

# IFMIF LLRF CONTROL SYSTEM ARCHITECTURE BASED ON EPICS\*

Julio Calvo<sup>†</sup>, Angel Ibarra

Centro de Investigaciones Energéticas Mediomambientales y Tecnológicas, Ciemat, Spain  
Miguel Angel Patricio, Departamento de Ciencias de la Computación e Inteligencia Artificial  
Universidad Carlos III Madrid, Spain

Mark Rivers, Department of Geophysical Sciences and Center for Advanced Radiation Sources  
The University of Chicago, USA

## Abstract

The IFMIF-EVEDA (International Fusion Materials Irradiation Facility - Engineering Validation and Engineering Design Activity) linear accelerator will be a 9 MeV, 125mA CW (Continuous Wave) deuteron accelerator prototype to validate the technical options of the accelerator design for IFMIF. The primary mission of such facility is to test and verify materials performance when subjected to extensive neutron irradiation of the type encountered in a fusion reactor to prepare for the design, construction, licensing and safe operation of a fusion DEMO (Fusion Demonstration Reactor). The RF (Radio Frequency) power system of IFMIF-EVEDA consists of 18 RF chains working at 175MHz with three amplification stages each. The LLRF (Low-Level Radio Frequency) controls the amplitude and phase of the signal to be synchronized with the beam and it also controls the resonance frequency of the cavities. The system is based on a commercial cPCI (Compact Peripheral Component Interconnect) FPGA (Field Programmable Gate Array) board provided by Lyrtech and controlled by a Windows Host PC. For this purpose, it is mandatory to communicate the cPCI FPGA Board with an EPICS Channel Access [1], building an IOC (Input Output Controller). A new software architecture to design a device support, using AsynPortDriver class and CSS as a GUI (Graphical User Interface), is presented.

## INTRODUCTION

The RF System is defined as the equipment necessary to convert the high-voltage AC (alternating current) primary power to suitably conditioned RF power for input to the IFMIF-EVEDA accelerator cavities [2]. The quality of the RF delivered to the accelerator cavities is controlled to within  $\pm 1$  degree in phase and to within  $\pm 1\%$  in amplitude, using a low-level RF-drive modulated control system. Each RF Module Local Control System (RF Module-LCS) is a device that will monitor and control all physical parameters within the RF Chains located in the same RF Module. Each RF module comprises 2 RF Chains, so the 18 RF Chains will be monitor and controlled by 9 RF Module-LCS connected via Ethernet to the Central Control System

\* This work is funded by Ministerio de Ciencia e Innovación del Gobierno de España, under the Project No. AIC10-A-000441.

<sup>†</sup> julio.calvo@ciemat.es

(CCS) [3], this Local Control System scheme is shown in Fig. 1. Furthermore, The primary role of the Low Level Radio Frequency system (LLRF) is to control the amplitude and the phase of each cavity (fast regulation) and to control the tuning of each cavity to keep its resonant frequency close to the accelerator operating frequency. For doing so, the LLRF will generate the RF signals (RF drives) for the amplifiers feeding the cavities, depending on their voltage and forward power. The IFMIF-EVEDA LLRF system has to work under CW mode operation and it has also to support pulse mode operation (during the commissioning and tuning of the prototype accelerator) [3].

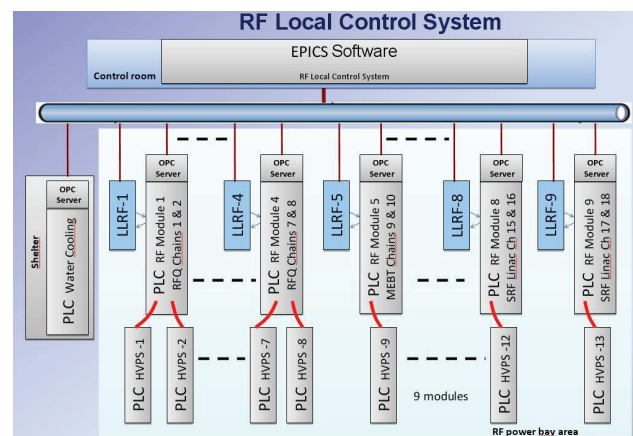


Figure 1: LLRF local control system scheme.

For controlling and operating this LLRF System, IFMIF-EVEDA decided to chose EPICS some years ago. EPICS is a set of Open Source software tools, libraries and applications developed collaboratively and used worldwide to create distributed soft real-time control systems for scientific instruments such as a particle accelerators, telescopes and other large scientific experiments. Nowadays, EPICS is widely used in big accelerators like Diamond Light Source and huge experiments like ITER. EPICS provides a number of tools for creating and operating a control system. This minimizes the need for custom coding and helps ensure uniform operator interfaces [4]. These features make EPICS the most appropriate architecture for IFMIF-EVEDA, hence, the main purpose of this paper is to present the LLRF control system based on EPICS.

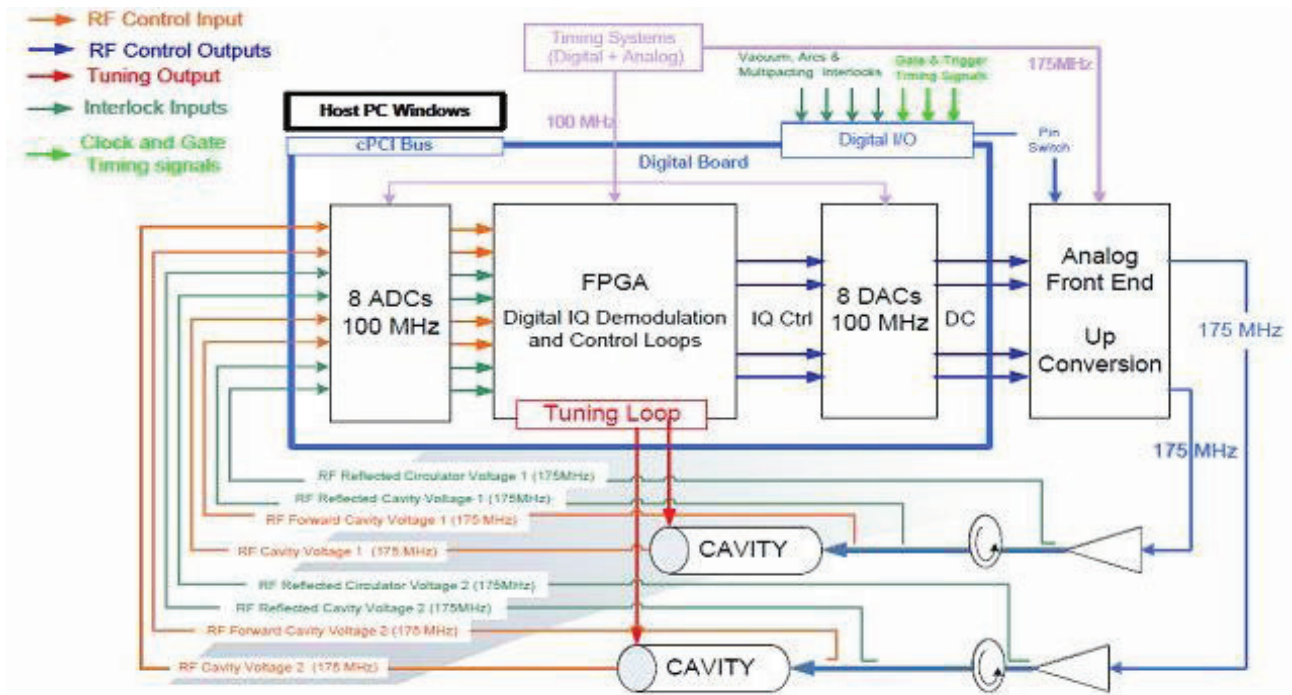


Figure 2: LLRF System general overview.

## SYSTEM ARCHITECTURE

### Hardware Architecture

The LLRF System is composed by three main subsystems: a digital board with fast FPGA, the analog front end and a local timing system.

- **Digital Board:** The digital board contains one Virtex-4 FPGA, 8 ADCs and 8 DACs with 14 bits resolution and capable to work up to 105MHz. It is a commercial board with cPCI format provided by Lyrtech and it will be controlled by a Windows CPU. This board will acquire different kind of signals: RF control inputs (cavity voltage and forward cavity power), RF interlock inputs (cavity reflected power and circulator reflected power), digital interlocks (arcs, vacuum and multi-pacting) and timing signals (gate and pulse signals). It will also provide the control outputs of the LLRF: Direct Current (DC) control outputs for signals. It will also provide the control outputs of the LLRF: DC controls amplitude, phase, tuning and interlocks outputs of the cavities. The LLRF general system overview is shown in Fig. 2.
- **The front end:** It is in charge if up-converting the DC control outputs from the digital board into RF. For doing so, the LLRF employs a quadrature IQ modulator.
- **The local timing system:** It consists of a PLL board with a 100MHz VCXO (CDC-7005-EVM from Texas Instruments). This board provide 100MHz TTL signal to clock the digital board. This signal will be phase

locked with an external 10MHz signal provided by the general timing system.

### Software Architecture

The presented architecture is developed over EPICS Base 3.14.11, using Visual Studio 2008 C++ Express Edition, Asyn4-13-1, Application Programming Interface (API) for Lyrtech boards and Control System Studio (CSS).

This solution consists of the following EPICS components:

- **IOC (Input/Output Controller):** The IOC is any platform that can support EPICS run-time components, including the database, and its access routines, device drivers, record types for various input and output and scanning and monitoring functionality [5].
- **Database:** The EPICS database [6] is a basic element in an IOC. The database is composed by numerous instances of records. A Record is a collection of Processes Variables (PVs) which is implemented on an IOC where a Process Variable (PV) is a piece of data which can be accessed by its unique name. Often the PV name consists of a record name and the name of a record field, connected by a dot [7]. In the LLRF control system, the records have an associated hardware, Lyrtech boards.
- **Device Support:** We could define it as an interface between records and hardware. A device support routine has knowledge of the record definition. It also knows

how to talk to the hardware directly or how to call a driver which interfaces to the hardware. Thus, device support routines are the interface between hardware specific fields in a database record and device drivers or the hardware itself [8].

- AsynDriver: It is a general purpose facility for interfacing device specific code to low level communication drivers. A primary target for asynDriver is EPICS IOC device support but it is independent of EPICS [9].
- LAN: Local Area Network. This is the computer based network which allows the communication between IOCs and Operator Interfaces (OPIs). EPICS provides a software component, Channel Access, which provides network transparent communication between a Channel Access client and an arbitrary number of Channel Access servers [5].
- OPI: Control System Studio (CSS) is a combined effort of several parties, including DESY (Hamburg, Germany) and SNS (Oak Ridge, TN). It provides a collection of control system tools in a common environment, based on Eclipse [10].

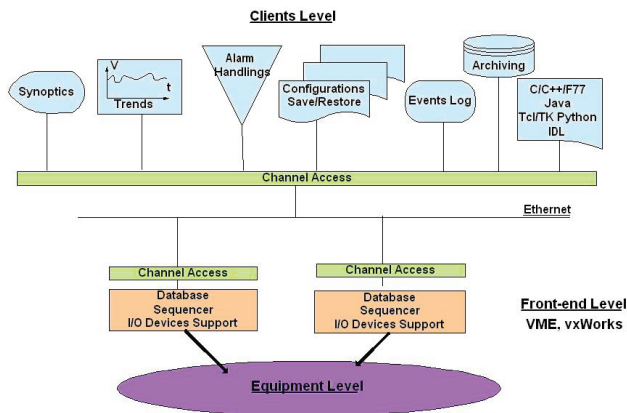


Figure 3: EPICS five layers structure.

This architecture is module based and it follows the EPICS five layer model, this architecture is shown in Fig. 3. In client level we have used CSS as software that allows PVs to be accessed and modified from the network, using the second layer, the Channel Access, that is the communication protocol used by EPICS to transfer information through the network. The next layer is the record support one, where the IOC is settled; here, we have the software which implements PVs for the use with Channel Access. The fourth Layer, device driver layer, it is the heart of the architecture and permits the communication between records in the DataBase and the device. The fifth level corresponds with the instrumentation level, FPGA boards. The proposed architecture is shown in Fig. 4. Due to the Common Software Platform of IFMIF-EVEDA, where all the computers must include the same version of Linux, Vx-works core and EPICS, but cPCI FPGA Board is controlled

by a Windows Host PC, carrying out independence of the hardware from the operating system was one of the main challenges of the presented architecture.

## USER INTERFACE

The graph user interface, Figure 4, has been designed using CSS. Initially, the proposed GUI package for IFMIF-EVEDA was Extensible Display Manager (EDM). Despite the initial effort that is necessary to implement a clean and robust design, we consider that we made the right decision. We have tried to design a friendly and operative GUI. The operator and the machine engineer can easily tune the system, control the parameters and program Lyrtech boards through the set of designed pages but, following the EPICS philosophy, the whole logic involved have not been developed in the client level, CSS, but on the IOC, then everyone can reach it.

## SUMMARY AND FUTURE PLAN

We have developed a Device Support for IFMIF-LLRF Control System. This new software architecture is running properly in a Windows CPU connected in the same chassis of the Lyrtech cPCI board and solves the operating system dependency. This new architecture allows both client and server run in different machines. This solution will be used to control the LLRF of two plants in the final accelerator prototype which will be built in Rokkasho, Japan, on the 2013. Thanks to CSS and EPICS Channel Access, process variables can be seen in any computer of the IFMIF-EVEDA Central Control System. The EPICS based characteristics of the system makes it useful because of its modularity and it can be easily upgradeable and modifiable, the choice of EPICS as a control toolset was very important to achieve this success.

The further work line moves on the testing of the system within the overall prototype accelerator. Else, adding more elements to the GUI and creating an auxiliary database in order to store the historical values of the diagnostics signals is necessary. When these signals are out of certain margins, some warning messages should appear next to the relevant diagnostic signal, these warning messages will be defined later on.

## REFERENCES

- [1] J.C. Yoon, J.W. Lee, K.M. Ha, J.H. Kim, J.M. Kim and J. Choie. "EPICS Based Control System for the KOMAC RF System". Proceedings of EPAC 2004, Lucerne, Switzerland. Pohang Accelerator Laboratory, POSTECH, Pohang 790-784, Korea, 2004.
- [2] International Energy Agency. "IFMIF Comprehensive Design Report, January 2004".
- [3] I. Kirpichev, P. Méndez, M. Weber, A. Ibarra, M.A. Falagan, M. Desmons, A. Mosnier. "RF Power System for the IFMIF-EVEDA Prototype Accelerator". Proceedings of

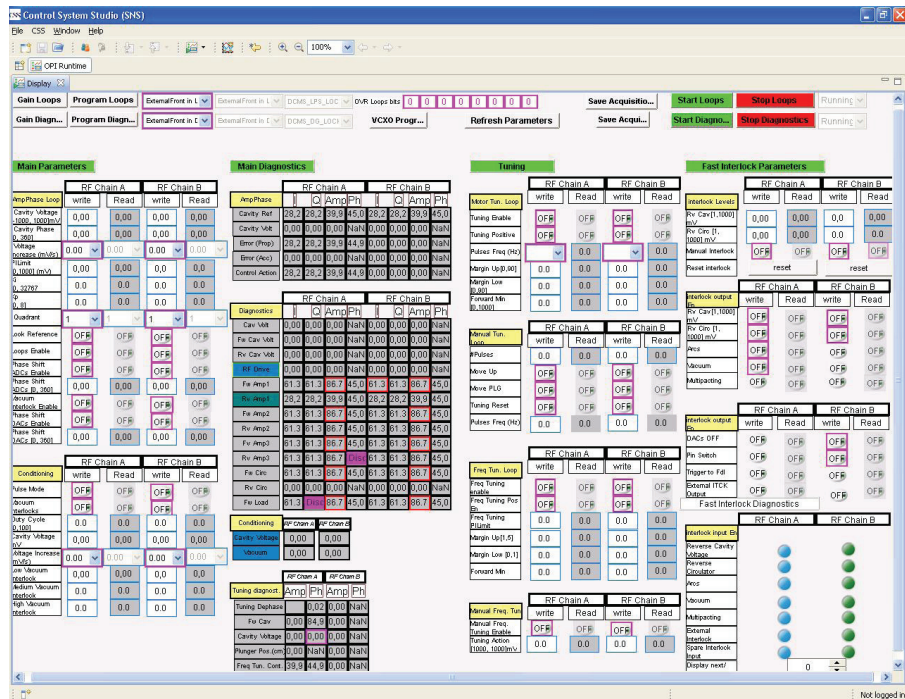


Figure 4: OPI CSS.

EPAC 2004, Lucerne, Switzerland. CIEMAT, Avda. Complutense 22, 28040 Madrid, Spain and CEA, IRFU, F-91191 Gif-sur-Yvette, France, 2008.

- [4] EPICS: <http://www.aps.anl.gov/epics/>
- [5] EPICS IOC: <http://www.aps.anl.gov/epics/base/R3-14/12-docs/AppDevGuide/node4.html>
- [6] Stanford Database: <http://www.slac.stanford.edu/comp/unix/package/epics/training/documents>
- [7] EPICS Base: [http://epics.web.psi.ch/style/training/handouts/e\\_basehandouts/node2.htm/](http://epics.web.psi.ch/style/training/handouts/e_basehandouts/node2.htm/)
- [8] Device Support: <http://www.aps.anl.gov/epics/docs/USPAS2010/Lectures>
- [9] Asyn: <http://www.aps.anl.gov/epics/modules/soft/asyn>
- [10] DESY: <http://css.desy.de/>