The LpGBT tester is an FPGA based test system for the LpGBT ASIC

Abstract

The low-power Gigabit transceiver, named LpGBT[1], is a new radiation tolerant ASIC developed to handle the communication between the electronics and the back-end electronics of the CERN experiments. Planned to be used in the HL-LHC upgrades, this chip was designed in order to support the communication between the detectors and the back-end electronics on the detector electronics (uplink) at 5.12 or 10.24 Gbps. In addition of the main stream used to transmit the timing, trigger and control data, the LpGBT supports several interfaces to monitor and control the ASIC's slow control interfaces (I2C, DAC, SPI). The above block diagram shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT is a new radiation tolerant ASIC developed to handle the communication between the electronics and the back-end electronics of the CERN experiments. Planned to be used in the HL-LHC upgrades, this chip was designed in order to support the communication between the detectors and the back-end electronics on the detector electronics (uplink) at 5.12 or 10.24 Gbps. In addition of the main stream used to transmit the timing, trigger and control data, the LpGBT supports several interfaces to monitor and control the ASIC's slow control interfaces (I2C, DAC, SPI). The above block diagram shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.

The LpGBT tester targets the Xilinx Virtex 7 FPGA hosted on the VC707 development kit. This setup provides an Ethernet interface used for the control of the test via the IPBus protocol. This standard provides an easy way to read and write registers using Python. Therefore, all of the modules developed to test the different interfaces of the LpGBT can be configured and monitored using a memory mapped bus. The diagram below shows the different cores implemented to check the functionality of the ASIC. The design can be easily separated into three parts: slow control features, data path and clocks.