
ALE 미들웨어를 위한 KKR 코드 변환

변지웅* · 변영철** · 이동철***

KKR code conversion for ALE middleware

Ji-Woong Byun* · Yung-Cheol Byun** · Dong-Cheol Lee***

본 연구는 교육과학기술부와 한국산업기술재단의 지역혁신인력양성사업으로 수행된 연구결과임

요 약

RFID 미들웨어는 RFID 리더 장치에서 들어오는 EPC 데이터를 응용이 원하는 형태로 가공하여 응용으로 전송하는 시스템 소프트웨어이다. 한편, 사실상 국제 표준을 선도하는 EPCglobal에서 제안한 RFID 미들웨어 스페인 ALE 미들웨어는 EPC 코드 체계만을 고려하고 있다. 따라서 국내 RFID 산업의 활성화를 위하여 NIDA에서 발표한 KKR이라는 새로운 코드 체계는 ALE 미들웨어에서 처리할 수 없는 문제점이 발생한다. 본 논문에서는 ALE 미들웨어에서 KKR 코드를 처리할 수 있는 방법을 제안하였으며, 이를 실제로 구현하여 ALE 미들웨어에 적용하여 테스트하였다. 또한 KKR 코드를 ALE 미들웨어에서 적절히 처리할 수 있도록 URN 데이터 형식을 제안하였다. 본 논문에서 제안하는 KKR 코드 변환 방법 및 데이터 표현 형식을 사용할 경우, 국제 표준을 따르는 기존의 ALE 미들웨어에서 KKR 코드를 효과적으로 처리할 수 있을 뿐만 아니라 RFID 미들웨어 및 기존 시스템과의 연동도 효과적으로 지원할 수 있다.

ABSTRACT

RFID middleware is the system software that converts EPC data collected from RFID reader devices into meaningful data, that users want, and transfers the result to the users with a variety of protocols. ALE specification, the RFID middleware specification proposed by EPCglobal that is a leading group of de facto international standards with respect to RFID, can handle only EPC code. Meanwhile, a new code system which is called KKR was proposed by NIDA in order to represent RFID tag data in a specific manner and cultivate the domestic RFID industry. In this case, the existing RFID middleware can not process KKR code system because of inherent attributes of the system. In this paper, we proposed a method of KKR code conversion to properly process KKR code in ALE middleware, implemented our method in real, and adopted it into existing ALE middleware. Also, we proposed a new URN format for RFID tag data to manipulate KKR code in ALE middleware. By using the proposed method, the existing ALE middleware conforming to international specification can handle KKR code effectively, and the integration among RFID middleware and legacy systems can be easily supported also.

키워드

RFID middleware, ALE, KKR, kCode, Code Conversion

* 제주대학교 대학원 컴퓨터공학과, 첨단기술연구소

접수일자 2008. 08. 14

** 제주대학교 통신컴퓨터공학부 교수, 교신저자

*** 제주대학교 경영정보학과 교수

I. 서 론

마크 와이저(Mark Weiser)가 주창한 차세대 유비쿼터스 컴퓨팅 패러다임을 실현하기 위한 기술로서 흡 네트워크, 무선 센서 네트워크 등의 기술이 주목받고 있다. 이와 함께 최근 주목받는 기술인 **RFID(Radio Frequency IDentification)**는 일련번호가 저장되어 있는 **RFID 태그**를 사물에 부착하여 비접촉 방식으로 자동 인식하는 기술로서 유비쿼터스 패러다임 실현을 위한 주요 기술 중 하나이다[1, 2].

RFID 기술의 경우 **RFID 태그**와 리더 장치와 같이 **RFID 하드웨어** 분야의 기술들이 주로 자리를 잡고 있으나 점차적으로 **RFID** 응용 소프트웨어와 미들웨어에 관한 연구가 활발하게 이루어지고 있다. **RFID** 리더 장치로 들어오는 태그 데이터를 수집하고 의미 있는 정보를 제공하여 응용으로 보내는 미들웨어가 없다면 하드웨어 장치 사용 및 응용 서비스 구현 측면에서 비효율적일 수 있다. 이때 **RFID** 하드웨어 및 응용 기술 간의 인터페이스 기능을 제공함으로써 기술의 효율성을 높여 주는 것이 **RFID 미들웨어**이다[3].

RFID 관련 사실상의 국제 표준 단체인 **EPCglobal**에서 제안하여 현재 표준으로 제정된 **EPC(Electronic Product Code)** 네트워크는 모든 객체 및 사물에 태그를 부착하여 객체 추적 및 조회, 상품 이동 조회 등을 할 수 있도록 설계되었다. **RFID** 사용자는 **RFID 태그**가 부착된 객체를 읽어서 **EPC 네트워크**를 통하여 객체와 관련된 상세한 정보를 얻을 수 있다[4]. 한편, **EPCglobal**에서는 **RFID 미들웨어**의 최근 스펙으로 **ALE(Application Level Events)**를 제안하였으나 기본적으로 **EPC 코드** 데이터만을 수용하고 있다.

한편, 국내에서 **EPC 코드**를 사용하기 위해서는 **EPCglobal** 단체에 가입하여 **EPC 코드**를 할당받아야 한다. 이럴 경우 회원사 당 약 1억 원의 연회비를 납부해야 할 뿐만 아니라 추가적으로 많은 비용과 시간이 소요된다[5]. 한편 한국인터넷진흥원(이하, NIDA)에서 국내 **RFID** 산업 활성화 및 **RFID** 서비스 간 상호 운용 제고를 위하여 **ISO/IEC 15459** 국제 표준을 준수하는 **KKR 코드** 체계를 제안하였다[6]. 그러나 현존 **ALE 미들웨어**는 **ISO/IEC 15459 KKR 코드** 체계 처리를 위한 방안이 없으므로 국내 **RFID** 연구 개발 주체는 여전히 **EPC 코드**를 사용하거나 자체적으로 개발한 코드 체계를 사용해야 하

는 등 **RFID** 연구의 영속성과 네트워크화를 하는 데 어려움이 예상된다[5, 6]. 따라서 본 논문에서는 **ALE** 기반 **RFID** 미들웨어에서 한국인터넷진흥원에서 제안한 **KKR 코드** 데이터를 효율적으로 처리할 수 있는 방법을 제안한다. 특히 **ISO/IEC 15459 KKR 코드**를 **ALE** 미들웨어에서 처리하기 위한 데이터 변환 방법에 대하여 제안하고 실제 **ALE** 미들웨어에 적용해 본다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로서 **EPCglobal**의 **EPC** 네트워크, **ALE** 미들웨어, 그리고 **ISO/IEC 15459 KKR 코드** 체계에 대하여 설명한다. 3장에서는 제안하는 **KKR** 변환 방법에 대하여 설명하고, 4장에서는 제안하는 방법을 구현하고 테스트한 결과에 대하여 논의한다. 마지막으로 5장에서 본 연구에 대한 결론을 맺는다.

II. 관련 연구

1. EPCglobal의 EPC 네트워크

EPCglobal은 상품 코드의 국제 표준 개발/관리 기구인 **EAN**과 **UCC**의 통합으로 탄생된 **GS1**이 2003년 11월 설립한 자회사로서 **EPC** 코드와 **EPCglobal** 네트워크의 보급을 총괄하고 있는 기구이다. **EPC** 네트워크는 유통 및 물류 망에서 상품을 자동 인식하고 정보를 공유하는 목적을 가진 국제 기술 표준 네트워크로서 특정 개체의 식별과 데이터 저장, 전달 방법의 정의 등을 목적으로 한다.

EPC 네트워크는 **ID System**(**EPC** 태그와 리더 장치), **EPC**, **ALE**, **ONS(Object Name Service)**, **EPC IS(EPC Information Service)**와 같은 구성요소를 갖는다. 기본적으로 유일한 개체를 식별 가능하도록 하는 태그인 **EPC**가 **RFID** 리더 장치에 의해서 읽혀지면 **RFID** 미들웨어는 수많은 데이터에 의한 **EPC** 네트워크의 트래픽을 감소시키기 위하여 응용이 원하는 정보만 필터링하여 **ONS**에게 보낸다. **ONS**은 사용자로 하여금 **EPC** 데이터 외에 더 많은 정보를 얻을 수 있는 인터넷 주소를 제공한다[7, 8].

2. ALE 미들웨어

사실상 국제 표준 단체인 **EPCglobal**에서 표준으로 제안한 **RFID** 미들웨어의 스펙 문서인 **ALE**는 **EPC** 미들웨

어에 대한 구체적인 구현 방법 및 특정 소프트웨어 모듈 내에서의 인터페이스 등을 기술하지 않고 외부 인터페이스만 정의하고 있다. ALE의 스펙 문서는 데이터를 수집, 필터링, 그리고 그룹핑하여 비즈니스 로직을 해석하는 이벤트를 생성하는 데에 초점을 둔다. 그렇게 함으로써 원시 EPC 데이터를 획득하는 하부 구조 모듈과 그 데이터를 필터링 및 카운팅하는 구조적 모듈, 그리고 데이터를 사용하는 클라이언트 응용 간의 독립성을 제공한다[9].

또한 ALE는 Reader Manager 모듈에서 리더 장치에서 들어오는 RFID 코드 데이터를 PML(Physical Mark-up Language) 문서로 바꿔서 처리를 한다. PML 문서로 바꾸는 과정 중에 원시 EPC 데이터를 URN 코드로 변환해야 한다. 변환된 URN 코드로 ALE 처리 이후에 다시 pure-identify 포맷 또는 다른 포맷으로 변환할 수 있어야 한다[10, 11].

3. ISO/IEC 15459 KKR 코드 체계

ISO/IEC 15459는 ISO/IEC JTC SC31/WG2에서 이동하는 운송 단위에 할당하기 위하여 제안된 코드이다. ISO/IEC 15459 등록기관 RA(Registry Authority)인 NEN (Nederlands Normalisatie instituut)에서 발행 기관 코드인 IAC(Issuer Allocation Code)를 할당받아 IA(Issuing Agency)가 자체적으로 구성할 수 있다. IAC 등록을 위해서는 발급 대리자가 명시한 등록번호 형식 준수하고 코드의 유일성이 보장되어야 한다.

NIDA가 2006년 하반기부터 코드 발급을 시작한 KKR 코드는 ISO/IEC 15459 국제 표준을 준수하는 국가 체계로 표 1과 같이 구성된다. 발급 기관 코드는 ‘National Public Administration’을 위하여 첫 글자를 ‘K’로, 이어 두 글자는 ISO 3166에 정의된 국가코드 두 문자가 오도록 정의하여 총 3문자의 ‘KKR’를 사용하도록 한다.

표 1. KKR 코드 체계
Table 1. KKR code system

구분	발급기관코드 (IAC)	기관코드 (CC: Company Code)	구분자 (Prefix)	객체종류식별코드 (IC: Item Code)	객체단위식별코드 (SC: Serial Code)
문자수	3	3	1	기변	기변
세부 설명	KKR	000~9ZZ		RFU (Reserved for Future Use)	
		A00~ZZZ	다음표 참조	기관별 자체 정의	기관별 자체 정의

기관 코드는 기관을 유일하게 식별할 수 있으며, 표 2와 같이 한 문자로 이루어진 구분자(prefix)는 객체 종류 식별코드의 문자수를 정의한다. 객체단위 식별코드(SC)는 가변 길이를 가지면서, 인코딩 시 코드 전체 길이를 나타내는 Object Length를 통하여 기관별로 자유롭게 SC의 길이가 정해진다[12, 13].

표 2. 구분자에 따른 IC 문자수
Table 2. Character number of IC by prefix

구분자 (Prefix)	0	~	9	A	B	C	D	E	F	G	중략	Z
IC의 문자수	RFU			1	2	3	4	5	6	7	중략	26

III. 제안하는 방법

1. KKR 코드를 위한 URN

ALE 기반 RFID 미들웨어는 RFID 리더에서 들어오는 EPC 코드 데이터를 공통의 URN 형식으로 변환하여 처리한다[11]. 따라서 본 논문에서는 KKR 코드를 URN 형식으로 변환함으로써 ALE 기반 RFID 미들웨어에서 KKR 코드를 효율적으로 처리할 수 있도록 한다. 한편 NIDA에서 제안하고 있는 KKR 코드의 URN 형식은 다음과 같다[13].

urn:ods:iso-iec:15459:1:KKR.AAA.C.ROM.ABCD

그러나 NIDA에서 제안하는 KKR 코드는 ODS (Object Discovery Service)를 위한 URN으로서, ‘KKR.AAA.C.ROM.ABCD’와 같이 알파벳 대문자로 코드 데이터를 표현하므로 기존의 URN과 형식이 다르기 때문에 ALE 미들웨어는 NIDA에서 제시하고 있는 URN 코드를 효과적으로 처리하기에 어려운 점이 있다[9, 13, 14]. 이를 해결하기 위하여 본 논문에서는 아래의 예와 같은 URN 형식을 정의하였다. 이 형식은 NIDA에서 제안하는 URN 형식을 따르면서도 ALE 미들웨어 내에서도 처리할 수 있다.

urn:ods:15459:1:11634.27457.2.130.1083492

URN 코드 부분 중 헤드(head) 부분인 urn:ods: 15459:1

은 URN 코드이면서 ISO/IEC 15459-1 코드 체계를 나타내고 있다. URN 코드 중 몸체(body) 부분에 해당하는 11634.27457.2.130.1083492는 KKR.AAA.C.ROM.ABCD를 십진수로 변환한 것이다. 알파벳 대문자를 십진수로 변환한 이유는 ALE 미들웨어에서 물리적인 코드 데이터를 URN으로 변경할 때 URN의 몸체 부분을 십진수로 처리하기 때문이다[10, 11, 15].

2. KKR 코드 변환 방법

본 논문에서는 ISO/IEC 18000-6C 지원 태그에 KKR 코드가 인코딩되어 있어서, RFID 리더 장치를 이용하여 태그를 읽어 ALE 기반 RFID 미들웨어에 보낼 수 있다는 것을 가정한다. 통상적으로 가장 많이 쓰이는 ISO/IEC 18000-6C 지원 태그의 메모리 구조는 표 3과 같다[14].

표 3. ISO/IEC 18000-6C 지원 태그의 메모리 구조
Table 3. Memory structure of ISO/IEC 18000-6C tag

구성 요소		주소(16진수)	설명
CRC-16		00 ~ 0F	태그의 리더 정보이동 어려 감사
PC	Length	10 ~ 14	UII data Word 길이 (PC + UII data word 길이 -1)
	User Memory Indicator(UMI)	15	User 메모리 사용여부 기록
	XPC Indicator(XI)	16	추가적인 PC 영역 사용여부 기록
	NSI	17	EPC(0), Non-EPC(1) 여부 기록
	AFI	18 ~ 1F	태그 종류 분야 식별 ID값
UII	DSFID	20 ~ 27	UII data 구조 및 ObjectID(OID) 저장 방식 기록
	Precursor	28 ~ 2F	ObjectID & Object의 구조 및 형식 기록
	ObjectID	Length	ObjectID의 Byte 길이 -1
	ObjectID	30 ~	ObjectID 값
	Object	Length	Object의 Byte 길이
	Object	Object	RFID 코드

ISO/IEC 15459 KKR 코드 체계를 사용할 경우 PC의 Toggle 비트가 ‘1’로 설정되고 AFI 코드가 11로 할당된다. 할당된 이후에 RFID 리더 장치는 KKR 코드 데이터를 읽어서 미들웨어로 보낸다.

UII(Unique Item Identifier) DATA는 RFID 코드 데이터를 변환할 때 필요한 부분이다. 통상적으로 ISO 태그 지원 RFID 리더 장치는 미들웨어에게 UII DATA를 보낸다.

대다수 RFID 리더 장치가 미들웨어로 보내는 코드 데이터는 일반적으로 16진수 형태이다. 예를 들어 16진수의 코드 데이터 값이 ‘05 34 09 5A E5 AD 04 44 10 42 21

90’일 경우 코드 데이터 값을 2진수로 일괄 변환한다. ISO/IEC 15459 코드 체계의 DSFID(Data Structure Formatted IDentifier) 값은 2진수로 ‘0000 0101’이다. 리더 장치에서 들어오는 코드 데이터 중 앞의 8비트가 ‘05’이므로, 이 코드 데이터가 ISO/IEC 15459 코드 체계임을 알 수 있다.

다음으로 Precursor 값을 통해서 KKR 코드 체계인지 아닌지를 확인할 수 있다. 5비트 압축을 사용하는 KKR 코드 체계는 ISO 15459 코드 체계 중 15459-1 코드 체계이나 15459-4 코드 체계로 나타날 수 있다. 그러므로 KKR 코드 체계의 Precursor 값은 ‘0011 0001’이나 ‘0011 0100’으로 기록되어야 한다. 리더 장치에서 보낸 데이터 중 Precursor에 해당되는 부분이 16진수로 ‘34’이고 이것을 2진수로 변환하면 ‘0011 0100’이므로 이 코드 데이터는 KKR 코드 체계를 따르고 있다는 것을 확인 가능하다.

Object 요소의 Length는 Object 바이트 수를 2진수로 기술하는데, 예시로 든 코드 데이터의 값이 16진수로 ‘09’이어서 2진수로 변환하면 ‘0000 1001’이므로 Object의 길이가 9 바이트로 총 72비트이다. KKR 코드는 5비트 압축된 아스키 문자로 코드 데이터를 표현하므로 14개의 아스키 문자가 있다. 여기에서 남은 2비트는 아스키 문자가 아니라 단지 ‘0’을 붙인 것이다.

아스키 문자를 십진수로 변환하기 위하여 아스키 문자에 해당되는 Object 값을 각각 5비트씩 잘라서 ‘01’을 붙인다. 그 이유는 아스키 문자가 7비트로 이루어져 있지만 KKR 코드 체계에서는 그 아스키 문자를 5비트 압축을 해서 사용하기 때문이다. 이와 같은 방식으로 2진수에서 KKR 코드 데이터의 바디 부분을 아스키 문자로 변환한다. 변환된 아스키 문자에서 7번째 자리인 Prefix 값을 얻어서, IC 자리수가 몇 자리인지를 알아낸다. 마지막으로 IC 값을 제외한 나머지는 SC이다. 앞서 언급하였듯이 아스키 문자는 ALE 미들웨어에서 처리하기가 어렵기 때문에 KKR 코드 체계의 IAC, CC, Prefix, IC, SC 값을 십진수로 변환하여 URN에 적용한다.

3. KKR 코드 변환 순차 흐름도

위에서 언급하였던 ALE 미들웨어를 위한 KKR 코드 변환 방법을 순차 흐름도로 표현하면 그림 1과 같다. 우선 RFID 리더 장치에서 들어오는 코드 데이터가 EPC 코드인지 ISO/IEC 15459 코드인지를 확인한다. EPC 코드일 경우 기존의 EPC 코드 체계에 따라 코드를 변환한다.

그러나 ISO/IEC 15459 코드 일 경우에는 Precursor 구조를 통하여 KKR 코드 체계인지를 확인한다. 들어오는 RFID 코드 데이터가 KKR 코드 체계라면 3장 2절에서 언급하였던 KKR 코드 변환 방법에 따라 변환한다. 만약에 KKR 코드 체계가 아닐 경우 예외처리가 일어나면서 종료된다.

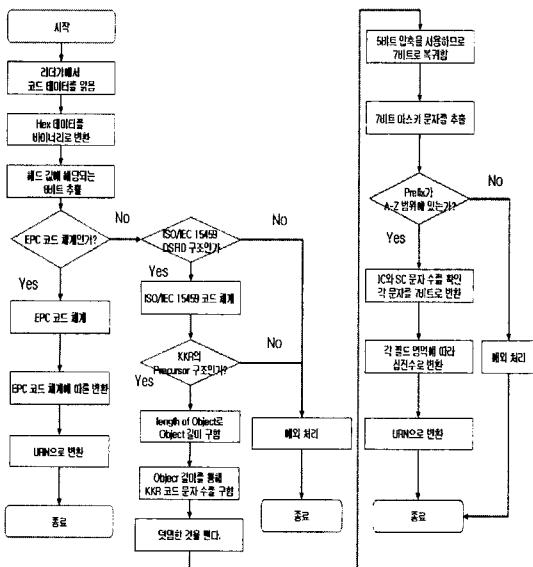


그림 1. KKR 코드 변환 순차 흐름도
Fig. 1. The flowchart for KKR code conversion

4. KKR 코드 변환 클래스 다이어그램

KKR 코드 변환을 위하여 주요 패키지로서 그림 2와 같이 iso 패키지와 util 패키지가 있다.

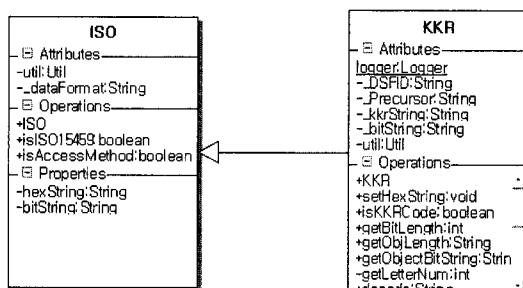


그림 2. iso 패키지의 클래스 다이어그램
Fig. 2. Class diagram of iso package

iso 패키지의 ISO 클래스는 ISO 코드에 해당되는 특성과 ISO/IEC 15459 코드 체계의 특성이 들어 있으며 이를 2진수로 일괄적으로 변경할 수 있는 메소드가 들어 있다. ISO 클래스를 상속받는 KKR 클래스는 KKR 코드를 인코딩 및 디코딩을 할 수 있는 메소드가 들어 있다. 객체 지향 프로그래밍 관점에서 KKR 코드는 ISO 코드 체계를 따르고 있기 때문에 ISO 클래스를 상속 받는다. 그리고 ISO 코드 체계를 의미하는 iso 패키지에 종속되어 있다.

5. KKR 코드 변환 순차 다이어그램

그림 3은 KKR 클래스의 decode 메소드를 호출할 경우 어떤 흐름으로 URN 코드로 변환되는지 보여주는 순차 다이어그램이다. KKR 클래스 안에 URN 코드로 변환하기 위하여 필요한 정보가 들어 있으며, 따라서 다른 클래스를 참조하지 않는다.

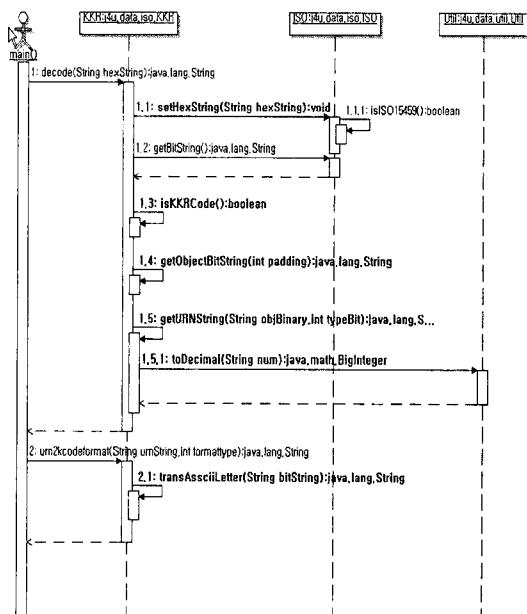


그림 3. KKR 코드 변환의 순차 다이어그램
Fig. 3. The sequence diagram for KKR code conversion

코드 변환 과정은 다음과 같다. 먼저 메인 함수에서 ISO 클래스를 상속받는 KKR 클래스의 decode 메소드를 호출하면 KKR 클래스는 16진수로 들어오는 데이터

를 2진수로 바꾼다. 그러면 ISO 클래스는 `isISO15459` 메소드를 호출하여 ISO 코드 체계인가를 확인한다. 그 이후에 KKR 클래스는 다른 클래스에 의존적이지 않고 내부 메소드를 사용하여 코드를 변환한다. 그리고 `urn2kcodedeformat` 메소드를 호출하면 메소드는 인자 값에 따라서 다양한 URN 코드로 변환된다.

IV. 구현 및 실험

그림 4는 KKR 클래스의 `getURNString` 메소드 코드의 일부분이다. `getURNString` 메소드는 KKR 코드를 URN 코드로 변환하는데 중요한 역할을 한다. 즉, 2진수로 된 코드를 십진수로 변환하는 것이다. 알파벳 대문자에 해당되는 KKR 코드를 십진수로 변환하기 위해서는 알파벳 대문자에 해당되는 아스키 문자의 16진수를 알아야 하고 그것을 2진수로 변환 다음에는 2진수를 5비트 압축을 한다.

```

String nationdigit = util.toDecimal(
    objBinary.substring(0, typeBit*3)).toString(); //국번은: IC 문자 수 불변
String officaldigit = util.toDecimal(
    objBinary.substring(typeBit*3, typeBit*6)).toString(); //기관코드: SC 문자 수 불변
String prefixdigit = util.toDecimal(
    objBinary.substring(typeBit*6, typeBit*7)).toString(); //prefix : IC 문자 수 불변

int companydigit = typeBit*7+Integer.parseInt(prefixdigit)*typeBit; //IC와 숨 문자 개수 합
String companystr = util.toDecimal(
    objBinary.substring(typeBit*7, companydigit)).toString(); //Company Code : IC 문자 수 기반
String Serialstr = util.toDecimal(
    objBinary.substring(companydigit, objBinary.length())).toString(); //Item Code : ABCD 문자 수 기반

```

그림 4 `getURNString` 메소드의 일부분
Fig. 4. A part of `getURNString` method

일반적으로 KKR 코드는 앞의 세 자리인 KKR과 기관 코드, 구분자는 문자수가 불변이기 때문에 연속적으로 이어지는 2진수를 잘라내기는 쉽다. IC와 SC의 문자수는 구분자에 의해서 가변적으로 변한다. 구분자를 통하여 IC의 문자수를 알아내고 SC의 문자수는 IC를 제외한 나머지에 해당된다.

```

0534095AE5AD044410422190
[2008-03-07 15:05:09] - ISO/IEC 15459-4
[2008-03-07 15:05:09] - KKR code 맞습니다.
[2008-03-07 15:05:09] - 압축 사용 :5 bits
[2008-03-07 15:05:09] - padding Length :2 bits
[2008-03-07 15:05:09] - Object Length :70 bits
Result :urn:ods:15459:1:11634.27457.2.130.1083492
0534095AE4318C94A483AD68
[2008-03-07 15:05:09] - ISO/IEC 15459-4
[2008-03-07 15:05:09] - KKR code 맞습니다.
[2008-03-07 15:05:09] - 압축 사용 :5 bits
[2008-03-07 15:05:09] - padding Length :2 bits
[2008-03-07 15:05:09] - Object Length :70 bits
Result :urn:ods:15459:1:11634.3171.4.676417.27482
0534095AE410802110863210
[2008-03-07 15:05:09] - ISO/IEC 15459-4
[2008-03-07 15:05:09] - KKR code 맞습니다.
[2008-03-07 15:05:09] - 압축 사용 :5 bits
[2008-03-07 15:05:09] - padding Length :2 bits
[2008-03-07 15:05:09] - Object Length :70 bits
Result :urn:ods:15459:1:11634.1058.1.1.69907524
0534095AE4119058C0443214
[2008-03-07 15:05:09] - ISO/IEC 15459-4
[2008-03-07 15:05:09] - KKR code 맞습니다.
[2008-03-07 15:05:09] - 압축 사용 :5 bits
[2008-03-07 15:05:09] - padding Length :2 bits
[2008-03-07 15:05:09] - Object Length :70 bits
Result :urn:ods:15459:1:11634.1124.2.792.1117317
0534095AE4110C2108864298
[2008-03-07 15:05:09] - ISO/IEC 15459-4
[2008-03-07 15:05:09] - KKR code 맞습니다.
[2008-03-07 15:05:09] - 압축 사용 :5 bits
[2008-03-07 15:05:09] - padding Length :2 bits
[2008-03-07 15:05:09] - Object Length :70 bits
Result :urn:ods:15459:1:11634.1091.1.1.35754150

```

그림 5. KKR 코드 변환 모습
Fig. 5. A snap shot of code conversion

그림 5는 RFID 리더 장치에서 받은 ISO/IEC 15459 KKR 코드 데이터 값을 본 논문에서 제안하는 URN 코드로 변환 과정이다. 그림 5의 로그 정보를 보면 순차 흐름도 대로 KKR 코드가 URN으로 변환되는 과정을 볼 수 있다. 총 다섯 개의 KKR 코드 데이터가 순차적으로 URN 코드로 변경되어 출력하였다. **KKR.ZZA.B.DB.AABCD** 형식으로 인코딩된 KKR 코드 데이터를 입력 받아 다양한 URN 형식으로 변환된 결과이다. 16진수로 들어온 KKR 코드 데이터를 본 논문이 제안하는 URN 코드와 코드 변환 방법을 통하여 ALE 미들웨어가 처리한 이후, 응용으로 보내기 전에 다양한 URN 코드로 나타낼 수 있다. 그림 6은 제안한 방법을 이용하여 코드를 변환한 예이다.

```

[2007-10-13 16:19:59] - NIDA URN :urn:ods:iso-iec:15459:1:KKR.ZZA.B.DB.AABCD
[2007-10-13 16:19:59] - URN for ALE Middleware :urn:ods:15459:1:11634.27457.2.130.1083492
[2007-10-13 16:19:59] - EPC RawDecimal :urn:kcode:raw:96.x0534095AE5AD044410422190
[2007-10-13 16:19:59] - EPC RawHex :urn:kcode:raw:96.161033369781887417408692624

```

그림 6. URN 코드 생성의 예
Fig. 6. An example of URN code generation

KKR 코드 변환에 대한 각 결과 값을 다음과 같이 표 4로 정리하였다. 표 4를 살펴보면 발급기관 코드인 KKR

에 해당되는 부분이 십진수인 11634로 똑같다는 것을 알 수 있다. 그러나 기관 코드에 해당되는 부분은 각각 다르고, 구분자에 해당되는 부분은 한 자리 십진수로 변경되었다. 이후의 값은 IC와 SC에 해당된다.

표 4. 코드 변환된 URN 결과
Table 4. The results of KKR code conversion

KKR 코드	변환된 URN 코드
0534095AE5AD044410422190	urn:ods:15459:1:11634.2745 7.2.130.1083492
0534095AE4318C94A483AD68	urn:ods:15459:1:11634.3171. 4.676417.27482
0534095AE410882110863210	urn:ods:15459:1:11634.1058. 1.1.69307524
0534095AE4119058C0443214	urn:ods:15459:1:11634.1124. 2.792.1117317
0534095AE4110C2108864298	urn:ods:15459:1:11634.1091. 1.1.35754150

V. 결 론

본 논문에서는 ALE 미들웨어에서 KKR 코드를 처리하기 위한 방법에 대하여 살펴보았다. 특히 KKR 코드를 URN 코드로 변환하기 위한 방법을 제안하고 이를 구현하였다. 또한 ALE 미들웨어에 적용하기 위한 URN 코드 형식을 제안하였고, KKR 코드 변환을 위한 구체적인 방법에 대해서 살펴보았다. KKR 코드 변환 방법을 구현하고 ALE 미들웨어에 적용함으로써 다양한 형태의 URN 코드를 생성할 수 있음을 확인할 수 있었다.

제안하는 방법을 이용할 경우 기존 RFID 네트워크의 다양한 시스템 간 네트워크화 및 영속성을 극대화할 수 있으며, 저비용의 KKR 코드를 ALE 미들웨어에 처리할 수 있도록 지원할 수 있다. 이와 함께 RFID 네트워크 구축 시 서로 다른 RFID 코드를 이용하더라도 시스템 간의 연계 및 상호운용성을 극대화 할 수 있다.

참고문헌

- [1] 노영식, 변영철, “상황인식기반 지능형 홈 서비스에 관한 연구”, 한국해양정보통신학회 논문지, 제11권, 4호, 2007. 4.
- [2] 최일선, 정희경, “XML 및 모바일 RFID 기반의 문화재 안내 시스템”, 한국멀티미디어학회지, 제10권, 제1호, pp.41-46, 2006.
- [3] Taesu Cheong, Youngil Kim, Yongjoon Lee, “REMS and RBPTS: ALE-compliant RFID Middleware Software Platform”. ICACT 2006 Vol 1. pp.699-704. 2006.2.
- [4] EPCglobal Inc., EPCglobal Architecture Framework Final Version, 1 July. 2005.
- [5] 김경호, 정한영, 이상훈, “국방 RFID 태그 코드 선정 및 ONS 구축방안”, 정보과학회지 제25권 제9호, 2007. 9
- [6] 한국 인터넷 진흥원, “RFID 코드 인코딩 지침서 V1.0”, 2006.10
- [7] 한국 인터넷 진흥원, “국내외 RFID 정책 및 기술 동향 보고서”, 2006. 10.
- [8] Jieun Song, Howon Kim, “The RFID Middleware System Supporting Context-Aware Access Control Service”, 2006 ICACT, Vol.1 pp.863-866, 2006.
- [9] 홍연미, 변영철, “ALE 기반 RFID 미들웨어 시스템 설계 및 구현”, 한국해양정보통신학회 논문지, 제11권, 제4호, 2007. 4.
- [10] EPCglobal Inc., The Application Level Events Specification, Version 1.0, 2005. 9.
- [11] EPCglobal Inc., EPCglobal Tag Data Translation 1.0, 2006.
- [12] ISO/IEC15459-1, “Information technology - Unique identifiers”, 2005. 12
- [13] 한국 인터넷 진흥원, “RFID 검색 시스템 구축 및 운영 지침서 V1.2”, 2006.12
- [14] 한국 인터넷 진흥원, “사례제시를 통한 RFID 적용 본사, 지사간 자산출입관리 시스템 구축 가이드”, 2006.12
- [15] EPCglobal Inc., EPCglobal Architecture Framework Final Version, 1 July. 2005.

저자소개



변지웅(Ji-Yoong Byun)

2005년 제주대학교 영어영문과 학사
2007년 ~ 현 제주대학교 컴퓨터공학
과 석사과정

※ 관심분야 : RFID 미들웨어, 시맨틱 웹, 유비쿼터스
미들웨어, 온톨로지



변영철(Yung-Cheol Byun)

2001년 연세대학교 컴퓨터공학과
박사
2001년 한국전자통신연구원 선임
연구원

2002년 ~ 현 제주대학교 컴퓨터공학전공 교수

※ 관심분야 : 패턴인식, 시맨틱 웹, 지능형 컴퓨팅, 유
비쿼터스 미들웨어



이동철(Dong-Cheol Lee)

2002년 성균관대학교 산업공학과
박사
2003년 ~ 현 제주대학교 경영정보학
과 조교수

※ 관심분야 : Agent, EC, MIS 응용