Embedded System for Remote Control of a 3D Camera Rig

Paweł Plewiński, Dariusz Makowski, Aleksander Mielczarek, and Andrzej Napieralski

***Abstract*—The production of stereoscopic motion pictures has been recently getting increasingly popular. To provide the best quality of the resulting image, the parameters controlled must precisely correspond to the values calculated by the stereoscopic image analysis platform or chosen by a camera operator. Currently no integrated system exists, which can provide remote monitoring and control of the parameters of 3D rig and cameras on it. The parameters should be controllable both by the Rig Controller software, tablet application or hand controller.**

The paper discusses an innovative system for remote control of 3D camera rig created within the Recording of 3D Image (ROS3D) research project. The system controls several parameters of both camera rig – stereo base, convergence and camera lenses – focus distance, aperture and focal length. Several approaches of implementation of a solution for the aforementioned problem are presented and compared.

The first proposed approach used custom servo motor controller with CAN bus as the communication interface between connected devices, i.e. Rig Controller board and hand manipulators. Another solution contains a commercial servo motor controller connected to the Rig Controller board via RS232 and commercial hand controller. D-Bus Inter-Process Communication (IPC) system is used to communicate between the servo motor controller drivers and Rig Controller software.

***Index Terms*—3D Camera Rig; Remote Control; 3D Image Recording; Embedded Systems**

I. Introduction

RECORDING of 3D Image (ROS3D) is a research project aimed at the research of 3D image analysis methods, integrated with development of IT tools helping to automatize 3D image recording. [1] One of the key elements of the project is the system for remote control of 3D camera rig's and camera lenses' parameters.

Obtaining the correct configuration for best quality of 3D image is a difficult task, which requires long tuning of parameters by a group of skilled technicians. One of the aims of the ROS3D system is to automatize the setup process to the highest extent possible. In calibration mode the Rig Controller will adjust the stereo base and convergence of the stereoscopic rig based on the real-time image analysis provided by the Image Analyzer also created within the ROS3D project [2, 3, 4].

P. Plewiński, D. Makowski, A. Mielczarek and A. Napieralski are with the Department of Microelectronics and Computer Science, Lodz University of Technology, Łódź, Poland (e-mail: pplewinski@dmcs.pl).

The research leading to these results has been supported by the National Centre for Research and Development (NCBR) within the projects "Registration of Stereoscopic Image" (acronym ROS3D), agreement no. UOD-DEM-1-023/001; under DEMONSTRATOR+ programme and "Hardware and software for automation of 3D filming process"; agreement no. INNOTECH-K3/HI3/16/227587/NCBR/14 under INNOTECH.

The article describes the development process of hardware and software solution aimed at helping in producing stereoscopic movies by making it possible to remotely control several parameters of 3D camera rig with the use of direct current servo motors and explains the design decisions taken.

II. Requirements

The aim of the system is to set the parameters via servo motors given by the user. The supported parameters with the supported range for both lenses and rig are presented in Table 1. The axes for lenses must be controlled simultaneously to ensure the correct stereoscopic effect in the image. Errors in this process directly affect the viewing experience, as the incorrect 3D image may cause inconveniences to the audience, such as nausea or eye strain [5]. High precision of control is essential to provide constant, high quality of the resulting image, since even small inaccuracies in setting the parameters may result in observable distortion of picture or notably impede the process of calibration.

The system provides two control methods – with the Rig Controller developed within the project and with hardware manipulator. Preferably, it should be possible to use both types of control at the same time.

TABLE I.
Parameters supported by the system

Parameter	Unit	Supported range
Lens focus distance	m	0 – ∞
Lens aperture width	f-stops	*f*/0.7 – *f*/64
Lens focal length	mm	0 – 1000
Rig baseline	mm	0 – 200
Rig convergence	degrees	0 – 180

III. General System Architecture

A. Hardware solutions

Obtaining possibly low weight and dimensions of the rig server and motor controller was one of the priorities in the design, since these parts are to be installed on the camera rig, which imposes strict limits in terms of both of these parameters.

Figure 1 presents the architecture of components of the system. The dotted lines show the physical separation of the respective parts of the solution.

ISSN 2080-8755

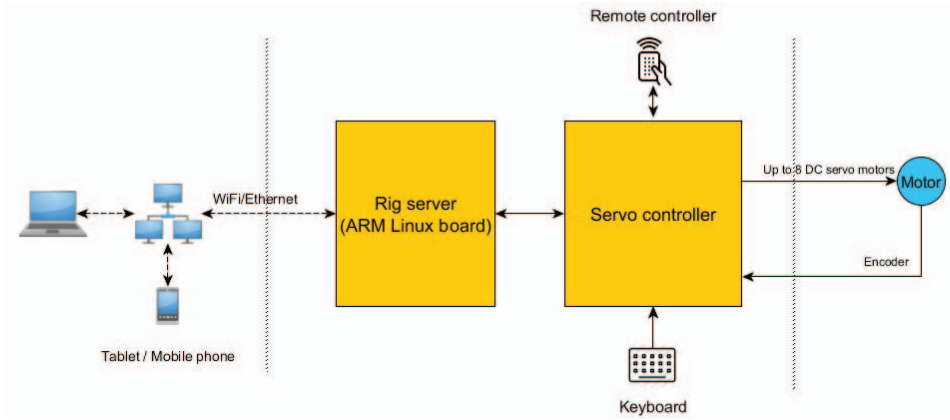

Fig. 1. Architecture of the physical components of the system.

1) *First prototype*

The first implementation of the motor controller was internally designed by the Department of Microelectronics and Computer Science (DMCS). It uses STM32F303 ARM Cortex-M4 processor. Four servo motors are controlled with a pair of TB6612 PWM-controlled motor drivers [7]. The driver was chosen after testing several ICs on the first prototype board.

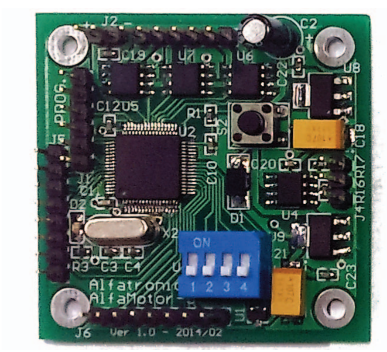

Fig. 2. Servo motor controller board.

CAN protocol was chosen for communication with other elements of the system. The protocol utilizes CAN 2.0A frames (Fig. 3) [8]. It was designed to support control from multiple sources, e.g. Rig Controller and hand manipulator. Each device is assigned an individual address on CAN bus with the use of DIP switches.

The supported commands include moving the motor on an axis to a given position or by a given number of steps, automatic calibration of the axis, reading the range of motion in steps obtained after calibration and reading the status of the device or individual axis.

Later in the design process, SPI communication protocol was added. The custom SPI protocol provides the same functionality in terms of control of the device as CAN protocol, yet with better transfer speeds achievable. Moreover, it makes it easier to integrate with a single board computer used in the project, which was not equipped with CAN peripherals.

2) *Chrosziel Aladin MARK II motor controller*

The described solution can also use Chrosziel Aladin MARK II, a commercially available motor controller dedicated for use in systems related to image recording with support for up to eight motors. The receiver stores a database of axis parameters and correction tables for individual lenses; it also supports synchronization of lens parameters for use in stereoscopy. It has support for own hand controller, which allows simultaneous control over up to 5 different axes with the use of precise manipulators [8].

The Aladin MARK II motor controller can be connected to another system with RS232 or RS422 standard, using a proprietary protocol provided by the manufacturer. The supported commands include moving the motor on an axis to a given position, automatic calibration of the axis, reading the status of the device or individual axis and setting the control of an individual axis either to the hand controller or to the external system connected.

Fig. 4. Chrosziel Aladin MARK II servo motor controller and hand manipulator [8].

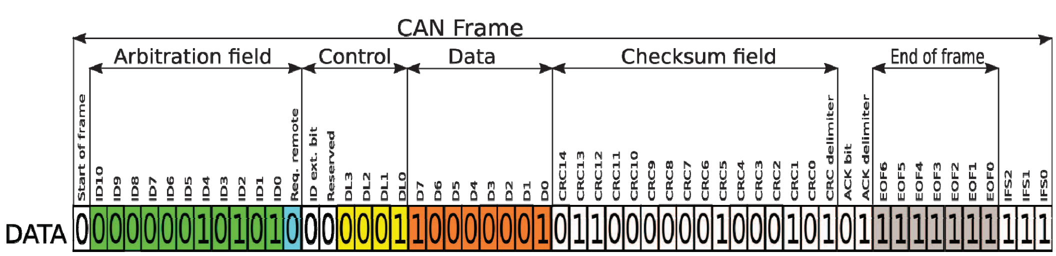

Fig. 3. CAN 2.0A frame format [6].

3) Prototype Rig Controller module

Wandboard Quad ARM board is used as the main logic board of the Rig Controller. It uses a quad core Freescale i.MX6 Quad ARM Cortex A9 processor and integrates 2 GB of DDR3 RAM, Gigabit Ethernet, and IEEE 802.11n wireless network card and provides two UART ports and one SPI port [9]. Processor from this family was selected to provide the possibility of extending the functionality and to keep possibly high compatibility with other components of the system, namely the Image Analyzer, which uses a similar solution. Favorable performance compared to energy use, the aforementioned wide range of connectivity options and small dimensions (9.5 cm x 9.5 cm) were other important factors.

Fig. 5. Wandboard Quad ARM board [9].

The design of the board as a detachable processor board in a form of EDM module and separate board with peripherals and power supply makes it possible to create own carrier board created on the base of the available project thus decreasing the size and weight of the Rig Controller and gaining access to greater number of input/output ports, which are not exposed on the standard peripherals board.

4) Final Rig Controller hardware

The aforementioned Rig Controller module has been evaluated both in laboratory and real life conditions – in studios and on movie sets. These tests led to the conclusion that, while the functionality offered by the discussed system substantially helped in the filmmaking process, the size of all the necessary components, i.e. the Rig Controller single board computer in the case and wireless router was not acceptable in all situations. Therefore, the design needed to be minimized to better fit in the limited space on the stereoscopic rig.

What is more, during the tests executed in the course of the ROS3D project, it turned out that the cabling of the stereoscopic rig was regularly a point of failure. Initially, the synchronization signals such as genlock and timecode, which need to be delivered to both cameras on the rig, were connected with the use of splitters hanging loose on the rig. Not only has it resulted in the cables regularly falling out of sockets, but also complicated the cabling, which resulted in difficulties during setting up the camera system as well as problems to identify possible failed connections. Thus, a requirement has emerged to design a solution, which could better arrange the cabling of the system.

To accommodate the aforementioned requests, the decision was taken to design a new hardware for the Rig Controller. To save a significant amount of space, a new Single Board Computer was selected, namely Banana Pi BPI-R1 [10]. The unique feature of this board is that it includes a 5-port Gigabit Ethernet switch and IEEE 802.11n wireless network card. Other interfaces exposed on 2.54 mm headers include several UART-s, SPI and I^2C.

The processor on the board is Allwinner A20 dual-core Cortex A7 [10]. While it has lower computing power than the Freescale i.MX 6 found on the previous prototype, the tests have proven that it is sufficient to perform all the tasks required from the Rig Controller.

To improve the cable organization on the rig, a custom Printed Circuit Board (PCB) has been developed. The board is responsible for delivering power to the SBC (either from integrated battery or external power supply) and routing the signals for the cameras and Aladin controller. Separate signals for genlock and time code are inputted into the device and outputs dedicated for RED cameras are available. The board is attached directly onto Banana Pi board and uses signals from all headers available on it. Figure 6 shows a picture of the discussed PCB.

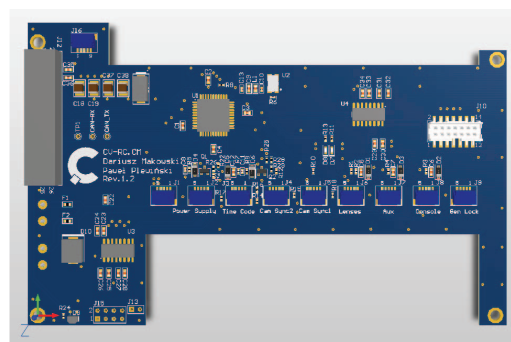

Fig. 6. Image of the power/signal board in the Rig Controller.

A Xilinx XCR3064XL Complex Programmable Logic Device (CPLD) was used to route all the signals on the board [11]. Use of such a logic device makes it possible to modify the routing without making any physical changes to the board. Thus, it is possible to easily adapt the project to other cinematic cameras or extend the functionality, e.g. to implement reading the timecode by the Linux board.

The device has been commercialized as CinemaVision Rig Controller [13]. The final device installed on a stereoscopic rig is presented in Figure 7.

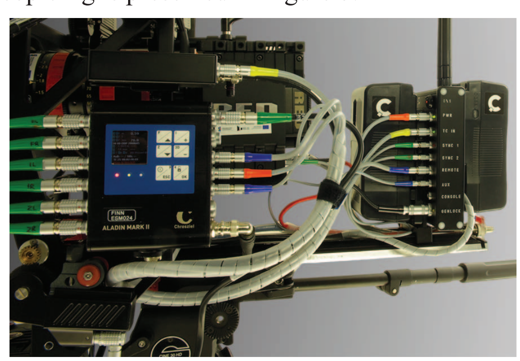

Fig. 7. Final version of Rig Controller installed on a stereoscopic rig [12]

B. *Software*

To achieve the required functionality, several pieces of specialized software needed to be developed as a part of the ROS3D project. The software architecture of the system is presented in Figure 8.

1) *Servo Motor Controller Linux Daemon*

A Linux daemon was created to provide the communication between the main Rig Controller program and servo motor controllers. The needed functionality is control over the requested position of all axes of the controller, requesting calibration of selected axes as well as reading the position and status of the axes. Possibility of simultaneous control over selected axes with Rig Controller software and hand manipulator was proposed as an optional extension of the basic functionality.

D-Bus inter-process communication (IPC) system is used to communicate between the daemon and Rig Controller software. The daemon in first versions supported the abovementioned own servo motor controller via CAN and SPI protocols, further revisions were adapted to support controlling Chrosziel Aladin MARK II driver via proprietary RS232 or RS422 serial protocol.

For evaluation purposes the first prototype was developed in versions for three environments, namely C++ with Qt framework, C++ with D-Bus-c++ library for D-Bus support and spidev lib for communication via SPI protocol and Python 2 with pySerial library for accessing serial port and embedded D-Bus bindings. The implementation using Qt framework was rejected due to large size of binaries necessary to run the application. The version with D-Bus-c++ did not have the abovementioned problem, yet the library was found obsolete, no proper long-term support was provided by the developers. Moreover, it was not possible to fully port the library for the ARM-HF architecture using Yocto build system and the toolchain provided as a part of that project.

Finally, implementation in Python was selected for future development. The benefits such as portability and good support for the constantly developed language and libraries used in the described project greatly overweigh the minuscule drawbacks, mostly in terms of run speed and latency, introduced by the interpreted nature of the language.

2) *Rig Controller Application*

The abovementioned Linux daemons are used by the main Rig Controller application developed by a partner of consortium engaged in the ROS3D Project.

The application built with the use of Python platform and Tornado HTTP framework exposes a dedicated REST Web Service interface, which makes it possible to send commands to the system with the use of mobile applications or web interface. Therefore, it is possible to control the parameters of the setup directly from the graphical interface of the mobile stereoscopic calculator [14].

Rig Controller captures the signals sent by the daemon and stores the values of all parameters. It also handles fault situations, e.g. when some hardware is unavailable. Appropriate information is then logged and cached values are presented to the users.

IV. Evaluation

Contrary to the prototype created within the project, Aladin MARK II is a commercial solution; therefore its availability is notably higher and high build quality and reliability guaranteed by a third party. Moreover, since it is a design dedicated for cinematic applications, dedicated mounts for camera on a rig are provided, while the prototype at the current stage on development does not yet have a case fitted for mounting on a stereoscopic rig.

To better conform to the standards in cinematography, robust connectors, which allow quickly connecting and disconnecting cables, while at the same time preventing it from falling from the socket, such as Lemo, should be used. This is the case in the commercial solution, but has not yet been implemented in the custom design.

Due to the low availability of cinema-grade servo motors guaranteeing high precision and stability of movement, the custom prototype was only tested with standard motors.

The Rig Controller is prepared to work correctly with both solutions using the unified D-Bus API, however full access to the source code in the custom design makes it possible to modify the functionality according to the specific needs of the project.

During the development of the discussed solution multiple tests have been performed in studios and on real movie sets and the problems reported by the users were corrected in the following versions of the solution. In the later versions of the discussed system, no serious issues were reported either in hardware or software.

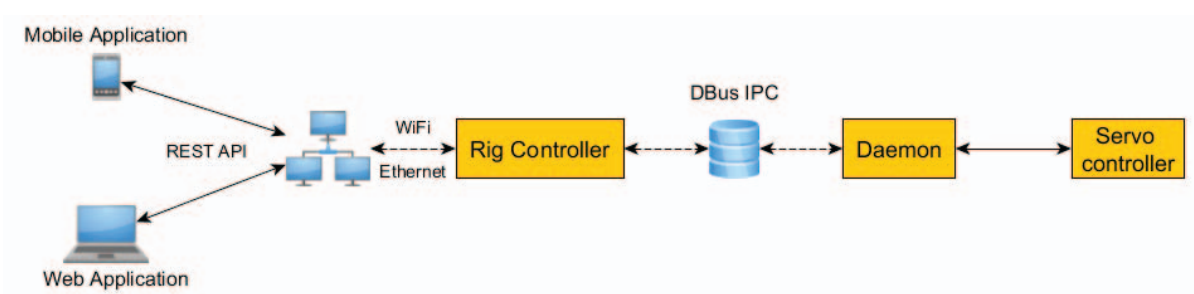

Fig. 8. Software architecture of the rig controller.

V. CONCLUSIONS

The evaluation of the first prototype provided plentiful valuable information on the issues, which can be encountered in cinematic systems and made it possible to refine the architecture of the whole solution, e.g. by specifying the interfaces according to the needs of the users identified in the course of development.

The commercial solution was selected as the main device for further development due to the time schedule issues and necessity to provide fully tested and functionally complete device, which can be immediately installed on the rig. The custom design is still in use for internal prototyping on a simplified rig.

Two versions of Rig Controller hardware have been developed. Thorough tests of the prototype conducted together with its future users made it possible to implement certain vital improvements in the final design, as a result of which the final version gained functionality, which greatly enhances its commercial value.

Before continuing the development of the custom prototype system, full tests on the target stereoscopic rig with cinema-grade servo motors should be conducted to identify all the weak points of the hardware and eventually provide the necessary corrections.

REFERENCES

[1] Stipendium Foundation, “Recording of 3D Image Project”. [Online] Available: http://stipendium.pl/ros3d

[2] A. Mielczarek, P. Perek, D. Makowski, A. Napieralski, “Calibration of Stereoscopic Camera Rigs Using Dedicated Real-Time SDI Video Processor”, Mixdes 2015, *in press*

[3] P. Perek, D. Makowski, A. Mielczarek, A. Napieralski, „Towards Automatic Calibration of Stereoscopic Video Systems”, Mixdes 2015

[4] A. Mielczarek, D. Makowski, P. Perek, *et. al.*, „FPGA-based Image Analyser for Calibration of Stereo Vision Rigs”, Real Time Conference 2016, Padova

[5] F. Zilly, M. Müller, P. Eisert, P. Kauff, “The Stereoscopic Analyzer –An Image-Based Assistance Tool for Stereo Shooting and 3D Production”, 2011

[6] CAN in Automation , “CAN Specification 2.0”. [Online] Available: http://www.can-cia.org/fileadmin/cia/specifications/CAN20A.pdf

[7] Toshiba, “TB6612FNG Datasheet”, 2007. [Online] Available: http://toshiba.semicon-storage.com/info/docget.jsp?did=10660&prodName=TB6612FNG

[8] Chrosziel, “Aladin MARK II Datasheet”, 2012. [Online] Available: http://www.chrosziel.com/data/chrosziel/produkte/documents/Aladin2_remote_control_E.pdf

[9] Wandboard.org, “Wandboard User Guide”, 2012. [Online] Available: http://www.wandboard.org/images/downloads/wbquad-revb1-userguide.pdf

[10] Sinovoip, “Banana Pi BPI-R1 Specifications”, 2016, [Online] Available: http://www.banana-pi.org/r1.html

[11] Xilinx, “XCR3064XL Product Specification”, 2008 [Online] Available: https://www.xilinx.com/support/documentation/data_sheets/ds017.pdf

[12] CinemaVision, “Rig Controller User Guide”, 2016, [Online] Avaliable: http://cinemavision.com/storage/app/media/hardware/CV.RC.UG%20EN%20Rig%20Controller%20User%20Guide%20v2016.06.15.pdf

[13] CinemaVision, “Rig Controller”, 2016, [Online] Available: http://cinemavision.com/rc

[14] M. Wlodarczyk, M. Borzecki, T. Pluciennik, A. Skwarna, „Mobile Application for Remote Control of Stereoscopic Images Acquisition System”, Mixdes 2015.

Paweł Plewiński received MSc degree in computer science from the Lodz University of Technology (TUL), Poland in 2016. His main areas of interest are embedded and mobile systems and software development. He has been involved in projects coordinated by such facilities as Deutsches Elektronen-Synchrotron (DESY), European Spallation Source (ESS) and International Thermonuclear Experimental Reactor (ITER).

Dariusz Makowski is currently the Professor Assistant at the Lodz University of Technology, focusing his research on digital control, data and image acquisition or processing. Dariusz Makowski is leading the Control and Data Acquisition (CADAQ) group involved in development of complex, distributed controls systems based on Advanced Telecommunications Computing Architecture (AdvancedTCA), Micro Telecommunications Computing Architecture (MicroTCA) and PXI/PXIe standards. The developments cover the whole path from the hardware design, drivers, low and high-level firmware programming including HMI. Dariusz Makowski holds a Ph.D. and M.Sc degree in electronics from Lodz University University of Technology.

Aleksander Mielczarek received his Ph.D. degree in electronics at the Faculty of Electrical, Electronic, Computer and Control Engineering in 2017. His main areas of interests are high-speed data acquisition and processing, gigabit communication, integrated sensors and embedded systems. He is involved in the development of control and data acquisition systems for European X-Ray Free-Electron Laser (E-XFEL), a project carried by Deutsches Elektronen-Synchrotron (DESY). He also participates in the realization of Piezo Compensation System for European Spallation Source (ESS).

Andrzej Napieralski received the MSc and PhD degrees from the Lodz University of Technology (TUL), Poland in 1973 and 1977, respectively, and a DSc degree in Electronics from the Warsaw University of Technology, Poland and in Microelectronics from the Université de Paul Sabatiér, France in 1989. Since 1996 he has been Director of the Department of Microelectronics and Computer Science. Between 2002 and 2008 he was Vice-President of TUL. He is an author or co-author of over 960 publications and editor of 19 conference proceedings and 12 scientific Journals. He has supervised 45 PhD theses; six of them received the Prime Minister of Poland prize. In 2008 he received the Degree of Honorary Doctor of Yaroslaw the Wise Novgorod State University, Russia.