

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОДОВОГО ЗАМКА НА БАЗЕ МИКРОКОНТРОЛЛЕРА СЕМЕЙСТВА АТМЕГА

В.С. Лучников¹, Н.Ю. Юдина¹

¹ФГБОУ ВО «Воронежский государственный лесотехнический университет
имени Г.Ф. Морозова»

Аннотация. В работе рассматриваются разработка кодового замка на основе микроконтроллера Atmega. Создание этой системы на базе микропроцессора позволит получить безопасное, удобное, простое в обращении и несложное в изготовлении устройство. Актуальность разработки данного устройства обусловлена тем, что кодовые замки являются эффективным средством предотвращения доступа посторонних лиц к охраняемым помещениям. Несомненным преимуществом кодовых замков будет отсутствие необходимости изготовления ключей.

Ключевые слова: Программирование микропроцессоров, Atmega32, кодовые замки.

DEVELOPMENT OF A COMBINATION LOCK SOFTWARE BASED ON ATMEGA FAMILY MICROCONTROLLERS

V.S. Luchnikov¹, N.Yu. Yudina¹

¹Voronezh State University of Forestry and Technologies named after G.F. Morozov

Abstract. The paper discusses the development of a combination lock based on the Atmega microcontroller. The creation of this microprocessor-based system will make it possible to obtain a safe, convenient, easy-to-use and easy-to-manufacture device. The relevance of the development of this device is due to the fact that combination locks are an effective means of preventing unauthorized persons from accessing guarded premises. The undoubted advantage of combination locks will be the absence of the need to manufacture keys.

Keywords: Programming of microprocessors, Atmega32, combination locks.

Задача данной статьи состоит в разработке программного обеспечения кодового замка на базе микроконтроллера семейства Atmega с помощью про-

граммы автоматизированного проектирования электронных схем Proteus 8 Professional и интегрированной среды разработки Atmel Studio 7. Разработка будет производиться на языке СИ.

Реализация данного устройства поможет обеспечить безопасность помещения. Это могут быть жилые, рабочие помещения. В ходе проектирования необходимо решить ряд задач: подключить матричную клавиатуру и LCD дисплей к микроконтроллеру, написать ряд подпрограмм, позволяющих решить поставленную задачу.

К основным функциям устройства, можно отнести:

1. Считывание ввода комбинации чисел с матричной клавиатуры “Keypad-Phone”;
2. Вывод введенной комбинации на LCD дисплей “LM044L”;
3. Проверка введенного кода на правильность.

Исходя из функциональных требований, предъявленных к устройству, электронный кодовый замок должен включать в себя следующие элементы:

1. Микроконтроллер на базе семейства Atmega;
2. Клавиатура. Для ввода запрашиваемого пароля;
3. Устройства сигнализации об открытии двери и о попытке подбора кода. Для реализации данного пункта будут использованы два светодиода: красного цвета, для информирования о попытке подбора кода и зеленого цвета для информирования об успешном открытии двери;
4. LCD дисплей. Для вывода нажатых клавиш, для отображения информационных надписей.

Главный программный файл main.c содержит в себе 3 функции: функция для начала работы с кодовым замком writeTitle(), функция инициализации портов ввода/вывода port_init() и главная функция int main(), в которой реализована главная функциональная часть. Для этого программного файла подключается директива #define определяющая частоту процессора в герцах.

Также подключается библиотека для работы с вводом/выводом микроконтроллера io.h, библиотека для работы со строками string.h, библиотека для работы с программной задержкой delay.h и две пользовательские библиотеки для работы с матричной клавиатуры keypad.h и для работы с LCD дисплеем lcd.h.

Функция writeTitle() представленная на рис. 1 позволяет инициировать процесс работы с кодовым замком, она выполняет вывод сообщения о просьбе ввести код и включает курсор для дальнейшего ввода. Также функция включает

0 бит порта A, что позволяет активировать свечение светодиода красного цвета, который будет сигнализировать о том, что дверь закрыта.

```
//функция начала работы
void writeTitle(void){
    //очистка lcd
    writeCommand(LCD_CLEAR);
    setXy(0,0);
    //вывод сообщения
    writeString("Enter code: ");
    //включение курсора
    writeCommand(CURSOR_ON);
    setXy(0,1);
    //включение 0 бита порта A
    PORTA=0b0000001;
}
```

Рисунок 1 – Функция начала работы с кодовым замком

Функция port_init(), представленная на рис. 2, позволяет инициализировать порт D, как порт для ввода/вывода информации.

```
//определение функции инициализации портов ввода/вывода
void port_init(void)
{
    DDRD=0xFF;
    PORTD=0x00;
    DDRA=0xFF;
}
```

Рисунок 2 – Функция инициализации портов ввода/вывода

На рис. 3 отображен фрагмент кода, который выполняет проверку введенного пароля и в случае выполнения условия на правильность выводит соответствующее сообщение и включает 1 бит порта A, для зажигания светодиода зеленого цвета. Данный код вступает в силу при введении 4 символов.

```
// если введено 4 символа
if(lenSymbol == 4){
    // сравнение введенных символов с паролем
    if(strcmp(str, "3108") == 0){
        // очистка экрана
        writeCommand(LCD_CLEAR);
        // включение 1 бита порта A
        PORTA=0b0000010;
        //вывод сообщения о успешном открытии замка
        writeString("Access is allowed!");
        writeCommand(CURSOR_OFF);
        //обнуление переменных и элементов массива
        lenSymbol = 0;
        str[0] = "";
        str[1] = "";
        str[2] = "";
        str[3] = "";
        // замок закроется автоматически через 30 секунд
        _delay_ms(30000);
        // снова запускается функция начала работы
        writeTitle();
    }
}
```

Рисунок 3 – Проверка введенного пароля на правильность

Рис. 4 позволяет отобразить действия кодового замка при неправильно введенной комбинации пароля. Если пароль неправильно введен меньше 3 раз, то выводится информирующее сообщение и запускается программная задержка длиной в 3 секунды, после которой снова запускается функция начала работы кодового замка writeTitle(). Если пользователь ввел неправильную комбинацию 3 раза, то на кодовый замок накладывается блокировка на 5 минут, после которой снова будет доступна возможность ввода пароля.

```

else{//если код неверный
// инкрементируем переменную количества попыток
numAttempts++;
// если количество попыток меньше 3
if(numAttempts < 3){
writeCommand(LCD_CLEAR);
// вывод сообщения на экран
writeString("Try again!");
writeCommand(CURSOR_OFF);
//обнуление переменных и элементов массива
lencymbol = 0;
str[0] = "";
str[1] = "";
str[2] = "";
str[3] = "";
// задержка 3 секунды
_delay_ms(3000);
// снова запускается функция начала работы
writeTitle();
}
else{// иначе
writeCommand(LCD_CLEAR);
//вывод сообщения о блокировке доступа и просьба подождать 5 минут
writeString("X Access denied!");
setXy(0,1);
writeString("Too many attempts.");
setXy(0,2);
writeString("Wait 5 min.");
writeCommand(CURSOR_OFF);
// задержка 5 минут
_delay_ms(300000);
// снова запускается функция начала работы
writeTitle();
//количество попыток равно 0
numAttempts = 0;
}
}
}

```

Рисунок 4 – Обработка при неправильно введенном пароле

В ходе анализа функциональных требований к устройству, можно сделать вывод, что для корректной работы электронного кодового замка будет необходимо использовать 6 элементов. Это такие электронные компоненты программы Proteus как:

1. Микроконтроллер Atmega32. Данный микроконтроллер был выбран, потому что имеет хорошую пропускную способность команд. Также к данному микроконтроллеру возможно подключить периферийные устройства. Несомненным преимуществом данного микроконтроллера перед некоторыми представителями семейства Atmega, будет тот факт, что Atmega32 имеет 4 порта, которые могут работать, как на вход, так и на выход;

2. LCD дисплей “LM044L”. Данный дисплей был выбран, потому что он является достаточно крупным представителем среди своих конкурентов по соотношению сторон, что позволит единоразово выводить большое количество

информации. Преимуществом дисплея является тот факт, что он без проблем интегрируется с микроконтроллерами семейства Atmega;

3. Матричная клавиатура “Keypad-Phone”. Данная клавиатура была выбрана, так как является самым минималистичным представителем матричных клавиатур в среде Proteus;

4. 2 светодиода зеленого и красного цветов;

5. Кнопка сброса.

На рис. 5 отображена электронная схема кодового замка с подключенными устройствами. Можно заметить, что для каждого периферийного устройства выделен свой порт. Так, для LCD дисплея выделен порт D и конкретной шине данных D2...D3 соответствует свой бит порта. Для матричной клавиатуры выделен порт C, для столбцов определены биты от 0 до 2, а для строк от 4 до 7. Первые 2 бита порта A соответствуют двум светодиодам.

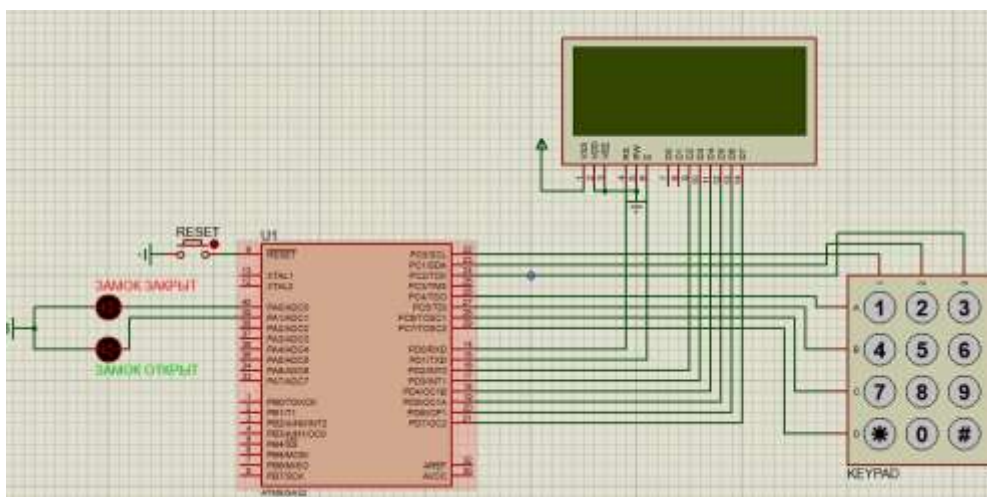


Рисунок 5 – Электронная схема кодового замка

После описания электронной схемы и процесса установки прошивки, можно перейти к моделированию работы устройства. На рис. 6 отображено информирующее сообщение о вводе пароля. Данное сообщение отображается сразу после запуска симуляции.

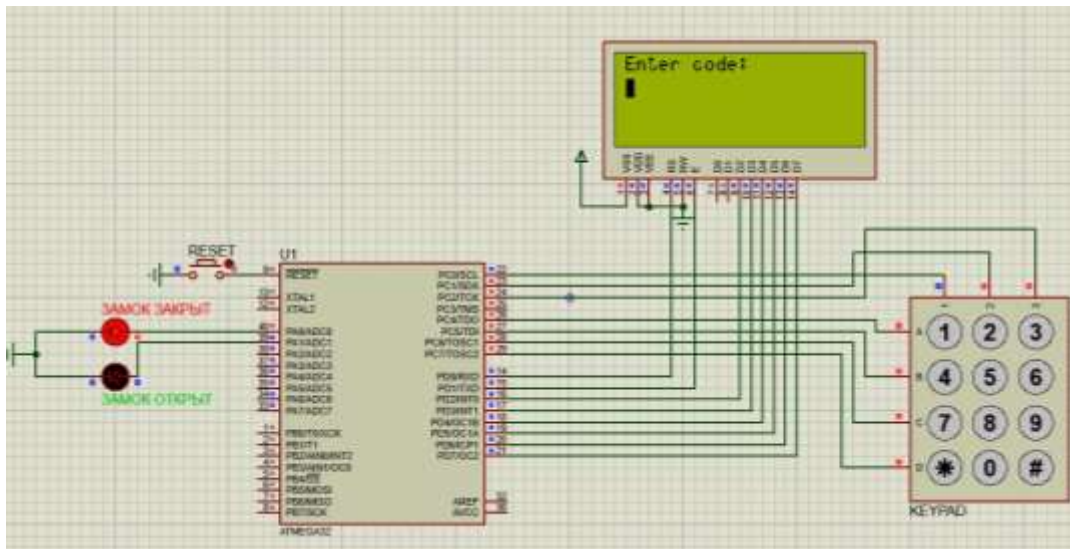


Рисунок 6 – Начало работы с кодовым замком

Если код был введен верно, то на LCD высветится информирующее сообщение об этом и загорится светодиод зеленого цвета. На рис. 7 отображена правильная попытка ввода кода.

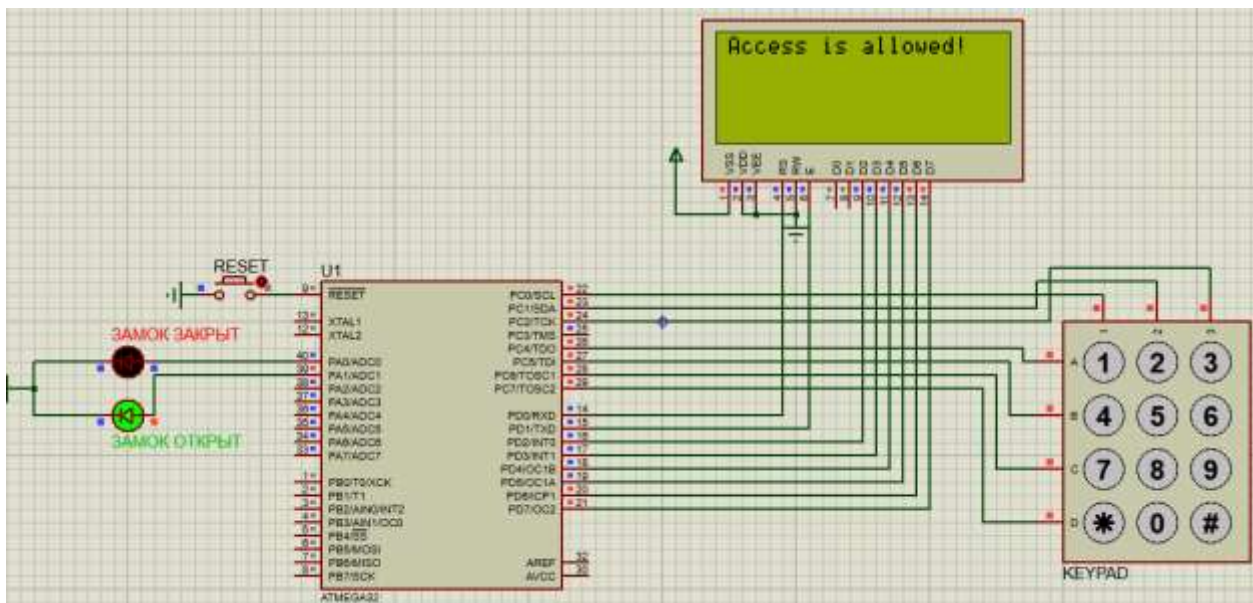


Рисунок 7 – Правильная попытка ввода кода

На рис. 8 отображена работа кодового замка при неверной попытке ввода пароля.

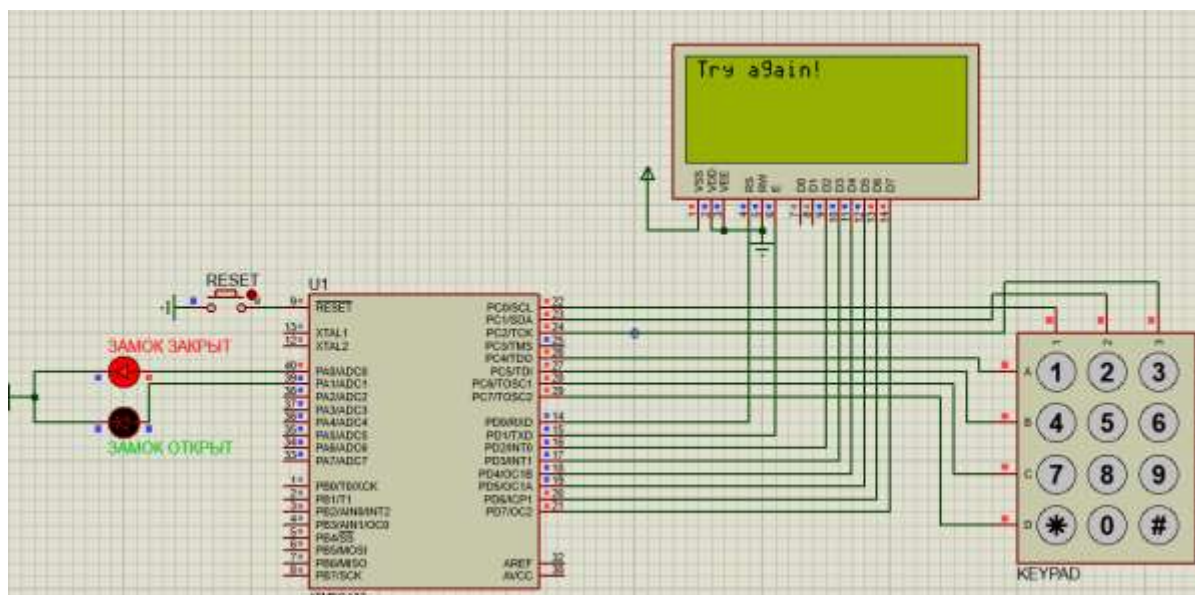


Рисунок 8 – Неверная попытка ввода пароля

На рис. 9 отображена блокировка замка после достижения трех попыток. На LCD дисплей выводится информирующее сообщение о возможности повторной попытки ввода пароля через 5 мин.

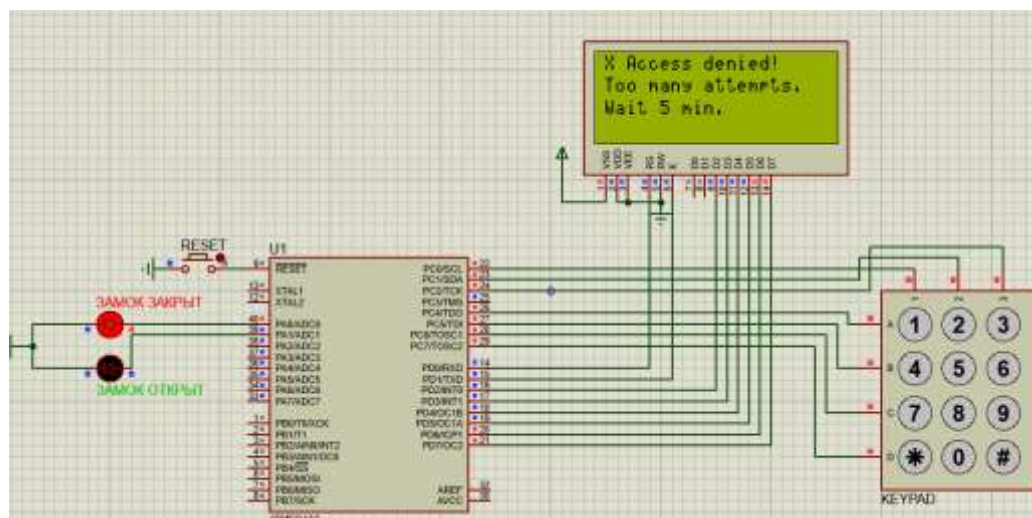


Рисунок 9 – Блокировка замка после достижения фиксированного числа попыток

В ходе проведения моделирования работы устройства, было проведено проектирование электронной схемы кодового замка в программе Proteus v8.15 SP1 Professional. В результате проверки работы схемы, можно сделать вывод, что устройство работает корректно и полностью отвечает функциональным требованиям.

ЗАКЛЮЧЕНИЕ

Данная статья отображает процесс проектирования кодового замка с микроконтроллерным управлением. Разработка кодового замка выполнялась на основе микроконтроллера Atmega32, с использованием периферийных устройств, таких как: матричная клавиатура KeyPad-Phone, LCD дисплей “LM044L”, два светодиода и кнопка сброса.

Для решения поставленной задачи были изучены аналоги данных устройств, включая анализ режимов работы. По результатам изучения были выделены необходимые электронные компоненты, которые позволили реализовать кодовый замок. Также отдельно была изучена техническая документация от компании “Atmel” и документация для программного средства Proteus v8.15 SP1 Professional. В ходе проведения функционального анализа программ для проектирования электронных схем, удалось прийти к выводу, что использование Proteus v8.15 SP1 Professional было самым оптимальным решением, так как данная программа смогла полностью покрыть поставленную задачу.

Разработанная система может быть применена для обеспечения безопасности помещения. Это могут быть жилые, рабочие помещения. Электронный кодовый замок имеет несомненное преимущество перед обычным дверным, так как нет необходимости переносить ключи, которые можно перепутать, потерять, поломать.

При проведении тестирования и отладки устройства, можно сказать, что электронный кодовый замок полностью соответствует установленным требованиям и данное устройство выполняет свои функциональные задачи корректно.

Список литературы

1. Евдокимова, С.А. Математико-статистическая оценка результатов теста на основе IRT / С.А. Евдокимова, М.А. Кащенко // Моделирование систем и процессов. – 2020. – Т. 13, № 3. – С. 16-22.
2. Полуэктов А.В., Макаренко Ф.В., Ягодкин А.С. Использование сторонних библиотек при написании программ для обработки статистических данных // Моделирование систем и процессов. – 2022. – Т. 15, № 2. – С. 33-41.
3. Новикова, Т.П. Разработка алгоритма и модели функционирования информационной системы для платного отделения стоматологической поликлиники / Т.П. Новикова, А.А. Бодин, С.А. Евдокимова // Моделирование систем и процессов. – 2021. – Т. 14, № 1. – С. 51-58. – DOI: 10.12737/2219-0767-2021-14-1-51-58.

4. Левенталь, Л. Введение в микропроцессоры: Программное обеспечение, аппаратные средства и программирование / Л. Левенталь. - М.: Энергоатомиздат, 2021. - 464 с.

5. Официальная техническая документация “Atmel”. URL: <https://www.microchip.com/content/dam/mchp/documents/atmel-start>.

References

1. Evdokimova, S.A. Mathematical and statistical evaluation of test results based on IRT / S.A. Evdokimova, M.A. Kashchenko // Modeling of systems and processes. – 2020. – Vol. 13, No. 3. – pp. 16-22.

2. Poluektov A.V., Makarenko F.V., Yagodkin A.S. The use of third-party libraries when writing programs for processing statistical data // Modeling of systems and processes. - 2022. – vol. 15, No. 2. – pp. 33-41.

3. Novikova, T.P. Development of an algorithm and a model for the functioning of an information system for a paid department of a dental clinic / T.P. Novikova, A.A. Bodin, S.A. Evdokimova // Modeling of systems and processes. - 2021. – Vol. 14, No. 1. – pp. 51-58. – DOI: 10.12737/2219-0767-2021-14-1-51-58.

4. Leventhal, L. Introduction to microprocessors: Software, hardware and programming / L. Leventhal. - М.: Energoatomizdat, 2021. - 464 p.

5. Official Atmel technical documentation. URL: <https://www.microchip.com/content/dam/mchp/documents/atmel-start>.