ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR009

DEVELOPMENT OF EVENT RECEIVER ON Zynq-7000 EVALUATION BOARD

H. Sugimura*, High Energy Accelerator Research Organization (KEK), Tsukuba, Japan

Abstract

The SuperKEKB accelerator uses “Event Generator” and “Event Receiver”, an event timing module developed by Micro Research Finland company. It is suitable for pulse-by-pulse control because Injector Linac is generated various parameters of timings for the multi rings in each pulse. I tried to develop a new event receiver module by using FPGA (Zynq) evaluation board so that the specification can be changed flexibly according to the nature of the accelerator. I focus on the convenience of Zynq-chip such as serial data optical transfer (GTX) and embedded processor system. Finally, I aim for developing standalone event receiver module, and composite module that are integrated with BPM and RF system.

INTRODUCTION

KEK accelerator facility in Tsukuba, Japan is managed five rings (PF, PF-AR, SuperKEKB (LER, HER, DR) [1]). The first two rings are synchrotron radiation facility, and others are collider rings for particle physics. These rings are handled electron and positron beam, and are injected from one linac simultaneously at the repetition of 50 Hz. The timing system of our accelerator facility is required various parameter to fire at pulsed magnets such as Kicker and Septum magnets. And also, RF system and pulsed magnets (dipole, quadrupole and steering ...) at linac are required to identify which particle and which energy will be accelerated because the linac is needed to consider the five rings. So, we introduced event timing system to deliver some timings and identification codes simultaneously [2]. We focused on the function of event code delivery made by Micro Research Finland Oy(MRF) [3], and introduced “Event Generator(EVG)” and “Event Receiver(EVR)” as main component of a timing station. Furthermore, very flexible modules, EVG and EVR (EVE and EVO series), were developed by SINAP which conforms to MRF modules. The advantage of the SINAP timing module have a fine delay function. The resolution is 5ps. SINAP EVE and EVO are used sub-timing station at LER and HER in SuperKEKB.

MRF timing system is assigned the first 8-bit is event code and the last 8-bit is data buffer mode or distributed bus bit mode alternatively in a frame. The 16-bit of the event frame is transferred in synchronization with 114.24 MHz of RF clock which is obtained by dividing S-band 2856 MHz by 25. So, the data size is 114.24 × 20 Mbyte/s (16-bit data is transferred with 8b10b conversion). We integrated beam gate control to event timing system by using the distributed bus bit region [4], and some kind of shot information by using data buffer region.

The commissioning of SuperKEKB is going well and there is no critical error in event timing system.

EVENT RECEIVER ON Zynq EVALUATION BOARD

Purpose

Just as additional function has been added in event timing module in SINAP, we might update to suit our accelerator specification. At such time, we need to update firmware easily and quickly. In this situation, it will be important to develop the circuit coding of FPGA ourselves to keep stable accelerator operation. From another point of view, there is an advantage that development costs can be reduced by making it by ourselves. In particular, the difference becomes large when a lot of mass production is performed.

Requirement for the Event Receiver

Since the event code is transmitted in synchronization with the RF clock of 114.24 MHz, including 8b10b conversion, a communication speed of 2.5 Gb/s or higher is required, and a low jitter of less than 20 ps is also required. Therefore, in consideration of cost, we decided to use GTX embedded in Xilinx FPGA. The GTX is included in Kintex-7 in Xilinx 7-series FPGAs, and is also used in MRF 300 series. The SINAP event timing modules are also using GTX transceiver in Virtex-6.

Open Source Event Receiver

MRF released the source code of the FPGA including GTX configuration so that the MRF module’s user could make event receivers by themselves. The released code uses Zynq7000 to describe the circuit. GTX is included in Kintex-7, and Zynq7000 also have a part of equivalent to Kintex-7 in programmable logic part (only for Z7030 and Z7040 series), so GTX can be used. The merit of Zynq is that it can be controlled by the ARM core. This can be a standalone module without going through the bus control, and also can be run in EPICS IOC in Zynq. Since the released code was developed using Avnet’s picozed [5], we decided to purchase a similar board and proceed with development based on open source code.

Figure 1 shows the picture of picozed. It is a dark green substrate in the center of the figure, and is used by inserting it into a carrier card, which is a vermilion substrate. The carrier card model number is AES-PZCC-FMC-V2-G, and picozed model number is AES-Z7PZ-7Z030-SOM-I-G.

GTX Set Up

We constructed GTX configuration with IP core tool of Vivado design suite. At first, I setup using Vivado version of

[*] hitoshi.sugimura@kek.jp

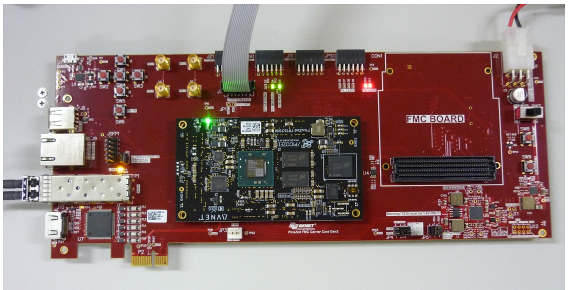


Figure 1: The picture of picozed.

17.4 that is same version of open source in MRF. Then, I also setup the latest version of 19.1. The difference of the version is just the version of IP core tool, so main programmable logic is same. Therefore, I succeed to program in the latest version. The data decoding logic is shown in Fig. 2.

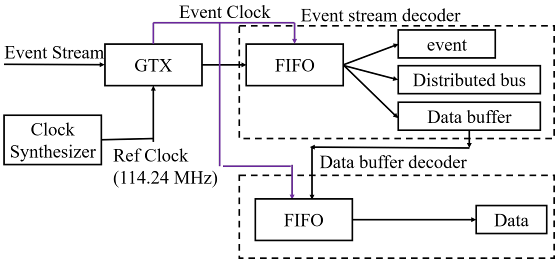


Figure 2: Schematic view of the event data stream.

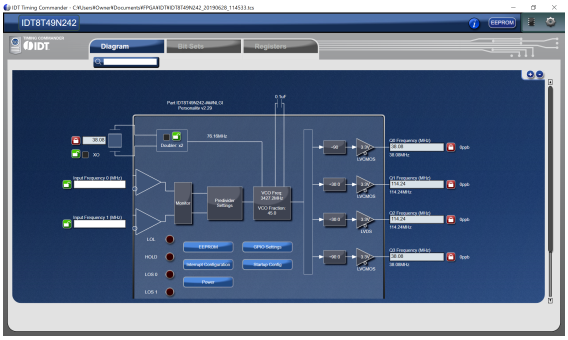
At first, comma alignment is performed using K28.5 sent from EVG as a synchronization event every 4 cycles. If K28.5 is appeared, the data is distributed to event code and data buffer and distributed bus bit in each 16-bit of data frame. The event clock is generated according to reference clock. The input reference clock rate was set to 114.24 MHz using the universal frequency converter (IDT-8T49N242) [6] mounted on the carrier card.

To change reference clock to our RF clock, IDT timing commander tool is used. Since the default reference clock is about 250 MHz, I had to change it. Figure 3 shows the capture of IDT tool window. The source of the reference clock is set at 38.08 MHz and after divide and multiplied the frequency, two of four frequency is set to 114.24 MHz. The configuration is exported to include Vivado, then write to EEPROM (24AA025T). Here, the version of Vivado has to use 2015.4 because the Avnet company is already configured IDT tool in this version. The reference clock setting value is programmed to the frequency converter at power-on.

The data received using GTX is taken into the FIFO created in Block-RAM and distributed to 8-bit for event code and 8-bit for data buffer or distributed bus bit.

Data Buffer

Since the data buffer has a maximum size of 2 Kbytes, reserve that amount in BRAM in advance. Then, K28.2 (0x5C) which is the value of the start of the data buffer is looked up from the sorted 8-bit data. When K28.2 is received, data from the next 8-bit is put into the dual port block RAM, and the process ends with reception of K28.1 (0x3C) which is the value of the end of the data buffer.


IDT Timing Commander - C:¥Users¥Owner¥Documents¥FPGA¥IDT¥IDT8T49N242_20190628_114533.tcs

IDT8T49N242

Part IDT8T49N242-###NLGI
Personality v2.29

38.08
Doubler: x2
76.16MHz
VCO Freq: 3427.2MHz
VCO Fraction: 45.0

Q0 Frequency (MHz) 38.08 38.08MHz 0ppb
Q1 Frequency (MHz) 114.24 114.24MHz 0ppb
Q2 Frequency (MHz) 114.24 114.24MHz 0ppb
Q3 Frequency (MHz) 38.08 38.08MHz 0ppb

Figure 3: Screenshot of IDT timing commander tool.

MONITORING BY LOGIC ANALYZER

Since there is no digital output in picozed and carrier card, output signals cannot be seen. Therefore, by setting up ILA (Integrated Logic Analyzer) provided as an IP core tool, I try to monitor a data buffer and event code reception from EVG on the test bench. Figure 4 shows logic analyzer windows of Vivado design suite. The upper waveform is the event signal monitored using Vivado, and 0x3 after receiving K28.2 (0x5C) is the data buffer value.

In addition, K28.1 (0x3C) is received subsequently. The lower waveform is the obtained waveform by extracting the data buffer from the event signal, and it was confirmed that it was correctly extracted.

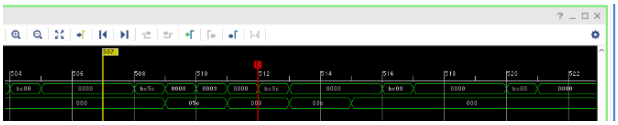


Figure 4: Logic analyzer of Vivado design suite.

FUTURE PROSPECTS

Although it was simple in this evaluation, the event signal was received, and the event code and data buffer were identified and processed. This is the basic part of Event Receiver, GTX, was able to operate correctly, so it was able to handle the minimum functions as Event Receiver. The picozed carrier card does not have a digital output, but if an FMC card with digital output can be inserted, a timing signal will be able to output. Therefore, timing output is the next issue. There are various other evaluations such as jitter measurement of timing signals. Based on these evaluations, I plan to develop a so-called standalone Event Receiver that does not use standards such as VME and μTCA in the future. Furthermore, system development with extensibility such as module development integrated with BPM and RF system will be the next goal.

REFERENCES

[1] SuperKEKB project, http://www-superkekb.kek.jp/

[2] H. Kaji *et al.*, "Upgrade of Event Timing System at SuperKEKB", in *Proc. 14th Int. Conf. on Accelerator and Large Experimental Control Systems (ICALEPCS'13)*, San Francisco, CA, USA, Oct. 2013, paper THCOCA04, pp. 1453–1456.

[3] Micro-Research Finland Oy, http://www.mrf.fi/

[4] H. Sugimura *et al.*, "Injection and Extraction Timing Controls at SuperKEKB Damping Ring", in *Proc. 12th International Workshop on Personal Computers and Particle Accelerator Controls (PCaPAC'18)*, Hsinchu City, Taiwan, Oct. 2018, pp. 201–203. doi:10.18429/JACoW-PCaPAC2018-THP17

[5] PicoZed, http://picozed.org

[6] IDT-8T49N242, https://www.idt.com/document/dst/8t49n242-datasheet