

Школа – Инженерная школа информационных технологий и робототехники
 Направление подготовки – 15.03.06 Мехатроника и робототехника
 Отделение школы (НОЦ) – Отделение автоматизации и робототехники

БАКАЛАВРСКАЯ РАБОТА

Тема работы			
Разработка программно-аппаратного обеспечения управления мобильным роботом			
УДК 004.415.2:004.42:004.942:621.865.8-182.3			
Студент			
Группа	ФИО	Подпись	Дата
8Е62	Киселева Дарья Владимировна		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Мамонова Татьяна Егоровна	К.Т.Н.		

Консультант

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОАР ИШИТР	Беляев Александр Сергеевич			

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Конотопский Владимир Юрьевич	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ООД ШБИП	Матвиенко Владимир Владиславович			

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Мамонова Татьяна Егоровна	К.Т.Н.		

Оглавление

Актуальность.	3
Аналоги.	4
Задачи.	8
Разработка программно-аппаратного обеспечения для управления мобильным роботом.....	9
Разработка структурной схемы.	9
Выбор программных и аппаратных компонентов.....	14
Функциональная схема системы управления.	19
Система приема-передачи данных между компонентами.....	22
Математическая модель мобильной платформы.....	26
Алгоритм снятия параметров динамической модели.	36
Вывод.....	39
Список использованной литературы.....	40

Введение.

В современном мире происходит стремительный рост темпа внедрения роботов в промышленности и быту. За последний год в мире было продано более 484 тысяч промышленных роботов, а по оценкам специалистов в 2021 году это число достигнет 630 тысяч [1]. Большая часть внедренных роботов – это роботы, применяемые в помещении, то есть Indoor типа. Объемы мирового рынка Indoor роботов также стремительно растут и к 2026 году достигнут 50 миллиардов долларов [2]. В существующую статистику включены все промышленные роботы indoor типа, однако большая часть из них – это мобильные робототехнические платформы, которые внедряются в складские помещения и туда, где особенно важна скорость и точность перевозки грузов.

Повсеместно происходит замещение человека роботами и привлечение мобильных роботов для решения задач, которые ранее решались без их использования. Процесс внедрения мобильных роботов на производство очень трудоемкий. Необходимо разработать систему управления данным роботом, а также подготовить инфраструктуру помещения, в котором будет эксплуатироваться мобильный робот. Все эти задачи входят в понятие построения программно-аппаратного обеспечения управления мобильным роботом.

Аналоги.

Основным преимуществом мобильных робототехнических платформ является их универсальность. Одни и те же мобильные платформы могут быть применены для решения различных задач, например, транспортировки грузов и построения карты местности. В зависимости от собственных характеристик, для каждой мобильной платформы может быть создана система управления для решения конкретной задачи.

Для того, чтобы познакомиться с существующими решениями по внедрению роботов на производство и склады, были рассмотрены следующие системы: складские роботы Ronavi H1500 (производство Россия), Kiva Amazon Robotics, Toru Cube, SBR robot. Все из перечисленных роботов применяются для работы в складских помещениях, однако базируются на различных алгоритмах и методах навигации в пространстве, требуют различных инфраструктурных и программных внедрений. Их интересующие особенности описаны ниже.

Мобильная платформа Ronavi H1500 предназначена для перемещения паллет и стеллажей весом до 1500 килограмм. Она оснащена поворотными колесами и способна осуществлять всенаправленное движение без поворота платформы. Навигация в помещении осуществляется за счет считывания QR-кодов на полу помещения, а обеспечение безопасности от столкновения с преградой – за счет лидара на борту платформы. Многофункциональность данной платформы обусловлена наличием у нее на борту операционной системой для роботов ROS, что позволяет легко осуществить перепрограммирование и подстройку робота для выполнения определенной задачи. [3]



Рисунок 1 - Мобильная платформа Ronavi H1500

Мобильные роботы Kiva, применяемые на складах Amazon являются лидерами по количеству применяемых в настоящий момент. Только в 2017 году их насчитывалось около 45 тысяч. Роботы Kiva способны перемещать грузы весом до 317 кг, оснащены неповоротным колесами и вращающейся платформой и, так же, как и Ronavi, ориентируются в пространстве по специальной разметке на полу и их бортовой компьютер оснащен операционной системой для роботов ROS. [4]



Рисунок 2 – Складские роботы Kiva

Мобильный робот TORU Cube так же, как и предыдущие роботы, имеет встроенную ROS, но в остальном значительно отличается от рассмотренных выше. Его система управления не требует дополнительного оснащения помещения специальными метками, робот перемещается по построенной заранее карте и по командам оператора, что не может обеспечить полную автономность. Система безопасности робота избегает столкновения с

препятствиями или человеком в рабочей зоне за счет лидаров на борту робота. Особенностью данной системы является возможность постоянного обновления карты для навигации роботов за счет соединения их в локальную беспроводную сеть. Таким образом, если один из роботов обнаружил изменение на складе или новое препятствие, данные об этом вносятся на общую карту. [5]



Рисунок 3 - Мобильный робот TORU Cube

Робот SBR от Украинской компании SB Robotics собран на базе складского робота Буксир-1 компании "Стандарт-ПАК". Данные роботы предназначены для переноса грузов весом до 23 килограмм. Роботы SBR позиционируются как многофункциональные роботы-перевозчики. За счет малой грузоподъемности им сложно найти применение на складах, однако данные роботы хороши для автоматизированной курьерской доставки, в качестве работы официанта и помощников в переносе негабаритных грузов. Для перемещения внутри помещения роботу необходима разметка в виде специальных направляющих линий. Однако, так как робот оснащен микроконтроллером Arduino, его навигация может осуществляться за счет подсоединения дополнительных датчиков и настройки их обработки с помощью Arduino. [6][7]



Рисунок 4 – Робот SBR

Из обзора существующих решений по управлению мобильными робототехническими платформами на складах, в промышленности и для иных целей был выделен недостаток, присущий не только рассматриваемым системам, но и большинству существующих на рынке решений. Основным и самым распространенным недостатком является необходимость подстройки инфраструктуры помещения для внедрения в него мобильной платформы. К таким методам подстройки относятся QR-метки на полу, направляющие линии, штрих-коды и направляющие рельсы. Все это значительно усложняет первичное внедрение роботов, а также их дальнейшую эксплуатацию, так как подобные объекты в промышленных помещениях требуют постоянного обновления, контроля и не дают возможности моментальных изменений внутри помещения.

Основным преимуществом большинства рассмотренных систем является наличие предустановленной специализированной для конкретной модели операционной системы для робота. ROS позволяет легко перенастраивать роботов под выполнение конкретных задач, что значительно расширяет сферу применения каждой мобильной платформы, а ее распространенность позволяет разработчикам систем управления не подстраиваться под конкретную модель мобильного робота.

Задачи.

Исходя из выявленного в обзоре аналогов недостатка, в рамках выпускной квалификационной работы бакалавра ставится цель разработать программно-аппаратный комплекс для управления мобильной робототехнической платформой Kuka Youbot. Для этого цель была разбита на следующие задачи:

- Разработать структурную схему планируемой системы управления;
- Осуществить выбор программных и аппаратных компонентов;
- Разработать функциональную схему системы управления;
- Построить систему приема-передачи данных между выбранными программными и аппаратными компонентами;
- Вывести математическую модель используемой мобильной платформы;
- Разработать алгоритм идентификации параметров математической модели;
- Произвести анализ разработанной математической модели.

Разработка программно-аппаратного обеспечения для управления мобильным роботом

В рамках данного раздела будут продемонстрированы результаты разработки программно-аппаратного обеспечения управления мобильным роботом для решения задачи обезвешивания солнечных панелей. В данной работе не будут рассмотрены особенности процесса обезвешивания, так как необходимо рассмотреть процесс управления мобильной платформой, не опираясь на решение конкретной задачи.

Разработка структурной схемы.

Для грамотного выбора программных и аппаратных компонентов системы необходимо в первую очередь определить требования к проектируемой системе управления и составить ее структурную схему.

При проектировании структурной схемы были проанализированы следующие требования:

- Обеспечить движение мобильной платформы без использования дополнительной инфраструктуры в помещении;
- Создать возможность введения в систему дополнительных датчиков, сенсоров и исполнительных элементов;
- Минимизировать вычислительные ресурсы бортового компьютера мобильной платформы.

Для построения системы управления мобильным роботом необходимо в первую очередь определиться с требованиями к самому мобильному роботу. Для решения задачи по обезвешиванию солнечных панелей, робот должен передвигаться в ограниченном пространстве без препятствий, а его траектория может изменяться только в пределах данного пространства. Таким образом нет необходимости оснащать робота датчиками навигации в помещении, для предотвращения его столкновения с возможными препятствиями. Однако, так как регулировка скорости мобильной платформы носит основополагающий характер, робот, помимо двигателей, должен быть оснащен энкодерами на

каждом из приводов. Для считывания и обработки данных с энкодеров, а также преобразования управляющих команд во вращение двигателей, мобильная платформа должна быть оснащена бортовым компьютером со встроенной системой управления. Таким образом, блок мобильной платформы на структурной схеме должен содержать также блок встроенной системы управления, блок двигателей и блок энкодеров.

Чтобы обеспечить движение платформы без использования дополнительной инфраструктуры, мобильный робот должен быть оснащен рядом датчиков и осуществлять движение согласно командам, подаваемым системой управления. Так как система управления, способная одновременно анализировать данные с датчиков мобильной платформы и строить динамически-меняющуюся траекторию движения робота, требует больших вычислительных ресурсов, а среди поставленных задач требуется минимизировать вычислительные ресурсы встроенного бортового компьютера, было принято решение о внедрении в систему управления мобильной платформой персонального компьютера, который будет анализировать данные с датчиков мобильной платформы, а также с внешних устройств и передавать на робота управляющие команды. Физически связь между персональным компьютером и мобильной платформой может осуществляться проводным или беспроводным путем. В лабораторных условиях и при тестировании системы управления связь будет осуществляться проводным путем, по каналу, который предусматривает выбранная мобильная платформа, чаще всего это Ethernet. При эксплуатации системы на производстве для минимизации дополнительных устройств и проводов соединение и передача данных будет осуществляться беспроводным путем по протоколу, который предусматривает выбранная мобильная платформа, в большинстве случаев мобильные платформы оснащены Wi-Fi модулем.

Для установки связи персонального компьютера и мобильной платформы, для формирования сигналов управления для двигателей, а также для построения программной части системы управления необходимо выбрать

среду разработки и методы или модели программирования. Рабочая среда может быть объединена в некоторый программный пакет, работа с которым значительно упростит процесс создания системы управления.

Одним из требований к системе является возможность внедрения в систему внешних датчиков и устройств. Для упрощения обработки сигналов внешних устройств и возможности беспрепятственного добавления в систему новых датчиков, необходимо установить в систему микроконтроллер, который будет выполнять ЦАП и АЦП функции, а также фильтровать шумы. На стадии проектирования и тестирования, микроконтроллер должен передавать данные с внешних датчиков на персональный компьютер, а также получать с него команды для управления внешними устройствами. Связь между персональным компьютером и микроконтроллером осуществляется по поддерживаемому микроконтроллером протоколу, в большинстве случаев таким протоколом выступает USB. В дальнейшем, для упрощения эксплуатации системы микроконтроллер необходимо будет соединить с бортовым компьютером мобильной платформы.

Внешние устройства, необходимые для решения задачи обезвешивания, в разрабатываемой системе – это тензодатчики, энкодеры, а также линейный привод. Информационные и управляемые каналы данных устройств подключаются к микроконтроллеру по стандартному протоколу MISO/MOSI.

В результате вся система состоит из четырех основных блоков: блок мобильной платформы, блок персонального компьютера, блок микроконтроллера и блок внешних устройств. Физически данные блоки соединяются между собой посредством проводных каналов связи. Обмен данными же происходит между их подблоками. На рисунке 5 представлена разработанная структурная схема, отражающая данную связь.

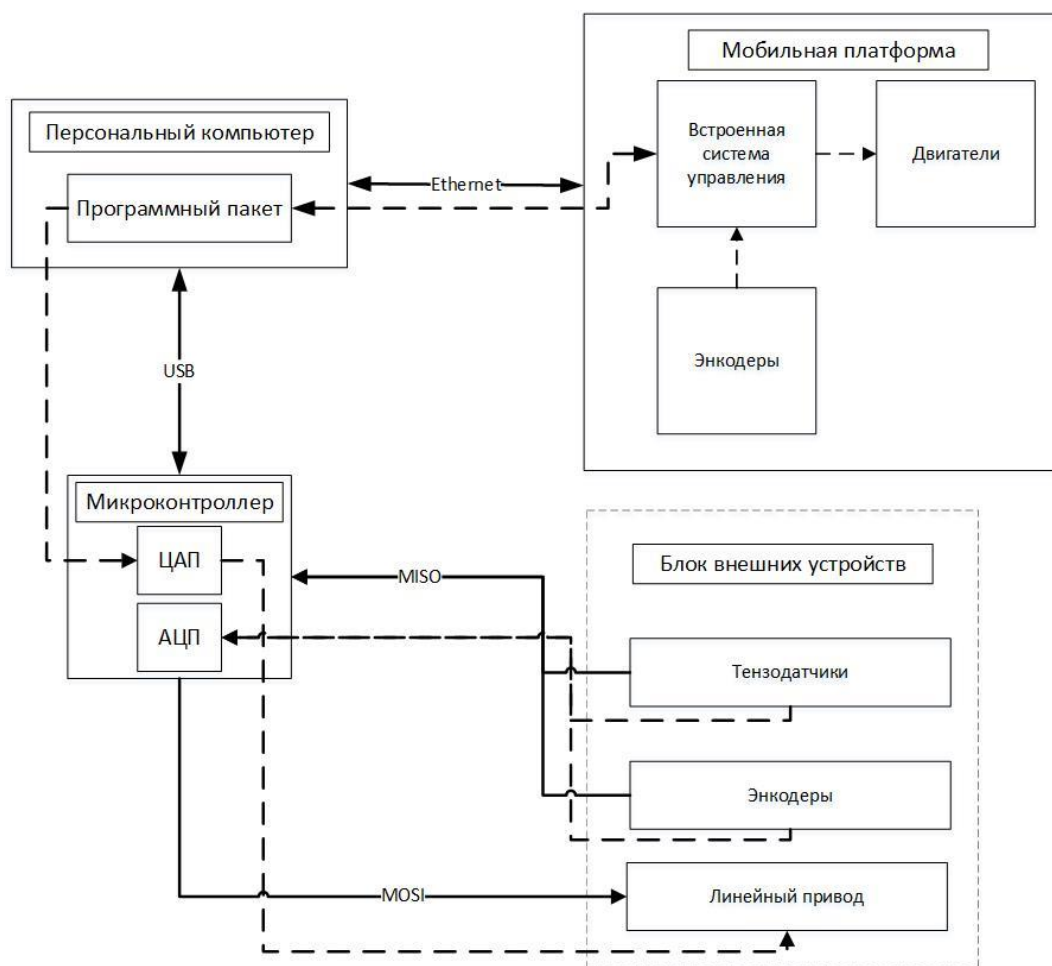


Рисунок 5 – Структурная схема системы управления

На представленной схеме сплошными линиями отражены физические соединения блоков, а также указаны протоколы их соединения. Пунктирными линиями обозначены каналы передачи данных.

Классическая структурная схема для управления мобильным роботом включает в себя следующие блоки: нижний уровень управления (приводный), тактический уровень, стратегический уровень, систему навигации и информационно-измерительную систему [8]. В данной системе роль нижнего уровня управления играют двигатели мобильной платформы. На тактическом уровне происходит преобразование команд управления в скорости двигателей, поэтому тактический уровень в данной системе представляет собой встроенную систему управления мобильной платформой. Роль стратегического уровня управления выполняет программный пакет на персональном компьютере, в котором будут реализованы основные

алгоритмы управления мобильным роботом. Целью информационно-измерительной системы является получение и обработка информации с внешних сенсоров. В данной системе такие функции выполняет микроконтроллер. Система навигации в данном случае представлена лишь энкодерами двигателей мобильной платформы и встроенной системой управления мобильным роботом, которая анализирует данные с энкодеров и передает их на стратегический уровень управления. Блок внешних датчиков представлен тензодатчиками и энкодерами, а также линейным приводом.

После построения структурной схемы необходимо произвести выбор аппаратных и программных компонентов для составления функциональной схемы.

Выбор программных и аппаратных компонентов.

Для реализации спроектированной структурной схемы и преобразования ее в функциональную схему, необходимо произвести выбор аппаратных и программных компонентов.

Значительную роль в выборе аппаратных компонентов сыграло их наличие на базе отделения автоматизации и робототехники ТПУ. В качестве управляемого мобильного робота была выбрана мобильная платформа Kuka YouBot, представленная на рисунке 1.



Рисунок 1. Мобильная робототехническая платформа Kuka YouBot.

Конкурентом при выборе мобильной платформы могла стать мобильная робототехническая платформа для исследований и обучения Robotino компании Festo, однако, в отличие от Kuka YouBot, она не обладает достаточной грузоподъемностью для применения в задаче обезвешивания солнечных панелей, а именно 40 килограмм. В таблице 1 представлены значимые характеристики мобильной робототехнической платформы Kuka YouBot [9].

Таблица 1 – Технические характеристики Kuka Youbot.

Характеристика	Значение
Масса	20 кг
Минимальная скорость	0,01 м/с
Максимальная скорость	0,8 м/с
Стандартное подключение	EtherCAT
Тип колес	Всенаправленные omni-колеса

Аналогично выбору мобильной платформы был произведен выбор микроконтроллера Arduino UNO, который представлен на рисунке 2.



Рисунок 2. Микроконтроллер Arduino UNO.

Значимые технические характеристики используемого микроконтроллера приведены в таблице 2 [10].

Таблица 2 – Характеристики микроконтроллера Arduino UNO.

Базовый микроконтроллер	ATmega328
Число цифровых входов/выходов	14, 6 из них можно использовать для ШИМ
Число аналоговых входов	6

Тактовая частота	16 МГц
Питание и связь	USB

Особое внимание было уделено выбору программных компонентов для реализации спроектированной системы.

Исходя из анализа различных систем управления мобильными платформами, было установлено, что ROS является наиболее распространенной операционной системой, уже установленной в мобильных роботах. Robot Operating System позволит значительно упростить процесс построения системы управления, а также расширит возможности использования созданной системы управления на других мобильных платформах без необходимости адаптации и перестройки системы.

При выборе среды разработки важным критерием являлось простое взаимодействие с ROS, а также имеющийся опыт работы с выбранной средой. В качестве вариантов построения системы управления рассматривался программный пакет MatLab и IDE Visual Studio для написания программного кода управления на языке C++. Сравнительные характеристики двух рассматриваемых вариантов приведены в таблице 3.

Таблица 3. Характеристики среды разработки.

Критерий	Matlab Simulink	Visual Studio
Язык программирования	Язык операторно-структурных схем	C++
Взаимодействие с ROS	Не требует дополнительного ПО	Необходимо дополнительное ПО
Наличие библиотек для работы с ROS	ROS-Toolbox	ros.h

Так как работа с Matlab не потребует в будущем дополнительного программного обеспечения, и опыт работы с данным программным пакетом значительно больше, чем с программированием на языке C++, было принято решение о решении задачи управления роботом и обмена с ним данными с помощью взаимодействия ROS – операционной системы для роботов и пакета Matlab Simulink, установленного на сторонний персональный компьютер. Данная комбинация, хотя и усложняет само управление, но открывает ряд возможностей для создания более совершенного алгоритма для решения конкретной задачи, а также обеспечивает быстроедействие системы и упрощает сложные вычисления, которые могут быть реализованы за счет встроенных возможностей Simulink.

Для того, чтобы более точно понимать построение взаимодействия между Matlab и ROS, необходимо изучить структуру операционной системы роботов и ее особенности. ROS (Robot Operating System) – это фреймворк, необходимый для программирования роботов, робототехнических систем и мобильных платформ [11]. Данная библиотека работает с Unix-подобными системами, включая Ubuntu Linux, которая установлена в том числе на Kuka YouBot. За счет ROS обеспечиваются стандартные службы операционной системы, такие как передача сообщений процессов, управление пакетами и реализация функций. Основными элементами, обеспечивающими работоспособность системы ROS, являются узлы, темы и сообщения, а также мастер, сервис и сервер параметров. Узлы управляют заданными функциями, производят вычисления и обеспечивают связь с другими узлами посредством сообщений определенного типа данных, передаваемых в конкретные темы, доступ к которым у узла обеспечивается за счет сервисов. Каждая тема, как и узел, имеют уникальные имена, а стандартные узлы и темы несут под собой также определенную функцию. Мастер в свою очередь необходим, чтобы обеспечить связь со сторонним компьютером. Сервер параметров обеспечивает изменение конфигураций узлов, работает с памятью и сохраняет передаваемые данные. [12]

Чтобы обеспечить соединение и прием-передачу данных между Matlab и ROS в работе будет применен специализированный пакет ROS-Toolbox, который обеспечивает создание узлов ROS и организует передачу сообщений между двумя системами. Основным преимуществом ROS-Toolbox является то, что он позволяет автоматически генерировать C++ код из модели Simulink и проверять его работу с помощью симуляции или физически на роботе, что позволит в дальнейшем отказаться от использования в системе персонального компьютера, так как построенная модель может работать прямо с бортового компьютера мобильной платформы. Также взаимодействие с Simulink позволяет в режиме реального времени изменять параметры и переменные в системе. Именно это и оправдывает усложнение в целом всей системы введением дополнительного компьютера. Возможности, которые обеспечивает Simulink в связи с ROS, не оставляет сомнения в необходимости обеспечения подобного взаимодействия для решения сложных задач с применением мобильных роботов.

Функциональная схема системы управления.

В соответствии с выбранными программными и аппаратными компонентами была спроектирована функциональная схема системы управления, представленная на рисунке 6.

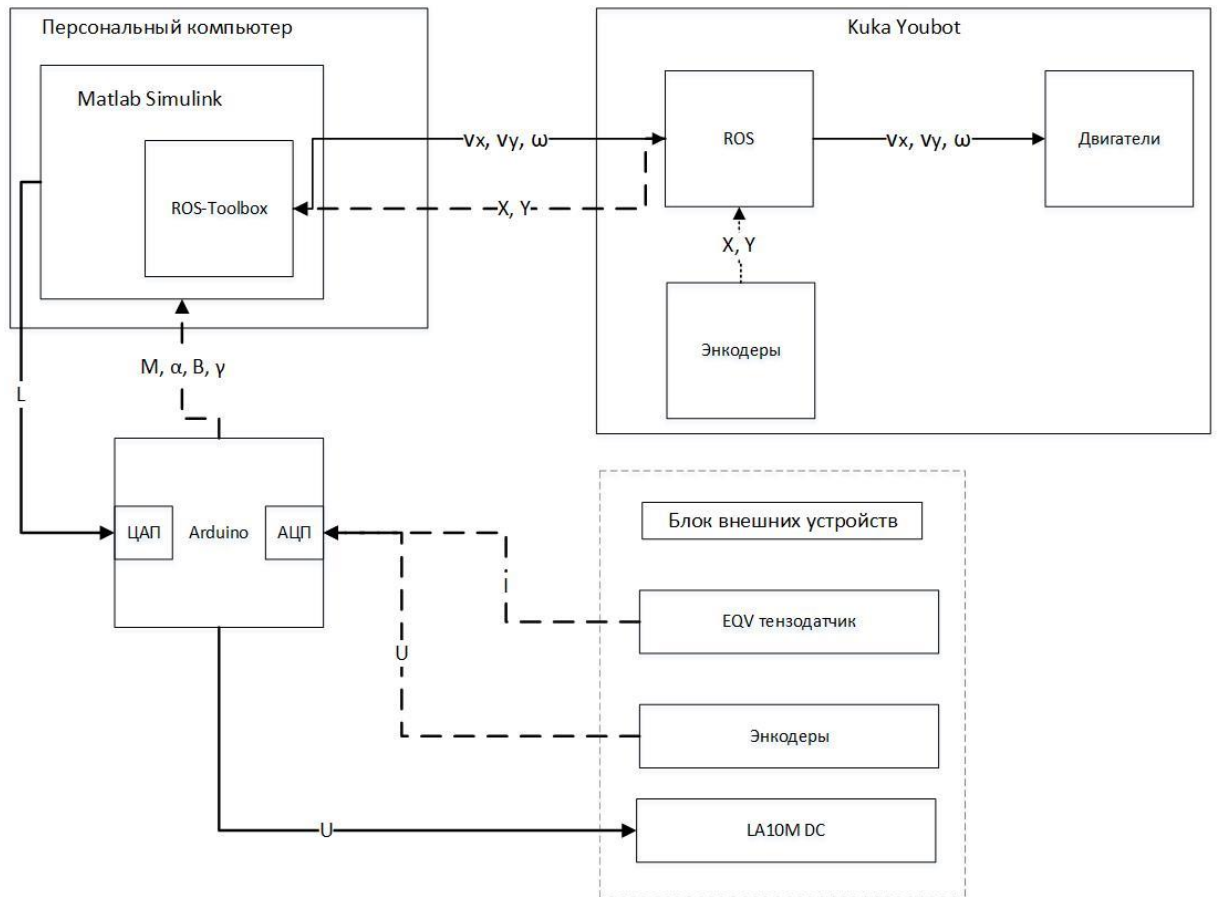


Рисунок 6 – Функциональная схема системы управления

На представленном рисунке сплошные линии обозначают передачу команд управления, а пунктирные – передачу информации. Указаны передаваемые переменные или физические величины – ток и напряжение.

Полноценная система управления работает следующим образом: чтобы мобильная платформа пришла в движение, ROS-Toolbox в Matlab передает на ROS в мобильной платформе сигналы управления в виде сообщений, а именно – три значения скорости (линейные по оси X и Y и угловую). ROS в мобильной платформе преобразует полученные сообщения в команды управления и подает их на двигатели. Платформа приходит в движение. Параллельно ROS

считывает данные с энкодеров приводов и передает их обратно в ROS-Toolbox в виде сообщений. ROS-Toolbox обрабатывает полученные сообщения и вводит их в регулятор скорости платформы в Matlab как обратную связь. Чтобы сформировать значения скоростей для мобильной платформы, Matlab анализирует данные, полученные с Arduino. Также на основе этих данных Matlab передает на Arduino сигналы для приведения в движение линейного привода, а именно высоту, на которую необходимо поднять или опустить привод, который и корректирует высоту устройства удержания солнечной панели. Arduino производит аналогово-цифровое и цифро-аналоговое преобразование поступающих в него данных с Matlab и с внешних датчиков. Остановка робота производится при полном раскрытии солнечной панели. В упрощенном представлении алгоритм данной работы системы представлен на рисунке 7.

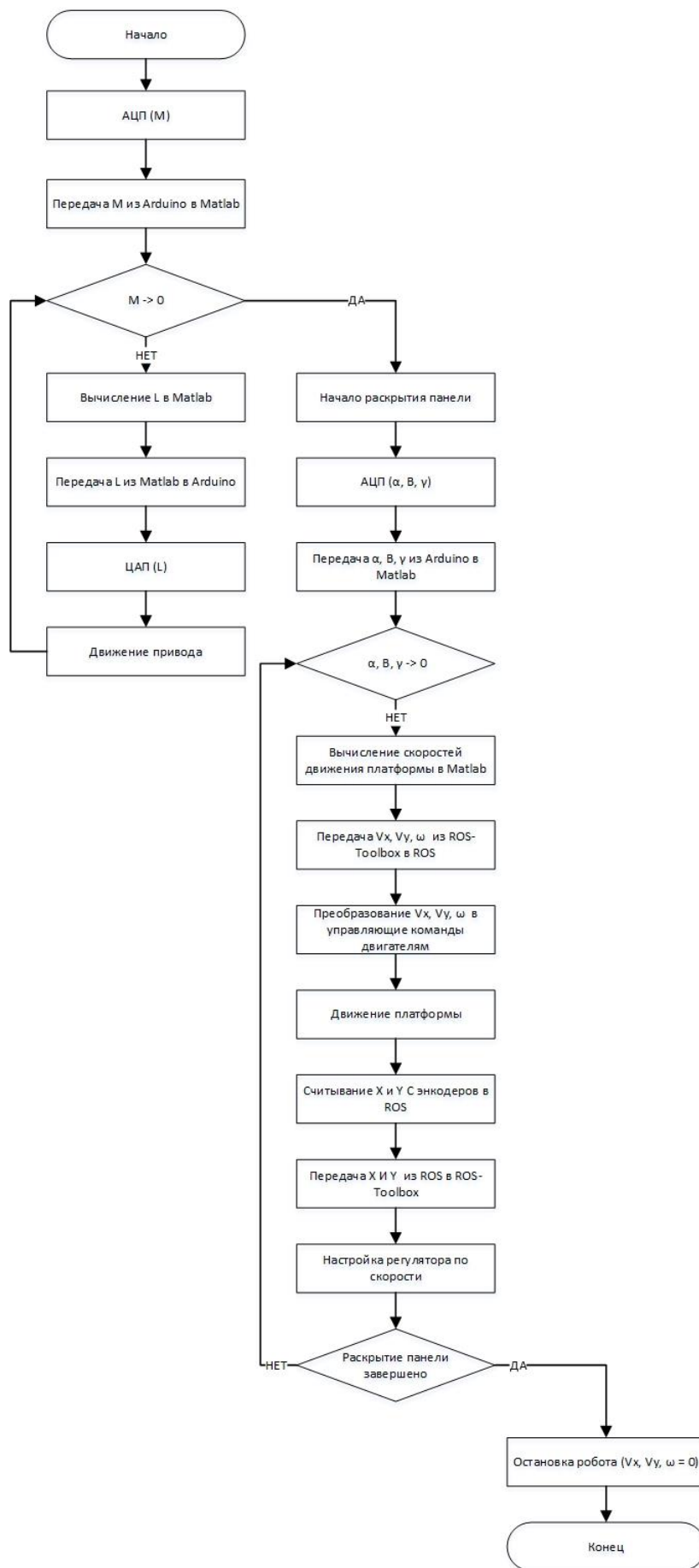


Рисунок 7 – Блок-схема алгоритма работы системы

Система приема-передачи данных между компонентами.

Основой спроектированной системы управления является взаимодействие ROS и Matlab. Как уже было сказано выше, основной обмен данными происходит через специализированный пакет ROS-Toolbox.

Обмен данными между Simulink и ROS может осуществляться тремя основными способами: издатель-подписчик, клиент-сервер и дерево параметров, главные особенности которых представлены ниже.

Модель обмена данными издатель-подписчик требует передачи сообщений по теме, каждая из которых будет иметь свое уникальное имя в ROS, а каждая тема имеет под собой определенный тип данных, которые возможно передать в сообщении данной темы. Узел становится издателем, если хочет отправить сообщение в тему, в то время как другой узел становится подписчиком для получения сообщения из темы. Такая система позволяет реализовывать общение множества узлов, без необходимости их прямой связи, а значит и обеспечивает возможность беспрепятственного удаления или модификации каждого узла [13]. На рисунке 8 представлено графическое отображение модели обмена данными издатель-подписчик.

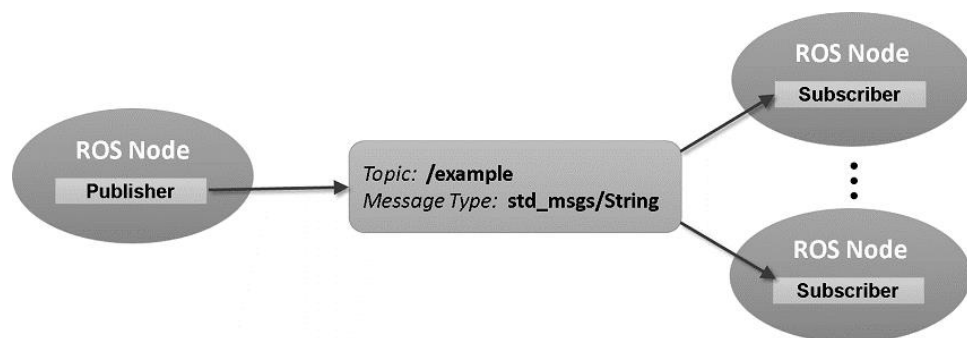


Рисунок 8 - Модель издатель-подписчик

Общение с помощью модели клиент-сервер обеспечивает более сложный обмен данными между узлами. В данной модели клиент отправляет сообщение-запрос и получает сообщение-ответ от сервера. В рассмотренной выше модели обмена данными издатель не требовал ответа от подписчика. Клиент-сервер обеспечивает прямую связь между двумя узлами,

одновременно запрос не может быть отправлен более чем на один сервер. Это подразумевает, с одной стороны, упрощение системы и уменьшение объемов памяти, но с другой стороны невозможность изменять параметры в процессе обмена данными, невозможность реализации множественных вопросов и блокировку узлов клиента до момента ответа сервера [14]. На рисунке 9 представлено графическое отображение модели обмена данными издатель-подписчик.

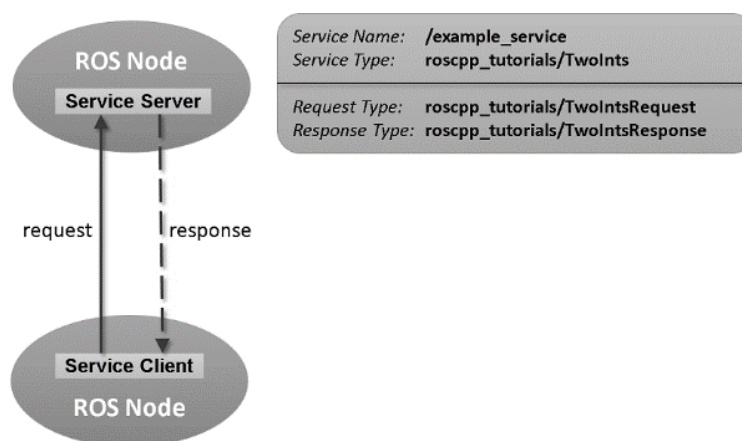


Рисунок 9 - Модель клиент-сервер

Модель общения при помощи дерева параметров отражает взаимодействие, при котором объект Matlab обращается к серверу параметров ROS. Такой метод общения применим для обращения к неизменяемым во времени данным, таким как, например, параметры конфигурации, максимально-допустимые значения переменных и так далее.

Для начала построения системы передачи данных между мобильной платформой и Simulink в первую очередь было выполнено их физическое соединение посредством Ethernet. Далее была проведена настройка локальной сети и протестирована передача данных. Для этого необходимо было узнать IP-адрес самого робота и создать на персональном компьютере локальную сеть с соответствующим IP-адресом. Когда локальная сеть была создана, необходимо было убедиться в том, что мобильная платформа действительно подключена именно к созданной сети – это проверяется в настройках соединения. Тестирование передачи данных осуществлялось при помощи

запуска команды пинга на соответствующий IP-адрес в терминалах соединенных устройств.

После установки соединения, было необходимо настроить на работе мастер ROS, запустив его и проверив его IP-адрес. В случае необходимости нужно было выполнить настройку мастера, передав ему IP-адрес установленного соединения и адрес хоста. В модели Simulink при ее первом запуске адрес мастера определился автоматически, так как до этого связь была установлена верно, а ROS запущен на работе. Иначе настройку мастера необходимо было бы провести вручную в соответствующем окне ROS-Toolbox.

Для отправки данных применялся блок издателя, а также блок создания сообщения для отправки его в тему. В настройках блока издателя были указаны соответствующая тема и тип сообщения, которые будут переданы в узел, также тип сообщения был указан в блоке пустого сообщения. Данные темы и тип сообщений в ROS отвечают именно за задание скоростей мобильного робота. Генерация самого сообщения может быть осуществлена за счет стандартных средств Matlab. Объединение сгенерированного сообщения и блоков издателя осуществлялось с помощью шины Bus Assignment. Проверка передачи данных была проведена с помощью команды отображения активных узлов и задействованных тем в терминале робота. При корректной настройке модели, после вызова данных команд, помимо стандартного узла `rosout`, был также указан новый созданный узел, осуществляющий связь с Matlab. А в темах стали отображаться использованные в модели темы.

Создание подписчика было произведено аналогично и значительно проще, так как оно включает создание только одного блока. В настройках блока подписчика была указана тема, из которой будут считываться данные и тип сообщения, а для использования в модели полученных данных был применен Bus Selector. Тип сообщения и его тем заданы аналогично типам в

блоке издателя. Из темы odometry извлекаются данные о положении мобильной платформы, что необходимо для построения регулятора в будущем.

Структурная схема модели, использованной для тестирования соединения приведена на рисунке 10.

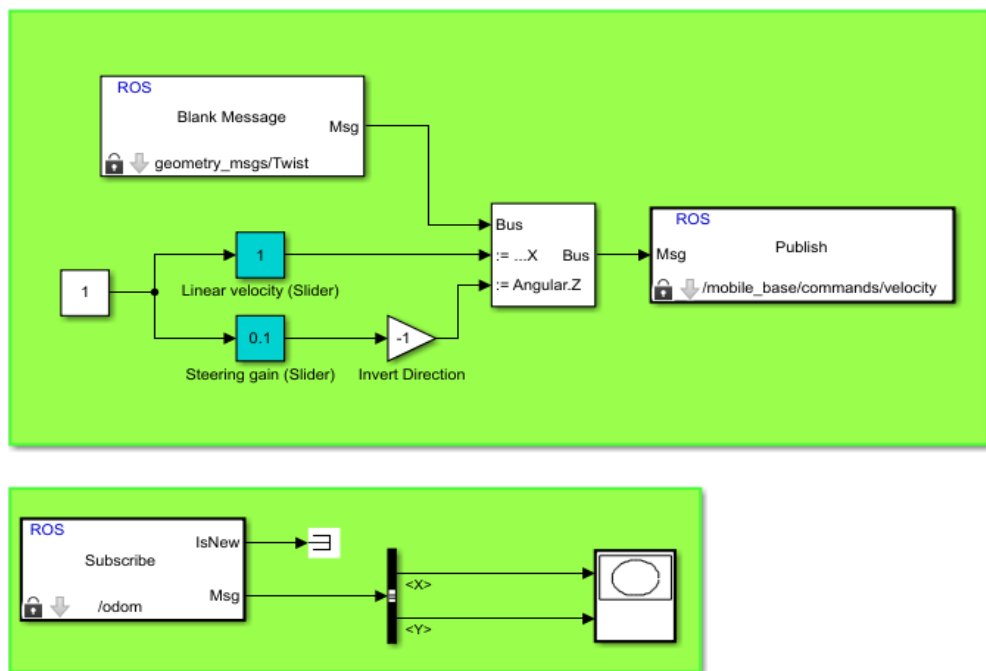


Рисунок 10 – Структурная схема модели

Математическая модель мобильной платформы.

Вывод математической модели робота необходима для построения регулятора, а также проверки работы системы управления в отсутствие непосредственно мобильной платформы. Для полноценного моделирования необходимо вывести кинематическую и динамическую модель.

Кинематическая модель мобильной робототехнической платформы представляет собой решение прямой и обратной задачи кинематики, то есть вывод зависимостей положения платформы от положения ее движителей, и наоборот. Для мобильной платформы Kuka YouBot кинематическая модель представляет собой связь между скоростью движения платформы и скоростью вращения каждого из ее колес.

Kuka YouBot – это четырехколесная мобильная платформа на колесах Илона, поэтому для вывода ее кинематической модели было принято использовать общую формулу (1), представленную в работе [15]. Данная формула может быть использована для вывода кинематической модели любой прямоугольной многоколесной omni-платформы.

$$\dot{\varphi} = - \frac{1}{R \cdot \sin(\delta)} * (\sin(\alpha + \delta) * (v_y + \omega * a_x) + \cos(\alpha + \delta) * (v_x - \omega * a_y)) \quad (1)$$

В данной формуле:

- $\dot{\varphi}$ – Угловая скорость n-ного omni-колеса мобильной платформы;
- R – Радиус колеса;

δ – Угол наклона ролика omni-колеса;

α – Угол наклона колеса относительно платформы;

- a_x, a_y – Координаты центра колеса относительно геометрического центра платформы;

- v_x, v_y, ω – Линейные и угловая скорости мобильной платформы.

В соответствии с параметром наклона колеса мобильного робота Kuka YouBot, а именно $\alpha = 0$, формула (1) преобразуется в формулу (2) за счет внесения $\sin(\delta)$ в общую скобку.

$$\begin{aligned} \dot{\varphi} &= \frac{1}{R \cdot \sin(\delta)} * (\sin(\delta) * (v_y + \omega * a_x) + \cos(\delta) * (v_x - \omega * a_y)) = \\ &= \frac{1}{R} * (v_y + \omega * a_x + \operatorname{ctg}(\delta) * (v_x - \omega * a_y)) \end{aligned} \quad (2)$$

Также для удобства сгруппируем переменные в главной скобке, получив формулу (3).

$$\dot{\varphi} = \frac{1}{R} * (\operatorname{ctg}(\delta) * v_x + v_y + (a_x - \operatorname{ctg}(\delta) * a_y) * \omega) \quad (3)$$

Представим мобильную платформу таким образом, который приведен на рисунке 11. На данном рисунке указаны половины габаритных размеров платформы, обозначенные соответственно L и l , а также на колесах изображены наклоны роликов, контактирующих с подстилающей поверхностью.

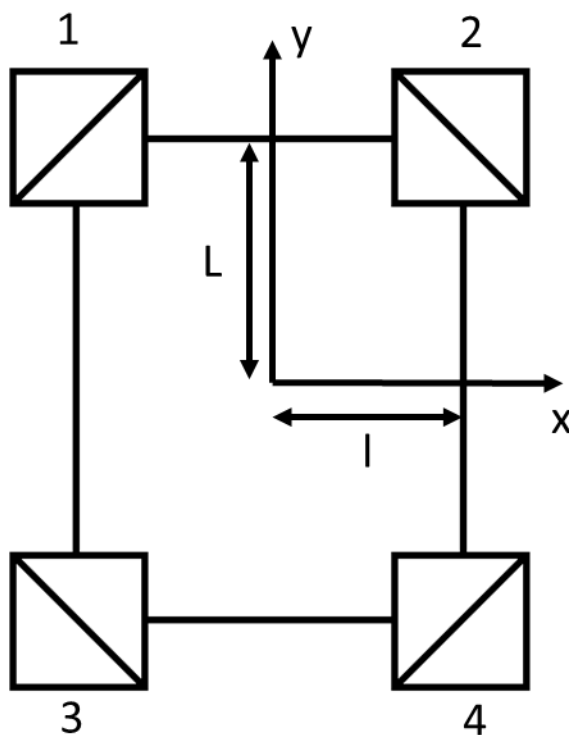


Рисунок 11 – Схематичное изображение мобильной платформы

Укажем координаты центра каждого из omni-колес, а также угол наклона его роликов:

- 1 колесо: $a_x = -l, a_y = L, \delta = 45^\circ$;
- 2 колесо: $a_x = l, a_y = L, \delta = -45^\circ$;
- 3 колесо: $a_x = -l, a_y = -L, \delta = -45^\circ$;
- 4 колесо: $a_x = l, a_y = -L, \delta = 45^\circ$.

Подставив приведенные значения в формулу (3), получим систему уравнений из угловых скоростей каждого колеса (4).

$$\begin{cases} \dot{\varphi}_1 = \frac{1}{R} * (v_x + v_y + (-L - l) * \omega) \\ \dot{\varphi}_2 = \frac{1}{R} * (-v_x + v_y + (L + l) * \omega) \\ \dot{\varphi}_3 = \frac{1}{R} * (-v_x + v_y + (-L - l) * \omega) \\ \dot{\varphi}_4 = \frac{1}{R} * (v_x + v_y + (L + l) * \omega) \end{cases} \quad (4)$$

Запишем полученные уравнения в матричной форме в виде (5), где K – матрица коэффициентов, строки которой являются коэффициентами из системы (4) для каждого колеса, а столбцы – соответствующие скорости v_x, v_y и ω .

$$\dot{\varphi} = - \frac{1}{R} * K * \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix} \quad (5)$$

Таким образом имеем матрицу K (6).

$$K = \begin{pmatrix} 1 & 1 & -L - l \\ -1 & 1 & L + l \\ -1 & 1 & -L - l \\ 1 & 1 & L + l \end{pmatrix} \quad (6)$$

Просуммировав все уравнения системы (4) и сгруппировав по соответствующим переменным, запишем систему (7), которая является решением прямой задачи кинематики для данной мобильной платформы.

$$\begin{cases} v_x = \frac{R}{4} * (\dot{\varphi}_1 - \dot{\varphi}_2 - \dot{\varphi}_3 + \dot{\varphi}_4) \\ v_y = \frac{R}{4} * (\dot{\varphi}_1 + \dot{\varphi}_2 + \dot{\varphi}_3 + \dot{\varphi}_4) \\ \omega = \frac{R}{4(L+l)} * (\dot{\varphi}_1 + \dot{\varphi}_2 - \dot{\varphi}_3 + \dot{\varphi}_4) \end{cases} \quad (7)$$

Подставив известные числовые данные о габаритах мобильной платформы и радиусе колеса, получим численную зависимость скорости движения мобильной платформы от угловой скорости вращения каждого колеса (8).

- $R = 0.05$ м;
- $L = 0,33$ м;
- $l = 0,15$ м.

$$\begin{cases} v_x = 0,0125 * (\dot{\varphi}_1 - \dot{\varphi}_2 - \dot{\varphi}_3 + \dot{\varphi}_4) \\ v_y = 0,0125 * (\dot{\varphi}_1 + \dot{\varphi}_2 + \dot{\varphi}_3 + \dot{\varphi}_4) \\ \omega = 0,026 * (\dot{\varphi}_1 + \dot{\varphi}_2 - \dot{\varphi}_3 + \dot{\varphi}_4) \end{cases} \quad (8)$$

Динамическая модель робота представляет собой оценку влияния внешних силовых факторов на движение мобильной платформы, а также позволяет рассчитать необходимую силу тяги и момент двигателя, чтобы обеспечить требуемое движение мобильной платформы.

Опираясь на выведенную ранее кинематическую модель, рассмотрим движение мобильной платформы вдоль оси ОУ. Согласно системам (7) и (8), для того, чтобы платформа двигалась вдоль оси ОУ, направления вращения всех четырех колес должны быть в одну сторону и по часовой стрелке. Для построения упрощенной динамической модели будут рассмотрены следующие силы, действующие на каждое колесо: сила тяги, сила трения качения колеса, сила трения качения ролика. Расстановка векторов указанных сил приведена на рисунке 12.

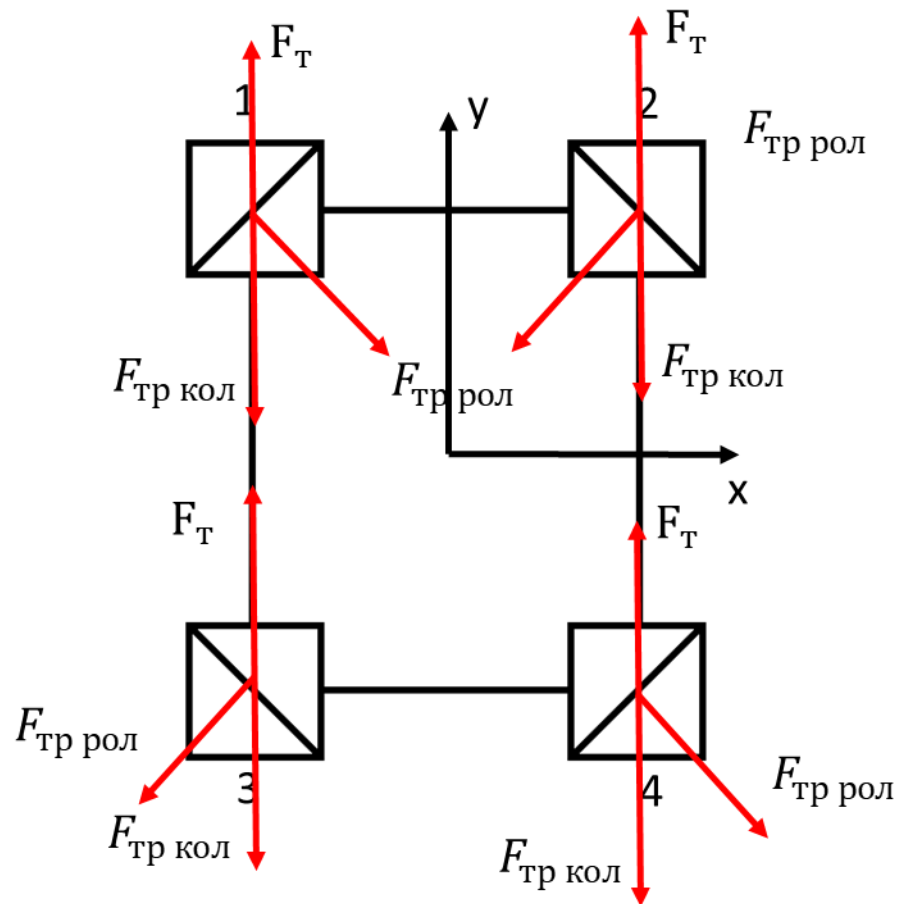


Рисунок 12 – Расстановка сил, действующих на колеса мобильной платформы

На рисунке 12:

- F_T – Силы тяги двигателя;
- $F_{\text{тр кол}}$ – Сила трения качения колеса;
- $F_{\text{тр рол}}$ – Сила трения качения ролика.

Сила тяги направлена по направлению движения мобильной платформы, сила трения качения колеса – в противоположную ей сторону. Сила трения качения ролика направлена перпендикулярно ему в сторону противоположную направлению вращения [16].

На основе рисунка 12 составим уравнение, согласно второму закону Ньютона в проекциях на ось OY (9) [17].

$$F = F_{T1} + F_{T2} + F_{T3} + F_{T4} - F_{\text{тр кол1}} - F_{\text{тр кол2}} - F_{\text{тр кол3}} - F_{\text{тр кол4}} - F_{\text{тр рол1}} * \cos(\delta) - F_{\text{тр рол2}} * \cos(\delta) - F_{\text{тр рол3}} * \cos(\delta) - F_{\text{тр рол4}} * \cos(\delta)$$

(9)

Так как, согласно (7) и (8), для движения вдоль оси ОУ скорости вращения колес должны быть одинаковы, а также допустить, что распределение массы всей мобильной платформы происходит равномерно на каждое колесо, можно утверждать, что все взятые во внимание силы равны для каждого колеса, то есть:

$$\begin{aligned} F_{T1} &= F_{T2} = F_{T3} = F_{T4} = F_T \\ F_{\text{тр кол1}} &= F_{\text{тр кол2}} = F_{\text{тр кол3}} = F_{\text{тр кол4}} = F_{\text{тр кол}} \\ F_{\text{тр рол1}} &= F_{\text{тр рол2}} = F_{\text{тр рол3}} = F_{\text{тр рол4}} = F_{\text{тр рол}} \end{aligned}$$

(10)

Таким образом второй закон Ньютона приобретает вид (11).

$$F = 4 * F_T - 4 * F_{\text{тр кол}} - F_{\text{тр рол}} * \cos(\delta)$$

(11)

В соответствии с [16] и [17] запишем формулы для нахождения сил (12).

$$\begin{aligned} F &= M * a \\ F_{\text{тр}} &= \frac{k}{R} * N \end{aligned}$$

(12)

Для данных уравнений:

- M – Масса мобильной платформы;
- a – Ускорение ее движения;
- k – Коэффициент трения качения;
- R – Радиус колеса;
- N – Сила реакции опоры.

В соответствии с [18] сила реакции опоры определяется формулой (13).

$$N = m * g$$

(13)

В данной формуле:

- m – Масса платформы, приходящаяся на одно колесо;
- g – Ускорение свободного падения.

Так как было принято, что масса платформы равномерно распределена между колесами, можно считать, что: $m = \frac{M}{4}$, тогда формула силы трения качения преобразуется из (12) в (14).

$$F_{\text{тр}} = \frac{k}{R} * \frac{M}{4} * g \quad (14)$$

Будем считать, что R – это радиус колеса, а r – радиус ролика. Тогда с учетом всех вышеупомянутых формул запишем полное уравнение второго закона Ньютона.

$$M * a = 4 * F_{\text{Т}} - 4 * \frac{k}{R} * \frac{M}{4} * g - 4 * \frac{k}{r} * \frac{M}{4} * g * \cos(\delta) \quad (15)$$

Путем переноса слагаемых, выразим из уравнения 15 силу тяги двигателя.

$$F_{\text{Т}} = \frac{1}{4} * (M * a + 4 * \frac{k}{R} * \frac{M}{4} * g + 4 * \frac{k}{r} * \frac{M}{4} * g * \cos(\delta)) \quad (16)$$

В соответствии с [9] и [16] приведены числовые значения переменных для формулы (16).

- $M = 20$ кг;
- $a = 1$ м/с² (ускорение, с которым раскрываются солнечные панели);
- $g = 9.8$ м/с²;
- $k = 0,01$ м (для трения резины по бетону);
- $R = 0,05$ м;
- $r = 0,014$ м;
- $\delta = 45^\circ$.

Подставив числовые значения, найдем силу тяги для каждого колеса.

$$F_{\text{Т}} = \frac{1}{4} * \left(20 * 1 + 4 * \frac{0.01}{0.05} * \frac{20}{4} * 9.8 + 4 * \frac{0.01}{0.014} * \frac{20}{4} * 9.8 * 0.7 \right) = 39,3 \text{ Н} \quad (17)$$

Теперь рассчитаем момент на валу каждого двигателя по формуле (18) [19].

$$M_{вр} = F_T * R \quad (18)$$

Подставив числовое значение, получим:

$$M_{вр} = 39,3 * 0,05 = 1,965 \text{ Н * м}$$

Номинальное значение вращающего момента на мобильной платформе Kuka YouBot [9] составляет 2,15 Н*м, что незначительно превышает полученное выше значение. Если брать во внимание запас по моменту, который есть на двигателях мобильной платформы, можно сделать вывод, что данная динамическая модель имеет может быть использована в дальнейшем при симуляции работы робота. Однако для увеличения точности модели необходимо экспериментально снять значения сил трения качения.

Общая динамическая модель.

Для вывода зависимости между моментами вращения колес и скоростью движения робота, выведем формулу для обобщенной силы тяги мобильной платформы при движении вдоль оси X и Y. Для этого выразим значение силы тяги из формулы (18), а суммарный момент примем пропорциональным скорости вращения колес из формулы (7) кинематической модели. В результате имеем:

$$\begin{aligned} F_{Tx} &= \frac{1}{R} * (M_1 - M_2 - M_3 + M_4) \\ F_{Ty} &= \frac{1}{R} * (M_1 + M_2 + M_3 + M_4) \end{aligned} \quad (19)$$

Полученное значение обобщенной силы тяги при движении вдоль оси OY подставим в формулу (15), преобразовав ее. В результате имеем:

$$m * a = F_{Ty} - 4 * \frac{k}{R} * \frac{m}{4} * g - 4 * \frac{k}{r} * \frac{m}{4} * g * \cos(\delta) \quad (20)$$

Преобразуем полученную формулу, приняв ускорение как производную от скорости движения вдоль оси OY.

$$m * \dot{v}_y = \frac{1}{R} * (M_1 + M_2 + M_3 + M_4) - 4 * \frac{k}{R} * \frac{m}{4} * g - 4 * \frac{k}{r} * \frac{m}{4} * g * \cos(\delta)$$

Обозначив влияние сил трения качения постоянным коэффициентом μ_1 , получим зависимость скорости мобильной платформы от моментов вращения колес:

$$\dot{v}_y = \frac{1}{m * R} * (M_1 + M_2 + M_3 + M_4) - \frac{1}{m} * \mu_1$$

Аналогично выразим зависимость для движения вдоль оси ОХ. Распределение сил при данном движении продемонстрировано на рисунке 13.

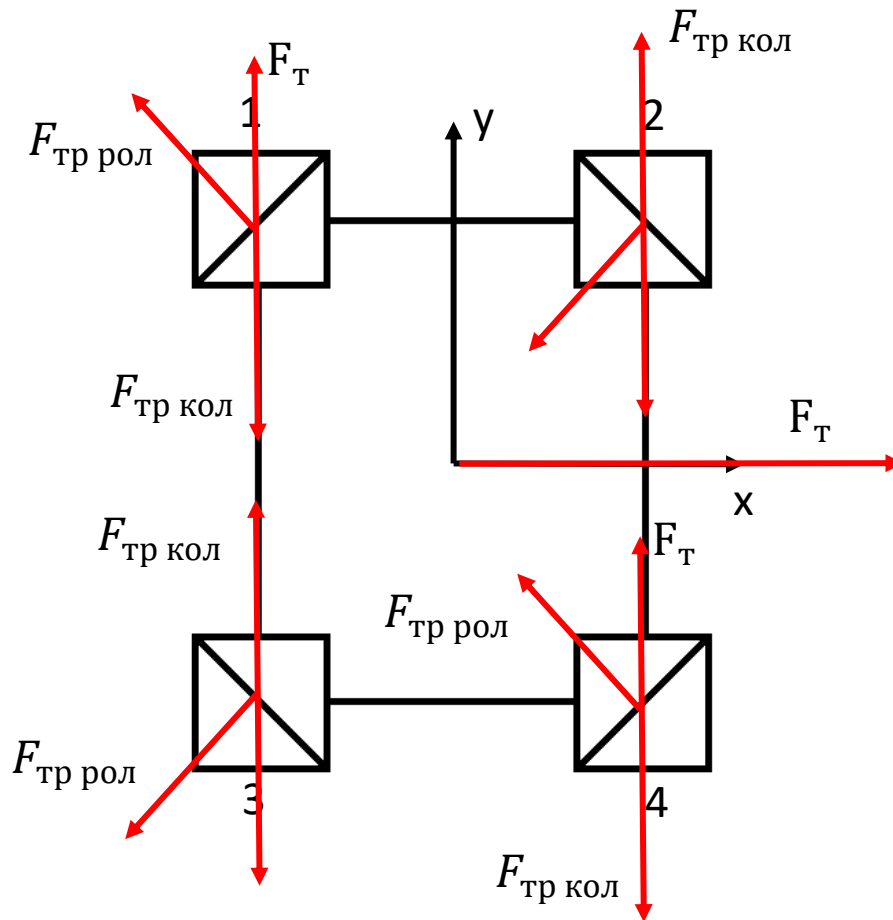


Рисунок 13 – Расстановка сил при движении вдоль оси ОХ

Как видно из рисунка 13 силы трения колес не оказывают влияния на движение вдоль оси ОХ, таким образом, для данного движения уравнение (15) принимает следующий вид:

$$m * a = F_{T_x} - 4 * \frac{k}{r} * \frac{m}{4} * g * \cos(\delta)$$

Преобразуем полученную формулу, приняв ускорение как производную от скорости движения вдоль оси ОХ.

$$m * \dot{v}_x = \frac{1}{R} * (M_1 - M_2 - M_3 + M_4) - 4 * \frac{k}{r} * \frac{m}{4} * g * \cos(\delta)$$

Обозначив влияние силы трения качения постоянным коэффициентом μ_2 , получим зависимость скорости мобильной платформы от моментов вращения колес:

$$\dot{v}_x = \frac{1}{m * R} * (M_1 - M_2 - M_3 + M_4) - \frac{1}{m} * \mu_2$$

Таким образом, общая динамическая модель принимает вид:

$$\dot{v}_y = \frac{1}{m * R} * (M_1 + M_2 + M_3 + M_4) - \frac{1}{m} * \mu_1$$

$$\dot{v}_x = \frac{1}{m * R} * (M_1 - M_2 - M_3 + M_4) - \frac{1}{m} * \mu_2$$

Где:

$$\mu_1 = 4 * \frac{k}{R} * \frac{m}{4} * g + 4 * \frac{k}{r} * \frac{m}{4} * g * \cos(\delta)$$

$$\mu_2 = 4 * \frac{k}{r} * \frac{m}{4} * g * \cos(\delta)$$

Алгоритм снятия параметров динамической модели.

Для точной идентификации параметров математической модели робототехнической платформы Kuka YouBot необходимо разработать алгоритм снятия данных параметров. В ходе выполнения нескольких калибровочных движений будет производиться фиксация реальных скоростей колес и сравниваться с задающими и предполагаемыми моделью скоростями. На рисунке 13 приведен алгоритм для снятия параметров математической модели.

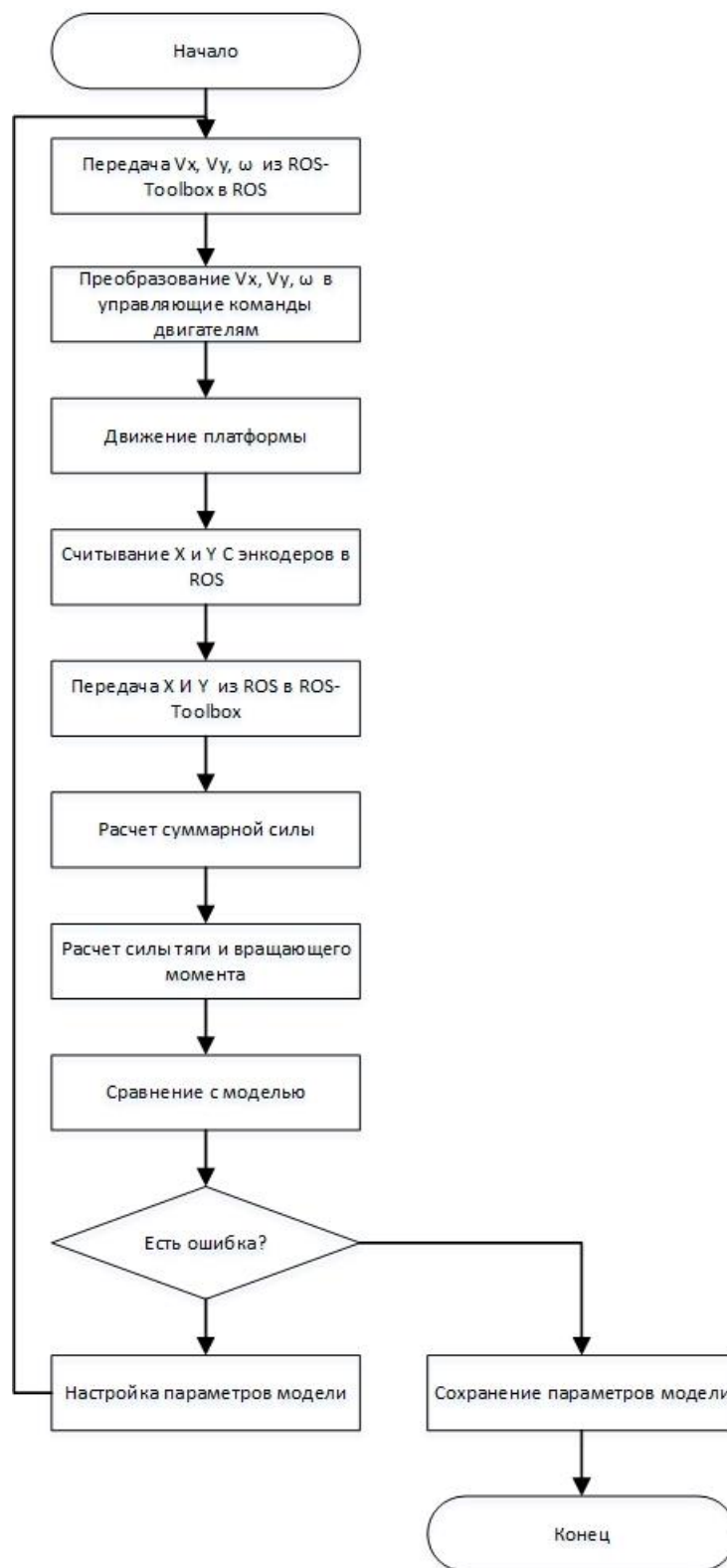


Рисунок 13 – Алгоритм идентификации параметров модели

Реализация данного алгоритма идентификации с применением средств Matlab Simulink и ROS-Toolbox представлены на рисунке 14.

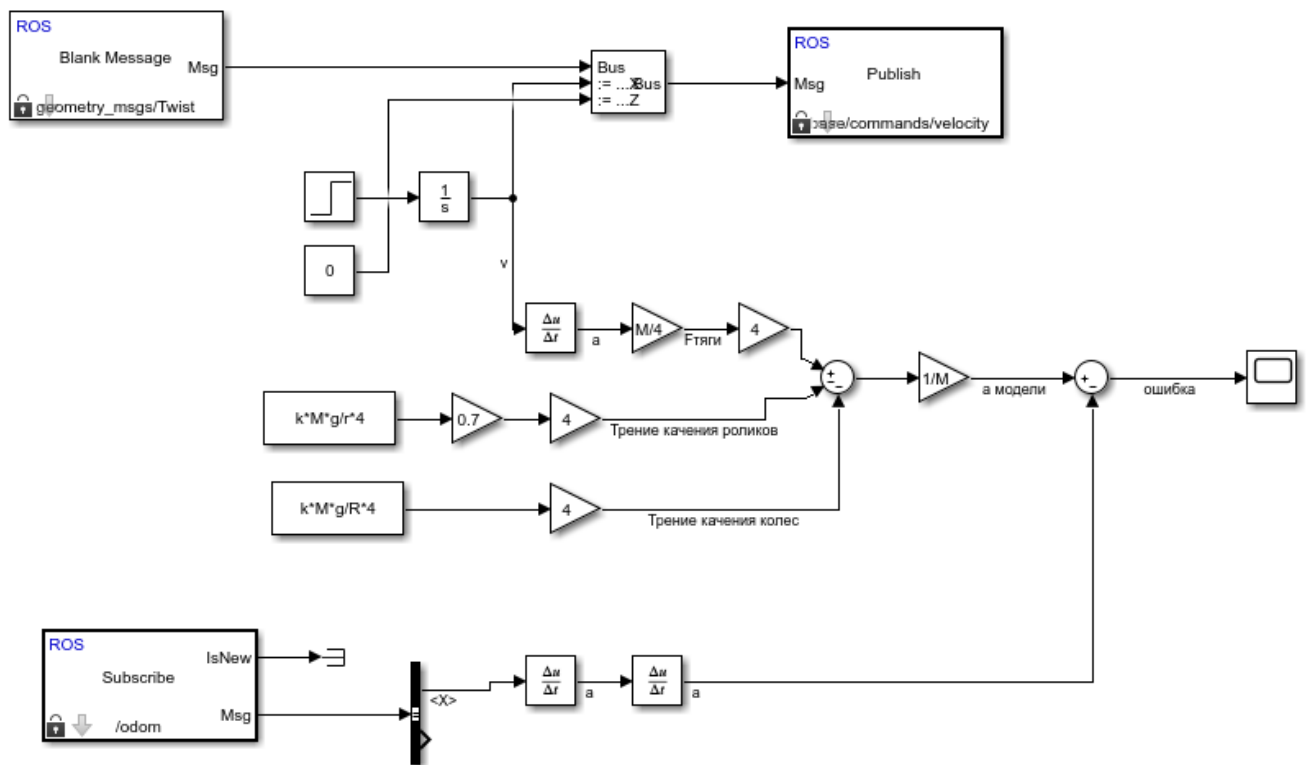


Рисунок 14 – Структурная схема для снятия параметров модели

На рисунке 14 в ROS отправляется сообщение с функцией скорости, а также принимается значение положения мобильной платформы. Так как описанная выше модель рассматривалась для прямолинейного движения вдоль одной из осей, в сообщении ROS с командами скорости указывается лишь одно значение скорости. Для проверки работы математической модели отправленные команды скоростей дифференцируются для получения значения ускорения. Далее рассчитывается идеальное значение силы тяги двигателей робота, без учета возможных сил трения. После этого, на основе данных об идеальной силе тяги и возможных силах трения вычисляется реальное ускорение мобильной платформы. В то же время, после двойного дифференцирования значения положения робота, мы имеем реальное ускорение. Реальное ускорение сравнивается с ускорением, предсказанным моделью и, в случае значительной разницы, совершаются корректировки значения сил трения в модели.

Вывод.

В ходе разработки программно-аппаратного обеспечения управления мобильным роботом были рассмотрены существующие аналоги систем управления мобильными роботами, проведен выбор аппаратных и программных компонентов для реализации системы управления мобильной платформой Kuka Youbot. Были также спроектированы структурная и функциональная схемы будущей системы. Была построена система приема и передачи данных между отдельными компонентами системы, а также выведена математическая модель мобильной платформы, разработан алгоритм снятия параметров модели и спроектирована операторно-структурная схема для снятия данных параметров. Из поставленных перед проектом задач в настоящий момент, из-за сложившейся ситуации, не решена лишь одна – снятие параметров модели с реального робота.

Список использованной литературы.

1. Аналитический обзор мирового рынка робототехники. Сбербанк. https://www.sberbank.ru/common/img/uploaded/pdf/sberbank_robotics_review_2019_17.07.2019_m.pdf
2. Статистика мирового рынка роботов indoor типа. <https://www.statista.com/statistics/822051/global-indoor-robots-market-size/#statisticContainer>
3. Складской робот Ronavi H1500. <http://robotrends.ru/robopedia/ronavi-h1500>
4. Складские роботы Amazon. <http://robotrends.ru/robopedia/kiva>
5. Робот TORU Cube. <https://tqm.com.ua/likbez/interview/avg-roboty-na-sklade>
6. Роботы SBR. <http://robotrends.ru/robopedia/sbr>
7. Складской робот Буксир-1. <http://robotrends.ru/robopedia/buksir>
8. Системы управления интеллектуальных мобильных роботов в среде Dyn-Soft RobSim 5 <http://robsim.dynsoft.ru/design3.pdf>
9. Полная спецификация мобильного робота Kuka YouBot. http://www.youbot-store.com/wiki/index.php/YouBot_Detailed_Specifications
10. Технические характеристики ардуино <http://arduino.ru/Hardware/ArduinoBoardUno>
11. Операционная система робота (РОС) [Электронный ресурс] / Официальный сайт Matlab – URL: <https://ww2.mathworks.cn/help/ros/gs/robot-operating-system-ros.html>
12. Операционная система РОС. [https://ru.wikipedia.org/wiki/ROS_\(операционная_система\)](https://ru.wikipedia.org/wiki/ROS_(операционная_система))
13. Модель обмена данными издатель-подписчик <https://docs.exponenta.ru/R2019a/robotics/examples/exchange-data-with-ros-publishers.html>
14. Модель обмена данными клиент-сервер. <https://docs.exponenta.ru/ros/ug/call-and-provide-ros-services.html>

15. A. Gferrer, Geometry and kinematics of the Mecanum wheel. Comput. Aided Geom. Design 25 (2008), no. 9, 784-791.
16. Сила трения качения. https://ru.wikipedia.org/wiki/Трение_качения
17. Второй закон Ньютона.
<https://sites.google.com/site/zakonifizika/home/zakony-nutona/vtoroj-zakon-nutona>
18. Сила реакции опоры. <http://fizmat.by/kursy/dinamika/sily>
19. Крутящий момент. https://ru.wikipedia.org/wiki/Момент_силы