



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

석사학위 논문

스마트폰 분실방지 어플리케이션에
대한 연구

A Smartphone Application for
Loss Prevention

2015년 6월

승실대학교 대학원

IT융합학과

오 승 용

석사학위 논문

스마트폰 분실방지 어플리케이션에
대한 연구

A Smartphone Application for
Loss Prevention

2015년 6월

승실대학교 대학원

IT융합학과

오 승 용

석사학위 논문

스마트폰 분실방지 어플리케이션에
대한 연구

지도교수 신 오 순

이 논문을 석사학위 논문으로 제출함

2015년 6월

송실대학교 대학원

IT융합학과

오승용

오 승 용 의 석 사 학 위 논 문 을 인 준 함

심 사 위 원 장 유 명 식



심 사 위 원 노 동 건



심 사 위 원 신 오 순



2015년 6월

송실대학교 대학원

목 차

국문초록	v
영문초록	vii
제 1 장 서론	1
1.1 연구 배경	1
1.1.1 스마트폰 분실을 통한 1차 피해	3
1.1.2 스마트폰 분실을 통한 2차 피해	4
1.2 연구 목적	5
1.3 논문 구성	6
제 2 장 관련연구	8
2.1 스마트폰 분실예방 어플리케이션 관련 연구	8
2.2 아두이노 플랫폼 관련 연구	10
2.2.1 아두이노 보드 구성요소	11
2.2.2 아두이노 보드 종류	12
2.2.3 Arduino IDE	13
2.3 안드로이드 플랫폼 관련 연구	14
2.3.1 안드로이드 플랫폼 구조	14
2.3.2 안드로이드 플랫폼 초기화 과정	17
2.3.3 어플리케이션 구조	18
2.4 무선 네트워크 환경에서의 위치인식 방식	19
2.4.1 WLAN을 이용한 위치 인식 방식	19

2.4.2	RSSI (Received Strength Signal Indicator)	19
2.4.3	초음파를 이용한 위치인식 방식	20
2.4.4	기타 위치인식 방식	21
제 3 장	스마트폰 분실방지 어플리케이션 설계	25
3.1	제안시스템	25
3.2	아두이노 플랫폼 설계	26
3.2.1	아두이노 통합 개발 환경 소프트웨어 설치	26
3.2.2	아두이노 스케치	27
3.2.3	아두이노 하드웨어 설계	29
3.3	안드로이드 어플리케이션 설계	31
3.3.1	AndroidManifest.xml 설계	32
3.3.2	MainActivity 설계	33
3.3.3	신호간섭 제어 알고리즘 설계	39
제 4 장	구현 및 성능평가	41
4.1	구현	41
4.2	성능평가	43
4.2.1	RSSI 성능 테스트 결과	44
4.2.2	잡음제거 알고리즘을 사용한 성능 테스트 결과	45
제 5 장	결론	46
	참고문헌	47

표 목 차

[표 2-1] 아두이노 보드의 종류와 특징	12
[표 2-2] 안드로이드 플랫폼의 각 계층별 특징	16
[표 2-3] 컴포넌트 모델	18
[표 2-4] 초음파 기반 위치측정 장단점 분석	21
[표 3-1] Send signal과 Receive signal의 기능 표	33
[표 4-1] 안드로이드 시스템 개발환경	43
[표 4-2] 성능측정결과	44
[표 4-3] 잡음제거 알고리즘을 통한 성능측정결과(단위 dbm)	45

그림 목 차

[그림 1-1] TRACKING THE DEVICE	4
[그림 2-1] iBeacon 분실 방지 시스템 구조도	9
[그림 2-2] 아두이노 재단의 아두이노 소개 화면	10
[그림 2-3] 아두이노 ADK 보드 기본 구성요소	11
[그림 2-4] 아두이노 IDE	14
[그림 2-5] 안드로이드 플랫폼 구조	15
[그림 2-6] 안드로이드 플랫폼 초기화과정	17
[그림 2-7] IEEE 802.15.4a Alternative PHY	22
[그림 2-8] 토폴로지 시스템의 구조	24
[그림 3-1] 제안 시스템 구성도	25
[그림 3-2] 아두이노 스케치 라이프 사이클	28
[그림 3-3] 아두이노 setup() 소스코드	28
[그림 3-4] 아두이노 회로도	30
[그림 3-5] activity의 Life cycle	31
[그림 3-6] 어플리케이션 동작 소스코드	35
[그림 4-1] 아두이노 구현	42
[그림 4-2] 안드로이드 어플리케이션 구현	42

국문초록

스마트폰 분실방지 어플리케이션에 관한 연구

오승용

IT융합학과

송실대학교 대학원

본 논문은 스마트폰의 분실 도난을 예방하기 위한 어플리케이션이다. 기존의 논문은 분실 후 사후 대처 기능을 제공하기 위해 GPS를 통한 스마트폰의 위치를 추적하고자 했다. 하지만 본 논문에서 제안하는 어플리케이션은 분실을 예방하기 위한 방편으로서 아두이노(Arduino)와 안드로이드 어플리케이션간 블루투스 통신을 통한 신호감도를 이용해 스마트폰이 사용자의 주변에 있음을 항상 감시한다. 블루투스 모듈을 연결한 아두이노 보드는 안드로이드 스마트폰간 양방향 통신을 하게 되는데 양단간 신호감도를 측정하는 RSSI 방식을 이용해 신호감도를 실시간으로 측정하여 관리하고 사용자가 설정한 신호 감도이하의 수신감도가 감지되는 경우 아두이노에 특정 메시지를 전송하여 알람과 진동 그리고 LED가 동작하게 함으로서 스마트폰이 사용자 주변에 없음을 자각할 수 있도록 구현하였다. 또한 스마트폰에서도 벨소리, 진동 등이 발생하게 되어 분실했거나 도난당한 스마트폰의 위치정보를 전달할 수

있다. 본 논문에서 제안한 어플리케이션의 프로토타입을 통해 기존 어플리케이션보다 정밀한 신호감도를 측정하고 제어 할 수 있음을 입증하였으며 실생활에 유용하게 사용할 수 있는 상용화의 가능성을 제시할 수 있었다. 이에 따라 기존의 분실 후 사후대처 기능을 가진 기존 연구에 비해 보다 정밀한 센싱 기능을 통한 예방기능을 제시함으로써 분실행위에 대해 신속하게 대응 할 수 있는 이점을 가진다.

ABSTRACT

A Smartphone Application for Loss Prevention

OH, SEUNG YONG

Department of IT Convergence

Graduate School of Soongsil University

This paper presents an application to prevent lost of smart-phones. Previous works usually realized a reactive function that tracks the location of a smart phone through GPS after it is lost. In this paper, a new application is proposed to prevent loss of smart phones by measuring signal strength for Bluetooth communication between Arduino and Android application that always monitors the location of a smart phone around the user. Arduino board connected to the Bluetooth module can provide two-way communication between Android and the smart phone. Once the received signal strength indicator (RSSI) falls below a predetermined threshold, Android sends a specific alarm message to the Arduino and turns on vibration and LED lamp, making the smart phone user recognize loss of the smart phone. A noise reduction algorithm is also presented to improve the performance of the application.

제 1 장 서 론

1.1 연구 배경

국내 스마트폰 유저는 4천만명에 육박한다. 휴대폰과 개인용 휴대단말기의 장점이 결합된 스마트폰은, 휴대폰 기능에 인터넷접속, 팩스 송·수신, 일정관리 등과 같은 PDA기능에 데이터 통신기능을 통합시킨 것으로서, 완제품으로 출시되어 정해진 기능만 사용하던 기존의 휴대폰과는 달리 수천여 종의 다양한 어플리케이션 (응용프로그램)을 사용자가 원하는 대로 설치, 추가 또는 삭제할 수 있다는 점이 가장 큰 장점이다.

또한 무선인터넷을 통한 여러 브라우저를 이용하여 다양한 방법으로 인터넷에 직접 접속할 수 있다는 점, 사용자가 원하는 어플리케이션을 직접 개발하고, 다양한 어플리케이션을 통해 자신에게 맞는 인터페이스를 설정할 수 있다는 점 그리고 스마트폰 간 어플리케이션을 공유할 수 있다는 점 등도 기존 휴대폰이 갖지 못한 장점으로 부각되면서 이제 현실생활에서 스마트폰의 사용은 평범한 일상이 되어버릴 만큼 우리의 삶과 밀접하게 접근해있다.

하지만 스마트폰의 열풍의 그 이면엔 휴대폰 제조업체들에겐 점유율 확대를 위한 경쟁과, 무분별한 보급 확대로 인한 분실, 도난, 절도, 개인정보유출 등의 사회적 문제가 발생되었으며 더욱이 기업형 장물 스마트폰의 밀반출이 성행하면서 고가의 스마트폰을 통한 범죄가 점점 더 조직화 되고 있다.

지금까지 170개국을 넘는 1,500만명 이상의 아이폰 및 안드로이드 스마트폰 사용자를 위한 보안 어플리케이션을 제공하고 있는 미국의 모바일 보안 전문기업 Lookout은 ‘2012 모바일 분실과 발견’이라는 모바일 기기 분실 현황에 대한 조사 결과를 공개했다. 해당 내용에 따르면 Lookout에서 제공한 모바일 기기 위치 검색 서비스를 전 세계 스마트폰 사용자가 이용한 횟수는 2011년 한해 약 900만회에 이르고 있으며, 3.5초에 1대 비율로 분실된 모바일 기기의 위치를 찾고 있다고 한다. 더불어 2012년 한해 미국 내의 스마트폰 도난/분실에 따른 손실액은 약 \$300억에 이르는 것으로 추정 있다. 이중 ‘세계 도시별 모바일 기기 분실 정도를 알 수 있는 인터랙티브한 웹페이지’에서 분실주기, 지역별 손실 상당액과, 빈번히 발생하는 분실장소등을 살펴볼 수 있는데 조사결과는 다음과 같다.

- 1) 세계에서 모바일 기기를 가장 많이 분실하는 곳은 영국 맨체스터
- 2) 오후 9시~오전 2시가 가장 분실하기 쉬운 시간대
- 3) 분실대상 장소로는 BAR와 Pub이 가장 많음
- 4) 크리스마스 시즌에 분실할 가능성이 가장 높으며, 2011년에는 1,100만달라에 이르는 모바일 기기 분실
- 5) 2011년 서울에서 분실한 모바일 기기의 분실추정 금액은 \$200만, 일본 동경에서 분실한 모바일 기기의 추정금액은 \$380만 상당
- 6) 서울 유저는 4년에 1대, 동경 유저는 2년에 1대 비율로 분실 중

위의 조사결과에 따르면 보편적으로 스마트폰을 분실하게 되는 상황은 특정연령이나 성별 또는 지역에 따라 다른 문제가 아닌 언제 어디서든 누구나 겪을 수 있는 이슈임을 확인 할 수 있으며 분실로 인한 금전적인

피해로 끝나는 문제가 아니라 대량의 개인정보 유출로 인한 2차 피해가 발생할 수 있을 것으로 판단된다. 다음 절에선 분실하게 된 휴대폰으로 인해 겪게 될 1차 피해와 2차 피해 조사를 통해 본 연구의 타당성을 증명하도록 한다.

1.1.1 스마트폰 분실을 통한 1차 피해

스마트폰 절도 단속이 더욱 강화됐지만 터미널과 인터넷 카페에 여전히 ‘주운폰 삽니다’란 글이 게시되는 등, 분실폰 거래가 줄어들지 않고 있으며 이와 함께 장물업자들의 분실 폰 처리방법은 날로 다양해지는 것으로 나타났다. 대구지방경찰청에 따르면 2013년도 상반기 통신 3사 휴대전화 분실 접수내역은 약 137만 5천 490건으로 하루 평균 약 7천 641대가 도난되거나 분실되었으며 이들 중 대부분은 중국등의 해외로 밀반출되고 있는 것으로 나타났다. 분실폰 거래는 휴대폰 습득자가 분실폰 업자에게 문자메시지로 기종만 알려주면 장물업자들이 가격을 알려주는 방식으로 거래되고 있으며, 거래시 10여분 동안 경찰인지 아닌지를 파악하고 접근하는 것으로 알려졌다. 장물업자들의 분실폰 처리방법은 국외로 밀반출하여 USIM 칩만 꽂아 사용할 수 있도록 처리하는 방법과 스마트폰을 분해해 부품방식으로 개별 유통한 뒤 국외에서 재조립하는 방식, 그리고 국내 휴대전화 수리업체에 중고 부품으로 공급해 흔적을 남기지 않고 있기 때문에 현재로서는 분실된 스마트폰을 자의로 되찾기는 불가능 하다고 볼 수 있다.

1.1.2 스마트폰 분실을 통한 2차 피해

Security Perspectives사는 기업 및 개인에 대한 유사한 데이터를 저장한 50대의 스마트폰을 LA, 워싱턴, 뉴욕, 샌프란시스코, 캐나다의 오타와 등과 같은 변화가 일부러 분실하고 습득자들의 신고현황을 조사하는 실험을 수행하였다. 그리고 습득자들이 스마트폰에서 어떤 작업을 수행했는지 원격으로 모니터링 할 수 있는 툴을 마련하여 해당 내용도 함께 체크했는데, 실험결과 분실된 스마트폰이 돌아올 확률은 약 50% 정도이고 습득자의 96%가 특정 데이터에 접근했다는 통계가 나타났다. 즉 스마트폰 습득자 중 절반은 원주인에게 돌려주려고 했지만 96%는 스마트폰의 내부 데이터를 살펴보고 이 중 83%는 기업과 관련되어 있는 것으로 예상되는 파일명에 접근하려 하였다. 또한 60%는 소셜 미디어와 이메일 정보와 같은 민감한 개인정보를 확인하려고 했고, 43%는 은행 계좌에 접근 시도를 한 것으로 나타났다.



[그림 1-1] TRACKING THE DEVICE

위와 같은 사례를 통해 스마트폰의 분실 또는 도난은 1차적인 물질적인 손해뿐만 아니라 개인정보유출과 함께 금융사고와 같은 2차 피해의 발생 소지가 다분한 것으로 증명되었으며 자의로는 분실된 휴대폰을 되찾기 불가능하다고 판단된다. 그렇기 때문에 스마트폰을 분실한 이후 위치를 추적하는 사후 대책 방법 보다는 분실을 미연에 방지하거나 예방하는 어플리케이션의 필요성이 대두되고 있으며 많은 기업에서 스마트폰 분실 방지 관련 기술이나 기법에 대해 연구 되고 있다.

1.2 연구목적

본 연구에서는 스마트폰의 분실이나 도난을 미연에 방지하기 위해 스마트폰 분실시 빠른 대응을 할 수 있는 아두이노와 안드로이드 앱과의 블루투스 통신 시스템을 설계하고 이를 구축하기 위한 스마트폰 분실방지 어플리케이션을 제안한다.

사용자가 소지한 아두이노 플랫폼과 스마트폰의 블루투스 통신을 통해서로의 거리를 파악해 일정거리 이상이 되어 신호의 세기가 약해지거나 단절되는 경우 장비와 스마트폰 사이에 알람과 진동 그리고 경고등이 발생하여 사용자가 스마트폰을 분실했음을 인지하기 쉽게 하였으며 다양한 사용자 편의 기능을 추가하여 분실 및 도난에 대비할 수 있도록 안정성을 높였다. 또한 잡음제거 알고리즘의 추가로 잡음을 제어하고 기존 분실 방지 플랫폼과의 비교, 분석을 통해 본 논문에서 제안하는 스마트폰 분실 예방 어플리케이션의 우수성을 검증하고 스마트폰의 분실로 인한 금전적 피해와 개인정보 유출로 인한 2차 피해를 막을 수 있을 수 있도록 하였다. 본 연구에서는 그에 사용된 기술과 구현과정을 기술한다.

1.3 논문 구성

본 논문은 서론을 시작으로 2장에서는 아두이노 플랫폼과 안드로이드 플랫폼 간의 요소 기술과 관련 연구를 살펴보고, 3장에서는 2장에서 소개된 관련기술 및 알고리즘을 활용하여 감지기능이 향상된 스마트폰 분실방지 어플리케이션을 제안 및 구현과정을 서술한다. 그리고 4장에서는 해당 연구를 통해 완성된 어플리케이션으로 기능 검증 및 기존의 상용 어플리케이션과 본 논문에서 제안하는 시스템을 비교 분석 함으로서 평가를 하고 결론을 도출함으로서 논문을 마친다.

제 2장 관련 연구에서는 첫 번째로 오픈 하드웨어인 아두이노의 관련 기술과 구성요소 , 하드웨어 그리고 통합개발환경에 대해 살펴보고, 두 번째로 안드로이드 플랫폼 구조와, 라이브러리 그리고 초기화 과정과 어플리케이션 구조에 대해서 서술하며 마지막으로 안드로이드와 아두이노 간의 통신기술에 대한 상용화된 기술과 관련 연구에 대해 살펴본다.

제 3장 스마트폰 분실방지 어플리케이션 설계에서는 본 연구를 제안하고 안드로이드 플랫폼과 잠음제어알고리즘을 포함한 블루투스 및 안드로이드 소스코드를 살펴보고, 아두이노 플랫폼의 구조도, 회로도, 기능을 설계함으로서 본 연구의 전반적인 제작 과정을 통해 결과물이 만들어 지기까지의 과정에 대해 서술한다.

제 4장 구현 및 평가에서는 3장에서 제안한 시스템을 구현하고 스마트폰 분실방지 어플리케이션의 정상동작 여부와 정밀한 신호감도측정이 가능한지 성능 테스트를 통하여 본 논문에서 제안하는 스마트폰 분실방지

어플리케이션의 우수성을 검증한다.

제 5장에서는 결론 및 향후 발전방향을 제시하며 논문을 마친다.

제 2 장 관련연구

2.1 스마트폰 분실방지 어플리케이션 관련 연구

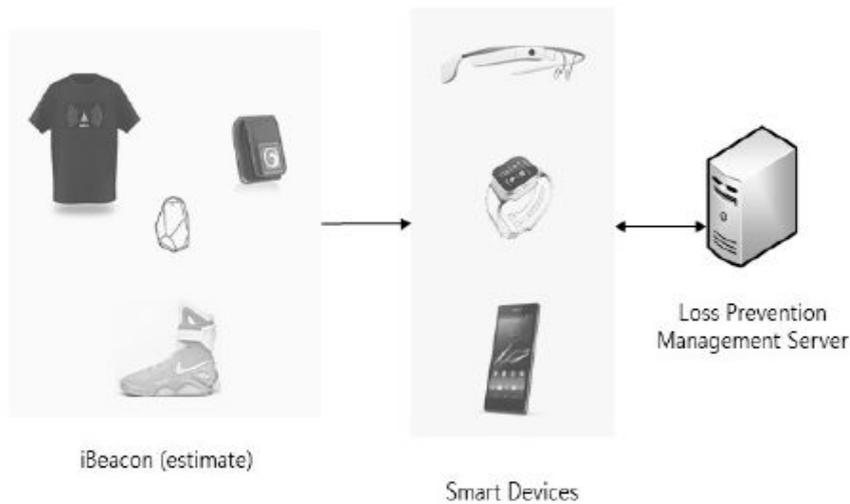
스마트폰의 분실을 방지하기 위한 연구는 주로 사용자가 이미 스마트폰을 분실한 이후 사후 추적을 위한 연구가 주류를 이루었다. 현재 대중적으로 상용화된 분실방지 어플리케이션인 여기요와 진돗개도 위와 같이 사후 추적을 위한 어플리케이션으로 분실된 스마트폰의 위치를 실시간으로 전송해주는 어플리케이션인데 이 방법은 주로 GPS를 활용하는 방법으로서 신호 음영 지역이 발생하기 때문에 측위가 불가능한 지역이 반드시 존재할 수 밖에 없으며 오차범위로 인해 대략적인 위치만 알 수 있을 뿐 정확한 위치의 추적이 어렵다. 또한 서론에서 밝힌 바와 같이 이미 분실된 스마트폰을 되찾기는 불가능에 가깝기 때문에 사후추적 어플리케이션은 분실에 대한 대응이 아주 미약하다고 볼 수 있다.

이러한 단점을 극복하기 위한 방편으로 네트워크 기반의 위치 측정 방법이 그 대안으로 대두되었지만 이 기술 역시 기지국과의 밀도에 따라 오차가 300m에서 수 km에 달할 정도로 오차 범위가 크기 때문에 현실적인 대안이 되지 못했으며, 이밖에 Wi-Fi 신호 기반 위치정보 측위 기술 및 위성 신호 기반 위치 정보 측위 방안 등 다방면의 연구가 이루어졌지만 비용적인 문제와 제한된 장소에서의 사용 그리고 넓은 오차범위로 인해 효율적이지 못했다.

최근 국내에서 연구된 측위기술로 스마트폰과 안드로이드 폰과의 적

외선 센싱을 통한 분실방지 어플리케이션에 대한 연구[3]가 있었는데 적외선 센서의 인식률문제와 함께 디바이스를 감지할 수 있는 거리가 매우 짧은 단점으로 인해 현실적인 대안이 되기 어렵다고 판단된다.

이밖에도 iBeacon을 이용한 웨어러블 스마트 디바이스 분실 방지 시스템이 연구되었는데[4] 각종 의류에 iBeacon을 장착하여 스마트폰과 서버와의 3자 통신을 통해서 장비의 거리, 현재 또는 분실한 시간, 사용자등의 정보를 확인할 수 있는 장점이 있었으나 스마트폰을 분실하게 되는 경우 이를 되찾을 방법이 없고 나머지 웨어러블 디바이스 역시 스마트폰의 부재로 인하여 분실 및 도난으로 처리될 수 밖에 없는 구조적 한계를 지니고 있었다.

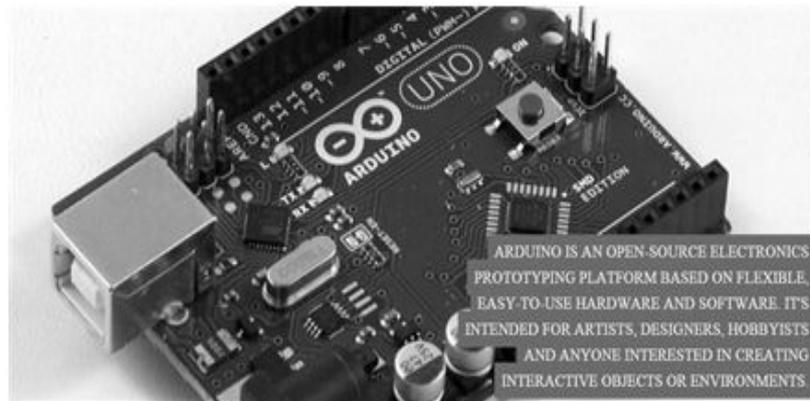


[그림2-1] iBeacon 분실 방지 시스템 구조도

이런 관련연구를 토대로 스마트폰을 분실하게 되는 경우 스마트폰과 또 다른 디바이스 간 곧바로 사용자에게 알림을 제공할 수 있는 실시간

감지 어플리케이션에 대한 연구가 필요하다고 판단되며 이와 함께 거리 측정오차가 크지 않고 음영지역에 강하며 분실로 판단되는 거리를 사용자가 임의로 지정할 수 있는 어플리케이션에 대한 방안이 요구된다.

2.2 아두이노 플랫폼 관련 연구

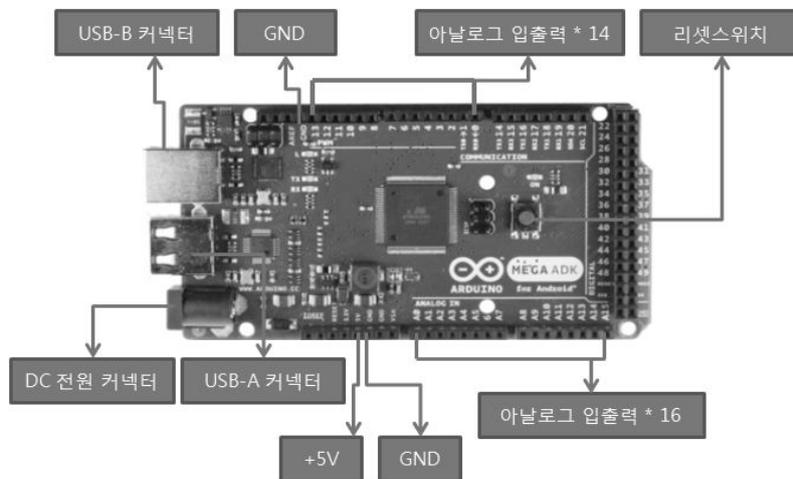


[그림 2-2] 아두이노 재단의 아두이노 소개 화면

2005년 이탈리아에서 탄생된 아두이노(Arduino)는 현재 가장 유명하고 널리 활용되는 오픈소스 하드웨어 플랫폼이다. 아두이노를 이용하면 우리가 상상하는 많은 것들을 만들 수 있는데, 아두이노의 마이크로 컨트롤러를 컴퓨터에 연결한 후 명령어(코드)를 작성하여 메모리에 로드하면 기판에 연결한 다양한 센서나 부품 등의 장치를 제어할 수 있다. 대표적으로 RC카, 로봇 청소기, 경보시스템, 정해진 시간에 동물들에게 먹이를 공급하는 장치등이 있으며 고도의 개발을 요하지 않고도 다양한 제품들을 제작할 수 있다.

아두이노는 전용 개발 환경인 Arduino Integrated Development Environment(아두이노 IDE)를 통해서 프로그래밍을 할 수 있고, 교육용 그래픽 유저 인터페이스 개발 도구인 스킵치(Scratch)를 아두이노 프로그래밍에 접목시킨 S4A (Scratch for Arduino)와 같은 개발 환경도 제공되고 있다. 아두이노, 아두이노와 관련된 모든 프로그램은 Creative Common Attrubute Sharelike License에 의해 저작자표시, 동일 조건 변경허락의 조건으로 저작권의 보호를 받고 있다.

2.2.1 아두이노 보드 구성요소



[그림 2-3] 아두이노 ADK 보드 기본 구성요소

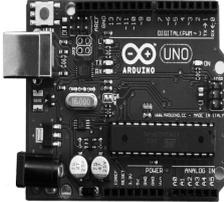
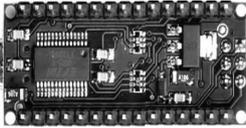
아두이노 보드의 구성요소는 보드의 종류에 따라서 약간씩 차이가 있지만 기본적으로 16개의 아날로그 입력 핀, 14개의 디지털 입출력 핀, 전원(+5V와 홍), 상태표시 LED(L,TX,RX 등), 리셋스위치 등이 있다. 전원 공급은 USB 포트 또는 DC 전원으로 할 수 있다.

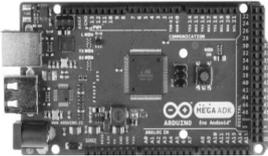
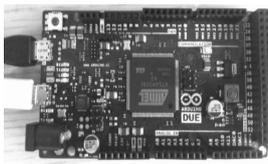
디지털 입출력 핀들을 이용해서 외부의 이진 신호를 읽거나 이진 신호를 내보낼 수 있고 아날로그 입력 핀은 외부의 아날로그 입력 값을 읽어들이는 핀으로 주로 센서와 연결하여 사용된다. 아날로그 출력 핀은 0~5V사이의 전압 값(256단계)을 가질 수 있다.

2.2.2 아두이노 보드 종류

아두이노 보드는 하드웨어와 소프트웨어 구조가 모두 개방된 오픈소스 플랫폼이기 때문에 다양한 종류의 공식 보드들이 있고, 그 공식 보드들과 호환되는 수많은 보드들이 있다. 대표적인 공식 아두이노 I/O 보드는 다음표를 통해 확인할 수 있다.

[표 2-1] 아두이노 보드의 종류와 특징

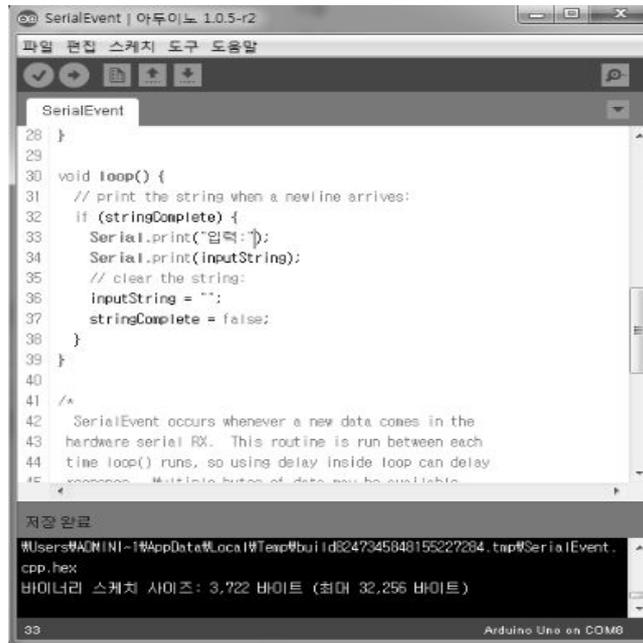
아두이노 보드명	아두이노 보드	특징
아두이노 우노 (Arduino Uno)		<ol style="list-style-type: none"> 1) 가장 많이 사용되는 기본적인 아두이노 보드이다. 2) atmega328p라는 프로세서를 기본으로 한 보드이다. 3) 보드의 핀 배열이 거의 표준과 같이 사용된다.
아두이노 나노 (Arduino Nano)		<ol style="list-style-type: none"> 1) 아두이노 우노 보드와 거의 동일한 구성이고 우노보드보다 훨씬 작다 2) 우노보드에 비해 비싸다. 3) 나노 보드 USB-시리얼 어댑터가 장착되어있지 않기 때문에 프로그래밍을 하려면 외장형 USB-시리얼 어댑터를 장착해야 한다.
아두이노 레오나르도 (Leonardonano)		<ol style="list-style-type: none"> 1) USB 기능이 내장된 atmega32u4 메인 프로세서로 사용한다. 2) 프로그램 다운로드와 시리얼통신

		<p>포트가 독립적으로 동작한다.</p> <p>3) 우노 보드보다 저렴하다.</p> <p>4) 키보드, 마우스 등 주변기기로 인식시킬 수 있어 활용도가 높다.</p>
아두이노메가 ADK (Arduino Mega ADK)		<p>1) ATmega2560프로세서로 사용한다.</p> <p>2) 우노보드보다 기능과 핀수가 많다.</p>
아두이노 두에 (Arduino Due)		<p>1) 32-bit ARM core 프로세서를 사용한다.</p> <p>2) 기능과 성능이 높고 핀 수가 매우 많아서 전문적인 제품 개발과 연구 목적으로 사용할 수 있다.</p>

2.2.3 Arduino IDE

아두이노 IDE(통합개발환경)은 아두이노 공식 웹사이트([arduino.cc /en/Main/Software](http://arduino.cc/en/Main/Software)) 에서 다운받을 수 있다. 지원되는 OS는 Mac, Windows, Linux이다. 아두이노 IDE는 아두이노 스케치에서 소스를 작성하고 컴파일 후 아두이노 보드상에 업로드할 수 있고 보드를 통해서 결과를 확인할 수 있다.

아두이노 보드를 USB 케이블을 통해 연결한 후 USB 디바이스를 인식시키기 위해서는 아두이노 보드용 드라이버를 설치해야한다. 아두이노 호환 보드에서 실행되는 소프트웨어를 스케치라고 하는데 스케치는 C언어와 아두이노에서 제공하는 라이브러리를 이용하여 작성할 수 있다.



[그림 2-4] 아두이노 IDE

2.3 안드로이드 플랫폼 관련 연구

본절은 스마트폰 분실방지 어플리케이션의 코어부분에 해당하는 안드로이드 앱에 해당하는 안드로이드의 소개와 플랫폼 구조, 초기화과정 그리고 동작원리에 대해 소개한다.

2.3.1 안드로이드 플랫폼 구조

안드로이드는 휴대용 장치에 미들웨어, 인터페이스를 내포하고 있는 모바일 운영체제 이다. Java로 응용프로그램을 작성하도록 하였으며, 컴파일된 코드를 구동하도록 런타임 라이브러리를 제공한다. 또한 개발 키트

(SDK)로 응용프로그램을 개발하기 위한 각종 도구와 라이브러리 및 API를 제공한다[10]. 안드로이드는 리눅스 커널 위에서 동작하며, 다양한 안드로이드 시스템에서 사용되는 C/C++ 라이브러리를 내포하고 있다. 안드로이드는 Java 가상 머신과는 달리 달빅(Dalvik)이라는 가상 머신을 통해 Java로 작성되어진 응용프로그램을 별도의 프로세스환경에서 실행하는 구조로 되어 있다. 현재는 주로 모바일 기기에서 많이 활용되고 있지만 다양한 정보 가전기기에서도 적용할 수 있도록 지원범위가 넓어지고 있는 추세이다[1].



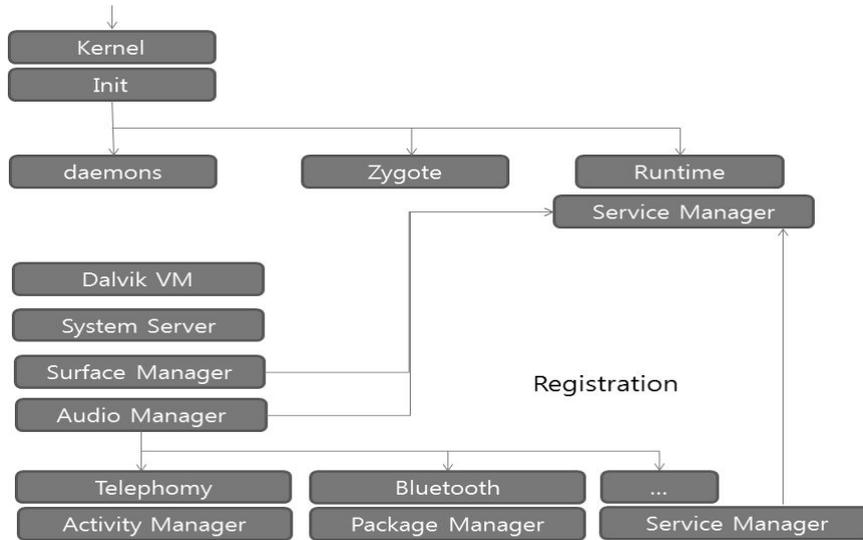
[그림 2-5] 안드로이드 플랫폼 구조

안드로이드는 위 [그림 2-5]와 같이 Application Framework, Applications, Linux Kernel, Libraries과 같은 4가지 계층으로 구성된다. Applications과 Application Framework 계층은 자바 언어로 구성되고, Libraries와 Linux Kernel은 C언어로 구성되어 있다. 안드로이드 플랫폼의 각 계층 특징을 살펴보면 [표 2-2]와 같다.

[표 2-2] 안드로이드 플랫폼의 각 계층별 특징

계층	특징
리눅스 커널	Android는 보안, 프로세스 관리, 메모리 관리, 드라이버 모델, 그리고 네트워크 스택과 같은 핵심 시스템 서비스에 대해서는 리눅스 2.6버전에 의존한다. 또한 리눅스 커널은 안드로이드 하드웨어와 안드로이드 플랫폼 스택 사이의 계층 간의 추상화된 계층 역할을 담당한다.
라이브러리	Android는 다양한 안드로이드 시스템 컴포넌트에서 사용되는 C++/C 라이브러리들을 포함하고 있다. 라이브러리는 장비의 처리속도를 결정짓는 중요한 요소이며, 이미지나 비디오 및 오디오 재생, 그래픽, 디스플레이 등의 각종디바이스를 제어할 수 있도록 도와준다
Application Framework	응용 프로그램은 하위의 커널이나 시스템 라이브러리를 직접 호출하지 않고 API를 통해 기능을 요청한다. API가 하드웨어 액세스를 추상화하면 UI제작에 필요한 위젯을 정의하고 시스템 라이브러리와 통신할 수 있는 다양한 레퍼런스를 제공한다.
Applications	사용자들이 직접 사용하는 응용프로그램이며, 어플리케이션 개발자들이 만들어야 하는 대상이다. 어플리케이션은 프로그래밍 언어인 Java를 사용하여 작성되었다. 응용프로그램은 프레임워크를 통해 하위커널이나 시스템 라이브러리를 사용한다.
Android Runtime	안드로이드 런타임은 Dalvik VM과 Java코어 라이브러리로 구성된다. 안드로이드는 모바일 환경에 최적화된 Dalvik VM을 사용하며 Dalvik VM은 각각 프로세스 별로 별도의 Dalvik VM이 할당되어 메모리 사용량을 낮추면서 안정성이 높은 다수의 가상 머신이 효율적으로 동작하도록 설계되어있다.

2.3.2 안드로이드 플랫폼 초기화 과정



[그림 2-6] 안드로이드 플랫폼 초기화과정

- 1) 부트로더(bootloader)가 커널을 로딩하고 Init 프로세스는 저수준 하드웨어 인터페이스를 담당하는 데몬(daemon)들을 시작한다. (debugger, usb, adb, radio 등)
- 2) 초기 상태의 Dalvik VM 프로세스인 “Zygote”가 생성된다.
- 3) 런타임 프로세스(Runtime)가 바인더(Binder)와 IPC 통신을 담당하는 서비스 매니저(Service Manager)를 시작한다.
- 4) 런타임 프로세스가 “Zygote” 프로세스에게 시스템 서버(System Server)를 실행하기 위해 필요한 새로운 Dalvik VM 인스턴스를 생성 요청한다.
- 5) Audio Manager & Surface Manager가 각각의 오디오 결과물과 그래픽 처리한다.
- 6) 기타 안드로이드 컴포넌트들이 시작된다.

2.3.3 어플리케이션 구조

안드로이드 어플리케이션은 Content Provider, Broadcast Receiver, Service, Activity 총 4가지로 구성이 되어있다. 어플리케이션이 어떤 컴포넌트를 사용할지 결정하고 나면, AndroidManifest.xml 파일에 컴포넌트 목록을 등록해야 한다. Manifest는 어플리케이션에 대한 전반적인 정보 및 권한정보를 담고 있는 파일인데 Manifest 파일 안에는 어플리케이션 컴포넌트(서비스, 액티비티 등)뿐만 아니라 사용하는 라이브러리, 어플리케이션의 이름 등과 같은 모든 정보들이 포함되어 있다. 즉 어플리케이션의 프로필과 같은 역할을 수행한다. 다음 [표 2-3]은 컴포넌트 모델에 관한 설명이다[6].

[표 2-3] 컴포넌트 모델

Activity	어플리케이션에서 하나의 화면을 Activity라 하는데, 기본 class를 상속하여 구현한 하나의 클래스를 각 Activity라 호칭하고, 사용자에게 그래픽 인터페이스를 화면에 뿌려주는 역할을 수행하기 위한 View와 Event응답으로 이루어져 있다
Broadcast Receiver	Broadcast Receiver는 데이터를 사용하는 네트워크 접속을 활성화 시키는 것과 같은 외부 이벤트를 처리하거나 작성된 어플리케이션의 소스코드 안에서 핸드폰에 통화가 걸려올 때 사용된다
Service	UI(User Interface)와 별도로 오랫동안 존재하며 실행되는 코드를 Service라 한다. 다른 어플리케이션이 실행된 상태에서 새로운 어플리케이션이 실행되면 미디어 플레이어의 Context.startService()를 이용해 Service로 실행되어 시스템은 음악 서비스나 동영상 재생 서비스를 멈출 때까지 계속 재생하게 된다.

Content Provider	어플리케이션은 생성된 자기 자신의 데이터를 SQLite 데이터베이스나 다른 방법을 사용하여 파일에 저장한다. 다른 어플리케이션이 데이터를 저장하거나 또는 데이터를 가져오는 작업을 수행한다.
---------------------	---

2.4 무선 네트워크 환경에서의 위치인식 방식

무선기술을 이용한 위치 인식 기술은 UWB, 적외선, 초음파, RF, RFID 등 다양한 분야에서 활발하게 연구 중에 있다. 이동통신 및 위성 통신 기술의 경우 서비스제공영역이 넓어 실외 환경에서는 탁월한 능력을 발휘하지만 건물내부 및 음영지역의 활용에 있어서 잡음 및 차단현상이 발생하여 많은 제약이 따른다[6].

2.4.1 WLAN을 이용한 위치 인식 방식

IEEE 802.11을 사용하는 무선 랜 환경을 기반으로 한 실내 측위 방법은 부가적인 보조 장비의 설치 없이 건물 내에 설치된 기존의 Access Point를 사용한다는 것이 큰 장점이다. 공통적으로 WLAN 기반의 건물 내부 측위 기술은 Access Point에서 송신하는 신호의 세기(RSSI)를 이용하여 단말의 위치를 결정하는 방법이다[2].

2.4.2 RSSI (Received Strength Signal Indicator)

RSSI는 수신되는 신호의 강도가 거리에 따라 달라지는 점에 기준을 두고 거리를 추정하는 방식이다. 송신 노드는 정해진 세기로 신호를 전

송하게되며 이를 수신하는 노드에서는 감쇠된 신호의 세기를 이용하여 거리를 측정할 수 있다[2].

RSSI 방식을 이용하기 위해서는 미리 지정된 다양한 환경에서 신호 세기들을 측정해야 하는데 이러한 과정을 수행하고 난 뒤 송신 신호들을 각각의 노드들이 수신함으로써 발생하는 신호의 감쇠 정도를 추정한 뒤 확률적 방법을 통해 미리 수집된 RSSI 표준과 비교하여 목표의 위치를 추정할 수 있다. 하지만 타겟과 센서간 많은 장애물이 존재하거나 실내 환경이 복잡할 경우 거리 측정에 극심한 오차범위가 발생할 수 있는 단점이 있다.

2.4.3 초음파를 이용한 위치인식 방식

초음파 중에는 사람의 귀로 직접 들을 수 있는 20Hz~20KHz 범위의 가청음이 존재하는데, 일반적으로 이런 가청음보다 더 높고 사람의 귀로 직접 들을 수 없는 주파수의 영역을 초음파라고 한다. 초음파를 이용한 방식으로는 느린 초음파 빠른 RF 신호와의 상대적인 전송 속도차를 이용해 대상의 위치를 찾아내는 방법이 있다. 하지만 잘못된 반사 신호로 인한 신호 수신, 방해물 그리고 높은 음향에 의한 간섭현상이 발생하는 단점이 존재하기도 한다. [표 2-4]는 초음파 기반 측위 기술의 장단점을 정리한 것이다.

[표 2-4] 초음파 기반 위치측정 장단점 분석

장점	단점
매우 정확한 위치 정확도	고가의 인프라 설치 및 도입비용
3차원 위치 측정이 가능	송신기의 위치 정보를 미리 알고 있어야 하며 송신기 배치위치에 따라 신호간섭 문제가 발생할 가능성이 높음

○ Active Bat

초음파를 이용한 대표적인 방식으로 Active Bat는 건물내부의 사무실 천장에 초음파 수신기를 부착하고, 사물 또는 사람에 초음파 송신기를 부착한다. 초음파 수신기와 송신기는 각각의 고유 식별 번호를 갖고 있는데, 서버에 연결되어 있는 수신기가 특정 송신기를 호출함으로서 호출된 송신기는 초음파를 발생하게 되고, 발생된 초음파를 천장에 설치되어 있는 수신기들이 감지하여 초기화함으로서 이후 초음파 신호를 수신 할 때까지 지연시간을 순차적으로 저장 한다[5].

빠른 RF 신호화 상대적으로 느린 초음파의 전송 속도차를 이용하여 대상 물체의 위치를 찾아내는 이 시스템은 3차원의 위치인식이 가능하고 저전력, 저비용의 시스템을 구성 할 수 있는 점과 천정에 많은 고정된 센서를 설치하여 센서의 간격을 조밀하게 함으로써 더욱 정교한 위치를 측정할 수 있는 장점이 있지만 확장에 대한 많은 비용과 값비싼 초기 설치비용이 단점으로 지적되고 있다.

2.4.4 기타 위치인식 방식

○ UWB

IEEE 802.15.4a 표준은 Chirp Spread Spectrum과 Impulse Radio

Ultra Wideband 기술을 채택 했으며, 이 두 기술은 Spread Spectrum 기술을 이용하기 때문에 간섭의 영향을 최소화 할 수 있다. [그림 2-7]은 IEEE 802.15.4a 표준의 구성을 설명하고 있다.



[그림 2-7] IEEE 802.15.4a Alternative PHY

UWB는 높은 주파수 대역으로 인해 파장이 매우 짧고, 기존에 무선통신에서 사용해왔던 변·복조 기능이 전혀 없어도 통신이 가능하기 때문에 저가격 저전력을 기반으로 한 고속데이터 통신에 적합하며 낮은 중심 주파수대역에서 동작하고 투과성도 높아 건물 내부에서 사용하기 좋다. 즉 위치 측정 정확도를 가지면서 적은 기반 시설을 요구하는데 극도로 짧은 무선 펄스를 반복하여 전송함으로써 수 GHz의 아주 넓은 광대역 주파수 스펙트럼을 사용하는 반면, 매우 낮은 전력 밀도를 갖는다.

그러나 UWB는 임펄스를 사용하기 때문에 대역폭이 확산되어 점차 많은 대역폭을 차지하게 됨으로서 기존에 사용중인 기타 무선 통신 시스템에 장애를 일으킬 여지가 있다. 따라서 상업적인 목적으로 사용되지 못하다가 최근 미국 통신위원회(Federal Communication Commission)에서 규정을 새로이 제정하면서 상업적으로 사용할 수 있게 허가한 이후 많은 회사들이 차세대 이동통신의 핵심 기술로 개발하고 있다.

○ RFID

RFID는 2차 대전 당시 아군비행기와 적군비행기를 원거리에서 판별하기 위해 개발되었으나 현재는 RFID 리더의 인식 범위 이내에 물체가 위치하고 있는지의 여부를 판단하는 기능으로서 주로 톨게이트의 하이패스로 많이 사용되고 있다.

태그와 리더로 구성된 RFID시스템은 리더의 인식 범위 내에 물체가 위치하게 되면 태그에 저장되어 있는 ID를 가장 가까운 리더에게 전송하게 되고 최종적으로 서버는 입력된 데이터를 근거로 하여 해당 리더의 인식 범위 이내에 대상 물체가 있는 것으로 판단하게 된다.

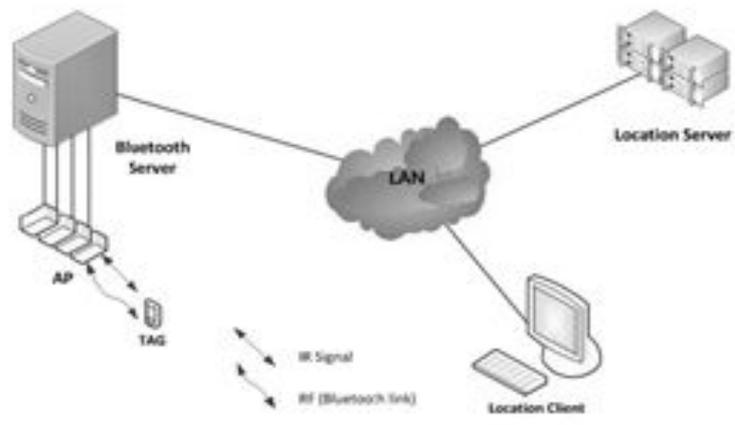
태그는 전원부가 포함되는지 그 여부에 따라 패시브 태그(passive tag)와 액티브 태그(active tag)로 구분하는데, 패시브 태그는 저렴하고 소형이며 수명도 반영구적이지만 액티브 태그에 비해 주변의 영향을 많이 받고 인식거리가 더 짧은 단점이 있으며, 액티브 태그는 50~100m의 비교적 먼 거리에서도 인식이 가능한 점 등 환경의 영향을 적게 받는 등의 장점이 있으나, 패시브 태그에 비해 수명이 더 짧고 고가이며 사이즈가 큰 단점이 있다.

○ Bluetooth

Bluetooth는 다양한 무선 장치(노트북, PDA, 휴대전화 등)가 약 10m~15m 정도의 비교적 짧은 거리에서 데이터 또는 음성을 전송할 목적으로 개발된 단거리 무선통신 표준으로서, 위치추적을 위해 설계된 통신기술이 아닌 데이터 통신을 위해 설계된 통신 기술이다[10, 11].

Bluetooth를 이용한 위치 추적 시스템은 Analog Location (RSSI

based)방식과 Binary Location (Room oriented)방식으로 구분되는데, Analog Location은 측정하고자 하는 위치에서 10~15m 정도의 간격을 두고 고정노드를 설치하며 이동노드와의 사이에 거리를 삼각측량법을 사용하여 얻는 방식이며, Binary Location은 하나의 AP를 하나의 영역에 설치하고, 이동시 가장 가까운 고정태그를 찾아 위치를 추정하는 방식이다. 이러한 위치 인식 시스템은 기존 Bluetooth 장비를 위치 추적 시스템에 연계할 수 있어 시스템의 구축비용을 크게 절감할 수 있고 또 저전력으로 활용할 수 있는 큰 장점이 있지만 저속통신기기이기 때문에 송수신 지연이 비교적 크고 이동상황에서의 측위 성능이 취약한 단점이 있다. Bluetooth를 이용한 위치 인식 시스템은 대표적으로 Antti연구팀에서 개발한 Bluetooth Local Positioning Application (BLPA) 과 Topaz시스템과 등이 있다[7].



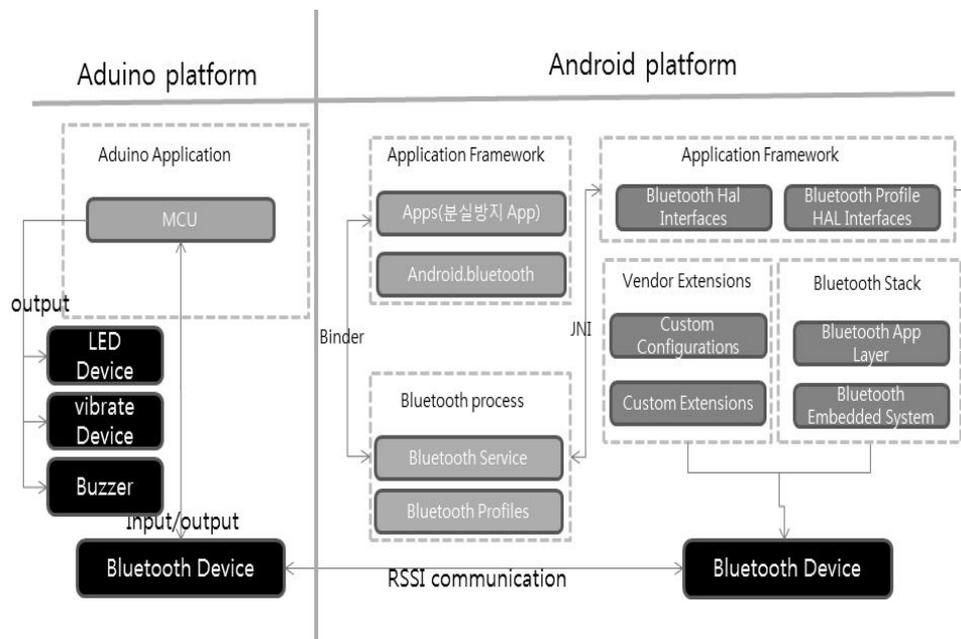
[그림 2-8] 토폴라즈 시스템의 구조

제 3장 스마트폰 분실방지 어플리케이션 설계

본 논문에서는 안드로이드 어플리케이션과 아두이노 플랫폼간의 블루투스 통신을 통해 양방향 통신을 함으로써 스마트폰의 분실이나 도난을 미연에 방지하고자 스마트폰 분실방지 어플리케이션을 제안한다.

3.1 제안시스템

제안 시스템의 전체 구성도는 [그림 3-1]과 같다.



[그림 3-1] 제안 시스템 구성도

제안 시스템은 안드로이드 사이트에서 Application을 다운로드하거나 JAVA 개발 어플리케이션인 이클립스를 통해서 테스트 폰에 설치된다. 안드로이드 어플리케이션은 스마트폰이 분실되었을 경우 스마트폰에 진동이나 벨 또는 특정 지인에게 전화를 울리게 함으로서 스마트폰이 분실되었음을 알리는 설정과, 아두이노 플랫폼에 진동, 벨, LED 등을 동작하게 하는 설정이 있는데 각각의 기능을 모두 다 사용하거나 일부를 사용하게 함으로서 사용 편의성을 제공한다. 이후 안드로이드 어플리케이션에 해당 아두이노 플랫폼의 고유 식별 정보를 등록하는 과정을 통해 안드로이드의 RSSI API를 사용하여 아두이노 플랫폼과 블루투스 통신을 하게 되며 페어링된 양 플랫폼의 신호 세기를 실시간으로 감지해서 안드로이드 어플리케이션에 측정된 신호감도를 보여준다. 페어링이 끊기거나 신호의 세기가 사용자가 등록한 기준치에 도달하지 못할 경우 안드로이드 어플리케이션에서 주어진 설정에 맞게 핸드폰이 분실되거나 도난당했음을 알린다.

3.2 아두이노 플랫폼 설계

제안 시스템에서 설명한 아두이노 플랫폼을 구현하기 위해서 아두이노 통합 개발환경 소프트웨어와 해당드라이버를 로컬피시에 설치하고 아두이노를 제어할 수 있는 스케치를 작성한 후 아두이노 보드에 다운로드하여 결과를 확인할 수 있다.

3.2.1 아두이노 통합 개발 환경 소프트웨어 설치

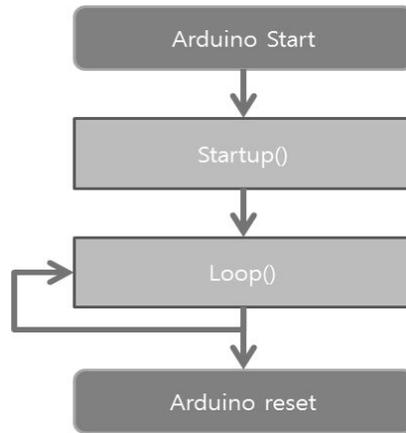
아두이노 스케치를 작성하고 컴파일하여 보드상에 업로드하기 위한 환

경구성을 하기 위해선 아두이노 통합 개발환경 소프트웨어를 설치해야 하는데 아두이노 웹사이트(<http://www.arduino.cc>)에 접속한 후 메인화면 상단의[Download] 메뉴를 클릭하여 다운로드가 가능하다.

아두이노 소프트웨어의 버전은 일정 주기로 업그레이드 되며 설치가 완료되면 아두이노 IDE 소프트웨어 프로그램(스케치 창)이 [그림 2-3] 실행된다. 설치가 완료되면 아두이노 보드와 컴퓨터를 연결해야하는데 아두이노 보드의 USB는 다음과 같이 세가지 기능을 제공한다. 첫 번째는 전원을 공급 받을수 있고 두 번째는 컴파일한 프로그램을 업로드할 수 있으며 세 번째는 시리얼 포트를 통해 디버깅 메시지를 볼 수 있다. 따라서 아두이노 보드는 USB케이블 하나로 컴퓨터에 연결될 수 있으며 아주 간단한 인터페이스를 이용하여, 개발을 진행할 수 있는 장점이 있다. USB가 연결되면 아두이노 보드가 인식될 수 있도록 아두이노 보드용 드라이버를 설치하는데 제어판의 장치관리자에서 드라이버 소프트웨어 업데이트를 통해 설치가 가능하고 추가된 포트 번호를 통해 아두이노 플랫폼을 제어할 수 있다.

3.2.2 아두이노 스케치

스케치는 크게 `setup()`메서드와 `loop()`메서드 두 파트로 구분된다. `setup()`은 장비의 초기 설정을 작성하고, `loop()`에는 반복 실행되는 내용을 작성한다. 즉 `setup()` 함수는 프로그램 실행 시 처음 한번만 실행되는 함수이고, `loop()` 함수는 프로그램을 계속 반복시키는 역할을 한다. 아두이노 스케치의 라이프 사이클은 [그림 3-2]와 같다.



[그림 3-2] 아두이노 스케치 라이프 사이클

[그림 3-2]와 같이 setup메서드에 하드웨어를 제어하기 위한 초기화 루틴을 작성할 수 있고 loop메서드는 보드가 리셋 될 때까지 반복해서 수행되는 루틴을 작성할 수 있다. setup 소스코드는 [그림 3-3]과 같다.

```

SoftwareSerial blueToothSerial(Rx0, Tx0);

void setup()
{
  Serial.begin(9600);
  pinMode(Rx0, INPUT);
  pinMode(Tx0, OUTPUT);

  pinMode(vibe, OUTPUT);
  pinMode(sound, OUTPUT);
  pinMode(led, OUTPUT);

  pinMode(button, INPUT);

  setupBluetoothConnection();
}
  
```

[그림 3-3] 아두이노 setup() 소스코드

시리얼 클래스의 begin 메서드를 통해 디버깅 메시지를 보게 될 시리얼 디바이스의 통신 속도를 9600bps로 설정 했으며 pinMode를 통해 led, button, sound, vibrate, buzzer 디바이스와 setupBluetoothConnection 함수를 통해 블루투스 디바이스를 초기화 했다.

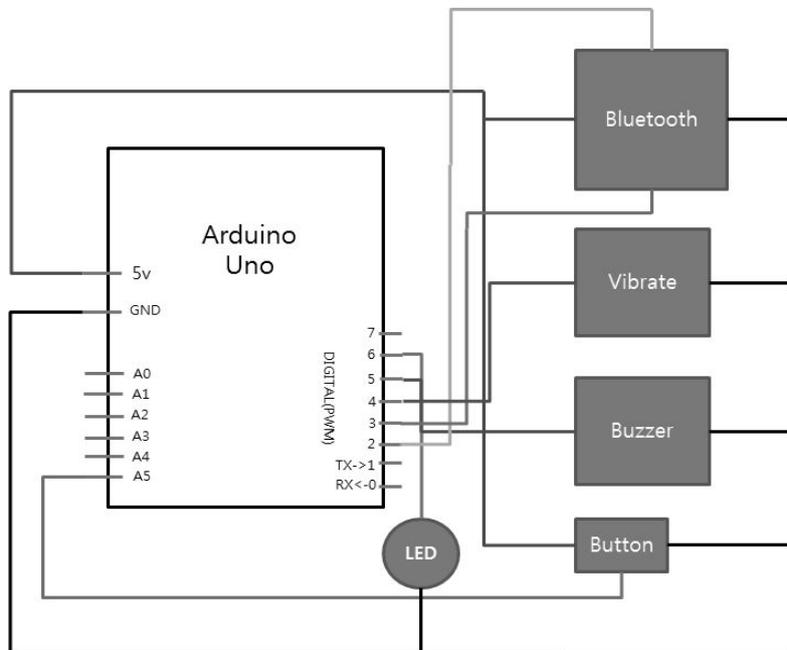
본문에 해당하는 Loop메서드는 안드로이드와 블루투스통신을 하기 위해 페어링이 된 경우 안드로이드에서 전송한 문자열을 읽어 들임으로서 해당 문자열에 따라 진동이나 led 발광, 사운드 등의 처리를 할 수 있도록 작성 되었다. 해당 루프문은 아두이노 플랫폼이 리셋되기 전까지 반복적으로 동작하여 안드로이드로부터 전송되는 문자열을 반복적으로 수신함으로써 자체 디바이스를 컨트롤 할 수 있다.

3.2.3 아두이노 하드웨어 설계

본 논문에서 제안한 아두이노 플랫폼에서 사용된 하드웨어는 아두이노 우노 R3, 피에조 부저, 진동 장치, led, 버튼, 블루투스 모듈(HC-05), 그리고 점퍼케이블 등이 있다. 먼저 블루투스과 아두이노의 연결방법은 아래와 같다.

- 1) Bluetooth를 breadboard 에 연결한다.
- 2) Bluetooth의 VCC는 Arduino의 5[V], Bluetooth의 GND는 Arduino의 접지에 연결한다.
- 3) Bluetooth의 CFG를 해제 한다.
- 4) Serial communication을 위해 Bluetooth의 TXD는 Arduino의 D2에, Bluetooth의 RXD는 Arduino의 D3에 연결한다.
- 5) 정상적으로 연결되면 Bluetooth쪽 LED가 1초에 4번 깜박인다.

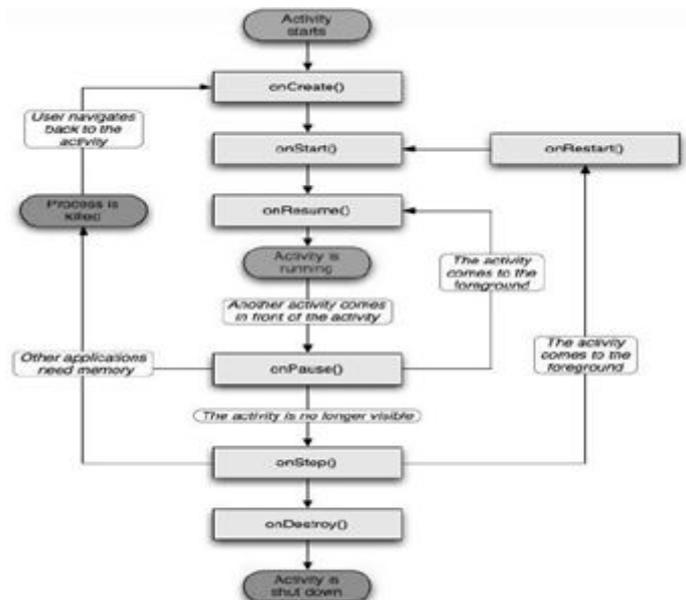
블루투스 모듈이 안드로이드와 통신함으로서 아두이노 플랫폼에 알람, 진동, 발광 등의 동작을 수행시킬 수 있다. 이중 대표적으로 소형진구가 발광 되는 플로우를 살펴보면 LED(발광다이오드)는 전류가 흐르면서 빛을 발하게 되는데 크기나 색깔, 동작 전압에 따라 여러 가지 형태의 LED가 존재한다. LED를 통과하는 전류는 오직 한 방향으로서 회로에 연결할 때 방향을 고려하여 긴 쪽 다리를 전원의 양극(VCC), 짧은 쪽 다리를 음극(GND)으로 연결한다. 추가로 피에조 부저로 경고음을 내기 위해선 피에조 부저의 검은색 선을 GND에 빨간색 선을 5번 핀에 연결하면 LED가 발광했던 방식을 통해 경고음이 발생한다. [그림 3-4]의 회로도를 통해 아두이노 플랫폼의 하드웨어 Making을 확인할 수 있다.



[그림 3-4] 아두이노 회로도

3.3 안드로이드 어플리케이션 설계

Android Application은 Task의 집합으로 안드로이드에서는 이 Task를 Activity라 지칭한다. 즉 Activity는 Application이 수행하는 고유한 하나의 Task라 정의할 수 있는데 이 Activity의 Life Cycle을 살펴보면 [그림 3-5]와 같다.



[그림 3-5] Activity의 Life Cycle

[그림 3-5]의 Life Cycle를 살펴보면 onCreate()를 통해 하나의 activity가 처음 시작되면서 초기 설정작업을 진행하게 된다. 이후 onStart() 메서드를 통해 Activity가 화면상에 나타나게 되며 onResume()메서드를 통해 Activity 실행에 필요한 자원(음악, 애니메이션)을 준비시키게 된다. 다른 Activity가 스택의 최상위에 호출되는 경우 현재 Activity가 Background로 밀려나게 되는데 이때 onPause()메서

드가 호출되며 멈추거나, Activity를 파괴 하는 과정을 통해 Activity의 Life Cycle이 종료 된다.

3.3.1 AndroidManifest.xml 설계

초기 화면에 뿌려질 Main Activity는 AndroidManifest.xml 파일에서 설정 할 수 있는데 이밖에 안드로이드 어플리케이션이 자체 디바이스에 접근할 수 있는 모든 권한도 설정할 수 있다. 본 연구에서 제안한 AndroidManifest 파일을 살펴보면 다음과 같다.

```
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.CALL_PHONE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.VIBRATE" />
```

위 소스 코드와 같이 uses-permission 을 통해서 디바이스에 대한 권한을 설정할 수 있는데 저장소에 읽고 쓰는 권한, 전화를 거는 권한, 핸드폰의 진동을 발생시키는 권한 마지막으로 블루투스에 대한 권한을 설정했음을 알 수 있다. 블루투스 권한 설정에 대해서 요약하자면 어플리케이션에서 블루투스 기능을 사용하기 위해서는 먼저 적어도 BLUETOOTH_ADMIN, BLUETOOTH 중 하나는 선언해야 하며, 연결 수락, 데이터 교환, 연결 등의 블루투스 통신을 수행하고자 한다면 BLUETOOTH 권한을 요청해야 한다. 블루투스 설정을 조작하거나 초기 장비를 검색하기 위해서는 BLUETOOTH_ADMIN 권한을 설정해야 하

는데 대부분의 어플리케이션 이 로컬 블루투스 장비들을 검색하기 위해 전적으로 이 권한을 필요로 하며 만약 사용자가 블루투스 디바이스에 대한 설정을 변경하는 슈퍼 관리자가 아니라면 이 권한을 요청할 필요는 없다. BLUETOOTH_ADMIN 권한을 소유하게 되면 BLUETOOTH 권한에서 허가된 기능도 부가적으로 수행할 수 있다. 이밖에 application의 activity에서 초기 페이지를 설정 할 수 있는데 o.s.y 패키지의 MainActivity임을 확인 할 수 있다.

3.3.2 MainActivity 설계

MainActivity를 설계하기 앞서 본 논문에서 제안한 어플리케이션을 구현하기 위해 사용되는 기능을 살펴볼 필요가 있다. 아두이노 플랫폼과 양방향 통신을 하기 위해 Send Signal과 Receive Signal 기능이 필요하고 이밖에 RSSI 통신 감도를 확인하여 실시간으로 신호의 감도를 측정하는 기능과 현재 스마트폰과의 접속 상태를 나타내는 기능이 필요하다. Send Signal과 Receive Signal의 상세 정보는 다음 [표 3-1] 과 같다.

[표 3-1] Send Signal과 Receive Signal의 기능 표

기능명칭	기능명칭	설명
Send Signal	Sound On/Off	아두이노의 경보음을 On/Off 하는 기능
	Vibrate On/Off	아두이노에 진동을 On/Off 하는 기능
	Led On/Off	아두이노 Led를 On/Off 하는 기능
Receive Signal	Call	대상의 핸드폰 연락처를 지정하는 경우 대상 핸드폰으로 신호의 감도가 약해질 경우 전화가 걸리는 기능

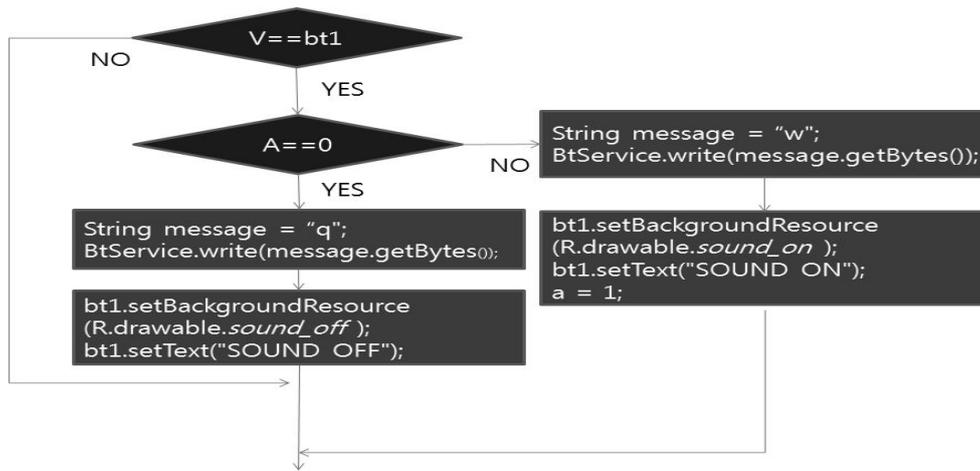
	Sound On/Off	스마트폰에 벨소리를 On/Off 하는 기능
	Vibrate On/Off	스마트폰의 진동을 On/Off 하는 기능
	None	스마트폰을 제어하지 않는 기능

[표 3-1]에서와 같이 Send Signal은 스마트폰이 아두이노 플랫폼을 컨트롤 하는 기능으로서 신호의 세기가 약해지거나 끊길 경우의 대처 동작을 미리 지정하는 것이며 Receive Signal은 이와 반대로 신호의 세기가 약해지는 경우 스마트폰의 대처 동작을 미리 지정하는 것임을 알 수 있다. 다음과 같은 기능을 토대로 해당 어플리케이션의 소스코드를 설계하고 분석하도록 하겠다.

가) 초기화 및 Life cycle 메서드 작성

초기화 하기 위해선 최초에 각종 변수에 대한 선언이 필요하며 그다음 onCreate를 통해 Activity를 초기화 하는 과정을 거쳐야 한다. 휴대폰 알람과 진동과 같이 어플리케이션 상에서 사용하게 될 디바이스들을 초기화 하고 activity의 로딩화면을 설정한 이후 [표 3-1]에서 설정한 기능의 레이아웃을 그려주게 된다.

각각의 버튼과 텍스트 그리고 이미지등의 레이아웃 구성이 끝나면 onClick과 onKeyDown 메서드를 통해 해당 버튼과 텍스트박스 또는 이미지 등이 어떻게 동작해야 할지에 대해 소스코드를 등록한다. 해당 소스코드의 예는 다음과 같다.



[그림 3-5] 어플리케이션 동작 소스코드

아두이노를 컨트롤하는 소스코드로서 아두이노의 사운드를 On/Off 할 수 있도록 작성된 소스코드이다. Sound on 버튼을 클릭하는 경우 아두이노에 “w”스트링을 블루투스 통신을 통해 전달하게 되며 그 이후 안드로이드 어플리케이션은 sound on에 해당하는 이미지를 보여주게 된다. 안드로이드와 아두이노 플랫폼간 페어링이 되어 있는 상황이라는 전제하에 아두이노에서 사운드가 작동하게 된다. 반대로 sound off 버튼을 클릭하는 경우 “q”스트링을 아두이노에 전달함으로써 작동된 사운드의 동작을 멈추게 할 수 있다.

나) 블루투스 설정

Application이 블루투스 통신을 시작하기 전에 먼저 블루투스가 장비에서 활성화 되어 있는지를 먼저 검증해야 한다. 만약 블루투스가 비활성화 되어 있다면 Application을 떠나지 않고 블루투스 디바이스를 활성화 시키도록 요청할 수 있는데, 이런 작업은 BluetoothAdapter를 사용하여 처리한다.

1) BluetoothAdapter 획득

BluetoothAdapter는 모든 액티비티에서 요청할 수 있는데 BluetoothAdapter를 얻기 위해서는 getDefaultAdapter() 메서드를 호출해야 한다. 이 메서드를 통해 장비가 소유중인 BluetoothAdapter에 대한 BluetoothAdapter 객체가 반환되어 application은 이 객체를 사용하여 어댑터와 연동할 수 있다. 만약 getDefaultAdapter()가 null값을 반환하게 되면 장비는 블루투스를 지원하지 않는 것으로 블루투스에 관한 어떠한 작업도 진행될 수 없다.

2) 블루투스 디바이스 활성화

다음으로는 블루투스를 활성화 시켜야 한다. isEnabled() 메서드를 사용함으로써 현재의 상태가 활성화 상태인지 확인할 수 있다. 만약 리턴값이 false이면 블루투스가 비활성화 되어 있는 것으로서, 활성화 상태로 변경하기 위해 ACTION_REQUEST_ENABLE Action Intent와 함께 startActivityForResult()를 호출해야 한다. 이 명령을 이용하여 application이 종료되지 않고 시스템 설정을 통해 블루투스를 활성화시킬 수 있다. 대부분의 어플리케이션과 같이 블루투스를 활성화시키기 위해 사용자의 권한을 요청하는 다이얼로그 창을 생성하여 사용자가 승인하는 경우 시스템은 블루투스를 활성화 시키는 작업을 수행하게 되며 활성화가 완료되면 application으로 다시 복귀하게 된다.

블루투스 활성화가 성공했다면 복귀된 액티비티는 onActivityResult() 메서드에서 리턴값을 RESULT_OK를 받게 되고 만일 "No"를 선택하여 블루투스 디바이스를 활성화 시키지 않았다면 리턴값은 RESULT_CANCELED이 된다. 추가적으로 Application은 시스템 내에서

블루투스의 상태가 변경되는지 그 여부를 알아내기 위해 ACTION_STATE_CHANGED의 BROADCAST INTENT를 받을 수 있도록 지정할 수 있다. 이 BROADCAST는 EXTRA_STATE 와 EXTRA_PREVIOUS_STATE에 대한 정보가 내포되어 있기 때문에 이전상태와 새롭게 설정된 상태를 둘 다 나타낼 수 있다. 이 data에 허용되는 지정 값은 STATE_TURNING_ON, STATE_TURNING_OFF, STATE_ON, STATE_OFF등이 있으며 이 데이터들을 통해 실행중인 Application의 블루투스 상태가 변화되는 것에 대응할 수 있다.

다) RSSI 통신

안드로이드와 아두이노는 블루투스 페어링을 통해 디바이스간 통신이 이루어 진다. 여기에 신호강도 측정법인 rssi API를 추가함으로써 양 플랫폼간 신호의 강도를 실시간으로 측정하여 신호가 약해지는 경우를 사전에 감지하여 핸드폰이 분실되거나 도난 되었을 경우를 미연에 방지할 수 있다.

```
int rssi_signal = intent.getShortExtra(BluetoothDevice.EXTRA_RSSI, Short.MIN_VALUE);
```

위의 소스 코드를 통해 블루투스간 신호강도를 체크하여 실시간으로 화면상에 보여줄 수 있는데 android.bluetooth 패키지 안의 클래스를 상속받아 주기적으로 RSSI 신호강도를 가져 올 수 있다. BluetoothSocket 클래스와 BluetoothServerSocket 클래스를 포함하여 블루투스에서 사용하는 API는 모두 'android.bluetooth'패키지에서 제공하고 있으며, 블루투스에서 사용하는 Socket 통신은 기본 안드로이드 Socket 통신과 사용법

이 같지만, TCP/IP와는 무관한 전혀 다른 블루투스 프레임을 사용하는 Class와 Method라고 볼 수 있다. 이들 중 BluetoothAdapter클래스는 개발자가 안드로이드에서 제공하는 SDK를 이용하여 블루투스를 사용할 수 있도록 거의 모든 기능을 제공하는데 이 클래스를 이용하여 디바이스를 활성화, 비활성화 시키거나 블루투스를 찾기 위한 검색작업을 실행 또는 외부 블루투스의 Device MAC주소를 사용해 BluetoothDevice 인스턴스를 생성할 수 있고 통신에 필요한 BluetoothServerSocket 객체를 만들 수 있다.

```

if (name.equals("HC-06")){
    if (rssi <= -50 ){
        Handler hd = new Handler();
        BluetoothDevice device = btAdapter.getRemoteDevice("30:14:11:10:10:15");
        BtService.connect(device);
        hd.postDelayed(new Runnable() {
            @Override
            public void run() { .....

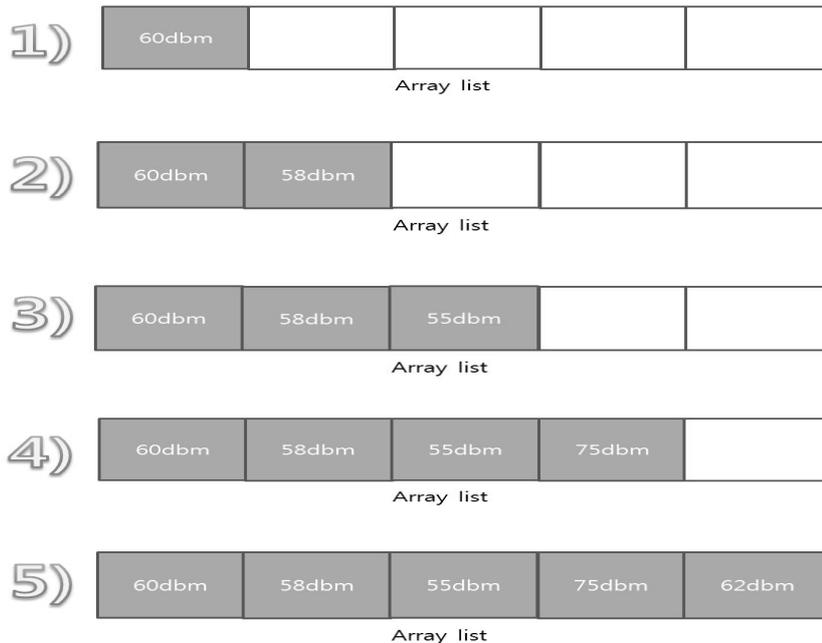
```

본 논문도 다음과 같이 블루투스의 디바이스 MAC주소를 이용하여 Device instance를 생성하였다. 이 메서드가 제공하는 Bluetooth Device 객체는 페어링으로 관리되는 디바이스이므로 현재 Local Bluetooth와 Network로 연결되어 있지 않다 하더라도 제공할 수 있으며 파라미터로 주어진 하드웨어 MAC주소는 위 소스와 같이 대문자로 제공해야한다. 다음과 같은 과정을 통해 플랫폼간 커넥트가 이뤄지며 이후 아두이노 플랫폼의 디바이스를 컨트롤 할 수 있다.

3.3.3 신호간섭 제어 알고리즘 설계

블루투스 통신은 무선네트워크 기기 또는 같은 블루투스 기기를 통해 신호간섭이 발생하기 쉽다. 신호간섭으로 신호의 세기가 튀거나 약해지는 현상이 발생하면 스마트폰 분실방지 어플리케이션의 오작동 확률이 발생함으로 이를 제어하기 위한 알고리즘을 구현 한다. 기본적인 알고리즘은 신호가 튀는 Max Signal 과 신호가 약해지는 Min Signal을 제외한 산술평균을 통해 잡음을 걸러내게 된다. 해당 알고리즘의 상세한 설명은 다음과 같다.

- o 5개의 빈배열에 순차적으로 신호강도를 채운다. 기본적으로 신호강도인 dBm 단위는 음수이기 때문에 Max Signal은 -75dBm, Min Signal은 -55dBm이 되며 이들을 뺀 산술평균은 -59dBm이 된다.



- o Max Signal: -75dBm / Min Signal: -55dBm

신호평균: -60dBm



- o 배열이 가득차는 경우 FIFO방식과 같이 먼저 들어온 값이 제거된다.
- o 새로 들어온 신호가 그 자리를 매꾸게 된다.



- o Max signal : -75dBm / Min Singnal: -55dBm

신호평균: -59dBm

- o 주기적으로 신호가 입력되면 배열을 지워나가며 신호가 배열에 추가되며 가장 강한 신호와 가장약한 신호를 잡음으로 간주하여 신호평균을 구함으로서 잡음을 제거할 수 있다.

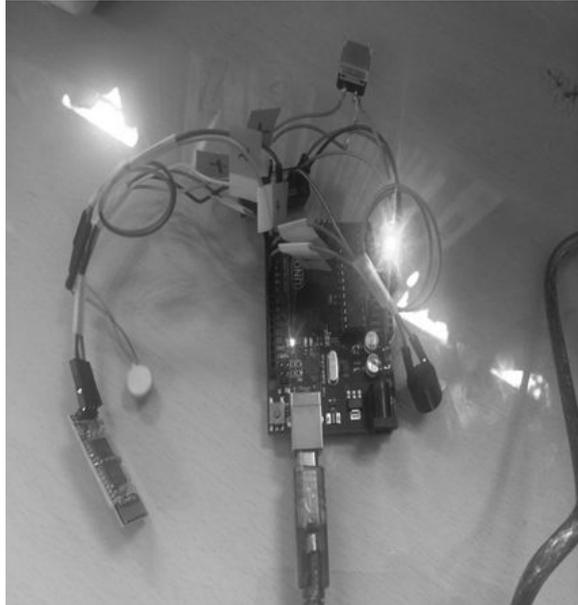
요약하자면 최신의 5개의 신호를 순차적으로 어레이리스트에 담고 그중 가장 강한 신호강도와 가장 약한 신호강도를 제외하여 산술 평균을 구한다. 수신된 새로운 신호값은 기존의 어레이 리스트에서 삭제된 빈 배열에 삽입되며 누적된 신호값을 토대로 다시 산술평균을 하여 튀거나 약한 신호 값을 걸러낸다. 이와 같은 알고리즘을 통해 블루투스통신의 불규칙한 신호강도를 균등하게 벨런싱하여 어플리케이션의 안정성을 높일 수 있다.

제 4 장 구현 및 성능평가

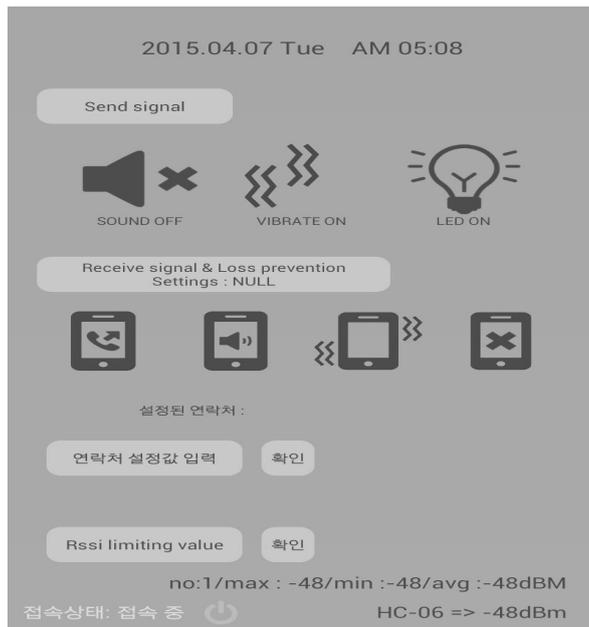
본 장에서는 스마트 분실방지 어플리케이션의 구현과 성능을 평가 한다. 아두이노 플랫폼과 안드로이드 어플리케이션의 구현을 통해 기존의 상용화된 어플리케이션과 비교분석하고 성능평가를 함으로써 제안한 스마트폰 분실방지 어플리케이션을 검증한다.

4.1 구현

3장에서 설계한 회로도를 기반으로 아두이노 플랫폼의 프로토타입을 구현했다. 전원은 USB를 통해 로컬피시로 공급받도록 하였으며 스케치를 통해 소스코드를 작성하고 컴파일하여 아두이노에 적재하였는데, 테스트 시연을 하기 위해 소스코드에 강제로 LED와 기타 디바이스를 ON 하여 작동여부를 테스트 하였다. [그림 4-1]과 같이 아두이노가 정상 작동됨을 확인할 수 있었고 추가적으로 블루투스 모듈을 스마트폰에서 검색하고 페어링 시킴으로써 동작 테스트를 완료하였다. 안드로이드 어플리케이션의 시스템 개발환경은 [표 4-1]과 같다.



[그림 4-1] 아두이노 구현



[그림 4-2] 안드로이드 어플리케이션 구현

[표 4-1] 안드로이드 시스템 개발환경

개발 운영체제	Window 7
Language	Java Development Kit 1.8
Developer Tool	Eclipse JUNO
SDK	SDK 4.4.2
Android Device	Galaxy note 1
Android OS	Android 4.0

안드로이드를 개발하기 위해 가장 대중화된 Java 개발 툴인 이클립스를 사용했다. 어플리케이션의 모의 테스트 결과를 확인하기 위해 갤럭시 노트 1을 Developer 모드로 전환하여 컴파일된 어플리케이션을 적재 하면서 소스코드 작성 및 디버깅 작업을 수행했다. 구현결과는 [그림 4-2]와 같으며 제안내용에 소개한 바와 같이 Send Signal과 Recevie Singnal로 기능을 나누어 아두이노와 안드로이드 플랫폼 간 동작 모드를 지정할 수 있고, 실시간으로 신호강도를 측정할 수 있음을 확인할 수 있었다.

4.2 성능평가

본 절에서는 RSSI를 통한 신호강도 테스트와 본 논문에서 제안된 알고리즘을 이용한 신호강도 테스트를 통해 각각의 성능을 비교 분석 함으로서 신호강도의 정상적인 측정 여부 및 신호간섭이 발생하는지 그 여부를 살펴보고 알고리즘을 통해 이를 제어할 수 있는지 살펴본다.

4.2.1 RSSI 성능 테스트 결과

[표 4-2] 성능측정결과(단위: dBm)

차수	거리	1m	2m	3m	4m	5m	6m	7m	8m
1차	신호 감도	-58	-65	-68	-82	-88	-92	-97	-108
2차	신호 감도	-68	-68	-72	-80	-89	-91	-99	-111
3차	신호 감도	-44	-69	-73	-79	-87	-92	-101	-108
4차	신호 감도	-55	-66	-73	-81	-91	-94	-101	-109
5차	신호 감도	-55	-69	-86	-84	-106	-95	-99	-108
6차	신호 감도	-58	-65	-74	-83	-87	-92	-100	-112
7차	신호 감도	-53	-68	-71	-79	-84	-93	-99	-106
8차	신호 감도	-52	-66	-74	-82	-86	-75	-98	-106
9차	신호 감도	-54	-65	-74	-84	-90	-93	-101	-108
10차	신호 감도	-57	-66	-72	-81	-87	-89	-102	-114

다음 성능측정은 일반적인 사무실 환경에서 테스트 되었고 다음과 같이 신호감도가 주변 잡음을 통해 약해지거나 강해지는 현상이 나타나는 것을 알 수 있었다. 보다 넓고 전자기기가 더 많은 상황에서 신호간섭이 더 발생할 여지가 많을 것으로 판단되어 본 논문에서 구현한 잡음 제거 알고리즘을 사용하여 2차 테스트를 진행하였다.

4.2.2 잡음제거 알고리즘을 사용한 성능 테스트 결과

[표 4-3] 잡음제거 알고리즘을 통한 성능측정결과(단위: dBm)

차수	거리	1m	2m	3m	4m	5m	6m	7m	8m
1차	신호 감도	-56	-65	-69	-78	-85	-91	-99	-109
2차	신호 감도	-55	-64	-71	-77	-87	-91	-100	-111
3차	신호 감도	-54	-66	-73	-79	-87	-92	-101	-109
4차	신호 감도	-55	-66	-72	-80	-91	-93	-100	-113
5차	신호 감도	-55	-69	-72	-80	-88	-92	-98	-108
6차	신호 감도	-54	-65	-69	-80	-87	-92	-101	-112
7차	신호 감도	-53	-63	-71	-79	-85	-93	-100	-108
8차	신호 감도	-52	-66	-70	-82	-86	-92	-99	-106
9차	신호 감도	-52	-65	-69	-81	-88	-93	-102	-109
10차	신호 감도	-49	-64	-71	-81	-87	-94	-102	-112

잡음제거 알고리즘은 최대신호강도와 최저신호강도를 제외한 신호평균
으로 이뤄지기 때문에 해당 차수별 일정거리에서 신호평균이 이루어지기
까지 일정 시간동안 대기한 후 그 성능을 평가했다. 잡음제거 알고리즘
을 사용한 후 신호가 튀거나 약해지는 현상을 감지함으로서 이를 제거하
여 차수별로 고른 신호 강도가 유지됨을 성능측정결과 확인할 수 있었
다. 다음 알고리즘을 통해 장비의 오작동이 현저하게 줄어들 것으로 판
단되며 감지의 안정성이 더욱 향상됨을 확인 할 수 있었다.

제 5 장 결론

스마트폰이 우리 생활 속에 자리 잡음으로서 스마트폰 보급 속도가 가히 폭발적이지만, 스마트폰의 보급 확대와 함께 도난 및 분실 건수도 급증하고 있는 추세이다. 그러나 스마트폰은 일반적인 휴대폰과 달리 고가이면서 민감한 개인 정보가 대량으로 저장되어 있어, 분실하게 되는 경우 1차적인 금전적인 피해 그리고 각종 모바일 공인인증서를 통해 금융계좌에 접근하여 범죄에 악용될 2차적인 피해, 마지막으로 개인정보 유출과 같은 3차 피해까지 발생할 여지가 다분하다. 본 논문에서 제안한 분실 예방 어플리케이션을 통해 스마트폰의 분실 우려를 최소화함으로써 사용자가 인지하게 될 시간적, 정신적, 물질적 피해를 최소화할 수 있을 것이라고 판단된다. 본 어플리케이션은 상용화를 목적으로 함으로서 100만원을 호가하는 스마트폰의 분실을 예방하기 위한 방편으로 적절한 가격을 책정한다면 시장성이 있다고 판단되지만 블루투스 통신방식의 한계로 사람과 장비가 많은 지역 넓거나 협소한 장소 등에 따라 시시각각변하는 신호의 세기는 현재로서 완벽한 제어가 불가능해 보인다. 하지만 보다 정확성이 높고 전력소모마저 적은 iBeacon을 통한 근거리 무선통신 방법으로 이를 극복해 나갈 수 있을 것으로 보여지며 감지의 안정성과 가격 효율성을 고려한 알고리즘을 개발하여 제품을 추가 구현할 예정이다.

참고문헌

- [1] 김익환, 김태현, “안드로이드 플랫폼에서 응용프로그램의 유연한 권한 설정 기법 설계 및 구현,” KIPS Transactions, 제18C권 3호, pp. 151-155, 2011.
- [2] 김송주, “RSSI를 이용하여 위치인식 가능한 저전력 다채널 무선통신 시스템에 관한 연구,” 전남대학교, 2012.
- [3] 남호찬, 박효빈, 이성주, “분실 예방을 위한 안드로이드와 하드웨어 및 센서를 이용한 알림 시스템,” 대한전자공학회 하계종합학술대회 논문집, 제37권 1호, pp. 309-310, 2014.
- [4] 남춘성, 정현희, 신동렬, “iBeacon을 이용한 스마트 디바이스 분실 방지 시스템,” 인터넷정보학회논문지, 제15권 6호, pp. 27-34, 2014.
- [5] 신용선, “무선랜의 신호세기를 이용한 실내측위 알고리즘 성능비교 시뮬레이터,” 경북대학교, 2009.
- [6] 허병희, “무선 센서 네트워크에서 Gaussian Model을 적용한 측위 향상기법,” 한국산업기술대학교, 2009.
- [7] A. Kotanen, M. Hannikainen, H. Leppakoski, and T. D. Hamalainen, “Experiments on local positioning with Bluetooth,” in Proc. IEEE International Conference Information Technologies: Computer Communications, Apr. 2003.
- [8] B. Davis, B. Sanders, A. Khodaverdian, and H. Chen. “I-arm-droid: A rewriting framework for in-app reference monitors

for android applications,” IEEE Mobile Security Technologies (MOST), San Francisco, CA USA, 2012

[9] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. “The cricket location support system,” in Proc. the 6th Annual ACM International Conference on Mobile Computing and Networking (MOBICOM), Aug. 2000.

[10] R. Xu, H. Saidi, and R. Anderson, “Aurasiom: Practical policy enforcement for android applications,” in Proc. the 21st USENIX Conference on Security Symposium, 2012.

[11] R. Thongthammachari and H. Olesen, “Bluetooth enables indoor mobile location services,” in Proc. IEEE Vehicular Technology Conference, vol. 3, Apr. 2003.

[12] R. Bruno and F. Delmastro, “Design and analysis of a Bluetooth-based indoor localization system,” in Proc. IEEE Personal Wireless Communication, Sept. 2003.