

---

# RFID 시스템에서 고속 태그 식별을 위한 충돌방지 알고리즘

임 인 택\*

Anti-Collision Algorithm for Fast Tag Identification in RFID Systems

In-Taek Lim\*

---

이 논문은 2005년도 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된  
연구임(KRF-2005-041-D00675)

---

## 요 약

본 논문에서는 RFID 시스템에서 식별영역 내에 있는 태그들을 식별하기 위하여 무기억 특성을 갖는 QT 알고리즘을 개선한 QT\_ecfi 알고리즘을 제안한다. QT\_ecfi 알고리즘에서는 질의 문자열이 식별코드의 처음 비트들과 일치하는 태그는 식별코드 중에서 질의 문자열을 제외한 나머지 비트들로만 응답한다. 리더는 태그들의 응답 문자열 중에서 충돌이 발생한 비트 위치를 알 수 있으므로 충돌이 발생한 위치가 태그 식별코드의 마지막 비트이면, 리더는 더 이상의 질의가 없이 두 개의 태그를 동시에 식별할 수 있다. 한편 태그가 응답하는 동안 리더가 충돌을 감지하면 리더는 중지 신호를 방송한다. 성능 분석의 결과, 본 논문에서 제안한 QT\_ecfi 알고리즘은 QT 알고리즘에 비하여 리더의 질의 횟수와 태그의 응답 비트 수가 월등히 적음을 알 수 있었다.

## ABSTRACT

In this paper, a QT\_ecfi algorithm is proposed for identifying all the tags within the identification range. The proposed QT\_ecfi algorithm revises the QT algorithm, which has a memoryless property. In the QT\_ecfi algorithm, the tag will send the remaining bits of their identification codes when the query string matches the first bits of their identification codes. When the reader receives all the responses of the tags, it knows which bit is collided. If the collision occurs in the last bit, the reader can identify two tags simultaneously without further query. While the tags are sending their identification codes, if the reader detects a collision bit, it will send a signal to the tags to stop sending. According to the simulation results, the QT\_ecfi algorithm outperforms the QT algorithm in terms of the number of queries and the number of response bits.

## 키워드

RFID, Anti-collision algorithm, QT protocol

## I. 서 론

RFID 시스템에서 리더는 무선채널을 통하여 각각의

태그들과 통신하는데, 모든 태그들은 리더가 보낸 신호를 동시에 듣게 된다[1][2]. 하나의 리더로부터 요청 메시지를 받은 태그들은 동시에 리더로 자신의 식별코드

를 전송하기 때문에 태그 충돌(Tag collision)이 발생한다. 이 때 하나의 리더가 동시에 응답한 여러 개의 태그를 식별해야하는 문제가 발생하는데 이를 해결하는 기술이 충돌방지(Anti-collision) 알고리즘이며, 이는 RFID 시스템에서 가장 핵심이 되는 기술이다[3][4].

RFID 시스템 성능의 두 가지 중요한 척도는 태그를 식별하기 위해 필요한 시간과 태그에 의해 소모되는 에너지이다. 리더에 의해 전송요구가 있을 때 태그가 보낸 데이터가 적을수록 태그식별 시간이 단축되고 태그에 의해 소모되는 에너지도 적다. 따라서 충돌방지 알고리즘은 RFID 시스템의 태그식별 속도 및 정확성을 결정하는 중요한 핵심기술이다[5].

RFID 시스템에서 태그 충돌을 해결하기 위한 방법은 크게 태그-주도 방식과 리더-주도 방식으로 나눌 수 있다. 태그-주도 방식은 리더가 데이터 전달을 제어할 수 없기 때문에 비동기적으로 동작한다. 따라서 태그-주도 방식은 인식속도가 느린 단점이 있어서 대부분의 적용분야에서 리더-주도 방식을 사용한다. 리더-주도 방식은 모든 태그의 응답이 리더에 의해 동시에 제어되어 동기적으로 동작하기 때문에 대부분의 적용분야에서 이 방식을 사용한다[6][7].

리더-주도 방식의 태그 식별 알고리즘 중에서 QT(Query-Tree) 알고리즘은 Auto-ID센터에서 제안한 무기억(Memoryless) 알고리즘으로서 모든 태그들은 태그 식별과정에서 일어난 이전 질의의 이력을 기억할 필요가 없다[7][8]. 리더의 질의에 대한 응답을 하기 위하여 질의할 때마다 태그가 응답할 식별코드의 비트 위치를 기억하는 알고리즘과는 달리 무기억 알고리즘에서는 매 질의마다 몇 비트의 프리픽스를 전송한다. QT 알고리즘인 경우, 리더가 질의한 프리픽스가 자신의 식별코드 중에서 처음 몇 비트들과 일치하는 태그는 전체의 식별코드로 응답한다. 또한 태그들로부터 식별코드를 수신하는 동안에 충돌이 발생하여도 이를 태그들에게 알리는 방법이 없으므로 태그는 자신이 응답한 식별코드가 충돌이 발생하였는지를 알 수 없다. 이로 인하여 태그가 전송하는 비트의 수가 많은 문제점이 있다. 따라서 본 논문에서는 이를 개선하여 식별코드 중에서 프리픽스를 제외한 나머지 비트들만으로 응답하고, 리더가 식별코드를 수신하는 도중에 충돌을 감지하면 태그들에게 전송을 중지하도록 하는 신호를 보내는 QT\_ecfi (QT for energy conserving and fast identification) 알고리즘을 제안

한다.

본 논문의 구성은 다음과 같다. 서론에 이어 II장에서는 고속의 태그식별을 위하여 본 논문에서 제안하는 QT\_ecfi 알고리즘을 설명하고, III장에서는 이들 알고리즘의 성능분석 결과를 기술하고, 마지막으로 결론을 맺는다.

## II. 제안하는 QT\_ecfi 프로토콜

본 논문에서 제안하는 QT\_ecfi 알고리즘인 경우, 리더는 태그의 응답 문자열 중에서 충돌이 발생한 비트 위치를 알 수 있다고 가정한다. QT\_ecfi 알고리즘에서는 QT 알고리즘에서와 같이 매 질의마다 몇 비트로 구성된 질의 문자열 프리픽스를 전송한다. 수신한 질의 문자열이 자신의 식별코드 중에서 처음 비트들과 일치하는 태그는 자신의 전체 식별코드를 전송하는 대신에 식별코드 중에서 질의 문자열에 해당하는 프리픽스를 제외한 나머지 문자열로 응답하고 리더로부터 다음의 질의를 기다린다. 반면 일치하지 않는 태그는 STAND-BY 상태로 천이하여 하나의 태그가 완전히 식별되어 다음 사이클이 시작될 때까지 리더의 질의에 응답하지 않는다.

전송한 질의 문자열에 대하여 둘 이상의 태그가 응답하면 충돌이 발생한다. 이 경우 리더는 이전의 질의 문자열에 충돌이 아닌 응답 문자열과 비트 '0', '1'을 각각 연장하여 새로운 질의 문자열을 구성하여 다음 단계를 반복한다. 태그로부터 응답을 수신하는 도중에 충돌을 감지하면 리더는 중지 신호를 태그들에게 방송한다. 또한 식별코드 중에서 마지막 비트에서 충돌이 발생한 경우, 리더는 마지막 비트만 다른 두 개의 태그가 식별영역 내에 존재한다는 것을 알 수 있으므로 두 개의 태그를 동시에 식별할 수 있다. 반면 리더가 충돌을 감지하지 않으면 하나의 태그를 완전히 식별한 경우이다. 이 경우, 리더는 질의 문자열을 갱신하고 다음 태그를 식별하기 위하여 위의 사이클을 반복한다.

QT\_ecfi 알고리즘의 동작은 다음과 같다. 리더의 질의 문자열을  $q(q \in A)$  정의하고,  $|q|$ 를 질의 문자열의 길이로 정의한다.  $w$ 를 태그의 식별코드 문자열이라 하고,  $r$ 를 태그의 응답 문자열이라 정의하면 태그의 응답 문자열은 태그의 식별코드 중에서 질의 문자열을 제외한 나머지 비트들로 구성된 문자열이므로 다음과 같이 정의된다.

$$r = r_{|q|+1} r_{|q|+2} \dots r_k \quad (1)$$

먼저 리더의 알고리즘은 다음과 같다.

- 1) 큐를  $Q = \langle q_1, q_2, \dots, q_l \rangle$  로 둔다.
- 2) 질의  $q_1$ 을 큐에서 꺼내서 방송한다.
- 3) 큐를  $Q = \langle q_2, \dots, q_l \rangle$ 로 갱신한다.
- 4) 태그로부터 수신한 응답에 대하여,
  - ① 태그로부터의 응답  $r$  중에서 충돌이 아닌 문자열을  $z$ 라 하고, 문자열  $z$ 의 길이를  $|z|$ 라 한다.
  - ②  $|z| = k - |q|$ , 즉 충돌이 아니면 태그의 식별코드  $q_1 r$ 을 식별한 것으로 표시하고 메모리에 삽입한다.
  - ③  $|z| = k - |q| - 1$ , 즉 식별코드의 마지막 비트에서 충돌이면 태그의 식별코드  $q_1 r_{|q|+1} r_{|q|+2} \dots r_{k-1} 0$ 과  $q_1 r_{|q|+1} r_{|q|+2} \dots r_{k-1} 1$ 을 식별한 것으로 표시하고 메모리에 삽입한다.
  - ④ 응답을 수신하는 도중에 충돌이면, 중지 신호를 방송하고, 큐를  $Q = \langle q_2, \dots, q_l, q_1 z 0, q_1 z 1 \rangle$ 로 갱신한다.
- 5) 큐  $Q$ 가 빌 때까지 위의 과정을 반복한다.

태그의 알고리즘은 다음과 같다.

- 1) 태그의 식별코드  $w$ 를  $w = w_1 w_2 \dots w_k$ 라 한다.
- 2) 리더로부터 수신한 질의를  $q$ 라 하고,  $q$ 의 길이를  $|q|$ 라 한다.
- 3)  $q = \varepsilon$  또는  $q = w_1 w_2 \dots w_{|q|}$ 이면, 태그는  $r = r_{|q|+1} r_{|q|+2} \dots r_k$ 로 응답한다. 즉 질의  $q$ 가 자신의 식별코드 처음 비트들 일치하면 식별코드 중에서 질의 문자열을 제외한 나머지 비트들을 리더로 보낸다.
- 4) 응답하는 도중에 전송 신호를 수신하면 전송을 중단한다.

그림 1은 위의 태그 알고리즘을 기반으로 하는 QT\_ecfi 알고리즘에서 태그의 상태 천이도를 나타낸 것이다. STAND-BY 상태에 있는 태그가 리더로부터 질의 문자열을 수신하고, 수신한 질의 문자열이 자신의 식별코드 처음 비트들과 일치하는 태그는 SENDING 상태로 천이한다. SENDING 상태에 있는 태그는 QT 알고리즘

과는 달리 자신의 식별코드 중에서 질의 문자열을 제외한 나머지 비트들만으로 응답한다. SENDING 상태에 있는 태그가 리더로부터 중지 신호를 수신하는 경우에는 즉시 전송을 중지하고 STAND-BY 상태로 천이한다. 한편 태그가 응답하는 도중에 중지 신호를 감지하지 못하면 식별코드의 전체 비트들을 전송한 후 STAND-BY 상태로 천이한다.

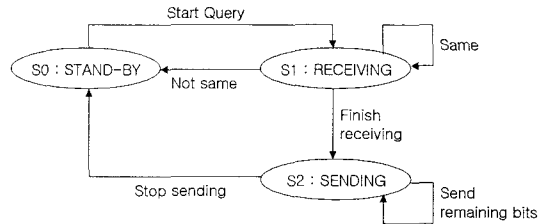


그림 1. QT\_ecfi 알고리즘의 태그 상태 천이도  
Fig. 1. State transition diagram of QT\_ecfi algorithm.

그림 2는 태그의 식별코드가 <0001, 0010, 1010, 1011>인 4개의 태그를 식별하기 위한 QT\_ecfi 알고리즘의 동작을 나타낸 것이다. 그림에서 나타낸 바와 같이 4개의 태그를 모두 식별하기 위하여 총 5번의 질의의 사이클이 필요하다. 이때 리더가 방송하는 질의 문자열의 총 비트 수는 8 비트이고, 모든 태그들이 전송하는 총 비트의 수는 16 비트이다.

Query string	Tag answers	Answered tags	Updated Queue
$\varepsilon$	X	all	$\langle 0, 1 \rangle$
0	0X	0001, 0010	$\langle 1, 000, 001 \rangle$
1	01X	1010, 1011	$\langle 000, 001 \rangle$
000	1	0001	$\langle 001 \rangle$
001	0	0010	$\langle \varepsilon \rangle$

그림 2. QT\_ecfi 알고리즘의 동작  
Fig. 2. Operation of QT\_ecfi algorithm.

### III. 성능 평가

본 논문에서는 시뮬레이션을 통하여 QT 알고리즘과 QT\_ecfi 알고리즘의 성능을 비교하였다. 시뮬레이션을 위하여 태그의 식별코드 길이는 64비트로 가정하였다. 태그의 식별코드가 무작위인 경우와 순차적인 경우 각

각에 대하여 식별영역 내의 모든 태그를 식별하기 위하여 태그의 수에 따른 리더의 질의 횟수 및 모든 태그가 보낸 비트 수를 성능평가 매개변수로 하였다. 식별코드가 순차적인 경우는 식별코드의 최하위 비트부터 순차적으로 증가하는 것으로 가정하였다.

그림 3과 4는 태그의 식별코드를 무작위로 한 경우와 순차적으로 한 경우, 모든 태그를 식별하기 위한 총 질의 횟수를 각각 나타낸 것이다. 태그의 식별코드가 무작위인 경우, 100개의 태그를 식별하기 위하여 QT 알고리즘은 299번의 질의를 필요로 하고, QT\_ecfi 알고리즘은 199번의 질의를 필요로 한다. 따라서 본 논문에서 제안한 QT\_ecfi 알고리즘이 QT 알고리즘에 비하여 월등히 우수함을 알 수 있다. 이는 QT\_ecfi 알고리즘인 경우, 충돌이 발생한 비트의 위치를 리더가 알 수 있기 때문이다. 또한 태그의 식별코드가 순차적인 경우, 100개의 태그를 식별하기 위하여 QT 알고리즘 319번의 질의를 필요로 하고, QT\_ecfi 알고리즘은 99번의 질의를 필요로 하므로 약 3.2배의 성능 향상을 얻을 수 있었다.

그림 5와 6은 태그의 식별코드를 무작위로 한 경우와 순차적으로 한 경우, 모든 태그를 식별할 때까지 태그들이 응답한 총 비트의 수를 각각 나타낸 것이다. 태그의 식별코드가 무작위이든 순차적이든 상관없이 본 논문에서 제안한 QT\_ecfi 알고리즘이 QT 알고리즘에 비하여 태그가 응답하는 비트의 수가 적다. 이는 본 논문에서 제안한 QT\_ecfi 알고리즘인 경우, 식별코드 중에서 질의 문자열을 제외한 나머지 비트들로만 응답하고, 응답을 전송하는 도중에 충돌이 발생하여 리더로부터 중지 신호를 수신하면 전송을 중지하기 때문이다. 태그가 전송하는 총 비트의 수는 QT 알고리즘인 경우 본 논문에서 제안한 QT\_ecfi 알고리즘에 비하여 태그의 식별코드가 무작위인 경우와 순차적인 경우 각각 약 7.9배와 약 59 배 더 많은 비트들을 전송하여야 한다. 이는 리더는 충돌이 발생한 비트의 위치를 알 수 있어서 마지막 비트에서 충돌이 발생한 경우 리더는 두 개의 태그를 동시에 식별할 수 있고, 응답을 수신하는 도중에 충돌을 감지하면 태그로 하여금 전송을 중지시키기 때문이다. 따라서 본 논문에서

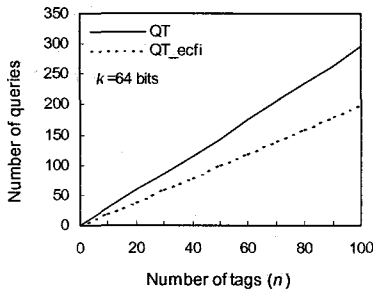


그림 3. 총 질의 횟수(무작위 식별코드)  
Fig. 3. Total number of queries(random ID).

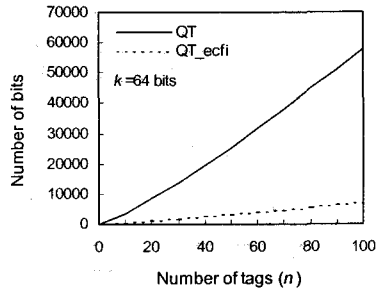


그림 5. 총 응답 비트의 수(무작위 식별코드)  
Fig. 5. Total number of response bits(random ID).

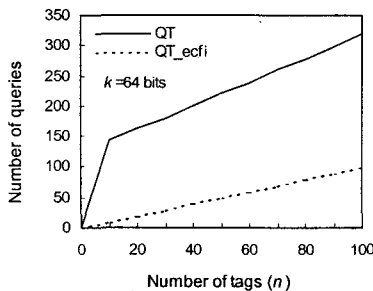


그림 4. 총 질의 횟수(순차적 식별코드)  
Fig. 4. Total number of queries(sequential ID).

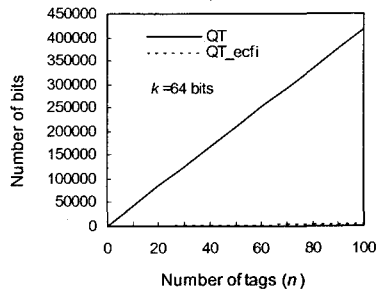


그림 6. 총 응답 비트의 수(순차적 식별코드)  
Fig. 6. Total number of response bits(sequential ID).

서 제안한 QT\_ecfi 알고리즘이 QT 알고리즘에 비하여 리더의 식별영역 내에 있는 모든 태그를 식별하기 위한 시간이 단축되고, 태그의 전력 소모량도 적음을 알 수 있다.

Auto-ID 센터의 클래스 0 RFID 시스템에서는 세 가지 형태의 64비트와 한 가지 형태의 96비트, 및 세 가지 형태의 256비트를 갖는 EPC (Electronic Product Code)를 정의하고 있다[9]. 식별코드의 길이가 길면 길수록 태그들이 응답해야 하는 비트의 수는 많아질 것으로 예상된다. 따라서 본 논문에서는 식별코드의 길이가 AutoID 센터에서 정의한 64, 96, 및 256비트일 때 태그 식별 알고리즘의 성능을 분석하였다.

표 1은 식별영역 내의 태그 수가 100개일 때, 식별코드의 길이에 따른 질의 횟수를 나타낸 것이다. 표에서 나타낸 바와 같이 식별코드가 무작위인 경우에는 식별코드 길이에 상관없이 두 알고리즘 모두 일정한 질의 횟수를 나타낸다. 또한 식별코드가 순차적인 경우 본 논문에서 제안한 QT\_ecfi 알고리즘은 길이에 관계없이 질의 횟수는 일정한 반면, QT 알고리즘의 경우, 길이가 64비트에서 256비트로 증가하면 약 2.2배의 질의 횟수를 더 필요로 한다.

표 2는 식별코드의 길이에 따른 태그들의 총 응답 비트 수를 나타낸 것이다. 표에서 나타낸 바와 같이 QT 알고리즘인 경우, 식별코드의 길이가 64비트에서 256비트로 증가할 때 응답 비트의 수는 무작위인 경우와 순차적인 경우에 각각 38.5배와 15.8배로 증가한다. 반면, 본 논문에서 제안한 QT\_ecfi 알고리즘은 무작위인 경우와 순차적인 경우 모두 약 3.7배만 증가한다. 따라서 식별영역 내에 있는 태그들의 식별코드가 무작위이든 순차적이든 상관없이 모든 RFID 응용에서는 본 논문에서 제안한 QT\_ecfi 알고리즘이 QT 알고리즘에 비하여 월등히 우수한 성능을 나타낼 수 있음을 알 수 있다.

표 1. 식별코드 길이에 따른 총 질의 횟수  
Table 1. Total number of queries vs. ID length.

길이(비트)	무작위 식별코드		순차적 식별코드	
	QT	QT_ecfi	QT	QT_ecfi
64	299	199	319	99
96	299	199	383	99
256	299	199	703	99

표 2. 식별코드 길이에 따른 총 응답 비트 수  
Table 2. Total number of response bits vs. ID length.

길이(비트)	무작위 식별코드		순차적 식별코드	
	QT	QT_ecfi	QT	QT_ecfi
64	57,929	7,095	416,000	7,088
96	85,906	10,294	931,200	10,288
256	229,193	26,305	6,579,200	26,288

#### IV. 결론

본 논문에서는 리더-주도 방식의 무기억 태그 식별 알고리즘인 QT 알고리즘을 개선한 QT\_ecfi 알고리즘을 제안하고, 이에 대한 성능을 분석하였다. 제안한 QT\_ecfi 알고리즘에서는 질의 문자열이 식별코드의 처음 비트들과 일치하는 태그는 식별코드 중에서 질의 문자열을 제외한 나머지 비트들로만 응답한다. 또한 리더가 충돌을 감지하면 중지 신호를 태그들에게 방송하고, 이를 감지한 태그들은 즉시 전송을 중지한다. 리더는 태그들의 응답 문자열 중에서 충돌이 발생한 비트 위치를 알 수 있다. 태그들로부터 수신한 응답 문자열에서 충돌이 발생했고, 충돌이 발생한 위치가 태그 식별코드의 마지막 비트이면 리더는 두 개의 태그를 동시에 식별할 수 있다. 이러한 QT\_ecfi 알고리즘의 특성으로 인하여 태그들이 전송하는 비트의 수와 리더의 질의 횟수가 QT 알고리즘에 비하여 월등히 적음을 시뮬레이션을 통하여 알 수 있었다.

참고문헌

[1] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, Carl Hanser Verlag GmbH & Co., 2002.

[2] H. Vogt, "Efficient Object Identification with Passive RFID Tags," *First International Conf. on Pervasive Computing, LNCS*, vol.2414, pp.99-113, Springer-Verlag, 2002.

[3] I. Papadimitriou, and M. Paterakis, "Energy-Conserving Access Protocol for Transmitting Data in Unicast and Broadcast Mode," *Proc. IEEE PIMRC2000*, vol.2, pp.984-988, Sept. 2000.

[4] I. Chlamtac, C. Petrioli, and J. Redi "Energy-Conserving Access Protocols for Identification Networks", *IEEE/ACM Trans. Networking*, vol.7, no.1, pp.51-59, Feb. 1999.

[5] S. E. Sarma, S. A. Weis, and D. W. Engels, "RFID Systems and Security and Privacy Implications," *CHES2002, LNCS*, vol.2523, pp.454-469, 2003.

[6] M. Jacomet, A. Ehram, and U. Gehrigm "Contactless Identification Device with Anticollision Algorithm," *Proc. IEEE CSCC'99*, Athenes Italy, July 1999.

[7] C. Law, L. Lee, and K. Y. Siu, "Efficient Memoryless Protocol for Tag Identification," Auto-ID Center, *MIT-AUTOID-TR-003*, Oct. 2000.

[8] 임인택, 최진오, "RFID 시스템에서 태그 식별을 위한 개선된 QT 프로토콜," 한국해양정보통신학회 논문지, 제10권, 제3호, pp.430-436, 2006.3.

[9] Auto-ID Center, "860MHz-930MHz Class 0 Radio Frequency Identification Tag Protocol Specification Candidate Recommendation, Version 1.0.0," June 2003.

저자소개

임 인 택(In-Taek Lim)



1984년 2월 울산대학교 전자계산학과 (공학사)

1986년 2월 서울대학교 계산통계학과 (이학석사)

1998년 2월 울산대학교 컴퓨터공학과 (공학박사)

1986년 1월~1993년 2월 : 삼성전자(주) 특수연구소 선임 연구원

1993년 3월~1998년 2월 : 동부산대학 전자계산과 조교수

1998년 3월~현재 : 부산외국어대학교 컴퓨터공학부 부 교수

※관심분야 : RFID, 이동통신, MAC 프로토콜