Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Схемотехника

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

МИКРОПРОЦЕССОРНОЕ УСТРОЙСТВО ДЛЯ ОБНАРУЖЕНИЯ МЕТАЛЛИЧЕСКИХ ОБЪЕКТОВ

БГУИР КП 1-40 02 01 111 ПЗ

Студент: группы 150501,   
Климович А. Н.

Руководитель: доцент каф. ЭВМ   
Селезнев И. Л.

Минск 2023

**СОДЕРЖАНИЕ**

[1 ОБЗОР ЛИТЕРАТУРЫ 4](#_Toc153266283)

[1.1 Требования к проектируемому устройству 4](#_Toc153266284)

[1.2 Микроконтроллер 4](#_Toc153266285)

[1.3 Катушка индуктивности 6](#_Toc153266286)

[1.4 Модуль часов реального времени 6](#_Toc153266287)

[1.5 Устройство отображения информации 7](#_Toc153266288)

[1.6 Устройство воспроизведения звука 8](#_Toc153266289)

[2 РАЗРАБОТКА СТРУКТУРЫ МИКРОПРОЦЕССОРНОГО УСТРОЙСТВА ДЛЯ ОБНАРУЖЕНИЯ МЕТАЛЛОВ 10](#_Toc153266290)

[2.1 Описание модулей 10](#_Toc153266291)

[2.2 Взаимодействие модулей 10](#_Toc153266292)

[2.3 Построение структурной схемы 11](#_Toc153266293)

[3 ОБОСНАВАНИЕ ВЫБОРА УЗЛОВ, ЭЛЕМЕНТОВ ФУНКЦИОНАЛЬНОЙ СХЕМЫ УСТРОЙСТВА 12](#_Toc153266294)

[3.1 Обоснование выбора микроконтроллера 12](#_Toc153266295)

[3.2 Обоснование выбора катушка индуктивности 13](#_Toc153266296)

[3.3 Обоснование выбора модуля часов реального времени 14](#_Toc153266297)

[3.4 Обоснование выбора информационной панели 15](#_Toc153266298)

[3.5 Обоснование выбора пьезодинамика 16](#_Toc153266299)

[3.6 Система питания 16](#_Toc153266300)

[3.7 Построение функциональной схемы 17](#_Toc153266301)

[4 РАЗРАБОТКА ПРИНЦИПИАЛЬНОЙ ЭЛЕКТРИЧЕСКОЙ СХЕМЫ УСТРОЙСТВА 18](#_Toc153266302)

[4.1 Расчет сопротивления для светодиода 18](#_Toc153266303)

[4.2 Условное обозначение разъемов микроконтроллера 18](#_Toc153266304)

[4.3 Описание подключения элементов к микроконтроллеру 20](#_Toc153266305)

[4.4 Построение принципиальной схемы устройства 21](#_Toc153266306)

[5 РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 22](#_Toc153266307)

[5.1 Требования к программе 22](#_Toc153266308)

[5.2 Разработка схемы алгоритма программы 22](#_Toc153266309)

[5.3 Разработка программы 23](#_Toc153266310)

[5.4 Описание текста программы 24](#_Toc153266311)

[ЗАКЛЮЧЕНИЕ 27](#_Toc153266312)

[ЛИТЕРАТУРА 28](#_Toc153266313)

[ПРИЛОЖЕНИЕ А 30](#_Toc153266314)

[ПРИЛОЖЕНИЕ Б 31](#_Toc153266315)

[ПРИЛОЖЕНИЕ В 32](#_Toc153266316)

[ПРИЛОЖЕНИЕ Г 33](#_Toc153266317)

[ПРИЛОЖЕНИЕ Д 34](#_Toc153266318)

[ПРИЛОЖЕНИЕ Е 35](#_Toc153266319)

[ПРИЛОЖЕНИЕ Ж 36](#_Toc153266320)

**ВВЕДЕНИЕ**

Металлоискатели широко используются в различных областях, таких как промышленность, безопасность, археология и даже хобби. Они позволяют быстро и эффективно обнаруживать металлические предметы, которые могут быть скрыты под землей, в стенах или в других местах, что может быть полезно для контроля качества продукции, безопасности на производстве, археологических исследований и хобби. В связи с этим, целью данного курсового проекта является разработка микропроцессорного устройства для обнаружения металлических объектов.

Задачи проекта – изучение принципов работы металлоискателей, разработка алгоритмов обнаружения металлических объектов, проектирование электронной схемы и создание программного обеспечения для микропроцессора.

Проектируемое устройство должно иметь следующий функционал:

– обнаружение металлических объектов разных размеров и глубин залегания;

– отображение необходимой информации;

– настройка устройства (чувствительность, громкость звука, режим работы);

– индикация состояния устройства;

Кроме того, разрабатываемое устройство должно быть достаточно легким и портативным, чтобы его можно было использовать в различных условиях.

# **1 ОБЗОР ЛИТЕРАТУРЫ**

## **1.1 Требования к проектируемому устройству**

Главной функцией проектируемого устройства является обнаружение металлических объектов. При приближении и отдалении устройства от металлов пользователь должен быть каким-либо образом уведомлен об этом.

Отличной функций является отображение текущего времени и даты.

Кроме того, устройство должно иметь элементы индикации работы устройства и признак обнаружения металлического объекта.

Также необходимо предусмотреть функцию настройки устройства (чувствительность, режим работы, настройка даты и времени).

Чтобы решить поставленные задачи, такое устройство должно иметь следующие компоненты:

– микроконтроллер;

– катушка индуктивности;

– модуль часов реального времени;

– устройство отображения информации;

– модуль пассивного динамика;

– органы управления;

– элементы индикации.

## **1.2 Микроконтроллер**

В данном проекте микроконтроллер используется для управления всеми функциями устройства, включая обработку сигналов, настройку параметров и управление устройством отображения информации.

На сегодняшний день на рынке представлен огромный выбор микропроцессоров. Для реализации устройства были рассмотрены следующие микроконтроллеры: ATmega328P, ATtiny26L и PIC16F84A.

ATmega328P – это микроконтроллер семейства AVR, основанный на архитектуре RISC. ATmega328P обладает высокой производительностью и широким набором функций, что делает его популярным выбором для различных проектов, требующих контроля и управления. Он имеет 32-кбайт флэш-памяти для программ, 2 кбайта ОЗУ и множество периферийных интерфейсов, таких как UART, SPI, I2C и GPIO. ATmega328P часто используется в Arduino-платформе и во многих других электронных устройствах.

ATtiny26L – это микроконтроллер семейства ATtiny, который обладает небольшим размером и низким энергопотреблением. ATtiny26L является 8-битным микроконтроллером с 2 кбайтами флэш-памяти, 128 байтами ОЗУ и набором периферийных интерфейсов для подключения к другим устройствам. Он предлагает достаточную функциональность для небольших проектов с ограниченными ресурсами и требованиями к энергопотреблению. ATtiny26L может быть использован в различных приложениях, таких как датчики, устройства управления и дистанционное управление.

PIC16F84A – это 8-битный микроконтроллер семейства PIC, созданный компанией Microchip Technology. PIC16F84A обладает 1 КБ флэш-памяти, 68 байтами ОЗУ и разнообразными периферийными интерфейсами, включая UART, SPI и GPIO. Он характеризуется низким энергопотреблением и простотой в использовании, что делает его популярным выбором для различных простых проектов, таких как управление светодиодами, считывание датчиков и создание электронных игр. PIC16F84A широко используется в образовательных целях и встраиваемых системах с ограниченными требованиями к ресурсам.

В таблице 1.1 приведена их характеристика, информация о которых взята из источников [1, 2, 3].

Таблица 1.1 – Характеристики микропроцессоров

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр сравнения | ATmega328P | ATtiny26L | PIC16F84A |
| Тактовая частота | 16 МГц | 16 МГц | 20 МГц |
| Flash-память | 32 КБ | 2 КБ | 2 КБ |
| SRAM-память | 2 КБ | 128 Б | 68 Б |
| EEPROM-память | 1 КБ | 128 Б | 64 Б |
| Циклы перезаписи для Flash-памяти | 10 тыс. | 10 тыс. | 10 тыс. |
| Циклы перезаписи для EEPROM-памяти | 100 тыс. | 100 тыс. | 10 млн. |
| Разрядность шины | 8-бит | 8-бит | 8-бит |
| Напряжение питания | от 2,7В до 5,5В | от 2,7В до 5,5В | от 2,0В до 5,5В |
| Температурный диапазон | от -40оС до +125oС | от -40оС до +85oС | от -40оС до +85oС |
| Количество регистров общего назначения | 32 | 32 | 68 |
| Количество векторов прерывания | 26 | 28 | 14 |
| Система команд | AVR Instruction Set | AVR Instruction Set | PIC Instruction Set |
| Таймер/счетчик | 8-бит (2 шт.),  16-бит (1 шт.) | 8-бит (1 шт.),  16-бит (1 шт.) | 8-бит (1 шт.) |
| Количество каналов ШИМ | 6 | 2 | — |
| Количество АЦП | 8 | 8 | — |
| Всего контактов | 32 | 20 | 18 шт. |
| Поддержка USART | Да | Да | Нет |
| Продолжение таблицы 1.1 | | | |
| Поддержка SPI | Да | Да | Да |
| Поддержка I2C | Да | Нет | Нет |
| Поддержка USB | Нет | Нет | Нет |

## **1.3 Катушка индуктивности**

Для поиска металлических предметов можно использовать катушку индуктивности. Она может быть сделана из разного материала, например: медная проволока, алюминиевая проволока, витая пара. Сравнительный анализ катушек индуктивности можно провести по таблице 1.2.

Катушка индуктивности, используемая в металлоискателях, работает на основе электромагнитного индукционного эффекта. Когда переменный электрический ток проходит через катушку, вокруг нее создается переменное магнитное поле.

При наличии металлического предмета рядом с катушкой, переменное магнитное поле взаимодействует с проводимыми металлом зарядами в предмете. Это приводит к возникновению в предмете индуцированного тока, который в свою очередь создает свое собственное магнитное поле.

Таким образом, металл вызывает изменение переменного магнитного поля в катушке. Эти изменения обнаруживаются детекторной системой металлоискателя и интерпретируются как наличие или отсутствие металла в области действия металлоискателя.

Таблица 1.2 – Характеристики катушек индуктивности

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр сравнения | Катушка из медной проволоки | Катушка из алюминиевой проволоки | Катушка из витой пары |
| Проводимость | 58 млн. См/м | 38 млн. См/м | 58 млн. См/м |
| Сопротивление | 17 мОм/м | 27 мОм/м | 50 – 100 мОм/м |
| Магнитная проницаемость | 0.999994 | 1.000022 | Зависит от материалов проводников |

## **1.4 Модуль часов реального времени**

Для определения текущего времени и времени поездки необходимы

часы реального времени. Наиболее распространёнными моделями часов

реального времени являются DS1302, DS1307, DS3231. Их параметры

представлены в таблице 1.4. Для составления таблицы использовался

источники [4, 5].

Часы реального времени – это устройство, которое отображает текущее

время в соответствии с глобальным стандартом времени. Они используются

во многих приборах и системах, таких как компьютеры, автомобили,

телефоны и другие устройства, которые требуют точной синхронизации

времени.

Таблица 1.4 – Параметры часов реального времени

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр сравнения | DS1302 | DS1307 | DS3231 |
| Напряжение питания | 2 – 5 В | 5 В | 3,3 – 5 В |
| Максимальный ток питания | 300 мА | 1,5 мА | 0,3 мА |
| Количество контактов | 5 | 4 | 4 |
| Точность | 5 сек/сутки | 2,5 сек/сутки | 2 – 3,5 сек/сутки |
| Максимальная частота | 32768 КГц | 100 КГц | 32768 КГц |
| Память | 32 Байтов | 56 Байтов | 4000 Байтов |
| Часовые режимы | 12, 24 | 12, 24 | 12, 24 |
| Поддерживаемые интерфейсы | I2C, SPI | I2C | I2C |

## **1.5 Устройство отображения информации**

Для реализации вывода информации с датчиков в понятной для человека форме, а также для настройки устройства необходим дисплей. Наиболее удобными и простыми в использовании для таких задач являются знакосинтезирующие жидкокристаллические дисплеи. Такие экраны могут одновременно отображать ограниченное количество символов.

В таблице 1.5 представлено три дисплея для сравнения: LCD1602, LCD2004 и LCD12864.

LCD1602 – это модуль жидкокристаллического дисплея, который состоит из 2 строк с по 16 символами в каждой. Он обеспечивает возможность отображения текстовой информации с помощью жидких кристаллов и подсветки заднего фона. Используется для вывода текстовых сообщений и данных в малых электронных устройствах, таких как адаптеры интернета вещей, микроконтроллерные системы и прочее.

LCD2004 – это модуль жидкокристаллического дисплея с расширенными характеристиками. Он состоит из 4 строк с по 20 символами в каждой, что позволяет отобразить большее количество информации. Благодаря своим возможностям по отображению текста, символов и графики, LCD2004 широко используется в проектах, требующих более объемного вывода данных, например, в промышленном оборудовании, системах мониторинга и встроенных системах.

LCD12864 – Это модуль жидкокристаллического графического дисплея с разрешением 128x64 пикселя. Он имеет возможность выводить не только текстовую информацию, но и графические элементы, такие как линии, точки и символы. Этот модуль широко используется в различных проектах, где требуется отображение динамичной информации или пользовательского интерфейса, например, в системах измерения и контроля, игровых консолях или устройствах с настраиваемым интерфейсом.

Данные о приведенных ЖК-модулях взяты из источников [6, 7, 8].

Таблица 1.5 – Характеристики ЖК-дисплеев

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр сравнения | LCD1602 | LCD2004 | LCD12864 |
| Контроллер | HD44780 | HD44780 | ST7920 |
| Габарит | 80 × 35× 11 мм | 80 × 35× 11 мм | 75 × 58 × 9 мм |
| Вес | 70 г | 70 г | 80 г |
| Разрешение | 16 × 2 символов | 20 × 4 символов | 128 × 64 пикселя |
| Размер пикселя | 2,95 × 4,35 мм | 2,95 × 4,35 мм | 2,95 × 4,35 мм |
| Размер символа | 0,5 × 0,5 мм | 0,5 × 0,5 мм | 0,5 × 0,5 мм |
| Напряжение питания | 5 В | 5 В | 5 В |
| Максимальный потребляемый ток | 120 мА | 180 мА | 250 мА |
| Интерфейс | I2C | I2C | I2C |
| Диапазон рабочих температур | от -20°С до +70°С | от -20°С до +70°С | от -20°С до +70°С |

## **1.6 Устройство воспроизведения звука**

В качестве устройства воспроизведения, оповещающего о превышении скорости, удобно использовать пьезодинамик. Для сравнения были выбраны активный и пассивный пьезодинамики. В таблице 1.6 приведены сравнительные характеристики данных моделей датчиков. Информация взята из источника [9].

Пассивный пьезодинамик – это динамик, который не имеет встроенного усилителя. Он требует подключения к внешнему усилителю для работы.

Активный пьезодинамик – это динамик, который имеет встроенный усилитель. Он может быть подключен непосредственно к источнику звука, такому как компьютер или мобильный телефон, без необходимости использования внешнего усилителя.

Таблица 1.6 – Параметры пьезодинамиков

|  |  |  |
| --- | --- | --- |
| Параметр сравнения | Активный пьезодинамик | Пассивный пьезодинамик |
| Максимальный потребляемый ток | 30 мА | 20 мА |
| Напряжение питания | 3,3 – 5 В | 3 – 12 В |
| Рабочая температура | от -20°С до +70°С | от -20°С до +70°С |
| Частоты излучения | 2,5 кГц | 2 кГц |
| Тип интерфейса | цифровой | цифровой |

# **2 РАЗРАБОТКА СТРУКТУРЫ МИКРОПРОЦЕССОРНОГО УСТРОЙСТВА ДЛЯ ОБНАРУЖЕНИЯ МЕТАЛЛОВ**

## **2.1 Описание модулей**

С учетом требований к устройству, описанных в п.1.1, можно выделить рабочие модули, из которых будет состоять проектируемое устройство:

– микроконтроллер;

– катушка индуктивности;

– модуль часов реального времени;

– информационная панель;

– органы индикации;

– органы управления;

– исполнительное устройство;

– блок питания.

## **2.2 Взаимодействие модулей**

Модуль микроконтроллера является главным управляющим компонентом проектируемого устройства. Микроконтроллер будет получать и анализировать данные от других модулей, принимать решения и контролировать работу остальных компонентов. Он будет обрабатывать сигналы, полученные от катушки индуктивности и модуля часов реального времени.

Катушка индуктивности используется для обнаружения металлических объектов в земле. Она создает переменное магнитное поле, которое будет изменяться при нахождении металла поблизости. Сигнал с катушки индуктивности будет передаваться микроконтроллеру для дальнейшей обработки.

Модуль часов реального времени является важной составляющей проекта. Он используется для точного отображения текущего времени и даты на приборе. Это полезно для пользователей, чтобы иметь представление о времени и контролировать продолжительность использования прибора.

Информационная панель будет отображать информацию о распознанных металлических объектах, текущей чувствительности и других параметрах. Микроконтроллер будет передавать соответствующие данные на информационную панель для вывода.

В состав органов индикации входят светодиоды, звуковой сигнализатор и частично дисплей на информационной панели. Они будут использоваться для отображения различных событий и предупреждений, например, обнаружении металла или изменении режима работы устройства.

К органам управления относятся кнопки, которые пользователь будет использовать для взаимодействия с металлоискателем. Они будут использоваться для включения/выключения устройства, выбора режимов работы и настройки других параметров.

Исполнительным устройством является модуль пассивного динамика. Пассивный динамик нужен для уведомления пользователя о приближении или отдалении от металлических объектов. Модуль будет получать сигналы от микроконтроллера и менять частоту воспроизведения звука.

Блок питания предоставляет энергию для работы всех модулей в устройстве. Он может быть реализован через батареи, аккумуляторы или внешний источник питания. Он обеспечивает питание для всех модулей и микроконтроллера.

## **2.3 Построение структурной схемы**

На основании полученных блоков в п.2.1 и п.2.3 построена структурная схема, которая представлена в приложении А.

# **3 ОБОСНАВАНИЕ ВЫБОРА УЗЛОВ, ЭЛЕМЕНТОВ ФУНКЦИОНАЛЬНОЙ СХЕМЫ УСТРОЙСТВА**

## **3.1 Обоснование выбора микроконтроллера**

После сравнения нескольких микроконтроллеров в п.1.2 было принято решение выбрать в качестве платы микроконтроллера Arduino UNO (см. рисунок 3.1).



Рисунок 3.1 – Изображение Arduino UNO

Arduino UNO представляет собой платформу, которая сочетает в себе простоту использования, доступность и возможность расширения. Она обладает широкой поддержкой сообщества разработчиков, что обеспечивает доступ к богатой базе знаний, библиотекам и примерам кода. Это является важным фактором, так как это значительно упрощает разработку проектов и сокращает время отладки.

При сравнении с другими микроконтроллерными платами, Arduino UNO предоставляет множество встроенных вариантов цифровых и аналоговых входов/выходов. Это позволяет легко подключать и взаимодействовать с различными сенсорами, устройствами и модулями расширения. Кроме того, возможность использования дополнительных модулей открывает новые горизонты для расширения функциональности проекта.

Дополнительным преимуществом Arduino UNO является его низкая стоимость и широкая доступность.

И, наконец, Arduino UNO предоставляет простой в использовании программный инструмент Arduino IDE. Это делает процесс программирования и отладки микроконтроллера более интуитивным и доступным даже для новичков.

Поэтому Arduino UNO является наиболее подходящей платой с микроконтроллером для данного проекта.

В таблице 3.1 приведены основные параметры Arduino UNO R3.

Для получения информации об Arduino UNO R3 использовался источник [10].

Таблица 3.1 – Характеристика Arduino UNO R3

|  |  |
| --- | --- |
| Параметр | Описание |
| Процессор | ATMega328P |
| Максимальный ток | 410 мА |
| Напряжение питания | 7 – 12 В |
| Цифровые входы/выходы | 14 шт. |
| Аналоговые входы/выходы | 6 шт. |
| Входы/выходы ШИМ | 6 шт. |
| Разрядность АЦП | 10 бит |
| Разрядность ШИМ | 8 бит |
| Интерфейсы | USART, I2C, SPI, USB |
| Габариты | 68.6 мм × 53.4 мм |
| Вес | 25 г |
| Flash-память | 32 КБ (ATMega328P) |
| SRAM-память | 2 КБ (ATMega328P) |
| EEPROM-память | 1 КБ (ATMega328P) |
| Диапазон рабочих температур | от -40°С до +85°С |

## **3.2 Обоснование выбора катушки индуктивности**

После тщательного сравнения нескольких катушек индуктивности в п.1.3 было принято решение выбрать медную катушку индуктивности для данного проекта. Такое решение основано на нескольких ключевых факторах и преимуществах, которые медь предлагает в контексте задачи обнаружения металлических предметов.

Во-первых, медь демонстрирует высокую электрическую проводимость, превосходящую проводимость алюминия и витой пары. Это означает, что медная катушка будет иметь меньшее сопротивление и позволит более эффективно передавать сигналы в металлоискателе. Это повысит его чувствительность и поможет обнаруживать даже малые металлические предметы

Во-вторых, проводимость переменного тока является критическим аспектом работы катушек индуктивности. Медь обладает лучшей способностью проводить переменный ток, что уменьшает потери и дает возможность более точно обнаруживать металлы. Скин-эффект и проникновение магнитного поля, связанные с прохождением переменного тока через катушку, у меди менее значимы, что помогает сохранить качество сигнала и минимизировать помехи.

Третий важный аспект – это магнитная проницаемость материала. Медь имеет почти такую же магнитную проницаемость, как и вакуум, что позволяет избежать искажений магнитного поля, создаваемого катушкой. Помехи и дополнительные эффекты, вызванные магнитной проницаемостью материала катушки, могут сильно повлиять на точность обнаружения металлов, поэтому выбор меди в данном случае является рациональным.

Кроме того, медь обладает прочностью и долговечностью, что является важным критерием при разработке металлоискателя. Она способна выдерживать эксплуатационные условия и противостоять механическим нагрузкам во время поиска металлических предметов на различных поверхностях. Таким образом, выбор медной катушки обеспечивает надежность и долговечность работы металлоискателя в течение продолжительного времени.

И, наконец, стоит упомянуть доступность и стоимостную эффективность меди. Медь является широкодоступным материалом, который предлагает хорошее соотношение цена-качество. Поэтому выбор медной катушки индуктивности может быть осуществлен без значительных финансовых затрат или трудностей с поиском материала.

## **3.3 Обоснование выбора модуля часов реального времени**

В пункте 1.6 были рассмотрены аналоги модулей часов реального времени DS1302, DS1307, DS3231, их параметры представлены в таблице 1.5.

Часы реального времени DS1302 имеет схожие параметры со своими аналогами, кроме того, что уступает в потребляемом токе. Однако данные часы реального времени являются низкими по цене и доступны на рынке.

Модуль часов реального времени DS1302 также имеет простой способ подключения к плате Arduino Uno (3 цифровых выхода). Также явным 17 преимуществом будет, что DS1302 имеет двойное питание – основное (от платы Arduino Uno) и запасное от батареи. Исходя из вышеперечисленных факторов, выбор пал на DS1302, их изображение представлено на рисунке 3.2.



Рисунок 3.2 – Модуль часов реального времени DS1302

## **3.4 Обоснование выбора информационной панели**

В ходе исследования различных ЖК-дисплеев, включая LCD1602 и LCD12864, было обращено особое внимание на функциональность, удобство использования и соответствие требованиям проекта. С уверенностью можно сказать, что LCD2004 стал наилучшим вариантом по нескольким ключевым причинам. Изображение LCD2004 приведено на рисунке 3.3.



Рисунок 3.3 – Изображение ЖК-модуля LCD2004

Во-первых, LCD2004 отличается относительно большим размером дисплея, что обеспечивает более комфортное отображение информации. Его 20-символьная и 4-строчная структура позволяет выводить расширенные данные и обеспечивает лучшую читаемость. Для данного курсового проекта это будет весьма полезным.

Во-вторых, LCD2004 обладает расширенными возможностями конфигурации и управления. Он поддерживает контрастную настройку и обратное освещение, что позволяет настроить оптимальные параметры для максимальной видимости и удобства работы. Также он имеет возможность программной настройки адреса, что облегчает подключение нескольких модулей к микроконтроллеру.

В-третьих, LCD2004 отличается высокой совместимостью с различными платформами и микроконтроллерами. Он совместим с широким спектром плат, включая Arduino, Raspberry Pi и многие другие. Это гарантирует простое и гибкое подключение к уже существующей системе или возможность выбора наиболее подходящей платформы для проекта.

Также, LCD2004 доступен и широко распространен в среде разработчиков, что обеспечивает доступ к качественной поддержке, документации и примерам кода. Это значительно облегчает процесс интеграции, программирования и отладки.

В свете всех этих причин и факторов, выбор ЖК-модуля LCD2004 для данного курсового проекта является обоснованным и соответствует требованиям эффективного и удобного отображения информации.

## **3.5 Обоснование выбора пьезодинамика**

В пункте 1.6 представлено описание активного и пассивного пьезодинамиков, а их параметры представлены в таблице 1.6. В качестве устройства воспроизведения звук был выбран пассивный пьезодинамик QYF-068.

Изображение пьезодинамика QYF-068 представлено на рисунке 3.4.

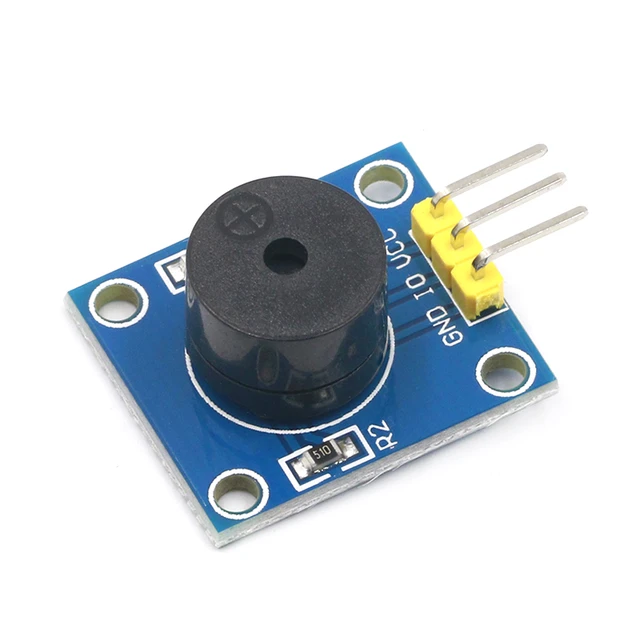


Рисунок 3.4 – Модуль пассивного пьезодинамика QYF-068

Пассивный пьезодинамик не имеет встроенного усилителя. Он требует подключения к внешнему усилителю для работы. Активный пьезодинамик уже имеет встроенный усилитель. Он может быть подключен непосредственно

к источнику звука без необходимости использования внешнего усилителя. Активные пьезодинамики обычно более дорогие и тяжелее, чем пассивные, но

они более удобны в использовании и могут предложить более высокое качество звука.

## **3.6 Система питания**

Проектируемое устройство будет питаться от одного общего источника питания. Для его выбора требуется провести анализ всех модулей устройства.

Электронные характеристики выбранных компонентов для проектируемого устройства приведены в таблице 3.2.

Таблица 3.2 – Электронные характеристики компонентов устройства

|  |  |  |  |
| --- | --- | --- | --- |
| Модуль | Uпит, В | Iмакс, мА | Pпотр, Вт |
| Плата Arduino UNO | 7,4 | 50 | 0,5 |
| Дисплей LCD2004 | 5 | 180 | 1,0 |
| Модуль DS1302 | 5 | 150 | 1,0 |
| Продолжение таблицы 3.2 | | | |
| Модуль пассивного динамика | 5 | 40 | 0,2 |
| Светодиоды (3 шт.) | 5 | 90 | 0,5 |
| Итого: | | | 3,2 |

Сделав анализ таблицы 3.2, можно сделать вывод, что для питания устройства требуется два напряжения 7,4 В и 5 В. В качестве выходного напряжения источника питания будет напряжение 7,4 В.

После получения суммарной Pпотр следует взять запас по мощности источника питания приблизительно на 20%. Таким образом, мощность источника питания должна быть равной Pпит = 3,84 Вт.

Теперь нужно рассчитать максимальный ток источника питания по формуле 3.1:

(3.1)

Далее делаем запас по току на 20% и таким образом получаем итоговый ток источника питания, равный Iист = 0,62 А.

## **3.7 Построение функциональной схемы**

Функциональная схема микропроцессорного устройства для обнаружения металлических объектов приведена в приложении Б.

# **4 РАЗРАБОТКА ПРИНЦИПИАЛЬНОЙ ЭЛЕКТРИЧЕСКОЙ СХЕМЫ УСТРОЙСТВА**

## **4.1 Расчет сопротивления для светодиода**

Для данного курсового проекта предусмотрены три светодиода различного цвета для индикации состояния устройства, которые подключаются к соответствующим входам микроконтроллера (см. рисунок 4.1).



Рисунок 4.1 – Схема подключения светодиода

Поскольку напряжение питания используемых светодиодов находится в пределах от 2 до 3 В, а цифровой выход выбранной платы Arduino UNO выдает напряжение 5 В, то нужно обеспечить падение напряжения и ограничить ток через светодиоды. Для этого нужно использовать резистор с сопротивлением, которое можно найти по формуле 4.1.

, (4.1)

где – напряжение питания, – напряжение, падающее на светодиоде, – прямой ток светодиода.

Для нашего случая = 5 В, = 2,5 В и = 30 мА. Подставляя данные значения в формулу, найдем, что сопротивление резистора 𝑅 = 125 Ом.

Резистора на 125 Ом не было найдено. Было решено использовать резисторы с сопротивлением равным 160 Ом. Из-за использования резистора с большим номиналом свет, испускаемый светодиодом, будет тусклее.

## **4.2 Условное обозначение разъемов микроконтроллера**

В разделе 3 была выбрана плата с микроконтроллером Arduino UNO. К ней будут подключаться все элементы устройства.

Выбранная плата обладает различными разъёмами подключения (см. рисунок 4.2).

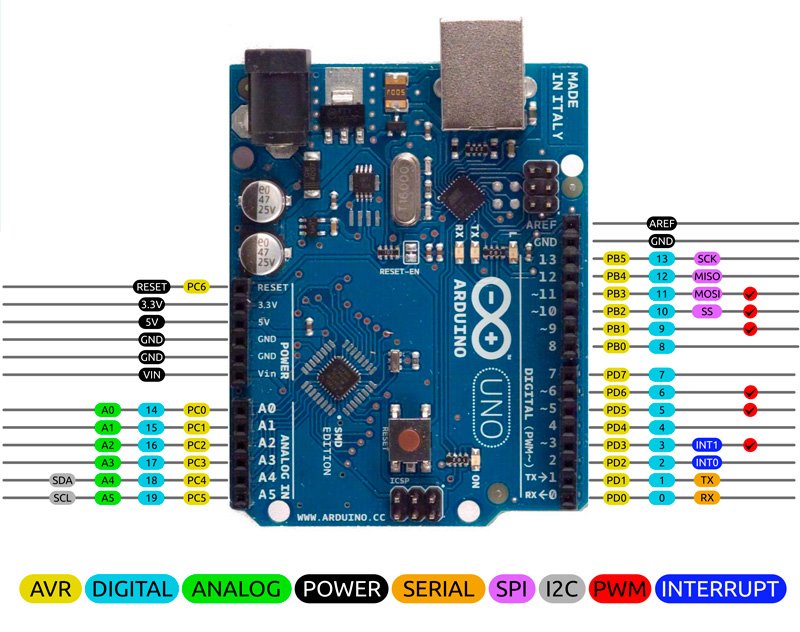


Рисунок 4.2 – Распиновка Arduino UNO

На принципиальной схеме разрабатываемого устройства необходимо заполнить таблицу соответствий названий разъёмов платы и их условными обозначениями на схеме (см. таблицу 4.1).

Таблица 4.1 – Соответствие разъёмов платы с их условными обозначениями на принципиальной схеме

|  |  |  |
| --- | --- | --- |
| Описание разъёма | Обозначение контакта на плате | Условное обозначение контакта на схеме |
| Цифровой вход/выход, Serial RX | D0 | 401 |
| Цифровой вход/выход, Serial TX | D1 | 402 |
| Цифровой вход/вывод | D2 | 403 |
| Цифровой вход/вывод, ШИМ | D3 | 404 |
| Цифровой вход/вывод, ШИМ | D4 | 405 |
| Цифровой вход/вывод, ШИМ | D5 | 406 |
| Цифровой вход/вывод, ШИМ | D6 | 407 |
| Цифровой вход/вывод | D7 | 408 |
| Цифровой вход/вывод | D8 | 409 |
| Цифровой вход/вывод, ШИМ | D9 | 410 |
| Цифровой вход/вывод, ШИМ, SPI SS | D10 | 411 |
| Цифровой вход/вывод, ШИМ, SPI MOSI | D11 | 412 |
| Цифровой вход/вывод, SPI MISO | D12 | 413 |
| Цифровой вход/вывод, SPI SCK | D13 | 414 |
| Земля для опорного напряжения (V-) | GND | 501 |
| Продолжение таблицы 4.1 | | |
| Пин опорного напряжения | AREF | 502 |
| Аналоговый пин с 8-ми битным АЦП, I2C SDA | SDA | 201 |
| Аналоговый пин с 8-ми битным АЦП, I2C SCL | SCL | 202 |
| Напряжение 5 В (V+) | 5V | 301 |
| Пин перезагрузки | RES | 305 |
| Напряжение 3,3 В | 3.3V | 306 |
| Напряжение 5 В (V+) | 5V | 307 |
| Пин питания (соединен с “+” разъема питания) | VIN | 308 |
| Земля (V-) | GND | 302 |
| Земля (V-) | GND | 303 |
| Аналоговый пин с 8-ми битным АЦП | A0 | 101 |
| Аналоговый пин с 8-ми битным АЦП | A1 | 102 |
| Аналоговый пин с 8-ми битным АЦП вход | A2 | 103 |
| Аналоговый пин с 8-ми битным АЦП | A3 | 104 |
| Аналоговый пин с 8-ми битным АЦП, I2C SDA | A4 | 105 |
| Аналоговый пин с 8-ми битным АЦП, I2C SCL | A5 | 106 |

## **4.3 Описание подключения элементов к микроконтроллеру**

Плата Arduino UNO питается от источника питания, напряжение которого подается на вход VIN.

К аналоговому входу А3 подключается выход от генератора.

Выходы дисплея LCD2004 SDA и SCL подключаются ко входам SDA и SCL соответственно. Питание же самого дисплея идет с выхода VOU модуля стабилизатора напряжения AMS1117.

Контакты GND (302) и GND (303) подключены к земле источника питания.

К цифровым разъемам D13, D12 и D11 подключается модуль часов реального времени входами CLK, DAT и RST соответственно.

К цифровым разъемам D2, D3 подключается красная кнопка SB1 для калибровки смены режимов работы, а к разъемам D10 и D9 белая и синяя кнопка (SB2 и SB3).

Цифровой вход D4 соединен с входом I/O модуля активного динамика QYF-068.

К выходам D6 – D8 подключаются светодиоды HL1, HL2, HL3 для осуществления индикации состояния устройства.

Генератор соединен с микроконтроллером через контакт D5.

Остальные разъемы микроконтроллера не используются.

## **4.4 Построение принципиальной схемы устройства**

Принципиальная схема микропроцессорного устройства для обнаружения металлических объектов представлена в приложении В, а в приложении Г представлен перечень элементов.

# **5 РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

## **5.1 Требования к программе**

Для разрабатываемого устройства требуется написать программу, которая будет иметь следующий функционал:

- считывание показаний с модуля часов реального времени;

- обработка данных с модуля часов реального времени;

- настройка даты и времени;

- настройка чувствительности устройства к металлическим объектам;

- переход через различные меню;

- смена режимов работы устройства;

- вывод всей информации на дисплей;

- воспроизведение звука на пьезодинамике;

- получение данных с генератора.

Также исполняемый файл программы не должен превышать объем флеш-памяти и объем ОЗУ микроконтроллера.

Предполагается, что будет 2 режима работы устройства: статический и динамический.

При статическом режиме по нажатию на красную кнопку запоминается текущая частота, а далее устройство издает звуковой сигнал, когда есть отклонение.

При динамическом режиме частота постоянно стремится к текущей, устройство реагирует только на резкое появление металла.

## **5.2 Разработка схемы алгоритма программы**

Схема алгоритма программы данного курсового проекта представлена в приложении Д.

Далее представлено описание блоков схемы:

1. Начало.

2. Инициализация переменных и констант.

3. Настройка режимов работы контактов.

4. Установка Timer 1 на счет.

5. Опрос синей и белой кнопок на нажатие.

6. Условный оператор: нажата ли синяя или белая кнопка? Если да, то переход на шаг 7, иначе – переход на шаг 11.

7. Условный оператор: нажата белая кнопка и либо включено главное меню, либо включено меню RTC и курсор в меню RTC указывает на строку “BACK”? Если да, то переход на шаг 8, иначе – переход на шаг 9.

8. Смена меню.

9. Условный оператор: включено ли меню RTC? Если да, то перейти на шаг 10, иначе – перейти на шаг 11.

10. Выполнить функцию в меню RTC.

11. Условный оператор: нажата ли красная кнопка? Если да, то перейти на шаг 13, иначе – перейти на шаг 12.

12. Сброс времени удержания красной кнопки. Переход на шаг 17.

13. Условный оператор: время удержания красной кнопки равно 0? Если да, то переход на шаг 16, иначе – переход на шаг 14.

14. Условный оператор: время удержания красной кнопки больше его граничного значения? Если да, то переход на шаг 15, иначе – переход на шаг 17.

15. Смена режима устройства на противоположный и сброс времени удержания красной кнопки. Переход на шаг 17.

16. Начать отсчет времени удержания красной кнопки.

17. Условный оператор: установлен ли признак готовности к изменениям? Если да, то переход на шаг 18, иначе – переход на шаг 25.

18. Условный оператор: выбран статический режим? Если да, то перейти на шаг 21, иначе – перейти на шаг 19.

19. Условный оператор: выбран динамический режим? Если да, то перейти на шаг 20, иначе – перейти на шаг 25.

20. Выполнить функцию динамического режима. Переход на шаг 22.

21. Выполнить функцию статического режима.

22. Фильтрация сигнала, настройка чувствительности.

23. Вывод информации в меню.

24. Конвертация измерения частоты в звук и сброс готовности к изменениям.

25. Условный оператор: сработал ли счетный таймер? Если да, то переход на шаг 26, иначе – переход на шаг 5.

26. Увеличить значение счетного таймера, вычисление фазы генератора звука.

27. Условный оператор: включено ли главное меню? Если да, то перейти на шаг 28, иначе – перейти на шаг 29.

28. Воспроизвести звук, включить синий светодиод, выждать задержку.

29. Выключить звук, выключить синий светодиод. Переход на шаг 5.

## **5.3 Разработка программы**

Для написания программного обеспечения выбран язык программирования C++ и интегрированная среда разработки Arduino IDE 2.1.1. При разработке программы используются следующие библиотеки:

- LiquidCrystal\_I2C;

- iarduino\_RTC.

Первая из перечисленных библиотек позволяет работать с LCD дисплеем через интерфейс I2C. Вторая библиотека используется для взаимодействия с часами реального времени.

Данные библиотеки удобны в использовании и, несомненно, облегчают разработку программного обеспечения.

## **5.4 Описание текста программы**

Текст программы для данного проектируемого устройства приведен в приложении Е.

Далее приводится пояснение исходного кода:

- строки 8-9: подключение библиотек;

- строки 14-25: константы выводов микроконтроллера (блок 1);

- строки 30-32: константы работы с пьезодинамиком (блок 1);

- строки 33-34: константы режимов работы устроства (блок 1);

- строка 35: скорость передачи данных (блок 1);

- строки 36-38: константы для работы в динамическом режиме (блок 1);

- строки 43-46: макросы установки/очистки битов (блок 1);

- строка 51: значение чувствительности устройства (блок 1);

- строка 52: время последнего срабатывания Timer 1 (блок 1);

- строка 53: таймер для динамического режима (блок 1);

- строка 54: время между прерываниями (блок 1);

- строка 55: признак готовности к изменениям (блок 1);

- строка 56: начальная частота (блок 1);

- строка 57: значение измерения частоты (блок 1);

- строка 58: частота генератора звука (блок 1);

- строка 59: фаза генератора звука (блок 1);

- строка 60: счетный таймер (блок 1);

- строка 61: текущее меню (блок 1);

- строка 62: режим работы устройства (блок 1);

- строка 63: положение курсора в меню RTC (блок 1);

- строка 64: время начала удержания красной кнопки (блок 1);

- строка 65: объявление объекта дисплея LCD (блок 1);

- строка 66: объявление объекта модуля RTC DS1302 (блок 1);

- строки 75-78: инициализация объектов дисплея и модуля RTC (блок 2);

- строка 79: настройка режимов работы контактов микроконтроллера (блок 3);

- строка 80: отображение меню (блок 4);

- строки 83-85: установка Timer 1 на счет (блок 4);

- строки 92-159: главная бесконечная функция программы;

- строка 94: получение нажатой кнопки с помощью функции getPressedButton() (блок 5);

- строки 95-104: условный оператор (блок 6) – если была нажата кнопка, то, проверяем следующие условия (блок 7 и 11);

- строки 97-101: условный оператор (блок 7) – если соответствует смене меню, то выполняем функцию switchMenu() (блок 8), а иначе проверить следующее условие (блок 9);

- строки 102-103: условный оператор (блок 9) – если включено сейчас меню RTC, то выполнить функцию timeMenu() (блок 10), иначе перейти на блок 11;

- строки 105-117: условный оператор (блок 11) – если нажата красная кнопка, то проверяем условия удержания кнопки, а иначе сбросить время удержания кнопки (блок 12);

- строки 107-108: условный оператор (блок 13) – если время удержания красной кнопки равно нулю, то начать отсчет времени удержания кнопки (блок 16), а иначе перейти на блок 14;

- строки 111-115: условный оператор (блок 14) – если время удержания больше граничного значения, то сменить режим и сбросить время удержания кнопки (блок 15). Иначе – перейти на следующий условный оператор (блок 17);

- строки 121-140: условный оператор (блок 17) – если установлен признак готовности к изменениям, то перейти к выбору режима устройства (блок 18), иначе перейти к проверке срабатывания счетного таймера (блок 25);

- строки 123-124: условный оператор (блок 18) – если выбран статический режим, то выполнить функцию staticMode() (блок 21), а иначе проверить динамический режим (блок 19);

- строки 125-126: условный оператор (блок 19) – если выбран динамический режим, то выполнить функцию dynamicMode() (блок 20), иначе – перейти на блок 22;

- строки 127-128: фильтрация сигнала, получение чувствительности устройства (блок 22);

- строки 129-134: вывод информации в главное меню о частоте, чувствительности, а также вывод шкалы приближения к металлу (блок 23);

- строки 135-140: конвертация изменения частоты в звук, сброс готовности к изменениям (блок 24);

- строки 142-158: условный оператор (блок 25) – если сработал счетный таймер, то увеличиваем значение счетного таймера, вычисляем фазу генератора звука (блок 26), а иначе – переход на начало цикла программы;

- строки 149-154: условный оператор (блок 27) – если выбрано главное меню, то воспроизвести звук и включить синий светодиод, затем ожидать паузы (блок 28);

- строки 156-157: выключение пьезодинамика и синего светодиода (блок 29);

- строки 162-168: обработчик прерывания от Timer 1;

- строки 171-199: функция setupPins() для установки режима работы контактов (блок 3);

- строки 202-205: функция absf() для получения модуля вещественного числа (блок 22);

- строки 208-213: функция статического режима staticMode() (блок 21);

- строки 216-227: функция динамического режима dynamicMode() (блок 20);

- строки 230-243: функция смены меню switchMenu() (блок 8);

- строки 246-259: функция отображения главного меню showMainMenu() (блок 4, 8);

- строки 261-267: функция вывода частоты printFrequency() (блок 23);

- строки 270-276: функция вывода чувствительности printSensitivity() (блок 23);

- строки 279-289: функция вывода шкалы приближения металлических объектов printMetalScale() (блок 23);

- строки 291-309: функция отображения меню RTC showTimeMenu() (блок 8);

- строки 312-340: функция меню RTC timeMenu() (блок 10);

- строки 343-349: функция вывода времени printTime() (блок 10);

- строки 352-358: функция вывода даты printDate() (блок 10);

- строки 361-380: функция установки часов setHours() (блок 10);

- строки 382-401: функция установки минут setMinutes() (блок 10);

- строки 404-423: функция установки дня месяца setMonthDay() (блок 10);

- строки 426-445: функция установки месяца setMonth() (блок 10);

- строки 448-467: функция установки года setYear() (блок 10);

- строки 469-488: функция установки дня недели setWeekDay() (блок 10);

- строки 491-501: функция получения нажатой кнопки getPressedButton() (блок 5).

# **ЗАКЛЮЧЕНИЕ**

В результате работы было разработано микропроцессорное устройство для обнаружения металлических объектов. Данное устройство способно обнаруживать металлические объекты. Также в устройство был добавлен модуль часов реального времени, с помощью которого можно получить информацию о текущей дате или времени. Кроме того, осуществляется анализ полученных значений и соответствующая индикация показателей.

Таким образом, данный проект был спроектирован в соответствии с поставленными задачами, весь функционал был реализован в полном объеме.

Разработанное микропроцессорное устройство имеет следующие достоинства: относительно низкая стоимость, простота реализации и сборки.

В дальнейшем возможно усовершенствование данного курсового проекта, путем повышения чувствительности, точности и стабильности устройства, оптимизация программного обеспечения и обновление пользовательского интерфейса программы.

# **ЛИТЕРАТУРА**

[1] Документация ATmega328P [Электронный ресурс]. – Электронные данные. – Режим доступа : https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P\_Datasheet.pdf. Дата доступа : 12.10.2023.

[2] Документация ATtiny26L [Электронный ресурс]. – Электронные данные. – Режим доступа : https://ww1.microchip.com/downloads/en/DeviceDoc/1477S.pdf. Дата доступа : 12.10.2023.

[3] Документация PIC16F84A [Электронный ресурс]. – Электронные данные. – Режим доступа : https://ww1.microchip.com/downloads/en/devicedoc/35007b.pdf. Дата доступа : 12.10.2023.

[4] Документация модуля реального времени DS1302 [Электронный ресурс]. – Электронные данные. – Режим доступа : https://iarduino.ru/lib/f41e71c14462276c0a2a495372ba7689.pdf. Дата доступа : 06.11.2023.

[5] Сравнение часов реального времени [Электронный ресурс]. – Электронные данные. – Режим доступа : https://arduinomaster.ru/datchiki-arduino/arduino-chasy-rtc-ds1307-ds1302-ds3231/. Дата доступа : 06.11.2023.

[6] Документация LCD1602 [Электронный ресурс]. – Электронные данные. – Режим доступа : https://waveshare.com/datasheet/LCD\_en\_PDF/LCD1602.pdf. Дата доступа : 07.11.2023.

[7] Документация LCD2004 [Электронный ресурс]. – Электронные данные. – Режим доступа : https://cdn-shop.adafruit.com/datasheets/TC2004A-01.pdf. Дата доступа : 07.11.2023.

[8] Документация LCD12864 [Электронный ресурс]. – Электронные данные. – Режим доступа : https://wаveshare.com/datasheet/LCD\_en\_PDF/LCD12864-ST.pdf. Дата доступа : 07.11.2023.

[9] Статья о пьезодинамиках [Электронный ресурс]. – Электронные данные. – Режим доступа : https://www.hackster.io/baqwas/piezo-beeps-buzzers-b0ca43. Дата доступа : 07.11.2023.

[10] Документация Arduino UNO R3 [Электронный ресурс]. – Электронные данные. – Режим доступа : httрs://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf. Дата доступа : 20.11.2023.

[11] Документация переменного резистора 16К1 [Электронный ресурс]. – Электронные данные. – Режим доступа : https://belchip.by/sitedocs/16k1.pdf. Дата доступа : 01.12.2023.

[12] Документация транзистора 2N2219A [Электронный ресурс]. – Электронные данные. – Режим доступа : https://static.chipdip.ru/lib/704/DOC011704140.pdf. Дата доступа : 29.11.2023.

[13] Статья датского инженера Dzl [Электронный ресурс]. – Электронные данные. – Режим доступа : https://dzlsevilgeniuslair.blogspot.com/2013/07/diy-arduino-based-metal-detector.html. Дата доступа : 29.11.2023.

[14] Статья русского инженера Alex Gyver [Электронный ресурс]. – Электронные данные. – Режим доступа : https://alexgyver.ru/metall-detector-1/. Дата доступа : 05.12.2023.

[15] Дубровский, С. Л. Как собрать металлоискатели своими руками / С. Л. Дубровский. – СПб: Наука и Техника, 2010. – 256 с.

[16] Форум металлодетекторов [Электронный ресурс]. – Электронные данные. – Режим доступа : http://md4u.ru/viewforum.php?f=94. Дата доступа : 07.12.2023.

[17] Статья про металлоискатель на mastervintik.ru [Электронный ресурс]. – Электронные данные. – Режим доступа : https://www.mastervintik.ru/metalloiskatel-pirat-na-arduino-svoimi-rukami/. Дата доступа : 07.12.2023.

# **ПРИЛОЖЕНИЕ А**

**(обязательное)**

**Схема структурная**

# **ПРИЛОЖЕНИЕ Б**

**(обязательное)**

**Схема функциональная**

# **ПРИЛОЖЕНИЕ В**

**(обязательное)**

**Схема принципиальная**

# **ПРИЛОЖЕНИЕ Г**

**(обязательное)**

**Перечень элементов**

# **ПРИЛОЖЕНИЕ Д**

**(обязательное)**

**Схема алгоритма программы**

# **ПРИЛОЖЕНИЕ Е**

**(обязательное)**

**Код программы**

# **ПРИЛОЖЕНИЕ Ж**

**(обязательное)**

**Ведомость документов**