XVII International Scientific Conference on Industrial Systems (IS'17)

Novi Sad, Serbia, October 4. – 6. 2017.

University of Novi Sad, Faculty of Technical Sciences,
Department for Industrial Engineering and Management

Available online at http://www.iim.ftn.uns.ac.rs/is17

Increasing the efficiency of creating prototype devices, using the open-source hardware, software, NFC tag reader and bluetooth

Ivan Čobanić

(Faculty of technical sciences, Trg Dositeja Obradovića 6, Serbia, cobanic.ivan@gmail.com)

Abstract

Development and introducing a new hardware product to the market is based on previously conducted research and testing, and is preceded by construction of a prototype device. Developement itself requires a certain period of time, hiring skilled personnel and financial resources. With the evolution and mass production of open-source hardware and software the time needed for making the prototype is significantly shortened. Here we will describe the use of open-source hardware, its application possibilities, as well as modification possibilities and its use (rapid development tools), and how to build a prototype device for reading NFC (near field communication) tags - marks. The device sends read values via Bluetooth, using a 4.0 communication standard, to a computer or mobile device. The modules are shown which, assembled, build a prototype of a device. There are various usage possibilities for this device, for example, listing goods in a warehouse, labeling objects, event logging, making an authorization of a specific application or a software product

Key words: *NFC, bluetooth, open source, prototype*

1. INTRODUCTION

Starting the production process and introducing a new device into the product range is based on the results of research and testing from the previous period of time. In order to build a prototype of a particular device, it is necessary to start from the beginning of the production cycle. Each segment of the production process requires human resources, material resources and specific manufacturing tools. Building each individual segment of the device requires pre-designing, testing and leading to compatibility research for all parts of the designed device. In cases where the sum of the necessary expenditures of material and non-material expenditures exceeds available resources, the use of open source hardware and software is one of the alternative solutions in finishing the prototype.

The built-in prototype device is required to meet the following criteria:

- Realization in a limited timeline
- Functionality of wireless reading and result forwarding
- Compact design - small dimensions
- Autonomy in field work
- Possibility of functional upgrades
- Possibility if connecting to various different devices
- Low cost
- Possibility of meeting the serial production.

Using these requirements and facts, it came to the idea of using NFC - near field communication for cordless reading and Bluetooth 4.0 for wireless forwarding of the results obtained. Open source hardware - OSHW and software - OSSW with their availability successfuly meet the criteria. [1]

2. NFC

NFC-Near Field Communication is one of the communication standards for cordless data exchange. The exchange can be one-way or two-way. The frequency at which the communication takes place is (HF) 13.56 MHz. The most desirable standards are: Mifare1 S50, Mifare1 S70 Ultralight, Mifare Pro, DESFire [2], [3]. Figure 1. shows the NFC Mifare1 tag.

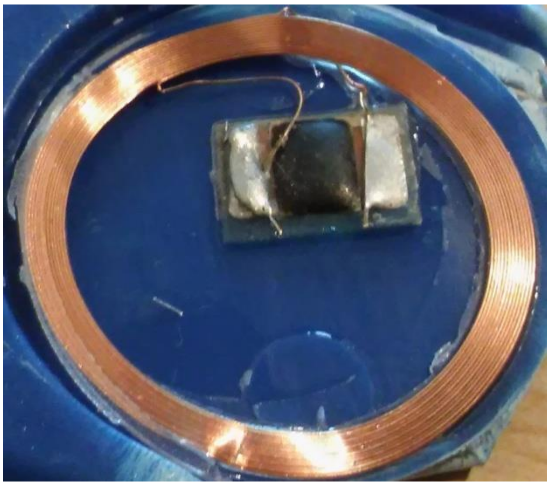

Figure 1. NFC tag

3. OPEN SOURCE HARDWARE AND SOFTWARE

Open source hardware - OSHW is commonly available and easily accessible methodology. The basic idea is to use light and widely available modular design, which is tested, easy to customize and suitable for connection with various other modules. There is also a large number of manufacturers of pre-fabricated OSHW modules on the market. Figure 2 shows one of those OSHW modules. The process of prototype construction itself is based on the purchase of the necessary modules, their modification, if required, and their connection.

Open source software - OSSW is a widely available program code bundle. Further modification and customization of the program code meets the necessary functionalities and user requirements. The application of OSHW and OSSW shortens the time required for design research, testing time and development of the base model. Using the already-fabricated modules and the program code bundle or librarys, the time for making the prototype of the device shortens significantly and resources are preserved. Table No.1 lists the OSHW modules used for making the prototype of wireless - contactless NFC tags - tags.

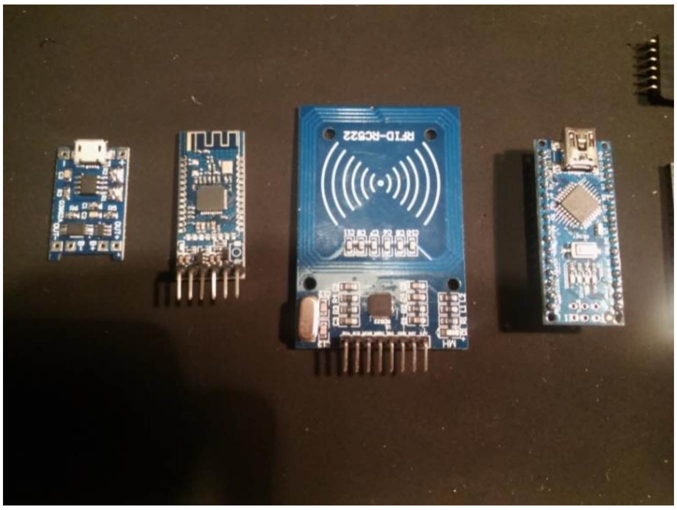

Figure 2. OSHW Modules

Table 1. Table of OSHW used

No	Hardware unit name	Description
1.	Arduino nano with atmega 328	Main microprocessor board with inputs, outputs and encripted software.
2.	Bluetooth HC-05	A cordless communication module with a computer / mobile device communication via Bluetooth 4.0 standard.
3.	RC-522 Module	NFC tag – tag reader.
4.	DC-DC converter	Power supply board and Li-ion battery charger.
5.	18650 battery	A 2200 mAh Li-ion battery which feeds the entire circuit.
6.	Raster board	Perforated board with printed copper circuit.

3.1 **OSHW and OSSW applied in NFC reader producing**

Arduino nano with atmega 328 AVR microcontroller is the main element in making a prototype. The module has 32KB available flash memory for storing software code. It has 14 digital input / output pins, 6 analog input / output jacks (pins), serial port (rs232) and SPI. The clock speed of the microprocessor is 16 MHz [4]. The total capacity of 2KB is taken for the bootloader code. Bootloader allows faster and easier loading of the compiled source code into the microprocessor itself, in a way that no dedicated programmer is required. This module contains a power controller on its board, an RS232 USB converter which greatly simplifies the process of prototype construction [5].

Bluetooth HC-05 is an easy solution for serial communication using rs232 or via Bluetooth 4.0 standard. The module is equipped with a voltage stabilizer, LED status indicator, a mode switch and an AT set of commands as well as connectors (pins) for connecting with other modules [6].

RC-522 module for reading NFC bookmarks - tags. Supports next standards: Mifare1 S50, Mifare1 S70 Ultralight, Mifare Pro, DESFire [6]. The module contains a MFRC522 chip with software installed by the manufacturer. The advantage of using this module is in getting raw data from NFC tag, filtering them, neutralizing noise and other unwanted impediments. The obtained and processed results are transmitted via the built-in SPI interface to other modules [7].

The power supply of any device that is expected to meet the criteria of small dimensions and the autonomy of work in conditions where there is no electrical energy infrastructure is one of the key problems. Resolving this criterum on a specific project is realized through the use of a DC-DC converter - module, which contains a voltage regulator, a Li-ion battery charger in the 3.7V-4V range, step-up voltage boosters in the 4.8V-5 interval, 1V as well as measuring and regulating functionality which has the role of protecting the modules powered by this converter. The task of this module is to measure the battery voltage as well as the current flowing through the electronic circuitry, protecting and creating a redundant power source for the device. The module provides a constant, optimal voltage of 5V and limits the current of 500mA, so that all prototype elements have optimum power supply.

The 18650 battery is a power source and provides the autonomy of the device's prototype operation. The nominal battery voltage is 4.2V and the capacity is 2200 mAh.

The raster board is a perforated plate of vitroplast. Copper lines are horizontally printed on the board with thermoplastics. Each duct is perforated with its entire length. Such a design of the board allows the rapid development of prototype devices, since the need for designing, routing and running lines is eliminated. It is easy to cut the unnecessary lines and switch the necessary lines with the insulated conductor. Figure 3. shows a set of modules used in the making of this prototype.

Figure 3. Modules and OSHW elements used

The Open Source Software - OSSW which was used in the making of this prototype is based on libraries in which methods, classes, and objects are encapsulated. By using libraries in the code, the time required for the development of their own libraries is quite shortened. OSHW libraries which were used to support OSHW:

- *SPI.h* - Provides communication with the reader;
- *MFRC522.h* - Provides communication with the reader;
- *SoftwareSerial.h* – Provides serial communication with the bluetooth module;

By implementing the SoftwareSerial.h library in the program code, after testing, a functional connection was established between master-slave, the prototype of the device and the computer / mobile device. The MFRC522.h library and SPI.h with their implementation provided functionality of the forwarding readout value of the NFC tag to the atmega 328 via the *SPI* interface. Further testing and program code developing resulted in a program version loaded in the flash memory of the atmega 328 microprocessor. The crown of this procedure is a functional prototype device with the functionality of reading the value of the NFC tag and forwarding that value over BlueTooth like virtual COM-serial port. Figure 4. shows the interior of the prototype with connected modules. By combining this OSHW and OSSW, a prototype device for contactless - wireless reading of the values recorded in the NFC tag has been finished. The read value is transmitted via Bluetooth to the device with which it is paired.

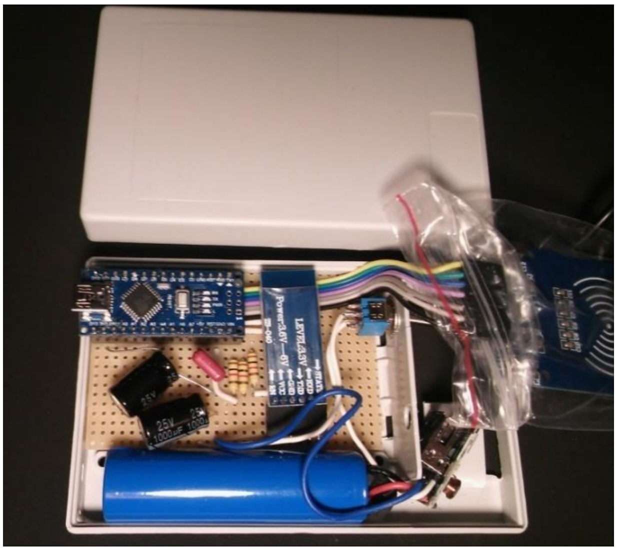

Figure 4. The inside of the device

4. APPLICATION OF OSSW IN DEVELOPING A PROTOTYPE APPLICATION FOR VISUALING FORWARDED DATA FROM THE NFC READER

Visualizing and displaying of forwarded data read by the NFC reader is done with the application software. The application software for Android and Windows operating system has been described. Android software was developed using the MIT App Inventor Development Environment. The main advantage of this environment next to GPL is the interaction between programmers and the environment. Creation, sequences, selections, iterations, functions, classes, objects, methods, and other elements of object-oriented programming is done through a simplified and intuitive user interface with limited capabilities. This performance allows the rapid development of software prototypes. Figure 5. shows the GUI of the development environment.

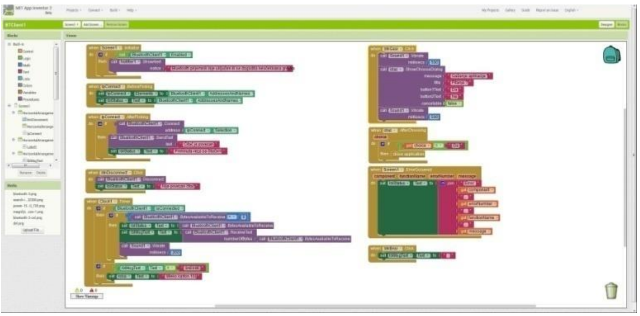

Figure 5. MIT App inventor - GUI

The code generated by the development environment of the MIT App inventory and compiled into the apk executable file is ready for execution on Android OS. For the purpose of demonstrating the functionality of the NFC tag prototype reader, the NFC marker on the product packaging was placed on the case of a list of goods in the warehouse. By contactless reading of the NFC tags from the packaging, a mobile device, which was Bluetooth-paired with a reader, was passed by the NFC UID value. Android app matches these values and finds the appropriate item and displays its properties. Figure 6. shows the GUI Android application that shows the name and quantity of the read article and well-known featured articles marked with NFC tags. [8]

Figure 6. Android app showing the results of an unknown and familiar article that has been read by the NFC tag apparel

In addition to the Android application developed with limited capabilities and resources, a test application in the Microsoft .NET development environment has been also developed in C# to demonstrate the prototype functionality. In addition to the settings that define the serial port number for establishing a paired Bluetooth connection, the application also contains user-defined fields that store the values of the attributes assigned to the individual item in the SQL database. This can show the manufacturer information, expiration date, etc. Figure 7. shows the GUI windows application displaying details about the NFC tag. [9]

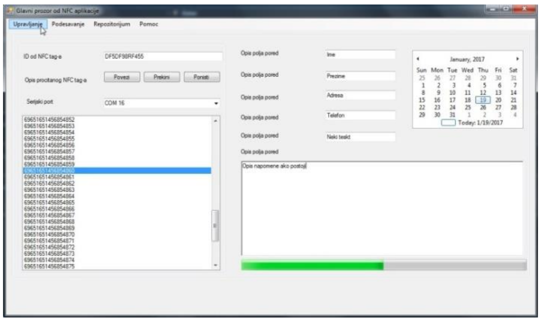

Figure 7. Windows application GUI to visualize the properties of an article whose NFC tag has been read

5. CONCLUSION

This document describes the procedure for selecting, connecting and adjusting compatibility between open source hardware modules and open source software. The final product of the precudere described was a prototype device that satisfies the criteria set at the beginning. The advantage of this approach is shortened development and testing time of the prototype and the final product. By further investment and engagement of the project team, it is possible to bring the prototype into a state when the device is ready for serial production. There is no hardware and device usage constraint. By customizing the software in the device and application visualization software it is possible to extend the scope of application and operation of the described prototype. Further plans to improve the prototype might be a more advanced concept solution for power generation, reduced power consumption by automatically shutting down the device after each transmission, thus achieving longer autonomy, as well as reducing armor dimensions and increasing ergonomics. Also, with the development of software, services and libraries to be implemented, there is a possibility that the final prototype can be even more efficient and fast developed and applied in many other areas.

6. REFERENCES

[1] R. Vinyard, B. Forrest, R. DiResta (June 2015), “The Hardware Startup” O'Reilly Media, Inc. 79-138

[2] V. Coskun , K. Ok, B. Ozdenizci (2012), “*Near Field Communication (NFC): From Theory to Practice 1st Edition*” 50-76

[3] Product component datasheet : http://www.atmel.com/Images/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Summary.pdf (accessed on 10.1.2017). 5-16

[4] A. Trevennor (October 16, 2012), “*Practical AVR Microcontrollers*” Apress; 1st ed. edition 57-77

[5] K. Townsend, C. Cufí (2014), “*Getting Started with Bluetooth Low Energy*” Oreilly Media, 97-111

[6] T. Igoe, D. Coleman (2014), “*Beginning NFC: Near Field Communication with Arduino, Android, and PhoneGap*” Oreilly Media 11-21

[7] Product data sheet http://www.nxp.com/documents/data_sheet/

[8] MFRC522.pdf 1 -34 (accessed on 10.1.2017)

[9] Ostojić, G., Stankovski, S., Lazarević, M., Škrinjar, D. (2006), "*Application of RFID technology in the process of storing goods to supermarkets*", Elaboration summary of the 9th International Conference, Quality and Reliability Management DQM-2006, Belgrade.

[10] John Allwork (2010), "*Visual Studio C# 2010 Programming and PC Interfaces"* 248-255