VME Configurations and Test for a Digital Control System of A 120-keV Helium Ion RF Implanter**

Ying-Min Li [ab*], Young-Gi Song [b], Sung-Su Cha [ab], Ji-Ho Jang [b], Hyeok-Jung Kwon [b], Han-Sung Kim [b], Kyung-Tae Seol [b], Yong-Sub Cho [b]

[a] *The Accelerator and Beam-nano Engineering, University of Science and Technology, 176 Gajung-dong, 217 Gajungro Yuseong-gu,305-350, Daejeon, Korea*

[b] *Proton Engineering Frontier Project, Korea Atomic Energy Research Institute, Daedeok-daero 989-111, Yuseong-gu, Daejeon, Korea*

[*] *Corresponding author: liyingmin1979@kaeri.re.kr*

1. Introduction

Proton Engineering Frontier Project (PEFP) has developed a 120-keV RF implanter for used as heavy ion implantation. In case of its working environment, there are some unexpected disturbances, so that the cavity accelerating gradient cannot always work at the peak point. In order to maximize the RF field, we designed a digital control system for this RF implanter. It is a Field Programmable Gate Array (FPGA) based feedback loop to keep the resonance of the cavity. The basic control logic includes a Proportional Integral (PI) compensator. And also there is a VME PowerPC SBC which is used as the carrier board to hold the FPGA processor card. We also use a Host system which running a boot image server for the VME SBC. It also provides a reverse remote data storage. The real physical system consists of the cavity and amplifier. In this paper, we mainly talked about system architecture, FPGA control logic and VME configuration and API test. [1-3]

2. The Design Concepts

In this section we describe the reason why PEFP needs a frequency control system, and benefits of digital schemes, and also explains the FPGA digital control system design. Finally we give a signal flow.

2.1 Motivations

Because PEFP RF Implanter was placed at a normal-temperature environment, and the ambient parameters were not controlled. Such as the ground vibrations, pump vibrations and ambient temperature. These unavoidable disturbances are seriously reduced the stability of cavity resonance frequency. Thus it can decrease the accelerating gradient of RF implanter's cavity. That is the reason for us to control the resonance for the cavity.

2.2 The Possible Resolved Schemes

For classic cavity frequency control system, (1) there are tuner-based IQ control system. And/or (2) also a variable RF source is tracking the resonance of the cavity. For the electronics of the compensator usually occurred in classic design, there are analog and digital design.

Fig. 1 shows the signal flows of our designs of frequency control. In our scheme, we choose the variable RF source and a FPGA based digital system to achieve goals. An Agilent E4438C is supporting the 52 MHz sampling clock, and SRS 535 /or BNC 565 pulse generator triggers FPGA system. The VME SBC hosted the FPGA processor card (PMC). We use a Windows XP host to develop/stationed VxWorks 5.5.1 RTOS boot image/API program for booting the VME system.

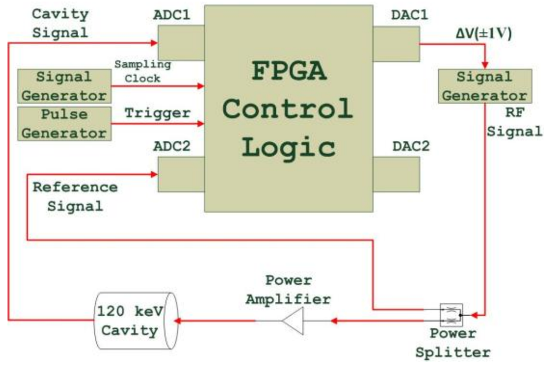

Fig. 1 Signal Flows of The RF Implanter's Control System

3. The FPGA Control Logic and VME Configurations

In this section the key control logic and VME configurations and API test are shown. Before the VME/FPGA system is working, Host was transferring the VxWorks image and API code to VME computer's RAM in order to let FPGA processor card working.

3.1 FPGA Control Logic and Its Functions

The key part of the control system resided in the FPGA card, the logic is shown in Fig. 2. The FPGA processor XC2V4000 contains the custom signal processing logic. From the incident reference and cavity signals, FPGA logic directly sampled both RF signals in an equal-time interval pattern-style with 52-MHz timing, and making the Q, I –Q, -I… sequence. And then we use a rotation matrix to minus the phase difference induced due to mismatched cable lengths.

Next we use CORDIC algorithms to obtain their phases, and compared both of them to obtain the difference. And then send it to a PI compensator. Finally we output the loop error signal to DAC. The converted signal goes into an external Signal Generator. So as to produce a proper RF signal, its frequency is identical with the resonance frequency of cavity.

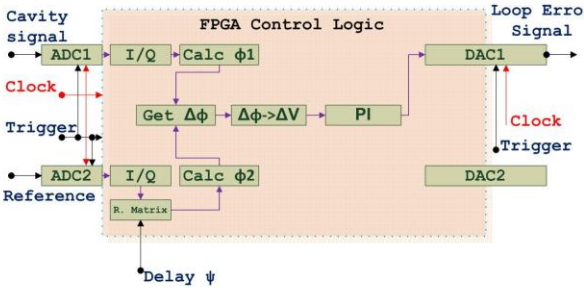

Fig. 2 The FPGA Control Logic

A host computer support reversed user data storage for FPGA logic through PCI (VME/PMC) and TCP/IP (Host/Target), such as the sampled I/Q data. The control logic was managed by API through PCI while the control system is running. Initially API will program PROM with a Xilinx FPGA configuration file *.exo which is made by us.

3.2 The VME PowerPC SBC Configuration and Test

In Fig. 3, VxWorks is running from VME's RAM (target), and then load API module from host. The FPGA card was triggered by a 20 ns LVTTL signal. Then the control logic was excited. And VME/VxWorks API is always monitoring the status of FPGA card through PCI bus. In PEFP, earlier time we use a Sun Ultra5/360 workstation with Solaris OS, in which stationed image/API codes. Nowadays we use a Windows XP x86 PC to do the same task. The procedure for VME configurations are:

1. Run Host development tools (Tornado 2.2.1 or Workbench 2.5). Develop/Build the user application/API/driver (to compile and link with MV-5100 BSP) in Host.
2. In the Host, select/configure desired VxWorks components and then builds the VxWorks boot image.
3. In the Host, set-up a boot image FTP server, and also run a telnet program for booting the target hardware.
4. Powering up Emersion® MVME-5100 PowerPC SBC. Logon to telnet server, and wait for VxWorks booting process, in case set the appropriate boot parameter.
5. Configure the target server in Host and download the API to MVME-5100.
6. From telnet, run the API and check the running result.

Under the circumstance, we made a small API program just to examine our VME SBC configuration and function. For MVME-5100, because PowerPC has a ±32MB limited relative branch range. Our system has installed 512 MB SDRAM de facto. API code may load in memory that is > 32 MB from the kernel routines that it needs to link against. So for correctly building API code, we might as well use the "mlongcall" switch to convert function calls going outside of a module into the multi-instruction sequence necessary to make a full 32-bit address branch.

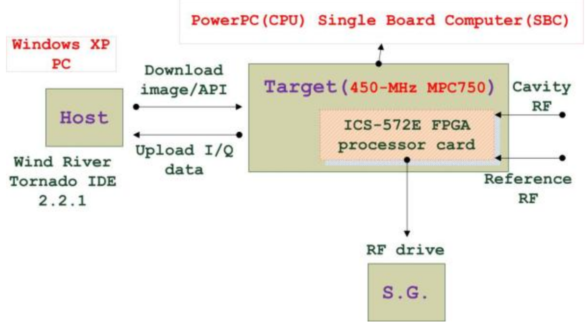

Fig. 3 Classical VME PowerPC SBC Configurations for Digital Control System Platform

Under both of the Wind River® shell and the real VME computer (through terminal), we examined the configuration and API function. It works well. Fig. 4 is the real system we used in test.

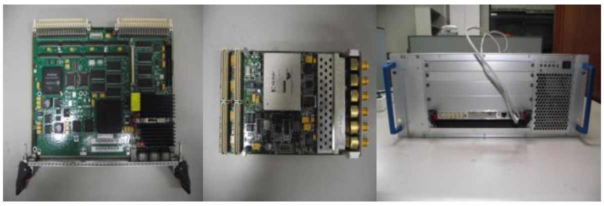

Fig. 4 The Physical Hardware used in VME Configurations

4. Conclusions

In this paper, we studied a control system to keep the invariant resonance condition of the cavity for PEFP RF Implanter. The FPGA logic was given and VME system was correctly configured and a simple demo API can run over it without problem.

**Work supported by Ministry of Education, Science, and Technology of the Korean Government through PEFP.

REFERENCES

[1] Young-Gi Song, Han-Sung Kim, et al., EPICS-based Software Development of a Low-Level RF system for the PEFP 100-MeV Proton Accelerator, Vol. 59, No. 2, August 2011, pp.577~580

[2] Kyung-Tae Seol, Hyeok-Jung Kwon, et al., Improvement of the Low-level RF Control System for the PEFP 100-MeV Proton Accelerator, Vol. 59, No. 2, August 2011, pp.627~630

[3] Ying-Min Li, Sung-Su Cha, et al., The Design and Test of the Analog Frequency Control System for Suppressing the 750Hz Single Frequency Disturbances for the PEFP 13-MHz RF Implanter, Journal of Korea Physical Society, to be published soon.