An Embedded Asterisk Platform Instructional Design to Teach Voice over IP in Information Technology Undergraduate Courses

Using Raspberry PI and Asterisk to Build an Embedded Portable Didactic Tool

M. C. Dias[1,2], L. C. Bezerra[1], D. Morais[1], C. F. Gabi[2] and A. Perkusich[2]

[1]*Electrical Engineering Coordination, Laboratory of Telephony and Convergent Networks – LTCON, Federal Institute of Education, Science and Technology of Paraíba – IFPB, João Pessoa, Brazil*

[2]*Post-graduate Program in Electrical Engineering – PpgEE – COPELE, Electrical Engineering Department, Federal University of Campina Grande – UFCG, Campina Grande, Brazil*

Keywords: Learning, Embedded Platform, Asterisk, Voice over IP - VoIP.

Abstract: This paper shows the building of an embedded Asterisk platform as a key technique for providing Information Technology classes with an extra laboratorial environment for learning Voice over Internet Protocol. Besides building the platform, the students are asked to mount a functional telephony system with it that meets the criteria of low cost, low building complexity and high adaptability to convergent networks. The Raspberry PI version B was the board used to provide portability for the students to work outside the laboratory environment. Asterisk was presented as a useful tool on which the students could create a flexible system that could be easily deployed on convergent networks. This work was developed in the Telephony subject for the Electrical Engineering and the Telecommunications Systems undergraduate courses from the Federal Institute of Education, Science and Technology of Paraíba – IFPB, and presented good results in the students´ learning achievements.

1 INTRODUCTION

Nowadays, part of the theory in technical education is narrowed by what is practiced in the laboratory, mediated by the infrastructure that is offered. Among these theories, one that demands extensive practice is IP Telephony Technology, which is essential to have an entire comprehension in Converged Networks structure. The lab training about these technologies is necessary to meet the need for qualified workforce (Eady and Lockyer, 2013).

Nonetheless, the networking laboratorial infrastructure has an increased complexity and a high cost, due to the purchase of Gateways, IP phones and other several devices, what makes the individual acquisition impractical (Spencer, 2003); (Rowe, 2007).

However, according to Lamar (2012) and Rowe, (2007) students would not be so restricted at the school labs if they could make their own experiments anywhere, by carrying all the resources compatible with the academic laboratories.

The use of free Asterisk Platform in VoIP education has become a very recurrent methodology lately, especially when combined with the experiments on Converging Data Networks. A recent study on this matter was proposed by Dias et al (2013), who presented a successful experience using this software. Another study, based on the configuration of an Embedded PBX Platform, driven by Abid et al (2012), discussed the elaboration of a compact system for VoIP aplications and the details about how to get a better performance.

Our work shows how the building of a portable platform can improve the learning of students from Theory Information classes that work with IP Telephony and Converged Networks. In addition to developing the theoretical content with practice during the construction of the portable platform, students no longer have to hold themselves to the laboratory environment, because they have the possibility to develop experiments in extra-class environments.

In addition to this, the construction of the prototype shown in our work also considered factors such as low cost, complexity of construction and adaptation to converged networks. The main goal is to provide the students with a didactic tool that enables them to reproduce situations and scenarios

Dias M., Bezerra L., Morais D., F. Gabi C. and Perkusich A..
An Embedded Asterisk Platform Instructional Design to Teach Voice over IP in Information Technology Undergraduate Courses - Using Raspberry PI and Asterisk to Build an Embedded Portable Didactic Tool.
DOI: 10.5220/0005489504610466
In *Proceedings of the 7th International Conference on Computer Supported Education* (CSEDU-2015), pages 461-466
ISBN: 978-989-758-107-6

only available previously in the laboratory environment.

The paper is organized in sections, as follows: section 2 will focus on a bibliographical review on the foremost concepts concerning VoIP and a presentation about portable platforms, involving the Asterisk concepts; section 3 will describe the materials and methods applied in the design of the proposed device; in section 4, the results obtained from the construction and usability test will be presented; in section 5, conclusions are discussed.

2 VOICE OVER IP (VOIP) AND ASTERISK CONCEPTS

2.1 Voice over IP (VoIP)

Voice over IP is a set of networking protocols that transport the voice over a TCP/IP data network. The analog waves of the human voice are quantified, digitalized and sent in packets from the origin call point to their destination (Meggelen; Smith and Madsen, 2007); (Hartpence, 2013).

To make this possible, a lot of resources were used: specifics sessions protocols like SIP or H323; codecs to transcode and optimize the sound; and quality of service (QoS) to maintain the integrity on limited connections.

2.2 Embedded Platforms

An Embedded Platform is a micro-processing system completely encapsulated and dedicated to the device, or system, controlled by him. Differently from a computer, whose architecture is diversified on countless ends, an embedded platform has a structure particularly designed to perform specific and predefined operations.

These characteristics result in a considerable economy on resources and components, generally, reducing their material costs, labour costs and, mainly, the physical size of the devices that embrace this kind of system (Ganssle and Barr, 2003).

Embedded platform system devices examples:

- Automotive navigating systems;
- Smartphones and PDAs;
- Biometric access control systems;
- Air-conditioning temperature control;
- MP3 players;
- Printers;
- Networking devices;
- Portable measuring devices;
- Medical monitoring systems.

There are a number of boards available in the market that make possible to implement embedded systems, like the Beaglebone Black (Richardson, 2013), the Cubie Board (Schinagl, 2014), The Intell Galileo (Ramon, 2014), and the Raspberry PI (Upton and Halfacree, 2014). For this work, we chose the Raspberry PI because it's a low cost and well known board that has plenty of working material available. Therefore, the next session shows the main features of the Raspberry PI.

2.2.1 The Raspberry PI

Raspberry PI is a development board, with small dimensions, manufactured since 2012 by Raspberry PI Foundation (UK). The primary goal from this design is to stimulate the computing science learning on children (Upton and Halfacree, 2014).

The board, in its B version (Figure 1), comes with System on Chip (SoC) Broadcom CM2835, which includes:

- ARM1176JZF-S of 700 MHz Processing;
- GPU Video Core IV;
- 512 MB RAM (SoC);
- SD card entry;
- Two USB ports 2.0 ;
- HDMI video and VCA video outputs;
- Ethernet 10/100 (RJ45);
- Connecting port with 26 pins, which 17 are GPIO;
- Audio output via 3,5 mm jack connector.

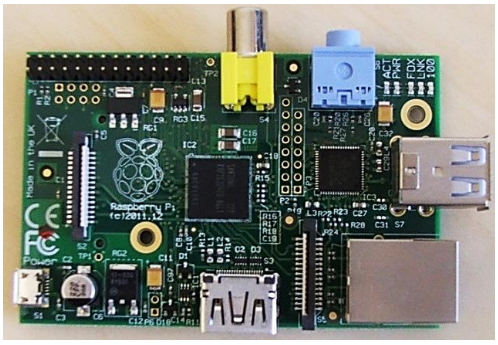

Figure 1: Raspberry PI board, version B.

To reach a larger audience, the Raspberry PI average price needed to be as low as possible. So, the project's creators convinced the North American Broadcom to sell the processors to be used in the boards at very low prices. (Cooper and Knight, 2014).

In fact, this device has been used by children worldwide, and it has been helping them on computing learning. Also, Raspberry PI has been applied in other several projects in many different fields of education, like Automation and Electronics, Programming languages like Python, Web services and others. (Upton and Halfacree, 2014).

2.3 Linux and Asterisk

The Unix is an Operational System created by Ken Thompson in the late 60s, to fulfil the necessity of many researchers whom, in order to proceed with their works, required a multiuser and multitask system, which also could be easily converted to different hardware platforms (Machtelt, 2007); (Hartpence, 2013).

Asterisk is an open source platform, distributed by Digium under a General Public License (GPL), initially developed for Linux Systems, combining all the characteristics of a Public Switched Telephony Network (PSTN) with an integrated set of customizable applications (Meggelen; Smith and Madsen, 2007).

All Asterisk architecture (Figure 2) was elaborated to be highly flexible, intending to provide support on several applications and many signalling profiles originated from public networking. As so, it supplies interfaces to any kind of hardware or software without limitations (Martin, 2009); (Spencer, 2003).

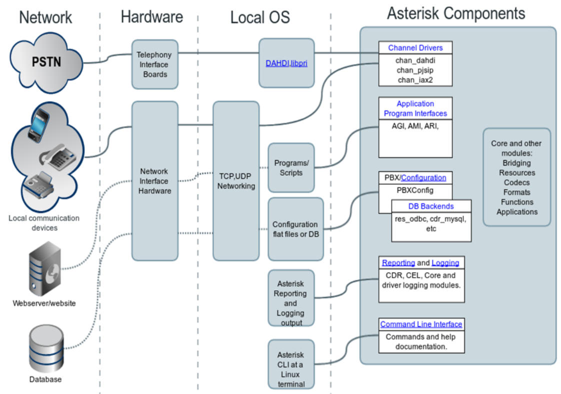

Figure 2: Asterisk Architecture. (Asterisk Project, 2014).

3 MATERIALS AND METHODS

To show the device effectiveness and its applicability, an experiment was mounted under a simple topology, composed of three elements, placing the Embedded Asterisk Platform connected to a LAN and two access peers as presented in Figure 3. The described attempt about the equipment to communicate was monitored by a Network Sniffer, which revealed the data traffic, signalling and connection.

The experiment was divided into three steps:

1. Hardware mount;
2. Embedded systems installing;
3. Base scenario execution and results.

As learning goals, the following aspects were defined: a) Features on embedded systems and free software; b) VoIP definition and its architecture; c) Asterisk server configuration and functioning; d) Protocols operations, as SIP (Session Initiation Protocol) (Hartpence, 2013) and RTP (Real-time Transport Protocol, and its signalling (Meggelen; Smith and Madsen, 2007).

On the first step, scenario preparing has as choice of criteria a versatile topology, easily implemented, taking into consideration the resources availability, without limiting the range of experiments that can be tested. If the scope of applications on this stage was limited, it would lower the amount of knowledge to be put as proof by the students.

The proposal of this scenario (Figure 3) is that the student can, at first, implement it in any place where he/she wants to practice, based on its high portability feature. Also, step 3 is bounded to the understanding of the functioning of the Asterisk server, and how it is possible, through the configuration files, to make and receive calls among IP terminals, whichever they are - telephones or softphones - in a regular computer.

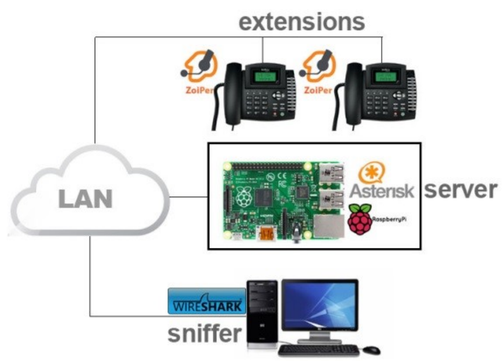

Figure 3: Functionality Test Scenario: proposed topology for the connection between the Embedded Server and the Peers.

Another important concern is related to acknowledge how the SIP and RTP protocols act, its signalling messages and media flow to determinate the connectivity between the peers, the call quality and which error messages are likely to appear on a SIP signalling session.

On the experiment, it was used an LG A550 notebook with I5 2,8 Ghz Processor, 4 GB RAM memory, and a softphone Zoiper to authenticate 3 testing extensions, 3001 and 3002 (Zoiper.com, 2015); (Gonçalves, 2007).

On step 3, we proceeded installing the systems to test the proposed scenario, under the Raspberry PI B version hardware model, provided on IFPB's Telephony and Convergent Networks Laboratory. According to the equipment instruction manual, we started on the peripherals connections. To the LAN connection, we used the RJ 45 ports; the USB port to the keypad; HDMI port to link the monitor, RCA to connect the sound and the standard port to receive the power supply (Figure 4).

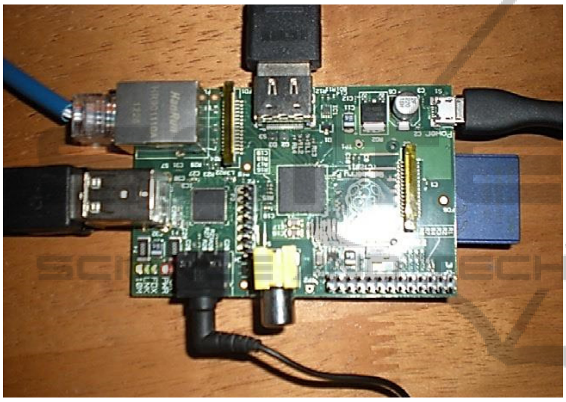

Figure 4: Peripherals connections in the Raspberry PI board.

Once it was connected, we initiated the Operating System install, as shown in Figure 5. Following the recommended version, it was used the RaspbianOS distribution, which is based on the Linux Debian Distribution.

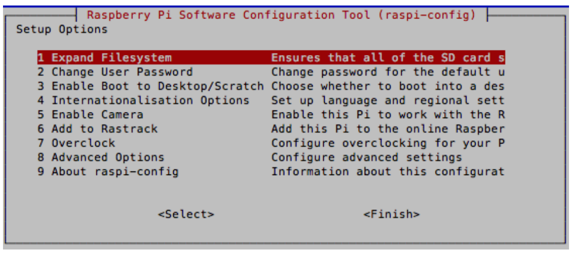

Figure 5: Operating System installing details.

As the OS installation was concluded, the Asterisk software has been installed and configured to make possible the creation of the SIP channels and extensions recognition, as demonstrated in the code below:

```
[general]
udpbindaddr=0.0.0.0
context=default
disallow=all
allow=alaw,ulaw,gsm
nat=yes
canreinvite=no
language=pt_BR

[common](!)
type=friend
secret=rstc2015
dtmfmode=rfc2833
host=dynamic

[3001](common)
callerid=<3001>"RstcVoip1"

[3002](common)
callerid=<3002>"RstcVoip2"
```

In order to the IP phones communicate with each other, the Embedded Asterisk dialling plan must also be created. The extensions.conf file is responsible for this feature in the server. The code presented below is part of a dialling plan made to the proposed scenario.

```
[default]
exten => 3001,1,Dial(SIP/3001,20,tT)
exten => 3001,2,Hangup()

exten => 3002,1,Dial(SIP/3002,20,tT)
exten => 3002,2,Hangup()
```

At last, after configuring the extensions and the dialling plan, the SIP accounts shall be configured on the IP phones to send the request for registration to the Asterisk Embedded server, using the credentials set up in sip.conf file. Figure 6 shows how this configuration is made in the Zoiper Softphone.

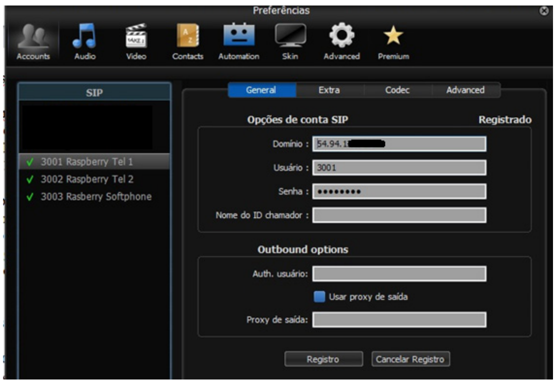

Figure 6: Zoiper Softphone set up.

For monitoring, the Asterisk platform has a console prompt that can exhibit the program current version, the dialling plan, latency among the registered extensions, used ports and on-going connections, as seen in Figure 7.

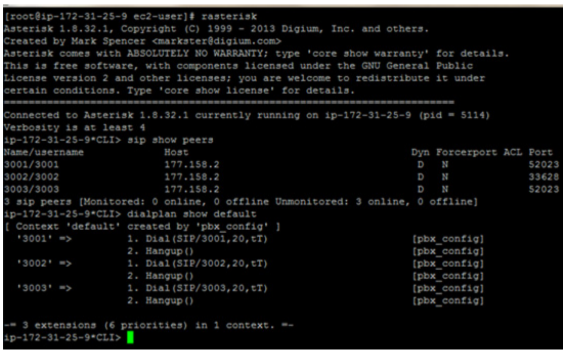

Figure 7: Asterisk platform monitoring console prompt.

4 RESULTS

As proposed in the primary scenario (Figure 2), we observed that the Embedded Asterisk Server does the intermediating role in communications among the devices linked to the LAN. In order to verify the system communication integrity, we used the Wireshark software to check the local networking, capturing the packets that flow between the server and the peers. Figure 8 shows the kinds of packets captured by the tool (Wireshark, 2011).

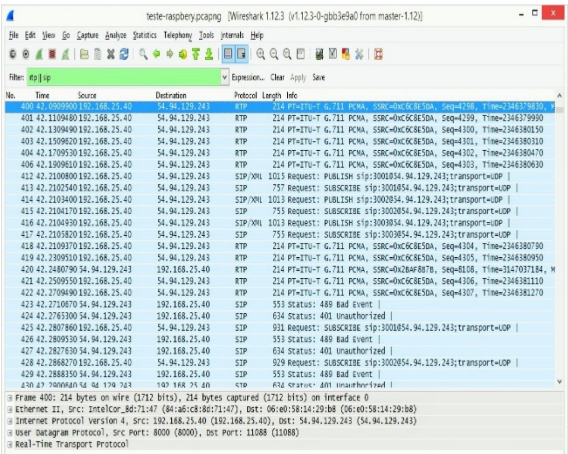

Figure 8: Packets captured with Wireshark Software.

On detailed mode, it was also possible to observe the protocols used to establish the VoIP connection: the SIP protocol was used for peer-to-peer signalling; the Codec (G711) applied to audio codification and RTP protocol for the media flow. The trial sequence and connection request among terminals, ringing signalling, connection ACK, media flow, disconnection request and the final confirmation between the end points are shown in Figure 9.

By using this low cost embedded platform, the student will be able to examine the behaviour of the connection to be tested, accessing detailed information about the integrity of the peer-to-peer link, passing through signalling pattern sequel, codecs usability or even media flow interruption. Knowing about this is useful for problem resolutions proposed by the teacher with scenarios that are much more complex, where the apprentice must develop his/her own solution. Even though a complex scenario cannot be unveiled during class, the portable solution will be able to be carried anywhere inside the student's backpack or pocket thanks to its small dimensions, as demonstrated in Figure 10.

By using this low cost embedded platform, the student will be able to examine the behaviour of the connection to be tested, accessing detailed information about the integrity of the peer-to-peer link, passing through signalling pattern sequel, codecs usability or even media flow interruption. Knowing about this is useful for problem resolutions proposed by the teacher with scenarios that are much more complex, where the apprentice must develop his/her own solution.

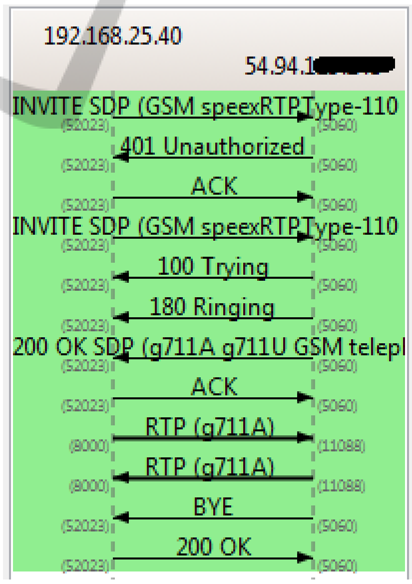

Figure 9: Details about trial connection status, extracted from the Wireshark program on running.

Even though a complex scenario cannot be unveiled during class, the portable solution will be able to be carried anywhere inside the student's backpack or pocket thanks to its small dimensions.

The assessment by the teacher showed that all students correctly answered the questions and that 100% of students were satisfied or very satisfied with the practice. And all of them said that the most attractive platform was mobility.

5 CONCLUSIONS

The results confirm that a portable Embedded Server, besides being completely functional, it is most likely a very important and viable resource in the educational field, particularly in Technology Courses.

Besides the possibility of customization, likewise the Free Software, including Asterisk, the program used to build this tool is the Open Source and it is freely distributed, appearing to be a low cost solution for the educational institutions.

In addition to its computational versatility and portability, Raspberry PI provides us a new perspective in regard to educational material to apply inside and outside classes, amplifying the range of testing scenarios and problem-solving situations to be presented to the students, enriching their academic degree.

In future works, more experiments that use this platform will be developed. Furthermore, this tool will be used during other academic terms so as to consolidate and extend its validation.

REFERENCES

Eady, M. J. And Lockyer, L. 2013. Tools for learning: Technology and Teaching Strategies. In: *Learning to Teach in the Primary School*. Queensland University of Technology. Available from: http://ro.uow.edu.au/cgi/viewcontent.cgi?article=1413 andcontext=asdpapers.

Rowe, D. 2007 *Free Telephony Project: Open Embedded Telephony*. Avaiable from www.rowetel.com/ucasterisk.

Spencer, M. 2003. *The Asterisk Handbook. Asterisk Documentation Team.* Available at www.asterisk.org.

Lamar, Diego G. et al. 2012. Experiences in the Application of Project-Based Learning in a SwitchingMode Power Supplies Course. *IEEE Transactions on Education*, Vol. 55, no. 1, February 2012.

Dias, M. et al. 2014. Using Asterisk as a Tool for Teaching Telephony Subject for Telecommunication Classes. In: *Proceedings of the International Telecomunications Symposium*, São Paulo, August 2014. São Paulo: IEEE.

Abid, F.; Izeboudjen, N.; Bakiri, M.; Titri, S.; Louiz, F.; Lazib, D. 2012. Embedded implementation of an IP-PBX /VoIP gateway. In: *Microelectronics (ICM)*, 2012 24th International Conference on.

Meggelen, J. V.; Smith, J. And Madsen, L. (2007) *Asterisk - The Future of Telephony*. 2nd ed. Sebastopol: O'Reilly Media, Inc.

Hartpence, B. 2013. *Packet Guide to Voice over IP*. Sebastopol: O'Reilly Media, Inc.

Ganssle, J. G. And Barr, M. 2003. *Embedded Systems Dictionary*. San Francisco: CMP Books.

Upton, E. And Halfacree, G. 2014 *Meet the Raspberry PI*. United Kingdom: John Wiley and Sons Ltda.

Cooper, J. And Knight, J. 2014. *Evaluating possible uses of a Raspberry PI in an academic library environment*. D-Lib Magazine.

Machtelt, G. 2007. *Introduction to LINUX - A hands-on Guide*. Fultus Corporation.

Dulaney, 2014. *Linux All-in-One For Dummies*. 5th ed. United Kingdom: John Wiley and Sons Ltda.

Pritchard, S.; Pessanha, B. G.; Langfeldt, N.; Stanger, J. And Deanivro, J. 2010. *LPI Linux Certification in a Nutshell*. Sebastopol: O'Reilly Media, Inc.

Martín, S. G. 2009. *Contribution to Asterisk Open Source Project*. Thesis (MSc). Universitat Oberta de Catalunya.

Asterisk Project, 2014. *Asterisk Architecture, The Big Picture*. [Online]. Available from: https://wiki.asterisk.org/wiki/display/AST/Asterisk+Architecture%2C+The+Big+Picture [Accessed 2/9/2015].

Zoiper.com. 2015. *Free VoIP SIP softphone dialer with voice, video and instant messaging.* [Online] Available from: http://www.zoiper.com/en/voip-softphone/download/zoiper3/for/windows.

Gonçalves, F. 2007. *Configuration Guide for Asterisk* PBX. V. Office Networks Ltda.

Wireshark, 2011.*VoIP calls* [Online]. Available from: http://wiki.wireshark.org/VoIP_calls [Accessed 29/01/2015].

Richardson, M. 2013. *Getting Started with BeagleBone: Linux-Powered Electronic Projects with Python and Java Script.* Paperback.

Schinagl, O. M. 2014. *Getting Started with Cubieboard.* Paperback.

Ramon, M. C. 2014. *Intel Galileo and Intel Galileo Gen 2: Api Features and Arduino Projects for Linux Programmers*. Paperback.