PG2L100H LVDS 7TO1 Application Guide

(AN04008, V1.1)

(22.03.2022)

Shenzhen Pango Microsystems Co., Ltd.

Revisions History

Document Revisions

Version	Date of Release	Revisions
V1.1	22.03.2022	Initial release.

About this Manual

Terms and Abbreviations

Terms and Abbreviations	Meaning
LVDS	Low-Voltage Differential Signaling
LPC	Lane per channel

Table of Contents

Tables

Figures

Chapter 1 Overview

1.1 Introduction

This document serves as a system-level application scheme for the PG2L100H LVDS7TO1 product launched by Shenzhen Pango Microsystems Co., Ltd. It mainly introduces the function list, design architecture, interface definitions, and reference design of LVDS7TO1.

1.2 Main Functions

The main functions supported include:

- According to the chip IO and clock resources, each Channel supports multiple data Lanes, and each Bank supports multiple Channels.
- Static and dynamic alignment modes supported
- The receiver and transmitter can be called separately.
- The delay of each lane is adjustable. The output delay can be adjusted for up to 128 steps, with a delay of 5ps for each step. The input delay can be adjusted for up to 248 steps, with a delay of 10ps for each step.
- The maximum rate of a single lane is 900Mbps.

1.3 1.3 Design Information

PG2L100H LVDS7TO1	
Supported Devices	PG2L family FPGA products
Provided Design Files	
Design File	Encrypted file
Reference Designs	Verilog file
Simulation File	Verilog file
Constraint File	fdc file
Development Tools	
Design Tools	Pango Design Suite 2020.3
Simulation tool	Modelsim

1.4 Resource Usage

When the number of data lanes is 8, the resource usage of the LVDS7TO1 reference design scheme is as follows:

Table 1-1 Resource Usage

Device	DRM	FF	LUT	PLL
PG2L100H	45	1507	1309	2

Chapter 2 Function Description

2.1 Transmitter

2.1.1 Transmitter Logic Block Diagram

Figure 2-1 describes the logic architecture of the LVDS7TO1 transmitter. Through data splicing, the 7-bit data is first converted into 14-bit parallel data, which is then converted to 7-bit serial data through the 14:1 serialisation mode.

The external clock goes through the FPGA internal PLL and becomes 3 clocks: IO_CLK, which is sent to the OSERDES unit as the serialised high-speed IO clock; DIV_CLK, which is used by the 7-bit to 14-bit conversion unit; and PCLK, which serves as the external parallel data interface clock.

The frequency ratio of the three clocks is 7:1:2.

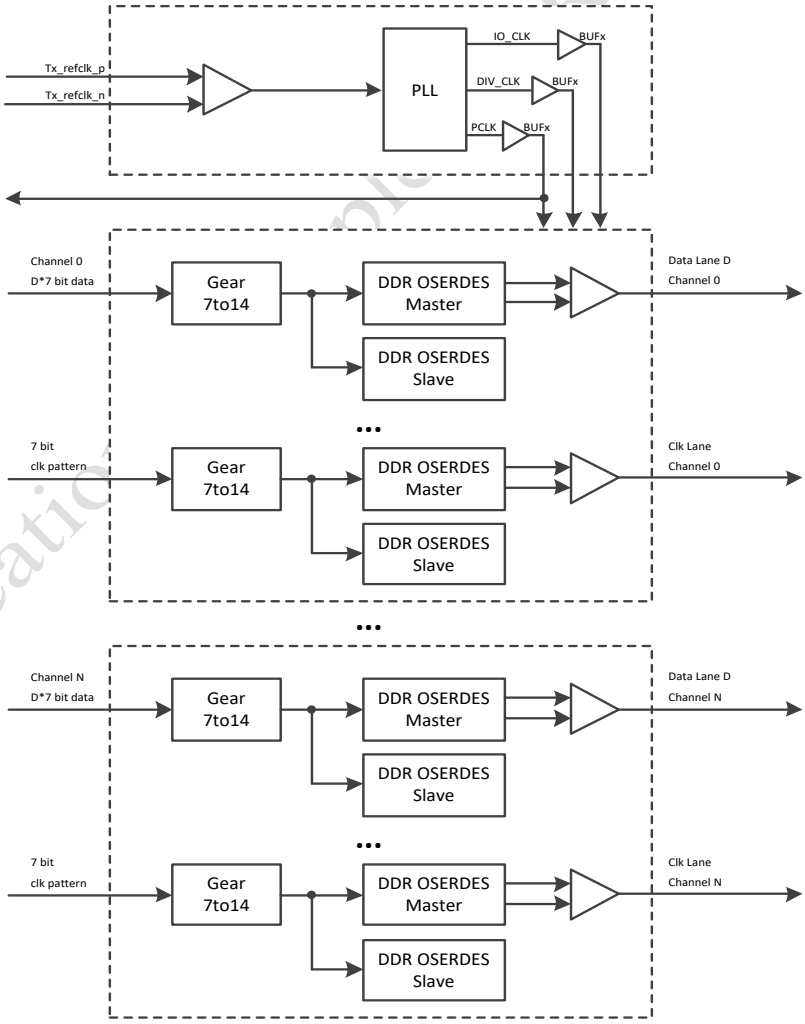

Figure 2-1 Transmitter Logic Block Diagram

The synchronous clock is obtained through the serialised output of a specific clk Pattern at a 7:1 ratio. It is used as the receiver source clock and for data boundary confirmation.

2.1.2 Transmitter Interface Definitions

Table 2-1 Transmitter Interface Definitions

Signal	Input/Output	Bit width	Description
tx_refclk_i	input	1	Master clock, provided by an external crystal oscillator, 50MHz
rstn_i	input	1	System reset signal, active-low
chn0_txdata_i	input	7* LPC	Parallel data input, with a bit width of 7 * LPC
chn0_dataout_p/n _o	output	LPC	Transmitter data differential signal
chn0_clkout_p/n_o	output	1	Transmitter clock differential signal
tx_pclk_o	output	1	Parallel data interface clock

2.1.3 Transmitter Parameter Definitions

Table 2-2 Transmitter Parameter Definitions

parameter	Description
TX_CLK_PTN	Clock pattern, with a default value of 7'b1100001.
LPC	The number of data lanes per channel, with a default value of 8.
FRAME_TYPE	Optional between PER_LINE and PER_CLK, corresponding to different interface mapping methods.

2.1.4 Transmitter Interface Timing

When FRAME_TYPE = PER_LINE, the interface timing is shown in Figure 2-2.

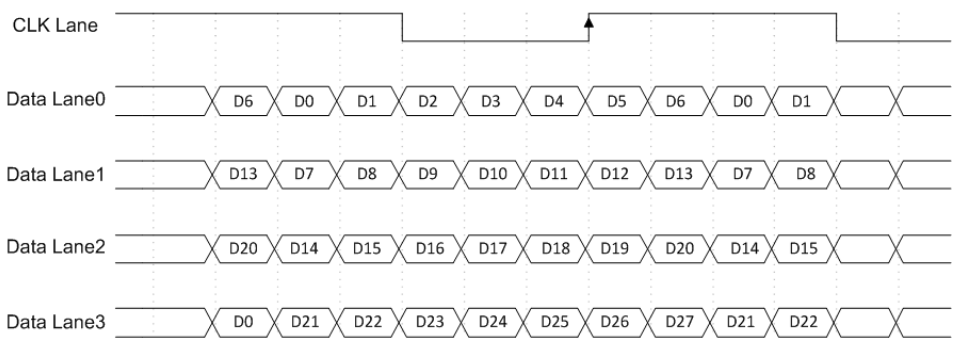

Figure 2-2 Transmitter Interface Timing 1

When FRAME_TYPE = PER_CLK, the interface timing is shown in Figure 2-3.

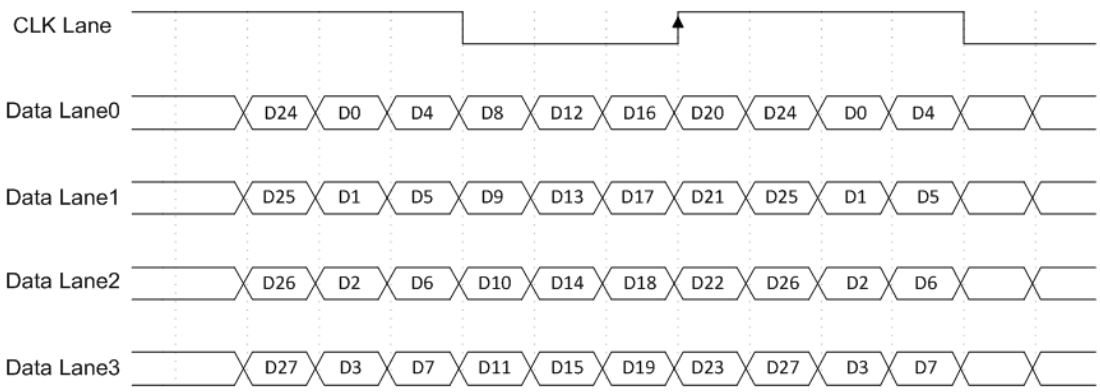

Figure 2-3 Transmitter Interface Timing 2

2.2 Receiver

2.2.1 Receiver Logic Block Diagram

Figure 2-4 describes the logic architecture of the LVDS7TO1 receiver. The differential clock signal is converted to a single-ended clock signal through the IOB, then passes through a PLL to generate a deserialisation clock and a parallel data interface clock. Meanwhile, the differential clock signal is also used as data to recover the parallel 7-bit clk_pattern through the ISERDES unit. While restoring the clock pattern, the receiver system automatically calculates the clock channel window and the optimum IO Delay value, finds the boundary of the parallel data through Bitslip, and then applies these values to all data channels.

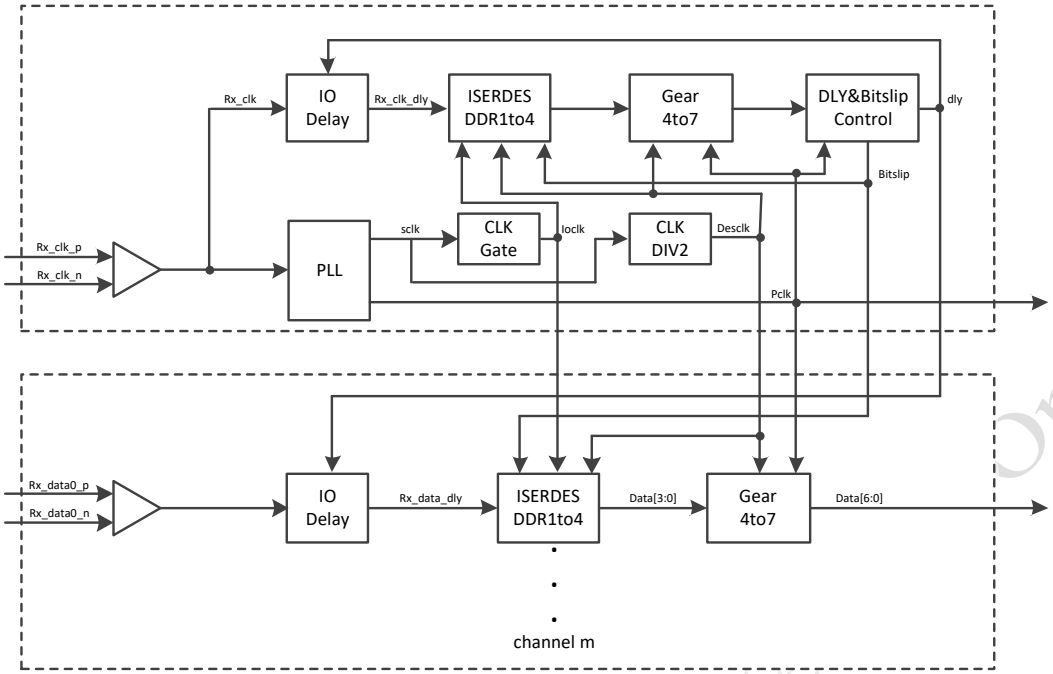

Figure 2-4 Receiver Logic Block Diagram

2.2.2 Receiver Interface Definitions

Table 2-3 Receiver Interface Definitions

Signal	Input/Output	Bit width	Description
rstn_i	input	1	System reset signal, active-low
data_in_p/n_i	input	LPC	Receiver data differential signal
clkin_p/n_i	input	1	Receiver clock differential signal
rx_pclk_o	output	1	Receiver parallel data interface clock
rx_data_o	output	7*LPC	Receiver parallel data

2.2.3 Receiver Parameter Definitions

Table 2-4 Receiver Parameter Definitions

parameter	Description
TX_CLK_PTN	Clock pattern, with a default value of 7'b1100001.
LPC	The number of data lanes per channel, with a default value of 8.
FRAME_TYPE	Optional between PER_LINE and PER_CLK, corresponding to different interface mapping methods.

2.2.4 Receiver Interface Timing

When FRAME_TYPE = PER_LINE, the interface timing is shown in Figure 2-5.

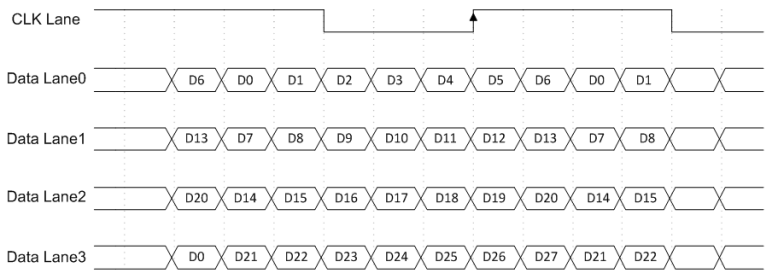

Figure 2-5 Receiver Interface Timing 1

When FRAME_TYPE = PER_CLK, the interface timing is shown in Figure 2-6.

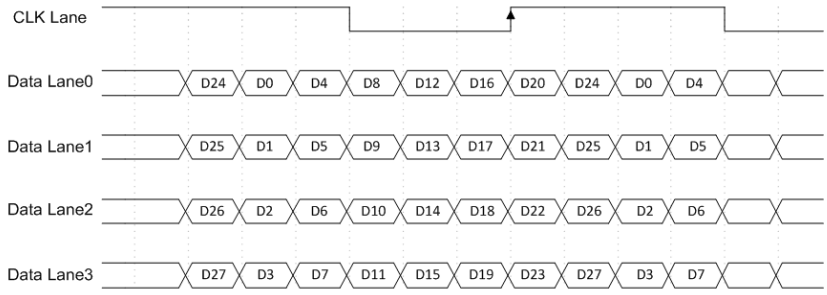

Figure 2-6 Receiver Interface Timing 2

Chapter 3 Reference Design

3.1 Reference Design Project Directory

pgr_lvds_7to1_ddr_8lane

```
├─bench                                  //Simulation file
├─docs                                   //Design document
├─pnr                                    //Project directory
│ │─pgr_lvds_7to1.pds                    //PDS project file
├─sim                                    //Simulation project directory
│ ├─file_list.f                          //Design file list
│ ├─lvds_sim.bat                         //Simulation scripts
│ ├─lvds_sim.tcl                         //Simulation TCL scripts
│ ├─wave.do                              //Simulation waveform scripts
│ ├─sim_lib                              //Simulation library files
├─src                                    //RTL files included in the design example
│ ├─LVDS_7to1_DDR_RX                     //LVDS receiver source code
│ ├─LVDS_7to1_DDR_TX                     //LVDS transmitter source code
│ ├─LVDS_TEST                            //LVDS test source code
│ ├─lvds_7to1_ddr_top.v                  //Top-level module of reference design
│ ├─P04I100KF01_LVDS.fdc                 //Project constraint file
```

Code design:

lvds_7to1_ddr_top: top-level module, external interfaces and module mapping

lvds_tx_top_7to1_ddr: transmitter top-level modules, including clock and data modules, which implement high-speed differential output of clock and data.

lvds_rx_top_7to1_ddr: receiver top-level modules, including clock, data and alignment modules, which convert high-speed differential signals to parallel data.

Pgr_test_top_inst: a test module that generates test data and verifies the received data. Through the tx_mode signal, the module can switch between 4 types of test data for output, including pattern_clk type (i.e., tx_data[6:0]=7'b1111000), bit toggle type (i.e., tx_data[6:0]=7'b1010101), CNT type (tx_data[6:0] varying linearly between 7'd0 and 7'd127), and PRBS-7 type data (i.e., tx_data[6:0]=PRBS-7).

IP:

In this project, both the transmitter and the receiver need one PLL IP.

When the LVDS line rate is 900Mbps, the specific configuration of the tx-side PLL is shown in Figure 3-1, where Clkout2 is the LVDS SDRAM clock frequency.

The frequency ratio of Clkout0, Clkout1 and Clkout2 is 7:2:1.

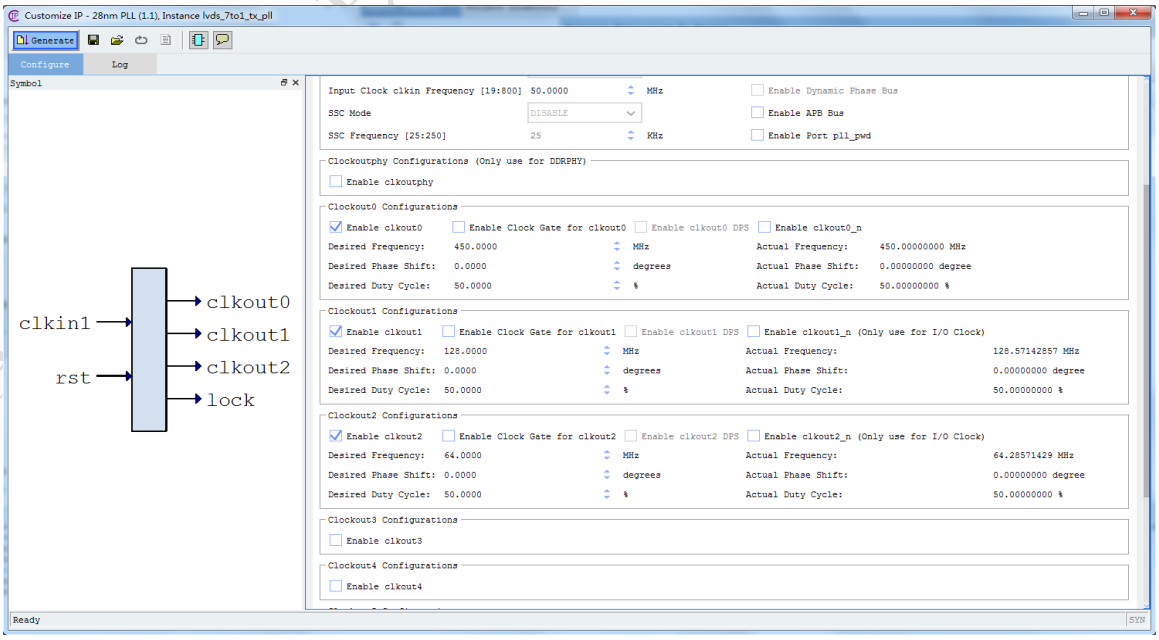

Figure 3-1 Transmitter PLL Configuration

The rx-end configuration is shown in Figure 3-2.

The input clock frequency is set to the received LVDS SDRAM clock frequency. The frequency ratio of clkin, clkout0, and clkout1 is 2:7:2.

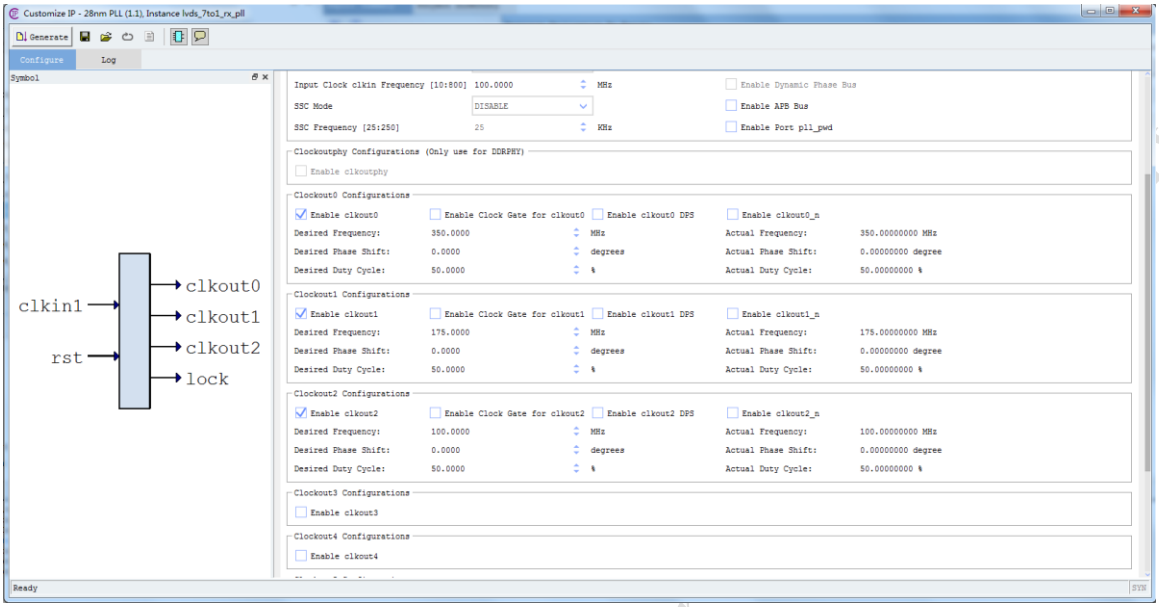

Figure 3-2 Receiver PLL Configuration

3.2 Reference Design Simulation

Modelsim version: SE-64 10.2c.

Double-click the lvds_sim.bat file in the sim folder of the project path to start the simulation. The simulation waveform is shown in Figure 3-3.

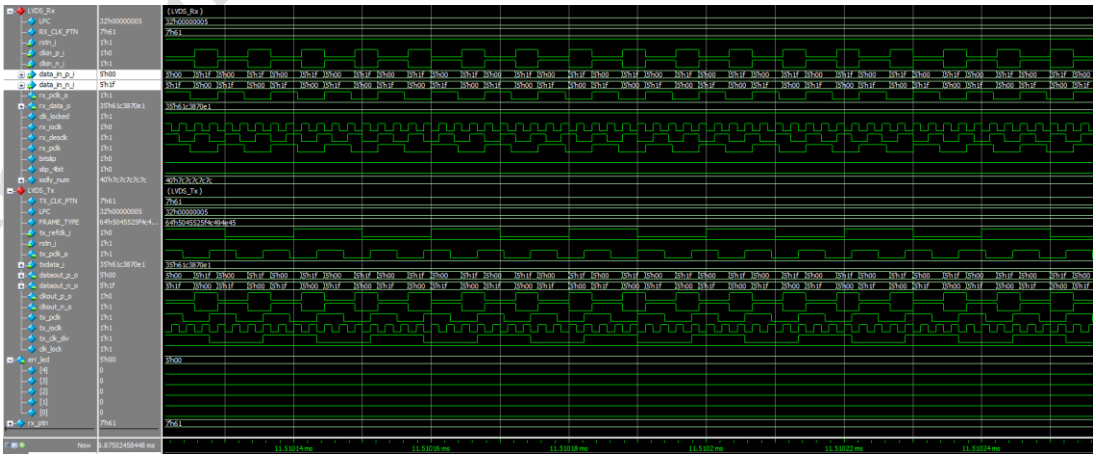

Figure 3-3 Simulation Waveform

3.3 Reference Design On-board Verification

3.3.1 Self-loop Verification

Software environment: Pango Design Suite 2020.3

Hardware environment: P04I100KF01_A2, FMC_LVDS_TEST_A0

This reference design can be used for single-board self-loop verification or inter-board docking verification. This document takes single-board self-loop verification as an example, with a block diagram as shown in Figure 3-4.

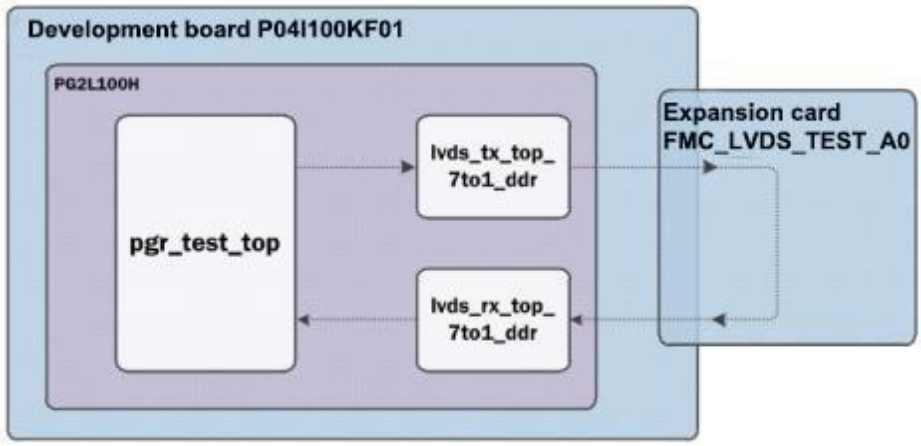

Figure 3-4 Self-loop Verification Block Diagram

The hardware environment for verification is shown in Figure 3-5:

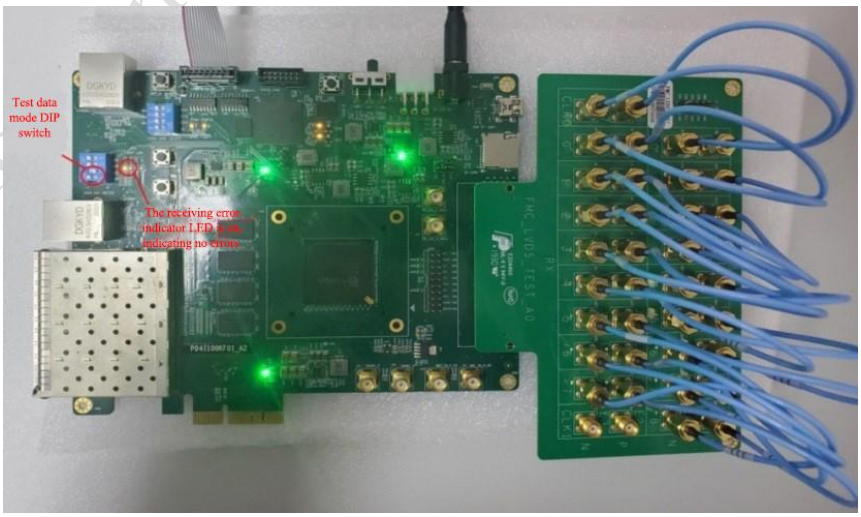

Figure 3-5 Hardware Environment for Self-loop Verification

The transmitter and receiver are connected by an SMA coaxial cable, with the IO Bank voltage for the LVDS transmitter/receiver set to 2.5V. Additionally, if the receiver uses external termination matching resistors, the on-die matching resistors must be turned off.

3.3.2 Screen Lighting Verification

Software environment: Pango Design Suite 2020.3

Hardware environment: P04W100AS01_A0, LVDS Screen

For the transmitter, perform screen lighting verification separately. The LVDS interface screen lights up normally, as shown in Figure 3-6 .

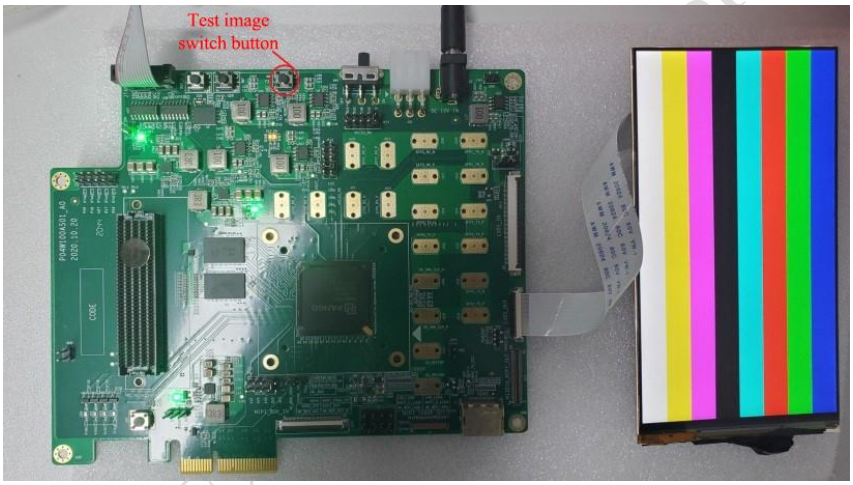

Figure 3-6 Hardware Environment for Screen Lighting Verification

For any needs about specific screen lighting verification projects, please consult our FAE.

Disclaimer

Copyright Notice

Disclaimer

1. This document only provides information in stages and may be updated at any time based on the actual situation of the products without further notice. The Company assumes no legal responsibility for any direct or indirect losses caused by improper use of this document.
2. This document is provided "as is" without any warranties, including but not limited to warranties of merchantability, fitness for a particular purpose, non-infringement, or any other warranties mentioned in proposals, specifications, or samples. This document does not grant any explicit or implied intellectual property usage license, whether by estoppel or otherwise.
3. The Company reserves the right to modify any documents related to its series products at any time without prior notice.
4. The information contained in this document is intended to assist users in resolving application-related issues. While we strive for accuracy, we cannot guarantee that the document is entirely free from flaws. Should any functional abnormalities and performance degradation arise due to deviation from the prescribed procedures outlined herein, our company will neither be held liable nor concede that such issues stem from product deficiencies. The solutions presented in this document are just one of the feasible options and cannot cover all application scenarios. Consequently, if users encounter functional abnormalities or performance degradation despite adhering to the prescribed procedures outlined herein, we cannot assure that such issues are indicative of product deficiencies.