

Филяков А.А. Аппаратно-программная робототехническая система, Выпускная квалификационная работа бакалавра по направлению: «Информатика и вычислительная техника», Нижегородский государственный технический университет им. Р.Е. Алексеева, кафедра: «Вычислительные системы и технологии», Нижний Новгород, 2015.
Руководитель: доцент кафедры «Вычислительные системы и технологии» Гай В.Е.

Работа посвящена проектированию и реализации аппаратно-программной робототехнической системы, решающей задачу перемещения робототехнического устройства в пространстве с помощью дистанционного управления. Описывается структура системы, методы и способы реализации модулей, входящих в систему. Подробно рассматривается реализация системы на аппаратном уровне, а также процесс создания программного обеспечения для робототехнического устройства.

В результате тестирования установлено, что разработанная система корректно решает поставленную задачу дистанционного управления перемещающимся в пространстве робототехническим устройством.

Объем работы 55 страниц. Использовано источников – 8, рисунков – 11, таблиц – 1.

Перв. примен.	<h2>Содержание</h2> <p>В случае, если использование операционной системы понадобится на одном или нескольких модулях робототехнического устройства (например, на контроллере управления), для данного модуля может быть выбрана и использована другая операционная система в зависимости от сложившихся условий и требований..... 10</p> <p>Рисунок 1 – Структурная схема робототехнической системы..... 12</p> <p>С учетом использования в качестве устройства управления полноценного одноплатного компьютера Raspberry Pi появляется еще один вариант реализации пользовательского интерфейса – реализация его в виде программы на самом устройстве управления и последующий запуск и работа с этой программы с помощью удаленного доступа к терминалу Raspberry Pi с помощью протокола SSH. Именно этот вариант является наиболее предпочтительным по той причине, что в данном случае отсутствует необходимость организации дополнительного интерфейса обмена данными между персональным компьютером и устройством управления, так как эта задача возлагается на уже существующие и проверенные средства. К тому же при таком варианте реализации будут максимально задействованы встроенные возможности Raspberry Pi. 15</p> <p>Данная структурная схема была получена путем переработки схемы, представленной на рисунке 1. 16</p>									
	Справ. №									
Подпись и дата										
Изм. № дубл.										
Взам. инв. №										
Подпись и дата										
Инв. № подл.	Изм.	Лист	№ докум.	Подпись	Дата	<p align="center">ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</p> <p align="center">Аппаратно-программная робототехническая система</p>				
	Разраб.	Филяков А.А.								
	Провер.	Гай В.Е.								
	Реценз.									
	Н. Контр.									
	Утверд.									
Лит.	Лист	Листов								
	3	55								
						НГТУ им. Р.Е. Алексеева				

Перв. примен.		<h2 style="text-align: center;">Введение</h2> <p>В последнее время в связи с популяризацией и растущей доступностью информационных технологий свое развитие получила и область робототехники. В повседневной жизни, быту, промышленности, науке и технике встречается все больше современных устройств, которые в той или иной степени можно отнести к разряду робототехнических: начиная от банального робота-пылесоса и заканчивая роботами, способными автономно управлять автомобилями или сложнейшими станками на производстве.</p> <p>Одним из примеров робототехнических устройств является устройство телеприсутствия – устройство, позволяющее пользователю удаленно наблюдать за событиями в определенном месте посредством камеры и в определенной мере (в зависимости от типа устройства) влиять на эти события. Подобные роботы находят применение в таких областях, как исследования слабо доступных человеку местностей (горы, дно океана, поверхность других планет); ядерные технологии; медицина; военная техника и др.</p> <p>Разработка и создание роботов стали доступны практически всем желающим в домашних условиях. В интернете и специализированной прессе можно увидеть огромное количество материалов о создании различных устройств, имеющих возможность самостоятельно выполнять определенные функции, основываясь на данных об окружающей среде или удаленном управлении с компьютера. Развитие робототехнической отрасли среди энтузиастов обусловлено ещё и появлением в последнее время на рынке широкого выбора доступных элементов и модулей, необходимых для разработки робототехнических устройств.</p> <p>В связи со всем вышесказанным возникла идея создания устройства, способного перемещаться в пространстве под управлением команд, поступающих с компьютера, и выполнять определенную прикладную функцию, например, запись звука или видео.</p> <p>Таким образом, данная работа посвящена разработке и созданию робототехнического устройства, способного принимать управляющие команды с персонального компьютера и осуществлять на основе них перемещение в пространстве.</p> <p>В дальнейшем планируется доработка данного устройства, а именно, добавление возможности управления устройством посредством голосовых команд.</p>										
	Справ. №											
Подпись и дата		<div style="display: flex; justify-content: space-between; align-items: center;"> <div> <table border="1" style="width: 100%;"> <tr><td>Изм.</td><td>Лист</td><td>№ докум.</td><td>Подпись</td><td>Дата</td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </table> </div> <div style="text-align: center; flex-grow: 1;"> <h3>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</h3> </div> <div style="width: 100px; text-align: right;"> Лист 4 </div> </div>	Изм.	Лист	№ докум.	Подпись	Дата					
	Изм.		Лист	№ докум.	Подпись	Дата						
	Инд. № дубл.											
	Взам. инв. №											
Подпись и дата												
Инд. № подл.												

Перв. примен.							
Справ. №							
<p>1 Техническое задание</p> <p>1.1 Назначение разработки и область применения</p> <p>Разрабатываемое робототехническое устройство предназначено для перемещения в пространстве под управлением команд, передающихся с персонального компьютера. Под командами понимаются данные, закодированные в определенном установленном формате. Для работы системы должен существовать модуль управления, доступный с персонального компьютера, и модуль, располагающийся на робототехническом устройстве и управляющий непосредственно этим устройством.</p> <p>Области применения разрабатываемого устройства:</p> <ul style="list-style-type: none"> • Перемещение в пространстве посредством удаленного управления. • Запись видео- и звуковых сигналов на расстоянии посредством удаленного управления. <p>Разрабатываемое устройство должно иметь возможность последующей доработки для выполнения прикладной функции (подключение устройства записи видео/звука).</p> <p>1.2 Технические требования</p> <p>Разрабатываемая система должна выполнять следующий функционал:</p> <p>На персональном компьютере:</p> <ul style="list-style-type: none"> • Обеспечение возможности ввода управляющего сигнала пользователем (интерфейс пользователя). • Передача управляющих сигналов на робототехническое устройство по беспроводному интерфейсу (Wi-Fi, 3G, Bluetooth и т.п.). <p>В составе робототехнического устройства:</p> <ul style="list-style-type: none"> • Прием управляющих сигналов с персонального компьютера по беспроводному интерфейсу. • Обработка устройством принятых сигналов. • Осуществление перемещения в пространстве на основе принятых управляющих сигналов с помощью стандартных средств передвижения (колеса, гусеницы и т.п.). • Возможность подключения устройства для записи звука/видео. <p>Требования, предъявляемые к персональному компьютеру для работы с разрабатываемой системой:</p> <ul style="list-style-type: none"> • ОС семейства Windows. 							
Подпись и дата	Изм.	Лист	№ докум.	Подпись	Дата	<div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div>	Лист
							5

Инов. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	Справ. №	Перв. примен.	<ul style="list-style-type: none"> • Требования к аппаратной части персонального компьютера определяются операционной системой. • Устройства ввода для взаимодействия с пользователем (клавиатура, мышь). • Беспроводной интерфейс передачи данных (Wi-Fi, 3G, Bluetooth и т.п.). 				
Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ					Лист	
										6	

Перв. примен.	<h2>2 Анализ технического задания</h2> <h3>2.1 Выбор варианта реализации робототехнического устройства</h3> <p>В классическом варианте своей реализации робототехническая система, управляемая человеком непосредственно и взаимодействующая с окружающей средой, включает в себя следующие подсистемы [5]:</p> <ul style="list-style-type: none"> • Управляющая подсистема. • Исполнительная подсистема. • Подсистема связи. <p>Более подробно о структуре робототехнической системы в целом, функционале каждой подсистемы и вариантах реализации различных ее модулей будет рассказано в пункте 3 «Разработка системы на структурном уровне». Здесь же будут кратко описаны концепции реализации устройства и сделан выбор в пользу одной из них.</p> <p>Можно выделить две основные концепции реализации робототехнической системы:</p> <ul style="list-style-type: none"> • Использование набора для робототехники (например, Lego Mindstorms). • Использование отдельных независимых устройств для реализации каждого модуля и интеграция их в единую систему путем использования различных интерфейсов связи. <p>Рассмотрим обе эти концепции подробнее.</p> <p>Lego Mindstorms – набор сопрягаемых деталей и электронных блоков, предназначенный для создания программируемого робота. Данный набор состоит из деталей, используемых для построения физического каркаса робота (балки, оси, колеса, шестерни, двигатели), и деталей, используемых для программирования робота (сенсоры, датчики, программируемые блоки). Преимуществом использования данного варианта реализации робототехнического устройство является наличие полного комплекса необходимых деталей и модулей, необходимый для построения робота, а также наличие устойчивых и проверенных связей между ними. Основным недостатком этой концепции является высокая стоимость данного набора на рынке.</p> <p>Другая концепция реализации робототехнического устройства предполагает использования в качестве каждого из модулей независимых устройств (каркас, колесная база, двигатели, управляющий контроллер и т.д.) и объединение их в единую систему. Преимуществом данного варианта является сравнительно низкая рыночная стоимость деталей, использование минимального набора необходимых деталей. Недостатком</p>																	
	Справ. №																	
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.													
<table border="1"> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> <td rowspan="2"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ </td> <td>Лист</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td>7</td> </tr> </table>						Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист						7
Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист												
						7												

Перв. примен.						
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
						8

Перв. примен.		<h2 style="text-align: center;">2.2 Выбор ОС для персонального компьютера</h2> <p>В настоящее время имеется широкий выбор операционных систем для персональных компьютеров. Остановим свой выбор на двух семействах операционных систем: Windows и Linux:</p> <ul style="list-style-type: none"> Windows – одно из самых распространенных семейств операционных систем. Данное семейство было разработано компанией Microsoft и получило широкое применение как в коммерческом, так и в частном использовании. Имеется множество версий данной операционной системы, последними из них являются Windows 7, Windows 8 и Windows 10. Linux – совокупность Unix-подобных операционных систем, основанных на одноименном ядре. Каждая версия Linux представляет собой готовый дистрибутив с определенным набором программ, собранных в зависимости от потребностей пользователя. Несмотря на свободное распространение и открытый исходный код, Linux имеет меньшую популярность, чем Windows. <p>Проведем сравнение этих операционных систем по некоторым критериям:</p> <ul style="list-style-type: none"> Информационная безопасность. В силу более широкого распространения ОС Windows для нее имеется значительно большее количество вредоносных программ. Поэтому Windows является менее безопасной операционной системой. Стабильность. Несмотря на то, что на операционной системе Linux также случаются сбои, она является гораздо более стабильной, чем Windows. Совместимость с оборудованием. Операционные системы Windows и Linux имеют широчайший выбор поддерживаемого оборудования. Хотя многие производители оборудования обеспечивают свою продукцию драйверами только для операционной системы Windows. Программное обеспечение. Вследствие своей наибольшей популярности операционная система Windows имеет наибольший выбор программного обеспечения. Многие популярные программы имеют версии только для этой операционной системы. Возможность разработки программного обеспечения. Операционная система Windows имеет такие языки программирования и средства разработки (например, C#), которые недоступны для других операционных систем. Среди распространенных языков программирования нет таких, которые не работали бы на операционной системе Windows. 					
	Справ. №						
Инов. № подл.	Подпись и дата					<div style="text-align: center; font-weight: bold; font-size: 1.2em;">ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div>	Лист
	Взам. инв. №						<div style="text-align: center; font-weight: bold; font-size: 1.2em;">9</div>
	Инов. № дубл.						
	Подпись и дата						
	Изм.	Лист	№ докум.	Подпись	Дата		

Перв. примен.						
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		10

Исходя из всего вышесказанного для использования на персональном компьютере при разработке и управлении робототехнической системой была выбрана операционная система Windows, так как она соответствует всем предъявляемым требованиям.

В случае, если использование операционной системы понадобится на одном или нескольких модулях робототехнического устройства (например, на контроллере управления), для данного модуля может быть выбрана и использована другая операционная система в зависимости от сложившихся условий и требований.

Перв. примен.						<h3>3 Разработка системы на структурном уровне</h3> <h4>3.1 Структура робототехнического устройства</h4> <p>Согласно выбранной концепции реализации робототехнической системы, она должна включать в себя следующие подсистемы (модули): управляющую, систему связи, исполнительную (моторную).</p> <p>Управляющая подсистема – подсистема, которая на основании поступающих в нее команд и заложенной заранее программы формирует поведение робота. В данном случае управляющая подсистема должна представлять из себя контроллер, способный принимать сигналы с персонального компьютера по беспроводному интерфейсу и преобразовывать их в сигналы управления моторной подсистемой. Управляющая подсистема должна находиться непосредственно на робототехническом устройстве.</p> <p>Исполнительная подсистема – подсистема, которая непосредственно выполняет команды, переданные оператором. В данном случае она будет состоять из каркаса, моторов и колесной части.</p> <p>Подсистема связи – подсистема, которая отвечает за передачу данных между всеми подсистемами робота и внешним миром. В данном случае она будет включать в себя каналы передачи данных различного типа (проводные и беспроводные). Также данная подсистема должна обеспечивать пользователю возможность ввода управляющих команд посредством пользовательского интерфейса на персональном компьютере.</p> <p>Структурная схема робототехнической системы представлена на рисунке 1:</p>
	Справ. №					
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.	<div> <div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div> <div> <div>Изм.</div> <div>Лист</div> <div>№ докум.</div> <div>Подпись</div> <div>Дата</div> </div> </div>
	Лист					

Перв. примен.						
Справ. №						
Инв. № подл.						
Подпись и дата						
Взам. инв. №						
Инв. № дубл.						
Подпись и дата						
<div><div>Пользовательский интерфейс</div><div>Исполнительная подсистема</div><div>Устройство управления</div></div> <p>Рисунок 1 – Структурная схема робототехнической системы</p>						
Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	
						Лист

Перв. примен.		<h3>3.2 Выбор вариантов реализации подсистем</h3> <h4>3.2.1 Исполнительная подсистема</h4> <p>Исполнительная система данного робототехнического устройства должна состоять из стандартных компонентов: каркаса, колесной части и двигателей, приводящих колесную часть в движение.</p> <p>Для того чтобы избежать трудностей, связанных с разработкой и построением каркаса устройства; интеграцией каркаса и ходовой части в единую систему, удобно использовать в качестве заготовки каркас автомобиля на радиоуправлении. Выбор данного варианта имеет следующие преимущества:</p> <ul style="list-style-type: none"> Наличие всех необходимых частей исполнительной подсистемы. Наличие устойчивого интерфейса (передачи) между двигателями и колесной частью. <p>При выборе такого варианта реализации подсистемы необходимо будет решить следующие проблемы:</p> <ul style="list-style-type: none"> Создание интерфейса между двигателями и управляющей подсистемой (использование усилителей сигнала). Закрепление управляющей подсистемы (контроллера) на каркасе устройства. <h4>3.2.2 Устройство управления</h4> <p>В данном робототехническом устройстве роль устройства управления должен выполнять контроллер, способный реализовывать функции приема команды с персонального компьютера по беспроводному интерфейсу и передавать эту команду двигателю в понятном ему (двигателю) виде. Исходя из этого были сформулированы требования к контроллеру:</p> <ul style="list-style-type: none"> Возможность подключения адаптера для обмена данными по беспроводному интерфейсу. Наиболее подходящим интерфейсом является Wi-Fi из-за своей распространенности и более высокой дальности приема сигнала по сравнению с Bluetooth. Наличие порта ввода/вывода для выдачи управляющих сигналов для двигателей. Возможность удобного программирования данного контроллера для работы с портом ввода/вывода на языке высокого уровня. 						
	Справ. №							
Подпись и дата								
Инв. № дубл.								
Взам. инв. №								
Подпись и дата								
Инв. № подл.								
		Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
								13

Перв. примен.					
Справ. №					
Подпись и дата					
Инв. № дубл.					
Взам. инв. №					
Подпись и дата					
Инв. № подл.					
<p>В настоящее время на рынке имеется множество различных контроллеров, имеющих в своем арсенале данный функционал. Среди них можно выделить платы семейства Arduino и им подобные, а также одноплатный компьютер Raspberry Pi.</p> <p>Arduino - торговая марка аппаратно-программных средств для построения простых систем автоматики и робототехники, ориентированная на непрофессиональных пользователей. Программная часть состоит из бесплатной программной оболочки (IDE) для написания программ, их компиляции и программирования аппаратуры. Аппаратная часть представляет собой набор смонтированных печатных плат, продающихся как официальным производителем, так и сторонними производителями. Полностью открытая архитектура системы позволяет свободно копировать или дополнять линейку продукции Arduino.</p> <p>Raspberry Pi – одноплатное устройство, которое, несмотря на свои размеры, содержит в себе функционал полноценного компьютера. Raspberry Pi выпускается в нескольких комплектациях: модель «А», модель «В», модель «В+» и модель «2 В». Первые три версии оснащены ARM11 процессором Broadcom BCM2835 с тактовой частотой 700 МГц и модулем оперативной памяти на 256МБ/512МБ, размещенными по технологии «package-on-package» непосредственно на процессоре. Модель «2 В» оснащается процессором с 4 ядрами Cortex-A7 с частотой 1ГГц и оперативной памятью размером 1ГБ. Модель «А» оснащается одним USB 2.0 портом, модель «В» двумя, а модели «В+» и «2 В» — четырьмя. Также в моделях «В», «В+» и «2 В» присутствует порт Ethernet. Помимо основного ядра, BCM2835 включает в себя графическое ядро с поддержкой OpenGL ES 2.0, аппаратного ускорения и Full HD-видео и DSP-ядро. Одной из самых интересных особенностей Raspberry Pi является наличие портов GPIO (general purpose input/output). Благодаря этому "малиновый" компьютер можно использовать для управления различными устройствами. В модели «В» платы присутствуют 26 портов, а в модели «В+» и «2В» - 40 портов GPIO.</p> <p>Как и любой другой полноценный компьютер, Raspberry Pi предполагает использование операционной системы. Наиболее часто используемая система – система семейства Linux – Raspbian.</p> <p>Исходя из наличия встроенного USB-порта, необходимого для подключения Wi-Fi адаптера, портов GPIO и возможности удаленного доступа по сети к терминалу и файловой системе одноплатного компьютера, был выбран вариант с использованием Raspberry Pi в качестве контроллера для управления робототехническим устройством. Также немаловажной причиной данного выбора послужило наличие у автора работы в распоряжении устройства Raspberry Pi модели «В».</p>					
					Лист
ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ					14
Изм.	Лист	№ докум.	Подпись	Дата	

Перв. примен.					
Справ. №					
Подпись и дата					
Инв. № дубл.					
Взам. инв. №					
Подпись и дата					
Инв. № подл.					
<div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> </div>					
<div> <div>Изм.</div> <div>Лист</div> <div>№ докум.</div> <div>Подпись</div> <div>Дата</div> </div>					<div>Лист</div> <div>15</div>

Raspberry Pi будет соединяться с двигателями с помощью проводов через специальное устройство – усилитель. Данное устройство является необходимым для организации управления подачей питания на двигатели выходными сигналами с портов GPIO Raspberry Pi, так как напряжение на выходах Raspberry Pi ниже, чем напряжение питания двигателей. Подробно о разработке и создании усилителей будет рассказано далее.

3.2.3 Подсистема связи

В данной робототехнической системе подсистема связи должна включать в себя все интерфейсы, используемые для соединения других подсистем, а также обеспечивать пользователю возможность ввода управляющих команд и передавать их на устройство управления.

Интерфейсы, соединяющие различные модули робототехнической системы можно разделить на две группы:

- Проводные.
- Беспроводные (Wi-Fi).

Интерфейс пользователя может быть реализован на персональном компьютере в консольном или графическом режиме.

С учетом использования в качестве устройства управления полноценного одноплатного компьютера Raspberry Pi появляется еще один вариант реализации пользовательского интерфейса – реализация его в виде программы на самом устройстве управления и последующий запуск и работа с этой программы с помощью удаленного доступа к терминалу Raspberry Pi с помощью протокола SSH. Именно этот вариант является наиболее предпочтительным по той причине, что в данном случае отсутствует необходимость организации дополнительного интерфейса обмена данными между персональным компьютером и устройством управления, так как эта задача возлагается на уже существующие и проверенные средства. К тому же при таком варианте реализации будут максимально задействованы встроенные возможности Raspberry Pi.

Перв. примен.																						
Справ. №																						
<div>3.3 Структурная схема устройства</div> <div>В связи с приведенным выше описанием подсистем робототехнической системы и интерфейсов между ними и внешним миром структурная схема устройства будет иметь следующий вид:</div> <div><div><div>ПК + специальное ПО</div><div>↓ Wi-Fi, SSH</div><div>Контроллер (Raspberry Pi)</div></div><div><div>Двигатели + каркас + колесная часть</div><div>↑ Проводное соединение, плата усилителя</div><div>Контроллер (Raspberry Pi)</div></div></div> <div>Рисунок 2 – Структурная схема устройства.</div> <div>Данная структурная схема была получена путем переработки схемы, представленной на рисунке 1.</div>																						
Подпись и дата																						
Инв. № дубл.																						
Взам. инв. №																						
Подпись и дата																						
Инв. № подл.																						
<table><tr><td></td><td></td><td></td><td></td><td></td><td rowspan="3">ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</td><td>Лист</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td>16</td></tr><tr><td>Изм.</td><td>Лист</td><td>№ докум.</td><td>Подпись</td><td>Дата</td></tr></table>										ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист						16	Изм.	Лист	№ докум.	Подпись	Дата
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист																
						16																
Изм.	Лист	№ докум.	Подпись	Дата																		

Перв. примен.	<h2>4 Разработка системы на аппаратном уровне</h2> <h3>4.1 Конфигурирование Raspberry Pi</h3> <p>Конфигурирование одноплатного компьютера Raspberry Pi– важнейший этап, необходимый для обеспечения возможности работы данного устройства в качестве контроллера управления робототехнической системой. Конфигурирование Raspberry Pi можно разделить на два основных этапа:</p> <ul style="list-style-type: none"> Первоначальная настройка. Данный этап включает в себя первый запуск одноплатного компьютера, установку операционной системы, задание первичных настроек устройства. Настройка Wi-Fi адаптера. Данный этап включает в себя загрузку и установку драйвера для используемого адаптера, а также настройку беспроводного сетевого интерфейса. <h4>4.1.1 Первоначальная настройка Raspberry Pi</h4> <p>Данный этап конфигурации одноплатного компьютера необходимо выполнять всегда во время первого запуска устройства вне зависимости от целей его дальнейшего использования.</p> <p>Оборудование, используемое для выполнения первоначальной настройки Raspberry Pi:</p> <ul style="list-style-type: none"> Одноплатный компьютер Raspberry Pi модели «В». Клавиатура с USB-разъемом. Мышь с USB-разъемом. SD-карта памяти объемом 8 Гб. Адаптер питания с разъемом Micro-USB. Монитор с HDMI-разъемом. HDMI-кабель. Кардридер для чтения SD-карт памяти. <p>Программное обеспечение, используемое для выполнения первоначальной настройки Raspberry Pi:</p> <ul style="list-style-type: none"> Образ операционной системы Raspbian. Программа Win32DiskImager для записи образа операционной системы на карту памяти. 								
	Справ. №								
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.				
	<div style="display: flex; justify-content: space-between; align-items: center;"> <table border="1" style="width: 100%;"> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> </tr> </table> <div style="text-align: center; flex-grow: 1;"> <h1>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</h1> </div> <div style="width: 100px; text-align: center;"> Лист 17 </div> </div>					Изм.	Лист	№ докум.	Подпись
Изм.	Лист	№ докум.	Подпись	Дата					

Перв. примен.								
Справ. №								
Подпись и дата								
Инв. № дубл.								
Взам. инв. №								
Подпись и дата								
Инв. № подл.								
		Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
								18

После того, как все необходимое для первого запуска Raspberry Pi оборудование и программное обеспечение было собрано, первоначальная настройка устройства состояла из следующих шагов:

1. Запись образа операционной системы Raspbian на SD карту памяти с помощью программы Win32DiskImager.

Raspbian – операционная система, разработанная специально для одноплатных компьютеров Raspberry Pi. Она представляет собой особую версию операционной системы Debian, относящейся к семейству операционных систем Linux. Raspbian учитывает аппаратное обеспечение и другие особенности одноплатных компьютеров Raspberry Pi, поэтому является одной из наиболее популярных и рекомендуемых операционных систем для этих устройств. Важной особенностью и преимуществом этой операционной системы является возможность полноценной работы в ней как в режиме командной строки, так и с использованием графического интерфейса.

Win32DiskImager – свободно распространяемый программный продукт, предоставляющий возможности записи данных на физические носители, в том числе и создание носителей с образами операционных систем. Данная программа работает в операционной системе Windows.

В результате выполнения данного шага была получена SD карта памяти с записанным на нее образом операционной системы Raspbian.

2. Подключение Raspberry Pi.

Данный шаг включает в себя объединение всего описанного выше оборудования в единую компьютерную систему, а именно:

- Подключение SD карты памяти к соответствующему разъему Raspberry Pi.
- Подключение USB-клавиатуры и мыши к USB-разъемам Raspberry Pi.
- Подключение монитора к Raspberry Pi посредством HDMI-кабеля.
- Подключение MicroUSB адаптера питания к соответствующему разъему Raspberry Pi.

В результате выполнения данного шага к одноплатному компьютеру были присоединены все необходимые для работы периферийные устройства. После выполнения указанных выше действий адаптер питания был подключен к сети и началась первая загрузка операционной системы.

3. Настройка Raspberry Pi.

В процессе первой загрузки операционной системы одноплатного компьютера на экран монитора было выведено меню Raspi-config, которое позволяет задать все необходимые начальные настройки системы:

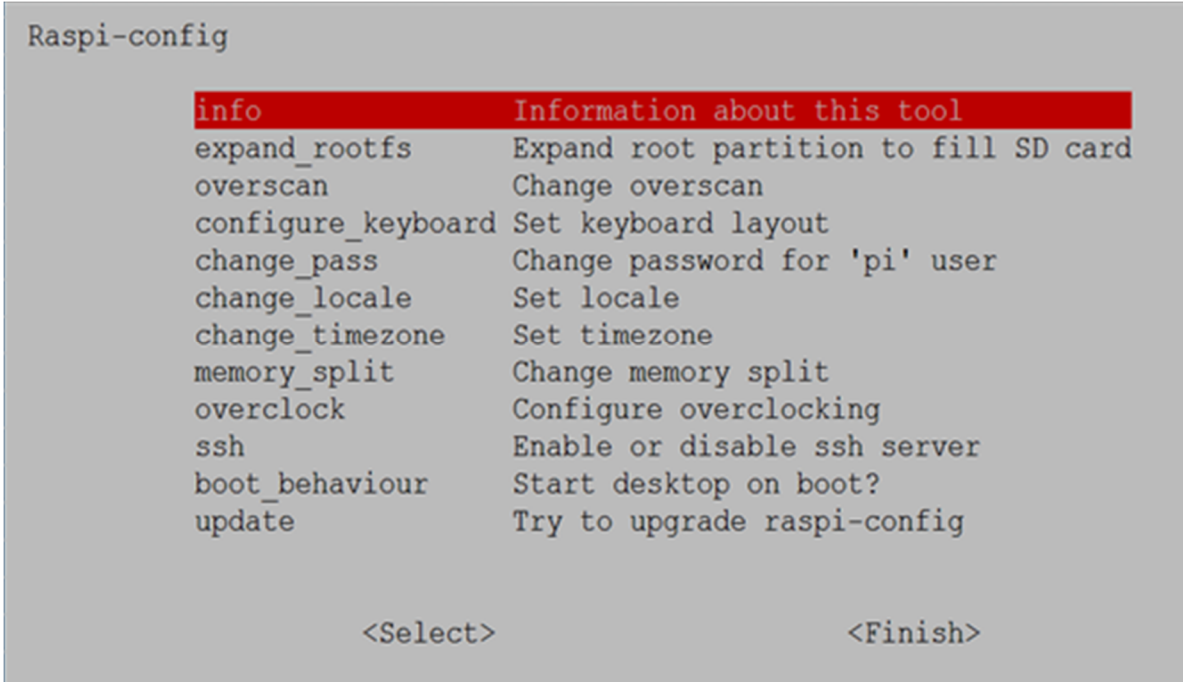


Рисунок 3 – Меню Raspi-config

Данное меню позволяет задать следующие настройки одноплатного компьютера:

- expand_rootfs – возможность увеличения размера раздела root на весь раздел карты памяти. Данная возможность была включена для увеличения ресурса памяти компьютера.
- change_pass – смена пароля для встроенного пользователя pi. Пароль был изменен по причине неудобства использования стандартного пароля.
- change_locale – смена языка системы. К списку языков была добавлена русская локализация с кодировкой UTF-8.
- change_timezone – смена часового пояса. в Raspberry Pi отсутствуют встроенные часы, поэтому для правильной загрузки системного времени из сети Интернет был выбран нужный часовой пояс.
- memory_split – выделение оперативной памяти для работы компьютера. Чтобы обеспечить возможность комфортной работы в графической оболочке было выбрано максимально доступное значение количество памяти – 256 Мб.
- ssh – возможность включения ssh-сервера на компьютере. Данная возможность была включена, так как использование ssh-сервера будет

Перв. примен.					
Справ. №					
	<p>необходимо в дальнейшем для удаленного доступа к одноплатному компьютеру.</p> <ul style="list-style-type: none"> • <code>boot_behaviour</code> – возможность включения загрузки графической оболочки при загрузке системы. Данная возможность была включена, так как проведение некоторых дальнейших настроек удобнее выполнять с помощью графической оболочки. <p>В результате выполнения данного шага были выставлены все необходимые для работы Raspberry Pi начальные настройки.</p> <p>После выполнения всех шагов первоначальной настройки Raspberry Pi на данном устройстве была запущена операционная система Raspbian с графической оболочкой. После этого одноплатный компьютер оказался полностью готов к дальнейшей настройке как для выполнения функционала, возлагаемого для него в робототехнической системе, так и для использования в качестве персонального компьютера.</p>				
	<h4>4.1.2 Подключение и настройка Wi-Fi адаптера для Raspberry Pi</h4>				
	<p>В связи тем, что перемещение робототехнической системы в пространстве должно быть как можно менее ограниченным, возникает необходимость передачи данных между персональным компьютером и устройством управления робототехнической системы (Raspberry Pi) по беспроводному каналу. В силу своей распространенности и доступности в качестве этого канала был выбран Wi-Fi интерфейс. Методы и средства для работы по данному интерфейсу заложены практически в каждую современную операционную систему, в том числе и в операционные системы, использующиеся в данной работе (Windows и Raspbian).</p>				
	<p>Однако Raspberry Pi в своей базовой комплектации не имеет встроенного Wi-Fi адаптера или иного устройства, способного передавать и принимать Wi-Fi сигнал. Для организации работы одноплатного компьютера по Wi-Fi интерфейсу использовался Wi-Fi адаптер с USB-разъемом – TP-LINK TL-WN723N v3. Данное устройство обладает следующими характеристиками:</p>				
	<ul style="list-style-type: none"> • Интерфейс: USB 2.0. • Размеры: 36.8×17×8.4. • Диапазон частот: 2400 – 2483,5 МГц. • Стандарт беспроводных сетей: IEEE 802.11n, IEEE 802.11g, IEEE 802.11b. • Скорость передачи данных: до 150 Мбит/с. 				
Инв. № подл.					Лист 20
Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата		
<div> <div>Изм.</div> <div>Лист</div> <div>№ докум.</div> <div>Подпись</div> <div>Дата</div> </div> <div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div>					

Перв. примен.	<p>По своим характеристикам данный Wi-Fi адаптер является вполне подходящим для реализации беспроводного обмена данными между Raspberry Pi и персональным компьютером.</p> <p>После физического подключения данного Wi-Fi адаптера к соответствующему порту Raspberry Pi в терминале одноплатного компьютера была выполнена команда:</p> <ul style="list-style-type: none"> • lsusb <p>Данная команда выводит на экран все USB-устройства, подключенные к системе. В данном случае информация об используемом Wi-Fi адаптере выведена на экран не была. Одной из возможных причин отсутствия данной информации могло являться отсутствие в операционной системе драйвера для работы используемого устройства.</p> <p>Для того, чтобы операционная система компьютера могла работать с тем или иным подключаемым устройством, необходимо наличие в операционной системе специального системного программного продукта – драйвера устройства. Чаще всего драйверы выпускаются производителями устройств и поставляются совместно с ними. Кроме того, современные операционные системы имеют в своем составе большое количество встроенных драйверов для определенного перечня распространенных устройств. Именно поэтому зачастую для полноценной работы устройства достаточно лишь подключить его к компьютеру – поиск и установку необходимого драйвера операционная система произведет сама.</p> <p>Однако нередки случаи, когда поиск и установку драйверов для определенных устройств пользователю приходится выполнять вручную. Именно такой случай и имел место при настройке данного Wi-Fi адаптера. Более того, в поставку устройства был включен драйвер только для работы в операционной системе Windows, ситуация с драйверами, доступными на официальном сайте производителя, оказалась аналогичной.</p> <p>Основная часть Wi-Fi адаптера – чипсет – микросхема, позволяющая осуществлять взаимодействие по беспроводному интерфейсу. При наличии в операционной системе драйвера чипсета адаптер сможет выполнять свою основную функцию – взаимодействие по Wi-Fi. Используемый в данной работе адаптер, TP-LINK TL-WN723N v3, работает на основе чипсета RTL8188EUS производства фирмы Realtek. Производитель данного устройства свободно распространяет драйвер для многих операционных систем, в том числе и для систем семейства Linux, к которому относится Raspbian.</p> <p>Версия драйвера, необходимая для работы адаптера, напрямую зависит от версии ядра используемой операционной системы. Для того чтобы узнать точную версию ядра была выполнена команда: <code>uname -a</code>.</p>																							
	Справ. №																							
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.																			
<table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td rowspan="3"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ </td> <td>Лист</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td>21</td> </tr> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> <td></td> </tr> </table>											ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист						21	Изм.	Лист	№ докум.	Подпись	Дата	
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист																		
						21																		
Изм.	Лист	№ докум.	Подпись	Дата																				

Перв. примен.		<p>Таким образом была выяснена версия ядра операционной системы Raspbian: 3.18.11+ #781. Соответствующая данному ядру версия драйвера чипсета – 8188eu-20150406. Данная версия и была загружена с официального сайта производителя.</p> <p>После загрузки и распаковки архива с драйвером в терминале операционной системы был запущен скрипт <code>install.sh</code>, входящий в комплект с исходными кодами драйвера. Основные команды данного скрипта:</p> <ul style="list-style-type: none"> • <code>sudo cp 8188eu.conf /etc/modprobe.d/</code>. – копирование файла конфигурации драйвера в соответствующую директорию. • <code>sudo install -p -m 644 8188eu.ko /lib/modules/3.18.11+/kernel/drivers/net/wireless</code> – установка модуля драйвера в соответствующую директорию. • <code>sudodepmod 3.18.11+</code> - обновление файла зависимостей модулей. <p>В результате выполнения описанных выше действий драйвер чипсета Realtek RTL8188EUS был установлен. После перезагрузки одноплатного компьютера была снова выполнена команда:</p> <ul style="list-style-type: none"> • <code>lsusb</code> <p>В ее выводе появилась новая строка, соответствующая подключенному Wi-Fi адаптеру. Это говорит о том, что установка драйвера прошла успешно, и операционная система распознала устройство и готова к работе с ним.</p> <p>Но установки драйвера Wi-Fi адаптера недостаточно для работы компьютера по беспроводным сетям. Необходимо еще установить сетевые настройки системы для работы в составе определенной сети.</p> <p>Для обзора имеющихся в операционной системе сетевых интерфейсов была выполнена команда:</p> <ul style="list-style-type: none"> • <code>ifconfig</code> <p>Данная команда выводит на экран все имеющиеся сетевые интерфейсы. Среди данных интерфейсов был обнаружен и интерфейс <code>wlan0</code>, который соответствует беспроводному соединению. Однако у данного интерфейса отсутствовали сетевые настройки (IP адрес интерфейса, маска подсети и др), т.е. данный интерфейс не был сконфигурирован. Аналогичный результат выдала команда <code>iwconfig</code>, которая выводит на экран параметры беспроводных сетевых интерфейсов.</p> <p>Для проверки корректности работы Wi-Fi адаптера была выполнена команда:</p> <ul style="list-style-type: none"> • <code>sudo iwlist wlan0 scan</code> <p>Данная команда выводит список беспроводных сетей, которые удалось обнаружить указанному интерфейсу, и их параметры. Среди нескольких найденных сетей в выводе</p>				
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум.	Подпись	Дата	<div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div> <div>Лист 22</div>	

Перв. примен.		<p>команды была обнаружена сеть, на работу в которой планировалось настроить интерфейс. Это еще раз подтвердило правильность работы адаптера после установки драйвера.</p> <p>Параметры всех сетевых интерфейсов, как проводных, так и беспроводных в операционной системе Linux хранятся в конфигурационном файле /etc/network/interfaces. Для конфигурирования беспроводного интерфейса wlan0 в данный файл были добавлены следующие строки:</p> <ul style="list-style-type: none"> • iface wlan0 inet dhcp • wpa-ssid "Filyakov" • wpa-psk "Password" <p>Первая строка указывает на работу интерфейса по протоколу DHCP, т.е. динамическое выделение IP адреса. Названия параметров во второй и третьей строках (wpa-ssid, wpa-psk) обусловлено названием используемого в данной сети механизма шифрования – WPA.</p> <p>После сохранения изменений в файле и перезагрузки одноплатного компьютера была снова выполнена команда:</p> <ul style="list-style-type: none"> • ifconfig <p>Вывод данной команды показал, что интерфейсу wlan0 присвоен IP адрес в локальной сети: 192.168.0.101. Также были заданы и другие параметры (маска подсети, адрес широковещательной рассылки). После этого одноплатный компьютер Raspberry Pi стал доступен по указанному адресу, что было проверено командой ping. Аналогично другие компьютеры данной сети стали доступны с Raspberry Pi.</p> <p>В ходе настройки беспроводного интерфейса на одноплатном компьютере возникла проблема, заключающаяся в том, что несмотря на задание настроек в файле /etc/network/interfaces, интерфейс wlan0 оказывался не сконфигурированным. После изучения данного вопроса была выявлена причина неполадок – недостаточное питание одноплатного компьютера. Используемый адаптер питания имел силу тока на выходе 1 А. Согласно документации к Raspberry Pi, такое питание вполне достаточно для работы самого компьютера, но вот для полноценной работы некоторых периферийных устройств, в том числе и Wi-Fi адаптеров, необходимо более мощное питание. После подключения Raspberry Pi с помощью адаптера питания с силой тока на выходе 2 А данная проблема была полностью устранена и компьютер оказался подключенным к сети по беспроводному интерфейсу.</p> <p>В результате выполнения описанных выше действий была настроена работа одноплатного компьютера Raspberry Pi по беспроводному сетевому интерфейсу с</p>				
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум.	Подпись	Дата	<p>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</p>	
					Лист	
					23	

помощью Wi-Fi адаптера TP-LINK TL-WN723N v3. Фото данного устройства в рабочем состоянии представлено ниже:



Рисунок 4 – Raspberry Pi в рабочем состоянии

[illegible]

Перв. примен.		<h2 style="text-align: center;">4.2 Разработка платы усилителя</h2> <p>Согласно документации к одноплатному контроллеру Raspberry Pi, его порты ввода/вывода (GPIO) имеют выходное напряжение до 3,3 В. Двигатели, используемые в данной робототехнической системе, имеют напряжение питания 6 В. Отсюда возникает проблема совместимости выходного напряжения контроллера и входного напряжения двигателей. В случае прямого проводного соединения выходов GPIO Raspberry Pi и входов двигателей имеется высокая вероятность выхода из строя используемых портов. Для решения данной проблемы в данной робототехнической работе используется отдельное устройство – усилитель.</p> <p>Усилитель – электротехническое устройство, позволяющее с помощью меньшего напряжения на выходах контроллера (3,3 В) управлять подачей большего напряжения питания (6 В) на управляемые контроллером устройства (двигатели). Усилитель, используемый в данной работе, основывается на методе токового ключа. Данный метод заключается в том, что более слабый сигнал, идущий с контроллера, не поступает напрямую на вход двигателя, а является своеобразным ключом, позволяющим открывать и закрывать подачу питания на двигатель. Такое использование выходных сигналов портов ввода/вывода позволяет защитить порты от появления неисправностей.</p> <p>Требования к плате усилителя:</p> <ul style="list-style-type: none"> • Напряжение питания: 6 В. • Использование в составе робототехнического устройства, перемещающегося в пространстве. • Соответствие электрической принципиальной схеме устройства. • Количество усилителей на плате: 2. <p>В данном робототехническом устройстве используются усилители, изготовленные по уже существующей разработке. Электрическая принципиальная схема одного усилителя приведена на рисунке 5:</p>															
	Справ. №																
Инов. № подл.	Подпись и дата	Инов. № дубл.	Взам. инв. №	Подпись и дата	<div style="display: flex; justify-content: space-between; align-items: center;"> <div> <table border="1" style="width: 100%;"> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> </tr> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </table> </div> <div style="text-align: center; flex-grow: 1;"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ </div> <div style="text-align: right;"> <table border="1" style="width: 100%;"> <tr> <td>Лист</td> </tr> <tr> <td>25</td> </tr> </table> </div> </div>	Изм.	Лист	№ докум.	Подпись	Дата						Лист	25
	Изм.	Лист	№ докум.	Подпись		Дата											
	Лист																
	25																

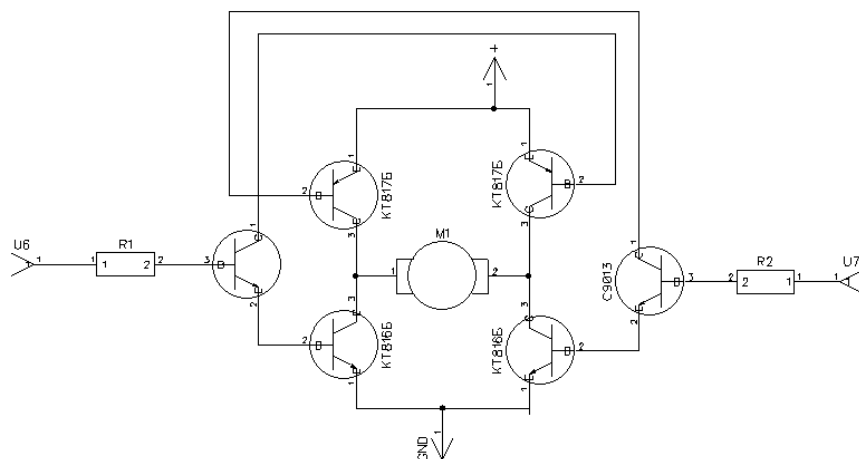



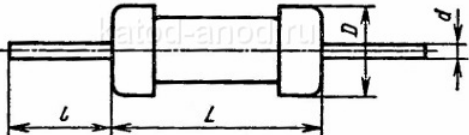
Рисунок 5 – Электрическая принципиальная схема усилителя

Условные обозначения на рисунке 5:

- + - напряжение питания (6 В).
- GND – заземление.
- R1,2 – резисторы.
- M1 – двигатель.
- U6,7 – управляющие входы.
- KT817Б – транзистор.
- KT816Б – транзистор.
- C9013 – транзистор.

Согласно приведенной электрической принципиальной схеме был произведен сбор информации об элементной базе.

Таблица 1 – Сведения об элементной базе

№	Тип компонента	Символ	Конструкция	Примечание
1	Резистор С2-33Н 3,3 кОм		 D = 2,4 мм L = 7,3 мм l = 16 мм d = 0,6 мм	4 шт.

Изм.	Лист	№ докум.	Подпись	Дата	Перв. примен.	
					Справ. №	
					Подпись и дата	
Изм.	Лист	№ докум.	Подпись	Дата	Взам. инв. №	
					Инв. № дубл.	
					Подпись и дата	
Изм.	Лист	№ докум.	Подпись	Дата	Инд. № подл.	
					Инд. № подл.	
					Инд. № подл.	
2	Транзистор C9013		4 шт.			
3	Транзистор КТ817Б		4 шт.			
4	Транзистор КТ816Б		4 шт.			
<p>На основе анализа сведений об элементной базе и некоторых других данных были определены параметры конструкции:</p> <ul style="list-style-type: none">Шаг установки выводов: 2,5 мм; 2,54 мм.Размер выводов: ~0,5 мм.Конструктивная сложность модуля: 5,4 у.е.						
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	
					Лист	
					27	

Перв. примен.					
Справ. №					
Подпись и дата					
Инв. № дубл.					
Взам. инв. №					
Подпись и дата					
Инв. № подл.					

Электронный модуль усилителя был изготовлен в виде печатной платы. Перед изготовлением были определены параметры платы:

- Способ крепления: 2 винта М3.
- Компонентная структура: 1А (элементы печатной платы располагаются на одной стороне и монтируются в отверстия).
- Класс точности: 2.
- Габариты платы: 50х30х1,5.
- Ширина печатного проводника: 0,6.
- Материал: текстолит.

Печатная плата была спроектирована согласно указанным параметрам. Проводящий рисунок печатной платы и расположение на ней элементов представлено на рисунках 6, 7:

Рисунок 6 – Проводящий рисунок печатной платы

						ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
							28
Изм.	Лист	№ докум.	Подпись	Дата			

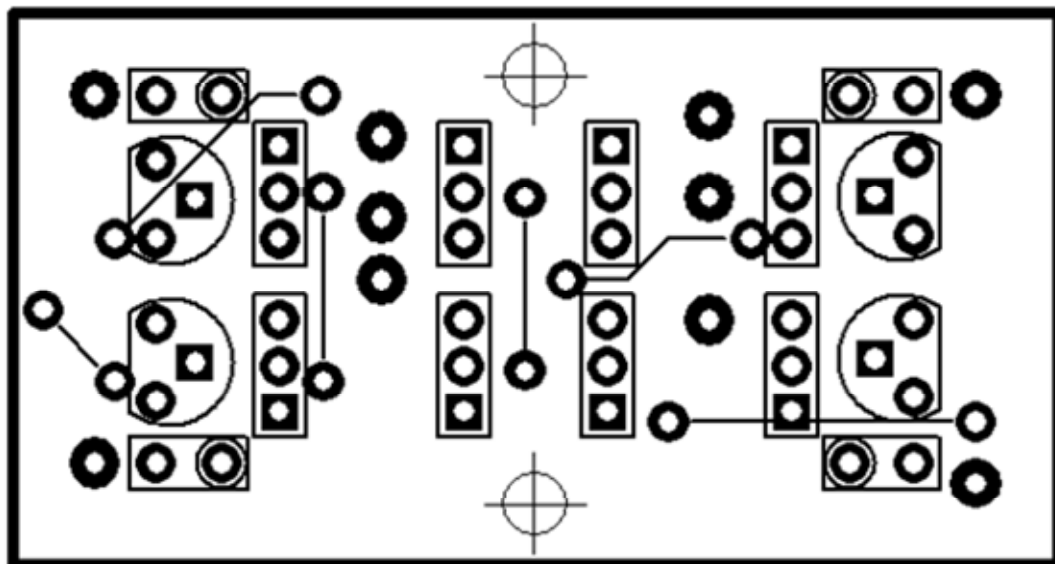


Рисунок 7 – Расположение элементов на печатной плате

При изготовлении печатной платы с указанными выше параметрами были выполнены следующие этапы:

- Нанесение проводящего рисунка печатной платы на текстолитовую пластину.
- Травление рисунка печатной платы.
- Сверление отверстий.
- Пайка элементов.

После выполнения данных этапов изготовления была получена плата усилителя. Работоспособность данной платы была протестирована в ходе работы в составе робототехнического устройства. Внешний вид платы усилителя представлен на рисунке 8:

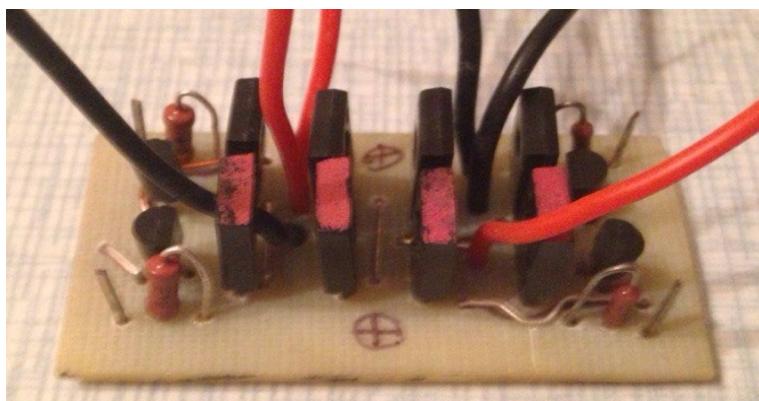
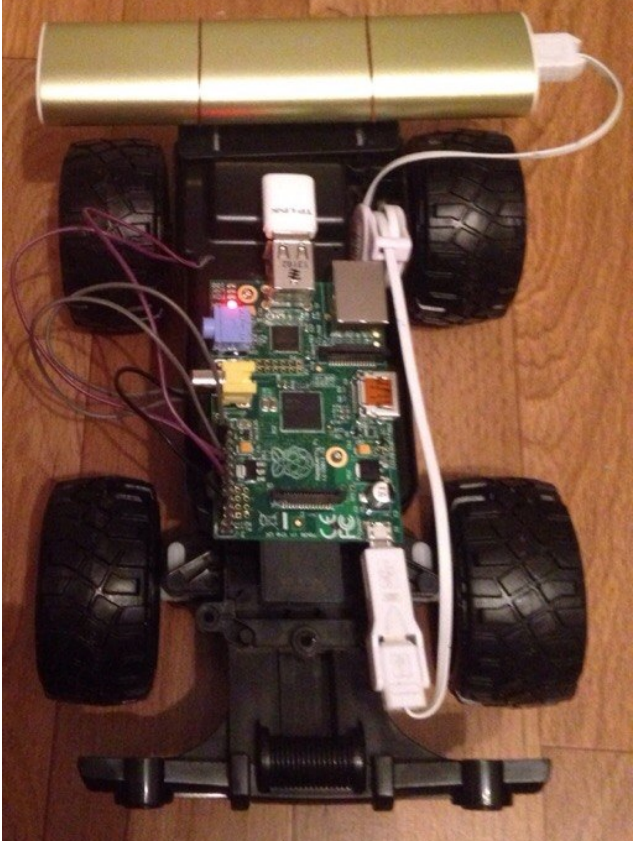


Рисунок 8 – Плата усилителя

Перв. примен.	<h3>4.3 Сборка робототехнического устройства</h3> <p>Сборка робототехнического устройства из составных частей в единую систему – финальная часть выполнения устройства на аппаратном уровне. В данной работе для сборки робототехнического устройства были использованы следующие составные части, детали, элементы:</p> <ul style="list-style-type: none"> • Колесная база с каркасом устройства, 1 шт. • Линейный двигатель, 2 шт. • Батарейки АА, 4 шт. • Плата усилителя, 1 шт. • Одноплатный компьютер Raspberry Pi (в комплектации, описанной в пункте 4.1 «Первоначальная настройка Raspberry Pi»), 1 шт. • Внешний аккумулятор Yoobao YB6014 Pro, 1 шт. • Кабель USB– MicroUSB, 1 шт. • Соединительные провода, 11 шт. • Винт М3, 3 шт. • Гайка М3, 2 шт. <p>Плата усилителя закрепляется одним винтом М3 внутри каркаса устройства. Входы питания платы усилителя припаиваются к выходам батарейного отсека каркаса с помощью двух проводов с соблюдением полярности. Выходы платы усилителя припаиваются к соответствующим входным контактам двигателей с помощью двух проводов для каждого двигателя (всего 4 провода).</p> <p>Одноплатный компьютер Raspberry Pi модели «В» имеет шину портов ввода/вывода (GPIO), состоящую из 26 портов. Некоторые из них имеют определенный функционал по умолчанию (заземление, питание и др.), другие же предназначены исключительно для прикладных целей. В данной работе используются 4 порта, относящиеся ко второй группе. Расположение портов GPIO Raspberry Pi модели «В» приведено на рисунке 9:</p>																							
	Справ. №																							
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.																			
<table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td rowspan="3"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ </td> <td>Лист</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td>30</td> </tr> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> <td></td> </tr> </table>											ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист						30	Изм.	Лист	№ докум.	Подпись	Дата	
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист																		
						30																		
Изм.	Лист	№ докум.	Подпись	Дата																				

Перв. примен.		Справ. №		Подпись и дата		Инв. № дубл.		Взам. инв. №		Подпись и дата		Инв. № подл.							
																			
<p>Рисунок 9 – Расположение портов GPIO Raspberry Pi модели «В»</p> <p>В данной работе используются порты GPIO18, GPIO23, GPIO24, GPIO25, а также порт Ground. Данные порты соединяются с платой усилителя следующим образом:</p> <ul style="list-style-type: none">• Провод, идущий от порта Ground, припаивается к проводу питания платы усилителя для создания так называемой «общей земли».• Порты GPIO18 и GPIO23 проводами соединяются с ножками усилителя, подключенного к двигателю, отвечающему за движение «вперед-назад».• Порты GPIO24 и GPIO25 проводами соединяются с ножками усилителя, подключенного к двигателю, отвечающему за поворот. <p>После соединения контроллера и усилителя отсек каркаса, в котором находится усилитель, закрывается крышкой, которая закрепляется с помощью 1 винта М3. Провода, соединяющие плату усилителя и Raspberry Pi, выводятся наружу отсека через отверстие в крышке. Одноплатный компьютер закрепляется наверху крышки с помощью двух винтов и гаек М3.</p> <p>Одноплатный компьютер Raspberry Pi настраивался и тестировался с помощью питания от электрической сети. Такой тип питания недопустим для робототехнического устройства ввиду отсутствия портативности. Для решения данной проблемы используется внешний аккумулятор. Внешний аккумулятор присоединяется к нижней части каркаса. Кабель USB - MicroUSB соединяет выход внешнего аккумулятора (с силой тока на выходе 2.1 А) и вход питания Raspberry Pi.</p>																			
Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ							Лист							
												31							

Инд. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	Справ. №	Перв. примен.	<p>Батарейки АА (4 шт.) вставляются в специальный отсек для батареек, имеющийся на каркасе устройства.</p> <p>После выполнения всех описанных выше действий робототехническое устройство было собрано и готово к работе. Внешний вид устройства в собранном виде приведен на рисунке 10:</p>				
											
<div> <div>Изм.</div> <div>Лист</div> <div>№ докум.</div> <div>Подпись</div> <div>Дата</div> </div> <div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div> <div> <div>Лист</div> <div>32</div> </div>											

Перв. примен.	<h2>5 Разработка системы на программном уровне</h2> <h3>5.1 Разработка архитектуры программного уровня системы</h3> <p>Программный уровень – неотъемлемая составляющая любой робототехнической системы. Программная часть реализует логику и формирует поведение робота в зависимости от поступления внешних раздражений.</p> <p>Итак, главная задача программного уровня робототехнической системы – формирование поведения робота в зависимости от внешних воздействий. Из этого следуют требования к программной части:</p> <ul style="list-style-type: none"> • Возможность работы с портами GPIO Raspberry Pi. • Организация пользовательского интерфейса, доступного с персонального компьютера. • Организация логики работы с двигателями (подача допустимых сигналов, запрет подачи недопустимых сигналов). • Запуск и работа программных средств на одноплатном компьютере Raspberry Pi, управление ими с персонального компьютера. <p>Исходя из этих требований была разработана архитектура программной системы, которая основана на клиент-серверном принципе. Клиент-сервер - вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами. Преимущества такой архитектуры:</p> <ul style="list-style-type: none"> • Возможность использования множества клиентов для управления робототехническим устройством. • Снижение вычислительных требований к машинам-клиентам, так как все вычисления выполняются на сервере. • Удобная организация защиты данных, хранящихся на сервере. <p>Недостатки архитектуры клиент-сервер:</p> <ul style="list-style-type: none"> • Неработоспособность одного сервера делает неработоспособной всю сеть вне зависимости от количества клиентов. <p>В данном случае вычислительная нагрузка на систему не очень высока, однако требуется управлять робототехническим устройством с других вычислительных устройств. Клиент-серверная архитектура наиболее подходит в этом случае, так как она позволяет организовать управление сервером с множества клиентов, в некоторых случаях, основанных на различных платформах.</p>																							
	Справ. №																							
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.																			
<table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td rowspan="3"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ </td> <td>Лист</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td>33</td> </tr> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> <td></td> </tr> </table>											ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист						33	Изм.	Лист	№ докум.	Подпись	Дата	
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист																		
						33																		
Изм.	Лист	№ докум.	Подпись	Дата																				

Перв. примен.	<h2>5.2 Разработка и реализация серверной части</h2> <p>Как уже отмечалось выше, серверная часть программной системы призвана решать следующие задачи:</p> <ul style="list-style-type: none"> Взаимодействие с портами ввода/вывода GPIO Raspberry Pi. Прием запросов по сети от клиентской части. <p>Для решения этих задач реализация серверной части была выполнена с помощью специального фреймворка WebIOPi. WebIOPi - законченный фреймворк для работы с портами ввода вывода Raspberry Pi. Данный фреймворк позволяет контролировать состояние и управлять всеми портами GPIO локально или удаленно.</p> <p>Серверная часть WebIOPi представляет собой демон в операционной системе Linux, который может быть запущен как в фоновом, так и в обычном режиме. В данной работе этот демон был добавлен с список программ автозагрузки операционной системы Raspberry Pi с помощью команды:</p> <ul style="list-style-type: none"> <code>sudo update-rc.d webiop defaults</code> <p>Данное действие позволило стартовать этот сервис сразу же после загрузки операционной системы одноплатного компьютера.</p> <p>Немаловажной частью сервера WebIOPi является файл конфигурации - config. Данный файл позволяет задавать следующие настройки:</p> <ul style="list-style-type: none"> путь к файлу серверного скрипта в формате name=sourcefile. номер порта, по которому будет работать сервер (по умолчанию - 8000). путь к файлу с хешем пароля сервера. другие настройки. <p>Той часть сервера, которая непосредственно работает с портами ввода/вывода GPIO, т.е. выполняет основную задачу сервера, является серверный скрипт. Согласно спецификации WebIOPi, серверный скрипт должен быть написан на языке программирования Python, а путь к его исходному файлу должен быть добавлен в файл конфигурации сервера. Причем данный фреймворк поддерживает работу с Python как второй, так и третьей версии.</p> <p>Python - высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Его основные архитектурные черты: динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений, удобные высокоуровневые структуры данных. Стандартная библиотека данного языка включает в себя набор удобных средств для работы с портами</p>					Справ. №
Подпись и дата	Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.		
					<div> <div>Изм.</div> <div>Лист</div> <div>№ докум.</div> <div>Подпись</div> <div>Дата</div> </div>	
					<div> <div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div> <div>Лист</div> <div>34</div> </div>	

Перв. примен.		<p>ввода/вывода Raspberry Pi. Именно поэтому Python наиболее удобен для использования при реализации серверной части.</p> <p>Исходный код скрипта на языке программирования Python представляет набор функций и переменных, организующих управление портами ввода/вывода. Рассмотрим основные фрагменты исходного кода:</p> <p>1. Задание номеров портов:</p> <pre># Forward-back motor GPIOs FORWARD_PIN = 18 BACK_PIN = 23 # Right-left motor GPIOs RIGHT_PIN = 25 LEFT_PIN = 24</pre> <p>Номера используемых портов ввода/вывода записываются в переменные, доступные всем функциям программы. Это удобно из-за наличия возможности менять используемый номер порта только в одном месте программы.</p> <p>2. Функция инициализации системы:</p> <pre>def setup(): # Setup GPIOs GPIO.setFunction(RIGHT_PIN, GPIO.OUT) GPIO.setFunction(LEFT_PIN, GPIO.OUT) GPIO.setFunction(FORWARD_PIN, GPIO.OUT) GPIO.setFunction(BACK_PIN, GPIO.OUT)</pre> <p>Все используемые порты GPIO настраиваются как выходы.</p> <p>3. Функции управления портами ввода/вывода:</p> <pre># GPIO control functions def _stop_left(): GPIO.output(LEFT_PIN, GPIO.LOW) def _stop_right(): GPIO.output(RIGHT_PIN, GPIO.LOW) def _stop_forward(): GPIO.output(FORWARD_PIN, GPIO.LOW) def _stop_back(): GPIO.output(BACK_PIN, GPIO.LOW) def _stop_all():</pre>				
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум.	Подпись	Дата	<p>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</p>	
					Лист	35

Перв. примен.		<pre> _stop_forward() _stop_back() _stop_left() _stop_right() def _forward(): GPIO.output(FORWARD_PIN, GPIO.HIGH) def _back(): GPIO.output(BACK_PIN, GPIO.HIGH) def _right(): GPIO.output(RIGHT_PIN, GPIO.HIGH) def _left(): GPIO.output(LEFT_PIN, GPIO.HIGH) </pre> <p>Данные функции соответствуют всем допустимым движениям робототехнического устройства (вперед, назад, влево, вправо), а также их прекращению.</p>				
Справ. №		<p>4. Макросы WebIOPi:</p> <pre> # webiopi macroses @webiopi.macro def forward(): _forward() @webiopi.macro def back(): _back() @webiopi.macro def left(): _left() @webiopi.macro def right(): _right() @webiopi.macro def forward_left(): _left() _forward() </pre>				
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум.	Подпись	Дата	<p><i>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</i></p>	
						<p>Лист</p> <p>36</p>

Перв. примен.	<h3>5.3 Разработка и реализация клиентской части</h3> <p>Как уже отмечалось выше, клиентская часть программного уровня робототехнической системы призвана решать следующие задачи:</p> <ul style="list-style-type: none"> • Предоставление пользователю удобного интерфейса для управления робототехническим устройством. • Взаимодействие с серверной частью. <p>В данной работе было принято решение реализации клиентской части в виде веб-страницы на языке HTML с добавлением скрипта на языке JavaScript. Такое решение имеет следующие преимущества:</p> <ul style="list-style-type: none"> • Возможность реализации экранного интерфейса. • Возможность обработки событий клавиатуры для реализации управления робототехническим устройством с клавиш курсора. • Возможность использования интерфейса с помощью браузера на устройствах различных платформ: от персонального компьютера до смартфона. • Возможность вызовов серверных скриптов с помощью макросов WebIOPi. <p>Рассмотрим основные фрагменты исходного кода клиентской части программного уровня:</p> <ol style="list-style-type: none"> 1. Функции для вызова скриптов серверной части: <pre>function forward() { webiopi().callMacro("forward"); } function back() { webiopi().callMacro("back"); } function right() { webiopi().callMacro("right"); } function left() { webiopi().callMacro("left"); } function forward_left() { webiopi().callMacro("forward_left"); }</pre>				
	Справ. №				

Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
						38

Перв. примен.		<pre>function forward_right() { webiopi().callMacro("forward_right"); }</pre> <pre>function back_left() { webiopi().callMacro("back_left"); }</pre> <pre>function back_right() { webiopi().callMacro("back_right"); }</pre> <p>Каждая из этих функций вызывает по имени один из макросов WebIOPi, описанных в пункте 5.2 "Разработка и реализации серверной части". Также имеются функции для вызова макросов остановки движения.</p> <p>2. Создание и добавление экранных кнопок:</p> <pre>button = webiopi().createButton("bt_forward_left", "_", forward_left, stop_forward_left); \$("#up").append(button);</pre> <pre>button = webiopi().createButton("bt_forward", "/\\", forward, stop_forward); \$("#up").append(button);</pre> <pre>button = webiopi().createButton("bt_forward_right", "_", forward_right, stop_forward_right); \$("#up").append(button);</pre> <p>Данный метод создания кнопок - createButton() - позволяет устанавливать кнопке обработчик события нажатия (mousedown) и отжатия (mouseup). Наличие таких событий крайне удобно для организации управления робототехническим устройством.</p> <p>3. Обработка событий клавиатуры:</p> <pre>\$(document).keydown(function(e) { switch(e.which) { case 37: left(); break;</pre>				
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум.	Подпись	Дата	ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	
					Лист 39	

Перв. примен.	
Справ. №	

Подпись и дата	
Инв. № дубл.	
Взам. инв. №	
Подпись и дата	
Инв. № подл.	

```

case 38:
    forward();
    break;
case 39:
    right();
    break;
case 40:
    back();
    break;
case 32:
    stop();
    break;
}

});

```

Язык JavaScript позволяет анализировать код нажатой кнопки (событие клавиатуры keydown) и вызывать при ее нажатии определенную функцию. Также в программе имеется подобный обработчик для события keyup.

В результате создания клиентской части веб-интерфейс приложения имеет вид:

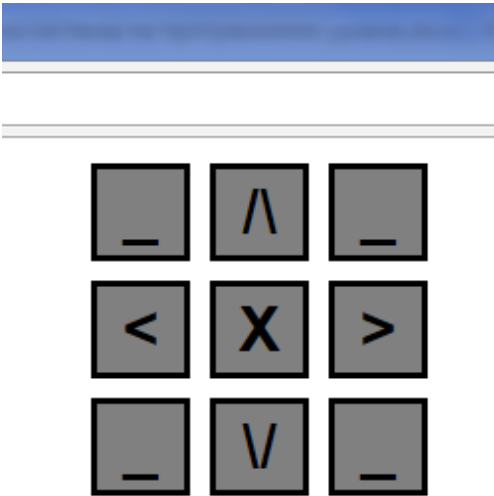


Рисунок 11 - Пользовательский интерфейс

Было организовано 2 способа управления робототехническим устройством:

1. Управление с помощью экранных кнопок. Экранный интерфейс веб-страницы представляет собой 9 кнопок. 4 кнопки со стрелками отвечают за движения влево, вправо, вперед и назад. 4 кнопки со знаком "_" отвечают за комбинации движений вперед + вправо, вперед + влево, назад + вправо, назад + влево. 1 кнопка в центре отвечает за прекращение всех движений.

Перв. примен.	<h2 style="text-align: center;">6 Тестирование системы</h2> <p>Тестирование аппаратно-программной робототехнической системы – процесс исследования, испытания данной системы, имеющий следующие цели: демонстрация того, что произведенный продукт соответствует требованиям; выявить ситуации, в которых поведение системы является неправильным, нежелательным или несоответствующим спецификации.</p> <p>Один из основных видов тестирования – функциональное тестирование. Функциональное тестирование – тестирование продукта в целях проверки реализуемости функциональных требований. Именно этот вид тестирования использовался при тестировании робототехнической системы в данной работе.</p> <p>Перед началом тестирования были разработаны так называемые «тестовые последовательности». Такая последовательность в данном случае представляет собой последовательность действий, совершаемых пользователем системы. Для каждой тестовой последовательности описывается ожидаемый результат, к которому, согласно требованиям, должны привести действия пользователя. После выполнения последовательности полученный результат сравнивается с ожидаемым и делается вывод о прохождении или непрохождении системой данного теста.</p> <p>Тестирование системы производилось в домашних условиях с использованием персонального компьютера с операционной системой Windows и браузера Opera для открытия веб-интерфейса. Управление производилось как с экранных кнопок, так и с клавиш курсора.</p> <p>Для тестирования робототехнического устройства были сформированы следующие тесты (данные тестовые последовательности соответствуют упражнениям, выполняемым на автомобилях при сдаче экзамена на получение водительского удостоверения):</p> <ol style="list-style-type: none"> Последовательность «параллельная парковка»: нажатие кнопок «назад» + «вправо» - нажатие кнопки «назад» - нажатие кнопок «назад» + «влево». <p>Ожидаемый результат: движение назад и вправо – движение назад – движение назад и влево.</p> <p>Полученный результат: соответствует ожидаемому.</p> <p>Вывод: тест пройден.</p> <ol style="list-style-type: none"> Последовательность «змейка»: нажатие кнопок «вперед» + «вправо» - нажатие кнопок «вперед» + «влево» - нажатие кнопок «вперед» + «вправо» - нажатие кнопок «вперед» + «влево». 																		
	Справ. №																		
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.														
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 50%; text-align: center; vertical-align: middle;"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ </td> <td style="width: 10%; text-align: center; vertical-align: middle;"> Лист 42 </td> </tr> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> <td colspan="2"></td> </tr> </table>											ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист 42	Изм.	Лист	№ докум.	Подпись	Дата		
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист 42													
Изм.	Лист	№ докум.	Подпись	Дата															

Перв. примен.	<p>Ожидаемый результат: движение вперед и вправо – движение вперед и влево – движение вперед и вправо – движение вперед и влево.</p> <p>Полученный результат: соответствует ожидаемому.</p> <p>Вывод: тест пройден.</p>							
	<p>3. Последовательность «разворот»: нажатие кнопок «вперед» + «вправо» - нажатие кнопок «назад» + «влево» - нажатие кнопки «вперед».</p> <p>Ожидаемый результат: движение вперед и вправо – движение назад и влево – движение вперед.</p> <p>Полученный результат: соответствует ожидаемому.</p> <p>Вывод: тест пройден.</p>							
Справ. №	<p>4. Последовательность «заезд в гараж»: нажатие кнопок «вперед» + «влево» - нажатие кнопки «назад».</p> <p>Ожидаемый результат: движение вперед и влево – движение назад.</p> <p>Полученный результат: соответствует ожидаемому.</p> <p>Вывод: тест пройден.</p>							
	<p>Для наглядности прохождения всех тестовых последовательностей запечатлено в видеоматериалах, приложенных к данной работе.</p> <p>В процессе тестирования было сформировано 4 тестовых последовательности, всех из которых были успешно пройдены разработанной системой. Это говорит о том, что система полностью соответствует предъявленным к ней требованиям. Стоит также отметить несколько интересных фактов, выявленных при тестировании:</p> <ul style="list-style-type: none"> Робототехническое устройство имеет не самую высокую маневренность. Это обусловлено особенностью конструкции колесной базы, а именно - большим углом разворота. Именно поэтому при выполнении некоторых упражнений пришлось выполнить больше движений, чем предполагалось изначально, что и запечатлено на видео. Разработанный веб-интерфейс позволяет управлять робототехническим устройством с вычислительных машин не только с операционной системой Windows. Было успешно протестировано управление устройством со смартфона компании Apple с операционной системой iOS 8.3 и браузером Safari. Клавиши курсора являются более удобным и точным средством управления роботом, чем кнопки веб-страницы. Возможно, это обусловлено отсутствием необходимости постоянно смотреть на экран и позволяет более точно управлять устройством. С другой стороны, есть вероятность, что удобство 							
Подпись и дата	Подпись и дата	Изм.	Лист	№ докум.	Подпись	Дата	<p align="center">ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</p>	Лист
Изм.	Лист	№ докум.	Подпись	Дата				43

Перв. примен.		Справ. №		<p>управления тем или иным средством зависит от особенностей конкретных пользователей.</p>					
Подпись и дата		Инв. № дубл.		Взам. инв. №		Подпись и дата		Инв. № подл.	
Изм.		Лист		№ докум.		Подпись		Дата	
								ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	
								Лист	
								44	

Перв. примен.						<p>Заключение</p> <p>В результате выполнения выпускной квалификационной работы была спроектирована на различных уровнях и изготовлена аппаратно-программная робототехническая система. Основными задачами данной системы были прием управляющих сигналов от пользователя и перемещение в пространстве на основании этих сигналов.</p> <p>Как показало тестирование робототехнического устройства (см. пункт 6 «Тестирование системы»), весь требуемый функционал выполняется системой в полном объеме. В связи с этим можно заявить, что задачи, поставленные перед началом разработки системы, являются выполненными.</p> <p>Разработанное устройство в настоящее время может использоваться как устройство телеприсутствия. Для этого необходимо его оснащение видео- и/или звукозаписывающей аппаратурой. Данный вариант использования робототехнического устройства продемонстрирован в графических приложениях к работе.</p> <p>Планируется дальнейшее развитие данного проекта. Разработанное робототехническое устройство может быть использовано в качестве основы для аппаратно-программной робототехнической системы распознавания изолированных речевых команд. Для этого необходима реализация одного из алгоритмов распознавания голоса и интеграция его в программное обеспечение робототехнической системы. Работы в этом направлении ведутся автором данной работы совместно с другими участниками проекта по созданию систем распознавания голоса.</p>																		
	Справ. №																							
Подпись и дата		Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.																			
<table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td rowspan="3"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ </td> <td>Лист</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td>45</td> </tr> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> <td></td> </tr> </table>											ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист						45	Изм.	Лист	№ докум.	Подпись	Дата	
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист																		
						45																		
Изм.	Лист	№ докум.	Подпись	Дата																				

Перв. примен.														
Справ. №														
Подпись и дата														
Инв. № дубл.														
Взам. инв. №														
Подпись и дата														
Инв. № подл.														
<p>Список литературы</p> <ol style="list-style-type: none"> 1. Ричардсон М., Уоллес Ш. Заводим Raspberry Pi / пер. с англ. – Амперка, 2013. – 230 с. – ISBN 978-5-4465-0280-6. 2. Прохоренок Н.А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера. – 3-е изд., перераб. и доп. –СПб.: БХВ-Петербург, 2010. – 912 с.– ISBN 978-5-9775-0540-6. 3. Днищенко В.А. 500 схем для радиолюбителей. Дистанционное управление моделями. –СПб.: Наука и техника, 2007. – 464 с.: ил.– ISBN 978-5-94387-358-4. 4. Википедия – Свободная энциклопедия [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/ 5. Основы робототехники и общие сведения о роботах. Устройство роботов [Электронный ресурс]. Режим доступа: http://www.gruppa-prolif.ru/content/view/28/50/ 6. Raspberry Pi – Teach, Learn and Make with Raspberry Pi [Электронный ресурс]. Режим доступа: https://www.raspberrypi.org/ 7. Веб-контроль Raspberry Pi GPIO / Хабрахабр [Электронный ресурс]. Режим доступа: http://habrahabr.ru/post/162651/ 8. Raspberry Pi: первое знакомство Занимательная робототехника [Электронный ресурс]. Режим доступа: http://edurobots.ru/raspberry-pi-pervoe-znakomstvo-urok-4-upravlenie-cherez-gpio/ 														
<table border="1"> <tr> <td>Изм.</td> <td>Лист</td> <td>№ докум.</td> <td>Подпись</td> <td>Дата</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>					Изм.	Лист	№ докум.	Подпись	Дата					
Изм.	Лист	№ докум.	Подпись	Дата										
<div style="display: flex; justify-content: space-between;"> ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ Лист </div> <div style="display: flex; justify-content: space-between;"> 46 </div>														

Перв. примен.						
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
						49
Изм.	Лист	№ докум.	Подпись	Дата		

Перв. примен.					
Справ. №					
<pre>@webiopi.macro def stop(): _stop_all() # Called by WebIOPi at script loading def setup(): # Setup GPIOs GPIO.setFunction(RIGHT_PIN, GPIO.OUT) GPIO.setFunction(LEFT_PIN, GPIO.OUT) GPIO.setFunction(FORWARD_PIN, GPIO.OUT) GPIO.setFunction(BACK_PIN, GPIO.OUT) # Called by WebIOPi at server shutdown def destroy(): # Reset GPIO functions GPIO.setFunction(FORWARD_PIN, GPIO.IN) GPIO.setFunction(BACK_PIN, GPIO.IN) GPIO.setFunction(RIGHT_PIN, GPIO.IN) GPIO.setFunction(LEFT_PIN, GPIO.IN)</pre>					
Подпись и дата					
Инв. № дубл.					
Взам. инв. №					
Подпись и дата					
Инв. № подл.					
					Лист
ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ					50
Изм.	Лист	№ докум.	Подпись	Дата	

Перв. примен.		<div> <div> <h2>Приложение 2. Исходный код клиентской части (index.html)</h2> <pre> <html> <head> <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"> <meta name="viewport" content = "height = device-height, width = 420, user-scalable = no" /> <title>Robot's Control</title> <script type="text/javascript" src="/webiopi.js"></script> <script type="text/javascript"> function init() { var button; button = webiopi().createButton("bt_forward_left", "_", forward_left, stop_forward_left); \$("#up").append(button); button = webiopi().createButton("bt_forward", "\\\"", forward, stop_forward); \$("#up").append(button); button = webiopi().createButton("bt_forward_right", "_", forward_right, stop_forward_right); \$("#up").append(button); button = webiopi().createButton("bt_left", "<", left, stop_left); \$("#middle").append(button); button = webiopi().createButton("bt_stop", "X", stop); \$("#middle").append(button); button = webiopi().createButton("bt_right", ">", right, stop_right); \$("#middle").append(button); button = webiopi().createButton("bt_back_left", "_", back_left, stop_back_left); \$("#down").append(button); button = webiopi().createButton("bt_back", "\\\"", back, stop_back); \$("#down").append(button); </pre> </div> </div>				
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум.	Подпись	Дата	<div> <div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div> <div>Лист 51</div> </div>	

Перв. примен.						
Справ. №						
Подпись и дата						
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
					ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ	Лист
						52
Изм.	Лист	№ докум.	Подпись	Дата		

```
stop_back_right);

button = webiopi().createButton("bt_back_right", "_", back_right,
$("#down").append(button);

$(document).keydown(function(e)
{
    switch(e.which)
    {
        case 37:
            left();
            break;
        case 38:
            forward();
            break;
        case 39:
            right();
            break;
        case 40:
            back();
            break;
        case 32:
            stop();
            break;
    }
});

$(document).keyup(function(e)
{
    switch(e.which)
    {
        case 37:
            stop_left();
            break;
        case 38:
            stop_forward();
            break;
        case 39:
            stop_right();
            break;
        case 40:
```


Перв. примен.		<pre> } function stop_back() { webiopi().callMacro("stop_back"); } function stop_left() { webiopi().callMacro("stop_left"); } function stop_right() { webiopi().callMacro("stop_right"); } function stop_forward_left() { webiopi().callMacro("stop_forward_left"); } function stop_forward_right() { webiopi().callMacro("stop_forward_right"); } function stop_back_left() { webiopi().callMacro("stop_back_left"); } function stop_back_right() { webiopi().callMacro("stop_back_right"); } function stop() { webiopi().callMacro("stop"); } webiopi().ready(init); </script> <style type="text/css"> button { margin: 5px 5px 5px 5px; width: 50px; } </pre>					Справ. №	
Подпись и дата								
Инв. № дубл.								
Взам. инв. №								
Подпись и дата								
Инв. № подл.								
		<div>ВКР-НГТУ-230100-(11-В-2)-015-15 ПЗ</div>						
Изм.	Лист	№ докум.	Подпись	Дата	Лист			
					54			

