

## 라즈베리 파이에서 MQTT 프로토콜 이용 시 전력 소모 분석

강승우\*

### Characterizing Power Consumption of MQTT Protocol Usage on Raspberry Pi

Seungwoo Kang\*

School of Computer Science and Engineering, Korea University of Technology and Education, Cheonan 31253, Korea

#### 요 약

라즈베리 파이는 사물 인터넷 애플리케이션 개발을 위한 범용 하드웨어 플랫폼으로 널리 사용되고 있다. 사물 인터넷에 연결된 각종 기기들이 수행하는 기본적인 작업은 다양한 센서를 이용하여 주변 환경 정보 혹은 기기의 상태 정보를 획득하고 이를 게이트웨이 혹은 원격 서버로 전송하는 것이다. 이를 위해 사물 인터넷 응용 프로토콜인 MQTT가 많이 사용되고 있다. 사물 인터넷 기기는 보통 배터리 파워로 동작하는 경우가 많고 배터리 교체 없이 장시간 동작할 수 있어야 한다. 따라서 센서 데이터 획득 및 전송 작업을 처리하는 데 있어서 에너지 효율성이 매우 중요한 문제 중 하나이다. 에너지 효율적인 사물 인터넷 애플리케이션을 만들기 위해서는 해당 애플리케이션의 전력 소모 특성에 대한 정보가 필수적이다. 하지만 사물 인터넷 기기에서 사물 인터넷 통신 시 전력 소모를 분석하는 연구는 많이 이루어지지 않았다. 본 논문에서는 사물 인터넷 하드웨어 플랫폼으로 널리 사용되는 라즈베리 파이에서 MQTT를 이용하여 데이터를 송수신하는 다양한 상황의 전력 소모를 측정하고 분석한다.

#### ABSTRACT

Raspberry Pi has been widely used for a hardware platform to develop the Internet of Things (IoT) applications. The basic task of diverse IoT devices is to obtain their status or environmental information by using various sensors and to send them to a gateway or a remote server. For the purpose, one of application layer protocols for IoT, MQTT is widely used. IoT devices are often powered by batteries and they are required to operate for a long time without replacing or recharging their batteries. Thus, energy efficiency is one of the most critical problems for obtaining and sending sensor data. To develop energy-efficient IoT applications, the information about power consumption characteristics of the applications is necessary. However, there has been little study to analyze power consumption of IoT communication on IoT devices. This paper presents a study to measure and analyze the power consumption of sending/receiving data under various conditions via the MQTT protocol on Raspberry Pi.

**키워드** : 라즈베리 파이, 사물 인터넷, MQTT, 전력 소모, 전력 측정

**Key word** : Raspberry Pi, Internet of Things, MQTT, Power consumption, Power measurement

Received 28 July 2017, Revised 07 August 2017, Accepted 22 August 2017

\* Corresponding Author Seungwoo Kang(E-mail: swkang@koreatech.ac.kr, Tel: +82-41-560-1406)

School of Computer Science and Engineering, Korea University of Technology and Education, Cheonan 31253, Korea

Open Access <https://doi.org/10.6109/jkiice.2017.21.12.2347>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.  
Copyright © The Korea Institute of Information and Communication Engineering.

## I. 서 론

최근 사물 인터넷 애플리케이션이나 시스템 개발을 위한 범용 하드웨어 플랫폼이 많이 출시되고 있다. 여기에는 라즈베리 파이, 인텔 Edison, 삼성 ARTIK, 아두이노 등 여러 가지 플랫폼이 있다. 그 중에서 라즈베리 파이는 저가의 싱글 보드 컴퓨터로 임베디드 리눅스 기반 개발 보드로 널리 사용되고 있다. 그리고 라즈베리 파이를 이용한 사물 인터넷 시스템 개발을 위한 연구도 이루어지고 있다 [1, 2].

사물 인터넷에 연결된 각종 기기들이 수행하는 기본적인 작업은 다양한 센서를 이용하여 주변 환경 정보 혹은 기기의 상태 정보를 획득하고 이를 근접한 게이트웨이 혹은 원격 서버로 전송하는 것이다. 여기에는 사물 인터넷 환경에 적합하게 만들어진 사물 인터넷 응용 프로토콜이 주로 사용된다. MQTT는 발행/구독(publish/subscribe) 기반의 경량 메시지 프로토콜로서 TCP/IP 상에서 동작하는 응용 계층 프로토콜이다 [3, 4]. 현재 ISO 표준이며, 사물 인터넷 통신을 위한 프로토콜로 주목받고 있다. MQTT는 다양한 사물 인터넷 플랫폼 및 서비스에서 지원하고 있다. 대표적인 예로 국내에서 개발한 oneM2M 표준 기반의 사물 인터넷 플랫폼인 Mobius, 아마존의 사물 인터넷 플랫폼인 AWS IoT, IBM의 Watson IoT Platform, 그리고 클라우드 기반의 분산 MQTT 브로커 서비스인 CloudMQTT 등에서 MQTT 프로토콜 기반의 통신을 지원하고 있다.

사물 인터넷 기기는 보통 배터리 파워로 동작하는 경우가 많고 배터리 교체 없이 장시간 동작할 수 있어야 한다. 따라서 센서 데이터 획득 및 전송 작업을 처리하는 데 있어서 에너지 효율성이 매우 중요한 문제 중 하나이다. 배터리로 동작하지 않고 외부 전원을 공급받는 기기라고 하더라도 향후 사물 인터넷 기기의 수가 기하급수적으로 증가하게 되면, 에너지 소모량은 그에 비례하여 증가할 것이다. 따라서 이 경우에도 에너지 효율성 문제는 여전히 중요한 문제라 하겠다. 에너지 효율적인 사물 인터넷 애플리케이션 혹은 시스템을 만들기 위해서는 해당 애플리케이션이나 시스템의 전력 소모 특성에 대한 정보가 기본적으로 필요하다. 기존 연구에서 다양한 스마트폰이나 웨어러블 기기의 전력 소모를 측정하고 그 특성을 분석하는 일들이 많았지만 [5-10], 사물 인터넷 기기에서 사물 인터넷 통신 시 전력 소모

를 분석하는 연구는 많이 이루어지지 않았다.

본 논문에서는 사물 인터넷 하드웨어 플랫폼으로 널리 사용되는 라즈베리 파이에서 대표적인 사물 인터넷 응용 프로토콜인 MQTT 사용 시 전력 소모를 측정하고 분석한다. 이를 위해 MQTT를 이용하여 데이터를 전송하고 수신하는 다양한 상황과 데이터 송수신을 증계하는 상황을 고려하였다. 본 논문의 결과는 기존 연구에서 별로 다루지 않았던 여러 가지 MQTT 프로토콜 사용 시나리오에 따른 에너지 소모를 측정하고 그에 따른 결과를 보여주는 것으로서 의의가 있다.

본 논문의 구성은 다음과 같다. 2장에서 관련 연구에 대해서 논의한다. 3장에서는 라즈베리 파이에서 MQTT 이용 시 전력 소모 측정을 위한 실험 방법을 기술하고, 4장에서 그 결과를 분석하였다. 끝으로 5장에서 한계점 및 향후 연구 이슈를 기술하면서 논문을 맺는다.

## II. 관련 연구

본 논문의 관련 연구로 크게 두 가지로 나누어 살펴 보도록 하겠다. 첫째, 모바일 및 사물 인터넷 디바이스의 전력 소모 분석에 대한 연구에 대해 기술한다. 둘째, 사물 인터넷에서 사용되는 대표적인 응용 프로토콜을 소개하고 기존 인터넷 응용 프로토콜과 비교하는 연구를 소개한다.

### 2.1. 모바일 및 사물 인터넷 기기 전력 소모 분석 연구

스마트폰이 대중화되고 많이 이용되면서 스마트폰의 배터리 동작 시간이 중요한 이슈가 됨에 따라 스마트폰의 배터리 소모에 대한 측정과 분석, 이를 바탕으로 한 배터리 효율성 향상 방법에 대한 연구가 많이 이루어져왔다 [5-7]. 최근에는 다양한 웨어러블 기기와 사물 인터넷 하드웨어 플랫폼들이 나오면서 이 기기들의 에너지 소모를 분석하는 논문들도 발표되고 있다. 대표적으로 안드로이드 스마트워치의 에너지 소모를 측정하고 하드웨어 컴포넌트 별로 에너지 소모를 분석한 연구가 있었다 [8]. 그리고 구글 클래스의 에너지 소모를 다양한 사용 시나리오에 대해서 측정하여 분석하고 발열 특성을 분석한 연구가 있었다 [9]. 또한 모바일 및 사물 인터넷 기기를 대상으로 하는 Snapdragon 800, Tegra K1, Intel Edison에서 딥러닝 모델을 실행하는데

필요한 CPU time, 메모리, 에너지 등의 자원 소모를 측정하고 분석한 연구도 있었다 [10].

본 논문에서는 이전 연구와 달리 사물 인터넷 하드웨어 플랫폼으로 널리 사용되는 라즈베리 파이의 에너지 소모를 분석하는 것을 목표로 한다. 에너지 소모 분석 대상 워크로드로는 대표적인 사물 인터넷 프로토콜인 MQTT를 이용하여 데이터를 전송/수신하는 다양한 시나리오를 적용하였다. 여러 가지 MQTT 프로토콜 사용 시나리오에 따른 에너지 소모를 측정하고 그에 따른 결과를 보여주는 것으로 의의가 있다.

## 2.2. 사물 인터넷 프로토콜 관련 연구

사물 인터넷에서는 기존 인터넷과 달리 제한된 컴퓨팅 자원과 안정적이지 않은 네트워크 연결 환경에서도 소형 기기들이 센싱을 하고 로컬 네트워크의 게이트웨이나 원격 네트워크의 서버와 통신을 하며 데이터를 전송을 할 수 있어야 한다. 이와 같은 환경을 고려하여 제안된 프로토콜로서 MQTT [3], CoAP [11] 등이 있다. MQTT는 국제 표준화 단체인 OASIS (Organization for the Advancement of Structured Information Standards)에 의해서 사물 인터넷을 위한 OASIS 표준 프로토콜로 지정되어 있으며, 2016년 현재, ISO 표준(ISO/IEC 20922)이 되었다. CoAP은 인터넷 표준 단체인 IETF (Internet Engineering Task Force)에서 다양한 기기가 인터넷에 연결될 수 있도록 만든 표준규약으로서 기존의 HTTP 웹 프로토콜에 대응되는 프로토콜이다. MQTT와 CoAP은 세계 각국의 표준화 기구가 참여하여 제안하고 있는 표준 IoT 플랫폼인 oneM2M 플랫폼에서도 메시지 전송을 위한 애플리케이션 계층 프로토콜로 포함되어 있다 [12, 13].

사물인터넷 기술에 대한 연구가 활발히 이루어지면서, HTTP와 같은 기존 인터넷 응용 프로토콜과 비교하여 사물 인터넷 프로토콜의 효율성을 보여주는 연구들도 수행되고 있다. 효율적인 센서 데이터 수집을 위해 MQTT 프로토콜을 기반으로 하는 데이터 수집 솔루션을 제안하는 논문이 최근 발표되었다 [14]. 이 논문에서는 실험을 통해 HTTP를 사용하는 방법과 비교하여, MQTT를 사용하는 방법이 초당 데이터 처리량, 응답 시간 등의 측면에서 매우 우수한 성능을 보임을 밝혔다. 또한 OpenMTC 플랫폼을 기반으로 M2M 데이터를 전송하는데 HTTP와 CoAP 프로토콜의 성능 차이를 비교

한 논문도 최근 발표되었다 [15].

본 논문에서는 앞서 언급한 기존 연구와 달리 사물 인터넷 프로토콜인 MQTT 사용 시 에너지 소모를 분석하는데 초점을 둔다. 이를 위해 대표적인 사물 인터넷 하드웨어 플랫폼인 라즈베리 파이를 대상으로 하였다.

## III. MQTT 전력 소모 측정 실험 방법

이 장에서는 MQTT 프로토콜의 다양한 사용 시나리오에 따른 전력 소모 측정을 위해서 필요한 하드웨어 및 소프트웨어에 대해 설명하고 측정 방법에 대해 기술한다. 여기서 MQTT 프로토콜의 사용 시나리오는 크게 두 가지로 구분하였다. 첫째, MQTT 클라이언트로 동작하는 상황이다. 여기에는 MQTT 메시지를 발생하여 브로커로 전송하는 발행(publish) 클라이언트와 브로커에 접속하여 관심 있는 메시지를 요청하고 이를 수신하는 구독(subscribe) 클라이언트가 있다. 둘째, MQTT 브로커로 동작하는 상황이다. MQTT 브로커는 MQTT 발행 클라이언트로부터 메시지를 받아 MQTT 구독 클라이언트로 메시지를 전송하는 역할을 한다. 이 실험에서는 MQTT 메시지 송수신에 따른 전력 소모를 분석하는데 초점을 두므로 MQTT 클라이언트에서는 메시지 생성 및 송수신 이외의 다른 연산은 하지 않는다.

### 3.1. 하드웨어 셋업 및 전력 소모 측정 방법

실험을 위해 필요한 하드웨어 셋업은 사물 인터넷 기기와 PC, 그리고 이를 연결하기 위한 네트워크가 있으며, 이와 더불어 전력 소모 측정을 위한 측정 장비가 있다. 그림 1은 실험에 사용된 전력 소모 측정을 위한 하드웨어 셋업을 보여준다. 그리고 그림 2는 파워 측정을 위해 사용된 셋업 환경을 도식화하여 보여준다.

MQTT 클라이언트와 브로커를 실행하는 사물 인터넷 기기로 저가의 싱글보드 컴퓨터로 전 세계적으로 널리 사용되고 있는 라즈베리 파이를 사용하였다. 현재 라즈베리 파이는 라즈베리 파이 3 모델 B까지 출시되어있으며, 본 실험에서는 라즈베리 파이 2 모델 B를 사용하였다. 라즈베리 파이를 구동하기 위한 운영체제로는 라즈베리 파이 재단에서 제공하는 공식 운영체제인 라즈비안(Raspbian)을 사용하였다. 버전은 Raspbian GNU/Linux 8이며 커널 버전은 4.4이다.

소모 전력 측정을 위한 장비는 Monsoon Power Monitor를 사용하였다. Monsoon Power Monitor는 리튬 배터리로 동작하는 모바일 기기의 소모 전력을 측정하는 용도로 널리 사용되고 있다. 그림 1에서 우측의 흰색 장치가 Monsoon Power Monitor이다. 전력 소모 측정 대상이 되는 라즈베리 파이가 Monsoon Power Monitor에 전원 공급용 케이블로 연결이 되어 있다. 그리고 Power Monitor는 노트북 PC에 USB 케이블로 연결이 되어 있어 측정 데이터가 PC로 전송된다.

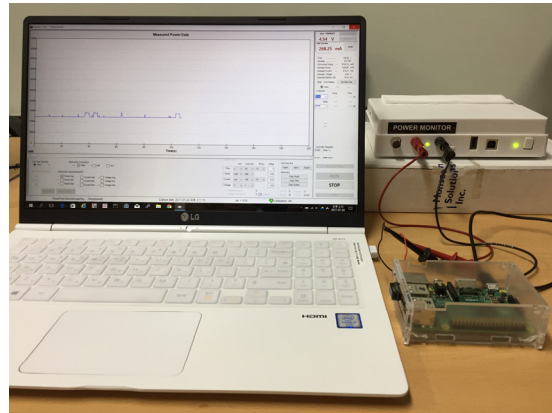


Fig. 1 H/W setup for power measurement.

그림 2에 표시된 것처럼 Monsoon Power Monitor를 통해 전력 소모를 측정하고자 하는 라즈베리 파이에 일정 전압이 인가되고, 소모 전류와 전력이 측정된다. 측정된 데이터는 USB로 연결된 PC에서 실행되는 전용 소프트웨어인 Power Tool 프로그램에서 실시간으로 그 수치가 표시되고 그래프로 가시화 된다. 또한 측정 결과 데이터는 파일로 저장할 수 있다. 현재 Power Monitor 장비에서 지원 가능한 인가 전압에 제한이 있어, 측정 실험 시 라즈베리 파이에 인가된 전압은 최대 값인 4.54V로 설정되었다.

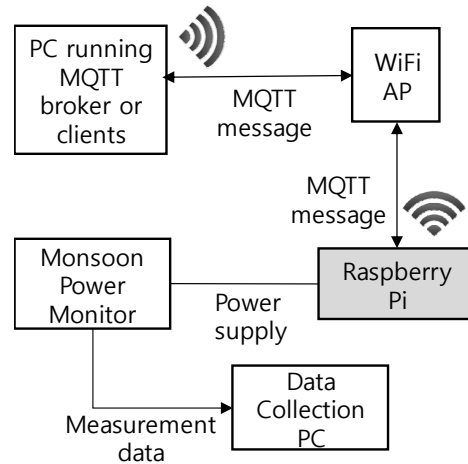


Fig. 2 Diagram of power measurement setup.

라즈베리 파이에서는 MQTT 클라이언트 혹은 브로커를 실행할 때 소모 전력을 측정하게 된다. 이 때, 라즈베리 파이와 MQTT 메시지를 교환하기 위한 MQTT 브로커 혹은 MQTT 클라이언트가 동작하고 있어야 한다. 이를 위해 MQTT 브로커 혹은 클라이언트를 실행 하는 별도의 PC를 두었다. 그리고 이 PC와 라즈베리 파이는 와이파이(WiFi) AP(Access Point)를 통해서 연결된다. 예를 들어, 라즈베리 파이에서 발행 클라이언트가 동작 하는 경우 이 PC에서는 MQTT 브로커가 실행되고 있어서 클라이언트로부터 발행 메시지를 수신한다. 라즈베리 파이에서 MQTT 브로커가 동작하는 경우에는 PC에서 발행 클라이언트와 구독 클라이언트가 실행되고 있어서 서로 간에 메시지가 송수신된다. 여기서 사용한 PC는 Intel Core i7-6500U CPU와 8GB 램을 탑재하고 Windows 10 운영체제로 동작한다. 그리고 라즈베리 파이 2 모델 B는 이더넷만 기본 제공하고 있기 때문에 IEEE 802.11n을 지원하는 USB 와이파이 어댑터를 사용하여 AP에 연결하였다.

### 3.2. MQTT 브로커 및 클라이언트 소프트웨어

MQTT 브로커로는 오픈소스 MQTT 브로커로 널리 사용되고 있는 Mosquitto를 사용하였다 [16]. 사용 버전

은 1.4.10이며 현재 MQTT 버전 3.1/3.1.1을 지원하고 있다. 실험에 사용된 라즈베리 파이에 Mosquitto를 설치하고 파워 소모 측정 시에 이를 실행하였다. Mosquitto 브로커 실행 시 설정은 기본 설정을 그대로 사용하였다.

MQTT 클라이언트는 오픈소스 MQTT 클라이언트를 제공하는 이클립스 재단의 Paho 프로젝트를 기반으로 구현하였다. Paho 프로젝트는 현재 자바, 파이썬, C, 자바스크립트 등 다양한 언어의 구현을 지원하고 있다 [17]. 본 실험에서는 파이썬으로 구현된 MQTT 클라이언트 라이브러리를 사용하여 발행 및 구독 클라이언트를 구현하였다.

발행 클라이언트는 실행 시 발행 주기, 메시지 크기, QoS 값을 입력 받을 수 있도록 하여 다양한 워크로드 조건으로 실행되는 상황에서 전력 소모를 측정할 수 있게 만들었다. 구독 클라이언트 실행 시에는 구독 토픽 개수를 입력 받을 수 있도록 만들었다. 구독 토픽 개수는 1개에서 4개까지 지원되도록 하였다.

### 3.3. 워크로드

다음으로 MQTT 프로토콜 사용 시 소모 전력 측정을 위한 워크로드에 대해서 기술한다. MQTT 메시지 전송을 위한 기본 설정은 다음과 같다. 기본 메시지 크기는 2 바이트(Byte)로 하였다. 라즈베리 파이에서 별도의 센싱 작업은 수행하지 않고 더미(dummy) 데이터를 생성하여 이용하였다. MQTT 프로토콜에서는 메시지 전달의 신뢰성을 위한 QoS를 지원하며 보안을 위한 TLS를 지원한다. QoS는 0, 1, 2 세 가지 값을 가질 수 있다. 본 실험에서는 기본 옵션으로 QoS는 0을 사용하였으며, TLS는 사용하지 않는 것으로 했다. QoS 파라미터를 1 혹은 2로 설정하였을 때의 영향을 관찰하기 위한 실험은 추가로 수행하였다.

MQTT 프로토콜은 발행/구독 기반의 메시지 송수신 프로토콜로서 브로커와 클라이언트로 동작이 구분되는데, 본 실험에서는 발행 클라이언트, 구독 클라이언트, 브로커의 세 가지 경우에 대해 측정을 하였다. 발행 클라이언트의 전력 소모 측정을 위해서 사용한 워크로드 설정을 위한 파라미터로는 메시지 발행 주기, 메시지 크기, QoS 값을 적용하였다. 구독 클라이언트를 위한 워크로드는 메시지 주기, 메시지 크기, 구독 토픽 개수를 파라미터로 적용하였다. 브로커를 위한 워크로드는 메시지 발행 수, 메시지 구독 수를 조정하였다.

소모 전력은 위에서 기술한 워크로드에서 한 가지 조건을 설정하고 1회에 5분 동안의 평균 소모 전력을 측정하는 방식으로 구하였다. 그리고 동일 조건에서 3회 측정 후 평균값을 구하여 소수점 첫째자리에서 반올림한 값을 보고한다.

## IV. 실험 결과

### 4.1. 기본 전력 소모

기본 전력 소모는 실험용 라즈베리 파이에 전원이 공

급되어 라즈비안 OS가 명령줄 인터페이스 모드로 구동되고 OS 기본 프로세스 실행 이외에 아무런 작업도 하지 않는 상태에서 측정하였다. 앞에서 언급한 것처럼 라즈베리 파이 2의 경우 와이파이 모듈이 내장되어 있지 않아 USB 와이파이 어댑터를 사용하여 와이파이 AP에 연결되어 있다. 기본 전력 소모는 와이파이 어댑터를 장착하지 않은 경우와 장착한 경우를 나누어서 측정하였다.

와이파이 어댑터가 없는 경우 평균 소모 전력은 961 mW이고, 와이파이 어댑터를 장착한 경우 평균 소모 전력은 1193 mW로 나타났다. USB 와이파이 어댑터를 사용하는 경우 약 230 mW의 전력이 더 소모되는 것을 알 수 있다. 기본 소모 전력 1193 mW의 경우 4.54 V에서 소모 전류로 환산하면 약 263 mA이다. 이는 5000 mAh 용량의 배터리를 사용한다고 했을 때, 완충 후 19 시간을 사용할 수 있는 정도의 전력 소모량이다. 현재 라즈베리 파이는 저전력 모드를 지원하지 않고 있어 상대적으로 큰 소모 전력을 보인다. 이는 애초 라즈베리 파이가 모바일 환경에서 사용되는 것을 목적으로 디자인 된 것이 아니기 때문에 그런 것으로 보인다.

### 4.2. MQTT 발행 클라이언트 전력 소모

다양한 상황에서 MQTT 프로토콜의 메시지 발행 시 전력 소모를 분석하기 위하여 메시지 발행 주기, 메시지 크기, QoS 값을 바꾸어 가면서 소모 전력을 측정하였다. 첫 번째 실험은 메시지 발행 주기에 따른 전력 소모 결과를 보여준다. 발행 주기는 1초에서 5초까지 1초 간격으로 증가하였다. 나머지 조건은 앞서 언급한 기본 설정을 적용하였다. 즉, 메시지 크기는 2 바이트, QoS 값은 0, TLS 미적용 상태이다.

그림 3은 측정 결과를 보여준다. 예상할 수 있듯이 메시지 발행 주기가 가장 짧은 1초일 때 소모 전력이 약 1254 mW로 가장 크다. 기본 소모 전력 대비 약 60 mW를 더 소모하는 것을 알 수 있다. 주기가 3초가 되면 1207 mW로 크게 감소하고, 5초일 때 1203 mW가 되었다. 기본 소모 전력보다 약 10 mW 정도 더 소모하는 것을 알 수 있다.

두 번째 실험은 발행 메시지 크기에 따른 소모 전력의 변화를 알아본다. 메시지 크기는 최소 2 바이트에서 10, 100, 1000, 3000, 5000, 7000 바이트까지 증가하였다. MQTT 프로토콜을 이용하여 단순한 센서 데이터를

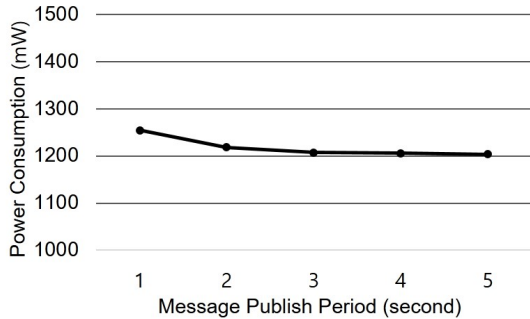


Fig. 3 Power consumption depending on the message publish period.

전송하는 데는 수 바이트에서 수십 바이트의 메시지 크기로 충분한 경우가 있을 것이고, 더 큰 용량의 데이터를 전송하는 고려하여 수 킬로바이트 크기의 메시지 발행 상황을 고려하였다. 여기서 발행 주기는 1초이고, 나머지 조건은 앞의 실험과 같다.

그림 4는 측정 결과를 보여준다. 메시지 크기가 1000 바이트 이하일 경우 평균 소모 전력이 약 1242-1254 mW 정도로 큰 차이를 보이지 않았다. 3000 바이트일 때 1314 mW로 약 60 mW가 증가하고, 5000 바이트일 때 1385 mW로 약 70 mW 증가하여 포화상태에 다다른 것을 알 수 있다. 7000 바이트일 때 약 1386 mW로 차이가 거의 없었다. MQTT는 TCP를 기반으로 동작하는 프로토콜이다. TCP의 최대 전송 단위(Maximum Transfer Unit: MTU)가 있는데, 실험에서 사용한 라즈비안 OS에서 MTU는 1500 바이트로 설정되어 있다. 따라서 3000 바이트 메시지부터는 복수개의 TCP 세그먼트로 나뉘어서 전송이 되어야 하므로 전력 소모가 증가

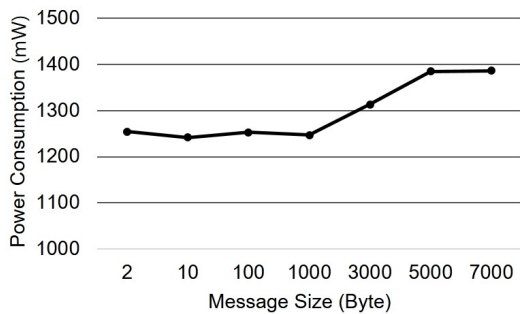


Fig. 4 Power consumption depending on the published message size.

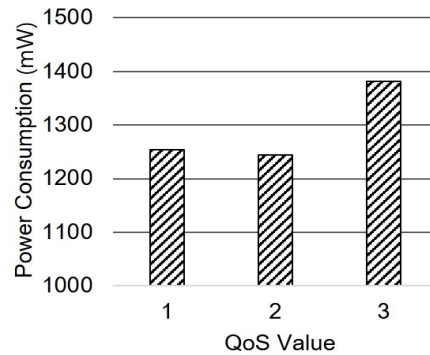


Fig. 5 Power consumption depending on the QoS value of publish message.

한 것으로 볼 수 있다.

세 번째 실험은 QoS 값에 따른 전력 소모의 변화를 측정한다. MQTT 프로토콜에서는 메시지 전송의 QoS 지원을 위해서 메시지 발행 및 구독 시 원하는 QoS 값을 지정할 수 있다. 기본값은 0으로 설정되어 있으며, 이는 메시지를 ‘많아야 한 번(at most once)’ 전달하는 것으로 메시지가 유실될 수도 있는 경우이다. QoS 1의 경우는 메시지를 ‘적어도 한 번(at least once)’ 전달하는 것으로 메시지가 유실되는 일이 없이 전달이 보장되지만 메시지 중복 전달이 발생할 수 있다. QoS 2의 경우는 메시지를 ‘정확히 한 번(exactly once)’ 전달되도록 한다. MQTT 발행 클라이언트가 QoS 0으로 메시지를 발행할 경우, 즉 PUBLISH 메시지를 전송할 경우 브로커는 이에 대해 응답하지 않는다. QoS 1인 경우, 브로커는 PUBACK 메시지로 메시지를 받았다는 응답을 하게 된다. QoS 2의 경우 메시지 중복 전달을 방지하기 위하여 두 번의 확인 과정을 거치게 된다. 클라이언트가 PUBLISH 메시지를 전송하면, 브로커는 PUBREC 메시지로 응답을 하고, 클라이언트는 다시 PUBREL 메시지를 전송한다. 브로커는 PUBCOMP 메시지로 응답을 하여 완료가 된다. 따라서 QoS 값이 2인 경우 부가적인 메시지 송수신 확인 과정으로 인해 전력 소모가 증가하게 된다.

측정 결과는 그림 5에서 볼 수 있다. QoS 값이 1인 경우 QoS 0인 경우와 별 차이가 없었고, 2인 경우 1382 mW로 크게 증가함을 알 수 있었다. 이는 기본 소모 전력에 비하여 190 mW 정도 더 소모하는 것으로서, 5000

mAh 배터리를 사용하는 상황에서 사용 시간이 약 2.6 시간 감소하는 결과를 가져오는 것이다.

### 4.3. MQTT 구독 클라이언트 전력 소모

MQTT 메시지 구독의 경우 메시지 발행 주기, 메시지 크기, 구독 토픽의 개수를 변경해 가면서 소모 전력을 측정하였다. 소모 전력 측정용 라즈베리 파이에서 구독 클라이언트를 실행하고, 구독 클라이언트에 메시지가 전달되게 하기 위해서 별도의 라즈베리 파이에서 발행 클라이언트를 실행하였다.

첫 번째 실험은 구독 클라이언트가 수신하는 메시지의 주기에 따른 전력 소모 결과를 보여준다. 이를 위해 발행 클라이언트에서 메시지 발행 주기를 1초에서 5초까지 1초 간격으로 증가하였다.

결과는 그림 6에서 확인할 수 있다. 1초일 때 평균 소모 전력이 1255 mW이고, 5초 일 때 1216 mW로 감소한다. 앞의 발행 클라이언트의 경우와 비교하여 2-5초일 때 7-10 mW 정도 더 소모하지만 큰 차이는 없음을 알 수 있다.

두 번째 실험은 구독 클라이언트가 수신하는 메시지 크기에 따른 소모 전력의 변화를 알아본다. 메시지 크기는 최소 2 바이트에서 10, 100, 1000, 3000, 5000 바이트까지 증가하였다. 여기서 메시지 발행 주기는 1초이고, 나머지 조건은 앞의 실험과 같다.

그림 7은 측정 결과를 보여준다. 메시지 크기가 1000 바이트 이하일 경우 평균 소모 전력이 약 1255-1263 mW 정도로 큰 차이를 보이지 않았다. 이는 앞의 발행 클라이언트의 전력 소모 결과와 비슷한 추세를 보이는 것이다. 소모 전력은 약 10 mW 정도 증가한 것을 알 수

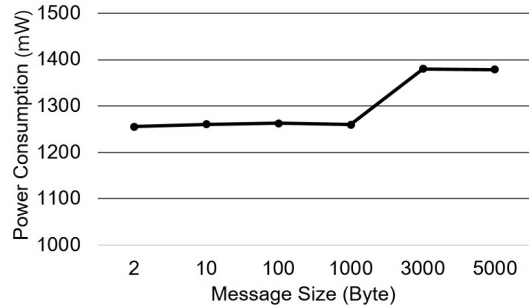


Fig. 7 Power consumption depending on the subscribed message size.

있다. 그런데 발행 클라이언트의 경우와 달리 3000 바이트일 때 이미 1380 mW로 5000 바이트일 때와 비슷하게 포화상태에 다다른 것을 알 수 있다.

세 번째 실험은 구독 클라이언트가 구독하는 메시지의 토픽 개수에 따른 소모 전력의 변화를 알아본다. 구독하는 토픽의 개수가 많아지면 그만큼 많은 메시지를 수신하게 되므로 전력 소모가 증가하게 된다. 이 실험에서는 메시지 주기를 1초와 5초 두 경우를 두고, 토픽의 개수를 2-4개로 하여 측정하였다.

측정 결과는 그림 8에서 볼 수 있다. 왼쪽 4개의 바는 메시지 주기가 1초인 경우, 오른쪽 4개의 바는 주기가 5초인 경우 결과이다. 주기가 1초인 경우 구독하는 토픽 개수가 2개만 되어도 전력 소모가 급격히 증가하여 3, 4개인 경우와 큰 차이가 없음을 알 수 있다. 약 1380 mW의 전력 소모를 보인다. 반면 주기가 5초인 경우는 토픽 개수가 증가하면 소모 전력도 증가하긴 하지만 주기가 1초인 경우에 비해 증가 폭이 상대적으로 작다. 토픽 개

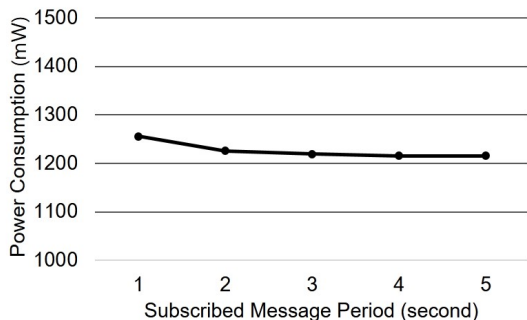


Fig. 6 Power consumption depending on the subscribed message period.

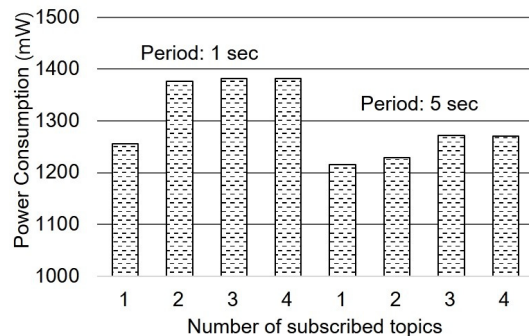


Fig. 8 Power consumption depending on the number of subscribed message topics.

수가 3, 4개인 경우 약 1270 mW의 전력 소모를 보인다.

#### 4.4. MQTT 브로커 전력 소모

MQTT 브로커는 앞서 언급한 것처럼 MQTT 발행 클라이언트로부터 메시지를 받아 MQTT 구독 클라이언트로 메시지를 전송하는 역할을 한다. MQTT 브로커 동작 시 소모 전력은 발행 클라이언트에서 전송 받는 메시지의 워크로드 혹은 구독 클라이언트에게 메시지를 전달해주어야 하는 워크로드에 영향을 받는다. 따라서 브로커의 전력 소모 측정 실험을 위하여 발행 메시지 수와 구독 메시지 수를 변경해 가면서 소모 전력을 측정하였다. 그 수는 1-3 사이의 값으로 하였다. 또한 발행 메시지의 주기를 1초로 한 경우와 5초로 한 경우 두 가지에 대해서 소모 전력을 측정하였다.

측정 결과는 그림 9에서 볼 수 있다. 맨 좌측의 1 pub 1 sub의 경우 하나의 발행 클라이언트가 하나의 토픽에 대해서 메시지를 발행하고, 이를 하나의 구독 클라이언트가 전달받는 경우이다. 1 pub 2 sub의 경우는 하나의 발행 클라이언트가 발행하는 메시지를 두 개의 구독 클라이언트가 동일하게 수신 받는 경우이다. 2 pub 1 sub의 경우는 두 개의 발행 클라이언트가 각각 다른 토픽에 대해서 메시지를 발행하고, 하나의 구독 클라이언트에서 2개의 토픽에 대해서 구독 요청을 하여 동시에 메시지를 전달 받는 경우이다.

소모 전력은 브로커가 처리해야 하는 워크로드가 증가하면 커지는데, 메시지 주기가 1초인 경우와 5초인 경우 그 양상이 다르게 나타났다. 전자에서는 발행 클라이언트 하나, 구독 클라이언트 하나인 경우에 이미

포화상태에 이르러 발행 혹은 구독 메시지 수를 더 증가하여도 소모 전력에 별다른 차이를 보이지 않았다. 약 1390 mW 정도의 전력을 소모하는 것으로 나왔다. 반면 메시지 주기가 5초인 경우 1 pub 1 sub 조건에서 1221 mW의 평균 소모 전력을 보였고, 워크로드가 증가하면 소모 전력도 증가하여 3 pub 1 sub 조건에서 1299 mW가 된다.

끝으로 QoS 설정에 의한 차이를 보기 위하여 메시지 주기 5초, 1 pub 1 sub 조건에서 QoS 값을 2로 설정하였을 때 소모 전력을 측정하였다. QoS 0인 경우 대비 67 mW가 증가하여 소모 전력은 1288 mW로 나왔다. 앞에서 기술하였듯이 QoS 2인 경우 부가적인 메시지 송수신으로 발행 및 구독 클라이언트 수가 각각 1개인 경우에도 QoS 값이 0이면서 발행/구독 클라이언트 수가 복수 개인 경우와 비슷한 전력 소모를 보임을 알 수 있다.

## V. 결 론

본 논문에서는 MQTT 프로토콜 사용의 전력 소모를 분석하였다. 이를 위해 사물 인터넷 애플리케이션 및 시스템 개발에 활용되는 라즈베리 파이를 이용하였다. MQTT 프로토콜 사용 예를 메시지 발행 및 구독 클라이언트 그리고 브로커로 동작하는 경우로 나누어서 메시지 주기, 크기, QoS 등을 변경하면서 전력 소모를 관찰하였다. 기본 전력 소모는 약 1190 mW를 보였고, 본 논문에서 사용한 실험 조건에서 최대 약 1390 mW까지 소모함을 알 수 있었다. 기존 연구에서 다루지 않았던 다양한 MQTT 프로토콜 사용 상황에 따른 전력 소모를 측정하고 그 결과를 보여주는 것으로서 의의가 있다.

본 논문에서 보고한 결과에는 몇 가지 한계점이 있으며 그에 따른 추가적인 연구 이슈도 있다. 첫째, 이 논문에서 대상으로 한 사물 인터넷 디바이스는 라즈베리 파이 2로서 이 논문에서 기술한 결과는 해당 하드웨어를 사용하는 경우 유효할 것이다. 현재 사물 인터넷 서비스 개발을 지원하는 다양한 하드웨어 플랫폼이 나와 있는데, 대표적으로 인텔 Edison, 삼성 ARTIK 등이 있고, 라즈베리 파이도 버전 3이 출시되었다. 이러한 여러 가지 다른 하드웨어 상에서의 전력 소모에 대한 향후 연구도 필요할 것이다.

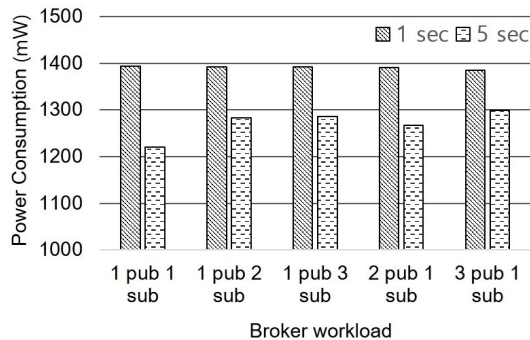


Fig. 9 Power consumption depending on the number of published and subscribed messages handled by the broker.



둘째, 본 논문에서는 MQTT 메시지 송수신에 소모되는 전력을 분석하는 것이 목적이었기 때문에 그 이외 다른 워크로드는 없다고 가정하고 측정하였다. 사물 인터넷 서비스 실행 시 센서 데이터 획득, 데이터 처리 등 추가적이고 복잡한 연산 과정이 들어간다면 전력 소모는 달라질 것이다. 이와 같은 종합적인 워크로드 상황에서 전력 소모 분석이 향후 더 필요할 것이다[18].

셋째, 본 논문에서는 MQTT 메시지 발행 및 구독, 그리고 브로커 동작 시 전력 소모를 관찰하기 위해 파이썬으로 된 Paho 클라이언트 라이브러리를 사용하였고, 브로커는 mosquitto를 사용하였다. Paho 클라이언트도 파이썬 이외에 자바, 자바스크립트, C 등 여러 언어를 지원하고 있으며, Paho 외에도 다양한 MQTT 클라이언트가 존재한다. 또한 MQTT 브로커도 다양하다. 이런 다양한 환경에서 차이를 추가적으로 살펴볼 수도 있을 것이다.

넷째, MQTT 프로토콜 사용 시나리오에서 실험에서 다루지 않은 좀 더 복잡하고 다양한 상황에 대한 추가 실험도 가능할 것이다. 예를 들면, 메시지 전송 시 TLS를 적용한다거나, 토픽 수를 다르게 한다거나 복수 개의 토픽에 대해서 메시지를 발행/구독한다거나 하는 등의 상황을 고려해 볼 수 있을 것이다.

### ACKNOWLEDGMENTS

This work was supported in part by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP: Ministry of Science, ICT & Future Planning) (No. 2017R1C1B1010619). This paper was also partially supported by the Education and Research Promotion Program of KOREATECH (2016).

### REFERENCES

[ 1 ] C. P. Kruger, A. M. Abu-Mahfouz, G. P. Hancke, "Rapid prototyping of a wireless sensor network gateway for the internet of things using off-the-shelf components," in *Proceedings of 2015 IEEE International Conference on*

*Industrial Technology (ICIT)*, pp. 1926-1931, 2015.

[ 2 ] P. H. Kulkarni, P. D. Kute, V. N. More, "IoT based data processing for automated industrial meter reader using Raspberry Pi," in *Proceedings of the International Conference on Internet of Things and Applications (IOTA)*, pp. 107-111, 2016.

[ 3 ] MQTT Version 3.1.1 Protocol Specification. [Internet]. Available: <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html>.

[ 4 ] S. Myung, S. Kim, "The design of Open IoT Platform based on oneM2M Standard Protocol," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 21, no. 10, pp. 1943-1949, Oct. 2017.

[ 5 ] M. Dong, T. Lan, L. Zhong, "Rethink energy accounting with cooperative game theory," in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pp. 531-542, 2014.

[ 6 ] A. Pathak, Y. C. Hu, M. Zhang, P. Bahl, Y. M. Wang, "Fine-grained power modeling for smartphones using system call tracing," in *Proceedings of the 6th European Conference on Computer Systems (EuroSys)*, pp. 153-168, 2011.

[ 7 ] A. Pathak, Y. C. Hu, M. Zhang, "Where is the energy spent inside my app?: fine grained energy accounting on smartphones with eprof," in *Proceedings of the 7th European Conference on Computer Systems (EuroSys)*, pp. 29-42, 2012.

[ 8 ] X. Liu, F. Qian, "Poster: Measuring and Optimizing Android Smartwatch Energy Consumption," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom)*, pp. 421-423, 2016.

[ 9 ] R. LiKamWa, et al, "Draining our Glass: An Energy and Heat Characterization of Google Glass," in *Proceedings of 5th Asia-Pacific Workshop on Systems (APSys)*, Article No. 10, 2014.

[10] N. D. Lane, et al, "An Early Resource Characterization of Deep Learning on Wearables, Smartphones and Internet-of-Things Devices," in *Proceedings of the 2015 International Workshop on Internet of Things towards Applications (IoT-App)*, pp. 7-12, 2015.

[11] CoAP specification (RFC 7252). [Internet]. Available: <http://tools.ietf.org/html/rfc7252>.

[12] oneM2M, MQTT Protocol Binding (TS-0010-V2.4.1), 2016. [Internet]. Available: [http://www.onem2m.org/images/files/deliverables/Release2/TS-0010-MQTT%20Protocol%20Binding-V2\\_4\\_1.pdf](http://www.onem2m.org/images/files/deliverables/Release2/TS-0010-MQTT%20Protocol%20Binding-V2_4_1.pdf).

- [13] oneM2M, CoAP Protocol Binding (TS-0008-V1.3.2), 2016. [Internet]. Available: [http://www.onem2m.org/images/files/deliverables/TS-0008-CoAP\\_Protocol\\_Binding-V1\\_3\\_2.pdf](http://www.onem2m.org/images/files/deliverables/TS-0008-CoAP_Protocol_Binding-V1_3_2.pdf).
- [14] S. Kim, D. Kim, H. Oh, H. Jeon, H. Park, "The Data Collection Solution Based on MQTT for Stable IoT Platforms," *Journal of the Korea Institute of Information and Communication Engineering* vol. 20, no. 4, pp. 728-738, Apr. 2016.
- [15] C. Berrhouma, et al, "Performance Evaluation of an M2M Platform in Different Deployment Setups," in *Proceedings of 2016 IEEE First International Conference on Internet-of-Things Design and Implementation*. pp.223-228. 2016.
- [16] Mosquitto. [Internet]. Available: <https://mosquitto.org/>.
- [17] MQTT Paho Project. [Internet]. Available: <https://github.com/eclipse/paho.mqtt.python>.
- [18] K. Park, J. Lee, M. Woo, J. Park, "Development of an IoT System Based on the oneM2M Communication Protocol," *Asia-pacific Journal of Multimedia Services Convergent with Art, Humanities, and Sociology*, vol.6, no.3, pp. 41-49, Mar. 2016.



강승우(Seungwoo Kang)

2010년 한국과학기술원 전산학과 공학박사  
2015년-현재: 한국기술교육대학교 컴퓨터공학부 조교수  
※관심분야 : IoT/모바일 시스템, 모바일/웨어러블/유비쿼터스 컴퓨팅