# Application using timing system of RAON accelerator

Sangil Lee silee7103@ibs.re.kr, C.W. Son scwook@ibs.re.kr, H.J. Jang lkcom@ibs.re.kr,
*Rare Isotope Science Project, Institute for Basic Science, Daejeon, South Korea*

## 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-scaled 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 planing to establish the step motor controller testbed applying the FPGA chip. The testbed 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.

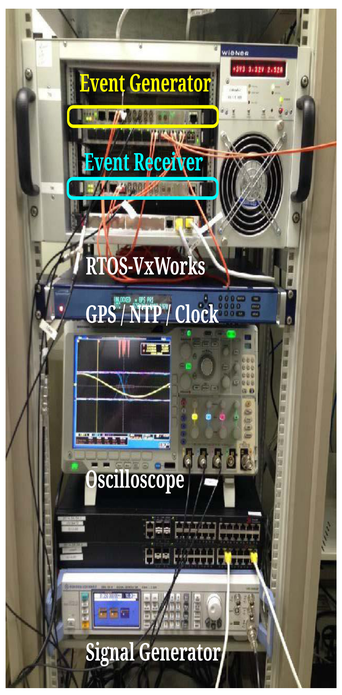
## The RAON Introduction

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 planed the several modes of operation scheme which require fine-tuned synchronous controls, asynchronous controls, or both among the accelerator complexes.

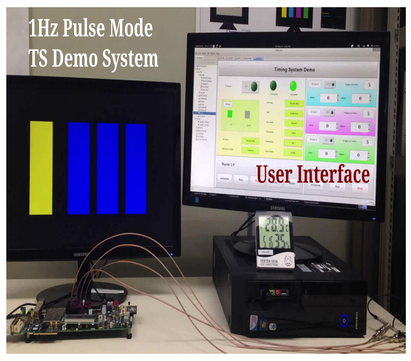
## Timing System of RAON

- 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

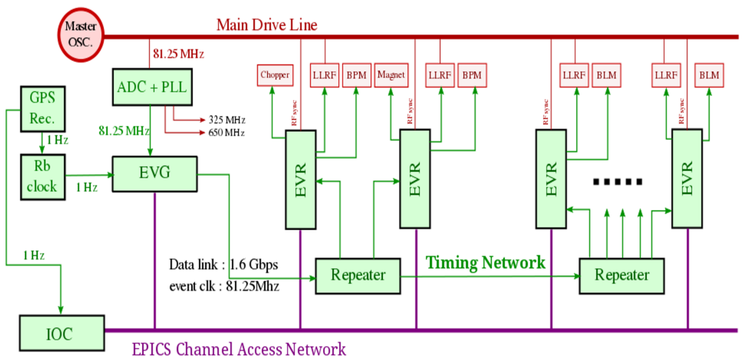






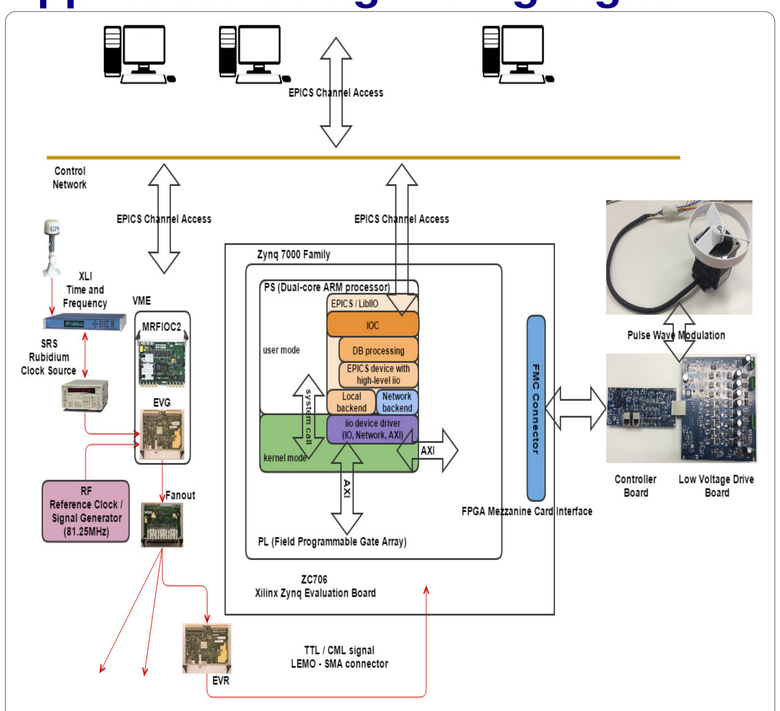

- Hardware Configurations :
  - XLi GPS Time System
  - Rubidium Frequency Standard Clock Source (FS725)
  - Event Trigger System (EVG/EVR, Fan-out Repeater)
  - MVME 6100 / MVME 3100 Controller
  - SMA 100a RF Signal Generator
  - VME Wiener Crate
- Software Configurations :
  - Workbench 3.3, VxWorks IDE
  - VxWorks 6.9 Real-time OS (on MVME 6100, EVG)
  - RTEMS Real-time OS (on MVME 3100, EVR)
  - EPICS framework (R3.14.15.2)
  - MRFIOC2 / SRSIOC
  - Network Time Protocol (NTP)

## Timing Network



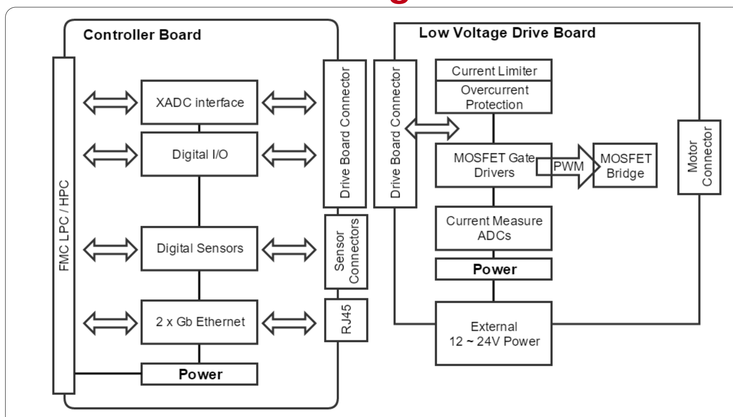

- GPS receiver synchronizes with RB clock and EVG to 1PPS
- Master OSC. generates reference RF (81.25 MHz) to EVG
- Two signals of EVG are synchronized by locking phase
- Synchronized signal of EVG is generated and distributed to EVR according to event code of MRFIOC2 on VME system

## Application using Timing Signal



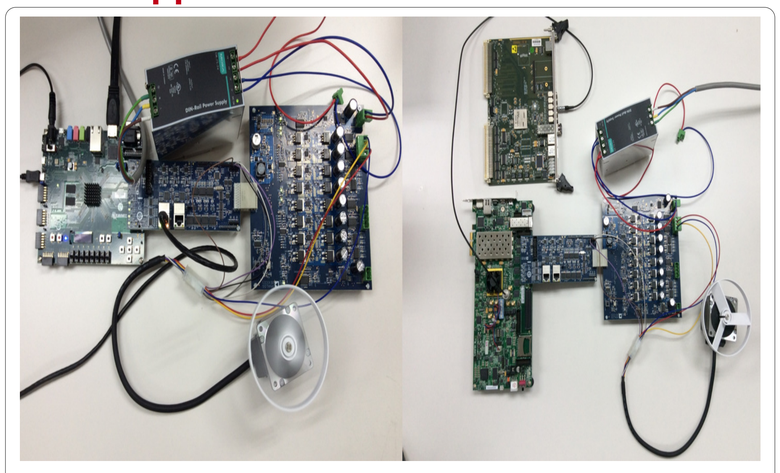

- System on Chip: FPGA - Zynq[3, 4]
  - Zynq as SoC is divided into PS and PL
    - Interface between PS and PL is through the AXI of AMBA[5]
- ZC706 Evaluation Board[6]
- Linux on Zynq PS (ARM Processor)
  - ARM Cross Compile Tool Chain (arm-linuxgnueabihf)
  - Linux Kernel Source (Linaro)[7]
  - Bootloader (BOOT.BIN: FSBL.elf, U-Boot.elf, uImage, Zynq.bif, User.bit)[8, 9]
  - Board Support Package (Linux Device Tree)
  - Root File System (Busybox)[10]
- Linux Device Driver
  - Industrial IO linux device driver of Analog Devices[11]
  - Libiio library[12]
- Software Interface
  - EPICS Base R3.14.15.2 : IOC developed using Libiio
  - FPGA VerilogHDL 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 FMC Connector
- FMC Connection: XADC, Digital I/O, 2xGB Ethernet
- XADC and digital I/O signals to LVDB
- PWM Signal Generation through MOSFET Gate Drivers
- External Power: 12 ∼24 DC to LVDB

## Test Stepper Mortor



- 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

| Hardware | Contents | Company |
|---|---|---|
| GPS | GPS antena<br>XLi time and frequency | Symmetricom |
| Clock Sync. | Rubidium frequency standard model FS725 | Standford Research |
| Ref. Clock | Signal Generator or RF Reference (81.25 MHz) | |
| EVG | Event Generator | Micro Research Finland Oy |
| EVR | Event Receiver | |
| FanOut | Event Repeater | |
| MVME6100 | VME Controller | |
| ZC706 | Zynq 7000 family Eval. kit<br>Processing System (dual-core ARM processor)<br>Progrmmable Logic | Xilinx |
| Controller | Motor controller board | Analog Devices |
| Low Volt. | Low voltage drive board | |
| Motor | Stepper or BLDC | |

| Software | Contents | Module |
|---|---|---|
| EPICS | Base R3.14.15.2 | PS of Zynq |
| Libiio | Industrial I/O library supplied Analog Devices | |
| Kernel | Linaro kernel source including iio device driver | |
| Bootloader | U-Boot for zynq | |
| IOC | In-house using Libiio | |
| FPGA | Analog Devices and In-house code | PL of Zynq |
| VxWorks | Real-time OS for VME including BSP | Timing |
| MRFIOC2 | EPICS IOC for timing | |
| SRSIOC | IOC for rubidium clock | |
| Workbench | Workbench3.3 for VxWorks | Development Tool |
| Vivado | Vivado14.4 including SDK with node lock license | |
| Busybox | for rootfs system (free) | |
| Toolchain | for ARM cross-compile (GNU) | |

## 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 underway, 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. Zynq 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

[1] Y. K. Kwon, *et. al*,“Status of Rare Isotope Science Project in Korea”, Few-Body Syst 54, 961-966, (2013).

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

[3] Xilinx-FPGA website: http://www.xilinx.com

[4] Zynq-Book Document website: http://www.zynqbook.com

[5] Advanced Microcontroller Bus Architecture http://en.wikipedia.org/wiki/Advanced_Microcontroller_Bus_Architecture

[6] ZC706 Evaluation Board Document: http://www.xilinx.com/support/documentation/boards_and_kits/zc706/ug954-zc706-eval-boardxc7z045-ap-soc.pdf

[7] Linaro Document website: https://en.wikipedia.org/wiki/Linaro

[8] Boot Image Document website: http://www.wiki.xilinx.com/Prepare+boot+image

[9] U-Boot Document website : http://www.denx.de/wiki/U-Boot

[10] Busybox Document website : http://www.busybox.net/

[11] Analog Devices website: http://www.analog.com

[12] Industrial I/O Document website: https://wiki.analog.com/resources/tools-software/linux-software/libiio

[13] Xilinx Vivado Document website: http://www.xilinx.com/products/design-tools/vivado.html