РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ УПРАВЛЕНИЯ ГОЛОНОМНОЙ РОБОТОТЕХНИЧЕСКОЙ ПЛАТФОРМЫ

Кликич А.С.[1], Поберёзкин Н.И.[2]
[1] Томский Политехнический университет, ИШИТР, студент гр. 8Е11, e-mail: ask182@tpu.ru
[2] Томский Политехнический университет, ИШИТР, ассистент, e-mail: nip6@tpu.ru

Аннотация

В статье рассматривается разработка программного обеспечения для образовательной робототехнической платформы с использованием ROS. Основной упор делается на управлении и передаче данных между компонентами робота. Дополнительно обсуждаются планы по дальнейшему расширению функционала для повышения универсальности и эффективности робота.

Ключевые слова: робот, программное обеспечение, робототехническая платформа, управление роботом, робот на базе ROS.

Введение

В наше время робототехника проникла в различные сферы жизни, от медицины до атомной промышленности, и ключевую роль в этой области играет программное обеспечение. Фактически, роботы не могут функционировать без подходящего программного обеспечения [1].

На данный момент ведущими центрами разработки робототехники являются такие страны, как Япония, Германия и США. В России разработка программного обеспечения для роботов встречает определенные трудности, в частности из-за санкций, что приводит к снижению количества роботов с отечественным программным обеспечением. Это создает проблему нехватки подходящего программного обеспечения для различных типов роботов, что становится все более актуальным [2].

Образовательный процесс в современных университетах сосредоточен на системах управления, которые студенты изучают без использования программного обеспечения на реальных роботах. Например, в Томском Политехническом университете имеется образовательный стенд в виде голономной робототехнической платформы, на котором студенты должны освоить навыки управления роботами, включая кинематику и решение траекторных задач. Однако отсутствие программного обеспечения на этом роботе стало препятствием для полноценного обучения.

Поэтому основной целью данной работы является разработка программного обеспечения для образовательного стенда в виде голономной робототехнической платформы. Это позволит студентам изучать и решать задачи в области управления современными роботами.

На текущий момент сформулированы следующие задачи:

Разработка программного обеспечения для управления робототехнической платформой;

Реализация передачи данных между программным обеспечением и одноплатным компьютером.

Взаимосвязь ПК и одноплатного компьютера

Поскольку робот оснащен двойной аппаратной системой управления, включающей персональный компьютер и одноплатный компьютер, первоочередной задачей было обеспечить передачу данных между этими двумя комплексами. В качестве протокола для передачи данных был выбран интерфейс Ethernet. Для обеспечения коммуникации между этими двумя устройствами была выбрана архитектура клиент-подписчик, позволяющая обеспечить удобный доступ к общим данным [3].

Для взаимодействия между одноплатным компьютером и ПК устанавливаются определенные принципы работы. Один из них выступает в роли издателя, который регулярно публикует данные, в то время как другой действует в качестве подписчика, ожидая получения опубликованных данных. После успешного приема данных подписчик отправляет сообщение издателю о том, что данные получены.

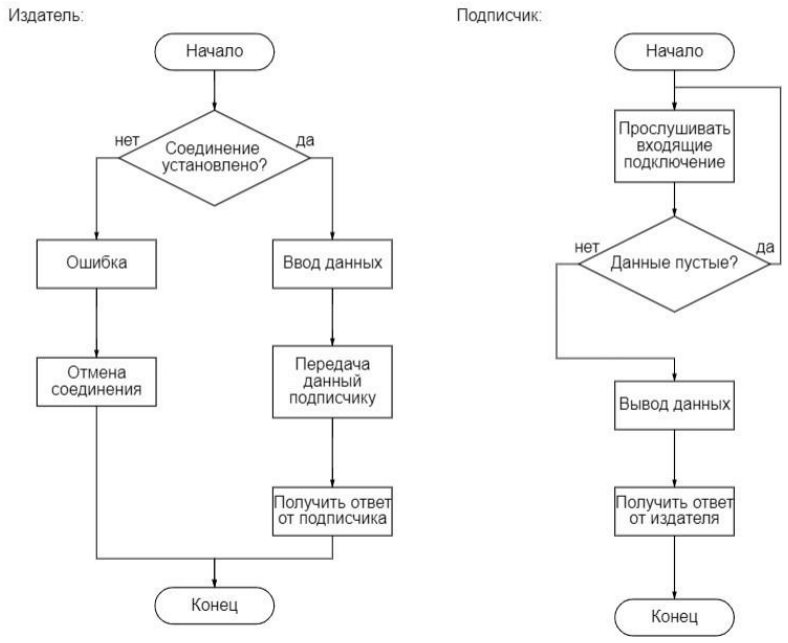

Рис. 1. Блок-схема алгоритма передачи данных

Разработка программного обеспечения

После настройки передачи данных началась разработка пользовательского интерфейса для управления низкоуровневыми операциями. Этот интерфейс включает в себя возможность управления поворотом выбранного пользователем двигателя, направлением движения, скоростью движения или углом поворота, который в дальнейшем будет называться демонстрационным управлением.

На данный момент функция управления низкоуровневыми операциями ограничивается только демонстрационным управлением двигателем. В будущем планируется добавить регулятор управления двигателями, в качестве регулятора управления уже был выбран PID регулятор, в результате необходимо будет настроить три коэффициента регулирования.

После создания алгоритма управления, для более полного понимания взаимосвязей между процессами в программном обеспечении, было разработано дерево вызова процедур (рис. 2). Это дерево отображает иерархию всех процессов внутри программы, их взаимосвязи и последовательность выполнения. Стоит отметить, что данное дерево процессов создано исключительно для функции управления низкоуровневыми операциями, а не для всего программного обеспечения в целом.

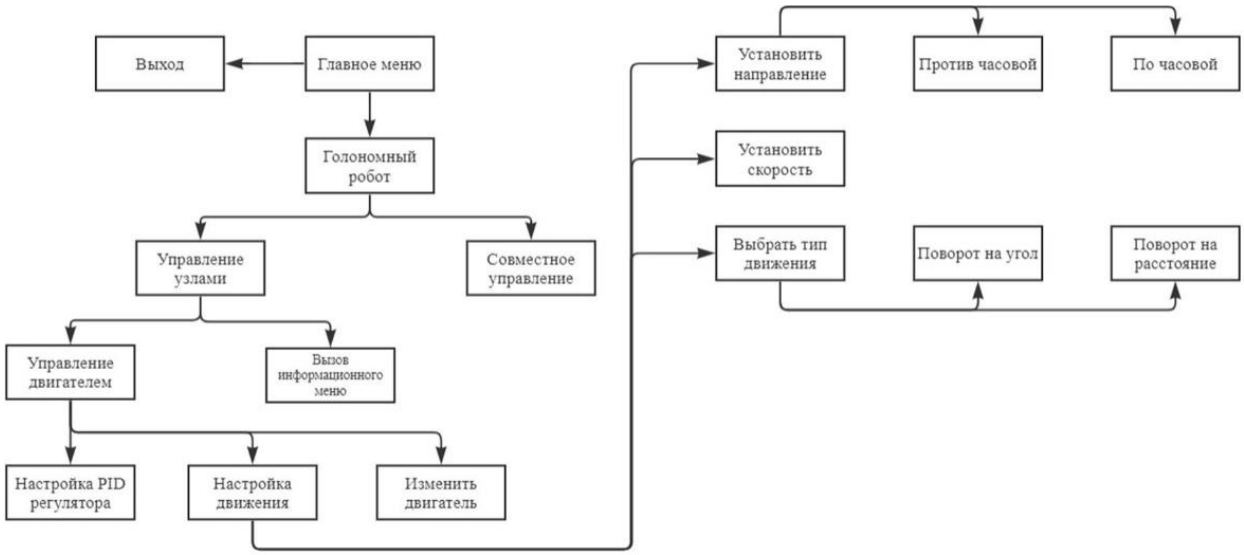

Рис. 2. Дерево вызова процедур

По составленному дереву вызова процедур было реализовано часть приложения для управления низкоуровневыми операциями голономной робототехнической платформы (рис. 3).

Рис. 3. Приложение для демонстрационного управления двигателями

Взаимосвязь ПО и одноплатного компьютера

Передача данных играет важную роль в приложении, так как обеспечивает связь между приложением и одноплатным компьютером. Для организации этой передачи данных было решено объединить две задачи в единое целое, что стало следующим этапом разработки. Результатом этого совмещения стала возможность передачи данных с программного обеспечения на одноплатный компьютер.

ROS

В настоящее время большинство роботов создано на базе Robot Operation System. Для обеспечения универсальности необходимо, чтобы разработанное приложение могло использоваться на любом роботе, который также основан на базе ROS [4, 5].

Алгоритм приема и передачи данных, реализованный на ROS, аналогичен описанному выше. Основное различие заключается в упрощении кода, что является важным фактором для дальнейшего развития программного обеспечения.

Сначала определяются параметры, по которым будет осуществляться процесс движения (рис. 4).

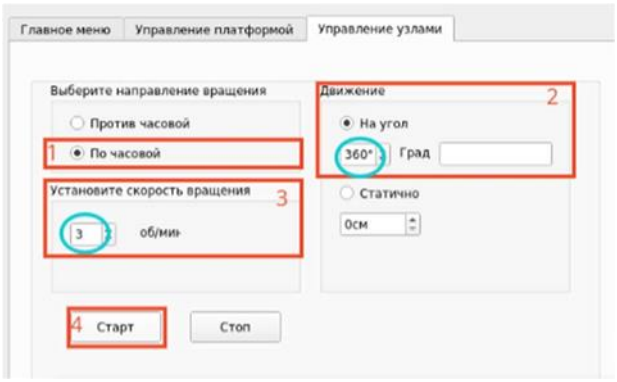

Рис. 4. Ввод параметров

После ввода необходимых параметров и запуска приложения, происходит однократная передача данных в узел подписчика, о скорости и углу вращения колеса (рис. 5).

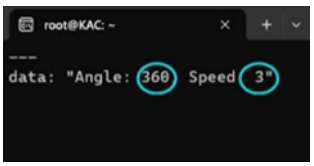

Рис. 5. Вывод параметров

Для проверки корректности передачи данных можно воспользоваться графическим плагином rqt_graph, который позволяет визуализировать процесс передачи данных в ROS (рис. 6).

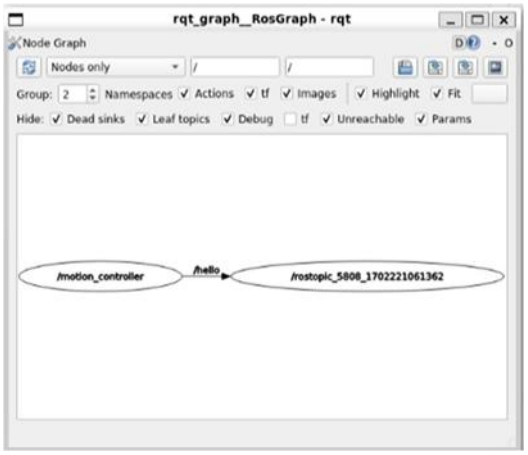

Рис. 6. Визуальный процесс передачи данных

В настоящее время осуществляется передача данных с использованием ROS в пределах симулятора, а в дальнейшем планируется передача результатов на реального робота.

Заключение

В ходе работы было разработано программное обеспечение, разделенное на управление нижним уровнем и протокольный уровень. Это обеспечивает управление роботом и задание параметров движения.

В процессе разработки было выявлено, что количество функций в дереве вызова процедур достаточно для решения поставленной задачи, однако в дальнейшем их можно расширить путем введения дополнительных функций. Функции, которые могут быть введены для расширения дерева вызова процедур, включают дополнительные режимы движения или поведения робота, реализацию различных алгоритмов управления, введение функций диагностики и обратной связи, интеграцию дополнительных сенсоров или актуаторов. Это позволит сделать робота более гибким и адаптивным к различным задачам и ситуациям, улучшив его универсальность и эффективность.

Список использованных источников

1. Роботы заменят людей на опасных участках производства [Электронный ресурс]. – URL:https://rg.ru/2016/10/25/roboty-zameniat-liudej-na-opasnyh-uchastkah-proizvodstva.html (дата обращения 20.02.2023).

2. Почему не нужно боятся импортозамещения PRA-решений [Электронный ресурс]. – URL:https://vc.ru/471724-pix-robotics/455894-pochemu-ne-nuzhno-boyatsya-importozameshcheniya-rpa-resheniy (дата обращения 20.02.2023).

3. TCP/IP. Теория и практика [Электронный ресурс]. – URL: https://habr.com/ru/companies/samsung/articles/571302/ (дата обращения 20.01.2023).

4. Robot Operation System (ROS) [Электронный ресурс]. – URL: https://www.ros.org/ (дата обращения 20.02.2023).

5. Введение и применение ROS [Электронный ресурс]. – https://habr.com/ru/companies/samsung/articles/571302/ (дата обращения 20.01.2023).