

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
образования

ПЕТРОЗАВОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНСТИТУТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

09.03.04 – Программная инженерия

Отчет о прохождении производственной практики

Автономное движение и навигация мобильного робота(NASA) + ROS

Выполнил студент группы 22207

Лисицкий Олег Витальевич

Направление подготовки:

Системное и прикладное программное обеспечение

Место прохождения практики:

Кафедра ИМО, ЦИИ

Сроки прохождения практики:

30.05-09.06

Руководитель практики:

к.т.н., доцент

Богоявленская Ольга Юрьевна

Оценка _____

Дата _____

Петрозаводск — 2022

Перечень терминов

В ходе описания отчёта по производственной практике будут использоваться следующие термины:

- Gazebo - программа с открытым исходным кодом, моделирующая работу робота.
- ROS - Robot Operating System - программная платформа, назначение которой – написание программного обеспечения робота
- SLAM - Simultaneous Localization and Mapping - метод навигации.
- Turtlebot3 Burger - модель мобильного робота.

Содержание

Введение	4
1 ROS	4
1.1 Что такое ROS?	4
1.2 использование ROS для выполнения заданий практики	5
2 Gazebo	6
3 Автономное движение и навигация мобильного робот	7
4 Заключение	9
Список литературы	10

Введение

Марсоходы созданы для того, чтобы исследовать безлюдные пр В ходе учебной практики передо мной была поставлена цель в изучении способов Автономного движения и навигация, использующихся в исследовательской робототехники НАСА, а также получить практические навыки работы с операционная система для роботов

Таким образом, вытекают следующие задачи практики:

- получить базовые умения и знания об ОС для роботов
- найти способ организации автономного движения и навигации с использованием её
- используя полученные знания создать компьютерную симуляцию автономного движения и навигации некоторой модели существующего мобильного робота

1 ROS

1.1 Что такое ROS?

ROS (Robot Operating System) — Операционная система для роботов — это экосистема для программирования роботов, предоставляющая функциональность для распределённой работы. ROS был первоначально разработан в 2007 году под названием *switchyard* в Лаборатории Искусственного Интеллекта Стэнфордского Университета для проекта.

ROS обеспечивает стандартные службы операционной системы, такие как: аппаратную абстракцию, низкоуровневый контроль устройств, реализацию часто используемых функций, передачу сообщений между процессами, и управление пакетами. ROS основан на архитектуре графов, где обработка данных происходит в узлах, которые могут получать и передавать сообщения между собой. Библиотека ориентирована на Unix-подобные системы

1.2 использование ROS для выполнения заданий практики

Мной были пройдены пошаговые обучающие инструкции начального уровня сложности, предложенные на официальном сайте её разработчиков. Инструкции включали в себя следующие темы:

- Установка и настройка среды ROS
- Навигация по файловой системе ROS
- Создание пакета ROS
- Сборка пакета ROS
- Понимание узлов(nodes) ROS
- Понимание тем(topics) ROS
- Общие сведения о службах и параметрах ROS
- Использование утилит `rqt_console` и `roslaunch`
- Использование утилиты `roscd` для редактирования файлов в ROS
- Создание файлов ROS `msg` и `srv`

Для выполнения задач практики был установлен наиболее актуальный дистрибутив ROS, ROS noetic, в который входили инструмент работы с симуляциями Gazebo. Используя изученные обучающие материалы, мной было создано рабочее пространство с пакетами для работы с моделью робота turtlebot3:

Листинг 1 – Подготовка необходимых пакетов для работы

```
$ source /opt/ros/noetic/setup.bash
$ mkdir -p ~/catkin_ws/src
$ cd ~/catkin_ws/
$ catkin_make
$ source devel/setup.bash
$ cd ~/catkin_ws/src/
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3_msgs.git
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3.git
$ cd ~/catkin_ws && catkin_make
```

turtlebot3 существует в трех различных комплектациях, и для выполнения связанных с ним команд в терминале необходимо указать, какой будет использован. Была выбрана малогабаритная комплектация burger. Для этого достаточно в `/.bashrc` с помощью **gedit** прописать дополнительную строчку **export TURTLEBOT3_MODEL=burger**. После этого надо добавить `/.bashrc` как источник зависимостей, а также получить пакеты для симуляции робота в Gazebo:

Листинг 2 – Подготовка необходимых пакетов для работы

```
$ source ~/.bashrc
$ cd ~/catkin_ws/src/
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git
$ catkin_make
$ cd ~/catkin_ws && catkin_make
```

2 Gazebo

Gazebo — это 3D - симулятор робототехники с открытым исходным кодом . Он интегрировал физический движок ODE , рендеринг OpenGL и код поддержки для моделирования датчиков и управления приводами. Использован версия Gazebo 11.

В ходе практики использовался как среда для симуляции движения и навигации робота для запуска симуляции используется следующая команда:

Листинг 3 – использование Gazebo

```
$ roslaunch turtlebot3_gazebo turtlebot3_world.launch
```

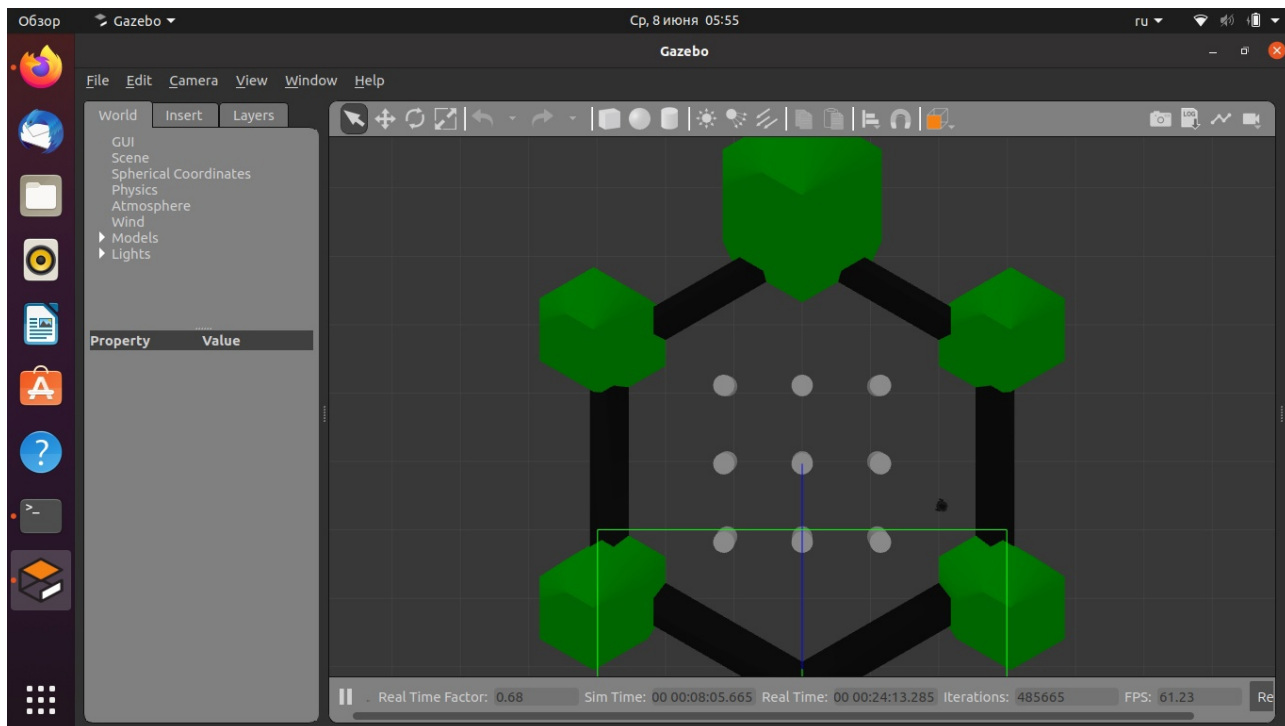


Рис. 1 – Симуляция в Gazebo

3 Автономное движение и навигация мобильного робот

Для выполнения этой задачи потребовалось до установить пакеты, связанные с составлением карт и навигации робота:

Листинг 4 – использование SLAM и автономное передвижение. Каждая из этих команд и команда из Листинга 3 выполняются в отдельных окнах терминала

```
sudo apt-get install ros-noetic-joy ros-noetic-teleop-twist-joy \
ros-noetic-teleop-twist-keyboard ros-noetic-laser-proc \
ros-noetic-rgbd-launch ros-noetic-rosserial-arduino \
ros-noetic-rosserial-python ros-noetic-rosserial-client \
ros-noetic-rosserial-msgs ros-noetic-amcl ros-noetic-map-server \
ros-noetic-move-base ros-noetic-urdf ros-noetic-xacro \
ros-noetic-compressed-image-transport ros-noetic-rqt* ros-noetic-rviz \
ros-noetic-gmapping ros-noetic-navigation ros-noetic-interactive-markers
```

Автономное движение и навигация были реализованы с помощью Метода SLAM (simultaneous localization and mapping) – метод одновременной локализации и построения карты. Используется в мобильных автономных средствах для построения или обновления карты неизвестной местности, одновременно отслеживая местоположение робота в ней.

Листинг 5 – использование SLAM и автономное передвижение. Каждая из этих команд и команда из Листинга 3 выполняются в отдельных окнах терминала

```
$ roslaunch turtlebot3_slam turtlebot3_slam.launch slam_methods:=gmapping  
$ roslaunch turtlebot3_gazebo turtlebot3_simulation.launch
```

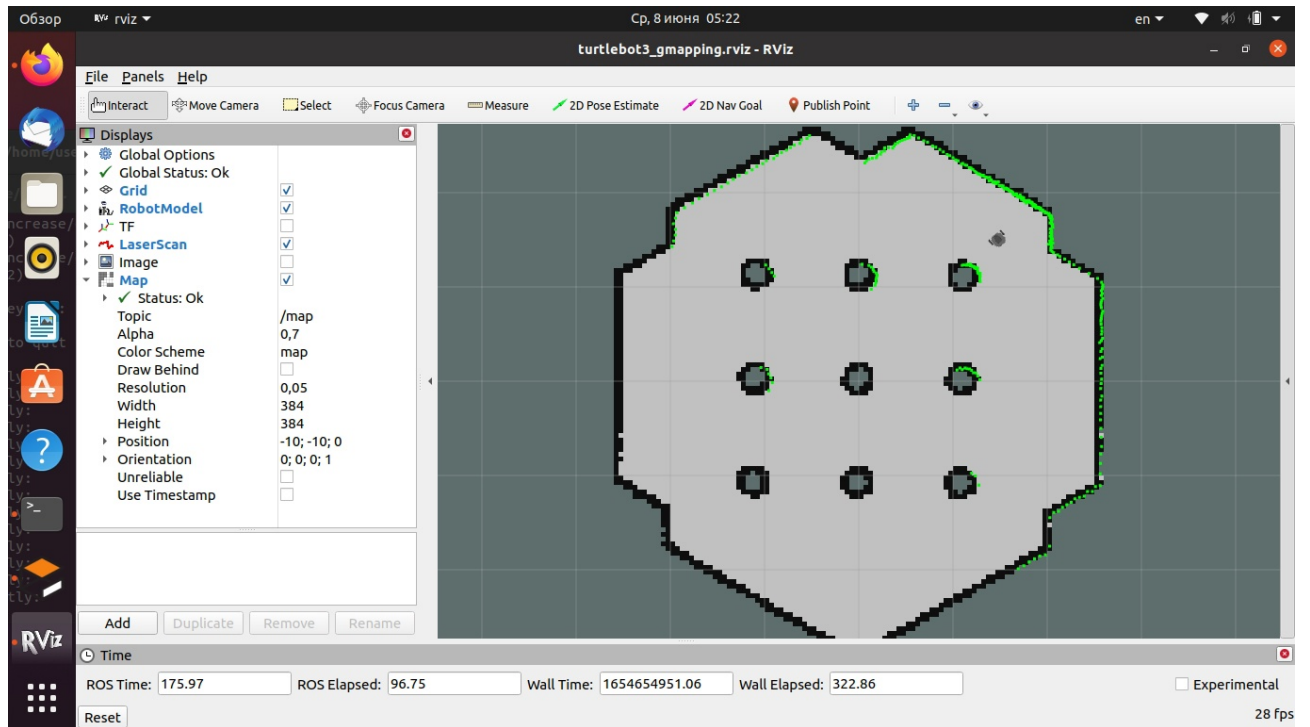


Рис. 2 – Результат составления карты при автономном движении

При автономном движении робот обходит предоставленное ему пространство и заполняет карту, которая затем сохраняется для автоматического оптимальных маршрутов и ориентирования робота по ней

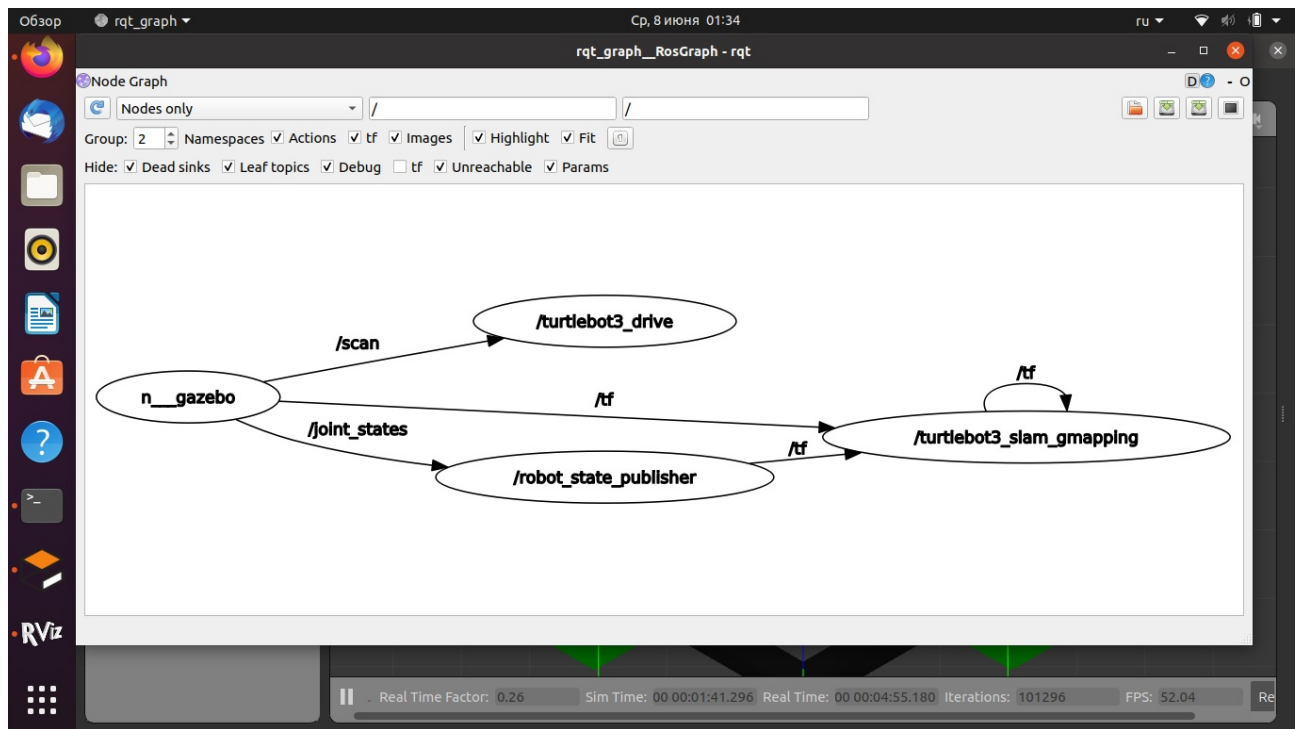


Рис. 3 – Демонстрация графа взаимодействия процессов

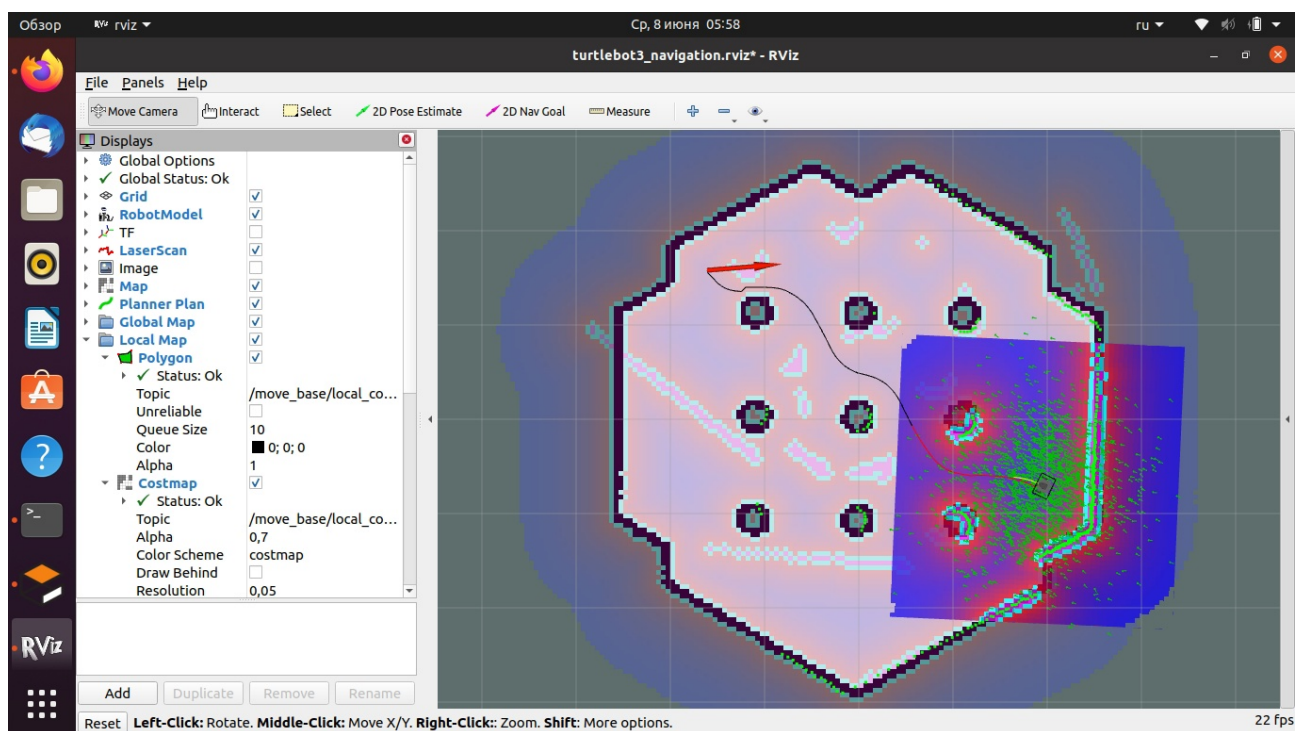


Рис. 4 – Навигация по полученной карте

4 Заключение

Мной были получены практические навыки работы ROS, а также был предложен способ реализации автономное движения и навигации мобильного робота, то есть выполнены цели и задачи практики.

Список литературы

1. emanual.robotis.com: [Электронный ресурс]. - [Б.м], 2022. - URL: <https://emanual.robotis.com/> (дата обращения: 07.06.2022). - Текст: электронный.
2. gazebosim.org: [Электронный ресурс]. - [Б.м], 2022. - URL: <https://gazebosim.org/> (дата обращения: 07.06.2022). - Текст: электронный
3. wiki.ros.org/geometry_msgs: [Электронный ресурс]. - [Б.м], 2022. - URL: http://wiki.ros.org/geometry_msgs/ (дата обращения: 07.06.2022). - Текст: электронный.
4. researchgate.net: [Электронный ресурс]. - [Б.м], 2022. - URL: <https://www.researchgate.net/> (дата обращения: 07.06.2022). - Текст: электронный.
5. jpl.nasa.gov: [Электронный ресурс]. - [Б.м], 2022. - URL: <https://www.jpl.nasa.gov/> (дата обращения: 07.06.2022). - Текст: электронный.