DESIGN AND IMPLEMENTATION OF STEPPER MOTOR CONTROL OF THE LINAC HIGH POWER RF SYSTEM BASED ON FPGA

R. Rujanakraikarn†, C. Dhammatong, W. Phacheerak
Synchrotron Light Research Institute, Nakhon Ratchasima, Thailand

Abstract

In this paper, the new motion control system that governs the position of high power attenuators and phase shifters in the linac's RF system at SLRI is described. The drive system, which was originally driven by a set of AC reversible motors, is replaced by a new set of stepper motors. The hardware selection and installation is presented in detail. The digital control circuits are designed in VHDL and implemented on a commercial Field Programmable Gate Array (FPGA) board. The main software part, implemented in MicroBlaze Microcontroller System (MCS), is coded in C to control the position of stepper motors relative to the DC voltage reference points of the hardware system. A LabVIEW GUI is designed to interface with the control system to provide reference points and display position values via RS-232 and PLC interfaces. This stepper motor control system can be used to effectively implement the phase and amplitude control system of the linac's RF signals in the future.

INTRODUCTION

Synchrotron Light Research Institute (SLRI) has been operating its synchrotron facility complex called SIAM Photon Source (SPS) to serve various research activities since December 2001. Its 40-MeV linac has been regularly operated only twice a day, with an approximate duration of one hour each, for electron injection. The linac consists of five different parts to accelerate the electron beam. These parts are pre-buncher 1, pre-buncher 2, buncher, a 20-MeV accelerating tube 1, and a 20-MeV accelerating tube 2 [1]. The phase and amplitude of a 2,856 MHz pulsed-RF are controlled manually by adjusting the high power phase shifters and attenuators, respectively. The mechanical parts of these high power elements consist of seven piston rods along the RF distribution waveguides depicted in a diagram in Figure 1. They are operated as controlling actuators for adjusting the phase shifters and amplitude attenuators. They had been driven by reversible AC induction motors since the beginning of the first light.

Even though the phase and amplitude of the RF signal along the waveguides can be adjusted, the actual numerical values of these RF parameters are not directly measured. During the normal operation, the position of each of the piston rods; i.e., the position of each of the phase shifters and amplitude attenuators, is referenced to the DC voltage across potentiometer in a motor drive circuit. This voltage is displayed at the electronic front panel in a control room as shown in Figure 2. Each day this set of values is recorded as RF operating point for beam injection and future reference. The operating point is occasionally changed if beam injection problem occurs or machine parameters are needed to be changed.

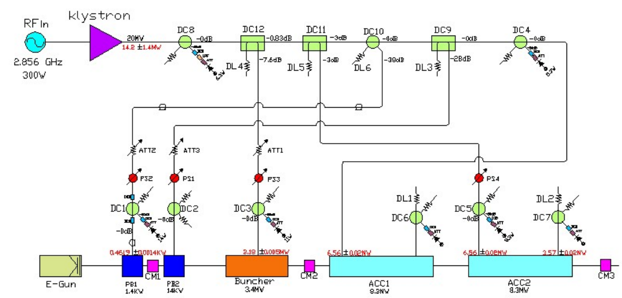

Figure 1: Linac RF distribution diagram.

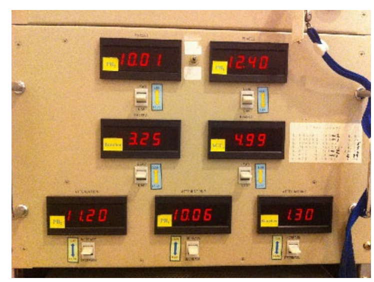

Figure 2: Electronic front panel display of the phase shifters and amplitude attenuators as DC voltages.

In this paper, new hardware selection and installation is described in the next section. Digital control circuit design of the system on FPGA board is also included. Software implementation and GUI are presented in the following section. System performance and conclusion are presented at the end of this paper.

HARDWARE

Stepper Motors, Electronic Drivers and Controllers

During annual shutdown in 2013 all reversible AC induction motors driving the high power phase shifters and amplitude attenuators were replaced by commercial stepper motors and their electronic controllers. The selected stepper motors and controllers are Oriental Motor's RK Series with Pulse Input Type and RKII Series with Built-in Controller Type, depending on the size and location of the phase shifters and amplitude attenuators in the linac system. The new motors were carefully chosen to meet mechanical characteristics and intended to provide better position control of the phase shifters and amplitude attenuators. Figure 3 shows an example of the motor replacement in the waveguide system. All electronic motor drivers and

† roengrut@slri.or.th

controllers including power supplies are installed in a standard equipment rack as shown in Figure 4.

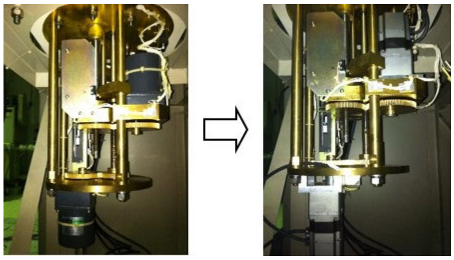

Figure 3: Stepper motor replacement for one phase shifter and one amplitude attenuator in the waveguide system.

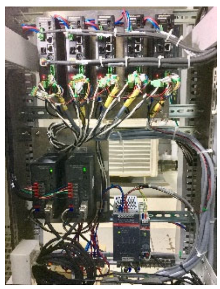

Figure 4: Installation of electronic drivers and controllers of stepper motors in a standard equipment rack.

In order to control the selected stepper motors, appropriate electrical signals must be generated to interface with their commercial electronic controllers. For the Pulse Input Type controllers [2], at least a motor direction signal and a pulse signal with specific frequency must be generated. For the Built-in Controller Type controllers [3], at least start and stop signals are needed. Other output signals from the motors, for example, Ready, Move, and Timing are considered for position feedback and motor rotation check purposes.

FPGA-Based Control Implementation

In this development, FPGA evaluation board is our choice for fast digital system implementation. Xilinx's ML605 Evaluation Kit [4] featuring a Virtex-6 XC6VLX240T FPGA is chosen to be a main controller of the design. Because of a number of signal interface wires to all seven stepper motors, several I/O pins of the FPGA board are needed. Xilinx's FMC XM105 Debug Card [5] is our choice for this purpose. Installation of the FPGA board, the debug card, and opto-isolator modules in a standard equipment box is shown in Figure 5.

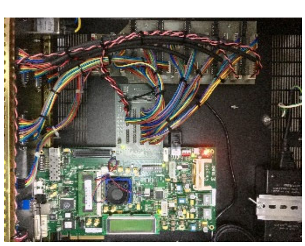

Figure 5: Virtex-6 ML605 FPGA board and FMC XM105 Debug Card daughter board connected to opto-isolator circuits.

In the FPGA, several digital circuit blocks, for example, frequency divider, frequency counter, comparator, multiplexer, and de-multiplexer, are designed using VHDL [6] in Xilinx's ISE 14.7 IDE. They are carefully implemented and interconnected in order to perform pulse processing tasks in this stepper motor control system. In this particular system, the frequency of the pulses for all seven stepper motors are fixed, depending on the types of motors and their controllers. Once the number of pulses and the motor IDs are specified by a LabVIEW GUI, appropriate signals are generated in this digital circuit part of the FPGA and sent to stepper motors through XM105 Debug Card. To complete this embedded control system, the main controller of the design is a MicroBlaze Micro Controller System (MCS). It is a complete standalone processor system intended for controller applications. Also, it is highly integrated which includes the MicroBlaze processor, local memory, as well as a tightly coupled IO module [7].

Several input and output signals are required to provide complete interface with all seven stepper motors. A complete list of these signals connecting MicroBlaze MCS, digital circuit blocks, and electronic drivers and controllers in this design is shown in Table 1.

Figure 6 shows the digital circuit part for determining motor direction and generating pulse signals with specific number of pulses for the motors with Pulse Input Type controllers. Motor start/enable and stop signals, together with specific number of pulses needed, for the motors with Built-in Controller Type controllers are generated in a similar fashion.

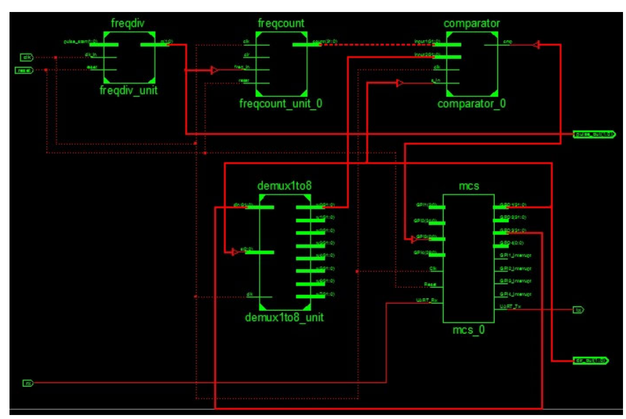

Figure 6: Digital circuit for generating direction and pulse signals.

Table 1: General Purpose Input and Output Signal

Port(bits)	Signal Description
GPI1(3,…,0)	Switches on ML605 board
GPI2(31,…,0)	32-bit value from Frequency Counter
GPI3(0)	Frequency Counter Done Flag
GPI4(29,…,0)	Motor Output Signals

GPO1(31,…,0)	32-bit Motor Control Signals (Motor0, Motor1, Motor2, Motor3, Motor4)
GPO2(31,…,0)	32-bit Motor Control Signals (Motor5, Motor6, clear data)
GPO3(31,…,0)	32-bit value of Number of Pulses
GPO4	Not used

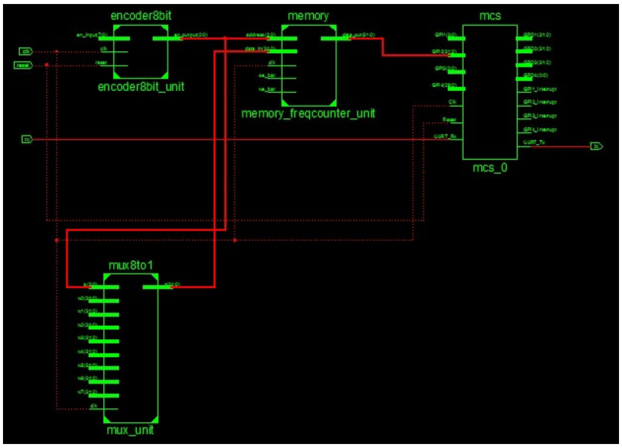

Figure 7: Digital circuit for receiving the number of pulses from GUI and storing in a memory for each motor.

After user determines motor position values through GUI, which are referred to the DC voltage across potentiometers, the calculated numbers of pulses are sent to the FPGA board and stored in the memory for further processing in the MicroBlaze MCS. Figure 7 shows the digital circuit part of this design.

SOFTWARE

The software of the system can be separated into two parts, LabVIEW GUI and MicroBlaze MCS software. The GUI part is responsible for receiving the position and direction setpoints, and displaying the current position of all phase shifters and amplitude attenuators. The second part on the MicroBlaze MCS is written in C. Its main purposes are to receive the setpoints from the GUI for further processing and to communicate with other digital circuit blocks in the FPGA logic fabric.

LabVIEW GUI

Figure 8 shows the main LabVIEW GUI of the system. User can determine the desired stepper motor position as the DC voltage through this GUI by either manually entering numeric setpoint, or clicking the arrow buttons to increase or decrease the desired position value of each of the phase shifters and amplitude attenuators. Each desired setpoint is compared to the current position of each motor. The rotation direction, either clockwise or counter-clockwise, is determined, and the number of needed pulses are calculated.

Figure 8: LabVIEW GUI of the system.

The pulse calculation of each of the motor is highly dependent on the hardware and software tests and parameter settings. A full-scale traveling distance of each motor from the minimum value to the maximum value of the DC voltage defines the maximum number of pulses. The full-scale traveling distance of all seven motors were carefully recorded. Several trials were performed to obtain the best full-scale reading. This gives us all necessary maximum number of pulses for all motors, thus providing the crucial information for pulse calculation. Once the calculation is finished, motor IDs, rotation direction, and number of pulses are sent to the FPGA board via UART. In addition, the current motor positions, which are displayed as numeric values on the screen, are obtained from PLC. Figure 9 shows program flowchart of the LabVIEW GUI.

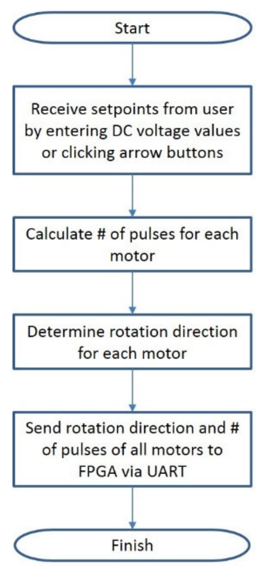

Figure 9: Program flowchart of LabVIEW GUI.

MicroBlaze MCS Software

The control software part on the MicroBlaze MCS is written in C using Xilinx's SDK. The main purpose of this implementation is to perform decision making on position control. The input and output signals of the MicroBlaze MCS are interfaced via the GPI and GPO ports, respectively, as presented in the Table 1. The detail of the C-program can be explained in the flowchart in Figure 10.

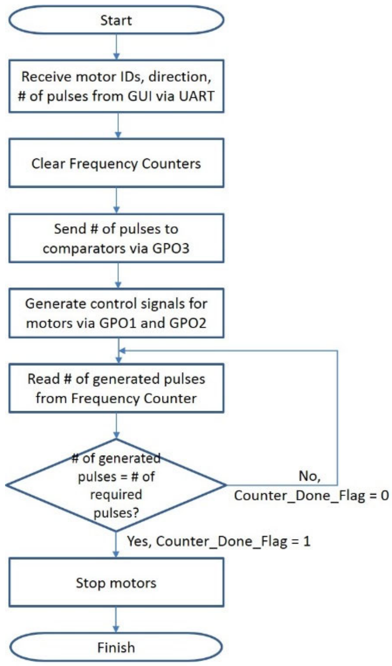

Figure 10: C-Program flowchart implemented on MicroBlaze MCS.

Figure 11: Position of phase shifters and amplitude attenuators after system implementation.

SYSTEM PERFORMANCE

The digital circuit blocks designed by VHDL were carefully tested to make sure they function correctly. The tests were performed both by using standard peripherals on the FPGA board and by connecting with multiple stepper motors and their electronic drivers and controllers in laboratory. By implementing several digital circuit blocks in the FPGA, the complication in designing the main software written in C on the MicroBlaze MCS can be considerably reduced.

The accuracy of the position control is the main important part of this stepper motor control system. Hardware and software tests and parameter settings play a very crucial role in the system implementation. In order to obtain the correct number of pulses required for any setpoint value, several trials were performed for all of the motors that were already installed in the linac's waveguide system. Figure 11 shows the result of final motor positions from the electronic front panel display after testing the new control system. We can see that, when compared to the reference inputs from LabVIEW GUI shown in Figure 8, the result is very satisfactory. The accuracy of the position control is within 0.4 % error.

CONCLUSION

The motion control system that governs the position of high power phase shifters and amplitude attenuators in the linac's RF system is implemented. The old drive system using reversible AC induction motors was replaced by a new set of commercial stepper motors with electronic drivers and controllers. Several digital control circuit blocks are designed in VHDL and implemented on a commercial FPGA board. The Xilinx's MicroBlaze MCS is the main controller to perform control decision based on the LabVIEW GUI commands and the digital signals from other digital circuit blocks in the FPGA fabric.

A number of tests were conducted to successfully achieve the desired result. The digital circuit blocks in the FPGA help achieve the appropriate control action and reduce the size and complication of the main C program on the MicroBlaze MCS. The design is good for our system because there are a number of signals between the FPGA board and electronic drivers and controllers of seven stepper motors in the system. The LabVIEW GUI is used to communicate with the FPGA via UART to provide reference points and display position values. The overall performance of the control system, both hardware design and software implementation, is very satisfactory. This stepper motor control system is also planned to be used with phase and amplitude detectors described in [1] in order to implement a linac's RF phase and amplitude control system in the future.

REFERENCES

[1] R. Rujanakraikarn, "FPGA-Based Pulsed-RF Phase and Amplitude Detector at SLRI", in *Proc. 16th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS2017)*, Barcelona, Spain. doi:10.18429/JACoW-ICALEPCS2017-TUPHA068

[2] 5-Phase Stepping Motor Unit RK Series, Operating Manual, http://www.orientalmotor.com

[3] New 5-Phase Stepping Motor and Driver Package RKII Series, User Manual, http://www.orientalmotor.com

[4] ML605 Hardware User Guide (UG534 v1.7), Jun. 2012, http://www.xilinx.com

[5] FMC XM105 Debug Card User Guide (UG537 v1.3), Jun. 2011, http://www.xilinx.com

[6] K. L. Short, *VHDL for Engineers*, Pearson Ed. 2009.

[7] LogiCORE IP MicroBlaze Micro Controller System Product Specification (DS865 v1.0), Jan. 2012, http://www.xilinx.com