GBT-FPGA Tutorial

Introduction

Schedule

- 11:00 – 12:30: Introduction to the GBT-FPGA
  - General overview
  - GBT-FPGA structure
  - How to importing the GBT-FPGA into your project ?
- 13:25 – 15:00: How to use the GBT-FPGA IP in standard mode
  - Demo setup (Arria 10 based)
  - How to create a reference design from scratch ?
  - How to debug the design ?
- 15:20 – 16:20: How to use the GBT-FPGA IP in latency-optimized mode
  - How to improve the design to use the latency-optimized mode ?
  - How to debug the design ?
- 16:20 – 17:00: How to use an existing reference design (KC705)
- 17:00 – 18:00: Tips and tricks

GBT-FPGA Tutorial

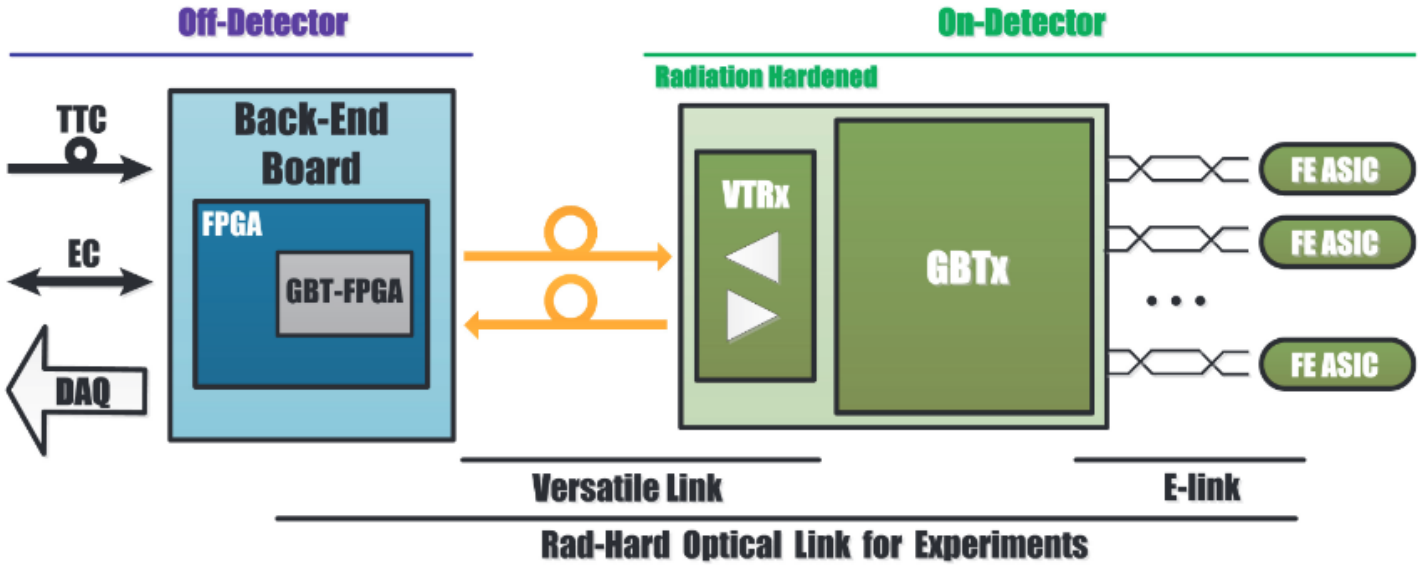
Overview: code structure

Outline

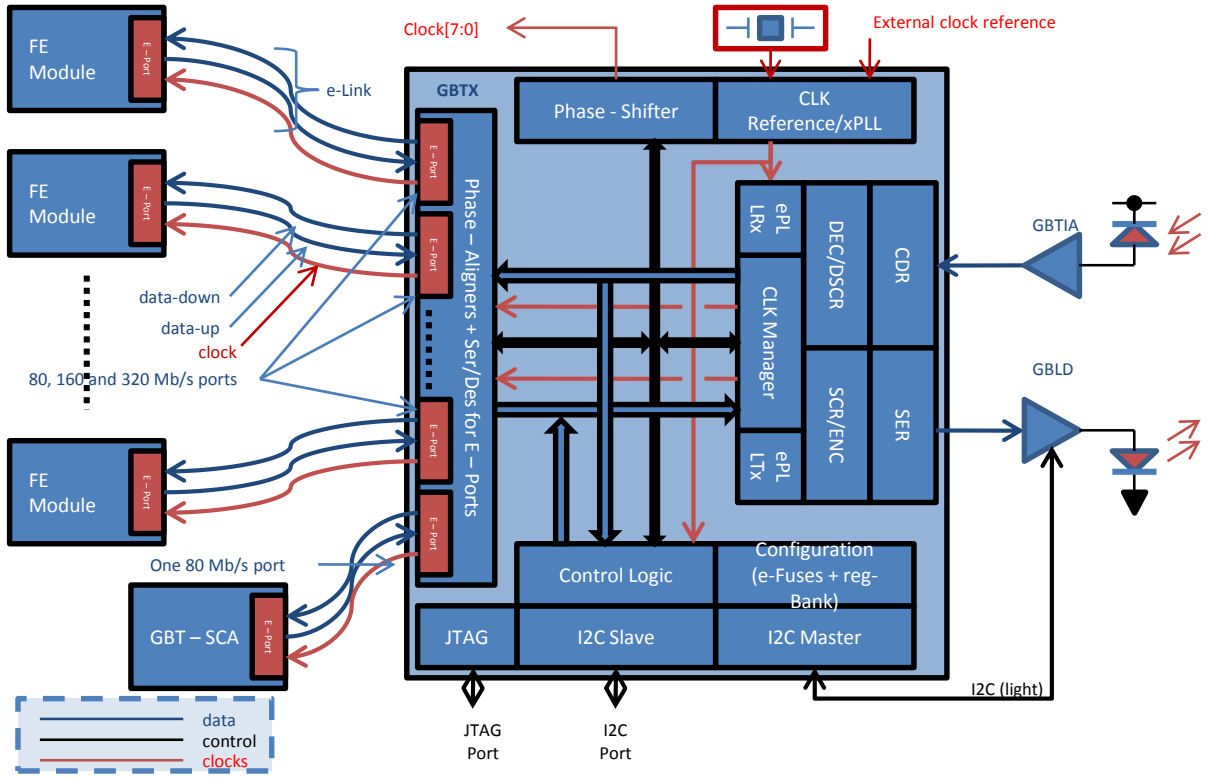
- General overview
- SVN Repository architecture
- Import the GBT-FPGA IP into my project

General overview

- What is the GBT-FPGA ?
  - Project initiated in 2009
  - Library to emulate a GBT serdes in an FPGA (e-links not handled)
  - Targets FPGA from Altera and Xilinx
  - Supported by EP-ESE-BE section at CERN



GBTx


FE Module
E – Port
e-Link
Clock[7:0]
External clock reference
GBTX
Phase - Shifter
CLK Reference/xPLL
Phase – Aligners + Ser/Des for E – Ports
ePL LRx
DEC/DSCR
CDR
CLK Manager
SCR/ENC
SER
ePL LTx
GBTIA
GBLD
data-down
data-up
clock
80, 160 and 320 Mb/s ports
One 80 Mb/s port
GBT – SCA
Control Logic
Configuration (e-Fuses + reg-Bank)
JTAG
I2C Slave
I2C Master
I2C (light)
JTAG Port
I2C Port
data
control
clocks

GBTx serial streams

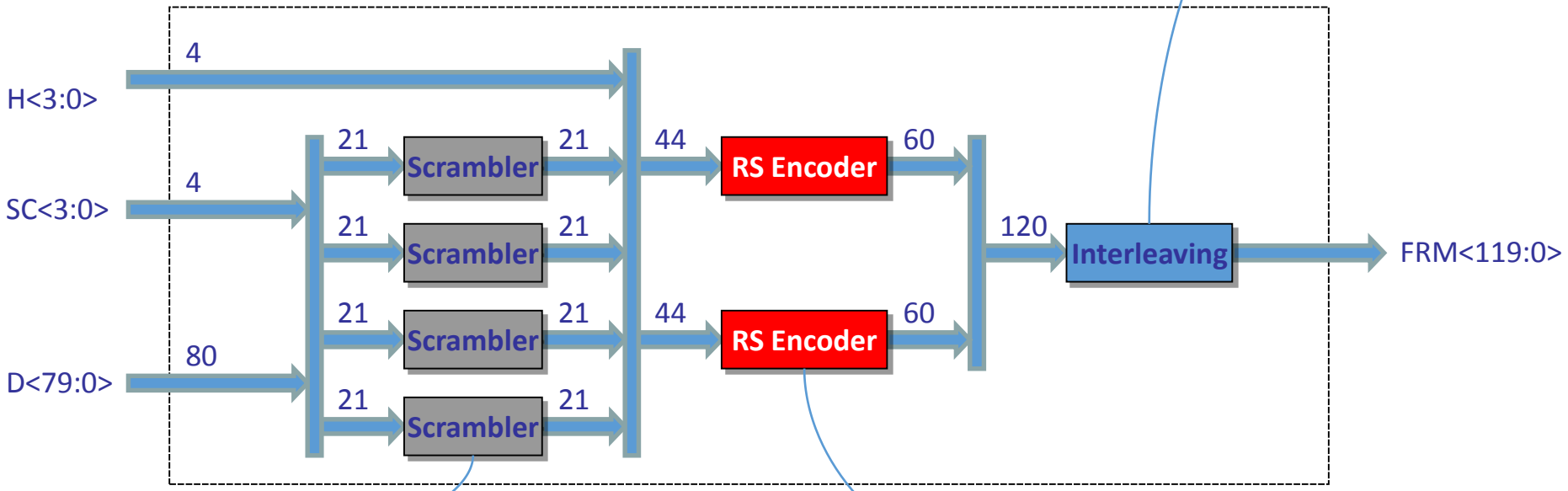
- Downstream: GBT frames
  - Scrambler
  - Reed Solomon encoder
  - Interleaver

- Upstream:

  GBT frames…

  … or Widebus frames

GBT Frame

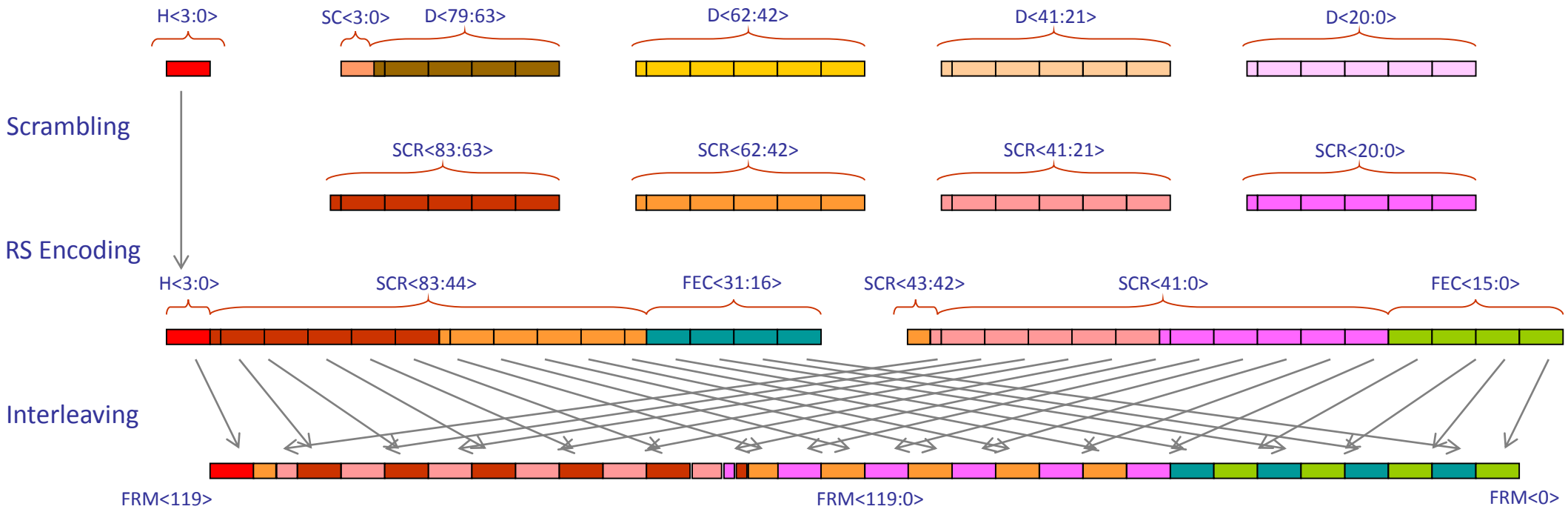


Adding the interleaving allows having up to 16 consecutive bit errors to be decoded

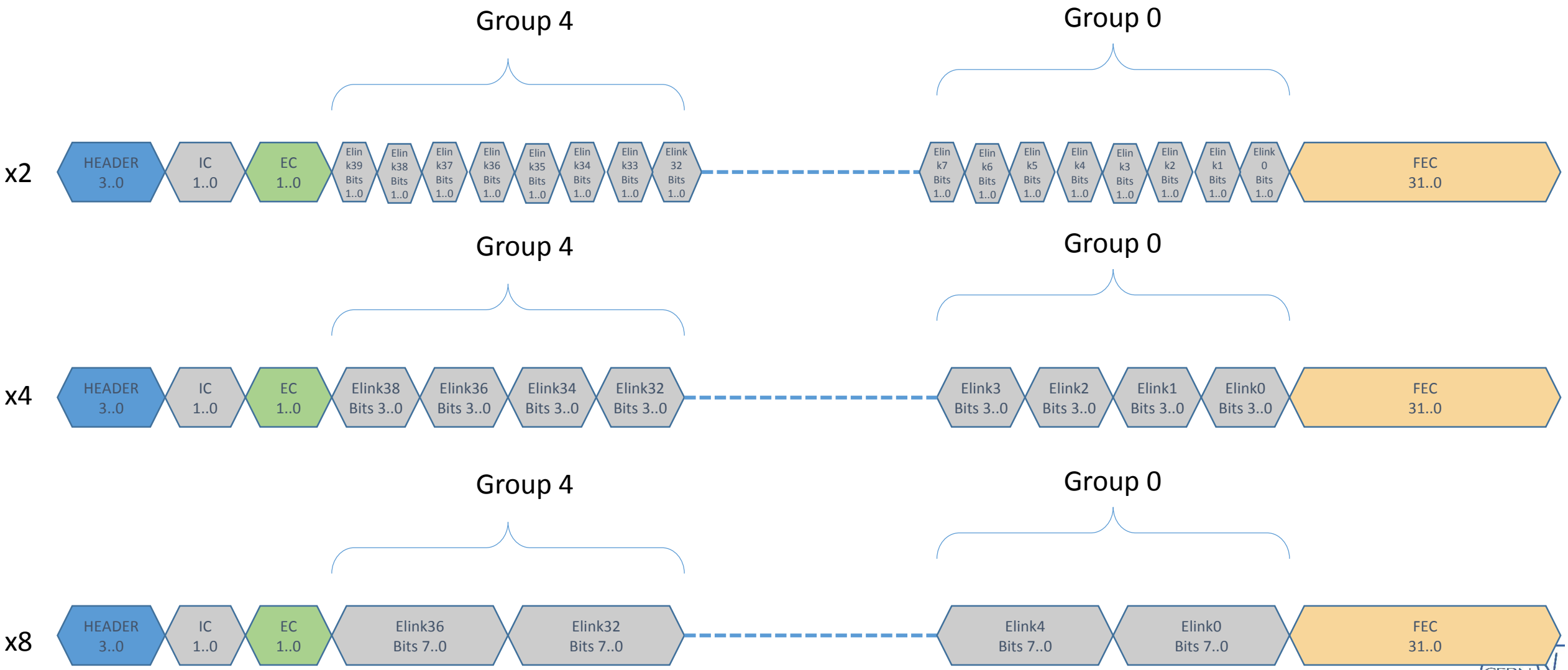
"randomization" of the data to ensure a good balance of the signal

Ad-hoc encoding scheme allowing the correction of 2 symbols of 4 bits per encoder

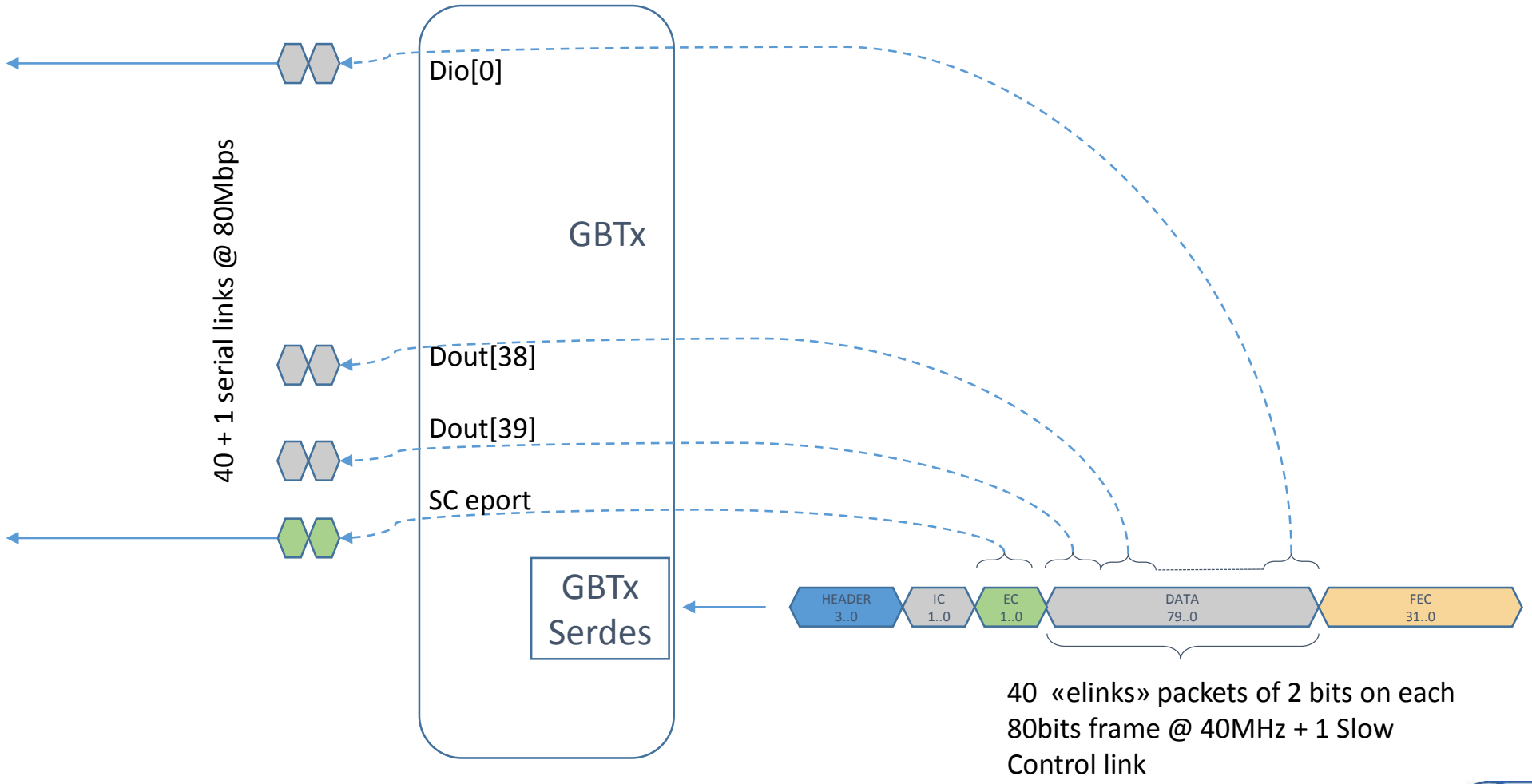
GBT Frame


H<3:0>
SC<3:0>
D<79:63>
D<62:42>
D<41:21>
D<20:0>
Scrambling
SCR<83:63>
SCR<62:42>
SCR<41:21>
SCR<20:0>
RS Encoding
H<3:0>
SCR<83:44>
FEC<31:16>
SCR<43:42>
SCR<41:0>
FEC<15:0>
Interleaving
FRM<119>
FRM<119:0>
FRM<0>

GBT Frame construction vs elink modes

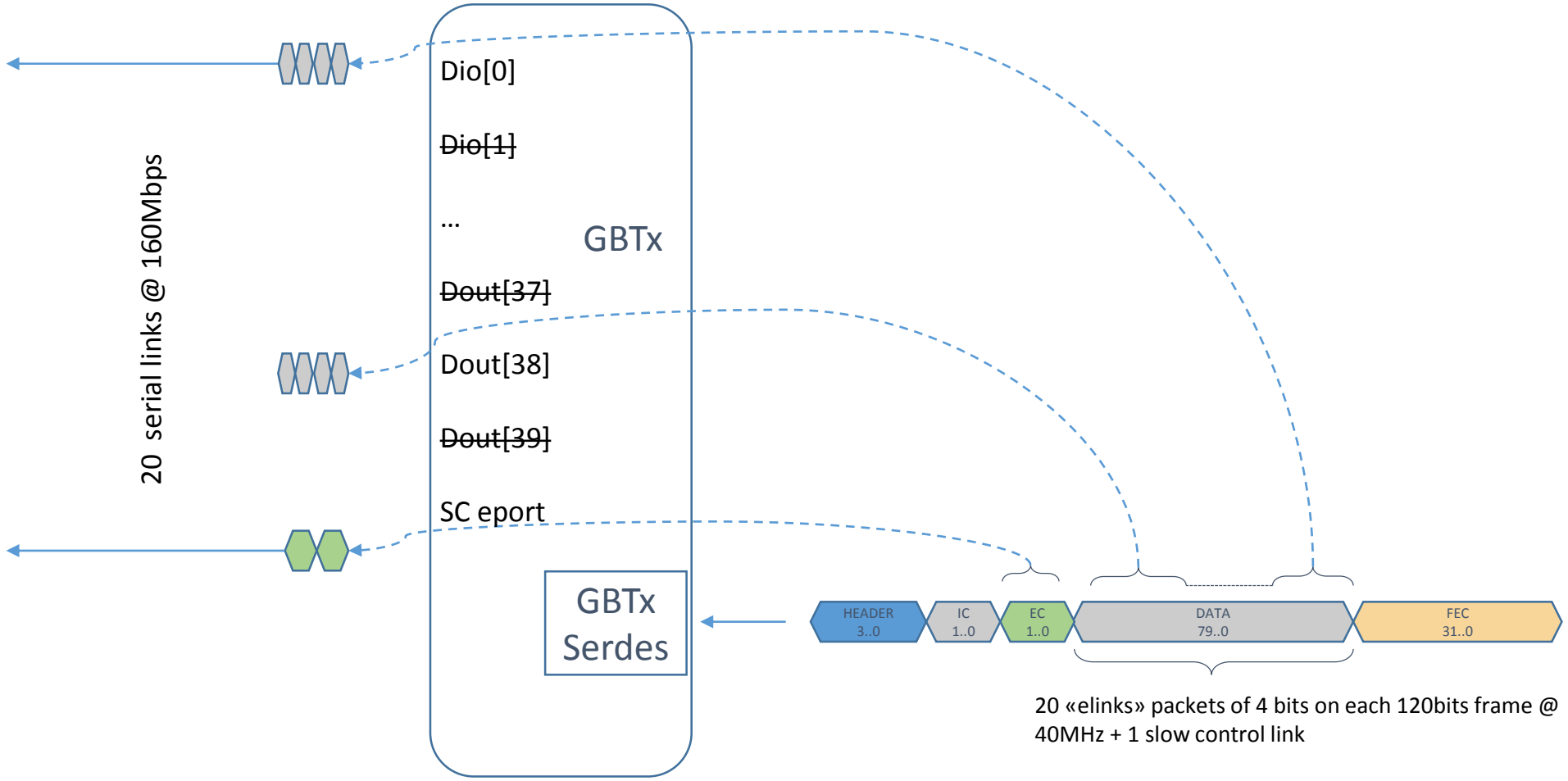


Downstream Elink mode «x2» (80Mb/s)

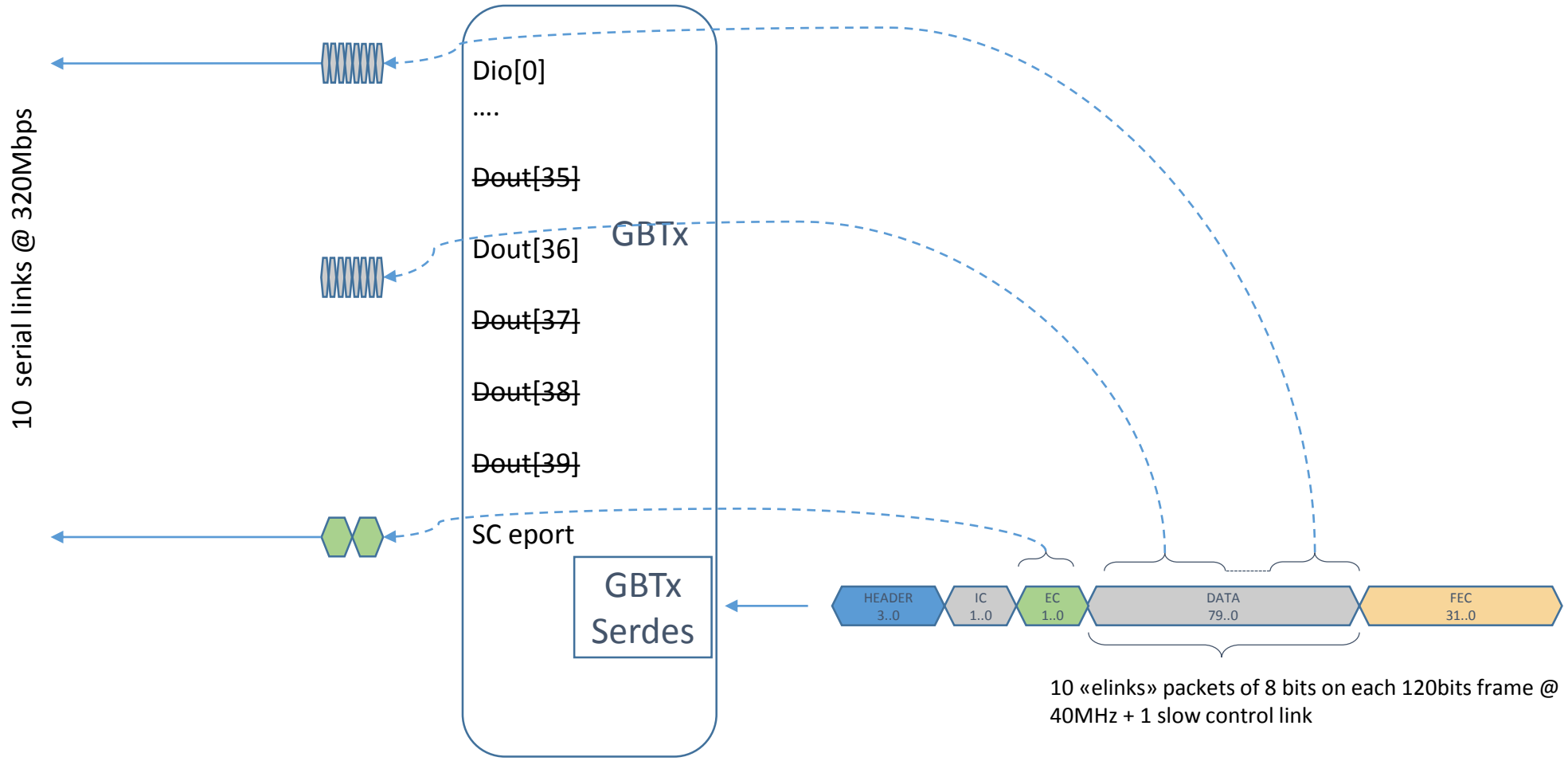


40 «elinks» packets of 2 bits on each 80bits frame @ 40MHz + 1 Slow Control link

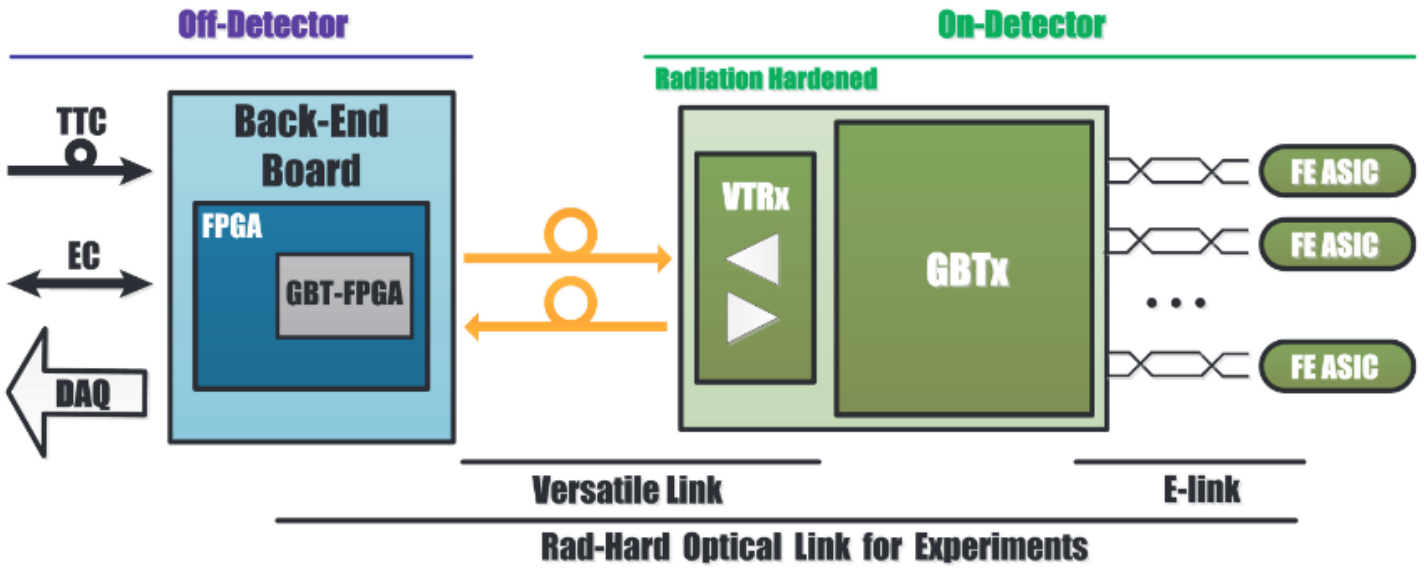
Downstream Elink mode «x4» (160Mb/s)



Downstream Elink mode «x4» (320Mb/s)

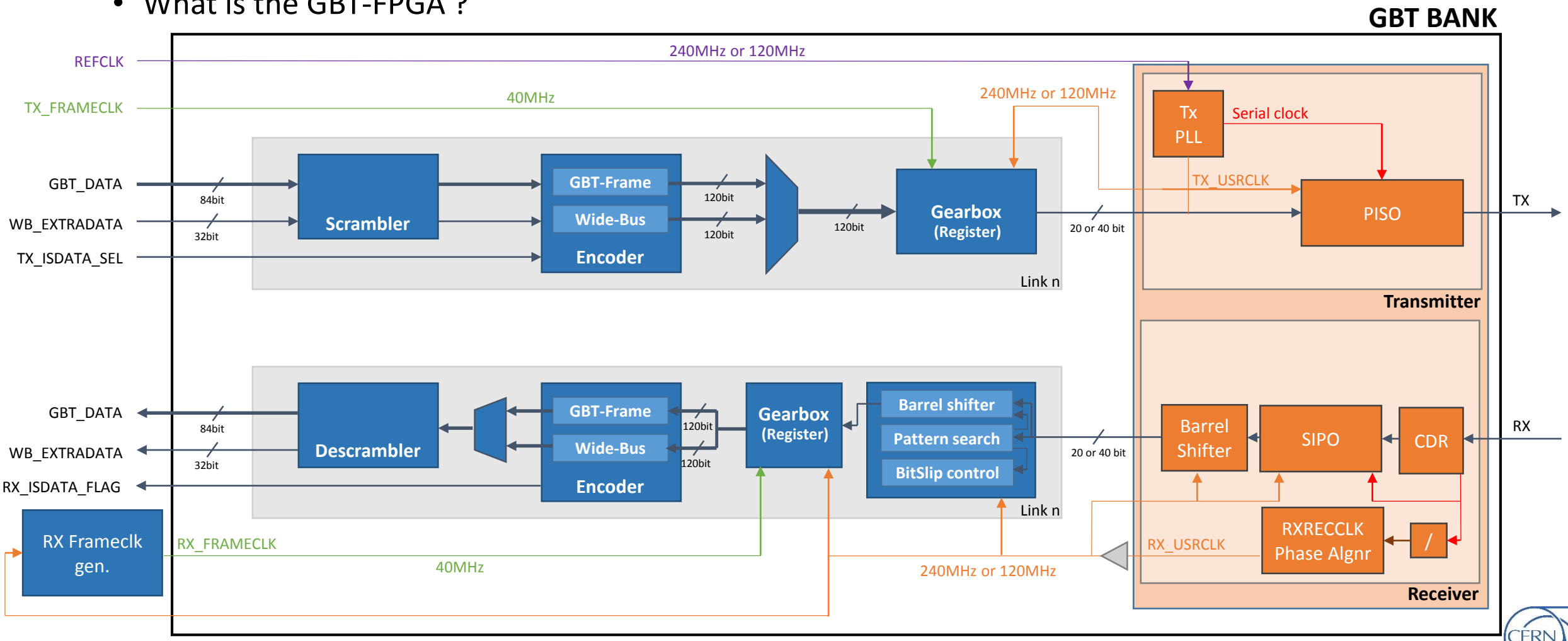


General overview



General overview

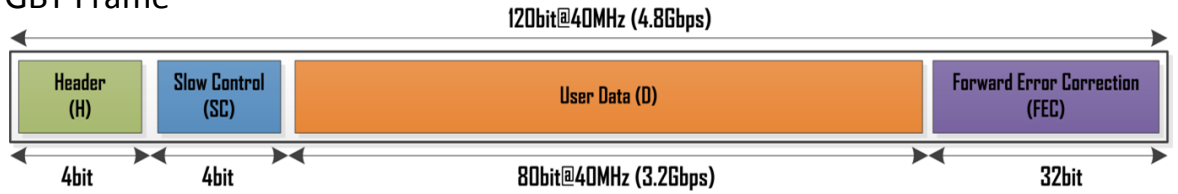
- What is the GBT-FPGA ?



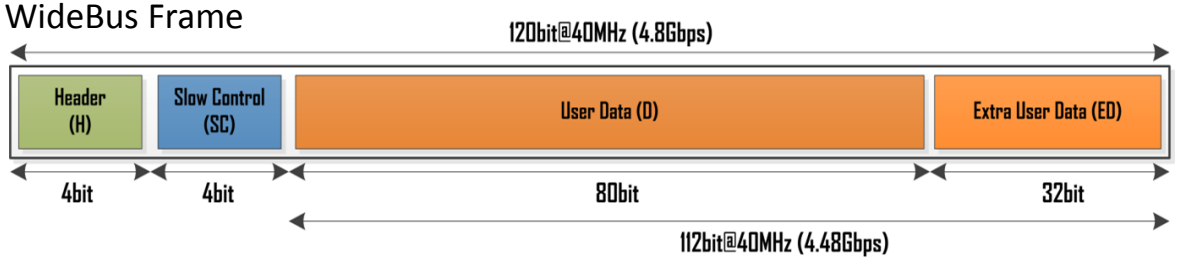
General overview

- What encoding are supported ?
  - GBT: based on Reed-Salomon
    - User data: 84bit
    - Can correct up to 4 consecutive symbols (4bit)
  - WideBus:
    - User data: 112bit
    - No Forward Error Correction (FEC)
- What are the mode supported ?
  - Standard mode
    - Latency: Non fixed, non deterministic and “high”
    - Easier to implement
  - Latency-optimized mode:
    - Latency: Fixed, deterministic and low
    - Complex to implement

GBT Frame



WideBus Frame



General overview

- What are the FPGAs characteristics?

FPGA		Mode		Encoding	
Manufacturer	**Device**	**Standard**	**Lat-optimized**	**GBT**	**WideBus**
Altera	Cyclone V	**Yes**	**No**	Tx/Rx	Tx/Rx
	Stratix V	**Yes**	**Yes**	Tx/Rx	Tx/Rx
	Arria V	*Yes*	*Yes*	*Tx/Rx*	*Tx/Rx*
	Arria 10	**Yes**	**Yes**	Tx/Rx	Tx/Rx
Xilinx	Virtex 6	**Yes**	**Yes**	Tx/Rx	Tx/Rx
	Virtex 7	**Yes**	**Yes**	Tx/Rx	Tx/Rx
	Kintex 7	**Yes**	**Yes**	Tx/Rx	Tx/Rx
	Kintex Ultrascale	*Yes*	*Yes*	*Tx/Rx*	*Tx/Rx*

SVN Repository architecture

- How to get the latest version of the GBT-FPGA IP ?
  - SVN: https://svn.cern.ch/reps/ph-ese/be/gbt_fpga/tags/<tag>
  - Latest tag: GBT_FPGA_4_0_0
  - Foreseen: GBT_FPGA_4_0_1 (July 2016)
    - Major bug fix: gbt_rx_decoder_gbtframe_chnsrch and gbt_rx_framealigner_pattsearch instances

- Are there any additional implementation ?
  - YES, different branches
  - SVN: https://svn.cern.ch/reps/ph-ese/be/gbt_fpga/branch/<branch>
  - E.g: GBT_FPGA_4_0_0_KC705_240MHZ

- Recommended tool: Tortoise SVN

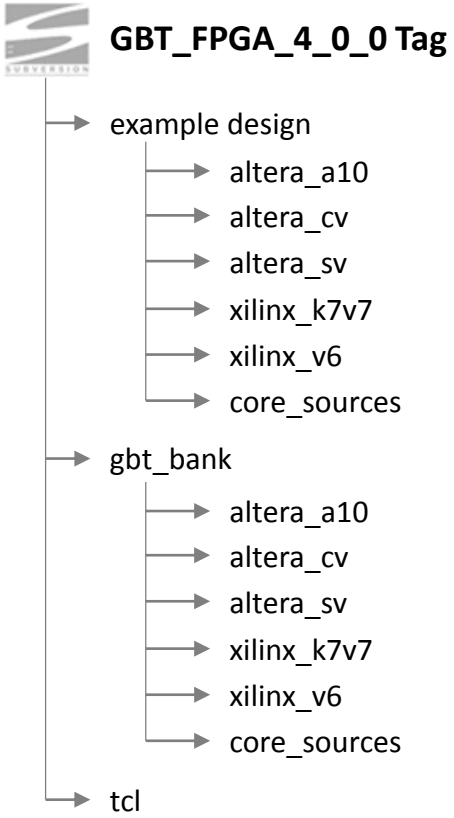
DEMO

SVN Repository architecture

- Example designs:
  - Arria 10 : PCIe40 and Altera GX development kit
  - Cyclone V : Altera GT development kit
  - Stratix V : AMC40 (with multi-links with Tx Lat-opt)
  - Kintex7 : FC7 and KC705
  - Virtex7 : VC707
  - Virtex 6 : ML605 and Glib

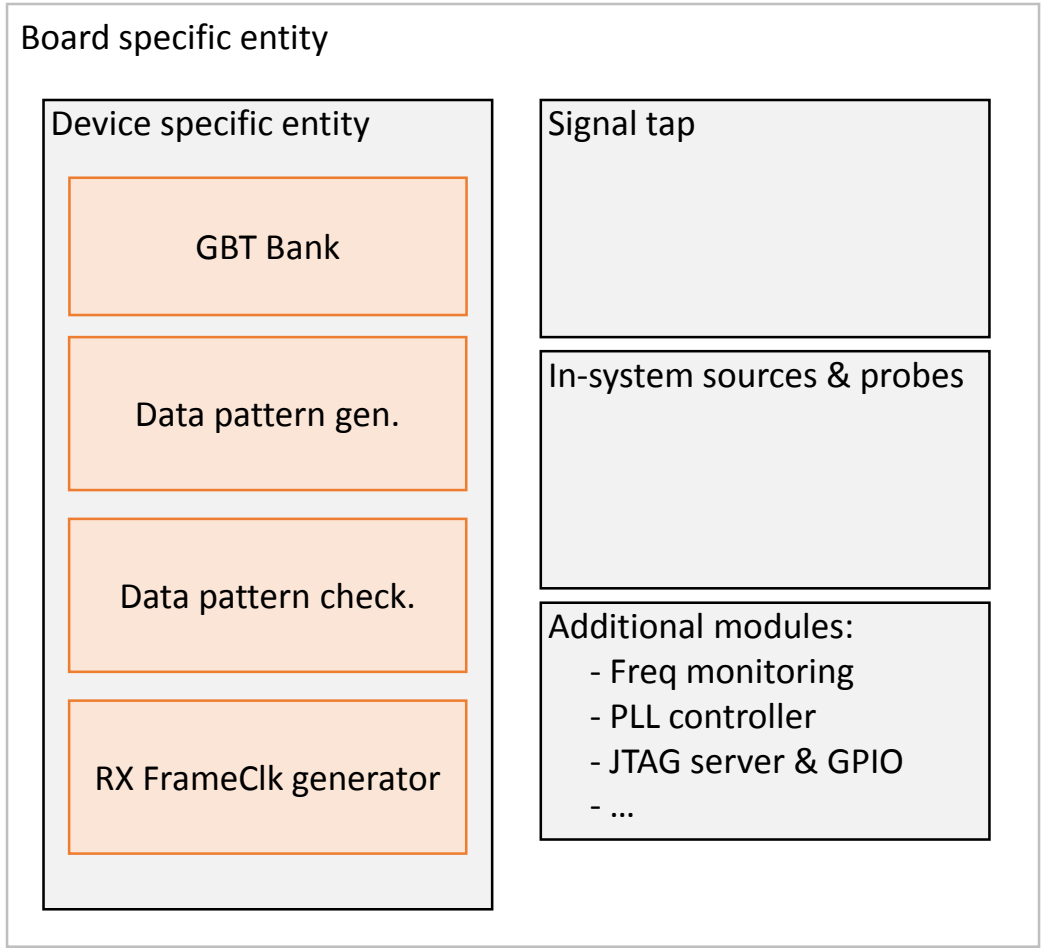
- IP Core:
  - core_sources: encoding/decoding modules
  - altera_* and xilinx_*: Device specific (e.g.: transceivers)

- TCL:
  - TCL script to source files
  - QSYS component description (Altera)



SVN Repository architecture

- What is the architecture of a reference design?



DEMO

SVN Repository architecture

- What modules are the modules included in the GBT-FPGA repository?
  - GBT-FPGA IP: Main module of the GBT-FPGA. It contains the encoder/decoder, transceivers …
  - Clocking modules:
    - RX Frameclk aligner
    - Clock divider
    - Clock frequency measurement
  - Data modules:
    - GBT Data pattern generator
    - GBT Data pattern checker
  - Latency measurement:
    - Data pattern matchflag
    - Latency measurement

- These modules will be used and detailed during the creation of the reference design

DEMO

Import the GBT-FPGA IP into my project

- How to use the TCL scripts?
  - Modification of the TCL file using an editor

```
#=================================================================================================#
#============================== Absolute Data Path Set By The User ===============================#
#=================================================================================================#

# Comment: The user has to provide the absolute data path to the root folder of the GBT-FPGA Core
#          source files.

set SOURCE_PATH ../../../..
set PROJECT_PATH .
```

  - Import it in the project: “source mytclfile.tcl”

DEMO