs) and embedded or enhanced array blocks (EABs). U.S. patent 5,550,782 describes a product with programmable logic and user memory and this pioneering patent is incorporated by reference. The LABs contain logic that is configurable by the user to implement whatever logic functions desired by the user. By using multiple LABs and linking them together, larger and more complex logic functions can be built. The EABs contain memory that can be used for storage of data. For example, the EAB can be used like a RAM or FIFO. The EAB memory can be integrated with the logical functionality programmed by the user. The principles of the present invention applies to programmable logic integrated circuits with EABs or ESBs.

In researching and studying the use of a PLD with embedded memory (or EABs) such as the FLEX® 10K, it was found that in many applications, multiple EABs are often "stitched" together to form larger memories. In typical PLD applications, there tends to be very few independent EAB blocks. And, when using stitched together EABs as logic (in p-term or product term mode), a significant part of the delay is spent on connections between the EAB blocks.

It was found that arranging the EABs in rows will improve performance. An EAB row arrangement came as a result of various investigations: study of possible speed-ups of row horizontal resources; analysis of EAB arrangement in the APEX family; study of components of critical path; and flip chip investigation.

Figure 6 shows a programmable logic (PLD) architecture with rows of memory blocks (e.g., EABs, ESBs). Memory blocks, EABs, and ESBs may be used interchangeably depending on the particular implementation. A row of memory blocks is shown at the edge of a LAB core, and at an edge of the integrated circuit. The figure shows a memory row, an IO (or I/O) band, a logic row, and a core. The IO band is for memory-to-logic and logic-to-memory interleaving vertical line connections. The IO band is the IO-to-logic and IO-to-memory interface.

Core 602 contains rows and columns of LABs and programmable interconnect (such as GVs and GHs), similar as described for the PLD architecture in Figure 2. A row of memory blocks 608 is placed adjacent a vertical or row edge of the LAB core. Between the memory row and a logic row 613 is an I/O-to-logic and I/O-to-memory interface 622. This interface includes memory-to-logic and logic-to-memory interleaving vertical (IV) connections 626. Also between the memory row and the logic row is an IO band. The IO band includes the pads, input circuit, output circuitry, or IO circuitry, or combinations of these, similar to the input-output driver blocks 230 of Figure

2. In the present invention, the programmable logic integrated circuit is packaged using flip chip technology.

In typical packaging technology, pads are provided at the edges of the integrated circuit in what is sometimes referred to as a pad ring. Using typical packaging technology, the memory row and logic rows would be formed within the pad ring. In contrast, flip chip packaging technology provides for bump pads at any position within the integrated circuit. Therefore, using flip chip technology, an IO band with bump pads may be placed between the memory row and logic row as shown in Figure 6, instead of at the edge of the integrated circuit as shown in Figure 2. Note that the bump pads themselves are not necessarily physically formed within the IO band. The IO band includes the support circuitry, such as IO buffers and circuitry, that will be connected to the bump pads. The bump pads are formed using a top layer metal and may be placed over the logic or memory row or other circuitry, without interfering with the those circuits. By using flip chip technology along with the programmable logic architecture of the present invention, the programmable logic integrated circuit will have improved performance, which will be discussed further below.

Figure 7 shows an overview diagram of a programmable logic integrated circuit with memory rows. There is a memory row 703 (e.g., EAB or ESB row) at a top row or edge of the integrated circuit and a memory row 706 at a bottom row or edge of the integrated circuit. In the top memory row there is a phase-locked loop (PLL), the EABs or ESBs, and control logic and pins. An address register 710 for the memory is adjacent the top memory row. In alternative embodiments, the address register may also be formed or placed in other locations of the integrated circuit, such as between the top memory row and adjacent IO band, below the memory row. A data register 715 is organized on one side of the integrated circuit.

In this specific implementation, there are twelve logic rows (or rows of LABs) arranged in four sections 722 of three rows each. Between each section of logic rows is a regular IO band 726. The regular IO band includes input-output circuit to programmably route and connect signals to the logic blocks of the logic rows. Signals from outside the integrated circuit are input via bump pads and the IO band. Between the memory row and a logic row section is a low voltage differential signal (LVDS) input IO band and PLL. This is the IO band used for IO-to-logic and IO-to-memory described in Figure 6. This IO band programmable routes and connects signals to the memory and logic blocks. This IO band also includes LVDS input buffers for high-speed input. There are no IO bands between the outside edges of the memory row and edges of the integrated circuit.

By using flip chip technology and arranging the ESBs in rows at the edges or peripheries of the integrated circuit, this generally increases the performance on the programmable logic integrated circuit for a number of reasons. In an implementation where the memory blocks are in the middle of the LAB array (e.g., column), the programmable interconnect is lengthened by the width or the length of the EAB. This can be a considerable distance since a typical EAB can be the wide as five LABs or more. By arranging the ESBs at the edge of the integrated circuit and placing the IO band adjacent the inside edge of the row of ESBs, the programmable interconnect is shortened since it does not need to run across the width or length of an ESB. The inside edge of the row of ESBs is adjacent the logic row or the edge of the row of ESBs nearest to the core of the integrated circuit. The programmable interconnect typically needs to extend to the IO band circuitry. However, in the present invention, because the IO band is inside the memory row, and not between the memory row and edge of the integrated circuit, the programmable interconnect only needs to extend to the inside edges of the memory rows. This shortens the vertical programmable interconnect of the integrated circuit, thus improving performance.

To illustrate this point, line 740 is the representative length of a programmable interconnect line that extends the full length of the integrated circuit, including the length of the memory blocks (to the outside edges of the ESBs). However, as represented by line 748, when the programmable interconnect only extends to the inside edge of the memory blocks, the length of the programmable interconnect is shorter. In particular, the vertical programmable interconnect lines extend from an inside edge of the top row of memory blocks to an inside edge of the bottom row of memory blocks. This reduces capacitance and loading, which improves the programmable logic performance.

ESBs are typically wider than a single LAB. In a specific implementation, an ESB is five LABs wide. Therefore, by arranging the ESBs into rows, the horizontal programmable interconnect lines will be shorter because the integrated circuit is less wide. Although the integrated circuit is taller, lengthened in the vertical direction, the vertical programmable interconnect lines are not lengthened greatly because the lines do not run across or through the ESB. The vertical programmable interconnect lines stop at the edge of the ESB. Since the vertical programmable interconnects do not need to extend through the ESBs, space is saved in the ESB row and may be used to provide for more internal ESB signal and other routing. This increases the overall functionality and performance of the ESB.

Furthermore, when ESBs and LABs are mixed in a single row, the programmable interconnect resources are typically tuned for the LABs or certain compromises are made so that the programmable routing resources are not necessarily optimum for memory blocks. When LABs and ESBs share programmable routing resources, sacrifices are made to allow both to operate well. By organizing the ESBs together in a row, however, the programmable routing resources for the ESB can be better optimized since the ESB row does not have LABs. In fact, the programmable routing resources for the ESB row can be optimized especially for ESBs. Also, the programmable interconnect resources for the logic rows can be optimized especially for LABs and LEs, and no compromises need to be made for ESB memory blocks. The IO band includes I/O pads, input pads, input circuitry, input/output circuitry. By having the IO bands embedded within the logic array blocks, this reduces the distances needed to route signals from external to the integrated circuit. This further enhances the performance of the integrated circuit.

The composition of the programmable interconnect resources for an ESB row will generally be different compared to a logic or LAB row. The ESB row will have more numbers of conductors in its programmable interconnect resources because more wires are needed to support memory operations, such as interconnections for data and addresses. Also, since ESBs are five LABs wide, some of the shorter programmable interconnections are not needed. So, generally, an ESB row will have more programmable interconnect resources that are longer when compared to a logic row.

Although this discussion is primarily focusing on ESBs organized in rows at the peripheries of the integrated circuit, the invention would also analogously be applicable to form ESBs organized in columns at the peripheries of the integrated circuit. Since flip chip technology is used, the IO bands will be formed within the inside edges of the columns of ESBs and not between the ESBs and at the integrated circuit edge. The horizontal programmable interconnect lines will be shorter since they only need to extend to the inside edges of the ESB columns.

In specific implementations, there will be six ESBs or ten ESBs per row. One EAB or ESB block will be 2K or 4K. An EAB or ESB block may also be larger than 4K. The width and depth size of the memory can be programmably adjustable. For example, a 4K EAB block can be programmably arranged as 4096x1, 2048x2, 1024x4, 512x8, and so forth. Multiple EABs can be combined and stitched together to form larger memories.

The implementation in Figure 7 has 12 rows with 40 LABs and 2 EAB rows. There are 4800 LEs and 48K bits total of memory. In other implementations of a programmable logic integrated circuit, the number of logic rows and LABs per row will vary. For example, one implementation has 6 rows of 40 LABs each and 2 EAB rows. Each logic section has three rows of LABs. There are 2400 LEs and 48K bits total of memory. Another implementation has 9 rows of 40 LABs each and 2 EAB row. Each logic section has three rows of LABs. There are 3600 LEs and 48K bits of memory total. Another implementation has 9 rows of 80 LABs each and 2 EAB rows. There are 7200 LEs and 112K bits of memory total. Another implementation has 14 rows of 80 LABs each and 2 EAB rows. There are 11,200 LEs and 112K bits of memory total. Another implementation has 18 rows of 80 LABs each and 2 EAB rows. There are 14400 LEs and 112K bits of memory total.

Figure 8 shows a more detailed diagram of the routing within a 3-by-2 portion of a LAB array. Each LAB includes ten LEs 805 and a driver block 809. In other embodiments, the LABs may include any number of LEs, such as eight, sixteen, twenty, and twenty-four. The LEs include the configurable logic circuitry configurable by the user to implement logical functions, both combinatorial and registered. The driver block includes driver, multiplexer, and other circuitry to drive out of the LAB to the programmable routing resources for that row.

The horizontal programmable interconnect 815 includes GH, dH, HNFL, and FGH lines. These for programmable interconnect of various lengths and special purposed. For example, there may be half-length line or “fast” GH lines (i.e., FGH). The vertical programmable interconnect 825 includes HV, V, FGV, and FHV lines. Each LE of a LAB can directly programmably connect to driver block 809 of the LAB itself and to the driver block of the adjacent LAB in the same row (to the right of the LAB) through a direct connection 828.

Another feature of the invention is that LEs also directly programmably connect (through direct connections 830 and 835) to the driver blocks or programmable resources in the adjacent row above and adjacent row below. Connections 835 and 830 are referred to as interleaving vertical or IV lines. The ESB row also has IV lines, similarly as shown for LABs. IV lines are used for LAB-to-LAB, ESB-to-LAB, and

LAB-to-ESB communication between different rows of LABs. IV lines provide direct connections from a particular row (LAB or ESB) to the routing resources of an adjacent row (LAB or ESB), without using the programmable interconnect resources, 825. Each IV line 830 and 835 includes a number of conductors. For example, in one implementation, there are five IV conductors between LABs or ESBs. These connections between the LAB and the adjacent driver blocks, in an adjacent row and column, are direct connections; the horizontal or vertical programmable interconnects, 815 and 825, are not used in the connection path. The IV connections allow one row, logic or memory, to connect to another row's (logic or memory) horizontal programmable interconnect resources 815. Moreover, the IV lines improve performance. With IV lines, any logic or memory row can reach another row's programmable routing resources as fast as it can reach its own row's programmable routing resources.

Moreover, in a programmable logic integrated circuit where a row has LABs mixed with ESBs, the ESBs can reach the programmable resources of other LABs in the same row directly. In the architecture of the present invention, ESBs are grouped into one row and IV lines allow jumps to other rows. Therefore, with IV lines, the connections between ESBs to a LAB row's programmable routing resources will not be slower than the architecture with LABs and ESBs. Also, LEs in the top or bottom logic row or the present architecture can reach more numbers of ESBs with relative fast performance than in the mixed LAB and ESB row architecture, which generally has fewer ESBs per row.

For the row of LABs adjacent the EAB (or ESB) or memory row, the IV lines from the LABs can be used to programmably connect to the ESB, and IV lines can be used to programmably connect from the ESB to the LABs in the adjacent row. This allows a direct connections between the LABs and ESBs in the row above (or below) without using the horizontal and vertical programmable interconnect. These relatively short interconnections (IV lines) are generally higher performance because there is less capacitance and loading. The horizontal and vertical programmable interconnect will be bypassed. Also, by using these direct connections, the longer vertical programmable interconnect resources may saved for other purposes.

An aspect of the invention is top and bottom EAB or ESB row placement. This will increase inter-row LE-to-LE speed. This will increase IO-to-LE speed (with flip chip PLD). This will increase the speed of some IO-to-memory connections, without penalty to IO-to-LE speed. By arranging the EABs or ESBs in rows, this will enhance memory performance, enhance logic performance, increase speed of IO interface, and provide a more efficient layout.

An aspect of the invention is to separate EABs (or ESBs) and logic by placing multiple EAB (or ESB) blocks in memory rows. In a specific implementation, any number of EABs (six, ten, or more) are grouped into a single row. Another aspect of the invention is to allow for interleaving vertical (IV) line connections between EABs (or ESBs) and LABs in adjacent rows. A further aspect is to place EAB rows on top and bottom of the core, and to terminate core vertical writing at memory-to-logic interface. This will permit faster interrow LE-to-LE speed, and free additional routing metal channels within the EAB (or ESB). A yet further aspect is that IO bands can be placed on memory-to-logic interface to allow some IOs to have fast IO-to-memory connections without a penalty to IO-to-logic speed.

A further aspect of the invention is the ability to fast stitch multiple EAB (or ESB) blocks together will increase p-term EAB performance. With rows of EABs, this will speed-up both regular and p-term modes of EAB operation. EAB-to-logic performance will not suffer due to availability of fast inter-row connections. An increased size of core of logic can feed the EAB fast. Increased layout efficient (area and time-wise) will result because LABs and EABs can be drawn independently. Inner-row LE-to-LE speed will increase. Provides square-root LE to memory bits dependence.

The Altera 20K product family provides tristatable GV lines so that multiple EABs in x1 mode can be stitched together to form deeper memories. For example, two 2K-bit EABs can be stitched together to form a 4Kx1 memory and four EABs can form an 8Kx1 memory. The Altera 20K tristatable GV pattern supports an address depth in the power of two efficiently. For example, the 20K400 product has twenty-six EABs per column, so it can support thirteen 4Kx1 memories or six 8Kx1 memories. However, it does not guarantee to support five 10Kx1 memories. A more populated GV pattern can support more deep-memory combinations but it requires more area. Furthermore, speed is affected because each EAB bit 0 output is required to drive more diffusion loading.

In the invention, EABs or ESBs are grouped together in rows such that tristatable GH lines are provided to support deep memories. Figures 9 to 12 show an ESB row with fifteen ESBs with can be stitched together with partially populated multiplexing. An specific optimized multiplexing structure is shown in Figures 9 to 12 such that it only requires seven tristatable GH or programmable interconnect lines (horizontal lines) with thirty-seven pass gates (circles) to support any possible deep memory combination. These pass gates or other programmable connection circuits are selectively programmed to implement the desired memory block stitching. As desired, the horizontal and vertical conductors will be connected together at the intersection "circles." In the figures, filled in or darkened circles indicate a connection for illustration purposes. The stitch pattern provides relatively high performance because only a maximum of three pass gates are need per ESB output (shown as vertical lines in the figures). This minimizes the loading at the ESB outputs.

For example, Figures 9 to 12 show the 15 ESBs stitched together to form a 30Kx1 memory (assuming 2K ESB sizes). Figure 10 shows the ESBs stitched together to form three 10Kx1 memories. Figure 11 shows a seven 4Kx1 memories. Figure 12 show two 14Kx1 memories. Figure 13 shows five 6Kx1 memories. In fact, all other possible memory combinations are supported as long as the total memory bits is less than 30K-bits (i.e., 2K x 15 blocks) and each memory depth is 2-multiple of 2K (e.g., 4K, 6K, 8K, 10K, and so forth). For example, 4K/4K/4K/4K/6K/8K, 6K/6K/8K/10K, or 10K/20K combination can easily be supported. Similar patterns can be created to support different number of ESBs per row with different ESB size. For example for an ESB size of 4K, the total sizes will be two times the numbers discussed above.

Partially populated multiplexing is used rather than fully populated multiplexing. Fully populated multiplexing refers to having a connection at every intersection to allow every possible stitching combination. However, this approach requires many more than 37 pass gates. For the implementation show, there would be 105 (7x15) pass gates. Partially populated multiplexing with the pattern shown in the figures provides for stitching of the ESB with the desired sizes, but at the same time, does not require as much integrated circuit area as fully populate multiplexing. Figures 9 to 12 include that all memory depth combinations are supported with a maximum of three output loading per ESB (in x1 mode).

Figures 9 to 12 show merely one example of a stitching pattern. There are many different combinations of stitching patterns that are equivalent to the one shown in Figures 9 to 12 that would allow the same resulting stitching of ESB blocks. Any one of these combinations may also be used.

The Altera 20K product family provides an embedded address decoder (cascade address decoder) in each ESB. Although each ESB only has 2K bits of memory with an 11-bit address, a total of sixteen address bits can be sent directly to each ESB where the upper five bits of the address decoded by a dedicated single-output decoder. As the result, no external decoding logic is required for up to 64K deep memories. However, extra die area is required for each ESB to include the embedded decoder, and more LAB lines and associated PIA (programmable interconnect resources) are needed for the five extra address lines per port.

In the invention, the EABs or ESBs are grouped into rows. Multiple EABs or ESBs can be accessed directly using EAB GH lines. Dedicated circuitry per EAB row can be added to support various peripheral EAB functions. For example, the cascade decoder in each 20K product EAB can be replaced by a single upper address decoder per EAB row. This is shown in Figure 14. This row-level decoder requires fewer transistors than the distributed version and does not need the extra LAB lines per EAB. This decoder circuitry can be used to form deep memories by using decoder circuitry. There may be one decoder for each ESB row enable this combining or stitching together of the memories. Since the decoder will be localized and grouped in the same row as the memory, this will ensure the decoding is fast.

The stitch decoder may also be implemented using programmable logic instead of a dedicated circuit. However, to minimize impact on programmable interconnect use, a dedicated decoder is preferred.

In Figure 14, there are 32 EABs or ESBs 1405. There are 11 address lines 1412 for the EABs, which are connected by a GHs. Each of the enable (En) inputs of the EABs is connected by a GH line to a decoder 1418 used to decode the 5-bit upper address. The decoder will be a 5-to-32-bit decoder. Consequently, based on the 5-bit upper address input, one of the EABs will be enabled. Then, the 11-bit address is used to select the particular bit of the selected EAB for outputting to the data bus.

Other peripheries for EAB can be supported in a similar fashion. For example, a dedicated counter circuitry with overflow/underflow detection can be provided to support deep FIFO/LIFO memories efficiently. This circuitry would be grouped with the EAB at the peripheries of the integrated circuit.

This description of the invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form described, and many modifications and variations are possible in light of the teaching above. The embodiments were chosen and described in order to best explain the principles of the invention and its practical applications. This description will enable others skilled in the art to best utilize and practice the invention in various embodiments and with various modifications as are suited to a particular use. The scope of the invention is defined by the following claims.

CLAIMS

1. A programmable logic integrated circuit comprising:

a top row of memory blocks formed along a top edge of the integrated circuit;

a bottom row of memory blocks formed along a bottom edge of the integrated circuit;

a plurality of logic blocks, arranged in rows and columns, programmably configurable to implement logic functions, formed between the top and bottom rows of memory blocks;

a plurality of horizontal programmable interconnect lines, programmably coupled to the logic blocks; and

a plurality of vertical programmable interconnect lines, programmably coupled to the logic blocks.

2. The programmable logic integrated circuit of claim 1 further comprising:

a plurality of IO bands comprising input and output circuitry, formed between the top and bottom rows of memory blocks.

3. The programmable logic integrated circuit of claim 1 wherein the vertical programmable interconnect lines extend from an inside edge of the top row of memory blocks to an inside edge of the bottom row of memory blocks.

4. The programmable logic integrated circuit of claim 1 wherein the top row of memory blocks comprises six blocks of memory which can be programmably coupled to form larger memory blocks.

5. The programmable logic integrated circuit of claim 1 wherein the logic blocks are grouped into sections of logic blocks and between each section is an IO band.

6. The programmable logic integrated circuit of

claim 5 wherein an IO band comprises IO pads and input circuitry.

7. The programmable logic integrated circuit of claim 5 wherein each section has three rows of logic blocks and at least forty columns of logic blocks.

8. The programmable logic integrated circuit of claim 1 further comprising:

a phase locked loop circuit formed along with the top row of memory blocks.

9. The programmable logic integrated circuit of claim 1 wherein a logic block comprises:

a plurality of logic elements and a driver block, wherein the plurality of logic elements are directly programmably coupled to a driver block of an adjacent logic array block is the same row, without using the horizontal or vertical programmable interconnect.

10. The programmable logic integrated circuit of claim 1 wherein a logic block comprises:

a plurality of logic elements and a driver block, wherein the plurality of logic elements are directly programmably coupled to a driver block in an adjacent logic array block in an above row, without using the horizontal or vertical programmable interconnect.

11. The programmable logic integrated circuit of claim 1 wherein a logic block comprises:

a plurality of logic elements and a driver block, wherein the plurality of logic elements are directly programmably coupled to a driver block in an adjacent logic array block in a row below, without using the horizontal or vertical programmable interconnect.

12. The programmable logic integrated circuit of claim 1 wherein each row of memory blocks further comprises a grid of stitching conductors to programmably stitch one memory block to other memory blocks in the row to form larger memory blocks.

13. The programmable logic integrated circuit of

claim 12 wherein the grid comprises fourteen first stitch conductors coupled to the stitch memory blocks and seven second stitch conductors, transverse to the first stitch conductors.

14. The programmable logic integrated circuit of claim 13 wherein pass gates to programmably couple first and second stitch conductors at intersections between a first first stitch conductor and first and third second stitch conductors,

a second first stitch conductor and first and third second stitch conductors,

a third first stitch conductor and first, second, and third second stitch conductors, and

a fourth first stitch conductor and first, second, and third second stitch conductors.

15. The programmable logic integrated circuit of claim 14 further comprising pass gates at intersections of

a fifth first stitch conductor and first and third second stitch conductors,

a sixth first stitch conductor and first and third second stitch conductors,

a seventh first stitch conductor and first, third, and fourth second stitch conductors, and

an eighth first stitch conductor and first and fourth second stitch conductors.

16. The programmable logic integrated circuit of claim 15 further comprising pass gates at intersections of

a ninth first stitch conductor and first, fourth, and fifth second stitch conductors,

a tenth first stitch conductor and first, fifth, and sixth second stitch conductors,

an eleventh first stitch conductor and first, sixth, and seventh second stitch conductors, and

a twelfth first stitch conductor and first; sixth, and seventh second stitch conductors.

17. The programmable logic integrated circuit of claim 16 further comprising pass gates at intersections of

a thirteenth first stitch conductor and first and seventh second stitch conductors,

a fourteenth first stitch conductor and first and seventh second stitch conductors, and

a fifteenth first stitch conductor and first and seventh second stitch conductors.

18. The programmable logic integrated circuit of claim 1 further comprising a circuit to stitch the EABs in a row comprising:

a decoder circuit coupled to enable inputs of each of the EABs.

19. The programmable logic integrated circuit of claim 18 wherein the decoder circuit takes a 5-bit upper address and selectively enables one of the EABs.

20. The programmable logic integrated circuit of claim 2 wherein no IO bands are formed between the top row of memory block and the top edge of the integrated circuit.

21. The programmable logic integrated circuit of claim 20 wherein no IO bands are formed between the bottom row of memory blocks and the bottom edge of the integrated circuit.

22. A programmable logic integrated circuit substantially as hereinbefore described with reference to Figures 1 to 14 of the accompanying drawings.

23. A programmable logic integrated circuit comprising:

a plurality of programmable interconnect lines arranged in rows and columns;

a plurality of logic array blocks between rows and columns of the programmable interconnect lines, wherein the logic array blocks are arranged in rows and columns and programmably coupled to the programmable interconnect lines, and the logic array blocks are programmably configurable to perform logical functions; and

a plurality of embedded array blocks between rows and columns of the programmable interconnect lines, wherein the embedded array blocks are arranged in a row and each embedded array block includes a RAM block.

24. The integrated circuit of claim 23 wherein the RAM block includes at least 2048 bits of memory.

25. The integrated circuit of claim 23 wherein the logic array blocks include logic elements.

26. The integrated circuit of claim 23 wherein two embedded array blocks of a row can be programmably combined to provide a larger memory than available with a single embedded array block.

27. The integrated circuit of claim 23 further comprising:

a decoder coupled to receive a first address and generating a plurality of output control lines, each coupled to an enable input one of the embedded array blocks.

28. The integrated circuit of claim 27 wherein the output control lines are programmable interconnect lines.

29. The integrated circuit of claim 27 wherein some of the programmable interconnect lines are coupled to provide a second address to address inputs of the embedded array blocks.

30. The integrated circuit of claim 27 wherein the first address comprises five bits.

31. The integrated circuit of claim 29 wherein the second address comprises eleven bits.

32. The integrated circuit of claim 23 wherein a row of embedded array blocks comprises thirty-two embedded array blocks.

33. The integrated circuit of claim 23 wherein a row of embedded array blocks is programmably configurable to be one 30Kx1 memory, two 14Kx1 memories, three 10Kx1 memories, five 6Kx1 memories, or seven 4Kx1 memories.

Application No:	GB 0016223.0	**Examiner:**	Mike Davis
Claims searched:	1-22	**Date of search:**	22 September 2000

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.R): G4H (HU)

Int Cl (Ed.7): H03K

Other:

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
A	WO 98/10517 A1 (XILINX) eg abstract and Fig.8B	-
A	WO 97/30517 A1 (ACTEL) eg abstract and Figs.1,2	-

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.