Measuring propagation delay over a coded serial communication channel using FPGAs

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

• It is feasible to measure propagation delay over an 8B/10B coded link over 100 Km of fibre

• A 3.125 Gbps serial link implemented in FPGA provides a resolution of 320 ps.

• The method presented originates from thinking about KM3NeT timing…

• …but applies more general to “Measurement and control applications”
Measurement and control application example - I

- KM3NeT

Distributed: 1 cubic Kilometer
Synchronize system timing
High precision: ~ 1 ns
Measurement and control application example - II

- (Super) LHC

Distributed: LHC diameter 8.6 km

Synchronize system timing
High precision: aim < 100 ps

From a presentation given at the ATLAS Upgrade "ROD" Workshop
Sophie Baron – CERN
June 18, 2009
Measurement and control application common

- Distributed
  - Large systems which (often) use (serial) communication channels

- Synchronize system timing
  - Know the time offsets between clocks in the system
  - Measure offsets = measure propagation delay

- Could we use existing serial communication channels to measure propagation delay?
# Serial Communication Coding Properties

1. **Clock & Data coded into one stream**
2. **DC-Balance**
3. **Special code-groups / Word Alignment**

<table>
<thead>
<tr>
<th></th>
<th>8B/10B</th>
<th>64B/66B and 64B/67B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Length</td>
<td>5</td>
<td>Relies on Scrambler</td>
</tr>
<tr>
<td>DC Balance</td>
<td>Excellent</td>
<td>64B/66B Not guaranteed, 64B/67B Over one frame Demanding for receiver</td>
</tr>
<tr>
<td>Bit Synchronization</td>
<td>Excellent</td>
<td>Re</td>
</tr>
<tr>
<td>Clock Recovery</td>
<td>Excellent</td>
<td>1</td>
</tr>
<tr>
<td>Word Alignment</td>
<td>&quot;Comma&quot; K-Characters</td>
<td>Sync-Header</td>
</tr>
<tr>
<td>Special code-groups</td>
<td>K-Characters</td>
<td>Control-Codes</td>
</tr>
</tbody>
</table>

Measure propagation delay with high precision

- Lock the Receiver to Transmitter Clock => Clocks are Isochronous:
  - Use the same time reference
  - But have an offset

- Use **SerDes Word Alignment** information
  - Resolution one Unit-Interval (bit time)

- Using property 1 and 3 is just a hardware implementation on the OSI-model **Data Link** layer that is transparent to higher levels of hierarchy:
  - IEEE 802.3 “CSMA/CD” (Ethernet)
  - IEEE 1588 “Precision Clock Synchronization Protocol”
Test setup block diagram

- Measure propagation delay using FPGA SerDes
- Word Alignment information
- @ 3.125 Gbps
**Test setup transmission scheme**

**Tx**

<table>
<thead>
<tr>
<th>K28.5</th>
<th>D16.2</th>
<th>K23.7</th>
<th>Dx.y</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDLE</td>
<td>CharExt</td>
<td>IDLE</td>
<td>IDLE</td>
<td>IDLE</td>
<td>IDLE</td>
<td>IDLE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Start**

2 x 50 Km fiber ~ 490 us

**Rx**

<table>
<thead>
<tr>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K28.5</th>
<th>D16.2</th>
<th>K23.7</th>
<th>Dx.y</th>
<th>K28.5</th>
<th>D16.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDLE</td>
<td>IDLE</td>
<td>IDLE</td>
<td>IDLE</td>
<td>IDLE</td>
<td>IDLE</td>
<td>CharExt</td>
<td>IDLE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**3.125 Gbps / 20 bits**

= 156.25 MHz system speed (6.4 ns)

Without additional information the Stop Time is known with a 6.4 ns resolution
Real Test setup

RX (Stop) Xilinx Virtex-5 Board (ML507)

50 Km fiber

Stop

Test-bed Receiver

Test-bed Transmitter

TX (Start) Lattice LFSCM25 Board

Start

50 Km fiber

Test-bed

Real Test setup

RX (Stop) Xilinx Virtex-5 Board (ML507)

50 Km fiber

Stop

Test-bed Receiver

Test-bed Transmitter

TX (Start) Lattice LFSCM25 Board

Start
Measuring varying propagation delay

Constant Impedance Trombone Line

Reset (= resynchronize)

Resynchronize...

What is happening?
What happens during resynchronization

1. TX is transmitting a serial bit stream based on the reference clock

2. RX using the reference clock to try to lock its PLL in the CDR onto the incoming bit stream (*note: usually the TX and RX reference clock do not have the same source...*)

3. Once the PLL in the CDR is in phase, RX switches over from its reference clock to the RX recovered clock “RxRecClk” (*this happens on a random bit*)

4. Next the Word Aligner is searching for a “Comma”

5. Once a Comma is found the word aligner knows how to set its multiplexer and feed properly aligned sets of 20 bits to the FPGA fabric
Test setup block diagram

- **Lattice LFSCM25 SerDes**
- **Test-Bed Transmitter**
- **100 Km fiber**
- **Test-Bed Receiver**
- **Xilinx ML507 Board**
- **Xilinx Virtex-5 SerDes**
- **156.25 MHz**
- **LEDs**

The setup includes a 156.25 MHz signal flowing through the system from the Lattice LFSCM25 SerDes to the Test-Bed Transmitter, then through the Test-Bed Receiver, and finally to the Xilinx ML507 Board. The system is monitored by a test equipment with a green graph and the words "Start" and "Stop" indicate the start and stop points of the test.
Resynchronization in action

Algorithm:
Propagation Delay = “Start-Stop” Delay + “LED Value” * 320 ps
Algorithm Propagation Delay:
“Start-Stop” Delay + “LED Value” * 320 ps

- Leds: “00000” * 320 ps delay => “add 0 ps”
- Leds: “00001” * 320 ps delay => “add 320 ps”
- Leds: “00011” * 320 ps delay => “add 960 ps”
Add delay and Resynchronize

Algorithm:
Propagation Delay = “Start-Stop” Delay + “LED Value” * 320 ps
FPGA SerDes remarks

- The Receiver Deserializer should provide a means to (manually control) “Bit Slip”.
- Tested in:

<table>
<thead>
<tr>
<th>FlexiPCSGXBGTXSerDes Name</th>
<th>Xilinx</th>
<th>Altera</th>
<th>Lattice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Family</td>
<td>Virtex-5</td>
<td>Stratix-IV-GX</td>
<td>SC/M</td>
</tr>
<tr>
<td>SerDes Name</td>
<td>GTX</td>
<td>GXB</td>
<td>FlexiPCS</td>
</tr>
<tr>
<td>Bit Slip</td>
<td>RxSlide</td>
<td>Rx_BitSlip</td>
<td>x</td>
</tr>
<tr>
<td>Test</td>
<td>Okay</td>
<td>Okay</td>
<td>Fail</td>
</tr>
</tbody>
</table>

- Implementation verified at CEA-SAACLAY
Conclusion

- It is feasible to measure propagation delay over an 8B/10B coded link over 100 Km of fibre.
- A 3.125 Gbps serial link provides a resolution of 320 ps.
- This can be implemented in an FPGA

Thank you