Abstract
RAON is a particle accelerator to research the interaction between the nucleus forming a rare isotope as Korean heavy-ion accelerator. RAON accelerator consists of a number of facilities and equipments as a large-scale experimental device operating under the distributed environment. For synchronization control between these experimental devices, timing system of the RAON uses the VME-based EVG/EVR system. This paper is intended to test high-speed device control with timing event signal. To test the high-speed performance of the control logic with the minimized event signal delay, we are planning to establish the test motor controller testbed applying the FPGA chip. The tested controller will be configured with Zynq 7000 series of Xilinx FPGA chip. Zynq as SoC (System on Chip) is divided into PS (Processing System) and PL (Programmable Logic). PS with the dual-core ARM cpu is performing the high-level control logic at run-time on linux operating system. PL with the low-level FPGA I/O signal interfaces with the step motor controller directly with the event signal received from timing system. This paper describes the content and performance evaluation obtaining from the step motor control through the various synchronized event signal received from the timing system.

The RAON Introduction
The RAON[1] is a new heavy ion accelerator under construction in South Korea, which is to produce a variety of stable ion and rare isotope beams to support various researches for the basic science and applied research applications. To produce the isotopes to fulfill the requirements we have planned the several modes of operation scheme which require fine-tuned synchronous controls, asynchronous controls, or both among the accelerator complexes.

Timing System of RAON
- Characteristics of MRF Timing System[2]:
  - Event Driven System, to 256 event codes
  - External RF Reference Clock
  - 50 - 120 MHz Frequency
  - Multi Counters
  - Event Cascaded
  - Different Clock Synchronization

Timing System Prototype

Application using Timing Signal
- System on Chip:
  - FPGAs - Zyng(3, 4)
  - Zyng as SoC is divided into PS and PL
  - Interface between PS and PL is through the AXI of AMBA[5]
- ZC706 Evaluation Board[6]
- Linux on Zynq PS (ARM Processor)
- ARM Cross Compile Tool Chain (arm-linux-gnueabihf)
- Linux Kernel Source (Linaro[7])
- Bootloader (BOOTBIN: FSBL.elf, U-Boot.elf, ulimage, Zyng.elf, User.elf[8, 9])
- Board Support Package (Linux Device Tree)
- Root File System (Busybox[10])
- Linux Device Driver
  - Industrial I/O linux device driver of Analog Devices[11]
  - Libix-bib[12]
- Software Interface
  - EPICS Base R3.14.15.2 : IOC developed using Libio
  - FPGA Verilog HDL by Vivado[13] (Used the HDL code of Analog Devices)

Controller / Low Voltage Drive Board
- Controller and Low Voltage Drive Board of Analog Devices
- Controller communicates with ZC706 via FPGA Connector
- Power Connection: XADC, Digital I/O, 2x16 Ethernet
- XADC and digital I/O signals to LVDB
- FPG Signal Generation through MOSFET Gate Drivers
- External Power: 12 - 24 DC to LVDB

Test Stepper Motor
- Not fully implemented, still is developing FPGA code to receive the timing EVR input signal
- Left shows to drive stepper motor using EPICS interface
- Right shows the connection for timing signal between EVR and ZC706

Summary
- Data Acquisition
  - XADC and digital I/O signals to LVDB
  - EVG/EVR system
- Testbed 
  - Linux on Zynq PS (ARM Processor)
  - ARM Cross Compile Tool Chain (arm-linux-gnueabihf)
  - Linux Kernel Source (Linaro[7])
- Software Configuration
  - EPICS on Xilinx SoC
  - XADC and digital I/O signals to LVDB
  - Controller communicates with ZC706 via FPGA Connector
- Hardware Configuration
  - Zyng SoC with PS and PL
- Interface between PS and PL is through the AXI of AMBA[5]

Application using Timing Signal
- System on Chip: FPGAs - Zyng[3, 4]
  - Zyng as SoC is divided into PS and PL
  - Interface between PS and PL is through the AXI of AMBA[5]
- ZC706 Evaluation Board[6]
  - Linux on Zynq PS (ARM Processor)
  - ARM Cross Compile Tool Chain (arm-linux-gnueabihf)
  - Linux Kernel Source (Linaro[7])
  - Bootloader (BOOTBIN: FSBL.elf, U-Boot.elf, ulimage, Zyng.elf, User.elf[8, 9])
  - Board Support Package (Linux Device Tree)
  - Root File System (Busybox[10])
- Linux Device Driver
  - Industrial I/O linux device driver of Analog Devices[11]
  - Libix-bib[12]
- Software Interface
  - EPICS Base R3.14.15.2 : IOC developed using Libio
  - FPGA Verilog HDL by Vivado[13] (Used the HDL code of Analog Devices)

Conclusion
Timing system can distribute the fine synchronized event signal at a high speed. The objective of the stepper motor control testbed is to know how to operate the timing system and how to apply it to the high speed controller. The overall implementation is still undereway, however if the distributed control system using EPICS makes a connection with the high speed parallel processing of FPGA, it is possible to improve the performance and efficiency of the control system. Zyng SoC can be considered as an ideal device to implement this high-speed control system.

Acknowledgement
This work is supported by the Rare Isotope Science Project funded by Ministry of Science, ICT and Future Planning (MSIP) and National Research Foundation (NRF) of KOREA.

References
http://an.wikipedia.org/wiki/Advanced_Microcontroller_Bus_Architecture
[6] ZC706 Evaluation Board Document:
[7] Linaro Document website:
https://en.wikipedia.org/wiki/Linaro
[8] Boot Image Document website:
[12] Industrial I/O Document website:
[13] Xilinx Vivado Document website:
http://www.xilinx.com/products/design-tools/vivado.html