LVDS Outputs on the ADS527x

Anand Udupa *High-Speed Data Products*

ABSTRACT

The ADS527x are a family of high-performance analog-to-digital converters (ADCs) that feature serialized low-voltage differential signaling (LVDS) outputs. Data in each channel are serialized and sent out on a pair of pins in LVDS format. In addition to reducing the pincount and package size of the multi-channel ADC, serialization also eases the routing of ADC outputs of the multiple channels to the receiver. The LVDS architecture offers multiple advantages, such as reduced effects of digital noise coupling to the internal analog circuit of the device. The serializer that provides data to the LVDS buffer also generates a 1x clock and a 6x clock, which are used for frame and bit identification, respectively. This application note describes the implementation of LVDS timings inside the ADS527x. The generation of the data and clock outputs from an internal 12x clock are detailed. It also discusses a method of specifying the LVDS timings from the standpoint of the receiver. Unless otherwise noted, this report refers to the ADS5270, ADS5271, ADS5272 and ADS5273 as the *ADS527x*.

LVDS Implementation

The ADS5270 uses an integrated internal PLL that generates a 12x sampling clock. The edges of this 12x clock are used to serialize the ADC data bits. The 12x sampling clock also generates the LVDS bit clock and the LVDS frame clock, which are output synchronously with the serialized data. The edges of the 12x clock are used as shown in Figure 1.

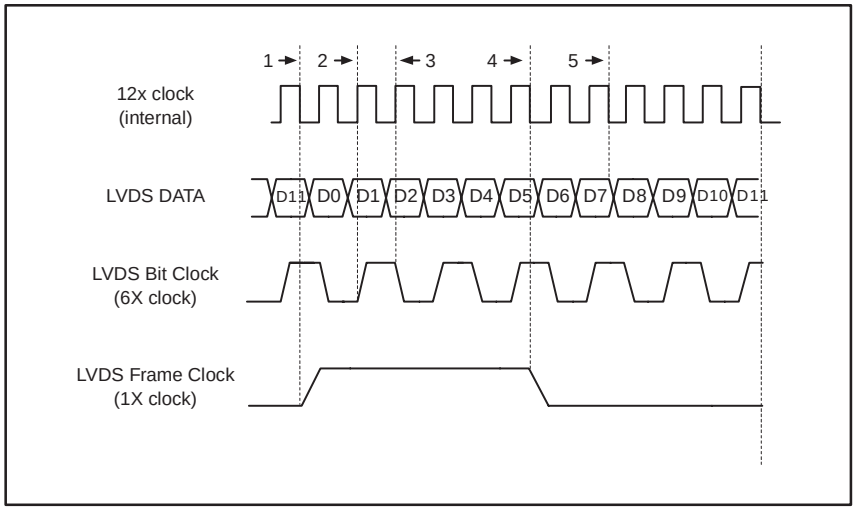

Figure 1. Generation of LVDS Data and Clocks from the Edges of the Internal 12x Sample Clock

Both the rising and the falling edges of the internal 12x clock are used for generating the LVDS bit clock (6x clock) and the LVDS frame clock (1x clock). The rising edges of the 12x clock are used to generate both the rising as well as falling edges of LVDS bit clock (arrows 2 and 3, respectively). The falling edges of the 12X clock are used to generate the switching instances of the data (arrow 5), as well as the rising and falling edges of the LVDS frame clock (arrows 1 and 4, respectively). The logic delays used in generating the transitions of all the data and clocks are closely matched. As a result of the matching design, the transition of the LVDS bit clock edge is expected to occur very close to the middle of the open eye of the data.

Setup and Hold Times

The setup and hold times are defined with respect to the transitions between the data and the LVDS bit clock (6x clock). Since the data is to be captured using both edges of the LVDS bit clock, one setup and hold time period can be defined with respect to the rising edge of the LVDS bit clock while another period can be defined with respect to the falling edge. There could be slight differences between the timings related to the two edges because of the slight differences in the rise and fall times during the generation of the logic for the LVDS bit clock. Such a difference would be on the order of 50ps. To simplify the timing specification, a single set of setup and hold times are indicated in the product data sheet, reflecting the lower value of the setup and hold times for either the rising or falling edge.

Figure 2 illustrates the method of measuring the setup and hold times from the differential waveforms of the LVDS bit clock and data.

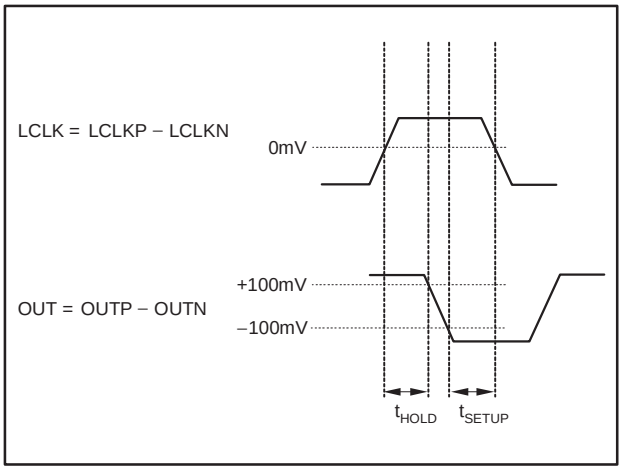

Figure 2. Setup and Hold Times

The rise and fall instances of the LVDS data are referenced to ±100mV, since these are the typical thresholds required for the receiver to identify them as logic levels. The rising time of the LVDS data waveform is measured as the time taken to swing from –100mV to +100 mV (and vice-versa for the falling time).

Jitter

A typical eye pattern of the LVDS waveforms is shown in Figure 3. The waveforms in Figure 3 are plotted using a high bandwidth oscilloscope in infinite persistence mode. The time for which the eye is *open* is a measure of the available setup and hold times available to the receiver. Typically, the LCLK rising and falling edges occur in the middle of the eye pattern of the data, thereby producing roughly equal setup and hold times.

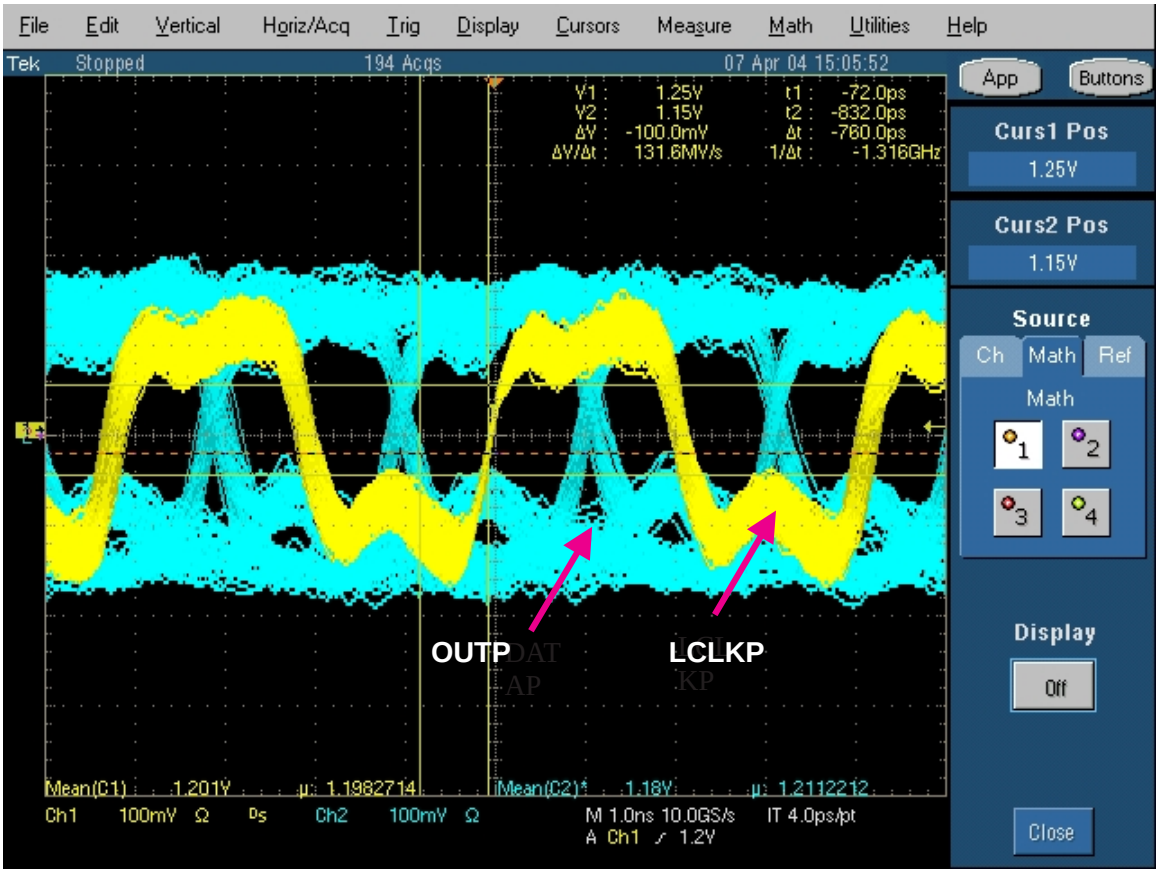

Figure 3. Eye Pattern of the LVDS Clock and Data

The jitter effectively reduces the timing margins available to the receiver and can result in bit errors. The timing numbers for the setup and hold times as mentioned in the datasheet are representative of the actual timing margins available to the receiver, and take into account the effect of jitter. The setup and hold time values can be improved by increasing the LVDS current setting to 4.5mA or 6mA. This current setting increase produces faster rise and fall times, thus increasing the open time of the eye pattern.

References

ADS5270 Datasheet (SBAS293D)

ADS5271 Datasheet (SBAS313)

ADS5272 Datasheet (SBAS324)

ADS5273 Datasheet (SBAS305B)

To obtain a copy of the referenced documents, visit the Texas Instruments web site at www.ti.com. *x* indicates the current revision letter for each document.

The Basestation Conference
Your window onto a $50 billion industry - May 8th-12th 2006 e-newsletter

New Serial Interface Standard for High Speed Data Converters Set to Reduce Design Complexity & Cost, Enhance Performance

By Matthias Feulner, Texas Instruments
Business Development Manager Wireless Infrastructure

Why Go Serial?

Serializing output data of high-speed analog-to-digital converters is a relatively new concept, even though first generation devices have been around for a couple of years now. The demand for changing from parallel to serial output format was primarily driven by high-density applications such as medical imaging, for example in ultra-sound scanners (see Figure 1), with channel counts reaching up to 256 channels per equipment as well as wireless radio base station receiver architectures in diversity or smart-antenna configurations that employ IQ sampling, requiring two AD converters per receiver chain (see Figure 2).

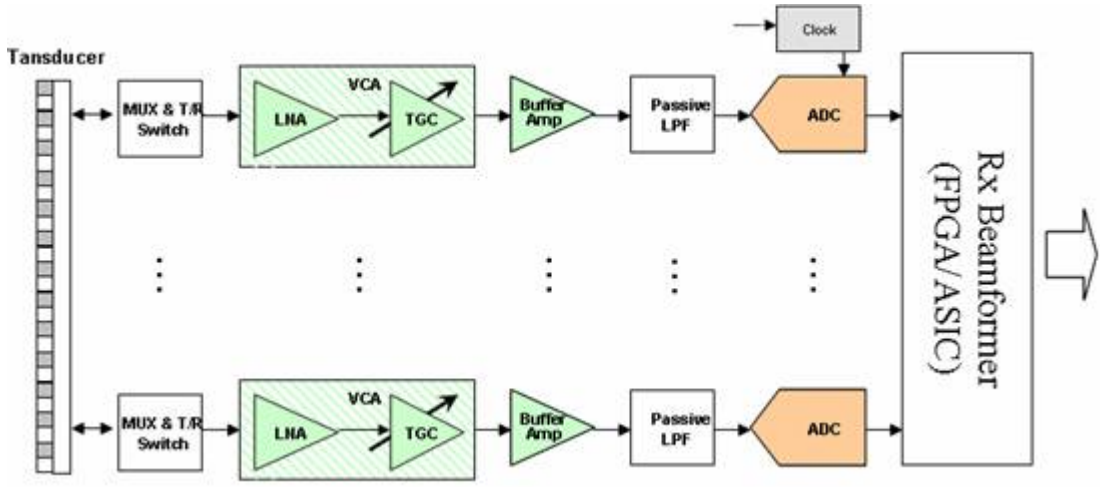

Figure 1: Medical ultrasound scanner receiver signal chain

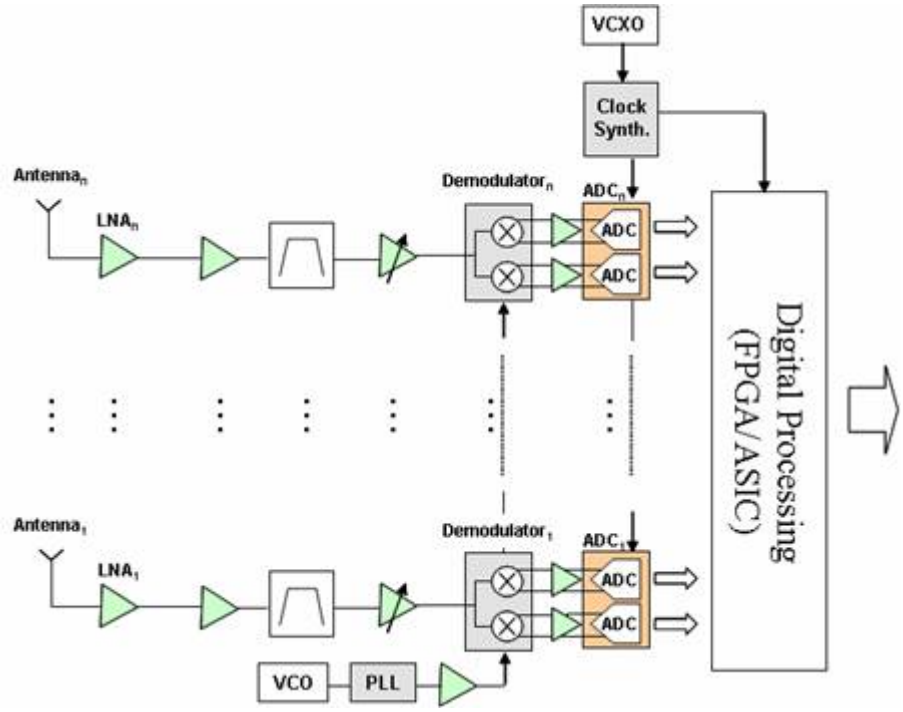

Figure 2: Wireless base station Rx with I/Q sampling in diversity (n=2) / smart antenna configuration

Today's Serial Interface Implementations

Most of today's commercially available flavors of serial output (high-speed) AD converters are based on the so-called 'clock-data-frame' interface, illustrated in Figure 3 with the example of an octal 12bit AD converter, that outputs on individual lines:

- serialized digital sampling data
- bit clock, which is 6x the ADC sampling clock frequency, to clock the serial interface in DDR mode both on the falling and rising edge
- frame synchronization clock, which is identical with the ADC sampling clock frequency to signal the data word boundary for the receiving device

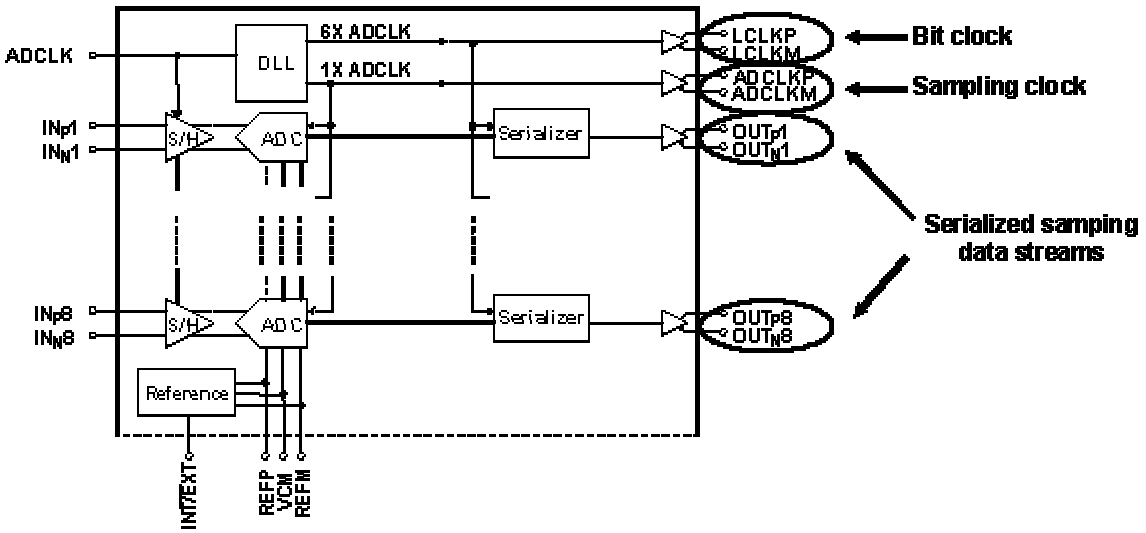

Figure 3: Serial AD converter with clock-data-frame output interface [1]

The electrical interface is based on low-voltage differential signaling (LVDS), a low-swing differential interface that allows for low-power driver implementation, limits electro-magnetic emissions and is highly interference tolerant.

Limitations of this interface are on maximum data throughput, which with LVDS is commonly limited to about 1 Gigabits per second (Gbps), reach (limited by skew between clock and data lines) and lack of compatibility with standardized serial interfaces.

JEDEC Interface Standardization and its Advantages

Overview

Now a new standard for a serial interface connecting (high-speed) data converters with logic devices such as ASICs and FPGAs has recently passed voting by JEDEC task group JC-16, supported by a large group of both users and manufacturers of data converters and logic devices. Publication of the official standard will follow shortly. The summary of requirements addressed by this standard is as follows:

- Cost-effectiveness, requiring underlying technology to be widely available to both data converter and ASIC / FPGA vendors with no IP hurdles.
- Serial interface format in order to minimize traces and pin counts.
- Minimum coding overhead and associated digital logic on the converter to limit added power and noise coupling which are directly affecting the converter's performance.

- Generally support for data converters with resolutions from 8 to 18 bits which results in gross data rate of 0.3125 – 3.125 Gbps (higher rates can be achieved by multiplexing data across multiple lanes in a future extension of the standard).
- Optimized for chip-to-chip interconnect, i.e. no cabling or connectors envisioned in the medium (even though a possible future extension).
- Up to 8" (20 cm) of trace length on FR-4 PCBs or up to 6 dB of loss.
- Differential and low-swing electrical interface for minimum electromagnetic emissions and high interference tolerance.
- Common interface regardless of data converter resolution.

The generic system view assumed in the standardization considers a very wide variety of possible configurations:

- One converter connected across a single serial link.
- Multiple converters (in a common package) connected across a single serial link.
- One converter connected across multiple serial links. (not described in the current version of the standard, but an option for future extension).
-

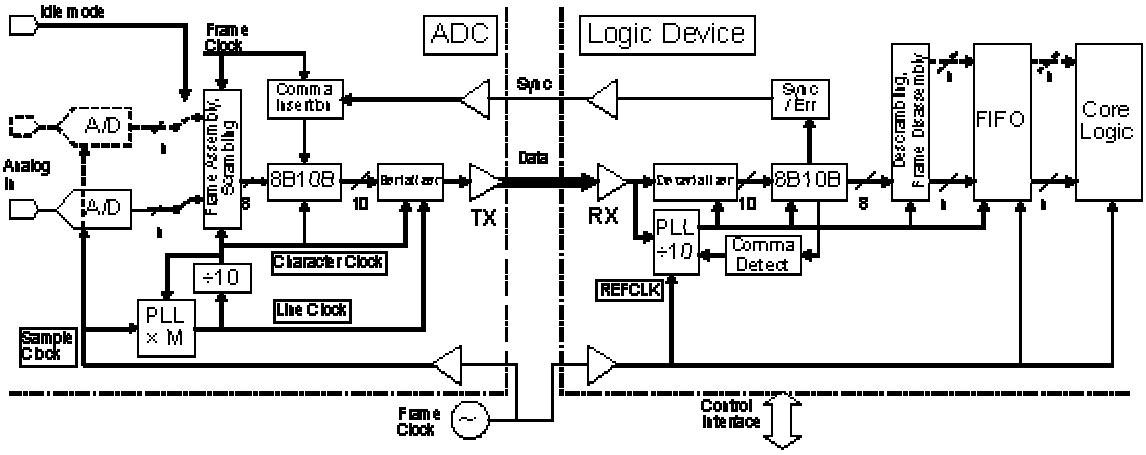

Figure 4: AD converter(s) connected to logic device using JEDEC interface [2, simplified]

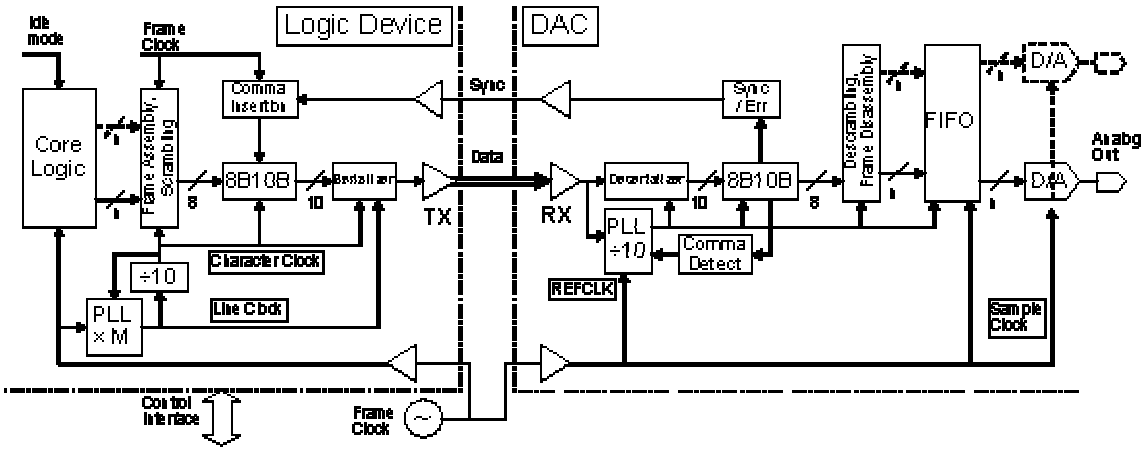

Figure 5: Logic device connected to DA converter(s) using JEDEC interface [2, simplified]

Electrical Interface

The solution that was decided to best meet conditions stated above is an electrical interface similar to the OIF's TFI-5 and SXI-5 specifications, what is commonly referred to as Current Mode Logic (CML). The major advantage of this choice is that numerous other applications have used that interface in the past and thus it is widely supported in the industry, particularly with suppliers of FPGAs and ASICs which readily include compliant interfaces with their standard I/O libraries. On top CML has proven to support data rates of up to 3.125Gbps (as required by the standard) and beyond, drive lines of up to 40 inches and allows for scalable voltage swing to save power in short-reach applications.

Data Stream & Formatting

The transport layer protocol is based on 8B10B coding, again a technique that is widely used in serial interface devices with only low overhead adding to the complexity of the device. This ensures both a DC-free line code, i.e. allows for AC coupling with the serial link, and encodes the line clock information with the data stream thus not requiring an additional clock line between transmitter and receiver.

Since an 8B10B coding based transport layer encodes data words of 8bit each, any converter output data word aside from 8 or 16 bits does not match up with the word length of the serial link interface. Several measures are taken to ensure that bandwidth efficiency is maximized and crosstalk between serial interface logic and the analog converter circuitry is minimized. First, converter data word packing is applied, i.e. if data converter resolution is other than 8 or 16 bits, data words from multiple converters in the same package may be concatenated as shown in Figure 6, thus avoiding wasting of excess bits.

Second, to circumvent the cross-talk issue, the bit and word clocks of the SerDes interface should be an integer multiple of the converter's sampling clock. With word clock to bit clock ratio of 10 defined by the 8B10B coding scheme, one still needs to ensure that word clock and sample clocks are linked by an integer ratio. This is resolved by adding padding bits to fill up the last data word of a data frame transmitted.

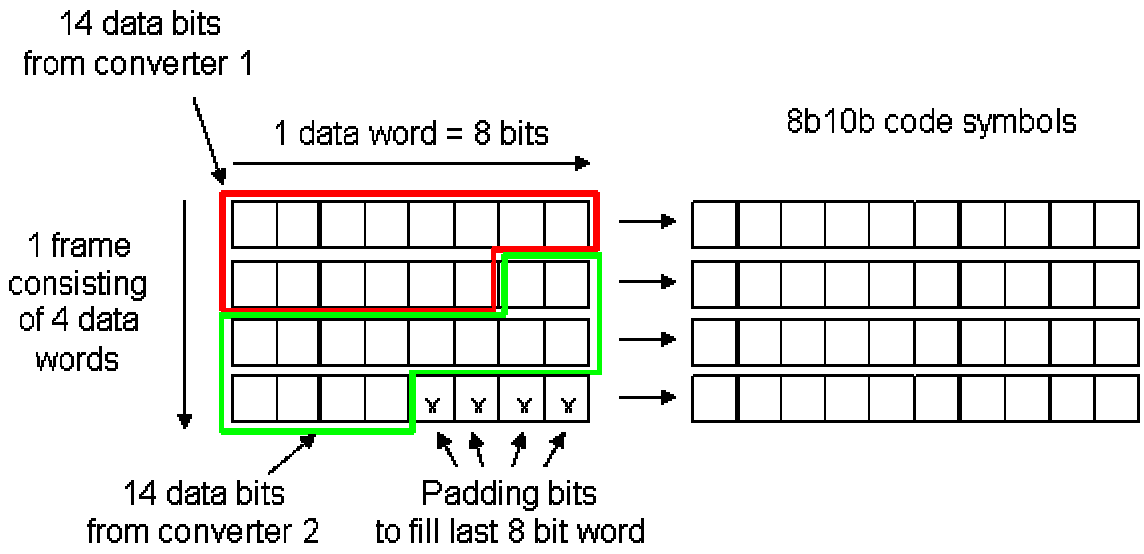

Figure 6: Word packing and bit padding to fit data from two 14bit converters to 8bit data words [3]

Eventually, to ensure that the link is 'always on', dealing with systems where there isn't a continuous data stream and avoiding issues with re-lock time of the receiver PLL upon link re-start as well as keeping frame format if only one out of multiple converters feeding a data frame is turned off, an idle sequence needs to be sent for the inactive converter(s).

As a means of establishing link synchronization during start-up and maintaining frame synchronization during operation a sync signal is being looped back from the receiving device to the sending that triggers certain start-up and link maintenance procedures. More detail can be found in [3].

Resulting Benefits

The benefits resulting from a JEDEC-based serial interface implementation on data converters are manifold:

- Reduction in board space needed for both data converter package and routing of I/O lines on the board, thus easier layout and potentially a reduction on the number of board layers required

- Package cost reduction by eliminating I/Os: A particularly heavy-weighing advantage in wireless communications where high-resolution data converters are being employed and consequently packages of both data converters and logic devices tend to be I/O bound rather than chip-size bound, i.e. the size of the package, and thereby its cost, is dominated by the I/O count rather than the chip size.
- Longer distances and / or higher through-put can be achieved both compared to parallel interfaces as well as the clock-date-frame serial interface implementation which struggle with skew between data lines themselves and data lines and clock line respectively.
- High degree of interoperability and thus easy interfacing with commercially available logic devices is enabled by relying on 8B10B coding based transport layer with electrical specifications complying with TFI5/SXI5-like CML-based physical layer.
- Generic interface, i.e. neither dependent on data converter resolution or sampling speed nor on number of converters integrated in a single package.

With the official standard in the publication process at this time, first data converters based on the new interface can be expected to both reduce design complexity and cost and enhance system performance in the near future.

References:

[1] TI ADS5270 Data Sheet
(http://focus.ti.com/docs/prod/folders/print/ads5270.html)
[2] JEDEC Data Converter Serial Interface Specification: Overview
[3] JEDEC Data Converter Serial Interface Specification: Data Stream
[4] OIF Implementation Agreements on TFI5/SXI5 Interfaces at
www.oiforum.com/public/impagreements.html