

Analysis of Binary Anti-collision Algorithm Based RFID Position Technology

Jinhua Liu^{1,a}, Fuying Wu^{2,b}

¹*Modern Educational Technology Centre, Xinyu University, Xinyu City, Jiangxi, 338004, China*

²*Software School, JiangXi Normal University, NanChang , JiangXi, 330022 , China*

^a*liuxianya2003@163.com*, ^b*wfyesa@aliyun.com*

Abstract.

Radio Frequency Identification (RFID) is an automatic identification technology towards maturity in 1980s and a non-touching automatic identifying technology. In this system, collision is a frequent issue. The introduction of binary anti-collision algorithm makes it possible for dynamic adjustments of transmitted data length and value, so that collision issues can be properly solved along with identifying electronic tags through the number of collision bit “1”.

Keywords: Binary anti-collision algorithm; RFID Positioning System; Analysis

Introduction

Developing based on regressive-style binary algorithm [1][2], binary anti-collision algorithm improves the intelligence of RFID positioning system markedly. Compared with traditional ways, RFID has the advantage of being convenient and speedy, huge volume, good privacy, short reaction time and fine adaptability to environment etc. This thesis analyses positioning system in terms of conventions for algorithm, binary search algorithm and algorithmic

implementation.

1. Conventions for algorithm

To Manchester code, logic “0” represents high-low signal sequence with data “0” transmitting to “1”; logic “1” represents low-high signal sequence with data “1” transmitting to “0”. No- signal sequence indicates invalid data and identification failure. No-signal sequence happens when multi-tags return to different positions, each of their high-low signal and low-high signal offsetting mutually. Then collision issue is produced, which in order to find, reader has to search continuously. We use Manchester code to identify the collision bit accurately. Suppose EPC owns 8 codes and 2 tags in RFID positioning system, Figure 1 demonstrates the identified collisions based on Manchester code.

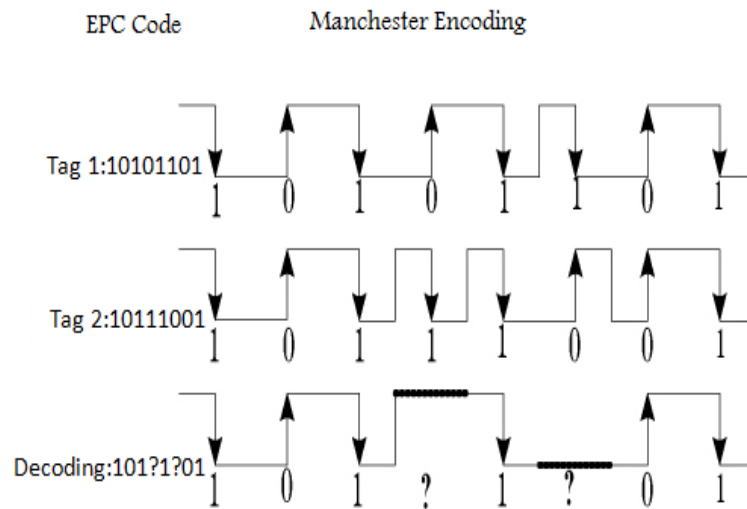


Figure 1: Manchester Encoding

To describe algorithm precisely, we introduce 4 instructions: 1. Request (DATA).2. Select (EPC)3. Read.4. Quite

2. Binary Search Algorithm

2.1 Mechanism

In the condition of no collision, it will execute the regressive-style search for reading tags orderly; in the condition of collision, it will execute forward search to jump over along with highest digit. There are 3 features : ① When regressive-style search executes, for confirming the requisite DATA parameter and executing the Request command preferably, it can plus 1 to each collision successively . ② When forward search executes, for reading tags properly, it can count on the number of “1” in the collision. ③ When the bit in instruction-bit is lower than that in conflict-bit, instruction-bit will not be sent, length of instruction will be regulated dynamically and then reduced.

2.2 Procedure

① Reader sends Request (1), then all tags in this range answer. ② .In the condition of collision, tags identification counts on the number of “1” of collision bit, then compare the bit of data with the highest level of collision zero. If higher, keep the same, if lower, elide and get required data parameter for the next Request instruction; in the condition of no collision ,it can identify tags directly. ③ After tags being identified, use “Select” to activate identified tags and read by “Read” instruction. ④ Tags are selected by “Quiet” instruction, then tags will be shielded. ⑤ In the condition of only one collision after tags being identified, it will take next steps to acquire DATA parameter: set this collision-bit as an origin point backward , add 1 to each collision until each one reaches 1. Then stop command Request and return to Request (1).

2.3 Implementation of algorithm

Suppose 8 tags within the reader scope, ID’s value is 8 bit. At the beginning, tags are unknown, Reader instructs “Request (1)” and tags answer. Picture 2 demonstrates the concrete process.

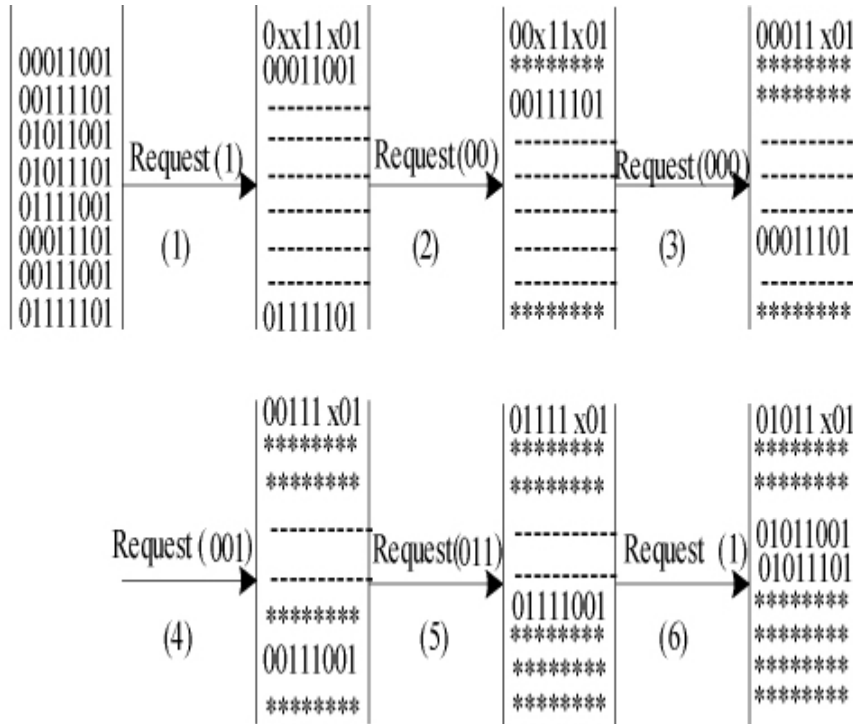


Figure 2: algorithm request procedure

In Figure 2, each line with “-----”, “*****” and numbers or letters stands for tags of quiet state, tags of no response and normal tags respectively.

From Figure 2, 8 tags can be identified after 6 operations of executing Request(1) instruction and the query will be completed. Comparing to the regressive-style algorithm, it reduces 9 operations and improves efficiency significantly.

The specific process is as follows:

① Reader sends an instruction “Request(1)” to eight tags, tags receive instruction and feed back with data “0xx11x01”, in which we see collisions happen to the first, second and fifth data. So the tags whose number of collision “1” is 0 or 3 are identified directly. They are shielded tags 01111001 and 000011001.

② Instruct the first data to be 0 and reader sends “Request(00)” to tags, tags with 00 as the top two numbers receive the instruction and feed back with date 00x11x01, in which we see collisions happen to the second and fifth data. So the

tags whose number of collision bit "1" is 0 or 2 are identified directly. Since tags whose collision bit number is "0" have been identified and shielded in step 1, what are identified and shielded now is the tag whose collision bit "1" is 2, namely 00111101.

③ Instruct the second data to be 0 and reader sends "Request (000)" to tags, tags with 000 as the top three numbers receive the instruction and feed back with data 00011x01. So the tags whose number of collision bit "1" is 0 or 1 are identified directly. Since tags whose collision bit number is "0" have been identified and shielded in step 1, what is identified and shielded now is the tag whose collision bit "1" is 1, namely 00111001.

④ After the steps above, there is only one collision bit in re-identifying tags. The instruction should be sent, according to the third paragraph of section 2.2 is Request (001), which makes tags with 001 as top 3 numbers feed back with data 00111x01. Then the tags whose collision bit "1" is 0 or 1 are identified directly. Since the tags whose collision bit number "1" is 1 has been identified and shielded in step 2, what are identified and shielded now are tags whose collision bit "1" is 1, namely 00111001.

⑤ In process of this identifying, there is only one collision-bit. According to the third paragraph of section 2.2, the instruction should be sent now is Request (011), tags with 011 as the three top numbers receive the instruction and feed back with data 01111x01. Then tags whose collision bit "1" is 0 or 1 are identified. Since tags whose collision bit number "1" is 1 has been identified and shielded in step 1, what are identified and shielded now are tags whose collision bit "1" is 0, namely 01111001.

⑥ In process of this identifying, all of collision-bit is 1. Reader sends Request(1), which is equal to returning step 1, all the tags receive and feed back with data 01011x01. So tags whose collision bit "1" is 0 or 1 are identified and shielded, namely 01011101 and 01011001.

Conclusion

In RFID positioning system, the point of issue is whether or not tags can be

identified quickly and accurately. Binary search algorithm in this thesis can not only identify the tags by collision bit “1”, but also adjust the transmitted data length and value dynamically, which is dramatically time-saving for the reader to query tags.

References

- [1].Gao Jinhui, Zhen Xiaoyan, Binary search in RFID improved algorithm of collision(J),Computer measurement and control,2012,20(10):2754-2756.
- [2].Wu Yueqian, Gu Daguang, FanZhenyue.etc. The analysis and comparison of the improved algorithm of anti-collision algorithm in RFID system (J), Computer engineering and Application, 2009, 45(3):210-213.
- [3].Jiang Lifen, Lu Guizhang, Xin Yunwei.Research on anti-collision algorithm in RFID [J].Computer Engineering and Applications, 2007, 43(15), 29-32.
- [4].Li Jin, Research on Anti-collision Algorithm for Wireless RFID System[D].Master’s Degree, Beijing Jiaotong University,2007
- [5] Gao Le, Research on Anti-collision Algorithm of RFID [D].Master’s Degree, University of Electronic Science and Technology of China,2006
- [6]. Xie Zhenhua, Lai Shengli, Chen Peng, RFID Technique and Anti-collision Algorithm, [J].Computer Engineering and Applications, 200743(6):223-225.
- [7].Jianwei Wang,Dong Wang,Yuping Zhao, Timo. Korhonen. Fast Anti-Collision Algorithms in RFID Systems[J]. International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM'07) , 2007