|  |  |
| --- | --- |
| *Описание: voenmeh* | МИНОБРНАУКИ РОССИИ  федеральное государственное бюджетное образовательное учреждение  высшего образования  **«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»**  **(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова»)** |
| БГТУ.СМК-Ф-4.2-К5-01 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Факультет |  | И |  | Информационные и управляющие системы |
|  |  | шифр |  | Наименование |
| Кафедра |  | И9 |  | Систем управления и компьютерных технологий |
|  |  | шифр |  | Наименование |
| Дисциплина |  | Технологии программирования | | |

КУРСОВАЯ РАБОТА

на тему:

|  |
| --- |
| Разработка программного обеспечения для главного |
| модуля автоматизированной системы управления |
| домашней фермы |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент группы | | | |  | | И9М33 |
| Перминова А.А. | | | | | | |
| Фамилия И.О. | | | | | | |
| **РУКОВОДИТЕЛЬ** | | | | | | |
| Арсеньев Б.П. | |  |  | | | |
| Фамилия И.О. Подпись | | | | | | |
| Оценка |  | | | |  | |
| «\_\_\_\_\_» |  | | | | 20\_\_\_\_ г. | |

САНКТ-ПЕТЕРБУРГ

2017 г.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc500240481)

[1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 5](#_Toc500240482)

[2. ПРЕКТОРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 7](#_Toc500240483)

[3. РЕАЛИЗАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 10](#_Toc500240484)

[4. ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 13](#_Toc500240485)

[5. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 14](#_Toc500240486)

[6. ВЫВОДЫ 17](#_Toc500240487)

[ЗАКЛЮЧЕНИЕ 18](#_Toc500240488)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 19](#_Toc500240489)

# ВВЕДЕНИЕ

В наше время для автоматизации различных процессов используются микропроцессоры и микроконтроллеры. Они являются широко распространенными, даже среди простых обывателей, но также и незаметным элементом повседневной жизни. Повседневно человек постоянно сталкивается микроконтроллерами и микропроцессорами. Только все это воспринимается нами, как «умные» устройства. Это используемые в повседневной жизни вещи, такие как микроволновки или стиральные машины. А так же микроконтроллеры и микропроцессоры используются и в управлении более сложных технических средств, это автомобили, самолеты, спутники, 3D-принтеры [1].

Автоматизация коснулась и таких сфер жизни, как домашнее хозяйство и разведение животных. Благодаря доступности, всех материалов, датчиков и микроконтроллеров, появилась возможность автоматизировать некоторые процессы разведения и содержания животных, в рамках небольшого домашнего хозяйства. Таким образом, повышается эффективность разведения и выращивания животных. Использование автоматических систем позволяет экономить бюджет, время и силы, затрачиваемые на уход за животными, улучшает качество жизни животных, тем самым увеличивая продолжительность жизни [2].

Некоторые процессы, которые могут быть автоматизированы: климат контроль (поддержание оптимальный температуры и влажности), световой режим (включение и выключение света в определенное время), процесс кормления, уборка вольеров.

Процессы, подлежащие автоматизации, зависят от вида животного и запросов владельца фермы. Для этих целей используются сложные системы, состоящие из отдельных модулей, для автоматизации разных процессов, находящиеся под управлением одного главного модуля, благодаря которому осуществляется взаимодействие с пользователем. Основная проблема в том, что главный модуль, в разных системах, может кардинально различаться, в зависимости от количества дополнительных модулей и функций, предоставляемых пользователю. В то время, как периферийные модули более однообразны, приложения и готовые решения находятся в свободном доступе.

Целью курсовой работы является разработка программного обеспечения для главного модуля автоматизированной домашней фермы, в которой автоматизированы такие процессы как: световой режим, климат-контроль, кормление, обеспечение животных питьевой водой, уборка вольеров[3].

В данной работе освещено создание следующих функций:

1. Взаимодействие главного модуля с пользователем через меню;
2. Взаимодействие главного модуля с пользователем удаленно;
3. Управление и взаимодействие с модулями системы.

# АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

В данной курсовой работе представлена разработка программного обеспечения для главного модуля автоматизированной системы. В данной системе используется смешанная архитектура аппаратных средств, так как у каждого модуля, будь то основной или периферийный, существует определенный набор датчиков и исполнительных устройств, но в то же время все модули подключены к одной сетевой шине, по которой главный контроллер опрашивает второстепенные. Таким образом, можно уловить признаки централизованной и распределенной аппаратной архитектуры [4]. Такая структура позволяет создать более сложную систему, повысить скорость обработки информации, снизить расходы на техническую реализацию подключаемых датчиков и исполнительных устройств, а также, при необходимости, ее довольно просто расширить за счет общей сетевой шины.

Архитектура системы управления фермой представлена на рисунке 1.

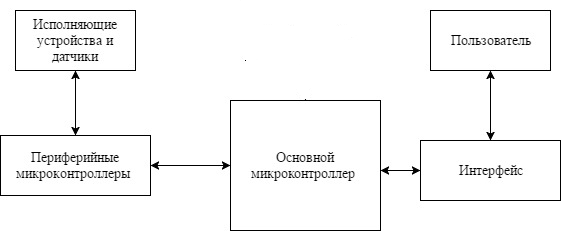


Рисунок 1 – Архитектура разрабатываемой системы

Данная система состоит из нескольких модулей: светового режима, климат-контроля, кормления, обеспечения животных питьевой водой, уборки вольеров и основного модуля, централизованного управления. Все модули представляют собой автоматизированную подсистему, состоящую из управляющего микроконтроллера ARDUINO UNO и набора управляющих элементов и датчиков. Рассмотрим, как работают периферийные модули каждый отдельно.

Модули освещения и уборки работают только по входящим сигналам от главного модуля. В зависимости от полученного сигнала, принимается решение о включении или выключения одного из режимов освещения, в случае с модулем уборки производится уборка помещения или прервать уборку. Освещение имеет 2 режима: откорм и стандарт. После выполнения одного из действий модули отправляют код завершения действия, главному модулю: успешно или ошибка, для модуля уборки еще сообщение «прервано».

Модуль климат-контроля получает сигналы от основного модуля и данные с датчиков. С помощью главного модуля задаются интервалы температуры и влажности, которые необходимо поддерживать. В зависимости от поступающей информации, а также от заданных интервалов включаются те или иные управляющие элементы, реле. За счет включения и выключения отопления, кондиционера, вытяжки и увлажнителя происходит регуляция микроклимата в помещении. Для этого модуля есть несколько управляющих сигналов: задать параметр, игнорировать интервалы параметров, включение и выключение отопления/вытяжки/увлажнителя/кондиционера, не игнорировать интервалы параметров. На выход имеются сигналы об успешности работы каждого элемента модуля, успешно или ошибка, а также дополнительные сигналы: слишком жарко – система не справляется, слишком холодно – система не справляется, повышенная влажность – система не справляется, слишком сухой воздух – система не справляется. По запросу от главного модуля передает данные о текущих показаниях температуры и влажности.

Модуль кормления получает данные с датчиков расстояния и веса, а так же сигналы от главного модуля, обрабатывает полученную информацию. После обработки отправляются сигналы на управляющие устройства, на шаговые двигатели и сервоприводы. Данный модуль принимает от главного модуля такие сигналы, как: задать вес порции для одного конкретного животного, задать вес порции для всех один, задать время кормления и частоту кормления для всех и для одного конкретного животного, включить кормление всех животных, включить кормление одного конкретного животного, прервать процесс кормления. Все процессы сопровождаются сигналами об успешной работе или об ошибке, также имеются такие сигналы как «корма осталось на одно кормление», «корм в бункере хранения закончился».

Модуль обеспечения водой работает по внутреннему счетчику, обновляя воду в поилках каждые 24 часа. На главный модуль отправляются сигналы о состоянии системы. А также принимаются сигналы о включении или отключения модуля, о запуске механизма вне заданного расписания.

Все сигналы имеют свои идентификаторы, благодаря которым, идет распознавание, от какого модуля пришли данные.

У главного модуля, для взаимодействия с пользователем, имеется дисплей с клавиатурой.

# ПРЕКТОРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

На основе анализа предметной области можно составить таблицу, которая будет наиболее полно отображать, какие функции были реализованы, и с помощью чего они были решены.

Ниже, в Таблице 1, представлены данные функции и решения.

Таблица 1 — Проводимые работы

|  |  |  |
| --- | --- | --- |
| **№** | **Наименование работ** | **Результат** |
| 1 | Реализация взаимодействия с пользователем через меню посредством клавиатуры из 5 кнопок. | Модуль меню. |
| 2 | Реализация задания параметров для периферийных модулей. | Модуль задания и передачи параметров для подсистем кормления, освещения и климат-контроля. |
| 3 | Реализация обработки сигналов от периферийных модулей | Модуль обработки сообщений подсистем. |
| 4 | Реализация обработки сигналов от пользователя: включение и выключения периферийных модулей и их функций, вне расписания и в случае внештатной работы. | Модуль обработки сообщений пользователя. |

Для разработки архитектуры программного обеспечения, целесообразно использовать шаблон трехслойной архитектуры [5].

Представим основные высказывания по каждому слою архитектуры.

1. Слой представления: предоставляет услуги по оповещению пользователя о работе системы (изменение температуры, отсутствие корма для животных), услуги по вводу параметров.

2. Слой предметной области: выполняет вычисления на основе вводимых и хранимых данных, проверку всех элементов данных и обработку команд, поступающих от слоя представления, а также передачу информации слою источника данных.

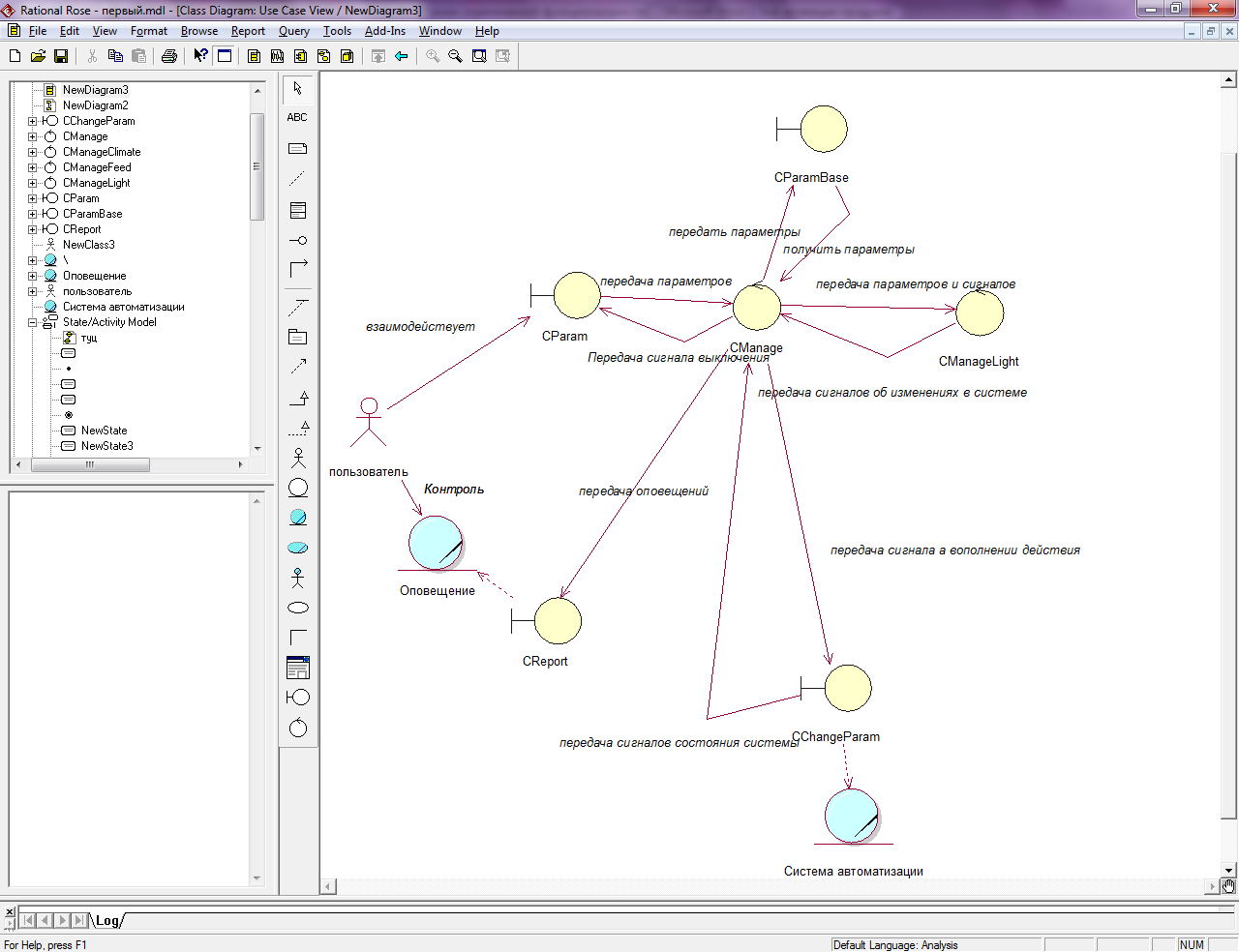
3. Слой источника данных: выполняет обращения к элементам и к их свойствам.

В таблице 2 предоставлено назначения классов концептуальной модели ПО [6].

Таблица 2 — Назначение классов концептуальной модели программного обеспечения

| **№** | **Наименование слоя** | **Необходимые высказывания** |
| --- | --- | --- |
| Слой представления | | |
|  | CParam | Граничный класс предоставления интерфейса пользователя. |
|  | CReport | Граничный класс, отвечающий за оповещение в электронном виде. |
| Слой предметной области | | |
|  | CManage | Управляющий класс, отвечающий за управление микроконтроллером. Получение параметров, передача параметров модулям, формирование сигналов оповещения. |
|  | CManageLight | Управляющий класс, отвечающий за взаимодействие системы с модулем освещенности. Передача параметров, получение сообщений об изменениях в системе для оповещения пользователя. |
|  | CManageFeed | Управляющий класс, отвечающий за взаимодействие системы с модулем кормления. Передача параметров, получение сообщений об изменениях в системе для оповещения пользователя. |
|  | CManageClimate | Управляющий класс, отвечающий за взаимодействие системы с модулем климат-контроля. Передача параметров, получение сообщений об изменениях в системе для оповещения пользователя. |
|  | CParamBase | Класс хранения, содержащий данные параметров системы. |
| Слой источника данных | | |
|  | CChangeParam | Граничный класс, выполняет обращения к системе, отвечает за получение, изменение и отправку параметров. |

На рисунке 2 представлена диаграмма классов разрабатываемого ПО. Здесь CManageFeed и CManageClimate не указаны на диаграмме так как аналогичны CManageLight.

 Рисунок 2 **—** Диаграмма классов ПО

# РЕАЛИЗАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В качестве языка реализации были выбраны язык программирования устройств Ардуино, основанный на C/C++ и скомпонованный с библиотекой AVR Libc, что позволяет использовать любые ее функции. Вместе с тем он прост в освоении, и на данный момент Arduino — это, пожалуй, самый удобный способ программирования устройств на микроконтроллерах.[7]

В качестве среды разработки выбрана Arduino IDE[8], это бесплатная официальная среда программирования «Arduino IDE», работающая под Windows, Mac OS и Linux. Среда разработки представляет собой текстовый редактор программного кода, область сообщений, окно вывода текста(консоль), панель инструментов и несколько меню. Для загрузки программ и связи среда разработки подключается к аппаратной части Arduino. Также имеется веб-версия данной среды разработки.

Загрузка исполняющего кода на микроконтроллер производится посредством подключения Ардуино к компьютеру, через порт USB.

Далее рассмотрим реализацию по каждому пункту из таблицы 1.

1. Реализация взаимодействия с пользователем через меню посредством клавиатуры из 5 кнопок.

Результатом работы, над этим пунктом, стал модуль, написанный на выбранном языке программирования, реализующий работу меню системы. После включения модуля, включается режим ожидания нажатия кнопок:

int PushButton(){

int state = 0;

int x = analogRead (0);

lcd.setCursor(0,0);

if (x < 100) { //Right

state = 1;

} else if (x < 200) //Up

state = 2;

} else if (x < 400){//Down

state = 3;

} else if (x < 600){//Left

State = 4;

} else if (x < 800){//Select

state = 5;}

return state;

}

Данный режим, включается часто, так чтобы отследить нажатия клавиш и сделать соответствующее нажатию действие.

При вызове функции главного меню на экране отображается список: установка, включение и выход:

void MainMenu(){

int state = 0;

int currentMenuItem = 0;

int x = analogRead (0);

lcd.setCursor(0,0);

state = PushButton();

currentMenuItem = StateMenu(state, 2,0,true);

if(currentMenuItem == 0){

SetMenu();

}

else if(currentMenuItem == 1){

On\_Off\_Menu();

}

else if(currentMenuItem == 2){

ExitMenu();

}

}

При выборе установки появляются вариант:

• установить время;

• установить порцию;

• установить время кормления

• установить интенсивность освещения;

• установить время включения и выключения освещения.

При выборе пункта включения возможные варианты:

• уборка;

• кормление;

• включение освещения;

• выключение освещения.

Меню «установить порцию» позволяет задать размер порции для всех одинаковую или только для одной кормушки, а меню «установить интенсивность освещения» имеет два подпункта: «Нормальный режим» и «Режим откорма».

Навигация меню реализована с помощью пяти кнопок: «Set», «Up», «Down», «Left» и «Right». При входе в меню, на дисплее отображается один из пунктов и подпунктов меню. С помощью кнопок «Up» и «Down» происходит перемещение между пунктами. Для выбора одного из вариантов, этот вариант должен быть отображен на дисплее, подтверждение выбора происходит нажатием на кнопку «Set». Таким образом, происходит использование основных пунктов меню. Так же имеются такие подпункты, в которых необходимо задавать числовое значение (выставить время, указать вес порции и прочее). Выбор числового значение происходит с помощью кнопок «Left» и «Right», от 0 и до заданного программой максимума. При выборе значения, подтверждение происходит так же с помощью кнопки «Set».

1. Реализация задания параметров для периферийных модулей.

Модуль, для решения данной задачи, связан с предыдущим модулем. Различные параметры задаются с помощью меню главного модуля, и сохраняются в переменных данного модуля. Далее идет формирование сигнала, состоящий из номера модуля, которому предается сигнал, тип сигнала, что это именно параметр, а не служебный сигнал, и самого параметра. Параметры хранятся до тех пор, пока не будут снова изменены.

1. Реализация обработки сигналов от периферийных модулей

Данный модуль, принимает сигналы от подсистем и обрабатывает их. Сигналы могут быть двух типов: служебные и сигналы от датчиков. Служебные сигналы имеют больший приоритет, в случае внештатной работы системы, с помощью данного модуля, формируются соответствующие сообщения, которые отображаются на дисплее главного модуля. В случаях же получения сигналов от датчиков, система сохраняет их в переменных для дальнейшей обработки или отображения на дисплее по желанию пользователя.

1. Реализация обработки сигналов от пользователя: включение и выключения периферийных модулей и их функций, вне расписания и в случае внештатной работы.

Модуль обработки сигналов от пользователя связан с модулем меню. С помощью него отправляются сигналы о запуске или прерывании различных процессов. Периферийная подсистема получает сигнал игнорировать заданное расписание, после производится остановка процесса или запуск, в зависимости от сигнала. Далее производится отмена сигнала игнорирования, и в случае если процесс должен был произойти в ближайшее время, сигнал игнорирования будет отменен только после наступления времени запуска процесса.

# ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

При тестировании данного программного обеспечения было использовано функциональное тестирование. Функциональное тестирование проводится для определения, насколько компонент или система соответствуют заданным функциональным требованиям, описанным в спецификациях.

Данный вид тестирования может проводиться на всех уровнях тестирования: компонентом, интеграционном, системном и приемочном, т.е. на всех этапах разработки программного обеспечения[9].

В зависимости от степени доступа к коду можно выделить два типа функциональных испытаний:

- тестирование «черный ящик» – проведение функционального те-стирования без доступа к коду,

- тестирование «белый ящик» – функциональное тестирование с доступом к коду [10].

Тестирование «черный ящик» — это стратегия или метод тестирования, базирующейся только лишь на тестировании по функциональной спецификации и требованиям, при этом не смотря во внутреннюю структуру кода и без доступа к базе данных. Фактически мы знаем, какой должен быть результат при определенном наборе данных, которые подаются на вход. Результат проверяем с пользовательского интерфейса на уровне простого пользователя. На данный момент такая стратегия является наиболее часто применима в IT компаниях [10].

Преимущества функционального тестирования:

- имитирует фактическое использование ПО;

Недостатки функционального тестирования:

- возможность упущения логических ошибок в программном обес-печении;

- вероятность избыточного тестирования.

Тестовые задания для тестирования по типу «черный ящик» приведены ниже.

1. Задание параметров для периферийных модулей.

Входные данные: нажатие кнопок, выбор с помощью них необходимого пункта меню и установка параметров.

Ожидаемый результат: система реагирует на нажатие кнопок, после установки параметров отображено сообщение об успешной передаче параметров необходимому модулю.

Результат тестирования программного продукта: система реагирует на нажатие кнопок, после установки параметров отображено сообщение об успешной передаче параметров необходимому модулю.

1. Включение процесса вне расписания.

Входные данные: нажатие кнопок, выбор с помощью них необходимого пункта меню и включения процесса.

Ожидаемый результат: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит включение процесса, на дисплее отображается сообщение об успешном выполнении операции.

Результат тестирования программного продукта: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит включение процесса, на дисплее отображается сообщение об успешном выполнении операции.

1. Прерывание процесса.

Входные данные: прерываемый процесс выполняется в данный момент, поступают сигналы от нажатия кнопок, выбор с помощью них необходимого пункта меню и прерывания процесса.

Ожидаемый результат: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит прерывание процесса, подсистема приходит в состояние ожидания следующего процесса и приема сообщений от главного модуля, на дисплее отображается сообщение об успешном выполнении операции.

Результат тестирования программного продукта: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит прерывание процесса, подсистема приходит в состояние ожидания следующего процесса и приема сообщений от главного модуля, на дисплее отображается сообщение об успешном выполнении операции.

1. Выключение периферийного модуля

Входные данные: выключаемый модуль работает, поступают сигналы от нажатия кнопок, выбор с помощью них необходимого пункта меню, выключение периферийного модуля.

Ожидаемый результат: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит выключение выбранного модуля, на дисплее отображается сообщение об успешной завершении процесса.

Результат тестирования программного продукта: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит выключение выбранного модуля, на дисплее отображается сообщение об успешной завершении процесса.

1. Включение периферийного модуля

Входные данные: включаемый модуль не работает, поступают сигналы от нажатия кнопок, выбор с помощью них необходимого пункта меню включение периферийного модуля.

Ожидаемый результат: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит включение выбранного модуля, на дисплее отображается сообщение об успешной завершении процесса.

Результат тестирования программного продукта: система реагирует на нажатие кнопок, после выбора необходимого пункта меню, происходит включение выбранного модуля, на дисплее отображается сообщение об успешной завершении процесса.

В ходе функционального тестирования ошибок в работе программного продукта обнаружено не было, тестирование пройдено успешно.

# РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

* 1. Общие сведения

Главный модуль, автоматизированной системы домашней фермы, предназначен для централизованного управления и взаимодействия с автоматизированной системой. На рисунке 3 представлен модуль в виде дисплея и клавиатуры.

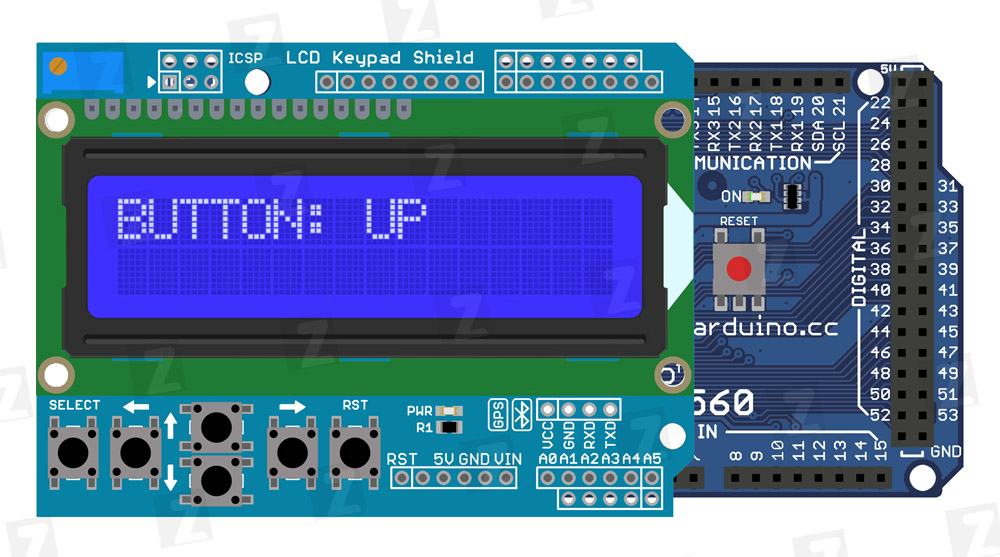


Рисунок 3 - Модуль для взаимодействия с пользователем

Как видно по рисунку дисплей может отображать 2 строки текста по 16 символов. Для управления имеется 5 кнопок: «SET», «UP», «DOWN», «LEFT» и «RIGHT». На рисунке № данные кнопки обозначены соответствующими стрелками и надписью «SELECT». Шестая кнопка необходима для перезагрузки, включения и выключения главного модуля. Данный модуль имеет два режима работы: ожидание и меню

Для пролистывания отображаемой информации используйте кнопки «UP», «DOWN», для выбора одного из пунктов меню или подтверждения набранной информации используйте кнопку «SET». У главного модуля есть функция задания некоторых параметров в виде числовых значений. Значения этих параметров изменяются с помощью кнопок «LEFT» и «RIGHT» в диапазоне возможных значений. Если на дисплее отображается вопрос, то нажатие кнопки «LEFT» соответствует ответу «да», а кнопки «RIGHT» - «нет».

* 1. Включение и выключение модуля, начало работы

После подключения системы к питанию происходит инициализация системы, в результате которой, на дисплее отображаются сообщения о состоянии каждого модуля и их готовности к работе.

В случае штатной работы, на дисплее отображается сообщение: «Система запущена впервые или была сброшена, необходимо задать необходимые параметры» или «Система требует настройки параметров?». В случае, когда отображается первое сообщение необходимо нажать кнопку «Set», далее будет отображен список как в пункте «установка» главного меню (см. пункт 5.5 Режим установки). Если при запуске системы отображается второе сообщение с вопросом, то для настройки системы необходимо нажать кнопку «LEFT», включается режим установки как при первом запуске системы, иначе - «RIGHT». В последнем случае система перейдет в режим ожидания.

Для того, что бы выключить систему, необходимо нажать кнопку «Reset», отмеченную на модуле как «RST». На экране дисплея отобразиться вопрос о подтверждении выключения, нажмите кнопку «Reset» повторно. После данных действий возможно отключение системы от питания.

Для перезагрузки системы необходимо зажать кнопку «Reset» на 2-3 секунды.

* 1. Работа в режиме ожидания

Режим ожидания включается при выходе из режима меню и автоматически, если 4-6 минуты не было произведено никаких нажатий на копки, которые находятся на панели. В данном режиме, на дисплее отображаются дата и время. При нажатии кнопок «UP» и «DOWN» производится смена отображаемой информации: заданный температурный диапазон и текущая температура, заданный диапазон влажности и текущая влажность, номер модуля и его режим работы. При возникновении ошибок в работе системы, на дисплее будет отражено соответствующее сообщение.

* 1. Работа в режиме «Меню»

Для включения данного режима, необходимо нажать кнопку «SELECT» и удерживать в течение 3-х секунд. Далее на дисплее отобразиться три пункта меню, перемещаться между которыми можно с помощью кнопок «Up» и «Down». Для выбора одного из пунктов меню необходимо, чтобы он отображался на дисплее в данный момент, подтвердите свой выбор нажатием кнопки «Set». Далее работа с пунктами и подпунктами меню идет по тому же принципу: необходимый пункт должен отображаться на дисплее, подтверждение выбора с помощью кнопки «Set».

Главное меню имеет такую иерархию.

* + - Установка
      * Дата и время
      * Порция
        + Для одного
        + Для всех
      * Время кормления
        + Для одного
        + Для всех
      * Время работы освещения
        + Включение
        + Выключение
      * Режим работы освещения
        + Обычный
        + Откорм
    - Вкл/Выкл
      * Модуль
        + 1 – Кормления
        + 2 – Уборка
        + 3 – Освещение
        + 4 – Климат-контроль
        + 5 – Обеспечение питьевой водой
      * Процесс
        + Уборка
        + Процесс выдачи порции

Для одного

Для всех

* + - * + Вытяжка
      * Обогреватель
      * Кондиционер
      * Увлажнитель
      * Освещение
  1. Режим установки

При выборе пункта «установка» в главном меню или при необходимости задания начальных параметров, после запуска системы, система переходит в данное подменю. С помощью данного режима можно задавать такие параметры как: дата и время, размер порции для одного животного или для всех сразу, время кормления для одного животного и для всех сразу, установка режима и времени работы освещения.

Установка времени для системы, времени кормления и времени работы освещения работает по одному принципу. Последовательность настроек для задания даты и времени системе: часы, минуты, число, месяц, год. Последовательность настроек времени для остальных пунктов: часы, минуты. Для установки необходимых значений используйте кнопки «Up» и «Down», а «Set» для подтверждения установки.

# ВЫВОДЫ

Результаты проделанной работы продемонстрированы в таблице 3.

Таблица 3 - Результаты проведенных работ в соответствии с техническим заданием

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Наименование работ** | **Ожидаемый результат** | **Конечный результат** |
| 1 | Реализация взаимодействия с пользователем через меню посредством клавиатуры из 5 кнопок. | Модуль меню. | Соответствует ожидаемому |
| 2 | Реализация задания параметров для периферийных модулей. | Модуль задания и передачи параметров для подсистем кормления, освещения и климат-контроля. | Соответствует ожидаемому |
| 3 | Реализация обработки сигналов от периферийных модулей | Модуль обработки сообщений подсистем. | Соответствует ожидаемому |
| 4 | Реализация обработки сигналов от пользователя: включение и выключения периферийных модулей и их функций, вне расписания и в случае внештатной работы. | Модуль обработки сообщений пользователя. | Соответствует ожидаемому |

Вывод: в ходе проделанной работы был выполнен весь объем работ в соответствии с поставленным заданием.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы был создано программное обеспечение для главного модуля автоматизированной домашней фермы. Был произведен анализ предметной области, на основе которого были выдвинуты требования к программному обеспечению. Было выполнено проектирование, реализация приложения, выполнено функциональное тестирование программного продукта, а также написано руководство пользователя.

Данное программное обеспечение планируется использовать в будущем, по завершению сборки автоматизированной системы и установки ее на домашнюю ферму. В процессе разработки программного обеспечения для периферийных модулей, возможны изменения для разработанного программного обеспечения. Благодаря использованию различных функций для каждого модуля, любые изменения не потребуют много времени, что упрощает дальнейшую работу и поддержку программного обеспечения.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Латушкин, Ю.М., Водовозов А.М., Летавин И.Д., Система автоматизации жилого дома на платформе Arduino. Наука сегодня: сборник научных трудов по материалам VII международной научно-практической конференции. Вологда: ОАО «Маркер», 2015. С.74.
2. Курочкин А. А., Шабурова Г. В., Гордеев А. С., Завражнов А. И., Оборудование и автоматизация перерабатывающих производств. М.: Колосс, 2007. 276 с.
3. Кондратенков Н. И., Грачев Г. М., Антони В. И., Курсовое проектирование по электроприводу в сельском хозяйстве: Учебное пособие. Челябинск: ЧГАУ, 2002. 236с.
4. Денисенко В. В., Компьютерное управление технологическим процессом, экспериментом, оборудованием. М.: "Горячая линия-Телеком", 2009. 608 с.
5. Элиенс, А. Принципы объектно-ориентированной разработки программ / А. Элиенс. – М.: Издательский дом «Вильямс», 2012. – 496 с.
6. Ларман, К. Применение UML и шаблонов проектирования: Уч. Пос / К. Ларман. - М.: Издательский дом «Вильямс», 2001. - 496 с.
7. What is an Arduino [Электронный ресурс]. URL: https://learn.sparkfun.com/tutorials/what-is-an-arduino (дата обращения 26.11.2017).
8. Arduino - Software [Электронный ресурс]. URL: https://www.arduino.cc/en/main/software (дата обращения 30.11.2017).
9. Дастин, Э. Тестирование программного обеспечения. Внедрение, управление и автоматизация / Э. Дастин, Д. Рэшка, Д. Пол; Пер. с англ. М. Павлов. - М.: Лори, 2013. - 567 c.
10. Плаксин, М.А. Тестирование и отладка программ - для профессионалов будущих и настоящих / М.А. Плаксин. - М.: БИНОМ. ЛЗ, 2013. - 167 c.