제 출 문

국토교통부 장관 (국토교통과학기술진흥원장) 귀하

이 보고서를 국토교통기술사업화지원사업 과제의 보고서로 제출합니다.

2014. 08.

주관연구기관명 : 인소팩 주식회사
주관연구책임자 : 손 동 철
연 구 원 : 김 나 영
〃 : 박 홍 렬
〃 : 서 정 헌
〃 : 이 명 수
〃 : 신 원 섭
〃 : 이 회 건
〃 : 윤 여 옥
〃 : 노 지 영
〃 : 정 창 현
〃 : 염 민 정
협동연구기관명 : 한국교통연구원
협동연구책임자 : 한 재 현

<table>
<tr><td>과제고유번호</td><td>12기촉-
사업화-17</td><td>해 당 단 계
연 구 기 간</td><td>3년</td><td>단 계 구 분</td><td></td></tr>
<tr><td>연 구 사 업 명</td><td colspan="5">국토교통기술사업화지원사업</td></tr>
<tr><td rowspan="2">연 구 과 제 명</td><td>최 상 위
과 제 명</td><td colspan="4">공항시설 안전통합 관리시스템 전용단말기 개발</td></tr>
<tr><td>단위과제명</td><td colspan="4"></td></tr>
<tr><td>연 구 책 임 자</td><td>손 동 철</td><td>총연구기간
참 여
연구원수</td><td>총 : 11명
내부 : 명
외부 : 명</td><td>총연구비</td><td>정부 :
500,000천원
기업 :
354,925 천원
계 :
854,925 천원</td></tr>
<tr><td>연구기관명 및
소 속 부 서 명</td><td colspan="2">인소팩 주식회사</td><td>참여기업명</td><td colspan="2">한국교통연구원</td></tr>
<tr><td>국제공동연구</td><td colspan="5">상대국명 : 상대국연구기관명 :</td></tr>
<tr><td colspan="4">요약(연구결과를 중심으로 개조식 500자 이내)</td><td>보고서면수</td><td>301</td></tr>
<tr><td colspan="6">1. 기존 상용 내비게이션과 차별화 된 안전성과 특수성을 고려한 공항 전용단말기 개발
2. 공항 내 작업차량에서 신속하게 GPS 정보를 확인하기 위하여 사용이 용이한 7inch 정전압(Capacitive Sensing) 방식의 터치스크린이 탑재된 공항 전용단말기 개발
3. 사용자 편의성을 고려한 외부 연동 모듈의 내부 실장 및 연동 케이블 최소화 구현
4. 전용단말기를 구성하고 있는 각종 부품의 인터페이스의 드라이버 구현
5. 공항 내 존재하는 이동체(항공기, 차량)의 모니터링 및 차량의 위치정보 전송기능 제공
6. 공항 내 주요 상황(안전규정 위반, 사고, FOD(Foreign Object Debris)) 정보 전송기능 제공
7. 라이선스 문제가 없는 최적의 실시간 운영체제(RTOS)와 국산 GIS 엔진이 탑재된 공항시설 안전통합 관리시스템 전용단말기 개발</td></tr>
<tr><td rowspan="2">색 인 어
(각 5개 이상)</td><td>한 글</td><td colspan="4">전용단말기, 차량용 단말기, 안전관리, 내비게이션, GIS 엔진</td></tr>
<tr><td>영 어</td><td colspan="4">Dedicated terminal, Car terminal, Safety Management, Navigation, GIS engine</td></tr>
</table>

요 약 문

Ⅰ. 제 목

공항시설 안전통합 관리시스템 전용단말기 개발

Ⅱ. 연구개발과제 목적 및 필요성

○ 국토교통부는 국내의 G-10 수준의 공항 운영 및 안전관리 기술 향상을 위하여 『항공선진화』 사업의 일환으로 공항시설 안전통합관리 시스템 (Airport Facilities Safety Integrated Management System, 이하 AFSIS-6)을 5년간 개발함

○ AFSIS-6는 이착륙중인 항공기, 계류 중인 항공기, 조업차량 등의 이동체의 현황을 관제소에서 모니터링 할 수 있으며 차량에 장착된 단말기를 통하여 주변의 이동체를 파악하고, 신호등을 제어하여 이동체를 통제하며, 전력시설 및 지적 관리 등의 서비스를 제공하여 활주로 외의 사고를 예방함

○ 그러나 국책과제로 개발이 진행된 AFSIS-6는 시제품(Prototype)수준으로 상용화를 위해서는 QoS가 보장되는 확장성, 편리성, 기술성을 고려한 개선 작업 및 안정화 작업이 필요함

○ AFSIS-6를 상품화하여 수출하거나 타 공항에 설치할 경우 가장 문제가 되는 것은 공항내 조업차량에 장착되는 단말기로, 전체 시스템에서 가장 큰 비중을 차지하지만 여건상 개발을 하지 않고 UMPC와 같은 상용 제품을 사용함에 따라 비용문제를 안고 있으며, 상품의 경쟁력과 부가가치를 떨어뜨리는 요인이 되고 있음

※ 김포공항의 경우 이동체 차량은 2,000대, 인천공항의 경우 20,000대 임

○ 본 사업으로 공항 내 작업차량에서 전용으로 사용할 수 있는 단말기를 개발하여 AFSIS-6의 국내 및 해외 수출 시 비용측면에서 경쟁력을 갖추고 상대적인 부가가치가 발생하도록 함

○ 전용단말기는 일반 상용 내비게이션과 같이 속도 및 위치 표시 등의 일반 기능을 제공하며 공항 내 안전관리를 위한 전용 기능을 포함하여 범용성과 특수성을 동시에 확보할 수 있음

Ⅲ. 연구개발의 내용 및 범위

본 사업을 통하여 라이선스 문제가 없는 실시간 운영체제(RTOS)와 국산 GIS 엔진이 탑재된 공항시설 안전통합 관리시스템 전용단말기의 연구개발을 수행하였으며 연구개발

내용 및 범위는 다음과 같음

1. 공항시설 안전통합 관리시스템 전용단말기의 하드웨어 개발

- ○ 차량에서 신속하게 GPS 정보를 확인하기 위하여 사용이 용이한 7inch 정전압(Capacitive Sensing) 방식의 터치스크린 탑재
- ○ 배터리, 터치스크린, 안테나, 무선부 및 주제어부 등을 위한 메인보드와 외부 케이스를 구성하고 내부 배치 및 기구의 내구성 고려
- ○ 사용자 편의성을 고려한 외부 연동모듈의 내부 실장 및 케이블 최소화 구현
- ○ 기존 상용 내비게이션과 차별화 된 안전성과 특수성을 고려한 설계 반영

2. 공항시설 안전통합 관리시스템 전용단말기의 소프트웨어 개발

- ○ 전용단말기를 구성하고 있는 각종 부품의 인터페이스에 대한 드라이버 구현
- ○ 안드로이드 운영체제 기반의 국산 GIS 엔진 개발 및 최적화
- ○ 공항 내 존재하는 이동체(항공기, 차량)의 위치를 화면에 표출하여 사용자가 모니터링 할 수 있고, 통합서버(미들웨어)에 차량의 위치정보 전송기능 구현
- ○ 공항 내 주요 상황(안전규정 위반, 사고, FOD(Foreign Object Debris)) 정보를 운영센터와 공유할 수 있도록 송수신 인터페이스 및 기능 제공

3. 공항시설 안전통합 관리시스템 관련 정책적 지원 및 표준화 방안 수립

- ○ 기 구축된 테스트베드의 원활한 활용을 위한 유관기관과의 정책적 연계 지원
- ○ 유관 정부부처와의 협의를 통한 기술 활성화 도모
- ○ 유관 정부부처와의 협의를 통한 국내 표준화 방안 분석

Ⅳ. 연구개발 수행내용

1. 공항시설 안전통합 관리시스템 전용단말기 요구사항 도출

- ○ 운용성, 사용성, 지속성, 호환성을 고려한 주요 요구사항 도출
  - 차량 내 전용단말기 설치 후 작업자의 시야확보가 용이하도록 개발

- 내비게이션 장비와 유사하게 거치 또는 매립이 가능한 크기로 개발
- 라이선스 문제가 없는 최적의 실시간 운영체제와 국산 GIS 엔진을 적용하여 개발
- GPS, Wi-Fi 통신장치를 내장형으로 제작하여 복잡한 선으로 인해 운용 중 연결이 해제되지 않도록 개발
- GPS, Wi-Fi 통신장치의 외장형 안테나 연결 커넥터를 제공하여 무선 성능 저하 방지
- 전용 하드웨어와 운영체제를 사용하여 지속적인 운영 및 유지보수가 가능하도록 함

○ 요구사항에 따른 하드웨어 및 소프트웨어 설계

- 하드웨어는 메인부(Main Part), 입력부(Input Part), 출력부(Output Part), 전원부(Power Part)로 구성하고, 디자인은 상용 내비게이션 형태를 반영하였으며 보드의 회로, 아트워크(Artwork), 레이아웃(Layout) 설계를 통한 제작 진행
- 소프트웨어는 외부 인터페이스를 위한 패킷 송수신 구조, GIS 등의 데이터 가공을 위한 처리 모듈, 사용자 인터페이스(GUI) 설계 진행

2. 전용단말기의 공급 및 유지보수를 위해 수명주기가 길고 운영체제의 공급 및 유지보수 지원이 가능한 하드웨어 개발

○ 전용단말기 하드웨어 개발을 위한 주요부품 선정

○ LCD, 메인보드, 배터리 사이즈에 맞는 시제품 디자인 설계 및 목업

○ 전용단말기 디자인 개발

○ 기구/보드 설계 및 시제 제작

○ 전용단말기의 각 모듈별 기능 검증 및 성능 안정화

○ 자체 및 공인인증기관을 통한 성능 검증

3. 라이선스 문제가 없는 안드로이드 운영체제 기반, GIS 엔진이 탑재된 공항시설 안전통합관리시스템 전용단말기의 소프트웨어 개발

○ 안드로이드 운영체제 포팅 및 최적화

○ 전용단말기 응용프로그램 GUI 개발 및 기능구현

○ 국산 GIS 엔진 개발 및 관련 기능구현

4. 김포공항 테스트베드 구축 및 운용시험

○ 1~4차에 걸친 김포공항 현장시험 외 GPS, Wi-Fi 성능시험, 단위 및 통합시험 진행

○ 현장시험 완료 후 2013년 12월에 한국공항공사의 제설차, 진에어 외 4개 조업사에 전용 단말기 10대 설치 완료 및 운용시험 진행

5. 공항시설 안전통합 관리시스템 관련 정책적 지원 및 표준화 방안 수립

○ 기 구축된 테스트베드의 원활한 활용을 위한 유관기관과의 정책적 연계 지원

○ 유관 정부부처와의 협의를 통한 기술 활성화 도모

○ 유관 정부부처와의 협의를 통한 국내 표준화 방안 분석

Ⅴ. 연구개발 결과 및 효과

본 사업을 통한 연구개발 결과물(최종 성과물)의 형상과 전용단말기 개발을 통해 발생되는 기술적, 경제적 효과는 다음과 같음



○ 기존의 고비용 라이선스를 사용하는 소프트웨어로 구성된 관리시스템 장비에서 라이선스 비용을 지불하지 않고 순수 국산 하드웨어와 소프트웨어 기술이 적용된 공항시설 안전통합 관리시스템 전용단말기의 양산에 기초를 다짐

○ 본 사업을 통하여 차량에 설치되는 공항 전용단말기의 비용을 약 100만원으로 절감

○ 현재의 운영 상황보다 개선된 효율성, 체계적인 인력과 장비에 대한 관리 가능 및 비용 절감 등 긍정적인 효과 기대

○ 우리나라의 경우 20년간(1980년 ~ 1999년)의 항공사고는 45건으로 이로 인한 총 비용은 1조 1,150억원으로 항공기 사고 1건을 예방할 수 있다면 약 250억 이상의 직/간접비용을 절감 가능

Ⅵ. 연구개발 성과물의 활용계획

○ 본 사업을 통하여 국토해양 R&D의 「항공선진화사업」 일환으로 개발한 「공항시설 안전통합 관리시스템(AFSIS-6)」의 김포공항 테스트베드를 지속적으로 활용할 수 있도록 지원 가능

○ 연구개발이 완료된 전용단말기를 적용한 AFSIS-6를 공항 외 타 분야에도 이용 가능

- 대규모 기간산업시설의 이동체 안전관리 분야에 활용
- 대규모 기간산업시설의 도로 파손 및 이물질 관리 분야에 활용
- 산업단지 내의 위험물 취급 차량 등 특수 차량관리 분야에 활용
- 대규모 기간산업시설의 사고 발생 시 비상상황 대응 지원 분야에 활용

SUMMARY

Ⅰ. Project Title

The Terminal Development for Airport Facilities Safety Integrated Management System

Ⅱ. Objectives and Necessities of the Project

- AFSIS-6(Airport Facilities Safety Integration System), a ground surveillance system, of preventing collisions between aircraft and GSE(Ground Service Equipment) vehicle in airside of an airport has been developed by using the location information with DGPS technology.
- The safety management system for moving object in airside has been designed, and it provides real time services in airside such as surveillance for GSE vehicles, automatic traffic signal control, management of FOD(Foreign Object Debris). etc.
- However, It is necessary to improve and stabilize the operation because the trial AFSIS-6 levels for the commercial.
- When exporting to other airports or installing AFSIS-6, the biggest problem is a terminal that is mounted in the vehicle operation in the airport.
- Vehicle Terminal is the largest proportion in AFSIS-6, but the price is more expensive to use the UMPC device.(installation costs increase)
- Through this project, the development of a device that can be used as a private vehicle in the airport operations. Increase the value and competitiveness when exporting AFSIS-6.
- Terminal is provides general functions such as speed and position, as general navigation also terminal is includes features a safety management for the airport. Including the ability for the airport safety device has the versatility and specificity.

Ⅲ. The contents and range of research and development

The device was developed with real-time operating system(RTOS) and domestic GIS Engine. Research and development content and range of the following equality.

1. Hardware Development

- ○ Check the GPS information quickly through 7inch(Capacitive Sensing) with touch screen
- ○ Consider module placement and the durability of the instrument
- ○ Built-in communication modules and cables to minimize
- ○ Consider the safety and specificity of differentiated and existing commercial navigation

2. Software Development

- ○ Implementation of the various components drivers
- ○ Domestic development of Android operating system-based and GIS Engine
- ○ Provides monitoring of aircraft and vehicle
- ○ Vehicle position information to the integrated server(middleware) transfer function implemented
- ○ Provides the interface and features that can be shared with the airport operations center in vehicle

3. Political support and establish standardized methods

- ○ Political support for the test-bed utilization
- ○ Enabling technology plan through consultation with relevant government departments
- ○ Analysis of national standardized measures in consultation with relevant government departments

Ⅳ. Research and Development

1. Deriving the main requirements

- ○ Considering the continuity, compatibility, Interoperability, usability
  - Developed to facilitate the visibility of the operator
  - Development of this size can be mounted or embedded

- Real-time operating system and domestic GIS engine applies
- Prevent release of the cables connected to the module with built-in GPS and Wi-Fi
- Wireless performance degradation prevention
- Should enable continued operation and maintenance

○ Hardware and software design

- Hardware consists of a Main Part, Input Part, Output Part, Power Part
- Design reflect like the commercially available navigation
- Circuit, artwork, layout design
- Software is consisted a packet transmission and reception structure, GIS data processing module, user interface(GUI) Design

2. Hardware development as a long life cycle maintenance support for the supply and maintenance

○ Major components selected for the development of dedicated hardware devices

○ Design and mock-up prototype design

○ Terminal design development

○ Board design and prototype production

○ Each functional verification and performance stability of the device-specific module

○ Performance verification by the certification institution and itself

3. Software development(equipped with the Android operating system and the domestic GIS Engine)

○ Android OS porting and optimization

○ GUI Development and Implementation

○ Domestic GIS Engine Development and Implementation

4.. Gimpo airport built a test bed and test operation

- Gimpo field test & GPS, Wi-Fi performance testing, unit and integration testing progress
- terminal installation and operational testing completed

5. Political support and establish standardized methods

- Political support for the test-bed utilization
- Enabling technology plan through consultation with relevant government departments
- Analysis of national standardized measures in consultation with relevant government departments

Ⅴ. R&D's result and effect



Technical and economic effects that occur through the development of the terminal is equal to.

- Production is possible without having to pay a license fee to apply to local hardware and application software technology.
- Can be reduced to about 1,000,000 won the cost of the device, which is installed in a vehicle.
- The positive effects can be expected according to the efficiency and cost reduction, etc.
- The aircraft accident that occurred in our country 20 years 45 cases. Can be reduced at least about 25 billion direct/indirect costs, if one can prevent the aircraft accident.

Ⅵ. R&D Utilization and application of planning

- AFSIS-6 has the common platform expandable to the applications in other fields. The system has been tested in the airport environment successfully.
- It is expected that the safety management system development for the application in an airport could be used for the similar applications in the fields of military and security as well.

< CONTENTS >

Chapter 1. R&D Overview ····· 28

Section 1. R&D background and necessity ····· 28

1. R&D Promotion background ····· 28
2. Need for R&D ····· 29

Section 2. Objectives and scope of R&D ····· 30

1. Define R&D ····· 30
2. R&D content and scope ····· 31

Section3. National and international technical developments ····· 32

1. National and international trends and analysis ····· 32
2. Comparison with similar systems ····· 34

Chapter 2. R&D content and results ····· 38

Section 1. Strategy and Methodology ····· 38

1. Configuration for the promotion ····· 38
2. Strategies for Development ····· 39
3. Ability of researchers ····· 46

Section 2. Research process and content ····· 47

1. Annual performance goals ····· 47
2. R&D Highlights ····· 49
3. Final results ····· 231

Section 3. Criteria and verification methods ····· 232

Chapter 3. Final study results and apply results ····· 235

Section 1. The main research results ····· 235

1. Hardware Development ····· 235
2. Software Development ····· 237
3. Results information through research and development ····· 238

Section 2. Testbed deployment and testing ····· 240

1. Overview ····· 240
2. Gimpo airport building testbed ····· 242
3. Test for the actual operating ····· 248

Section 3. A view of the domestic and overseas market ····· 279

1. Definition and market ····· 279
2. Potential for application ····· 285
3. Demand for technology & Expected outcomes ····· 292

Chapter 4. Achieve the effect of the research objectives ····· 293
Section 1. Achievement of the final goal of R&D ····· 293
Section 2. Effect of R&D Results ····· 296
1. Technical effects ····· 296
2. Economic effects ····· 296
3. Policy effects ····· 298

Chapter 5. Utilization and the need for further research ····· 299
Section 1. How to use the results ····· 299
1. Utilization of domestic and foreign ····· 299
2. The need for further studies ····· 303
Section 2. Strategies for Commercialization ····· 304
1. Plans for commercialization ····· 304
2. Strategies for Commercialization ····· 305
3. Configuration for commercialization ····· 307

[References] ····· 308

[Appendix] ····· 309

< 목 차 >

1장. 연구개발과제 개요 ····· 28
1절. 연구개발 배경 및 필요성 ····· 28
1. 연구개발의 추진배경 ····· 28
2. 연구개발의 필요성 ····· 29
2절. 연구개발의 목표 및 범위 ····· 30
1. 연구개발의 정의 ····· 30
가. 연구개발 최종목표 ····· 30
나. 최종목표의 성격 및 설정 근거 ····· 30
2. 연구개발 내용 및 범위 ····· 31
3절. 관련분야의 국내외 기술개발 현황 ····· 32
1. 국내외 기술개발 동향 및 분석 ····· 32
2. 유사 시스템과의 비교 ····· 34

2장. 연구개발 수행 내용 및 결과 ····· 38
1절. 연구수행 전략 및 방법론 ····· 38
1. 추진체계 ····· 38
2. 기술개발 추진전략 ····· 39
가. 기술개발 추진과정 ····· 39
나. 도구를 활용한 체계적인 사업관리 ····· 40
다. 세부 추진계획 및 방법 ····· 41
3. 연구 추진역량(주관연구기관의 연구진 인적 능력) ····· 46
2절. 연구수행과정 및 내용 ····· 47
1. 연차별 정량적 성과목표 및 성과지표 ····· 47
2. 연구개발 주요내용 ····· 49
가. 공항시설 안전통합 관리시스템 전용단말기 요구사항 도출 ····· 51
나. 공항시설 안전통합 관리시스템 전용단말기 하드웨어 개발 ····· 122
다. 공항시설 안전통합 관리시스템 전용단말기 소프트웨어 개발 ····· 170
3. 최종 성과물 ····· 231
3절. 연구수행 결과 평가기준 및 검증 방법 ····· 232

3장. 최종 연구성과 및 적용실적 ····· 235
1절. 주요 연구성과 ····· 235
1. 전용단말기 하드웨어 개발 ····· 235
가. 기존 문제점 대비 개선 구현 사항 ····· 235
나. 전용단말기의 디자인 진행 ····· 236

다. 전용단말기의 기구/보드 설계 및 시제 제작 ····· 236
2. 전용단말기 소프트웨어 개발 ····· 237
3. 연구개발을 통한 성과 정보 ····· 238
2절. 테스트베드 구축 및 운용시험 ····· 240
1. 개요 ····· 240
2. 김포공항 테스트베드 구축 ····· 242
가. 공항 내 실제 작업차량에 전용단말기 설치 ····· 242
나. 운영센터 설치 ····· 244
다. 전용단말기 사용현황 및 불편사항 문의 ····· 245
3. 현장시험 및 실제 운용시험 진행 ····· 248
가. 전용단말기 현장시험 (1~4차) ····· 248
나. GPS 및 Wi-Fi 성능시험 ····· 269
다. 국토교통부와의 AFSIS-6 연구개발 검토 회의 및 시연 ····· 275
라. 국교통기술사업화지원사업 현장평가 진행 ····· 277
3절. 기술개발 성과의 국내외 시장전망 ····· 279
1. 목표시장의 정의 및 시장규모 ····· 279
가. 목표시장의 정의 ····· 279
나. 예상 시장규모 ····· 283
2. 현장적용 가능성 ····· 285
가. 네트워크 인프라 측면 ····· 285
나. 서버 적용 측면 ····· 289
다. 공항시설 안전통합 관리시스템 전용단말기 측면 ····· 289
3. 기술 수요 및 추가 예상 성과 ····· 292

4장. 연구목표 달성 및 효과 ····· 293
1절. 연구개발 최종목표 달성도 ····· 293
2절. 연구개발 성과의 효과분석 ····· 296
1. 기술적 효과분석 ····· 296
2. 경제적 효과분석 ····· 296
3. 정책적 효과분석 ····· 298

5장. 연구성과의 활용 및 추가연구 필요성 ····· 299
1절. 연구성과 향후 활용방안 ····· 299
1. 국내외 활용방안 ····· 299
2. 추가연구 필요성 ····· 303
2절. 사업화 추진전략 ····· 304
1. 사업화 추진계획 ····· 304

2. 사업화 추진전략 ····· 305
가. 공항 운영기관과 협력관계 확보 ····· 305
나. 컨소시엄 구성 ····· 305
다. 전시회 및 워크샵 참여 ····· 305
라. ICAO의 USOAP를 통한 이미지 마케팅 ····· 306
마. 국가 정책지원 활용 ····· 306
3. 사업화 추진체계 ····· 307

[참고문헌] ····· 308

[부 록 - 프로그램 등록증 3부] ····· 309

<표 목차>

[표 1] 최종목표의 성격 및 설정 근거 ····· 30
[표 2] 국외에서 연구가 진행 중인 경쟁기술 ····· 32
[표 3] 공항시설 안전통합 관리시스템(AFSIS-6)의 6가지 서비스 ····· 35
[표 4] 사업화 대상기술과 경쟁기술 비교 및 분석 ····· 35
[표 5] 1차년도 세부추진 계획표 ····· 41
[표 6] 2차년도 세부추진 계획표 ····· 43
[표 7] 3차년도 세부추진 계획표 ····· 45
[표 8] 정량적 성과목표 및 성과지표 ····· 47
[표 9] 연구개발 주요내용 ····· 49
[표 10] 전용단말기 주요 요구사항 ····· 51
[표 11] 전용단말기의 기능적 요구사항 ····· 52
[표 12] 파트별 모듈 구성 ····· 54
[표 13] 소프트웨어의 상태별 설명 ····· 54
[표 14] 소프트웨어 모듈별 역할 및 데이터 처리 과정 ····· 55
[표 15] 패킷 수신 모듈의 순서도 및 기능설명 ····· 58
[표 16] 패킷 파싱 모듈의 순서도 및 기능설명 ····· 59
[표 17] 데이터 파싱 모듈의 순서도 및 기능설명 ····· 60
[표 18] 소프트웨어 점검 모듈의 순서도 및 기능설명 ····· 61
[표 19] 로그인 모듈의 순서도 및 기능설명 ····· 62
[표 20] 패킷 오류 검출 모듈의 순서도 및 기능설명 ····· 63
[표 21] 패킷 오류 제어 모듈의 순서도 및 기능설명 ····· 64
[표 22] 자차 위치 표출 모듈의 순서도 및 기능설명 ····· 65
[표 23] 타 차량 위치 표출 모듈의 순서도 및 기능설명 ····· 66
[표 24] 항공기 위치 표출 모듈의 순서도 및 기능설명 ····· 67
[표 25] 운영센터 메시지 분석 모듈의 순서도 및 기능설명 ····· 68
[표 26] FOD 등록 처리 모듈의 순서도 및 기능설명 ····· 69
[표 27] FOD 처리 모듈의 순서도 및 기능설명 ····· 70
[표 28] 사고지점 등록 처리 모듈의 순서도 및 기능설명 ····· 71
[표 29] 사고 처리 모듈의 순서도 및 기능설명 ····· 72
[표 30] 속도위반 확인 모듈의 순서도 및 기능설명 ····· 73
[표 31] 항공기 근접 확인 모듈의 순서도 및 기능설명 ····· 74
[표 32] 기동지역 침범 확인 모듈의 순서도 및 기능설명 ····· 75
[표 33] 제·방빙 작업구역 침범 확인 모듈의 순서도 및 기능설명 ····· 76
[표 34] 신호위반 확인 모듈의 순서도 및 기능설명 ····· 77
[표 35] AFSIS-6의 서브시스템 설명 ····· 78

[표 36] 프로토콜 통신절차 ····· 81
[표 37] 프로토콜 구조 ····· 81
[표 38] AFSIS-6의 서비스 종류 ····· 83
[표 39] 서브시스템별 서비스 명령어 흐름 ····· 83
[표 40] 차량 위치정보 패킷 유형 ····· 85
[표 41] 항공기 위치정보 패킷 유형 ····· 85
[표 42] 위반 등록 및 출동요청 패킷 유형 ····· 85
[표 43] 위반 사항에 대한 경고 패킷 유형 ····· 86
[표 44] 출동요청, 유지보수, 출동지시 및 조업승인에 대한 확인 패킷 유형 ····· 86
[표 45] 위반 조치완료 패킷 유형 ····· 86
[표 46] FOD/포장면 파손 등록 패킷 유형 ····· 87
[표 47] 유지보수 지시 패킷 유형 ····· 87
[표 48] 유지보수 조치완료 패킷 유형 ····· 87
[표 49] 사고 등록, 상황전파 및 출동지시 패킷 유형 ····· 88
[표 50] 사고 관련 지원요청 패킷 유형 ····· 88
[표 51] 사고 조치완료 패킷 유형 ····· 88
[표 52] 제·방빙 조업 시작요청 패킷 유형 ····· 89
[표 53] 제·방빙 조업승인, 조업 완료보고 패킷 유형 ····· 89
[표 54] 소프트웨어 버전 확인 요청 패킷 유형 ····· 89
[표 55] 소프트웨어 버전 확인 결과 패킷 유형 ····· 90
[표 56] 로그인 확인 요청 패킷 유형 ····· 90
[표 57] 로그인 확인 결과 패킷 유형 ····· 90
[표 58] 차량 위치정보 인터페이스 정의 ····· 92
[표 59] 항공기 위치정보 인터페이스 정의 ····· 93
[표 60] 위반 등록 인터페이스 정의 ····· 94
[표 61] 위반 사항에 대한 경고 인터페이스 정의 ····· 95
[표 62] 위반 출동요청 인터페이스 정의 ····· 96
[표 63] 위반 출동요청 확인 인터페이스 정의 ····· 97
[표 64] 위반 조치완료 인터페이스 정의 ····· 98
[표 65] 로그인 확인 요청 인터페이스 정의 ····· 99
[표 66] 로그인 확인 결과 인터페이스 정의 ····· 100
[표 67] 신호제어 인터페이스 정의 ····· 102
[표 68] FOD/포장면 파손 등록 인터페이스 정의 ····· 104
[표 69] 유지보수 지시 인터페이스 정의 ····· 105
[표 70] 유지보수 지시 확인 인터페이스 정의 ····· 106
[표 71] 유지보수 조치완료 인터페이스 정의 ····· 107
[표 72] 제·방빙 조업 시작요청 인터페이스 정의 ····· 109

[표 73] 제・방빙 조업승인 확인 인터페이스 정의 ····· 110
[표 74] 제・방빙 조업 완료보고 인터페이스 정의 ····· 111
[표 75] 사고 등록 인터페이스 정의 ····· 113
[표 76] 사고 상황전파 인터페이스 정의 ····· 114
[표 77] 사고출동 지시 인터페이스 정의 ····· 115
[표 78] 사고 출동지시 확인 인터페이스 정의 ····· 116
[표 79] 사고 관련 지원요청 인터페이스 정의 ····· 117
[표 80] 사고 조치완료 인터페이스 정의 ····· 118
[표 81] 소프트웨어 버전 확인 요청 인터페이스 정의 ····· 120
[표 82] 소프트웨어 버전 확인 결과 인터페이스 정의 ····· 121
[표 83] 주요부품 선정 목록 ····· 122
[표 84] 부품별 위치명(Reference) 및 품명(Part Number) ····· 123
[표 85] 1차년도 기구 디자인 형상 ····· 129
[표 86] 전용단말기의 실제 사이즈 도면 제작 ····· 130
[표 87] EXYNOS4412 모듈과 메인보드 연결 파트의 회로도 ····· 131
[표 88] 전원부(POWER), CAM 모듈부, KEY, LED 파트의 회로도 ····· 132
[표 89] USB Host 파트의 회로도 ····· 132
[표 90] UART, SD PORT, Wi-Fi, GPS, MINIUSB 파트의 회로도 ····· 133
[표 91] 3.3V Voltage Trans, MINIUSB 파트의 회로도 ····· 134
[표 92] LAN 파트의 회로도 ····· 134
[표 93] AUDIO 파트의 회로도 ····· 135
[표 94] LCD 및 Touch 연결 파트의 회로도 ····· 135
[표 95] LCD 파트의 회로도 ····· 136
[표 96] Wi-Fi 파트의 회로도 ····· 136
[표 97] 센서 파트의 회로도 ····· 137
[표 98] CHARGER, DC TRANSFORMER, POWER KEY 파트의 회로도 ····· 137
[표 99] 층별 신호 라인 배치도 ····· 138
[표 100] 레이아웃 모듈 구성 ····· 139
[표 101] 전용단말기의 보드 구성 모듈 및 설명 ····· 140
[표 102] 주제어부의 모듈 설명 ····· 141
[표 103] 무선부의 GPS 모듈 설명 ····· 142
[표 104] 무선부의 Wi-Fi 모듈 설명 ····· 143
[표 105] 디자인 제작 시 요청사항 ····· 144
[표 106] 상용 스마트 단말기 디자인 조사 ····· 145
[표 107] 디자인 개발 보완 적용 사항 ····· 146
[표 108] 전용단말기의 설계 디자인 검토 ····· 146
[표 109] 전용단말기의 단계별 디자인 ····· 147

[표 110] 전용단말기의 단계별 진행 내용 ····· 147
[표 111] 기구 설계용 배치 검토 및 진행내용 ····· 148
[표 112] 기구 구성 부품 리스트 및 특성 ····· 149
[표 113] 전용단말기 최종 랜더링 ····· 150
[표 114] 기구 사출 ····· 151
[표 115] 전원검사 진행 결과 예시 ····· 152
[표 116] 상용 안테나 조사 ····· 153
[표 117] 안테나 성능 비교 ····· 154
[표 118] Wi-Fi 안테나 장착 ····· 156
[표 119] 운용 시험 장비 목록 ····· 157
[표 120] 공인인증 기관을 통한 운용 시험 장비 목록 ····· 161
[표 121] 항목별 세부 시험 사진 ····· 165
[표 122] 고온 시험 결과 ····· 167
[표 123] 저온 시험 결과 ····· 167
[표 124] 습도 시험 결과 ····· 168
[표 125] 기본 동작 시험 결과 ····· 168
[표 126] 센서의 종류 ····· 180
[표 127] 응용프로그램의 UI 구성 및 디자인 변경 전, 후 ····· 188
[표 128] 개발용 장비의 하드웨어 및 소프트웨어 정보 ····· 191
[표 129] 안드로이드 관련 개발 툴 설치 절차 ····· 192
[표 130] 개발보드의 USB 드라이버 설치 절차 ····· 193
[표 131] 전용단말기 응용프로그램의 기본 제공 기능 ····· 200
[표 132] 지도 확대 기능 수행 전, 후 예시 ····· 206
[표 133] 지도 축소 기능 수행 전, 후 예시 ····· 207
[표 134] Grid Map 출력(On) 또는 제거(Off) 화면 ····· 208
[표 135] 전용단말기 응용프로그램에서 요구되는 GIS 엔진 관련 기능 ····· 220
[표 136] 연구개발 최종 성과물(전용단말기) 형상 ····· 231
[표 137] 연구개발 결과물의 평가기준 ····· 232
[표 138] 기존 문제점 대비 개선 구현 사항 ····· 235
[표 139] 전용단말기 디자인 ····· 236
[표 140] 전용단말기 기구 설계 및 사출 ····· 236
[표 141] 전용단말기 보드 설계 및 시제 제작 ····· 237
[표 142] Android 운영체제 기반의 전용단말기 소프트웨어 화면 ····· 237
[표 143] 전용단말기의 프로그램 등록 현황 ····· 238
[표 144] 전용단말기의 시제품 제작 현황 ····· 239
[표 145] 전용단말기의 테스트베드를 통한 현장시험 진행사항 ····· 239
[표 146] 김포공항 테스트베드 운용시험 진행현황 ····· 240

[표 147] 김포공항 작업차량에 전용단말기 설치 작업 ····· 242
[표 148] 전용단말기 사용관련 1차 문의사항 정리 ····· 245
[표 149] 전용단말기 사용관련 2차 문의사항 정리 ····· 246
[표 150] 전용단말기 사용관련 3차 문의사항 정리 ····· 247
[표 151] 차량단말 서브시스템 시험 기능 목록 ····· 250
[표 152] 이동체 안전관리 서브시스템 시험 기능 목록 ····· 251
[표 153] AP별 802.11a(2.4GHz) 네트워크 채널 ····· 252
[표 154] 시험(1~3차) 시 사용된 안테나의 종류 ····· 253
[표 155] 패치 안테나 고정 방안 ····· 254
[표 156] Wi-Fi 시험 결과(2차) ····· 255
[표 157] Wi-Fi 거리별 시험 결과(3차) ····· 258
[표 158] 차량단말 서브시스템 기능 시험 결과 ····· 264
[표 159] 이동체 안전관리 서브시스템 기능 시험 결과 ····· 267
[표 160] 김포공항 내 전용단말기 및 운영센터 운용시험 시험 진행 ····· 268
[표 161] 50m 이동 후 오차율 확인 ····· 274
[표 162] 100m 이동 후 오차율 확인 ····· 274
[표 163] 150m 이동 후 오차율 확인 ····· 274
[표 164] 항행안전시설의 국내 설치현황 ····· 279
[표 165] 항행안전시설의 국외 설치현황 ····· 280
[표 166] 기술개발 후 국내·주요 판매처 현황 ····· 283
[표 167] 기술개발 후 국외 주요 판매처 현황 ····· 284
[표 168] 공항 내 AP 수량 및 용도 ····· 285
[표 169] 주파수별 안테나 통달거리 하타 시뮬레이션 ····· 286
[표 170] 국내 공항 면적 비교 정보 ····· 286
[표 171] 통신 방식에 따른 통신 모듈 ····· 288
[표 172] 공항별 면적 대비 조업차량 대수 비교 ····· 290
[표 173] GPS 오차 보정을 위한 관련 시스템 ····· 291
[표 174] 연구개발 수행 진도율(연차별) ····· 293
[표 175] AFSIS-6의 차량 설치용 단말기 비용 절감 ····· 296
[표 176] 안전사고 예방 시 경제적 기대효과 ····· 297
[표 177] 국토지리정보원을 통한 지도 데이터 구매 관련 정보 ····· 299
[표 178] 본 연구 성과물의 활용가능 분야 ····· 302
[표 179] 사업화 추진 계획 ····· 304
[표 180] 시스템 홍보 및 수출을 위한 노력(최근 실적) ····· 306

<그림 목차>

[그림 1] AIRNET 시스템 구성 ····· 33
[그림 2] AIRNET의 차량 단말기 ····· 33
[그림 3] 지상안전관리 기술 예시 ····· 34
[그림 4] 연구수행 추진체계 ····· 38
[그림 5] 연구개발 추진과정 ····· 39
[그림 6] 형상관리 및 사업진도관리가 가능한 TRAC 화면 ····· 40
[그림 7] 개발문서 및 소스코드의 버전관리가 가능한 SVN ····· 41
[그림 8] 참여 연구진 현황 ····· 46
[그림 9] 전용단말기의 하드웨어 블록도 ····· 53
[그림 10] 차량단말 서브시스템 응용프로그램의 상태 천이도 ····· 54
[그림 11] 전용단말기의 소프트웨어 블록도 ····· 55
[그림 12] AFSIS-6 시스템 구성도 ····· 78
[그림 13] 서브시스템간의 인터페이스 구성도 ····· 80
[그림 14] CRC(Cyclic Redundancy Check) 계산 ····· 82
[그림 15] Airside 이동체 안전관리 서비스 인터페이스 구성도 ····· 91
[그림 16] 지능형 신호제어 서비스 인터페이스 구성도 ····· 101
[그림 17] FOD/포장면 안전관리 서비스 인터페이스 구성도 ····· 103
[그림 18] 제·방빙 차량 원격관리 서비스 인터페이스 구성도 ····· 108
[그림 19] 비상상황 대응 지원 서비스 인터페이스 구성도 ····· 112
[그림 20] 소프트웨어 유지보수 서비스 인터페이스 구성도 ····· 119
[그림 21] 전용단말기의 목업 제작용 도면 ····· 130
[그림 22] LCD를 통한 안드로이드 메인화면 출력 확인 ····· 141
[그림 23] GPS 데이터 수신확인 화면 ····· 142
[그림 24] Wi-Fi 포팅 후 무선 AP 수신 확인 ····· 143
[그림 25] 고온, 저온, 습도 시험 구성도 ····· 158
[그림 26] 고온 시험 결과(1차, 2차, 3차 그래프 확인) ····· 158
[그림 27] 저온 시험 결과(1차, 2차, 3차 그래프 확인) ····· 159
[그림 28] 습도 시험 결과(1차, 2차, 3차 그래프 확인) ····· 159
[그림 29] 고온 시험조건 예시 ····· 161
[그림 30] 저온 시험조건 예시 ····· 161
[그림 31] 습도 시험조건 예시 ····· 162
[그림 32] 고온, 저온, 습도 시험 구성도 ····· 162
[그림 33] 기본 동작 시험 구성도 ····· 162
[그림 34] 공항 전용단말기 시험 결과서 ····· 169
[그림 35] Sourcery G++ 실행화면 ····· 170

[그림 36] Sourcery G++ 설치 확인 ····· 171
[그림 37] 부트 로더 빌드 실행 파일 열기 ····· 171
[그림 38] 부트 로더 이미지 파일 경로 설정 ····· 171
[그림 39] 부트 로더 빌드를 위한 컴파일러 설정 ····· 172
[그림 40] 부트 로더 빌드 실행 ····· 172
[그림 41] 부트 로더 빌드 완료 확인 ····· 172
[그림 42] 환경설정 모드 진입 ····· 173
[그림 43] 환경설정 모드 ····· 174
[그림 44] 환경설정 파일 로드 ····· 174
[그림 45] 커널 빌드를 위한 컴파일러 설정 ····· 174
[그림 46] 커널 빌드 완료 확인 ····· 174
[그림 47] 안드로이드 운영체제 파일 압축 해제 ····· 175
[그림 48] 안드로이드 운영체제 빌드 설정 ····· 175
[그림 49] 안드로이드 운영체제 빌드 파일 열기 ····· 175
[그림 50] 안드로이드 빌드 시 참조할 커널 파일의 경로 설정 ····· 176
[그림 51] 안드로이드 빌드 후 생성파일 경로 설정 ····· 176
[그림 52] 안드로이드 운영체제 빌드 실행 ····· 176
[그림 53] 안드로이드 운영체제 빌드 확인 ····· 177
[그림 54] 안드로이드 운영체제 구조 ····· 177
[그림 55] 안드로이드 커널 구조 ····· 178
[그림 56] 입력 장치(Input Device) 구조 ····· 179
[그림 57] 센서 서브시스템 구성 ····· 180
[그림 58] 오디오 서브시스템 구조 ····· 181
[그림 59] CD Detector PIN GPIO 설정 ····· 182
[그림 60] s3c_gpio_cfgpin 함수 ····· 182
[그림 61] s3c_gpio_setpull 함수 ····· 183
[그림 62] s5p_gpio_get_drvstr 함수 ····· 183
[그림 63] gpio_set_value 함수 ····· 184
[그림 64] Wi-Fi 포팅 후 무선 AP 수신 확인 ····· 184
[그림 65] 운영체제 보안해제 설정 ····· 185
[그림 66] 아이콘 터치 음량 변경 ····· 186
[그림 67] 운영체제 지원 언어 변경 ····· 186
[그림 68] Home.apk 생성 디렉토리 ····· 187
[그림 69] 운영체제 기본 배경화면 저장 위치 ····· 187
[그림 70] 개발용 PC와 개발보드의 연결도 ····· 191
[그림 71] 차량단말 서브시스템 응용프로그램의 로그인 화면 ····· 197
[그림 72] 배경이미지 추가를 위한 함수 사용 ····· 197

[그림 73] 배경이미지의 풀 스크린 적용을 위한 액티비티 클래스 설정 ····· 197
[그림 74] 입력버튼의 UI 구성을 위한 함수 추가 예시 ····· 198
[그림 75] 구성된 UI의 기능구현 예시 ····· 198
[그림 76] UI 구성을 위한 함수 추가 예시 ····· 199
[그림 77] 텍스트 박스 기능구현 예시 ····· 199
[그림 78] 소프트웨어 버전 확인을 위한 패킷 생성 로직(1) ····· 200
[그림 79] 소프트웨어 버전 확인을 위한 패킷 생성 로직(2) ····· 201
[그림 80] 버전 확인 결과에 따른 처리 분기 ····· 201
[그림 81] 최신버전의 설치파일 다운로드 ····· 202
[그림 82] 다운로드가 완료된 최신버전의 설치파일 실행 ····· 202
[그림 83] 소프트웨어 업데이트 진행 화면 ····· 203
[그림 84] 사용자 계정정보 확인을 위한 패킷 전송 ····· 203
[그림 85] 통합서버(미들웨어)의 로그인 승인 로그 ····· 203
[그림 86] 통합서버(미들웨어)의 로그인 거부 로그 ····· 204
[그림 87] 로그인 승인 요청 결과에 따른 분기 처리 ····· 204
[그림 88] 전용단말기 사용을 위한 로그인 화면 ····· 205
[그림 89] 로그인 화면의 차량종류 도움말 ····· 205
[그림 90] 지도 확대 기능 처리 로직 ····· 206
[그림 91] 지도 축소 기능 처리 로직 ····· 207
[그림 92] Grid Map 출력상태(On/Off) 설정 ····· 208
[그림 93] GPS 모듈 사용을 위한 객체 생성 및 설정 ····· 209
[그림 94] GPS 데이터 수신 주기 설정 및 파싱 결과 출력 ····· 209
[그림 95] GPS 데이터 수신 정보 출력(위도 및 경도) ····· 210
[그림 96] 출동지시 정보 수신 처리 로직(일부) ····· 210
[그림 97] 시스템 종료 여부 확인(1) ····· 211
[그림 98] 시스템 종료 여부 확인(2) ····· 211
[그림 99] 항공사진 출력을 위한 Raster 데이터(CMG 파일) 처리 흐름 ····· 212
[그림 100] Raster 데이터(CMG 파일) 처리 로직(일부) ····· 213
[그림 101] 레이어 출력을 위한 Vector 데이터(SHP 파일)의 처리 흐름 ····· 214
[그림 102] SHP 및 SHX 파일 처리 로직(일부) ····· 215
[그림 103] DBF 파일(속성정보) 처리 로직(일부) ····· 215
[그림 104] 레이아웃 인플레이션을 이용한 레이어 출력용 뷰(View) 설정 ····· 216
[그림 105] 레이어 출력 설정 예시 ····· 216
[그림 106] 항공사진(또는 일반사진) 출력을 위한 비트맵 처리 로직(일부) ····· 217
[그림 107] 레이어 출력을 위한 처리 로직(일부) ····· 217
[그림 108] 항공사진(좌) 및 일반사진(우) 출력화면 ····· 218
[그림 109] 화면좌표를 지도좌표로 변환하는 기능함수(일부) ····· 218

[그림 110] 지도좌표를 화면좌표로 변환하는 기능함수(일부) ····· 218
[그림 111] 지도화면 업데이트 기능함수(일부) ····· 219
[그림 112] 최적경로 산출을 위한 기능함수(일부) ····· 219
[그림 113] 지도화면 업데이트 처리 로직 예시 ····· 220
[그림 114] 지도 이동 시 화면 업데이트 예시 ····· 221
[그림 115] 최적경로 및 거리 산출 처리 로직(일부) ····· 221
[그림 116] 최적경로 및 거리 산출 결과화면 예시 ····· 221
[그림 117] 위치정보가 포함된 사고정보 처리 로직(일부) ····· 222
[그림 118] 사고지점 등록 시 사용자 인터페이스 흐름(예시) ····· 222
[그림 119] 사고 발생지점 이미지와 팝업 가이드 출력(예시) ····· 223
[그림 120] 사고지점 이미지 및 팝업 출력 처리 로직(일부) ····· 223
[그림 121] 정보 등록여부 팝업 가이드 처리 로직(일부) ····· 224
[그림 122] 이동체 출력을 위한 처리 로직(일부) ····· 225
[그림 123] 이동체 출력 화면 예시 ····· 225
[그림 124] 안전규정 위반 중 속도위반 검사 로직(일부) ····· 226
[그림 125] 내 차량의 속도위반 예시 ····· 226
[그림 126] 내 차량의 위치 출력 처리 로직(일부) ····· 227
[그림 127] 내 차량의 현재 위치정보 출력 예시 ····· 227
[그림 128] 항공사진 또는 일반사진 변경 처리 로직(일부) ····· 228
[그림 129] 항공사진 또는 일반사진 로딩 처리 로직(일부) ····· 228
[그림 130] 환경 설정에서 제공되는 지도 선택 기능 ····· 228
[그림 131] 내 차량이 위치한 Grid 구역 정보 검사 로직(일부) ····· 229
[그림 132] 내 차량이 위치한 Grid 구역 정보 출력 예시 ····· 229
[그림 133] 김포공항 활주로의 레이블 출력(32L-14R, 32R-14L) ····· 230
[그림 134] 활주로 레이블 출력 로직(일부) ····· 230
[그림 135] 전용단말기 하드웨어 형상(최종 성과물) ····· 235
[그림 136] 연차별 보고서 및 프로그램 등록증 형상 ····· 238
[그림 137] 키오스크 설치완료 화면(전체) ····· 244
[그림 138] 키오스크 모니터(1) ····· 244
[그림 139] 키오스크 모니터(2) ····· 244
[그림 140] 김포국제공항 내 AP 설치 위치 ····· 252
[그림 141] 무선 AP 설치 및 안테나 방향 ····· 252
[그림 142] 기능시험을 위한 장비 구성도 ····· 254
[그림 143] 안테나 장착(1) ····· 260
[그림 144] 안테나 장착(2) ····· 260
[그림 145] 공항 내 Wi-Fi 음영지역 ····· 260
[그림 146] 7-8번 AP 지점 ····· 261

[그림 147] 6-9번 AP 중간지점(1) ····· 261
[그림 148] 6-9번 AP 중간지점(2) ····· 261
[그림 149] 6-9번 AP 중간지점(3) ····· 261
[그림 150] 7-9번 AP 지점 ····· 261
[그림 151] GPS 인테나 사용 시 수신감도 ····· 262
[그림 152] GPS 외장 안테나 사용 시 수신감도 ····· 262
[그림 153] 현재 사용 중인 지도화면 ····· 262
[그림 154] '12년 8월 31일자 구글 지도화면 ····· 262
[그림 155] 주행 시 GPS 오차 발생 ····· 262
[그림 156] 주행 시 위치출력 차이 ····· 263
[그림 157] 정차 시 위치출력 차이 ····· 263
[그림 158] AP 상태 확인 ····· 263
[그림 159] 58km/h 주행 시 속도출력 확인 ····· 266
[그림 160] GPS 오차로 인한 기동지역 침범위반 발생 ····· 266
[그림 161] 주행 시 잔디밭에 차량위치 ····· 266
[그림 162] 김포공항 테스트베드에서의 GPS 정확도 시험 진행 구성 ····· 270
[그림 163] GPS 위치정보 보정 시나리오 ····· 271
[그림 164] 일렬로 주행 시 트래킹 화면 ····· 272
[그림 165] 나란히 주행 시 트래킹 화면 ····· 272
[그림 166] 추월 주행 시 트래킹 화면 ····· 272
[그림 167] 전용단말기별 위치정보 수신 타이밍 예시 ····· 273
[그림 168] 운영센터에서의 DGPS 기능을 통한 차량 이동 화면 예시 ····· 273
[그림 169] 김포공항 내 시연 진행 장소 ····· 275
[그림 170] 본 연구개발과제의 현장평가(최종평가) 진행 장소 ····· 277
[그림 171] 차량 내 전용단말기 설치 ····· 278
[그림 172] 무선 네트워크를 위한 안테나 설치 ····· 278
[그림 173] 사업담당자에게 기능 설명(시연진행 중) ····· 278
[그림 174] 전용단말기 설치 차량 운행(시연진행 중) ····· 278
[그림 175] 국내 시장규모 ····· 281
[그림 176] 국외 시장규모(예상) ····· 282
[그림 177] 김포공항 무선 네트워크 커버리지 구성도 ····· 285
[그림 178] 김포공항 내 서버 및 운영센터 설치 위치 ····· 289
[그림 179] AFSIS-6 기능의 다양한 활용 방안 ····· 301
[그림 180] AFSIS-6의 사업화 추진전략 ····· 305
[그림 181] GBP 참가(공항/철도분야) ····· 307
[그림 182] 해외 전시회 참가(AUSA) ····· 307
[그림 183] 사업화 추진체계 ····· 307

1장. 연구개발과제 개요

1절. 연구개발 배경 및 필요성

1. 연구개발의 추진배경

○ 공항은 안전의 중요성이 매우 강조되고 있지만 현재 대부분 공항에서는 항공기와 조업차량, 항공기와 항공기 등과 같은 이동체간 충돌방지와 관련된 사고예방이 대부분 수작업으로 수행되고 있음

○ 그러나 늘어나는 트래픽을 감당하기에는 한계가 있고 이를 해결하기 위해 몇 개 국가가 신규 기술을 적용하는 노력이 시도 되고 있으나 기존 시설과의 연동, 운용개념의 변경으로 발생할 지도 모르는 안전사고로 인해 현장에 적용되기는 쉽지 않음

○ 이에 국토교통부는 국내의 G-10 수준의 공항 운영 및 안전관리 기술 향상을 위하여 『항공선진화』 사업의 일환으로 공항시설 안전통합관리 시스템 (Airport Facilities Safety Integrated Management System, AFSIS-6)을 5년간 개발함

○ AFSIS-6는 타 국가와는 달리 우리나라의 장점인 IT 기반기술을 접목하여 공항 주요시설과 항공기, 지상조업차량 등 공항 내 이동체의 안전성 확보는 물론 공항운영의 효율성을 증대할 수 있는 차별화된 기술로써, 미래지향적 안전가치를 실현하는 지능형 공항을 창조하는 목표로 개발됨

○ AFSIS-6의 실제 필드 적용 가능성을 시험하기 위해 김포 국제공항에 테스트베드를 구축하여 시범운영 중임

○ 본 시스템은 이착륙중인 항공기, 계류 중인 항공기, 조업차량 등의 이동체의 현황을 관제소에서 모니터링 할 수 있으며 차량에 장착된 단말기를 통하여 주변의 이동체를 파악하고, 신호등을 제어하여 이동체를 통제하며, 전력시설관리, 지적관리 등의 서비스를 제공하여 활주로 외의 사고를 예방함

○ 그러나 국책과제로 진행된 본 시스템은 시제품(Prototype)수준으로 상용화를 위해서는 QoS가 보장되는 확장성, 편리성, 기술성을 고려한 개선 작업 및 안정화 작업이 필요함

2. 연구개발의 필요성

- 현재 공항시설 안전통합 관리시스템은 시제품(Prototype) 단계이기 때문에 상용화를 위해서는 많은 개선이 필요함

- 본 시스템을 상품화하여 수출을 하거나 타 공항에 설치할 경우 가장 문제가 되는 것은 공항 내 조업차량에 장착되는 단말기임

- 전체 시스템에서 차량에 장착되는 단말기는 가장 큰 비중을 차지하지만 여건상 개발을 하지 않고 UMPC와 같은 상용 제품을 사용함에 따라 비용문제를 안고 있으며, 상품의 경쟁력과 부가가치를 떨어뜨리는 요인이 되고 있음

- 종합적으로 기존 시스템은 인프라 측면에서 어느 정도 완성단계이지만 단말기 측면에서는 개발이 거의 이루어지지 않아 전용단말기에 대한 기술 개발이 필요함

- 기존 단말기의 경우 여러 분야의 상용 제품을 단순히 시스템 통합(System Integration)하는 수준이므로 경제적인 측면에서의 부가가치성이 떨어짐

- 전용단말기의 경우 비용측면에서 경쟁력을 갖출 수 있고 수출 시 상대적인 부가가치가 발생하므로 개발의 필요성이 높음

※ 김포공항의 경우 이동체 차량은 2,000대, 인천공항의 경우 20,000대 임

- 전용단말기는 일반 상용 내비게이션과 같이 속도 및 위치 표시 등의 일반 기능을 제공하며, 공항 내 안전관리를 위한 전용 기능을 포함하여 범용성과 특수성을 동시에 확보할 수 있음

- 전용단말기 개발 시 실제 공항 내 작업자의 요구사항 반영으로 소비자 및 사용자 측면에서 비효율적인 업무 능률을 향상시킬 수 있음

- 기존 기능에 상용 및 독자 기술을 추가하여 기술의 향상과 사업성을 증대시키는 효과를 기대할 수 있음

- 휴대폰 또는 내비게이션 등 스마트 기기의 기능에 익숙한 사용자에게 친밀한 인터페이스와 최신 사양의 하드웨어 및 소프트웨어를 제공하여 사용자 중심의 개발이 필요함

2절. 연구개발의 목표 및 범위

1. 연구개발의 정의

가. 연구개발 최종목표

○ 라이선스 문제가 없는 최적의 실시간 운영체제(RTOS)와 국산 GIS 엔진이 탑재된 공항시설 안전통합 관리시스템 전용단말기 개발

나. 최종목표의 성격 및 설정 근거

번호	핵심 성과물	연구 기간	내용	최종 목표	목표성능 설정근거
1	소프트웨어 등록	'12.08.31 ~ '14.07.31	공항시설 안전통합 관리시스템 전용 단말기에 탑재될 응용프로그램 개발	소프트웨어 등록 3건	해당사업의 평가대상기간 동안 등록한 국내 지적재산권을 행사하기 위한 목표치 설정
2	보고서	'12.08.31 ~ '14.07.31	공항시설 안전통합 관리시스템 전용단말기의 개발문서	보고서 3건	해당사업의 수행결과로 주요 부품 및 제품을 위한 설계도면 및 사양서 개발 성과평가를 위한 목표치 설정
3	보고서	'12.08.31 ~ '14.01.31	공항시설 안전통합 관리시스템의 적용을 위한 유관 정부부서의 업무지원 지침안	보고서 2건	유관 정부부서의 업무지원을 위한 지침안 개발

[표 1] 최종목표의 성격 및 설정 근거

2. 연구개발 내용 및 범위

- ○ 차량에서 신속하게 GPS 정보를 확인하기 위하여 사용이 용이한 7inch 정전압(Capacitive Sensing) 방식의 터치스크린이 탑재된 공항 전용단말기 개발
- ○ 배터리, 터치스크린, 안테나, 무선부 및 주제어부 등을 위한 메인보드와 외부 케이스를 구성하고, 내부 배치 및 기구의 내구성을 고려한 전용단말기 개발
- ○ 사용자 편의성을 고려한 외부 연동 모듈의 내부 실장 및 연동 케이블 최소화 구현
- ○ 전용단말기를 구성하고 있는 각종 부품의 인터페이스에 대한 드라이버 구현
- ○ 공항 내 존재하는 차량의 위치를 화면에 표출하여 사용자가 모니터링 할 수 있고 통합서버(미들웨어)에 차량의 위치정보를 전송하는 기능 제공
- ○ 공항 내 주요 상황(안전규정 위반, 사고, FOD(Foreign Object Debris)) 정보를 통합서버(미들웨어)에 전송하는 기능 제공
- ○ 공항 내 테스트베드를 통한 성능 개선 연구 및 실 사용자 요구사항을 반영하여 개발
- ○ 개발된 전용단말기의 하드웨어 및 소프트웨어 성능의 안정성 확보
- ○ 향후 국내 및 국외 공항 설치 등 확장성을 고려한 기반 구축
- ○ 기존 상용 내비게이션과 차별화 된 안전성과 특수성을 고려한 공항 전용의 단말기 개발
- ○ 시스템의 문제점을 파악하고 지속적인 수정과 성능 개선 수행
- ○ 한국교통연구원, 한국공항공사 등 공인 기관과의 연계된 연구 개발 진행

3절. 관련분야의 국내외 기술개발 현황

1. 국내외 기술개발 동향 및 분석

○ 공항 이용 및 운영의 효율성과 안전성을 향상시키고 편리성을 제공하는 항공 안전 기술의 개발은 다음의 표와 같이 국외에서도 연구가 진행되고 있음

기관(업체)	주요 기술명(프로젝트명)	기술 개요
유럽위원회 (EC)	AIRNET (AIRport NETwork for Mobiles Surveillance and Alerting)	공항 내 이동지역의 안전관리 개선과 효율적인 운용을 목적으로 하는 시스템
ARINC	Airport Systems Integration Services	공항 내 운용되는 시스템들의 통합 서비스 제공을 목적으로 하는 시스템
SITA	AMS (Airport Management Solution)	공항의 기반시설, 승객관리, 화물 등 공항 운영 및 통합 관리를 목적으로 하는 시스템
SIEMENS	AODB (Airport Operational DataBase)	공항 운영 시 통합적인 정보관리 및 보완을 목적으로 하는 시스템

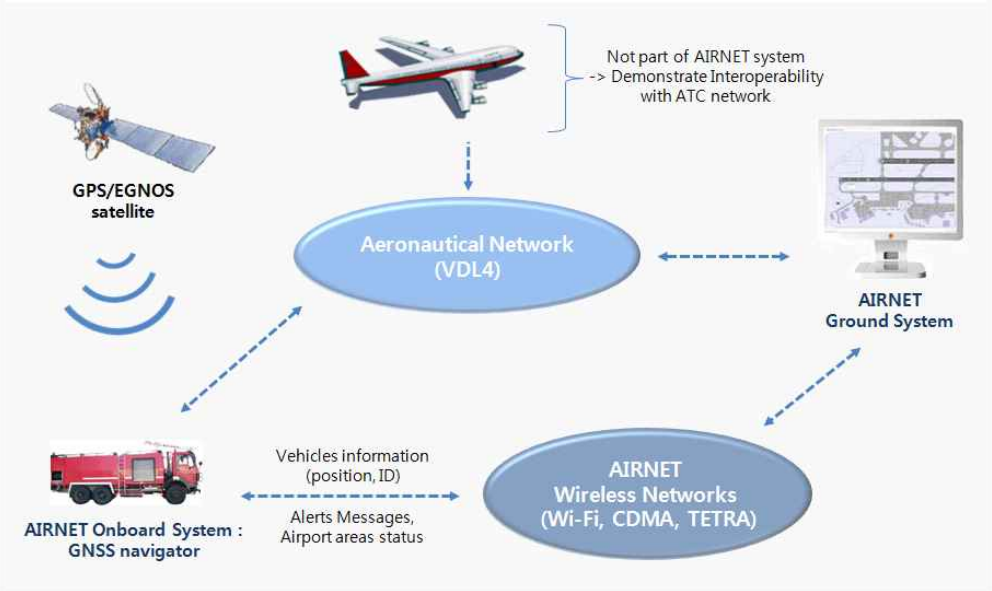
[표 2] 국외에서 연구가 진행 중인 경쟁기술

○ 경쟁기술 중 본 사업화 기술대상과 유사한 시스템 구성을 갖춘 AIRNET의 주요 특징은 다음과 같음

○ AIRNET는 유럽위원회(EC)의 STREP(Specific Targeted Research Project) 일환으로 2004년 1월부터 2006년 12월까지 진행된 공항 내 차량 운영의 안전성과 효율성을 향상시키는 EGNOS(European Geostationary Navigation Overlay Service) 기반의 안전관리 시스템임

○ AIRNET은 시제품(Prototype)으로 개발되어 포르투갈의 Oporto 공항에서 시범운영 되었고, 저비용으로 인프라 구축이 가능하기 때문에 중소형 공항에 적합한 시스템임

○ AIRNET의 시스템 구성은 다음과 같으며 공항 내 교통 상황 및 이동체 모니터링은 Air Traffic Controller, Airport Operations Officer, Ground Handling으로 구성된 Ground System에서 이루어짐



[그림 1] AIRNET 시스템 구성

○ 차량에 장착되는 단말기에서는 공항 내 운행 중인 차량의 위치와 ID 및 속도 등의 정보를 확인할 수 있음

○ 차량 단말기 화면에는 공항 레이아웃을 표시하는 지도가 출력되어 활주로와 유도로 등의 영역을 구분함

○ 차량 단말기에서는 공항 내 운행 중인 차량이 브로드케스트 하는 위치 정보와 Air Traffic Controller에서 송신하는 항공기 정보를 수신하여 이동체의 위치를 출력함

○ 차량에서 충돌, 침범, 과속을 할 경우 차량 단말기 모니터에 관련 알림 정보가 출력됨

○ Ground System에서는 공항 내 이동체의 위치와 속도를 출력하며 항공기-차량 간 충돌에 대한 모니터링과 출입 권한이 없는 차량을 통제함



[그림 2] AIRNET의 차량 단말기

○ 공항 내 지상안전관리 기술은 미국, 유럽 등에서 GIS, 스마트기기 기반, 무선 네트워크 기술 등을 적용하여 사용 중이며 FAA 안전규정 관련 업무 자동화, 지상점검 업무효율 제고, 차량자산 관리, 보호구역 안전관리 등의 효과를 기대함

- 주요 활용분야

  • Air/Landside 차량 관리, FAA139 공항운영증명(점검업무 적용)

  • 공항교통 관리(택시 대기 통제), 자동 톨게이트

  • 주차관리, VIP 차량 관리, 차량 자산 관리

  • 보호구역 관리(보안측면)

- 주요 수요자

  • Airport Operations Management

  • Daily Airfield Inspectors

  • Airfield Maintenance Crew

  • 지상 조업사, 항공화물운송업체 등

- 기술 적용 사례

  • 미국 내 62개 공항, 아르헨티나 2개 공항, 캐나다 4개 공항, 일본/노르웨이/스페인/타이완/영국/우루과이 각 1개 공항에 적용되었으며 주차관리(39곳) 및 Airside/Landside 차량관리/지상 안전관리(35곳)에 가장 많이 활용됨

  • 독일 베를린 Tegel 공항, 포르투갈 Porto 공항, 벨기에 Brussel 공항에 적용되었으며 Airside 차량 안전 및 업무 효율 관리에 가장 많이 활용됨



[그림 3] 지상안전관리 기술 예시

2. 유사 시스템과의 비교

○ 경쟁기술인 AIRNET은 공항시설 안전통합 관리시스템의 6가지 서비스 중 Airside 이동체 안전관리 기술과 유사성을 가짐

AFSIS-6의 서비스	AIRNET	AFSIS-6
Airside 이동체 안전관리 기술	O	O
항공기-차량 주요 교차지점 지능형 신호제어 기술	X	O
이동지역 이물질(FOD)/포장면 파손 대응 능동형 안전관리 기술	X	O
제·방빙(De-icing) 등 특수지원 차량 원격관리 기술	X	O
비상상황 시 소방 및 구급차량 등 외부협정기관의 대응 및 지원기술	X	O
공항 전력시설 통합모니터링 기술	X	O

[표 3] 공항시설 안전통합 관리시스템(AFSIS-6)의 6가지 서비스

○ Airside 이동체 안전관리 기술을 중심으로 AFSIS-6와 AIRNET을 비교 및 분석한 내용은 다음과 같음

구분	AIRNET	AFSIS-6
차량 운용	• 차량에 전용 단말기를 장착하여 사용	• 기존 시스템 운용 시 차량에 UMPC를 장착하여 사용 • 본 과제를 통해 공항전용 차량 단말기 개발 완료 • 조업차량별 로그인 계정에 따라 제공하는 기능이 다름
이동체 위치 정보 획득	• GPS/EGNOS를 사용하여 차량의 위치정보 획득 후 Air Traffic Controller의 네트워크를 통하여 브로드케스팅	• GPS를 사용하여 차량에서 미들웨어로 위치 정보 송신 후 미들웨어에서 공항 내 모든 차량에 브로드케스팅
GIS 활용	• 공항 내 레이아웃을 표시하는 레이어로 활주로와 유도로 등의 영역을 구분하여 출력	• 위성사진과 레스터 방식의 비트맵 이미지 중 사용자가 지도 화면을 선택할 수 있는 기능 제공

구분		AIRNET	AFSIS-6
		• 지도 확대, 축소, 이동과 같은 조작 기능 제공	• 지도 확대, 축소, 이동, 전체보기, 회전과 같은 조작 기능 제공
공항 내 운용	차량	• 공항 내 항공기 및 차량의 위치 표시	• 공항 내 항공기 및 차량의 위치 표시 • 사고, 이물질, 위반 사항 발생 시 관제 센터로 정보 등록 가능 • 사고, 이물질의 발생 지점 표시
	관제 센터	• 공항 내 항공기 및 차량의 위치 표시 • 이동체간 충돌 감시 및 출입 제한 차량 통제	• 공항 내 항공기 및 차량의 위치 표시 • 이동체간의 충돌 감시 • 사고, 이물질 발생 지점 표시 및 모니터링 • 신호등 상태 모니터링 • 공항 내 시설물 조회 • 이동체 접속 확인 및 조회 • 지정한 날짜의 이동체 접속 이력 조회
안전 관리	사고 발생시	• 공항 내 사고 발생 시 관련 정보를 관제 센터에서 차량 단말기로 송신	• 공항 내 사고 발생 시 차량 단말기에서 관제 센터로 사고 정보 전송 가능 • 관제 센터에 등록된 사고 정보를 운용자가 확인한 후 소방차와 구급차에 출동 요청 • 소방 및 구급활동이 완료된 후 관제 센터로 사고 처리에 대한 메시지를 전송하여 상황 종료함
	위반 발견시	• 충돌, 침범, 과속 등의 위반사항 발생 시 차량 단말기에 알림 메시지 출력	• 충돌, 침범, 과속 등의 위반사항 발생 시 차량 단말기에서 위반 사항 알림 출력 • 관제 센터로 위반 정보 전송 • 관제 센터에서 위반 정보 확인 후 순찰차에 출동 지시 또는 경고 메시지를 전송하여 처리
	이물질 발견시	• 해당 기능 없음	• 공항 내 이물질 발견 시 차량 운용자는 사진 촬영 후 이물질 정보를 관제 센터로 전송

구분	AIRNET	AFSIS-6
		• 관제 센터에서 이물질 정보 확인 후 수거를 위한 유지보수지시 진행
사용자 편의성	• 항공기의 이착륙 정보가 제공되어 차량 운전자가 실시간으로 확인 가능 • 관제 센터로부터 텍스트 메시지 수신 가능	• 공항 내 발생되는 사고, 이물질, 위반 사항에 대한 통합적인 정보 관리 및 모니터링 가능 • 다양한 지도 조작 기능 제공 • 사고 발생 지점까지의 최단경로 기능 제공 • 현재 공항의 날씨 제공 • 식별하기 용이한 UI 제공
시스템 구축	• 중소형 공항에 구축하기 적합함	• 모든 공항에 구축 가능

[표 4] 사업화 대상기술과 경쟁기술 비교 및 분석

2장. 연구개발 수행 내용 및 결과

1절. 연구수행 전략 및 방법론

1. 추진체계

○ 국토교통부, 국토교통과학기술진흥원 사업을 총괄하는 기관으로 개발기간동안 요구사항 및 사업화를 위한 협의 및 수출을 위한 협조를 담당함

○ 주관기업인 인소팩(주)은 본 사업의 상용화를 위한 R&D를 담당하며 저렴하고 우수한 제품을 개발하는데 주력하며 또한 사업화를 위한 다각적인 노력을 담당함

○ 한국공항공사는 현재 5년간 국책과제의 산출물로 김포공항에 설치되어 있는 테스트베드를 지원하며, 상용화를 하는 기간 혹은 수출시 바이어의 현장 방문 홍보를 위해 협조함

○ 개발하는 동안 기 개발에 참여한 업체들의 협조가 필요한 경우 협력을 하여 노하우 및 개선사항의 협조를 득하고 수출시 공동 컨소시엄을 구성하여 공동 노력함

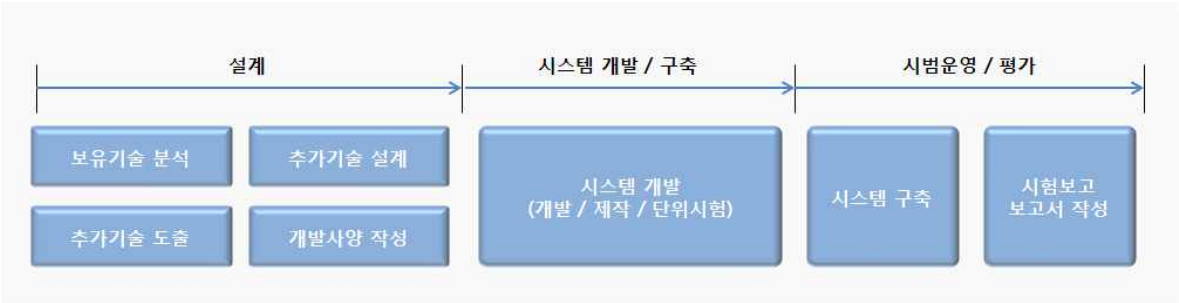
○ 필요에 따라 5년간 기존의 시스템을 개발하면서 자문 및 평가위원을 포함한 위원회를 구성하여 자문역할을 담당하여 문제 해결에 대한 논의 및 방안을 검토함



[그림 4] 연구수행 추진체계

2. 기술개발 추진전략

가. 기술개발 추진과정



[그림 5] 연구개발 추진과정

○ 시스템 설계

- 연구총괄을 담당하는 주관연구기관의 보유기술 분석 및 추가기술 도출, 설계 개발사양 등의 시스템 설계를 통해 시스템 현황 분석 및 개발방향 설정

○ 시스템 개발 및 구축

- 하드웨어 개발 : 전용단말기 개발을 위한 하드웨어 설계 및 제작을 진행하게 되며, 소프트웨어 검증 및 개발에 따른 하드웨어 성능 최적화 및 튜닝 수행

- 소프트웨어 개발 : 하드웨어 제작에 따른 초기 보드 기능 검증을 기본으로 하여 OS 및 드라이버 포팅을 수행한다. OS 포팅이 완료된 시점에서 SDK 배포와 함께 응용프로그램 개발 수행

○ 시범운영 총괄

- 김포공항에 구축된 테스트베드를 활용하여 개발된 제품이 현장에서 직접 사용자가 운용시 검증을 통하여 문제점 보완 및 개선

- 외국바이어들을 초청하여 시스템을 홍보하고 운용상 문제가 없음을 확인시켜주는 기회와 동시에 외국공항의 환경에서 적합한지 개선의 여지가 있는지를 통하여 제품의 품질 향상

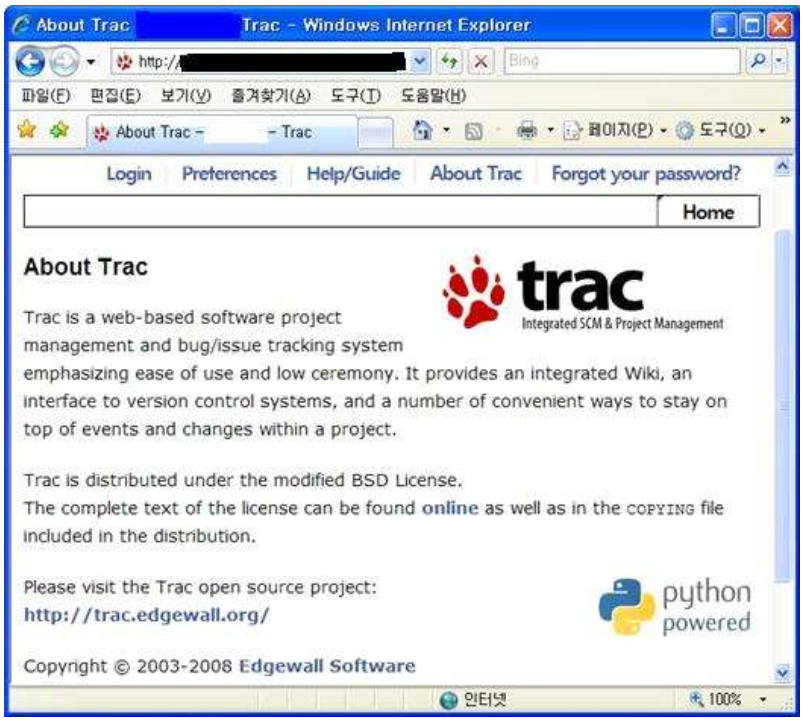
○ 정책적 지원 및 지침 안 개발

- 김포공항에 기 구축된 테스트베드의 지속적인 활용을 위한 정책적 지원 및 유관 정부부서 지원을 위한 지침 안 개발

나. 도구를 활용한 체계적인 사업관리

(1) TRAC 서버를 활용한 산출물 형상관리 및 사업 진도관리

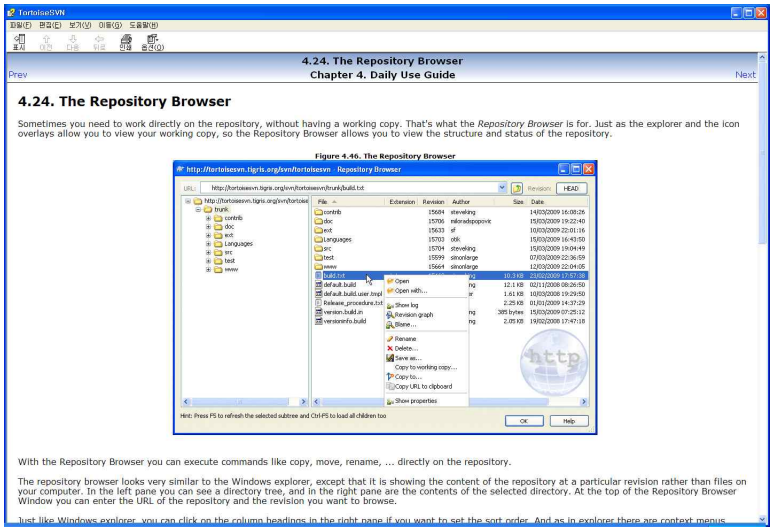
- 성공적인 사업수행을 위해서 산출물 형상관리 및 사업 진도관리가 가능한 TRAC 서버를 설치하여 사업기간동안 참여기관의 산출물 형상관리 및 사업 진도관리 진행
- TRAC 서버의 이력관리 기능을 이용하여 하드웨어 및 소프트웨어를 포함한 설계문서와 소프트웨어 구현문서 등 사업기간동안 발생하는 모든 산출물을 대상으로 형상관리 수행
- TRAC 서버의 Ticket 발송/접수/처리 기능을 이용하여 참여기관의 사업 수행 현황을 파악하여 성공적인 사업 완료를 위한 진도관리 수행



[그림 6] 형상관리 및 사업진도관리가 가능한 TRAC 화면

(1) SVN을 활용한 개발문서의 버전 관리

- 사업기간동안 발생하는 개발문서는 프로젝트 단위로 버전 관리 기능을 제공하는 SVN(SubVersion) 적용
- 소프트웨어 설계문서 및 소스코드를 대상으로 SVN을 통한 버전 관리
- 버전관리 도구를 이용한 체계적인 개발문서 버전 관리를 통한 소프트웨어의 품질 향상
- SVN과 TRAC 서버를 연동하여 형상관리 가능



[그림 7] 개발문서 및 소스코드의 버전관리가 가능한 SVN

다. 세부 추진계획 및 방법

(1) 1차년도 세부 추진계획 및 방법

○ 본 사업은 보유기술인 “공항시설 안전관리 기술개발”과제를 통해 공항시설 안전통합관리 시스템 개발에 참여한 인소팩(주)과 한국교통연구원이 공동으로 상용화, 사업화 도모

○ 공항시설 안전통합관리 시스템”은 현재 김포공항에 테스트베드가 구축되어 있으며 본 사업에 당해 테스트베드를 활용하여 기술의 완성도를 높이는 것이 중요

○ 이를 위하여 본 사업에서는 기 구축된 테스트베드를 활용할 수 있도록 사용자 기관과의 관계 정립 및 정부 부서의 협조를 통하여 시스템 보완하고 사업화 완성도 도모

○ 본 시스템의 국내 표준화를 통해 기술의 활성화 도모

순번	연구개발내용	세부추진 계획 및 방법	수행기간 (월)	해당 기관
주관연구기관				
1	요구사항 도출 및 설계 구축 주요부품 선정 및 확보	• 공항시설 안전통합 관리시스템 전용 단말기의 블록도, 모듈별 기능도, 요구 사양 관련 문서 작성 • 주요 부품 BOM File 생성 • 주요 부품 공급 업체 선정 및 가격 산출	3	인소팩 (주)

순번	연구개발내용	세부추진 계획 및 방법	수행기간 (월)	해당 기관
2	시제품 디자인 확보 개발보드 설계 및 구현 후 각 모듈별 기능 검증	• 기구부, 입출력부, 주제어부, 메모리부, 무선부, 전원부, 외부인터페이스부에 대한 설계 및 구현, 모듈별 기능 검증 수행	3	인소팩 (주)
3	시제 제작 및 검증 GUI 개발 및 기능구현	• 하드웨어 목표 사양을 통한 제작 • 기구부, 입출력부, 주제어부, 무선부 시제 제작 및 검증 수행 • 유선 인터페이스 프로토콜 설계 • 전체 GUI 메뉴 구성	3	인소팩 (주)
4	GUI 개발 및 기능구현 GIS를 활용한 운용환경 응용프로그램 개발	• 기구부, 입출력부, 주제어부, 무선부에 대하여 하드웨어 성능 검증 결과를 반영하여 수정 및 보완 • GIS 미들웨어 구축 및 연동	3	인소팩 (주)
공동(위탁)연구기관				
1	테스트베드 관리 및 정책적 지원	• 기 구축된 테스트베드의 원활한 활용을 위한 유관기관과의 정책적 연계 지원 • 유관 정부부처와의 협의를 통한 기술 활성화 도모	1.5	한국교통 연구원
2	국내 표준화 방안 분석	• 유관 정부부처와의 협의를 통한 국내 표준화 방안 분석	1.5	한국교통 연구원
3	중간 보고서 작성	• 중간 보고서 작성	0.25	한국교통 연구원

[표 5] 1차년도 세부추진 계획표

(2) 2차년도 세부 추진계획 및 방법

○ 2차년도의 연구개발은 1차년도에 이어 지속적인 활용을 위한 정책적 지원을 하고, 상용화 될 시스템에 대한 만족도를 조사하여 국내 표준화 방안을 위한 지침안 개발을 범위로 설정

○ 본 사업을 통해 주관기관에서 개발하게 될 상용화 시스템에 대하여 유관 정부부서와 협조를 통한 지침 안 개발을 통한 국내 공항 구축방안 수립

순번	연구내용	세부추진 계획 및 방법	수행 기간 (월)	해당 기관
주관연구기관				
1	전용단말기 기구 디자인 및 금형제작	• 상용 내비게이션 디자인 수집 및 벤치마킹 수행 • 자체 디자인 부서 업무 협조 진행 • 자체 기구 설계 부서 업무 협조 진행 • 기구 제작 전문 업체를 통한 QDM 금형 및 기구 제작 • 사용 환경을 고려한 기구 재질 선정	2	인소팩 (주)
2	전용단말기 하드웨어 개발 및 수정 보완	• 성능 개선 사항과 회로 오류 사항에 대한 수정 설계 • 무선 특성(Wi-Fi, GPS) 개선을 위한 송수신단 설계 • 기구 적용을 위한 보드 면적 축소 설계 • 디자인을 고려한 불필요한 외장 연결 소자 축소 설계 • 사용자의 편의성을 고려한 입력 장치 설계 • 주변 케이블 정리 장치 적용	6	인소팩 (주)
3	GUI 개발 및 기능 최적화(계속)	• 상용 내비게이션 지도 맵 참조 개발 수행 • 공항 내 운용 개념과 편의성을 고려하여 설계 • 다수의 사용자를 고려한 맞춤형 GUI 설계 • ASDE 등 공항 내 장비 연동 데이터 전시 설계 • FOD 화면 연동 전시 설계	9	인소팩 (주)
4	GIS 엔진 개발 및 최적화(계속)	• GPS 수신 데이터 오차 수정 설계 • 2D/3D 엔진 선택 적용 설계 • 휴대용 GIS 엔진이 구현되도록 경량화된 프로그램 설계 • 상용 GIS 엔진을 모티브로 안정적이며 독자적인 GIS 엔진 설계	9	인소팩 (주)

순번	연구내용	세부추진 계획 및 방법	수행 기간 (월)	해당 기관
5	실제 설치용 시제 제작 및 검증	• 사용자 운용 테스트 • 사업화 전 개선 사항 도출 • 실 운용시험 전 지속 테스트	3	인소팩 (주)
6	테스트베드 환경 구축 및 운용시험	• 기구 수정 사항 도출 및 개선 • GPS 모듈 내장형 안테나(FPCB 안테나 개발) 성능 개선 • 하드웨어 성능 안정화 • 테스트베드 테스트 • GIS 미들웨어 구축 및 연동	3	인소팩 (주)
공동(위탁)연구기관				
1	테스트베드 관리 및 정책적 지원(계속)	• 기 구축된 테스트베드의 원활한 활용을 위한 유관기관과의 정책적 연계 지원 • 유관 정부부처와의 협의를 통한 기술 활성화 도모	3	한국교통 연구원
2	국내 표준화 방안 수립	• 유관 정부부서의 업무 지원을 위한 지침안 개발	3	한국교통 연구원
3	최종 보고서 작성	• 최종 보고서 작성	0.5	한국교통 연구원

[표 6] 2차년도 세부추진 계획표

(3) 3차년도 세부 추진계획 및 방법

○ 1, 2차년도 연구개발을 통해 축적된 기술과 경험을 기준으로 시스템을 안정화하고 김포공항 내 테스트베드에서 실제 사용 시 발생되는 문제점에 대한 보완 및 반영을 목표로 진행

○ 3차년도 연구개발 진행 중인 전용단말기의 기구 수정·보완을 위하여 다양한 요구사항을 도출하고 해결 방안 검토 진행

○ 특히 2차년도 평가 시 도출된 자체 성능시험의 대안으로 ETRI의 시험평가 부서를 통한 환경 관련 시험성적서 획득을 통하여 공정성 확인

- 소프트웨어 수정을 통한 사용자 불편 사항 개선

- 지속적인 운용 시험을 통한 문제점 도출 시도 및 보완

- 성공적으로 사업완료가 이루어지도록 체계적인 문서 관리 진행

순번	연구내용	세부추진 계획 및 방법	수행 기간 (월)	해당 기관
주관연구기관				
1	GUI 개발 및 기능 최적화(계속)	• GUI 기능 최적화	4.5	인소팩 (주)
2	GIS 엔진 개발 및 최적화 (계속)	• GIS 기능 제공을 위한 인터페이스 최적화 • GIS 엔진 최적화	4.5	인소팩 (주)
3	전용단말기 기구 수정보완	• 전용단말기 QDM 금형 수정 제작	4.5	인소팩 (주)
4	운용시험	• 기 구축된 테스트베드를 활용한 운용시험 • 공정성 확보를 위한 외부 시험기관을 통한 환경시험 진행 후 성적서 획득	3	인소팩 (주)
5	보고서 작성	• 최종보고서 작성	1	인소팩 (주)

[표 7] 3차년도 세부추진 계획표

3. 연구 추진역량(주관연구기관의 연구진 인적 능력)

○ 연구책임자는 30년 동안 본 사업의 전문가로서 산학연의 경험을 두루 갖추고 있으며 사업 책임자로서 충분한 자질을 갖추고 있음

○ 참여연구진은 AFSIS-6 개발에 참여한 경험이 있거나 민수, 방산 단말기 개발경험을 갖고 있는 우수 연구진으로 구성함

○ 공동연구 기관인 한국교통연구원(KOTI) 연구진은 AFSIS-6 시스템 개발에 참여 했으며 공공기관으로서의 역할, 상용화를 위한 국내표준에 해당하는 고시연구에 참여함

○ 차세대 지능형 공항시스템 구축을 위한 최적의 인적구성 형성을 위하여 한국교통연구원을 중심으로 실제 공항운영을 담당하는 양대 공항공사 및 각계 전문기관과 산・학・연 협력 체제를 구축하고 당해 기관의 소속 연구원을 적극 활용함

○ 공항시설 및 업무부문 표준화 지침을 도출함

○ 시설물 GIS 구축부문에 설계/분석/구현 경험이 많은 GIS 전문가를 투입함

○ 공항 내 무선 통신 네트워크 분석 및 설계 연구를 위한 인적구성

- RF분야 7년 이상 설계 경험이 있고, 고주파대역의 연구경험이 있는 개발인력을 구성함
- 통신프로토콜 연구개발 분야는 네트워크 설계 분야 경험의 참여 연구인력으로 구성함
- 본 사업 참여 개발인력은 다수 프로젝트를 통해 상용기술 개발 경험이 있음



[그림 8] 참여 연구진 현황

2절. 연구수행과정 및 내용

1. 연차별 정량적 성과목표 및 성과지표

연차	연차별 성과목표	성과지표		측정방법	최종 목표치	가중치 (0~1)
1차	중간보고서	주관 기관	보고서	요구사항 도출 및 설계 시제품 디자인 전용단말기 개발 GUI 개발 및 기능구현 GIS 미들웨어 개발 및 최적화	1건	50%
	지식재산권 행사를 위한 소프트웨어 등록		소프트웨어 등록	전체 GUI 확인 전용단말기 운용환경 프로그램 시연 소프트웨어등록증 확인	1건	50%
	소계(소계 x 가중치(0.8))					80%
	테스트베드 관리 및 정책적 지원	공동 연구 기관	보고서	업무협의 성사 건수	1건	70%
	공항시설 안전통합 관리시스템(AFSIS-6) 관련 지침 현황 분석 및 문제점 도출			보고서 작성	1건	30%
	소계(소계 x 가중치(0.2))					20%
	합계					100%
2차	전용단말기 기구디자인 및 금형 제작	주관 기관	기구제작	전용단말기 기구 디자인 확보 전용단말기 기구 목업 및 QDM금형 제작	1건	30%
	지식재산권 행사를 위한 소프트웨어 등록		소프트웨어 등록증	전체 GUI 확인 전용단말기 운용환경 프로그램 시연 소프트웨어 등록 확인	1건	35%

연차	연차별 성과목표	성과지표		측정방법	최종 목표치	가중치 (0~1)
	중간보고서		보고서	GUI 개발 및 기능 최적화 GIS 미들웨어 개발 및 최적화 전용단말기 하드웨어 수정보완 테스트베드 환경구축 및 운용시험	1건	35%
	소계(소계 x 가중치(0.8))					80%
	테스트베드 관리 및 정책적 지원	공동 연구 기관	보고서	업무협의 성사 건수	1건	70%
	공항시설 안전통합 관리시스템(AFSIS-6) 관련 지침개정안 제시			연구보고서	1건	30%
	소계(소계 x 가중치(0.2))					20%
	합계					100%
3차	전용단말기 기구 수정보완	주관 기관	기구제작	전용단말기 기구 QDM 금형 수정 제작	1건	30%
	지식재산권 행사를 위한 소프트웨어 등록		소프트웨어 등록증	전용단말기 운용환경 프로그램 시연 소프트웨어 등록 확인	1건	40%
	최종보고서		보고서	GUI 개발 및 기능 최적화 GIS 미들웨어 개발 및 기능 최적화 운용시험	1건	30%
	소계					100%

[표 8] 정량적 성과목표 및 성과지표

2. 연구개발 주요내용

연차	연구목표	주요 연구내용
1차년도	요구사항 도출 및 설계구축	• 공항시설 안전통합 관리시스템 전용단말기 블록도, 모듈별 기능도, 요구사항 문서 작성 • 주요 부품 BOM 파일 생성 • 주요 부품 공급 업체 선정 및 가격 산출
	시제품 디자인 확보, 개발보드 설계 및 구현, 각 모듈별 기능 검증	• 시제품 디자인 구상 및 아크릴 재질의 기구 형상 제작 • 기구부, 입출력부, 주제어부, 무선부에 대한 하드웨어 성능 검증 결과를 반영하여 수정 및 보완
	개발보드 설계, 시제 제작 및 검증	• 하드웨어 목표 사양을 통한 제작 • 기구부, 입출력부, 주제어부, 무선부 시제 제작 및 검증 • 전용단말기의 안드로이드 운영체제 포팅
	GUI 개발 및 기능구현, GIS를 활용한 운용환경 프로그램 개발	• 유선 인터페이스 프로토콜 설계 • 전체 GUI 메뉴구성 • GIS 엔진 구축 및 연동
	테스트베드 관리 및 정책적 지원	• 기 구축된 테스트베드의 원활한 활용을 위한 유관기관과의 정책적 연계 지원
	유관 정부부처와의 협의를 통한 기술 활성화 도모	• AFSIS-6 관련 지침 현황 분석 및 문제점 도출
2차년도	전용단말기 기구디자인 및 금형제작	• 전용단말기 기구 디자인 개발 • 전용단말기 QDM 금형 및 기구 제작
	GUI 개발 및 기능 최적화 (계속)	• GUI 개발 및 기능구현
	GIS 엔진 개발 및 최적화 (계속)	• GIS 기능 제공을 위한 인터페이스 구현 • GIS 엔진 개발
	전용단말기 하드웨어 수정보완	• 하드웨어 제작 및 성능 안정화 • 전용단말기의 안드로이드 운영체제 최적화

연차	연구목표	주요 연구내용
	테스트베드 환경구축 및 운용시험	• 실내 테스트를 위한 테스트베드 환경 구축 • 실험실 수준의 운용시험 • 기 구축된 테스트베드를 활용한 운용시험
	테스트베드 관리 및 정책적 지원	• 기 구축된 테스트베드의 원활한 활용을 위한 유관기관과의 정책적 연계 지원
	유관 정부부처와의 협의를 통한 기술 활성화 도모	• AFSIS-6 관련 지침 현황 분석 및 문제점 도출
3차년도	GUI 개발 및 기능 최적화 (계속)	• GUI 기능 최적화
	GIS 엔진 개발 및 최적화 (계속)	• GIS 기능 제공을 위한 인터페이스 최적화 • GIS 엔진 최적화
	전용단말기 기구 수정보완 및 전용단말기 하드웨어 제작	• 전용단말기 QDM 금형 수정 제작 • 최적화된 전용단말기 하드웨어 제작
	운용시험	• 기 구축된 테스트베드를 활용한 운용시험 • 공정성 확보를 위한 외부 시험기관을 통한 환경시험 진행 후 성적서 획득
	보고서 작성	• 최종보고서 작성

[표 9] 연구개발 주요내용

가. 공항시설 안전통합 관리시스템 전용단말기 요구사항 도출

(1) 연구개발 목적 및 내용

○ 전용단말기의 공급 및 유지보수를 위한 수명주기가 길고 운영체제의 공급 및 유지보수 지원이 가능한 하드웨어 개발

- 전용단말기 하드웨어 설계 및 구현
- 전용단말기 각 모듈별 기능 검증
- 전용단말기 운영체제 포팅

○ 라이선스 문제가 없는 최적의 실시간 운영체제(RTOS)와 GIS 엔진이 탑재된 공항시설 안전통합 관리시스템 전용단말기 응용프로그램 개발

- 전용단말기 개발 환경 구성 및 포팅 및 개발 진행
- 전용단말기 GUI 개발 및 기능구현
- GIS 미들웨어 개발 환경 구성 및 포팅 진행

(2) 요구사항 도출

(가) 주요 요구사항 도출

순번	문제점	요구사항	비고
1	단말기의 크기가 12형으로 설치 후 작업자의 시야확보가 어려움	전용단말기 설치 후에도 작업자의 시야확보가 용이하도록 개발	운용성
2	상용 노트북의 사용으로 외부에 거치하여야 하므로 작업자의 차량 내 공간 사용이 협소함	내비게이션 장비와 유사하게 거치 또는 매립이 가능한 크기로 개발	운용성
3	상용 노트북 내장 Wi-Fi 통신장치 사용 시 외장형 안테나 연결을 할 수 없으므로 무선 성능 저하 발생	전용단말기에 Wi-Fi 통신장치를 내장하고 외장형 안테나를 연결할 수 있도록 개발하여 무선 성능 저하 방지	사용성
4	단말기의 GPS 수신기 및 Wi-Fi 통신장치가 외장형으로 구성되어 운용 중에 연결커넥터가 해제될 위험이 있음	전용단말기에 GPS 수신기 및 Wi-Fi 통신장치를 내장형 또는 Built-in 형식으로 제작하여 운용 중 연결이 해제 되지 않도록 개발	사용성

순번	문제점	요구사항	비고
5	단말기의 GPS 수신기 및 Wi-Fi 통신장치가 외장형으로 구성되어 단말기에 연결 시 케이블 라우팅에 문제가 발생함	전용단말기에 GPS 수신기 및 Wi-Fi 통신장치를 내장형 또는 Built-in 형식으로 제작하여 외부 케이블을 최소화하여 개발	사용성
6	단말기 운용을 위하여 운영체제 및 GIS 엔진 등의 라이선스 비용을 지불해야하기 때문에 초기 투자비용과 운용 중 고가의 고정비용 지출이 발생	라이선스 문제가 없는 최적의 실시간 운영체제를 선택하여 개발하고 GIS 엔진 라이선스에 문제가 없도록 가볍고 저렴한 국산 GIS 엔진을 적용하여 개발	운용성
7	상용제품을 사용하기 때문에 제품의 단종 또는 운영체제 등 주요 품목의 단종 시 유지보수의 문제가 발생	전용 하드웨어와 운영체제를 사용하여 지속적인 운영 및 유지보수가 가능하도록 함	지속성
8	공항시설 안전통합 관리시스템에서의 사용가능성	공항시설 안전통한 관리시스템의 전용단말기로 활용하기 위하여 인터페이스 개발	호환성

[표 10] 전용단말기 주요 요구사항

(나) 기능적 요구사항 도출

순번	항목	사양
1	공통 기능	• 화면에 공항의 이동지역 지도 표출
		• 통합서버(미들웨어)와의 양방향 통신 가능
		• 차량의 현재 위치 표출과 통합서버(미들웨어)로의 전송
		• 현장에 위반/사고/FOD 등의 상황 발생 시 정보를 입력하여 통합서버(미들웨어)로의 전송
		• 통합서버(미들웨어)에서 전송한 정보를 화면 표시하여 운전자(탑승자)에게 알림
2	이동체 안전관리/ 신호제어/ 제·방빙 조업	• 자신의 현재 위치 및 타 이동체의 위치를 지도상에 표출
		• 이동체는 심볼 및 레이블로 표시하여 식별의 용이성 제공

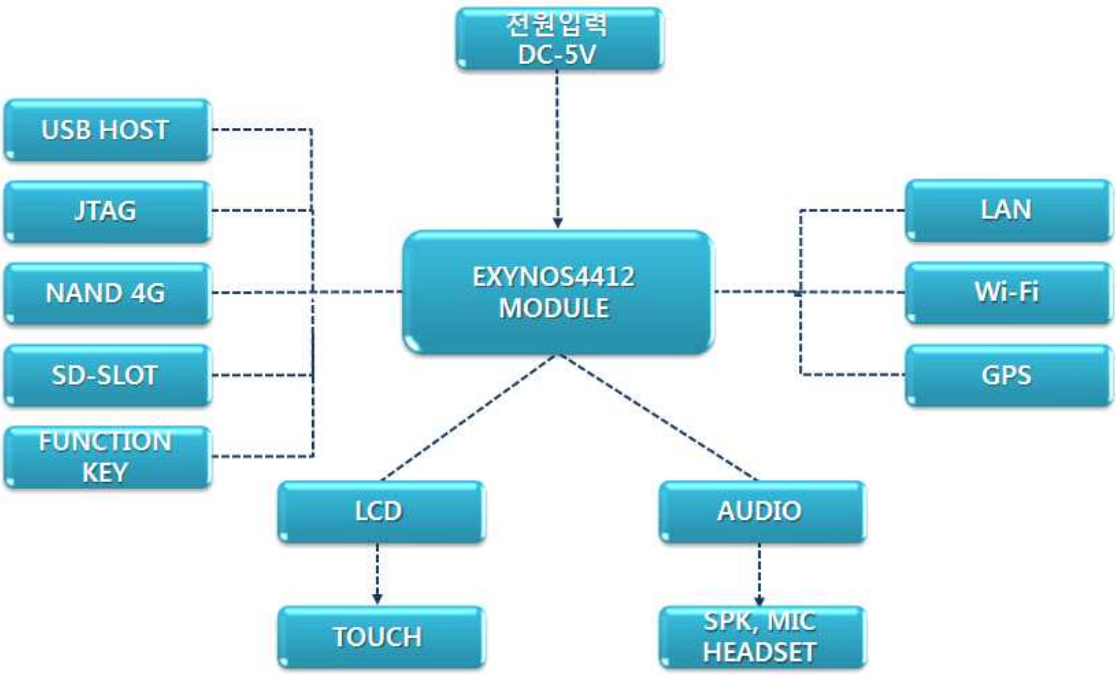
순번	항목	사양
		• 과속 등 주요 위반사항 발생 시 운전자(탑승자)가 인지할 수 있는 경고 표출
		• 교차로 및 신호상태, 항공기 유무 확인 기능 제공
3	FOD/포장면 안전관리	• FOD 등 주요 이물질/포장면 이상에 대한 상황정보를 등록하여 통합서버(미들웨어)로 전송
4	제·방빙 조업	• 제·방빙 조업구역을 지도화면에 표출
5	소방출동 등 협정기관 지원	• 이동지역 내 자신의 현재 위치 및 사고지점의 위치 표출
		• 사고지점까지의 최적경로 분석 및 표출

[표 11] 전용단말기의 기능적 요구사항

(3) 요구사항에 따른 하드웨어 및 소프트웨어 설계

(가) 하드웨어 블록도 작성

○ 전용단말기의 하드웨어 설계 전 블록도를 다음과 같이 작성함



[그림 9] 전용단말기의 하드웨어 블록도

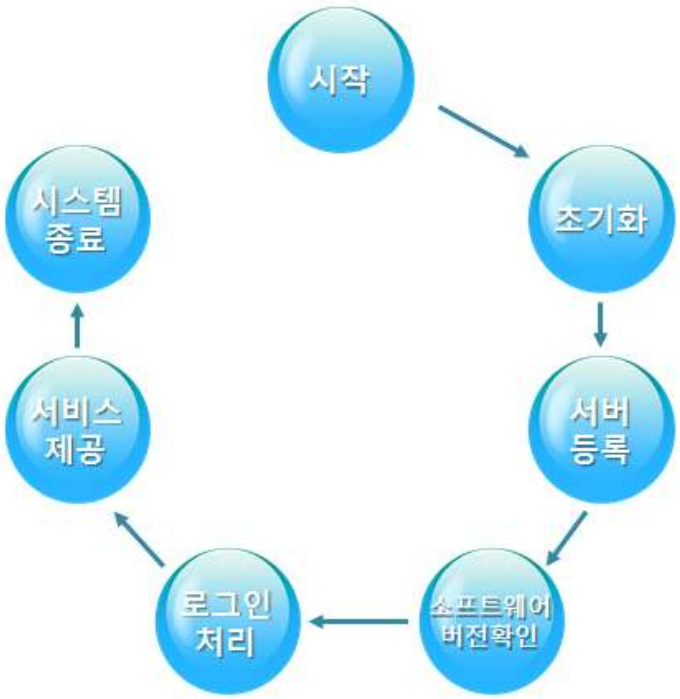
○ 하드웨어는 메인부(Main Part), 입력부(Input Part), 출력부(Output Part), 전원부(Power Part)로 구성되며 각 파트별 모듈 구성은 다음과 같음

구분	구성 모듈
메인부(Main Part)	• Exynos 4412 CPU 모듈
입력부(Input Part)	• JTAG, USB host, Wi-Fi, SD 카드
출력부(Output Part)	• LCD, CAM, Audio
전원부(Power Part)	• 5V DC-JACK, MAIN_BAT

[표 12] 파트별 모듈 구성

(나) 소프트웨어 블록도 작성

① 소프트웨어 상태 천이도



[그림 10] 차량단말 서브시스템 응용프로그램의 상태 천이도

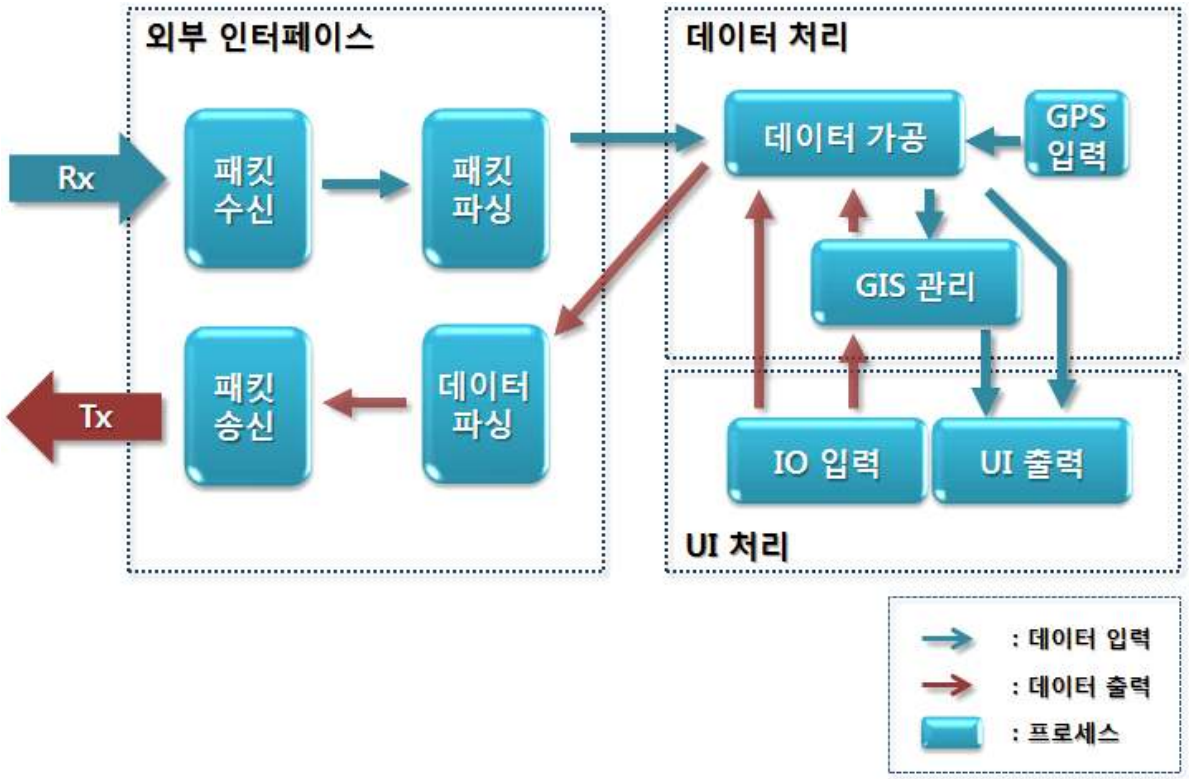
○ 각 단계별 소프트웨어 상태는 다음과 같음

상태	설명
시작	• 전원이 입력된 상태로 전용단말기 하드웨어 구동 후 운영체제 구동

상태	설명
초기화	• 차량단말 서브시스템 응용프로그램이 실행되고 운용을 위한 초기화 작업 진행
서버등록	• 전용단말기가 통합서버(미들웨어)에 접속을 하여 전용단말기 등록 진행
소프트웨어 버전확인	• 현재 구동되고 있는 소프트웨어의 버전확인을 통합서버(미들웨어)에 요청
로그인 처리	• 전용단말기의 차량단말 서브시스템 응용프로그램 사용을 위한 사용자 로그인 진행
서비스 제공	• 전용단말기를 이용하여 이동체 안전관리, FOD 유지보수 등 공항시설 안전통합 관리시스템에서 제공하는 서비스 이용
시스템 종료	• 전용단말기의 하드웨어 및 소프트웨어의 사용 종료

[표 13] 소프트웨어의 상태별 설명

② 소프트웨어 블록도



[그림 11] 전용단말기의 소프트웨어 블록도

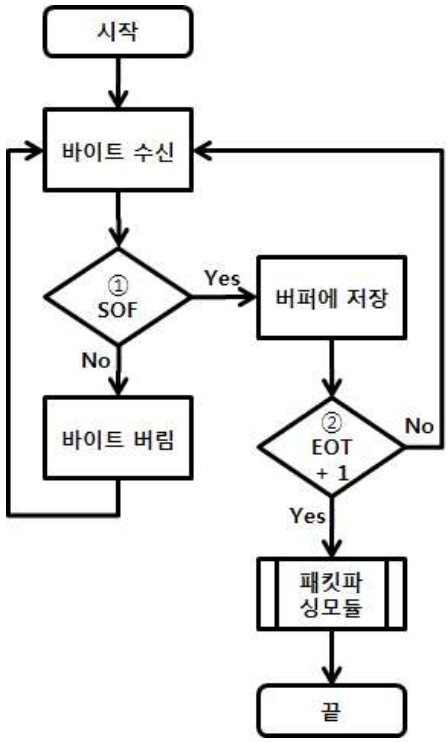
대분류	중분류	소분류
외부 인터페이스	패킷수신 : Wi-Fi를 통해 통합서버(미들웨어)에서 송신하는 운영센터 및 타 차량단말의 정보, 항공기 정보의 패킷수신	• 통합서버(미들웨어)에서 송신한 패킷을 식별하여 SOF(Start Of Frame)부터 ETX (End Of Transmission, EOT)까지의 패킷정보 수신 • 수신된 패킷의 CRC 값을 계산하여 유효 패킷여부 검사
	패킷 파싱 : 수신된 패킷의 정보를 분석하여 패킷에 담겨있는 정보의 종류를 식별하고 해당 처리 프로세스로 정보 전달	• 항공기의 위치정보 데이터 분석 • 타 차량의 위치정보 데이터 분석 • 운영센터의 조업 및 사고관련 출동 데이터 분석 • 신호제어 데이터 분석
	데이터 파싱 : 가공된 데이터를 분석하여 SOF(Start Of Frame)부터 ETX(End Of Transmission, EOT)까지의 정보와 CRC 값을 첨부한 패킷을 생성하여 통합서버 (미들웨어)로 전송	• 운영센터로 송신할 패킷 생성 • 타 차량에 송신할 패킷 생성
	패킷 송신 : Wi-Fi를 통해 차량단말의 정보를 통합서버 (미들웨어)로 송신	• 패킷 생성 완료 후 통합서버(미들웨어)로 송신
데이터 처리	데이터 가공 : 외부 인터페이스부에서 처리된 데이터를 분석하여 수신된 정보의 ID에 따라 정해진 프로세스를 진행하고, GPS 입력부에서 수신된 GPS 데이터를 분석하여 내부 처리 진행	• GPS 데이터 분석 후 GIS 관리 프로세스로 전달 • 타 차량 위치정보 분석 후 GIS 관리 프로세스로 전달 • 항공기 위치정보 분석 후 GIS 관리 프로세스로 전달 • 운영센터로부터 수신한 데이터 분석 후 GIS 관리 프로세스 및 UI 처리부로 전달
	GPS 입력 : GPS 수신기를 통한 GPS 데이터 수신 및 전달	• GPS 수신기를 통하여 수신한 GPS 데이터를 분석한 후 데이터 가공 프로세스로 전달
	GIS 관리 : 데이터 가공 프로세스에서 분석한 차량 및 항공기의 위치정보와 FOD 및 사고 발생지점의 정보를 기반으로 지도화면에 표시	• GIS를 이용한 지도 이미지 출력 • 공간정보 분석을 통한 GIS 기능제공 • 속성정보 분석을 통한 GIS 기능제공 • 항공기 위치정보를 이용한 위치 출력 • 타 차량 위치정보를 이용한 위치 출력 • FOD 위치정보를 이용한 위치 출력 • 사고 위치정보를 이용한 위치 출력

대분류	중분류	소분류
UI (User Interface) 처리	UI 출력 : GIS 관리 프로세스에서 처리된 지도화면과 사용자의 화면 조작을 통하여 출력되는 정보출력 창 제공	• 공항 지도를 출력하여 사용자에게 제공 • 자차의 위치정보를 이용하여 현재위치 출력 • 타 차량의 위치정보를 이용하여 현재 위치 출력 • 항공기의 위치정보를 이용하여 현재위치 출력 • 수신된 조업관련 정보 출력 • FOD 및 사고의 위치정보를 이용하여 발생위치 출력 • FOD 및 사고 정보등록 등 송신 정보의 생성을 위한 입력창 제공 • 전용단말기의 GPS 등 환경설정을 위한 입력창 제공 • 음성을 통한 위반에 대한 경고 및 수신 정보 알림
	IO 입력 : 사용자로부터 입력된 정보를 확인하여 GIS 관리 및 데이터 가공 프로세스로 전송	• 입력된 정보를 분석한 후 데이터 처리부로 전달

[표 14] 소프트웨어 모듈별 역할 및 데이터 처리 과정

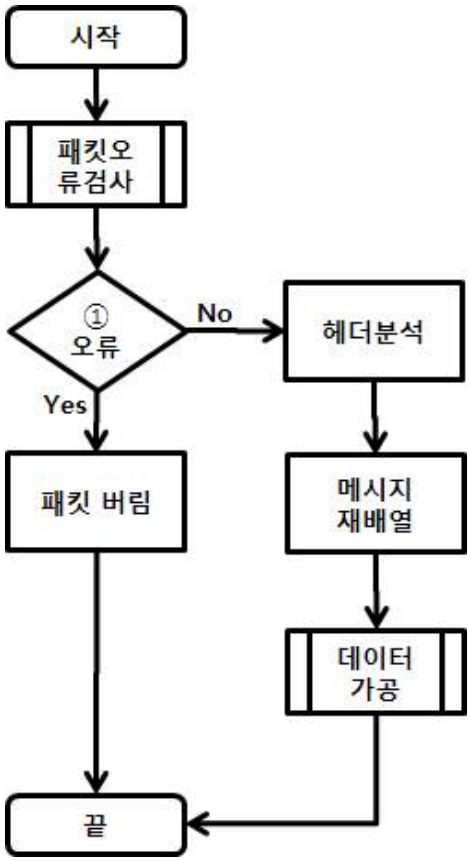
(다) 소프트웨어 모듈별 기능도 작성

① 패킷 수신 모듈

순서도	
기능 설명	통합서버(미들웨어)에서 송신한 스트림을 수신하여 유효 패킷으로 재배열 후 패킷 파싱 모듈에서 처리 가능한 포맷으로 패킷 생성
분기 조건	① 수신된 바이트를 검사하여 SOF 바이트를 수신하였는지 검사 - 수신된 바이트가 SOF가 아닐 경우 수신된 바이트를 제거하고 다시 바이트 수신대기 상태로 전환 - 수신된 바이트가 SOF일 경우 SOF부터 EOT(+1바이트)까지의 바이트를 버퍼에 저장 ② 수신된 바이트가 EOT(+1바이트)가 맞는지 검사 - 수신된 바이트가 EOT(+1바이트)이 아닐 경우 바이트 수신대기 상태로 전환 - 수신된 바이트가 EOT(+1바이트)일 경우 저장된 버퍼를 패킷 파싱 모듈로 전송

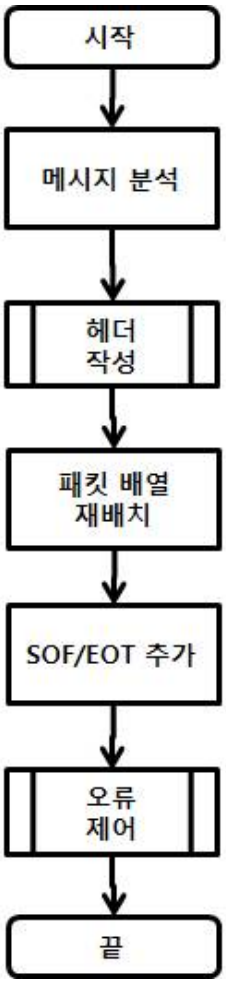
[표 15] 패킷 수신 모듈의 순서도 및 기능설명

② 패킷 파싱 모듈

순서도	
기능 설명	통합서버(미들웨어)에서 수신된 패킷의 내부처리가 가능하도록 SOF, 헤더 분석, EOT를 분석하여 데이터를 재배열 및 생성
분기 조건	① 패킷오류를 검사하여 결과 확인 - 수신된 패킷의 오류검사 결과 패킷이 정상적이지 않을 경우 수신된 패킷은 제거 - 수신된 패킷의 오류검사 결과 패킷이 정상일 경우 헤더를 분석하여 데이터 재배열 후 데이터 가공부로 전달

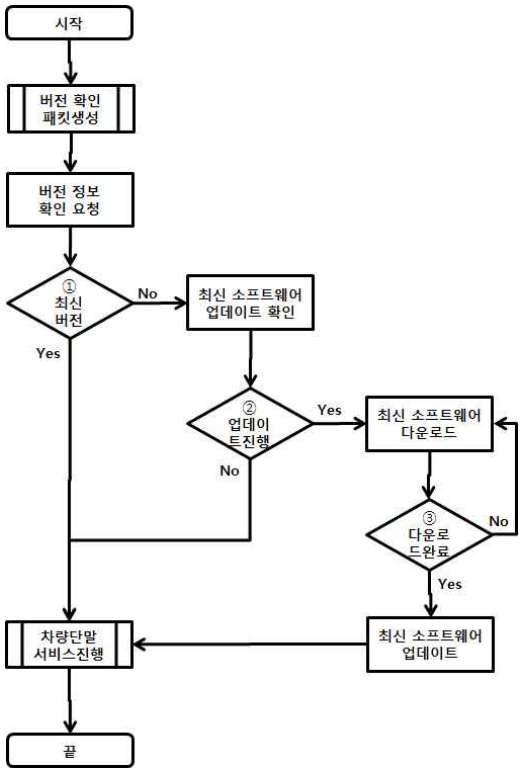
[표 16] 패킷 파싱 모듈의 순서도 및 기능설명

③ 데이터 파싱 모듈

순서도	
기능 설명	데이터 처리부에서 전달된 데이터를 분석하여 송신 패킷을 생성할 수 있도록 바이트 배열 형식으로 재배치하며 SOF, 헤더 생성, EOT를 생성하여 패킷 형태를 구성
분기 조건	없음

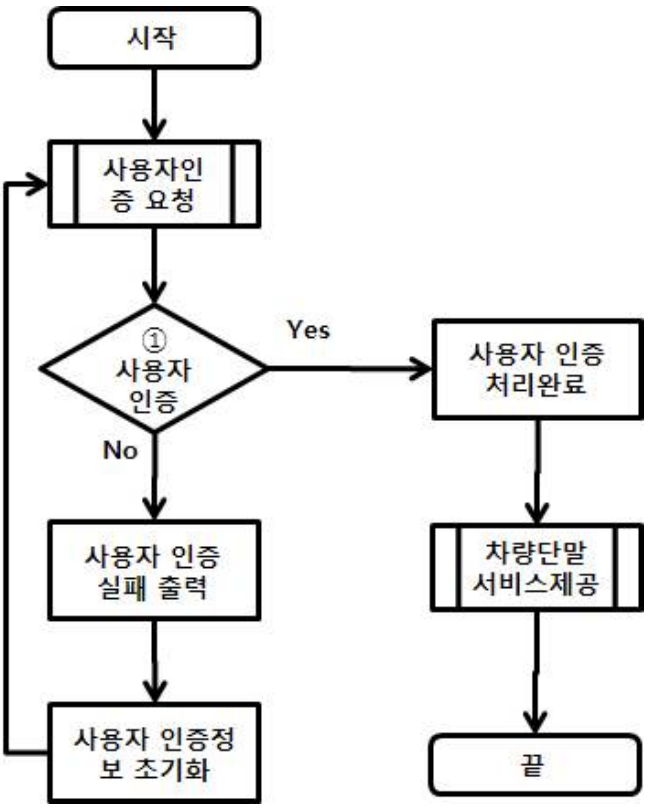
[표 17] 데이터 파싱 모듈의 순서도 및 기능설명

④ 소프트웨어 점검 모듈

순서도	
기능 설명	소프트웨어가 구동되기 전 현재 실행되고 있는 소프트웨어의 버전이 최신 버전인지 통합서버(미들웨어)에 버전확인 요청을 하여 현재 실행 중인 버전이 최신 버전일 경우 계속 프로그램을 실행하고, 최신버전이 아닐 경우 최신버전을 다운로드하여 업데이트할 수 있도록 수행
분기 조건	① 소프트웨어 버전 검사 - 소프트웨어가 최신버전일 경우 차량단말 서비스 진행 - 소프트웨어가 최신버전이 아닐 경우 사용자에게 최신 소프트웨어로의 업데이트 진행 여부 확인 ② 소프트웨어 업데이트 진행여부 검사 - 소프트웨어의 업데이트를 진행할 경우 통합서버(미들웨어)에서 최신 소프트웨어의 다운로드 진행 - 소프트웨어의 업데이트를 진행하지 않을 경우 차량단말 서비스 진행 ③ 최신버전의 소프트웨어 다운로드 진행완료 검사 - 최신버전의 소프트웨어 다운로드가 완료되면 소프트웨어의 재설치 진행 - 최신버전의 소프트웨어 다운로드가 완료되지 않았을 경우 계속해서 최신 소프트웨어 수신 - 최신버전의 소프트웨어 다운로드 실패 시 통합서버(미들웨어)로의 재 다운로드 요청

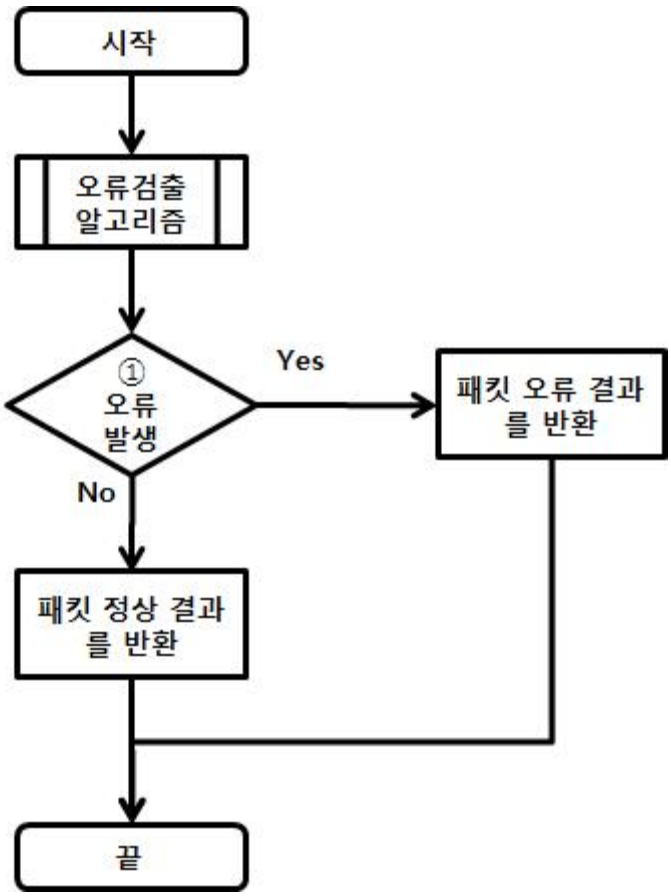
[표 18] 소프트웨어 점검 모듈의 순서도 및 기능설명

⑤ 로그인 모듈

순서도	
기능 설명	전용단말기 사용자의 승인을 위해 통합서버(미들웨어)로 사용자 인증을 요청하고 완료되었을 경우 차량단말 서비스를 제공하며 승인이 되지 않은 사용자의 경우 차량단말 서비스를 이용하지 못 함
분기 조건	① 사용자 인증 처리 여부 검사 - 사용자 인증 검사 결과 인증 실패 시 관련 정보를 출력하고 입력했던 사용자 정보를 모두 초기화한 후 사용자로부터 사용자 인증 정보를 재입력받음 - 사용자 인증 검사 결과 인증 성공 시 차량단말 서비스를 제공함

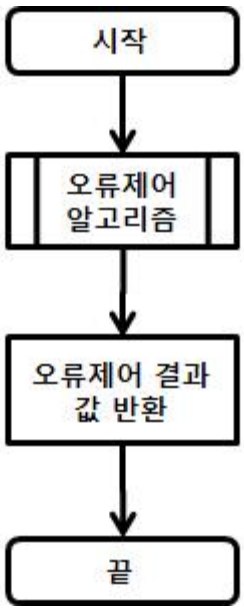
[표 19] 로그인 모듈의 순서도 및 기능설명

⑥ 패킷 오류 검출 모듈

순서도	
기능 설명	통합서버(미들웨어)로부터 수신된 패킷에 오류가 없는지 패킷에 첨부되어 있는 CRC 값을 분석하여 패킷의 무결성 확인
분기 조건	① 오류 검출 알고리즘 처리 결과 검사 - 패킷의 오류 검출 결과 오류가 발생한 경우 패킷 오류 결과 반환 - 패킷의 오류 검출 결과 정상일 경우 패킷 정상 결과 반환

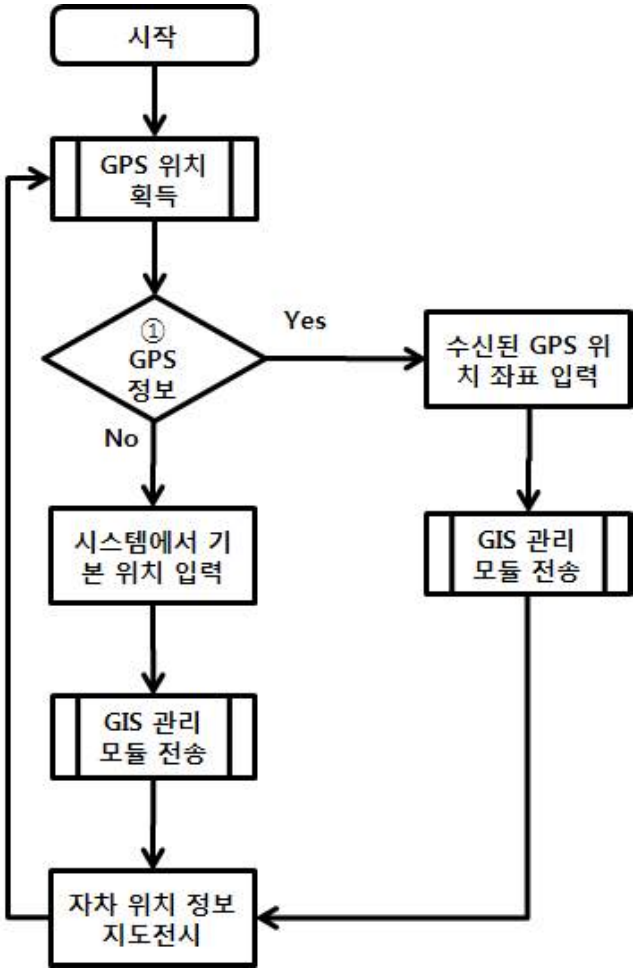
[표 20] 패킷 오류 검출 모듈의 순서도 및 기능설명

⑦ 패킷 오류 제어 모듈

순서도	
기능 설명	통합서버(미들웨어)로 송신하는 패킷의 무결성을 입증하기 위한 CRC 값을 생성하여 패킷에 포함
분기 조건	없음

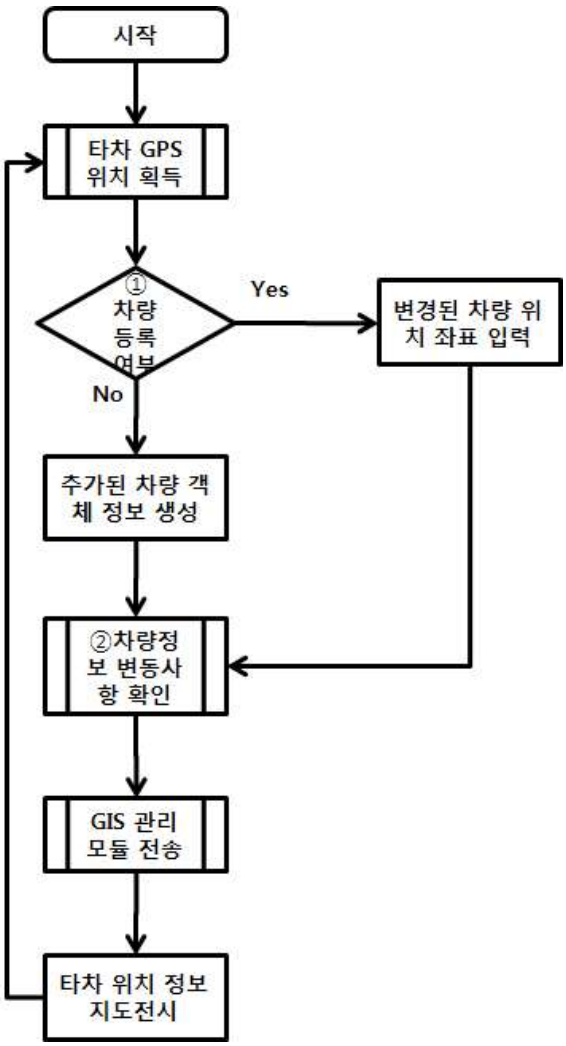
[표 21] 패킷 오류 제어 모듈의 순서도 및 기능설명

⑧ 자차 위치 표출 모듈

순서도	
기능 설명	GPS 위치정보를 입력받아 현재 차량의 위치를 지도화면에 출력하고 갱신된 GPS 위치정보는 데이터 처리를 통해 타 차량 및 운영센터로 전송됨
분기 조건	① GPS 위치정보 획득 여부 검사 - GPS 수신 모듈에서 유효한 GPS 위치정보를 획득하지 못하였을 경우 시스템에서 기본으로 입력된 좌표를 기본 좌표로 하여 자차의 위치를 지도에 출력 - GPS 수신 모듈에서 유효한 GPS 위치정보를 획득한 경우 획득한 위치정보를 사용하여 자차의 위치를 지도에 출력

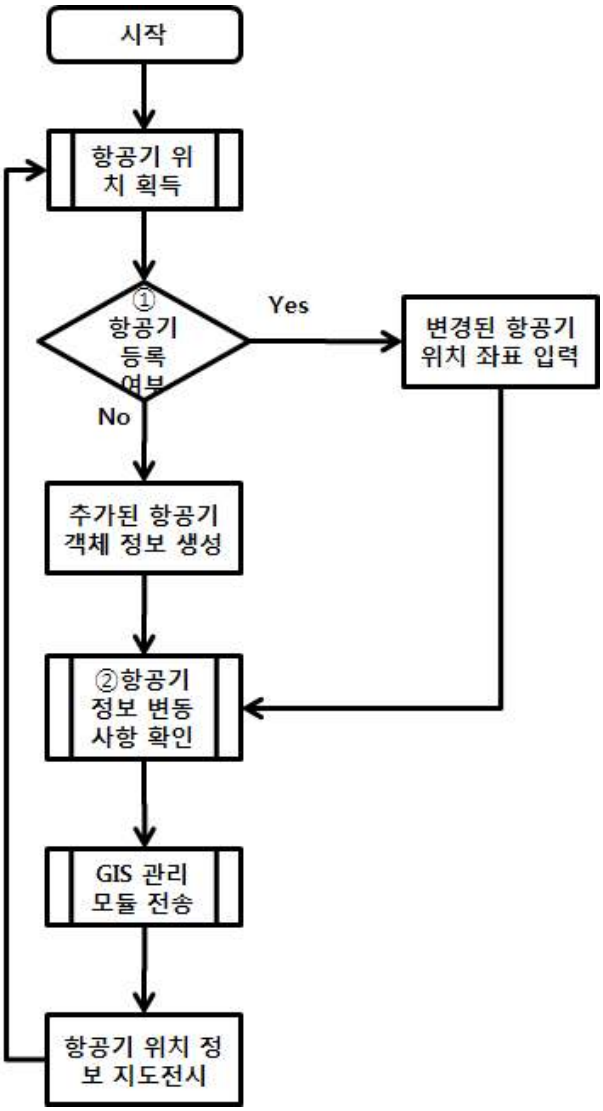
[표 22] 자차 위치 표출 모듈의 순서도 및 기능설명

⑨ 타 차량 위치 표출 모듈

순서도	
기능 설명	외부 인터페이스를 통하여 수신된 타 차량의 위치정보를 처리하여 지도화면에 위치를 출력함. 차량의 접속 상태를 확인하여 차량접속이 일정시간 유지되지 않을 경우 지도화면 표시 제거
분기 조건	① 차량 등록여부 확인 - 수신된 타 차량의 위치정보가 기 등록된 정보가 아닌 경우 새로 추가된 차량의 ID 및 정보를 입력하여 정보 출력을 위한 객체 생성 - 수신된 타 차량의 위치정보가 기 등록된 정보인 경우 차량의 위치정보를 갱신하여 차량의 위치 출력 ② 차량접속 정보 변동사항 확인 - 기존에 수신된 차량 정보 중 일정시간동안 위치정보의 갱신이 이루어지지 않은 차량에 대하여 현재 지도화면에 출력되어 있는 객체 확인 후 해당 차량의 객체를 삭제하고 지도화면에서 이미지를 제거

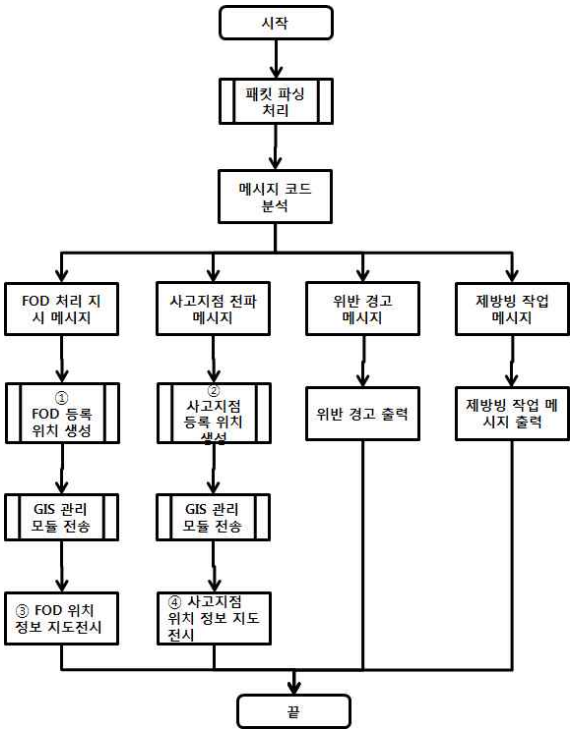
[표 23] 타 차량 위치 표출 모듈의 순서도 및 기능설명

⑩ 항공기 위치 표출 모듈

순서도	
기능 설명	외부 인터페이스를 통해서 수신된 항공기의 위치정보를 처리하여 지도화면에 위치를 출력함. 항공기의 위치정보를 확인하여 일정시간 위치정보가 수신되지 않을 경우 지도화면에서 삭제함
분기 조건	① 항공기 등록여부 확인 - 수신된 항공기의 위치정보가 기 등록된 정보가 아닌 경우 새로 추가된 항공기의 ID 및 정보를 입력하여 정보 출력을 위한 객체를 생성 - 수신된 항공기의 위치정보가 기 등록된 정보인 경우 위치정보를 갱신하여 위치를 출력 ② 항공기 정보의 변동사항 확인 - 기존에 수신된 항공기 정보 중 일정시간동안 위치정보의 갱신이 이루어지지 않은 항공기에 대하여 지도화면의 객체를 삭제하고 지도화면에서 이미지를 제거

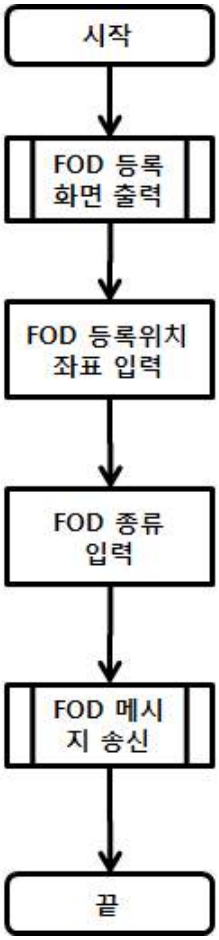
[표 24] 항공기 위치 표출 모듈의 순서도 및 기능설명

⑪ 운영센터 메시지 분석 모듈

순서도	
기능 설명	운영센터에서 송신한 운영메시지를 분석하여 수신 메시지 창에 출력함. FOD 및 사고 등록의 경우에는 해당 지점에 위치 출력
분기 조건	① FOD 등록 위치 출력 처리 - FOD 메시지를 분석하여 메시지 ID와 기존에 수신된 메시지 ID와 비교 후 수신된 메시지가 아닐 경우 객체를 추가 생성 - FOD 메시지가 처음 수신된 경우 출력 처리를 위한 객체 생성 ② 사고지점 등록위치 출력 처리 - 사고지점 메시지를 분석하여 메시지 ID와 기존에 수신된 메시지 ID와 비교 후 수신된 메시지가 아닐 경우 객체를 추가 생성 - 사고지점 메시지가 처음 수신된 경우 출력 처리를 위한 객체 생성 ③ FOD 위치정보의 지도 전시 처리 - 새로 수신된 FOD 객체를 지도화면에 출력처리를 하고 현재 선택된 FOD 객체를 색상으로 구분함 - 새로 수신된 FOD 객체를 수신 확인 후에 FOD 객체의 색상을 다른 FOD 객체의 색상과 동일하게 표현함 ④ 사고지점 위치정보 지도 전시 처리 - 새로 수신된 사고지점 객체를 지도화면에 출력처리를 하고 현재 선택된 FOD 객체를 색상으로 구분함 - 새로 수신된 사고지점 객체를 수신 확인 후에 FOD 객체의 색상을 다른 사고지점 객체의 색상과 동일하게 표현함

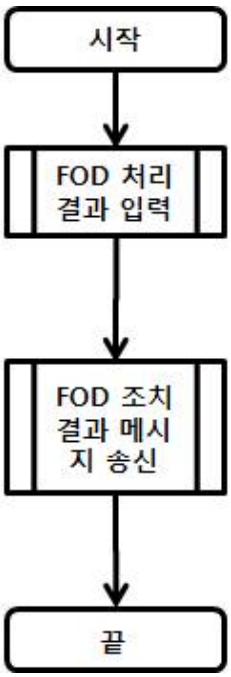
[표 25] 운영센터 메시지 분석 모듈의 순서도 및 기능설명

⑫ FOD 등록 처리 모듈

순서도	
기능 설명	공항 내에서 FOD 발견 시 FOD 등록 처리를 위한 절차를 진행하며, FOD 발견 위치정보는 GPS를 통해서 획득한 자차의 위치정보를 입력하여 운영센터로 전송함 - FOD 등록 시 사진첨부 여부는 사용자의 선택에 의해 진행 - FOD 등록 시 FOD의 위치정보는 자차의 GPS 데이터를 자동으로 입력
분기 조건	없음

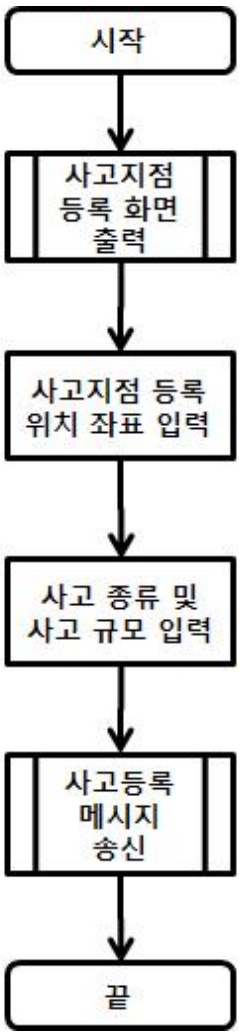
[표 26] FOD 등록 처리 모듈의 순서도 및 기능설명

⑬ FOD 처리 모듈

순서도	
기능 설명	FOD 수거 조치완료 후 관련 정보를 운영센터로 전송
분기 조건	없음

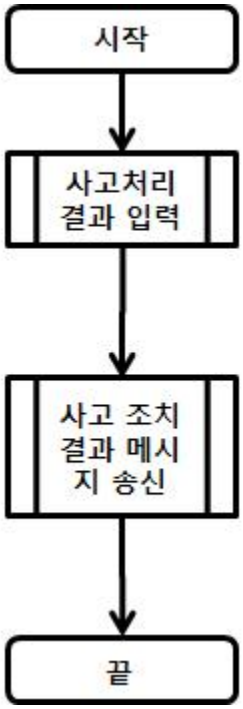
[표 27] FOD 처리 모듈의 순서도 및 기능설명

⑭ 사고지점 등록 처리 모듈

순서도	
기능 설명	공항 내 사고 발생 시 등록 처리를 진행하며, 사고지점의 위치는 사용자가 지도화면에서 선택하여 입력 후 운영센터로 전송 - 사고지점의 위치는 사용자가 지도화면에서 위치를 선택하여 입력 - 사고 종류 및 인명피해 여부 등의 정보를 입력 후 운영센터로 전송
분기 조건	없음

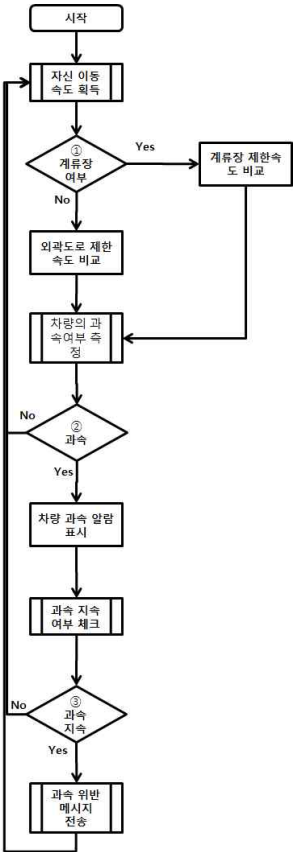
[표 28] 사고지점 등록 처리 모듈의 순서도 및 기능설명

⑮ 사고 처리 모듈

순서도	
기능 설명	사고 조치완료 후 관련 내용을 운영센터로 전송
분기 조건	없음

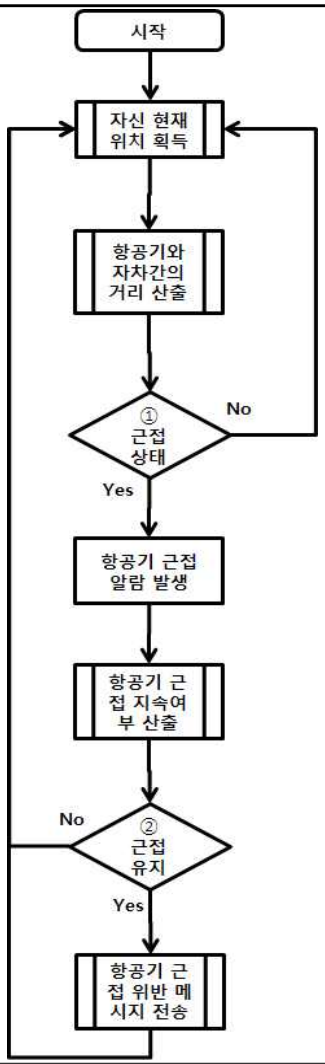
[표 29] 사고 처리 모듈의 순서도 및 기능설명

⑯ 속도위반 확인 모듈

순서도	
기능 설명	자차의 GPS 데이터를 습득하여 차량의 과속여부를 확인함. GIS를 이용하여 구현된 위반 처리 알고리즘을 적용하여 지점별 제한속도를 기준으로 과속여부를 판별함. 차량의 이동속도가 제한속도보다 일정시간 빠르게 확인될 경우 전용단말기에서는 속도위반 경고를 사용자에게 표출하고 운영센터로 관련 정보를 전송
분기 조건	① 현재 위치가 계류장 지역인지 확인 - 현재 차량의 위치가 계류장일 경우 내부도로의 제한속도를 기준으로 차량의 과속여부를 결정 - 현재 차량의 위치가 계류장이 아닐 경우 외곽도로의 제한속도를 기준으로 차량의 과속여부를 결정 ② 자차의 과속 여부 확인 - 현재 속도위반이 아닐 경우 다음 이동속도를 획득 처리 이동 - 현재 속도위반을 하고 있을 경우 차량 운전자에게 속도위반 알람을 제공 ③ 지속적인 과속상태 여부 확인 - 지속적으로 속도위반 상태가 아닐 경우 다음 이동속도를 획득 처리 - 지속적으로 속도위반 상태가 유지될 경우 운영센터로 차량의 속도위반 정보 전송 후 다음 이동속도를 획득 처리

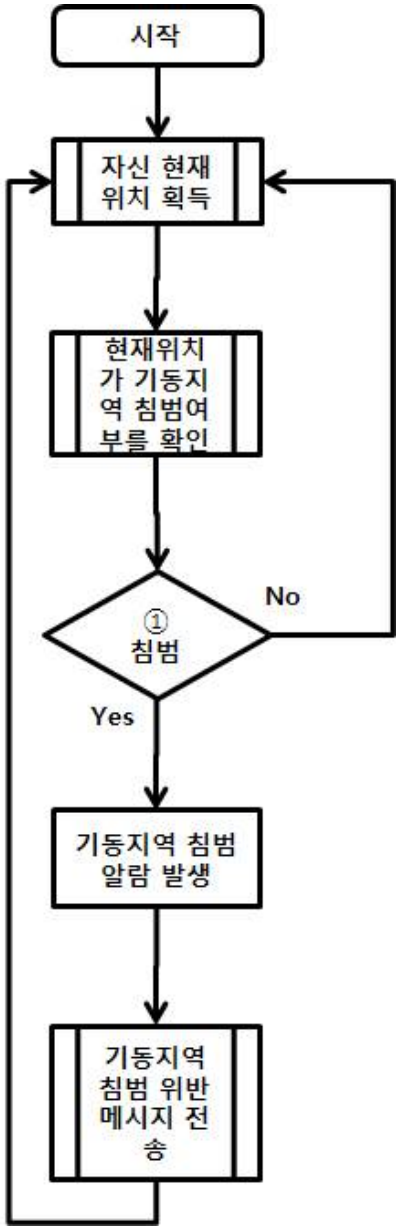
[표 30] 속도위반 확인 모듈의 순서도 및 기능설명

⑰ 항공기 근접 확인 모듈

순서도	
기능 설명	자차의 GPS 위치정보를 사용하여 항공기의 위치와 근접상태를 판단함. 항공안전관리규정에 정의되어 있는 항공기 근접거리를 기준으로 근접여부를 판별함. 차량이 항공기 주변에 일정시간 이상 근접해 있을 경우 전용단말기에서 항공기 근접 경고를 사용자에게 표출 후 운영센터로 관련 정보 전송
분기 조건	① 항공기 근접상태 확인 - 항공기와 근접하지 않을 경우 다음 위치 획득을 준비 - 항공기와 근접상태일 경우 사용자에게 항공기 근접 알람 제공 ② 항공기 근접상태 지속여부 확인 - 항공기 근접상태가 일정시간 이상 유지되지 않을 경우 다음 위치 획득 준비 - 항공기 근접상태가 일정시간 이상 유지될 경우 운영센터에 항공기 근접 위반 정보 전송

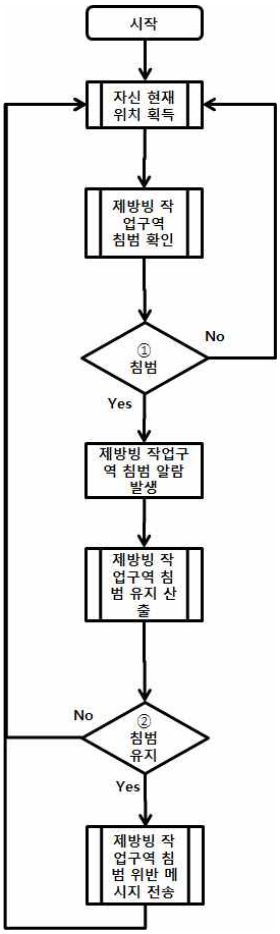
[표 31] 항공기 근접 확인 모듈의 순서도 및 기능설명

⑱ 기동지역 침범 확인 모듈

순서도	
기능 설명	자차의 GPS 위치정보를 사용하여 현재 차량의 위치가 기동지역을 침범하였는지 여부를 판별함. 기동지역 침범을 하였을 경우 전용단말기에서 기동지역 침범 경고를 사용자에게 제공 후 운영센터로 관련 정보 전송
분기 조건	① 자차의 기동지역 침범여부를 확인 - 기동지역 침범을 하지 않았을 경우 다음 위치 획득 준비 - 기동지역을 침범하였을 경우 사용자에게 기동지역 침범 알람 제공 후 운영센터에 관련 정보 전송

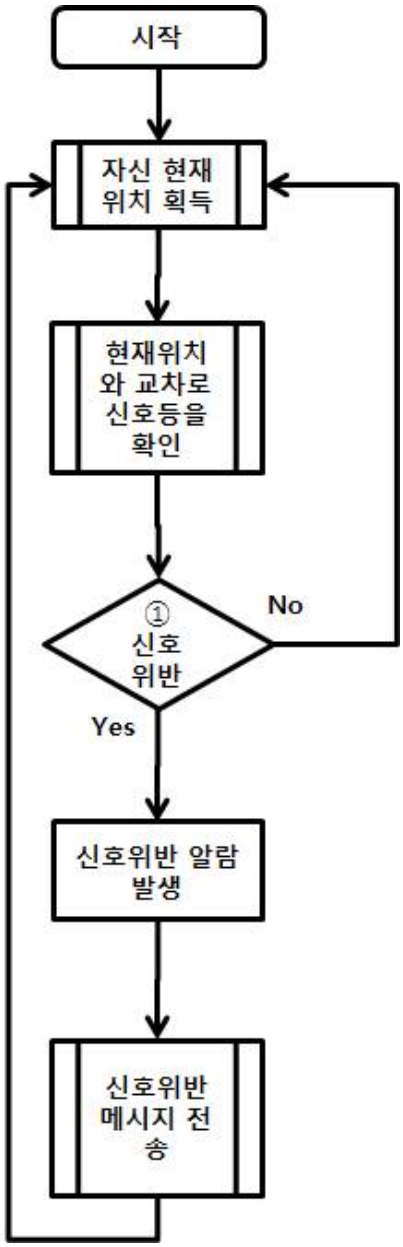
[표 32] 기동지역 침범 확인 모듈의 순서도 및 기능설명

⑲ 제・방빙 작업구역 침범 확인 모듈

순서도	
기능 설명	자차의 GPS 위치정보를 사용하여 현재 차량의 위치가 제・방빙 조업구역을 벗어났는지 여부를 판별함. 제・방빙 조업 금지구역을 침범하였을 경우 전용단말기에서 제・방빙 조업 금지구역 침범 경고를 사용자에게 제공 후 운영센터로 관련 정보 전송(본 기능은 제・방빙 조업차량에만 제공됨)
분기 조건	① 자차의 위치가 제・방빙 조업 금지구역을 침범하였는지 확인 - 제・방빙 조업 금지구역을 침범하지 않았을 경우 다음 위치 획득 준비 - 제・방빙 조업 금지구역을 침범하였을 경우 사용자에게 관련 알람 제공 ② 자차의 위치가 지속적으로 조업 금지구역을 침범하였는지 확인 - 제・방빙 조업 금지구역에서 일정시간 체류하지 않았을 경우 다음 위치 획득 준비 - 제・방빙 조업 금지구역에서 일정시간 이상 체류하였을 경우 운영센터에 관련 정보 전송

[표 33] 제・방빙 작업구역 침범 확인 모듈의 순서도 및 기능설명

⑳ 신호위반 확인 모듈

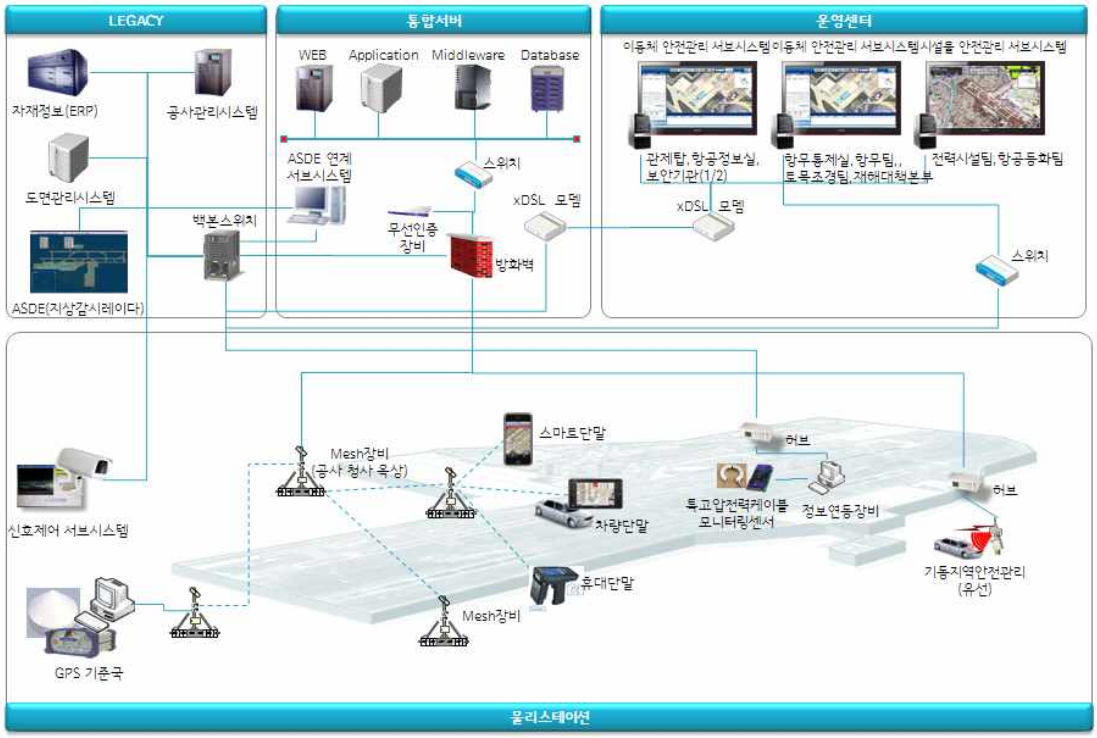
순서도	
기능 설명	자차의 GPS 위치정보와 신호제어 정보를 사용하여 현재 차량이 교차로의 정지신호를 제대로 지키고 있는지 여부를 판별함. 차량이 정지신호를 무시하고 교차로를 이동하였을 경우 전용단말기에서 신호위반 경고를 사용자에게 제공하고 운영센터로 관련 정보 전송
분기 조건	① 신호위반 여부 확인 - 신호위반을 하지 않았을 경우 다음 위치 획득 준비 - 신호위반을 하였을 경우 사용자에게 신호위반 알람 제공 후 운영센터에 관련 정보 전송

[표 34] 신호위반 확인 모듈의 순서도 및 기능설명

(라) 통신 인터페이스 프로토콜 설계

① 공항시설 안전통합 관리시스템 구성도

○ 공항시설 안전통합 관리시스템의 시스템 구성도는 다음과 같음



[그림 12] AFSIS-6 시스템 구성도

○ 공항시설 안전통합 관리시스템을 구성하는 각 서브시스템의 역할은 다음과 같음

구성	서브시스템	설명
물리스테이션 PS (Physical Station)	차량단말 서브시스템 VMS(Vehicle Mobile Subsystem)	차량의 위치정보를 사용자와 센터에 제공하고 센터와 통신을 통해 지시를 받거나 현장정보를 수집
	휴대단말 서브시스템 PMS(Portable Mobile Subsystem)	시설을 점검하는 작업자가 휴대하여 자신 및 시설물의 위치를 확인하고 정보를 조회
	스마트단말 서브시스템 SMS(Smart Mobile Subsystem)	스마트단말에 응용프로그램을 설치하여 항공기 및 자신의 위치를 확인하고 사고 또는 FOD를 등록
	공항전력 장애예방 서브시스템	부분방전센서를 공항 전력시설에 부착하여 감지된 시설상태정보를 통합 DB에 전송하는 기

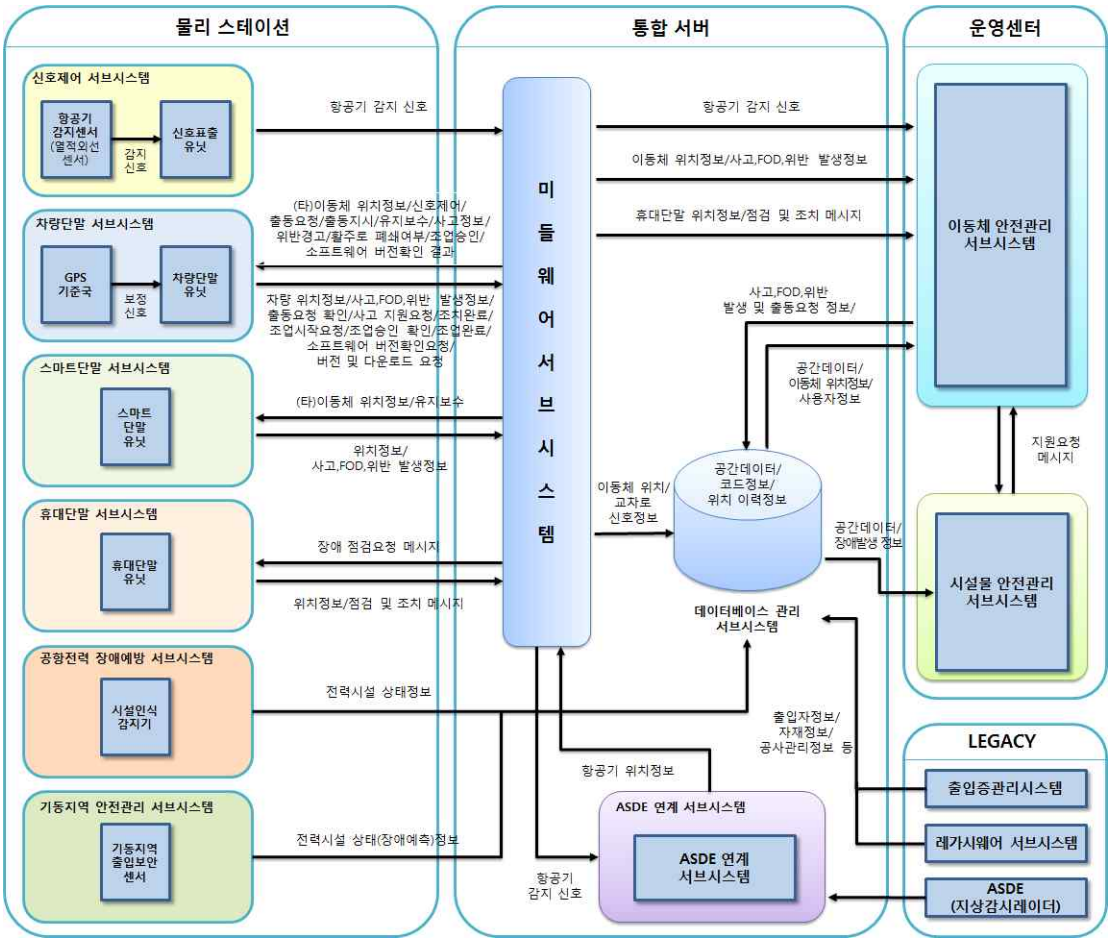
구성	서브시스템	설명
	EFS (Electrical Facility Subsystem)	능으로 공항 전력시설의 장애징후를 사전에 감지하여 안전관리정보를 제공
	기동지역 안전관리 서브시스템 MSS (Movement Safety Subsystem)	GPS방식의 안전관리를 보완하여 항공기 안전운항의 필수시설인 활주로 등의 보호를 위해 비인가 차량의 기동지역 접근을 무선식별장치(RFID)를 이용해 통제하는 등 공항시설 안전도를 제고하는 기능을 제공
	차량단말 GIS 엔진 VGE(Vehicle GIS Engine)	차량단말 서브시스템에 적용되는 차량용 GIS엔진 제공
	신호제어 서브시스템 SCS(Signal Control Subsystem)	항공기-차량 교차로를 주행하는 항공기를 감지하여 정지신호 정보를 제공
통합서버 IS (Integrated Server)	미들웨어 서브시스템 MWS (Middleware Subsystem)	물리스테이션 및 운영센터, ASDE 등에 대해 서버로 동작하며 운영 중 발생하는 패킷을 수신, 분석하여 필요사항을 처리하고 목적지 IP로 패킷을 전달
	레가시웨어 연계 서브시스템 LCS(Legacyware Connect Subsystem)	운영센터 등에서 필요로 하는 자산관련 정보 요청에 대해 ERP 시스템과 연계, 데이터를 추출하여 응답으로 전달
	ADS-B 연계 서브시스템 ABS(ADS-B Connect Subsystem)	ADS-B 항공기정보를 수신하고, AFSIS-6의 차량정보를 ADS-B 메시지로 송신.
	ASDE 연계 서브시스템 ACS(ASDE Connect Subsystem)	각 레이더로부터 데이터를 수신하여 항공기 정보(위치정보, 편명정보, 열감지센서 정보)를 생성하고 미들웨어 서브시스템에 전송
	데이터베이스 관리 서브시스템 DMS(Database Management Subsystem)	AFSIS-6에서 사용되는 정보들을 관리
운영센터 OC (Operation Center)	이동체 안전관리 서브시스템 MOS(Moving Object safety Subsystem)	공항 이동지역의 차량 및 항공기 정보를 모니터링하여 안전도를 향상시킴
	시설물 안전관리 서브시스템 FSS(Facility Safety Subsystem)	공항 시설안전관리를 위해 휴대단말의 위치 및 지하시설의 정보를 확인

구성	서브시스템	설명
	지적정보 서브시스템 LIS(Land Information Subsystem)	지적과 관련된 토지 및 건축물 정보, 공항시설현황을 조회한
	서버용 GIS 엔진 SGE(Sever GIS Engine)	운영센터에 적용되는 서버용 GIS 엔진 제공
네트워크인프라 NI (NetworkInfra)	네트워크 인프라 서브시스템 NIS(Network Infrastructure Subsystem)	AFSIS-6의 네트워크 기본구성으로 서브시스템별 요소를 유기적으로 통합 연결시켜 정보를 전송, 공유 등을 가능하게 함

[표 35] AFSIS-6의 서브시스템 설명

② 공항시설 안전통합 관리시스템 인터페이스 구성도

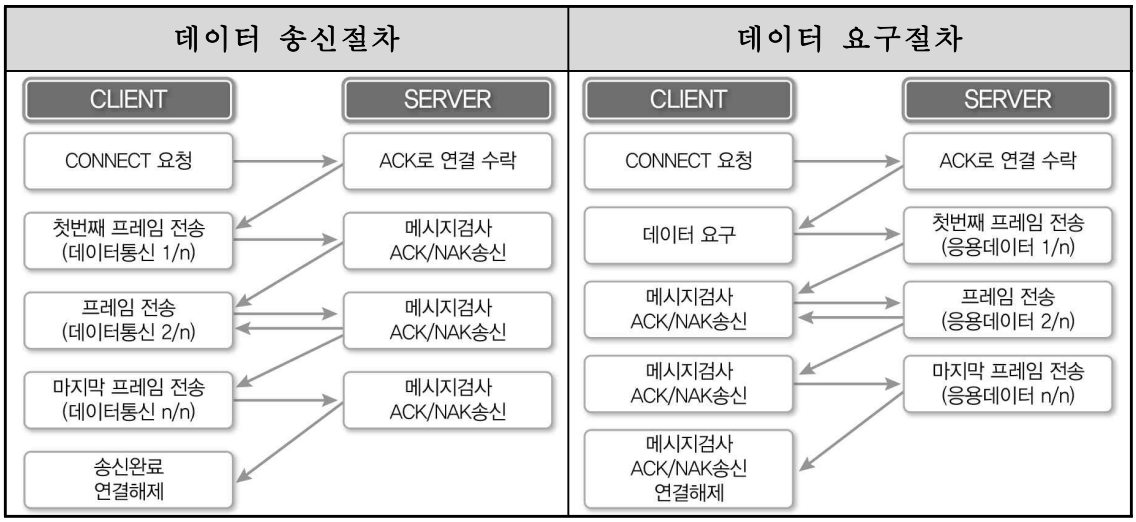
○ 공항시설 안전통합 관리시스템의 하드웨어 간 연계되는 정보 및 이동 방향을 표시한 구성도는 다음과 같음



[그림 13] 서브시스템간의 인터페이스 구성도

③ 프로토콜 정의

○ 통합서버(미들웨어)와 서브시스템(클라이언트)간 통신절차는 다음과 같음



[표 36] 프로토콜 통신절차

- Server는 통합서버(미들웨어)를 말함

- Client는 통합서버(미들웨어)를 제외한 서브시스템을 말함

○ 예외처리

- 통신이 단절 되었을 경우 2초 주기로 재 연결을 시도

- 재 연결 후 전송이 실패한 메시지를 확인하여 우선적으로 전송

- 통합서버(미들웨어)에서는 운영센터에서 전용단말기의 접속 상태를 확인할 수 있도록 연결 상태를 데이터베이스에 기록

○ 프로토콜 구조

구조	헤더										데이터	테일	
상세	STX	Protocol Code	Sender	Receiver Length	Receiver	…	Receiver	Data Code	Data Length	Date	DATA	ETX	CRC16
설명	시작 부호	TCP /UDP 구분자	발신자	수신자 크기	수신자	…	수신자	데이터 종류	데이터 크기	생성 일시	데이터	종료 부호	에러 체크
크기	1	1	4	1	4	…	4	3	2	14	n	1	1

[표 37] 프로토콜 구조

- STX(Start of Transmission, 0x02) : 프레임의 시작을 알리는 기호

- ETX(End of Transmission, 0x03) : 프레임의 마지막을 알리는 기호

- 발신자, 수신자 : byte 타입으로 IP주소의 4자리 숫자들을 연속적으로 표시

- 수신자 크기 : 수신대상의 수

- 데이터종류 : 서비스ID(1byte) + 명령어ID(1byte) + 분류코드(1byte)

- 헤더 크기 : 26byte + (4 x N(수신자 크기))byte

- 테일 크기 : 2byte

- CRC16 계산식

```
// dataSize : 수신한 데이터의 길이
// data[i] : 수신된 데이터 묶음
for (int i = 0; i < dataSize - 1; ++i)
{
   if ((i % 2) == 1)
   {
      checksum += data[i];
   }
   else
   {
      checksum ^= data[i];
   }
}
```

[그림 14] CRC(Cyclic Redundancy Check) 계산

○ 서비스 종류별 명령어 구분

- 공항시설 안전통합 관리시스템은 아래와 같이 7가지의 서비스로 분류됨

서비스 종류	코드값 (Hex)	설명
Airside 이동체 안전관리 서비스	0x11	활주로 침범, 과속, 안전거리 확보
항공기-차량 주요 교차지점 지능형 신호제어 서비스	0x21	항공기 이동 자동 감지 및 신호제어
이동지역 이물질 안전관리 서비스	0x31	공항 내 이물질 발생 시 위치 및 영상정보 제공
제·방빙 등 특수지원 차량 원격관리 서비스	0x41	특수 작업차량의 위치 및 환경사고 예방
비상상황 대응 지원 서비스	0x51	항공사고 발생 시 신속한 대처 가능 및 최적 경로 제공
소프트웨어 유지보수 서비스	0x71	소프트웨어 업데이트 및 재설치 등 유지보수 지원

[표 38] AFSIS-6의 서비스 종류

- 공항시설 안전통합 관리시스템에 대한 메시지 코드 분류표

서비스	코드값 (Hex)	서비스 명령어	서브시스템	
			차량단말	미들웨어
Airside 이동체 안전관리	0x11	차량 위치정보	↔	↔
	0x12	항공기 위치정보	←	↔
	0x13	활주로 폐쇄	←	↔
	0x14	보정신호	←	↔
	0x15	휴대단말 위치	←	↔
	0x16	위반 등록	→	↔
	0x17	경고	←	↔
	0x18	위반 출동요청	←	↔
	0x19	위반 출동요청 확인	→	↔
	0x1A	위반 조치완료	→	↔
	0x1B	차량단말 로그인 확인	→	→

서비스	코드값 (Hex)	서비스 명령어	서브시스템	
			차량단말	미들웨어
	0x1C	차량단말 로그인 확인 결과	←	←
지능형 신호제어	0x21	신호제어	←	↔
FOD/포장면 안전관리	0x31	FOD/포장면 파손 등록	→	↔
	0x32	유지보수 지시	←	↔
	0x33	유지보수 지시 확인	→	↔
	0x34	유지보수 조치완료	→	↔
특수지역 원격관리	0x41	조업 시작요청	→	↔
	0x42	조업승인	←	↔
	0x43	조업승인 확인	→	↔
	0x44	조업 완료보고	→	↔
구조소방 지원	0x51	사고 등록	→	↔
	0x52	상황전파	↔	↔
	0x53	사고 출동지시	←	↔
	0x54	사고 출동지시 확인	→	↔
	0x55	지원요청	→	↔
	0x56	사고 조치완료	→	↔
소프트웨어 유지보수	0x71	버전 확인 요청	→	←
	0x72	버전 확인 결과	←	→
	0x74	다운로드 요청	→	←

[표 39] 서브시스템별 서비스 명령어 흐름

○ 데이터 패킷 유형

- 차량 위치정보

총 데이터 크기(byte)	24 byte				
패킷 구조	차량단말 ID	경도(X좌표)	위도(Y좌표)	차량코드	속도
데이터 형	char[ ]	double	double	char	int(short)
데이터 크기 (byte)	5	8	8	1	2

[표 40] 차량 위치정보 패킷 유형

- 항공기 위치정보

총 데이터 크기(byte)	33 byte x N			
패킷 구조	항공기편명	경도(X좌표)	위도(Y좌표)	속도
데이터 형	char[ ]	char[ ]	char[ ]	char[ ]
데이터 크기 (byte)	8	11	11	3

[표 41] 항공기 위치정보 패킷 유형

- 위반 등록 및 출동요청

총 데이터 크기(byte)	30 byte					
패킷 구조	업무 ID	위반 분류코드	차량단말 ID	경도 (X좌표)	위도 (Y좌표)	기타
데이터 형	char[ ]	char	char[ ]	double	double	int (short)
데이터 크기 (byte)	6	1	5	8	8	2

[표 42] 위반 등록 및 출동요청 패킷 유형

- 위반사항에 대한 경고

총 데이터 크기(byte)	6 byte	
패킷 구조	차량단말 ID	위반 분류코드
데이터 형	char[ ]	char
데이터 크기 (byte)	5	1

[표 43] 위반 사항에 대한 경고 패킷 유형

- 출동요청, 유지보수, 출동지시 및 조업승인에 대한 확인

총 데이터 크기(byte)	7 byte	
패킷 구조	업무 ID	수용/거부
데이터 형	char[ ]	char
데이터 크기 (byte)	6	1

[표 44] 출동요청, 유지보수, 출동지시 및 조업승인에 대한 확인 패킷 유형

- 위반 조치완료

총 데이터 크기(byte)	92 byte			
패킷 구조	업무 ID	차량단말 ID	조치 분류코드	특이사항
데이터 형	char[ ]	char[ ]	char	char[ ]
데이터 크기 (byte)	6	5	1	80

[표 45] 위반 조치완료 패킷 유형

- FOD/포장면 파손 등록

총 데이터 크기(byte)	44 byte							
패킷 구조	업무 ID	차량 단말 ID	FOD 분류코드	경도 (X좌표)	위도 (Y좌표)	이미지 여부	이미지 개수	일시
데이터 형	char[ ]	char[ ]	char	double	double	char	char	char[ ]
데이터 크기 (byte)	6	5	1	8	8	1	1	14

[표 46] FOD/포장면 파손 등록 패킷 유형

- 유지보수 지시

총 데이터 크기(byte)	39 byte						
패킷 구조	업무 ID	조치 분류코드	경도 (X좌표)	위도 (Y좌표)	이미지 여부	이미지 개수	일시
데이터 형	char[ ]	char	double	double	char	char	char[ ]
데이터 크기 (byte)	6	1	8	8	1	1	14

[표 47] 유지보수 지시 패킷 유형

- 유지보수 조치완료

총 데이터 크기(byte)	34 byte							
패킷 구조	업무 ID	차량단 말ID	조치 분류코드	골재 개수	금속 개수	부품 개수	기타 개수	조업 완료일시
데이터 형	char[ ]	char[ ]	char	int (short)	int (short)	int (short)	int (short)	char[ ]
데이터 크기 (byte)	6	5	1	2	2	2	2	14

[표 48] 유지보수 조치완료 패킷 유형

- 사고 등록, 상황전파 및 출동지시

총 데이터 크기(byte)	31 byte							
패킷 구조	업무 ID	차량 단말ID	사고 분류코드	경도 (X좌표)	위도 (Y좌표)	처리 상태	인명 피해	화재 여부
데이터 형	char[ ]	char[ ]	char	double	double	char	char	char
데이터 크기 (byte)	6	5	1	8	8	1	1	1

[표 49] 사고 등록, 상황전파 및 출동지시 패킷 유형

- 사고 관련 지원요청

총 데이터 크기(byte)	16 byte						
패킷 구조	업무 ID	차량 단말 ID	소방차 추가	구급차 추가	사다리차 추가	물탱크차 추가	조명차 추가
데이터 형	char[ ]	char[ ]	char	char	char	char	char
데이터 크기 (byte)	6	5	1	1	1	1	1

[표 50] 사고 관련 지원요청 패킷 유형

- 사고 조치완료

총 데이터 크기(byte)	98 byte					
패킷 구조	업무 ID	차량단말 ID	사고 차량유무	사고 차량대수	사고차량 단말 ID	특이사항
데이터 형	char[ ]	char[ ]	char	char	char[ ]	char[ ]
데이터 크기 (byte)	6	5	1	1	5	80

[표 51] 사고 조치완료 패킷 유형

- 제·방빙 조업 시작요청

총 데이터 크기(byte)	41 byte				
패킷 구조	업무 ID	차량단말 ID	경도 (X좌표)	위도 (Y좌표)	등록일시
데이터 형	char[ ]	char[ ]	double	double	char[ ]
데이터 크기 (byte)	6	5	8	8	14

[표 52] 제·방빙 조업 시작요청 패킷 유형

- 제·방빙 조업승인, 제·방빙 작업완료 보고

총 데이터 크기(byte)	25 byte		
패킷 구조	업무 ID	차량단말 ID	승인일시 or 조업완료 일시
데이터 형	char[ ]	char[ ]	char[ ]
데이터 크기 (byte)	6	5	14

[표 53] 제·방빙 조업승인, 조업 완료보고 패킷 유형

- 소프트웨어 버전 확인 요청

총 데이터 크기(byte)	16 byte		
패킷 구조	장비 ID	장비종류	버전정보
데이터 형	char[ ]	char	char[ ]
데이터 크기 (byte)	5	1	10

[표 54] 소프트웨어 버전 확인 요청 패킷 유형

- 소프트웨어 버전 확인 결과

총 데이터 크기(byte)	6 byte	
패킷 구조	차량단말 ID	버전확인 결과
데이터 형	char[ ]	char
데이터 크기 (byte)	5	1

[표 55] 소프트웨어 버전 확인 결과 패킷 유형

- 로그인 확인 요청

총 데이터 크기(byte)	21 byte	
패킷 구조	차량단말 ID	차량단말 비밀번호
데이터 형	char[ ]	char[]
데이터 크기 (byte)	5	16

[표 56] 로그인 확인 요청 패킷 유형

- 로그인 확인 결과

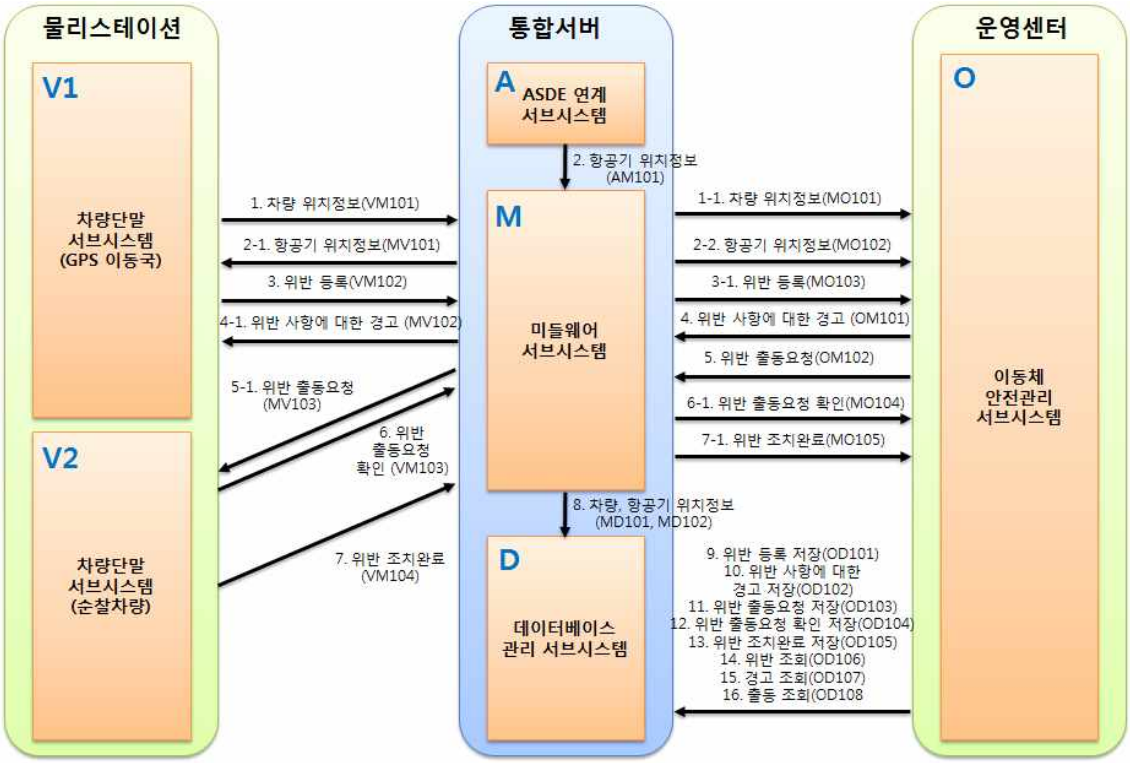
총 데이터 크기(byte)	6 byte	
패킷 구조	차량단말 ID	로그인 확인 결과
데이터 형	char[ ]	char
데이터 크기 (byte)	5	1

[표 57] 로그인 확인 결과 패킷 유형

④ 서비스별 인터페이스

○ Airside 이동체 안전관리 서비스

- Airside 이동체 안전관리 서비스 인터페이스 구성도



[그림 15] Airside 이동체 안전관리 서비스 인터페이스 구성도

- Airside 이동체 안전관리 기술은 위치측위 기술 기반의 이동체 안전관리 시스템으로써 이동지역 내 차량 및 항공기 운항 증가에 따른 위협요인을 제거함
- 활주로 침범, 이동체 속도, 차로 위반 등 실시간 위치정보 및 운행상황 감지
- 안전 위반사항 처리에 대한 자동화 체제 구현

- 통신 인터페이스

• 차량 위치정보

<table>
<tr><th colspan="2">항 목</th><th colspan="5">내 용</th></tr>
<tr><td colspan="2">명령어</td><td colspan="5">차량 위치정보</td></tr>
<tr><td colspan="2">서비스 구분</td><td colspan="5">Airside 이동체 안전관리 서비스</td></tr>
<tr><td colspan="2">명령어 코드값</td><td colspan="5">0x11</td></tr>
<tr><td colspan="2">인터페이스 ID</td><td colspan="2">VM101</td><td colspan="2">MO101</td><td>MV101</td></tr>
<tr><td colspan="2">송신자</td><td colspan="2">차량단말 서브시스템</td><td colspan="2">미들웨어 서브시스템</td><td>미들웨어 서브시스템</td></tr>
<tr><td colspan="2">수신자</td><td colspan="2">미들웨어 서브시스템</td><td colspan="2">이동체 안전관리
서브시스템</td><td>차량단말 서브시스템</td></tr>
<tr><td colspan="2">주요 정보</td><td colspan="5">차량의 위치정보, 속도 등</td></tr>
<tr><td colspan="2">프로토콜</td><td colspan="4">TCP/IP</td><td>UDP</td></tr>
<tr><td colspan="2">주기</td><td colspan="5">1초 당 1회</td></tr>
<tr><td colspan="2">총 데이터 크기</td><td colspan="5">24 byte</td></tr>
<tr><td colspan="2">패킷 구조</td><td>차량단말 ID</td><td>경도(X좌표)</td><td>위도(Y좌표)</td><td>차량코드</td><td>속도</td></tr>
<tr><td colspan="2">데이터 크기</td><td>5 byte</td><td>8 byte</td><td>8 byte</td><td>1 byte</td><td>2 byte</td></tr>
<tr><td rowspan="5">차량
위치
정보</td><td>차량
단말 ID</td><td colspan="5">P0000(순찰차량), F0000(구조소방차량), E0000(구급차량), C0000(FOD차량), D0000(제・방빙차량), G0000(제초차량), S0000(제설차량), W0000(그 외 작업차량)
※ 0000부분은 순차적인 숫자로 구성됨</td></tr>
<tr><td>경도
(X좌표)</td><td colspan="5">소수점 8자리
예) 126.12345678</td></tr>
<tr><td>위도
(Y좌표)</td><td colspan="5">소수점 8자리
예) 37.12345678</td></tr>
<tr><td>차량코드</td><td colspan="5">P(순찰차량), F(구조소방차량), E(구급차량), C(FOD차량), D(제・방빙차량), G(제초차량), S(제설차량), W(그 외 작업차량)</td></tr>
<tr><td>속도</td><td colspan="5">속도(㎞/h)
예) 44</td></tr>
<tr><td colspan="2">비고</td><td colspan="5">없음</td></tr>
</table>

[표 58] 차량 위치정보 인터페이스 정의

- 항공기 위치정보

<table>
<tr><th colspan="2">항 목</th><th colspan="4">내 용</th></tr>
<tr><td colspan="2">명령어</td><td colspan="4">항공기 위치정보</td></tr>
<tr><td colspan="2">서비스 구분</td><td colspan="4">Airside 이동체 안전관리 서비스</td></tr>
<tr><td colspan="2">명령어 코드값</td><td colspan="4">0x12</td></tr>
<tr><td colspan="2">인터페이스 ID</td><td>AM101</td><td colspan="2">MO102</td><td>MV101</td></tr>
<tr><td colspan="2">송신자</td><td>ASDE 연계
서브시스템</td><td colspan="2">미들웨어 서브시스템</td><td>미들웨어 서브시스템</td></tr>
<tr><td colspan="2">수신자</td><td>미들웨어 서브시스템</td><td colspan="2">이동체 안전관리
서브시스템</td><td>차량단말 서브시스템</td></tr>
<tr><td colspan="2">주요 정보</td><td colspan="4">항공기의 위치정보, 편명, 속도 등</td></tr>
<tr><td colspan="2">프로토콜</td><td colspan="3">TCP/IP</td><td>UDP</td></tr>
<tr><td colspan="2">주기</td><td colspan="4">1초 당 1회</td></tr>
<tr><td colspan="2">총 데이터 크기</td><td colspan="4">33 byte x N</td></tr>
<tr><td colspan="2">패킷 구조</td><td>항공기 편명</td><td>경도(X좌표)</td><td>위도(Y좌표)</td><td>속도</td></tr>
<tr><td colspan="2">데이터 크기</td><td>8 byte</td><td>11 byte</td><td>11 byte</td><td>3 byte</td></tr>
<tr><td rowspan="4">항공기
위치
정보</td><td>항공기
편명</td><td colspan="4">항공사명칭 3자리 + 숫자 5자리(숫자가 3자리일 경우 3자리 + 공백 2자리)
예) UFN0267</td></tr>
<tr><td>경도
(X좌표)</td><td colspan="4">11자리
예) 126.793106</td></tr>
<tr><td>위도
(Y좌표)</td><td colspan="4">11자리(위도가 9자리일 경우 : 9자리 + 공백 2자리)
예) 37.558933</td></tr>
<tr><td>속도</td><td colspan="4">속도(㎞/h) 3자리(속도가 1자리 경우 : 1자리 + 공백 2자리)
예) 291</td></tr>
<tr><td colspan="2">비고</td><td colspan="4">N은 위치가 확인된 항공기의 수</td></tr>
</table>

[표 59] 항공기 위치정보 인터페이스 정의

- 위반 등록

<table>
<tr><th colspan="2">항 목</th><th colspan="6">내 용</th></tr>
<tr><td colspan="2">명령어</td><td colspan="6">위반 등록</td></tr>
<tr><td colspan="2">서비스 구분</td><td colspan="6">Airside 이동체 안전관리 서비스</td></tr>
<tr><td colspan="2">명령어 코드값</td><td colspan="6">0x16</td></tr>
<tr><td colspan="2">인터페이스 ID</td><td colspan="3">VM102</td><td colspan="3">MO103</td></tr>
<tr><td colspan="2">송신자</td><td colspan="3">차량단말 서브시스템</td><td colspan="3">미들웨어 서브시스템</td></tr>
<tr><td colspan="2">수신자</td><td colspan="3">미들웨어 서브시스템</td><td colspan="3">이동체 안전관리 서브시스템</td></tr>
<tr><td colspan="2">주요 정보</td><td colspan="6">차량의 위반정보</td></tr>
<tr><td colspan="2">프로토콜</td><td colspan="6">TCP/IP</td></tr>
<tr><td colspan="2">주기</td><td colspan="6">위반 발생 시</td></tr>
<tr><td colspan="2">총 데이터 크기</td><td colspan="6">30 byte</td></tr>
<tr><td colspan="2">패킷 구조</td><td>업무 ID</td><td>위반
분류코드</td><td>차량단말
ID</td><td>경도
(X좌표)</td><td>위도
(Y좌표)</td><td>기타</td></tr>
<tr><td colspan="2">데이터 크기</td><td>6 byte</td><td>1 byte</td><td>5 byte</td><td>8 byte</td><td>8 byte</td><td>2 byte</td></tr>
<tr><td rowspan="6">위반
등록</td><td>업무 ID</td><td colspan="6">위반 등록 시 ID는 000000을 사용</td></tr>
<tr><td>위반
분류코드</td><td colspan="6">11(과속), 12(신호위반), 13(항공기 근접), 14(출입구역 위반), 15(기동지역 침범), 16(제・방빙 금지구역 침범)</td></tr>
<tr><td>차량단말
ID</td><td colspan="6">위반 차량의 차량단말 ID</td></tr>
<tr><td>경도
(X좌표)</td><td colspan="6">소수점 8자리
예) 126.12345678</td></tr>
<tr><td>위도
(Y좌표)</td><td colspan="6">소수점 8자리
예) 37.12345678</td></tr>
<tr><td>기타</td><td colspan="6">과속일 경우 속도 정보를 추가
(나머지 위반 사항에 대해서는 사용되지 않음)</td></tr>
<tr><td colspan="2">비고</td><td colspan="6">위반 분류코드는 ASCII 코드를 사용</td></tr>
</table>

[표 60] 위반 등록 인터페이스 정의

• 위반사항에 대한 경고

항 목		내 용	
명령어		위반 사항에 대한 경고	
서비스 구분		Airside 이동체 안전관리 서비스	
명령어 코드값		0x17	
인터페이스 ID		OM101	MV102
송신자		이동체 안전관리 서브시스템	미들웨어 서브시스템
수신자		미들웨어 서브시스템	차량단말 서브시스템
주요 정보		위반 사항에 대한 경고 및 주의	
프로토콜		TCP/IP	
주기		발생 시	
총 데이터 크기		6 byte	
패킷 구조		차량단말 ID	위반 분류코드
데이터 크기		5 byte	1 byte
경고	차량단말 ID	추후 확인용으로 차량단말 ID 전송	
	위반 분류코드	11(과속), 12(신호위반), 13(항공기 근접), 14(출입구역 위반), 15(기동지역 침범), 16(제・방빙 금지구역 침범)	
비고		위반 분류코드는 ASCII 코드를 사용	

[표 61] 위반 사항에 대한 경고 인터페이스 정의

- 위반 출동요청

항 목		내 용					
명령어		위반 출동요청					
서비스 구분		Airside 이동체 안전관리 서비스					
명령어 코드값		0x18					
인터페이스 ID		OM102			MV103		
송신자		이동체 안전관리 서브시스템			미들웨어 서브시스템		
수신자		미들웨어 서브시스템			차량단말 서브시스템		
주요 정보		위반 관련 출동요청 사항					
프로토콜		TCP/IP					
주기		발생 시					
총 데이터 크기		30 byte					
패킷 구조		업무 ID	위반 분류코드	차량단말 ID	경도 (X좌표)	위도 (Y좌표)	기타
데이터 크기		6 byte	1 byte	5 byte	8 byte	8 byte	2 byte
출동 요청	업무 ID	이동체 안전관리 서브시스템에서 부여한 ID 사용					
	위반 분류코드	11(과속), 12(신호위반), 13(항공기 근접), 14(출입구역 위반), 15(기동지역 침범), 16(제・방빙 금지구역 침범)					
	차량단말 ID	위반 차량의 차량단말 ID					
	경도 (X좌표)	소수점 8자리 예) 126.12345678					
	위도 (Y좌표)	소수점 8자리 예) 37.12345678					
	기타	과속일 경우 속도 정보를 추가 (나머지 위반 사항에 대해서는 사용되지 않음)					
비고		위반 분류코드는 ASCII 코드를 사용					

[표 62] 위반 출동요청 인터페이스 정의

- 위반 출동요청 확인

항 목		내 용	
명령어		위반 출동요청 확인	
서비스 구분		Airside 이동체 안전관리 서비스	
명령어 코드값		0x19	
인터페이스 ID		VM103	MO104
송신자		차량단말 서브시스템	미들웨어 서브시스템
수신자		미들웨어 서브시스템	이동체 안전관리 서브시스템
주요 정보		위반 관련 출동요청에 대한 수용 및 거부	
프로토콜		TCP/IP	
주기		발생 시	
총 데이터 크기		7 byte	
패킷 구조		업무 ID	수용/거부
데이터 크기		6 byte	1 byte
출동 요청 확인	업무 ID	출동요청 관련 ID	
	수용/거부	0(수용), 1(거부), 2(미확인)	
비고		없음	

[표 63] 위반 출동요청 확인 인터페이스 정의

- 위반 조치완료

항 목		내 용			
명령어		위반 조치완료			
서비스 구분		Airside 이동체 안전관리 서비스			
명령어 코드값		0x1A			
인터페이스 ID		VM104		MO105	
송신자		차량단말 서브시스템		미들웨어 서브시스템	
수신자		미들웨어 서브시스템		이동체 안전관리 서브시스템	
주요 정보		위반 차량에 대한 조치결과 및 특이사항			
프로토콜		TCP/IP			
주기		발생 시			
총 데이터 크기		92 byte			
패킷 구조		업무 ID	차량단말 ID	조치 분류코드	특이사항
데이터 크기		6 byte	5 byte	1 byte	80 byte
조치 완료	업무 ID	위반 사항에 대한 출동요청 관련 ID			
	차량 단말 ID	담당자의 차량단말 ID			
	조치 분류코드	4(구두경고), 5(고지서 발급), 6(정비지시)			
	특이사항	텍스트 입력을 지원함(80byte 이내)			
비고		없음			

[표 64] 위반 조치완료 인터페이스 정의

• 로그인 확인 요청

항 목		내 용	
명령어		차량단말 로그인 확인 요청	
서비스 구분		Airside 이동체 안전관리 서비스	
명령어 코드값		0x1B	
인터페이스 ID		VM	
송신자		차량단말 서브시스템	
수신자		미들웨어 서브시스템	
주요 정보		차량단말의 로그인 확인 정보	
프로토콜		TCP/IP	
주기		발생 시	
총 데이터 크기		21 byte	
패킷 구조		차량단말 ID	차량단말 비밀번호
데이터 크기		5 byte	16 byte
확인 요청	차량단말 ID	차량단말 로그인 ID	
	차량단말 비밀번호	차량단말 로그인 비밀번호	
비고		없음	

[표 65] 로그인 확인 요청 인터페이스 정의

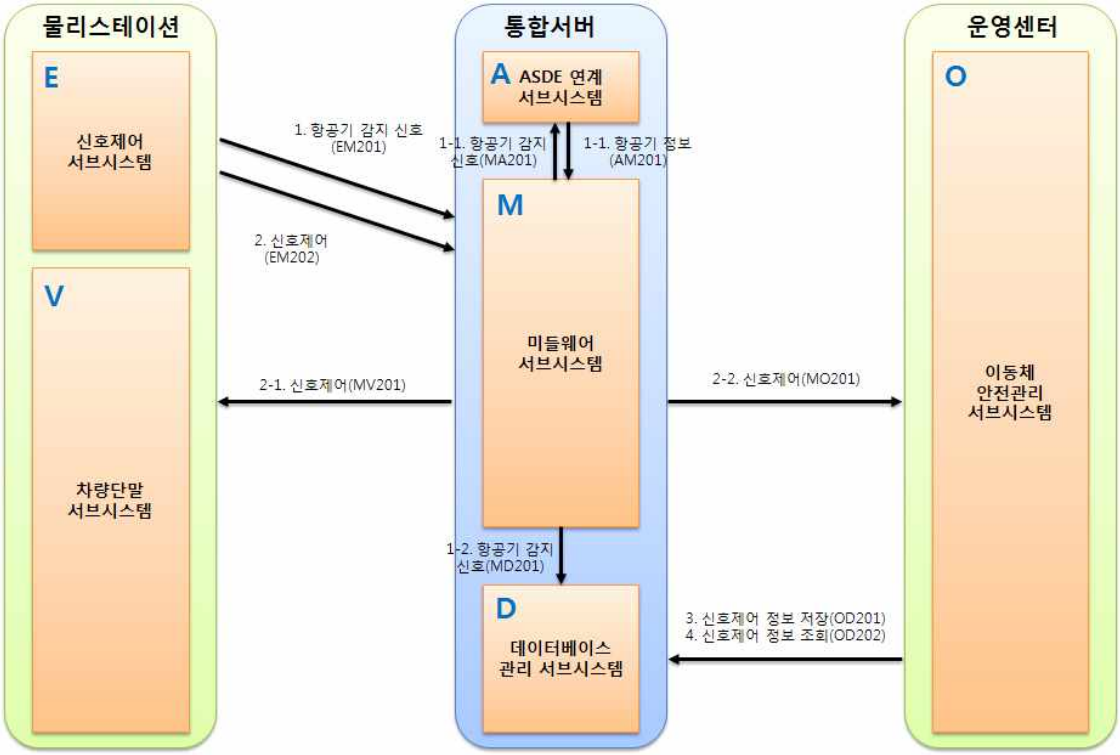
• 로그인 확인 결과

항 목		내 용	
명령어		차량단말 로그인 확인 결과	
서비스 구분		Airside 이동체 안전관리 서비스	
명령어 코드값		0x1C	
인터페이스 ID		MV	
송신자		미들웨어 서브시스템	
수신자		차량단말 서브시스템	
주요 정보		차량단말의 로그인 확인 결과	
프로토콜		TCP/IP	
주기		발생 시	
총 데이터 크기		6 byte	
패킷 구조		차량단말 ID	로그인 확인 결과
데이터 크기		5 byte	1 byte
확인 결과	차량 단말 ID	차량단말 로그인 ID	
	로그인 확인결과	차량단말의 로그인 유무 : Y(성공), F(실패)	
비고		로그인 확인 결과는 대소문자를 구별함	

[표 66] 로그인 확인 결과 인터페이스 정의

○ 지능형 신호제어 서비스

- 지능형 신호제어 서비스 인터페이스 구성도



[그림 16] 지능형 신호제어 서비스 인터페이스 구성도

- 항공기-차량 교차로 신호제어 기술은 센서 기반 지능형 신호제어 기술로써 기존 수동 제어 방식의 개선을 통해 불의의 사고를 미연에 방지하고 관제사 및 지상조업차량의 업무효율을 증대함
- 신호제어기 동작상황 모니터링 및 제어
- 이동지역 내 교차지점 항공기 및 조업차량 자동 통제
- 이동체의 불필요한 대기시간 최소화

- 통신 인터페이스

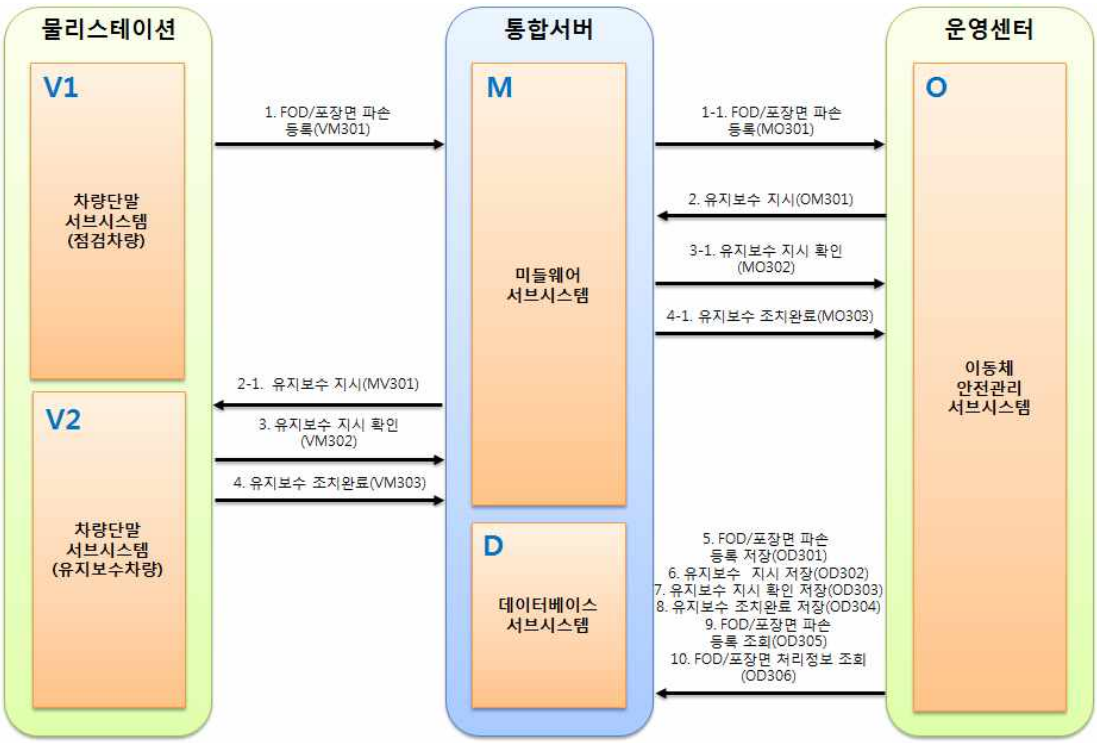
• 신호제어

항 목		내 용			
명령어		신호제어			
서비스 구분		지능형 신호제어 서비스			
명령어 코드값		0x21			
인터페이스 ID		EM202	MV201	MO201	
송신자		신호제어 서브시스템	미들웨어 서브시스템	미들웨어 서브시스템	
수신자		미들웨어 서브시스템	차량단말 서브시스템	이동체 안전관리 서브시스템	
주요 정보		교차로 신호등의 적색 신호 표출정보			
프로토콜		TCP/IP			
주기		발생 시			
총 데이터 크기		23 byte			
패킷 구조		신호등 ID	센서 ID	신호상태	일시
데이터 크기		5 byte	5 byte	1 byte	12 byte
신호 제어 정보	신호등ID	교차로 신호등 ID			
	센서ID	열감지 센서 ID			
	신호상태	53(적색신호 표출), 54(해제)			
	일시	신호제어 일시			
비고		없음			

[표 67] 신호제어 인터페이스 정의

○ FOD/포장면 안전관리 서비스

- FOD/포장면 안전관리 서비스 인터페이스 구성도



[그림 17] FOD/포장면 안전관리 서비스 인터페이스 구성도

- 이동지역 이물질(FOD) 안전관리 기술은 모바일 단말기와 네트워크 통신을 기반으로 이물질 또는 포장면 파손을 능동적으로 관리할 수 있도록 함
- 이물질 발생현장의 위치 및 상황정보 파악
- 이물질 및 포장면 파손 정보를 데이터베이스화하여 체계적 안전관리 및 이력관리
- 모바일 단말기를 통한 신속한 대응

- 통신 인터페이스

• FOD/포장면 파손 등록

항 목		내 용							
명령어		FOD/포장면 파손 등록							
서비스 구분		FOD/포장면 안전관리 서비스							
명령어 코드값		0x31							
인터페이스 ID		VM301				MO301			
송신자		차량단말 서브시스템				미들웨어 서브시스템			
수신자		미들웨어 서브시스템				이동체 안전관리 서브시스템			
주요 정보		FOD/포장면 파손 관련 사항							
프로토콜		TCP/IP							
주기		발생 시							
총 데이터 크기		44 byte							
패킷 구조		업무 ID	차량 단말 ID	FOD 분류코드	경도 (X좌표)	위도 (Y좌표)	이미지 여부	이미지 개수	일시
데이터 크기		6 byte	5 byte	1 byte	8 byte	8 byte	1 byte	1 byte	14 byte
FOD/ 포장면 파손 등록	업무 ID	FOD/포장면 파손 등록 시 ID는 000000을 사용							
	차량단말 ID	FOD/포장면 파손 정보 등록자의 차량단말 ID							
	분류코드	41(이물질 발견), 42(포장면 파손), 43(복합)							
	경도 (X좌표)	소수점 8자리 예) 126.12345678							
	위도 (Y좌표)	소수점 8자리 예) 37.12345678							
	이미지 여부	이미지 파일명은 "일시+일련번호(0부터 시작)"로 명명 있음(1), 없음(0)							
	이미지 개수	이미지 파일 개수							
	일시	FOD/포장면 파손 등록 일시							
비고		이미지의 경우 네트워크 드라이브를 이용							

[표 68] FOD/포장면 파손 등록 인터페이스 정의

- 유지보수 지시

항 목	내 용						
명령어	유지보수 지시						
서비스 구분	FOD/포장면 안전관리 서비스						
명령어 코드값	0x32						
인터페이스 ID	OM301			MV301			
송신자	이동체 안전관리 서브시스템			미들웨어 서브시스템			
수신자	미들웨어 서브시스템			차량단말 서브시스템			
주요 정보	FOD/포장면 파손 관련 유지보수 지시 사항						
프로토콜	TCP/IP						
주기	발생 시						
총 데이터 크기	39 byte						
패킷 구조	업무 ID	조치 분류코드	경도 (X좌표)	위도 (Y좌표)	이미지 여부	이미지 개수	일시
데이터 크기	6 byte	1 byte	8 byte	8 byte	1 byte	1 byte	14 byte
유지보수 지시	업무 ID	이동체 안전관리 서브시스템에서 부여한 ID 사용					
	조치 분류코드	61(청소작업), 62(포장작업), 63(도색작업)					
	경도 (X좌표)	소수점 8자리 예) 126.12345678					
	위도 (Y좌표)	소수점 8자리 예) 37.12345678					
	이미지 여부	이미지 파일명은 "일시+일련번호(0부터 시작)"로 명명					
	이미지 개수	이미지 파일 개수					
	일시	FOD/포장면 파손 등록 일시					
비고	이미지의 경우 네트워크 드라이브를 이용						

[표 69] 유지보수 지시 인터페이스 정의

- 유지보수 지시 확인

항 목		내 용	
명령어		유지보수 지시 확인	
서비스 구분		FOD/포장면 안전관리 서비스	
명령어 코드값		0x33	
인터페이스 ID		VM302	MO302
송신자		차량단말 서브시스템	미들웨어 서브시스템
수신자		미들웨어 서브시스템	이동체 안전관리 서브시스템
주요 정보		FOD/포장면 파손 관련 유지보수 지시에 대한 수용 또는 거부 사항	
프로토콜		TCP/IP	
주기		발생 시	
총 데이터 크기		7 byte	
패킷 구조		업무 ID	수용/거부
데이터 크기		6 byte	1 byte
유지보수 지시 확인	업무 ID	유지보수 지시 관련 업무 ID	
	수용/거부	유지보수 지시 관련 수용 또는 거부 사항 0(수용), 1(거부), 2(미확인)	
비고		없음	

[표 70] 유지보수 지시 확인 인터페이스 정의

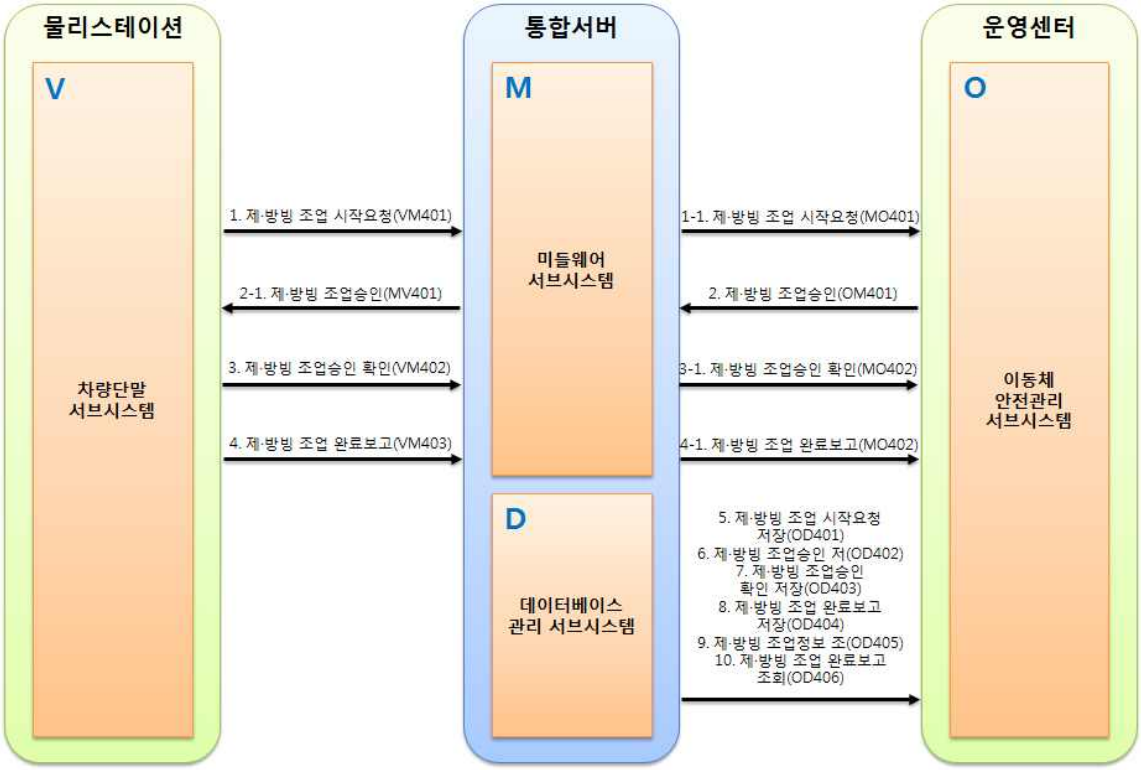
- 유지보수 조치완료

항 목	내 용							
명령어	유지보수 조치완료							
서비스 구분	FOD/포장면 안전관리 서비스							
명령어 코드값	0x34							
인터페이스 ID	VM303				MO303			
송신자	차량 단말 서브시스템				미들웨어 서브시스템			
수신자	미들웨어 서브시스템				이동체 안전관리 서브시스템			
주요 정보	FOD/포장면 파손 관련 유지보수 조치완료 사항							
프로토콜	TCP/IP							
주기	발생 시							
총 데이터 크기	34 byte							
패킷 구조	업무 ID	차량단말 ID	조치분류 코드	골재 개수	금속 개수	부품 개수	기타 개수	조업완료 일시
데이터 크기	6 byte	5 byte	1 byte	2 byte	2 byte	2 byte	2 byte	14 byte
유지보수 조치완료	업무 ID	유지보수 관련 업무 ID						
	차량단말ID	담당자의 차량단말 ID						
	조치 분류코드	61(청소작업), 62(포장작업), 63(도색작업)						
	골재 개수	수거된 골재 개수						
	금속 개수	수거된 금속 개수						
	부품 개수	수거된 부품 개수						
	기타 개수	수거된 기타 FOD 개수						
	조업완료 일시	유지보수 조업완료 일시						
비고	이미지의 경우 네트워크 드라이브를 이용							

[표 71] 유지보수 조치완료 인터페이스 정의

○ 제・방빙 차량 원격관리 서비스

- 제・방빙 차량 원격관리 서비스 인터페이스 구성도



[그림 18] 제・방빙 차량 원격관리 서비스 인터페이스 구성도

- 제・방빙(De-Icing) 등 특수지원 차량관리 기술을 통해 이동지역 내 특수차량의 위치정보, 작업 일시 및 특정지점 체류시간 등을 실시간 모니터링, 위반 사항의 발생을 분석함
- 제・방빙 등 특수 조업차량의 고유ID 및 속성 정보 확인
- 지정된 De-Icing 패드 표시
- 실시간 위치확인을 통한 효율적 업무 배치

- 통신 인터페이스

• 제・방빙 조업 시작요청

항 목		내 용				
명령어		제・방빙 조업 시작 요청				
서비스 구분		제・방빙 차량 원격관리 서비스				
명령어 코드값		0x41				
인터페이스 ID		VM401		MO401		
송신자		차량단말 서브시스템		미들웨어 서브시스템		
수신자		미들웨어 서브시스템		이동체 안전관리 서브시스템		
주요 정보		제・방빙 조업 시 시작 요청				
프로토콜		TCP/IP				
주기		발생 시				
총 데이터 크기		41 byte				
패킷 구조		업무 ID	차량단말 ID	경도(X좌표)	위도(Y좌표)	등록일시
데이터 크기		6 byte	5 byte	8 byte	8 byte	14 byte
조업 시작 요청	업무 ID	시작요청 ID는 000000 사용				
	차량 단말 ID	제・방빙 차량의 차량단말 ID				
	경도 (X좌표)	소수점 8자리 예) 126.12345678				
	위도 (Y좌표)	소수점 8자리 예) 37.12345678				
	등록일시	조업 시작요청 일시				
비고		없음				

[표 72] 제・방빙 조업 시작요청 인터페이스 정의

• 제・방빙 조업승인

항 목		내 용	
명령어		제・방빙 조업승인 확인	
서비스 구분		제・방빙 차량 원격관리 서비스	
명령어 코드값		0x43	
인터페이스 ID		VM402	MO402
송신자		차량단말 서브시스템	미들웨어 서브시스템
수신자		미들웨어 서브시스템	이동체 안전관리 서브시스템
주요 정보		제・방빙 조업승인에 대한 수용정보	
프로토콜		TCP/IP	
주기		발생 시	
총 데이터 크기		7 byte	
패킷 구조		업무 ID	수용/거부
데이터 크기		6 byte	1 byte
조업 승인 확인	업무 ID	조업 시작요청 관련 업무 ID	
	수용/거부	조업승인 확인 시 항상 수용해야 함 0(수용), 1(미확인)	
비고		없음	

[표 73] 제・방빙 조업승인 확인 인터페이스 정의

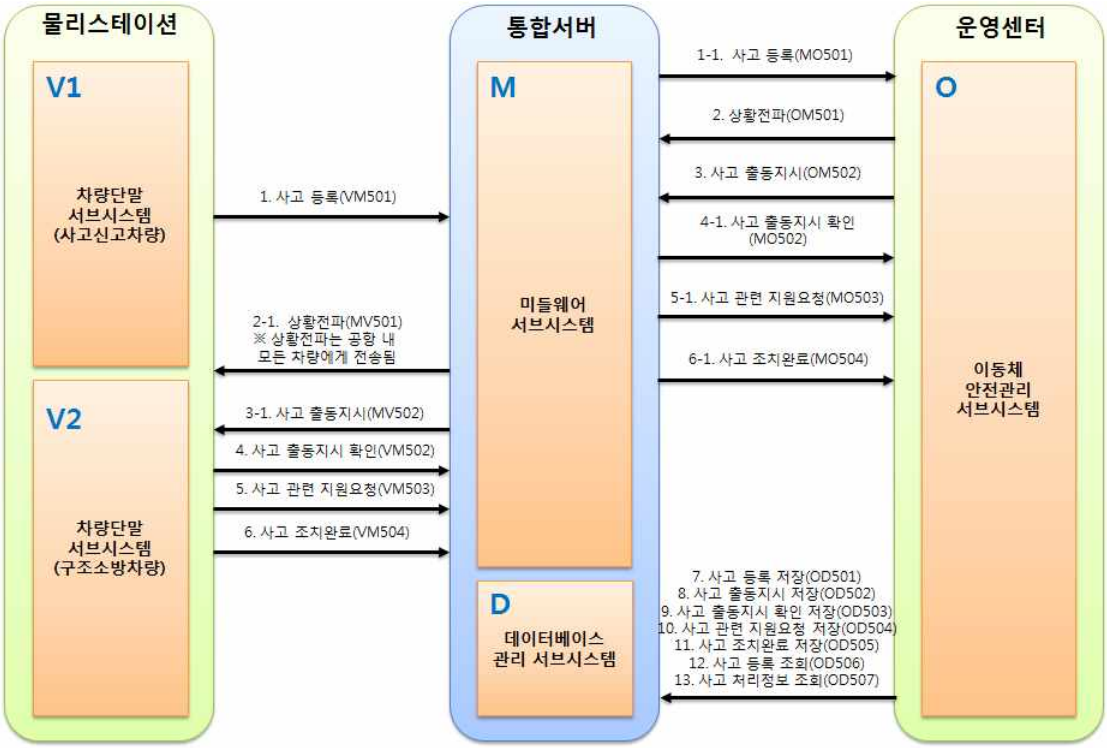
• 제・방빙 조업 완료보고

항 목		내 용		
명령어		제・방빙 조업 완료보고		
서비스 구분		제・방빙 차량 원격관리 서비스		
명령어 코드값		0x44		
인터페이스 ID		VM403	MO402	
송신자		차량단말 서브시스템	미들웨어 서브시스템	
수신자		미들웨어 서브시스템	이동체 안전관리 서브시스템	
주요 정보		제・방빙 조업에 대한 완료보고 사항		
프로토콜		TCP/IP		
주기		발생 시		
총 데이터 크기		25 byte		
패킷 구조		업무 ID	차량단말 ID	조업완료 일시
데이터 크기		6 byte	5 byte	14 byte
조업 완료 보고	업무 ID	조업 시작요청 관련 ID		
	차량 단말 ID	담당자 ID		
	조업완료 일시	조업완료 일시		
비고		없음		

[표 74] 제・방빙 조업 완료보고 인터페이스 정의

○ 비상상황 대응 지원 서비스

- 비상상황 대응 지원 서비스 인터페이스 구성도



[그림 19] 비상상황 대응 지원 서비스 인터페이스 구성도

- 사고 등의 비상상황에 대한 대응 지원 기술을 통해 외부협정기관의 구급 및 소방차에 목표지점의 정확한 위치, 최적경로, 상황정보, 조치사항을 안내하여 신속한 조치가 가능하도록 함
- 외부 지원 차량에 사고 상황 단문메시지 정보 제공
- 외부 지원차량에 GIS 기반 공항지도 및 사고현장까지 최적경로 제공
- 통제센터와 현장간의 데이터 교환

- 통신 인터페이스

• 사고 등록

<table>
<tr><th colspan="2">항 목</th><th colspan="8">내 용</th></tr>
<tr><td colspan="2">명령어</td><td colspan="8">사고 등록</td></tr>
<tr><td colspan="2">서비스 구분</td><td colspan="8">비상상황 대응 지원 서비스</td></tr>
<tr><td colspan="2">명령어 코드값</td><td colspan="8">0x51</td></tr>
<tr><td colspan="2">인터페이스ID</td><td colspan="4">VM501</td><td colspan="4">MO501</td></tr>
<tr><td colspan="2">송신자</td><td colspan="4">차량단말 서브시스템</td><td colspan="4">미들웨어 서브시스템</td></tr>
<tr><td colspan="2">수신자</td><td colspan="4">미들웨어 서브시스템</td><td colspan="4">이동체 안전관리 서브시스템</td></tr>
<tr><td colspan="2">주요 정보</td><td colspan="8">비상상황 발생 시 사고 등록</td></tr>
<tr><td colspan="2">프로토콜</td><td colspan="8">TCP/IP</td></tr>
<tr><td colspan="2">주기</td><td colspan="8">발생 시</td></tr>
<tr><td colspan="2">총 데이터크기</td><td colspan="8">31 byte</td></tr>
<tr><td colspan="2">패킷 구조</td><td>업무 ID</td><td>차량
단말 ID</td><td>사고
분류코드</td><td>경도
(X좌표)</td><td>위도
(Y좌표)</td><td>처리
상태</td><td>인명
피해</td><td>화재
여부</td></tr>
<tr><td colspan="2">데이터 크기</td><td>6 byte</td><td>5 byte</td><td>1 byte</td><td>8 byte</td><td>8 byte</td><td>1 byte</td><td>1 byte</td><td>1 byte</td></tr>
<tr><td rowspan="8">사고
등록</td><td>업무 ID</td><td colspan="8">사고 등록 시 ID는 000000 사용</td></tr>
<tr><td>차량단말
ID</td><td colspan="8">사고 등록자의 차량단말 ID</td></tr>
<tr><td>사고
분류코드</td><td colspan="8">20(항공기 단일), 21(항공기-항공기), 22(항공기-차량), 23(항공기-시설),
24(차량-차량), 25(차량-시설), 26(복합)</td></tr>
<tr><td>경도
(X좌표)</td><td colspan="8">소수점 8자리
예) 126.12345678</td></tr>
<tr><td>위도
(Y좌표)</td><td colspan="8">소수점 8자리
예) 37.12345678</td></tr>
<tr><td>처리상태</td><td colspan="8">10(조치완료), 11(등록), 12(조치 중)</td></tr>
<tr><td>인명피해</td><td colspan="8">0(사상자 없음), 1(사상자 발생), 2(다수 사상자)</td></tr>
<tr><td>화재여부</td><td colspan="8">30(화재 없음), 31(소화 중), 32(진화완료)</td></tr>
<tr><td colspan="2">비고</td><td colspan="8">사고 분류코드는 ASCII 코드를 사용</td></tr>
</table>

[표 75] 사고 등록 인터페이스 정의

- 사고 상황전파

항 목		내 용							
명령어		사고 상황전파							
서비스 구분		비상상황 대응 지원 서비스							
명령어 코드값		0x52							
인터페이스 ID		OM501				MV501			
송신자		이동체 안전관리 서브시스템				미들웨어 서브시스템			
수신자		미들웨어 서브시스템				차량단말 서브시스템			
주요 정보		사고 발생정보							
프로토콜		TCP/IP				UDP			
주기		발생 시							
총 데이터 크기		31 byte							
패킷 구조		업무 ID	차량 단말 ID	사고 분류코드	경도 (X좌표)	위도 (Y좌표)	처리 상태	인명 피해	화재 여부
데이터 크기		6 byte	5 byte	1 byte	8 byte	8 byte	1 byte	1 byte	1 byte
사고 상황 전파	업무 ID	이동체 안전관리 서브시스템에서 부여한 ID 사용							
	차량 단말 ID	사고 등록자의 차량단말 ID							
	사고 분류코드	20(항공기 단일), 21(항공기-항공기), 22(항공기-차량), 23(항공기-시설), 24(차량-차량), 25(차량-시설), 26(복합)							
	경도 (X좌표)	소수점 8자리 예) 126.12345678							
	위도 (Y좌표)	소수점 8자리 예) 37.12345678							
	처리상태	10(조치완료), 11(등록), 12(조치 중)							
	인명피해	0(사상자 없음), 1(사상자 발생), 2(다수 사상자)							
	화재여부	30(화재 없음), 31(소화 중), 32(진화완료)							
비고		사고 분류코드는 ASCII 코드를 사용							

[표 76] 사고 상황전파 인터페이스 정의

- 사고 출동지시

<table>
<tr><th colspan="2">항 목</th><th colspan="8">내 용</th></tr>
<tr><td colspan="2">명령어</td><td colspan="8">사고 출동지시</td></tr>
<tr><td colspan="2">서비스 구분</td><td colspan="8">비상상황 대응 지원 서비스</td></tr>
<tr><td colspan="2">명령어 코드값</td><td colspan="8">0x53</td></tr>
<tr><td colspan="2">인터페이스 ID</td><td colspan="4">OM502</td><td colspan="4">MV502</td></tr>
<tr><td colspan="2">송신자</td><td colspan="4">이동체 안전관리 서브시스템</td><td colspan="4">미들웨어 서브시스템</td></tr>
<tr><td colspan="2">수신자</td><td colspan="4">미들웨어 서브시스템</td><td colspan="4">차량 단말 서브시스템</td></tr>
<tr><td colspan="2">주요 정보</td><td colspan="8">사고 처리에 대한 출동지시</td></tr>
<tr><td colspan="2">프로토콜</td><td colspan="8">TCP/IP</td></tr>
<tr><td colspan="2">주기</td><td colspan="8">발생 시</td></tr>
<tr><td colspan="2">총 데이터 크기</td><td colspan="8">31 byte</td></tr>
<tr><td colspan="2">패킷 구조</td><td>업무 ID</td><td>차량단말 ID</td><td>사고 분류코드</td><td>경도 (X좌표)</td><td>위도 (Y좌표)</td><td>처리 상태</td><td>인명 피해</td><td>화재 여부</td></tr>
<tr><td colspan="2">데이터 크기</td><td>6 byte</td><td>5 byte</td><td>1 byte</td><td>8 byte</td><td>8 byte</td><td>1 byte</td><td>1 byte</td><td>1 byte</td></tr>
<tr><td rowspan="8">사고 출동 지시</td><td>업무 ID</td><td colspan="8">이동체 안전관리 서브시스템에서 부여한 ID 사용</td></tr>
<tr><td>차량 단말 ID</td><td colspan="8">사고 등록자의 차량단말 ID</td></tr>
<tr><td>사고 분류코드</td><td colspan="8">20(항공기 단일), 21(항공기-항공기), 22(항공기-차량), 23(항공기-시설), 24(차량-차량), 25(차량-시설), 26(복합)</td></tr>
<tr><td>경도 (X좌표)</td><td colspan="8">소수점 8자리
예) 126.12345678</td></tr>
<tr><td>위도 (Y좌표)</td><td colspan="8">소수점 8자리
예) 37.12345678</td></tr>
<tr><td>처리상태</td><td colspan="8">10(조치완료), 11(등록), 12(조치 중)</td></tr>
<tr><td>인명피해</td><td colspan="8">0(사상자 없음), 1(사상자 발생), 2(다수 사상자)</td></tr>
<tr><td>화재여부</td><td colspan="8">30(화재 없음), 31(소화 중), 32(진화완료)</td></tr>
<tr><td colspan="2">비고</td><td colspan="8">사고 분류코드는 ASCII 코드를 사용</td></tr>
</table>

[표 77] 사고출동 지시 인터페이스 정의

• 사고 출동지시 확인

항 목		내 용	
명령어		사고 출동지시 확인	
서비스 구분		비상상황 대응 서비스	
명령어 코드값		0x54	
인터페이스 ID		VM502	MO502
송신자		차량 단말 서브시스템	미들웨어 서브시스템
수신자		미들웨어 서브시스템	이동체 안전관리 서브시스템
주요 정보		사고 출동지시 에 대한 수용 또는 거부 사항	
프로토콜		TCP/IP	
주기		발생 시	
총 데이터 크기		7 byte	
패킷 구조		업무 ID	수용/거부
데이터 크기		6 byte	1 byte
사고 출동 지시 확인	업무 ID	사고 출동지시 관련 업무 ID	
	수용/거부	출동지시에 대한 수용 또는 거부 사항 0(수용), 1(거부), 2(미확인)	
비고		없음	

[표 78] 사고 출동지시 확인 인터페이스 정의

- 사고 관련 지원요청

항 목		내 용						
명령어		사고 관련 지원요청						
서비스 구분		비상상황 대응 시스템						
명령어 코드값		0x55						
인터페이스 ID		VM503			MO503			
송신자		차량 단말 서브시스템			미들웨어 서브시스템			
수신자		미들웨어 서브시스템			이동체 안전관리 서브시스템			
주요 정보		사고 발생 시 사고 관련 지원 요청						
프로토콜		TCP/IP						
주기		발생 시						
총 데이터 크기		16 byte						
패킷 구조		업무 ID	차량단말 ID	소방차 추가	구급차 추가	사다리차 추가	물탱크차 추가	조명차 추가
데이터 크기		6 byte	5 byte	1 byte	1 byte	1 byte	1 byte	1 byte
사고 관련 지원 요청	업무 ID	사고 출동지시 관련 ID						
	차량 단말 ID	담당자 ID						
	소방차 추가	필요 여부 필요(0), 불필요(1)						
	구급차 추가	필요 여부 필요(0), 불필요(1)						
	사다리차 추가	필요 여부 필요(0), 불필요(1)						
	물탱크차 추가	필요 여부 필요(0), 불필요(1)						
	조명차 추가	필요 여부 필요(0), 불필요(1)						
비고		한 대 이상 출동지시가 가능하지만 지원요청은 대표차량 한 대에서만 가능함 지원요청은 한 대의 차량에서만 가능하되 정보를 잘못 전송했을 경우를 고려하여 재전송이 가능하도록 함						

[표 79] 사고 관련 지원요청 인터페이스 정의

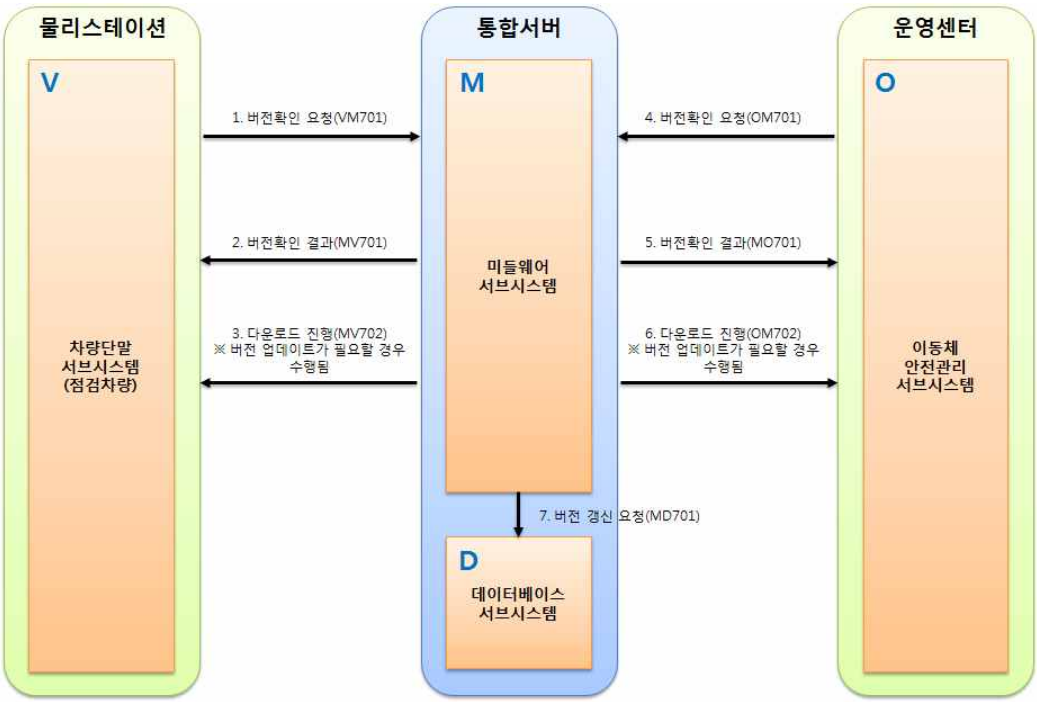
- 사고 조치완료

항 목		내 용					
명령어		사고 조치완료					
서비스 구분		비상상황 대응 서비스					
명령어 코드값		0x56					
인터페이스 ID		VM504		MO504			
송신자		차량 단말 서브시스템		미들웨어 서브시스템			
수신자		미들웨어 서브시스템		이동체 안전관리 서브시스템			
주요 정보		사고 조치완료 사항					
프로토콜		TCP/IP					
주기		발생 시					
총 데이터 크기		98 byte					
패킷 구조		업무 ID	차량단말 ID	사고 차량 유무	사고 차량 대수	사고차량 단말ID	특이사항
데이터 크기		6 byte	5 byte	1 byte	1 byte	5 byte	80 byte
사고 조치 완료	업무 ID	사고 출동지시 관련 업무 ID					
	차량단말 ID	담당자의 차량단말 ID					
	사고 차량유무	사고발생 차량의 존재 유무 있음(1), 없음(0)					
	사고 차량대수	사고발생 차량 대수					
	사고차량 단말ID	사고가 발생한 차량의 차량단말 ID					
	특이사항	텍스트 입력을 지원함(80byte 이내)					
비고		사고차량이 여러 대일 경우 대표 차량단말 ID만 전송					

[표 80] 사고 조치완료 인터페이스 정의

○ 소프트웨어 유지보수 서비스

- 소프트웨어 유지보수 서비스 인터페이스 구성도



[그림 20] 소프트웨어 유지보수 서비스 인터페이스 구성도

- 통합서버에서 차량단말기 운용소프트웨어를 관리하여 차량단말기의 운용소프트웨어의 업데이트 필요에 따라서 원격으로 소프트웨어 업데이트를 제공하여 차량단말기의 신속한 유지보수가 가능하도록 함
- 차량단말기 운용소프트웨어의 현재 버전과 통합서버의 관리버전을 비교하여 운용소프트웨어의 업데이트 정보를 제공
- 사용자에 의한 운용소프트웨어 업데이트 확인을 통한 소프트웨어 업데이트 제공

- 통신 인터페이스

• 소프트웨어 버전 확인 요청

항 목		내 용		
명령어		소프트웨어 버전 확인 요청		
서비스 구분		소프트웨어 유지보수 서비스		
명령어 코드값		0x71		
인터페이스 ID		VM701, OM701		
송신자		차량단말 서브시스템, 이동체 안전관리 서브시스템		
수신자		미들웨어 서브시스템		
주요 정보		차량단말 서브시스템의 응용프로그램에 대한 버전 정보		
프로토콜		TCP/IP		
주기		차량단말 서브시스템 실행 시		
총 데이터 크기		16 byte		
패킷 구조		장비 ID	장비종류	버전정보
데이터 크기		5 byte	1 byte	10 byte
버전 확인 요청	장비 ID	버전 확인을 요청하는 차량의 장비 ID		
	장비 종류	차량단말 서브시스템 : V, 이동체 안전관리 서브시스템 : O, 시설물 안전관리 서브시스템 : F, 지적정보 서브시스템 : P		
	버전정보	응용프로그램 버전 정보		
비고		장비 종류는 대소문자를 구별하므로 대문자로 작성되어야 함		

[표 81] 소프트웨어 버전 확인 요청 인터페이스 정의

• 소프트웨어 버전 확인 결과

항 목		내 용	
명령어		소프트웨어 버전 확인 결과	
서비스 구분		소프트웨어 유지보수 서비스	
명령어 코드값		0x72	
인터페이스 ID		MV701, MO701	
송신자		미들웨어 서브시스템	
수신자		차량단말 서브시스템, 이동체 안전관리 서브시스템	
주요 정보		차량단말 서브시스템의 응용프로그램에 대한 버전 확인 결과	
프로토콜		TCP/IP	
주기		버전 확인 요청 시	
총 데이터 크기		6 byte	
패킷 구조		차량단말 ID	버전확인 결과
데이터 크기		5 byte	1 byte
버전 확인 결과	차량단말 ID	버전 확인 요청 관련 차량단말 ID	
	버전확인 결과	버전확인 결과 70(완료), 71(다운로드 필요)	
비고		소프트웨어 버전 확인 결과 업데이트가 필요할 경우 실행파일의 다운로드를 진행함	

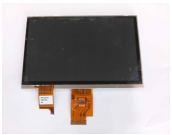
[표 82] 소프트웨어 버전 확인 결과 인터페이스 정의

나. 공항시설 안전통합 관리시스템 전용단말기 하드웨어 개발

(1) 설계구축

(가) 주요부품 선정 및 업체 선정

○ 전용단말기의 하드웨어 개발을 위해 최적의 주요부품들을 선정함

순번	이미지	종류	제품명	제조사	사양
1		LCD TOUCH MODULE	LCD: EJ070NA-01C TOUCH: EP0700ML	NLK	• 7“ WVGA LCD • 1024 X 600 • 멀티 터치 정전압 방식
2		MAIN CHIP MODULE	EXYNOSS 4412 MODULE	MICROVISION	• CPU : Coretex-A9 • Quad Core 1.2 GHz • RAM : LPDDR2 1 GByte • Mem : 4 GByte • TYPE : 분리형모듈
3		GPS	ASK6C	ASCENKOREA	• 감도-165dbm • 갱신주기-10Hz • 1PPS 시간동기화 • 위성 궤도예측 알고리즘을 통한 빠른 위치정보 수신 • 최대 12채널 다중신호간섭제거기 • AGPS 지원, 로거기능 내장
4		Wi-Fi	WYSBMVGXB	TAIYO YUDEN	• IEEE802.11 a/b/g/n 지원 • W-LAN+Bluetooth mod • Dualband W/O X’tal

[표 83] 주요부품 선정 목록

(나) BOM 파일 생성

○ 회로기판에 실장 되는 부품 확인을 위한 BOM 파일 생성 후 주요부품에 대한 체크 리스트를 작성하여 확인

- BOM 파일은 다음의 표와 같은 형식으로 작성되며 주로 인덕터(L), 저항(R), 캐패시터(C), 칩셋(U), 다이오드(D), 커넥터(CN), 테스트 포인트(TP), 스위치(SW) 등으로 분류하여 기재

- 실장 되는 주요 자재에 대한 실장 위치명(Reference), 품명(Part Number), 시정수 값, 자재 수량 등을 기재

순번	수량	위치명(Reference)	품명(Part Number)
1	1	BT1	MS621
2	82	CB1,CB2,CB3,CB4,C4,CB5,C5,CB6,C6, CB7,CB12,C12,C13,CB14,C14,CB15, CB16,CB17,CB18,CB19,C19,CB20,C20, CB2,C21,CB22,C22,CB23,C23,CB24,C24, CB25,C25,CB26,C26,CB27,CB28,C31, CB32,CB33,CB34,CB35,C35,CB36,CB37, CB38,CB39,CB40,CB41,CB42,CB43,CB44, CB45,CB50,CB51,CB52,CB57,CB59,CB61, CB62,CB64,CB65,C65,CB66,CB67,C67, CB68,CB69,C69,CB70,CB71,C71,CB73, CB74,CB75,C77,C78,C86,C87,C90,C98,C99	100nF/C1005
3	4	CB8,CB9,CB10,CB11	22pF/C1005/NC
4	1	CB13	1nF/C1005
5	5	CB29,CB30,CB31,C59,CB63	10nF/C1005
6	4	CB53,CB54,CB55,CB60	1uF/C1005
7	1	CB56	1uF/50V/C2012
8	2	C29,CB58	100pF/C1005
9	1	CB72	1000P/C1005
10	1	CFG1	TDA06H0SK1
11	6	CN1,CN2,CN3,CN4,CN5,CN6	AXK760127G
12	1	CN7	3050-24R-0.5MSF
13	1	CN8	HIF3F-20PA-2.54DS (Box, Male, Right Angle)
14	2	CN9,CN11	D-7103US-02-003/USB_A
15	1	CN10	HD4/2MM
16	1	CN12	AXA2R73061

순번	수량	위치명(Reference)	품명(Part Number)
17	1	CN13	FEMALE_DB9
18	5	CN14,CN15,CN27,CN28,CN29	HD6/2MM
19	2	CN16,CN18	MINI_USB_B
20	1	CN19	J1B121ZDD
21	2	CN20,CN21	12505WR-02/NL
22	1	CN22	HD20X2/2MM
23	1	CN23	DC_JACK_DC005
24	1	CN25	HD15X2/2MM
25	1	CN26	HD9X2/2MM
26	2	CON1,CON2	PJ-327A
27	30	CT1,CT2,CT5,CT6,CT7,CT8,CT9,CT10,CT11,CT12,CT13,CT14,CT17,CT18,CT22,CT23,CT24,CT25,CT29,CT30,CT32,CT33,CT34,CT35,CT36,CT37,CT38,CT40,C52,C54	10uF/10V/T1608
28	2	CT3,CT4	100uF/10V/B/2012
29	2	CT15,CT16	220uF/6.3V/B/3528
30	2	CT26,CT39	4.7uF/16V/T3216
31	1	CT27	10uF/10V/T2012
32	2	CT28,CT31	220uF/10V/7343
33	4	C1,C8,C9,C95	18PF/C1005
34	2	C2,C3	1nF/C1005/NC
35	1	C7	0.1uF/C1005
36	2	C10,C16	100nF/C1005/NC
37	2	C11,C17	1.0uF/C1005
38	2	C15,C18	4.7uF/C1608
39	2	C27,C28	13pF/C1005
40	3	C30,C41,C42	33pF/C1005
41	2	C32,C33	1uF/10V/C1005
42	1	C34	10uF/2012/10V
43	4	C36,C37,C38,C39	15pF/C1005/NC
44	1	C40	22nF/C1005
45	5	C43,C46,C50,C51,C53	0.22uF/C1005
46	4	C44,C48,C62,C63	220pF/C1005
47	4	C45,C47,C60,C61	4.7uF/6.3V/C1608
48	6	C49,C55,C56,C57,C94,C96	1uF/16V/C1608
49	1	C58	4.7uF/1608/6.3V

순번	수량	위치명(Reference)	품명(Part Number)
50	4	C64,C66,C68,C70	10uF/10V/C1608
51	6	C72,C73,C74,C100,C101,R256	NC/C1005
52	4	C75,C76,C85,C89	10uF/1608/10V
53	5	C79,C80,C81,C82,C83	30pF/NC/C1005
54	1	C84	22pF/C1005
55	1	C88	15pF/C1005
56	2	C91,C92	10uF/10V/1608
57	1	C93	1uF/10V/C1608
58	1	C97	100NF/16V/C1005
59	16	D1,D2,D3,D4,D5,D6,D7,D8,D9, D10,D11,D13,D15,D16,D17,D18	Rclamp0502B/SOT523
60	2	D12,D14	PGB1010603NR
61	2	D20,D22	DAN217
62	1	D21	UDZS-6.8B
63	1	D23	UDZS-20B
64	2	D24,D26	RSX101VA-30
65	1	D25	P6SMBJ6.0
66	29	FB1,FB2,FB3,FB5,FB7,FB8,FB10,FB11, FB12,FB13,FB14,FB15,FB16,FB17, FB19,FB21,FB24,FB25,FB26,FB27, FB28,FB29,FB30,FB31,FB32,FB37, FB38,FB39,FB41	HH-1M1608-600JT
67	7	FB4,FB6,FB9,FB18,FB22,FB23,FB42	HH-1M1608-600JT/NC
68	1	FB20	HH-1M1608-600J
69	5	FB33,FB34,FB35,FB36,FB40	HH-1M1005-100JT
70	1	F1	MINISMDC200F-2
71	1	J1	HDR2/2mm
72	1	J4	S510-121
73	2	J5,J6	HDR2X12_2MM
74	1	J7	HDR2X8_2MM
75	1	J8	HDR5_2MM
76	2	J9,J10	HDR2X6_2MM
77	1	J11	05004HR-40A01
78	1	J12	FH19C-8S-0.5SH
79	4	LB1,LB2,LB3,LB4	MPZ1608S221AT
80	2	LED1,LED2	LED_RED/2012
81	6	LED3,LED4,LED5,LED6,LED7,LED8	LED_GREEN/2012

순번	수량	위치명(Reference)	품명(Part Number)
82	2	L1,L12	VLCF4020T-2R2N1R7
83	4	L2,L3,L4,L5	CL-2M2012-900JT
84	2	L6,L7	YCDH32T-2R2
85	2	L9,L13	VLCF4020T-100MR85
86	1	L10	LQW18AN5N6C00D/L1608
87	1	L11	VLCF4020T-6R8N1R0
88	5	M1,M2,M3,M4,M5	HOLE-TH
89	5	Q1,Q2,Q3,Q7,Q8	MMBT3904WT1
90	2	Q4,Q5	SI2333DS/SOT-23
91	1	Q6	MMBT3904WT1/NC
92	1	Q9	SI3443DV
93	12	RR1,R139,R180,R196,R267,R268, R271,R272,R291,R296,R297,R326	0/R1005/NC
94	3	R1,R51,R126	100K/R1005/NC
95	3	R2,R3,R181	1K/R1005/NC
96	42	R4,R7,R9,R10,R14,R15,R16,R17,R18, R19,R20,R21,R22,R23,R24,R25,R28,R29,R63 ,R64,R65,R66,R67,R68,R69,R70,R71,R72,R7 3,R74,R75,R76,R113,R114,R131, R144,R159,R160,R161,R162,R166,R286	10/R1005
97	20	R5,R37,R38,R40,R42,R47,R48,R49, R55,R96,R97,R98,R108,R171,R215, R279,R308,R314,R317,R318	1K/R1005
98	6	R6,R11,R13,R26,R123,R124	R1005/NC
99	1	R8	232K/F/R1005
100	1	R12	51K/F/R1005
101	47	R27,R119,R120,R121,R122,R138,R140, R147,R148,R163,R164,R167,R168,R169,R1 70,R173,R174,R175,R176,R177,R178,R179, R182,R184,R190,R191,R192,R193,R194,R1 95,R213,R216,R222,R249,R251,R253,R265, R266,R269,R270,R293,R298,R300,R301,R3 03,R309,R330	0/R1005
102	28	R30,R39,R53,R77,R78,R86,R89,R100, R106,R107,R149,R151,R152,R165,R183,R2 19,R229,R245,R278,R280,R287,R288,R294, R306,R320,R321,R322,R324	10K/R1005
103	24	R31,R33,R34,R36,R43,R44,R50,R52, R57,R87,R88,R109,R115,R116,R150, R155,R172,R244,R247,R273,R274, R276,R292,R295	100K/R1005

순번	수량	위치명(Reference)	품명(Part Number)
104	6	R32,R35,R41,R45,R46,R135	200/R1005
105	1	R54	12K/R1005
106	2	R56,R141	1M/R1005
107	4	R58,R59,R132,R134	330/R1005
108	6	R60,R61,R104,R105,R111,R112	10/R1005/NC
109	2	R62,R81	0/1005
110	8	R79,R80,R99,R101,R281,R282,R285,R299	100/R1005
111	11	R82,R83,R84,R85,R90,R91,R92,R93, R94,R95,R277	49.9K/R1005
112	14	R102,R110,R117,R118,R136,R137,R143,R1 46,R187,R188,R189,R275,R328,R329	4.7K/R1005
113	1	R103	38.3K/F/R1005
114	1	R125	100K/F/R1005
115	4	R127,R128,R129,R130	49.9/F/R1005
116	1	R133	12.4K/F/R1005
117	1	R142	4.7K/R1005/NC
118	1	R145	1K/F/R1005
119	1	R153	2.2K/R1608
120	3	R154,R323,R325	2.2K/R1005
121	2	R156,R157	47K/R1005
122	7	R158,R238,R239,R240,R241,R242,R243	33/R1005
123	6	R185,R186,R257,R258,R260,R261	1.2K/R1005
124	1	R214	1K/R1608
125	1	R217	1.5K/R1608
126	2	R218,R225	110K/R1005
127	2	R220,R227	15K/R1005
128	1	R223	2K/R1005
129	1	R224	33K/R1005
130	11	R226,R228,R230,R302,R304,R307, R310,R311,R313,R315,R327	NC/R1005
131	1	R246	100K/NC/R1005
132	5	R248,R250,R252,R254,R255	47K/NC/R1005
133	4	R259,R262,R263,R264	150/R1005
134	1	R283	34.8K/F/R1005
135	1	R284	2.0K/F/R1005
136	1	R289	232K/F/R1005
137	1	R290	120K/F/R1005
138	1	R305	NC(0)/R1005
139	1	R312	NC/R1608

순번	수량	위치명(Reference)	품명(Part Number)
140	2	R316,R319	2R2/R1608
141	1	SW1	ST-1501
142	1	SW2	MM8130-2600RA2
143	2	SW3,SW4	ST-1102
144	6	SW22,SW23,SW24,SW25,SW26,SW27	LS8J2M-T
145	35	TP1,TP2,TP3,TP4,TP5,TP6,TP7,TP8, TP9,TP10,TP11,TP12,TP13,TP14,TP15, TP17,TP18,TP19,TP20,TP21,TP22,TP23, TP24,TP28,TP29,TP30,TP31,TP32,TP33, TP34,TP35,TP36,TP37,TP38,TP40	TP
146	3	TP25,TP26,TP27	T POINT R
147	2	U1,U26	NCP1529ASNT1G
148	1	U2	MIC2026-1_SO8
149	1	U3	USB4640-HZH-03
150	1	U4	SP3232EEY
151	1	U5	MAT1316
152	1	U6	TPS2553DRV
153	3	U7,U9,U15	SN74AVC4T245DGV
154	1	U8	CP2102
155	1	U10	SN74LVC1G08DCK
156	1	U11	LAN9220-ABZJ
157	1	U12	93LC46-I/SN
158	1	U13	WM1800GEFL
159	1	U14	SN75LVDS83BDGG
160	2	U16,U21	FXMA2102L8X
161	1	U18	MP1541DJ
162	1	U19	WYSBMVGXB
163	1	U20	XC6210F182MR
164	1	U22	MAX17040G
165	1	U23	ALPU-MR-6L/SOT23-6
166	1	U24	TPS61032PWP
167	1	U25	BQ24032ARHLR
168	1	U27	MAX16054AZT
169	1	U28	MP3302DJ
170	1	U114	53047-0210RB
171	1	U115	RGP13T3
172	1	X1	25MHz/SX32
173	1	Y1	24.000MHz/3225
174	1	Y2	7XZ-32.768KBA-T
175	1	Y3	DSB321SCL(38.4MHz 3225)

[표 84] 부품별 위치명(Reference) 및 품명(Part Number)

(2) 시제품 디자인 확보

(가) 시제품 디자인

○ LCD 및 메인보드, 배터리 사이즈에 맞게 디자인 설계 진행

- 사이즈: 170(W) x 100(M) x 15(H) mm

순번	분류	이미지	설 명
1	전면부		• LCD를 통하여 현재 공항상황에 대한 모니터링 가능
2	측면부		• 좌측 : 음량 크기 조절용 방향키 • 우측 : 업데이트 및 소프트웨어 디버깅 작업을 위한 USB 포트 및 SD 카드 슬롯
3	후면부		• 상단 좌/우측 : 스피커 설치 • 중앙 : 내비게이션 거치대 장착 가능

[표 85] 1차년도 기구 디자인 형상

(나) 디자인 목업을 위한 도면 제작

○ LCD, 메인보드, 배터리, 커넥터의 사이즈에 맞추어 기구 도면 설계

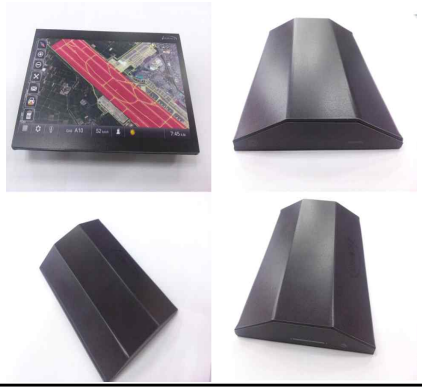
○ 측면은 다음과 같이 다각형 모양으로 설계

○ 후면은 거치대 장착 홈을 기준으로, 상단과 하단은 측면에 맞추어 설계



[그림 21] 전용단말기의 목업 제작용 도면

(다) 아크릴 목업 제작

분류	이미지	설명
이미지 부착전		도면에 따라 실제 사이즈를 적용하여 아크릴 모형 제작 진행
이미지 부착후		모형 제작완료 후 디자인 이미지를 시트지에 프린트하고 아크릴 모형에 부착하여 제작을 완료

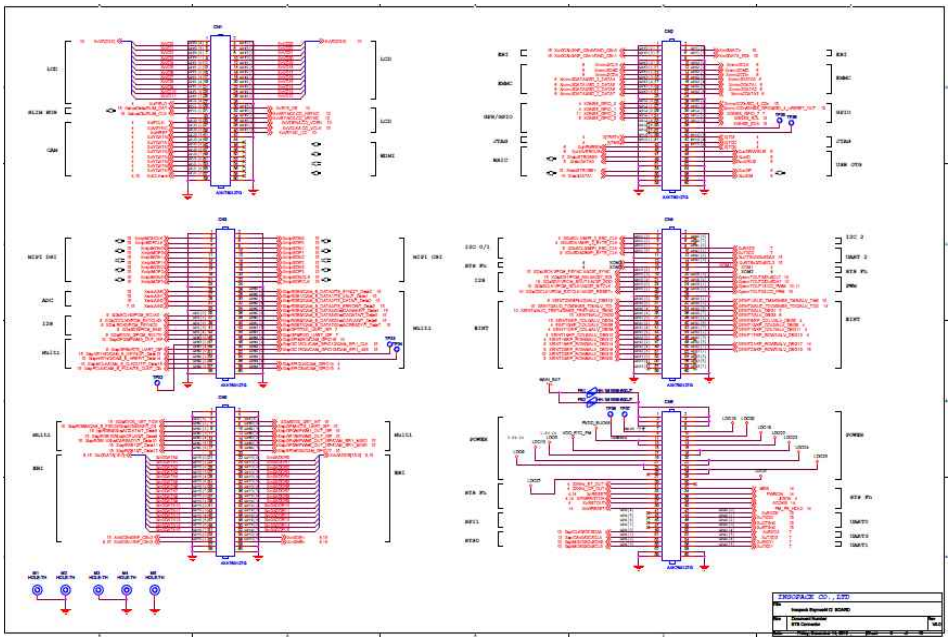
[표 86] 전용단말기의 실제 사이즈 도면 제작

(2) 개발보드 설계 및 구현

(가) 회로도 설계

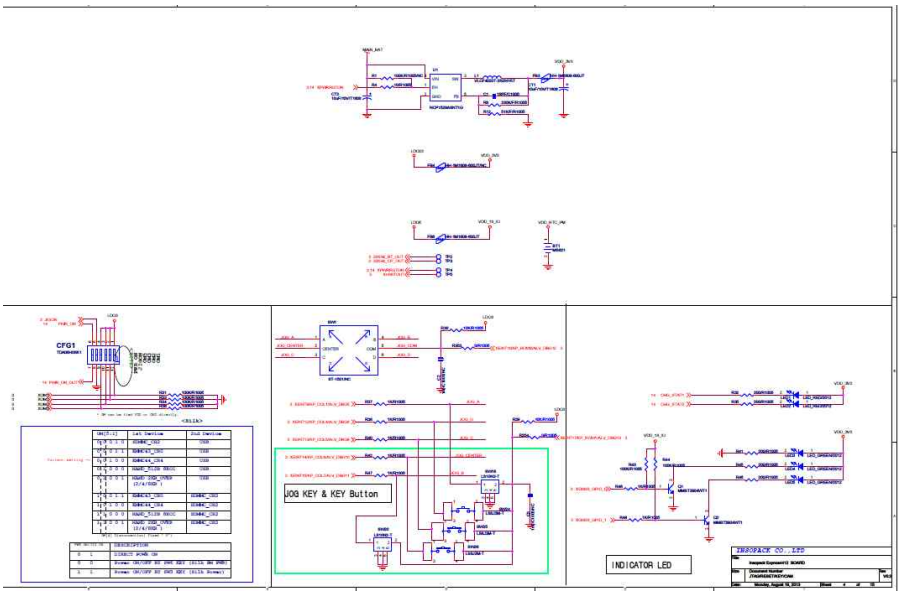
○ 이론적 • 실험적 접근방법을 적용한 연구 내용 및 결과에 따른 회로 설계 진행

- EXYNOS4412 모듈과 메인보드 연결 파트의 회로도

도면	
설명	AXK760127G 커넥터 6개를 사용하여 EXYNOS4412 모듈과 연결될 전원 및 데이터 신호 라인들을 AXK760127G에 맵핑

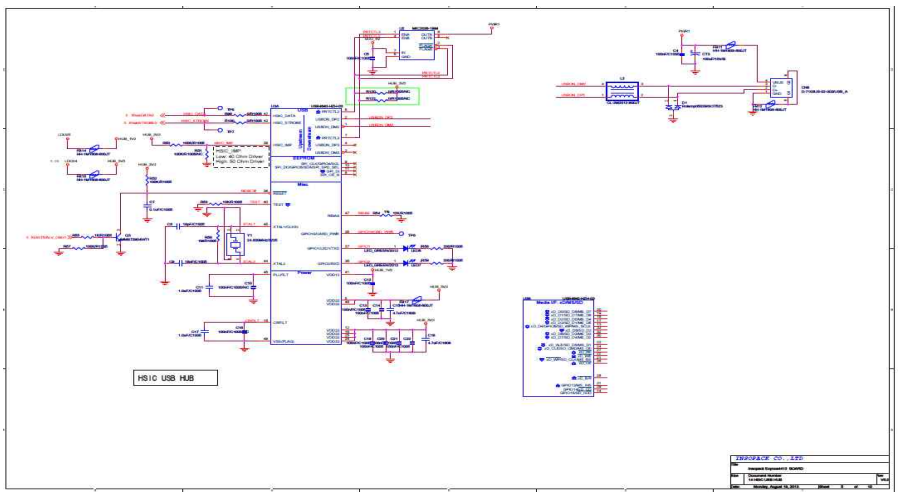
[표 87] EXYNOS4412 모듈과 메인보드 연결 파트의 회로도

- 전원부(POWER), CAM 모듈부, KEY, LED 파트의 회로도

도면	
설명	• 전원부(POWER) : 메인 배터리와 RTC BAT를 사용할 수 있도록 설계 • CAM 모듈부: 커넥터를 이용하여 CAM 모듈을 사용할 수 있도록 설계 • JOG KEY : 상, 하, 좌, 우로 이동 가능하도록 설계 • FUNCTION KEY : 상, 하, Home, Back 기능을 사용할 수 있도록 설계 • LED : 메인보드 작동 여부 확인이 가능하도록 설계

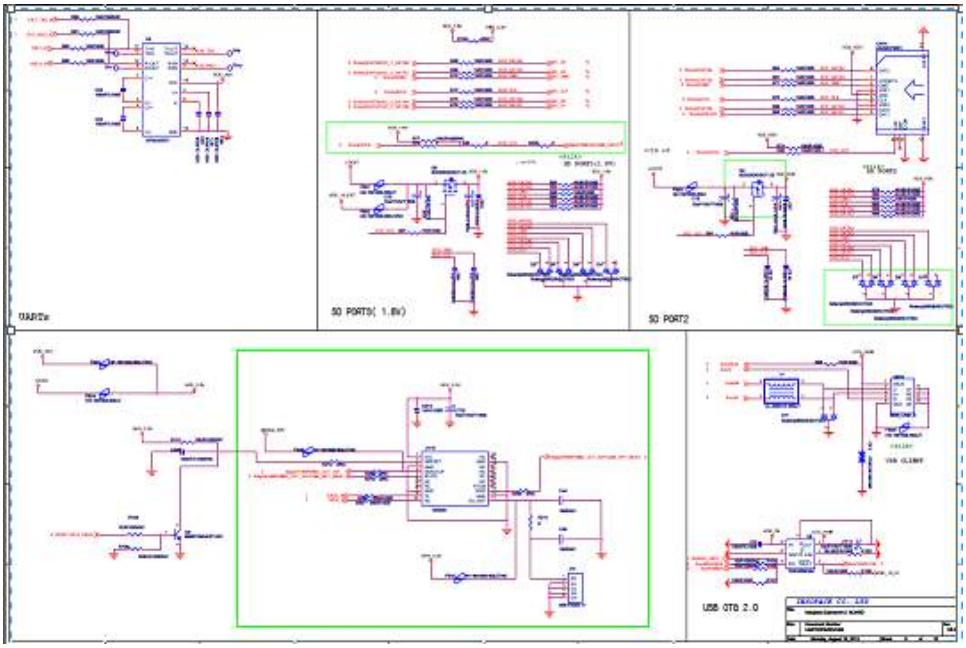
[표 88] 전원부(POWER), CAM 모듈부, KEY, LED 파트의 회로도

- USB Host 파트의 회로도

도면	
설명	USB Host 기능을 이용하여 마우스, 키보드, USB 등 다양한 USB 인터페이스 매체들의 사용이 가능하도록 설계

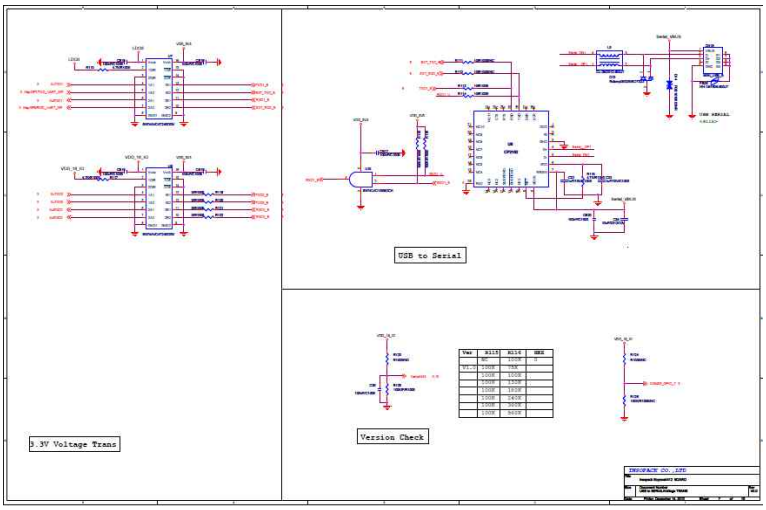
[표 89] USB Host 파트의 회로도

- UART, SD PORT, Wi-Fi, GPS, MINIUSB 파트의 회로도

도면	
설명	• UART : UART 인터페이스를 이용하여 디버깅 및 데이터 전송이 가능하도록 설계 • SD 포트 : 외부 스토리지 및 SD 타입의 모듈 구동 및 부팅 모드 시 사용 가능 하도록 설계 • Wi-Fi : 공항 내 무선 AP와 연동하여 네트워크를 사용할 수 있도록 설계 • GPS : 위성을 통하여 현재 자신의 위치 데이터 수신이 가능하며 GPS 데이터를 이용하여 다양한 응용프로그램 개발이 가능하도록 설계 • MINIUSB : 개발과정에서 부트 로더, 커널, 안드로이드 운영체제 바이너리 다운로드 및 응용프로그램 설치가 가능하도록 설계

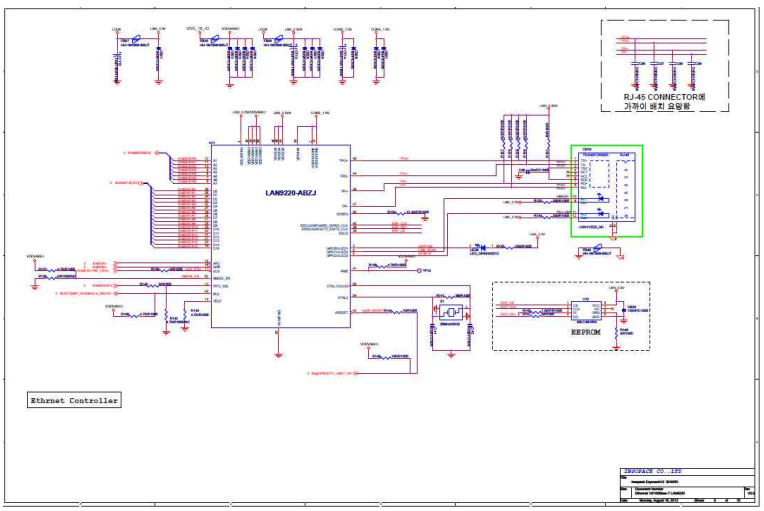
[표 90] UART, SD PORT, Wi-Fi, GPS, MINIUSB 파트의 회로도

- 3.3V Voltage Trans, MINIUSB 파트의 회로도

도면	
설명	• Voltage Trans : 5 V를 3.3 V로 변환하기 위하여 설계 • MINIUSB : USB to Serial Port를 통해 디버깅이 가능하므로 운영체제의 구동상태 확인에 사용이 가능하도록 설계

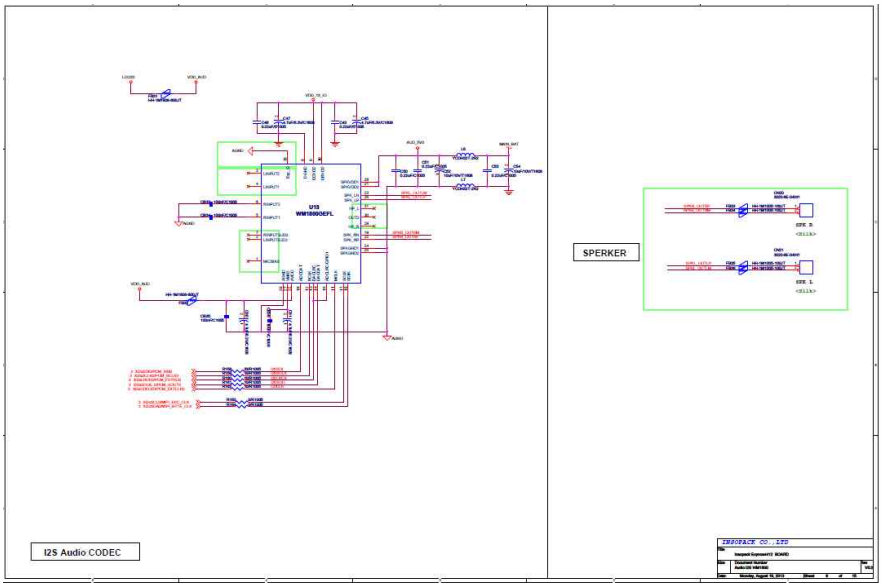
[표 91] 3.3V Voltage Trans, MINIUSB 파트의 회로도

- LAN 파트의 회로도

도면	
설명	• UTP : 유선 랜 네트워크 연결 및 부트 로더, 커널, 안드로이드 운영체제 바이너리 다운로드에 사용하며 추후 프로그램 업데이트 및 외부 장비 연동에 사용하기 위해 설계 • 현재는 디버깅 용도로 NC(Not Connect) 처리하여 설계

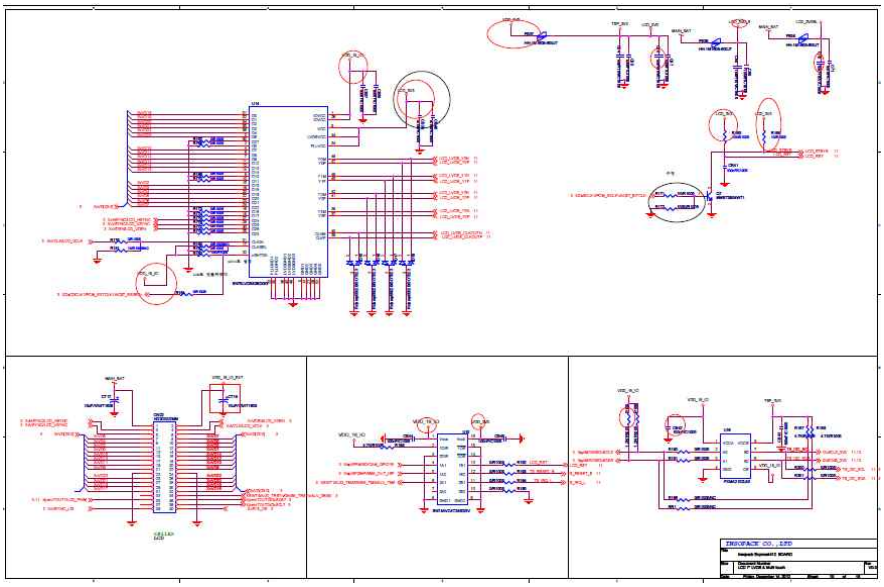
[표 92] LAN 파트의 회로도

- AUDIO 파트의 회로도

도면	
설명	스피커를 사용하여 사용자에게 미디어 재생 기능을 제공하도록 설계

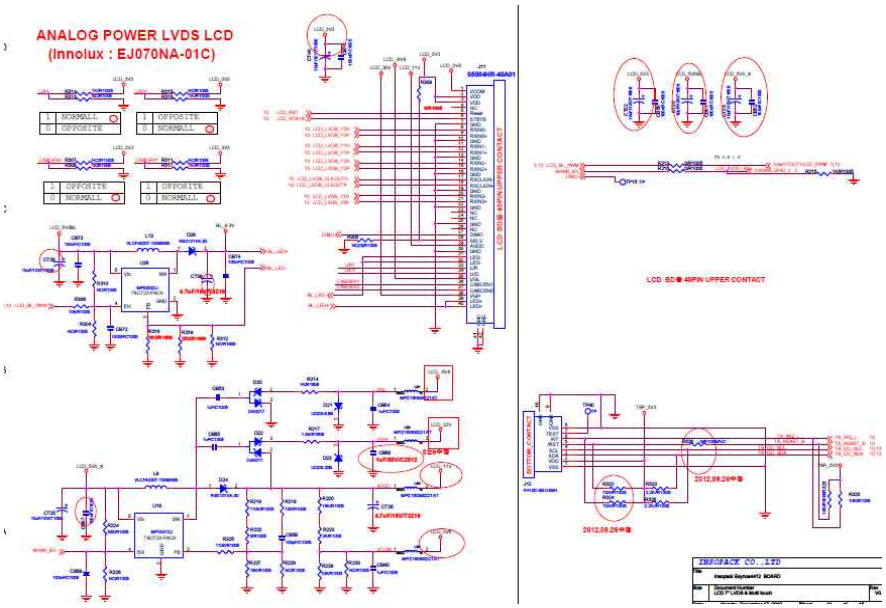
[표 93] AUDIO 파트의 회로도

- LCD 및 Touch 연결 파트의 회로도

도면	
설명	LCD & Touch : 다양한 정보의 디스플레이 및 응용프로그램을 구동하기 위해 터치스크린과 LCD를 메인보드에 연결하도록 커넥터 설계

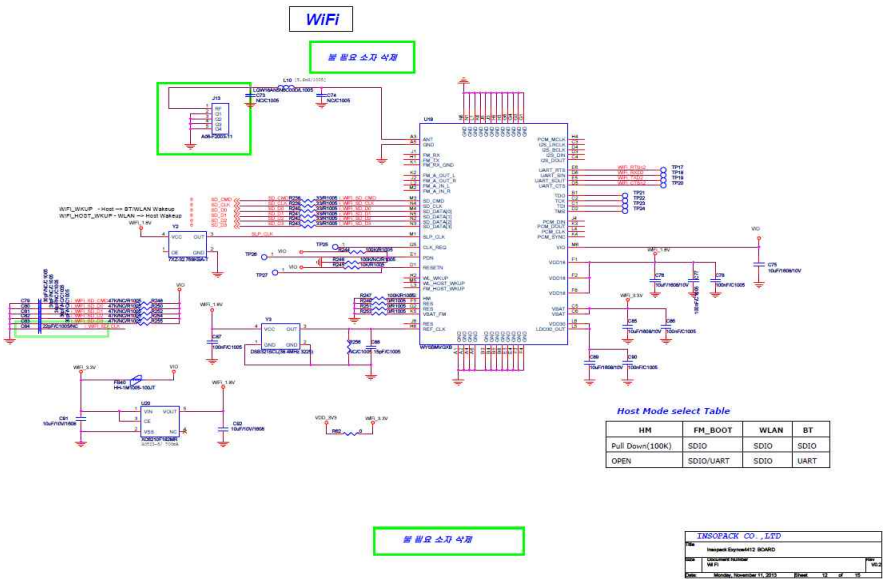
[표 94] LCD 및 Touch 연결 파트의 회로도

- LCD 파트의 회로도

도면	
설명	LCD : 다양한 정보의 디스플레이 및 LCD 구동을 위하여 설계

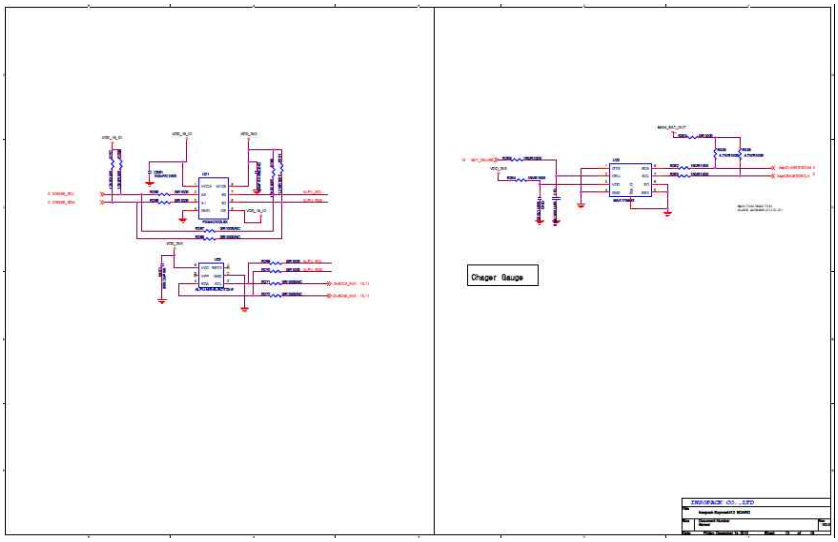
[표 95] LCD 파트의 회로도

- Wi-Fi 파트의 회로도

도면	
설명	Wi-Fi : 공항 내 무선 AP와 연동하여 네트워크 사용가능 하도록 설계

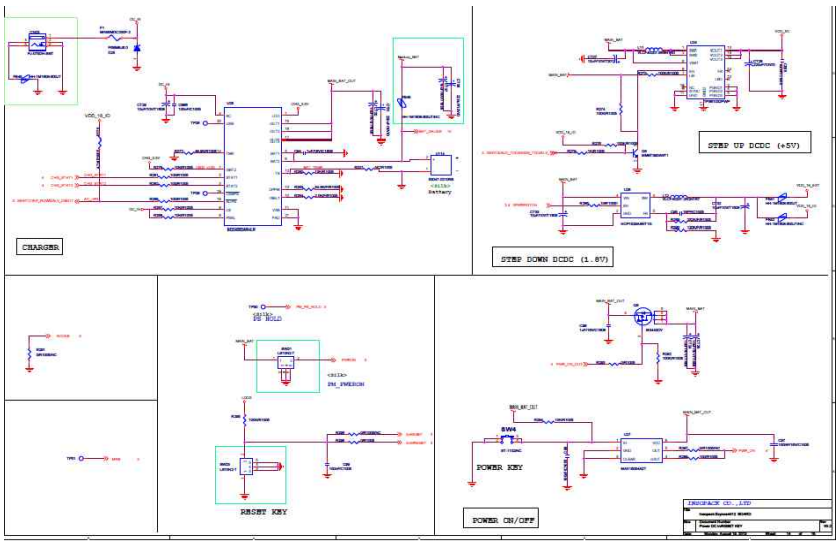
[표 96] Wi-Fi 파트의 회로도

- 센서 파트의 회로도

도면	
설명	Charger Gauge : 메인 배터리 및 센서와 연결되어 충전 진행사항을 확인할 수 있도록 설계

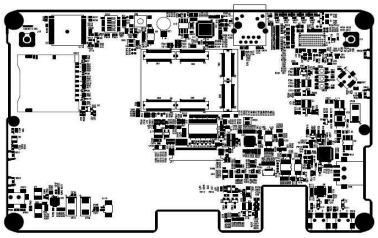
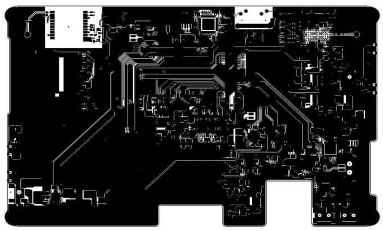
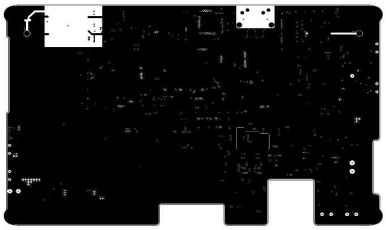
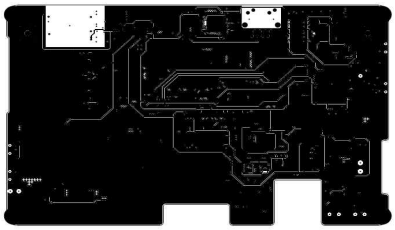
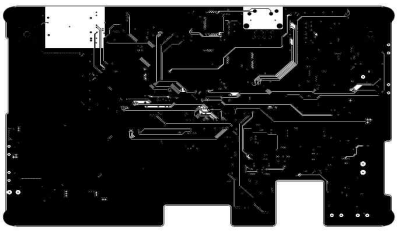
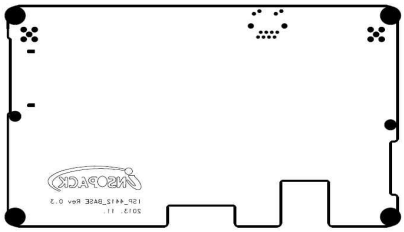
[표 97] 센서 파트의 회로도

- CHARGER, DC TRANSFORMER, POWER KEY 파트의 회로도

도면	
설명	• Charger : 전원 인가 시에 충전기를 통하여 충전이 가능하도록 설계 • DC TRANSFORMER : 5V로 전압 상승 및 1.8V 전압 하강을 위하여 설계 • RESET KEY : 전원 재부팅을 위하여 설계 • POWER ON/OFF KEY : 전원 ON/OFF 설정을 위하여 설계

[표 98] CHARGER, DC TRANSFORMER, POWER KEY 파트의 회로도

(나) 아트워크(Artwork)

층	이미지	내 용
1		보드 정면부에 자재 실장
2		그라운드(GND)층으로 많은 VIA 작업이 되어있음
3		주요 신호라인으로 다른 신호와의 간섭을 최소화하여 라인배치
4		주요 전원라인이 배치되어 있으며 다른 신호라인보다 두께가 두꺼움
5		이외 나머지 신호라인들을 배치
6		보드 후면부의 자재 실장

[표 99] 층별 신호 라인 배치도

(다) 레이아웃(Layout)

○ 회로도 및 BOM 파일을 기반으로 주요부품 배치

○ 커넥터 종류는 향후 기구물이 안착된 상태를 고려하여 외곽에 배치

○ Wi-Fi와 GPS는 모듈간의 간섭이 최소화되는 위치에 배치

○ 이외 실장 되는 부품들도 간섭 등을 고려하여 배치

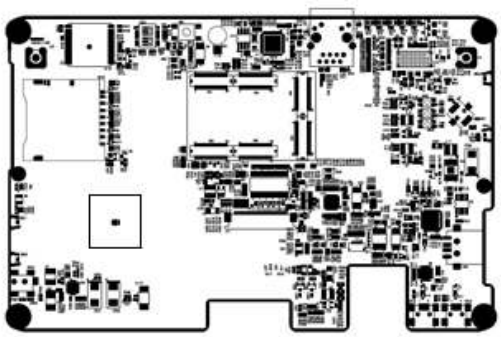
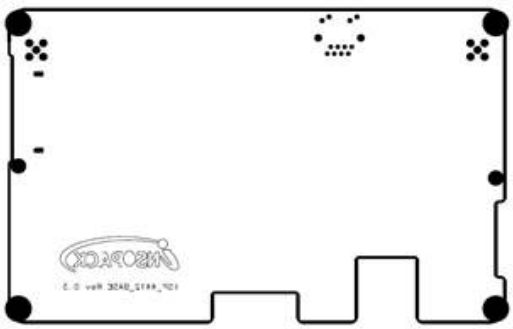
○ 레이아웃 설계에서도 기구 실장을 고려하여 최소의 크기로 설계

- 기구 최소 크기 구현을 위한 불필요 항목 삭제(Ethernet, Jog Key 등)

- 실장 및 부품 간 이격 공간 최소화

- GPS 내장형 안테나 위한 내부 회로 구성

- 양산 및 생산에 영향이 없는 USB Debug 회로 삭제

전면부 모듈 구성	후면부 모듈 구성
	

[표 100] 레이아웃 모듈 구성

(라) 보드 제작

○ BOM 파일, 회로도 및 레이아웃 설계 기반으로 실제 보드 제작완료

○ 3차년도 하드웨어 제작 보드는 불필요 소자 삭제, 부품간의 이격 거리 최소화, 고성능 GPS 내장안테나 설치, Wi-Fi, GPS 무선 성능 개선을 위한 임피던스 최적화 진행

구분	보드 이미지	구성 모듈 및 설명
전면부		• SD 카드 : 외부 저장소로 SD 타입의 모듈 구동 및 부팅모드 시 사용 가능(전 버전 대비 포트 변경) • MINIUSB : 개발과정에서 안드로이드 운영체제의 바이너리 다운로드 등을 위해 사용(최소 수량 만 남김) • UART : 디버깅 및 데이터 전송 기능으로 사용 가능 • USB : USB Host 기능을 이용하여 마우스, 키보드, USB 등의 다양한 USB 인터페이스 매체 사용 가능 • GPS : 위성을 통해 현재 자신의 위치정보 수신 가능(외장형태에서 내장형태로 변경을 위한 회로 수정) • MCU : 여러 칩들을 제어하는 역할을 하고 있으며 현재는 모듈 타입의 MCU 사용 • AUDIO : 사용자에게 음성 정보 제공 • Wi-Fi : 공항 내 무선 AP와 연동하여 네트워크 사용 가능 • LCD : 다양한 정보를 디스플레이하기 위해 사용 • TOUCH : 터치를 이용하여 사용자의 기능 제어 및 정보 입력 가능 • DC_IN : 5V 직류(DC) 전원 공급 • 스피커(SPK) : 단말기에서 출력되는 소리를 들을 수 있음 • FUNCTION 키 : 기능키를 통해 UP, DOWN, BACK, MENU, HOME 기능 사용가능 • IDLE 키 : 전원부의 예비키 • RESET 키 : 전원의 재부팅을 위한 키 • 전원(PW) 키 : 주 전원의 ON/OFF 동작을 위한 키
후면부		• 기구 디자인과 최소 높이 구현을 위하여 후면부에 부품 실장을 지양 • 성능에 영향이 없도록 부품 간의 특성을 고려한 설계 • LCD 부품 간섭이 없도록 설계 • LCD 동작 시 노이즈 간섭이 없도록 설계

[표 101] 전용단말기의 보드 구성 모듈 및 설명

(3) 시제 제작 및 검증

(가) 입출력부 검증

○ LCD를 통하여 정상적으로 안드로이드 화면의 위젯과 메뉴의 정상동작 확인



[그림 22] LCD를 통한 안드로이드 메인화면 출력 확인

○ 안드로이드 동작 시 발생하는 모든 소리는 메인보드 뒷면의 스피커를 통해 출력

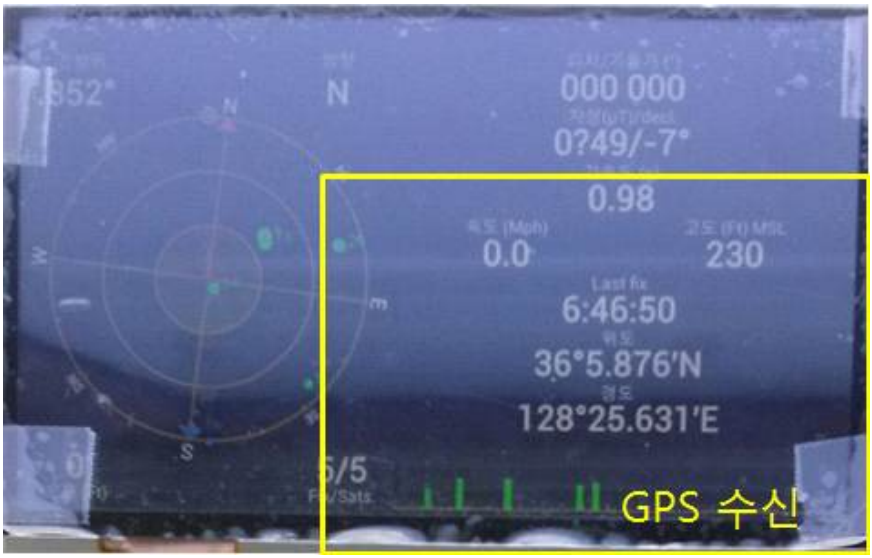
(나) 주제어부 검증

항목		설명
CPU	EXNOS4412	ARM Cortex A9 Core base CPU x 4 32/32/32/32KB I/D Cache , 1MB L2 Cache Internal ROM 64KB, Internal RAM 256KB 1.4Ghz Operating Frequency 0.4mm ball pitch, 804-FCMSP, 12mm x 12mm
LPDDR2 메모리	LPDDR2	1GB LPDDR2
EMMC 메모리	SDIN5D2-4G	4GB eMMC Ver4.4.1 적용, 용량 확장 가능 (Optional) 0.5mm Ball Pitch, 168 ball TFBGA, 12mm x 16mm
PMIC	S5M8767A	Step-Down Convertor 3Port, 3-LDO 제공 Out Current : DC-DC1-1.6A,DC-DC2 -1A, DC-DC3-0.8A input Voltage Range : 2.5 ~ 6V 32-Pin 5.0mm x 5.0mm QFN Package
커넥터	AXK860125WG	360pin (0.4mm pitch 60pin x 6)
사이즈	(WxDxH)	46 mm x 33 mm x 1mm (EPO유효기간:7/ 14 /30일)

[표 102] 주제어부의 모듈 설명

(다) 무선부 검증

○ GPS 테스트용 응용프로그램을 통한 GPS 정상동작 유무 확인

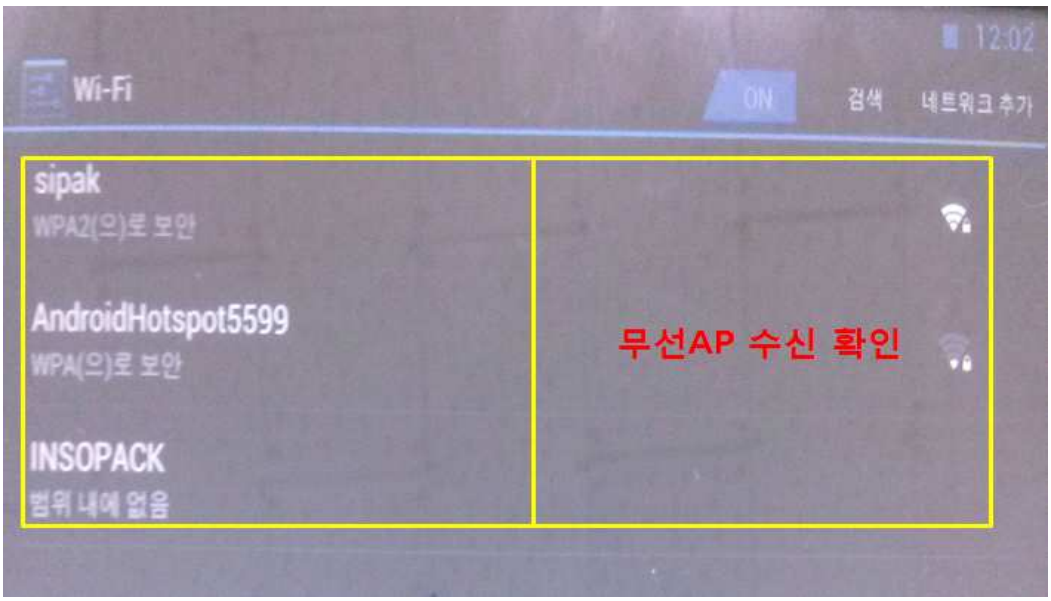


[그림 23] GPS 데이터 수신확인 화면

항목	설명
사이즈	15(W) x 15(D) x 4(H) mm
감도	-165 dBm, 개활지 수신시 45 dBm
갱신주기	최대 10Hz
채널	최대 12채널 다중신호 간섭 제거기
동기화 지원	고정밀 1pps 시간 동기화 지원
AGSP 사용	AGPS 지원을 통한 빠른 위성 연결 (EPO 유효기간:7/ 14 /30일)
로거 기능 내장	GPS 엔진의 내부 플래시 메모리를 사용하여 GPS 데이터 저장 가능
소모전류	초기 소모 전류 : 25mA, 동작 시 소모 전류 : 20mA

[표 103] 무선부의 GPS 모듈 설명

○ Wi-Fi 전원 설정 후 무선 AP 수신 확인



[그림 24] Wi-Fi 포팅 후 무선 AP 수신 확인

항목	설명
사이즈	10(w)X4(d)X1(h)mm
IEEE802.11	IEEE802.11 a/b/g/n
블루투스 지원	Bluetooth 2.1+EDR
인터페이스	SDIO, UART, PCM
보완	WEP(64/128), AES, TKIP, WPA/WPA2, WAPI
사용 전압 범위	3.2V~4.3V
수신 감도	-72dBm
주파수 범위	2.4Ghz, 5Ghz

[표 104] 무선부의 Wi-Fi 모듈 설명

(4) 전용단말기 기구 디자인 및 금형제작

(가) 디자인 개발을 위한 항목 별 요구 사항 도출

○ 2차년도 연구개발 진행 중인 전용단말기의 기구 디자인 개발을 위하여 다양한 요구사항을 도출하고 해결 방안 검토

항목	요구사항
GPS 외장형 안테나	• 외장형 안테나 연결이 가능한 홀 적용 • Wi-Fi 안테나 위치와 동일하도록 적용
전면부 디자인	• LCD 모듈 사이즈(165.75x105.39x3.4)를 고려하여 디자인 설계 필요 • 기존 사각형 디자인에서 곡선 처리된 디자인 적용 필요 • 전면부 좌측에 VOL UP/DOWN, HOME, CANCEL 키 위치 • 전면부 우측에 POWER, RESET 키 위치 • 상단 좌/우측에 스피커 위치 및 방사홀 필요
후면부 디자인	• 단말기 모델명 및 업체 명 표시 적용 • 발열 방지를 위한 대안 적용 • 차량용 거치대 장착이 가능하도록 적용
측면부 디자인	• 좌측면 USB 커넥터가 위치하며 덮개 필요 • 좌측면 Wi-Fi 안테나 커넥터 위치하며 덮개 필요 • 좌측면 이어잭 및 마이크잭 위치 • 우측면 하단 Mini USB, 24핀 커넥터 위치하며 덮개 필요 • LCD 모듈, PCB, BATTERY(35.5x63x7.1)사이즈를 고려하여 설계 필요
하단부 디자인	• 하단면 우측부 SDHC 카드 슬롯이 위치하며 덮개 필요 • 하단면 우측부 Ethernet 커넥터 위치
규격 표현	실제 예상 크기 구현
로고 디자인	배젤 적용

[표 105] 디자인 제작 시 요청사항

(나) 다양한 형태의 상용 유사 단말기 디자인 조사

○ 디자인 작업을 진행하기 위하여 7인치 LCD를 적용한 상용 내비게이션, 태블릿 PC 등 벤치 마케팅 형태의 조사 진행

형태	진행내용
테블릿 PC	
내비게이션	

[표 106] 상용 스마트 단말기 디자인 조사

(다) 디자인 보완 및 설계 반영

○ 디자인 개발 진행 중 지속적인 검토와 업무 협력으로 개선된 결과 산출

구분	진행사항
높이	기구 디자인을 위한 PBA 부품 별 실장 높이 조사
사양	• Connector, LCD, Plug, Module 최적 부품 선정 • 기구 재질은 PC(폴리카보네이트) 형태로 결정
주 보드 검토	• 배치 및 연결 커넥터 위치 수정 적용 • 디자인을 위한 버전 제작 검토 • LCD 실장을 위한 Bottom 면 실장 부품 제외 검토 • GPS, Wi-Fi 외장형 안테나 커넥터 교체 검토 • DC 잭 소형 교체 검토 • 배터리 및 스피커 실장 위치 변경 검토
부가 기능	• 거치대 관련 3D 데이터 적용 • 배터리 적용 시뮬레이션 진행 • 방열 대비 통풍 형태 적용 • 방수 및 방진 관련 적용 검토 • 낙하, 충격 등 기본 사양 적용 검토

[표 107] 디자인 개발 보완 적용 사항

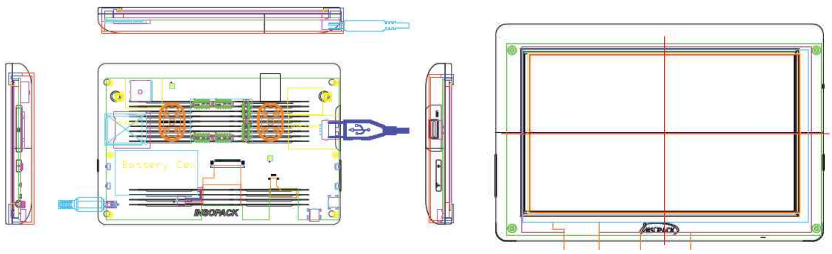
○ 최종 디자인 선정을 위하여 여러 종류의 디자인 검토

구분	내용
설계 디자인	

[표 108] 전용단말기의 설계 디자인 검토

(라) 디자인 보완 및 설계 반영

○ 시안 별로 여러 단계를 거쳐 디자인 수정 및 보완 진행

단계	진행 디자인
초기	
중간	
최종	

[표 109] 전용단말기의 단계별 디자인

단계	진행내용
초기	• 배젤의 크기와 위치 조정의 필요성 확인 • 전면부의 LED 필요성 확인 • 측면에 커넥터 종류 식별 • 측면 커넥터 커버 필요성 확인
중간	• 배젤 크기와 LCD 위치 결정 • 외부 연결 커넥터 시뮬레이션 진행 • 스피커 위치 및 상/하 면 크기 결정
최종	• 기구 크기 확정 • 기구 색상 확정 • 후면 디자인 선정 • QDM 진행을 위한 디자인 검토

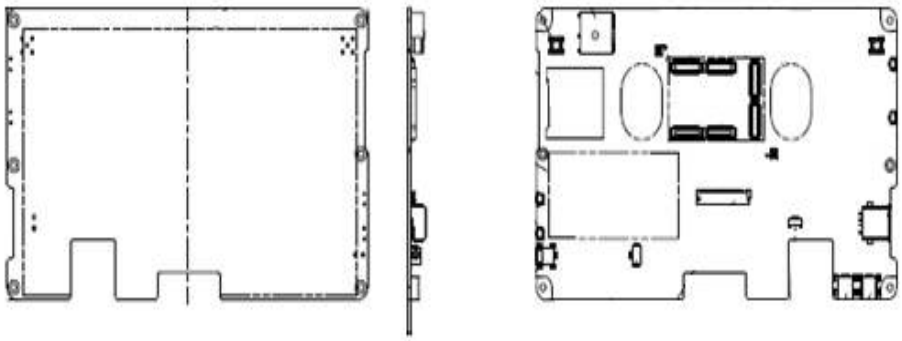
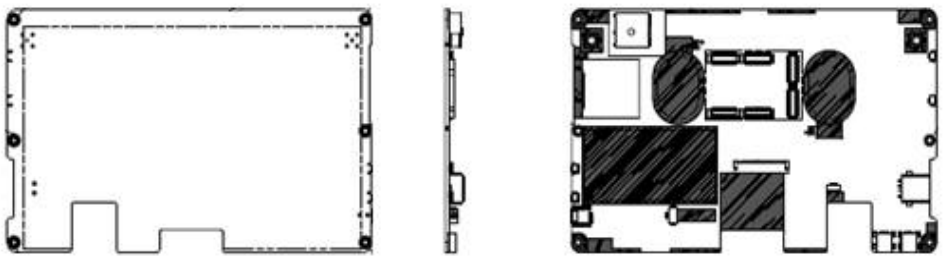
[표 110] 전용단말기의 단계별 진행 내용

(마) 전용단말기 기구 제작

① 기구 설계 진행

○ 기구 내부에 조립되는 보드의 크기와 연결되는 부품을 고려하여 기구 간의 유격과 조립 나사의 배치 검토

○ 기구 설계를 위하여 PCB 전/후면을 기준으로 구성 모듈에 따른 고정 및 간섭 관계를 고려하여 설계

구성	진행내용
전면부	
	• 메인 CPU 모듈지지 홀 설계 • LCD FPCB 지지 위치 설계 • Side Key 기능 별 위치 설계
후면부	
	• LCD 안착 및 고정지지 홀 설계 • 나사 홀 위치 설계

[표 111] 기구 설계용 배치 검토 및 진행내용

② 기구 구성 부품 리스트 생성

○ 기구를 구성하는 부품 확인을 위한 부품 리스트 파일 생성 후 체크 리스트를 작성하여 확인

- 기구를 구성하는 주요 자재에 대한 실장 구성 기구류(Item), 품명(Part Name), 구성 재질(Material), 표면처리방안(Surface Finish) 등을 기재

- Front Cover, Rear Cover, Key Volume, Key Power, SD Card Cover는 각각 1 Set 씩 구성

- Antenna Cover는 GPS, Wi-Fi 2 Set로 구성

- 주 재질은 폴리카보네이트(이하 PC)로 진행하고, Antenna Cover의 경우 Silicone 타입으로 적용하여 사용자의 삽입 또는 제거를 용이하도록 고려함

- 표면 처리는 Spray 방식 후 UV 코팅으로 마무리하여 스크래치 등이 쉽게 나타나지 않도록 설계

- QDM 진행이지만 Screw의 고정과 풀림이 용이하도록 4개의 Inset를 적용하며 쉽게 풀리지 않도록 0.2mm로 설계

번호	구성기구류(Item)	품명 (Part Name)	구성 재질(Material)	표면처리 방안 (Surface Finish)
1	Injection Mold	Front Cover	PC	Spray + UV
2	Injection Mold	Rear Cover	PC	Spray + UV
3	Double Injection Mold	Key Volume	PC + Urethane Rubber(hardeness 90%)	Spray + UV
4	Double Injection Mold	Key Power	PC + Urethane Rubber(hardeness 90%)	Spray + UV
5	Double Injection Mold	SD Card Cover	PC + Urethane Rubber(hardeness 90%)	Spray + UV
6	Rubber	Antenna Cover	Silicone Rubber(hardeness 50%)	Spray + UV

[표 112] 기구 구성 부품 리스트 및 특성

③ 기구 랜더링 진행

○ 기구 사출 전 크기 및 최정 형상에 대한 최종 시뮬레이션 결과물 산출

구분	내용
전체 형상	
측면 형상	
후면 형상	

[표 113] 전용단말기 최종 랜더링

④ 기구 사출 진행

○ 전용단말기의 기구 형상은 다음과 같음

전면부	후면부
	

[표 114] 기구 사출

(5) 하드웨어 수정보완 및 성능 안정화

(가) 제작 보드 검증

○ 설계 및 제작된 보드에 대하여 전원부 동작 확인을 위한 전원 검사 진행

순번	측정 포인트	위치	소스 출력신호	정상 시 전압	#1	#2	#3
1	FB3	TOP	VDD_3V3	3.3V	3.40	3.38	3.40
2	FB8	BOT	VDD_18_IO	1.8V	1.81	1.79	1.80
3	FB41	TOP	VDD_18_IO_EXT	1.8V	1.77	1.78	1.75
4	FB14	TOP	HUB_1V2	1.2V	1.21	1.19	1.20
5	FB15	TOP	HUB_3.3V	3.3V	3.03	3.05	3.01
6	FB24	TOP	GPS_3.3V	3.3V	3.32	3.28	3.33
7	C31	TOP	VDD_5V	5V	4.98	5.03	4.99
8	FB27	TOP	LAN_3.3V	3.3V	3.00	3.01	3.02
9	CT14	TOP	CORE_1.8V	1.8V	1.85	1.84	1.81
10	C51	BOT	AUD_5V0	5.0V	4.42	4.49	4.30
11	CB39	BOT	LCD_3V3	3.3V	3.45	3.40	3.36
12	FB39	TOP	LCD_5V0	5.0V	4.30	4.33	4.38
13	LB1	TOP	LCD_-6V8	-6.8V	-6.75	-6.78	-6.69
14	LB2	TOP	LCD_20V	20V	19.62	19.79	19.68
15	LB3	TOP	LCD_11V	11V	10.30	10.30	10.50
16	LB4	TOP	LCD_3V8	3.8V	3.80	3.84	3.74
17	R323	TOP	TSP_3V3	3.3V	3.4	3.39	3.36
18	CB74	TOP	BL_9.3V	9.3V	8.30	8.30	8.20
19	C87	TOP	WiFi_1.8V	1.8V	1.69	1.73	1.77
20	CT29	TOP	DC_IN	5.0V	5.09	5.19	5.23
21	CT27	TOP	MAIN_BAT	5V	4.42	4.39	4.42
22	R294	BOT	MAIN_BAT_OUT	5V	4.39	4.40	4.42
23	R278	BOT	CHG_3V3	3.3V	3.24	3.27	3.20

[표 115] 전원검사 진행 결과 예시

(나) Wi-Fi 안테나 성능 개선

○ 2차년도 연구개발 진행 시 Wi-Fi 안테나 성능 및 사용성 개선을 위하여 샤크, 옴니, 패치, 다이폴 형태의 안테나 성능시험 및 분석 진행

※ **여러 형태의 안테나 성능시험 내용은 "2절. 테스트베드 구축 및 운용시험, 3. 현장시험 및 실제 운용시험 진행, 가. 전용단말기 현장시험(1~4차), (6) 시험결과, (나) Wi-Fi 안테나 거리 및 음영지역 시험(1~3차)" 참고**

○ 김포공항 내 테스트베드에서 성능 확인 결과, 샤크 형태의 안테나에 비하여 다이폴 형태의 안테나가 약 2배 정도 높은 성능을 보임

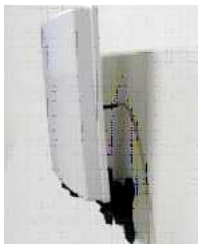
형태	진행내용
샤크	
옴니	
패치	
다이폴	

[표 116] 상용 안테나 조사

○ 상용 안테나 성능 확인

- 이득, 주파수, VSWR, Power, 크기 등을 비교하여 안테나 성능 확인

번호	안테나 형상	주파수	이득	VSWR	Power	Connector
1		2.4~5 GHz	3~5dBi	≤1.5	50W	SMA / N TYPE
2		2.4~2.7 GHz	9.5dBi	≤1.5	50W	TNC, PLUG(male)
3		2.4~2.483 GHz	20dBi	≤1.5	50W	N(female)
4		2.4~2.483 GHz	20dBi	≤1.5	100W	N(female)

번호	안테나 형상	주파수	이득	VSWR	Power	Connector
5		2.4~2.483 GHz	9dBi	≤1.5	50W	N(female)
6		2.4~2.483 GHz	16dBi	≤1.5	50W	N(female)
7		2.4~2.5 GHz	16dBi	≤1.5	50W	N(female)
8		2.4~2.483 GHz	14dBi	≤1.5	50W	N(female)

[표 117] 안테나 성능 비교

- 위와 같이 8가지 안테나의 실내 및 실외 성능시험을 진행한 결과, 5번 다이폴 안테나(표의 음영처리 부분)의 성능이 가장 높은 것으로 확인됨

- 시험결과에 따라 기존의 Wi-Fi 안테나보다 성능이 우수하고, 크기와 수량을 축소하여 적용 완료함

○ 커넥터를 사용한 탈·부착 방식의 안테나

- 다음과 같이 차량용 단말기와 Wi-Fi 안테나는 커넥터로 연결되며 사용자가 쉽게 탈·부착 가능

순번	품명	형상	순번	품명	형상
1	Wi-Fi 안테나		4	차량용 단말기 Wi-Fi 커넥터 (전용단말기 뒷면 우측 상단)	
2	안테나 자석 베이스		5	Wi-Fi 안테나 커넥터와 전용단말기 결합	
3	Wi-Fi 안테나와 안테나 자석 베이스 결합		6	차량 장착	

[표 118] Wi-Fi 안테나 장착

(다) 기 구축된 테스트베드를 활용한 자체 운용시험

① 운용시험 항목

○ 고온 시험(High Temperature Test)

- 고온 50℃ 10시간 유지 시 단말기 동작 이상 유무 확인

○ 저온 시험(Low Temperature Test)

- 저온 -20℃ 10시간 유지 시 단말기 동작 이상 유무 확인

○ 습도 시험(Dampness Test)

- 고온 50℃, 습도 85±5% 10시간 유지 시 단말기 동작 이상 유무 확인

○ 기본 동작 시험(Function Test)

- 기본 상태에서 단말기 기능 동작 이상 유무 확인

② 운용 시험 방법

○ 고온, 저온, 습도 시험을 위한 장비 목록은 다음과 같음

장비명	모델 및 규격	제조사	수량
항온항습기 (TEMP & HUMI CHAMBER)	TH-GU-180 (온도범위 : -50℃ ~ +150℃, 습도범위 : 30% ~ 98%, 내부용적 : 600(W) × 600(D) × 600(H))	(주)제이오텍	1
공항 전용단말기	공항 전용 단말기	인소팩(주)	3
정전압 어댑터	DYC-0520A(DC 5V, 2A)	대유전자(주)	3

[표 119] 운용 시험 장비 목록

○ 고온, 저온, 습도 시험을 위한 구성도는 다음과 같음

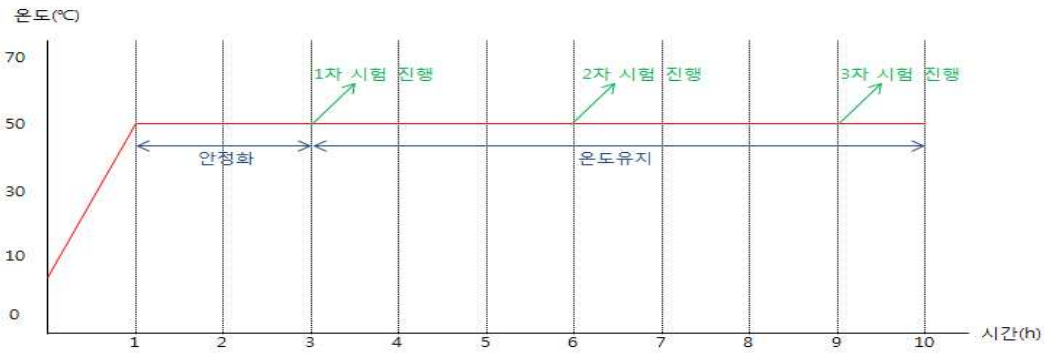


[그림 25] 고온, 저온, 습도 시험 구성도

○ 고온 시험(High Temperature Test)

- 온습도 챔버는 고온 50℃로 설정하여 10시간을 유지하며 3시간 단위로 전용단말기 동작 이상 유무 확인을 위한 기능 검사 진행

- 고온 시험은 다음과 같이 진행됨



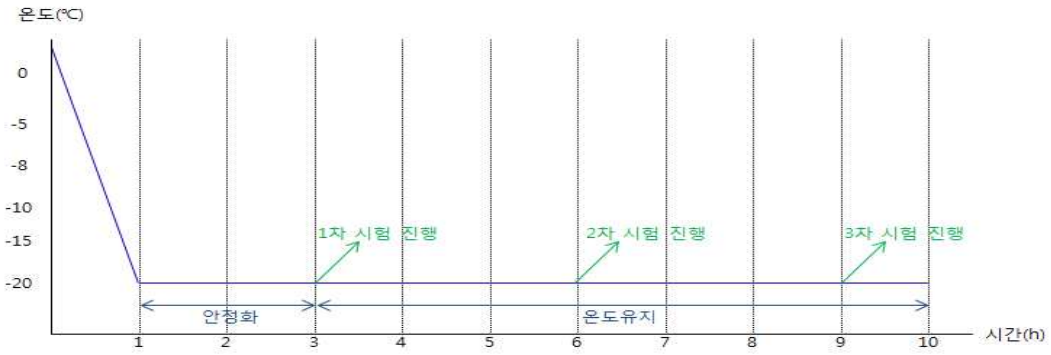
[그림 26] 고온 시험 결과(1차, 2차, 3차 그래프 확인)

- 고온 시험 결과 시료 3대 모두 이상 없음

○ 저온 시험(Low Temperature Test)

- 온습도 챔버는 저온 -20℃로 설정하여 10시간을 유지하며 3시간 단위로 전용단말기 동작 이상 유무 확인을 위한 기능 검사 진행

- 저온 시험은 다음과 같이 진행됨



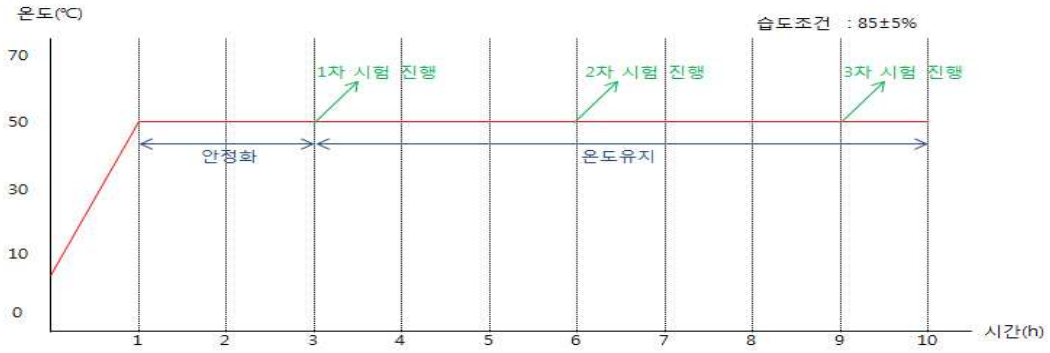
[그림 27] 저온 시험 결과(1차, 2차, 3차 그래프 확인)

- 저온 시험 결과 시료 3대 모두 이상 없음

○ 습도 시험(Dampness Test)

- 온습도 챔버는 상온 50℃, 습도 85±5%로 설정하여 10시간을 유지하며 3시간 단위로 전용단말기의 동작 이상 유무 확인

- 습도 시험은 다음과 같이 진행됨



[그림 28] 습도 시험 결과(1차, 2차, 3차 그래프 확인)

- 습도 시험 결과 시료 3대 모두 이상 없음

○ 기본 동작 시험(Function Test)

- 실온(기본 상태)에서 전용단말기 기능 동작 이상 유무 확인

- 실온 시험 결과 시료 3대 모두 이상 없음

(라) 공인인증 기관을 통한 운용 시험

○ 전용단말기 표준화 적용을 위한 솔루션 설계 및 제작을 통해 내비게이션 성능 구현 가능성 시험 진행

○ 단말기 및 기반 솔루션에 대한 설계를 통한 구체적인 단말기 표준 모델 구성

○ 전용단말기 안테나는 GPS 및 데이터 통신 신호의 송수신이 가능하도록 충분한 적용 시험 진행

○ 설계 후에는 전용단말기 및 솔루션을 구축하여 통합 테스트 진행

○ 전용단말기에 대한 테스트 항목은 주요 기능에 대한 구현 여부, 온도 및 습도에 따른 정상 작동 여부 등을 반영한 다양한 시험을 통해 도출되는 기술 및 정보를 근거로 표준화 인증기관의 검토 및 인증 절차 진행

① 운용시험 항목

○ 고온 시험(High Temperature Test)

- 상온 50℃ 10시간 유지 시 단말기 기본 동작 이상 유무 확인

○ 저온 시험(Low Temperature Test)

- 저온 -20℃ 10시간 유지 시 단말기 기본 동작 이상 유무 확인

○ 습도 시험(Dampness Test)

- 상온 50℃, 습도 85±5% 10시간 유지 시 단말기 기본 동작 이상 유무 확인

○ 기본 동작 시험(Function Test)

- 실온에서 단말기 기본 동작 이상 유무 확인

② 운용시험 방법

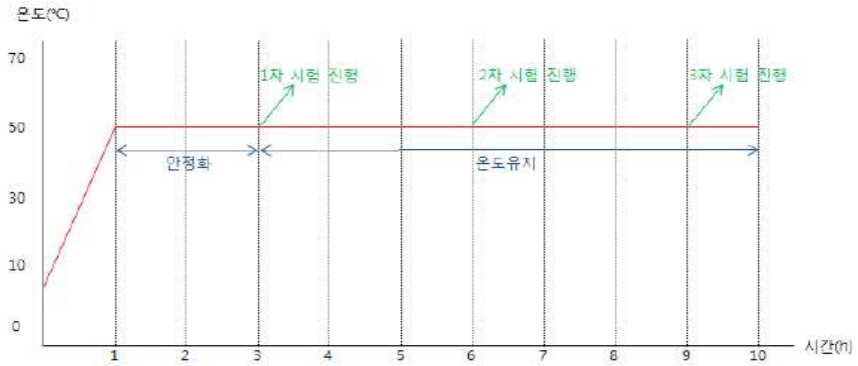
○ 시험장비 목록은 다음과 같음

장비명	모델명	제조사	수량
항온항습기 (TEMP & HUMI CHAMBER)	OS-THB04-C10	오성에스티(주)	1
공항 전용 단말기	공항 전용 단말기	인소팩(주)	10
정전압 어댑터	DYC-0520A	대유전자(주)	10

[표 120] 공인인증 기관을 통한 운용 시험 장비 목록

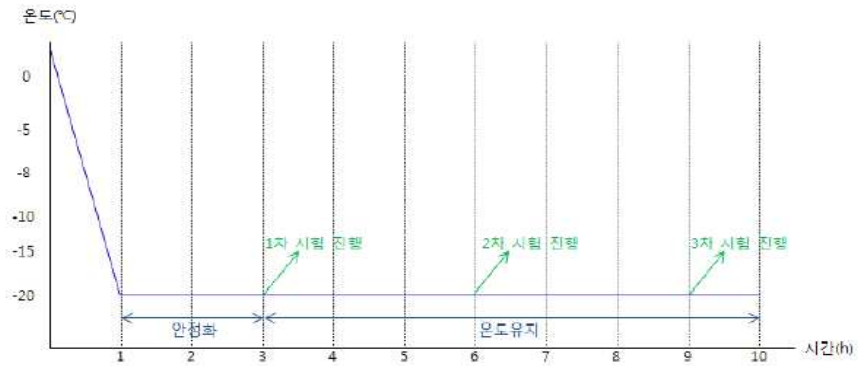
○ 시험조건

- 고온 시험(High Temperature Test) : 상온 50℃, 10시간 유지



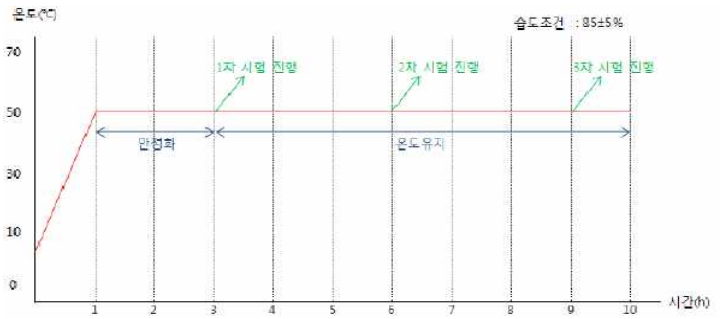
[그림 29] 고온 시험조건 예시

- 저온 시험(Low Temperature Test) : 저온 -20℃, 10시간 유지



[그림 30] 저온 시험조건 예시

- 습도 시험(Dampness Test) : 상온 50℃, 습도 85±5%, 10시간 유지



[그림 31] 습도 시험조건 예시

○ 고온, 저온, 습도 시험을 위한 구성도는 다음과 같음



[그림 32] 고온, 저온, 습도 시험 구성도



[그림 33] 기본 동작 시험 구성도

○ 고온 시험(High Temperature Test) 절차

- 고온 50℃에서 안정화된 시료를 3시간마다 시험 진행
- USB 동작시험 : 단말기에 마우스를 연결 후 정상동작 확인
- SD 카드 동작시험 : 단말기에 SD 카드 삽입 후 SD 카드에 있는 그림 파일이 정상적으로 로딩이 되는지 확인
- GPS 동작 시험 : GPS Test 프로그램 실행 후 위성 데이터가 정상적으로 수신되는지 확인
- Audio 동작 시험 : 단말기에서 Ringtone 실행 후 Audio가 정상적으로 재생되는지 확인
- TSP 동작 시험 : 단말기의 TSP 터치 시 정상 동작되는지 확인
- LCD 동작 시험 : 단말기의 LCD를 육안으로 확인하여 색변화 등 이상유무 확인
- Wi-Fi 동작 시험 : 단말기의 Wi-Fi 동작 후 SCAN 및 접속이 정상적으로 되는지 확인

○ 저온 시험(Low Temperature Test) 절차

- 저온 -20℃에서 안정화된 시료를 3시간마다 시험 진행
- USB 동작시험 : 단말기에 마우스를 연결 후 동작되는지 확인
- SD 카드 동작시험 : 단말기에 SD 카드 삽입 후 SD 카드에 있는 그림 파일이 정상적으로 로딩이 되는지 확인
- GPS 동작 시험 : GPS Test 프로그램 실행 후 위성 데이터가 정상적으로 수신되는지 확인
- Audio 동작 시험 : 단말기에서 Ringtone 실행 후 Audio가 정상적으로 재생되는지 확인
- TSP 동작 시험 : 단말기의 TSP 터치 시 정상 동작되는지 확인
- LCD 동작 시험 : 단말기의 LCD를 육안으로 확인하여 색변화 등 이상유무 확인
- Wi-Fi 동작 시험 : 단말기의 Wi-Fi 동작 후 SCAN 및 접속이 정상적으로 되는지 확인

○ 습도 시험(Dampness Test) 절차

- 고온 50℃, 습도 85±5%에서 안정화된 시료를 3시간마다 시험 진행
- USB 동작시험 : 단말기에 마우스를 연결 후 동작되는지 확인
- SD 카드 동작시험 : 단말기에 SD 카드 삽입 후 SD 카드에 있는 그림 파일이 정상적으로

로딩이 되는지 확인

- GPS 동작 시험 : GPS Test 프로그램 실행 후 위성 데이터가 정상적으로 수신되는지 확인
- Audio 동작 시험 : 단말기에서 Ringtone 실행 후 Audio가 정상적으로 재생되는지 확인
- TSP 동작 시험 : 단말기의 TSP 터치 시 정상 동작되는지 확인
- LCD 동작 시험 : 단말기의 LCD를 육안으로 확인하여 색변화 등 이상유무 확인
- Wi-Fi 동작 시험 : 단말기의 Wi-Fi 동작 후 SCAN 및 접속이 정상적으로 되는지 확인

○ 기본 동작 시험(Function Test) 절차

- 고온, 저온, 습도 시험 완료 후 진행
- USB 동작시험 : 단말기에 마우스를 연결 후 동작되는지 확인
- SD 카드 동작시험 : 단말기에 SD 카드 삽입 후 SD 카드에 있는 그림 파일이 정상적으로 로딩이 되는지 확인
- GPS 동작 시험 : GPS Test 프로그램 실행 후 위성 데이터가 정상적으로 수신되는지 확인
- Audio 동작 시험 : 단말기에서 Ringtone 실행 후 Audio가 정상적으로 재생되는지 확인
- TSP 동작 시험 : 단말기의 TSP 터치 시 정상 동작되는지 확인
- LCD 동작 시험 : 단말기의 LCD를 육안으로 확인하여 색변화 등 이상유무 확인
- Wi-Fi 동작 시험 : 단말기의 Wi-Fi 동작 후 SCAN 및 접속이 정상적으로 되는지 확인

③ 운용 시험 결과

○ 항목별 세부 시험은 다음과 같이 진행됨

구분	시험 진행 시
기본동작 시험	
Wi-Fi 시험	
GPS 시험	

USB 시험	
SD 카드 시험	

[표 121] 항목별 세부 시험 사진

○ 항목별 세부 시험 결과

- 고온 시험(High Temperature Test) 측정 결과

구분	1	2	3	4	5	6	7	8	9	10
USB	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
SD card	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
GPS	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Audio	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
TSP	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
LCD	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Wi-Fi	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상

[표 122] 고온 시험 결과

- 저온 시험(Low Temperature Test) 측정 결과

구분	1	2	3	4	5	6	7	8	9	10
USB	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
SD card	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
GPS	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Audio	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
TSP	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
LCD	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Wi-Fi	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상

[표 123] 저온 시험 결과

- 습도 시험(Dampness Test) 측정 결과

구분	1	2	3	4	5	6	7	8	9	10
USB	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
SD card	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
GPS	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Audio	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
TSP	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
LCD	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Wi-Fi	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상

[표 124] 습도 시험 결과

- 기본 동작 시험(Function Test) 측정 결과

구분	1	2	3	4	5	6	7	8	9	10
USB	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
SD card	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
GPS	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Audio	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
TSP	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
LCD	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상
Wi-Fi	정상	정상	정상	정상	정상	정상	정상	정상	정상	정상

[표 125] 기본 동작 시험 결과

○ 공인 인증기관 시험 결과서는 다음과 같음

"혼(魂).창(創).통(通)을 지닌 Future Creator"

ETRI 한국전자통신연구원

수신자 인소팩(주)
(경유)
제 목 2014년도 인소팩(주) 공항 전용단말기 신뢰성 시험결과서

1. 귀 사의 무궁한 발전을 기원합니다.

2. 당 부서에서는 정보통신 중소기업에 고주파 시험 및 장비 등의 기술지원을 수행하고 있습니다.

3. 이와 관련하여 아래와 같이 시험의뢰한 귀 사의 공항 전용단말기 1종에 대한 신뢰성 시험결과서를 붙임과 같이 송부하오니 참고하시기 바랍니다.

- 아 래 -

가. 의뢰기업 : 인소팩(주)
나. 업체주소 : 경북 구미시 진평동 12-1 MHK 빌딩 401호
다. 시험대상 : 공항 전용단말기, 10 대
라. 시험조건 : 고온, 저온, 습도시험 후 기본동작 시험(세부시험조건 첨부문서 참조)
마. 시험기간 : 2014.04.28.~2014.04.30.
바. 시험확인서 : "TR14-054 인소팩(주) 공항전용단말기 신뢰성 시험결과서"1부. 붙임 참조

붙임 : "TR14-054 인소팩(주) 공항전용단말기 신뢰성 시험결과서"1부. 끝.

한국전자통신연구원 원장

수신자 인소팩(주)

담당 김영구 팀장 김태홍

협조자

시행 9363-2014-10134 (2014.05.07.) 접수 ( )

우 305-700 대전 유성구 가정로 218(가정동) / www.etri.re.kr

[그림 34] 공항 전용단말기 시험 결과서

다. 공항시설 안전통합 관리시스템 전용단말기 소프트웨어 개발

(1) 안드로이드 운영체제 포팅 및 최적화

(가) 개발환경 구성

① 부트 로더 설정

○ Embedded Linux BSP는 부트 로더와 커널, 파일 시스템의 3가지 이미지 파일들로 구성

○ 부트 로더는 커널을 메모리 적재하기 위해 필요한 프로그램으로 설정은 다음과 같음

- 기존 부트 로더 파일의 압축을 해제

  • #tar vxfz u-boot-mv4412-dev.tar.gz

- vi 에디터를 이용하여 “mv4412.h”파일을 열기 하면 부트 로더에 필요한 기본 환경 설정을 할 수 있음(예: TFTP, CPU 클럭, DDR Program Counter)

- /SRC/Android/q3-compiler 안에 있는 “arm-2009q3-67-arm-none-linux- gnueabi.bin”를 설치

  • #./arm-2009q3-67-arm-none-linux-gnueabi.bin

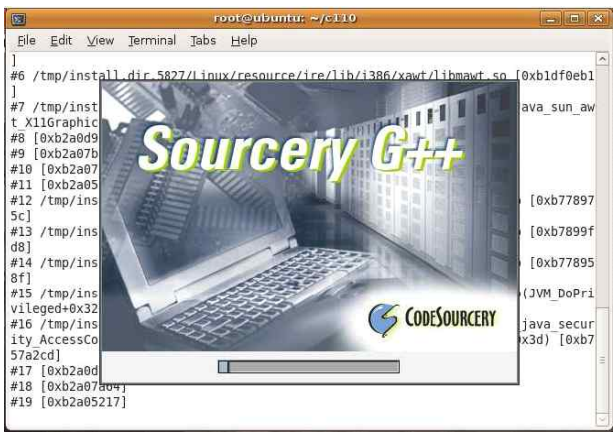
- Q3를 설치 할 때는 GUI 화면 창으로 설치하기 때문에 콘솔이 아닌 리눅스 PC가 설치되어 있는 PC에서 설치해야 함

- 만약 “% sudo depkg-reconfigure -plow dash” 라는 메시지가 출력된다면 다음의 명령을 실행

  • sudo dpkg-reconfigure -plow dash

- Yes/No 화면이 출력되면 “No”를 선택한 후 처음에 진행하였던 명령어를 추가로 입력

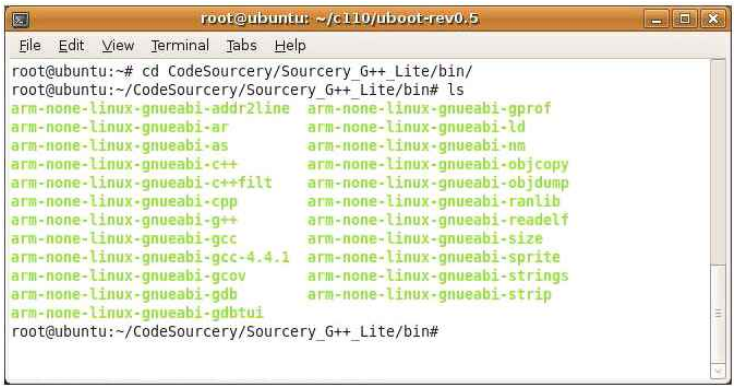
- ./sudo sh arm-2009q3-67-arm-none-linux-gnueabi.bin



[그림 35] Sourcery G++ 실행화면

- GUI의 설명을 통하여 추가적인 설치 진행

- 설치가 완료되면 “/opt/CodeSourcery/Sourcery_G++_Lite/bin”안에 설치된 Q3 라이브러리 확인



[그림 36] Sourcery G++ 설치 확인

- Q3 설치가 완료되면 부트 로더의 컴파일을 위하여 build.sh 수정

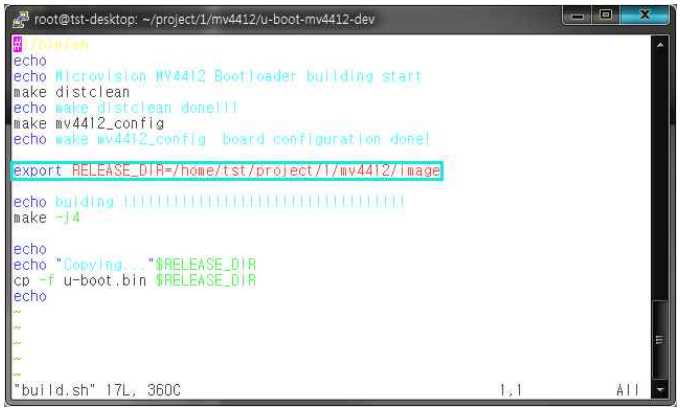
  - #vi build.sh



[그림 37] 부트 로더 빌드 실행 파일 열기

- 컴파일 후에 생성된 이미지가 복사될 폴더의 경로 설정

  - export RELEASE_DIR=/home/tst/project/mv4412/image로 수정한다.



[그림 38] 부트 로더 이미지 파일 경로 설정

- Q3 컴파일러의 실행 경로 설정

  - #export PATH=$PATH:/opt/Code Sourcery/Sourcery_G++_Lite/bin
  - #export CROSS_COMPILE=arm-none-linux-gnueabi-gcc
  - #source ~/.bashrc



[그림 39] 부트 로더 빌드를 위한 컴파일러 설정

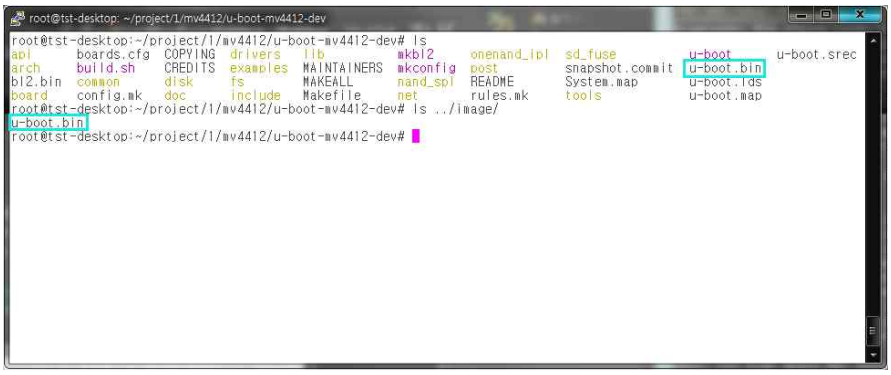
- 설정 완료 후 컴파일 실행

  - #./build.sh



[그림 40] 부트 로더 빌드 실행

- 컴파일 완료 후 설정한 이미지 복사 경로에 u-boot.bin 파일 확인



[그림 41] 부트 로더 빌드 완료 확인

② 리눅스 커널 컴파일(버전: 3.0.15)

○ 기본 커널 파일의 압축을 해제

○ 커널을 컴파일하기 위하여 build.sh 수정

- #vi build.sh

○ 컴파일 후 생성된 이미지가 복사될 폴더의 경로 설정

- export RELEASE_DIR=/home/tst/project/mv4412/image로 수정한다.

○ 커널 바인딩 과정은 크게 커널 설정 과정과 커널 소스 컴파일 과정으로 분류됨

- 커널 설정 과정

  • 커널(Kernel)이란 운영체제(OS)를 일컫는 말로 커널에서 제공하는 기능은 다양하기 때문에 리눅스에서는 이를 각각 설정할 수 있도록 지원해주어야 함

  • 설정 시 반드시 필요한 부분은 커널이 탑재될 대상이 되는 프로세스(CPU)의 설정이며, 이외 타겟 보드의 하드웨어 사양에 따라 커널에서 기본적으로 설정되어 있는 값들을 가감함

  • 커널 모듈에 관한 지원 옵션의 설정 또한 추후 디바이스 드라이버의 모듈 형태 파일을 Insmod 명령어를 통해 인식시킬 수 있으니 주의가 필요함

- 커널 소스 컴파일 과정

  • 커널 설정 완료 후 컴파일 하는 방법은 다른 일반적인 소스 컴파일과 동일하게 make 유틸리티를 이용함

  • 커널은 다른 일반 소스처럼 컴파일과 링킹 과정을 완료한 실행 가능한 ELF 파일을 그대로 사용할 수 없으므로 arm-linux-objcopy라는 binutil을 사용하여 ELF 파일의 헤더 부분을 삭제하고 바이너리 파일 타입으로 변경한 후 다시 이를 gzip으로 압축한 커널 이미지, 즉 zImage 파일을 생성하는 것이 최종 목적임. 다른 일반 소스와는 달리 컴파일 명령을'make zImage'로 실행시킴

○ 컴파일을 위한 커널 환경설정 모드 실행

- #make menuconfig



[그림 42] 환경설정 모드 진입

○ 커널 환경설정 모드에서 “Load an Alternate Configuration File” 메뉴를 선택하여 “arch/arm/configs/mv4412_defconfig”를 로딩



[그림 43] 환경설정 모드



[그림 44] 환경설정 파일 로드

○ 로딩이 완료되면 설정된 파일의 확인 및 저장 후 종료

- “save an Alternate Configuration File”메뉴 선택 후 저장

○ Q3 컴파일러의 실행 경로 설정

- #export PATH=$PATH:/opt/Code Sourcery/Sourcery_G++_Lite/bin

- #export CROSS_COMPILE=arm-none-linux-gnueabi-gcc

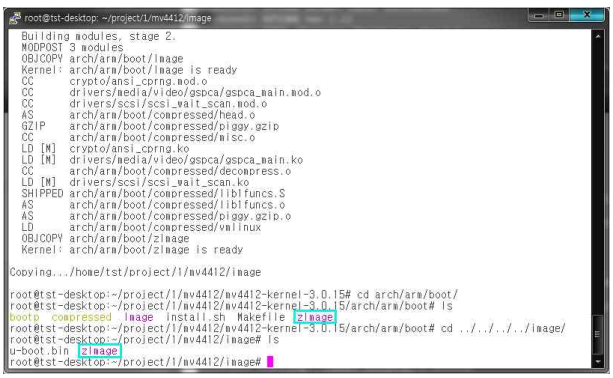
- #source ~/.bashrc



[그림 45] 커널 빌드를 위한 컴파일러 설정

○ 설정 완료 후 컴파일 실행

- #./build.sh

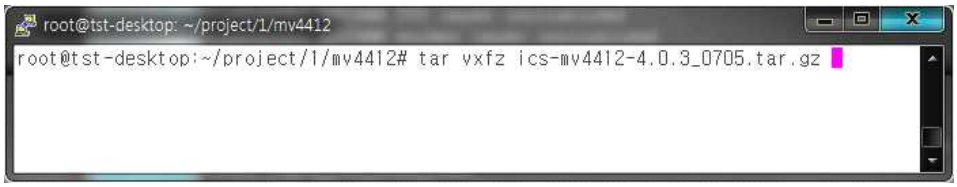


[그림 46] 커널 빌드 완료 확인

③ 안드로이드 운영체제 컴파일(버전: 아이스크림 샌드위치(ICS) 4.0.3)

○ 기본 안드로이드 운영체제 파일의 압축 해제

- #tar vxfz ics-mv4412-4.0.3_0705.tar.gz



[그림 47] 안드로이드 운영체제 파일 압축 해제

○ 컴파일 전 /u-boot-mv4412-dev/tools 안에 있는 컴파일 완료된“mkimage”파일 export

- #export PATH=$PATH:/home/tst/project/mv4412/u-boot-mv4412-dev/ tools

- #source ~/.bashrc

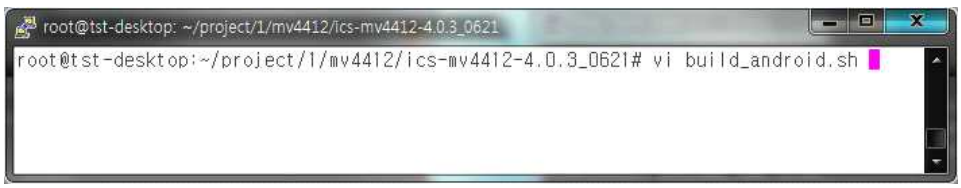


[그림 48] 안드로이드 운영체제 빌드 설정

○ 안드로이드 운영체제인 아이스크림 샌드위치를 컴파일하기 위하여 JDK1.6 버전 설치

○ “build_android.sh”파일을 열기 후 JDK가 설치된 경로 설정

- #vi build_android.sh



[그림 49] 안드로이드 운영체제 빌드 파일 열기

○ Kernel 경로 설정

- KERNEL_DIR=/home/tst/project/mv4412/mv4412-kernel-3.0.15



[그림 50] 안드로이드 빌드 시 참조할 커널 파일의 경로 설정

○ 컴파일 후 생성된 이미지가 복사될 폴더 경로 설정

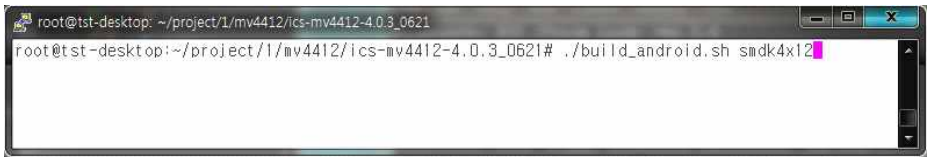
- export RELEASE_DIR=/home/tst/project/mv4412/image



[그림 51] 안드로이드 빌드 후 생성파일 경로 설정

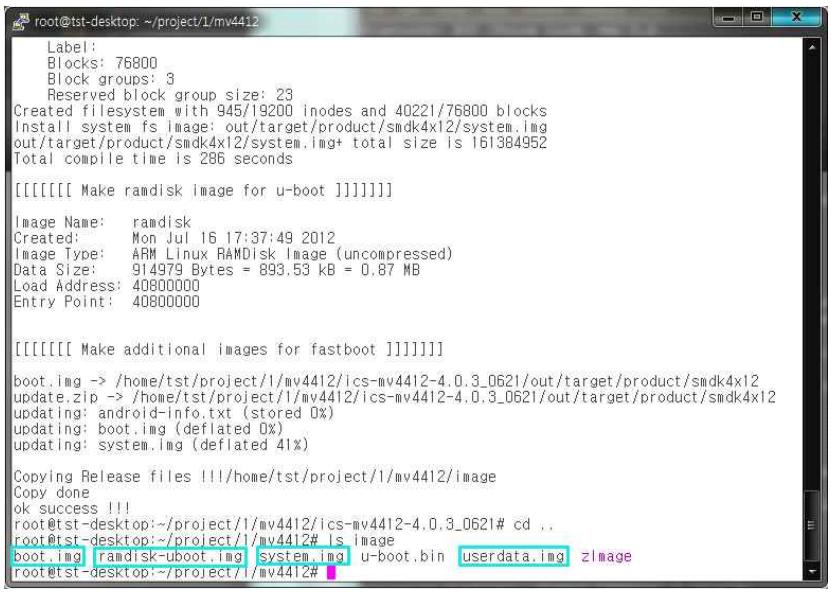
○ 컴파일 실행

- #./build_android.sh smdk4x12



[그림 52] 안드로이드 운영체제 빌드 실행

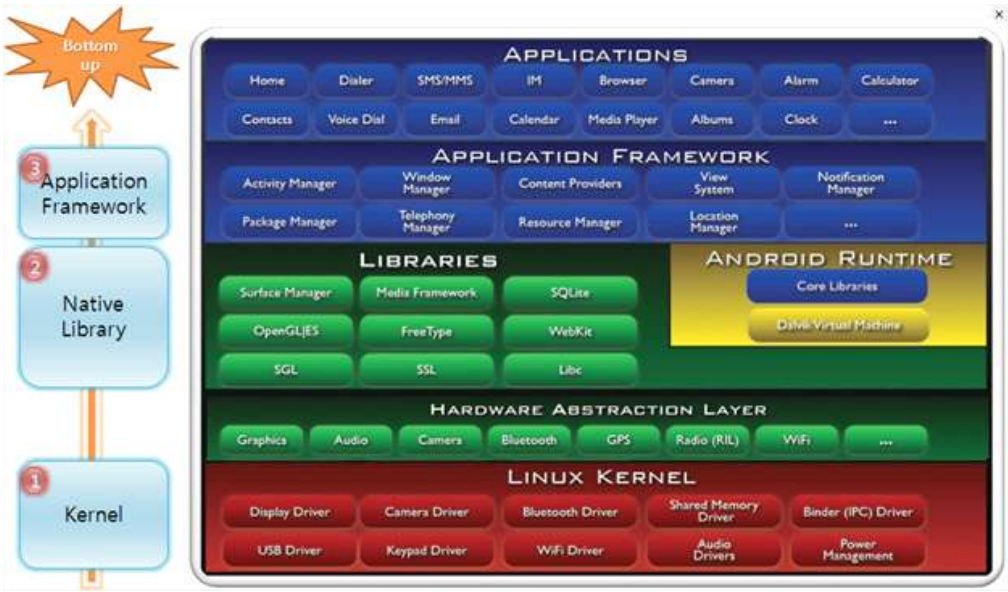
○ 컴파일 성공 후 생성된 이미지 확인



[그림 53] 안드로이드 운영체제 빌드 확인

(나) 안드로이드 운영체제 분석

① 운영체제 구조



[그림 54] 안드로이드 운영체제 구조

○ 안드로이드 운영체제는 크게 커널, 하드웨어 추상 레이어, 라이브러리, 응용프로그램 프레임워크로 분류됨

○ 라이브러리들은 기존의 여러 프로젝트들을 통합하였기 때문에 대부분 C 언어로 구현되어 있음

○ 응용프로그램 프레임워크와 라이브러리 사이에는 Java 언어와 C 언어사이의 서로 다른 호출 규약 등을 맵핑할 수 있는 JNI라는 마셜러가 존재

○ 임베디드 리눅스와 안드로이드의 차이점 중 한 가지는 장치 드라이버들(그래픽, 오디오, 카메라 등)이 커널 영역이 아닌 유저 모드에서 동작한다는 것임

○ 유저 모드로 동작시킬 경우 자신에게 할당되어 있는 메모리 번지에만 접근할 수 있기 때문에 조금 더 안전하지만 하드웨어 제어 시에는 MMU 등을 참조하는 등의 추가적인 작업이 필요함

○ 드라이버들을 유저 영역에 포함시키는 이유는 라이선스 문제와 안정성, 문서화의 편리성, 플러그인 방식으로 드라이버 배포가 가능하기 때문임

○ 동일한 라이브러리라도 안드로이드의 라이브러리는 스마트폰에 탑재하는 것을 기준으로 불필요한 코드를 제거하고 최적화되었기 때문에 동작하지 않음

○ 안드로이드가 리눅스 커널을 기반으로 개발된 이유는 오픈 소스이며 안전성이 입증된 드라이버 모델 적용, 메모리 프로세스 관리 모델 제공, 하드웨어의 입출력관련 보안모델 제공, 공유 라이브러리 지원 등으로 개발 비용과 시간을 절감할 수 있기 때문임



[그림 55] 안드로이드 커널 구조

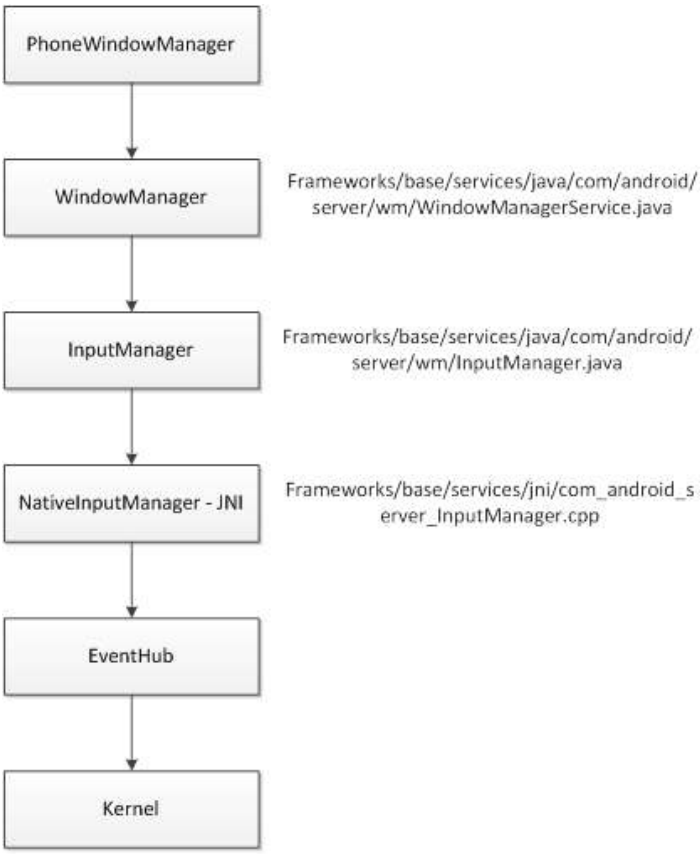
② 입력 장치(Input Device) 분석

○ 리눅스에서 입력 장치는 디바이스 드라이버 표준화 작업을 말함

○ 키보드 디바이스 드라이버의 경우 커널 2.4 버전에서는 tty 디바이스 드라이버로 설정되지만 커널 2.6버전 이후에서는 입력 장치의 이벤트 인터페이스로 먼저 구성되고 이후에 tty 디바이스 드라이버로 매핑되는 구조로 변경됨

○ 커널에 따른 변화는 앞으로의 기술 발전에 따라 어떠한 카테고리의 입력 장치가 개발되고 탄생할지 알 수 없는 상황에서 모든 입력 장치에 대한 카테고리를 만드는 일 자체가 거의 불가능하고 정리 되지 않기 때문임

○ 위와 같은 문제점을 해결하고 동일한 인터페이스로 모든 입력장치를 처리하기 위하여 이벤트 인터페이스라는 새로운 인터페이스를 중간 계층에 두고 그 이후에 입력 장치에 대한 카테고리로 매핑을 할 수 있도록 커널의 구조를 새롭게 구성함



[그림 56] 입력 장치(Input Device) 구조

○ 위의 그림은 입력 매니저부터 네이티브 함수까지의 처리 과정을 나타냄

○ 기본적인 소스파일 경로는 다음과 같음

- Frameworks/base/services/java/com/android/server/wm/WindowManagerService.java
- Frameworks/base/services/java/com/android/server/wm/InputManager.java
- Frameworks/base/services/jni/com_android_server_InputManager.cpp
- Frameworks/base/services/input/InputManager.cpp

○ WindowManagerService.java와 InputManager.java는 응용프로그램 프레임워크 관련 코드이며 com_android_server_InputManager.cpp는 JNI, InputManager.cpp 코드는 JNI를 통하여 실행되는 네이티브 코드 파일임

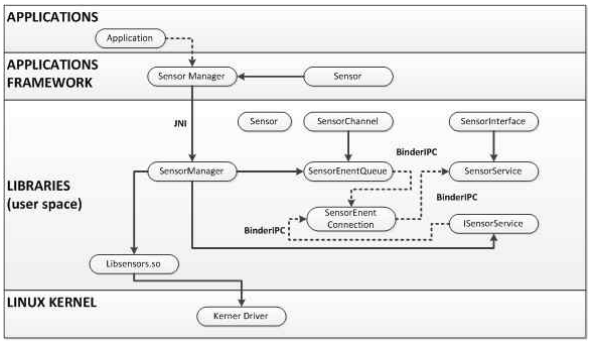
③ 센서 서브시스템 분석

○ 안드로이드 운영체제 2.2 버전인 프로요와 비교하여 진저브레드 이후의 버전에서 변화된 구조를 살펴보면 우선 SensorService가 SensorManager에서 분리되어 있음

○ SensorService아래에 ISensorServer라는 바인더 클래스가 존재하며 실제 SensorService는 ISensorService를 상속받으므로 의미상 SensorServer가 되며 이는 SensorManager가 클라이언트 역할을 하고 SensorService가 서버 역할을 하며 동작한다는 것을 의미함

센서	Java 상수	센서 타입	값
가속도	TYPE_ACCELOMETER	SENSOR_TYPE_ACCELOMETER	1
자기장	TYPE_MAGNETIC_FIFLD	SENSOR_TYPE_MAGNETIC_FIFLD	2
위치	TYPE_ORIENTATION	SENSOR_TYPE_ORIENTATION	3
수평	TYPE_GYROSCOPE	SENSOR_TYPE_GYROSCOPE	4
조도	TYPE_LIGHT	SENSOR_TYPE_LIGHT	5
압력	TYPE_PRESSURE	SENSOR_TYPE_PRESSURE	6
온도	TYPE_TEMPERATURE	SENSOR_TYPE_TEMPERATURE	7
근접	TYPE_PROXIMITY	SENSOR_TYPE_PROXIMITY	8
중력	TYPE_GRAVITY	SENSOR_TYPE_GRAVITY	9
선가속도	TYPE_LINEAR_ACCELERATION	SENSOR_TYPE_LINEAR_ACCELERATION	10
회전축	TYPE_ROTATION_VECTOR	SENSOR_TYPE_ROTATION_VECTOR	11

[표 126] 센서의 종류



[그림 57] 센서 서브시스템 구성

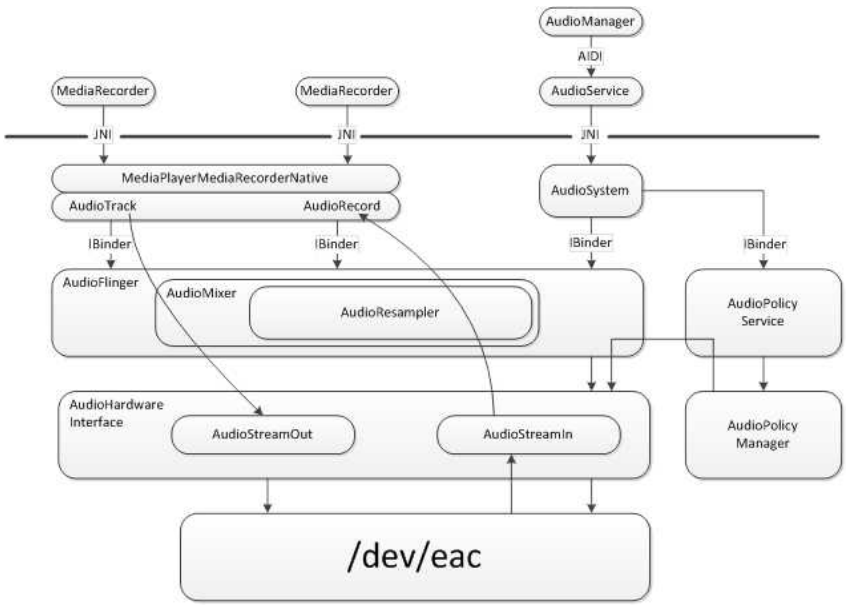
④ 오디오 서브시스템 분석

- 오디오 서브시스템이 많은 장치 및 응용프로그램, 다른 서브시스템과 연결되는 경우가 많기 때문에 가장 복잡한 부분임

- 안드로이드 제품을 개발할 경우 오디오 부분 많은 문제가 발생되므로 사운드와 연관되어 있는 멀티미디어 플레이어/레코드, VOIP, DMB 등 관련 개발 시 많은 수정과 테스트가 필요함

- 오디오 서브시스템은 크게 Audio Manager부터 AudioService, AudioSystem, AudioPolicyService, AudioPolicyManager로 가는 제어 경로와 MedioPlayer/ MediaRecord 부터 AudioFlinger를 이용하는 데이터 경로로 분류됨

- 오디오 서브시스템은 다른 서브시스템과 마찬가지로 수평적인 구조의 형태로 본다면 바인더를 이용한 클라이언트-서버의 구조로 이루어짐



[그림 58] 오디오 서브시스템 구조

(다) Wi-Fi 드라이버 포팅

○ 기본적인 Wi-Fi 구성은 SDIO 인터페이스를 이용하여 Wi-Fi와 연결되어 있으며 SD 카드 타입의 모듈을 내장하였기 때문에 CD Detector 기능을 추가적으로 포팅

○ Wi-Fi 드라이버의 CD Detector 기능은 인터럽트, GPIO, 그라운드(GND)의 3가지 방식이 있지만 하드웨어 구성은 GPIO로 연결되어 있기 때문에 GPIO 제어를 통한 CD Detector 기능의 구현 필요

- 경로: Kernel/mv4412-kernel-3.0.15/arch/arm/mach-exynos/mach-smdk4x12.c

```
s3c_gpio_cfgpin(EXYNOS4212_GMP4(7), S3C_GPIO_OUTPUT);
s3c_gpio_setpull(EXYNOS4212_GMP4(7), S3C_GPIO_PULL_DOWN);
s5p_gpio_set_drvstr(EXYNOS4212_GMP4(7), S3C_GPIO_DRVSTR_LV4);

mdelay(100);

gpio_set_value(EXYNOS4212_GMP4(7), 0);
```

[그림 59] CD Detector PIN GPIO 설정

- s3c_gpio_cfgpin 함수

  • GPIO의 입력, 출력 모드 설정

  • S3C_GPIO_OUTPUT는 GPM4.7을 출력으로 설정

  • 함수 원본

```
int s3c_gpio_cfgpin(unsigned int pin, unsigned int config)
{
    struct s3c_gpio_chip *chip = s3c_gpiolib_getchip(pin);
    unsigned long flags;
    int offset;
    int ret;

    if(!chip)
        return -EINVAL;

    offset = pin - chip->chip.base;

    s3c_gpio_lock(chip, flags);
    ret = s3c_gpio_do_setcfg(chip, offset, config);
    s3c_gpio_unlock(chip, flags);

    return ret;
}
```

[그림 60] s3c_gpio_cfgpin 함수

- s3c_gpio_setpull 함수

  - GPIO의 초기 상태를 풀 업, 풀 다운 등으로 설정
  - S3C_GPIO_PULL_DOWN는 GPM4.7을 풀다운 상태로 설정
  - 함수 원본

```
int s3c_gpio_setpull(unsigned int pin, s3c_gpio_pull_t pull)
{
    struct s3c_gpio_chip *chip = s3c_gpiolib_getchip(pin);
    unsigned long flags;
    int offset, ret;

    if(!chip)
        return -EINVAL;

    offset = pin - chip->chip.base;

    s3c_gpio_lock(chip, flags);
    ret = s3c_gpio_do_setpull(chip, offset, pull);
    s3c_gpio_unlock(chip, flags);

    return ret;
}
```

[그림 61] s3c_gpio_setpull 함수

- s5p_gpio_set_drvstr 함수

  - GPIO의 우선순위를 레벨 형태로 설정
  - S5P_GPIO_DRVSTR_LV4는 Level.4를 설정
  - 함수 원본

```
s5p_gpio_set_drvstr_t s5p_gpio_get_drvstr(unsigned int pin)
{
    struct s3c_gpio_chip *chip = s3c_gpiolib_getchip(pin);
    unsigned int off;
    void __iomem *reg;
    int shift;
    u32 drvstr;

    if(!chip)
        return -EINVAL;

    off = pin - chip->chip.base;
    shift = off * 2;
    reg = chip->base + 0x0C;

    drvstr = __raw_readl(reg);
    drvstr = drvstr >> shift;
    drvstr &= 0x3;

    return (__force s5p_gpio_drvstr_t)drvstr;
}
```

[그림 62] s5p_gpio_get_drvstr 함수

- gpio_set_value 함수

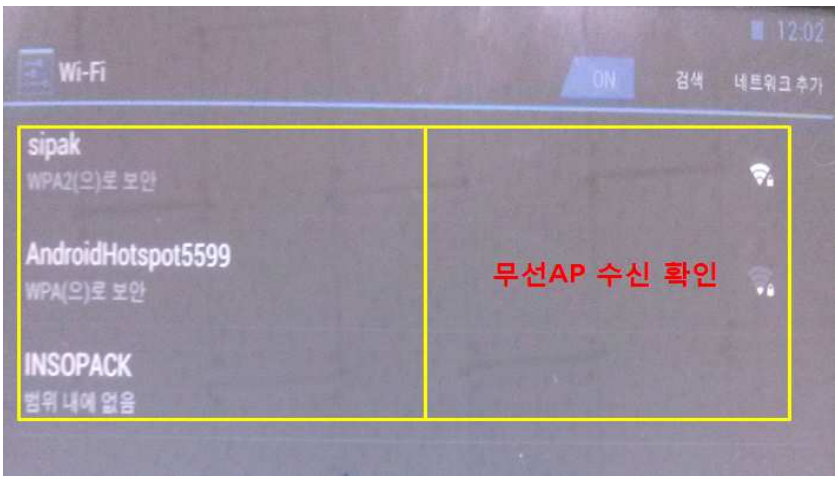
  - 해당 GPIO의 HIGH, LOW 값 설정
  - GPM4.7을 LOW로 설정
  - 함수 원본

```
void __gpio_set_value(unsigned int gpio, int value)
{
   struct gpio_chip  *chip;

   chip = gpio_to_chip(gpio);
   WARN_ON(chip->can_sleep);
   trace_gpio_value(gpio, 0, value);
   chip->set(chip, gpio - chip0->base, value);
}
```

[그림 63] gpio_set_value 함수

○ 안드로이드에서 Wi-Fi 기능을 활성화한 후 Wi-Fi 모듈을 이용한 주변 SSID 검색 및 접속으로 정상적인 동작 유무 확인



[그림 64] Wi-Fi 포팅 후 무선 AP 수신 확인

(라) 안드로이드 운영체제 최적화

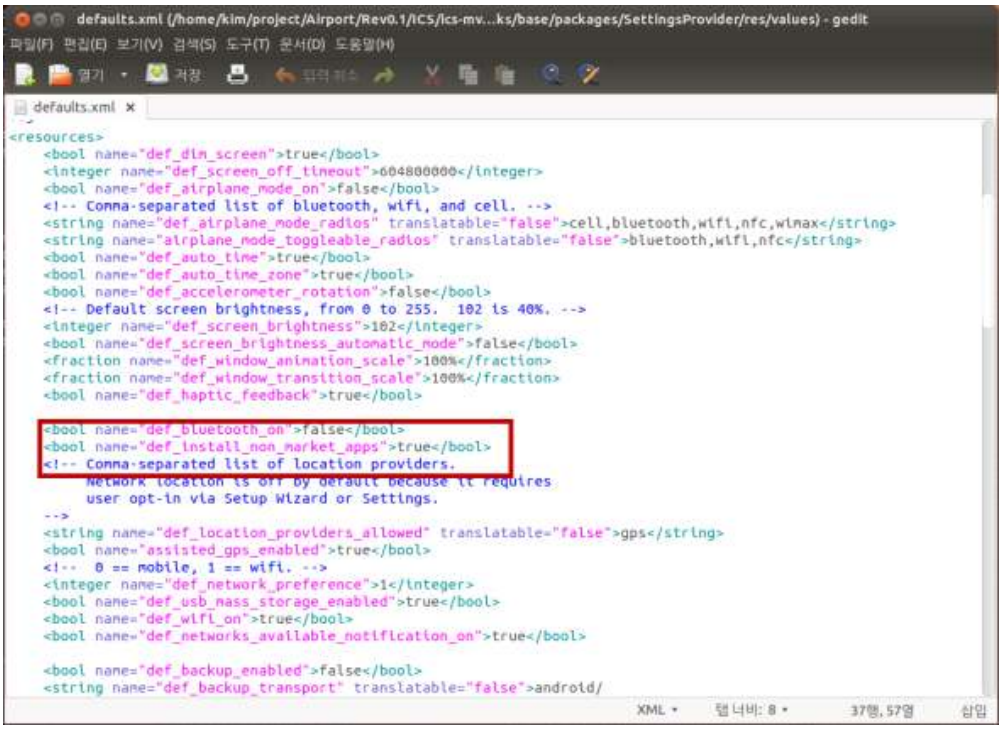
① 보안 해제 설정

○ 안드로이드 운영체제는 기본적으로 외부 소스 파일들의 다운로드가 불가능 하도록 보안 설정되어 있음

○ 전용단말기의 최신 버전 업데이트 기능을 구동하기 위해 보안 해제 상태를 기본 설정으로 변경함

- 운영체제 소스 위치 : ICS(OS_Version)/frameworks/base/packages/ SettingsProvider/ res/values/defaults.xml

- Line 37의 def_install_non_market_apps의 속성을 false에서 true로 변경



[그림 65] 운영체제 보안해제 설정

② 아이콘 터치 음량 수정

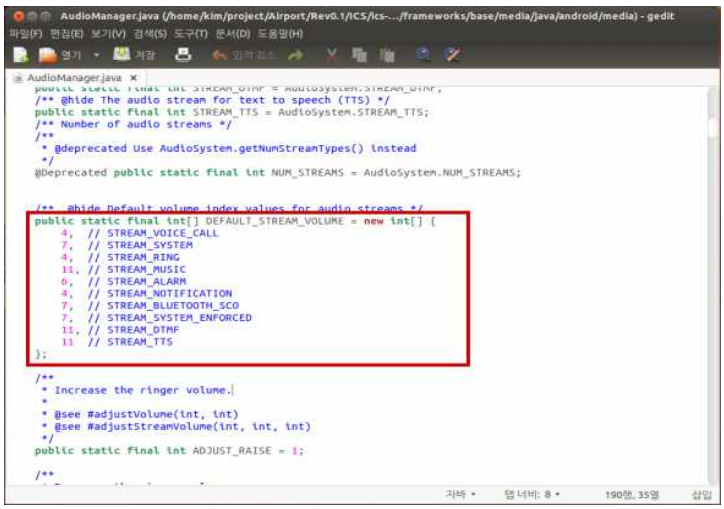
○ 전용단말기의 볼륨 조절키는 운영체제의 미디어 볼륨만 제어가 가능하기 때문에 기타 나머지 볼륨들은 기본값으로 수정함

○ 아이콘을 터치하였을 때 발생하는 음량은 차량 운전 중, 인식성을 향상시키기 위해 높은 음량을 필요로 함

- 운영체제 소스 위치 : ICS(OS_Version)/frameworks/base/media/java/ android/media/

AudioManager.java

- Line 176의 DEFAULT_STREAM_VOLUME 배열의 값 중 STREAM_RING을 3에서 4로 수정



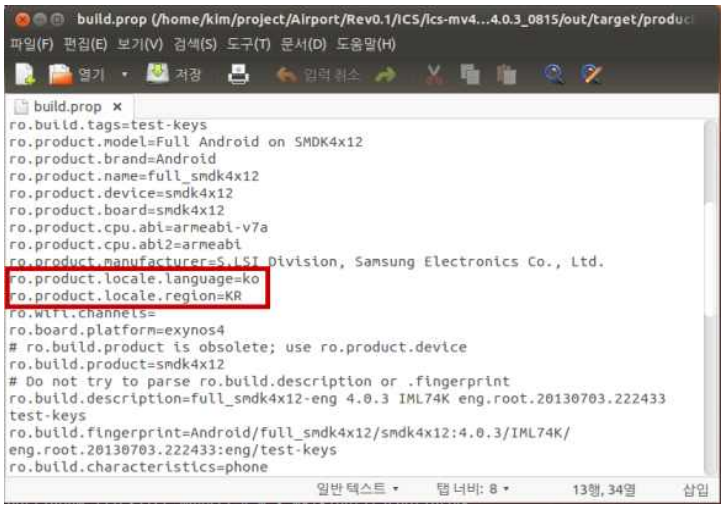
[그림 66] 아이콘 터치 음량 변경

③ 운영체제 지원 언어 변경

- 운영체제의 기본 지원 언어는 영어로 되어 있으며 사용자의 편의를 위하여 기본 지원 언어를 한글로 변경함

- 운영체제 소스 위치 : ICS(OS_Version)/out/target/product/smdk4x12/ system/build.prop

- build.prop 속성 중 “ro.product.locale.language”를 en에서 ko로 수정하였으며 “ro.product.locale.region”을 EN에서 KR로 수정



[그림 67] 운영체제 지원 언어 변경

④ Home.apk Launch 삭제

○ 전용단말기는 부팅 시 응용프로그램이 자동으로 실행되기 때문에 기본적인 Launch 프로그램은 삭제함

○ 안드로이드 운영체제에는 기본적으로 Home Launch 프로그램이 구동됨

○ Home Launch 프로그램을 삭제하지 않게 되면 초기 부팅 시 운영체제 화면이 출력된 후 개발된 응용프로그램이 실행되기 때문에 삭제함

- 운영체제 소스 위치 : ICS(OS_Version)/development/samples/Home

- Out 디렉토리에 생성되는 Home.apk 파일은 Home 디렉토리에서 생성되며 폴더 전체를 삭제한 후 Out 디렉토리의 Home.apk 파일도 삭제함



[그림 68] Home.apk 생성 디렉토리

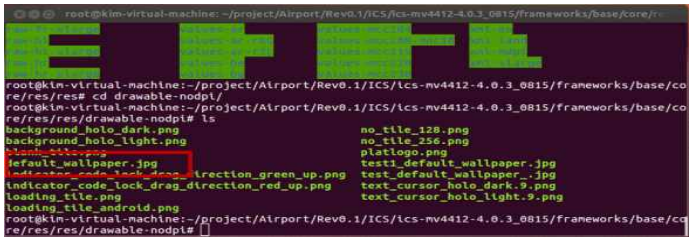
⑤ 운영체제 기본 배경화면 수정

○ 운영체제에 설정되어 있는 기본 배경화면을 수정함으로써 전용단말기에 대한 사용자 인식을 향상시킴

○ 전용단말기의 부팅 애니메이션이 구동된 후 배경을 검은색 화면 이미지로 변경함으로써 자연스러운 프로그램 인식을 유도함

- 운영체제 소스 위치 : ICS(OS_Version)/frameworks/base/core/res/res/ default-wallpaper.jpg

- default-wallpaper.jpg를 검은색 화면 이미지 파일로 수정

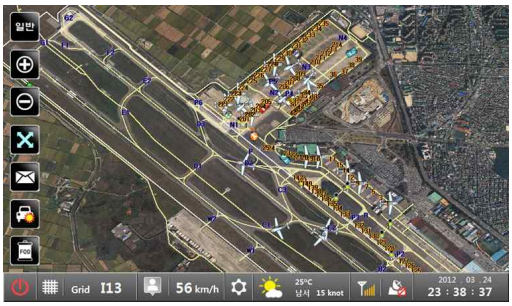


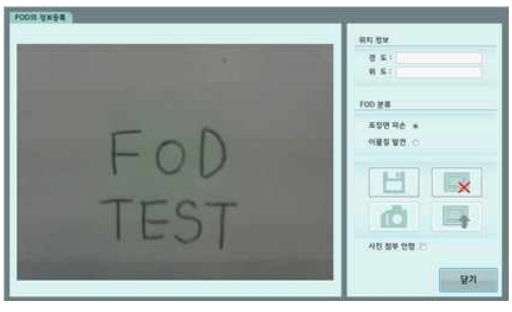
[그림 69] 운영체제 기본 배경화면 저장 위치

(2) GUI 개발 및 기능 구현

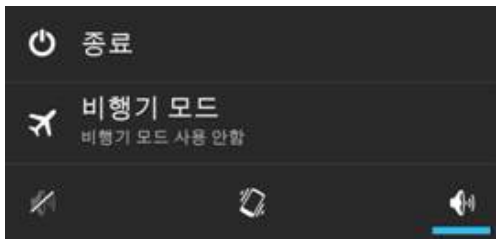
(가) UI 구성 및 디자인 변경

○ 응용프로그램의 기능화면별 UI 디자인 변경 전과 후는 다음과 같음

<table>
<tr><th>구분</th><th>변경 전</th><th>변경 후</th></tr>
<tr><td rowspan="2">메인
화면</td><td> 
</td><td> 
</td></tr>
<tr><td colspan="2">• 공항 전용단말기 개발로 인해 화면의 해상도가 변경됨에 따라 지도화면에 출력되는 정보를 최소화 할 수 있도록 지도조작 외의 기능을 하나의 메뉴로 통합
• UI 구성 및 디자인 변경 후 하단 바에는 현재 차량의 속도, 위치정보에 대한 Grid 정보, 통신상태 설정 등 디스플레이용 기능들을 배치</td></tr>
<tr><td rowspan="2">메뉴
화면</td><td> 
</td><td> 
</td></tr>
<tr><td colspan="2">하단 바 및 지도화면에 위치하였던 수신 메시지 확인, FOD 및 사고 등록, Grid Map, 환경설정 기능 아이콘을 모두 메뉴로 통합하여 전체적인 구성을 변경</td></tr>
</table>

구분	변경 전	변경 후
수신 메시지 확인 화면		
	수신 메시지를 확인할 수 있는 정보창의 상, 하 버튼 및 사고 등의 발생 위치로 이동 버튼, 조치완료 버튼과 전체적인 배경 이미지 변경	
사고 정보 등록 화면		
	공항 내 사고 발생 시 관련 정보를 등록할 수 있는 정보창의 전체적인 디자인과 사고종류, 인명피해, 화재피해 좌, 우 선택 버튼의 이미지 변경	
FOD 정보 등록 화면		
	공항 내 FOD 발견 시 관련 정보를 등록할 수 있는 정보창의 전체적인 디자인과 버튼 등의 구성 및 이미지 변경	

<table>
<tr><th>구분</th><th>변경 전</th><th>변경 후</th></tr>
<tr><td rowspan="2">설정
화면</td><td> 
</td><td> 


</td></tr>
<tr><td>• 무선 네트워크 설정 및 응용프로그램의 정보 확인을 위한 환경 설정창의 전체적인 디자인 변경
• 항공기와 차량의 범례는 상세보기 버튼을 추가로 배치하여 별도의 창에서 정보를 제공하도록 변경</td></tr>
<tr><td rowspan="2">시스템
종료
화면</td><td> 
</td><td> 
</td></tr>
<tr><td>기존 UMPC 등의 사용으로 시스템 종료, 시스템 재부팅, 차량단말 서브시스템의 프로그램 종료로 분류되어 있던 종료 기능을 전용단말기에 최적화하여 하드웨어 전원 버튼을 누를 경우 프로그램부터 전용단말기의 전원까지 모두 종료되도록 UI 구성 및 디자인 변경</td></tr>
</table>

[표 127] 응용프로그램의 UI 구성 및 디자인 변경 전, 후

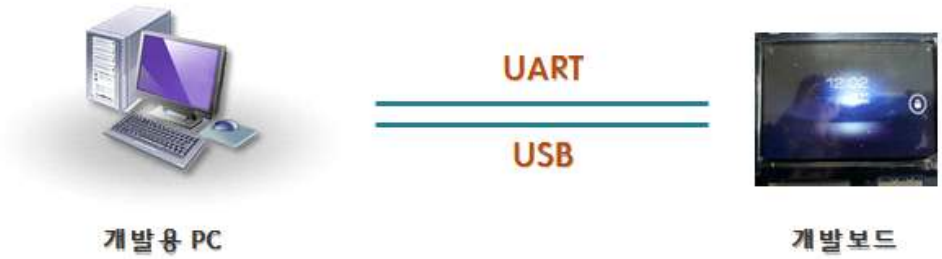
(나) 응용프로그램 개발을 위한 환경설정

① 하드웨어 환경설정

○ 응용프로그램 개발을 위한 하드웨어 환경을 다음과 같이 구성함

○ 개발을 진행할 PC와 개발보드의 인터페이스는 USB와 UART를 사용

○ 윈도우 7(64bit)의 운영체제와 4.0 버전의 Android 운영체제를 기반으로 설치환경 구성



[그림 70] 개발용 PC와 개발보드의 연결도

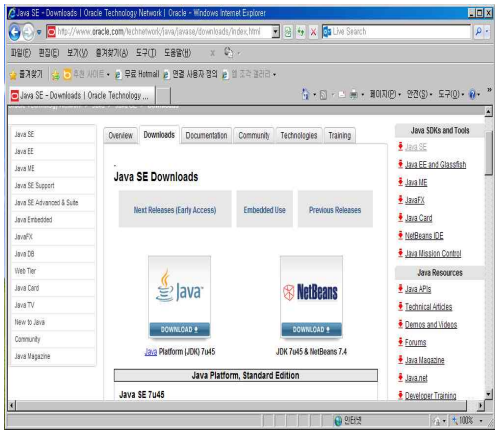
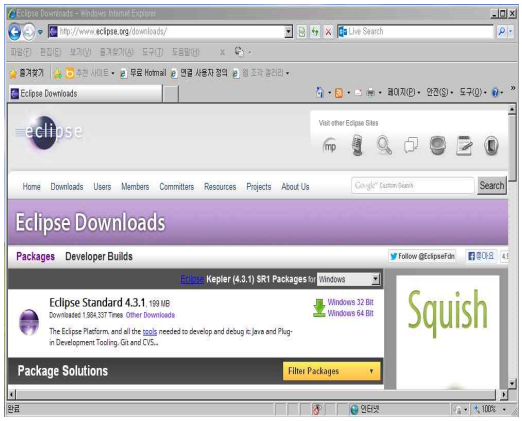
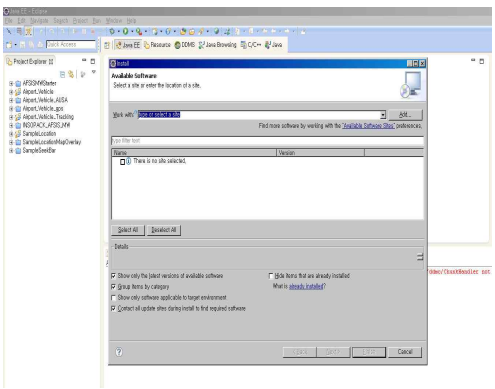
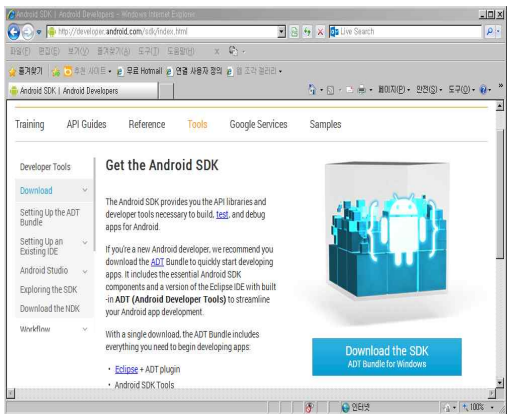
○ 개발용 PC와 개발보드의 하드웨어 및 소프트웨어 사양은 다음과 같음

항목		세부 항목
개발용 PC 환경	운영체제	Windows 7 Professional K Service Pack 1 64Bit
	모델명	HP EliteBook 8570P
	프로세스	Intel(R) Core(TM) i7-3610QM Cpu @ 2.30 GHz
	램	4.00 GB
	개발 툴	Android Developer Tools(Eclipse)
개발보드 환경	모델명	MV4412 - LCD 또는 개발보드
	운영체제	Android 4.0 (API 14)

[표 128] 개발용 장비의 하드웨어 및 소프트웨어 정보

② 소프트웨어 환경설정

○ JAVA 언어 기반의 안드로이드 응용프로그램 개발을 위한 환경설정 방법은 다음과 같음

순번	설치 화면	순번	설치 화면
1	 • 최신 자바 플랫폼 - JAVA SE, JDK 7u45 - http://www.oracle.com	2	 • 이클립스 개발도구 설치 - Eclipse Package, classic 3.7.1 - http://www.eclipse.org/downloads/
3	 • 안드로이드 개발 도구 설치 - ADT Plugin 15.0.0 - http://developer.android.com/sdk/eclipse-adt.html	4	 • 안드로이드 SDK 설치 - SDK Tools r15 - http://developer.android.com/sdk/index.html

[표 129] 안드로이드 관련 개발 툴 설치 절차

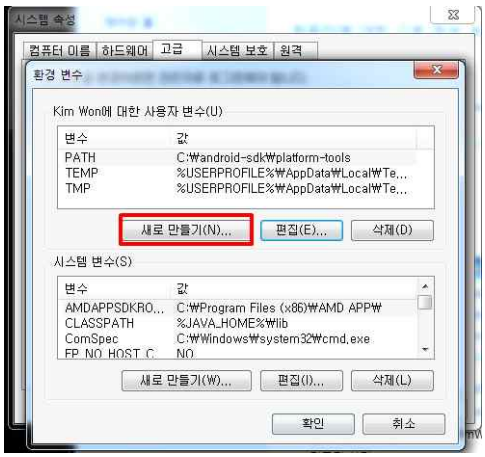
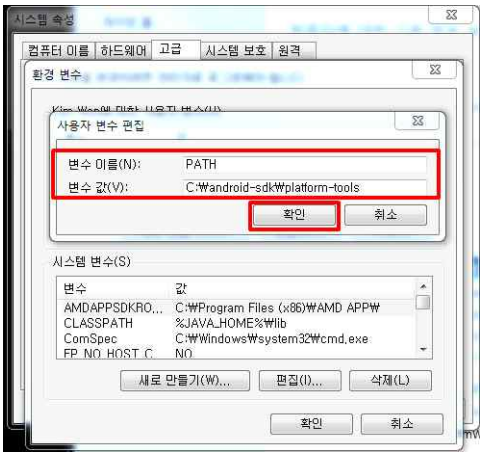
○ 개발보드 사용을 위한 USB 드라이버 설치

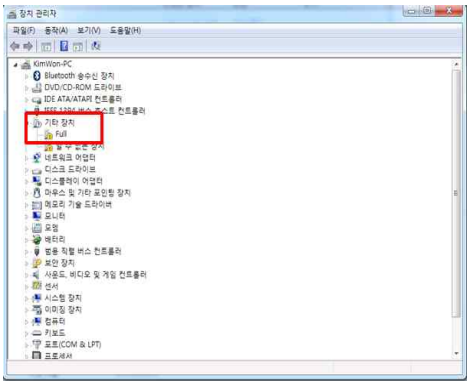
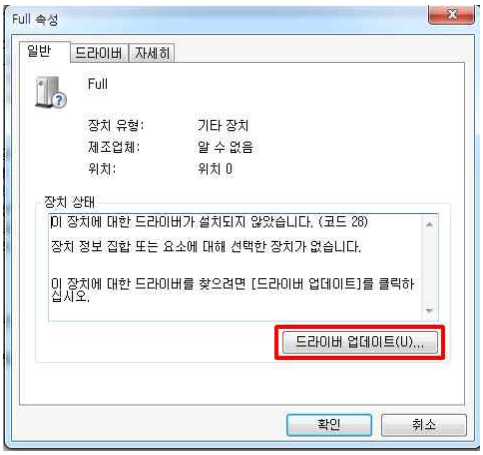
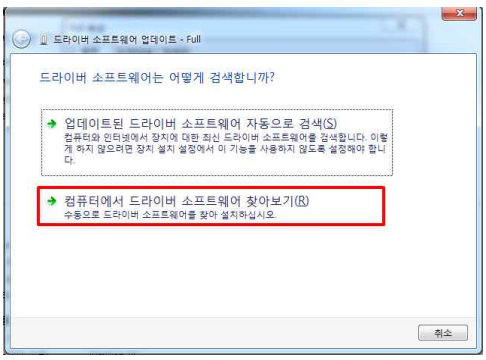
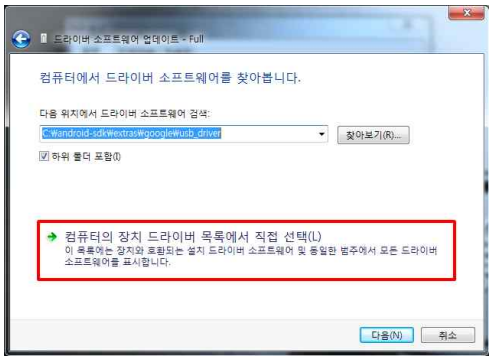
- 개발보드의 USB 드라이버는 Android SDK를 이용하여 설치가 진행됨

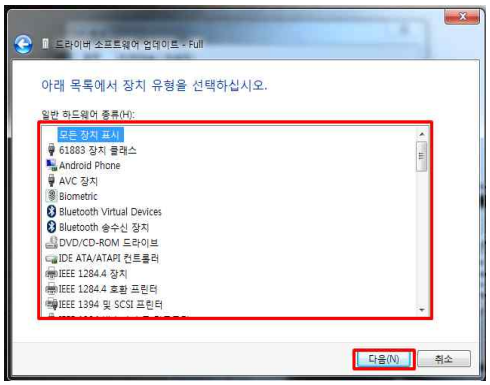
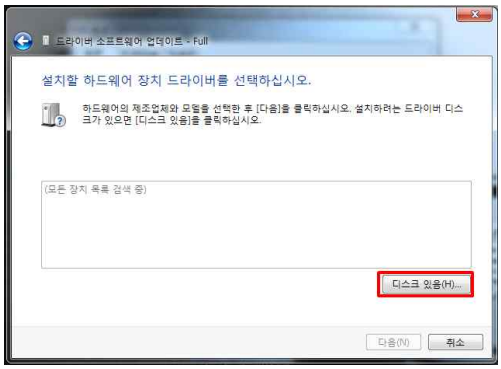
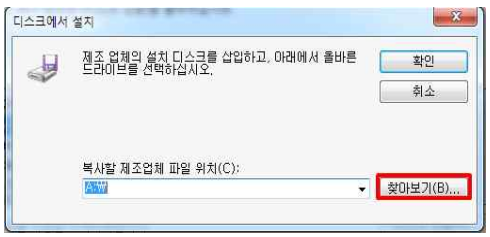
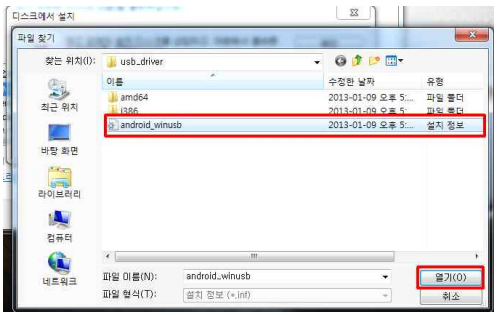
- 기존 파일경로에 위치한 “android-sdk” 폴더를 다음의 변경된 파일경로로 이동시킨 후 USB 설치를 진행(C드라이브에 Android 폴더가 존재하지 않을 경우 생성)

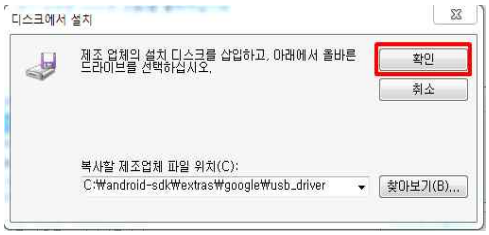
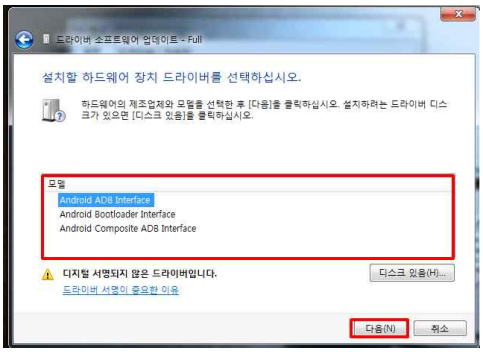
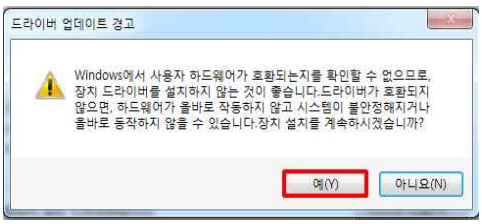
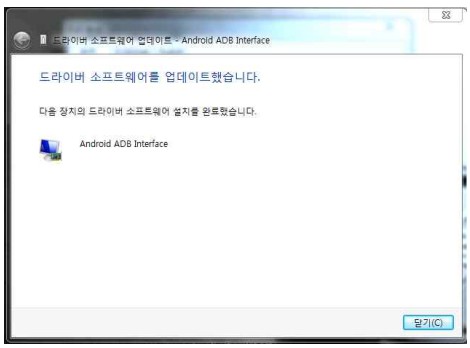
  • 기존 파일경로 : C:\Users\(PC계정)\AppData\Local\Android

  • 변경된 파일경로 : C:\Android

순번	설치 화면	순번	설치 화면
1	 • Android SDK 사용을 위한 환경변수 설정 - 내컴퓨터 마우스 우클릭 후 다음 선택 : 속성 > 고급 > 환경변수 > 새로 만들기	2	 • 다음과 같이 정보 입력 후 확인 버튼 클릭 - 변수이름 : PATH - 변수 값 : C:\Android\android-sdk\platform-tools
3	 • 개발보드를 PC와 연결 해제 후 다음의 경로의 파일을 64bit용 파일로 교체 - 경로 : C:\Android\android-sdk\extras\google\usb_driver	4	 • 윈도우 상태 표시줄에 다음과 같은 팝업이 출력됨을 확인

순번	설치 화면	순번	설치 화면
5	 • 내컴퓨터 마우스 우클릭 > 속성 > 장치 관리자 > 기타 장치 부분에 “Full“ 이라는 장치가 출력되는 것을 확인	6	 • Full 장치 선택 > 속성 > 드라이버 업데이트 버튼 클릭
7	 • 드라이버 업데이트 시 “컴퓨터에서 드라이버 소프트웨어 찾아보기” 클릭	8	 • 컴퓨터의 장치 드라이버 목록에서 직접 선택 클릭

순번	설치 화면	순번	설치 화면
9	 • 모든 장치 표시 선택 후 다음 버튼 클릭	10	 • 디스크 있음 버튼 클릭
11	 • 찾아보기 버튼 클릭	12	 • 다음의 경로로 이동하여 android_winusb 파일 선택 후 열기 버튼 클릭 - 경로 : C:\Android\android-sdk\extras\google\usb_driver

순번	설치 화면	순번	설치 화면
13	 • 파일 위치를 확인한 후 확인 버튼 클릭	14	 • 모델 목록 중 Android ADB Interface 선택 후 다음 버튼 클릭
15	 • 경고문 팝업 시 예 버튼 클릭	16	 • 드라이버 설치완료

[표 130] 개발보드의 USB 드라이버 설치 절차

(다) 응용프로그램의 GUI 및 기능 개발

① 응용프로그램의 GUI 개발

○ GUI 개발 방법 및 절차는 다음의 로그인 화면 구성과 동일함

○ 로그인 화면의 전체 UI를 다음과 같이 구성함



[그림 71] 차량단말 서브시스템 응용프로그램의 로그인 화면

- 로그인 화면의 배경이미지를 출력하기 위하여 xml 파일에 다음의 함수 추가

```
android:orientation="vertical"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:background="@drawable/background"
```

[그림 72] 배경이미지 추가를 위한 함수 사용

- 배경이미지를 풀 스크린으로 적용하기 위하여 해당 액티비티의 클래스 파일에 다음의 함수 추가

```
final Window win = getWindow();
win.setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
        WindowManager.LayoutParams.FLAG_FULLSCREEN);
requestWindowFeature(Window.FEATURE_NO_TITLE);
requestWindowFeature(Window.FEATURE_PROGRESS);

setContentView(R.layout.login);
```

[그림 73] 배경이미지의 풀 스크린 적용을 위한 액티비티 클래스 설정

○ 로그인을 위한 숫자 등 입력 버튼의 UI 구성 및 기능구현

- 숫자 등 입력 버튼의 UI 구성을 위하여 XML 파일에 다음의 함수 추가

```
<Button
    android:id="@+id/btn1"
    android:layout_width="80.0dp"
    android:layout_height="50.0dp"
    android:layout_column="3"
    android:layout_marginLeft="10.0dp"/>

<Button
    android:id="@+id/btn2"
    android:layout_width="80.0dp"
    android:layout_height="50.0dp"
    android:layout_column="4"
    android:layout_marginLeft="8.0dp"/>
```

[그림 74] 입력버튼의 UI 구성을 위한 함수 추가 예시

- 구성된 버튼을 사용하기 위하여 해당 액티비티의 클래스 파일에 리소스 사용 선언과 기능 구현

```
btn_array[BTN_IMAGE_1_TYPE] = (Button) findViewById(R.id.btn1);
btn_array[BTN_IMAGE_2_TYPE] = (Button) findViewById(R.id.btn2);
```

```
btn_array[BTN_IMAGE_1_TYPE].setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
        if (usernameEntry.isFocused()) {
            usernameEntry.append("1");
        } else if (passwordEntry.isFocused()) {
            resetPwEdit("1");
        }
    }
});

btn_array[BTN_IMAGE_2_TYPE].setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
        if (usernameEntry.isFocused()) {
            usernameEntry.append("2");
        } else if (passwordEntry.isFocused()) {
            resetPwEdit("2");
        }
    }
});
```

[그림 75] 구성된 UI의 기능구현 예시

○ 사용자 ID 및 비밀번호 입력을 위한 텍스트 박스 UI 구성 및 기능구현

- 사용자 ID 및 비밀번호 입력을 위하여 XML 파일에 다음의 함수를 추가하여 텍스트 박스 UI를 구성

```
<EditText
    android:id="@+id/usernameEntry"
    android:layout_width="150.0dp"
    android:layout_height="40.0dp"
    android:layout_marginLeft="125.0dp"
    android:layout_marginTop="5dp"
    android:inputType="text"
    android:cursorVisible="true"
    android:maxLength="5" />

<EditText
    android:id="@+id/passwordEntry"
    android:layout_width="150.0dp"
    android:layout_height="40.0dp"
    android:layout_marginLeft="125.0dp"
    android:layout_marginTop="5dp"
    android:imeOptions="actionSearch"
    android:inputType="text"
    android:password="true"
    android:maxLength="16">
<requestFocus />
</EditText>
```

[그림 76] UI 구성을 위한 함수 추가 예시

- 구성된 텍스트 박스를 사용하기 위하여 해당 액티비티의 클래스 파일에 리소스 사용 선언과 기능 구현

```
// 리소스를 사용하여 사용자 ID를 입력할 수 있는 텍스트 박스 객체를 생성한다
usernameEntry = (EditText) findViewById(R.id.usernameEntry);
passwordEntry = (EditText) findViewById(R.id.passwordEntry);

public void onClick(View v) {
    if (usernameEntry.isFocused()) {
        usernameEntry.append("8");
    } else if (passwordEntry.isFocused()) {
        resetPwEdit("8");
    }
}
```

[그림 77] 텍스트 박스 기능구현 예시

② 응용프로그램의 기능 개발

○ 전용단말기 응용프로그램에서 제공되는 기능은 다음과 같음

※ 다음의 기능 외 GIS 엔진 관련 기능 내용은 "(3) GIS를 활용한 운용환경 응용프로그램 개발, (나) GIS 엔진 개발, ⑥ GIS 엔진 관련 기능 개발" 참조

기능	설명
소프트웨어 자동 업데이트	최신 버전의 전용단말기 응용프로그램 다운로드 및 설치
사용자 로그인	전용단말기를 사용하기 위한 사용자 인증 진행
지도조작 (확대/축소/이동)	메인화면에 제공되는 지도화면을 확대/축소/이동할 수 있는 조작 버튼 및 터치 제공
Grid Map 출력상태 설정	공항 내 Grid(격자) 구역을 표시하는 지도 구획의 ON/OFF 상태 설정
GPS 데이터 수신	GPS 데이터를 수신하여 현재 자신의 차량 위치 및 연결 상태 정보 제공
출동지시 정보 수신 알림 제공	사고, FOD, 위반 사항에 대하여 운영센터로부터 출동지시 정보 수신 시 관련 알림 음성 제공
속도, 날씨, 시계, 무선 네트워크 상태	속도, 날씨, 시계, 무선 네트워크, GPS, 경/위도 정보의 현재 상태 제공
시스템 종료	전용단말기 종료

[표 131] 전용단말기 응용프로그램의 기본 제공 기능

㉮ 소프트웨어 자동 업데이트

○ 전용단말기 실행 시 설치되어 있는 소프트웨어의 버전 확인이 진행된 후 최신버전이 아닐 경우 업데이트를 진행

- 소프트웨어 버전 확인을 위하여 통합서버(미들웨어)로 현재 버전 정보를 전송

```
try {
    for(int i=0;i<TransmitPacket.nMaxRetryTime && bDataReceive == BasicInfo.NO_RESP ;i++) {
        if( i % TransmitPacket.nRetryInterval== 0 ){
            bTransmit = TransmitPacket.Transmit_SOFTWARE_VERSION_CONFIRM_REQ_INFO( sCarID,
                                                                                   "V",
                                                                                   TransmitPacket.GetProgramVer());
        }

        Thread.sleep(1000);
    }
    Log.e("RequestVersionToServer", "Terminated thread resp =  " + bDataReceive);
} catch (Exception e) {
    e.printStackTrace();
}
```

[그림 78] 소프트웨어 버전 확인을 위한 패킷 생성 로직(1)

```
if( bDataReceive == BasicInfo.NO_RESP ){
    TransmitPacket.WriteErrorLog("Packet Retry Fail/Transmit_AIRSIDE_VEH_LOGIN_REQ_INFO");
    if( bTransmit == 0 ){
        Message msg = mTransmitHandler.obtainMessage(BasicInfo.CONFIRM_SEND_FAIL);
        mTransmitHandler.sendMessage(msg);
    }else{
        Message msg = mTransmitHandler.obtainMessage(BasicInfo.CONFIRM_NETWORK_ERR);

        mTransmitHandler.sendMessage(msg);
    }
}
```

[그림 79] 소프트웨어 버전 확인을 위한 패킷 생성 로직(2)

- 통합서버(미들웨어)로부터 버전 확인 결과 수신 후 관련 처리 진행

  - 최신버전이 아닐 경우 : 최신버전으로 업데이트를 진행
  - 최신버전일 경우 : 로그인 화면으로 진입

```
@Override
public void onReceive(Context context, Intent intent ){
    final String action = intent.getAction();
    Log.i("TcpReceiver" , "Action : " + action);

    try {

        if( intent != null ){
            String VersionState = intent.getExtras().getString("VersionState");

            Log.i("onReceive" , "VersionState : " + VersionState);

            if( VersionState != null ){
                if( VersionState.equals("Y") ){ // 다운로드필요

                    if( bDataReceive == BasicInfo.NO_RESP ){
                        bDataReceive = BasicInfo.RESP_OK;

                        Message msg = mTransmitHandler.obtainMessage(BasicInfo.CONTINUE_UPGRADE);
                        mTransmitHandler.sendMessage(msg);
                    }

                }else{      // 다운로드불필요
                    if( bDataReceive == BasicInfo.NO_RESP ){
                        bDataReceive = BasicInfo.RESP_NOK;

                        Message msg = mTransmitHandler.obtainMessage(BasicInfo.CONFIRM_NO_UPGRADE);
                        mTransmitHandler.sendMessage(msg);
                    }

                }
            }
        }
```

[그림 80] 버전 확인 결과에 따른 처리 분기

- 소프트웨어 업데이트 필요 시 통합서버(미들웨어)로부터 최신버전의 설치파일 다운로드

```
try {
    Thread.sleep(500); //화면 갱신을 위해.

    socket = new Socket();
    socket.bind(null);
    socket.setSoTimeout(3000);  // 송수신 대기 시간
    socket.connect(new InetSocketAddress(upsvrIP, nUpsvrPort), 3000); // connect 대기 시간

    InputStream = new BufferedInputStream(socket.getInputStream());

    int nof_file = ByteStream.toInt(InputStream);
    if( nof_file <= 0 ){
        Log.e("upgrade_prepare", "error Socket : timeout");
        Message msg = mTransmitHandler.obtainMessage(BasicInfo.CONFIRM_NO_UPGRADE);
        mTransmitHandler.sendMessage(msg);
        return;
    }

    for (int cur_file = 0; cur_file < nof_file; cur_file++)
    {
        String file_name = ByteStream.ToString(InputStream);

        if( file_name == null ){
            Message msg = mTransmitHandler.obtainMessage(BasicInfo.CONFIRM_NO_UPGRADE);
            mTransmitHandler.sendMessage(msg);
            return;
        }

        if( !ByteStream.toFile(InputStream, file_name) ){
            Message msg = mTransmitHandler.obtainMessage(BasicInfo.CONFIRM_NO_UPGRADE);
            mTransmitHandler.sendMessage(msg);
            return;
        }
    }
```

[그림 81] 최신버전의 설치파일 다운로드

- 다운로드가 완료된 설치파일(APK 파일) 실행

```
try
{
    Intent intent = new Intent(Intent.ACTION_VIEW);
    Uri path = Uri.fromFile(new File(configFilePath));
    intent.setDataAndType(path, "application/vnd.android.package-archive");

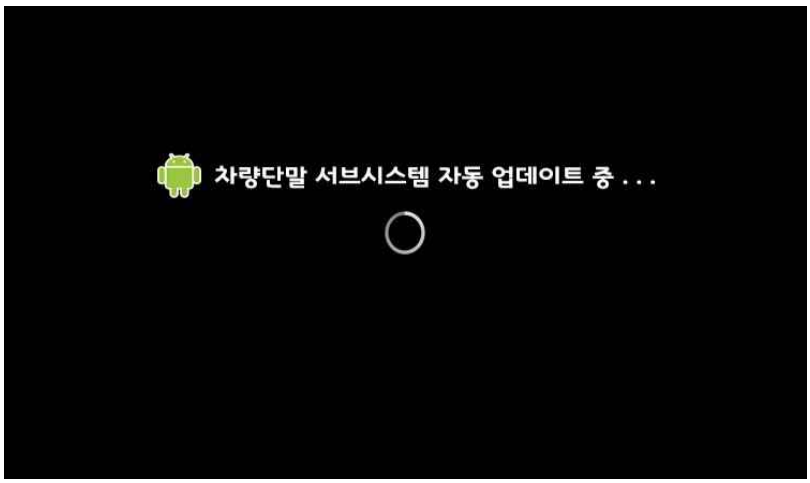
    startActivity(intent);

}
catch (ActivityNotFoundException e)
{
    Log.e("UPgrade", "ActivityNotFoundException", e);
    setResult(BasicInfo.RESP_CODE_NO_UPGRADE);
    finish();
    return;
}

setResult(BasicInfo.RESP_CODE_UPGRADE_EXIT);
finish();
```

[그림 82] 다운로드가 완료된 최신버전의 설치파일 실행

- 소프트웨어 업데이트 진행 시 출력되는 화면은 다음과 같음



[그림 83] 소프트웨어 업데이트 진행 화면

㉰ 사용자 로그인

○ 전용단말기 사용을 위한 사용자 로그인 계정의 승인 절차 진행

- 사용자로부터 입력받은 사용자 ID 및 비밀번호를 통합서버(미들웨어)로 전송

```
for (int i = 0; i < TransmitPacket.nMaxRetryTime
        && bDataReceive == BasicInfo.NO_RESP; i++) {
    if (i % TransmitPacket.nRetryInterval == 0) {
        Log.d("LoginPacket", "interval = " + i);
        bTransmit = TransmitPacket.Transmit_AIRSIDE_VEH_LOGIN_REQ_INFO(
                                            usernameEntry.getText().toString(),
                                            passwordInfo.toString());
    }
    Thread.sleep(1000);
}
```

[그림 84] 사용자 계정정보 확인을 위한 패킷 전송

- 통합서버(미들웨어)에서 수신한 로그인 계정 정보가 유효할 경우 다음의 로그 출력 및 로그인 승인 정보 전송

```
RECV = 53
C0001 1111
동일함
3월 20, 2013 5:12:12 오후 afsis.mw.unit.UnitManager sendMessage
INFO: Processed 2EA in 2278ms
3월 20, 2013 5:12:12 오후 afsis.mw.worker.WorkerManager$Worker run
INFO: SUCCESS : SERVICE_MUSMS-MSG_ULCG
TCP Sent : 172.25.53.19
```

[그림 85] 통합서버(미들웨어)의 로그인 승인 로그

- 통합서버(미들웨어)에서 수신한 로그인 계정 정보가 유효하지 않을 경우 다음의 로그 출력 및 로그인 거부 정보 전송

```
RECV = 53
C0001 1111
동일하지 않음
3월 20, 2013 5:15:57 오후 afsis.mw.unit.UnitManager sendMessage
INFO: Processed 1EA in 225015ms
3월 20, 2013 5:15:57 오후 afsis.mw.worker.WorkerManager$Worker run
INFO: SUCCESS : SERVICE_MUSMS-MSG_ULCG
TCP Sent : 172.25.53.19
```

[그림 86] 통합서버(미들웨어)의 로그인 거부 로그

- 통합서버(미들웨어)로부터 수신한 로그인 승인 결과에 따라 처리

  - 로그인 승인 정보일 경우 : 응용프로그램의 메인화면 출력
  - 로그인 거부 정보일 경우 : 사용자 정보의 재입력 요구 알림창 출력 및 정보 입력란 초기화

```
@Override
public void onReceive(Context context, Intent intent) {
    final String action = intent.getAction();
    Log.i("TcpReceiver", "Action : " + action);

    try {

        if (intent != null) {
            String LoginState = intent.getExtras().getString(
                    "LoginState");

            Log.i("TcpReceiver", "LoginState : " + LoginState);

            if (LoginState != null) {
                if (LoginState.equals("Y")) {

                    if (bDataReceive == BasicInfo.NO_RESP) {
                        bDataReceive = BasicInfo.RESP_OK;

                        Message msg = mAlertThreadHandler
                                .obtainMessage(BasicInfo.CONFIRM_SEND_OK);
                        mAlertThreadHandler.sendMessage(msg);
                    }

                } else {
                    if (bDataReceive == BasicInfo.NO_RESP) {
                        bDataReceive = BasicInfo.RESP_NOK;

                        Message msg = mAlertThreadHandler
                                .obtainMessage(BasicInfo.CONFIRM_LOGIN_FAIL);
                        mAlertThreadHandler.sendMessage(msg);
```

[그림 87] 로그인 승인 요청 결과에 따른 분기 처리

- 다음과 같이 로그인 화면에서 부여박은 사용자 ID와 비밀번호를 입력한 후 계정의 승인 절차를 진행



[그림 88] 전용단말기 사용을 위한 로그인 화면



[그림 89] 로그인 화면의 차량종류 도움말

㉰ 지도 확대 및 축소

○ 지도화면은 총 5단계로 확대 및 축소되어 출력됨

○ GIS 엔진에서 제공하는 인터페이스를 사용하여 메인화면에 출력되는 공항 지도화면의 확대 및 축소 기능을 다음과 같이 구현함

- 메인화면의 지도 확대 버튼 클릭 시 현재 중심을 기준으로 지도화면이 확대됨

```
// 지도 확대 버튼 클릭 시 발생되는 이벤트로 현재 터치 모드를 지도 확대 모드로 설정한다.
btn_mapZoomIn.setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
        nCurrentStatus = CURRENT_STATUS_ZOOM_IN;
        setMapIconEnableMode();
        System.out.println("::::::::::::::: 지도 확대 모드 :::::::::::::::");

        boolean bAvailablePosition = false;
        double x = 0;
        double y = 0;

        if(mapLevel < mMaxLevel)
        {
            zoomDistance = confirmScope(zoomDistance / 1.5);        // 지도확대
            mapLevel++;
        }else{
            return;
        }

        ShowToast(mapLevel);

        // 지도 레벨 정보를 출력한다.
        TextView_Array[TEXTVIEW_COUNT_TYPE].setText(Integer.toString(mapLevel));

        if( mLastLongitude.length() != 0 ){
            x = Double.parseDouble(mLastLongitude);
            y = Double.parseDouble(mLastLatitude);

            if(     !(x < mMin_x_value ||
                    x > mMax_x_value ||
                    y < mMin_y_value ||
                    y > mMax_y_value ) )
            {
                bAvailablePosition = true;
            }
        }
        if( bAvailablePosition && IsMyPosition ){
            currentPos.x = x;
            currentPos.y = y;
            // 내 위치를 중심으로 지도화면이 출력되도록 업데이트 한다.
            mPrevGridInfo = mGridInfo;
            updateMap();
        }else{
            // 현재 좌표를 지도좌표로 변환한다.
            currentPos = getCoordinatePerLevel(mapLevel, m_Control.ScreenToMap(middleX, middleY)
            // 지도상태를 업데이트 한다.(지도조작 결과 반영)
            updateMap();
        }
```

[그림 90] 지도 확대 기능 처리 로직

- 지도확대 기능 수행 결과는 다음과 같음

지도 확대 전	지도 확대 후
	

[표 132] 지도 확대 기능 수행 전, 후 예시

- 메인화면의 지도 축소 버튼 클릭 시 현재 중심을 기준으로 지도화면이 축소됨

```
        // 지도 축소 버튼 클릭 시 발생되는 이벤트로 현재 터치 모드를 지도 축소 모드로 설정한다.
        btn_mapZoomOut.setOnClickListener(new OnClickListener() {
            public void onClick(View v) {
//              nCurrentStatus = CURRENT_STATUS_ZOOM_OUT;
//              setMapIconEnableMode();
//              System.out.println("::::::::::::::: 지도 축소 모드 ::::::::::::::");


                boolean bAvailablePosition = false;
                double x = 0;
                double y = 0;

                if(mapLevel > mMinLevel)
                {
                    zoomDistance = confirmScope(zoomDistance * 1.5);        // 지도축소
                    mapLevel--;
                }else{
                    return;
                }

                ShowToast(mapLevel);

                // 지도 레벨 정보를 출력한다.
                TextVew_Array[TEXTVIEW_COUNT_TYPE].setText(Integer.toString(mapLevel));

                if( !( mapLevel == mMinLevel && mPrevLevel == mapLevel) ){

                    if( mLastLongitude.length() != 0 ){
                        x = Double.parseDouble(mLastLongitude);
                        y = Double.parseDouble(mLastLatitude);

                        if( !(x < mMin_x_value ||
                            x > mMax_x_value ||
                            y < mMin_y_value ||
                            y > mMax_y_value) )
                        {
                            bAvailablePosition = true;
                        }
                    }
                }

                if( bAvailablePosition && IsMyPosition ){
                    currentPos.x = x;
                    currentPos.y = y;
                    // 내 위치를 중심으로 지도화면이 출력되도록 업데이트 한다.
                    mPrevGridInfo = mGridInfo;
                    updateMap();
                }else{
                    // 현재 좌표를 지도좌표로 변환한다.
                    currentPos = getCoordinatePerLevel(mapLevel, m_Control.ScreenToMap(middleX, middleY));
                    // 지도상태를 업데이트 한다.(지도조작 결과 반영)
                    updateMap();
                }


            }
        });
```

[그림 91] 지도 축소 기능 처리 로직

- 지도축소 기능 수행 결과는 다음과 같음

지도 축소 전	지도 축소 후
	

[표 133] 지도 축소 기능 수행 전, 후 예시

㉮ Grid Map 출력상태(On/Off) 설정

○ Grid Map은 공항지역을 격자로 나누어 표현한 지도로 GIS 엔진에서 Grid Map 레이어를 제공하여 출력(On) 또는 제거(Off) 설정이 가능

○ GIS 엔진에서 제공하는 인터페이스를 사용하여 Grid Map 레이어의 출력상태(On/Off)를 설정하는 기능을 다음과 같이 구현함

- 메인메뉴 창에서 Grid Map 버튼 클릭 시 현재 지도화면에 Grid Map 출력 또는 제거

```
// 추가내용 : 기존 메인화면이 다시 표시될 때의 지도상태를
//              Grid Map ON/OFF 상태에 따라 분기하여 설정한다.
if(linear_gridmap != null)
{
    if (gridMapOnOffStatus != STATUS_ON)
    {
        // Grid Map 레이어를 지도화면에 출력하지 않는다.
        linear_gridmap.setVisibility(View.INVISIBLE);
    }
    else
    {
        // Grid Map 레이어를 지도화면에 출력한다.
        linear_gridmap.setVisibility(View.VISIBLE);
    }
}
```

[그림 92] Grid Map 출력상태(On/Off) 설정

- Grid Map 출력상태(On/Off) 설정 기능 수행 결과는 다음과 같음

Grid Map 출력(On)	Grid Map 제거(Off)
	

[표 134] Grid Map 출력(On) 또는 제거(Off) 화면

㈐ GPS 데이터 수신

○ GPS 데이터를 수신하여 현재 내 차량의 위치정보 사용이 가능함

- GPS 데이터 수신을 위한 모듈 사용 로직 구현

```
// GPS NMEA Read service
if( isGpsBindService ){
    mServiceConnection = new ServiceConnection() {
        @Override
        public void onServiceConnected(ComponentName arg0, IBinder arg1) {
            // TODO Auto-generated method stub
            // IBinder타입을 PlayService.MyBinder타입으로 type-casting하고
            // getService로 서비스 가져온다.
            //3. Binding이 완료되면 Service 정보를 가져오고
            pGpsService = ((gpsService.MyBinder)arg1).getService();
        }

        @Override
        public void onServiceDisconnected(ComponentName arg0) {
            // TODO Auto-generated method stub
            pGpsService = null;
        }
    };

    bindService(new Intent(this,gpsService.class), mServiceConnection, Context.BIND_AUTO_CREATE);
}
```

[그림 93] GPS 모듈 사용을 위한 객체 생성 및 설정

- GPS 데이터의 수신 주기 설정 및 파싱 결과 출력

```
if( pGpsService != null && isGpsBindService ){

    com.project.airportvehicle.main.gpsService.gpsInfo mData = pGpsService.getGpsInfo();

    if( mData.satelliteCnt == 0 && mData.maxsatelliteCnt == 0 ){    // GPS 를 놓친경우...
        mGpsLossCnt++;
        if( mGpsLossCnt > 5 ){  // GPS 를 5초 이상 놓친경우 reset하며 이하에서는 이전 데이타를 가진다.
                                // 이경우 위/경도는 놓치기전의 데이터를 갖는다.
//          mLongitude = mData.longitude;
//          mLatitude = mData.latitude;
            mSpeed= mData.speed;
            nSatelliteCount = mData.satelliteCnt;
            nMaxSatelliteCont = mData.maxsatelliteCnt;
        }
//      calendar.setTimeInMillis(mData.timeValue);
    }else{
        mGpsLossCnt = 0;
        mLongitude = mData.longitude;
        mLatitude = mData.latitude;
        mSpeed= mData.speed;
        nSatelliteCount = mData.satelliteCnt;
        nMaxSatelliteCont = mData.maxsatelliteCnt;
//      calendar.setTimeInMillis(mData.timeValue);
    }

}
```

[그림 94] GPS 데이터 수신 주기 설정 및 파싱 결과 출력

- 수신된 GPS 데이터로부터 다음과 같이 하단 바 좌측에 위도 및 경도 정보를 출력하며 지도화면에 내 차량의 현재 위치를 표시함



[그림 95] GPS 데이터 수신 정보 출력(위도 및 경도)

㉦ 출동지시 정보 수신 시 음성알림 제공

○ 운영센터로부터 사고, FOD, 위반 관련 현장 출동에 대한 정보를 수신하였을 경우 사용자에게 음성 알림을 제공함

```
private void playViolationSound(int vioType)
{
   // 다른 위반 음성이 출력 중이지 않을 경우 수행된다.
   if(!violationSound.isPlaying())
   {
      violationSound.release();
      int soundKind = 0;

      switch (vioType)
      {

         ...

         case 7: // 출동지시 음성
            soundKind = R.raw.runtothere;
            break;
      }

      violationSound = MediaPlayer.create(this, soundKind);
      violationSound.setLooping(false);
      violationSound.start();
      }
}
```

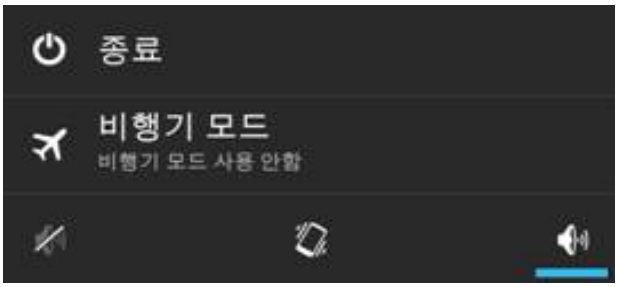
[그림 96] 출동지시 정보 수신 처리 로직(일부)

㉮ 시스템 종료

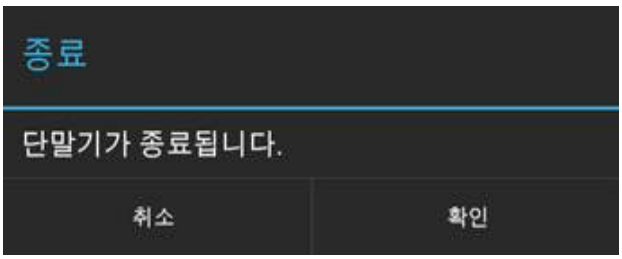
○ 응용프로그램을 수동으로 종료시키기 위한 기능으로 전용단말기 우측 전원 버튼을 사용함

- 전용단말기 우측의 전원 버튼을 길게 누를 경우 차량단말 서브시스템 응용프로그램 및 단말 종료

- 시스템 종료 시 사용자에게 다음의 정보창을 제공하여 선택 여부를 확인



[그림 97] 시스템 종료 여부 확인(1)



[그림 98] 시스템 종료 여부 확인(2)

(3) GIS를 활용한 운용환경 응용프로그램 개발

(가) GIS 엔진 개발을 위한 환경설정

※ 상세 내용은 "(2) GUI 개발 및 기능 구현, (나) 응용프로그램 개발을 위한 환경설정" 참조

(나) GIS 엔진 개발

① Raster 데이터(항공사진) 처리

○ 항공사진을 출력하기 위하여 원본 이미지 파일을 Raster 형식의 파일로 변환하여 사용함

○ Raster 란 GIS의 대표적인 자료구조 중 하나로 2차원 표면을 일정한 격자모양의 셀로 나누어 데이터의 위치와 값을 표현함

○ 공항의 항공사진 파일을 Raster 형식의 파일로 변환한 CMG 파일을 안드로이드에서 처리할 수 있도록 관련 로직을 다음과 같이 구현함



[그림 99] 항공사진 출력을 위한 Raster 데이터(CMG 파일) 처리 흐름

- 항공사진 원본 파일(TIFF 파일)을 좌표정보가 포함된 Raster 데이터(CMG 파일)로 변환
- 변환된 Raster 데이터로부터 버전 정보, 전체 이미지 크기, 타일 이미지 크기, 이미지 레벨 개수, 데이터 시작 offset 등의 정보 로딩
- 각 레벨의 이미지 좌표정보, 크기 등 저장
- 각 이미지 스케일의 셀간 거리 계산 후 저장
- 현재 스케일의 이미지 행과 열의 개수 산출 후 항공사진 출력

```
    // 1. CMG 파일 출력을 위한 영역(Boundary)을 설정한다.
    double Xmin = 126.7383;                 // left(West)
    double Ymin = 37.58683;                 // top(North)
    double Xmax = 126.850163888888;         // right(East)
    double Ymax = 37.5234022222223;         // bottom(South)

    cmgBoundary = new Envelope(Xmin, Xmax, Ymin, Ymax);

    // 2. CMG 파일(Binary File)을 읽기 위하여 스트림 객체를 생성한다.
    m_CMGReader = new DataInputStream(cmgFileStream);

    // 3. CMG 파일 버전 등의 헤더에 포함되어 있는 정보를 읽어온다.
    byte[] buffer = new byte[9];
    m_CMGReader.read(buffer, 0, 9);

    m_Header.version = new String(buffer, 0, 9).trim();                     // CMG 파일 버전 정보를 저장한다.
    m_Header.width = readBinaryAsCShapeReadInt32(m_CMGReader);              // 전체 이미지의 넓이를 저장한다.
    m_Header.height = readBinaryAsCShapeReadInt32(m_CMGReader);             // 전체 이미지의 높이를 저장한다.
    m_Header.tilesize = readBinaryAsCShapeReadInt32(m_CMGReader);           // 타일 이미지의 크기를 저장한다.
    m_Header.levelcount = readBinaryAsCShapeReadInt32(m_CMGReader);         // 이미지 레벨 개수를 저장한다.
    m_Header.indexoffset = readBinaryAsCShapeReadInt64(m_CMGReader);        // 데이터 시작 offset을 저장한다.

// 4. offset 만큼 byte를 제외한다. (offset 위치부터 시작되므로 -1, 0레벨의 정보가 출력되지 않아 32비트 앞으로 이동함)
m_CMGReader.skip(m_Header.indexoffset - 33);

// 5. 파일로부터 정보를 읽는다.
for (int i = 0; i < m_Header.levelcount; i++)
{
    CMGIndex index = new CMGIndex();
    indexcount = readBinaryAsCShapeReadInt32(m_CMGReader);
    index.level = readBinaryAsCShapeReadInt32(m_CMGReader);                 // CMG 레벨을 저장한다.
    index.xcount = readBinaryAsCShapeReadInt32(m_CMGReader);                // x축으로의 갯수를 저장한다.
    index.ycount = readBinaryAsCShapeReadInt32(m_CMGReader);                // y축으로의 갯수를 저장한다.

    for (int j = 0; j < indexcount; j++)
    {
        CMGIndexItem item = new CMGIndexItem();
        item.offset = readBinaryAsCShapeReadInt64(m_CMGReader);             // 데이터 위치값을 저장한다.
        item.length = readBinaryAsCShapeReadInt32(m_CMGReader);             // 이미지 바이트 길이를 저장한다.
        index.items.add(item);
    }

    m_IndexList.add(index);
}

// 6. 스케일 및 한 셀의 거리를 계산한다.
double xscale = this.cmgBoundary.getWidth() / m_Header.width;
m_IndexList.get(m_IndexList.size() - 1).cellDist = xscale * 128;

for (int i = m_IndexList.size() - 2; i >= 0; i--)
    m_IndexList.get(i).cellDist = m_IndexList.get(i + 1).cellDist * 2;
```

[그림 100] Raster 데이터(CMG 파일) 처리 로직(일부)

③ Vector 데이터(레이어) 처리

○ 항공사진 위에 내/외부도로, 기동지역, 계류장 번호, 유도로 등의 레이어를 출력하기 위하여 SHP, SHX, DBF 파일 등의 Vector 데이터를 사용함

○ Vector 란 GIS의 대표적인 자료구조 중 하나로 공간요소를 점, 선, 면 객체로 정의하여 표현함

○ SHP 파일 등의 Vector 데이터를 안드로이드에서 처리할 수 있도록 관련 로직을 다음과 같이 구현함



[그림 101] 레이어 출력을 위한 Vector 데이터(SHP 파일)의 처리 흐름

- 도형정보를 저장하고 있는 SHP 파일과 도형의 인덱스 정보가 저장되어 있는 SHX 파일 로딩(속성정보가 요구되는 레이어는 DBF 파일도 로딩)

- SHP 파일을 구성하는 도형정보의 Row를 메모리에 로딩

- Geometry 생성을 위한 컨트롤 객체 설정

- Shape type에 따른 Geometry 생성

- 레이어의 원본 좌표 저장 후 화면좌표로 변환

- 변환된 화면좌표를 기반으로 레이어 드로잉

```
RowSetSHP rowSet = access.getRowSet();
int rowCount = rowSet.getRowCount();

for (int fid = 0; fid < rowCount; ++fid)
{
   RowSHP row = rowSet.getRow(fid + 1);      // FID는 1부터 시작한다.

   if (row.load())
   {
      Geometry geom = row.getGeometry();
      drawHandle.setGeometry_original(geom);

      Coordinate[] area =  geom.getBoundary().getCoordinates();
      Coordinate[] temp = new Coordinate[area.length];

      for (int i = 0; i < area.length; i++)
      {
         temp[i] = m_Control.MapToScreen(area[i].x, area[i].y);
      }

      // Outline 좌표를 화면좌표로 변환한 후 리스트 형태로 저장한다.
      drawHandle.setCoordList_current(temp);
      row.unload();
   }
}
```

[그림 102] SHP 및 SHX 파일 처리 로직(일부)

```
for (int i = 0; i < rowCountDBF; ++i)
{
   RowDBF row = rowSetDBF.getRow(i + 1);

   if (row.load())
   {
      // 내/외부도로의 제한속도와 최단거리 계산에 필요한 속성정보를 저장한다.
      if(fileName.contains("Network"))
      {
         // FIRESERVIC : 첫번째 필드, FT_MINUTES  : 네번째 필드, TF_MINUTES    : 다섯번째 필드
         String[] temp = new String[]{row.getValueAsString(0), row.getValueAsString(3), row.getValueAsString(4)};
         shpLayerProperty.get(LAYER_NETWORK).getDrawHandle().setNetworkPathAttribute(temp);
         shpLayerProperty.get(LAYER_NETWORK).getDrawHandle().setLimitSpeedList(row.getValueAsString(6));
      }

      if(fileName.contains("Grid") || fileName.contains("Network") || fileName.contains("Sign"))
      {
         // GRIDMAP_ID 컬럼(필드)의 값, 내/외부도로의 제한속도, 신호위반 지역번호 정보를 저장한다.
         tempList.add(row.getValueAsString(filedIndex));
      }

      row.unload();
   }
}
```

[그림 103] DBF 파일(속성정보) 처리 로직(일부)

④ 안드로이드에서의 지도화면 출력 처리

○ 안드로이드에서 항공사진 및 각종 레이어의 출력 처리는 다음과 같이 레이아웃 인플레이션(Layout Inflation) 기능을 사용함

○ 레이아웃 인플레이션은 XML 형태로 생성한 레이아웃을 안드로이드 액티비티(Activity)에 뷰(View) 형태로 출력할 수 있도록 함

○ 항공사진이 출력되는 메인 액티비티에 레이어 데이터를 출력하기 위하여 계층구조로 뷰 객체를 구성함

```
// 항공사진 출력 후 view 위에 인플레이션으로 레이아웃을 중첩시킨다.
//  1) 시스템 서비스 객체로 인플레이터 객체를 가져온다.
LayoutInflater inflater = (LayoutInflater) getSystemService(Context.LAYOUT_INFLATER_SERVICE);

//  2) 중첩할 레이아웃을 읽어온다.(SHP 데이터용 레이어)
LinearLayout linear_airside = (LinearLayout) inflater.inflate(R.layout.view_airside, null);          // 기동지역
LinearLayout linear_deiceProhibit = (LinearLayout) inflater.inflate(R.layout.view_nosnow, null); // 제방빙, 제설작업 금지구역
LinearLayout linear_network = (LinearLayout) inflater.inflate(R.layout.view_network, null);      // 소방구조경로
linear_apron = (LinearLayout) inflater.inflate(R.layout.view_apron, null);                       // 주기장
linear_gridmap = (LinearLayout) inflater.inflate(R.layout.view_gridmap, null);                   // Grid Map
linear_vehiclePos = (LinearLayout) inflater.inflate(R.layout.view_vehicle, null);                // 타 이동체
linear_myPos = (LinearLayout) inflater.inflate(R.layout.view_mypos, null);                       // 내 차량 위치
linear_airplanePos = (LinearLayout) inflater.inflate(R.layout.view_airplane, null);              // 항공기

// SHP 데이터용 레이어 중첩 시 사용할 파라미터를 설정한다.
LinearLayout.LayoutParams param = new LinearLayout.LayoutParams(LinearLayout.LayoutParams.MATCH_PARENT,
                                                                LinearLayout.LayoutParams.MATCH_PARENT);

// 3) 기존의 항공사진이 출력되었던 레이아웃(RenderView)에 새로운 레이아웃를 중첩시킨다.
window.addContentView(linear_airside, param);
window.addContentView(linear_network, param);
window.addContentView(linear_apron, param);
window.addContentView(linear_gridmap, param);
window.addContentView(linear_vehiclePos, param);
window.addContentView(linear_airplanePos, param);
window.addContentView(linear_myPos, param);
```

[그림 104] 레이아웃 인플레이션을 이용한 레이어 출력용 뷰(View) 설정

```
drawHandle.setDrawLayerType("Airside");
drawHandle.initDrawStatus(Color.argb(255, 152, 0, 0), false, -1);
viewID = R.id.airside_linearLayout;

Paint outLine = new Paint();
outLine.setColor(Color.YELLOW);
drawHandle.moveAreaOutLine = outLine;

LinearLayout layout = (LinearLayout) findViewById(viewID);
layout.addView(drawHandle);
```

[그림 105] 레이어 출력 설정 예시

○ CMG 파일(항공사진)과 SHP 파일(레이어) 등의 공간 데이터를 기반으로 안드로이드에서 제공하는 그래픽을 사용하여 지도 출력 로직을 구현함

```
// 지도가 업데이트되어 기존 설정된 초기 좌상좌표와 같지 않을 경우 수행된다.
if((initPosX != accessCMG.getDrawPoint().x) || (initPosY != accessCMG.getOffsetY()))
{
   setInitPos();
}

for (int i = 0; i < accessCMG.getRowDrawingCount(); i++)                    // 행 개수만큼 수행한다.
{
   for (int j = 0; j < accessCMG.getColumnDrawingCount(); j++)              // 열 개수만큼 수행한다.
   {
      canvas.drawBitmap(bitmapList.get(var), drawPosX, drawPosY, null);     // 각 좌표에 해당하는 비트맵 이미지를 출력한다.

      drawPosX += offsetX;     // 각 행에 열 개수만큼 비트맵 이미지를 출력하기 위하여 X좌표를 증가시킨다.
      var++;                   // 비트맵 카운트를 증가시킨다.
   }

   drawPosX = initPosX;        // 다음 행을 출력하기 위하여 X좌표를 초기화 시킨다.
   drawPosY += offsetY;        // 다음 행을 출력하기 위하여 Y좌표를 증가 시킨다.
}

var = 0;                       // 비트맵 카운터를 초기화 한다.

drawPosX = initPosX;           // X좌표를 초기화 한다.
drawPosY = initPosY;           // Y좌표를 초기화 한다.
```

[그림 106] 항공사진(또는 일반사진) 출력을 위한 비트맵 처리 로직(일부)

```
// 레이어별 drawing 할 RECORD NUMBER(Point 묶음 개수)만큼 for 문을 수행한다.
for (int i = 0; i < coordList_current.size(); i++)
{
   Coordinate[] temp = coordList_current.get(i);

   // RECORD NUMBER 별 Point 개수만큼 라인을 drawing 한다.
   for (int j = 0; j < temp.length; j++)
   {
      startPos = j;

      if (j == (temp.length - 1))
      {
         if (drawLayerName.contains("Network"))
            endPos = j;
         else
            endPos = 0;
      }
      else
         endPos = j + 1;

      canvas.drawLine((float) temp[startPos].x,(float) temp[startPos].y, // drawing 시작점을 설정한다.
                      (float) temp[endPos].x, (float) temp[endPos].y,  // drawing 종료점을 설정한다.
                      outLine);                                        // paint 객체를 설정한다.
   }

   if (drawLayerName.contains("Grid"))       // Grid Map 의 구역별 레이블을 출력한다.
   {
      Coordinate gridIDPos = gridIDPosList_current.get(i);
      canvas.drawText(gridIDList.get(i), (float) gridIDPos.x, (float) gridIDPos.y, label);
   }
}
```

[그림 107] 레이어 출력을 위한 처리 로직(일부)

- 지도화면은 항공사진과 일반사진 두 가지 형태로 제공되며 일반사진이 기본으로 설정되어 있으므로 사용자 편의에 따라 변경이 가능함

- 항공사진이 배경으로 출력되면 유도로, 기동지역, 공항 바운더리, 계류장 번호, 제방빙 지역, 구조소방 도로 등의 레이어가 출력됨



[그림 108] 항공사진(좌) 및 일반사진(우) 출력화면

⑤ GIS 기능 제공을 위한 인터페이스 개발

○ 지도를 확대, 축소, 이동하기 위하여 터치로 입력되는 화면좌표를 지도좌표로 변환하는 인터페이스를 구현함

```
public final Coordinate ScreenToMap(Coordinate pt) {

   if (m_Map != null)
   {
      Coordinate ret = new Coordinate(m_InverseMatrix.M11 * pt.x +
                                      m_InverseMatrix.M21 * pt.y +
                                      m_InverseMatrix.DX,
                                      m_InverseMatrix.M21 * pt.x +
                                      m_InverseMatrix.M22 * pt.y +
                                      m_InverseMatrix.DY);

      return ret;
   }

   return null;
}
```

[그림 109] 화면좌표를 지도좌표로 변환하는 기능함수(일부)

○ 이동체(항공기, 차량) 또는 레이어 등을 화면에 표시하기 위하여 지도좌표를 화면좌표 변환하는 인터페이스를 구현함

```
public final Coordinate MapToScreen(Coordinate pt) {
   if (m_Map != null)
   {
      Coordinate ret = new Coordinate(m_Map.getTransformMatrix().M11 * pt.x +
                                      m_Map.getTransformMatrix().M21 * pt.y+
                                      m_Map.getTransformMatrix().DX,
                                      m_Map.getTransformMatrix().M21 * pt.x +
                                      m_Map.getTransformMatrix().M22 * pt.y +
                                      m_Map.getTransformMatrix().DY);
      return ret;
   }

   return null;
}
```

[그림 110] 지도좌표를 화면좌표로 변환하는 기능함수(일부)

○ 지도 확대 등의 결과 출력을 위하여 지도화면의 업데이트 관련 인터페이스를 구현함

```
// 전용단말기 화면의 해상도를 지도좌표로 변환한 우상, 좌하값이 설정된다.
if (m_Control.SetMapState(currentPos.x, currentPos.y, zoomDistance, 0))
    accessCMG.setScreenBoundary(m_Control.getMap().getBoundary());

bitmapList = accessCMG.BitmapListForDrawing(getSxMax(), getSyMax(), m_Control, mapLevel);
drawPoint = accessCMG.getDrawPoint();

if(shpLayerProperty.size() != 0)
{
   for(int i = 0; i < shpLayerProperty.size(); i++)
   {
      // 1. 백업해 놓은 원본 데이터(SHP 파일의 지도좌표)를 가져온다.
      ArrayList<Geometry> originalData = shpLayerProperty.get(i).getDrawHandle().getGeometry_original();
      String layerType = shpLayerProperty.get(i).getDrawHandle().getDrawLayerType();

      // 2. 기존 drawing 시 사용되었던 좌표정보를 제거한다.
      shpLayerProperty.get(i).getDrawHandle().clearArrayList();

      // 3. 해당 레이어의 좌표정보를 설정한다.
      for(int j = 0; j < originalData.size(); j++)
      {
         Coordinate[] area = null;
         area = originalData.get(j).getCoordinates();

         Coordinate[] temp = new Coordinate[area.length];

         for (int k = 0; k < area.length; k++)
            temp[k] = m_Control.MapToScreen(area[k].x, area[k].y);

         shpLayerProperty.get(i).getDrawHandle().setCoordList_current(temp);
      }
   }
}
```

[그림 111] 지도화면 업데이트 기능함수(일부)

○ 다익스트라 알고리즘을 구현하여 최적경로 및 거리를 산출 기능을 구현함

```
public final ArrayList<Integer> Run(int startID, int endID)
{
   double[][] G = new double[Vertices.size()][Vertices.size()];

   for (int i = 0; i < Vertices.size(); i++)
      for (int j = 0; j < Vertices.size(); j++)
         G[i][j] = Double.POSITIVE_INFINITY;}

   // 노드를 기반으로 가중치를 계산한다.
   for (Edge edge : Edges)
      G[edge.getfrom()][edge.getto()] = edge.getweight();

   // 최적 경로를 산출한다.
   ArrayList<Integer> result = new ArrayList<Integer>();

   DijkstraAlgorithm dijk = new DijkstraAlgorithm(G, startID, endID, result);
   result = dijk.returnValue;

   return result;
}
```

[그림 112] 최적경로 산출을 위한 기능함수(일부)

⑥ GIS 엔진 관련 기능 개발

○ 전용단말기 응용프로그램에서 요구되는 GIS 엔진 관련 기능은 다음과 같음

기능	설명
지도화면 업데이트	터치, 멀티터치 동작을 통한 지도화면 업데이트
최적경로 및 거리 제공	공항 내 사고 발생 시 차량의 현재 위치로부터 사고 발생 지점까지의 최적경로 및 거리 제공
FOD, 사고 및 제방빙 승인요청 등록 처리	공항 내 발생된 FOD 또는 사고, 제방빙 조업구역에 대한 정보 전송을 위하여 해당 위치정보의 획득 및 표시
이동체 출력	지도화면에 공항 내 존재하는 항공기 및 차량의 위치 출력
위반 알림	운행 중인 차량의 과속, 항공기와의 근접 등의 위반사항에 대한 화면 표시 및 음성 알림 제공
내 차량 위치 출력	지도화면에 내 차량의 현재 위치 출력
항공 또는 일반 사진으로 변경	메인화면에 출력되는 지도화면을 항공 또는 일반 사진으로 변경
Grid 구역 출력	현재 내 차량이 위치한 Grid(격자) 구역 정보 출력
활주로 레이블 출력	2본으로 구성되어 있는 김포공항 활주로의 레이블 제공

[표 135] 전용단말기 응용프로그램에서 요구되는 GIS 엔진 관련 기능

○ 지도화면 업데이트

- 터치 또는 멀티터치를 통한 지도 확대, 축소, 이동 시 지도화면에 출력되어 있는 항공사진 및 레이어의 상태를 업데이트함

```
// 지도조작 모드에 따라 확대거리(zoomDistance)에 따른 확대/축소 상태를 설정한다.
if(nCurrentStatus == CURRENT_STATUS_ZOOM_IN)
{
   zoomDistance = confirmScope(zoomDistance / 1.5);     // 지도확대
   if(mapLevel < mMaxLevel)
      mapLevel++;
}

if(nCurrentStatus == CURRENT_STATUS_ZOOM_OUT)
{
   zoomDistance = confirmScope(zoomDistance * 1.5);     // 지도축소
   if(mapLevel > mMinLevel)
      mapLevel--;
}

// 현재 좌표를 지도좌표로 변환한다.
currentPos = getCoordinatePerLevel(mapLevel, m_Control.ScreenToMap(movX, movY));

// 지도상태를 업데이트 한다.(지도조작 결과 반영)
updateMap();
```

[그림 113] 지도화면 업데이트 처리 로직 예시



[그림 114] 지도 이동 시 화면 업데이트 예시

○ 최적경로 및 거리 제공

- 내/외부 도로의 Geometry 정보를 사용하여 노드와 노드간 거리의 비교, 합산 처리를 통하여 최적경로를 탐색하고 거리를 산출함

```
private int nodeSearch(Geometry geo)
{
   int retValue = -1;
   double dDistance = (0.001 * 1) / 111;

   // 저장되어 있는 Node의 좌표정보를 사용하여 해당 위치에서 1m 범위내 위치한 Node를 찾는다.
   for (int i = 0; i < lstNode.size(); i++)
   {
      // Node의 좌표정보에 대한 Geometry를 생성한다.
      Coordinate temp = new Coordinate(lstNode.get(i).X, lstNode.get(i).Y);
      Geometry madeGeo = new GeometryFactory().createPoint(temp);

      // 해당 위치로 부터 1m 범위내에 Node가 존재하는지 검사한다.
      if (geo.isWithinDistance(madeGeo, dDistance))
      {
         retValue = i;
         break;
      }
   }

   return retValue;
}
```

[그림 115] 최적경로 및 거리 산출 처리 로직(일부)



[그림 116] 최적경로 및 거리 산출 결과화면 예시

○ FOD, 사고 및 제방빙 승인요청 등록 처리

- FOD 또는 사고가 발생된 지점과 제방빙 조업을 진행할 구역을 지도화면에서 선택하여 지도좌표로 변환함

```
// 사고지점을 선택하여 정보를 등록한다.
if( nCurrentStatus == CURRENT_STATUS_ACC )
{
   // 사고지점 등록 시 선택된 지점의 화면좌표를 지도좌표로 변환한다.
   Coordinate temp = m_Control.ScreenToMap((float)event.getX(), (float)event.getY());

   // 선택된 지점이 공항 내에 포함되어 있는지 검사한다.
   if(airportBoundaryInOut.contains((float) temp.x, (float) temp.y))
   {
      String Longitude = String.format("%.8f", temp.x);   // 경도를 소수점 8자리까지 출력한다.
      String  Latitude = String.format("%.8f", temp.y);   // 위도를 소수점 8자리까지 출력한다.

      Intent AccIntent = new Intent(getBaseContext(), RescueActivity.class);
      AccIntent.putExtra("CarID", Veh_Login_ID);
      AccIntent.putExtra("Longitude", Longitude);
      AccIntent.putExtra("Latitude", Latitude);

      startActivityForResult(AccIntent, BasicInfo.REQUEST_CODE_ACC);
   }
   else
   {
      bToastMessage = true;
      if( toast != null ) toast.cancel();
      toast = Toast.makeText(getBaseContext(),
                             "사고지점 선택은 공항 내부만 가능합니다! 다시 선택해 주세요.",
                             Toast.LENGTH_SHORT);
      toast.show();
   }
}
```

[그림 117] 위치정보가 포함된 사고정보 처리 로직(일부)

- 지도화면에서 선택된 지점의 위치정보와 관련 정보를 입력하여 운영센터로 전송가능



[그림 118] 사고지점 등록 시 사용자 인터페이스 흐름(예시)

- 사용자가 선택한 사고 발생지점(또는 제·방빙 작업구역)에는 다음과 같이 빨간색 포인트 이미지가 출력되며 선택한 사고지점의 정보를 등록할지 여부를 확인하는 팝업 가이드를 제공함



[그림 119] 사고 발생지점 이미지와 팝업 가이드 출력(예시)

- 선택 지점에 이미지와 팝업 가이드를 출력 처리하는 로직은 다음과 같음

```
if(airportBoundaryInOut.contains((float) temp.x, (float) temp.y))
{
    // 사고지점 등록 시 사용자가 선택한 위치에 이미지를 출력한다.
    pointPrintDrawHandle.selectPoint = new Coordinate(temp);
    pointPrintDrawHandle.selectPointDrawingFlag = true;

    String Longitude = String.format("%.8f", temp.x);   // 경도를 소수점 8자리까지 출력한다.
    String Latitude = String.format("%.8f", temp.y);    // 위도를 소수점 8자리까지 출력한다.

    AccIntent = new Intent(getBaseContext(), RescueActivity.class);
    AccIntent.putExtra("CarID", Veh_Login_ID);
    AccIntent.putExtra("Longitude", Longitude);
    AccIntent.putExtra("Latitude", Latitude);

    currentPos.x = temp.x;
    currentPos.y = temp.y;
    updateMap();            // 선택한 사고지점을 화면 중앙에 출력한다.

    setConfirmDialog(true);   // 선택지점 확인용 다이얼로그를 출력한다.
}
```

[그림 120] 사고지점 이미지 및 팝업 출력 처리 로직(일부)

- 선택된 사고 발생지점 또는 제·방빙 작업구역에 대한 팝업 가이드를 제공하기 위한 처리는 다음과 같이 공통 로직을 사용함

```
private void setConfirmDialog(final boolean divKind)
{
    AlertDialog.Builder builder = new AlertDialog.Builder(this);

    int msgDiv = R.string.pointPrint_accident;

    if(!divKind)
    {
        msgDiv = R.string.pointPrint_deicing;
    }

    builder.setMessage(getResources().getText(msgDiv).toString());
    builder.setCancelable(false);

    // 확인버튼 클릭 시 해당 정보전송을 위한 창을 출력한다.
    builder.setPositiveButton(getResources().getText(R.string.ok).toString(), new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int whichButton) {
            if(AccIntent != null && divKind)
            {
                startActivityForResult(AccIntent, BasicInfo.REQUEST_CODE_ACC);
            }
            else if(SnowIntent != null && !divKind)
            {
                startActivityForResult(SnowIntent, BasicInfo.REQUEST_CODE_SNOW);
            }
        }
    });

    // 취소버튼 클릭 시 선택지점의 이미지를 제거한다.
    builder.setNegativeButton(getResources().getText(R.string.cancel).toString(), new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int whichButton) {
            pointPrintDrawHandle.selectPointDrawingFlag = false;
        }
    });

    // 확인용 다이얼로그의 위치, 크기를 재설정 한다.
    AlertDialog dialog = builder.show();

    WindowManager.LayoutParams params = dialog.getWindow().getAttributes();
    params.width = 380;
    params.height = 400;
    params.gravity = Gravity.RIGHT;

    dialog.getWindow().clearFlags(WindowManager.LayoutParams.FLAG_DIM_BEHIND);
    dialog.getWindow().setAttributes(params);
}
```

[그림 121] 정보 등록여부 팝업 가이드 처리 로직(일부)

○ 이동체(항공기, 차량) 출력

- 항공기 또는 차량 등 이동체의 위치정보를 수신하여 화면에 출력함

```
Iterator<String> iteratoir = rcvObjectInfoList.keySet().iterator();

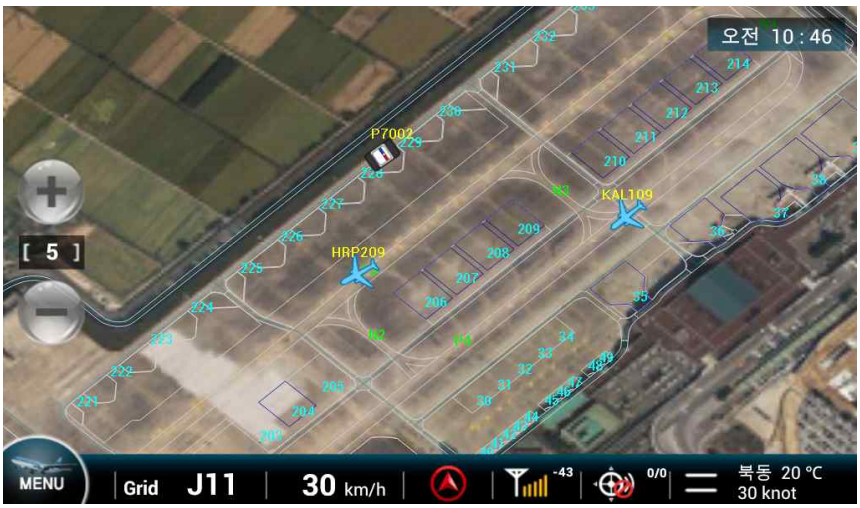
while (iteratoir.hasNext())
{
   String key = (String) iteratoir.next();              // 항공기 편명을 가져온다.
   String[] value = rcvObjectInfoList.get(key);         // 경도, 위도, 속도, 방향값을 가져온다.
   int nStopCount = Integer.parseInt(value[7]);         // 해당 항공기의 stop 수신 횟수를 카운트 한다.

   if(Integer.parseInt(value[4].trim()) == 3 && ( nStopCount == 0 || nStopCount == 3 ) )
   {
      Matrix matrix = new Matrix();                     // 방향값에 따라 이미지를 회전시킬 메트릭스 객체를 생성한다.
      matrix.postRotate(Float.parseFloat(value[3]));    // 방향값을 적용한다.

      Bitmap rotateBitmap = Bitmap.createBitmap(unitImg.get(0), 0, 0,
                                          unitImg.get(0).getWidth(), unitImg.get(0).getHeight(), matrix, true);
      Coordinate convertPos = mapControl.MapToScreen(Double.parseDouble(value[0]), Double.parseDouble(value[1]));

      // 항공기 이미지 및 레이블을 출력한다.
      canvas.drawBitmap(rotateBitmap,
                        (float)(convertPos.x - rotateBitmap.getWidth()/2),
                        (float)(convertPos.y - rotateBitmap.getHeight()/2), null);
   }
}
```

[그림 122] 이동체 출력을 위한 처리 로직(일부)



[그림 123] 이동체 출력 화면 예시

○ 위반 처리

- 속도위반, 출입구역 위반, 기동지역 침범, 항공기 근접, 제방빙 금지구역 침범, 신호위반 등 안정규정 위반 여부를 검사하여 처리함

- 다음과 같이 내부도로에서 제한속도(30km/h) 이상 주행 시 속도위반에 대한 화면 표시 및 음성 알림을 제공함

```
ArrayList<Geometry> temp = handleList.get(7).getDrawHandle().getGeometry_original();

Envelope env = new Envelope(pos.x - offset, pos.y - offset, pos.x + offset, pos.y + offset);
Geometry madeGeo = new GeometryFactory().toGeometry(env);

for(int i = 0; i < temp.size(); i++)
{
   Geometry geo = temp.get(i);

   // 해당 좌표가 위치해있는 도로의 영역(Envelope)을 찾는다.
   if (geo.getEnvelopeInternal().contains(pos))
   {
      if(geo.intersects(madeGeo))
      {
         limitSpeed = Integer.parseInt(handleList.get(7).getDrawHandle().getLimitSpeedList().get(i));

         if(limitSpeed == 100)
         {
            limitSpeed = 50;
         }
      }
   }
}
```

[그림 124] 안전규정 위반 중 속도위반 검사 로직(일부)



[그림 125] 내 차량의 속도위반 예시

○ 내 차량 위치 출력

- GPS로부터 수신되는 위치정보를 기반으로 내 차량의 현재 위치를 출력함

```
// 내 차량의 위치를 출력한다.
if(drawLayerName.contains("my") && mapControl != null )
{
    Matrix matrix = new Matrix();
    matrix.postRotate((float) myHeading);          // 내 위치 이미지를 방향값에 따라 회전한다.

    Bitmap rotateBitmap = Bitmap.createBitmap(unitImg.get(0), 0, 0,
                                              unitImg.get(0).getWidth(), unitImg.get(0).getHeight(), matrix,
                                              true);

    // 내 차량의 지도좌표를 화면좌표로 변환한다.
    Coordinate convertPos = mapControl.MapToScreen(new Coordinate(myPos.x, myPos.y));

    canvas.drawBitmap(rotateBitmap,
                      (float)(convertPos.x - rotateBitmap.getWidth()/2),
                      (float)(convertPos.y - rotateBitmap.getHeight()/2), null);
}
```

[그림 126] 내 차량의 위치 출력 처리 로직(일부)



[그림 127] 내 차량의 현재 위치정보 출력 예시

○ 항공 또는 일반 사진 변경

- 환경 설정에서 제공하는 지도 선택 기능을 통하여 항공사진을 일반사진으로 변경하거나 일반사진을 항공사진으로 변경할 수 있음

- 변경한 설정은 전용단말기를 재부팅 할 경우 적용됨

```
btn_setting_image.setOnClickListener(new OnClickListener() {
   public void onClick(View v) {
      Toast.makeText(getApplicationContext(),
                     "이미지 설정 정보가 저장 되었습니다.",
                     Toast.LENGTH_SHORT).show();

      if( setting_image_idx == 0 )
          TransmitPacket.SetImageSelect(getApplicationContext(), "0");
      else{
          TransmitPacket.SetImageSelect(getApplicationContext(), "1");
   }
});
```

[그림 128] 항공사진 또는 일반사진 변경 처리 로직(일부)

```
imageNumber = TransmitPacket.GetImageSelect(MainActivity.this);
if( imageNumber.equals("0"))
   cmgFileName = "back_GIMPO_pic.cmg";          // 항공사진
else
   cmgFileName = "back_GIMPO.cmg";              // 일반사진

cmgFileStream = openFileInput(cmgFileName);
accessCMG = new AccessCMG();
accessCMG.setCMGFilePath(this.getFileStreamPath(cmgFileName).toString());
```

[그림 129] 항공사진 또는 일반사진 로딩 처리 로직(일부)



[그림 130] 환경 설정에서 제공되는 지도 선택 기능

○ Grid 구역 출력

- 현재 내 차량이 위치한 Grid 구역의 정보를 검사하여 하단 바 좌측에 출력함

```
public synchronized String selectGridArea(Coordinate pt)
{
   DrawingView gridHandle = shpLayerProperty.get(8).getDrawHandle();

   // Grid map 레이어의 좌표정보를 사용하여 위치해있는 Grid 구역 정보를 반환한다.
   for (int i = 0; i < gridHandle.getGeometry_original().size(); i++)
   {
      // Grid 구역을 구성하는 좌표정보를 가져온다.(각 구역의 좌표정보)
      Geometry temp = gridHandle.getGeometry_original().get(i);

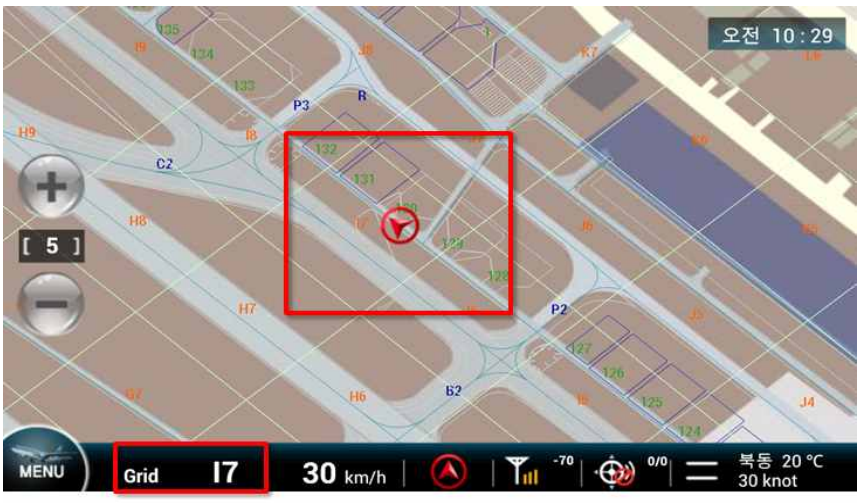
      // 해당 좌표가 위치해있는 Grid 구역 polygon의 영역(Envelope)을 찾는다.
      if (temp.getEnvelopeInternal().contains(pt))
      {
         Coordinate[] area = temp.getBoundary().getCoordinates();
         PointInPolygon findPointInOut = new PointInPolygon();

         for(int j = 0; j < area.length-1; j++)
            findPointInOut.addPoint((float) area[j].x, (float) area[j].y);

         // 해당 좌표가 Grid 구역에 포함되어 있는지 여부를 검사한다.
         if(findPointInOut.contains((float) pt.x, (float) pt.y))
            return gridHandle.getGridIDList().get(i);
      }
   }

   return "";
}
```

[그림 131] 내 차량이 위치한 Grid 구역 정보 검사 로직(일부)



[그림 132] 내 차량이 위치한 Grid 구역 정보 출력 예시

○ 활주로 레이블 출력

- 2본으로 구성되어 있는 김포공항 활주로의 레이블을 지도화면에 출력함



[그림 133] 김포공항 활주로의 레이블 출력(32L-14R, 32R-14L)

- 김포공항에는 3200m x 60m의 32L-14R(쓰리투레프트-원포라이트) 신활주로와 3600m x 45m의 32R-14L(쓰리투라이트-원포레프트) 구활주로가 있음

```
// [김포공항 활주로 : 2본으로 구성]
// 1. 3200m x 60m : 32L-14R(쓰리투레프트-원포라이트) - 신활주로
// - 32L : 126.80256136 37.54677422
// - 14R : 126.77392325 37.56968471
// 2. 3600m x 45m : 32R-14L(쓰리투라이트-원포레프트) - 구활주로
// - 32R : 126.80859720 37.54656874
// - 14L : 126.77674853 37.57204767
//
// [활주로 명 저장 순서]
// 32L, 14R, 32R, 14L
drawHandle.runway_original.add(new Coordinate(126.80256136, 37.54677422)); // 32L
drawHandle.runway_original.add(new Coordinate(126.77422412, 37.56946212)); // 14R
drawHandle.runway_original.add(new Coordinate(126.80859720, 37.54656874)); // 32R
drawHandle.runway_original.add(new Coordinate(126.77703746, 37.57181209)); // 14L

if(runway_current.size() > 0)
{
    for(int i = 0; i < runway_current.size(); i++)
    {
        Coordinate runway = runway_current.get(i);
        canvas.drawText(runwayLable[i], (float) runway.x, (float) runway.y, label);
    }
}
```

[그림 134] 활주로 레이블 출력 로직(일부)

3. 최종 성과물

○ 본 과제를 통하여 연구개발이 완료된 공항 전용단말기의 형상은 다음과 같음

구분	내용
전면과 후면 형상	

[표 136] 연구개발 최종 성과물(전용단말기) 형상

○ 공항 전용단말기와 상용 내비게이션의 사양은 다음과 같음

항목	팅크웨이 아이나비	파인디지털 파인드라이브	공항 전용단말기
화면크기	7인치	7인치	7인치
해상도	800 x 480	800 x 480	1024 x 600
운영체제	Windows CE 6.0	Windows Embedded Compact 7	Android
CPU	Prima II	TCC8930	Exynoss 4412 (Coretex-A9)
RAM	256 MB	512 MB	1 GB
GPS	사용	사용	사용
통신모듈	없음	없음	Wi-Fi

[표 137] 공항 전용단말기와 상용 내비게이션 비교

3절. 연구수행 결과 평가기준 및 검증 방법

구분	핵심 성과물 (level1)		단위 성과물 (level2)		당해년도 성과목표				해당 기관	예산 (백만)	정량목표
					질적목표 (측정방법)		목표치	검증 방법			
1차 년도	1	공항시설 안전관리 통합 시스템 전용 단말기 개발	1-1	시제품제작	①	공항시설 안전통합 관리시스템 전용 단말기 시제품 디자인 및 제작	1	시제품 확인 (보드수준)	인소팩	179	시제품 : 1건 소프트웨어 등록 : 1 건 보고서 : 1 건
			1-2	지식재산권 행사를 위한 등록	①	해당 사업을 통해 산출된 소프트웨어 등록 건수	1	본 사업을 통해 개발된 소프트웨어 등록 건수			
			1-3	보고서	①	보고서 작성	1	보고서 확인			
	2	공항시설 안전관리 통합 시스템 적용 메뉴얼 연구	2-1	보고서	①	보고서 작성	1	보고서 확인	한국 교통 연구원	30	보고서 : 1 건
계							4			209	시제품 : 1건 보고서 : 2 건 소프트웨어 등록 : 1 건

2차 년도	1	공항시설 안전관리 통합 시스템 전용 단말기 개발	1-1	기구제작	①	공항시설 안전통합 관리시스템 전용 단말기 기구 제작	1	시제품 확인	인소팩	284.9	시제품 : 1건 소프트웨어 등록 : 1 건 보고서 : 1 건
			1-2	지식재산권 행사를 위한 등록	①	해당 사업을 통해 산출된 소프트웨어 등록 건수	1	본 사업을 통해 개발된 소프트웨어 등록 건수			
			1-3	보고서	①	보고서 작성	1	보고서 확인			
	2	공항시설 안전관리 통합 시스템 적용 메뉴얼 연구	2-1	보고서	①	보고서 작성	1	보고서 확인	한국 교통 연구원	40	보고서 : 1 건
계							3			324.9	시제품 : 1건 보고서 : 2 건 소프트웨어 등록 : 1 건

3차 년도	1	공항시설 안전관리 통합 시스템 전용 단말기 개발	1-1	기구제작	①	공항시설 안전통합 관리시스템 전용 단말기 기구 제작	1	시제품 확인	인소팩 (주)	320.9	기구제작 : 1건 소프트웨어 등록 : 1 건 보고서 : 1 건
			1-2	지식재산권 행사를 위한 등록	①	해당 사업을 통해 산출된 소프트웨어 등록 건수	1	소프트웨어 등록 건수			
			1-3	보고서	①	보고서 작성	1	보고서 확인			
계							3			320.9	기구제작 : 1건 소프트웨어 등록 : 1 건 보고서 : 1 건

[표 138] 연구개발 결과물의 평가기준

3장. 최종 연구성과 및 적용실적

1절. 주요 연구성과

1. 전용단말기 하드웨어 개발

○ 라이선스 문제가 없는 Android 운영체제 기반의 공항시설 안전통합 관리시스템 전용단말기 하드웨어 개발 완료



[그림 135] 전용단말기 하드웨어 형상(최종 성과물)

가. 기존 문제점 대비 개선 구현 사항

○ 기존 AFSIS-6 차량단말기의 문제점인 상용 노트북의 크기 축소 및 통신장비의 모듈화

- 차량 내 운전자의 시야 확보 및 복잡한 배선 문제를 7인치 크기의 모니터와 Wi-Fi 통신장치의 내장화 등을 통한 간단한 선 정리

- 2개의 크고 긴 안테나 사용으로 인한 이탈 문제를 1개의 작은 크기로 안테나로 성능 구현

작업차량에 설치 시 이탈 위험성 존재	문제점 개선 후
	

[표 139] 기존 문제점 대비 개선 구현 사항

나. 전용단말기의 디자인 진행

○ 단계 별 디자인 시안 검토를 통하여 최적의 디자인 창출

○ 상용 내비게이션의 형태 유지와 공항 내 사용성을 고려한 디자인 개발

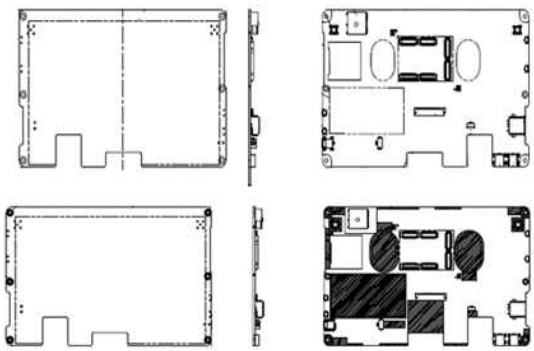
개념 및 전체 형상을 위한 디자인 (초기단계)	기구 사출 및 운용을 위한 디자인 (최종 단계)
	

[표 140] 전용단말기 디자인

다. 전용단말기의 기구/보드 설계 및 시제 제작

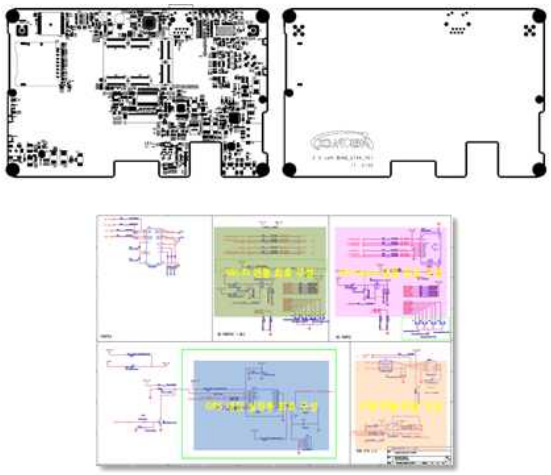
○ 기구 내부에 조립되는 보드의 크기와 연결되는 부품을 고려한 설계 진행

○ QDM 사출과 PC 재질의 기구 형상 제작 완료

전/후면 기구 설계	사출 기구 형상
	

[표 141] 전용단말기 기구 설계 및 사출

○ 기구 크기 축소, 실장 공간 최소화, GPS 내장 안테나 구현, 개발 단계 회로 삭제 등 회로 구현 완료

전용단말기 보드 설계 및 구현	시제 제작 및 각 모듈별 동작 검증
	

[표 142] 전용단말기 보드 설계 및 시제 제작

2. 전용단말기 소프트웨어 개발

○ 공항 전용단말기에 탑재되는 Android 운영체제 기반의 GIS 엔진 및 응용프로그램 개발을 완료함

GUI 구성 및 모듈별 기능 구현	Android 운영체제 기반의 GIS 엔진 및 기능 개발(지도화면 등)
	

[표 143] Android 운영체제 기반의 전용단말기 소프트웨어 화면

3. 연구개발을 통한 성과 정보

○ 본 연구개발 과제를 통하여 연차별 보고서, 프로그램 등록증 등이 성과로 산출됨

○ 주관 연구기관인 인소팩(주)은 공항시설 안전통합 관리시스템 전용단말기 개발을 통하여 전용단말기 제작, 하드웨어 및 소프트웨어 개발보고서 3건, 프로그램 등록증 3건이 연구결과 성과를 달성함

○ 공동 연구기관인 한국교통연구원은 AFSIS-6의 적용 매뉴얼 연구를 통하여 테스트베드 관리 및 정책적 지원에 대한 연구 보고서 2건을 성과로 달성함



[그림 136] 연차별 보고서 및 프로그램 등록증 형상

○ 연구개발 결과로 등록된 소프트웨어 정보는 다음과 같음

순번	프로그램 명칭	창작일	등록일	등록번호	저작자명/ 저작권자명
1	안드로이드 운영체제용 공항시설 안전통합 관리시스템 전용단말기 프로그램 Ver 0.0	2013.03.04	2013.03.27	제 C-2013-006267 호	인소팩(주)
2	안드로이드 운영체제용 공항시설 안전통합 관리시스템 전용단말기 프로그램 Ver 1.0	2013.12.09	2013.12.24	제 C-2013-028525 호	인소팩(주)
3	안드로이드 운영체제용 공항시설 안전통합 관리시스템 전용단말기 프로그램 Ver 2.0	2014.06.20	2014.06.23	제 C-2014-014372 호	인소팩(주)

[표 144] 전용단말기의 프로그램 등록 현황

○ 전용단말기의 시제품 제작 관련 정보는 다음과 같음

순번	시제품명	제작일	제작업체명	설치장소
1	공항시설 안전통합 관리시스템 전용단말기	2013.11	인소팩(주)	김포공항
2	공항시설 안전통합 관리시스템 전용단말기	2014.04	인소팩(주)	김포공항

[표 145] 전용단말기의 시제품 제작 현황

○ 김포공항 테스트베드를 통한 전용단말기 현장시험 정보는 다음과 같음

순번	현장시험명	시험일	시험장소	주요내용
1	김포공항 현장시험 1차	2013.09.26. ~ 2013.09.27	김포공항	GPS, Wi-Fi, 전용단말기 기능시험
2	김포공항 현장시험 2차	2013.10.07. ~ 2013.10.09	김포공항	GPS, Wi-Fi, 전용단말기 기능시험
3	김포공항 현장시험 3차	2013.10.18	김포공항	Wi-Fi 거리 및 음영지역 시험 GPS 오차 및 위반 기능시험
4	김포공항 현장시험 4차	2013.11.06 ~ 2013.11.15	김포공항	전용단말기 단위/통합 기능시험

[표 146] 전용단말기의 테스트베드를 통한 현장시험 진행사항

2절. 테스트베드 구축 및 운용시험

1. 개요

○ 김포공항에 테스트베드를 통하여 공항 내 실제 작업차량에서 전용단말기를 사용할 수 있도록 환경을 구축함

○ 테스트베드에서 현장시험 진행 후 '13년도 12월에 전용단말기 실제 운용시험을 위하여 공항 내 작업차량에 설치를 완료함

○ 작업차량 내 실제 설치 후에도 사용자 개선사항 해결을 위하여 Wi-Fi, GPS 성능시험 등을 진행함

회차	일시	장소	내용
1	2013년 01월 11일	한국공항공사 전산실	• AFSIS-6 장비 및 소프트웨어 1차 점검
2	2013년 07월 17일	한국공항공사	• 국토부 회의 - AFSIS-6 소개 및 상용화 개발 현황
3	2013년 08월 09일	한국공항공사 전산실	• AFSIS-6 장비 및 소프트웨어 2차 점검
4	2013년 09월 26일 ~2013년 09월 27일	공항 내부도로/ 소방관제센터	• 전용단말기 1차 현장시험 - Radius 인증 및 거리시험 - 전용단말기 기능시험
5	2013년 10월 07일 ~2013년 10월 09일	공항 내부도로/ 소방관제센터	• 전용단말기 2차 현장시험 - Radius 인증 및 거리시험 - GPS 성능시험 - 전용단말기 기능시험
6	2013년 10월 18일	공항 내부도로/ 소방관제센터	• 전용단말기 3차 현장시험 - Radius 인증 및 거리시험 - GPS 성능시험 - Wi-Fi 안테나 거리시험 및 음영지역 검증 - 전용단말기 기능시험

회차	일시	장소	내용
7	2013년 11월 06일 ~2013년 11월 15일	공항 내부도로/ 소방관제센터/ 재해대책본부	• 전용단말기 4차 현장시험 - 전용단말기 단위/통합 기능시험
8	2013년 12월 09일 ~2013년 12월 10일	재해대책본부/ 조업사	• 전용단말기 10대, 운영센터 1대 설치
9	2013년 12월 27일	재해대책본부/ 조업사	• 제설차량에 설치된 전용단말기 5대 교체 (자동 전원기능 적용) • 동보공항 작업차량의 전용단말기 사용법 전달 및 운행 확인 • GPS 정확도 시험 • 미들웨어 및 운영센터 바이너리 교체
10	2014년 03월 14일	재해대책본부/ 조업사	• 동보공항, 샤프, 진에어, 이스타항공, 아스항공 작업차량의 전용단말기 교체(자동 전원기능 적용) • 미들웨어 점검 및 항공기 출력 확인
11	2014년 05월 19일 ~2014년 05월 21일	공항 내부도로/ 소방관제센터/ 재해대책본부	• GPS 정확도 시험 • 이스타항공 작업차량 장비 교체
12	2014년 06월 03일 ~2014년 06월 12일	한국공항공사/ 재해대책본부	• 테스트베드 장비 점검 • 국토부 회의 준비
13	2014년 06월 13일	한국공항공사/ 재해대책본부	• 국토부 회의 - AFSIS-6 연구개발 검토회의 및 시연
14	2014년 07월 01일 ~2014년 07월 04일	공항 내부도로/ 소방관제센터/ 재해대책본부	• GPS 정확도 시험 • Wi-Fi 성능 시험
15	2014년 08월 18일 ~2014년 08월 20일	공항 외부도로/ 재해대책본부	• 국토교통기술사업화지원사업(본 연구개발 과제)의 현장평가(최종평가) 준비를 위한 운용시험 및 평가수행

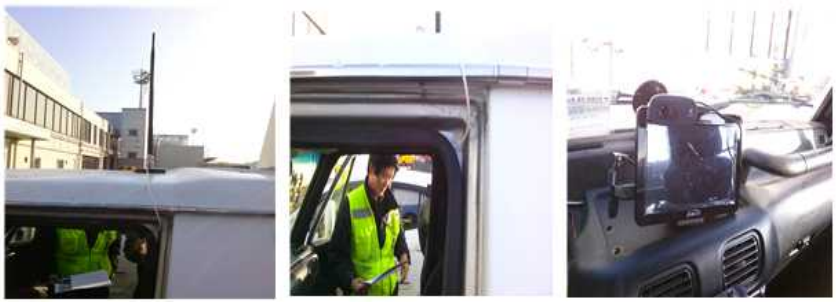
[표 147] 김포공항 테스트베드 운용시험 진행현황

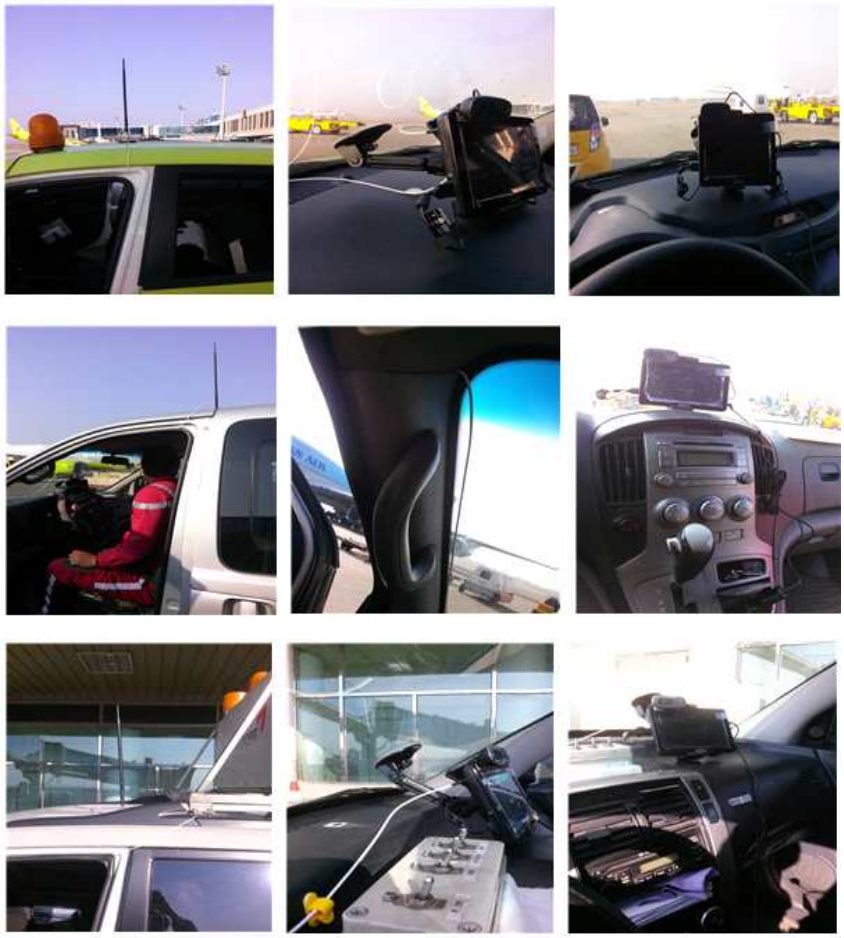
2. 김포공항 테스트베드 구축

가. 공항 내 실제 작업차량에 전용단말기 설치

○ 김포공항공사와의 협조를 통하여 실제 운용 차량 내 설치 및 운용 지원

- 다양한 차량 종류에 맞추어 사용자의 조작 및 운용이 용이하도록 설치

설치 차량	내 용
토목/정비	
제설차량	
순찰차량	
지상조업 차량	

설치 차량	내 용
안전보안 차량	

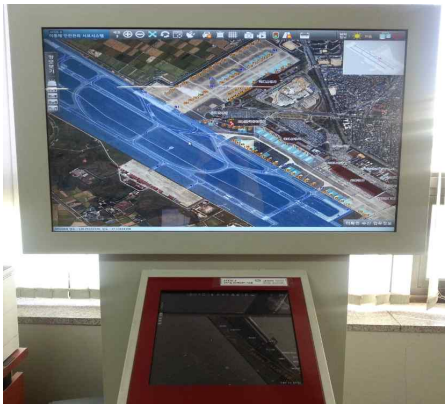
[표 148] 김포공항 작업차량에 전용단말기 설치 작업

나. 운영센터 설치

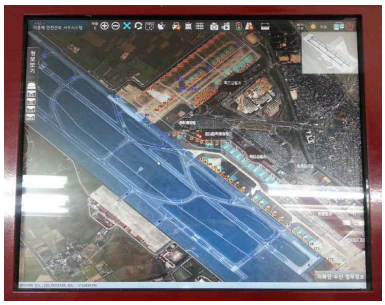
○ 김포공항 내 재해대책본부에 운영센터 1대를 설치하여 항공기 및 차량의 모티링 제공



[그림 137] 키오스크 설치완료 화면(전체)



[그림 138] 키오스크 모니터(1)



[그림 139] 키오스크 모니터(2)

다. 전용단말기 사용현황 및 불편사항 문의

(1) 1차 문의

○ 일시 : 2014년도 02월 19일(수)

전용단말기 설치부서	사용현황 및 불편사항	개선 후 적용현황
한국공항공사 서울지역본부 토목조경팀	• 제설차량 8대 중 4대에만 설치가 되어 운영센터에서 실 차량의 상태 모니터링이 어려움 • 국내선, 국제선 계류장에서 차량 보이지 않음(AP 문제) • 차량의 시동을 키면 전용단말기도 자동으로 전원 켜졌으면 함	차량시동과 함께 전용단말기 부팅기능 적용완료
진에어 안전보안팀	• 실 사용자에게 문의 후 불편사항 발생 시 전달요청	-
아스공항 김포지점	• 현재 작업 시 운용하고 있음 • 불편사항 발생 시 전달요청	-
동보공항 안전보안팀	• 퀵 가이드(또는 매뉴얼)을 전달받지 못하였음 • 현재 작업 시 운용하고 있음 • 불편사항 발생 시 전달요청	전용단말기퀵 가이드 전달 및 설명완료
샤프 지상조업부	• 전용단말기 장착확인만 함, 사용방법을 숙지한 후 공식적으로 사용예정 • 작업 시 사용해보고 불편사항 발생 시 전달요청	-
이스타항공 안전보안팀	• 실 사용자에게 문의 후 불편사항 발생 시 전달요청	-

[표 149] 전용단말기 사용관련 1차 문의사항 정리

(2) 2차 문의

○ 일시 : 2014년도 03월 20일(목)

전용단말기 설치부서	사용현황 및 불편사항	개선 후 적용현황
한국공항공사 서울지역본부 토목조경팀	• 운영센터에서 모니터링 시 음영지역에서는 차량이 보이지 않음 • 제설작업 시 전용단말기와 운영센터 사용하고 있음	-
진에어 안전보안팀	• 현재 작업 시 운용하고 있음 • 불편사항 발생 시 전달요청	-
아스공항 김포지점	• 현재 작업 시 운용하고 있음 • 불편사항 발생 시 전달요청	-
동보공항 안전보안팀	• 퀵 가이드(또는 매뉴얼)을 전달받음	작업시 전용단말기 사용요청
샤프 지상조업부	• 업무 중 사용하고 있음	-
이스타항공 안전보안팀	• 램프작업차량에 장착되어 매일 운행되고 있음 • 불편사항 발생 시 전달요청	-

[표 150] 전용단말기 사용관련 2차 문의사항 정리

(3) 3차 문의

○ 일시 : 2014년도 04월 22일(화)

전용단말기 설치부서	사용현황 및 불편사항	개선 후 적용현황
한국공항공사 서울지역본부 토목조경팀	• 운영센터에서 모니터링 시 음영지역에서는 차량이 보이지 않음	-
진에어 안전보안팀	• 불편사항 발생 시 전달요청 • 차량시동 후 자동 로그인 되어 사용 중(주요 작업구역 : 램프)	-
아스공항 김포지점	• 현재 작업 시 운용하고 있음	-
동보공항 안전보안팀	• 주로 국내선청사에서 작업 진행(동보공항 스팟)	-
샤프 지상조업부	• 설치차량이 업무용으로 사용되는 차량으로 사용빈도가 낮음 • 사용할 경우에 특별한 문제는 없었음(주요 작업구역 : 램프)	-
이스타항공 안전보안팀	• 램프작업차량에 장착되어 매일 운행되고 있음 • 불편사항 발생 시 전달요청	-

[표 151] 전용단말기 사용관련 3차 문의사항 정리

3. 현장시험 및 실제 운용시험 진행

가. 전용단말기 현장시험 (1~4차)

(1) 목적

- Radius 인증사용(전용단말기 보안)시 정상 작동 유무
- 공항 내에서 GPS 작동 시 전용단말기에 차량단말 서브시스템 구동 시 차량 출력 및 기능정상 작동 유무 확인
- 고장난 AP를 제외한 공항 내 모든 지역에서의 Wi-Fi 출력 조정 및 스캔 인터벌 조정을 통한 최적의 AP 스캔 방안 검토, 안테나별 거리 및 음영지역 시험
- 공항 내에서 전용단말기와 운영센터 송수신 테스트
- 지향성 패치 안테나를 통한 Wi-Fi 성능 개선 유무 확인

(2) 범위

- 전용단말기 접속간 Radius server로부터의 인증 정상작동 여부를 확인하여 단말 보안 접속기능이 정상적으로 수행되는지 확인
- 공항 내에서 GPS 작동 시 위치 출력 및 기능동작 확인
- Wi-Fi 출력 및 스캔 인터벌 조정을 통한 고장난 AP를 제외한 모든 구역에서의 300m마다 사진 송수신을 통한 Wi-Fi 안테나별 끊김 유무 확인
- 전용단말기와 운영센터간 업무정보의 송수신 정상동작 확인
  - 차량단말 서브시스템과 미들웨어 서브시스템간 위치정보 송수신 확인
  - 차량단말 서브시스템과 미들웨어 서브시스템간 업무정보 송수신 확인
  - 미들웨어 서브시스템과 이동체 안전관리 서브시스템간 업무정보 송수신 확인
  - 미들웨어 서브시스템과 차량단말 서브시스템 및 이동체 안전관리 서브시스템간 차량의 위치정보 송수신 확인
  - 차량단말 서브시스템에서 송신한 사고, FOD, 위반, 제방빙 관련 업무정보 저장 및 확인
  - 이동체 안전관리 서브시스템에서 처리한 사고, FOD, 위반, 제방빙 관련 업무정보의 송수신 및 저장 확인

(3) 일시

○ 2013년 09월 26일(목) ~ 27일(금) (1차)

○ 2013년 10월 07일(월) ~ 09일(수) (2차)

○ 2013년 10월 18일(금) (3차)

○ 2013년 11월 06일(수) ~ 15일(금) (4차)

(4) 시험장소

○ 김포국제공항 내부도로(주기장) 및 소방관제센터 4층

(5) 시험내용

(가) Radius 인증 및 거리시험 계획 (1~2차)

① 공항 내 소방관제센터에서 단독 작업

○ Radius server에 접속하기 위해 14번 AP만을 접속 시킨 후 단독 테스트 진행 예정

○ 단말 접속간 Radius server로부터 인증이 정상적으로 수행되는지 확인하여 단말 보안 접속기능이 정상적으로 수행되는지 확인

② GPS 작동 시 공항 전 지역에서의 위치 출력 및 기능동작 확인 (1~3차)

○ 차량 이동 시 GPS를 통한 자신의 차량 위치 출력 확인

○ 항공사진 출력에 따른 자신의 차량 위치 및 속도 정상출력 확인

○ GPS 펌웨어 변경에 따른 자신의 차량 위치 출력 확인

- Baudrate이 4800으로 설정되어 있을 경우
- Baudrate이 9600으로 설정되어 있을 경우

③ 공항 내 고장난 AP 지역을 제외하고 Wi-Fi 연결 상태 검증 및 음영지역 시험

○ 공항 내 모든 AP를 히든으로 작동시키고, 고장난 AP 지역을 제외한 나머지 AP 지역에서 샤크안테나2로 사진 전송을 통해 300m 거리별 Wi-Fi 끊김 상태 확인

○ 현재 3, 16, 17, 18 AP 고장으로 판단되며 추가로 고장 및 미작동 되는 AP 확인 후 테스

트 진행 예정

○ 14번 AP 거리 테스트는 Radius 인증이 테스트가 끝난 후 진행

○ AP 로밍 시 자동 갱신 적용 여부 확인

(나) 기능시험 계획

① 시험 절차

○ 장비 설치 및 기능시험 환경 구성

○ 차량단말 서브시스템에서의 자기차량과 타 차량 출력 확인

○ 이동체 안전관리 서브시스템에서의 차량 출력 확인

○ 차량단말 서브시스템과 이동체 안전관리 서브 시스템간 업무정보 송수신 시험

② 시험 기능 목록

○ 차량단말 서브시스템(전용단말기)

기능	설명
로그인	• 차량단말 서브시스템을 사용하기 위한 사용자 인증 진행
GPS	• GPS 연결 상태정보 제공
무선 네트워크	• 무선 네트워크 연결 상태정보 제공
수신 메시지 목록	• 수신한 사고, FOD, 위반 관련 업무정보 목록 및 상세정보 제공 • 출동요청 등에 대한 조치완료 정보 전송 인터페이스 제공 • FOD 관련 이미지 제공 • 사고 관련 정보일 경우 현재 위치로부터 사고 발생 지점까지의 최적 경로 및 거리 제공
FOD 등록	• 이동체 안전관리 서브시스템으로 공항 내 발견된 이물질 및 포장면 파손 등에 대한 FOD 정보 전송

기능	설명
사고지점 등록	• 이동체 안전관리 서브시스템으로 공항 내 발생된 사고 정보 전송
이동체 출력	• 지도화면에 공항 내 위치하는 항공기 및 차량 출력
위반 알림	• 운행 중인 차량의 과속, 항공기와의 근접 등의 위반 사항 발생 시 화면 표시 및 음성 알림
내 위치 출력	• 지도화면에 내 차량의 위치 출력
격자 구역 출력	• 차량이 위치한 격자 구역 출력
경도, 위도 출력	• 차량의 위치 정보(경도, 위도) 출력
속도	• 내 차량의 속도 표시
설정	• 프로그램 버전, 단말기 ID, 이동체 범례 등의 정보 제공

[표 152] 차량단말 서브시스템 시험 기능 목록

○ 이동체 안전관리 서브시스템

기능	설명
로그인	• 이동체 안전관리 서브시스템을 사용하기 위한 사용자 인증 진행
통신상태 확인	• 미들웨어 서브시스템과의 통신연결 상태 출력
미확인 수신 업무정보 목록	• 수신된 미확인 업무정보의 목록 및 상세정보 제공
이동체 출력	• 지도화면에 공항 내 위치하는 항공기 및 차량 출력
FOD 등록	• 이동체 안전관리 서브시스템으로 공항 내 발견된 이물질 및 포장면 파손 등에 대한 FOD 정보 전송
사고지점 등록	• 공항 내 발생된 사고 정보 등록
사고 및 FOD 발생 지점 확인	• 지도화면에 사고 및 FOD의 발생 지점 표시

[표 153] 이동체 안전관리 서브시스템 시험 기능 목록



[그림 140] 김포국제공항 내 AP 설치 위치



[그림 141] 무선 AP 설치 및 안테나 방향

순번	무선 AP 설치장소	네트워크 채널
1	신 소방 구조대 (NS)	CH1, CH 13
2	신 소방 구조대	CH13, CH 9

3	구 소방 구조대 #1	CH13, CH 9
4	구 소방 구조대 #2	CH5, CH 1
5	국내선 청사	CH13, CH 9
6	화물 청사	CH1, CH 5, CH 9
7	초소 1번 지점	CH9, CH 1, CH 9
8	초소 2번 지점	CH13, CH 9
9	초소 3번 지점	CH5, CH 13
10	초소 4번 지점	CH5
11	초소 5번 지점	CH9, CH 13
12	초소 6번 지점	CH5, CH 1
13	초소 9번 지점	CH9
14	초소 10번 지점	CH13, CH 9
15	초소 11번 지점	CH5, CH 1

[표 154] AP별 802.11a(2.4GHz) 네트워크 채널

다이폴 안테나	샤크 안테나(1, 2)	지향성 안테나
	 샤크 안테나1   샤크 안테나2	

[표 155] 시험(1~3차) 시 사용된 안테나의 종류

루프랙 고정	자석부착
	
• 차량에 설치된 루프랙에 패치 안테나를 고정하여 사용	• 지향성 패치 안테나 하단부에 설치된 브라켓을 제거한 후 아랫면에 자석을 부착하여 샤크 안테나처럼 차량에 장착하여 사용

[표 156] 패치 안테나 고정 방안

③ 전용단말기 기능시험을 위한 장비 구성

○ 전용단말기(차량단말 서브시스템)

- 순찰 차량, FOD 차량, 제방빙 차량으로 구성하며 무선 네트워크로 연결

- 순찰 차량, FOD 차량, 제방빙 차량을 기본으로 설정하여 출력하되 사고, FOD, 위반 등 업무정보 송수신 시험 시 구조소방 차량, 구급 차량 등으로 차량의 종류를 변경하여 진행

○ 미들웨어 및 데이터베이스 서브시스템

- 하나의 서버 PC에 설치하여 연동하며 공항 내 AP에 연결

○ 이동체 안전관리 서브시스템

- 별도의 PC에 설치 및 실행하며 미들웨어 서브4시스템과 LAN 케이블로 연결



[그림 142] 기능시험을 위한 장비 구성도

(6) 시험결과

(가) Radius 인증 시험 (1~2차)

○ 현장시험 2차 시 Wi-Fi 접속간 Radius 인증 성공

○ AP roaming 시 radius 접속실패 현상 해결

(나) Wi-Fi 안테나 거리 및 음영지역 시험 (1~3차)

○ 시험 1~3차 진행 시 사용된 안테나 종류는 다음과 같음

- 1차 시험 : 샤크 안테나2와 다이폴 안테나 성능시험

- 2차 시험 : 샤크 안테나2, 다이폴 안테나, 지향성 안테나

- 3차 시험 : 샤크 안테나2, 다이폴 안테나, 지향성 안테나

※ 7번 시료 : 샤크 안테나2, 8번 시료 : 다이폴 안테나, 9번 시료 : 지향성 안테나

거리		0M			300M			600M			900M			비 고
시료 / AP		7	8	9	7	8	9	7	8	9	7	8	9	
1AP	AP	1번 e2:89	1번 e2:89	1번 dd:91	1번 e2:89	1번 e2:89	1번 e2:89	1번 e2:89	1번 e2:89	1번 e2:89	1번 e2:89	1번 e2:89	1번 e2:89	
	감도	-67	-51	-56	-60	-51	-48	-68	-54	-66	-77	-63	-68	
	수신	o	o	o	o	o	o	o	o	o	o	o	o	
2AP	AP	1번 e2:89	1번 e2:89	2번 e8:83	1번 e2:89	1번 e2:89	1번 e2:89	2번 e8:83	1번 e2:89	-	2번 e8:83	1번 e2:89	4번 e7:b5	
	감도	-70	-56	-52	-84	-66	-85	-86	-81	-	-87	-82	-70	
	수신	o	o	o	o	o	o	o	o	x	o	x	o	
4AP	AP	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	5번 e4:eb	
	감도	-49	-31	-44	-69	-56	-53	-74	-56	-66	-77	-61	-57	
	수신	o	o	o	o	o	o	o	o	o	o	o	o	

거리		0M			300M			600M			900M			비 고
AP \ 시료		7	8	9	7	8	9	7	8	9	7	8	9	
5AP	AP	4번 e7:b5	4번 e7:b5	5번 e4:eb	4번 e7:b5	4번 e7:b5	5번 e4:eb	4번 e7:b5	4번 e7:b5	5번 e4:eb	4번 e7:b5	4번 e7:b5	5번 e4:eb	
	감도	-74	-58	-70	-82	-64	-53	-83	-67	-68	-82	-65	-71	
	수신	o	o	o	o	o	o	o	o	o	o	o	o	
6AP	AP	4번 e7:b5	4번 e7:b5	5번 e4:c3	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	4번 e7:b5	x	4번 e7:b5	15번 e7:93	
	감도	-82	-65	-68	-81	-65	-66	-83	-68	-68	x	-81	-70	
	수신	o	o	o	o	o	o	o	o	o	x	o	o	
7AP	AP	7번 e5:15	8번 0d:ad	x	x	x	x	x	x	x	x	x	9번 e6:f7	도로 안에 AP 위치하여 테스트 불가
	감도	-86	-59	x	x	x	x	x	x	x	x	x	-75	
	수신	o	o	x	x	x	x	x	x	x	x	x	o	
8AP	AP	7번 e5:15	8번 0d:ad	x	x	x	x	x	x	x	x	x	9번 e6:f7	7AP 위치 동일
	감도	-86	-59	x	x	x	x	x	x	x	x	x	-75	
	수신	o	o	x	x	x	x	x	x	x	x	x	o	
9AP	AP	9번 e6:f7	9번 e6:f7	9번 e6:f7	9번 e6:f7	9번 e6:f7	9번 e6:f7	9번 e6:f7	9번 e6:f7	9번 e6:f7	10번 e6:e3	9번 e6:f7	9번 e6:f7	
	감도	-70	-50	-52	-75	-64	-47	-63	-64	-47	-76	-79	-55	
	수신	o	o	o	o	o	o	o	o	o	o	o	o	
10AP	AP	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	10번 e6:e3	
	감도	-75	-51	-77	-76	-59	-43	-75	-55	-37	-81	-68	-63	
	수신	o	o	o	o	o	o	o	o	o	o	o	o	

거리		0M			300M			600M			900M			비 고
AP \ 시료		7	8	9	7	8	9	7	8	9	7	8	9	
11AP	AP	11번 e8:e3	10번 e6:e3	11번 dd:7f	11번 e8:e3	10번 e6:e3	11번 dd:7f	x	12번 07:c9	12번 07:c9	12번 07:c9	12번 07:c9	12번 07:c9	
	감도	-68	-63	-39	-86	-74	-72	x	-63	-60	-77	-45	-60	
	수신	o	o	o	o	o	o	x	o	o	o	o	o	
12AP	AP	12번 07:c9	12번 07:c9	12번 07:c9	x	12번 07:c9	12번 07:c9	x	x	12번 e6:eb	x	x	x	
	감도	-83	-63	-56	x	-71	-59	x	x	-56	x	x	x	
	수신	o	o	o	x	o	o	x	x	o	x	x	x	
13AP	AP	x	x	x	x	x	x	15번 dd:6f	15번 dd:6f	12번 e6:eb	14번 c3:28	15번 dd:6f	14번 c3:28	
	감도	x	x	x	x	x	x	-69	-52	-60	-63	-75	-35	
	수신	x	x	x	x	x	x	o	o	o	o	o	o	
14AP	AP	15번 dd:6f	15번 dd:6f	15번 dd:6f	x	x	8번 0d:ed	x	x	x	x	x	x	계류장 AP 사용 불가 영향 받음
	감도	-66	-44	-52	x	x	-67	x	x	x	x	x	x	
	수신	o	o	o	x	x	o	x	x	x	x	x	x	
15AP	AP	15번 dd:6f	15번 dd:6f	15번 dd:6f	x	x	8번 0d:ed	x	x	x	x	x	x	15AP 위치 동일
	감도	-66	-44	-52	x	x	-67	x	x	x	x	x	x	
	수신	o	o	o	x	x	o	x	x	x	x	x	x	
확률		100%	100%	92.3 %	61.5 %	69.2 %	61.5 %	61.5 %	69.2 %	69.2 %	61.5 %	53.8 %	61.5 %	

[표 157] Wi-Fi 시험 결과(2차)

○ 다음의 내용은 현장시험 3차(4번 AP 지역에서 거리시험) 진행 결과임

※ 7번 시료 : 다이폴 안테나, 8번 시료 : 샤크 안테나2, 9번 시료 : 지향성 안테나

거리		0M			300M			600M			900M			비고
시료 / AP		7	8	9	7	8	9	7	8	9	7	8	9	
4AP 첫번째	AP	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	● 900M에서 9번 시료 사용 시 이미지 전송 실패
	감도	-37	-40	-35	-47	-51	-60	-53	-62	-66	-60	-63	-72	
	수신	O	O	O	O	O	O	O	O	O	O	O	O	
	거리	1200M			1500M			1800M						비고
		7	8	9	7	8	9	7	8	9				
	AP	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5				● 1200M 9번 시료 메시지 송부만 가능 ● 1800M 8번 시료 메시지 송부만 가능
	감도	-61	-67	-74	-65	-68	-76	-65	-71	-79				
	수신	O	O	X	O	O	X	O	X	X				

거리		0M			300M			600M			900M			비고
시료 / AP		7	8	9	7	8	9	7	8	9	7	8	9	
4AP 첫번째	AP	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	● 900M 9번 시료 메시지 송부만 가능 ● 1800M 8번 시료 메시지 송부만 가능
	감도	-39	-35	-35	-47	-51	-60	-53	-60	-66	-60	-63	-71	
	수신	O	O	O	O	O	O	O	O	O	O	O	X	
	거리	1200M			1500M			1800M						비고
		7	8	9	7	8	9	7	8	9				
	AP	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5				● 1500M부터 Wi-Fi 자동갱신 되지 않는 것으로 보임
	감도	-61	-68	-75	-63	-67	-75	-60	-69	-75				
	수신	O	O	X	O	O	X	O	O	X				

거리		0M			300M			600M			900M			비고
시료 / AP		7	8	9	7	8	9	7	8	9	7	8	9	
4AP 세번째	AP	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	
	감도	-44	-31	-38	-47	-52	-57	-52	-59	-65	-60	-64	-71	● 900M 9번 시료 메시지 송부만 가능
	수신	O	O	O	O	O	O	O	O	O	O	O	X	
	위치	O	O	O	O	O	O	O	O	O	O	O	O	
	거리	1200M			1500M			1800M						비고
		7	8	9	7	8	9	7	8	9				
	AP	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5	E7:B5				
	감도	-60	-67	-73	-64	-67	-75	-61	-68	-75				● 1200M 부터 7, 8번 시료 본인 위치만 확인 가능
	수신	O	O	X	O	O	X	O	O	X				
	위치	9X	9X	O	9X	9X	O	9X	9X	O				

[표 158] Wi-Fi 거리별 시험 결과(3차)

○ 공항 내 설치된 모든 AP 구간의 음영지역 검증(8100Km까지 검증) 시험은 현장시험 3차에서 이루어졌으며, 지향성 안테나는 AP 앞에서는 성능이 좋으나 거리가 멀어질수록 다이폴 형태의 안테나 성능이 더 우수함을 확인함

※ 시험결과 적용내용은 "2장. 연구개발 수행 내용 및 결과, 2절. 연구수행과정 및 내용, 2. 연구개발 주요내용, 나. 공항시설 안전통합 관리시스템 전용단말기 하드웨어 개발, (5) 하드웨어 수정 보완 및 성능 안정화, (나) Wi-Fi 안테나 성능 개선" 참고

○ 시험 진행 시 안테나는 다음과 같이 왼쪽부터 다이폴 안테나, 지향성 안테나, 샤크 안테나2 순서로 장착함

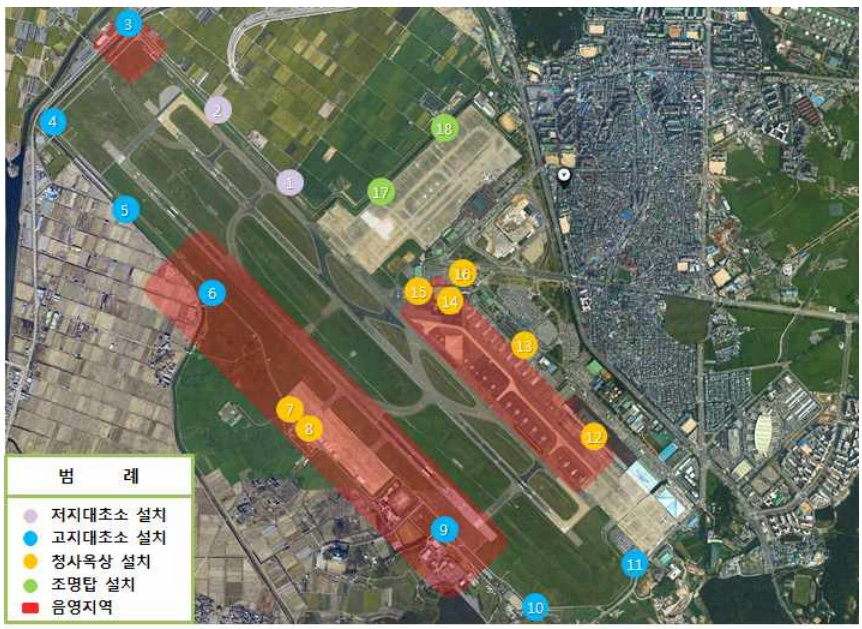


[그림 143] 안테나 장착(1)



[그림 144] 안테나 장착(2)

○ 음영지역 검증시험 결과는 다음과 같음



[그림 145] 공항 내 Wi-Fi 음영지역



[그림 146] 7-8번 AP 지점



[그림 147] 6-9번 AP 중간지점(1)



[그림 148] 6-9번 AP 중간지점(2)



[그림 149] 6-9번 AP 중간지점(3)



[그림 150] 7-9번 AP 지점

- 7-8번 AP 지점 : AP 안테나가 도로 반대쪽 방향을 보고 있으며 맞은편에 설치된 안테나는 건물에 가려져 신호 수신에 어려움이 있음
- 6-9번 AP 중간지점(1) : 6번 AP부터 9번 AP 중간 지점이며 7번 AP 가기전 코너 구간에 벽면으로 가려져 6번 AP 신호 수신에 어려움이 발생
- 6-9번 AP 중간지점(2) : 7-8번 AP에서 9번 AP로 가는 지점 코너 구간에 벽으로 막혀있음
- 6-9번 AP 중간지점(3) : 도로 안쪽의 건물은 도로보다 높은 곳에 위치해있으며 도로 맞은편의 신호 수신이 어려운 상황임
- 7-9번 AP 지점 : 9번 AP 가기 전에는 가로수들로 인해 9AP를 받기에 힘든 상황임

(다) GPS 성능 시험 (1~4차)

- 현장시험 1차 진행 시 응용프로그램에서 GPS 연결이 정상적이지 않은 문제 발생하여 GPS 외장 안테나를 이용한 추가적인 테스트 진행
- GPS Status Application을 이용한 수신감도 비교결과 단순한 인테나와 외장 안테나의 수신 위성개수 및 수신감도 차이 확인

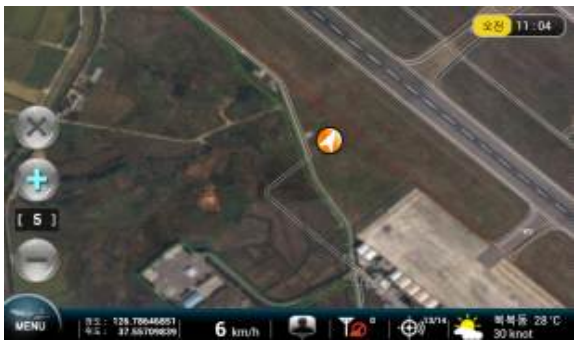


[그림 151] GPS 인테나 사용 시 수신감도

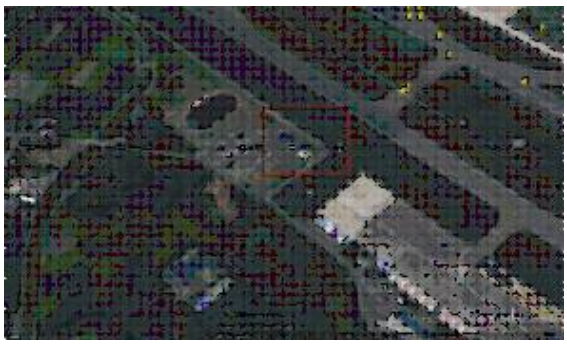


[그림 152] GPS 외장 안테나 사용 시 수신감도

○ 구글지도('12년 8월 31일 기준)와 지도화면 비교한 결과 현재 6 ~ 7번 AP 구역의 도로가 변경됨



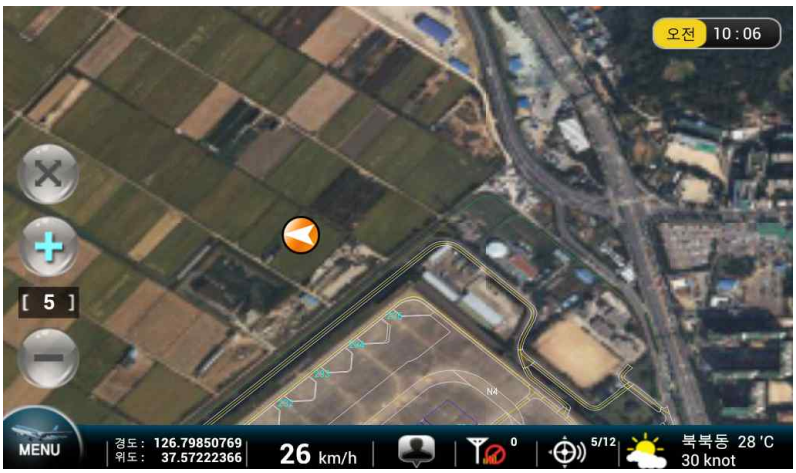
[그림 153] 현재 사용 중인 지도화면



[그림 154] '12년 8월 31일자 구글 지도화면

○ 다음의 내용은 현장시험 3차 진행 시 1차때 발생된 문제가 재발생 됨

○ GPS 유효위성 개수가 3개 이상일 경우에도 상당한 위치 오차 발생



[그림 155] 주행 시 GPS 오차 발생

○ Wi-Fi 거리시험 시 이동체 안전관리 서브시스템에서 확인된 펌웨어 업데이트 시료 2대와 GPS 외장 안테나 사용 시료 1대의 위치출력 차이가 다음과 같이 나타남



[그림 156] 주행 시 위치출력 차이



[그림 157] 정차 시 위치출력 차이

(라) Strix AP 점검 (1차)

○ 1차 시험 전 NS(Network Server) 상의 AP 상태를 점검함



[그림 158] AP 상태 확인

- Choso#3 : AP 고장 (확인 필요)
- Choso#9 : 1번 Card (무선 확장 테스트 시, 등록된 Card: 사용하지 않음)
- Gusobang#2 : 1번 Card (무선 확장 테스트 시, 등록된 Card: 사용하지 않음)
- Jomyung#1 : AP 고장 (확인 필요)
- Jomyung#2 : AP 고장 (확인 필요)

- Sinsobang, Sinsobang(ns) 2개의 AP는 Radius 인증 테스트를 위해 Hidden 해제, Radius 인증 적용한 상태로 테스트

- 그 외의 AP(고장 AP 제외)는 Hidden 설정, Radius 인증 해제 상태로 적용하여 Wi-Fi 테스트를 진행함

- 전용단말기 응용프로그램 기능 테스트를 진행 후, Sinsobang, Sinsobang(ns) 2개의 AP 또한 동일하게 Hidden 설정, Radius 인증 해제 상태로 적용하여 나머지 Wi-Fi 거리 테스트를 진행 완료

(마) 차량단말 서브시스템 기능동작 시험 (1~4차)

○ 기능동작 시험은 58번 시료와 디버깅 작업용 시료를 사용하여 진행함(총 2대)

○ 2대의 시료에 모두 다이폴 안테나를 연결함

○ 58번 시료에는 GPS 외장 안테나를 연결하였으며 나머지 1대는 인테나를 사용함

기능		시험횟수	동작상태	문제점
로그인		3회 이상	정상동작	Wi-Fi가 연결되어 있지만 네트워크 접속이 안 되어 로그인 실패 2회 발생
GPS		실시간	-	• GPS 외장 안테나를 연결한 58번 시료는 정상동작 하였으나 인테나를 사용한 전용단말기에서는 오차 및 수신감도 저하 발생 • GPS 오차로 인하여 적합하지 않은 위반이 간헐적으로 발생(외부도로에서의 기동지역 침범 등)
무선 네트워크		실시간	-	AP가 고장나지 않은 지역에서 연결은 정상적으로 되었으나 로그인 또는 메시지 전송기능이 제대로 수행되지 않는 경우 발생
수신 메시지 목록		1회 이상	정상동작	
FOD	등록	2회	정상동작	
	이미지 전송	2회	정상동작	
	유지보수 지시 확인	2회	정상동작	

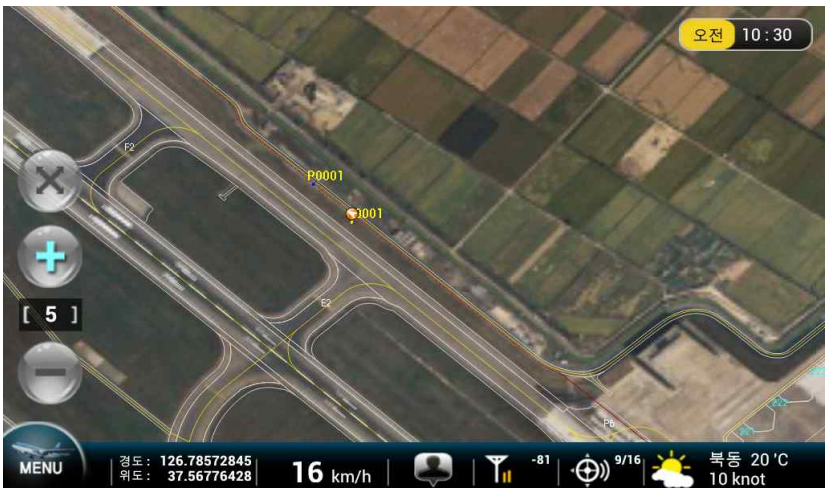
기능		시험횟수	동작상태	문제점
	조치완료	2회	정상동작	
사고	등록	1회	정상동작	• 수신된 상황전파, 출동지시 정보 중 인명피해 정보가 등록한 정보와 상이함 • 특이사항 입력 시 한글지원 안 됨
	출동요청 확인	1회	정상동작	
	지원요청	1회	정상동작	
	조치완료	1회	정상동작	
위반	속도위반 알림	1회 이상	정상동작	• GPS 연결 시 속도가 표시되지 않아 임의의 값을 설정하여 시험을 진행함 • 위반 알림 음성이 출력될 경우에만 상/하 버튼으로 음향 조정이 가능함
	속도위반 등록	1회 이상	정상동작	
	제방빙 금지구역 위반 알림	-		알고리즘 수정 필요
	제방빙 금지구역 위반 등록	-		알고리즘 수정 필요
	기동지역 침범 알림	1회 이상	정상동작	외부도로 주행 시 GPS 인테나 사용으로 발생되는 오차로 인해 의도치 않게 기동지역 침범 위반 확인이 가능하였음
	기동지역 침범 등록	1회 이상	정상동작	
	경고 확인	1회 이상	오류	이동체 안전관리 서브시스템에서 자동 또는 수동으로 전송한 경고 정보가 수신되지 않음
	출동지시 확인	1회 이상	정상동작	
	조치완료	1회 이상	정상동작	
제방빙	작업 시작 요청	1회	정상동작	
	작업 승인 확인	1회	정상동작	
	작업완료	1회	정상동작	
내 위치 출력		실시간	정상동작	
타차량 위치 출력		실시간	오류	이동 시 타차량이 표시되지 않는 경우 발생
격자 구역 출력		실시간	정상동작	
경도, 위도 출력		실시간	정상동작	

기능	시험횟수	동작상태	문제점
속도 출력	실시간	오류	0 km/h로 출력됨
설정	3회 이상	정상동작	

[표 159] 차량단말 서브시스템 기능 시험 결과

○ 내부 및 외곽도로를 주행하여 속도위반 기능동작 확인

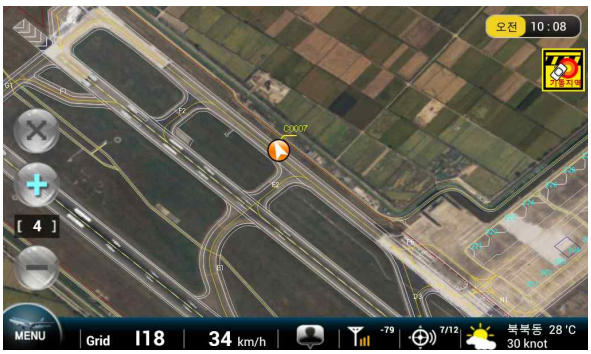
○ 외곽도로 주행 시 현재 차량의 속도와 출력되는 속도가 일치하지 않는 경우 발생



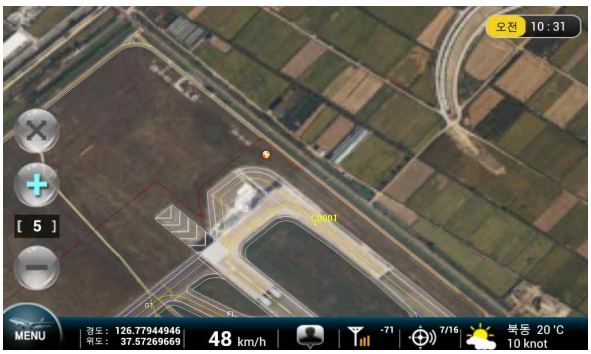
[그림 159] 58km/h 주행 시 속도출력 확인

○ 현장시험 2차 후 GPS 오차로 인하여 발생되었던 기동지역 침범 위반에 대한 위반판단을 위한 영역조정 후 현장시험 3차에서 재확인한 결과 업데이트된 GPS 펌웨어(Baudrate 9600) 사용시료에서 기동지역 침범위반 발생

○ 항공 이미지 출력 시 GPS 데이터의 처리의 CPU 부하문제로 판단되어 지도 데이터 처리 관련 최적화 예정



[그림 160] GPS 오차로 인한 기동지역 침범위반 발생



[그림 161] 주행 시 잔디밭에 차량위치

(바) 이동체 안전관리 서브시스템 기능동작 시험 (1~4차)

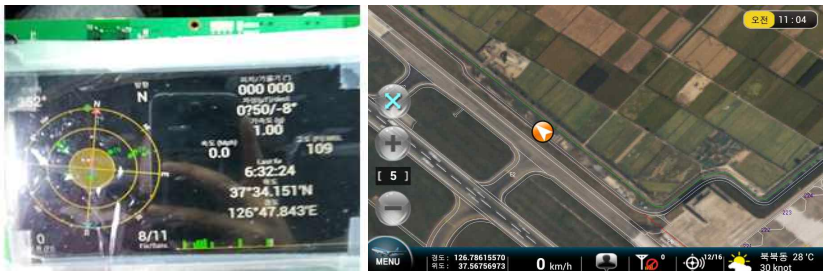
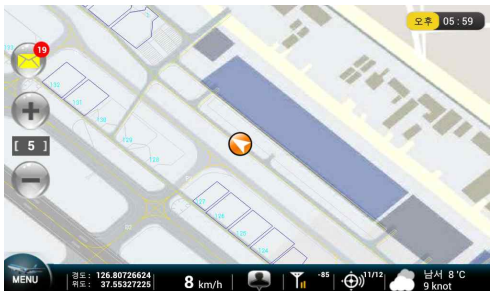
○ 차량단말 서브시스템으로부터 수신되는 사고, FOD 등록 정보의 시간 오류 발생(오전, 오후 시험에서 GPS로부터 수신한 시간정보는 정상적으로 출력되지만 서버로 전송된 시간은 오류가 발생)

○ 이동체 접속 끊겼을 경우 데이터베이스 서브시스템에 상태 업데이트 되지 않음(미들웨어 동작 확인 필요, 음영지역 발생 가능, 5-10초 비접속 차량 상태 업데이트 : 운영센터 담당 수정)

기능		시험횟수	동작상태	문제점
로그인		1회 이상	정상동작	
통신상태 확인		1회 이상	정상동작	
FOD	등록	2회	정상동작	네트워크 드라이브 경로 변경 필요 (기존 “FOD“폴더를 ”files“로 변경 필요)
	발생지점 확인	-	-	
	유지보수 지시	2회	정상동작	
	조치완료 확인	2회	정상동작	
사고	등록	1회	정상동작	
	발생지점 확인	-	-	
	출동요청	1회	정상동작	
	지원요청 확인	1회	정상동작	
	조치완료 확인	1회	정상동작	
위반	등록	1회 이상	정상동작	
	경고 전송	1회 이상	오류	자동 또는 수동으로 전송하였으나 전용단말기에서 수신되지 않음
	출동지시	1회 이상	정상동작	속도위반일 경우만 전용단말기에서 정상수신
	조치완료	1회 이상	정상동작	
제방빙	작업 요청 등록	1회	정상동작	
	작업 승인	1회	정상동작	
	작업완료 확인	1회	정상동작	
이동체 출력		실시간	정상동작	

[표 160] 이동체 안전관리 서브시스템 기능 시험 결과

○ 4차 현장시험을 진행하여 김포공항 내 테스트베드에 설치하기 전 문제점과 개선사항을 확인하고 반영함

항목	진행내용
Wi-Fi 운용성	
	샤크 안테나(좌)와 다이폴 안테나(우)를 사용하여 공항 내 설치된 AP에 연결 후 Wi-Fi 운용성을 검증함
GPS 동작 확인	
	GPS 상용 응용프로그램(좌)에서 GPS 유효 위성 개수와 수신감도를 확인하여 내장된 GPS 모듈의 동작 상태 확인 실제 전용단말기 응용프로그램에서 현재 위치에 대한 정보를 GPS로부터 수신하여 출력(우)
응용프로그램 운용성 확인	
	공항 내/외부 도로 운행 시 전용단말기의 응용프로그램을 통하여 현재 위치, 속도, 날씨 등의 정보 확인

항목	진행내용
운영센터의 기능동작 확인	
	공항 내 존재하는 항공기, 차량의 위치 모니터링 및 업무정보 송수신 기능 동작 확인

[표 161] 김포공항 내 전용단말기 및 운영센터 운용시험 시험 진행

나. GPS 및 Wi-Fi 성능시험

(1) 목적

- ○ 김포공항에 구축된 테스트베드를 통하여 정지상태에서의 전용단말기 GPS 위치정보의 흔들림 현상 해결 및 오차 보정 등을 위한 자체 시험 진행
- ○ 이스타항공 작업차량에 설치되어 있는 전용단말기의 Wi-Fi 커넥터와 선정리기 교체 후 카메라 장착(1차 시험 시)

(2) 범위

- ○ 전용단말기와 GPS용 액티브 안테나를 이용한 위치정보 오차율 확인 시험
- ○ 전용단말기의 GPS 펌웨어(속도제한 기능 추가) 교체 후 위치정보 오차율 확인 시험
- ○ 김포공항 내 AP 지역(강전계) 내 Wi-Fi 성능 시험

(3) 일시

- ○ 2014년 05월 19일(월) ~ 21일(수) (1차)
- ○ 2014년 07월 01일(화) ~ 04일(금) (2차)

(4) 시험장소

○ 김포공항 4~5번 AP 지역, 재해대책본부

(5) 시험내용

○ 고이득 GPS 외장 안테나 적용을 통한 정지 및 운행상태에서의 위치정보 흔들림 현상 보정

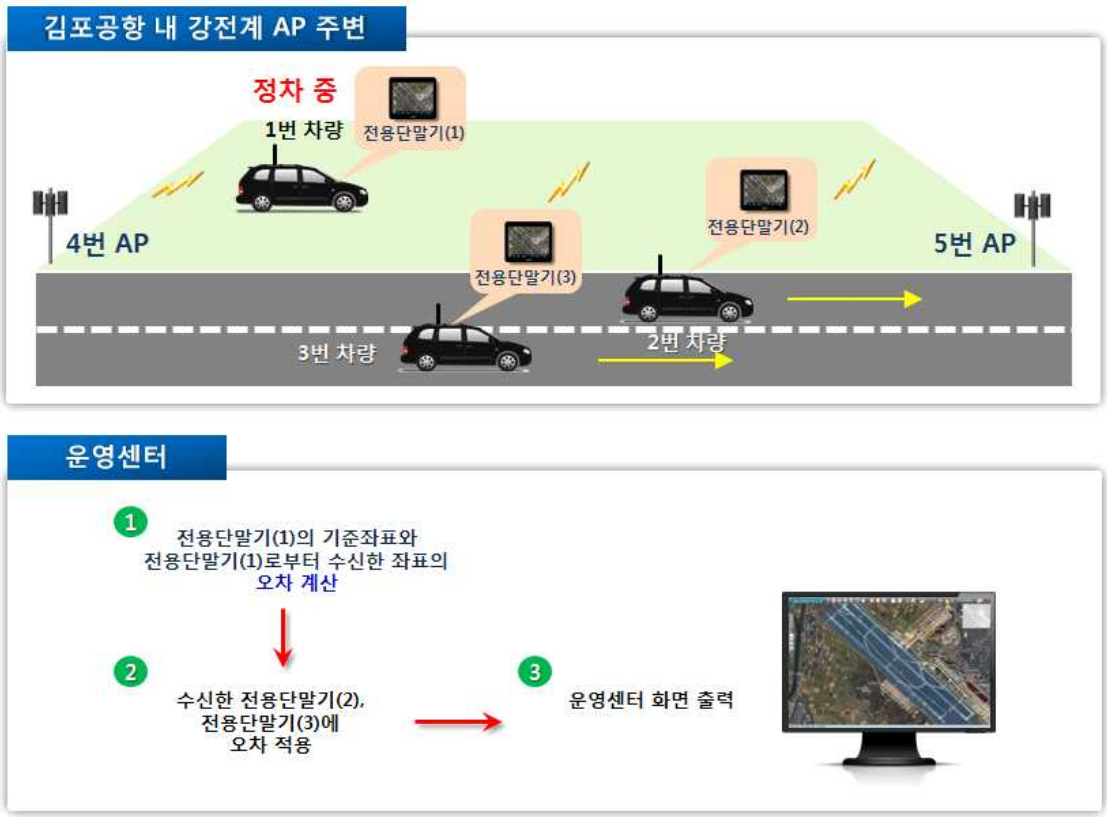
○ 전용단말기 응용프로그램에서 GPS 위치정보 전송주기 조절(500ms)

○ 운영센터에서 기준국 전용단말기의 오차를 통한 위치정보 보정(DGPS 기능 적용)

- 정지상태의 전용단말기(1) 1대의 위치를 기준좌표로 설정(GPS 기지국 역할)
- 전용단말기(1)에는 수신률이 최상인 GPS 외장 안테나 장착 후 운영센터로 위치정보 전송
- 운영센터에서는 전용단말기(1)의 기준좌표와 전용단말기(1)로부터 수신받은 좌표를 통해 오차 계산
- 운영센터에 수신되는 전용단말기(2), 전용단말기(3)의 좌표에 오차를 반영하여 보정 후
- 전용단말기(1) ~ 전용단말기(3)의 현재 위치 출력



[그림 162] 김포공항 테스트베드에서의 GPS 정확도 시험 진행 구성



[그림 163] GPS 위치정보 보정 시나리오

(6) 시험결과

(가) GPS 액티브 안테나를 장착하여 시험 및 데이터 수집(30km)

○ 차량간 일정거리를 유지하여 순차적으로 주행

○ 차량을 나란히 위치하여 주행

○ 차량간 서로 추월하여 순위를 변경하면서 주행

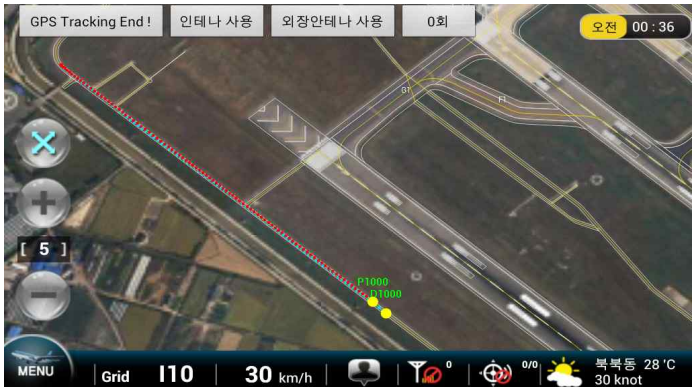
(나) 전용단말기 상태로 시험 및 데이터 수집(GPS 액티브 안테나 제거)

○ GPS 액티브 안테나를 사용한 단말기 대비 GPS 위성 수신수가 2~3개 적게 수신됨

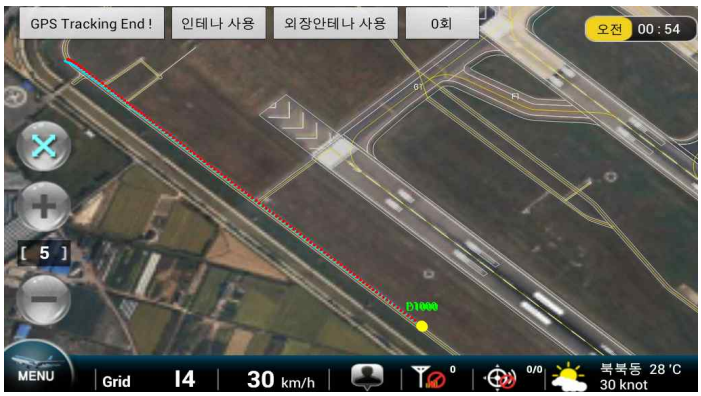
○ GPS 액티브 안테나 보다 오차가 많이 발생하여 시험에서 제외함

(다) 시험 진행 시 트래킹 화면

○ 일렬, 나란히, 추월 주행 시험 후 획득한 로그를 사용하여 출력한 트래킹 화면은 다음과 같음



[그림 164] 일렬로 주행 시 트래킹 화면



[그림 165] 나란히 주행 시 트래킹 화면



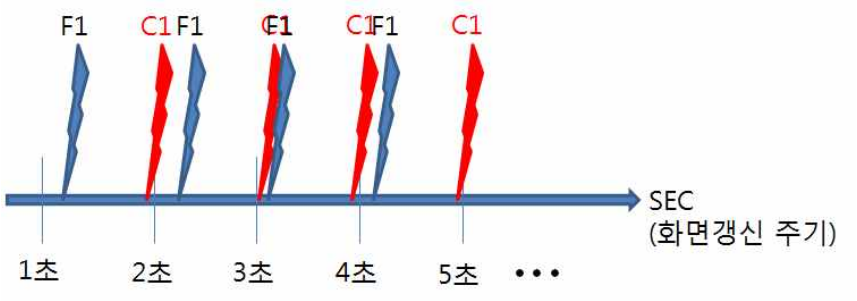
[그림 166] 추월 주행 시 트래킹 화면

○ 각 차량의 GPS 모듈의 위치좌표 갱신 주기가 상이하여 차량의 위치 오차가 발생하는 것으로 판단됨

○ 차량이 종으로 이동시에는 GPS 위치좌표 갱신주기가 동일하지 않아도 위치의 오차를 확인하기 어려움

○ 차량이 횡으로 이동시에는 GPS 위치좌표 갱신주기가 동일하지 않기 때문에 한 차량이 먼저 앞으로 이동하고 다른 차량이 따라붙는 것 같이 GPS 위치좌표 오차가 발생하는 것처럼 보여짐

○ 아래는 운영센터에서의 화면 갱신 주기와 단말기별 위치 정보 수신 타이밍을 그림으로 표현함



[그림 167] 전용단말기별 위치정보 수신 타이밍 예시

○ 차량의 위치 편차가 좌우로 발생을 하여 나란히 이동 중 차량의 위치 정보가 겹치는 현상이 발생함

○ 차량이 횡으로 이동시가 종으로 이동시보다 차량의 위치오차가 상대적으로 쉽게 확인됨

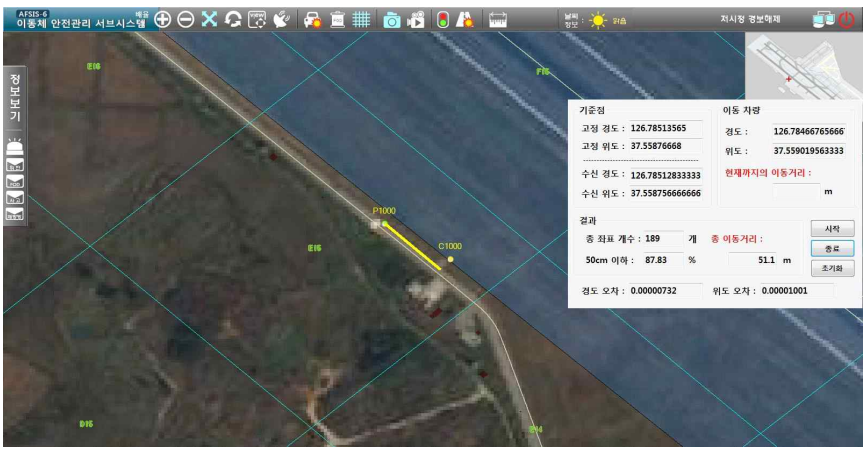
(라) DGPS 기능을 이용한 거리오차 보정 알고리즘 구현 후 시험

○ 시작점을 기준으로 차량을 이동하여, 각 지점별 위치 오차를 확인

○ 기준점에 GPS 장비를 위치하여, 기준점과 수신되는 GPS 위치 좌표의 오차를 계산

○ 시작점에서 차량을 50m 단위로 이동하여 위치보정 알고리즘 적용 값과 GPS 수신값을 계산하여 각각의 거리 오차를 계산

○ 시작점에서 50m, 100m, 150m 이동시 위치를 계산함



[그림 168] 운영센터에서의 DGPS 기능을 통한 차량 이동 화면 예시

○ 50m 이동 후 오차율 확인

구분	DGPS 위치 보정	위치보정 미적용
1회	51.10m	51.14m
2회	51.00m	51.03m
3회	52.10m	52.11m
평균오차	1.40m	1.43m

[표 162] 50m 이동 후 오차율 확인

○ 100m 이동 후 오차율 확인

구분	DGPS 위치 보정	위치보정 미적용
1회	104.30m	104.22m
2회	102.80m	102.84m
3회	102.90m	102.95m
평균오차	3.33m	3.34m

[표 163] 100m 이동 후 오차율 확인

○ 150m 이동 후 오차율 확인

구분	DGPS 위치 보정	위치보정 미적용
1회	154.00m	153.97m
2회	153.70m	153.67m
3회	153.10m	153.18m
평균오차	3.60m	3.61m

[표 164] 150m 이동 후 오차율 확인

다. 국토교통부와의 AFSIS-6 연구개발 검토 회의 및 시연

(1) 목적

- 국토교통부 안전과에서 공항시설 안전통합 관리시스템(AFSIS-6)에 대한 현장 확인 요청
- 김포공항 테스트베드를 활용하여 전용단말기와 운영센터에서의 차량, 항공기 모니터링에 대한 시연 진행

(2) 일시

- 시연준비 : 2014년 06월 03일(화) ~ 06월 12일(목) (7일간)
- 김포공항 현장 점검 : 2014년 06월 12일(목) (1일간)
- 국토부 시연 : 2014년 06월 13일(금) (1일간)

(3) 시험장소

- 김포공항 테스트베드
  - 차량운행 장소 : 재해대책본부 ~ 우리들병원 앞
  - 운영센터 설치 장소 : 재해대책본부



[그림 169] 김포공항 내 시연 진행 장소

(4) 국토교통부와의 회의 내용 및 결과

(가) 김포공항 설치 AFSIS-6 현황 검토

- 기술사업화 및 업무지원 등으로 지속적인 테스트 진행 중에 장비 교체와 성능 개량화가 요구됨
- 국토교통부 시연 시와 같이 AP 동작 불가로 인한 제한적인 운용되고 있음
- 실제 전용단말기를 사용 중인 한국공항공사 토목조경팀 외 5개 업체에 확인 결과 불편사항으로 AP 동작 불가에 따른 음영지역 다수 발생을 문제점으로 제시함
- AP 등의 실외 설치 장비의 경우 천재지변 등에 의한 고장이 지속적으로 발생되며 그에 따른 관리가 요구됨
- 전용단말기 및 운영센터에서 전시되는 항공기 정보의 경우 ASDE 장비에서 오류 데이터를 송신하므로 항공기의 위치 표시가 정상적으로 이루어지지 못함
- AFSIS-6의 시스템 개량기간동안 사용자들의 의견을 적극적으로 반영하여 완성도 높은 시스템이 되도록 현업부서에 노력 요청함
- 한국공항공사의 현업부서 중 AFSIS-6의 사용이 필요한 부서에서 지속적으로 사용할 수 있도록 조치가 필요함

(나) 테스트베드 활용 및 유지보수의 필요성 검토

- AFSIS-6 사업 종료 후 지속적인 관리 및 점검이 필연적으로 필요함
- ASDE 장비와 AFSIS-6 연동에 대한 검증 및 보완이 필요함

라. 국교통기술사업화지원사업 현장평가 진행

(1) 목적

○ 국토교통기술사업화지원사업(본 연구개발 과제) 현장평가(최종평가)를 위하여 김포공항 테스트베드에서 연구개발 성과물인 전용단말기의 기능 시험 및 평가를 준비함

(2) 일시

○ 현장평가 준비 : 2014년 08월 18일(월) ~ 19일(화) (2일간)

○ 현장평가 : 2014년 08월 20일(수) (1일간)

(3) 참석인원

○ 주관연구기관(인소팩(주)) : 연구개발 책임자 1명, 참여 연구원 5명

○ 국토교통과학기술진흥원 기술사업화 담당자 : 1명

○ 평가위원 : 1명

(4) 시험장소

○ 김포공항 테스트베드

- 차량운행 장소 : 5 ~ 6번 AP 지역
- 운영센터 설치 장소 : 재해대책본부



[그림 170] 본 연구개발과제의 현장평가(최종평가) 진행 장소

(5) 현장평가 진행 내용

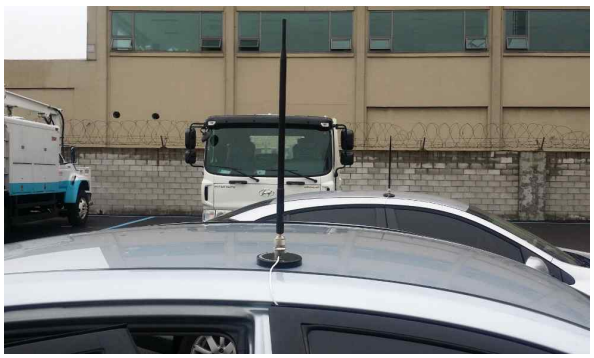
- 평가위원과 진흥원 인원을 대상으로 본 연구개발 사업의 개요, 연구개발 수행 내용, 전용단말기의 주요기능, 현장 적용 현황 등 소개 및 발표
- 연구개발 성과물(전용단말기) 관련 질의응답
- 3차년도 연구개발 내용 및 산출물 확인
- 전용단말기 주요 기능 시연



[그림 171] 차량 내 전용단말기 설치



[그림 173] 사업담당자에게 기능 설명(시연진행 중)



[그림 172] 무선 네트워크를 위한 안테나 설치



[그림 174] 전용단말기 설치 차량 운행(시연진행 중)

3절. 기술개발 성과의 국내외 시장전망

1. 목표시장의 정의 및 시장규모

가. 목표시장의 정의

(1) 시장규모

○ 현재 공항안전시설 관련 산업은 **고부가가치의 산업으로 분류되며 시스템은 첨단기술을 적용한 완제품 형태로 납품**됨

○ 항행안전 시장은 초기 대규모의 연구개발 및 시설에 대한 투자와 수익창출에 오랜 시간이 필요하므로 후발주자의 시장 진입이 어려운 실정이며 국내뿐만 아니라 전 **세계적으로 소수의 기업이 시장을 주도**하고 있음

○ 국내의 경우 **항공법 시행 규칙에 따라 제 10조법 제2조17호**에 따른 **공항안전시설을 분류하여 관리**하고 있음

○ 국내 공항별 항행안전시설 관련 현황은 다음과 같으며 확대 설치가 가능하다고 판단됨

구 분	LLZ/DME (거리측정시설)	ASDE (레이더)	VOR (전방향표지시설)	TACAN (전술항행표시설)
인천공항	6	2	2	-
김포공항	4	1	1	-
김해공항	2	1	1	-
제주공항	2	-	1	-
청주공항	1	-	1	1
양양공항	1	-	1	-
대구공항	1	-	1	1
광주공항	1	-	2	1
원주공항	-	-	1	1
예천공항	1	-	-	1
포항공항	1	-	1	1
울산공항	1	-	1	-

구 분	LLZ/DME (거리측정시설)	ASDE (레이더)	VOR (전방향표지시설)	TACAN (전술항행표시설)
사천공항	2	-	1	1
여수공항	1	-	1	-
군산공항	-	-	1	-
목포공항	1	-	1	-
합 계	25	1	17	7

[표 165] 항행안전시설의 국내 설치현황

※ 항행안전시설관련산업 활성화 용역 최종보고서_국토교통부

○ 국외의 경우 ICAO(국제민간항공기구)를 중심으로 급증하는 항공교통 수요에 대비하고자 정기적인 항행 관련 회의를 개최하고 미래의 항행 시스템에 대한 기술의 채택 및 연구를 진행하고 있음

○ 국가별로 지속적인 공항안전시설 제작 산업의 활성화 및 선진화 방안을 연구하고 있음

○ 또한 ICAO(국제민간항공기구)에서는 CNS/ATM(Communication, Navigation, Surveillance/Air Traffic Management) 아래 신개념의 항행 지원시스템 구축을 진행하고 있음

○ 국제적으로 공항안전시설 관련 장비 및 설치장소는 다음과 같음

권역별 공항 (ICAO 기준)	장비설치장소	LLZ/DME (거리측정시설)	VOR (전방향표지시설)	TACAN (전술항행표시설)
아프리카·인도양	974	236	69	2
아시아·태평양	2,032	650	60	1
카리브해·남미	1,246	413	49	-
유럽	2,840	702	69	80
중동아시아	346	120	3	-

권역별 공항 (ICAO 기준)	장비설치장소	LLZ/DME (거리측정시설)	VOR (전방향표지시설)	TACAN (전술항행표시설)
북미	3,652	520	60	70
기타	45	11	1	1
합 계	11,135	2,652	311	154

[표 166] 항행안전시설의 국외 설치현황

※ 참고 : 항행안전시설관련산업 활성화 용역 최종보고서_국토교통부

○ 국내 16개 공항, 국외 4개 공항 등에서 공항시설 안전통합 관리시스템에 관심을 보이고 있으며, 향후 공항시설 안전통합 관리시스템의 구축 완료와 동시에 약 1700여개 이상의 전 세계 공항이 사업의 시장임

○ 세계 1700여 공항 협의체인 ACI의 중요정책을 결정하는 최고의사결정기구인 ACI 세계이사회 임원으로 인천국제공항공사 사장이 임기하고 있어, 세계 시장 개척에 큰 도움이 될 것으로 예상됨



[그림 175] 국내 시장규모



[그림 176] 국외 시장규모(예상)

(2) 기술수요

- 국내 정보통신부, 산업자원부 등 정부기관을 중심으로 정보화추진전략 수립 및 시범사업 등의 시행으로 IT 기술 도입을 유도하고 있음
- 항공분야는 여타 운송 분야(고속버스, 고속철도 등)와 비교하여 보안 및 안전 분야가 더욱 강조되고 있어 신기술의 필요성이 부각되고, 신기술에 대한 수요가 늘어나는 추세임
- 아시아 태평양 지역에서 항공교통관리의 주도권 경쟁이 나타나고 있고, 아시아 태평양 지역은 중국의 급성장과 동남아 국가들의 발전으로 인해 타 지역에 비해 항공교통량의 증대가 상대적으로 높으며, 한중일 인접국의 영공방어 및 정치적 논리에 의해 자국의 항공운송 편의성을 극대화하는 정책을 추구하려 함
- 인천공항, 일본의 나리타공항, 중국의 푸동공항은 모두 동북아 허브공항이라는 목표를 가지고 시설확장, 공항이용료 경쟁, 서비스 제공 수준 변화 등의 전략을 가지고 항공노선 및 항공수요 유치에 박차를 가하고 있어 공항시설 안전통합 관리시스템의 존재는 경쟁에 중요한 역할을 할 수 있음
- 현재 우리나라가 ICAO 상임이사국임에도 불구하고, 기술력 부족으로 항공 선진국이 주도하고 있는 국제표준 제정 등 각종 국제회의에 주체가 되지 못하였으나, 공항시설 안전관리 통합시스템의 기술 확보를 계기로 선도적인 국가로 성장할 수 있음
- 공항시설 안전관리 통합시스템을 필두로 하여 항공교통분야 국제표준화 활동의 적극적인 참여를 통해 시장 선도 기회를 모색하고, ICAO 회원국들의 관심으로 세계 항공 표준화의 가능성이 높아져 세계적인 수요가 증가 할 것으로 예상됨

나. 예상 시장규모

○ 공항시설의 안전관리에 대한 관심이 세계적으로 높아지는 추세

- KOTRA의 「글로벌 공항조달 플라자 2013」 행사를 통해 해외 5개 바이어와 수출 상담 진행, KOICA기금 활용, 대기업과의 공동 수출 추진

- AUSA, ADEX 등의 전시회를 통하여 시스템 기반의 수출 필요성을 확인

○ 「공항시설 안전통합 관리시스템 전용단말기」 개발을 통해서 향후 해외시장 진출 시 비용에 의한 진입장벽을 최소화 및 부가가치의 극대화를 기대함

(1) 국내 주요 예상 판매처 현황

판매처	국가 명	판매 단가 (천원)	예상 연간 판매량(Set)	예상 판매기간(년)	예상 총판매금 (천원)	관련제품
인천국제공항	한국	2,900,000	1	1	2,900,000	차량단말 서브시스템, 이동체 안전관리 서브시스템, 미들웨어 서브시스템
제주국제공항	한국	1,215,000	1	1	1,215,000	
김해국제공항	한국	1,849,000	1	1	1,849,000	
청주국제공항	한국	1,371,000	1	1	1,371,000	
대구국제공항	한국	1,205,000	1	1	1,205,000	
양양국제공항	한국	1,175,000	1	1	1,175,000	
무안국제공항	한국	1,184,000	1	1	1,184,000	
군산공항	한국	693,000	1	1	693,000	
여수공항	한국	776,000	1	1	776,000	
포항공항	한국	1,401,000	1	1	1,401,000	
울산공항	한국	895,000	1	1	895,000	
원주공항	한국	1,121,000	1	1	1,121,000	
사천공항	한국	1,302,000	1	1	1,302,000	
김포공항	한국	2,087,000	1	1	2,087,000	

[표 167] 기술개발 후 국내·주요 판매처 현황

○ 전용단말기 가격은 하기와 같이 개발하는데 소요되는 비용을 기준으로 책정

- 운영센터 기반 솔루션 구축, 단말기 제작, 시험 및 표준화 추진, 이동체 안전 연구 및 보급 활성화 등에 따른 비용
- 설계비 : 메인 시스템 설계, 단말기 제작을 위한 설계 등의 추진 비용
- 시스템 구축비 : 시스템 개발 시 투입 인력에 대한 용역비, 하드웨어 시스템 도입비, 소프트웨어 도입 비용
- 단말기 제작·시험 및 표준화 추진에 따른 비용 : 전용단말기 실시 설계안에 따라 단말기를 제작하고, 위치정보 제공 및 데이터 송수신 솔루션을 탑재하여 시험을 통한 단말기 표준화 추진에 소요 되는 비용
- 이동체 안전 연구 및 보급 활성화에 따른 비용 : 전용단말기 사용자에 대한 보급 및 활용도 강화를 위해 추진되는 정책적인 비용으로 항공기 및 이동체에 대한 안전 기반 구현을 위해 필요한 기술 연구 및 정보개발 활동에 소요

(2) 국외 주요 예상 판매처 현황

판매처	국가 명	판매 단가 (천원)	예상 연간 판매량(Set)	예상 판매기간(년)	예상 총판매금 (천원)	관련제품
상파울로 국제공항	브라질	2,900,000	10	1	29,000,000	
마닐라 공항	필리핀	2,900,000	1	1	2,900,000	
자카르타 공항	인도네시아	1,500,000	1	1	1,500,000	
모나코 공항	모나코	1,000,000	1	1	1,000,000	

[표 168] 기술개발 후 국외 주요 판매처 현황

2. 현장적용 가능성

가. 네트워크 인프라 측면

○ 공항시설 안전통합 관리시스템의 테스트베드인 김포공항을 예시로 타 공항의 현장적용 가능성을 명시함

○ 김포공항의 부지면적은 8,635,937㎡로 네트워크 인프라 구축을 위해서 15개의 무선 AP를 설치하여 공항의 모든 지역이 커버되도록 설계됨



[그림 177] 김포공항 무선 네트워크 커버리지 구성도

○ 테스트베드에서 전용단말기와 운영센터 간의 패킷전송을 위해 사용되는 Wi-Fi 대역은 2.4GHz(IEEE802.11b/g의 무선 표준)

○ 5.8GHz 주파수는 AFSIS-6 초기 연구개발 시 백홀 통신용으로 지정되어 주파수간의 간섭 방지를 위하여 클라이언트 통신용과 분리됨

○ 전용단말기는 클라이언트 통신에 해당되므로 설계 초기부터 백홀 통신과 구분되어 2.4GHz 주파수 대역으로 설정

주파수(GHz)	적용 AP 수량	용도	비고
2.4	20	클라이언트 통신	전용단말기 통신
5.8	13	백홀 통신	중앙센터 통신

[표 169] 공항 내 AP 수량 및 용도

○ 클라이언트의 경우 김포공항 전 지역 통신이 목적이기 때문에 통달거리가 넓은 주파수 대역 설정이 필요

○ 주파수 특성 상 5.8GHz 대역 보다 2.4GHz 대역이 통달거리가 더 넓음(하타 전파분석모델을 적용하여 주파수에 따른 전파 통달거리 계산 예시)

구분	내용	
주파수(GHz)	2.4	5.8
거리(km)	1.1	0.5

[표 170] 주파수별 안테나 통달거리 하타 시뮬레이션

○ 주파수가 높을수록 데이터량은 많아지지만 회절성이 부족하여 통달거리가 좁아짐

○ 클라이언트의 데이터는 많은 데이터량이 필요하지 않기 때문에 높은 대역의 주파수가 불필요

○ 김포공항의 경우 주변 주택가 또는 건물과 이격되어 설계되었기 때문에 10mW 이내의 상용 Wi-Fi 주파수 출력에 대한 영향은 미비

○ 자체 암호화로 IEEE802.11i, AES 및 WEP 등을 지원하며 별도의 보안인증 포함

○ 국가정보 관련 기관에서 보안성에 대한 평가를 받음

○ 김포공항의 규모는 중·대형급 공항으로, 국내에서는 인천공항을 제외한 모든 공항들이 김포공항보다 작은 규모임

○ 15개의 무선 AP 설계 및 구축을 통해 김포공항의 무선 네트워크를 구성한 것과 같이, 다른 공항들은 적은 무선 AP 설계와 구축으로도 충분히 네트워크 인프라를 구축할 수 있음

공항명	면적			
	부지면적(㎡)	계류장면적(㎡)	가로	세로
김포공항	8,635,937	1,215,487	1.94km	4.57km
김해공항	6,518,572	389,358	802m	3.87km
	3,946,688			
제주공항	3,561,679	384,458	3.12km	400m
				2km
대구공항	6,617,283	41,582	360m	159m
	171,308			
울산공항	919,977	33,605	512m	2.5km

공항명	면적			
	부지면적(㎡)	계류장면적(㎡)	가로	세로
청주공항	6,739,778 1,909,645	91,047	683m	140m
무안공항	2,585,733	90,692	610m	3.49km
광주공항	5,854,564 150,599	44,300	540m	2.94km
여수공항	1,327,834	41,868	177m	302m
포항공항	4,035,563 2,478,709	32,617	2.35km	610m
양양공항	2,448,500	45,250	772m	3.24km
사천공항	4,039,465 45,299	13,140	3.23km	720m
군산공항	- 142,803	13,758	-	-
원주공항	5,675,650 31,960	6,741	180m	3.31km
인천공항	-	-	3.35km	5.72km

[표 171] 국내 공항 면적 비교 정보

○ 타 공항 적용 시 기구축 네트워크 인프라의 사용이 가능할 경우 네트워크에 따라 전용단말기의 통신 모듈 및 방식 변경 가능

- Wi-Fi 통신

  • Wi-Fi 통신 모듈 사용 가능

  • Wi-Fi 모듈 사이즈 : 13.1(W)mm * 6.1(D)mm *1.35(H)mm

  • 음성통화와 데이터통신 분리

  • 유선 구축이 어려운 장소나 유선 대비 가격 경쟁력 확보 가능

  • 초기 투자비용 기지국 설치에 비해 상당히 저렴하고 무선 데이터망 구축 가능

- Wi-Max 통신

  • Wi-Max 통신 모듈 사용 가능

- Wi-Max 모듈 사이즈 : 31(W)mm * 22(D)mm *3.8(H)mm
- 무선랜 서비스 반경보다 넓고 빠름
- 자동차등 교통수단을 이용해 빠른 속도로 움직이면서도 빠른 속도록 인터넷 사용 가능
- 휴대 단말기 활용도 높임

- LTE 통신

- LTE 통신 모듈 사용 가능
- LTE 통신 모듈 사이즈 : 36.5(W)mm * 28.5(D)mm *2.3(H)mm
- WCDMA, CDMA 기반의 대부분의 이통사가 채택
- 3G 망과의 연동 용이
- 빠른 전송 속도로 다양한 서비스 사용 가능
- 망 투자 비용을 줄일 수 있고 서비스 지역을 넓음

통신 모듈		
Wi-Fi 모듈	Wi-Max	LTE
		

[표 172] 통신 방식에 따른 통신 모듈

나. 서버 적용 측면

○ 공항시설 안전통합 관리시스템의 서버는 김포공항 내 전산실에 설치가 되어 있다. 여러 서버와 함께 있어 관리 및 보안이 용이하고, 기존 공항 서버와의 연계가 편하다는 장점이 있음

○ 공항의 내부네트워크 망을 이용하여 운영센터의 네트워크를 구축해야하고, 공항 내 이동체를 관리하기 용이한 곳에 운영센터를 설치해야 하기 때문에 아래의 그림과 같이 김포공항과 유사한 위치에 서버와 운영센터를 설치할 수 있다면 타 공항의 현장적용에 무리가 없을 것으로 예상됨



[그림 178] 김포공항 내 서버 및 운영센터 설치 위치

다. 공항시설 안전통합 관리시스템 전용단말기 측면

○ 본 사업 완료 후, 공항시설 안전통합 관리시스템 전용단말기 개발과 프로그램 유지보수 시스템 개발이 완료되어 김포공항 테스트베드에 적용할 예정임

○ 기존에는 조업차량에 상용 차량 단말기 최초 설치 후, 유지보수 방안이 없어 모든 상용 차량 단말기 수거 또는 현장 수동 작업을 통해 프로그램 업데이트를 진행해야 했음

○ 김포공항의 경우, 약 2,000대, 인천공항은 약 20,000대에 달하는 조업차량의 상용 차량 단말기를 모두 수동 업데이트해야하기 때문에 사실상 현장적용이 어려운 문제점을 가짐

○ 공항시설 안전통합 관리시스템 전용단말기 개발과 프로그램 유지보수 시스템 개발을

통해 현장적용 가능성은 매우 높아진다. 공항시설 안전통합 관리시스템 전용단말기를 개발하여 기존 상용 차량 단말기보다 크기가 작고 가벼워 조업차량에 최초 설치 시간이 줄어듦

○ 프로그램 유지보수 시스템 개발을 통해 추후 프로그램 업데이트를 수동이 아닌 네트워크 인프라를 이용한 자동 업데이트가 되기 때문에 실질적으로 현장적용에 따른 유지보수가 가능함

○ 중대형 공항인 김포공항 테스트베드에 공항시설 안전통합 관리시스템 전용단말기와 프로그램 유지보수 시스템이 적용된다면, 국내의 타 공항에 위 시스템 현장적용이 가능함

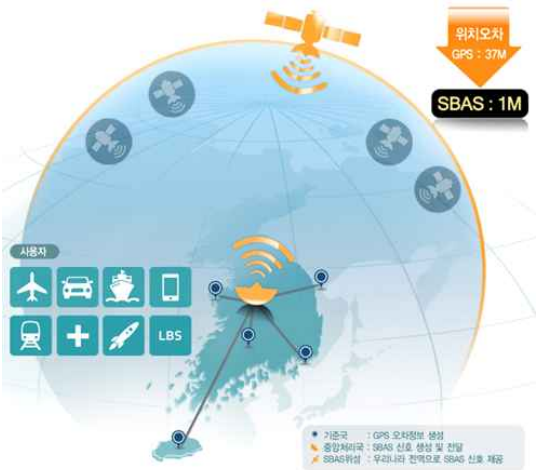
○ 아래 표는 김포공항에 공항시설 안전통합 관리시스템 전용단말기 설치를 완료 했을 시, 규모에 따른 타 공항의 현장적용 가능성을 표출하기 위해 작성됨

공항별		조업차량 대수	공항 면적 비율(%)	비고
대형거점 공항	김포공항	2000	100	기준값
	김해공항	2424	121	현장적용 가능
	제주공항	825	41	현장적용 가능
	인천공항	20000	222	추가 설계 필요
소형거점 공항	대구공항	1573	79	현장적용 가능
	무안공항	599	30	현장적용 가능
	양양공항	568	28	현장적용 가능
	여수공항	309	15	현장적용 가능
	울산공항	214	11	현장적용 가능
	청주공항	800	40	현장적용 가능
	광주공항	2400	120	현장적용 가능
일반 공항	사천공항	946	47	현장적용 가능
	원주공항	1322	66	현장적용 가능
	포항공항	1509	75	현장적용 가능
	군산공항	400	20	현장적용 가능

[표 173] 공항별 면적 대비 조업차량 대수 비교

○ 공항 내 이동체(항공기, 차량)의 정확한 위치정보가 안전관리의 핵심이므로 단일 GPS를 사용할 경우 정확도가 낮아질 수 있음

○ 타 공항에 시스템 적용 시 정밀단독측위(PPP, Precise Point Positioning) 알고리즘을 적용하거나 '18년도에 시범서비스가 진행될 실시간 1m 이내 정밀 위치정보를 제공해주는 SBAS 등을 이용하여 GPS 정확도 보완 예정

기술	내용
DGPS (Differential Global Positioning System)	• 위성항법보정시스템 • GPS의 오차를 최소화하기 위하여 육상에 GPS 오차보정용 국을 따로 설치하여 운영 • 기준국의 정확한 위치값과 GPS 위치값을 비교하여 실시간으로 사용자에게 오차 제공(통신방향 : 기준국 → 사용자) • 팔미도와 어청도 등에 설치되어 우리나라의 전 해역을 11개국으로 이중커버하고 있음
IDGPS (Inverted Differential Global Positioning System)	• DGPS와 동일한 정확도, 특정 트래킹 작업의 비용절감위치 보정 • 이동 사용자는 자신의 대략적인 위치를 관제국에 전송, 관제국에서는 DGPS 기준국에서 계산한 보정정보를 이용해 보다 정확한 사용자의 위치를 산출(통신방향 : 사용자 → 기준국)
SBAS (Satellite Based Augmentation System	• 초정밀 GPS 보정 시스템 • 국내에서는 '18년부터 시범서비스 예정으로 GPS 위치오차를 실시간 1m 이내로 정확하게 보정하여, 정지궤도 위성을 통해 전 국토에 제공 • 인공위성을 통한 서비스 제공으로 이용에 제한이 없어 항공, 해양, 교통, 정보통신, 물류, 응급구조 등 모든 분야 활용이 가능 • 미국, 유럽, 일본에서 개발완료 되었으며 러시아, 인도에서는 개발 진행 중 

[표 174] GPS 오차 보정을 위한 관련 시스템

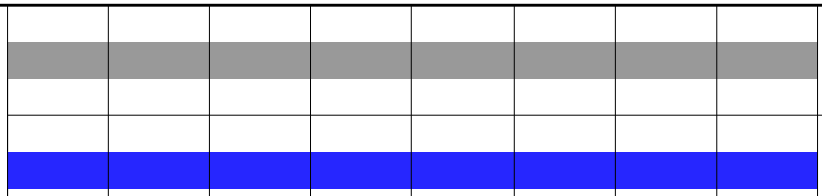
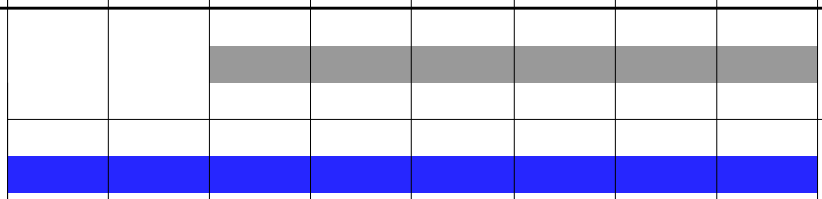
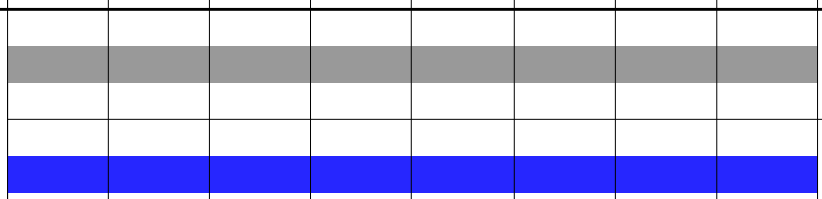
3. 기술 수요 및 추가 예상 성과

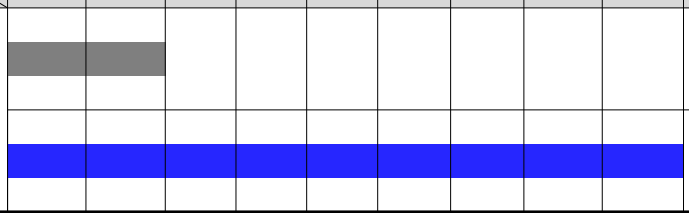
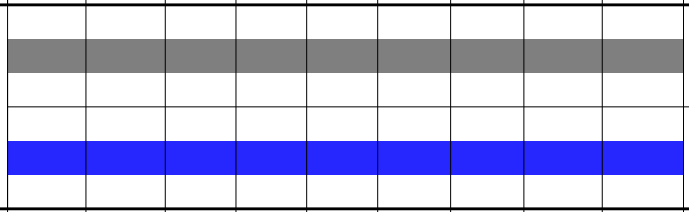
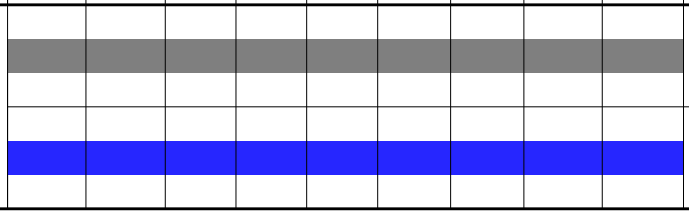
- 현재 김포공항에서 시험 운영 중이며 지속적인 보완과 후속 조치를 통하여 국내 다른 공항으로 확장을 생각하여 진행함
- 공항 시설안전 통합관리 기반기술 개발은 기존 각 단위 시스템별로 관리 프로세스 통합에 따른 안전한 공항 이미지 부각이 예상됨
- Airside 지역 이동차량 관리 시스템은 공항 Airside 지역에 적용 가능한 기술로서는 세계 최초의 연구개발에 속하므로 타 공항의 적용이 가능하며, 타 유사한 성격의 이동체 관리 시스템에 적용이 용이함
- 본 기술의 점유 가능성은 ICAO에 Airside 안전관리 표준화도 가능하며 기존 국내외 공항 및 세계시장을 대상으로 진출이 가능 할 것으로 예상됨
- 비정상적인 상황에 대한 안전체계를 강화함으로써 공항의 가동률 향상에 기여하는 기술적 토대를 마련함
- 해외 주요 선진국에 치우친 기술 의존도를 해소하고 우리나라 공항에 가장 적합한 공항 안전기술을 창출함
- 공항은 도로, 철도, 항만 등의 교통 인프라시설과 유사한 환경적 특성이 있으므로 본 기술의 향후 확대가 용이할 것으로 예상됨
- 공항 내 다양한 전파의 사용에 대한 간섭 및 상호충돌의 가능성을 검토하고 Airside와 같은 광대역 지역에 대한 무선 네트워크 적용에 대한 기술적 검증을 실시함으로써 u-City 등과 같은 유비쿼터스 환경 하에서 무선 네트워크 시스템 구현을 위한 기술적 배경을 제공할 수 있음

4장. 연구목표 달성 및 효과

1절. 연구개발 최종목표 달성도

구 분 / 개발내용	1차년도 연구개발 기간								진도 (%)
	1	2	3	4	5	6	7	8	
요구사항 도출 및 설계 구축									계획 (100%)
									실적 (100%)
주요부품 선정 및 확보									계획 (100%)
									실적 (100%)
시제품 디자인 확보									계획 (100%)
									실적 (100%)
개발보드 설계 및 구현 후 각 모듈별 기능 검증									계획 (100%)
									실적 (100%)
시제 제작 및 검증									계획 (100%)
									실적 (100%)
운영체제 포팅 및 검증									계획 (100%)
									실적 (100%)

GUI 개발 및 기능구현		계획 (80%) 실적 (100%)
전체 메뉴 구성 작성		계획 (90%) 실적 (100%)
테스트베드 관리 및 정책적 지원		계획 (100%) 실적 (100%)
공항시설 안전통합 관리시스템 (AFSIS-6) 관련 지침 현황 분석 및 문제점 도출		계획 (100%) 실적 (100%)

구 분 / 개발내용	2차년도 연구개발 기간									진도 (%)
	1	2	3	4	5	6	7	8	9	
전용단말기 기구디자인 및 금형제작										계획 (100%) 실적 (100%)
GUI 개발 및 기능 최적화										계획 (100%) 실적 (100%)
GIS 엔진 개발 및 최적화										계획 (100%) 실적 (100%)

전용단말기 하드웨어 개발 및 수정보완										계획 (100%)
										실적 (100%)
테스트베드 환경구축 및 운용시험										계획 (100%)
										실적 (100%)
구 분 / 개발내용	3차년도 연구개발 기간									실적 (%)
	1	2	3	4	5	6				
GUI 개발 및 기능 최적화 (계속)										계획 (100%)
										실적 (100%)
GIS 엔진 개발 및 최적화 (계속)										계획 (100%)
										실적 (100%)
전용단말기 기구 수정보완										계획 (100%)
										실적 (100%)
운용시험										계획 (100%) 실적 (100%)
보고서 작성										계획 (100%)
										실적 (100%)

[표 175] 연구개발 수행 진도율(연차별)

2절. 연구개발 성과의 효과분석

1. 기술적 효과분석

- 공항시설 안전통합 관리시스템 전용단말기 개발을 통하여 공항의 항공안전 구현을 위한 공항시설의 효율적, 경제적 안전관리에 활용
- 기존의 고비용 라이선스를 사용하는 소프트웨어로 구성된 관리시스템 장비에서 라이선스 비용을 지불하지 않고 순수 국산 하드웨어와 소프트웨어 기술이 적용된 공항시설 안전통합 관리시스템 전용단말기의 양산에 기초를 다짐
- AFSIS-6는 공항 내 Airside에 적용 가능한 기술로써 세계 최초의 연구개발에 속하므로 공항뿐만 아니라 타 유사한 성격을 가진 이동체 관리 시스템에 적용이 용이함
- ICAO에 Airside 안전관리 표준화도 가능하며 기존 국내 공항 외에도 세계 공항을 대상으로 진출이 가능 할 것으로 예상
- 항공기 사고 등 비정상적인 상황에 대한 안전체계를 강화함으로써 공항의 가동률 향상에 기여하는 기술적 토대를 마련
- 해외 주요 선진국에 치우친 기술 의존도를 해소하고 국내 공항에 가장 적합한 공항 안전기술을 창출
- 동남아시아 등 후발국가의 신규 공항 또는 국내의 소규모 공항 건설 시 축적된 기술력을 바탕으로 설계 및 컨설팅에 참여하여 새로운 부가가치 창출
- 개발도상국에 공항과 관련된 첨단기술의 수출 가능
- 중단 없는 공항 운영을 위한 주요 시설물의 안정성 확보, 감시 및 관리 기술, Airside 보안관련 기술의 향상

2. 경제적 효과분석

- 본 연구개발을 진행하여 차량에 설치하는 공항 전용단말기의 비용을 약 100만원으로 절감

구분	AFSIS-6 개발 시점	기술사업화 완료 시점
차량단말기 가격	**265만원**	**100만원**
단말기 하드웨어	100만원	90만원

구분	AFSIS-6 개발 시점	기술사업화 완료 시점
운영체제 라이센스	하드웨어에 포함	무료
GIS 엔진 라이센스	100만원	무료
외장 랜카드	40만원	하드웨어에 포함
외장 안테나	20만원	10만원
GPS 모듈	5만원	하드웨어에 포함

[표 176] AFSIS-6의 차량 설치용 단말기 비용 절감

○ 공항시설의 유지관리 능력 및 안전성 확보로 공항서비스 세계 경쟁력 1위를 유지함으로써 국가브랜드의 가치 제고 및 공항 이용객에게 양질의 서비스 제공

○ 공항시설의 안전성 증대로 항공기 운항의 정시성이 향상되고, 이로 인해 항공기 운송수요의 증가 및 잠재적 사고 비용 하락

○ 공항 설계 및 공항시설의 운영·유지관리 기술의 축적으로 관련 기술의 해외 진출 기회가 증대되고 국내 기술 수준의 국제적 위상 제고

○ 현재의 운영 상황보다 좀 더 효율적이고 체계적인 인력과 장비에 대한 관리 가능 및 비용절감 등의 긍정적인 경제적 효과 기대

○ 우리나라의 경우, 20년간(1980 ~ 1999)의 항공사고는 45건으로 이로 인한 총 비용은 1조 1,150억원으로 항공기 사고 1건을 예방할 수 있다면 약 250억 이상의 직/간접비용을 절감할 수 있음

구분	20년 총 비용	건당 평균비용	비고
직접비용	6,567억	147억	• 근거 :"항공사고비용의 산정과 국제 간 비교연구"(한국교통연구원,2000) 1970년에서 2002년까지 총 사고 76건 발생에 피해(직접비용)는 7,541억으로 건당 약 100억의 피해비용 발생(국토교통부 통계)
간접비용	4,583억	100억	
총 비용	1조 1,150억	247억	

[표 177] 안전사고 예방 시 경제적 기대효과

○ Airside 차량, 장비 통제 시스템 구축에 따른 잠재적 교통사고 감소 예상

- 교통사고 1일 사회적 비용 손실 : 350,000,000 원

- 비용절감액 = 350,000,000 원 + α= 500,000,000원(예상치)

※ 산출기준 : 교통사고 사회적 비용손실 + 항공기안전사고 예상 기준

○ 항공기 과주사고로 인한 고가의 항공기 손상비용 절약 및 항공기 과주사고 시 여객의 인명보호 및 안전성확보

○ 선진 공항운영 기술의 축적, 전문 인력 양성을 통하여 공항분야 기술진보를 위한 도약의 기회 제공 및 해외시장 진출 기회 확대

○ 축적된 기술과 노하우를 사업화하여 국내외 응용시장(하드웨어, 소프트웨어)에 보급함으로써 새로운 부가가치 및 고용창출 효과 발생

○ 현재 미국 ESCO 社의 기술력 독점에 의한 시공비 및 지속적인 유지관리비 과다 소요(2,055,370원/㎡) 되고, 국내 기술개발 시 미국 ESCO 社의 13% 수준에서 시공이 가능할 것으로 판단되어 해외수출 효과가 기대됨

○ 국내에 1개 공항만 시공하더라고 경제적으로 국내 기술개발투자비(20억)이상의 효과 발생

○ 정부의 에너지 절감방안 시책에 부합하는 절전시스템과 유지보수 관리 비용을 최대로 절감하는 시스템 개발로 직접비 및 소모자제 원가를 절감하는 기반 마련

○ 김포공항 외 국내공항, 국외의 브루나이 공항 등 아세안 국가와 KOICA 기금 활용 가능

○ 한국공항공사 및 대기업(SI)과의 공동 수출 추진 가능

3. 정책적 효과분석

○ 공항시설 안전 통합관리 기반기술 개발로 인하여 기존의 단위 시스템별 관리 프로세스 통합에 따른 안전한 공항 이미지 부각 예상

○ 세월호 침몰과 같은 전국가적인 대형 사고로 인하여 제기되고 있는 국가재난망 구축에서 정책적 기준이 정립되지 않아 지연되고 있는데 반하여 공항 내 시설안전 통합관리의 정책 마련의 계기가 됨

○ 공항은 도로, 철도, 항만 등의 교통 인프라 시설과 유사한 환경적 특성을 가짐으로 확대 및 범용적 활용 방안을 정책화할 경우 도입이 용이할 것으로 판단됨

5장. 연구성과의 활용 및 추가연구 필요성

1절. 연구성과 향후 활용방안

1. 국내외 활용방안

○ 본 연구를 통하여 국토해양 R&D의 「항공선진화사업」 일환으로 개발한 「공항시설 안전통합 관리시스템(AFSIS-6)」의 김포공항 테스트베드를 지속적으로 활용할 수 있도록 지원 가능

○ 본 사업을 완료함과 동시에 전용단말기의 개발이 완료되며, 전용단말기를 다음과 같은 용도로 활용할 예정임

- 김포공항 외 국내 15개 공항 및 국외의 공항에 적용

  • 김포공항 테스트베드를 통한 운용성 확인 후 타 공항으로 확장성 검토

  • 타 공항 적용 시 통신 방식, 소프트웨어 등의 최적화 작업이 필요하며 위성사진 등의 지도 데이터는 관련 기관 등의 협조 또는 구매를 통해 획득 예정

  • 국내 타 공항에 적용할 경우 국토지리정보원을 통해 레스터, 벡터의 지도 데이터를 구매 또는 해당 기관으로부터 공급받아 적용

항목	내용	비고
기관	국토교통부 국토지리정보원	• 국토측량, 항공사진(영상) 촬영, 국가기본도, 수치지형도 제작 (항공사진은 1966년도부터 5~10년, 2010년부터는 2년 주기로 전국 촬영) • 측량 및 지형공간정보 기술 연구 • 국토지리 정보 제공
공간정보 서비스	국토공간영상정보 서비스	• 영상정보 검색 • 온라인 자료 신청 및 실시간 다운로드(항공사진, 정사영상)
지도구매	오프라인, 온라인 판매	• 판매점 방문 및 전화주문(오프라인) • 수치지도(온라인) : 공간정보사이트, 국가공간정보유통센터 (결제 후 개인 PC에 다운로드) • 구지형도 : 구지형도공급센터(출력 또는 택배발송 신청)

항목		내용	비고
지도 데이터 (온라인 제공)	레스터	항공사진(영상)	• 영상 데이터 - 전자적 이미지(NIX 포맷)으로 발급
	벡터	수치지형도	• DXF 파일 제공
		수치지형도 Ver2.0	• NGI 파일 제공
		연속수치지형도	• SHP, GDB 파일 제공 • 원하는 레이어를 도엽(도곽) 제한 없이 자유롭게 선택하여 주문 가능
		토지특성도	• SHP 파일 제공
		토지이용현황도	
		연안해역기본도	
회원	기관 회원	지도 데이터 무상공급	• 위성영상을 비롯한 모든 자료 신청 가능 • 기관회원으로 등록 후 지도 무상공급 신청가능 (정부기관, 지방자치단체, 공공기관, 학교/연구소 등)
	일반 회원	지도 데이터 유상공급	• 1매당 전자수입인지 : 20,000원

[표 178] 국토지리정보원을 통한 지도 데이터 구매 관련 정보

- 대규모 기간산업시설의 이동체 안전관리 분야에 활용

  • 항만/조선소/제철소 등 면적 및 규모가 크고 작업차량의 종류와 수량이 많아서 관리에 어려움이 발생하는 기간산업시설에 전용단말기를 적용하여 작업 차량의 위치와 작업 중인 차량에 대한 업무파악이 가능

  • GIS 정보를 접목시켜 군경 보안시설 내 작업차량의 비 권한지역 침범 관리 가능

  • 작업차량 모니터링을 통하여 주요시설 침범 차량에 대한 최적경로를 제공 및 신속한 사고 처리 가능

- 대규모 기간산업시설의 도로 파손 및 이물질 관리 분야에 활용

  • 보안지역 내 주요시설의 손상이 발생한 경우 전용장비를 이용하여 실시간으로 상황정보 획득 및 관리 가능

  • 운용센터에서 주요시설에 발생한 손상에 대한 이력관리 가능

- 상시관리가 필요한 도로의 노면 상태 및 노면위의 이물질에 대한 실시간 정보 획득 및 관리 가능

- 위락시설 내 놀이기구 등 주요시설의 안전관리 현황 파악이 가능하며, 문제 발생 시 영상정보를 통하여 유지보수를 위한 자세한 정보의 수집에 적용가능

- 산업단지 내의 위험물 취급 차량 등 특수 차량관리 분야에 활용

- 오염 물질 및 유독물의 취급이 많은 산업단지 내의 위험물질 취급차량의 이동경로 및 위치 파악 가능

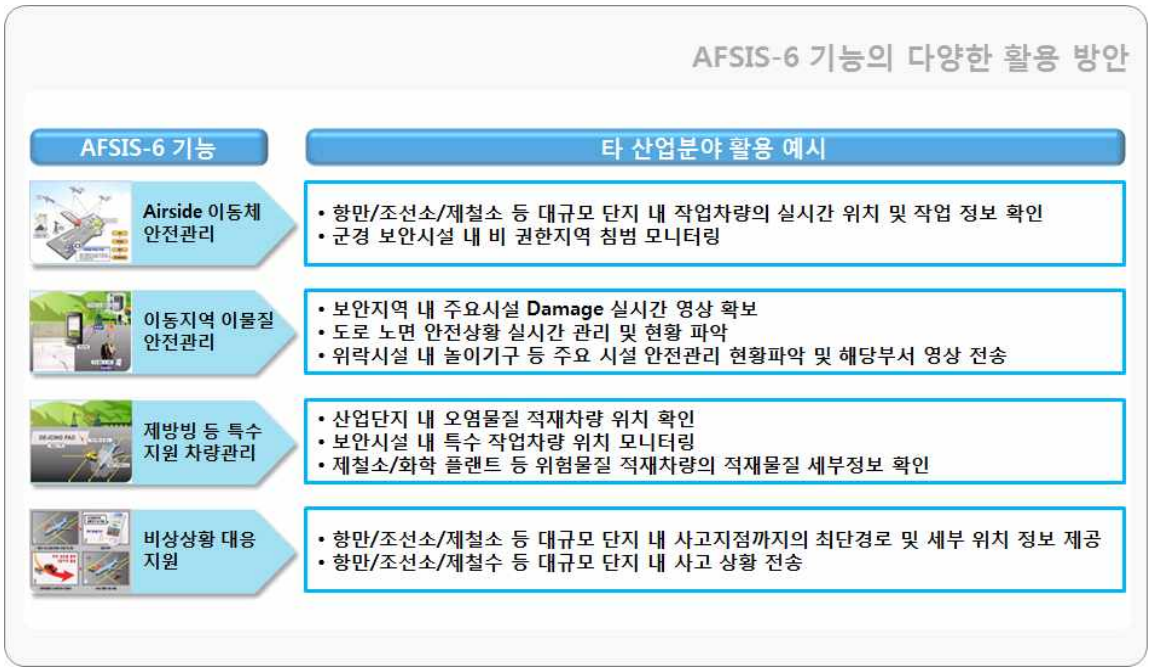
- 위험물질 취급차량의 불법 행위 및 사고 발생 시 수집된 위치를 기반으로 빠른 초동조치가 가능하여 피해 최소화

- 위험물질 취급차량의 이동경로를 관리하여 허가된 이동경로 이탈 시 경고메시지를 통한 즉시 확인 및 조치 가능

- 대규모 기간산업시설의 사고 발생 시 비상상황 대응 지원 분야에 활용

- 대규모 기간산업시설 내 사고 발생 시 사고 발생지점까지 최적경로 및 세부 위치 정보를 제공하여 사고 처리를 위한 신속한 출동이 가능

- 실시간으로 사고 처리 상황파악 및 공유 가능



[그림 179] AFSIS-6 기능의 다양한 활용 방안

○ 본 연구 성과물을 활용 가능한 분야는 아래와 같음

활용가능분야	내용
SMS	Safety Management System 관련 시설안전 데이터 통계자료 제공
A-SMGCS 기술개발	항공등화시스템과 연계하여 A-SMGCS 기술 개발의 기초기술로 활용 가능
주기장 관리	위치정보 기반 정밀 항공기 주기장 관리
소음관리	공항 내 소음지역에 대한 위치정보 제공
공사 인·허가	공항 내 각종 공사 시행 전 시설안전여부 판단
조류퇴치	공항에 출현하는 조류에 대한 정확한 위치정보 관리 및 통계 생성
항공기 정밀유도	개별 점·소등 방식의 항공등화를 이용하여 항공기 정밀유도
실세계 좌표기반 signage	각종 안내용 signage에 실세계 좌표 속성을 반영하여 공항 운영에 활용
토지 등 자산관리	정밀 위치정보 기반 자산관리
임대 관리	정밀 위치정보 기반 공항시설 임대 관리
공항 외곽 경비보안	과학경비 시설과 결합한 위치정보 기반 보안 모니터링
비행시뮬레이션/ 경로계획	비행 모니터링/ 궤도 분석
비상계획/ 사고 보안관리	사고관리, 보안관리
공항시설관리	활주로 관리, 배수로관리
공항주변 시설물관리	비행장 모니터링
공항건설 계획/기타	건설계획
환경	조류출몰 모니터링
위치추적	차량, 화물 등에 대한 위치 모니터링
개요도 제작	복잡한 선로를 개요도를 제작하여 관리
편의시설 관리	화장실 등 각종 편의시설 관리

[표 179] 본 연구 성과물의 활용가능 분야

2. 추가연구 필요성

- 공항시설 관리를 위한 시스템의 개발은 활발하게 진행되고 있지만, 공항시설 안전통합관리시스템과 같이 공항시설관리 업무를 총괄적으로 서비스하는 제품의 개발은 이루어지지 않고 있음
- 공항에 구축되어 있는 시스템들의 인터페이스를 통합하여 체계적인 안전관리 필요
- 주차관리, 통행료 부가, GPS를 통한 차량 주행거리 관리 등 안전관련 분야에도 확장이 가능하도록 방안마련 필요

2절. 사업화 추진전략

1. 사업화 추진계획

○ 본 사업을 통하여 연구개발을 완료한 전용단말기를 AFSIS-6에 적용하여 다음의 단계와 같이 사업화 추진 예정

단계	항목	세부 설명
안정화	안정화 작업 및 필드테스트 지속화	• 본 시스템은 일부는 실용화를 위한 작업이 진행되었으나 일부는 아직 시제품수준에 머물러 있어 상용화를 위한 QoS가 보장되는 확장성, 편리성, 기술성을 고려한 개선작업 및 안정화의 지속적인 추진이 필수적임 • 추가로 발생하는 비용의 절감을 위해 'R&D정책인프라사업 기술사업화'와 같은 국책과제 등을 통한 상용화 추진
사업화	홍보	• 언론기사홍보 및 영상홍보자료 제작 • KOTRA 및 중기청, 바이어, 중개상 등의 홈페이지활용 • 각종 유명 신문 및 잡지 광고 활용 • 각종 항공관련 세미나 및 전시회 참석
	국외 공항 수출	• 국토교통부, 국토교통과학기술진흥원, 한국교통연구원, 한국공항공사 공조 협조 • 대기업 SI업체의 해외 망 활용 • KOTRA를 활용한 해외 공항 AGENCY 조사 • 미국 AGENCY를 통한 국방부 공항 및 민간 공항 조달 추진 • 개발도상국을 대상으로 한 KOICA 자금을 활용한 수출추진 • 북미, 유럽 등 선진국 수출
	국내 공항에 구축	• 인천공항 등 국내공항 구축 추진 • 국토부 국책사업화 추진
활용화	통합관제시스템	• 기 구축된 CCTV망, 레이더망과 안전통합관제시스템 통합망 사업 제안 추진 • 향후 미래 관제시스템으로 확장이 가능한 로드맵작성에 따라 사업 추진
	타 시스템 적용	• ETRI와 아시아나 IDT 물류시스템과 융합 • 방범 방재 시스템 • 항만, 조선소, 제철소, 군경보안시설, 산업단지

[표 180] 사업화 추진 계획

2. 사업화 추진전략



[그림 180] AFSIS-6의 사업화 추진전략

가. 공항 운영기관과 협력관계 확보

○ 한국공항공사와 MOU 체결 및 김포공항 테스트베드를 통한 시스템 검증 진행

○ 국토교통부, 한국공항공사와의 협력관계를 통하여 국내 13개 공항에 시스템 구축 추진

○ 실 사용자의 개선의견 반영 및 제품화 추진

○ 한국공항공사의 마케팅 역량을 활용, 전 세계 공항을 대상으로 시스템 수출 사업화 추진(필리핀, 인도네시아, 모나코, 선진국 등)

나. 컨소시엄 구성

○ 대기업과 컨소시엄을 구성하여 국외 공항에 시스템 구축 추진

○ 기존 구축 시스템 개발업체와의 컨소시엄을 통한 수익 창출 공유

다. 전시회 및 워크샵 참여

○ 사업적인 파트너쉽 구축 등 국내·외 항공 기술 전시회 참여를 통한 마케팅 추진

○ 지속적인 홍보물 제작 및 원가 절감을 통한 시범운용 제도 활용

- GLOBAL BUSINESS PLAZA(KOTRA 주관)

- EUROSATORY(프랑스)

- AUSA(미국)

- DPBB(미국, 유럽)

○ 서울 에어쇼, 파리 에어쇼 등 국내외 항공기술 전시회 참여를 통한 마케팅 추진

라. ICAO의 USOAP를 통한 이미지 마케팅

○ 국제민간 항공기구(ICAO)의 회원국 190개국을 대상으로 8가지 분야를 평가하는 항공안전 종합평가(USOAP)에서 본 시스템을 활용

○ 항공안전종합평가 결과를 토대로 ICAO 회원국 대상의 이미지 마케팅 추진

마. 국가 정책지원 활용

○ 시장개척단, KOTRA, 바이어 상담회 참여 등 적극적인 제품 홍보와 마케팅 활동 추진

- K-HIT, 유럽 & 미국 유통망 활용

- 중소기업 미국시장 진출 전략 세미나 참가를 통하여 미국 현지 바이어에 시스템 홍보 및 수출 추진

- 미국 GSA(조달청) 유력 조달벤더 상담회를 통한 시스템 홍보 및 수출 추진

- 대형 공항 바이어 상담회를 통한 시스템 홍보 및 수출 추진

순번	실적	일정
1	글로벌 공항조달 플라자(Global Business Plaza) 바이어 상담회 참가(KOTRA 주관)	2013.04
2	해외 전시회 참가(AUSA)	2013.10
3	국내 전시회 참가(IDEX)	2013.11
4	김포공항 시스템 운용 지원(전용단말기)	2013.11
5	글로벌 공항조달 플라자(Global Business Plaza) 바이어 상담회 참가(KOTRA 주관)	2014.04
6	국토교통부 검토 회의 및 시연	2014.06
7	해외 전시회 참가 예정(AUSA)	2014.10

[표 181] 시스템 홍보 및 수출을 위한 노력(최근 실적)



[그림 181] GBP 참가(공항/철도분야)



[그림 182] 해외 전시회 참가(AUSA)

3. 사업화 추진체계

○ 국토교통부, 국토교통과학기술진흥원은 시스템의 사업화를 총괄하는 기관으로 개발기간 동안 요구사항 및 평가 등 사업관리 측면의 업무를 진행함

○ 주관연구기관인 인소팩(주)은 본 연구개발 기술의 사업화를 위한 연구개발을 담당하며, 가격 경쟁력과 높은 기술 수준의 제품 개발을 위하여 주력하고 실제 수익 창출이 진행되도록 영업 및 홍보 활동을 수행함

○ 한국공항공사는 김포공항의 테스트베드 제공과 사용자 측면에서의 요구사항 도출 및 의견 제시를 통한 제품화에 노력함

○ AFSIS-6 개발기간 동안 사업에 참여한 업체들의 노하우를 활용 및 필요에 따라 다양한 분야의 자문 및 평가위원을 포함한 위원회를 구성함



[그림 183] 사업화 추진체계

[참고문헌]

[1] 박용화,“인천국제공항 경쟁력 강화 방안”, 2001.09

[2] 홍석진,“국가차원의 항공안전 위기관리체계 구축 방안 연구(1단계)”, 2003.04

[3] 김제철 외 2인,“공항시설 사업의 타당성평가 방법론 개선 연구”, 2010.03

[4] 한재현 외 3인,“차세대 항공교통관리시스템(NAGAS) 도입을 위한 기초연구”, 2010

[5] 송기한, 김연명 외 2인,“공항의 경제적 파급효과 분석방법 개발 연구”, 2011

[6] 한재현, 김장환, 안미진,“항공안전기술 발전을 위한 기초연구”, 2011

[7] 한양사이버대학교,“차세대 지능형 공항 시스템 개발 기획연구보고서”, 2007.10.8.

[8] 건국대학교,“항공관제시스템 개발 기획 연구 보고서”, 2007.10.29.

[9] 서울대학교,“항공안전기술개발사업단_총괄보고서”, 2007.10.31.

[10] 박용화,“우리나라 공항운영의 구조개혁방안”, 1999.02

[11] 김제철 외 1인,“인천국제공항의 허브화 전망과 대책“, 1997.05

[12] 김제철 외 3인,“항공부문 중장기 발전 전략 수립 연구”, 2006.10

[13] 김연명, 박진서,“지방공항 운영체계 개선방안 연구”, 2005.06

[14] 국토교통부 항공안전본부,“항공안전관리매뉴얼(SMS)", 2007

[15] 한국건설교통기술평가원,“차세대 지능형 공항 시스템 개발 기획연구보고서”, 2007.10

[16] 한국교통연구원, 『공항시설 안전관리 기술개발 - 공항 시설 안전통합관리시스템 개발』, 2008

[17] www.aaae.org (American Association of Airport Executives)

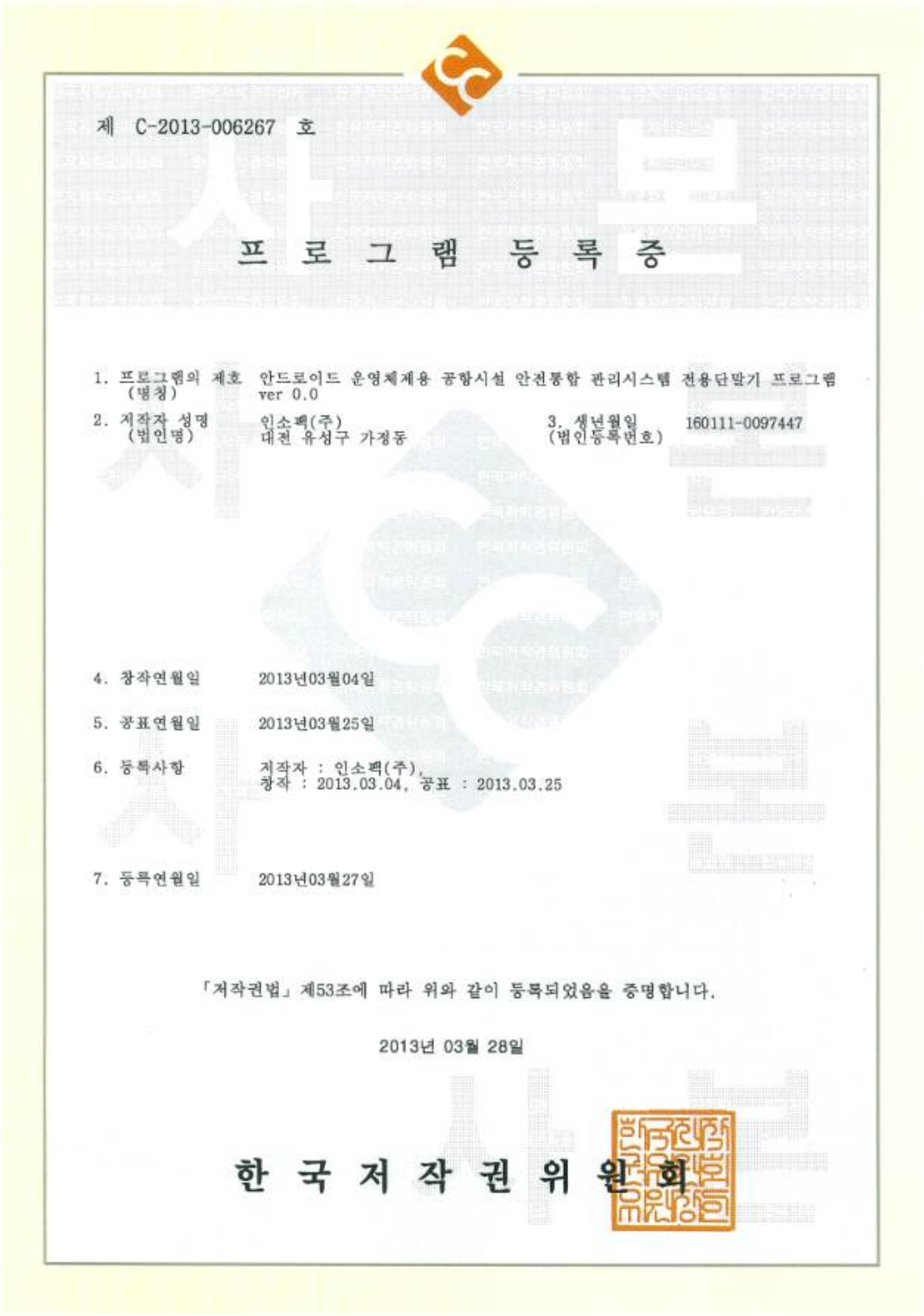
[18] www.eurocontrol.int (European Organization for the Safety of Air Navigation)

[19] www.faa.gov (Federal Aviation Administration)

[20] www.icao.int (International Civil Aviation Organization)

[21] http://digitais.ist.utl.pt/msn/PROJECTOS/AIRNET.htm

[부 록 - 프로그램 등록증 3부]



제 C-2013-006267 호

프 로 그 램 등 록 증

1. 프로그램의 제호 (명칭)	안드로이드 운영체제용 공항시설 안전통합 관리시스템 전용단말기 프로그램 ver 0.0		
2. 저작자 성명 (법인명)	인소㈜ 대전 유성구 가정동	3. 생년월일 (법인등록번호)	160111-0097447
4. 창작연월일	2013년03월04일		
5. 공표연월일	2013년03월25일		
6. 등록사항	저작자 : 인소㈜, 창작 : 2013.03.04, 공표 : 2013.03.25		
7. 등록연월일	2013년03월27일		

「저작권법」 제53조에 따라 위와 같이 등록되었음을 증명합니다.

2013년 03월 28일

한 국 저 작 권 위 원 회

제 C-2013-028525 호

프 로 그 램 등 록 증

1. 프로그램의 제호 (명칭)	안드로이드 운영체제용 공항시설 안전통합 관리시스템 전용단말기 프로그램 ver 1.0		
2. 저작자 성명 (법인명)	인소팩 주식회사 대전광역시 유성구 가정로	3. 생년월일 (법인등록번호)	160111-0097447
4. 창작연월일	2013년12월09일		
5. 공표연월일	2013년12월19일		
6. 등록사항	저작자 : 인소팩 주식회사, 창작 : 2013.12.09, 공표 : 2013.12.19		
7. 등록연월일	2013년12월24일		

「저작권법」 제53조에 따라 위와 같이 등록되었음을 증명합니다.

2013년 12월 26일

한 국 저 작 권 위 원 회

제 C-2014-014372 호

프 로 그 램 등 록 증

1. 프로그램의 제호 (명칭)	안드로이드 운영체제용 공항시설 안전통합 관리시스템 전용단말기 프로그램 ver 2.0		
2. 저작자 성명 (법인명)	인소텍 주식회사 대전광역시 유성구 가정로	3. 생년월일 (법인등록번호)	160111-0097447

4. 창작연월일 2014년06월20일

5. 공표연월일 2014년06월20일

6. 등록사항 저작자 : 인소텍 주식회사,
창작 : 2014.06.20, 공표 : 2014.06.20

7. 등록연월일 2014년06월23일

「저작권법」 제53조에 따라 위와 같이 등록되었음을 증명합니다.

2014년 06월 24일

한 국 저 작 권 위 원 회

주 의

1. 이 보고서는 국토교통부에서 시행한 국토교통기술사업화지원사업의 연구보고서입니다.
2. 이 보고서 내용을 발표하는 때에는 반드시 국토교통부에서 시행한 사업의 연구결과임을 밝혀야 합니다.
3. 국가과학기술 기밀유지에 필요한 내용은 대외적으로 발표 또는 공개하여서는 아니됩니다.