

# Anti-collision Protocol Development for Passive RFID Tags

Nowshad Amin, Puah Weng Lin

Dept. of Electrical, Electronics and System Engineering, Faculty of Engineering

Solar Energy Research Institute

National University of Malaysia

43600 UKM, Bangi, Selangor,

MALAYSIA

Tel: +60-3-89216325, Fax: +60-3-89216146

*Abstract:* - Tag identification is an important measure in RFID systems with applications for monitoring and tracking. A RFID reader recognizes tags through communication over a shared wireless channel. When multiple tags transmit their IDs simultaneously, the signals collide and this collision disturbs a reader's identification process. Collision problem can be divided into reader collision and tag collision. Tag collision is taken into account in this study since most of the applications involve tag collision problem. Therefore, an anti-collision protocol is developed to solve this issue. This study focuses on the most widely used passive tag, as it is simple and cheaper. Here, a circuit has been proposed that can implement the anti-collision protocol for passive tags, and its functionality has also been investigated. Among several anti-collision protocols, the Binary Tree Protocol is chosen and a circuit based on Binary Tree Protocol has been constructed using SIMUAID simulator. In Binary Tree Protocol, the tag ID with lowest bit has the priority to be identified first. The circuit is able to successfully solve the anti-collision problem in RFID system to recognize the tag's ID.

*Key-Words:* - RFID, Passive RFID tag, Anti collision protocol, Binary tree protocol

## 1 Introduction

RFID (radio frequency identification) tags can be divided into two categories, such as passive tags and active tags. Passive tags require no internal power source, whereas active tags require power source. The circuit developed is based on passive RFID system since the passive RFID systems have more practical use. Furthermore, passive tag is simple and cheaper. RFID system is always made up of three main components; the transponder, the interrogator and the data management system.

The transponder represents the actual data-carrying device of an RFID system. When the transponder, which does not usually possess its own voltage supply (battery), is not within the interrogation zone of a reader, it is totally passive. The transponder is only activated when it is within the interrogation zone of a reader. The power required to activate the transponder is supplied to the transponder through the coupling unit (contactless) of the reader. The transponder communicates with the reader, thus the reader can get the data stored in the transponder. If there are many tags within the interrogation zone of the reader, and they communicate with the reader at the same time, the reader will not know which one it should communicate with at first. A problem which is called as "collision" will arise. The reader will get the wrong information of the tags if it cannot tell the individual IDs of the tags [1-4].

To solve collision problems in a RFID system, there are three methods, such as spatial domain method, frequency domain method and time domain method. However, in most RFID systems, time domain method which means every tag communicates with the reader in the different time is used as it is easier to be put in practice. Binary

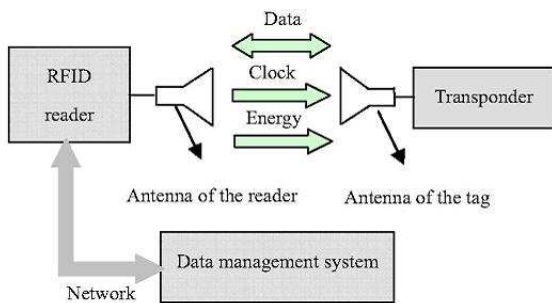


Fig.1 RFID (passive) system structure

Tree Protocol used in the developed circuit belongs to time domain method.

## 2 Methodology: Implementation of Binary Tree Protocol

Binary Tree Protocol requires tags to remember the previous inquiring results, thus reduces the average inquiry time. However it makes the circuit structure more complex, and also makes the tag more sensible to the imperfection of power supply, which comes from the electro-magnetic wave emitted from the reader.

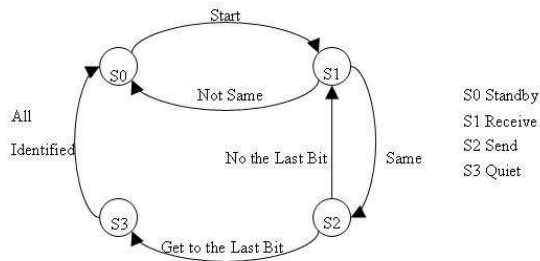


Fig 2. State Diagram of Binary Tree Scheme

Table 1: The inquiry process of Binary Tree Protocol

Reader sends	Reader Receives	Tag answers				Identified tag (remaining tags reset to standby)
		0001	0010	1000	1100	
start						
0	0	0	0			
0	*	0	1			
0	1	1				0001
start			standby			
0	0		0			standby
0	1		1			
1	0		0			0010
start						
0	-			0	1	
1	*			0		
0	0			0		
0	0	quiet		0	standby	1000
start						
1	1				1	
1	0				0	
0	0				0	
start						
1	-					
Eliminated						

In this protocol, once a tag has been completely identified, it will be eliminated. In every tag, there is a pointer. Every time the tag is reset, the pointer points to the highest bit of the tag's ID. With the ongoing of inquiring, it moves towards the lowest bit. During inquiring, the reader sends one inquiring bit at one time. The tags whose pointed bit is the same as the inquiring bit will back send their next hit to the reader while those who aren't will convert to the state of "quiet", and will not answer the remaining inquires in this round of inquiring until one tag has been eliminated and all the remaining tags are reset. If the reader senses a non-collision answer, it uses it as its next-step inquiring bit. Otherwise if a collision is sensed, reader uses '0' as next-step inquiring bit. Thus for every round of inquiry, one tag, and only one tag will be identified, when its pointer finally gets to the lowest bit of the identified tag. Then the identified tag will be eliminated and all tags that have already entered the state of "quiet" will be reset, followed by a new round of inquiring beginning from the highest bit. After *I* round of inquiry, the IDs in the *k* tags will all be identified. Table 1 illustrates the process of inquiry, supposing 4 tags with IDs 0001, 0010, 1000 and 1100 respectively. Here, the asterisk (\*) denotes a collision. Figure 2 shows the state diagram of the Binary Tree Protocol.

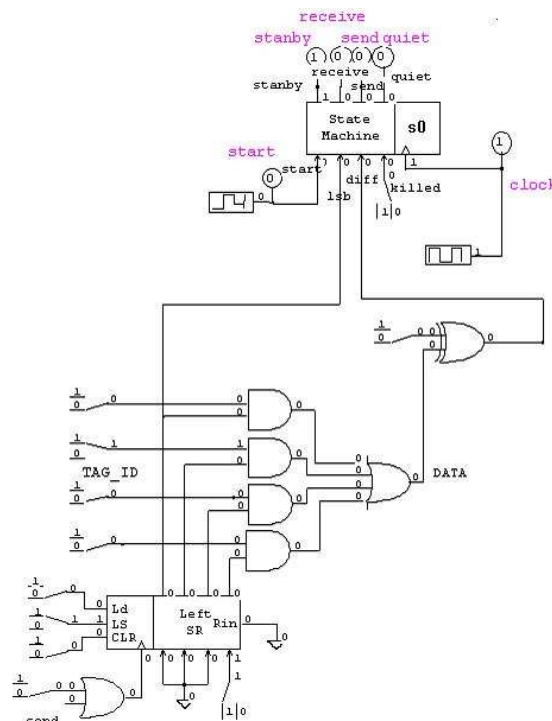


Fig 3. Schematic of Binary Tree Protocol

### 3 Results and Discussion

Chronological test cases have been investigated to develop and analyze the circuit performance. All these situations are named as case studies. The following results describe only 4 cases for detailed investigation as it evolves to the final circuitry for successful anti-collision protocol implementation in RFID systems.

#### 3.1 Case 1

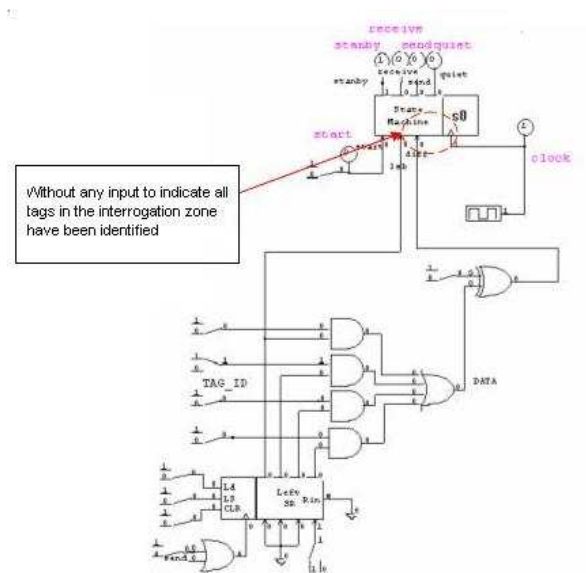


Fig.4 Circuit diagram for case 1

Case 1 assumes 2 tags in the interrogation zone, tag 1 with ID 0100 and the simulated tag with ID 0010. Figures 4 and 5 show the circuit diagram for case 1 and the simulation result, respectively. Note ‘1’ in the Figure 5 means that the ‘send’ state is on and the tag is sending ID bit to the reader. This is in fact a failure test case. The simulated tag (ID =0010) has the priority to be identified first compare to another tag (ID=0100) according to the Binary Tree Protocol. As shown in figure 5, it can be noticed that the state machine change from the quiet state to standby state after 1 clock time. It is because the state machine was set to automatically change from state 3 (quite state) to state 0 (standby state). It should be one condition to change from state 3 (quiet state) to state 0 (standby state). The condition is that all the tags in the interrogation zone of the reader are being identified. Since another tag has not been identified, if the reader changes to state

standby automatically, it will continue answering to the reader’s inquiry. Due to that, the reader cannot recognize the other tag (ID=0100).

The reader should send a signal to the tag to indicate that all tags in the interrogation have been identified by the reader. An input for this signal is added to the state machine. Therefore, the tag will change from state ‘quiet’ to state ‘standby’, if the reader sends this signal to the reader.

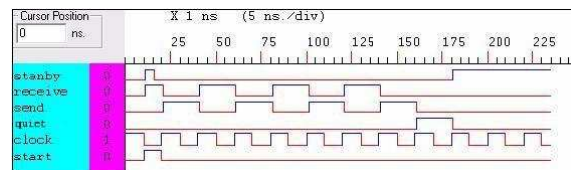


Fig.5 Results for case 1

#### 3.2 Case 2

Case 2 assumes 2 tags in the interrogation zone, where tag 1 has ID 0001 and the other simulated tag with ID 0010. Figure 6 shows the simulation result for case 2. From figure 6, it can be noticed that the ‘send’ state and ‘receive’ states are on at the same time. In fact, the ‘send’ state should be on after the ‘receive’ state. It is because state machine change to send depends on the receive bit compared to the pointed ID bit. It is unlikely to happen at the same time. This error is for the wrong timing of ‘start’ input from the reader that is manually configured.

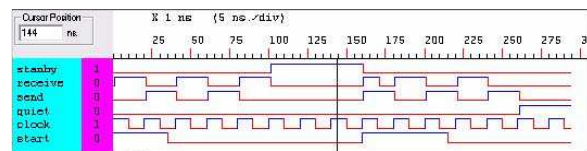


Fig.6 Results for case 2

#### 3.3 Case 3

The next test case 3 assumes 2 tags in the interrogation zone with a tag having ID 0100 and the other simulated tag with ID 0010. Table 2 illustrates the inquiry process of case 3.

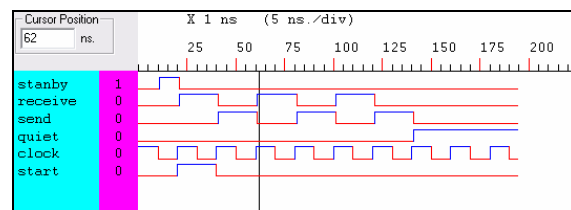


Fig.7 Results for case 3

Note ‘1’ in the figure 7 means that the ‘send’ state is on and the tag is sending ID bit to the reader. After comparing the simulated results with the actual inquiry process, the simulated results are found to match the actual result.

Reader sends	Reader Receives	Tag answers		Identified tag (remaining tags reset to standby)
		0100	0010 (simulated tag)	
start				
0	*	1	0	
0	1		1	
1	0	standby	0	0010
start				
0	1	1	quiet	
1	0	0		
0	0	0		0100

Table 2: The inquiry process of case 3  
\*denotes collision

Referring to table 2, the reader at first sends ‘0’ to the tags. Then, the tags compare the received inquiry bit to the pointed bit. As mentioned earlier, the pointed bit initially points to the highest bit. The highest bit for both the tags is ‘0’ and this matches with the inquiry bit. When the inquiry bit is as same as the pointed bit, it will trigger the state machine to switch to the ‘send’ state. Thus, both the tags will switch to the ‘send’ state. When ‘send’ is on, it will clock the shift point register and the shift point register shifts to the second highest bit. Therefore, both tags send second highest bit as answering bit to the reader. Reader will then detect a collision. When a collision occurs, the reader will always send ‘0’ as next inquiring bit. This is same with the simulated tag and is different from the ‘0100’ tag. Thus the ‘0100’ will switch to the state ‘standby’ and will not answer to the inquiry of the reader. On the other hand, the simulated tag will switch to ‘send’ state and sends the third highest bit (‘1’) to the reader. The reader will send back the received bit to the simulated tag. This is same with the simulated tag’s pointed bit for sure since only the simulated tag will now answer to the reader’s inquiry. The simulated tag switches to the ‘send’ state and the pointer now shifts to the lowest bit. As mentioned earlier, when the pointer points to the lowest bit the tag will be identified and now it will switch to the ‘quiet’ state until all tags in the interrogation zone have been identified. When all

the tags have been identified, the reader will send a signal to all tags, so that all tags switch to ‘standby’ state and prepare to answer the next round of inquiry.

### 3.3 Case 4

Table 3: The inquiry process of case 4

Reader sends	Reader Receives	Tag answers		Identified tag (remaining tags reset to standby)
		0001	0010 (simulated tag)	
start				
0	0	0	0	
0	*	0	1	
0	1	1	standby	0001
start				
0	0	quiet	0	
0	1		1	
1	0		0	0010

\*denotes collision

This test case 4 assumes 2 tags in the interrogation zone, the first one with ID 0001 and the other simulated tag with ID 0010. Table 3 illustrates the inquiry process of case 4. Note ‘1’ in the figure 8 means that the ‘send’ state is on and the tag is sending ID bit to the reader. Comparing the simulated results with the actual inquiry process in table 3, the simulated result is found to be matching with the actual result.

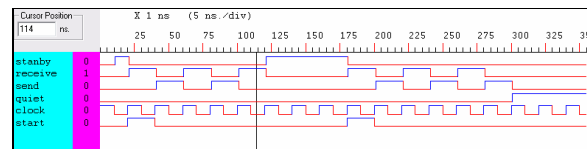


Fig.8 Results for case 4

As shown in table 3, the reader at first sends ‘0’ to the tags. Then the tags compare the received inquiry bit to the pointed bit. As mentioned earlier, the pointed bit initially points to the highest bit. The highest bit for both the tags is ‘0’ and this matches with the inquiry bit. When the inquiry bit is as same as the pointed bit, it will trigger the state machine to switch to the ‘send’ state. Thus both the tags will switch to the ‘send’ state. When ‘send’ is on, it will clock the shift point register and the shift point register shifts to the second highest bit. Therefore, both the tags will send second highest bit as answering bit to the reader. The second highest bit for both the tags is ‘0’, so the reader will send back the ‘0’ as next inquiry bit. The third highest bit is

different for both the tags. Reader will therefore detect a collision. When a collision occurs, the reader will always send '0' as next inquiry bit. This is same with the '0001' tag and different from the simulated tag. Thus the simulated tag will switch to the state 'standby' and will not answer the inquiry of the reader until one tag has been identified. Reader begins new round of inquiry by sending a 'start' signal. Now the simulated tag is the only tag that will answer to the reader's inquiry. The reader's inquiry and tag answering process repeats until the simulated tag gets to the lowest bit and has been identified.

#### 4 Conclusion

The Anti-collision protocol is an important factor in RFID systems to recognize the tag's ID. A circuit that can implement the anti-collision protocol has been developed based on Binary Tree Protocol. In Binary Tree Protocol, the tag ID with lowest bit has the priority to be identified first. Some obstacles such as wrong assumption of the state machine and ignorance about the timing problem for the input of the state machine have been detected and finally overcome. However, all such constraints finally led to a complete circuit that can successfully overcome the anti-collision problem in RFID systems. RFID has growing interest in auto-identification system and is predicted to replace conventional barcode system as it possesses a lot advantages compared to other auto-identification systems. Since the simulation done here involves only tags, scopes exist in developing anti-collision protocol for readers as well.

#### References:

- [1] Feng Zhou, Chunhong. Chen, Dawei Jin, Chenling Huang, and Hao Min, Evaluating and optimizing power consumption of anti-collision protocols for applications in RFID systems. *Proc. Int'l Symp. Low Power Electronics and Design*, 2004, pp. 357-362.
- [2] Feng Zhou, Chunhong. Chen, Dawei Jin, Chenling Huang, and Hao Min, Optimizing the Power Consumption of Passive Electronic Tags for Anti-collision Schemes, *In Proceedings of the 5th ASICON 2003*, pp. 1213-1217.
- [3] Finkenzeller Klaus, RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification, 2<sup>nd</sup> Ed. England: John Wiley & Sons Ltd. 2003.

[4] James Waldrop, Daniel W. Engel, Sanjay E. Sarma.. Colorwave, An Anti-collision Algorithm for the Reader Collision Problem, *Proc. IEEE Int'l Conf. Communications 2003*, pp. 1206-1210.

[5] Jeremy Landt, Shrouds of Time the history of RFID, *An AIM Publication*, 2001.

[6] Jihoon Myung, Wonjun Lee, Jaideep Srivastava, Timothy K. Shih, Tag-Splitting: Adaptive Collision Arbitration Protocols for RFID Tag Identification, *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS*, pp. 1-13.

[7] Leian Liu, Zhenhua Xie, Jingtian Xi, Shengli Lai, An Improved Anti-collision Algorithm in RFID System, *Mobile Technology, Applications and Systems 2nd International Conference on*, 2005.