

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»**

Кафедра динамического моделирования и биомедицинской инженерии

**Учебный практикум по программированию на языке Си:
«Исполнитель – колесный робот с микроконтроллерным
управлением»**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 461 группы

направления 12.03.04 «Биотехнические системы и технологии»

факультета Нано- и биомедицинских технологий

Курбако Александра Васильевича

Научный руководитель

доцент, к.ф.-м.н. _____

Караваяев А.С.

Зав. кафедрой

профессор, д.ф.-м.н. _____

Селезнев Е.П.

Саратов 2019 г.

Введение. Довольно трудно представить современную медицину без различных биотехнических устройств. К ним можно отнести электромеханические медицинские изделия, общепольничное оборудование, вспомогательные изделия для пациентов с ограниченными возможностями, лабораторные оборудования и многое другое. Однако, для того, чтобы конструировать, использовать, обслуживать и ремонтировать биотехнические системы, необходим квалифицированный специалист.

В процессе обучения студент должен получить необходимые навыки и знания, которые помогут ему в дальнейшем при работе с биотехническими устройствами, а так же научиться применять навыки и знания, полученные в ходе лекционных занятий. К сожалению, в настоящее время известно малое количество практикумов, способных дать будущему специалисту необходимые навыки.

В связи с этим, **целью** моей работы было создание лабораторной установки колесного робота – исполнителя и комплекса программных средств для поддержки серии практикумов для получения необходимых в сфере биоинженерии навыков и умений.

Для решения поставленной цели были сформулированы следующие **задачи**:

1. Проектирование и создание колесного шасси на основе коллекторных моторов постоянного тока;
2. Конструирование и сборка схемы управления для шасси на основе микроконтроллера;
3. Разработка микропрограммного обеспечения для управляющего микроконтроллера;
4. Создание библиотеки на языке Си для использования колесного робота в качестве исполнителя;
5. Отладка и настройка всей системы.

Колесный робот по своей структуре имеет блоки, соответствующие блокам реальной биотехнической системы. Как и любое по сложности

биотехническое устройство, исполнитель включает в себя блок питания, датчики различных физических величин, подвижные электромеханические детали, которые нуждаются в точном управлении, схемы обратной связи и устройство для обработки сигналов и управления (микроконтроллер). Используя колесного робота для обучения, будущий специалист не только освоит навыки работы с отдельными блоками, но и получит опыт по отладке и работе со сложными системами, состоящими из множества связанных блоков.

Полученный по итогу колесный робот – исполнитель будет использован на трех этапах обучения:

1. Обучение студентов начальных курсов основам программирования на языке Си (или на любом другом языке программирования) с использованием колесного робота в качестве исполнителя;

2. Обучение студентов магистратуры работе с микроконтроллерами на примере сложной системы, состоящей из множества блоков;

Обучение студентов комплексной работе со сложным биотехническим устройством и компьютером, создание аппаратно-программного комплекса.

Основное содержание работы. На настоящий момент на рынке известно небольшое количество практикумов, способных помочь будущему специалисту получить навыки, необходимые для работы с биотехническими системами. Среди этих практикумов можно выделить следующие типы наиболее близких по значению и получивших наибольшее распространение:

- 1) Макетные установки;
- 2) Блочные конструкторы;
- 3) Программные исполнители;
- 4) Эмуляторы.

Макетная установка представляет собой макетную плату с различными приборами для изучения работы аналоговой или цифровой схемы, различные электромеханические блоки, из которых составляется сложная система, и питание. Макетные установки успешно применяются для решения ряда задач: конструирование электрических аналоговых и цифровых схем, сборка систем

из блоков, написание микропрограммного обеспечения для микроконтроллеров и микрокомпьютеров и др. Однако, работа на установке, состоящей из блоков, не позволяет конструировать и отлаживать готовые устройства. Так же макетная установка ограничивает возможности работы с мехатронными системами.

Блочный конструктор состоит из набора электрических и электромеханических блоков, которые крепятся к основному блоку управления. Блок управления представляет собой микрокомпьютер в корпусе с возможностью крепления и подключения к нему других блоков. На блоке управления собирается электромеханическая система, а в микрокомпьютер загружается написанное на одном из языков программирования микропрограммное обеспечение. При помощи блочного конструктора появляется возможность отладки и настройки готового устройства. Однако конструктор так же имеет ряд недостатков, не позволяющих студенту в полной мере освоить необходимые навыки. Во первых, блоки в такой системе практически идеально подходят друг к другу, не возникает проблем с совмещением устройств в готовую систему, что необходимо при создании реального устройства. Во вторых, возможности работы с управляющим блоком ограничены самой компанией LEGO. Так же стоит заметить, что в реальной жизни не встречаются устройства, построенные на основе подобных блоков, и практически не используется микропрограммное обеспечение, написанное под блок управления подобной системы.

Программный исполнитель – это компьютерная программа, которая при помощи виртуального исполнителя выполняет программный код, написанный на одном из распространённых языков программирования или на внутреннем языке среды разработки. Среда разработки представляет собой текстовый редактор, в котором составляется тело программы, и область, в которой исполнитель выполняет команды, написанные в коде. Таким образом, программные исполнители направлены на получение навыков работы с алгоритмами, изучение структуры программы и знакомство с синтаксисом

языка, на котором пишется код. Виртуальный исполнитель способствует лучшему пониманию принципов программирования. Однако исполнитель является идеализированной моделью и не имеет проблем, которые встречаются у реальных объектов, что уменьшает область их применения.

Эмулятор – это компьютерная программа, направленная на воспроизведение программными или аппаратными средствами, либо их комбинацией, работы других программ или устройств. При помощи эмуляторов возможно создавать виртуально относительно сложные системы для их конструирования, тестирования и отладки. В качестве сложной системы может выступать электрическая схема, механическая конструкция, микропроцессорная система и др. Эмуляторы успешно используются для изучения и проектирования электрических аналоговых и цифровых схем. Помимо этого, эмуляторы дают возможность составлять и отлаживать микропрограммное обеспечение для микроконтроллеров. Однако одним из главных недостатков является идеализированность эмулируемых элементов, что исключает выявление некоторых проблем, возникающих в реальной системе.

Из всего выше сказанного можно сделать вывод, что на настоящее время не существует идеально подходящих практикумов, позволяющих будущему специалисту комплексно получить необходимые навыки работы со сложными биотехническими системами.

Учебный практикум по программированию на языке Си, описанный в дипломной работе, состоит из двух частей: исполнителя - колесного робота и библиотеки на языке Си.

Задачей исполнителя является выполнение команд, посланных с компьютера или любого другого устройства. Колесный робот способен исполнять следующие команды:

- 1) Сделать шаг вперед (проезд на 10 сантиметров вперед);
- 2) Сделать шаг назад (проезд на 10 сантиметров назад);
- 3) Поворот налево (на 45 градусов);
- 4) Поворот направо (на 45 градусов);

5) Замер расстояния до препятствия впереди исполнителя (результат измерения отправляется на компьютер).

Основными частями исполнителя являются колесное шасси, плата управления и аккумуляторное питание.

Колесное шасси представляет собой платформу на 3х колесах. Два ведущих колеса расположены соосно сзади и приводятся в движение коллекторными моторами-редукторами постоянного тока. Третье опорное колесо расположено впереди по центру, имеет подшипник и способно свободно вращаться на 360 градусов. Снизу под платформой располагаются аккумуляторные батареи. В ходе обзора существующих компоновок колесных шасси была выбрана именно эта из-за лучших характеристик маневренности, заносоустойчивости, устойчивости, простоты в изготовлении и маленькой себестоимости.

Основной частью колесного робота является плата управления. Она выполняет следующие функции:

- 1) Управление коллекторными моторами (изменение скорости и направления вращения);
- 2) Измерение расстояния до препятствия впереди;
- 3) Измерение напряжения на аккумуляторе для определения степени разряженности;
- 4) Прием и передача информации на компьютер по беспроводной связи.

Плата управления состоит из следующих компонентов (рисунок 1): микроконтроллер (1), датчик беспроводной связи (2), датчик расстояния (3), токовый драйвер (4), датчик обратной связи от колес (5), индикаторные светодиоды (7), делитель напряжения с аккумуляторов (8) и схема развязки питания (6).

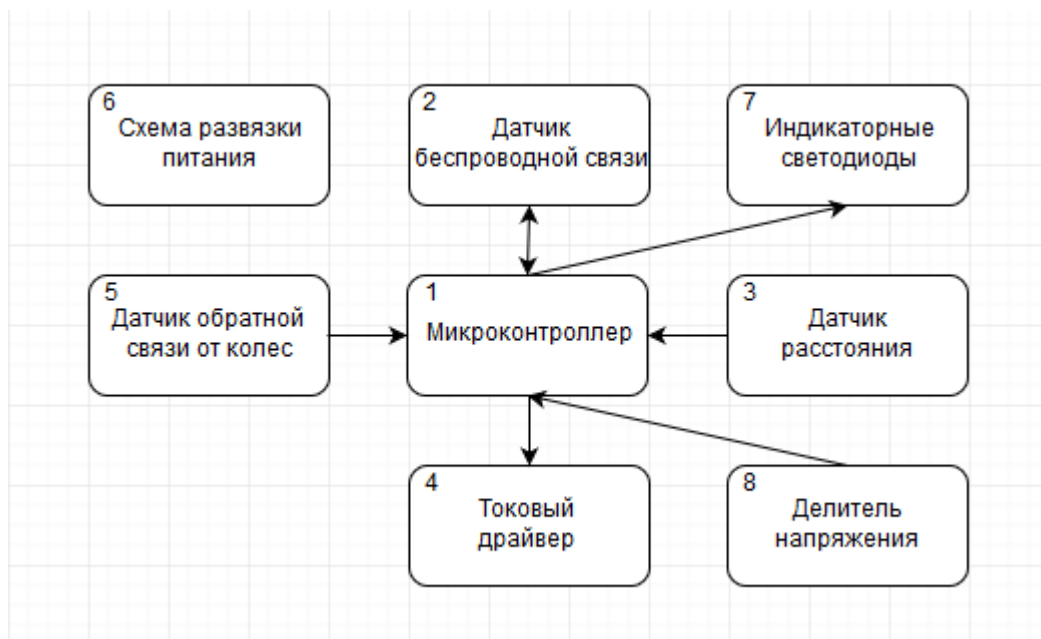


Рисунок 1 - Основные компоненты платы управления

Микроконтроллер был выбран фирмы AVR ATmega48PA-PU по соотношению цена\необходимый функционал. В качестве датчика беспроводной связи был выбран bluetooth-модуль HC-05. Датчиком расстояния является ультразвуковой дальномер HC-SR04. Токвый драйвер L293D используется для управления скоростью и направлением вращения мотор-редукторами. Датчик обратной связи от колес построен на основе оптических колесных энкодеров, способных измерять угол поворота колеса и скорость его вращения. Индикаторные светодиоды используются для отображения различных процессов, а именно красный светодиод индицирует о наличии питания на плате управления, а желтый оповещает о разряде аккумуляторных батарей. Для измерения степени заряженности аккумуляторных батарей на плате предусмотрен делитель напряжения, подключенный к АЦП микроконтроллера. Вся схема управления для питания нуждается в стабилизированном напряжении постоянного тока 3.3 и 5 вольт, что обеспечивается схемой питания на основе линейных стабилизаторов.

Микроконтроллер, являющийся основой платы управления, должен собирать информацию с датчиков (bluetooth-модуля, колесных энкодеров,

датчика расстояния и АЦП) и в зависимости от нее управлять работой двигателей или выполнять различные действия. Для этого было составлено микропрограммное обеспечение для микроконтроллера на языке Си. При подачи питания на микроконтроллера происходит инициализация портов, таймеров, прерываний и др. Далее в алгоритме предусмотрена задержка на 2.5 секунды для установки колесного робота на площадку, где он будет работать. После задержки проводится автокалибровка, в ходе которой исполнитель проезжает вперед и назад на 1.5 метра для подбора скоростей вращения моторов таким образом, чтобы обеспечить прямолинейность езды с минимальными отклонениями. После калибровки исполнитель отправляет на компьютер сообщение об окончании калибровки и переходит в режим ожидания команд. Когда на исполнитель приходит команда, он ее исполняет, после чего отправляет сообщение об окончании исполнения и переходит в режим ожидания. При выполнении команды измерения расстояния, микроконтроллер также отправляет расстояние до препятствия перед ним. Также в процессе работы микроконтроллер периодически проверяет степень заряда аккумулятора. Если он разряжен, то включается индикация и запрещается выполнение команд.

Для регулировки скорости вращения моторами используются ШИМ регуляторы на основе двух таймеров микроконтроллера. Третий таймер (16 битный) используется в роли часов реального времени и помогает измерять длительность импульсов в сигнале или длительность задержек.

В процессе перемещения при помощи датчика обратной связи от колес колесный робот измеряет скорости вращения колес и подстраивает их таким образом, чтобы обеспечить прямолинейность езды. Также по информации с датчиков исполнитель узнает какое расстояние он преодолел.

Для измерения расстояния до препятствия используется ультразвуковой датчик расстояния HC-SR04. Вначале микроконтроллер посылает импульс длительностью 10 микросекунд на вывод Trig датчика, запуская измерение. Дальномер измеряет расстояние до препятствия, испуская ультразвуковой

импульс и измеряя время, за которое импульс вернется. Далее датчик отправляет с выхода Echo импульс длиной равной времени прохождения ультразвукового импульса. Микроконтроллер при помощи прерывания по фронту и таймера 1 измеряет время между передним и задним фронтом импульса, измеряя его длину. Зная скорость звука в воздухе и время прохождения импульса можно вычислить расстояние до препятствия, от которого ультразвуковой импульс отразился.

Для облегчения управления исполнителем с компьютера используется составленная библиотека на языке Си. Основные задачи библиотеки:

1. Организация связи компьютера и исполнителя с помощью Bluetooth;
2. Передача команд, соответствующих написанному обучающимся коду, на исполнителя;
3. Прием сигнала подтверждения окончания выполнения команды и значений расстояния, измеренных при помощи ультразвукового дальномера

В библиотеке имеются пять команд, дублирующих действия, которые исполнитель может выполнять. Ход выполнения команд отображается в консоли. Так же в консоли отображаются сообщения об ошибке выполнения с указанием ее причины.

Связь компьютера происходит по протоколу Bluetooth. Робот-исполнитель в своем составе имеет bluetooth-модуль HC-05, при помощи которого исполнитель принимает и отправляет сигналы. Компьютер организует связь при помощи встроенного или внешнего bluetooth-модуля. Для этого bluetooth устройство инициализируется как виртуальный com-порт.

Для работы с com-портом на языке Си используется стандартная библиотека Windows API. Windows API – это набор базовых функций интерфейсов программирования приложений операционных систем семейства Microsoft Windows.

В начале выполнения команды процедура отправляет номер команды, которую должен выполнить исполнитель, а в консоли отображается

сообщение о начале выполнения команды. Далее в цикле `while` ожидается, когда будет получено сообщение об окончании выполнения команды. В это время в консоли отображается сообщение о том, что идет выполнение команды. По окончании выполнения команды исполнитель отправляет сообщение о заверении выполнения и, если нужно, передает дополнительные значения (расстояние до препятствия при выполнении команды измерения расстояния). Тогда процедура обрабатывает полученное сообщение и возвращает необходимое значение: 0 или -1 для команд движения и расстояние до препятствия для команды измерения расстояния.

Созданная библиотека будет использована для обучения студентов начальных курсов азам программирования на языке Си. В связи с этим необходимо создать ряд простых в использовании команд, которые будут изолировать обучающегося от сложностей, связанных со связью исполнителя и компьютера. Используя ряд простых команд для управления исполнителем для решения ряда обучающих задач, студент будет осваивать основные команды языка, методы работы с языком Си, изучать простые алгоритмы.

Для того чтобы упростить работу с исполнителем объявление переменных и используемые процедуры были инкапсулированы при помощи директивы `#define`.

В конечном итоге, обучающемуся предлагается использовать следующие команды:

- 1) `beginWorkingWithRobot`
- 2) `endWorkingWithRobot`
- 3) `robotGoForward`
- 4) `robotGoBack`
- 5) `robotTurnLeft`
- 6) `robotTurnRight`
- 7) `robotGetLength`

Первые две команды используются для начала и завершения работы с исполнителем. Остальные пять команд соответствуют командам, которые выполняет исполнитель.

Составленный аппаратно-программный комплекс был протестирован. Для этого был составлен программный код, выполняющий тестовую задачу, а именно подъезд к препятствию и его объезд. В цикле while исполнитель проверяет расстояние до препятствия. Если расстояние позволяет ему сделать шаг вперед, то есть расстояние больше 10 сантиметров, то исполнитель делает шаг. Если же расстояние меньше 10 сантиметров, то робот выполняет маневр объезда. Колесный робот справился с поставленной задачей.

Заключение. В ходе дипломной работы был создан комплекс аппаратно-программных средств для поддержки серии практикумов для получения необходимых в сфере биоинженерии квалификаций. Для этого был спроектирован и реализован колесный робот – исполнитель на основе колесного шасси, а так же создана библиотека на языке Си для обучения азам языка программирования.

Для разработки практикума был проведен обзор существующих практикумов и выявлены их достоинства и недостатки.

В ходе проектирования исполнителя была создана колесная платформа на основе коллекторных моторов постоянного тока, а также была разработана и реализована плата управления этой платформой на основе микроконтроллера. Для выбора компоновки колесной платформы был проведен обзор существующих платформ.

Работоспособность созданной системы была проверена. Для этого был написан пример кода для решения поставленной задачи – объезда препятствия. В ходе тестирования комплекс аппаратно-программных средств полностью выполнил поставленную для него задачу.