Multi-port Remote JTAG over Optical Fibers under Radiation Environment

M. Nakao, Y. Nakazawa, H. Sudo, R. Honda, and N. Taniguchi

***Abstract*— The JTAG (Joint Test Action Group) protocol is a popular method to program FPGA (Field Programmable Gate Array) devices where a more intelligent technology is not applicable. However, the original JTAG protocol is designed for a short distance connection and not necessarily suitable when the FPGA device is located in a remote radiation area. We developed a custom optical transmission technique for the JTAG protocol based on discrete devices, and implemented in a small test board receiver and a multiport JTAG distributor. We also developed a technique to overcome the latency due to serialization and cable length. We present the evaluation results and future applications.**

I. Introduction

The JTAG (Joint Test Action Group) protocol defined by the IEEE standard 1149.1 [1] is a popular method to program FPGA (Field Programmable Gate Array) devices. Modern high energy physics (HEP) experiments use a large number of FPGAs in order to process a huge number of detector signal channels in a real-time manner. The Belle II experiment [2] at the SuperKEKB e^+e^- collider [3] in Tsukuba, Japan, is also one of such HEP experiments.

The JTAG protocol is a low-level serial protocol to access a device via three input signals (TCK, TMS, TDI), and one output signal (TDO). (One more optional reset signal (TRST) is unused in FPGA programming.) The protocol is popular because of its simplicity. A typical FPGA has these four pins, and a set of tools is provided. In case of Xilinx [4] FPGAs, software tools to program FPGAs (Impact, ChipScope, Vivado programs) and a USB (universal serial bus) cable to generate and capture the JTAG signals are provided.

The protocol drives a state machine using the TCK signal. There are 16 states, and the next state upon TCK is defined by the TMS signal as shown in Fig. 1. These states are used to write and read an instruction register (IR) and data registers (DR); TDI is the input signal, and TDO is the output signal. The TCK signal, despite its name, is not a fixed frequency clock signal, but is a trigger for the state transition with a minimal interval. The TDO signal has to arrive within the TCK interval, and this constraint limits the TCK interval and hence the JTAG cycle frequency, or the length of the JTAG cable.

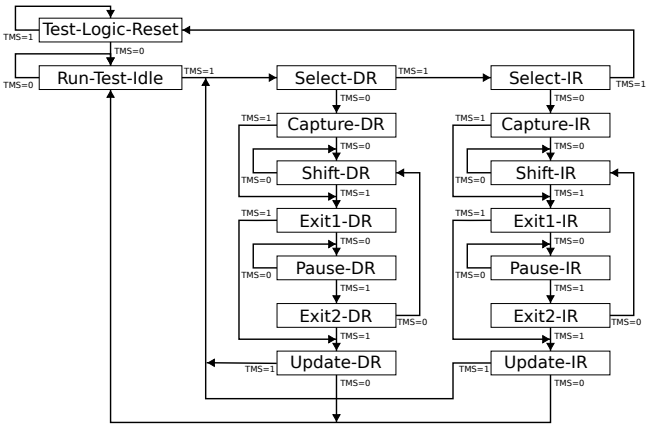

Fig. 1. JTAG state transition.

Multiple devices can be connected in a chain as shown in Fig. 2. As TDO output is identical for the same programming sequence of the same device and firmware, it is also possible to program multiple devices in parallel, just by distributing the same TDI signal and using one of TDO signals, or using a logical-AND or logical-OR of TDO signals.

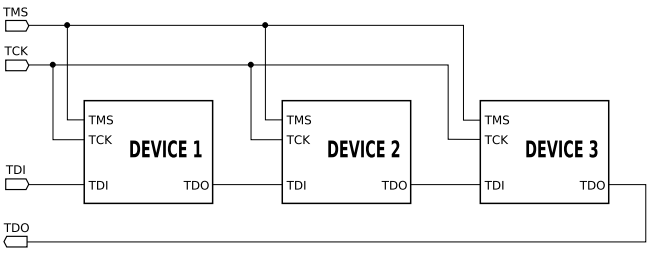

Fig. 2. JTAG chain.

In a HEP experiment, the FPGA devices are often located at a remote place, far from the PC where the programming software runs. A typical way is to convert the signal into LVDS (low voltage differential signaling) or other differential signaling. Since JTAG signal is not DC balanced, AC coupling cannot be used, and the electrical ground has to be connected over a long distance. It is not possible to use SFP (small form-factor pluggable) optical transceiver modules or other commodity modules that are popular for high speed serial data transmission, as they can only be used with AC coupled

Manuscript submitted May 20, 2024.

This work is partially supported by Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research C 21K03595.

M. Nakao is with High Energy Accelerator Research Organization (KEK), Tsukuba, Ibaraki Japan (e-mail: mikihiko.nakao@kek.jp).

Y. Nakazawa is with High Energy Accelerator Research Organization (KEK), Tsukuba, Ibaraki, Japan (e-mail: nakancyo@post.kek.jp).

H. Sudo is with University of Tokyo, Hongo, Tokyo, Japan (email: hiroto.sudo@hep.phys.s.u-tokyo.ac.jp).

R. Honda is with High Energy Accelerator Research Organization (KEK), Tsukuba, Ibaraki, Japan (e-mail: rhonda@post.kek.jp).

N. Taniguchi is with High Energy Accelerator Research Organization (KEK), Tsukuba, Ibaraki, Japan (e-mail: nanae@post.kek.jp).

signals. In addition, the latency due to the long distance requires a larger TCK interval and hence lower programming speed. Typically, for a 10m LVDS extension setup, the default 6MHz programming of the Xilinx Impact program does not work, and the frequency has to be reduced to 3MHz and hence the FPGA programming takes twice more time.

In the following sections, we present a new solution to solve these problems.

II. JTAG at Belle II

The Belle II experiment is a flavor physics experiment to search for physics beyond the Standard Model from an unprecedentedly large sample of decays of bottom mesons, charm hadrons, and tau leptons. The Belle II detector consists of seven subdetectors, of which three of them, the central drift chamber (CDC), the time-of-propagation counter (TOP), and the aerogel ring-image Cherenkov counter (ARICH), use FPGA boards inside the detector volume where the space is very limited, cabling is very tight, and the radiation due to gamma-rays and neutrons is very harsh. CDC uses 299 FPGA boards each with a Xilinx Virtex 5 FPGA, TOP uses 64 FPGA boards with Xilinx ZYNQ, and ARICH uses 72 FPGA boards with Xilinx Virtex 5.

These FPGA boards are located up to 10 m away from a more radiation-safe area. The current FPGA boards use JTAG signals translated to the LVDS level for the long distance DC coupling transport over category-7 Ethernet cables. We also use similar category-7 Ethernet cables to distribute the clock, trigger, and other timing signals to these FPGA boards using a custom *b2tt* serial protocol [5] which is DC-balanced and capable for AC-coupled connections. The other end of the JTAG signals is implemented on the Frontend-Timing-Switch (FTSW) module [5], which has up to 20 RJ-45 output ports directly connected to an FPGA to drive and receive the LVDS signals in any format. The FTSW module can also equip 8 SFP optical transceiver by reducing the output RJ-45 ports to 12.

The JTAG sequences are distributed over the *b2tt* protocol, as a part of the timing distribution. Functions corresponding to *initialize chain*, *get idcode*, *program*, and *verify* are implemented in a custom program *jtagft*, which directly manipulates an FTSW module to generate the sequence and distribute to the FTSW modules next to the FPGA boards. In the case of the TOP subdetector, the FTSW module is used just as the switcher of the JTAG sequence provided by the Vivado software and a USB cable with a long distance extension, since the *jtagft* program does not support the programming of the processor part of the ZYNQ FPGA.

Belle II is now in the phase to design the next generation FPGA boards, to increase the data bandwidth, processing capability, and radiation tolerance. One of the concerns is the directly connected metal Ethernet cables which would be a source of electric disturbance to the stable operation of the FPGA boards. A major obstacle is a lack of the technique to program the FPGA using an optical-fiber-only connection.

III. Optical JTAG Design

In order to realize the JTAG protocol over optical fibers, we designed a circuit made of a small number of discrete devices to receive a custom encoded JTAG protocol (*optjtag*) from an AC-coupled connection, which is suitable for an optical fiber transport over a long distance.

Our design is inspired by Ref. [6], which uses a high speed serial link in a similar application. The idea is simple: 3-bit parallel signals of TCK, TMS, TDI are serialized into a DC-balanced serial stream at the sender, and deserialized by a receiver. For the TDO signal, only one bit has to be serialized. For the TCK signal, a protection mechanism is necessary to avoid a spurious JTAG sequence which can easily confuses an FPGA into an unrecoverable state without a power cycle.

To be used in a radiation area with a tight space constraint, our circuit includes no programmable devices, and uses only a small area on the remote FPGA board. The *optjtag* protocol can be easily implemented at the back-end on an inexpensive FPGA using no high speed transceivers. The protocol uses a 10-bit LVDS serializer and deserializer (TI DS92LV1023 / DS92LV1224), operated at a data rate around 400–660 Mbps, and a local clock source on the FPGA board at the front-end. For the protection mechanism, simple 74-series AND/NAND gates and a supervisor chip (ROHM BD46272G) are combined to safely use the lock signal of the deserializer. A conceptual diagram is shown in Fig. 3.

The back-end is implemented in an FTSW module for the Belle II application. The Virtex 5 FPGA of FTSW runs on an independent clock source, and can easily handle multiple *optjtag* ports in applications where many remote FPGA boards have to be programmed. For the Belle II application, we use a 381.6 Mbps data rate, which is slightly lower than the lower bound of the specification, in order to use a clock which can be derived from the system clock of 127.2 MHz. This corresponds to the JTAG signal of 31.8 MHz, and the internal clock of 190.8 MHz inside the FTSW module. It is fast enough as a typical JTAG programming speed is only up to about 10 MHz.

The optical JTAG circuit is operated by an independent clock source. The deserializer at the remote end is capable to absorb the clock frequency difference; the deserializer implemented in the FTSW module absorbs the clock frequency difference by oversampling the serial signal, which contains only one-bit information in a 31.8 MHz cycle.

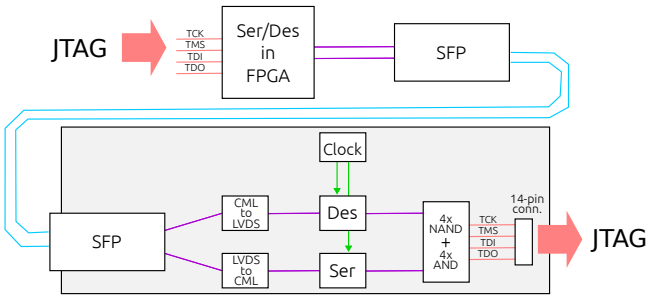

Fig. 3. Conceptual design of *optjtag* system using a back-end FPGA board and a ciruit on a remote FPGA board.

IV. Evaluation with a Test Board

We developed two types of small prototype test boards to demonstrate the *optjtag* functions. These boards serve as an adapter to translate the *optjtag* signal received by an SFP optical transceiver module into a Xilinx-standard JTAG 14-pin flat cable. The entire circuit fits within 2cm $\times$ 5cm area, besides the optical transceiver.

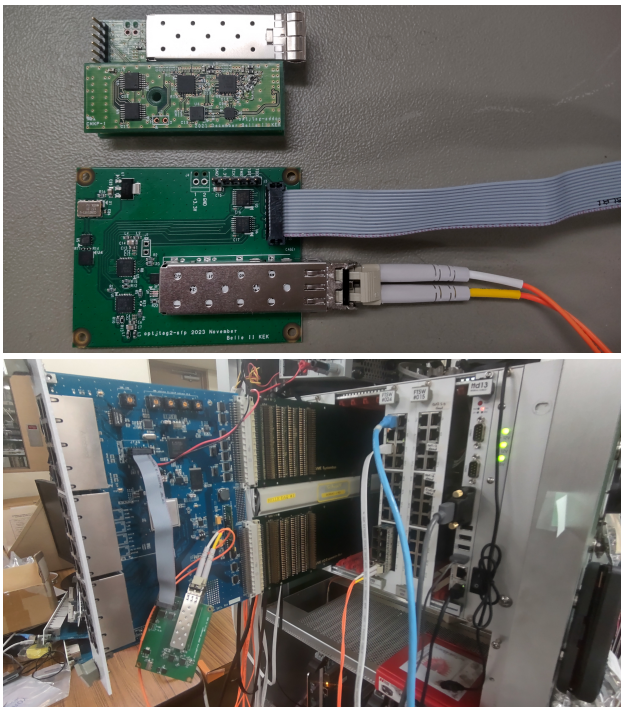

Fig. 4. Prototype receiver boards for the *optjtag* evaluation (top), and a test setup for the evaluation (bottom).

The back-end of the *optjtag* protocol is implemented on the FTSW module, which receives the JTAG signal from a USB JTAG program cable, and transmits via an SFP module. The target device of the JTAG protocol is another FTSW module. The prototype boards and the test setup are shown in Fig. 4.

With these prototype boards, we found a few minor issues. First, we found the CML (current-mode logic) output of the SFP transceiver cannot stably drive the LVDS input of the deserializer. Second, we found that the lock signal of the deserializer is not reliable, as it spuriously asserts the lock signal even when the SFP transceiver input is open. The former just requires a translation buffer, and the latter is solved by adding a supervisor chip to wait for a stable lock signal. Besides these issues, the JTAG operation was confirmed, but only at a four times slower rate of 1.5 MHz due to the overhead of the serializer and deserializer.

We irradiated one of the test boards up to 2 kGy of ^{60}Co γ-ray source, and found no loss of the functionality. Our requirement is to survive at 1 kGy, which corresponds to 10-year operation of Belle II at the most radiation-harsh FPGA-board area.

V. Overcoming the JTAG latency

A potential issue of a long JTAG line is the round-trip time of the signal, which limits the JTAG operation frequency. This can be overcome by introducing an artificial FPGA device (emulator) at the end of the JTAG chain, generated by a logic in the back-end FPGA.

TDO has to be received within one TCK interval ($\sim$160 ns for 6 MHz) if there is one FPGA in the JTAG chain, but it can be within N TCK intervals for N FPGAs in the chain. A real FPGA has a fixed latency for the TDO output, but an artificial emulator can bypass this latency.

The emulator does not need to support all JTAG cycles. The minimum set of JTAG cycles are: (1) *initialize chain* in which the ID code of the device is sent out, (2) IR (instruction register) cycle, and (3) DR (data register) cycles for *bypass*, *get idcode*, and *monitor*. The IR is 10-bit long for the Virtex 5 FPGA, and the DR size differs depending on the IR: *bypass* register is 1-bit long, and *idcode* and *monitor* registers are 32-bit long. Therefore, the amount of delay by the emulator has to be adjusted accordingly.

We developed a JTAG latency absorber logic, which includes two FPGA emulators as shown in Fig. 5. The delay has to be 20-bit (10-bit $\times$ 2) for the IR cycle, 64-bit (32-bit $\times$ 2) for the initialize chain where both emulators have to generate their ID codes, 2-bit (1-bit $\times$ 2) if both are in bypass DR cycle, or 33-bit (32-bit + 1-bit) for other DR cycles. We design a ring buffer, from which the delayed output of the target device is fetched with a proper delay. The timing to write into the ring buffer is a TCK timing delayed by the physical delay due to the remote FPGA, but since JTAG is a slow signal, the timing requirement is not severe.

In this latency absorber logic, the first FPGA emulator works as an emulator of the JTAG state machine of the remote FPGA. It also bypasses the TDI input signal for the succeeding FPGA emulators in the chain, for the JTAG cycle of those emulators to work as expected. The amount of necessary delay is known from the JTAG state of the first emulator, and hence a proper position of the ring buffer from the remote FPGA is known. The overall TDO output is either the delayed TDO from the remote FPGA, or the output of the FPGA emulator chain.

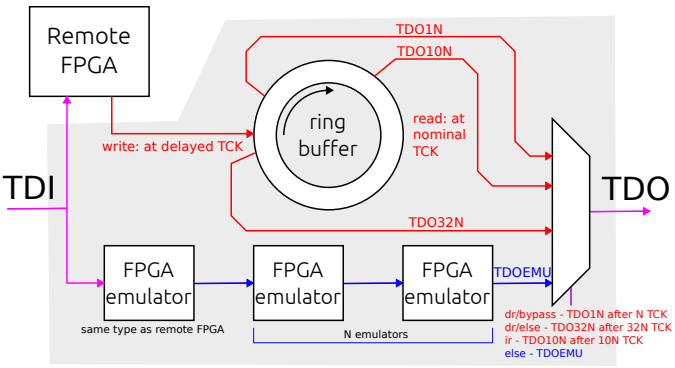

Fig. 5. Conceptual design to absorb the JTAG latency by two FPGA emulators.

Two FPGA emulators inserted in the JTAG chain gains 230 ns extra latency, measured in our test setup including some other overheads, when Xilinx Impact was operated at the default speed of 6 MHz. We successfully operated the JTAG chain with a 20m category-7 Ethernet cable in this setup at the default speed. Screenshots of the Xilinx Impact program and ChipScope program are shown in Fig. 6. Here, the target remote FPGA is XC5VLX30 Virtex 5 FPGA, while for the emulator, we randomly chose them to be XC5VLX20T and XC5VLX85, just by changing the ID code that the emulator returns.

The *optjtag* part of the JTAG transport is yet to be combined with this FPGA emulator, to test an even longer distance, e.g., 50m, using optical fibers. One can add multiple dummy devices to absorb any latency of the long JTAG line and additional devices.

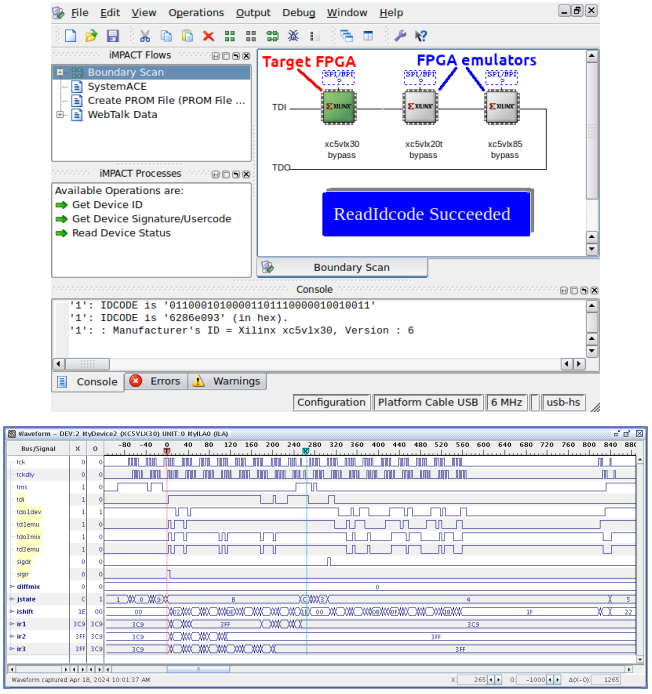

Fig. 6. Example of the successful JTAG operation with two FPGA emulators, shown with the Xilinx Impact program (top) and ChipScope program (bottom).

VI. Applications at Belle II

The main target application is the next version of the FPGA board for the Belle II CDC readout. The board is designed to be operated with fully optical input of clock, trigger, and JTAG. As it is operated in the radiation area, it is unavoidable to reprogram the FPGA several times a day to recover from unrecoverable single event upset errors due to the background neutrons. Since the FPGA size and hence the configuration memory size increases with respect to the current one, a faster JTAG programming solution is necessary. Second prototype production is on-going, while the target replacement date is in 2026 or later.

A new FTSW module with a large number of optical clock, trigger, and JTAG distribution is also planned. Using 14 QSFP ports, up to 48 optical connections to the FPGA boards will be possible. This board will be the timing and JTAG distributor board for the upgraded CDC FPGA boards.

By combining these two developments, we will be able to program a large number of FPGAs in the radiation area of the Belle II experiment.

VII. Summary

We present a new optical JTAG transport, and evaluated its function. The additional circuit to be added in the radiation area consists of discrete devices only, and does not require an extra large area.

We also present a new technique to overcome the JTAG latency by implementing artificial FPGA emulators in the JTAG chain, and demonstrated its function.

These solutions will be used in the planned upgrade of the Belle II readout system, which requires a large number of FPGA boards and hence JTAG programming ports.

References

[1] IEEE 1149.1 Working Group. [Online] Available: https://grouper.ieee.org/groups/1149/1/

[2] T. Abe *et al.* (Belle II Collaboration), "Belle II Technical Design Report," arXiv:1011.0352 [physics.ins-det].

[3] K. Akai, K. Furukawa, H. Koiso *et al.* (SuperKEKB Collaboration), "SuperKEKB collider," *Nucl. Instrum. Methods Phys. Res., Sect. A* vol. 907, pp. 188–199, Nov. 2018, 10.1016/j.nima.2018.08.017.

[4] AMD Xilinx. [Online] Available: https://www.xilinx.com/products/silicon-devices/fpga.html

[5] M. Nakao, *Timing Distribution for the Belle II Data Acquisition System,* JINST **7**, C01028 (2011).

[6] B. Deng et al., *JTAG-based remote configuration of FPGAs over optical fibers,* JINST **10** C01050 (2015).