

# Reading data of sensor nodes belonging to an integrated RFID-sensor network with an RFID handheld reader\*

Sándor Király, Tibor Radványi, Csaba Biró

Eszterhazy Karoly University of Applied Sciences  
ksanyi@aries.ektf.hu, dream@aries.ektf.hu, birocs@aries.ektf.hu

## Abstract

The RF mesh network, deployed with IQRF mesh network devices, works in the 868 and 916 MHz frequency band and it allows the operation of 65000 devices per network. Our own nodes, named exNode, have been developed and in them the IQRF unit is responsible for the communication. This is complemented with a sensor-line or more precisely an UHF RFID interface that is intended to possess a self-designed protocol that enables the nodes to be located by an UHF RFID reader. If some segments of the sensor network are still connected it is possible to read the measurement data of the sub-graph nodes through only one node. In this article, it is expounded how a software for a handheld RFID reader that can change the settings of and read and interpret data of the sensor nodes even with different types of sensors, can be developed.

*Keywords:* sensor network, sensor, extended node, RFID handheld reader, sensor-line, RF mesh network, IQRF, PDA programming

*MSC:* 68Qxx, 68Uxx, 94Cxx

## 1. Introduction

This node is part of an RF Mesh Network where these kinds of devices can communicate with and send the stored data to each other. But what happens if they fail or changing the settings of exNodes is necessary? This is the reason why this exNode

---

\*Supported by: FutureRFID Development possibilities in the RFIDNFC technology TÁMOP-4.2.2.C-111KONV-2012-0014.

also has an RFID interface. To be precise an SL900 RFID transponder has been integrated into the board. What needs to be known about this extended node? It holds 8 physical slots where sensors can be inserted, that also means sensors must be assigned to slots. Sensors can detect changes in quantities that are stored in memory blocks of 8. That is why memory blocks must be assigned to slots. There are also 8 virtual slots for storing the data of sensors that can detect changes in two different quantities in parallel. The measured values are stored in memory blocks

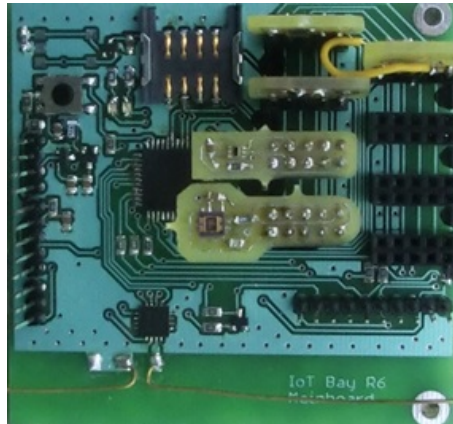


Figure 1: Extended Node (exNode)

that have a FIFO structure. Before saving data into the memory the following operations can be performed on the measured data: min, max, average or nothing, which means that the last measured value will be stored. Naturally, the UHF RFID transponder that has been integrated into this node has a particularly important role in getting the stored data and identifying the exNode.

## 2. Communication with the exNode

First and foremost its EPC code or a string in the TID memory can be used to identify the node. [1, 2, 3] Secondly, in order to get the stored data from the node it is necessary to send a command code to it. Commands must start with  $0xCC$  then the command code and its parameters must follow. For example:  $0xCC1001$  where 10 is the command code and 01 is the parameter. In this case 10 is the command code for getting stored data from the memory block of 01. The command must be written into the first part of the UserMemBank at address  $0x0$  so the  $CC$  will be stored there. Starting with the next address the command code must be written with its parameters as it is shown in Figure 2.

Then the node controller reads this command from the UserMemBank and always replies and writes the response to the UserMemBank memory at address

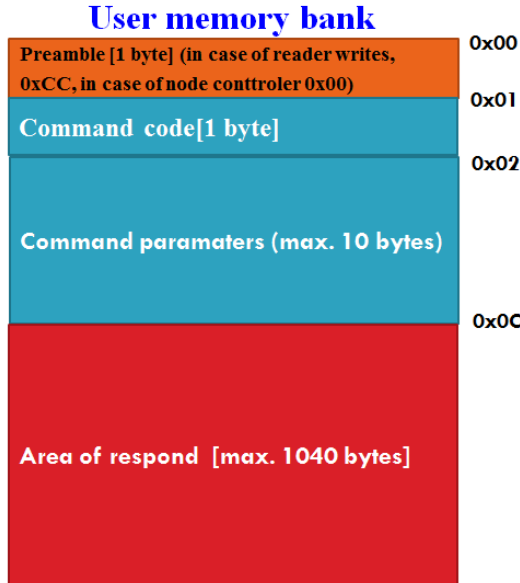


Figure 2: Communication area

0x0C. The first byte is always double zero then the command and its parameters, then the required data.

### 3. The RFID reader

In order to access the desired measure values stored in the node only an RFID reader with the proper software is necessary. For this it is required to write the command to the memory of the transponder and then we are able to read the resulting data. Development Environment:

- Visual Studio 2005 or 2008
- NET Compact Framework: Version 2.0 or later
- SDK (for ALH-9010/9011 only): Windows Mobile Professional SDK 6.5 or later
- For Win7-8: Windows Mobile Device Center

For controlling the RFID module there are a number of DLL files created by the manufacturer. This is very typical for any PDA, only calling API procedures and functions are necessary. In case of this PDA, the first step in getting data from the transponder is the instantiation of the RfidApi class that can be found in the

`NRfidApi.dll`. Then only the calling the proper methods of this class is needed. To activate the reader we call the `PowerOn` method, to open the communication channel we call the `Open` method. The most useful methods for us (apart from the mentioned) are the `ReadMemBank` and `WriteMemBank` methods.

```
RFID_RESULT ReadMemBank(bool bSyncMode, MEM_BANK MemBank,
UINT nWordPtr, UINT nWordCount, BOOL bContinuous String AccessPassword,
String MemBankData);
```

This method allows reading an arbitrary memory location of a tag in one-word units. (1 word-unit is two bytes, 16 bits in this case.) The `nWordPtr` parameter defines the starting position and the number of words to be read is passed in as the fourth `NWordCount` parameter. The result value is always stored in a string variable (`MemBankData`). `WriteMemBank` method works very similarly but it allows writing an arbitrary memory location of a tag in one-word units. [3, 4]

## 4. The developed software for reading exNodes

For working with an `exNode` and an RFID tag the developed program contains two classes:

`cTag` and `exNode`

`cTag` contains the methods used for reading the EPC code and arbitrary memory locations, writing any memory location of the `UserMemBank`, and the necessary constants for executing these methods. [5]

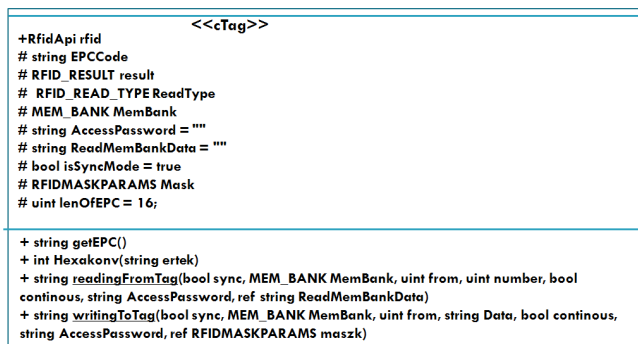


Figure 3: Class of `cTag`

### `exNode`

The result of the reading operation is stored in a list, for binding sensors with the slots and slots with memories two dictionaries are used. [6]The most important

```

<<exNode>>
+ List<string> store = new List<string>()
+ Dictionary<string, data> sensor_map
+ Dictionary<string, string> sensor_codes
# int number_of_sensors = 8
# int number_of_memories = 8

+ string command_interpreter(string command_code, string param2, string param3, string param4)
# string get_memory_info(string command_code, string memory)
# string reset(string memory)
# string slot_get_info(string command_code, string slot)
# string mem_set_slot(string command, string memory, string slot)
# string mem_set_operation(string command, string memory, string operation)
# string mem_set_log(string command, string memory, string operation)
# string slot_set_sensor(string command, string slot, string sensor)
# string slot_set_measure_period(string command, string slot, string period)
+ int Hexakonv(string ertek)

```

Figure 4: Class of exNode

method is the public command interpreter that analyses the first parameter that stores the command code and calls the corresponding protected methods.

The developed software can display the slot, sensor, memory block bindings, attributes of memory blocks and slots, as well as reset the slots. It is also possible to save the logged values of a selected sensor or all sensors to a flash card in the RFID reader (Figure 5). Changing the bindings and the attributes of slots and memory blocks is also possible as it can be seen in Figure 6.

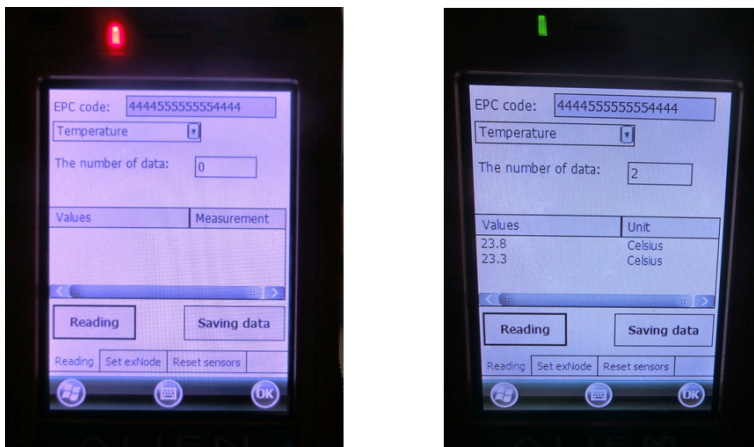


Figure 5: Saving measured data

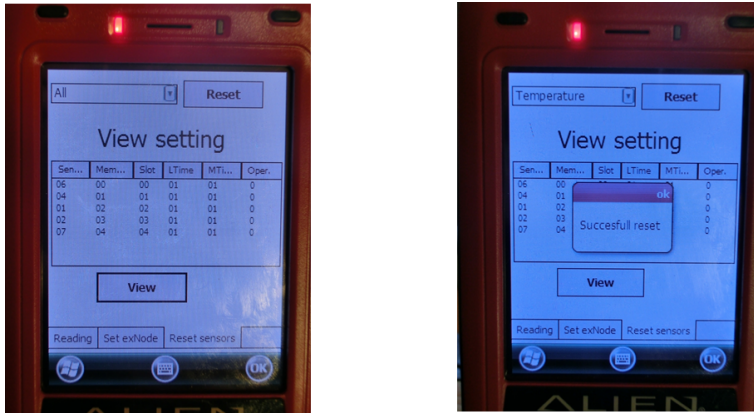


Figure 6: Changing settings

## 5. Summary

The primary aim of this paper is to demonstrate the way of developing software that enables a PDA with built in RFID reader to read the stored data measured by the sensors of extended nodes using their the integrated RFID interface. Data collected by different sensors of an exNode can be obtained by writing the proper commands into the UserMemBank of the RFID tag integrated into the board. After the controller of the exNode interprets the command it replies and writes the answer into the UserMemBank, which also means that the required data can be obtained from this memory region by an RFID reader. Using the developed software not only the stored data measured by the sensors of exNodes can be read and collected but it is also possible to change the settings of these extended nodes.

## References

- [1] KLAUS FINKENZELLER: RFID HANDBOOK Fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication, third edition, WILEY, ISBN: 978-0-470-69506-7, June 2010
- [2] EPCGLOBAL INC.: Class-1 Generation-2 UHF RFID Conformance Requirements Specification v. 1.0.2. (2005).
- [3] [ftp://ftp.alientechnology.com/pub/handheld/alh901x/docs/ALH-901x\\_Users-Guide\\_2013-12.pdf](ftp://ftp.alientechnology.com/pub/handheld/alh901x/docs/ALH-901x_Users-Guide_2013-12.pdf), 2013, 10.11.2013
- [4] [ftp://ftp.alientechnology.com/pub/handheld/alh901x/ALH-900x-901x\\_Developers-Guide.pdf](ftp://ftp.alientechnology.com/pub/handheld/alh901x/ALH-900x-901x_Developers-Guide.pdf), 2013, 10.11.2013
- [5] TOUNSI, WIEM, NORA CUPPENS-BOULAHIA, JOAQUIN GARCIA-ALFARO, YANNICK CHEVALIER, AND FRÉDÉRIC CUPPENS.: A key establishment and derivation protocol

for EPC Gen2 RFID systems. *Journal of Network and Computer Applications*, Volume 39, March 2014, Pages 152–166

[6] ANDREW STELLMAN, JENNIFER GREENE: *Head First C#*, O'Reilly Media, Inc., 2012.