DESIGNING 2Gbps PARALLEL I/O

WITH THE LatticeSC FPGA

A Lattice Semiconductor White Paper

February 2006

Lattice Semiconductor
5555 Northeast Moore Ct.
Hillsboro, Oregon 97124 USA
Telephone: (503) 268-8000
www.latticesemi.com

Introduction

As I/O standards continue to evolve towards serialization in both backplane and, more recently, chip-to-chip applications, high speed parallel I/O still has an important role in specific chip-to-chip applications in which current serial technologies are cost prohibitive.

Apart from established NPU, framer and module based source synchronous I/O standards such as SPI4.2, SFI4.1, XGMII, HyperTransport, Rapid IO and CSIX, the next generation of clock forwarded interfaces are being implemented on SRAM and DRAM memories (DDR1/2/3, RLDRAM1/2 and QDR2) as well as ADCs and DACs. The I/O speeds required for these next generation applications are expected to exceed 1Gbps.

FPGAs are being increasingly used as programmable SoCs, designed in as an integral part of the system data path. However, with this usage comes the expectation that these devices are capable of performing high-speed I/O translation and processing. As programmable ASSPs, there is also the expectation for compliance with past, existing and emerging I/O standards.

Conversely, the inherent nature of FPGAs requires that they serve a broad base of applications, and so are not typically tuned to the highly sensitive requirements of specific source synchronous I/Os to the same degree as their ASSP and ASIC counterparts. The question becomes: how can this level of performance be achieved in an FPGA array, given its universal applicability and lack of specialization?

Although electrical compliance and high-speed signal integrity are required features, these alone do not address the bandwidth issue. The FPGA I/O must also have circuitry to manage and maintain the clock and data relationships of these high speed signals, as well as provide the necessary “gearbox” functionality in order to support the transfer of the high speed I/O data to the FPGA fabric to perform the desired processing functions.

FPGA vendors have tackled this problem in various ways, using pre-engineered, dedicated I/O circuitry to perform these specialized functions. The driving objective is to provide a transparent scheme that maximizes performance yet minimizes the need for board level tweaks, reducing development time and cost.

This white paper examines how the LatticeSC PURESPEED I/O architecture addresses these system level concerns by delivering the highest performing and most feature rich source synchronous I/Os of any existing or announced FPGA family, surpassing even its own previous best-in-class FPSC-based I/Os.

Electrical Performance

As system speeds continue to increase, more demands are placed on device I/O buffers. In order to achieve increased performance, the chip manufacturer must take care to ensure linearity and robust signal integrity. This includes not only the actual buffer design, but also critical layout techniques that must be considered. Additionally, the buffers must also address designer concerns such as power, pin capacitance and drive strength, to name a few. Combine these issues with the flexibility that is required of an FPGA in order to support the numerous emerging and legacy industry standards, and it becomes clear why the I/O buffer has become a key driver in programmable technology.

The I/O buffers of the LatticeSC have been designed to address all of the above concerns, and more. With features such as duty cycle balancing, true dedicated differential drivers/receivers and on-chip termination capability, the LatticeSC FPGA is equipped to tackle the most stringent system level interface requirements. The following sections will discuss the details of these buffers.

Buffer Performance

In terms of buffer design, there are many considerations in order to achieve the performance levels required of I/O in today's systems. The LatticeSC buffers have numerous design features that result in their industry leading performance.

One key element to high performance I/O is obviously the transistor design itself. The LatticeSC transistor has been optimized for 2.5V operation, while still providing 3.3V capability. This approach offers consistently higher performance, especially for lower voltage standards (1.8V, 1.5V, 1.2V), an area where the performance of other transistor designs falls off.

Another key factor in buffer performance is pin capacitance. With clock periods in the nanosecond range, rise and fall times are, to say the least, critical. If pin capacitance is too high, a major portion of the clock period will be lost due to the slow rise time, severely limiting overall buffer performance. To combat this, the LatticeSC I/O pads have reduced capacitance (see Table 1), allowing Lattice to achieve best-in-class performance of 2Gbps.

Pin Description	*LatticeSC*	Virtex 4
User IO Vertical (top/bottom)	7.5 pF	12.5 pF
User IO Horizontal (left/right)	7.5 pF	12.5 pF

Note: The Lattice SC supports high speed Inputs (LVDS, RSDS, HyperTransport) on all sides of the device, while high speed Outputs are available on the left and right sides only. Support for 3.3V Inputs/Outputs is available in the top and bottom banks of the device.

Table 1 - Pin capacitance comparison

Other features that enhance signal integrity include an externally driven voltage reference option, which provides greater signal integrity of bias circuits, duty cycle balancing for all circuits in the I/O path for high performance DDR interfaces (400MHz) and true, dedicated LVDS drivers and receivers for clean data eyes and exceptional sensitivity at data rates up to 2Gbps. Figure 1 is a capture of the PURESPEED I/O LVDS buffer data eye performing at a 2Gbps DDR data rate on the LatticeSC demonstration board.

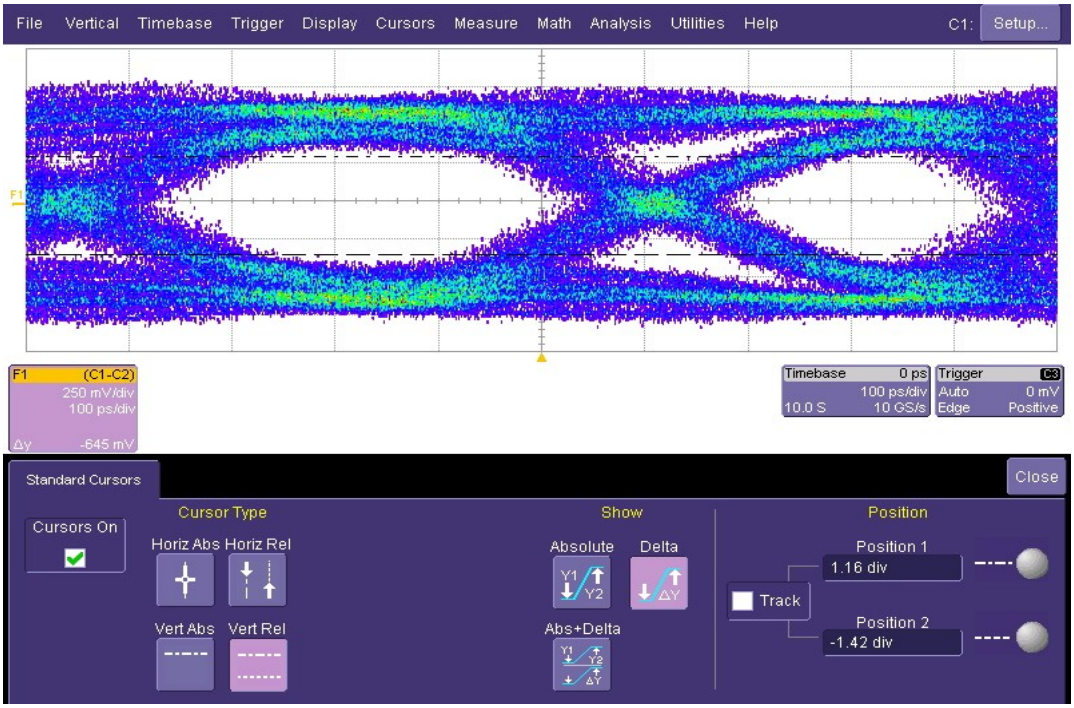

Figure 1- PURESPEED I/O LVDS buffer data eye (2Gbps)

Versatile Standards Support

As a result of this excellent buffer performance, the LatticeSC is able to support numerous single-ended and differential buffer types. Table 2 provides an overview of the electrical characteristics for the various buffers.

Single Ended (all speed grades over PVT)

I/O Standard	Voltage	Programmable drive strength	Clock	Pin Throughput
PCI/PCI-x33 (w/clamp)	3.3V	pre-determined by spec	66/133 MHz	66/133 Mbps
PCI-x15	1.5V	pre-determined by spec	266 MHz	266 Mbps
HSTL (class I,II)	1.5V/1.8V	pre-determined by spec	400 MHz	800 Mbps
SSTL (class I,II)	1.8V/2.5V/3.3V	pre-determined by spec	400 MHz	800 Mbps
LVCMOS	1.2V/1.5V/1.8V/2.5V	2mA,4mA,6mA,8mA	200 MHz	400 Mbps
LVCMOS/LVTTL	3.3V	8mA,16mA,24mA	200 MHz	400 Mbps
* GTL+	(open drain)	8mA,16mA,24mA	66/100/133 MHz	66/100/133 Mbps

Differential (all speed grades over PVT)

I/O Standard	Voltage	Programmable drive strength	Clock	Pin Throughput
LVDS	2.5V	2mA,3.5mA,4mA,6mA	1 GHz	2 Gbps
RSDS	2.5V	2mA,3.5mA,4mA,6mA	400 MHz	800 Mbps
HyperTransport	2.5V	3.5mA,4mA,6mA	400 MHz	800 Mbps
** LVPECL	3.3V	24mA	200 MHz	400 Mbps
Differential HSTL	1.5V/1.8V	pre-determined by spec	400 MHz	800 Mbps
Differential SSTL	1.8V/2.5V/3.3V	pre-determined by spec	400 MHz	800 Mbps
*** BLVDS/MLVDS	2.5V	pre-determined by spec	200 MHz	400 Mbps

* - GTL+ support requires multiple output buffers combined in open drain mode. GTL+ inputs are fully supported.
** - LVPECL output support is emulated using LVCMOS 3.3V outputs w/external resistors. LVPECL inputs are fully supported up to 1GHz.
*** - Outputs emulated, inputs fully supported up to 1GHz.

Table 2 – PURESPEED I/O Electrical Characteristics

The wide range of buffer and voltage types available on the LatticeSC enables the support of numerous emerging and legacy standards. Table 3 lists the various industry standards supported by the LatticeSC. Industry leading buffer performance, combined with embedded I/O logic, offers the system designer a complete I/O solution that addresses the needs of high performance designs.

I/O Standard	*Buffer Type*	*Bus Width*	*Data Rate*	*Clock Frequency*	*Pin Throughput*
Generic LVDS	LVDS	Variable	DDR	1 GHz	Up to 2 Gbps
RAPID I/O	LVDS	8, 16	DDR	125 - 500 MHz	250 Mb -1 Gbps
HyperTransport	LVDS	2 - 32	DDR	200 - 400 MHz	400 - 800 Mbps
SPI-4 (PL4)	LVDS	16	DDR	311 - 450 MHz	622 - 900 Mbps
SFI-4 / XSBI	LVDS	16	SDR	622/645 MHz	622/645 Mbps
XGMII	HSTL	32	DDR	156 MHz	311 Mbps
CSIX L1	HSTL	32 - 128	SDR	250 MHz	250 Mbps
QDR SRAM	HSTL	Variable	DDR	300 MHz	600 Mbps
DDR I / II SDRAM	SSTL2	Variable	DDR	200 / 300 MHz	400 / 600 Mbps
PCI / PCI-x	LVTTL	32/64	SDR	66 / 133 MHz	66 / 133 Mbps
RLDRAM I / II	HSTL	Variable	DDR	300 / 400 MHz	600 / 800 Mbps

NOTE: All parallel inputs supported up to 2Gbps

Table 3 - PURESPEED I/O Standards Support

On-Die Termination

On-die termination is a desirable feature that allows superior signal integrity due to the lack of package inductance and receiver reflections inherent in board-based termination schemes. Using the proper termination scheme is essential to maintaining good signal integrity, since improper terminations can lead to poor quality due to reflections or signal attenuation. To address this, on-chip impedance matching offers serial and parallel termination on single-ended I/Os and differential termination on differential I/Os. The obvious benefit of on-chip termination is that it eliminates the board space needed for external termination resistors and, more importantly, minimizes both board and package reflections. Although this feature has been offered in the past, the lack of programmability and, more importantly, linearity of the terminations has limited its usefulness. Additionally, traditional on-

chip termination schemes have been limited to Thevenin equivalent implementations, resulting in higher power dissipation for the device.

The LatticeSC changes all of that by offering multiple, PVT compensated on-die termination options that have the flexibility, lower power and exceptional linearity required without compromise.

Thevenin Equivalents

Accepted as an industry standard, Thevenin equivalents are a widely accepted termination option because no additional on-chip voltage sources are needed. Many Thevenin solutions are offered by the LatticeSC, providing designers with a wide variety of methods to match the termination impedance of their systems. Although this termination scheme is popular due to the simplicity of using existing voltages, it does come with the price of higher power dissipation.

Vtt (True Termination)

As an alternative to the Thevenin equivalents, the LatticeSC offers a "true termination" option. As shown in Figure 2, this scheme involves a "true" (not a Thevenin equivalent value) resistance tied to a dedicated voltage rail. In some cases, this can reduce power by up to 70% over the Thevenin equivalent, making this a very desirable implementation. As previously stated, this scheme does require an additional voltage rail (Vtt), but Lattice is the only FPGA vendor that offers this lower power option.

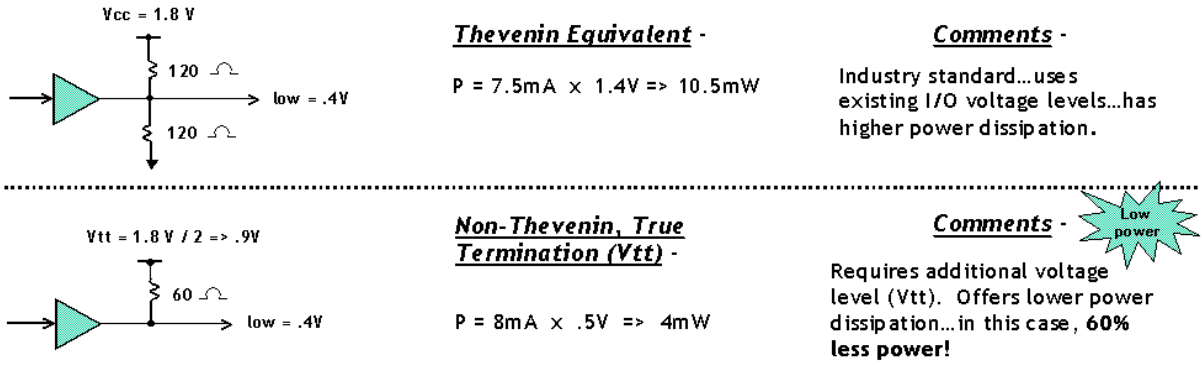

Figure 2 - Power comparisons for termination schemes

Switchable Terminations

As memory interfaces continue to increase in performance, so does the need for high-speed, on-die switched terminations that are located as close to the driver as possible, ensuring a higher quality signal than if the terminating resistor was placed on the board. Additionally, memory interfaces such as DDR2 require dynamic "on/off" switching of the termination that is off when driving, on when receiving. The LatticeSC is the only FPGA that offers a switched "true termination" that meets the needs of these high performance interfaces. As mentioned above, not only does the switched true termination scheme reduce on-chip power by up to 70%, but when compared to board based implementations, the tri-state controlled switching of the LatticeSC also achieves the performance necessary to support these high speed memory interfaces. If the interface is symmetric (ie 50% input, 50% output) this reduces the termination power by an additional 50%.

Programmable Common Mode and CTAP

Another user option is a programmable common mode implementation for differential receivers. This mode provides an LVDS termination resistor and an optional center tap (CTAP) capacitor on-chip. The CTAP capacitor is used to hold the common mode, since the DC balance will not be perfect and will tend to drift, offering unparalleled signal quality.

Linearity

While other FPGA suppliers offer on-chip terminations, historically the linearity of these resistances as a function of voltage has been poor, resulting in inferior signal integrity in high-speed operations. In the case of differential I/O, this non-linearity can contribute to different edge rates between signal pairs, impacting timing and reducing the data valid window. If these issues are not addressed at the device architecture level, the advantages of on-chip termination become meaningless.

In the LatticeSC, the linearity of on-chip termination resistance has been specifically considered and designed into the device architecture, resulting in industry leading performance that is available on a per pin basis for added flexibility. As an example of this, Figure 3 shows the measured I/V curve of the LatticeSC on-chip termination

resistor. This exceptional linearity provides true transmission line matching and the improved signal integrity needed at 2Gbps data rates.

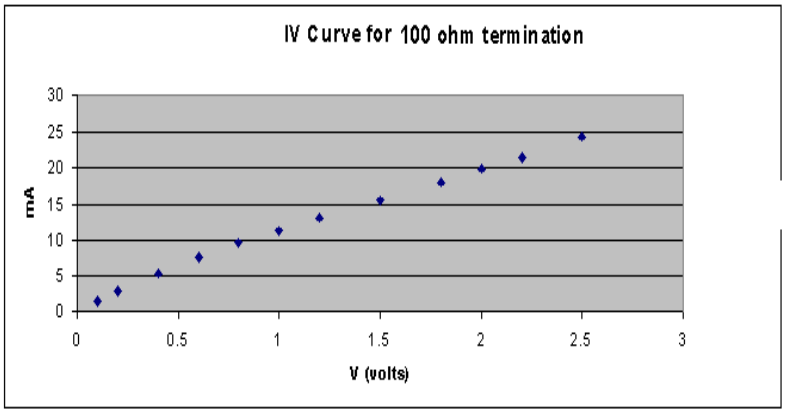

Figure 3 - I/V curve for 100 Ohm differential termination

Simultaneous Switching Noise (SSN)

When multiple output drivers switch simultaneously, the relative ground voltage within the device is raised momentarily and the power supply drops, resulting in the phenomenon known as ground bounce, or simultaneous switching noise (SSN). It is clear that with data rates on the rise and package pin counts continuously increasing, this issue will become more and more prevalent.

The designer can alleviate or mitigate the effects of SSN with various board level techniques, the most practical being intelligent I/O placement and strategic decoupling/filtering of the I/O. Although helpful, this approach does have its limitations, since in reality it is the symptom, rather than the source of the problem, that is being treated. The following paragraphs will highlight some of the techniques the LatticeSC has employed to address the phenomenon of SSN at its source: the package and silicon level.

Ground pin placement has been optimized in the LatticeSC package to balance both SSO performance and board level routing for the user. Ground pins have been distributed throughout the package's ball array to minimize ground return loop inductances and to optimize signal integrity. Within the package, the signal layers have been referenced to internal ground planes in order to provide continuous return paths.

The PURESPEED I/O buffer itself is both PVT and slew rate controlled. The PVT control maintains constant buffer drive strength to limit the detrimental effects of buffer overdrive for fast switching conditions. The drive strength control, combined with the buffer's linear response, allows customers to match board line impedances for optimal signal integrity. This optional slew rate control can be selected to limit the di/dt switching current during signal transitions, without impacting the drive strength of the I/O buffer.

On-die decoupling has been included on the LatticeSC device to allow for the immediate sourcing of current during simultaneous switching events. These capacitors improve signal integrity and improve on-die power regulation for greater timing control of wide buses. In addition to these on-die capacitors, it is recommended that users follow Lattice guidelines on the selection and placement of board level de-coupling capacitors.

Finally, good board design should not be overlooked in the quest to obtain optimal signal integrity and SSO results. Signal layers on the board need reference planes with reference vias placed close to any signal vias. Return loop inductance paths must be minimized with no reference plane discontinuities. Also, the stacking of ground planes and power planes together effectively produces low ESR board level de-coupling and is highly recommended.

Flexibility

In many cases a programmable device is utilized in bridging applications, in which one of the most critical features to be considered is the need to support the multitude of emerging and legacy I/O standards. Pure I/O count is critical, but the ability of the I/O to have the flexibility necessary to support various buffer types, such as single ended and differential with voltage ranges from 3.3v down to 1.2v, has become a driving factor. Without proper flexibility, the designer may be forced to select a larger, more costly device/package combination to address the I/O needs.

Most programmable devices today have I/O architectures that are based on a banking scheme, in which typically there is only a single input voltage (Vref) available for referenced input buffers. This approach not only limits the user in terms

of board layout, but also forces all I/O in that bank to run off of that common voltage reference level, essentially dedicating the bank to a given I/O standard. But what happens if there is a need to support multiple input signal references in a given I/O bank, such as SSTL 2.5V and HSTL 1.8V? With only one voltage reference available, the various I/O types must be mapped to "unique" banks that are dedicated to support that given voltage level, leading to very inefficient use of the available device I/O and making pin assignment extremely challenging.

To address this the LatticeSC supports multiple Vrefs within a bank, in which any pin in a bank can be used as a voltage reference, allowing maximum flexibility when assigning device pinout. Figure 5 shows an example of multiple Vrefs allowing two independent I/O types supported within a single bank.

Additionally, for output buffers, AC filtering plus power supply noise rejection and circuit optimization have been incorporated to allow both LVDS and single-ended outputs such as HSTL, to coexist in the same bank, resulting in a highly efficient architecture with few or no wasted I/O pins.

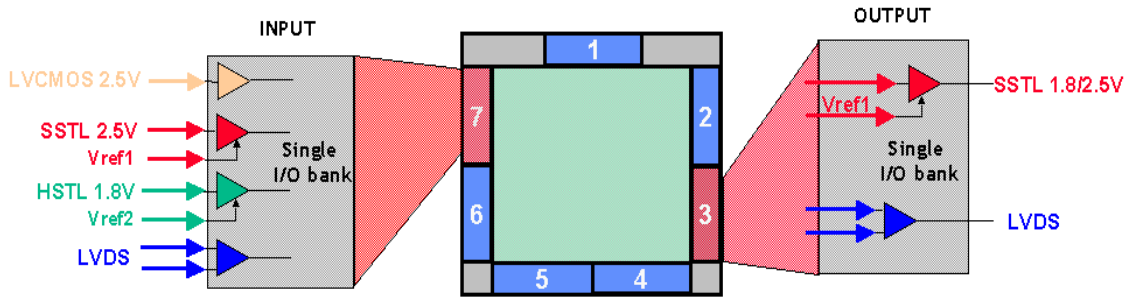

Figure 5 – Example of a flexible banking scheme for PURESPEED I/O

The flexibility of mixing and matching independent buffer types within a bank not only eases the task of pin assignment, but its efficiency also has the potential to allow for the use of a smaller device, leading to board space and cost savings.

Robust Chip Design Practices For Superior Signal Integrity

In addition to all the features designed into the best-in-class LatticeSC I/O buffers, careful design methodology also plays a major role in delivering 2Gbps I/O technology to the user.

Protection circuits are designed into the buffers that prevent inter-pad signal integrity issues/coupling due to overshoot and undershoot. Sufficient I/O power supply pins, combined with robust on-chip supply rails, help reduce inductive and resistive supply degradation.

Finally, the best design in the world can be rendered useless without the use of proper physical layout techniques. When used by experienced analog layout designers, techniques such as shielding, matching, common centroid and others provide the optimal signal integrity needed for high performance designs.

Embedded I/O Logic

In addition to the need for I/O buffers to achieve increasing levels of electrical performance, today's high-speed source synchronous interfaces also pose three other challenges for the designer:

1) Managing and maintaining the clock to data relationship
2) Managing the data to data skew (word alignment)
3) Clock domain transfer of these high-speed signals to the FPGA fabric

The data to data relationship (word alignment and deskew) portion is fairly straightforward and can be handled by FPGA logic, but the delay sensitive clock to data relationship and clock domain transfers are more challenging.

For bit and bus deskew, designers have traditionally relied on methods such as matching bus trace lengths or on PLLs and DLLs to manipulate the clock signal, eliminating clock injection delay and/or phase shifting the clock some pre-determined percentage of the clock cycle in order to maximize the clock to data relationship. While helpful, these approaches are not sufficient at higher speeds, since their compensation is clock-based and applied globally to all signals of the bus. These methods also have shortcomings in their compensation, since they are static and don't account for the delay variations that can occur over process, voltage and temperature. Today's high speed interfaces require bit-based compensation due to the increased difficulty of meeting and maintaining adequate setup and hold time margins for shrinking clock cycle times. This issue is exacerbated for high-speed

parallel protocols, such as SPI4.2, in which dynamic bit-based alignment and word alignment are key elements of the total system solution. Figure 6 shows the effects of dynamic bit alignment. Without any compensation, the clock is not centered on the data eye and barely meets the set up time for bit "A" on channel 0. Even worse, the incorrect data bit is sampled on channel 1. As can be seen, when bit-based compensation is applied, the clock is centered and the proper bit sampled for both channels.

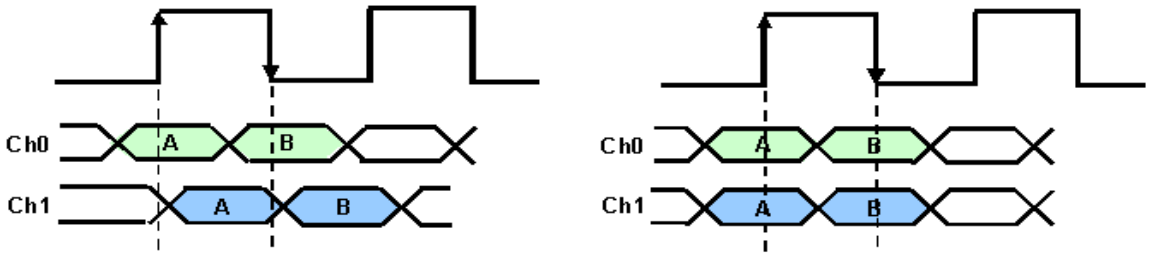

Figure 6 - Parallel bus skew and the effects of bit-based Dynamic Alignment

Gearbox

Once the data is sampled, the task of transferring the high-speed clock and data to the FPGA fabric still remains. In this case "gearbox" logic is required to ensure that the signals can be passed seamlessly to the FPGA fabric at the lower speeds that are suitable for further processing.

These can be daunting tasks for the system designer, taking a tremendous amount of design and verification time. The critical issue in this process is that the transfer of all bits on the high-speed bus must be transferred to the low-speed bus on the same low-speed clock edge, requiring this synchronization to be timed using the high-speed clock. This synchronization must be built into the FPGA as dedicated logic since it cannot be done reliably at high speeds using FPGA gates. The LatticeSC PURESPEED I/O logic block helps simplify this process by providing a complete, highly flexible and proven I/O system. It truly is a self contained I/O "system," with all of the embedded circuitry necessary to solve system level design challenges while improving system performance and reliability. All without requiring the use of any generic FPGA logic resources. The remainder of this section will examine the capabilities of this block in more detail.

Delay based alignment

While PLLs and DLLs can be used to align data and clock, the simplest way to address applications in which the clock to data relationship is well known is by utilizing an input delay block. For this purpose, the LatticeSC PURESPEED I/O logic block provides the user with a 144 tap (40ps step size, typical) delay block that can be used independently in two modes:

Static Alignment (INDEL) provides the user the simplest solution for data rates up to ~200Mbps. Delay settings are user defined and are pre-configured via the bit stream, based on pre-determined conditions. This mode, shown in Figure 7 below, is useful in applications in which the conditions are static and the removal of clock injection delay is desired for improved I/O performance.

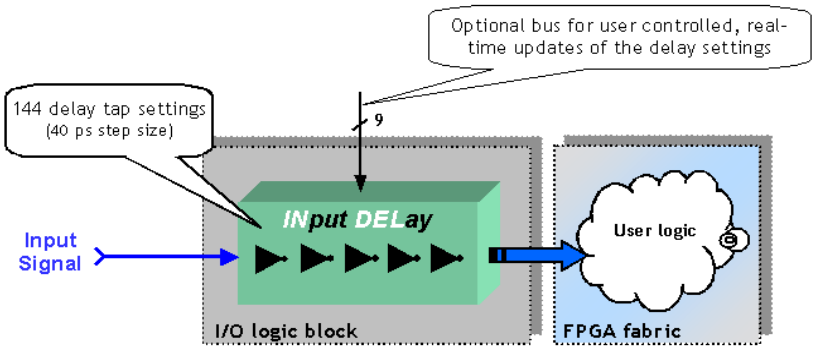

Figure 7 - PURESPEED I/O, Static Alignment mode

Bus-based Dynamic Alignment (INDEL + DLL) - a more robust solution compared to the stand-alone mode, combining the input delay block with a DLL to provide bus-based alignment capability for data rates up to ~600Mbps. This mode (Figure 8) preserves a fixed clock/data phase relationship by aligning the incoming clock and data bus under DLL-control. Another advantage of this mode is that it automatically tracks/compensates for delay variations due process, voltage and temperature.

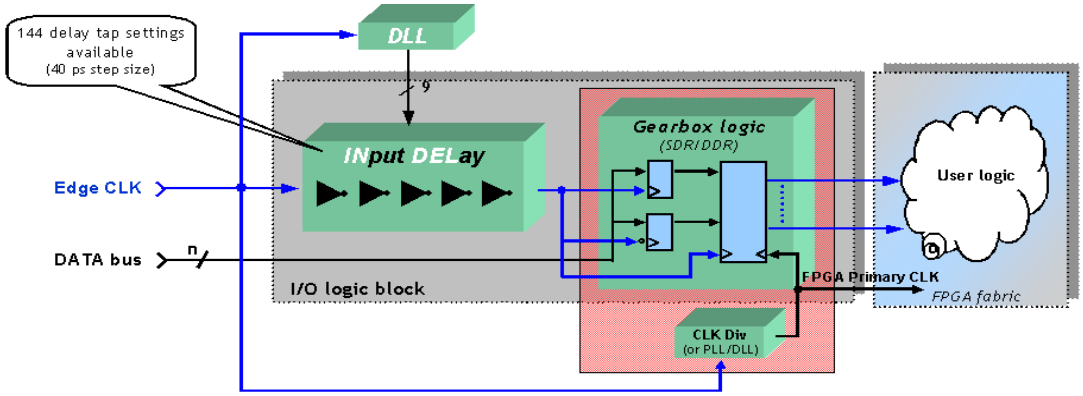

Figure 8 - PURESPEED I/O, Bus-based Dynamic Alignment Mode

Although both the Static and Bus-based DLL controlled modes are useful for some applications in which the clock to data relationship is known, they have inherent limitations when it comes to dynamic clock to data compensation for high speed, source synchronous interfaces. This is because the compensation in these modes is applied globally to all bits of the data bus, not allowing for the bit-based accuracy needed in applications above 600Mbps. So what about applications that require source synchronous interfaces running >600Mbps, such as SPI4.2…how do you handle that?

Delay based Dynamic Alignment with Adaptive Input Logic

For higher speed interfaces, bus-based alignment does not provide the precision necessary, since the delay compensation is applied globally. What is needed is a closed-loop, control and monitoring circuit that dynamically maintains proper setup and hold time margins on a bit by bit basis. For this reason, there is a third mode within the LatticeSC PURESPEED I/O block in which the input delay block is used in conjunction with embedded Adaptive Input Logic to tackle high-speed applications in which the clock to data relationship is unknown. This mode therefore handles PVT compensation not only on the LatticeSC device, but also handles the variations on the driving device.

Bit-based Dynamic Alignment (INDEL+ AIL) - the most robust configuration (Figure 9), the user can establish and dynamically maintain the clock to data relationship on a bit by bit basis, providing the proper resolution necessary to support speeds of up to 2Gbps on a single pin.

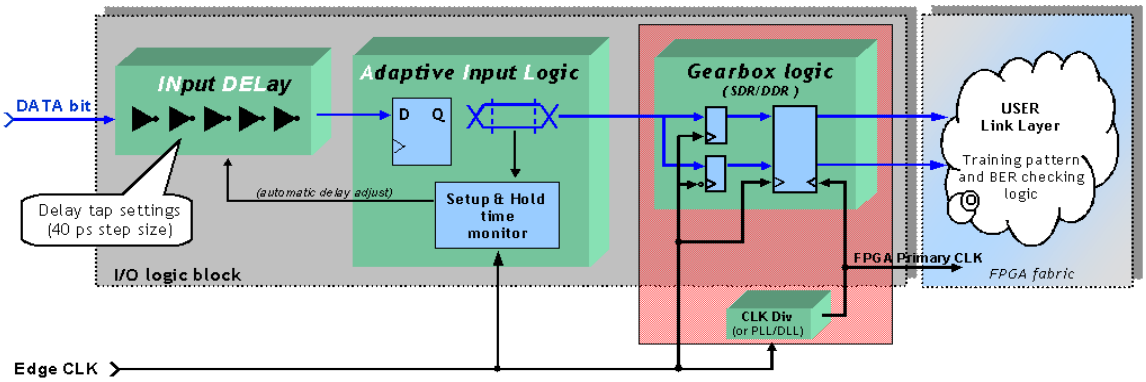

Figure 9 - PURESPEED I/O, Bit-based Dynamic Alignment mode

The key to this mode is the fact it is a self-contained, closed loop system that can be enabled/disabled or updated under FPGA control. The closed loop design also allows for tracking and compensating for delay variations due to process, voltage and temperature conditions. Here is an example to show how the LatticeSC dynamic alignment circuitry actually works. The SPI4.2 protocol will be used as a reference, since it's a popular, high-speed source synchronous interface that requires dynamic alignment.

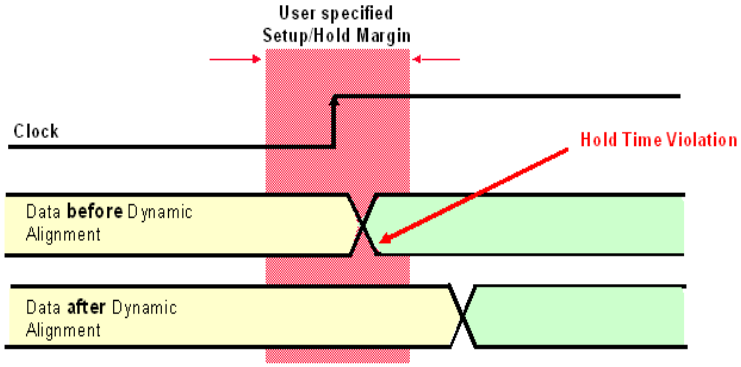

Figure 10 - Specified data 'window' for Bit-based Dynamic Alignment (AIL)

As seen in Figure 10, the user specifies a data valid window in which no transitions should occur. Because this is a closed-loop system, once these settings are made and the "window" established, the AIL circuit will continuously monitor and control the clock to data relationship of each bit to ensure no data transitions occur within those bounds. Figure 11 shows the benefit of the AIL circuit and the GUI used to configure the AIL circuit.

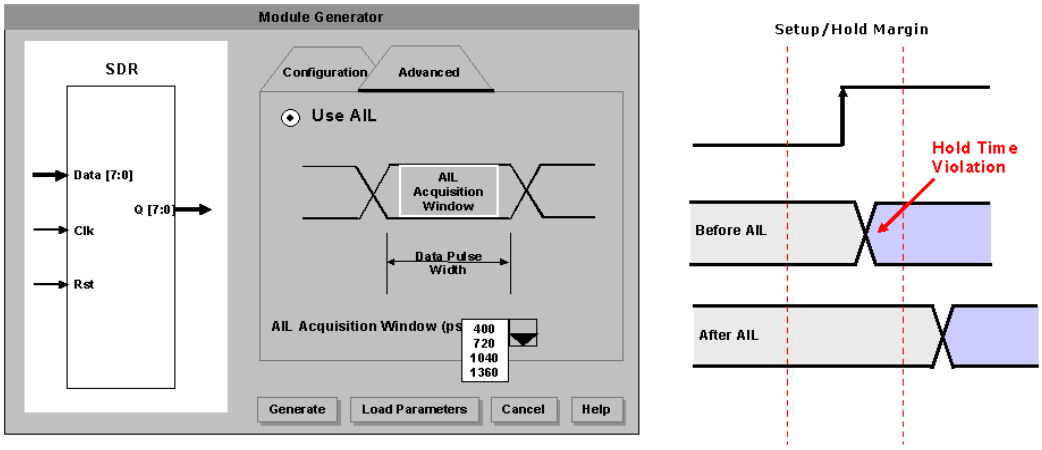

Figure 11 - GUI for Adaptive Input Logic (AIL) mode

Gearbox Logic (High speed Data and Clock Domain transfer)

Due to the high speed nature of these interfaces, gearbox logic must be utilized to slow these signals down to manageable speeds for the FPGA fabric. As shown in Figure 12, the LatticeSC PURESPEED I/O block provides this gearbox logic for either SDR or DDR interfaces.

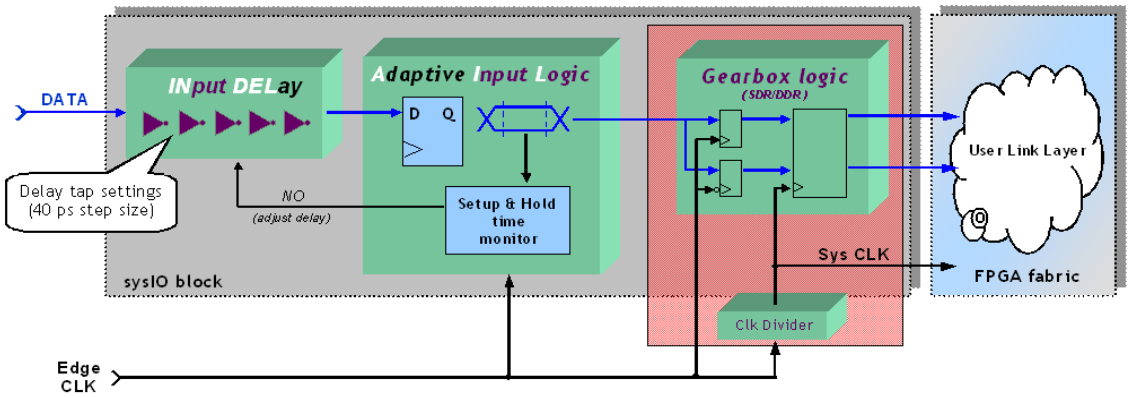

Figure 12 - PURESPEED I/O input with embedded Gearbox Logic

On-die clock dividers, in which both the divided and non-divided outputs are phase matched, are also provided to support the clocking requirements of the gearbox logic, alleviating the need to use generic PLL/DLL resources for this purpose. Table 4 below shows an example of the gearbox functionality. Another feature of the

gearing logic is to provide the proper domain transfer of the high speed edge clock to the lower speed FPGA system clock, guaranteed across process, voltage and temperature. Although an input example is shown, the gearing logic is available for outputs as well. These features will be discussed in more detail in the following section.

	Clock Frequency	Clock Type	SDR Bus	DDR Bus
At external pins	400 MHz at I/O	Edge Clk	400Mbps, 8-bits	800Mbps, 8-bits
1x Gearing	400 MHz in FPGA	Primary Clk	8-bits	16-bits
2x Gearing	200 MHz in FPGA	Primary Clk	16-bits	32-bits
4x Gearing	100 MHz in FPGA	Primary Clk	32-bits	64-bits

Table 4 - Example gearing for an 8-bit bus

Note: x1 gearing is used to ensure the guaranteed transfer of the high-speed edge clock to the FPGA system clock

PLLs, DLLs and Clocking

At clock speeds reaching up to 1GHz, it becomes imperative to have dedicated, low skew clock resources to support the PURESPEED I/O logic. In addition, the proper clock domain transfer of this high-speed data from the I/O block to the FPGA fabric must be ensured. Also, many applications benefit from both clock synthesis and phase compensation capabilities. The LatticeSC architecture provides the user complete flexibility without compromise by providing all of the above, including both PLLs and DLLs on chip, as well as phase matched Clock Dividers and dedicated clock routing/resources for the improved I/O and system level performance today's high speed interfaces require. Figure 13 gives a high level view of the clocking architecture for the LatticeSC.

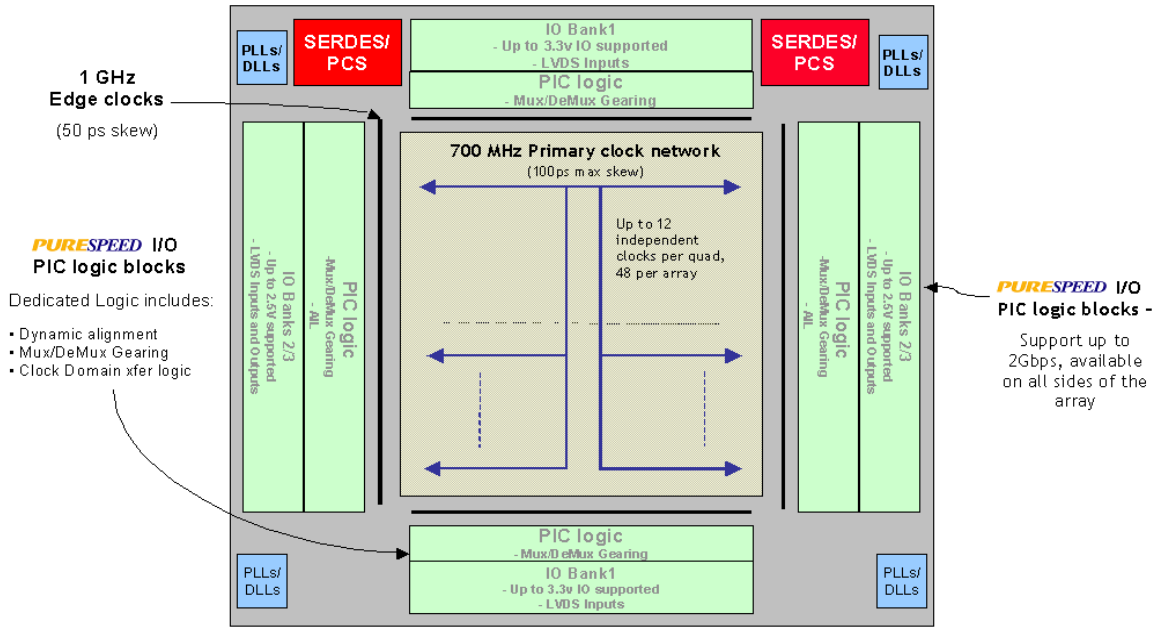

Figure 13 - LatticeSC clocking architecture overview

PLLs

The LatticeSC architecture supports 8 true analog PLLs operating at speeds up to 1GHz. In addition to its traditional use for frequency synthesis, a PLL can be utilized to fine tune clock timing for high-speed interfaces. It is ideal for the removal of clock insertion delay of a clock signal, reducing the clock to out latency. The LatticeSC PLL has a single clock input and two outputs and can also be used to create phase offsets between its two outputs, allowing the user to establish clock to data relationships in a forwarded clock scheme. When compared to a DLL, the high stability and low jitter of the PLL makes it ideal for these applications. A subset of the available PLL settings are register accessible to a master controller, such as a microprocessor or FPGA logic, when connected to the internal FPGA system bus. This provides the user with the ability to change the runtime behavior of the PLL through a dedicated memory map.

DLLs

In addition to the PLLs, the LatticeSC architecture supports 12 DLLs operating at speeds up to 700MHz. Because of their ability to track incoming jitter, DLLs are an excellent choice for managing incoming clocks. They can also be utilized for other applications such as clock injection delay removal in order to “fine tune” clock timing

for high-speed interfaces. Other uses of the DLL are to phase match two independent input clocks, which is useful when transferring data between two frequency locked clocks of unknown phase, and for supporting clocking schemes in which the clock is not continuous, such as memory interfaces.

As with the PLLs, the DLLs are also controllable via the FPGA system bus through a dedicated memory map.

Clocking

There are three types of clock routing networks available in the LatticeSC architecture. High speed, dedicated Edge clocks serve the needs of the I/O logic and are the only clocks that can support >700MHz. Traditional Primary and Secondary clock networks provide the clock distribution in the FPGA fabric. For the purposes of this white paper, the focus will be on the high-speed edge clocks.

There are a total of 40 Edge clocks on a given device. These are high speed (1GHz), low skew (50ps) clock routes located directly on the edges of the device designed specifically for use with the PURESPEED I/O logic block. There are eight Edge clocks on the left and right sides that span the entire edge of the device, while the Edge clocks at the top and bottom only span a given bank of I/O. The LatticeSC also provides 20 dedicated, phase matched clock dividers on chip. These clock dividers support a divide by 2 or by 4 and utilized to provide lower speed FPGA clocks for shift register and DDR gearbox logic for I/O interfaces.

Clock domain transfers

The LatticeSC PURESPEED I/O block also includes circuitry to ensure the proper transfer of the I/O data from the high-speed edge clocks to the lower speed FPGA fabric clocks. To accommodate this clock domain transfer, the SDR and DDR elements have two clock inputs, one for the edge clock and one to clock data on to the FPGA fabric clock. This approach guarantees error free transfers over process, voltage and temperature variations.

Memory interface support

The LatticeSC devices implement dedicated high-speed I/O circuitry to facilitate implementation of memory controllers supporting the various high-speed memory devices: DDR I/II SDRAM, QDR I/II SRAM and RLDRAM I/II. Next-generation memory controllers operate at HSTL (High-Speed Transceiver Logic) or SSTL (Stub-Series Transistor Logic) voltage levels. HSTL is the de facto I/O standard for high speed SRAM memory devices, while SSTL is the standard for high speed DDR SDRAM memories. The LatticeSC implements the industry's highest performance and lowest power programmable I/Os that include on-chip termination as well as DQS circuitry (shut off and edge detect) on every pin that simplifies the memory interface design and ensures robust operation.

Summary

This white paper has identified the continued importance of parallel I/O in contemporary systems and the demands put on FPGA architectures to deliver multi-gigabit I/O performance. These demands can be met when an FPGA has an I/O architecture that is designed as a true system level device, taking into account the entire suite of features necessary to deliver a true system level solution. These features must include best-in-class I/O buffers, termination technology that delivers superior linearity and signal integrity, I/O logic that deals seamlessly with dynamic clock and data alignment, and clocking and gearing resources that manage the processing and transfer of the high speed signals to the FPGA fabric. Above all, this white paper demonstrates that, to compete in the aggressive world of data path applications where only the strong survive, all FPGA I/O are not created equal.

###