AC 2011-1603: ANIMAL RECORD MANAGEMENT USING AN EMBEDDED RFID-BASED SYSTEM

Mr. David Bowker

David Bowker is currently working as an Embedded Application Engineer at a consumer electronics company in Nashville, TN. He received his BS degree in Computer Engineering Technology and MS degree in Engineering Technology from Middle Tennessee State University.

Saleh M. Sbenaty, Middle Tennessee State University

Dr. Sbenaty is currently a Professor of Engineering Technology at Middle Tennessee State University. He received the BS degree in EE from Damascus University and the MS and Ph.D. degrees in EE from Tennessee Technological University. He is actively engaged in curriculum development for technological education. He has written and co-authored several industry-based case studies. He is also conducting research in the area of mass spectrometry, power electronics, lasers, and instrumentation.

Mark Newton Johnson, Equipass ID

Mark Johnson is the founder and President of Equipass ID a sub-corporation of Newton RFID. He received a B.S. in Animal Science and Equine Husbandry from Middle Tennessee State University in 1992 and has spent a life time in the Horse Industry. Having co-founded two prior technology companies, Mr. Johnson's professional career includes performing the first micro- payment trials in a closed loop application utilizing RFID wristband technology in 2004 and later being the first company to offer RFID micro-payment solutions in a professional sports venue. His background in RFID technology combined with a passion for the horse industry lead to the creation and development of a read/write record management system for the animal industry. Mr. Johnson continues to develop complimentary technology solutions utilizing the collaborative talents of private industry, educational institutions and government entities.

Animal Record Management Using an Embedded RFID-Based System

Abstract

The current paper describes the design and implementation of a radio frequency identification (RFID) system that uses an embedded microchip in conjunction with an RFID reader for the purpose of storing and retrieving pertinent animal information. The microchip is implanted into an animal, with the initial focus on equine applications. It will be demonstrated that further developments for using such a system in other animals is possible. While the use of RFID technology for animal identification is not necessarily new, the current application focuses on using an RFID with read/write capability. This brings the functionality of storing and retrieving information directly onto the RFID microchip, which can be updated whenever necessary. In addition, the larger memory capability of the current system allows the storage of important medical information directly onto the RFID chip; hence, allowing a veterinarian the immediate access to vital information in case of emergencies. Typically, an RFID holds only an identification number that must be referenced to a database on a remote server in order to retrieve the animal record. The current system, however, allows for the most important information to be stored directly onto the microchip such that it can be read instantly without the need for any other referenced information. A handheld device is used to read and write the information to the microchip in a field or office setting. This is very useful when time is critically important or when the animal is in a remote location such that retrieving information from an internet connected database is not possible. The work has been the result of a close collaboration between a local industry (Passage Inc/NIS Franklin, TN) and the Engineering Technology and Agribusiness Agriscience Departments at Middle Tennessee State University.

Background

Animal records management is a considerable problem with no simple solution. Animals require paperwork, tests, and documentation on a regular basis, especially when traveling across state lines. To comply with local, state, and federal laws, there are various different medical tests that are required in order to show that the animal has been tested and vaccinated against any number of diseases. The owner must show a proof of ownership of the animal and that the animal is indeed that one which is reflected on the paperwork. When focusing on the equine industry, many of the current measures fall short and become cumbersome. Tags and branding are often out of the question as some of these animals are used for show purposes and cannot have such measures in place. RFID tags have been used to identify animals, however, they have an enormous shortcoming. Current RFID tagging methods only provide a read-only identification number that must be cross referenced to a database to obtain any pertinent information. This becomes quite cumbersome and slow especially in situations where time is a factor or the database containing the information is not readily accessible, such as a remote area or a farm where internet access is not available.

Our method has the capability of 2048 bit (or 2kb) read/write storage, which can provide the needed data to anyone who wants to access information about the animal. All types of pertinent information can be stored onto this chip including: owner and animal life information such as age, breed, and sex as well as health information including vaccinations, medications, surgeries, or diseases veterinary information.

All this becomes not only a quick and efficient way to store and retrieve data, but since it is stored on a chip that is placed inside the animal, questions about its authenticity also are taken away, which eliminates cases of fraud and counterfeiting. Since the storage capability is still somewhat limited, there will still be a source for less important information stored on an internet accessible database that can be accessed using the animals ID code as a key.

Technical Information

The RFID chip that is used in this study is a HITAG S2048 Transponder IC. For the purpose of testing the system, a Windows Mobile based PDA with an ACG LF MultiTag R/W Module are used. The ACG Read/Write Module allows the PDA to communicate with the RFID tag to store and retrieve information. The module connects to the Compact Flash port of the PDA to communicate with the custom software designed specifically for this purpose. The software is written in C# and provides an easy-to-use graphical user interface (GUI) to interact with the system. A single button press allows the user to read information from the chip and a simple form interface for filling in any required information. The interface is divided into "pages" to reflect the different areas of information stored onto the device.

The ACG R/W module provides a dynamic link library (DLL) that contains standard methods used to make calls to the device in order to perform a number of functions such as reading and writing to the RFID microchip. These methods are specific to the ACG module, but are standard calls that can easily be adapted to any read/write device.

RFID Tag

The RFID Tag device that is being used for the current implementation of the system is the HITAG S2048 Transponder IC. A block diagram of the chip is shown in Figure 1.

The memory of the RFID tag has a total capacity of 2048 bits divided into 16 pages of 4 blocks as shown in Figures 2 and 3. Each block consists of 4 bytes, they are addressed from 00h to 3Fh. Block 00h is a read only block that holds the device serial number. Block 01h is used for configuration of the device and the remaining blocks (02h - 3Fh) are designated for user data. So there are effectively 1984 bits available to store data. This limitation means that the programmer must be conservative and creative in how data is stored onto the device. Plain text is the most expensive type of data stored in the current configuration as it leaves little room for allowing the software to compact the data or translate it into a less storage intensive form. Figure 3 below shows the details of how data is currently stored in memory locations onto the device. It should be noted that these items, locations, and sizes are the preliminary criteria created for the demo system and have already changed a few times and will definitely change in the Future.

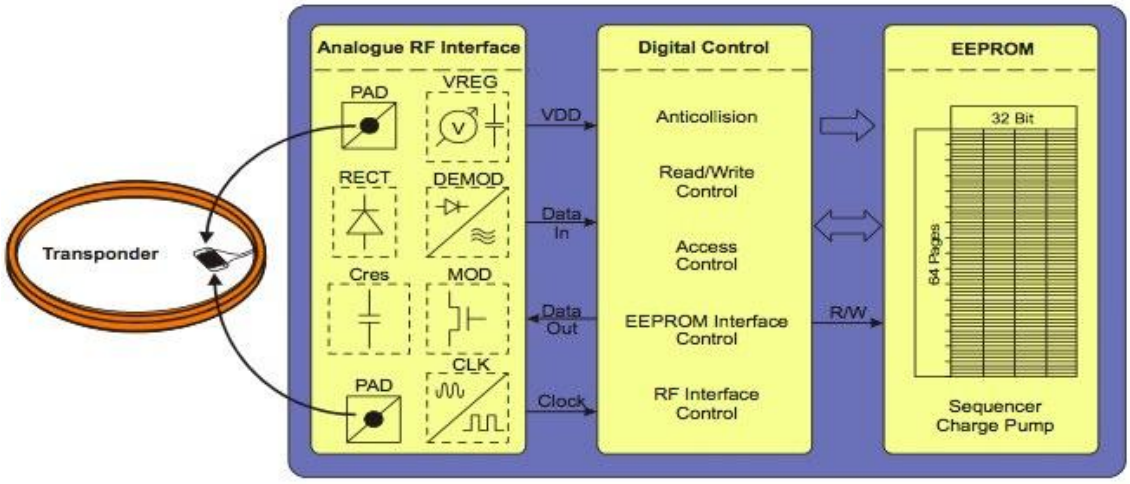

Figure 1 - RFID Block Diagram

BLOCK 0	0x00				
	0x01				
	0x02				
	0x03				
BLOCK 1	0x04	First Name			
	0x05				
	0x06				
	0x07				
BLOCK 2	0x08	Last Name			
	0x09				
	0x0A				
	0x0B				
BLOCK 3	0x0C	Address 1			
	0x0D				
	0x0E				
	0x0F				
BLOCK 4	0x10	Address 2			
	0x11				
	0x12				
	0x13				
BLOCK 5	0x14	City			
	0x15				
	0x16				
	0x17				
BLOCK 6	0x18	Zip			
	0x19				
	0x1A	State		Country	
	0x1B	Title			
BLOCK 7	0x1C	USEF Registration #			
	0x1D				
	0x1E				
	0x1F				
BLOCK 8	0x20	Breed/Discipline		Amateur/Pro	Member Status

Figure 2 - Details of Memory Storage

Block	Function	Access
00h	Serial number	R/O
01h	Configuration word	Read/Write
02h – 3Fh	User data	R/W

Figure 3 - RFID Tag Memory Overview

This will be in order to accommodate items needed for animal information, certified veterinary inspection reports, and other medical documentation. Currently, discussions are being made with the Tennessee Department of Agriculture and the United States Equestrian Federation in an attempt to determine what they would like to see in the release version of the software. This will help increase the chances of their buy-into the final product.

Read/Write Module

Figure 4 - ACG Read/Write Module

The software can communicate with the Reader Module (Fig. 4) using two different protocols: ASCII or Binary. For this design, the ASCII protocol was selected for its straightforward simplicity and the fact that it does not require a device driver for the Reader Module. Using the ASCII protocol allows for simple method calls and easy debugging. The methods can even be called from a terminal program to allow for quick testing of functionality before implementation. This is because it uses standard serial communication to the Compact Flash port of the device when using this protocol.

The read range for the ACG R/W Module is limited to 3-4 inches. This should be sufficient for reading the chip once it is implanted into the animal as the location of the implant is close to the

surface of the skin. It is possible though to obtain a larger read range with different reader devices. For future implementation of the system, it is probably best to move away from the ACG R/W Module because of its Compact Flash interface that is not available on many newer, more modern devices. The possibilities of using a Bluetooth read/write module are being considered such that it can be interfaced with any Bluetooth capable device. This would permit future implementations of the system to have software written for multiple platforms to allow for more widespread acceptance and usage.

Software

The software is written in the C# programming language and is intended for the Windows Mobile platform. This language was chosen for its versatility to be adapted to any Windows platform without requiring too many changes to the source code. All the languages supported for the ACG Read/Write Module are for the Windows platform and C# provided the desired power, ease of implementation, and versatility.

There are several types of data being stored to the RFID microchip, as it was shown in Figure 2. Some is plain text and some are pre-defined numeric representations of other data. The items that can be enumerated in this manner, such that they can be easily represented by a number value, are the best in terms of storage efficiency. For instance, there are nineteen breeds that can be selected for the “Association Information:”

1. "Andalusian/Lusitano"
2. "Arabian/half-Arabian/Anglo-Arabian/National Show horse"
3. "Connemara"
4. "Dressage"
5. "Driving"
6. "Endurance"
7. "English Pleasure"
8. "Enveting"
9. "Friesian"
10."Hunter"
11."Jumper"
12."Morgan"
13."ParaEquestrian"
14."Paso Fino"
15."Reining"
16."Saddlebred/Hackney/Roadster/Shetland/Parade"
17."Vaulting"
18."Welsh"
19."Western"

In the software, these are enumerated in a list of strings and simply represented by a number. Therefore, instead of requiring several bytes of data to save the actual text of the breed itself, only 2 bytes of storage space is reserved for the number associated with that breed. This permits two different breeds to be selected, per the association standards and allows for up to a total of 255

different breed types if necessary. The location of 00h is reserved for no breed being selected.

Since memory space is at a premium, it will likely be necessary in future implementations to get even more creative with the storage of data as more information is desired to be saved onto the chip. It is possible then to use the example above as a demonstration of how even more space could be saved. Since there are only 19 different breeds that are currently being used and will most likely never approach the limit of 255, it would be possible to split a byte into two in order to represent the two breed types, primary and secondary. This would reduce the maximum number of breeds possible to 126, still far more than would likely be needed. The method for implementing such an approach would work by splitting the storage byte as follows:

The upper half of the byte could represent the primary breed and the lower half the secondary breed. Then it would only be necessary to mask out or ignore the data that is not currently relevant. This approach could be very helpful in creating more free space as it becomes necessary.

Figure 5 - Software Main Page Interface

Currently the software has several interface screens to view and edit different information:

1. **Main Page**: As shown in Figure 5, it has buttons to access the different areas of information that is stored onto the chip. Currently only the Owner Info and Member Info areas are implemented. This is because the criteria to properly implement Animal Info, Medical History, Vet Records, and CVI Reports for the partners have not been obtained yet. Since the current system serves as a proof of concept and since inputs and buy-ins from different organizations are being sought, it is believed that the remaining areas will be tailored to the needs and wants of such organizations. This screen also has the buttons that initiate the Read and Write functions to the RFID chip. There is also a button that will clear any data from the system that has been entered or read from the RFID chip.

2. **Owner Info**: Accessed by clicking the "Owner Info" button from the main screen or by selecting "Owner Info" from the "Window" menu. This page displays information about the owner of the animal.

3. **Member Info:** Accessed by clicking the "Member Info" button from the main screen or by selecting "Member Info" from the "Window" menu. This page displays the association membership information for the animal.

4. **Init Info:** Accessed by selecting "Init Info" from the "Diag" menu. This shows the information from the Compact Flash R/W Module collected when it is initialized at startup.

5. **R/W:** Accessed by selecting "R/W" from the "Diag" menu. This screen will permit the user to read or write a single byte of data anywhere onto the chip, which is useful for diagnostic purposes. This feature is not intended to be included in the final consumer release.

6. **Read All:** Accessed by selecting "Read All" from the "Diag" menu. This allows the user to read all the data from the RFID chip and then display the data in the textbox below, which is useful for diagnostic purposes. This feature is not intended to be included in the final consumer release.

Conclusions

While the current system is not ready to be marketed and deployed to horse owners, it does serve as a proof of concept and demonstrates that the system will work and would be ready for a test environment. From the above, it has been shown that it is possible to reliably store a large amount of data onto a small implantable RFID chip, where information about the animal can be stored and updated as necessary to reflect any information that may have changed. This will be a great benefit to everyone that may be involved in the animal's life, whether it be the owner, veterinarian, or officials at a show or competition.