

스마트폰용 USB 액세서리 Plug and Play를 위한 통합 앱 구조

강선구¹ · 채의근[†] · 은성배²

(원고접수일 : 2014년 4월 17일, 원고수정일 : 2014년 5월 17일, 심사완료일 : 2014년 5월 23일)

Unified app architecture for plug and play of smart phone accessory

Sun Goo Kang¹ · Yi Geun Chae[†] · Sung Bae Eun²

요약: 현재 많이 보급되어 보편적인 이동단말장치로 사용되는 스마트폰은 USB 등의 인터페이스를 지원하며 휴대성과 이동성이 편해 기존 모니터링 장비의 대체가 가능하다. 최근 방사능, 나트륨, 전자파 등의 주변 환경을 탐지하는 센서를 USB 액세서리 형태의 입력장치로 개발되어 많이 판매되고 있으나, 기존 방식의 센서 Plug and Play 기술을 스마트폰 USB 액세서리 장치를 연결하는데 문제점이 있다. 본 논문에서는 스마트폰 USB 액세서리 Plug and Play를 위한 통합 앱 구조를 제안하고, 통합 앱과 센서 액세서리 플랫폼 그리고 센서 웹서버와의 데이터교환 방식을 설계하였으며 다양한 센서들의 측정 데이터를 하나의 앱에서 수집하고 표시할 수 있도록 구현하였다.

주제어: 스마트폰, 센서디바이스, 환경 모니터링, Plug and Play, 통합 앱

Abstract: Recently, smart phones are spread out as a universal mobile terminal equipment. Its provide USB interface to connect with various devices. Smart phones may be replaced with high priced monitoring equipments because of portability and mobility as its merit. In addition to that, the various sensor devices detecting surrounding environment of the radioactivity, sodium or electromagnetic waves have been announced. But the plug and play methods of sensor devices have some problem to connect smart phone with USB accessory device. We propose an integrated methodology that can connect smart phone with USB sensor devices and also, we realized USB accessory plug and play with one App that can collect measurement data through various sensor devices.

Keywords: Smart phone, sensor device, environment monitoring, Plug and Play, integrated App.

1. 서 론

2007년 A사의 스마트폰이 출시된 후 이제는 언제 어디서든 스마트폰을 사용하고 있다. 스마트폰이 없으면 불편함이 느껴지는 시대가 된 것이다 [1]. 2008년에는 스마트폰용 안드로이드 OS가 발표되었으며 특히 안드로이드 OS는 오픈소스로 배포하며 수많은 제조회사들이 스마트폰을 공급하게

되었다.

스마트폰이 대중화되고 다양한 연결 장치들이 출시되면서 스마트폰을 위한 액세서리 시장이 크게 성장하고 있다. 2009년 265억 달러였던 전 세계 시장규모가 2015년에는 500억 달러 규모로 성장할 것으로 예상된다.

다양한 스마트폰용 센서 액세서리가 개발되고

[†] Corresponding Author: Department of Computer Engineering, College of Engineering, Kongju National University, Seobuk-Gu Cheonan-Daero 1223-24, Cheonan, Chungnam, 331-717, Korea, E-mail: ygchae@kongju.ac.kr Tel: 041-521-9233

¹ Department of Information and Communication Engineering, College of Engineering, Hannam University, 133 Ojeong-dong, Dedeok-Gu, Daejeon, 306-791, Korea

³ Department of Information and Communication Engineering, College of Engineering, Hannam University, 133 Ojeong-dong, Dedeok-Gu, Daejeon, 306-791, Korea, E-mail: sbeun@hannam.ac.kr, Tel: 042-629-7928

보급될수록 산업현장에서 더욱 더 활용될 것이다. 예를 들면 많은 산업현장에는 운습도 센서 같은 값 비싼 계측기가 상시 설치되어 있다. 센서 자체의 가격보다는 설치 장치로써 세트 구성되어 판매되고 있으며 가격이 고가이다.

이러한 고가의 계측기를 센서 액세서리를 장착한 스마트폰이 이를 대신한다면 측정이 용이하고 데이터의 전송과 보관 그리고 축적된 데이터의 활용측면에서 편리할 것이다.

또 다른 활용 측면에서는 U-Health에 적용할 수 있다. 맥박이나 혈압, 당뇨 등의 소형 센서를 개발하고 이를 이동성이 좋고 휴대성이 편한 스마트폰에 센서 액세서리로써 적용한다면 장소와 시간의 제약이 없으며 보다 용이하게 U-Health 기기를 사용할 수 있다. 또한 선박이나 플랜트 혹은 생산시설과 같이 큰 공간에서 이동성이 높은 측정 장비를 이용하여 측정값을 확보하고 이를 Wi-Fi나 인터넷을 이용하여 집적할 수 있다면 상당히 편리할 것이다[2][3].

2. 기존 센서 Plug and Play 기술과 문제점

2.1 센서 추상화와 Plug and Play 기술

Figure 1에서 보는 바와 같이 센서 Plug and Play를 지원하는 전체 시스템은 센서 디바이스 드라이버 관리와 센서노드에서의 처리 부분으로 나누어진다. 센서 디바이스 드라이버 관리는 센서 디바이스 드라이버 검색 프로토콜과 디렉토리 서비스 프로토콜 그리고 새로운 드라이버 등록 프로토콜 등으로 이루어진다[4]-[7].

센서노드에서의 처리는 센서 식별 정보 수신, 센서 디바이스 드라이버 비교 프로토콜 그리고 센서 디바이스 드라이버 요청 절차 등이다. 이러한 센서노드 플랫폼을 구현하기 위해서는 센서 투명성을 지원하는 센서노드용 운영체제의 구조와 센서 투명성을 지원하는 센서 디바이스 매니저 등의 역할이 중요하다[4].

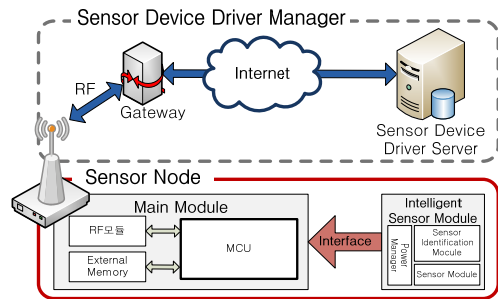


Figure 1: Plug and Play system structure

2.2 기존 Plug and Play 기술의 한계점

센서 Plug and Play를 지원하는 시스템은 일반적인 센서들을 다루기 위한 기술이기 때문에 스마트폰과 연결하기 위한 USB Host 기능이 없다. 또한 통신에 필요한 안드로이드 액세서리 프로토콜이 구현되어 있지 않기 때문에 통신이 불가능하다. 본 연구에서는 이러한 문제점을 해결하는 방안을 제시하고 그 가능성을 간단한 구현으로 실현됨을 확인할 수 있었다[7].

3. 통합 앱 구조 설계

3.1 통합 앱 시스템 구조

Figure 2는 USB센서 Plug and Play를 지원하는 스마트폰 통합 앱의 전체 시스템이다. 구성은 센서 액세서리 플랫폼과 센서 웹서버 그리고 통합 앱으로 나누어진다.

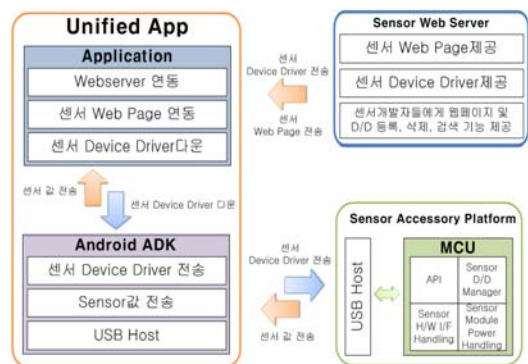


Figure 2: System structure of integrated App

센서 액세서리 플랫폼은 센서를 제어하기 위해 개발된 별도의 센서용 보드로써 USB Host, API, Sensor D/D Manager, sensor H/W I/F Handling, Sensor Module Power Handling 기능으로 이루어진 다.

센서 웹서버는 센서모듈의 디바이스 드라이버와 웹페이지를 제공한다. 통합 앱에서 디바이스 드라이버나 웹페이지다운 요청이 있을시 다운로드 해주는 기능을 제공한다. 웹서버는 센서 개발자에게 업데이트되는 디바이스 드라이버를 웹페이지 등록하고 삭제하고 검색을 할 수 있는 기능을 제공한다.

통합 앱은 스마트폰에 탑재되어 ADK를 통해 센서 액세서리 플랫폼과 통신을 한다. 그리고 웹서버에서 디바이스 드라이버 및 센서 웹페이지 다운의 기능을 제공한다. 센서의 값을 받아서 웹페이지와 연동하여 사용자에게 보여주고, 디바이스 드라이버를 센서 액세서리 플랫폼에 다운로드 해준다.

3.2 통합 앱과 센서 액세서리 플랫폼

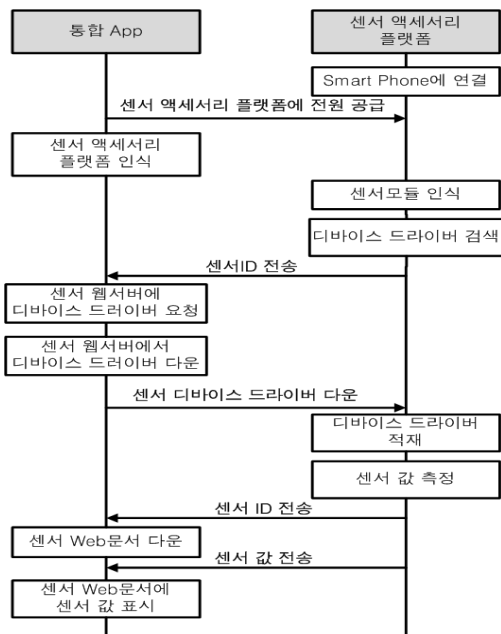


Figure 3: Procedure of detecting sensor accessory platform

Figure 3은 통합 앱에 센서 액세서리 플랫폼이 연결 되었을 때 인식 및 디바이스 드라이버의 다운로드 및 센서 값 전송에 관한 처리 과정이다. 통합 앱은 사용자에게 센서 액세서리 플랫폼의 연결을 알리고 사용자가 확인하면 연결을 승인한다. 센서 모듈이 디바이스 드라이버의 유무를 확인하고 없을 경우 통합 앱으로 센서 ID를 전송한다.

통합 앱은 요청한 센서ID를 통해 웹서버에서 해당 디바이스 드라이버를 다운로드해 플랫폼으로 전송한다. 플랫폼은 디바이스 드라이버를 적재하고 센서와 통신해 측정된 값을 통합 앱으로 전송한다. 통합 앱은 해당 센서의 웹페이지 유무를 확인 후 없을 경우 웹서버에 요청하여 다운로드한다.

3.3 통합 앱과 센서 웹서버

Figure 4는 통합 앱이 센서 웹서버에서 센서의 디바이스 드라이버 및 웹페이지 다운요청의 처리 과정을 보여준다. 플랫폼에서 디바이스 드라이버의 요청 시 통합 앱은 웹서버에 센서ID를 전송한다. 센서ID를 통해 검색하고 결과를 통해 통합 앱에서는 디바이스 드라이버를 다운로드한다.

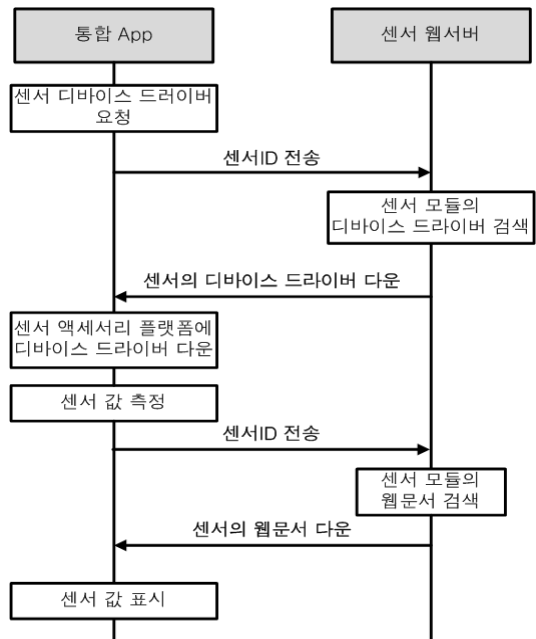


Figure 4: Procedure of download request to server

플랫폼은 통합 앱으로부터 전송받은 디바이스 드라이버를 사용해 센서모듈을 인식하고 정보를 측정한다. 센서모듈이 최초로 연결 시에만 다운로드 하기 때문에 재측정 때는 보다 빠르게 정보를 표시 할 수 있다.

4. 통합 앱 구현

4.1 구현환경

통합 앱을 구현하기 위해서 안드로이드 2.4.1부터 적용되는 ADK (Android Open Accessory Development Kit) API를 활용하였다. 안드로이드 ADK를 사용하는 이유는 안드로이드폰 혹은 태블릿에서 사용되는 USB 포트와의 연결을 보다 용이하게 개발할 수 있기 때문이다[6][9].

마이크로칩의 ADK 보드를 구성하는 소프트웨어 구조는 Figure 5와 같다. ADK 보드를 운영하고 관리하는 부분과 USB 호스트 기능을 관리하는 부분 그리고 안드로이드 폰과의 연결 및 인식 관리하는 부분으로 구성되어 있다[10][11].



Figure 5: ADK internal structure

하드웨어 추상화 레이어는 ADK 보드의 하드웨어를 초기화하고 관리한다. USB 호스트 스택은 USB 호스트 기능을 담당하는 소프트웨어로 ADK 보드에서 인식하도록 한다. 안드로이드 클라이언트 드라이버는 ADK 보드에서 장치를 인식하기 위한 드라이버이다[9][10].

4.2 센서 액세서리 플랫폼

본 연구에서는 이미 개발하여 계속 적용중인 ATmega128에 USB통신을 할 수 있는 MAX3421칩

을 연결하여 제작하였다[5][6]. 이 센서노드 플랫폼의 구조는 Figure 6에서 보는 바와 같다[6].

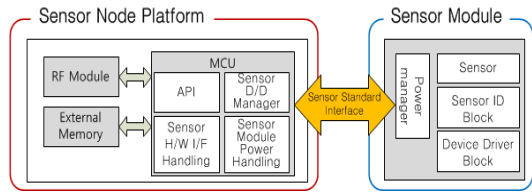


Figure 6: System structure of sensor platform

USB HOST 기능을 탑재한 칩인 MAX3421을 사용하고 구글 ADK API에서 제공하는 안드로이드 액세서리 프로토콜을 바탕으로 구현하였다[8][10]. Sensor Standard H/W Interface는 센서 Plug and Play 액세서리 플랫폼과 식별정보를 주고받고 센서 특성에 따라 ADC, I2C, SPI, Interrupt 등 다양한 인터페이스를 제공하여 Plug and Play가 가능하도록 해준다.

4.3 통합 앱

통합 앱은 Figure 7과 같이 메인시스템과 파일 다운로드부분으로 구성된다. 메인 시스템은 안드로이드에서 동작하는 일반적인 애플리케이션으로 통합 앱의 메인을 담당한다. 웹페이지는 HTML5와 CSS3, JQuery로 제작되었다.

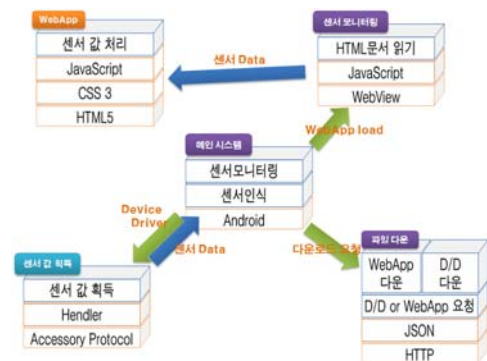


Figure 7: Integrated App service system

서버에는 JSON을 통해 정보를 주고받는다. 원하는 정보가 있을 때 웹 주소에 데이터의 정보를 실어 서버에서 해당 결과를 반환한다. 통합 앱은 반

환된 결과를 해석하고 디바이스 드라이버 및 웹페이지를 다운로드 한다.

4.4 센서 웹서버

센서 모듈의 디바이스 드라이버와 센서 값을 표시해주는 웹페이지를 제공해주는 웹서버를 제작하였다. 웹 서버에서 메인 페이지는 그림 8과 같이 구성하였다. 센서 액세서리 플랫폼은 모든 센서 모듈에 대한 정보를 가지고 있을 수 없기 때문에 필요한 디바이스 드라이버와 웹페이지를 서버에서 제공하도록 하였다.

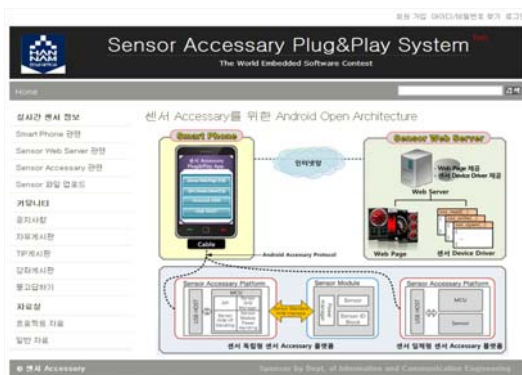


Figure 8: Main page of sensor accessory Plug and Play system

또한 센서 개발자들이 자신이 만든 센서 디바이스 드라이버 및 웹페이지를 서버에 등록하도록 하였다. 홈페이지는 등록하고 삭제하고 검색하는 기능을 제공한다.

5. 결 론

본 연구에서는 USB 액세서리 Plug and Play를 위한 스마트폰 통합 앱 구조를 제안하였다. 통합 앱에서 센서 액세서리 플랫폼과의 연결 방법 및 센서 웹서버에서 디바이스 드라이버 다운로드 방법을 제시하였다. 또한 연결된 센서 모듈에 맞는 웹페이지를 웹서버에서 다운로드 하는 방법도 제시하였다.

스마트폰 USB액세서리 Plug and Play를 위한 통합 앱구조를 통해 다양한 센서들이 동작하는 것을

보였다. 하나의 앱에서 수많은 센서 값을 표기하기 위한 웹문서를 제작하고 필요시 다운로드하는 것을 보였다. 또한 한번 다운로드 된 디바이스 드라이버와 웹문서는 재사용 시 대기시간이나 로딩시간을 줄이고 빠른 응답 효과를 기대할 수 있었다.

스마트폰의 USB에 연결할 수 있는 다양한 센서 장치들이 개발되고 있으며 본 연구의 결과를 활용하면 센서 설치가 용이하지 않은 공간이나 통신선로가 설치되어 있지 않은 공간 즉 선박이나 플랜트 혹은 대단위 생산시설과 같은 큰 공간에서 이동이 용이한 스마트폰을 통하여 각종 계측이 이루어 질 수 있으며 그 결과를 인터넷을 통하여 데이터를 집적할 수 있을 것으로 기대된다.

본 연구와 병행해서 향후 다양한 센서 값을 웹 문서에서 표기하기 위한 표준방법의 설계가 필요하다. 또한 센서들의 물리량을 처리하기 위한 Calibration 라이브러리 제작에 대한 연구도 필요할 것이다.

참고문헌

- [1] Y. J. Jang and C. W. Kim, "Evolution of smart phone and effect of android", The Communications, The Korean Institute of Information Scientists and Engineers, vol. 28, no. 5, pp. 48-56, 2010 (in Korean).
- [2] L. K. Weon and K. S. Rhyu, "Implementation of sensor network for toxic gas detecting on shipboard", Proceedings of The Korean Society of Marine Engineering Conference, pp. 234-235, 2012 (in Korean).
- [3] C. D. Park, B. J. Lim, H. R. Lee, B. W. Choi, S. J. Hwang, and K. Y. Chung, "A basic study of wireless sensor monitoring system configuration for active machinery", Proceedings of The Korean Society of Marine Engineering Conference, pp. 119-120, 2012 (in Korean).
- [4] S. B. Eun, S. S. So, and B. H. Kim, "A sensor node operating system architecture providing sensor transparency", Proceedings of The Korean Institute of Information Scientists and Engineers, vol. 35, no. 1, pp. 311-312, 2008 (in

Korean).

- [5] S. H. Bang and S. B. Eun, "A sensor device manager supporting sensor transparency" Proceedings of Autumn Conference of The Korea Information Processing Society, vol. 15, no. 2, pp. 998-1000, 2008 (in Korean).
- [6] J. S. Park, S. B. Eun, N. S. Kim, and Y. B. Moon, "Dynamic loading and linking of sensor device drivers for sensor module plug & play", Proceedings of Autumn Conference of The Institute of Embedded Engineering of Korea, pp. 71-74, 2011 (in Korean).
- [7] Y. B. Park, A Sensor Node Platform Supporting Sensor Plug & Play, Thesis of Master Degree, Department of Information Communication, Hannam University, 2010 (in Korean).
- [8] ATmega128(L) Data Sheet, Atmel Corporation, June 2011.
- [9] Android Accessibility, http://eyes-free.googlecode.com/svn/trunk/documentation/android_access/index.htm, Accessed March 10 2014.
- [10] Android Development Kit, <http://developer.android.com/tools/adk/index.html>, Accessed March 10 2014.
- [11] Reality of ADK device control, http://www.imaso.co.kr/?doc=bbs/gnuboard.php&bo_table=article&wr_id=38180, Accessed March 10 2014.