Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2017) 25: 1554 – 1562

doi:10.3906/elk-1511-146

Communications protocol for power management in smart homes

Milan BJELICA*, **Mirjana SIMIĆ PEJOVIĆ**
Department of Communications, Faculty of Electrical Engineering, University of Belgrade, Belgrade, Serbia

Received: 17.11.2015 • **Accepted/Published Online:** 30.05.2016 • **Final Version:** 10.04.2017

Abstract: Emergency situations, incentives for energy efficiency, new pricing plans, and distributed electricity market jointly require home automation systems that would conform an instantaneous load to a dynamically set limit. In this paper, a novel machine-to-machine communications protocol that interconnects a smart grid and smart house is described. "Intelligent" domestic appliances use it to coordinate their switch-on times so that the assigned power quota will not be exceeded. In this way, the end-user experience can be improved by reducing the electricity bills, conveniently conforming to prepaid pricing plans, or even by providing continued service at possibly reduced consumption levels during power shortages.

Key words: Communication protocols, network optimization, machine-to-machine communications, smart grid, smart home

1. Introduction

During the past decade, the paradigms of smart grid and smart buildings/homes have drawn much attention within both the research community and standardization bodies. As far as communication protocols are considered, this attention has been mostly focused on smart meter remote reading, appliance remote controlling, or power-line communications, while few attempts have been made to integrate these technologies into a seamless system. An illustrative example of this are blackouts, as a still common way of handling an imbalance between electric power generation and demand. They are unfair by nature, as they affect all consumers that are connected to the same distribution line regardless of their conformance with the available amount of power. Let us now consider a different scenario, in which a distribution network operator communicates the available power quota to each house that it serves. If loads (i.e. appliances) within these houses could coordinate their switch-on times or power consumption levels, it would be possible to avoid blackouts by conforming the total load to the assigned quota, and thus to provide continuous service. This service would indeed be of a somewhat reduced level, but still better than none.

In this paper, a novel machine-to-machine communication protocol that interconnects smart appliances and allows their integration into a smart grid is described. The protocol enables a master controller to communicate with appliances and thus to coordinate their switch-on times so that the desired power load profile can be met. It is worth noting that this load shaping should not be motivated solely by reasons of emergency; indeed, users in a modern, decentralized power market may want to reduce their bills by controlling the appliances remotely, e.g., switching heating/air conditioning on when they finish work and head home, or by shifting the peak load to a part of the day when electricity is cheaper. Moreover, it is not difficult to conceive of pricing plans where users will be charged not only for the total energy but also for the peak power they draw.

*Correspondence: milan@etf.rs

2. Background and related work

As already stated, the primary objective of our research is to develop a communications protocol that will provide continuous service (i.e. power supply) to high priority loads (e.g., freezers, lighting, communication devices) even during power shortages. A literature survey on home automation reveals that the topic of appliance management during emergency situations has received virtually no research attention. Indeed, the majority of popular commercial off-the-shelf systems, like X10 [1], Insteon [2], and Universal Power Bus [3], are aimed at nothing more than remote control of lighting or HVAC and rely upon communication protocols that are optimized for these applications [4,5]. As their action is triggered solely by the user, they could not be easily modified to interconnect to the network center or to another building and proactively respond to the announced power shortage. The obstacles the existing systems face are identified as high cost, inflexibility, poor manageability, and insufficient security [6].

Customers in smart grids are expected to be given automated incentives to offload nonessential appliances in order to reduce both system faults and operating costs [7]. Knowing this, our objective is to develop a simple and flexible protocol for communication between a controller and connected appliances within a building, which could then be easily interfaced to a smart grid. During the periods of regular operation, the system could be used to control the devices remotely, or to shape the power load in accordance with the pricing plan.

The demand for simplicity implies low cost of both the equipment and the overall system deployment. While some research efforts rely upon higher-level protocol mechanisms [8,9], our proposal operates on the data link layer (DLL); to make it as simple as possible, we further discard certain DLL functionalities that we find unnecessary for the intended use, but which are foreseen in similar proposals, e.g., segmentation and reassembly or ARQ as in [10]. For the same reason, we decided not to use some well-established (e.g., Ethernet) and upcoming technologies (e.g., Internet of Things, as in [11]), but rather to develop a specialized protocol from scratch.

The demand for flexibility implies that different transmission media can be used to interconnect physical system components. As a rule, many research papers propose solutions that are designed for a particular physical layer, which makes the interworking of these systems virtually impossible; wireless technologies, like GSM [12], Bluetooth [13], or ZigBee [14], are good examples of this. As full wireless coverage of certain buildings might be a demanding task [5], especially when, as in most of Europe, reinforced concrete is used as building material, our proposal implies the use of power line communications (PLCs), i.e. existing power lines as a transmission medium; it is, however, worth noting that the protocol per se is independent of the physical layer (i.e. wireline/wireless), channel (attenuation, noise power density), and signal properties (modulation scheme, signal level, bit rate, etc.), so different transmission technologies can be used in practical implementations. Another reason to opt for the PLC is that for the time being wireless networks are less reliable and more prone to interference and attacks, which raises security problems previously unknown to power grids [15]. The question of network security in smart homes still needs to be considered.

3. Description

The simplified system topology is shown in Figure 1. In the assumed three-phase system, each line (L1, L2, and L3) acts as a communication bus. Coexistence of different home networks that are connected on the same power line is achieved by band reject filters within a metering device, as these prevent communication signals from leaving their networks and entering another one.

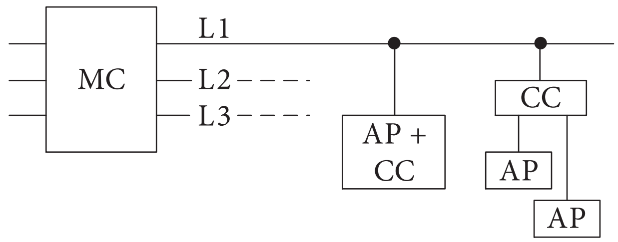

Figure 1. Network topology.

The user (or grid operator) assigns to the master controller (MC) the maximal amount of power that can be drawn at a given instance of time. This amount is referred to as the quota. The master controller is incorporated into a smart meter; it ensures that the power quota is never exceeded. To accomplish this, each appliance (AP) asks for permission to power on; this is done through dedicated communication controllers (CCs). While some larger appliances might have intrinsic communication controllers, it is also possible that some smaller loads will communicate with the master controller through a shared communications controller, which could be incorporated into an extension cord. To be fully operational within a system, each communication controller must be registered with its master controller.

Different priorities can be assigned to appliances. The master controller decides on the power-on request by not only judging the power balance, but also by considering the appliance priority, its usage pattern, or other operator/user-defined criteria. Only those appliances that are granted permissions through their CCs can be powered on, while the others must wait until some devices are switched off or the quota is increased. If the quota is decreased, the master controller can withdraw the permissions issued to some devices, thus causing them to power off.

The CCs can communicate only with the master controller and not with each other. The master controller uses the unique address of each CC to label a recipient of the downlink data. On the uplink, however, there are many possible senders that are unaware of each other, so collisions may occur.

Protocol state diagrams for master and communications controllers are shown in Figures 2 and 3, respectively. State transitions are denoted as condition/action. The conditions that are generated outside the communication subsystem are written in italics. For the reason of simplicity, irregular situations are omitted.

The master controller normally resides in the IDLE state. Should it receive a power-on request from a CC (POW_ASK), it would consider it and reply with POW_GRANT. In the event of power overload, the MC will issue the POW_OFF message, ordering some appliances to power off.

The MC can open a registration procedure, through which newly connected or previously offline appliances register. Registration is triggered by an algorithm intrinsic to the master (*internal* in Figure 2). The MC announces registration start with the REG_START message and then goes to the REG state. Timer T1 determines the duration of the registration. While in the REG state, the MC replies to each REG_REQ message with REG_GRANT.

As shown in Figure 3, the communications controller can be in one of the states IDLE, POW, or REG. The CC is normally in the IDLE state, no matter if the corresponding appliance is powered on or off. Should the controller receive the POW_OFF message addressed to it, it would switch the appliance off the mains.

If the CC was not registered to the master, it waits for the REG_START message, starts timers T1 and T2, and enters the REG state. The T1 timer once again determines the registration duration, while T2 counts random delay, needed to minimize the number of collisions on a shared uplink. After this random delay, the REG_ASK message is sent to the master. Upon the receipt of a confirmation (REG_GRANT) or after T1 has

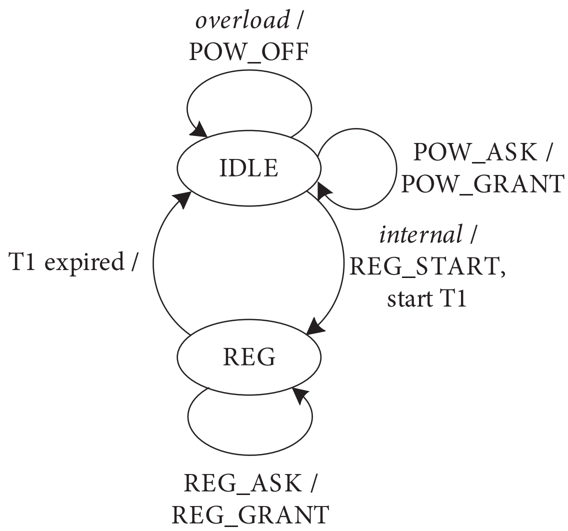

Figure 2. State diagram for master controller.

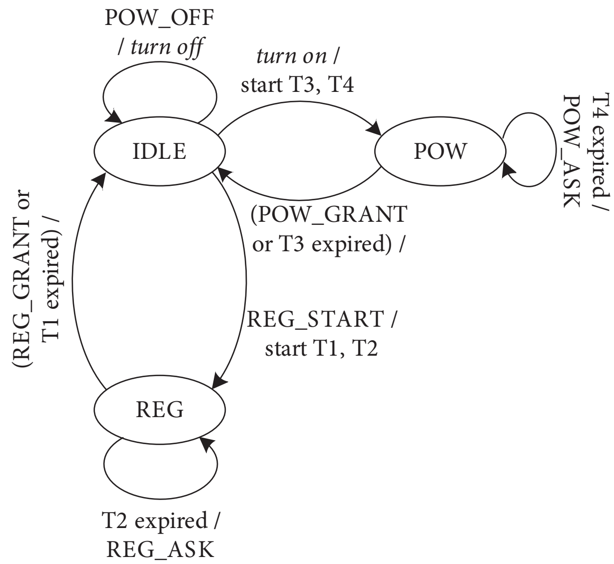

Figure 3. State diagram for communications controller.

expired, the CC returns to the IDLE state; in the former case, the registration was successful, while in the latter case the CC remains unregistered and must wait for another registration round.

When an appliance needs to be powered on, or its current power increased, its controller starts timers T3 and T4 and proceeds to the POW state. The T4 timer determines the random delay after which a request to power on (POW_ASK) will be sent to the master; the random delay is once again needed to avoid collisions. The answer is awaited for the amount of time defined by T3. Should the CC receive an answer (POW_GRANT) before T3 has elapsed, it will return to the IDLE state and remain in it until the appliance wants to increase its power. Should no answer be received before T3, the CC would conclude that a collision has occurred; should the user still want to power the appliance on, the CC would again go to the POW state and the described procedure would be repeated.

The MC is not explicitly notified when an appliance wants to decrease its power; it learns this by monitoring the instantaneous load.

As long as the registration is in progress (e.g., the T1 timer has not expired), the registered CCs will restrain from issuing POW_ASK messages.

Communication messages are sent through the channel as packets (frames), whose format is shown in Figure 4.

Each packet starts and ends with a flag, a 1-byte sequence of 0111111110. The flag frames the packet content and helps establish synchronization on the receiver side. To ensure transparent transmission, the flag sequence should not appear within the packet content. This is accomplished by bit stuffing technique: should a sequence of a zero and five adjacent ones (i.e. 011111) appear in the packet body on the sender side, a zero would be inserted after the fifth 1 so that the sequence 0111110 is actually transmitted. The inserted zeros are systematically removed on the receiver side.

Flag	ADR	CF	PL	FCS	Flag

Figure 4. Packet format.

The ADR field contains the local address of the sending/receiving CC, which is assigned to it after it has successfully registered. The local addresses are used for security reasons, i.e. to prevent malicious appliances from denying a service to the legitimate ones. The address "all ones" is reserved for broadcast purposes, so that the master could address all the CCs when announcing the registration (REG_START) or when ordering an emergency power-off (POW_OFF). As for the Internet protocol, the local addresses may be hierarchically organized so that the appliances with the same priorities are grouped into common "subnets". To preserve the emergency supply of high priority appliances, the master could order that the lower priority subnets power off.

The CF is the control field, which codes different messages that are used in the protocol.

The PL is the payload field. It contains additional information specific to the particular message (i.e. the CF field). When asking to be registered, the CC uses a reserved ADR "all zeros" and sends its factory-set identifier as the payload. This identifier is equivalent to a hardware-set MAC address and, as for the universal serial bus, may consist of the vendor ID, product ID, and serial number. The master will reply with REG_GRANT, with ADR equal to the assigned local address and PL equal to the factory-set identifier. For POW_ASK, the PL contains the binary-coded amount of power that is requested. In POW_GRANT, the PL shows if this request was granted (all ones) or denied (all zeros).

FCS is the packet (frame) check sequence, generated by the cyclic redundancy check (CRC) of the ADR, CF, and PL fields:

$$\text{FCS} = \text{Res}\ \left\{ \frac{D(x) \cdot x^n}{G(x)} \right\}. \tag{1}$$

Here, $D(x)$ is a GF(2) polynomial corresponding to the contents of the concatenated ADR, CF, and PL fields; $G(x)$ is a CRC generating polynomial of degree n; and Res$\{\bullet\}$ is the remainder after modulo-2 division. It is worth noting that the proposed protocol is independent on a particular error detection scheme. The CRC is chosen as it offers a good trade-off between computational complexity and performances.

Let us note that reliable delivery of the communication messages is achieved by the retransmission scheme implicitly implemented through FCS, timers T1 and T3, and confirmation messages REG_GRANT and POW_GRANT, respectively; as will be shown in the next section, the use of this DLL functionality presents a good trade-off between system complexity and performances.

4. Performance evaluation

Functional hardware prototypes of the master and communication controllers (Figure 5) were implemented to verify both the plausibility and logical consistency of the proposed protocol. The prototype is based on two general-purpose microcontrollers, both with 1 MHz clock and 512/256 kB (kilobyte) RAM. Its parameters were then fine-tuned by the aid of computer simulation, which is described in the remainder of this section.

As the uplink is a shared medium, messages REG_ASK and/or POW_ASK that originate from different CCs might collide; in Figure 6, this is illustrated for the REG_ASK messages. Without loss of generality, only the T1 and T2 timers will be considered in the following discussion; a similar analysis can be applied to optimization of T3 and T4.

The registration phase duration is determined by timer T1, which is common to both the master and the CCs. Upon the beginning of registration, each unregistered CC waits for some random time determined by T2. The longer this wait time is, the fewer REG_ASK messages will collide, but the master will have to wait more for the registration requests to arrive. Until the registration ends, the requests to power on cannot be processed. It is therefore important to set the system parameters in such a way that the CCs register in the

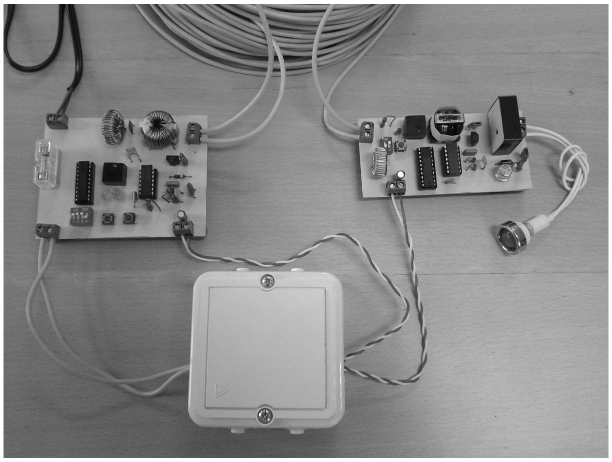

Figure 5. Prototype assembly: master controller (left), communications controller (right), AC/DC power supply (bottom).

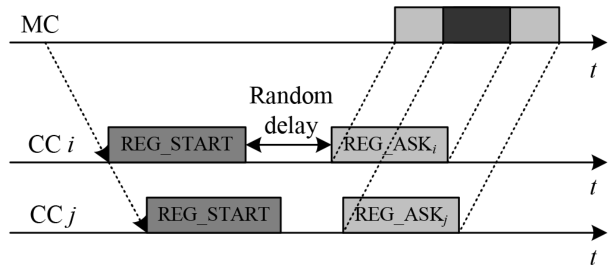

Figure 6. Two REG_ASK messages colliding at the MC.

shortest time possible. To determine the appropriate timer values, an approach similar to the optimization of Ethernet passive optical networks [16] can be applied.

Let T_1 denote the value assigned to T1 and T_2 denote the maximal wait time (random delay). Having in mind Figure 6, we can write

$$T_1 = 2t_p + T_2, \quad (2)$$

where t_p is the maximal one-way propagation time, corresponding to the CC that is farthest from the master. The total duration of the registration phase is now

$$T = T_{REG\_START} + T_1, \quad (3)$$

where T_{REG_START} is the REG_START message duration.

Now let the round-trip (two-way propagation) delays corresponding to the observed CCs i and j be t_i and t_j, and their random wait times w_i and w_j, respectively; it is obvious that $t_i,\ t_j \leq 2t_p$ and $w_i,\ w_j \leq T_2$. Let T_{REG_ASK} be the duration of a REG_ASK message. Two REG_ASK messages will then collide if (and only if) the difference of their arrival times at the master is less than or equal to the REG_ASK message duration:

$$|(t_i + w_i) - (t_j + w_j)| \leq T_{REG\_ASK}. \quad (4)$$

In the network of $N \geq 2$ CCs, this condition should apply to each pair of them.

A computer simulation was run to test the performances of the CC registration. The following packet structure was assumed: two flags, each of 1 B (byte) length; 1 B for the ADR; 1 B for the CF; 5 B for payload; and 1 B for the FCS, with the CRC-8 CCITT polynomial ($x^8 + x^2 + x + 1$). This yields a message total length of 10 B. Simulation parameter values are listed in Table 1. They were chosen to correspond to a residential house scenario [5,17] and in accordance to the novel IEEE 1901 PLC standard [18]. Signal-to-noise ratio and bit error rate values were adopted from simulation results reported by Jing et al. [19]. The simulation was run in Python 2.7. Independent Monte Carlo experiments included generation of the REG_ASK messages by the CCs and their reception by the MC. The CC propagation delays and wait times were drawn from uniform distribution, the former from the interval $[0,\ 2t_p]$ and the latter from $[0,\ T_2]$. The trials were repeated until either the probability of successful CC registration was estimated with 90% confidence with relative error not greater than 2%, or a maximum number of 10^6 runs was reached. The obtained results are shown in Figure 7, and an excerpt from them of interest to the discussion to follow is given in Table 2.

Table 1. Simulation parameters.

Parameter	Value
Number of CCs	50
Cable span	50 m
Propagation speed	2×10^5 km/s
Frequency band	CENELEC-A
Channel throughput	100 kb/s
Signal-to-noise ratio	8 dB
Bit error rate	10^{-4}

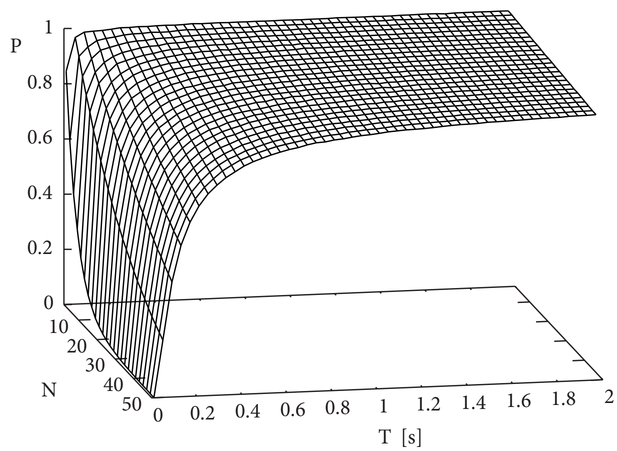

Figure 7. Probability of successful registration (P) vs. number of competing CCs (N) and registration phase duration (T).

Table 2. Some registration probabilities.

N	T	P
50	1.9 s	0.96
50	100 ms	0.46
27	100 ms	0.71
8	100 ms	0.90
1	100 ms	1.00

In the network of 50 CCs per phase, it would take 1.9 s until on average 48 CCs are registered (success probability: 0.96). Should the user need to wait this amount of time before getting a response from an appliance, it would most probably be frustrating to her/him. However, the shape of the surface in Figure 7 suggests that it is possible to achieve greater efficiency within a shorter period of time. As an illustrative example, let the MC announce not one long but four shorter successive registration rounds of 100 ms each. As Table 2 shows, in this way all the CCs would register in on average 400 ms: 23 (out of 50) will register in the first round, 19 (out of the remaining 27) in the second, 7 (out of 8) in the third, and the last remaining in the fourth round.

This outperforms both theoretical (10–220 ms for two nodes [5]) and experimental results (600 ms [14]) for the latencies in home automation networks, which are reported in the available literature and which relate to residential usage scenarios.

5. Conclusion

A novel machine-to-machine communication protocol has been developed. This protocol enables smart domestic appliances to negotiate their power-on times so that the assigned power quota is not exceeded; in this way, the users could still be provided with a certain amount of power supply even in the event of system disturbance. The operation of both the master and the appliance controllers has been described through finite state machine models. Communication messages have been defined and their format explained. A functional hardware prototype was implemented and computer simulation was used to set the timer values and to optimize the system performance.

Future work on this topic might include consideration of power prediction protocols for the master side, serving the appliances with different tolerances to latency, and network security.

Acknowledgments

The authors would like to thank Prof Predrag Pejović for his valuable support in conducting this research. This work was funded in part by the Serbian Ministry of Education and Science under Grant TR32028.

References

[1] BMB Electronics. X10 Product Sheets. Veen, the Netherlands: BMB Electronics B.V., 2008.

[2] SmartLabs. Insteon Developer's Guide. 2nd ed. Irvine, CA, USA: SmartLabs Technology, 2009.

[3] Powerline Control Systems. UPB Technology Description, Version 1.4. Northridge, CA, USA: PCS, 2005.

[4] Kailas A, Cecchi V, Mukherjee A. A survey of communications and networking technologies for energy management in buildings and home automation. J Comput Netw Commun 2012; 2012: 932181.

[5] Gomez C, Paradells J. Wireless home automation networks: a survey of architectures and technologies. IEEE Commun Mag 2010; 48: 92-101.

[6] Brush AJ, Lee B, Mahajan R, Agarwal S, Saroiu S, Dixon C. Home automation in the wild: challenges and opportunities. In: SIGCHI Conference on Human Factors in Computing Systems; 7–12 May 2011; Vancouver, Canada. New York, NY, USA: ACM. pp 2115-2124.

[7] Rajkumar R. A cyber-physical future. P IEEE 2012; 100: 1309-1312.

[8] Choi J, Shin D, Shin D. Research and implementation of the context-aware middleware for controlling home appliances. IEEE T Consum Electr 2005; 51: 301-306.

[9] Byun J, Hong I, Hwang Z, Park S. An intelligent cloud-based home energy management system based on machine to machine communications in future energy environments. In: The Eighth International Conference on Systems; 27 January–1 February 2013; Seville, Spain. Wilmington, DE, USA: IARIA. pp. 40-45.

[10] International Telecommunication Union Telecommunication Standardization Sector. Recommendation ITU-T G.9961: Unified High-Speed Wireline-Based Home Networking Transceivers - Data Link Layer Specification. Geneva, Switzerland: ITU, 2015.

[11] Institute of Electrical and Electronics Engineers. IEEE Std. 1888-2014 – IEEE Standard for Ubiquitous Green Community Control Network Protocol. New York, NY, USA: IEEE, 2014.

[12] Yuksekkaya B, Kayalar AA, Tosun MB, Ozcan MK, Alkar AZ. A GSM, internet and speech controlled wireless interactive home automation system. IEEE T Consum Electr 2006; 52: 837-843.

[13] Piyare R, Tazil M. Bluetooth based home automation system using cell phone. In: IEEE 15th International Symposium on Consumer Electronics; 14–17 June 2011; Singapore. New York, NY, USA: IEEE. pp. 192-195.

[14] Gill K, Yang SH, Yao F, Lu X. A ZigBee-based home automation system. IEEE T Consum Electr 2009; 55: 422-430.

[15] Chen PY, Cheng SM, Chen KC. Smart attacks in smart grid communication networks. IEEE Commun Mag 2012; 50: 24-29.

[16] Bjelica M, Peric A. Allocation of optimal discovery slots in IEEE 802.3av networks. AEU-Int J Electron C 2012; 66: 211-213.

[17] Anatory J, Theethayi N. Broadband Power-Line Communication Systems: Theory and Applications. Southampton, UK: WIT Press, 2010.

[18] Institute of Electrical and Electronics Engineers. IEEE Std. 1901.2-2013 – IEEE Standard for Low-Frequency (Less Than 500 kHz) Narrowband Power Line Communications for Smart Grid Applications. New York, NY, USA: IEEE, 2013.

[19] Jing L, Pande T, Han KI, Batra A, Evans BL. Time-frequency modulation diversity to combat periodic impulsive noise in narrowband powerline communications. IEEE T Commun 2015; 63: 1837-1849.