

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«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»**

Пояснительная записка

Выполнили
студенты гр.
3331506/20401

(подпись)

Базельцев А. А.
Пантелеймонов Б. Б.

Работу принял

(подпись)

Ананьевский М.С.

Санкт-Петербург
2025 г.

Оглавление

Техническое задание.....	3
1. Введение.....	3
2. Теоретические сведения	3
2.1 Ручное управление	3
2.2 Программное управление.....	4
3. Подготовительные работы	7
3.1 Юстировка	7
3.1.1 Зачем нужна юстировка?.....	7
3.1.2 Комплект для юстировки	7
3.1.3 Подготовка к юстировке	8
3.1.4 Порядок юстировки	8
3.2 Калибровка инструмента робота KUKA	10
3.2.1 Зачем нужна калибровка?	10
3.2.2 Комплект калибровки.....	10
3.2.3 Подготовка к калибровке	10
3.2.4 Порядок калибровки	11
3.3 Калибровка базы	12
3.3.1 Зачем нужна калибровка базы?	12
3.3.2 Методы калибровки базы.....	13
3.3.3 Комплект калибровки.....	13
3.3.4 Подготовка к калибровке	13
3.3.5 Порядок калибровки методом трех точек	13
4. Ход работы.....	15
5. Программный код	19
6. Результаты работы программы.....	22
7. Заключение	23
8. Список литературы	24

Техническое задание

Необходимо произвести подключение Манипулятора KUKA и захвата, произвести наладку оборудования, а также изучить основные принципы управления промышленным манипулятором с захватом, с написанием программы для построения башни из 3 кубиков в качестве теста.

1. Введение

Современные промышленные роботы становятся неотъемлемой частью автоматизированных производственных процессов. Они обеспечивают высокую точность и скорость выполнения операций, что позволяет значительно повысить производительность труда и снизить затраты на изготовление продукции. Одним из наиболее популярных и востребованных решений на рынке промышленных роботов является манипулятор KUKA, который широко применяется в различных отраслях промышленности — от автомобильной сборки до прецизионной обработки деталей.

2. Теоретические сведения

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

2.1 Ручное управление

Робот имеет возможность управляться в режиме реального времени с помощью KUKA smartPAD представленного на рисунках 2.1.1 и 2.1.2



Рисунок 2.1.1 - Передняя сторона пульта управления KUKA smartPAD.



Рисунок 1.1.2 - 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 различных системах координат (WORLD, BASE, TOOL), 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 используются кнопки “+” и “-” (обозначение 5 на рисунке 2.1.1). Этот блок кнопок либо управляет каждым из звеньев, либо меняет положение инструмента по координатам X, Y, Z с возможностью вращения осей A, B, C. Также необходимо чтобы кнопка 3 на рисунке 2.1.1 была отжата, а одна из кнопок 3 или 5 на рисунке 2.1.2 нажата.

2.2 Программное управление

Кроме ручного управления можно осуществлять программное управление с помощью пульта или 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 2.2.1

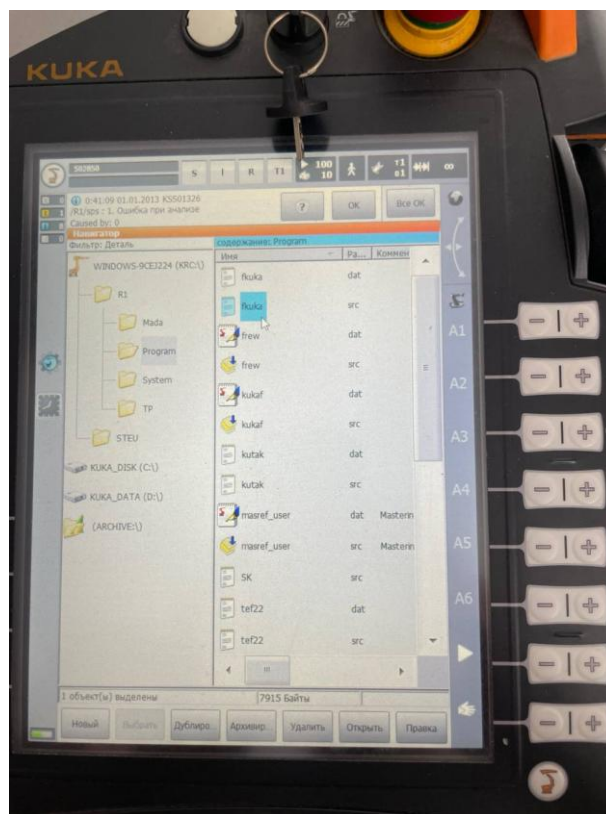


Рисунок 2.2.1 - Расположение программ

Рассмотрим основные команды, используемые для управления манипулятором:

1. Движение

- *PTP*

Команда PTP 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.

A, B, C — углы ориентации инструмента

- *LIN*

Команда LIN перемещает робота по прямой линии в заданную точку.

Аргументы:

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

A, B, C — углы ориентации инструмента.

- *CIRC*

Команда CIRC перемещает робота по дуге окружности через промежуточную точку.

Аргументы:

Первая точка — промежуточная точка на дуге.

Вторая точка — конечная точка.

2. 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.

3. Управление вводом/выводом

- *OUT*

Устанавливает значение выхода.

Аргументы:

Номер выхода.

TRUE или FALSE.

- *IN*

Читает значение входа.

Аргументы:

Номер входа.

TRUE или FALSE.

4. Управление переменными

- *DECL*

Объявляет переменную.

Аргументы:

Тип переменной (например, INT, REAL, BOOL).

Имя переменной.

- *SET*

Присваивает значение переменной.

5. Управление подпрограммами

- *DEF*

Определяет подпрограмму.

- *CALL*

Вызывает подпрограмму.

6. Управление условиями

- *IF*

Условный оператор.

- *SWITCH*

Множественный выбор.

7. Управление инструментом

- *TOOL*

Выбирает инструмент.

- *BASE*

Выбирает базовую систему координат.

8. Управление скоростью и ускорением

- *VEL.CP*

Устанавливает скорость движения.

3. Подготовительные работы

3.1 Юстировка

Юстировка робота KUKA — это обязательная процедура для обеспечения точности и стабильности работы манипулятора. Она позволяет устранить различия между механическим и электрическим положениями осей и корректирует возможные смещения, вызванные нагрузкой.

3.1.1 Зачем нужна юстировка?

Юстировка необходима в следующих случаях:

1. При первом вводе робота в эксплуатацию.
2. После технического обслуживания, связанного с компонентами определения положения (например, двигатель с синус-косинусным преобразователем).
3. После механического ремонта или столкновений.
4. После перемещения осей без системы управления (например, вручную).
5. При нарушении стартового положения.

3.1.2 Комплект для юстировки

Для юстировки робота KUKA используются следующие устройства:

1. **Юстировочное устройство EMD (Electronic Mastering Device)** - для настройки механических нулевых положений.
2. **Адаптерный кабель (KR C2)** - для подключения устройства к системе управления.

Юстировочный комплект представлен на рисунке 3.1.2.1.



Рисунок 3.1.2.1 – Юстировочный комплект

3.1.3 Подготовка к юстировке

1. **Режим работы:** активировать режим T1 (ручной режим с пониженной скоростью).
2. **Подключение устройства EMD:** подсоединить кабель EtherCAT к разъему X32.
3. **Проверка нулевого положения:** ось A6 необходимо привести в юстировочное положение по нанесенным меткам.

3.1.4 Порядок юстировки

1. Деюстировка осей:

- Перейти в меню: **Пуск в эксплуатацию> Юстировка> EMD> С коррекцией нагрузки> Первичная юстировка.**
- Деюстировать все оси перед началом настройки.

2. Подвод осей к предъюстировочному положению:

- На рисунке 3.1.4.1 все оси находятся в положении, соответствующем механическим нулевым точкам:
 - A1: 0°
 - A2: -90°
 - A3: +90°
 - A4: 0°
 - A5: 0°

- A6: 0°.

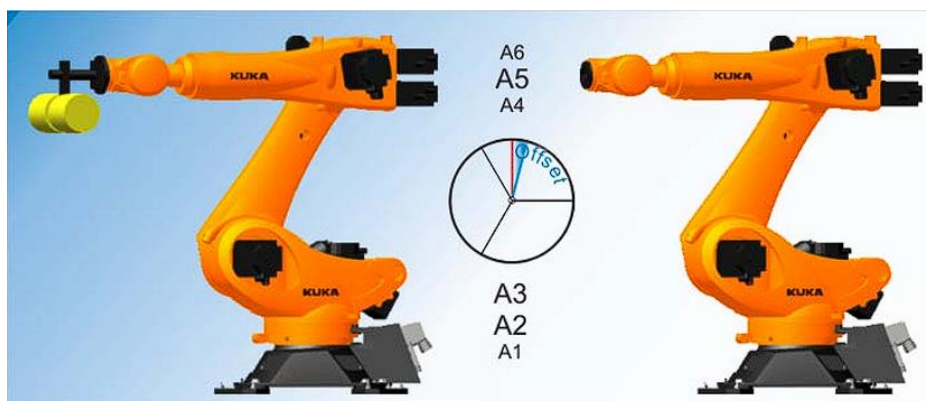


Рисунок 3.1.4.1 – Предъюстировочное положение

4. Юстировка с использованием EMD:

- Снять защитную крышку с измерительного патрона.
- Навинтить устройство EMD на патрон.
- Подключить провод к разъему X32 на коробке выводов робота.
- Активировать процесс юстировки через меню на пульте управления KUKA smartPAD.
- Держать клавишу запуска и подтвердить действие.
- Дождаться завершения юстировки и затемнения оси в окне настроек.

Подключённые элементы юстировочного комплекта во время юстировки представлены на рисунке 3.1.4.2.



Рисунок 3.1.4.2 – Подключённые элементы юстировочного комплекта

5. Завершение юстировки:

- Отсоединить провод от устройства и разъема.
- Установить защитную крышку на патрон.
- Проверить точность юстировки путем тестового перемещения.

Проверка и фиксация данных

- Все результаты юстировки сохраняются в лог-файле Mastery.log по пути:
- C:\KRC\ROBOTER\LOG\Mastery.log
- В лог-файле фиксируются:
 - Дата и время юстировки.
 - Серийный номер оси.
 - Значение юстировки (FirstEncoderValue).
 - Разница энкодера (Encoder Difference).
 - Номер инструмента.

3.2 Калибровка инструмента робота KUKA

Калибровка инструмента робота KUKA — это процесс определения точки TCP (Tool Center Point) и ориентации инструмента относительно фланца манипулятора. Основная цель калибровки — обеспечение точности позиционирования и правильного выполнения рабочих операций роботом.

3.2.1 Зачем нужна калибровка?

Калибровка инструмента позволяет:

1. Точно определить точку центра инструмента (TCP).
2. Обеспечить правильное позиционирование и ориентацию инструмента.
3. Улучшить точность выполнения задач с инструментом.
4. Гарантировать правильное движение робота относительно точки TCP и ориентации инструмента.

3.2.2 Комплект калибровки

Для калибровки инструмента используется:

1. **Электронное калибровочное устройство (EMD)** - для настройки и измерения.
2. **KUKA smartPAD** - для управления процессом калибровки.
3. **Захват или штифт** - инструмент, используемый в качестве эталона.

3.2.3 Подготовка к калибровке

1. **Режим работы:** активировать режим T1 (ручной режим с пониженной скоростью).
2. **Проверка безопасности:** убедиться в активации аварийного останова.
3. **Выбор инструмента:** установить и зафиксировать инструмент на фланце робота.
4. **Активировать систему координат инструмента (TOOL):** проверить, что координатная система активна.

3.2.4 Порядок калибровки

Калибровка методом «XYZ, 4 точки»

1. Запуск процедуры:

- Выбрать последовательность меню:
- Пуск в эксплуатацию> Калибровка> Инструмент> XYZ, 4 точки
- Присвоить номер и имя инструменту.
- Нажать кнопку **ОК** для подтверждения.

2. Подвод точки ТСП к отсчетной точке:

- Сначала подвести инструмент к первой точке калибровки.
- Нажать кнопку **ОК** для сохранения точки.

3. Измерение в других направлениях:

- Повторить измерения с трех оставшихся направлений.
- После каждого измерения нажимать кнопку **ОК**.

4. Сохранение данных:

- Нажать кнопку **Сохранить** для фиксации данных инструмента.

Визуализация процесса калибровки инструмента представлена на рисунке 3.2.4.1.

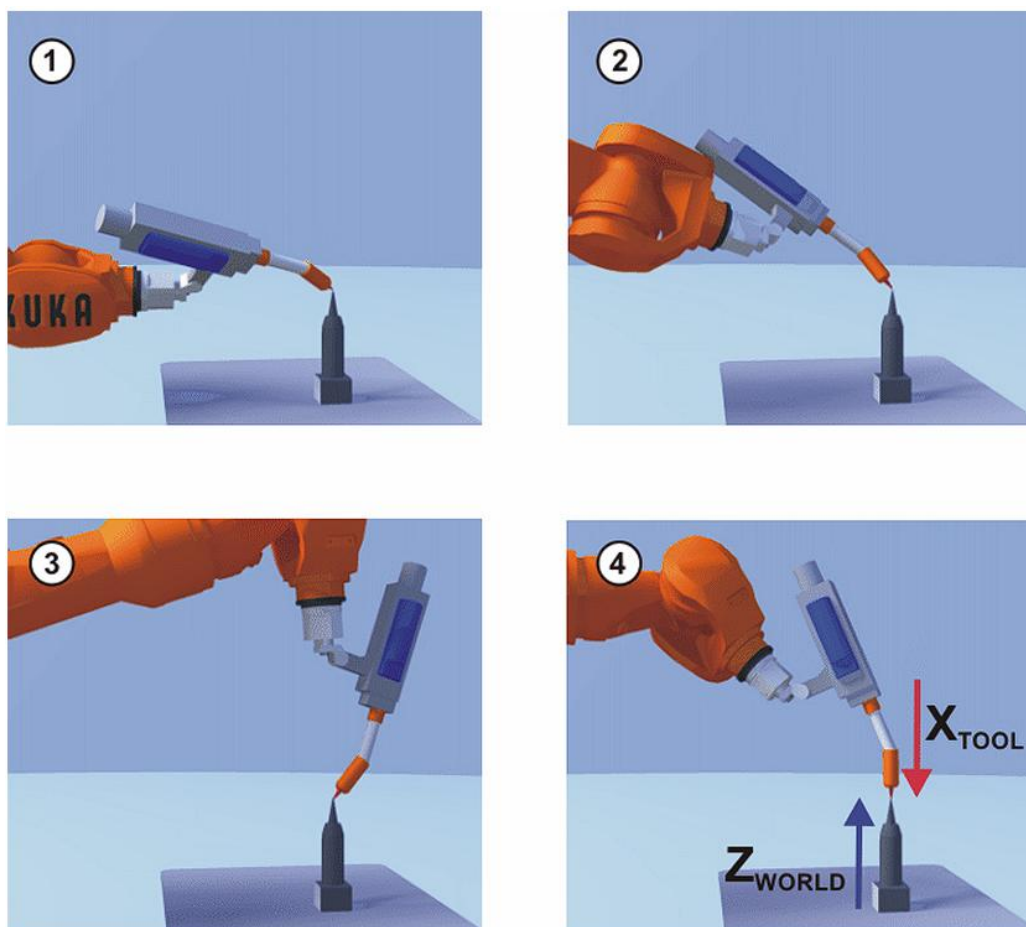


Рисунок 3.2.4.1 – Калибровка методом «XYZ, 4 точки»

3.3 Калибровка базы

Калибровка базы робота KUKA — это процесс настройки системы координат базы робота, чтобы обеспечить точное перемещение инструмента относительно рабочей поверхности. Этот процесс особенно важен при выполнении задач, требующих высокой точности, таких как сварка или сборка.

3.3.1 Зачем нужна калибровка базы?

Калибровка базы позволяет:

1. Определить начальную точку системы координат.
2. Настроить направление координатных осей.
3. Выполнять перемещения вдоль кромок заготовок и других элементов рабочей зоны.

4. Обеспечить использование нескольких базовых систем координат в зависимости от этапа программы (до 32 систем).

3.3.2 Методы калибровки базы

Для настройки базовой системы координат используются три основных метода:

1. **Метод трех точек (3 точки):** определение начала координат, направления оси X и направления оси Y.
2. **Косвенный метод:** используется, если физический доступ к точке базы невозможен.
3. **Цифровой ввод:** прямой ввод координат относительно универсальной системы координат (X, Y, Z) и поворота (A, B, C).

3.3.3 Комплект калибровки

Для проведения калибровки базы требуется:

1. **KUKA smartPAD:** для управления процессом калибровки.
2. **Откалиброванный инструмент (например, захват):** точка TCP которого известна и откалибрована ранее.
3. **Рабочее поле:** например, прямоугольник.

3.3.4 Подготовка к калибровке

1. **Режим работы:** убедиться, что робот находится в режиме T1 (ручной режим с пониженной скоростью).
2. **Проверка безопасности:** деблокировать аварийный останов и убедиться в готовности системы.
3. **Выбор инструмента:** активировать откалиброванный инструмент на пульте управления.

Во время калибровки базы было принято решение воспользоваться методом трех точек, так как он просто в исполнении и точен.

3.3.5 Порядок калибровки методом трех точек

1. Настройка калибровки

1. На пульте управления выбрать:
2. Пуск в эксплуатацию > Калибровка > База > 3 точки
3. Присвоить базе номер и имя (например, "Синяя база"). Нажать кнопку **Далее** для подтверждения.

2. Выбор инструмента

1. Ввести номер инструмента, точка TSP которого будет использоваться для калибровки базы.
2. Подтвердить выбор нажатием кнопки **Далее**.

3. Определение начала координат

1. С помощью TSP подвести инструмент к началу системы координат новой базы.
2. Нажать программируемую клавишу **Калибровка** и кнопку **Да** для подтверждения положения.

4. Определение положительного направления оси X

1. Переместить инструмент к точке на положительной оси X.
2. Нажать кнопку **Калибровка** и кнопку **Да** для фиксации положения.

5. Определение плоскости XY

1. Подвести инструмент к точке с положительным значением Y на плоскости XY.
2. Нажать кнопку **Калибровка** и кнопку **Да** для подтверждения положения.

6. Сохранение данных

1. Нажать кнопку **Сохранить**.
2. Закрыть меню калибровки.

Визуализация процесса калибровки базы представлена на рисунке 3.3.5.1.

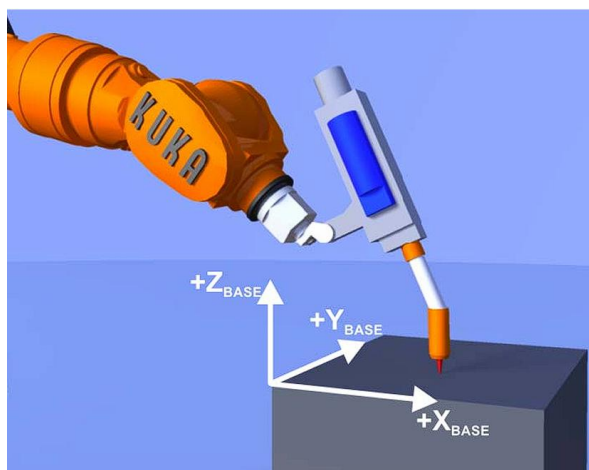


Рисунок 3.3.5.1 – Калибровка методом трех точек

Проверка калибровки базы

1. Переместить инструмент к началу координат новой базы.
2. Вывести фактическое положение на экран в прямоугольных координатах.
3. Сравнить полученные значения с расчетными.

4. Ход работы

Для начала переходим в режим “Эксперт” (рисунок 4.1).

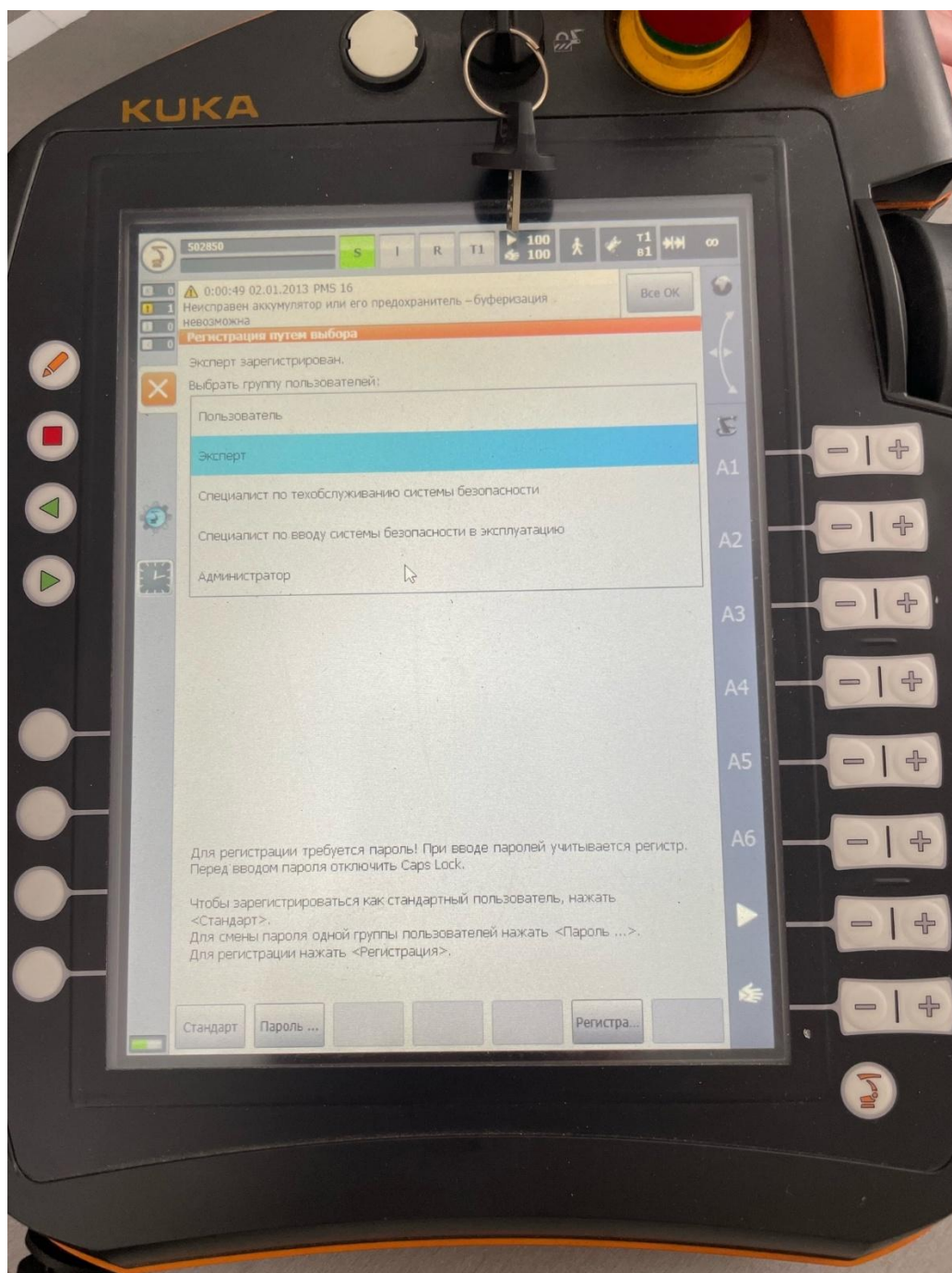


Рисунок 4.1 - Режим эксперта

Производим калибровку инструмента, что показано на рисунке 4.2.

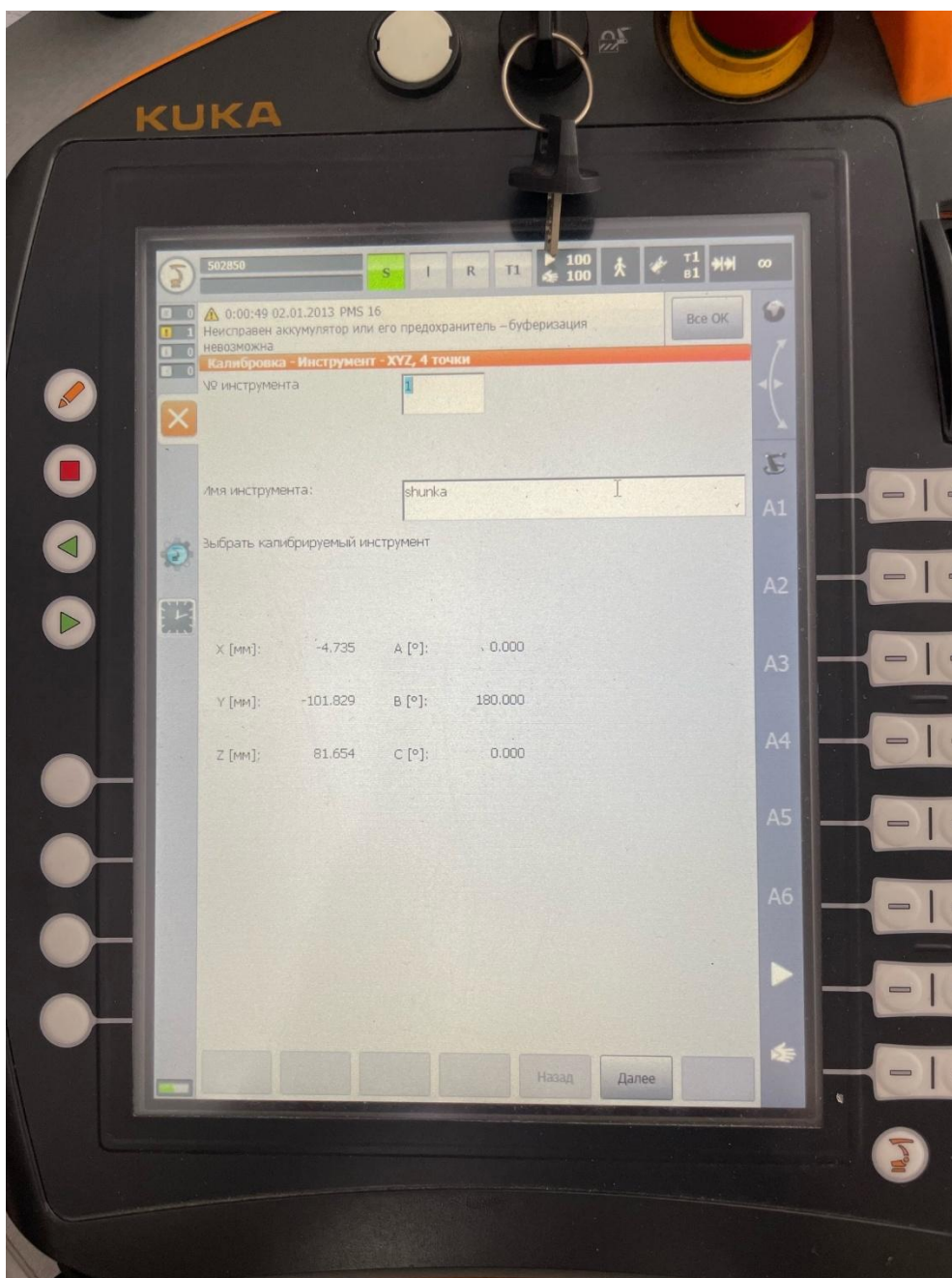


Рисунок 4.2 - Калибровка инструмента

Создаём базу для удобства управления (рисунок 4.3).

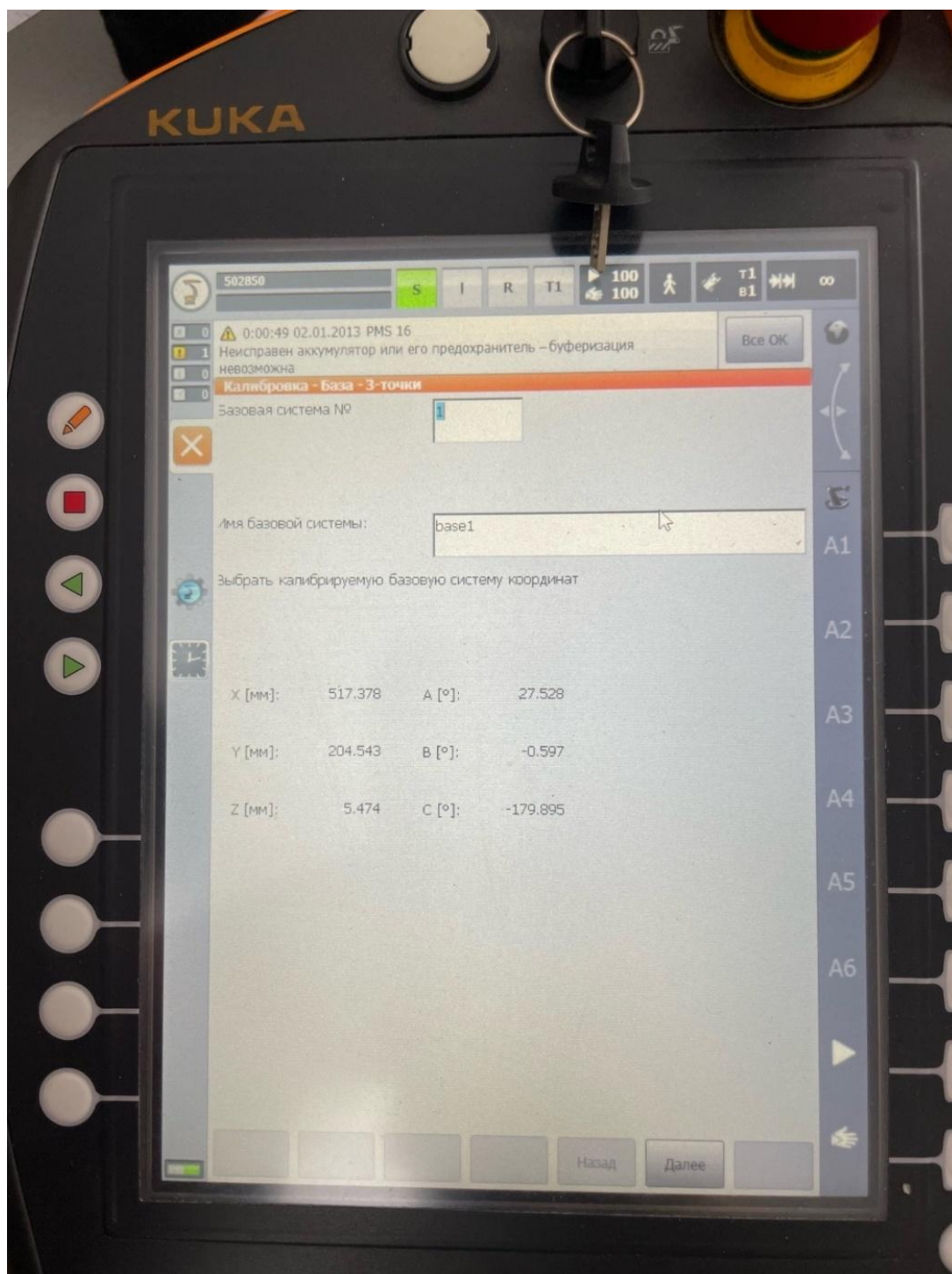


Рисунок 4.3 - Создание базы

Далее размещаем в рамках базы 3 кубика, которые будет перемещать манипулятор, как показано на рисунке 4.4.

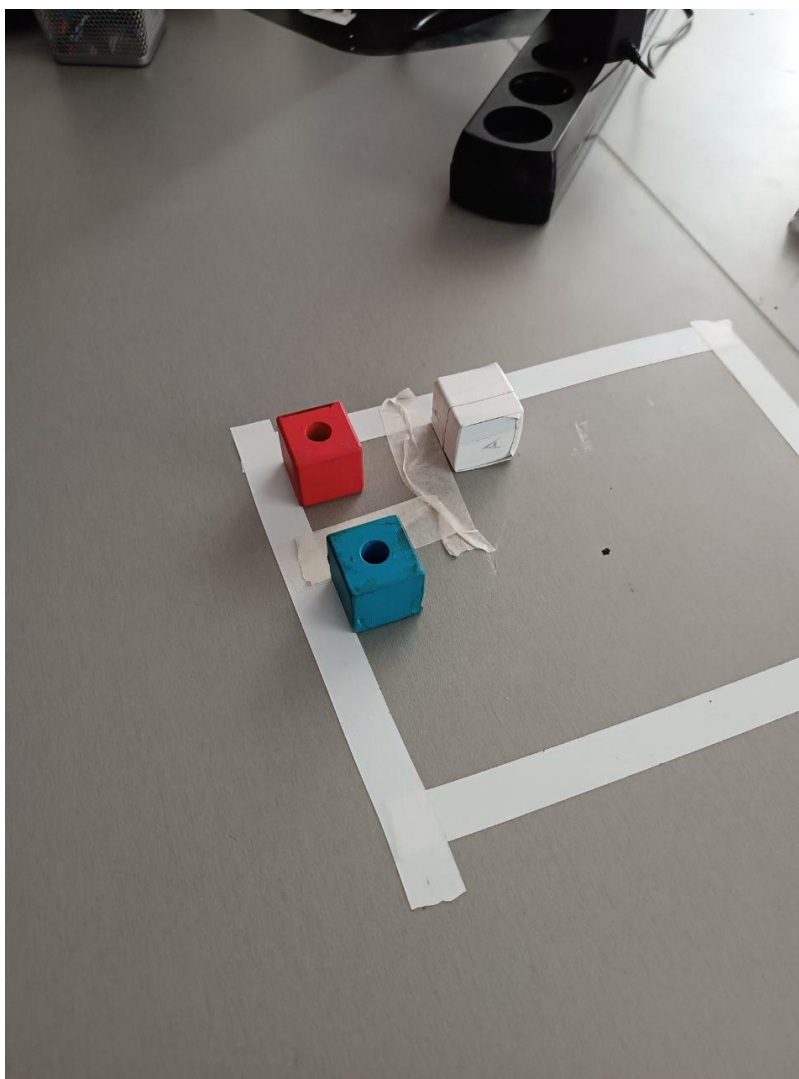


Рисунок 4.4 - Расположение кубиков перед работой

Место размещения кубиков должно быть строго определено, так как программа устройство не имеет СТЗ, и манипулятор по программе ожидает, что кубики будут в конкретных точках. Расположить кубики лучше на углах базы, так как в этом случае кубики будут точно определены по всем координатам, также можно расположить кубики в ряд (при условии, что ряд также начинается с одного из углов базы), но в этом случае манипулятор должен работать с максимальной точностью, чтобы не задеть другие кубики. В промышленности некоторые детали также фиксируются.

Мы расположили кубики, сделав метки у одного из углов базы, в этом случае манипулятор будет работать в более маленькой области и не будет задевать другие кубики.

5. Программный код

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

```
DEF russia()
```

```
INI
```

```
    ; Move to the home position with a speed of 100%
```

```
    PTP HOME Vel= 100 % DEFAULT
```

```
    ; Wait for 1 second
```

```
    WAIT Time=1 sec
```

```
    ; Activate output signal 3 (release)
```

```
    OUT 3 'razgim' State=TRUE CONT
```

```
    ; Wait for 2 seconds
```

```
    WAIT Time=2 sec
```

```
    ; Deactivate output signal 3 (release)
```

```
    OUT 3 'razgim' State=FALSE CONT
```

```
    ; Wait for 2 seconds
```

```
    WAIT Time=2 sec
```

```
    ; Move to point P4 with a speed of 100%
```

```
    PTP P4 Vel=100 % PDAT4 Tool[1]:shunka Base[1]:base1
```

```
    ; Linear movement to point P5 with a speed of 2 m/s
```

```
    LIN P5 Vel=2 m/s CPDAT1 Tool[1]:shunka Base[1]:base1
```

```
    ; Activate output signal 4 (grip)
```

```
    OUT 4 'zagim' State=TRUE CONT
```

```
    ; Wait for 2 seconds
```

```
    WAIT Time=2 sec
```

```
    ; Deactivate output signal 4 (grip)
```

```
    OUT 4 'zagim' State=FALSE CONT
```

```
    ; Linear movement to point P6 with a speed of 2 m/s
```

```
    LIN P6 Vel=2 m/s CPDAT2 Tool[1]:shunka Base[1]:base1
```

```
    ; Move to point P7 with a speed of 100%
```

```
    PTP P7 Vel=100 % PDAT5 Tool[1]:shunka Base[1]:base1
```

```
    ; Linear movement to point P8 with a speed of 2 m/s
```

```
    LIN P8 Vel=2 m/s CPDAT3 Tool[1]:shunka Base[1]:base1
```

```

; Activate output signal 3 (release)
OUT 3 'razgim' State=TRUE CONT
; Wait for 2 seconds
WAIT Time=2 sec
; Deactivate output signal 3 (release)
OUT 3 'razgim' State=FALSE CONT
; Linear movement to point P9 with a speed of 2 m/s
LIN P9 Vel=2 m/s CPDAT4 Tool[1]:shunka Base[1]:base1
; Move to point P10 with a speed of 100%
PTP P10 Vel=100 % PDAT6 Tool[1]:shunka Base[1]:base1
; Linear movement to point P11 with a speed of 2 m/s
LIN P11 Vel=2 m/s CPDAT5 Tool[1]:shunka Base[1]:base1
; Activate output signal 4 (grip)
OUT 4 'zagim' State=TRUE CONT
; Wait for 2 seconds
WAIT Time=2 sec
; Deactivate output signal 4 (grip)
OUT 4 'zagim' State=FALSE CONT
; Linear movement to point P13 with a speed of 2 m/s
LIN P13 Vel=2 m/s CPDAT6 Tool[1]:shunka Base[1]:base1
; Move to point P14 with a speed of 100%
PTP P14 Vel=100 % PDAT8 Tool[1]:shunka Base[1]:base1
; Linear movement to point P15 with a speed of 2 m/s
LIN P15 Vel=2 m/s CPDAT7 Tool[1]:shunka Base[1]:base1
; Activate output signal 3 (release)
OUT 3 'razgim' State=TRUE CONT
; Wait for 2 seconds
WAIT Time=2 sec
; Deactivate output signal 3 (release)
OUT 3 'razgim' State=FALSE CONT
; Linear movement to point P16 with a speed of 2 m/s
LIN P16 Vel=2 m/s CPDAT8 Tool[1]:shunka Base[1]:base1

```

```

; Move to point P17 with a speed of 100%
PTP P17 Vel=100 % PDAT9 Tool[1]:shunka Base[1]:base1
; Linear movement to point P18 with a speed of 2 m/s
LIN P18 Vel=2 m/s CPDAT9 Tool[1]:shunka Base[1]:base1
; Activate output signal 4 (grip)
OUT 4 'zagim' State=TRUE CONT
; Wait for 2 seconds
WAIT Time=2 sec
; Deactivate output signal 4 (grip)
OUT 4 'zagim' State=FALSE CONT
; Linear movement to point P19 with a speed of 2 m/s
LIN P19 Vel=2 m/s CPDAT10 Tool[1]:shunka Base[1]:base1
; Move to point P20 with a speed of 100%
PTP P20 Vel=100 % PDAT10 Tool[1]:shunka Base[1]:base1
; Linear movement to point P21 with a speed of 2 m/s
LIN P21 Vel=2 m/s CPDAT11 Tool[1]:shunka Base[1]:base1
; Activate output signal 3 (release)
OUT 3 'razgim' State=TRUE CONT
; Wait for 2 seconds
WAIT Time=2 sec
; Deactivate output signal 3 (release)
OUT 3 'razgim' State=FALSE CONT
; Linear movement to point P22 with a speed of 2 m/s
LIN P22 Vel=2 m/s CPDAT12 Tool[1]:shunka Base[1]:base1
; Return to the home position with a speed of 100%
PTP HOME Vel= 100 % DEFAULT
END

```

6. Результаты работы программы

В результате проверки программы она успешно завершилась и выполнила поставленные задачи, что можно увидеть на рисунке 5.1.



Рисунок 5.1 - Результат работы программы

7. Заключение

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

Была произведена юстировка, калибровка инструмента и базы. Результатом курсового проекта стало создание программа для построения башни.

8. Список литературы

1. <https://wikis.utexas.edu/display/SOAdigitech/KUKA+Programming+KRL+Examples>
2. <https://drstienecker.com/tech-332/11-the-kuka-robot-programminglanguage/>
3. https://swsu.ru/sveden/files/PROGRAMMIROVANIE_PROMYSHLENNOGO_ROBOTA_KUKA_LAB.pdf
4. https://www.youtube.com/watch?v=GtxShP_Wtec&t=171s&ab_channel=FutureRobotics
5. KUKA Roboter GmbH «Программирование робота 1 KUKA System Software 8 Учебная документация», 2015.