***Я.М. Зубов****,*

***И.И. Ильин****,*

*КГУ,*

*г. Курган*

**Модель системы контроля и управления доступом на предприятии на базе Arduino**

**Arduino-based access control system model for enterprise**

Аннотация: Целью работы является разработка открытой системы контроля и управления доступом (СКУД) на базе программно совместимого аналога arduino, которая будет существенно более выгодной в коммерческом плане по сравнению с существующими промышленными аналогами за счёт бесплатности созданного в процессе программного обеспечения и доступности составляющих аппаратную часть компонентов. Статья является отчётом о промежуточном состоянии наработок.

Ключевые слова: инженерно-техническая защита информации, предохранительные устройства и мероприятия, контроль территории, носимые устройства идентификации, OneWire, iButton, Arduino.

Annotation: The aim of the labor is developing of an open arduino-based access control and monitoring system which will be much more commercially benefitial comparing with represented on the market industrial solutions due to gratuity of simultaneously created software and to cheapness of electronical components. The article is a report about the current condition of the work.

Keywords: engineering and technology information defense, safety devices and actions, territory control, wearing identification devices, OneWire, iButton, Arduino.

С изобретением и широким распространением микроконтроллеров появились небывалого разнообразия возможности для развития пользовательской электроники. Одна из сфер применения программируемых чипов – различные СКУД. Такие системы разрабатываются и продаются, но цены на готовые комплексы представляются чрезмерно большими. Например, стоимость минимального комплекта программного обеспечения (ПО) SP09 от ведущего российского производителя систем безопасности – PERCo составляет 265 евро [1].

Необходимость расходов, которые несут потребители для того, чтобы возместить производителям постоянные затраты материальных, человеческих ресурсов (на производство преграждающих устройств, замков, печатных плат и т.д.) не вызывает сомнений. Однако, в смете имеет место и программное обеспечение. Проприетарное, не обновляемое, оно, единожды окупив продажей некоторого количества своих лицензий затраты на разработку, далее не может справедливо также высоко цениться. Тем временем по всему миру разворачиваются проекты категории free and open-source software (FOSS), а исходный код различных уважаемых, известных платных решений становится общедоступным. Последнее нередко происходит в связи с давлением перспективы быть вытесненными с рынка за нерентабельностью использования в сравнении с применением свободных аналогов. В связи с этим мы считаем, что создание программно-аппаратного решения с открытыми исходным кодом и архитектурой поможет решить выше описанную проблему.

В ходе разработки нами были изучены:

1. Touch memory – технология, Touch memory key,

2. OneWire – технология, протокол взаимодействия,

3. Промышленные аналоги СКУД,

4. Проектировочная платформа Arduino.

Основой нашего программно-аппаратного комплекса стал контроллер ATMega328p, используемый на плате Arduino. Основной причиной его использования послужил низкий порог вхождения в процесс разработки: готовые IDE (англ. Integrated Development Environment), набор открытых библиотек и примеров к ним, дешевизна оборудования для программирования контроллера.

1-Wire(OneWire) – технология, изобретённая для упрощения и удешевления подключения небольших, не требующих высокой мощности питания и пропускной способности канала устройств. Данная технология позволяет подключать по одному физическому каналу несколько устройств, управляющихся одноимённым протоколом[2]. Эта возможность обеспечивается используемым принципом передачи данных. Биты синхронизируются временными слотами (TimeSlots) и определяются их продолжительностью.

Устройством для хранения UID Touch memory или iButton [3] - оригинальное семейство микросхем, разработанных и выпускаемых фирмой Dallas Semiconductor, USA. Каждая такая микросхема заключена в стальной герметичный цилиндрический корпус, служащий для защиты. Нами были использованы ключи, относящиеся к семейству DS1990A с восьмибайтовым ПЗУ [4, 5]. В младшем байте содержится код семейства, для DS1990A он всегда будет равен 01h. В шести последующих байтах содержится серийный номер ключа (уникальный 48-битный код). Последний байт называется cyclic redundancy check (CRC) - это контроль четности, обеспечивающий подлинность переданных данных (рис. 1).



Рис. 1. Структура ПЗУ ключа

Работа контроллера состоит из считывания идентификатора ключа, сигнализирования, отправки информации контролирующему СКУД устройству посредством интерфейса USB, получения от последнего команды (на открытие, игнорирование и ожидание нового ключа или просто на игнорирование) и реакции на команду. Со стороны ПО на персональном компьютере пользователям, в соответствии с уровнем их прав, предоставляются различные возможности взаимодействия со СКУД. При получении данных от контроллера производится сверка UID (англ. User Identifier – идентификатор пользователя) с хранимыми в базе данных и, в случае совпадения, на мониторе отображается информация. Был разработан простой протокол взаимодействия. Для определения, вход или выход совершается сотрудником, используется дополнительный байт. Чтобы ускорить процесс реализации, была создана визуализация протокола в виде блок-схемы (рис. 2).

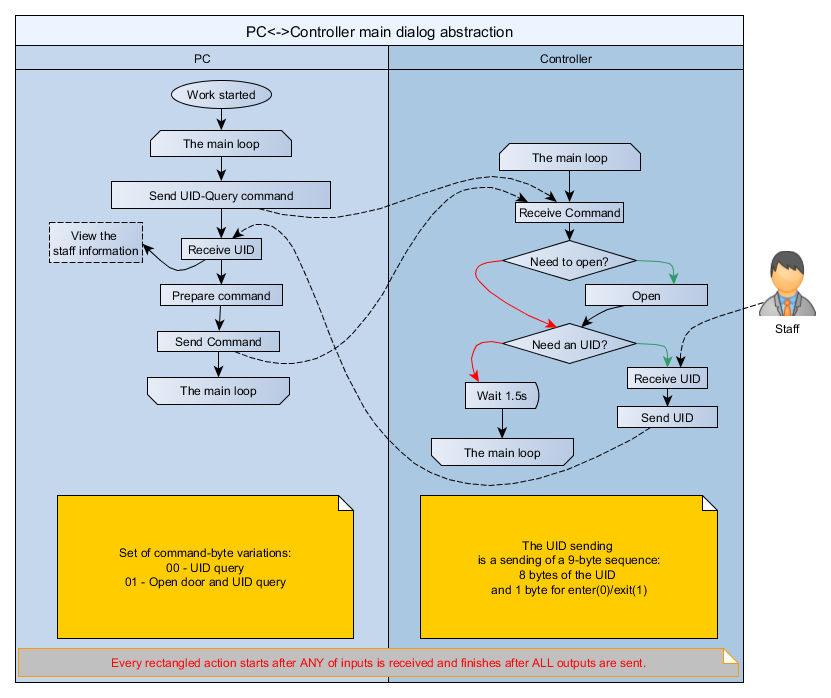


Рис. 2. Блок-схемы визуализации протокола

Выбранная в качестве базы для разработки системы платформа Arduino снабжена специализированным ПО[6].

Бесплатно распространяемая интегрированная среда разработки Arduino IDE 1.6.1 предоставляет условия для удобных написания, компилирования и загрузки управляющего кода.

Ниже представлены схема сборки и внешний вид аппаратной части системы (рис. 2,3).

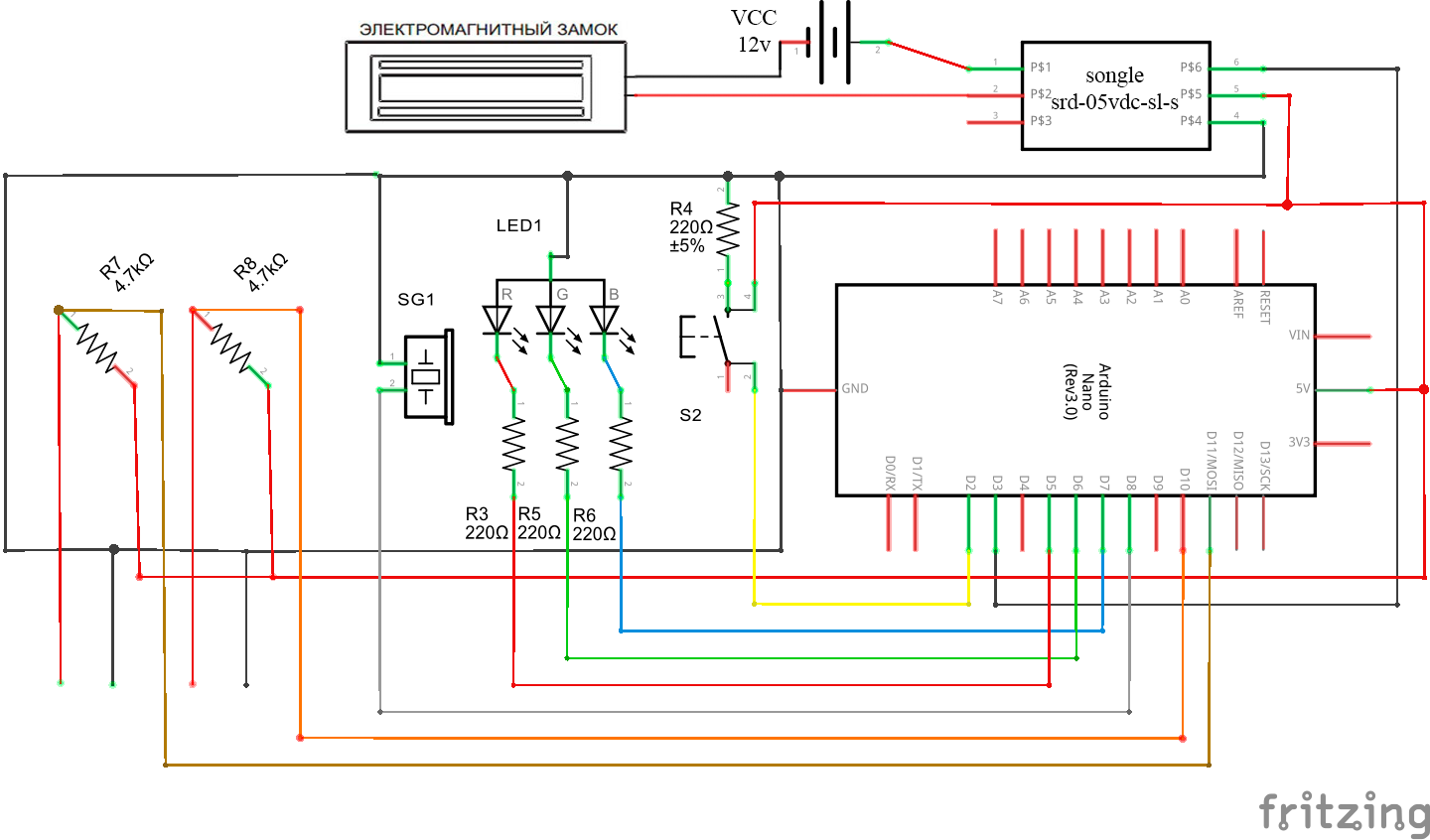


Рис. 2. Принципиальная схема

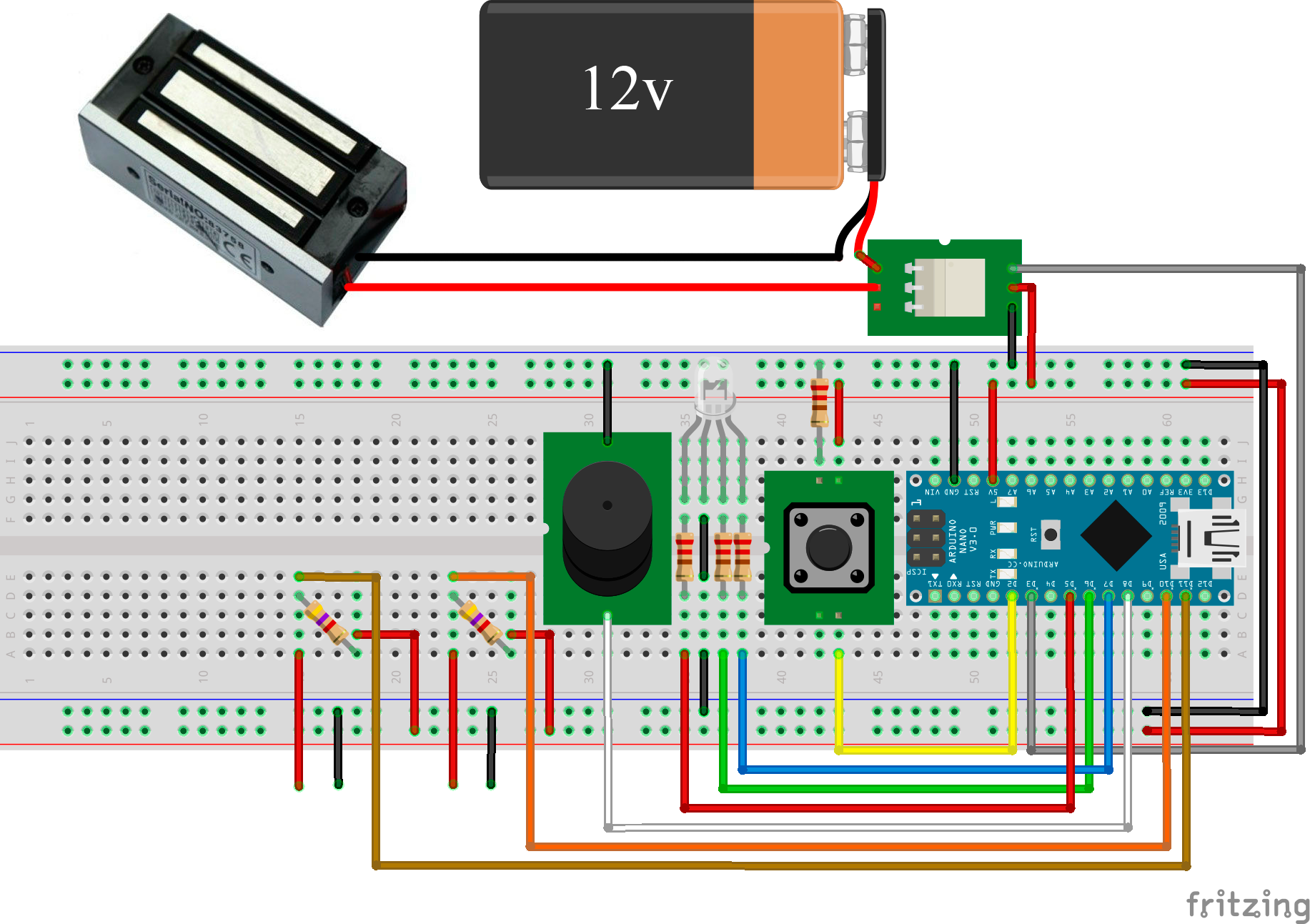


Рис. 3. Внешний вид результата сборки

Пользователи ПО могут обладать одним из 3х типов привилегий: администратор, директор, охранник. Для каждого из них предусмотрен индивидуальный интерфейс, позволяющий охраннику сверять фотографию на мониторе с лицом проходящего через контрольно-пропускной пункт (КПП), директору – просматривать журнал посещения предприятия сотрудниками, администратору – обеспечивать работоспособность системы.

“СКРИНШОТ ФОРМЫ ОХРАННИКА”

Для примера возьмем комплект ПО и оборудования PERCo-KT02.3 с активированным PERCo-SP13, его стоимость составит 1135 евро [7]. Стоимость же компонентов для нашей СКУД составит:

Таблица 1

Итоговая стоимость компонентов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Обозначение | Кол-во | Тип | Свойства | Стоимость |
| LED1 | 1 | RGB LED | 4х контактный RGB светодиод | 0,93 € / 1 шт. |
| R3,R4,R5,R6 | 4 | 220Ωрезистор | допуск ± 5% сопротивление 220Ω | 1,74 € / 200 шт. |
| SG1 | 1 | Buzzer 12mm | Зуммер 12мм | 0,79 € / 1 шт. |
| S2 | 1 | Button | Контактная кнопка 12мм | 0,29 € / 1 шт. |
| R7,R8 | 2 | 4,7kΩрезистор | допуск ± 5% сопротивление 4,7kΩ | 1,79 € / 200 шт. |
| VCC 12V | 1 | VCC | Блок питания 12В 1А | 2,14 € / 1 шт. |
|  | 1 | МК | Arduino Nano (Rev. 3) | 3,19 € / 1 шт. |
|  | 1 | Реле | songle srd-05vdc-sl-s | 1,07 € / 1 шт. |
|  | 1 | Lock | Магнитный замок | 16,72 € / 1 шт. |
| Итого: ~ 25,19 € | | | | |

На данном этапе цель работы продолжает казаться достижимой: цена на комплектующие, уже использующиеся в нашем аппаратном комплексе, составляет менее 5% от цены, установленной за комплект оборудования, поставляемого известными производителями. Кроме заметной разницы в стоимостях аппаратных систем, следует ещё раз подчеркнуть отсутствие затрат, связанных с проприетарностью нашей СКУД, благодаря абсолютной открытости её исходного кода и архитектуры. Любой желающий имеет ничем не ограниченную возможность обратиться к серверам GitHub с целью полного или частичного копирования для себя наших наработок и пользоваться ими на условиях открытой лицензии GPL третьей или более поздней версии[8].

Ожидается увеличение (не более, чем в два раза) издержек за счёт добавления таких периферийных компонентов, как удлинённые кабели (USB и провода к интерфейсам считывания и к магнитному замку) и корпус из оргстекла или подобного материала.

ПО требует доработки не в меньшей степени. Набор библиотек для графического интерфейса, который мы использовали для его создания — Windows Forms. Это так, потому что только с ним мы были достаточно знакомы на момент начала разработки. Теперь, когда нам начали преподавать теорию распределённых вычислительных систем (РВС), мы понимаем, что программа для ПК должна быть полностью переработана — переписана с использованием Web-технологий: необходимо вынести логическую часть из приложения клиента в приложение сервера, оставив первому только возможность получения интерфейса в Web-браузере.  
Проблема выбора платформы для реализации этих планов пока остаётся для нас открытой: поверхностное ознакомление с возможностями и особенностями каждой из них ещё только начинается.

***Литература и примечания:***

[1] Комплекты ПО S-20 <https://www.perco.ru/products/sistemy-kontrolya-dostupa/sistema-kontrolya-dostupa-s-20/komplekty-programmnogo-obespecheniya/>

[2] OneWire <http://en.wikipedia.org/wiki/1-Wire>

[3] iButton <http://www.ibutton.ru/about/info/>

[4] DS1990A datasheet <http://datasheets.maximintegrated.com/en/ds/DS1990A.pdf>

[5] Копирование ключей iButton DS1990A <http://electromost.com/news/kopirovanie_kljuchej_ibutton_ds1990a/2011-05-04-26>

[6] Arduino code tutorials and examples <http://arduino.cc/en/Tutorial/HomePage>

[7] Прайс-лист электронные проходные с ПО <https://www.perco.ru/download/price/price_PERCo.pdf>

[8] Страница проекта на GitHub <https://github.com/veter069/OneWireArdiuno2015>