Measuring propagation delay over a coded serial communication channel using FPGAs

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

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

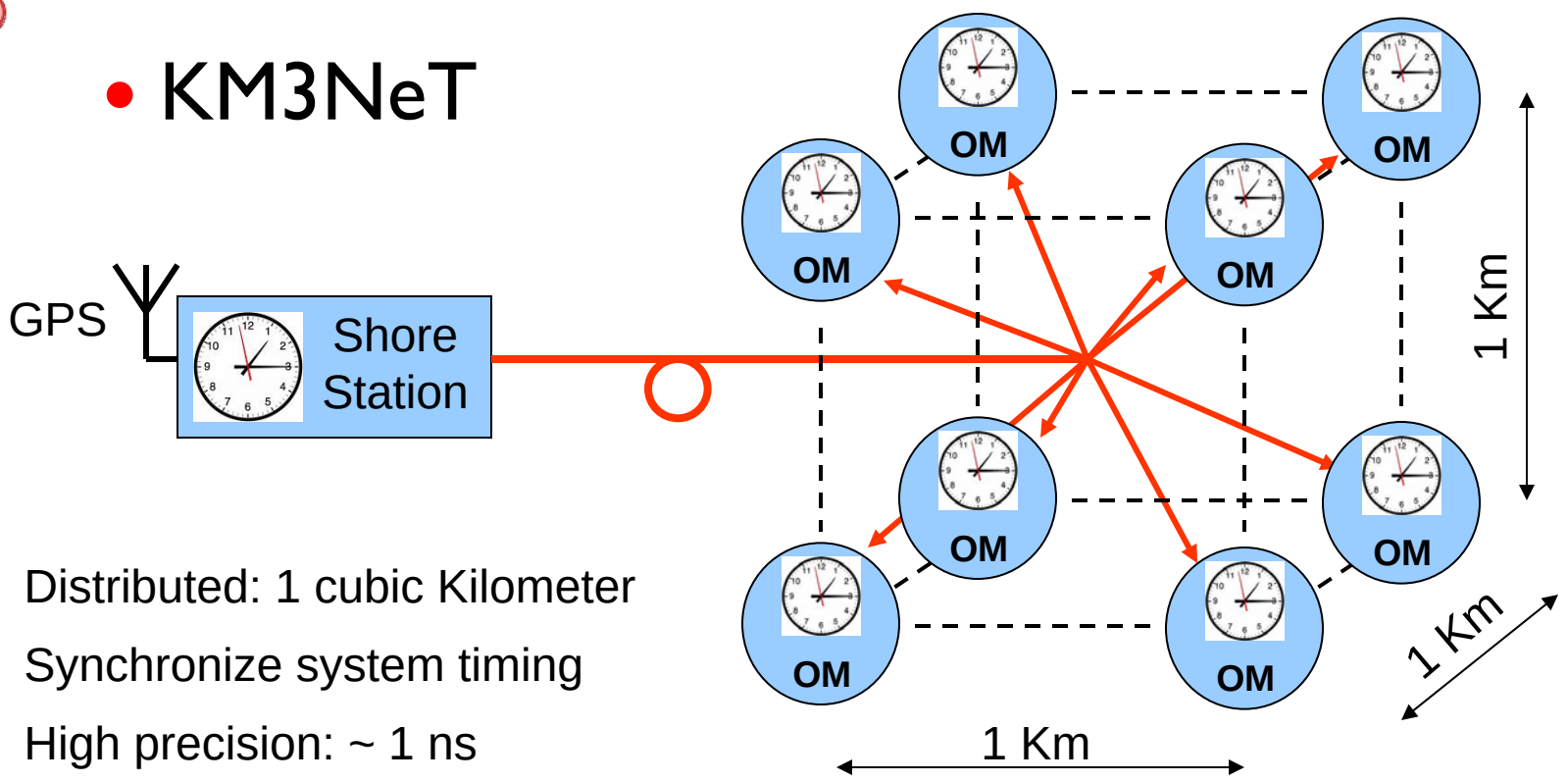
Introduction

- 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"

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

Measurement and control application example - I

- KM3NeT



Distributed: 1 cubic Kilometer

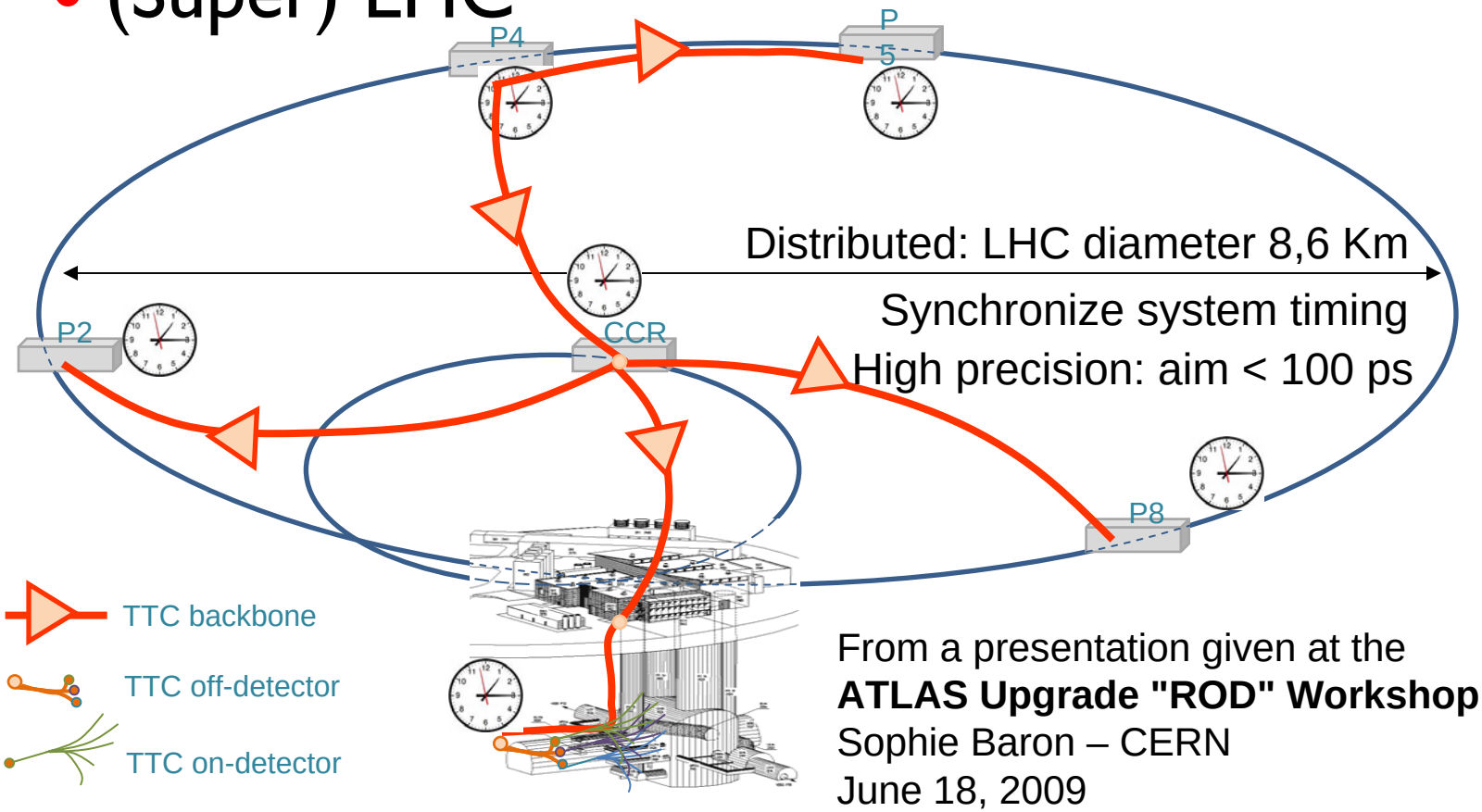
Synchronize system timing

High precision: ~ 1 ns

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

Measurement and control application example - II

- (Super) LHC



From a presentation given at the
ATLAS Upgrade "ROD" Workshop
Sophie Baron – CERN
June 18, 2009

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

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?

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

Serial Communication Coding Properties

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

	8B/10B	**64B/66B** and **64B/67B**
Run Length	5	Relies on Scrambler
DC Balance	Excellent	64B/66B Not guaranteed, 64B/67B Over one frame Demanding for receiver
Bit Synchronization Clock Recovery	**Excellent** (1)	Relies on Scrambler
Word Alignment	**"Comma" K-Characters**	Sync-Header
Special code-groups	**K-Characters** (3)	Control-Codes

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

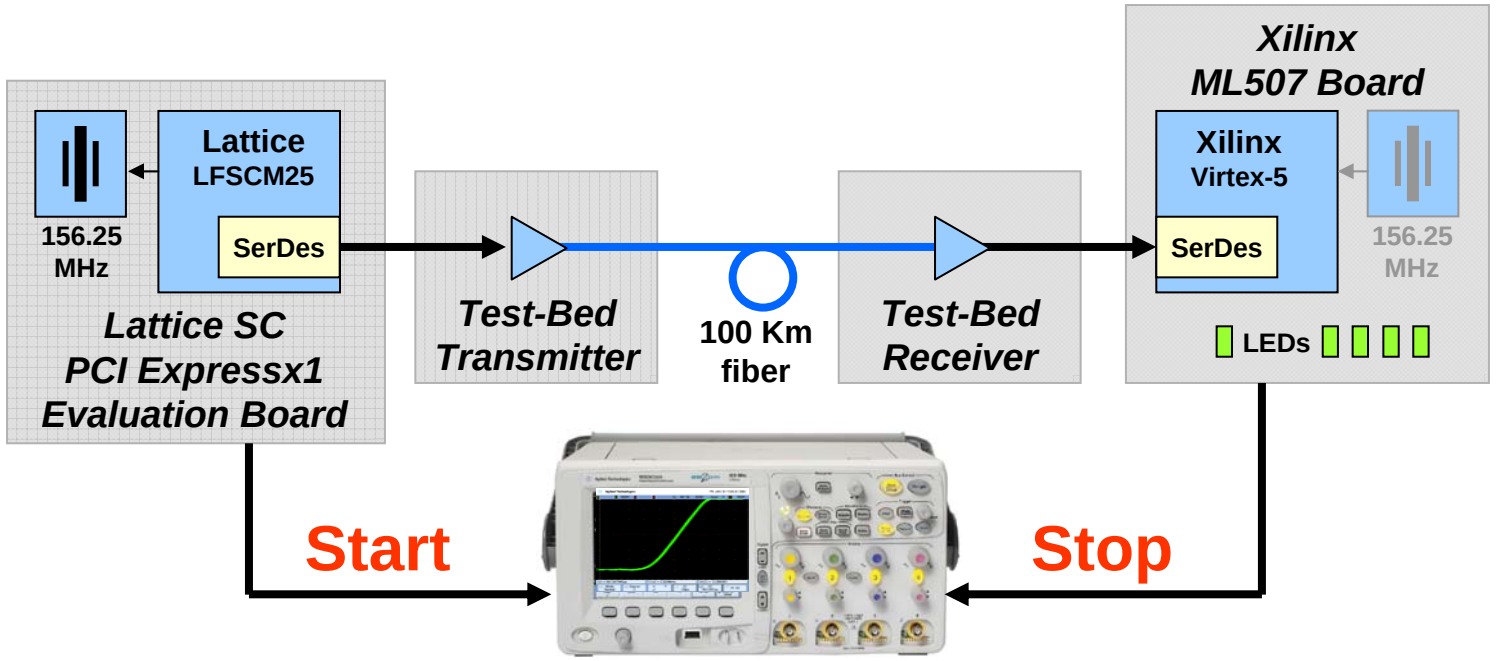
Measure propagation delay with high precision

- Lock the Receiver to Transmitter Clock => Clocks are Isochronous: 1
  - Use the same time reference
  - But have an offset
- Use **SerDes Word Alignment** information 3
  - 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"

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

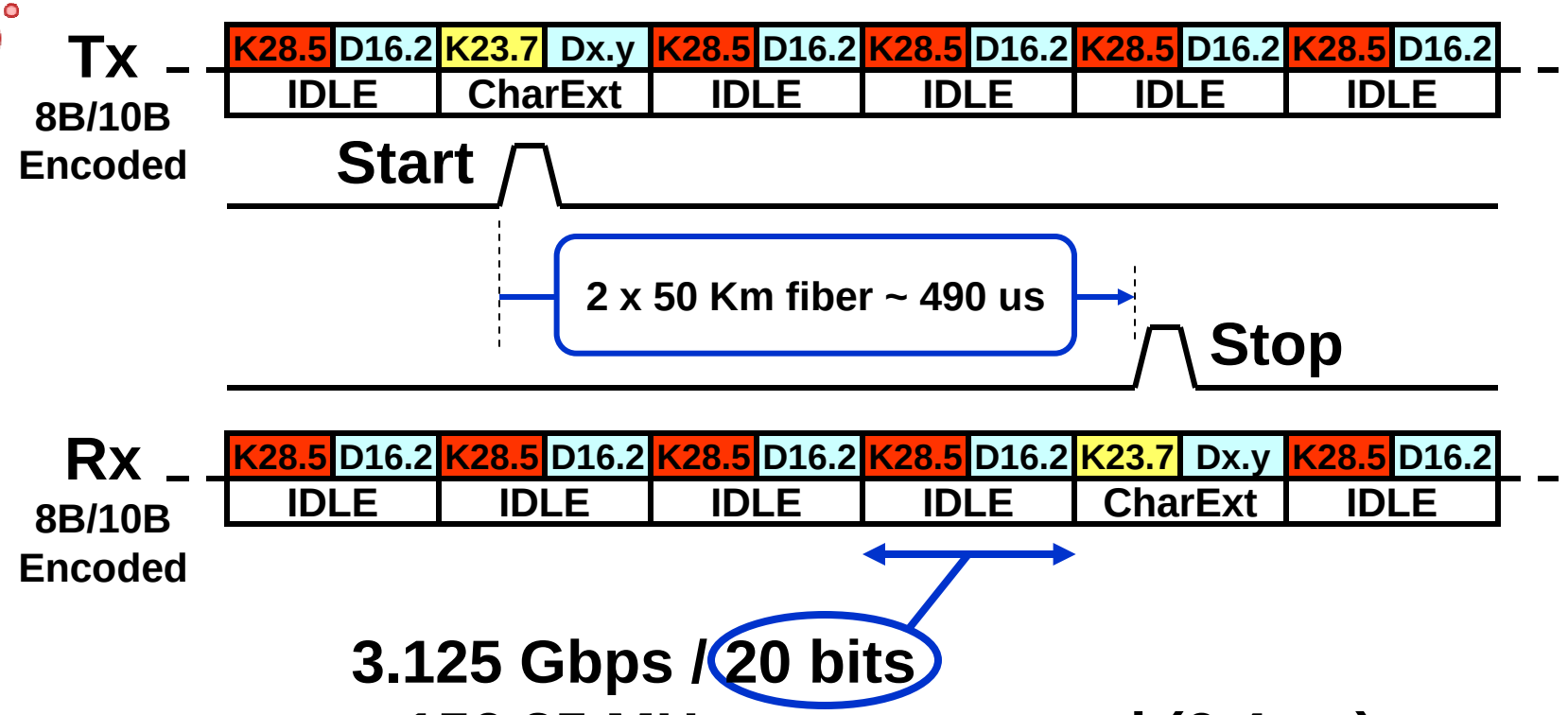
Test setup block diagram

- Measure propagation delay using FPGA SerDes Word Alignment information
- @ 3.125 Gbps



P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

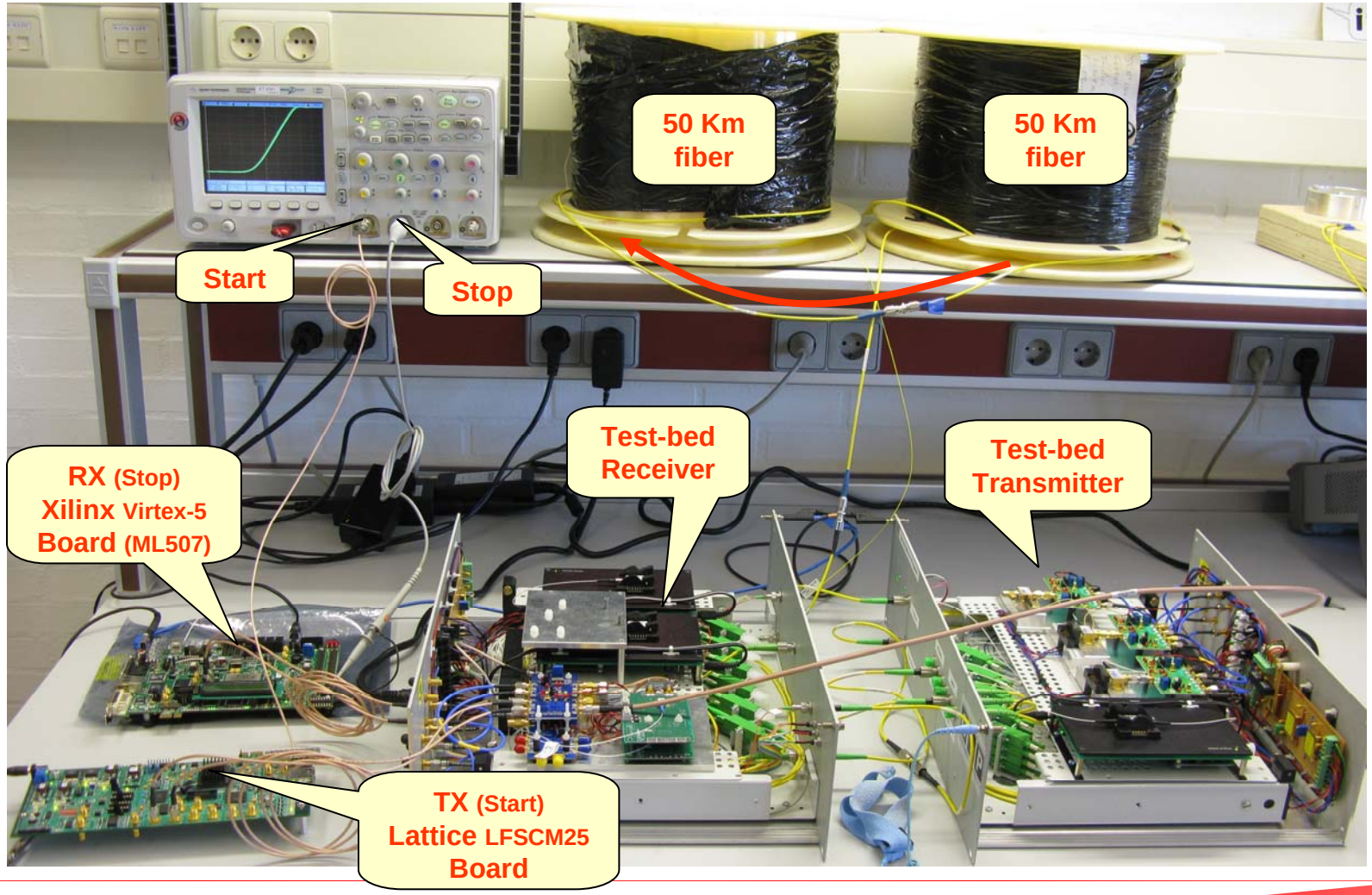
Test setup transmission scheme



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

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

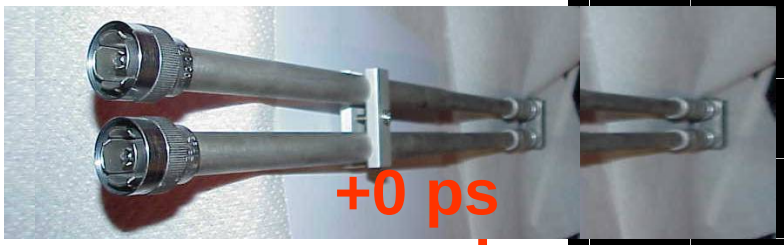
Real Test setup

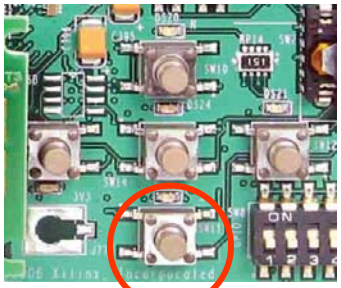


P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

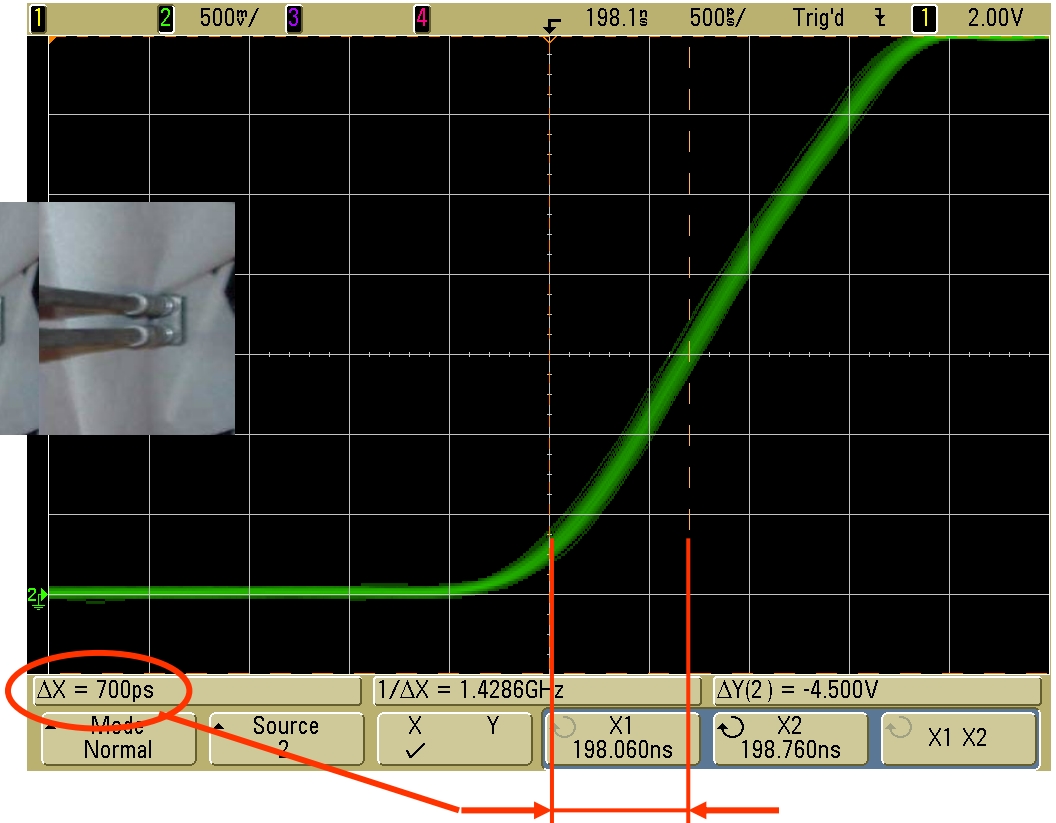
Measuring varying propagation delay

Constant Impedance
Trombone Line





Reset
(= resynchronize)



Resynchronize…
What is happening?

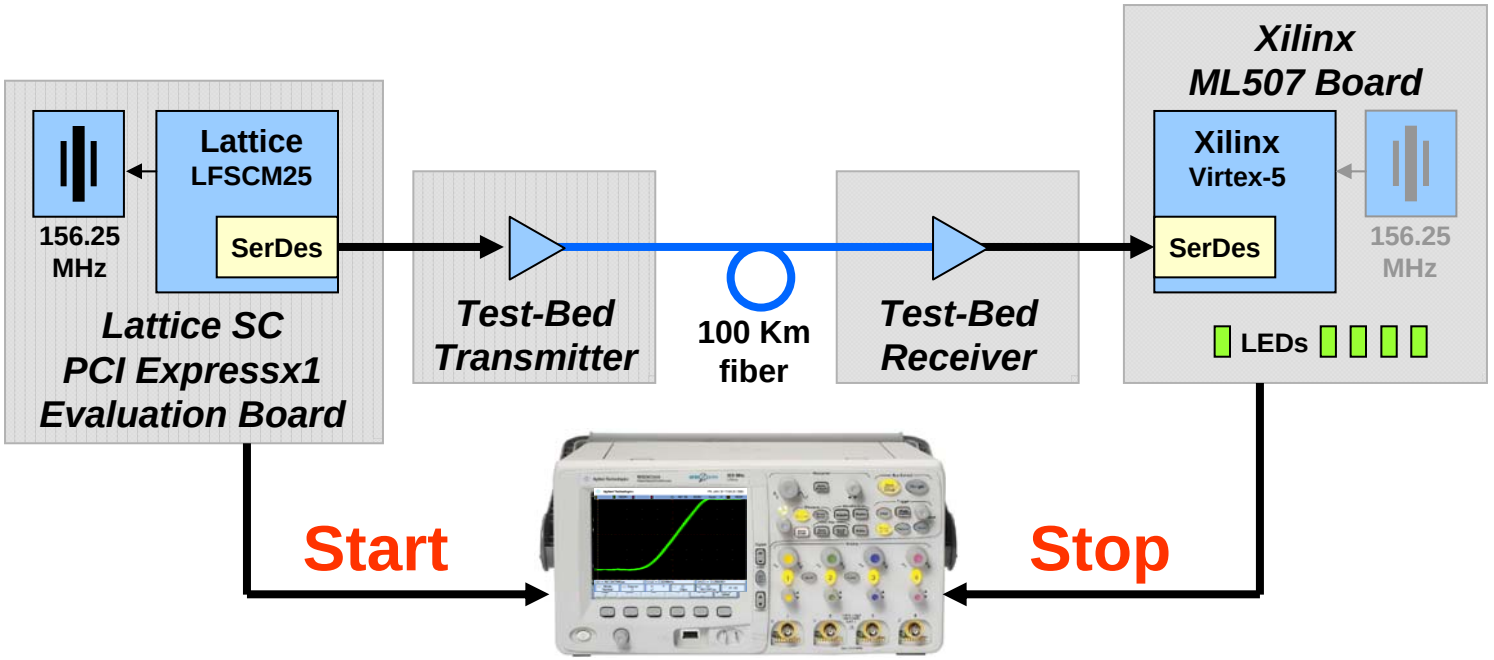
P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

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

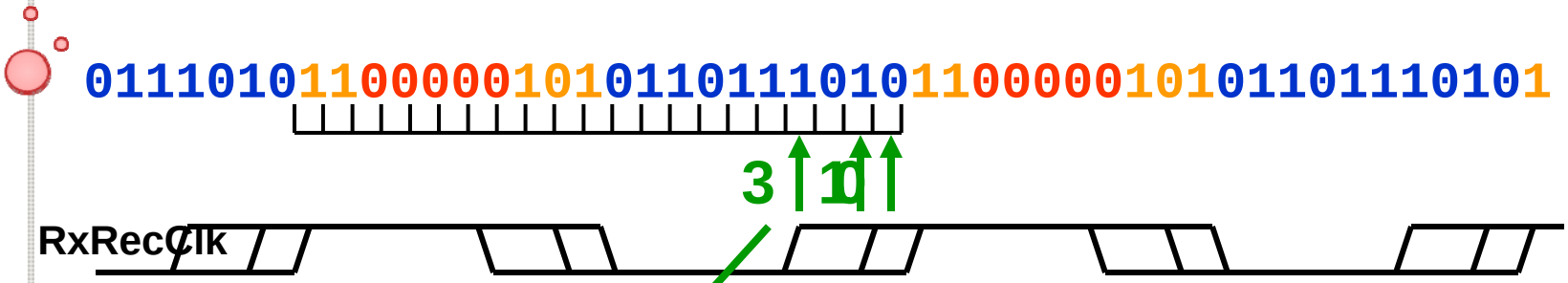
P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

Test setup block diagram



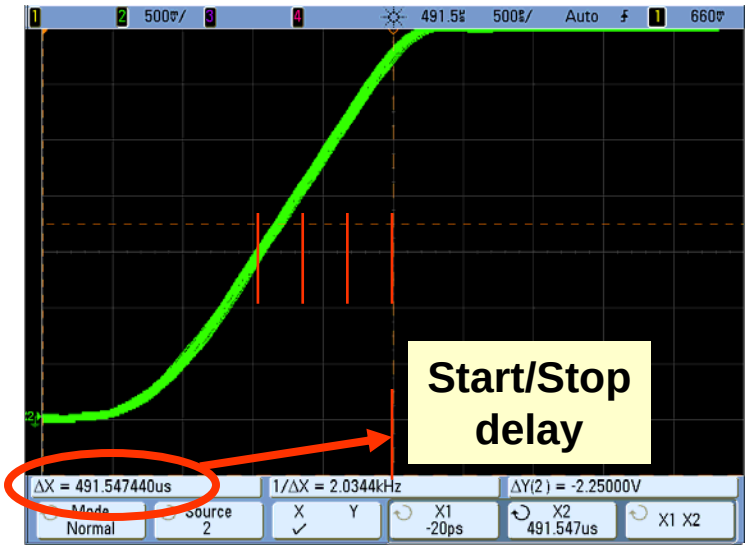
P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

Resynchronization in action



BitSlide(4:0)

0 0000 = 0



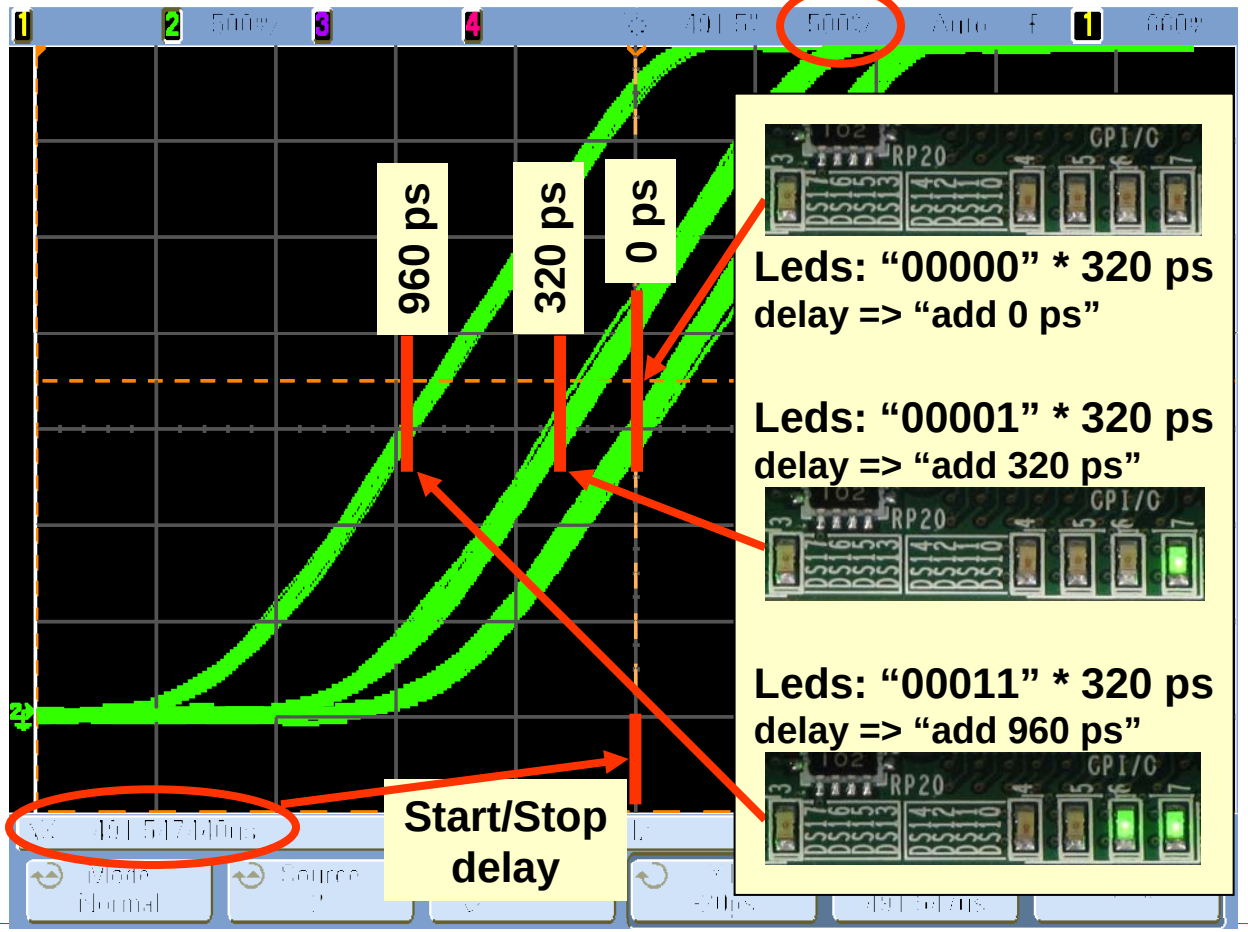
Algorithm:

Propagation Delay = "Start-Stop" Delay + "LED Value" * 320 ps

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

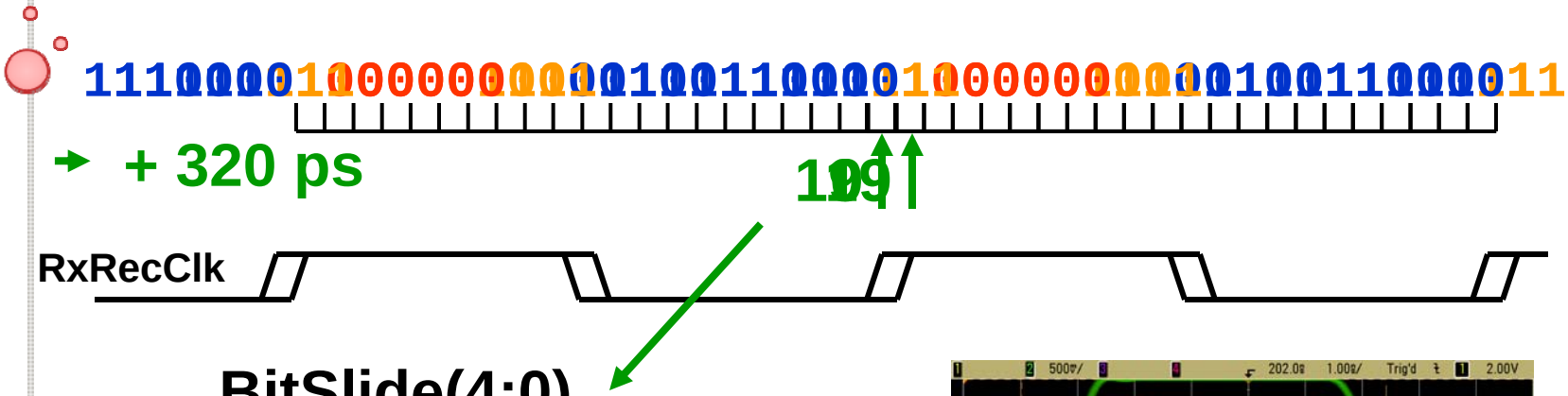
Algorithm Propagation Delay:

"Start-Stop" Delay + "LED Value" * 320 ps



P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

Add delay and Resynchronize



BitSlide(4:0)

Algorithm:

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

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

FPGA SerDes remarks

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

	Xilinx	Altera	Lattice
Family	Virtex-5	Stratix-IV-GX	SC/M
SerDes Name	GTX	GXB	FlexiPCS
Bit Slip	RxSlide	Rx_BitSlip	x
Test	Okay	Okay	Fail

- Implementation verified at CEA-SACLAY

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology

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

P. Jansweijer
Nikhef
Amsterdam
Electronics-Technology