I. Troch, F. Breitenecker, eds.   ISBN 978-3-901608-35-3

Managing data in an RFID application

M. Trebar
University of Ljubljana, Ljubljana, Slovenia

Corresponding author: M. Trebar, University of Ljubljana, Faculty of Computer and Information Science
Trzaska cesta 25, 1000 Ljubljana, Slovenia; mira.trebar@fri.uni-lj.si

Abstract. A radio frequency identification (RFID) application performs an inventory of food products which can be used to predict the lunch menu based on the recipes saved in a data base. The information about food products is saved in a custom part of the RFID tag's memory. At present, no individual food products have attached RFID tags and no encoding formats for custom data exist. Because of that, the companies would be able to define their own encoding formats.

To design and to implement an RFID system dealing with the identification of food products to perform inventory we needed to define our own data encoding model. Two different encoding formats were analysed. The first one, ASCII encoding is very straightforward, is simpler to represent the information but at the same time it can be very complex which can enable overload of data in the transmission process between the reader and RFID tags attached to the products. The second analysed is BCD encoding which requires far less memory space, the transmission of data demands less time but additional encoding tables are needed to obtain the information. In the comparison we presented the benefit of having a smaller RFID tag's memory and the feasibility to organize the fields of data into the same block when using BCD encoding. It also enables faster communication between a reader and RFID tags due to less data that are transmitted.

1 Introduction

Radio frequency identification (RFID) is playing an essential role with systems deployments in a number of industries such as pharmaceutical, health care, transportation, logistics, retail and many others [1]. In the future, it can be expected, that many retailers, also in food production, will use RFID systems to track their products to the point of sale. Many home appliance producers tend to build RFID systems in their products, where the fridge is the one that will use the possibility of food identification. The information from each individual product would be possible to use in many kinds of data analysis when it moves through different locations. It becomes also very important in food industry to improve customer service and to bring benefits of reduced exposure to safety risks [4]. Additionally, a traceability system would define the types of data that should be collected and transferred to each state of the supply chain management and also to interested customers.

Actually, many people think that RFID will replace bar code as predominating automatic identification technology [2]. But it is most likely that they will co-exist in various fields of their use because each technology has its own characteristics with advantages and weaknesses. Although current use of bar coding is highly optimized by using wireless communication to a computer, the process demands extensive human intervention when reading each item which is time consuming and is also error prone. Using RFID technology the human intervention is mostly eliminated, no 'line of sight' reading is needed and large number of items is simultaneously identified.Over the past few years, many suppliers already attached RFID tags on the pallets and the cases of goods they shipped to the distribution centres of the major organizations. Many companies become aware of true potential of the technology, especially in the areas of logistic, item tracking, building access control systems and many other applications. In supply chain management it is applied with the intention of building a warehouse management database to study, evaluate and improve the company's logistic system.

The most important component of the RFID system that is composed of readers and tags is data stored in the tag's memory. Several times per second, readers broadcast a signal and receive a response from tags that are in the reading range. During this communication a list of tags is created and sent to the computer attached to the RFID system [2]. After that, the collected RFID data needs to be converted into valuable information [5] by the decoding model which is a subject to the selection of RFID tags, data encoding formats and custom data options to support the application. RFID tags are encoded with an Electronic Product Code (EPC) which is a globally unique identifier of tagged object. They mostly use a 96-bit EPC with a number of different standardised encoding formats that depend on the tagged object. As an additional option, RFID tags also include custom data that can be added with individually defined RFID tag's data encoding formats. The decision of which tags will be used in the application, encoding tag data requires the analysis of how much memory is available to write custom data, how to organize data to simplify the process of reading/writing RFID tags which utilizes the read rate and consecutive the identification rate of the reader.

2 Food products identification

In this section we will introduce essential characteristics of an RFID system which is used to perform a prediction of the menu based on inventory of food products and recipes saved in a database. The application is going to be tested in a real RFID system using a high frequency RFID reader and RFID tags of type I-CODE SLI Smart label IC. Until now, we haven't found a real RFID tags attached to individual food products. To implement our application, we needed to define: (i) what information about the products we want to use as custom data, (ii) the best encoding format of RFID tags attached to our simulated food products. All the information about the food products is stored in the custom part of RFID tags memory.

2.1 RFID system architecture

The basic RFID system consists of RFID tags, an RFID reader and a computer running application software [2]. The reader identifies RFID tags attached to objects and transmits the captured information to the computer where it is decoded and used for further processing. RFID tags consist of a small digital device for receiving and transmitting the signals. Taking into account a memory function of a microchip, they enable read-write operations. Based on the power supply there exist two main classes of RFID tags: active and passive. In contrast to active RFID tags with their own power source passive RFID tags are without it's own power source generally smaller, lighter and less expensive, maintenance free and will last for years. They are activated only within the response range of RFID readers and will be used in our application. Passive RFID tags are mainly used and can be found in different shapes, like: tokens, labels, buttons, animal-tags, cards and others. An RFID reader is a digital device which transmits data, performs read and write functions, powers a passive RFID tag. It can have added three more critical functions: implementation of an anti-collision algorithm, authentication of RFID tags and data encryption. The anti-collision algorithm permits simultaneous reading of large numbers of tagged objects, while ensuring that each RFID tag is read only once. Authentication and data encryption are used to ensure a secure transmission of data to an authorized user. The most important considerations in building an RFID system are: (i) the frequency bands divided into: Low Frequency (LF: 125-134kHz), High Frequency (HF: 13,56 MHz), Ultra High Frequency (UHF: 868 - 928MHz), and microwave frequency (2,45 GHz). With the frequency of an RFID system data the read range is defined; (ii) RFID standards - a radio communication technology is the subject of governmental regulations to establish order on the airwaves, best practices, safety and maximum permissible interference guidelines. In industry are the most important ISO (International Organization for Standardization) and IEC (International Electro-technical Commission) standards, and in supply chain management applications are EPC global standards.

2.2 ISO 15693 RFID tags memory

The most important component in each RFID system is the data obtained from RFID tags memory in the reading process. A widely used and accepted standard that only pertains to high frequency RFID systems is ISO 15693 which is used in our RFID tags defining food product examples with I-CODE SLI Smart label IC [3]. Producers of RFID tags offer various tags memory length from 128 bits to 4 Kbytes. The memory is divided into blocks as the smallest access units in read/write process. Each block consists of 4 bytes (1byte is 8 bits) where bit 0 represents the least significant bit (LSB) and bit 7 the most significant bit (MSB), respectively. Complete RFID tag memory is divided into the next three parts with some subfields having special use:

- Unique Identifier (UID) consists of 2 blocks and is programmed during the production process which cannot be changed afterwards. It consists of: UID 7–fixed value: "E0", UID 6–manufacturer code; UID 5–tags type, UID 4 - UID 0–IC manufacturer serial number;
- Special functions use the next 2 blocks: Electronic Article Surveillance mode (EAS), Application Family Identifier (AFI), Data Storage Format Identifier (DSFID), write access conditions;
- The rest is custom data where read and write commands are allowed to define special applications.

3 Custom data model in RFID application

RFID tags attached to food products contain custom data that give the information about food product type and name, price of the product, type of measures used for food, expiration date, and many other. The information will be saved in the tag's memory that is considered as a matrix of encoded bytes organized in blocks which are transfered in RFID system during the read or write cycle. Readers normally return the binary or hexadecimal representation of data. For the UID part of data a special encoding format that depends on the particular RFID tag is used which is not the case for custom data of food products in our application. In the data model we analyse two encoding formats to show their impact on the tag's size of memory and also on the communication rate in RFID system.

3.1 Food products data encoding

At the beginning, we defined data fields that describe food products which are presented and described in 'Table 1'. All defined fields are used in our application. The number of fields is not directly limited, they can be added for

Data field	Description
Food product type	Milk product, vegetables, fruits, ...
Food product name	Whole milk, carrot, orange, ...
Price	Value
Currency	Euro, Dollar, Pound
Production date	ddmmyyyy (day, month, year)
Expiration date	ddmmyyyy (day, month, year)
Measure value	Value
Measure type	Weight: gram, kg, ...
	Volume: litre, mililitre, ...
	Number of pieces

Table 1: RFID data fields that describe food products. All eight data fields are defined and described with some examples that are used in the application.

Data field	ASCII encoding		BCD encoding	
	Number of characters	Number of blocks	Number range	Number of blocks
Food product type	T	T/4	0-9999	1/2
Food product name	N	N/4	0-99999999	1
Price	P	P/4	0-999999	3/4
Currency	C	C/4	0-99	1/4
Production date	10	3	0-99 0-99 0-9999	1
Expiration date	10	3	0-99 0-99 0-9999	1
Measure value	V	V/4	0-9999	3/4
Measure type	M	M/4	0-99	1/4

Table 2: Encoding formats for each food product with eight data fields: ASCII encoding is presented with number of characters (1st column) and with the number of blocks (2nd column) used in RFID tag; BCD encoding is presented with the range of numbers which define the maximum number of items in each field (3rd column) and also with the number of blocks (4th column).

use in different applications as long as it would be possible to choose that encoding format which will fit all data into the tag's memory.

The length of data fields differs and is defined with the encoded method according to the application demands defined in advance. It would be recommended that individual fields are within one block which simplifies the decoding of blocks read subsequently from the reader and used in further computing. All fields are presented with the number of blocks in 'Table 2'. In ASCII encoding one byte is used to define each character of the field description which exactly defines the number of bytes for production and expiration date. For all other fields, we define the maximum length of each field with variables (T, N, P, C, V, M) that will be set later in the application when the comparison of encoding formats is explained. In BCD encoding we need to use the encoding tables for following fields: Product type, Product name, Currency and Measure type. For other fields BCD numbers are directly defined values used in the application.

BCD codes for four data fields are shown in 'Table 3': (i) the first one, Product type uses only first half of first block (2 bytes); (ii) the second one, Milk product uses a whole block; (iii) the next one, Currency uses only one byte in the same block with the Price, and (iv) the last one, Measure type defines the amount of food product in the same block with Measure value.

Food product type	BCD (2 bytes)	Milk product	BCD (4 bytes)
Milk product	0061	Whole milk	02301192
Vegetables	0101	Cheese	02500015
Fruits	0150	Yogurt	02215002

Currency	BCD (1 byte)	Measure type	BCD (1 byte)
Euro	01	litre	01
US-Dollar	12	mililitre	04
Pound	21	gram	11

Table 3: Three examples of food products in BCD code: Product type is defined with two bytes in one block, Milk product is defined with four bytes, Currency and Measure type use one byte.

3.2 Comparison of ASCII and BCD encoding

To explain the importance of the proposed BCD encoding format used in the application we present here an example of two RFID tags for a whole milk product shown in 'Table 4'. For each field in ASCII encoding the first column shows the information about the product and the second column presents the number of bytes with the ASCII code of characters in each field. This is only an example with relatively small number of bytes which is not suitable for many other products with larger number of characters. This leads to the predefined maximum number of characters for each field. It is important that this number of bytes is divisible by 4 which characterizes the number of blocks for each field.

Data field	ASCII encoding		BCD encoding	
	ASCII record	Number of bytes	BCD record	Number of bytes
Food product type	Milk product	T=12	0061xxxx	4
Food product name	Whole milk	N=12	02301192	4
Price	0,78	P=4	000078	3
Currency	Euro	C=4	01	1
Production date	12-11-2008	12	12112008	4
Expiration date	23-01-2009	12	23012009	4
Measure value	1000	V=4	001000	3
Measure type	mililitre	M=12	04	1

Table 4: An example of RFID tags that describe whole milk in ASCII and BCD encoding formats. Number of characters defined in variables T, N, P, C, V, M is adapted to this application, normally they could be longer. Number of bytes in BCD encoding is fixed.

From the described ASCII record it is easy to recognize the food product with attached RFID tag which is not the case in BCD encoding. 'Table 5' presents raw hexadecimal data record of both encoding formats where it is obvious that BCD encoding uses much less memory of RFID tags.

Data format	UID	Custom data
ASCII	E0040100 158C0C12	4D696C6B 2070726F 64756374 57686F6C 65206D69 6C6Bxxxx 302C3738 4555524F 31322D31 312D3230 3038xxxx 32332D30 312D3230 3039xxxx 31303030 6D696C69 6C697472 65xxxxxx
BCD	E0040100 158C0BD0	0061xxxx 02301192 00007801 12112008 23012009 00100004

Table 5: RFID data define the whole milk product in ASCII and BCD encoding. Each UID consists of two blocks and is defined with RFID tag's encoding format (2^{nd} column). ASCII encoding format uses 18 blocks of custom data, and BCD encoding uses only 6 blocks of custom data. Value x represents a part of field not used in encoding.

4 Conclusion

In this paper we identify the problem of encoding of custom data in RFID application. ASCII and BCD encoding models were analysed in order to describe food products where the information was divided into eight data fields. Both encoding models were compared and we used BCD encoding with much smaller number of custom data blocks in our RFID application. Afterwards, several food products were chosen, encoding tables were defined, the database of the products was prepared and data was saved to RFID tags. The application runs on the experimental RFID system where we analyse the benefits of the proposed encoding format. The decoding of information based on our model of BCD encoding was made on the personal computer before it was shown to the user on his computer. The rest of I-CODE SLI Smart label IC with 1024 bits EEPROM memory which is 22 blocks can be used for additional data fields in other applications or we can use RFID tags with less memory. The results clearly indicate the possibility of using RFID tags with smaller amount of memory to store the food products information when using BCD encoding. The encoding can improve readers identification and read rate which are very important in real applications.

5 References

[1] Ahson S., and Ilyas M.: *A RFID Handbook Applications, Technology, Security, and Privacy*. CRC Press, New York, 2008.

[2] Brown E. D.: *RFID Implementation*. McGraw-Hill, New York, 2007.

[3] *I CODE SLI, Smart Label IC SL2 ICS20*. Data Sheet, Product Specification, Revision 3 (2003).

[4] Jones P., Clarce-Hill C., Comfort D., Hillier D., and Shears P.: *Radio frequency identification and food retailing in the UK*. British food journal, 107(6) (2005), 356–360.

[5] Teskey M.: *Turning RFID Data into Information*. http://www.devx.com/enterprise/Article/32251/1954, (2008).