

Инструкция по эксплуатации и программированию
промышленных роботов Robotech
для конечного пользователя

Оглавление

1.	Введение	3
2.	Описание продукта.....	3
2.1.	Функции, выполняемые промышленным роботом	3
2.2.	Основные компоненты промышленного робота	3
3.	Безопасность.....	3
3.1.	Общая информация.....	3
3.2.	Описание основных терминов безопасности	4
4.	Режимы управления роботом.....	6
5.	Ввод в эксплуатацию.....	8
5.1.	Юстировка осей робота и дополнительных осей.....	8
5.2.	Калибровка инструмента робота	10
5.3.	Калибровка базы робота	13
5.4.	Конфигурирование периферийных сигналов робота	14
5.5.	Ручное управление периферийным оборудованием	15
6.	Ручной режим управления	17
6.1.	Ручное управление периферийными сигналами	18
7.	Создание управляющих программ	19
7.1.	Структура программы робота.....	20
7.2.	Типы данных	22
7.3.	Предопределенные имена переменных	23
7.4.	Операторы.....	24
7.5.	Команды движения робота	26
7.6.	Встроенные функции языка.....	27
8.	Выполнение программы.....	29
8.1.	Отображение состояния переменных.....	30
9.	Программирование с помощью мастера команд	31
10.	Диагностика	32

1. Введение

Данная документация предназначена для пользователей со следующими знаниями:

- базовые знания о промышленных роботах

2. Описание продукта

2.1. Функции, выполняемые промышленным роботом

- Перемещение по траекториям
- Управление устройствами ввода/вывода
- Обмен данными с вспомогательными устройствами, а также системами управления

верхнего уровня

- Управление файлами с программами робота
- Функции диагностики
- Управление правами пользователей
- Дополнительные функции

2.2. Основные компоненты промышленного робота

Промышленный робот состоит из следующих компонентов:

- Манипулятор
- Система управления роботом
- Пульт управления
- Программное обеспечение
- Дополнительное оборудование, принадлежности

3. Безопасность

3.1. Общая информация

Устройство, описанное в настоящем документе, представляет собой промышленный робот или один из его компонентов.

Компоненты промышленного робота:

- Манипулятор
- Система управления роботом
- Ручной программатор
- Соединительные кабели
- Дополнительные оси (дополнительное оборудование), например, линейный блок, поворотный откидной стол, устройство позиционирования
- Программное обеспечение
- Дополнительное оборудование, принадлежности

Промышленный робот выполнен в соответствии с существующим уровнем развития техники и общепринятыми правилами техники безопасности. Однако при ошибочном применении робота существует опасность для здоровья и жизни, а также риск повреждения робототехнической системы и нанесения материального ущерба.

Эксплуатация промышленного робота допускается только в безупречном техническом состоянии, а также согласно назначению и с учетом предписаний по технике безопасности и существующих рисков. Эксплуатация должна осуществляться согласно настоящему документу, входящей в комплект поставки промышленного робота. Неисправности, которые могут отрицательно повлиять на безопасность, необходимо незамедлительно устранять.

Указания по технике безопасности не могут быть истолкованы против производителя. Даже при соблюдении всех указаний по технике безопасности нет гарантии того, что в рамках эксплуатации промышленного робота исключена опасность травм или материального ущерба. Без разрешения производителя вносить какие-либо изменения в конструкцию промышленного робота запрещено. Изменения без разрешения ведут к отмене рекламационных претензий и претензий по качеству. В состав промышленного робота могут быть включены дополнительные компоненты (инструменты, программное обеспечение и т. д.), не входящие в комплект поставки производителя. Ответственность за повреждения промышленного робота или иного имущества, причиненные в связи с установкой данных компонентов, несет пользователь. Кроме главы «Безопасность» в данной документации содержатся дополнительные указания по технике безопасности. Они также подлежат обязательному соблюдению.

3.2. Описание основных терминов безопасности

Рабочая зона оси	Зона действия каждой оси, указанная в градусах или миллиметрах, в пределах которой она может перемещаться. Зона перемещения оси должна определяться для каждой оси отдельно.
Остановочный путь	Остановочный путь = путь реакции + тормозной путь Остановочный путь является частью опасной зоны.
Рабочая зона	Зона, в которой манипулятор может передвигаться. Рабочая зона состоит из отдельных рабочих зон осей.
Опасная зона	Опасная зона включает в себя рабочую зону и остановочные пути манипулятора и дополнительных осей (опция).
Срок службы	Срок службы важных для безопасности деталей начинается с момента ее поставки заказчику. На продолжительность срока службы не влияет фактическая продолжительность эксплуатации детали, так как важные для безопасности детали подвергаются старению и во время хранения.
Безопасный рабочий останов	Безопасный рабочий останов – это функция контроля останова. Он не останавливает перемещение самого робота, а контролирует состояние останова осей робота. Если оси перемещаются при безопасном рабочем останове, срабатывает безопасный останов STOP 0. Безопасный рабочий останов также можно выполнить внешними средствами.
Безопасный останов STOP 0	Останов, который выполняется по команде системы управления безопасностью. Система управления безопасностью мгновенно отключает приводы и подачу питания к тормозам.
Безопасный останов STOP 1	Останов, который выполняет и контролирует система управления безопасностью. Процесс торможения выполняется компонентом системы управления роботом, который не относится к безопасности, и контролируется системой управления безопасностью. Как только манипулятор останавливается, система управления безопасностью отключает приводы и подачу питания к тормозам.
Безопасный останов STOP 2	Останов, который выполняет и контролирует система управления безопасностью. Процесс торможения выполняется компонентом системы управления роботом, который не относится к безопасности, и контролируется системой управления

	безопасностью. Приводы остаются включенными, а тормоза – разомкнутыми. Сразу после остановки манипулятора срабатывает безопасный рабочий останов.
T1	Ручной режим работы с пониженной скоростью (линейная скорость ≤ 250 мм/с, угловая < 30 °/с)
T2	Ручной режим работы с заданной скоростью (допускается > 250 мм/с или 30 °/с)

4. Режимы управления роботом

Системой управления робота предусмотрено четыре режима управления:

- Ручной с замедленной скоростью
- Ручной с заданной скоростью
- Автоматический
- Дистанционный (в разработке)

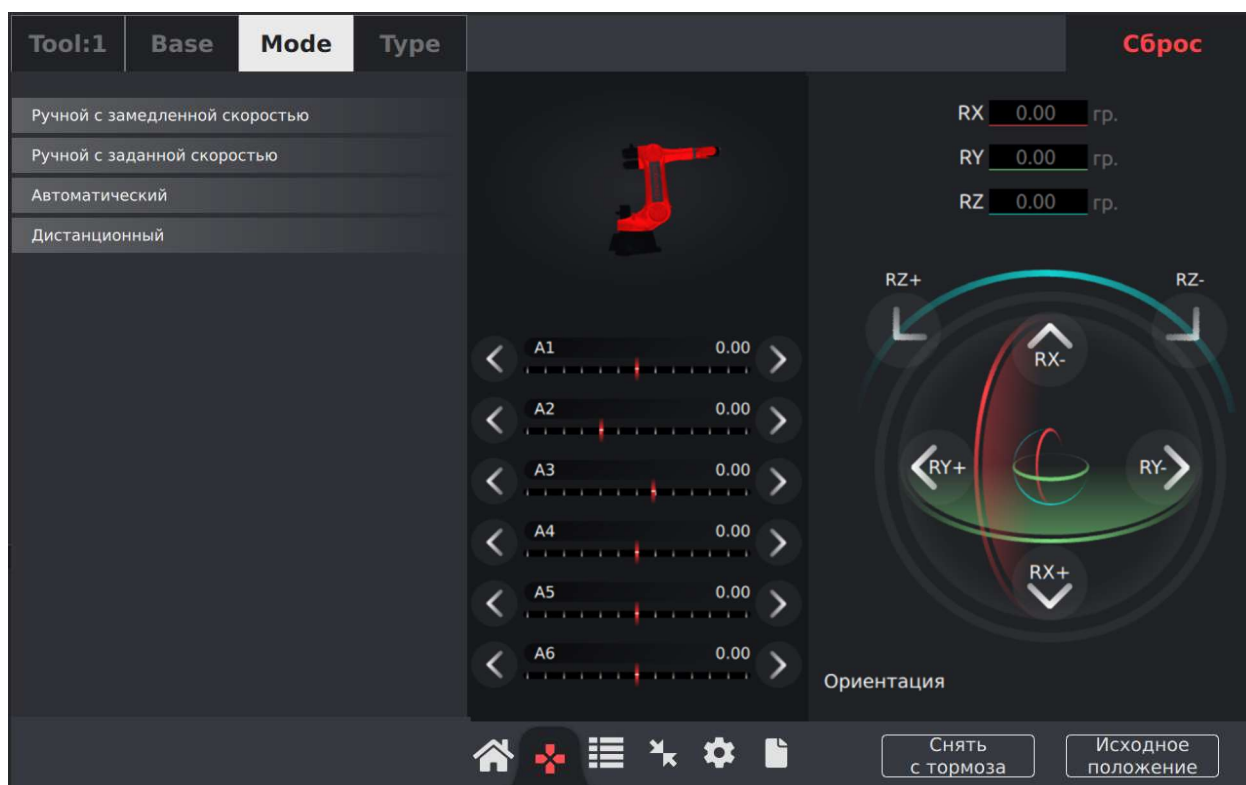
Ручные режимы управления предполагается использовать для создания и отладки управляющих программ.

Ручной режим с замедленной скоростью предполагает движения, ограниченные линейной скоростью 250 мм/с и угловой 30 °/с

Ручной режим с заданной скоростью позволяет перемещать ПР на номинальных скоростях.

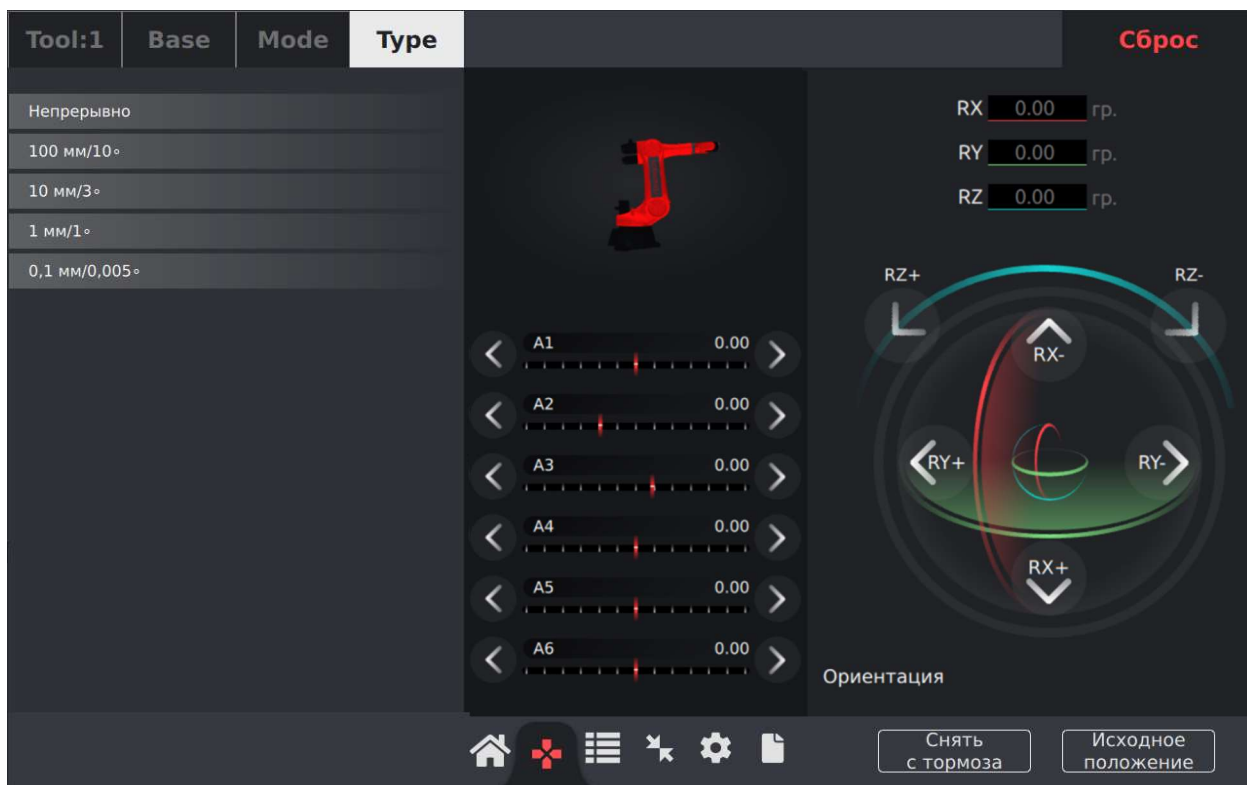
Автоматический режим используется для запуска программ ранее созданных управляющих программ, включающих управление перемещением ПР, а также управление периферийными устройствами, подключенными к нему.

Дистанционный режим предполагает внешнее управление роботом с системы управления верхнего уровня. Управление допускается как через дискретные входа и выхода робота, так и по сети. Данный режим находится в разработке.



Системой управления предусмотрено несколько типов ручного перемещения.

- Непрерывный, при котором робот движется в заданном направлении пока удерживается соответствующая кнопка
- Инкрементальный, с различными вариантами задаваемого смещения: 100/10/1/0,1 мм и 10/3/1/0,005 градуса. При выборе инкрементального типа движения робот двигается на заданное расстояние, при нажатой кнопке



5. Ввод в эксплуатацию

5.1. Юстировка осей робота и дополнительных осей

При поступлении робота от производителя все оси находятся в юстированном состоянии, т.е. для каждой из осей определено нулевое положение относительно которого идет отсчет угловой или линейной координаты.

В случае потери нулевого положения какой-либо из осей, необходимо повторно провести юстировку. В противном случае эксплуатация робота не допускается.

Потеря нулевого положения может быть вызвана:

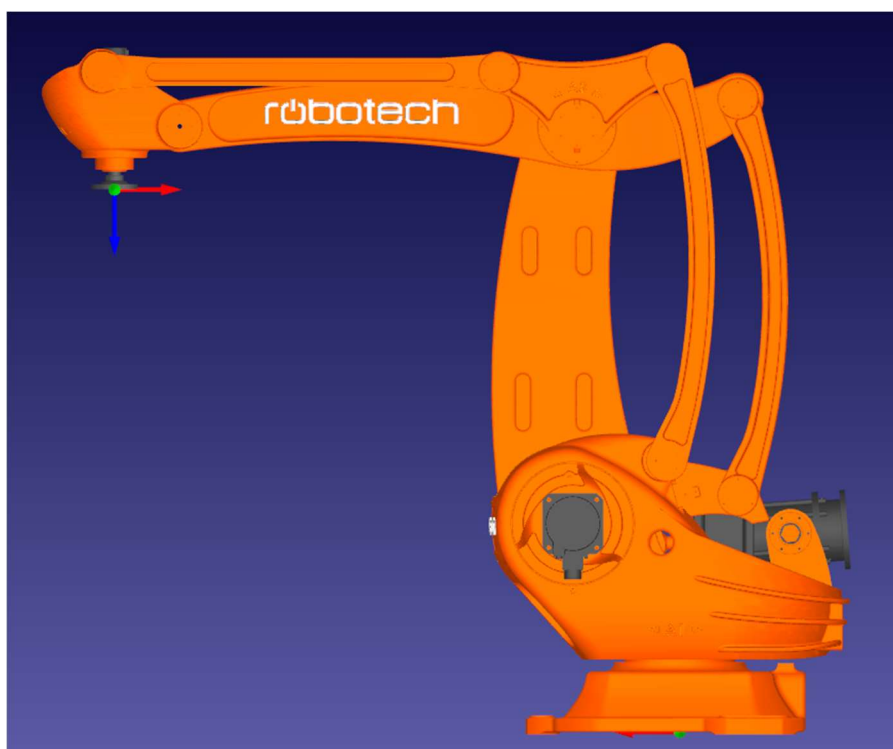
Аварийными событиями:

- Выходом из строя сервопривода
- Механическим повреждением робота (столкновением с каким-либо объектом)

Запланированными событиями:

- Проведением сервисных работ (замена редуктора, смазка неразборных редукторов или пр.)
- Некорректным обновлением версии системы управления робота с переписыванием информации о нулевых позициях осей.

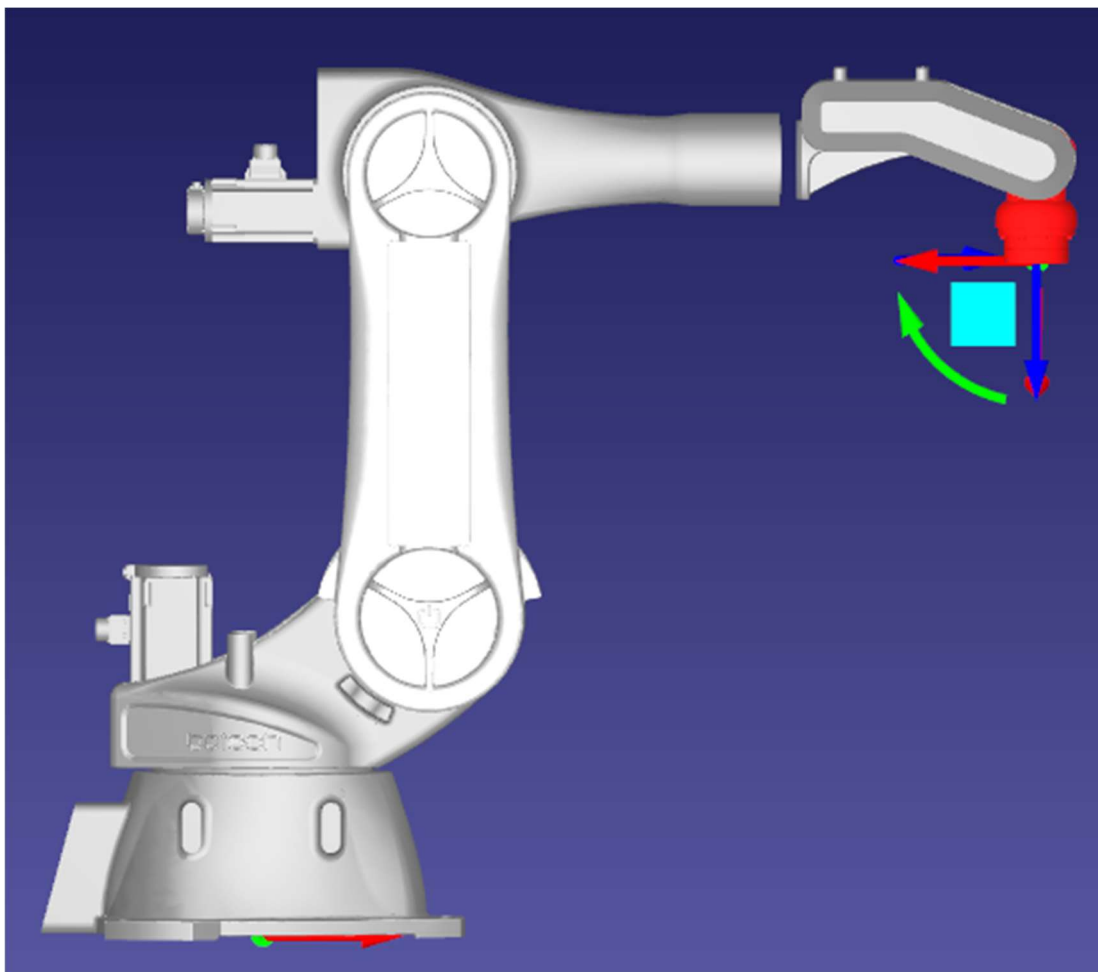
Для робота паллетайзера домашним положением является следующая позиция:



В указанной позиции оси имеют следующие угловые координаты:

№ оси	1	2	3	4
Угловая координата	0.0	-90.0	0.0	0.0

Для 6-ти осевого робота домашним положением является следующее:



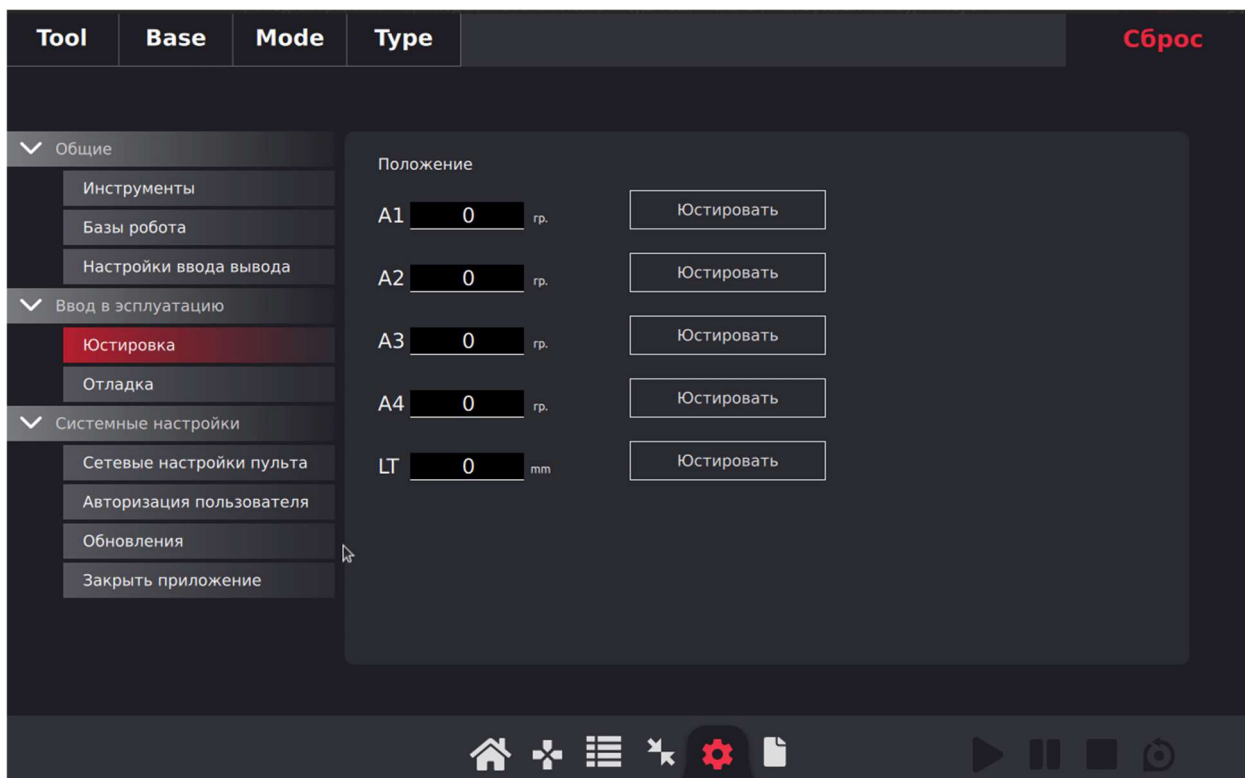
В указанной позиции оси имеют следующие угловые координаты:

№ оси	1	2	3	4	5	6
Угловая координата	0.0	-90.0	0.0	0.0	90.0	0.0

Процесс юстировки можно разделить на два сценария: юстировка после аварии или юстировка после запланированных работ.

Для проведения юстировки необходимо:

- Включить робота и пульт оператора
- Авторизоваться в системе управления с правами доступа «Сервисный инженер»
- Перейти на экран ручного управления и в ручном режиме выставить необходимую ось (или все оси) согласно рисунку выше. При выставлении необходимо руководствоваться метками, имеющимися на каждой из осей робота, а также внешних осей.
- Перейти на экран «Настройки», выбрать меню «Ввод в эксплуатацию» - «Юстировка»



В графе «Положение» необходимо вбить текущее положение угла оси*

В зависимости от кинематики робота кол-во осей на экране юстировки может изменяться, однако процесс остается неизменным.

В случае выполнении юстировки после запланированных работ, сюда стоит вбивать значения, указанные в таблице выше. При выполнении юстировки после аварийного сценария необходимо, до выполнения всех вышеуказанных пунктов сохранить последнее положение оси.

К примеру: во время движения по траектории произошел аварийный останов, связанный с ошибкой сервопривода 1 оси, сигнализирующей о необходимости его замены. В таком случае, до проведения работ, по его замене, необходимо сохранить последнее положение оси, после чего производить замену оборудования. При наличии такого значения и выполнении повторной юстировки, необходимо вбить его в соответствующую графу и только после этого нажимать кнопку юстировать.

Если при аварийном сценарии Вы не сохранили последнее положение и сразу произвели замену сервопривода, то может случиться ситуация, при которой робот перестанет двигаться и будет выдавать ошибку расчета траектории. В таком случае необходимо произвести юстировку в два этапа: юстировать текущее положение как нулевое, после чего робот позволит перемещаться, затем выполнить все вышеуказанные пункты.

ВНИМАНИЕ: а таком режиме работы имеется риск повреждения конструкции или кабелей. Необходимо проявить внимание и осторожность

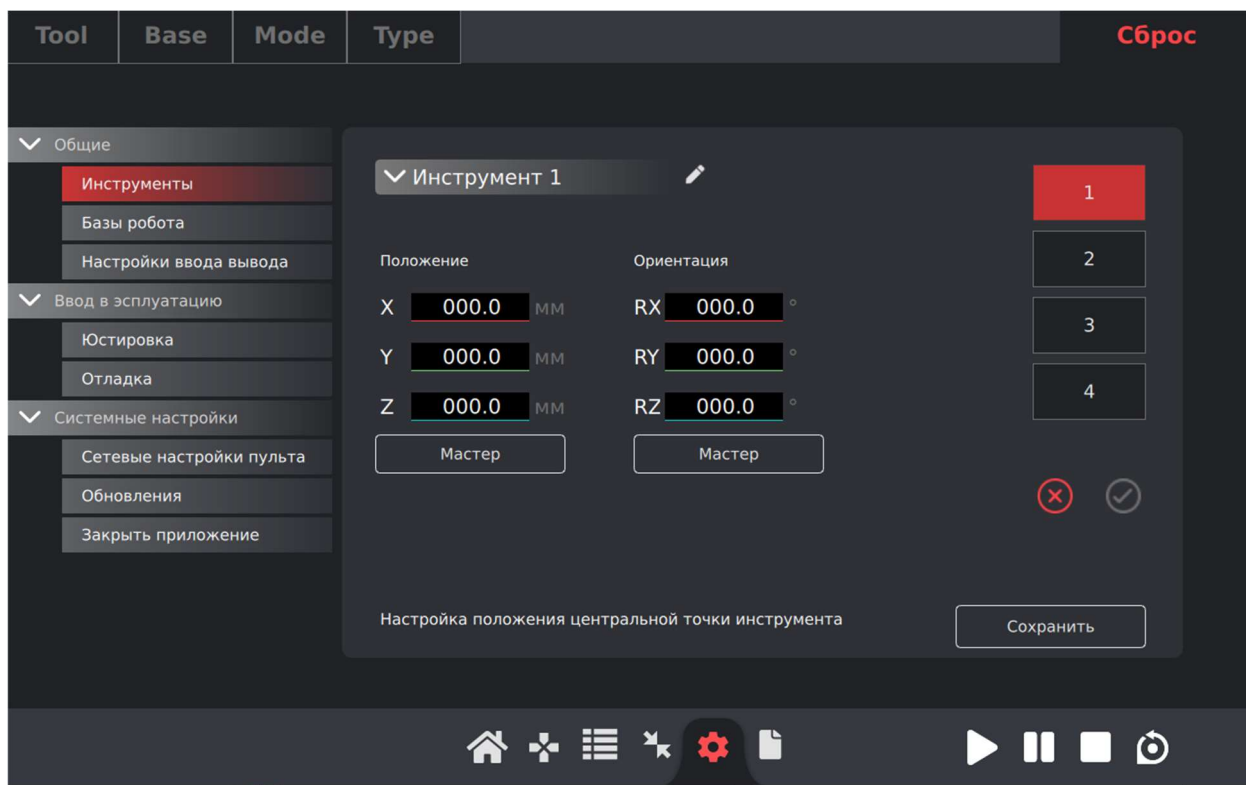
5.2. Калибровка инструмента робота

Для удобства перемещения ПР с закрепленным к его фланцу инструментом удобно использовать пользовательские системы координат (далее СК). Системой предусмотрено 10 пользовательских СК, каждую из которых можно индивидуально откалибровать и задать ей имя.

Калибровку инструмента можно выполнить двумя способами:

Ручным заданием смещения по каждой из координаты X, Y, Z и ручным заданием отклонений RX, RY, RZ. Такой подход часто используется при наличии 3д модели закрепляемого инструмента.

Выполнением последовательной калибровки положения начала СК рабочей точки инструмента (далее TCP), калибровки ориентации TCP.



Калибровка положения производится в 4 этапа, в каждом из которых необходимо сопоставить физическую позицию TCP с неподвижным объектом в пространстве таким образом, чтобы максимально разнести положения робота в пространстве.



При задании каждой из промежуточных позиций система будет переводить на экран ручного управления, с которого вы должны переместить ПР в новую позицию и нажать подтверждающую кнопку.



По завершению калибровки инструмента необходимо нажать кнопку «сохранить» для завершения процесса.

Для калибровки ориентации инструмента предусмотрен механизм с заданием трех точек:

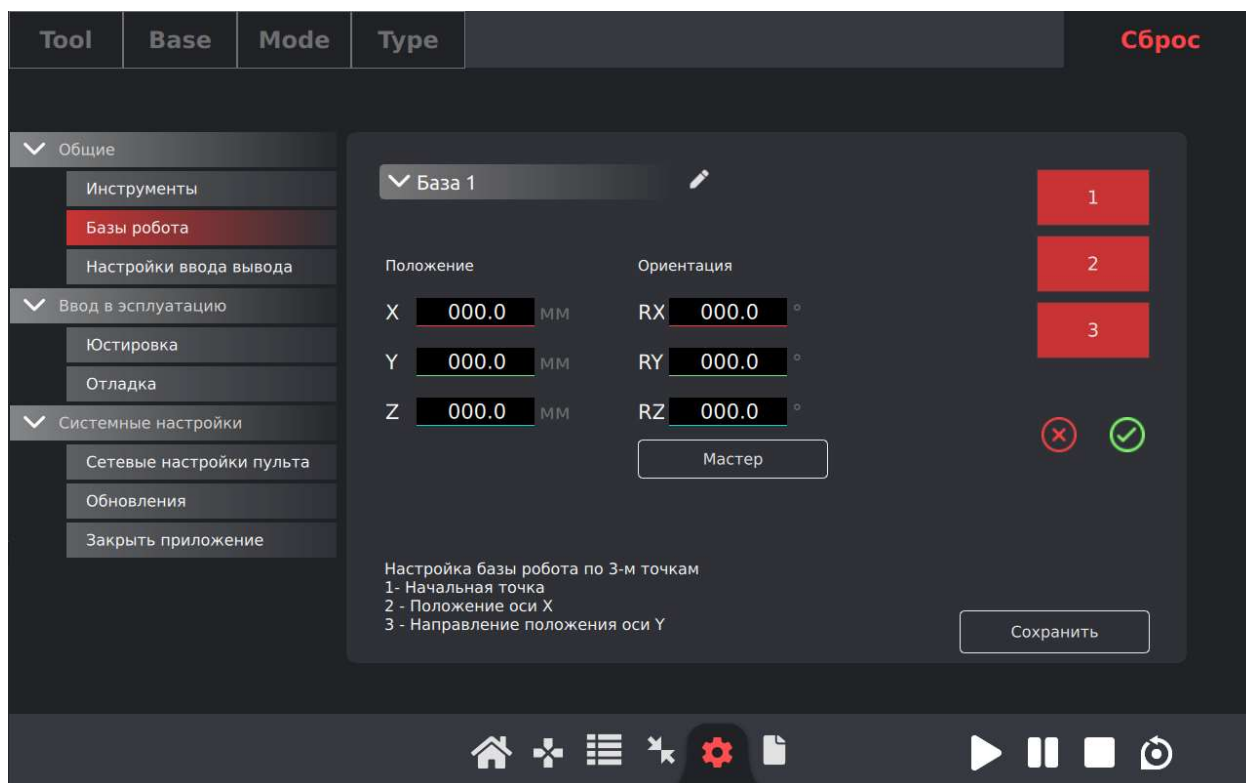
- Начало СК инструмента
- Точка, находящаяся на оси X СК инструмента
- Точка, лежащая в плоскости XY СК инструмента

Примечание: ввиду ограничения кинематики робота-паллетайзера, калибровку ориентации инструмента рекомендуется задавать только с прямыми углами относительно главной СК робота, в противном случае траектория движения ПР может быть не достижима.

5.3. Калибровка базы робота

Процесс калибровки базы робота аналогичен калибровке ориентации инструмента робота и включает в себя следующие этапы:

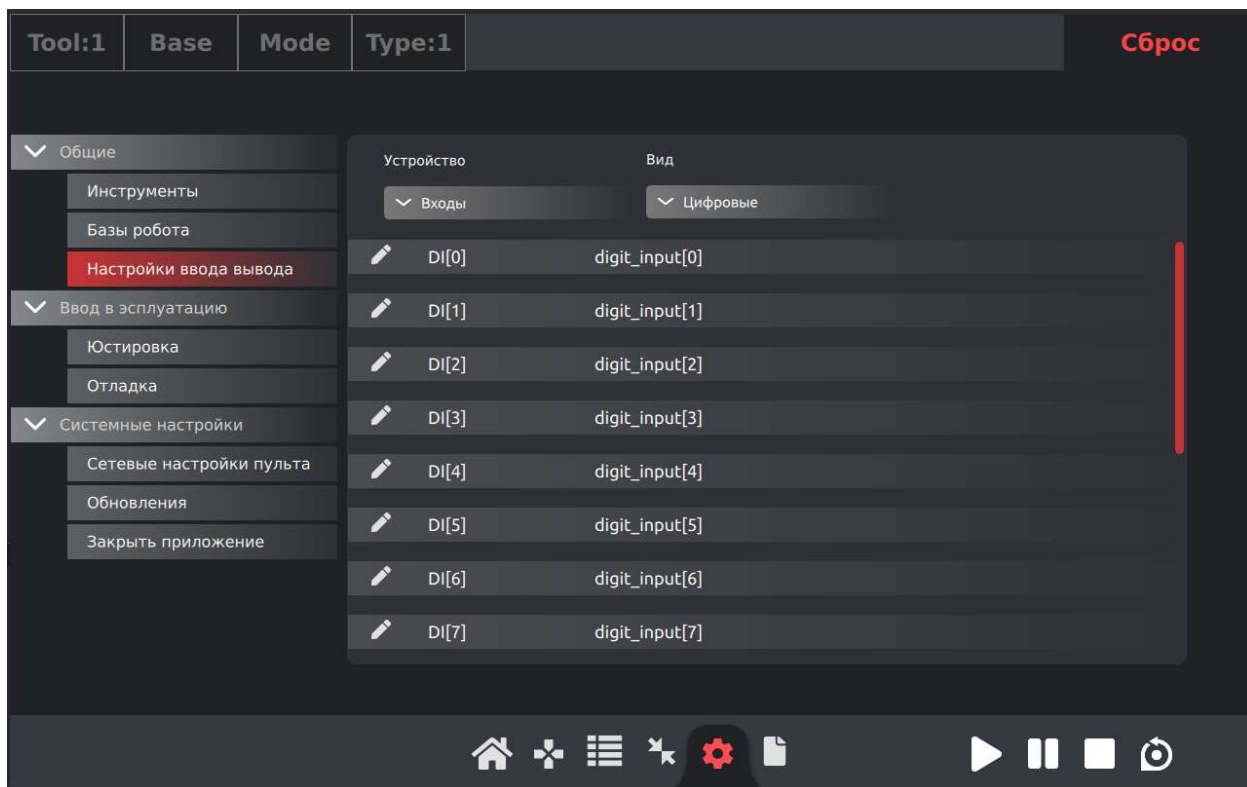
- Задание начальной точки системы координат
- Задание точки, лежащей на положительном направлении вектора оси X
- Задание точки, лежащей в плоскости XY



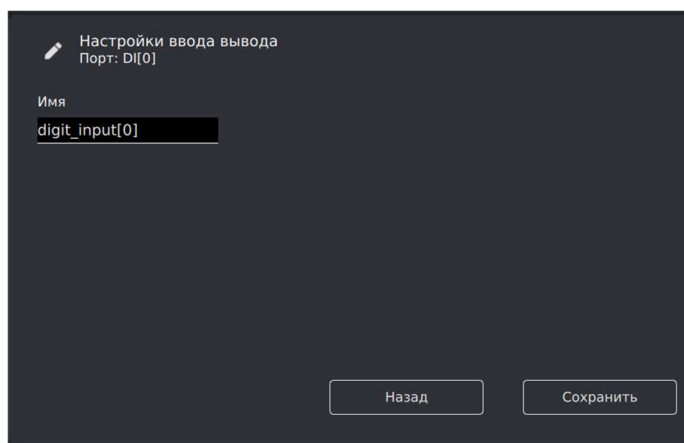
5.4. Конфигурирование периферийных сигналов робота

Для управления периферийным оборудованием, подключенным к ПР, системой управления предусмотрено управление аналоговыми и дискретными сигналами. В зависимости от потребности к роботу может быть подключено разное кол-во приборов.

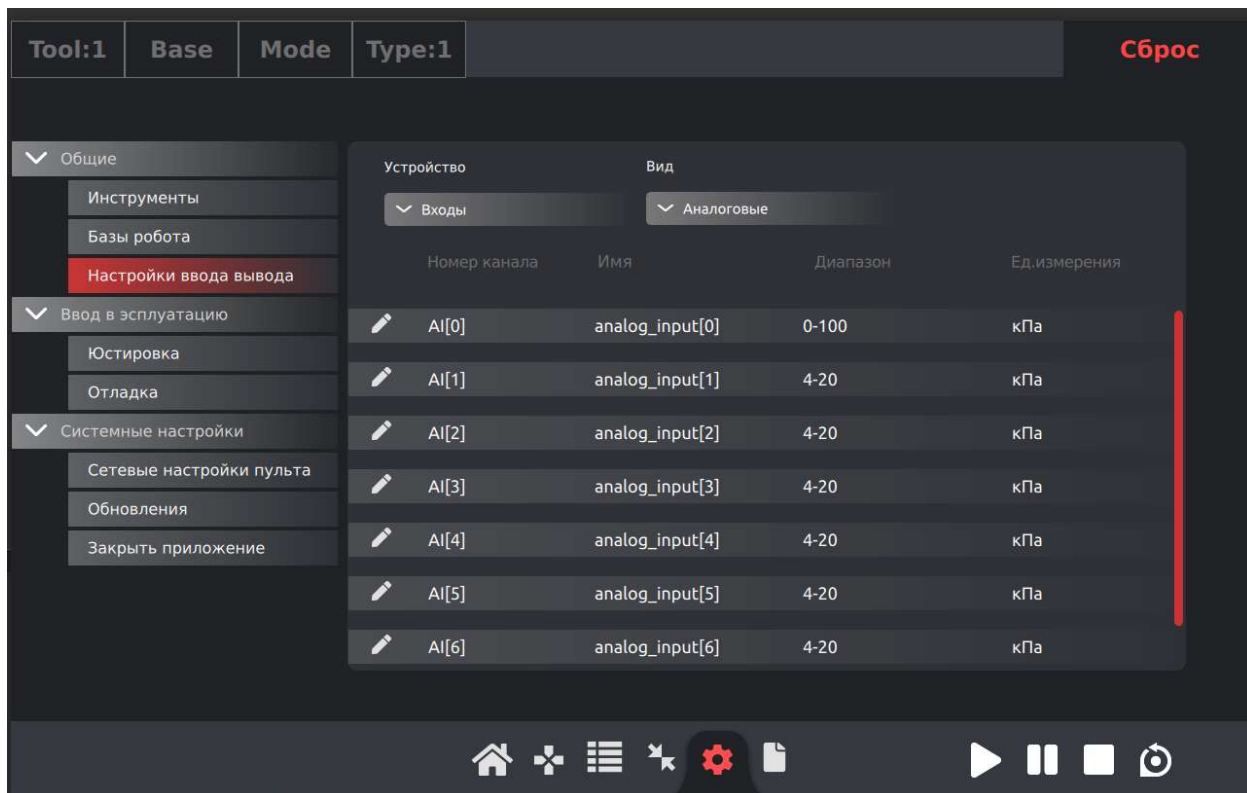
При введении робота в эксплуатацию необходимо сконфигурировать перечень используемых каналов.



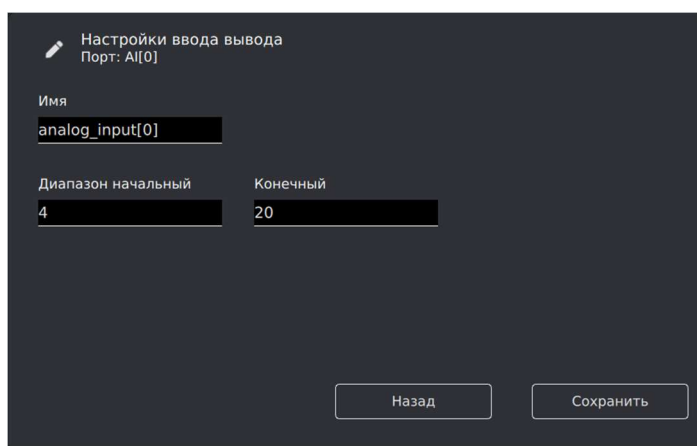
Для дискретных входов/выходов задается имя канала, по которому можно обращаться из программы управления ПР.



Для аналоговых входов/выходов задается имя канала, по которому можно обращаться из программы управления ПР, диапазон измерения датчика/диапазон управления прибором, единицы измерения.



К примеру, при подключении к ПР аналогового датчика температуры, со шкалой измерения - 50..150 градусов вам необходимо нажать на кнопку редактирования и в открывшемся окне задать имя канала, величину начального диапазон измерения равной -50, величину конечного диапазона измерения равным 150 и единицу измерения в °С



5.5. Ручное управление периферийным оборудованием

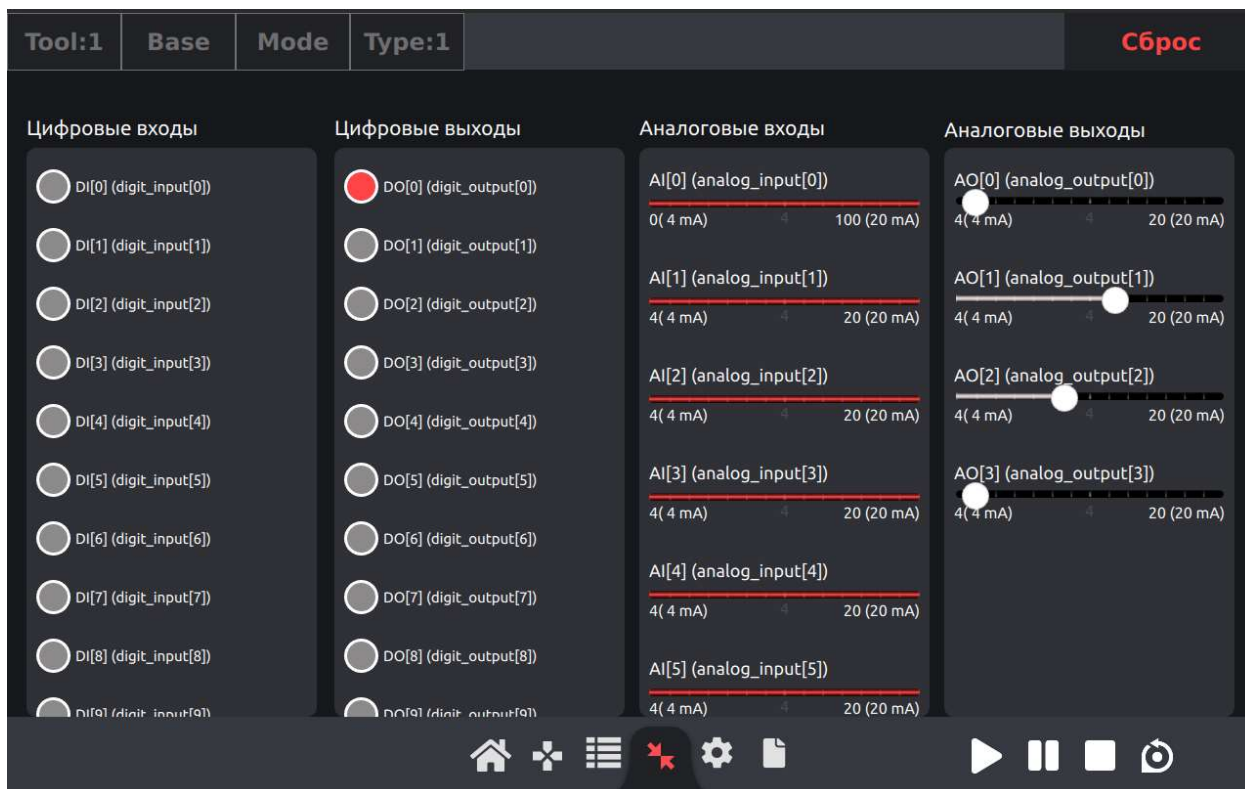
Для ручного управления выходами и просмотру входов робота предусмотрен соответствующий экран.

При наличии сигнала на соответствующем цифровом входе, индикатор становится красного цвета, при отсутствии сигнала – серого цвета.

Для управления цифровыми выходами необходимо нажать на соответствующий индикатор. При включении он становится красного цвета, при отключении – серого.

Для индикации аналоговых сигналов предусмотрены цветные шкалы с указанием единиц измерения канала и его диапазона измерения (параметры задаются в настройках).

Для управления аналоговыми сигналами предусмотрены скролл-бары с возможностью задания величины выхода.



6. Ручной режим управления

Экран ручного управления содержит в себе актуальное положение всех осей робота, положение выбранной СК инструмента относительно выбранной базовой системы координат. По умолчанию отображается положения СК фланца робота (0 СК инструмента), относительно СК основания (0 СК базы).

Для управления перемещением предусмотрены кнопки вращения в положительном и отрицательном направлении каждой из осей, а так же кнопки декартового перемещения в положительном и отрицательном направлении оси X, Y, Z и вращения вокруг оси RX, RY, RZ. Так же имеются кнопки снятия робота со стояночного тормоза и возврат в домашнее положение.

Примечание: кинематика 4х осевого робота-паллетайзера по умолчанию позволяет вращаться только вокруг оси, направленной перпендикулярно фланцу робота (у СК фланца это ось RZ). Если будет выбрана пользовательская СК, с повернутой ориентацией, то робот не сможет осуществить вращение.



Для выбора активной СК инструмента и СК базы необходимо осуществить выбор из выпадающего меню

6.1. Ручное управление периферийными сигналами

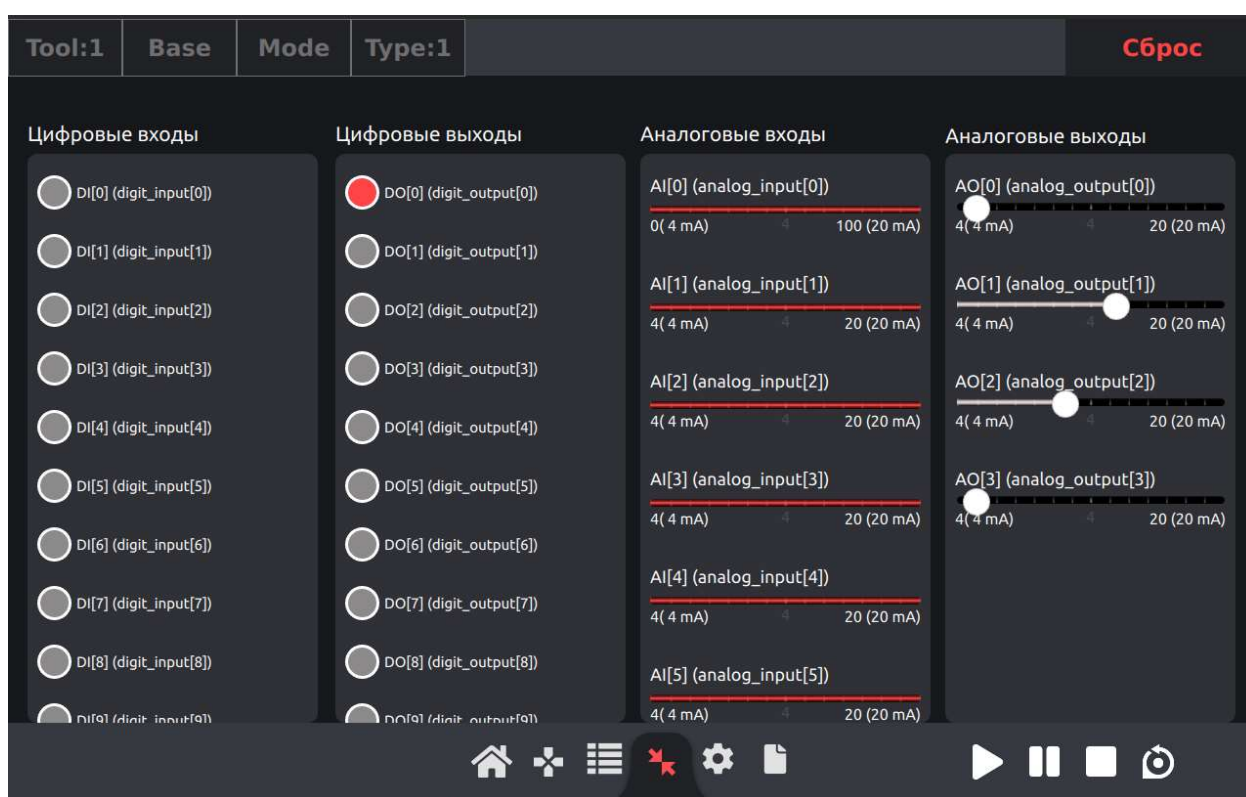
Для ручного управления выходами и просмотру входов робота предусмотрен соответствующий экран.

При наличии сигнала на соответствующем цифровом входе, индикатор становится красного цвета, при отсутствии сигнала – серого цвета.

Для управления цифровыми выходами необходимо нажать на соответствующий индикатор. При включении он становится красного цвета, при отключении – серого.

Для индикации аналоговых сигналов предусмотрены цветовые шкалы с указанием единиц измерения канала и его диапазона измерения (параметры задаются в настройках).

Для управления аналоговыми сигналами предусмотрены скролл-бары с возможностью задания величины выхода.

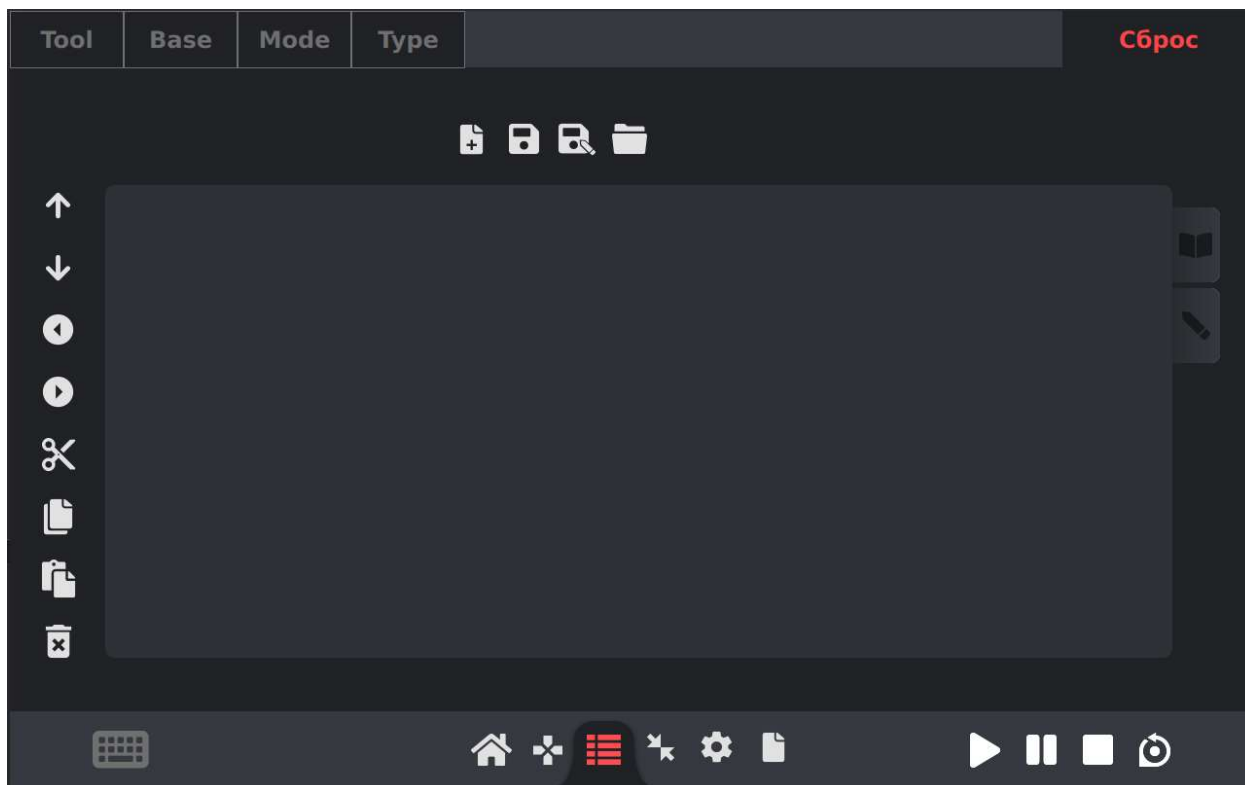


7. Создание управляющих программ

Для работы в автоматическом режиме необходимо сконфигурировать программу с траекториями движения робота и управлением периферийными устройствами. Создание программ доступно для пользователей с правами доступа «Программист».

Создать новую программу можно с соответствующего режима. Существует два подхода к разработке программ:

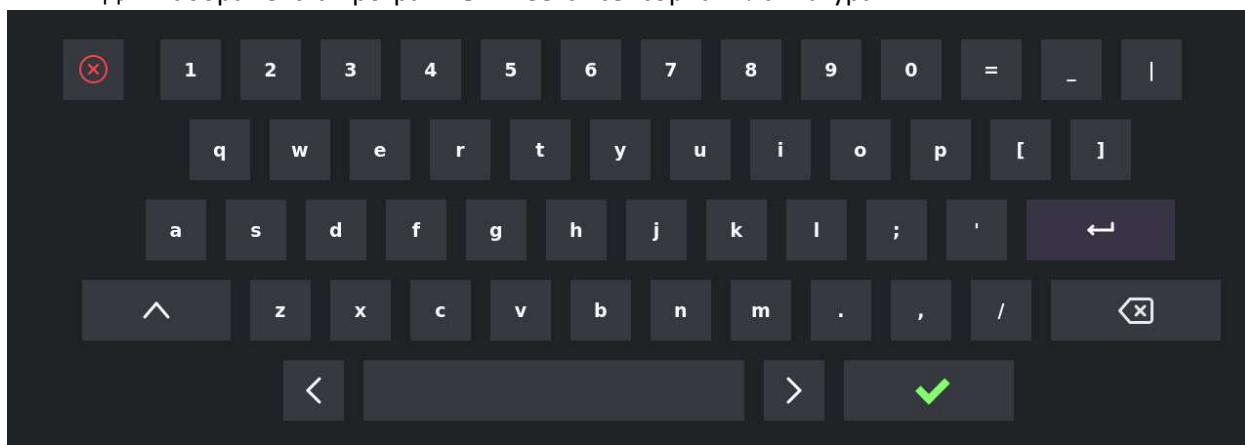
- Разработка с помощью текстового редактора
- Разработка с использованием мастера команд. Мастер команд позволяет формировать текст программы используя



Для работы с файлом программы имеются следующие кнопки: «создать», «сохранить», «сохранить как», «открыть»;

для редактирования текста программы: переместить курсор на строку выше, на строку ниже, на символ вправо, на символ влево, вырезать, копировать, вставить, удалить;

для набора текста программы имеется сенсорная клавиатура.



7.1. Структура программы робота

Программа управления промышленным роботом содержит следующую структуру и синтаксис

- Заголовок программы
PROGRAM Идентификатор;

Пример:

PROGRAM test;

- Секция объявления переменных
VAR
Объявление переменной;
...
Список идентификаторов: Тип;

Пример:

VAR

i, j: integer;

boo: boolean;

r: real;

s1, s2: string;

c: char;

b: byte;

- Секция определения констант
CONST
Определение константы;
...
Идентификатор = Значение;

Пример:

CONST

i = 100;

j = 1.5;

s = 'hello world';

c = 'A';

Константу со значением логического типа объявить нельзя, так как для этого уже есть встроенные константы TRUE и FALSE

- Секция определения потоков
THREADS
Определение потока;
...
Определение потока
THREAD Идентификатор;
BEGIN
Операторы;
END;

Пример:

```
THREADS
THREAD t1;
BEGIN
...
END;
```

- Основная секция

```
BEGIN
Операторы;
END.
```

Пример программы робота, содержащий все основные секции:

```
PROGRAM test;
VAR
  i, j: integer;
  boo: boolean;
  r: real;
  s1, s2: string;
  c: char;
  b: byte;
  j: array [6] of integer;
CONST
  k = 100;
  f = 1.5;
  s = 'hello world';
  v = 'A';
THREADS
  THREAD t1;
  BEGIN
    ...
  END;
BEGIN
  j[1] := 5;
  ...
END.
```

Секции объявления программы и основная секция являются обязательными. Остальные используются при необходимости. Строки отделяются друг от друга символом “;”. После ключевых слов **VAR**, **CONST**, **THREADS**, **BEGIN** символ “;” не ставится. В конце программы обязательно наличие символа “.”.

Секция объявления потоков допускает создание произвольного кол-ва потоков программы. Внутри блока операторов, у потоков запрещается использовать функции движения.

Допустимые типы данных описываются в разделе 6.2.

Между ключевыми словами **BEGIN** и **END** указывается список операторов программы. Подробнее о них в разделе 6.4.

Переменные, используемые в основной секции и в потоках программы, предварительно объявляются в секции **VAR**.

7.2. Типы данных

Языком программирования робота поддерживаются следующие типы данных:

Тип данных	Описание	Минимальное значение	Максимальное значение
boolean	логический	false	true
byte	байтовый	0	255
int	целочисленный	-2 147 483 648	2 147 483 647
float	вещественный	+/- 3.4E-38	+/- 3.4E+38
char	символьный	0	255
string	строковый	0 байт	255 байт
array [N] of ...	Одномерный массив, содержащий N элементов		

В языке программ робота реализовано следующее неявное приведение типов:

Таблица приведения типов

В \ Из	bool	byte	int	float	char	string	array [] of "type"
bool	+	+	+	-	-	+, в "true" или "false"	-
byte	-	+	+	-	+, в ASCII СИМВОЛ	+, в СИМВОЛ	-
int	+	+	+	+	+, в ASCII СИМВОЛ	+, в ЧИСЛО	-
float	-	-	+, с откидыванием действительной части	+	-	+, в ЧИСЛО	-
char	-	+, в код ASCII СИМВОЛА	+, в код ASCII СИМВОЛА	-	+	+	-
string	-	-	-	-	+, первый СИМВОЛ	+	-
array	-	-	-	-	-	-	+, при соответствии типа type

Приведение типов реализовано в операторе присваивания, а также при передаче аргументов в функции.

7.3. Предопределенные имена переменных

Для цифровых входов/выходов определены следующие стандартные имена переменные типа Integer:

- Цифровые входы

DI_0

DI_1

...

DI_N

- Цифровые выходы

DO_0

DO_1

...

DO_N

Переменные цифровых входов автоматически принимают значения при изменении значений соответствующих им физических входов.

Например, при подаче напряжения на цифровой вход 0, значение переменной **DI_0** установится в значение 1.

При записи значений переменных цифровых выходов соответствующие им физические выходы принимают установленные значения.

Например, при записи значения 1 в переменную **DO_0**, цифровой выход 0 принимает значение HIGH.

7.4. Операторы

Операторы программы робота могут быть нескольких типов:

- Простые
- Составные
- Ключевые слова
- Комментарии

К простым операторам можно отнести:

- Оператор присваивания «:=»;
- Математические операторы «+», «-», «*», «/»;
- Логические операторы сравнения «<», «>», «!=», «=»
- Оператор управления выходами робота

SET_DO (0, FALSE);

- Оператор Задержки. Величина задается в мс

DELAY (1000)

- Оператор ожидания

WAIT (DI_0 = 1) - ждем пока вход станет = 1

WAIT (DI_0 = 0) - пока вход = 0 - ждем

К составным операторам можно отнести:

- Оператор условного перехода

IF Выражение **THEN** Оператор
IF Выражение **THEN** Оператор **ELSE** Оператор

Пример:

```
IF (a > 0) THEN  
    s := 'positive'  
ELSE  
    s := 'negative';
```

или

```
IF (x != 0) THEN  
BEGIN  
    a := 0;  
    x := x + 100;  
END;
```

- Оператор цикла

WHILE Выражение **DO** Оператор

Пример:

```
WHILE (A != 100) DO BEGIN  
    T := TRUE;  
    A := A + 1 ;  
END;
```

К ключевым словам, помимо уже указанных, относится:

Задание зоны сглаживания сегментов траектории:

```
BLEND_BEGIN  
BLEND_END;
```

Комментирование кода выполняется путем взятия необходимой строки в фигурные скобки

```
{текст комментария}
```

7.5. Команды движения робота

Системой управления робота предусмотрены следующие типы движения:

- Наискорейшее движение
- Линейное движение
- Движение по окружности
- Движение дополнительными осями

Для непрерывного движения по много сегментной траектории используется сглаживание движений.

Для создания наискорейшего движения используется следующая команда:

- **PTP (X, Y, Z, Rx, Ry, Rz, v, a, smooth, tool, base);**

Альтернативной командой создания движения является:

- **PTP_AXIS (A1, A2, A3, A4, v, a, smooth);**

Для создания линейной траектории:

- **LIN (X, Y, Z, Rx, Ry, Rz, v, a, smooth, tool, base);**

Для движения по окружности:

- **CIRC (X1, Y1, Z1, Rx1, Ry1, Rz1, X2, Y2, Z2, Rx2, Ry2, Rz2, v, a, smooth, tool, base);**

Для движения дополнительной оси:

- **MOVE_AXIS (num, pos, v, a);**

Аргументами функции являются:

X, Y, Z, Rx, Ry, Rz – Декартовы координаты целевой позиции. Ед. изм. Мм для позиции и ° для ориентации. Для движения по окружности, координаты первой точки относятся к точке, лежащей на дуге, второй – конечной точке дуги;

V – Скорость при движении в точку. Ед. изм. Мм/с для декартовых перемещений, % от номинала – для угловых;

A – Ускорение при движении в точку. Ед. изм. Мм/с², для декартовых перемещений, % от номинала – для угловых;

Smooth – величина сглаживания. Ед. изм. Мм для декартовых движений и % от траектории для угловых.

Tool – выбранный инструмент робота

Base – выбранная база робота

Num – номер дополнительной оси.

Pos – целевая позиция. Единицы измерения этого атрибута, а также следующих двух (скорости и ускорения) зависят от типа оси. Для линейного трека это мм (мм/с, мм/с²).

Сгладить можно следующие типы траекторий:

- Линейную с линейной
- Линейную с окружностью
- Окружность с окружностью
- Наискорейшее с наискорейшим

При задании параметра сглаживания отличным от нуля, обязательным является использованием конструкции. У последней команды, находящейся в группе, величина сглаживания должна быть нулевой.

BLEND_BEGIN

...

BLEND_END

7.6. Встроенные функции языка

Системой предусмотрены встроенные функции для обмена данными между роботом и сторонним устройством используя механизм TCP/IP сокетов. Робот выступает клиентом в данном подключении. Для организации обмена данными имеются следующие команды:

- **SOCKET_OPEN (IP address, Port, Name);**
IP address – Адрес стороннего устройства. Пишется в формате '192.168.0.1';
Port – Порт стороннего устройства. Пишется в формате 8888;
Name – Имя создаваемого сокет-клиента. Пишется в формате 'Camera'. Имя сокета используется в дальнейших командах для работы с подключением;
При открытии сокета функция SOCKET_OPEN возвращает True. При невозможности открытия – False;
- **SOCKET_CLOSE (Name);**
Name – имя закрываемого сокета;
- **SOCKET_WRITE_INT32 (Name, Value)**
Name – Имя сокета, для записи данных;
Value – Целочисленное значение передаваемое в сокет;
- **SOCKET_WRITE_FLOAT (Name, Value)**
Name – Имя сокета, для записи данных;
Value – Вещественное значение передаваемое в сокет;
- **SOCKET_WRITE_STRING (Name, Value)**
Name – Имя сокета, для записи данных;
Value – Строковое значение передаваемое в сокет;
- **SOCKET_WRITE_BYTE (Name, Value)**
Name – Имя сокета, для записи данных;
Value – Байтовая величина передаваемая в сокет;
- **SOCKET_READ_INT32 (Name)**
Name – Имя сокета, для чтения данных;
Функция возвращает целочисленное значение, считанное из сокета;
- **SOCKET_READ_FLOAT (Name)**

Name – Имя сокета, для чтения данных;
Функция возвращает вещественное значение, считанное из сокета;

- **SOCKET_READ_INT32_ARRAY (Name, Size)**

Name – Имя сокета, для чтения данных;
Size – кол-во считываемых переменных;
Функция возвращает массив целочисленных значений, считанный из сокета;

- **SOCKET_READ_FLOAT_ARRAY (Name, Size)**

Name – Имя сокета, для чтения данных;
Size – кол-во считываемых переменных;
Функция возвращает массив вещественных значений, считанный из сокета;

- **SOCKET_READ_BYTE (Name)**

Name – Имя сокета, для чтения данных;
Функция возвращает байтовое значение, считанное из сокета;

- **SOCKET_READ_STRING (Name)**

Name – Имя сокета, для чтения данных;
Функция возвращает строковое значение, считанное из сокета. В конец передаваемой строки должен находиться управляющий символ ASCII '\n' (0x0A , 10 в десятичной системе счисления).

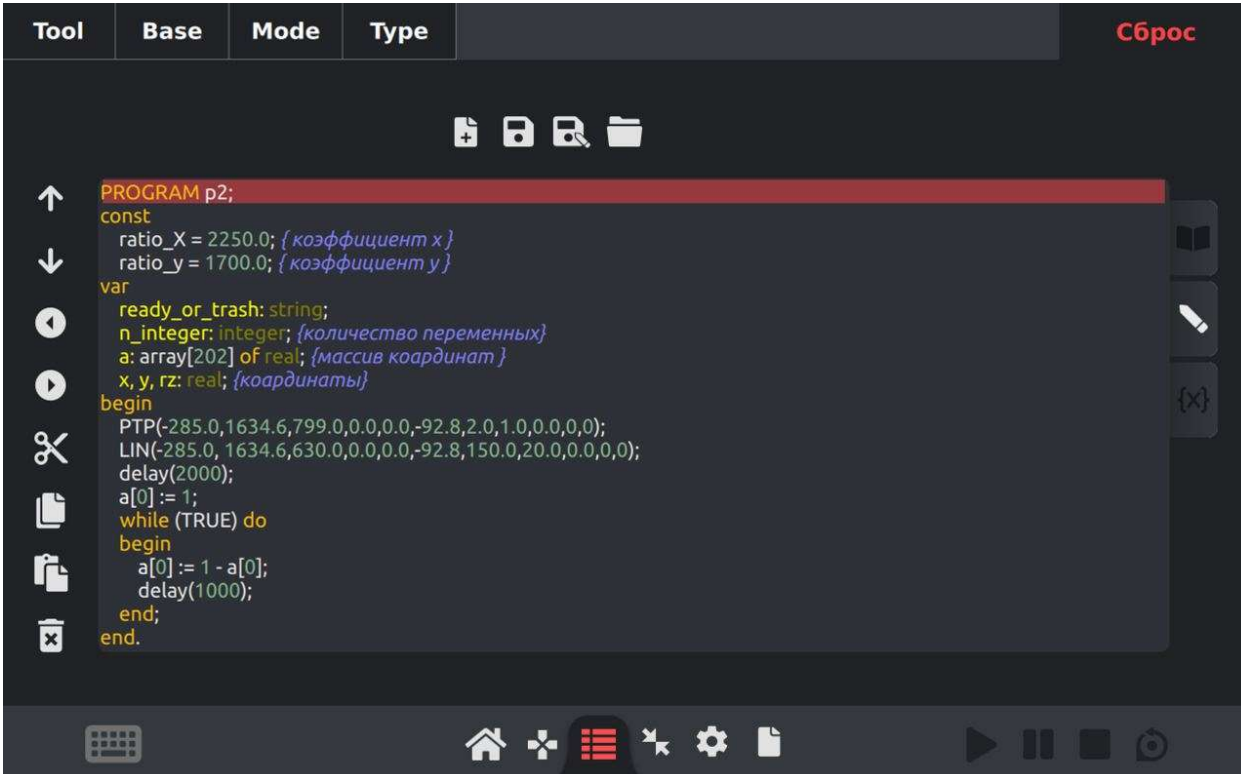
8. Выполнение программы

Перед выполнением программы необходимо убедиться, что робот из своей текущей позиции сможет безопасно проехать в первую точку автоматической программы. Если такое движение не гарантировано, то перед выполнения автоматической программы необходимо выставить робота в ручном режиме.

Для выполнения программы необходимо:

- Переключить робота в автоматический режим
- Нажать кнопку «старт»

После запуска программы произойдет проверка синтаксиса на наличие ошибок. Если эта проверка выполнена успешно, то программа начнет свое построчное выполнение, если нет – соответствующее описание появится в верхней части экрана.



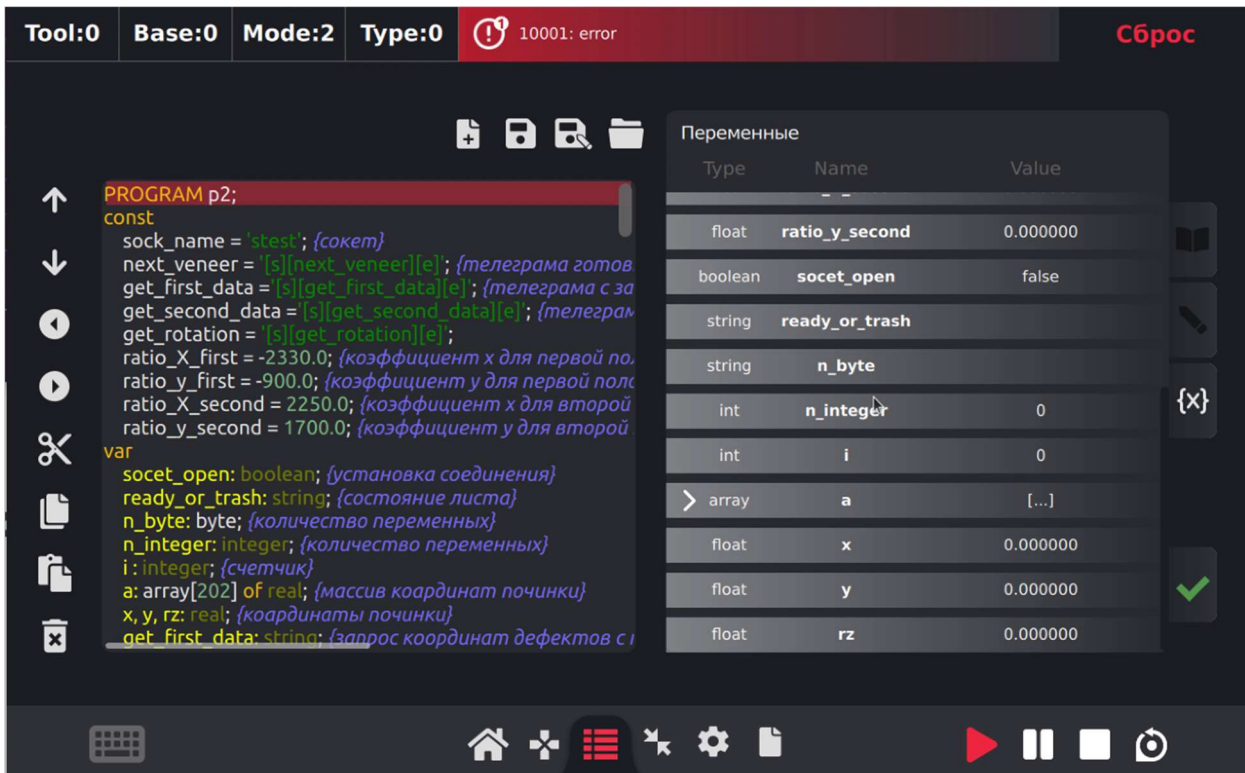
```
PROGRAM p2;
const
  ratio_X = 2250.0; { коэффициент x }
  ratio_y = 1700.0; { коэффициент y }
var
  ready_or_trash: string;
  n_integer: integer; { количество переменных }
  a: array[202] of real; { массив координат }
  x, y, rz: real; { координаты }
begin
  PTP(-285.0,1634.6,799.0,0.0,0.0,-92.8,2.0,1.0,0.0,0,0);
  LIN(-285.0, 1634.6,630.0,0.0,0.0,-92.8,150.0,20.0,0.0,0,0);
  delay(2000);
  a[0] := 1;
  while (TRUE) do
  begin
    a[0] := 1 - a[0];
    delay(1000);
  end;
end.
```

По нажатию кнопки «Пауза» робот закончит выполнение текущей строки и перейдет в режим ожидания, сохраняя текущий шаг программы.

По нажатию кнопки «Стоп» робот мгновенно остановит выполнение программы и сбросит шаг проблемы.

8.1. Отображение состояния переменных

Во время работы программы можно наблюдать актуальные значения пользовательских переменных, перейдя на соответствующую вкладку. В появившемся окне можно наблюдать имя переменной, тип данных и ее значение.



The screenshot shows a debugger interface with a code editor on the left and a variable watch window on the right. The code editor displays the following code:

```
PROGRAM p2;
const
  sock_name = 'stest'; {сокет}
  next_veneer = '[s][next_veneer][e]'; {телеграмма готов}
  get_first_data = '[s][get_first_data][e]'; {телеграмма с за}
  get_second_data = '[s][get_second_data][e]'; {телеграм}
  get_rotation = '[s][get_rotation][e]';
  ratio_X_first = -2330.0; {коэффициент x для первой по}
  ratio_y_first = -900.0; {коэффициент y для первой пол}
  ratio_X_second = 2250.0; {коэффициент x для второй}
  ratio_y_second = 1700.0; {коэффициент y для второй}
var
  socket_open: boolean; {установка соединения}
  ready_or_trash: string; {состояние листа}
  n_byte: byte; {количество переменных}
  n_integer: integer; {количество переменных}
  i: integer; {счетчик}
  a: array[202] of real; {массив координат починки}
  x, y, rz: real; {координаты починки}
  get_first_data: string; {запрос координат дефектов с I
```

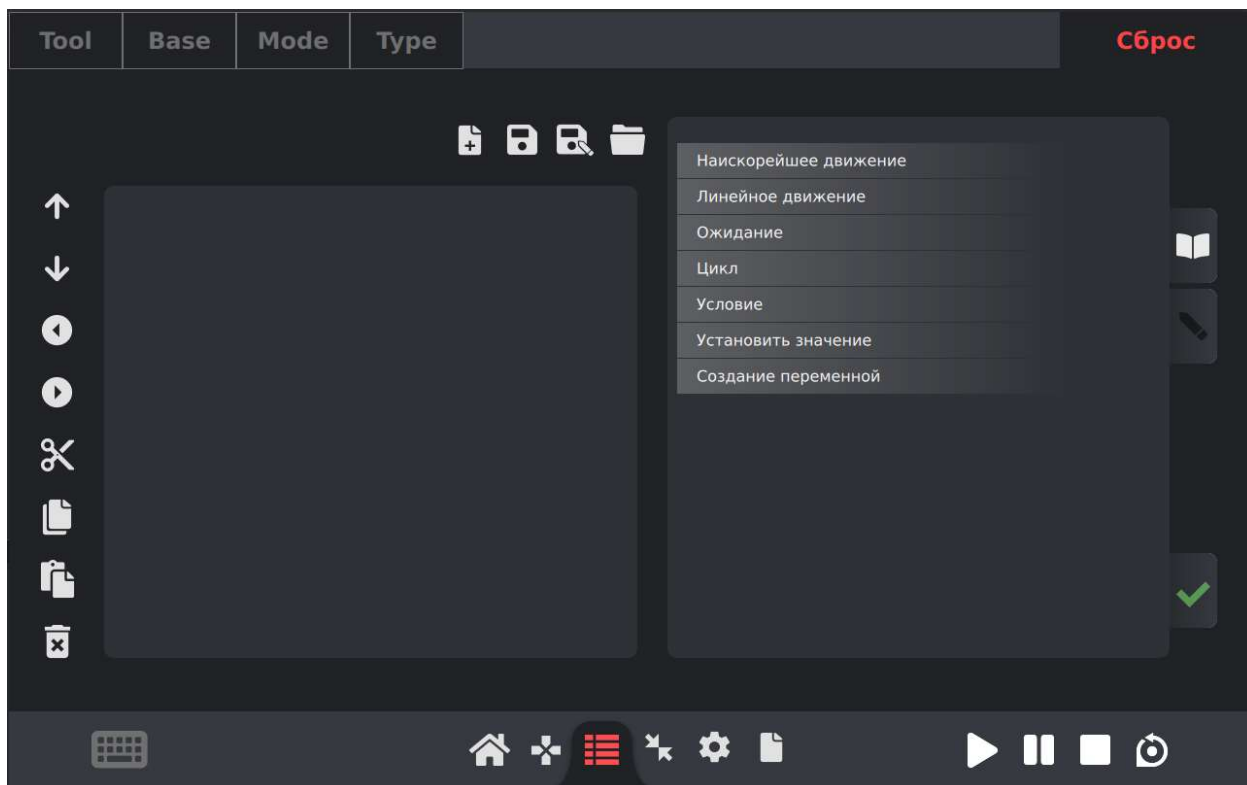
The variable watch window, titled "Переменные", displays the following table:

Type	Name	Value
float	ratio_y_second	0.000000
boolean	socket_open	false
string	ready_or_trash	
string	n_byte	
int	n_integer	0
int	i	0
> array	a	[...]
float	x	0.000000
float	y	0.000000
float	rz	0.000000

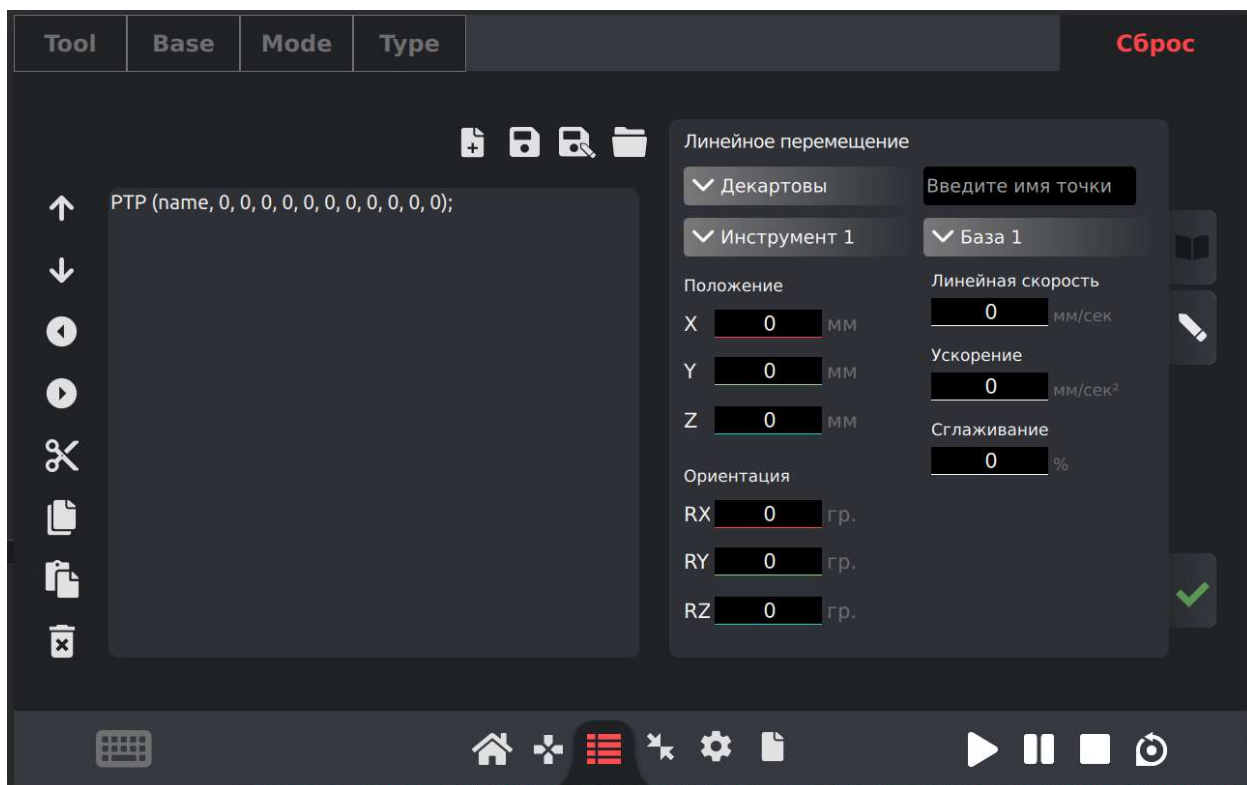
The interface includes a top status bar with "Tool:0", "Base:0", "Mode:2", "Type:0", an error indicator "10001: error", and a "Сброс" button. The bottom toolbar contains icons for keyboard, home, window, search, settings, file, play, pause, and refresh.

9. Программирование с помощью мастера команд

Для упрощения процесса разработки управляющих программ предусмотрен мастер команд. С его помощью возможно создавать программу без непосредственного ввода текста



К примеру, при выборе из мастера команд линейное перемещение у Вас появляется диалоговое окно, в котором выделены все необходимые параметры для задания.



10. Диагностика

Системой управления промышленного робота предусмотрена индикация ошибок, индикация о которых отображается на соответствующем экране пульта управления роботом.

Все ошибки имеют соответствующий код, первый символ которого отвечает за соответствующий блок-системы управления:

1. Ошибки сетевого модуля сервера робота
2. Ошибки подключенных устройств к сети робота.
3. Ошибки расчета траекторий
4. Ошибки интерпретатора языка робота
5. Ошибки среды выполнения программы

Итоговый код = номер блока x 10000 + код ошибки.

Коды ошибок с 1 по 4999 являются аварийными, с 5000 по 9999 – предупредительными.

Подробное описание ошибок и мероприятий по их устранению можно найти в соответствующих руководствах к подключаемому оборудованию.

Пример:

40001 - ошибка с кодом 1 в блоке интерпретатора

45000 - предупреждение с кодом 5000 в блоке интерпретатора

21001 - Ошибка с кодом 1 у первой оси робота

Возникающие ошибки логируются и хранятся на пульте оператора, с глубиной сохранения 1 неделя.

Код ошибки	Описание
20011	Ось 1 OvervoltageError
20021	Ось 1 MainPowerSupplyUndervoltage
20031	Ось 1 MainCircuitPowerSupplyPhaseLossError
20041	Ось 1 OvercurrentError
20051	Ось 1 PowerModuleError
20061	Ось 1 RegenerationCircuitErrorDetectedduringPowerON
20071	Ось 1 InrushCurrentPreventionCircuitError
20081	Ось 1 ServoDriveOverheat
20091	Ось 1 MotorOverheatError
20101	Ось 1 OverloadError
20111	Ось 1 RegenerationOverloadError
20121	Ось 1 RegenerationCircuitError
20131	Ось 1 RegenerationProcessingError
20141	Ось 1 RunawayDetected
20151	Ось 1 EncoderCommunicationsDisconnectionError
20161	Ось 1 EncoderCommunicationsError
20171	Ось 1 ExcessivePositionDeviationError
20181	Ось 1 ExcessiveSpeedDeviationError
20191	Ось 1 ExcessiveSpeedError
20201	Ось 1 AbsoluteValueCleared
20211	Ось 1 PulseOutputOverspeedError
20221	Ось 1 PulseOutputSettingError
20231	Ось 1 FollowingErrorCounterOverflow

20241	Ось 1 GeneralInputAllocationDuplicateError
20251	Ось 1 GeneralOutputAllocationDuplicateError
20261	Ось 1 SoftwareLimitExceeded
20271	Ось 1 FPGAWDTErrors
20281	Ось 1 SystemError
20291	Ось 1 Self-diagnosisError
20301	Ось 1 Non-volatileMemoryDataError
20311	Ось 1 Non-volatileMemoryHardwareError
20321	Ось 1 DriveProhibitionInputError
20331	Ось 1 DriveProhibitionDetected
20341	Ось 1 AbsoluteEncoderCounterOverflowError
20351	Ось 1 EncoderMemoryError
20361	Ось 1 1-rotationCounterError
20371	Ось 1 AbsoluteEncoderMulti-rotationCounterError
20381	Ось 1 AbsolutePositionDetectionError
20391	Ось 1 OverspeedError
20401	Ось 1 MainCircuitTemperatureMonitoringCircuitFailure
20411	Ось 1 FanError
20421	Ось 1 ControlRightReleaseError
20431	Ось 1 SafetyParameterError
20441	Ось 1 SafetyCommunicationsSettingError
20451	Ось 1 FSoESlaveAddressError
20461	Ось 1 SafetyFrameError
20471	Ось 1 SafetyCommunicationsTimeout
20481	Ось 1 EtherCATStateChangeError
20491	Ось 1 EtherCATIllegalStateChangeError
20501	Ось 1 CommunicationsSynchronizationError
20511	Ось 1 SynchronizationError
20521	Ось 1 SyncManagerWDTError
20531	Ось 1 BootstrapStateTransitionRequestError
20541	Ось 1 ErrorStopInput
20551	Ось 1 ESCInitializationError
20561	Ось 1 SynchronizationInterruptionError
20571	Ось 1 SIIVerificationError
20581	Ось 1 ESCError
20591	Ось 1 MailboxSettingError
20601	Ось 1 PDOWDTSettingError
20611	Ось 1 SMEEventModeSettingError
20621	Ось 1 DCSettingError
20631	Ось 1 SynchronizationCycleSettingError
20641	Ось 1 RxPDOSettingError
20651	Ось 1 TxPDOSettingError
20661	Ось 1 RxPDOMappingError
20671	Ось 1 TxPDOMappingError
20681	Ось 1 NodeAddressUpdated
20691	Ось 1 CommandError
20701	Ось 1 ElectronicGearSettingError
20711	Ось 1 FunctionSettingError

20721	Ось 1 MotorNon-conformity
20731	Ось 1 MotorReplacementDetected
20741	Ось 1 BrakeInterlockError
20751	Ось 1 STO
20012	Ось 2 OvervoltageError
20022	Ось 2 MainPowerSupplyUndervoltage
20032	Ось 2 MainCircuitPowerSupplyPhaseLossError
20042	Ось 2 OvercurrentError
20052	Ось 2 PowerModuleError
20062	Ось 2 RegenerationCircuitErrorDetectedduringPowerON
20072	Ось 2 InrushCurrentPreventionCircuitError
20082	Ось 2 ServoDriveOverheat
20092	Ось 2 MotorOverheatError
20102	Ось 2 OverloadError
20112	Ось 2 RegenerationOverloadError
20122	Ось 2 RegenerationCircuitError
20132	Ось 2 RegenerationProcessingError
20142	Ось 2 RunawayDetected
20152	Ось 2 EncoderCommunicationsDisconnectionError
20162	Ось 2 EncoderCommunicationsError
20172	Ось 2 ExcessivePositionDeviationError
20182	Ось 2 ExcessiveSpeedDeviationError
20192	Ось 2 ExcessiveSpeedError
20202	Ось 2 AbsoluteValueCleared
20212	Ось 2 PulseOutputOverspeedError
20222	Ось 2 PulseOutputSettingError
20232	Ось 2 FollowingErrorCounterOverflow
20242	Ось 2 GeneralInputAllocationDuplicateError
20252	Ось 2 GeneralOutputAllocationDuplicateError
20262	Ось 2 SoftwareLimitExceeded
20272	Ось 2 FPGAWDTErrors
20282	Ось 2 SystemError
20292	Ось 2 Self-diagnosisError
20302	Ось 2 Non-volatileMemoryDataError
20312	Ось 2 Non-volatileMemoryHardwareError
20322	Ось 2 DriveProhibitionInputError
20332	Ось 2 DriveProhibitionDetected
20342	Ось 2 AbsoluteEncoderCounterOverflowError
20352	Ось 2 EncoderMemoryError
20362	Ось 2 1-rotationCounterError
20372	Ось 2 AbsoluteEncoderMulti-rotationCounterError
20382	Ось 2 AbsolutePositionDetectionError
20392	Ось 2 OverspeedError
20402	Ось 2 MainCircuitTemperatureMonitoringCircuitFailure
20412	Ось 2 FanError
20422	Ось 2 ControlRightReleaseError
20432	Ось 2 SafetyParameterError
20442	Ось 2 SafetyCommunicationsSettingError

20452	Ось 2 FSoESlaveAddressError
20462	Ось 2 SafetyFrameError
20472	Ось 2 SafetyCommunicationsTimeout
20482	Ось 2 EtherCATStateChangeError
20492	Ось 2 EtherCATIllegalStateChangeError
20502	Ось 2 CommunicationsSynchronizationError
20512	Ось 2 SynchronizationError
20522	Ось 2 SyncManagerWDTError
20532	Ось 2 BootstrapStateTransitionRequestError
20542	Ось 2 ErrorStopInput
20552	Ось 2 ESCInitializationError
20562	Ось 2 SynchronizationInterruptionError
20572	Ось 2 SIIVerificationError
20582	Ось 2 ESCError
20592	Ось 2 MailboxSettingError
20602	Ось 2 PDOWDTSettingError
20612	Ось 2 SMEEventModeSettingError
20622	Ось 2 DCSettingError
20632	Ось 2 SynchronizationCycleSettingError
20642	Ось 2 RxPDOSettingError
20652	Ось 2 TxPDOSettingError
20662	Ось 2 RxPDOMappingError
20672	Ось 2 TxPDOMappingError
20682	Ось 2 NodeAddressUpdated
20692	Ось 2 CommandError
20702	Ось 2 ElectronicGearSettingError
20712	Ось 2 FunctionSettingError
20722	Ось 2 MotorNon-conformity
20732	Ось 2 MotorReplacementDetected
20742	Ось 2 BrakeInterlockError
20752	Ось 2 STO
20013	Ось 3 OvervoltageError
20023	Ось 3 MainPowerSupplyUndervoltage
20033	Ось 3 MainCircuitPowerSupplyPhaseLossError
20043	Ось 3 OvercurrentError
20053	Ось 3 PowerModuleError
20063	Ось 3 RegenerationCircuitErrorDetectedduringPowerON
20073	Ось 3 InrushCurrentPreventionCircuitError
20083	Ось 3 ServoDriveOverheat
20093	Ось 3 MotorOverheatError
20103	Ось 3 OverloadError
20113	Ось 3 RegenerationOverloadError
20123	Ось 3 RegenerationCircuitError
20133	Ось 3 RegenerationProcessingError
20143	Ось 3 RunawayDetected
20153	Ось 3 EncoderCommunicationsDisconnectionError
20163	Ось 3 EncoderCommunicationsError
20173	Ось 3 ExcessivePositionDeviationError

20183	Ось 3 ExcessiveSpeedDeviationError
20193	Ось 3 ExcessiveSpeedError
20203	Ось 3 AbsoluteValueCleared
20213	Ось 3 PulseOutputOverspeedError
20223	Ось 3 PulseOutputSettingError
20233	Ось 3 FollowingErrorCounterOverflow
20243	Ось 3 GeneralInputAllocationDuplicateError
20253	Ось 3 GeneralOutputAllocationDuplicateError
20263	Ось 3 SoftwareLimitExceeded
20273	Ось 3 FPGAWDTErrors
20283	Ось 3 SystemError
20293	Ось 3 Self-diagnosisError
20303	Ось 3 Non-volatileMemoryDataError
20313	Ось 3 Non-volatileMemoryHardwareError
20323	Ось 3 DriveProhibitionInputError
20333	Ось 3 DriveProhibitionDetected
20343	Ось 3 AbsoluteEncoderCounterOverflowError
20353	Ось 3 EncoderMemoryError
20363	Ось 3 1-rotationCounterError
20373	Ось 3 AbsoluteEncoderMulti-rotationCounterError
20383	Ось 3 AbsolutePositionDetectionError
20393	Ось 3 OverspeedError
20403	Ось 3 MainCircuitTemperatureMonitoringCircuitFailure
20413	Ось 3 FanError
20423	Ось 3 ControlRightReleaseError
20433	Ось 3 SafetyParameterError
20443	Ось 3 SafetyCommunicationsSettingError
20453	Ось 3 FSoESlaveAddressError
20463	Ось 3 SafetyFrameError
20473	Ось 3 SafetyCommunicationsTimeout
20483	Ось 3 EtherCATStateChangeError
20493	Ось 3 EtherCATIllegalStateChangeError
20503	Ось 3 CommunicationsSynchronizationError
20513	Ось 3 SynchronizationError
20523	Ось 3 SyncManagerWDTError
20533	Ось 3 BootstrapStateTransitionRequestError
20543	Ось 3 ErrorStopInput
20553	Ось 3 ESCInitializationError
20563	Ось 3 SynchronizationInterruptionError
20573	Ось 3 SIVerificationError
20583	Ось 3 ESCError
20593	Ось 3 MailboxSettingError
20603	Ось 3 PDOWDTSettingError
20613	Ось 3 SMEEventModeSettingError
20623	Ось 3 DCSettingError
20633	Ось 3 SynchronizationCycleSettingError
20643	Ось 3 RxPDOSettingError
20653	Ось 3 TxPDOSettingError

20663	Ось 3 RxPDOMappingError
20673	Ось 3 TxPDOMappingError
20683	Ось 3 NodeAddressUpdated
20693	Ось 3 CommandError
20703	Ось 3 ElectronicGearSettingError
20713	Ось 3 FunctionSettingError
20723	Ось 3 MotorNon-conformity
20733	Ось 3 MotorReplacementDetected
20743	Ось 3 BrakeInterlockError
20753	Ось 3 STO
20014	Ось 4 OvervoltageError
20024	Ось 4 MainPowerSupplyUndervoltage
20034	Ось 4 MainCircuitPowerSupplyPhaseLossError
20044	Ось 4 OvercurrentError
20054	Ось 4 PowerModuleError
20064	Ось 4 RegenerationCircuitErrorDetectedduringPowerON
20074	Ось 4 InrushCurrentPreventionCircuitError
20084	Ось 4 ServoDriveOverheat
20094	Ось 4 MotorOverheatError
20104	Ось 4 OverloadError
20114	Ось 4 RegenerationOverloadError
20124	Ось 4 RegenerationCircuitError
20134	Ось 4 RegenerationProcessingError
20144	Ось 4 RunawayDetected
20154	Ось 4 EncoderCommunicationsDisconnectionError
20164	Ось 4 EncoderCommunicationsError
20174	Ось 4 ExcessivePositionDeviationError
20184	Ось 4 ExcessiveSpeedDeviationError
20194	Ось 4 ExcessiveSpeedError
20204	Ось 4 AbsoluteValueCleared
20214	Ось 4 PulseOutputOverspeedError
20224	Ось 4 PulseOutputSettingError
20234	Ось 4 FollowingErrorCounterOverflow
20244	Ось 4 GeneralInputAllocationDuplicateError
20254	Ось 4 GeneralOutputAllocationDuplicateError
20264	Ось 4 SoftwareLimitExceeded
20274	Ось 4 FPGAWDTErrror
20284	Ось 4 SystemError
20294	Ось 4 Self-diagnosisError
20304	Ось 4 Non-volatileMemoryDataError
20314	Ось 4 Non-volatileMemoryHardwareError
20324	Ось 4 DriveProhibitionInputError
20334	Ось 4 DriveProhibitionDetected
20344	Ось 4 AbsoluteEncoderCounterOverflowError
20354	Ось 4 EncoderMemoryError
20364	Ось 4 1-rotationCounterError
20374	Ось 4 AbsoluteEncoderMulti-rotationCounterError
20384	Ось 4 AbsolutePositionDetectionError

20394	Ось 4 OverspeedError
20404	Ось 4 MainCircuitTemperatureMonitoringCircuitFailure
20414	Ось 4 FanError
20424	Ось 4 ControlRightReleaseError
20434	Ось 4 SafetyParameterError
20444	Ось 4 SafetyCommunicationsSettingError
20454	Ось 4 FSoESlaveAddressError
20464	Ось 4 SafetyFrameError
20474	Ось 4 SafetyCommunicationsTimeout
20484	Ось 4 EtherCATStateChangeError
20494	Ось 4 EtherCATIllegalStateChangeError
20504	Ось 4 CommunicationsSynchronizationError
20514	Ось 4 SynchronizationError
20524	Ось 4 SyncManagerWDTErrror
20534	Ось 4 BootstrapStateTransitionRequestError
20544	Ось 4 ErrorStopInput
20554	Ось 4 ESCInitializationError
20564	Ось 4 SynchronizationInterruptionError
20574	Ось 4 SIIVerificationError
20584	Ось 4 ESCError
20594	Ось 4 MailboxSettingError
20604	Ось 4 PDOWDTSettingError
20614	Ось 4 SMEEventModeSettingError
20624	Ось 4 DCSettingError
20634	Ось 4 SynchronizationCycleSettingError
20644	Ось 4 RxPDOSettingError
20654	Ось 4 TxPDOSettingError
20664	Ось 4 RxPDOMappingError
20674	Ось 4 TxPDOMappingError
20684	Ось 4 NodeAddressUpdated
20694	Ось 4 CommandError
20704	Ось 4 ElectronicGearSettingError
20714	Ось 4 FunctionSettingError
20724	Ось 4 MotorNon-conformity
20734	Ось 4 MotorReplacementDetected
20744	Ось 4 BrakeInterlockError
20754	Ось 4 STO
50011	Ось 1 OverloadWarning
50021	Ось 1 RegenerationOverloadWarning
50031	Ось 1 FanRotationWarning
50041	Ось 1 EncoderCommunicationsWarning
50051	Ось 1 MotorVibrationWarning
50061	Ось 1 CapacitorLifetimeWarning
50071	Ось 1 InrushCurrentPreventionRelayLifetimeWarning
50081	Ось 1 BrakeInterlockOutputRelayLifetimeWarning
50091	Ось 1 LifetimeInformationCorruptionWarning
50101	Ось 1 EncoderLifetimeWarning
50111	Ось 1 AbsoluteEncoderCounterOverflowWarning

50121	Ось 1 DataSettingWarning
50131	Ось 1 CommandWarning
50141	Ось 1 EtherCATCommunicationsWarning
50012	Ось 2 OverloadWarning
50022	Ось 2 RegenerationOverloadWarning
50032	Ось 2 FanRotationWarning
50042	Ось 2 EncoderCommunicationsWarning
50052	Ось 2 MotorVibrationWarning
50062	Ось 2 CapacitorLifetimeWarning
50072	Ось 2 InrushCurrentPreventionRelayLifetimeWarning
50082	Ось 2 BrakeInterlockOutputRelayLifetimeWarning
50092	Ось 2 LifetimeInformationCorruptionWarning
50102	Ось 2 EncoderLifetimeWarning
50112	Ось 2 AbsoluteEncoderCounterOverflowWarning
50122	Ось 2 DataSettingWarning
50132	Ось 2 CommandWarning
50142	Ось 2 EtherCATCommunicationsWarning
50013	Ось 3 OverloadWarning
50023	Ось 3 RegenerationOverloadWarning
50033	Ось 3 FanRotationWarning
50043	Ось 3 EncoderCommunicationsWarning
50053	Ось 3 MotorVibrationWarning
50063	Ось 3 CapacitorLifetimeWarning
50073	Ось 3 InrushCurrentPreventionRelayLifetimeWarning
50083	Ось 3 BrakeInterlockOutputRelayLifetimeWarning
50093	Ось 3 LifetimeInformationCorruptionWarning
50103	Ось 3 EncoderLifetimeWarning
50113	Ось 3 AbsoluteEncoderCounterOverflowWarning
50123	Ось 3 DataSettingWarning
50133	Ось 3 CommandWarning
50143	Ось 3 EtherCATCommunicationsWarning
50014	Ось 4 OverloadWarning
50024	Ось 4 RegenerationOverloadWarning
50034	Ось 4 FanRotationWarning
50044	Ось 4 EncoderCommunicationsWarning
50054	Ось 4 MotorVibrationWarning
50064	Ось 4 CapacitorLifetimeWarning
50074	Ось 4 InrushCurrentPreventionRelayLifetimeWarning
50084	Ось 4 BrakeInterlockOutputRelayLifetimeWarning
50094	Ось 4 LifetimeInformationCorruptionWarning
50104	Ось 4 EncoderLifetimeWarning
50114	Ось 4 AbsoluteEncoderCounterOverflowWarning
50124	Ось 4 DataSettingWarning
50134	Ось 4 CommandWarning
50144	Ось 4 EtherCATCommunicationsWarning
20805	Линейный трек Overcurrent
20815	Линейный трек Overvoltage
20825	Линейный трек Undervoltage

20835	Линейный трек MotorError
20845	Линейный трек RegenerationError
20855	Линейный трек Overload
20865	Линейный трек Overspeed
20875	Линейный трек AbnormalPulseControlCommand
20885	Линейный трек ExcessiveDeviation
20895	Линейный трек EncoderError
20905	Линейный трек AdjustmentError
20915	Линейный трек EmergencyStopActivated
20925	Линейный трек ReverseLimitSwitchError
20935	Линейный трек ForwardLimitSwitchError
20945	Линейный трек IGBTtemperatureError
20955	Линейный трек MemoryError
20965	Линейный трек EncoderOutputError
20975	Линейный трек SerialCommunicationError
20985	Линейный трек SerialCommunicationTimeout
20995	Линейный трек InputPowerPhaseLoss
21005	Линейный трек EarlyWarningForoverLoad
21015	Линейный трек MotorProtectionError
21025	Линейный трек U_V_W_WiringError
21035	Линейный трек Full_closedLoopExcessiveDeviation
21045	Линейный трек DSPFirmwareUpgrade
21055	Линейный трек CANopenDataInitialError
21065	Линейный трек ForwardSoftwareLimit
21075	Линейный трек ReverseSoftwareLimit
21085	Линейный трек EtherCATConnectionError_ServoOff_
21095	Линейный трек NodeGuardingOrHeartBeatError_ServoOff
21105	Линейный трек CanRearWriteError
21115	Линейный трек CANopenError
21125	Линейный трек SafetyTorqueOff
30001	UNKNOWN_ERROR
30002	AXIS_1_OUT_OF_LIMITS
30003	AXIS_2_OUT_OF_LIMITS
30004	AXIS_3_OUT_OF_LIMITS
30005	AXIS_4_OUT_OF_LIMITS
30006	INVERSE_KINEMATIC_C4_NEGATIVE_AND_NOT_ZERO
30007	START_POINT_EQUAL_END_POINT
30008	NEW_COORDINATES_DIFFER_FROM_INITIAL
30009	LINEAR_TRAJECTORY_FIND_TIME_FOR_A_MAX_SOMETHING_WRONG
30010	SOME_ANGLES_ARE_NAN
30011	SINGULAR_MATRIX
30012	UNKNOWN_XYZA4
30013	SOME_TROUBLES_WITH_INTERPOLATE
30014	UNABLE_TO_SYNC
30015	START_VELOCITY_BIGGER_THEN_LIMIT_VELOCITY
30016	IMPOSSIBLE_TO_CALC_DIR_COS
30017	ERROR_IN_BEZIER_BUILDING
30018	ERROR_ABRUPT_VELOCITY_CHANGING

30019	DISCRIMINANT_BELOW_ZERO
30020	NO_ROOTS
30021	ERROR_CANT_FIND_ANGLE_ON_CIRCLE
30022	UNKNOWN_VELOCITY_A4
30023	UNKNOWN_ACCELERATION_A4
30024	TIMING_CALC_ERROR
30025	IMPOSSIBLE_TO_LIMIT_VELOCITY_OR_ACCELERATION
30026	SMOOTHING_ERROR
30027	NO_DIRECTION_FOR_MANUAL_MOTION
30028	INVALID_VALUE_OF_BLEND
30029	EXTERNAL_AXIS_OUT_OF_LIMITS
35001	WARNING_JERK_IS_EXCEEDED
35002	WARNING_LIM_ACCELERATION_EXCEEDS