

논문 2009-46CI-2-16

ID 프로파일을 이용한 RFID 멀티 코드 변환기 연구

(The development of RFID multi-codes converter based on ID profiles)

이창열*, 모희숙**

(ChangYeol Lee and Hee-Sook Mo)

요약

산업에서 사용되는 ID는 기록 매체나 응용에 따라서 다른 표현 형태를 가질 수 있으며, 특히 RFID 태그에 기록되기 위하여 ISO/IEC 15962의 인코딩/디코딩 규칙에 따라 표현되어야 한다. 본 연구는 ID의 인터넷상에서 표현과 RFID 태그에서 표현을 자유롭게 상호 변환 지원할 수 있는 효과적인 알고리즘 개발에 초점을 두었다. 알고리즘의 핵심은 변환 대상이 되는 ID에 대한 XML 기반의 선언적 프로파일 정의와 이를 기반으로 효과적인 변환을 위한 3단계 논리적 변환 코드 체계를 도입하는 것에 있다. 본 알고리즘은 ISO/IEC 18000-6C 태그 기반 RFID에서 표현될 수 있는 3가지 유형의 메타 코드 체계(EPC, Non-EPC ID 이면서 OID 사용 코드와 사용하지 않는 코드)에 해당되는 대표적인 실존 IDs(EPC, ISO15459 KKR 코드, mCode)를 대상으로 적용하여 그 정확성에 대한 검증을 하였다.

Abstract

There are many different ID representation forms depending on the media or applications. In case of RFID tag, ID representation form must be followed by the rule of ISO/IEC 15962. In this study, we developed the efficient ID conversion algorithm between ID representation form on RFID tag and Internet. The main idea is on the use of XML based ID profiles and three step logical IDs forms. The algorithm was tested by the typical three kinds of real IDs such as EPC, ISO/IEC 15459 KKR Code, and mCode which are the typical meta-IDs can be defined in ISO/IEC 18000-6C tag.

Keywords : RFID tag, RFID code, ISO/IEC 18000-6C, ISO/IEC 15962, RFID code conversion

I. 서론

여러 산업에서 적용 대상이 되는 객체나 정보를 식별하기 위하여 ID(Identifier)를 사용한다. 예를 들어, 도서를 위한 ISBN (International Standard Book Number), 사람을 위한 주민등록번호, 또는 콘텐츠에 부여되는 DOI (Digital Object Identifier) (<http://www.doi.org/>)나 UCI (Universal Content Identifier) (<http://www.uci.or.kr/>)가 그 사례이다. 이러

한 ID는 사용 규칙과 표기를 위한 자체의 가이드라인이 존재하며, 해당 원칙에 의하여 표기와 사용을 하게 된다. 동일한 ID라도 다른 표기가 존재할 수 있기 때문에 표기법을 관리하는 것은 중요하다.

ID가 RFID(Radio Frequency Identification) 태그에 기록될 때, 이러한 표기법 관리는 더욱 중요하다. 왜냐하면, RFID는 도서에도 부착되고, 박스에도 부착되고, 차량에도 부착되는 범 산업적 용도로 사용하기 때문이며, 또한 사람이 볼 수 없는 태그 속에 이진법 형태로 기록되기 때문에 어느 분야에서나 일관되게 이를 해석하고 기록하는 것이 필요하다.

이러한 문제를 해결하기 위하여 다양한 형태의 시도가 있었다. 첫째는 코드 표기를 위한 규칙으로 RFID 태그에 데이터를 기록하는데 필요한 인터페이스 규약(ISO/IEC 15961^[1]) 정의와, 태그 인코딩/디코딩 규칙

* 정회원, 동의대학교 컴퓨터공학과
(Department of Computer Engineering,
DongEui University)

** 정회원, 한국전자통신연구원 융합기술연구부문
(IT convergence technology research lab., ETRI)

※ 본 연구는 한국전자통신연구원의 지원을 받아 수행되었음

접수일자: 2008년12월3일, 수정완료일: 2009년2월26일

ISO/IEC 15962^[2]) 정의를 통하여 어느 산업에서도 해당 ID를 정확히 해석할 수 있는 규칙을 제공하는 것이다.

두 번째 방법은 산업에서 사용하는 다양한 코드를 좀 더 커다란 RFID용 단일 코드의 하부 코드로 편입시키는 방법이다. 예를 들어 미 국방부의 CAGE (Commerical and Government Entity) 코드^[3]를 EPC (Electronic Product Code)의 하부 코드로 편입시키는 작업 등이 그러한 사례이다. ISO의 ISO/IEC 15454 코드^[4~5], EPCglobal의 EPC, 일본 u-IT 센터의 uCode^[6], 국내 모바일 RFID 포럼의 mCode^[7~8] 등이 이러한 메타 코드 체계를 표방한 대표적인 코드이다.

마지막으로 RFID 태그의 메모리 구조를 정의하는 것으로 태그의 메모리 구조는 태그의 주파수 영역의 표준과 밀접한 관련이 있다. 가장 보편적으로 사용하는 UHF 대역의 ISO/IEC 18000-6C^[9] 태그나 13.56MHz의 ISO/IEC 18000-3m3^[10] 태그는 동일한 메모리 구조를 가지고 있으며, EPC와 Non-EPC를 동시에 사용할 수 있는 태그로써 해당 주파수 대역의 사실상 표준 태그로 자리 잡았다.

그러므로 ISO/IEC 18000-6C 태그(이하 ISO/IEC 18000-3m3 언급은 생략함)에서 ISO/IEC 15962의 인코딩/디코딩 규칙을 기반으로 다양한 산업의 코드를 효과적으로 처리할 수 있는 알고리즘 개발은 RFID 시스템 전체의 상호운용성과 성능 향상에 반드시 필요한 기술이다.

코드 관련 유사 연구로는 NIDA에서 제시한MDS (Multi-Code Directory Service)에 관한 다양한 연구^[1~12]가 있었는데, 이들 연구는 특정 RFID 태그에 기록된 ID가 EPC인지, ISO/IEC 15459 인지, 아니면 uCode 인지를 구별하는 방법을 제시하고 있지만, 이는 실제로 태그에 기록된 ID를 구별하는 것이 아니라(즉 ISO/IEC 15962와 태그 메모리 구조를 고려하지 않고) 문서나 컴퓨터상에서 표현되는 2진법 기반 코드의 헤더 정보를 기반으로 코드를 구별하는 방법을 제시하고 있기 때문에 코드 구별에 현실성을 가지지 못하고 있다.

한국정보사회진흥원 가이드북^[13]에서는 RFID에서 사용하는 메타 코드 기반의 코드 체계에 대한 현황 분석을 제공하고 있지 효과적으로 코드를 상호 변환하는 것에 대한 연구를 진행하지는 못하였다.

ISO/IEC 18000-6(C) 태그에서 태그 사이 충돌 방지^[14]와 ISBN 코드를 효과적으로 기록하기 위한 연구^[15]가 있었지만 이 또한 다양한 코드의 변환을 위한 연구

는 아니었다.

본 논문에서는 ISO/IEC 15962 인코딩 규칙을 ISO/IEC 18000-6C 태그의 메모리 구조에 따라 설명을 진행하였고, 대표적인 메타 코드로 EPC, ISO/IEC 15459 KKR 코드, 그리고 mCode에 대한 소개를 하였다.

RFID 멀티 코드 변환 알고리즘에서는 다양한 ID를 처리하기 위하여 ID 정보를 XML로 표현한 ID 프로파일 개념과 ID가 ISO/IEC 18000-6C 태그 구조와 ISO/IEC 15962 기반 체계로 효과적으로 변환하기 위한 방법에 대한 연구를 진행하였다.

II. 본 론

1. 태그 인코딩 규칙

가. 태그 전체 구조^[16]

ISO/IEC 15962는 인코딩/디코딩 규칙을 정의할 뿐, 태그의 메모리 구조를 정의하지는 않는다.

ISO/IEC 18000-6C의 태그 구조는 그림 1과 같이 4개의 Bank로 구성되어 있다. Bank 0인 Reserved 영역은 접근 통제와 태그 Kill을 위한 패스워드를 넣는 영역이고, Bank 1인 UII(Unique Item Identifier)는 산업에서 사용하는 ID를 넣는 곳이며, Bank 2은 칩 제조업체가 칩의 ID를 기록하는 Tag ID(TID) 영역이며, Bank 3은 User 메모리로 선택적으로 태그를 사용하는 산업에서 임의로 값을 넣을 수 있게 하였다. 본 연구에서 대상이 되는 위치는 ID를 기록할 수 있는 Bank 1인 UII 영역에 한정한다.

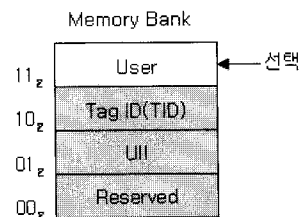


그림 1. ISO/IEC 18000-6C와 3m3 태그 구조

Fig. 1. ISO/IEC 18000-6C & 3m3 Tag Structure.

가. UII 영역과 인코딩 규칙

(1) UII 영역 세부 구조

UII 영역의 상위 레벨 구조는 그림 2와 같이 3개의

구성 요소		주소(16진수)	설명
CRC - 16(공통)		00 ~ 0F	태그와 리더 정보이동 매려 검사
PC(공통)	Length	10 ~ 14	Ull data Word 길이 (PC + Ull data word 길이 - 1)
	User Memory Indicator	15	User 메모리 사용여부 기록
	XPC Indicator(XI)	16	추가적인 PC 영역 사용여부 기록
	NSI	Toggle(T)	17
	AFI	18 ~ 1F	태그 응용 분야 식별 ID 값
Ull data (EPC인 경우 직접 값을 적고, Non-EPC인 경우 알 구조에 따라 기술)	DSFID	20 ~ 27	Ull data 구조 및 ObjectID(OID) 저장 방식 기록
	Precursor	28 ~ 2F	ObjectID & Object의 구조 및 형식 기록
ObjectID	Length	30 ~	ObjectID의 Byte 길이 - 1
	ObjectID		ObjectID 값
	Length		Object의 Byte 길이
	Object		RFID 코드(Non-EPC코드)

그림 2. Ull 영역 상세 구조
Fig. 2. Ull Bank Detail Structure.

세부 영역으로 구성되었다. 3개의 영역은 CRC (Cyclic Redundancy-Check) 영역, PC (Protocol Control) 영역, 그리고 Ull Data 영역으로 구성된다. 3개의 영역 중에서 실제적으로 ID를 기록하는 영역은 Ull Data 영역이며, 일반적으로 RFID 태그의 비트는 Ull Data 영역의 비트를 이야기한다. 예를 들어 96비트 RFID 태그라는 뜻은 Ull Data 영역 크기가 96비트라는 뜻이다.

(2) CRC 영역(16비트)

CRC는 PC 영역과 Ull Data 영역의 값에 따라 그림 3의 논리 연산기 구조에 의하여 자동으로 계산되어 기록된다^[16].

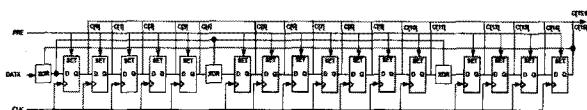


그림 3. CRC 계산 논리 회로
Fig. 3. CRC Calculation Logical Module.

(3) PC 영역(16비트)

PC 영역은 다음과 같이 구성된다.

- Length(5비트)
PC + Ull Data 길이 - 1의 Words(1 Word는 16비트) 값을 표현한다.
- User Memory Indicator(UMI) (1 비트)
User Memory의 사용 여부를 기록한다. 사용하면 "1"을 사용하지 않으면 "0"을 기록한다.
- XPC Indicator(XI)(1비트)
추가적인 PC 영역 사용 여부를 기록한다.
- NSI(Numbering System Indicator)(9비트)

* Toggle 비트(1비트)

EPC 사용시 "1"을, Non-EPC인 경우 "0"을 표현한다.

* AFI(Application Family Identifier)(8비트)

태그가 목표로 하는 응용의 유형을 기술한다. ISO 15961에서 정의되어있으며 대부분의 코드가 국제적으로 중요한 기관이나 응용에 할당되어 있으며, 상세한 사항은 관련 문서를 참조하면 된다^[1].

Toggle 비트 값이 "1"(EPC 사용)인 경우, AFI 코드를 사용하지 않기 때문에 2진수로 "0000 0000"을 표시한다^[3]. 참고적으로 본 논문에서 인용부호("")안에 있는 숫자는 2진수로 간주한다. AFI의 상세 정보는 계속 추가되고 변화될 수 있으며, 표 1은 2007년 6월 버전이다^[17].

표 1. AFI 코드 표
Table 1. AFI Code Table.

주소 (16진수)	용도	ISO15459 부호	ISO 규격
00-8F			
01	mCode(mCode 문서에서 정의)		15693-3
91	SCM 수송단위	15459-1	15961
92	SCM 수송단위, 위험물질 포함	15459-1	
93	SCM 회수자산(컨테이너)	15459-1/5	
94	SCM 회수자산(컨테이너), 위험물질	15459-1/5	
B1	ISO/IEC 15459 items	15459-4	
B4	ISO/IEC 15459 items, 위험물질 포함	15459-4	
C1	IATA 수화물		
C2	EDITEUR 도서 임대시 도서		
C3	ISBT(IS Blood Products)		
C4	ISO 17363 화물 컨테이너		
C14	ISO 17363 화물 컨테이너, 위험물질		
D0-FF			6816

(4) Ull Data 영역

Ull Data 영역은 EPC를 사용할 경우 첫 비트부터 EPC 코드 값을 기록하면 된다^[3]. 그렇지만 EPC를 사용하지 않을 경우 각 필드는 ISO/IEC 15962 규칙에 따른다.

□ DSFID(Data Storage Format Identifier)(8비트)

- Access Method(2비트)

RFID 태그 메모리에 데이터가 기록되는 형식을 정의한다. Directory 방식(메모리에 표시가 있어

표시대로 따라가면서 데이터를 찾는 방식), Non-Directory 방식(데이터가 메모리에 순차적으로 기록됨), Self-Mapping Tag 방식이 있다. 보통 Non-Directory 방식을 채택하며 값으로는 "00"으로 표현한다.

- Reserved(1비트) : 예약 비트. "0"으로 기록한다.
 - Data Format(5비트)
RFID 코드가 기록되는 Object 영역의 ID에 대한 정보를 기록한다. 다음과 같은 4가지 유형으로 구분된다.
 - * OID(Object ID)를 사용하지 않는 경우. 값 "00000"
 - * OID 값 자체를 UII Data ObjectID영역에 표시. 값 "00001"
 - * rootOIDEncoded를 이용. 값 "00010"
 - * 미리 정의된 RFID 코드를 이용. ISO 15459인 경우 "01001"로 표시.
- 표 2에 상세 정보 값이 기록되어 있다.

표 2. DSFID 값
Table 2. DSFID Value.

10진수	이름	설명
0	Not Formatted	오류발생. 형식화 불가능
1	FullFeatured	Object Identifier로 표현
2	RootOidEncoded	최소 2개 arc를 가진 root-OID
3	ISO 15434	
4	ISO 6523	ICD(I. Code Designator) 수준 코드로 DUNS, IP, NATO 등을 포함한다.
5	ISO 15459	
8	ISO 15961 combined	첫 arc가 {1 0 15961}이 된다.
9	EAN.UCC	
10	DI(Data Identifier)	
11	UPU	UPU 의견 없이 할당함
12	IATA	
30	Closed System	"0"은 폐쇄영역과 비형식화에도 쓴다.
31	Reserved	DSFID 확장을 위한 예약

- Precursor(8비트)
 - Offset(1비트) : Precursor가 더 있는지 없는지 표시한다. "0"일 경우 마지막 Precursor, "1"일 경우 추가로 Precursor가 더 있음.
 - Compaction Type Code(3비트) : UII Data object

영역에 기록되는 데이터 인코딩 정보를 알려준다. 상세 정보는 표 3과 같다.

표 3. 코드 압축 유형
Table 3. Code Compaction Type.

10진수	이름	설명
0	응용에서정의	응용에서 정의함. 처리 안함
1	Integer	Integer(2비트)
2	Numeric	숫자 (4비트)
3	5비트	알파벳 대문자
4	6비트	알파벳 대문자, 숫자 등
5	7비트	ASCII Code
6	8비트 스트링	unaltered 8비트
7	UTF-8 스트링	외부 압축. 처리 안함

- Relative OID(5비트) : 하부 OID 값을 기록하는 한다. 목표 ID를 정의한 표준 문서의 하위 문서 번호를 기록한다.

□ ObjectID(가변 길이)(ID 유형에 따라 생략됨)
OID는 특정 서비스/네트워크를 식별하기 위한 ITU에서 부여한 ID 체계로 계층(Arc) 별로 ID 상세 구조가 구성된다. 10진수로 구성되어 있으므로, RFID 태그에 기록 시 2진수로 변환하여야 한다. DSFID의 Data Format 중에서 '미리 정의된 RFID 코드'를 이용하는 경우 본 OID를 생략한다. 예를 들어 ISO 15459 코드인 경우 OID를 생략한다.

- ObjectID Length(8비트) : ObjectID의 바이트 -1을 표현한다.
- ObjectID(길이 가변적임) : OID를 태그에 기록하기 위한 2진수로 변환하는 과정은 표준 문서^[2]에 기록되어 있으며, 해당 변환 규칙에 따라 나타나는 결과 비트의 길이는 가변적이다. mCode^{[10][11]}인 경우 OID 값이 {0 2 450 1}이며, 이를 2진수로 변환하면, "1101 1111 0000 0010 1000 0011 0100 0010 0000 0001"로 표현된다.

- Object
 - 표현하고자 하는 객체의 ID를 표현한다.
 - Object Length(8비트) : 객체에 할당된 바이트를 표현한다.
 - Object(가변적임) : ID를 태그에 인코딩 원칙에 따라 구현한다. Precursor의 Compaction Type에 따라 ID를 기록한다.

2. 기존 코드 분석

가. EPC

EPC는 EPCglobal Inc.가 운영하는 코드체계로 공급망에서 사실상 표준으로 사용한다. 헤더 부분 정보에 따라 GID, SGTIN, SSCC, GRAI, GIAI, GSRN, GDTI, 그리고 DoD 코드를 포함할 수 있다^[3].

나. ISO 15459 KKR 코드

□ ISO 15459

ISO 15459^[4-5]는 여러 산업 사이에 물품이 교환될 때 산업 사이에 해당 물품에 부여한 ID의 충돌을 방지하기 위하여 산업별로 수송 단위에 Prefix를 부여하는 방식의 표준이다. 여기서 할당하는 Prefix를 IAC(Issuing Agency Code; 발급기관 코드)라고 부른다. 예를 들어 국제 우편 연합(UPU)에는 'J'라는 IAC를, 일본의 전자거래진흥원은 'LA'라는 IAC를 발급하였다^[18]. GS1에도 IAC를 발급하였지만, GS1이 수용을 하지 않고 독자적인 EPC 체계를 사용함으로써 예외적 상황이 발생하였고, 본 사항에 대한 타협안으로 ISO 18000-6C 태그가 출현한 것이다. 즉 ISO 18000-6C 태그는 ISO 15459의 코드 체계와 EPC 코드 체계를 동시에 수용하는(Toggle Bit를 사용하여) 태그인 것이다.

□ ISO 15459 KKR 코드

ISO 15459의 IAC를 부여받기 위하여 해당 산업이 국제적으로 교류를 하면서 ID의 충돌을 발생시킬 수 있는 분야임을 증명하는 수단으로써 ISO 회원국 2개국 이상에서 사용함을 증명하여야 한다. 이러한 원칙에 예외적으로 모든 나라의 행정 분야에서 사용할 수 있는 'K'로 시작하는 IAC를 국가별로 할당하고 있다. 즉 'K'와 해당 국가의 ISO 표기 코드(ISO 3166-2) 2자리를 결합한 코드를 각 국가별로 사용하게 한 것이다. 이러한 규칙을 적용하면 우리나라의 행정용 코드로 'KKR'이라는 IAC가 배정되는 것이다.

이에 따라 우리나라의 'KKR' IAC를 부여한 기관은 기술표준원의 추천에 따라 유통물류진흥원(GS1 KR)이 선정되었다. 즉 유통물류진흥원은 ISO 15459의 IAC 'KKR'을 관리하는 기관으로 된 것이며, 이에 따라 국내에 세부 코드 배정안을 행정자치부와

함께 마련하였다^[18-19]. 2008년 현재 '조달청', '국가기록원', '검찰청', '문화재청' 등이 IAC 'KKR'로 시작하는 코드의 세부 기관 번호를 부여받아 적용하고 있는 상태이다.

그렇지만 아직 'KKR' 코드에 대한 호칭, 구문 규칙 등을 공개하지 않은 상태이기 때문에(연구 보고서는 있지만 공식적으로 'KKR' 코드 운영 지침 서 등이 발표되지 않았음) 본 연구에서는 편의상 ISO 15459의 'KKR'를 IAC로 가지는 코드를 Administrative RFID Code(ARC)라는 용어로 표현하기로 한다.

다. mCode

mCode^[7-8]는 모바일용 코드로 MRF(Mobile RFID Forum)에서 제정하고, NIDA가 운용하는 코드이다. mCode는 ITU로부터 우리나라에 배정된 OID 번호를 사용하여 국내 표준안을 제출하였으며, 세부 코드 운영안을 마련하였고, 확산에 주력하고 있다.

3. RFID 멀티 코드 변환 알고리즘

가. 구성 및 흐름도

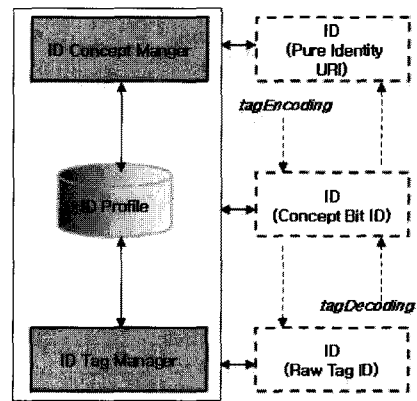


그림 4. RFID 멀티코드 변환 알고리즘 개념 구성도
Fig. 4. RFID Multi-codes Conversion Algorithm.

RFID 멀티 코드 변환 알고리즘은 본 논문에서 정의하는 다음과 같은 3가지 유형의 ID를 취급한다.

- (1) Pure Identity URI (Uniform Resource Identifier) : 응용에서 사용하는 URN 기반^[20] ID로 인코딩과 무관한 정보를 표현하는 'Canonical Form'을 표현한다.

- (2) Concept Bit ID : 특정 태그와 무관한 표현으로 Pure Identity URI가 비트 스트링으로 변환된 형태로 해석할 수 있다.
- (3) Raw Tag ID : 특정 태그에 기록된 비트 스트링으로 ISO/IEC 15962 규칙과 ID 사용 일반 원칙에 따라 기록된다.

코드 인코딩 과정은 *tagEncoding* 함수를 통하여 (1) -> (2) -> (3)을 거치는 것이고, 디코딩 과정은 *tagDecoding* 함수를 통하여 (3) -> (2) -> (1)의 과정을 거치는 것이다. 각 과정에서 필요한 정보는 다음과 같다 :

- (1) -> (2) 과정 : Pure Identity URI에서 표현이 생략된 항목에 관한 정보, 가변 길이를 사용하는 항목인 경우 가변 길이 변환 규칙, 코드 해석 단위(문자당 해석/전체 값으로 해석) 정보 등의 정보가 필요하다.
- (2) -> (3) 과정 : User Memory 사용여부, AFI 값 정보, 코드 압축 원칙, OID 사용 여부 등의 정보가 필요하다.

위에 기술한 바와 같이 (1) -> (2) -> (3) 또는 역으로 코드를 변환하는데 다양한 정보를 필요로 하며 일부는 코드를 선택한 순간 고정된 값으로, 일부는 선택적 값으로 정해질 수 있으며 이러한 정보를 기록 관리하는 수단으로 본 연구에서 'ID 프로파일' 체계를 제시하였다.

나. ID 프로파일

(1) 정의와 단위

ID 프로파일은 RFID 태그에 기록할 ID에 대한 요약 정보를 기록하는 것으로 본 연구에서 XML 기반 구조를 정의하였다. 적용 대상 코드의 단위는 구문 구조 정보가 다른 ID로 정의한다. 예를 들어 다음과 같은 사례를 가질 수 있다 :

- EPC인 경우 : 세부 헤더 종류(SGTIN, SSCC 등)에 따라, 그리고 목표 태그 비트에 따라(96 비트, 128비트 등) 구별하여 각각 프로파일을 기록한다. 본 연구에서 샘플로 제시한 'ID 프로파일' 대상은 96비트 EPC SGTIN 코드이다.
- ARC인 경우 : 헤더 개념이 없으므로 목표 태그 비트에 따라 구별하여 프로파일을 기록한다. 본 연구에서

는 96비트 ARC를 사례로 들었다.

- mCode인 경우 : Class에 따라 구문 구조가 다르므로 클래스(A, B, C, D, E, F, G)에 따라 구별하여 프로파일을 기록한다. 또한 목표 태그 비트에 따라 구별하여야 한다. 본 연구에서는 160비트 mCode E Class를 기준으로 사례를 살펴보았다.

본 알고리즘에서 정의한 3가지 유형의 ID(EPC, ARC, mCode)는 RFID 태그에서 사용할 수 있는 대표적인 유형의 코드를 대변한다. 즉 ISO/IEC 18000-6C 태그는 EPC와 Non-EPC 겸용 태그이며, Non-EPC인 경우 메모리에서 'OID' 필드를 사용하는 코드와 사용하지 않는 코드로 분류가 되는 것이다. 즉 'ARC'는 'OID' 미사용 코드이며, 'mCode'는 'OID' 사용 코드로 분류되기 때문에 이들 3가지 유형에 대하여 본 ID 프로파일을 적용하는 것은 RFID에 적용할 수 있는 모든 ID 유형에 대한 검증과 같은 것이다.

(2) 상위 레벨 구조

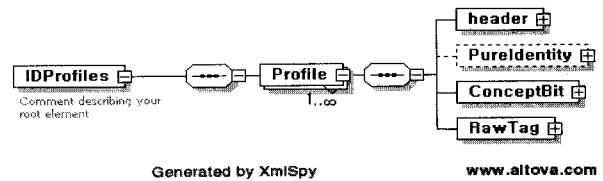


그림 5. ID 프로파일의 상위 레벨 구조
Fig. 5. Top-Level Structure of ID profiles.

ID 프로파일에는 각 대상 코드의 헤더 정보, Pure Identity URI 정보, Concept Bit ID 정보, 그리고 Raw Tag ID 정보를 기록한다.

(3) 헤더

헤더는 목표 ID에 대한 일반적 정보를 기록한다. 전체 구조는 그림 6에 기술되었다. 각각의 내용을 살펴보면

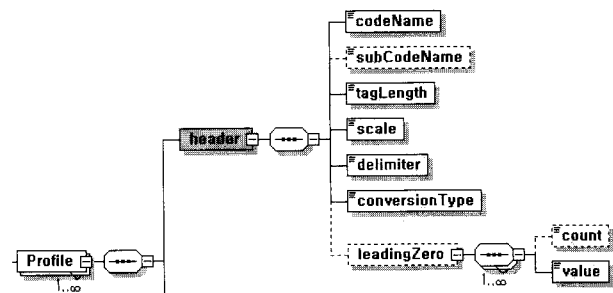


그림 6. 헤더 부분
Fig. 6. Header part.

기로 한다.

- codeName : 대상 코드 이름을 기록한다.
- subCodeName : 하위 코드 이름으로 EPC인 경우 SGTIN 등을 가질 수 있으며, mCode는 E Class의 예를 들었다.
- tagLength : 목표로 하는 RFID 태그 길이를 기록한다. mCode인 경우 OID를 사용하기 때문에 최소한 160비트 이상이 되어야 한다.
- scale : Pure Identity URI에서 표기하는 문자의 범위로 EPC는 'decimal'를, ARC는 'alphanumeric'을, mCode는 'hexa'(decimal)을 기록한다.
- conversionType : 각 항목이 비트로 변환될 때, 1개 단위(문자)씩 정해진 길이의 비트로 변환하는지 (each) 아니면, 전체 값을 계산하여 변환하는지 (group)을 표현한다. EPC는 전체 값을 계산하여 적절한 값에 대한 비트를 변환(group)한다.
- leadingZero : conversionType의 값이 'each'인 경우 항상 leadingZero가 항상 'true'(Pure Identity URI에서 표기 시 특정 항목의 앞부분에 '0'을 채워서 기록한다)이기 때문에 기록하지 않는다. 'group' 값을 가지는 것만 본 사항에 대하여 기록을 하며, 디폴트가 'true'가 되기 때문에 'true'가 아닌 사항만 기록하면 된다. EPC/SGTIN인 경우 마지막 항목(6번째 항목; 표 5에 설명)인 'SerialNumber'만 leading zero가 'false'가 된다. 상세 정보는 표 4에 기록되었다.

표 4. 헤더 정보
Table 4. Header data.

codeName	EPC	ARC	mCode
subCodeName	SGTIN		E
tagLength	96	96	160
scale	decimal	alphanumeric	hexa
delimiter	dot	dot	blank
conversionType	group	each	each
leadingZero	count	6	
	value	false	

(4) Pure Identity URI

Pure Identity URI 수준에서 코드에 대한 구문 구성과 Concept Bit ID로 변환 시 숨겨진 정보를 확인한다. Pure Identity URI 부분은 구문구조 정보를 기록하는 부분('code' 부분)과 숨겨져 있어 표현되지 않지만

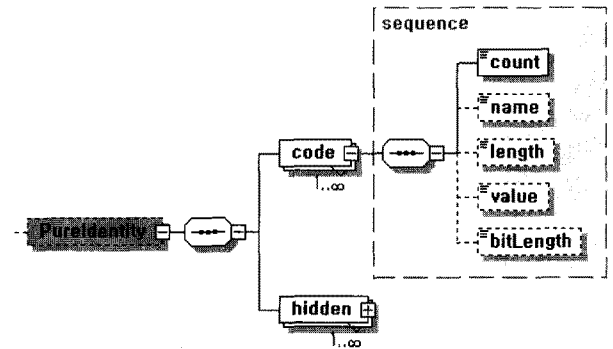


그림 7. Pure Identity URI 부분
Fig. 7. Pure Identity URI part.

표 5. Pure Identity URI 정보
Table 5. Pure Identity URI Data.

		EPC	ARC
code	count	4	2
	name	CompanyPrefix	InstitutePrefix
	count	5	3
	name	ItemReference	ObjectNumber
	count	6	4
	name	SerialNumber	SerialNumber
hidden	count	1	1
	name	Header	IAC
	count	2	
	name	Filter	
	count	3	
	name	Partition	

Concept Bit ID에서 표현해야하는 부분('hidden' 부분)을 함께 기록한다. 그림 7에서 코드 정보는 'sequence'라는 'complexType'를 사용하여 표현하였으며, 본 Pure Identity URI 부분에서는 'sequence' 구조의 'count'와 'name'만 이용한다. 'name' 부분이 선택이지만 Pure Identity URI에서는 필수적으로 기록해야한다.

mCode인 경우, 응용에서 데이터 표현 자체가 비트 스트링(표현은 16진법)이기 때문에 Pure Identity URI 표현과 Concept Bit ID와 동일하다. 그러므로 Pure Identity URI 표현을 여기서 생략한다.

(5) Concept Bit ID

Concept Bit ID는 Pure Identity URI를 RFID를 고려하지 않은 비트 스트링으로 변환된 ID라고 생각하면 쉽다. Pure Identity URI가 비트 스트링으로 표현하기 위한 정보, Pure Identity URI에 명시되지 않은 항목에 대한 상세 기술 정보 등을 표현하는 것이다.

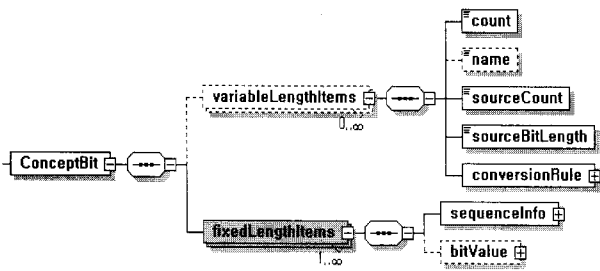


그림 8. Concept Bit ID 부분
Fig. 8. Concept Bit ID part.

표 6. Concept Bit ID 일부 정보
Table 6. The sample elements of Concept Bit ID.

고정 길이 항목 일부	EPC	ARC	mCode
count	1	1	2
name			Class
length		3	1
value		KKR	4
bitLength	8	18	4
fixedValue	00110000	0010110010110100 10	0100

Concept Bit ID를 정의한 구조는 그림 8에 표현하였다. 여기에서 모든 항목을 가변 길이 항목('variableLengthItems')과 고정 길이 항목('fixedLengthItems')으로 구별하였고, 고정 길이 항목 중에서 가변 길이 항목의 길이를 정하는 항목은 가변 길이 항목 내에 변환 규칙 정보로 표현하였다. 즉 가변길이 항목 중에서 'source'로 시작하는 모든 요소는 가변 길이 항목의 길이를 결정하는 고정 길이 항목에 관한 사항이다. 일부 정보만 살펴보면 표 6과 같다.

표 6은 고정 길이 항목 중 일부만 표현한 것이다. 대부분이 'name'을 생략한 것은 Pure Identity URI에서 언급되었기 때문이며, mCode는 Pure Identity URI를 생략했기 때문에 'name'에 값을 넣었다. 'length'와 'value'는 해당 표기('header'에 있는 'scale'의 단위)에 따른 길이를 기록하였다.

(6) Raw Tag ID

Raw Tag ID는 ISO 15962의 인코딩/디코딩 규칙에 따라 Concept Bit ID를 RFID 태그에 맞게 변환하여 저장하는 형식이다.

EPC인 경우 'PC' 영역만 기록하고 바로 'Object' 영역의 'value'를 기록하면 된다.

전체에서 사용하는 'listType'은 XML 표기법으로

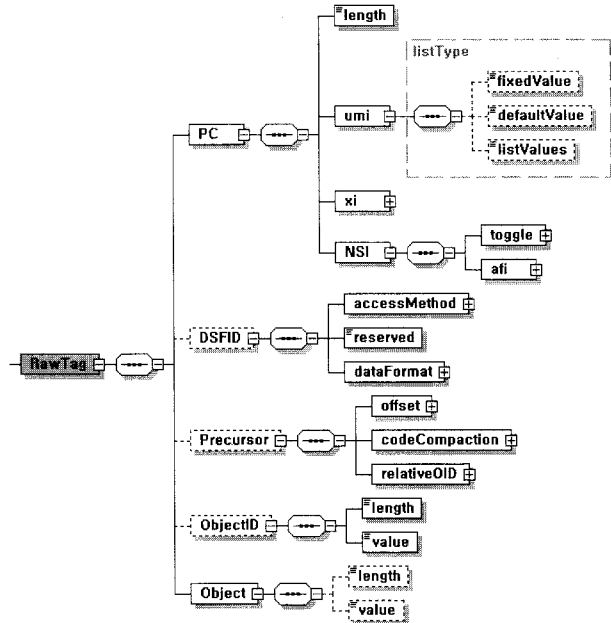


그림 9. Raw Tag ID 부분
Fig. 9. Raw Tag ID part.

complexType을 가지며 특정 항목이 가지는 값(비트)의 정보를 고정된 값, 디폴트 값, 또는 나열형을 기록하는데 사용한다. 일부 항목만 살펴보기로 한다.

- PC 영역의 'length' : PC 영역과 UII Data 영역을 합친 길이 Word에서 1을 뺀 사항으로, EPC/ARC인 경우 96비트 태그를 사용하기 때문에 $(16 + 96)/16 - 1 = 6$ 을 "00000"부터 기록하기 때문에 "00101"이 되며, mCode인 경우 160비트 태그를 사용하기 때문에 "01001"을 사용한다.
- DSFID 영역의 'dataFormat' : DSFID는 UII Data 영역으로 EPC인 경우 해당 사항이 없으며^[8], ARC인 경우 5에 해당하는 "01001"을, mCode인 경우 1에 해당하는 "00001"로 정의된다.
- Precursor 영역의 'codeCompaction' : 코드 압축은 'Object' 영역의 값을 기록하는 코드 체계를 정의하는 것으로 표 3에서 정의한다. ARC인 경우 6비트 압축을 사용^[2]하기 때문에 "100"으로, mCode인 경우 응용에서 정의하기 때문에 "000"으로 정의한다^[10~11].
- ObjectID 영역의 'value' : ARC인 경우 DSFID에 정의된 코드를 사용하기 때문에 ObjectID를 생략한다. mCode인 경우만 해당된다. mCode는 OID 표현으로 {0 2 450 1} 으로 정의하기 때문에 OID 계산 법^[2, 10]에 의하면 "0000 0010 1000 0011 0100 00100000 0001"와 같은 값이 도출된다.

다. 비트 배정 사례

그림 10은 ID 프로파일 정보가 기록하는 정보 중에서 96비트 ARC에 대하여 비트별로 할당된 값을 보여주고 있다. 96비트 태그에 실제적으로 사용할 수 있는 비트는 그림 10의 맨 우측에 있는 54비트이며, 6비트 인코딩을 하기 때문에 9개의 alphanumeric 문자를 기록할 수 있는 것이다.

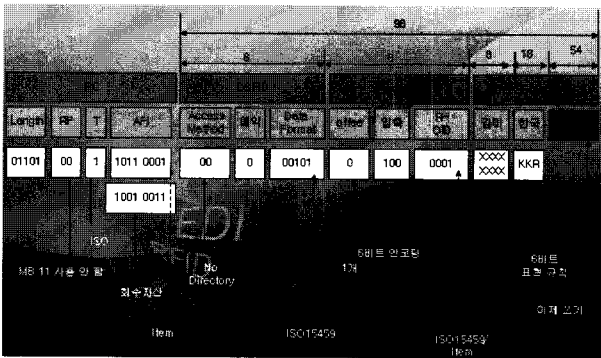


그림 10. 96비트 RFID 태그를 위한 ARC 비트 할당 예 Fig. 10. ARC Bit Assignment sample for RFID 96 bit tag.

라. 구현

그림 11의 멀티 코드 변환기는 입력으로 'Pure Identity URI' 또는 'Raw Tag ID'를 입력받아서 코드를 변환하는 시스템으로 샘플 코드(ARC 코드)를 입력 후, 'Confirm' 버튼을 클릭 한 후 화면이다.

버튼 'Confirm'은 입력 창이 'Pure Identity URI' 창인지, 'Raw Tag ID' 창인지 확인 후, 코드를 구별하고, 해당 코드에 대한 'ID 프로파일'을 로드하여, 화면에 필수 사항, 선택 사항 등에 대한 정보를 보여준다.

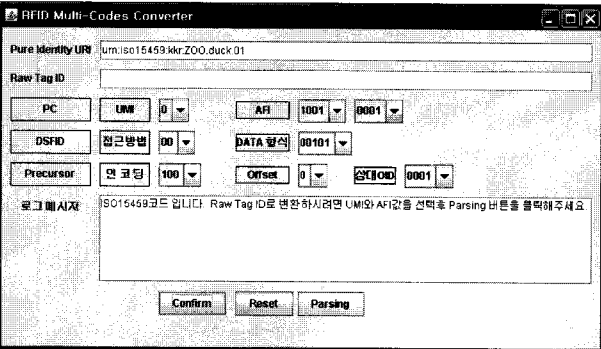


그림 11. RFID 멀티 코드 변환기 - 'Confirm' 버튼 클릭 후 화면 Fig. 11. RFID multi-codes converter - Display after click 'Confirm' button.

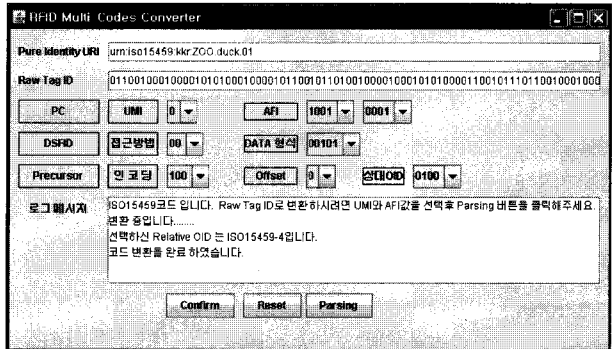


그림 12. RFID 멀티 코드 변환기 - 'Parsing' 버튼 클릭 후 화면 Fig. 12. RFID multi-codes converter - Display after click 'Parsing' button.

그림 12는 사용자가 선택해야하는 메뉴(그림 11 화면에서 '상대 OID' 값이 "0001"이었는데, 그림 12에서 '상대OID' 값이 "0100"으로 바뀐 것을 알 수 있다)에 대하여 선택을 한 뒤 'Parsing' 버튼을 클릭한 화면으로 최종적으로 Raw Tag ID 값이 해당 화면에 나타난 것을 볼 수 있다.

III. 결 론

본 논문에서는 여러 산업에서 사용하는 코드를 RFID에 구현하는데 필요한 ID 프로파일 기반 효과적인 변환 알고리즘에 대한 연구를 진행하였다. 변환 알고리즘은 ID 프로파일을 우선 로드한 상태에서, 입력 ID를 받아서 해당 프로파일에 적용하면서 단계별로 ID를 변환하여 가는 것으로 구성되어 있다. 즉 'Pure Identity URI' 형태의 입력(인코딩)은 'Concept Bit ID'를 거쳐서 최종적으로 'Raw Tag ID'로 변환되는 것이며, 입력이 'Raw Tag ID'가 오면 그 반대로 진행(디코딩)한다.

본 연구에서 대상으로 한 ID는 'EPC', 'Non-EPC이면서 OID 미사용 코드', 그리고 'Non-EPC이면서 OID 사용 코드' 3종류에 대하여 적용하였으며, 이는 ISO/IEC 18000-6C 태그에서 물리적 메모리 구조가 다른 3가지 형태를 각각 반영하고 있기 때문에, ISO/IEC 18000-6C 태그에서 허용 가능한 모든 유형의 ID에 적용한 것으로 간주할 수 있다.

참 고 문 헌

[1] ISO/IEC 15961, "Information technology - RFID

- for Item Management - Data Protocol : application interface”, IS, ISO/IEC, October, 2004.
- [2] ISO/IEC 15962, “Information technology - RFID for Item Management - Data Protocol : data encoding rule and logical memory functions”, IS, ISO/IEC, October, 2004.
- [3] EPCglobal Inc., “EPCglobal Tag Data Standard” Version 1.4 ratified specification, June 11, 2008.
- [4] ISO/IEC15459-1, “Information technology - Unique identifiers - Part 1: Unique identifiers for transport units”, Dec., 2005.
- [5] ISO/IEC15459-2, “Information technology - Unique identifiers - Part 2: Registration procedures”, Dec., 2005.
- [6] Ubiquitous ID Center, www.uidcenter.org
- [7] 한국인터넷진흥원, “RFID 코드 설계 및 적용 지침서”, NIDA, 2007년 12월
- [8] 한국정보통신표준협회, “모바일 RFID 코드체계 및 태그 데이터 구조”, TTAS, KO-06.0105/R1, 2006년 12월
- [9] ISO/IEC 18000-6C, “Information technology - RFID for item management - Part 6 C : Parameters for air interface communications at 860 MHz to 960 MHz”, Jul., 2005.
- [10] ISO/IEC 18000-3, “Information technology - RFID for Item Management : - Part 3 : Parameters for air interface communications at 13.56 MHz”, Jul., 2005.
- [11] 박은정, “글로벌 RFID 네트워크의 다중 코드 디렉토리 서비스를 위한 코드 필터링 방법”, 아주대학교 정보통신전문대학원 석사학위 논문, pp 18 - 38, 2005.
- [12] 이용환, “다중 Unique Item ID 체계를 위한 트리 기반 식별 알고리즘”, 아주대학교 정보통신전문대학원 석사학위 논문, pp29-34, 2006.
- [13] 한국정보사회진흥원, “RFID 코드체계 - RFID 적용을 위한 가이드북”, pp38-55, 2007.
- [14] 강봉수, 임정현, 김홍수, 양두영, “RFID 프로토콜 (ISO 18000-6)에서의 충돌방지 알고리즘 성능 분석”, 대한전자공학회 논문지 44권 CI편 5호, pp 20 - 27, 2007년 9월
- [15] 정필성, 정원수, 오영환 “도서 관리를 위한 RFID 태그 데이터 포맷 설계 및 시스템 구현”, 대한전자공학회 논문지 44권 CI편 3호, pp 1 -7, 2007년 5월
- [16] EPCglobal Inc., “EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860Mhz - 960Mhz”, Version 1.1.0 Draft 1, 19 July 2005.
- [17] ISO/IEC 15961, “ISO15961-2 Data Construction Register”, Draft SG1, ISO/IEC, June, 2007
- [18] 이창열, “RFID(무선인식), 고유 식별자 표준화 연구 보고서”, 한국표준협회, 2006년 5월
- [19] 이헌배, 이창열, “행정기관 RFID 식별코드 관리방안 연구보고서”, 행정자치부, 2007년 12월
- [20] M. Mealling, “A URN Namespace of Object Identifiers”, RFC 3061 IETF, February 2001.

저 자 소 개



이 창 열(정회원)
1985년 고려대학교 수학 학사 졸업.
1991년 고려대학교 전산학 석사 졸업.
1997년 Paris VII 전산학 박사 졸업.

1987년~1994년 ETRI
1997년~2000년 KERIS
2000년~현재 동의대학교
<주관심분야 : RFID, USN, 메타데이터>



모 희 숙(정회원)
1989년 전남대학교 전산통계 학사 졸업
2002년 충남대학교 컴퓨터과학 석사 졸업
1990년~현재 ETRI

<주관심분야 : RFID/USN, ad-hoc 네트워크, 위성 관제 >