Design and Implementation of an IoT Gateway for Zigbee and Wifi

RAY-I CHANG[1], YING-CHEN CHEN[1], CHI-CHENG CHUANG[1], CHIA-HUI WANG[2]
[1]Department of Engineering Science and Ocean Engineering, National Taiwan University,
No. 1, Sec. 4, Roosevelt Road, Taipei 10617, TAIWAN;
[2]Department of Computer Science and Information Engineering, Ming Chuan University,
No. 5 Der-Ming Rd., Gwei Shan District, Taoyuan City 333, TAIWAN

Abstract: - This paper proposed a framework called IAPA (Integrated Access Point Architecture) that applied an off-the-shelf AP (Access Point) device on WLAN (Wireless Local Area Network) as an IoT (Internet-of-Things) gateway to integrate Zigbee and Wifi. This Wifi-Zigbee AP (called WiZAP) has the advantages of low cost, small size and low power consumption. However, the coexistence of Wifi and Zigbee in unlicensed bands will introduce a serious data collision problem. In this paper, we propose a linear-time balance-first configuration strategy to adjust Wifi parameters to reduce data collision. A strategy to dynamically configure these parameters for MAC layer communication is also proposed. Experiments test the influence of network transmission in real application scenarios. Results show that our WiZAP can work efficiently with the coexistence of Wifi and Zigbee.

Key-Words: - WSN; WLAN; Wifi; Zigbee; System integration; Parameter configuration; IoT.

Received: September 16, 2021. Revised: May 5, 2022. Accepted: June 6, 2022. Published: July 2, 2022.

1 Introduction

In Wireless Sensor Network (WSN) application, sensed data are adopted by users through Internet, bringing out the demand of IoT (Internet-of-Things) gateway that connects Zigbee and Internet for information interchange. Traditionally, a personal computer is applied as a gateway to link WSN and Internet [1]. It encountered the following problems. (1) Its cost is high. (2) It consumes excessive power. Nowadays, the development of WLAN (Wireless Local Area Network) is increasingly prosperous. The WLAN AP (Access Point) has small size and low price. It motivates us to use an off-the-shelf WLAN AP to integrate Zigbee and Wifi as an IoT gateway.

In this paper, we design a new framework called IAPA (Integrate AP Architecture). Then, an open source software Oleg [6] is adopted to implement IAPA in a general WLAN AP called WiZAP (Wifi-Zigbee AP). It makes an embedded system which can actually support WSN by the plug of a sensor node on USB interface of WLAN AP to directly communicate with other sensor nodes. **Fig. 1** shows the application architecture of WiZAP. Our system is consistent with current Internet, transparent for user behavior, flexible for choosing routing protocol, and easy for integrating other networks with low setup cost. In addition, this IAPA framework further allows to perform integration between various application devices such as printer and webcam to develop value-add services.

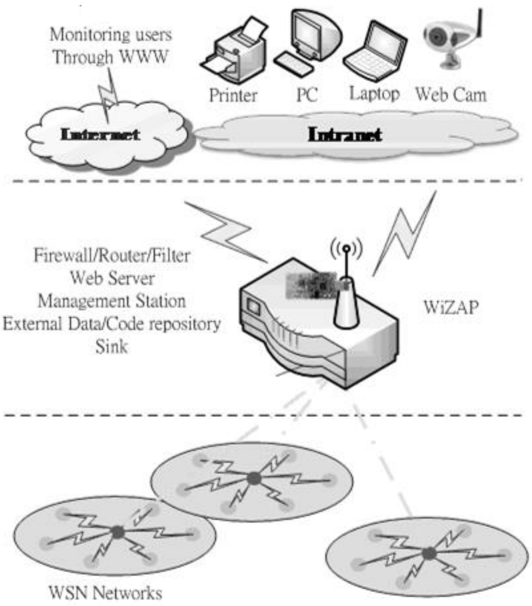


Fig. 1 The application architecture of WiZAP.

As our method has the advantages in low cost, small size and low power consumption, it can greatly expand the application of WSN. However, Zigbee and Wifi use the same frequency band, ISM 2.4GHz, which causes interference in transmission [2]. The coexistence of Zigbee and Wifi may introduce the data collision problem. If Wifi is too powerful, the throughput of Zigbee would be decreasing with high access delay. In some cases, connections have been re-established continuously and that even causes Zigbee to be unable to communicate. As the agreement of IEEE 802.15.4 and Zigbee does not clearly explain how to solve this problem, it results in a great obstruction in integration.

In this paper, we propose a linear-time balance-first configuration strategy to adjust Wifi parameters to reduce data collision. A strategy to dynamically configure these parameters for MAC layer communication is also proposed. Experiments test the influence of network transmission in real application scenarios. Results show that our WiZAP can work efficiently with the coexistence of Wifi and Zigbee. The remainder of this paper is organized as follows. In Section 2, our IAPA framework and its related works are introduced. In Section 3, we describe the proposed WiZAP system and Wifi parameter configuration strategy. Section 4 gives the experimental results. Conclusion and future works are shown in Section 5.

2 IAPA and its Related Works

Current approaches to integrate Internet and WSN can be classified as overlay-based approach (OBA) [4][5] and gateway-based approach (GBA) [1][3]. OBA modifies the network protocol of one or more nodes on one network and allows these nodes to communicate with another network. GBA sets a physical gateway between Internet and WSN for data exchange. GBA can be categorized into three methods: Application-Level Gateway (ALG) [1], Delay Tolerant Network (DTN) [3] and Virtual-IP Gateway (VIG). In ALG, an application program layer is added between heterogeneous networks to establish the connection. Different from ALG, DTN adds a bundle layer in the protocol stacks of each protocol to store data and transmits data through a store-and-forward manner. In VIG, the addresses of Internet and WSN are changed in the gateway by IP address mapping.

In this paper, for the simplification in design and implementation, we base on ALG to propose the IAPA framework. As the framework diagram shown in **Fig. 2**, IAPA is mainly divided into four parts. In short, RA (Router Agent) is the base system with drivers (hardware interfaces) which allows IAPA to adopt different network and storage devices. SC (Service Core) provides RPA (RePlicant Agent) as custom-made program to co-operate data in DB (Database). CA (Control Agent) is a module which is specially designed to communicate with WSN. SA (Service Agent) provides an interface for users to monitor and control the system.

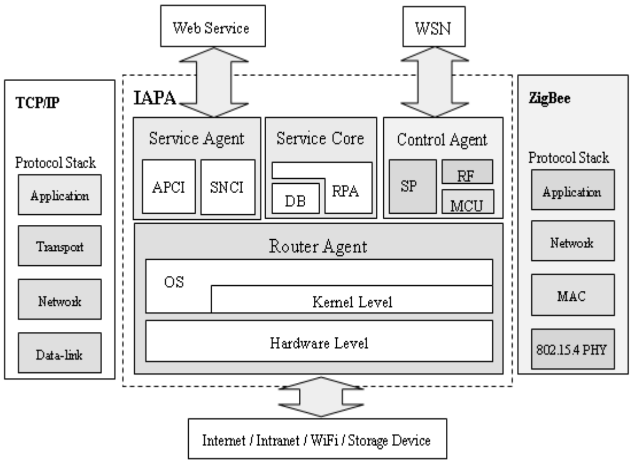


Fig. 2 Framework diagram of IAPA.

RA is divided into two levels, hardware and OS (Operating System). Hardware level is responsible for activating hardware device and can drive the hardware device at the bottom layer, such as wireless antenna, LAN port, WAN port and USB port. OS level controls the devices of hardware level, and provides a platform for establishing other modules. The kernel level in OS includes hardware driver programs which allows IAPA to control and access hardware level.

SC is consisted of RPA and DB for custom-made program. In order to develop application programs on embedded systems, a cross compiler is used to produce an executable binary file. This file is then moved to the system for execution by RPA. In SC, DB is used to stores data processed by the system. Users can view past WSN data through DB to reduce the power consumption of sensors.

CA can be divided into three components: RF (Radio Frequency), MCU (Micro-Controller Unit) and SPA (Serial Port Agent). RF is the physical layer in IEEE 802.15.4 to transmit and receive packets through Zigbee protocol. SPA is the communication bridge to transform the data collected by WSN. It can interpret and transmitting the data through connection ports. MCU processes the communication and coordination of all CA components. It is responsible for controlling the network/MAC layers in WSN protocol.

SA can be divided into APCI (AP Control Interface) and SNCI (Sensor Networks Control Interface). APCI offers an interface for users to configure the router. The interface can set various network services in the router. SNCI shows the current status of WSN and performs meaningful processes to the WSN data for checking the current environment status.

The advantages of IAPA are summarized as follows. (1) Consistent with Internet: IAPA is designed to support both IPv4 and IPv6 [7]. It has characteristics of consistency network working model for Internet. (2) Transparent for users: In IAPA, users can operate through webpage interface. They do not need to operate bottom information of network. (3) Flexible for routing: IAPA is a gateway used for communication between Internet and WSN. It can be compatible with different network protocols. (4) Easy for integration: IAPA allows to make a custom-made program to access and operate data from WSN. It allows to develop value-add services by integrating application devices such as printer and webcam. (5) Low setup cost: IAPA simply modifies the application layer, instead of the protocol stack. The setup cost is low.

3 WiZAP and its Configurations

This paper implements IAPA in an off-the-shelf WLAN AP, ASUS WL-500W. An open source software Oleg [6] is adopted to make the embedded system. The hardware of this WLAN AP has the basic ability to communicate with Wifi. By the plug of a sensor node on USB interface of WLAN AP USB-RS232 chip, we enable the communication of Zigbee. **Fig. 3** illustrates our installed WLAN AP called WiZAP.

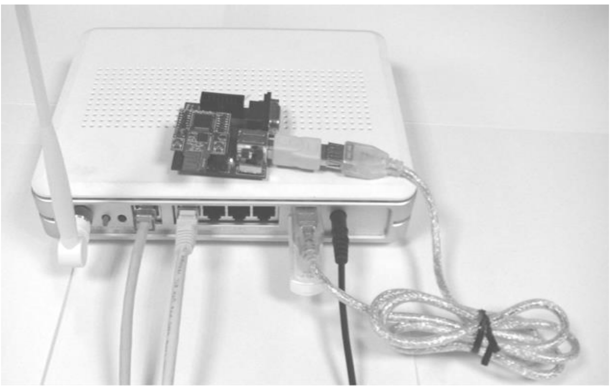


Fig. 3 Illustration of our WiZAP system.

WiZAP establishes Wifi and Zigbee to co-exist simultaneously in the same environment. As the same frequency band is applied, it may lead to signal interference and packet losses called the data collision problem. For resolving this problem, we propose a way to configure Wifi parameters to balance the transmission quality of Wifi and Zigbee by regulating the MAC layer communication of WLAN AP. In order to test the transmission quality of Wifi, we referred to the relevant values related to data transmission and network interface cards in SNMP. Then, we measure the communication quality of Zigbee by Eq. (1).

$$\text{PRR (Packet Receive Rate)} = G / A \quad (1)$$

where A is total amount of packets to be received and G is the packet amount received.

Note that, altering certain Wifi parameters in the different configurations may increase PRR ratio of Zigbee, but decrease the transmission speed of Wif. Therefore, how to establish appropriate parameter configurations dynamically according to the current system situations is an important matter. In this paper, we propose a linear-time balance-first configuration strategy for acquiring network transmission balance which allows effective enhancement in the transmission quality of Zigbee where the PRR value is relatively low.

Our network configuration algorithm is shown as follows where WIFI_CH is the index of Wifi channel applied, WIFI_POWER is the transmission power of Wifi channel, DATA_RATE is the data rate applied, WIFI_FT is the fragmentation threshold of Wifi packet size, and the beacon interval BEACON_INV is the time interval for packet transmission.

```
ALGORITHM: Balance-first configuration
    IF (has no transmission) THEN Exit;
    IF (has Wifi transmission only) THEN {
        Use default parameter values of WLAN AP;
        Exit;
    }
    IF (has Zigbee transmission only) THEN {
        Increase BEACON_INV;
        Decrease WIFI_POWER;
        Exit;
    }
    IF (has both transmission) THEN {
        Select an unused Wifi channel;
        IF (all Wifi channels are used) THEN Exit;
        IF ( ZIGBEE_PRR < THRESHOLD) THEN {
            Decrease DATA_RATE;
            Decrease WIFI_POWER;
            Decrease WIFI_FT;
            Increase BEACON_INV;
        }
        ELSE {
            Increase DATA_RATE;
            Increase WIFI_POWER;
            Increase WIFI_FT;
            Decrease BEACON_INV;
        }
    }
```

4 Experiments

In this paper, we test WiZAP to observe the interference situations that occurred during the communication between WiZAP and Zigbee. The system architecture of our experiment is shown in **Fig. 4**. **Table 1** shows the parameter setting of Zigbee used in our experiments. We set Zigbee to use the last RF channel 26 to avoid data collision. **Table 2** shows the parameter setting of Wifi tested in our network configuration algorithm. Notably, we end the test of Wifi channels at 11 as the channel 10 starts to have high data collision with Zigbee.

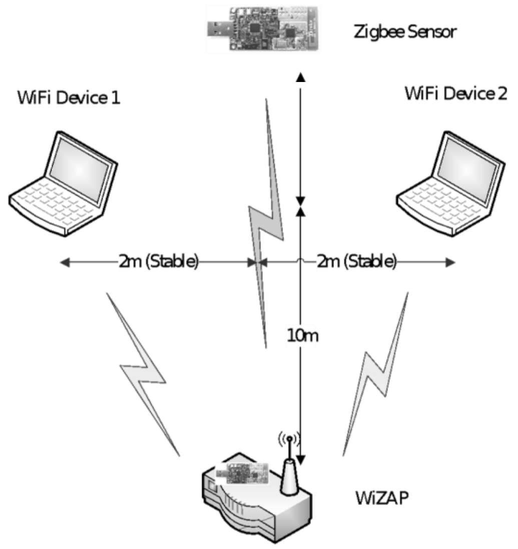


Fig. 4 The system architecture of our experiment.

Table 1 The parameter setting of Zigbee.

Zigbee Parameters	Setting
Zigbee Channel (index)	26
Transfer Rate (kbps)	200
Payload size (bytes)	75

Table 2 The parameter setting of Wifi.

Wifi Parameters	Setting
Wifi Channel (index)	6, 7, 8, 9, 10, 11
WIFI_POWER (mW)	1, 19, 40, 62, 84
DATA_RATE (Mbps)	1, 2, 5.5, 11
BEACON_INV (ms)	10, 100, 500, 1000, 2000, ..., 65000, 65536
WIFI_FT (bytes)	256, 512, ..., 2346

Based on these parameter configurations, two experiments are made to test the influence on Wifi's performance by Zigbee. (1) First, by deactivating or activating Zigbee, the effects on Wifi were observed. (2) By comparing the effects on Zigbee with different Wifi parameters, we adjusted and determined the parameters that are good to enhance the system performance. As shown in **Fig. 5**, the gray bar is Zigbee_OFF (deactivating Zigbee) and the black bar is Zigbee_ON (activating Zigbee). According to the experiment results, deactivating or activating Zigbee shows little influence on Wifi. It motivates us to keep Zigbee_ON and tries to adjust and determine suitable Wifi parameters.

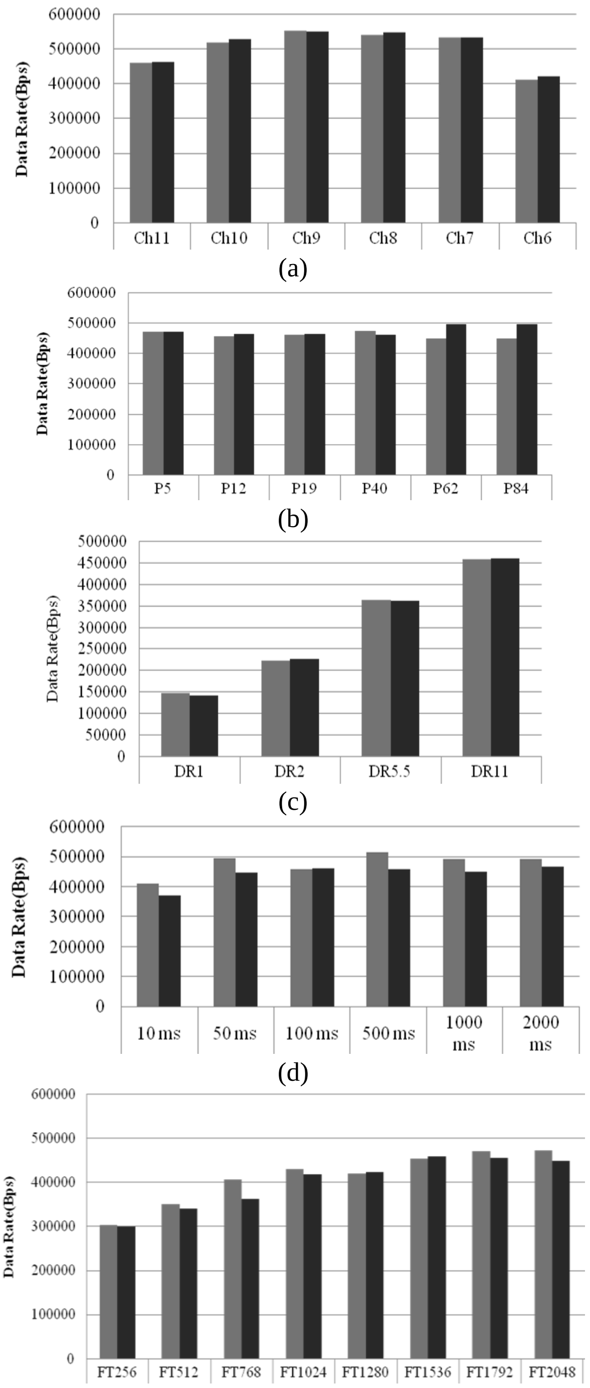
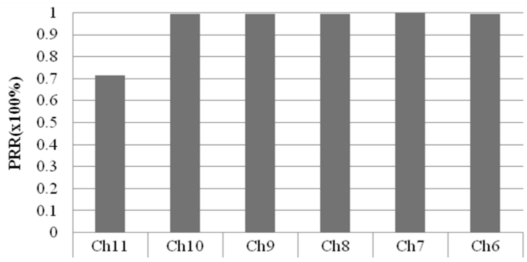
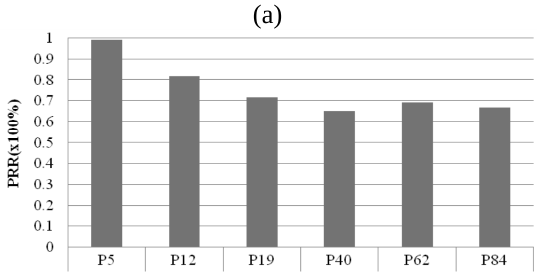


Fig. 5 The influence on Wifi's performance by Zigbee where the gray bar is Zigbee_OFF and the black bar is Zigbee_ON for different Wifi parameters.
(a) Wifi Channel. (b) WIFI_POWER. (c) DATA_RATE. (d) BEACON_INV. (e) WIFI_FT.

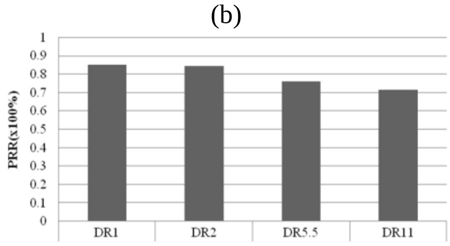
The experiment results for the influence on Zigbee PRR by the adjusted Wifi parameters are shown in **Fig. 6**. When adjusting the Wifi channels, the channel 11 of Wifi has data collision with channel 26 of Zigbee. The reduction of WIFI_POWER can significantly improve the Zigbee PRR. The reduction of DATA_RATE can contribute to the improvement of PRR. Zigbee PRR is reduced when BEACON_INV is adjusted downwards, and increased vice versa. By the way, the configuration of WIFI_FT has little influence on the PRR of Zigbee.



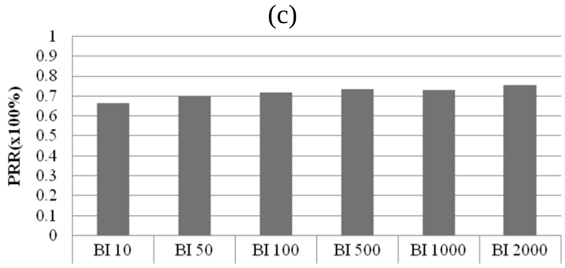
(a)



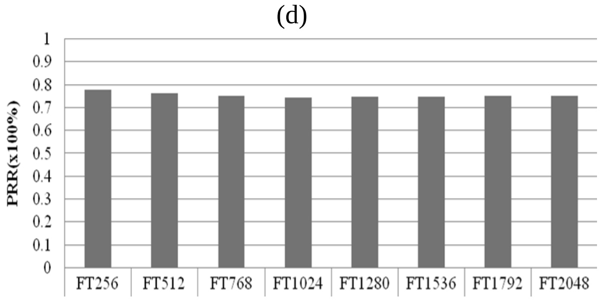
(b)



(c)



(d)



(e)

Fig. 6 The influence on Zigbee PRR by the adjusted Wifi parameter configurations.
(a) Wifi Channel. (b) WIFI_POWER. (c) DATA_RATE. (d) BEACON_INV. (e) WIFI_FT.

By summarizing these experiment results, we can show a procedure to adjust and determine suitable Wifi parameters to maximize Zigbee PRR with consistent Wifi connection. It shows that adjusting Wifi channel to a non-interference channel achieves the best improvement (about 28%). Otherwise, set WIFI_POWER at the smallest value can also enhance the PRR of Zigbee (about 27%). If lowering WIFI_POWER leads to disconnection, then adjust DATA_RATE, BEACON_INV and WIFI_FT to promote the PRR of Zigbee. The improvements are about 14%, 10%, and 6%, respectively.

5 Conclusion and Future Works

This paper adopts an off-the-shelf WLAN AP and proposes an integral architecture, IAPA, to implement the WiZAP system. It allows the connection of two heterogeneous networks, Zigbee and Wifi, by a modular and low cost manner. By adjusting MAC layer communication parameters of Wifi, we address some observations on the data collision problems faced when Zigbee and Wifi co-exist. Then, we propose a linear-time balance-first configuration strategy for acquiring network transmission balance. WiZAP has advantages of high elasticity, low cost, convenient establishment and low power consumption. It can be a reference model for the next generation WLAN AP for IoT applications.

References:

[1] R. Bosman, J. Lukkien, R. Verhoeven. (2011). Gateway Architectures for Service Oriented Application-Level Gateways. IEEE Trans. on Consumer Electronics. 57. 453-461.

[2] R. Musaloiu-E., A. Terzis. (2008). Minimising the effect of WiFi interference in 802.15.4 wireless sensor networks. IJSNet. 3. 43-54.

[3] D. Adamou, V. Vèque, A. Mouradian, S. Hoteit, L. Zitoune. (2021). Efficient Data Collection using Aerial Mobile Sinks for Delay Tolerant Wireless Sensor Networks. 17-24.

[4] A. Dunkels, J. Alonso, T. Voigt, H. Ritter, J. Schiller, "Connecting wireless sensor nets with TCP/IP networks," WWIC, 2004.

[5] H. Dai, R. Han, "Unifying micro sensor networks with Internet via overlay networking," IEEE EmNets-I, 2004.

[6] http://oleg.wl500g.info/

[7] D. Yang, Q. Guo. (2012). DACH: an efficient and reliable way to integrate WSN with IPv6. International Journal of Distributed Sensor Networks. 2012.