

# RFID 다중 태그 인식을 위한 STACK Bit-by-Bit 알고리즘

이재구\*, 유대석\*, 최재원\*, 최승식\*  
\*인천대학교 컴퓨터공학과  
e-mail:whistler29@incheon.ac.kr

## A Stack Bit-by-Bit Algorithm for RFID Multi-Tag identification

Jae-ku Lee\*, Dea-suk Yoo\*, Jae-won Choi\*, Seung-sik Choi\*  
\*Dept of Computer Engineering, University of Incheon

### 요 약

RFID 리더기가 영역내의 다수의 태그를 인식하기 위해선 충돌방지 알고리즘이 필수적으로 요구된다. 본 논문은 Auto ID Class 0에서 정의한 충돌방지 알고리즘인 Bit-by-Bit 이진트리 알고리즘(BBB)의 충돌 위치를 스택에 저장하고 이를 통해 다음 질의어를 결정함으로써 성능이 크게 개선된 Stack-bit-by-bit(SBBB) 알고리즘을 제안한다. 시뮬레이션을 통한 검증결과 질의-응답 횟수, 질의어의 크기, 응답어의 크기의 모든 면에서 성능이 개선된 것을 확인할 수 있었다.

### 1. 서론

RFID 기술이란 관리할 모든 개별 사물에 전자태그를 부착하여 리더기로 무선 전파를 태그에 보내면 태그가 자신의 ID정보를 리더기로 전달하는 방식으로 비접촉으로 태그 정보를 인식 할 수 있는 기술이다. 바코드의 문제점을 극복 할 수 있을 뿐만 아니라 태그의 정보를 인식하여 인터넷을 통하여 우리생활의 대변혁을 가져올 기술이다. 하지만 기기들 사이의 이질성, 리더기 사이의 충돌, 태그 사이의 충돌 등 아직 해결해야할 문제점이 있는 상황이다. 여기서의 충돌이란 태그들 간의 통신 과정에서 태그들 간에 간섭을 뜻하며 RFID 시스템의 성능과 안정성을 위해 반드시 해결해야할 문제이다. 이렇듯 다중 태그 환경에서 태그간의 충돌 현상을 회피하는 기술을 다중 태그인식 기술[1]이라 한다. 본 논문은 Auto ID Class 0 에서 정의된 Bit-by-Bit(BBB) 알고리즘[2]의 성능을 개선한 Stack Bit-by-Bit(SBBB) 알고리즘을 제안한다. SBBB 알고리즘은 기존의

BBB 알고리즘에 태그 충돌 위치 정보를 스택에 저장하고 이를 활용하여 성능을 개선하는 방안이다.

RFID 표준화는 현재 EPCglobal에 의해 진행 중으로 EPC(Electronic Product Code) code는 96비트가 사실상 RFID 표준코드로 자리잡아가고 있다. EPC code는<Version Header, EPC manager, Product Class, Serial Number>의 계층적 구조를 갖는다[3]. 시뮬레이션을 통한 검증결과 모든 경우에서 성능이 개선된 것을 확인할 수 있었으며 특히 태그 ID의 중복성이 클수록 우수한 성능을 보였다.

### 2. 관련 연구

#### 2.1 충돌방지 알고리즘의 분류

다중태그 식별을 위한 충돌회피 기술은 크게 트리기반의 결정적 알고리즘과 슬롯 알로하 기반의 확률적 알고리즘으로 구분 할 수 있다. 결정적 알고리즘은 태그 식별자들이 이진비트로 표현되어 이진트리를 구성한다. 인식은 이진트리를 순회하면서 태그들을 차례로 인식하는데 알고리즘에 따라 태그의 인식 순서가 결정되게 된다. 반면 확률적 알고리즘은 태그가 시간차에 따라 서로 다른 슬롯을 선택함으로써

“본 연구는 산업자원부 지정 인천대학교 동북아전자물류연구센터의 지원에 의한 것임”

써 태그를 인식하는 방법이다. 슬롯의 수가 적을 경우 충돌이 발생한 슬롯의 재전송에 따른 성능 저하가 심하고 슬롯의 개수가 필요이상 많으면 또한 성능을 저하시키는 요인이 된다. 트리기반 알고리즘은 태그의 메모리 유무에 따라 메모리형 알고리즘[4]과 비메모리형 알고리즘[5]으로 분류할 수 있으며 알고리기반의 알고리즘은 ID-슬롯 방식[6]과 bit-슬롯 방식[7]으로 구분된다.

<표 1> 충돌방지 알고리즘의 분류[8]

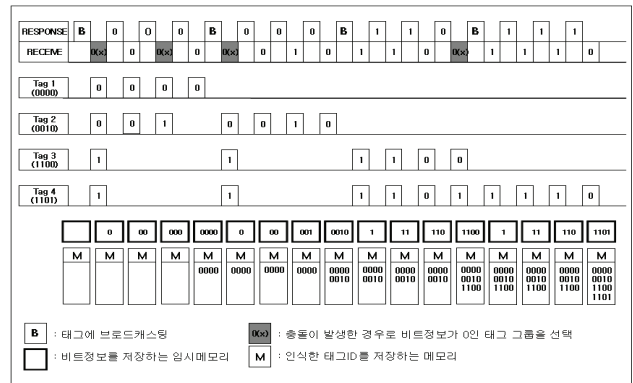
충돌 방지 알고리즘의 분류		
트리 기반 알고리즘 (결정적)	메모리형	비트-중재 알고리즘
		분할트리 알고리즘
	비메모리형	트리-위킹 알고리즘
		충돌추적트리알고리즘
슬롯-알로하기반 알고리즘 (확률적)	ID-슬롯형	I-Code
		STAC
	비트-슬롯형	Bit-Slot

2.2 Bit-by-Bit 알고리즘

태그의 초기화 과정이 끝나면 태그들은 자신의 최상위 비트(MSB)를 응답한다. 이때 응답은 두가지 경우가 존재하게 된다. 첫 번째는 리더기가 0 또는 1 둘 중 하나만을 수신하는 경우이다. 충돌이 발생하지 않은 경우로써 리더기는 응답받은 MSB를 메모리에 저장하고 다시 태그에 전송한다. 자신의 첫 비트를 수신한 태그는 다음 비트를 전송한다. 두 번째 경우는 0 과 1을 동시에 수신하여 충돌이 발생한 경우로 리더기는 0을 전송한 그룹과 1을 전송한 그룹 중 알고리즘에 의해 하나의 그룹을 선택하고 그 비트를 태그에게 전송한다. 전송받은 태그는 자신의 다음 번째 비트를 다시 리더기에 전송한다. 만일 선택 되지 않은 태그는 일시적으로 태그의 전송 명령에 응답하지 않는 Inactive 상태가 된다. 하나의 태그를 인식하게 되면 Inactive 상태의 태그를 다시 Active 상태로 바꾸게 된다. 따라서 태그ID의 길이가 j이고 태그의 개수가 n개라면 태그를 인식하기 위한 반복회수(R)는 (1)과 같다.

$$R = j \times n \quad (1)$$

<그림 1>는 태그 ID가 0000, 0010, 1100, 1101 인 4개의 태그에 대한 BBB 알고리즘을 적용하였을 경우의 일련의 태그 인식 과정을 나타낸 것이다.



<그림 1> BBB 알고리즘의 동작 예

<표 2> BBB 알고리즘의 전체 과정

Bit-by-Bit 프로토콜
리더기는 응답어를 임시로 저장할 메모리와 인식 완료한 태그ID를 저장할 메모리를 가짐
<b>리더기 동작</b>
Step 1 : 모든 태그에 브로드캐스팅
Step 2 : 태그로부터 응답 확인
- 0 또는 1 둘 중 하나의 신호를 수신한 경우 Step 3으로
- 0과 1을 동시에 수신한 경우 Step 4로
- 태그의 마지막 비트를 수신한 경우 Step 5로
- 모든 태그를 인식 하였다면 Step 6으로
Step 3 : 충돌 미발생
- 메모리에 수신한 비트를 저장
- 수신한 비트를 태그에 질의
- Step 2로
Step 4 : 충돌 발생
- 0을 송신한 그룹을 선택
- 1을 수신한 그룹을 Inactive 상태로 전환
- 메모리에 0을 저장
- 0을 송신
- Step 2로
Step 5 : 하나의 태그 인식
- Inactive 상태의 태그를 Active 상태로 전환
- Step 2로
Step 6 : 종료
- 모든 태그가 식별 된 상태로 알고리즘 종료
<b>태그 동작</b>
Case 1 : 태그가 활성화 상태인 경우
- 질의어와 매칭 될 때
- 다음 비트를 송신
- 질의어와 매칭되지 않을 경우
- 대기
Case 2 : 태그가 비활성화 상태인 경우
- 리더의 질의에 응답하지 않음
Case 3 : 비활성화 명령 수신한 경우
- 비활성화 상태로 전환
Case 4 : 활성화 명령 수신한 경우
- 활성화 상태로 전환

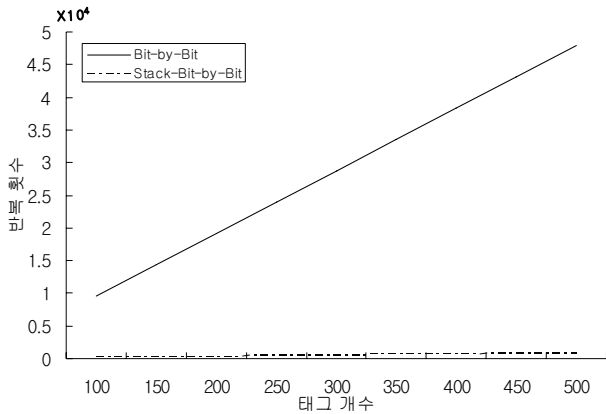
3. 제안된 SBBB 알고리즘

BBB 알고리즘은 이미 읽은 태그 정보를 활용을 전혀 하지 못하고 처음부터 다시 태그 인식을 시작한다. 이에 본 논문이 제안 하는 알고리즘은 리더기 내에 스택 변수를 두어 태그에서 발생한 충돌 위치 n을 스택에 저장하고 다음 태그 인식 시에 n-1번째 까지의 데이터를 다음번 질의어로 사용하는 것이다. 또한 스택에서 가져온 n 값이 태그 ID의 마지막 부분을 나타낸다면 한번의 질의-응답 과정을 통해 두 개의 태그 인식이 가능하게 된다.

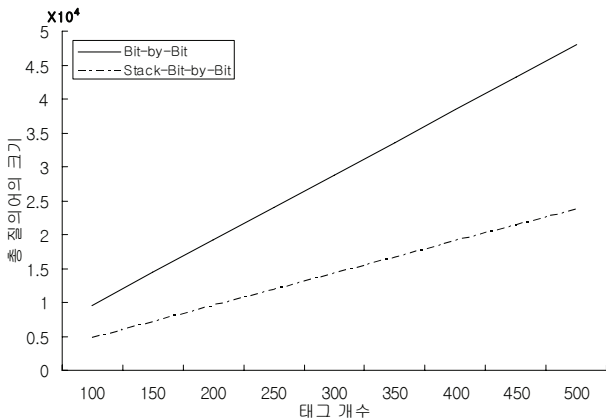


위 그림들은 Serial Number가 랜덤한 경우 각각의 성능을 나타내고 있다. 모든 경우에서 SBBB 알고리즘이 BBB 알고리즘에 비해 좋은 성능을 나타내고 있다.

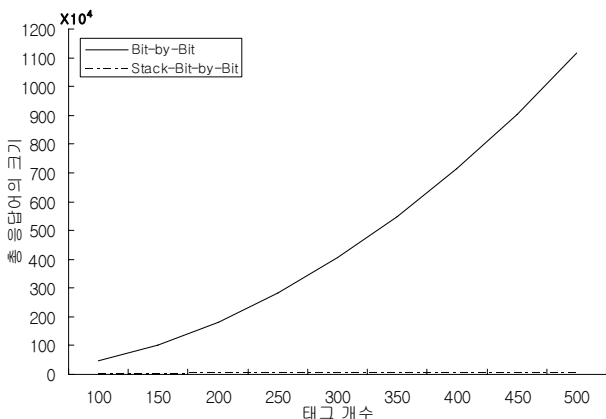
#### 4.2 Serial Number 순차적인 경우



<그림 6> Serial Number가 순차적인 경우 반복 횟수



<그림 7> Serial Number가 순차적인 경우 총 질의어의 크기



<그림 8> Serial Number가 순차적인 경우 총 응답어의 크기

Serial Number가 순차적인 경우 BBB 알고리즘에 비해 SBBB 알고리즘의 성능이 더 크게 개선된 것을 확인 할 수 있다. 이는 태그의 중복성이 클수

록 SBBB 알고리즘의 성능이 좋아 지기 때문이다.

#### 5. 실험 및 평가

본 논문은 Auto ID Class 0 에서 정의된 BBB 알고리즘의 성능을 개선한 SBBB 알고리즘을 제안하였다. BBB 알고리즘과는 달리 SBBB 알고리즘은 충돌 위치를 스택에 저장하여 다음번 질의에 이를 활용하였다. 시뮬레이션을 통한 성능 측정 결과 모든 경우에서 BBB 알고리즘에 비해 SBBB 알고리즘의 성능이 우수한 것을 확인 할 수 있었다. 특히 태그 ID의 중복성이 크고 태그 수가 많을수록 SBBB 알고리즘이 우수하였다.

#### 참고문헌

- [1] 이현지, 김종덕, "충돌 비트 위치를 활용한 RFID 다중 태그 인식 알고리즘," 한국통신학회논문지, Vol. 31, No. 4A, December 2005.
- [2] Auto-ID Center, Draft Protocol Specification for a Class 0 Radio Frequency Identification tag, 2003.
- [3] Auto-ID Center. Draft protocol specification for a 900MHz Class 0 Radio Frequency Identification Tag. Auto-ID Center. February 23, 2003.
- [4] Don. R. Hush, and Wood Cliff, "Analysis of Tree Algorithms for RFID Arbitration," In IEEE International Symposium on Information Theory, Pages 107-. IEEE, 1998.
- [5] A. Juels, R. Rivest and M. Szydlo. "The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy," Proceedings of the 10th ACM conference on Computer and communication Security, ISBM:1-58 113-738-9, Pages 103-111. 2003
- [6] H. Vogt, "Efficient Object Identification with Passive RFID Tags," In International Conference on Pervasive Computing, LNCS. 2002
- [7] Changsoon Kim, Kyunglang Park, Hiecheol Kim and Shindug Kim, "An Efficient Stochastic Anti-collision Algorithm using Bit-Slot Mechanism," PDP'2004, July 2004.
- [8] 권성호, 홍원기, 이용두, 김희철, "저비용 RFID 시스템에서의 충돌방지 알고리즘에 대한 성능평가," 한국통신학회논문지, Vol. 30, No. 1B, January 2005.