# Application for Supervision and Management of ATCA-based Systems

Jan Wychowaniak, Dariusz Makowski, *Member, IEEE*,
Pawel Predki, and Andrzej Napieralski, *Senior Member, IEEE*

***Abstract*—The Advanced Telecommunications Computing Architecture (ATCA) standard describes a powerful, high performance platform. Its implementation has been considered among candidates for a base of the control system of the X-ray Free Electron Laser (X-FEL), which is being built at Deutsches Electronen-Synchrotron (DESY) in Hamburg, Germany. The Low Level Radio Frequency (LLRF) control system is designed as a set of ATCA Carrier Boards. Each Carrier Board hosts an Intelligent Platform Management Controller (IPMC), which is developed in compliance with the PICMG specifications. IPMC is responsible for management and monitoring of components installed on Carrier Boards and pluggable Advanced Mezzanine Card (AMC) modules.**

**The ATCA Shelf Manager is the main control unit of a single ATCA shelf, responsible for power management, fan modules and Carrier Boards installed in ATCA shelf. It provides a set of control and diagnostic capabilities regarding the shelf and its sub-modules. These capabilities are available for operators and can be used by higher level applications.**

**This publication presents a software component intended to support management and supervision of the ATCA shelf and its sub-modules, including Carrier Boards with AMC modules. The application provides enhanced mechanisms of control and allows to acquire detailed information regarding status and parameters of crucial devices (e.g. power supply voltages, temperatures, presence of reference clocks). The information supplied from Shelf Manager combined with graphical user interface of the application provides visual representation of selected system components and contributes towards efficient control and supervision of Carrier Boards and entire ATCA-based platform.**

***Index Terms*—Advanced Telecommunications Computing Architecture, Carrier Board, Intelligent Platform Management Interface, Java, Shelf Manager, X-ray Free Electron Laser**

## I. Introduction

The Advanced Telecommunications Computing Architecture (ATCA) standard is under consideration as a basis for the Low Level Radio Frequency (LLRF) control system of the X-Ray Free Electron Laser (X-FEL). Its implementation is considered to potentially supplant the architecture based on the Versa Module Eurocard (VME), as the new solution offers extraordinary reliability and system availability levels, which is appealing to the industry [1].

The main monitoring and control unit of a single ATCA shelf is the ATCA Shelf Manager (ShM) [2]. An ShM offers a group of interfaces for shelf-external access, which enable for remote shelf management and diagnostics [3]. Such possibilities are beneficial during both the implementation development and the working system supervision.

Among the interfaces available at the Pigeon Point Systems ShMs there are the Web Interface and the Command Line Interface (CLI) [4]. The former is accessible via a web browser and interaction with it is based on HTTP forms. Data from a shelf is presented to a shelf operator in a form of a web page generated by an ShM, which needs to be read and interpreted each time an operator intends to query a shelf for any new piece of information.

Communication via the CLI consists in using a text terminal. A shelf-external workstation acts as a remote terminal to an ShM, via which an operator can issue requests in a form of text-based commands with optional parameters, and an ShM responds in a similar manner. The interface defines a set of commands available for an operator to use. Here also communication is a matter of assembling a request and then reading and interpreting the response text each time an action or piece of information is required.

When considering supervision and controlling activities of a shelf sub-components, this need of paying the repetitive effort to compose a request and interpret the response causes them to be inefficient. In particular, in this form those activities are not applicable for real-time monitoring. Thus there is a need for a tool that would facilitate the automation of those messaging processes and present the operation results in a graphical form.

A software tool being a subject of this paper offers a means of monitoring ATCA shelf and its sub-modules and altering their operation conditions. By using it an operator is able to comfortably perform efficient diagnostics in real-time, as well as control the shelf inventory. It also provides access to custom functionality implemented in hardware. The benefits offered by this tool are appreciated at the stage of implementing the ATCA standard, when testing activities involving e.g. newly introduced components gain significant support. They are also noticeable later, when failure detection and supervision of a complete and working system can be

The research leading to these results has received funding from the European Commission under the EuCARD FP7 Research Infrastructures grant agreement no. 227579 and Polish National Science Council Grant 642/N-TESLAXFEL/09/2010/0. The authors are scholarship holders of project entitled "Innovative education ..." supported by European Social Fund.

J. Wychowaniak, D. Makowski, P. Predki, A. Napieralski are with the Technical University of Lodz, Department of Microelectronics and Computer Science 93-590 Lodz, Poland (e-mail: jwychowa@wpk.p.lodz.pl).

performed remotely.

## II. Requirements for the Diagnostic and Control Application

The direct use of the diagnostic and control interfaces that the Shelf Managers from Pigeon Point are equipped with, offers no support for real-time monitoring of shelf operation. Nor is it a proper means for delivering efficiency at management and supervision, as the need of participation of an operator in the processes of command composition and data interpretation requires a certain dose of effort.

A tool, the task of which is to address these issues, is expected to automate the creation of requests regarding specified parameters of the shelf operation. In case of a need for a constant focus upon selected shelf properties the tool should be able to transmit appropriate requests periodically in order to offer real-time tracking capabilities. What is more, the tool is expected to provide means for automatic interpretation of data obtained from ShM in order for its relevant pieces to be presented to an operator in a clear and cohesive manner, preferably with the use of a graphical user interface (GUI), that would also offer interfaces for intuitive management. The tool is expected to run on a shelf-external personal computer, from where it would connect to an ShM over Ethernet. It should be capable of communicating both via a direct link, as well as remotely (e.g. basing on SSH tunnelling). The internal structure of the application is expected to support expandability.

## III. Functionality Provided by the Application for the ATCA-based LLRF Systems

The application is being developed and tested with using a 6-slot ATCA shelf with the ShMM-500 Management Module-based ShM from Pigeon Point Systems.

### *A. Monitoring Capabilities*

As far as supervision is concerned, the application is able to:

- Scan a shelf for Field Replaceable Units (FRUs) visible to an ShM. For each FRU it determines its hot-swap state, reads its Intelligent Platform Management Interface (IPMI) address and obtains several basic pieces of information from its Sensor Data Records (SDR), like the device identifier data or the device type (the FRU device types are specified as described in the IPMI specification). The application presents to an operator all the obtained information for every detected FRU in a tabular form (Fig. 1).
- Scan a shelf for Intelligent Platform Management Controllers (IPMCs) visible to an ShM, and therefore to examine every intelligent FRU an ShM is able to detect. For every such FRU the application determines a hot-swap state and an Intelligent Platform Management Bus (IPMB) address the FRU is visible under, as well as examines its SDR for a list of the IPMI functions and 'logical device' commands that the intelligent FRU supports. The information on all the detected IPMCs is presented in a tabular form.
- Create a graphical software representation of the shelf front panel, as an element of the application GUI (Fig. 2, Fig.3). It is aimed at reflecting the presence of the shelf hot-swappable modules, as well as their status, including their hot-swap state or indication of potential difficulties encountered during the activation. This visualisation operates on a real-time basis, by periodical examination of the shelf sub-components.

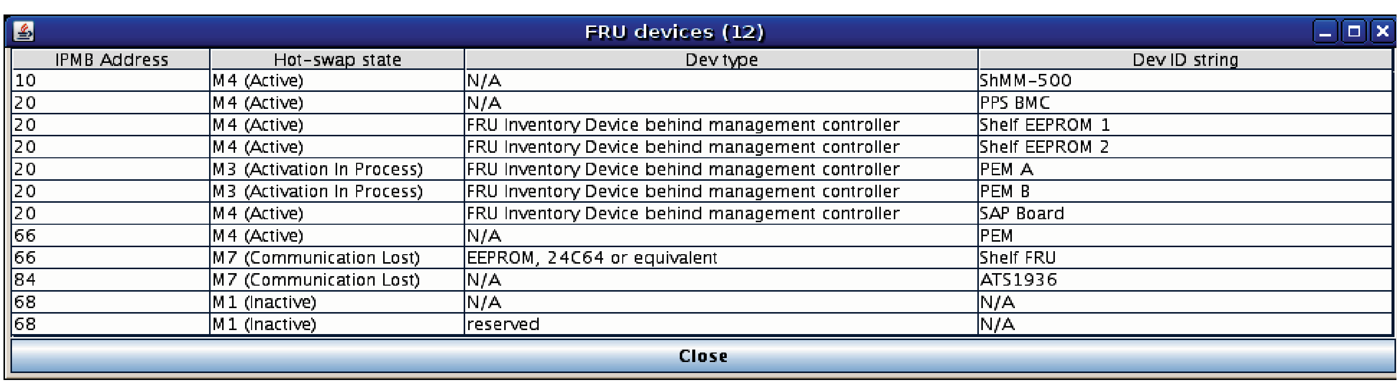
Those monitoring capabilities are useful at the development stage, where immediate and effective insight into the operation of selected (e.g. newly introduced) sub-components is required, as well as for remote monitoring of a working system.

### *B. Management Capabilities*

The summary below highlights the features of the application regarding the management capabilities it offers.

The application:

- Is able to control a specified FRU. If a shelf includes a sub-component, that is not subject to automatic activation by an ShM after insertion, the application allows to manually trigger this process. Its is also capable of ordering deactivation of a specified device, as well as for reactivation a FRU deactivated this way.
- Is capable of ordering a reboot of a specified FRU. The option allows to perform both resetting a FRU with interrupting its activities at a specified moment, and causing its graceful restart. The application offers this functionality for a given FRU provided that this FRU supports it. The application therefore facilitates determining of available options for a FRU.




| IPMB Address | Hot-swap state | Dev type | Dev ID string |
|---|---|---|---|
| 10 | M4 (Active) | N/A | ShMM-500 |
| 20 | M4 (Active) | N/A | PPS BMC |
| 20 | M4 (Active) | FRU Inventory Device behind management controller | Shelf EEPROM 1 |
| 20 | M4 (Active) | FRU Inventory Device behind management controller | Shelf EEPROM 2 |
| 20 | M3 (Activation In Process) | FRU Inventory Device behind management controller | PEM A |
| 20 | M3 (Activation In Process) | FRU Inventory Device behind management controller | PEM B |
| 20 | M4 (Active) | FRU Inventory Device behind management controller | SAP Board |
| 66 | M4 (Active) | N/A | PEM |
| 66 | M7 (Communication Lost) | EEPROM, 24C64 or equivalent | Shelf FRU |
| 84 | M7 (Communication Lost) | N/A | ATS1936 |
| 68 | M1 (Inactive) | N/A | N/A |
| 68 | M1 (Inactive) | reserved | N/A |

Fig. 1. A GUI element of the application enabling to inspect presence and state of FRUs visible to a ShM.


Fig. 2. A graphical software representation of the shelf front panel. The monitored shelf is empty. Buttons from the button section to the right, which refer to devices that are absent at a given moment, are inactive. This makes them slightly less distinguishable from their background.


Fig. 3. A graphical software representation of the shelf front panel. The monitored shelf contains two cooling units and a Carrier Board with two AMCs.

- Controls the operation of the shelf cooling units by adjusting the cooling level the units are supposed to deliver (Fig. 4). The control includes also setting a minimal level, below which the cooling units operation intensity is not permitted to drop, even at no threat from the thermal conditions of a shelf.
- Enables to monitor the System Event Log (SEL) of a specified intelligent FRU and to view its contents for specified sets of events. It also allows to erase its contents.
- Presents the information regarding IPMB-0 connectivity from the view of determined group of IMPCs. The operation of IPMB-A and IPMB-B links for each controller can be examined separately. The activity of those links can also be modified with the application.
- Is capable of sending custom IPMI-compliant messages to an arbitrarily specified IPMC in a shelf. A dedicated interface is offered by the application for generating and issuing messages, as well as for viewing data or request execution results from a selected IPMC. This feature gives an operator a possibility to perform custom control over intelligent FRUs, which becomes especially important when a FRU happens to implement additional, OEM-provided properties, that may step beyond the ATCA specification [5]. That described application feature is particularly useful for cooperation with Carrier Boards for LLRF [6]. Those boards may be often equipped with elements providing additional, non ATCA-specific functionality, like e.g. monitoring board reference clocks or switching channels for data transmission that a board may use. A group of non-standard IPMI commands defined specifically for such boards allows to perform control over these processes. The discussed IPMI messaging interface of the application includes support for those commands, which allows an operator to manage the custom board properties via the application.
- Provides a similar IPMI messaging functionality for AMC modules.

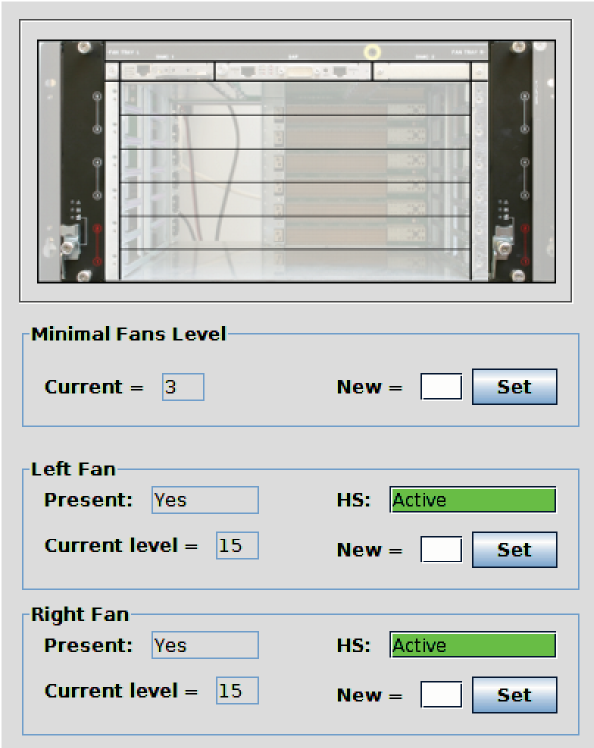



Fig. 4. A GUI element of the application responsible for supervising and controlling the operation of the shelf cooling units.

## IV. Communication with Shelf Manager

The operation of the application is based upon a connection between an ShM and a shelf-external PC station, via the Ethernet-based System Manager link that an ShM is supposed to be provided with. The ShM used for the development of the application is based on the ShMM-500 Management Module equipped with Dual 10/100 Mbit Ethernet connection, which provides access to a group of interfaces for external use that the ShM implements. These include the CLI, the Web interface and other, like Remote Management Control Protocol (RMCP, required by ATCA) [3, 4].

The System Manager can be described as a logical concept, which may include human as well as programmatic operators. The interface for the System Manager, which may be implemented by an ShM, provides an array of mechanisms of access to specified means of control and diagnostics regarding a shelf [2].

The application uses the CLI as the access interface. In general, the CLI provides a means of communication with the IMPCs across a shelf, as well as with an ShM itself. A system of text-based requests and responses is used for this purpose. The CLI, accessible via Telnet, defines a set of commands for generating and exchanging messages in the IPMI-compliant format. A human operator or a higher-level management software can take advantage of this feature. It enables to influence the shelf operation, as well as to obtain information on the shelf state like sub-modules visible to an ShM, sensors (their values and threshold settings), events, and general shelf functioning conditions.

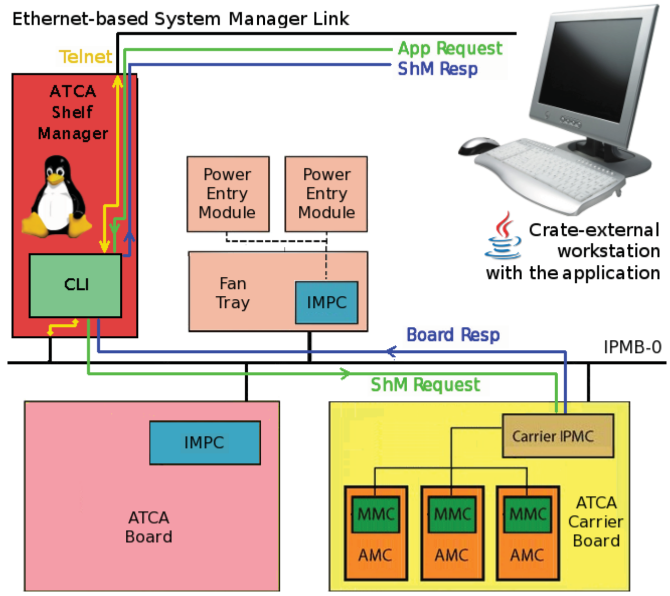



Fig. 5. Diagram of the application cooperation with an ATCA ShM.

The application at start-up uses the System Manager link to establish a Telnet connection to Linux operating on an ShM [8] (Fig. 5). When the connection is operational, the application engages in exchanging the IPMI-based messages with shelf IMPCs via the CLI mechanisms [4].

The application uses a specified sub-set of the commands provided by the CLI. Below is a short summary of the monitoring-oriented part of them.

- *ipmc* - The command investigates and returns data regarding an IPMC under a specified IPMB address or all IPMCs visible to an ShM, if no address is given. The result is a set of records, related to a single controller each, containing information including its IPMB address, hot-swap state and a group of other details obtained from its SDR.
- *fru* - The command returns data regarding a FRU specified by the IPMB address, or all the shelf FRUs in case of the address being unspecified. It outputs in a similar manner to the ipmc command, however it is not limited to intelligent FRUs. This command results in obtaining data set which includes IPMB address and hot-swap state of every FRU, as well as the type a particular FRU represents in a short description.
- *board* - Enables to specify a board by the number of a physical shelf slot the board occupies. This way it is possible to determine, whether a slot of a given number is occupied by a Carrier Board or vacant.
- *fans* - This command generates queries regarding presence of cooling devices within a shelf, to which an ShM responds by enumerating all the cooling devices it is able to detect, together with the IPMB addresses they reside at.

The following is a summary of the management-oriented sub-group of the utilised command set.

- *activate, deactivate* - The commands instruct an ShM to trigger activation or deactivation of a specified FRU.

The former applies to FRU devices in the M2 state (Activation Request), waiting for activation by an ShM. Mostly a FRU within this state progresses by being automatically activated, but it is not always the case, depending on the ShM configuration. The command is used to manually cause an ShM to trigger a FRU activation, if the automatic activation is not due to occur.

- *setlocked* - This controls the FRU Locked bit, the purpose of which is to allow or prevent a FRU from automatically progressing from state M1 (Inactive) to M2. This bit is set each time a FRU is deactivated, so as to prevent it from immediately setting on the activation path again. Clearing this bit with the setlocked command allows to reactivate a FRU without physically extracting it from a shelf.
- *frucontrol* - Enables to trigger activities aimed at restarting a specified FRU. The restart options enumerate cold reset, warm reset, or graceful reboot. Not all of those options are required to be implemented on ATCA-compliant FRUs, but the command can examine a particular FRU in order to discover the ones the support for which is present.
- *minfanlevel* - Dictates the operation level the shelf cooling devices should maintain as the lowest. No lower level should be reached, even despite the shelf thermal conditions possibly allowing it.
- *getfanlevel, setfanlevel* - Inspect and set the level of cooling units operation intensity at a given moment.
- *sel* - Facilitates the activity of inspecting the SEL of a specified FRU. The records the given FRU SEL contain can be viewed altogether or in selected sections. The contents of such SEL can be cleared with the use of this command.
- *getipmbstate, setipmbstate* - Determine and alter the operation of IPMB-0 connections at a specified shelf IMPC.
- *sendcmd, sendamc* - Serve as a basis for the application interface for exchanging custom-selected IPMI messages with an arbitrarily chosen IMPC (of both those directly attached to IPMB-0 - sendcmd, and AMC controllers - sendamc). They also enable to encapsulate portions of user-specified data into those messages, which can be further processed in an OEM-defined (not necessarily ATCA-compliant) way at the target device. The commands are responded to with the execution result and optionally with a specified piece of data from the target device.

The default manner a shelf operator communicates with an ShM via CLI consists in typing commands optionally accompanied by parameters. The presented application is a kind of a programmatic operator, the task of which is to automatically generate and issue appropriate commands and then to intercept and interpret their results.

## V. Internal Structure of the Application

The application is based upon cooperation of two elements:

- The Communication Library, a lower-level entity encapsulating network and data exchange routines.
- The Data Processing and User Interaction Layer, a higher-level application layer undertaking data processing and providing a GUI.

The communication library was created with the C++ programming language, while the higher-level layer is based on Java and takes advantage of rich set of Java libraries for GUI development. Those elements cooperate properly via the Java platform component called Java Native Interface (JNI) [11], which enables Java software to use C and C++ code.

### A. Communication Library

The responsibility of this component is to take care of Telnet sessions and issues related to maintaining Ethernet connections. The library encloses all the low-level communication routines used by the application, enabling the higher-level layer to be separated from connection-related activities and fully focused upon the actual ATCA shelf domain. The library performs management over the network sockets used in communication with ShM via its System Manager interface and provides data for transmission with a form capable of being conveyed over the Telnet protocol.

### B. Data Processing and User Interaction Layer

This element provides the actual application functionality. It generates queries and requests regarding specific shelf aspects. Those messages are then set for transmission by the communication library via the Telnet channel to a shelf. The response issued by ShM is interpreted and, with desired pieces of information extracted, processed and prepared for presentation in a graphical form. This higher-level application layer consists of three sub-sections referred to as a processing block, a data composition block and a GUI block (Fig. 6). The purpose of the first one is to perform examination of responses coming from an ShM in order to extract valuable pieces of data. Its operation is based on a set of patterns built with the use of regular expressions. Each pattern is associated with a single type of response to a given CLI command that the application uses. Information resulted from a match of given single or multiple shelf responses against corresponding

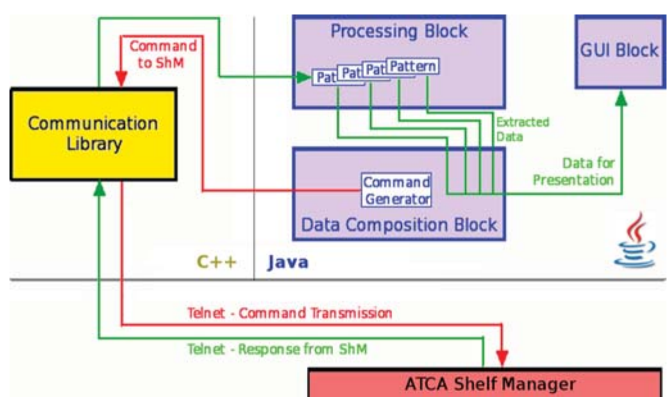



Fig. 6. Diagram of dataflow between the application blocks.

patterns is then assembled by the data composition block to a cohesive data structure, which is then passed to the GUI block to be presented to an operator.

The cooperation of the higher-level layer blocks described above can be presented by the example of a single shelf query cycle. The purpose of this cycle is to obtain information required for creating or updating the graphical shelf front panel representation rendered by the application. The representation engine component prepares and sets for transmission a group of requests, the responses for which are routed to the processing block for extraction of relevant elements. The first request is based on the CLI board command and attempts to determine the occupation of shelf board slots. For those slots, where Carrier Boards are detected, subsequent request based on the CLI fru command is issued in order to search for AMC modules, determine their possible properties and fetch the properties of a Carrier Board itself. As a next step, requests based on fans and ipmc are transmitted in order to obtain information on presence and state of the cooling units in a shelf. The data those processes result in is gathered by the composition block, before the GUI block is capable of using it for refreshing the software reflection of the shelf front panel.

The block architecture of the application facilitates its further development, as each block can be expanded to a significant extent autonomously.

## VI. Conclusion

The ATCA standard appears to be a suitable successor to VME, as well as an appropriate choice as a basis for control systems for high-energy physics experiments like X-FEL. A system based on this new architecture is complex. For maintaining its exceptionally high level of reliability, an efficient and comfortable means of diagnostics and control is crucial. The vendor-provided tools currently offered for this purpose by ShM firmware are not sufficient and particularly not applicable for real-time activities. The software presented in this paper adds the missing real-time capabilities and enables to supervise and manage the ATCA-based system being performed in more effective and convenient manner. What is more, it also provides support for customised management options of Carrier Boards for LLRF.

The diagnostic and control application is being developed and tested using an ATCA shelf managed by an ShM based on the ShMM-500 Management Module from Pigeon Point, accompanied with Carrier Boards provided by RadiSys and custom Carrier Boards designed by the Department of Microelectronics and Computer Science (DMCS) of the Technical University of Lodz. The acceptance tests conducted against this hardware prove that the requirements stated for the application are fulfilled and the application does offer substantial aid during the shelf control and supervision activities. By this means this software is expected to be helpful during development of components for the ATCA implementation, as well as later, when the system becomes operational, for improved monitoring and management. Therefore there can be made an assumption that it will help ensuring the reliability level that the ATCA-based LLRF control system for X-FEL is required to deliver.

Further development plans for the application are intended to focus upon adding support for sensors with plotting voltage and thermal charts, as well as on extending the management capabilities, which will allow to remotely configure ShM Ethernet settings or govern the shelf Electronic Keying resources. Possibility of cooperating with multiple ATCA shelves is also investigated.

## Acknowledgment

The research leading to these results has received funding from Polish National Science Council Grant 642/N-TESLAXFEL/09/2010/0.

## References

[1] Jezynski, T.; Simrock, S. (2006) 'The low level radio frequency system architecture for the european X-FEL.' Proceedings of the International Conference on Mixed Design of Integrated Circuits and Systems, 2006. MIXDES 2006.

[2] AdvancedTCA PICMG 3.0 Revision 2.0, “AdvancedTCA Base Specification”, 28 October 2005

[3] Pigeon Point Systems (2008) 'Shelf Manager User Guide', Release 2.6.1, December 1 2008

[4] Pigeon Point Systems (2008) 'Shelf Manager External Interface Reference' Release 2.6.1 December 2, 2008

[5] Intel Hewlett-Packard NEC DELL, “Intelligent Platform Management Specification version 1.5”, Document Revision 1.1, 20 February 2002

[6] Zawada, A., Makowski, D., Jezynski, T., Simrock, S., Napieralski, A., "ATCA Carrier Board with IPMI supervisory circuit", Mixed Design of Integrated Circuits and Systems, 2008. MIXDES 2008. 15th International Conference on 19-21 June 2008 Page(s):101 – 105

[7] B. Eckel, "Thinking in Java", 4th Edition, Prentice Hall, 2006

[8] Pigeon Point Systems (2005) 'Monterey Linux User’s Guide' Au1550 Edition, Release 2.1.2

[9] Zawada, A.; Makowski, D.; Jezynski, T.; Simrock, S.; Napieralski, A. "Prototype AdvancedTCA Carrier Board with three AMC bays", Nuclear Science Symposium Conference Record, 2008. NSS '08. IEEE, 19-25 Oct. 2008 Page(s):53 – 57

[10] Przygoda, K.; Piotrowski, A.; Jablonski, G.; Makowski, D.; Pozniak, T.; Napieralski, A. (2008) 'ATCA-based control system for compensation of superconducting cavities detuning using piezoelectric actuators' Nuclear Science Symposium Conference Record, 2008. NSS '08. IEEE 19-25 Oct. 2008 Page(s):38 – 43

[11] Sun Microsystems Inc. (2010) “Primitive Data Types”, available from <http://download.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html>

[12] Intel Hewlett-Packard NEC DELL, “Intelligent Platform Management Specification Second Generation v2.0”, Document Revision 1.0, 14 February 2004

[13] Intel Hewlett-Packard NEC DELL, “Platform Management FRU Information Storage Definition version 1.0”, Document Revision 1.1, 27 September 1999

[14] Sun Microsystems Inc. (2010) “The Java Native Interface – Introduction”, available from <http://java.sun.com/docs/books/jni/html/intro.html>

[15] PCI Industrial Computer Manufacturers Group PICMG, AdvancedMC – PICMG AMC.0 Revision 1.0 ECR-002 - 'Advanced Mezzanine Card Base Specification.' 28 April 2006

[16] Sun Microsystems Inc. (2010) “The Java Native Interface – Getting Started”, available from <http://java.sun.com/docs/books/jni/html/start.html>

[17] Sun Microsystems Inc. (2010) “Overview – Java Platform SE 6”, available from <http://download.oracle.com/javase/6/docs/api/>

**Andrzej Napieralski** received the M.Sc. and Ph.D. degrees from the Technical University of Lodz (TUL) in 1973 and 1977, respectively, and a D.Sc. degree in electronics from the Warsaw University of Technology (Poland) and in microelectronics from the Université de Paul Sabatié (France) in 1989. Since 1996 he has been a Director of the Department of Microelectronics and Computer Science. In 2002 he has been elected and in 2005 reelected as the Vice-President of TUL. He is an author or co-author of over 900 publications and editor of 18 conference proceedings and 12 scientific Journals. He supervised 44 PhD theses; four of them received the prize of the Prime Minister of Poland. In 2008 he received the Degree of Honorary Doctor of Yaroslaw the Wise Novgorod State University (Russia).

**Jan Wychowaniak** graduated from the Technical University of Lodz in 2010, earning MSc in Electronics and Telecommunications, in the field of microprocessors systems and reprogrammable devices. He is a PhD student at TUL, with the main fields of interests in software development and digital systems. He is involved in development of diagnostics and management facilities for systems based on the ATCA and uTCA architectures.

**Dariusz Makowski** received PhD degree in electrical engineering at the Department of Microelectronics and Computer Science Technical University of Lodz in 2006. His main areas of interests are digital electronics, embedded systems and programmable devices. He is engaged in the development of xTCA standards for High Energy Physics. He is involved in the design of distributed data acquisition and control systems based on ATCA, MTCA and AMC standards.

**Pawel Predki** graduated from the International Faculty of Engineering at the Techical University of Lodz in 2009 with an MSc degree in Telecommunications and Computer Science. He continues his education as a PhD student at TUL and is involved in many projects. His main fields of interest are embedded systems and digital signal processing. He also deals with control systems and data acquisition systems based on the ATCA and uTCA architectures.