University of Stavanger

Faculty of Science and Technology

MASTER'S THESIS

Study program/ Specialization: Master of Science in Computer Science	Spring semester, 20.10..... Open / Restricted access
Writer: Shun Xue	…… (Writer's signature)
Faculty supervisor: Chunming Rong External supervisor(s):	
Titel of thesis: Multi-tag content access control in RFID system	
Credits (ECTS): 30	
Key words: RFID; Anti-collision protocol; Security technology; Content Access control; Modeling and simulation.	Pages: ……64…… + enclosure: …0…… Stavanger, …….. Date/year

Frontpage for master thesis
Faculty of Science and Technology
Decision made by the Dean October 30th 2009

Abstract

Radio Frequency Identification (RFID) makes great flexibility and high efficiency for data acquisition in industry and daily life. At the other side, it brings the privacy risks and multiple tags collision issue. Current research in RFID system focuses on the security and privacy issue which is based on authentication protocols between a tag and a Reader. There is a need to design a reasonable protocol which takes care of both multi-tag anti-collision and security issue.

This thesis presents a new protocol which combines both multi-tag anti-collision protocol and multi-tag content access control protocol. Firstly, the multi-tag anti-collision protocol and security protocol are analyzed and discussed. According to the theoretical basis, the new protocol is designed in order to meet the requirements of security and privacy for multi-tag in the context of anti-collision, improve the efficiency of identify process and the utilization of signal channel. This new protocol which uses hash lock is built on ALOHA anti-collision protocol. A priority register is introduced to the protocol, which is useful to deal with the multiple communications between Reader and tags.

Simulation of the proposed protocol is designed under the environment of Matlab, using the graphic modeling tool GPenSIM. The main parameters of the simulation are referenced from ISO/IEC 18000 international standard. The results shows that the new protocol performs well at content access control together with the anti-collision, in this way multiple tags can be identified in batch.

At the end of the thesis, one application about subway tickets access control system based on the new protocol is proposed.

Keywords：RFID；Anti-collision protocol；Security technology；Content Access control；Modeling and simulation

Acknowledges

The author would like to express her special gratitude to:

Prof. Dr. Chunming Rong from University of Stavanger;

Research fellow Liang Yan from University of Stavanger.

Index of Contents

Index of Figures

Index of Tables

Chapter1 Introduction

1.1 Background

Automatic Identification technology is a kind of methods about automatically identifying objects without human involvement. RFID reader can collect data from tags that attached to objects and store the data to the computer as inputs. With the help of computer system, highly automated information or data acquisition process can be achieved. Automatic Identification technology is developed rapidly and globally in recent decades. Using various media as the information carrier forms different automatic identification technology types, such as barcodes technology, biometrics technology, magnetic stripes technology, Radio Frequency Identification (RFID), Optical Character Recognition (OCR), and voice recognition, etc.[1]

Radio Frequency Identification (RFID) is a technology that uses radio waves for Automatic Identification that was developed in 1980's[1][2]. Typically RFID system depicted in Figure1 is composed by two parts: RFID Tags that attached to the objects and RFID Readers that can read information from tags. A RFID tag includes an integrated circuit that contains information about the object and an antenna to receive signals from RFID Readers and transmit information to RFID Readers.

According to different power support methods, RFID tags can be classified into two types: active RFID tags and passive RFID tags. Active tags contain batteries while passive tags need external power to provoke signal transmission instead. The maximum working range of passive tags is 10 meters. Active tags include battery assisted passive tags, or called semi passive tags, with the working range from 100 meters to 1000 meters respectively. Due to the different applications of RFID tags, there are five classes of RFID tags: Class0, which is passive tag, has Electronic Article Surveillance function but without storage. Class1 is read only RFID tag, which is used for identification. Class2 has read and write functions used for data logging. Both Class1 and Class2 tags

can be active or passive. Class3 and Class4 are active RFID tags which have both read and write functions. The difference is that Class3 is used for environmental sensor while Class4 is mainly for Ad hoc network[11].

RFID system has two normal working patterns: RTF(Reader Talks First) and TTF(Tag Talks First). RTF model is that Reader sends command to tags first, tags will reply when receive the command. To the opposite, tags will directly send messages to Reader in TTF model.

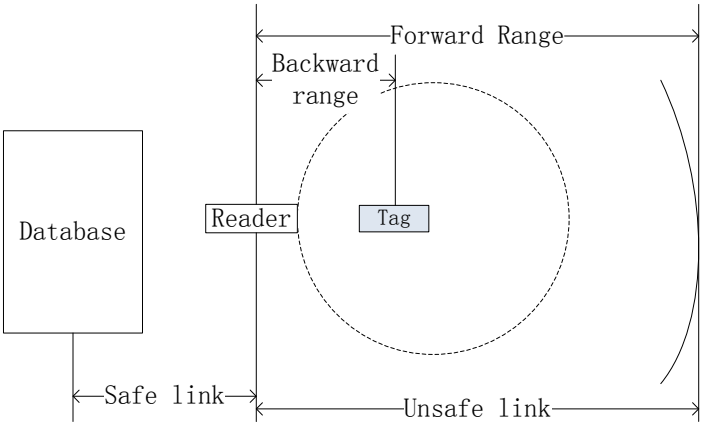


Figure1-1 RFID System

Recently RFID tags are used much more popular than traditional barcode. This is because RFID systems have many advantages compared with barcode.[37] First, RFID Reader does not use line of sight to read RFID tags while the barcodes must scan within the line of sight. Second, the valid distance between RFID tags and a Reader is much longer than the barcode system. Third, RFID Readers can scan tags in batch. Fourth, most of RFID tags use silicon technology, functions such as large memory for storage and calculation ability to support different kinds of security and privacy algorithms can be added. In addition, barcode cannot change the stored information which is imprinted, but the RFID tags with write function can change the stored information. According to these advantages, RFID tags are widely used in various areas, such as animal identification, social retailing and product tracking and so on. RFID technology helps los of companies to manage their productions and transportations to reduce their cost.

The technical characteristics of RFID bring some privacy problems because most RFID tags are readable by commodity RFID Readers. If RFID tags carry information

about the items that they are attached to, Readers can get some private information of the person who brings these items. With the large amount of RFID tags and Readers will be deployed in the near future, RFID privacy arouse more and more people attention. To protect the privacy of tag owner, tag access control is required. Access control means authorized Readers need to authenticate themselves to tag before getting tag's content. Unauthorized access to the tag's memory should be avoided.

When RFID Reader sends command to tags, tags respond simultaneously, the signals will interfere with each other. This is typically called a collision and the result of the communication is failed. If one Reader communicates with more than one tag, anti-collision methods must be employed. Due to the characters of RFID tags, the anti-collision has limited resources such as limited computation power and status information etc. Collisions might be difficult to detect because of the vary signal strengths between tags.

1.2 Related research

Multi-tag anti-collision technologies handle the collision in different ways, which can be concluded in Figure1-2.[3][4][5] TDMA is the most common technology among them. Tags can respond in probabilistic pattern or deterministic pattern. Most of probabilistic response algorithms are based on ALOHA. Before tag responds to the Reader, it does not check the channel whether busy or free. The efficiency of the protocol is influenced due to the rollback scheme. Some improved approaches, slotted ALOHA and dynamic frame ALOHA for example, are developed to increase the efficiency.[6] Reader can choose a certain tag to establish communication according to the tag's ID, this model is called deterministic. The most typical deterministic pattern is Binary Tree based. If the tags' response is related to the current status, then it is called Memorial algorithm such as Splitting Tree algorithm and Bit-Arbitration algorithm[7], otherwise the response is called Memoryless which means it is independent of the present status. For instance, Query Tree algorithm, Collision Tracking Tree algorithm,

and Tree-walking algorithm are classic ones.[8]

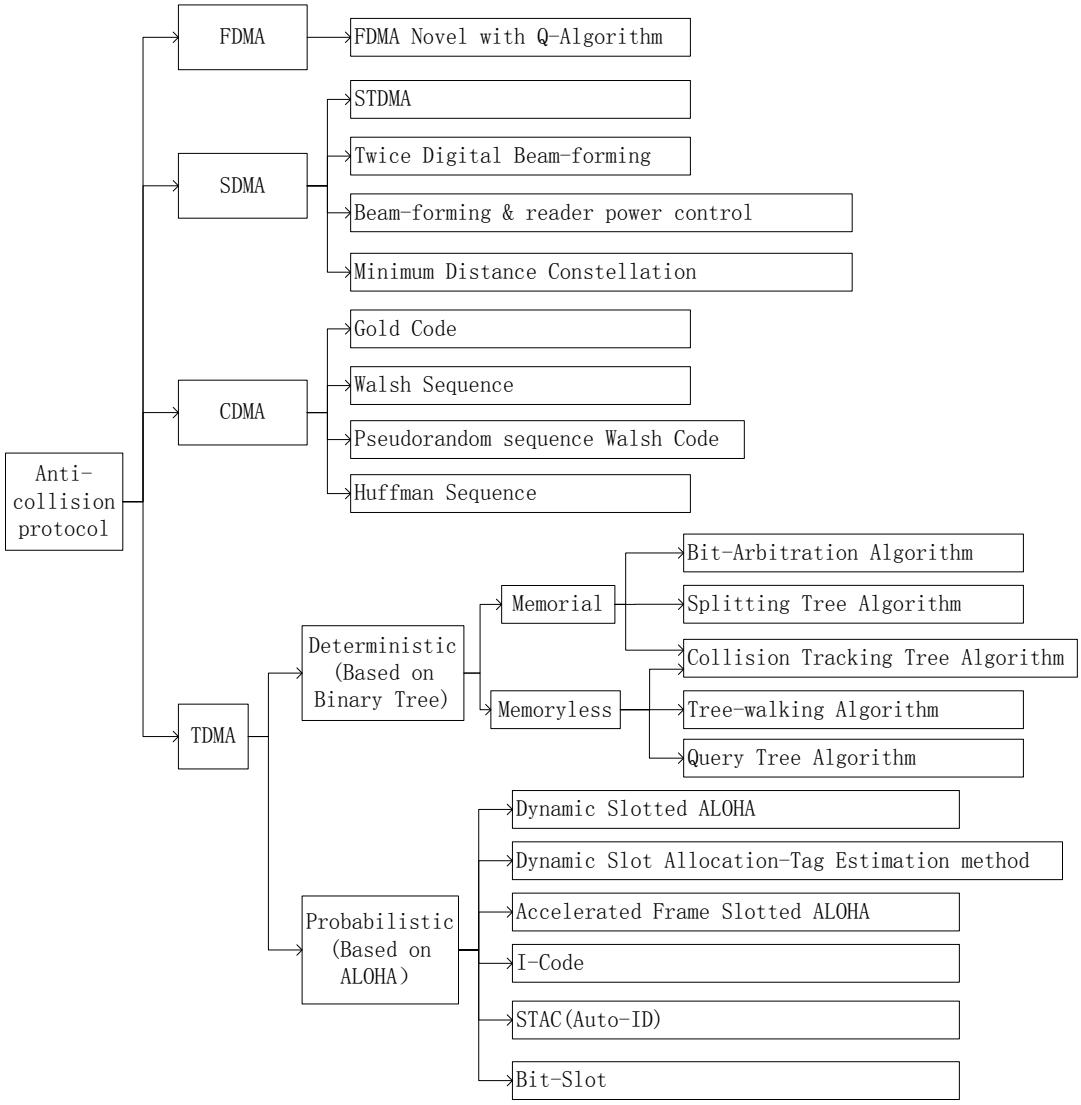


Figure1-2 Taxonomy of Anti-collision methods in RFID System

So far, there is no unique standard or regulation for RFID system. There are various of RFID standards including International Organization for Standardization (ISO), International Electrotechnical Commission (IEC), Ubiquitous ID (UID) from Japan and Electronic Product Code (EPC)[9] or EPCglobal mainly used in Europe. Among all these standards, ISO/IEC 18000[10] and EPC are the most widely used standard in RFID applications. In year 2006, the ultrahigh frequency standard which is proposed by EPCglobal is accepted as ISO/IEC 18000-6C regulation. RFID technology can be used in various frequency channels. These protocols have different multi-tag anti-collision algorithms as listed in Table 1-1.

Table 1-1 Anti-collision algorithms in ISO protocol

Protocol	Frequency	Anti-collision algorithm
ISO/IEC 18000-2	<135kHz	Query Tree
ISO/IEC 18000-3	13.56MHz	Query Tree, Pure ALOHA, Flame slotted ALOHA
ISO/IEC 18000-4	2.45GHz	Adaptive Binary Tree
ISO/IEC 18000-6	860-960MHz	Flame slotted ALOHA (type A/C), Adaptive Binary Tree (type B)
ISO/IEC 18000-7	433MHz	Flame slotted ALOHA

RFID system offers great efficiency but brings cost of both security and privacy. At present, there are physical and cryptography patterns to handle with these risks.[12][13] Physical scheme import extra components to protect the information privacy, normally like kill mechanism, shielding mechanism or interference mechanism, etc. All these manners can perform effectively but cost a bit more. Compare with the physical scheme, cryptograph methods do not need extra components. There are many cryptograph approaches proposed in recent researches[14][15][16], such as Hash-lock protocol, Hash-Chain protocol, David's Digital Library Protocol and Distributed RFID Query-Response Protocol and so on. All these protocols which are based on authority and authentication mechanism can protect the privacy to a certain extend. Risks, for instance physical vulnerabilities attacks, counterfeiting and eavesdropping, may threaten the unprotected tags. In order to satisfy more requirements of security and privacy, some improved protocol should be developed.

From the RFID system overview, anti-collision algorithms are widespread as well as security protocols. But the discussion about these two kinds of protocols is quite separately[17][18][19], i.e. the security issue is mainly discussed under one Reader and one tag system. However, security and privacy protocol should be introduced together with anti-collision algorithms in multi-tag applications.[20][21] It is significant to design a reasonable multi-tag content access control protocol in order to guarantee the privacy,

system efficiency and ultimate channel capacity as well.

1.3 Thesis overview

This thesis proposed a multi-tag content access control protocol, which combines both multi-tag anti-collision protocol and security and privacy protocol, to meet the requirements of security and privacy for multi-tag in the context of anti-collision, and to improve efficiency of identify process and utilization of signal channel. A priority register mechanism is introduced into this proposed protocol, which is useful during the frequent communications between Reader and tags. Modeling and simulation of the new protocol is designed under the environment of Matlab. The simulation results state clearly that this presented protocol performs well at content access control together with the anti-collision. In this way, multiple tags can be identified in batch.

The reminder of this thesis is organized as follow:

Chapter 1 introduces the background of RFID system, security and multi-tag collision issues.

Chapter 2 contains a brief discussion and analysis about some classic anti-collision algorithms and security and privacy protocols, some related research approaches and proposed multi-tag access control problems.

Chapter 3 presents the access control protocol for a multi-tag system. The system performance is discussed based on theoretical derivation, and an alternative approach is proposed.

Chapter 4 is the modeling and simulation part. The proposed protocol model is designed under the environment of Matlab.

Chapter 5 is conclusion and outlines future research. It illustrates one open application about subway tickets access control system based on the new protocol.

Chapter2 Theory

2.1 Multi-tag anti-collision protocol

2.1.1 Multi-tag collision issue

In RFID system, when more than one tag send signal to one Reader simultaneously, the signals will be interfered with each other, as Figure 2-1 illustrates. Reader can receive signal from Tag1 correctly during period t1. At period t3, Tag2 and Tag3 send signals to Reader while collision occurs at period t2. Under this situation, the Reader cannot recognize the correct signal either from Tag2 or Tag3, and this is called incomplete collision. Period t4 shows the complete collision between Tag2 and Tag3. Collisions caused by conflicting communication signals will influence the system efficiency and the data transmission rate. Multi-tag system must employ an anti-collision algorithm to avoid these collisions.

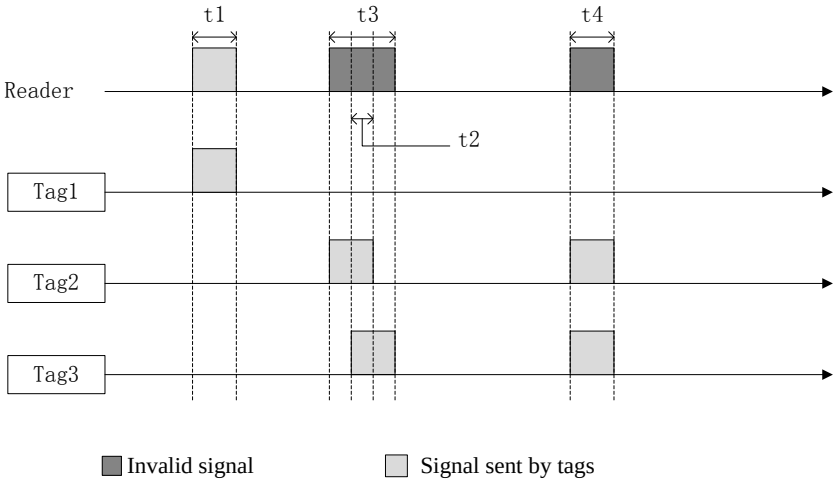


Figure2-1 Multi-tag collision in RFID system（Based on ALOHA）

One Reader can communicate with multiple tags at the same time by using FDMA, SDMA, CDMA or TDMA technologies. Frequency Division Multiple Access, or FDMA, assigns several frequency bands or channels to multiple tags individually. Space Division Multiple Access, or SDMA, creates parallel spatial multiplexing to the

communications between Reader and multiple tags. Code Division Multiple Access, or CDMA, which is a kind of spread spectrum technology, offers several tags to share a bandwidth of different frequencies by using special coding scheme. In this way, multiple tags can communicate with one Reader simultaneously over one channel. Time Division Multiple Access (TDMA) divides access by time, which means multi-tag share one frequency channel by sending signals at different time slots. Two common anti-collision methods using TDMA is tag control method which is mainly considered as ALOHA method and Reader control method which is Binary Tree based. Due to low cost and limited resources of RFID system, TDMA becomes the most common multiplexing technology in various applications.

2.1.2 Anti-collision protocols

Binary Tree based algorithms[22][23] that using Manchester coding scheme, require precise timing synchronization to polling all tags. Meanwhile the collision bits can be marked by using coding methods. If a conflict bit is detected, the tags will be divided into two parts. For instance, as Figure2-2 illustrates there are five tags: tag010, tag 100, tag101, tag 110 and tag 111. The collision occurs at the very first bit. So the tags divided into ‘0’ and ‘1’ index. And then, tag010 is selected, but the rest of the tags still conflict by ‘1’ index. Repeat the divided step, tag100 will be selected by twice filter and so as the rest.

Binary search algorithm has good rate of identification which can reach 50% efficiency without identify error. The weaknesses of this algorithm are long time delay and poor security and privacy. From the other side, if a more complicated coding scheme is employed, the process of identification will becomes much tougher.

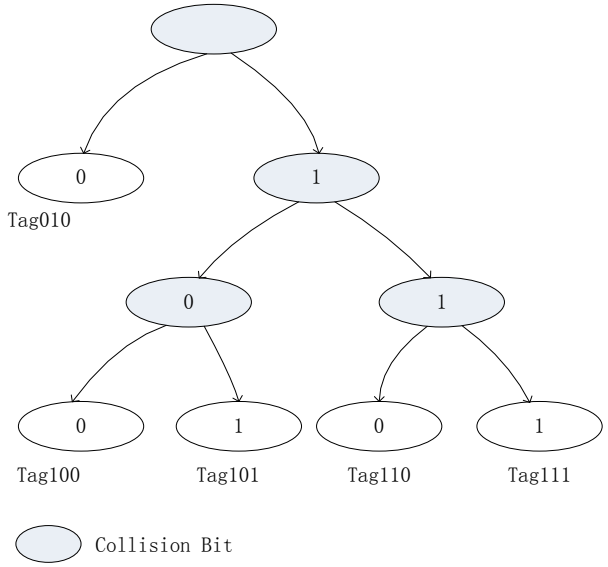


Figure2-2 Example of Tree Based Algorithm

Pure ALOHA anti-collision algorithm[24] is the simplest method based on TDMA which allows tags to send signals after random time t. If more than one tag sends signal to Reader when Reader is communicating with another tag, signals will conflict with each other and none of the signals will be received correctly. Tags are required to send the signal again once the Reader detects the collision.

Both incomplete and complete collision exist in the pure ALOHA algorithm with the maximum throughput rate 18.4%. This anti-collision algorithm is costly and normally applied to the tag read only system which has uncertain number of tags.

Slotted ALOHA[25] which is a Reader-driven scheme allows RFID tags to send signals at a certain time slot. As Figure 2-3 shows, time slots t1 and t2 are the correct periods that Reader can identify the certain tag, Tag1 and Tag2. t3 is a collision time slot while t4 is an idle one. Slotted ALOHA divides time to several slots and tags have to send signal at one slot, which means there is no incomplete collision but only complete collisions. Thus, the collision rate is half compared to pure ALOHA, and the efficiency, or throughput rate, of slotted ALOHA system is as twice as pure ALOHA. This higher performance requires the Reader providing a necessary synchronization.

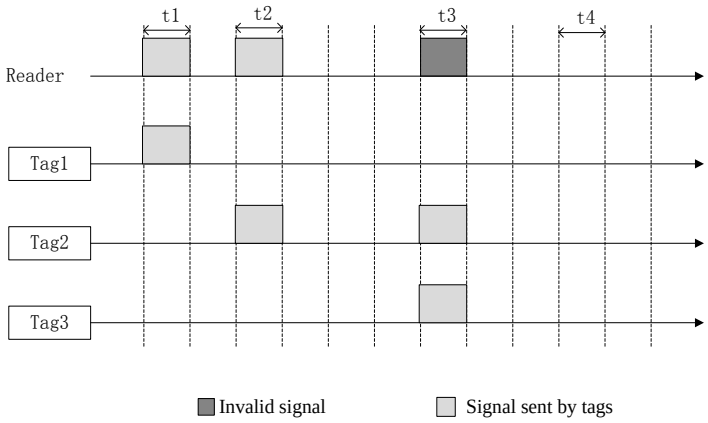


Figure2-3 Example of Slotted ALOHA

Framed Slotted ALOHA algorithm pack several time slots as one frame, tags must select a certain slot from one frame to transmit signal. As Figure 2-4 illustrates, t0 is one frame which contains several time slots. All tags have to send their data in t0. Slots t1 and t2 are no conflict period when Reader can identify Tag1 and Tag4 correctly, while t3 is a collision slot and t4 is an idle one. The collision tags will send their data at the next frame. It is clearly that the possibility of collisions by using framed slotted ALOHA is reduced significantly compared with the slotted ALOHA algorithm.

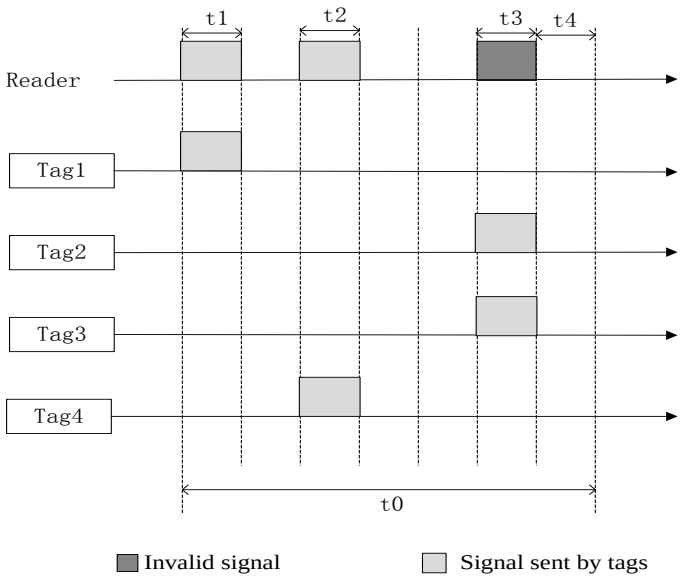


Figure2-4 Example of Framed slotted ALOHA

Framed Slotted ALOHA which uses fixed frame size brings a waste when frame size is too long while the number of tags is small, or causes more collisions when there are huge numbers of tags but short frame size. To deal with this shortcoming, Dynamic

Framed Slotted ALOHA[26] is developed by using changeable frame size. As a result, it increases the system efficiency significantly.

According to the theoretical researches[3][27], assume that all the frames have the same length; and the data transmitted from tags obeys Poisson distribution. Let 'S' be the system throughput and 'G' be the mean of the transmission amounts, i.e. there will be G transmission times in one frame.

So in the Pure ALOHA algorithm, the average utilization or throughput is:

$$S = G \times e^{-2G} \tag{2.1}$$

From equation (2.1), it is easy to see that system will be stable only if $G \leq 0.5$, and the maximum throughput is 18.4 % when $G = 0.5$. This means only in 18.4% of the whole process, Reader can receive correct signals.

The throughput of Slotted AOLHA algorithm is:

$$S = G \times e^{-G} \tag{2.2}$$

From equation (2.2), it shows that system will be stable only if $G \leq 1$, and the maximum throughput is 36.8%.

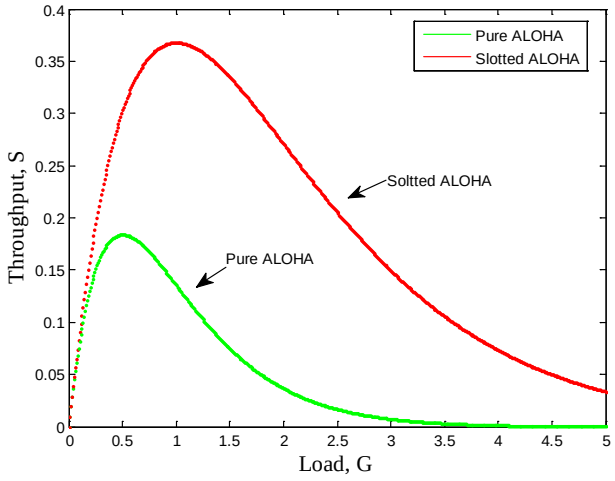


Figure 2-5 Throughput curves of Pure ALOHA and Slotted ALOHA

Average packet delay rate should be discussed in ALOHA algorithms as well. The total time is the sum of transmission time and channel link time, the waiting time is not included. Assume that $\overline{B}$ is the average rollback delay which is the variation in various statistical functions; $\overline{D}$ is the average data delay.

In Pure ALOHA algorithm:

$$\overline{D} = e^{-2G} + (e^{-2G} - 1)\overline{B} \tag{2.3}$$

While in Slotted ALOHA algorithm:

$$\overline{D} = 0.5 + e^{-G} + (e^{-G} - 1)\overline{B} \tag{2.4}$$

From equation (2.3) and (2.4), $\overline{D}$ has its minimum value when $\overline{B} = 0$.

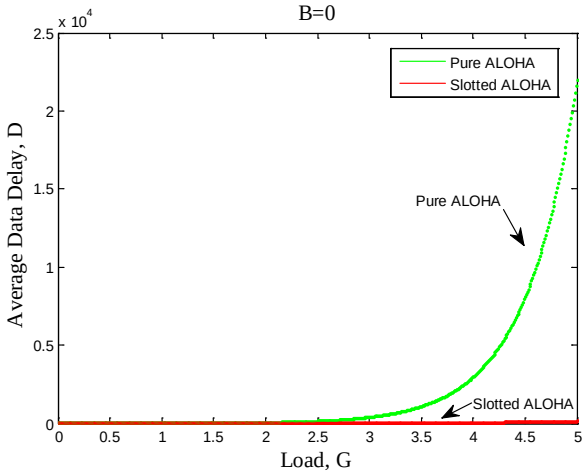


Figure 2-6 Average Data Delay curves of Pure ALOHA and Slotted ALOHA

In [4], [26] and [27] there are some improved slotted ALOHA algorithms which can increase the system efficiency, but all these algorithms cannot ensure all the tags can be identified correctly. When the number of tags is increasing, the number of unrecognized tags increases. Assume that there are N tags, and M slots for collision detection, the probability of the number of tags in one slot, q, obeys binomial distribution:

$$P_q = \binom{N}{q}\left(\frac{1}{M}\right)^q\left(1 - \frac{1}{M}\right)^{N-q} \tag{2.5}$$

If there is only one tag sends signal to Reader, there is no collision. The probability of only one tag occupying one slot is P_1:

$$P_1 = N\left(\frac{1}{M}\right)\left(1 - \frac{1}{M}\right)^{N-1} \tag{2.6}$$

So, the number of signals received correctly is N_r:

$$N_r = M \times P_1 = N \times \left(1 - \frac{1}{M}\right)^{N-1} \tag{2.7}$$

Thus, system efficiency is:

$$E = \frac{\sum \text{time of receive signalcorrectly}}{\sum \text{time}} = \frac{N_r \times t_r}{N_r \times t_r + t \times M} \tag{2.8}$$

In (2.8) t_r refers to the time of receiving data, while t is the time of detecting the collision.

Combine equation (2.6) and (2.7), and then:

$$E = \frac{N \times \left(1 - \frac{1}{M}\right)^{N-1} \times \left(\frac{t_r}{t \times M}\right)}{1 + N \times \left(1 - \frac{1}{M}\right)^{N-1} \times \left(\frac{t_r}{t \times M}\right)} \tag{2.9}$$

Figure 2-7 illustrates the efficiency of improved slotted ALOHA, when $\frac{t_r}{t} = 20$ or 50, $M = 64$ or 128. According to this result, if $\frac{t_r}{t}$ keeps the same, there is an intersection point with different M values. When the RFID system has a small number of tags (as the Figure 2-7 shows the point of intersection is around 100 tags, i.e. when the tag number is less than 100, it can be considered as a small size RFID system.), lower M value makes higher efficiency.

On the contrary, if there are a large number of tags in RFID system (more than 100 tags in this example), larger M value is needed to get high efficiency. From other side, if M keeps the same, the efficiency has no point of intersection with value of various $\frac{t_r}{t}$ values. Higher efficiency can be achieved by increasing $\frac{t_r}{t}$ value. If both M and $\frac{t_r}{t}$ increase, the system's ability to handle large number of tags will increase.

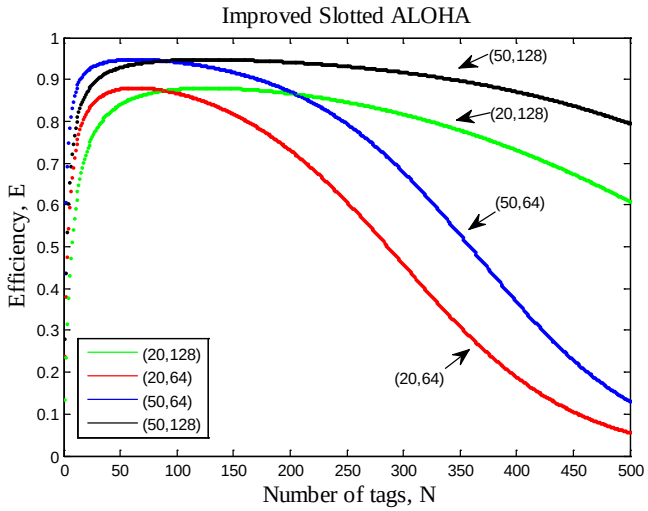


Figure 2-7 Efficiency curves of Improved slotted ALOHA

Based on the above analysis of anti-collision algorithms, normal ALOHA protocols

are only suitable for RFID system with low speed and small number of tags due to the hardware limitation. Tree-based approaches can be used in RFID system with large number of tags, but the exposed information may cause problem of security and privacy. According to some recent research works[28-32], this thesis is only consider ALOHA based algorithm which performs well in small size RFID systems.

2.2 RFID security protocol

2.2.1 Security and privacy issue

One problem related with RFID system is unauthorized Reader can read the tag information from large distance. For example, even ISO 14443 specifies that the nominal read range of a contactless smartcard tag is about 10cm, Readers equipped with powerful antenna can scan this kind of tags at a range of 50cm[12]. Because the communication between the Reader and tag is wireless, it is hard for the tag owner to indicate which tag to read. The threat to people privacy grows when the tag information is combined with personal information. For example, a standard EPC tag will response with its identity to any Reader request. According to EPCglobal Tag Data Standards[38], this tag identity may include information about manufacture name, item type and serial number. Using this information, Readers can harvest some private information about person who brings this tag. For example, from the type of medicine a people brings, Reader can deduce the illness type that this person is suffering from. Another example is e-passports that using RFID technology. RFID tag in this e-passport carries not only some sensitive data such as holder’s name, birthday and nationality but also some kinds of biometrics data such as people’s facial image, iris and fingerprint data that used for biometric authentication. Leakage of these kinds of information will bring problem not only for passport authentication, but also bring potential security problems for other biometric systems.

2.2.2 Security and privacy approaches

RFID system is faced with various security and privacy challenges, but this thesis focuses on the content access control issue. Clearly, to protect the privacy of tag owner, tag contents should be read only by authorized Readers.

In [12], [30] and [34] etc, some approaches about RFID tag contents access controlling are introduced. These RFID privacy protection approaches can be classified as: killing and sleeping, agent scheme and on-tag encryption scheme.

One kind of killing approach is used in EPC Gen-2 standard[13]. In this scheme, to protect the privacy of the consumer, tag that attached to an item will be killed at the point of sale. This means after a tag receives a kill command, it will not response to any query. This kill command will be sent together with a tag-specific PIN to prevent illegal killing of tags. Killing the tag can protect the tag's content and consumer's privacy, but it will discard all the information of tags. Similar with this killing approach is the sleeping approach.

For sleeping approach, tag can enter a sleeping state and can be waken when needed. To wake a sleeping tag, user also required to transmit a tag specific PIN to the tag. How to manage the PINs for different tags may be a problem for the users.

People can also use some privacy–enforcing devices to support tag content access control. These devices can be carried separated by the consumer or be integrated into a mobile phone. For example, in [35], to protect the consumer privacy, a watchdog tag is used to decode and show the command transmitted by a Reader to help the consumer to judge whether this Reader will hurt his privacy. But these devices have more computing power and more sophisticated polices that can be used for privacy protection. For example, they can jam the communication between the Reader and tag if the tag is scanned by a Reader which is not located at a certain place. Challenges related with agent scheme include: consumers sometimes need to specify their privacy policies by themselves; these agents needs to understand these consumers' privacy policies.

On-tag schemes are the methods that Readers and tags can communicate directly

and tags will control access to their contents. Because some RFID tags can offer on-chip computation, most of on-tag schemes are based on encryption. Thus, on-tag encryption schemes are thought to be more secure for tag access control.

The first hash-lock access control approach was proposed by Weis[14][33] to prevent unauthorized Readers from reading tag contents. In this approach as shown in Figure 2-8, each RFID tag will have a temporary *metaID* and will work in either a locked or unlocked state. To lock a tag, tag owner will first hash a *key* and this hashed value will be stored in the tag's memory as *metaID*=H（*key*）. The tag owner will store *key* and *metaID* in a back-end database. After receiving this *metaID* from tag owner, tag will turn to locked state. At locked state, tag will response its *metaID* to all Reader interrogation without other functionality. If tag owner wants to unlock a tag, it will query the tag to get this tag's *metaID* and use this *metaID* to get *key* from the back-end database. Then the tag owner sends *key* to tag. Then, tag will hash the received *key* and compare the hashed value with its *metaID* that stored in his memory. If they match, tag will enter unlocked state and offers its full functionality to Reader.

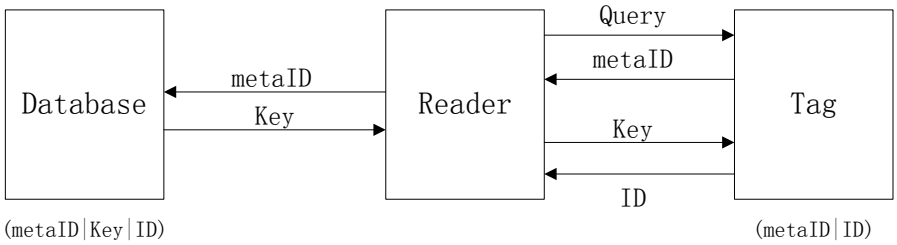


Figure 2-8 Hash-Lock access control scheme

There are some disadvantages with this hash-lock access control protocol. First, tags need to compute a cryptographic hash function. Secondly, this approach has a key management issue. In this hash-lock access control protocol, each Reader that wants to access tags content is required to communicate with back-end database to get the key required by the protocol. Meanwhile, if any user of a tag wants to know the content of a tag, this user is also required to have rights to access these databases as well. This leads to the problem of key distribution and access. But this thesis is not going to discuss about this issue.

Randomized Hash-Lock access protocol has a random number generator which is based on Hash-Lock function. This protocol is summarized as follow and it is illustrated in Figure 2-9.

(1) Reader sends a *Query* to tags.

(2) Tag generates a random number *R* and hashes the (*ID, R*) pair. Then tag responses the hash result *M1*=H（*ID*，*R*）together with the random number *R* to the Reader.

(3) After receives the (*R*| H*(ID,R)*), Reader will connect with the back-end database to search all legitimate tags in its database.

(4) The database responses all the tags' ID which the Reader has the authority to access the content.

(5) Reader will hash all IDs with the random number *R*, in order to find one matched tag ID_t who has the same hash value as *M1*. If there is one ID matches, Reader will send the ID_t to tag.

(6) Tag will compare the ID_t with its own *ID*. If the two IDs are the same, then the tag would be unlocked.

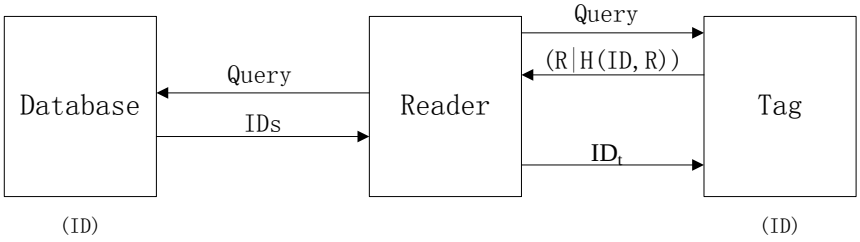


Figure 2-9 Randomized Hash-Lock access control scheme

Another on-tag access control approach is to use public-key authentication[36]. In this approach, each Reader and tag store its own private key and other party's public key.[37] The Figure 2-10 shows the scheme of key generation and distribution.

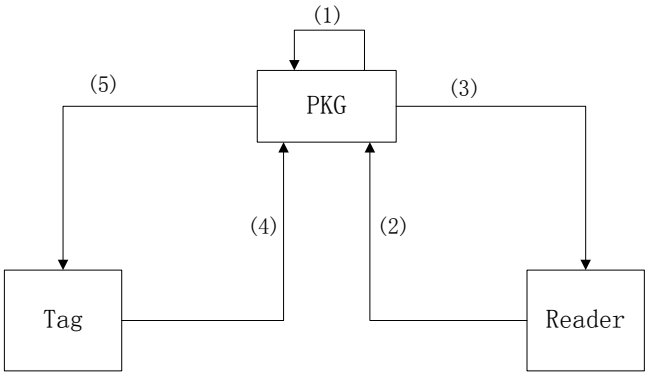


Figure 2-10 Key generation and distribution

PKG is the Public Key Generator which generates the key pair. First, PKG generates a public key and a relative private key. When Reader sends ID to PKG, which is process (2), PKG will send the public key to Reader which refers to step (3). Each tag uses its ID to authenticate itself by step (4). At step (5), PKG will respond the valid ID a private key which is unique to the certain tag together with the public key.

If a Reader wants to access the content of a tag, it has to authenticate itself to this tag first. Reader will send a query to this tag and this tag will response with a random data. Reader uses its private key to encrypt this data, and then sends it back to the tag. After the tag decrypts the received cipher text and compares to the original data, the tag can verify whether this Reader is an authorized one. The shortcoming of public key cryptography is it requires the tag to do complex computations. Considering the resources of low-cost RFID tags, it maybe not easy to implement a public key authentication protocol when the tag offers low cost.

2.3 Summary

This chapter first discussed several multi-tag anti-collision protocols. After specified the principle of multi-tag collision issue, different kind of anti-collision algorithms are compared and analyzed which is prepared for the anti-collision part of proposed protocol.

As the same to security and privacy protocol, most common access control approaches are described and discussed in this chapter in order to propose an improved protocol.

Chapter 3 Multi-tag Content Access Control Protocol

3.1 Communication protocol model between Reader and tag

The communication protocol between Reader and tags can be illustrates as Figure 3-1. Multi- tag collisions happen at the communication layer, while the authentication process mainly takes place at the application layer. Therefore, there is a challenge to protect the security and privacy together with the anti-collision methods. In [20] and [21], some relative approaches are proposed.

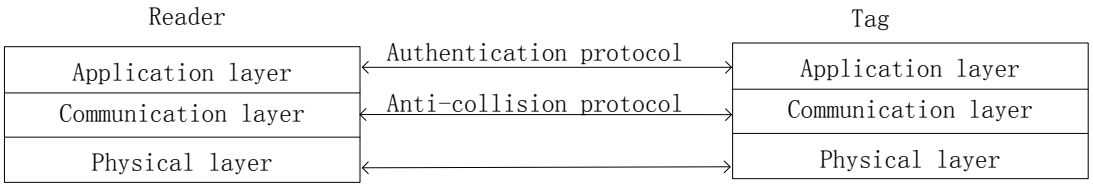


Figure 3-1 Communication Model between Reader and Tag

3.2 Anti-collision proposal

Design a reasonable anti-collision protocol need to consider the following metrics. First, all the tags who is active in the Reader's valid working extent should be recognized correctly. Second is the performance, i.e. identification efficiency, security, noise and error tolerance, should satisfy the application requirements. Third, the power cost and bandwidth requirements should not be too much. Except to make sure that all the tags can be searched by Reader, the key factor is the communication throughput between Reader and tags, which influences the system efficiency and power consumption directly. Last part of this section will present an anti-collision proposal based on dynamic frame ALOHA.

3.2.1 Complete monitor anti-collision protocol

In this proposal, we assume RFID tags have the ability to monitor the

communication channel, i.e. active tag can detect the conflict when there is more than one tag send signal to Reader at the same time. Here is one example to implement the monitor function. Suppose the signal ‘0’ is the coding of ‘1100’ while ‘1010’ stands for signal ‘1’. If the tag receives signal such as ‘1000’ and other check bits indicate the error as well, it can be thought as a collision.

Reader broadcasts *Query* to all tags at its working range first. Once a tag receives the *Query*, it will become active and start to monitor the channel. A random time Δt is introduced in this proposal. If the link is occupied at this moment, the tag will keep on listening the situation of channel until the link is free during Δt. The signal will transmit to Reader when the tag believes the link is available. This monitor mechanism works during the whole transmission procedure. Once conflict is detected, the tag will rollback and send the signal again when the channel is free. This algorithm is described as Figure 3-2.

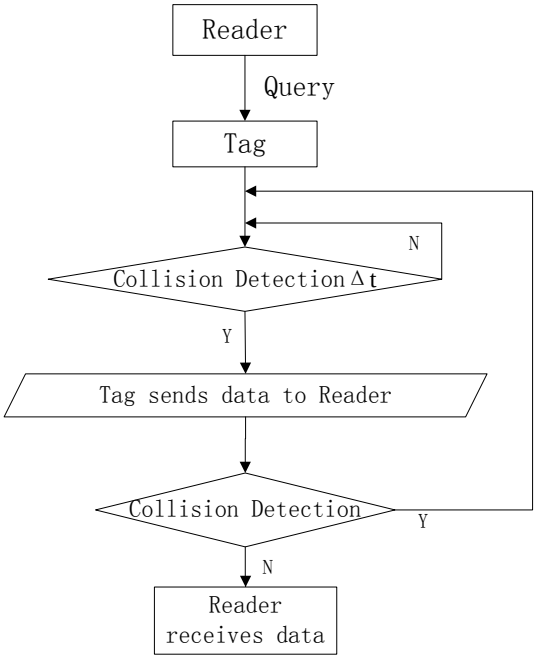


Figure 3-2 Anti-collision with channel monitor

This algorithm guarantees that all tags can be identified with the cost of high channel throughput without incomplete collision. Tags should listen to the link through the whole communication procedure. Thus, the power consume is more than other normal ALOHA-based approaches. The advantage of this algorithm is easy to

implement and suitable for the dynamic system with uncertain number of tags. As same as other anti-collision protocols, the proposed system will be crash or breakdown when attacker occupies the channel all the time, therefore the useful signals are interfered and the efficiency will decrease because none valid signal can be received correctly.

3.2.2 Incomplete Monitor Anti-Collision Protocol

In ALOHA based protocol, there are some tags which can be thought as hide nodes whose signal is stronger than the signal of tag who is transmitting data. This hide nodes problem is illustrated in Figure 3-3. Within the Reader's working range, Reader can communication with tag A, B, C, D and E. Tag F is not included in this Reader's working group, and for the simplest assumption that tag F can be considered as working at a different frequency, i.e. it won't interfere the present RFID system.

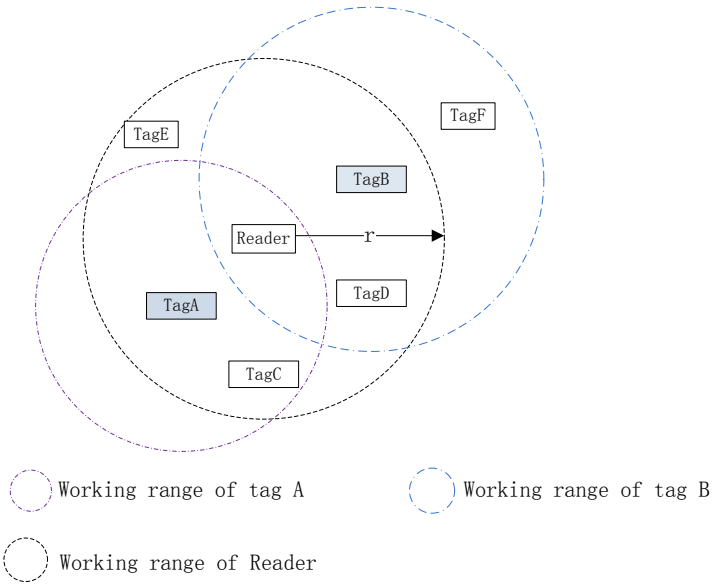


Figure 3-3 Example of hide node problem

Considering about tag A, within its working range, it can detect whether tag C is transmitting signals to Reader. Tag D can be heard by tag B when tag D is communicating with the Reader. But if tag A and tag B send data to the Reader at the same time t, both tag A and B cannot detect each other due to so weak signal which might be considered as noise. And then, the Reader will receive conflicted data and the monitor scheme is invalid as well. Thus, the detecting range of tag should choose

reasonable. There is a similar situation that occurs to multi-Reader system, but this thesis will not focus on that issue.

When considering cost and efficiency of monitor mechanism, a new protocol is proposed in this section. One incomplete monitor approach is proposed that is tag only listens to the channel before send the signal. The idea of this approach is that tag will not keep listening during all communication procedure. Figure 3-4 is the incomplete monitor protocol.

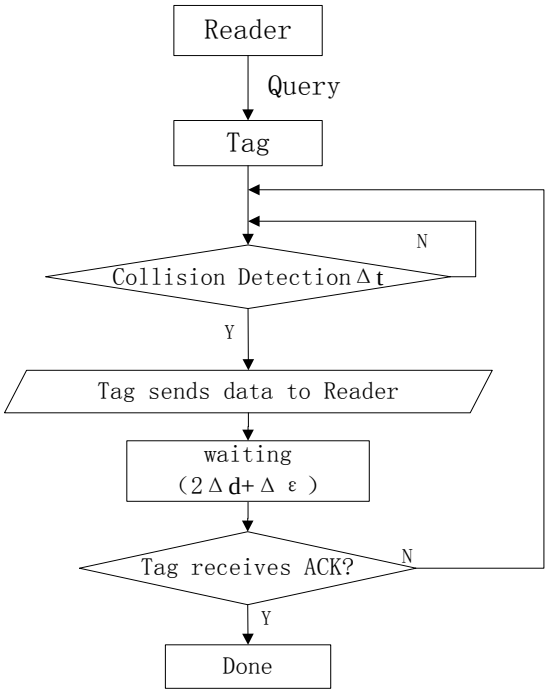


Figure 3-4 Incomplete Monitor Anti-Collision Protocol

Tag will be active when receives the *Query* from Reader. It starts to listen to the channel, check whether the link is available. This step is the same as complete monitor protocol, after keep listening the idle link for a random time Δt, the tag will send data to Reader. And tag will send signal again after each $(2\Delta d + \Delta\varepsilon)$ time period, where Δd refers to the maximum data delay according to the Reader's working range. After the Reader receives the correct signal, one ACK message will be sent to the corresponding tag from reader. Once the tag receives the ACK, it will stop to send signal every $(2\Delta d + \Delta\varepsilon)$ time period which is not periodicity repeating.

By using ACK mechanism to response the communication, the requirement of keep detecting system throughout is not necessary any more. Although the reliability of

identification will increase, the load of system link becomes much heavier. This protocol will lose its advantage if the system has a large number of tags in or the tag data size is big size, because this will result in effective data reduction and system performance degradation.

3.2.3 Theoretical derivation and analysis

Assume that there are N tags in the system. Regardless of the data rollback delay, the probability of *q* tags sending signals in random time Δt obeys binomial distribution:

$$P_q = \binom{N}{q}\left(\frac{1}{N}\right)^q\left(1-\frac{1}{N}\right)^{N-q} \tag{3.1}$$

If there is only one tag sends signal to Reader, there is no collision, and the probability of only one tag sending data in Δt is P_1:

$$P_1 = \left(1-\frac{1}{N}\right)^{N-1} \tag{3.2}$$

So, the number of receive correct signals is N_r:

$$N_r = N \times P_1 = N \times \left(1-\frac{1}{N}\right)^{N-1} \tag{3.3}$$

Thus, system efficiency is:

$$E = \frac{\sum \text{time of receive signalcorrectly}}{\sum \text{time}} = \frac{N_r \times t_r}{N_r \times t_r + \Delta t \times N} \tag{3.4}$$

In equation (3.4) t_r refers to the time of receiving data, while Δt is the time of detecting the collision.

Combine equation (3.2) and (3.3), then :

$$E = \frac{N\times\left(1-\frac{1}{N}\right)^{N-1}\times\left(\frac{t_r}{\Delta t\times N}\right)}{1+N\times\left(1-\frac{1}{N}\right)^{N-1}\times\left(\frac{t_r}{\Delta t\times N}\right)} = \frac{\left(1-\frac{1}{N}\right)^{N-1}\times\left(\frac{t_r}{\Delta t}\right)}{1+\left(1-\frac{1}{N}\right)^{N-1}\times\left(\frac{t_r}{\Delta t}\right)} \tag{3.5}$$

Let $u = \frac{t_r}{\Delta t}$,

$$w = \left(1-\frac{1}{N}\right)^{N-1}\times\left(\frac{t_r}{\Delta t}\right) = \left(1-\frac{1}{N}\right)^{N-1}\times u \tag{3.6}$$

thus, the system efficiency is:

$$E = \frac{w}{1+w} \tag{3.7}$$

When the system has large size of N, i.e. there are a larger number of tags, the system

efficiency is depends on u.

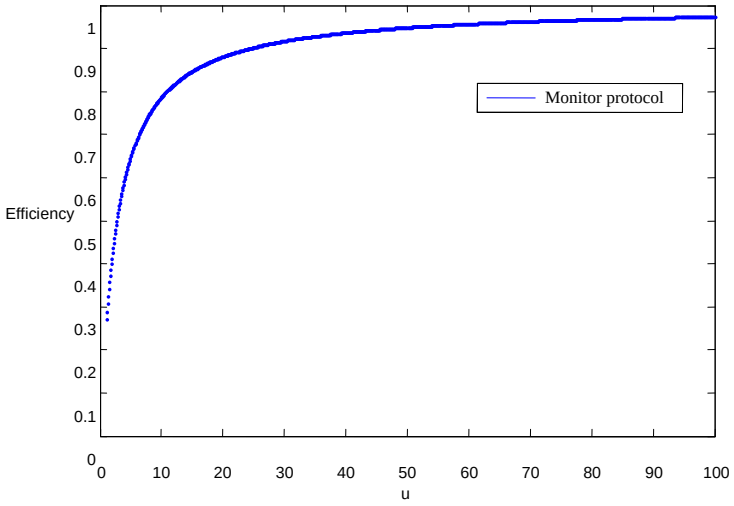


Figure 3-5 Efficiency curves of monitor protocol

From Figure 3-5, it is obvious that the system has a larger efficiency when larger value of $u = \frac{t_r}{\Delta t}$ is employed.

3.3 Security proposal using Hash-Lock

According to the research and analysis in Chapter2, this section introduces a Hash-Lock based content access control method as a part of the protocol which will be given in the next section.

Current Hash-Lock based approaches cannot handle the security or privacy issue in parallel. That means the Reader cannot gain the content access permits from multiple tags at the same time. In order to implement the multi-tag content access control, this new method introduces a priority register which is used to assign the priority of multiple tags

The structure of the register is designed as Figure 3-6. Assuming the length of the priority is 16bits. The 10 low bits are assigned to represent the priority of tags, while the rest 6 high bits are set to ‘0’ as reservation. If the length of priority is 8bits, then the system can arrange 2^8=256 tags at most, however 2^{10}=1024 tags can satisfy most of the RFID applications’ requirements right now.

Table 3-1 The structure of the priority register

MetaID	R(Priority)
Fixed length(e.g. 160bits)	16bits(10bits for priority)

The initial value of the priority register *R* is zero. Once a tag enters the queue, the register assigns the current value of R to the tag, and the value of R will add one. The register will empty the R values when there is a clear command.

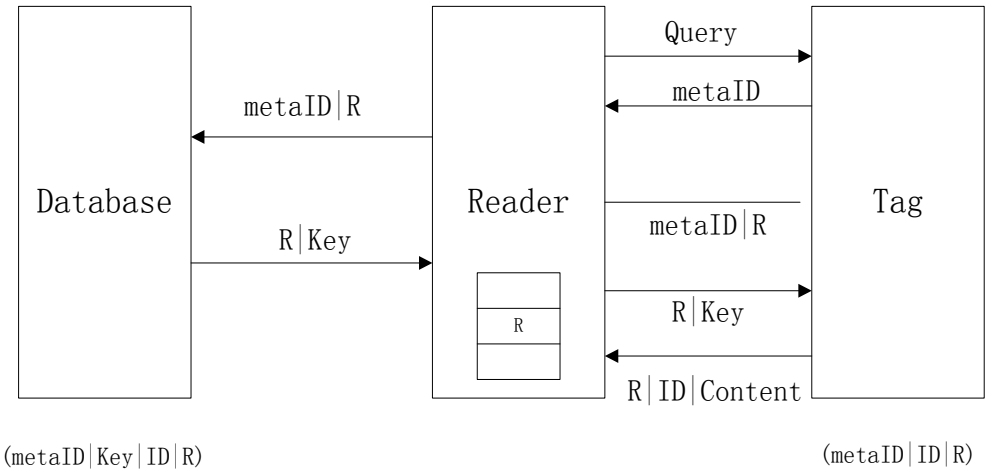


Figure 3-7 Hash-Lock based protocol

Figure 3-7 is the protocol for multi-tag communication with one Reader based on Hash-Lock. Reader will broadcast the *Query* to tags at first. When the tag receives the *Query*, it will respond its *metaID* which is stored in memory already. Reader will put the *metaID* into the priority register as soon as received, and assigns the priority value *R*. Then the data (*metaID*|*R*) is sent to both tag and back-end database. The database looks for the Hash table with *metaID* and picks up the matched *key*. By using the *R* value as index, database will return (*R*|*key*) to Reader. And (*R*|*key*) will be sent to tag. When tag gets the *key*, it will hash the *key*, and compares the new *metaID*=H(*key*) with the one in memory. If the two values are same, the Reader is considered to be an authorized one, and the privacy communication can continue. Otherwise, the Reader cannot access more protected information about the tag.

As mentioned above, the priority value *R* plays a role as tag's second ID. It brings a lot of convenience. It is a fast path for frequent communication between tag and

Reader, and supports an optimized approach for multi-tag access control with anti-collision which will be discussed in section 3.4.

3.4 Multi-tag content access control protocol using Hash-Lock

3.4.1 Procedure of proposed protocol

This section presents a multi-tag content access control protocol using proposed anti-collision algorithm in section 3.2.1 and Hash-Lock based method in section 3.3. By combining communication layer and application layer, this proposed protocol is designed to meet the privacy requirements in anti-collision algorithm and the efficiency demands in multi-tag authentication procedure. The detail of the protocol is illustrated in Figure 3-8.

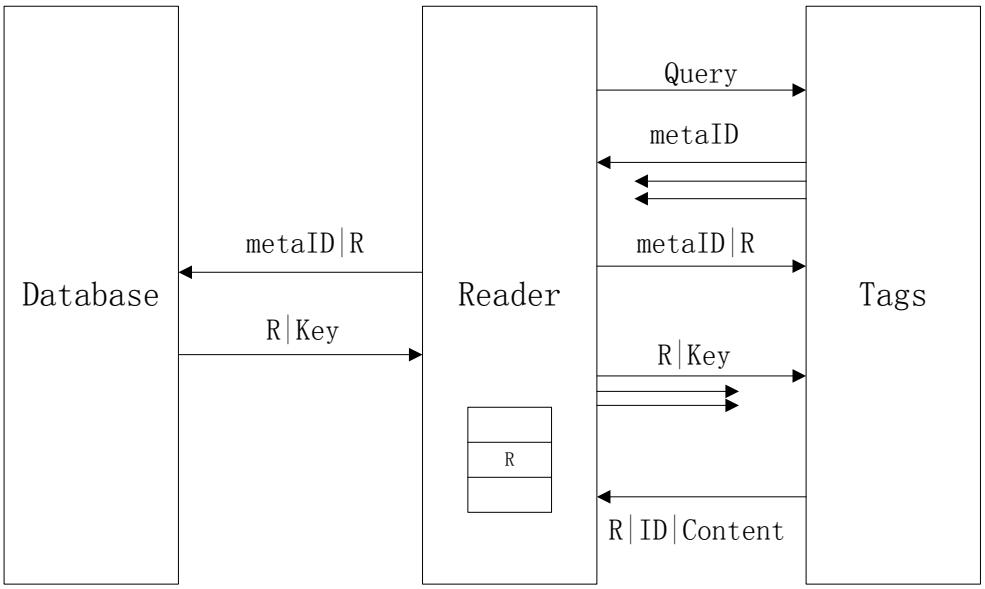


Figure 3-8 Multi-tag content access control protocol using Hash-Lock

The input and output of Reader should be serial. The link of Reader and back-end database is considered as reliable. The communication between Reader and tags is based on the monitor mechanism which requires the tag has ability to detect the collision.

It is worth noting that stack storage mechanism is introduced as well. When the Reader responds to the tags, it will put the data into a stack first, which obeys the first in

first out principle (FIFO). This is the basic method in order to implement the Reader channel control function.

One regulation from ISO/IEC 18000 standard[10] says that the tag should respond the Reader in 32μs once receives the request. Due to the complexity of many encryption algorithms, the length of the key and the compute ability of the tag, one simple conclusion is shown in Table 3-2.[16][37][39][40] Thus, there is a need to modify the protocol in order to avoid the invalid operation from tags.

Table 3-2 Timing of on-tag cryptographic algorithms

Approach	Timing (clk cycles)	Time (with 100kHz clock frequency)
Hash	85-1274	850μs-12.74ms
AES	1016	10.16ms
Public key	401-1088	4.01ms-10.88ms

One approach to deal with the timing issue is to add one control switch. When tag receives the Reader's command, the switch turns on which makes the tag becomes active to send data. After the communication finishes, the switch turns off and tag stops sending data and enters the Lock status of Hash-Lock. Tag will turn to work status again only if the Reader sends request later.

The flow diagram of the proposed protocol is described in Figure 3-9.

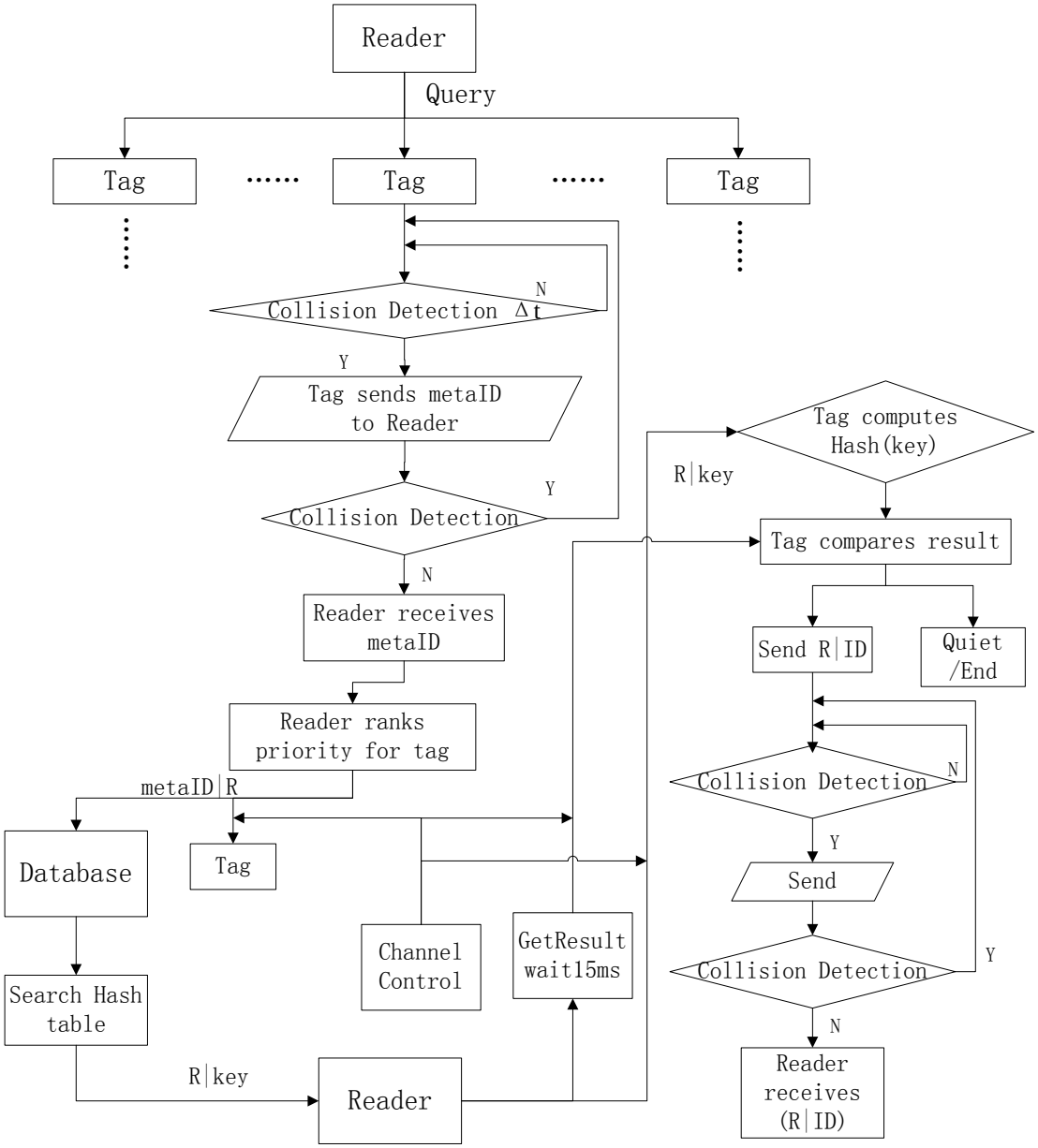


Figure 3-9 Flow diagram of proposed protocol

Note that, in the figure there are two “Reader”s which are exactly the same one. And all the “Collision Detection” functions refer to the same mechanism in tags, and independent to each other.

The procedure of the protocol is as followed:

(1) Reader broadcasts the *Query* to tags.

(2) Tag turns to be active when receives the *Query*. Then tag gets its *metaID* ready and starts collision detect function. If the channel is available during a random time period Δt, tag sends *metaID* to Reader. Meanwhile, the collision detection protocol

works through transmission. Once confliction is detected, rollback and repeat this step.

(3) Reader puts the received *metaID* into Register, and assigns the priority value *R* to tag. *R* value adds one.

(4) Reader sends (*metaID*|*R*) to tag and back-end database.

(5) Database looks up Hash table to find the matched *key* for the *metaID* and then replies (*R*|*key*) to Reader.

(6) Reader transfers (*R*|*key*) to tag, records the time t.

(7) Tag hashes the *Key*, when receives (*R*|*key*) from Reader and then compares the hash value with the old *metaID*, and buffers the result.

(8) Reader sends (*R*|*GetRseult*) to tag 15ms after t.

(9) Tag checks the buffer when receives (*R*|*GetRseult*), if the authentication is verified, then responses to the Reader with ID or Content using collision detection. Otherwise, tag keeps Quiet.

There is a channel control method employed in the Reader sending system. Due to the one tag-one Reader RFID model without channel compete, there is no requirement to use this method. As to multi-tag application, using this channel control can manage the sending data sequence, and therefore guarantee the system accuracy and integrality.

3.4.2 Analysis of proposed protocol

First of all, all the assumptions of the parameter refer to the ISO/IEC 18000 standard, which ensure the reliability of the protocol.

Considering the principle mentioned above that tag should respond in 32μs, the step (8) is designed which supports 15ms waiting before sends the <GetResult> requirement.

Secondly, security and privacy. The anti-collision algorithm combines the Hash-Lock security technology which is using *metaID* as index instead of original ID. The length of Hash value, *metaID*, is fixed no matter the input length. Hash function has

strict mapping relationship which is good enough to protect the privacy of tag's information.

Thirdly, integrality. In terms of theory, all the tags can be identified by Reader within its working range. Since the use of detection function, all signals will be received by Reader eventually. If collision is detected, tag will repeat to send the data when the channel is available. And the use of random detection time Δt decreases the collision rate and increases the efficiency.

The efficiency of the system is improved significantly by using the priority identification *R* as well. *R* changes the communication model in one Reader and multi-tag system. And it works efficiently for the frequent communication application.

From the other side, this protocol has its disadvantages. The expense of the increased efficiency and the integrality is high power consumption and the cost of tags. And other security method should be used when there is more security request. Although this proposal has some limitations, from the economic and efficiency aspects, it is reasonable and suitable.

3.5 Expanded protocol

As mentioned before, there are some challenges of collision detection. And this section presents another approach which based on the incomplete collision detection method which discussed in section 3.2.2.

The procedure of the protocol is as followed:

(1) Reader broadcasts the *Query* to tags.

(2) Tag becomes active when receives the *Query*. Then tag gets its *metaID* ready and starts collision detect function. If the channel is available during a random time period Δt, tag sends *metaID* to Reader. And tag will repeat it every period $(2\Delta d + \Delta\varepsilon)$ until it receives the ACK, i.e. (*metaID*|*R*), from Reader.

(3) Once Reader receives *metaID*, it will check about the priority register to make sure whether the *metaID* is already there. If the *metaID* exists, Reader will not do any

operations. Otherwise, Reader puts the received *metaID* into register, and assigns the priority value *R* to tag. *R* value adds one.

(4) Reader sends (*metaID*|*R*) to tag and back-end database.

(5) Database looks up Hash table to find the matched *key* for this the *metaID*. Then replies (*R*|*key*) to Reader.

(6) Reader transfers (*R*|*key*) to tag and recordsthe time t.

(7) Tag hashes the *Key* when it receives (*R*|*key*) from Reader. Then tag compares the hash value with the old *metaID*, and buffers the result.

(8) Reader sends (*R*|*GetRseult*) to tag 15ms after t.

(9) Tag checks the buffer when it receives (*R*|*GetRseult*). If the authentication is verified, tag responses to the Reader with ID or Content similar to step (2). Otherwise, tag keeps Quiet.

3.6 Summary

This chapter proposes a multi-tag content access control protocol based on the theory of chapter 2. Firstly, two main modules are designed and discussed separately. One is proposed multi-tag anti-collision algorithm; the other is the protocol of security and privacy protection for multiple tags using Hash-Lock. Then, new protocol is presented with details and performance analysis. Finally, one expanded protocol is designed.

Chapter 4 Modeling and Simulation

4.1 Modeling with GPenSIM

The modeling part is using GPenSIM[41], which is a tool based on Matlab. There are many Petri Net modeling tools, and most of them are flexible to handle with complex system. GPenSIM is non-graphic manner using a code programming with Matlab platform which is a much simpler but still reliable choice. It has three files to implement and analyze the simulation which are Petri net Definition Files (PDF) which contains the static information, Transition Definition Files (TDF) and Main Simulation Files (MSF) which contains the dynamic details.

The methodology of this simulation has two main steps: first, define the PDFs are considered as static definition phase as well. The elements include Places, Transitions and Arcs. After establishes the Petri net graph, TDFs are used to describe the transfer or fire condition. The most important step is definite the dynamics variables which include the initial tokens in places and the firing times during transitions.

4.2 Simple Hash-Lock protocol model

The protocol of Hash-Lock in single tag and Reader system is illustrated in Figure 4-1. Reader sends *Query* to tag is the step that ‘pSR’ send token to ‘tRT’. When ‘pRRQ’ receives token, it means the tag has got the *Query* from Reader. Then the *metaID* will be sent from ‘pSRS’ to Reader by firing ‘tTRID’.

Reader receives the *metaID* at place ‘pRRID’, and then transfers it to back-end database through ‘tRDB’. The transition ‘tDBR’ is fired to send *key* to Reader after the database searches the Hash table. Reader will transfer the *key* to tag, when‘tRK’ fires. Meanwhile the Reader has to wait for 15ms before sends <GetResult> request which is controlled by ‘tTS’. When ‘tHK’ fires, it means the tag is hashing the *key* and buffering

the result. Then ‘tCon’ is fired if each ‘pB2’ and ‘pB3’ has one token, which means tag will return the result to Reader after tag stores the result and gets the <GetResult> request.

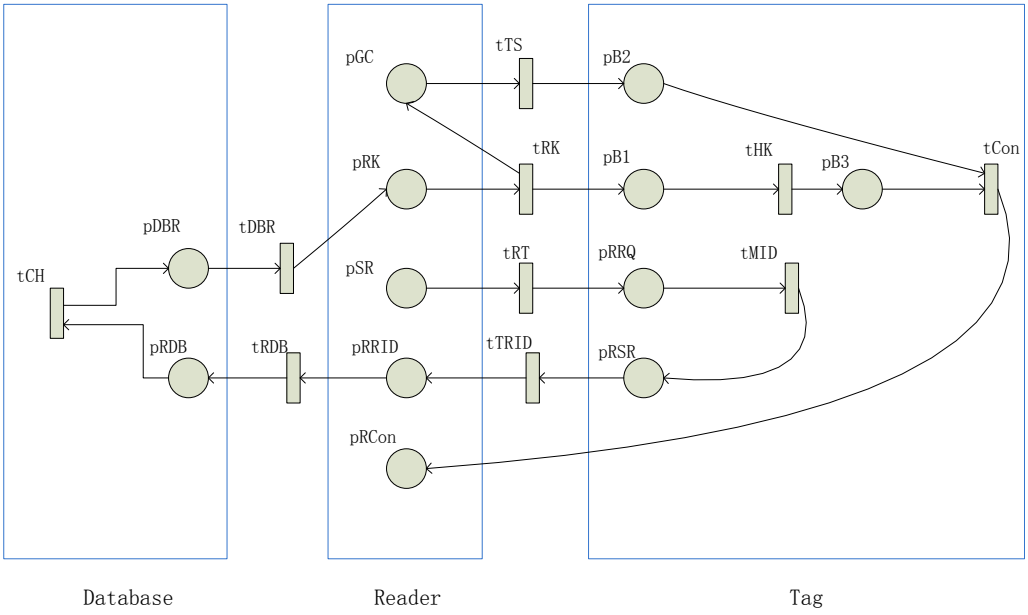


Figure 4-1 Model of Hash-Lock authentication protocol

4.3 Multi-tag content access control protocol model

According to the multi-tag content access control protocol using Hash-Lock presented before, the model of this protocol is shown in Figure 4-2.

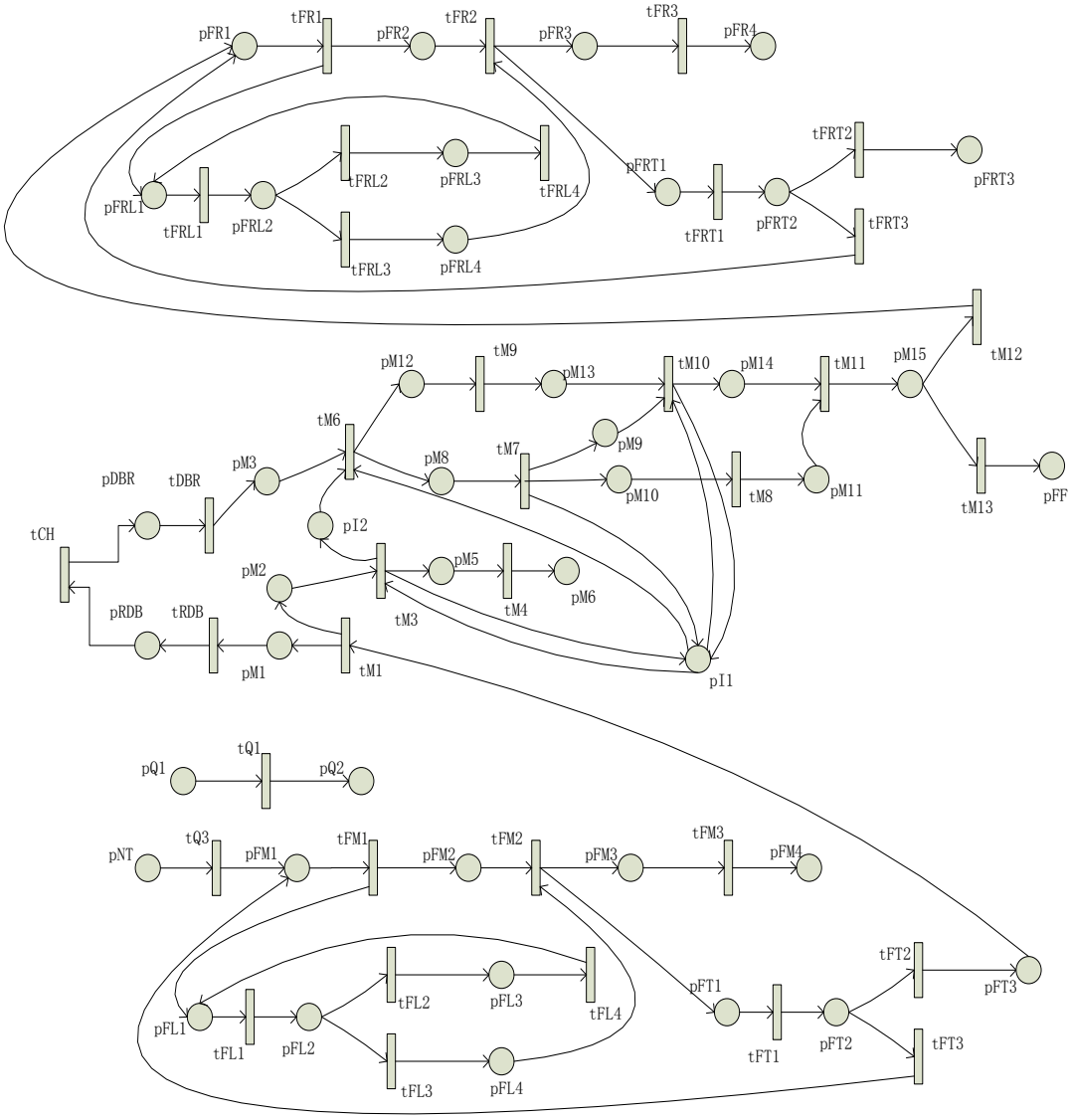


Figure 4-2 Proposed protocol model

4.4 Modules analysis

4.4.1 Tag active control module

Figure 4-3 Tag active control module

Figure 4-3 shows the process that Reader sends *Query* first, and then the tag is active when gets the *Query*. Reader at place ‘pQ1’ sends *Query* to tags by firing the

'tQ1'. If there is a tag at place 'pQ2', the broadcasted data from Reader will be received by this tag. The number of tokens in 'pNT' means the number of tags who received the *Query* and ready for the next operation. The requirement to fire 'tQ3' is the number of tokens in 'pQ2' is one and at least one in 'pNT'. This is shown in Figure 4-4.

```
% tQ3_def
function[fire,new_color,override,selected_tokens,global_info]...
    =tQ3_def(PN,new_color,override,selected_tokens,global_info)
    b4=get_place(PN,'pQ2');
    b5=get_place(PN,'pNT');
fire=(b4.tokens==1)&(b5.tokens>0);
```

Figure 4-4 TDF of fire 'tQ3'

4.4.2 Anti-collision module

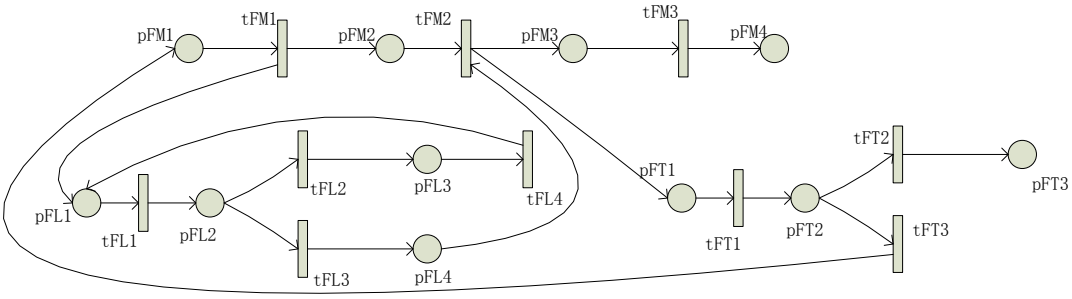


Figure 4-5 Anti-collision module

Figure 4-5 shows the module of anti-collision algorithm using monitor mechanism. It refers to the tag's internal design. 'pFM1' represents the tag is ready to send data. The monitor function will turn on by firing 'tFM1' to transfer the token to 'pFL1'. 'tFL1' represents tag listens to the channel for Δt(ms), if the link is busy, tag will fire 'tFL2', otherwise tag will send signal by the means of fire 'tFL3'. And the collision detection during the transmission is active, i.e. 'tFM2' is fired. If there is no conflict, 'tFT2' is fired which means the Reader can receive the signal correctly. And if collision is detected, then 'tFT3' is fired which means the signal will be sent again when the channel is available.

4.4.3 Channel control module of Reader

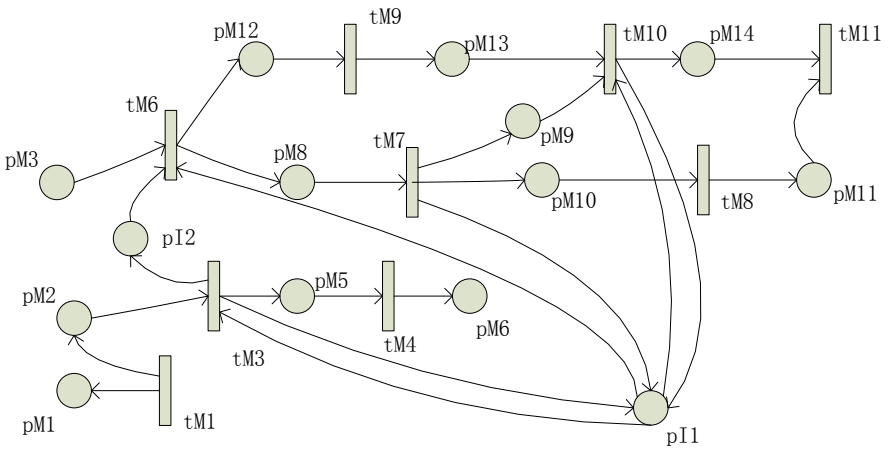


Figure 4-6 Authentication and channel control module

This channel control module in Figure 4-6 refers to the design of Reader. When Reader receives the *metaID* from tag, ‘tM1’ is fired which means Reader will store the *metaID* and assign the priority *R* value. (*metaID*|*R*) will be sent to database through ‘pM1’ which is connected with the database. Meanwhile Reader will reply (*metaID*|*R*) to tag by firing ‘tM3’. Reader receives (*metaID*|*key*|*R*) from database which is represented by ‘pM3’ receives a token. ‘tM6’ is fired if and only if place ‘pI1’, ‘pI2’ and ‘pM3’ has a token each. There are two directions after ‘tM6’ fired. One direction is to fire transition ‘tM7’ sending (*R*|*key*) to tag and ‘pM9’ receives one token which works as a control token. After receives (*R*|*key*), tag will hash the *key*, and then stores the result in buffer with one token assigned to ‘pM11’. The other direction is the waiting and getting result implementation. After ‘pM12’ gets the token, ‘tM9’ is fired and 15ms later ‘pM13’ receives one token. ‘tM10’ can be fired if and only if ‘pM13’, ‘pM9’ and ‘pI1’ has one token each, which means Reader can send (*R*|*GetResult*) to tag when Reader has sent (*R*|*key*) more than 15ms and the channel is available at that moment. After that, ‘pM14’ gets one token. When tag has the result of authentication, ‘pM11’ has a token as the same as ‘pM14’, ‘tM11’ is fired to reply the result from tag to Reader.

Notice that this module considers about the character of Reader’s communication channel which is serial. Thus, ‘pI1’ is the symbol of Reader’s channel. When Reader is

sending a signal, ‘pI1’ has no token, and ‘pI1’ has one token if the channel is available. And the sequence of sending the data is controlled by ‘pI2’ and ‘pM9’.

As mentioned in the former chapters, this proposed protocol uses stack technology which obeys FIFO principle. It increases the rate of identification and provides a well-organized communication mode between Reader and tags. For example, if there are *i* tags that marked {t1, t2, ……, ti} with *metaID* {m1, m2, ……, mi} in the system. These tags share the priority values {R1, R2, ……, Ri} respectively. Assume that ‘pI1’ always fires ‘tM3’ when the three transitions ‘tM3’, ‘tM6’and ‘tM10’ are ready to fire, which means ‘tM3’ is prior to fire compared to ‘tM6’ and ‘tM10’. The result is that there are huge numbers of tokens stored in place ‘pI2’ and ‘pM3’. But in terms of using priority identification mechanism, all the tokens in the stack are sorted and matched, i.e. the top token has the minimum *Ri*. Only after the tag received its own priority *Ri*, the communication can use R as the index, and the place ‘pI2’ controls this sequence. ‘pM9’ can make sure that the Reader has kept waiting for at least 15ms after sent the data (*R*|*key*) to tag, and allows Reader to send the get result command.

It is clear to see that the stack technology in the Reader control module makes the communication orderly and efficient, but the cost is Reader needs have enough memory storage to handle some big system with a large number of tags.

4.4.4 Respond result module

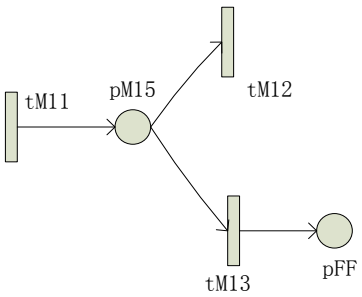


Figure 4-7 Respond the authentication result module

Colored token is used in this module as shown in Figure 4-7. If the result is failed, then tag will mark the token as color type ‘FAIL’, otherwise will mark the token as color

type ‘PAS’. The color of the token in place ‘pM15’ decides which one of ‘tM12’ and ‘tM13’ can be fired. If the color is ‘PAS’, ‘tM12’ will be fired and tag will turn to monitor status and be ready to send signal. Figure 4-8 is the TDF of fire ‘tM12’. Otherwise, the transition ‘tM13’ is fired and the TDF of ‘tM13’ is shown in Figure 4-9.

```
%tM12_def
function [fire, new_color, override, selected_tokens, global_info]...
    =tM12_def(PN, new_color, override, selected_tokens, global_info)
[selected_tokens, nr_token_av]=select_token_color(PN,'pM15',1,'PAS');
fire=nr_token_av;
```

Figure 4-8 TDF of ‘tM12’

```
%tM13_def
function [fire, new_color, override, selected_tokens, global_info]...
    =tM13_def(PN, new_color, override, selected_tokens, global_info)
[selected_tokens, nr_token_av]=select_token_color(PN,'pM15',1,'FAIL');
fire=nr_token_av;
```

Figure 4-9 TDF of ‘tM13’

4.4.5 Communication module between Reader and back-end database

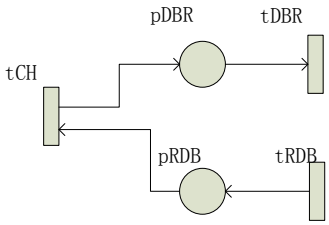


Figure 4-10 Encapsulated database module

Since the communication channel between Reader and back-end database is a kind of secure link, thesis does not deep discuss about it. Thus, this part is encapsulated as a whole which is shown in Figure 4-10. It is the same module in simple Hash-Lock protocol and the proposed one.

4.5 Coverability tree analysis

Coverability tree[41] is an important tool to analyze the reliability, stability and reachability of the system model. All the reachable states can be determined from a

given initial status.

In order to running the coverability tree analysis, one complete discrete simulation system should be built by defining the PDFs, TDFs and MSF. According to the model's structure the initial status assume that 'pQ1', 'pNT' and 'pI1' has one token for each, and the other places has no token. The initial status is shown in Figure 4-11.

state:1	ROOT node						
pNT	pQ1	pQ2	pI2	pI1	pFF	pM1	pM2
pM3	pM5	pM6	pM8	pM9	pM10	pM11	
pM12	pM13	pM14	pM15	pFM1	pFM2	pFM3	
pFM4	pFR1	pFR2	pFR3	pFR4	pFL1	pFL2	pFL3
pFL4	pFT1	pFT2	pFT3	pFRL1	pFRL2	pFRL3	pFRL4
pRDB	pDBR	pFRT1	pFRT2	pFRT3			
1	1	0	0	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0					

Figure 4-11 Initial status of coverability tree

After running the coverability tree test, there are 5806 determined states from the initial status. Figure 4-12 to Figure 4-16 are five states from coverability tree.

state:5799	Firing event: tFM3						
pNT	pQ1	pQ2	pI2	pI1	pFF	pM1	pM2
pM3	pM5	pM6	pM8	pM9	pM10	pM11	
pM12	pM13	pM14	pM15	pFM1	pFM2	pFM3	
pFM4	pFR1	pFR2	pFR3	pFR4	pFL1	pFL2	pFL3
pFL4	pFT1	pFT2	pFT3	pFRL1	pFRL2	pFRL3	pFRL4
pRDB	pDBR	pFRT1	pFRT2	pFRT3			
0	0	1	0	1	0	0	0
0	0	1	0	0	0	0	0
0	0	0	0	0	Inf	11	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1					
Node type: ' '	Parent state: 5775						

Figure 4-12 Coverability tree (state=5799)

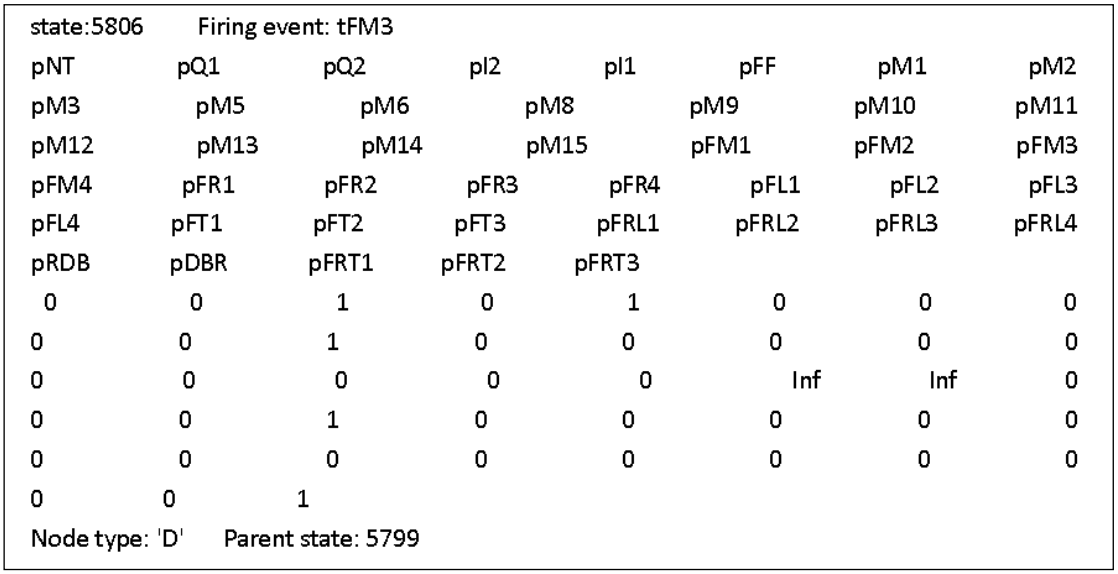


Figure 4-13 Coverability tree (state=5806)

In Figure 4-13, the “Parent state” means that state ‘5799’ is its father state which is shown in Figure 4-12. State ‘5806’ is reachable after fires ‘tFM3’. The ‘Inf’ indicates that the number of token is increasing. Type ‘D’ means this is a duplicate state that can be reached by other accesses and may have child state, while ‘T’ means a terminal status that has no child state. If the type of state is none that means it is a transitional state which has child state.

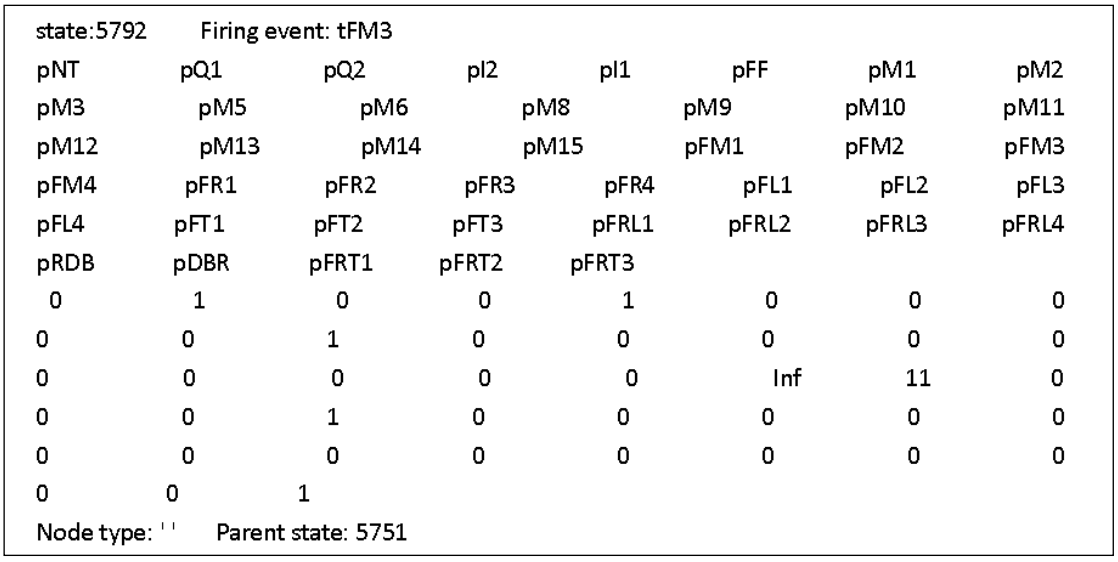


Figure 4-14 Coverability tree (state=5792)

state:5804 Firing event: tQ1

pNT	pQ1	pQ2	pI2	pI1	pFF	pM1	pM2
pM3	pM5	pM6	pM8	pM9	pM10	pM11	
pM12	pM13	pM14	pM15	pFM1	pFM2	pFM3	
pFM4	pFR1	pFR2	pFR3	pFR4	pFL1	pFL2	pFL3
pFL4	pFT1	pFT2	pFT3	pFRL1	pFRL2	pFRL3	pFRL4
pRDB	pDBR	pFRT1	pFRT2	pFRT3			
0	0	1	0	1	0	0	0
0	0	1	0	0	0	0	0
0	0	0	0	0	Inf	11	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1					

Node type: 'D' Parent state: 5792

Figure 4-15 Coverability tree (state=5804)

state:5805 Firing event: tFM3

pNT	pQ1	pQ2	pI2	pI1	pFF	pM1	pM2
pM3	pM5	pM6	pM8	pM9	pM10	pM11	
pM12	pM13	pM14	pM15	pFM1	pFM2	pFM3	
pFM4	pFR1	pFR2	pFR3	pFR4	pFL1	pFL2	pFL3
pFL4	pFT1	pFT2	pFT3	pFRL1	pFRL2	pFRL3	pFRL4
pRDB	pDBR	pFRT1	pFRT2	pFRT3			
0	1	0	0	1	0	0	0
0	0	1	0	0	0	0	0
0	0	0	0	0	Inf	Inf	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1					

Node type: 'D' Parent state: 5792

Figure 4-16 Coverability tree (state=5805)

From Figure 4-14 to Figure 4-16, we can conclude the partially coverability tree as Figure 4-17 shown. State ‘5792’ can be reached when state ‘5751’ fires ‘tFM3’. If state ‘5792’ fires ‘tQ1’, state ‘5804’ is reachable, otherwise if state ‘5792’ fires ‘tFM3’, state ‘5805’ is reached. The tokens moving from one state to another is clearly shown from these figures.

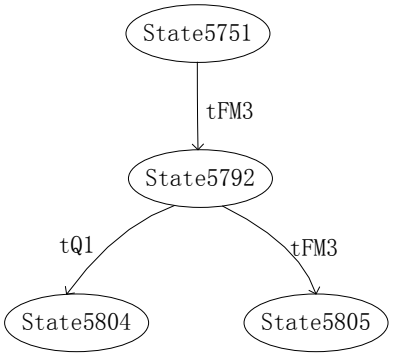


Figure 4-17 Partially coverability tree

4.6 Parameters

To have a fair comparison, the parameters used here are referred from the ISO/IEC 18000 standard. We assume the date rate between tag and Reader is 40kbps. In the proposed protocol, SHA-1 is employed as the Hash-Lock algorithm with 160bits output.

Main parameters are set as follow:

(1) The length of *Query* is 64bits.

(2) In single tag Hash-Lock algorithm, the tag replies *metaID* with a random period $\Delta\varepsilon \in [150,320]\mu s$.

(3) In the proposed anti-collision algorithm, the detection period is $\Delta t \in [1,10]\mu s$.

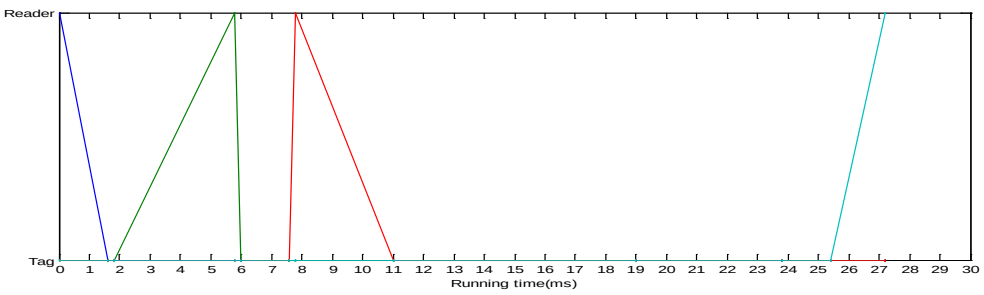
(4) The time of looking up Hash table in database is $\Delta\tau \in [1,4]ms$.

(5) According to the Table 3-2, $\Delta\delta \in [0.85,12.74]ms$. In [39] and [40], the encrypt time of tag is about 3.5ms when using 8bit SHA-1 algorithm with 100kHz clock frequency.

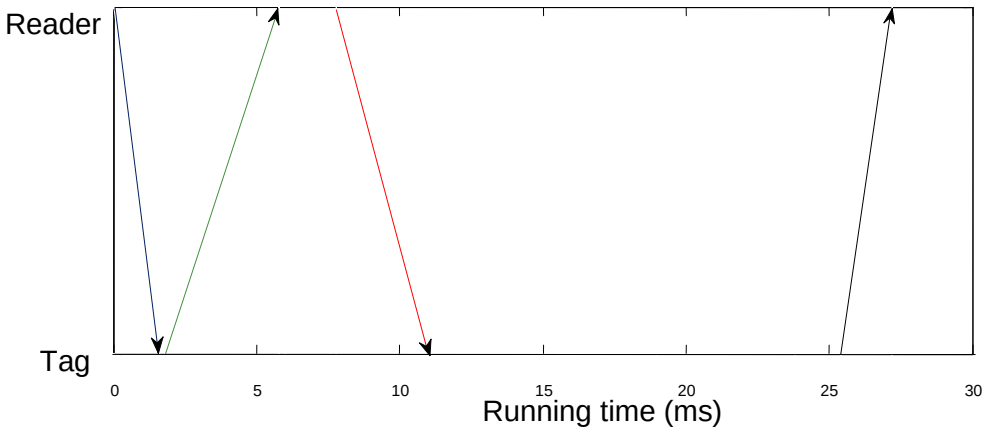
4.7 Results and analysis

4.7.1 Simple Hash-Lock protocol with single tag

When the length of key is 128bits, the result of simulation is shown in Figure 4-13.



(a)



(b)

Figure 4-18 Single tag Hash-Lock protocol (the length of key is 128 bits)

In order to simplify the analysis, we modify the simulation curves of Figure (4-18.a) to Figure (4-18.b). At time t=0s, Reader sends *Query* to tag which cost time t1:

$$t1 = {}^{64\text{bits}}/_{40\text{kbps}} = 1.6\text{ms} \tag{4.1}$$

After the tag receives the *Query*, it replies *metaID* during $\Delta\varepsilon \in [150,\ 320]\mu s$. Reader receives after time t2:

$$t2 = {}^{160\text{bits}}/_{40\text{kbps}} = 4\text{ms} \tag{4.2}$$

Reader transfers the *metaID* to database, and receives the *key* as reply from database at around 7.6ms. Reader will send the *key* to the tag which take time t3, and wait for 15ms to get result.

$$t3 = {}^{128\text{bits}}/_{40\text{kbps}} = 3.2\text{ms} \tag{4.3}$$

At about 11ms, the tag receives the *key*, and starts to hash the *key*. At time 25.4ms, tag receives the get result require from Reader, and then replies the result. Finally, this

communication stops at 27.2ms.

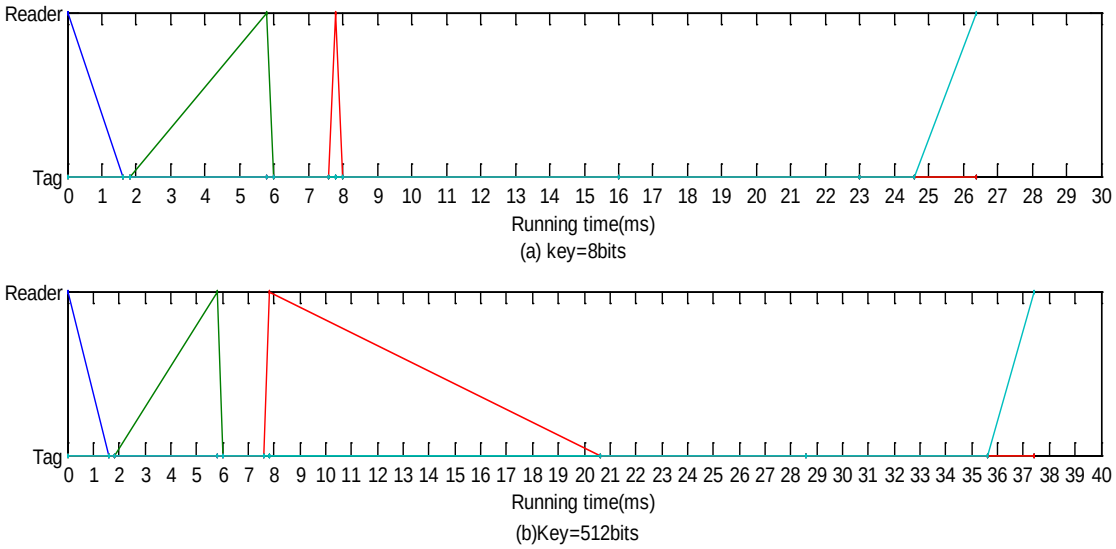


Figure 4-19 Single tag Hash-Lock protocol

(with the length of key: 8bits and 512bits)

From the figures in Figure 4-19, the authentication procedure that using 8bits length key costs 26.4ms while using 512bits length key costs 37.4ms.

4.7.2 System efficiency of multi-tag content access control protocol

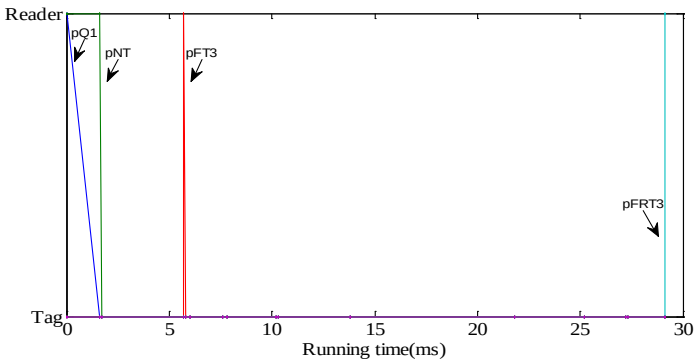


Figure 4-20 Content access control protocol with one tag, complete time: 29.137ms

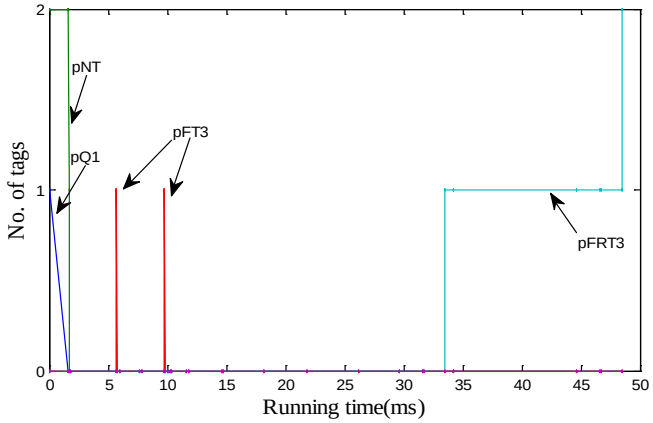


Figure 4-21 Content access control protocol with two tags, complete time: 48.537ms

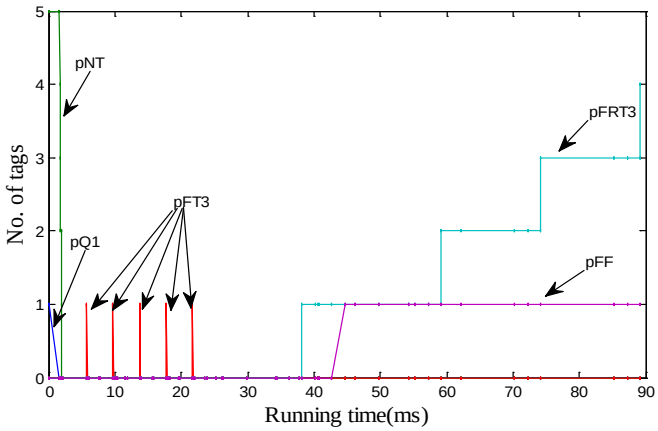


Figure 4-22 Content access control protocol with five tags, complete time: 89.161ms

In these Figures, ‘pFT3’ stands for the numbers of correct signals received from tags. ‘pFRT3’ represents the number of tags which Reader can access, while ‘pFF’ refers to the number of tags which Reader cannot access.

As mentioned in Figure 4-20, one tag authentication using our proposed protocol costs almost the same time as the single tag Hash-Lock protocol. If there are two tags, our proposed method takes 48.5ms which is 6ms less than the time that original Hash-Lock running twice. The new protocol saves about 11% time. If the number of tags increases to five, our new approach costs only about 90ms which is 45ms less than the old protocol costs.

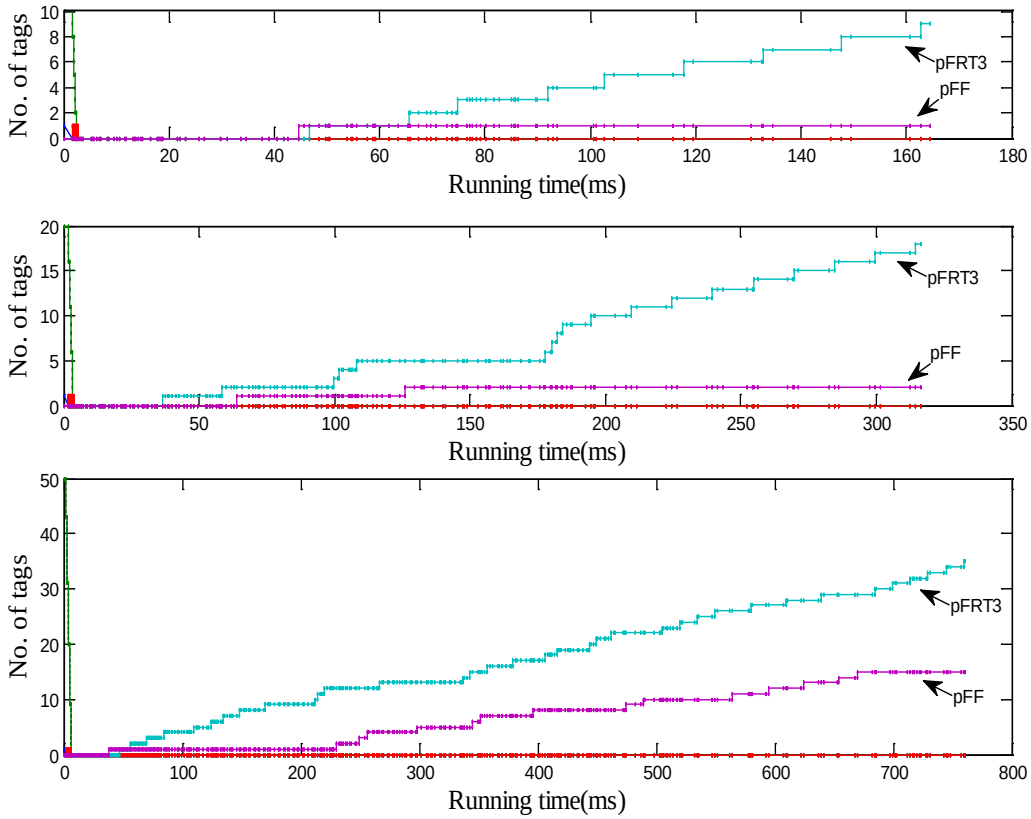


Figure 4-23 Simulation of proposed protocol with 10,20,50 tags

Figure 4-23 shows that authentication time for 10, 20 and 50 tags is about 165ms, 318ms and 770ms respectively. It is obvious that the efficiency is improved about 39%, 41% and 43% respectively.

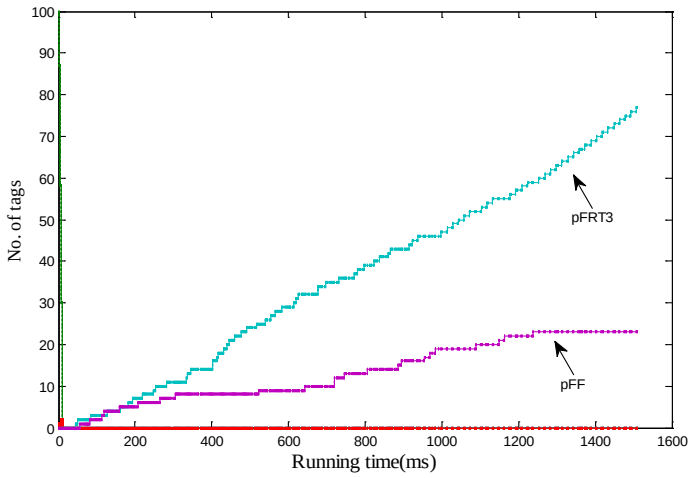


Figure 4-24 Simulation of proposed protocol with 100 tags

When there are 100 tags in the RFID system, time needed to finish the identification is approximate 1528ms, as shown in Figure 4-24. Therefore the system

efficiency is improved about 43% compared with the old approach.

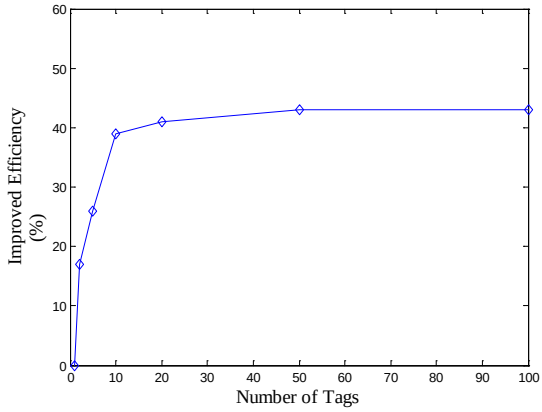


Figure 4-25 Improved efficiency by proposed protocol

As shown in Figure 4-25, in a small size system (e.g. less than 20 tags) when the number of tags is increasing, the efficiency is improved greatly. When there are about 20 to 50 tags, the improved efficiency of the system will grow steady and it will hold flat at the maximum value about 43% if the number of tags is more than 50.

4.7.3 Channel utilization rate of multi-tag content access control protocol

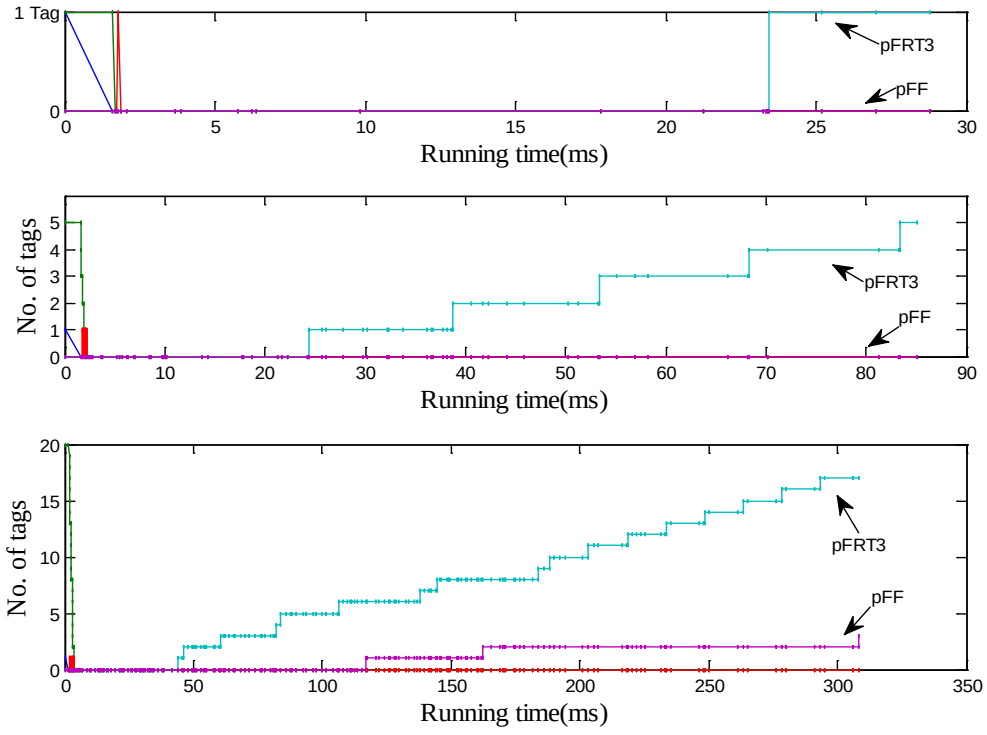


Figure 4-26 Simulation result of 40% channel utilization

When the channel utilization rate is 40%, the simulation result of one tag, 5 tags and 20 tags is illustrated in Figure 4-26. Compared to original protocol, the system efficiency is improved 0%, 36% and 43% respectively, with the implementation period about 28.5ms, 86ms and 310ms. The result is almost the same as the figures shown in Figure 4-25 whose channel utilization is 10%.

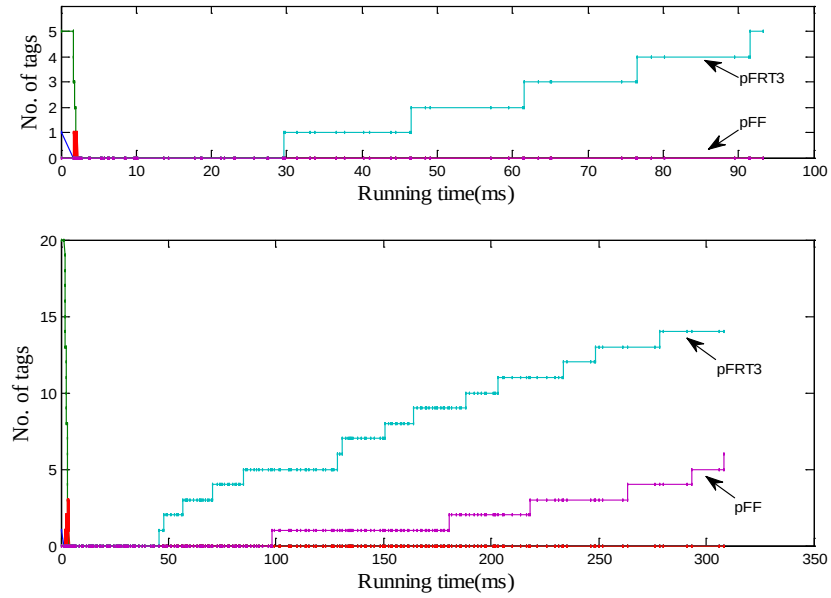


Figure 4-27 Simulation result of 60% channel utilization

When the channel utilization rate is 60%, the cost to implement the authentication with 5 or 20 tags is about 93ms or 310ms respectively illustrated in the Figure 4-27. Compared to the original protocol, the efficiency is improved as 31% or 43%.

According to the results, with different channel utilizations, such as 10%, 40% and 60%, the improved efficiency curve is almost the same. Thus, it can be concluded that the proposed protocol can improve the efficiency greatly and can hold high channel utilization.

Furthermore, the proposed protocol combines both anti-collision and security authentication mechanisms, while the original protocol just consider about the authentication part. If add the anti-collision module into the original, the cost of time will be more.

4.8 Summary

This chapter focuses on the modeling and simulation of the proposed protocol. First establish the protocol model, then analysis the modules. The simulation results demonstrate that the new system using proposed protocol has better efficiency and higher channel utilization rate.

Chapter 5 Conclusion and Further Developments

5.1 Conclusion

Compared with other traditional identification technologies, RFID has huge application potential according to its characters. But under different application contexts, there is a need to develop one protocol which can combine security protocol with anti-collision algorithm.

This thesis proposed a multi-tag content access protocol using Hash-Lock mechanism and ALOHA based anti-collision algorithm. Several approaches are introduced into this protocol. For RFID Reader, one priority register is designed to rank the order of multiple tags. Channel control mechanism is applied to control the serial transmission of Reader. For RFID tag, a monitor mechanism is used to increase the system efficiency. The advantages and disadvantages of the proposed protocol are discussed. Simulation results indicate that the new protocol performs well when applied to RFID tag content access control together with the anti-collision, in this way multiple tags can be identified in batch.

5.2 Further developments

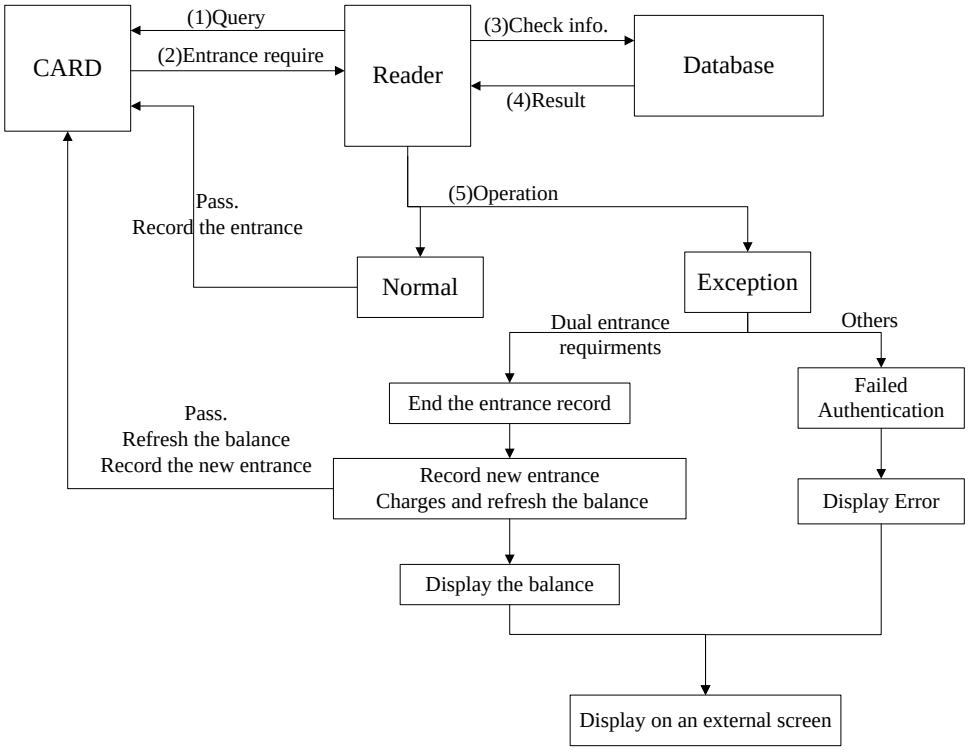
The research on RFID security is developing rapidly. According to the idea of this thesis, Tree-based anti-collision combine with ECC encryption algorithm might be discussed and developed in future.

In terms of this proposed protocol, some applications can be designed and implemented in daily life. One open application research based on this protocol is the subway check out system.

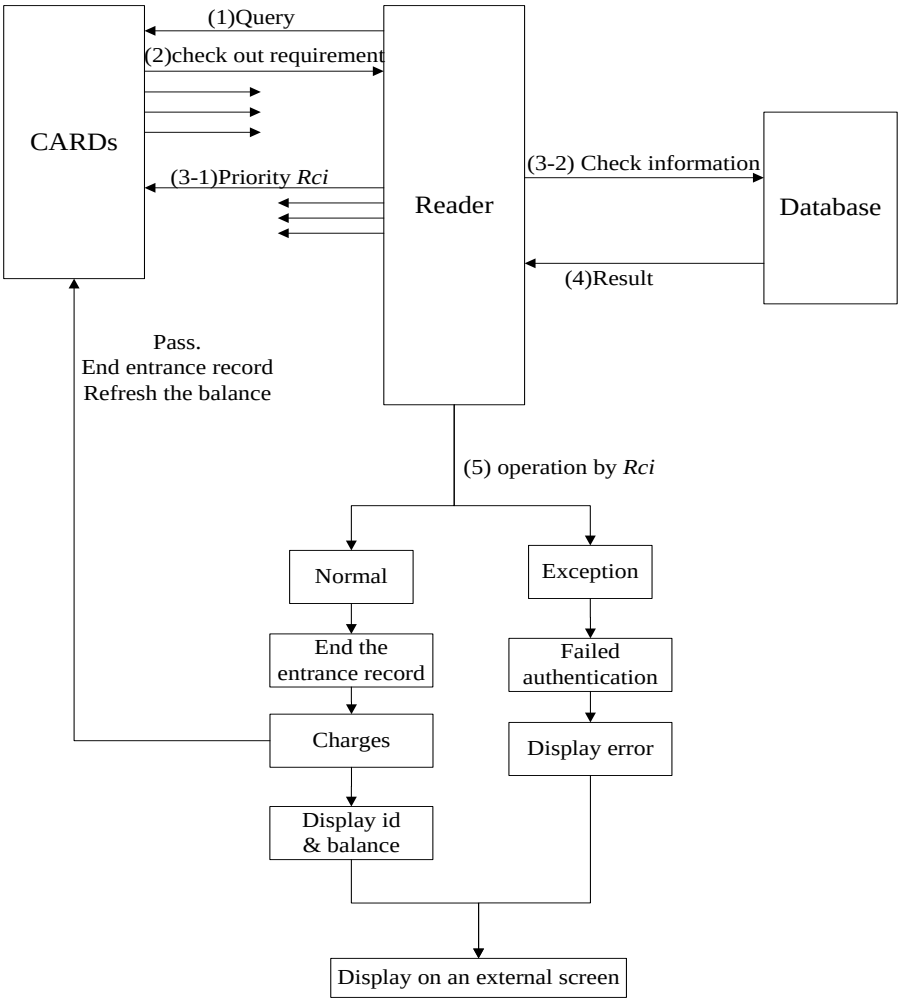
In many cities, the payment of subway tickets is according to the miles or the number of stations. And the simple model is like this: reader will check the travel card

and make a record when people enter the subway and check out and charges the money when people exit. Most of the exit ports require the customers identified one by one which cost lots of time when there is large number of customers.

As mentioned before, the proposed protocol supports identify of multi-tag in batch. Thus, one scenario might be designed as follow. When customers enter the subway, reader will check the smart travel card check and record the entrance information. At the exit port, a check-out table should be set which is a kind of Reader that can deal with multi-tag content access control protocol. The Reader identifies the cards, ends the entrance record, charges and displays the verified information. If the card asks for a dual entrance without an end record of last entrance, Reader will charge for the last trip with the maximum fee and then make a new entrance record. All the information will be displayed on a connected screen. Of course, the personal information is protected using another distinguished id to the privacy ID. The proposed application structures can be specified in the Figure 5-1.



(a) Identification of subway ticket system (Entrance)



(b) Identification of subway ticket system (Check-out)

Figure 5-1 Subway ticket system using multi-tag content access control

The proposed application in subway system is feasible though more factors need to be considered in real system. Just like the RFID technology, the cost and efficiency should trade off at some certain balance point.

It is far away from enough to develop a perfect protocol to deal with complex applications during the short period. For instance, the power consumption, the system's bit error rate and the implementation on hardware are the following research directions.

There might be some mistakes or thoughtless of the thesis, it should be grateful to be modified in future.

。

References

[1] http://en.wikipedia.org/wiki/Automatic_identification_and_data_capture.

[2] R.Want. An Introduction to RFID Technology. IEEE Pervasive Computing, vol. 5, no. 1, pp. 25-33, Jan. 2006.

[3] W.Su, N.V.Alchazidis, T.T. Ha. Multiple RFID Tags Access Algorithm. IEEE transactions on mobile computing, Vol.9, No.2, Feb.2010.

[4] TF.La Porta, G.Maselli, C.Petrioli. Anti-collision Protocols for Single-Reader RFID Systems: Temporal Analysis and Optimization. IEEE transactions on mobile computing, 2010.

[5] S.Sarma, S.Weis, D.Engels. RFID Systems and Security and Privacy Implications. Cryptographic Hardware and Embedded Systems, pp.1-19, Jan. 2003.

[6] H.Liu. The Approaches in Solving Passive RFID Tag Collision Problems. Radio Frequency Identification Fundamentals, Cristina Turcu (Ed.), 2010.

[7] C.S.Kim, K.L.Park, H.C.Kim, S.D.Kim. An Efficient Stochastic Anti-collision Algorithm using Bit-Slot Mechanism. Proc. of Int. Conf. on Parallel and Distribute Processing Techniques and Applications, 2004.

[8] C.H.Quan, W.K.Hong, H.C.Kim. Performance Analysis of Tag Anti-collision Algorithms for RFID System. Emerging Directions in Embedded and Ubiquitous Computing, Volume 4097, 2006.

[9] EPC standard. http://www.epcglobalinc.org/standards/.

[10] ISO/IEC 18000 standards. http://www.iso.org.

[11] S.A.Weis. Security and Privacy in Radio-frequency Identification Devices. Master's thesis, 2003

[12] A.Juels. RFID Security and Privacy: A Research Survey. Selected Areas in Communications, IEEE Journal, 2006.

[13] J.Beak, J.Newmarch, R.Safavi-Naini, W.Susilo. A Survey of Identity-Based Cryptography. Proceeding of the 10th Annual Conference for Australian Unix and Open

System User Group (AUUG 2004), pp 95-102, 2004.

[14] A.Juels, S.Weis. Defining Strong Privacy for RFID. PerCom Workshops, White Plains, USA, pp. 342–347, 2007.

[15] D.Boneh, M.Franklin. Identity-based Encryption from the Weil Pairing Advances in Cryptology. Preceeding of Crypto'01, LNCS 2139, Springer-Verlag, 2001.

[16] S.A.Weis, S.E.Sarma, R.L.Rivest, D.W.Engels. Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems. Security in Pervasive Computing, Boppard, Germany, pp. 201–212, 2003.

[17] T.LaPorta, G.Maselli, C.Petrioli. Anti-collision Protocols for Single-Reader RFID Systems: Temporal Analysis and Optimization. IEEE Transactions on Mobile Computing, 2010.

[18] L.Liu, Z.Xie, J.Xi, S.Lai. An Improved Anti-collision Algorithm in RFID System. Mobile Technology, Applications and Systems, 2005 2nd International Conference on 15-17, 2005.

[19] S.Geng, W.Wu, L.Hou, W.Zhang. Anti-collision Algorithms for Multi-Tag RFID. Radio Frequency Identification Fundamentals, 2010.

[20] X.Yang, Z.Wang. An Improved Anti-collision Algorithm Using Hash Method in RFID System. Proceedings of ICCTA, 2009.

[21] E.O.Blass, R.Molva. New Directions in RFID Security. International Federation for Information Processing 2009, AICT 309, pp. 76–84, 2009.

[22] C.H.Quan, W.K.Hong, Y.D.Lee, H.C.Kim. A Study on the Tree Based Memoryless Anti Collision Algorithm for RFID Systems. The KIPS Transactions, Vol. 11, Korean Information and Processing Society, Korea, pp. 851-862, 2004.

[23] J.Myung, W.Lee, T.K.Shih. An Adaptive Memoryless Protocol for RFID Tag Collision Arbitration. IEEE Transactions on Multimedia, pp. 1096-1101, Oct. 2006.

[24] S.M.Wasikon, Z.Suradi. A Framework of Tag Anti-collision Algorithm for Fast Identification in RFID System. Computer and Communication Engineering, ICCCE 2008, International Conference on13-15, pp. 1019 – 1022, May. 2008.

[25] J.Eom, T.Lee. Framed-Slotted ALOHA with Estimation by Pilot Frame and Identification by Binary Selection for RFID Anti-collision. Proceedings of the International Symposium on Communications and Information Technologies (ISCIT 2007), pp.1027-1031, Oct. 17-19 2007.

[26] J.Eom, T.Lee. Accurate Tag Estimation for Dynamic Framed-Slotted ALOHA in RFID Systems. IEEE communications letters, Vol.14, No.1, Jan. 2010.

[27] G.Bagnato, G.Maselli, C.Petrioli, C.Vicari. Performance Analysis of Anti-collision Protocols for RFID Systems. IEEE 69th Vehicular Technology Conference. VTC Spring, pp.1–5, Apr. 2009.

[28] W.T.Chen. An efficient Anti-Collision Method for Tag Identification in a RFID System. IEICE Transactions on Communications, 2006.

[29] M.A.Bonuccelli, F.Lonetti, F.Martelli. Tree Slotted Aloha: A New Protocol for Tag Identification in RFID Networks. World of Wireless, Mobile and Multimedia Networks, International Symposium on 26-29 Jun. 2006.

[30] A.Juels, R.Rivest, M.Szydlo. The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. Proc. of ACM Conf. on Computer and Communication Security, pp.103-111, 2003.

[31] H.Yeo, Y.Kim, H.Lim, Y.Park, K.Seon.Ahn. ID Prediction Algorithm for Tag Collision Arbitration in RFID System. Proceedings of the 13th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2007), pp. 476-481, Aug. 2007.

[32] L.Liu, Z.Xie, J.Xi, S.Lai. An Improved Anti-collision Algorithm in RFID System. Mobile Technology, Applications and Systems, 2nd International Conference, pp. 67-71, Nov. 15-17 2005.

[33] M.Singh, D.Garg. Choosing Best Hashing Strategies and Hash Functions. IEEE International Advance Computing Conference, 2009.

[34] Y.Xiao, X.Shen, B.Sun, L.Cai. Security and Privacy in RFID and Applications in Telemedicine. IEEE Communications Magazine, Vol.44 Issue 4, pp.64-72, Apr. 2006.

[35] C.Floerkemeier, R.Schneider, M.Langheinrich, Scanning with a Purpose - Supporting the Fair Information Principles in RFID Protocols. In Ubiquitious Computing Systems, H. Murakami, et al., Editors, Springer-Verlag: Tokyo, Japan, 2004.

[36] L.Batina, J.Guajardo, T.Kerins, N.Mentens, P.Tuyls, I.Verbauwhede. Public-Key Cryptography for RFID-Tags. Proceeding of the 5th International conference on Pervasive Computing and Communications Workshops, 2007.

[37] Y.Liang, R.Chunming. Strengthen RFID Tags Security Using New Data Structure. International Journal of Control and Automation, 2009.

[38] EPCglobal Tag Data Standards. Version 1.3.

[39] MO'Neill. Low-cost SHA-1Hash Function Architecture for RFID Tags. Conference on RFID Security, Budapest, Hungary, 2008.

[40] M.Feldhofer, S.Dominikus, J.Wolkerstorfer. Strong Authentication for RFID Systems Using the AES Algorithm. Cryptographic Hardware and Embedded Systems, pp.85-140, 2004.

[41] R.Davidrajuh. Developing a New Petri Net Tool for Simulation of Discrete Event Systems. Second Asia International Conference on Modeling &Simulation, pp.861-866, 2008.