International Journal of Control and Automation
Vol.7, No.10 (2014), pp.133-144
http://dx.doi.org/10.14257/ijca.2014.7.10.13

Modified M-ary QT Algorithm to Improve the Performance of the Anti-collision Protocol for RFID Tag Identification

Byun-Gon Kim[1] and Kwan-Woong Kim[2]

[1]*Department of Electronic Engineering, Kunsan National University, Kunsan, Korea*
[2]*Thunder Technology, Director of Digital Signal Processing Team, ChonJu, Korea*
bgkim@kunsan.ac.kr, kwkim@thunderspeaker.com

Abstract

The Radio Frequency Identification (RFID) system is a newemerging technology used to identify a tag attached to an object. To make RFID systems practical, the cost of RFIDtags is very important. Further, an interrogator should be able to identify thenumber of tags in the reading range in a short time. The most important technical issue in radio frequency identification (RFID) is to correctly and quickly recognize the number of tag IDs in the reader's communication range. Unlike the probabilistic RFID tag identification schemes, a query tree(QT)-based protocol guarantees the identification of all the tags, where the distribution of tag IDs is assumed to be uniform. In this paper, we proposed an extended M-ary QT protocol, which effectively reduces unnecessary query-response cycles by using a non-sequential m-bit recognition method. The simulation results show that the proposed scheme significantly outperforms M-ary QT schemes in terms of identification time, efficiency and communications overhead.

Keywords: *RFID, Manchester Coding, Collision Free, M-ary QT*

1. Introduction

The radio frequency identification (RFID) system consists ofradio tags, readers and the backend database system. Readerscommunicate with tags at a distance through wirelesstransmission. With the help of a backend system, a reader canidentify a radio tag. Many manufacturers look for the intendedapplications of RFID technology, and deploy RFID ininventory control, the distribution industry and supply chainmanagement[1].

An RFID system is the wireless non-contact use of radio-frequency electromagnetic fields to transfer data, for the purposes of automatically identifying and tracking tags attached to objects[2]. Several unsolved issues need to be addressed to use RFID technology in various industrial fields. One of the important issues is identifying multiple tags.To make RFID systems practical, the cost of an RFIDtag is very important. Additionally, an interrogator should identify thenumber of tags in the reading range in a short time. The RFIDtags can be classified as Passive, Semi-passive, and Active tags.

A passive tag is charged by the reader'selectromagnetic waves, instead of using an internal powersupply. Thus, the passive tag is the cheapest solution for RFIDsystems and we focus on passive tags in this paper. In thesection 3, we use a tag instead of a passive tag.The most important role of a reader is identifyingtags (or reading IDs of tags) in its reading range. Then,the readergives instruction to tags to carry out such as a reador write command. A tag is able to perform readingor writing when the reader sends the command. Sincethe tag has no carrier sense ability, tags' responses can becollided. Depending on the number of tags that respond to the reader, there are three cases of communication betweentags and the reader.

ISSN: 2005-4297 IJCA

Many tag identification mechanisms have been proposed. Tag identification mechanisms are classified into two categories, the probabilistic approach and deterministic approach. Protocols based on slotted ALOHA are popular in the probabilistic approach, whereas the tree-based approach is a popular deterministic approach[2-7].

ALOHA-based schemes canreduce the collision probability, but they have the tag starvationproblem whereby a particular tag may not be identified for a longtime[6]. Tree-based protocols do not cause the tag starvationproblem, but they have a relatively long identification delay.Therefore, we consider tree-based protocols and seek to reducethe identification delay. Collision resolution algorithms of tree-basedprotocols are very similar to the binary search algorithm.

The basic idea of the tree-based protocols [8-12] is to repeatedly split the group of tags encountering collisions into subgroups until there is only one tag in a subgroup to be identified without collisions.Typical tree-based algorithms are the query tree, collision tree, and M-ary query tree.Tag collision free algorithms based on a tree are similar to the binary search algorithm. In a tree-based collision free algorithm, each node that composes a tree needs a query-response process. Therefore, we can reduce the required query-response processing by half, if one level is reduced in the binary tree.

In the M-ary tree, it reduces the number of query-responses by 1/M compared to a binary tree[12]. The M-ary tree algorithm generates the next query message by analyzingthe response message from tags and the current query message. Also it can recognize sequential m-bits of tag ID using a mapping function. But it cannot generate a query message for non-sequential m-bit recognition. Therefore, the M-ary tree algorithm requires an additional query-response cycle to identify the non-sequential m-bit of tag IDs.

In this study, we proposed an enhanced tag identification algorithm based on aM-ary query tree. The proposed algorithm can reduce tag identification time by using collision bit location information, which is based on Manchester coding.The proposed tag identification algorithm can generate a query message that recognizes non-sequential m-bits of ID. Tags generate a response message if the received query from the reader is identical to its ID, except the collision bits.Therefore, the proposed algorithm can reduce tag identification cycles compared to M-ary tree-based algorithms.

2. Related Work

As mentioned in the introduction, the RFID systemfor solvingmulti-tag collision problemsis currentlydivided into ALOHA-based anti-collision algorithmsand tree-based anti-collision algorithms [1-2]. ALOHA-based identification algorithms such as Slotted ALOHA, framed slottedALOHA, dynamic framed slotted ALOHA, and enhanceddynamic framed slotted ALOHA are based on the probabilisticmethod, inwhich each tag selects a time slot in a random manner to respond to the query of the reader in the event of a collision.

If there are a large number of tags, the collision will happenwhen several tags repeatedly back off after the collision, so therewill be a long time for the tag to be identified. Tree-based identification algorithms, such as binary search algorithms, query tree algorithms, and tree-splitting algorithmsare based on deterministic methods. In the tree-based identification algorithm, the reader iteratively queries a subsetof tags that match a given property until all tags areidentified.

Before presenting the proposed tag identification scheme, this section introduces related previous work: Manchester coding, query tree protocol, and M-aryquery protocols, because they are the bases of the proposed scheme.

2.1. Query Tree Protocol

The query tree protocol (QT) uses tag IDs to splittags into two groups. The reader transmits a query includinga bit string called a prefix. The tag in the range of the readercompares the prefix with its ID and transmits its ID to thereader in the case the result compared is true. If there is acollision, the reader queries for a 1-bit longer prefix, and thegroup is split into two groups that march the new prefixesrespectively. Tags in one group transmit their IDs to the reader.

Tags in the other group wait for the next query of the reader. Thereader repeats the division of tags into two groups until the numberof tags in a group is one. When there is only one tag in agroup, the reader identifies the tag. Once a tag is identified, thereader starts a new round of queries with another prefix,untilall tags are identified. QT is also called a memory-less protocol,because the current response of each tag only depends on thecurrent query of the reader but not on the past history of thereader's queries [8].

Since QT and other tree-based protocols [8-10]cannot avoid idle cycles, in which there is no tag answeringthe query sent by the reader, the identificationperformance and efficiency of them are limited.

2.2. Basics of Manchester Coding

In QT, it could not get any information from collisions when there were more than two tag responses to the reader simultaneously. Manchester coding which has been used in binary search protocols.In Manchester coding, we can get the location of collision bits. Each bit represents a transition of a logic value. Bit '0' is a positive transition and bit '1' is a negative transition. If more than two tags respond at the same time, both positive transitions and negative transitions induce cancelation, thus 'no transition' will take place, which is not allowed in Manchester coding. Therefore, 'no transition' of a specific bit position indicates a collision. By using this phenomenon, we know the bit positions of no collision as well as bit positions of collision.

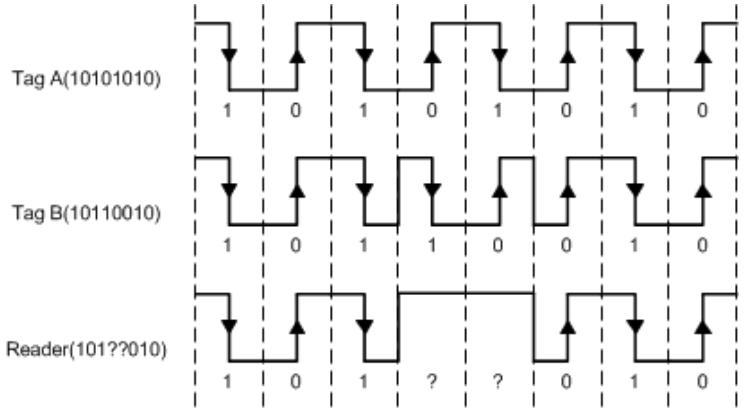

Figure 1. Collision Example of Manchester Coding

2.3. Collision Tree Protocol

The collision tree protocol (CT)[11] is based on the QT and memory-less protocols. The main characteristicof CT is that both generating prefixes and splitting tags areaccording to the first collided bit. For query $q_1q_2 \ldots q_k$ andresponse $p_1p_2 \ldots p_{c-1}p_c$, where p_i, $q_i \in \{0, 1\}$, and p_c is thefirst collided bit, the reader uses "$q_1q_2 \ldots q_kp_1p_2 \ldots p_{c-1}0$" and"$q_1q_2 \ldots q_kp_1p_2 \ldots p_{c-1}1$" as new prefixes. The group of tagsthat match "$q_1q_2 \ldots q_k$" is split into two groups: the tags inone of them match "$q_1q_2 \ldots q_kp_1p_2 \ldots p_{c-1}0$", and the tags inthe other match "$q_1q_2 \ldots$

$q_k p_1 p_2 \ldots p_{c-1}1$". Meanwhile,in CT, the tags only transmit their IDs, except the part that isthe same as the received prefix. Figure 2 shows the flow diagramof CT, in which the stack is used as a prefix pool to store theprefixes for the next query.

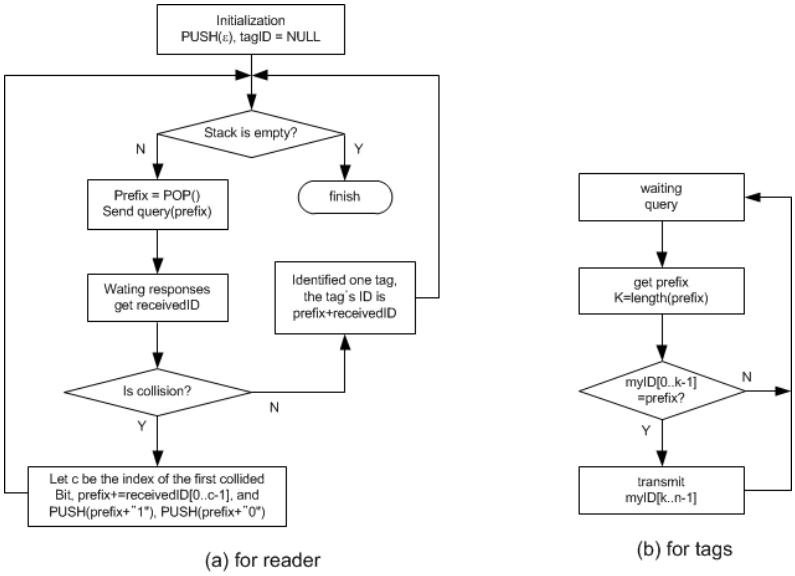

Figure 2. Flow Chart of CT: (a) Reader, (b) Tags

A collision tree is a finite set of nodes that is either emptyor consists of a root r and two disjoint collision trees calledthe left and right sub-trees. The roots of the sub-trees are calledthe left and right children of r. The sub-tree is still a collisiontree. In tags identification, if there is no response (i.e., no tagresponds to the query), the collision tree is defined to be NULL;if there is no collision (i.e., only one tag responds to the query),the collision tree has only one node; if there is a collision(i.e., at least two tags respond to the query), the collision treehas three nodes.

In CT, a tag set is split into two subsets only in a collisioncycle, and the splitting is continued until there is only one tagin a set. Thus, there are at least two tags in an internal nodeand exactly only one tag in a leaf node. On the other hand,tag identification is coincident with a tree search, which startsat the root of the tree for finding nodes of readable cycles.Thus, there is no idle node in a collision tree and no idlecycle in the tags identification using CT.

2.4. M-aryQuery Tree

M-aryQT(Query Tree) is a method that identifiesthe m-bit of a collision by using a mapping function while QT provides only one bit identification.In MQT, m-bits arbitration is made at a timewhere m is $log_2 M$. This is because MQT is not based on abinary tree and Manchester code makes it possible to tracea collision to an individual bit.

The MQT protocol [12] tries to split the group of colliding tags into M (M>2)sub-groups at a time, instead of 2 in the QT protocol, by appending ⌈⌉ bits to the request string S. In the M-Tree protocol, the number of iterations with tag collisions decreases with M and the number of iterations with no tag response increases with M. Achieving a compromise between the two numbers is thus important for the MQTprotocol. As shown in the paper [12], the identification performance of the MQT protocol is optimal when M = 3.

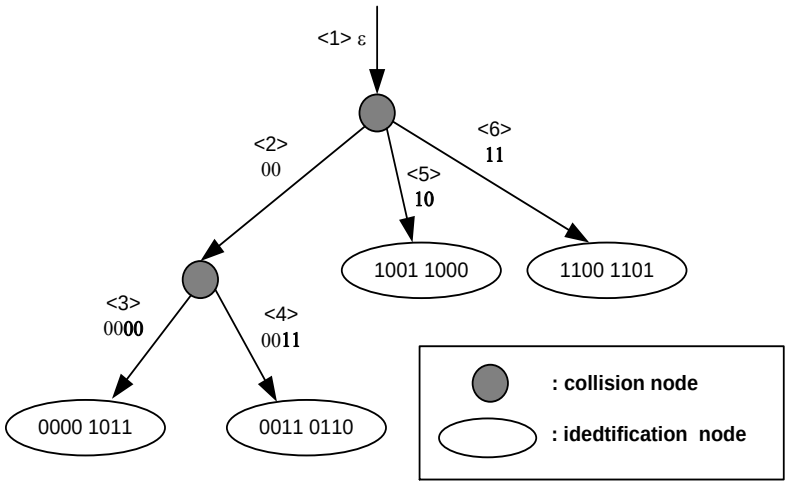

Figure 3. Tag Identification Process of MQT Protocols

Each tag converted the received m-bits string to an M(2^m)-bit string using a mapping function. For example, 2-bits data '00', '01', '10', '11' are converted into '0001', '0010', '0100', '1000' respectively.The identification process can be represented byan MQT query tree, as shown in Fig. 3.

The reader de-queuesthe prefix, which is formed of bits of string, from the query queue and broadcasts the prefix to tags. Each tag comparesits own ID with the prefix, if it is matched, and then sends a response message to the reader.

In the response message, significant m-bits of ID exclude prefix are converted into M-bits string using a mapping function and the remaining bits are attached in the tail of the message.The reader also converts M-bits into m-bit and extracts a part of the ID information using the same mapping table. As a result, the reader updates the new prefix and sends it to tags. The same process is repeated until all tags are uniquely identified.

3. Proposed Collision Resolution Method

3.1. The Proposed Modified M-ary Query Tree

In this section, we introduce our enhanced tag identification protocol based on M-ary QT using Manchester coding. The proposed tag identification algorithm providesa faster ID search method, which is modified M-ary QT.The proposed algorithm can identifya unique RFID by using the collision bit position and non-collision bit position information. The proposed tag identification algorithm can generate a query message, which recognizes non-sequential m-bits of ID using Manchester coding. Therefore the proposed algorithm gives better performance than a traditional M-ary tree protocol.

Figure 4 shows how the proposed algorithm generates a prefix to accelerate the search process. Where '*' is 'no transition'.

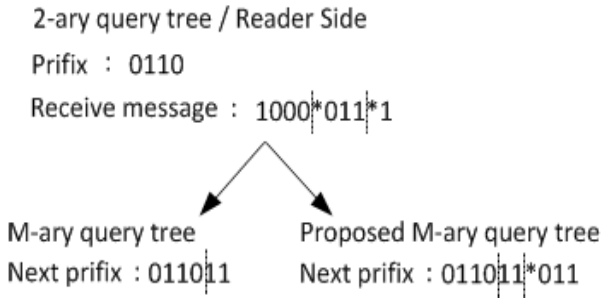

Figure 4. Examples of the Next Prefix Generation of the M-ary Query Tree and Proposed Algorithm

In the 2-ary query tree, if the received response message is '1000*011*1 of prefix '0110', the next prefix will be '011011' ('1000' converted into '11' by the mapping table).But we know that several RFID nodes have ID '011011*011*1'.Therefore, the proposed algorithm generates '011011*011' as the next prefix.

When tags receive the prefix that contains '*' (no-transition), they generate the response message using bit positions of '*'s. For example, when tags receive '011011*011' as shown in Figure 1, the tags generate an extended 2-ary code for the 7th bit '*' from the 11th bit of own ID. The tag of ID '01101100111' generates extended code '0010' from '01', which underlined bits of ID. The final response message is '00101' which rest bits are added tail of '0010'. In the case of '01101110111', the response code will be '10001' as same manner. In this way, the reader repeats the same procedures until all tags are identified.

The tag identification procedure of the proposed algorithm is described as follows.

<Reader>

1) The reader sends out a query q. Initially, it starts with an empty string ε.

2) Two possible cases can arise based on the mapping part from the tag's response. The mapping part is a string of 2^m bits from the MSB of the entire tags' response message.

- If collision bits exist in the mapping part and there is no '*' in the entire tag's response string, the number of '*' tags is identified.

- Otherwise: the reader adds'*'s to the previous query to make the number '*' m in the next query. Let the number of '∗'s t in the mapping part, then $r_1, \ldots, r_t$ are the strings that can be inversely mapped from the mapping table. Query $qr_1, \ldots, qr_t$are prepared in the next cycle.

- Add the bit stream MSB to m-th'*' in the remaining part of the response to the tail of each query.

- Remove sequential '*' from the tail of a new query, and then push new query to the query stack.

3) Repeat steps 1)–2) until all tags are identified.

<Tag>

1) When the tag receives a query from the reader, itconverts '*'s of query to don't care bit and the query string matches the prefix of the tag ID, then it respondsto the restof the tag ID, except the prefix part.

At this time, the tags generate a bit stream, whichis mapped into M(=2^m) bits by the mapping table using the most significant bits of the remaining part and the bits of do not care.

2) The tags generate a response message by combining the remaining part of the ID to a bit stream and sending it to the reader.

The example of the tag identification procedure is illustrated in Figure 5 and Table 1, and the identification process can be represented by a query tree, as shown in Figure 5.

In the initial step of Table 1, all tags send a response, and the reader receives "*00**10***". The reader generates two queries: 00*10, 11*10, and two bits of MSB are the inverted value of the mapping part of response. "*10" is from the remaining part of the response.

In <2> of Table1,the reader pop query "00*10" from the query stack, and sends it to the tags. Five tags (0001 0000, 0001 0001, 0001 0011, 00010110, 00010111) having the prefix of 00,respond with 000100 and 000101, 000111, 001010, 001011, respectively, whichproduce 00****. The reader generates two queries000100, 000101 with response 00**** and the previous query.

In the next step, tags(00010000, 00010001, 00010011) respond with 0001, 0010, 0100, which produce 0*** for query 000100. The reader can identify 3 tags 00010000, 00010001, 00010010 from 0*** (underlined two numbers is decoded value by mapping function). The rest of the identification procedure is the same as the previous process.

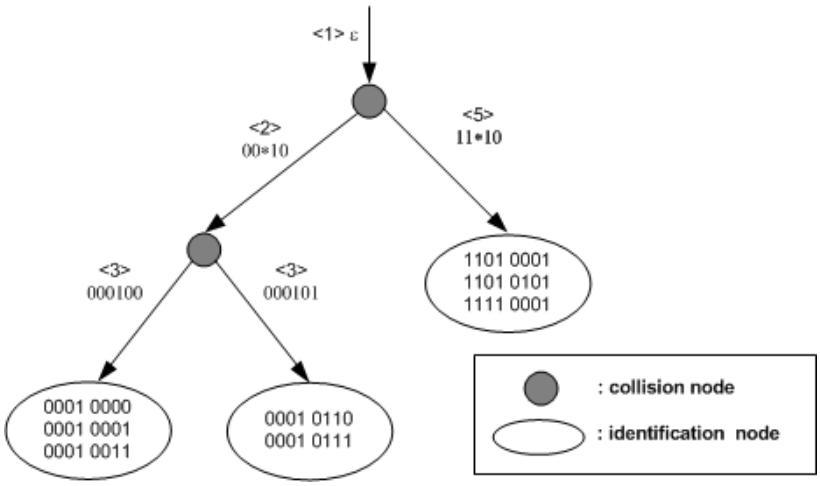

Figure 5. Query Tree of Proposed Algorithm for Table 1

Table 1. Example of Tag Identification Procedure (m = 2)

Cycle	query	Response		query stack	identification
		mapping part	remaining part		
<1>	ε	*00*	*10***	00*10 11*10	
<2>	00*10	00**	**	000100 000101 11*10	

<3>	000100	*0**		000101 11*10	0001 0000 0001 0001 0001 0011
<4>	000101	**00		11*10	0001 0110 0001 0111
<5>	11*10	0***	01		1101 0001 1101 0101 1111 0001

Table 2 Mapping Table for 2bits (m=2)

2 bits	$4(2^m)$ bits
00	0001
01	0010
10	0100
11	1000

4. Experimental Performance Evaluations

In this paper, we compare the performance of M-ary QT and the proposed scheme using computer simulation. In the simulation, the length of the RFID tag is 64bits. The number of tags varies from 30 to960 and the RFID of the tags are generated in uniform distribution. The value of 'M' is 4.

To evaluate the efficiency of algorithms, three metrics are considered as follows.

1) Number of query-response cycles& identification efficiency

We measure the required cycle to identify all tags. Fast identification is one of the most significant factors in tag identification protocols.

The identification efficiency is the number of tags/number of responses

2) Communication overhead

Communication overhead: This metric is the transmittedbits for one tag identification. Note that this influencesthe amount of power consumption and it must be lowdue to the lack of the power source of tags.

Figure 6 shows the total number of query and response cycles in 4-ary QT and the proposed scheme. The proposed scheme needs 1322 query-response cycles while that of 4-ary QT is2282 when the number of tags is960. Thus, the proposed scheme reduces tag identification time more than 40% over 4-ary QT.

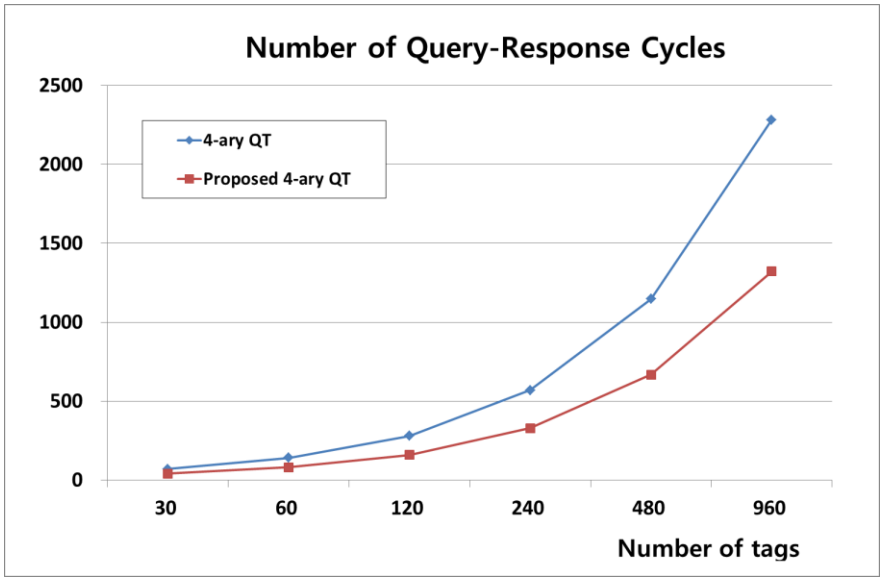

Figure 6. Total Query-Response Cycles of 4-ary QT and the Proposed Algorithm

Figure 7 shows the identification efficiency of algorithms. 4-ary QT requires about 0.75 cycleswhile the proposed algorithm requires .43. It means that 0.43 query-response cycles are required to complete one tag identification.

Figure 8 shows the required communication overhead of 4-ary QT and the proposed scheme. The communication overhead is the total bits of query and response over the number of tags.

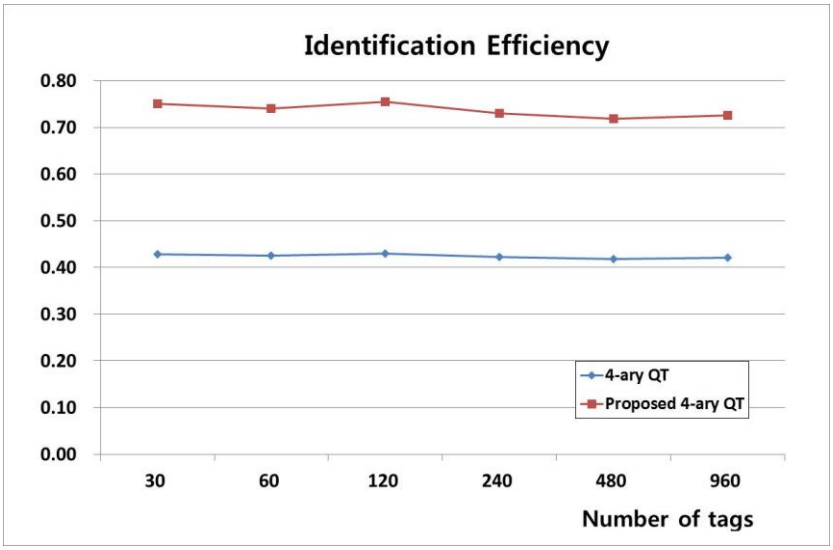

Figure 7. Identification Efficiency of 4-Ary Qt and the Proposed Algorithm

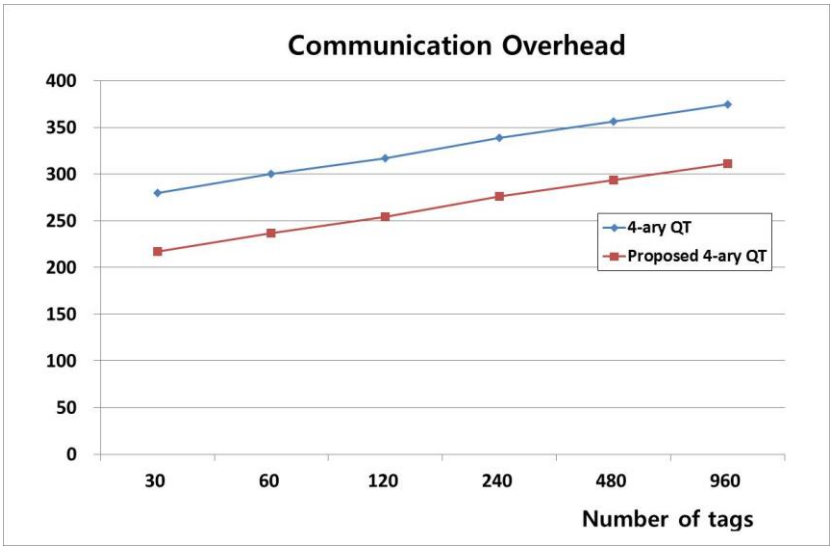

Figure 8. Communication Overhead of 4-ary QT and the Proposed Algorithm

In general, the experimental results show that the proposed scheme cansignificantly reduce the required cycles and transmitted bitsfor tag identification, and thus the identification speed can beimproved remarkably compared to M-ary QT, while the communication overhead is reduced.

5. Conclusion

Radio-frequency identification (RFID) is the wireless non-contact use of radio-frequency electromagnetic fields to transfer data, for the purposes of automatically identifying and tracking tags attached to objects. Several unsolved issues need to be addressed to use RFID technology in various industrial fields. One of the important issues is identifying multiple tags. Many tag identification mechanisms have been proposed.The M-ary QT scheme is the most popular deterministic scheme based on Manchester coding.

In this paper, we proposed an RFID identification scheme based on m-bits recognition and non-sequential prefix generation that efficiently reduces unnecessary query-response cycles.

From the simulation results, the proposed scheme outperforms the original M-ary QT scheme in terms of identification time, efficiency and communication overhead.

6. References

[1] K. Finkenzeller, “RFID Handbook:Fundamentals and Applications in Contactless Smart Cards and Identification”, John Wiley & Sons, **(2003).**

[2] H. Vogt, “Efficient Object Identification with Passive RFID Tags”, Proceedings of International Conference of Pervasive Computing, Zurich, Switzerland, **(2003)** August.

[3] J. Myung, W. Lee, J. Srivastava and T. K. Shih, “Tag-splitting: Adaptive collision arbitration protocols for RFID tag identification,” IEEE Trans. Parallel Distributed System, vol. 18, no. 6, **(2007)**, pp. 763-775.

[4] C. H. Oh, “Location Estimation Enhancement Using Space-time Signal Processing in Wireless Sensor Networks: Non-coherent Detection”, lnformation Communal Converge, England, vol. 10, no. 3, **(2012)**, pp. 269-275.

[5] X. Jia, Q. Feng and C. Ma, “An efficient anti-collision protocol for RFID tag identification,” IEEE Commun. Lett., vol. 14, no. 11, **(2010)**, pp. 1014-1016.

[6] S. Lee, S. D. Joo and C. W. Lee, “An enhanced dynamic framed slottedaloha algorithm for RFID tag identification,” in ACM Mobiquitos, **(2005)** July.

[7] Y. J. Park and Y. B. Kim, "Accelerating RFID Tag Identification Processes with Frame Size Constraint Relaxation," J. lnf. Commun. Converg. Eng., vol. 10, no. 3, **(2012)**, pp. 242-247.
[8] C. Law, K. Lee and K. Y. Siu, "Efficient Memoryless Protocol for Tag Identification", Proceedings of Fourth International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM'00), **(2000)**, New York, NY, USA.
[9] J. Myung, W. Lee, J. Srivastava and T. K. Shih, "Tag-splitting: adaptivecollision arbitration protocols for RFID tag identification," IEEE Trans. Parallel Distrib. Syst., vol. 18, no. 6, **(2007)**, pp. 763-775.
[10] G. Bagnato, G. Maselli, C. Etrioli and C. Vicari, "Performance analysisof anti-collision protocols for RFID systems", Proceedings of VTC Spring 2009-IEEE 69th Vehicular Technology Conference, Barcelona, Spain, **(2009)** April.
[11] X. Jia and Q. Feng, "An Efficient Anti-Collision Protocol for RFID Tag Identification", IEEE Communications Letters, vol. 14, no. 11, **(2010)**.
[12] P. Mathys and P. Flajolet, "Q-ary collision resolution algorithms in random-access systems with free or blocked channel access", IEEE Trans. Inform. Theory, vol. 31, no. 4, **(1985)**, pp. 217-243.

Authors

Byun-Gon Kim, he received his B.S. from Korea Aerospace Univ. in 1990, M.S. and Ph. Ddegrees in Electronic Engineering fromChonbuk National University in 1997 and 2001, respectively. Currently, he isa senior researcher in thunder technology. Hisinterests include traffic management in telecommunicationnetworks and Quality of Service in wirelessnetworks& RFID systems.

Kwan-Woong Kim, he received his B.S., M.S. and Ph. Ddegrees in Electronic Engineering fromChonbuk National University in 1996,1998 and 2002, respectively. Currently, he isa senior researcher in thunder technology. His interests include traffic management and routing protocols in telecommunicationnetworks and Quality of Service in wirelessnetworks.