THE FPGA AS A FLEXIBLE AND LOW-COST DIGITAL SOLUTION FOR WIRELESS BASE STATIONS

A Lattice Semiconductor White Paper

March 2007

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

Introduction

Within wireless base station system design, manufacturers continue to seek ways to add value and performance while increasing differentiation. Transmit/receive functionality has become an area of focus as designers attempt to address the need to move data from very high frequency sample rates to chip processing rates. Digital down/up converter (DDC/DUC) sub-systems are among the main digital components of the transmitter/receiver functionality within a base station. A significant portion of the silicon real estate is allocated for these sub-systems. Therefore, an efficient low-cost DDC/DUC implementation, which can save not only power and cost but also enhance performance, is highly desired. However, there are several significant implementation challenges on the way to achieving this goal.

There is a growing requirement for device reusability to accommodate flexibility in supporting several communications standards. The DDC/DUC design often is required to support different wireless standards and different spectrums of input sample frequencies. Support for these requirements means that a given DDC/DUC design must support different conversion rates. Unfortunately, ASIC-based designs cannot meet this requirement. Consequently, a growing number of designers are switching to FPGA-based designs to satisfy the needs of design reusability and flexibility.

Moreover, DDC/DUC design is typically multiplier-rich. DDC/DUC systems are constructed from Mixers, Numerical Controlled Oscillators (NCO) and FIR filters. All of these elements are multiplier-intensive. Furthermore, with the new 3G communications standards, a high number of channels are squeezed into the device containing the DDC/DUC system in order to achieve higher system cost-effectiveness. The combination of a higher number of channels, with accompanying single channel implementation complexity, and multiplier-rich functionality poses a formidable challenge to achieving low cost, especially to DSP processor-based designs.

Also, high speed analog to digital converters (ADC) with sampling rates higher than one GHz recently have been developed that enable moving higher switching frequencies from the RF domain to the digital domain. This means more processing is done in the

digital domain and higher DDC/DUC conversion rates. Higher input sample rates impose a new, but surmountable, challenge to the high-speed I/O interfaces of newer FPGA devices.

DDC/DUC systems that are based on DSP processors are limited by the DSP processors' computational power. As a result, large numbers of expensive DSP processors are needed to supply the high number of multipliers required. DDC/DUC systems that are ASIC-based don't have the flexibility that is required for different communication standards. Therefore, neither DSP processors nor ASICs are desirable solutions for meeting efficiency and low-cost requirements.

This white paper addresses new and practical techniques that improve FPGA high-speed I/O interface performance for next-generation wireless communications. New low-cost 90 nanometer FPGA device families from Lattice Semiconductor are well suited to meet all the challenges of DDC/DUC design. The practicality and competitiveness of such techniques, versus traditional ASIC and DSP processor-based design, will be discussed.

Hundreds or thousands of multipliers can be constructed from Lattice's sysDSP blocks, embedded block RAMs (EBRs) and programmable functional units (PFUs) to satisfy the DDC/DUC system demand for multiplier-intensive solutions. Flexible conversion rates also can be achieved easily by reprogramming the EBR look-up tables' (LUTs) content. Lattice's proprietary high-speed ADC interface technique enables extending the DDC/DUC input sample frequency higher into the RF domain frequency and increases the benefits of digital signal processing. Finally, Lattice's suite of DSP intellectual property (IP) cores offers user-configurability and silicon efficiency that enable fast time-to-market with DDC/DUC designs.

Device Reusability Requirements

Semiconductor devices targeting the wireless market have different design strategies for base stations and cell phones. Cell phones are high volume consumer products and therefore developing application specific standard product (ASSP) for each wireless standard is a viable development strategy. Base stations are not high volume

consumer products and therefore a different development strategy is needed. The base station market cannot justify ASSP NRE (non-recurring expense) cost. FPGAs have many features that are attractive for the base station market: there is no NRE cost, and the FPGA's flexibility enables the same FPGA device to support different wireless standards.

DDC and DUC architectures or implementations can be influenced significantly by the following parameters:

- Sample rate conversion ratio - The ratio between ADC or DAC sample rate and the system chip rate
- Number of base station antennas
- Number of base station channels

DDC/DUC system designers have a large variety of ADC/DAC to choose from. The ADC/DAC sample rate typically can be a couple of hundred Msps up to a couple of Gsps using the recent high-speed ADC or DAC devices.

The chip rate depends on the wireless standard that the base station supports. Table 1 provides some examples of common wireless standards chip rates.

Chip rate	***Wireless Standards***
270.833 Kbps	GSM, GPRS, EDGE
1 Mcps	Bluetooth 1.2
1.28 Mcps	TD-SCDMA
1.2288 Mcps	EIA-95 A/B, CDMA200, 1xEV-DO
2 Mcps	IEEE 802.15.4 ZigBee
3.84 Mcps	W-CDMA, HSDPA/HSUPA, TD-CDMA
11 Mcps	WLAN IEEE 802.16e

Table 1- Common Wireless Standards Chip Rates

FPGA flexibility enables the same device or device family to support a large variety of DAC/DUC boards that are designed for different standard base stations. The reusability

capacity of FPGA devices to support a huge variety of base station applications makes them an attractive design solution.

Implementation

Since DDC and DUC sections consume a significant portion of the digital part of a base station, it is important to have a cost effective DDC and DUC implementation. Cost effectiveness is achieved by time-sharing the same hardware for a maximum number of channels and by efficiently using different FPGA resources (DSP Blocks, Memories, and LUT fabric) to maximize the throughput and capacity of the FPGA. The good news for base station designers is that today low-cost, medium-size FPGA devices have the capacity to support tens of DDC and DUC channels.

As described in Figure 1, the DDC is composed of the following components: an I/Q splitter that is based on a numerical controlled oscillator (NCO) that modulates the input signal that comes from the RF section with sine and cosine waves, using two mixers and a decimation section that can be configured from 3 levels of FIR decimation filters, or a FIR decimation filter followed by a cascaded integrator comb (CIC) filter.

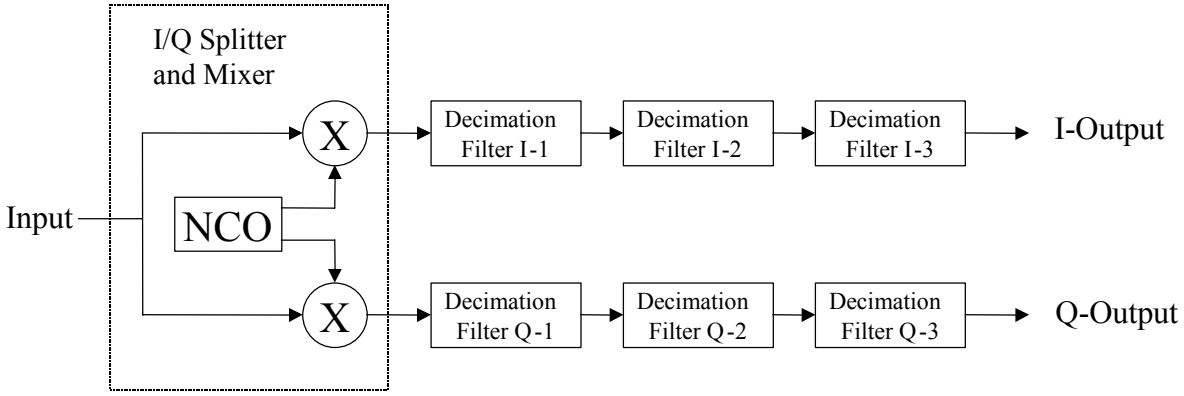


Figure 1 - DDC Structure

As described in Figure 2, the DUC is composed of the following components: 3 levels of FIR interpolation filters or a CIC filter followed by a FIR interpolation filter. An I/Q mixer based on an NCO and two mixers that demodulates the I and Q output signals before they are sent to the RF section.

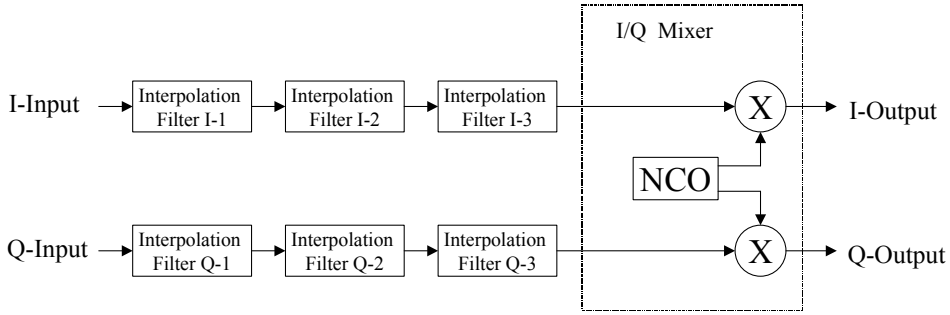


Figure 2 - DUC Structure

Implementation Considerations

- The DDC/DUC system is a multipliers-intensive system

A close look at the DDC and DUC components reveals their implementation can be multiplier-intensive: decimation and interpolation filters are typically implemented by an array of multipliers and adders, mixer is a multiplier, and an efficient method to implement NCO is based on phase shifting using complex multipliers.

- Overcoming the multipliers-intensive system challenge with splitting and cascading filters

A large FIR decimation filter or FIR interpolation filter with decimation/interpolation factor N can be broken down into two or three smaller and simpler cascading filters with N1, N2 and N3 decimation/interpolation factors. The decimation/interpolation factors satisfy the following equation: N = N1 * N2 * N3

Breaking down FIR decimation filters or FIR interpolation filters into two or three separate filters reduces the total number of taps required to implement the entire filter. A single filter with decimation or interpolation factor N would need a large number of taps (multipliers) to satisfy decent filter attenuation and noise characteristic requirements. Breaking down the filter into two or three smaller and simpler filters reduces the entire filtering system number of taps (multipliers). Additionally, the lower

sampling rate of the second and third cascading filters enables time multiplexing to reduce the implementation size even further.

- Symmetry decimation and interpolation filters

The symmetry feature of the DDC decimation filters and DUC interpolation filters can be used to achieve additional size reduction. In the case of symmetry, the n taps FIR filter coefficients h(0), h(1), …,h(n) satisfy the condition of h(k) = h(n-k) {0 =< k =< n}.

Since h(k) = h(n-k) one multiplication of h(k) with the sum of the two, correspondent samples can be done and therefore the number of multipliers required can be reduced by a factor of approximately 2.

- Efficient use of FPGA resources as multipliers

Efficient use of FPGA resources is extremely important for multiplier-intensive applications such as DDC or DUC. SysDSP Block multipliers are obvious candidates for use in this case. The use of memories and LUT fabric resources as multipliers is not so obvious. However, it can increase implementation efficiency significantly. Embedded Block RAM (EBR) and the fabric's distributed memories can be used as FIR filter multipliers using the technique of distributed memories, also known as the soft multipliers technique. Using this technique, the number of multipliers in Lattice FPGA devices typically can be increased by a factor of 2 to 5.

Figure 3 demonstrates how EBR can be used as a FIR filter by implementing a distributed arithmetic technique. The samples are serially shifted into the EBR address bus. Inside the EBR there is a table of pre-computed result multiplications and summation of each input sample bit (address bit) with its appropriate coefficient. The accumulator accumulates the n (n is sample bit resolution) intermediate results and provides a complete FIR filter result after n clock cycles.

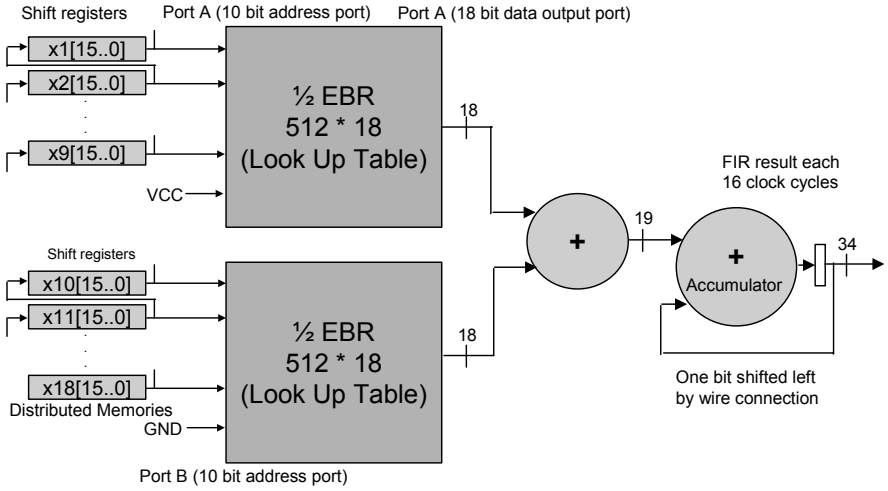


Figure 3 - Use of Lattice Block Memories as FIR Multipliers

- CIC filter

Replacement of some interpolation/decimation FIR filter chain portion with Cascaded Integrator-Comb (CIC) multipliers is another method to reduce the amount of multipliers needed for implementation. CIC multipliers have no multipliers. They are based on adders and subtractors. Digital Up/Down converters usually require a large rate change on the order of a few hundreds. High rate changing interpolation or decimation filters tend to be very expensive in hardware. CIC filters, also called Hogenauer filters, can serve as inexpensive high-factor decimation or interpolation filters [1]. They are used to achieve arbitrary and large rate changes in digital systems and can be implemented efficiently by using only adders and subtractors. With FPGAs having fast carry chains for implementing adders, a CIC filter is very amenable for FPGA implementation.

A CIC filter is built using two basic blocks: an integrator and a comb. An integrator is a single pole IIR filter which essentially has a low pass filter characteristic. A comb is a FIR filter defined by the following equation:

$$x[n] - x[n - RM]$$

Here M is the differential delay and R is the rate change parameter. A comb has the frequency characteristics of a high-pass filter.

The structure and characteristics of integrator and comb are listed in Table 2.

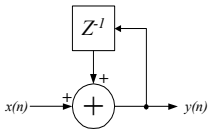
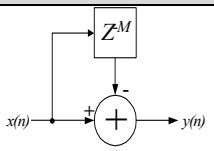
Items	Integrator	Comb
Structure		
Sampling Rate	f_s	f_s / R
$y[n]$	$y[n-1]+x[n]$	$x[n]-x[n-RM]$
$H_I(z)$	$\frac{1}{1-z^{-1}}$	$1-z^{-RM}$
$\left\|H\left(e^{jw}\right)\right\|^2$	$\frac{1}{2(1-\cos\omega)}$	)

Table 2 - Structure and Characteristics of Comb and Integrator

A typical CIC filter is built as a cascade of multiple integrator and comb sections and a rate changer between the two sections. The arrangement of the comb, integrator and rate conversion sections depends on whether a decimator or interpolator is realized. Typical CIC decimator and CIC interpolator are given in Figure 4 and Figure 5.

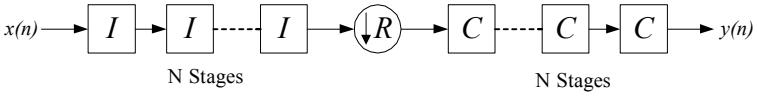


Figure 4 - CIC Decimator

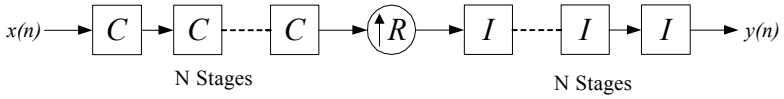


Figure 5 - CIC Interpolator

The transfer function of an N stage CIC filter is given by the following equation [2]:

$$H(z) = H_I^N(z)H_C^N(z) = \frac{(1-z^{-RM})^N}{(1-z^{-1})^N} = \left(\sum_{k=0}^{RM-1} z^{-k}\right)^N$$

The magnitude response of the filter is given by:

$$|H(f)| = \left|\frac{\sin \pi M f}{\sin \frac{\pi f}{R}}\right|^N$$

Using the approximation sin(x) = x for small x, the above can be approximated as:

$$|H(f)| \approx \left|RM\frac{\sin \pi M f}{\pi M f}\right|^N \text{ for } 0 \le f < \frac{1}{M}$$

The frequency response for a CIC filter is that of a low-pass filter. The power response for a 4 stage with M=1 and R=7 is shown in Figure 6.

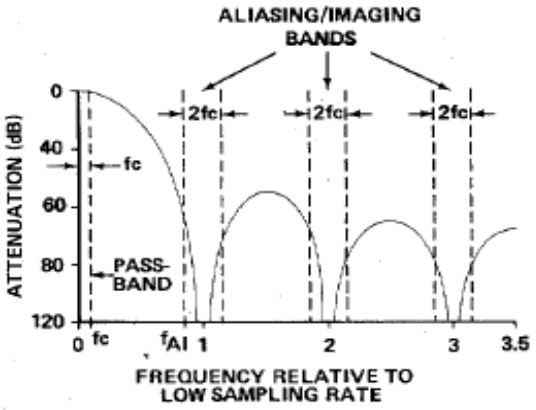


Figure 6 - Frequency Response for a CIC Filter

Implementing DDC/DUC Using Lattice ispLeverCORE IPs

It is fairly simple to implement a DDC or DUC in Lattice FPGAs, thanks to the availability of the constituent components as ispLeverCORE IPs. Two applications that use CIC

filters as interpolators or decimators in digital rate conversion are shown in the following figures. Figure 7 shows the use of a CIC interpolator for up-conversion for digital radio applications (Figure extracted from Texas Instrument's GC 4116 Quad Multi-Standard Digital Up-converter data sheet).

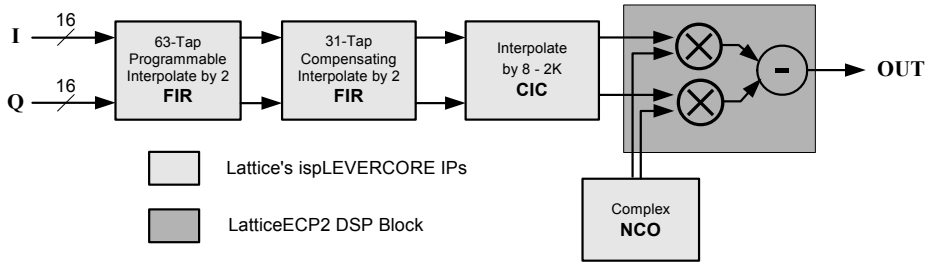


Figure 7 - Digital Up-Converter for Digital Radio Application

As shown in Figure 8, the digital up-converter uses the following IP core configurations:

1. FIR Filter (63-tap, interpolating filter)
2. FIR Filter (31-tap, interpolating filter)
3. CIC Filter (Interpolating CIC filter with rates programmable between 8 and 2K)
4. NCO (NCO with sine and cosine outputs)

Figure 8 shows a digital down-converter for a WCDMA application that uses a CIC decimator.

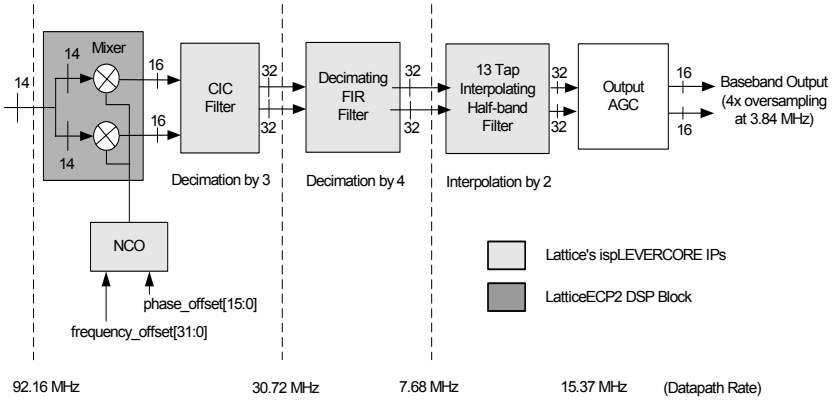


Figure 8 - Digital Down-Converter for WCDMA Application

As shown in Figure 8, the digital down-converter uses the following IP core configurations:

1. CIC Filter (Decimating CIC filter with a rate of 3)
2. FIR Filter (Decimate by 4 filter)
3. FIR Filter (13-tap, interpolating filter)
4. NCO (NCO with sine and cosine outputs and phase and frequency offset inputs)

ADC/DAC Device Trends and Their Challenges to Digital Logic Design

The trend in analog-to-digital converters is to support higher sample rates and wider bus widths. The higher sample rates are particularly ideal for applications such as base stations for wireless protocols (WiMAX being one example). Other useful applications include satellite set-top boxes and test instrumentation, including digital oscilloscopes.

In such wireless applications, the transceiver architecture traditionally has been divided into an RF and baseband portion. Baseband processing receives chip-rate data from the RF portion, as defined by an in-phase channel and quadrature phase channel (I[] and Q[] respectively). The RF-to-baseband interface was accomplished with low-speed A/D converters, in the range of a few million samples/second (msps).

In the drive to reduce component and system cost, the trend is to increase the functionality of cheap baseband logic (implemented with low cost digital logic ICs) and reduce the functionality of RF (implemented with expensive RF & mixed signal devices). This is done by increasing the sample rates of the ADC and DAC, thus moving "baseband" closer to the transceiver antenna. This is shown in Figure 9.

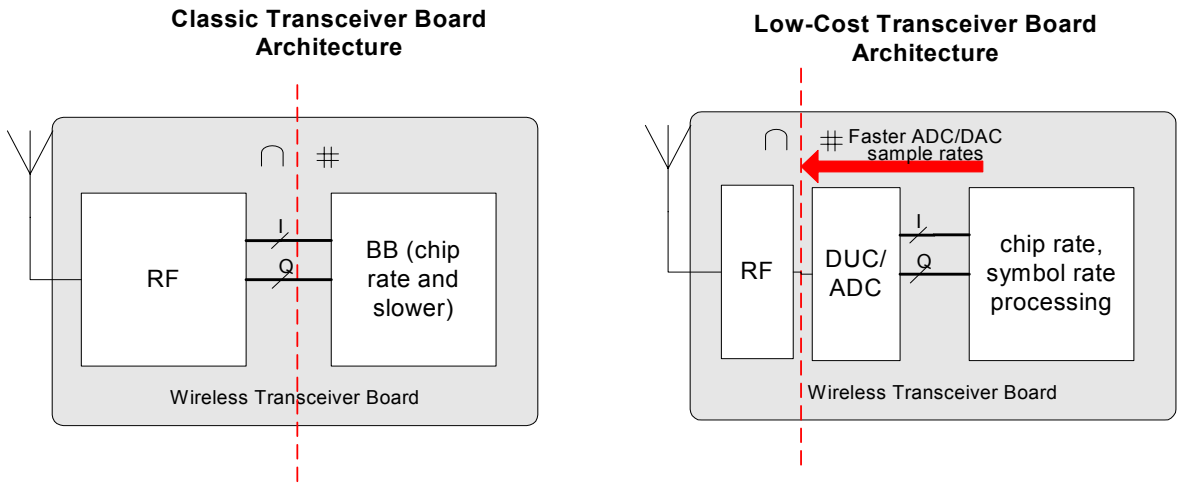


Figure 9- Wireless Transceiver Board Architectures

Type	Analog Sample Rate	Sample Clock (rising & falling edge)	Resolution	LVDS Bus Width (pairs)	LVDS data rate
ADC (N)	3 Gsps	1.5 GHz	8-bit	32	750 Mbps
ADC (N)	2.5 Gsps	1.25 GHz	8-bit	32	625 Mbps
DAC (F)	1 Gsps	500 MHz	14-bit	14	1 Gbps

Table 3 - Examples of Potential ADC and DAC Device Characteristics[1]

Therefore, the challenge for DUC and DDC logic is to convert this ADC/DAC digital sample rate down to chip rate processing rates. The architectural implementation of these blocks must support real-time data transfer (via I/Os) and processing (via DSP blocks) to meet both current ADC/DAC sample rates as well as future sample rates. As seen in Table 3, a solution must be found that supports various transfer rates using LVDS signaling, up to and including 1 gigabit per second data.

[1] As of 3Q 2006. The trend going forward is for even higher sample rates of analog data

Lattice FPGA Solutions vs. Alternatives for DUC/DDC

Lattice provides an optimal solution to meet the requirements of implementing high-speed DUC and DDC. The following FPGA features are available in Lattice FPGA families, and can be specifically configured to address DUC/DDC functionality. Further details are available in the datasheets of Lattice FPGA families.

I/O Specific Features

- LVDS differential signaling available on programmable I/Os, compatible with the digital interface of ADC/DACs.
- An IOLOGIC register gearbox to convert high-speed data at the FPGA periphery to slower and wider data in the core of the FPGA, and vice versa. Dedicated hardwired logic supports 1:4 gearing on the inputs, and 4:1 gearing on the outputs.
- Abundant resources to shift and distribute a forwarded clock, which will center the clock relative to incoming LVDS data at the input registers. A DLL-controlled delay block is optimal for shifting, as the shift is tightly controlled across voltage and temperature conditions.
- Abundant resources to generate and distribute a forwarding clock for the output interface (to DAC). PLL and edge clock resources are ideal for generating clock and data outputs with very tight skew relative to each other.

DSP Block Features

- High performance DSP blocks, with dedicated multipliers, adders and pipeline registers in an ASIC-like block.
- This high performance is especially useful for real-time DUC/DDC filtering requirements, given current high-speed ADC/DAC converters.
- DSP blocks address the future trend of steadily increasing ADC rates used to increase digital logic in a wireless transceiver and reduce system cost.

Advantages Over ASIC

- Flexibility - FPGAs have a high degree of programmability and flexibility that is very useful for DUC / DDC implementation. It is desirable to be able to adjust filter coefficients to optimize their function. Also, the interpolation and decimation factors may be adjusted to trade off between high operating frequency and high accuracy.
- Economical - Given the quantities of wireless communications equipment developed by vendors, it is far more economical to implement in FPGAs rather than ASICs. Although unit costs of FPGAs are higher than ASICs, leading edge ASICs have very high NRE costs (i.e. $1.5 million for 65 nm), and can be justified only for quantities of several hundred thousand to millions of units.
- Immune to changes in standards - protocols or standards that are in the process of definition still can be implemented in FPGAs without concern. For example, if the chip rate of a standard changes, this can be addressed in an FPGA but not in an ASIC. There are real world examples of chip rate adjustment in the harmonization of CDMA-2000 and WCDMA standards.

Advantages Over DSP Processors

- Performance- DSP processors are ideal for implementing certain symbol rate processing found in the baseband functionality. They also are ideal for implementing vocoders (voice coders/decoders) in wireless handsets. However, DUCs and DDCs have fast real-time processing requirements that cannot be met using DSP processors.
- High degree of parallelism - The functions performed for filtering are very repetitive (multiply-add, multiply accumulate), which strongly suggests that the hardwired logic found in FPGAs is more efficient than a sequence of instructions from a DSP. At every clock cycle, FPGA logic can perform the same operation on new data and produce the updated results. DSPs are more useful for performing a sequence of instructions, and not an identical function that is repeated frequently with various data.

Advantage over Xilinx and Altera

- High performance and low-cost DSP architecture - Lattice is the only programmable logic company that provides a high-performance, full-featured DSP block in a low-cost FPGA device family. Competitive low-end families support a hardwired, simple 18x18 multiplier. This solution will encounter performance degradation when adder/accumulator logic is constructed in the lookup table fabric. The Lattice DSP block architecture provides support for current and future DDC/DUC implementations with a consistent, deterministic level of performance.

References

1. Hogenauer, “An Economical Class of Digital Filters for Decimation and Interpolation"

An Economical Class of Digital Filter for Decimation and Interpolation, Eugene B. Hogenauer, IEEE. Trans. ASSP, Vol. 29, No. 2, pp.155-162, April 1981.

2. Ray Andraka, “High performance Digital Down-Converter for FPGA,” Xilinx Xcell Journal Issue Number 38, 4Q 2000

3. Hunt Engineering, “Digital Down-Converter using FPGA,” http://www.hinteng.co.uk/

4. Matthew P. Donadio, “CIC Filter Introduction,” Iowegian, www.users.snip.net/~donadio/cic.pdf.

5. Asher Hazanchuk and Sheac Yee Lim, “Optimizing Up/Down Conversion with FPGA Techniques” CommsDesign Magazine, Dec 23, 2003

6. T. Hollis, “Digital Down-Conversion at the heart of today’s communications systems,” Global DSP Magazine, Vol. 2 Issue 10, October 2003

7. Asher Hazanchuk, “Soft Multipliers for DSP Applications” GSPx Conference, April 2003