

Using Reconfigurable Processing to Take Advantage of 0.13 μ m CMOS Technology

Introduction

Since the invention of the Integrated Circuit in 1958, the industry has continually advanced the state of semiconductor process technology. Today tens of millions of transistors can be placed on a single silicon chip less than one-half inch on a side. The ever-increasing transistor density was predicted by Gordon Moore in 1965 in what is, arguably, the most widely cited article in the history of the industry. Since 1965, and in accordance with Moore's prediction, every 18 months has seen a doubling of transistor density resulting in the faster, smaller and less expensive electronics products we all use everyday.

Less well publicized are the design methodologies that have been created to enable the widespread adoption of the higher density semiconductor technology. These advances have been made on a much less frequent basis but have been equally important in the advance of the industry. As transistor dimensions have been reduced, to 130 nanometers and smaller, so have design complexities, leading to design costs that have become unaffordable for all but the highest volume applications. It is our view that the industry is approaching a significant barrier to innovation caused by increasing design and fabrication costs. This barrier may prevent many smaller companies from bringing exciting silicon applications to market.

Advanced technologies such as 130nm (also referred to as .13 micron) and smaller hold significant value in performance and density if we can develop the methodology to unlock it and use it to full advantage. Here, we will look at the advantages and disadvantages of current design methods (Full Custom, Logic Synthesis, ASIC, and FPGA) and also explain why we believe an innovative new technology called Reconfigurable Computing will provide an

answer to the limitations of the current methodologies.

Custom Design

Maximum performance and minimum cost can be extracted from any process technology by the use of Full Custom design practices. In its purest form, this methodology requires complete control of the design at every level of the process. This includes, design of the individual transistor structures based on their function, control of the gate design, all custom circuit design and full control of the physical placement of structures from the lowest level to the full chip floor planning. Associated with virtually every custom design is the design of an integrated, robust clock routing structure, one of the most challenging aspects of the Full Custom approach.

While Full Custom design has been a viable methodology for many products, each process generation has increased the complexity of this approach and hence increased the design time and cost. The situation today is that Full Custom is applicable today to only the highest volume products. In fact, it could be argued that only memories and microprocessors have the volume profiles that justify the significant time and expense of this approach.

Other segments of the industry still use Full Custom but in limited areas of a chip design, usually only in those areas of the chip that require ultimate performance.

Logic Synthesis

Logic Synthesis is a design technique that allows designers to specify system architectures and functionality in a high level language (typically VHDL or Verilog) and leave the physical design of the gates and transistors to a series of software design tools that 'synthesize'

the abstract description to a physical implementation. Although the technique sounds promising, in practice the designs produced, using existing design tools, are relatively slow and large compared to the product results of an experienced Full Custom or ASIC design team.

ASIC Design

ASIC design methodology has provided a robust solution to low, moderate, and high volume as well as high performance design problems since the early 1980’s. This methodology has certainly been the primary design process behind much of the growth of the industry over the last twenty years, both for proprietary designs and standard product designs.

The basic premise of the ASIC process is that designs can be produced faster by using a library of pre-defined elements that do not need to be redesigned for every application. The pre-defined elements can be simple gate structures or collections of gates configured to perform some particular function. While the ASIC design process is certainly more cost effective than Full Custom it does not address the back end issues of place-and-route and timing closure. These design functions are becoming an increasing costly part of today’s complex designs.

With each technology generation the increasing non-recurring (NRE) costs have pushed the volume requirements for a profitable Return On Investment (ROI) to levels that exceed the available market size of many applications. In 130 nm CMOS, much has been written about the costs associated with bringing a design to full production status. Present estimates are \$5 - \$10M to develop a new ASIC (EE Times 10/25/02). With mask costs, tool costs, large talented design teams capable of handling the complex design issues, and IP purchases, many projects actually exceed these estimates (see Figure 1).

For example, with NRE costs of \$50,000 (1985) the NRE added only \$10 (20%) to a \$50 device at 5000 units. If the NRE costs are now \$5,000,000 (Masks and tools for 130 nm) 500,000 units are required to amortize the NRE down to \$10. This changes the viability of ASIC solutions for all but the highest volume and/or highest performance designs.

The heavy upfront costs are contributing to ASIC design starts being down 12% in 2002 following a 36% drop in 2001 (EE Times 10/25/02). Even when ASICs are desirable for performance, functionality, or recurring cost reasons, it is difficult to overcome the critical development time and NRE costs of product design using this methodology.

Figure 1: Estimated Costs for a Typical ASIC Design (EE Times estimates)

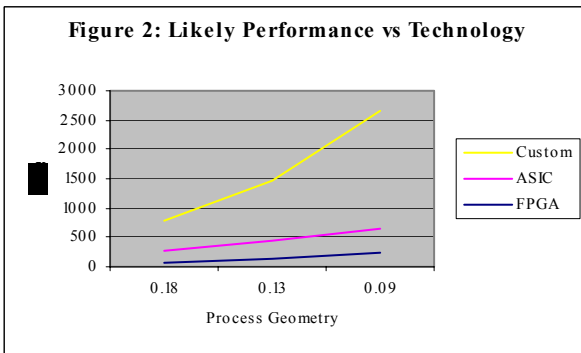
Item	Unit	Per Unit	Units	Cost
Designers	Man Year	\$150,000	35	\$5,250,000
Design Tools/Platforms	Man Years	\$100,000	35	\$3,500,000
.13u Masks	1 Set	\$750,000	1.5	\$1,125,000
Engineering Wafers	Wafer	\$15,000	12	\$180,000
IP Costs	1 database	\$50,000	3	\$150,000
Estimated Costs				\$10,205,000

The two most critical factors that determine the economic success of a product are Time-to-Market and Time-in-Market. Time-to-Market means how long it takes from the time a company from recognizes a market opportunity until they have a product to take advantage of it. Time-in-Market is how long a company can keep a solution in the market, e.g. how long before a product is obsolete. Time-to-Market determines market share, volume, revenue, and profit of a product. ASICs presently have long lead-time developments (design, layout, fabrication, test, debug, re-spin, test) due to the complexity of modern processes. This lead-time directly limits a company’s ability to get to market in a timely manner. Time-in-Market determines the long-term value of a system solution. The fixed gate design inherently limits the flexibility of an ASIC design in its adaptability to changing standards. This

inflexibility limits the Time-in-Market for many ASIC designs.

FPGA Design and Use

It is easy to see the attraction that FPGAs have in trying to solve the Time-to-Market and NRE cost issues associated with ASIC designs. FPGAs eliminate much of the time and cost of the ASIC design cycle but have performance limitations and recurring costs that make them increasingly non-competitive compared to ASICs as volumes increase. The large (30x) area overhead associated with the Look Up Table (LUT) architecture of FPGAs limits their ability to serve larger volume requirements. FPGAs are also limited by the nature of their interconnect and routing structure in the performance they can achieve. The result is a situation like that in Figure 2, where there is abundant performance available in a technology that is becoming harder and harder to use to economic advantage for low to medium volume applications.



With FPGA performance affecting the design, designers are forced to move to parallel implementations of algorithms, or divide the problem up over multiple chips. Also, as designers attempt to utilize as many of the available gates as possible, the placement and routing challenges add to the design time and decrease performance further. This forces prolonged design cycles and increases complexity. At some level of problem division, the control and communication overhead

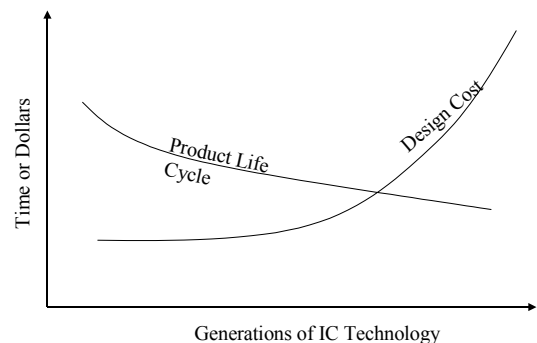
between multiple chips becomes so great that the problem can become intractable.

The greatest problem with FPGAs is recurring cost. If you are building 20,000 parts at \$500 each, the ASIC NRE starts to look pretty attractive. While most companies plan cost reduction programs, they seldom happen. The design team is off to the next generation design in order to hit market windows with expanded capabilities to match competition and increase market presence. Even relatively minor modifications to a complex design cost nearly as much time and money as the original development. As a result, the first implementation needs to be one that has the right economic return for the program.

Total Cost of Ownership

Another disturbing trend has been that as design costs have increased, the life cycles of most products have decreased. The dramatic cost increases presented over the last few years are displayed qualitatively with product life in Figure 3. The increasing disparity means that IC suppliers and their customers must increasingly consider total cost of ownership in product decisions at 130 nm and smaller geometries.

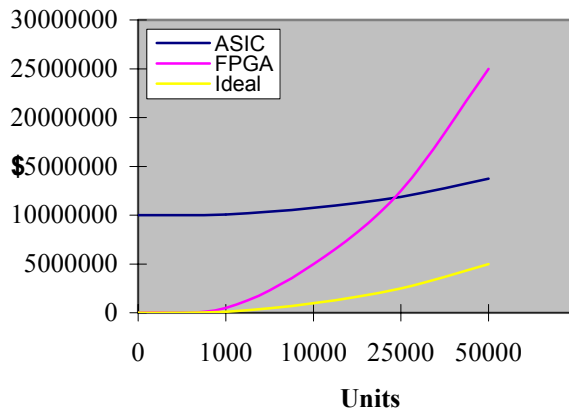
Figure 3: Product Life Plotted with Design Costs



Someone, supplier or customer, must bear the cost of development. As life cycles get shorter

the total lifetime profit from a product is reduced unless the supplier raises prices to a level unsustainable in the market. At some point the upfront investment required becomes unjustifiable. The ideal solution would display an economic profile that lowers the recurring cost curve, reduces the upfront investment cost and extends the product lifetime. If this can be accomplished, the lifetime profit profile becomes much more attractive for both supplier and customer and will bring the substantial benefits of 130 nm (and smaller) technology to a wider range of applications. This ideal situation is shown in Figure 4 which contrasts the cost profiles for ASICs, FPGAs and MathStar's Silicon Object technology.

Figure 4: Total Cost of Solutions



These curves look only at the cost factor, not the performance factors that can also be significant. Whereas the ASIC curve shows a favorable recurring cost profile, it starts at too high a number because of the inherently large upfront costs. In contrast, the FPGA curve begins at an acceptable point, but shows too steep a recurring cost slope. The ideal situation, one we believe will be achieved through MathStar's approach, would start low and stay well below the lifetime cost curves of either of the competing technologies.

The MathStar Approach

There are four primary drivers to the MathStar approach.

- 1) Provide a programmable platform chip that is suitable for a wide variety of applications, thus amortizing the upfront development costs over a larger volume.
- 2) Change the abstraction level for system and application designers from the gate level to the functional block level.
- 3) Provide a heterogeneous collection of programmable hardware elements tightly interconnected by a high-performance, flexible communication structure.
- 4) Use Full Custom design techniques to increase performance to the Gigahertz range and reduce costs to be competitive with ASICs.

The motivation behind each of these principles is discussed below:

Providing a Programmable Platform Chip

The advantages of producing a programmable platform that can be used for many different applications are obvious. Indeed, this philosophy has been the cornerstone of the FPGA market and the primary reason behind the growth of the FPGA suppliers.

Changing the Abstraction Level

The languages system designers use to express their ideas are not the language of semiconductor gate structures. Their more natural languages are Block Diagrams, Data Flow Graphs and Mathematics. Hence, there exists a 'semantic gap' between the language of the system designers and the chip designers. Between the system design tools and silicon gates there is a level of abstraction that can provide robust functionality with reasonable costs, rapid time to market, and excellent

performance. MathStar has followed the example of the software industry in trying to cope with the semantic gap issue.

Several years ago, the software industry created object oriented programming. Object oriented programming allowed for the development of very tightly coded modules that could be used to create programs. These code pieces (objects) were optimized once and re-used many times by programmers. Software programs then had the advantage of high performance (dense and fast) code that changed the efficiency of programmers without giving up much in overall code efficiency. Another cornerstone of the Object Oriented Programming methodology was the use of fully defined, robust interfaces to each of the objects.

Much the same way, MathStar is changing the interface for chip designers to be one of “Silicon Objects”, rather than Logic gates. This medium-grained level of abstraction moves from millions of gates to hundreds of tightly connected elements. MathStar is developing several different object types to include in the array. Over time, the range of object types will be expanded but the intent is that a relatively small number of object types will satisfy a wide range of applications. Also, MathStar will offer a family of platform chips that vary in the total number of objects and in the mix of object types on a particular member of the family.

Provide a heterogeneous collection of programmable hardware elements tightly interconnected by a high-performance, flexible communication structure.

Density is achieved through arrayed blocks of functionality that have routing already built in and completed. These blocks are regular, custom designed, structures that can be used to take full advantage of the density of 130 nm, while keeping the problem tractable for design. Silicon Objects allow designers to use 130 nm density very effectively by changing the level

of abstraction and reducing the size of the overall design problem. The performance metrics achieved through the use of 130 nm can be used to affect either a very high throughput, low latency design or to temporally stage the use of object resources to economize the total object count requirement. In addition, the high performance communication structure allows significant spatial freedom in assigning functionality to objects.

Use Full Custom design techniques to increase performance to the Gigahertz range and reduce costs to be competitive with ASICs.

The relatively small size of each Silicon Objects means Full Custom techniques can be used in the design. This allows the critical timing paths of the object to be optimized to achieve gigahertz clock rates with no wait states or pipelining stages inserted. In a similar manner, the regular, fixed spacing of the communications matrix means it also can be efficiently designed in Full Custom. The combination of these architectural features means that each object type needs to be custom designed only once. Multiple instantiations of each type can readily be placed within the array with very little additional design effort. The medium grained nature of the silicon object architecture allows for the design and implementation of a robust and high performance clock scheme (1Ghz) and for the design of skew and cycle tolerant physical design. As a result, we believe the best way to take full advantage of the tremendous performance and density of 130 nm is to implement an ultra high performance medium grained reconfigurable architecture.

Summary

There are tremendous advantages to the newest IC process technologies, but to get to these advantages, the conventional approach of ASIC and FPGA must change. The density and interconnect performance limits imposed by

these traditional design approaches is too expensive for applications that have even a moderate production volume. There are new techniques, enabled by 130 and 90 nm, being developed that change the fundamental cost and performance equations. This architecture embodying techniques of medium grained, high performance, reconfigurable processing elements is being developed today by MathStar and will bring the benefits of advanced processing technology to a wide range of existing and future applications.