We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

An Efficient Cut-through Mechanism for Tree-based RFID Tag Identification Schemes

Nai-Wei Lo and Kuo-Hui Yeh
Dept. of Information Management
National Taiwan Univ. of Science and Technology, Taipei,
Taiwan

1. Introduction

Radio Frequency IDentification (RFID) technology successfully integrates radio transmitter and receiver, small size memory space and control circuitry to remotely store and retrieve data via wireless RF transceiver. As low-cost RFID tag can be massively manufactured by semiconductor industry, new applications associated with RFID technology have emerged rapidly such as inventory tracking, book storage and management, and airport baggage handling. Generally an RFID system is composed of many RFID tags (or tagged objects), at least one RFID reader, and one backend application system with database. An RFID tag gains its operation energy from query signals that are radio waves transmitted by RFID reader. Once a tag responds the query with its stored information to RFID reader, usually a unique identity number or string, the RFID system can recognize the tag carrier and perform required business logic operation.

As RFID technique is widely used in supply chain management, the corresponding RFID application systems are expected to process and manage large amount of tagged objects at a predefined period of time regularly. Therefore, for large-scaled RFID applications, tag reading throughput is very critical since it will affect the total data processing time. To measure tag reading throughput of a RFID system, two performance criteria are usually adopted: tag reading delay and reader energy consumption [5]. As RFID reader and tag always communicate over a shared wireless channel, it is very easy to have signal collisions during a normal tag identification process when multiple tags exist in reader's interrogation area. Signal collision results in query or data retransmission and eventually increases communication overhead and time delay of tag identification. Hence, an efficient tag collision arbitration mechanism is very important and critical for RFID systems to achieve effective performance.

Existing RFID tag identification schemes can be classified into aloha-based solutions [3, 10, 12-13, 16, 21, 23-24] and tree-based solutions [1, 2, 4, 6-9, 14-15, 17-20, 22]. In aloha-based schemes, a reader estimates the number of tags in its interrogation area and broadcasts the total number of timeslots to each tag. Then each RFID tag randomly selects its own timeslot to transmit its ID without knowing which timeslots have been selected by other tags. By distributing tag responses to distinct timeslots instead of getting all tag responses at the same time, aloha-based schemes effectively reduce the probability of signal collision. However, aloha-based schemes have tag starvation problem in which certain tags may not

be identified for a long period of time and consequently the predefined processing time for a RFID application system to recognize all existing tags may not be achieved. Tree-based schemes are based on tag-set splitting mechanism. When querying RFID tags, the RFID reader continuously splits signal-collided tags into two subsets until each tag set contains only one tag. This technique guarantees that every tag can be identified in a certain period of query time. As tree-based schemes are based on current tag responses to determine how tag subset should be formed and queried, higher energy consumption and increase of query time due to vast splitting and querying operations are emerged.

In this study, we introduce a cut-through mechanism (*CM*) to avoid serious signal collision cases in advance and improve tag identification efficiency for RFID system in consequence. From the viewpoint of tree structure, all tag IDs can be represented as leaf nodes of the abstract tree. When the number of tags to be queried is large, intermediate branch nodes of the abstract tree, that represent partial tag IDs sent in tag responses and locate only several levels above the leaf level, will have a significant probability to be visited during a tag identification process. By adopting *CM*, a tag identification scheme can bypass these colliding intermediate nodes during a tag recognition process. In addition, *CM* is a generic mechanism and it can be embedded with any given tree-based RFID tag identification protocol.

2. Cut-through Mechanism (*CM*)

Fig. 1 shows the scenario in a traditional binary-tree based tag identification protocol, once all current tags are spread in the lowest layer, i.e. leaf node, the upper several layers of the abstract tree structure will be full of collided nodes. The higher identification efficiency can be expected if we can ignore the visiting on these collided intermediate nodes. In other words, if we start the binary-tree based tag identification protocol from each node at the 4th layer instead of the root node, we will reduce all collided nodes (i.e. collided cycles) located at the first 3 layers. This leads to another interesting question: which is the best layer to start performing the tag identification procedure? For our analysis and simulation results, the solution depends on the number of tags, i.e. q, and the position of these tags, i.e. which leaf nodes they are located. Moreover, the impact weight of the number of tags is more important than the other one. Hence, the number of tags will be the main factor while considering how much layers (or collided intermediate nodes) we should cut off to pursue better system efficiency in a tag identification session.

The goal of *CM* is to avoid more collided nodes in the abstract tree structure before performing a tree-based tag identification protocol. This will make the whole tag identification process smoother and accordingly complete the tag recognition in a shorter time. More specifically, if we can recognize more readable nodes without visiting their collided ancestor nodes, the system throughput during the tag identification procedure will be better. This indicates that we can save all collision resolution operations which are supposed to perform on each of such collided ancestor nodes. On the other hand, in a tree based tag recognition protocol, the identification efficiency will be decreased if the reader has to visit some idle nodes without any useful feedback. As a result, it is critical to find the target layer h which possesses the best trade-off between the number of collided nodes and the number of idle nodes in such layer. If we cut off more collided (and useless) nodes, the better identification efficiency is obtained. However, the idle nodes should be carefully handled as the elimination of them may result in poor performance. This is because each idle node is produced by at least one descendant idle node in the abstract tree structure

corresponding to the target tag identification scheme. In the following, we present our simulation results to show how many layers should be eliminated before invoking a tree based tag identification protocol to pursue better efficiency.
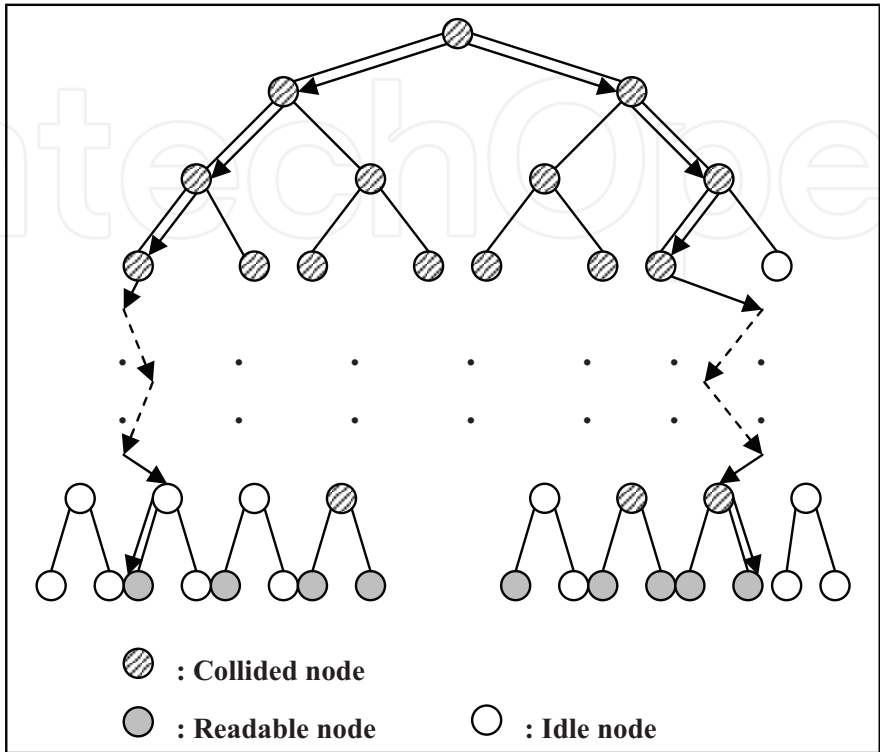

Fig. 1. A scenario in traditional binary-tree based tag identification protocol.
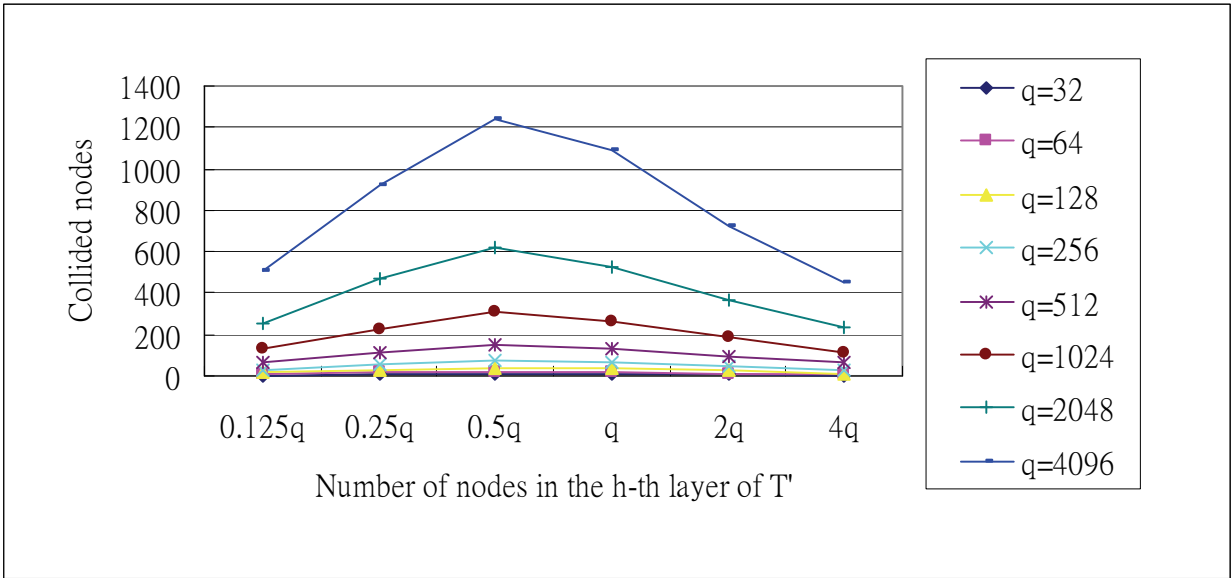

Fig. 2. Collided nodes in the hth layer of T' with different number of tags. Note that the length of ID is 96 bits.

From Fig. 2 and 3, we find that the hth layer, where the number of nodes (2^h) is closest to $0.5q$, is the best candidate layer to start the target tag identification scheme. In such layer, the

ratio of the number of collided nodes to the number of total nodes, i.e. 2^h, is the highest, and at the same time the number of idle nodes is not so large (within an acceptable level). If we invoke the target tree based tag identification protocol on each node of the hth layer instead of root node, all collided intermediate nodes in the ancestry layers of the hth layer can be ignored to visit. More precisely, performance improvement can be gained by ignoring the visiting on the collided nodes located from the 1st to the $(h-1)$th layers. Meanwhile, the reader only wastes a little time to visit few idle nodes at the target layer h. This interesting finding is utilized to construct the detailed procedure of *CM* in the next subsection.
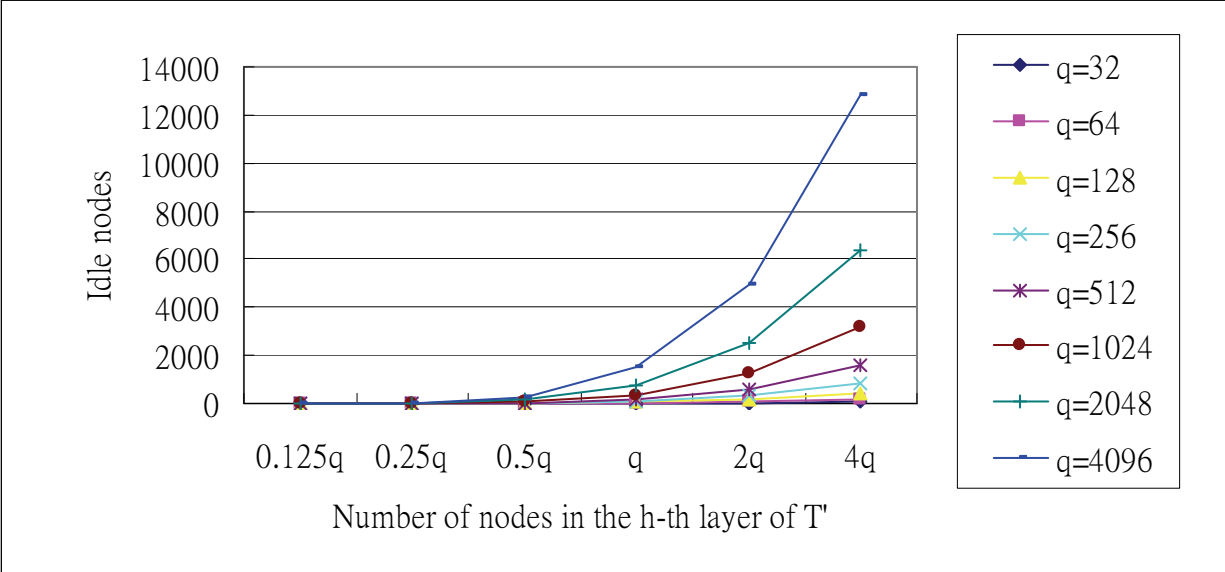

Fig. 3. Idle nodes in the hth layer of T' with different number of tags. Note that the length of ID is 96 bits.

Cut-Through Mechanism (the operation at reader side)
/* Cut off useless collided nodes of target tree T*/

1   Input the number of current tags q
2   Search the left-most readable node *LN* and the right-most readable node *RN* of *T*
3   Construct a new tree *T'* with the trajectory of finding *LN* and *RN*
4   Search the hth layer of *T'* in which the value of 2^h is the closest one compared to 0.5q
5   For each node in the hth layer of *T'*
        Invoke "targeted tree-based tag identification protocol such as QT, BS and *k*-TAS"

Fig. 4. *CM* algorithm at RFID reader side.

As mentioned before, this study focuses on solving the tag signal collisions in the circumstances of the RFID applications in section II. In such applications, it is reasonable to assume that the number of tags is known. Under this assumption, we present the detail of *CM* in Fig. 4. The reader first probes the left-most readable node and right-most readable node in sequence. With the retrieval trajectory of these two nodes, a new tree structure T', which is contained in the target tree T, can be constructed. An example of T' is referred to the partial tree structure surrounded by the black arrow lines (and black dotted arrow lines) in Fig. 1. Next, as the number of current tags, i.e. q, is known in *CM*, the reader searches the hth layer of T' in which $0.5q$ is closest to 2^h. Finally, for each nodes on the hth layer of T', the reader recursively invoke the target tag identification schemes such as Query Tree (QT) [1, 4, 8, 15, 22], Binary Search (BS) [7, 11], ETIP [25] and k-TAS [26].
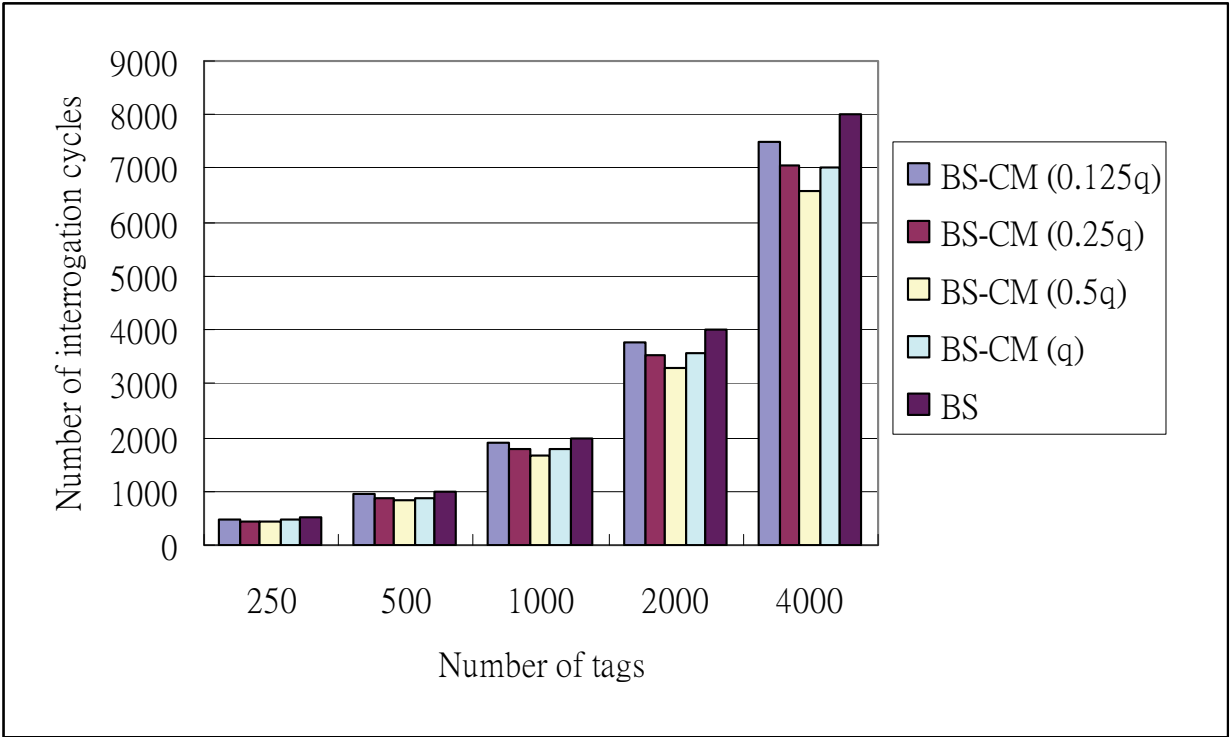

Fig. 5. The communication overhead of BS protocol embedded with *CM*. Note that the length of ID is 96 bits.

Fig 5 and 6 present the identification delay and communication overhead of BS embedded with *CM* in terms of the total interrogation cycles and the amount of transmitted bits. Note that BS-CM (0.25q) denotes the BS protocol is invoked at each node in the hth layer of target tag identification tree in which 2^h is closest to 0.25q. For the aspect of the identification delay (Fig. 5), BS-CM (0.5q) shows the best performance as the total interrogation cycles for identifying all tags is reduced between 15.5% (q=250) and 17.7% (q=4000). As the number of tags rises, the improvement is more significant. This results from that the rise of the number of tags will increase the number of collided cycles, and accordingly complicate the tag identification procedure itself. Since the focus of *CM* is to cut off unnecessary collided nodes, the reader embedded with *CM* can obtain better performance by avoiding more collided cycles (i.e. collided intermediate nodes in the abstract identification tree structure). In Fig. 6, BS-CM (0.5q) and BS-CM (q) both show performance improvement by eliminating

4.8%-6.2% and 6%-10% of total transmitted bits, respectively. Similarly, *CM* is more efficient on the communication overhead when the number of tags becomes larger (i.e. the number of collided nodes rises). Based on these two results, we can conclude that *CM* is significantly effective in reducing the collided cycles during a tag identification process. As *CM* is simple and independent of existing protocol properties, any tree-based arbitration protocol can easily embed *CM* for better performance. This embeddedness characteristic of *CM* is valuable.

Fig. 6. The identification delay of BS protocol embedded with *CM*. Note that the length of ID is 96 bits.

In the following, we evaluate the performance of *k*-TAS scheme embedded with *CM* ($0.5q$) in terms of identification delay and communication overhead, where q is the number of current tags. From the simulation results in Fig. 7 and 8, we summarize four interesting findings. First of all, the efficiency improvement of embedding *CM* into a tree-based tag identification protocols such as *k*-TAS and BS is significant. In Fig.7, *CM* can improve system performance for *k*-TAS (*i*=2) scheme between 8% and 9.9% and for BS scheme between 15.3% and 18.3% in terms of reduction on total interrogation cycles. Secondly, we discover that the improvement ratio of *CM* depends on the structure of target tag identification tree, i.e. how many degrees each node has. With the increase of node degree, the improvement ratio is decreased. That is, *CM* can enhance more efficiency when the node degree in the target identification tree is smaller. From Fig.7, *CM* reduces identification delay for *k*-TAS (*i*=2) protocol between 8% and 9.9% and for *k*-TAS (*i*=3) between 2% and 4%, respectively. Since in BS each node possesses only two degrees, efficiency improvement is the most significant, i.e. between 15.3% and 18.3%. This phenomenon is because that *CM* always cuts a fixed number, i.e. $0.25q$, of collided intermediate nodes (or cycles) in each identification session. Since these $0.25q$ collided nodes will actually appear in the abstract tree structure corresponded to BS, the performance improvement of BS-CM is better. As the nature of *k*-TAS is to efficiently reduce many collided nodes by utilizing a synchronized data sequence,

only some parts of these 0.25*q* collided nodes emerge in *k*-TAS. Therefore, efficiency improvement is comparatively smaller. In brief, if more collided nodes are contained in the several top layers of abstract tree structure when performing a tree based tag identification scheme, adopting *CM* into such tree-based tag identification protocol can still gain some extra performance.
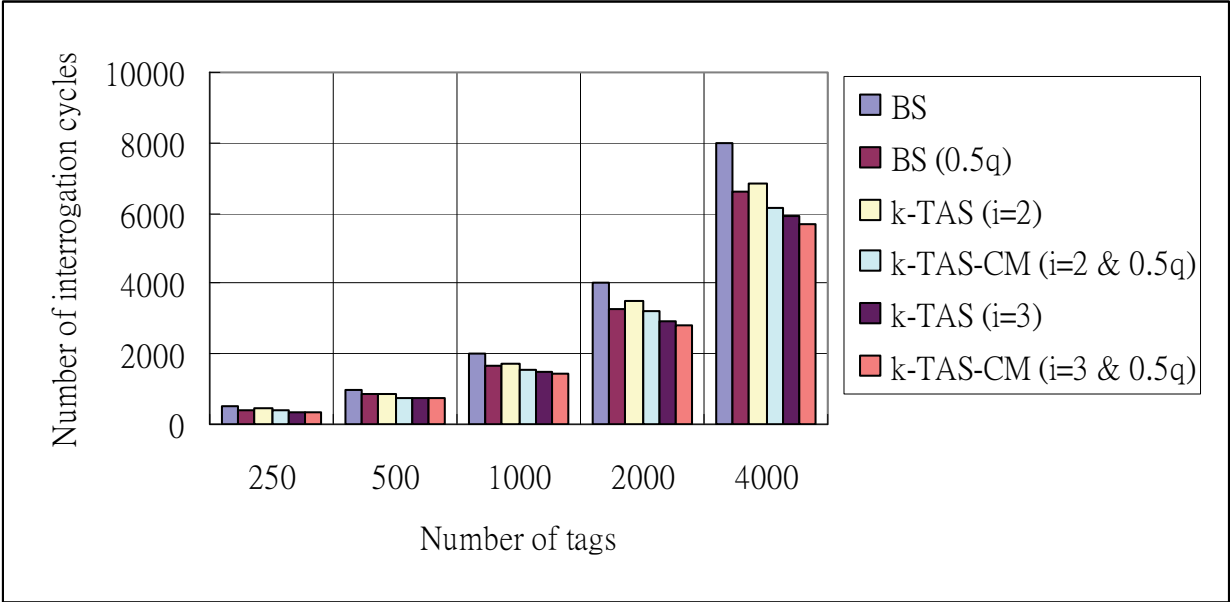

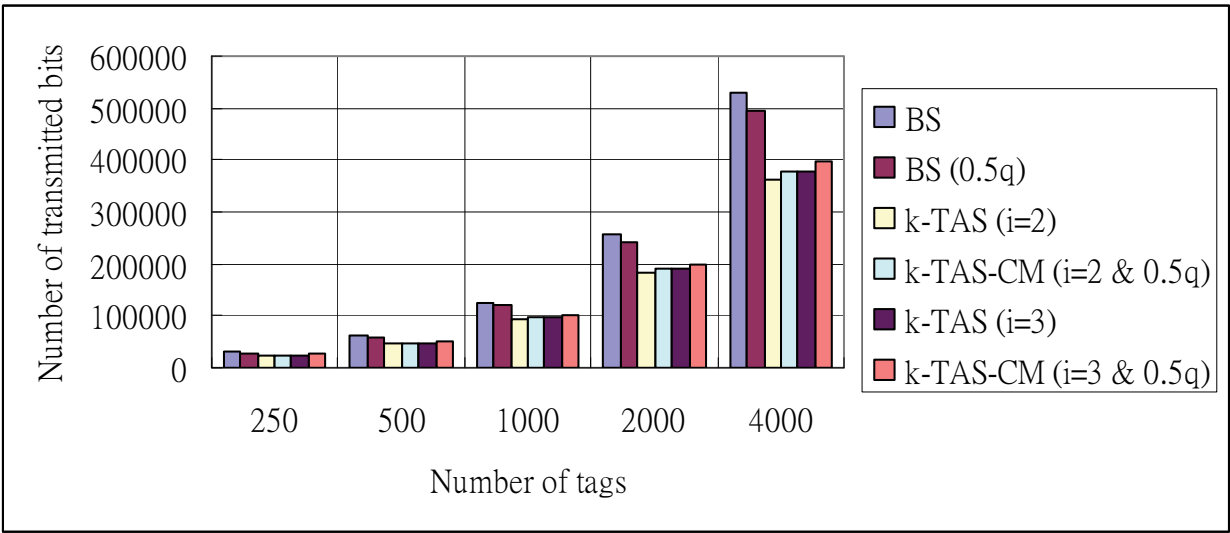Fig. 7. The identification delay of *k*-TAS embedded with *CM* in which the length of ID is 96 bits.

Fig. 8. The communication overhead of *k*-TAS embedded with *CM* in which the length of ID is 96 bits.

Thirdly, Fig. 7 shows that the performance of BS (0.5*q*) is similar to the results in *k*-TAS (*i*=2), *k*-TAS-CM (*i*=2 & 0.5*q*) and *k*-TAS (*i*=3). These results show performance obstacle in *k*-TAS scheme when many collided nodes appear at the top several layers of the corresponding abstract tree structure. If we intend to improve the performance of *k*-TAS, we should modify

the tag identification procedure corresponding to the top several layers instead of the middle layers or the bottom layers. We mark this open question as our future research direction. Fourthly, for the aspect of communication overhead, Fig. 8 demonstrates that overhead reduction is not significant when embedding *CM* into *k*-TAS. Protocol performance between pure *k*-TAS and *k*-TAS with *CM* is almost the same. This is because the reader in *k*-TAS with *CM* has to visit some idle nodes on the hth layer, where 2^h is the closest value to $0.5q$ and q is the number of tags. In such idle nodes, some basic inquiry commands (bit strings) are required to be broadcast to all tags. This will increase the amount of transmitted bits. We infer that the total transmitted bits collected from all idle nodes at the hth layer is almost equal to bits collected from all collided nodes from the 0th layer to the $(h-1)^{th}$ layer. However, in general performance improvement is significant in terms of the transmitted bits for existing binary tree-based tag identification protocols (such as BS and QT) to embed with *CM*.
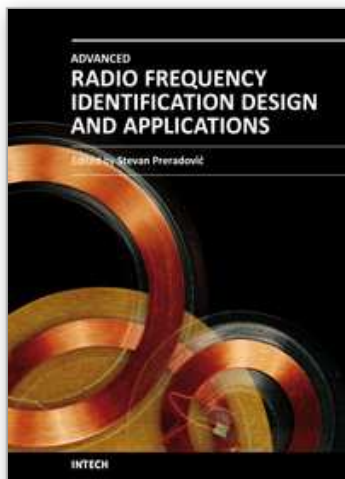
3. Conclusion

Efficient collision resolution is critical for an RFID tag identification protocol. In this study, we present a cut-through mechanism to enhance the performance of current existing anti-collision protocols in which a significant portion of signal-collided query cycles can be removed from tag identification protocols such as QT, BS and *k*-TAS. In the future, we would like to investigate how to effectively remove signal-collided nodes in the top node levels of the abstract tree structure to pursue better protocol efficiency; especially in the category of *k*-ary tree based anti-collision schemes.

4. Reference

[1] 860 MHz – 930 MHz Class 1 Radio Frequency Identification Tag Radio Frequency and Logical Communication Interface Specification Candidate Recommdation Version 1.0.1, Auto-ID Center, 2002.

[2] J. I. Capettanakis, "Tree Algorithms for Packet Broadcast Channels," IEEE Trans. on Information Theory, vol.25, 1979, pp.505-515.

[3] T. Cheng and L. Jin, "Analysis and Simulation of RFID Anti-Collision Algorithms," in proc. of 9th International Conference on Advanced Communication Technology, 2007, pp.697-701.

[4] K. W. Chiang, C. Hua and T. S. Peter Yum, "Prefix-Randomized Query-Tree Protocol for RFID Systems," in proc. of IEEE International Conference on Communications, 2006, pp.1653-1657.

[5] Chamtac, C. Petrioli and J. Redi, "Energy-Conserving Access Protocols for Identification Networks," IEEE/ACM Trans. on Networking, vol.7, no.1, 1999, pp.51-59.

[6] J. S. Cho, J. D. Shin and S. K. Kim, "RFID Tag Anti-Collision Protocol: Query Tree with Reversed IDs," in proc. of 10th International Conference on Advanced Communication Technology, 2008, pp.225-230.

[7] H. S. Choi, J. R. Cha and J. H. Kim, "Fast Wireless Anti-Collision Algorithm in Ubiquitous ID System," in proc. of IEEE 60th Vehicular Technology Conference, 2004, pp.4589-4592.

[8] J. H. Choi, D. Lee and H. Lee, "Query Tree-based Reservation for Efficient RFID Tag Anti-Collision," IEEE Communications Letters, vol.11, no.1, 2007, pp.85-87.

[9] Draft Protocol Specification for a 900MHz Class 0 Radio Frequency Identification Tag, Auto-ID Center, 2003.

[10] EPCTM Radio-Frequency Identification Protocols Class 1 Generation-2 UHF RFID Protocol for Communication at 860-960 MHz Version 1.0.9, EPCGlobal Inc., Dec. 2005.

[11] K. Finkenzeller, RFID Handbook: Radio-Frequency Identification, Fundamentals and Applications, John Wiley & Sons Ltd, 1999.

[12] Information Technology-Radio Frequency Identification for Item Management-Part 6: Parameters for Air Interface Communications at 860 MHz to 960 MHz, Amendment 1: Extension with Type C and Update of Types A and B, ISO/IEC 18000-6:2004/Amd. 1:(E), Jun. 2006.

[13] D. K. Klair and K. W. Chin, "A Novel Anti-Collision Protocol for Energy Efficient Identification and Monitoring in RFID-enhanced WSNs," in proc. of 17th International Conference on Computer Communications and Networks, 2008, pp.1-8.

[14] Y. C. Lai and C. C. Lin, "Two Blocking Algorithms on Adaptive Binary Splitting: Single and Pair Resolutions for RFID Tag Identification," IEEE/ACM Trans. on Networking, vol.17, no.3, 2009, pp.962-975.

[15] C. Law, K. Lee and K. Y. Siu, "Efficient Memoryless Protocol for Tag Identification," in Proc. of the 4th International Workshop on Discrete Algorithm and Methods for Mobile Computing and Communication, 2000, pp.75-84.

[16] Y. Maguire and R. Pappu, "An Optimal Q-algorithm for the ISO 18000-6C RFID Protocol," IEEE Trans. on Automation Science and Engineering, vol.6, no.1, 2009, pp.16-24.

[17] J. Mosely and P. Humblet, "A Class of Efficient Contention Resolution Algorithms for Multiaccess Channels," IEEE Trans on Communication, vol.33, no.2, 1985, pp.145-151.

[18] J. Myung, W. Lee and T. K. Shih, "An Adaptive Memoryless Protocol for RFID Tag Collision Arbitration," IEEE Trans. on Multimedia, vol.8, 2006, pp.1096-1101.

[19] J. Myung, W. Lee, J. Srivastava and T. K. Shih, "Tag-Splitting: Adaptive Collision Arbitration Protocols for RFID Tag Identification," IEEE Trans. on Parallel and Distributed Systems, vol.18, no.6, 2007, pp.763-775.

[20] H. S. Ning, Y. Cong, Z. Q. Xu, T. Hong, J. C. Chao and Y. Zhang, "Performance Evaluation of RFID Anti-Collision Algorithm with FPGA Implementation," in proc. of 21st International Conference on Advanced Information Networking and Applications Workshops, 2007, pp.153-158.

[21] F. Schoute, "Dynamic Frame Length Aloha," IEEE Trans. on Communication, vol.31, no.4, 1983, pp.565-568.

[22] T. P. Wang, "Enhanced Binary Search with Cut-Through Operation for Anti-Collision in RFID Systems," IEEE Communications Letters, vol.10, no.4, 2006, pp.236-238.

[23] J. E. Wieselthier, A. Ephremides and L. A. Michaels, "An Exact Analysis and Performance Evaluation of Framed Aloha with Capture," IEEE Trans. on Communication, vol.37, no.2, 1989, pp.125-137.

[24] C. P. Wong and Q. Feng, "Grouping based Bit-Slot ALOHA Protocol for Tag Anti-Collision in RFID Systems," IEEE Communications Letters, vol.11, no.12, 2007, pp.946-948.

[25] K. H. Yeh, N. W. Lo and E. Winata, "An efficient tree-based tag identification protocol for RFID systems," in proc. of 22nd International Conference on Advanced Information Networking and Applications Workshops, 2008, pp.966-970.

[26] Chia-Chun Lin, A Tree-based Anti-collision Protocol for RFID Tag Identification in Supply Chain Network (master's thesis), National Taiwan Univ. of Science and Technology, Taipei, Taiwan, 2010.

Advanced Radio Frequency Identification Design and Applications

Edited by Dr Stevan Preradovic

Radio Frequency Identification (RFID) is a modern wireless data transmission and reception technique for applications including automatic identification, asset tracking and security surveillance. This book focuses on the advances in RFID tag antenna and ASIC design, novel chipless RFID tag design, security protocol enhancements along with some novel applications of RFID.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds