

<http://dx.doi.org/10.7236/IIBC.2015.15.6.123>

IIBC 2015-6-17

IETF CoAP 기반의 데이터 정제 기법을 이용한 IoT 미들웨어 설계 및 구현

Design and Implementation of IoT Middleware Using Data Refinement Scheme based on IETF CoAP

왕건*, 사공준*, 곽호영**, 김도현***

Jian Wang*, June SaKong*, Ho-Young Kwak**, DoHyun Kim***

요 약 최근 IoT(Internet of Things)에서 다양한 사물을 인터넷에 연결하여 정보를 교환하는 통신 프로토콜에 대한 관심이 집중되고 있다. 현재 IETF(Internet Engineering Task Force) 표준화 기구에서는 제한된 환경에서 센서나 구동체와 같은 사물 간의 통신을 지원하는 CoAP(Constrained Application Protocol) 프로토콜을 표준으로 채택하고 있다. 향후 IoT 환경에서 서버뿐만 아니라 스마트 폰에서 센서로 부터 데이터를 수집하거나, 구동체에게 명령을 전달하고, 사물을 관리하는 역할을 담당할 것으로 예상된다. 본 논문에서는 IETF CoAP 을 기반으로 스마트 폰과 IoT 노드 사이의 연결 구성을 설계하고, 스마트 폰에서 데이터 정제 기능을 갖는 미들웨어를 설계하고 구현한다. 제안된 데이터 정제 기법을 이용하여 송수신 과정에서 발생하는 오류 데이터와 중복 데이터를 확인하고 제거함으로써 제한된 외부 환경에서 신뢰성 있는 정보를 전달하는 데 기여할 것이다.

Abstract Recently, a communication protocol for connectivity between various sensors and actuators is issue the key technology in IoT(Internet of Things). IETF(Internet Engineering Task Force) CoAP(Constrained Application Protocol) protocol supports the communication between things like sensors and actuator nodes in a limited environment. Smart-phone will work for the data processing of the sensors and the management of IoT nodes at the next generation communication environment. In this paper, we design the connection for smart-phone and IoT node based using IETF CoAP protocol, and implement the middleware with the proposed data refinement on smart-phone. The middleware detects and eliminates a error data and duplicated data using the proposed data refinement. The proposed data refinement scheme supports to transfer a reliability information in the constrained environment.

Key Words : IETR CoAP, Middleware, IoT, Data refinement

I. 서 론

최근 휴대용 스마트 단말기의 높은 보급과 더불어 스마트 폰에 내장된 근거리 통신기술(Bluetooth, WiFi,

WiFi-Direct, NFC 등)을 활용한 다양한 IoT(Internet of Things) 서비스에 대한 관심이 증가하고 있다. 그러나 아직 스마트 폰을 이용하여 주위 장치나 IoT 서비스를 발견하고 제공하는 기술이 부족한 실정이다. 더불어 스마

*준회원, 제주대학교 컴퓨터공학과

**정회원, 제주대학교 컴퓨터공학과

***정회원, 제주대학교 컴퓨터공학과(교신저자)

접수일자: 2015년 11월 13일, 수정완료: 2015년 12월 7일

게재확정일자: 2015년 12월 11일

Received: 13 November, 2015 / Revised: 7 December, 2015 /

Accepted: 11 December, 2015

***Corresponding Author: kimdh@jejunu.ac.kr

Dept. of Computer Engineering, Jeju National University, Korea

트 폰에서 IoT 통신 프로토콜을 이용하여 사용자가 직접 IoT 서비스를 검색하고 해당 응용을 사용하는 서비스도 미흡하다^{[1][2]}.

최근 IoT 응용 통신 프로토콜로 HTTP(Hypertext Transfer Protocol), CoAP(Constrained Application Protocol), MQTT(Message Queuing Telemetry Transport)등에 대한 표준화가 진행되고 있다. 이들 중 IETF CoRE(Constrained RESTful environments) 워킹 그룹에서 표준화하고 있는 CoAP 프로토콜은 작은 용량의 메모리와 저전력 등 제한된 환경에서 센서나 구동체 노드 간의 통신을 지원한다^[3]. IETF CoAP 프로토콜은 UDP 기반의 응용 통신 프로토콜이며, 작은 자원을 가진 IoT 노드에서 인터넷을 통해 데이터 교환을 최적화하고 있다. CoAP은 IoT 표준 프로토콜로 널리 채택될 것이지만 현재 CoAP 기반의 IoT 노드와 스마트 단말기 간 통신 및 미들웨어에 관한 연구는 미비한 실정이다^[4].

본 논문에서는 IETF CoAP을 기반으로 스마트 폰과 IoT 노드 사이의 연결 구성 및 데이터 정제를 이용한 스마트 폰 미들웨어를 설계하고 구현한다. 스마트 폰 상에서 CoAP 클라이언트 기반의 미들웨어를, CoAP 서버 기반의 IoT 노드를 설계하고, 이를 안드로이드와 리눅스 기반으로 구현하여 상호 호환성을 검증한다. 여기서 열악한 환경으로 발생하는 오류 데이터를 자동으로 처리하는 데이터 정제 기법을 제안한다. 제시한 데이터 정제 기법은 중복된 센서 데이터도 자동적으로 제거할 수 있다. 중복 데이터를 제거하기 위해 데이터의 세그먼트를 분할하고, 중복된 영역을 제거하여 유일한 고유블록을 한번만 저장하는 백업기술을 이용한다. 그리고 3배수 표준편차 값을 이용하여 오류 데이터를 제거한다.

본 논문은 서론에 이어 2장에서는 IETF CoAP 프로토콜과 IoT 미들웨어 등 관련연구를 서술한다. 3장에서는 IETF CoAP 기반의 스마트 폰과 IoT 노드 사이의 연결 구성, 스마트 폰 상에 데이터 정제 기법을 이용한 미들웨어를 설계하고 구현한다. 그리고 4장에서는 IoT 노드와 스마트 폰 미들웨어에 대해 실험하고 검증한 결과를 보여주고, 마지막으로 5장에서는 결론과 향후 연구 계획에 대하여 기술한다.

II. 관련 연구

1. CoAP과 IoT 미들웨어

IETF 워킹그룹에서 2010년 초부터 본격적으로 개발되기 시작한 CoAP 프로토콜은 여러 차례의 변화를 거쳐 2014년 RFC 7252를 제정했다. CoAP은 UDP를 기반으로 하며, HTTP를 통하여 웹 서버 및 데이터베이스로 구성된 서버에게 데이터 작업을 요청하고, 그 결과를 받는 RESTful API 구조의 데이터 모델까지 제공할 수 있다.^[5] CoAP은 응용계층과 UDP계층 사이에 Request/Response 계층, Message 계층을 구현하고 있다.

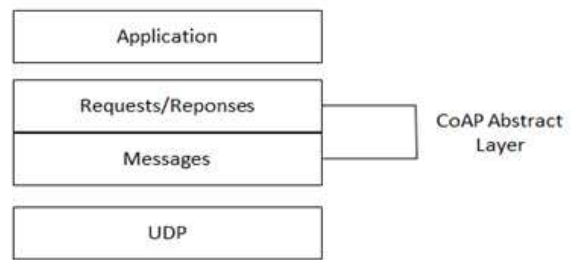


그림 1. CoAP 프로토콜 구조
Fig. 1. CoAP protocol stack

IoT 미들웨어는 센서 네트워크 미들웨어에서 발전하고 있으며, 장치 관리, 상호 동작, 플랫폼 이식성, 상황인식, 보안 및 정보보호 기능에 따라 여러 유형으로 구분할 수 있다.^[2] 그러나 이들 대부분의 IoT 미들웨어는 서버, 네트워크 노드, IoT 노드 중심으로 개발되고 있으나, 아직 스마트 폰에서의 CoAP을 이용한 미들웨어 연구는 미흡하다.

III. CoAP 기반의 데이터 정제 미들웨어

1. CoAP 기반의 데이터 정제 미들웨어 설계

그림 2는 CoAP 기반의 IoT노드와 스마트폰 기반 미들웨어 연결 기본 구성이다. 여기서 IEEE 802계층은 LAN, PAN, MAN 표준을 이용할 수 있으며 UDP/IP 를 이용하여 데이터 전송을 수행한다. CoAP 서버는 CoAP 라이브러리를 사용하여 스마트 폰과 통신하기 위한 서버이며 CoAP 클라이언트는 CoAP을 이용하여 IoT 노드로부터 센싱 데이터를 받기 위한 응용프로그램이다. 미들웨어는 수집된 데이터의 처리를 위해 CoAP 클라이언트를 포함하고 있으며, 중복 데이터와 에러 데이터 처리 기능을 제공한다.

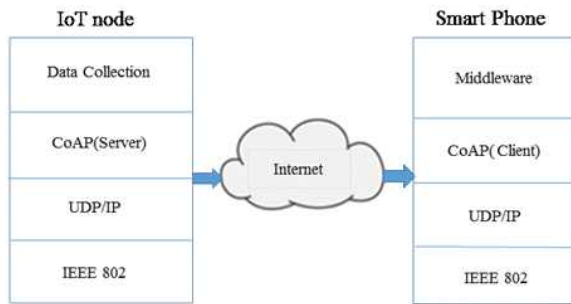


그림 2. CoAP 기반의 IoT 노드와 스마트 폰 연결 구성
 Fig. 2. Connection configuration of IoT node and Smart-phone based on CoAP

CoAP 기반의 스마트 폰과 IoT 노드 간 통신을 위해 CoAP 서버, 미들웨어, CoAP 클라이언트, 데이터 처리, 사용자 인터페이스를 설계한다.

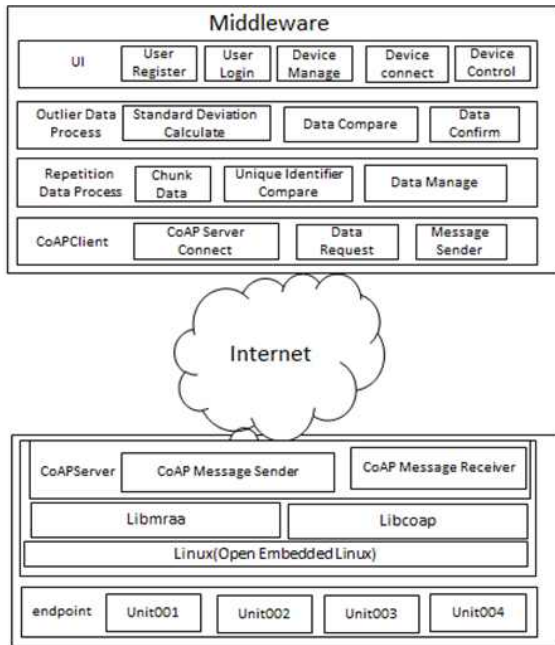


그림 3. CoAP 기반의 데이터 정제 미들웨어 구조
 Fig. 3. The middleware architecture with data refinement based on CoAP

그림 3에서 IoT 노드는 리눅스 기반으로 CoAP 서버를 설계, 구성한다. 여기서 CoAP endpoint는 CoAP 메시지가 도착되어야 하는 최종 목표를 의미한다. Libcoap은 IoT노드 상의 디바이스와 CoAP Server를 연결하기 위한 라이브러리이며, Libmraa는 자바 스크립트 및 파이썬을 바인딩하고, IoT 노드 및 기타 플랫폼의 IO를 연결하는 라이브러리이다. 그리고 CoAP 서버는 CoAP 클라

이언트와의 데이터 송수신을 위한 CoAP Message Sender, CoAP Message Receiver로 구성한다.

스마트 폰의 미들웨어는 데이터 정제 기법을 이용하고 있으며, CoAP 클라이언트 기반으로 설계된다. 여기서 CoAP 클라이언트는 CoAP 서버와의 연결, CoAP 서버로부터 데이터 수신, CoAP 서버로 메시지 전송 등의 역할을 수행한다. 더불어 Repetition Data Process는 원시 데이터를 오류 데이터와 정상 데이터로 구분하고 유일한 블록을 통해 중복 데이터를 비교한다. 오류 데이터 정제와 중복 데이터 제거는 Outlier DataProcess가 담당한다. 사용자 인터페이스(UI : User Interface)는 사용자 생성, 사용자 등록, IoT 노드 상의 디바이스를 스마트폰 미들웨어에 추가, 디바이스를 선택하여 디바이스 측에 데이터 수집 및 제어하는 역할을 제공한다.

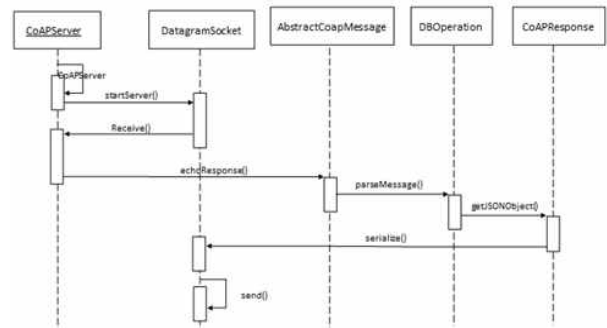


그림 4. CoAP 서버 시퀀스 다이어그램
 Fig. 4. The CoAP server sequence diagram

2. CoAP 기반의 데이터 처리

스마트 폰의 COAP 서버에서는 URI QUERY를 통하여 노드를 식별하며, CoAP 요청을 하여 데이터베이스에서 해당 IoT 노드 정보를 조회한다. IoT 노드의 CoAP 클라이언트는 노드의 ID와 Token 값, 데이터가 포함된 정보를 Json 포맷으로 전송한다.

제한한 데이터 정제 알고리즘은 원시 데이터를 유일한 식별자를 통하여 블록으로 식별한다. 다음으로 저장소에 유일한 블록이 존재하는지 확인하고, 없으면 추후 중복을 검사하기 위해 저장소에 등록한다. 그리고 오류 데이터를 판별하기 위해 3배수 표준편차를 계산한 후 3 배수 표준편차와 수신한 데이터를 비교한다. 마지막으로 조건을 만족하면 사용자 인터페이스를 통해 도시하고, 아니면 데이터를 제거한다. 그림 5에서는 데이터 정제 흐름도를 보여주고 있다.

중복 데이터 처리는 저장소에 유일한 블록 또는 오브젝트의 존재 여부를 확인하여 동일한 데이터가 있으면 중복 데이터를 제거하고, 없으면 사용자에게 전달한다.

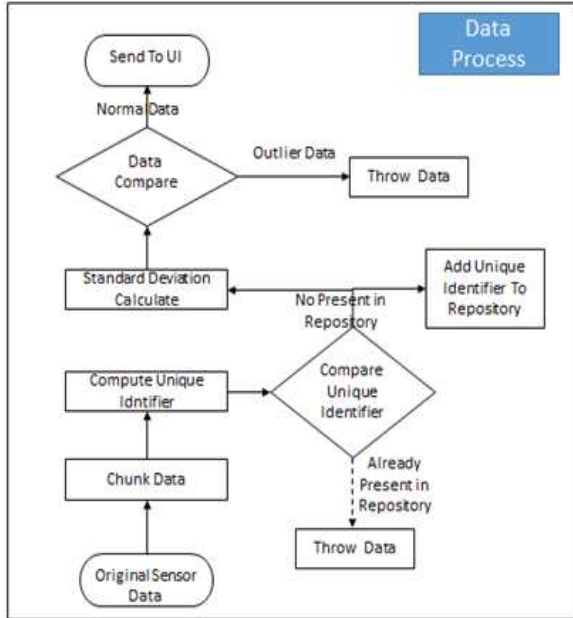


그림 5. 데이터 정제 흐름도
Fig. 5. Flowchart for data refinement

3. CoAP 기반의 IoT 노드와 스마트 폰의 미들웨어 구현

CoAP 기반의 IoT 노드와 스마트 폰의 미들웨어를 구현하기 위해 Yocto Linux 1.6, JDK1.7, Eclipse juno, MySQL database, Android 4.1 platform, Libmraa, gcc 등의 개발 환경을 이용한다.

그리고 IoT 노드는 인텔 에디슨(Edison) 보드, sensor node는 TINKERKIT 등의 하드웨어를 사용한다. IoT 노드의 CoAP 서버를 구현하기 위하여 Libcoap 라이브러리를 이용한다. 그림 6에서는 인텔 에디슨 보드와 TINKERKIT 기반의 IoT 노드를 보여주고 있다.

더불어 스마트 폰 하드웨어는 VEGA IM-A850L 이고 운영체제는 안드로이드 4.1이다. CoAP 클라이언트를 구현하기 위하여 Californium 프레임워크^[7]를 이용하며 스마트폰에서 오류 데이터를 정제 처리하기 위하여 자바 프로그래밍 언어를 이용하여 미들웨어를 구현한다.

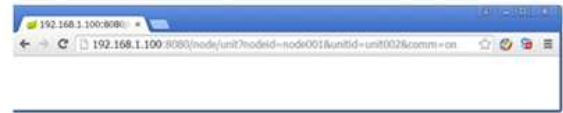


그림 6. 인텔 에디슨 보드와 TINKERKIT 기반의 IoT 노드
Fig. 6. IoT node with Intel Edison board and TINKERKIT

스마트 폰의 CoAP 클라이언트는 Respond클래스, Handler 클래스, Request 클래스, CoAPClient 클래스 등으로 구성한다. 그리고 데이터 정제를 위한 Outlier Data 프로세스는 StandardDeviationCalculate 클래스, DataCompare 클래스, DataConfirm 클래스로 이루어져 있다. StandardDeviationCalculate 클래스는 표준편차 계산 기능을 제공하며 DataCompare 클래스는 3배수 표준편차와 수신한 데이터를 비교하는 기능을 제공하고 DataConfirm 클래스는 데이터 제거 및 전송 기능을 수행한다. CoAPClient 클래스를 통하여 각 클래스의 객체생성 함수를 호출한다. 그림 7에서는 데이터 정제를 위한 클래스 다이어그램을 보여주고 있다.

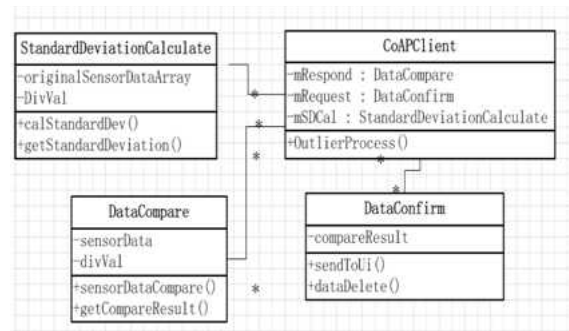


그림 7. 데이터 정제를 위한 클래스 다이어그램
Fig. 7. Class diagram for data refinement

사용자 인터페이스는 MainWindow 클래스, UserRegister 클래스, UserLogin 클래스, DeviceConnecter 클래스, DeviceManager 클래스, DeviceControler 클래스를 이용하여 구현한다.

IV. 실험 및 결과

CoAP 기반의 IoT 노드와 스마트폰의 미들웨어를 구현하고, 오류 데이터에 대한 정제를 실험한다. 이를 위해 IoT 노드로부터 조도 데이터를 수신하여 스마트폰 상에서 오류 데이터를 처리하고 그 결과를 보여준다. 표 1은 조도 센서를 이용하여 수집된 데이터이다. 왼쪽의 표는 처리하기 전의 데이터이며 오른쪽은 처리 후의 데이터이다. 표기 ①, ②를 기준으로 한 데이터 집합에서 표준편차의 값은 15.5로 계산되었다. 3배 표준 편차는 46.5이며 표기 ①, ②에서 데이터 집합 평균값은 296 이다. 데이터의 값과 평균값의 차가 3배 표준편차보다 더 크므로 이상한 센서 값으로 정한다.

표 1. 조도 센서 데이터
 Table 1. Light sensor data

Light Original Data ^{a)}			Light Processing Data ^{a)}		
2015/6/10	10:07:23 ^{a)}	306 ^{a)}	2015/6/10	10:07:23 ^{a)}	306 ^{a)}
2015/6/10	10:07:24 ^{a)}	310 ^{a)}	2015/6/10	10:07:24 ^{a)}	310 ^{a)}
2015/6/10	10:07:25 ^{a)}	312 ^{a)}	2015/6/10	10:07:25 ^{a)}	312 ^{a)}
2015/6/10	10:07:26 ^{a)}	313 ^{a)}	2015/6/10	10:07:26 ^{a)}	313 ^{a)}
2015/6/10	10:07:27 ^{a)}	317 ^{a)}	2015/6/10	10:07:27 ^{a)}	317 ^{a)}
2015/6/10	10:07:28 ^{a)}	321 ^{a)}	2015/6/10	10:07:28 ^{a)}	321 ^{a)}
2015/6/10	10:07:29 ^{a)}	323 ^{a)}	2015/6/10	10:07:29 ^{a)}	323 ^{a)}
2015/6/10	10:07:31 ^{a)}	327 ^{a)}	2015/6/10	10:07:31 ^{a)}	327 ^{a)}
2015/6/10	10:07:33 ^{a)}	322 ^{a)}	2015/6/10	10:07:33 ^{a)}	322 ^{a)}
2015/6/10	10:07:34 ^{a)}	324 ^{a)}	2015/6/10	10:07:34 ^{a)}	324 ^{a)}
2015/6/10	10:07:35 ^{a)}	326 ^{a)}	2015/6/10	10:07:35 ^{a)}	326 ^{a)}
2015/6/10	10:07:36 ^{a)}	66 ^{a)}	2015/6/10	10:07:36 ^{a)}	66 ^{a)}
2015/6/10	10:07:37 ^{a)}	329 ^{a)}	2015/6/10	10:07:37 ^{a)}	329 ^{a)}
2015/6/10	10:07:38 ^{a)}	332 ^{a)}	2015/6/10	10:07:38 ^{a)}	332 ^{a)}
2015/6/10	10:07:39 ^{a)}	335 ^{a)}	2015/6/10	10:07:39 ^{a)}	335 ^{a)}
2015/6/10	10:07:40 ^{a)}	78 ^{a)}	2015/6/10	10:07:40 ^{a)}	78 ^{a)}
2015/6/10	10:07:41 ^{a)}	335 ^{a)}	2015/6/10	10:07:41 ^{a)}	335 ^{a)}
2015/6/10	10:07:42 ^{a)}	337 ^{a)}	2015/6/10	10:07:42 ^{a)}	337 ^{a)}
2015/6/10	10:07:43 ^{a)}	337 ^{a)}	2015/6/10	10:07:43 ^{a)}	337 ^{a)}
2015/6/10	10:07:44 ^{a)}	336 ^{a)}	2015/6/10	10:07:44 ^{a)}	336 ^{a)}
2015/6/10	10:07:45 ^{a)}	335 ^{a)}	2015/6/10	10:07:45 ^{a)}	335 ^{a)}
2015/6/10	10:07:46 ^{a)}	336 ^{a)}	2015/6/10	10:07:46 ^{a)}	336 ^{a)}
2015/6/10	10:07:47 ^{a)}	335 ^{a)}	2015/6/10	10:07:47 ^{a)}	335 ^{a)}
2015/6/10	10:07:48 ^{a)}	65 ^{a)}	2015/6/10	10:07:48 ^{a)}	65 ^{a)}

그림 8에서 표기된 ①, ②, ③, ④는 이상한 데이터이다. 데이터 정제 기법을 이용하여 오류 데이터 및 중복 데이터를 처리한 결과를 그림 9에서 보여주고 있다.

V. 결론

본 논문에서는 IETF CoAP 프로토콜을 기반으로 인텔 에디슨 IoT 노드 사이의 통신을 위하여 안드로이드 스마트폰 상에 미들웨어를 설계하고 구현한다. 더불어 중복데이터와 오류 데이터를 제거하는 데이터 정제 기법을 제안하고, 스마트폰 상에 CoAP 클라이언트 기반으로 미들웨어를 구현하고 동작을 확인하였다. 향후 CoAP 프로토콜을 기반으로 주변의 다양한 센싱 정보를 수집, 전

송하여 스마트폰 미들웨어에서 다양한 상황을 분석해서 지능적인 정보를 제공하는 연구할 예정이다.

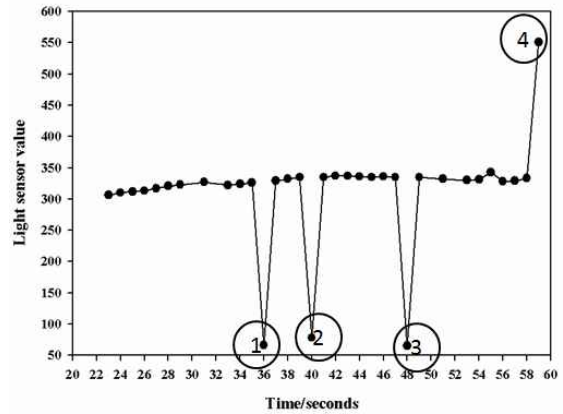


그림 8. 조도 센서 원시 데이터 그래프
 Fig 8. Light sensor raw data before refinement

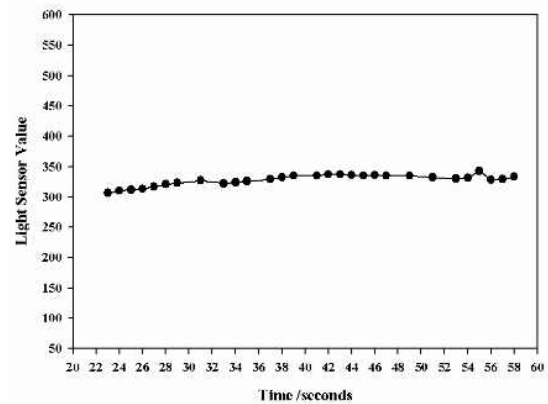


그림 9. 정제 후의 조도 데이터 그래프
 Fig 9. Light sensor data after refinement

References

- [1] Perera, C; Zaslavsky, A; Christen, P; Georgakopoulos, D., "Context Aware Computing for The Internet of Things: A Survey", Communications Surveys & Tutorials, IEEE 2014, Vol. 16, Issue 1, 2014
- [2] S. Bandyopadhyay, M. Sengupta, S. Maiti, and S. Dutta, "Role of middleware for internet of things," International Journal of Computer Science and Engineering Survey, Vol. 2, 2011.
- [3] Wen-Quan JIN, Do-Hyeun Kim "Implementation and Experiment of CoAP Protocol Based on IoT for Verification of Interoperability", The Journal of

The Institute of Internet, Broadcasting and Communication (JIIBC), Vol. 14 No. 4, August 2014

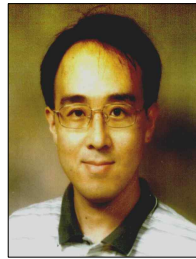
- [4] CoAP protocol, http://en.wikipedia.org/wiki/Constrained_Application_Protocol
- [5] Shelby, Z, Hartke K. and Bormann C., "The Constrained Application Protocol(CoAP)," IETF RFC 7252, June 2014.
- [6] Chen Nam, Do-Hyeun Kim, "A Study of Indoor Context Monitoring Service Based on Comfort Index", The Journal of the Institute of Internet, Broadcasting and Communication (JIIBC), Vol. 14 No. 1, 2014
- [7] Californium CoAP framework, <https://github.com/mkovatsc/Californium-inf.ethz.ch>, 2014

곽 호 영(정회원)



- 1999년 9월 ~ 현재 : 제주대학교 공과대학 컴퓨터공학과 교수
- 2014년 3월 ~ 현재 : 한국컴퓨터정보학회 부회장
- <주관심분야 : 시스템 소프트웨어, IoT, RFID/USN>

김 도 현(정회원)



- 2004년 9월 ~ 현재 : 제주대학교 공과대학 컴퓨터공학과 교수
- 2013년 1월 ~ 현재 : 대한전자공학회 M2M/IoT 연구회 회장
- <주관심분야 : , IoT/M2M, 임베디드 시스템, 모바일컴퓨팅>

저자 소개

왕 건(준회원)



- 2013년 9월 ~ 2015년 8월 : 제주대학교 석사
- <주관심분야 : 시스템 소프트웨어, 네트워크 프로토콜, IoT>

사 공 준(준회원)



- 1976년 3월 ~ 1983년 8월 : 경북대학교 학사
- 1993년 3월 ~ 1995년 8월 : 경북대학교 석사
- 1996년 3월 ~ 1998년 2월 : 경북대학교 박사수료
- <주관심분야 : 임베디드시스템, IoT>

※ 본 논문은 2015년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원(No.10043907, 개방형 고성능 표준 IoT 디바이스 및 지능형 SW 개발)과 미래창조과학부 및 정보통신기술진흥센터의 대학ICT연구센터육성 지원사업의 연구결과로 수행되었음" (IITP-2015-H8501-15-1017). 교신저자 : 김도현