

공 학 석 사 학 위 논 문

냉동컨테이너 모니터링 시스템을 위한 모바일 서비스 구현



2012년 2월

부 경 대 학 교 대 학 원

전 자 공 학 과

이 성 준

공 학 석 사 학 위 논 문

냉동컨테이너 모니터링 시스템을 위한 모바일 서비스 구현

지도교수 이 원 창

이 논문을 공학석사 학위논문으로 제출함.

2012년 2월

부 경 대 학 교 대 학 원

전 자 공 학 과

이 성 준

이성준의 공학석사 학위논문을 인준함.

2012년 2월 24일



주 심 공학박사 강 근 택 (인)

위 원 공학박사 주 문 갑 (인)

위 원 공학박사 이 원 창 (인)

목 차

I. 서론.....	1
II. 냉동 컨테이너 모니터링 시스템.....	5
1. 시스템 구성.....	5
2. 통신 모듈.....	6
3. 서버.....	11
4. Simulator.....	14
III. 모바일 서비스.....	18
1. 시스템 개요.....	18
2. Socket_Client 클래스.....	19
3. 서비스와 브로드캐스트 리시버.....	19
4. CELL.....	21
5. VENDOR.....	22
6. Container History Activity.....	24
7. ALARM.....	25
8. SETTING.....	26
IV. 실험.....	27
V. 결론.....	31
VI. 참고문헌.....	32
감사의 글.....	

그림 목차

그림 1. PCT 모델과 Interrogator Port의 장착을.....	3
그림 2. 시스템 구성.....	5
그림 3. 통신모듈의 시스템 개요.....	6
그림 4. 통신모듈 동작 순서.....	9
그림 5. 통신모듈 IP 설정용 프로그램.....	10
그림 6. 서버.....	11
그림 7. 모니터링용 서버 동작 흐름.....	12
그림 8. 냉동 컨테이너 정보 클래스의 데이터 구조	13
그림 9. 서버의 데이터베이스 구조.....	13
그림 10. Simulator 모듈	15
그림 11. 각 제조사 모드의 Flow Chart.....	16
그림 12. Simulator Setting 프로그램.....	16
그림 13. Simulator 확인	17
그림 14. 모바일 어플리케이션 메인.....	18
그림 15. Notification	20
그림 16. 구역 목록 (안드로이드).....	21
그림 17. Container Item.....	21
그림 18. 구역 목록 (IOS).....	22
그림 19. AllConInfo 데이터 구조.....	22
그림 20. VENDOR 화면 (안드로이드).....	23
그림 21. VENDOR 화면 (IOS).....	24
그림 22. 냉동 컨테이너 과거 데이터 표시.....	25
그림 23. 알람 리스트.....	26
그림 24. SETTING Activity.....	26
그림 25. 필드 테스트.....	27
그림 26. Lab 테스트.....	28
그림 27. 냉동 컨테이너 온도 변화.....	28
그림 28. 데이터베이스와 어플리케이션의 온도 데이터 비교.....	29
그림 29. 데이터베이스와 어플리케이션의 온도 그래프 비교.....	30

표 목차

표 1. 부산항 컨테이너 물동량.....	1
표 2. Atxmega128A1 스펙.....	7
표 3. W5300 스펙.....	8
표 4. 모바일 단말기용 서버 명령 목록.....	14
표 5. Socket_Client 함수 목록	19

Mobile Service for Reefer Container Monitoring System

Seong Jun Lee

Department of Electronic Engineering, The Graduate School,
Pukyong National University

Abstract

Recently there have been many researches for developing reefer container monitoring systems to improve the efficiency of maritime transport and to reduce the risk of personal injury. IMO(International Maritime Organization) recommended to use the reefer container monitoring system with PCT(Power Cable Transmission) modem complying with ISO10368 standard, but it is not being widely used due to its several weaknesses such as data loss of PCT modem, no installation of PCT modem in more than 45% of reefer containers being used currently and so on. Therefore monitoring reefer containers still rely on human inspection. In this thesis we present a new reefer container monitoring system using interrogator port fitted to every reefer container and its mobile service. In this system a smart communication module is used to gather information including temperature and alarm data from reefer container and to transmit it to server via TCP/IP. With mobile service transport companies and container terminal management companies can get very useful information for their reefer containers easily and everywhere in real time.

I. 서론

냉동컨테이너는 특성상 온도에 민감한 화물들을 적재하기 때문에 온도 특성을 주기적으로 확인해야 한다. 2000년대 초반에는 단순 운송비 절감만을 고려한 생산성 향상을 위한 기술에만 집중해서 사실상 관리가 잘 이루어지지 않았다. 하지만 최근에는 물류 환경의 변화와 IT산업의 발전으로 인해 서비스 극대화를 통한 경쟁력 향상이 요구되고 있다. 전 세계적으로 총 컨테이너 물동량이 증가해감에 따라 냉동 컨테이너 또한 물동량이 꾸준히 증가하고 있다. 표1은 컨테이너 물동량이 세계 5위에 속하는 부산항의 컨테이너 물동량이다[1].

표 1. 부산항 컨테이너 물동량

(단위 : 톤)

연도	총계	조제식품,음료,주료	비율(%)
1998	96,432,544	1,690,817	2%
1999	107,757,128	1,874,697	2%
2000	117,228,960	3,196,948	3%
2001	149,661,433	3,639,610	2%
2002	165,676,590	3,649,545	2%
2003	187,212,125	5,204,346	3%
2004	219,760,112	6,419,590	3%
2005	223,454,110	6,584,405	3%
2006	229,938,598	9,225,045	4%
2007	243,564,954	9,395,686	4%
2008	241,683,032	9,478,390	4%
2009	226,181,574	8,196,546	4%
2010	262,962,142	7,012,185	3%

항목별 물동량 중 냉동 컨테이너에 관련된 화물인 조제식품, 음료, 주료 품목의 물동량을 보면 계속 증가해가는 추세이며, 앞으로도 계속 증

가해 나갈 추세이다. 계속되는 컨테이너 선박의 대형화 추세로 컨테이너 터미널의 냉동컨테이너 장치 수요가 증가함에 따라 효율적인 모니터링 방법의 필요성이 커지고 있다[2]. 이러한 냉동 컨테이너는 고장 등의 이유로 온도유지를 하지 못하였을 경우 심각한 화물의 손상을 가져오고, 막대한 재산 피해가 발생한다[3]. 이에 따른 책임 소재에 따른 분쟁이 발생하게 되므로 냉동컨테이너의 고장유무를 빠르게 판단하여 보수작업을 하는 것이 필요하며, 이를 위한 효율적인 모니터링 방법에 대한 많은 연구가 활발히 진행되고 있다. 현재의 모니터링 시스템의 경우 국제해사기구(IMO : International Maritime Organization)에서 1992년에 제정된 ISO 10368 Freight thermal containers - Remote condition monitoring 표준에 따른 전력선 통신(PCT : Power Cable Transmission)을 사용하는 모니터링 시스템을 권장하고 있다[4]. 냉동 컨테이너 모니터링 시스템은 이러한 냉동 화물의 파손의 위험을 줄이고, 냉동 컨테이너의 관리를 용이하게 하기 위하여 꼭 필요한 시스템이다. 기존의 전력선 모뎀을 이용한 방식은 크게 Conventional(4-Pole)방식과 PCT를 이용한 방식으로 나누어진다[5]. 먼저 4-Pole 방식을 살펴보면, 컨테이너에 부착되어 있는 4-Pole Signal Plug에서 출력되는 각 신호들의 상관관계를 연산하여 아래의 4가지 상태를 관리할 수 있다[6].

- Temperature In Range : ON/OFF (AC 24V)
- Compressor Running : ON/OFF (AC 24V)
- Defrost Running : ON/OFF (AC 24V)
- Power Connection : ON/OFF (AC 24V)

PCT를 이용한 방식은 냉동 컨테이너에 접속된 전력선을 직접 이용하

여 외부로 신호를 전송하는 방식으로 모뎀의 전송 속도에 따라 HRCD(High data rate Remote Communications Device)와 LRCD(Low data rate Remote Communications Device)로 구분하는데, 전력선을 통과하는 60Hz의 주파수를 가지고 고주파 신호를 변조, 복조하여 데이터를 상호 교환하는 방식이다[7]. PCT를 이용한 냉동 컨테이너 모니터링 시스템이 권장하고 있기는 하지만 PCT 모뎀은 장착율에서 Interrogator Port에 비하여 많이 떨어지는 상황이다. 그림 1은 냉동 컨테이너의 PCT 모뎀과 Interrogator Port의 장착율을 나타낸 것이다. 2008년 기준으로 보면 Interrogator Port는 100%에 가까운 장착율을 보이는 반면 PCT 모뎀은 약 55%정도의 장착율을 보인다.

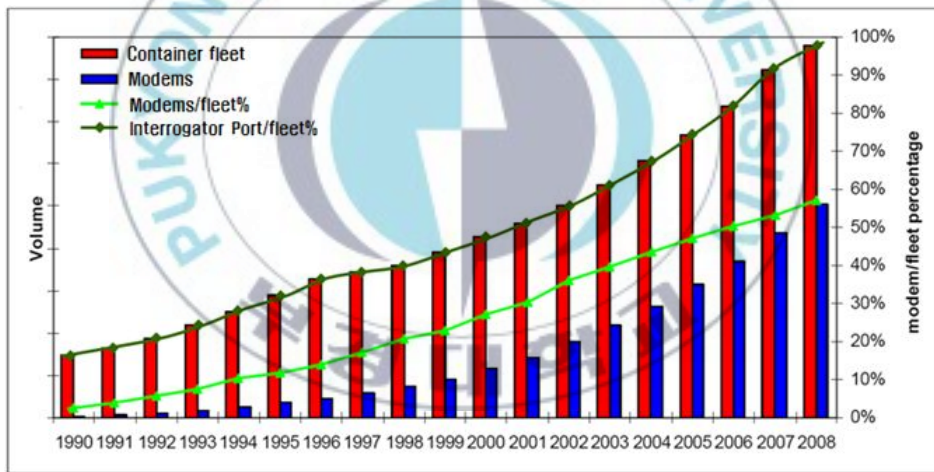


그림 1. PCT 모뎀과 Interrogator Port의 장착율

그리고 전력선 모뎀의 특성상 전력부하의 변동에 따라 데이터 손실 등 치명적인 단점으로 인하여 현재 인력에 의존하고 있다[8]. 때문에 현재 많은 냉동 컨테이너의 상태를 감시하기 위하여 많은 인력과 시간이 필요한 상황이다. 매 2시간 간격으로 순찰/기록하면서 3교대 방식으로 운영되는 재래방식은 냉동 컨테이너의 장애 발생 시 즉각 대처가 어려운 점

이 있다[9]. 그리고 냉동컨테이너 제조사별로 데이터패킷의 구성 또한 다르기 때문에 각 제조사별로 단말기를 휴대해야 한다는 불편함이 존재한다.

본 논문에서는 이러한 문제를 해결하기 위하여 항만 물류 시스템에서 활용 가능한 새로운 냉동컨테이너 모니터링 시스템과 모바일 서비스를 통한 모니터 기술을 제안한다. PCT 모뎀의 단점을 극복한 새로운 냉동컨테이너의 실시간 모니터링 기술과 모바일 시스템 환경을 접목하여 냉동컨테이너 상태를 실시간으로 모니터링 하여 현재 온도, 알람데이터, 온도 로깅 데이터 등을 쉽게 통합관리하고, 모바일 단말기에서 해당 데이터들을 열람할 수 있도록 하여 냉동컨테이너 모니터링 시스템의 효율성을 높였다. 냉동컨테이너 모니터링 기술로는 현재 대부분의 냉동컨테이너에 내장되어있는 Interrogator Port로부터 데이터를 수집하여 각 제조사별 데이터 처리 알고리즘에 따라 분석, 변환한 데이터를 TCP/IP통신을 통하여 서버로 전송한다. Interrogator Port는 전력선모뎀과 달리 전력 부하에 따른 오류가 없어서 데이터에 대한 신뢰도가 높고, 대부분의 냉동컨테이너에 장착되어있기 때문에 효율적인 모니터링이 가능하다. 그리고 TCP/IP통신을 사용함으로써 하나의 서버로 모든 냉동컨테이너의 모니터링이 가능하기 때문에 관리자는 쉽게 전체 냉동컨테이너의 상태를 파악할 수 있다. 냉동컨테이너로부터 수집된 온도, 알람 데이터 등을 서버의 DB에 저장하고, 이를 모바일 단말기와 Wi-Fi를 통하여 공유하여 서버와 모바일 단말기의 동일한 모니터링 환경을 제공한다. 모바일 단말기를 사용한 냉동컨테이너 모니터링 시스템은 시간, 장소에 구애받지 않는 실시간 모니터링을 가능하게 하며, 항만회사나 냉동컨테이너 관리회사에게 더 나은 모니터링 환경을 제공할 수 있다.

II. 냉동 컨테이너 모니터링 시스템

1. 시스템 구성

냉동 컨테이너의 모니터링과 모바일 서비스를 구현하기 위한 시스템의 구성은 그림 2와 같다.

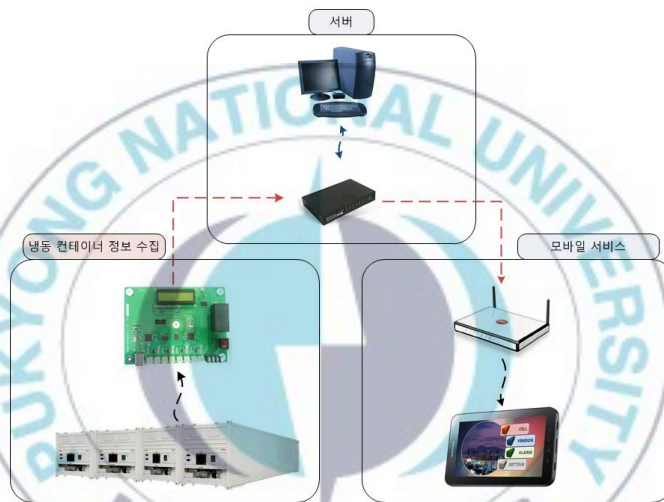


그림 2. 시스템 구성

냉동 컨테이너로부터 데이터를 수집하는 통신모듈과 여러 통신모듈을 통합하여 서버에 연결해주는 허브, 데이터를 관리하고 모바일 단말기에서 접속이 가능한 서버가 필요하다. 냉동 컨테이너의 Interrogator Port와 통신모듈은 RS-232통신을 사용하여 통신하며, 통신모듈과 서버는 TCP/IP방식의 유선 이더넷 통신을 사용한다. 서버와 모바일 단말기는 WiFi를 통하여 통신한다.

2. 통신 모듈

2.1 시스템 개요

통신모듈은 냉동컨테이너의 데이터를 수집하여 해당 제조사별 알고리즘에 맞게 변환하여 사용자들이 알아볼 수 있는 데이터로 변환하여 TCP/IP통신을 통하여 서버로 전송한다. 통신모듈 하나에서 8개의 냉동컨테이너의 정보를 처리하기 위하여 8개의 UART통신이 가능한 메인 프로세서로 Atmel社의 Atxmega128A1을 사용하였다. 이 프로세서로 냉동컨테이너로부터 수신 받은 데이터를 각 제조사의 알고리즘에 따라 분석, 변환한 뒤 Wiznet社의 W5300을 통하여 TCP/IP통신 방식으로 서버에 전송한다.

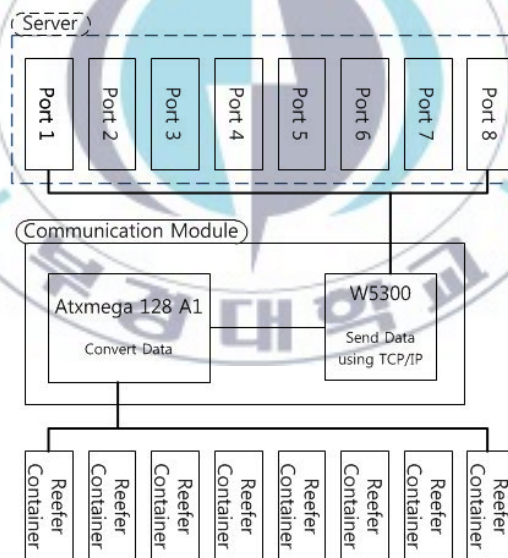


그림 3. 통신모듈의 시스템 개요

2.2. 통신모듈 구성

2.2.1. 프로세서

통신모듈의 메인 프로세서로 Atmel社의 Atxmega128A1을 사용하였으며, 이 프로세서는 컨테이너로 명령어를 보내고, 그에 해당하는 정보를 수신 받으며, 각 제조사의 알고리즘에 따라 데이터를 변환한 뒤 W5300으로 보내는 등의 핵심적인 알고리즘을 내장하는 역할을 한다. 표 2는 Atxmega128A1의 간단한 스펙이다[10].

표 2. Atxmega128A1 스펙

Manufacturer	Atmel
Product Category	Microcontrollers (MCU)
Data Bus Width	8 bit
Family / Core	ATxmega
Instruction Set Architecture	RISC
Package / Case	TQFP-100
Program Memory Type	Flash
Program Memory Size	128 KB
Data Ram Size	8 KB
Maximum Clock Frequency	32 MHz
On-Chip ADC	12 bit
On-Chip DAC	12 bit
Interface Type	PWM, USART, TWI
Operating Supply Voltage	1.6 V to 3.6V
Product	MCU
Packaging	Tray
Operating Temperature	-40 °C to 85 °C

2.2.2. 이더넷 컨트롤러

TCP/IP 통신을 위하여 Wiznet社의 W5300을 사용하였으며, 이 컨트롤러는 Atxmega128A1으로부터 변환된 데이터들을 서버로 전송 가능하게 한다. 여러 프로토콜 중 신뢰성과 안정성을 위하여 TCP/IP방식을 사용하였다. 표 3은 W5300의 스펙에 대해 나타내었다[11].

표 3. W5300 스펙

Supports hardwired TCP/IP protocols TCP, UDP, ICMP, IPv4, ARP, IGMPv2, PPPoE, Ethernet
Supports 8 independent SOCKETs simultaneously
High network performance Up to 80 Mbps (DMA)
Supports hybrid TCP/IP stack(software and hardware TCP/IP stack)
IP Fragmentation is not supported
Internal 128Kbytes memory for data communication
More flexible allocation internal TX/RX memory according to application throughput
Supports memory-to-memory DMA(only 16bit Data bus width & slave mode)
Embedded 10BaseT/100BaseTX Ethernet PHY
Supports auto negotiation (Full-duplex and half duplex)
Supports auto MDI/MDIX(Crossover)
Supports a external PHY instead of the internal PHY
Supports 16/8bit data bus width
External 25MHz operation frequency
3.3V operation with 5V I/O signal tolerance
Embedded power regulator for 1.8 core operation

2.2.3. 외부메모리

프로세서로 사용된 Atxmega128A1의 RAM은 8Kbyte로 추가 확장성을 고려하여 Hyundai Electronics America社의 HY62256ALT1-70을 이용하였다. 이 메모리는 32Kbyte의 SRAM으로 추가될 수 있는 데이터들을 넉넉하게 저장하기 위함으로 사용되었다.

2.2.4. 기타

보안을 위한 Altera社의 EPM3032A, 외부 디버깅을 위한 JTAG, AC 220V를 DC 5V로 변압하기 위한 AC-DC 컨버터, IP등의 정보를 확인하기 위한 LCD등이 포함되었다.

2.3. 동작 구조

처음 통신모듈이 켜지면 EEPROM을 읽고, 정해진 IP로 부팅된다. 그리고 통신모듈은 서버로 접속하는 동작을 계속 반복한다. 서버에 접속 되면 서버로부터 명령을 받아서 각 포트별로 컨테이너 상태를 수집한다. 만약 수신되는 데이터가 없을 경우 컨테이너는 없다고 판단하고 다음 포트로 역할을 넘긴다. 수신되는 데이터가 있으면 해당 제조사 알고리즘으로 변환하여 사용자가 알 수 있는 데이터로 변환하여 서버로 전송한다. 그림 4는 통신모듈의 동작 알고리즘을 간단히 나타낸 것이다.

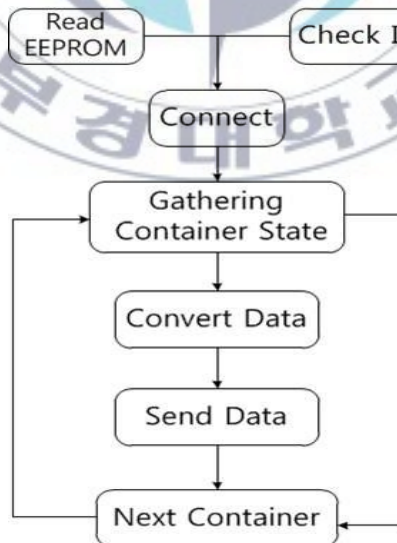


그림 4. 통신모듈 동작 순서

통신모듈은 서버로부터 시작, 정지, 리셋, 설정 등 4가지의 명령을 수신한다. 시작 명령은 냉동 컨테이너로부터 정보 수집을 시작하는 명령이고, 이 명령에는 다시 시작하기 위한 시간정보가 포함된다. 만약 정해진 시간 안에 시작 명령이 수신되지 않으면 통신 모듈을 리셋 한다. 정지 명령은 통신 모듈이 냉동 컨테이너의 정보를 수집하면서 저장한 상태정보를 초기상태로 복구하고 수집 동작을 멈춘다. 리셋 명령은 서버에서 판단하여 통신모듈에 문제가 발생하였다고 판단될 때 보내는 명령으로 명령을 받은 즉시 통신모듈을 리셋 한다. 설정명령은 TCP/IP통신을 통하여 통신모듈의 IP정보와 서버 정보를 수정하는 명령으로, 명령에 포함된 IP정보와 서버 정보는 EEPROM에 저장되어 통신모듈이 켜질 때 마다 해당 IP정보로 설정된다.

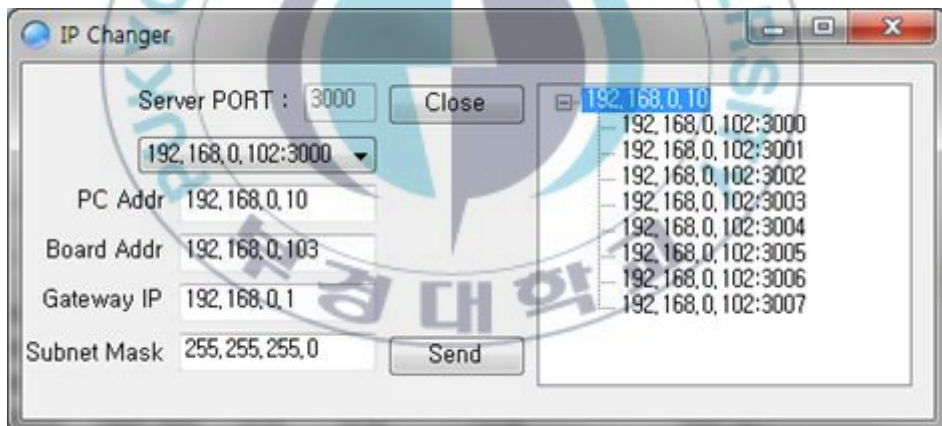


그림 5. 통신모듈 IP 설정용 프로그램

3. 서버

3.1. 시스템 개요

서버는 컨테이너 정보를 수신하는 포트와 모바일 기기와의 통신을 하는 포트가 나뉜다. 모니터링용 서버는 통신모듈로부터 수신 받은 데이터를 관리자에서 보여주고, 그 데이터를 데이터베이스에 저장한다. 그리고 모바일 기기용 서버는 모바일 기기가 요청하는 명령에 따라 구역 정보, 컨테이너 정보 등을 모바일 기기로 전송한다. 모니터링과 서로 간섭받지 않는 안정된 통신을 하기 위하여 모니터링용 서버와 모바일 기기용 서버는 한 서버 내에서 포트를 달리 두고 있다. 그림 6은 서버의 동작화면이다.

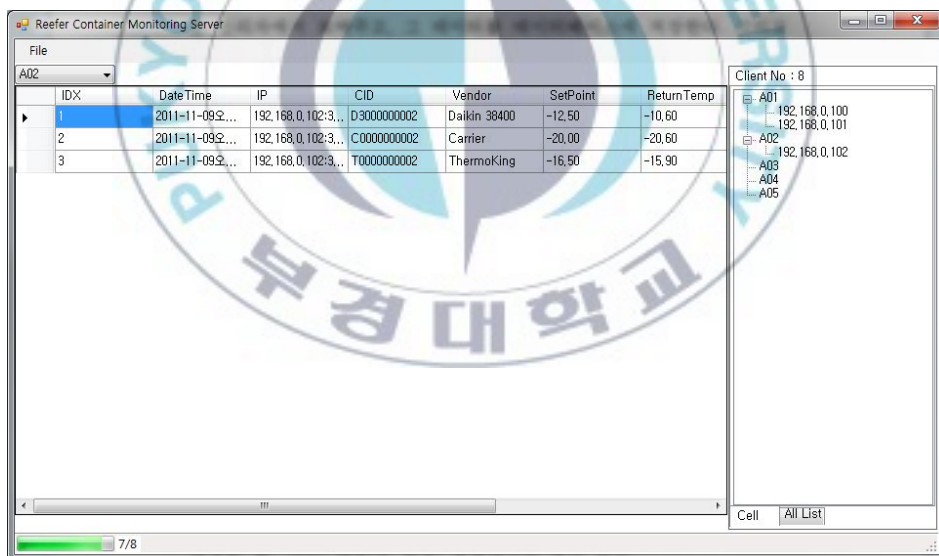


그림 6. 서버

3.2. 모니터링용 서버

먼저 모니터링용 서버는 송신부, 수신부, 데이터베이스로 나뉜다. 송신부에서는 주기적으로 통신모듈로 시작 명령을 보내서 컨테이너 정보들을 수집한다. 만약 서버에 접속되어있는 클라이언트의 개수보다 적은 데이터가 수신되면 통신모듈을 리셋 하여 다시 서버에 접속할 수 있도록 함으로써 연결의 신뢰성을 높인다. 수신부에서는 실제 컨테이너의 정보를 수신 받으며, 수신된 데이터는 구역, 컨테이너ID별로 정리하여 데이터베이스에 저장된다. 데이터베이스는 수신되는 데이터 중 컨테이너 정보를 담은 데이터만 저장되며, 저장되는 데이터는 구역정보, IP, 시간, 제조사, 온도, 알람데이터 등이 있다. 그림 7은 모니터링용 서버의 동작흐름을 나타낸 것이다.

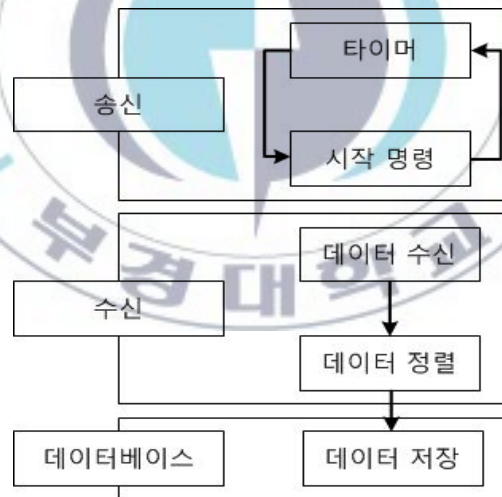


그림 7. 모니터링용 서버 동작 흐름

수신한 데이터는 Container_Info 클래스에 항목별로 나눈 후 이 클래스를 기반으로 데이터베이스의 모든 데이터를 알맞게 배분하여 저장한다. 그림 8은 Container_Info 클래스의 데이터 구조이다.

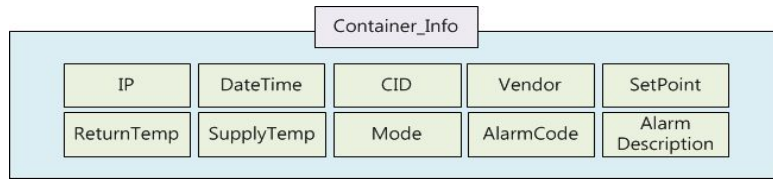


그림 8. 냉동 컨테이너 정보 클래스의 데이터 구조

데이터베이스는 구역과 IP정보들을 매칭 시키기 위한 테이블 'Cell', 각 구역의 현재 온도상태를 저장하는 각 Cell name의 테이블¹⁾, 각 컨테이너의 과거 정보를 저장하기 위한 각 컨테이너 ID의 테이블, 알람 정보를 저장하는 'Alarm_List' 테이블로 구성되었다. 그림 9는 전체 RCMS데이터베이스의 테이블과 필드를 나타낸 것이다.

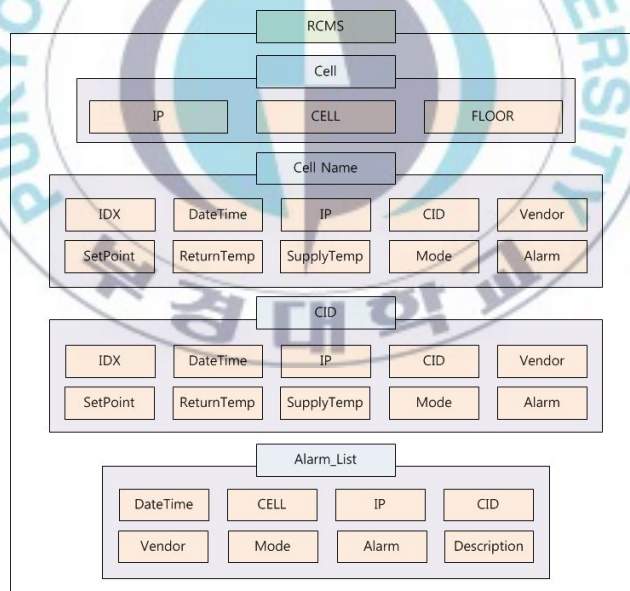


그림 9. 서버의 데이터베이스 구조

1) 본 논문에서는 A01, A02, ...의 순서로 정하였다.

3.3. 모바일 단말기용 서버

모바일 단말기용 서버는 모니터링용 서버와 함께 동작하지만 포트를 달리 두어서 서로 간섭받지 않도록 하였다. 모바일 단말기로부터 명령을 수신하여 그에 따른 데이터베이스에 저장되어 있는 정보를 모바일 단말기로 전송한다. 모바일 단말기로부터 수신하는 명령의 종류는 아래 표 2와 같다.

표 4. 모바일 단말기용 서버 명령 목록

명령	내용
CellList	냉동 컨테이너 야적장의 구역 목록
CellInfo	야적장 해당 구역의 컨테이너 현재 상태정보
ConHistory	해당 컨테이너 ID의 냉동 컨테이너 상태 기록 정보
CellAlarm	해당 냉동 컨테이너 야적장 구역의 알람 데이터 정보
AlarmHistory	전체 알람 데이터 정보

4. Simulator

4.1. 시스템 개요

냉동 컨테이너의 부피가 크기 때문에 Lab 테스트를 용이하게 하기 위하여 냉동 컨테이너의 동작 알고리즘을 내장한 소형 보드를 제작하였다. 냉동 컨테이너의 Interrogator Port와 동일한 RS-232 프로토콜을 사용하여 통신모듈로부터 데이터를 수신하면 제조사와 명령을 비교하여 그에 해당하는 데이터를 통신모듈로 송신한다. 하나의 Simulator를 통하여 모든 냉동 컨테이너의 역할을 할 수 있도록 하기 위하여 동작 모드를 나누어 전환 가능하게 하였고, 동작하는 상황을 쉽게 인지하기 위하여 LCD와 FND를 통하여 간단한 정보를 출력하게 하였다.

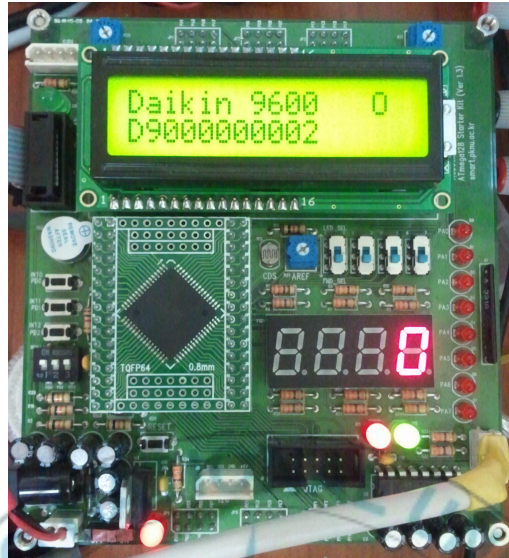


그림 10. Simulator 모듈

4.2. 동작 구조

Simulator는 메인 프로세서로 Atmega128을 사용하였고, Daikin, Carrier, Thermoking, Starcool 등의 제조사별 모드와 CID, 설정 온도 등을 설정할 수 있는 Setting 모드의 총 5가지의 동작 모드로 동작한다. 각 제조사별 모드는 외부 버튼 1으로 동작하고, Setting 모드는 외부 버튼 2로 동작한다. 각 모드는 RXD 인터럽트를 사용하며, 한 바이트를 수신할 때마다 타이머 인터럽트로 시간을 체크하여 데이터 수신 종료를 체크한다. 수신한 데이터가 해당 모드의 명령일 때만 그에 맞는 데이터를 송신하고, 해당하는 명령이 아닐 경우는 올바른 데이터가 들어올 때 까지 버퍼를 덮어쓰며 수신 대기한다.

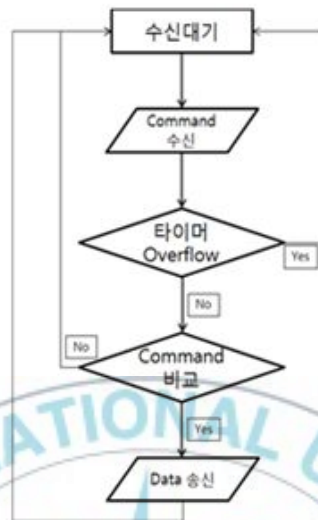


그림 11. 각 제조사 모드의 Flow Chart

Setting 모드는 냉동 컨테이너의 CID, 설정 온도 등의 정보를 불러오거나 EEPROM에 저장하여 Simulator의 전원이 켜짐과 동시에 EEPROM을 체크하여 저장된 상태로 부팅이 가능하게 하는 동작을 한다. 그림 12는 Simulator의 Setting 모드와 호환하여 동작하는 프로그램으로 CID, 각 컨테이너의 알람 데이터 유무, 설정온도, 초기 시작 모드 등을 설정하거나 저장되어있는 설정 환경을 불러올 수 있다.

or		Daikin 9600	Daikin 38400	Carrier	Thermoking	Starcool	Start Mode		
CID Num	Event : Present	None	Present	None	None	Starcool	Load	Set	
2	SetPoint : -18	-12.5	-20	-16.5	-25				

그림 12. Simulator Setting 프로그램

EEPROM에 저장된 데이터를 기반으로 부팅이 완료되고 나면, 각 제조

사 모드는 각 설정온도에서 $\pm 5.0^{\circ}\text{C}$ 의 오차를 가지고 임의의 현재 온도 값과 공급 온도 값을 송신함으로써 실제 냉동 컨테이너와 비슷한 환경을 구성할 수 있다.

4.3. 동작 확인

실제 냉동 컨테이너와 Simulator의 데이터 규격과 알고리즘의 일치를 확인하기 위하여 실제 냉동 컨테이너에서 동작하는 Viewer Program을 제작하고, 이를 Simulator과 연동하여 동작여부를 체크하였다. 그림 13은 Starcool의 냉동 컨테이너 Simulator를 확인한 장면이다. EEPROM에 저장되어있는 환경과 실제 데이터를 수신하여 변환한 값이 일치함을 볼 수 있다.

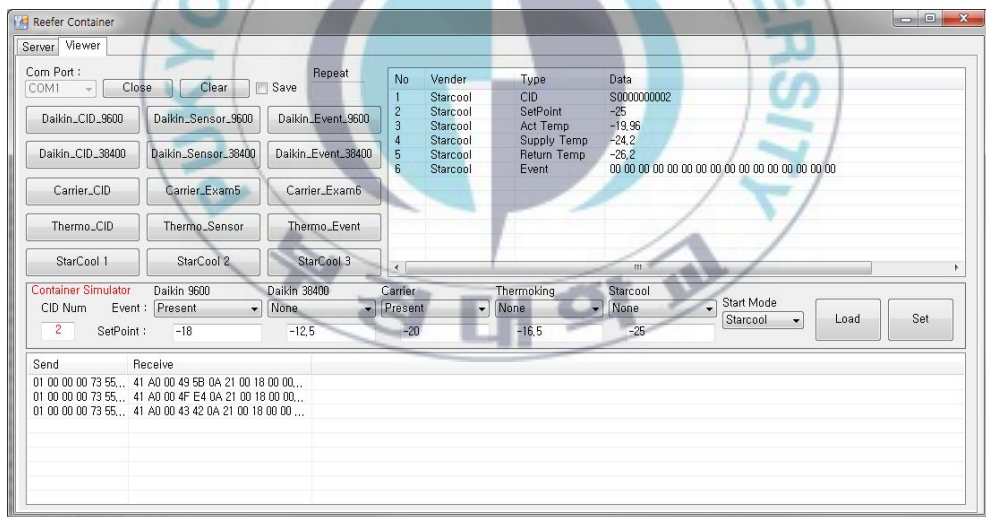


그림 13. Simulator 확인

Ⅲ. 모바일 서비스

1. 시스템 개요

모바일 서비스는 모바일 단말기로부터 냉동 컨테이너 상태 정보를 열람하기 위한 서비스로 모바일 단말기와 서버는 WiFi를 통하여 연결된다. 모바일 단말기에서 열람할 수 있는 정보는 야적장 구역의 전체 냉동 컨테이너에 대한 현재 상태, 냉동 컨테이너의 상태 정보에 대한 과거 기록, 오류나 고장이 발생한 냉동 컨테이너에 대한 정보이다. 본 논문에서 모바일 어플리케이션을 위하여 사용한 OS는 안드로이드 2.3.4와 IOS 4.3이다. 모바일 어플리케이션의 메인은 그림 14와 같이 구역별로 컨테이너를 정리하여 볼 수 있는 'CELL'메뉴, 제조사별로 정리하여 볼 수 있는 'VENDOR'메뉴, 컨테이너 고장여부에 관하여 열람할 수 있는 'ALARM'메뉴, 모바일 어플리케이션의 설정정보를 변경할 수 있는 'SETTING'메뉴로 구성되었다.



그림 14. 모바일 어플리케이션 메인

2. Socket_Client 클래스

모바일 단말기가 서버의 데이터베이스에 있는 정보를 모두 가져오기 위해서는 6가지의 방법이 필요하다. 안드로이드에서는 각 함수가 실행될 때 소켓연결을 시도하며, 연결이 되면 해당 정보를 수신하고 통신을 끊는다. 그리하여 불필요한 통신 연결 상태를 없애고 오류를 발생시킬 가능성을 낮추었다. 표 3은 Socket_Client 클래스에서의 함수 목록이다. 사용자의 요구에 맞게 해당 함수가 실행되어 정보를 표시한다.

No	함수	내용
1	public String[] Get_CellList()	구역 리스트
2	public String[] Get_CellInfo(String Cell)	해당 구역 현재 상태
3	public String[] Get_ConHistory(String CID)	컨테이너 과거 정보
4	public String[][] Get_AllConInfo()	모든 구역의 현재 상태
5	public String[][] Get_Alarm_Cell()	각 구역의 알람 데이터
6	public String[] Get_Alarm_History()	모든 구역의 알람 데이터

표 5. Socket_Client 함수 목록

IOS에서는 여러 Network계층 중에서 CFNetwork레이어의 CFSocket을 사용하여 Socket의 생성, 연결차단, 데이터 전송 처리를 위한 Callback함수로 구성하였다. 필요한 데이터는 표 4의 명령어를 전송하여 수신한다.

3. 서비스와 브로드캐스트 리시버

안드로이드에서는 서비스와 브로드캐스트 리시버라는 개념을 사용한다[12]. 서비스는 실제 어플리케이션이 실행되고 있지 않더라도 백그라운드에서 실행되는 동작 등을 말하며, 본 논문에서 구현된 어플리케이션에

서는 알람 데이터를 정해진 시간마다 새로 수신하기 위하여 사용한다. 서비스에서 알람 데이터를 계속하여 받아오기 위하여 AlarmManager를 사용한다. AlarmManager는 안드로이드에서 원하는 시간에 작업을 수행하기 위하여 사용되는데, AlarmManager를 통해 설정할 수 있는 알람의 형식은 4가지가 있다.²⁾ 그 중 본 논문에서 사용한 형식은 부팅 후 시간을 기준으로 카운트 하여 정해진 시간이 지나면 Intent를 발생시키는 방식으로 ELAPSED_REALTIME를 사용하였다. Intent가 발생하면 서버로부터 알람 데이터를 수신한다. 수신한 알람 데이터는 단말기의 데이터베이스를 확인하여 새로운 알람 데이터인지 확인하고, 새로운 알람 데이터나 확인하지 않은 알람 데이터가 있을 경우 Notification Bar에 알람을 발생하여 사용자에게 알린다.

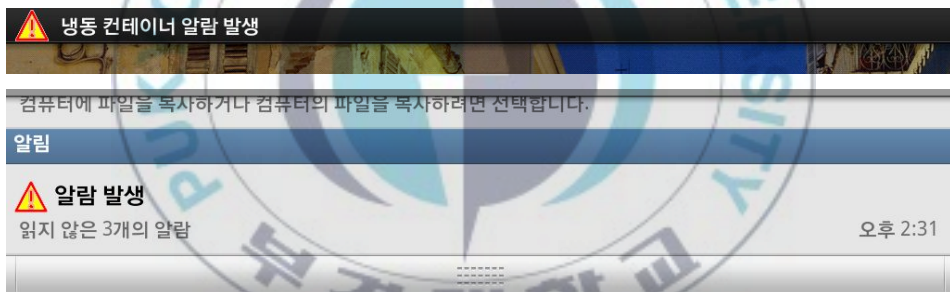


그림 15. Notification

브로드캐스트 리시버는 단말기에서 특정 이벤트가 발생하였을 때, 이 이벤트를 수신하여 원하는 동작을 하기 위하여 사용된다. 본 논문에서의 어플리케이션에서는 해당기기의 부팅이 완료되었을 때 알람 데이터를 받아오기 위한 서비스를 실행하기 위하여 사용된다.

2) RTC, RTC_WAKEUP, ELAPSED_REALTIME, ELAPSED_REALTIME_WAKEUP가 있다.

4. CELL

CELL메뉴는 냉동 컨테이너 정보를 구역별로 정리하여 나타낸다. 먼저 서버로부터 구역정보를 수신하여 화면에 나타낸다. 안드로이드에서는 CELL Activity가 실행되면 표 3의 1번 함수를 실행하여 구역 목록을 수신한 뒤 ListView에 나타낸다.

A01
A02
A03
A04
A05

그림 16. 구역 목록 (안드로이드)

Listview의 항목을 선택하면 표 3의 2번 함수를 실행하여 해당 구역 내의 모든 냉동 컨테이너의 현재 상태를 불러온다. 수신한 데이터를 항목별로 나누어 Container Class에 할당하여 Container Item을 구성한다. 구성된 container ArrayList는 Custum Listview Activity가 실행되면서 항목별로 나타낸다.

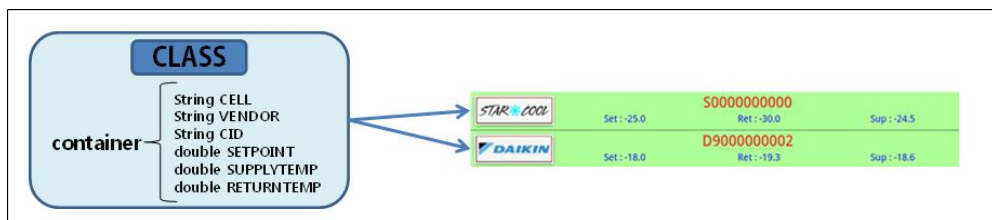


그림 17. Container Item

IOS에서는 CELL Action이 실행되면 서버로부터 구역정보를 수신하여 TableView에 나타낸다.



그림 18 구역 목록 (IOS)

IOS에서도 TableView의 항목을 선택하면 해당 구역 내의 모든 냉동 컨테이너의 현재 상태를 요구하게 되고, 수신한 데이터는 항목별로 나누어 NSMutableArray에 할당하고, 할당된 데이터는 CustomTableView가 실행될 때 항목별로 나타낸다.

5. VENDOR

VENDOR메뉴는 모든 냉동 컨테이너의 상태정보를 제조사별로 정렬하여 나타낸다. 안드로이드에서는 VENDOR Activity가 실행되면 표 3의 4번 함수를 실행하여 정보를 수신한다. 함수가 실행되면 2차원 배열이 생성되고, 배열의 구성은 그림 19와 같다.



그림 19. AllConInfo 데이터 구조

수신한 데이터는 가독성을 높이기 위하여 구역별로 나누어 표시하는데, TabHost와 ExpandableListView를 사용하여 제조사와 구역을 분리한다. 수신한 데이터를 구역과 제조사별로 ArrayList와 HashTable을 생성하고, Custom ExpandableListView를 사용하여 화면을 구성한다. 구성한 화면은 그림 20과 같다.

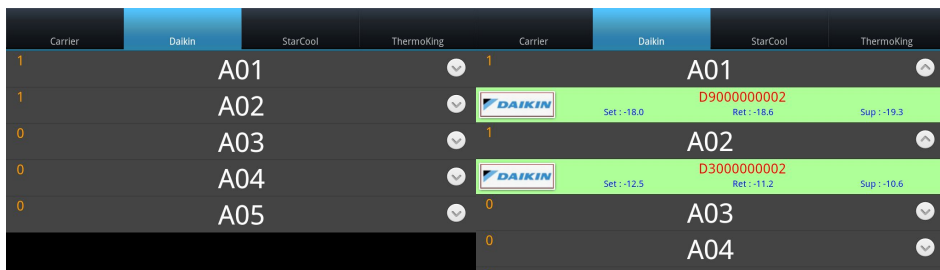


그림 20. VENDOR 화면 (안드로이드)

IOS에서는 VENDOR Action이 실행되면, 수신한 정보를 각 제조사별로 분리하여 NSMutableArray에 할당하고, TabBar의 항목을 제조사로 두어 안드로이드와 동일한 화면을 제공한다.

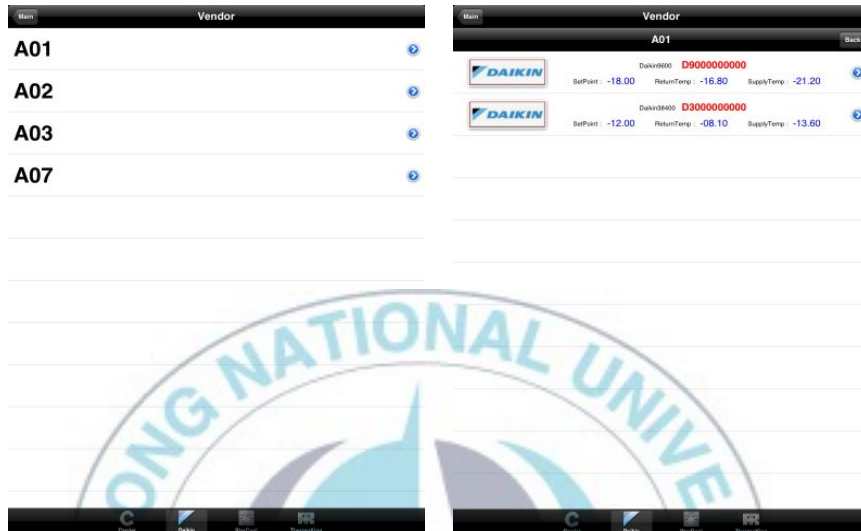


그림 21. VENDOR 화면 (IOS)

6. Container History Activity

Cell메뉴와 Vendor 메뉴는 모두 Container Item을 포함한다. 이 Item을 클릭하게 되면 해당 냉동 컨테이너의 과거 정보를 열람하게 된다. Container History Activity가 실행되면 표 3의 3번째 함수를 실행하게 되고, 최대 48개의 데이터를 수신한다. 수신한 데이터는 안드로이드에서는 그래프와 테이블 표시하고, IOS에서는 테이블로만 제공한다. 그림 22는 Carrier社의 'C00000000002'라는 CID를 가지는 냉동 컨테이너에 대한 과거 데이터를 안드로이드의 그래프로 나타낸 것이다.³⁾

3) 'C00000000002'는 Simulator의 제조사 구분을 쉽게 하기 위하여 임의로 지어진 CID이다.

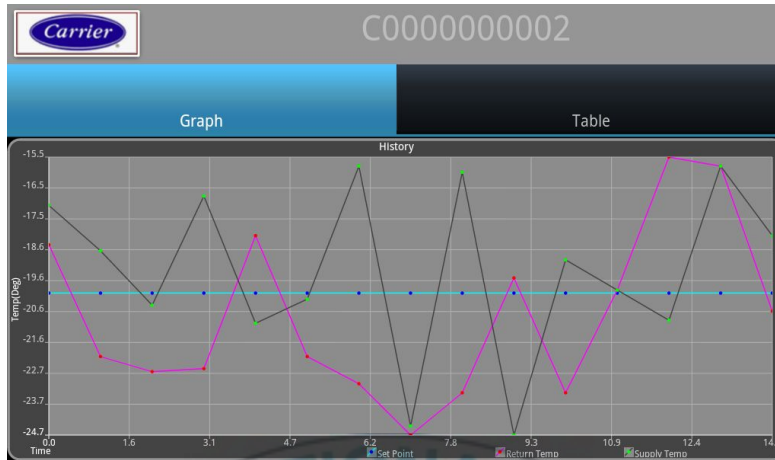


그림 22. 냉동 컨테이너 과거 데이터 표시

7. ALARM

Alarm메뉴는 냉동 컨테이너의 고장여부를 확인하는 화면으로, 안드로이드와 IOS 모두 서버의 데이터베이스에 존재하는 알람 데이터를 수신하여 표시한다. 안드로이드에서는 표 3의 5번째 함수를 실행하여 나타내며, 서비스에서 정해진 시간마다 수신하는 알람데이터에서 발생한 Notification 또한 Alarm Activity로 연결된다. IOS는 안드로이드의 서비스와 동일한 기능이 없어서 Alarm메뉴를 선택하였을 때만 알람 데이터를 수신한다. 확인한 알람 데이터와 확인하지 않은 알람데이터를 구분하기 위하여 수신한 알람 데이터는 SQLite를 사용하여 데이터베이스에 저장한다. 수신한 데이터를 데이터베이스의 데이터와 비교 한 후 같은 내용이 있을 경우에는 해당 항목의 Read 인덱스를 참조하여 확인한 알람 데이터와 확인하지 않은 알람 데이터를 구분하여 표시하며, 다른 내용일 경우에는 데이터베이스에 새로운 알람 데이터를 저장한다. Alarm Activity의 화면 구성은 그림 23과 같다.



그림 23. 알람 리스트

8. SETTING

SETTING메뉴는 어플리케이션에 대한 전체 환경설정을 담당하는 메뉴로, 안드로이드는 알람 데이터를 수신하는 시간, 서비스 실행여부, 서버의 IP설정을 포함하고, IOS에서는 IP설정을 포함하였다. 모든 정보들은 System Property에 저장하여 어플리케이션이 종료되더라도 정보들을 다시 불러들일 수 있도록 한다.

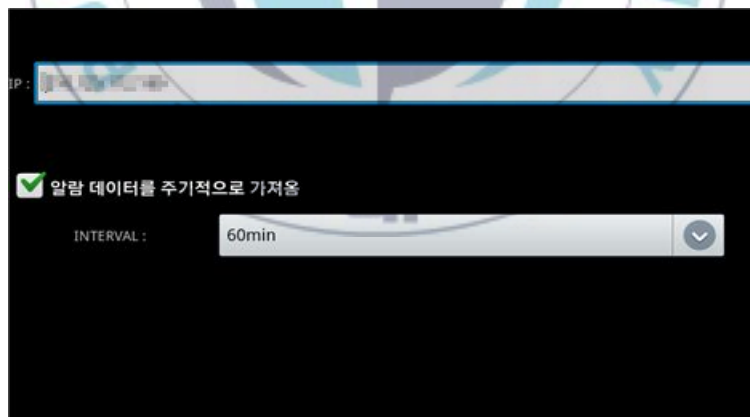


그림 24. SETTING Activity

IV. 실험

냉동 컨테이너 모니터링 시스템을 실험하기 위하여 컨테이너 터미널에서 실제 냉동 컨테이너를 대상으로 한 Field Test와 실내에서 Simulator를 대상으로 한 Lab Test를 진행하였다.



그림 25. 필드 테스트



그림 26. Lab 테스트

데이터베이스에 저장된 데이터 중 하나의 냉동 컨테이너에 대한 온도 데이터를 살펴보면 그림 27과 같다.

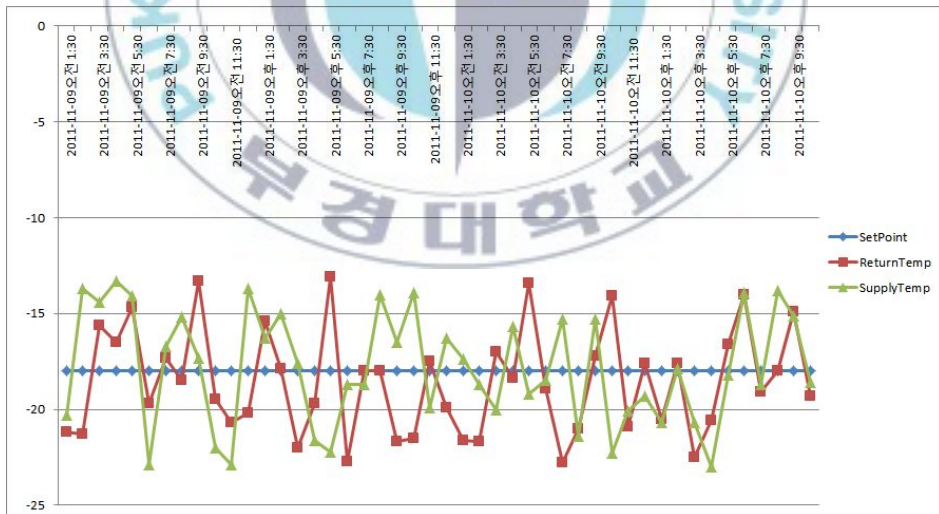


그림 27. 냉동 컨테이너 온도 변화

데이터베이스의 내용과 어플리케이션의 내용의 비교해 보면 아래 그림 28, 그림 29와 같다.



그림 28. 데이터베이스와 어플리케이션의 온도 데이터 비교

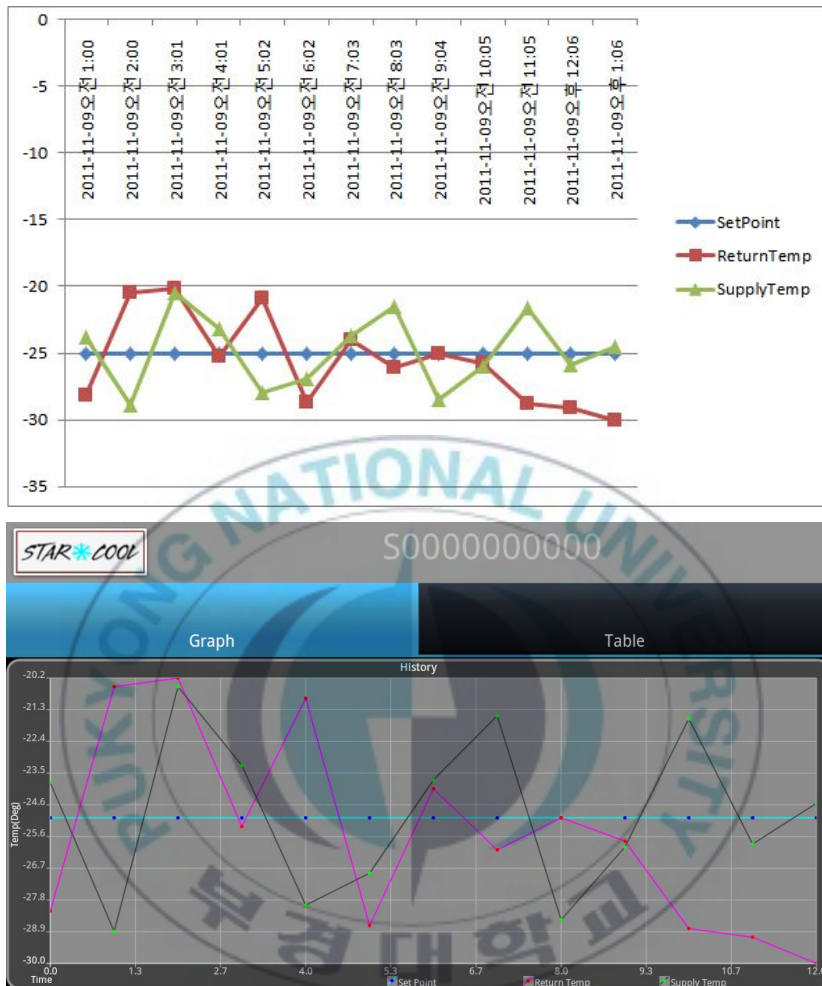


그림 29. 데이터베이스와 어플리케이션의 온도 그래프 비교

데이터베이스와 어플리케이션을 비교해보았을 때 서버와 어플리케이션에서 동일한 데이터를 확인할 수 있었고, 따라서 서버에서 실시간으로 데이터를 확인하는 작업과 동일한 모니터링 환경을 모바일 어플리케이션에서 제공할 수 있다는 것을 알 수 있다.

V. 결론

본 논문에서는 냉동 컨테이너 모니터링 시스템을 위해 Interrogator Port를 사용하여 데이터를 수집하여 서버의 데이터베이스에 저장하고, 서버와 모바일 서비스를 융합하여 냉동 컨테이너의 관리를 보다 효율적으로 할 수 있는 방법을 제안하였다. 데이터를 수집하기 위하여 사용한 Interrogator Port는 기존의 PCT방식이나 4-Pole방식에 비해 데이터의 신뢰도나 장착율에서 높은 성능을 보여준다. 서버는 냉동 컨테이너의 상태 데이터를 실시간으로 확인하고 통합 관리하면서, 냉동 컨테이너를 관리하기 위해 필요한 인력이나 시간의 절약을 할 수 있다. 모바일 서비스는 시간, 장소에 구애받지 않는 모니터링 환경을 제공한다. 실험을 통하여 서버와 어플리케이션에서 동일한 냉동 컨테이너 모니터링 환경을 확인할 수 있었다. Interrogator Port를 이용한 모니터링은 데이터 수집을 위한 단자를 연결하기 위해 유선의 플러그가 필요하며, 이는 길이와 무게에 대하여 단점으로 지적된다. 성능 개선을 통하여 길이와 무게에 대한 단점을 극복할 수 있는 연구가 필요할 것으로 판단된다.

최근 IT기술의 발달로 터미널의 자동화, 지능화가 요구되고 있다. 냉동 컨테이너 모니터링 시스템은 관리자와 고객 모두에게 더 나은 냉동 컨테이너 관리 환경을 제공할 것이고, 모바일 서비스를 통해 그 활용도는 더 커질 것이다. 본 논문에서 제안한 냉동 컨테이너 모니터링 시스템과 모바일 서비스는 항만 물류 전자화, IT화를 위한 유비쿼터스 항만 환경을 구현하기 위한 하나의 프로토 타입으로써 의미가 있다고 할 수 있다.

VI. 참고문헌

- [1] <http://www.busanpa.com>
- [2] 전준수, 종합물류의 이해, 박영사, 2007.
- [3] Ruiz García, Luis, “Development of monitoring applications for refrigerated perishable goods transportation”, E.T.S.I. Agrónomos (UPM), 2008.
- [4] ISO 10368, “Freight thermal containers - Remote condition monitoring”, 1992.
- [5] Nishioka, J. Tsuzuki, S. Yoshida, M. Kawasaki, H. Shinpo, T. Yamada, Y., “Power Line Communications and Its Applications”, pp217-222, 2009.
- [6] 해양수산부, 과학기술부, “컨테이너 터미널 통합 운영시스템 개발 : 최종 보고서”, 해양수산부, 2003.
- [7] DRINGY WILD, “Comparison between porthole and integrated reefer system containers from technical, operational and economic aspects”, Proceedings of the IIF-IRF Commission D, 1994.
- [8] Zimmermann, M, Dostert, K, “Analysis and modeling of

impulsive noise in broad-band powerline communications”, IEEE Electromagnetic Compatibility Society, pp.249 - 258, 2002.

[9] 양창호, “차세대 컨테이너터미널 운영시스템의 기술개발 방향과 전략수립에 관한 연구”, 한국해양수산개발원, 2003.

[10] <http://www.atmel.com>

[11] <http://www.wiznet.co.kr>

[12] Reto-Meier, Professional Android Application Development, Jpub, 2009.



감사의 글

어느덧 졸업논문의 마지막을 적고 있지만, 졸업이라는 것이 아직은 실감이 나지 않습니다. 학부 4학년 2학기라는 늦은 시기에 연구실 생활을 시작하여 짧은 기간이나마 많은 것을 배우고 나가게 되었습니다. 대학교 1학년부터 석사 2년까지 무사히 졸업할 수 있도록 도와주신 분들에게 감사의 글을 올립니다.

언제나 학생들을 진심으로 대해주시고, 모든 부분에서 많은 도움을 주신 이원창 교수님께 우선 감사를 드립니다. 짧은 시간이었지만, 저에게 있어서 많은 부분에 대하여 느끼고, 또 생각하게 하는, 존경이라는 단어가 어색하지 않은 분이셨습니다. 항상 감사하게 생각하고 있습니다.

그리고 강근택 교수님, 항상 편하게 대해주시고, 웃음으로 대해주셔서 연구실 생활을 더 즐겁게 할 수 있었던 것 같습니다. 논문 심사를 도와주신 자매연구실인 CAL 연구실의 주문갑 교수님께도 감사의 말씀 올립니다.

세 분 모두 학교생활 뿐 아니라 생활 전반적인 부분에서 많은 것을 배우고, 많은 도움이 되었습니다. 항상 감사하는 마음 잊지 않고 살아가겠습니다.

그리고 제가 생활한 연구실의 틀을 잡아주신 연구실 선배님들, 일감으로 항상 귀찮게 해드렸던 준영 선배, 동아리 선배이자 연구실 선배 현중 선배, 처음에는 무섭기만 했지만, 이제는 든든한 선배로 느껴지는 기주

선배, 짧게나마 연구실 생활을 같이 했지만 많은 것을 배울 수 있었던 부건 선배, 태형 선배, 그나마 오래 같이 생활해서 항상 귀찮게 해드렸던 미국가신 재연 선배, 많은 도움을 주진 못했지만 잘하고 있는 작은 준영이 에게도 고마움을 전합니다.

항상 고민거리가 있으면 위로해주고 상담해주던 유종이, 동기지만 선배같은 순태, 늦게 알게 되었지만 많이 친해진 재식이 형, 동아리 때부터 학교생활을 같이 해온 잘생긴 동주, 연구실은 나보다 선배인 국진이, 다른 연구실 석사 선배인 친구 현수, 조금은 어색한 중백이, 또 다른 패밀리 정훈이, 호현이, 채한이, 근교, 재준이, 엘리트 후배 지현, 재밌는 애령이 등, 학교생활의 좋은 추억을 남길 수 있게 해준 친구들과 후배, 동생들에게도 감사드립니다.

항상 감사하고 죄송한 부모님, 대화가 많지는 않지만 많은 걱정 해주시는 아버지, 매번 자식이 우선인 어머니에게 무한한 감사를 드립니다. 부모님들 덕분에 가족의 소중함을 깨우치게 되었고, 앞으로도 가족들을 위해서 살겠다고 다짐하게 되었습니다.

마지막으로 3년을 훨씬 넘긴 긴 시간동안 항상 옆에서 힘이 되어준 현경이 에게 마지막으로 감사하고 고맙다는 말을 전하고 싶습니다.