



Department of Physics and Technology

Master Thesis

**Interface Design for the
Gigabit Transceiver Common Readout Unit**

Anders Østevik

June 2016

Introduction

oooooo

PCB Design

oooooo

GBT Control Software

oooooooooo

Testing and Verification

oooo

Conclusion

oo

Overview

Introduction

LHC Upgrade

Gigabit Transceiver System

Primary Objectives

PCB Design

Design Discussion

Transmission Lines

Design Parameters

GBT Control Software

Hardware

Software

Testing and Verification

Hardware

Software

Conclusion



●ooooo

oooooo

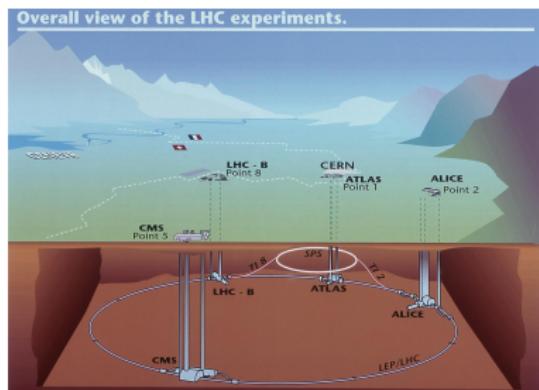
oooooooooooo

oooo

oo

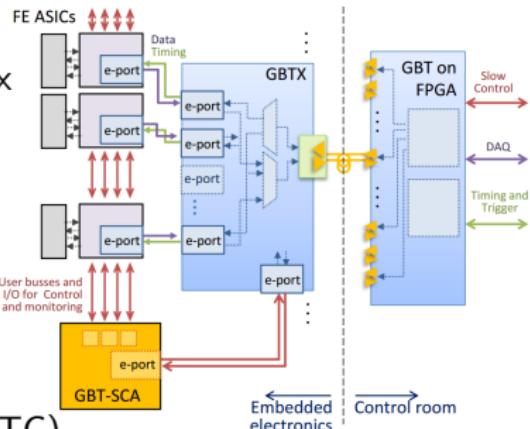
LHC Upgrade

- Large Hadron Collider (LHC)
 - Particle accelerator
 - 27 km circular tunnel
 - 13 TeV
- High-Luminosity LHC
 - 10x beam luminosity
 - Increase in radiation and amount of data
 - → Gigabit Transceiver



Gigabit Transceiver System

- On-detector - Custom ASICs
 - GBTx, GBT-SCA, VTTx/VTRx
 - E-links
- Off-detector - Control room
 - CRU (FPGA)
 - > 4.8 Gbit/s transceivers
 - GBT-FPGA
- Optical communication
 - Timing and Trigger Control (TTC)
 - Data Acquisition (DAQ)
 - Slow Control (SC)



○○●○○○

○○○○○○

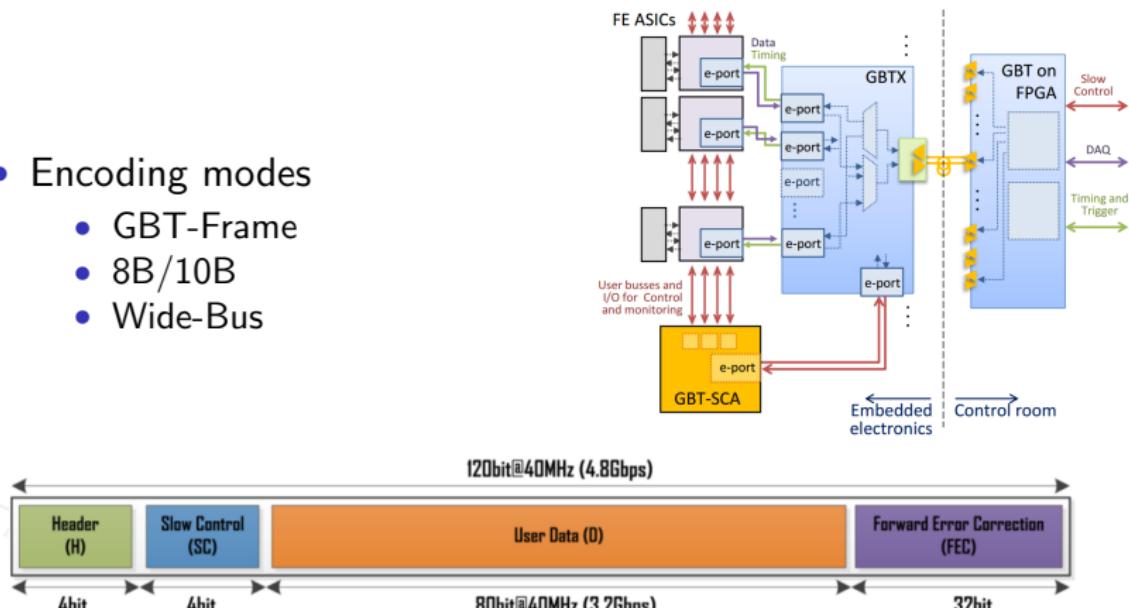
○○○○○○○○○○

○○○○

○○

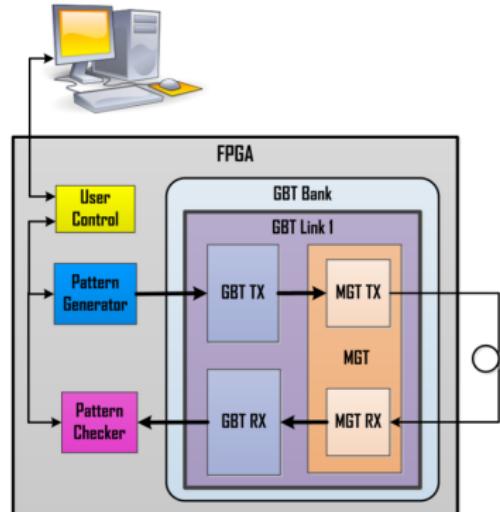
Gigabit Transceiver System

- Encoding modes
 - GBT-Frame
 - 8B/10B
 - Wide-Bus

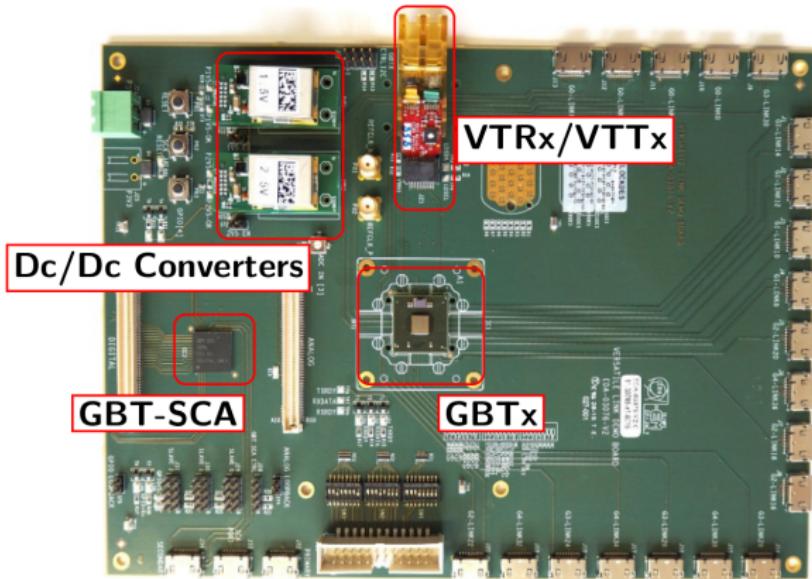


GBT-FPGA

- Firmware library for Altera/Xilinx FPGAs
- GBT Link
 - "Standard", "Latency-Optimized"
 - GBT Rx, GBT Tx, GBT MGT
- GBT-example Design



Versatile Link Demo Board



oooooo•

oooooo

oooooooo

oooo

oo

Primary Objective

- Software Design
 - Serial communication between PC and CRU
 - Interface allowing control over CRU
- PCB Design
 - Connection between CRU and VLDB



oooooo

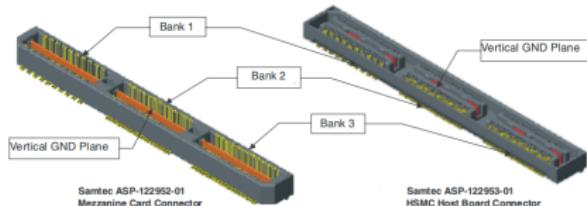
●ooooo

oooooooooooo

oooo

oo

PCB Design



Specifications:

- Connect to CRU using HSMC-connector
- E-links → 320 Mbit/s detector data → LVDS
- Optical-Fiber → 4.8 Gbit/s GBT data → PCML



oooooo

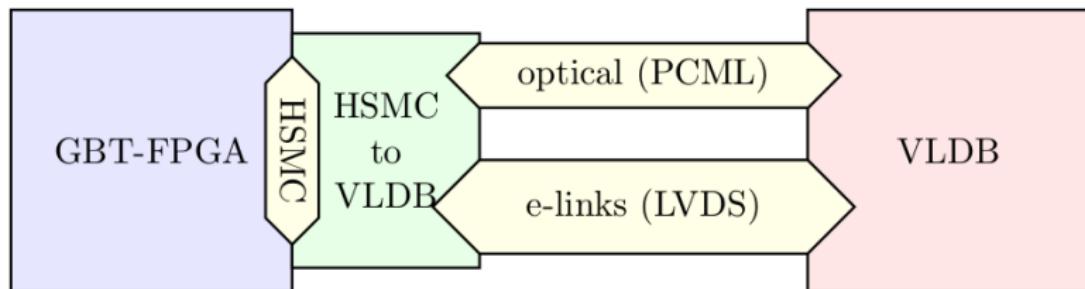
o•oooo

oooooooooooo

oooo

oo

PCB Design



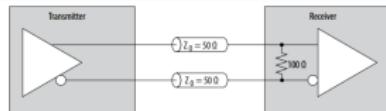
Transmission Lines

$$v = \frac{c}{\sqrt{\epsilon_r}}, \quad \epsilon_r \approx 4$$

- $v = 15 \text{ cm/ns}$
- 4.8 Gbit/s (0.2 ns) → transmission line if trace < 3.1 cm
- Characteristic impedance, $Z_0 = 50$
- Differential impedance, $Z_{diff} = 100$

Figure 5-17: Differential Input OCT

The Cyclone V devices support OCT for differential LVDS and SLVS input buffers with a nominal resistance value of 100 Ω , as shown in this figure.



oooooo

oooo●ooo

oooooooooooo

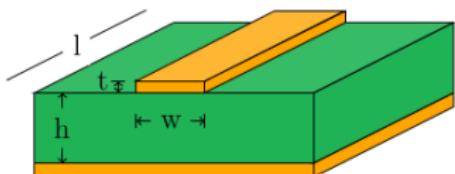
oooo

oo

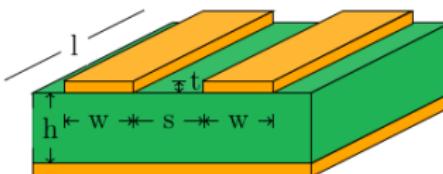
Transmission Lines

$$Z_0 = \frac{60}{\sqrt{0.475\epsilon_r + 0.67}} \times \ln \frac{4h}{0.67(0.8w + t)}$$

$$Z_{diff} = 2 \times Z_0 [1 - 0.48e^{(-0.96 \times \frac{s}{h})}]$$



(a) Single-ended.



(b) Differential.



oooooo

oooo●○

oooooooooo

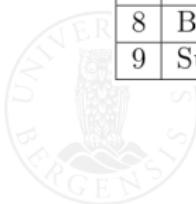
oooo

oo

Design Parameters

- Routing differential signals:
 - Equal trace lengths on each pair
 - As close as possible, but keep distance to other pairs
 - As straight paths as possible

	Layer	Type	t [μm]	ϵ_r	w [μm]	$Z_0[\Omega]$	Spacing [μm]	$Z_{diff}[\Omega]$
1	Surface	Air		1				
2	Top	Conductor	18		100	51.3	300	102
3		Dielectric	65	4.05				
4	Voltage	Plane	35					
5		Dielectric	1400	4.05				
6	Gnd	Plane	35					
7		Dielectric	65	4.05				
8	Bottom	Conductor	18		100	51.3	300	102
9	Surface	Air		1				



oooooo

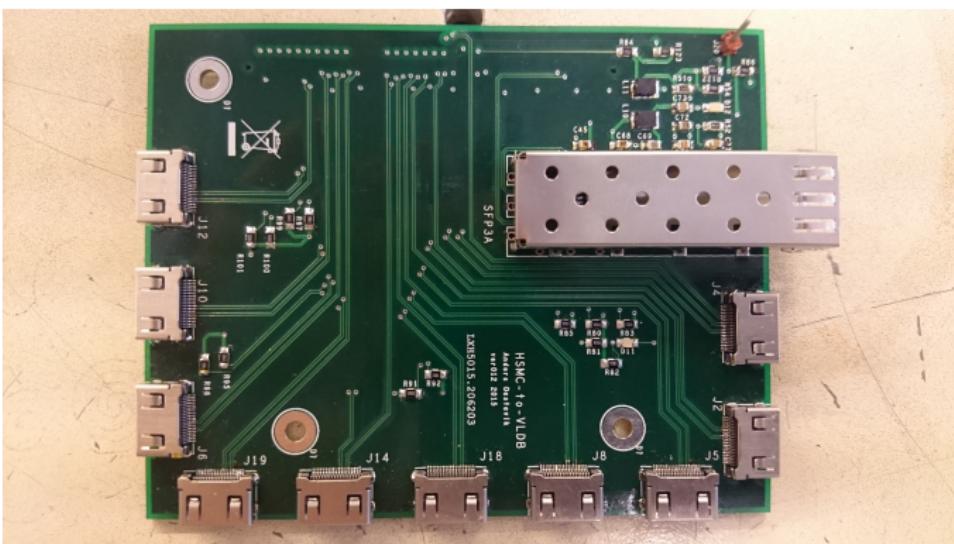
ooooo●

oooooooooooo

oooo

oo

Result



- Each HDMI has an input pair and an output pair
- J4 has an additional pair for input clock from VLDB



oooooo

oooooo

●oooooooo

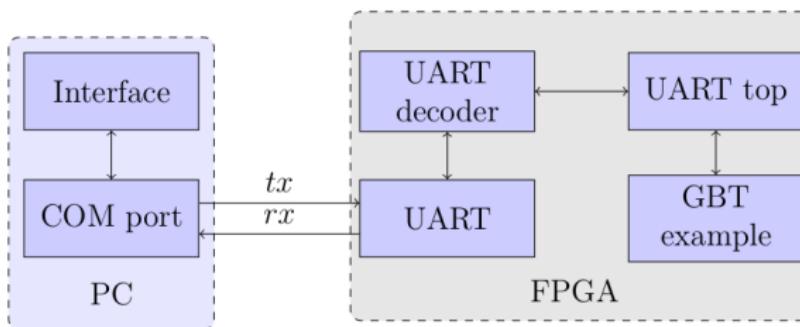
oooo

oo

GBT Control Software

Specifications:

- Provide communication between PC and FPGA
- Read and write GBT control signal information
- Provide a simple User Interface



	Switches	Probes
GBT register	0x00...0x23	0x24...0x41



oooooo

oooooo

o●oooooooo

oooo

oo

Hardware

- Treat incoming requests sent from PC
- Components:
 - UART (RX, TX, Buffers, baudrate generator)
 - Decoder
- Goal: implement into GBT Example Design



oooooo

oooooo

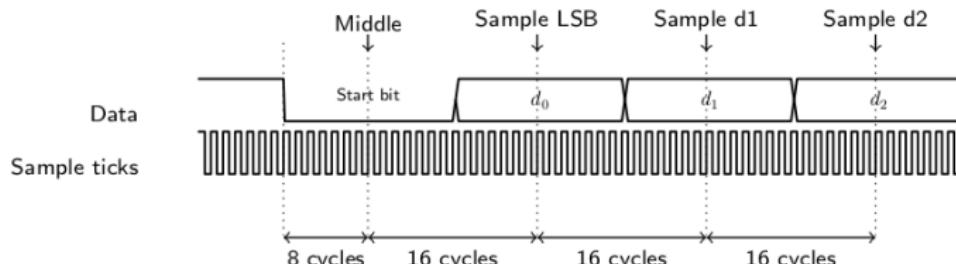
○○●○○○○○

○○○○

○○

UART

- Synchronisation → Oversampling
- $16 \times \text{baudrate}$
- Stores bytes in FIFO-buffers
- $19200 - 8 - N - 1 \rightarrow 307200$ sample clock



oooooo

oooooo

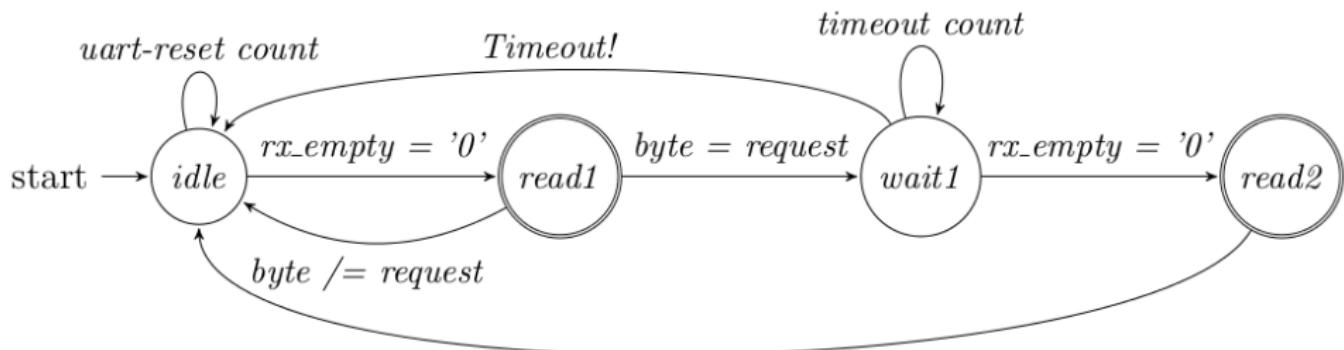
oooo●ooooo

oooo

oo

Decoder

- Requests:
 - read → 0xDD
 - write 0 → 0xEE
 - write 1 → 0xFF
- Legal addresses:
 - 0x00 → 0x41
- Bytes sent to PC:
 - 0x00 → 0xC1



oooooo

oooooo

oooo●oooo

oooo

oo

Software

Modules:

- Send/Receive
 - RS-232 library
 - Continuously transmits pattern to GBT-registers
 - Reads it back for transmit confirmation
- User Interface
 - ncurses library
 - User commands: write, read, status, open, close, exit
 - Dummy registers instead of real GBT-registers
- GBT register data → stored and maintained using a custom *Signals*-library



oooooo

oooooo

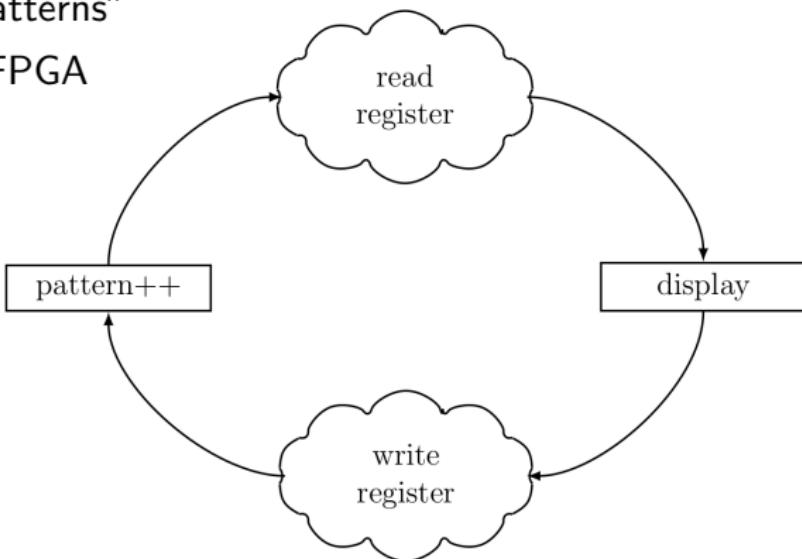
oooooo●oooo

oooo

oo

Transmit/Receive Module

- Transmit 2 bytes, receive 1 byte
- 2D-array with "patterns"
- COM-port \longleftrightarrow FPGA



oooooo

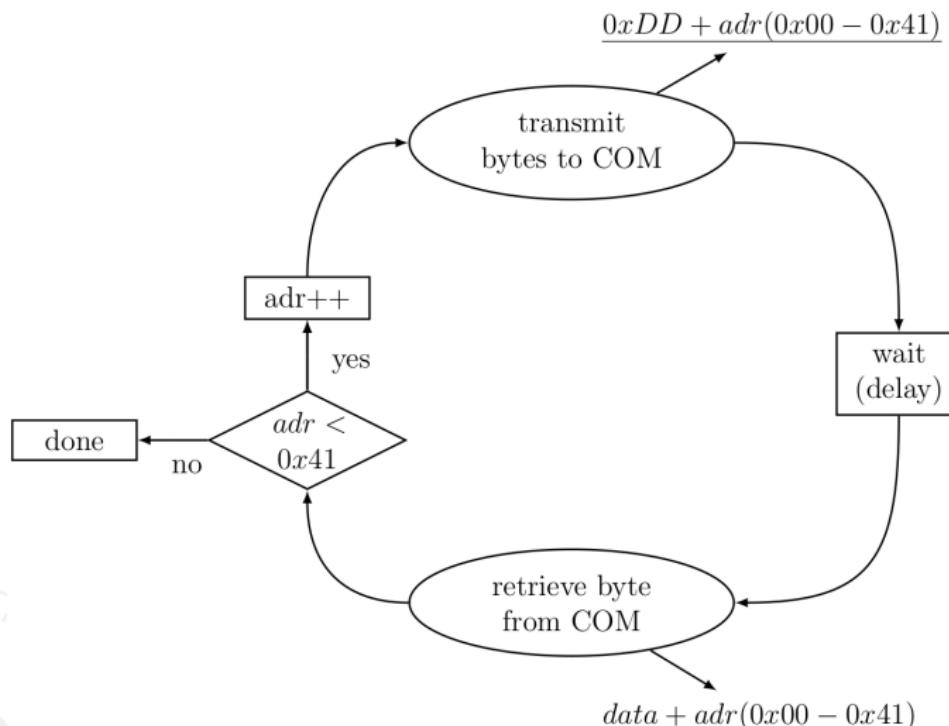
oooooo

oooooooo●○○

○○○○

○○

Read register



oooooo

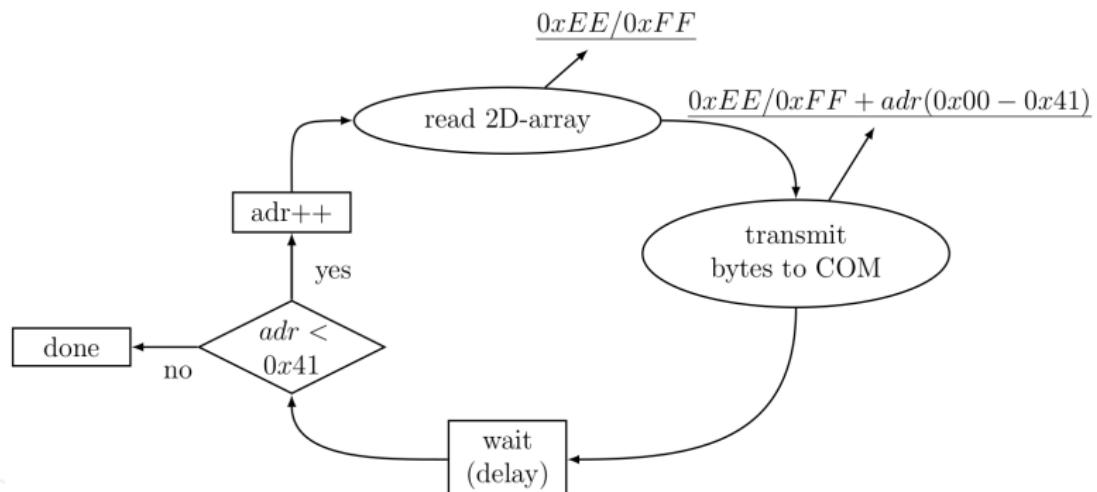
oooooo

oooooooo●○

oooo

oo

Write register



ooooooo

ooooooo

oooooooo●

oooo

oo

User Interface

Command History:-

```
: status
usage: status [settings/comport]
: st
status   [settings/comport]
exit    [status (1 or 0)]
: a
read   [[name0] [name1] ...]
write  [data [name1] [name2] ...]
status  [settings/comport]
exit   [status (1 or 0)]
: r
read   [[name0] [name1] ...]
write  [data [name1] [name2] ...]
status  [settings/comport]
open   [comport]
close  [comport]
exit   [status (1 or 0)]
: e
read   [[name0] [name1] ...]
write  [data [name1] [name2] ...]
status  [settings/comport]
open   [comport]
close  [comport]
exit   [status (1 or 0)]
: status settings
Current settings:
Baud Rate -> 57600
COMport nr.-> 4 (see rs232.c)
COMport node -> 8N1

: status comport
COMport status -> closed
```

Signals:-

Index:	Name:	Data:
S00	LOOPBACK ('0' -> NORMAL '1' -> LOOPBACK)	1
S01	GENERAL RESET	0
S02	MGT TX PLL RESET	0
S03	TX RESET	0
S04	RX RESET	0
S05	PATTERN SELECT ('1' -> COUNTER '2' -> STATIC)	1
S06	PATTERN SELECT ('1' -> COUNTER '2' -> STATIC)	0
S07	TX HEADER SELECTOR ('0' -> IDLE '1' -> DATA)	1
S08	RESET DATA EXTRA DATA ERROR SEEN FLAGS	0
S09	RESET RX GBT READY LOST FLAG	0
S10	TX_FRAMECLK PHASE ALIGNER - MANUAL RESET	0
S11	TX_FRAMECLK PHASE ALIGNER - GBT LINK 1 STEPS	0
S12	TX_FRAMECLK PHASE ALIGNER - GBT LINK 1 STEPS	0
S13	TX_FRAMECLK PHASE ALIGNER - GBT LINK 1 STEPS	0
S14	TX_FRAMECLK PHASE ALIGNER - GBT LINK 1 STEPS	0
S15	TX_FRAMECLK PHASE ALIGNER - GBT LINK 1 STEPS	0
S16	TX_FRAMECLK PHASE ALIGNER - GBT LINK 1 STEPS	0
S17	TX_FRAMECLK PHASE ALIGNER - ENABLE	0
S18	TX_FRAMECLK PHASE ALIGNER - TRIGGER	0
S19	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S20	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S21	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S22	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S23	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S24	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S25	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S26	TX_WORLDCLOCK MONITORING - THRESHOLD UP	0
S27	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S28	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S29	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S30	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S31	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S32	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S33	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S34	TX_WORLDCLOCK MONITORING - THRESHOLD LOW	0
S35	TX_WORLDCLOCK MONITORING - TX RESET ENABLE	0

Commandline:-

-

- Ex.: "write 1 s01 s02 s03"

oooooo

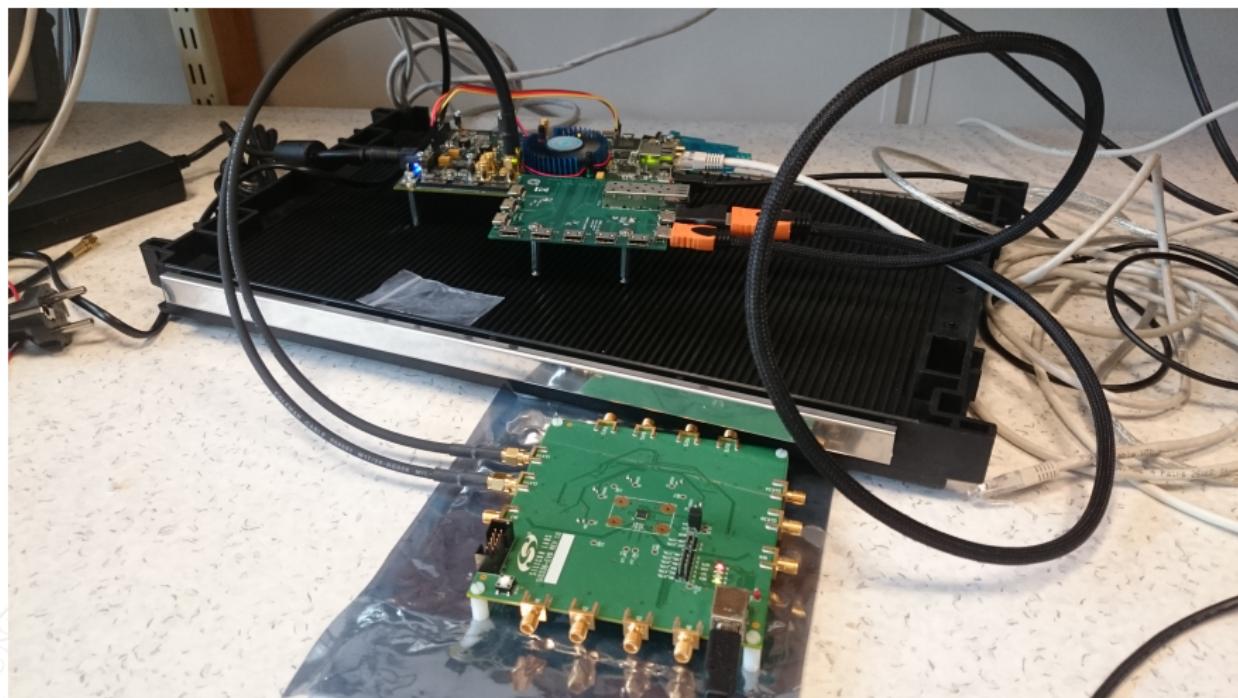
oooooo

oooooooooooo

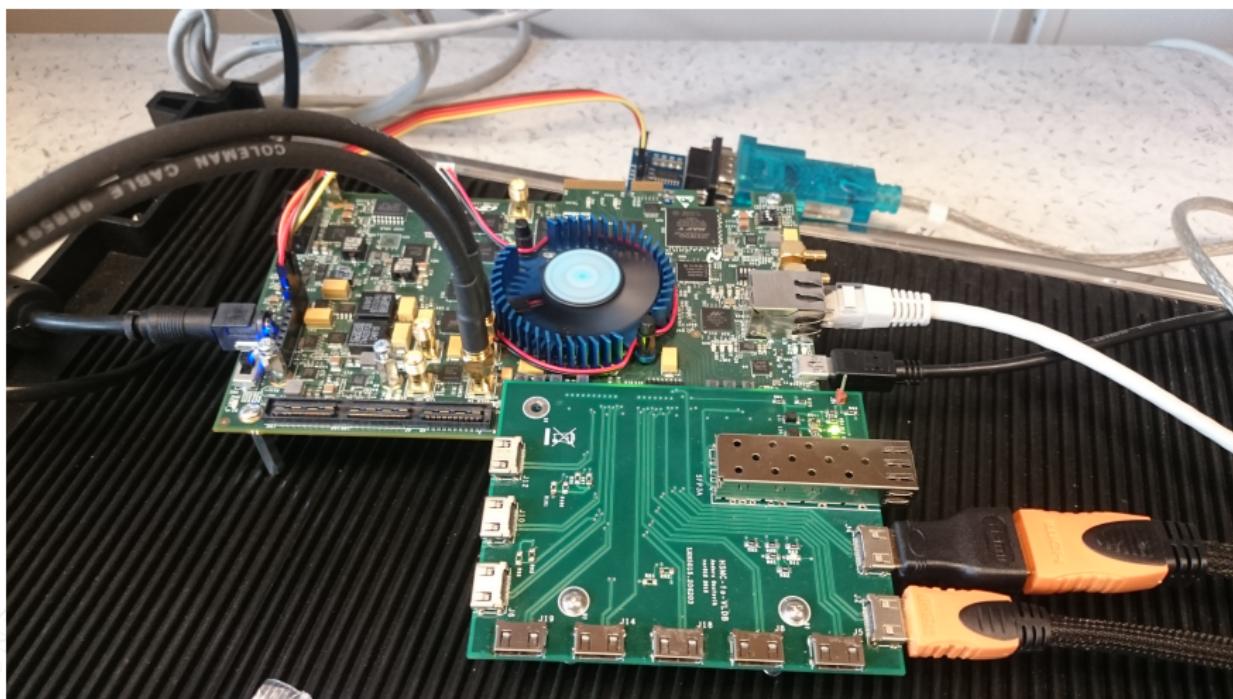
●○○○

OO

Test Setup



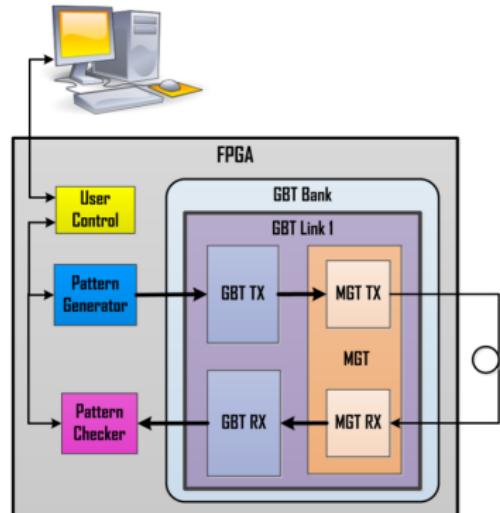
Test Setup



External Loop-back Test

Optical Fiber Connector (PCML):

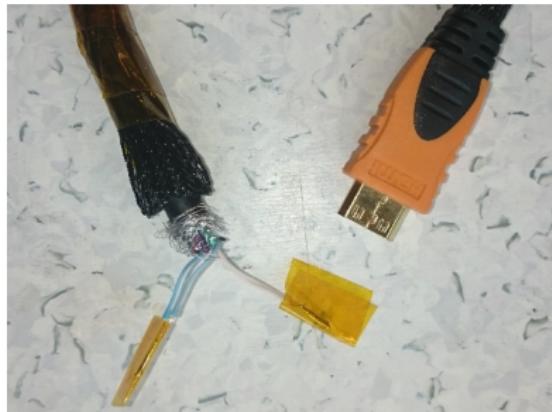
- GBT example Design
- 4.8 Gbit/s *counter*
- SignalTap II → sample TX and RX



External Loop-back Test

HDMI connector (LVDS):

- Purpose: Transmission of 320 Mbit/s E-link detector data
- Test: Loop-back a 300 MHz Pseudo-random counter
- Measure:
 - bit-error count
 - eye-diagram
 - crosstalk
 - reflections



oooooo

oooooo

oooooooooooo

oooo

●○

Conclusion

- PCB:
 - Optical Fiber module **works** → 4.8 Gbit/s external loopback
 - HDMI LVDS connectors **needs more testing** → eye-diagram, crosstalk, reflections, bit-error count.
- Hardware:
 - UART + Decoder **works**
 - Implementation into the GBT Example Design remains
- Software:
 - Send/Receive and User Interface modules **works**
 - Both needs to be merged
- Final test of the whole system:
 - GBT example + Hardware + Software + PCB-loopback
 - Test system together with VLDB



Introduction

PCB Design

GBT Control Software

Testing and Verification

Conclusion

oooooo

oooooo

oooooooo

oooo

○●

Thank you!

