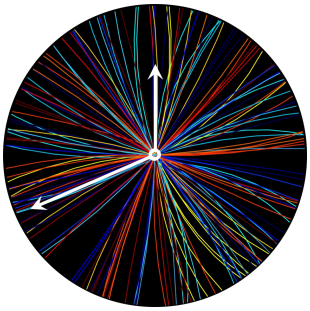

EDMS Document Number
XXXXXXX

HPTD project URL
https://espace.cern.ch/HighPrecisionTiming

Date: 25 March 2020
Revision No. 1.3

Reference Note

TCLink

A Timing Compensated Link for the CERN HL-LHC experiments

Abstract

This short reference note describes the usage of the TCLink IP core, used for FPGA-based timing distribution links which require fine phase measurement and compensation.**TCLink is a low-resources protocol-agnostic soft FPGA IP, which can work in principle for any fixed-phase high-speed link requiring a high phase stability in the long-term**. The user has the freedom to enable/disable the compensation scheme and choose the loop parameters (helped by a Python script). Example designs targetting the Xilinx **VCU118** and **KCU105** FPGA evaluation boards are made available.

Prepared by	***Checked by***	***Approved by***
E. B. S. Mendes CERN/EP-ESE 1211 Geneva 23 Switzerland *eduardo.brandao.de.souza.mendes@cern.ch*	**S. Baron** CERN/EP-ESE 1211 Geneva 23 Switzerland *sophie.baron@cern.ch*	**-**

Document History

Rev. No	*Date*	*Pages*	*Description of Changes*
0.1	11 November 2019	*All*	First version
1.3	25 March 2020	*All*	Generate TCLink heterodyne clock with internal MMCM

Contents

1 Technique overview

The **TCLink** is a protocol-agnostic FPGA core envisaged to mitigate long-term phase variations in high-speed optical links. The concept is to have monitoring and picosecond-level online adjustment capabilities which can be tailored by the user to best fit his/her own convenience and application requirements. We refer to a timing compensated link when the phase of the recovered clock is stabilized in a controlled-fashion to minimize variations arising from environmental changes in the experiments. The principle of phase monitoring/compensation is to rely on a roundtrip variation measurement to estimate the downlink phase variation as shown in figure 1.

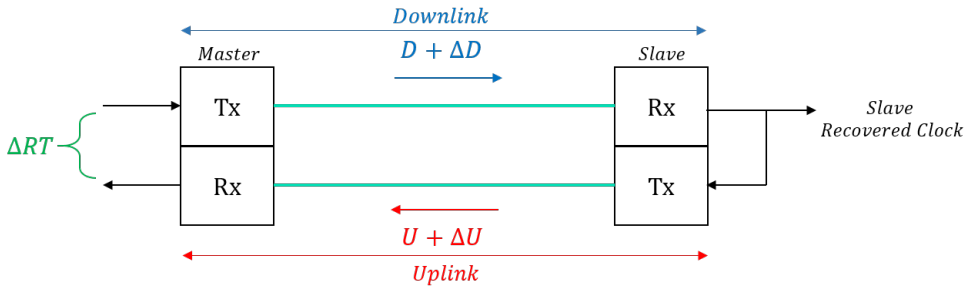

Figure 1: TCLink concept

In order to implement a timing compensated link, three main requirements are needed which can be observed in figure 2.

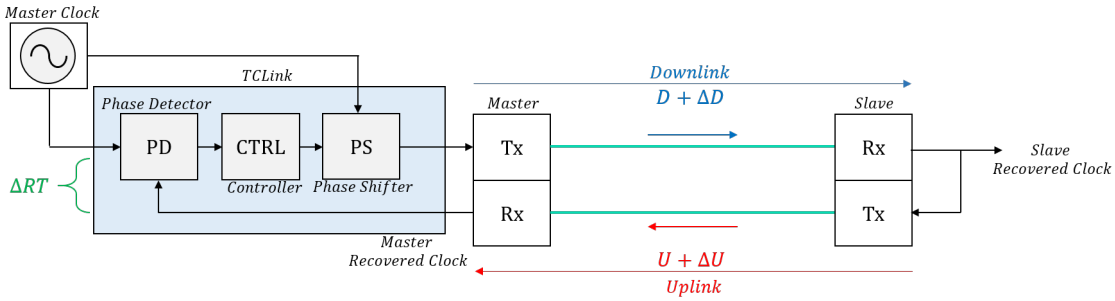

Figure 2: TCLink simplified block diagram

- A synchronous bidirectional link is established. The slave re-uses its recovered clock for the uplink transmission.
- A high-accuracy roundtrip phase measurement capability is available on the master side. For the compensation, a phase shifter is also required.
- An underlying hypothesis relating the downlink phase variations to the total roundtrip phase variations must be made ($\Delta D = \alpha \times \Delta RT$).

The **TCLink** core offers to the user the capability of changing all control parameters (correction bandwidth, phase-detector bin-size, phase-detector averaging, α coefficient, ...). For more information on how to configure the core, read sections 2 and 3. Example designs are made available and more information can be found in section 4 and in the files **quick_start_guide_vcu118.pdf** and **quick_start_guide_kcu105.pdf**.

2 Core Overview

This section is intended to give an overview of the **TCLink** core.

*It is highly recommended to use the default configuration for all ports. In case this is not possible, it is recommended to read the section 3 to learn more on how to configure **TCLink** and validate the new configuration in the example design. Users are very welcome to contact the HPTD team in case of questions.*

Architecture

The **TCLink** architecture is shown in figure 3. The **TCLink** core is composed by the phase-detector and the controller (which are device and protocol agnostic). The phase-shifter used for this implementation is Xilinx Ultrascale dependent. A brief description os each of those blocks is given here below:

- **Phase-detector:** The phase-detector used is the **Digital Dual Mixer Time Difference (DDMTD)** core created in the **CERN White-Rabbit** project (`https://ohwr.org/project/white-rabbit`). The **DDMTD** relies on a heterodyne mixing in order to perform a phase-measurement. Therefore, a third clock with a small frequency offset is necessary for the phase-measurement, it is recommended to use a clock from an external PLL for the mixing clock (the example design uses an internal MMCM for those purposes to ease the usage for a first approach with the core). It can have a resolution of $o(ps)$.
- **Controller:** Digital controller using sigma-delta modulation and capable of mirrorring the control plant in order to emulate different α coefficients.
- **Phase-shifter:** The phase-shifter used is the **HPTD IP** core created by the **CERN HPTD** project (`https://gitlab.cern.ch/HPTD/tx_phase_aligner`). It can have a resolution of $o(ps)$.

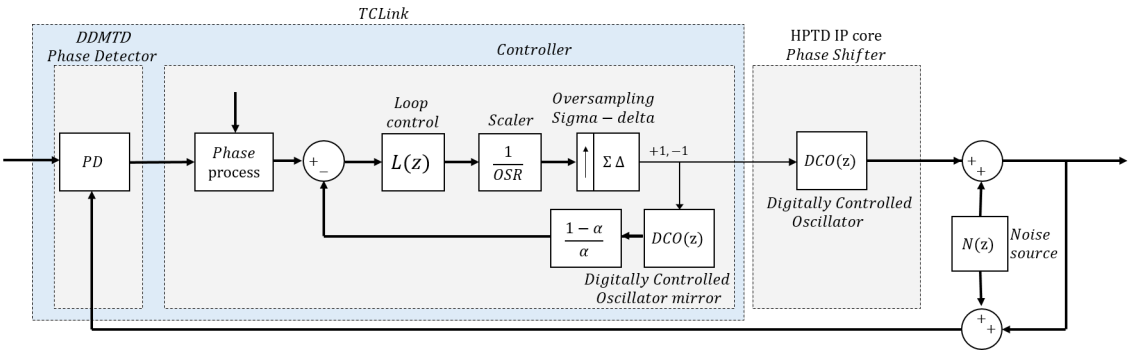

Figure 3: TCLink high-level block-diagram

The purpose of the phase-process block inside the controller is to ensure that no big phase-jumps are present when the compensation is enabled. The user shall measure the phase-offset when the compensation is disabled and this offset is removed by this block. This requires a user intervention after the first reset (and before enabling compensation). More details are given in the next subsection.

Ports description and default values

The ports of the **TCLink** core are described in table 1 together with the default values for a $10.24Gpbs$ protocol. If **any default value** has to be changed or a different line-rate protocol is used, it is recommended to read the section 3.
To ensure that no big-jumps are present when the compensation is enabled (port *close_loop_i*), the offset phase value has to be measured and removed. This shall be done only once after the first reset by reading one value of the port *error_controller_o* and writing it to the port *offset_error_i*.

obs: $DDMTD\_UNIT[ps] = 10^{12} \times \frac{FREQ(clk\_tx\_i) - FREQ(clk\_offset\_i)}{FREQ(clk\_tx\_i) \times FREQ(clk\_offset\_i)}$ where $FREQ(clk)$ denotes the frequency of clk in Hz

Port	Dir	Clk Domain	Description	Functionality
clk_sys_i	in	—	System clock. Connect to a free-running source **Default is 125MHz**	general
tx_ready_i	in	async	Used internally as core reset Keep to zero while MGT tx not ready	general
rx_ready_i	in	async	Used internally as core reset Keep to zero while Rx framing is not locked	general
clk_tx_i	in	—	Tx user clock - transceiver txusrclk2 **Default is 320MHz**	phase detector
clk_rx_i	in	—	Rx user clock - transceiver rxusrclk2 **Default is 320MHz**	phase detector
clk_offset_i	in	—	Heterodyne clock for phase measurement **Default is 319.376MHz (coming from same PLL as transceiver Tx reference clock)**	phase detector
metastability_deglitch_i[15::0]	in	clk_sys_i	Phase detector metastability window **Default is 0x0052**	phase detector cfg
phase_detector_navg_i[11::0]	in	clk_sys_i	Phase detector averaging number **Default is 0x040**	phase detector cfg
phase_detector_o[31::0]	out	clk_sys_i	Phase detector response Signed complement 2 number. Conversion to ps: DDMTD_UNIT/phase_detector_navg_i	phase detector stat
modulo_carrier_period_i[47::0]	in	clk_sys_i	Tx/Rx user clock period **Default is 0x00007ff48348**	controller cfg
offset_error_i[47::0]	in	clk_sys_i	Error-offset for phase-control Recommended to freeze with an initial value read by port *error_controller_o*	controller cfg
error_controller_o[47::0]	out	clk_sys_i	Error-signal for controller Signed complement 2 number. Conversion to ps: (2**-16)*DDMTD_UNIT/phase_detector_navg_i	controller stat
close_loop_i	in	clk_sys_i	Close TCLink loop (enables compensation)	controller cfg
Aie_i[3::0]	in	clk_sys_i	Loop controller integral coefficient scaled **Default is 0x0**	controller cfg
Aie_enable_i	in	clk_sys_i	Enables Loop controller integral coefficient **Default is 0b0**	controller cfg
Ape_i[3::0]	in	clk_sys_i	Loop controller proportional coefficient scaled **Default is 0xe**	controller cfg
sigma_delta_clk_div_i[15::0]	in	clk_sys_i	Clock-divider for sigma-delta modulation **Default is 0x0197**	controller cfg
enable_mirror_i	in	clk_sys_i	Enables mirror compensation scheme **Default is 0b1**	controller cfg
Adco_i[47::0]	in	clk_sys_i	Mirror DCO coefficient scaled **Default is 0x0000000ffe90**	controller cfg
master_rx_slide_mode_i	in	clk_sys_i	Master Rx MGT slide mode (0=PMA, 1=PCS)	Master Rx cfg
master_rx_ui_period_i[47::0]	in	clk_sys_i	UI period of master Rx link **Default is 0x000003ffa41a**	Master Rx cfg
master_rx_slide_clk_i	in	—	Clock used by master Rx for frame alignment. Typically is the same as clk_rx_i	Master Rx cfg
master_mgt_rx_ready_i	in	clk_sys_i	MGT Rx ready	Master Rx cfg
master_rx_slide_i	in	master_rx_slide_clk_i	Rx slide bit issued by master frame aligner	Master Rx cfg
phase_acc_o[15::0]	out	clk_sys_i	DCO phase accumulated Signed complement 2 number. Conversion to ps: PI_UNIT. Check your transceiver user guide to see the PI_UNIT.	DCO interface
operation_error_o	out	clk_sys_i	DCO communication error (a *strobe_o* was issued and no *done_i* was received)	DCO interface
strobe_o	out	clk_sys_i	DCO shift phase	DCO interface
inc_ndec_o	out	clk_sys_i	DCO increment (1) or decrement (0)	DCO interface
phase_step_o	out	clk_sys_i	DCO phase step	DCO interface
done_i	in	clk_sys_i	DCO shift was performed	DCO interface
debug_tester_enable_stimulis_i	in	clk_sys_i	Enable TCLink tester stimulis	TCLink tester
debug_tester_fcw_i[9::0]	in	clk_sys_i	Tester NCO frequency control word	TCLink tester
debug_tester_nco_scale_i[4::0]	in	clk_sys_i	Tester NCO scaling factor	TCLink tester
debug_tester_enable_stock_out_i	in	clk_sys_i	Enable TCLink tester stock DCO phase to RAM	TCLink tester
debug_tester_addr_read_i[9::0]	in	clk_sys_i	Tester RAM address to be read-out	TCLink tester
debug_tester_data_read_o[15::0]	out	clk_sys_i	Phase read-out from tester RAM	TCLink tester

Table 1: **TCLink** core ports description. async: synchronized internally

3 TCLink Configuration

If any parameter has to be changed in the design (clock-frequencies or protocol), the port coefficients have to be changed. In order to help the user, a high-level model is available. The configuration can be changed in the *software/config/default.csv* file. This file is composed of two parts, user parameters and expert user parameters. The first part regard the link and digital design parameters which may be eventually different than the default configuration as it can be observed in figure 4.

```
#============================================================================
#
#                          TCLink user configuration
#
#============================================================================
# Design parameters - Link/architectural design fixed parameters
#============================================================================
# Free-running system clock frequency in Hz
clk_sys_freq, 125e6

# Transmitter data-rate in b/s
tx_datarate, 10.24e9
# Transmitter high-speed PLL clock divider (check your transceiver design and read
Ultrascale user guide to know more about txoutdiv parameter)
txoutdiv, 1

# Carrier frequency (low-speed Tx and Rx clocks) in Hz
carrier_freq, 320e6

# Rx word width in bits
rx_word_width, 32
```

Figure 4: TCLink user parameters

The second part (shown in figure 5) are the phase-detector and controller parameters which require a more in-depth understanding on TCLink design. Users are also welcome to contact us in case they want to change those.

```
#============================================================================
#
#                          TCLink expert user configuration
#
#============================================================================
# TCLink loop parameters - Loop parameters for TCLink design
#============================================================================
# --- Phase-detector ---
# Phase-detector (DDMTD) beat-frequency (Hz) and averaging number
ddmtd_beat_freq, 624e3
ddmtd_avg, 64

# --- Mirror compensation ---
# Enables TCLink mirror compensation branch
enable_mirror, 1
# Alpha coefficient for mirror compensation branch (variation downstream /
variation roundtrip)
alpha, 0.5

# --- Loop dynamics ---
# Loop natural frequency in Hz (it is recommended to choose a value at least 50x
smaller than ddmtd_beat_freq/(ddmtd_avg+1))
# For a proportional-only loop (enable_Ki=0), this corresponds to the -3dB cutoff
frequency
natural_freq, 100
# Loop damping coefficient (only useful when enable_Ki=1)
damping, 1.44
# Enable Integral part of the controller
enable_Ki, 0
# Sigma-delta oversampling factor (multiple of 2)
SD_OSR, 32

#============================================================================
# TCLink tester parameters - Tester parameters (only useful when tester is enabled)
#============================================================================
# Sinus amplitude in ps
amplitude_sinus,200.0
# Sinus frequency in Hz
frequency_sinus, 15.0
```

Figure 5: TCLink expert user parameters

Once all parameters are changed, the user can run the script *software/model_transfer_function.py* in order to calculate the TCLink port values. This script gives a table of the core parameters and the TCLink closed-loop transfer-function as shown in figure 6.
Our reference design, explained in section 4 is the ideal platform to validate a new parameter and to compare the link behaviour in real-life with the high-level model. The files *software/fpga_transfer_function_kcu105.py* or *software/fpga_transfer_function_vcu118.py* can be used for this purpose. More information is given in the example design quick start guide.

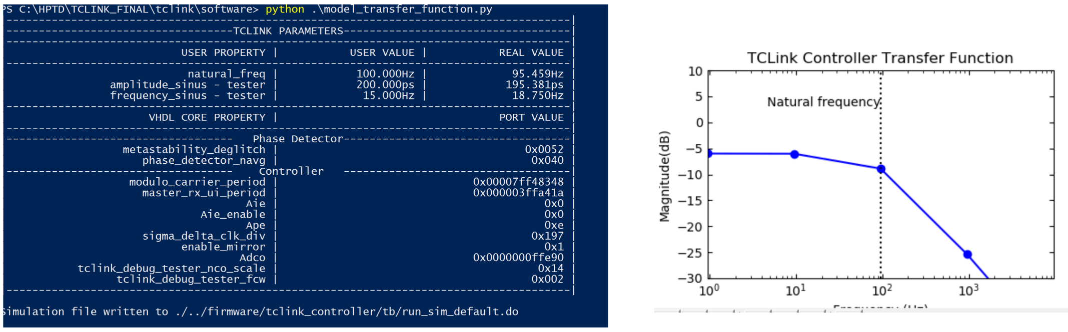

Figure 6: TCLink model example

4 Reference designs

Overview

Two reference designs are available on GIT (*https://gitlab.cern.ch/HPTD/tclink*) targetting Ultrascale GTY transceivers on the Xilinx VCU118 board and GTH transceivers on the Xilinx KCU105 board. In order to bring-up the hardware designs, check the files **quick_start_guide_vcu118.pdf** or **quick_start_guide_kcu105.pdf**. The design block diagram for VCU118 is shown in figure 7.
The example design integrates TCLink, HPTD IP, the communication protocol (lpGBT10G for VCU118 and lpGBT-FPGA for KCU105) and a mesochronous clock-domain crossing (40MHz to 320MHz for the Tx and 320MHz for 40MHz to the Rx) in a single core to enable an easy integration for the user.
The transceivers are generated with a data-rate of 10.24Gb/s (typical LpGBT-FPGA data-rate) with a reference clock of 320MHz. A 40MHz user clock locked to the master transmitter reference clock has to be provided for the user logic. The **DDMTD** heterodyne clock frequency is 319.376MHz, please note that this clock shall come from the same PLL generating the transceiver reference clock and in the example it is generated using an internal MMCM. The system clock frequency is 125MHz. The example design also instantiates the **HPTD IP** core which has to be used together with **TCLink**.
The example design transmitted data for the **VCU118** example design is the lpGBT-10G protocol (bidirectional lpGBT10G-FEC5 protocol). This example design targets back-end applications and an example of master timing and master slave is given.
The example design transmitted data for the **KCU105** example design is the lpGBT-FPGA protocol. This example design targets back-end to front-end applications and therefore only an example of master timing is given (the slave is the lpGBT itself). Eight masters are integrated in this example design.
In case another protocol is used, the user shall adapt the example design for its own protocol.

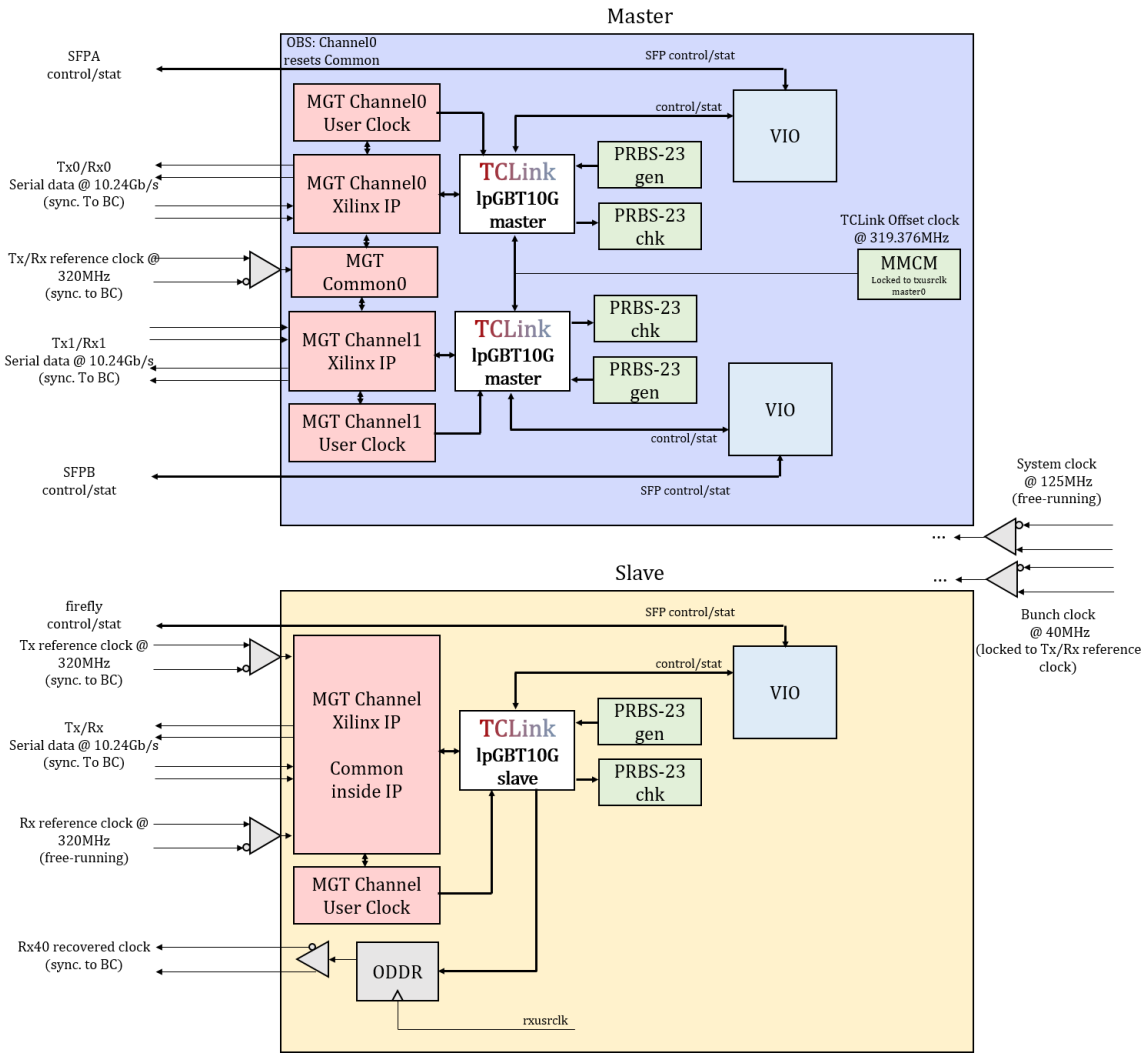

Figure 7: TCLink example design simplified block-diagram for VCU118 using lpGBT10G protocol - Back-end ↔ Back-end applications