

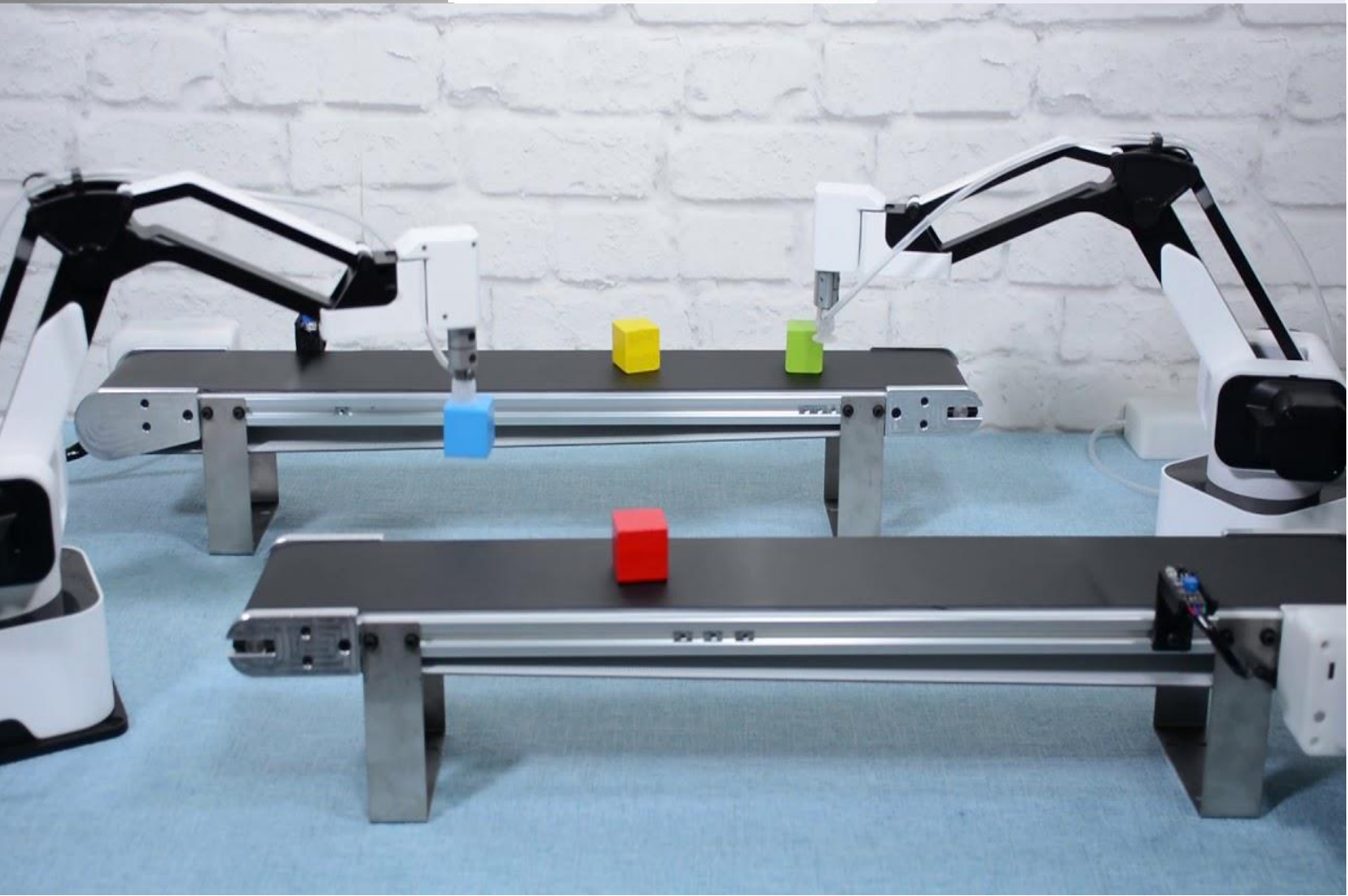
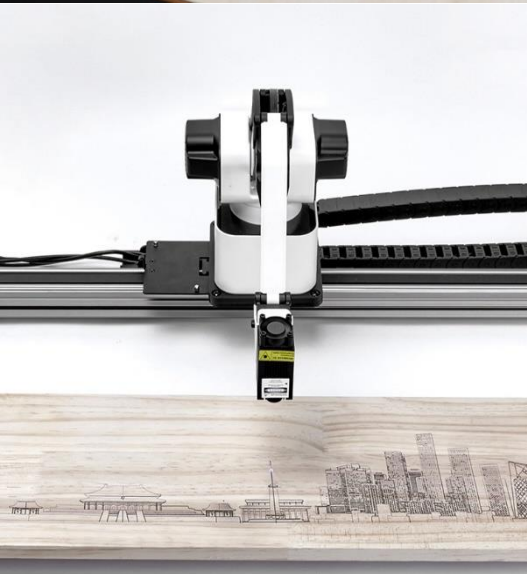


Rotrics DexArm

Руководство пользователя

Инновационный Центр «Dosti»

Алматы, 2021 (20211117-2124)



Автор: *Мольганов Андрей*

Rotrics DexArm. Руководство пользователя / Инновационный Центр «Dosti», 2021. – 50 с.: ил.

Данная книга знакомит читателя с роботом Rotrics DexArm, который помогает обучаться в различных дисциплинах и компетенциях – программирование, робототехника, мехатроника, 3D-печать и электроника.

Издание предназначено для школьников, студентов, разработчиков и преподавателей, желающих обучиться программированию, робототехнике, мехатронике и моделированию, а также обучать других с помощью данной книги.

Благодаря модульной конструкции роботизированная рука Rotrics может легко переключаться между различными функциями, такими как писание и рисование, лазерная гравировка, 3D-печать и сбор, и размещение.

Благодаря простому в использовании программному обеспечению Rotrics предоставляет вам интуитивно понятный и интерактивный способ управления и воплощения ваших идей всего за несколько кликов.

Rotrics также предоставляет опытным разработчикам API на нескольких языках, таких как Python и C++. Вы можете легко управлять роботизированной рукой с помощью программирования SDK и интегрировать ее в свой существующий проект.

ОГЛАВЛЕНИЕ

Введение

1. Глава 1. Руководство пользователя по работе с роботом
 - 1.1. Характеристики робота;
 - 1.2. Настройка робота;
 - 1.3. Дополнительные расширения
 - 1.4. Движение робота в пространстве
 - 1.5. Математическое объяснение операций робота в пространстве
2. Глава 2. Программирование на языке Scratch
 - 2.1.
3. Глава 3. Программирование на языке Python
 - 3.1. Установка программного окружения
 - 3.2. Основы языка Python
 - 3.2.1. Последовательность операторов
 - 3.3. Движение робота
4. Глава 4. Дополнительные материалы
 - 4.1. Приложение №1

ВВЕДЕНИЕ

Робототехника – очень сложная наука, находящаяся на стыке математики, физики и информатики. Для реализации самого простого робота надо знать несколько законов из электротехники, иметь опыт программирования микроконтроллеров и знать математические основы движения робота в пространстве.

Но с помощью Rotrics DexArm занятия робототехникой станут увлекательнее, проще и намного разнообразнее в плане создания творческой атмосферы как для личного (семейного), так и для школьного (студенческого) творчества.

Данный робот снижает порог вхождения в робототехнику, с помощью упрощения некоторых незначительных, но одного из самых трудоемких процессов – программирования микроконтроллера.

Высокоточный манипулятор программируется на языке Scratch, понятным даже для 6 летнего ребенка с помощью простых и удобных визуальных конструкций. Язык Scratch является полностью программируемым, то есть на нем можно реализовать почти все программные теории и методы, которые применяется при разработке серьезного программного обеспечения.

Но самое главное преимущество робота – модульность. Робот был сконструирован с упором на модульность – когда пользователь можно поменять любой модуль на абсолютно другой. Это действительно очень просто и увлекательно!

Данная книга расширяет горизонты применения робота, и использует в своем арсенале такой мощный, гибкий и простой в использовании язык программирования – Python. С помощью данного языка программирования можно автоматизировать робота и придумать самые различные творческие сценарии.

Например – автоматизация производства. Вы никогда не сортировали ящики по цвету? Это просто, круто и что самое главное – доступно для всех. И все это в этой книге.

Удачного пути!

ГЛАВА 1. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ



Рисунок 1.0 – Основные и дополнительные компоненты DexArm, а также сменные модули и насадки

Rotrics DexArm – это универсальный настольный робот-манипулятор для лазерной гравировки, 3D-печати, изобразительного искусства, а также может являться наглядным пособием в процессе обучения студентов и учащихся различными направлениями – программированием, робототехникой, мехатроникой, и электроникой.

Благодаря модульной конструкции роботизированная рука Rotrics может легко переключаться между различными функциями, такими как писание и рисование, лазерная гравировка, 3D-печать и сбор, и размещение.

Благодаря простому в использовании программному обеспечению Rotrics предоставляет вам интуитивно понятный и интерактивный способ управления и воплощения ваших идей всего за несколько кликов.

Rotrics также предоставляет опытным разработчикам API на нескольких языках, таких как Python и C++. Вы можете легко управлять роботизированной рукой с помощью программирования SDK и интегрировать ее в свой существующий проект.

1.1. Характеристики робота

Спецификации робота:

- Точность операций: 0,05 мм;
- Объем упаковки собранного робота: 220 x 155 x 160 мм;
- Размер полностью собранного робота: 175 x 128 x 315 мм;
- Вес: 2,4 кг (без дополнительных модулей – конвейерная лента, рельсовая направляющая и т.д.);

Комплектация:

1. Робот;
2. Модуль «Держатель»;
3. Модуль «Лазер»;
4. Модуль «3D-принтер»;
5. Сенсорный экран;
6. Кабели и сервисные инструкции;

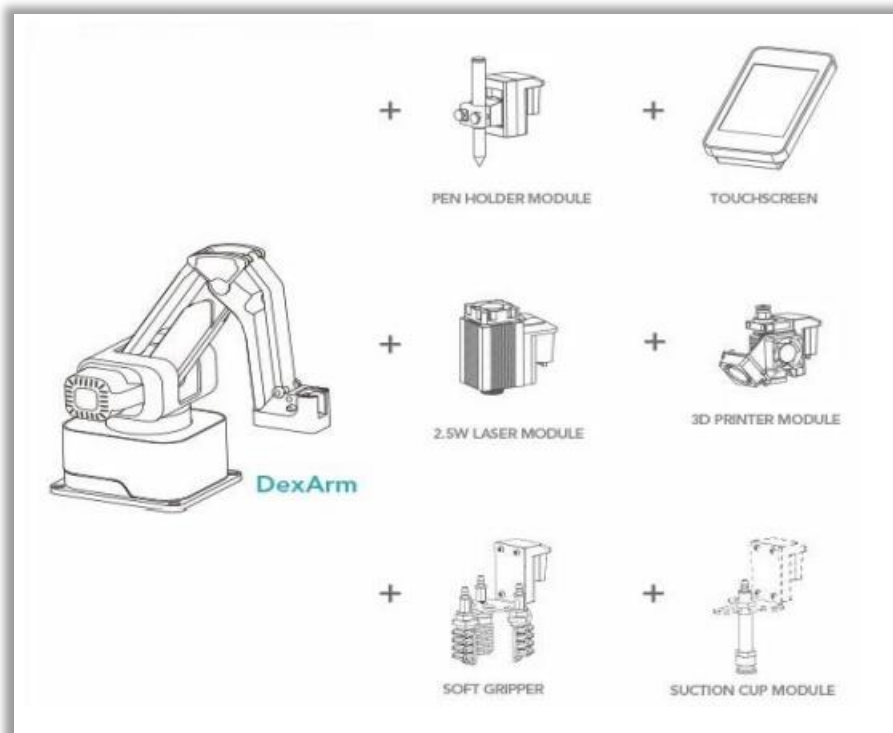


Рисунок 1.1 – Комплектация робота

1.2. Настройка робота:

2. Поместите робот на чистую и ровную поверхность;
3. Скачайте программное обеспечение Rotrics Studio по ссылке - <https://www.rotrics.com/pages/downloads>
4. Подключите робота к питанию с помощью Power Adapter, и к персональному компьютеру с помощью кабеля USB Type C <-> USB Type A



Рисунок 1.2 – (слева направо) Power Adapter, USB Type A <-> USB Type-C, Подключение кабелей к роботу

5. Не используйте двойной кабель USB Type C <-> USB Type C для подключения к персональному компьютеру. Данный кабель используется только в случае подключения дополнительных модулей к роботу:



Рисунок 1.3 – Двойной кабель USB Type C

6. После подключения робота, зайдите в меню «Пуск», и найдите утилиту «Управление компьютером», как показано на рисунке ниже:

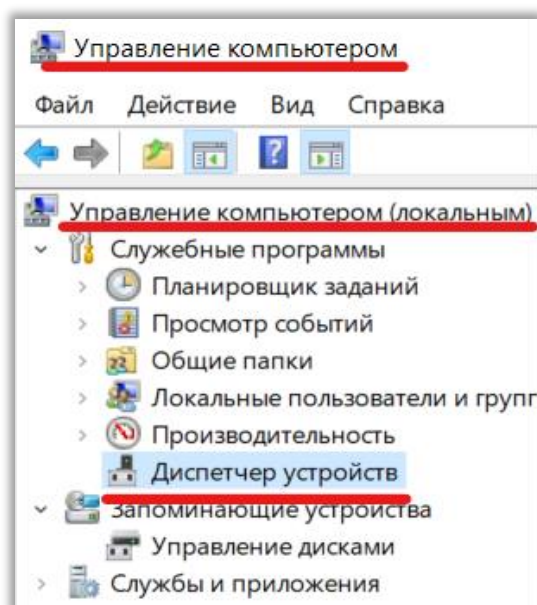


Рисунок 1.4 – Основной вид утилиты «Управление компьютером»

7. В подменю «Диспетчер устройств», нажмите на вкладку «Порты (COM и LPT)». В выпадающем меню будет показан подключенный робот с последовательным портом и номером этого порта. Например: «Устройство с последовательным портом USB (COM8)», где 8 – номер вашего порта:

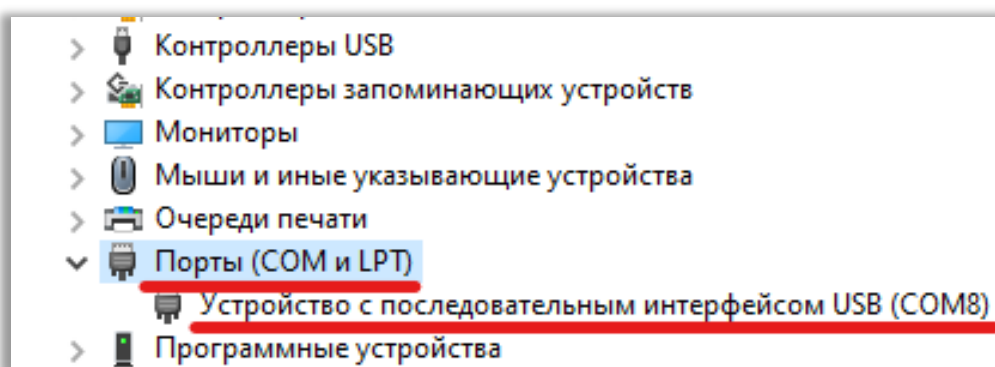


Рисунок 1.5 – Основной вид подменю «Порты (COM и LPT)»

8. Откройте программу Rotrics Studio, и выберите ниже изображенную кнопку на верхней панели:

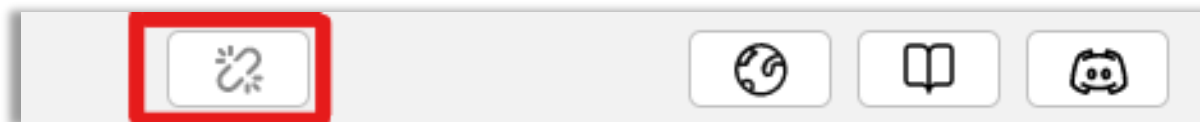


Рисунок 1.6 – Кнопка соединения с роботом

9. В открывшемся диалоговом окне, выберите в поле «Status», доступный в данный момент COM-порт для соединения с роботом:

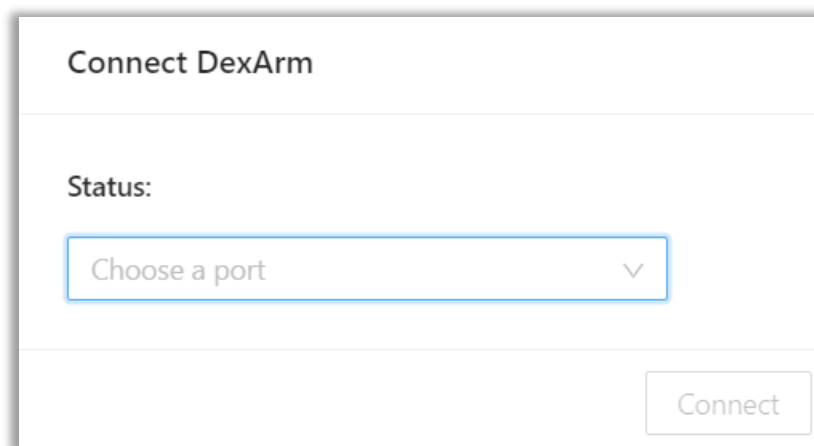


Рисунок 1.7 – Выбор COM-порта для подключения к роботу

10. В открывшемся диалоговом окне выбираем доступный COM-порт для подключения к роботу, нажимаем кнопку «Connect» для подключения:

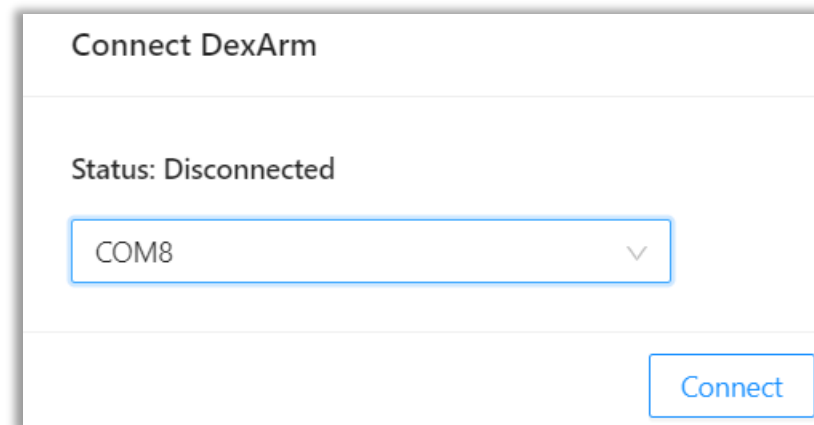


Рисунок 1.8 – Подключение к роботу с помощью COM-порта

11. После подключения к роботу, в диалоговом окне, поле «Status» изменится с «Disconnected» на «Connected»:

Status: Connected

Рисунок 1.9 – Статус подключения

12. Нажмите кнопку «Home» в правой панель управления на вкладке «Control Panel»:

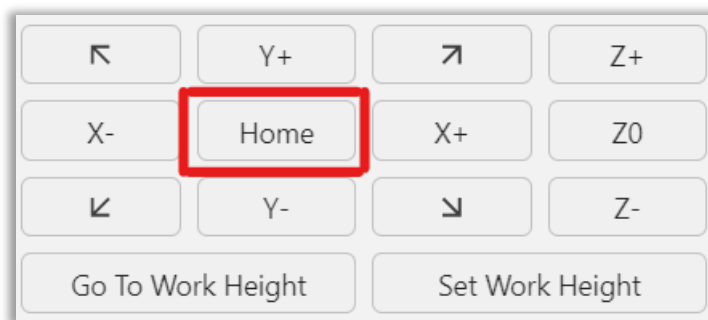


Рисунок 1.10 – Начальная позиция робота

13. Управление роботом осуществляется с помощью кнопок, изображенных на рисунке 1.8:

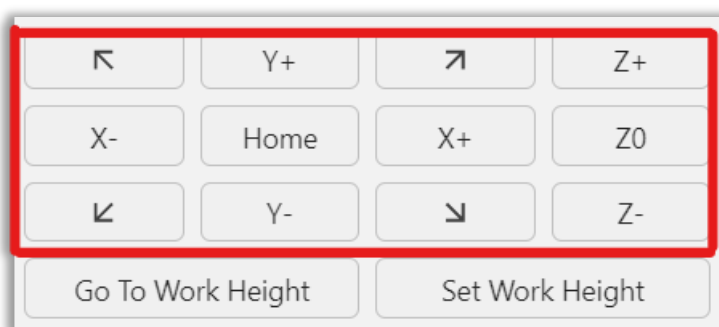


Рисунок 1.11 – Панель управления роботом

14. Также управление роботом возможно с помощью сенсорного экрана, который реализует основные функции управления роботом на расстоянии и без использования персонального компьютера:

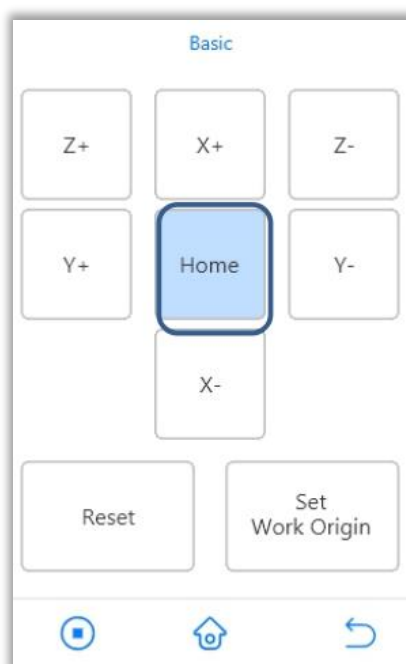


Рисунок 1.12 – Интерфейс сенсорного экрана

1.3. Дополнительные расширения для DexArm:

Для работа также существуют дополнительные расширения, а именно:

№	Название расширения	Количество
1	Конвейерная лента	1
2	Рамка	2
3	Подставка	2
4	Модуль камеры	1
5	Деревянный блок	16
6	Карта производственной линии	1
7	Кабель двигателя	1
8	Шестигранный ключ М4	1
9	Антискользящую ленту	4
10	Шестигранный винт М4	16
11	Скользящая гайка М4	7

Таблица 1.2 – Дополнительные расширения

Сборка конвейерной линии:

1. Приклейте «антискользящую ленту» на заднюю часть подставок как показано на рисунке ниже:



Рисунок 1.13 – Установка подставки для конвейерной ленты

2. Для сборки конвейерной ленты используйте шестигранные, с их помощью прикрутите 6 шестигранных винтов М4 к раме конвейерной ленты с помощью скользящих гаек. Не закрепляйте винты до следующего шага (3):

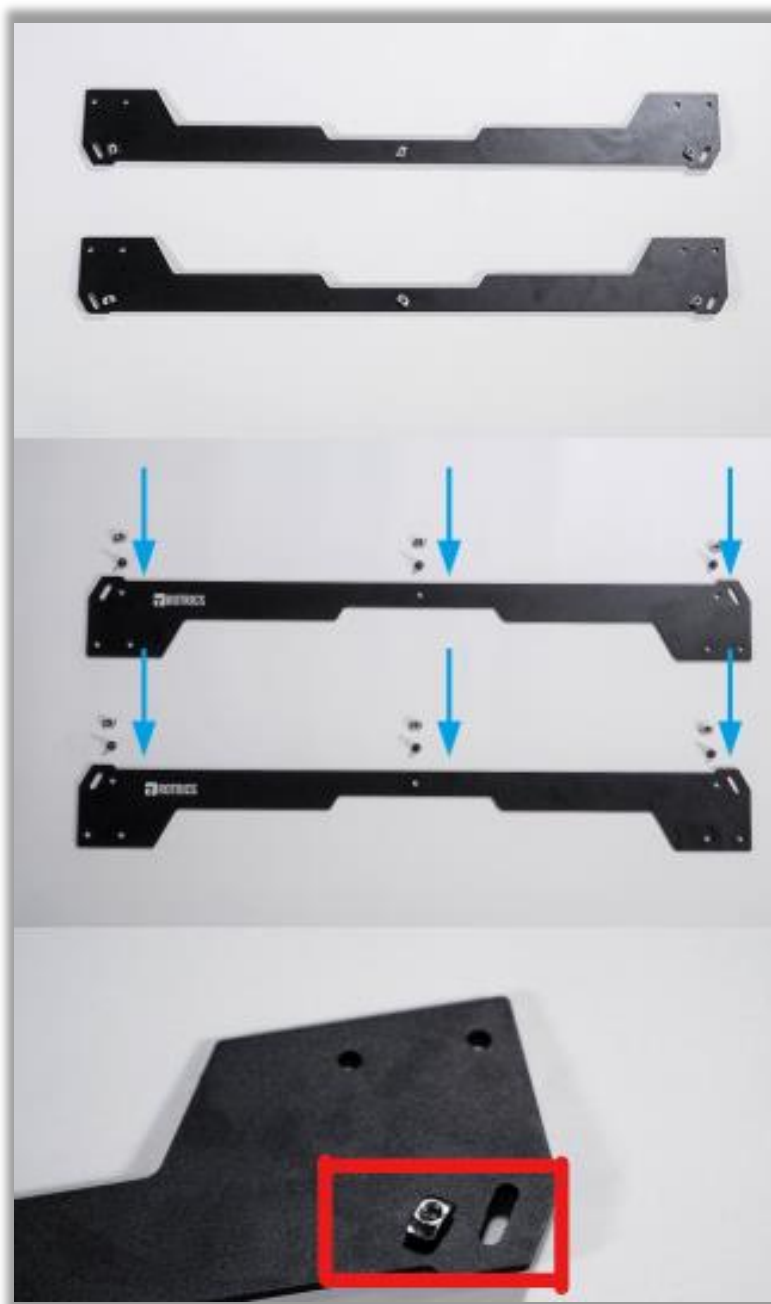


Рисунок 1.14 – Подготовка рамы конвейерной ленты

3. Если вы хотите разместить конвейерную ленту на наклонном столе, вы можете использовать наклонные отверстия, которые находятся выше стационарных отверстий. С их помощью вы сможете быстро и гибко регулировать угол наклона конвейерной ленты:



Рисунок 1.15 – Подготовка рамы для использования под наклоном

4. Возьмите конвейерную ленту и закрепите по бокам рамы:

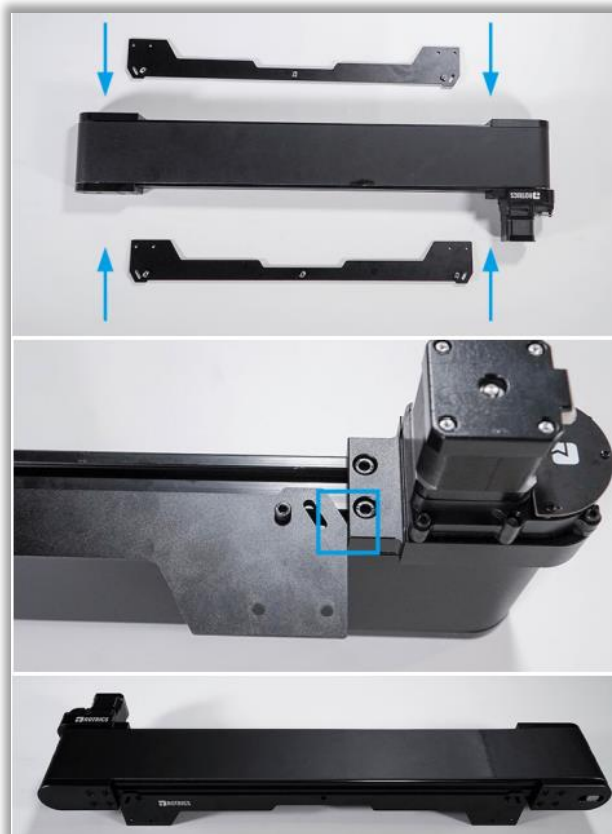


Рисунок 1.16 – Сборка верхней части конвейерной ленты

5. Соберите нижнюю часть конвейерной ленты, а именно ножки-подставки с антискользящей лентой:

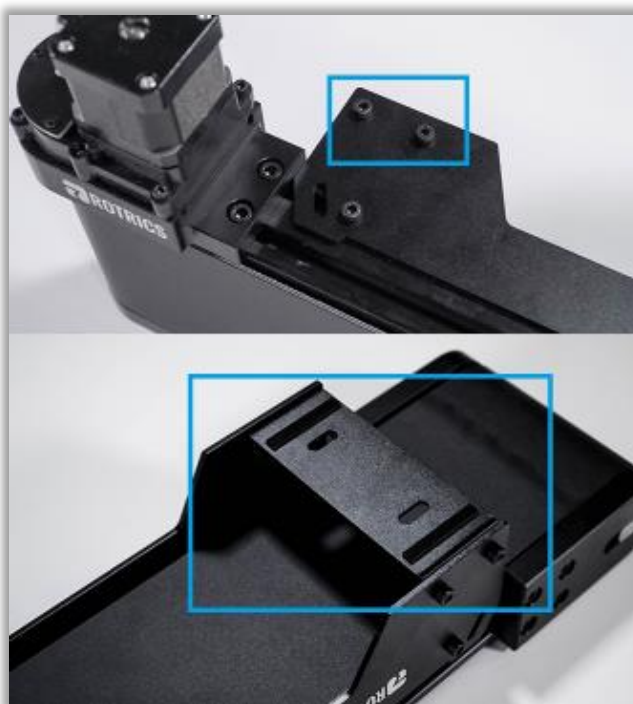


Рисунок 1.17 – Сборка нижней части конвейерной ленты

6. Закрепите модуль камеры с помощью скользящей гайки в скобу, рядом с двигателем конвейерной линии:

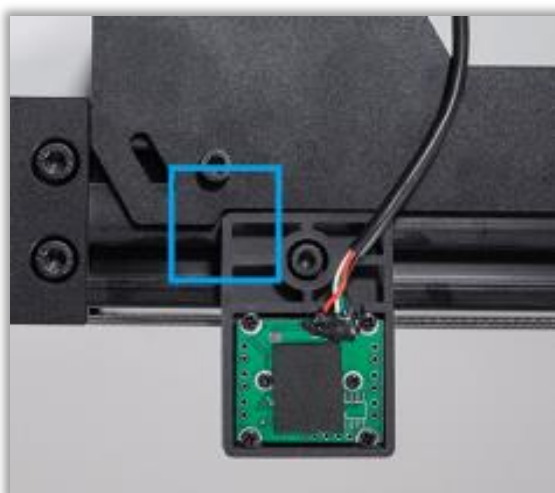


Рисунок 1.18 – Сборка модуля камеры

7. Подключите кабель двигателя с помощью внутри комплектного кабеля:

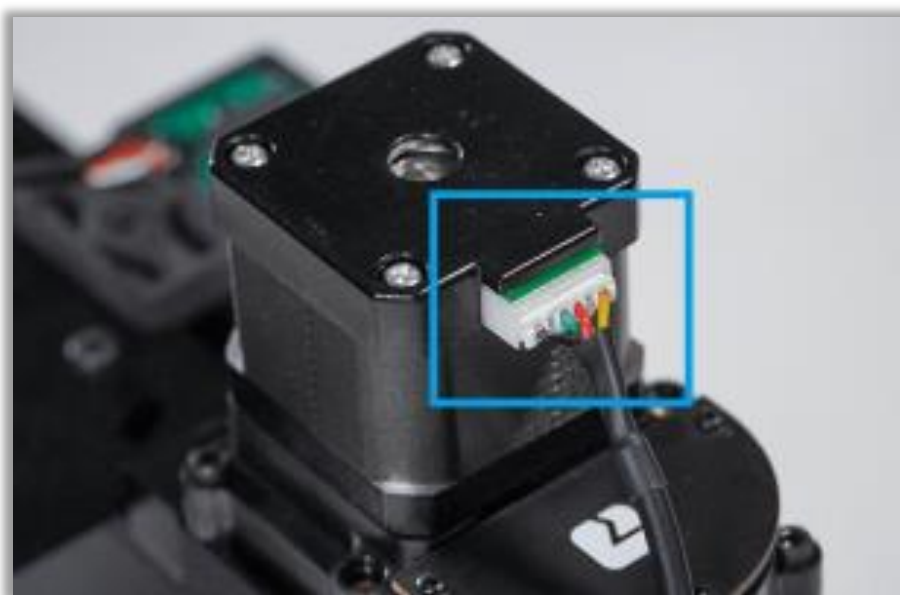


Рисунок 1.19 – Подключение конвейерной линии

8. Если вы выполнили все вышеперечисленные пункты, можно приступить к взаимодействию с конвейерной лентой. Но перед этим, проверьте правильность подключения конвейерной ленты и робота:



Рисунок 1.20 – Подключение конвейерной ленты к роботу

9. В левом вертикальном меню Rotrics Studio, выберите вкладку «Code», далее подменю «Conveyor Belt»:

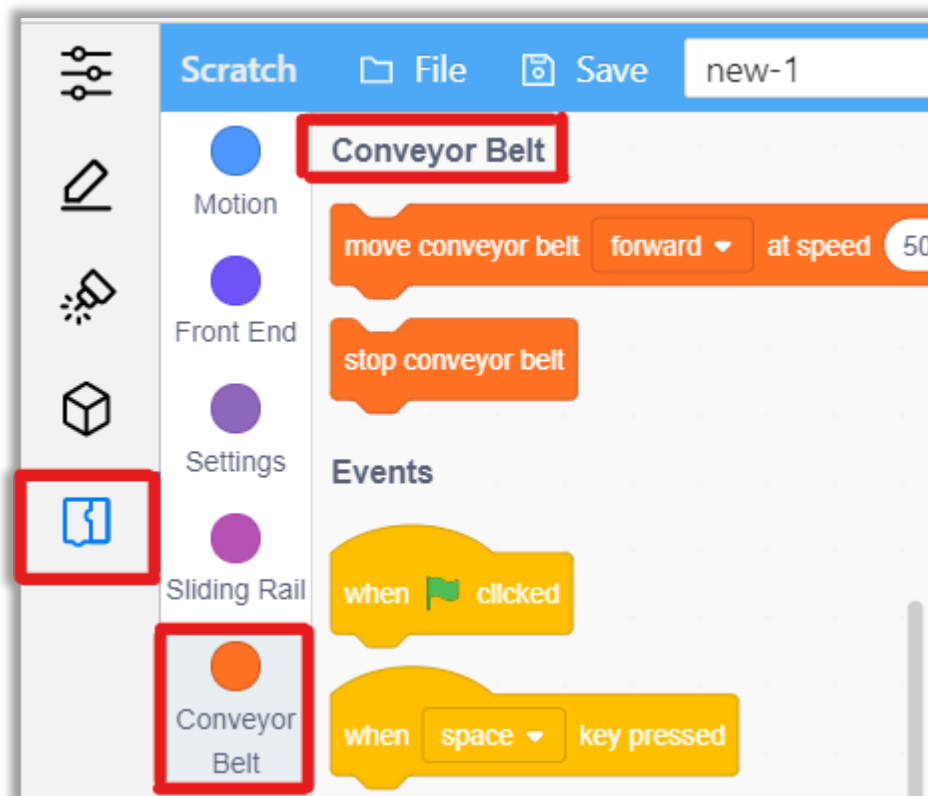


Рисунок 1.21 – Программирование конвейерной ленты с помощью Scratch

1.4. Движение робота в пространстве

Рассмотрим основные правила движения робота в пространстве по координатным осям OX , OY , OZ . На рисунке 1.22, изображены координатные оси, по которым робот с помощью встроенного акселерометра ориентируется в пространстве и выполняет все команды оператора.

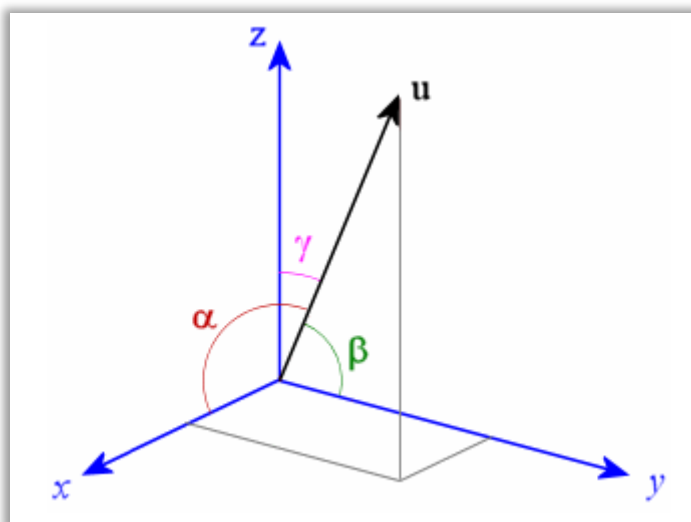


Рисунок 1.22 – Координатные векторы и оси робота

На рисунке 1.22 четко видны следующие векторы и координатные оси:

- Ось X (OX) – ось абсцисс;
- Ось Y (OY) – ось ординат;
- Ось Z (OZ) – ось аппликат (от лат. – applicata – прилегающая);
- Вектор U ;
- Углы α , β и γ .

Вектор U – это такая точка в пространстве, которая задается оператором и передается роботу с помощью трех координатных осей. Например: 100, 300, -75. Данные координаты расшифровываются так:

- 100 – точка на координатной оси X (OX);
- 300 – точка на координатной оси Y (OY);
- -75 – точка на координатной оси Z (OZ).

Основы движения робота в пространстве подчиняются математическим законам, а именно «Прямоугольной системе координат в пространстве».

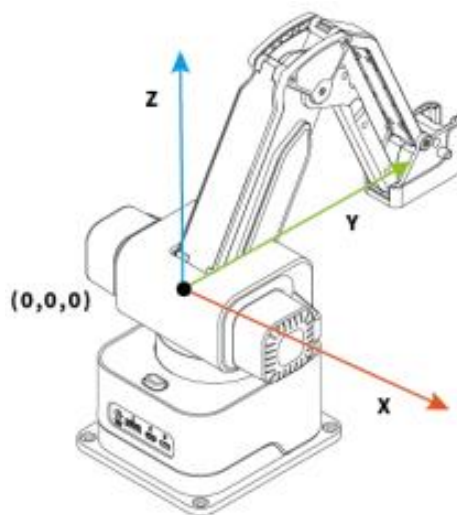


Рисунок 1.23 – Прямоугольная система координат в пространстве которая применяется к роботу

Прямоугольная система координат в пространстве образуется тремя взаимно перпендикулярными осями координат – OX, OY и OZ. Ось координат пересекается в точке O, которая называется началом координат, на каждой оси выбрано положительное и в нашем случае – отрицательное направления. Единицы измерения в наше случае это миллиметры.

Символически – как на письме, так и в электронном виде координаты будут записываться в скобках с указанием заглавной латинской буквы – названием точки в пространстве. Например:

$$A = (x, y, z) \text{ или } A(x, y, z)$$

Если же точек в пространстве очень много (соответственно и координат), то координаты дополнительно обозначают латинскими буквами в нижнем регистре. Например:

$$x_A, y_A, z_A$$

Вектор – направленный отрезок прямой, то есть отрезок, для которого указано, какая из его граничных точек является началом, а какая – концом (Рисунок 1.23).

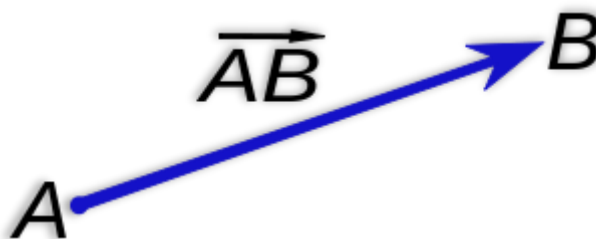


Рисунок 1.23 – Вектор AB

1.5. Математическое объяснение операций робота:

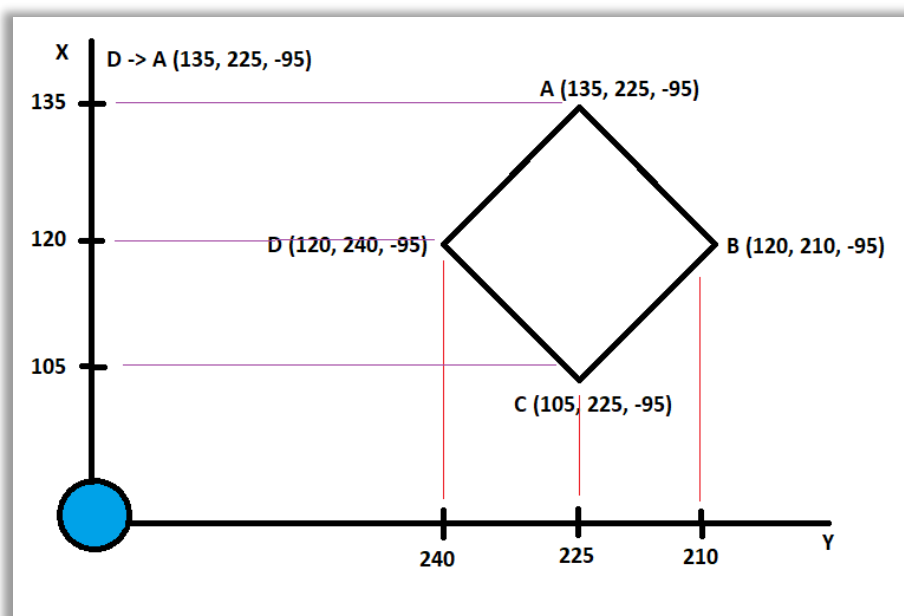


Рисунок 1.24 – Математическое объяснение операций робота в листинге 3.1

1. Для того чтобы нарисовать квадрат или любую другую фигуру на бумаге, дереве или другом материале, следует помнить то, что для модуля «Ручка» установлено минимальная высота, а именно -95 (минус 95), а для модуля «Лазер» минимальная высота составляет -70 (минус 70);
2. Координатные оси распределены следующим образом:

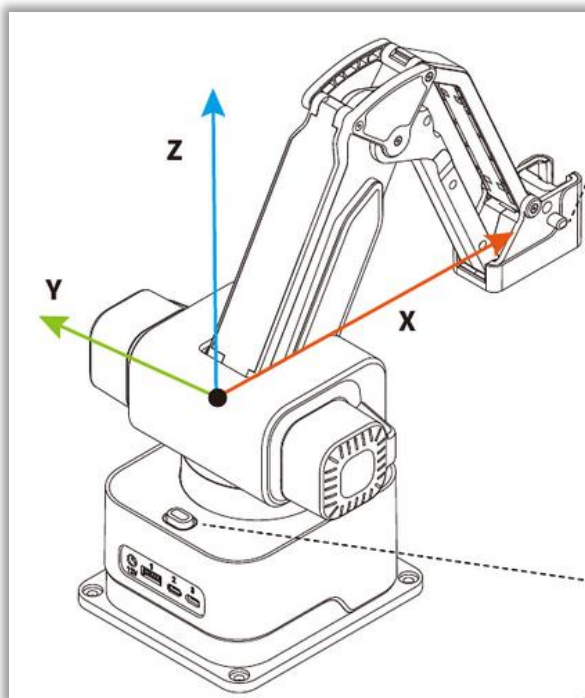


Рисунок 1.25 – Координатные оси робота в 3D-проекции

3. Таким образом, оси X и Y поменялись местами в отличие от математической модели осей, где вверху ось Y, а внизу ось X.
4. При работе с роботом, следует не забывать про конструктивные ограничения, а именно:

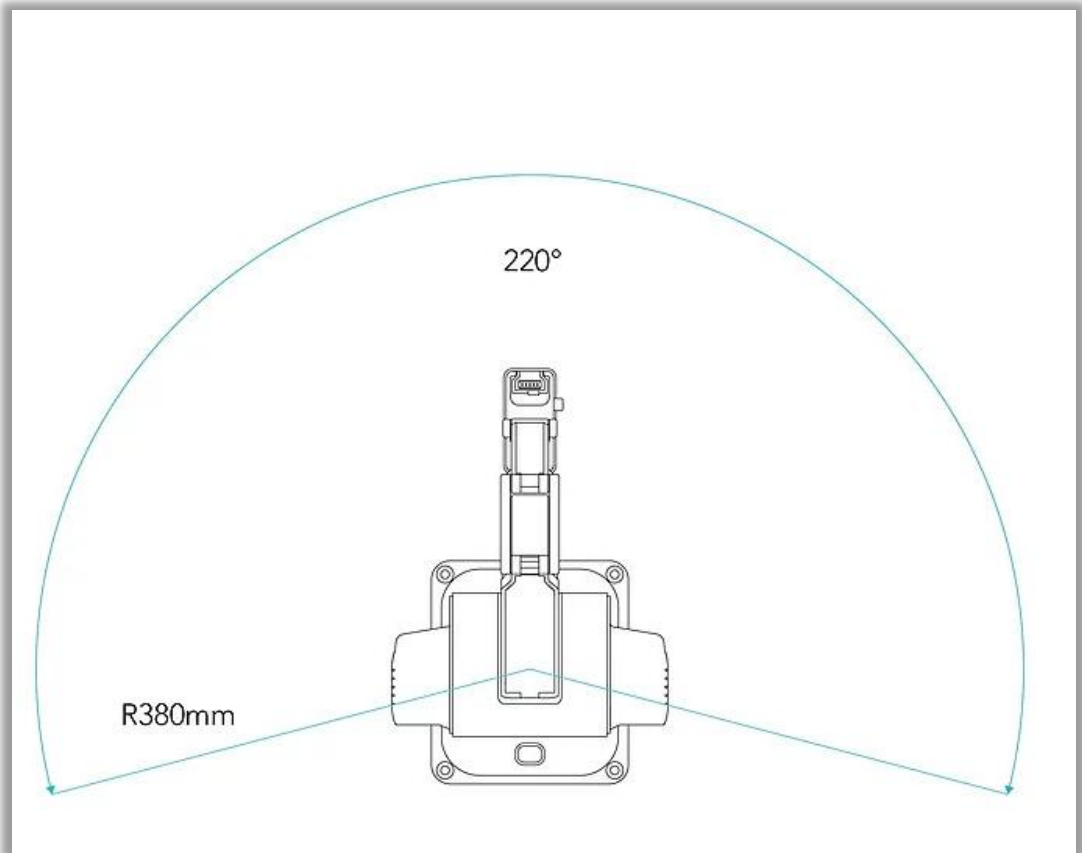


Рисунок 1.26 – Максимальные угловые и радиальные операции доступные для робота

- Максимальный угол для движения по оси X и Y составляет 220 градусов;
 - При этом максимальный радиус для операций составляет 380 мм;
 - Причем, координаты в нашей программе мы принимаем за 1 мм, то есть (100, 300, -75), означает что робот по оси X перейдет в точку 100 (100 мм от начала координат 0; 0; 0), по оси Y в точку 300 (300 мм от начала координат 0; 0; 0) и по оси Z в точку -75 (-75 мм от начала координат 0; 0; 0);
5. При всем этом, мы должны обязательно помнить то, что нельзя выходить за рамки координат!

ГЛАВА 2. ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ SCRATCH

ГЛАВА 3. ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PYTHON

Python – высокоуровневый язык программирования назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества путем структурного облегчения языка за счет упрощения некоторых конструкций – циклы, ветвления, массивы и т.д.



Синтаксис максимально минималистичен за счет упрощения языка. Язык портирован и работает почти на всех известных платформах – от КПК до мейнфреймов. Существуют порты под все известные архитектурные и программные платформы – от Windows и Unix до iOS, Android OS и микроконтроллерные ОС.

3.1. Установка программного окружения

Перед установкой программного окружения, ознакомьтесь со следующими системными требованиями:

1. Операционная система – Windows 7+, MacOS 10.11+, Linux 4.0+;
2. Python – 3.10+;
3. Git – 2.20+;
4. IDE – Visual Studio 2022, Visual Studio Code, PyCharm, Emacs, NeoVim;

Ознакомьтесь с системными характеристиками лабораторного персонального компьютера, на котором были созданы все программы и примеры для данной книги:

1. Операционная система – Windows 10 21H1 v19043.1165;
2. Python – 3.10;
3. Git – 2.31.1.windows.1;
4. IDE – Visual Studio Code 1.62.2;
5. ОЗУ – 8 Гб DDR4;
6. ПЗУ – 1 Тб HDD;

Перейдем к установке программного окружения:

1. После установки необходимого программного обеспечения, установите утилиту `pip` для дополнительной установки необходимых пакетов Python:

```
python get-pip.py
```

2. После установки утилиты `pip`, установите дополнительные пакеты для корректной работы с роботом

```
pip install opencv-python  
pip install pyyaml  
pip install pyserial
```

3. С помощью утилиты `git` клонируйте репозиторий по следующему адресу:

```
git clone https://github.com/AndreM07/dex-arm
```

4. Открываем IDE, во вкладке на верхней панели, щелкаем на меню «Открыть», выбираем папку с нашим клонированным репозиторием;

5. В открывшемся дереве директории, увидим нашу папку с проектом, в котором есть два файла – «`example.py`» и «`pydexarm.py`»

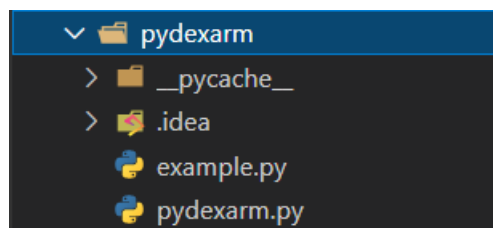


Рисунок 3.1 – Дерево проекта

6. Файл «`example.py`» – содержит своеобразный hello-world для робота, а файл «`pydexarm.py`» – содержит библиотеку-интерфейс для взаимодействия с роботом на языке G-code.

3.2. Основы языка Python

Гвидо ван Россум, создатель языка программирования Python:

«Python - интерпретируемый, объектно-ориентированный высокоуровневый язык программирования с динамической семантикой. Встроенные высокоуровневые структуры данных в сочетании с динамической типизацией и связыванием делают язык привлекательным для быстрой разработки приложений.»

«Кроме того, его можно использовать в качестве сценарного языка для связи программных компонентов. Синтаксис Python прост в изучении, в нем придается особое значение читаемости кода, а это сокращает затраты на сопровождение программных продуктов. Python поддерживает модули и пакеты, поощряя модульность и повторное использование кода. Интерпретатор Python и большая стандартная библиотека доступны бесплатно в виде исходных и исполняемых кодов для всех основных платформ и могут свободно распространяться.»

Программа на языке Python может состоять из одного или нескольких модулей. Каждый модуль представляет собой текстовый файл в кодировке, совместимой с 7-битной кодировкой ASCII.

3.2.1. Последовательность операторов

Последовательные действия описываются последовательными строками программы. В программах важны отступы

3.3. Движение робота

Рассмотрим листинг 3.2.1, в котором показаны основные команды управления роботом с помощью языка программирования Python:

Листинг 3.2.1

```
from pydexarm import Dexarm
dexarm = Dexarm("COM8")
dexarm.go_home()
dexarm.move_to(135, 225, -95) # A
dexarm.move_to(120, 210, -95) # B
dexarm.move_to(105, 225, -95) # C
dexarm.move_to(120, 240, -95) # D
dexarm.move_to(135, 225, -95) # D to A
dexarm.go_home()
```

Рассмотрим подробнее листинг 3.2.1:

- **from pydexarm import Dexarm** - включает в основной код библиотеку-интерфейс для общения с роботом при помощи связи между Python и G-code;
- **dexarm = Dexarm("COM8")** - отправляет скомпилированный код в робот на указанный COM-порт, а именно COM8
- **dexarm.go_home()** - дает команду роботу вернуться в начальное или исходное положение. Функция не требует аргументов, так как робот настроен на начальное положение в (0; 0; 0)
- **dexarm.move_to(135, 225, -95)** - определяет координаты по трем осям, и отправляет роботу команду для перемещения манипулятор робота в данную точку, а именно – по оси X – 135; по оси Y – 225; по оси Z – -95.

ГЛАВА 4. ДОПОЛНИТЕЛЬНЫЕ МАТЕРИАЛЫ

Приложение №1 – Полное описание библиотек-интерфейса:

Группа команд	Команда	Описание
Общие	<code>def __init__()</code>	Инициализация
	<code>def _send_cmd()</code>	Отправка команды
Движение	<code>def go_home(self)</code>	Перемещение в начальную позицию (X = 0, Y = 300, Z = 0)
	<code>def set_workorigin()</code>	Сброс рабочей высоты (Z = 0)
	<code>def set_acceleration()</code>	Включение акселерометра
	<code>def move_to()</code>	Перемещение в заданную позицию
	<code>def fast_move_to</code>	Быстрое перемещение в заданную позицию
	<code>def get_current_position()</code>	Вычисление текущей позиции
Задержки	<code>def dealy_ms()</code>	Задержка в миллисекундах
	<code>def dealy_s()</code>	Задержка в секундах
Модули	<code>def set_module_kind()</code>	Инициализация модуля
	<code>def get_module_kind()</code>	Распознавание модуля
Щупальца	<code>def soft_gripper_pick()</code>	Сжать щупальцу
	<code>def soft_gripper_place()</code>	Разжать щупальцу
	<code>def soft_gripper_nature()</code>	Возврат в исходное состояние или разжатие щупальцы
	<code>def soft_gripper_stop()</code>	Остановка подачи воздуха
Присоска	<code>def air_picker_pick()</code>	Сжать присоску
	<code>def air_picker_place</code>	Разжать присоску
	<code>def air_picker_nature</code>	Возврат в исходное состояние или разжатие присоски
	<code>def air_picker_stop</code>	Остановка подачи воздуха
Лазер	<code>def laser_on()</code>	Включить лазер
	<code>def laser_off()</code>	Выключить лазер
Конвейер	<code>def conveyor_belt_forward</code>	Прокрутить конвейер вперед
	<code>def conveyor_belt_backward</code>	Прокрутить конвейер назад
	<code>def conveyor_belt_stop</code>	Остановить конвейер

Приложение № 2 – Список литературы:

- [1] - <https://itlectorium.com/video/Dexarm/rotrics-firststart>
- [2] - <https://itlectorium.com/competition/kidskills>