

RFID시스템에서 향상된 쿼리 트리 기반 충돌 알고리즘

한재일^o, 서현곤

한라대학교 정보통신공학부

kaeill@hanmail.net^o hgseo@hit.halla.ac.kr

KISS Korea Computer Congress 2006

Jaeil Han^o, Hyungon Seo

School of Information and Communication Engineering, Halla Univ.

요 약

RFID는 RF신호를 이용하여 물체에 부착된 태그(tag)를 읽어 물체를 식별하는 비 접촉 인식 기술이다. RFID 리더의 식별영역에 여러 개의 태그가 있는 경우 이들 사이의 충돌(collision)이 발생되기 때문에 이들을 식별할 수 있는 메커니즘이 있어야 한다. 본 논문에서는 먼저 기존에 제안된 트리 기반 메모리스 충돌방지 알고리즘들을 살펴보고, 본 논문에서 제안하는 향상된 쿼리 트리 기반 충돌 알고리즘(AQT : Advanced Query Tree based Collision Algorithm)을 소개한다.

1. 서론

RFID(Radio Frequency Identification)은 유비쿼터스 컴퓨팅 환경의 초기 기반 기술로 물체에 전자태그를 부착하여 RF신호로 물체의 태그 정보를 읽어와 물체를 식별하는 비 접촉 인식기술이다. 태그 생산 가격이 저렴해지고 표준화됨에 따라 RFID 시스템의 도입 시기는 한층 빨라지고 있다. RFID 기술은 기존의 바코드를 대체하여 상품관리를 네트워크화 및 지능화함으로써 유통 및 물류관리뿐만 아니라 보안, 안전, 소방/방재, 환경관리, 생산자동화 등 다양한 분야에 적용될 수 있다 [1].

RFID 시스템은 무선태그의 정보를 읽을 수 있는 리더와 사물 또는 물체의 정보를 저장하고 있는 무선 태그로 구성된다. 무선 태그는 전원의 유무에 따라 passive태그, active태그로 구분되는데, active태그는 자체에 전원이 내장되어 있으나 passive 태그는 리더기에서 전송된 반송파를 이용하여 생산된 전원을 이용한다. 사용하는 주파수에 따라 저주파(125kHz, 135kHz), 고주파(13.56MHz), 극초단파(433.92MHz, 860~960MHz) 등으로 구분한다.

하지만 RFID기술의 확산을 위해서는 해결해야 할 문제점들이 있다. 즉, 태그의 저가격, 저전력, 초소형화 문제와 사용자 프라이버시 및 정보 보호를 위한 방법, 리더기 영역 내의 다중 태그들의 충돌문제 등을 해결해

야 한다. 본 논문에서는 다중 태그들의 충돌을 해결하기 위한 알고리즘을 제안한다. 제안하는 알고리즘은 쿼리 트리에 기반 한 메모리스 알고리즘으로 기존의 쿼리 트리를 향상시킨 기법이다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 소개하고 3장에서는 본 논문에서 제안하는 향상된 쿼리 트리 기반 충돌 알고리즘을 제안한다. 4장에서는 기존 알고리즘과 본 논문에서 제안하는 방법의 성능을 비교하고 5장에서 결론을 내린다.

2. 관련 연구

다중 식별태그 문제는 충돌방지(anti-collision) 알고리즘을 통하여 해결할 수 있다. 현재까지 다양한 충돌방지 알고리즘들이 제안되었는데 크게 트리 기반 알고리즘(tree based algorithm)과 슬롯 알로하 기반 알고리즘(slot aloha based algorithm)으로 구분될 수 있다. 슬롯 알로하 기반 알고리즘은 확률적(probabilistic) 알고리즘으로 슬롯간의 시간차를 이용하여 태그 충돌을 회피한다. 하지만 적절한 슬롯의 개수와 종료시점에 따라 태그 인식이 증속되기 때문에 정확하게 모든 태그를 파악한다고 말하기 어렵다.

이에 비해 트리 기반 알고리즘은 결정적(deterministic) 알고리즘으로 이진 트리를 순회하며 모든 태그를 식별한다. 또한 트리 기반 알고리즘은 메모리형(memory) 알고리즘과 메모리리스형(memoryless) 알고리즘으로 구분

하는데 태그에 대한 질의만으로 태그를 구분할 경우 이는 메모리래스형 알고리즘이 되고 리더기의 질의에 대해 태그마다 상태 정보를 저장하고 관리에 의하여 태그를 식별할 경우 이를 메모리형 알고리즘이라 한다. 이 방법은 리더기가 태그들의 식별을 위해 반복적인 질의, 응답과정을 통하여 이루어지기 때문에 태그에는 인식을 위한 별도의 기억 장치가 필요 없기 때문에 태그의 저가격, 저 전력, 초소형화를 할 수 있다.

트리 워킹(TW: Tree Walking) 알고리즘은 대표적인 메모리래스 알고리즘으로 bit-by-bit 쿼리 방식으로 이진트리의 깊이 우선 탐색방법으로 태그를 식별한다[2]. 리더기가 d비트 길이의 프리픽스 $B(B=b_1b_2, \dots, b_d)$ 를 모든 태그에게 질의하면, 태그들은 자신의 식별자와 프리픽스를 비교하여 그 값이 동일하면 태그는 d+1비트를 리더에게 전송한다. 모든 태그가 "0" 또는 "1"을 전송했을 경우 충돌이 발생한 것이 아니기 때문에 리더기는 새로운 프리픽스 B_{d+1} 를 생성하여 다시 모든 태그에게 전송한다. 하지만 "0"과 "1"이 동시에 리더기에 수신된 경우 충돌이 발생하기 때문에 리더기는 $B \parallel 0$ 와 $B \parallel 1$ 을 프리픽스로 하여 다시 질의를 반복한다. 트리 워킹 알고리즘의 경우 비트단위로 태그를 검출하기 때문에 리더기 영역 내에 태그가 하나만 있는 경우도 마지막 비트까지 질의-응답을 해야 하고, 영역내의 많은 태그가 있을 경우 자주 충돌이 발생하기 때문에 많은 시간과 성능저하의 문제점을 가지고 있다.

쿼리 트리(QT: Query Tree) 알고리즘은 트리 워킹 보다 향상된 방법이다. 리더기가 d비트 길이의 프리픽스 $B(B=b_1b_2, \dots, b_d)$ 를 모든 태그에게 질의하면, 태그들은 자신의 식별자와 프리픽스를 비교하여 그 값이 동일하면 태그는 d+1비트에서 마지막 n비트까지 리더기에게 전송한다[3]. 리더기가 한 태그에게서만 응답을 받을 경우 B_{d+1} 태그가 식별된 것이고, 다수의 태그가 동시에 응답했을 경우 충돌이 발생되므로 리더기는 $B \parallel 0$ 와 $B \parallel 1$ 을 프리픽스로 하여 다시 질의를 반복한다. 하지만 쿼리 트리의 경우 리더기의 질의에 대해 무응답인 경우가 발생하고 많은 충돌이 발생하기 때문에 이를 처리하기 위한 부담이 증가한다.

최근에 제안된 충돌 추적 트리(collision tracking) 알고리즘[4]과 개량된 쿼리 트리(improved query tree) 알고리즘[1]은 그 기법이 동일한 것으로, 충돌을 감지하여 태그를 식별하는 아니라 충돌을 감지하여 태그를 식별하는 방법으로 정의할 수 있다. 리더기가 d비트 길이의 프리픽스 $B(B=b_1b_2, \dots, b_d)$ 를 모든 태그에게 질의하면, 태그들은 자신의 식별자와 프리픽스의 비교를 통

해 매칭이 이루어질 경우 태그 식별자의 d+1비트에서 마지막 n비트까지를 리더기에게 순서적으로 전송한다.

리더기는 태그가 전송한 식별자 정보를 수신함과 동시에 수신된 비트의 충돌 발생여부를 검사한다. 이때 "0"과 "1"이 동시에 수신된 경우 충돌이 발생하기 때문에 태그들에게 나머지 비트의 송신을 중지하도록 "전송중지 명령"을 전달한다. 전송중지 명령을 받으면 태그는 즉시 나머지 태그의 전송을 중지하고 새로운 프리픽스 B' 를 받을 준비를 한다. 리더기는 충돌이 발생한 k비트 이전까지 비트를 프리픽스 B 에 추가하여 새로운 프리픽스 B' ($B'=b_1b_2, \dots, b_d, \dots, b_{k-1}$)를 생성하여 다시 모든 태그들에게 전송한다.

하지만 충돌 추적 트리 알고리즘과 개량된 쿼리 트리(improved query tree) 알고리즘의 경우 해결해야 할 몇 가지 문제점들이 있다. 충돌이 발생하여 리더기가 전송 중지 명령을 태그들에게 전달하면 태그는 이를 식별할 수 있는 별도의 기능이 추가 되어야 한다. 즉, 프리픽스 B 를 받고 자신의 식별자와 매칭 되는 경우 태그는 한 비트씩 차례로 전달하는 도중에 전송중지 명령을 받기 때문에 이를 감지하고 처리할 수 있는 하드웨어와 소프트웨어가 태그에 추가적으로 포함되어야 한다. 이는 태그의 생산 비용을 증가 시킬 뿐만 아니라 초경량화 및 저 전력에 기반을 둔 태그의 생산에 많은 어려움을 주게 된다.

3. 향상된 쿼리 기반 충돌 알고리즘

본 절에서는 쿼리 트리에 기반을 둔 새로운 충돌감지 알고리즘인 향상된 쿼리 트리 기반 알고리즘(AQT: Advanced Query Tree based Collision Algorithm)을 제안한다. AQT는 트리 워킹 알고리즘, 쿼리 트리 알고리즘과 같이 충돌 회피 기법이며, 충돌 추적 트리 알고리즘이나 개량된 쿼리 트리(improved query tree) 알고리즘과 같이 충돌이 발생한 비트의 위치를 리더기가 검출하는 충돌 검출기능을 함께 가지고 있다.

기본 알고리즘은 쿼리 트리 알고리즘과 동일하다.

리더기가 d비트 길이의 프리픽스 $B(B=b_1b_2, \dots, b_d)$ 를 모든 태그에게 질의하면, 태그들은 자신의 식별자와 프리픽스를 비교하여 그 값이 동일하면 태그는 d+1비트에서 마지막 n비트까지 리더기에게 전송한다. 리더기가 한 태그에게서만 응답을 받을 경우 쿼리 트리와 같이 하나의 태그가 식별된다. 그런데 다수의 태그가 동시에 응답했을 경우 충돌이 발생한다. 이 경우 쿼리 트리 알고리즘의 경우 리더기는 새로운 프리픽스 $B \parallel 0$ 와 $B \parallel 1$ 을 생성하지만, AQT에서는 응답한 B_{d+1}, \dots, B_n 사이에서

최초 충돌이 발생한 비트 위치 k 비트를 찾고, 새로운 프리픽스 $B(=b_1b_2, \dots, b_d, \dots, b_{k-1})\|0$ 와 $B(=b_1b_2, \dots, b_d, \dots, b_{k-1})\|1$ 을 생성하는 것이 차이점이다. 이것은 충돌 추적 트리 알고리즘과 개량된 쿼리 트리(improved query tree) 알고리즘에서의 전송중지 명령을 전달하는 방법과는 다르다. AQT에서는 별도의 추가적인 명령이 필요 없고 또한 태그에 추가적인 하드웨어나 소프트웨어 없이 리더기에서 수행 가능하다.

예를 들어 리더기가 그림1과 같이 프리픽스 $B=1101001110101000$ 를 전달했을 때 태그 1, 2, 3, 4가 모두 프리픽스와 일치하기 때문에 프리픽스 이후 마지막 비트열까지 리더기에게 전달한다. 충돌이 발생한 경우 리더기는 태그들에게 전달받은 응답 비트 열을 검사하여 충돌된 비트 위치를 검출하여 새로운 프리픽스 $B'=11010011101010000011$ 을 생성하여 다시 태그들에게 전송한다.

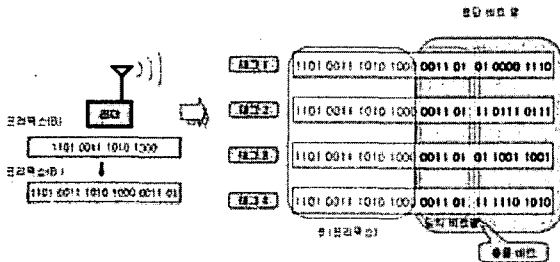


그림 2 AQT 충돌 감지 예

4장 성능평가

AQT의 성능 평가를 위해 C언어로 시뮬레이션 프로그램을 작성하여 트리 워킹 알고리즘, 쿼리 트리 알고리즘과 본 논문에서 제안하는 AQT에 대하여 질의/응답 횟수, 충돌 횟수, 전송 비트수를 각각 비교하였다. 태그 비트는 EPC Global 표준인 96비트로 하였고 태그의 개수는 $2^2, 2^4, 2^6, 2^8, 2^9, 2^{10}, 2^{11}$ 에 대하여 실험을 하였다. 각 실험 결과는 5회 반복 시험한 결과의 평균값이다.

실험 결과에서 트리 워킹 알고리즘은 쿼리 트리 알고리즘과 AQT보다 성능이 많이 떨어짐을 알 수 있다. 쿼리 트리 알고리즘과 AQT는 많은 차이는 없지만 대부분의 평가 항목에서 AQT가 더 좋은 성능을 보임을 알 수 있다.

Tag 개수	4	16	64	256	512	1024	2048
Tree Walking	질의/응답 횟수	505	1987	7824	30727	60923	120942
	무응답 횟수	0	0	0	0	0	0
	충돌 횟수	3	15	63	255	511	1023
	Tag당 평균 응답 횟수	126	124	122	120	119	118
Query Tree	질의/응답 횟수	8	44	172	704	1468	2871
	무응답 횟수	1	7	22	96	223	412
	충돌 횟수	4	22	85	351	734	1435
	Tag당 평균 응답 횟수	2	3	3	3	3	3
Advanced Query Tree	질의/응답 횟수	7	31	127	511	1023	2047
	무응답 횟수	0	0	0	0	0	0
	충돌 횟수	3	15	63	255	511	1023
	Tag당 평균 응답 횟수	2	2	2	2	2	2

표 1 성능 실험 결과

5장 결론

본 논문에서는 기존의 트리 기반 메모리리스 알고리즘인 쿼리 트리 알고리즘의 성능을 개선한 향상된 쿼리 트리 알고리즘(AQT)을 제안하였다. 제안한 AQT는 충돌을 회피하고 충돌 위치를 감지하여 빠른 시간 안에 리더기 영역내의 모든 태그들을 감지한다.

참고문헌

[1] Feng Zhou, Dawei jin, Chenling Huang and Hao Min, "White Paper: optimize the power Consumption of Passive Electronic Tags for Anti-collision Schemes," Auto-ID center Fudan Univ., October, 2003.

[2] Ari Juels, Ronald L. Ivest and Michael Szydlo, "The Block Tag : Selective Blocking of Tags for Consumer Privacy," *Proceedings of the 10th ACM conference on Computer and communication security*, ISBN:1-58113-738-9, pp.103-111, 2003.

[3] Ching Law, Kayi Lee and Kai-Yeung Sju, "Efficient Memoryless Protocol for Tag Identification," *Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communication*, ACM, pp. 75-84, August, 2000.

[4] 권성호, 홍원기, 이용두, 김희철, "RFID 시스템에서 트리 기반 메모리리스 충돌방지 알고리즘에 관한 연구", 정보처리학회논문지 C 제11권 제6호, 12월, 2004.