

석사 학위 논문

영어 상황학습을 위한
RFID 미들웨어 및 응용 연구

지도교수 김 성 백



제주대학교 교육대학원

컴퓨터교육전공

양 경 미

2006년 8월

석사 학위 논문

영어 상황학습을 위한
RFID 미들웨어 및 응용 연구

지도교수 김 성 백



제주대학교 교육대학원

컴퓨터교육전공

양 경 미

2006년 8월

영어 상황학습을 위한
RFID 미들웨어 및 응용 연구

지도교수 김 성 백

이 논문을 교육학 석사학위논문으로 제출함

2006년 6월 일

제주대학교 교육대학원 컴퓨터교육전공



양경미의 교육학 석사학위논문을 인준함

2006년 6월 일

심사위원장 _____ 조 정 원 _____ 인

심 사 위 원 _____ 김 성 백 _____ 인

심 사 위 원 _____ 김 철 민 _____ 인

영어 상황학습을 위한 RFID 미들웨어 및 응용 연구

양 경 미

제주대학교 교육대학원 컴퓨터교육전공

지도교수 김 성 백

유비쿼터스 사회(Ubiquitous Society)의 핵심기술인 RFID(Radio Frequency Identification)를 물류, 유통, 교통, 의료 등 다양한 분야에서 개발, 적용하는 연구가 활발하게 진행되고 있다. 그러나 RFID를 이용하여 U-Campus, U-Library 등 유비쿼터스 교육환경을 마련해주는 연구나 직접적인 학습에 적용한 연구는 아직 미미한 실정이다. 본 연구에서는 RFID의 특성을 살려 세계화에 필수적인 영어를 학습하는 방안을 제시하고 있다. RFID 태그와 리더의 무선 통신 기술을 이용하여 학습자의 위치와 상황을 인지하고 그에 부합하는 영어 상황 학습 서비스를 제공하고자 한다. 이로 인해 학습자는 위치와 상황에 따른 영어 학습을 할 수 있다.

RFID 시스템을 이용하기 위해 본 연구에서 제안하는 RFID 미들웨어는 기존의 범용 RFID 미들웨어와 달리 PDA 기반에서 동작하는 모바일 RFID 미들웨어로 필수 API(리더 제어, 태그 제어, 큐 관리, 필터관리)를 중심으로 개발하여 최적화하였다. 개발한 RFID 미들웨어는 상단부분(Upper Layer)과 하단부분(Lower Layer)으로 구성되어 있다. RFID 미들웨어의 상단부분은 영어 상황학습을 중심으로 RFID 응용에서 필요한 요소의 API를 정의하고, 응용 프로그램과 RFID 미들웨어의 하단부분과의 연결고리 역할을 수행한다. RFID 미들웨어의 하단부분은 시리얼 통신을 통해 디바이스 드라이버의 역할을 수행하는 부분으로, RFID 리더와의 통신 프로토콜을 정의하고 리더와 미들웨어 상단부분을 매개한다. RFID 미들웨어를 상단부분과 하단부분 나누는 구조로 개발한 이유는, 디바이스 독립성을 갖도록 RFID 리더가 바뀌었을 경우 미들웨어 하단부분의 수정만으로 상단 미들웨어와 응용이 그대로 동작하게 하기 위함이다.

영어 상황학습 시스템은 문장, 해석, 단어, 나만의 단어장, 단어검색 등 크게 5가지 기능을 가지고 있다. RFID 미들웨어를 통해 전송된 학습자의 위치와 상황에 따른 영어 문장과 단어를 제시하고 제시된 문장에 대한 해석보기 기능을 제시해준다. 그리고 제시된 영어단어 중 학습자가 모르는 단어 또는 학습하고자 하는 단어를 자신만의 단어장에 등록, 추가, 삭제할 수 있다. 학습자는 학습 도중 제시된 단어 이외의 단어를 알고자 할 때에 단어검색을 통해 영어사전에서 검색된 결과를 자신만의 단어장에 등록하여 학습할 수 있다.

※ 본 논문은 2006년 6월 제주대학교 교육대학원 위원회에 제출된 교육학 석사학위 논문임.

목 차

I. 서론	1
1. 연구의 배경 및 목적	1
2. 연구의 내용과 방법	3
II. 이론적 배경	4
1. RFID 소개	4
2. 상황학습 이론	8
3. 영어마을	9
III. 영어 상황학습을 위한 RFID 미들웨어	11
1. 미들웨어 설계	11
2. 미들웨어 구현	20
IV. 영어 상황학습을 위한 응용 시스템	27
1. 응용 시나리오	27
2. 응용 시스템 설계	32
3. 응용 시스템 구현	37
V. 결론 및 추후연구	41
참고문헌	43
<Abstract>	44
<부록> RFID 미들웨어 상단 API	46

표 목 차

표 1. RFID 주파수 대역	5
표 2. 매체별 인식기술 비교	6
표 3. 컬러코드와 RFID의 비교	7
표 4. 리더 제어	12
표 5. 태그 제어	14
표 6. 큐 관리	14
표 7. 필터 관리	15
표 8. 시리얼 포트 제어	16
표 9. 호스트와 리더간의 명령어	17
표 10. 에러 코드	19
표 11. U-Learning API	20
표 12. PDA 시스템 구성	21
표 13. 제시되는 영어문장 예	30
표 14. 제시되는 영어단어 예	30



그림 목차

그림 1. RFID 시스템 구성요소	4
그림 2. 컬러코드	7
그림 3. 영어마을의 3E 학습	9
그림 4. RFID 미들웨어 설계	11
그림 5. RFID 미들웨어 동작과정	21
그림 6. 초기화/콜백 등록시	22
그림 7. RM_Open()	23

그림 8. RM_Read()	24
그림 9. RM_ReadData()	24
그림 10. RM_ReadComport()	25
그림 11. RM_WriteData()	26
그림 12. 병원_접수	28
그림 13. 병원_진료	28
그림 14. 약국_조제	29
그림 15. 액션카드	31
그림 16. 영어 상황학습 시스템 구조	32
그림 17. 데이터베이스 구조	36
그림 18. 로그인 화면	37
그림 19. RFID 태그 인식과 문장 제시화면	38
그림 20. 해석보기 화면	39
그림 21. 문장 속에 단어내용 보기	39
그림 22. 단어 제시 화면	39
그림 23. 나만의 단어장	39
그림 24. 단어 검색 화면	40
그림 25. 나만의 단어장에 중복 단어 추가 시 경고	40



I. 서론

1. 연구 배경 및 목적

세계화, 국제화에 발맞춰 영어교육의 필요성은 점점 증대되고 있다. 초등학교 3학년에서부터 고등학교를 졸업하는 9년간 영어공부를 하지만 원어민과의 대화에서 자신감을 가지고 의사소통을 원활하게 하는 학생은 일부에 속한다. 중·고등학교에서의 영어공부는 입시위주의 문제 풀이 식에 지나지 않고 대학에 들어와서는 많은 학생들의 자신의 영어 실력을 제대로 알지 못해 기초 영어 문법부터 다시 시작하는 모습을 보였다. 우리나라 영어교육이 입시에 초점이 맞춰져 정작 의사소통을 목적으로 하는 언어로서의 영어의 본질을 흐리게 만들었다.

언어는 생활 문화인 동시에 습관이다. 즉, 영어를 생활화하여야 한다. 하지만 한국은 인종적으로 단일 민족일 뿐만 아니라 언어적으로도 단일한 환경을 유지해 왔기 때문에 영어가 사회적인 상호작용을 하는데 거의 쓰이지 않고 있다. 따라서 우리나라와 같이 영어에 대한 지속적인 노출 환경을 만들어 줄 수 없는 단일 언어국가에서 영어를 배우는 것은 홍콩과 같은 이중 언어 국가에서 영어를 배우는 것보다 훨씬 어렵다[1].

따라서 본 연구에서는 이러한 문제점을 해소하기 위해 유비쿼터스 사회의 핵심기술인 RFID(Radio Frequency Identification)을 이용하고자한다. 그러기 위해서 RFID를 교육에 적용한 기존 연구들을 살펴볼 필요가 있다.

강명곤외 2명[2]은 U-캠퍼스에서의 도서 대출 시스템, 열람실 좌석관리, 학생 서비스 시스템(스마트카드 시스템)을 제시하고 있으며 U-캠퍼스에서의 도서대출시스템은 기존 바코드 시스템의 갖는 단점을 보완한 것이었다. 바코드 시스템의 단점으로는 바코드가 부착된 사물에 바코드 리더기를 가까운 거리까지 접근 시켜야 하는

점(짧은 인식거리)과 바코드를 사물에 부착하면 부착된 바코드를 떼어 내어 다른 사물에 바코드를 다시 부착하여 사용할 수 없는 점(바코드 재사용 불가), 한 번에 한 개의 사물만 인식하는 점을 들 수 있다. 인식거리가 길고 비접촉식 방식으로 동시에 여러 개의 자료 인식이 가능하고 반영구적으로 활용 가능한 RFID를 적용하여 소장 자료에 대한 관리를 쉽게 할 수 있게 하고 이용자에게도 편리한 자가 대출 및 반납 서비스를 제공할 수 있게 하였다.

열람실 좌석관리 시스템은 RFID 칩이 내장된 학생증카드로 사용자 인증을 거친 후에 사용자가 자율적으로 좌석을 지정하여 사용하는 시스템을 말하며 열람 좌석 이용의 기회 균등을 제공할 수 있게 하였다.

학생 서비스 시스템(스마트카드 시스템)은 RFID칩을 내장한 스마트카드로 신분증 기능이외에 현금(직불)카드 기능, 전자출결기능, 도서관 도서 대출 기능 및 열람실 좌석 관리기능, 식당 및 매점을 이용할 수 있는 전자화폐기능, 교통카드 기능 등의 다양한 서비스를 원스톱으로 제공하여 학생, 교수, 교직원들의 학내 생활을 편리하게 이용할 수 있게 하였다. 위 시스템들은 RFID를 이용하여 유비쿼터스 교육환경을 마련해 준 것에 불과하였고 Hiroaki Ogata와 Yoneo Yan[3]은 TANGO(Tag Added Learning Objects)라는 영어 어휘 학습 시스템을 제시하였다. TANGO 시스템은 언어를 새로 시작하는 초보자들에게 주변 물체에 라벨을 붙여 단어를 상기 시켜주는 학습이 언어학습에 도움이 된다는 것에 착안하여 개발되었다. TANGO는 학습자 주변의 물체에 RFID 태그를 붙여 학습자가 PDA에 장착된 RFID 리더를 통해 RFID 태그 값을 읽어 들여 어휘 학습을 할 수 있도록 제안하고 있다. TANGO는 RFID 태그가 붙여진 사물을 이용하여 어휘 학습을 하는 것으로 학습자가 특정 공간의 사물을 모두 학습하면 학습할 새로운 사물이 필요하다는 단점이 있었다.

따라서 본 논문에서는 RFID 장점을 분석하고 이를 영어 상황학습에 접목시켜 교육의 효과를 높일 수 있는 방법을 제시하고자 한다.

2. 연구의 내용과 방법

본 연구에서 수행되는 연구의 내용과 방법은 다음과 같다.

첫째, RFID 시스템이 갖는 특성을 바탕으로 효과적으로 영어 상황학습을 하기 위한 PDA기반의 RFID 미들웨어를 연구한다. 영어 상황학습에서 요구되는 미들웨어 API들을 정의하고 RFID 미들웨어 내부 구조를 설계 및 구현한다. 특히, 영어 상황학습에서 요구되는 별도의 U-learning API들을 새롭게 정의하고 각 API를 구현한다.

둘째, RFID 미들웨어 기반의 영어 상황학습을 위한 응용 시스템을 연구한다. RFID 미들웨어에서 제공되는 기능을 이용하여 영어 상황학습 서비스를 학습자에게 제공하기 위한 응용 시스템을 설계 및 구현한다. 구체적으로 영어 상황학습을 위한 응용 시나리오와 RFID 미들웨어 API를 이용하여 학습자에게 필요한 기능들을 개발한다.



II. 이론적 배경

1. RFID 소개

RFID(Radio Frequency Identification)는 마이크로 칩을 내장한 태그에 저장된 데이터를, 무선주파수를 이용하여 리더에서 비접촉식으로 자동으로 인식하는 기술을 말한다. 필요한 모든 것(곳)에 RFID 태그를 부착하고 이를 통하여 기본적인 사물의 인식정보 뿐만 아니라 주변의 상황정보, 환경정보까지도 인식가능하다.



<그림 1> RFID 시스템 구성요소

RFID 시스템은 <그림 1>과 같이 RFID 태그와 리더, 리더를 지원하는 호스트로 구성된다. 태그는 메모리와 안테나를 포함하고 있으며, 메모리에 저장된 정보를 RFID 리더로 전송하는 기능을 수행한다[4].

RFID 태그는 크기와 모양이 다양하며, 전원공급 유무에 따라 수동형(Passive)태그와 능동형(Active)태그로 분류되며 사용하는 주파수 대역에 따라 <표 1> 과 같이 저주파(125KHz, 134KHz), 고주파(13.56MHz), 극초단파(433.92MHz, 860~960MHz), 마이크로파(2.45GHz)등으로 분류된다[5].

수동형 태그는 태그에 배터리가 없으며 능동형 태그에 비해 가볍고, 가격도 저렴

하면서 반영구적으로 사용가능하다. 하지만 인식거리가 짧고 RFID 리더에서 많은 전력을 소모하는 단점이 있으며, 능동형 태그는 태그에 배터리가 부착되어 있고 리더와의 인식거리를 멀리할 수 있는 장점이 있으나 수동형에 비해 고가인 단점이 있다.

<표 1> RFID 주파수 대역

주파수	저주파	고주파	극초단파		마이크로파
	125,134KHz	13.56MHz	433.92MHz	860~960MHz	2.45GHz
인식거리	60cm미만	60cm까지	~50~100m	~3.5~10m	1m이내
일반특성	-비교적 고가 -환경에 의한 성능저하의 없음	-저주파보다 저가 -짧은 인식거리와 대용량 태그인용 분야에 적합	-긴 인식거리 -실시간 추적 및 컨테이너 내부 습도, 풍격 등 환경 센싱	-IC기술 발달로 가장 저가로 생산가능 -다중태그인식거리와 성능이 가장 뛰어남	-900대역 태그와 유사한 특징 -환경에 대한 영향을 많이 받음
동작방식	-수동형	-수동형	-능동형	-능동/수동형	-능동/수동형
적용분야	-공정자동화 -출입통제/보안 -동물관리	-수화물관리 -대여 물품 관리 -교통카드 -출입통제/보안	-컨테이너 관리 -실시간 위치 추적	-공급망 관리 -자동통행료 징수	-위조방지
인식속도	저속 ----- 고속				
환경영향	강인 ----- 민감				
태그크기	대형 ----- 소형				

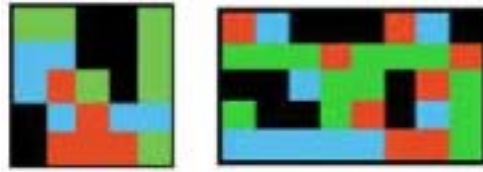
<표 2>와 같이 RFID와 기존 사용되고 있는 인식 매체(바코드, 자기카드, IC카드)들과 비교해보면, RFID가 비접촉식으로 한 번에 다중 태그를 인식하며 다른 매체에 비해 인식속도가 빠른 특징을 가지고 있는 것을 알 수 있다. 또한 다른 매체에 비해 인식거리가 상대적으로 길고 금속을 제외한 장애물의 투과도 가능하다. 그리고 재활

용성도 뛰어나 반영구적으로 사용할 수 있으며 복제가 불가능하여 높은 보안 능력을 가지고 있다[6].

<표 2> 매체별 인식기술 비교

	바코드	자기 카드	IC 카드	RFID
인식 방법	비접촉식	접촉식		비접촉식
인식 거리	~50cm	리더기에 삽입		~27m
인식 속도	4초	4초	1초	0.01~0.1초
인식률	95%이하	99.9% 이상		
투과력	불가능			가능(금속제외)
사용 기간	—	1만번이내(4년)	1만번(5년)	10만번(60년)
데이터 저장	1~100byte	1~100byte	16~64Kbyte	64Kbyte이하
Data Write	불가	가능		
카드 손상률	매우 잦음	잦음	잦음	거의 없음
태그 비용	가장 저렴	저렴	높음(\$10이상)	보통(\$0.5~\$1)
보안 능력	거의 없음	거의 없음	복제 불가	복제불가
재활용	불가능		가능	

RFID는 기존의 인식 매체들과 달리 여러 장점을 가지고 있다. 이와 더불어 최근 새로운 기술로 각광 받고 있는 컬러코드(Colorcode)는 <그림 2>와 같이 최소 3×3mm 크기의 매트릭스 형태로 4가지 색상(적색, 흑색, 녹색, 청색)을 조합하여 나올 수 있는 경우의 수를 이용하여 각종 디지털 정보를 기록·저장하는 새로운 개념의 데이터 표현 기술이다. 컬러코드는 자체에 정보를 입력하는 것이 아니라 온라인 정보(콘텐츠)의 URL을 데이터베이스에 저장하고 컬러코드 인식을 통해 데이터베이스에 저장된 URL만 불러오는 방식이다[7].



<그림 2> 컬러코드

이러한 컬러코드와 RFID는 몇 가지 차이점과 공통점을 가지고 있으며 <표 3>과 같다. 차이점을 살펴보면 컬러코드는 10만 화소급 이상의 카메라가 부착된 단말기로 인식이 가능하지만 RFID는 고가의 RFID 리더가 필요하다. 그리고 컬러코드는 한 번에 단일 코드를 인식할 수 있는 반면에 RFID는 한 번에 다중 태그를 인식할 수 있으며 컬러코드에 비해 보안능력이 뛰어나다. 공통점은 컬러코드와 RFID 모두 통신환경을 필요로 한다는 것이다. [8].

<표 3> 컬러코드와 RFID의 비교

구분	컬러코드	RFID
인식기	단말기에 부착된 카메라	별도의 RFID 리더기 부착
패턴 수	5×5의 경우 170억개	10 ³⁸
통신환경	필요	필요
가격	저가	고가
유통방식	인쇄매체 등으로 배포가능	물리적인 미디어이므로 태그 자체에 한정됨
보안	낮다	높다

무엇보다 RFID와 컬러코드의 가장 큰 차이점으로 인식방법을 말할 수 있다. RFID는 RFID 태그가 어디에 붙여져 있는지 모르더라도 인식 가능한 거리에 RFID 리더가 위치하면 자동으로 태그를 인식하여 태그에 부합하는 서비스를 받을 수 있

다. 하지만 컬러코드는 컬러코드가 부착된 곳을 찾아 단말기(PDA, 핸드폰)의 카메라를 이용하여 인식시켜야 그에 맞는 서비스를 제공 받을 수 있다.

컬러코드를 비롯한 다른 인식 매체와 달리 이러한 RFID 특성에 기반하여 본 논문에서는 기존 인식 매체가 제공하기 어려운 학습 관련 응용서비스를 제안하고자 한다.

2. 상황학습 이론

이전의 전통적인 학습방법은 모든 학생들이 똑같은 내용을 배우고 똑같은 방식으로 평가를 받았다. 단순 암기식 평가에만 길들여져 있어서 정작 실제 생활에서 배운 것을 적용하고 그것을 활용할 수 있는 능력은 기를 수가 없었다. 이러한 문제를 해결하기 위한 대안적 방법의 하나가 상황학습 이론(Situated Learning theory)이다[9].

상황학습 이론은 구성주의에 입각한 것으로 지식이나 기능은 유의미한 맥락 안에서 제공될 때 효과적으로 학습될 수 있다고 주장한다. 예를 들면, 미분이나 적분은 과학자나 수학자들이 기본적으로 사용하는 중요한 원리이지만 학교 교육에서는 그것이 사용되는 맥락에 대한 정보 없이 그 자체만 가르친다. 이렇게 맥락과 독립된 지식은 그 자체의 의미를 잃어버리기 때문에 이해하기 어렵고 또한 배우고 난 후에도 언제, 어떻게 적용하는지 알기 어렵다.

이렇듯 학교에서는 지식이 실제로 사용되는 맥락과는 분리되어 가르쳐지지만 일상생활에서는 지식이 맥락과 연결되어 있고 단순한 암기보다는 문제 해결을 위해 지식을 이용한다. 이러한 학교와 일상생활에서의 사교의 차이는 학교에서 가르치는 지식을 일상생활에 적용하는데 한계를 가져오게 된다는 것이다. 맥락은 이러한 학교와 일상생활의 격차를 메워주는 역할을 할 수 있으며, 그런 의미에서 맥락의 중요성이 강조된다[10].

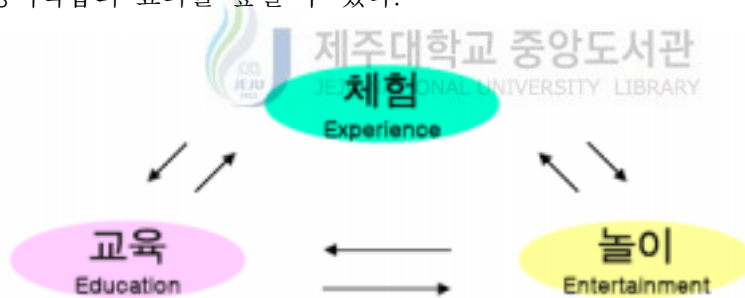
또한 상황학습 이론에서 지식은 상황적이고 그 지식이 사용될 행동과 문화 안에서 생성된다고 한다. RFID는 학습자에게 상황학습을 시킬 수 있는 적절한 기술의

하나이다. RFID 태그를 학습하고자 하는 모든 것(곳)에 부착하고 무선 주파수를 이용하여 비 접촉식으로 리더에서 태그를 인식하면 학습자 처한 매 상황에 맞는 학습 콘텐츠를 제공할 수 있기 때문이다.

3. 영어마을

영어마을은 텍스트위주의 학습을 탈피한 능동적이고 체험위주의 학습형태를 갖춘 체험공간이다. 영어마을에서는 영어권 국가의 모습을 그대로 재현해 놓고 생활영어 뿐만 아니라 세계문화에 대해 체험학습 할 수 있다.

해외여행을 가려고 할 때 제일 먼저 거쳐야 하는 출입국 심사과정에서부터 여행 시 접하는 상황들 예를 들어, 호텔, 은행, 병원, 경찰서 등의 실제와 유사하게 구성된 체험공간에서 외국인과 한국인 교사가 2인1조로 구성되어 체험활동을 지도하고 학생들은 원어민 강사로부터 영어권 문화와 생활을 배우며 외국인과의 대화에서 자신감을 갖게 된다. 강의실 형태의 주입식 교육이 아닌 실제 체험공간에서의 참여형 학습으로 영어학습의 효과를 높일 수 있다.



<그림 3> 영어마을의 3E 학습

영어마을 학습방법의 특징으로 3E 학습을 꼽을 수 있다. 즉 ‘영어로 놀기’ 이다. 3E란, 교육(Education), 체험(Experience), 놀이(Entertainment)를 뜻한다[11].

교육(Education)은 주말프로그램, 방학프로그램, 5박 6일 프로그램 등의 체류형 영

어 학습프로그램을 개발하여 한국인에게 맞는 생활 영어를 학습한다는 것이다.

체험(Experience)은 직접 체험하며 영어를 실생활에 적용하는 것이며, 놀이(Entertainment)는 즐겁게 놀면서 영어 능력을 높이는 것으로 힙합과 재즈, 마술, 요리, 문화체험 등을 통해 영어를 즐겁게 학습하는 것을 의미한다.



Ⅲ. 영어 상황학습을 위한 RFID 미들웨어

1. 미들웨어 설계



<그림 4> RFID 미들웨어 설계

지금까지 RFID와 관련된 연구는 주로 사물에 부착하기 위한 태그와 이를 무선을 통해 자동으로 인식하기 위한 칩, 리더 등 하드웨어 중심으로 발전되어 왔다. 그러나 최근 들어 기존 시스템과 RFID 기술 간의 통합을 지원하기 위해 태그 데이터의 수집, 정제 및 관리 등을 수행하는 미들웨어에 대한 관심이 증가하고 있다. 즉, 다양한 RFID 디바이스와 응용시스템간의 유연한 연결을 지원하고, 대량의 태그 데이터가 리더로부터 실시간 인식되는 환경에서 데이터를 수집하고 처리하는 부하를 최소

화하며, 빠르고 효율적으로 태그 이벤트를 응용시스템에 전달하는 기능을 수행하는 새로운 형태의 RFID 미들웨어가 요구되고 있다[12].

이러한 요구와 더불어 본 연구에서는 기존의 대용량 RFID 미들웨어에 비해 규모가 작은 PDA기반의 모바일 RFID 미들웨어를 상단부분과 하단부분으로 나누어 설계, 구현하였다.

구현된 모바일 RFID 미들웨어의 상단 부분은 리더 제어, 태그 제어, 큐 제어, 필터 제어의 필수 API를 중심으로 태그 데이터를 수집하고 정제한다. 이를 의미 있는 정보로 요약하여 영어 상황학습 시스템에 전달하는 역할을 수행하며 미들웨어 하단 부분과의 연결고리 역할을 한다. 미들웨어의 하단 부분은 RFID 리더의 디바이스 드라이버 역할을 하며 리더와의 통신프로토콜을 정의하고 리더와 미들웨어 상단부분과 매개한다. 미들웨어를 상단 부분과 하단부분으로 2계층화한 것은 다수의 이기종의 RFID리더를 지원해주기 위한 것이다. U-learning API는 영어 상황학습 시스템에게 RFID 태그로 인식된 학습자의 위치와 상황에 따른 학습 콘텐츠를 제공해주며 학습자 자신만의 단어장을 만드는 기능을 제공한다.

1) 미들웨어 상단 부분 API



(1) 리더 제어

리더 제어는 RFID 리더를 제어하고 상태감시 기능을 제공하는 명령어이다. 즉, <표 4>와 같이 RFID 리더의 전원을 관리하고 지역에 따른 가용 주파수를 설정하며 태그를 인식하는 과정에서의 상태들을 알려주는 명령어이다.

<표 4> 리더 제어

명령어	설명
RM_Power()	RFID 리더의 전원을 ON/OFF 한다.

RM_Open()	RFID 리더가 태그에 대한 제어 명령어를 받아 수행할 수 있는 상태가 되도록 한다.
RM_Close()	RFID 리더와 통신하는 논리적 채널을 닫는다.
RM_Reset()	RFID 리더 내의 모든 레지스터 값을 초기화하고, 최초 Open된 상태가 되게 한다.
RM_SetArea()	RFID 리더가 지역에 따른 전파 규격에 맞추어 동작하도록 가용 주파수를 설정한다.
RM_GetArea()	RFID 리더에 설정된 지역 값을 구한다.
RM_GetRFStrength()	RFID 리더에 현재 설정된 최대 송출 전력 대비 백분율의 값을 얻는다.
RM_IsOpen()	RFID 리더의 오픈 상태 여부를 얻는다.
RM_IsBusy()	RFID 리더가 현재 태그 제어 명령을 수행하고 있는지 여부를 얻는다.
RM_SetReadCycle()	RFID 리더에 설정된 태그 읽기를 수행하는 횟수를 얻는다.
RM_SetReadDelayTime()	한번 태그 읽기를 수행하고 다음 태그를 읽기를 수행하기까지의 대기 시간을 설정한다.
RM_GetReadDelayTime()	한번 태그 읽기를 수행하고 다음 태그를 읽기를 수행하기까지의 대기 시간을 얻어온다.
RM_ReaderStatus()	RFID 리더의 상태와 기본 정보를 반환한다.

(2) 태그 제어

태그 제어는 RFID 태그를 제어하기 위한 기능을 제공하는 명령어이다. 즉, 태그 ID를 읽고 인식된 결과를 큐에 저장하며, RFID 리더의 동작 결과에 따라 이벤트를 발생시킨다. 또한 현재 수행중인 태그 데이터 읽기 동작을 제어하거나 태그를 더 이상 사용하지 않게 설정할 수 있는 명령어이다.

<표 5> 태그 제어 명령어

명령어	설명
RM_ReadTID()	RFID 리더로 태그 ID를 읽게 하여 큐에 저장하고, 리더의 동작 결과에 따라 정의된 이벤트 발생시킨다.
RM_StopRead()	RFID 리더로 현재 수행중인 태그 데이터 읽기 동작을 중지하게 한다.
RM_ReadUserData()	RFID 리더로 태그의 사용자 메모리에 있는 데이터를 읽게 하여 그 결과를 전달받아 큐에 저장하고 리더의 동작결과에 따른 이벤트를 발생시킨다.
RM_KillRFTag()	RFID 리더로 태그가 더 이상 사용되지 않도록 태그를 설정하게 하고 그 수행 결과를 전달받아 그에 다른 이벤트 발생시킨다.
RM_LockType()	RFID 리더로 태그 내의 특정 메모리 블록에 데이터를 쓰거나 읽기를 수행할 수 없게 하고, 그 수행 결과를 전달받아 그에 따른 이벤트를 발생시킨다.

(3) 큐 관리



큐 관리는 RFID 리더가 읽어온 데이터를 효율적으로 저장하고 제공하는 기능을 하는 명령어이다. 큐를 생성하고 삭제하는 기능과 큐에 저장된 데이터를 저장하고 삭제하며, 큐에 유효하지 않은 데이터를 삭제하는 등의 명령어로 큐를 효과적으로 관리할 수 있게 해준다.

<표 6> 큐 관리 명령어

명령어	설명
RM_CreateQueue()	RFID 리더에서 읽혀진 태그의 정보를 저장하는 큐를 생성한다.

RM_DestroyQueue()	RFID 리더에서 읽혀진 태그의 정보를 저장하는 큐를 삭제한다.
RM_ReadQueue()	큐에 저장된 데이터를 읽어온다.
RM_WriteQueue()	큐에 데이터를 저장한다.
RM_DeleteQueue()	큐의 데이터 한 개를 삭제한다.
RM_ClearQueue()	큐의 모든 데이터를 삭제한다.
RM_GetNumQueue()	큐에 현재 저장되어 있는 데이터의 개수를 얻어온다.
RM_GetMaxQueue()	큐에 현재 저장할 수 있는 최대 데이터의 개수를 얻어온다.
RM_SortQueue()	큐에 저장된 데이터를 정렬한다.
RM_ValidateQueue()	큐에 저장된 데이터 중에 유효하지 않은 데이터 삭제한다.

(5) 필터 관리

필터 관리는 RFID 리더가 읽어온 데이터 중 RFID 응용 프로그램에서 불필요한 데이터를 제거하거나 필요한 데이터만 걸러내는 기능을 하는 명령어이다.

<표 7> 필터 관리

명령어	설명
RM_AddFilter()	RFID 리더는 태그의 정보를 읽고 큐에 저장하는 과정에서 필터링하여 저장하도록 필터를 추가한다.
RM_DeleteFilter()	필터를 삭제한다.
RM_EnableFilter()	필터를 동작하도록 설정한다.
RM_DisableFilter()	필터가 동작하지 못하도록 설정한다.

2) 미들웨어 하단 부분 API

(1) 시리얼 포트 제어

본 미들웨어에서는 RFID 리더와 PDA가 통신하기 위해 시리얼 통신을 이용한다. 사용되는 명령어는 <표 8>과 같으며 시리얼 통신이 동작을 살펴보면 첫째, RML_OpenComport()를 이용하여 포트를 오픈한다. 대부분의 PDA 포트는 COM1을 사용한다. 둘째, 포트오픈 후 포트 설정(RML_ConfigComport())을 해주는데 포트의 전송 속도, 데이터 비트수, 패리티 비트, 스톱비트 수를 설정한다. 설정을 마치면 PDA와 RFID 리더가 통신할 준비를 갖추게 되며 셋째, RFID 리더에서 읽어 들인 태그 ID값을 RML_ReadTID()를 통해 PDA에서 읽어온다. 넷째, 통신프로토콜에 의해 통신을 하고 통신이 종료되면 RML_CloseComport()로 포트를 닫아주면 통신이 종료된다.

<표 8> 시리얼 포트 제어

명령어	설명
RML_OpenComport()	시리얼 포트를 열어준다.
RML_ConfigComport()	포트의 전송 속도, 데이터 비트수, 패리티비트, 스톱 비트수, Flow Control을 설정한다.
RML_ReadTID()	태그 ID값을 읽어온다.
RML_CloseComport()	시리얼 포트를 닫는다.

(2) 통신 프로토콜 관리

RFID 미들웨어의 하단 부분에서 리더와 호스트(PDA)간의 통신 프로토콜을 관리한다. 통신 프로토콜에서 사용되는 프레임은 Tag-Frame과 ACK, NAK, Firmware DATA으로 구성되어 있으며 각각의 설명은 아래와 같다.

(1) Tag-Frame

SF (1byte)	Command (1byte)	Length (1byte)	Value (nbytes)	EF (1byte)	FCS (1byte)
----------------------	---------------------------	--------------------------	--------------------------	----------------------	-----------------------

Tag-Frame에서의 데이터는 기본적으로 16진수를 사용하며 태그 프레임은 SF, Command, Length, Value, EF, FCS로 구성된다. Tag-Frame의 구성과 각각에 대한 설명은 다음과 같다.

- SF(Start of Frame) : Frame의 전송시작을 나타내며 0x01의 16진수 값을 갖는다.
- Command : Host to Reader시에는 수행할 command이며
Reader to Host시에는 수행한 command를 echo하여 보낸다.
- Length : 데이터의 길이를 나타내며 Value 필드의 byte수를 나타낸다.
- Value : Host to Reader 시는 command 별로 전송되어야 할 data를 나타내며,
Reader to Host시에는 응답 data를 표시한다. 만약 command에 따른
부수적 data가 없을 경우 Length는 0x00이 되고, value field는 생략된다.
- EF(End of Frame) : Frame 전송의 끝을 나타낸다.
- FCS(Frame Check Sequence) : SF와 EF를 제외한 나머지 모든 필드들의
Exclusive-OR 값을 나타낸다.

<표 8>은 호스트와 리더간의 주요 명령어를 나타낸 것으로 Tag-Frame의 Command 부분에 속한다.

<표 9> 호스트와 리더간의 명령어

인코딩 값	명령어	설명
0x01	Read	안테나 영역 내 검색되는 Tag ID를 읽는다.
0x02	Set Config	레지스터의 특정 값을 설정한다.

0x03	Get Config	특정 레지스터의 값을 읽는다.
0x04	Save	EEPROM에 레지스터 값을 저장한다.
0x05	Get F/W Version Info	레지스트리 8byte F/W Version 정보를 읽어온다.
0x06	F/W Update	펌웨어를 업데이트 한다.
0x07	CHK	Live Check Signal임을 알려준다.

(2) ACK Frame

RFID 리더에서 명령수행 후 특별한 응답 메시지가 없을 경우 ACK을 전송하여 명령수행 사실을 알린다.

ACK (1byte)	- ACK(Acknowledgement) : ACK frame임을 나타냄.(0x03)
-----------------------	---

(3) NAK Frame



잘못된 명령의 수행 혹은 명령 수행 중 오류 발생 시 전송하며 <표 9>는 NAK 프레임이 전송되었을 때의 에러 코드를 나타낸 것이다.

NAK (1byte)	EC (1byte)
-----------------------	----------------------

- NAK(Negative Acknowledgement) : NAK frame임을 나타냄(0x04)
- EC : Error code

<표 10> 에러 코드

에러코드	설명
0x01	Unknown Command
0x02	Invalid register address
0x03	Invalid register value
0x04	Writing fail
0x05	Frame Checksum mismatch
0x06	Invalid command data length

(4) Firmware Data

펌웨어는 RFID 리더내의 프로그램으로 업데이트를 할 경우 호스트로부터 리더로 전송한다.

F/W DATA (nbytes)	Checksum (1byte)
----------------------	---------------------

- F/W DATA : UPTATE할 실제 F/W 데이터
- Checksum : F/W DATA의 256 바이트 각각을 모두 더한 결과의 최하위 1바이트 값.

3) U-Learning API

RFID 응용 프로그램으로 영어 상황학습 시스템에서 학습자의 위치와 상황을 인지하여 그에 부합하는 알맞은 문장과 단어를 <표 11>의 API를 이용하여 제시해주며 학습자가 학습하고자 하는 단어를 나만의 단어장을 통해 저장, 삭제, 검색할 수 있는 명령어이다.

<표 11> U-Learning API

명령어	설명
RA_ShowSentence()	학습자의 위치와 상황에 맞는 문장을 보여준다.
RA_ShowVocabulary()	학습자의 위치와 상황에 맞는 단어를 보여준다.
RA_MakeVocabulary()	학습자 자신만의 영어 단어장을 생성한다.
RA_DeleteVocabulary()	학습자 자신만의 영어 단어장을 삭제한다.
RA_ShowWords()	학습자 자신만의 영어 단어장을 보여준다.
RA_AddVocabulary()	학습자가 추가하고자 하는 단어를 자신만의 단어장에 추가한다.
RA_DeleteVocabulary()	학습자가 삭제하고자 하는 단어를 자신만의 단어장에서 삭제한다.
RA_SearchVocabulary()	학습자가 찾고자 하는 단어를 검색한다.

2. 미들웨어 구현

1) 시스템 구현 환경

본 연구에서 호스트 역할을 하는 PDA의 하드웨어 구성요소는 <표 11>과 같다. RFID 미들웨어는 Embedded Visual C++ 4.0으로 개발하였으며 테스트베드로 IPAQ

4700 모델을 사용하여 테스트 하였다.

<표 12> PDA 시스템 구성

선택 사양	구성내용
운영체제	한글MS 윈도우즈 모바일 2003 Second Edition 포켓PC 프로페셔널 버전
무선통합	WLAN 802.11b,블루투스,IrDA(FIR)
프로세서	Intel PXA270 624MHz프로세서
디스플레이	4인치 반투과형 VGA TFT컬러, 6.5K컬러
메모리	128MB 플래시롬 64램(iPAQ File Store 80MB사용가능)

2) RFID 미들웨어 동작과정

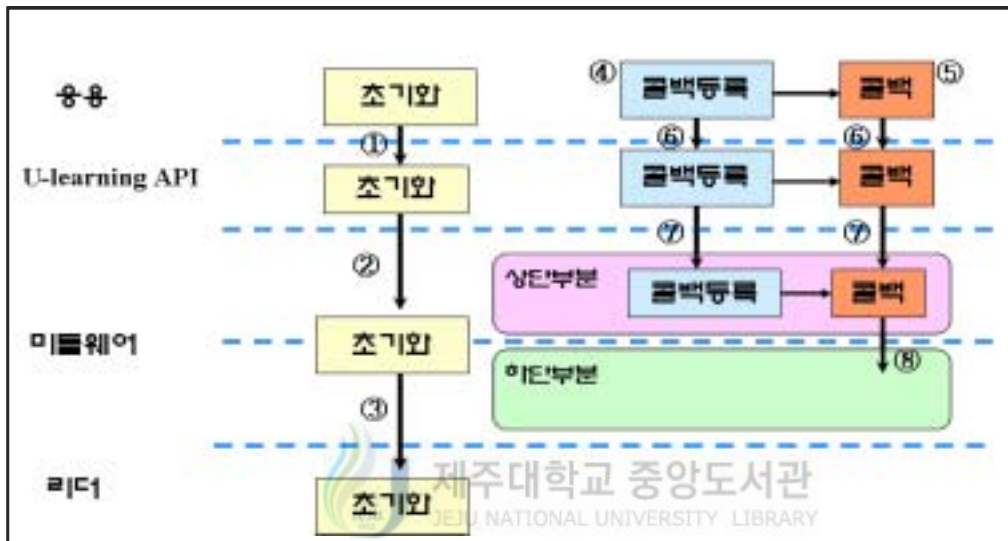


<그림 5> RFID 미들웨어 동작과정

<그림 5>는 미들웨어의 상단부분과 미들웨어의 하단부분을 연결하고 영어 상황학습 시스템에서부터 리더에 이르기까지의 태그가 인식되는 과정을 나타낸다.

3) RFID 미들웨어 명령어 흐름도

- 초기화/콜백 등록시



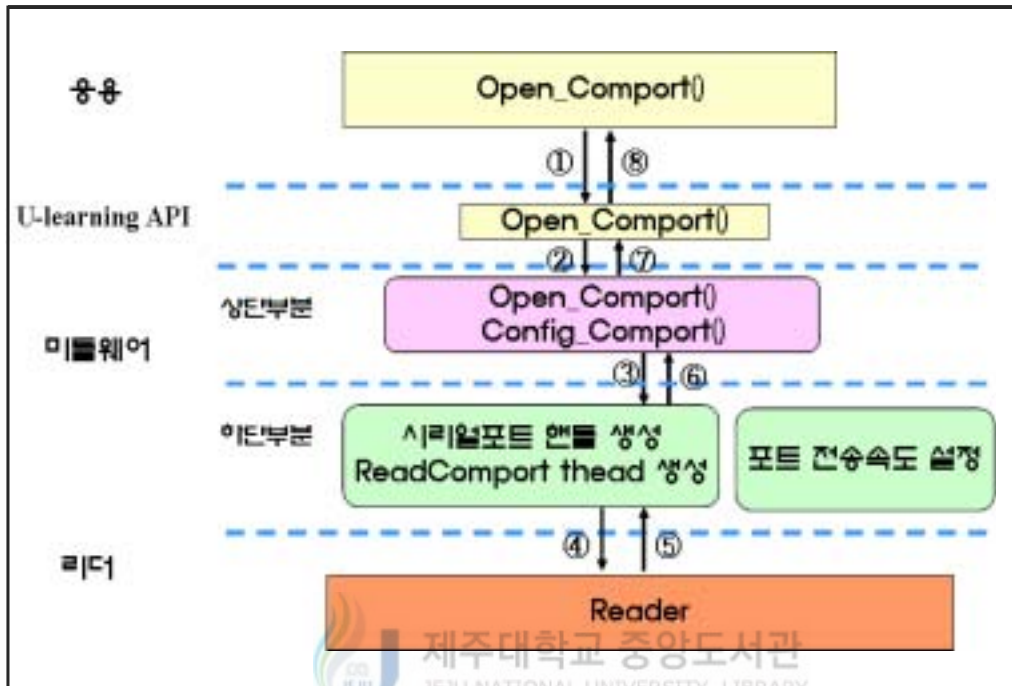
<그림 6> 초기화/콜백 등록 시

응용에서 초기화 화면(초기화 호출시) U-learning API는 미들웨어 상단부분에 명령을 전달하고 미들웨어의 상단부분은 미들웨어 하단부분에 콜백 등록을 한 후 응용에서 콜백 등록을 하여 응용에서 리더까지 연결되게 되는 과정이다.

- RM_Open()

응용에서 RM_Open() 명령을 내리면 U-learning API는 미들웨어 상단부분에 명령

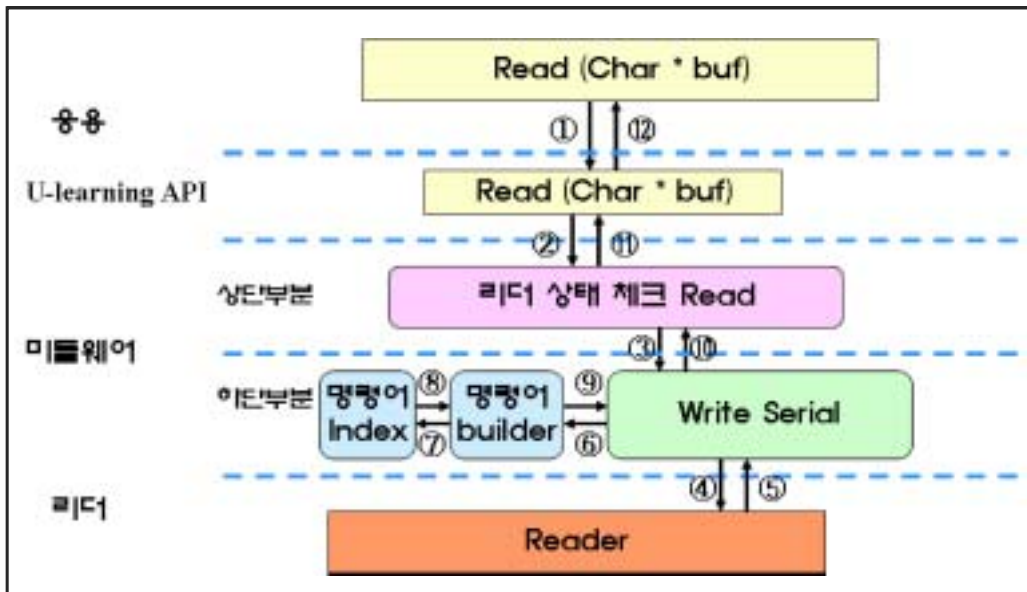
어를 전달하고 상단부분에서는 콤포트를 오픈하고(Open_Comport()) 콤포트를 설정(Config_Comport())한다. 미들웨어 하단부분은 시리얼포트 핸들 생성과 ReadComport 스레드를 생성하고 포트 전송속도를 설정하여 리더와 연결한다.



<그림 7> RM_Open()

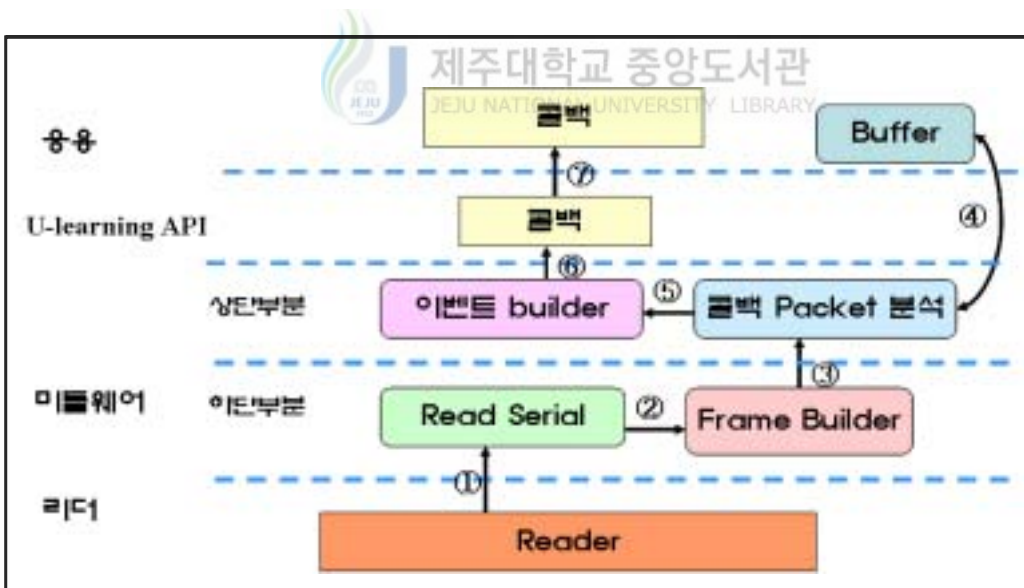
- Read()

응용에서 Read() 명령을 내리면 U-learning API에서는 Read() 명령어를 전달하고 미들웨어 상단부분에서는 리더 상태체크를 한다. 미들웨어 하단부분에서는 명령어 Index와 Frame builder를 이용하여 시리얼 포트에 기록한다. 리더는 미들웨어와의 통신프로토콜 중에 ACK를 보내어 응용까지 그 결과가 전송되게 한다.



<그림 8> Read ()

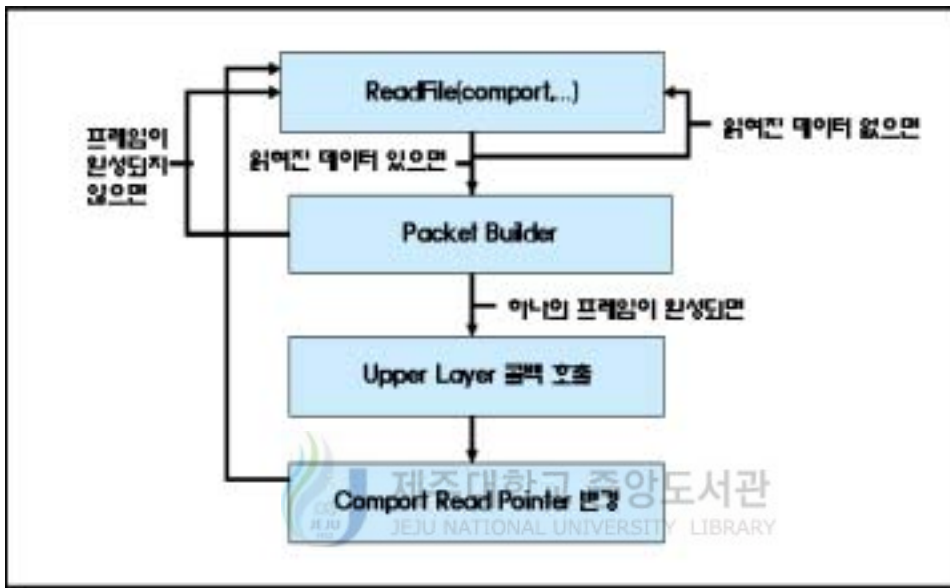
- ReadData()



<그림 9> ReadData ()

리더는 Read() 명령을 수행한 결과를 미들웨어의 하단부분으로 보내면 Frame Builder에서 하나의 패킷으로 만들고 미들웨어 상단부분으로 전송하면 상단부분에서는 패킷분석을 하여 Read된 data임을 알고 Buffer에 저장한 후 이벤트 Builder를 이용하여 U-learning API를 거쳐 응용에게 콜백으로 전송한다.

- ReadComport()

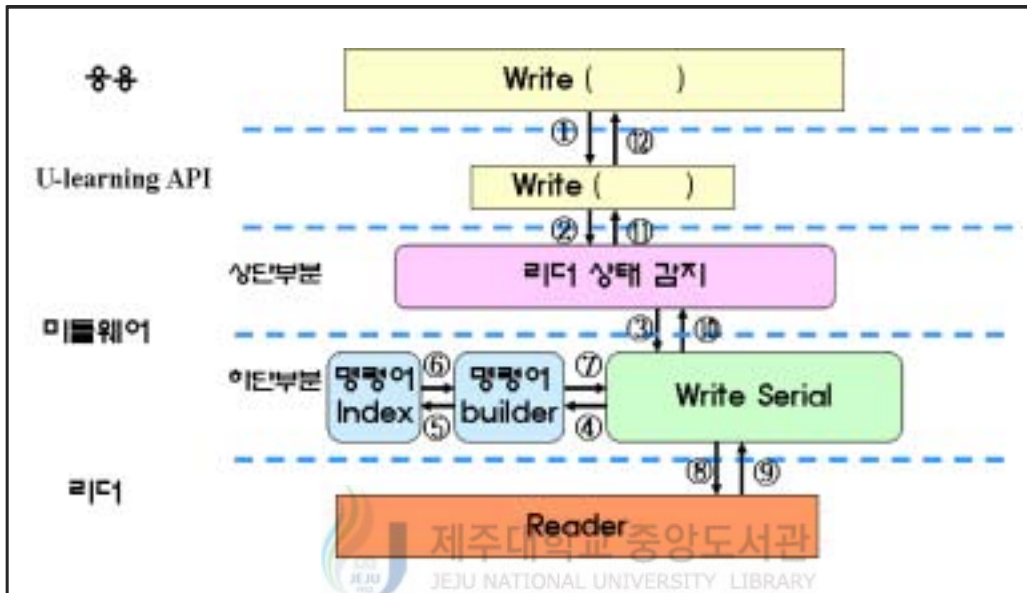


<그림 10> ReadComport()

ReadFile()후 읽혀진 데이터가 없으면 다시 읽게 된다. 읽혀진 파일이 있으면 Packet Builder를 이용하여 하나의 프레임이 완성하여 미들웨어 상단부분에서 콜백을 호출하고 Comport Read Pointer를 변경한다. 하나의 프레임이 완성되지 않은 경우에는 다시 ReadFile()을 하게 된다.

- WriteData()

응용에서 Write 명령을 내리면 미들웨어 상단부분에서는 리더상태체크를 하고 하단부분에서는 명령어 Index와 FrameBuilder를 이용하여 시리얼 포트에 기록한다. 리더는 받은 데이터를 태그에 기록하고 결과를 미들웨어로 전송하게 된다.



<그림 11> WriteData()

IV. 영어 상황학습을 위한 응용 시스템

구현된 RFID 미들웨어에 기반하여 영어 상황학습 시스템을 설계 및 구현하였다. 구현된 시스템은 영어 교육을 목적으로 하는 외국어 고등학교, 영어마을 등 다양한 곳에 적용시킬 수 있으나 본 연구에서는 영어마을을 선택하여 영어마을에서의 호텔, 은행, 병원, 경찰서 등의 실제와 유사하게 구성된 체험공간에서 영어 상황학습 시스템 시나리오를 구상하였다.

1. 응용 시나리오

시나리오를 구상하는데 있어 전제 조건은 다음과 같다.

첫째, 병원, 은행, 호텔 등의 실제 체험 공간에 RFID 태그가 붙여져 있다.

둘째, 학습자는 RFID 리더가 부착된 PDA를 소유하고 있다.

셋째, 학습자는 사전에 PDA 사용법에 대한 교육을 받았다.

넷째, 학습자는 중학교 1~2학년을 대상으로 하며 제시되는 문장과 단어의 수준이 학습자의 수준에 맞춰져 제시된다.

본 논문에서는 영어마을의 호텔, 은행 등의 다양한 상황 체험 공간 중 병원을 선택하여 병원에서 있을 수 있는 상황으로 크게 접수·진료, 약국에서 조제 상황을 설정하여 시나리오를 만들었다.



<그림 12> 병원_접수



<그림 13> 병원_진료



<그림 14> 약국_조제

학습자가 병원에 들어서면 학습자의 PDA에서 병원 건물에 부착된 RFID 태그를 자동으로 인식하게 된다. 태그가 인식되면 학습자의 PDA는 학습자가 어느 장소인지 어떤 상황인지를 파악할 수 있다. 따라서 학습자의 위치와 상황에 따라 영어 문장과 단어를 제시하여 학습자에게 학습정보를 제공해준다. 제시되는 문장은 <표 13>과 같이 다이얼로그 형식으로 제시되며 학습자의 위치와 상황에 적합한 문장으로 제시된다[13]. 제시되는 단어는 <표 14>와 같이 제시된 문장에 사용된 단어 또는 학습자의 현재 처한 상황에서 자주 쓰이는 단어를 제시한다[14]. 문장과 단어는 음성으로도 지원하여 원어민의 발음을 들을 수 있다. 또한 시스템에서 제시된 단어 중 학습자가 모르는 단어가 제시 되었을 경우 모르는 단어를 '나만의 단어장'에 저장하고 학습 도중 모르는 단어가 생겼을 시에는 '단어검색'을 통해 검색된 결과를 자신만의 단어장에 저장할 수 있다. 그러므로 학습자만의 맞춤형 영어 단어장을 만들 수 있다.

<표 13> 제시되는 영어문장 예

Doctor : How do you feel?
 Patient : I feel < sick >. / I feel < terrible >. /< I don't feel well >.
 Doctor : What are your symptoms?
 Patient : I have a runny nose and a fever.
 Doctor : You have a cold. Drink lots of water, and get some rest. Ok?
 Now, please go to the pharmacist for your medicine.
 Here is the prescription.
 Patient : Thank you very much.
 Doctor : You're welcome. Take care.

<표 14> 제시되는 영어단어 예

1. runny nose 콧물	2. headache 두통
3. scratch 긁다	4. cold 감기
5. sick 아픈	6. cough 기침
7. skin 피부	8. doctor 의사
9. sneeze 재채기하다	10. emergency room 응급실
11. sore throat 목이 아픈	12. fever 열
13. splint 부목	14. stomach ache 복통

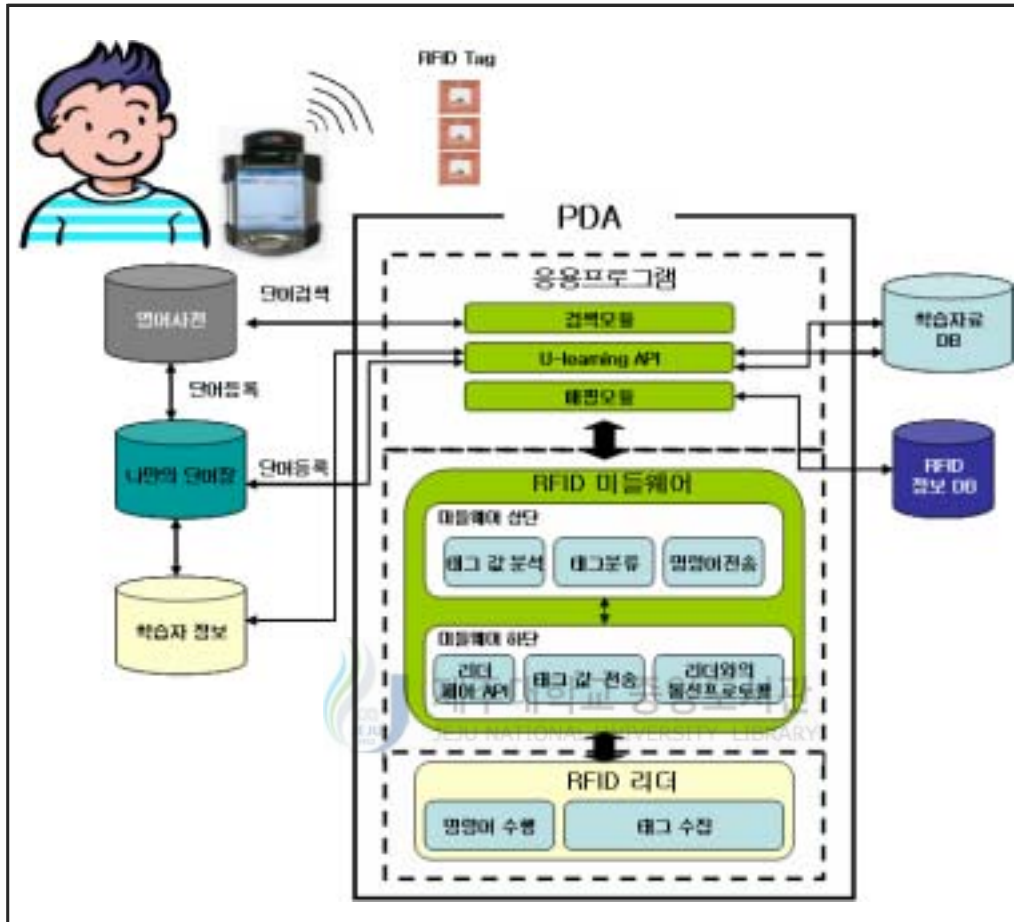
영어 상황학습 시스템에서 제시된 문장을 학습 후 학습자 A와 학습자 B는 의사와 환자의 역할을 선택하여 역할극을 통해서도 학습이 이루어질 수 있다. 환자 역할을 하는 학습자는 제시된 문장과 단어를 이용하여 <그림 15>와 같이 액션카드에 나타난 환자의 증상을 상대방인 의사역할을 하는 학습자에게 설명한다. 의사 역할을 하는 학습자는 환자의 건강상태를 살피고 환자 역할을 하는 학습자가 설명한 증상에 대해 처방한다. 이러한 방법을 통해 제시된 문장과 단어를 효과적으로 활용하면서 영어 회화능력, 어휘력 등을 높일 수 있으며 상황학습의 효과를 높일 수 있다.



<그림 15> 액션카드

영어마을은 입소와 동시에 영어 마을 내에서는 오직 영어로만 대화해야한다. 이 때, 학습자의 성향에 따라 학습자가 외향적 성격이라면 영어마을의 프로그램을 적극적으로 임하지만 학습자가 내성적 성격이라면 영어마을의 프로그램을 외향적 성격의 학습자보다 소극적인 자세로 임하게 될 것이다. 본 시스템을 사용한다면, 내성적인 학생 또는 영어에 대해 자신감이 떨어지는 학생들은 상황에 맞게 주어지는 문장들을 통해 자유롭게 의사소통 할 수 있어 자신감 있게 학습에 임할 수 있을 것이다.

2. 응용 시스템 설계



<그림 16> 영어 상황학습 시스템 구조

<그림 16>은 PDA에서 동작하는 영어 상황학습 시스템 구조를 나타낸 것이다. 위 그림에서의 RFID 미들웨어는 미들웨어의 기능을 간략화한 것이다. 학습이 필요한 곳에 부착된 RFID 태그는 학습자의 PDA에 부착된 RFID 리더를 통해 수집되

고 RFID 미들웨어를 통해 정제 및 관리된다. 매핑모듈과 RFID 정보 데이터베이스를 통해 인식된 RFID태그는 학습자의 위치와 상황 나타낸다.

매핑모듈에서 인식된 태그 값에 맞는 학습자료 즉, 학습자의 위치와 상황에 맞는 문장과 단어를 학습자료 데이터베이스에서 U-learning API를 통해 학습자에게 제공된다. 제시된 단어 중 학습자가 모르는 단어 또는 학습하고자 하는 단어는 나만의 단어장에 저장할 수 있으며 저장된 결과는 나만의 단어장 데이터베이스에 저장되게 된다. 제시되는 문장과 단어는 음성으로 들을 수 있으며 학습자가 학습도중 제시된 단어 이외에 모르는 단어가 생기는 경우에는 검색모듈을 통해 영어사전을 이용하여 검색하게 되고 검색된 결과를 나만의 단어장으로 저장할 수 있게 한다.

1) 데이터베이스 설계

(1) 학습자 정보

간단한 학습자의 정보를 입력받아 저장해두는 데이터베이스이다. 학습자의 이름이 기본 키가 되며 학습자 인증과 자신만의 단어장을 이용할 때 사용된다.

Table Name	Users					설 명	회원정보
NO.	Field Name	Type	LEN	P.K	Null	설 명	
1	Name	varchar	20	●	×	이름	
2	UserID	varchar	50			아이디	
3	UserPW	varchar	50			비밀번호	

(2) 문장정보

U-learning API를 통해 학습자에게 제시되는 영어 문장들을 저장하는 데이터베이스이다. UID는 문장들의 일련번호를 나타내며 기본 키가 된다. RFID 태그는 태그 ID값에 알맞은 영어 문장을 제시해주기 위한 것이며, 영어문장과 해석, 소리파일 경로를 저장해두어 제시된 문장에 대한 해석과 음성을 보고 들을 수 있게 지원한다. 또한, 제시된 문장에 나오는 단어를 누르면 단어에 대한 정보를 볼 수 있게 링크단어를 저장하는 데이터베이스이다.

Table Name	Contents		설 명			문장정보
NO.	Field Name	Type	LEN	P.K	Null	설 명
1	UID	int	4	●	×	
2	RFIDTag	varchar	50			태그ID
3	English	text				영문문장
4	Korean	text				해석
5	Voice	varchar	50			소리파일경로
6	Words	text				링크단어

(3) 단어정보

U-learning API를 통해 제시되는 영어 단어 정보를 저장하는 데이터베이스이다. 단어 정보 데이터베이스는 영어 단어와 단어의 발음기호, 뜻 등의 영어 단어에 대한 설명을 저장해 두며 단어에 대한 음성을 지원하기 위해 소리파일 경로를 저장한다.

Table Name	Words	설 명	단어정보
------------	-------	-----	------

NO.	Field Name	Type	LEN	P.K	Null	설 명
1	UID	int	4	●	×	
2	Word	varchar	50			단어
3	Explain	text				설명
4	Voice	varchar	50			소리파일경로

(4) 나만의 단어장

학습자 자신만의 단어장을 만들어 학습자하고자 하는 단어를 저장해두는 데이터베이스이다. 학습자 아이디를 기준으로 분류하기 위해 학습자 아이디를 저장하며 학습자가 저장한 단어를 WUID(단어장 UID : 단어장 일련번호)로 저장하여 단어 정보 데이터베이스에서 단어에 대한 정보를 얻어온다.

Table Name	MyWords	설 명	나만의 단어장 정보
------------	---------	-----	------------

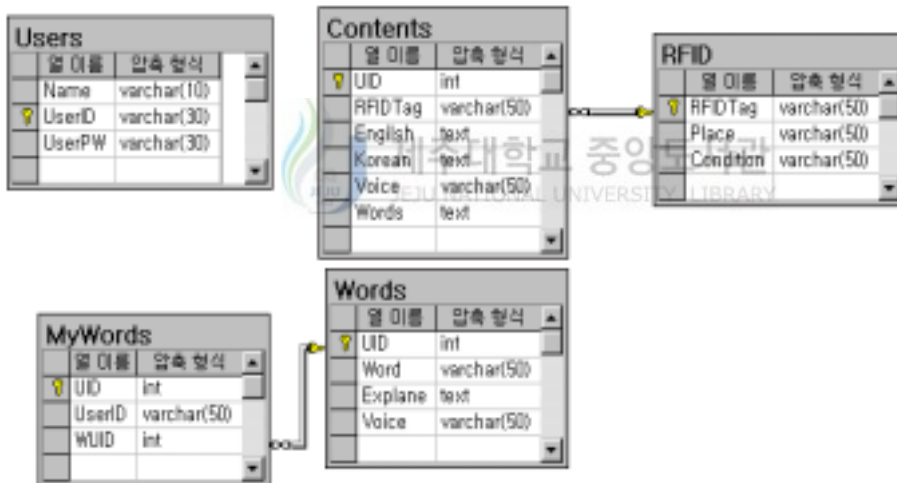
NO.	Field Name	Type	LEN	P.K	Null	설 명
1	UID	int	4	●	×	
2	UserID	varchar	50			아이디
3	WUID	int	4			단어장 UID

(5) RFID 태그 정보

RFID 태그에 대한 정보를 갖고 있는 데이터베이스로서 태그 ID값과 학습자의 위치와 상황에 대한 정보를 저장하고 있다. RFID 태그 ID가 기본 키가 되며 RFID 태그 정보를 통해 학습자에게 위치와 상황에 맞는 문장과 단어를 제시해준다.

Table Name	RFID	설 명	회원정보
------------	------	-----	------

NO.	Field Name	Type	LEN	P.K	Null	설 명
1	RFIDTag	varchar	50	●	×	태그ID
2	Place	varchar	50			위치
3	Condition	varchar	50			상황



<그림 17> 데이터베이스 구조

<그림 17>은 학습자 정보, 문장 정보, 단어 정보, 나만의 단어장, RFID 태그 정보 데이터베이스의 구조를 나타낸 그림이다.

3. 응용 시스템 구현

본 연구에서는 RFID 미들웨어에 기반을 두어 영어 상황학습 시스템을 Embedded Visual C++ 4.0으로 구현하였다. 영어 상황학습 시스템의 기능은 학습자의 상황에 맞는 문장 제시, 제시된 문장에 대한 해석보기, 상황에 맞는 단어 제시, 나만의 단어장, 단어 검색 기능을 가지고 있다.



<그림 18> 로그인 화면

영어 상황학습 시스템은 <그림 18>과 같이 로그인을 통해 학습자의 정보를 얻는다. 로그인 화면에는 학습자의 아이디 찾기와 비밀번호 찾기 기능이 있다.

영어 상황학습 시스템에서의 로그인 후, '검색' 버튼을 이용하여 RFID 리더를 동작시켜 학습자의 현재 위치와 상황의 저장되어 있는 RFID 태그를 인식한다. 태그

가 인식이 되면 학습자의 위치와 상황이 화면에 나타나며 그에 맞는 문장이 제시된다. <그림 19>에서는 학습자가 내과에서 접수·진찰 후 약국에서의 조제중인 상황을 나타내며 그에 맞는 의사와 환자, 환자와 약사와의 대화형식의 문장이 제시되었다.

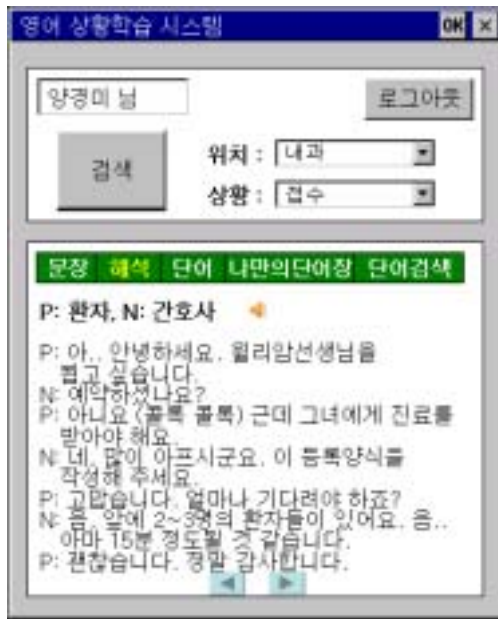


<그림 19> RFID 태그 인식과 문장 제시화면

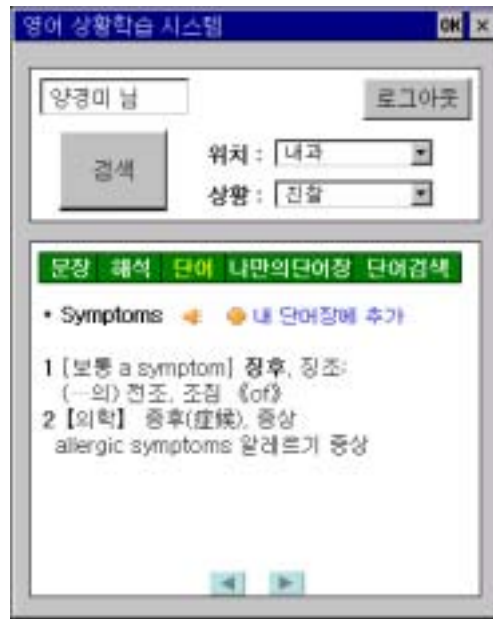


<그림 20>은 제시된 영어 문장에 대한 해석을 볼 수 있는 화면이다. 학습자에게 영어 문장에 대한 이해를 돕고자 하는 것으로 <그림 19>의 내과에서 접수 상황에서 제시된 문장에 대한 해석을 나타낸 것이다. <그림 21>은 <그림 19>에서 학습자가 내과에서 진찰하는 상황에 제시된 문장 중 Symptoms 단어를 클릭하여 단어의 뜻, 품사, 예문을 제시해주는 것을 나타낸 것이다.

<그림 22>는 학습자가 내과에서 진찰 중인 상황에 사용할 수 있는 단어를 제시한 화면이다. 제시된 단어 중 학습자가 나만의 단어장에 추가하고자 하는 단어를 체크하여 단어장에 추가할 수 있다. <그림 23>은 나만의 단어장으로 학습자가 학습하고자 하는 단어들 저장되어 있는 장소이며 단어를 등록, 추가, 삭제할 수 있다.



<그림 20> 해석보기 화면



<그림 21> 문장 속에 단어 내용보기



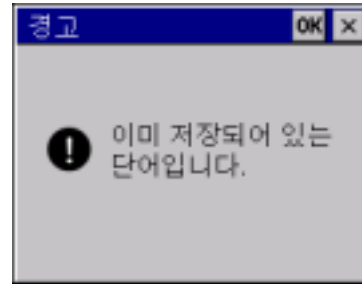
<그림 22> 단어제시 화면



<그림 23> 나만의 단어장



<그림 24> 단어검색 화면



<그림 25> 나만의 단어장에
중복 단어 추가 시 경고

영어마을의 학습 도중 또는 영어 상황학습 시스템을 이용하여 학습하다 학습자가 찾고 싶은 단어가 떠오르거나 모르는 단어를 검색하고 싶을 때 ‘단어검색’을 이용하여 <그림 24>와 같이 단어를 검색할 수 있다. 검색된 결과는 자신의 단어장으로 이동할 수 있으며 나만의 단어장에 저장할 때에 중복된 단어는 다시 저장될 필요가 없기 때문에 저장 시 중복된 단어를 저장하려고 하면 <그림 25>와 같은 메시지가 뜬다. 기존 영어 어휘학습을 하기 위해 학습자는 수첩에 단어와 발음 기호를 일일이 적고 가지고 다니면서 암기했었지만 본 시스템을 이용하여 학습하게 되면 학습자가 처한 상황에 맞는 대화 가능 문장과 단어를 제시해줌으로써 영어 회화능력에 도움을 줄 수 있을 뿐만 아니라 자신만의 단어장을 통해 학습자만의 영어 어휘 맞춤학습을 할 수 있다.

V. 결론 및 추후연구

본 논문에서는 RFID를 이용하여 학습자의 위치와 상황을 인식하여 그에 맞는 서비스 즉, 영어 상황학습 시스템을 개발하였다. 영어 상황학습 시스템은 PDA 기반의 모바일 RFID 미들웨어를 기반으로 동작하며 학습자의 위치와 상황에 따른 문장과 단어를 제시하여 학습할 수 있도록 하는 방안에 대해 연구하였다. 본 시스템의 장점은 다음과 같다.

첫째, RFID태그와 리더의 무선통신 기술을 이용하여 학습자의 위치와 상황에 따른 영어 문장과 단어를 제시해줌으로서 학습자는 영어 상황학습을 할 수 있다.

둘째, RFID 시스템을 이용하기 위해 PDA 기반의 모바일 RFID 미들웨어와 U-learning API를 개발하여 학습자에게 보다 효과적인 학습 콘텐츠를 제공해 준다.

셋째, 학습자들은 '나만의 단어장을'을 통해 영어 어휘 맞춤형 학습을 할 수 있다. 기존 학습자들은 자신만의 단어장을 만들기 위해 수첩에 영어 단어와 발음기호, 뜻을 적고 수첩을 들고 다니며 학습하였다. 그러나 본 시스템에서는 학습자가 처한 상황에 따라 제시된 영어 문장과 단어들 중 학습자가 학습하고자 하는 단어를 쉽게 저장할 수 있게 하였으며 '나만의 단어장'을 빠르고 쉽게 만들 수 있다.

넷째, '단어검색'인 영어사전 기능을 통해 학습자가 학습 도중 모르는 단어 또는 알고자 하는 단어를 검색할 수 있으며 검색된 결과를 '나만의 단어장'으로 저장하여 학습할 수 있다.

끝으로 본 연구와 관련하여 보다 발전된 연구를 위해 향후 연구 과제를 제시한다.

첫째, 본 논문에서는 학습자들에게 영어 상황학습시스템을 적용시켜보지 못하였으므로 학습자들에게 프로그램을 적용하여 본 프로그램에 대한 평가와 학습자들의 학습능력에 어떠한 영향을 미치는지에 대한 분석이 필요하다.

둘째, U-learning 환경에서의 RFID와 컬러코드 등의 새로운 기술을 이용하여 보다 효과적으로 교육에 활용할 수 있는 연구가 필요하다. 예를 들어, 컬러코드 기반의 영어 어휘학습은 교수자가 학습시키고자 하는 사물에 컬러코드를 부착하고 학습자는 부착한 사물을 추리하는 즉, 스무고개 형태의 게임형식으로 영어 학습을 유도하면 학습자에게 흥미를 이끌 수 있을 뿐 아니라 학습자의 능동적 참여를 유도해내어 학습자의 영어 능력을 향상시킬 수 있을 것이다.



참 고 문 헌

- [1] 박영애(1997), 영어마을 운영사례에 관한 연구, TESOL'97, 54권 1호.
- [2] 강명곤, 김명희, 남인숙(2005), "RFID 태그 시스템에 기반한 U-캠퍼스를 위한 통합관리시스템", 한국컴퓨터종합학술대회 논문집, Vol.32. No.1.
- [3] Hiroaki Ogata, Yoneo Yan(2004), "Context-aware support for computer-supported ubiquitous learning", Wireless and Mobile Technologies in Education(WMTE), p.27~34.
- [4] 김형준(2005), "모바일 RFID", TTA Journal, 99호.
- [5] 조대진(2005), RFID 이론과 응용, 흥릉과학출판사.
- [6] 이은곤(2004), "RFID 확산 전망 및 시사점", 정보통신정책, 제16권 13호.
- [7] 칼라짚미디어, www.colorzip.co.kr.
- [8] 김상태(2003), "RFID개요 및 국내외 동향분석", 정보통신연구진흥원
- [9] 김세리(1998), "Good Ideas to Foment Educational Revolution", 교육공학연구회, Vol.38, No.1.
- [10] 박성일, 임철일, 이재경, 최정임(2006), 교육방법의 교육공학적 이해", 교육과학사.
- [11] 영어마을 안산캠프, <http://english-village.gg.go.kr/ansan/>
- [12] 정태수, 김영일, 이용준(2005), RFID 미들웨어 플랫폼 기술, Telecommunications Review, 제15권 2호.
- [13] 서울 영어마을 풍납캠프, <http://pungnap.sev.go.kr/>
- [14] 서울 영어마을 수유캠프, <http://suyu.sev.go.kr/>

<Abstract>

An RFID Middleware and Application System for an English Situated Learning

Yang, Kyoung-Mi

Computer Education Major

Graduate School of Education, Cheju National University

Cheju, Korea

Supervised by Professor Kim, Seong Baeg

Recently, there has been much research to develop and apply RFID technology, which has a key role in the upcoming ubiquitous society. in many fields such as physical distribution, traffic control, medical service, and so on. However, there has been little research on a ubiquitous education or learning including 'U-Campuses' and 'U-Libraries'. Based on the characteristics of RFID, this paper proposes a system for English learning required in globalization age. RFID tags and sensors utilize wireless communications to track the location and status information of the user to deliver English situated learning services.

The current RFID-based system should use quite a different middleware,

compared with a general-purpose middleware on server or desktop. The RFID system is used on a mobile PDA and consists of essential APIs such as reader and tag control, queue and filter management. The RFID middleware we proposed is composed of upper and lower layers. The upper layer focus on situated-learning for English and provide the appropriate APIs, which is required for applications. The lower layer plays a role to make a connection between the high-level layer and a hardware-dependent device driver, enabling communications between them. Dividing the middleware into two parts is to ensure device-independent operation. This is, if an RFID reader in the system is changed, the only lower layer can be replaced with the proper one without modifying the upper layer.

There are 5 major functions in English situated learning system : Sentences, Description, Vocabulary, User-defined Vocabulary Dictionary, and Vocabulary Search. The middleware determines the location of the user and delivers English vocabulary appropriate to the situation as well as Korean explanations corresponding to English expressions. The new words can be added to the user-defined vocabulary dictionary for further study. Users can also search the words from English dictionary and add the results to their personal vocabulary dictionaries.

* A thesis submitted to the Committee of the Graduate School of Education, Cheju National University in partial fulfillment of the requirements for the degree of Master of Education in August, 2006.

<부록 1> RFID 미들웨어 상단 API

- 변수타입

```
typedef unsigned char M_Boolean // unsigned 8bit. TRUE 또는 FALSE
typedef unsigned int M_Uint32 // unsigned 32 bit type
typedef unsigned short M_Uint16 // unsigned 16bit type
typedef unsigned char M_Uint8 // unsigned 8bit type
typedef signed int M_Int32 // signed 32bit type
typedef signed short M_Int16 // signed 16bit type
typedef signed char M_Int8 // signed 8bit type
typedef char M_Char // char type
typedef unsigned char M_Byte // unsigned 8bit type
```

```
#define NULL 0 //NULL 정의
```

```
#define TRUE 1 //TRUE정의
```

```
#define FALSE 0 //FALSE정의
```

```
#define inline __inline //inline정의
```

- 이벤트

⇒ 이벤트 타입

```
enum _RM_Event {
    .
    .
    RM_RFID_TAG_CONTROL_EVENT // RFID 리더의 태그 제어에
        대한 동작 결과를 통지해 주는 이벤트
};
```

⇒ 이벤트 종류

```
Enum RM_CodeControl_Event{
    RM_SUB_RFID_RIDSUCCESS // UII, TID 읽기 명령을 성공적으로 완료
                           했음을 알림
    RM_SUB_RFID_RIDFAIL // 읽기 명령 수행을 실패했음을 알림
    RM_SUB_RFID_NOTAG // 읽은 태그가 없을 때 알림
    RM_SUB_RFID_KILLSUCCESS // Kill 명령을 성공적으로 완료했음을 알림
    RM_SUB_RFID_KILLFAIL // Kill 명령 수행을 실패했음을 알림
    RM_SUB_RFID_NOTAGID // Kill 명령을 수행할 태그가 없음을 알림
    RM_SUB_RFID_INVALIDPWD // Kill password가 일치하지 않음을 알림
    RM_SUB_RFID_RUDSUCCESS // UserData 읽기 명령을 성공적으로
                           완료 했음을 알림
    RM_SUB_RFID_RUDFAIL // UserData 읽기 명령 수행을 실패했음을 알림
    RM_SUB_RFID_NOTAG // UserData 읽기 명령을 수행할 태그가 없음을 알림
    RM_SUB_RFID_SRSUCCESS // Read 명령에 대한 수행중지를 성공했음을 알림
    RM_SUB_RFID_SRFAIL // Read 명령에 대한 수행중지를 실패했음을 알림
}
```

- 명령어

미들웨어 상단의 API들은 확장성을 고려하여 만들어져 있기 때문에 RFID가 현재 지원되지 않는 API가 존재 할 수 있다. 이 때문에 API들은 공통적으로 M_E_NOT_SUPPORT 이벤트를 포함한다.

⇒ 기본 명령어

void RM_Init(void) : 미들웨어 상단(Upper Layer)과 미들웨어 하단(Lower

Layer)의 연결

void RM_RegisterAddRcvEvent(FuncAddRcvEvent *func) : 이벤트 콜백 등록함수

▷ 리더 제어 명령어

1) **M_Int32 RM_Power(M_Boolean onoff) :** RFID 리더의 전원을 ON/OFF 한다.

▶ 매개변수

on, off

TRUE : Power ON

FALSE : Power OFF

▶ 반환 값

성공 0

실패

M_E_READER_POWERON 리더가 이미 Power On 상태에 있음을 알리는 값

M_E_READER_POWEROFF 리더가 이미 Power Off 상태에 있음을 알리는 값

M_E_READER_NOTEXIST 휴대 단말기에 리더가 없음을 알리는 값

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

2) **M_Int32 RM_Open(void) :** RFID 리더가 태그에 대한 제어 명령어를 받아 수행할 수 있는 상태가 되도록 한다. 리더와 통신할 수 있는 논리적 채널이 마련된다.

▶ 반환값

성공 0

실패

M_E_READER_OPEN	리더가 이미 Open 상태임을 알리는 값
M_E_READER_NOTEXIST	휴대 단말기에 리더가 없음을 알리는 값
M_E_CMD_FAIL	리더가 명령 수행을 실패했을 때

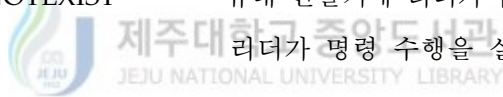
3) **M_Int32 RM_Close(void)** : RFID 리더와 통신하는 논리적 통신 채널을 닫는다. 이 함수가 호출되면 RFID 리더는 수행하던 태그 제어 명령어를 모두 종료한다.

▶ 반환 값

성공 0

실패

M_E_READER_CLOSE	리더가 Close 상태임을 알리는 값
M_E_READER_NOTEXIST	휴대 단말기에 리더가 없음을 알리는 값
M_E_CMD_FAIL	리더가 명령 수행을 실패했을 때



4) **M_Int32 RM_Reset(void)** : RFID 리더 내의 모든 레지스터 값을 초기화하고, 최초 Open된 상태가 되게 한다. 즉, 리더 내의 중앙처리장치와 메모리를 초기화한다. 이는 리더의 전원을 켜고, 통신 채널을 할당하는(즉 RM_Power(TRUE)와 RM_Open()을 수행한) 결과와 동일하다.

▶ 반환 값

성공 0

실패

M_E_READER_CLOSE	리더가 Close 상태임을 알리는 값
------------------	----------------------

M_E_READER_NOTEXIST 휴대 단말기에 리더가 없음을 알리는 값
M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

5) M_Int32 RM_SetArea(M_Uint8 area) : RFID 리더가 지역에 따른 전파 규격에 맞추어 동작하도록 가용 주파수를 설정한다.

▶ 매개 변수

[in] area 지역을 나타내는 8bits의 binary 값

값	의미	가용주파수(Mhz)
0000 0001(기본값)	대한민국	908.5 ~ 914
0000 0010	미국	902 ~ 928
0000 0100	유럽	865 ~ 868
0000 1000	일본	950 ~ 956
0001 0000	중국	-

▶ 반환 값

성공 0

실패

M_E_PARAM_INVALID 전달된 파라미터 값이 유효하지 않을 때

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

6) M_Int32 RM_GetArea(M_Uint8* area) : RFID 리더에 설정된 지역 값을 구한다.

▶ 매개 변수

[out] region 리더에 설정된 지역에 해당하는 값 (8bits binary 값)

값	의미	가용주파수(Mhz)
0000 0001(기본값)	대한민국	908.5 ~ 914
0000 0010	미국	902 ~ 928
0000 0100	유럽	865 ~ 868
0000 1000	일본	950 ~ 956
0001 0000	중국	-

▶ 반환 값

성공 0

실패

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

7) **M_Int32 RM_SetRFStrength(M_Uint8 rfstrength)** : RFID 리더의 RF 송출 신호의 세기를 설정한다. 이 명령을 활용하여 리더가 태그 인식을 위해 불필요한 송출 전력을 사용하지 않도록 하여 휴대 단말기의 전력 소비를 줄일 수 있음과 동시에 너무 많은 태그를 읽음으로 인하여 사용자에게 불필요한 태그 정보를 제공하는 것을 예방할 수도 있다.



▶ 매개 변수

[in] rfstrength RF 송출신호의 세기
(해당 RFID 리더의 최대 송출전력의 백분율로 표현)

▶ 반환 값

성공 0

실패

M_E_PARAM_NOTVALID 전달된 파라미터 값이 유효하지 않을 때

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

8) **M_Int32 RM_GetRFStrength(M_Uint8* rfstrength)** : RFID 리더에 현재 설정된, 최대 송출 전력 대비 백분율의 값을 얻는다.

▶ 매개 변수

[out] rfstrength RF 송출 신호의 세기를 저장
(해당 RFID 리더의 최대 송출전력의 백분율로 표현)

▶ 반환 값

성공 0

실패

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

9) **M_Int32 RM_IsOpen(M_Boolean* open)** : RFID 리더의 오픈 상태 여부를 얻는다.

▶ 매개 변수

[out] open 리더가 오픈 상태이면 true를, 아니면 false를 저장

▶ 반환 값

성공 0

실패

M_E_READER_NOTEXIST 휴대 단말기에 리더가 없음을 알리는 값

M_E_ERROR 그 외 에러를 알리는 값

10) **M_Int32 RM_IsBusy(M_Boolean* busy)** : RFID 리더가 현재 태그 제어 명령을 수행하고 있는지 여부를 얻는다.

▶ 매개 변수

[out] busy 리더가 태그 제어 명령을 수행하고 있으면 true를, 아니면 false를 저장

▶ 반환 값

성공 0

실패

M_E_READER_NOTEXIST 휴대 단말기에 리더가 없음을 알리는 값

M_E_ERROR 그 외 에러를 알리는 값

11) M_Int32 RM_SetReadCycle(M_Uint16 cycle) : RFID 리더가 태그 읽기를 수행하는 횟수를 설정한다.

▶ 매개 변수

[in] cycle 리더의 읽기 횟수

▶ 반환 값

성공 0

실패

M_E_PARAM_NOTVALID 전달된 파라메터가 유효하지 않을 때

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

12) M_Int32 RM_GetReadCycle(M_Uint16* cycle) : RFID 리더에 설정된 태그 읽기를 수행하는 횟수를 얻는다.

▶ 매개 변수

[out] cycle 리더에 설정된 읽기 횟수



▶ 반환 값

성공 0

실패

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

13) RM_SetReadDelayTime(M_Uint8 delaytime) : RFID 리더가 두 번 이상 태그 읽기를 수행할 경우, 한 번 태그 읽기를 수행하고 다음 태그 읽기를 수행하기까지의 대기 시간을 설정한다.

▶ 매개 변수

[in] delaytime 다음 태그 읽기를 수행하기까지 대기 시간(단위: ms)

▶ 반환 값

성공 0

실패

M_E_PARAM_NOTVALID 전달된 파라미터가 유효하지 않을 때

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

14) RM_GetReadDelayTime(M_Uint8* delaytime) : RFID 리더가 두 번 이상 태그 읽기를 수행할 경우, 한 번 태그 읽기를 수행하고 다음 태그 읽기를 수행하기까지 대기 시간을 얻어온다.

▶ 매개 변수

[out] delaytime 다음 태그 읽기를 수행하기까지의 대기 시간(단위:ms)

▶ 반환 값

성공 0

실패

M_E_CMD_FAIL 리더가 명령 수행을 실패했을 때

15) RM_ReaderStatus : RFID 리더의 상태와 기본 정보를 반환한다.

▶ 매개 변수

[out] rs 리더의 상태 정보가 저장

▶ 반환 값

성공 0

실패

NULL

▷ TagControl API

1) M_Int32 RM_ReadTID(M_Byte* uiiblock) : RFID 리더로 하여금 태그의 TID 를 읽게 하여 파라미터로 전달받은 uiiblock에 해당하는 큐에 결과를 저장하고, 리더의 동작 결과에 따라 정의된 이벤트를 발생시킨다.

▶ 반환값

성공 0 전달된 모든 파라미터가 유효할 경우

실패

M_E_READER_NOTOPEN 리더가 오픈되어 있지 않을 때

M_E_READER_BUSY 리더가 태그를 제어중일 때

void RM_StopRead(void) : RFID 리더로 하여금 현재 수행 중인 태그 데이

터(예: UII 셋, UII 블록, TID 혹은 User Data) 읽기 동작을 중지하게 한다. 리더는 이 명령을 수신하면 수행 중인 동작을 즉각 중지한다. 리더의 동작 중지 결과는 이벤트로 응용 프로그램에 전달된다.

2) M_Int32 RM_ReadUserData(M_Byte* uiiblock, M_Uint16 startaddr, M_Uint16 size) : RFID 리더로 하여금 ISO 18000-6 Type C 태그의 사용자 메모리(User Memory Bank)에 있는 데이터를 읽게 하여 그 결과를 전달받아 큐에 저장하고, 리더의 동작 결과에 따른 이벤트를 발생시킨다.

▶ 매개 변수

uiiblock	데이터를 읽고자 하는 태그의 UII 블록
startaddr	데이터를 읽고자 하는 태그 메모리의 시작 주소 (Byte 주소)
size	데이터를 읽고자 하는 태그 메모리의 크기 (Byte 단위)

▶ 반환 값

성공 0	전달된 모든 파라미터가 유효할 경우
실패	
M_E_READER_NOTOPEN	리더가 오픈되어 있지 않을 때
M_E_READER_BUSY	리더가 태그를 제어중일 때

3) M_Int32 RM_KillRFTag(M_Byte* uiiblock, M_Uint32 accpwd, M_Uint32 killpwd) : RFID 리더로 하여금 태그가 더 이상 사용되지 않도록 태그를 설정하게 하고(즉 Kill 명령이 수행된 이후에는 더 이상 해당 태그에 대해 데이터를 읽거나 쓸 수 없다.) 그 수행 결과를 전달받아 그에 따른 이벤트를 발생시킨다.

▶ 매개 변수

uiiblock	설정하고자 하는 태그의 UII 블록
----------	---------------------

accpwd 설정 시 사용되는 액세스(access) 패스워드 (32bit)
 killpwd 설정 시 사용되는 킬(kill) 패스워드 (32bit)

▶ 반환 값

성공 0 전달된 모든 파라미터가 유효할 경우

실패

M_E_READER_NOTOPEN 리더가 오픈되어 있지 않을 때
 M_E_READER_BUSY 리더가 태그를 제어중일 때

4) **M_Int32 RM_LockType(M_Byte* uiiblock, M_Uint32 mask, M_Uint32 password) :** RFID 리더로 하여금 마스크(mask) 비트에 따라 태그 내의 특정 메모리 블록에 데이터를 쓰거나 읽기를 수행할 수 없게 하고, 그 수행 결과를 전달 받아 그에 따른 이벤트를 발생시킨다.

▶ 매개 변수

[in] uiiblock 설정하고자 하는 태그의 UII 블록
 [in] mask ISO 18000 C태그의 Lock을 위한 20bit Mask 비트
 [in] password 태그 제어 시 필요한 액세스 패스워드(32bit)

▶ 반환 값

성공 0 전달된 모든 파라미터가 유효할 경우

실패

M_E_READER_NOTOPEN 리더가 오픈되어 있지 않을 때
 M_E_READER_BUSY 리더가 태그를 제어중일 때

▷ QueueControl API

- 태그 데이터 구조체

```
struct TagData {  
    M_Int32 index; //TagData 구조체의 인덱스  
    M_Uint8 tagtype; //태그의 규격 (ReaderStatus 구조체 참조)  
    M_Byte* uiblock; //태그의 UII Memory Bank의 내용  
    M_Byte* tid; //태그의 TID Memory Bank의 내용  
    M_Byte* udata; //태그의 User Memory Bank의 내용  
    M_Uint64 time; //태그를 읽은 시간  
};
```

1) **M_Int32 RM_CreateQueue(M_Uint8 size)** : RFID 리더에서 읽혀진 태그의 정보를 저장하는 큐를 생성한다.

▶ 매개 변수

[in] size 설정할 태그의 정보를 저장할 개수 (1 ~ 255개)

▶ 반환 값

성공 0

실패

M_E_QUEUE_EXIST 큐가 벌써 존재할 때

M_E_QUEUE_OUTOFBOUND 생성할 수 있는 큐의 최대 설정 범위를 초과할 때

2) **M_Int32 RM_DestroyQueue(void)** : RFID 리더에서 읽혀진 태그의 정보를 저장하는 큐를 삭제한다.

▶ 매개 변수

없음

▶ 반환 값

성공 0

실패 M_E_QUEUE_NOTEXIST 삭제하려는 큐가 존재하지 않을 때

3) **M_Int32 RM_ReadQueue(M_Boolean update, TagData* td) :** 큐에 저장된 데이터를 읽어온다.

▶ 매개 변수

[in] update 이 값이 TRUE 이면 읽어간 데이터를 큐에서 삭제하고, FALSE 이면 읽기만 수행한다.

[out] 큐의 첫 번째 데이터를 저장한다.

▶ 반환 값

성공 0

실패

M_E_QUEUE_EMPTY	큐에 읽을 데이터가 없을 때
M_E_QUEUE_NOTEXIST	큐가 존재하지 않을 때



4) **M_Int32 RM_WriteQueue(TagData* data, M_Boolean duplicate) :** 큐에 데이터를 저장한다.

▶ 매개 변수

[in] data 큐에 저장할 태그 관련 데이터

[in] duplicate TRUE이면 큐에 같은 데이터가 있어도 중복해서 저장하고,

FALSE 이면 같은 데이터가 큐에 있으면 이를 모두 삭제하고, 저장한다.

▶ 반환 값

성공 0

실패

M_E_QUEUE_NOTEXIST 큐가 존재하지 않을 때

M_E_QUEUE_FULL 큐에 데이터가 다 차서 더 이상 저장할 수 없을 때

5) **M_Int32 RM_DeleteQueue(void)** : 큐의 데이터 한 개를 삭제한다.

▶ 반환 값

성공 0

실패

M_E_QUEUE_NOTEXIST 큐가 존재하지 않을 때

M_E_QUEUE_DATANOTFOUND 삭제할 데이터가 큐에 없을 때

6) **M_Int32 RM_ClearQueue(void)** : 큐의 모든 데이터를 삭제한다.

▶ 반환 값

성공 0

실패

M_E_QUEUE_NOTEXIST 큐가 존재하지 않을 때

M_E_QUEUE_DATANOTFOUND 삭제할 데이터가 큐에 없을 때

7) **M_Int32 RM_GetNumQueue(void)** : 큐에 현재 저장되어 있는 데이터의 개수를 얻어온다.

▶ 반환 값

성공 큐에 저장된 데이터의 개수 (1 ~ 255)

실패

M_E_QUEUE_NOTEXIST 큐가 존재하지 않을 때

8) **M_Int32 RM_GetMaxNumQueue(void)** : 큐에 현재 저장할 수 있는 최대 데이터의 개수를 얻어온다.

▶ 반환 값

성공 큐에 저장할 수 있는 최대 데이터의 개수

실패 M_E_QUEUE_NOTEXIST 큐가 존재하지 않을 때

9) **M_Int32 RM_SortQueue(M_Unicode field1, M_Unicode field2, M_Boolean ascending)** : 큐에 저장된 데이터를 정렬한다.

▶ 매개 변수

[in] field1 데이터를 정렬하는 첫 번째 기준 필드

[in] field2 데이터를 정렬하는 두 번째 기준 필드

1 : 태그의 UI 셋

2 : 저장된 시간

[in] ascending 데이터를 정렬하는 방식

true : 오름차순

false : 내림차순

▶ 반환 값

성공 0

실패

M_E_QUEUE_NOTEXIST 큐가 존재하지 않을 때
M_E_QUEUE_DATANOTFOUND 정렬할 데이터가 없을 때

1 0) **M_Int32 RM_ValidateQueue(M_Boolean duplication, M_Int64 starttime, M_Int64 duration)** : 큐에 저장된 데이터 중에 유효하지 않은 데이터를 삭제한다. 유효성에 대한 검사는 중복된 데이터, 태그 정보가 읽힌 시간에 의하여 수행한다.

▶ 매개 변수

[in] duplication 중복된 데이터를 찾아 삭제할지 여부를 지시
 true : 중복 데이터 삭제
 false : 중복 데이터 미 삭제

[in] starttime 데이터가 유효한 시작 시간 (milli-second 단위의 시간)

[in] duration 데이터가 유효한 기간 (milli-second 단위의 시간)

▶ 반환 값

성공 0

실패



M_E_QUEUE_NOTEXIST 큐가 존재하지 않을 때
M_E_QUEUE_DATANOTFOUND 삭제할 데이터가 없을 때

▷ **FilterControl API**

1) **M_Int32 RM_AddFilter(M_Char* name, M_Byte* value, M_Byte* mask, M_Boolean inclusive)** : RFID 리더는 태그의 정보를 읽고 큐에 저장하는 과정에서 필터링을 하여 저장하도록 필터를 추가한다. 필터는 태그의 정보

에 대해 mask에 set된 비트에 대하여, value 값과 AND 연산을 한 결과가 value 값과 동일한 지 확인하여, 그 결과가 true이면 inclusive 값에 따라 태그의 정보를 저장하도록 한다. 추가된 필터는 기본적으로 enable 상태이다.

▶ 매개 변수

[in] name 필터의 이름

[in] value 필터링을 할 때, 이 값과 리더가 태그로부터 읽은 값 (UII Block)을 AND 연산하고, 그 결과가 동일한지 확인하기 위한 값

[in] mask 태그로부터 읽은 값 중, value와 필터링을 하기 위한 값을 선택하기 위한 마스크 값

[in] inclusive 이 값이 true이면 필터 결과가 true인 태그 정보만 큐에 저장 하고, false이면 필터 결과가 false인 태그 정보만 큐에 저장한다.

▶ 반환 값

성공 0

실패

M_E_FILTER_EXIST 저장할 필터값이 이미 존재할 때

2) M_Int32 RM_DeleteFilter(M_Char* name) : 필터를 삭제한다.

▶ 매개 변수

[in] name 필터의 이름

 null: 모든 필터를 삭제

▶ 반환 값

성공 0

실패

M_E_FILTER_NOTFOUND 삭제할 필터가 없을 때

3) **M_Int32 RM_EnableFilter(M_Char* name)** : 필터를 동작하도록 설정한다.

▶ 매개 변수

[in] name 필터의 이름
 null: 모든 필터를 설정

▶ 반환 값

성공 0

실패

M_E_FILTER_NOTFOUND 필터링을 수행할 필터 이름이 없을 때

M_E_FILTER_NOTEXIST 필터가 없을 때

4) **M_Int32 RM_DisableFilter(M_Char* name)** : 필터가 동작하지 못하도록 설정한다.

▶ 매개 변수

[in] name 필터의 이름
 null: 모든 필터를 설정



▶ 반환 값

성공 0

실패

M_E_FILTER_NOTFOUND 파라미터로 전달된 필터 이름이 없을 때

M_E_FILTER_NOTEXIST 필터가 없을 때