The following publication M. H. Tsoi, T. H. Ng, D. P. K. Lun, Y. S. Choy and S. W. Y. Mung, "Reconfigurable frequency switching and synchronization for noise avoidance in power line communication system," 2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia), Seoul, Korea (South), 2020, pp. 1-4 is available at https://doi.org/10.1109/ICCE-Asia49877.2020.9276974.

Reconfigurable frequency switching and synchronization for noise avoidance in power line communication system

Man Ho Tsoi[1 *], Tsz Hong Ng[2], Daniel P. K. Lun[3], Yat Sze Choy[4], Steve W. Y. Mung[5]

[1,4]*Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hong Kong*

[2]*Linked-Technologies Limited, Hong Kong*

[3,5]*Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong*

[*]*Email address: man-ho.tsoi@connect.polyu.hk*

Abstract

The power line communication is thriving due to its high compatibility with well-established power grids. However, the inevitable noises in the power lines decrease the reliability of the power line communication system. A reconfigurable frequency switching and synchronization is proposed in this paper to avoid the noise in frequency channels. A stable frequency channel is selected automatically and finally occupied to enhance the communication stability to avoid the presence of noise in frequency channels.

Keywords: PLC, IoT, noise, AMI

1. Introduction

Power line communication (PLC) is one of the attractive Internet of Things (IoT) solutions because of the well-established power grid. It has the advantages of low cost without any extra-lines installation, no concern of battery life due to AC power availability and relatively stable transmission path without geographical constraints [1]. PLC is mainly implemented in automated infrastructure and advanced metering infrastructure (AMI) [2]-[4]. In these IoT applications [2]-[4], low cost is prior to high bit rate, therefore, low bit rate and low cost solution are desirable. The power grid with noisy electric appliances is used by PLC. This noise interferes with the signal and corrupts the data packet, which causes the data transmission reliability problem. In the extreme case, no communication could be done in that frequency channel until the noise is disappeared. In this paper, a protocol on reconfigurable frequency switching and synchronization to avoid interference is proposed, which increases the PLC stability and efficiency.

2. Noise in power line

Noise sources in power lines are categorized into color background noise (CBG), narrowband interference (NBI) and impulsive noise (IN), [5]. Impulsive noise could be further distinguished into periodic impulsive noise synchronous to main frequency (PINS), periodic impulsive noise asynchronous to the main frequency (PINAS) and asynchronous impulsive noise (AIN). However, NBI and IN are the dominant noise sources in narrowband PLC applications [6]. The main sources of IN are the switching of electrical devices in the main power line and electrical devices with AC/DC converter. Loads with a bandwidth smaller than 5 kHz are due to NBI. All the above noise sources corrupt the data packet in the power line.

3. Methodology

Different PLC chipsets are available in the market. Narrow Band PLC (NB-PLC) and Broadband PLC (BB-PLC) are distinguished in terms of operating frequency range [7]. Various modulation methods used in PLC include Binary Frequency Shift Keying (BFSK), Binary, Quadrature and 8-Phase Shift keying (B/Q/8-PSK) and even Orthogonal Frequency Division Multiplexing (OFDM) [8]. The BB-PLC and other advanced modulations are commonly used for local high data rate networks such as WiFi PLC extender. Extra cost is used to pay for high performance is an unnecessary waste in this PLC application. Therefore, ST7540 NB-PLC with BFSK modulation from STMicroelectronics is suitable to implement the proposed frequency switching and synchronization protocol.

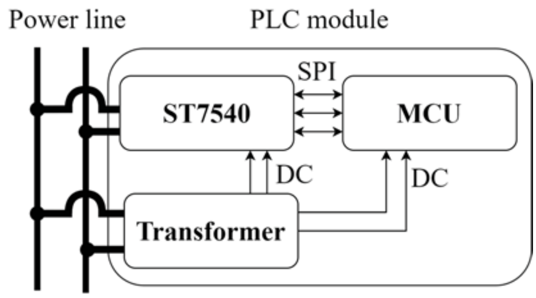

Fig. 1. Circuit diagram of ST7540 PLC module

Table 1: ST7540 register setting

Frequency channel	86/110/132.5kHz
Baud rate	4800 bps
Frequency detection time	5ms
Detection method	Carrier detection with conditioning

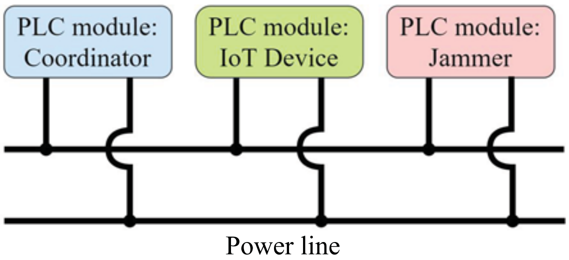

Fig. 2. Setup for the experiment.

Fig.1 shows the circuit diagram of the PLC module, which consists of ST7540, AC to DC transformer, micro-controller (MCU). The MCU is responsible for executing the protocol and initializing the control register of ST7540. The selection of essential registers is listed in Table 1.

Three frequencies were selected to be demonstrated in this experiment. All PLC modules were mainly utilized one of the channels for communication. The detection method register in ST7540 was set to carrier detection instead of preamble detection so that ST7540 informed the hosting MCU without verifying the value of preamble. And MCU was able to receive the noise signal. Frequency detection delay with 5ms was chosen to ignore the short ripples in the power line.

Three PLC modules with ST7540 were used in the experiment, which were in the roles of coordinator, IoT device and jammer respectively as shown in Fig. 2. The coordinator sends valid messages every 5 seconds. Jammer randomly sends invalid messages in the power line. The invalid messages violate the communication protocol so it is considered as noise. It also collides and corrupts the valid messages sending from the coordinator to IoT device. The corrupted messages will be examined and then considered as noise. The detail of protocol and noise identification procedure is discussed in the next section.

4. Proposed protocol

The packet structure overview is shown in Fig. 3. This proposed packet structure is designated to identify noise and valid packet as well as tackle the bit shifting problem. The packet is composed of two sections, header and payload. The payload contains the content and header contains protocol related information. Header is decomposed into four parts, synchronizer, customized preamble, payload length and play load cyclic redundant check (CRC).

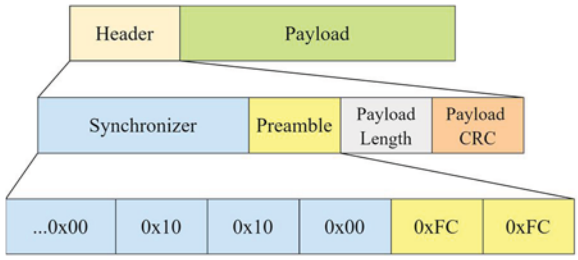

Fig. 3. Overview of packet structure.

Case 1: Correct synchronizer

Binary	0000 0000	**0001 0000**	**0001 0000**	0000 0000
Hex	0x00	**0x10**	**0x10**	0x00

Case 2: Synchronizer with 1 bit shifted to left

Binary	0000 0000	**0010 0000**	**0010 0000**	0000 0000
Hex	0x00	**0x20**	**0x20**	0x00

Case 3: Synchronizer with 3 bits shifted to left

Binary	**0000 0001**	**0000 0001**	0000 0000	0000 0000
Hex	**0x01**	**0x01**	0x00	0x00

Case 4: Invalid packet or noise

Binary	0000 0000	**0010 0100**	**1010 0000**	0000 0000
Hex	0x00	**0x24**	**0xA0**	0x00

Fig. 4. Different cases of synchronizer.

Fig. 4 shows the example of correct synchronizer (case 1), synchronizer with one bit shifted to left (case 2) and 3 bits shifted to left (case 3). All three synchronizers could be received when the coordinator sends the same packet. It is because the received bits are not properly aligned, bit shifting occurs for both header and payload. The hexadecimal values of each byte are changed. This bit shifting problem is solved by bit shifting operation. For example, case 1 already has the correct synchronizer, so no bit shifting is required. The packet from case 2 is shifted 1 bit to the left by comparing its synchronizer with the correct synchronizer, so 1 bit shifting to right for whole packet is required. 3 bits shifting to right with a similar procedure is required for the whole packet. In case 4, bits are not possible to be corrected to two consecutive 0x10 by bit shift operation, so this packet is considered as noise.

The valid header should pass all three synchronizers, preamble and payload CRC verifications. The noise classification flow is found in Fig. 5. When a noise or valid packet is received, the noise count and valid count will increase by one accordingly.

When the noise count exceeds the limit, which is selected to be 30 times in 60 seconds, the coordinators and IoT devices will change their communication channel. The coordinator will regularly transmit packets in all three possible channels. All IoT devices within the same power line will be informed which frequency the coordinator has chosen. If the IoT devices find their current frequency is not the same as the coordinator, they will follow the coordinator and switch frequency even the noise count does not reach the limit. The flow chart of frequency switching is shown in Fig. 6.

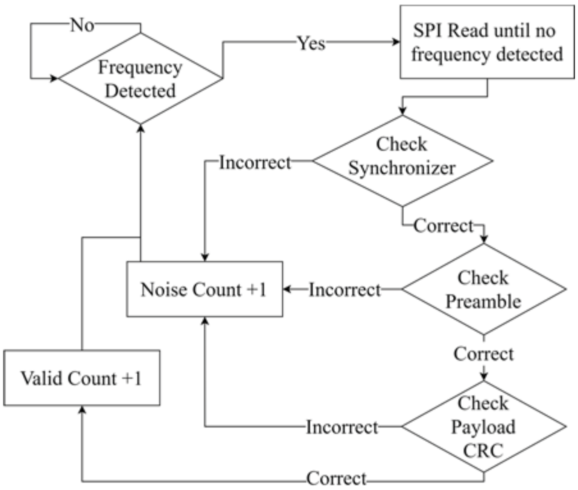

Fig. 5. Noise identification.

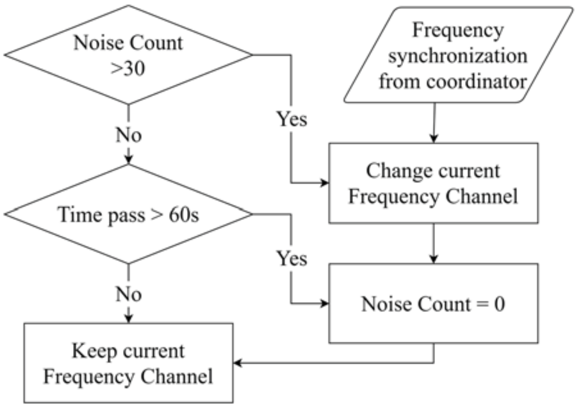

Fig. 6 Frequency switching logic for IoT device and coordinator.

5. Experimental Result

The noise count and valid count of the IoT devices are shown in Fig. 7. The IoT device, coordinator and jammer were started at 110 kHz and jammer was not powered up at the beginning. Before T_1, the IoT device consistently received valid packets from the coordinator in every 2 seconds. Therefore the growth rate of valid count is 0.5 packets per second from 10 to 30 seconds. Since the jammer was activated at T_1, the noise count started increasing and the valid count increased much slower, the growth rate of valid count is dropped to 0.2 packets per second from 40 to 60 seconds showing the valid packets collided to the noises generated by the jammer. The noise count exceeded the threshold of 30 in one minute, so the IoT device and the coordinator switch the operation frequency at T_2. Therefore, the growth rate of valid count turns to zero.

The IoT device and coordinator were switched to 132 kHz and 86 kHz respectively to demonstrate the capability of frequency selection synchronization. In real application, IoT devices and coordinator reconfigure their frequency following the same sequence, from 110 kHz to 132 kHz, then 86 kHz and finally back to 110 kHz. Normally, all IoT devices and coordinator at the same power line suffer from the same interference from the noise. Therefore, they should switch their frequency almost at the same time. However, if IoT devices and coordinates are placed with great distance, noise may not affect all devices and coordinator. The local noise only triggers some of the devices switching their frequency while others remain their frequency. The frequency selection synchronization is applied to avoid all the IoT devices and coordinator are not in the same frequency channel due to the local noise.

As the coordinator, jammer and IoT device were located at three different frequency channels from T_2 to T_3, both noise count and valid count remain stagnation. At T_3, the coordinator found the lost device by broadcasting frequency synchronization messages into the channels where the lost device was located. The messages commanded the device to follow coordinator's frequency selection. After T_3, since device and coordinator were in the same frequency channel, the valid count rose again without the disturbance of jammer.

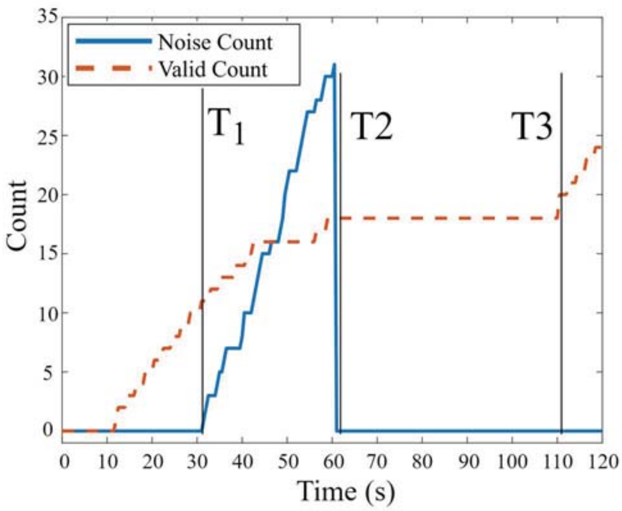

Fig. 7. Valid count and noise count of the device.

6. Conclusion

In this paper, a new reconfigurable frequency switching and synchronization protocol have been proposed. Its capability to identify noise from valid messages was proved. When the noise count reaches the threshold, new frequency channel is reconfigured independently. The switching process requires no communication with other devices and coordinators, so the switching process could be done even the channel is severely interfered. Frequency selection synchronization is applied and it has the capability to reconfigure all devices and coordinators into one frequency channel. The reconfigurable frequency switching and synchronization protocol, therefore is proposed to enhance the communication stability of PLC system.

This protocol is especially suitable for IoT application with large number of IoT devices as the proposed protocol does not require extra hardware. Also the protocol could be applied to the PLC chipsets with the simplest modulation method and enhances the PLC system performance. Therefore, low cost and reliable systems could be built on top of the proposed protocol.

References

[1] F. Corso, Y. Camargo and L. Ramirez, "Wireless Sensor System According to the Concept of IoT -Internet of Things-," *2014 International Conference on Computational Science and Computational Intelligence*, Las Vegas, NV, 2014, pp. 52-58, doi: 10.1109/CSCI.2014.17.

[2] F. Aalamifar and L. Lampe, "Optimized Data Acquisition Point Placement for an Advanced Metering Infrastructure Based on Power Line Communication Technology," in *IEEE Access*, vol. 6, pp. 45347-45358, 2018, doi: 10.1109/ACCESS.2018.2865592.

[3] M. Lisowski, R. Masnicki and J. Mindykowski, "PLC-Enabled Low Voltage Distribution Network Topology Monitoring," in *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6436-6448, Nov. 2019, doi: 10.1109/TSG.2019.2904681.

[4] Y. Ben-Shimol, S. Greenberg and K. Danilchenko, "Application-Layer Approach for Efficient Smart Meter Reading in Low-Voltage PLC Networks," in *IEEE Transactions on Communications*, vol. 66, no. 9, pp. 4249-4258, Sept. 2018, doi: 10.1109/TCOMM.2018.2828849

[5] M. Zimmermann and K. Dostert, "Analysis and modeling of impulsive noise in broad-band powerline communications," in *IEEE Transactions on Electromagnetic Compatibility*, vol. 44, no. 1, pp. 249-258, Feb. 2002, doi: 10.1109/15.990732.

[6] L. Bai, M. Tucci, S. Barmada, M. Raugi and T. Zheng, "Impulsive Noise Characterization in Narrowband Powerline Communication," in *Energies*, Vol.11 (4), p.863, April 2018, DOI: 10.3390/en11040863.

[7] S. Galli and T. Lys, "Next generation Narrowband (under 500 kHz) Power Line Communications (PLC) standards," in *China Communications*, vol. 12, no. 3, pp. 1-8, Mar. 2015, DOI: 10.1109/CC.2015.7084358.

[8] P. Achaichia, M. Le Bot and P. Siohan, "OFDM/OQAM: A Solution to Efficiently Increase the Capacity of Future PLC Networks," in *IEEE Transactions on Power Delivery*, vol. 26, no. 4, pp. 2443-2455, Oct. 2011, doi: 10.1109/TPWRD.2011.2140341.