**Lattice** ®
**Semiconductor**
**Corporation**
**Bringing the Best Together**

# DELIVERING FPGA-BASED PRE-ENGINEERED IP USING STRUCTURED ASIC TECHNOLOGY

## MACO
### Masked Array for Cost Optimization

A Lattice Semiconductor White Paper
February 2006

**Lattice Semiconductor**

**5555 Northeast Moore Ct.**

**Hillsboro, Oregon 97124 USA**

**Telephone: (503) 268-8000**

**www.latticesemi.com**

# A Silicon Survey

In the communications and networking markets, designers face a number of competitive pressures: time-to-market, bandwidth, port density and protocol compliance. Since networking equipment evolves quickly and continuously (e.g., DSLAM, cable head-end, optical switch, wireless base station), quick time-to-market confirms that the proverbial "early bird gets the worm." Increasing bandwidth and protocol compliance (e.g., 10GbE, GFP, etc) are non-negotiable requirements for all customers. Port density is a factor that directly affects the economic value-added for a piece of networking equipment. While all these pressures squeeze designers from different directions, a variety of silicon technologies are available to help designers carry the load.

Gate Arrays have been an inexpensive alternative that provides reasonable time-to-market. However, gate arrays are typically offered in process technologies that are 2-3 generations old (.25μ or .18μ CMOS), severely limiting density and performance while consuming lots of power. These silicon dinosaurs are still useful in cost-reduction situations, but are poorly suited for high-performance communications systems.

Application-specific standard products (ASSPs) offer designer off-the-shelf availability at the expense of customization and differentiation. Additionally, they typically require custom logic to serve as a bridge from one ASSP to another. The standards that these devices manage are being moved as intellectual property (IP) blocks into ASICs and FPGAs.

Standard cell ASICs have long been the choice for communications systems designers due to their performance, density and support for intellectual property portfolios. However, time-to-market is always compromised with ASICs, and design tool suites are complex and expensive. The real limiting factor at 90nm is the non-recurring engineering (NRE) charges. These charges include mask set cost, support engineering charges and sample device expenses. It is estimated that a 90nm design can easily exceed $3million (US) per design spin. That's an expensive risk to take when less expensive alternatives are available.

SRAM-based FPGAs have become a favorite of designers in the communications space because of their $0 NRE and inexpensive design tool costs. Their biggest advantage over competing technologies is time-to-market. SRAM-based FPGAs have always been at the leading edge of the technology curve, with several companies already offering 90nm devices.

FPGAs have steadily gained market share at the expense of all other options as densities and performance continue to increase.

Structured ASICs have gained in popularity recently due to their density and performance relative to SRAM-based FPGA devices.  Unlike full-custom or standard cell ASICs, structured ASICs cost far less to design because they require only one to seven metal layer changes to accomplish their task.  This results in significantly lower NRE than full custom or standard cell ASICs, as well as quicker turn-around time.  A failing of structured ASICs is that they are manufactured in processing technologies that are several generations old.  Most structured ASICs available today are fabricated in $0.18\mu$ or $0.13\mu$ CMOS, limiting their usefulness in multi-gigabit communication systems.

Recognizing significant bandwidth, time-to-market, port density and protocol pressures in the communications market, Lattice Semiconductor has introduced a new silicon technology to meet the challenges for next-generation systems: the LatticeSCM.

## *LatticeSCM FPGAs*

The LatticeSCM family of FPGAs combines a high-performance FPGA fabric, 3.8Gbps SERDES, high-performance I/Os, large embedded RAM and embedded ASIC blocks (MACO) in a single industry-leading architecture.  This FPGA family is fabricated on a state of the art 90nm technology to provide one of the highest performing FPGAs in the industry.

This family of devices also includes specific features to meet the needs of today's communication network systems.  These features include SERDES with embedded advance PCS (Physical Coding Sublayer), up to 7.8 Mbits of sysMEM embedded block RAM and dedicated logic to support system level standards such as RapidIO, HyperTransport, SPI4.2, SFI-4, UTOPIA, XGMII and CSIX.  The LatticeSCM devices feature clock multiply, divide and phase shift PLLs, numerous DLLs and dynamic glitch free clock MUXs that are required in today's high end system designs.  High speed, high bandwidth I/O makes this family ideal for high throughput systems.  And, for higher-performance, higher density logic, the LatticeSCM family offers up to 12 embedded MACO blocks per device.

## Table 1 - LatticeSC Family Selection Guide

| Device | Logic Block LUT4 | EBR 18Kb SRAM Blocks | EBR SRAM Mbits | SERDES 3.8Gbps | MACO Blocks | Analog PLL | DLL | Max User I/O |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| LFSC15 | 15,168 | 56 | 1.03 | 8 | 4 | 8 | 12 | 300 |
| LFSC25 | 25,424 | 104 | 1.92 | 16 | 6 | 8 | 12 | 484 |
| LFSC40 | 40,366 | 216 | 3.98 | 16 | 10 | 8 | 12 | 562 |
| LFSC80 | 80,080 | 308 | 5.68 | 32 | 10 | 8 | 12 | 904 |
| LFSC115 | 115,200 | 424 | 7.8 | 32 | 12 | 8 | 12 | 942 |

Figure 1 illustrates the layout of a LatticeSCM chip, with the MACO blocks available on the periphery of the device.

## Figure 1- Layout of Lattice LFSC15 with 4 MACO blocks
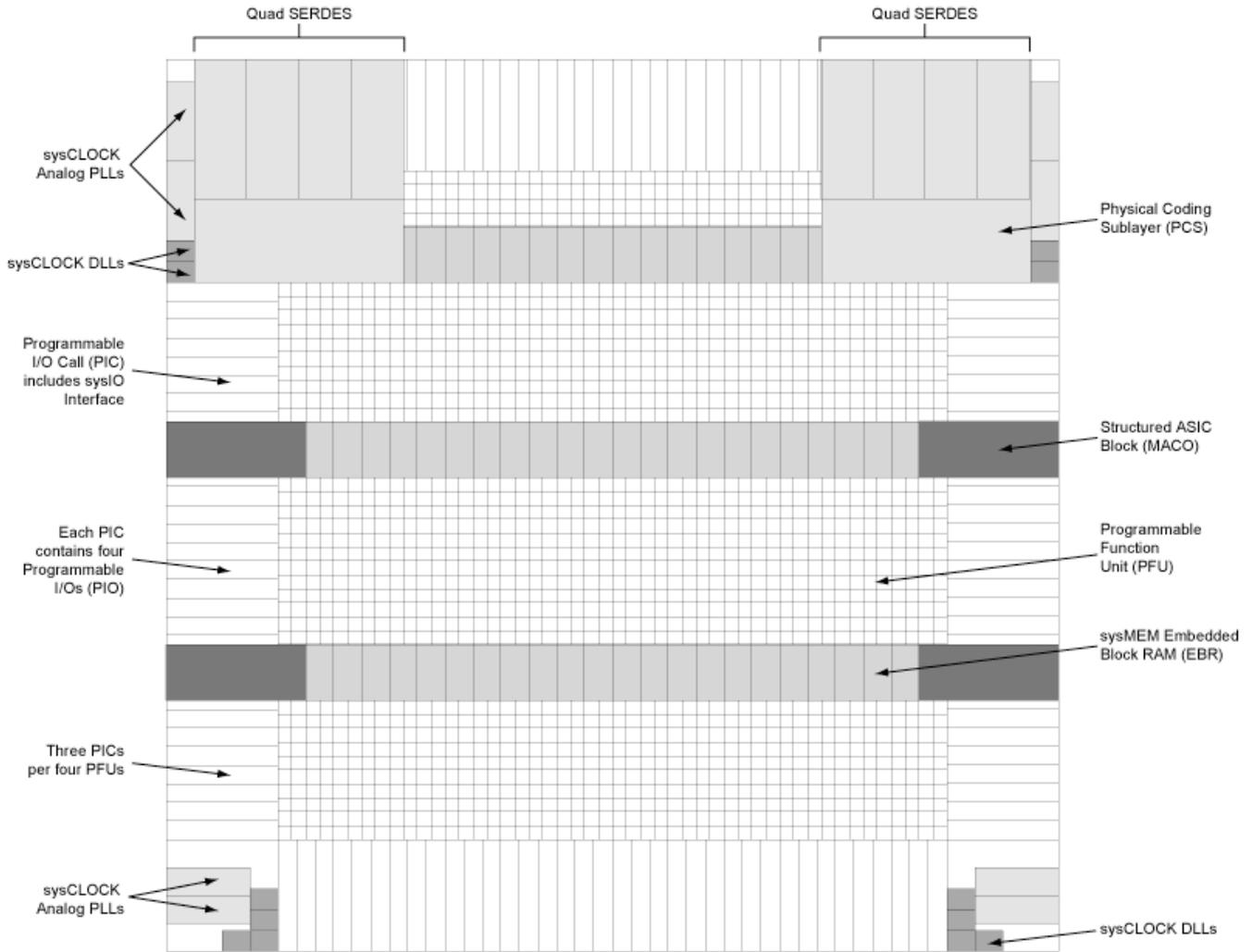


The major architectural elements listed in Figure 1 are described in greater detail in Table 2. For a comprehensive view of the LatticeSCM architecture, please refer to the LatticeSCM Data Sheet, which can be found on the Lattice Semiconductor web site at www.latticesemi.com.

**Table 2 - Major FPGA architectural features on the LatticeSC**

| Feature | Description | Function |
|---|---|---|
| SERDES | **SER**ializer-**DES**erializer | Embedded transceiver that converts parallel data to serial data and vice-versa. The transmitter section is a serial-to-parallel converter, and the receiver section is a parallel-to-serial converter. Connects to LVDS PICs on periphery of chip, and PCS on FPGA side. |
| PCS | **P**hysical **C**oding **S**ublayer | Embedded block that contains logic to simultaneously perform alignment, coding, de-coding and other functions. May be bypassed to form SERDES-FPGA direct connect. |
| PLL | **P**hase **L**ocked **L**oops | Provides the ability to synthesize clock frequencies. Each PLL has four dividers associated with it: input clock divider, feedback divider and two clock output dividers. The input divider is used to divide the input clock signal, while the feedback divider is used to multiply the input clock signal. |
| DLL | Digital **D**elay-**L**ocked **L**oops | Similar to PLLs, DLLs assist in the management of clocks and strobes. DLLs are well suited to applications where the clock may be stopped or transferring jitter from input to output is important, for example forward clocked interfaces. Used for clock injection match, duty cycle correction, and single delay cell. |
| EBR | **E**mbedded **B**lock **RAM** | Large, dedicated, fast memory blocks. They can be configured as RAM, ROM or FIFO. These blocks have dedicated logic to simplify the implementation of FIFOs. |
| PFU | **P**rogrammable **F**unction **U**nit | Primary logic cell that can be programmed to perform Logic, Arithmetic, Distributed RAM and Distributed ROM functions. |
| PIC | **P**rogrammable **I/**O **C**ell | Each PIC contains four programmable I/O buffers that are then connected to the I/O pads. The PIO block supplies the output data and the Tri-state control signal to I/O buffers, and receives input from the buffer. The PIO contains advanced capabilities to allow the support of speeds up to 2Gbps. |
| CIB | **C**onfigurable **I**nterface **B**lock | This block serves as the interface between PLC, PIC and EBR blocks. It has a routing block, and a logic block. The CIB logic block can buffer signals and generate control signals for other blocks. |

The LatticeSC architecture allows FPGA designers to approach performance levels previously available only with full custom or standard cell ASICs. Table 3 illustrates some commonly used benchmarks implemented in the LatticeSC FPGA fabric (no MACO gates used).

**Table 3 - Performance Benchmarks for LatticeSC FPGA Fabric**

| Function's Performance | (MHz) |
|---|---|
| 32-bit Address Decoder | 522 |
| 64-bit Address Decoder | 496 |
| 32:1 Multiplexer | 561 |
| 64-bit Adder (ripple) | 328 |
| 64-bit Counter (up or down counter, non-loadable) | 376 |

## *Masked Array for Cost Optimization (MACO)*

The layout of the LatticeSCM FPGA is a regular and homogeneous array of programmable logic cells (PFUs) surrounded by programmable I/O cells (PICs). At the top of the device are embedded SERDES channels that connect to embedded multi-purpose physical coding sub-layer (PCS) blocks for managing high-speed serial data transfers. The PCS block can be

Delivering FPGA Based Pre-Engineered IP Using Structured ASIC Technology

A Lattice Semiconductor White Paper

bypassed to transfer serial data directly to the FPGA fabric.  Rows of embedded block RAM (EBR) are "striped" across the array for efficient connectivity to the PFUs.  Special configurable interconnect blocks (CIB) contain dedicated resources for routing signals to/from the block RAM.  At the end of each EBR row (see Figure 1) is an area of silicon that Lattice has made available for a "structured ASIC" block, allowing designers the ability to commit logic to high-performance, high-density 90nm arrays.  Lattice calls this concept the Masked Array for Cost Optimization (MACO).

The LatticeSCM family provides designers access to pre-engineered, high performance IP blocks designed in 90nm structured ASIC blocks.  Combined with a state-of-the-art FPGA array and world-class SERDES technology, these blocks offer the most flexible and high-performance programmable platform available today.

## MACO Block Architecture

Figure 2 illustrates a block diagram for each of the MACO sites on a LatticeSCM FPGA.  The major features of each block include three 64x40 RAMs, a MACO interface block (MIB) for managing FPGA-to-MACO connectivity and a sea of gates for implementing digital logic.

**Figure 2 - MACO block diagram with adjacent FPGA features**
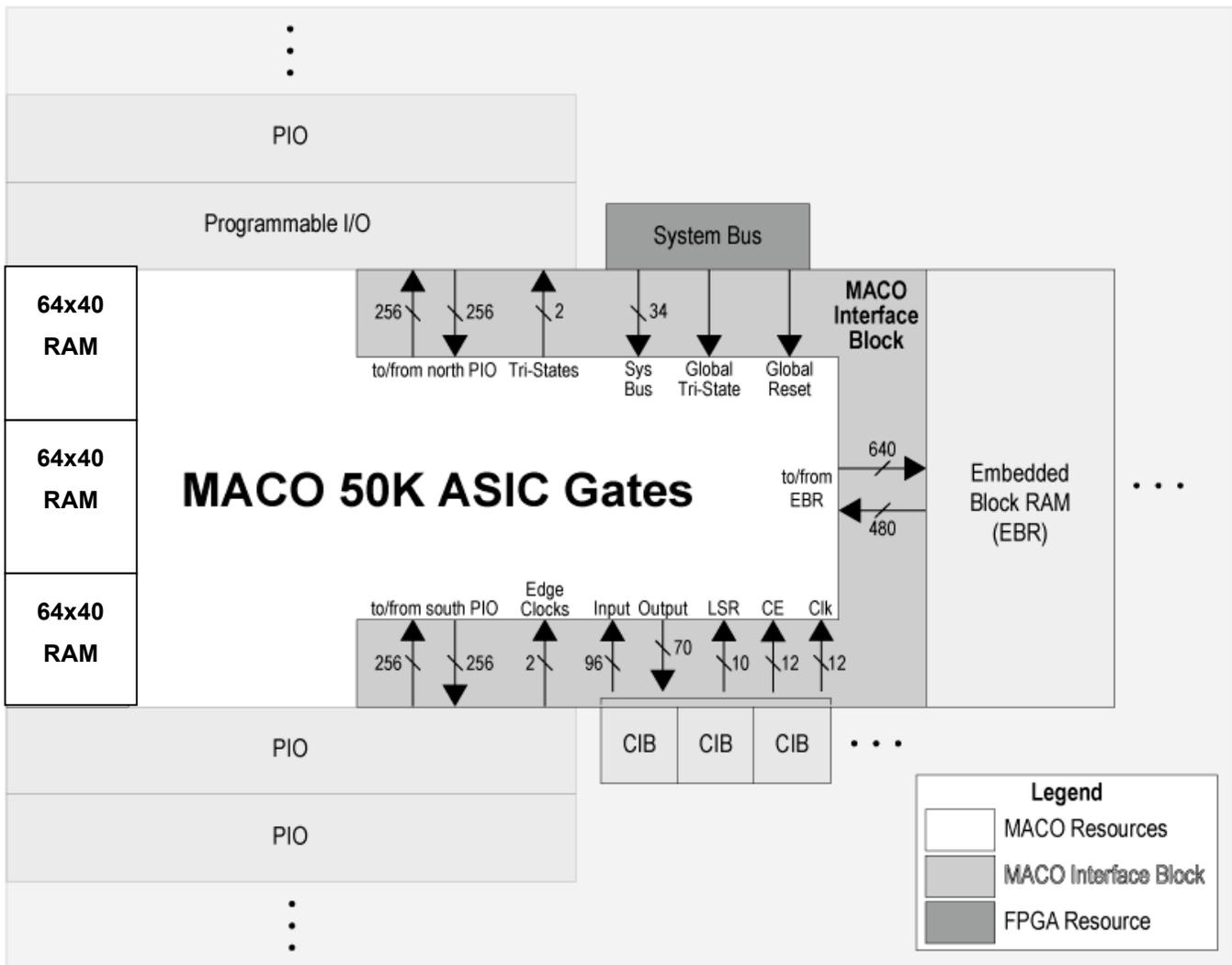


## *64x40 RAMs*

Each MACO block contains three 64x40 asynchronous RAMs with dual-port addressing that permit simultaneous reading and writing of data. Dedicated latches are on all address, data, and enable ports. Synchronous read and write operations are permitted through the read and write ports, respectively. The 64x40 memory blocks can be initialized during FPGA device configuration using a pre-load bus that sits between the 64x40 memory blocks and the rest of the MACO hard IP logic that access memories during normal operation.

These 64x40 memories are intended to augment the EBR in the FPGA fabric, which are available to MACO logic through the MACO interface block (MIB).

## *MACO Interface Block*

The MACO interface block (MIB) acts as a bridge between the MACO blocks and the rest of the FPGA fabric.  All MACO signals must either originate from or route through the MIB.  Table 4 lists the MIB resources accessible to the designer and their function.

**Table 4 - MACO Interface Block connections**

| Connection | Description | Number of Connections |
|---|---|---|
| MACO ⇔ EBR | MACO is able to directly access up to 10 EBRs through these connections. If MACO does not connect to EBR, some of these connections can be used to connect to FPGA logic/routing. | • MACO Inputs from EBR: 48x10=**480**<br>• MACO Outputs to EBR: 64x10=**640**<br>• If no EBR used, up to 480 inputs and outputs can connect to FPGA fabric |
| MACO ⇔ PIO | Connects MACO to PIOs for I/O routing. If MACO is unconnected to I/Os these connections are used as route-throughs, allowing PIO routing to cross over MACO block. | • PIOs above MACO: **256** In, **256** Out<br>• PIOs below MACO: **256** In, **256** Out |
| MACO ⇔ CIB | Connects MACO to PFU logic array. Connections include I/O data signals, clocks, local resets, and clock-enables. All connections to/from PFU or routing go through CIBs. | • MACO data inputs from CIBs: **70**<br>• MACO data outputs to CIBs: **96**<br>• Clocks inputs to MACO: **12**<br>• Clock-enable inputs to MACO: **12**<br>• Local reset inputs to MACO: **10** |
| Edge Clocks | The LatticeSC devices have 8 edge clocks around the periphery of each array for the purpose of facilitating high-speed I/O. | • Four (**4**) four edge clock inputs that can be driven from any of the eight (8) edge clocks. |
| Tri-States | The tri-states are available for connection to the global reset (GSR). | • Two (**2**) outputs connect to GSR |

The primary connectivity between the FPGA fabric and MACO block is through a CIB.  This resource limits the designer to 96 data outputs and 70 data inputs with additional ports available for resets, clock-enables and clocks.  However, if EBR adjacent to the MACO block are unused, then up to 480 MACO ⇔ EMB connections may be used for highly flexible connectivity to the FPGA PFU array.

For high-speed I/O connections, 512 I/Os on each MACO block are defined for PIO connectivity.  The I/O capabilities on LatticeSC devices operate at up to 2Gbps, and with ample connectivity to fast MACO gates, this combination of technologies offers the industry's highest throughput solution.

## *MACO Gates*

At the heart of the MACO block is a sea of 50,000 usable ASIC gates.  A library of cells was created with Fujitsu's 90nm CMOS process technology and optimized for speed, power dissipation and area.  The library contains 257 cells for combinatorial, sequential, and special-purpose functions.

Note that there are no I/O cells in the MACO cell library. They are not required because FPGA IOs are used for the MACO core chip-level I/Os. Additionally, there is no memory compiler or memory cells (other than the three 64x40 RAMs) for MACO because the LatticeSC FPGA fabric contains ample, flexible embedded memory with excellent connectivity to each MACO block.

## The Advantages of MACO

### FPGA vs. MACO Performance

MACO provides multiple advantages: speed, density and lower power dissipation. Because MACO is implemented in 90nm ASIC gates, the performance increase over SRAM-based lookup-table architecture can exceed 100%.

The best way to compare the performance advantage of a 90nm cell-based technology to a 90nm LUT-based technology is to take a design fragment and implement it in both. For this white paper a 32-bit cyclic redundancy check (CRC) calculation was selected that employs a dense PRBS sequence chosen to produce a large logic tree. The code is written in Verilog HDL and is included in Appendix A.

The design was first run through the Lattice ispLEVER v5.1 design tool targeting the Lattice LFSC25 FPGA, a new, 90nm device that offers the industry's highest performance. The design was mapped, placed and routed with minimal effort and a 300MHz timing preference on the output clock. A static timing report was generated to analyze the worst-case path to determine a maximum operating frequency. A portion of the report from ispLEVER's TRACE tool is shown in Figure 3.

**Figure 3 - TRACE static timing report for CRC calculation in LatticeSC FPGA fabric.**

```
================================================================================
Preference: FREQUENCY NET "oclk2" 300.000000 MHz ;
            802 items scored, 0 timing errors detected.
--------------------------------------------------------------------------------

Passed:  The following path meets requirements by 0.358ns

 Logical Details:  Cell type  Pin type        Cell name  (clock net +/-)

   Source:         FF         Q               dec/XZ0Z_8  (from oclk2 +)
   Destination:    FF         Data in         dec/X_19  (to oclk2 +)

   Delay:               2.900ns  (21.7% logic, 78.3% route), 6 logic levels.

 Constraint Details:

      2.900ns physical path delay SLICE_7 to SLICE_1 meets
      3.333ns delay constraint less
      0.000ns skew and
      0.075ns DIN_SET requirement (totaling 3.258ns) by 0.358ns

 Physical Path Details:

    Name    Fanout   Delay (ns)          Site                  Resource
  REG_DEL    ---      0.305      R29C8B.CLK to      R29C8B.Q0 SLICE_7 (from oclk2)
  ROUTE       5       0.728      R29C8B.Q0 to       R28C9A.B0 dec/XZ2Z_8
  CTOF_DEL   ---      0.065      R28C9A.B0 to       R28C9A.F0 SLICE_59
  ROUTE      10       0.614      R28C9A.F0 to       R28C7C.B1 dec/GZ0Z_3
  CTOF_DEL   ---      0.065      R28C7C.B1 to       R28C7C.F1 SLICE_11
  ROUTE       3       0.374      R28C7C.F1 to       R27C7C.C1 dec/GZ0Z_17
  CTOF_DEL   ---      0.065      R27C7C.C1 to       R27C7C.F1 SLICE_13
  ROUTE       1       0.344      R27C7C.F1 to       R27C8A.C1 dec/un921_X_0_1Z0Z_0
  CTOF_DEL   ---      0.065      R27C8A.C1 to       R27C8A.F1 SLICE_12
  ROUTE       1       0.210      R27C8A.F1 to       R27C8B.C0 dec/un921_X_0Z0Z_0
  CTOF_DEL   ---      0.065      R27C8B.C0 to       R27C8B.F0 SLICE_1
  ROUTE       1       0.000      R27C8B.F0 to       R27C8B.DI0 dec/XZ1Z_20 (to oclk2)
                     --------
                      2.900 (21.7% logic, 78.3% route), 6 logic levels.

Report:  336.134MHz is the maximum frequency for this preference.
```

As with all SRAM-based FPGAs, routing delay tends to dominate critical path delay. For this circuit, actual logic is implemented in 6 levels constituting only 21.7% of the critical path delay. The routing delay consumes the remaining 78.3%. Nonetheless, for a circuit with a rather large logic tree, 6 levels of logic resulting in 336MHz performance is outstanding. Note: the max frequency is calculated as 1/(Path_Delay + Setup_Requirement).

The same code was targeted at the MACO technology, also available on the LatticeSC FPGA. Synopsys Design Compiler (v 2004.06.1) was used to synthesize the design and Cadence SOC Encounter (2004.1) was utilized for place and route. After parasitic extraction, the design

**Figure 4 - PrimeTime static timing report for CRC calculation implemented in LatticeSC**

```
****************************************
Report : timing
        -path full_clock
        -delay max
        -input_pins
        -nets
        -nworst 4
        -max_paths 5
        -transition_time
        -capacitance
Design : crc_hard
Version: V-2004.06
Date   : Wed Jul 13 16:53:08 2005
****************************************

  Startpoint: X_reg_31_V (rising edge-triggered flip-flop clocked by clk)
  Endpoint: X_reg_3_V (rising edge-triggered flip-flop clocked by clk)
  Path Group: clk
  Path Type: max                                        Clock PERIOD

  clock clk (rise edge)                        0.000    3333.000    3333.000
  clock source latency                                     0.000    3333.000
  clk (in)                                    12.690       4.033 +  3337.033 r
  clk (net)                       1    0.008
  clk_bufi/A (ckx4m4mce)                      13.368       0.131 *  3337.165 r
  clk_bufi/Z (ckx4m4mce)                      61.442      53.255 *  3390.419 r
  clk_buf (net)                   6    0.089
  clk_buf__L1_I0/A (ckx4m4mce)                79.958      30.227 *  3420.646 r
  clk_buf__L1_I0/Z (ckx4m4mce)                62.793      67.937 *  3488.583 r
  clk_buf__L1_N0 (net)           15    0.085
  X_reg_3_V/CK (mxbffprqnx1m4mce)             74.650      22.550 *  3511.134 r
  library setup time                                     -43.727 *  3467.406
  data required time                                                3467.406
  ----------------------------------------------------------------------------
  data required time                                                3467.406
  data arrival time                                                -1574.362
  ----------------------------------------------------------------------------
  slack (MET)                                                       1893.044
```

**Max Frequency = 1 / (3333-1893.044) = 694 MHz**

was analyzed in Synopsys PrimeTime. A portion of the PrimeTime report is shown in Figure 4.

The MACO version of this implementation resulted in a critical path that contained 17 logic levels. Minimal effort was applied during the synthesis phase, resulting in the use of many 2-input combinatorial gates. Nonetheless, without much effort, this design was able to achieve 694MHz max frequency, in large part due to the low routing delays associated with a cell-based design. Note: the max frequency is derived as 1/(Clock_Period – Slack).

Table 5 summarizes the results of the performance comparison. Even with inefficient packing and almost 3x the number of logic levels in the worst-case path, the 90nm cell-based MACO technology easily outperformed the 90nm SRAM-based LUT technology by over 100%.

**Table 5 - Performance comparison for LatticeSC FPGA vs. MACO technologies**

| Technology | Max Frequency | Logic Levels |
|---|---|---|
| LatticeSC FPGA | 336.134 MHz | 6 |
| LatticeSCM MACO | 694 MHz | 17 |

## *FPGA vs. MACO Area*

Many FPGA designs contain intellectual property (IP) designed to handle chip-to-chip communications standards. For instance, SPI4.2 is a popular packet interface employed on many MACs and network processors. Since it is an established standard, it is desirable to implement this function in hard gates (i.e., MACO) rather than use valuable FPGA gates. Other FPGA vendors offer SPI4.2 solutions as soft IP that consume FPGA resources and make user designs more difficult to map, place, route and achieve timing.

A sample application using an FPGA plus a SPI4.2 interface is an Ethernet-to-SONET bridge. In Figure 5 the FPGA is used to manage the bridge, including flow control and packet buffering, usually with a memory controller and external buffer. If the SPI4.2 interfaces in this example have to be implemented in soft IP using FPGA gates, the design becomes considerably more difficult and may require a larger device. In this case, the real value-added is not the SPI4.2 interface but the bridging functions.

**Figure 5 - FPGA as a SONET-to-Ethernet bridge employing SPI4.2**
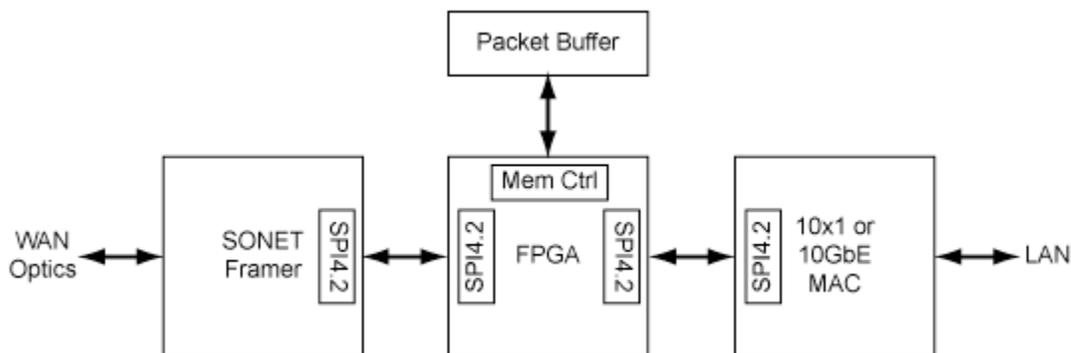


The LatticeSCM FPGA employs MACO technology to implement communication standards such as SPI4.2. The blocks are pre-engineered, resulting in a significant up-front development savings for the designer. The FPGA gates are available to the designer for value-added and differentiating functions. For an example of the savings for the example, refer to Table 6.

**Table 6 - Resources required to implement SONET-to-GbE bridge**

| Block | LatticeSC FPGA w/MACO | Competitor FPGA |
|---|---|---|
| User Design LUTs | 20,000 | 20,000 |
| 2 x Memory Controller LUTs | 0 | 5000 |
| 2 x SPI4.2 LUTs | 3000 | 16000* |
| Total LUTs Required | 23,000 | 41,000 |
| Device Required | LFSC25 | XC4VLX40 |
| Device Utilization | 92% | 100% |

*Xilinx SP4.2 v7.4 Core for Virtex4: DS209

The LFSCM25 includes two high-speed memory controllers implemented in MACO and not requiring many LUTs. The SPI4.2 MACO cores utilize approximately 1500 LUTs each to manage the SPI4.2 status path and ingress/egress FIFOs in the data path. The FIFOs are implemented in EBR, since MACO has little embedded RAM. Essentially, the bulk of the SPI4.2 data path is committed to MACO, while the user-accessible status path consumes just a handful of LUTs. This is so that the customer is able to modify the status controller if desired. As a comparison, a competitor device requires over 20K LUTs just to implement this industry-standard IP.

Lattice studies have found that each MACO core is equivalent to at least 5,000 LUTs, while occupying only 10% of the area of an equivalent FPGA implementation. With multiple MACO blocks per device, this results in a substantial cost savings to the designer, both by lowering the development cost and saving significant silicon area.

## *FPGA vs. MACO Power*

In today's high performance systems, power dissipation budgets are tightening even as clock rates and data paths rapidly increase. As a cell-based technology, MACO power consumption is significantly less for a given design at an equivalent operating frequency. For a comparison of power dissipation, we will again use a SPI4.2 core, a common packet interface found in communication systems.

The Lattice SPI4.2 solution using the LatticeSCM FPGA utilizes MACO technology for the majority of a SPI4.2 core. Only 1500 LUTs and 10 EBRs are used in the LFSC FPGA fabric, primarily for FIFOs and the SPI4.2 status path.

Competitive SPI4.2 solutions use only FPGA gates to implement entire SPI4.2 cores. A survey of the power consumption for each implementation reveals the LatticeSC solution employing MACO uses considerably less power then competitive devices. This is due, in part,

to the power savings offered by MACO. Lattice studies on a variety of cores implemented in MACO technology show that MACO power, even when running at up to 700MHz, does not exceed 200mW per MACO block.

Table 7 shows the results of a survey of SPI4.2 core power in competitive SRAM-based FPGAs. The comparison employs the fastest speed grade in each case. With a SPI4.2 interface running at 1+ Gbps, much power is dissipated by the I/Os. The balance of power is generated by the core implementation. While Altera's SPI4.2 core uses less LUTs and embedded memory than Xilinx, it is at the expense of substantial power. While the Xilinx implementation is less power hungry than Altera's, 2.1W is a lot of power to expend for a standard interface. Adding another SPI4.2 core to a Xilinx solution would leave little power budget margin for value-added functionality.

**Table 7 - A survey of power for SPI4.2 cores in competitive FPGAs**

| Vendor | Max LVDS Rate | Speed Grade | LUT4s | Memory (Kbits) | Total Power @ Max Rate | Total Power @ 800Mbps |
|--------|---------------|-------------|-------|----------------|------------------------|----------------------|
| Xilinx[1] | 1+ Gbps | -12 (Fastest) | 8000 | 306 | 2.1 W | 1.7 W |
| Altera[2] | 1.04 Gbps | -3 (Fastest) | 6100 | 154 | ~4.4W | 3.4 W |
| Lattice | 1+ Gbps | -7 (Fastest) | 1500 | 180 | 1.05 | 0.85 W |

In contrast, the LatticeSCM solution greatly benefits from the low power dissipation of the MACO technology. In fact, a LatticeSC FPGA could implement four SPI4.2 cores for the same power budget that Altera requires for a single core.

## *Overview of System Connectivity*

Figure 6 below summarizes the basic roles of FPGAs on a typical communications line card. At a high level, these basic functions are common across the major industries served by FPGAs, which include telecom, datacom and storage. Figure 7 also highlights the many interfaces and standards that FPGAs are called upon to support.

---

[1] http://www.xilinx.com/xlnx/xebiz/designResources/contentContainer.jsp?key=spi-42_faq

[2] http://www.altera.com/products/devices/stratix2/features/st2-competitive.html

## Figure 6 - FPGAs in a typical Communications Line Card
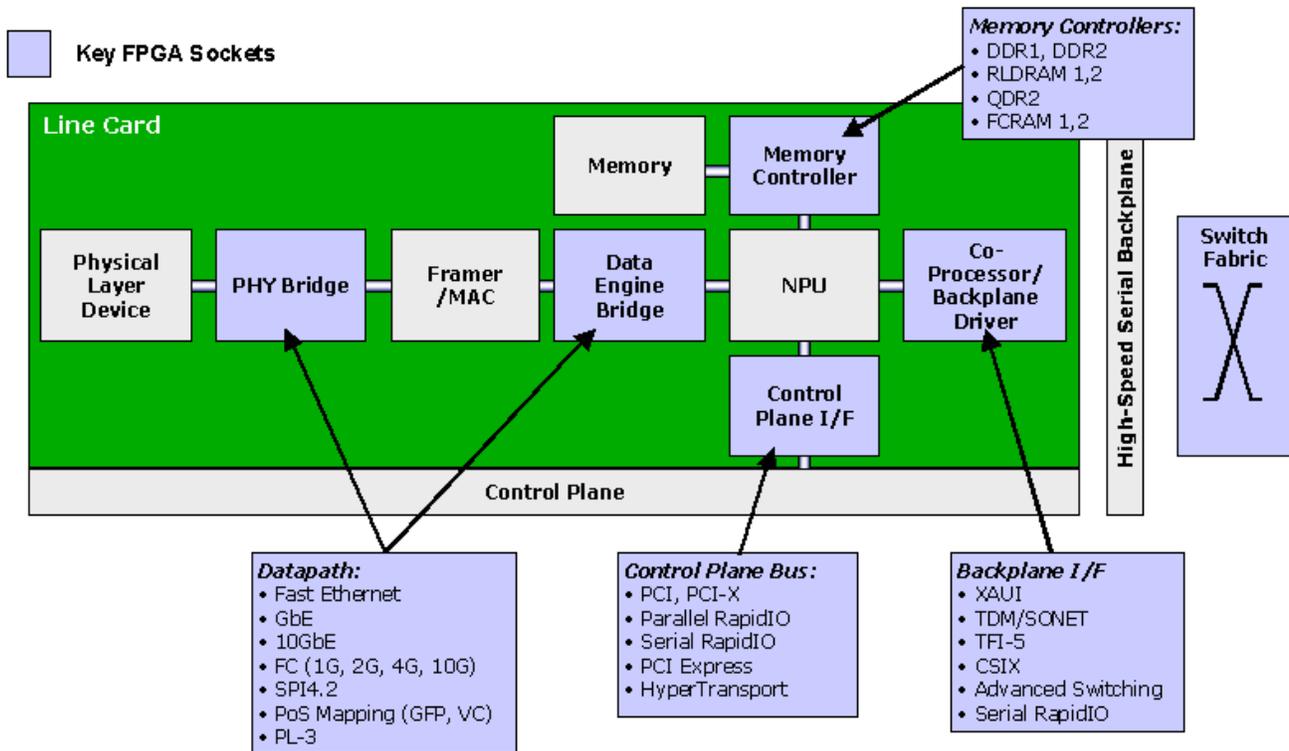


## Pre-Designed MACO Blocks

The initial LatticeSC devices will offer pre-designed IP based on industry-standard interfaces. Included on the initial LFSC80 devices will be memory controllers, SPI4.2 interfaces, and flexible MAC blocks supporting GbE, 10GbE and PCI Express protocols. Figure 7 illustrates the MACO block layout for the LFSCM25 FPGA.

**Figure 7 – LFSCM25 FPGA with MACO block layout.**
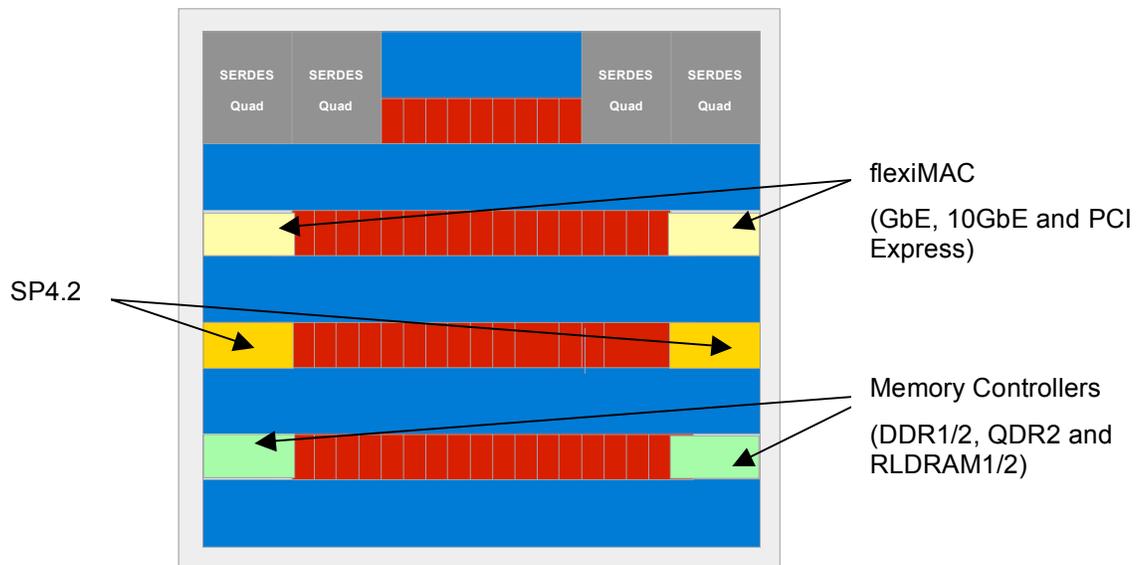


## Embedded Memory Controllers

The LatticeSCM devices provide dedicated high-speed memory controllers with MACO technology supporting the major high-speed memory standards implemented in many communications systems: DDR 1/2 SDRAM, QDR 1/2 SRAM, and RLDRAM 1/2.

Implementing high performance DDR memory interfaces requires very careful design of the read and write interface blocks of the memory controller. DDR2 memory devices pose a greater challenge due to their higher speeds of operation and the bi-directional DQS signal. The LatticeSC memory controller utilizes on-chip PLLs and DLLs, along with programmable delay elements at the input buffers to align DQS and DQ signals. These elements work together to compensate for process, voltage and temperature variations, providing reliable operation at all operating conditions and various frequencies. The LatticeSC devices also contain dedicated DDR register structures in the inputs (for read operations) and in the outputs (for write operations). All of these blocks are critical for implementing reliable high-speed DDR and DDR2 SDRAM Controllers. Typically, designers have problems implementing high-speed memory controllers in FPGAs because of the complexity of the DQS logic.

QDR SRAMs are very popular in low latency applications requiring simultaneous reads and writes. The LatticeSC devices also implement an embedded QDR SRAM memory controller

for interfacing to QDR2 SRAM memory devices.  The memory controller supports both 2- and 4-word burst modes.

Reduced Latency DRAM (RLDRAM) provides an SRAM-type interface with non-multiplexed addresses.  RLDRAM2 technology provides minimized latency and reduced row cycle times that are very suitable for applications requiring critical response time and very fast random accesses, such as next generation  (10 Gbps and beyond) networking applications.  The LatticeSC implements on-chip RLDRAM memory controllers in MACO supporting both RLDRAM1I and RLDRAM2 devices.  Additionally, the LatticeSCM memory controller supports both types of RLDRAM2 memory: Common I/O (CIO) and Separate I/O (SIO).

**Figure 8 - LatticeSCM DDR input for dedicated Memory Controller support**
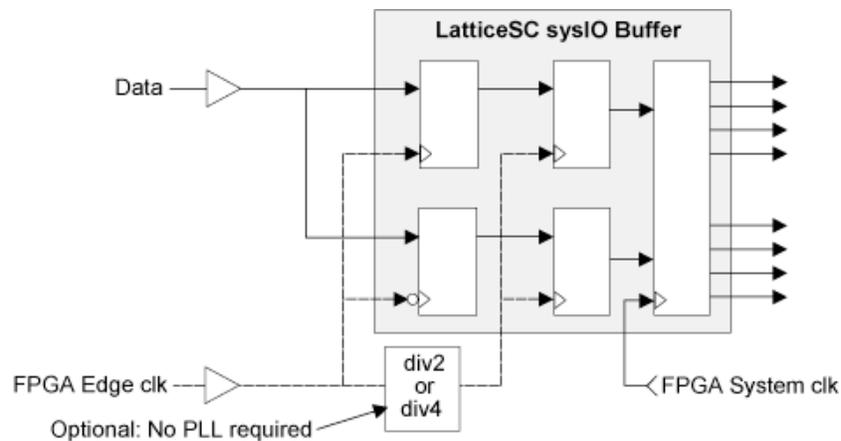


## SPI4.2 Interface Cores

Initial LatticeSCM FPGAs will incorporate one or two SPI4.2 cores implemented in MACO. Each is fully compliant with the OIF-SPI4-02.0 specification, offering up to 256 logical ports, with transmit/receive data paths that are 16-bits wide with in-band port address, SOP, EOP indication and error control.  Each SPI4.2 MACO core interfaces directly with LatticeSCM LVDS I/Os that offer source synchronous double edge clocking at 311MHz minimum and in static or dynamic alignment modes.  The core supports up to 1 Gbps with dynamic phase alignment and up to 700 Mbps in static alignment mode.  The SPI4.2 MACO core interfaces with the FPGA via a 128-bit user accessible data path.  Only 1500 FPGA LUTs and 10 EBR are used, primarily for buffering FIFOs and status path management.

Delivering FPGA Based Pre-Engineered IP Using Structured ASIC Technology

A Lattice Semiconductor White Paper

**Figure 9 - SPI4.2 implemented in LatticeSCM FPGA with MACO**
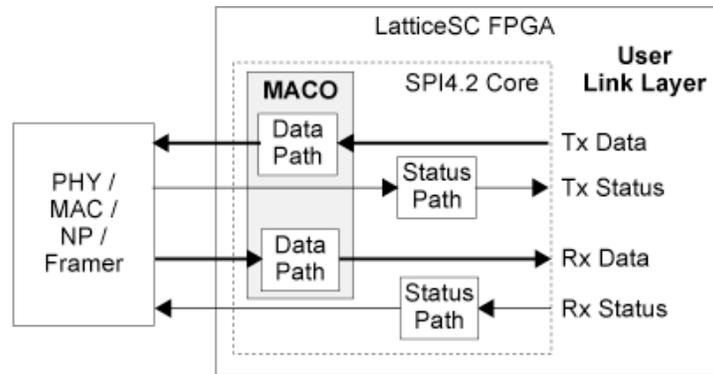


## *flexiMAC*

flexiMAC is a flexible packet framer and parser that can implement Layer 2 (Data Link Layer or MAC) functionality for various standards. Implemented in MACO technology, the flexiMAC functionality complements the LatticeSCM SERDES and the Layer 1 (Physical Layer) multi-protocol functionality of the Physical Coding Sublayer (PCS). This results in a complete Layer 1/Layer 2 solution for 1G/10G Ethernet standards and provides customers with integrated 1GE/10GE solutions without using up valuable FPGA gates.

**Figure 10 - Functional block diagram of flexiMAC in LatticeSCM FPGA**
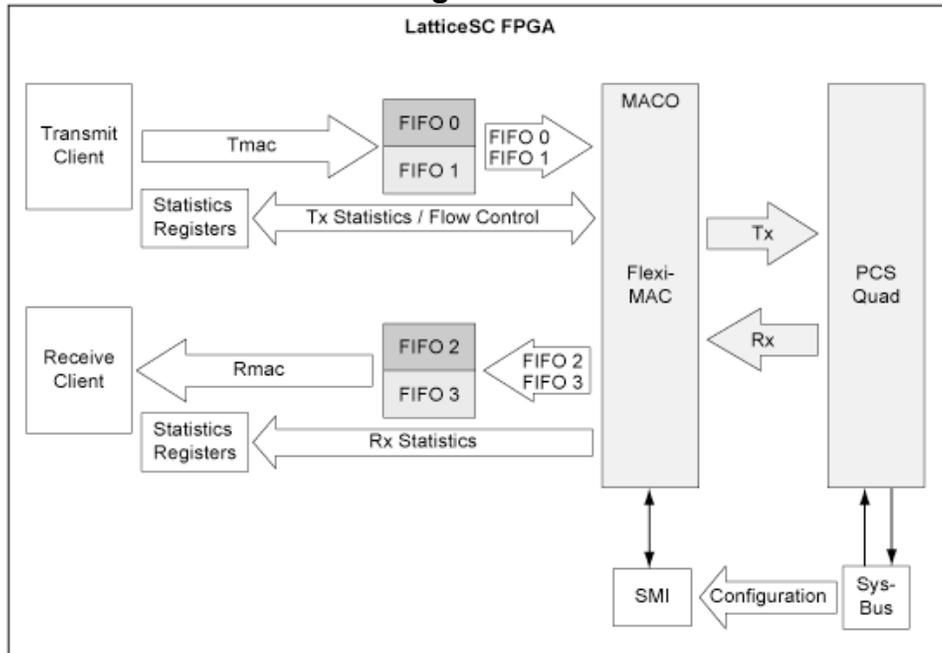


# *Conclusion*

Lattice has introduced a new silicon technology to meet today's ever-increasing demand for performance, density, low power and time-to-market. Called MACO, this technology adds a

Delivering FPGA Based Pre-Engineered IP Using Structured ASIC Technology

A Lattice Semiconductor White Paper

90nm structured ASIC capability to an advanced FPGA + SERDES platform. The MACO gates provide superior density and performance with significantly lower power dissipation. MACO is ideal for industry-standard IP such as memory controllers, SPI4.2, and MACs that are necessary for today's communications design, but cannot be optimized in soft IP. This results in substantial development cost reduction because standards are committed to dedicated silicon and the FPGA gates are reserved for value-added and differentiating features.

MACO conserves power by implementing IP previously targeted for LUT-based architectures in 90nm cell-based technology. MACO blocks do not exceed 200mW per site, even at 700MHz performance. This type of power dissipation can't be matched in 90nm LUT architectures running at similar clock rates.

MACO conserves area by shrinking 5,000 equivalent LUTs into a much smaller silicon area. With multiple MACO blocks per device, this significantly boosts device density. Since MACO is ideal for industry-standard IP cores, this means that valuable LUT-based silicon is reserved for value-added design features.

Finally, MACO improves device performance. Since MACO is a 90nm cell-based technology like a structured ASIC, it is capable of > 700Mhz performance with little design effort. Ample connectivity is provided to connect each MACO to LatticeSCM I/O technology, as well as embedded RAM and programmable logic blocks.

###