Lattice™ Semiconductor Corporation

Selecting the Right High-Density Device

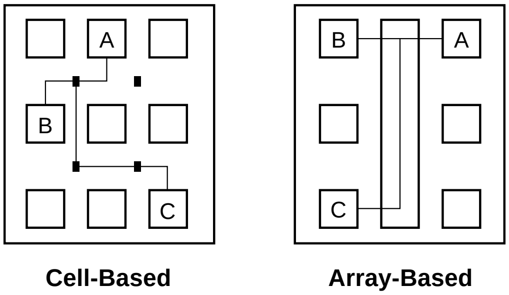
Introduction

Board designers today have several options for implementing designs in high-density programmable devices. Due to technology and design considerations, no single device provides the best solution for the challenges facing designers. To address this, design engineers often use multiple types of high-density devices on a single board. This application note will outline various applications issues and examine the appropriate high-density solutions. It will also examine from the perspective of the user, the impact of design implementation on the process of selecting a device.

High-density programmable devices can be broadly classified into two major types: Field-Programmable Gate Arrays (FPGA) and High-Density Programmable Logic Devices (HDPLD). FPGA devices are cell-based and usually have small grain-sized logic blocks with distributed interconnects across the device. High-Density Programmable Logic Devices are array-based and have large grained AND-OR array logic blocks with centralized interconnects (see Figure 1). Similarly, board designs can be broadly classified into two types: control intensive and data intensive. Control intensive designs usually contain such subfunctions as Cache control, DRAM control and DMA control and require limited data manipulation. Data-intensive designs require complex manipulation of data bits which are typically found in telecommunications applications. To select a high-density device, a designer must examine:

- Performance
- Utilization
- Ease-of-Use

Figure 1. Cell-Based and Array-Based Devices



Cell-Based **Array-Based**

Performance

When implementing a logic design into a high-density device, it is typically partitioned into multiple logic blocks or cells and then the various cells are connected together using interconnect resources. The performance of a design is determined by the combination of the cell speed and the interconnect speed.

Cell Speed

A logic function is divided into subfunctions which fit the basic building block of the high-density device. Often the number of inputs is the most important consideration. The subfunctions should require no more inputs than are available in the logic block of the device. Smaller logic blocks tend to be faster but they offer fewer inputs. Functional implementation often requires a number of logic blocks cascaded into multiple levels of delay to implement the logic. This slows down the functional speed of the logic dramatically. Control functions are typically input-intensive and will be faster in devices with building blocks that allow for a large number of inputs. Data functions require fewer inputs and may be faster in devices which have fewer inputs per logic block.

Cell-based devices are very granular and have small logic blocks. They have four to eight inputs per logic block with cell speeds that are about 75 percent of the larger array-based building blocks. While these devices can implement critical data functions at fast speeds, for most control functions they require two to three levels of cascading delays.

The array-based devices have larger building blocks with 16 to 48 inputs and delays that are approximately 25 percent slower than cell speeds. They can accommodate most control logic function requirements in terms of inputs to the logic block and implement them in one level of delay. However, an overly large input logic block is ineffective as it only adds to the logic block delay.

Another alternative is the ispLSI® and pLSI® devices which offer a large number of inputs (18-24) in every Generic Logic Block (GLB). These inputs are sufficient to accommodate the logic requirements of control functions from eight to 12 inputs with the fastest possible speed. They also accommodate data functions which require two to four inputs per output, while maintaining high speed.

Figure 2. ispLSI GLB

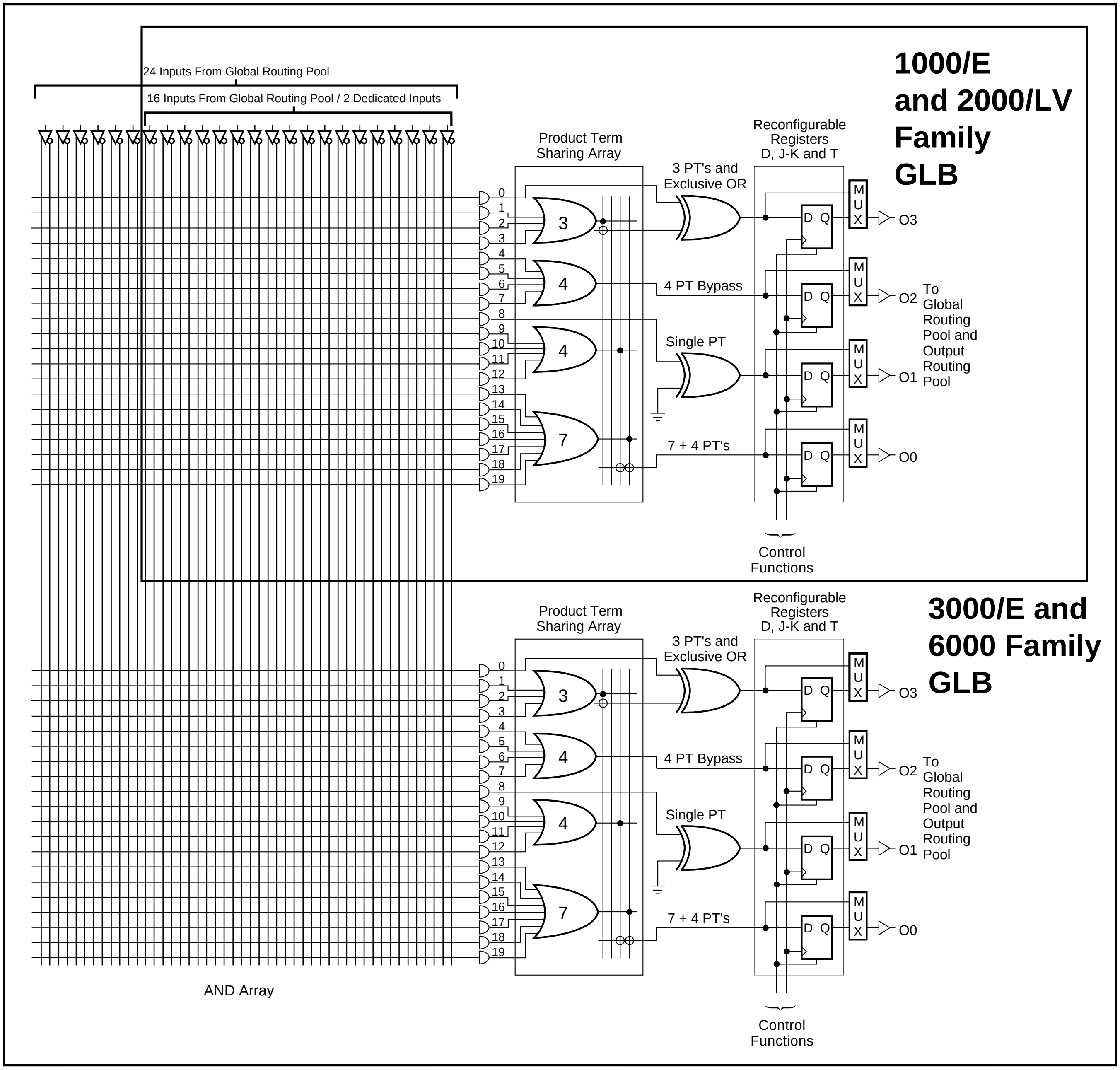
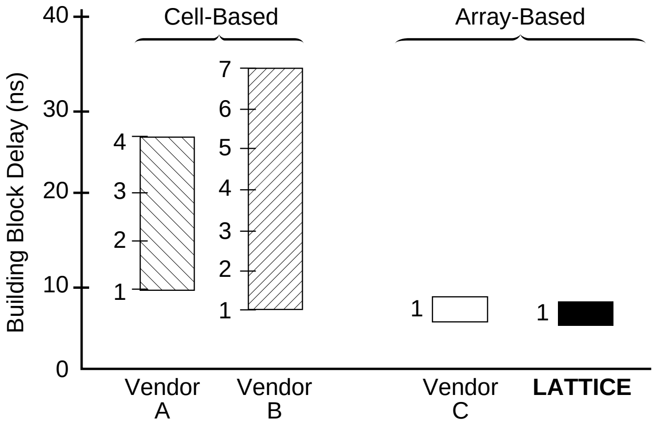

24 Inputs From Global Routing Pool
16 Inputs From Global Routing Pool / 2 Dedicated Inputs
1000/E and 2000/LV Family GLB
Product Term Sharing Array
Reconfigurable Registers D, J-K and T
3 PT's and Exclusive OR
4 PT Bypass
Single PT
7 + 4 PT's
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
3
4
4
7
D Q
MUX
O3
O2
O1
O0
To Global Routing Pool and Output Routing Pool
Control Functions
3000/E and 6000 Family GLB
AND Array
0845a

Figure 3 illustrates the multiple logic block delays required to perform common logic functions in some of the popular high-density devices available today. Due to the limited number of inputs available in cell-based devices, the number of logic blocks cascaded to perform a function can be as high as four to seven. The array-based devices (Vendor C) require only one level of delay. ispLSI and pLSI devices require only one logic block for most functions. The logic block delay is small and is comparable to most cell-based devices.

Figure 3. Building Block Performance



Source: Lattice Semiconductor Applications, Vendor Literature

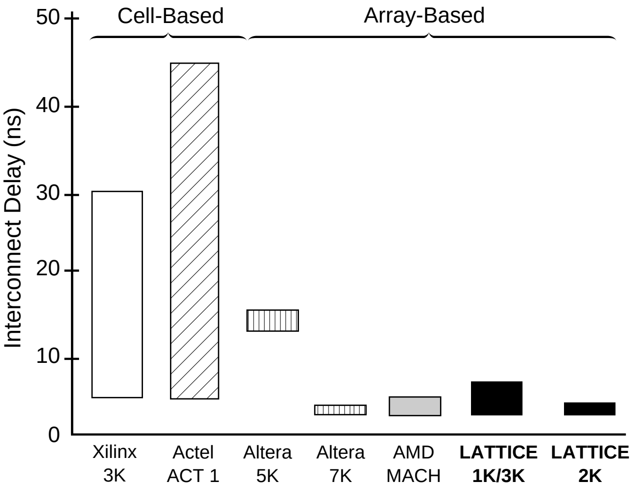
Interconnect Speed

Interconnect speed is another important consideration not only for connecting subfunction logic blocks, but also for connecting signals from one logic function to another. Interconnects affect final system performance as much as logic blocks do.

Cell-based devices offer distributed interconnects with variable length lines spanning the length and width of the device and interconnecting various logic blocks with finite delay interconnect points. Often, signals have to traverse multiple line segments and interconnect points for a connection. In general, closely located logic blocks have a shorter delay since signals travel through fewer lines and interconnect points. The opposite is true for logic blocks located further apart. There is a large variation in the interconnect delays based on the placement of the related logic blocks (see Figure 4). To improve system performance for control-oriented functions in a cell-based device, a large number of signals and related logic blocks need to be placed in close proximity. Frequently this is physically impossible and/or requires many placement iterations. For designs such as state machines and counters, the final performance is often determined by the worst case signal speed. In such cases, cell-based devices with distributed interconnects offer slower interconnect performance and consequently slower overall system performance. Data functions require fewer delays and can be implemented relatively faster. Placement of related data bits close to each other facilitates fast performance for data-oriented functions. This is sometimes difficult to achieve with a large number of data bits.

Array-based devices with centralized interconnects offer uniform interconnect delays. The ispLSI and pLSI devices offer uniform interconnect delays with significantly faster interconnect performance than existing array-based devices, and consistently provide best case cell-based device delays as illustrated in Figure 4.

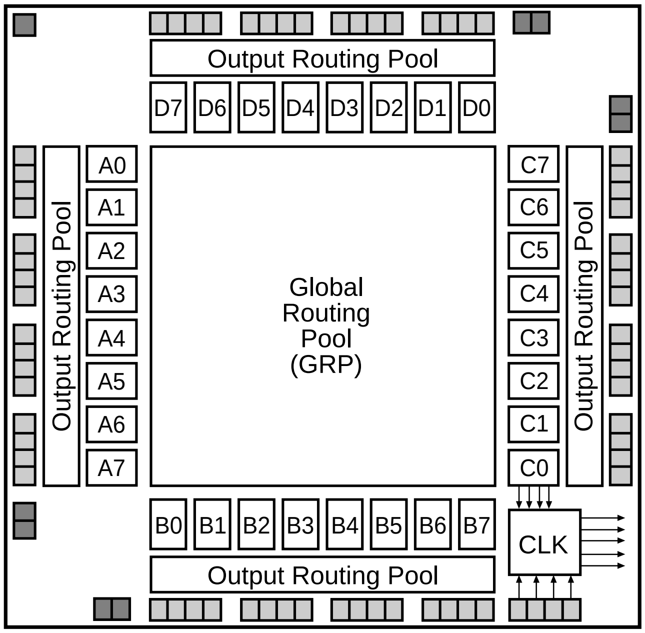
Figure 4. Interconnect Performance



Source: Lattice Semiconductor Applications, Vendor Literature

State-of-the-art Automatic Place and Route (APR) software has not yet reached the point where interconnect performance can be considered optimal. In general, cell-based devices either require a long APR time, typically a number of hours or days, in order to reach near-optimal interconnect speeds. Shorter route times result in reasonable performance. Uniform delays in array-based devices eliminate the need for intelligently placing related logic blocks closer, thereby reducing APR time to a few minutes. ispLSI and pLSI devices go a step further and offer faster interconnect delays using the proprietary centralized Global Routing Pool (GRP) (see Figure 5) which retains fast APR times. This is especially good for data-intensive designs where all data bits perform equally.

Figure 5. ispLSI 1032 Block Diagram



Utilization

When mapping a design, utilization is defined by how much of a device is used. In general, granular architectures are more effective in offering higher utilization for data-intensive designs than large logic block architectures which are better for control-intensive designs.

Array-based architectures typically require sophisticated synthesis algorithms to compress logic from multiple stages into a large single stage block to increase utilization. While array-based devices implement control-intensive designs more effectively, cell-based architectures with their smaller logic blocks have less need to compress logic into one stage of the design. Data-intensive designs which typically require a large number of registers are more effectively implemented in a cell-based architecture which has higher register-to-logic ratios.

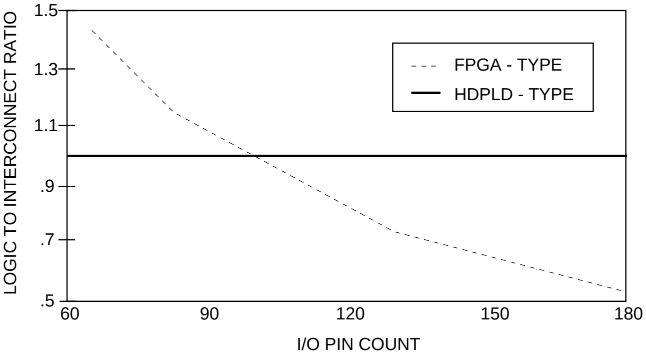
The ispLSI and pLSI devices are neither as granular as some cell-based architectures, nor as large as some array-based architectures, since they offer a grain size of four outputs per logic block (GLB). However, they offer effective utilization since designs combine data and control functions.

ispLSI and pLSI devices offer a Product Term Sharing Array (PTSA) within each GLB. The PTSA optionally shares GLB product terms between the four GLB outputs thereby enhancing logic block utilization.

As devices are scaled to higher densities, interconnect resources should increase at the same pace as logic resources. This ensures that all available logic is fully utilized in the device. The distributed nature of cell-based interconnects does not lend itself well to this scaling. Figure 6 shows the logic-to-interconnect ratio for one family of cell-based high-density devices. At higher densities, this means a lower utilization of the device since the logic cannot be mapped as easily as at the lower end of the spectrum. Array-based devices scale the interconnect resources at the same level as logic resources, but offer better utilization at the higher-end range of devices.

The ispLSI and pLSI Global Routing Pool (GRP) provides all signals globally to all device GLBs. The GRP size is scaled to provide full 1:1 logic-to-interconnect ratio, ensuring all device logic is fully utilized, regardless of the device size.

Figure 6. Logic to Interconnect Ratio



Ease-of-Use

Most designers use high-density devices as a means for logic implementation and concentrate on the main functionality of the board (e.g., the microprocessor or the graphics section). Their time is spent on the overall functionality of the board and not the basic logic. Ease-of-use and quick design turnaround times are critical to any digital designer. Ease-of-use is determined by a number of factors. Some critical factors directly related to the choice of device architecture are:

- Predictability of Performance
- Design Rework
- Design Entry
- Turnaround Time

Predictability of Performance

The performance of the design is determined by the system considerations and is usually driven by the processor requirements or other considerations like graphics and screen resolution. High-density devices frequently do not determine the final system speed. Designers need to know in advance the final performance of the logic implemented in the high-density device to determine the feasibility of the part selected. A designer also needs to know the speed grade required in advance, in order to estimate the cost of the design.

For cell-based devices, the number of delay levels necessary to implement the design function is not typically known. Modifications to the design often may cause a change in the number of delay levels. Similarly, it is difficult to predict how the software will place the logic blocks, as explained earlier. Even if only one out of ten critical signals is slow, it will slow down the device system speed. Array-based devices, ispLSI and pLSI devices have predictable levels of logic and interconnect delays which allow the designer to estimate speed in advance and maintain fast speeds.

Design Rework

Very few designs work the first time logic is entered into a device. Most designs not only require logic addition or subtraction, but also pinout changes and rework. This rework is often due to logic debugging or changes in the specification of the final product. Many digital designers prefer an incremental design approach where small portions of designs are implemented at a time and debugged before new portions are added.

For cell-based devices, every logic change requires a new set of logic mappings into the device cells and a new set of interconnect mapping into device interconnect lines. This leads to significant changes in the performance of the device and to undesired pinout change.

Array-based devices typically do not have any adverse performance changes due to logic changes. However, pinout changes frequently occur.

ispLSI and pLSI devices were developed to allow users to make logic changes without performance impact and to freeze pinouts when incremental design changes are done. ispLSI and pLSI devices offer an Output Routing Pool (ORP), allowing GLB outputs to be routed to many different I/O pins. Also, the ispLSI and pLSI GRP allows I/O pin inputs to be available to all GLBs. These two features combined offer the flexibility necessary to maintain pinouts in subsequent design iterations while maintaining consistent performance.

Design Entry

There are two categories of digital designers using high-density devices. The first is the designer who is a PLD user, such as GAL® devices, and is familiar with Boolean, state machine or HDL design entry syntax. For these designers, array-based devices offer direct mapping correlation from the entry syntax to design implementation, which is helpful in control-intensive designs. This makes such factors as logic implementation, speed of functions and race conditions predictable to the designer and simplifies the design task. The other category is the gate array designer who migrates to programmable gate array devices. For these designers, cell-based devices offer a closer correlation between schematic entry to design implementation. These designers implement data-intensive designs effectively, since they have a number of TTL type data function macros available to them. With synthesis techniques, however, schematic entry is also offered for array-based devices. Familiar design entry methodology speeds design entry time and simplifies the design process.

ispLSI and pLSI devices offer direct correlation with Boolean/HDL/state machine entry syntax. Extensive synthesis techniques are also used in the ispLSI and pLSI Development System software to offer easy schematic capture along with a large library of TTL-type macros.

Turnaround Time

Once a design is entered, the next critical step is design compilation and programming of the part. For cell-based devices with smaller logic blocks, the distributed nature of interconnects complicates matters. The Place and Route algorithm needs to satisfy multiple and often conflicting requirements for:

- Placing related logic closer for faster speed.
- Moving logic to satisfy critical timing requirements.
- Moving logic due to the lack of interconnect resources.
- Repartitioning logic to satisfy the above three conditions.

These basic requirements are interrelated and complex, making the compilation process very time consuming. Typical cell-based devices require two to eight hours for compilation in order to achieve reasonable system performance objectives.

Array-based devices with global connectivity and uniform interconnect delays eliminate the need to closely place related logic. The ispLSI and pLSI devices with centralized interconnect offer compilation times of minutes versus hours for cell-based devices, thereby improving designer productivity. This combined with the ispLSI version of the family which allows on-board reprogramming of multiple devices simultaneously, offers a whole new dimension for logic design.

Conclusion

A digital designer has multiple choices available for high-density designs. The current solutions broadly categorized as cell-based and array-based devices offer alternative advantages and disadvantages. The best solution depends on the type of logic functions being implemented, the performance required and other design-specific trade-offs. The ispLSI and pLSI devices offer the advantages of both cell-based and array-based devices.

High Performance	ispLSI and pLSI devices are designed to be extremely fast for both control and data-intensive functions and are excellent for functions requiring more than eight inputs per logic block.
Predictable Delays	The centralized GRP structure combined with wide input GLBs offers uniform delays which allow the designer to determine system speed in advance and maintain constant speeds in subsequent design iterations.
High Utilization	The ability to scale interconnect resources at the same level as logic resources combined with built-in flexibility of the GRP and ORP assure high device utilization. And the PTSA adds another level of flexibility for increasing logic block utilization.
Ease-of-Use	Predictable performance, quick design entry and rework time provide fast design turnaround. This simplifies the design process and shortens time-to-market.

The combination of high-performance, predictable delays, high utilization and ease-of-use not only offers a superior solution for design requirements, it is delivered in E^2CMOS® technology with reprogrammability and 100% testability, resulting in unparalleled device quality.

Lattice Semiconductor Corporation (LSC) products are made under one or more of the following U.S. and international patents: 4,761,768 US, 4,766,569 US, 4,833,646 US, 4,852,044 US, 4,855,954 US, 4,879,688 US, 4,887,239 US, 4,896,296 US, 5,130,574 US, 5,138,198 US, 5,162,679 US, 5,191,243 US, 5,204,556 US, 5,231,315 US, 5,231,316 US, 5,237,218 US, 5,245,226 US, 5,251,169 US, 5,272,666 US, 5,281,906 US, 5,295,095 US, 5,329,179 US, 5,331,590 US, 5,336,951 US, 5,353,246 US, 5,357,156 US, 5,359,573 US, 5,394,033 US, 5,394,037 US, 5,404,055 US, 5,418,390 US, 5,493,205 US, 0194091 EP, 0196771B1 EP, 0267271 EP, 0196771 UK, 0194091 GB, 0196771 WG, P3686070.0-08 WG. LSC does not represent that products described herein are free from patent infringement or from any third-party right.

LATTICE SEMICONDUCTOR CORPORATION
5555 Northeast Moore Court
Hillsboro, Oregon 97124 U.S.A.
Tel.: (503) 681-0118
FAX: (503) 681-3037
http://www.latticesemi.com

November 1996