

Improved Anti-Collision Algorithm for Tag Identification in Future Internet of Things

Mahmoud Moshref

Computer Science Department, the University of Jordan, Amman, 11942, Jordan
E-mail: Moshref2008@gmail.com

Abstract—The most important research in the world in these days, research that looking at the internet of thing's (IoT) topics and their applications. Most of these applications depend on RFID system, which includes RFID readers and tags. The important issues in RFID system or network are how we can reduce anti-collision between readers to identify and read tags data. In these paper, we suggest an Improved anti-collision protocol, which can be used to connect RFID readers with RFID tags and reduce the number of RFID tag's collisions. The simulation shows that an Improved Class-1 Generation 2 algorithm is better than Slotted Aloha, Class-1 Generation-2 (Number of Tags Known), Class-1 Generation-2 (Number of Tags Unknown) algorithms.

Index Terms—Internet of Things, RFID System, Slotted Aloha, EPS global Class-1 Generation-2.

I. INTRODUCTION

Research's that interest in an internet of things (IOT) become from the important research now. This new internet field depends on RFID system (Radio Frequency Identification) in it is different applications, such as medical, geology, and tracking applications.

The most important challenges that limited RFID anti-collision algorithms are accurately estimating the number of tags and the other is improving the efficiency of RFID systems [10]. So our work will concentrate to solve this challenges.

However, when we deploy RFID readers, we get several collision problems such as tag to tag, tag to the reader, reader to the reader [1]. Some research goes to improve RFID anti-collision algorithms through the algorithm search times, communication capacity and throughput of index analysis [11]. Other research goes to solve tags identification collision and improve identification efficiency in RFID system, by using a flood division anti-collision (FDAC) algorithm, these algorithm launches an estimation of the number of tags, then all tags are grouped, and avoid repeatedly transmitting them between the reader and the tags [12].

Some new studies in this field suggest using a tag authentication method based on multi-reader and anti-collision algorithm in order to utilize an efficient RFID system [13]. But others present an anti-collision algorithm based on tag estimation and slot location. The key technique can be divided into two parts: precisely tag

estimation and slot location [14]. In our research, we will go to find an Improved anti-collision algorithm that decreases the collision based on delay time.

To solve the collision problems by using anti-collision protocols. we will suggest an Improved anti-collision algorithm for RFID tags identification to connect with RFID reader. Depending on anti-collision protocols such as Pure Aloha (Additive Link On-line Hawaii), Slotted Aloha, Framed Slotted Aloha (FSA), Dynamic Framed Slotted Aloha (DFSA), Binary Tree, and Class-1 Generation-2 protocol [6]. We will do a comparison between some of these algorithms to get an efficient anti-collision algorithm of tags identification to connect with RFID Readers.

But we must know that RFID readers work with different transmission frequency range, LF (low frequency), HF (high frequency), UHF (ultra-high frequency), and microwave. In our research, we will talk about UHF reader which ranges each to 5m [2].

The rest of the paper organized as following. Section 2 describes an overview and related work. Section 3 simulation and discussion. Section 4 concludes the results and presents the future works.

II. OVERVIEW AND RELATED WORK

In this section, we will present the anti-collision algorithms that we use in our work and some other anti-collision algorithms. These include Slotted Aloha, Class-1 Generation-2 (Number of Tags Known), Class-1 Generation-2 (Number of Tags Unknown) according to related work in this field.

A. Pure Aloha Algorithm

Aloha anti-collision algorithm is based on TDMA method. In this algorithm, the reader command is divided into slots. Also, this algorithm has advanced versions such as Slotted ALOHA, Framed Slotted ALOHA, and Dynamic Framed Slotted ALOHA.

In pure aloha algorithm. As we mentioned before, each reader command is divided into slots of time. In this algorithm. The tag itself decides the data transmission time randomly as soon as it is activated [3]. Also, this algorithm increases the probability of collisions.

When more than one tag responses to reader command at the same time, the collision occurs. This collision may be complete or partial. At this time the reader will give the probability of identifying tags, and the reader sends

the REQUEST "command to the tags. When the tag is selected, the reader broadcast SELECTs command" for identified tag. However, the tags that don't identify response to other commands stop response for request command this shown in figure1.

In figure 1 we can see a partial collision or complete collision between tag 1(001), and tag 2 (010). But tag 3 don't have any collision. So the reader will read tag 1 then the reader will read tag 2 and tag 3 to solve collision problem.

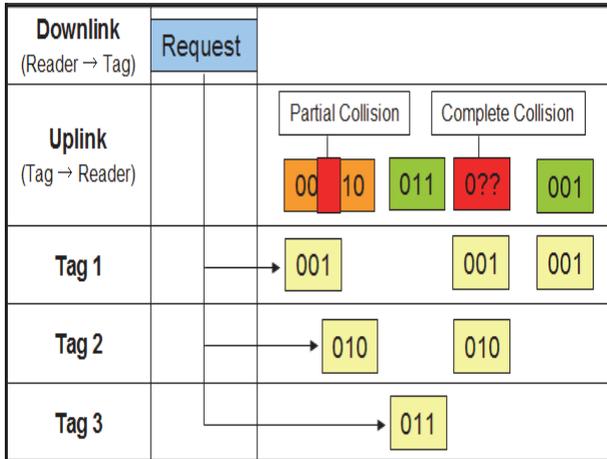


Fig.1. Pure Aloha Anti-collision Algorithm

B. Slotted Aloha Algorithm (SA)

This algorithm can be considered as an advanced version of Aloha algorithm. Tao Cheng, Li Jin. stated that Slotted Aloha is divided into several slots, the tag can select its slot, so this decreases the probability of collision Slotted Aloha algorithm is used if few tags in the area are existed [3].

Zornitza Prodanoff, and Seungnam Kang. the authors stated, that the read cycle is divided into slots in this algorithm and each tag can choose the slot to transmit and receive data randomly. In this algorithm, we have three cases for slots, collide completely, don't collide, or succeed. There is no partial collision, so this algorithm reduces waste of reading cycle [4].

We can say that Slotted Aloha algorithm is an efficient algorithm if there are few tags in the area, but its efficiency will reduce when the number of tags becomes large. When RFID reader sends "REQUEST" command to identify tags, the tags found in the area of this reader response and transmit their data. If we have a three-time-slot on the reader and five tags in the reader's zone, we can find a collision in one or more time slots in this reader, but tags that are read without collision will be succeeded, and send "SELECT" command to the readers, then the reader will send other "REQUEST" commands and so on which represent in figure 2.

This algorithm improved to Framed Slotted Aloha Algorithm (FSA) which is come next.

Reader → Tags	REQUEST	Time Slot 1	Time Slot 2	Time Slot 3	SELECT 1011010
Tags → Reader		Collision	Collision	Success	
Tag 1			1011010		
Tag 2		1010011			
Tag 3		1101101			
Tag 4			11110101		
Tag 5				1011010	

Fig.2. Slotted Aloha Anti-collision Algorithm

C. Framed Slotted Aloha Algorithm (FSA)

This algorithm uses frames and each frame consists of slots. Tao Cheng, Li Jin. Say each reader is divided into frames and each frame consists of several slots. Each frame has fixed size and doesn't change size during tag identification, tags generate a random number to select a slot in one frame [3].

Zornitza Prodanoff and Seungnam Kang say that in each read cycle there are multiple frames and each frame is divided into the same number of slots. So, this algorithm may reduce the number of collisions and give the best results compared with Slotted Aloha [4].

Jun DING, Falin LIU consider that when frame size is equal to tag's number, we can get the greatest throughput. However, when the number of tags is large, the number of collisions will increase and throughput will decrease. On the other hand, when the number of tags is small, the number of the empty slot will increase and throughput will decrease [5]. This represented in figure 3.

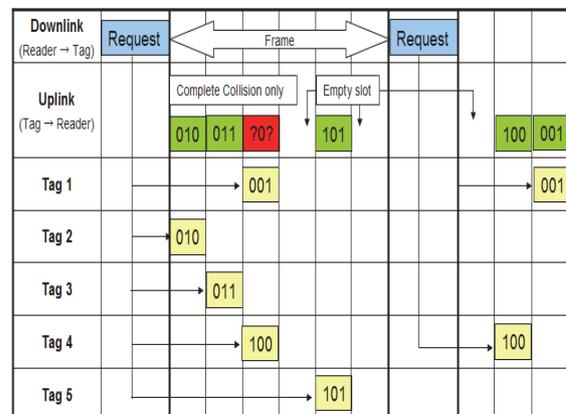


Fig.3. Framed Slotted Aloha Anti-collision Algorithm

Majid Alotaibi, Adam Postula, Marius Portmann. represent that the efficiency of this algorithm will drop down if there is a large number of tags in the field of a reader, then they suggest other anti-collision algorithms such as QT Query Tree, TSA Query Tree Aggressive, and other tree-based methods.

Their results show that QT Query Tree and TSA Query Tree Aggressive algorithms give the best performance respectively in tree-based method and ALOHA based method [6].

D. Dynamic Framed Slotted Aloha Algorithm (DFSA)

Tao Cheng, Li Jin. Says (DFSA) algorithm changes frame size for tag identifications. This algorithm determines frame size by using its information when there are a large number of colliding slots it increases slot number in the frame, and when there are a small number of empty slots, it decreases slot number in the frame. So, this algorithm is more efficient than others [3].

GENG Shu-qin, WU Wu-chen, HOU Li-gang and ZHANG Wang. Illustrate that frame size can be changed dynamically depending on last frame slots with collisions, and a number of empty slots. In this algorithm, we have the interrogator which initializes a read cycle by broadcast request. In each request, the dynamic parameter called frame length is present. When a tag selects a time slot and transmits its ID, you can get one of three outcomes, idle, successful transmits, or collision. Therefore, if the number of collisions is greater than zero, the interrogator estimates the number of tags in the beginning and the number of unread tags. Based on this, the interrogator determines frame length. When the number of slots with collision is over the upper limit, the interrogator increases the number of slots, but if the collision is less than the lower limit, the interrogator decreases the number of slots [7].

As a result, we see that we can control the frame length dynamically as we need, depending on the number of tags in a previous reading cycle, the number of unread tags, and the number of time slots within a collision.

To increase performance for anti-collision algorithm. The researcher in this field invent new version from anti-collision algorithms, we discuss this in next section.

E. EPC global Class-1 Generation-2 Algorithm

M. H. Maheesha H. De Silva. used that EPC global Class-1 Generation-2 Algorithm is another version of Aloha anti-collision algorithm which is known as EPC Class 1 Gen 2, or Q algorithm. This algorithm is developed to increase the performance and reliability of UHF tags. It has several advantages such as providing greater security by including password protection, password authorization, and complex encryption. Also, this protocol provides better read and write for the tag. this protocol offers other advantages such as faster operating speeds, the powerful tag identifying, improved security and privacy, and ability to extend higher function system [8].

This algorithm depends on two algorithms which are dynamic frame slotted Aloha (DFSA) and Q- algorithm with a key parameter C to adjust the frame size. This is stated by Wen-Tzu Chen and Wen-Bin Kao [9]. the EPC global algorithm regulates the interaction between an interrogator (reader) and tags with three procedures, these procedures are the select process, inventory process, and access process. In the select process, the interrogator

selects a tag population for inventory and access.

In inventory process, the interrogator identifies all tags needed to be accessed. The interrogator transmits Query command, tags reply, but the interrogator detects a single tag. In access process, the interrogator transacts with the individual tags. Figure 4 represented the relations or processes that done between reader and tags.

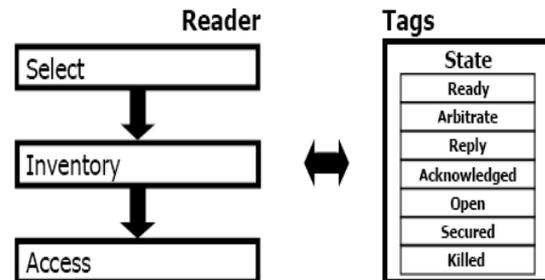


Fig.4. The Process Between Reader and Tags

Q- Algorithm: The idea of this algorithm is to assign reading slots number dynamically by exchanging Q value between reader and tags. The reader transmits a Query command with Q parameter. This parameter value is between 0 and 15. Tags that are participating to identification select a random value between 0 and $2^Q - 1$ and store it in slot counter. This process will help in reducing collisions M. H. Maheesha H. De Silva [8].

When the interrogator transmits Query command, there are three possible outcomes for tag response: single tag replay (value of Q parameter remains the same without changes), collided replay or no replay, in these cases value of Q parameters is modified. The interrogator maintains Q value by using a floating-point parameter Qfp, in each inventory round we can use $(Q = \text{round}(Qfp))$, and small key parameter C, the value of C is > 0.1 and < 0.5 , this value is used to adjust the frame size Wen-Tzu Chen and Wen-Bin Kao [9].

As we see in figure 5, in the first case, when there is one tag reply, Q value is used without any change and we can use QueryRep. In the second case, when there is collided tags replay, we must increase Qfp with C value, and we can use QueryAdjust after updating Q value. In the third case, when there is no tag replay, we must decrease Qfp with C value, and we can use QueryAdjust after update Q value.

F. Resent Anti-Collision Algorithms

Zhihui Fu, Fangming Deng, and Xiang W presented a 4-ary query tree Additive Link On-line HAWAII (ALOHA) protocol is that combines the merits of query tree algorithm and frame slotted ALOHA, and avoids their weaknesses [10].

Hua Huo, Jun Qiang Liu, and Yong Jie Wang proposed a new anti-collision algorithm based on flood division idea to reduce data bits transmitted and identification time delay [12]. In Chang-Su Kim1, Bong-Im Jang and Hoe-Kyung Jung analyzed tag identification time to minimize tag collisions and improve tag identification

time by applying various tag anti-collision algorithms in the suggested method based on an optimum number of readers used in the RFID system environment [13].

G. The Proposed Anti-Collision Algorithm

Wen-Tzu Chen and Wen-Bin Kao. proposed new anti-collision algorithm by splitting C parameter into two parameters to increase the reading speed. When single tag replies, the value of Q parameter remains the same without change. When no tag replies, we must decrease Q_{fp} with C1 value, the Ceil function is used to give the smallest integer greater than the function argument.

We can use QueryAdjust after updating Q value, and when collided tags reply, we must increase Q_{fp} with C2 value, the Floor function is used to give the greatest integer less than the function argument, and we can use QueryAdjust after updating Q value [9]. We can see this in figure 6.

III. SIMULATION AND DISCUSSION

We do simulation in this part by using RFID simulator built it on MATLAB 7.0 tools to improve EPC global Class-1 Generation-2 protocol using Q- Algorithm to connect between RFID reader and tags. This simulator is developed to evaluate the performance of slotted Aloha and EPC global Class-1 Generation-2 Protocols [8].

In this simulator, we can make a comparison between anti-collision protocols to get the efficient protocol. Our contribution is concentrated on adding a new anti-collision algorithm to the RFID simulator, making a

comparison between these algorithms based on delay time when RFID readers can identify RFID tags, and finding the efficient algorithm.

After we added a new algorithm to the RFID simulator, we could consider this algorithm as an Improved algorithm of EPC global Class-1 Generation-2. And we will add new parameters which are controlling this algorithm.

In the Q algorithm, we have parameters such as Q value, Q_{fp} value, C value, and reader – tag rate parameter. While in the new algorithm we have new parameters such as C1 and C2 values.

A. Slotted ALOHA Algorithm Simulation

We used RFID simulator to calculate the delay time to any number of RFID tags when these tags reply to RFID reader Query. Our result represented in table 1.

Table 1. Simulation Result for Slotted ALOHA Algorithm

Number of Tags	Delay Time (slot time =1 ms)
50	130.665
100	265.899
150	401.741
200	537.093
250	674.931
300	807.200
350	943.729
400	1079.69
450	1217.09
500	1351.06

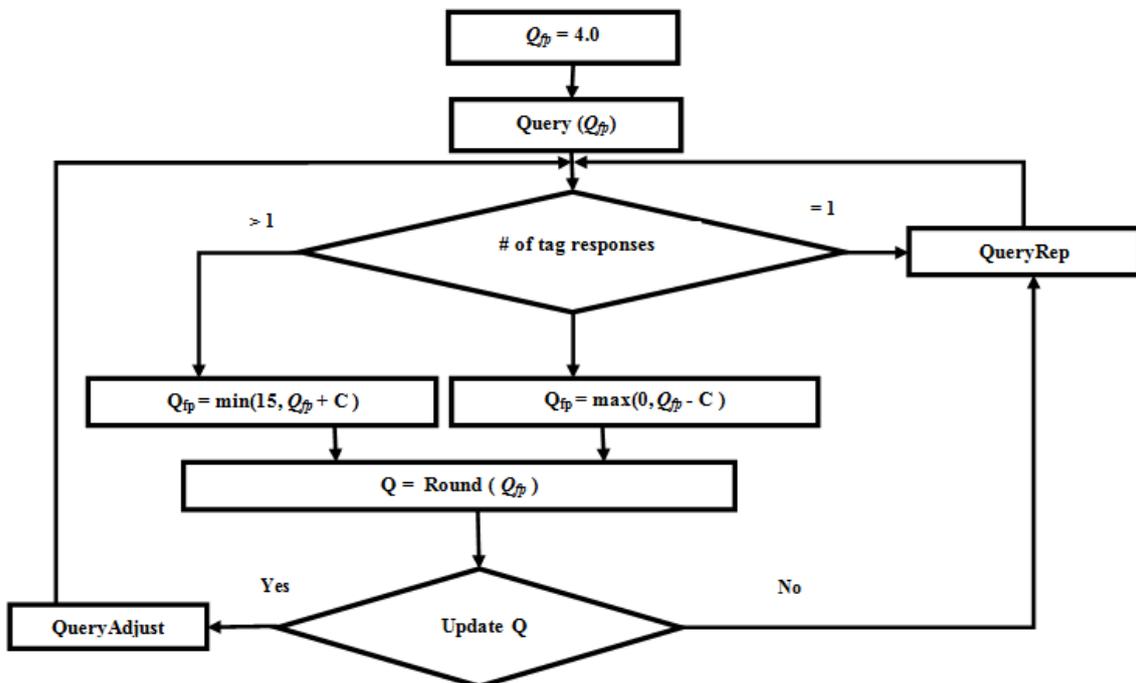


Fig.5. The basic Method for Q- Algorithm

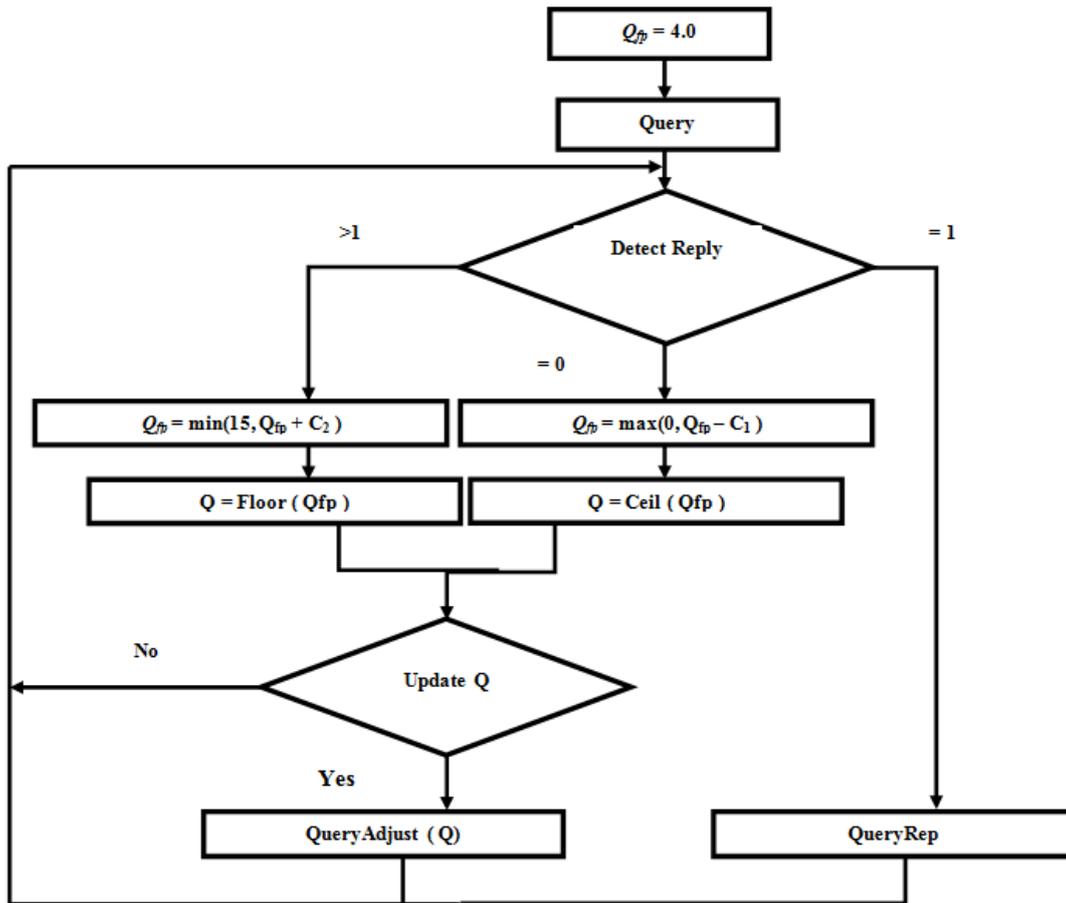


Fig.6. The Proposed Method for Q- Algorithm

B. EPC global Class-1 Gen-2 Algorithm Simulation

In this section, we make a simulation for EPCglobal Class-1 Gen-2 Algorithm using RFID simulator. In the simulator, we have two options for EPCglobal Class-1 Gen-2 Algorithm. The first option gives the time to identify all tags when the RFID reader knows the number of tags and the second option gives the time to identify all tags when the RFID reader doesn't know the number of tags [8].

We interest in getting a good result in this simulation, so we changed parameters such as Q algorithm parameters (Q value, Qfp value, C value), and reader – tag rate parameter.

Tags Unknown: in this algorithm, we will enter the number of tags and values for (Q, Qfp, and C) parameters. In slotted ALOHA algorithm we used fixed slot time 1 (ms), but in this algorithm, we put tag-reader rate, which controls the slot time. Here we impose that RFID reader doesn't know tags number.

In the equations below, we have different time parameters, all of these parameters depend on tag – reader rate which is measured in (kbps). So, when we change tags-reader rate, we can get time for tags identification in three possible outcomes for tag response as we noted before: single tag replay collided replay or no replay.

The equations are:

$$Single\ tag\ reply\ time = T_{Query} + (2 * T1) + (2 * T2) + TRN16 + TACK + TEPC + TXPC + TCRC \quad (1)$$

$$Collided\ reply\ time = T_{Query} + T1 + T_{Collision} \quad (2)$$

$$No\ tag\ reply\ time = T_{QueryRep} + T1 + T3 \quad (3)$$

Where:

T1: Time from reader transmission to tag response.

T2: Time from tag response to reader transmission.

T3: Time a reader waits, after T1, before it issues another command.

TQuery: Time for the reader to start inventory and to sending a Query command, with the length of 22 bits.

TQueryRep: Time for repeating for Query Command, with the length of 4 bits.

TCollision: Time for collision-detecting and more one tag replay with a 16-bit random number (RN16), with the length of 16 bit.

TACK: Time for Acknowledgment for a single tag, with the length of 18 bits.

TXPC: Time for two XPC words or time to implement tag words, with the length of 16 bits.

TEPC: Time for reading electronic product code EPC in the tag memory, with the length of 144 bits.

TCRC: Time for cyclic redundancy check, with length

16 bit.

TRN16: time for tags of a 16-bit random number when its slot counter reaches zero. With a length of 16 bit.

R: which is the communication rate between tags and readers in (Kbps).

Tpri: which is the backscatter link pulse repetition interval or the tag-to-reader link period.

From these parameters and from previous equations we can calculate the time for single tag replay, collided replay, or no replay. So we can find each of the parameters as follows.

$$T_{pri} = \frac{1}{R * 1024} * 1000 \tag{4}$$

$$T_2 = 20.0 * T_{pri} \tag{5}$$

$$T_1 = T_2 \tag{6}$$

$$T_3 = T_{pri} \tag{7}$$

$$T_{Query} = \frac{22}{R * 1024} * 1000 \tag{8}$$

$$T_{QueryRep} = \frac{4}{R * 1024} * 1000 \tag{9}$$

$$T_{ACK} = \frac{18}{R * 1024} * 1000 \tag{10}$$

$$TRN16 = \frac{16}{R * 1024} * 1000 \tag{11}$$

$$T_{Collision} = \frac{16}{R * 1024} * 1000 \tag{12}$$

$$T_{XPC} = \frac{32}{R * 1024} * 1000 \tag{13}$$

$$T_{EPC} = \frac{144}{R * 1024} * 1000 \tag{14}$$

$$T_{CRC} = \frac{16}{R * 1024} * 1000 \tag{15}$$

These equations (1 to 15), depending on communication rate between tags and readers (R parameter), we will use RFID simulator for EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags. In these cases, the simulation will continue until (Q value = 0), because RFID reader doesn't know the total number of tags [8].

We simulated fixed number of tags (total number of tags= 200), fixed C value so let (C = 0.4), fixed Q value, let (Q =7), Qfp value, let (Qfp =7), and changing communication rate between tags and readers (R parameter) to get the best result. we use RFID simulator to simulate EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags, then we present results in table 2.

Table 2. EPC global Class-1 Generation-2 Algorithm when Tags Number Unknown (Using Different Reading Rate)

Reader- Tag Rate (R)	Delay Time in ms
20	3974.69
40	1984.98
80	994.451
100	794.513
120	661.799
160	496.719

From previous results, we recommend using RFID readers with highest reading rate (160 Kbps) to get good results, as we see that when we increase the communication rate or when we use RFID readers with the highest rate the delay time decrease.

In the second case, we will do simulation by fixing the number of tags (total number of tags= 200), C value so let (C = 0.4), and communication rate between tags and readers let (R= 160), and will change Q value to get a good result for delay time, Q value between 1 to 15. We use RFID simulator to simulate EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags, then we present results in table 3 in the following.

Table 3. EPC global Class-1 Generation-2 Algorithm when Tags Number Unknown (Using Different Q Value)

Q Value	Delay Time in ms
1	501.447
2	501.311
3	500.58
4	500.199
5	500.00
6	498.712
7	498.229
8	498.313
9	498.325
10	498.329
11	498.600
12	499.094
13	499.097
14	499.157
15	499.658

From table 3, We present the result in figure 7, as we

can see the lowest delay time is when (Q = 7), so we will choose these values and fixed Q in the next experiments. In [15] propose an optimal Q algorithm that determines the optimal values of Q according to the number of remaining tags and in turn to optimize tag identification speed (TIS).

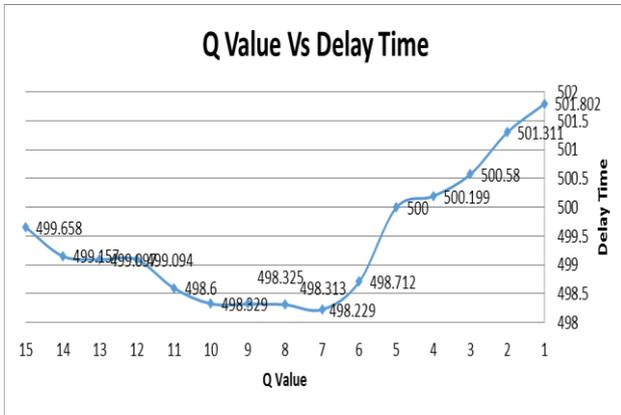


Fig.7. Q Value Vs Delay Time

In the third case, we will do simulation by fixing Q value so let (Q= 7), and communication rate between tags and readers let (R= 160), then let C value (0.2, 0.3, 0.4), and the total number of tags (50, 100, 150, 200, 250, 300, 350, 400, 450, 500). then we present results in table 4.

Table 4. EPC global Class-1 Generation-2 Algorithm when Tags Number Unknown with C Value (0.2, 0.3, 0.4)

Number of Tags	Delay Time in ms		
	C=0.2	C=0.3	C=0.4
50	121.802	122.199	122.823
100	244.909	245.496	247.36
150	368.483	370.049	372.053
200	492.322	494.435	496.252
250	617.263	619.676	621.841
300	740.817	744.390	747.138
350	864.777	868.813	872.890
400	990.433	992.964	998.334
450	1113.94	1118.28	1123.03
500	1238.32	1242.79	1248.37

Table 4, show that when we use (C = 0.2), we get the lowest delay time for a different number of tags when we use (C= 0.3), get delay value higher than the previous case. And when we use (C= 0.4), then we get delay value more than when we use 0.2, or 0.3.

Tags are known: We will use RFID simulator for EPC global Class-1 Gen-2 Algorithm when the RFID reader knows the total number of tags. In this case, the simulation will continue until RFID reader reads all tags.

When RFID reader identifies all tags, the simulation is stopped.

In this algorithm, we will do simulation by fixing Q value so let (Q = 7), and communication rate between tags and readers let (R= 160), C value (0.2, 0.3, 0.4), and the total number of tags (50, 100, 150, 200, 250, 300, 350, 400, 450, 500). we use RFID simulator to simulate EPC global Class-1 Gen-2 Algorithm when the RFID reader knows the total number of tags, we present results in table 5. We see that we get the best results with the lowest delay time when we use EPC global Class-1 Gen-2 Algorithm when the RFID reader knows the number of tags, but we get the largest delay time when we use EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags.

This means that the EPC global Class-1 Gen-2 Algorithm when the RFID reader knows the number of tags gives a better performance more than EPCglobal Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags.

Table 5. EPC global Class-1 Generation-2 Algorithm when Tags Number Known with C Value (0.2, 0.3, 0.4)

Number of Tags	Delay Time in ms		
	C=0.2	C=0.3	C=0.4
50	120.753	121.530	122.250
100	243.593	245.401	246.665
150	367.339	369.321	371.257
200	491.378	493.878	496.030
250	615.739	618.833	621.282
300	739.596	743.090	746.874
350	863.587	867.525	871.889
400	989.109	992.455	997.442
450	1113.32	1118.18	1122.18
500	1237.98	1242.41	1247.65

So, from all of the above results, we can use parameters with the following values, (Q = 7), (Q_{fp} = 7), communication rate between reader and tags (R= 160), and (C = 0.2), to get the best results (lowest delay time).

We added a new algorithm, which we called Improved EPC global Class-1 Gen-2 Algorithm when the RFID reader knows the number of tags. We will illustrate this in next section.

C. Improved EPC global Class-1 Generation-2 Algorithm (Tags Known)

In the improved algorithm, we divide C parameter into two parameters C1 and C2. Value of C1, C2 parameters must be (0.1<C1 <0.5), and (0.1<C2 <0.5).

In this case, to determine C1, and C2 values, we simulate to get the best values, before we use these values to get the lowest delay time. So let (Q = Q_{fp} = 7), (R = 160), (C1= 0.15), number of tags (number of tags = 100), and change C2 (0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45). We present this in table 6. And illustrate it in figure 8.

Table 6. Improved EPC global Class-1 Generation-2 Algorithm When Fixed C1 Value to 0.15

<i>C2 Value</i>	<i>Delay Time in ms</i>
0.15	244.850
0.20	243.200
0.25	242.449
0.30	241.931
0.35	241.733
0.40	241.808
0.45	242.001

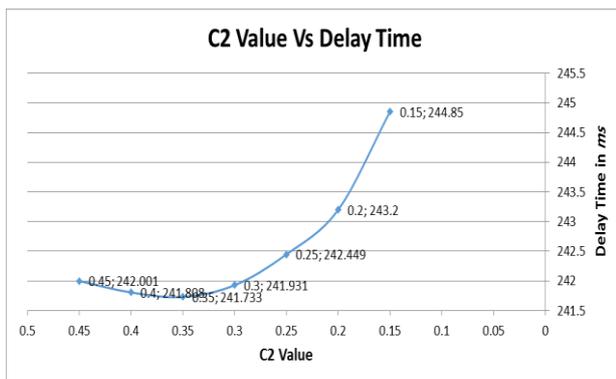


Fig.8. C2 Value vs Delay Time

From table 6 and figure 8, we find that C2 value gives the best delay time (lowest delay). When (C2= 0.35), we will use this value to do the simulation.

To determine the best value for C1, we will do simulation for Improved EPC global Class-1 Generation-2 Algorithm one more time. So let (Q= Qfp= 7), (R= 160), (C2= 0.35), number of tags (# tags= 100), and change C1 (0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45). We present this in table 7, and figure 9.

Here we find that the best C1 value when we use (C1=0.15) because it gives lowest delay time value as presented in table 7 and figures 9.

Table 7. Improved EPC global Class-1 Generation-2 Algorithm When Fixed C2 Value to 0.35

<i>C1 Value</i>	<i>Delay Time in ms</i>
0.15	241.690
0.20	242.539
0.25	244.093
0.30	245.587
0.35	247.371
0.40	250.221
0.45	252.251

In this algorithm, we must increase Qfp by C2, and decrease Qfp by C1. We can conclude that the decrement of Qfp is required to be less than its increment [9]. All of this results will lead us to find an Improved anti-collision algorithm depending on EPC global Class-1 Gen-2 Algorithm.

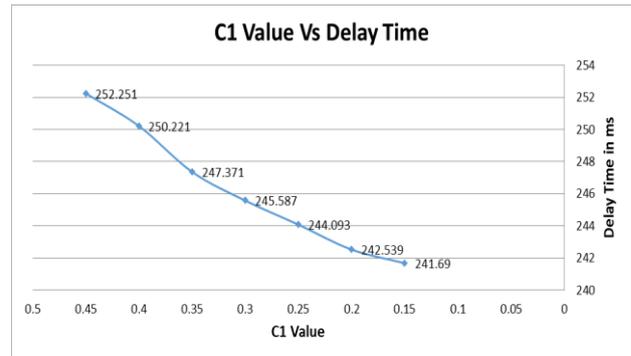


Fig.9. C1 Value vs Delay Time

Simulation results for Improved EPC global Class-1 Generation-2 Algorithm using values, (Q = Qfp = 7), (C1= 0.15, C2= 0.35), (R = 160 kbps), and tags number (50, 100, 150, 200, 250, 300, 350, 400, 450, 500). presented, and Table 8.

Table 8. Improved EPC global Class-1 Generation-2 Algorithm When Tags Number Known with C1=0.15, C2=0.35

<i>Number Of Tags</i>	<i>Delay Time in ms</i>
50	120.977
100	242.080
150	362.866
200	485.256
250	606.670
300	727.326
350	849.777
400	971.312
450	1093.42
500	1214.66

When we compare the results that we got from Improved EPC global Class-1 Generation-2 Algorithm with those obtained from EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags, EPC global Class-1 Gen-2 Algorithm when the RFID reader knows the number of tags, and Slotted ALOH Algorithm, we found that this algorithm gives good simulation results (lowest delay time).

We can do a comparison between those algorithms as we see in figure 10, we do simulation for all Algorithms using RFID simulator. In figure 10 we use the following parameters: (number of tags = 500), (reader- tags- rate =160 Kbps), (C Value = 0.2), Based on proposed idea in [15].

We adjust the parameter Q based on the number of remaining tags so in our work, an optimal Q value when (Q Value = Qfp Value = 7), and for Enhanced EPCglobal Class-1 Generation-2 Algorithm we use (C1 Value = 0.15, C2 Value = 0.35).



Fig.10. RFID Simulator Interface for All Algorithms, With 500 Tags

In figure 10 we use colors for each algorithm, pink for Slotted Aloha, red for EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags, blue for EPC global Class-1 Gen-2 Algorithm when the RFID reader know the number of tags, and green for an Improved EPC global Class-1 Generation-2 Algorithm.

The green color gives us the lowest delay time when we use it to read 500 tags. Since the delay time when we use Slotted Aloha equal 1355.37 ms, the delay time when we use EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the number of tags equal 1238.85 ms, the delay time when we use EPC global Class-1 Gen-2 Algorithm when the RFID reader know the number of tags equal 1236.65 ms, however, the delay time when we use the proposed Improved EPC global Class-1 Gen-2 Algorithm equal 1214.13 ms. In table 9, we put all simulation result when we use a different number of tags. based on these results, we do a comparison between these algorithms.

We got a good result when we used Improved EPC global Class-1 Gen-2 Algorithm, so we can suggest using this algorithm in RFID System in real life to make the connection between RFID readers and tags.

We will have recommended using this algorithm as an effective anti-collision algorithm to connect RFID tags with RFID readers in the field of the internet of things (IOT).

The improved algorithm could reduce collision between RFID tags when RFID readers read it and could reduce delay time when RFID readers read tags. Also, this algorithm could save energy by reducing energy consumption.

Table 9. Simulation Result for All Algorithms

Number of Tags	Delay Time in ms			
	Slotted ALOHA	EPCglobal Class-1 Gen-2 Algorithm $Q = Q_{fp} = 7$		
		Number of Tags Unknown $C=0.2$	Number of Tags Known $C=0.2$	Improved Algorithm $C1=0.15, C2=0.35$
50	130.665	121.802	120.753	120.977
100	265.899	244.909	243.593	242.080
150	401.741	368.483	367.339	362.866
200	537.093	492.322	491.378	485.256
250	674.931	617.263	615.739	606.670
300	807.200	740.817	739.596	727.326
350	943.729	864.777	863.587	849.777
400	1079.69	990.433	989.109	971.312
450	1217.09	1113.94	1113.32	1093.42
500	1351.06	1238.32	1237.98	1214.66

IV. CONCLUSION & FUTURE WORK

In this research, we do a simulation to find an effective anti-collision algorithm to do the connection between tags and readers. We used RFID simulator which contains three algorithms: Slotted Aloha, EPC global Class-1 Gen-2 Algorithm when the RFID reader doesn't know the total number of tags, EPC global Class-1 Gen-2 Algorithm when the RFID reader know the total number of tags, and we proposed a new algorithm called it Improved EPC global Class-1 Gen-2 Algorithm.

Finally, a new algorithm Improved EPC global Class-1 Gen-2 Algorithm, give good results for connecting tags with readers. This is based on dividing C value to C1 value and C2 value. Also, we used $Q = Q_{fp} = 7, R=160$ Kbps, $C1 = 0.15$, and $C2=0.35$. This is because we use dynamic frame size when we make the connection between tags and readers. We use a small value ($C1=0.15$) to decrease, and large value ($C2=0.35$) to increase the Q value.

Future work in this field focus on the method to obtain the optimal frame size using anti-collision algorithms, and how we can verify the performance of each algorithm in terms of identification time, throughput, system efficiency, accurately estimating the number of tags, and collision ratio.

REFERENCES

- [1] Daniel W. Engels. "The Reader Collision Problem". auto-id center massachusetts institute of technology, 77massachusetts avenue, bldg 3-449, cambridge, ma 02139-4307, February 1, 2002.
- [2] Okkyeong Bang (ICU), Ji Hwan Choi (ICU), Dongwook Lee (ICU), Hyuckjae Lee (ICU). "Efficient Novel Anti-collision Protocols for Passive RFID Tags". Auto-ID Labs White Paper WP-HARDWARE-050 March 2009.
- [3] Tao Cheng, Li Jin. "Analysis and Simulation of RFID Anti-collision Algorithms". School of Electronics and Information Engineering, Beijing Jiaotong University, Beijing 100044, P.R. China. Feb. 12-14, 2007 ICACT2007.
- [4] Zornitza Prodanoff, and Seungnam Kang. "RFID Model for Simulating Framed Slotted ALOHA Based Anti-Collision Protocol for Muti-Tag Identification".

- University of North Florida, National Seoul University. USA, South Korea.
- [5] Jun DING, Falin LIU. "Novel Tag Anti-Collision Algorithm with Adaptive Grouping". Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, China. *Wireless Sensor Network*, 2009, 1, 475-481.
- [6] Majid Alotaibi, Adam Postula, Marius Portmann. "Tag Anticollision Algorithms in RFID Systems - a New Trend". *Information Technology and Electrical Engineering*, The University of Queensland Brisbane QLD 4072, AUSTRALIA., *WSEAS TRANSACTIONS ON COMMUNICATIONS*. Issue 12, Volume 8, December 2009.
- [7] NG Shu-qin, WU Wu-chen, HOU Li-gang, and ZHANG Wang. "Anti-collision Algorithms for Multi-Tag RFID". LSI and System Lab, Beijing University of Technology, Beijing 100022, P. R. China.
- [8] M. H. Maheesha H. De Silva. "AN EXPERIMENTAL STUDY OF EPCGLOBAL CLASS-1 GENERATION-2 ANTI-COLLISION PROTOCOL FOR RFID SYSTEMS". Master of Science, Wichita State University, August 2010.
- [9] Wen-Tzu Chen and Wen-Bin Kao. "A Novel Q-algorithm for EPCglobal Class-1Generation-2 Anti-collision Protocol". *World Academy of Science, Engineering, and Technology* 78 2011.
- [10] Zhihui Fu, Fangming Deng and Xiang Wu. "Design of a Quaternary Query Tree ALOHA Protocol Based on Optimal Tag Estimation Method". MDPI School of Electrical and Automation Engineering, East China Jiaotong University, Nanchang 330013, China. 30 December 2016.
- [11] Jiangang Jin. "Research on an Improved RFID Anti-collision Algorithm in the Internet of Things". *International Journal of Security and Its Applications*. Software Technology Vocational College, North China University of Water Resources and Electric Power, Zhengzhou 450045, China. 2016.
- [12] Hua Huo, Jun Qiang Liu, and Yong Jie Wang. "Flood Diversion Algorithm for Anticollision in RFID System". *International Journal of Distributed Sensor Networks*. School of Software, Key Laboratory of Intelligent Information Processing, Henan University of Science and Technology, Luoyang 471003, China. Volume 2015.
- [13] Chang-Su Kim, Bong-Im Jang, and Hoe-Kyung Jung. "Performance Analysis of Anti-collision Algorithm for Tag Identification Time Improvement ". *International Journal of Software Engineering and Its Applications*. Pai Chai University, Doma2-Dong, Seo Gu, Daejeon, Korea. Vol.8, No.3 (2014), pp.1-10.
- [14] Yong Liu, and Xingzhong Xiong. "An Anti-Collision Algorithm based on Slot Location and Tag Estimation". *International Conference on Computer and Information Application (ICCIA 2012)*. Sichuan University of Science & Engineering.
- [15] Chonggang Wang, Mahmoud Daneshmand, Bo Li, and Kazem Sohraby. "Performance Improvement of Generation-2 RFID Protocol". *ICST*. July 2008.

Authors' Profiles



Mahmoud Moshref Mahmoud obtained his Bc in computer Science from An-Najah University in 2003, then he completed his study in a master of computing from Birzeit University in 2012. he worked at Palestine Technical University, Tulkarem, Palestine as a part-time lecturer. Now he is a Ph.D. student in The University of Jordan. He interests in networks, network security, and an internet of things, and RFID system fields.

How to cite this paper: Mahmoud Moshref, "Improved Anti-Collision Algorithm for Tag Identification in Future Internet of Things", *International Journal of Computer Network and Information Security(IJCNIS)*, Vol.9, No.3, pp. 11-20, 2017. DOI: 10.5815/ijcnis.2017.03.02