Novel Query Tree Algorithm Based on Reservation and Time-Divided Responses to Support Efficient Anti-Collision Protocol

Ji-Hwan Choi
Networks Business
Samsung Electronics
Suwon-si, 16677, Korea

Hyuckjae Lee
School of Electrical Engineering
KAIST
Daejeon, 34141, Korea

***Abstract*—To support IoT ecosystem, the closest technology to date is RFID sensor networks. In this paper, we propose an outstanding algorithm of anti-collision protocol to improve the readability of multiple tags in RFID systems. The proposed reservation and time-divided responses based query tree algorithm guarantees 100% readability and faster tag identification performance. As a result by simulating based on the system in the EPC Class 1 Gen. 2, the performance of the proposed scheme is significantly improved compared to that of the enhanced schemes based on query tree algorithm.**

***Index Terms*—RFID, Anti-collision, tag collisions, query tree algorithm, reservation based protocol, IoT.**

I. Introduction

In terms of the fourth industrial revolution, the realization and collaboration of the fifth generation mobile networks (5G) and internet of things (IoT) is the biggest trends [1]. The combination of the 5G represented by massive machine type communications and IoT ecosystem will creates a number of new business areas, from simple inventory management to manufacturing management and autonomous vehicles [2]. In order to build a platform to support such businesses, it is essential not only to support contactless automatic identification but also to measure, collect and update various fundamental information, and therefore the closest technology to date is radio frequency identification (RFID) sensor networks [2], [3].

Basically, an RFID system consists of a reader that collects information from tags in its coverage and many tags that responds its information to the reader either passively or actively [4]. For operating the RFID systems, the reader transmits the request message to periodically collect at least the unique identifier, i.e., electronic product code (EPC) [5], of each tag. In response to the reader's request, the tags transmit their EPC to the reader. In that case, collision happens when some of tags simultaneously transmit their EPCs to the reader, and then the reader cannot identify any of these tags. To overcome this collision problem, the deterministic and probabilistic algorithms of various anti-collision protocols have been devised [4].

In the deterministic algorithms, the reader is theoretically able to recognize all tags after resolving a collision by iteratively splitting collided tags into subgroups. Note that, in collecting the EPC, a group of EPCs is divided into two lower groups having either '0' and '1' according to the bit corresponding to the collision location, that is, the collision bit [6], [7]. However, this approach solicits relatively long identification latency until tags are recognized because the time for identification increases as the number of tags increases. In the probabilistic algorithms, time is divided into slots, and collisions are avoided by sequentially identifying the tags in each slot by the reader. However, the performance of this approach is constrained by the computational overhead of determining the total number of slots and the scheduling of tags in each slot. Also, many approaches including collision-tolerant dynamic frame slotted Aloha using Walsh codes have been proposed, it does not guarantee 100% of the probability for tag identification [8]–[10].

After considering the advantages and disadvantages of two approaches, anti-collision algorithms using Manchester code adopting query tree based slot-reservation have been proposed [11], [12]. The reader collects the EPCs sequentially using the probabilistic approach, which is a dynamic frame slotted Aloha (DFSA), after a fast tag recognition using the deterministic approach, which is a query tree algorithm (QTA), based on the temporary identifier (TId) of tags. At this time, Manchester code is used identify where the collision bits are located. The TId is improved from 16-bit random number (RN16) to enhanced RN16 for increasing its randomness by exclusive OR operation (XOR-ing) between RN16 and 16-bit tuple of reverse EPC [11]. However, despite this methodological approach, there is demand for faster tag identification process as well as a solution to a missing or undetected tag during the tag identification process.

This paper proposes a novel reservation and time-divided responses based query tree algorithm (RTQTA) to support efficient anti-collision protocol, which is described in detail in Section III. The differentiation of the proposed RTQTA is summarized as follows. First, in order to reduce the additional bits to be recognized caused by using the RN16, the proposed scheme directly performs XOR-ing using the first N bits of reverse EPC, where N is 16. Second, in the process of tag

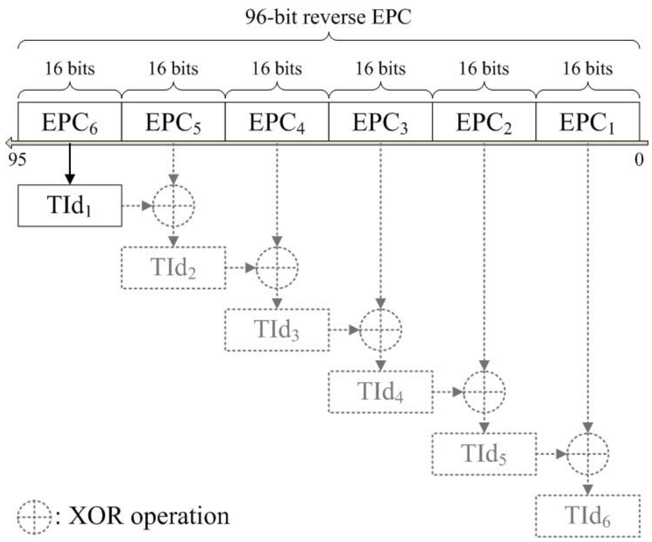


Fig. 1: Generation of TId_{iFlag}, $1 \leq iFlag \leq 6$, using reverse EPC.

recognition, the proposed scheme adopts that the tag responds into time-divided two sub-slots depending on the value of the collision bit, respectively [7]. Then, the delay and iteration for the tag recognition step is reduced due to the response of two tags with a time difference according to one prefix requested from the reader. Third, in the process of post-processing for collecting EPCs, we propose a two-bit acknowledgment (TBACK) that adaptively requests a response according to the result of two sub-slots divided in the previous process. The value of the TBACK transmitted from the reader indicates whether one subslot or two subslots are used for the response of the EPCs. Also, the response to one TId is reserved by each bit having '1', accordingly the delay and iteration for the post-processing step is reduced. Finally, it provides 100% readability by using a detected flag ($dFlag$) and iteration flag ($iFlag$) inside of a tag for the tags that are missing, undetected, or newly entered to the reader's coverage during the tag identification process. Thus, the proposed RTQTA scheme reduces the latency due to the tag identification process, and increases reliability of the tag identification process.

The remainder of this paper is organized as below. In Section II, the brief review of the previous works is illustrated. In Section III, we describe the proposed RTQTA. Finally, in Section IV, the performance of the proposed scheme is analyzed and finally this article is concluded.

II. Brief review of the previous works

The basic QTA is based on a query from a reader and responses from multiple tags. After the reader transmits a prefix from its queue, each tag in its coverage transmits the remaining parts of its EPC to the reader when the prefix matches to the first part of the EPC [11]. The prefix is stored at the queue of the reader after constructed by bit splitting of '0' and '1' for the collision bit occurred from the responses of tags. Since the QTA performs bit splitting only for the collision bits, its performance is significantly improved compared to the tree working algorithm [6] that simply search for all cases sequentially for all bits. However, in the case of such QTA, since the long EPC is directly used, the number of bits used in the query-response is large, so the performance improvement is limited.

Therefore, there has been an effort to improve performance using TIds without using EPCs directly. Each tag generates a TId, which is shorter than the length of its EPC, in response to the '$null$' prefix of the reader. Then, the reader recognizes the tags by performing the QTA with these TIds, and finally confirms each EPC by using the DFSA based on a detected TId (dTId), sequentially [11]. At this time, by using shortened length of TIds, it is probable that each TId is not unique when there are many tags. To overcome this problem, when a collision occurs when confirming a EPC by using a dTId, a method of distinguishing tags having the same TId by generating new TIds to the tags is added [11]. To generate new TIds, XOR-ing between RN16 and 16-bit tuple of reverse EPC is performed. As a result of these efforts, the performance of the enhanced QTA (i.e., xRN16QTA) is maintained for more tags.

III. Description of The Proposed RTQTA

To support faster tag identification of many tags, this paper attempts to improve the performance of previous xRN16QTA.

1) TId generation: in order to eliminate the redundancy caused by the use of TId, the first 16 bits in a reverse EPC of a tag are used to the first TIds of the tag as shown in Fig. 1. The last 36 bits of the EPC are the serial number of a product [5], so the last 16 bits of the serial number is equivalent to RN16 in terms of randomness, whether consecutive serial number or random serial number of a product. Particularly, in terms of logistics distribution, products with consecutive serial numbers are stored, so the collisions are likely to be at the beginning of the prefix, which can lead to additional performance improvements. This allows to use the last part of the EPC for XOR-ing instead of the RN16, eliminating the need to transmit additional 16 bits.

2) Tag recognition step: in terms of query-response in the QTA, all first bit of collision bits leads to split the bit into '0' and '1' for the next prefix. After the collision occurs at the n_{th} bit, a pair of prefixes is the same except for the n_{th} bit when the prefix of length n is used [7]. Accordingly, by branching the slots used in the tag recognition step of the proposed RTQTA to a pair of subslots sequentially indicating '0' and '1' in response to the $n-1$ bits prefix, the results branched at the n_{th} bit is simultaneously transmitted as shown in Fig. 2. Then, the delay and iteration for tag recognition is reduced due to the response of two tags with a time difference according to one prefix requested from the reader.

3) Post-processing step: because of using of branching as a pair of subslots according to the value of n_{th} bit in the previous tag recognition step, the post-processing step to receive the remaining EPC of a tag except for the 16 bits used as a TId is modified as well. Otherwise, it should adjust the interval

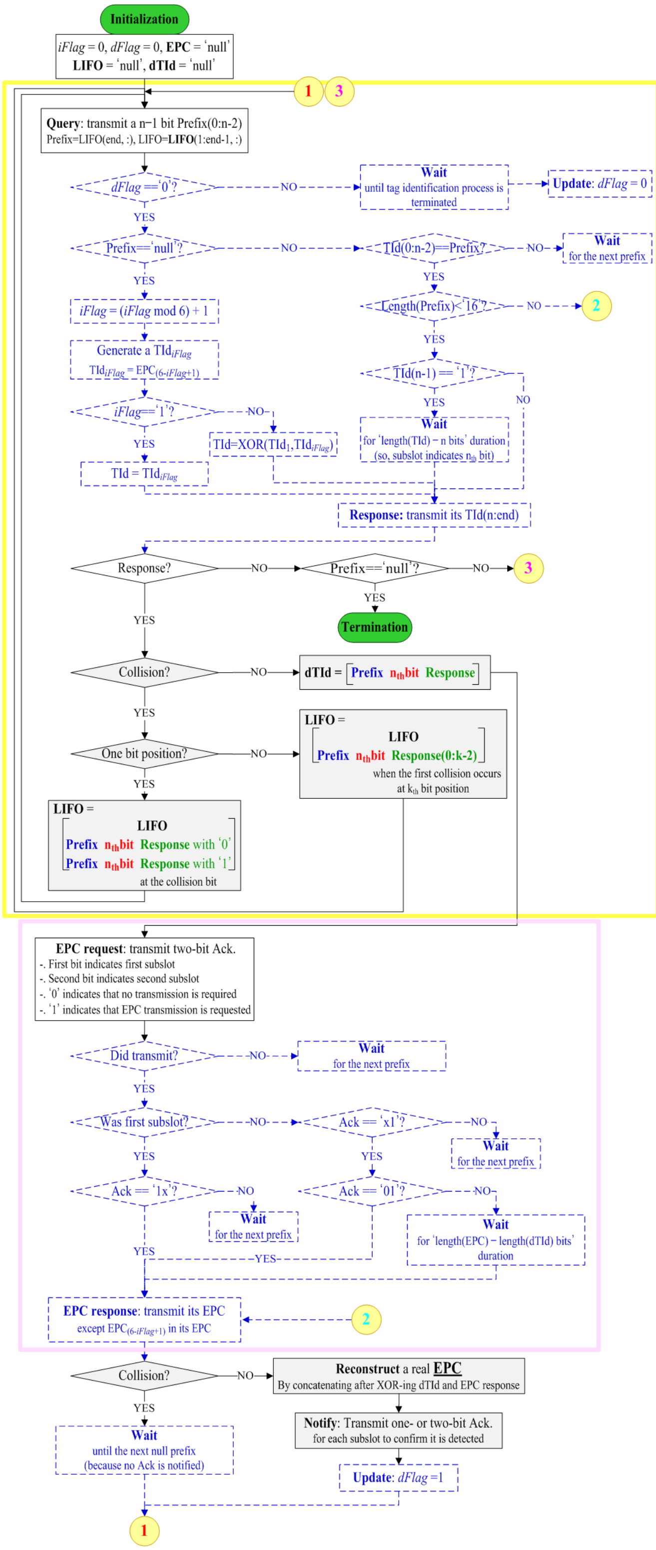


Fig. 2: The flow chart of the proposed RTQTA (solid line: operations in a reader, dotted line: operations in each tag, grey box: operating sequentially for each subslot, first bold-line box: tag recognition step, and second bold-line box: post-processing step).

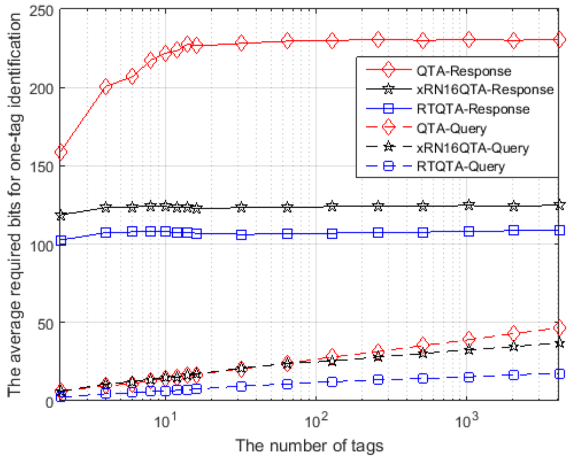
between subslots in order to transmit an acknowledgment (ACK) to request the remaining EPC transmission from a tag, and is difficult. TBACK is adopted to resolve this problem as shown in Fig. 2. After receiving one slot, i.e., a pair of two subslots, the reader transmits an ACK consisting of two bits, each of which has the following meaning. The first bit of TBACK indicates the first subslot, and the second bit indicates the second subslot. The value of '0' indicates that no transmission is required, and that of '1' indicates that EPC transmission is requested. Therefore, '00' means that transmission for the remaining EPC is not requested, so it is not necessary to use but able to be used for reliability. Either '01' or '10' means to use only one subslot, so the other subslot is excluded in time-domain. '11' means that the corresponding tags are transmitted in both subslots. Accordingly the delay and iteration for the post-processing is reduced because the response to each TId is reserved by each bit having '1'.

4) Handling of missing, undetected, or newly entered tags: to support 100% readability, anti-collision protocols should enhance its performance to cover tags that are missing, undetected, or newly entered to the reader's coverage during the tag identification process. To cover these tags, the proposed RTQTA adopts two flags, which is the $dFlag$ and $iFlag$, inside of a tag as shown in Fig. 2. The $iFlag$ is incremented by 1 in response to the '$null$' prefix, and an additional modulo operation is added because a EPC is divided into six equal parts by 16 bits. The tag transmits its TId in response to the '$null$' prefix transmitted from a reader. At this time, the $iFlag$ is used to generate a TId using the EPC of a tag. When generating the TId, the reverse EPC is divided into 16 bits, as shown in Fig. 1, and these are mapped to TIds in order. That is, $\text{EPC}_{(6-if+1)} = \text{TId}_{iFlag}$. Therefore, the collision caused by having more than one tag having the same TId in the post-processing step is avoided and the randomness of concatenated TId_{iFlag} is ensured. Thus, the proposed RTQTA scheme increases reliability of the tag identification process.

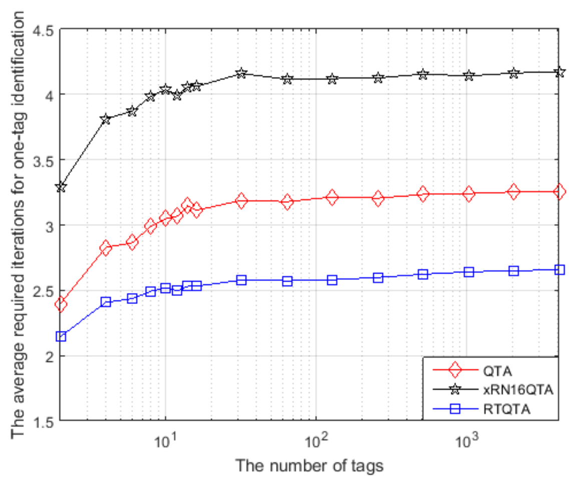
The dFlag indicates that a tag has been detected by a reader in the current tag identification process and the activity of the corresponding tag is terminated. If the reader transmits either one- or two-bit ACK having the number of bits equal to the number of value '1' in TBACK, the tag checks the subslot in the ACK corresponding to the subslot for the response of remaining EPC transmitted. If it is '1', the tag is confirmed that the tag is detected and terminates the current tag identification process. If it is '0', a tag having the same TId as itself exists, and therefore, new TId is regenerated in the next '$null$' prefix. Because the tags that are missing, undetected, or newly entered are covered by the transmission of additional '$null$' prefixes, the tag recognition rate is improved to 100% through the proposed RTQTA.

IV. Performance Analysis

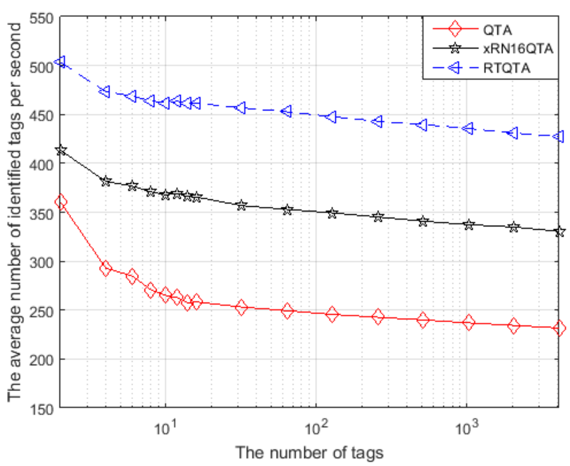
This section compares the performance of the proposed RTQTA with that of the xRN16QTA and QTA as shown in Fig. 3. For fair comparison, the number of bits of the prefixes for the query, the number of bits of the responses, and the



(a) Average required bits of query and response for one tag identification.



(b) Average required iteration for one tag identification.



(c) Average identified tags per second.

Fig. 3: Performance comparison of various anti-collision schemes including the QTA, xRN16QTA, and proposed RTQTA.

number of iterations for the query-response procedure are averaged by the number of tags. The simulation is performed with a probability of 0.25% for any missing, undetected, or newly entered tags, which is randomly included in the tag identification process, besides the collision due to the same TId. This is a factor that interferes with the fast tag identification in terms of many tags along with a collision that occurs in the post-processing step because two or more tags have the same TId. It is already included in the results, but this paper does not mention it separately in each individual explanation. In addition, the delay factors including the data rate and slot configuration parameters in EPC Class 1 Gen. 2 are adopted for estimating the average identified tag per second in the Monte Carlo simulation.

Fig. 3a shows the average required bits of query and response for one tag identification. In the case of the QTA, it is observed that the used bits in both the query and response is significantly more than those of the other schemes because of using long EPC directly. On the other hands, xRN16QTA and RTQTA use 16-bit TIds to reduce the required bits for both the query and response. Also, in the case of the RTQTA, it reduces more bits used in both the query and response by adopting a pair of subslots indicating the collision bit, and significantly reduces the bits used in the response by directly using a part of the EPC to the first TId.

Fig. 3b shows the average required iteration between the query and response to obtain the EPC from a tag. The use of a pair of subslots representing the collision bit leads to response by two subgroups corresponding to one prefix, thereby the number of iterations required for the tag identification process is significantly reduced in the RTQTA. Note that this paper is focused on the performance improvement to ensure 100% tag readability, so the use of ACKs is also counted.

As a result by simulating based on the system in the EPC Class 1 Gen. 2, the performance of the proposed RTQTA is greatly improved compared to the previous enhancements on the QTA as shown in Fig. 3c.

This paper aims to efficiently support the anti-collision protocol to improve the performance of the RFID tag identification process. We propose an efficient anti-collision algorithm through the QTA-based reservation using EPC directly and the time-divided responses. As a result, it is confirmed that the proposed RTQTA is a good alternative algorithm that provides not only stable tag identification rate of 100 $\%$ but also improved tag identification speed of 30 $\%$ or more.

References

[1] K. Tsang *et al.*, "Guest Editorial Industrial Wireless Networks: Applications, Chanllenges, and Future Directions," *IEEE Trans. Ind. Inf.* , vol. 12, iss. 2, pp.755–757, Mar. 2016.

[2] L. Xu *et al.*, "Internet of Things in Industries: A Survey," *IEEE Trans. Ind. Inf.* , vol. 10, iss. 4, pp.2233–2243, Jan. 2014.

[3] H. Zuerner *et al.*, "The Internet of Things as Greenfiend Model: A Categorization Attempt for Labeling Smart Devices," in *2014 IEEE World Forum on Internet of Things (WF-IoT)*, Mar. 2014.

[4] D. Klair *et al.*, "A Survey and Tutorial of RFID Anti-Collision Protocols," *IEEE Commun. Serveys Tuts.*, vol. 12, iss. 3, pp. 400–421, 3rd Quart. 2010.

[5] *EPC Tag Data Standard*, GS1 Std., Rel. 1.10, Mar. 2017

[6] D. Hush *et al.*, "Analysis of Tree Algorithms for RFID Arbitration," in *Proc. IEEE Inter. Symp. on Information Theory*, pp. 107, Aug. 1998.
[7] J. Choi *et al.*, "Bi-slotted Tree based Anti-collision Protocols for Fast Tag Identification in RFID systems," *IEEE Commun. Lett.*, vol. 10, iss. 12, pp. 861–863, Dec. 2006.
[8] J. Su *et al.*, "An Effective Frame Breaking Policy for Dynamic Framed Slotted ALOHA in RFID," *IEEE Commun. Lett.*, vol. 20, iss. 4, pp. 692–695, Apr. 2016.
[9] H. Ahmed *et al.*, "Backwards Compatible Improvement of the EPCglobal Class 1 Gen 2 Standard," in *Proc. Int. Conf. RFID*, pp. 114–119, Sep. 2015.
[10] J. Su *et al.*, "A Collision-Tolerant-Based Anti-Collision Algorithm for Large Scale RFID System," *IEEE Commun. Lett.*, vol. 21, iss. 7, pp. 1517–1520, Jul. 2017.
[11] C. Yang *et al.*, "An Effective 16-bit Random Number Aided Query Tree Algorithm for RFID Tag Anti-Collision," *IEEE Commun. Lett.*, vol. 15, iss. 5, pp. 539–541, May 2011.
[12] D. Zhang *et al.*, "A Novel Approach to Mapped Correlation of ID for RFID Anti-Collision," *IEEE Trans. Serv. Comp.*, vol. 7, iss. 4, pp. 741–748, Oct.–Dec. 2014.