

# 스마트폰을 이용한 가전기기 제어

백승범, 김용휘  
 한국산업기술대학교 컴퓨터공학과  
 e-mail:{123tmd123, kyhwi125}@kpu.ac.kr

## Home Appliance Device Control using Smart Phone

SeungBeom Baek, YongHwi Kim  
 Dept. of Computer Engineering, Korea Polytechnic University

### 요 약

본 연구에서는 Java기반 Android App과 라즈베리파이 보드를 기반으로 Wi-fi 통신을 통해 스마트폰과 라즈베리파이를 연동하여 집안의 커튼, 전구, 카메라 등과 같은 기기들을 제어하는 시스템을 구현하는 방법을 제시하였다. 기존의 가정의 전구와 커튼과 같은 전자기기들을 스마트폰을 통하여 자택에서 편안하게 제어할 수 있으며 향후 지어지는 신축 아파트들이나 병원과 같은 시설에서 이러한 IoT 기술이 접목될 가능성이 있기 때문에 라즈베리파이와 하드웨어를 연동하여 가정기기를 제어하는 시스템을 구현하였다.

### 1. 서 론

최근에 사물인터넷을 이용한 기술들이 발전하고 있다. L 사의 반려동물 케어 스마트 홈 IoT 시스템의 가입자는 2017년 6월 기준으로 80만 명이 넘었다. 다른 국내건설사들도 주차장, 태양열 발전 설비, 냉난방시스템 등에 IoT 기술을 적용하여서 고급스러운 이미지를 이용하여 마케팅하려고 한다.[1]



그림 1. 국내외 IoT 시장 규모 전망

그림 1의 전문 컨설팅 업체인 Machina Research에 따르면 세계 IoT 시장 규모는 2015년 약 3천억 달러에서 2020년 1조 달러로 연평균 28.8% 성장이 전망되며, 국내 IoT 시장 규모는 동기간 3.3조원에서 17.1조원으로 연평균 38.5%로 국내외 IoT 시장은 연평균 20% 이상 고성장이 예상된다.[2]

구분	내 용	잠재적 경제효과 (억달러, 연간)	
사용자	가사 자동화	가사소모 시간 절약 17%	1,340 ~ 1,970
	에너지 관리	냉난방 등 에너지 20% 절약	510 ~ 1,080
	안전·보안	재난 대비 재산 피해 10% 절감	150 ~ 220
공급자	사용기반 설계	제품 사용 모니터링을 통해 제품 개선	30 ~ 170
	판매전 분석	소비자의 제품 구입 가능성 판단	0 ~ 50
계		2,030 ~ 3,490	

자료 : McKinsey & Company.

그림 2 IoT 기반 스마트 홈의 잠재적 경제효과

그림 2는 컨설팅전문회사 McKinsey & Company의 자료이다. 자료에 따르면 스마트 홈은 2025년경 까지 가사 자동화, 에너지 관리 효율화 등을 통해 연간 최대 3,490억 달러의 경제적 효과를 창출할 것으로 기대된다.[2]

### 2. 관련연구

#### 2.1 사물인터넷

IoT(Internet of Things)는 정보통신기술을 기반으로 여러 기기들을 네트워크로 연결하여 사람과 사물 또는 사물과 사물사이를 통신하거나 정보를 교환하는 것을 말한다.

구분	스마트홈 1.0	스마트홈 2.0	스마트홈 3.0
기간	~ 2009년	2010년 ~	2015년 ~
주 고객 대상	대형주택 소유 부유층	고급주택 혹은 공동주택	주택 뿐이 아닌 사회적인 단위
서비스 모델	홈 콘트롤	보안, 에너지 관리, 홈 콘트롤	보안, 제어는 물론 데이터에 의한 예측 서비스
제어 방식	전용 콘트롤 패널	전용 패널+유,무선 전화	스마트 폰 및 모든 유,무선 단말
네트워크 방식	유선(Ethernet,RS-485)	유선(Ethernet,RS-485),무선	무선(WPAN),유선
플랫폼	전용 플랫폼	오픈 플랫폼(레세 프로토콜)	Global Standard 오픈 플랫폼

그림 3. 홈 네트워크 패러다임의 변화

그림 3과 같이, 홈 네트워크 시장은 2009년까지 유선 기반의 스마트 홈 1.0세대에서 전용 패널에 유,무선 전화를 합한 오픈 플랫폼 스마트 홈 2.0 세대를 거쳐 주택뿐만 아니라 사회적인 단위로 고객층이 옮겨간 IoT 및 클라우드 기반의 3.0으로 진화 하였다. 또한 홈 컨트롤이 주력 서비스 모델에서 모바일 기반의 보안, 제어, 에너지 관리는 물론 데이터에 의한 예측 서비스를 제공하는 융합형 서비스로 패러다임이 변화하였다. [3]

#### 2.2 라즈베리파이

본 연구에서는 라즈베리파이3 모델B를 사용하였다. 라즈베리파이 3의 하드웨어 스펙은 다음과 같다.

Name	Raspberry Pi 3 Model
Price	\$35
Processor	ARM Coretex-A53
Clock Speed	1.2Ghz
RAM	1GB
Multi Core	Quad
Flash	Micro SD Card
GPIO	40
Ethernet	10/100
USB	USB 2.0 X 4
Power	5V
Video Out	HDMI, Composite
Audio Out	HDMI, Analog

표 1 라즈베리파이3 스펙



그림 4 GPIO 핀맵

라즈베리파이의 GPIO핀을 이용하면 각종 센서, 모터를 이용한 원격 시스템을 개발할 수 있다. GPIO핀은 하드웨어를 제어할 수 있도록 40개의 핀이 있는데 이러한 핀에는 3v, 5v, GROUND핀, 다른 블루투스 통신모듈을 이용하거나 온습도 같은 데이터 값을 송수신할 수 있도록 하는 I2C 핀, 하드웨어에 제어신호 송신을 위한 GPIO 전용 핀이 제공된다. 하드웨어로 측정된 데이터를 통해 사용자의 환경을 분석하고 학습할 수 있으며, 이러한 데이터를 기반으로 생활시설을 조정하여 사용자가 원하는 환경에 가깝게 최적화하는 제어기술에 사용될 수 있다. 라즈베리파이 출시 이전에는 ‘아두이노’ 라는 임베디드 플랫폼이 있었다. 아두이노는 무선인터넷 이용에 별도로 인터넷 모듈을 장착해야 했으나 라즈베리파이는 기본적으로 지원되고 영상촬영의 경우에 CPU처리성과 RAM의 기억공간이 소형PC에 버금가기 때문에 카메라로 이미지를 촬영하거나 영상을 녹화해서 저장하고 핸드폰이나 PC에 스트리밍하는 것이 가능하다. 또한 CSI포트와 USB포트를 이용해 카메라를 여러 개 이용할 수 있으며 영상처리 기법으로 숫자, 문자, 사람의 형태를 인식하여 응용하기 용이하다. 이에 따라 라즈베리파이는 아두이노에 비해 발전된 기능과 성능을 제공하여 확장성과 응용성이 넓다는 점에서 라즈베리파이를 이용하기로 하였다.

### 3. 설계 및 구현

#### 3.1 개발환경

구분	환경
Platform	Android 7.1 Nougat
Structure	Activity(Mainactivity, Rcvthread, camera Activity, Camera Pan/Tilt, curtain, light control Button)
Layout	Xml (System control button design)
Hardware Control	Raspberry Pi, Python
Client Control	Java language

표 2. 스마트폰을 이용한 가정기기 제어 개발환경

#### 3.2 하드웨어 I/O 모듈 상세설계

하드웨어 제어는 라즈베리파이의 GPIO포트를 사용하였다. 라즈베리파이 전구의 경우에는 5V핀과 GROUND로 전원을 주고 GPIO 24번 핀을 이용하여 제어하였다. 라즈베리파이 커튼의 경우에는 5가지의 핀을 사용하였다. 먼저 라즈베리파이 커튼에 사용하는 모터의 경우에는 일반 커튼의 봉을 들어 올릴만한 힘이 있어야하기 때문에 토크 0.4~6kg, 감속비 1/500, 회전수 5000rpm, 12V의 DC모터를 사용하였다. 위와 같은 스펙의 모터를 사용하기 위해서 L298N이라는 모터드라이버를 사용했다. 모터드라이버는 AA배터리 4개를 연결한 12V배터리 팩으로 VCC(IC power-supply pin)와 GROUND로 전원을 주었다. 그리고 라즈베리파이의 GPIO와 연결하여 IN1은 19번 핀, IN2는 26번 핀, VMS는 13번 핀을 신호 핀으로 사용하였다. 그리고 모터드라이버는 모터의 +, - 극을 연결해서 모터 드라이버와 접지시켰다. 카메라의 경우에 외부카메라는 라즈베리파이의 CSI포트를 사용하고 내부카메라의 경우에는 USB포트를 사용하였다. 내부카메라는 Pan/Tilt 기능을 구현하는데 서보모터를 사용하였다. Pan/Tilt기능은 카메라가 상하좌우로 움직여서 고정된 카메라보다 더 넓은 시야를 확보할 수 있도록 하려는 기능이다. 이 때 Pan기능은 좌우로 움직이는 기능이고 Tilt는 상하로 움직이는 기능을 의미한다. Pan/Tilt 기능을 구현할 때 서보모터를 사용하였는데 둘 다 라즈베리파이의 5V와 GROUND로 전원을 인가하였다. 그리고 Pan 기능의 모터에는 GPIO 18번으로 신호를 주었고 Tilt 기능의 모터는 GPIO 23번으로 신호를 주었다. 그리고 외부카메라의 경우에는 외부인을 확인하고 문을 열어주는 기능을 구현하는데 서보모터를 사용하였는데 5V와 GROUND로 전원을 인가하고 GPIO 17번 핀을 사용하여 신호를 주었다.

#### 3.3 안드로이드와 네트워크 설계

안드로이드 측을 클라이언트로 사용하여 라즈베리파이와 통신하여 하드웨어를 컨트롤하는 기능을 구현하였다. 그림 5는 라즈베리파이와 안드로이드사이의 네트워크와 하드웨어 제어구조이다. 안드로이드와 라즈베리파이는 UDP로 방식으로 연결되었으며 주변의 와이파이 공유기를 통해 연결하여 네트워크가 공유된 상태에서 컨트롤하였다. 안드로이드 측에서 연결할 라즈베리파이의 IP주소와 공유할 와이파이의 주소를 설정하여서 연결하였다. 안드로이드 쪽에서 구현된 UI의 특정 버튼을 누르면 버튼에 따른 문자열을 라즈베리파이로 전달하여 문자열로 통신이 되도록 하였는데 이 때 스마트폰으로 신호를 보냈을 때 신호가 정확히 수신이 되었다면 라즈베리파이에서 안드로이드 쪽에 수신한 신호를 다시 보내서 문자열을 출력하도록 프로그래밍하였다. 또한 라즈베리파이에서 수신한 결과에 따라 Pan/Tilt 카메라의 각도, 전구, 커튼과 연결된 전동 모터 등이 작동된다. 안드로이드 쪽에서 하드웨어 제어를 위해 전달하는 문자열의 내용과 라즈베리파이 측에서 제어하는 기기는 표3과 같다.

하드웨어	명령어 문자열	하드웨어	명령어 문자열
전등	light on light off	커튼	curtain up curtain down curtain stop
도어락	door lock door open		
내부 카메라/Pan 서보모터	left right	내부 카메라/Tilt 서보모터	up down

표3

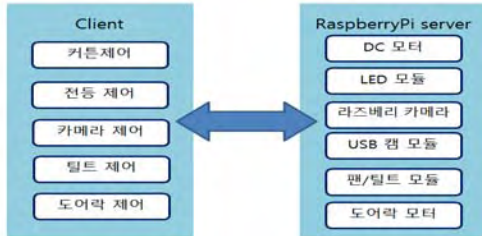


그림 5

### 3.4 라즈베리파이 제어

라즈베리파이에서는 하드웨어 제어를 비롯한 서버의 기능을 담당하고 안드로이드는 클라이언트의 기능을 담당하였다. 안드로이드 측의 내부카메라와 외부카메라는 UV4L이라는 스트리밍 서비스를 이용하였다. 라즈베리파이에서 내부카메라와 외부카메라의 포트를 라즈비안의 권한 설정에서 허용하고 카메라를 연결한다. 그리고 내부카메라와 외부카메라에서 촬영되고 있는 영상을 UV4L을 이용해서 웹페이지를 생성하고 실시간으로 스트리밍 하였다. 외부카메라는 라즈베리파이의 8080포트를 이용하여 스트리밍 하였고 내부카메라는 8090포트를 이용하여 스트리밍 하였다. 이렇게 스트리밍한 것을 안드로이드 폰으로 볼 수 있도록 스트리밍영상의 화면 비율에 맞게 안드로이드 측의 액티비티 화면을 설정하였다. 안드로이드 측에서 외부카메라를 통한 하드웨어 기기 제어는 python으로 프로그래밍 하였다. 라즈베리파이에서 내부카메라의 Pan을 제어하는 기능은 안드로이드에서 right/left 버튼을 누를 때마다 스마트폰에서 라즈베리파이로 신호를 전달하여 카메라가 좌우로 일정각도가 움직인다. 이 때 서보모터는 GPIO 18번으로 신호를 주었으며 default값으로 모터의 회전반경 180도에서 90도를 가리키도록 지정하였다. 이러한 초기상태에서 기준각도에서 right/left 버튼을 누를 때마다 전역변수로 설정해 놓은 초기 값의 변수를 +하거나 -하는 방식으로 일정각도가 움직이도록 프로그래밍 하였다. 이러한 원리에 따라서 Pan기능을 담당한 서보모터가 5도 각의 단위로 방향을 바꿀 수 있으며 방향을 회전하는 최소각도의 단위는 필요에 따라 프로그래밍 하여 변경할 수 있다. 마찬가지로 카메라의 Tilt기능은 GPIO 23번 포트로 신호를 보내고 안드로이드 클라이언트의 up/down버튼을 누를 때마다 문자열로 명령을 송신하였을 때 카메라의 Tilt를 담당하는 서보모터가 위와 같은 원리로 작동한다. 도어락의 경우 외부카메라에서 위와 같은 방법으로 스트리밍 되고 있는 영상으로 방문객을 확인하고 문을 열거나 닫는 기능이다. 도어락에서도 서보모터를 사용

하는데 도어락의 서보모터는 GPIO 17번으로 신호를 주었다. 서보모터의 default 값을 회전반경의 180도에서 90도로 설정해 놓았다. 이 때의 default 값은 도어락의 결쇠가 문을 열지 못하게 잠겨있는 상태에 위치되어 있는 잠금 상태이다. 이 상태에 안드로이드 측에서 doorlock open버튼을 누르면 90도에 잠금으로 설정되어 있던 서보모터가 12시 방향으로 회전하면서 문을 열 수 있는 상태가 된다. 이 때를 도어락의 열림 상태로 지정하였다. 전구의 경우에는 GPIO 24번 포트를 사용하여 신호를 전달하였으며 안드로이드 측에서 전구를 제어하는 버튼을 누르면 라즈베리파이가 수신하여 GPIO 24번 포트에서 신호를 주어서 켜지는 것에는 True 신호를 전달하고 꺼지는 기능에는 False 신호를 전달하여 전등을 제어한다. 커튼은 GPIO 19번, 26번, 13번 포트를 사용하였다. 안드로이드 측에서 내림, 올림, 정지에 대한 명령을 수신하여 제어한다. 내림에 대한 기능을 수행할 때 모터 드라이버의 IN1에 연결된 GPIO 19번 포트에는 True 값을 보내고 IN2에 연결된 26번 포트에는 False 신호를 보내서 모터를 오른쪽방향으로 회전하게 하여 제어한다. 반대로 올림에 대한 기능을 제어할 때는 19번 포트에는 False, 26번 포트에 True 신호를 보내서 모터가 왼쪽방향으로 회전하게 하여 제어한다. 그리고 정지에 대한 제어를 할 때는 19번과 26번 포트에 둘 다 False 신호를 주어서 모터의 동작을 정지시킨다. 그리고 13번 포트에는 신호를 주는 PWM(Pulse Width Modulation)값을 변경하여 모터의 회전속도를 조절하여 커튼이 내려가거나 올라가는 속도를 정하는 기능으로 작품에서는 속도 값을 95/100로 하였다.

구분	환경
LED	4-LED Light module
DC Motor	RB-35GM 3448D(12V)
Servo Motor	DF05BB
Motor Driver	L298N
Camera	Raspberry Pi camera NOIR V2
USB cam	Logitech C170
Board OS	Raspberry Pi 3 B

표4 최종 시연 모델 및 개발 환경

## 4. 결론 및 향후 연구과제

### 4.1 구현결과

라즈베리파이를 활용한 시스템 구현결과 내부카메라 스트리밍 및 5도 단위로 상하좌우의 각도를 변경 시키는 것이 가능한 Pan/Tilt 기능, 외부카메라 스트리밍 및 외부화면을 보고 도어락을 제어하는 기능, 커튼기능, 전구기능이 구현되었다. 아래의 그림 6을 보면 메인액티비티에서는 전등과 커튼을 제어하고 안드로이드 측의 외부카메라와 내부카메라를 안드로이드 폰에서 볼 수 있게 라즈베리파이에서 스트리밍한 영상을 안드로이드의 어플리케이션에 페이지를 생성하였고 스마트폰 카메라 뷰 아래에 도어락 제어버튼을 넣었

다. 그리고 내부카메라에서 촬영되고 있는 영상을 보고 카메라의 각도를 조정할 수 있게 Pan/Tilt 기능을 제어할 수 있는 버튼을 넣었다.

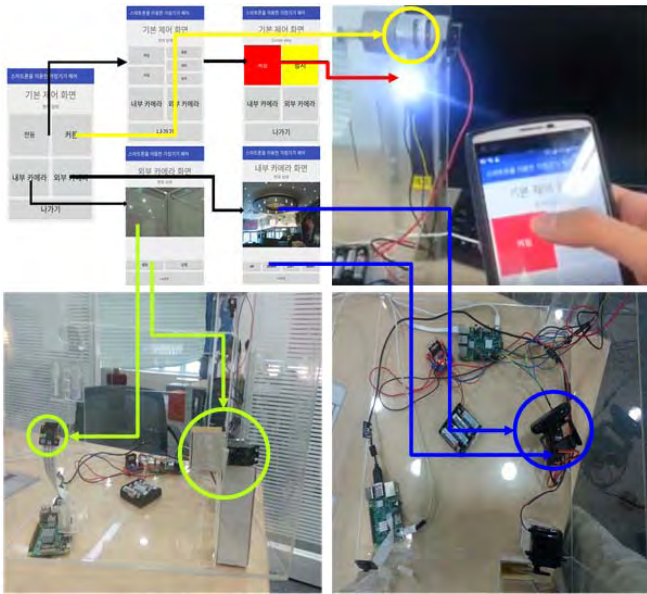


그림 6

#### 4.2 기대효과

본 논문에서는 라즈베리파이를 이용하여 커튼, 카메라, Pan/Tilt, 전구, 도어락을 제어할 수 있는 하드웨어를 구성하고 라즈베리파이에서 하드웨어와 연동하고 서버를 통해서 안드로이드와 연동하여 제어하는 시스템을 구현하였다. 기대효과로는 본 연구의 안드로이드를 통한 하드웨어 제어기술이 향후 아파트나, 병원 등과 같은 곳에 적용될 수 있다는 것이다. 서보모터와 DC모터를 통하여 부재중 애견을 돌보거나 집안의 가스밸브와 같은 기기들을 외부에서 컨트롤 하는 등의 응용 기술에 접목이 될 수 있다.

#### 4.3 본 연구의 한계점

본 연구의 구현상 한계점은 네트워크의 보안성이 취약하다는 점이다. 그리고 라즈베리파이 기관의 특성상 촬영된 영상을 웹 하드에 저장할 수는 있지만 블랙박스과 같이 찍힌 영상을 저장해놓기에는 저장 공간의 한계가 있다. 또한 제품의 네트워크를 연결하는 과정에서 iptime과 같은 와이파이 공유기를 이용하는데 다른 기기와 네트워크를 공유하고 데이터계층을 공유하면 라즈베리파이 내의 스트리밍 서비스는 원활하지만 컴퓨터나 스마트폰의 데이터 전송속도가 떨어질 수 있다는 단점이 있다.

#### 4.4 향후 목표과제 및 개선방안

첫 번째는 4.3에서 언급한 저장 공간에 대한 한계점에서 지적한 점을 보완하기 위해 영상 압축 표준인 35% 부호화 효율을 보이는 HEVC(High Efficiency Video Coding)를 사용하여 압축한 영상을 외부에 저장하여 블랙박스의 기능을 구현 할 것이다. 하루의 촬영시간에 내부카메라와 외부 카메라가 약 6기가의 용량을 저장하며 한 달이면 약 180기가의 용량이 축적된다. 이에 따라서 2TB 하드웨어를 이용하여 1년간의 자료를 보관하여 한 달 단위로 먼저 보관된

영상이 삭제되는 방식으로 영상을 저장하는 시스템을 구성하려고 한다. 두 번째로는 카메라에서 OpenCv를 이용해 얼굴인식을 하여 사전에 등록된 얼굴이 인식되었을 때 도어락이 열리는 기능을 추가하여 방법시스템의 보안효과를 개선할 것이다. 세 번째로는 실시간으로 이미지를 캡처하고 OpenCv를 이용하여 이미지를 구간으로 나누어서 RGB영상을 HSV영상으로 변환 후 이진화하여 Color값을 추출하여 Template Matching으로 Object Tracking Algorithm을 적용하여 물체의 움직임에 따라서 물체가 움직이는 방향으로 Pan/Tilt 기능이 작동하여 카메라의 방향이 자동으로 목표물을 따라가는 기능을 추가할 것이다. 네트워크를 공유하기 때문에 생기는 문제점에 대해서는 통신기술의 발전에 따라 이용 가능한 대역폭의 확장과 시간당 데이터의 전송량이 높아지는 변화에 맞추어 통신서비스와 설비에 대한 교체를 하여 전송속도에 대한 문제를 보완할 것이다. 보안에 대해서는 Wi-Fi의 기밀성 및 데이터의 무결성을 위해 AES 알고리즘을 사용하는 WPA2기술이 이미 기존의 네트워크에 적용되어 있지만 보안기술에 대한 특성상 네트워크 침입 기술의 변화와 이슈가 다양하기 때문에 보안에 대해서는 추후 점진적으로 보완해나가야 할 것으로 생각된다.

#### 5. 참고문헌

- [1] 매일경제: 이미연 기자, 「새 아파트에 '홈 IoT' 없으면 '팔소 없는 전빵」, 매일경제 MBN, 2017.07.24, 1쪽
- [2] 전해영 선임연구원, 『사물인터넷(IoT) 관련 유망산업 동향 및 시사점』, 현대경제연구원, 2016.07.11., 5-7쪽
- [3] 코맥스 황인철 부소장, 한국스마트홈산업협회, 국내 홈네트워크 현황 및 향후 발전 방향, [http://www.kashi.or.kr/board/index.html?action=view&board\\_id=pds&page=5&seq=15562](http://www.kashi.or.kr/board/index.html?action=view&board_id=pds&page=5&seq=15562), 2016.7.26
- [4] 이재상, 표윤석, 『라즈베리 파이 활용 백서 : 프로젝트 20』, 비제이퍼블릭, 2013.10.25.
- [5] 천인국, 『그림으로 쉽게 설명하는 안드로이드 프로그래밍』, 생능출판사, 2015.08.07.
- [6] 김성우, 『사물인터넷을 품은 라즈베리파이』, 제이펍, 2016.01.25.
- [7] 정수, 이종진, 정원기, 「라즈베리파이를 이용한 실내관리 시스템」(한국산학기술학회논문지, 제17권 제9호, 2016.09.), 8쪽
- [8] 강해용, 김용태, 이군석 「영상처리를 이용한 물체추적 알고리즘」 한국정보과학회 학술발표논문집 35(1C), 2008.6, 487-491 ( 5쪽)
- [9] 박인철, 「IDEC Newsletter, 반도체설계교육센터 (IDEC), 2013년 4월 30일, 3-4쪽