Nuclear Instruments and Methods in Physics Research A 626-627 (2011) S169–S172

Contents lists available at ScienceDirect

Nuclear Instruments and Methods in Physics Research A

journal homepage: www.elsevier.com/locate/nima

Measuring propagation delay over a coded serial communication channel using FPGAs

P.P.M. Jansweijer *, H.Z. Peek

Nikhef, Science Park 105, 1098 XG Amsterdam, Netherlands

ARTICLE INFO

Available online 5 May 2010

Keywords:
Timing
Network timing
Timing calibration
Propagation delay
FPGA
8B10B
8B/10B
Ethernet
IEEE 1588
Neutrino telescope

ABSTRACT

Measurement and control applications are increasingly using distributed system technologies. In such applications, which may be spread over large distances, it is often necessary to synchronize system timing and know with great precision the time offsets between parts of the system. Measuring the propagation delay over a coded serial communication channel using serializer/deserializer (SerDes) functionality in FPGAs is described. The propagation delay between transmitter and receiver is measured with a resolution of a single unit interval (i.e. a serial link running at 3.125 Gbps provides a 320 ps resolution). The technique has been demonstrated to work over 100 km fibre to verify the feasibility for application in the future KM3NeT telescope.

1. Introduction

Measurement and control applications are increasingly using distributed system technologies. In such applications, which may be spread over large distances, it is often necessary to synchronize system timing and know with great precision the time offsets between parts of the system.

An example of an application is a very large volume neutrino telescope to study astrophysical sources. Such a telescope consists of a large three-dimensional array of optical modules that contain light-sensitive photo multiplier tubes (PMTs). The photon detection time, with a subnanosecond resolution, is sent via a serial communication channel to a place where data is accumulated and further processed into a format suitable for physics analysis.

For a serial communication channel the obvious time resolution is as small as a single symbol, which is defined as one bit-time (unit interval). Often the time resolution is degraded, because available hardware supports only the boundary of a code-group or even a couple of code-groups defining an ordered set.

The technique described here can be efficiently implemented in FPGAs. It operates on a serial point to point communication channel and has a resolution of a single unit interval. For example, a 3.125 Gbps serial link provides a 320 ps resolution.

2. Communication channel

2.1. Serial communication channel coding

The industry widely adopted 8B/10B coding [1,2] as a standard for serial data transport. This coding scheme solves three important properties that are a concern for any serial connection.

1. Bit synchronization: It combines "clock" and "data" in one single serial stream by limiting the maximum run length. This results in enough edges in the serial stream, such that the bitclock can be recovered at the receiver.
2. DC-balance: To minimize transmission errors, DC-balance is achieved by mapping 8 bit characters into 10 bit code-groups. There are not enough DC balanced code-groups to map each 8 bit character directly onto one 10 bit code-group; therefore many of the 8 bit characters are mapped onto two 10 bit code-groups: one positive with more ones and the other negative with more zeros. Running disparity balances the number of positive and negative code-groups.
3. Special code-groups: For word synchronization and out-of-band information there are a number of special 10 bit code-groups available that are not used for in-band data transmission.

2.2. Transfer of timing information

For timing information transfer, properties 1 and 3 are of special interest. The recovered bit clock is used as the local clock at the

* Corresponding author. Tel.: +31 20 5922039.
E-mail address: peterj@nikhef.nl (P.P.M. Jansweijer).

0168-9002/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.nima.2010.04.126

receiving side (RX) and locked (property 1) to the transmitter clock (TX), ensuring a high accuracy. The two locked clock domains are isochronous, which means that there is an uncalibrated timing offset between them. This offset can be measured by sending a timing marker signal forth and back. A special code-group (property 3) can be used as a timing marker signal.

2.3. Serial communication synchronisation

The serial communication is based on transferring words (built out of 1, 2 or 4 code-groups) at the system clock speed. The transmitter serialization architecture contains a PLL that multiplies the system clock to the serial bit rate. Initially the receiver uses a local free running clock, which is locked onto the transmitter system clock after synchronisation. To completely understand the lock mechanism of a coded serial communication channel one needs to understand the steps that take place during synchronisation. First, a local free running clock at the receiver is used to set the centre frequency of the PLL in the clock data recovery circuit at the nominal bit rate of the serial data stream. Second, when a stable serial bit stream is received, the phase detector input of the PLL in the clock data recovery circuit switches from the local free running clock to the edges of the received bit stream. After the PLL acquired lock to the received bit stream, it is locked at a random bit position.

The next step is to accomplish word synchronization. During synchronization the transmitter sends special code-groups from which the receiver can recognize the word boundaries. The de-serialization architecture in the receiver aligns the data in the serial to parallel register. This process is called bit-slip, which is achieved by a barrel shifter. Word synchronization is reached when the synchronization code-groups are recognized and have the proper position within the word.

The phase relation between the transmitter system clock, the communication channel and the receiver recovered clock is determined by the number of barrel shifts needed to properly align the receiver.

3. Verification of the propagation delay measurement principle

3.1. Test setup description

In order to demonstrate the propagation delay measurement principle, a test setup was built that is able to measure the propagation delay of a serial communication channel. The data-link layer (OSI-model [3]) of the communication channel uses 8B/10B coding and transfers 20 bit words (two 10 bit code-groups; Fig. 1) at a speed of 3.125 Gbps; therefore the system speed is 156.25 MHz (6.4 ns period).

A "Start" pulse is generated when the transmitter sends a marker signal, which initiates a time-interval measurement. A "Stop" pulse is generated when the receiver has decoded the marker signal. In order to simplify the test setup an oscilloscope is used to measure the time between "Start" and "Stop" pulse.

The test setup (Fig. 2) consists of two separate FPGA evaluation boards connected by a test-bed to create a 100 km long communication channel.

The transmitter is implemented in a Lattice LFSCM25 FPGA [4,5] and the receiver is implemented in a Xilinx Virtex-5 FPGA [6,7].

Figure 3 shows a picture of the real test setup where the various sub parts are labelled for reference. Note the two spools that contain 50 km of fibre each.

The "Start"/"Stop" delay measurement has a resolution of 6.4 ns. The "Stop" signal at the receiver is in the receiver recovered clock domain. The link propagation delay can be computed by the number of barrel shifts multiplied by the unit interval (320 ps) added to the measured "Start"/"Stop" delay. Thus the link propagation delay has a resolution of one unit interval.

Figure 4 shows the re-synchronization behaviour of three "Stop" signal measurements. The oscilloscope is triggered on the "Start" signal. The picture shows a measured propagation delay of 491.547.440 ps at the rightmost trace where the number of barrel

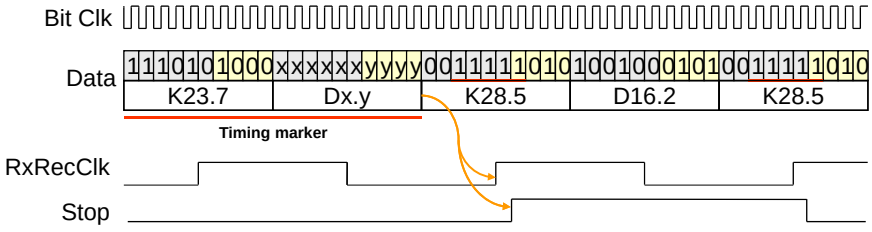

Fig. 1. Timing marker over an 8B/10B coded channel.

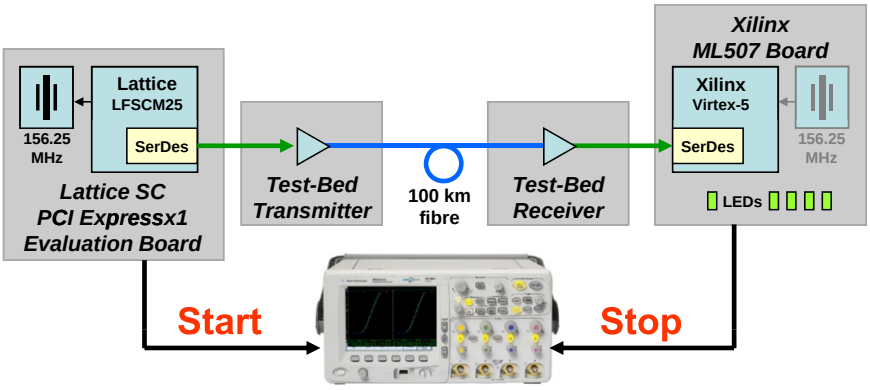

Fig. 2. Propagation delay measurement test setup.