Performance Evaluation of RFID EPC Gen2 Anti-collision Algorithm in AWGN Environment

Cheng Jin and Sung Ho Cho
Graduate School of information and communication
Hanyang University
17 Haengdang-Dong, Sungdong-Gu, Seoul, 133-791, Korea
{skim & shcho}@casp.hanyang.ac.kr

Ki Yong Jeon
Semiconductor Material R&D Center
Samsung Techwin Co., LTD
San14, Nongseo-Dong, Giheung-Gu, Yongin-Si, Gyeonggi-Do, 466-712, Korea
kiyong.jeon@samsung.com

Abstract — There have been various kinds of anti-collision algorithms available in the RFID system for efficiently identifying multiple tags. The performances of the anti-collision algorithms, however, are often very difficult to evaluate and predict in practice. In this paper, we present the performance of the anti-collision algorithm used in the EPCglobal Gen2 standard in an AWGN environment. We have observed that, when identifying 100 tags, it requires approximately 1.3 times more time in the AWGN environment having Eb/No 5 dB, and 4.7 times in the AWGN environment having Eb/No 4 dB, comparing with the noise free environment. Also shown is that the anti-collision performance drastically degraded if the Eb/No falls below 3 dB.

Index Terms – RFID, anti-collision, evaluation, Gen2

I. INTRODUCTION

RFID standards for “Radio Frequency Identification” and is used to store and retrieve information [1]. This information is stored in and retrieved from RFID tags, which are small and can be incorporated into many products. RFID does not require contact or line-of-sight for operation, and is based on the use of small tags, which contain a unique identification number that could be read at distance of up to 8.4 meters [2]. Organizations in Retail, Defense, and other industries are increasingly employing transportation, healthcare RFID technology to bring new efficiencies to supply chains, track assets, ensure product quality and consumer safety and more. Dramatically increased opportunities for RFID technologies provide clear, differentiating benefits compared with conventional ID technologies [3].

In a RFID system, there usually exists the situation in which more than one tag exist in the reader range. In this case, if the reader broadcasts a requesting message, all tags will receive the message and send back their responses to the reader. So the responses may collide with each other in which case the reader could not receive the response message correctly from a certain tag. This situation is just referred to as the “Tag – collision” [4] and the technique to resolve this problem is referred to as the anti-collision protocol. The ability to identify numerous tags simultaneously by applying a specific anti-collision algorithm is an important aspect to evaluate the effectiveness of a RFID system.

Recently, many creative researches [4] focusing on the performance improvement of the anti-collision are reported. However, most researches conducted in this field put most concentration on the development of the anti-collision algorithm, while few concerns are given with the differences between the theoretical conditions and the practical environment influences. Since in practical environment, where the RFID system usually shares frequency band with other wireless devices [5], the environment conditions could usually be influenced by many factors such as the frequency sharing and physical environment conditions. So the performance of a RFID system could often be affected by some level of environment interference, which makes it difficult to evaluate the system performance exactly only by the anti-collision algorithm’s theoretic assumptions.

So in order to investigate the environment influences to the RFID system performance ,especially when identifying multiple tags, in this paper, the performance of the anti-collision algorithm defined in RFID EPCglobal Gen2 [8] Protocol was evaluated in an ideal noise-free environment and in an environment with noise, respectively. From the simulation result, the relationship between the environment quality and the anti-collision performance of RFID EPCglobal Gen2 Protocol is clarified.

In Section II, we discuss some problems which may affect the performance of Gen2 [8] anti-collision algorithm, while in Section III, the anti-collision performance evaluation process in AWGN environment by simulation is described .In Section IV, the performance evaluation result of our simulation is presented. In Section V we draw the conclusion.

II. ANTI-COLLISION PROBLEM IN NOISE ENVIRONMENT

A. EPCglobal Gen2 Anti-Collision Protocol

The EPCglobal Gen2 anti-collision algorithm [6] is fundamentally the framed slotted ALOHA [7]. The anti-collision algorithm defined by EPCglobal Gen2 Protocol is illustrated in Fig.1 [5], which describes an inventory process for a multiple tags case. The tags inventory procedure [8] is described as the following:

All tags which receive a Query command from the reader will randomly select a slot number in which to reply between (0~2^Q-1), where the Q value is defined in the Query command and used to define the frame size of one inventory round. Since tags select the slot randomly, there usually exists the situation
that more than one tag select a same slot, in which case the tag responses collision will happen so that the reader could not receive the tag response correctly.

The inventory process could be specified as follows [8]:

1. Reader issues a Query command which includes a Q parameter to define the number of slots (2^Q - 1 slots) in this inventory round.
2. On receiving the Query command, each tag randomly selects a slot and generates a RN16(16 bit random number), then waits for the time slot to reply with its RN16.
3. In case of a single tag reply, in which situation the slot is occupied by only one tag, then the reader could start the tag reply receiving process: If the RN16 is received successfully, then the reader issues an ACK command containing this same RN16 to the tag to request its EPC number. If the EPC number is received correctly, then reader issues a QueryRep or QueryAdjust command to make the tag leave the inventory round and repeat the inventory process in the next slot, potentially causing another Tag to initiate a query-response dialog.
4. In case of a collision, in which situation that more than one tag have selected the same slot, reader issues a QueryRep or QueryAdjust command to repeat the inventory process in the next slot. QueryAdjust repeats a previous Query and may increment or decrement Q to initiate a new inventory round with a new frame length.
5. In case of no reply, in which situation that no tags have selected this slot, reader issues a QueryRep or QueryAdjust command to repeat the inventory process in the next slot.
6. If the slots are expired, the reader issues a QueryAdjust command to adjust Q value (increment, decrement, or leave unchanged), starting the next inventory round with a new frame length (slot number decided by Q).

B. Noise Influences to Anti-Collision

In a RFID system which is generally composed of a reader and a tag [1], the reader and tag communicate with each other through the air interface. So the performance of the RFID system has a close relationship with the air interface environment, which always includes many factors such as noise and reflection that may affect the system performance. In RFID system, the tag receives continuous electronic wave from reader for power supply, so the power of tag response is very weak compared with the signal transmitted by the reader [2]. Therefore, the response signal from the tag is more easily to be affected by the environment quality. So in this paper, we mainly focus on the tag response process in a noise environment simulated by us so as to evaluate the EPCglobal Gen2 anti-collision performance in the practical environment.

C. Problem Analysis in Noise Environment

Fig.2 [8] describes a Gen2 anti-collision tags inventory sequence, which corresponds to the Fig.1. It can be seen from Fig.2 that, typically, three results in the anti-collision inventory process are present: Single Tag Reply, Collided Reply, and No Reply. We focus on the Single Tag Reply process, in which the tag reply data may be corrupted by the noise.

It can be seen that, under theoretical conditions, where the environment is noise free, in case of a single tag reply, reader can inventory the tag successfully without accident. However, in practical environment, where noise always exists, the tag response data could usually be corrupted when noise overwhelms a certain level. So when evaluating the anti-collision performance in a practical environment, it is necessary to focus on the noise influences to the single tag reply process, which is a crucial stage for identifying the tag.

Given that in a Single Tag Reply process, reader issues a Query command and receives the RN16 [8] from tag, then reader issues an ACK command containing the same RN16, and finally receives the EPC code from tag. In order to investigate the noise affects to the tag replying link, we have considered 4 situations that may be caused by noise affects, which may happen in the Single Tag Reply process.

1. In the process of receiving RN16 from the tag, the RN16 preamble was corrupted by the noise in which case the reader will detect no reply of RN16.
2. In the process of receiving RN16 from the tag, while the RN16 preamble was correctly detected, the RN16 data was corrupted by the noise in which case the reader could not detect the RN16 error instantly since there is no CRC checking mechanism in the RN16 reply. However, by including this erroneous RN16 in the ACK command issued to the tag, reader will detect no EPC reply due to the corrupted RN16.
3. Given that the RN16 was received successfully, during the process of receiving EPC reply by issuing an ACK command including the RN16, the EPC preamble was corrupted by noise in which case the reader will receive no EPC reply. Since the tag has received a valid ACK command and replied with the EPC, it will transition to the inventoried state of ACKNOWLEDGED while in fact, the reader did not receive the tag’s EPC.

4. In the EPC receiving process, given that the EPC preamble was detected successfully, the EPC data was corrupted by the noise, in which case the tag will transition to the ACKNOWLEDGED state and reader will detect EPC error by checking the CRC value.

The 4 situations above are considered in our evaluation of the noise influences to the anti-collision algorithm.

III. SIMULATION

The simulation was performed in a C programming environment and the simulation flow chart is shown in Fig.3. From the flow chart it can be seen that the inventory process begins by the issuing of a Query [8] command including an initial Q value, after which all tags will generate a random number between 0 and 2^Q-1. Tags that generate a random number of 0 will be considered to match the reply condition and the total reply number will be recorded after an investigation of all tags. By detecting the total reply number of this slot, there are typically 3 results: No Reply, Collision, and Single Reply.

In No Reply and Collision situation, the Q value adjusting algorithm [8] is implemented. Under condition that Q value changes, this inventory round will terminate ahead of time by the issuing of a QueryAdjust command which updates the Q value and starts a new inventory round with a new Q value. On the other hand, if Q value does not change, the Qb value will be updated by the C value [8], [9], after which a QueryRep command will be issued to start the next slot inventory process if this inventory round does not finish while a QueryAdjust command will be sent to start a new inventory round with a same Q value if this inventory round finishes.

In case of detecting a Single Reply, the tag is made to generate a 16-bit random number (RN16), together with which the tag’s 96-bit EPC number will be encoded in a FM0 [8] coding rule. Then an AWGN noise sequence matching the encoded tag reply data length will be added to the tag reply data. The AWGN noise is simulated by an implementation of the classical Gaussian random number generation [11] method and could be controlled by a setting of SNR (Signal-to-noise ratio) level, in terms of which an Eb/No parameter is used in this simulation to control the AWGN noise level. Then the RN16 reply including a reply preamble mixed with AWGN noise is decoded [10], after which stage the 4 possible situations supposed by us (Part II C.) will be taken into consideration:

During the RN16 receiving process, if any error bit is detected in the preamble after FM0 decoding, then the RN16 preamble will be considered to be corrupted by the AWGN noise, in which case a QueryRep or QueryAdjust command will be issued to end this slot inventory and start the next step inventory.

Given that the RN16 preamble was detected successfully, if the RN16 data is detected to be corrupted by the noise, a QueryRep or QueryAdjust command will be issued.

Under condition that the RN16 was received successfully, an ACK command will be issued including this RN16 to request EPC data from tag. In EPC reply receiving process, after FM0 decoding, if EPC preamble is detected to be corrupted by noise, then a NAK command will be issued to transition the tag state from ACKNOWLEDGED back to ARBITRATE, following which a QueryRep or QueryAdjust command will be issued.

In case that the EPC preamble was detected successfully, however, after CRC checking of the received EPC data, the EPC data is detected to be corrupted by the noise. In this situation, a NAK command will be issued following which a QueryRep or QueryAdjust command will be sent.

If the EPC data was received correctly, the tag will be considered to be successfully inventoried and its EPC data will be recorded following with a QueryRep or QueryAdjust command which will be issued to start the next inventory step. The tag reply detecting and handling process will be repeated until the inventory round ends. In case that the inventory round reaches its end while there are still tags not inventoried left, a QueryAdjust command will be issued to start a new inventory.
round with a same Q value. The whole inventory process will be repeatedly performed until all tags are inventoried.

IV. RESULT

The anti-collision performance evaluation in an AWGN noise environment was performed by an investigation of the identification time cost according to the number of tags and the AWGN noise level. The AWGN noise level was controlled by adjusting the SNR (Signal to Noise Ratio) level in an Eb/No parameter, which value varies from 10 to 0 dB, following which the AWGN noise level will increase accordingly in our simulation. Both the forward link and return link communication speed was set to be 40 kbps and the timing specification was set according to Gen2 Protocol[8]. The simulation process was repeated 100 times under each condition, after which the average results were recorded.

Fig.4 shows the simulation result of the total number of error frames caused by AWGN noise according to the number of tags in environment of three different AWGN noise levels. The error frames consist of all the AWGN noise corrupted reply frames in the single reply processes: the erroneous RN16 preamble frames, the erroneous RN16 frames, the erroneous EPC preamble frames, and the erroneous EPC frames. It can be seen that, under condition of the same number of tags, the inventory process performed in a low Eb/No AWGN noise environment got more error frames. Specifically in Fig.4, in case of 100 tags inventoried, the 4dB Eb/No environment lead to about 500 error frames which is obviously more than that of the other two higher Eb/No AWGN environments.

Fig.5 shows the case of an even lower Eb/No AWGN noise environment of 3dB. It can be observed clearly from Fig.5 that, the error frame number increases exponentially compared with the inventory processes performed in other higher Eb/No AWGN noise environments illustrated in Fig.4. As shown in Fig.5, the Eb/No of 3dB case, when the inventoried tags number is 100, the total error frame number reaches more than 100,000, while this number is only about 500 in result of the inventory process in the 4dB Eb/No environment shown in Fig.4.

Fig.6 shows the simulation result of the tags inventory time cost according to the number of tags in different environments. It can be seen that, given the same number of tags, the inventory process performed in a low Eb/No AWGN noise environment costs more time. Specifically in Fig.6, given that inventorying 100 tags, the 4dB AWGN environment leads to about 3300 ms time cost, which is obvious more than that of the other three inventory processes.

Fig.7 shows the case of an even lower Eb/No of 3dB AWGN noise environment. It can be seen that the tags identification time required increases exponentially compared with the Eb/No of 4dB Eb/No environment. Considering such a long time delay which could not be endured in practical applications, it indicates that in Eb/No of 3dB environment, the anti-collision algorithm could hardly work effectively.
Fig. 7 Tags Inventory time in AWGN environment

Fig. 8 Tags Identification time performance according to Eb/No

Fig.8 shows the time required for identifying 100 tags in different Eb/No environments. It can be seen that, approximately 1.3 times more time in the AWGN environment having Eb/No as 5 dB, and 4.7 times in the AWGN environment having Eb/No as 4 dB is required comparing with the noise free environment.

V. CONCLUSION

In this paper, the performance of the RFID Gen2 anti-collision algorithm in an AWGN noise environment was evaluated.

From the simulation result, it can be seen that, as the environment noise level increases, the anti-collision performance is degraded. Typically, when identifying 100 tags, it requires approximately 1.3 times more time in the 5dB Eb/No environment, and 4.7 times in the 4dB Eb/No environment comparing with the noise free environment. So, in order to make the Gen2 anti-collision algorithm work efficiently, some effort is needed to guarantee the quality of the communication environment. Specifically, since the tags identification efficiency decreases to extremely low level under condition that the SNR falls below 4dB shown in the simulation, so we should at least ensure the Gen2 RFID system works in an environment where the SNR is not less than 4 dB.

On the other hand, in order to improve the performance of RFID system in a practical environment, we are also making effort in designing some schemes to implement on the RFID reader, like the scheme of modifying the demodulation process in the reader [12], and a method of error correction [13] applying on the reader. And more effort will be made towards the improvement of these methods in the future.

ACKNOWLEDGMENT

The research is a part of “Multi-band, Multi-standard RFID Reader” supported by SamSung Techwin Co., LTD R&D Center.

REFERENCES

[7] Luc Andre Burdet, “RFID Multiple Access Methods” ETH Zurich, Summer semester 2004