TIME SYNCHRONIZATION FOR DISTANT IOCS OF THE SuperKEKB ACCELERATORS

H. Kaji*, T. Naito, S. Sasaki, KEK, Ibaraki, Japan
Y. Iitsuka, East Japan Institute of Technology Co., Ltd, Ibaraki, Japan

Abstract

The accelerators for scientific research are operated with many distant computers connected via network. Their CPU times are traditionally synchronized by connecting with the NTP server. However more accurate synchronization system is needed if we discuss the time order of alarms issued from different CPU servers. The SuperKEKB accelerator integrates the precise time synchronization system based on the Event Timing System and abort trigger system. The precision of synchronization is remarkably improved. The new time synchronization system is one of the most outstanding advantages during the phase-2 operation of SuperKEKB.

INTRODUCTION

The accelerators for scientific research are extremely large machines and the beam-lines become larger than one kilometer. They are operated with many distant local computers which are connected via network. For example, ~130 CPU modules are separately installed belong 3km beam-lines of the SuperKEKB main rings (MRs) [1, 2].

The time synchronization for this kind of multi-CPUs system is always inconvenient. Usually the CPU times of individual local computers are roughly synchronized with the NTP server. However this precision is sometimes insufficient.

It is impossible to put alarms sequentially at time when they are issued by the different CPU servers. It is quite often happened in the accelerator operation. The single problem affects several hardware components so that alarms are issued from different CPU servers. The operators must investigate which alarm indicates the source of problem without time information.

Here we introduce the time synchronization system developed for and installed into the SuperKEKB accelerators. It consists of two kinds of systems with optical networks.

One is based on the Event Timing System [3]. This system manages the time synchronization of EPICS IOCs [4] at the injector linac (LINAC) [5]. Besides, it synchronizes also the CPU time of master module of the time synchronization system at MR described below.

The other is added into abort trigger system [6] at MR. This system consists of one master module and 36 local modules. The CPU times of local modules are synchronized with that of master module. Therefore we can know the time order of abort signals when the abort trigger system receives more than one abort request signals.

* hiroshi.kaji@kek.jp

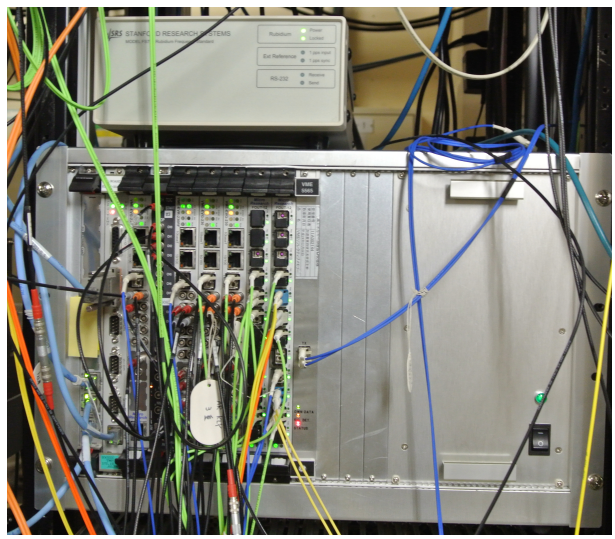

Figure 1: Picture of master IOC of Event Timing System: the master IOC is configured with three EVG and one EVR. The Rb clock, FS725, for the PPS signal is placed on the top of VME sub-rack.

TIME SYNCHRONIZATION BASED ON EVENT TIMING SYSTEM

The time synchronization of CPU modules at LINAC is realized with the Event Timing System. It synchronizes also the CPU time of the master module of time synchronization system at MR. The specifications for both the Event Timing System and its time synchronization function are described in this section.

Event Timing System

The Event Timing System is developed to manage triggers for accelerator operation. One set of Event Generator (EVG) and Event Receiver (EVR) provides many triggers with optional pulse-width and delay. Both EVG and EVR are an FPGA module with a SFP connector. They are connected via dedicated optical cable with each other. The EVG delivers data packet called "Event" to EVR. Then EVR launches several functions like TTL signal outputs and CPU interrupts when it receives Events.

We utilize the VME-EVG-230 and VME-EVR-230RF [3] for EVG and EVR modules, respectively. One of specific features for these modules is time synchronization. The EVR controls the CPU time of the EPICS IOC[1]. This function is synchronized among the EPICS IOCs with EVR by the

[1] Note, the EPICS IOC is configured on the VME64x bus with VME-EVR-230RF and the CPU module of MVME5500.

Table 1: Summary of Event-Code for Time Synchronization

Event-Code	EVR function
0x70	Put "0" to register
0x71	Put "1" to register
0x7D	Set CPU time with register

delivery of Events from EVG. We utilize this function for time synchronization for the SuperKEKB accelerators.

In the real case of accelerator operation, the Event Timing System configures the star-topology optical network and many EVRs are connected on the downstream of EVG at master IOC. Figure 1 is the picture of master IOC for the SuperKEKB accelerators. It is configured with three EVGs and one EVR.

Event-Codes for Time Synchronization

Individual Events have one of 256 kinds of Event-Codes, which is distinguished from "0x00" to "0xFF". We integrate many kinds of functions to EVR, such like the TTL signal output and the CPU interruption, with respect to Event-Codes. These functions are launched when EVR receives Event so that accelerator is operated properly.

Some Event-Codes are assigned the EVR functions by default. They are dedicated for those functions, thus, are not utilized for accelerator operations. The EVR controls the CPU time with this kind of Event-Codes. The Event utilized for time synchronization is summarized in Table 1.

The EVR has the 32-bits of Seconds Shift Register on its memory. The "0x7D" Event sets the CPU time of EVR IOC. The year, month, day, hour, minute, and second are set with the information of Seconds Shift Register. The under one second is determined with the FPGA operation clock. Its "zero" time is defined at the arrival of "0x7D" Event.

The EVR writes down "0" or "1" into the Seconds Shift Register when it receives "0x70" or "0x71" Events. The write is implemented into the least significant bit after throwing the most significant bit and shifting information in all other bits toward one bit left. Therefore the 32 Events are needed to fully rewrite the Second Shift Register.

The information on this register means the UNIX time. It becomes *16:00:00 of May 18th, 2017* if the register is "01011001000111010100011001110000".

Event Delivery System

Our Event Timing System [7–9] has two independent Event delivery systems. One is for the accelerator operation and the other is for the time synchronization.

Figure 2 is the schematic view of Event delivery system for time synchronization. This system provides the set of one "0x7D" Event and 32 "0x70 or 0x71" Events in every second. It is implemented with the PPS signal originated from GPS receiver. This PPS signal is finely stabilized with the Rb clock, SRS FS725 [10], and put into the EVG module.

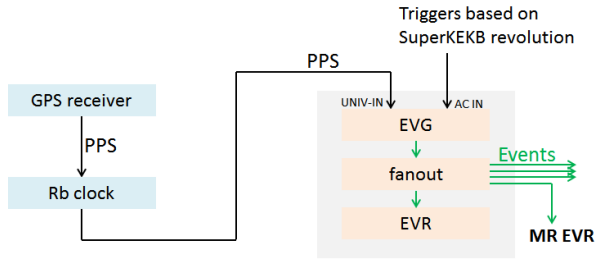

Figure 2: Schematic view of Event delivery system for time synchronization: EVG delivers a "0x7D" Event in every second with the PPS signals from the Rb clock.

The "0x7D" Event is delivered by the trigger-Event function of EVG. It is delivered with the quite precise one second interval since we use the PPS signal for the trigger.

The 32 "0x70 or 0x71" Events are delivered by the software trigger. One of downstream EVRs is installed in the same EPICS IOC. It is programed to launch the CPU interruption and to process the above mentioned software triggers when it receives the "0x7D" Event. The system is well optimized that the EVG delivers 32 "0x70 or 0x71" Events for next synchronization just after delivering "0x7D" Event.

The EVG IOC receives absolute time from the NTP server[2] only once when its EVR receives the first "0x7D" Event after booting. The 32 "0x70 or 0x71" Events are determined with this information. From the second interruption by "0x7D" Event, we calculate absolute time internally and determine corresponding 32 Events set. The synchronization of this process with GPS is guaranteed since the delivery of "0x7D" Events is synchronized with the GPS receiver.

The CPU times of all EVR IOCs are precisely synchronized. Therefore we can compare the times of EPICS processes among different IOCs. Especially the precision of comparison becomes 8.75 ns when the EPICS process is implemented by Events. This is actually one Event clock (FPGA operation clock) of our Event Timing System.

Software for Event Receiver Module

The EVR function to synchronize the CPU time of EPICS IOC has already been integrated into mrfioc2.0.3 device/driver [11]. However we made following customizations for more convenient and stable operation.

We change the behavior of synchronization process when the interval of two "0x7D" Events is slightly delayed from 1 second. It sometimes happens even though the PPS from Rb clock is accurately. The delivery of "0x7D" Event is delayed when it coincide with the Events for accelerator operation. The strange behavior caused by the original source of mrfioc2.0.3 is fixed in our device/driver.

We integrate the function to add offset to the absolute time. We can set the offset for individual EVR IOCs. This function is used for adjusting the difference of optical cable lengths between EVG and EVRs.

[2] Actually, the GPS receiver works as the NTP server. We use Japan Standard Time as the absolute time.

ISBN 978-3-95450-182-3

Figure 3: Picture of the 8ch optical input module.

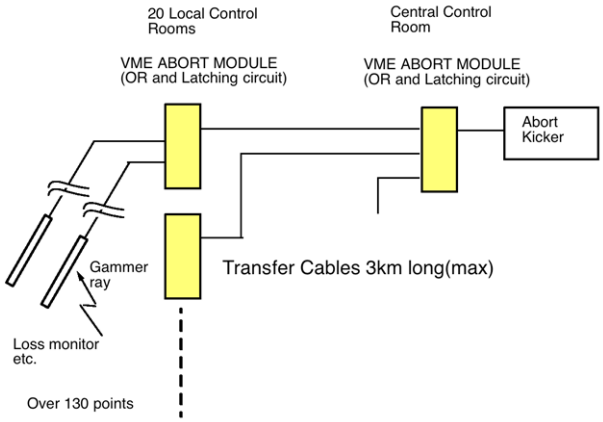

Figure 4: Schematic view of abort trigger system: the system configures the cascade topology network. It consists of 36 lower-level modules installed at Local Control Rooms and one master module installed at Central Control Room.

TIME SYNCHRONIZATION BASED ON ABORT TRIGGER SYSTEM

The abort trigger system at the SuperKEKB MR has time synchronization function. The abort trigger system and its function to synchronize the abort request time are introduced in this section.

Abort Trigger System

The abort trigger system collects more than 130 abort requests from individual hardware of MR. The abort request signals are firstly put into the 2ch E/O modules [6]. The several kinds of abort request signals, like TTL, RS422, or relay, from individual hardware are converted into optical signal on the 2ch E/O modules. Then those output signals are collected with the VME-based 8ch optical input modules.

Figure 3 is the picture of the 8ch optical input module. It collects 8 optical abort signals and output one optical signal with the "OR" logic. The output signal is kept until software reset is implemented.

The system configures the cascade topology network as shown in Figure 4. The 36 lower-layer modules are installed at 20 Local Control Rooms (LCRs). They collect the abort request signal at the individual areas and deliver output into the Central Control Room (CCR). The master module of abort trigger system is installed at CCR and it collects abort request signal from individual LCR modules. The trigger for the abort kicker magnet is generated and delivered with the output from the master module.

Time Synchronization

The 8ch optical input module has the timestamp function based on two kinds of internal counters. One is 10 MHz of internal clock counter which is synchronized with the FPGA operation clock. The internal clock counter is incremented automatically in 10 MHz and is reset when the module receives an external clock. The other is the trigger pulse counter and it is counted up when the module receives the external clock. All modules at both CCR and LCRs receive the external triggers from the software timing system [12]. Therefore the internal clock counters and trigger pulse counters are synchronized among the 8ch optical input modules.

This module latches timestamp from two counters when it receives an abort request signal. Then the absolute time of abort request is determined with the software processing. The CPU time of master IOC is adopted as the absolute time of abort trigger system.

The result of performance test for time synchronization is shown in Ref [6]. The absolute time when the individual modules receive the abort request signals can be determined with the precision of 100 ns. It is consistent with 10 MHz of the FPGA operation clock.

CONCLUSION

We develop time synchronization system for SuperKEKB. It consists of the Event Timing System and the abort trigger system at MR.

The Event Timing System synchronize the CPU time of EPICS IOC at LINAC. It also synchronizes the master module of the abort trigger system so that EPICS IOC at both LINAC and MR are synchronized. The time of EPICS process can be compared among EVR IOCs. The precision of comparison becomes 8.75 ns when the process is implemented by the Event.

The internal counters of individual modules of abort trigger system are synchronized. The absolute time when the individual modules receive the abort request signals can be determined as the CPU time of master IOC at CCR. Its precision is 100 ns. This can be compared with also the CPU time of EVR IOCs at LINAC.

The precision of time synchronization becomes remarkably improved from that based on the NTP server. We conclude the new time synchronization system is one of the most outstanding advantages during the phase-2 operation of SuperKEKB.

REFERENCES

[1] Y. Ohnishi *et al.*, "Accelerator design at SuperKEKB", *Prog. Theor. Exp. Phys.*, vol. 2013, p. 03A011, 2013.

[2] K. Abe *et al.*, "Letter of intent for KEK Super B Factory", KEK, Tsukuba, Japan, Rep. KEK 20014-4.

[3] MRF, `http://www.mrf.fi/index.php/vme-products`

[4] EPICS, `http://www.aps.anl.gov/epics/`

[5] M. Akemoto *et al.*, "The KEKB injector linac", *Prog. Theor. Exp. Phys.*, vol. 2013, p. 03A002, 2013.

[6] S. Sasaki *et al.*, "Upgrade of abort trigger system for SuperKEKB", in *Proc. ICALEPCS'15*, Melbourne, Australia, Oct. 2015, pp. 417–419.

[7] H. Kaji *et al.*, "Installation and commissioning of new event timing system for SuperKEKB", in *Proc. 12th Annual Meeting of PASJ*, Tsuruga, Japan, Aug. 2015, pp. 223–227.

[8] H. Kaji *et al.*, "Upgrade of event timing system at SuperKEKB", in *Proc. ICALEPCS'13*, San Francisco, CA, USA, Oct. 2013, pp. 1453–1456.

[9] H. Kaji *et al.*, "Construction and commissioning of event timing system at SuperKEKB", in *Proc. IPAC'14*, Dresden, Germany. Jun. 2014, pp. 1829–1832.

[10] Stanford Research Systems FS725 manual, `http://www.thinksrs.com/downloads/PDFs/Manuals/FS725m.pdf`

[11] mrfioc2, `http://epics.sourceforge.net/mrfioc2/`.

[12] T. Naito *et al.*, "Performance of the timing system for KEKB". in *Proc. ICALEPCS'99*, Trieste, Italy, Oct. 1999, pp 214–216.

ISBN 978-3-95450-182-3