The Latency Study of GBT-FPGA on Xilinx FPGA

Hucheng Chen, Kai Chen, Francesco Lanni
on behalf of the FELIX developers

Brookhaven National Laboratory

kchen@bnl.gov

October 30, 2014

Table of Contents

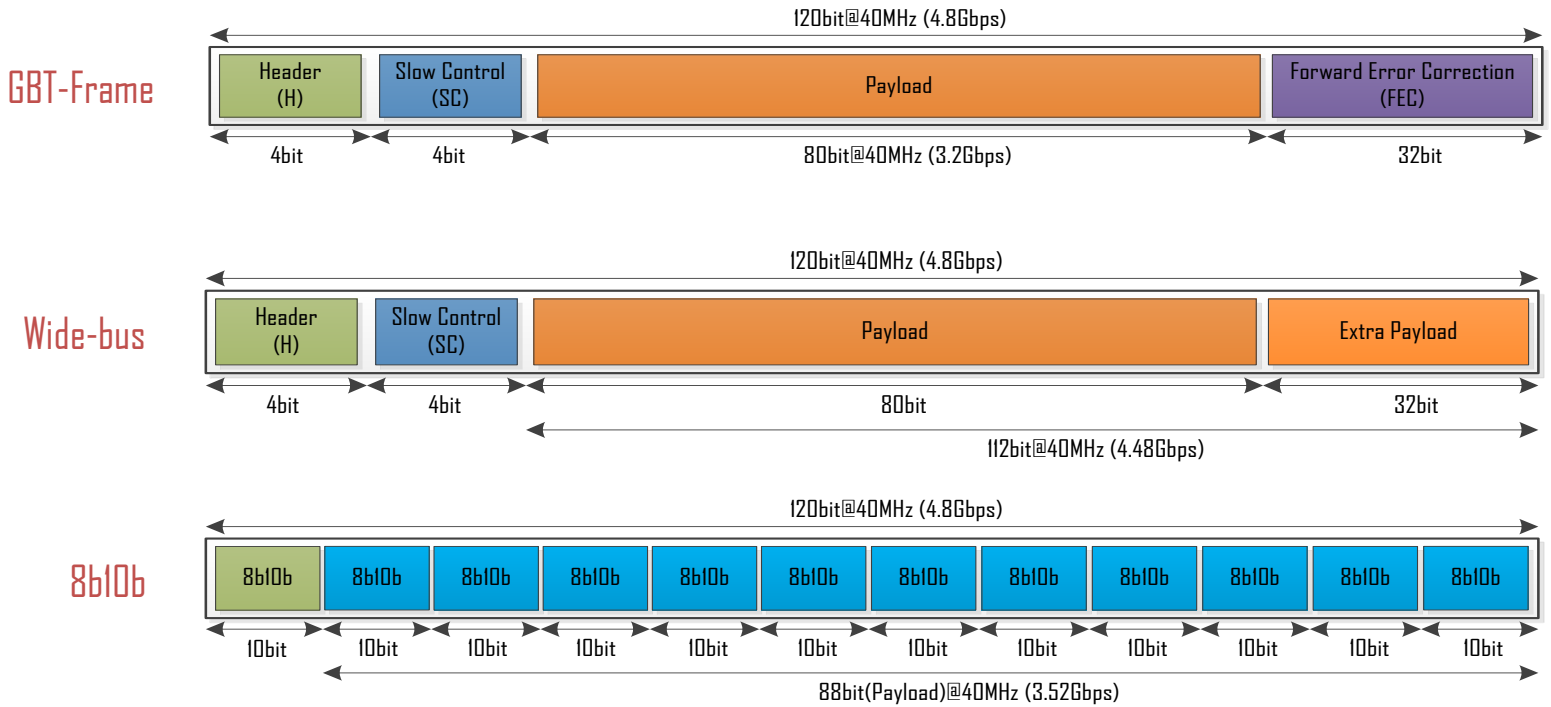
Three GBT-FPGA Encoding (from Manoel's report)

The context: "multiple needs" (3 of 5)

PH-ESE-BE

Multiple Configurations

- Encoding



Main page

Standard Mode & Latency Optimized Mode

If latency is not an issue, the **Standard version** is strongly recommended.

Table 1: Standard VS Latency-Optimized

	Standard	Latency-Optimized
Latency	Non Fixed, Higher, Non Deterministic	Fixed, Low, Deterministic
Logic Resources Utilization	Low	Low
Clocking Resources Utilization	Low	High
Clock Domain Crossing	Don't Care	Critical
Implementation	Simple	Complex

Table from GBT-FPGA user guide

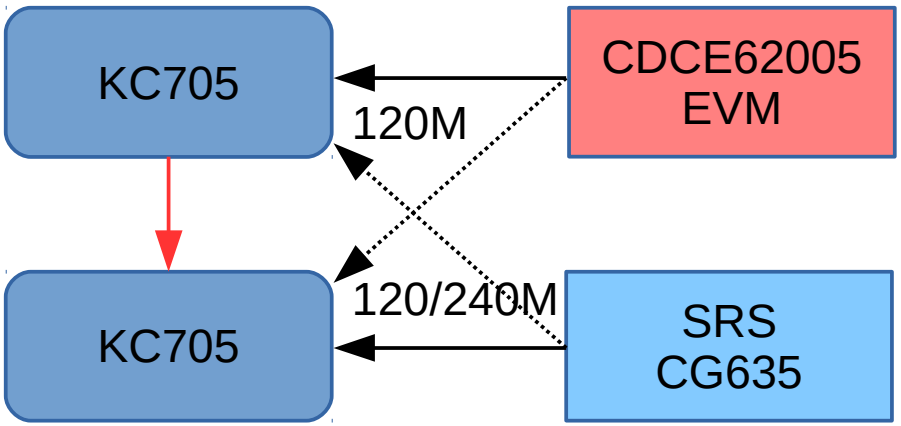
7 Series GTX Low-Latency Configuration

RXSLIDE_MODE	PMA
RX_DATA_WIDTH	40/20
TX_DATA_WIDTH	40/20
RXBUF_EN	FALSE
TXBUF_EN	FALSE
RXGEARBOX_EN	FALSE
TXGEARBOX_EN	FALSE
TX8B10BEN	tied_to_ground_i
RX8B10BEN	tied_to_ground_i
RXDDIEN	tied_to_vcc_i
RXCOMMADETEN	tied_to_ground_i [1]
RXMCOMMAALIGNEN	tied_to_ground_i
RXPCOMMAALIGNEN	tied_to_ground_i
RX_XCLK_SEL	RXUSR
TX_XCLK_SEL	TXUSR
RXOUTCLKSEL	010
TXOUTCLKSEL	011

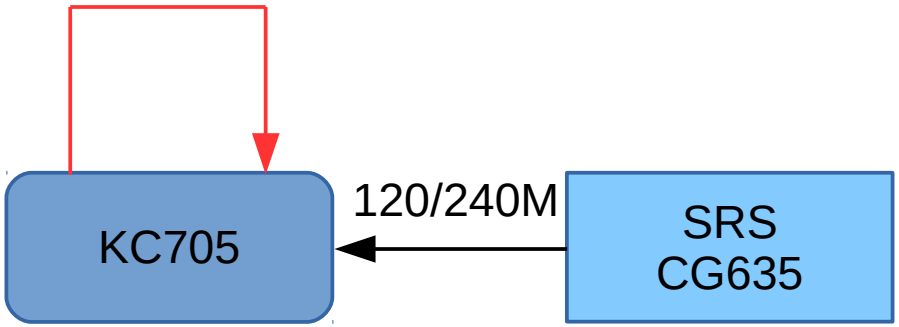
[1] different from the GBT-FPGA Core, save one WordCLK (USRCLK for XILINX GTX, 120M/240M) cycle. It's the same for Virtex-6 FPGA.

Table of Contents

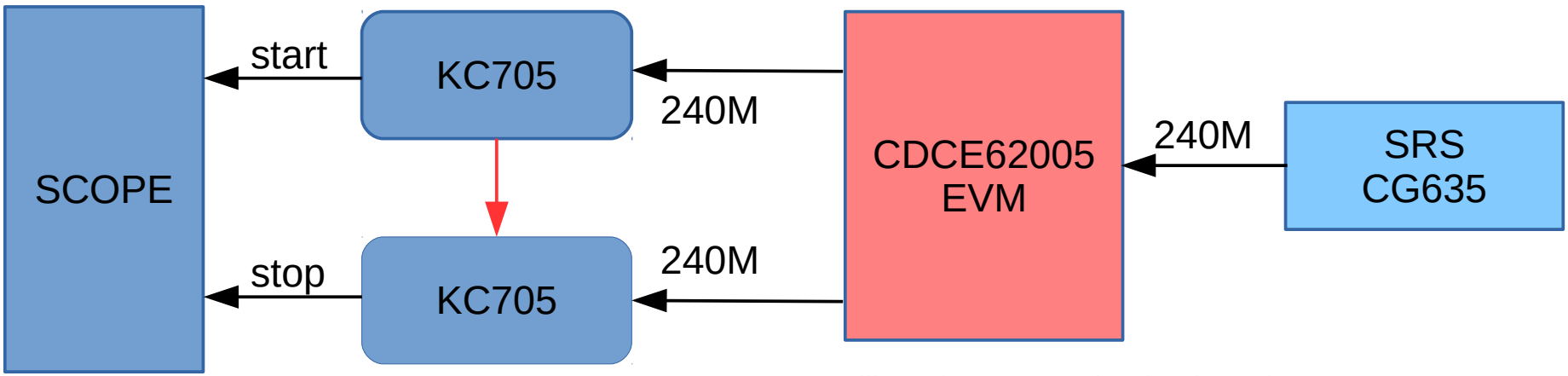
Test platform



Different reference clock sources are used by two KC705.
It's used to verify the latency is fixed.



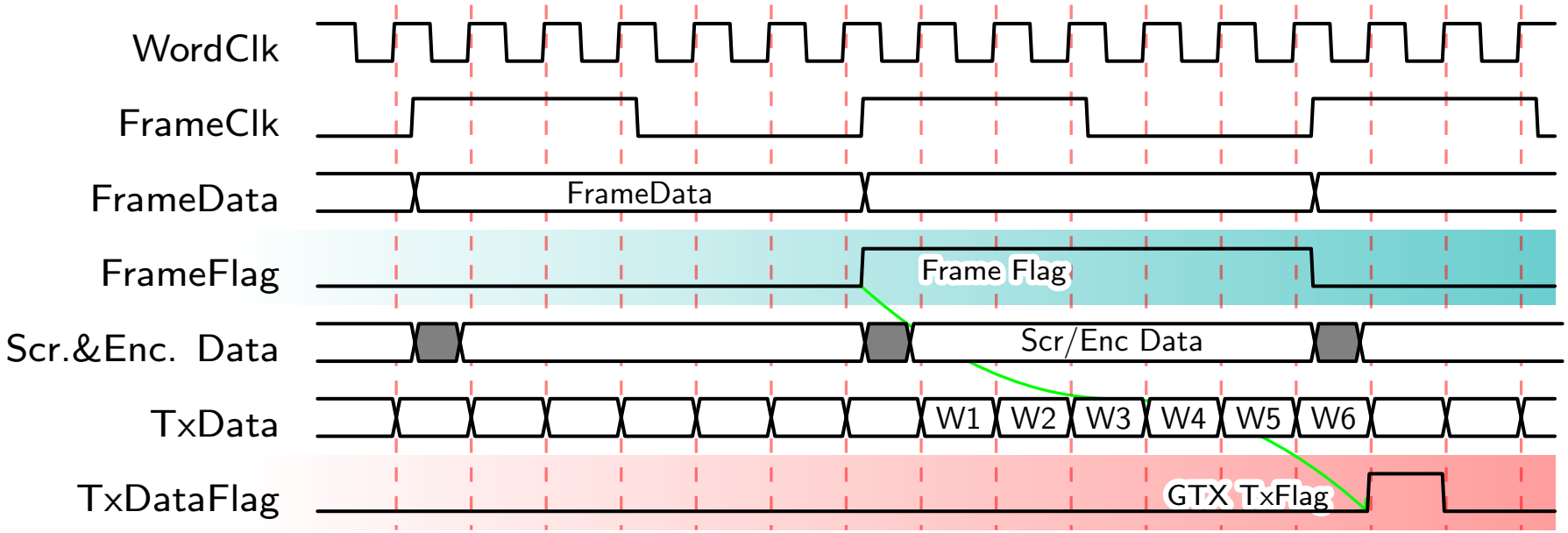
One KC705: loopback test.
Can be used to test the latency value.



One 240MHz reference clock source is used by two KC705.

KC705: Xilinx Kintex-7 Evaluation board.
SRS CG635: Clock generator.
CDCE62005 EVM: clock evaluation module with on-board 25 MHz clock source.

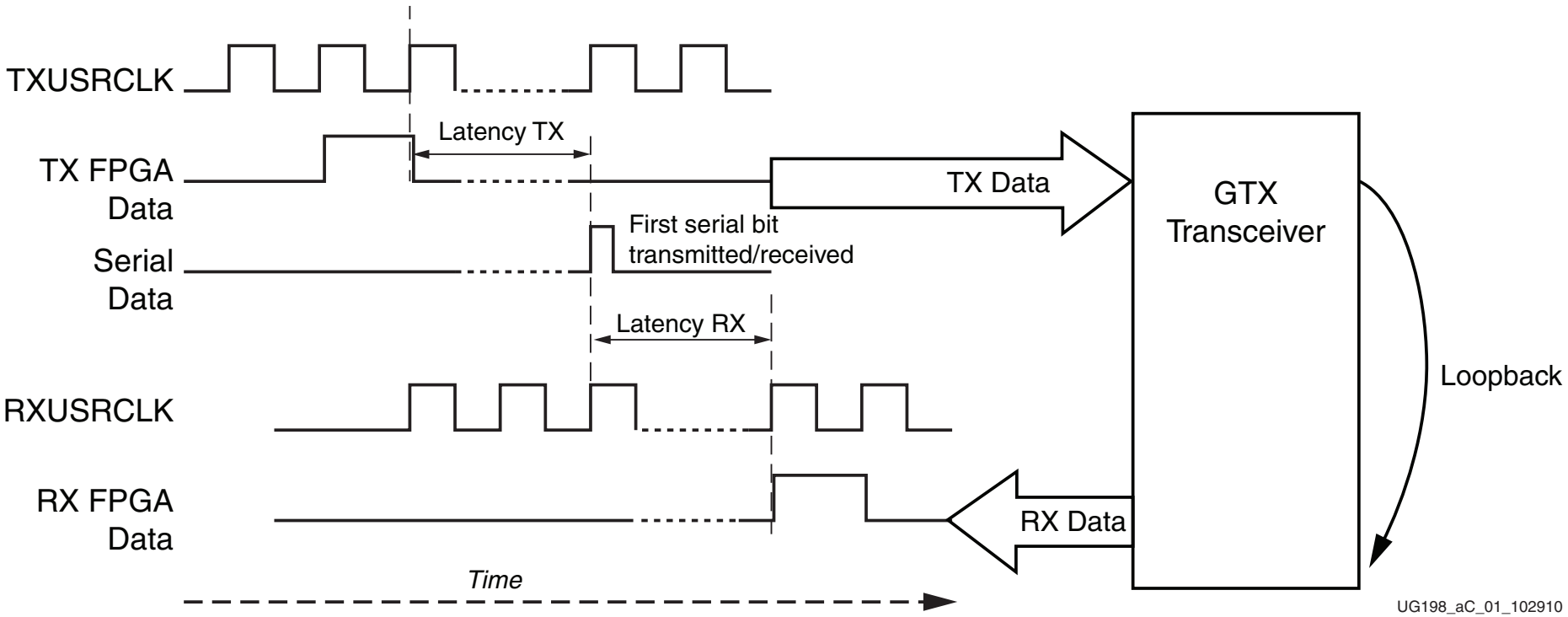
Definition of the GBT TX Latency



Latency of GBT-FPGA TX encoding

$T_F <$ Latency $< T_F + T_W$ (T_F is the FrameClk period, which is about 25 ns; T_W is the period of the WordClk, it's 1/120M or 1/240M.) Latency is from the falling edge of the TX FrameData to the falling edge of the last WordData sent to GTX.

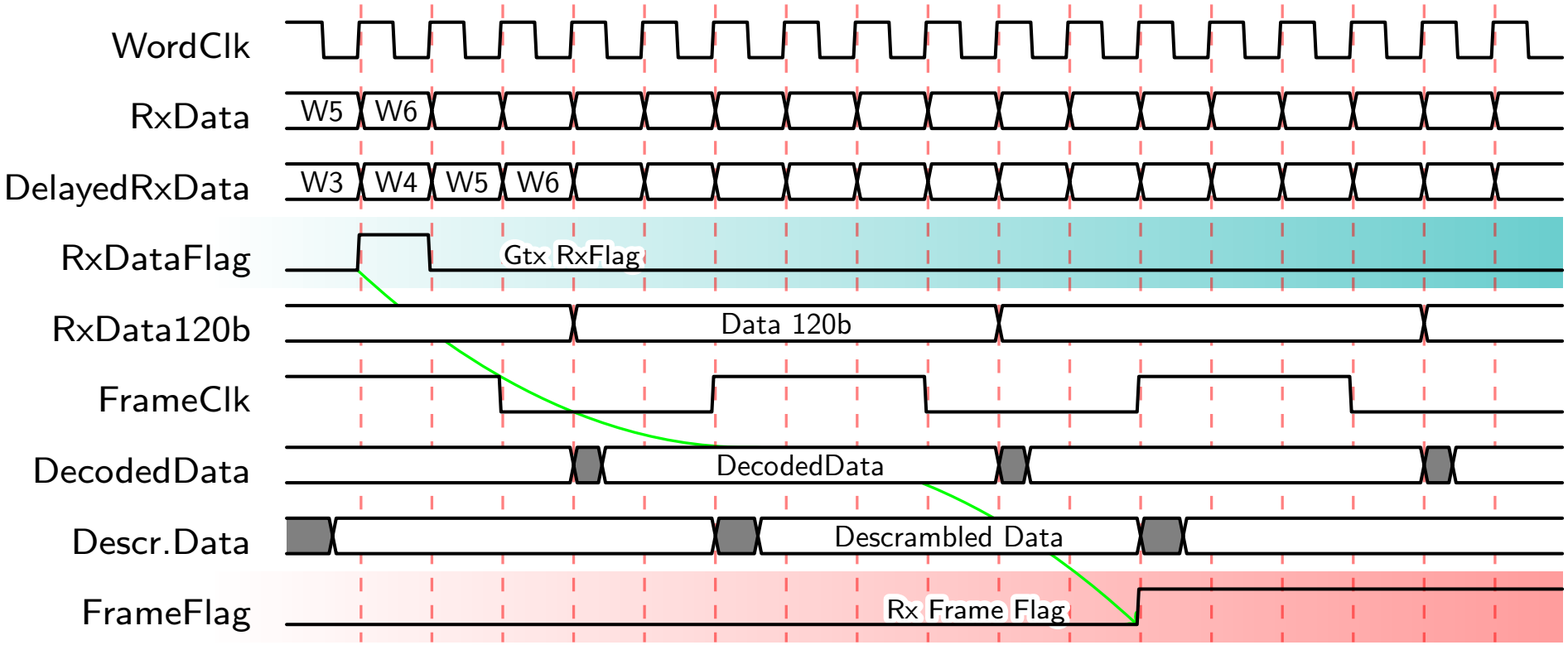
Definition of the GTX Latency from Transmitter to Receiver



The GTX transceiver latency: from the falling edge of the last TX WordData, to the rising edge of the last RX WordData.[2]

[2]The figure is from V6's GTX User Guide.

Definition of the GBT RX Latency



Latency of the GBT-FPGA RX encoding

$T_F + 5T_W$, (T_F is period of FrameClk (40M), T_W is period of WordClk (120M/240M))

Latency is defined as: from the rising edge of the last RX WordData from GTX, to the falling edge of the RX FrameData.

Possible Optimization to decrease latency

- GBT TX: original range is $[T_F, T_F + T_W]$
  - Start the scrambling and encoding $T_F - T_W$ ealier, using the clock domain of WordClk, or the FrameClk latched by WordClk.
  - Can decrease to $[T_W, 2T_W]$, at the most. Some margin should be kept for the encoding/scrambling, according to the requirement of the real design. If the fixed latency is not so important, then more margin is better.
- FPGA GTX RX side latency:
  Set RXCOMMADETEN to be '0', save one T_W.

Possible Optimization to decrease latency

- GBT RX: original latency is: $T_F + 5T_W$
  - Don't use the DelayedRxData as Gearbox input, directly use RxData, save $2T_W$, to $T_F + 3T_W$;
  - If keep using FrameClk for decoding/descrambler, the FrameClk can be left shifted for one cycle, save $1T_W$, to $T_F + 2T_W$;
  - If use WordClk or Latched FrameClk to decode/descramble, can decrease to $3T_W$;
  - Depends on the decoding/descrambler speed, sometimes we don't need to latch the last RX 20b/40b Word data. We directly decode the one cycle unchanged data(RX_DATA & the latched(W1-W5)), then the latency can decrease to $2T_W$. The gearbox logic can also work on falling edge of USR2CLK, giving more time for decoding/descrambler;
  - A $2T_W$ latency can be obtained. For different GBT encoding, the TX/RX logic can be adjusted carefully to obtain a low latency, if the latency is very critical. But some margin should be kept.

8B/10B encoding/decoding

- In order to realize the 8B/10B encoding with as few modifications to the GBT-FPGA core as possible, an external LUT based 8B/10B encoder/decoder is used now. It supports both synchronous or asynchronous mode.
- The 120b Frame data include one k-char 28.5. The receiver side logic doesn't check the decoded k-char for alignment, it still checks the 4 MSB bit of the frame data, this minimize the modification to the GBT-FPGA core.
- The test shows that the latency of the 8B/10B decoding/encoding is slightly bigger than 1/240MHz(WordClk), when not using any floorplanning constraint.
- Plan to use the encoder/decoder inside GTX module, but will not use the comma detector in the GTX module.
- *8B/10B coding scheme is not a main focus of FELIX development*

Table of Contents

Results for 240M x 20bit, KC705

	GBT TX	GTX	GBT RX	Measured Total
GBT-FPGA encoding	$T_F + Phase$[3]		$T_F + 5T_W$	
Modified[4] GTX sim.		19.33+35.63=54.96		
Xilinx Table Calc.[5]		23.70+35.52=59.22		
Modified GTX test	25+1.7	57.06	45.83	129.6
Optimized encoding	$T_W + Phase$		$2T_W$	
Test results	5.87	57.00	8.33	71.2

Note

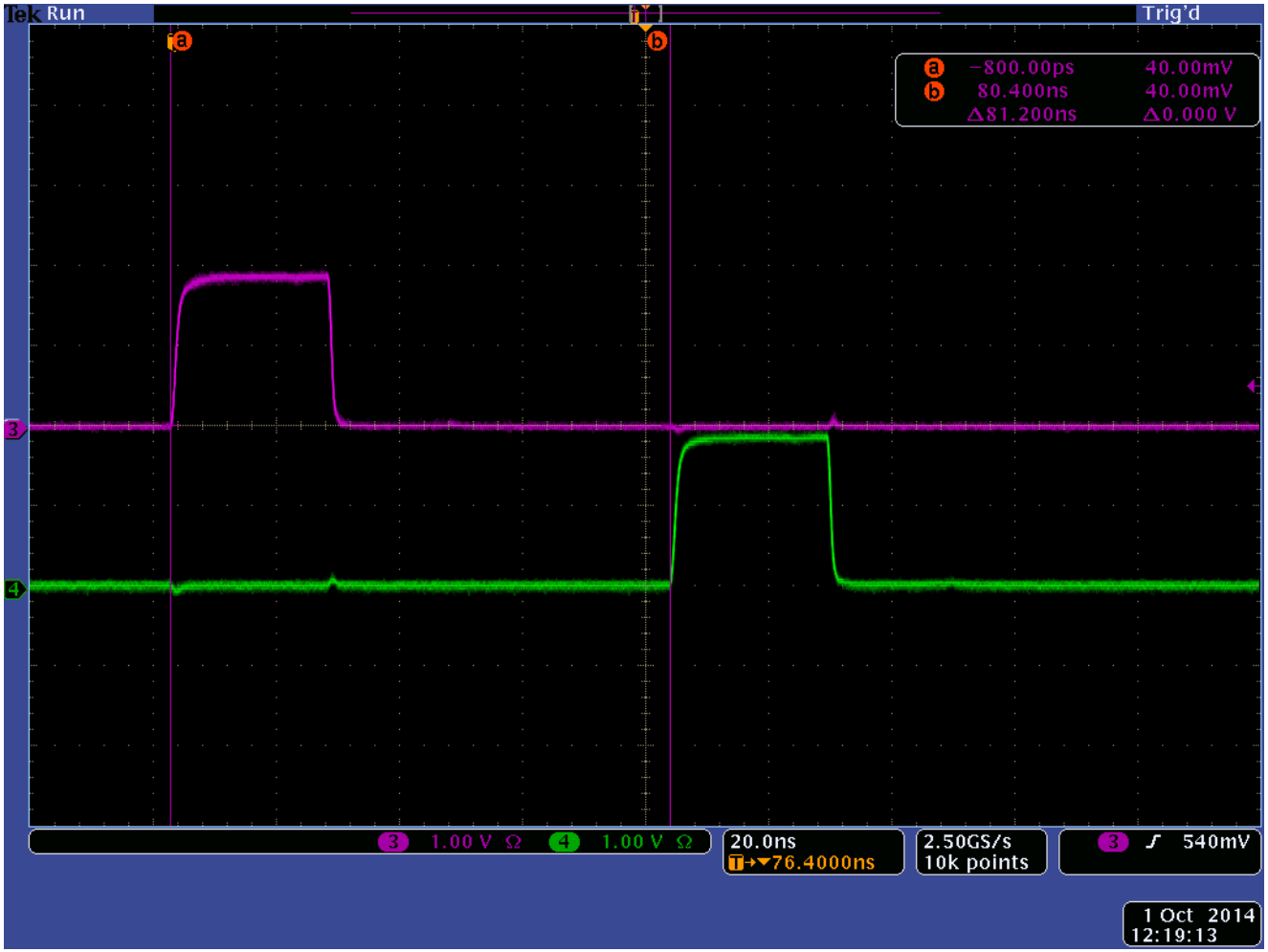
For all the three GBT-FPGA encodings, a Latency smaller than 3 bunch crossing can be obtained. If more time margin is kept, it will be a few T_W larger than the latency shown above.

[3] $0 < Phase < T_W$

[4] The comma detector is disabled.

[5] `http://www.xilinx.com/support/answers/42662.html`

Screenshort for 240M x 20bit



A 2 meters fiber is used to connect the transmitter and receiver, so the latency is about 81.2-10=71.2 ns.

Results for 120M x 40bit, KC705

Test	GBT TX	GTX	GBT RX	Measured Total
GBT FPGA encoding	$T_F + Phase$		$T_F + 5T_W$	
GTX simulation		36.75+77.51=114.26		
Xilinx Table Calc.		54.47+68.44=122.91		
GBT example test	25+4	118.33	66.67	214
modified[6] GTX sim.		36.75+68.55=105.30		
Xilinx Table Calc.		46.14+68.44=114.58		
GTX modified test	25+4	110	66.67	205.67
Optimized encoding	$T_W + Phase$		$2T_W$	
Estimated results	12.33	110	16.67	139

Note

The 120M x 40bit has almost twice latency, compared to 240M x 20bit. The GTX latency is calculated results, others are measured results.

[6] The comma detector is disabled.

Results for 240M x 20bit, Virtex-6

Test	GBT TX	GTX	GBT RX	Total
GBT-FPGA encoding	$T_F + Phase$		$T_F + 5T_W$	
Estimated from equations.	29.17	18.7+36.6=55.3	45.83	130.3[7]
GBT-FPGA manual's results	29.2	18.7+28.2=46.9	54.2?[8]	130.3
GTX sim. after GTX RX optimized[9]		18.44+30.16=48.60		
Xilinx Table Calc.		20.83+30.63=51.46		
Est. after GTX RX optimized	29.17	18.7+32.43=51.33	45.83	126.13
Est. for optimized Encoding	8.33	18.7+32.43=51.33	8.33	67.8

Note

- The test is not done yet, the results listed above is based on calculation and simulation.
- Simulation shows that: if RXCOMMADETEN is disabled, latency can decrease one T_W.
- Latency of Virtex 6 GTX is slightly smaller than Kintex 7.

[7] It's assumed to be the same with GBT-FPGA manual.
[8] $2T_W$ bigger than $T_F + 5T_W$.
[9] Disable the comma detector.

Table of Contents

Summary

- Some optimization can always be used for future projects.
- Some optimization need to be adjusted according the requirement of the application.
  - FPGA logic speed (routing, encoding/decoding), must keep some timing margin.
  - The TX FrameClk frequency, and the FrameData width.
  - Where the RX data will go, and the FrameData width.
- External 8B/10B encoding/decoding is used now (LUT based, can run at synchronous mode or asynchronous mode). Non-LUT based method is not tested. The 8B/10B encoding/decoding in the GTX will be used in the future, with external alignment logic.
- More parameters will be added to the top-level configuration file, to support different kinds of optimization.

Summary

- Several things happened during the testing:
  - Top level TX reset will reset RX logic.
  - The GTX RX buffer is bypassed, the default example of GBT-FPGA use auto alignment function provided by Xilinx. The manual mode is also provided[10], it was tested.
    - RXSLIDE doesn't work for an odd calculated bit_slip, this is not a bug, it is decided by the GTX architecture (the series data is DDR, so the minimum shift step is 2UI (Unit Interval)). A 1 UI shifted WordCLK(USRCLK) can solve it, but will need more clock resources. We simply keep sending RX reset, until an even bit_slip number is obtained.
    - when the calculated bit_slip number is 8, the RXSLIDE doesn't work. Resetting GTX RX module will solve it.
  - Sometimes phase of the generated RX FrameClk will change from run to run, with a step of T_W. A phase check module is added to obtain a stable phase, it checks the phase using a flag signal generated by the Gearbox.

[10] Page 240 of `http://www.xilinx.com/support/documentation/user_guides/ug476_7Series_Transceivers.pdf`

Table of Contents

Outlook

- Further development, simulation and test will be based on GTH transceivers of Xilinx 7 series FPGA.
  - HiTech Global HTG-710 will be the main development platform
- Latency optimization will be implemented on multiple transceivers (e.g. 24 pairs etc.) to verify the resources constraint.
- Will look into different link speed (e.g. 9.6 Gb/s etc.).
- Long term stability test will be performed after latency optimization.