

AN ADVANCED KLYSTRON INTERLOCK SOLUTION BASED ON NIOS-II PROCESSOR

Vahan Petrosyan[#] (PITZ), Stefan Weisse (PITZ), Holger Leich (EL/Z), Torsten Grevsmuehl (MHF-p) Philip Duval (MST).
DESY, Germany.

Abstract

PITZ (Photo Injector Test Facility Zeuthen) is a test facility at DESY Zeuthen for research and development on laser driven electron sources for Free Electron Lasers (FEL) and linear colliders. The existing Klystron interlock solution is based on a NIOS 32-bit RISC processor implemented in an Altera FPGA. Currently all necessary control functions to the slave modules in the interlock crate are performed by the NIOS processor. The interface to the PITZ control system is implemented via Ethernet through a meta-server which translates the binary data sent by the board to be understandable by a DOOCS control system. The new solution is based on a NIOS-II processor FPGA design. The main goal is to avoid the Meta-Server and thus to provide the full functionality directly on the network. Altera provides a software development kit for NIOS-II which includes a MicroC/OS-II real-time operating system (RTOS) and a lightweight IP (lwIP) TCP/IP Stack software component. The combination of these components gives the possibility to run a TINE (Three-fold Integrated Network Environment) server on the NIOS-II processor. As TINE control system software is able to talk to DOOCS clients, it is possible to provide a Klystron interlock solution directly on the controller board, delivering a real-time interface directly to the PITZ control system.

INTRODUCTION

A very compact, scalable interlock system for the XFEL-RF-stations has been designed. The main hardware parts of the system are:

- The interlock controller board (Fig.1) based on ALTERA-FPGA design which incorporates a 32-bit RISC NIOS-II CPU.
- The 19" 4U interlock crate (Fig.2) with dedicated backplane optimized to the application.
- several I/O modules that implement the interface to the process.



Figure 1: Interlock controller board.

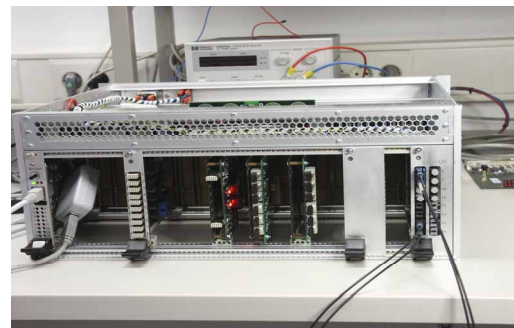


Figure 2: Interlock crate.

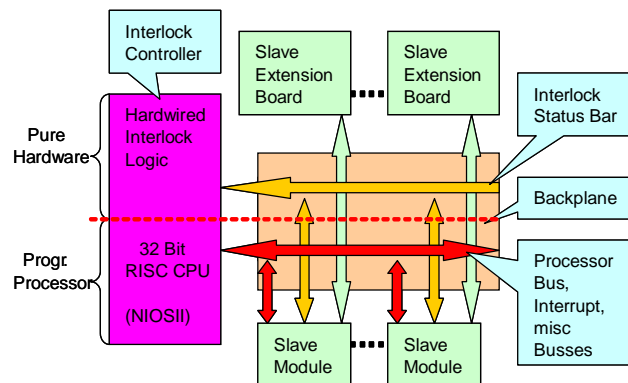


Figure 3: Interlock System Layout.

[#] on leave from YerPhi 375036 Yerevan, Armenia

HARDWARE COMPONENTS

The design of the new interlock controller board incorporates the following devices and components:

- a 32-bit RISC processor (NIOS-II within an ALTERA FPGA design)
- 10/100 Mbps LAN 91C111 Ethernet device
- 8MByte AMD AM29LV065D Flash memory device
- 16 MByte RAM memory device
- 4 Mbit Serial configuration device (EPCS4)
- Interface to the backplane to enable access to all slave modules in the crate.

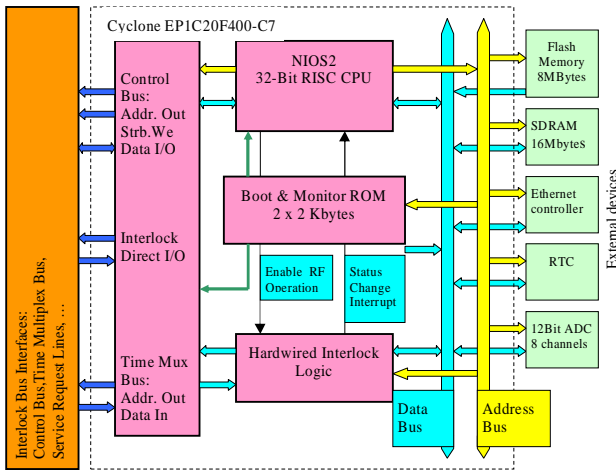


Figure 4: Interlock controller board layout.

The available connectors on the board are:

- JTAG UART
- RJ 45 for Ethernet device
- RS-232 interface

SOFTWARE COMPONENTS

All necessary software and hardware development tools are included in the Altera's NIOS-II development kit, which is a complete embedded systems development kit for NIOS-II embedded processor. It contains Quartus-II, IDE, SOPC builder, etc.

The Quartus-II Design Software is a comprehensive environment for developing hardware design files and output a configuration file to the target FPGA as well as for assigning I/O locations, performing timing analysis on the FPGA design. Designers use the SOPC Builder system integration tool to define and to integrate NIOS-II processor-based hardware systems.

The NIOS-II IDE (Integrated Development Environment) provides a full-featured, standard and stable environment for software designers. The NIOS-II IDE flash programmer is used to program any common

flash interface(CFI)-compliant flash device connected to the FPGA.

The NIOS-II IDE includes a MicroC/OS-II RTOS (portable, ROMable, scalable, preemptive, multitasking kernel) operating system and a lightweight IP (lwIP) TCP/IP stack software components. The NIOS-II IDE makes it easy to create an application project based on MicroC/OS-II and include the lwIP TCP/IP stack in a system.

Tight integration between SOPC Builder and NIOS-II IDE allows the HAL(hardware abstraction layer)system library to be generated automatically. The HAL system library is a lightweight runtime environment that provides a simple device driver interface for programs to communicate with the underlying hardware. The HAL API is integrated with the ANSI C standard library and allows you to access devices and files using familiar C library functions.

The GNU toolkit contains the basic tools for source code compilation such as an assembler, C/C++ compiler and linker.

The TINE control system parts were ported to NIOS-II based on that environment. The TINE is a multi-platform control system library that is used mainly at DESY accelerators as a communication and data exchanging protocol between servers running on front-end-computers (FEC) and clients.

IMPLEMENTATION

The interlock controller board is based on a Cyclone EP1C20F400-C7 FPGA design.

Inside the FPGA the following items are implemented:

- interlock function for components of the RF station based on signals that were pre-processed within the different I/O-modules
- interface to the main control system via Ethernet
- control of the I/O-modules in the crate
- master of all backplane busses
- source of the bus timing signals
- some slow control functions
- system check after power up or per command.

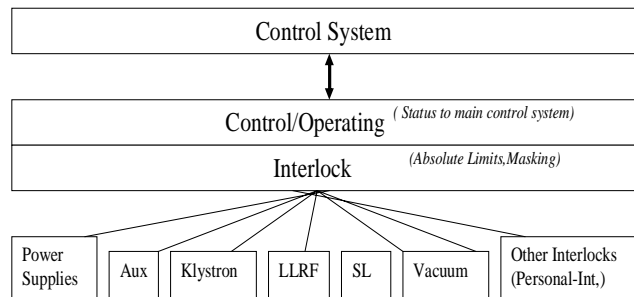


Figure 5: Interlock System Integration.

It has been decided that a TINE server should run on the controller board to provide an interface to the control system. For that purpose an existing TINE kernel was migrated to the NIOS-II CPU.

The lightweight IP (lwIP) is used for communication over an Ethernet network. Altera provides a lwIP driver to support the Ethernet LAN that is based on a SMC lan91c111 MAC/PHY device.

The TINE protocol uses pure BSD sockets for network communication. Currently, there is no proper BSD wrapper library for the proprietary interface of lwIP available on the market. The authors have created a wrapper BSD socket library for lwIP, so that easy porting of UNIX source code to Altera NIOS-II is possible. This BSD-style socket library was developed based on the raw TCP/IP interface of the lwIP stack. The program execution is driven by callbacks. The lwIP uses the MicroC/OS-II RTOS multi-threaded environment. The MicroC/OS-II was chosen as the main operating system because of pre-emptive multitasking with guaranteed response times. This is a requirement for getting the data out of the hardware registers before they are overwritten by the next event.

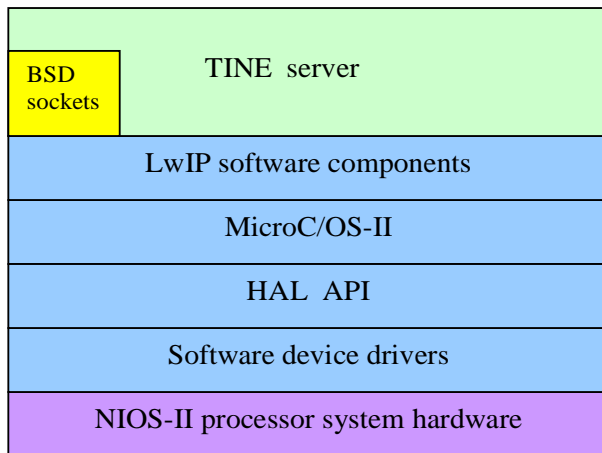


Figure 6: Software layout.

The following information will be available via Ethernet as control system properties :

- all status information and all channel mask data
- all actual values of analog input channels
- all commands to the interlock crate (interlock board).

CONCLUSION

A prototype of the TINE server (sine example) is running on the controller board based on the NIOS-II CPU. This server has been tested with the few client applications using both UDP and TCP connections for a long time. The results are very encouraging and the server is ready to be adapted to the interlock system registering all corresponding properties of the interlock system.

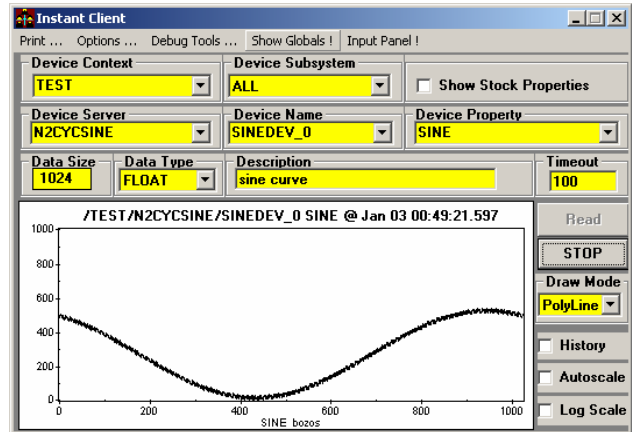


Figure 7: TINE sine example server readout by Instant Client.

In addition to the TINE implementation a Web server has been developed and is planned to be installed on the controller board. Using the web-server, a web interface, which provides the possibility to reconfigure the whole FPGA design or to upload a new software version via Ethernet interface, will be implemented on the board.

REFERENCES

- [1] TESLA Technical Design Report, DESY TESLA-2001-011,2001
- [2] S. Choroba, T. Grevsmuehl, H. Leich, S. Simrock, "The TTF2 / XFEL Klystron Interlock: Requirements and Implementation" (Requirements_Draft.doc)
- [3] H. Leich, Report at Spring Conference of the SEI, (SEI_290304.ppt)
- [4] TINE (Three-fold Integrated Networking Environment), <http://desyntwww.desy.de/tine>
- [5] Micrium Technologies Corporation <http://www.ucos-ii.com/>
- [6] Adam Dunkels , Swedish Institute of Computer Science, <http://www.sics.se/~adam/lwip/>, <http://savannah.nongnu.org/projects/lwip/>
- [7] Altera Corporation , <http://www.altera.com>