

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного
учреждения высшего образования
Пермский национальный исследовательский политехнический университет
Факультет профессионального образования

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

на тему «Разработка проекта четырёхногого шагающего робота с адаптивным

управлением на базе микроконтроллера»

студента группы КСК-18-1спо по специальности

09.02.01 Компьютерные системы и комплексы

Парамонова Кирилла Денисовича _____

Руководитель работы: _____ А.А. Щукина

Консультант по
экономической части: _____ К.В. Кондратьева

Консультант по промышленной экологии
и охране труда: _____ А.К. Гороцин

Рецензент: _____ (_____)

Допуск к защите: _____ М.Н. Апталаев

Лысьва, 2022 г

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного
учреждения высшего образования
«Пермский национальный исследовательский политехнический университет»
ПЦК «Естественнонаучных дисциплин»

Утверждаю:
Председатель ПЦК
_____ М. Н. Апталаев
«___» _____ 2022 г.

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

студенту ПАРАМОНОВУ Кириллу Денисовичу курса 4
группы КСК9-18-1спо
специальность 09.02.01 Компьютерные системы и комплексы

Тема задания «Разработка проекта четырёхногого шагающего робота с адаптивным управлением на базе микроконтроллера»

Структура работы такова:

а) Введение. Аргументировать актуальность выбранной темы, её теоретическое значение и практическую значимость для организации, сформулировать цель и конкретные задачи исследования. Конкретизировать объект и предмет исследования. Увязать решение темы ВКР с общими научно-техническими задачами цифровизации экономики страны.

б) Исследовательский раздел. Понятие «Мехатронная система». Понятие «Шагающий робот». Особенности кинематических схем движения роботов на шагающих опорных конструкциях. Обзор существующих на рынке четырёхногих шагающих роботов. Формирование требований к проектируемой системе.

в) Конструкторский раздел. Обоснование и выбор инструментального обеспечения проекта. Разработка структурной и функциональной схем проекта. Разработка кинематической схемы движения конечностей четырёхногого шагающего робота с учётом требований ТЗ. Разработка управляющей программы робота.

г) Охрана труда и промышленная экология. Анализ вредных и опасных производственных факторов на рабочем месте инженера-электроника. Разработка средств защиты от воздействия выбранного ВиОПФ. Экологические требования к утилизации вычислительной и оргтехники, а также их расходных материалов.

д) Организационно-экономический раздел. Выполнение технико-экономической оценки проекта.

е) Заключение. Краткое изложение решённых задач, актуальность работы, соответствие полученных результатов теме и заданию ВКР.

ж) Список использованных источников.

з) Приложения.

ВВЕДЕНИЕ

1. Исследовательский раздел
 - 1.1. Сущность мехатронных систем
 - 1.2. Обзор кинематических схем роботов на шагающих опорных конструкциях
 - 1.3. Формирование требований к проектируемой системе
2. Конструкторский раздел
 - 2.1. Выбор инструментального обеспечения проектирования

- 2.2. Разработка структурной и функциональной схем устройства
- 2.3. Разработка кинематической схемы устройства
- 2.4. Разработка алгоритмов работы управляющей программы устройства
- 3. Организационно-экономический раздел
 - 3.1. Расчёт себестоимости проекта
 - 3.2. Расчёт экономической эффективности проекта
- 4. Охрана труда и промышленная экология
 - 4.1. Анализ вредных и опасных производственных факторов при пайке деталей, узлов и наладке электронных устройств
 - 4.2. Расчёт технических средств обеспечения безопасности труда на рабочем месте инженера-электроника
 - 4.3. Утилизация компьютерной и оргтехники

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

ПРИЛОЖЕНИЯ

Дата выдачи _____

Срок окончания _____

Руководитель ВКР

_____/А. А. Щукина/

«___» _____ 2022 г.

Задание утверждено на заседании ПЦК «Естественнонаучных дисциплин» протокол №____
от _____ 2022 г.

Председатель ПЦК _____ / М. Н. Апталаев /

«_____» _____ 2022 г.

<ul style="list-style-type: none"> - подпись руководителя по экономической части - подпись руководителя по охране труда - подпись руководителя по промышленной экологии - презентация - доклад 																				
Устранение замечаний по всей ВКР	08.06-15.06																			
Рецензирование Сдача работ на кафедру	20.06																			
Диск с материалами ВКР	21.06																			
Защита ВКР	22.06 – 23.06																			

Руководитель ВКР _____ / _____ /

« ____ » _____ 2022 г.

Студент _____ / _____ /

« ____ » _____ 2022 г.

Согласовано:

Председатель ПЦК «Естественнонаучных дисциплин» _____ / М.Н. Апталаев /
« ____ » _____ 2022 г.

РЕФЕРАТ

Парамонов К.Д. РАЗРАБОТКА ПРОЕКТА ЧЕТЫРЕХНОГО ШАГАЮЩЕГО РОБОТА С АДАПТИВНЫМ УПРАВЛЕНИЕМ НА БАЗЕ МИКРОКОНТРОЛЛЕРА, выпускная квалификационная работа: стр.56, рис. 22, табл. 9, формул 39, библи. 22 назв.

МИКРОКОНТРОЛЛЕР, ШАГАЮЩИЕ РОБОТЫ, РОБОТ-СОБАКА, СЕРВОПРИВОДЫ, РАСПОЗНАВАНИЕ РЕЧИ, СТРУКТУРНАЯ СХЕМА, ПРИНЦИПИАЛЬНАЯ СХЕМА.

Объект исследования – шагающие роботизированные платформы.

Цель работы – разработка проекта четырёхногого шагающего робота с адаптивным управлением на базе микроконтроллера.

В процессе проектирования были получены следующие результаты:

- выполнен анализ проектов шагающих роботов;
- выполнен подбор инструментального обеспечения устройства и элементной базы;
- разработана принципиальная схема устройства, программный код;
- рассмотрены основы охраны труда и промышленной экологии;
- проведён расчёт экономических параметров разработанного устройства.

В результате работы был разработан проект четырёхногий шагающий робот с адаптивным управлением на базе микроконтроллера.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ	5
1.1 Сущность мехатронных систем	5
1.2 Обзор кинематических схем роботов на шагающих опорных конструкциях	14
1.3 Формирование требований к проектируемой системе	18
2 КОНСТРУКТОРСКИЙ РАЗДЕЛ	21
2.1 Выбор инструментального обеспечения проектирования	21
2.2 Выбор элементной базы	25
2.3 Разработка структурной, функциональной и принципиальной схемы 26	
2.4 Разработка кинематической схемы управления	29
2.5 Разработка алгоритмов работы управляющей программы устройства 31	
3 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКИЙ РАЗДЕЛ	33
3.1 Расчет затрат на разработку программы для микроконтроллера	33
3.2 Расчет затрат на внедрение программы для микроконтроллера	36
3.3 Расчет эксплуатационных текущих затрат по программе для микроконтроллера	38
3.4 Расчет экономической целесообразности разработки и внедрения информационных технологий	41
4 ОХРАНА ТРУДА И ПРОМЫШЛЕННАЯ ДЕЯТЕЛЬНОСТЬ	47
4.1 Техника безопасности на рабочем месте инженера-электроника	47
4.2 Опасные производственные факторы на рабочем месте инженера- электроника	51
4.3 Методы утилизации вычислительной техники	53
ЗАКЛЮЧЕНИЕ	55
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	56
Приложение А - Принципиальная схема	59
Приложение Б - Программный код	60

ВВЕДЕНИЕ

В настоящее время микроконтроллеры используются во всех сферах жизнедеятельности современного человечества, в огромном множестве устройств, окружающих его. В микроконтроллере есть много функциональных возможностей и его просто подключить с другими устройствами.

На базе микроконтроллера можно собрать робота, который будет либо ползать, либо передвигаться, либо летать или катиться. Робот может издавать звуки, поднимать тяжести или даже просто ходить. Также может управляться либо голосом, либо пультом управления, либо приложением с телефона, в зависимости от построения робота.

В современном мире большинство роботов стали замещать плюшевые игрушки и иногда даже домашних питомцев, если у человека аллергия. Суть в этих роботах, чтобы ребенок развивался, играя с роботом. Процесс обучения заложен в каждом существующем роботе-игрушке.

Когда ребенок играет с такими роботами, он уже получает первоначальное представление о том, как устроен механизм. Также прививает ребенку аспекты животного мира, его поведение. Также на основе этого можно вывести навыки дрессировщика.

Такой робот подойдет всем, не имея значения, взрослый это или ребёнок, главное чтобы было желание играть с таким роботом.

Объект исследования – шагающие роботизированные платформы.

Предмет исследования – проектирование четырёхногих шагающих роботов.

Целью выпускной квалификационной работы является разработка проекта четырёхногого шагающего робота с адаптивным управлением на базе микроконтроллера.

Для достижения поставленной цели были выделены следующие задачи:

- выполнить анализ существующих решений на рынке шагающих роботов;
- выполнить подбор инструментального обеспечения проекта;
- разработать функциональную, структурную и принципиальную схемы проектируемой системы;
- разработать управляющую программу для контроллера робота;
- провести расчет технико-экономических показателей проекта.

1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ

1.1 Сущность мехатронных систем

Мехатроника - это вид науки и техники, основанный на системной интеграции прецизионных механических компонентов, датчиков окружающей среды и самого объекта, энергии, исполнительных механизмов, усилителей и вычислительного оборудования. Система мехатроники представляет собой комбинацию электромеханических, электрогидравлических, электронных компонентов и компьютерного оборудования с непрерывным и динамически изменяющимся обменом энергией и информацией между ними, подключенную к общему автоматическому управлению с элементами искусственного интеллекта [1].

Из этого следует, что Мехатроника изучает особый подход к созданию машин с новыми функциями. И этот подход универсален и используется в самых разных машинах и системах. Управление высококачественной мехатронной системой происходит только тогда, когда учитывается конкретный объект, подлежащий управлению. Взаимосвязь механических, электрических и компьютерных элементов и устройств является основой для построения мехатронных систем. Каждый компонент имеет определенные независимые функции, и они объединяются, образуя новую систему, которая приобретает новые функции.

Мехатроника еще не имеет четких границ в терминологии и характеристиках, поскольку это новая область науки и техники. Из-за стремительного развития мехатроники растут и требования рынка на современную продукцию машиностроения. Такие требования включают:

- сверхвысокая точность движений для внедрения прецизионных технологий;
- сверхвысокие скорости перемещения в технологических процессах рабочих;

- высокие требования к подвижным заготовкам со сложными контурами и поверхностями;
- возможность перенастройки системы в зависимости от конкретных задач и функций;
- высокая надежность и безопасность эксплуатации;
- интеллект в поведении машин и аппаратов.

Всё это относится к созданию и использованию новых машин и оборудования, а также к обеспечению конкурентоспособности на мировом рынке.

Триединая сущность используется в качестве основы для построения систем мехатроники, в которых механические, электронные и компьютерные элементы соединены друг с другом. В результате наиболее распространенным графическим символом мехатронной системы стали три пересекающихся круга (Рисунок 1 [4]), размещенные на внешней стороне «производство» – «управление» – «требования рынка». Эти три элемента составляют системную интеграцию, которая является необходимым условием для построения мехатронной системы [5].

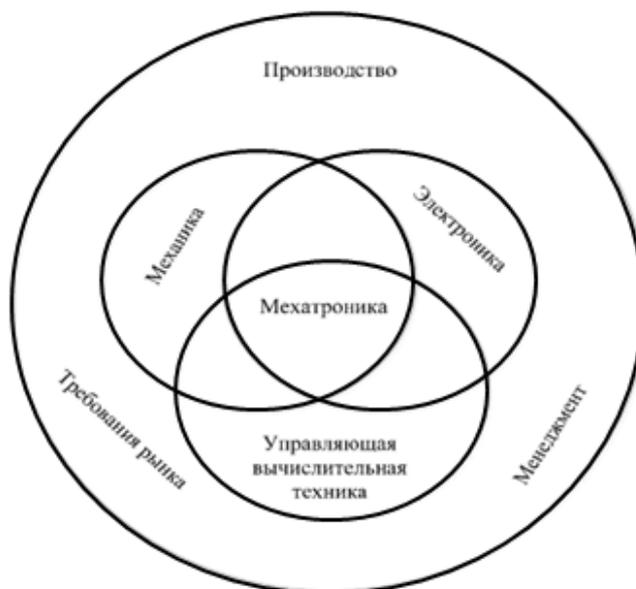


Рисунок 1 – Триединая сущность мехатронных систем

При изучении мехатроники основной целью является модуль мехатроники, который выполняет движение строго в соответствии с единой

управляемой координатой. Эти модули объединены в сложную систему с модульной архитектурой.

Системы мехатроники предназначены для выполнения технического процесса или определенного движения. Критерии качества для выполнения процесса или движения определяются дизайном конкретной применимой задачи и ориентированы на решение проблем. Примеры таких задач включают управление текущим состоянием критических элементов мехатронных систем, управление дополнительными техническими воздействиями на предмет работы с помощью комбинированных методов обработки, регулирование силовых контактов между рабочим органом и предметом работы во время механической обработки.

Для того чтобы обеспечить высококачественное выполнение движений, которые являются сложными и точными то для них используются интеллектуальные методы управления современных систем мехатроники. Такие методы имеют свой набор, который основывается на новых теориях управления, современных компьютерных аппаратных и программных обеспечениях, а также используется перспективный метод синтеза управления систем мехатроники.

Для выполнения заданного движения определяются мехатронные системы:

- управление механического перемещения в реальном времени безразмерной системы или модуля мехатроники обработанный сенсорной информацией;
- многофункциональное управление движением, отдельный способ координатного управления механическим движением;
- связь с оператором через независимые программные интерфейсы;
- обмен с краевой структурой [5].

На рисунке 2 представлена структурная схема модуля мехатроники [4].

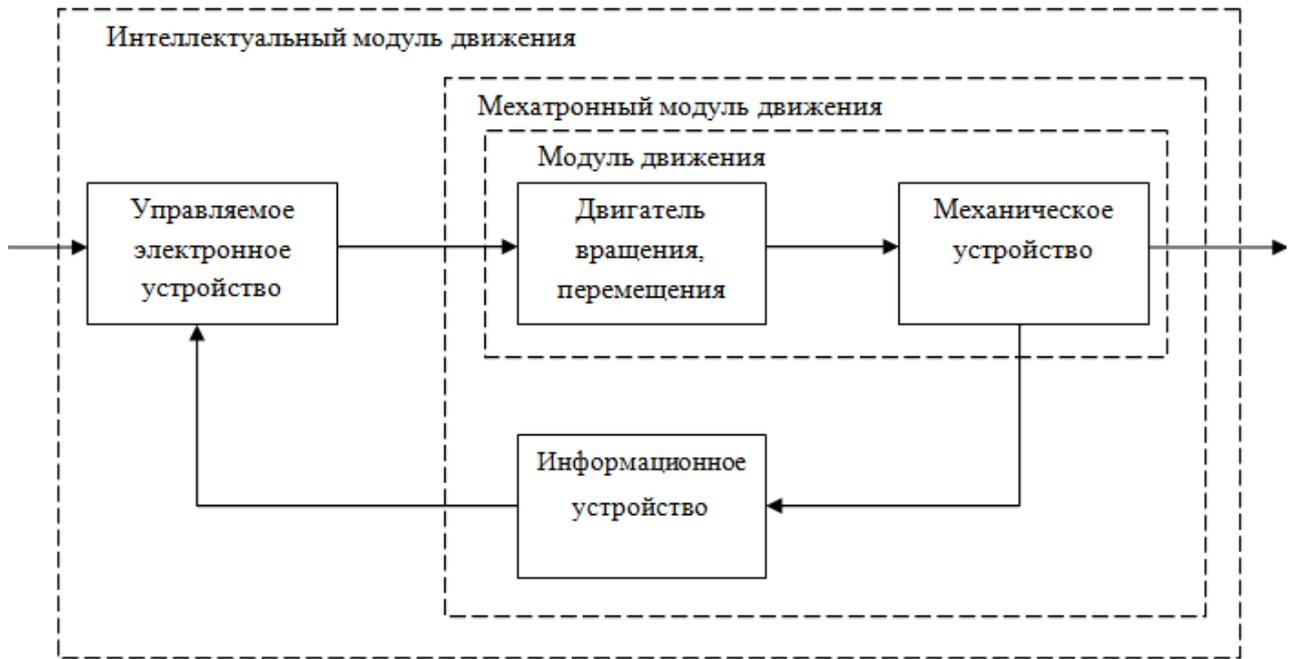


Рисунок 2 - Структурная схема модуля мехатроники

Функция системы мехатроники заключается в преобразовании входных данных с верхней плоскости управления, которая управляется управлением на основе обратной связи.

Метод, который включает в себя одновременный и корреляционный синтез всех элементов системы, является методологической основой для разработки систем мехатроники.

Характер изменения и структура электромеханического модуля должны быть проанализированы с помощью компьютерного управления. Входные данные с верхней плоскости управления направляют сигналы по цепям обратной связи датчиков. В преобразователях сигналы усиливаются и модулируются силовыми элементами, и для того, чтобы воздействовать на механические устройства на преднамеренное перемещение заготовки, сигналы передаются на исполнительные механизмы.

Если вы построите такую систему обычным способом, вам понадобится чрезмерное количество соединительных кабелей с соединительными устройствами и соединительными блоками.

Основываясь на этом выводе, можно сделать вывод о новых способах построения системы, основанной на синергетической интеграции элементов.

Синергетическая интеграция – это не только интеграция отдельных частей в систему через интерфейсные блоки, но и построение за счет эффективной интеграции и взаимодействия элементов в одном модуле, которые могут иметь разный физический характер.

Разница между традиционными и мехатронными методами создания модулей и машин на основе их сущности с помощью компьютерного управления заключается в реализации технических преобразователей и концепции конструкции.

Интерфейсы – это традиционные автономные устройства или узлы, которые изготавливаются на заказ специализированными компаниями или создаются и производятся пользователями.

Используя мехатронный метод, разработчик объединяет все элементы в отдельные блоки, сводя к минимуму промежуточные преобразования и устраняя интерфейсы в виде отдельных специальных блоков.

Отличие системы мехатроники от других систем – это наличие трех важных частей таких, как электроники, электротехники и компьютеров, которые все подключены к определенному потоку информации и электросети, это и будет являться его первой особенностью.

В состав электрической части будут входить: двигатель, датчик, электрические компоненты и рабочий орган. Также в состав электронных компонентов будут входить: силовые преобразователи, измерительные и информационные схемы, в том числе еще и электронные устройства.

Для сбора информации о внешней среде используют датчики, а также о фактическом состоянии рабочего блока и самого устройства, далее эта информация уже перерабатывается и отправляется в компьютерное управляющее устройство. Структура компьютерного управляющего устройства мехатронной системы в основном включает контроллеры управления движением и современные компьютеры [5].

Обобщенная структура автоматизированной системы с компьютерным управлением приведена на рисунке 3 [4].

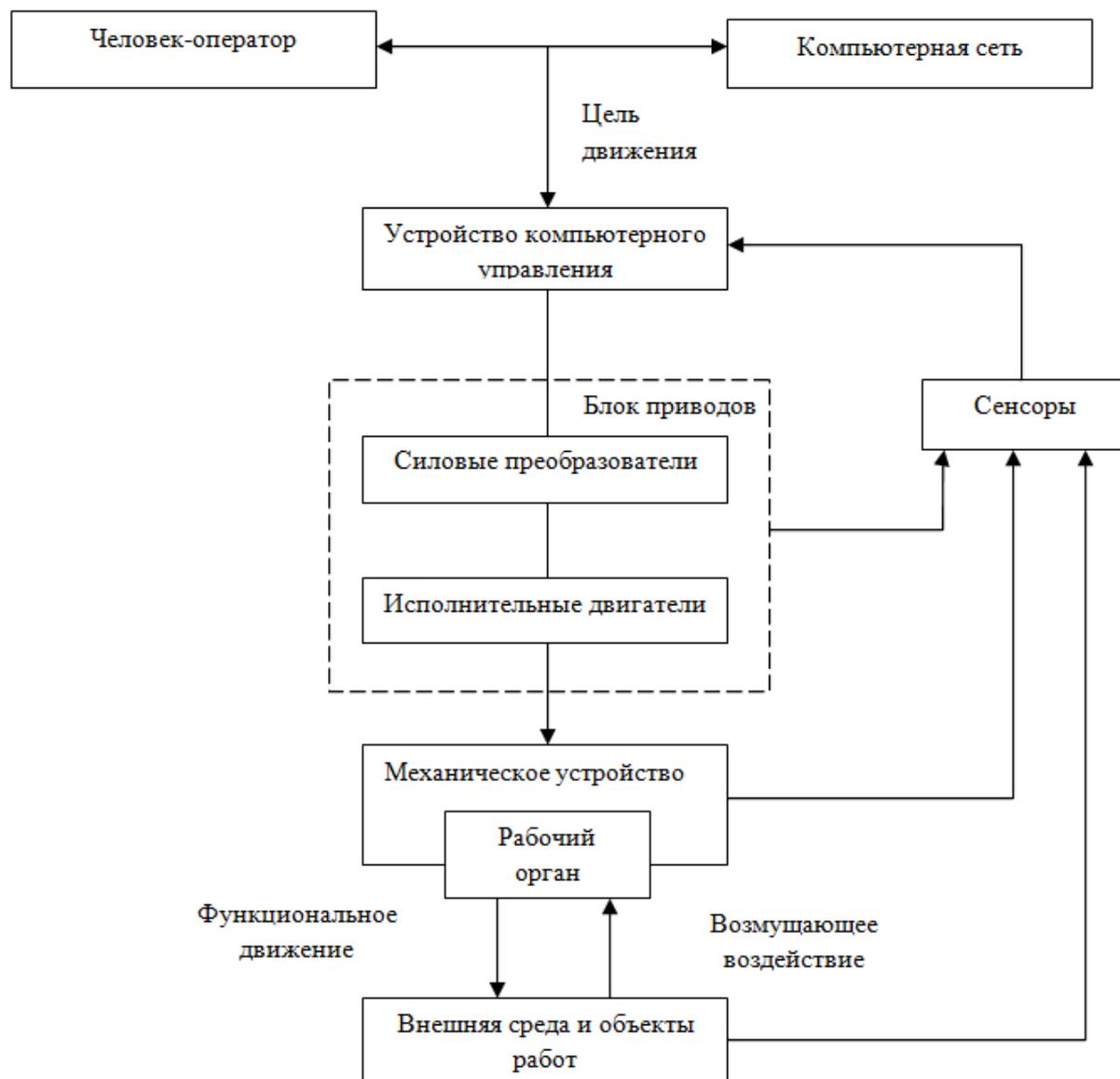


Рисунок 3 – Обобщенная схема машины с компьютерным управлением движением

Компьютерное управляющее устройство выполняет следующие основные задачи:

- через обработку сенсорной информации происходит управление в режиме реального времени;
- управление соответствующими внешними процессами и координация движения систем мехатроники, которая включается в

организацию управления многофункциональным движениям системы мехатроники. Для управления внешним процессом используется независимы ввод/вывод устройства;

- взаимодействие с человеко-компьютерным интерфейсом через человеко-компьютерный интерфейс во время движения, в режиме автономного управления;

- самостоятельный обмен информацией между периферией и датчиками, и другим системным оборудованием.

Важной составляющей системы мехатроники служит для того, чтобы входные данные с верхнего уровня управления преобразовывались в технический процесс, где эта система управляется либо на принципе обратной связи, либо уже заданного механического движения.

Использование электрической энергии стало естественным в качестве питания для современных систем.

Слияние различных элементов в единый технический модуль является сутью метода проектирования мехатроники. Один из элементов исключается для того, чтобы он являлся специальным устройством, который должен сохранять физические свойства преобразования, исполняемый этим же самым модулем [4].

Шагающие роботы – это роботы, которые перемещаются с помощью шарнирных или вращающихся опорных конструкций таким образом, что их перемещение перестраивается одновременно с электрическими, гидравлическими, механическими и другими приводами этих конструкций и их комбинаций.

Существуют некоторые критерии, связанные с концепцией шагающего робота, количество ног или лап, возможные типы ходьбы, положение и устойчивость ног.

Движение шагающего робота, имитирующего ходьбу животного, обычно называют биологической моделью. Наблюдая за животными, первые попытки создать роботов начались с имитации их походки [6].

На рисунке 4 представлено обычное расположение лап рептилий, млекопитающих и насекомых [6].

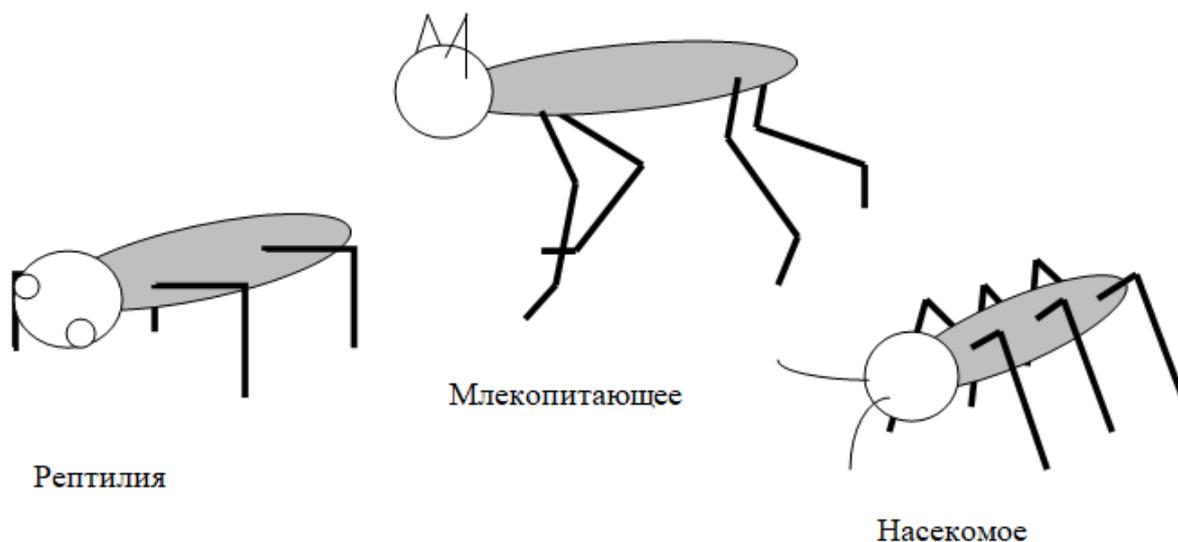


Рисунок 4 – Обычное расположение ног

Важными при создании шагающего робота являются ступни или лапы, которые связаны с биологической точки зрения. Тип ног робота может быть одинаковым или разным для пар ног, и это объединяет их не только с выбором, но и с различными движениями. В принципе, проще и дешевле делать роботов с такими же ногами.

В зависимости от количества ног можно определить тип устойчивости робота. В настоящее время существует три типа устойчивости: динамическая, статическая и квазистатическая устойчивая походка. Такие роботы выполняются на четырех ногах.

При динамической стабильной ходьбе робот может останавливаться только на определённое время, но остановка также может привести к падению робота на бок. Такие роботы являются шестиногими.

При статически стабильной ходьбе робот может остановиться в любой момент, сохраняя равновесие.

При квазистатической стабильной походке ноги представляют собой маятник. В общем, такие роботы - двуногие роботы с тяжелыми и большими ногами.

Что касается количества роботизированных ног, то минимальное их количество может быть равно нулю, но такие роботы ползают, как змеи. Но поскольку мы говорим о шагающих роботах, давайте проанализируем роботов по количеству ног.

У одноногого робота есть один вид шага, и это прыжки. Тип движения такого робота - динамический. Только благодаря мощным приводам, датчикам, сверхбыстрой системе управления и сложному алгоритму робот не падает во время движения.

Двуногие роботы имеют вид динамически устойчивой походки. И такие роботы обычно ходят на согнутых ногах, а биологическая структура таких роботов - это человек. Такие роботы также могут двигаться в квазистатическом стабильном движении. В этом случае робот имеет большие ноги, которые позволяют сохранять равновесие на большой площади, под которой находится центр тяжести робота.

Четвероногие роботы, такие роботы с биологической точки зрения основаны на млекопитающих или рептилиях. Такие роботы двигаются динамически устойчиво или статически устойчиво, перемещая ногу за ногой.

Роботы, у которых уже есть гораздо больше ног, движутся статически с постоянной скоростью. А максимальное количество возможных шагов определяется формулой 1 [6]:

$$N = (2 * k - 1)! \quad (1)$$

где N - количество возможных видов походки, k - количество ног.

Для робота важно положение ног, чтобы робот не падал, необходимо, чтобы его центр тяжести оставался внутри опорного многоугольника. Вершины в которых ступни касаются земли, называется опорным многоугольником. Для четвероногого робота с симметрично

расположенными ногами опорой будет прямоугольник или же квадрат; в случае поднятой ноги, опорный многоугольник становится из квадрата или прямоугольника квадратом.

При проектировании робота, который движется статически в устойчивом движении, центр тяжести должен располагаться так, чтобы он не выходил за пределы многоугольника. В роботах запас прочности используется для устранения любых дополнительных динамических эффектов и повышения стабильности. В этом случае ноги и туловище сконструированы таким образом, чтобы центр тяжести ни в коем случае не приближался к сторонам опорного многоугольника [6].

1.2 Обзор кинематических схем роботов на шагающих опорных конструкциях

Обратная кинематика - это метод получения движения, необходимого системе связанных объектов для достижения известного положения [11].

Для подъема робота вверх или вниз используется правило косинусов. Это можно представить в виде треугольника рисунок 5 [12].

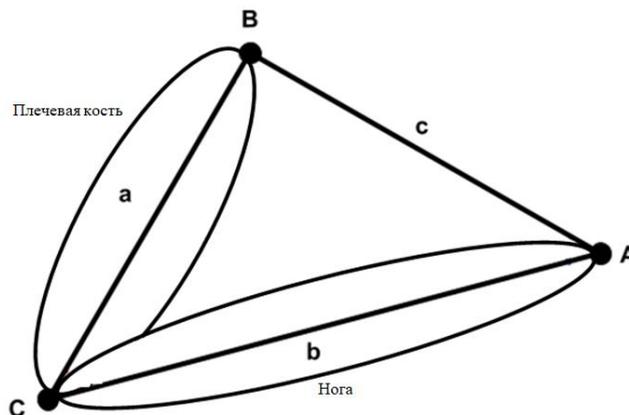


Рисунок 5 – Треугольник образованный от плечевой кости и ноги

Из заданных плечевой кости, ног и расстояния от тела к земле можно вычислить каждый из углов по формулам 2,3,4 [12].

$$\cos A = \frac{b^2 + c^2 - a^2}{2bc} \quad (2)$$

$$\cos B = \frac{a^2 + c^2 - b^2}{2ac} \quad (3)$$

$$\cos C = \frac{a + b^2 - c^2}{2ab} \quad (4)$$

Для передвижения робота вперед используется формула 5 [14]:

$$z^2 + (x - r)^2 = r^2 \quad (5)$$

Для разворота робота в бок используется формула 5 [14]:

$$z^2 + (x/2 - r)^2 = r^2 \quad (6)$$

Адаптивное управление - это набор методов в теории управления для синтеза систем управления, которые обладают способностью изменять параметры контроллера или структуру контроллера в зависимости от изменений параметров управления или внешних помех, влияющих на управление.

Адаптивные роботы более сложны, чем программируемые роботы. Адаптивные роботы адаптируются в особой степени и, оценив конкретную ситуацию, выполняют обязательное действие, в этом отличие от программируемых роботов [15].

BigDog

BigDog - это четвероногий шагающий робот с адаптивным управлением, созданный Boston Dynamics в 2005 году в сотрудничестве с Foster-Miller, Лабораторией реактивного движения и полевой станцией Конкорд Гарвардского университета.

Пример разработки представлен на рисунке 6 [8].

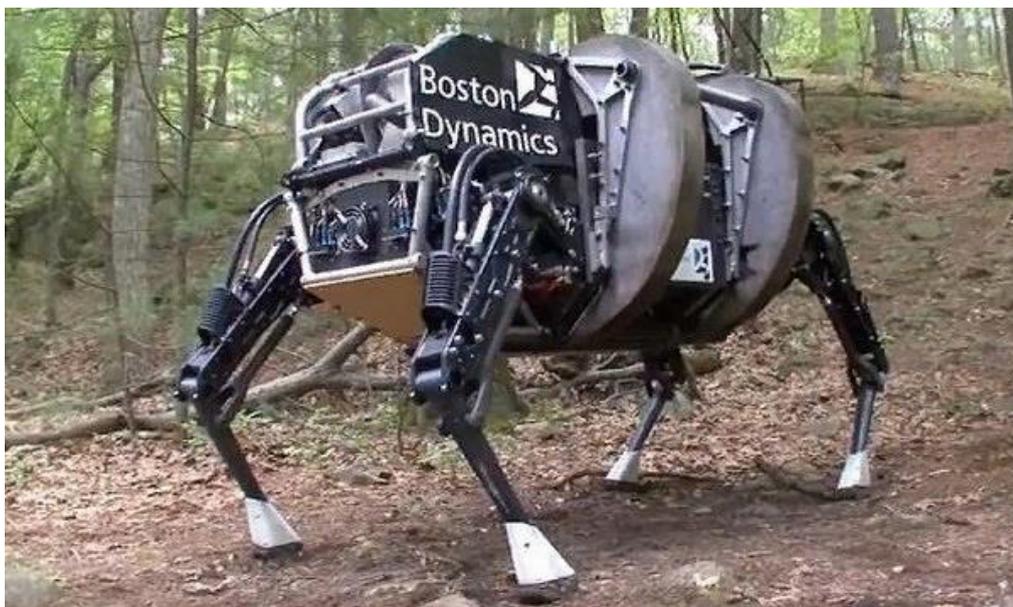


Рисунок 6 - BigDog

Проект BigDog предназначался для перевозки оборудования в местах, где обычные транспорты не способны проехать. Для этого вместо гусениц и колес BigDog оснащен четырьмя ногами. Также на ногах расположено достаточно большое количество датчиков, а также он оснащен системой бинокулярного зрения и лазерным гироскопом.

Длина данного робота составляет 0,91 метра, а в высоту 0,76 метра и имеет достаточно тяжелый вес в 110 килограмм. На данный момент такой робот может подниматься на 35 градусов в высоту, передвигаться по местности со скоростью 6,4 км/ч, а также перевозить груз в 154 кг. Контроль за движением у такого робота происходит за счет датчиков, а также система управляет навигацией и балансом.

Цена - 5,5 млн. рублей [8].

AIBO

Aibo - это серия собак-роботов, разработанная Sony.

Aibo может ходить, «видеть» окружающие его объекты с помощью видеокамеры и инфракрасных датчиков, распознавать команды и лица. Робот полностью независим: он может учиться и развиваться в зависимости от своего владельца, окружающей среды или других мотивов Aibo. Тем не

менее, он подчиняется правилам с помощью специальных программ. Данный робот имеет функцию «взрослая собака», где он использует все свои функции, а также имеется функция «щенок», где робот развивается имитируя щенка и постепенно раскрывает свои способности.

Пример разработки Aibo представлен на рисунке 7 [8].

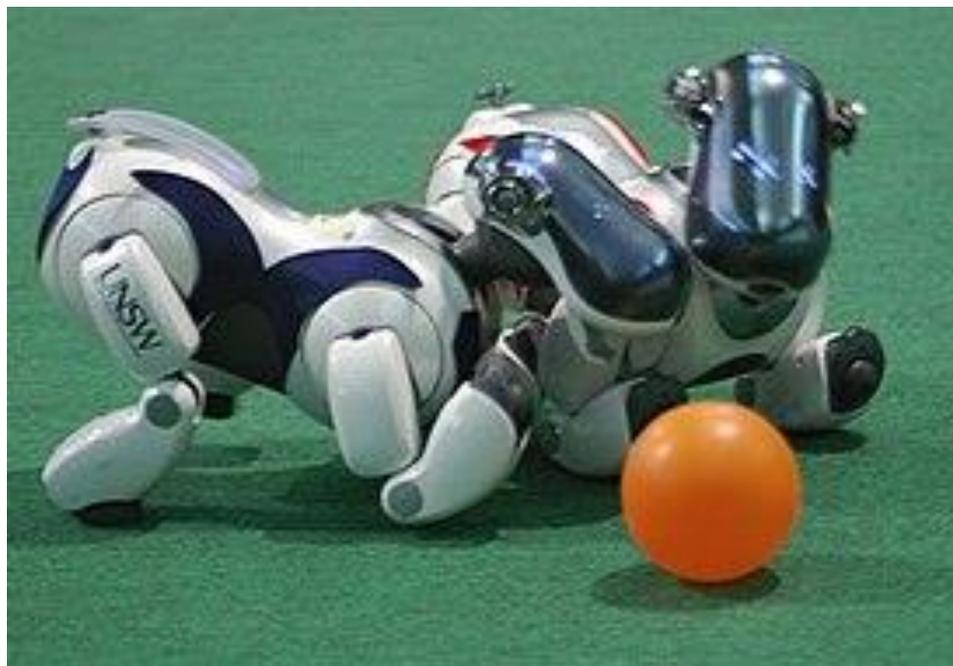


Рисунок 7 - Aibo

«Настроение» Aibo может меняться в зависимости от окружающей его среды и влиять на поведение. Инстинкты позволяют Aibo передвигаться, развлекаться со своими игрушками, играть и общаться с со своим хозяином, удовлетворять свое любопытство, заряжаться энергией и просыпаться после сна. Разработчики утверждают, что в Aibo есть симуляция шести эмоций: счастья, печали, страха, антипатии, удивления и гнева.

Цена – 419990 рублей [8].

CyberDog

Cyberdog работает на базе модуля Nvidia Jetson Xavier NX, который содержит 384 ядра CUDA, 48 тензорных ядер, шесть процессорных ядер Carmel и два ядра глубокого обучения. Робот оснащен 11 датчиками для ориентации в пространстве. Он может следовать за хозяином и выполнять различные трюки, такие как прыжки и кувырки. Cyberdog оснащен

несколькими камерами и микрофонами, которые позволяют роботу перемещаться в пространстве и анализировать ситуацию вокруг него.

Пример разработки CyberDog представлен на рисунке 8 [8].



Рисунок 8 - CyberDog

Устройство может быть активировано с помощью речи или специального приложения на вашем смартфоне. Робот идентифицирует владельца и выполняет только его или её приказы, следуя за ним или за ней.

Xiaomi позволяет подключать к нему различные устройства. Производитель пояснил, что данный робот имеет три порта USB-C и один HDMI.

CyberDog оснащен оригинальными сервоприводами, произведенными Xiaomi. Робот имеет грузоподъемность 3 кг, а скорость движения может достигать 3,2 м/с.

Цена – 115 тыс. рублей [10].

1.3 Формирование требований к проектируемой системе

Данное устройство в основном предназначено для детского развлечения.

В данном устройстве следует выделить следующие функциональные подсистемы:

- подсистема ввода – предназначена для получения внешних сигналов;
- подсистема обработки – предназначена для приведения указанных сигналов к подсистеме вывода;
- подсистема вывода – предназначена для вывода сигналов от подсистемы обработки.

Система должна поддерживать следующий режим функционирования: основной режим, в котором все подсистемы устройства выполняют все свои основные функции.

Устройство должно выполнять следующие функции:

- передвигаться по ровной местности;
- обнаружение и обход препятствия с помощью датчика, на уровне глаз;
- с помощью модуля принимать голосовые команды человека;
- махать хвостом, когда глядят по шее во время положения сидя;
- предупреждать, когда садится аккумулятор;
- предупреждать, когда аккумулятор заряжен.

Требования безопасности:

- разбор и ремонт устройство при выключенном питании;
- держать в защищённом от воды месте при подключенном питании.

Требования к хранению устройства:

- поддерживать электропитание устройства, раз в месяц;
- хранить при температуре от -5 C° до $+35\text{ C}^{\circ}$;
- хранить при относительной влажности не превышающей 70%;
- обеспечение надёжного хранения с целью подавления механических повреждений устройства.

Требования к эксплуатации устройства:

- не допускать устройству ходить на высокой поверхности, с которой оно может упасть;

- использовать устройство при температуре от -5 C° до $+35\text{ C}^{\circ}$;
- использовать устройство при относительной влажности не превышающей 70%;
- использовать устройство на прямой поверхности.

2 КОНСТРУКТОРСКИЙ РАЗДЕЛ

2.1 Выбор инструментального обеспечения проектирования

Для проектирования устройства «Четырёхногий шагающий робот с адаптивным управлением на базе микроконтроллера» рассмотрим несколько программ и выберем подходящую.

EasyEDA - это кросс платформенная веб-платформа для автоматизации электронного проектирования, включающая редактор собственных схем, редактор топологии своих печатных плат, симулятор spice, облачное хранилище данных своих работ, систему управления проектами и для заказа производственной печати.

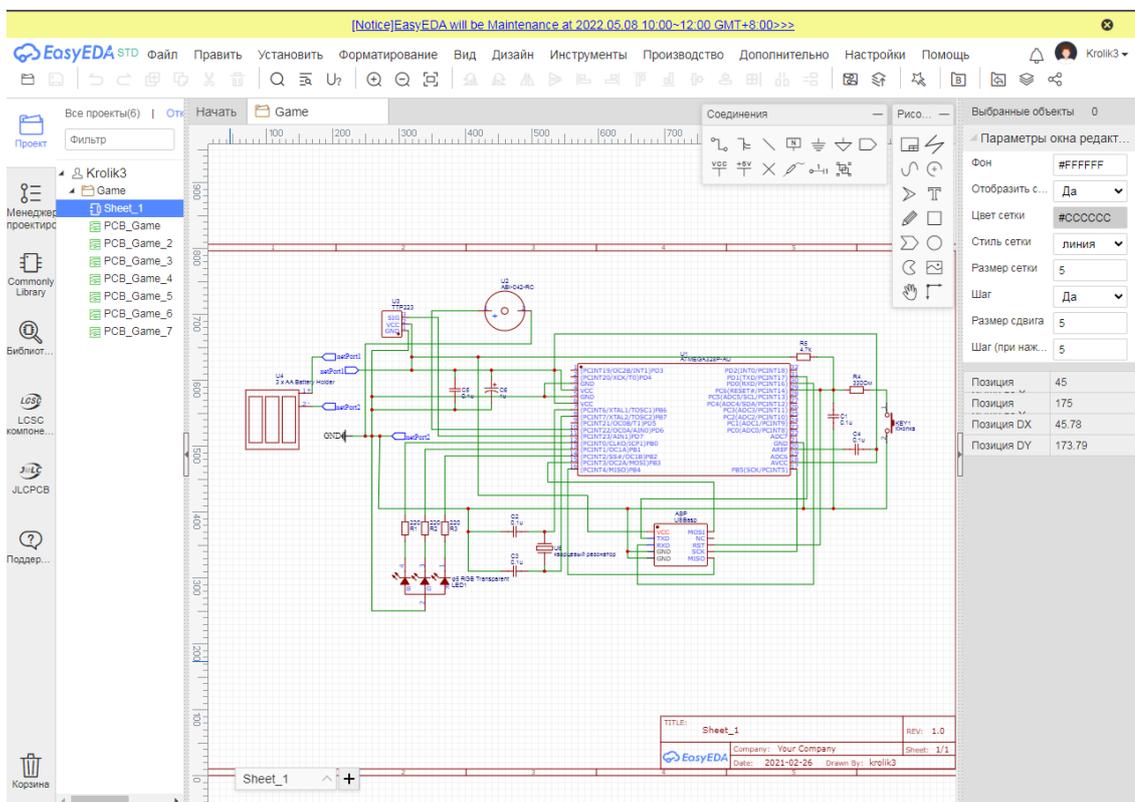


Рисунок 9 – Среда проектирования EasyEDA

Fritzing – бесплатное программное обеспечение для разработки систем автоматизированного проектирования, для проектирования электронного оборудования.

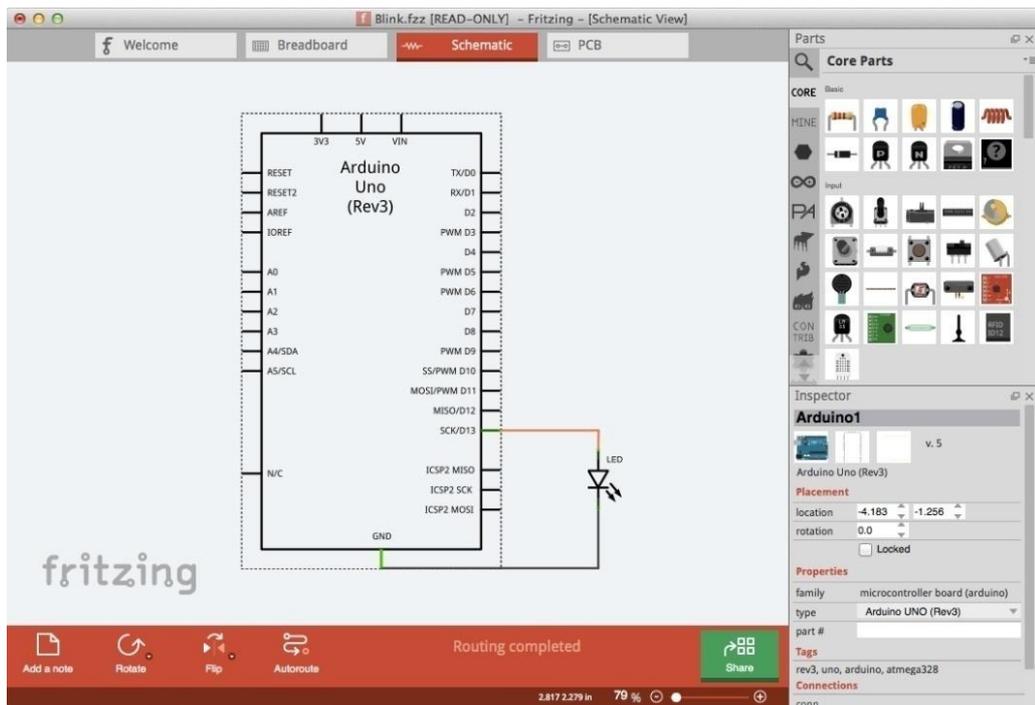


Рисунок 10 – Среда проектирования Fritzing

Proteus Design Suite — пакет программ для автоматизированного проектирования электронных схем.

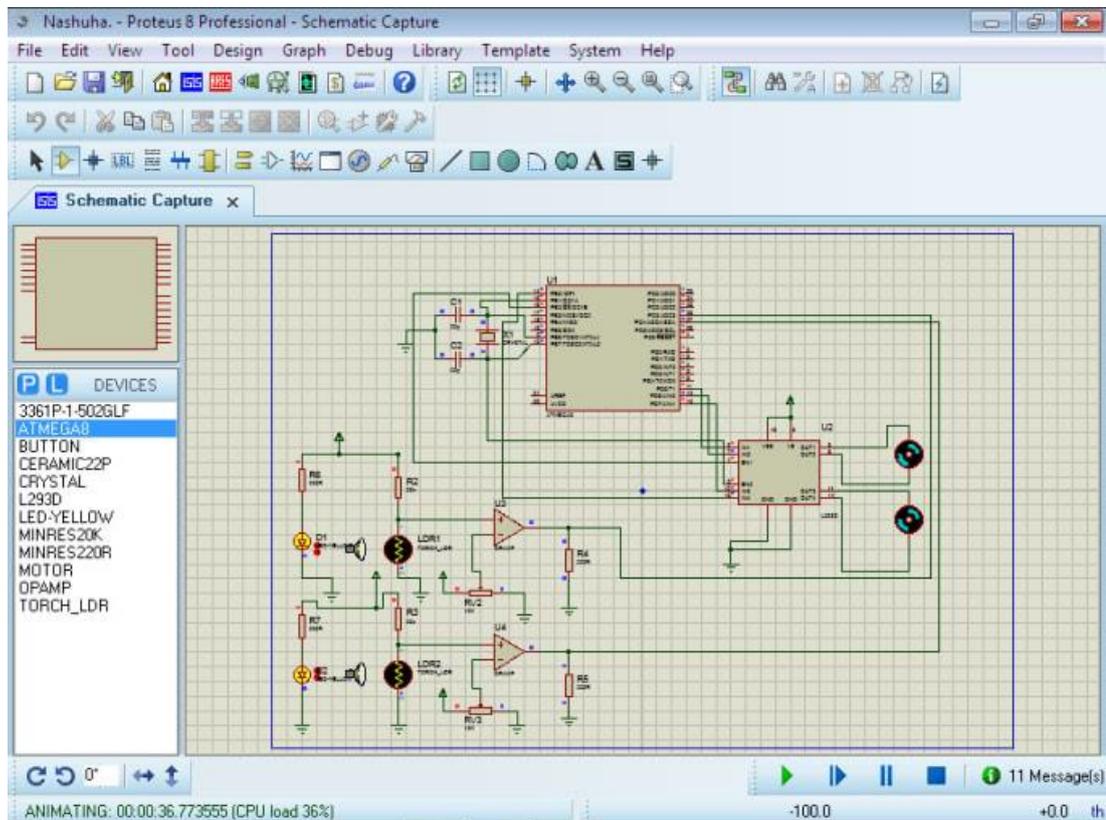


Рисунок 11 – Среда проектирования Proteus

Выберем программу EasyEDA, так как в ее библиотеке можно подобрать нужные элементы или сделать их самому, если не нашли нужные элементы.

Для программирования микроконтроллера "Четырехногий шагающий робот с адаптивным управлением на базе микроконтроллера" рассмотрим несколько программ и выберем одну из них.

Atmel Studio - это бесплатная проприетарная интегрированная среда разработки разных приложений, подходящая для 8-разрядных и 32-разрядных микроконтроллеров серии AVR от компании Atmel и 32-разрядных микроконтроллеров серии ARM.

Atmel Studio содержит внутренний компилятор GNU C/C++ и эмулятор, позволяющий отладить выполнение программы без загрузки в микроконтроллер.

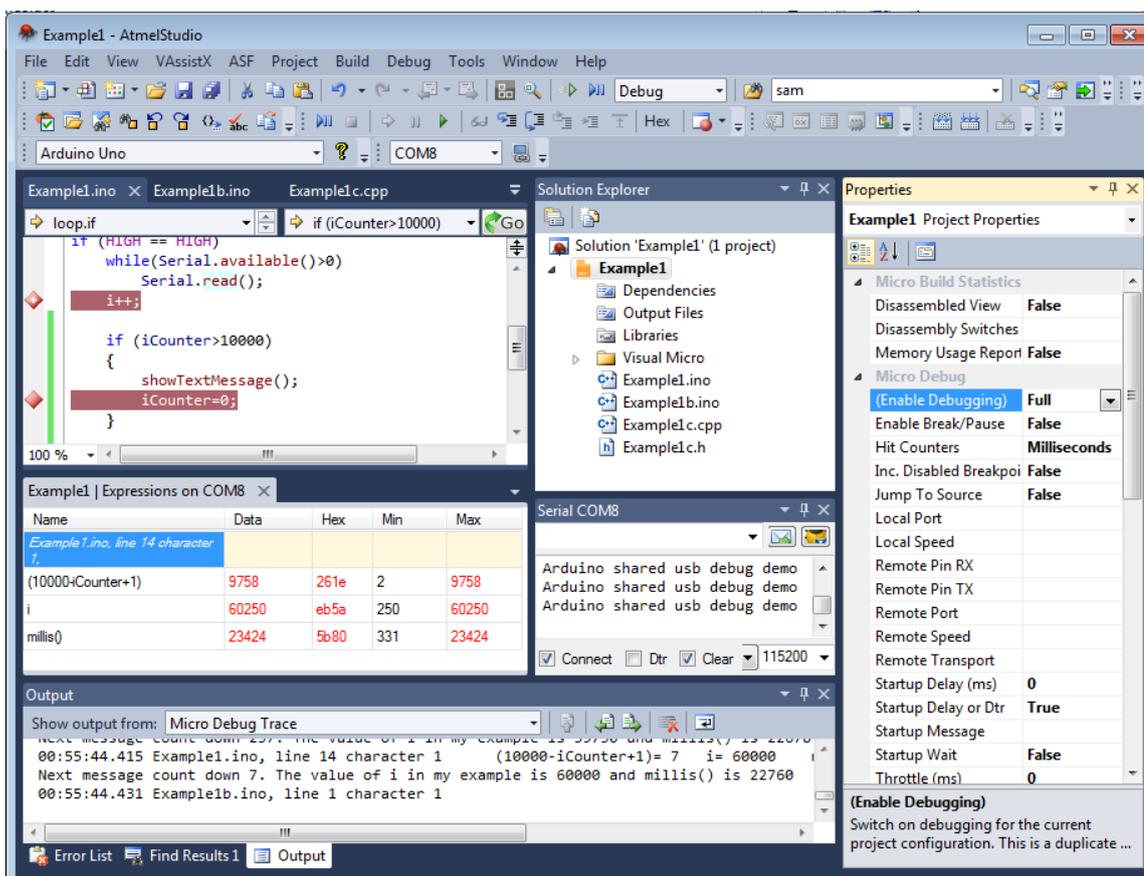


Рисунок 12 – Среда разработки Atmel Studio

Programmino – это малобюджетная IDE Arduino, Genuino или аналогичные платы с автозаполнением кода, подсветкой пользовательского

синтаксиса, аналоговым плоттером, просмотром оборудования, извлечением HEX-файла и редактор HTML5.

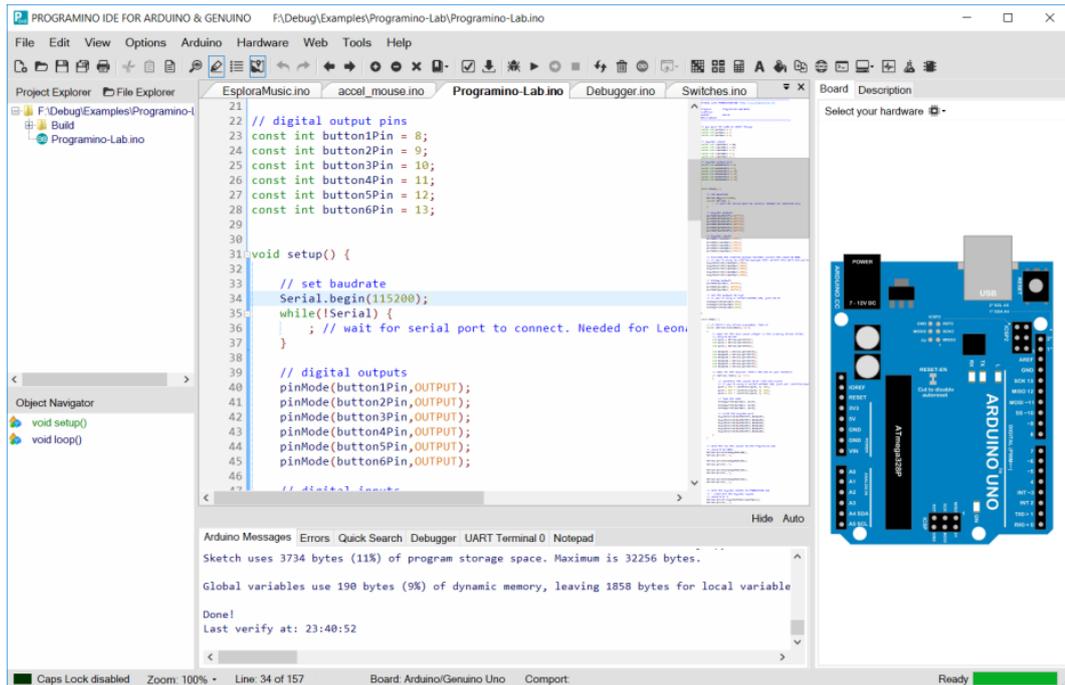


Рисунок 13 – Среда разработки Programino

Arduino IDE — интегрированная среда разработки, предназначенная для создания и загрузки программ на Arduino-совместимые платы, а также на платы других производителей.

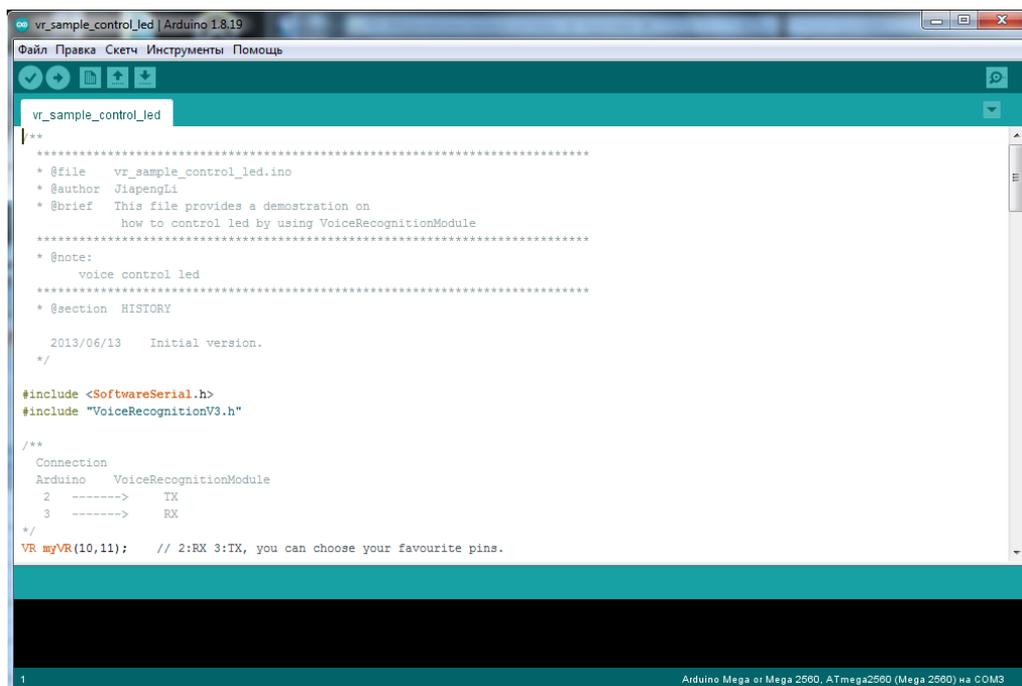


Рисунок 14 – Среда разработки Arduino IDE

Из этих программ выберем Arduino IDE, так как в ее среде проще работать, даже если человек начинающий программист или не умеет программировать микроконтроллер.

2.2 Выбор элементной базы

Перед тем, как проектировать устройство нужно выбрать элементы, подходящие для нашего устройства. Поэтому произведем выбор элементной базы.

Во-первых, выберем микроконтроллер для нашего устройства, которые представлены в таблице 1

Таблица 1 – Сравнение микроконтроллеров

	Atmega328p	Atmega2560	Atmega32u4
Цена, р	150	600	300
Тактовая частота, МГц	16	16	16
Флеш-память, Кб	32	256	32
Линии ввода-вывода	20	54	20
UART	1	4	1

Выберем согласно таблице 1 микроконтроллер Atmega2560 так, как он подходит по количеству линий ввода- вывода, а также по количеству флеш-памяти.

Далее выберем сенсорный модуль ТТР223 для включения системы и ввода сигналов, которые представлены в таблице 2.

Таблица 2 – Сравнение сенсорных датчиков

Наименование	Напряжение питания	Количество сенсорных кнопок
ТТР223	2,4-5,5В	1
ТТР224	2,4-5,5В	4
ТТР226	2,4-5,5В	8
ТТР229	2,4-5,5В	16

Для реализации функциональной системы требуется всего одна кнопка. Согласно сравнительной таблице 2, оптимальным количеством кнопок обладает сенсорный модуль ТТР223.

Далее необходимо выбрать ультразвуковой датчик для получения сигналов о препятствии, которые представлены в таблице 3.

Таблица 3 - Сравнение ультразвуковых датчиков

	HC-SR04	HY-SRF05
Рабочее напряжение	5 В постоянного тока	5 В постоянного тока
Ток покоя	< 2мА	<2 мА
Выходной сигнал:	высокий уровень 5В, низкий уровень 0В	высокий уровень 5В, низкий уровень 0В
Угол распространения (эффективный угол)	< 15 градусов	< 15 градусов
Диапазон измеряемого расстояния	2см-450см	2см-450см
Разрешающая способность (точность)	~3 мм	~2 мм
Входной сигнал запуска:	10мс TTL импульс	10мс TTL импульс
Эхосигнал	выходной сигнал TTL PWL	выходной сигнал TTL PWL
Pins	<ol style="list-style-type: none"> 1. VCC > плюс питания (+ 5 В) 2. trig(T) > вход сигнала триггерного управления 3. echo(R) > выход эхосигнала 4. GND > минус питания 	<ol style="list-style-type: none"> 1. VCC > плюс питания (+ 5 В) 2. trig(T) > вход сигнала триггерного управления 3. echo(R) > выход эхосигнала 4. OUT > цифровой выход есть/нет 5. GND > минус питания
Цена	220 р.	239 р.

Для этого устройства необязательна высокая точность, поэтому выберем ультразвуковой датчик HC-SR04, при этом он выйдет немного меньше по цене.

2.3 Разработка структурной, функциональной и принципиальной схемы

Прежде чем собирать устройство, нужно построить для него структурную и функциональную схемы, представленные на рисунках 15 и 16.



Рисунок 15 - Структурная схема устройства

В блок ввода входят: сенсорная кнопка, ультразвуковой датчик и модуль распознавания голоса.

В блок вывода входят: сервоприводы и динамик.

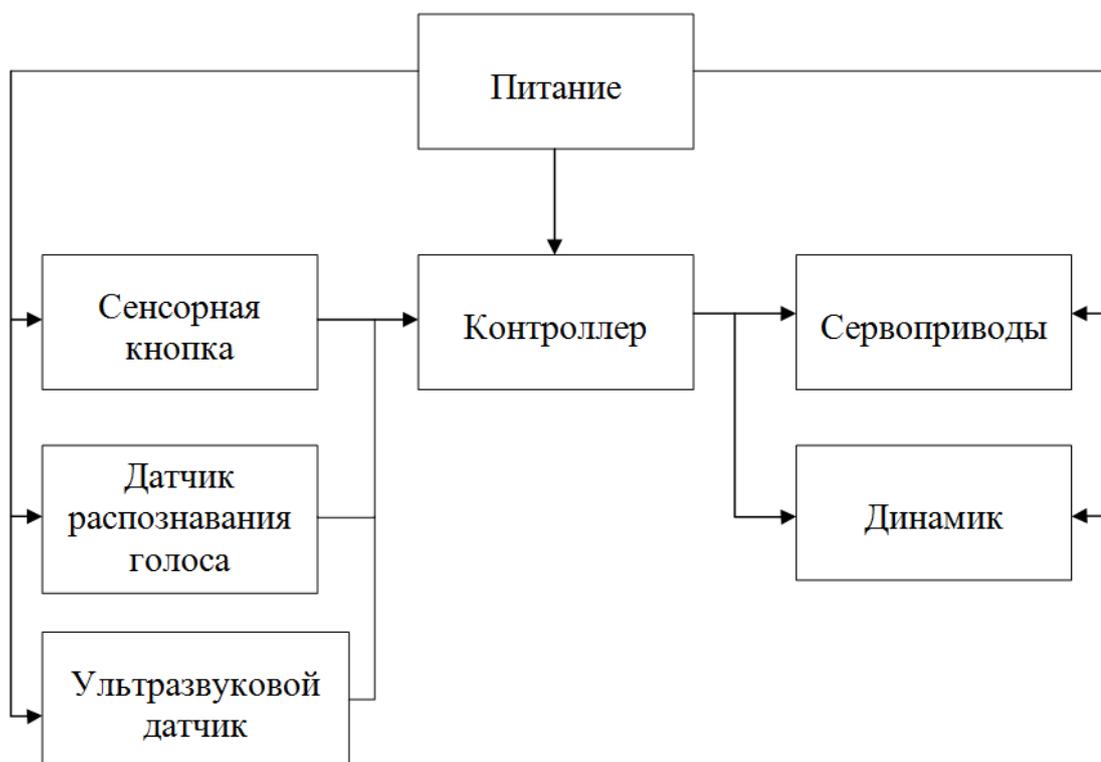


Рисунок 16 - Функциональная схема устройства

Питание подается на сенсорную кнопку, модуль распознавания голоса, ультразвуковой датчик, контролер и сервоприводы. С сенсорной кнопки, датчика распознавания голоса и ультразвукового датчика подаются сигналы

на контроллер. С контроллера передаются выходные сигналы на сервоприводы и динамик.

Далее для реализации устройства понадобится принципиальная схема, поэтому разработаем принципиальную схему устройства. Все элементы соединим с необходимыми для них местами. Данная принципиальная схема (Рисунок 17) в малом масштабе расположена в Приложении А.

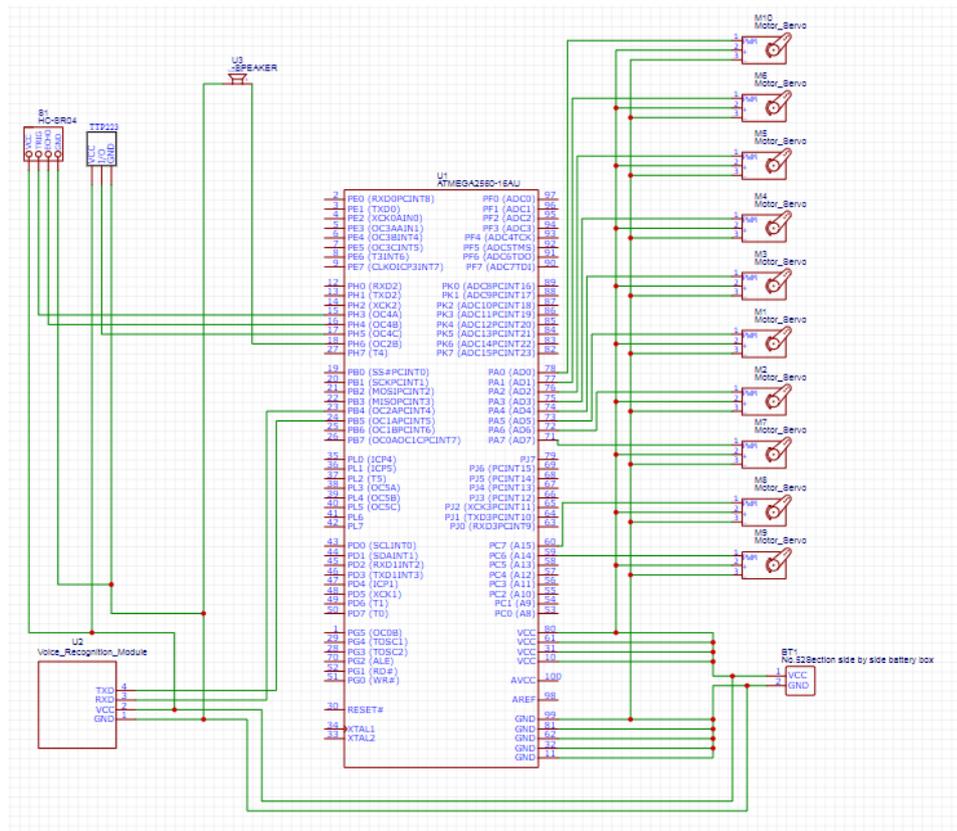


Рисунок 17 - Принципиальная схема

Держатель для батарей подключен к микроконтроллеру к выводам VCC(питание) и GND(земля). К GND также подключены сенсорная кнопка, динамик сервоприводы, модуль распознавания голоса, ультразвуковой датчик. Сервоприводы подключены к микроконтроллеру к выводам PA0-PA7, PC7 и PC6, и к VCC. Динамик подключен к микроконтроллеру к выводу PH6. Сенсорный модуль подключен к питанию VCC и микроконтроллеру к выводу PH5. Ультразвуковой датчик подключен к VCC и микроконтроллеру к выводам PH4 и PH3. Модуль распознавания голоса подключен к VCC и микроконтроллеру к выводам PB5 и PB4.

2.4 Разработка кинематической схемы управления

Разберёмся с передвижением робота с помощью кинематической схемы представленной на рисунке 18.

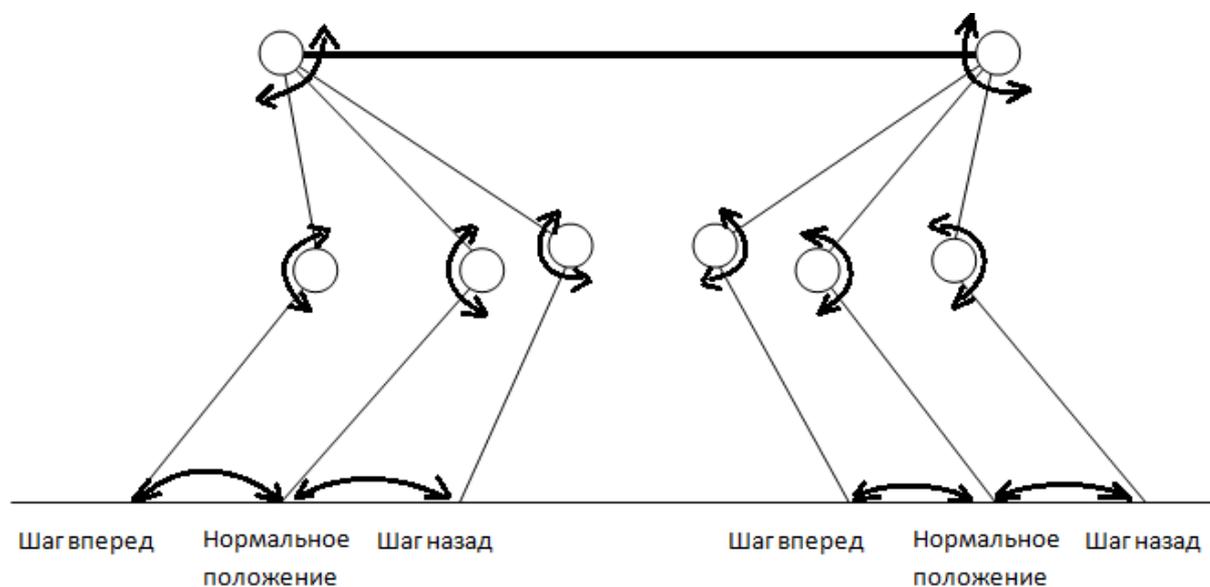


Рисунок 18 – Кинематическая схема

Круги на рисунке представляют движущие моменты, то есть сервоприводы. Стрелки у кругов указывают в какие стороны движется сервоприводы. Стрелки у ног указывают на ширину шага.

Для передвижения робота вперед используется формула:

$$z^2 + (x - 30)^2 = 30^2 \quad (7)$$

Диапазон передвижения по оси x 0-60.

Также рассмотрим, как робот должен выглядеть сидя, стоя и лёжа. Все эти состояния представлены на рисунках 19-21.

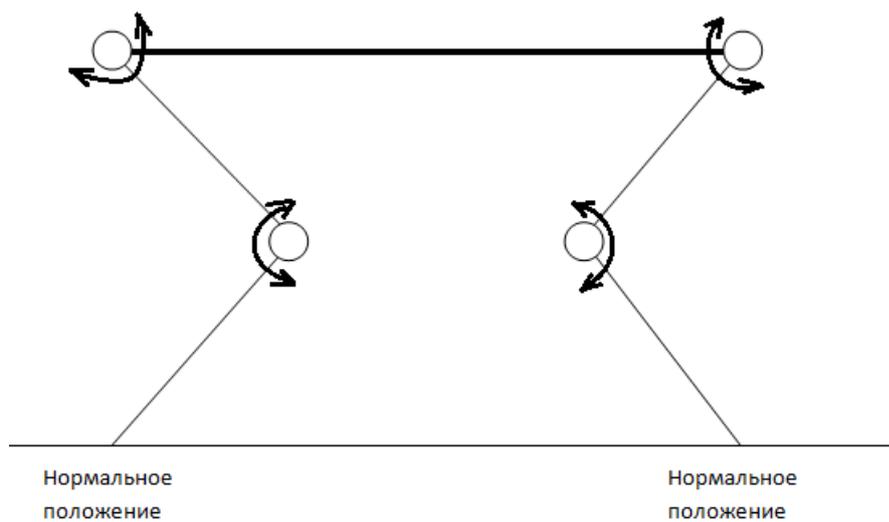


Рисунок 19 – Положение стоя

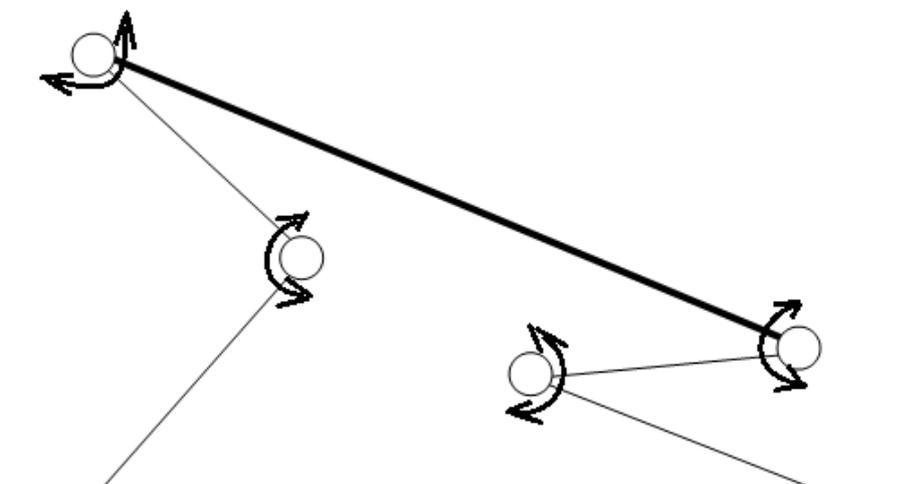


Рисунок 20 – Положение сидя

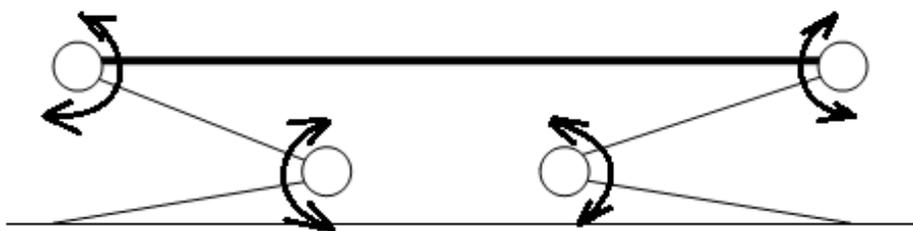


Рисунок 21 – Положение лежа

Для положений ног в положения сидя, стоя и лёжа используются формулы:

$$\cos A = \frac{b^2+c^2-a^2}{2bc} \quad (8)$$

$$\cos B = \frac{a^2+c^2-b^2}{2ac} \quad (9)$$

$$\cos C = \frac{a+b^2-c^2}{2ab} \quad (10)$$

Длины ног a и b равны 8 см, а длина c (от туловища до земли) равна 11 см. То по расчетам получается, что в положении «Стоя» углы будут равны:

$$\cos A = \frac{8^2+11^2-8^2}{2*8*11} = 0,6875 = 46^\circ$$

$$\cos B = \frac{8^2+11^2-8^2}{2*8*11} = 0,6875 = 46^\circ$$

$$\cos C = \frac{8^2+8^2-11^2}{2*8*8} = 0,0547 = 87^\circ$$

2.5 Разработка алгоритмов работы управляющей программы устройства

Распишем алгоритм работы робота.

Включение робота производится голосовой командой «Включить», после этого сервоприводам, сенсорной кнопке и ультразвуковому датчику поступает питание, а робот переходит в состояние стоя.

После включения можно использовать ещё 6 голосовых команд.

Команда «Выключить» переводит робота в режим лёжа и отключает питание к сервоприводам, сенсорной кнопки и ультразвуковому датчику.

Команда «Сидеть» переводит робота в режим сидя. Если в режиме сидя провести по шее, то хвост начнёт движение.

Команда «Лежать» переводит робота в положение лёжа.

Команда «Стоять» переводит робота в положение стоя.

Команда «Голос» производит из динамика голос собаки.

Команда «Идти» робот начинает движение, если робот был в положение сидя или лёжа, то вначале он перейдет в положение стоя. Если робот

встретит на своем пути препятствие с помощью ультразвукового датчика (на уровне глаз), то он осмотрится направо и налево, и пойдет в ту сторону в которой расстояние больше.

Далее для написания программы изначально построим блок-схему, которая показывает работу устройства. Блок-схема представлена на рисунке 22. Сам код программы находится в приложении Б.

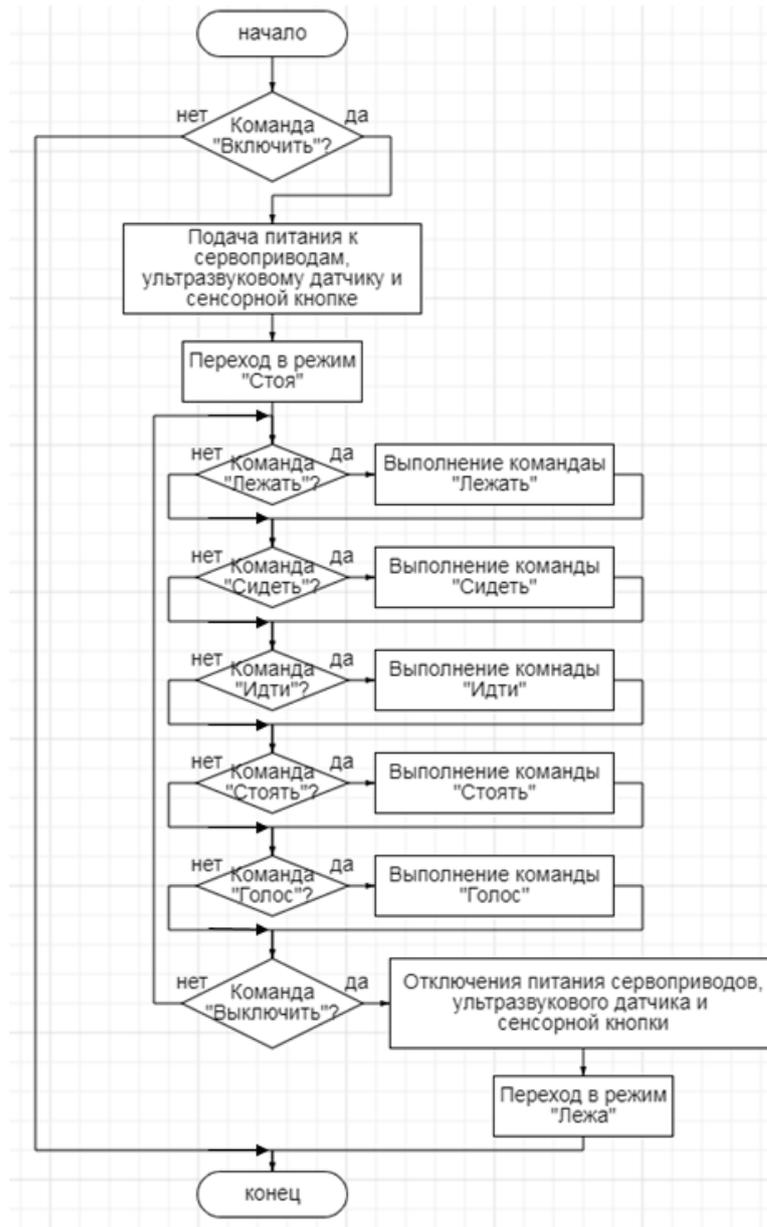


Рисунок 22 - Блок-схема программы

Как и говорилось выше изначально нужно выполнить команду «Включить», чтобы в дальнейшем работать с устройством.

3 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКИЙ РАЗДЕЛ

3.1 Расчет затрат на разработку программы для микроконтроллера

Затраты на разработку программы для микроконтроллера рассчитываются по следующей формуле:

$$Z_{РПР} = Z_{ФОТР} + Z_{ОВФ} + Z_{ЭВМ} + Z_{СПП} + Z_{ХОН} + P_H \quad (11)$$

где $Z_{ФОТР}$ - общий фонд оплаты труда разработчиков программы,

$Z_{ОВФ}$ - начисления на заработную плату разработчиков программы во внебюджетные фонды,

$Z_{ЭВМ}$ - затраты, связанные с эксплуатацией техники,

$Z_{СПП}$ - затраты на специальные программные продукты, необходимые для разработки программы для микроконтроллера,

$Z_{ХОН}$ - затраты на хозяйственно-операционные нужды (бумага, носители информации и т.п.),

P_H - накладные расходы ($P_H = 30\%$ от $Z_{ФОТР}$).

При разработке программы для микроконтроллера общее время разработки составило 1 месяца.

Фонд оплаты труда за время работы над программой для микроконтроллера:

$$Z_{ФОТР} = \sum_{j=1}^m O_{Pj} \cdot T_{РПРj} \cdot (1 + k_D)(1 + k_Y), \quad (12)$$

где O_{Pj} – оклад j -го разработчика. В разработке участвовал 1 человек, его оклад составляет 18000 руб.,

$T_{РПРj}$ – общее время работы над программой в месяцах, $T_{РПР} = 1$,

k_D – коэффициент дополнительной зарплаты, $k_D = 0,1$,

k_Y – районный коэффициент, $k_Y = 0,15$.

Таким образом,

$$Z_{\text{ФОТР}} = 18000 \cdot 1 \cdot (1 + 0,1) \cdot (1 + 0,15) = 22770 \text{ руб.}$$

Страховой взнос во внебюджетные фонды складывается из взносов на обязательное социальное страхование от несчастных случаев на производстве и профессиональных заболеваний.

Страховые взносы складываются из обязательного пенсионного страхования (ОПС), отчислений в фонд социального страхования и отчислений в фонд обязательного медицинского страхования.

Значения всех используемых ставок приведены в таблице 4.

Таблица 4 – Значения ставок страховых взносов

№	Наименование внебюджетного фонда	Размер ставок, %
1	Пенсионный фонд	22
2	Фонд социального страхования	2,9
3	Федеральный фонд обязательного медицинского страхования	5,1
	Итого:	30

Сумма начислений на заработную плату во внебюджетные фонды составляет:

$$Z_{\text{ОВФ}} = 0,3 \cdot Z_{\text{ФОТР}}, \quad (13)$$

$$Z_{\text{ОВФ}} = 0,3 \cdot 22770 = 6831 \text{ руб.}$$

Затраты, связанные с использованием вычислительной и оргтехники:

$$Z_{\text{ЭВМ}} = T_{\text{МРПР}} \cdot k_{\Gamma} \cdot n \cdot C_{\text{М-ч}} \quad (14)$$

где k_{Γ} – коэффициент готовности ЭВМ, $k_{\Gamma} = 0,95$,

n – количество единиц техники, равно 1,

$C_{\text{М-ч}}$ – себестоимость машино–часа, $C_{\text{М-ч}} = 10 \text{ руб.}$,

$T_{\text{МРПР}}$ – машинное время работы над программой, равно 1 мес.

Перевод рабочего времени в часы осуществляется по формуле:

$$T_{\text{час}} = T_{\text{мес}} \cdot \text{Ч}_{\text{РД}} \cdot T_{\text{см}} \cdot K_{\text{см}}, \quad (15)$$

где $T_{\text{час}}$ – рабочее время, ч,

$T_{\text{мес}}$ – рабочее время, мес, ($T_{\text{мес}} = 2$),

$Ч_{\text{рд}}$ – число рабочих дней, ($Ч_{\text{рд}} = 22$),

$T_{\text{см}}$ – продолжительность рабочей смены, ($T_{\text{см}} = 8$ ч),

$K_{\text{см}}$ – количество рабочих смен, ($K_{\text{см}} = 1$).

Таким образом, время на разработку программы для микроконтроллера с использованием ЭВМ составляет:

$$T_{\text{час}} = 2 \cdot 22 \cdot 8 \cdot 1 = 352 \text{ часа,}$$

$$З_{\text{ЭВМ}} = 352 \cdot 0,95 \cdot 1 \cdot 10 = 3344 \text{ руб.}$$

Затраты на специальные программные продукты, необходимые для разработки программы для микроконтроллера рассчитываются по формуле:

$$З_{\text{СПП}} = \sum_{p=1}^n Ц_p \quad (16)$$

где $Ц_p$ – цена p -го специального программного продукта.

Перечень программных продуктов специального назначения приведен в таблице 5.

Таблица 5 – Программные продукты специального назначения

№	Название ПП	Цена, руб.
1	Среда разработки Arduino IDE	0
2	EasyEDA	0
	Итого:	0

Использованные программные продукты бесплатны, поэтому:

$$З_{\text{СПП}} = 0 \text{ руб.}$$

Затраты на хозяйственно–организационные нужды приведены в таблице 6 и вычисляются по формуле:

$$З_{\text{ХОИ}} = \sum_{\tau=1}^n Ц_{\tau} \cdot K_{\tau} \quad (17)$$

где C_{τ} – цена τ -го товара, руб.,

K_{τ} – количество τ -го товара.

Таблица 6 – Затраты на хозяйственно–организационные нужды

Наименование	Цена за единицу (руб.)	Кол-во (шт.)	Всего (руб.)
USB–флеш–накопитель	400	1	400
Бумага	0,5	140	70
Итого:			470

$$Z_{\text{ХОИ}} = 400 \cdot 1 + 0,5 \cdot 140 = 470 \text{ руб.}$$

Накладные расходы:

$$P_H = Z_{\text{ФОТР}} \cdot k_{\text{НР}}, \quad (18)$$

$$P_H = 22770 \cdot 0,3 = 6831 \text{ руб.}$$

Таким образом, затраты на разработку программы для микроконтроллера рассчитанные по формуле 1 составят:

$$Z_{\text{ППР}} = 22770 + 6831 + 3344 + 0 + 470 + 6831 = 40246 \text{ руб.}$$

3.2 Расчет затрат на внедрение программы для микроконтроллера

Затраты на внедрение программы для микроконтроллера ($Z_{\text{ВПР}}$) рассчитываются по формуле:

$$Z_{\text{ВПР}} = Z_M + Z_{\text{КТС}} \cdot (1 + k_{\text{ТУН}}) + Z_{\text{ПО}} + Z_{\text{ФОТВ}} + Z_{\text{ОФВ}} + Z_{\text{ЭВМ}} + P_{\text{КОМ}} + P_H, \quad (19)$$

где Z_M – затраты на приобретение материалов, руб.,

$Z_{\text{КТС}}$ – затраты на приобретение комплекса технических средств, руб.,

$Z_{\text{ПО}}$ – затраты на приобретение программного обеспечения (включают стоимость разработанной программы, а также других существующих ПП, необходимых для функционирования системы), руб.,

$Z_{\text{ФОТВ}}$ – затраты на оплату труда работников, занятых внедрением проекта, руб.,

$Z_{\text{ФВ}}$ – страховой взнос во внебюджетные фонды с заработной платы работников, занятых внедрением проекта, руб.,

$Z_{\text{ЭВМ}}$ – затраты, связанные с эксплуатацией ЭВМ при внедрении проектного решения, руб.,

$P_{\text{ком}}$ – командировочные расходы, руб.,

$P_{\text{Н}}$ – накладные расходы, руб.,

$k_{\text{тун}}$ – коэффициент транспортирования, установки и наладки комплекса технических средств, определяется действующими нормативами организации, а также спецификой конкретного проекта.

Затраты на приобретение материалов (ЗМ) приведены в таблице 7.

Таблица 7 – Затраты на приобретение материалов

Наименование	Цена за единицу (руб.)	Кол-во (шт.)	Всего (руб.)
Arduino Mega2560 R3	863,54	1	863,54
Макетная плата Snc	80,22	1	80,22
Модуль распознавания голоса V3	1510,58	1	1510,58
Сенсорный кнопочный модуль TTP223	7,16	1	7,16
Сервоприводы SG90	103,94	10	1039,44
Ультразвуковой датчик HC-SR04	54,12	1	54,12
Корпус для робота	2000	1	2000
Динамик	27,26	1	27,26
Аккумуляторы 18650	389,4	2	845
Соединительные провода типа папа-мама	4,9	8	39,2
Соединительные провода типа папа-папа	4,9	37	181,3
Итого:			6647,82

Дополнительного приобретения компьютеров или других КТС не требуется, следовательно, $Z_{\text{КТС}} = 0$.

Затраты на приобретение программного обеспечения в данном случае равны затратам на разработку и составляют $Z_{\text{ПО}} = 47100$ руб.

Внедрением занят один системный инженер с окладом 14000 руб. Время внедрения – 0,2 месяца. По формуле рассчитываем затраты на оплату труда и страховой взнос во внебюджетные фонды.

$$Z_{\text{ФОТВ}} = 14000 \cdot 0,2 = 2800 \text{ руб.}$$

$$Z_{\text{ОВФ}} = 2800 \cdot 0,3 = 840 \text{ руб.}$$

Затраты, связанные с эксплуатацией ЭВМ при внедрении проектного решения составят:

$$Z_{\text{ЭВМ}} = 0,2 \cdot 22 \cdot 8 \cdot 10 = 352 \text{ руб.}$$

Командировочные расходы при внедрении программы для микроконтроллера не планируются, следовательно, $R_{\text{ком}} = 0$.

Так как коэффициент накладных расходов по данным организации составляет $k_{\text{НР}} = 0,3$, то величина накладных расходов равна 352 руб.

Суммарные затраты на внедрение составят:

$$Z_{\text{ВПР}} = 6647,82 + 0 + 47100 + 2800 + 840 + 352 + 0 + 352 = 58091,82 \text{ руб.}$$

3.3 Расчет эксплуатационных текущих затрат по программе для микроконтроллера

Годовые затраты на обработку результатов до внедрения разработанной программы для микроконтроллера рассчитываются по формуле:

$$C_1 = 3\Pi_1 + OT_{\text{ВН1}} + Z_{\text{ЭВМ1}} + M_{\text{з1}} + \text{НР}_1, \quad (20)$$

где $3\Pi_1$ – затраты на оплату труда сотрудника на выполнение функций до внедрения проектного решения,

$OT_{\text{ВН1}}$ – страховой взнос во внебюджетные фонды,

$Z_{\text{ЭВМ1}}$ – затраты, связанные с эксплуатацией ЭВМ,

$M_{\text{з1}}$ – годовые материальные затраты на сопровождение программы для микроконтроллера составляют 1400 руб.,

НР_1 – накладные расходы.

Временные затраты работы сотрудника в месяцах рассчитываются по формуле:

$$T_{1\text{мес}} = \frac{T_{1\text{час}}}{\text{Ч}_{\text{рд}} \cdot \text{Ч}_{\text{рч}}}, \quad (21)$$

где $T_{1\text{мес}}$, $T_{1\text{час}}$ – время, затрачиваемое сотрудником на обработку результатов, в месяцах и часах соответственно ($T_{1\text{час}} = 564$ часов),

$\text{Ч}_{\text{рд}}$ – число рабочих дней в месяц,

$\text{Ч}_{\text{рч}}$ – число рабочих часов в день.

$$T_{1\text{мес}} = \frac{564}{22 \cdot 8} = 3,2 \text{ мес},$$

Тогда затраты на оплату труда сотрудника составят:

$$ЗП_1 = O_c \cdot T_{1\text{мес}} \cdot (1 + k_D) \cdot (1 + k_Y), \quad (22)$$

где O_c – оклад сотрудника (оклад составляет 14000 руб.),

$$ЗП_1 = 14000 \cdot 3,2 \cdot (1 + 0,1) \cdot (1 + 0,15) = 56672 \text{ руб.}$$

Страховой взнос до внедрения вычисляют по формуле:

$$OT_{\text{ВН1}} = ЗП_1 \cdot 0,3, \quad (23)$$

$$OT_{\text{ВН1}} = 56672 \cdot 0,3 = 17001,6 \text{ руб.}$$

Рассчитываем затраты, связанные с эксплуатацией ЭВМ до внедрения по формуле:

$$З_{\text{ЭВМ1}} = T_{1\text{час}} \cdot C_{\text{М-Ч}}, \quad (24)$$

$$З_{\text{ЭВМ1}} = 564 \cdot 10 = 5640 \text{ руб.}$$

Подставив соответствующие значения в формулу 10, получим:

$$C_1 = 56672 + 17001,6 + 5640 + 1400 = 80713,6 \text{ руб.}$$

Годовые затраты на эксплуатацию системы после внедрения программы для микроконтроллера рассчитываются аналогично по формуле:

$$C_2 = 3П_2 + ОТ_{ВН2} + З_{ЭВМ2} + М_{з2} + НР_2, \quad (25)$$

где $3П_2$ – затраты на оплату труда сотрудника после внедрения,
 $ОТ_{ВН2}$ – страховой взнос во внебюджетные фонды,
 $З_{ЭВМ2}$ – затраты, связанные с эксплуатацией ЭВМ после внедрения,
 $М_{з2}$ – материальные затраты, годовые материальные затраты на сопровождение программы для микроконтроллера составляют 1900 руб.,
 $НР_2$ – накладные расходы.

Временные затраты работы сотрудника в месяцах:

$$T_{2мес} = \frac{T_{2час}}{Ч_{рд} \cdot Ч_{рч}}, \quad (26)$$

где $T_{2мес}$, $T_{2час}$ – время, затрачиваемое сотрудником на обработку результатов, в месяцах и часах соответственно ($T_{2час} = 300$ часов),

$Ч_{рд}$ – число рабочих дней в месяц,

$Ч_{рч}$ – число рабочих часов в день.

$$T_{2мес} = \frac{300}{22 \cdot 8} = 1,70 \text{ мес.}$$

Тогда затраты на оплату труда сотрудника:

$$3П_2 = O_c \cdot T_{2мес} \cdot (1 + k_D) \cdot (1 + k_Y), \quad (27)$$

где O_c – оклад сотрудника (оклад составляет 14000 руб.)

$$3П_2 = 14000 \cdot 1,70 \cdot (1 + 0,1) \cdot (1 + 0,15) = 30107 \text{ руб.}$$

Страховой взнос после внедрения вычисляются по формуле:

$$ОТ_{ВН2} = 3П_2 \cdot 0,3, \quad (28)$$

$$ОТ_{ВН2} = 30107 \cdot 0,3 = 9032,1 \text{ руб.}$$

Рассчитываем затраты, связанные с эксплуатацией ЭВМ после внедрения по формуле:

$$Z_{ЭВМ2} = T_{2\text{час}} \cdot C_{M-ч}, \quad (29)$$

$$Z_{ЭВМ2} = 300 \cdot 10 = 3000 \text{ руб.}$$

Подставив соответствующие значения в формулу, получим:

$$C_2 = 30107 + 9032,1 + 3000 + 1900 = 44039,1 \text{ руб.}$$

Таким образом, текущие затраты на содержание системы до внедрения разработанной программы для микроконтроллера составляют 80713,6 руб., после внедрения 44039,1 руб.

3.4 Расчет экономической целесообразности разработки и внедрения информационных технологий

Для разрабатываемого проекта расчет экономической эффективности производится исходя из следующих условий:

- годовые текущие затраты до внедрения автоматизированной системы, $C_1 = 80713,6$ руб.,
- годовые текущие затраты после внедрения системы, $C_2 = 44039,1$ руб.,
- горизонт расчета принимается исходя из срока использования разработки, $T = T_n = 2$ годам,
- шаг расчета равен одному году, $t = 1$ году,
- капитальные вложения равны затратам на создание системы, $K = 58091,82$ руб.,
- норма дисконта равна норме дохода на капитал, $E = 12\%$.

Ожидаемая условно-годовая экономия от внедрения системы рассчитывается по формуле:

$$\mathcal{E}_{\text{уг}} = C_1 - C_2 + \sum \mathcal{E}_i, \quad (30)$$

где $\mathcal{E}_{\text{уг}}$ – ожидаемая условно-годовая экономия, руб.,

C_1 – годовые текущие затраты до внедрения автоматизированной системы, руб.,

C_2 – годовые текущие затраты после внедрения системы, руб.,

$\sum \mathcal{E}_i$ – ожидаемый дополнительный эффект от различных факторов, руб.

Так как основным фактором, по которому производится расчет экономической эффективности от внедрения программы для микроконтроллера, является уменьшение времени обработки результатов тестирования и дополнительный эффект не учитывается, то $\sum \mathcal{E}_i = 0$.

Подставив вычисленные выше значения в формулу, получим:

$$\mathcal{E}_{\text{уг}} = 80713,6 - 44039,1 = 36674,5 \text{ руб.}$$

где $\mathcal{E}_{\text{уг}}$ – ожидаемая условно-годовая экономия, руб.

Величина ожидаемого годового экономического эффекта от внедрения ИС рассчитывается по формуле:

$$\mathcal{E}_{\text{г}} = \mathcal{E}_{\text{уг}} - K \cdot E_{\text{н}}, \quad (31)$$

где $\mathcal{E}_{\text{г}}$ – ожидаемый годовой экономический эффект, руб.,

$\mathcal{E}_{\text{уг}}$ – ожидаемая условно-годовая экономия, руб.,

K – капитальные вложения (равны затратам на создание ИС), руб.,

$E_{\text{н}}$ – нормативный коэффициент экономической эффективности капитальных вложений.

Нормативный коэффициент экономической эффективности капитальных вложений определяется по формуле:

$$E_{\text{н}} = \frac{1}{T_{\text{н}}}, \quad (32)$$

где $T_{\text{н}}$ – нормативный срок окупаемости капитальных вложений, лет.

Подставив вычисленные выше значения в формулу, получим:

$$\mathcal{E}_2 = 36674,5 - 58091,82 \cdot 0,5 = 7628,59 \text{ руб.}$$

Расчетный коэффициент экономической эффективности капитальных вложений составляет:

$$E_p = \frac{\mathcal{E}_{y2}}{K}, \quad (33)$$

где E_p – расчетный коэффициент экономической эффективности капитальных вложений,

\mathcal{E}_{y2} – ожидаемая условно–годовая экономия, руб.,

K – капитальные вложения на создание системы, руб.

Подставив вычисленные выше значения в формулу, получим:

$$E_p = \frac{36674,5}{58091,82} = 0,63$$

Расчетный срок окупаемости капитальных вложений составляет:

$$T_p = \frac{1}{E_p}, \quad (34)$$

где E_p – коэффициент экономической эффективности капитальных вложений.

Подставив вычисленные выше значения в формулу, получим:

$$T_p = \frac{1}{0,63} = 1,6 \text{ год.}$$

Срок окупаемости без дисконтирования 1,6 год.

Чистый дисконтированный доход (ЧДД) определяется как сумма текущих эффектов за весь расчетный период, приведенная к начальному шагу, или как превышение интегральных результатов над интегральными затратами.

Если в течение расчетного периода не происходит инфляционного изменения цен или расчет производится в базовых ценах, то величина ЧДД для постоянной нормы дисконта вычисляется по формуле:

$$\text{ЧДД} = \sum_{t=1}^T (P_t - Z_t) \cdot \frac{1}{(1 + E)^t} - K, \quad (35)$$

где P_t – ожидаемые результаты от внедрения предложенной ИС, руб.,

Z_t – ожидаемые затраты (капитальные и текущие) на создание и эксплуатацию ИС, руб.,

$\Delta t = (P_t - Z_t)$ – эффект, достигаемый на t -м шаге расчета,

K – капитальные вложения,

t – номер шага расчета ($t = 1, 2$),

T – горизонт расчета,

E – постоянная норма дисконта, 12%.

$\Delta t = (P_t - Z_t) = \Delta_{\text{уг}} = 36674,5$ руб. В том случае, если текущие затраты (Z_t) на весь срок использования разработки равны 0.

$t = 1, 2$ год, т.к. предполагается, что результат от внедрения предложенной ИС будет с текущего года внедрения ИС.

Если ЧДД инвестиционного проекта положителен, то проект является эффективным (при данной норме дисконта).

Тогда суммарный чистый дисконтированный доход за весь горизонт расчета равен:

$$\begin{aligned} \text{ЧДД} &= \Delta_1 \cdot \frac{1}{(1 + E)} + \Delta_2 \cdot \frac{1}{(1 + E)^2}, \quad (36) \\ \text{ЧДД} &= \frac{36674,5}{(1 + 0,12)} + \frac{36674,5}{(1 + 0,12)^2} - 58097,82 = 3884 \text{ руб.} \end{aligned}$$

Положительное значение чистого дисконтированного дохода, $\text{ЧДД} > 0$, свидетельствует о том, что инвестирование целесообразно и данная ИС может приносить прибыль в установленном объеме.

Индекс доходности (ИД) представляет собой отношение суммы приведенных эффектов к величине капитальных вложений и определяется по формуле:

$$ИД = \frac{1}{K} \sum_{t=1}^T (P_t - Z_t) \cdot \frac{1}{(1 + E)^t} \quad (37)$$

где K – величина капиталовложений или стоимость инвестиций.

$$ИД = \frac{61981,82}{58091,82} = 1,07$$

Инвестиции считаются эффективными, если индекс доходности выше единицы, $ИД > 1$, следовательно, инвестиции в данную ИС, эффективны.

Внутренняя норма доходности (ВНД):

при $E_1 \rightarrow ЧДД_1 > 0$

$E_2 \rightarrow ЧДД_2 < 0$

$$ВНД = E_1 + \frac{ЧДД_1}{ЧДД_1 - ЧДД_2} \cdot (E_2 - E_1), \quad (38)$$

при $E_1 \rightarrow ЧДД_1 > 0$

$E_2 \rightarrow ЧДД_2 > 0$

$$ВНД = E_1 + \frac{ЧДД_1}{ЧДД_1 + ЧДД_2} \cdot (E_2 - E_1) \quad (39)$$

$E_1 = 0,11$

$$ЧДД_1 = \frac{36674,5}{(1 + 0,11)} + \frac{36674,5}{(1 + 0,11)^2} - 58097,82 = 4708,11 \text{ руб.}$$

$E_1 = 0,13$

$$ЧДД_2 = \frac{36674,5}{(1 + 0,13)} + \frac{36674,5}{(1 + 0,13)^2} - 58097,82 = 3079 \text{ руб.}$$

$$E_1 \rightarrow ЧДД_1 > 0$$

$$E_2 \rightarrow ЧДД_2 > 0$$

$$ВНД = 0,11 + \frac{4708,11}{4708,11 + 3079} \cdot (0,13 - 0,11) = 0,12.$$

Таким образом, норма дисконта должна быть в пределах 10% – 13%.

Показатели экономической целесообразности разработки и внедрения программы для микроконтроллера сведены в таблицу 8.

Таблица 8 – Показатели экономической целесообразности разработки и внедрения программы для микроконтроллера

Наименование показателя	Значения
Затраты на разработку и внедрение ПП, руб.	58091,82
Ожидаемая экономия от внедрения ПП, руб.	36674,5
Чистый дисконтированный доход, руб.	3884
Индекс доходности	1,07
Внутренняя норма доходности	0,12
Дисконтированный срок окупаемости, лет	1,6
Срок морального старения, года	2

Произведенные расчеты свидетельствуют, что внедрение, разработанной в ВКР программы для микроконтроллера, позволит сократить временные затраты на обработку результатов тестирования, что приведет к сокращению годовых текущих затрат на 36674,5 рублей

Опираясь на оценку экономической эффективности можно сделать вывод о том, что разработка и внедрение предлагаемой программы для микроконтроллера является экономически обоснованной и целесообразной.

4 ОХРАНА ТРУДА И ПРОМЫШЛЕННАЯ ДЕЯТЕЛЬНОСТЬ

4.1 Техника безопасности на рабочем месте инженера-электроника

4.1.1 Требования безопасности перед эксплуатацией

4.1.1.1 Если инженер-электроник сомневается в безопасности своего рабочего места, то он в праве не начинать свой трудовой день.

4.1.1.2 Перед тем как начать работу инженер-электроник должен убедиться в правильной организации своего рабочего места.

4.1.1.3 Прежде чем подключать электронные устройства к электросети, нужно убедиться в исправности розеток, вилок, проводов и кабелей, которые используются для связи между собой устройств.

4.1.1.4 Прежде чем начинать техническое обслуживание и ремонт электронных устройств инженер-электроник должен принять все меры предосторожности при работе, включая и меры электробезопасности, которые включают в себя следующее:

- отключите электронные приборы, чтобы не произошло внезапное включение электронного устройства во время технического обслуживания или ремонта;

- проверьте осталось ли напряжение в электронном устройстве с помощью измерительного прибора;

- в случае чего необходимо защитить оставшиеся токопроводящие части, а также нужно установить необходимые предупреждающие знаки безопасности.

4.1.1.5 Если не все нормы безопасности соответствуют указанным, то работник в праве не начинать рабочий день.

4.1.1.6 Прежде чем начинать работу инженер-электроник должен убедиться о наличии всего необходимого электрозащитного оборудования .

4.1.1.7 При работе с персональным компьютером нужно учитывать следующие указания:

- на уровень электромагнитного излучения влияют расположение компьютеров, особенно если они находятся вблизи;
- рекомендуемое расстояние между компьютерами должно составлять минимум 1,5 м, чтобы уменьшить уровень электромагнитного излучения;
- лучшим расположением для компьютера будет, если системный блок будет установлен либо к стене, либо в сторону пустого коридора;
- не следует ставить друг на против друга мониторы.

4.1.2 Требования безопасности при эксплуатации

4.1.2.1 Выполнять работы по техническому обслуживанию и ремонту следует в соблюдении инструкций по пользованию от производителей или по действующим документам.

4.1.2.2 Все электронные устройства должны быть подключены к электросети только фирменными проводами, что шли в комплекте с устройством.

4.1.2.3 Для ремонта или технического обслуживания электронного устройства инженер-электроник должен пользоваться средствами электрозащиты, также если расстояние изоляции очень мало, работнику не следует приближаться к этим частям.

4.1.2.4 Для проверки наличия напряжения в устройстве следует воспользоваться измерительным прибором заводского изготовления, но изначально нужно убедиться исправен ли этот прибор.

4.1.2.5 Все инструменты должны иметь токоизолирующую рукоятку, в частности отвертки должны иметь изоляционный стержень.

4.1.2.6 Если работы ведутся при использовании изолирующего оборудования и при этом без снятия напряжения, то работник должен следовать следующим правилам по технике безопасности:

- за изолирующие части у защитного оборудования можно держать только за ограничительный круг дабы избежать повреждений;

- защитное оборудование не должно перекрывать поверхность изоляции, в которых находятся токопроводящие провода, а также избегать короткого замыкания в землю;

- работая с изоляцией следует убедиться, что материалы используемые при работе сухие и чистые, и не поврежденные;

- если у защитного оборудования повредилось лакокрасочное покрытие, то с таким оборудованием не следует работать.

4.1.2.7 Чтобы не получить никаких электротравм не следует оставлять после разборки электронных устройств концы оголенных проводов.

4.1.2.8 Если есть необходимость пайки деталей или же проводов, с использованием оловянно-свинцового припоя, то следует знать и следовать следующим мерам предосторожности для повышения безопасности :

- можно получить отравление парами от припоя так, как припой содержит свинец, что опасен для человека;

- чтобы в глаза не попали частицы припоя следует пользоваться пинцетом, чтобы удерживать провода;

- чтобы избежать возгорания следует воспользоваться подставками под паяльник, и не следует его класть на стол.

4.1.2.9 Если работник пользуется ручным электроинструментом, то и здесь работник должен знать и следовать следующим мерам предосторожности электробезопасности:

- провода электроинструмента должны быть в подвешенном состоянии, чтобы не мешали работе;

- заранее подготовиться к возможным коротким замыканиям и устранить их;

- если по указанному руководству выявляется, что инструмент неисправен, то таким инструментом не следует пользоваться и даже подключать его к электросети;

– если во время работы с электроинструментом внезапно отключается электропитание, то стоит сразу же отключить его от сети, даже если время перерыва все равно следует отключить инструмент от сети.

4.1.2.10 Снизить утомляемость работнику может хорошо организованное рабочее место, только в комфортной для работника обстановке можно повысить его производительность и снизить вероятность повреждений.

4.1.2.11 Чтобы не утомлять глаза работнику следует установить цвета на экране такие, как светло-зеленый, желто-зеленый, оранжевый, желто-коричневые, такие меньше всего нагружают глаза; а вот такие контрастные цвета, как красный, синий и ярко-зеленый лучше не использовать, чтобы не нагружать глаза.

4.1.2.12 Чтобы уменьшить зрительную и скелетно-мышечную усталость, инженер-электроник должен придерживаться установленного режима работы и отдыха.

4.1.2.13 Обязательным для работника следует делать время от времени небольшие паузы в диапазоне от 1 до 3 минут, для того чтобы сильно не было зрительного напряжения.

4.1.2.14 Чтобы убрать монотонность с работы следует поочередно выполнять действия, иногда редактировать текст, иногда вводить какие-либо данные, иногда даже паять если необходимо.

4.1.3 Требования безопасности в аварийных ситуациях

4.1.3.1 При обнаружении нарушений требований охраны труда, которые создают угрозу здоровью или личной безопасности, инженер-электроник должен связаться с мастером и сообщить ему об этом; до устранения угрозы вы должны прекратить работу и покинуть опасную зону.

4.1.3.2 В случае несчастного случая, отравления, внезапного заболевания необходимо оказать пострадавшему немедленную первую

помощь, вызвать врача или помочь доставить пострадавшего к врачу, а затем сообщить управляющему зданием о происшествии.

4.1.3.3 При обнаружении пожара или признаков горения (дым, запах гари, повышение температуры и т.д.) необходимо немедленно уведомить службу пожарной безопасности по телефону 01.

4.1.4 Требования безопасности по окончании работы

4.1.4.1 Когда у работника заканчивается его смена ему следует обязательно отключить все электронные устройства от сети, а также всю документацию работник должен убрать и привести свое рабочее место в порядок, и т.д.[16].

4.2 Опасные производственные факторы на рабочем месте инженера-электроника

Потенциальные опасные и вредные производственные факторы на рабочем месте инженера-электронщика могут включать следующие типы опасных и вредных производственных факторов (таблица 9):

Таблица 9 - опасные и вредные производственные факторы на рабочем месте инженера-электронщика

Классификация опасных и вредных производственных факторов	Обозначение опасных и производственных факторов
Физические	повышенная температура поверхности оборудования и обрабатываемых материалов; недостаточное освещение на рабочем месте; повышенное электромагнитное излучение; повышенное значение напряжения в цепи, замыкание которой может произойти через тело человека.

Химические	повышенная концентрация паров расплавленного олова в воздухе рабочей зоны
Биологические	-
Психофизиологические	перенапряжение зрительного анализатора

Высокая температура поверхности оборудования и обрабатываемых материалов. Оборудование и принадлежности, нагретые до температуры пайки, могут вызвать ожоги при случайном прикосновении или при расплавлении материала в процессе пайки. Такие травмы обычно возникают при нарушении технологической системы, при неправильной организации работ по пайке.

Недостаточное освещение на рабочем месте. Освещение производственных помещений вышеупомянутых предприятий оказывает значительное влияние на производительность и качество работ. Неудовлетворительное промышленное освещение приводит к повышенной утомляемости, снижению остроты зрения и, как следствие, снижению качества выполняемых работ, особенно в процессе пайки.

На таком рабочем месте используется комбинированное освещение.

Источниками электромагнитного излучения являются системные блоки и мониторы, а также паяльники, которые используются в качестве нагревательного оборудования

Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека. При работе паяльного оборудования используется напряжение 220 В. В случае повреждения изоляции опасное напряжение человека может возникнуть в непроводящих частях, например, на корпусе устройства, и привести к поражению электрическим током паяльника (ожоги, поражения электрическим током и т.д.).

Повышенная концентрация паров расплавленного олова в воздухе рабочей зоны. Источниками вредных химических веществ, попадающих в воздушную среду, являются припой, изготовленные из олова, и флюсы. Таким образом, олово проникает через пары в организм человека, и при плохом соблюдении требований безопасности эти вещества могут вызвать профессиональные заболевания или отравления.

Кроме того, при подготовке сборочного узла к пайке элементов с соединяемой поверхности удаляют оксидную пленку, масло и другие загрязнения, что достигается механической зачисткой, обезжириванием и химическим травлением. Обезжиренные и протравленные растворы обычно содержат вещества, которые, попадая в организм человека, могут вызвать отравление или ожоги. [19].

4.3 Методы утилизации вычислительной техники

Теоретически, есть три способа решить эту проблему: продать, передать или отменить и утилизировать оборудование. Продать - этот вариант верен только в том случае, если есть покупатель и ремонтпригодность самого оборудования. Но обычно списывают только самые ненужные из них.

Продажа или передача старого оборудования школе или детскому саду также не всегда является лучшим вариантом. Качество старого оборудования довольно низкое, и во многих случаях его больше нельзя использовать нормально.

Есть еще вариант – утилизация компьютерной и оргтехники. Но организация должна провести вывоз оборудования, подтвердив факт передачи компьютеров и оргтехники.

Стандартный процесс утилизации списанного и компьютерного оборудования включает в себя следующие этапы:

– компания, целью которой является списания устройств, создаст специальную внутреннюю комиссию. Его главная задача - принять коллегиальное решение о типе оборудования, которое пора списывать;

– это решение комиссии о выводе из эксплуатации компьютеров и оргтехники обязательно должно основываться на экспертном заключении. Эксперт может быть либо штатным сотрудником компании, либо независимым экспертом, привлеченным извне. Вам определенно нужен производитель компьютерной или офисной техники или другая компания с лицензией на обслуживание и ремонт этого оборудования. Такой акт технической экспертизы компьютеров и оборудования документирует, что оборудование неисправно, его ремонт нецелесообразен и настало время для его выхода на пенсию. Старое оборудование может быть списано и утилизировано;

– для того, чтобы вывод из эксплуатации офисного оборудования и компьютеров был окончательно завершен и забыт, также необходимо предоставить документальные доказательства того, что они были утилизированы надлежащим образом, без дальнейшего разрушения нашей экосистемы, и что они были разложены на тяжелые металлы и токсичные соединения [20].

В разделе «Охрана труда и промышленная деятельность» были рассмотрены следующие подразделы:

– в подразделе «Техника безопасности на рабочем месте инженера-электроника» были рассмотрены основные аспекты техники безопасности до начала рабочего дня, во время рабочего дня, в аварийных случаях и в конце рабочего дня;

– в подразделе «Опасные производственные факторы на рабочем месте инженера-электроника» были рассмотрены опасные и вредные факторы на рабочем месте инженера-электроника;

– в подразделе «Методы утилизации вычислительной техники» были рассмотрены известные методы утилизации техники и по каким причинам устраняют технику.

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы был разработан четырехногий шагающий робот с адаптивным управлением на базе микроконтроллера. Был выполнен анализ существующих на рынке шагающих роботов. И по этим роботам был спроектирован собственный робот.

Был выполнен подбор инструментально обеспечения проекта, а также подбор элементной базы. Которые были использованы в проекте. Сделана сравнительная таблица элементов робота.

Были разработаны функциональная, структурная и принципиальная схемы, а также разработана управляющая программа для контроллера робота. Для это были изучены программы для проектирования схем четырехногого шагающего робота и написания программы для контроллера.

Был проведен расчет технико-экономических показателей проекта. Были рассчитаны затраты на материалы, чтобы вывести себестоимость проекта.

Разработанное устройство может использоваться частными пользователями.

На основании данной работы может быть разработана более лучшая версия четырехногого робота, как в плане выполнения команд, так и в количестве используемых датчиков.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Подураев Ю. В. Мехатроника: основы, методы, применение. — 2-е. — М.: «Машиностроение», 2007. — С. 10. — 256 с.
- 2 Б. М. Готлиб. Введение в мехатронику. Учебное пособие. — Екатеринбург: Уральский государственный университет путей сообщения, 2007. — С. 8. — 782 с.
- 3 Шилин А.Н., Шилин А.А., Сницарук Д.Г. Программирование микроконтроллеров измерительных приборов: учебное пособие – Волгоградский государственный технический университет: г. Волгоград, 2018. – 109 с.
- 4 Носкова Е.Д. Основы мехатроники: учебное пособие – Губернаторский авиастроительный колледж г. Комсомольска – на – Амуре, 2020. - 36 с.
- 5 Мехатронные системы в управлении механическим движением и функциями технологических машин [Электронный ресурс] URL: https://studref.com/501307/tehnika/mehatronnye_sistemy_upravlenii_mehanicheskim_dvizheniem_funktsiyami_tehnologicheskikh_mashin (Дата обращения 28.02.2022г.)
- 6 Шагающие роботы - теория и основы [Электронный ресурс] URL: <https://monitorbank.ru/shagayushhie-roboty-teoriya-i-osnovy/> (Дата обращения 07.03.2022г.)
- 7 Шагающие роботы [Электронный ресурс] URL: <http://roboticslib.ru/books/item/f00/s00/z0000016/st027.shtml> (Дата обращения 07.03.2022г.)
- 8 Первая отечественная робособака [Электронный ресурс] URL: https://www.cnews.ru/news/top/2021-10-22_v_rossii_sozdana_pervaya_otchestvennaya (Дата обращения 07.03.2022г.)

- 9 Анализ походки четвероного робота [Электронный ресурс] URL: <https://cjme.springeropen.com/articles/10.1186/s10033-019-0321-2> (Дата обращения 08.03.2022г.)
- 10 Робопес CyberDog [Электронный ресурс] URL: <https://habr.com/ru/news/t/572224/> (Дата обращения 08.03.2022г.)
- 11 Инверсная кинематика [Электронный ресурс] URL: <http://www.gamedev.ru/articles/?id=30035> (Дата обращения 09.03.2022г.)
- 12 Инверсная кинематика в двухмерном пространстве [Электронный ресурс] URL: <https://habr.com/ru/post/358798/> (Дата обращения 09.03.2022г.)
- 13 Дизайн и разработка автономных четвероногих роботов [Электронный ресурс] URL: <https://iopscience.iop.org/article/10.1088/1757-899X/1012/1/012016/pdf> (Дата обращения 09.03.2022г.)
- 14 Кинематический анализ и моделирование четвероного робота [Электронный ресурс] URL: <https://ieeexplore.ieee.org/document/6045874> (Дата обращения 09.03.2022г.)
- 15 Адаптивные системы управления [Электронный ресурс] URL: https://studopedia.ru/4_145405_adaptivnie-sistemi-upravleniya.html (Дата обращения 28.02.2022г.)
- 16 Инструкция по охране труда для инженера-электроника [Электронный ресурс] URL: http://knacits.ru/images/Glavnaya/Ohrana_truda/ohrana_truda/Instrykcii/23.pdf (Дата обращения 17.04.2022г.)
- 17 Идентификация опасных и вредных производственных факторов на рабочем месте инженера-электронщика [Электронный ресурс] URL: https://studbooks.net/1510896/bzhd/identifikatsiya_opasnyh_vrednyh_proizvodstvennyh_faktorov_rabochem_meste_inzhenera_elektronschika (Дата обращения 17.04.2022г.)
- 18 Кукин П.П., Лапин В.Л. Безопасность жизнедеятельности. Безопасность технологических процессов и производств. Охрана труда. - М.: Высшая школа, 2002. - 318с

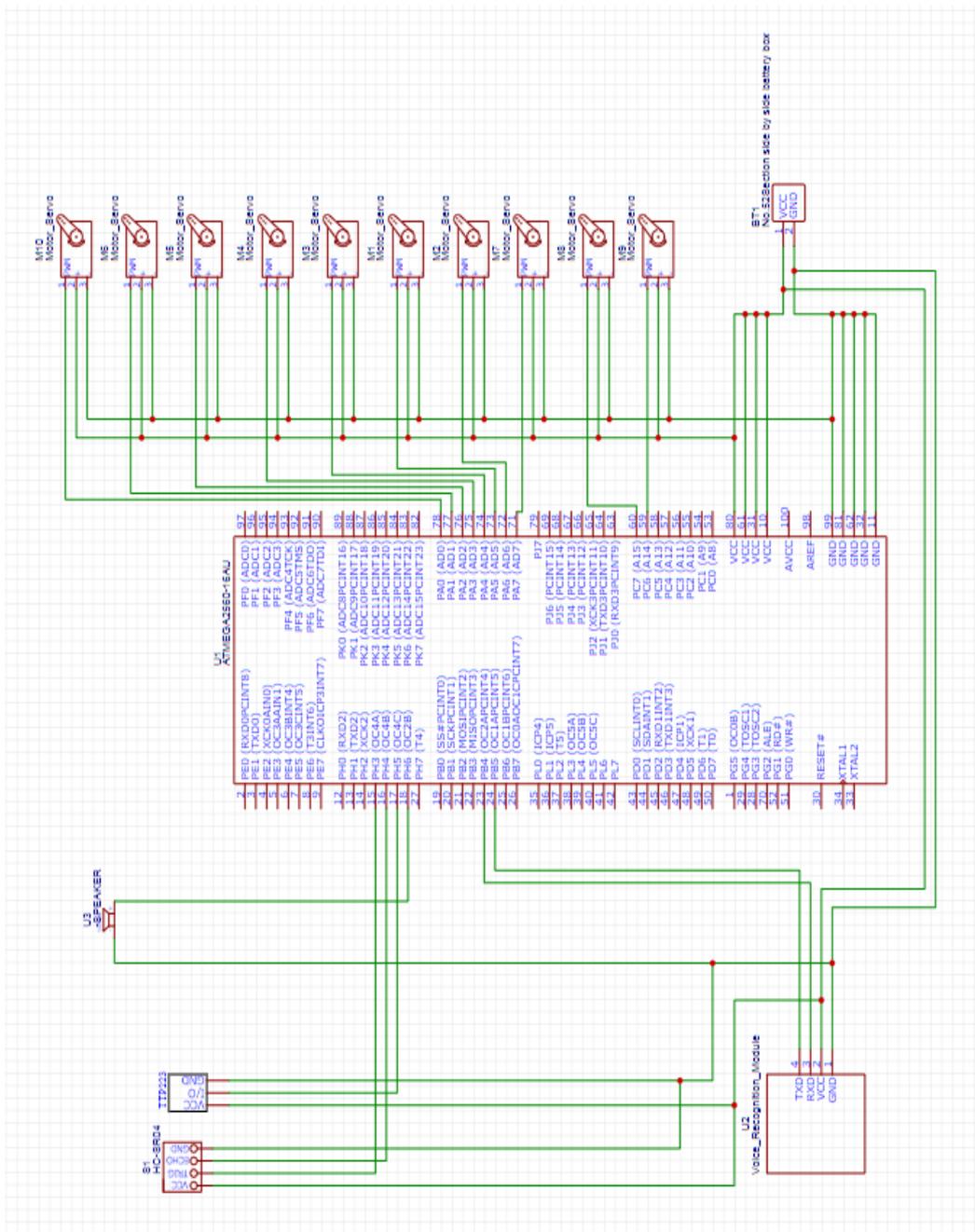
19 Чекунов И.П. Безопасность труда паяльщика.-- М.: Машиностроение, 1992.-- 57с.

20 Как утилизировать компьютерную технику? [Электронный ресурс] URL: <https://gupecosistem.ru/kak-utilizirovat-kompyuternuyu-tehniku/#i-35> (Дата обращения 18.04.2022г.)

21 Atmega2560 datasheet [Электронный ресурс] URL: <https://static.chipdip.ru/lib/934/DOC011934714.pdf> (Дата обращения 18.04.2022г.)

22 HC-SR04 datasheet [Электронный ресурс] URL : <https://static.chipdip.ru/lib/092/DOC001092302.pdf> (Дата обращения 18.04.2022г.)

Приложение А - Принципиальная схема



Приложение Б - Программный код

```
#include <Servo.h>
#include "VoiceRecognitionV3.h"

// F = Forward, B = Back, L = Left, R = Right, 1 = Top, 2 = Bottom
Servo Head;
Servo FL1;
Servo FL2;
Servo BL1;
Servo BL2;
Servo FR1;
Servo FR2;
Servo BR1;
Servo BR2;
Servo Tail;
int front_left_top_angle;
int front_right_top_angle;
int front_left_bottom_angle;
int front_right_bottom_angle;
int back_left_top_angle;
int back_right_top_angle;
int back_left_bottom_angle;
int back_right_bottom_angle;
byte com;
const int trig = 8;
const int echo = 9;
unsigned int time_us = 0;
unsigned int distance_sm = 0;
unsigned int distance_sm_right = 0;
```

```
unsigned int distance_sm_left = 0;
```

```
VR myVR(10,11); // 2:RX 3:TX, you can choose your favourite pins.
```

```
uint8_t records[7]; // save record
```

```
uint8_t buf[64];
```

```
#define onRecord (0)
```

```
#define offRecord (1)
```

```
#define sitRecord (2)
```

```
#define goRecord (3)
```

```
#define lieRecord (4)
```

```
#define standRecord (5)
```

```
#define voiceRecord (6)
```

```
void printSignature(uint8_t *buf, int len)
```

```
{
```

```
int i;
```

```
for(i=0; i<len; i++){
```

```
if(buf[i]>0x19 && buf[i]<0x7F){
```

```
Serial.write(buf[i]);
```

```
}
```

```
else{
```

```
Serial.print("[");
```

```
Serial.print(buf[i], HEX);
```

```
Serial.print("]");
```

```
}
```

```
}
```

```
}
```

```

void printVR(uint8_t *buf)
{
    Serial.println("VR Index\tGroup\tRecordNum\tSignature");

    Serial.print(buf[2], DEC);
    Serial.print("\t\t");

    if(buf[0] == 0xFF){
        Serial.print("NONE");
    }
    else if(buf[0]&0x80){
        Serial.print("UG ");
        Serial.print(buf[0]&(~0x80), DEC);
    }
    else{
        Serial.print("SG ");
        Serial.print(buf[0], DEC);
    }
    Serial.print("\t");

    Serial.print(buf[1], DEC);
    Serial.print("\t\t");
    if(buf[3]>0){
        printSignature(buf+4, buf[3]);
    }
    else{
        Serial.print("NONE");
    }
    Serial.println("\r\n");
}

```

```

void setup() {
  Head.attach(22);
  FL1.attach(24);
  FL2.attach(25);
  BL1.attach(26);
  BL2.attach(27);
  FR1.attach(28);
  FR2.attach(29);
  BR1.attach(30);
  BR2.attach(31);
  Tail.attach(23);
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(7,OUTPUT);
  pinMode(6, INPUT_PULLUP);
  /** initialize */
  myVR.begin(9600);

  Serial.begin(115200);
  Serial.println("Elechouse Voice Recognition V3 Module\r\nControl LED
sample");

```

```

if(myVR.clear() == 0){
  Serial.println("Recognizer cleared.");
}else{
  Serial.println("Not find VoiceRecognitionModule.");
  Serial.println("Please check connection and restart Arduino.");
}

```

```
    while(1);
}

if(myVR.load((uint8_t)onRecord) >= 0){
    Serial.println("onRecord loaded");
}

if(myVR.load((uint8_t)offRecord) >= 0){
    Serial.println("offRecord loaded");
}
}
```

```
void stand()
{
    Head.write(90);
    FL1.write(90);
    FL2.write(90);
    FR1.write(90);
    FR2.write(90);
    BL1.write(90);
    BL2.write(90);
    BR1.write(90);
    BR2.write(90);
    Tail.write(90);
}
```

```
void go()
{
```

```
front_left_top_angle = 90;
front_right_top_angle = 90;
front_left_bottom_angle = 90;
front_right_bottom_angle = 90;
back_left_top_angle = 90;
back_right_top_angle = 90;
back_left_bottom_angle = 90;
back_right_bottom_angle = 90;
BL2.write(105);
delay(100);
BL1.write(75);
delay(100);
BL2.write(75);
delay(100);
FR2.write(105);
delay(100);
FR1.write(105);
delay(100);
FR2.write(45);
delay(100);
BR2.write(105);
delay(100);
BR1.write(75);
delay(100);
BR2.write(75);
delay(100);
BL2.write(90);
delay(100);
BL1.write(90);
delay(100);
```

```

FL2.write(105);
delay(100);
FL1.write(105);
delay(100);
FL2.write(45);
FR2.write(90);
delay(100);
FR1.write(90);
delay(100);
BR2.write(90);
delay(100);
BR1.write(90);
delay(100);
FL2.write(90);
delay(100);
FL1.write(90);
delay(100);
digitalWrite(trig, HIGH); //Подаем сигнал на ВЫХОД

```

микроконтроллера

```

delayMicroseconds(10); //Удерживаем 10 мкс
digitalWrite(trig, LOW); //Затем убираем
time_us = pulseIn(echo, HIGH); //Замеряем длину импульса
distance_sm = time_us / 58; //Пересчитываем в сантиметры (для

```

данного датчика

```

//постоянная для перевода 58)
if(distance_sm <=20 ){
  Head.write(0);
  digitalWrite(trig, HIGH); //Подаем сигнал на ВЫХОД

```

микроконтроллера

```

delayMicroseconds(10); //Удерживаем 10 мкс

```

```
digitalWrite(trig, LOW); //Затем убираем
time_us = pulseIn(echo, HIGH); //Замеряем длину импульса
distance_sm_right = time_us / 58; //Пересчитываем в сантиметры
```

(для данного датчика

```
delay(1000);
Head.write(180);
digitalWrite(trig, HIGH); //Подаем сигнал на выход
```

микроконтроллера

```
delayMicroseconds(10); //Удерживаем 10 мкс
digitalWrite(trig, LOW); //Затем убираем
time_us = pulseIn(echo, HIGH); //Замеряем длину импульса
distance_sm_left = time_us / 58; //Пересчитываем в сантиметры (для
```

данного датчика

```
if(distance_sm_right > distance_sm_left){
    povorot_napravo();
}
else {
    povorot_nalevo();
}
}
```

```
void povorot_napravo(){
```

```
    int i;
    for(i=0; i<3; i++){
        BL2.write(105);
        delay(100);
        BL1.write(75);
        delay(100);
        BL2.write(75);
```

```
delay(100);
FR2.write(105);
delay(100);
FR1.write(96);
delay(100);
FR2.write(45);
delay(100);
BR2.write(105);
delay(100);
BR1.write(84);
delay(100);
BR2.write(75);
delay(100);
BL2.write(90);
delay(100);
BL1.write(90);
delay(100);
FL2.write(105);
delay(100);
FL1.write(105);
delay(100);
FL2.write(45);
FR2.write(90);
delay(100);
FR1.write(90);
delay(100);
BR2.write(90);
delay(100);
BR1.write(90);
delay(100);
```

```
FL2.write(90);  
delay(100);  
FL1.write(90);  
delay(100);  
}  
}
```

```
void povorot_nalevo(){
```

```
    int i;  
    for(i=0; i<3; i++){  
        BL2.write(105);  
        delay(100);  
        BL1.write(86);  
        delay(100);  
        BL2.write(75);  
        delay(100);  
        FR2.write(105);  
        delay(100);  
        FR1.write(105);  
        delay(100);  
        FR2.write(45);  
        delay(100);  
        BR2.write(105);  
        delay(100);  
        BR1.write(90);  
        delay(100);  
        BR2.write(75);  
        delay(100);  
        BL2.write(90);
```

```
    delay(100);
    BL1.write(90);
    delay(100);
    FL2.write(105);
    delay(100);
    FL1.write(96);
    delay(100);
    FL2.write(45);
    FR2.write(90);
    delay(100);
    FR1.write(90);
    delay(100);
    BR2.write(90);
    delay(100);
    BR1.write(90);
    delay(100);
    FL2.write(90);
    delay(100);
    FL1.write(90);
    delay(100);
    }
}
```

```
void sit()
{
    front_left_top_angle = 45;
    front_right_top_angle = 45;
    front_left_bottom_angle = 45;
    front_right_bottom_angle = 45;
    back_left_top_angle = 45;
```

```

back_right_top_angle = 45;
back_left_bottom_angle = 135;
back_right_bottom_angle = 135;
boolean valuebutton = digitalRead(6);
if (valuebutton == HIGH)
{
  Tail.write(45);
  delay(100);
  Tail.write(135);
  delay(100);
  Tail.write(45);
  delay(100);
  Tail.write(135);
  delay(100);
  Tail.write(45);
  delay(100);
  Tail.write(135);
  delay(100);
  Tail.write(90);
  delay(100);
}
}

```

```

void lie()
{
  front_left_top_angle = 45;
  front_right_top_angle = 45;
  front_left_bottom_angle = 135;
  front_right_bottom_angle = 135;
  back_left_top_angle = 45;

```

```
back_right_top_angle = 45;
back_left_bottom_angle = 135;
back_right_bottom_angle = 135;
}
```

```
void voice()
{
tone(7, 240, 50);
delay(10000);
}
```

```
void loop() {

int ret;
ret = myVR.recognize(buf, 50);
if(ret>0){
switch(buf[6]){
case onRecord:
/** turn on LED */
digitalWrite(22, HIGH);
digitalWrite(23, HIGH);
digitalWrite(24, HIGH);
digitalWrite(25, HIGH);
digitalWrite(26, HIGH);
break;
case offRecord:
/** turn off LED*/
digitalWrite(2, LOW);
break;
}
```

```
case sitRecord:
    sit();
    break;
case goRecord:
    go();
    break;
case lieRecord:
    lie();
    break;
case standRecord:
    stand();
    break;
case voiceRecord:
    voice();
    break;
default:
    Serial.println("Record function undefined");
    break;
}
/** voice recognized */
printVR(buf);
}
}
```