

논문 2018-13-16

저전력 네트워크 기반의 확장 용이한 스마트 홈 IoT 시스템

(An Extensible Smart Home IoT System Based on Low-power Networks)

이 준 영, 유 성 은*

(Jun-young Lee, Seong-eun Yoo)

Abstract : There are increasing interests on smart home systems. However, most of the existing works focus on the functionality itself. In this paper, we propose an extensible smart home system based on low power networking such as CoAP, 6LoWPAN, and Zigbee. The proposed home IoT system consists of Home APP, Home Server, Home Broker, and Power Devices. Each component of the system is connected by the low-power network technologies aforementioned. As the end device, Power Device senses the current consumption of the attached appliance and controls the power to it. Power Device reports the sensing data to Home Server via Home Broker. The Home Broker enhances the scalability of the system. Home Broker extends the service area and the user's services, and it manages the connection of the underlying devices and processes, and transmits data to Home Server from Power Devices. Through the experimental evaluation, we show that the proposed system achieves the design goals such as extensibility and low power networking.

Keywords : Low-power network, Home network, IoT, CoAP, Smart home

1. 서 론

IoT (Internet of Things)란 사물들을 지능화하고 통신 기술을 사용하여 통신과 컴퓨팅의 기능이 있는 모든 것을 연결하는 것이다 [1-3]. 이러한 연결된 모든 것은 상호 간의 정보를 교환하고 소통하여 지능형 서비스를 제공한다. 또한 스마트 디바이스, 클라우드, 빅데이터 등의 기술을 통해 모든 것이 연결되는 초-연결사회를 형성한다. 2017년에 Gartner는 연결된 “Things”가 2020년에는 204억 개에 도달할 것으로 전망한다 [4]. 현재 IoT의 응용 분야는 교통 분야, 생산 분야, 국방 분야 [5]와 복지 분야 [6] 뿐만 아니라 보다 사용자층이 넓은

*Corresponding Author (seyoo@daegu.ac.kr)

Received: Apr. 30 2018, Revised: May 24 2018,

Accepted: June 2 2018.

J.Y. Lee, S.E. Yoo: Daegu University

※ 이 논문은 2014학년도 대구대학교 학술 연구비 지원에 의하여 연구되었음.

스마트 홈 혹은 가전을 포함하여 매우 광범위하다 [7].

기존의 스마트 홈 시스템의 연구 [8]는 상용 스마트 홈 시스템이 built-in 방식이기 때문에 이후 구성을 변경하기 힘들다는 점과 통신사의 제휴를 통해서만 네트워크에 연결할 수 있다는 점을 문제점으로 들었다. 위 문제들을 해결하기 위하여 Open Hardware Platform 기반으로 Plug-in & Play 방식의 Power Device를 설계하였으며 사이트의 중앙 관리자인 IoT Gateway로 Power Device를 추가 및 제어할 수 있으므로 높은 확장성을 제공한다. 하지만, 해당 연구에서는 디바이스의 확장성을 주로 고려하여 네트워크 구성에 있어 통신 거리와 전력 소모 측면을 소홀히 한 면이 있다.

IoT 디바이스는 배터리를 사용해서 동작이 가능한 다수의 에너지 제약적인 디바이스로 구성되어 데이터 통신이 갖기 때문에 사물인터넷에서의 저전력 네트워킹 기술 분야는 매우 중요하다. 이와 같은 환경을 고려하여 IETF 6LoWPAN WG에서는 IEEE 802.15.4 기반 하드웨어에서 IPv6 주소체계와

IPv6 패킷 헤더를 압축하는 6LoWPAN 기술을 RFC4944를 통해 표준화 한다 [9]. 또한 IETF CoRE (Constrained RESTful Environments) WG 에서 CoAP (Constrain Application Protocol)이 표준화했다. CoAP은 저능능 노드와 저전력 손실 네트워크를 위해 REST (Representational State Transfer) 방식의 경량화된 메시지를 사용한다 [10].

따라서 본 논문에서는 저전력을 고려한 네트워크로 구성된 확장 용이한 스마트 홈 IoT 시스템을 제안한다. 제안된 홈 IoT 시스템은 Home APP, Home Server, Home Broker, Power Device로 구성되고 이들은 저전력 네트워크로 연결된다. 스마트 폰에서 동작하는 Home APP은 Power Device를 제어할 수 있고 Home Server에 저장된 각 센서의 전류 값을 그래프로 출력할 수 있다. Home Server는 Home Broker로부터 센서 값을 전달받고 Database에 저장한다. Home Broker는 기존 연구 [8]에서 Zigbee의 근거리 네트워크 문제점을 보완한다. 기존 시스템에서는 단일 홈 통신을 Zigbee로 구성하였기 때문에 서비스 제공범위가 좁아 사용하기에 많은 무리가 있었다. 이를 보완하기 위하여 Home Broker는 Power Device에서 전류 값을 전달받아서 Home Server로 전송하고 Power Device의 작동을 제어하고 연결 유무를 확인함으로써 Service area와 시스템의 확장성을 기존에 비해 강화한다. Power Device는 주기적으로 전류 사용량을 센싱하여 Home Broker로 그 값을 전송하고 Home Broker에서 전달받은 제어 메시지에 따라 Relay를 동작시켜 가전기기의 전원을 제어한다.

본 논문에서는 저전력 네트워크 기술인 응용 계층의 CoAP과 WPAN (Wireless Personal Area Network)에서 IPv6를 지원하도록 하는 적층 계층의 6LoWPAN, WPAN 기술인 Zigbee를 사용하여 시스템을 구성한다. Home APP은 기존 HTTP 대신 패킷의 사이즈가 작은 CoAP으로 Home Server에 데이터를 송수신한다. Home Server는 Home Broker와 저전력 통신을 위해 IEEE 802.15.4 통신을 사용하여 6LoWPAN으로 연결된다. Home Broker는 클라이언트가 서버의 리소스 상태를 감시할 수 있는 CoAP의 Observe 기능을 사용하였다. Home Broker와 다수의 Power Device는 XBee 모듈을 사용하여 저전력 통신 기술인 IEEE 802.15.4 Zigbee 통신으로 1:N으로 연결된다.

본 논문은 2장에서 본 시스템에서 사용된 기술에 대해 기술 및 기존 홈 IoT 연구에 대해 설명한다.

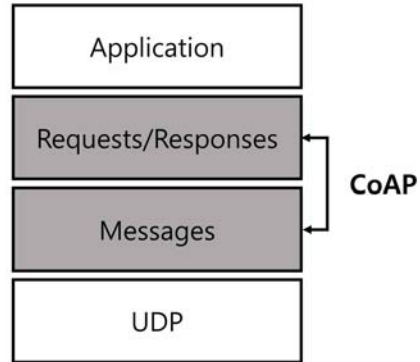


그림 1. CoAP의 추상 계층
Fig. 1 Abstract layering of CoAP

이를 바탕으로 3장에서 제안 시스템에 대한 설계 내용을 기술하고 4장에서 구현 내용 및 결과를 분석한다. 마지막으로 5장에서 전체 내용을 요약하고 추후 연구 방향을 제시한다.

II. 관련 연구

관련 연구에서는 본 논문에 사용한 주요 기술인 CoAP과 6LoWPAN에 대해 기술한다. 그리고 기존 홈 IoT 연구에 대해 설명한다.

1. 표준화 동향

IETF에서 2010년 3월에 CoRE 워킹 그룹을 결성하였다. 이후 RFC 7252 [10]를 2014년 6월에 제정하였다. CoAP은 인터넷의 제한된 노드 및 환경에서의 운영을 위한 웹 전송 프로토콜로 M2M (Machine-to-Machine) 애플리케이션을 위해 디자인되었다. 데이터 전송 크기를 줄이기 위해 축약된 메시지 헤더와 옵션 헤더를 가진다. CoAP 프로토콜은 클라이언트/서버로 구성되었으며 UDP기반의 RESTful한 프로토콜이다. 그러므로 기존 HTTP처럼 리소스를 사용하지만 적은 헤더를 가지고 있다는 장점이 있다.

그림 1은 IETF RFC 7252를 통해 CoAP의 구조를 정의한 CoAP의 추상적 계층을 나타낸다. CoAP은 데이터그램 방식의 전송계층 위에서 비동기적으로 전송한다. CoAP 프로토콜의 메시지는 그림 2와 같은 메시지 헤더와 옵션 헤더를 가지고 있다. 메시지는 4Byte의 Header (헤더), 0에서 8Byte의 토큰 (Token), 옵션 필드 (Option Filed),

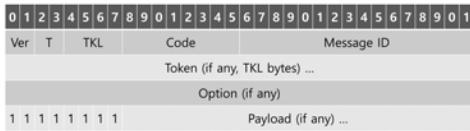


그림 2. CoAP 프로토콜의 메시지 포맷
Fig. 2 Message format of CoAP protocol

페이로드 (Payload)로 구성된다. 4Byte 헤더에는 v (버전), T (트랜잭션 타입), TKL (토큰 길이), Code (코드), MID (메시지 ID)로 구성된다. CoAP은 메시지의 크기를 줄이고 바이너리 인코딩을 사용한다.

CoAP의 메시지 타입은 Confirmable, Non-confirmable, Acknowledgement (ACK), Reset (RST)가 있다. Confirmable 메시지는 흐름은 그림 3과 같다. 송신에서 Confirmable 메시지를 전송하면 수신은 ACK를 반환한다. 그러면 Confirmable 메시지와 ACK 메시지의 MID를 확인한다. 문제가 있을 경우 RST 메시지를 전송한다. 신뢰성 있는 전달이 필요 없는 경우에는 Non-confirmable 메시지를 사용함으로 ACK를 기다리지 않는다. Code는 첫 3bit의 Class, 뒤의 5bit의 Detail로 구성된다. Class는 Indicate a request (0), Success response (2), Client error response (4), Server error response (5)이다. 요청 메시지의 경우 Class에 0을 사용하고 GET (1), POST (2), PUT (3), DELETE (4)을 사용한다 [10-12].

6LoWPAN은 기존의 IPv6를 저전력 WPAN에 응용하기 위해 표준화된 기술로 IPv6 패킷을 압축하는 기술을 활용한다. 6LoWPAN은 Adaptation 계층을 두어 IPv6 헤더를 압축하고 자동으로 MAC 주소에 따른 IPv6를 배포한다. 기존의 IEEE 802.15.4 표준을 따라 2.4GHz 대역에서 250Kbps의 저성능 통신을 한다. 그러므로 6LoWPAN은 저속, 저성능, 저비용의 무선 통신 네트워크에 적합하다. 기술적으로 단순하고 저렴한 뿐만 아니라 IPv6 주소 지정이 가능하기 때문에 무수한 디바이스를 연결할 수 있다. 기존 IPv4에서는 기하급수적으로 늘어나는 IoT 디바이스에 주소를 할당할 수 없을 것이다. 하지만 IPv6는 주소크기가 32bit에서 128bit로 증가하면서 무제한에 가까운 주소를 할당할 수 있다 [9, 13].

2. 기존 연구

기존 스마트 홈 IoT 시스템 [8]의 연구에서는 개방형 하드웨어를 기반으로 한 스마트 홈 아키텍

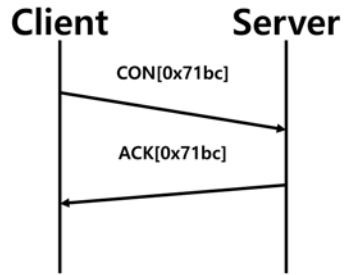


그림 3. CoAP의 확인 메시지
Fig. 3 Confirmable message

처를 제시한다. 갈릴레오 보드 (Galileo gen1)와 Zigbee 기술을 사용하여 저전력 시스템을 구성하였다. 서버는 디바이스로부터 센서 정보를 수집하고 제어한다. 디바이스는 Relay를 사용하여 쉽게 가전 제품과 연결이 가능한 Plug-in & Play 방식으로 설계되었다. 모바일 앱은 연결된 제품들의 모니터링과 제어 인터페이스를 제공한다. 하지만 단일 홈의 Zigbee로 구성하여 좁은 통신 거리에서만 서비스가 가능하고 모바일과의 HTTP 통신을 사용함으로 네트워크의 전력을 고려하지 않은 문제점이 있다.

CoAP/6LoWPAN 기반의 홈 네트워크 시스템은 6LoWPAN을 BLE (Bluetooth Low Energy)와 Zigbee/IEEE802.15.4 모듈의 게이트웨이를 이용해서 각각의 서버를 분리하여 구성하였다 [14]. 게이트웨이는 외부 인터넷으로부터 이더넷 연결을 통해서 웹서버로 접속을 가능하게 하였다. 사용자는 게이트웨이를 통해 다수의 센서를 관리하는 Server에 요청을 보내서 제어하도록 한다. 하지만 이 시스템은 각각의 Server를 두어야하는 문제가 있다.

BLE 네트워크에서 사물인터넷 서비스를 위한 CoAP과 6LoWPAN을 구현한 연구에서는 HTTP 통신과 6LoWPAN을 활용한 CoAP의 성능을 분석하였다 [15]. 성능을 분석한 결과는 HTTP보다 CoAP의 처리량이 2배가 상승하였고 평균 전송시간이 21% 감소하였고 패킷 전송량도 22% 적음을 확인하였다. 즉 저사양 하드웨어에서 HTTP에 비해 2배 이상의 높은 효율을 보인다 [16].

III. 시스템 설계 및 구성

본 논문에서는 기존 시스템의 확장성을 유지하며 CoAP, 6LoWPAN, Zigbee 통신 기술을 사용하여 저전력 네트워크를 고려한 스마트 홈 IoT 시스템을 설계하였다. 그림 4는 전체 시스템 구성도를

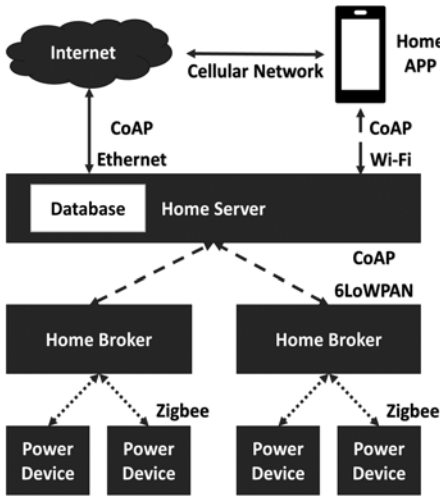


그림 4. 전체 시스템 구성도
Fig. 4 System block diagram

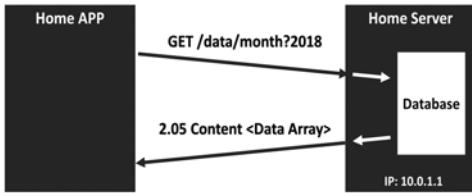


그림 5. Home APP에서의 요청 메시지
Fig. 5 Home APP request message

나타낸 것이다. 제안된 시스템은 Home APP, Home Server, Home Broker, Power Device로 구성되어 있다. 본 논문에서는 Top-Down 방식으로 시스템 설계 및 구성을 기술한다.

1. Home APP

사용자는 스마트 폰의 Home APP으로 Power Device를 제어할 수 있고 전류 값을 효율적으로 확인할 수 있다. Home APP과 Home Server간에는 CoAP으로 통신한다. Home APP은 Home Server에 메시지를 송신하면 Home Server는 사용자의 요청을 수행하고 결과 값을 전송한다. Home APP에서는 전송받은 결과 값을 사용하여 년, 월, 일, 최근의 사용량을 그래프로 보여준다.

Home APP은 Home Server로 URI (Uniform Resource Identifier)를 전송하여 원하는 서비스를 요청할 수 있다. 예로 Home APP에서 2018년 월

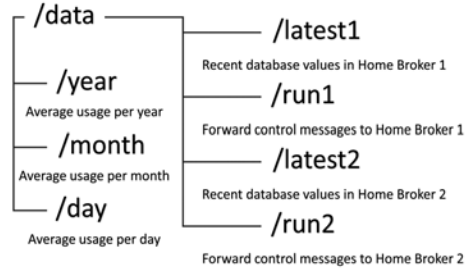


그림 6. Home server의 리소스
Fig. 6 Home server resources

별 사용량의 데이터가 필요하다면 다음과 같이 리소스를 조합한다. CoAP Server는 주소가 10.0.1.1이고 포트 번호는 5683을 사용한다고 가정한다. Home APP은 사용자가 월별 사용량에 대한 버튼을 클릭하면 Schema가 coap://이고 Home Server의 IP 주소, Path가 data/month, Query를 2018로 하는 coap://10.0.1.1:5683/data/month?2018을 전송한다. Home APP은 Home Server로부터 응답으로 받은 데이터로 그래프를 출력한다. 메시지 전송의 흐름은 그림 5와 같다.

2. Home Server

Home Server는 Home Broker로부터 받은 센서 값을 Database에 저장한다. 그리고 Home Server는 Home APP이 요청하는 URI에 따라 요청을 수행하고 결과를 전송하거나 제어 요청을 Home Broker로 전송하는 역할을 한다. 즉 Home Server는 연결된 모든 하위의 Home Broker에서 전송하는 데이터를 총괄하고 모든 Power Device를 제어할 수 있도록 한다.

그림 4를 보면 Home Server는 AP (Access Point)로 동작하여 Home APP과 바로 연결할 수 있다. 또한 Ethernet을 사용하여 인터넷에 연결된 경우에는 도메인 또는 IP 주소를 통해서 Home Server로 연결할 수 있다. 그러므로 외부 인터넷을 사용하지 않아도 시스템을 구성할 수 있을 뿐만 아니라 인터넷에 연결된 경우 외부에서도 제어할 수 있다는 장점이 있다. 이 기능을 사용하기 위하여 Home APP은 접속할 Home Server의 IP주소나 도메인을 지정할 수 있도록 한다.

그림 6은 Home Server에서 Home Broker가 2개인 경우 사용하는 리소스를 나타낸다. 년, 월, 일의 데이터를 처리하는 리소스와 최근 데이터 값을 요청하고 Power Device를 제어하는 리소스를 정의

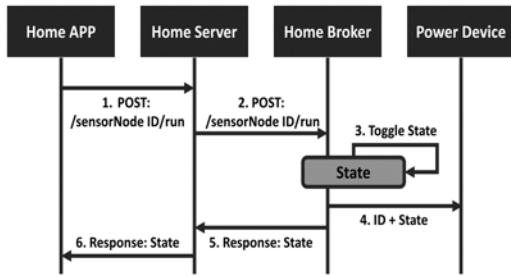


그림 7. 제어 메시지 전송과 응답

Fig. 7 Control message transmission and response

하였다. 예를 들어 Home APP에서 첫 번째 Home Broker에 있는 ID가 1인 Power Device를 제어하려면 Path가 /data/run1인 POST 메시지에 ID 번호인 1을 포함하여 전송한다.

Home Server는 각 Home Broker로 전달받은 Power Device의 센서 값을 Database에 저장한다. 전달받은 값은 각 Home Broker에 대한 테이블 안에 저장한 시간을 저장하는 time, Power Device의 ID 번호는 id, 센서 값은 data의 컬럼으로 저장한다.

3. Home Broker

Home Broker는 기존 Zigbee 통신의 좁은 Service area를 보완한다. Power Device를 통해 받은 값에 대해 처리를 하고 Home Server로 전달한다. 또한 하위의 Power Device에 연결된 Relay의 동작을 제어하는 메시지를 전송하고 상태를 저장하고 주기적으로 연결 상태를 확인하도록 한다.

Home Broker는 Power Device와는 Zigbee로 통신을 한다. 또한 Home Broker는 6LoWPAN으로 Home Server와 통신한다. Home Broker는 계속해서 값을 송수신하는 역할을 하는 디바이스로 저전력 네트워크로 구성한다. 그리고 IP 기반의 Home Broker를 사용하여 시스템을 효율적으로 관리할 수 있다.

Home APP이 Home Server로 제어할 Power Device의 ID에 대한 제어 신호를 전송하면 Home Server는 ID가 속해있는 Home Broker의 IP 주소로 Power Device의 ID가 포함된 리소스를 전송한다. Home Broker에서는 하위 Power Device의 동작 상태를 'on'과 'off'로 저장한다. Home Broker는 전달받은 ID에 해당하는 Power Device의 동작 상태를 반전 후 제어 메시지를 보낸다. 그리고 역순으로 제어된 상태를 전송함으로써 Home APP에서

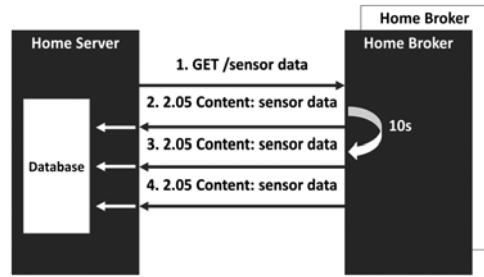


그림 8. Observe 기능에서의 메시지 교환

Fig. 8 Exchange of messages in the observe function

제어 상태를 확인할 수 있다. 그림 7은 Home Server에서 메시지를 받아서 Power Device를 제어하는 순서를 나타낸다. 또한 Home Broker는 Power Device와의 연결을 검사하는 기능을 수행한다. Home Broker가 Power Device에서 3초 동안 값을 전달받지 못하면 해당 Power Device의 연결이 단절된 상태로 전환한다. 즉 Home Broker는 각 Power Device에 대한 동작 상태와 연결 상태의 2가지 정보를 저장한다.

Home Broker에서는 각 Power Device에 대한 Observe [17] Server가 동작한다. Observe 기능은 CoAP Client가 CoAP Server에 감시 옵션을 추가한 GET 요청으로 리소스에 대한 관심을 등록한다. CoAP Server는 등록된 리소스의 상태가 변경될 때 관심을 등록한 CoAP Client에게 응답으로 알림 (Notification)을 보낸다. 그림 8은 Observe Server로 동작하는 Home Broker에서 Observe Client인 Home Server로의 메시지 교환을 나타낸다. Home Broker는 각 Power Device에 대해 Observe Server로 동작한다. Home Broker는 Home Server가 등록한 Observing Resources인 Sensor data에 대하여 감시하고 변화가 감지될 경우 Home Server로 알림을 전송한다. 만약 Power Device의 작동이 중지 상태 또는 연결이 단절 상태 일 경우는 값을 전송할 필요가 없기 때문에 Observe 기능을 중단한다.

4. Power Device

Power Device는 Relay와 전류 센서로 구성되어 있다. Home Broker로부터 Message를 통해 특정 Power Device의 Relay를 제어할 수 있다. 또한 각종 전자 제품이 소모하는 전류량을 1초 주기로 측정하고 Home Broker로 전송하는 기능을 수행한다.

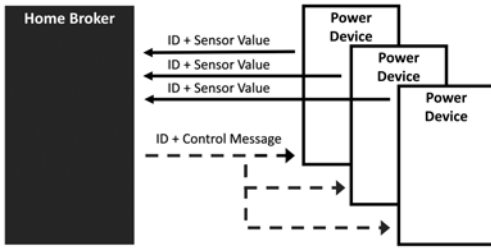


그림 9. Power device에서의 정보 교환

Fig. 9 Exchange of information on power device

Power Device는 Zigbee로 Home Broker와 통신한다. Zigbee는 낮은 전송 속도를 가지는 대신 다른 통신 기술과 비교하여 낮은 전력 소모의 특성을 가진다. 그러므로 Zigbee는 Power Device에 적합한 통신 기술이다.

그림 9와 같이 Power Device와 Home Broker 간 메시지 교환을 한다. Power Device는 Home Broker로 ID 필드와 센서 값을 함께 전송한다. Home Broker는 ID 필드를 통해 Power Device를 구분할 수 있다. Home Broker가 전송한 제어 메시지를 Power Device가 수신하면 ID 필드를 확인한다. 같은 ID의 Power Device는 Control Message를 통해 동작을 제어한다.

IV. 구현 결과 및 시스템 분석

본 장에서는 시스템 구현의 결과를 기술한다. 제안된 스마트 홈 IoT 시스템에서 Home APP과 Home Server, Home Broker간에는 CoAP으로 통신한다. 제안된 시스템은 이클립스 재단에서 관리하는 자바 기반의 오픈 소스 라이브러리인 Californium CoAP 라이브러리 [18]를 사용하여 구현되었다. 그림 10은 Home APP을 구현한 결과이다. Home APP은 사용자가 전류 사용량을 효과적으로 분석할 수 있도록 데이터를 그래프로 출력한다. 사용자는 년, 월, 일에 따른 평균 데이터를 확인할 수 있다. Home APP에서 최근 데이터도 조회할 수 있으므로 사용자는 실시간으로 사용량을 알 수 있다. 그리고 사용자는 Home APP에서 제어할 Power Device에 해당하는 버튼을 눌러 가전 기기의 전원을 쉽게 제어할 수 있다. 이를 통해 가정 내의 전력 소모량을 효율적으로 절약하는 데 도움을 준다.

Home Server와 Home Broker는 라즈베리파이 보드를 사용하여 구현되었다. Home Server와

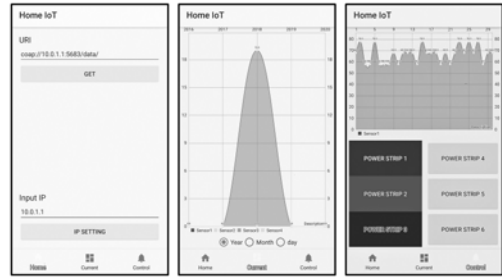


그림 10. 구현된 Home APP 화면

Fig. 10 Implemented Home APP screen

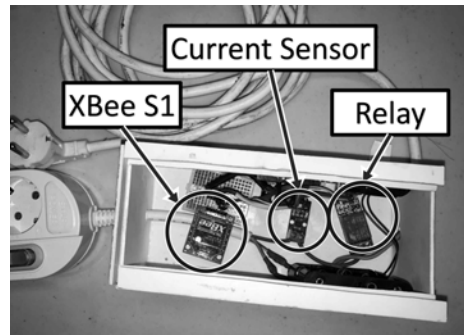


그림 11. Power device

Fig. 11 Power device

Home Broker 간의 6LoWPAN으로 연결하기 위하여 Home Server와는 IEEE 802.15.4를 지원하는 openlab의 Raspberry Pi 802.15.4 radio를 사용한다 [19]. 또한 Home Broker와 Power Device 간의 통신은 Zigbee 호환 모듈인 XBee S1 모듈이 사용되었다.

Power Device는 가전기기를 연결한 콘센트를 Relay를 통해서 제어함으로써 연결된 가전기기의 전원을 제어할 수 있다. 그림 11은 구현된 Power Device의 모습이다. Power Device는 Arduino를 메인보드로 사용하여 통신을 위한 XBee 모듈과 전원을 제어할 Relay, 전류 값을 측정할 전류 센서로 구성된다.

제안된 스마트 홈 IoT 시스템에서는 대기 중인 네트워크의 전류 소모도 고려하였다. 라즈베리파이에서 Wi-Fi와 Zigbee 그리고 6LoWPAN의 Network Interface 사용 시 대기 전력 소모를 비교하였다. 전류 측정은 오차율은 5% 내외로 50mA에서 5A의 전류를 측정할 수 있는 MV212 Coms USB를 사용하였다. 테스트에서 라즈베리파이의

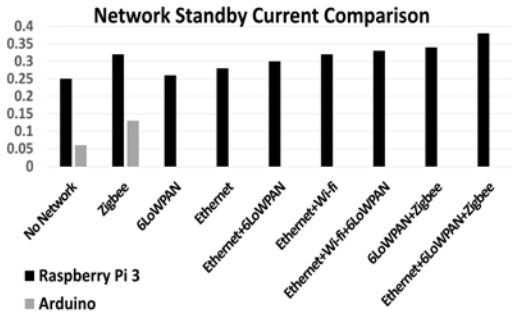


그림 12. 네트워크 대기 전력 비교

Fig. 12 Network standby current comparison

Network Interface는 ifdown 명령어로 중지하고 XBee 모듈은 연결하지 않은 상태에서 측정하였다.

그림 12은 대기 전류 사용 테스트의 결과이며, 송수신 시 소모 전류는 데이터 시트를 참고하였다. openlab의 802.15.4 radio에서는 AT86RF233 트랜시버를 사용하여 최대 송신 전류가 13.8mA이며 수신 시 11.8mA로 적은 송수신 전류 [20]를 사용하고 테스트에서도 가장 적은 대기 전류를 소모함을 알 수 있다. 라즈베리파이에 내장된 BCM43143 Wi-Fi는 송신 전류가 368mA~427mA, 수신 전류는 68mA~94mA로 비교적 높은 송수신 전류 소모량을 알 수 있다. 라즈베리파이에 내장된 BCM43143 Wi-Fi는 송신 전류가 368mA~427mA, 수신 전류는 68mA~94mA로 비교적 높은 송수신 전류 소모량 [21]을 가지고 802.15.4 radio에 비해 높은 대기 전류를 사용한다. XBee S1 모듈은 송수신 전류가 각각 45mA와 50mA로 상대적으로 낮은 전류 [22]를 사용하지만 다른 Network Interface에 비해 대기 전류 사용량이 높은 것으로 측정되었다. 그러므로 저전력성을 고려하면 Wi-Fi에 비해 802.15.4 radio가 적합하다고 고려된다.

기존 시스템 [8]에서는 HTTP 통신으로 사용자의 요청을 수행한다. 기존 시스템과 성능을 비교하기 위하여 HTTP 통신과 CoAP 통신의 에너지 효율성을 비교하였다. 에너지 효율성에 직접적인 영향이 있는 패킷과 바이트 수를 측정하기 위하여 네트워크 통계 유틸리티인 IPTraf를 사용하였다. 테스트는 클라이언트에서 동일한 Network Interface로 HTTP과 CoAP으로 센서 값을 서버에 요청한 경우 서버에서 사용된 모든 패킷과 바이트 수를 분석하였다. HTTP의 경우 총 34개의 패킷에 3,602Byte를 사용하였고 CoAP은 단 2개의 패킷에 117Byte를 사용하여 사용자에게 센서 값을 전달하였다. 즉, HTTP에 비해 CoAP이 데이터 송수신 시 전송량이 매우 적기 때문에 기존 시스템보다 CoAP을 사용하는 본 시스템이 에너지 효율적임을 알 수 있다.

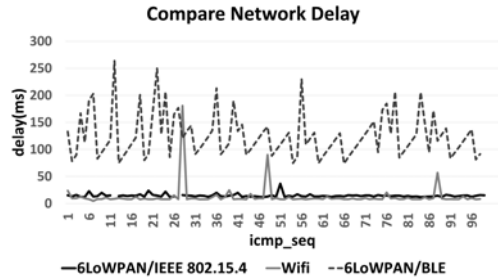


그림 13. 네트워크 delay 비교

Fig. 13 Compare network delay

다음으로 Network Interface의 Home Server와 Home Broker 간의 단일 Hop에서의 패킷 전송 딜레이를 분석하였다. 연결에 따른 통신 성능을 비교하기 위해서 ping 어플리케이션을 사용하여 홈 대 홈으로 100개의 패킷을 전송하여 딜레이와 패킷 손실률을 테스트하였다. 그림 13은 딜레이의 테스트 결과를 나타낸다. BLE에서 6LoWPAN을 사용한 6LoWPAN/BLE는 평균 138ms의 딜레이가 발생하지만 패킷의 손실은 없었다. Wi-Fi는 평균 9ms의 딜레이가 발생하고 6LoWPAN/IEEE 802.15.4에서 4%의 패킷의 손실과 평균 13ms의 딜레이가 발생하였다. 즉, 딜레이 측면에서는 Wi-Fi와 6LoWPAN/IEEE 802.15.4가 우수한 성능을 보였다. 다만, 6LoWPAN/IEEE 802.15.4에서 패킷 손실이 발생한 원인을 분석하기 위하여 IEEE 802.15.4 통신의 거리와 Tx_Power를 변경하면서 추가적인 테스트를 진행하였다. 최고 출력인 4dBm에서 두 디바이스간 거리가 10cm일 때 손실률이 없었으나 1m의 거리에서 평균 6%의 손실이 발생하였다. 동일한 거리에서 -4dBm 출력에서는 평균 12%의 패킷 손실을 보였다. 또한, 다른 시간대에 최고 출력으로 실험을 한 결과 3m거리에서 손실이 발생하지 않은 경우도 있었다. 즉, IEEE 802.15.4 통신은 출력 세기, 전송 거리 및 주변 간섭에 따라서 패킷 손실률의 편차가 크게 발생하였다.

V. 결론

본 논문에서는 기존 홈 IoT 시스템 [8]을 저전력 및 확장성 측면에서 보완하였다. 제한된 Home Broker는 기존의 IEEE 802.15.4 (Zigbee)의 좁은 Service area를 확장한다. Home Broker는 Power Device의 센서 값 및 Home Server에서의 제어 메시지를 전달하는 역할을 수행하여 Service area를

확장하였다. 사용자는 Home APP을 사용하여 Home Broker에 연결된 Power Device의 동작을 관리하고 연결 상태를 검사하는 기능을 통해 시스템의 확장이 용이하다. 그리고 기존의 홈 IoT 시스템은 저전력을 고려하지 않은 HTTP를 사용했다는 문제점을 가지고 있었다. 이를 해결하기 위해 HTTP보다 메시지 크기가 작고 저사양 하드웨어에서 전력 효율적인 CoAP을 사용하여 시스템을 구성하였다. Home Broker는 Home Server로의 잦은 센서 값의 전달을 위해 CoAP Observer 기능으로 동작함으로써 요청 메시지가 없어도 값을 전달한다. 시스템의 모든 구성 요소들은 저전력 네트워크 기술인 6LoWPAN과 Zigbee를 사용하여 매우 에너지 효율적이다. 추후 연구에서는 전력 소모가 큰 XBee 모듈을 대체할 IEEE 802.15.4 모듈과 저전력 Hardware Platform에서 6LoWPAN을 적용하여 모든 디바이스에서 CoAP을 통한 메시지 교환을 테스트할 예정이다.

References

- [1] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions," *Future Generation Computer Systems*, Vol. 29, No. 7, pp. 1645-1660, 2013.
- [2] M. Kong, H. Chae, B. You, "Trend and prospects of Internet of Things(IoT)," *Journal of Korea Design Knowledge*, Vol. 56, No. 2, pp. 32-36, 2016 (in Korean).
- [3] D.G. Nam, "The Development of Internet of Intelligent Things and Future Service Model," *Journal of the Korean Institute of Communication Sciences*, Vol. 27, No. 7, pp.3-9, 2010 (in Korean).
- [4] <https://www.gartner.com/newsroom/id/3598917>
- [5] T.M. Lee, J.H. Lee, J.H. Yoo, J.H. Wang, S.E. Yoo, "AND-OLSR: Adaptive OLSR Protocol Based on Average Node Distance in Dispersion Falling Mobility Model," under review (in Korean).
- [6] S. Min, Y. Oh, "G-eye: Route Guide System Based on Obstacle Map for Blind people," *Proceedings of HCI Society of Korea*, pp. 703-705, 2017 (in Korean).
- [7] S.E. Yoo, A.M. Cheng, P.K. Chong, T.S. Lopez, T.H. Kim, "Technological Advances in Wireless Sensor Networks Enabling Diverse Internet of Things Applications," *International Journal of Distributed Sensor Networks*, Vol. 14, No. 3, 2018.
- [8] J.H. Lee, G.I. Park, J.H. Shin, S.E. Yoo, "An Extensible Smart Home IoT System Based on Open Hardware Platforms," *IEMEK J. Embed. Sys. Appl.*, Vol. 11, No. 6, pp. 369-377, 2016 (in Korean).
- [9] G. Montenegro, N. Kushalnagar, J. Hui, D. Culler, "Transmission of IPv6 Packets Over IEEE 802.15.4 Networks," No. RFC 4944, 2007.
- [10] Z. Shelby, B. Frank, D. Sturek, "Constrained Application Protocol (CoAP)", No. RFC 7252, 2014.
- [11] S.K. Ko, S.C. Soon, S.H. Oh, B.T. Lee, "The Latest Standard Technology of IETF CoAP," *OSIA Standards & Technology Review*, Vol. 28, No. 4, pp. 74-86, 2015 (in Korean).
- [12] W.Q. Jin, D.H. Kim. "Implementation and Experiment of CoAP Protocol Based on IoT for Verification of Interoperability," *Journal of the Institute of Internet, Broadcasting and Communication*, Vol. 14, No. 4, pp. 7-12, 2014 (in Korean).
- [13] K.H. Kim, S.W. Yoo, K.M. Kim, C.S. Rim, J.S. Park, "Low-Power Sensor Network (6lowpan) Technology Based on IEEE 802.15.4," *Communications of the Korean Institute of Information Scientists and Engineers*, Vol. 25, No. 12, pp. 60-65, 2007 (in Korean).
- [14] D.S. Moon, G.J. Park, Y.H. Lee, B.K. Lee, E.H. Jeong, "Design and Implementation of Home Network System Based on CoAP/6LoWPAN," *Proceedings of Korea Software Congress*, pp. 1340-1342, 2017 (in Korean).
- [15] C.M. Kim, H.W. Kang, S.I. Choi, S.J. Koh, "Implementation of CoAP/6LoWPAN Over BLE Networks for IoT Services," *Journal of Broadcast Engineering*, Vol. 21, No. 3, pp. 298-306, 2016 (in Korean).
- [16] S.J. Lee, D.M. Kim, Y.C. Choi, J.G. Hong, J.I. Kim, S.J. Koh, "Analyze the Differences in

- per Second of Transfer Processing Amount Between the HTTP Protocol and CoAP Protocol”, Proceedings of Korean Institute of Information Scientists and Engineers, pp. 1838-1840, 2015 (in Korean).
- [17] K. Hartke, “Observing Resources in the Constrained Application Protocol (CoAP),” No. RFC 7641, 2015.
- [18] <https://github.com/eclipse/californium>.
- [19] <http://openlabs.co/OSHW/Raspberry-Pi-802.15.4-radio>
- [20] MCU Wireless AT86RF233 Datasheet, Atmel
- [21] CYW43143 Datasheet, Cypress
- [22] Digi XBee S1 802.15.4 RF Modules datasheet, Digi

Jun-young Lee (이 준 영)



He received B.S. degree in information and communication engineering from Daegu University in 2018. His research interests include IoT, linux system, and sensor networks.

Email: l jy908@daegu.ac.kr

Seong-eun Yoo (유 성 은)



Seong-eun Yoo received the B.S. degree in electronics and computer engineering from Hanyang University, Seoul, Korea, in 2003 and the M.S. and Ph.D. degrees in information and communications engineering from KAIST, Daejeon, Korea, in 2005 and 2010, respectively. Since September 2010, he has been a faculty member with the School of Computer and Communication Engineering, Daegu University, Gyeongsan, Korea. His research interests include real-time communication in wireless sensor networks and real-time embedded systems.

Email: seyoo@daegu.ac.kr