Design and Implementation of IEEE 802.11ah (HaLow) Dongle for IoT Wireless Networking

Min-Cheol Kim, Young-Tak Kim*
Dept. of Info. & Comm. Eng, Yeungnam University
kmc724@ynu.ac.kr, *ytkim@yu.ac.kr

Abstract

In this paper, we present an IEEE 802.11ah (HaLow) dongle that provides sub-1GHz wireless channel and USB (universal serial bus) interface for power supply and wired data link for easy Internet of Things (IoT) wireless networking. The HaLow dongle is implemented using CC1352P1 sub-1GHz smart RF chipset that provides 50Kbps ~ 4Mbps in 10m ~ 1Km range. The major advantages of the HaLow dongle include i) easy installation using USB interface, ii) energy efficient wireless networking with flexible channel selection and Tx rate/power adjustments, and iii) user-friendly API (application programming interface) for raw-socket-based application developments. The functional architecture of IEEE 802.11ah (HaLow) dongle is explained in detail and the performance is analyzed in practical environments.

Ⅰ. Introduction

The IEEE 802.11ah (WiFi-HaLow)-based IoT (Internet of Things) must connect up to 8000 IoT devices which are scattered at various positions within 1 Km [1-3]. Since the IEEE 802.11ah is using 900MHz frequency band and providing 50Kbps ~ 4Mbps in 10m ~ 1 Km range, very efficient medium access control (MAC) protocol is necessary to provide data exchanges for mixed realtime and non-realtime IoT applications [4-7].

Even though the necessity of IoT applications are rapidly increasing in smart city/transportation, smart health, and smart farm/factory, there are limited number of commercially available IoT communication module that can provide 50Kbps ~ 4Mbps data rate in 10m ~ 1Km range. The commercial product LoRaWan [8] can provide wireless communication capability up to 20 Km, but its data rate is limited to 290bps ~ 50Kbps and it does not provide any API (application programming interface) for efficient channel configurations/managements and socket-based networking. Also, LoRaWan implements proprietary protocol which is not based on international standard IEEE 802.11ah (HaLow).

In this paper we present an IEEE 802.11ah (HaLow) dongle that provides i) simple cabling and easy installation using USB interface, ii) energy efficient wireless networking with flexible channel selection and Tx rate/power adjustments, and iii) user-friendly API (application programming interface) for socket-based application developments. The proposed HaLow dongle is using USB (universal serial bus) 2.0 interface by which both data exchanges and power supply are provided with simple cabling. So, if the IoT device platform provides USB port, sub-1GHz wireless communication link with IEEE 802.11ah can be easily configured by simply connecting HaLow dongle. Sub-1GHz wireless communication of HaLow dongle is implemented by Texas Instrument (TI) CC1352P1 sub-1GHz smart RF module [9].

The rest of this paper is organized as follows. In section II the related work on IoT wireless networking are briefly explained. In section III the functional architecture and software module of HaLow dongle are explained. Section IV analyzes the performance measurement results obtained in simple IoT networking with HaLow dongles within 1 Km range. Section V concludes this paper with brief introduction of future work.

Ⅱ. Related Work

As commercially available technology for constructions of IoT networking, LoRaWAN [8] is widely used in Europe. In Korea, SK Telecom configured IoT network with LoRaWAN. Currently, three LoRaWAN modules are available: LoRa-to-USB module (LoryNet – uLory) for IoT device, LoRa-to-Serial module (LoryNet – sLory) for IoT device, and LoryGate for IoT gateway.

LoRaWAN provides long distance communication capability (up to 20 Km), but its data rate is limited to 290bps ~ 50Kbps. LoRaWAN is using proprietary protocol, not following international standard IEEE 802.11ah. LoRaWAN does not provide API for channel selection and configurations, such as adjustments of Tx power and Tx data rate.

Texas Instrument (TI) provides wireless communication modules for IoT networking in sub-1GHz RF range: CC1352P1 (including Bluetooth feature) [9]. CC1352P1 modules provide 50Kbps ~ 4Mbps in 10m ~ 1Km range, and provide API for channel selection and configurations, such as adjustments of Tx power and Tx data rate. They also provide functions of listen-before-talk function and frame error check. The functions of IEEE 802.11ah MAC (medium access control) and channel configurations (i.e., channel selection, channel bandwidth, and adjustments of Tx rate/power), however, must be implemented additionally. In this paper, we implemented HaLow dongle with TI CC1352P1 and FT4222 SPI/USB interface module [10].

Ⅲ. HaLow Dongle for IoT Networking based on IEEE 802.11ah/Sub-1GHz

3.1 Functional Architecture of IEEE 802.11ah/Sub-1GHz HaLow Dongle

Fig. 1 depicts the functional diagram of IEEE 802.11ah HaLow Dongle. HaLow dongle is composed of two major chipsets: CC1352P1 and FT4222H. TI CC1352P1 chipset provides wireless communication using sub-1GHz RF channel. IEEE 802.11ah MAC protocol was implemented using the basic API of TI RTOS (real-time operating system). CC1352P1 is using dual cores: Cortex M0 for RF configurations and Cortex M4F for multi-thread processing on TI RTOS.

CC1352P1 is connected to outside using SPI (serial peripheral interface) and GPIO (general purpose input output). Control message for CC1352P1 configurations and user data frames are delivered through SPI, while interrupt from CC1352P1 to FT4222H is delivered through GPIO. FT4222H module is used in HaLow dongle to provide USB (universal serial bus) 2.0 interface that greatly simplify cabling and power supply.

HaLow dongle provides socket_halow API for easy IoT network programming using raw socket in IoT applications. The socket_halow is implemented based on USB1352LIB and LibFT4222 library. Socket_halow interface is provided in application layer instead of Linux kernel in order to be used in various environment which is not using Linux operating system. It is extending its flexibility to provide easy adaptability similar to wiringPi API for Raspberry Pi.

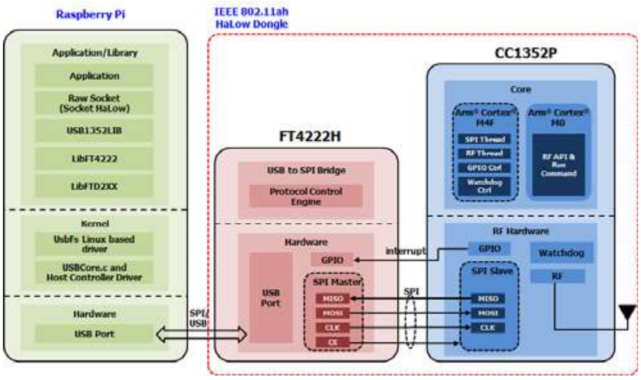

Fig. 1. Functional Block Diagram of IEEE 802.11ah HaLow Dongle

3.2 Socket_Halow API of HaLow Dongle

For easy development of IoT applications, the software module of HaLow dongle include socket_halow API that enables easy configurations of raw-socket communications between IoT devices and HaLow AP. The Socket_HaLow API includes socket(), probe(), and ioctl(). socket() is used to create a raw_socket for IoT wireless communication using HaLow dongle. probe() is used to check the connectivity and operational status of HaLow dongle, and is used before configurations of operational parameters by ioctl(). ioctl() is used to (re-)configure RF channel operational parameters, such as center frequency of RF channel, data rate, transmission (Tx) power. It is also used to obtain the current operational status information of RF channels, such as center frequency of each channel, RSSI (Receive Signal Strength Indication) and FER (Frame Error Ratio).

Since IoT devices are usually positioned at isolated locations, on-site checking and rebooting are not easy. HaLow dongle module provides remote power reset command using its ioctl command (HALOW_IOC_RESET) that makes the IoT device to reboot by itself and (re-)configure itself to the initialized operational parameters. During the practical field tests of the HaLow dongles, it was verified that the ioctl commands of socket_HaLow API provide great flexibility and adaptability for IoT devices located in diverse operational environment.

3.3 SPI Frame Structure

TI CC1352P1 Sub-1GHz RF module provides SPI (serial peripheral interface), I2C (inter-IC), I2S (integrated inter-chip sound), UART (universal asynchronous receiver transmitter), JTAG (joint task action group) interfaces to interconnect outside module. HaLow dongle uses SPI to connect CC1352P1 and Future Technology Device International (FTDI) FT4222H USB chipset. FT4222H USB-to-SPI bridge provides USB interface for HaLow dongle with simple cabling and power supply. The application programs on IoT device platform can configure SPI/USB connection by using LibFT4222 library function, and directly access the HaLow dongle [12].

We developed a software module (named USB1352LIB) for easy configuration, control, data frame transmissions and managements of CC1352P1 through USB interface. IoT application program can be easily configured with USB1352LIB, just as wiringPi supports easy IoT programming using GPIO (general purpose input output) interfaces of Raspberry Pi.

3.4 CC1352P1 RF Frame Structure

Fig. 2 depicts the RF frame structure of CC1352P1 module that supports 3 kinds of RF frame types (U-frame, S-frame, I-frame). U-Frame is used to transmit beacon, acknowledge frame of beacon, and handshaking management frames. When the IoT AP is configuring RF communication channel with an IoT device, the IoT AP obtains the identifier (ID) of the IoT device, sets the operation mode (i.e., normal response mode (NRM) or asynchronous balanced mode (ABM)), and checks the readiness of data exchanges by poll_RR (poll receiver ready). When there is any channel interference between IoT device and AP, the channel reconfiguration request (CHRC_Req) and channel reconfiguration acknowledge (CHRC_Ack) frames are exchanged to negotiate the channel reconfiguration.

For better performance of data transfer, HaLow dongle can configure block data transfer mode using three handshaking RF U-frames (i.e., Block Transfer Request, Block Transfer Request Acknowledge, and Block Transfer Acknowledge). Also, when there is serious problem in the IoT device, the AP can send request of power reset to the erred IoT device using Reset U-frame.

I-frames are used to deliver IoT application data frames between IoT device and IoT AP. I-frames carry addresses of sender and receiver and length of message stored in payload. I-frame FC (frame control) contains sender's sequence number (tx_seq) and receiver's sequence number (rx_seq) for flow control and error control of full-duplex bidirectional frame exchanges. If the receiver side does not have any frame to send with piggybacking of the receiver's sequence number, S-frame is used for flow control and/or error control.

Four S-frames (receiver ready (RR), reject(REJ), receiver not ready(RNR), and selective reject(SREJ)) are defined for flow control and error control of frame exchanges on CC1352P1 RF channel. If there is any frame error, a selective retransmission is requested by the receiver using SREJ S-frame.

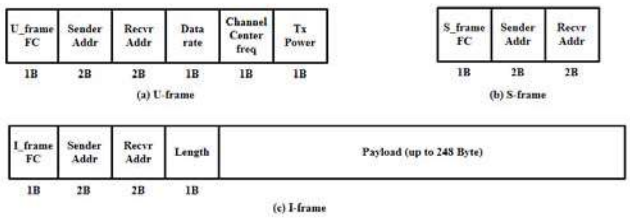

Fig. 2. CC1352P1 RF Frame Structure

3.5 AP –Device Interactions using HaLow Dongle

After power reset, the IoT AP initializes IoT network and broadcasts beacon RF frames periodically. Each un-registered IoT device makes registration after it receives beacon frame, as depicted in Fig. 3. The HaLow dongle is using hybrid slotted CSMA/CA-TDMA MAC, and for mitigation of massive collisions by registration requests from IoT devices, authentication control threshold (ACT) algorithm is implemented [5].

For each registered IoT device, the IoT AP selects the most appropriate RF channel for the IoT device, and exchanges CHRC (channel re-configuration) and CHRC_ACK frames. The CHRC frame contains the RF parameters of the selected channel. The IoT AP and IoT device make handshaking for synchronization of the selected RF channel using Poll_RR and RR frame exchanges. After channel configuration and synchronization, the IoT device exchanges application data to/from IoT AP. The IoT AP checks the available RF channel(s) and schedules the channel usages considering the amount of IoT application data exchanges from each IoT device.

The IoT AP and IoT devices continuously monitor the performance of the application data delivery using RF channel, and adjust Tx rate and Tx power. The performance of application data frame exchanges are maximized by using piggyback for bi-directional communication and block transfer for bulk data uploading/downloading.

When there are three consecutive frame losses while the RSSI is at good level, the receiver side determines that RF channel interference is occurred on the currently utilized channel between the IoT AP and the IoT device..

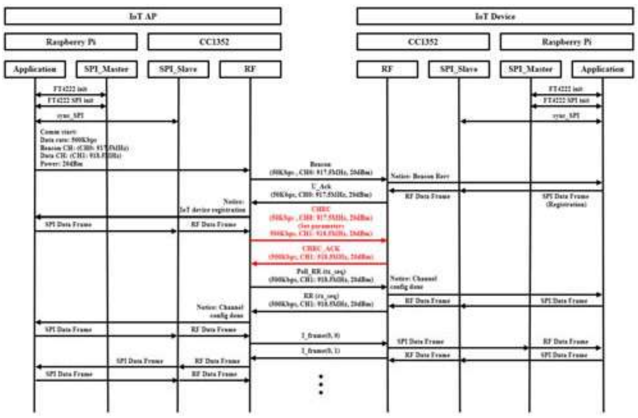

Fig. 3. Initial IoT Networking of IoT AP and IoT Devices

IV. Implementation and Performance Analysis of HaLow Dongle

4.1 Implementation of IEEE 802.11ah MAC on HaLow Dongle

Fig. 4 depicts the functional block diagram of IEEE 802.11ah HaLow dongle. HaLow dongle is equipped with FT4222H USB-to-SPI bridge that simplifies cabling between IoT device platform and HaLow dongle without separated power cabling. Fig. 5 shows the HaLow dongle attached to Raspberry Pi 4, and we can verify simple cabling.

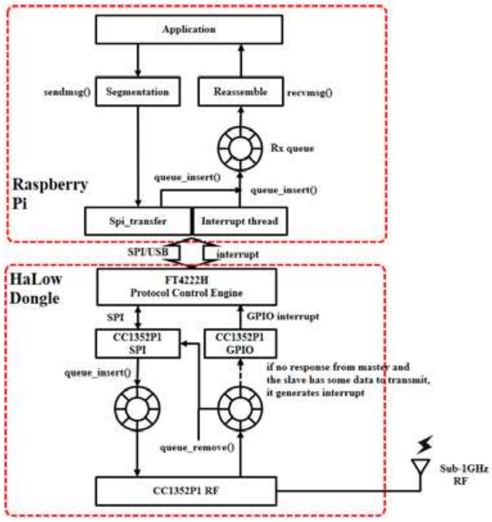

Fig. 4. Functional Block Diagram of IEEE 802.11ah HaLow Dongle

As explained before, HaLow dongle is using TI CC1352P1 that provides dual band low power RF (sub-1GHz and 2.4GHz Bluetooth), power amplifier (PA), and low noise amplifier (LNA). CC1352P1 can provide 5Kbps (long range mode) ~ 2Mbps (high speed mode) at 14dBm ~ 20dBm transmission power using 917MHz ~ 923MHz center frequency. CC1352P1 provides listen-before-talk which is essential in implementation of IEEE 802.11ah MAC protocol. For efficient usage of sub-1GHz RF channel, the hybrid slotted CSMA/CA-TDMA MAC protocol has been implemented using the TI RTOS on Cortex M4F of CC1352P1.

In current Proof of Concept (PoC) implementation, Raspberry Pi 4 with Raspbian operating system was used as IoT platform. For each implementation of IoT applications, raw socket interface (socket_HaLow) is provided with USB1352LIB. The detailed configuration and management functions of sub-1GHz RF channels are implemented using USB1352LIB APIs.

Fig. 5. HaLow Dongle attached to Raspberry Pi 4

4.2 Throughput of HaLow Dongle

In order to measure the maximum available throughput of HaLow Dongle for IoT applications, the performance was measured at near distance (5m) and maximum Tx power without frame bit error and channel interference. Table 6 compares the available throughput at each Tx rate of 5Kbps ~ 1Mbps. In order to measure the practically available throughput between IoT AP and IoT device, two RF frame exchange modes are used: (i) full-duplex bidirectional frame exchange with piggyback, (ii) half-duplex unidirectional frame transmission with acknowledgement by RF S-frame.

The measurement of maximum available throughput has been performed for Tx rate 5 Kbps ~ 1 Mbps, at distance 5 m, Tx power 20 dBm, RF frame size 200 bytes, antenna gain 2.5 dBi, and RSSI -25 dBm. In this configuration, there is no frame error if there is no channel interference. In full-duplex bidirectional frame exchange between IoT AP and IoT device, piggyback acknowledgement was used to enhance efficiency of RF channel. In half-duplex unidirectional frame transmission, downloading throughput was measured from IoT AP to IoT device with acknowledgements by RF S-frame.

From Table 1, we can find that HaLow dongle is providing up to 746 Kbps practical throughput at 1 Mbps Tx rate. In most other cases, the available throughputs were measured to be around 80 % of the Tx rate. The overheads in the sub-1GHz RF channel usage are RF channel (re-)configuration, Tx/Rx mode change, RF frame header, and handshake message exchanges for flow control and error control.

Table 1. Available Throughput at each Tx Rate

Tx Rate	Full-duplex (RSSI: -25dBm)	Half-duplex (RSSI: -25dBm)	Minimum RSSI	Remark
50Kbps	28.87Kbps	27.07Kbps	-95dBm	Distance between IoT AP and IoT Device = 5 m ~ 700 m Tx Power = 20dBm RF Frame Size = 254 bytes Antenna Gain = 2.5dBi
100Kbps	67.95Kbps	60.82Kbps	-94dBm	
200Kbps	159.93Kbps	141.41Kbps	-92dBm	
300Kbps	239.22Kbps	202.05Kbps	-88dBm	
400Kbps	321.48Kbps	269.55Kbps	-88dBm	
500Kbps	401.90Kbps	314.45Kbps	-80dBm	
1Mbps	746.38Kbps	566.15Kbps	-75dBm	

The minimum RSSI values for each Tx rate in Table 6 mean the threshold RSSI value for reliable delivery of RF frames. If RSSI value becomes lower than this threshold, RF frames are not correctly delivered. Fig. 6 depicts the frame error ratio (FER) according to RSSI and Tx rate at bidirectional RF frame exchange. In case of Tx rates less than 500Kbps, HaLow dongle provides RF communication with very low FERs if the RSSI is more than -70dBm. If RSSI is lower than -85dBm, the FERs were severely increased. In case of 1 Mbps Tx rate, FER increases slowly when RSSI is lower than -30dBm, and very high FER is measured when RSSI is lower than -60dBm.

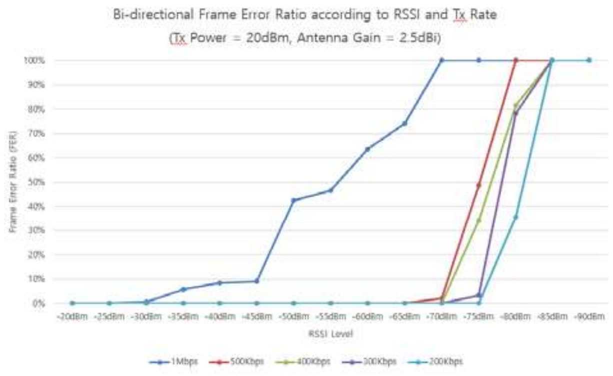

Fig. 6. Frame Error Ratio (FER) according to RSSI and Tx Rate (Bidirectional RF Frame Exchange)

V. Conclusion

In this paper, we presented an IEEE 802.11ah (HaLow) dongle that provides sub-1GHz wireless channel and USB (universal serial bus) interface for simplified cabling and power supply. The presented HaLow dongle is implemented using TI CC1352P1 sub-1GHz smart RF chipset that provides 5Kbps ~ 4Mbps in 10m ~ 1Km range. By using USB interface, the presented HaLow dongle greatly simplifies cabling without additional power supply.

The HaLow dongle implemented hybrid slotted CSMA/CA-TDMA MAC to maximize the utilization of sub-1GHz RF channel, and socket_HaLow API to support easy developments of IoT applications. The socket_HaLow API is also including ioctl (input and output control) commands that can be used in implementations of RF channel configurations and status monitoring for optimizations. Considering the fragile sub-1GHz license-free RF channel, HaLow dongle provides efficient flow control and error control scheme in hybrid slotted CSMA/CA-TDMA MAC layer.

The major advantages of the presented HaLow dongle include three folds: i) simplified cabling and easy installation using USB interface, ii) energy efficient wireless networking with flexible channel selection and Tx rate/power adjustments, and iii) user-friendly API (application programming interface) for raw-socket-based application developments. In this paper, we explained in detail the functional architecture of IEEE 802.11ah (HaLow) dongle and analyzed the performance in practical environments. As future work, various smart city/transportation and smart farm applications with IoT devices distributed within 1 Km range are under development using the presented HaLow dongle. And, smart control/configuration of RF channels is under development based on measured channel condition and actual throughput.

Acknowledgement

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program IITP-2020-2016-0-00313) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation).

References

[1] Jie Lin et.el, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications", IEEE Internet of Things JOURNAL Vol. 3 NO.5, oct, 2017, pp. 1125-1142.

[2] Stefan Aust, Venkatesha Prasad and Ignas G. M. M. Niemegeers, "Outdoor Long-Range WLANs: A Lesson for IEEE 802.11ah," IEEE COMMUNICATION SURVEYS & TUTORIALS, vol. 17, no. 3, 2015.

[3] T. Adame, A. Bel, B. Bellalta, J. Barcelo, and M. Oliver, "IEEE 802.11AH: The WiFi approach for M2M communications,"IEEE Wireless Communication, vol. 21, no. 6, pp. 144–152, Dec. 2014.

[4] Wireless Equipment Regulation, Ministry of Science and ICT 2016-125, 2016. 11. 30.

[5] Nurullah Shahin, Rashid Ali and Young-Tak Kim., "Hybrid Slotted-CSMA/CA-TDMA for Enhanced Registration of Massive IoT Devices," IEEE Access Vol. 6, pp 18366 - 18382, 2018.

[6] Min-Cheol Kim and Young-Tak Kim, "Smart Control of IoT Devices using IEEE 802.11ah/Sub-1GHz Wireless Networking," in Proc. of KNOM Conference 2019, 2019.

[7] Min-Cheol Kim and Young-Tak Kim, "Smart Control for Energy Efficient Networking of IEEE802.11ah-based IoT," in Proc. of APNOMS 2019, Matsue, Japan, Sept. 18-20, 2019.

[8] LoRa to USB converter (LoryNet – uLory), http://lory.co.kr/kor/LoryNet/view.php?part=1&idx=2.

[9] Texas Instrument (TI) CC1352P1 SimpleLink™ High-Performance Multi-Band Wireless MCU With Integrated Power Amplifier, http://www.ti.com/lit/ds/swrs192d/swrs192d.pdf?ts=1591592716336&ref_url=http://www.ti.com/tool/LAUNCHXL-CC1352P

[10] Future Technology Device International Ltd. FT4222H (USB2.0 to QuadSPI/I^{2}C Bridge IC), https://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT4222H.pdf.