

논문 2007-44TC-4-1

RFID의 효율적인 태그인식을 위한 Adaptive Decision 알고리즘

(Adaptive Decision Algorithm for an Improvement of RFID
Anti-Collision)

고영은*, 오경욱*, 방성일**

(Young-Eun Ko, Kyoung-Wook Oh, and Sung-II Bang)

요약

본 논문에서는 RFID Tag 충돌방지를 위한 Adaptive Decision 알고리즘에 대해 연구 하였다. 이를 위해 기존의 RFID Tag 충돌방지 기법인 ALOHA기반의 기법과 이진 검색 충돌방지 기반의 알고리즘을 먼저 비교·분석하였다. 기존 알고리즘은 태그를 인식하기 위한 탐색횟수와 전송하는 데이터량을 감소시키는데 한계점을 가지고 있었다. 제안한 Adaptive Decision 알고리즘은 인식범위 내의 태그를 구별하기 위해, 호출에 응답한 모든 태그의 ID 비트 별 '1'의 개수를 계산하고, 개수가 작은 그룹의 태그를 우선적으로 식별한다. 각 태그 ID 비트의 '1'의 개수는 리더의 메모리에 저장하고, 식별된 태그 ID 비트의 '1'의 개수를 감산한다. 이와 같은 과정을 반복함으로써 인식범위 내의 모든 태그를 식별한다. 논문에서 제안한 능동적인 태그 선택기준과 간단한 가감 과정을 통해 불필요한 탐색횟수를 줄일 수 있다. 알고리즘의 성능평가는 태그를 인식하기 위한 리더의 반복횟수와 전송 데이터 량으로 나타내었다. 성능평가 결과, 기존의 알고리즘과 비교하여 Adaptive Decision 알고리즘의 반복횟수가 16.8% 감소되었고, 전송 데이터 량도 1/4배 감소된 것을 확인할 수 있었다.

Abstract

in this paper, we propose the Adaptive Decision Algorithm for RFID Tag Anti-Collision. We study the RFID Tag anti-collision technique of ALOHA and the anti-collision algorithm of binary search. The existing technique is several problems; the transmitted data rate included of data, the recognition time and energy efficiency. For distinction of all tags, the Adaptive Decision algorithm identify smaller one ,each Tag_ID bit's sum of bit '1'. In other words, Adaptive Decision algorithm had standard of selection by actively, the algorithm can reduce unnecessary number of search even than the existing algorithm. The Adaptive Decision algorithm had performance test that criterions were reader's number of repetition and number of transmitted bits for understanding tag. We showed the good performance of Adaptive Decision algorithm better than existing algorithm.

Keywords : RFID, Anti-Collision, 충돌방지기법, binary tree, slotted binary tree

I. 서론

정보화 사회가 발달함에 따라 모든 사물에 부착된 센서를 통해 정보를 습득하고 관리하는 네트워크, 즉 USN (Ubiquitous Sensor Network)에 대한 관심이 높

아지면서 바코드나 스마트카드 같은 인식장치에 대한 연구가 활발하게 진행되고 있다. 더불어 기존의 인식장치 보다 많은 기능을 갖으면서 전송율과 전력효율이 우수한 인식장치에 대한 연구도 선행되고 있다. 이와 같은 수요의 대응 방안으로 다양한 RFID (Radio Frequency Identification) 기술이 제안되었다.

RFID는 하나의 리더와 다수의 태그를 이용하여 사물의 정보 및 주변 상황 정보를 감지하는 자동 인식기술이다. 그러므로 시간이나 공간의 제약 없이 다양한 환경에

* 학생회원, ** 평생회원, 단국대학교 전자공학과
(Dept. of Electronics Eng. Dankook University)
※ 본 연구는 서울시 산학연 협력사업(10544)에 의하여
수행되었음.
접수일자: 2007년3월20일, 수정완료일: 2007년4월16일

적용이 가능하다. RFID 태그는 내장되어 있는 칩에 태그의 식별 번호와 태그가 부착되는 물체의 정보를 저장하고, 고유의 ID정보를 리더기에 전송한다. 리더기는 외부의 데이터베이스로부터 태그 ID에 대한 정보를 얻어서 태그가 부착한 물체를 정확하게 식별한다. RFID 기술은 데이터를 자동으로 인식하기 때문에 기존의 바코드나 스마트카드 보다 더 많은 양의 데이터를 처리할 수 있다. 또한 비접촉 방식이므로 환경 적용 능력이 우수하며, 동시에 다수의 태그를 인식할 수 있으므로 적용 환경 내에서 많은 양의 물체를 관리할 수 있다.

개발 초기의 RFID 시스템은 10cm이내의 접촉식카드(13.56MHz)가 사용 되었으나, 최근에는 900MHz 및 2.4GHz 대역을 사용하는 비접촉식으로 3m이내의 거리에서 인식할 수 있다^[1]. UHF 대역의 RFID 표준화와 관련하여 ISO/IEC JTC1 SC31 WG4에서 ISO18000-6 (860 ~ 960 MHz 대역 무전접속 표준)의 표준으로 Type-A 및 Type-B 규격이 확정되었으며, MIT Auto ID 센터를 중심으로 EPC (Electronic Product Code) 표준 Class 0과 Class 1 규격이 확정되었다. EPC Class 1 Generation 2 (EPC Class 1 Gen 2) 규격은 ISO의 Type-C 표준으로 채택되었으며, 현재 FDIS (Final Draft International Standard) 상태에 있다^[2].

그러나 RFID는 리더기와 다수의 태그 간의 하나의 동일 무선채널을 이용하기 때문에, 동시에 전송되는 각각의 태그들의 신호가 서로 충돌을 일으키게 된다. RFID 리더가 인접한 거리에 위치한 태그 신호를 인식할 때 발생하는 간섭은 리더 충돌(reader collision)이다. 이와 같은 리더 충돌은 주파수 간섭(frequency interference)과 태그 간섭(tag interference)을 일으키므로 RFID 시스템의 성능을 저하시킨다. 그러므로 리더 충돌을 방지하기 위한 충돌 방지(Anti-Collision)기법이 요구된다^[3].

본 논문에서는 RFID 시스템의 성능을 개선하기 위한 Adaptive decision 충돌 방지 알고리즘을 제안하였다. 이 알고리즘은 간단한 가산 방식을 이용하여 수신되는 태그 ID들의 '1'의 개수를 비트별로 계산함으로써 반복횟수와 태그가 전송하는 총 데이터 량을 감소시켰다. 검색 첫 번째 과정에서 모든 태그 ID의 최상위 비트를 호출하여 '1'의 개수를 계산하여 검색횟수가 작은 그룹을 우선적으로 호출한다. 호출에 응답한 태그 ID들의 최상위 비트를 제외한 나머지 비트의 '1'의 개수를 계산하여 메모리에 값을 저장하고, 검색횟수가 작은 그룹을 호출한다. 이 과정을 반복하여 태그 ID를 검색한다. 여기서 반복횟수를 줄이기 위해 메모리에 저장된 비트별 '1'의 개수에서 검색

된 태그 ID의 1의 개수를 감산한다. 즉 슬롯 이진 트리 알고리즘의 동작과 유사하나 간단한 가산 과정을 통해 반복 횟수 및 검색을 위한 데이터 전송량을 감소시킬 수 있다.

본 논문의 구성은 다음과 같다. II장에서 기존에 제안된 RFID 충돌기법에 대하여 고찰하고, III장에서는 제안한 Adaptive decision 충돌 방지 알고리즘의 특징과 동작 과정 및 반복횟수와 전송된 데이터 량을 수학적으로 도출한다. IV장에서는 모의실험을 통해 성능을 평가하였으며, V장에서 결론을 맺는다.

II. RFID 충돌 방지 기법

본 논문에서는 ISO/IEC 18000-6 Type-B에 적용 가능한 충돌방지 기법을 제안하였다. 그러므로 본 장에서는 현재 ISO/IEC 18000-6 Type-B에서 사용하는 충돌 방지 기술에 대해 고찰한다.

2.1 기본 이진 검색 알고리즘

기본 이진 검색 알고리즘은 전송받은 태그 ID들 중 비트가 동일하지 않은 지점들은 하나씩 호출함으로써 태그를 인식하는 방식이다.

표 2.1과 같은 4개의 태그가 리더의 인식 영역에 존재할 때, 기본 이진 검색 알고리즘을 이용하여 태그를 식별

표 2.1 4개의 태그 ID
Table 2.1 Tag ID.

Tag ID
0010
0101
1010
1011

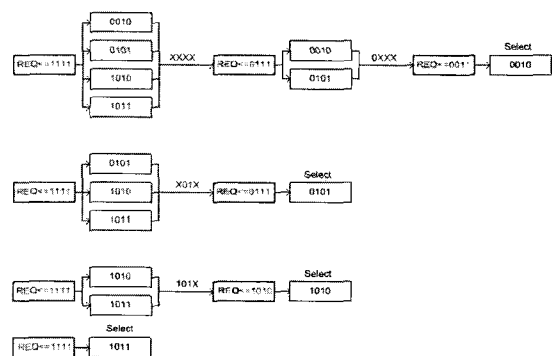


그림 2.1 기본 이진 검색 알고리즘의 동작
Fig. 2.1 Basic Binary Search Algorithm.

하는 과정은 그림 2.1과 같다.

처음에는 '≤1111'를 만족하는 태그를 요청한다. 즉 검색 범위 내의 모든 태그를 호출한다. 응답한 태그 ID를 비트별로 비교하여 태그 ID의 n번째 비트가 '1'이면 '1', '0'이면 '0', '1'과 '0'이 모두 존재하면 충돌(X)로 표현한다. 표 2.1의 태그는 ID의 모든 비트들이 서로 다르므로 'XXXX'이다. 충돌지점을 확인한 후 충돌이 발생한 최상위 비트를 '0'으로 지정하고 나머지 충돌 비트는 '1'로 지정하여 비교 값 '0111'을 생성하고, 비교 값 이하의 태그 ID(≤0111)를 호출한다. 이 때 응답한 태그는 2개이다. 그러면 다시 2개의 태그 ID 비트들을 비교하여 충돌지점을 확인하고, 위에서와 같이 비교 값을 생성하여 이하의 태그를 호출한다. 이와 같은 한 단계 과정으로 태그 ID '0010'을 검출하였고, 검출된 태그 ID는 비활성화 시키고 모든 태그 ID를 호출하여 같은 과정을 반복하면서 인식 영역 내의 모든 태그 ID를 검출한다.

인식 영역 내의 총 태그 수가 N개일 때, 하나의 태그를 인식하기 위한 반복횟수 L(N)은 식 (2.1)과 같다^[6,8].

$$L(N) = \frac{\log(N)}{\log(2)} + 1 \quad (2.1)$$

2.2 동적 이진 검색 알고리즘

기본 이진 검색 알고리즘은 태그가 리더에게 모든 ID bit를 전송기 때문에 ID 길이가 길어지면 하나의 태그를 인식하기 위해 전송되는 데이터 량이 증가한다. 이는 하나의 태그를 처리하는 시간을 증가시키고, 태그의 에너지 소모량도 증가시킨다.

동적 이진 검색 알고리즘은 기본 이진 알고리즘의 문제점을 보완하기 위하여 제안되었다. 기본적인 태그 ID 검출 방식은 동일하나 리더가 호출 명령 외에 VB(Valid Bit)를 추가하여 전송한다. VB는 충돌이 발생한 비트의

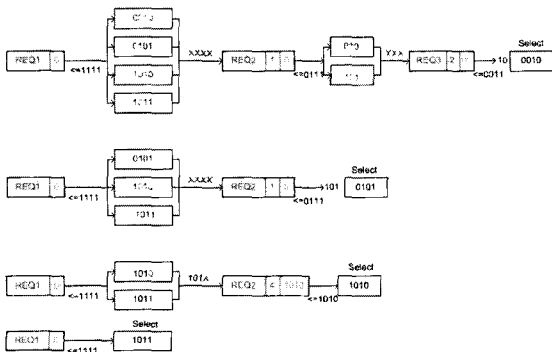


그림 2.2 동적 이진 검색 알고리즘의 동작
Fig. 2.2 Dynamic Binary Search Algorithm.

위치를 나타낸다. 표 2.1의 태그 ID를 동적 이진 검색 알고리즘으로 식별하는 과정은 그림 2.2와 같다.

기본 이진 검색 알고리즘과 같이 모든 비트를 요청하여 충돌지점을 확인한다. 그 후 충돌이 발생한 최상위 비트의 위치 '0'을 VB에 저장하고 태그 ID를 요청한다. 태그는 VB의 값에 따라 해당 위치의 비트가 '0'인 태그만 응답한다. VB를 이용하여 리더와 태그 모두가 충돌 발생 위치를 알게 되었으므로 태그는 VB의 비트를 제외하고 나머지 비트만 응답한다. 이와 같은 과정을 반복하여 인식 영역 내의 모든 태그 ID를 검출한다.

동적 이진 검색 알고리즘에서 하나의 태그를 검출하기 위한 반복횟수는 기본 이진 검색 알고리즘과 동일하나 데이터의 전송량은 감소하므로 효율성이 높다^[6].

2.3 슬롯 이진 트리 알고리즘

슬롯 이진 트리 알고리즘은 인식 영역 내의 태그 ID의 상위 비트부터 '0'과 '1' 그룹으로 나누어 요청하면서 태그 ID를 검출하는 방식이다. 태그 ID를 검색하기 위해 최상위 비트가 '0'인 그룹의 태그들을 호출한다. 호출에 응답한 태그는 활성화 시키고 응답하지 않은 태그는 비활성화 시킨다. 응답한 태그가 여러 개일 경우, 응답한 태그들 중 최상위 다음 비트가 '0'인 그룹의 태그들을 호출한다. 이와 같은 방법을 반복하여 하나의 태그를 검출하면 리더는 인식된 태그가 전 단계에서 분기된 슬롯에서 다른 그룹을 선택하여 요청 메시지를 보내 위의 인식과정을 반복하게 된다.

표 2.1의 태그 ID를 슬롯 이진 트리 알고리즘으로 검출하는 과정은 그림 2.3과 같다.

처음에 최상위 비트, 0 번째 비트가 '0' 그룹을 호출하면 '0010'과 '0101' 태그가 응답한다. 나머지 태그는 비활성화 시키고 이 두 개 태그의 1번째 비트가 '0'인 그룹을 호출한다. 이 과정을 통해 '0010'를 검출하였다. 다음으로 1번째 비트가 1인 그룹을 호출하면 '0101'을 검출할 수 있다. 이와 같은 과정으로 모든 태그를 인식한다^[8].

슬롯 이진 트리 알고리즘에서 '0' 또는 '1'인 그룹을 선

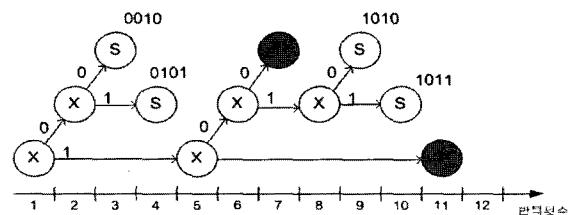


그림 2.3 슬롯 이진 트리 알고리즘의 동작
Fig. 2.3 Slotted Binary Tree Algorithm.

택할 확률을 p , 태그 수가 n 일 때, n 개의 태그를 인식하기 위하여 필요로 하는 총 반복횟수는 다음과 같다^[9].

$$I_{BEST} = 1 + \sum_{k=2}^n \binom{n}{k} \frac{2(k-1)(-1)^k}{[1-p^k - 1(1-p)^k]} \quad n \geq 2 \quad (2.2)$$

2.4 비트별 이진 트리 알고리즘

비트별 이진 트리 알고리즘은 리더의 요청 메시지에 따라 태그가 자신의 ID를 한 비트씩 전송하는 방식이다. 태그들이 보내는 비트 사이에 충돌이 발생하지 않으면 리더는 태그로부터 받은 비트를 메모리에 저장한 후 다음 비트를 요청하게 된다. 그러나 충돌이 발생하면 리더는 0 또는 1 중에서 하나를 선택하여 이에 해당하는 과정을 태그의 일련번호 길이만큼 반복함으로써 하나의 태그를 인식하게 된다. 그림 2.4는 비트별 이진 트리 알고리즘을 이용하여 표 2.1의 태그를 인식하는 과정이다.

리더기의 요청에 의해서 모든 태그는 자신의 첫 번째 bit를 전송하게 된다. 이 경우에는 첫 번째 비트에서 충돌이 발생하였으므로 리더는 '0'의 값을 가지는 태그만을 임의로 선택하여 메모리에 저장하고 '1'의 값을 가지는 태그는 일시적으로 비활성화 시킨다. 활성화된 나머지 태그 ID 중에서 다음 두 번째 비트를 호출한다. 활성화된 태그 ID '0010'과 '0101'은 두 번째 비트에서 충돌이 발생하였다. 위와서와 같은 방법으로 '0'을 메모리에 저장하고 '0' 값을 가지는 태그만 호출한다. 이후에 응답하는 태그는 '0010' 하나이므로 충돌 없이 한 비트씩 호출하여 메모리에 저장한다. 태그 ID가 검색되면 검색된 태그는 비활성화 시키고 이전에 비활성화시킨 태그를 활성화시켜서 과정을 반복하면 모든 태그 ID를 인식할 수 있다^[10].

태그 ID가 j 비트인 n 개 태그를 인식하기 위한 반복 횟수는 식(2.3)과 같다.

$$I_{BBT} = n \times j \quad (2.3)$$

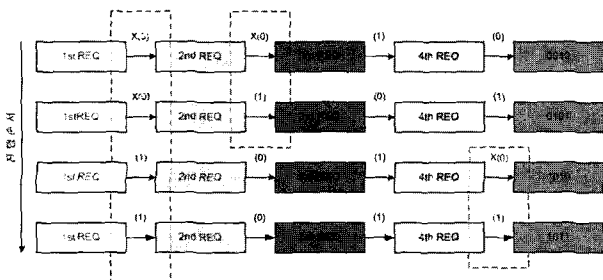


그림 2.4 비트별 이진 트리 알고리즘의 동작
Fig. 2.4 Bit by Bit Binary Tree Algorithm.

III. Adaptive Decision 알고리즘 설계

3.1 기존 알고리즘의 문제점

기존의 이진 검색 알고리즘에서 태그는 모든 ID 비트를 리더기에 전송한다. 태그 충돌이 일어난 시점으로부터 다음 비트를 검색하는 것이 아니라 처음부터 다시 모든 비트를 검색하게 된다. 따라서 태그 ID의 충돌이 일어난 비트수가 S 이고 태그의 ID 비트 수가 n 일 때, 태그 ID의 길이가 보내고자 하는 ID의 길이보다 S^{n-1} 만큼 더 길어지게 된다. 그러므로 태그 ID의 길이가 길어지면 하나의 태그를 인식하기 위해 전송해야 하는 데이터량이 증가할 뿐만 아니라 태그의 처리 시간 증가로 인해 에너지 소모량을 증가시키는 결과를 가져 온다.

슬롯 이진 트리 충돌방지 알고리즘과 비트별 이진 트리 충돌방지 알고리즘은 임의의 '0'과 '1'의 그룹을 나누어 반복적으로 태그를 구별함으로써 불필요한 검색과정을 거친다. 태그를 호출하는 반복횟수의 증가는 태그의 정보 전송 횟수를 증가시킴으로써 제한된 에너지를 사용하여야 하는 태그의 성능을 저하시키는 원인이 된다. 또한 태그에서 전송되는 정보비트의 증가는 리더의 정보처리능력에 큰 부담을 준다. 이는 결국 고가의 리더 연산장치를 요구하게 됨으로써 RFID 전체 시스템의 비용을 증대시키게 된다.

3.2 Adaptive Decision 알고리즘

본 논문에서 제안한 Adaptive Decision 알고리즘은 간단한 메모리와 가산기를 이용함으로써 태그의 호출횟수와 전송 데이터량을 줄였다. 제안된 알고리즘의 동작과정은 그림 3.1과 같다.

그림 3.1과 같은 Adaptive decision 알고리즘은 별도의 SB(sum of bit)를 메모리상에 할당하여 호출에 응답한 태그 ID들의 합을 계산하고, 저장한다. 이 SB의 값을 이

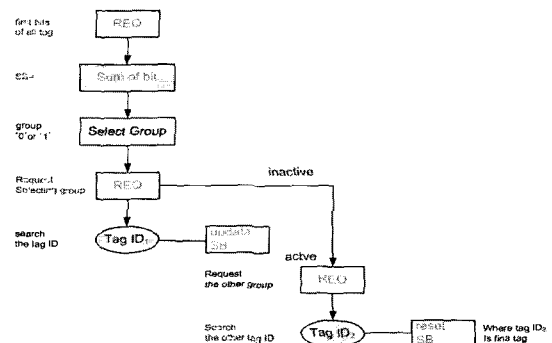


그림 3.1 Adaptive Decision 알고리즘의 동작
Fig. 3.1 Adaptive Decision Algorithm.

표 3.1 제안한 알고리즘의 '1'의 개수별 선택조건
Table 3.1 The selection condition of the proposed algorithm.

'1'의 개수가 0개인 경우	'0'선택
'1'의 개수가 2 ⁿ 개인 경우	'1'선택
'1'의 개수가 1~2 ⁿ⁻¹ 개인 경우	'1'선택
'1'의 개수가 2 ⁿ⁻¹ + 1~2 ⁿ - 1개인 경우	'0'선택

용하여 충돌 지점 및 태그의 ID를 간단한 논리로 연산함으로써 태그 식별에 있어서의 반복 횟수 및 전송 데이터를 줄였다.

첫 번째 단계에서 리더는 모든 태그 ID의 첫 번째 비트를 호출하고, 응답한 값의 총합을 SB에 저장한다. SB의 값에 따라 '1' 그룹과 '0' 그룹을 선택한다. 선택할 때에는 간단한 논리를 이용하여 태그가 작은 그룹을 먼저 선택한다. 예를 들어 2ⁿ개의 태그를 사용하는 경우의 그룹 선택기준은 표 3.1과 같다.

'1'의 개수가 0이면 그 비트는 무조건 '0'이므로 '0'그룹을 선택하고, '1'의 개수가 태그의 개수와 같으면 그 비트는 무조건 '1'이므로 '1'그룹을 호출한다. 이 경우 간단한 가산과정과 논리연산으로 인해 나머지 그룹에 대한 무의미한 반복과정을 줄일 수 있다. 만약 '1'의 개수가 2ⁿ⁻¹개와 같거나 작으면 전체 태그 중 최상위 비트가 1인 태그가 0인 태그와 같거나 작으므로 '1' 그룹을 선택한다. 반면에 '1'의 개수가 2ⁿ⁻¹보다 크면 최상위 비트가 '0'인 태그가 '1'인 태그보다 작으므로 '0' 그룹을 선택한다.

위와 같은 과정으로 그룹을 선택하고 나면 해당 그룹 태그들의 모든 ID 비트를 호출하고, 응답하지 않은 태그는 비활성화 시킨다. 응답받은 ID 비트는 각 비트별로 합산하여 SB_n에 저장한다. 저장한 SB_n의 최상위 비트에 따라 그룹을 선택하는 과정을 반복하여 태그 ID를 검출한다. 태그 ID 하나가 검출되면 저장된 SB_n에서 태그 ID 비트를 감산하고 SB_n 값을 업데이트 한다. 한 그룹의 태그 ID를 모두 검출하면 비활성화 시킨 태그를 활성화시켜서 같은 작업을 반복하여 태그를 모두 식별하고, 식별 작업이 끝나면 최종적으로 SB_n 값을 초기화한다.

제안한 알고리즘은 SB를 계산함으로써 태그 ID를 예상하여 호출할 수 있으므로 태그 검색을 위한 반복횟수를 줄일 수 있다. 예를 들어 '00'과 '01'의 태그를 검색하는 과정에서 '01'태그가 먼저 검색되면 검색해야 할 태그는 하나가 남았지만 '00', '10', '11'을 모두 검색해야 한다. 그러나 제안한 알고리즘을 사용하면 초기에 SB=01이고,

'01'을 검색하고 나면 SB=00이 되므로 남은 태그의 ID는 '00'임을 예상할 수 있다. 그러므로 '00' 하나만 검색하면 모든 태그 ID의 식별이 가능하다.

태그 ID 비트가 m비트인 태그 N개를 사용하는 경우 태그 ID를 검색하기 위한 평균 반복횟수, R(N)은 식 (3.1)과 (3.2)와 같다.

N이 짝수 일 경우 :

$$R_{even}(N) = 2 \times R\left(\frac{N}{2}\right) + 1 \quad (3.1)$$

N이 홀수 일 경우 :

$$R_{odd}(N) = R\left(\frac{N-1}{2}\right) + R\left(\frac{N+1}{2}\right) + 1 \quad (3.2)$$

여기서, 초기값 R(1) = 2, R(2) = 3, R(3) = 4이고, N > 3 이다.

또한 태그를 검색하는 과정에서의 평균 총 전송 비트 수, B_m(N)은 다음과 같다.

N=2 일 경우 :

$$B_2(N) = N + 2(m-1) \quad (3.3)$$

N=3 일 경우 :

$$B_3(N) = m \times N + 2 \quad (3.4)$$

N이 3보다 큰 짝수 일 경우 :

$$B_{m(even)}(N) = 2 \times B_{m-1}\left(\frac{N}{2}\right) + N \times (m-1) \quad (3.5)$$

N이 3보다 큰 홀수 일 경우 :

$$B_{m(odd)}(N) = B_{m-1}\left(\frac{N-1}{2}\right) + B_{m-1}\left(\frac{N+1}{2}\right) + N \times (m-1) \quad (3.6)$$

기존의 알고리즘과 비교하기 위해 제안한 알고리즘으로 표 2.1의 태그를 식별하는 과정을 그림 3.2에 나타냈다.

첫 번째 단계에서 모든 태그의 최상위 비트를 요청하여 합을 구하면 2가 된다. 표 3.2의 1 단계 선택의 조건에 의해 최상위 비트가 '1'인 그룹을 먼저 호출하고 '0' 그룹은 비활성화 시킨다. 두 번째 단계에서 '1' 그룹의 나머지 비트를 호출하여 응답한 태그의 SB를 산출하면 '021'이다. 2단계 선택 조건에 의해 2, 3번 비트는 충돌이 없고,

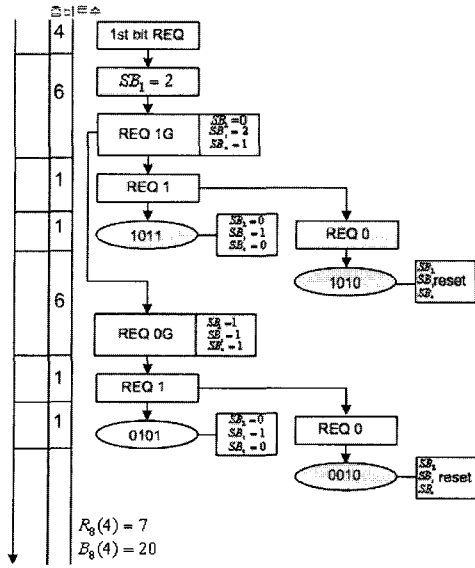


그림 3.2 Adaptive Decision 알고리즘을 이용한 4개 태그의 인식 과정

Fig. 3.2 The tag search process using Adaptive Decision algorithm.

표 3.2 단계별 SB에 따른 선택 조건
Table 3.2 The selection condition of stage.

	SB의 1의 개수	선택
1 단계	0	무조건 '0'
	4	무조건 '1'
	1~2	'1' 그룹
	3	'0' 그룹
2 단계	0	무조건 '0'
	2	무조건 '1'
	1	'1' 그룹

무조건 '0'과 '1'로 판별이 가능하므로 마지막 4번째 비트만 '1' 그룹을 선택하여 호출한다. 이 과정을 통해 '1011'을 검출할 수 있으며, SB는 '021'에서 '011'을 감산한 결과, '010'을 업데이트 한다. 그리고 이전에 2, 3번 비트는 결정이 되었으므로 4번째 비트가 '0'인 태그를 호출하여 '1010'도 검출한다. '1' 그룹을 모두 검출하였으면 첫 번째 단계에서 비활성화 시킨 '0' 그룹을 활성화 시켜 같은 과정을 반복하면 4개의 태그 ID를 모두 검색할 수 있다.

제안한 기법을 이용하여 표 2.1의 4가지의 태그를 찾는 경우, 검색을 위한 반복횟수는 7번이고, 이때 전송하는 데이터 량은 20 비트로 기존의 기법에 비해 반복횟수와 전송 데이터 량이 감소한 것을 알 수 있다.

IV. Adaptive Decision 알고리즘 성능 평가

RFID 시스템에서의 성능을 좌우하는 요소는 태그의

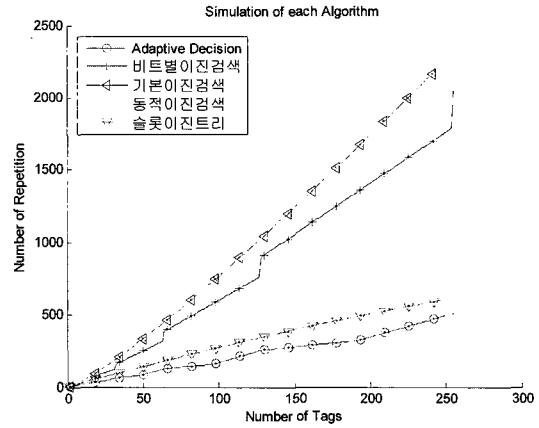


그림 4.1 태그 수에 대한 반복횟수 비교
Fig. 4.1 The number of repetition vs. tag.

탐색시간과 태그를 인식하기 위한 데이터 처리량, 태그 탐색의 정확도이다. 태그를 인식하기 위한 태그의 탐색시간은 태그를 인식하기 위한 반복횟수에 비례한다. 또한 태그를 인식하기 위한 반복횟수는 리더가 처리해야 하는 데이터량과 비례한다. 즉, 우수한 RFID 시스템은 적은 태그 탐색횟수와 데이터량으로 정확하게 태그를 찾을 수 있어야 한다.

본 논문에서 제안한 Adaptive Decision 알고리즘의 성능을 평가하기 위하여 태그 수에 따른 검색 반복횟수와 전송한 총 비트수를 기존 제안된 알고리즘들과 비교하였다. MATLAB을 이용하여 태그 ID 비트는 8비트 일 때, 사용하는 태그의 개수가 256개인 경우까지 분석하였다. 여기서 각각의 알고리즘은 모든 태그를 인식할 수 있다고 가정하였다.

그림 4.1은 기존의 충돌 방지 기법과 제안한 기법의 태그 수에 대한 반복횟수를 분석한 그림이다.

그림에서 보는 바와 같이 태그를 식별하기 위한 반복횟수가 기존의 충돌방지 기법에 보다 제안한 기법의 적은 것을 확인할 수 있었다. 이 결과를 태그의 개수에 따라 정량적으로 정리하면 표 4.1과 같다.

기존의 충돌기법 중에서 기본 이진 검색 알고리즘이 반복횟수가 가장 크고, 슬롯 이진 트리 알고리즘이 가장

표 4.1 알고리즘에 따른 태그별 반복횟수
Table 4.1 The number of repetition.

	2tag	4tag	8tag	16tag	32tag
Adaptive Decision	3	7	15	31	63
기본 이진 검색	4	12	32	80	192
동적 이진 검색	4	12	32	80	192
비트별 이진 검색	2	8	24	64	160
슬롯 이진 트리	5	11	23	46	92

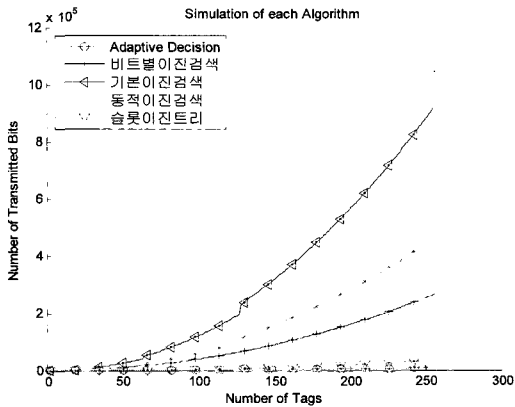


그림 4.2 태그 수에 따른 총 전송 데이터량 비교
Fig. 4.2 Total transmitted data vs. tag.

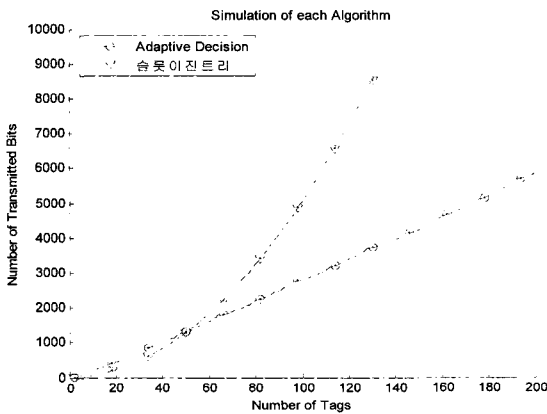


그림 4.3 슬롯 이진 트리 알고리즘과 Adaptive Decision 알고리즘의 전송 데이터량 비교
Fig. 4.3 Compared Slotted Binary Tree algorithm with Adaptive Decision algorithm.

표 4.2 알고리즘에 따른 태그별 총 전송비트수
Table 4.2 Total transmitted data.

	2tag	4tag	8tag	16tag	32tag
Adaptive Decision	16	56	152	368	769
기본 이진 검색	4	30	258	2464	26600
동적 이진 검색	2	15	129	1232	13300
비트별 이진 검색	3	20	108	544	2640
슬롯 이진 트리	4	12	32	80	192

작았다. 비교하여 보면 제안한 Adaptive Decision 알고리즘의 반복횟수가 기본 이진 검색 알고리즘보다 최대 77.8%개선되었고, 슬롯 이진 트리 알고리즘 보다 최대 16.8%개선되었다.

기존의 충돌 방지 기법과 제안한 기법의 태그 수에 따른 총 전송 데이터량을 분석하면 그림 4.2과 같고, 이를 자세히 표현하면 그림 4.3과 같다. 또 이와 같은 총 전송 데이터량을 태그의 개수에 따라 정량적으로 정리하면 표

4.2와 같다. 분석 결과, 태그를 인식하기 위한 총 전송 데이터량은 제안한 Adaptive Decision 알고리즘이 기본 이진 검색 알고리즘보다 최대 1/137배 감소되었으면 기존 알고리즘 중에서 전송 데이터량이 가장 적은 슬롯 이진 트리 알고리즘 보다는 최대 1/4배 감소되었다.

성능평가 결과, 본 논문에서 제안한 Adaptive Decision 알고리즘이 기존의 충돌방지 기법들보다 반복 횟수는 크게 감소하였으며, 총 전송 데이터량은 대부분의 기법들보다는 감소하였으나 일부 알고리즘보다는 조금 크게 나타났다. 이 두 가지를 성능을 종합하여 보면 본 논문에서 제안한 Adaptive Decision 알고리즘이 작은 데이터를 전송하면서 반복횟수를 크게 감소시켰으므로 태그 처리 시간 및 에너지 효율이 우수한 것으로 판단할 수 있었다.

V. 결 론

RFID 시스템에서 다수의 태그를 사용할 때, 태그의 신호들 간에 충돌이 발생하였다. 이러한 충돌은 RFID 시스템의 성능을 크게 저하시키는 요인이 되므로 다양한 충돌방지 기법을 이용하여 정확하게 태그를 인식하여야 한다. 본 논문에서는 이러한 충돌을 방지하기 위한 Adaptive Decision 충돌 방지 기법을 제안하였다.

기존의 ISO/IEC 18000-6 Type-B의 충돌 방지 기법으로 제안한 기본 이진 검색 알고리즘 방식, 동적 이진 검색 알고리즘 방식, 슬롯 이진 검색 알고리즘 방식과 비트별 이진 검색 알고리즘 방식들의 동작 방법과 장, 단점을 고찰하고, 문제점을 분석하였다. 고찰 결과를 바탕으로 태그를 인식하기 위한 반복횟수와 전송하는 총 데이터량을 감소하여 처리시간 및 에너지 소모량을 줄일 수 있는 방안을 모색하기 Adaptive Decision 충돌 방지 알고리즘을 제안하였다.

Adaptive Decision 충돌 방지 알고리즘은 전송되는 비트들의 '1'의 개수를 판별해서 검색하는 방식이다. 태그의 ID 비트를 호출하여 합을 계산하고 SB에 저장한다. 총 태그의 개수에 따른 각 단계에서의 선택 테이블을 작성하고, SB의 값을 검색하여 값을 선택하거나 '1'과 '0'의 그룹으로 나누고, 조건에 따른 하위 비트를 검색한다. 이와 같은 방법을 반복적으로 수행하여 리더의 영역 내의 모든 태그 ID를 식별한다.

제안한 기법의 성능을 평가하기 위해 태그의 개수에 따른 반복횟수와 그 때의 총 전송비트 수를 모의 실험하였다. 성능평가 결과, 제안한 Adaptive Decision 알고리즘

즘이 기존의 알고리즘보다 반복 횟수는 최대 77.8%, 최소 16.8% 감소시켰으며, 총 전송 데이터 수는 최대 1/137배, 1/4배 감소시켰다. 즉, 제안한 Adaptive Decision 알고리즘은 태그의 탐색시간을 크게 단축시켰고, 탐색을 위한 데이터 전송량도 감소시켰다.

본 논문에서 제안한 충돌방지 기법을 실제 RFID 시스템에 적용할 경우 무선 데이터의 전송의 신뢰성 향상과 태그 인식 시간의 단축 및 에너지 소모량을 줄일 수 있으므로 시스템의 성능향상에 크게 기여할 것으로 사료된다.

참 고 문 헌

- [1] 변상기, "RFID 태그 기술", 전자부품연구소
- [2] EPCglobal, "EPC Radio-Frequency Identity Protocols Generation 2UHF RFID Tag(Class 1) : Protocol for Communications at 860 MHz- 960MHz," Working Draft Version 1.0.4, February. 2004.
- [3] J. Waldrop, D. W. Engles, s. e. Sarma, "Colorwave : A Mac for RFID Reader Networks," in *Proc. of IEEE Conf. on Wireless Communication and Networking*, March 2003.
- [4] ISO/IEC JTC 1/SC 32/WG 4, "Information Technology automatic identification and data capture techniques-Radio frequency identification for item management - Part 6 : Parameter for air interference communications at 860MHz to 960 MHz," *ISO/IEC FDIS 18000-6*, November 2003.
- [5] European Telecommunications Standards Institute, "Elec-tromagnetic compatibility and Radio spectrum Matters(FRM) Frequency-agile Generic Sort Range Device using Listen-Before-Transmit (LBT)," *ETSI TR 102313*, Vol. 1.1.1, July. 2004.
- [6] Klaus Finkenzeller, 이근호, 강병권 역, *RFID HANDBOOK*, 영진닷컴, 2004.
- [7] 이수련, 이채우, "RFID를 위한 다중 태그 인식 알고리즘에 관한 연구", 한국전자통신연구원, 2004.
- [8] 이수련, 이채우, "RFID 시스템의 다중 인식 기술 현황", 한국전자과학회, pp.44-53, 2004.
- [9] 노경현, "RFID 시스템의 태그간 충돌방지를 위한 binary 알고리즘의 성능 분석", 경희대학교, 2005.
- [10] 최호승, 김재효, "RFID 시스템에서의 태그 인식 알고리즘 성능 분석", 대한전자공학회, 제42권 TC 제 5호, pp.47-54, May. 2005.

저 자 소 개



고 영 은(학생회원)
 2001년 단국대학교 공과대학
 전자공학전공(공학사)
 2003년 단국대학교 대학원
 전자공학전공(공학석사)
 2003년~현재 단국대학교 대학원
 전자컴퓨터공학과
 박사과정 수료

<주관심분야 : UWB, OFDM, 디지털 이동통신,
 RFID>



방 성 일(정회원)
 1984년 단국대학교 공과대학
 전자공학전공(공학사)
 1986년 단국대학교 대학원
 전자공학전공(공학석사)
 1992년 단국대학교 대학원
 전자공학전공(공학박사)

1992년~1993년 (주)대기정보통신 책임연구원
 1997년~2001년 (주)엘씨텍 연구소장
 2001년~현재 단국대학교 산학연 센터장
 1994년~현재 단국대학교 전기전자컴퓨터공학부
 교수

<주관심분야 : RF Amp, UWB, OFDM, 디지털
 이동통신, RFID >



오 경 욱(학생회원)
 1993년 단국대학교 공과대학
 전자공학전공(공학사)
 1995년 단국대학교 대학원
 전자공학전공(공학석사)
 1995년~현재 단국대학교 대학원
 전자컴퓨터공학과
 박사과정

<주관심분야 : UWB, OFDM, 디지털 이동통신,
 RFID>