

Gen2 기반의 다중리더기 운영모듈에 관한 연구

박상현* · 한수* · 신승호*

*인천대학교 컴퓨터공학과

A Study on Multi-Reader Management module Based On Gen2

Sang-Hyun Park* · Soo Han* · Seung-Ho Shin*

*Department of Computer Science & Engineering, Incheon University

Abstract

The ubiquitous, which is developed for human's convenience, uses the concept of situation awareness that communicates with computer and performs a certain work without user's recognition. In order to collect the information of situation, many devices are required. Also, in the condition of ubiquitous middleware, priority is given to effective controls of various devices. There are many devices for collecting the information of situation awareness such as sensor, RFID, etc. Among them, in the use of RFID, the researcher performed the experiment, in which multiple readers were used depending on the necessity of awareness information, and found the problem of intervention between readers occurring when multiple readers are used.

The paper handles the problem of intervention causing from using multiple readers and suggests middleware design module using session manager to solve the problem.

Keywords : RFID, Gen2, Middleware, Multi-Reader Management

1. 서론

인터넷은 사용자의 요구를 위해 새로운 개념의 설계를 필요로 하며 지금까지와는 다른 통신 체계를 요구한다. 이를 위해 미래인터넷이 대두되고 있다. 미국, 유럽, 일본에서는 2007년부터 연구를 시작하였고, 3년 이내에 새로운 모습의 통신 체계를 구축하는 연구를 진행하고 있다. 미래 인터넷은 여러 가지 요구조건을 필요로 하고 있고 그중 Scalability는 수백억~수천억 개의 노드를 효과적으로 지원 할 수 있어야 하며, Ubiquity는 언제, 어디서든 유무선 환경을 통해 접근이 가능해야 하고, Context-Awareness는 사용자의 환경, 경험,

특성을 고려한 서비스를 제공할 수 있어야 한다[1]. 이 세 가지 요구조건은 유비쿼터스 미들웨어에서도 동일하게 만족해야 한다.

유비쿼터스 구현을 위한 핵심 기술로 USN (Ubiquitous Sensor Network)과 RFID (Radio Frequency Identification)를 가장 주목하고 있으며, 이를 위한 여러 가지 미들웨어들도 설계되고 있다. 유비쿼터스의 핵심은 여러 장비들을 이용하여 사용자에게 편리한 환경을 만드는 것이다. 단적인 예로 물품창고에서 하나의 RFID 리더기를 이용하면 주파수 거리제한의 특성으로 많은 장비들의 인식이 불가능하므로, 창고의 크기나 업무 범위에 따라 2개 이상의 RFID 리더기를 사용하게 된다.

† 본 연구는 2007년도 인천대학교 학술연구 조성비 지원에 의하여 수행되었음.

† 교신저자: 신승호, 서울시 서대문구 남가좌동 173-18

M · P: 011-9055-8492, E-mail: shin0354@incheon.ac.kr

2008년 4월 접수; 2008년 6월 수정본 접수; 2008년 6월 게재확정

이를 위하여 소규모 환경에서 현재 사용되고 있는 RFID 900MHz의 표준인 Gen2를 이용한 실험을 진행하였고 여러 개의 RFID 리더기를 이용하면 리더기간 간섭이 발생하는 문제점을 발견하였다.

본 연구에서는 다중 리더기를 사용하는 경우 리더기간 간섭에 의하여 태그사 인식이 안되는 문제를 해결하기 위하여 태그 세션관리를 이용해 태그 간 충돌을 해결하는 방안을 제시한다.

본 논문의 구성은 다음과 같다. 2장에서 Gen2, 미들웨어의 요건과 FDMA(Frequency Division Multiple Access)와 리더와 태그의 통신 방법을 살펴보고 3장에서 다중 리더기를 이용한 실험과 문제점을 거론하고 4장에서 이를 해결하기 위한 모듈설계 방안을 시뮬레이션으로 진행하고 5장에서 결론을 맺는다.

2. 관련연구

2.1 Gen2

EPCglobal은 비영리 기구로서 EPC(Electronic Product Code) 코드의 보급과 EPC 시스템의 표준화, 코드관리 등을 담당하고 있다. EPCglobal에서는 상품 한 개 한 개에 96bit나 128bit등의 EPC code를 붙여 그 상품에 관한 생산정보나 유통 이력 등을 인터넷을 통해 알 수 있도록

하는 것을 목표로 하는데, 96bit의 경우 1,6000만 제조자 번호와 2억 7천만 상품번호 및 687억 개의 상품 시리얼 번호가 부여될 수 있도록 하고 있다. EPCglobal에서는 태그의 종류에 따라 Class 0부터 Class 5까지 분류하며, 현재 Class 0와 Class1 에 대한 규격이 완료되어 있다.

2004년부터 이뤄진 EPCglobal의 Gen2 표준은 과거 EPC 표준의 단점을 보완하고, ISO 규격과의 통합을 고려한 진보된 표준이라고 할 수 있다.

2005년도에 ISO/IEC는 ISO 18000-6의 수정작업을 하면서 Gen2를 ISO 18000-6 Type C로 국제표준화 하였으며, 향후 UHF RFID 시스템은 EPC CIG2 표준을 따를 것으로 기대되고 있어 더욱 관심을 모으고 있다[2].

Gen2의 기원은 2003년 6월 EPCglobal 내에 HAG (Hardware Working Group)를 설립하면서 시작되었으며, Gen2의 목표는 Gen1 (Class 0 및 Class 1)에서 문제 되었던 다양한 기술적인 문제를 해결하고 Gen1에 비해 우수한 성능의 규격 제정과 전 세계적으로 공통 표준을 목표로 다양한 RFID 업체의 의견을 받아들여 규격 제정을 시작하였다. 초기에는 고성능 태그를 목표로 할지, 저가의 태그를 목표로 할지 논의가 이루어졌으나, 반도체 기술의 발달 및 대량생산으로 고성능을 목표로 하더라도 태그당 5센트라는 목표를 맞출 수 있을 것이라는 합의에 도달해 태그의 고성능을 목표로 추진하게 되었다.

<표 1> Gen2 규격과 기존 규격과의 비교

Parameter	Type A	Type B	Gen 2
순방향 링크 (리더-태그) 인코딩	PIE	Manchester	PIE (Nominal 90%)
Modulation Index	27% ~ 100%	18% or 100%	80% ~ 100%
데이터 전송률	33 Kbps(평균) Tari=20usec	10/40 Kbps (국가나 지역의 규격에 따름)	128~26.7 Kbps Tari:6.25~25usec Data-1:1.5~2Tari
역방향링크 (태그-리더) 인코딩	FM0	FM0	FM0 혹은 Miller modulated subcarrier
Collision arbitration	ALOHA	Binary Tree	ALOHA
태그 unique identifier	64 bit (40 bit SUID)	64 bits	Protocol 상에서는 최대 256bits
순방향 링크 오류 검지	모든 명령에 대한 5/16 bit CRC	16 bit CRC	16 bit CRC
역방향 링크 오류 검지	16bit CRC	16bit CRC	16bit CRC
Collision arbitration 선형성	250 태그 이하	2 ²⁵⁸ 태그 이하	32767 태그 이하
Kill 명령	없음	없음	있음
보안 고려	없음	없음	RN-16을 이용encryption 32bit Access와 Kill password
Multi Session	없음	없음	4개의 inventory flag와 selected flag
CMD 길이	6bit	8bit	2-16 bit (사용 빈도 고려)
메모리 Addressing	256bit 이하의 blocks	Byte blocks 1,2,3,4 바이트 쓰기	16bit 단위로 Read/write Memory bank 전체를 한번에 read 가능

EPC Gen2 규격의 주요 특징 중 하나는 다른 규격에 비하여 다양한 세부 규격 및 선택 규격을 가지고 있다는 것이다. 예를 들어 모뎀방식 하나만 보더라도, DSB-ASK, SSB-ASK, PR-ASK의 3가지 방식의 모뎀을 규격에 포함하고 있다. 따라서 태그의 경우 이러한 모든 모뎀을 지원하여야 하나, 리더의 경우 Gen2를 선택적으로 만족하는 다양한 종류의 리더가 존재할 수 있게 된다. 리더 제조업자는 규격 중 자신에게 유리한 다양한 종류의 리더를 만들 수 있다.

Gen2 규격 중 대표적인 특징 중의 하나가 Dense 모드를 적용한 것이다. Dense 환경이란 리더의 송신 출력이 인접리더에 영향을 주지 않을 정도까지 감소되는 영역(보통 1km) 내에 채널 수 만큼 리더가 있는 경우를 의미한다. 예를 들어 우리나라 주파수의 경우 27개 채널이 있으므로 1km 반경 내에 27개의 리더가 있으면 Dense 환경이라고 생각할 수 있다. 이 경우 주파수 호핑(Frequency Hopping)이나 LBT를 사용하더라도 동일한 채널을 사용하는 리더가 존재할 확률이 높으며, 인접 리더의 신호가 태그 응답신호보다 큰 경우 태그 응답신호에 간섭을 줄 수 있다. 따라서 Gen2에서는 리더의 송신 대역과 태그의 응답대역을 분리하여 간섭을 최대한 억제하기 위한 Dense 모드 규격을 설정하였다.

Dense모드란 리더의 점유주파수 대역과 태그의 점유 주파수 대역을 주파수 채널 측면에서 분리 (FDD : Frequency Division Duplexing) 하도록 주파수 점유 스펙트럼을 엄격하게 규정하는 것이다. 이렇게 되면 태그와 리더간의 주파수 채널이 정확히 분리되므로, 태그는 태그끼리, 리더는 리더끼리 주파수 간섭은 발생할 수 있으나, 태그와 리더 사이의 주파수 간섭은 억제할 수 있다. Gen2규격을 기존 규격과 비교한 결과는 <표 1> 과 같다.

2.2 유비쿼터스 컴퓨팅 환경을 위한 미들웨어

유비쿼터스 컴퓨팅 환경에서 미들웨어의 역할은 생활공간에 설치된 센서들과 다양한 종류의 기기들의 정보를 여러 가지 네트워크의 종류를 통해서 얻은 후, 추론, 로직 들의 기능을 이용하여 고차원 지식으로 변환시켜 사용자에게 필요한 서비스를 제공해 주는 환경을 구축하는데 있다.

2.2.1 유비쿼터스 컴퓨팅 미들웨어의 조건

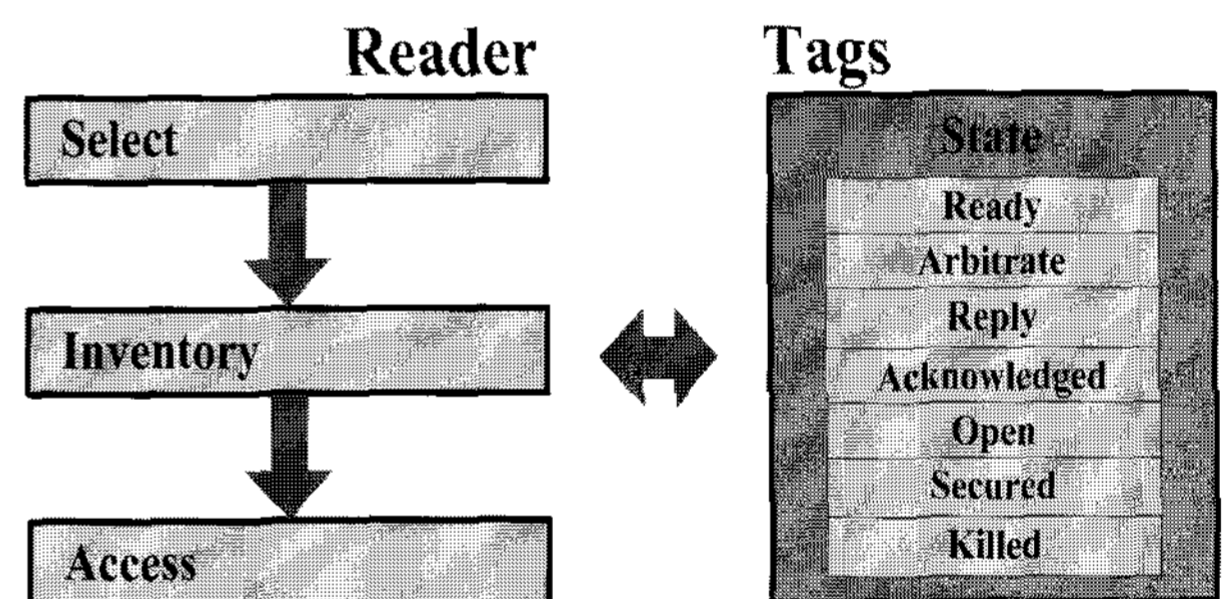
미들웨어를 설계하기 위해서는 센서 네트워크 미들웨어, 자치적 분산 시스템 및 상황 인식의 요구 사항 등 기존의 기술과 부과적인 요구 사항을 만족해야 한다[3].

- 효율적인 에너지 및 리소스 관리
- 하드웨어와 네트워크 등의 저 수준 층의 추상화된 인터페이스와 적절한 프로그래밍 모델을 통해 개발 용이성 향상
- 복제를 통해 fault tolerance를 향상
- 여러 응용프로그램들의 동시 실행관리
- 다양한 기기들과 서비스의 효율적인 발견
- 서비스 품질 관리
- 수집된 센서나 개인 정보 등을 수집하고 생활공간에 필요한 모델을 기반으로 고차원 지식 추론
- 개인 정보의 적절한 관리와 보호

이런 요구를 만족시키는 미들웨어의 디자인은 유비쿼터스 컴퓨팅 환경의 결정적인 요소이며 부분적인 요구를 만족 시키려는 많은 노력이 있었다.

2.3 리더와 태그의 통신 프로토콜

Gen2 프로토콜에서 리더와 태그가 통신하는 단계는 <그림 1> 과 같이 나타낼 수 있다.



<그림 1> 리더와 태그의 통신 단계와 상태

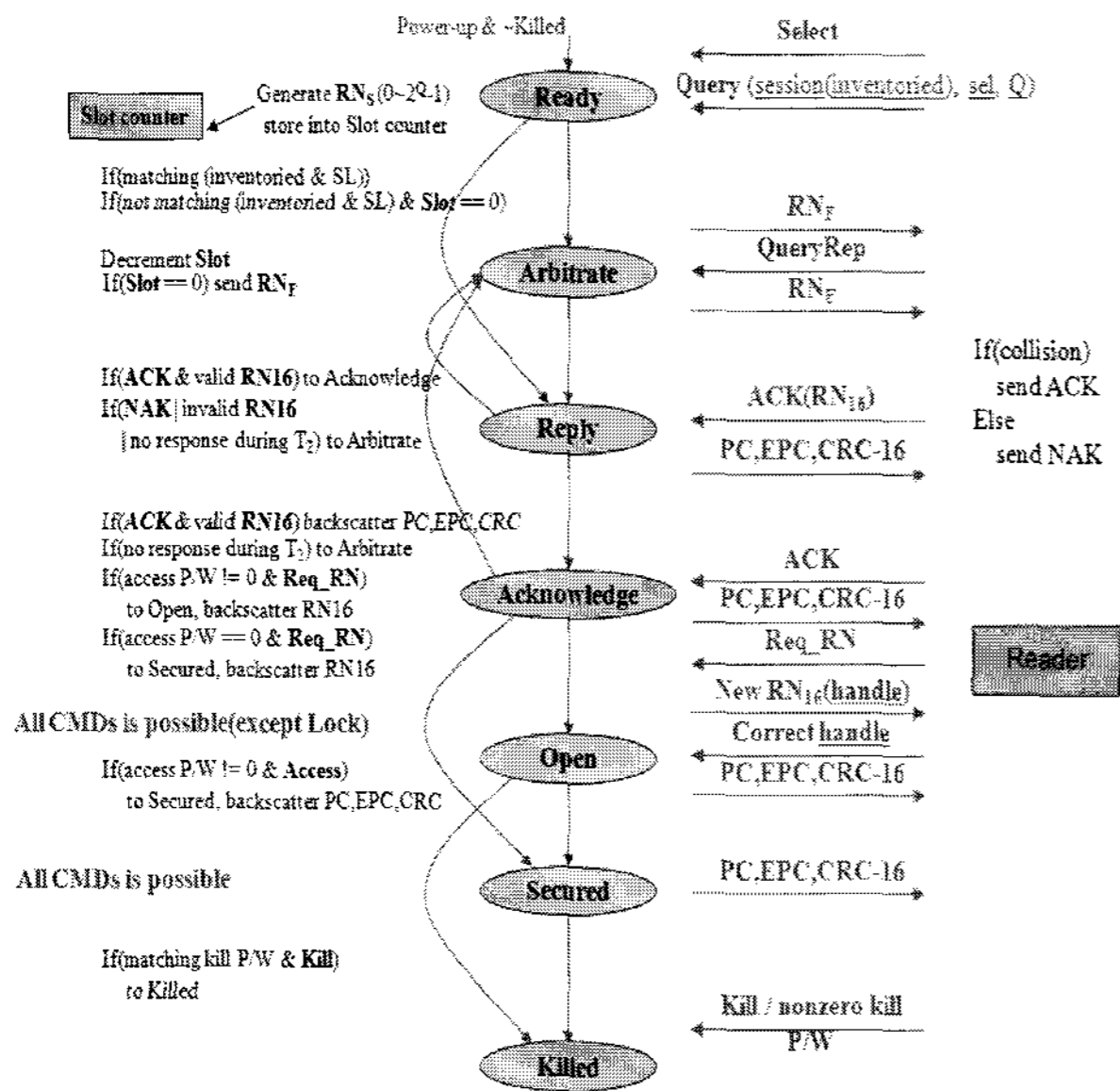
리더에는 크게 3가지 명령어 형태로 구분해 볼 수 있다. 우선 태그를 선택하기 위한 Select 명령, 선택한 태그를 인식하는 Inventory, 선택된 하나의 태그에 접근하는 Access 명령으로 구분할 수 있다. 좀 더 구체적으로 설명하면 Select 명령은 사용자가 검색하고자 하는 조건에 해당하는 태그 집단을 선택하는 명령이고, Inventory 명령은 선택된 태그들을 모두 인식하기 위해서 충돌방지 알고리즘을 수행하여 하나의 태그씩 선택하여 태그에 읽고 쓰는 작업을 할 수 있는 사전 작업을 한다. 이 과정은 태그를 인식하는 과정이다[4].

Access 명령은 선택된 하나의 태그 메모리에 읽기/쓰기, Lock, Kill 등의 작업들을 제어 하고 수행 한다.

태그들은 Access 전에 유일하게 식별되어야 하고 Access는 다양한 명령을 포함하고 있는데, 그 명령어

들은 Req_RN, Read, Write, Kill, Lock 등이 있다. 리더에서 태그로 가는 링크에서는 One-Time Pad 기반의 Cover-Coding 방법이 사용된다.

Cover-Code란 Ciphertext라 한다. 리더가 우선 태그에게 RN16을 요구하고 수신한 RN16과 패스워드를 XOR하여 태그에게 전송하면 태그는 리더에게 전송한 RN16으로 XOR 연산을 하고 자신의 패스워드와 비교하여 처리한다.



<그림 2> 태그 상태 다이어그램

태그의 상태는 <그림 2>와 같이 총 7개로 구성되며 Ready, Arbitrate, Reply, Acknowledged, Open, Secured, Killed 의 상태로 나누어진다.

충돌방지 알고리즘이 수행되는 Inventory 명령을 살펴보면, 우선 Inventory 초기에 선택한 태그집단의 태그들을 인식하는 작업을 하는데, 그 과정에서 네 개의 세션 중에서 하나의 세션을 선택하고 리더의 Query 명령 전송에 의해 Inventory 라운드가 시작된다. 그리고 한 개 이상의 태그가 응답을 하면 리더는 태그에게 PC, EPC, CRC-16을 요청한다. Inventory 작업은 다양한 명령을 포함하고 있는데 Query, QueryAdjust, QueryRep, ACK, NAK 등이 있다. 특히 Inventory 라운드에서 중요한 점은 한번에 한 세션만 작업한다는 점이다.

리더와 태그가 통신할 때 <그림 2> 와 같은 프로세스로 통신을 한다. 태그의 상태가 Ready 일 때 리더는 태그를 인식하기 위하여 Select 명령을 보내고 Query 명령을 보내는데, 이때 Query 명령에 인자로 session (inventoried flag), sel 그리고 Q 값을 함께 보낸다. 그러면 태그는 Q값을 기반으로 자신이 통신할 슬롯을 알게 되고, 해당 세션으로만 통신을 시작하게 된다. Q값

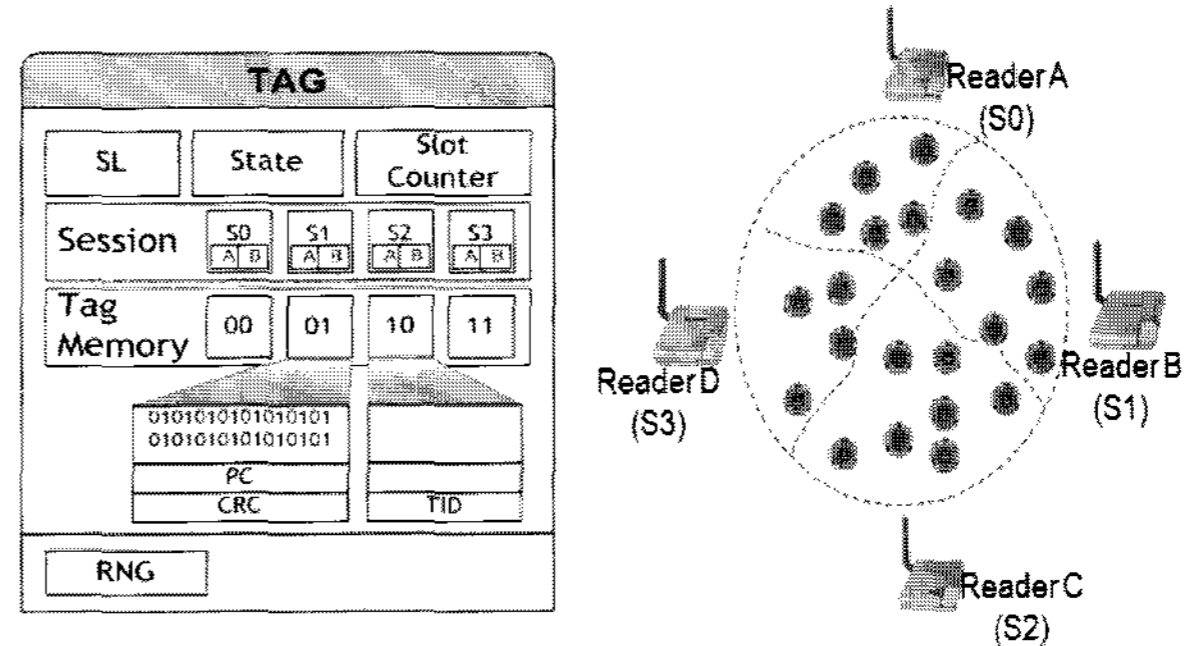
과 session (inventoried flag)는 충돌방지를 위한 알고리즘에 필요한 인자 값들이다.

다중 리더 운영에 대한 연구에서 가장 초점이 되는 부분이 바로 Q값을 구하는 알고리즘 이다.

세션의 경우 연결 되는 리더에 기본적으로 랜덤으로 부여 하는데, 이 경우 세션과 리더 간의 매칭 정보를 유지할 수 가 없다. 그래서 본 연구에서는 Q값 알고리즘은 배제하고 리더에 세션을 부여할 때 세션 정보를 관리하기 위하여 세션 관리자 모듈을 설계 및 구현한다.

2.4 Session & Inventoried Flag

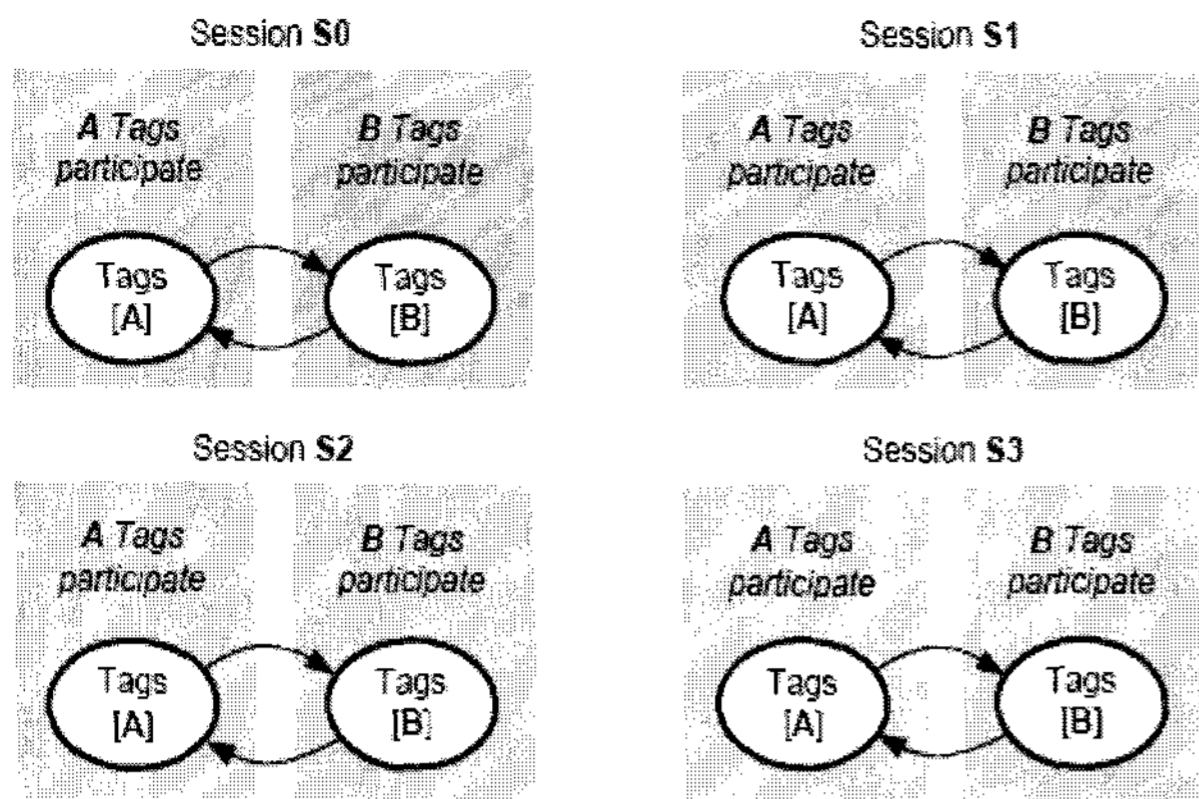
Gen2 프로토콜에서는 다중 리더 운영 시 여러 개의 리더가 모든 태그의 인식을 위하여 Session과 Inventoried Flag 를 제공한다.



<그림 3> 태그의 세션과 세션을 이용한 태그 분할

단일 리더가 인식할 수 있는 태그의 수가 제한적이기 때문에 다중 리더 운영 시 각 리더가 모든 태그를 인식하기 위하여 태그를 분할한다. Gen2 프로토콜에서는 4개의 세션을 지원하며, 태그에는 4개의 S0, S1, S2 그리고 S3로 표현되는 세션을 제공 한다. <그림 3>은 태그에서의 세션의 구조와 각 리더에 세션이 부여 되었을 때 분할 된 태그들을 인식할 수 있는 구조를 보여주고 있다[5].

태그는 Inventory 라운드에서 하나의 세션만 접근이 가능하다. 두 개 이상의 리더는 공통의 태그 집단을 독립적으로 Inventory 하기 위해 세션을 사용할 수 있으며, 태그는 각 세션마다 독립적으로 Inventory Flag를 포함한다. Inventory Flag는 각각 두 개의 값을 가지는데 각자 A, B로 표현 된다. Inventory 라운드를 시작할 때 리더는 한 세션의 Inventoried Flag값 중 하나를 선택하는데, 일단 태그가 하나의 세션 Inventory 라운드에 참가하면 다른 세션 Inventoried Flag의 사용 및 수정은 불가능하다. Inventoried Flag는 단지 태그를 구분하여 제공하기 위한 자원이며, 태그의 Inventoried Flag를 제외한 나머지 자원은 세션들 사이에서 공유된다.

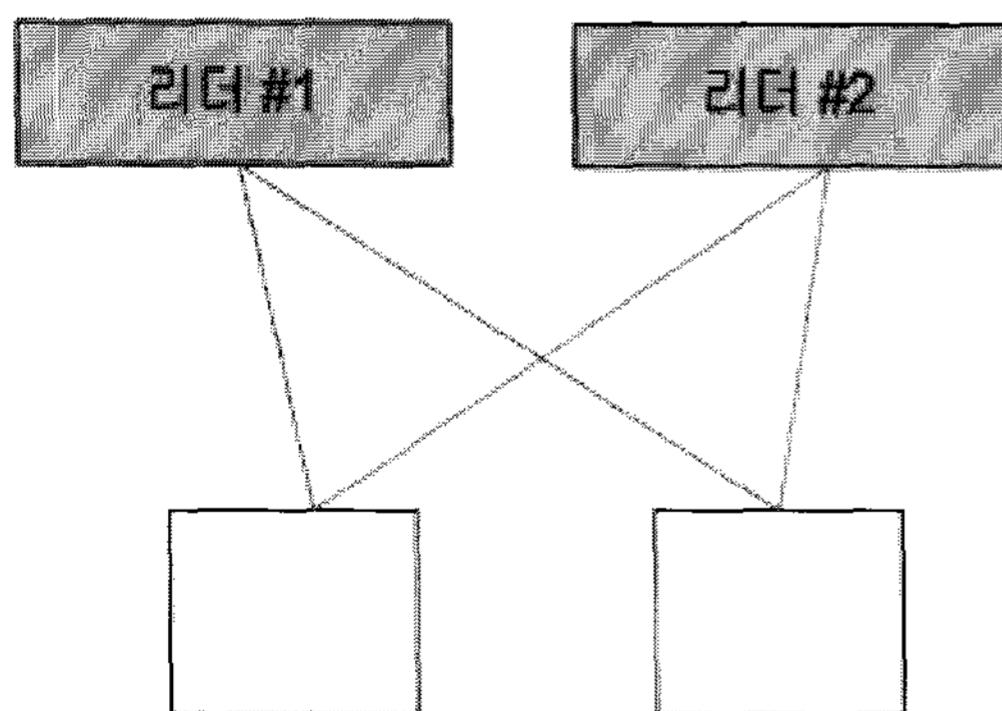


<그림 4> 세션 상태 다이어그램

<그림 4>는 세션의 상태 다이어그램을 나타낸다. 리더는 유일한 태그를 인식하면 현재의 태그 세션에 해당하는 Inventoried Flag의 값을 역으로 변경한다.

세션을 통한 리더와 태그의 통신과정을 예를 들어 설명하면 리더 1의 작동을 시작했을 때 리더는 Inventory 라운드를 초기화 하는 동안 태그의 세션 S2의 Inventoried Flag A를 B로 변경한다. 리더1은 세션 S2의 Inventoried Flag가 B인 모든 태그에 대해 반복적인 처리를 수행하며 처리된 태그는 세션 S2의 Inventoried Flag B를 A로 변경한다. 각 리더의 다단계 절차에 의해 모든 태그는 Inventoried Flag의 초기 상태와 관계없이 독립적으로 Inventory 된다.

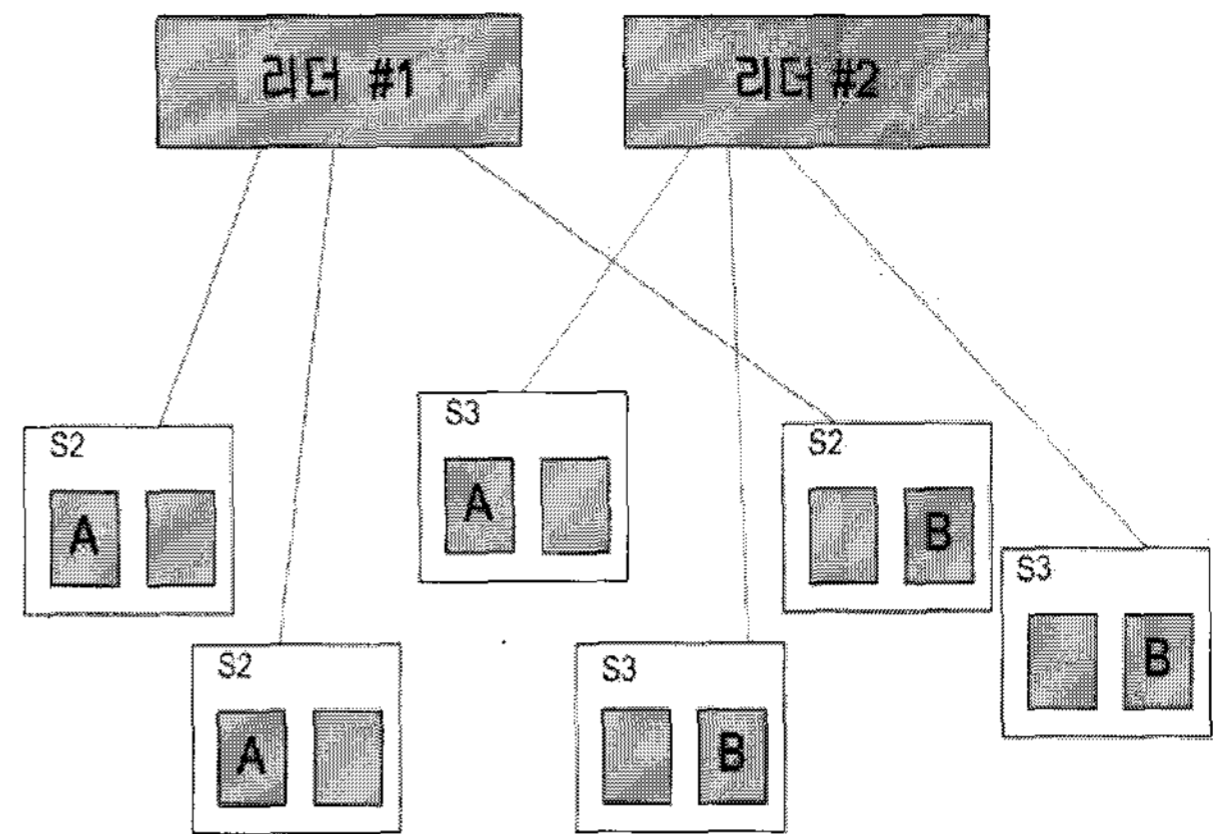
Gen2 프로토콜에서 세션기능이 없다면 <그림 5>와 같이 두 개의 리더가 한 개의 태그와 중복으로 통신을 시도하는 경우가 발생한다. 중복으로 통신을 시도하게 될 경우 간섭으로 인하여 잘못된 데이터가 수신 되거나 올바르게 태그 인식이 되지 않는다.



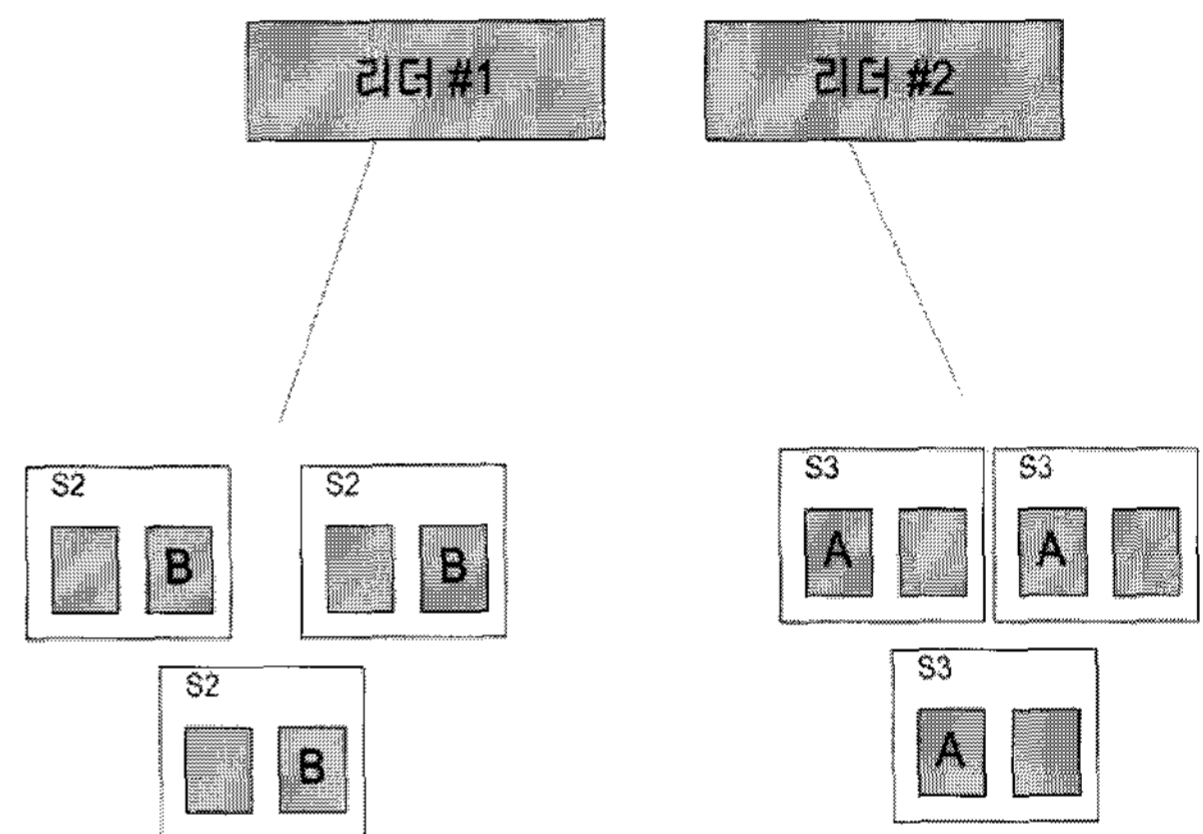
<그림 5> 세션기능이 없을 경우

세션 기능이 있는 Gen2 에서는 두 개의 서로 다른 리더는 각 세션을 통해서 태그와 통신을 시도 한다.

<그림 6>은 리더가 해당 세션을 가지고 통신을 시작 할 때 초기 상태를 나타낸다.



<그림 6> 각 리더가 태그와 통신하기 위한 초기 상태



<그림 7> 각 리더가 세션을 통하여 통신하고 있는 상태

초기에 리더가 해당하는 세션의 태그와 통신 준비가 완료 되면 각 리더는 <그림 7>과 같이 태그 집합군을 가지고 통신을 진행한다. 결국 이렇게 전송되고 수신되는 태그 데이터들은 미들웨어에 송신되어 다중 리더 운영에 있어서 태그들의 인식률을 높일 수 있다.

3. 다중 리더기 실험

Gen2 Protocol을 지원하는 Alien사의 ALR-9800을 이용하여 Field Test를 진행하였다. 문제점을 도출하기 위하여 안테나가 한 개일 때와 두 개인 상태에서 태그 인식률에 대한 테스트를 진행한다.

테스트 환경은 가능한 한 많은 양의 태그들을 테스트 할 수 있도록 태그를 매달 수 있는 틀을 제작하여 태그를 전시하고 안테나로부터의 거리는 인식률이 높은 35m 사이의 거리 중 3m 거리로 고정하여 설치하였다.

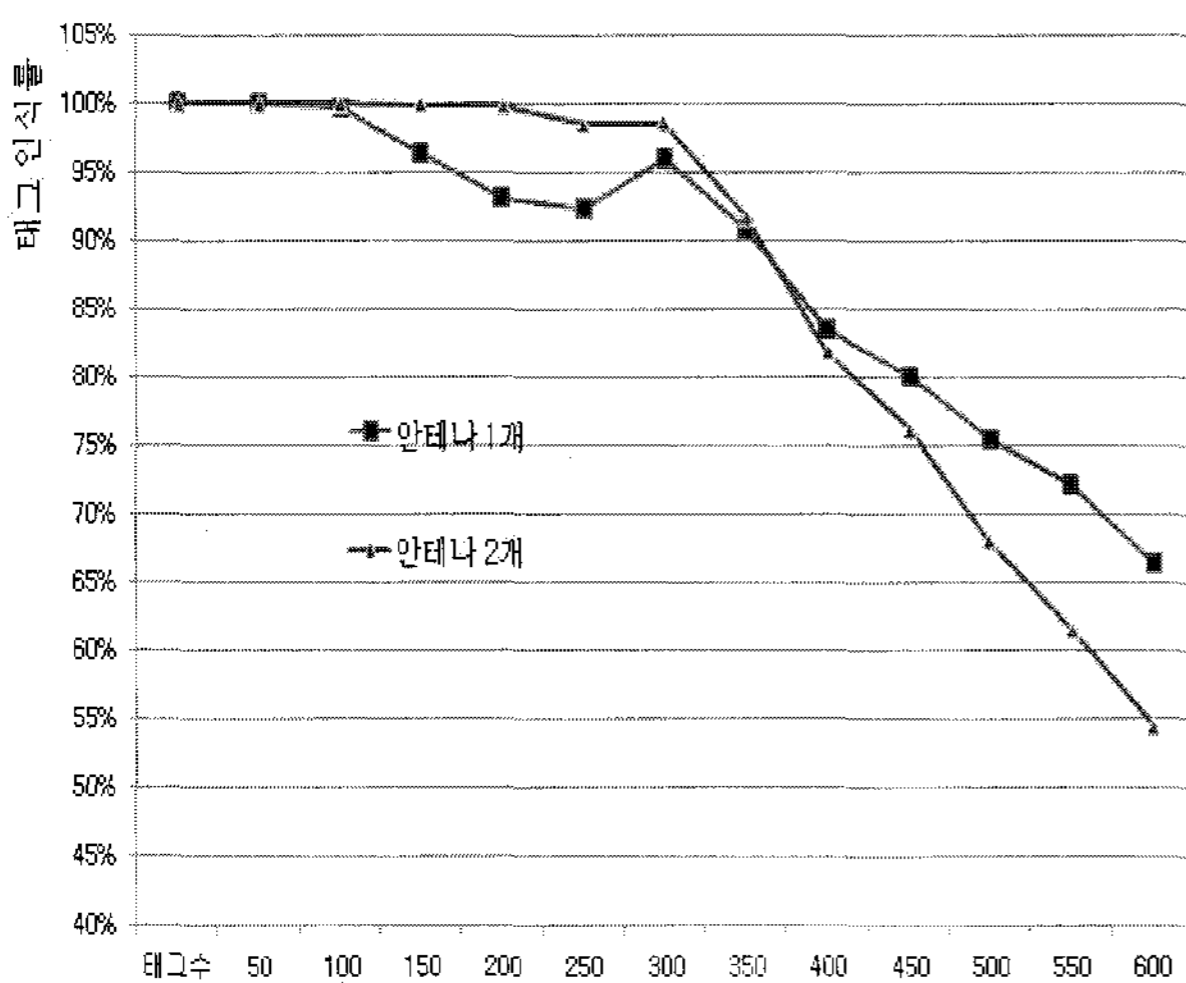
안테나가 두 개 일 경우 세션을 동일하게 고정하였다. 실험은 총3회 반복하였으며 변경 값은 태그의 수로 하였다.

<표 2> 필드 테스트 결과 테이블

태그수	실험 회수	1회		2회		3회	
	안테나 수	1개	2개	1개	2개	1개	2개
50		50	50	50	50	50	50
100		100	100	100	100	100	100
150		150	150	149	150	149	150
200		194	200	195	200	189	199
250		237	249	241	250	220	249
300		272	297	283	298	276	291
350		329	341	337	345	341	348
400		362	369	367	363	361	368
450		381	372	379	365	368	367
500		392	378	401	381	408	383
550		410	370	422	373	413	379
600		423	368	429	374	448	366
650		431	353	428	349	436	359

태그의 수가 늘어날 수록 한 개의 안테나와 두 개의 안테나의 인식률을 측정하였다. 그리고 안테나가 두 개 일 경우 리더가 주파수를 방사 하는 범위를 최대한 중복시키기 위해 가까이 붙여 놓았다. 인식하는 시간은 각 실험 마다 각 10초씩으로 제한하였다.

각 회수 마다 테스트를 진행한 결과 3회 연속 비슷한 실험 데이터를 볼 수 있었다. <표 2>는 각 회수 별로 인식 된 태그 수를 나타낸다.



<그림 8> 태그 변화량에 따른 태그 인식률 필드테스트

테스트 된 실험 데이터에서 각 회수별로 인식 된 태그들 평균 개수를 구하여 인식률을 구할 수 있었다. 그리고 <그림 8> 과 같이 그래프로 나타내었다. 태그 개수가 늘어날 수록 안테나가 하나일 경우보다 안테나가 두 개인 경우 점

점 인식률이 높아 졌다. 하지만 안테나 한 개가 인식 할 수 있는 태그 개수를 넘게 되면 두 개의 안테나를 사용하는 것이 한 개의 안테나를 사용할 때보다도 인식률이 낮아 졌다. 이 결과는 두 개의 안테나 사용 시 태그 개수가 많아짐에 따라 충돌이 발생하는 비율도 높아진다는 것을 알 수 있다. 이것은 RFID 시스템의 성능을 저하시키는 큰 원인이 되며, 다중 리더를 운영하기 위한 기술이 필요하다.

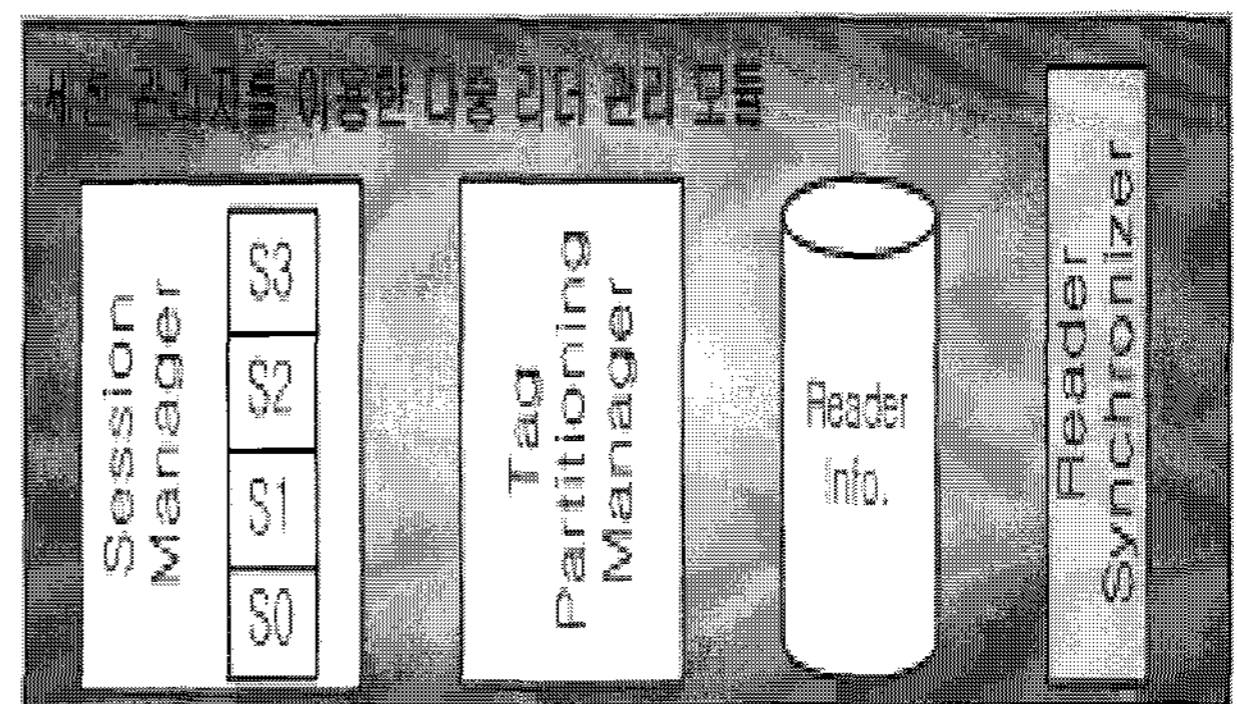
하지만 안테나가 두 개일 때 리더가 태그를 인식하는 범위가 완전 중복일 경우 충돌하는 확률이 두 배로 늘어나 태그 인식률이 안테나가 한 개일 때 인식률의 1/2 이 되어야 한다. 실제 실험에서 이론 적인 예상치 보다 높은 인식률을 나타낸 것은 Q값 설정을 통한 슬롯 개수 조절 기능과 같이 기타 다른 Gen2의 기술 때문으로 예상 된다.

4. 제안 시스템

4.1 제안 모듈 구성 요소

RFID 미들웨어의 다중 리더 관리 모듈 <그림 9>에서 리더 동기화(Reader Synchronizer)는 리더와 안테나 사이의 동기화를 유지하면서 리더의 상태를 모니터링 한다. 모니터링을 수행 하면서 안테나 연결 정보를 수집하여 리더 정보 관리자(Reader Info.)에게 데이터를 보낸다. 리더 정보 관리자는 안테나 연결 정보와 리더 정보들을 관리한다. 세션 관리자에서는 리더 정보 관리자를 통해 연결 된 안테나 개수를 알 수 있고 안테나 개수에 기반 하여 각 안테나에 세션을 부여 한다.

Gen2 프로토콜에 따르면 각 안테나는 Source Object 로 접근 가능하며 Source Object를 통해 각 안테나에 세션을 부여 할 수 있다[5]. 안테나 별로 유입되는 분할 된 태그들의 데이터들은 태그 분할 관리자에서 안테나의 정보와 태그 데이터와 함께 Data Collection & Filter 모듈로 보내진다.



<그림 9> 세션 관리자를 포함한 다중 리더 관리 모듈

각 컴포넌트들의 기능들을 요약하면 다음과 같다.

- 1) 리더 동기화 (Reader synchronizer)
 - : 리더와 미들웨어 시스템 사이에서 동기화를 유지하도록 하고 각 리더들의 모니터링을 담당 한다.
- 2) 리더 정보 관리자 (Reader Info Manager)
 - : 리더의 정보들과 안테나와의 연결 정보를 수집하여 관리한다. 이 정보들은 기타 다른 모듈 동작에 있어서 필요한 정보들을 제공하는 역할을 한다.
- 3) 세션 관리자 (Session Manager)
 - : 리더의 Source Object의 수와 기존에 부여된 세션을 배열로 관리한다. 이를 기반으로 각 리더의 Source Object에 새로운 세션을 부여 하고, 리더는 각 태그의 해당 세션을 통해서 통신한다. 각 Source Object는 논리적으로 각각의 안테나에 1:1로 매칭된다.
- 4) 태그 집합군 관리자 (Tag Partitioning Manager)
 - : 각 세션에서 수집된 태그 집합군들을 각 집합군별로 유효성 검사를 하며 고속 데이터 처리 모듈에서 정의하는 부하 처리기 역할을 한다.

4.2 다중 리더 운영 모듈의 구현 및 분석

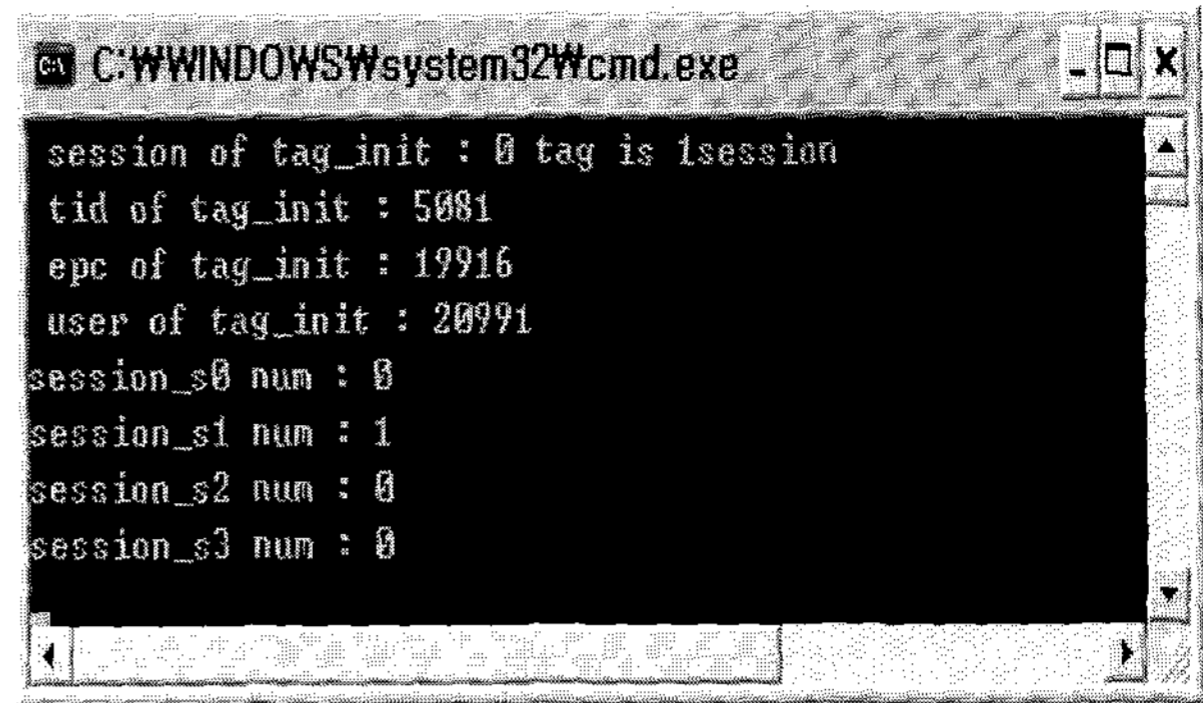
필드 테스트 결과에서 이론상의 데이터와 많은 차이가 있음을 확인하였다. 이는 리더 충돌을 방지하는 FDMA와 Q값을 이용한 슬롯개수 설정과 같은 Gen2 기술에 의한 것으로 판단된다. 그래서 동일한 환경에서 정확한 세션 관리자의 성능을 분석하기 위해 시뮬레이터를 구현하였다.

시뮬레이션 구현 환경은 OS는 Windows 2003 이고 개발도구는 Visual C++.Net 의 C언어를 사용하였다.

통신 프로토콜을 수행하는 함수는 "A Multiple Access Algorithm for Passive RFID tags"[6]에서 구현된 Gen2 프로토콜의 시뮬레이터를 참고 하였다. Main함수에 세션을 랜덤으로 부여하는 부분을 수정하여 본 연구의 시뮬레이션 목적에 맞게 구현하였다.

공통적으로 세션을 제외한 Q값 설정과 같은 충돌 기법을 고정값으로 두고 본 연구에서 제안하는 세션관리자를 통하여 리더 역할을 하는 함수에 세션을 부여한다.

여러 개의 가상 리더는 쓰레드를 이용하여 구현하였다. 리더의 인스턴스를 두 개 생성하여 세션을 이용하였을 때, 리더 충돌을 어느 정도 방지하는지 연구하였다.

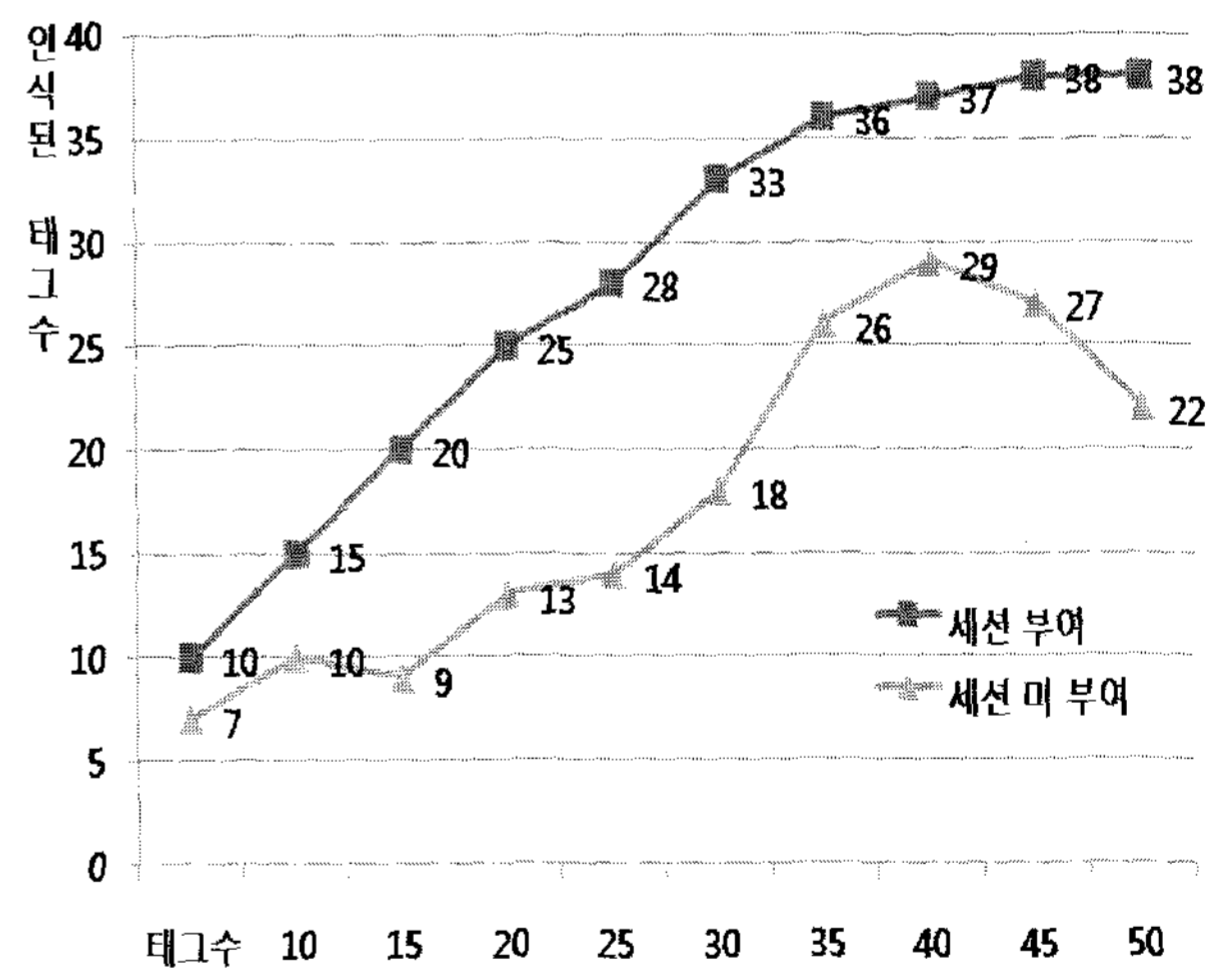


<그림 10> 시뮬레이터의 태그 초기화 수행

<그림 10> 은 시뮬레이터를 실행하였을 때 태그 하나에 대해 데이터의 초기화와 세션을 부여한 모습을 나타낸다. 두 개의 리더가 최대한 동시에 작동 되도록 쓰레드 타임을 1로 두고 배열로 구현된 태그의 접근은 랜덤으로 접근하게 하였다.

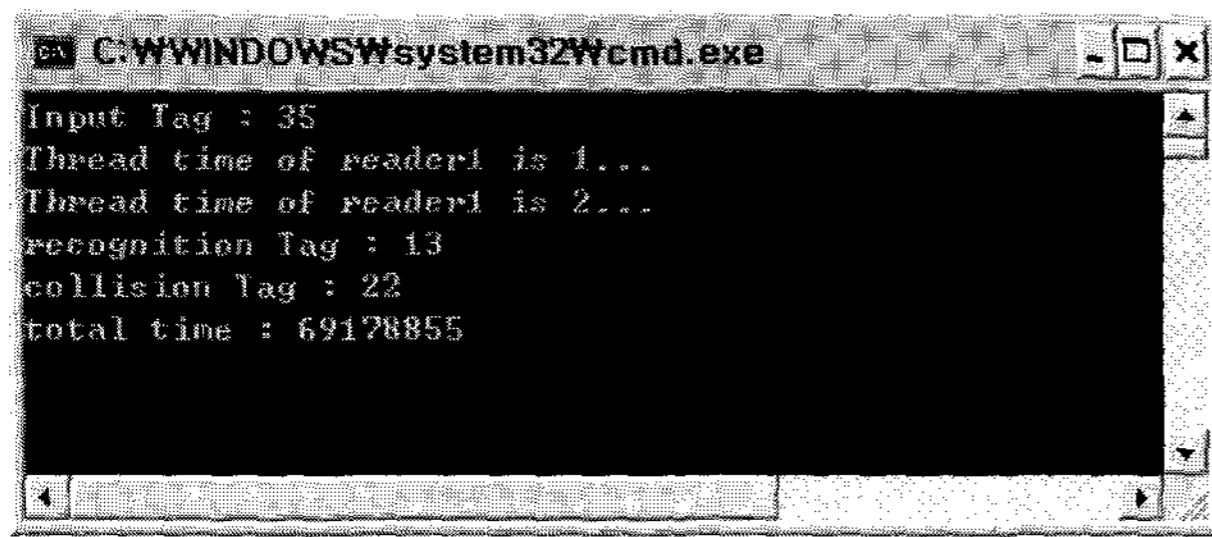
태그는 초기에 10개부터 5개씩 생성하여 세션을 부여하지 않았을 때와 제안된 모듈에 의하여 세션을 부여했을 때로 나누었다. 각 수행 횟수를 2번 씩 하여 평균치를 결과 값으로 나타내었다.

<그림 11>은 수행 결과에 대한 그래프이다.



<그림 11> 세션 관리자의 여부에 따른 인식된 태그 수

시뮬레이션 수행 시 하나의 리더가 인식할 수 있는 태그의 한계가 있었는데, <그림 11>의 결과로 동시 인식 태그수가 35개 이상이면 성능이 저하된다. 결론적으로 세션을 사용한 경우 세션을 사용하지 않고 두 개의 리더를 사용 하였을 때 보다 안정적으로 태그의 인식 프로세스를 수행하였다. 세션을 부여하지 않은 테스트 상황에서 <그림 12>와 같이 무려 2/3가량이 충돌 되는 상황도 발생하였다.



<그림 12> 세션이 없는 상태에서 2/3 가량의 충돌 발생

5. 결 론

물류 환경에서 관리를 위해 사용하는 RFID는 주파수의 인식 거리제한으로 한개 이상의 리더기를 설치하여 관리를 한다. 하지만 다중 리더기 실험을 통해 진행한 결과 <그림 11>과 같이 1개를 이용하는 경우보다 2개를 이용하는 경우의 인식이 더 낮았다.

본 연구에서는 이를 해결하기 위한 방안으로 세션 관리자를 포함한 다중 리더 관리 모듈을 제안하고 시뮬레이터를 구현하여 실험을 하였다. <그림 11>에서의 결과와 같이 35개 이후 세션이 없는 경우가 인식된 태그의 수가 2/3가량 충돌이 발생하는 결과를 얻었다.

효과적인 미들웨어를 위해서는 다중리더 운영에 관한 방안도 필요하지만, RFID의 데이터 특성상 동시에 많은 데이터가 유입이 된다. 이러한 많은 데이터 중 필요한 데이터만을 필터링 하기 위한 기술이나 입력 데이터의 부하를 해결하기 위한 다양한 방법에 대한 연구가 계속 진행되어야 할 것이다.

6. 참 고 문 헌

- [1] 최양희, 김대영, "미래인터넷 기술과 표준동향," 한국통신학회지, 제25권 3호, (2008.03) : 5-10
- [2] 장병준 등, "UHF RFID Air Interface 기술동향," RFID/USN 정보와 통신, (2007.08) : 14-22
- [3] 이서우, 이종권, "유비쿼터스 컴퓨팅 시대를 위한 상황인식 처리 기술," 정보과학회지, 제24권 3호, (2006.10) : pp.16-24
- [4] EPCTM, "Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz Version 1.0.9," (2004.01)
- [5] EPCTM, "Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz Version 1.0.9," (2004.01)
- [6] Sok-Won Lee, "A Multiple Access Algorithm for Passive RFID tags," Master's thesis, (2005.06), pp.34-46

저 자 소 개

박 상 현



인천대학교 공학 학사와 석사를 취득하였으며, 현재 인천대학교 컴퓨터 박사과정 중이며, 관심분야는 RFID / USN, 암호학, 컴퓨터 통신이다.

주소: 인천시 계양구 계산동 계산주공 아파트 105동 1010호

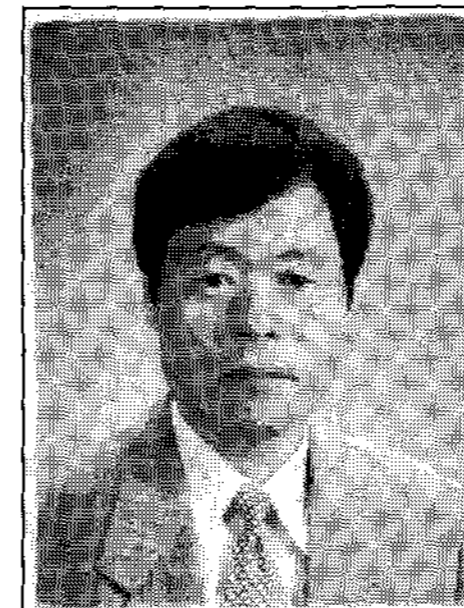
한 수



인천대학교 공학 학사와 석사를 취득하였다. 관심분야는 : RFID/USN, Gen2 Middleware이다.

주소: 서울특별시 강남구 역삼동 679-5

신 승 호



경희대학교에서 전자공학과 공학사를 취득하였으며, 경희대학교에서 공학석사와 박사를 취득하였으며, 현재 인천대학교 컴퓨터공학과 교수로 재직중이며, 관심분야는 컴퓨터 통신, 신호처리, 암호학이다.

주소: 서울시 서대문구 남가좌동 173-18