Charging Network for Electric Vehicles

Tiago Pinheiro[1,3], Mário S. Nunes[1,2,3], and Martijn Kuipers[2]

INOV[1] / INESC-ID[2] / IST-UTL[3]
1000-029, Lisbon, Portugal
{tiago.pinheiro,mario.nunes}@inov.pt,
martijn.kuipers@inesc-id.pt

Abstract. This paper proposes a full EV charging network architecture, based on the current test-pilot of a national energy provider. The Electric Vehicle Charging Station (EVCS) follows a modular approach, allowing multi-communication technologies, such as, General Packet Radio Service (GPRS), Wi-Fi and Ethernet. The EVCS was verified both in the functional, as well as in the electrical domain. The prototype implementation of the EVCS is already fully operational and integrated in an energy operator EVCS network.

Keywords: Electric Vehicle, EVCS, Charging Infrastructure, Prototype.

1 Introduction

The electrification of the automobile has been progressing over the last 10 years. With fuel price escalading and an increased consumer environmental awareness, the old combustion engine will need to be replaced by a more environmental friendly and economic solution. Recent plug-in vehicles require a charging infrastructure, whether at private or public locations. Taking into account that the cost for a single Electric Vehicle Charging Station (EVCS) unit, capable of charging two vehicles, is more than 4500 euros, the costs associated to such a large-scale infrastructure are very significant.

The global architecture comprises the charging structure, the communication medium, and the Charging Infrastructure Management System (CIMS). The latter only represents a small part of the budget. This means that cost reductions can only be obtained from the first two elements.

Recent market analysis estimates the number of EVCS available worldwide to be 4.7 million [1]. This includes solutions from specialized vendors as well as especially established consortia, which deploy an entire EVCS network. Several pilot networks have already been deployed [2], [3], [4].

This paper presents a complete charging network architecture, with the main focus on the development of the EVCS unit [5]. Special attention was given to the communication interface, providing a modular multi-technology platform based on GPRS, Wi-Fi or Ethernet. The solution also integrates a parking meter adding value to the final solution.

K. Pentikousis et al. (Eds.): MONAMI 2011, LNICST 97, pp. 405–415, 2012.

The remainder of this paper is organized as follows. Section 2 presents the target EV charging network requirements and architecture. Also the communication protocol used between a central server and the EVCSs is described. Section 3 presents the EVCS prototype developed in this work. Section 4 includes functional and performance results of the developed EVCS unit. Finally, Section 5 presents the conclusions.

2 EV Charging Network Architecture and Protocol

The main contribution of this work is the development of a smart EVCS unit, but before describing the unit in detail, the system requirements and a global overview of the charging infrastructure is introduced.

A set of specifications for the EVCS resulted from the defined requirements:

- Authentication with a Smart Card is used to permit any EV to charge in a public grid.
- Communication is initiated by the EVCS.
- Access to the charging points is remotely controlled.
- Control the allowed users charging process, supplying the power safely meeting the country electrical regulatory standards.
- Provide accurate metering of the energy consumption, such that the costumer can be billed accordingly.
- Assure safe operation, monitoring hazardous events to the public during charge and idle periods.
- Two charging points (outlets) per EVCS.
- Resistance to vandalism: Feedback to the user is provided optically (Light Emitting Diodes) and audibly (piezo buzzer), and security breaches are detected with a door sensor.
- Minimal maintenance.
- Resistance to natural elements, such as heat and humidity.
- The EVCS control should be executed by an 8 bit AVR micro controller and be based on an Arduino.
- The input voltage is 12 VDC, as the EVCS will only have a single voltage converter and the relays chosen for controlling the supply current to the EV and the outlet locking mechanism operate at 12V.
- The EVCS must be able to work temporarily without communication with the CIMS.

In the charging network architecture, the EVCS is an endpoint, providing the required interface with the user and the energy operator. A broader view of the system must also include an IP based communication network and the CIMS. In Fig. 1 are shown the various network elements, like the EV, the EVCS, CIMS and the IP network. Alternative deployment architectures can be implemented, like isolated EVCSs, or EVCS sub networks, where one EVCS plays the network coordinator role, forwarding the messages from and to the CIMS.

The EV charging infrastructure will provide more functionalities than that of the supply of energy to the vehicle, such as control and billing. For these purposes it requires a node, which provides communication with all charging equipment, exchanging information and controlling the entire grid of EVCSs automatically and giving human operators a single interface to manage the resources. Also interaction with a business framework will be required for billing purposes. This node is the CIMS.

The technologies employed for communication must provide low installation and utilization costs and assure data security, while providing "near real-time" operations. With an average size of 400 bytes per message, high baud rates are not required for this purpose.

GPRS can benefit the solution with low installation costs as the GSM network is widely available. Nevertheless, operational costs can be high due to the associated service rates.

Assuming the ownership of the communications network, the presence of Ethernet or Wi-Fi can reduce operational costs. However, this is a big assumption, which adds cost due to the required cabling or access points (APs) and cannot be neglected.

A star network topology using a LAN (wired or not) to form an EVCS sub network, and a Coordinator or Gateway equipped with a GPRS Modem to provide CIMS interaction, can be a good compromise between cost and modularity, balancing the operational and deployment costs.

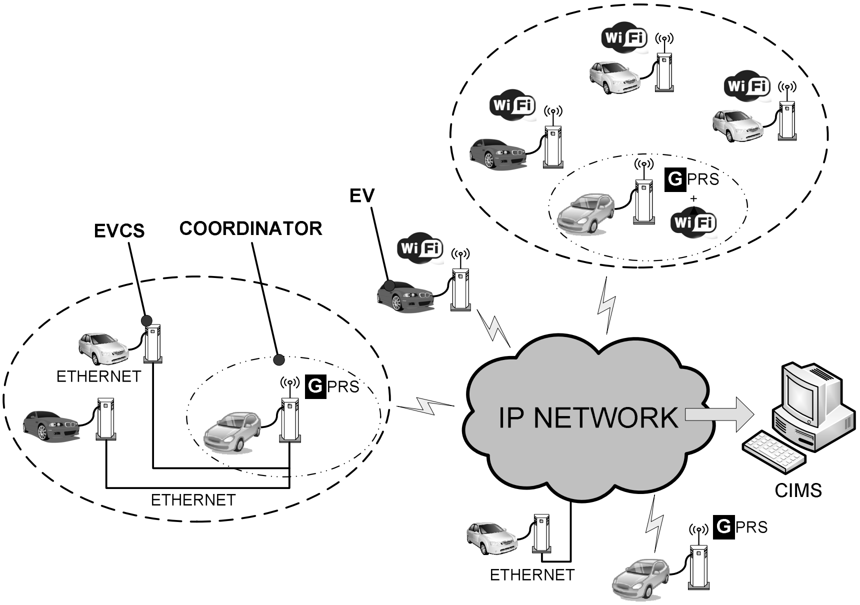

Fig. 1. EVCS network architecture

Other technologies like UMTS, PLC, Bluetooth or ZigBee can also be used. UMTS and PLC represent higher hardware costs, while Bluetooth and ZigBee limited range and cause interferences in the 2.4-GHz industrial, scientific and medical (ISM) band [6] [7]. Despite these disadvantages, some foresee a role for them in future Vehicle to Grid (V2G) implementations [8] [9].

The communication protocol must be carefully designed, as high processing capabilities are not available. A first view over some commonly used web services, like Simple Object Access Protocol (SOAP) or Representational State Transfer (REST) show their Extensible Markup Language (XML) parsing dependency, raising issues in this matter.

The protocol designed for EVCS to CIMS communication reduces this dependency. It is web-based and the HTTP/1.1 defined GET method is used for interaction. A GET request is issued to the server, encapsulating the data in the URL.

3 EV Charging Station Prototype

The EVCS design follows a modular approach allowing the addition of features or modules in the EVCS. This approach also simplified the replacement of a defective component by replacing the faulty module.

3.1 Hardware

The EVCS internal architecture is shown in Fig. 2. The Door Opening Sensor (DOS), on top, provides unauthorized access detection. Near it, the GSM antenna and the Smart Card reader, provide access to the GPRS network and RFID communication respectively. The two outlets, distributed on both sides of the structure, are responsible for energy delivery to the EV. A 12V battery is also present, serving as an alternative power supply in case of main power loss. On the left side of the battery is the controller, connected with a ribbon cable to a Printed Circuit Board (PCB), named "Connectors Board", providing interface to all the external peripherals, like energy meters, relays, locking mechanisms, AC detectors, buzzer, sensors, LEDs and Smart Card reader.

By including two outlets, two EVs can be charged at the same time, duplicating the energy module and the LEDs for optical feedback. This slightly increases the final cost for a unitary solution, but reduces the number of deployed units by 50%.

As shown in Fig. 3, the architecture is divided in three main interfaces. The User Interface where are included all the peripherals responsible for end user interaction, the Communications Interface, providing the required network access and the Energy Interface for power deliverance control, metering and safety assurance.

Since no specific type of contactless card was defined in the requirements, various alternatives were presented [10]. Three standards are defined for this type of cards, ISO/IEC 10536 for close coupled cards (CICC), with a maximum read distance of 2 mm, ISO/IEC 14443 for proximity cards (PICC), allowing a maximum read distance of 10 cm and data rates of 106 to 848 kbit/s, and ISO/IEC 15693 for vicinity cards (VICC) with a maximum read distance of 1 m and bit rate of 26.6 kbit/s. The CICC standard was immediately neglected, due the low bit rate, increased cost and low market penetration level. The VICC standard offers an increased read range, such that the transmitter power needs to be reduced. However, the VICC standard was disregarded, because of the higher cost of a VICC reader and antenna, compared to the PICC solution, and the lower bit rate. In this work, and following the market penetration statistics, ISO/IEC 14443 Type A Mifare Smart Cards are used as they have 75% market share [11]. During the final phases of development of the EVCS prototype, security problems with the card were reported [12] [13] [14]. At the moment the card reader is being replaced by a solution based on the ISO/IEC 14443 type B Calypso implementation.

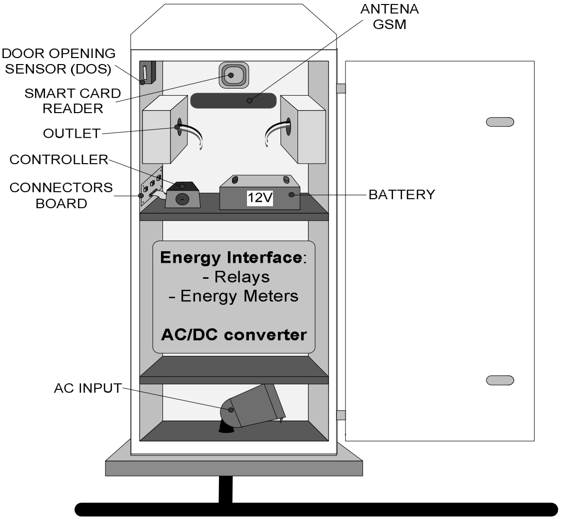

Fig. 2. EVCS internal structure

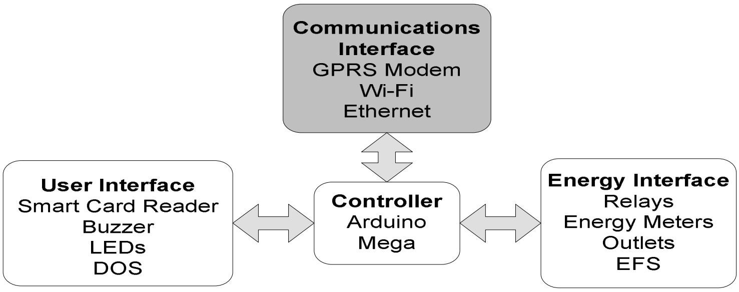

Fig. 3. EVCS architecture interfaces controlled by an 8 bit microcontroller

The metering and energy control is based on commercial off the shelf components, avoiding the long and expensive certification and approval procedures. There was also the need to implement AC to DC logic detection, providing the Relay status feedback and also the EFS.

Current GPRS security is projected to protect only the radio access network and wireless path. The backbone and wire-line connections are not covered by any particular mechanism. This leads to concerns when transmitting data trough the link established to the Internet or the company LAN, due its clear-text format [15], requiring additional security measures. To solve this problem, an Accelerated Private Network (APN) was created by the GSM operator, providing IP Security Protocol (IPSec) tunnels between the GPRS backbone and the CIMS, like shown in Fig. 4. Using fixed IP address for the SIM cards, an isolated private network was formed, managed by the mobile operator, for added safety and functionality.

Additionally to the GPRS interface, Ethernet and Wi-Fi interfaces were also implemented. The working scheme permits for simultaneous operationality of one of the latter technologies with GPRS.

Ethernet and/or Wi-Fi suffer from the same security issues as GPRS without the APN. Security was assured by encrypting the communications, using the Transport Layer Security (TLS) protocol [16], providing safe transactions between the EVCS and the CIMS.

Existing TLS implementations on 8 bit processors were analyzed and it was concluded that the limiting factor was not the embebbed TLS implementation footprint, where previous solutions [17] pointed to a 50 KB target, but the 20 KB SRAM space required, 150% more than the actual Arduino Mega size. This pointed the need of a hardware based implementation, which was accomplished with the use of one of two device servers, Nano Socket iWiFi™ [18] and Nano SocketLAN™ [19].

3.2 3.2 Software

The EVCS architecture developed considers N + 1 finite state machines (FSM), running sequentially, in an infinite loop, while timing and energy metering functions work asynchronously and where N is the number of Outlets present in the EVCS, as illustrated in Fig. 4. The Smart Card Reader is also modeled with a FSM, providing detection, authentication and reading procedures.

Transitions between these states are handled by messages received from the EVCS Main Loop FSM.

The EVCS Main Loop FSM is responsible for initializing from a cold start process or alternatively a warm start, provided that a backup is stored in the controller EEPROM. The initialization process can be requested in cold or warm scenarios. Both the scenarios lead to a first initialization action, where the EEPROM is verified, in search for a backup. If a backup is found, checksum verification is done, which, resulting in success, will lead to a recovery of the last saved state.

The alarm situations detection was implemented in a synchronous scheme inside the EVCS FSM. This decision was taken after verifying that the system processing timings were short enough to provide an efficient and on time alarm detection.

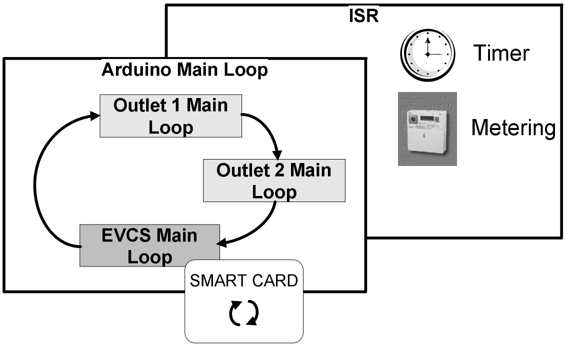

Fig. 4. Software architecture representation, with three FSMs

3.3 CIMS Emulation

In parallel with the EVCS implementation, the support base, required for testing and validation purposes was also developed. This system represents only a subset of the CIMS total functionality, since the business framework interaction is not considered.

The CIMS emulator is based on PHP. It consists of a textual flat database file, accessed by a back office engine, while the information is presented to the user graphically, by means of tables, accessed by differentiated web pages.

4 Evaluation of EVCS

A functional and performance analysis of the system was executed allowing validating the solution, verifying the electrical correctness and benchmarking its performance, providing a complete evaluation of the final result.

4.1 Hardware Functionality

Functional analysis, included measurements of electrical characteristics, like power consumptions, voltage levels and pulse periods. Power consumption was measured for all the EVCS elements to confirm the AC-DC converter conformity to the system requirements.

Measurements were also taken from the Energy Interface. The current resulting in a Relay switch was verified. Similarly to other sensors, the EFS levels were measured.

Communications Interface measurements, included the peak power drained over a part of a GPRS communication period (6.25 W) and the modem consumption when idle (337 mW). The NanoSocket LAN consumption (1.24 W) was verified to be lower than the specified value (1.47 W), while the iWiFi module presented approximately the expected (2.24 W for a declared 2.14 W).

The GSM signal available inside the EVCS structure was also verified, following the Portuguese National Communication Authority (ANACOM) methodology [20], for radio electric network availability.

The GPRS Modem was used as an RF Scanner, assuring receiver characteristics are equal in testing and operational environments, minimizing errors. The results obtained in different days were always higher than a RSSI of -69 dBm, equivalent to a “Good” classification, the maximum obtainable in a scale of four levels. These results are only valid for the geographical area where the EVCS is located, such that other scenarios must be analyzed, following the same or similar methodologies.

4.2 Software Functionality

To provide a fail proof system, the software developed was subject to a test bench, following both “white box” and “black box” [21] approaches.

The “white box” approach allows verifying the system exterior functionalities by covering and testing the code that realizes them. The software developed was analyzed thoroughly, and code items like “If”, “Case”, “For” and “While” sentences and cycle and variable boundaries were marked. These items were forced to subject conditions, enabling testing out of bounds, and cycle break conditions covering a full path verification, aiming to ascertain the maximum code extent.

Alternatively, the “black box” methodology is based on program specifications and not on the internals of the code. In this case, the system functional specification, that is, a description of the expected behavior of it, is used as a source of information for test case specification. The system was subject to the test suite with positive results.

4.3 Performance Evaluation

The communications timings were measured, allowing benchmarking the network performance regarding the various technologies. Timing and memory consumption were also analyzed to verify response times under current system load, but also to investigate if the chosen processor has sufficient resource available for extensions.

Efforts to reduce the SRAM consumption resulted in moving parts of the SRAM to Flash (program) memory, as shown in Table 1. The full EVCS system footprint could also be minimized, representing approximately 45 % of the controller capacity.

Table 1. Memory Occupancy

Memory Type	Total Available [Kbyte]	Total Occupied [Kbyte]	Occupation [%]
Flash	124	54.4	44
SRAM	8	4.0	50
EEPROM	4	0.8	20

In order to benchmark the Communications interface, timing measurements were performed, which results are shown in Table 2. The test consisted of measuring a send and receive cycle, from the instant the AT command start to be dispatched until the last response byte is received. A first noticeable result is the Wi-Fi interface maximum and minimum timing differences, explained by the weak RF signal and also the NanoSocket iWiFi signal caption, which proved to be irregular. The GPRS worst case communication represents less than 6 seconds, although this measure is location based, due to the signal dependence. As expected, Wi-Fi results are slightly better and Ethernet has a far superior performance.

Observing the Ethernet (10 Mbps) interface results, it is clear that the majority of the delay observed is associated to the controller and CIMS processing, which nevertheless represent less than 330 ms. The GPRS and Wi-Fi results show not only the slower protocols and lower transfer rates available (average timings), but also the signal quality dependence, resulting in transmission errors and/or dynamic rate scaling.

These last results however, may be considered to comply with the system requirement of “near real time” operation.

Table 2. Network Operational Timings

Communication Technology	Timing Measured (40 measures per case)		
	Average [ms]	Maximum [ms]	Standard Deviation [ms]
GPRS	5339	6098	221
Wi-Fi without TLS	1844	5107	1789
Wi-Fi with TLS	2294	4031	1456
Ethernet without TLS	320	326	7
Ethernet with TLS	508	666	77

In order to assess the global system performance, various EVCS processes where measured, such the time the system takes to read a card and start user interaction, the time to process an incoming message, the lag between an erroneous event and the following CIMS notification arrival and the time the EVCS is occupied processing a user request and are shown in Table 3.

Table 3. Timings for Various EVCS Operations

Operation	Timing Measured		
	Average [ms]	Maximum [ms]	Standard Deviation [ms]
Smart Card read	500	501	1
Message Processing	2	3	0
Alarm Detection when Idle	20	20	0
Alarm Report when Idle	156	156	1
Alarm Detection	788	788	0
Alarm Report	5469	5759	416

The Smart Card detection, selection, authentication and reading are completed in 500 ms or less. The alarm event detection was analyzed in various system states. First measures were done with the system idle, which shows an average value of 20 ms, while the time for the ALARM message to leave the EVCS is less than 200 ms. The worst-case scenario is when the alarm event occurs immediately after the Smart Card detection and the communication is GPRS based. The EVCS FSM and Outlets FSM loop is processed before an alarm is detected, resulting in detection in less than 800 ms. The maximum time that a report takes to be sent is affected by the previous CIMS pending interaction and is approximately 5 s.

5 Conclusions

The main objective of this work was to develop a modular EVCS to be integrated in an emerging charging network. A full hardware solution was built and is successfully integrated in an EVCS network. It integrates commercial off the shelf energy metering and controlling devices, like meters and relays, and provides a multi-technology communication platform.

The EVCS unit can charge 2 EVs at the same time and has accurate energy and time metering functions. Communication with a central server is established by GPRS, Wi-Fi or Ethernet. However, the modular approach makes it trivial to add other communication mechanisms.

The unit provides the end-user with audible and optical feedback only, as a result of the requirement to withstand vandalism. A door-sensor and power failure detection sensors further improve the end-user security.

The EVCS currently uses the Mifare Classic RFID cards (ISO/IEC 14443 Type A) as authentication. Since this technology was recently proven to be insecure, these units are now being replaced by Calypso (ISO/IEC 14443 Type B) cards.

Rigorous timing and memory consumption analysis was performed on the EVCS, such that minimum and maximum response times to events and free space in memory for future additions are known. The largest response time is the actual transmission and reception of a communication with the CIMS when using GPRS, which showed to be fewer than 6s. However, it was shown that using Wi-Fi this time reduces to under 4s and under 500 ms for Ethernet.

The firmware of the AVR processor was implemented using FSM for the main processes and interrupt service routines for the timing and energy metering. The correct functioning of the software was evaluated using both white-box and black-box approaches.

The solution was tested and delivered as a prototype to the energy operator and is already operational for a couple of months, with positive results. Future integration of the solution in the EVCS network is underway.

References

[1] Gartner, J., Wheelock, C.: Electric Vehicle Charging Equipment Charging Stations, Grid Interconnection Issues, EV Charging Business Models, and Vehicle-to-Grid Technology: Market Analysis and Forecasts, Research Report, Pike Research, Boulder, USA (2010)

[2] Vidigal, A.: Mobilidade Eléctrica. In: XVIII Congresso da Ordem dos Engenheiros, Aveiro, Portugal (October 2010)

[3] Kneeshaw, S.: Electric Vehicles in Urban Europe Baseline Report, Technical Report, URBACT, Saint-Denis La Plaine, France (May 2010)

[4] Reis, L.: Modelo e Sistema de Carregamento para Veículos Eléctricos em Portugal. In: 2010 Portuguese IMTT seminar Mobilidade Eléctrica: O Veículo, Lisbon, Portugal (March 2010)

[5] Pinheiro, T.: Electric Vehicle Charging Station, MSc. Thesis, Instituto Superior Técnico – Universidade Técnica de Lisboa, Lisboa, Portugal (April 2011)

[6] Hager, C.T., Midkiff, S.F.: An analysis of Bluetooth security vulnerabilities. In: Wireless Communications and Networking, New Orleans, LA, USA (March 2003)

[7] Jennic: Co-existence of IEEE 802.15.4 at 2.4 GHz, Application Note, Jennic, Sheffield, UK (February 2008)

[8] Zpryme Research & Consulting, LLC, V2G, Smart Grid Insights, Austin, USA (July 2010)

[9] Ritter, B.: The ZigBee Alliance - Close-up: Rapid ZigBee Adoption by Utilities. In: Wireless Congress Systems and Applications, Munich, Germany (October 2009)

[10] International Telecommunication Union, Ubiquitous Network Societies: The Case of Radio Frequency Identification. In: ITU Workshop on Ubiquitous Network Societies, Geneve, Switzerland (April 2005)

[11] Heikki, H.: Expanding the Global Market for NFC, NXP Market Update, NXP, Eindhoven, Netherlands (April 2008), `http://www.wima-nfc.com/pics/Image/Huomo.pdf`

[12] Nohl, K., Evans, D., Starbug, Plötz, H.: Reverse-engineering a crypto-graphic RFID tag. In: 17th USENIX Security Symposium, San Jose, USA (July 2008)

[13] Garcia, F.D., de Koning Gans, G., Muijrers, R., van Rossum, P., Verdult, R., Schreur, R.W., Jacobs, B.: Dismantling MIFARE Classic. In: Jajodia, S., Lopez, J. (eds.) ESORICS 2008. LNCS, vol. 5283, pp. 97–114. Springer, Heidelberg (2008)

[14] de Koning Gans, G., Hoepman, J.-H., Garcia, F.D.: A Practical Attack on the MIFARE Classic. In: Grimaud, G., Standaert, F.-X. (eds.) CARDIS 2008. LNCS, vol. 5189, pp. 267–282. Springer, Heidelberg (2008)

[15] Xenakis, C.: Security Measures and Weaknesses of the GPRS Security Architecture. International Journal of Network Security 6(2), 158–169 (2008)

[16] Dierks, T., Rescorla, E.: The Transport Layer Security (TLS) Protocol, Standards Track, IETF, Fremont, USA (August 2008)

[17] Stapko, T.: Embedded Systems Programming. Miller Freeman, San Francisco (2004)

[18] Connect One, Nano Socket iWiFi^{TM} data sheet, ver. 1.35, Datasheet, Connect One, San Jose, USA (September 2009)

[19] Connect One, Nano SocketLAN^{TM} data sheet, ver. 1.20, Datasheet, Connect One, San Jose, USA (July 2009)

[20] ANACOM, Avaliação da QoS dos Serviços de Voz, Videotelefonia e Cobertura das Redes GSM e WCDMA, nos Principais Aglomerados Urbanos e Eixos Rodoviários de Portugal Continental, Technical Report, ANACOM, Lisbon, Portugal (December 2010)

[21] Desikan, S., Ramesh, G.: Software Testing Principles and Practices, 6th edn. Pearson Education, New Jersey (2008)