

IT Infrastructure for Research and Teaching Combining an Online Lab, a UAV and Audio-Visual Communication

<http://dx.doi.org/10.3991/ijoe.v9i5.2729>

Thomas Volkert, Tobias Simon, Karsten Henke, Steffen Ostendorff
Ilmenau University of Technology, Ilmenau, Germany

Abstract— From previous experiences with student work, we figured out that an interesting real studying object highly motivates students to understand the theory by hands-on experiments. Based on our former work, this paper introduces a new concept, which combines three of our research topics to one complex infrastructure. We start with our existing online lab “*REAL*” and describe how it can be used in order to provide an experiment environment, which is available independent of laboratory opening times. Based on this, we explain how a new experimental platform can be integrated, which is focused on the stabilization of a gimbal-mounted UAV. In this context, different control techniques from different engineering disciplines such as control theory, fuzzy control, neuronal networks and reinforcement learning can be applied. As audio-visual communication solution for experiment coordination, we describe how the video conferencing tool “Homer Conferencing” can be used. This enables the teacher to give direct feedback to the students. Our overall approach is intended to provide a flexible infrastructure, which can be used among different experiment locations.

Index Terms— control engineering education, distance learning, unmanned airborne vehicles, video conferencing tools, virtual and remote laboratories, Web-based design tools.

I. INTRODUCTION

The Integrated Communication Systems Group at the Ilmenau University of Technology has over 10 years of experience in the area of integrated hard- and software systems, dealing with web based teaching [1]. In order to communicate a complete picture of the subject matter, the students also have to pass a hands-on examination in our labs. One example for tasks during this examination is to design an algorithm for a digital control system that controls one of the physical systems, e.g., an electro-mechanical model of an elevator, a process cell¹ or a positioning table. With our hybrid² online lab, we want to offer the students an environment that is close to reality. In comparison to simulations, disturbances can appear and lead to failures of the control algorithm under real laboratory conditions. They cannot be detected under virtual lab conditions. Therefore, it is important to include

such real disruptive factors for a realistic representation of practical conditions. Our lab gives students the chance to check their prepared control algorithms by the help of web based simulations and web based remote control of the physical system. The latter can be influenced interactively by students. Such interactive online labs [2-4] provide an experiment environment for a broad audience and are independent of laboratory opening times.

Additionally to integrated hard- and software system, unmanned airborne vehicles (UAVs) are one of our main research topics. In our previous work [5], UAVs were used as autonomous, mobility-controlled and reconfigurable network nodes in the context of communication networks. The main goals in this context were: creating a swarm of UAVs for creating airborne ad-hoc networks, airborne network node localization, placement of mobile nodes for connecting partitioned networks, and delay-tolerant message exchange in sparse communication networks. Besides this theoretical work, various practical and technological challenges have been addressed such as implementing device drivers, service/component models, logging frameworks, sensor calibration, and filter/controller design. Each of these areas has its own challenges. But the resulting overall system has to be stay manageable and safe to be used.

One further research topic of our work is the open source video conferencing tool and multimedia test bed „Homer Conferencing“(HC). It is available as cross-platform implementation for Windows, Linux and OS X [6]. The general software development is based on C++ and was started in 2008. Since the beginning the main focus of HC was on audio and video (A/V) streaming in the context of live conferencing. Additionally, the software supports A/V recording and screen casting (broadcast of the live picture from the local desktop). One of the main design principles of HC was to provide all application services in a flexible operation mode. For this purpose, the “Session Initiation Protocol” (SIP) [7] and its sub protocol “Session Description Protocol” (SDP) [8] were selected. They can be used in centralized as well as decentralized mode. For each conference session an audio and a video stream can exist in bidirectional directions. For the negotiation of the required A/V stream parameters, HC utilizes the features of the SDP based hand-shake mechanism. For the actual A/V transmissions, HC applies the “real-time transport protocol” (RTP) and its sub protocol “real-time transport control protocol” (RTCP). HC implements synchronization mechanisms which compensate time deviations within a stream and also between streams, e.g., for compensating drifts in A/V

¹ A machining center with circulating storage is called a process cell. It is commonly used for production processes in different industrial areas.

² Hybrid online labs provide both remote experiments on real electro-mechanical models (physical systems) in the remote lab as well as simulation models of these physical systems in virtual labs

playbacks. Currently, the HC software is used in the research for evaluating new network protocols and architectures [9]. Key benefits of HC are its high expandability and the provided possibilities to influence the actual transmission parameters. We already used the software to record measurement values per logical A/V stream [10].

In the following chapters, we describe how we combine the three previously described research areas in order to create a new infrastructure for online learning and experimenting.

II. CONCEPT OVERVIEW

The motivation for combining three research areas in the existing online laboratory is based on previous experience with student work. We learned that students are highly motivated to contribute to such complex and challenging platforms like UAV systems. But the restricted access to the lab and the limited numbers of available hardware platforms are very limiting factors. In

Figure 1, our proposed overall system architecture is depicted. It includes an UAV mounted on a 3D gimbal structure on the rightmost. A mounted camera is streaming the real time state of the experiment using the Homer Conferencing software. The iLab infrastructure [3] is used in order to manage the possible experiment time-slots and granting access to experiments. The teacher can be connected (on-demand) via remote desktop to the experiment control server, where the experiment administration is implemented. The video feed is distributed in a 1:n fan-out scheme from the experiment control server to each individual student's PC running the Homer Conferencing software. The teacher can see the students and also the experiment, while the students can only see the experiment. Using such audio-visual feedback and additional monitoring data from the UAV, the teacher can judge the quality of the implemented control strategy of each individual student. All of these three constituent parts are described in detail in the following. They have been proven to work separately.

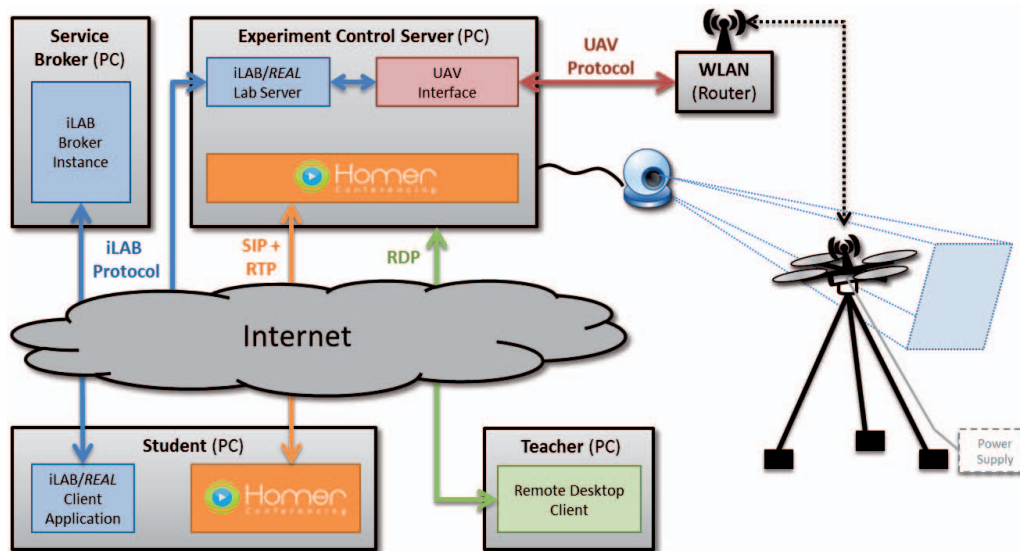


Figure 1: Overall system architecture

III. THE *REAL* INFRASTRUCTURE USING iLAB

In this section, we describe a hybrid interactive online lab supporting all the design steps for complex control tasks in order to control various electro-mechanical models - the *REAL*³ infrastructure [2]. It utilizes the “iLab Shared Architecture” of the MIT in order to coordinate with other experiment locations. The iLab architecture provides a standardized concept for online laboratories. The architecture allows a worldwide interaction and coordination between different experiment locations. The general setup of the iLab architecture is depicted in Figure 1 as blue boxes on the leftmost. It supports multiple client applications running each on a single student PC. They request for an experiment ticket from the service broker, which knows about all available iLab lab servers and their attached experiments. Additionally, the service broker is responsible for user authentication. The iLab lab server

controls the actual experiment and is informed by the iLab service broker instance about new tickets. It validates each ticket and grants the student access. If the access is granted, the student PC starts a direct communication with the additional experiment management platform, which is our *REAL* infrastructure in the scenario of Figure 1.

Figure 2 illustrates an example use-case of our Remote Lab using the *REAL* infrastructure for managing experiments. In general, the *REAL* client is used as real time display of the current experiment state and as input interface for experiment parameters, e.g., the control algorithm. On the opposite, the server side “remote lab” consists of three parts:

- an internal remote lab bus to interconnect all parts of the remote lab,
- per control unit a bus protection unit as interface to the remote lab bus, which protects the bus from blockage, misuse and damage,
- and per physical system a protection unit, which prevents the systems from deliberate or accidentally caused damage.

The goal of the *REAL* (Remote and Applications Laboratory) system is to provide new ways and opportunities for remote controlling and remote observation of real processes (e.g., in the fields of control engineering, robotics, tele-control engineering).

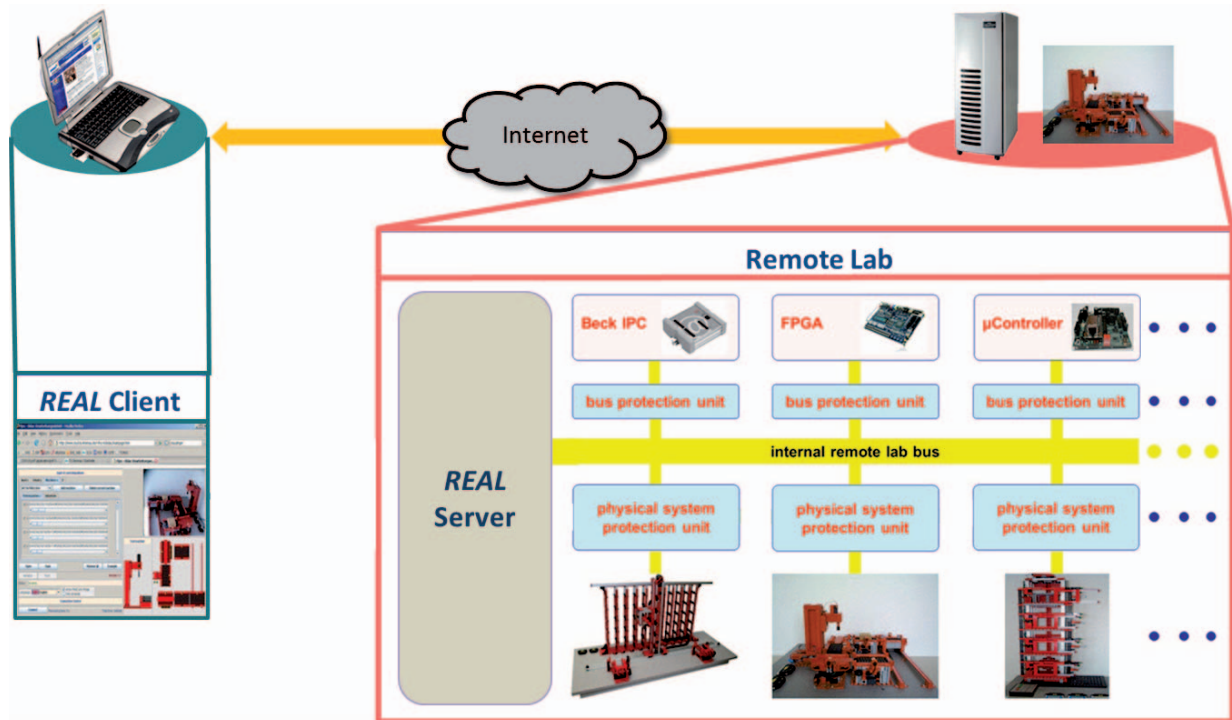


Figure 2: A *REAL* client connected to remote lab

Additionally, the *REAL* infrastructure supports:

- using simulation models of the physical system,
- step-by-step and parallel execution of algorithms,
- visualization of the simulation process by the help of tools (also used for specification),
- test pattern generation,
- and code generation for hardware and software synthesis.

During the entire experiment time, the students are enabled to adjust their algorithms in case of faults. This implements a good base for rapid prototyping in order to achieve a fault free solution. For further details consider [1, 11-15].

IV. THE UAV INFRASTRUCTURE

In order to extend the existing online lab, we propose to include our “unmanned airborne vehicle” (UAV), also denoted as “quadrocopter”. Our idea is to integrate this platform in order to create an experiment focusing on system stabilizing. In this context, algorithms from various engineering disciplines, e.g., control theory, fuzzy control, neuronal networks and reinforcement learning can be applied. Students solve an inter-disciplinary challenge. Besides learning aspects, other research topics and fellow universities could also use the proposed infrastructure. The UAV is basically composed of a central platform with four arms attached to it. At the end of each arm a combination of motor and propeller is mounted to generate a thrust and reaction torque. Two opposite motors constitute a pair, rotating in the same direction, as shown in Figure 3. In this configuration, hover flight is realized by compensating the individual torques applied to the center of mass. The overall thrust is the sum of the individual thrusts of each motor. Roll, pitch and yaw

movement is accomplished by changing the speed of the individual motors. Since it is impossible to generate forces other than in the direction of the yaw axis, the system is under-actuated, which is specifically interesting and challenging for control theory students. In contrast to other approaches, our experiment platform doesn’t apply a dedicated microcontroller for running state estimation and attitude control. Instead, a small general-purpose “computer-on-module” (COM) based on “Gumstix” [16] is running a Linux operating system with real time extensions. It executes the controller code. The processor module runs at 600MHz, features a floating point unit and contains a WLAN interface. The Linux operating system already provides abstraction for multi-threading, memory management, file systems, networking and peripheral hardware, which can be used without extra effort [5]. Such an integrated hardware/software system is ideal for students with a computer science, electrical engineering, or control theory background. It can be used in order to gain a first experience in complex systems design. Various individual properties and restrictions of the system need to be taken into account for achieving a reliable and predictable control strategy. This includes:

- operating system characteristics such as virtual memory, inter-process communication, mutual exclusion,
- measurement limitations for sensors, noise considerations and filtering,
- hardware limitations such as computation power, physical memory and I/O resources,
- and physical system dynamics and actuator restrictions.

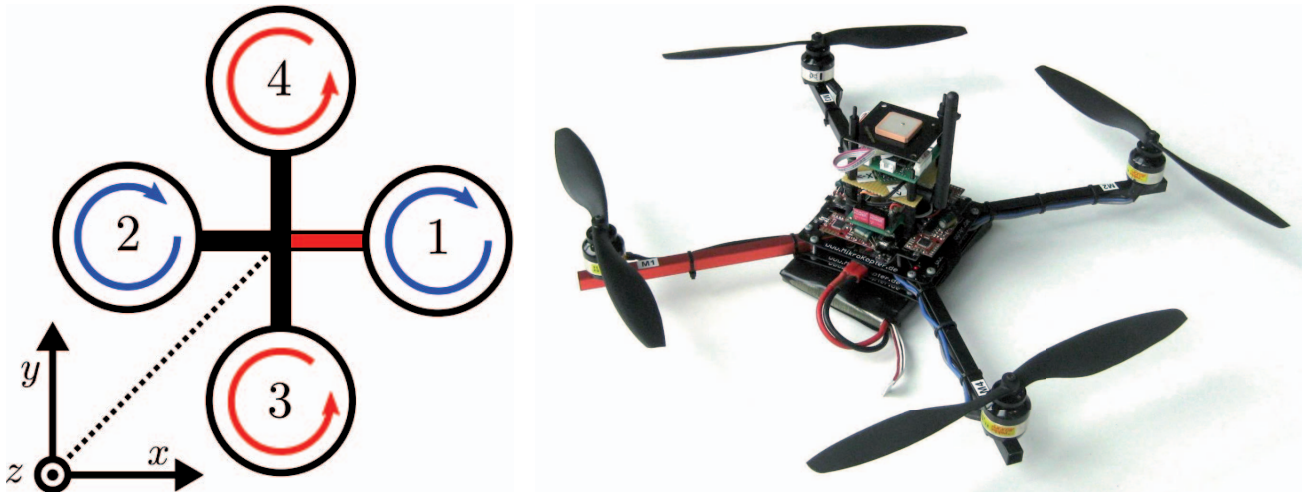


Figure 3: The UAV configuration and an example setup

Combining the presented hybrid online lab with the UAV extends the practical use of the online lab by new complex design tasks. The above mentioned properties and restrictions have to be taken into account for such UAV based experiments in the online lab. In detail, the task for the student can be to simulate and implement a control strategy, which stabilizes the UAV mounted on a gimbal structure, as shown in Figure 4 on the left-hand side, and stabilizes it to a defined 3D orientation set point. After stabilizing, the impulse response of the system to various 3D set points is measured and evaluated.

In order to provide various difficulty levels, a switchable fan, mounted horizontally close to the UAV, is used to generate additional wind disturbances, which have to be tackled by the student's control algorithm. The resulting control strategy of the student needs to be implemented in C/C++ programming language due to performance restrictions of the computer module. This code is uploaded to the UAV's computer module, where it is executed and delivers control feedback at a constant rate of 100Hz. In each step, a 9-dimensional input vector, consisting of 3D gyroscope, 3D orientation (Euler angles), and a 3D orientation set point is provided. The input vector is used to compute a 4D output vector consisting of the four individual motor speed set points. In the future, a reference model is required to check that the motor set points are within a certain safety range in order to protect the experiment hardware against damage.

V. HOMER CONFERENCING

For observing the current experiment state, we suggest to apply Homer Conferencing (HC) [6, 10]. It can be used as audio-visual communication platform to coordinate experiments based on live video conferencing sessions. Based on this, direct feedback from the teacher to each student as well as broadcasts to all students is possible. Moreover, it avoids undesired interaction between students during an experiment. Figure 4 gives an overview of the teacher's view of HC and shows that a teacher is able to interact with all students. He is enabled to give explicit feedback either to one or to all students simultaneously by text messages. The HC instance of

Figure 4 is horizontally split into four parts. The leftmost part lists shows known contacts. For example, this list can correspond to the list of students, whose participation in experiments was acknowledged for the respective day. Automatically generated contact lists are imaginable.

The next part (in Figure 4) near to the contact list is the BROADCAST view. It depicts the local live video, which is simultaneously transmitted to all conference participations (students). In the use-case scenario of our online lab, HC shows the live picture of the experiment platform. Below the live video a special audio level display is integrated, which gives optical feedback about local audio inputs. Moreover, the teacher could also use the local microphone to speak with students and give hints. The same view is depicted in the third and fourth part. They represent a one-to-one communication relation per student. Figure 4 depicts an example situation with two students. In general, the HC software doesn't limit the amount of participating students. But the available network bandwidth is a limiting factor in relation to the used A/V transmissions.

In contrast to the teacher's view, each student has only one communication partner: the teacher himself. In general, a student is only allowed to ask questions to the teacher and may use the live video stream from the experiment platform to observe the result of either his work or the work of other fellow students. Figure 5 shows the student view of HC. According to the descriptions about the teacher side, the student also sees his local BROADCAST view on the left side. However, the teacher's instance of HC doesn't provide any kind of message relay for incoming BROADCAST messages. Therefore, a BROADCAST message from a student reaches only the teacher side. On the opposite, a student receives BROADCAST messages from the teacher on the leftmost part of his HC instance, as shown in Figure 5. On the rightmost part, the student may directly interact with the teacher in a one-to-one chat. The student may use this to receive some explicit hints for the experiment. Additionally, he is able to observe the experiment by the help of the depicted live video stream from the experiment.



Figure 4: Homer Conferencing at teacher side

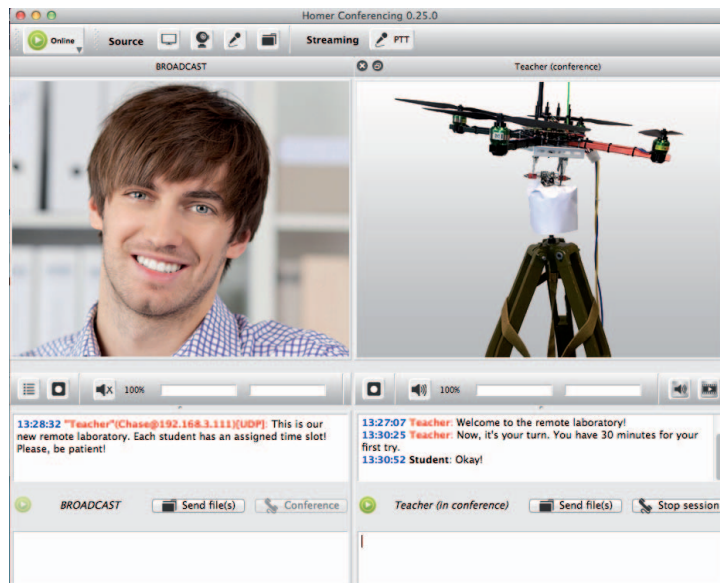


Figure 5: Homer Conferencing at student side

In general, the HC software supports high as well as low quality streaming. Even a low bandwidth Internet access for home users provides the required capacity to receive a live video stream from the experiment of the online lab. However, it might be possible that the uplink of such a home Internet access doesn't provide a high enough bandwidth. For such cases, the outgoing A/V stream can be disabled in order to decrease the acquired network resources. We already tested such a scenario consisting of bidirectional A/V streams and figured out that a common DSL 6000 (6 Mbit/s downstream) line delivers the needed minimum of network resources for a stable live video conferencing session. However, we strictly recommend connecting the online lab via a 100 Mbit/s (and better) access port to the Internet in order to prevent bottlenecks when delivering simultaneously the live video stream from the experiment to multiple students. Additionally, the GUI of HC can be used to adjust the actually used parameters of outgoing A/V

streams. This includes the applied codec, the output video frame rate, the output video quality and the resulting output bit rate. These settings may help to reduce the acquired network resources and might also enable using HC via a low bandwidth Internet access port.

VI. CONCLUSION AND FUTURE WORK

In this paper, we present a concept to improve teaching and research possibilities by combining existing, and already proven elements of an online lab, a UAV and a video conferencing tool. This work explains the concept's feasibility and illustrates the main benefits of all three core components. Firstly, we use the online lab to offer students and researchers an experiment platform, running independent of opening hours and on-site lab access. Secondly, the UAV experiment platform extends the possible application variety of the online lab and increases the range of people that have potential interest in using the online lab. Thirdly, the video conferencing tool "Homer

Conferencing“ is applied to provide an ideal audio-visual communication platform for local and remote lab sessions.

As future work, we plan to extend the proposed UAV experiment by a reference model, like it is already used for the classical experiments of the online lab [15]. Based on such a reference model, it is possible to give the students feedback about the quality of the current solution. Moreover, the teacher may use this quality judgment in order to conclude individual marks or suggest explicit improvements.

Because of the attractive subject and the large number of different and interesting projects, students can work on many challenging topics during their studies. We plan to use this as base and want to further improve the multimedia-based possibilities for interactivity between all involved communication end points: the students, the teacher and the experiment platform.

VII. ACKNOWLEDGMENTS

This work was supported by the “Friends of the Faculty of Computer Science and Automation” and by the DFG Research Training Group 1487 (MOBICOM).

REFERENCES

- [1] K. Henke and H.-D. Wuttke: “Web-based educational tool access”, IASTED International Conference Computers and Advanced Technology in Education (CATE), Rhodes/Greece, June 2003.
- [2] Integrated Communications Systems Group: “REAL-Remote Engineering and Applications Laboratory”, <http://ih7.theoinf.tu-ilmenau.de/applets/index.htm>.
- [3] Massachusetts institute of technology: “iLab project”, web site: <http://ilab.mit.edu>.
- [4] Y. Torroja et al.: “A Modular Environment for Learning Digital Control Applications”, Microelectronics Education, pages 185-188, Marcombo, S.A, 2002.
- [5] T. Simon, A. Mitschele-Thiel: “Micro Aerial Disaster Communication Systems“, IT Rescue Workshop (Informatik2012), Braunschweig/Germany, Sept. 2012.
- [6] T. Volkert: “Homer Conferencing”, web site: <http://www.homer-conferencing.com>.
- [7] J. Rosenberg, H. Schulzrinne et. Al: “SIP: Session Initiation Protocol”, IETF RFC 3261, June 2002.
- [8] M. Handley, V. Jacobson: “SDP: Session Description Protocol”, IETF RFC 3266, April 1998.
- [9] T. Volkert, A. Mitschele-Thiel: “Hierarchical routing management for improving multimedia transmissions and QoE”, in Proceedings of IEEE WoWMoM, San Francisco, June 2012.
- [10] T. Volkert, A. Mitschele-Thiel, M. Becke, E. P. Rathgeb: „Homer Conferencing – a Multimedia Test Bed for Various Experiments and Measurements“, 7th International Conference on Computer Sciences and Convergence Information Technology (ICCIT), Seoul/Korea, Dec. 2012.
- [11] K. Henke, H.-D. Wuttke and T. Braune: “Virtual and remote labs in the Educational Process”, International Conference on Remote Engineering and Virtual Instrumentation, REV2007, Porto, Portugal, June 25-27, 2007.
- [12] K. Henke, H.-D. Wuttke and S. Hellbach: “Laboratory via Internet – new ways in education and research”, Int. Journal of Computers and Applications, vol. 25, ACTA press, 2002.
- [13] K. Henke, H.-D. Wuttke and S. Hellbach: “Laboratory via Internet – new ways in education and research”, Int. Journal of Computers and Applications, vol. 25, ACTA press, 2002.
- [14] K. Henke, St. Ostendorff and H.-D. Wuttke: “A Flexible and Scalable Infrastructure for Remote Laboratories - Robustness in Remote Engineering Laboratories”, The Impact of Virtual, Remote and Real Logistics Labs - ImViReLL2012, Bremen, February 28 – March 02, 2012.
- [15] K. Henke, St. Ostendorff, H.-D. Wuttke and St. Vogel: “A Grid Concept for Reliable, Flexible and Robust Remote Engineering Laboratories”, International Conference on Remote Engineering and Virtual Instrumentation, REV2012, Bilbao, Spain, July 4-6, 2012.
- [16] Gumstix inc.: “Gumstix computer-on-module”, web site: <http://www.gumstix.com>

AUTHORS

Thomas Volkert is with the Ilmenau University of Technology, Faculty of Computer Science and Automation, Integrated Communication Systems Group, 98684 Ilmenau, Germany, POB 10 05 65 (e-mail: thomas.volkert@tu-ilmenau.de).

Tobias Simon is with the Ilmenau University of Technology, Faculty of Computer Science and Automation, Integrated Communication Systems Group, 98684 Ilmenau, Germany, POB 10 05 65 (e-mail: tobias.simon@tu-ilmenau.de).

Karsten Henke is with the Ilmenau University of Technology, Faculty of Computer Science and Automation, Integrated Communication Systems Group, 98684 Ilmenau, Germany, POB 10 05 65 (e-mail: karsten.henke@tu-ilmenau.de).

Steffen Ostendorff is with the Ilmenau University of Technology, Faculty of Computer Science and Automation, Integrated Communication Systems Group, 98684 Ilmenau, Germany, POB 10 05 65 (e-mail: steffen.ostendorff@tu-ilmenau.de).

This work was supported by the “Friends of the Faculty of Computer Science and Automation” and by the DFG Research Training Group 1487 (MOBICOM). Submitted, May, 02, 2013. Published as re-submitted by the authors 15 September 2013.