Санкт-Петербургский политехнический университет Петра великого

Институт машиностроения, материалов и транспорта

Высшая школа автоматизации и робототехники

**Курсовой проект**

По дисциплине «Объектно-ориентированное программирование»

«Программирование промышленного манипулятора KUKA с использованием захвата»

Студент Жигунов В.Н.

Группа 3331506/20101

Преподаватель Ананьевский М.С.

Санкт-Петербург

2025

**Технической задание**

Необходимо произвести подключение Манипулятора KUKA и захвата, изучить основные принципы управления промышленным манипулятором с захватом, с написанием программы для построения башни из 3 кубиков в качестве итогового продукта.

**Введение**

Современные промышленные роботы становятся неотъемлемой частью автоматизированных производственных процессов. Они повышают точность и скорость производства, а также удешевляют и упрощают его. Одним из популярных промышленных роботов является манипулятор KUKA, который широко применяется в различных отраслях промышленности.

**Теоретические сведения**

У данной модели существует 2 типа управления в ручном режиме и с помощью написанной заранее программы.

**Ручное управление**

Робот имеет возможность управляться в режиме реального времени с помощью KUKA smartPAD представленного на рисунках 1 и 2.



Рисунок 1 – Передняя сторона пульта управления KUKA smartPAD.



Рисунок 2 – Задняя сторона пульта управления KUKA smartPAD.

В ручном режиме с помощью пульта можно управлять в отдельности каждым звеном или управлять перемещением по координатам конечной точки. Также перемещение можно осуществлять в ралзичных системах координат, что очень удобно.

1. BASE – система координат, связанная с местом крепления робота.
2. WORLD – переназначаемая система, которую можно поставить в любое место для упрощения написания программы
3. TOOL – система, связанная с инструментом.

Каждая из этих систем координат может быть изменена и сохранена для дальнейшей работы.

В обоих случаях для перемещения используется кнопки “+” и “–“. Этот блок кнопок либо управляет каждым из звеньев, либо меняет положение инструмента по координатам X, Y, Z с возможностью вращения осей А, В, С. Также необходимо чтобы кнопка 3 на рисунке 1 была отжата, а одна из кнопок 3 или 5 на рисунке 2 нажата.

**Программное управление**

Кроме ручного управления можно осуществлять программное управление с помощью пульта иди компьютера. В данной работе мы рассмотрим управление с пульта.

Все роботы KUKA используют KRL – KUKA Robot Language. Все программы для управления располагают в следующем каталоге, представленном на рисунке 3.

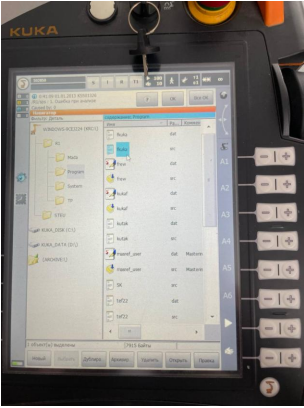


Рисунок 3 – Расположение программ

Рассмотрим основные команды, используемые для управления манипулятором:

1. **Движение**

* PTP

Команда РТР перемещает робота в заданную точку по кратчайшему пути. Это движение обычно выполняется по траектории, которая не является прямой линией.

Аргументы:

X, Y, Z – координаты целевой точки в пространстве.

А, В, С – углы ориентации инструмента

* LIN

Перемещает робота по прямой линии в заданную точку.

Аргументы:

X, Y, Z – координаты целевой точки в пространстве.

А, В, С – углы ориентации инструмента

* CIRC

Перемещает робота по дуге окружности через промежуточную точку.

Аргументы:

X, Y, Z – координаты целевой точки в пространстве.

А, В, С – углы ориентации инструмента

1. **Управление программой**

* END

Завершает выполнение программы.

* WAIT

Останавливает выполнение программы на заданное время или до выполнения условия.

Аргументы:

SEC – время ожидания в секундах

* LOOP

Создает бесконечный цикл.

1. **Управление вводом/выводом**

* OUT

Устанавливает значение выхода.

Аргументы:

Номер выхода

TRUE или FALSE

* IN

Читает значение входа

Аргументы:

Номер выхода

TRUE или FALSE

1. **Управление переменными**

* DECL
* SET

1. **Управление подпрограммами**

* DEF

Объявляет подпрограмму.

* CALL

Вызывает подпрограмму.

1. **Управление условиями**

* IF

Условный оператор.

* SWITCH

Множественный выбор.

1. **Управление инструментом**

* TOOL

Выбирает инструмент.

* BASE

Выбирает базовую систему координат.

1. **Управление скоростью и ускорением**

* VEL.CP

Устанавливает скорость движения.

**Ход работы**

Для начала переходим в режим «Эксперт»

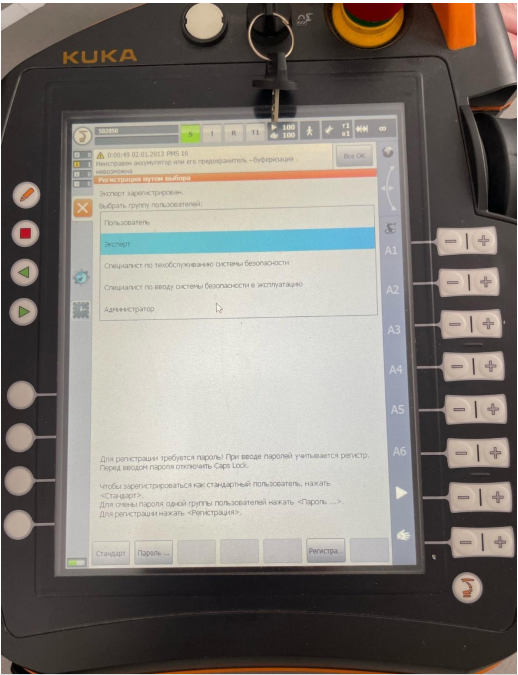


Рисунок 4 – Режим эксперта

Производим калибровку инструмента, что показано на рисунке 5

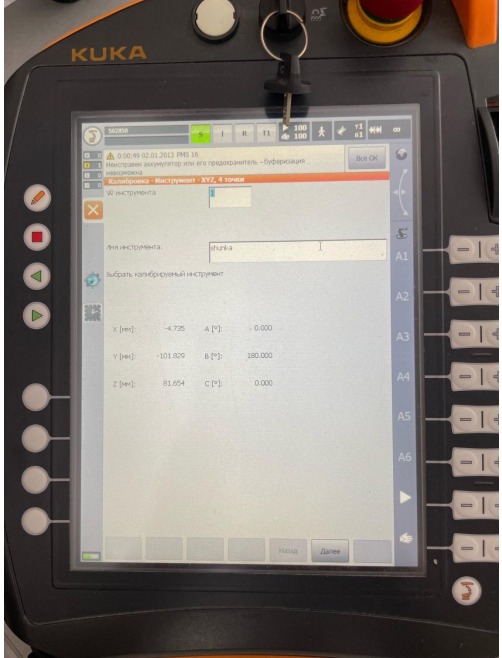


Рисунок 5 – Калибровка инструмента

Создаем базу для удобства управления

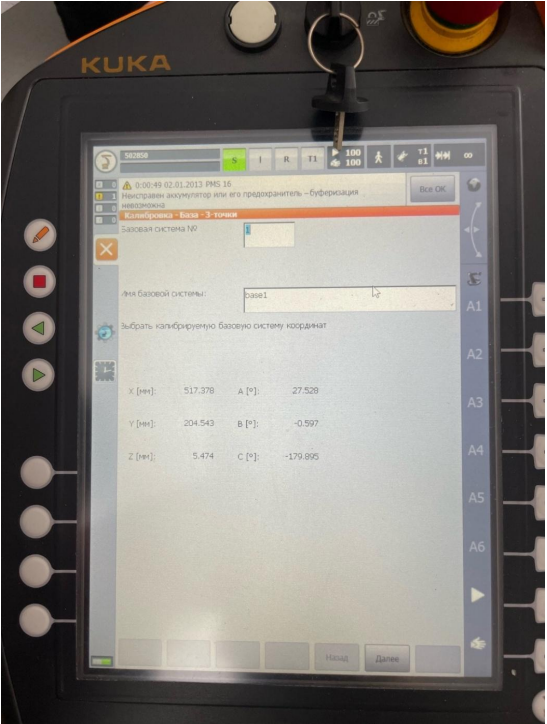


Рисунок 6 – Создание базы

Далее размещаем 3 кубика, которые будет перемещать манипулятор.

Место размещения кубиков должно быть строго определено, так как программа устройства не имеет СТЗ, и манипулятор по программе ожидает, что кубики будут в конкретных точках. Кубики располагаем на углах границ нашей рабочей зоны.

Мы разместили кубики, как показано на рисунке 7

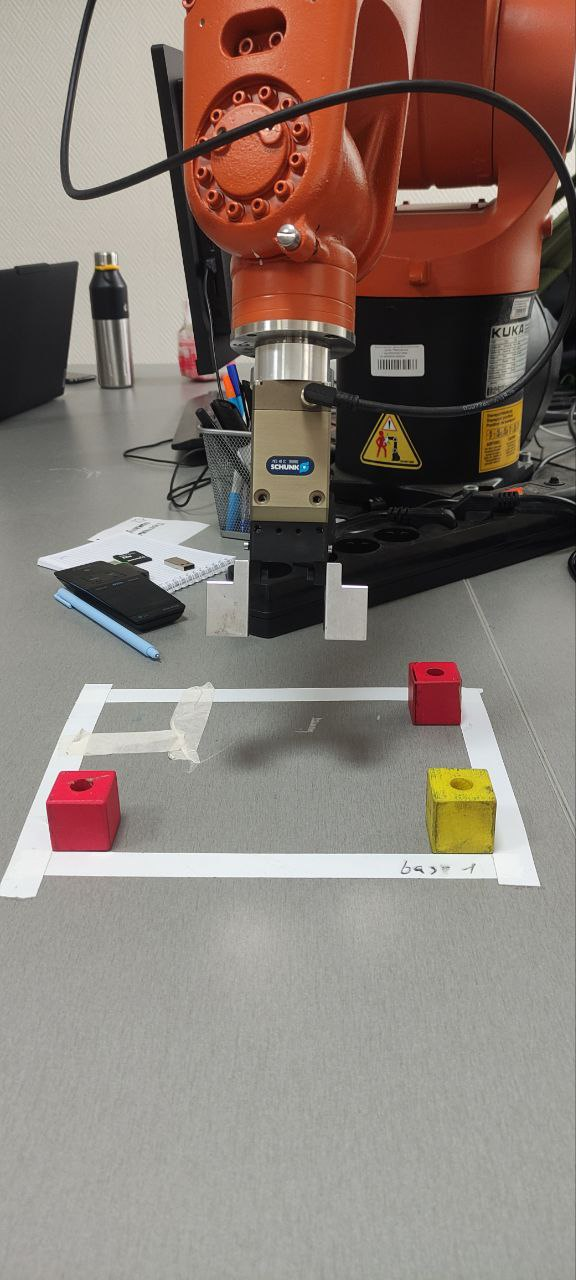


Рисунок 7 – Начальное положение кубиков.

**Программный код**

Пишем необходимую программу с помощью пульта управления для перемещения трех кубиков с построением из них башни. Код программы:

**1 DEF kuka( )**

**2 INI**

**3**

**4 PTP HOME Vel=50 % DEFAULT**

**5**

**6 WAIT Time=1 sec**

**7 OUT 3 ‘razgim’ State=TRUE CONT**

**8 WAIT Time=2 sec**

**9 OUT 3 ‘razgim’ State=FALSE CONT**

**10 WAIT Time=3 sec**

**11 PTP P4 Vel=80 % PDAT2 Tool[1]:shunka Base[1]:base1 CD**

**12 PTP P7 Vel=80 % PDAT3 Tool[1]:shunka Base[1]:base1 CD**

**13 PTP P8 Vel=80 % PDAT4 Tool[1]:shunka Base[1]:base1 CD**

**14 WAIT Time=2 sec**

**15 OUT 4 ‘zagim’ State=TRUE CONT**

**16 WAIT Time=2 sec**

**17 OUT 4 ‘zagim’ State=FALSE CONT**

**18 PTP P7 Vel=80 % PDAT5 Tool[1]:shunka Base[1]:base1 CD**

**19 PTP P4 Vel=80 % PDAT6 Tool[1]:shunka Base[1]:base1 CD**

**20 PTP P11 Vel=80 % PDAT7 Tool[1]:shunka Base[1]:base1 CD**

**21 WAIT Time=1 sec**

**22 OUT 3 ‘razgim’ State=TRUE CONT**

**23 WAIT Time=2 sec**

**24 OUT 3 ‘razgim’ State=FALSE CONT**

**25 PTP P4 Vel=80 % PDAT8 Tool[1]:shunka Base[1]:base1 CD**

**26 PTP P13 Vel=80 % PDAT10 Tool[1]:shunka Base[1]:base1 CD**

**27 PTP P12 Vel=80 % PDAT9 Tool[1]:shunka Base[1]:base1 CD**

**28 WAIT Time=1 sec**

**29 OUT 4 ‘zagim’ State=TRUE CONT**

**30 WAIT Time=2 sec**

**31 OUT 4 ‘zagim’ State=FALSE CONT**

**32 PTP P13 Vel=80 % PDAT11 Tool[1]:shunka Base[1]:base1 CD**

**33 PTP P4 Vel=80 % PDAT12 Tool[1]:shunka Base[1]:base1 CD**

**34 PTP P14 Vel=80 % PDAT13 Tool[1]:shunka Base[1]:base1 CD**

**35 WAIT Time=1 sec**

**36 OUT 3 ‘razgim’ State=TRUE CONT**

**37 WAIT Time=2 sec**

**38 OUT 3 ‘razgim’ State=FALSE CONT**

**39 PTP P4 Vel=80 % PDAT14 Tool[1]:shunka Base[1]:base1 CD**

**40 PTP P16 Vel=80 % PDAT16 Tool[1]:shunka Base[1]:base1 CD**

**41 PTP P15 Vel=80 % PDAT15 Tool[1]:shunka Base[1]:base1 CD**

**42 WAIT Time=1 sec**

**43 OUT 4 ‘zagim’ State=TRUE CONT**

**44 WAIT Time=2 sec**

**45 OUT 4 ‘zagim’ State=FALSE CONT**

**46 PTP P16 Vel=80 % PDAT17 Tool[1]:shunka Base[1]:base1 CD**

**47 PTP P4 Vel=80 % PDAT18 Tool[1]:shunka Base[1]:base1 CD**

**48 PTP P17 Vel=80 % PDAT19 Tool[1]:shunka Base[1]:base1 CD**

**49 WAIT Time=1 sec**

**50 OUT 3 ‘razgim’ State=TRUE CONT**

**51 WAIT Time=2 sec**

**52 OUT 3 ‘razgim’ State=FALSE CONT**

**53 PTP P4 Vel=80 % PDAT20 Tool[1]:shunka Base[1]:base1 CD**

**54 PTP HOME Vel=50 % DEFAULT**

**55**

**56 END**

**Результат работы программы**

В результате проверки программы она успешно завершилась и выполнила поставленные задачи, что можно увидеть на рисунке 8

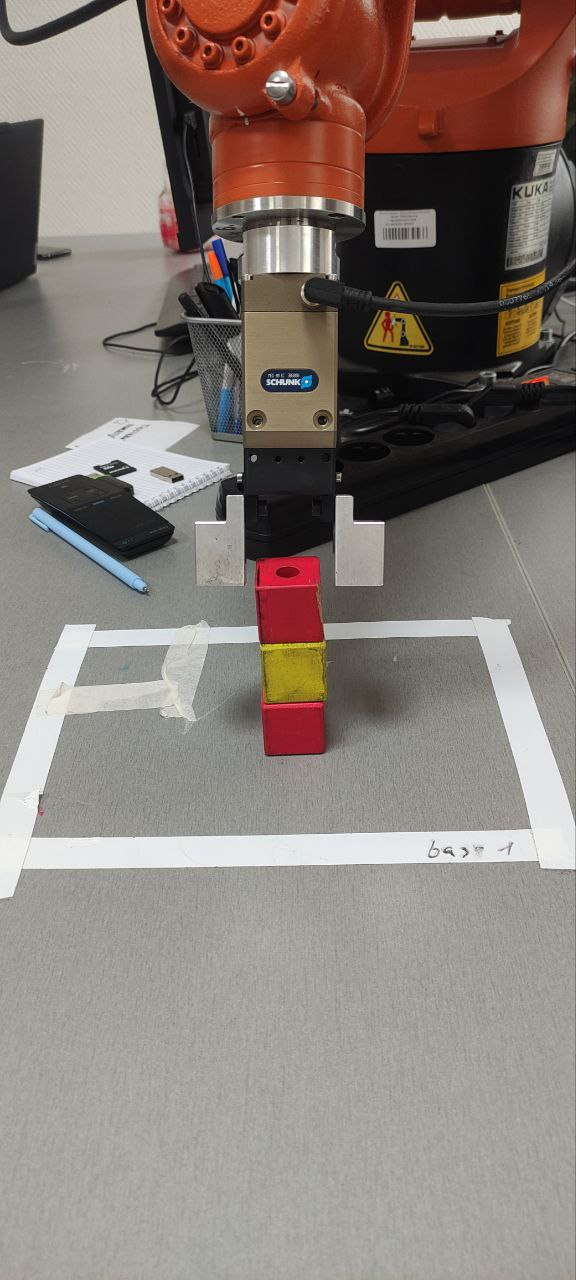


Рисунок 8 – Конечный результат программы.

**Заключение**

В ходе выполнения курсового проекта была проведена работа по исследованию манипулятора KUKA, включая изучение его конструкции и основных характеристик.

Результатом курсового проекта стало создание программы, для построение башни из кубиков.